

# Spatial Coupling of Serially-Concatenated Codes

Wei HOU

Kyoto, Japan

November, 2015



# Abstract

In the dissertation, two kinds of spatially coupled concatenated codes are introduced. The spatial coupling is a kind of design method to structure channel codes whose decoding performance approach to theoretical limit. A spatially coupled code is obtained by associating multiple identical base codes. More specifically, spatial coupling is to line up all of base codes, and then to exchange parts of parity relations of several adjacent base codes. The concatenated codes, serially concatenating an outer code and an inner code, are considered as base codes. With the concatenated base codes, more relax code construction can be built. Spatial coupling technique lets designer pick up outer and inner codes for some practical aspects (such as simple encoding implementation, flexible construction, rate adjustability, etc.) without considering performance issue. Based on this feature, two classes of codes are structured to fit different communication requirements.

One kind of codes is obtained by spatially coupling base codes consisting of repeater-combiner outer code and convolutional inner code. The codes, called spatially coupled repeater-combiner-convolutional (SC-RCC) codes, have simple encoder realization. And de-

coding analysis and computer simulation show that their decoding performances are nearer to theoretical limit than these of conventional spatially coupled codes on additive white Gaussian noise channels (AWGNC) and Gaussian multiple access channels (MAC). In particular, SC-RCC codes can achieve good decoding performance with not very long code length. This makes it possible for SC-RCC codes to be adopted in the future communication systems.

Another kind of spatially coupled concatenated codes is for rate-compatible communication problem that demand transmitter and receiver to process different rate codes with a single encoder and single decoder. A family of rate-compatible codes consists of multiple different rate codes, in which the higher rate member codes are embedded into the lower rate member codes. The rate-compatible spatially coupled low-density parity-check (SC-LDPC) codes are obtained by spatially coupling base codes including an LDPC outer code for spatial coupling and rate-adjustable repeat-accumulate (RA) inner code. The proposed codes are called RA-extended SC-LDPC codes, since they also can be considered as LDPC codes with RA-extensions. They achieve arbitrary rate in wide and continuous real number interval. Decoding analysis shows the all of member codes in a RA-extended SC-LDPC code family have decoding performance near to theoretical limit on binary erasure channels (BEC). In particular, in the low rate region, RA-extended SC-LDPC codes perform better than conventional codes. This makes the codes are promising in low rate coding workplace, e.g., non-orthogonal multiple access system discussing for next generation mobile communications.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Noisy Channel Coding Problem . . . . .	1
1.2	Channel Capacity . . . . .	3
1.2.1	Binary Erasure Channel . . . . .	4
1.2.2	Additive White Gaussian Noise Channel . . . . .	5
1.3	Spatial Coupling Technique . . . . .	6
1.4	Contributions . . . . .	8
1.4.1	SC-RCC codes for Point-to-Point Communication Systems . . . . .	10
1.4.2	SC-RCC Codes on Multiple Access Channels . . . . .	11
1.4.3	RA-Extended SC-LDPC Codes for Rate-Compatible Communication Systems . . . . .	12

<b>2</b>	<b>Elements of Graph-Based Codes</b>	<b>13</b>
2.1	Bipartite Graph Codes: LDPC Codes . . . . .	14
2.1.1	Basic Concept . . . . .	14
2.1.2	Belief Propagation Decoding . . . . .	17
2.1.3	BP Thresholds Analysis . . . . .	19
2.2	Spatial Coupling of Bipartite Graph Codes . . . . .	24
2.2.1	Protograph LDPC Code Ensembles . . . . .	24
2.2.2	Spatially Coupled Code Ensembles . . . . .	27
2.3	Generalized Graph-Based Code Ensembles . . . . .	31
2.3.1	Realization Graph of Convolutional Codes . . . . .	31
2.3.2	Generalized Graph-Based Code Ensembles: An Example . . . . .	34
2.4	Summary . . . . .	35
<b>3</b>	<b>Spatially Coupled Repeater-Combiner-Convolutional Codes</b>	<b>37</b>
3.1	Introduction . . . . .	38
3.2	Construction of SC-RCC Codes . . . . .	40
3.3	Iterative Decoding Analysis . . . . .	43
3.4	Numerical Results . . . . .	46

3.4.1	BP thresholds . . . . .	46
3.4.2	Simulation of Finite Length Codes . . . . .	50
3.5	Summary . . . . .	52
<b>4</b>	<b>SC-RCC Codes on Multiple Access Channels</b>	<b>53</b>
4.1	Introduction to MAC Coding . . . . .	53
4.2	SC-RCC Codes on MAC . . . . .	58
4.3	Iterative Detection-Decoding Analysis . . . . .	61
4.3.1	EXIT Function for Local Processing . . . . .	61
4.3.2	BP Thresholds . . . . .	63
4.4	Numerical Results . . . . .	65
4.5	Summary . . . . .	67
<b>5</b>	<b>Rate-Compatible RA-Extended SC-LDPC Codes</b>	<b>71</b>
5.1	Introduction . . . . .	71
5.2	Construction of RA-Extended SC-LDPC Codes . . . . .	74
5.3	Iterative Decoding Analysis . . . . .	78
5.3.1	Potential Threshold Analysis . . . . .	78

5.3.2	Potential Thresholds of Base Code Ensembles . . . . .	80
5.3.3	BP Thresholds of RA-Extended SC-LDPC Code Ensembles . . . . .	83
5.4	Numerical Results . . . . .	84
5.5	Summary . . . . .	89
<b>6</b>	<b>Concluding Remarks</b>	<b>91</b>

# List of Figures

1.1	Basic digital communication system established by C. E. Shannon. . . . .	2
1.2	Binary erasure channel with parameter $\epsilon$ . . . . .	4
1.3	Additive white Gaussian noise channel. . . . .	6
1.4	The diagram of (a) a single code and (b) spatially coupling multiple identical single codes. . . . .	7
2.1	Tanner graph for a 1/3-rate LDPC code in Example 2.1. . . . .	15
2.2	(a) Message passing from the $j$ -th variable node to the $i$ -th check node, and (b) message passing from the $i$ -th check node to the $j$ -th variable node. . . .	17
2.3	Illustrations of (a) probability density functions and (b) mutual information passing on variable and check nodes. . . . .	19
2.4	Illustration of (a) protograph with $n = 3$ and $n - k = 2$ , (b) $M = 3$ copies of the protograph, and (c) derived Tanner graph. . . . .	25

2.5	The erasure probability messages passing on protograph of RA code ensemble.	27
2.6	(a) $L(= 12)$ copies of protograph $(3, 6)$ -LDPC code ensemble, (b) the protograph of SC-LDPC code ensemble $\mathcal{L}(3, 6, L = 12)$ .	28
2.7	Erasure probability distribution diagram of SC-LDPC ensemble $\mathcal{L}(d_v, d_c, L, w)$ after $l$ round iterations.	30
2.8	The realization of convolution function $g(D) = a(D)/b(D)$ .	32
2.9	(a) The encoder, (b) trellis chart representation, and (c) coding realization graph of systematic convolutional code $G(D) = [1, \frac{1+D^2}{1+D+D^2}]$ .	33
2.10	(a) An example of generalized protograph with rate-1 convolutional code nodes and (b) its realization graph with $M = 3$ .	34
3.1	(a) Encoder and (b) factor graph of concatenated code $\mathcal{B}(q, \mathcal{C})$ , and (c) factor graph of SC-RCC code $\mathcal{S}(q, L, \mathcal{C})$ with $q = 3$ .	42
3.2	The message passing at the $i$ -th coupled position of $\mathcal{S}(q, L, \mathcal{C}, w)$ .	44
3.3	Rate versus threshold curves of SC-RCC codes with 4-state convolutional component codes $\mathcal{C}(1/7)$ , $\mathcal{C}(3/7)$ , and $\mathcal{C}(5/7)$ .	47
3.4	Rate versus threshold curves of SC-RCC codes with 8-state convolutional component codes $\mathcal{C}(13/15)$ , $\mathcal{C}(15/17)$ , and $\mathcal{C}(11/17)$ .	48

3.5	Rate versus threshold curves of the SC-RCC codes $\mathcal{S}(3, L, \mathcal{C}, 3)$ with $\mathcal{C}(5/7)$ and $\mathcal{C}(15/17)$ , SC-RA codes $\mathcal{S}(q, L, \mathcal{C}(1/3), w = q)$ with $q = 3, 4, 6$ , and SC-LDPC codes $\mathcal{L}(dl, dr, L, w = dl)$ with $(dl, dr) = (3, 6)$ and $(4, 8)$ . . . . .	49
3.6	Error correction performance of SC-RCC codes $\mathcal{S}(3, 30, \mathcal{C}(5/7))$ with $n = 15628, 21828$ , and $28028$ , SC-RA codes $\mathcal{S}(3, 30, \mathcal{C}(1/3))$ with $n = 27964$ and $\mathcal{S}(6, 30, \mathcal{C}(1/3))$ with $n = 28020$ , and SC-LDPC code $\mathcal{L}(dl = 3, dr = 6, L = 30)$ with $n = 28200$ . . . . .	51
4.1	(a) 2-user Gaussian MAC model, and (b) system diagram of coding on Gaussian MAC. . . . .	54
4.2	The boundary MAC-APR and boundaries of BP-APR for different codes. . .	56
4.3	(a) Factor graph of SC-RCC code $\mathcal{S}(q = 3, L = 6, \mathcal{C})$ at user $t, t = 1, 2$ and (b) illustration of the permutation and transmission of two users' symbol vectors. . . . .	59
4.4	EXIT illustration of (a) MACC, (b) EC, (c) SPC, and (d) CCC. . . . .	61
4.5	The Factor graph representation of joint iterative detection-decoding analysis at the $l$ -th coupling position. . . . .	64
4.6	BP-APR of $\mathcal{S}(3, 30, \mathcal{C}(1/5))$ on Gaussian MAC. . . . .	66
4.7	BER versus $h$ of the SC-RCC codes $\mathcal{S}(3, 30, \mathcal{C}(1/3))$ , $\mathcal{S}(3, 30, \mathcal{C}(1/5))$ , $\mathcal{S}(3, 30, \mathcal{C}(1/7))$ , and $\mathcal{S}(3, 30, \mathcal{C}(5/7))$ with $M = 100$ , and SC-LDPC codes $\mathcal{L}(dl = 3, dr = 6, L = 30)$ with $M = 104$ . . . . .	68

4.8	BER versus $h$ of the SC-RCC codes $\mathcal{S}(3, 30, \mathcal{C}(1/3))$ , $\mathcal{S}(3, 30, \mathcal{C}(1/5))$ , $\mathcal{S}(3, 30, \mathcal{C}(1/7))$ , and $\mathcal{S}(3, 30, \mathcal{C}(5/7))$ with $M = 200$ , and SC-LDPC codes $\mathcal{L}(dl = 3, dr =$ $6, L = 30)$ with $M = 207$ . . . . .	69
5.1	(a) Protograph of $\mathcal{B}(J, K, q, \alpha)$ and Tanner graph of $\alpha$ -extension, and (b) connection diagram at $i$ -th coupling position of $\mathcal{C}(J, K, L, w, q, \alpha)$ . . . . .	75
5.2	Protograph of $\mathcal{B}(J, K, q, \alpha)$ with virtual copies of variable nodes in $(J, K)$ -part	81
5.3	Potential gaps of $\mathcal{B}(3, K, q, \alpha)$ and BP gaps of $\mathcal{C}(3, K, 100, 5, q, \alpha)$ with $K =$ $6, 8, 10$ and $12$ . . . . .	85
5.4	Rates versus BP thresholds of codes $\mathcal{C}(3, K, 100, 5, q, \alpha)$ with $K = 6$ and $K = 12$ , $\mathcal{C}'(3, K = 12, 100, 5, q, \alpha)$ , and TET-extended rate-compatible code families with rate increments $0.1$ and $0.05$ . . . . .	86

# List of Tables

5.1	Rate-compatible code family of $\mathcal{C}(3, 6, 100, 5, q, \alpha)$ . . . . .	87
5.2	Rate-compatible code family of $\mathcal{C}(3, 12, 100, 5, q, \alpha)$ . . . . .	88



# Chapter 1

## Introduction

### 1.1 Noisy Channel Coding Problem

Today, digital communication systems are widely used in our lives. The frequently used examples include mobile phone, digital television, wired and wireless internet connections, and so on. Generally, each example fits into a common digital communication framework established by C. E. Shannon in his seminal paper, *A Mathematical Theory of Communication*, 1948 [1]. This framework is illustrated in Figure 1.1.

At transmitter of a digital communication system, there are four necessary processors, including source, source encoder, channel encoder, and modulator. The source digitizes its message, such as speech, audio, data, etc., and transmits the digitized message to a sink via a noisy physical channel. For guaranteeing efficiency of the transmission, a source encoder

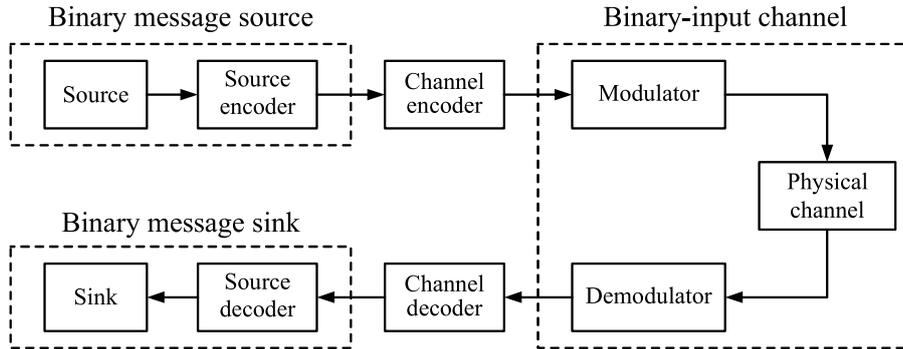


Figure 1.1: Basic digital communication system established by C. E. Shannon.

compress digital message into fewer bits representation. For guaranteeing reliability of the transmission, a channel encoder adds redundancy to protect the transmitted bits over a physical channel with noise, distortion, and interference. The output of channel encoder is called by codeword. Before transmitting codeword over physical channel, a modulator converts the codeword into analog signals suitable for transmission. At receiver, the original message is recovered by corresponding invert processors, including demodulator, channel decoder, source decoder, and sink.

In Shannon's seminal paper [1], the digital communication problem can be decomposed into two separate problems: source coding problem and noisy channel coding problem. In the dissertation, we mainly focus on the latter, *noisy channel coding problem*. In order to describe the channel coding problem more clearly, the digital communication can be simplified as binary message source/sink, channel encoder/decoder, and binary-input channel. The simplified framework is also shown in Figure 1.1.

To be mathematically more precise, the output of the binary message source is consid-

ered as a stochastic model. By the noisy channel coding theorem [1], if the channel coding rate  $R$  is less than the channel capacity  $C$ , then error-free transmission is possible. The channel capacity  $C$  depends on the noise of channel. The goal of channel coding research is to design capacity-approaching channel codes.

## 1.2 Channel Capacity

Mathematically, we use capital letters to represent random variables, and use corresponding lower cases to represent realizations. Let  $X$  and  $Y$  be the random variables of channel input and output, and  $x$  and  $y$  be realizations of  $X$  and  $Y$ , respectively. When  $X$  and  $Y$  are discrete random variables, the mutual information between  $X$  and  $Y$  is written as

$$I(X; Y) = H(Y) - H(Y|X) \quad (1.1)$$

where the entropy of  $Y$  and the conditional entropy of  $Y$  given  $X$  are given by

$$H(Y) = - \sum_y \Pr(y) \log_2 \Pr(y) \quad (1.2)$$

and

$$H(Y|X) = - \sum_x \sum_y \Pr(x, y) \log_2 \Pr(y|x) \quad (1.3)$$

respectively. Then, the channel capacity is defined as

$$C = \max_{\Pr(x)} I(X; Y). \quad (1.4)$$

For a binary-input channel in Figure 1.1, the channel capacity is the maximum mutual information over the channel input probability distribution. When a channel code has code

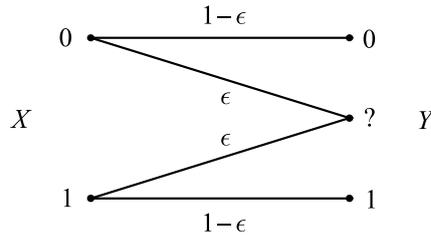


Figure 1.2: Binary erasure channel with parameter  $\epsilon$ .

rate  $R < C$ , the error-free communication is achievable.

In channel coding researches, the binary-input channels are usually considered as some simplified mathematical models [2] [3]. Although these simplified models cannot reflect all features of real channels, they make problem become more concise, clear and accurate. In practice use, the designs based on mathematical models also can perform as expected. Now, let us introduce two frequently used channel models.

### 1.2.1 Binary Erasure Channel

The binary erasure channel (BEC) is the simplest channel model introduced by Elias, 1954 [4]. Generally, a channel code can be easily analyzed on BEC, and the most of analyzed properties hold on other channel models [3].

The BEC model is illustrated in Figure 1.2. The channel input and output represented by  $X$  and  $Y$ ,  $X \in \{0, 1\}$  and  $Y \in \{0, ?, 1\}$ . The sign ‘?’ represent that nothing is known. For the BEC with parameter  $\epsilon$ , a bit of information is completely discarded with erasure probability  $\epsilon$  or error-freely transmitted to receiver with probability  $1 - \epsilon$ . The capacity of

the BEC with erasure  $\epsilon$  is [6]

$$C_{BEC}(\epsilon) = 1 - \epsilon. \quad (1.5)$$

The channel capacity means that information can be transmitted reliably over the BEC at a most rate of  $1 - \epsilon$  bits per channel use.

Since channel codes are usually design at fixed rates, we consider coding rate  $R = C$ . Thus, we have alternative target  $\epsilon^{\text{Sh}} = 1 - R$ . The  $\epsilon^{\text{Sh}}$  is called Shannon limit that is the largest possible channel erasure probability such that the error-free decoding is possible. If a channel coding scheme with rate  $R$  guarantees that error-free decoding can be achieved when  $\epsilon$  approaches to Shannon limit  $\epsilon^{\text{Sh}}$ , we say the channel codes are capacity-approaching.

### 1.2.2 Additive White Gaussian Noise Channel

The additive white Gaussian noise channel (AWGNC) is the most common channel model of digital communication system. It does not account for some practical influencing factors, such as fading, frequency selectivity, interference, and so on. However, AWGNC is a tractable models which are useful for investigating the behavior of a channel code before these practical influencing factors are considered.

The AWGNC model is illustrated in Figure 1.3. Let discrete binary input  $X \in \{\pm 1\}$  and additive white Gaussian noise (AWGN)  $Z \sim \mathcal{N}(0, \sigma^2)$ . We have channel output  $Y = X + Z$ . Here,  $\mathcal{N}(0, \sigma^2)$  presents Gaussian distribution with mean 0 and variance  $\sigma^2$ . The

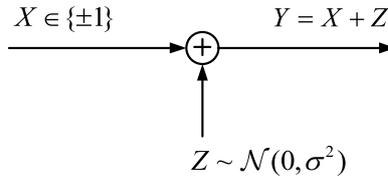


Figure 1.3: Additive white Gaussian noise channel.

channel capacity of the AWGNC is given by [5] [6]

$$C_{AWGNC}(\sigma) = \frac{1}{2} \sum_{x=\pm 1} \int_{-\infty}^{+\infty} p(y|x) \log_2 \left( \frac{p(y|x)}{p(y)} \right) dy \quad (1.6)$$

where  $p(y|x = \pm 1) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y \mp 1)^2}{2\sigma^2}\right)$  and  $p(y) = \frac{1}{2}p(y|x = +1) + \frac{1}{2}p(y|x = -1)$ .

In coding design, we focus on the smallest possible signal-to-noise-ratio (SNR) when coding rate  $R = C$ . The SNR is frequently measured by  $E_b/N_0$ , where  $E_b$  is the average energy per information bit and  $N_0/2 = \sigma^2$  is the 2-sided power spectral density of the noise [2]. The smallest  $E_b/N_0$  limit of guaranteeing error-free decoding is called as Shannon limit of  $E_b/N_0$  on AWGNC. For example, Shannon limit of 1/2-rate channel codes is 0.187 dB. When  $E_b/N_0 \geq 0.187$  dB, error-free decoding is possible.

### 1.3 Spatial Coupling Technique

In Shannon's seminal paper [1], he gave the theoretical limit of channel codes with error-free decoding on noisy channels. However, he do not provide any practical coding scheme to achieve the limit. This is a challenge for the upcoming researchers.

In the past semi-century, central objective was to structure codes that have decoding

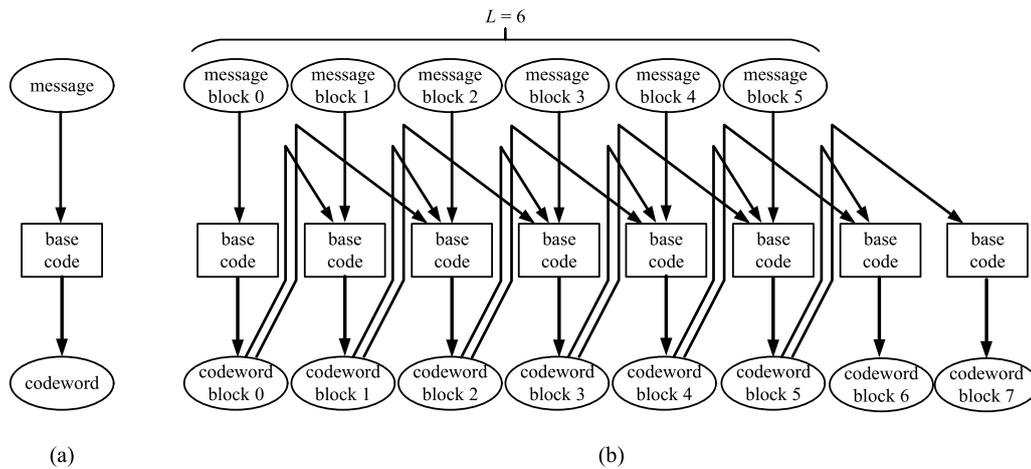


Figure 1.4: The diagram of (a) a single code and (b) spatially coupling multiple identical single codes.

performance near to Shannon limit on frequently used channels [7]. There are some effective schemes are presented, such as convolutional codes [4] and famous Viterbi decoding algorithm [8], turbo codes and their iterative decoding [9] [10], low-density parity-check (LDPC) codes [11] via degree-distribution optimization [12], and so on. These schemes become the base stones of modern coding theory.

Recently, a spatial coupling technique is presented to structure capacity-approaching codes [13]. Different from conventional schemes, the spatially coupled codes are obtained by associating multiple identical base codes. Spatially coupled codes universally well-perform on various channels due to their distinct chaining decoding [14], and thus become a potential coding scheme for future communication systems.

Figure 1.4(a) shows the diagram of a base code, which encode input message to a

codeword. In fact, the base code can be any conventional code. Figure 1.4(b) illustrates the construction of spatially coupled code. The message is divided into multiple blocks that are encoded by coupled coding processor. The output codeword block of each base code depends on not only its input message block but also some codeword blocks generated by several adjacent base codes.

Let us observe the chaining construction of spatially coupled codes. Since those base codes at two ends process the least message, they have more stronger decoding capability than that of base codes at middle positions. After error-free decoding are achieved at two ends, those base codes at adjacent positions of two ends become new ends. When the coupled chain is enough long, the rate of a spatially coupled code is approximately equal to the rate of corresponding base code. Under the chaining decoding mechanism, the spatially coupled code can achieve near-Shannon-limit performance, though the base code may be not very good. We will give the details of spatial coupling technique in Chapter 2.

## 1.4 Contributions

Through research of recent years, spatial coupling technique is shown to be a effective method of structuring good codes on various channels. One of the advantages of spatially coupled codes is that they are less demanding in base code. In other words, a base code without any optimization usually leads to a capacity-approaching spatially coupled code. This feature make it easy to design codes for some applications.

Our work mainly focus on spatial coupling of serially-concatenated codes. Serially-concatenated codes are a kind of conventional channel codes, which are obtained by concatenating an outer code with an inner code. Under spatial coupling, we have more relax conditions for component codes. We pick up the two component codes only based on their basic characteristics, such as rate variability, encoding simplification, decoding effectiveness, and so on. These selected characteristics will guarantee the resulted spatially coupled codes to fit different communication requirements. The good decoding performance will be guaranteed by spatial coupling technique.

Based on spatial coupling of serially-concatenated codes, we proposed two kinds of codes:

1. spatially coupled Repeater-Combiner-Convolutional (SC-RCC) Codes,
2. repeat-accumulate extended spatially coupled low-density parity-check (RA-extended SC-LDPC) codes.

With the two kinds of codes, we investigated spatially coupled serially-concatenated codes on three different communication scenarios:

1. point-to-point communication systems,
2. multiple access channels,
3. rate-compatible communication systems.

Next, we are going to briefly introduce our main contributions by different communication scenarios.

### 1.4.1 SC-RCC codes for Point-to-Point Communication Systems

We first structure spatially coupled codes for point-to-point communication systems. In research field of channel coding, an important evaluation of a code is decoding bit error ratio (BER) performance on AWGNC. Although the conventional spatially coupled codes are proven to be capacity-approaching on AWGNC, a large code length usually is demanded to achieve near-Shannon-limit BER performance. However, since a large code length means a large system cost, such as a big time delay, a large memory requirement, a large decoding complexity, and so on, the spatially coupled codes haven't been adopted by existing communication system. This motivate us to find out new spatially coupled codes that achieve near-Shannon limit BER performance with as possible as short code length.

We structure a kind of SC-RCC codes that are obtained by coupling multiple identical concatenated codes. The concatenated base codes consisting of a block outer code (a repetition code followed by a single-parity code) concatenated with a convolutional inner code of infinite impulse response. Because of the simple outer code and diverse inner code, the proposed codes have simple encoding implementation and flexible design space.

We employ extrinsic information transfer (EXIT) functions to analyse the iterative decoding threshold of proposed codes on AWGNC. The threshold help us to pick up the best

one from all considered codes. Numerical results demonstrate that the proposed codes have better BER performances than conventional spatially coupled coding schemes. In particular, the proposed codes with a rate of about 0.5 and length 15628 have a BER of  $10^{-5}$  at  $E_b/N_0 = 1.13$  dB that is less than 1 dB away from the Shannon limit. This code length may therefore satisfy the practical requirements of communication systems.

### 1.4.2 SC-RCC Codes on Multiple Access Channels

Beside point-to-point systems, channel coding also used in multiple-user communication systems for guaranteeing reliability of transmission. We consider a multiple-user system with multiple transmitters and a single receiver. The channel model is multiple access channel (MAC) [6]. On Gaussian MAC, there exist an achievable power region for given coding rates. Theoretically, with any power allocation in the achievable power region, the receiver may perfectly rebuild all messages of transmitters. However, conventional coding scheme cannot achieve all possible power allocations. It is necessary to design a coding scheme for specific power allocation.

Spatially coupled codes were proven to achieve all power allocation points in achievable power region, when code length is infinite. The property motivate us to investigate the performance of spatially coupled codes with finite code length on MAC.

We apply our proposed SC-RCC codes to Gaussian MAC. The results of EXIT functions analysis and BER simulation show that our proposed codes approximately achieve all

possible points in achievable power region with finite code length.

### 1.4.3 RA-Extended SC-LDPC Codes for Rate-Compatible Communication Systems

In some practical communication system, rate-compatible codes are usually used to adapt changing channel condition with various available coding rates. A family of rate-compatible codes consists of a set of member codes with different rates, in which the higher rate member codes are embedded into the lower rate codes, and all the member codes can be processed by a single encoder and a single decoder. The design problem of rate-compatible is to guarantee all of these embedded member codes capacity-approaching.

We structure RA-extended SC-LDPC codes by spatially coupling a kind of concatenated codes whose outer code is a typical base code for coupling and inner code is a simple rate-compatible repeat-accumulate (RA) code. The base codes provide simple construction and rate compatibility. Spatial coupling technique guarantees that all member codes are capacity-approaching. We analyse a performance indicator of the base code, potential threshold. The fact that potential thresholds with various rates are near to Shannon limits means that RA-extended SC-LDPC codes with various rate are capacity-approaching. The prediction is confirmed by numerical results. Compared with conventional rate-compatible coding schemes based on spatial coupling, our codes achieved arbitrary rates in continuous interval and perform better in the low rate region.

# Chapter 2

## Elements of Graph-Based Codes

In this chapter, we briefly introduce some channel codes based on graph. We first review bipartite graph codes: low-density parity-check (LDPC) codes. About LDPC codes, we describe their basic concept, graph representation, iterative decoding, and ensemble threshold analysis. We then demonstrate an effective method, spatial coupling, to structure bipartite graph codes. Finally, we show generalized graph-based codes. All of knowledge introduced in this chapter are important elements of our work in subsequent chapters.

## 2.1 Bipartite Graph Codes: LDPC Codes

### 2.1.1 Basic Concept

As the most typical of graph-based codes, low-density parity-check (LDPC) codes, which were invented by Gallager in 1960 [11], are a kind of block codes. An LDPC code with rate  $R = k/n$  is given by the null space of an  $(n - k) \times n$  non-singular parity-check matrix  $\mathbf{H} \in \{0, 1\}^{(n-k) \times n}$  that has a low density of 1s. In parity-check matrix  $\mathbf{H}$ , the number of 1s in each column is called by column weight, and the number of 1s in each row is called by row weight. If all column and row weights are both constant, the LDPC code is regular; on the contrary, the LDPC code is irregular.

Based on graph theory, parity-check matrix  $\mathbf{H}$  of an LDPC code corresponds to a bipartite graph [15]. Thus an LDPC code can be represented by a bipartite graph. Since this graph representation is proposed by R. M. Tanner, it is usually called Tanner graph. The two types of nodes in the bipartite graphs are called the variable nodes and the check nodes. Variable nodes are transmitted into channel. Consider an LDPC code with  $(n - k) \times n$  parity-check matrix  $\mathbf{H}$ . When the element  $h_{ij} = 1$  in  $\mathbf{H}$ , the  $i$ -th check node and the  $j$ -th variable node are connected. Similarly, when the element  $h_{ij} = 0$  in  $\mathbf{H}$ , the  $i$ -th check node and the  $j$ -th variable node are non-connected. Here,  $i = 0, 1, 2, \dots, n - k - 1$  and  $j = 0, 1, 2, \dots, n - 1$ .

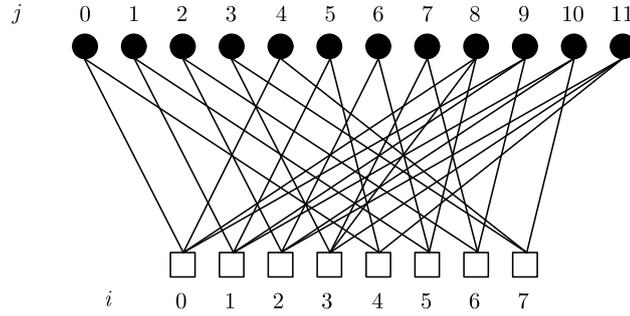


Figure 2.1: Tanner graph for a 1/3-rate LDPC code in Example 2.1.

**Example 2.1** Consider an LDPC code with  $8 \times 12$  parity-check matrix:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \quad (2.1)$$

Let  $\mathbf{c}$  be arbitrary codeword of the LDPC code,  $\mathbf{c} \in \{0, 1\}^{12}$ . We have constraint  $\mathbf{c}\mathbf{H}^T = \mathbf{0}$ , where  $\mathbf{H}^T$  is transposition of  $\mathbf{H}$ . The code has rate  $R = 1/3$ . Its corresponding Tanner graph representation is shown in Figure 2.1. The filled circles and blank squares are variable and check nodes, respectively. There are 8 check nodes (labelled by  $i = 0, 1, 2, \dots, 7$ ) and 12 variable nodes (labelled by  $j = 0, 1, 2, \dots, 11$ ) in the Tanner graph. The fact that the  $i$ -th check node and the  $j$ -th variable node are connected (or non-connected) means  $h_{ij} = 1$  (or

$h_{ij} = 0$ ) in  $\mathbf{H}$ . □

In Tanner graph, the number of edges connected to a variable (or check) node is called variable (or check) node degrees, which correspond to column (or row) weights in parity-check matrix  $\mathbf{H}$ . For *irregular* LDPC codes, all of variable and check node degrees are usually described by degree-distribution polynomials [16], denoted by  $\lambda(x)$  and  $\rho(x)$ , respectively. The polynomials are

$$\lambda(x) = \sum_i \lambda_i x^{i-1} \quad (2.2)$$

$$\rho(x) = \sum_i \rho_i x^{i-1} \quad (2.3)$$

where  $\lambda_i$  and  $\rho_i$  are the fractions of edges that connect to variable and check nodes of degree  $i$ , respectively.

An irregular LDPC code with  $\lambda(x)$  and  $\rho(x)$  is denoted by  $(\lambda, \rho)$ -LDPC code. A set of all possible irregular LDPC codes is called  $(\lambda, \rho)$ -LDPC code *ensemble*. The rate of  $(\lambda, \rho)$ -LDPC ensemble is given by

$$R = 1 - \frac{\int_0^1 \rho(x) dx}{\int_0^1 \lambda(x) dx}. \quad (2.4)$$

Specially, when  $\lambda(x) = x^{d_v-1}$  and  $\rho(x) = x^{d_c-1}$ , the corresponding LDPC codes are regular. Denoted by the set of regular codes  $(d_v, d_c)$ -LDPC code ensemble.

**Example 2.2** The LDPC code depicted in Example 2.1 is a specific realization in code ensemble with  $\lambda(x) = \frac{4}{7}x + \frac{3}{7}x^2$  and  $\rho(x) = \frac{3}{7}x^2 + \frac{4}{7}x^3$ . □

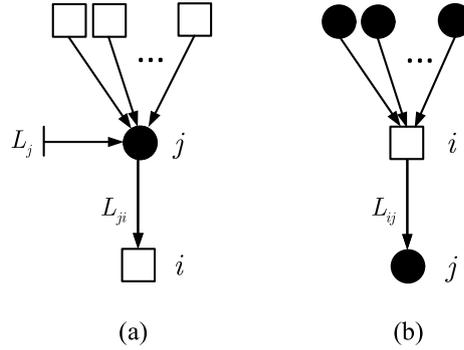


Figure 2.2: (a) Message passing from the  $j$ -th variable node to the  $i$ -th check node, and (b) message passing from the  $i$ -th check node to the  $j$ -th variable node.

### 2.1.2 Belief Propagation Decoding

An LDPC code is obtained by constructing a Tanner graph that satisfies given degree-distribution polynomials  $\lambda(x)$  and  $\rho(x)$ . Its decoding is also based on Tanner graph. The frequently used LDPC decoder is the belief propagation (BP) decoder [2].

In decoding process, the BP decoder processes each variable node and each check node with *a posteriori* probability (APP) decoding. The iterative information exchange occurs between all of variable nodes and all of check nodes that are two types of component decoders. In precisely, the output extrinsic information of all variable (or check) nodes are input into their connected check (or variable) nodes as *a priori* information.

Before iterative update, we initialize each message passing from the  $j$ -th variable node to the  $i$ -th check node with  $L_{ji}^{(0)} = L_j$ ,  $j = 0, 1, 2, \dots, n-1$  and  $i = 0, 1, 2, \dots, n-k-1$ . Here,

$$L_j = \log \frac{\Pr(y_j|x_j = 0)}{\Pr(y_j|x_j = 1)} \quad (2.5)$$

is the belief message of channel value at the  $j$ -th variable node, where  $x_j$  and  $y_j$  are transmitted bit and received value at the  $j$ -th variable node. For BEC, when  $y_j = 0, 1$ , and ‘?’,  $L_j = +\infty, -\infty$ , and  $0$ , respectively [2]. For AWGNC with modulation mapping  $\{0, 1\} \rightarrow \{+1, -1\}$ ,  $L_j = 2y_j/\sigma^2$ , where  $\sigma^2$  is an estimate of channel noise variance [2].

Consider the  $l$ -th round iteration,  $l = 1, 2, 3, \dots, l_{\max}$ , where  $l_{\max}$  is the maximum iterative round. As shown in Figure 2.2(a), the message passing from the  $j$ -th variable node to the  $i$ -th check node is given by [2]

$$L_{ji}^{(l)} = L_j + \sum_{i' \in N(j)/i} L_{i'j}^{(l)} \quad (2.6)$$

where  $N(j)$  expresses index set of those nodes connected to the  $j$ -th variable node, and  $N(j)/i$  expresses set  $N(j)$  removed element  $i$ . As shown in Figure 2.2(b), the message passing from the  $i$ -th check node to the  $j$ -th variable node is given by [2]

$$L_{ij}^{(l)} = 2 \tanh^{-1} \left( \prod_{j' \in N(i)/j} \tanh (L_{j'i}^{(l-1)}/2) \right). \quad (2.7)$$

where  $N(i)$  expresses index set of those nodes connected to the  $i$ -th check node, and  $N(i)/j$  expresses set  $N(i)$  removed element  $j$ .

When  $l = l_{\max}$ , the total belief message at each variable node is calculated by [2]

$$L_j^{\text{total}} = L_j + \sum_{i' \in N(j)} L_{i'j}^{(l_{\max})}. \quad (2.8)$$

From each  $L_j^{\text{total}}$ ,  $j = 0, 1, 2, \dots, n-1$ , the decision of each transmitted bit is obtained by

$$\hat{x}_j = \begin{cases} 0, & L_j^{\text{total}} > 0, \\ 1, & \text{else.} \end{cases}$$

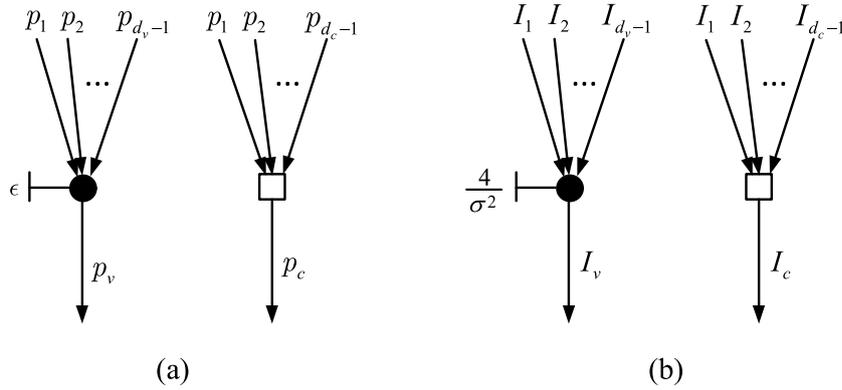


Figure 2.3: Illustrations of (a) probability density functions and (b) mutual information passing on variable and check nodes.

### 2.1.3 BP Thresholds Analysis

For evaluating iterative decoding performance of LDPC code ensembles, we calculate a theoretical limit under BP decoding, called BP threshold. Usually, there are two methods to determine LDPC code ensembles' BP thresholds. They are density evolution (DE) analysis [16] and extrinsic information transfer (EXIT) analysis [17] [18]. Both methods are based on an assumption of infinite code length that guarantees no circle in Tanner graph and accurate average belief message.

#### 2.1.3.1 DE Analysis

The DE analysis is based on principle of LDPC decoder. The difference is that the DE analysis update the average probability density function (*pdf*) of all passing messages instead of each specific belief message. In the decision process, a final *pdf* is used to judge error-free

decoding successful or not successful.

We derive DE analysis on BEC with channel parameter  $\epsilon$ . On BEC, the *pdfs* of all messages are erasure probabilities passing between variable and check nodes. Consider variable and check nodes with degrees  $d_v$  and  $d_c$ , respectively. The output erasure probabilities of variable (check) node relates to all input erasure probabilities. As shown in Figure 2.3(a), the input erasure probabilities of variable node are  $p_1, p_2, \dots, p_{d_v-1}$ , the input erasure probabilities of check node are  $p_1, p_2, \dots, p_{d_c-1}$ .

Then, the output can be calculate by

$$p_v = \epsilon p_1 p_2 \dots p_{d_v-1} \quad (2.9)$$

for variable node and

$$p_c = 1 - (1 - p_1)(1 - p_2) \dots (1 - p_{d_c-1}) \quad (2.10)$$

for check node.

From the basic calculation of variable and check nodes, we can obtain the DE update equation for LDPC code ensembles. Let the average erasure probabilities passing from variable node to check node be  $p$ . At the  $l$ -round update, we have

$$p^{(l)} = \epsilon \left( 1 - (1 - p^{(l-1)})^{d_c-1} \right)^{d_v-1} \quad (2.11)$$

for  $(d_v, d_c)$ -regular LDPC code ensemble and

$$p^{(l)} = \epsilon \lambda (1 - \rho (1 - p^{(l-1)})) \quad (2.12)$$

for  $(\lambda, \rho)$ -irregular LDPC code ensemble, where nodes degree-distributions  $\lambda(x) = \sum_i \lambda_i x^{i-1}$  and  $\rho(x) = \sum_i \rho_i x^{i-1}$ . By updating (2.11) or (2.12) with the initial erasure probabilities message  $p^{(0)} = \epsilon$ , we can determine code ensemble's BP threshold

$$\epsilon^{\text{BP}} \triangleq \sup\{\epsilon \in [0, 1] : p^{(l)}(\epsilon) \xrightarrow{l \rightarrow \infty} 0\}. \quad (2.13)$$

DE update equation (2.11) or (2.12) is built on BEC. On AWGNC, channel characteristic is not erasure probability. To determine BP thresholds, DE update equation should be rebuilt [3] [16].

### 2.1.3.2 EXIT Analysis

In principle, the EXIT analysis can be considered as another evolution analysis that uses mutual information between transmitted random variable  $X$  and passed message  $L$  to replace *pdfs*. For BEC, the mutual information is given by  $I = 1 - p$ . Thus, DE and EXIT analysis are completely consistent with each other on BEC. For AWGNC, besides infinite code length assumption, the EXIT analysis depended on another assumption: the *pdfs* of passed belief messages in (2.6) and (2.7) are Gaussian function. This assumption is approximately hold and greatly reduces the analysis complexity on AWGNC.

For brevity's sake, we just show the basic calculation of mutual information passing for variable and check nodes on AWGNC with noise variance  $\sigma^2$ . Similar to DE analysis, consider variable and check nodes with degree  $d_v$  and  $d_c$ , respectively. As shown in Figure 2.3(b), input *a priori* information of variable node are  $I_1, I_1, \dots, I_{d_v-1}$ , input *a priori* information of

check node are  $I_1, I_1, \dots, I_{d_c-1}$ . Then, the output information can be calculated by [2]

$$I_v = J \left( \sqrt{\frac{4}{\sigma^2} + \sum_{i=1}^{d_v-1} [J^{-1}(I_i)]^2} \right) \quad (2.14)$$

for variable node and

$$I_c = 1 - J \left( \sqrt{\sum_{i=1}^{d_c-1} [J^{-1}(1 - I_i)]^2} \right) \quad (2.15)$$

for check node, where  $J$ -function is given by [2]

$$J(x) = 1 - \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}x} \exp \left( -\frac{(r - x^2/2)^2}{2x^2} \right) \log_2 \left( 1 + e^{-r} \right) dr.$$

From the basic calculation of variable and check nodes, we can obtain the EXIT update equation for LDPC code ensembles. At the  $l$ -round update, we have [2]

$$I_v^{(l)} = J \left( \sqrt{\frac{4}{\sigma^2} + (d_v - 1) [J^{-1}(I_c^{(l)})]^2} \right) \quad (2.16)$$

$$I_c^{(l)} = 1 - J \left( \sqrt{(d_c - 1) [J^{-1}(1 - I_v^{(l-1)})]^2} \right) \quad (2.17)$$

for  $(d_v, d_c)$ -regular LDPC code ensemble and

$$I_v^{(l)} = \sum_i^{d_v-1} \lambda_i J \left( \sqrt{\frac{4}{\sigma^2} + (i - 1) [J^{-1}(I_c^{(l)})]^2} \right) \quad (2.18)$$

$$I_c^{(l)} = \sum_i^{d_c-1} \rho_i \left( 1 - J \left( \sqrt{(i - 1) [J^{-1}(1 - I_v^{(l-1)})]^2} \right) \right) \quad (2.19)$$

for  $(\lambda, \rho)$ -irregular LDPC code ensemble, where  $\lambda_i$  and  $\rho_i$  are degree-distributions coefficients.

By updating (2.16)-(2.17) or (2.18)-(2.19) with  $I_c^{(0)} = 0$ , we can determine regular or irregular LDPC code ensembles' BP threshold

$$\sigma^{\text{BP}} \triangleq \sup \{ \sigma \in (0, +\infty) : I_v^{(l)}(\sigma) \xrightarrow{l \rightarrow \infty} 1 \}. \quad (2.20)$$

Compared with DE analysis, EXIT analysis is a more effective method to determine BP threshold on AWGNC. This is because EXIT function costs lower calculation complexity than *pdf* that is obtained by complicated convolution operation [2].

### 2.1.3.3 Degree-Distribution Optimization of Code Ensembles

By above BP threshold analysis of LDPC code ensembles, code's iterative decoding performance can be evaluated. It is obvious that the BP threshold depends on the degree-distribution of ensemble. Thus, the problem of constructing a capacity-approaching code can be formulated as a degree-distribution optimization problem [12].

Consider  $(\lambda, \rho)$ -irregular LDPC ensemble with rate  $R$  on  $\text{BEC}(\epsilon)$ . Shannon limit of the code ensemble is  $\epsilon^{\text{Sh}} = 1 - R$ . Denoted by BP threshold gap to Shannon limit  $G^{\text{BP}}$ . The degree-distribution optimization problem is described by

$$\begin{aligned} \min_{\lambda(x), \rho(x)} \quad & G^{\text{BP}} = \epsilon^{\text{Sh}} - \epsilon^{\text{BP}} \\ \text{s.t.} \quad & \frac{\int_0^1 \rho(x) dx}{\int_0^1 \lambda(x) dx} = R. \end{aligned}$$

By working with the problem, one can design a code ensemble that has near-Shannon-limit BP threshold.

The degree-distribution optimization is a kind of effective method to structure a good channel code [19]. However, in some specific communication schemes, e.g., rate-compatible coding-transmission scheme that require multiple codes with related construction and different rates are capacity-approaching, the degree-distribution optimization cannot provide

a global optimal solution. In our work, we employ alternative method, spatial coupling technique, to structure capacity-approaching code ensembles.

## 2.2 Spatial Coupling of Bipartite Graph Codes

### 2.2.1 Protograph LDPC Code Ensembles

Before we get into spatially coupled codes, we first cover fundamentals of protograph LDPC codes [20]. Protograph is a small bipartite graph, which is used to construct LDPC codes as a design blueprint. The two types of nodes in a protograph are also variable and check nodes. LDPC code ensemble with rate  $R = k/n$  can be obtained from a protograph with  $n$  variable nodes and  $n - k$  check nodes. From a protograph, we obtain a Tanner graph by copy  $M$  times the protograph and permute (or interleave) those edges connected corresponding variable and check nodes in all copies. The code length is  $nM$ .

**Example 2.3** Figure 2.4(a) shows a protograph with 3 variable nodes and 2 check nodes. To copy the protograph  $M = 3$  times and permute those edges in dash ellipses in Figure 2.4(b), then a Tanner graph of LDPC code with rate  $R = 1/3$  and code length 9 is derived (see Figure 2.4(c)). □

Although the protograph of a code ensemble reflects the degree-distributions of variable and check nodes, the protograph-based ensemble is just a subset of the ensemble with corresponding degree-distributions. Therefore, when determining the BP threshold,

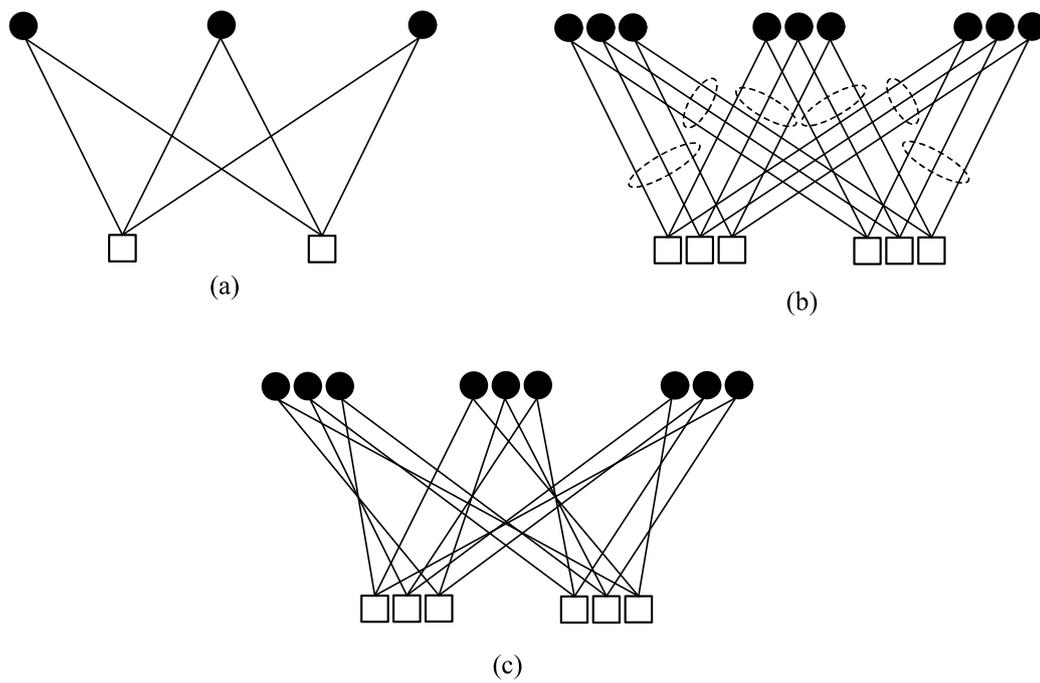


Figure 2.4: Illustration of (a) protograph with  $n = 3$  and  $n - k = 2$ , (b)  $M = 3$  copies of the protograph, and (c) derived Tanner graph.

a protograph-based DE analysis is more scientifically accurate than the DE based on degree-distributions [21]. We demonstrate the DE analysis of protograph ensemble by the following example.

**Example 2.4** Consider a protograph shown in Figure 2.5. There are two classes of variable node sets that are labelled by  $V_1$  and  $V_2$ , respectively. The check node set is labelled by  $C$ . In fact, the protograph corresponds a repeat-accumulate (RA) code ensemble. Let us analyse BP threshold of the ensemble on BEC with channel erasure  $\epsilon$ . Let  $p_{v,1}$ ,  $p_{v,2}$ ,  $p_{c,1}$ , and  $p_{c,2}$  be erasure probability messages passing from in  $V_1$  to  $C$ ,  $V_2$  to  $C$ ,  $C$  to  $V_1$ , and  $C$  to  $V_2$ , respectively. Based on principle of DE, at the  $l$ -th round iteration, we have

$$p_{v,1}^{(l)} = \epsilon \left( p_{c,1}^{(l)} \right)^2 \quad (2.21)$$

$$p_{v,2}^{(l)} = \epsilon p_{c,2}^{(l)} \quad (2.22)$$

$$p_{c,1}^{(l)} = 1 - \left( 1 - p_{v,2}^{(l-1)} \right)^2 \quad (2.23)$$

$$p_{c,2}^{(l)} = 1 - \left( 1 - p_{v,1}^{(l-1)} \right) \left( 1 - p_{v,2}^{(l-1)} \right). \quad (2.24)$$

By substituting (2.23) and (2.24) into (2.21) and (2.22), we have the DE update equations

$$p_{v,1}^{(l)} = \epsilon \left( 1 - \left( 1 - p_{v,2}^{(l-1)} \right)^2 \right)^2 \quad (2.25)$$

$$p_{v,2}^{(l)} = \epsilon \left( 1 - \left( 1 - p_{v,1}^{(l-1)} \right) \left( 1 - p_{v,2}^{(l-1)} \right) \right). \quad (2.26)$$

By updating (2.25) and (2.26) with the  $p_{v,1}^{(0)} = p_{v,2}^{(0)} = \epsilon$ , we can determine BP threshold of the RA code ensemble by

$$\epsilon_{\text{RA}}^{\text{BP}} \triangleq \sup \{ \epsilon \in [0, 1] : \mathbf{p}_v^{(l)}(\epsilon) \xrightarrow{l \rightarrow \infty} \mathbf{0} \} \quad (2.27)$$

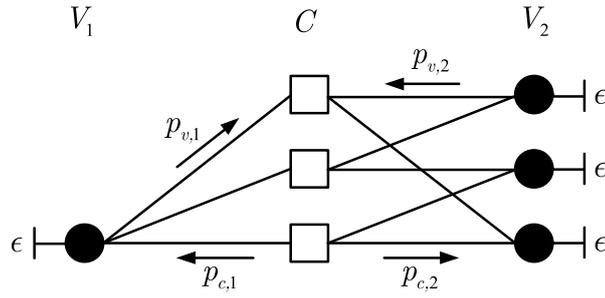


Figure 2.5: The erasure probability messages passing on protograph of RA code ensemble.

where  $\mathbf{p}_v^{(l)} = \{p_{v,1}^{(l)}, p_{v,2}^{(l)}\}$ . □

## 2.2.2 Spatially Coupled Code Ensembles

By using spatial coupling technique mentioned in Section 1.3, one can structure a kind of capacity-approaching channel codes, called spatially coupled codes [13]. Now, with spatially coupled LDPC (SC-LDPC) code ensembles, for example, let us present the spatial coupling technique.

The SC-LDPC code ensembles are obtained by associating the protographs of  $L$  identical LDPC code ensembles, base code ensembles. For the sake of simplicity, we introduce  $(d_v, td_v)$ -regular LDPC as base code ensembles and corresponding SC-LDPC code ensembles, denoted by  $\mathcal{L}(d_v, td_v, L)$ , where  $t$  is integer and  $t \geq 2$ . Figure 2.6(a) shows  $L = 12$  protographs of  $(3, 6)$ -regular LDPC ensembles, and Figure 2.6(b) shows the protograph of SC-LDPC ensemble  $\mathcal{L}(3, 6, L = 12)$ . The  $d_v$  edges emitted from variable nodes in the  $(d_v, td_v)$ -protograph  $i$  are reconnected to check nodes in the protographs  $i$  to  $i + d_v - 1$ ,

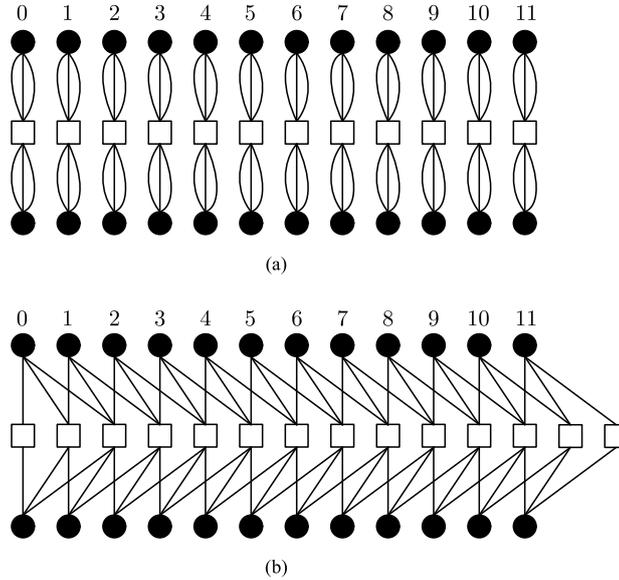


Figure 2.6: (a)  $L(= 12)$  copies of protograph  $(3, 6)$ -LDPC code ensemble, (b) the protograph of SC-LDPC code ensemble  $\mathcal{L}(3, 6, L = 12)$ .

where  $i = 0, 1, \dots, 11$ . There are  $d_v - 1$  additional check nodes to guarantee all edges are connected. The rate of  $\mathcal{L}(d_v, td_v, L)$  is

$$R = \left(1 - \frac{1}{t}\right) - \frac{d_v - 1}{tL}. \quad (2.28)$$

When  $L \gg d_v$ ,  $R \approx 1 - 1/t$ .

The exact BP threshold of  $\mathcal{L}(d_v, td_c, L)$  can be determined by DE analysis similar to that in Example 2.4. However, the DE update of  $\mathcal{L}(d_v, td_c, L)$  is run on many different edges so that their equations are complicated. Therefore, we also demonstrate a random version of spatially coupled ensemble that have simpler DE equations [13].

Consider the more general  $(d_v, d_c)$ -regular LDPC ensemble. Randomly couple  $L$  copies

of its protograph. The  $d_v$  edges emitted from variable nodes in the protograph  $i$  are *randomly* reconnected to check nodes in the protographs  $i$  to  $i + w - 1$ , where  $i = 0, 1, \dots, L - 1$  and  $d_v \leq w < L$ . The random version of the SC-LDPC ensembles are denoted by  $\mathcal{L}(d_v, d_c, L, w)$ . With random connection, there are  $w - 1$  possible additional check nodes to guarantee all edges are connected. The rate of  $\mathcal{L}(d_v, d_c, L, w)$  is [13]

$$R = \left(1 - \frac{d_v}{d_c}\right) - \frac{d_v}{d_c} \cdot \frac{w + 1 - 2 \sum_{i=0}^w \left(\frac{i}{w}\right)^{d_c}}{L}. \quad (2.29)$$

When  $L \gg w$ ,  $R \approx 1 - d_v/d_c$ .

For DE of  $\mathcal{L}(d_v, d_c, L, w)$  ensembles, all of the *pdfs* of messages passing into certain variable and check nodes are processed with their average. On BEC, let  $p_i$  be the erasure probability message emitted from a variable node in the  $i$ -th coupling position. At the  $l$ -th round iteration, the DE update equation is given by [13]

$$p_i^{(l)} = \epsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} p_{i+j-k}(l-1)\right)^{d_c-1}\right)^{d_v-1}. \quad (2.30)$$

By updating (2.30) with  $p_i = \epsilon$  for  $i = 0, 1, \dots, L - 1$  and  $p_i = 0$  for otherwise, the BP threshold of  $\mathcal{L}(d_v, d_c, L, w)$  can be determine. Specially, one can calculate the BP threshold of  $\mathcal{L}(d_v, d_c, L, w = d_v)$  as an approximate result of  $\mathcal{L}(d_v, d_c, L)$ .

S. Kudekar *et al.* [13] proven that, when  $L$  is large, the BP threshold of  $\mathcal{L}(d_v, d_c, L, w)$  approaches to Shannon limit of  $(d_v, d_c)$ -regular LDPC ensemble. In fact, when  $L$  is large, the most coupling positions of  $\mathcal{L}(d_v, d_c, L, w)$  are  $(d_v, d_c)$ -regular LDPC ensemble except a few positions at two terminals (see Figure 2.6(b)).

Figure 2.7 shows an erasure probability distribution diagram of SC-LDPC ensemble

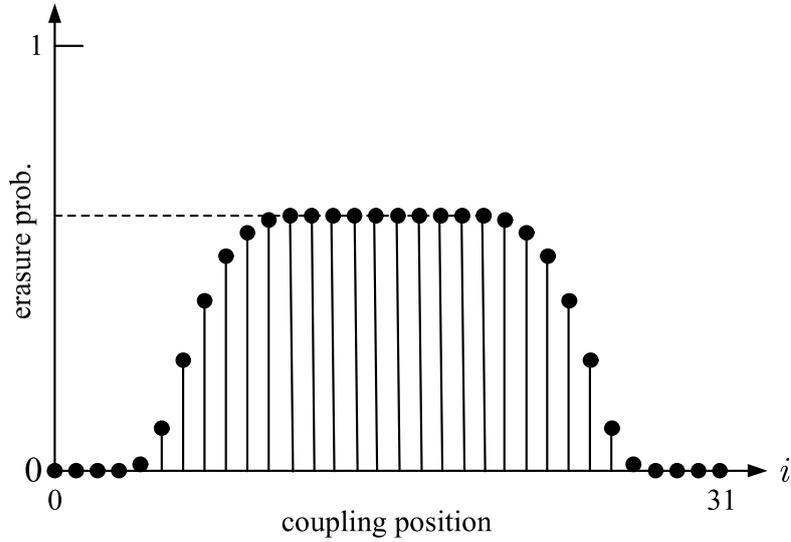


Figure 2.7: Erasure probability distribution diagram of SC-LDPC ensemble  $\mathcal{L}(d_v, d_c, L, w)$  after  $l$  round iterations.

$\mathcal{L}(d_v, d_c, L, w)$  after  $l$  round iterations in the process of DE. In the diagram, we see that the erasure probabilities at two terminals first converge to 0. This is because the lower check node degrees at two terminals provide more powerful decoding capability. And then, those positions with lower erasure probability will help their neighbouring positions to decline erasure probabilities. This chaining process will improve global decoding capability.

Compared with  $\mathcal{L}(d_v, d_c, L, w)$  ensemble, their uncoupled version  $(d_v, d_c)$ -regular ensemble has the worse decoding performance. Consider the DE of  $(d_v, d_c)$ -regular ensemble. Without effect of two terminals, the erasure probability output from variable node may stop at a non-zero fixed point (middle positions in Figure 2.7) when channel noise parameter  $\epsilon$  is same.

## 2.3 Generalized Graph-Based Code Ensembles

In the previous description, we review bipartite graph code ensembles and their constructions based on protograph. In fact, the concept of protograph can be extended to more general case. In other word, besides variable and check nodes, other types node constraints are possible. In this section, we introduce a concept of convolutional code node.

### 2.3.1 Realization Graph of Convolutional Codes

The convolutional codes were invented by P. Elias in 1955 [4]. A convolutional encoder is a linear convolution processor with transfer matrix  $G(D)$  [2]. Let input information sequence be  $\mathbf{u} \in \{0, 1\}^K$ . We denote  $\mathbf{u}$  by polynomial form

$$u(D) = u_0 + u_1D + u_2D^2 + \dots + u_{K-1}D^{K-1}.$$

Then the output sequence polynomial

$$v(D) = u(D)G(D). \quad (2.31)$$

Specifically, Figure 2.8 shows realization diagram of a kind of frequently used convolution function

$$g(D) = \frac{a(D)}{b(D)} = \frac{a_0 + a_1D + a_2D^2 + \dots + a_mD^m}{1 + b_1D + b_2D^2 + \dots + b_mD^m},$$

where  $\mathbf{a} = (a_0, a_1, \dots, a_m) \in \{0, 1\}^{m+1}$ ,  $\mathbf{b} = (b_1, b_2, \dots, b_m) \in \{0, 1\}^m$ , and integer  $m$  is memory order. The squares with  $D$  are  $m$ -bit registers. For simplicity,  $a(D)$  and  $b(D)$  usually are

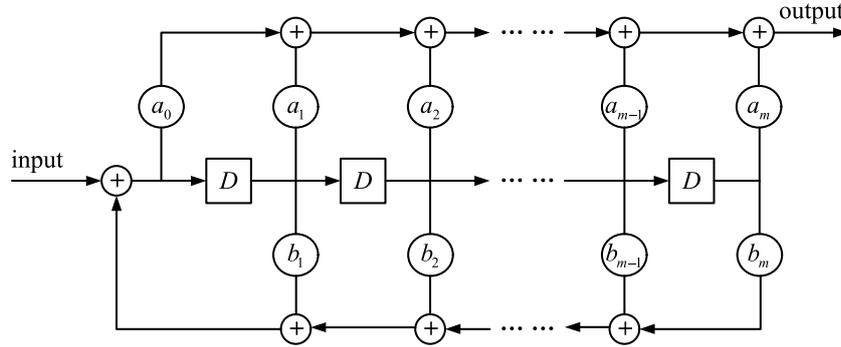


Figure 2.8: The realization of convolution function  $g(D) = a(D)/b(D)$ .

described by octal form. For example,  $g(D) = \frac{D+1}{D^3+D^2+1}$  is also denoted by  $(3/15)_{\text{octal}}$  or more simple form  $(3/15)$  [2].

**Example 2.5** Consider transfer matrix  $G(D) = [1, \frac{1+D^2}{1+D+D^2}]$  that leads to a systematic convolutional code with rate  $1/2$ . Its encoder is shown in Figure 2.9(a).

A convolutional code also can be represented by a trellis chart [22]. The trellis chart representation is shown in Figure 2.9(b). In trellis chart, filled squares are states of registers. Let  $s_0, s_1, s_2,$  and  $s_3$  are states '00', '01', '10', and '11', respectively. The edges represent state transfers (see Figure 2.9(b)). The numbers on edges denote input/output bits. For example, when the state of encoder is  $s_2 = '10'$ , an input information bit '1' transfer the state to  $s_1 = '01'$  and leads an output parity bit '0'.

Figure 2.9(c) shows coding realization graph [23] when input information sequence  $\mathbf{u} = \{101\}$ . In Figure 2.9(c), the squares with  $T$  are trellis nodes, the filled circles are variable nodes. We see that the output sequence  $\mathbf{v} = \{101\ 110\}$ . If we terminate the state



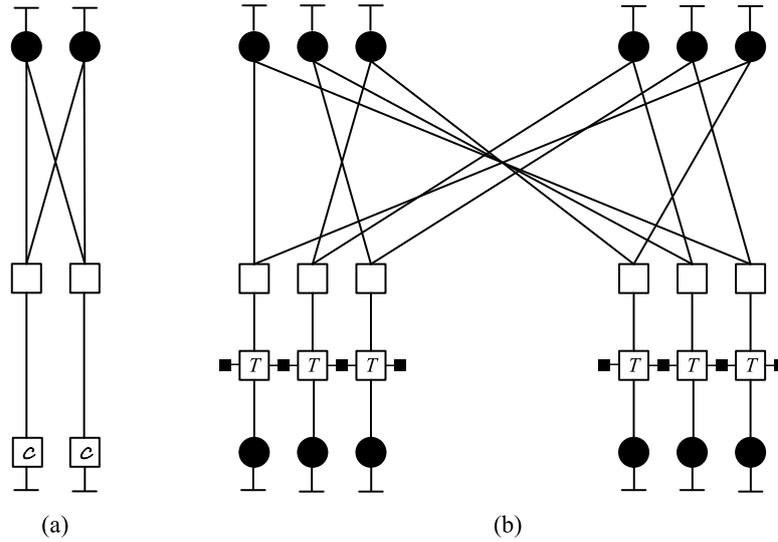


Figure 2.10: (a) An example of generalized protograph with rate-1 convolutional code nodes and (b) its realization graph with  $M = 3$ .

### 2.3.2 Generalized Graph-Based Code Ensembles: An Example

Based on above introduction of convolutional codes, we add a new type of nodes, convolutional code nodes, to generalize protograph. By this generalized protograph, we can obtain more degree of freedom to structure a code ensemble.

Figure 2.10 shows an example of generalized protograph and its coding realization graph with rate-1 convolutional code constraint nodes. The generalized protograph is usually called factor graph, too. In the generalized protograph, the squares with  $\mathcal{C}$  are convolutional code nodes. One can analyze the BP threshold on Figure 2.10(a), and makes an encoding and decoding of a code realization on Figure 2.10(b). Note that, when we determine BP threshold, the convolutional code constraint nodes should be considered as a whole to calculate extrinsic

information by BCJR decoding.

In the previous sections, we introduced the closed equations of variable and check nodes' calculation during DE or EXIT analysis. Unfortunately, the update equation of analytic expression is difficult to derive. For DE or EXIT analysis to convolutional code node, the Monte Carlo method is usually employed [25]. The numerical computation method is based on BCJR decoder. By an *a priori* information  $I_A$ , one can build an enough long *a priori* belief message sequence with Monte Carlo method. Run BCJR decoder with the built sequence, an output extrinsic belief message sequence can be obtained. By the output sequence, one can build extrinsic information  $I_E$ .

## 2.4 Summary

In this chapter, we introduced elements of graph-based codes. As main example, bipartite graph codes (LDPC codes) and their iterative decoding and ensemble decoding analysis are described. We also introduced a structure method of graph-based codes, spatial coupling. The chaining decoding process of spatially coupled codes provides more powerful decoding capability than conventional codes. Finally, we given a concept of generalized graph-based codes by introducing convolutional codes and their coding realization graph.



# Chapter 3

## Spatially Coupled

## Repeater-Combiner-Convolutional

## Codes

In this chapter, we propose a kind of spatially coupled serially-concatenated codes: spatially coupled repeater-combiner-convolutional (SC-RCC) codes. For the SC-RCC codes, we also analyze their theoretical iterative decoding performance and simulate their code realizations in point-to-point communication systems.

## 3.1 Introduction

Spatially coupled codes are obtained by connecting multiple identical base codes. For example, block low-density parity-check (LDPC) codes [26]. It has been demonstrated that the belief propagation (BP) threshold of a spatially coupled code ensemble asymptotically achieves the maximum *a posteriori* probability (MAP) threshold of the corresponding base code [13] [14]. The threshold saturation of spatially coupled codes has attracted a great deal of attention over the past few years.

While researchers have focused intensely on spatially coupled LDPC (SC-LDPC) codes, relatively little effort has been focused on how to couple serial concatenated codes. A serial concatenated code consists of an outer component code and an inner code [27]. Each of the two components can be block or convolutional codes, and proper component codes may improve the overall decoding performance. In particular, a convolutional inner code with infinite impulse response (IIR) is desirable since it ensures an interleaving gain [25].

The direct extension of SC-LDPC codes to spatially coupled concatenated codes is straightforward. The spatial coupling operation may involve one or both of the outer and inner component codes. Coupling convolutional-convolutional concatenated codes was attempted in [28]. By coupling both component codes of multiple base codes, the spatially coupled convolutional-convolutional concatenated codes asymptotically achieve MAP thresholds of their base code.

To structure spatially coupled serial concatenated codes, two efforts were made to

couple block-convolutional concatenated codes in which only the block outer codes are coupled. In [29], the base code is repeat-accumulate (RA) code, i.e., a block code (specifically, a repetition code followed by a combiner) concatenated with an accumulator. The coupled versions are called spatially coupled RA (SC-RA) codes. In [30], the base code is obtained by concatenating a block LDPC code with an RA code. Although both coupled codes [29] [30] provide BP thresholds near to the Shannon limits, they may be optimized by extending their accumulator to IIR convolutional codes.

In this chapter, we further investigate spatially coupled block-convolutional concatenated codes by proposing a form of spatially coupled repeater-combiner-convolutional (SC-RCC) codes. We begin with a base code that consists of the outer block LDPC code, repetition code (repeater) followed by a single-parity code (combiner), and inner code of a convolutional code. After copying the base code multiple times, we simply connect the outputs of the repetition codes to the inputs of several adjacent single-parity codes, while preserving unchanged the inputs and outputs of the convolutional inner code. Compared with conventional SC-LDPC and SC-LDPC-RA codes, in our coding scheme, the simple outer component code confirms simple encoding implementation. Compared with conventional SC-RA codes, the diverse construction of the convolutional component code provides more designation space to improve decoding performance. Notably, when the convolutional component code is chosen as a two-state IIR convolutional code accumulator, our code is reduced to a conventional SC-RA code.

We employed extrinsic information transfer (EXIT) functions to analyze the BP

thresholds of our proposed codes over additive white Gaussian noise channel (AWGNC). The numerical results show that the proposed codes provide BP thresholds and bit error rate (BER) performance closer to the corresponding Shannon limits than conventional SC-RA and SC-LDPC codes. In particular, when the rate is about 0.5, the proposed codes with a code length of 15628 have BER of  $10^{-5}$  at  $E_b/N_0$  of 1.13 dB, which is less than 1 dB away from the Shannon limit. This code length may therefore satisfy the practical requirements of communication systems.

## 3.2 Construction of SC-RCC Codes

We propose a kind of spatially coupled repeater-combiner-convolutional (SC-RCC) codes with a systematic block-convolutional concatenated base code.

Before proceeding further, let us explain the base codes, denoted by  $\mathcal{B}(q, \mathcal{C})$ . As shown in Figure 3.1(a),  $\boldsymbol{\mu} \in \{0, 1\}^M$  is an information bit vector. Each input bit of a  $q$ -repeater is repeatedly output  $q$  times:  $q \geq 3$ . Through interleaver  $\Pi_1$ , we get  $\boldsymbol{\alpha} \in \{0, 1\}^{qM}$ . For a  $q$ -combiner, every  $q$  input bit is mapped into one output bit that is the modulo-2 sum of those  $q$  input bits. Through interleaver  $\Pi_2$ ,  $\boldsymbol{\beta} \in \{0, 1\}^M$  is input into an encoder of convolutional code  $\mathcal{C}$  with rate  $R_{\mathcal{C}} = M/N$ . The output parity bit vector is  $\boldsymbol{\nu} \in \{0, 1\}^N$ . Vectors  $\boldsymbol{\mu}$  and  $\boldsymbol{\nu}$  are transmitted to the channel. Note that, as investigated previously [25], the base codes are actually enhanced RA codes by replacing the accumulator of the RA code with a stronger convolutional code.

Base codes  $\mathcal{B}(q, \mathcal{C})$  are also described by a factor graph in Figure 3.1(b). The filled circle, square, and square with a sign  $\mathcal{C}$  are the equation constraint (EC), single-parity constraint (SPC), and convolutional code constraint (CCC), which correspond to the  $q$ -repeater,  $q$ -combiner, and convolutional code in Figure 3.1(a), respectively. An edge between two constraints is an intermediate variable. In fact, each intermediate variable follows an interleave operation, which is not illustrated in the factor graph for brevity's sake [13] [29]. An edge connected to only one constraint is a channel variable. The left and right channel variables correspond to the information and parity parts of a transmitted word, respectively.

We can now describe the proposed SC-RCC codes denoted by  $\mathcal{S}(q, L, \mathcal{C})$ . To structure an SC-RCC code, we copy the factor graph of  $\mathcal{B}(q, \mathcal{C})$  by  $L$  times. Then, we reconnect only those edges between the ECs and SPCs, while preserving unchanged the input and output of the CCCs. Specifically, for an EC at the  $i$ -th position,  $0 \leq i \leq L - 1$ , we connect the  $\tau$ -th edge,  $0 \leq \tau \leq q - 1$ , to the SPC at the  $(i + \tau)$ -th position and thus obtain the factor graph of  $\mathcal{S}(q, L, \mathcal{C})$  in Figure 3.1(c). For the purpose of spatial coupling, it should be noted that  $q \geq 3$ . Here, the larger the value of  $q$ , the larger the number of edges and thus the higher the decoding complexity. Furthermore, to maintain compact representation, the interleavers are not illustrated in Figure 3.1(c), too.

In Figure 3.1(c), there are  $q - 1$  additional SPCs and CCCs at positions  $L$  to  $L + q - 2$ . The factor graph has  $L$  and  $L + q - 1$  channel variables of information and parity, respectively. Let  $M$  be the size of the information bit vector corresponding to each channel variable. The

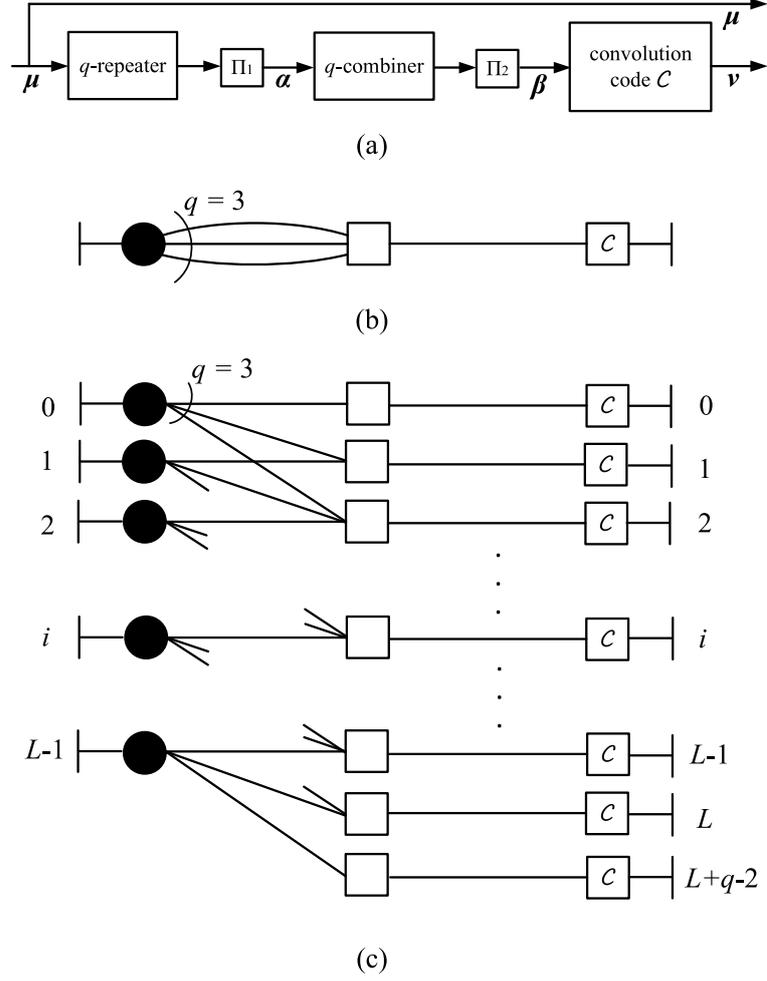


Figure 3.1: (a) Encoder and (b) factor graph of concatenated code  $\mathcal{B}(q, \mathcal{C})$ , and (c) factor graph of SC-RCC code  $\mathcal{S}(q, L, \mathcal{C})$  with  $q = 3$ .

rate of  $\mathcal{S}(q, L, \mathcal{C})$  is given by

$$R = \frac{LM}{LM + (L + q - 1)N} \approx \frac{R_c}{R_c + 1} \quad (3.1)$$

when  $L \gg q$ .

To decode the proposed code, we apply a BP algorithm to the corresponding factor graph. During the decoding process, the ECs and SPCs are decoded by a sum-product

algorithm and the CCCs are decoded by a BCJR algorithm.

Note that the proposed SC-RCC codes are an extension of the SC-RA codes [29]. By extending the accumulator into the convolutional code, we achieve much more design freedom for improving decoding performance. On the other hand, the proposed codes follow the coupling construction of SC-RA codes. Thus, our codes have simpler encoding implementation than SC-LDPC [13] and SC-LDPC-RA codes [30].

### 3.3 Iterative Decoding Analysis

In this section, we use EXIT function analysis [17] to determine the BP thresholds of SC-RCC code ensembles over AWGNC.

In order to simply analysis, similarly to [13], we modify the factor graph of Figure 3.1(c) by randomizing the edge connections: each of  $q$  state variable edges from the EC at position  $i$  is uniformly and independently connected to one of the SPCs from position  $i$  to  $i + w - 1$ , where  $w \leq L$ . We denote by  $\mathcal{S}(q, L, \mathcal{C}, w)$  the randomized version of our proposed codes. With this randomization, we can calculate the extrinsic information of each EC and each SPC by averaging *a priori* information. This kind of randomization is usually employed to assess the iterative decoding performance of the non-randomized ensemble [13] [29]. We also use the BP threshold worked out from  $\mathcal{S}(q, L, \mathcal{C}, w)$  to approximate the BP threshold of

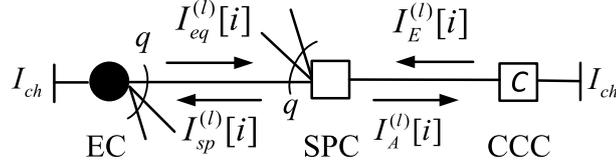


Figure 3.2: The message passing at the  $i$ -th coupled position of  $\mathcal{S}(q, L, \mathcal{C}, w)$ .

$\mathcal{S}(q, L, \mathcal{C})$ . Due to the randomness of the connection, the rate of  $\mathcal{S}(q, L, \mathcal{C}, w)$  becomes [13]

$$R = \frac{LR_{\mathcal{C}}}{LR_{\mathcal{C}} + \left(L + w + 1 - 2 \sum_{i=0}^w \binom{i}{w} q\right)}. \quad (3.2)$$

Figure 3.2 shows the  $l$ -th round message update at position  $i$  of the factor graph of  $\mathcal{S}(q, L, \mathcal{C}, w)$ . Let  $I_{ch}$  be channel information. Let  $I_{eq}^{(l)}[i]$ ,  $I_{sp}^{(l)}[i]$ ,  $I_A^{(l)}[i]$ , and  $I_E^{(l)}[i]$  be extrinsic information transferred from EC to SPC, SPC to EC, SPC to CCC, and CCC to SPC, respectively.

The EXIT functions of EC, SPC, and CCC are denoted by  $f$ ,  $g$ , and  $\lambda$ , respectively.

By averaging *a priori* information, the extrinsic information in  $\mathcal{S}(q, L, \mathcal{C}, w)$  are given by

$$I_{eq}^{(l)}[i] = f\left(I_{ch}, \frac{1}{w} \sum_{j=0}^{w-1} I_{sp}^{(l)}[i+j], q-1\right) \quad (3.3)$$

$$I_{sp}^{(l)}[i] = g\left(I_E^{(l-1)}[i], \frac{1}{w} \sum_{k=0}^{w-1} I_{eq}^{(l-1)}[i-k], q-1\right) \quad (3.4)$$

$$I_A^{(l)}[i] = g\left(1, \frac{1}{w} \sum_{k=0}^{w-1} I_{eq}^{(l-1)}[i-k], q\right) \quad (3.5)$$

$$I_E^{(l)}[i] = \lambda\left(I_{ch}, I_A^{(l)}[i]\right). \quad (3.6)$$

EXIT function  $f(I_1, I_2, d)$  (or  $g(I_1, I_2, d)$ ) above is given by a local decoding of variable (or check) node, where there are a single input of *a priori* information  $I_1$  and  $d$  identical

inputs of *a priori* information  $I_2$ . From the computation of EXIT functions and the  $J$ -function [25], we have

$$\begin{aligned} f(I_1, I_2, d) &= J\left(\sqrt{[(J^{-1}(I_1))^2 + d \cdot (J^{-1}(I_2))^2]}\right) \\ g(I_1, I_2, d) &= 1 - J\left(\sqrt{[J^{-1}(1 - I_1)]^2 + d \cdot [J^{-1}(1 - I_2)]^2}\right). \end{aligned}$$

Moreover, since CCC is decoded by the BCJR algorithm, the EXIT function  $\lambda$  can be determined by the Monte Carlo method [25].

By eliminating variables  $I_{sp}^{(l)}$  and  $I_A^{(l)}$  in (3.3) to (3.6), we obtain more concise update equations:

$$\begin{aligned} I_{eq}^{(l)}[i] &= f\left(I_{ch}, \frac{1}{w} \sum_{j=0}^{w-1} g\left(I_E^{(l-1)}[i+j], \right. \right. \\ &\quad \left. \left. \frac{1}{w} \sum_{k=0}^{w-1} I_{eq}^{(l-1)}[i+j-k], q-1\right), q-1\right) \end{aligned} \quad (3.7)$$

$$I_E^{(l)}[i+j] = \lambda\left(I_{ch}, g\left(1, \frac{1}{w} \sum_{k=0}^{w-1} I_{eq}^{(l-1)}[i+j-k], q\right)\right). \quad (3.8)$$

By updating (3.7) and (3.8) with initial values  $I_{eq}^{(0)}[i] = I_{ch}$ ,  $0 \leq i \leq L-1$ ,  $i \in Z$ , we determine the BP threshold of the mutual information of SC-RCC code ensemble  $\mathcal{S}(q, L, \mathcal{C}, w)$ . The BP threshold is defined by

$$I^{\text{BP}}(q, L, \mathcal{C}, w) \triangleq \inf\{I_{ch} \in [0, 1] : \mathbf{I}_{eq}^{(l)} \xrightarrow{l \rightarrow \infty} \mathbf{1}\} \quad (3.9)$$

where  $\mathbf{I}_{eq}^{(l)} = (I_{eq}^{(l)}[0], \dots, I_{eq}^{(l)}[L-1])$ . From  $I^{\text{BP}}$ , we can obtain the corresponding BP threshold of channel noise standard deviation by

$$\sigma^{\text{BP}} = \frac{2}{J^{-1}(I^{\text{BP}})}. \quad (3.10)$$

We can obtain the corresponding BP threshold of  $E_b/N_0$  by

$$(E_b/N_0)^{\text{BP}} = 10 \log_{10} \left( \frac{1}{2R(\sigma^{\text{BP}})^2} \right). \quad (3.11)$$

Note that equations (3.7) and (3.8) also can be employed to calculate the BP thresholds of the proposed codes over a binary erasure channel (BEC), where EXIT functions  $f$ ,  $g$ , and  $\lambda$  should be given over a BEC.

## 3.4 Numerical Results

### 3.4.1 BP thresholds

In this subsection, we update (3.7) and (3.8) to approximately calculate the BP thresholds of  $E_b/N_0$  of the proposed SC-RCC codes  $\mathcal{S}(q, L, \mathcal{C})$ . We set the parameter at  $w = q$ . The maximum iteration number of EXIT function evolutions is set at 50000 and the breakout condition of each  $1 - I_{eq}^{(l)}[i]$  is set at  $10^{-4}$ . Rate-1 recursive convolutional component codes, such as  $\mathcal{C}(5/7)$ , are employed, where (5/7) is generator polynomial  $\frac{1+D^2}{1+D+D^2}$  in octal form. Each rate versus the BP threshold performance is obtained by connecting the threshold-rate points of coupled lengths  $L = 5, 10, 15, 20, \dots, 150$ .

Focusing on the effect of the various 4-state convolutional component codes  $\mathcal{C}(1/7)$ ,  $\mathcal{C}(3/7)$ , and  $\mathcal{C}(5/7)$  in Figure 3.3, we see that, at the same rate, the proposed codes with  $\mathcal{C}(5/7)$  have BP thresholds nearest to the Shannon limits, with  $q = 4$  being slightly better than  $q = 3$ . As stated in Sec. II, for a given  $L$ , the smallest  $q$  ( $= 3$ ) gives the lowest decoding

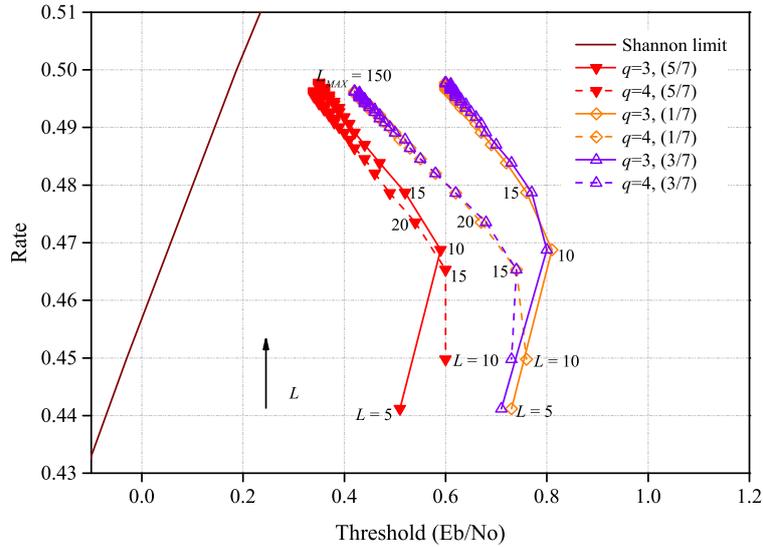


Figure 3.3: Rate versus threshold curves of SC-RCC codes with 4-state convolutional component codes  $\mathcal{C}(1/7)$ ,  $\mathcal{C}(3/7)$ , and  $\mathcal{C}(5/7)$ .

complexity.

Let us consider the effect of  $L$  for a given  $q$ , e.g., from the curve with  $q = 3$  and  $\mathcal{C}(5/7)$  in Figure 3.3. In the large  $L$  region ( $L > 10$ ), with an increase of  $L$ , rates tend to be 0.5 (base code's rate), while BP thresholds are near the corresponding Shannon limits. This is so-called threshold saturation phenomenon of spatial coupling [2]. In the small  $L$  region ( $L \leq 10$ ), unfortunately we do not observe this phenomenon. In fact, BP thresholds decrease as  $L$  (and thus rates) decrease. This is consistent with conventional (uncoupled) coding theory.

Then, looking at the proposed codes with the 8-state convolutional component codes  $\mathcal{C}(13/15)$ ,  $\mathcal{C}(15/17)$ , and  $\mathcal{C}(11/17)$  in Figure 3.4, we see that there is little difference in

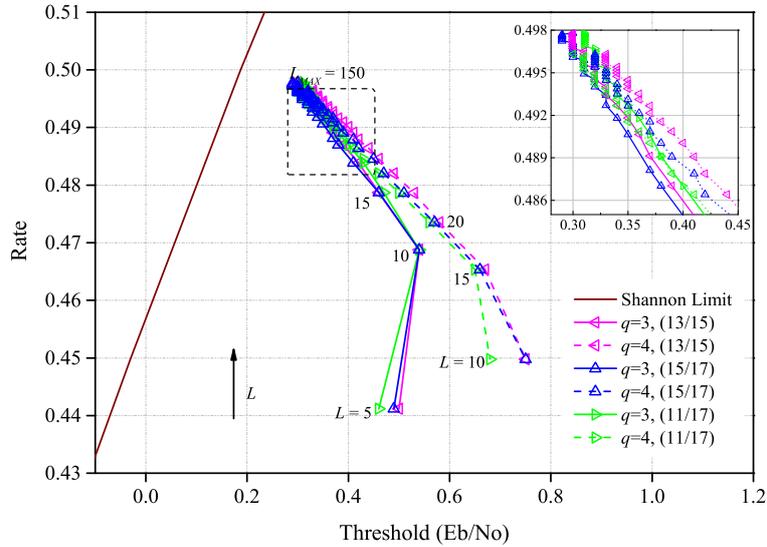


Figure 3.4: Rate versus threshold curves of SC-RCC codes with 8-state convolutional component codes  $\mathcal{C}(13/15)$ ,  $\mathcal{C}(15/17)$ , and  $\mathcal{C}(11/17)$ .

performance among the codes in the high-rate region.

Furthermore, we compare the performance among the proposed codes with 2-, 4-, and 8-state convolutional component codes in Figure 3.5. It is apparent that 4-state  $\mathcal{C}(5/7)$  provides obvious improvement over the 2-state  $\mathcal{C}(1/3)$  with  $q = 3, 4$ , and is slightly worse than that with  $q = 6$  at the same rate. For a given  $L$ ,  $\mathcal{C}(5/7)$  provides higher rates and better BP thresholds than  $\mathcal{C}(1/3)$  with  $q = 6$ . It should be pointed out that the codes with  $\mathcal{C}(1/3)$  are actually the conventional SC-RA codes of [29]. Moreover, although 4-state  $\mathcal{C}(5/7)$  results in slightly worse performance than 8-state  $\mathcal{C}(15/17)$ , the former has a smaller state number and thus less decoding complexity.

In Figure 3.5, we also compare the proposed codes with conventional SC-LDPC codes

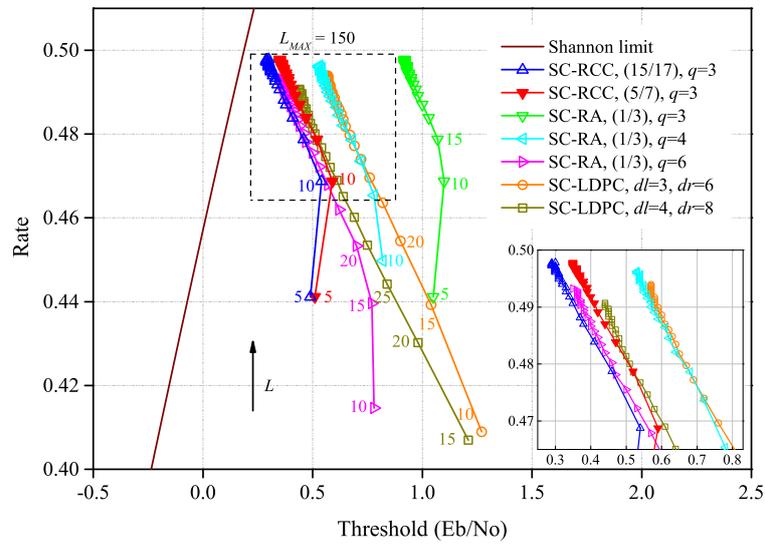


Figure 3.5: Rate versus threshold curves of the SC-RCC codes  $\mathcal{S}(3, L, \mathcal{C}, 3)$  with  $\mathcal{C}(5/7)$  and  $\mathcal{C}(15/17)$ , SC-RA codes  $\mathcal{S}(q, L, \mathcal{C}(1/3), w = q)$  with  $q = 3, 4, 6$ , and SC-LDPC codes  $\mathcal{L}(dl, dr, L, w = dl)$  with  $(dl, dr) = (3, 6)$  and  $(4, 8)$ .

$\mathcal{L}(dl, dr, L, w)$  [13], where  $dl$  and  $dr$  are variable and check node degrees. At the same rate, the proposed codes perform obviously better than  $\mathcal{L}(3, 6, L, 3)$  and slightly better than  $\mathcal{L}(4, 8, L, 4)$ . For a given  $L$ , the proposed codes have an obviously higher rate than the SC-LDPC codes.

Interestingly, it is known that, over a BEC, the BP thresholds of spatially coupled parallel and serial turbo codes with convolutional component codes  $\mathcal{C}(1, 5/7)$  are 0.4689 and 0.4969, respectively [28]. At the same rate, our code  $\mathcal{S}(3, L, \mathcal{C}(5/7), 3)$  with a large  $L$  (e.g., 1000) has an erasure probability BP threshold of 0.4972, which is better than the parallel one.

### 3.4.2 Simulation of Finite Length Codes

In this section, we simulate the error correction performance of finite length codes. The simulated codes are obtained by random interleavers. We set the maximum iteration number of decoders at 150 for the SC-RCC and SC-RA codes, and 1000 for the SC-LDPC codes. The coupled length  $L$  is fixed at 30.

Figure 3.6 shows the BER versus  $E_b/N_0$  curves of SC-RCC codes  $\mathcal{S}(3, 30, \mathcal{C}(5/7))$  with rate near 0.5 and code length  $n = 15628, 21828$ , and  $28028$ . We see that the code with  $n = 15628$  has a BER of  $10^{-5}$  at  $E_b/N_0$  of 1.13 dB.

In Figure 3.6, we also compare the proposed codes and conventional codes at almost the same code length (about 28000) and rate (near 0.5). The approximate BP thresholds of

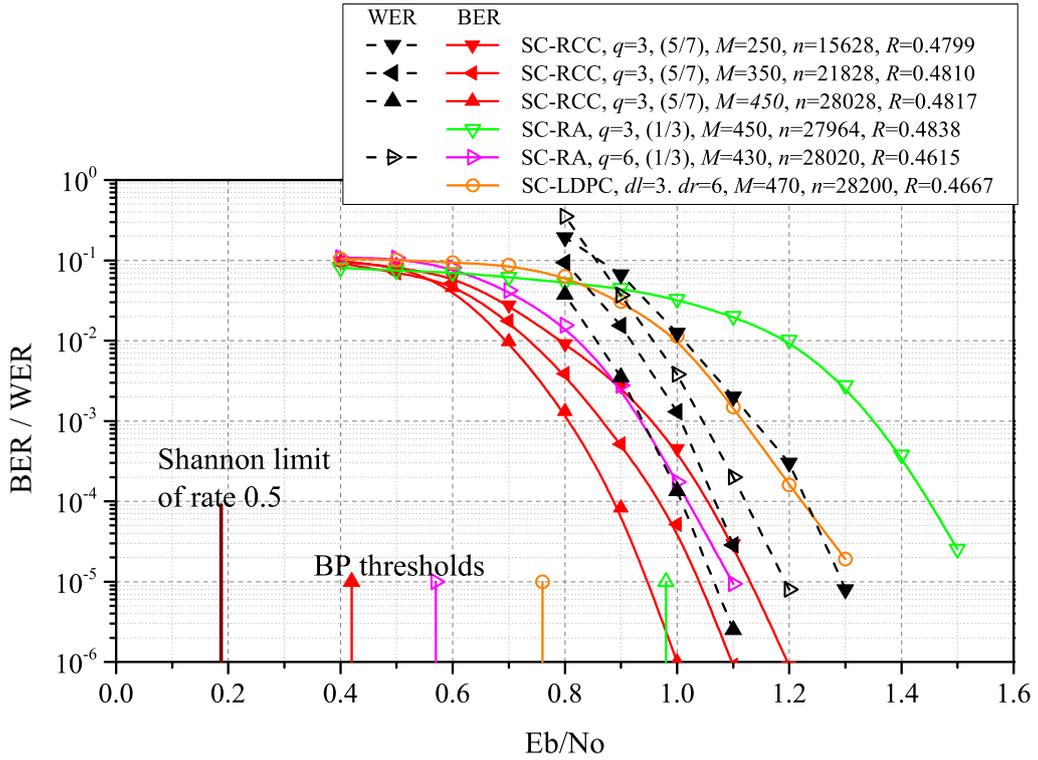


Figure 3.6: Error correction performance of SC-RCC codes  $\mathcal{S}(3, 30, \mathcal{C}(5/7))$  with  $n = 15628, 21828,$  and  $28028$ , SC-RA codes  $\mathcal{S}(3, 30, \mathcal{C}(1/3))$  with  $n = 27964$  and  $\mathcal{S}(6, 30, \mathcal{C}(1/3))$  with  $n = 28020$ , and SC-LDPC code  $\mathcal{L}(dl = 3, dr = 6, L = 30)$  with  $n = 28200$ .

the related codes provided in Sec. IV are also shown in Figure 3.6. We see that proposed code  $\mathcal{S}(3, 30, \mathcal{C}(5/7))$  obviously performs better BER than the SC-RA code with  $q = 3, 6$  and SC-LDPC code  $\mathcal{L}(3, 6, 30)$ . In addition, the consistent results are reflected in the word error ratio (WER) comparison. We do not illustrate the WER curves of the SC-RA code with  $q = 3$  and SC-LDPC code in Fig. 3.6 due to their obviously inferior performance.

### 3.5 Summary

We proposed SC-RCC codes that are obtained by coupling the outer codes of multiple block-convolutional concatenated codes. The proposed codes have flexible designation space and achieve better BP threshold and error correction performance than several related conventional codes. In particular, the proposed codes with a rate of about 0.5 and length of 15628 have a BER of  $10^{-5}$  at  $E_b/N_0$  of 1.13 dB that is less than 1dB away from the Shannon limit on an AWGN channel. Future work should investigate the optimization of interleavers for further improvement of the decoding performance.

# Chapter 4

## SC-RCC Codes on Multiple Access Channels

In this chapter, we use SC-RCC codes on Gaussian multiple access channels (MAC), and investigate their error correction performance in theory and code realizations.

### 4.1 Introduction to MAC Coding

Before proceeding, let us review the model of Gaussian MAC with 2 users. As shown in Figure 4.1(a), the MAC is defined by

$$Y = h_1X_1 + h_2X_2 + Z \tag{4.1}$$

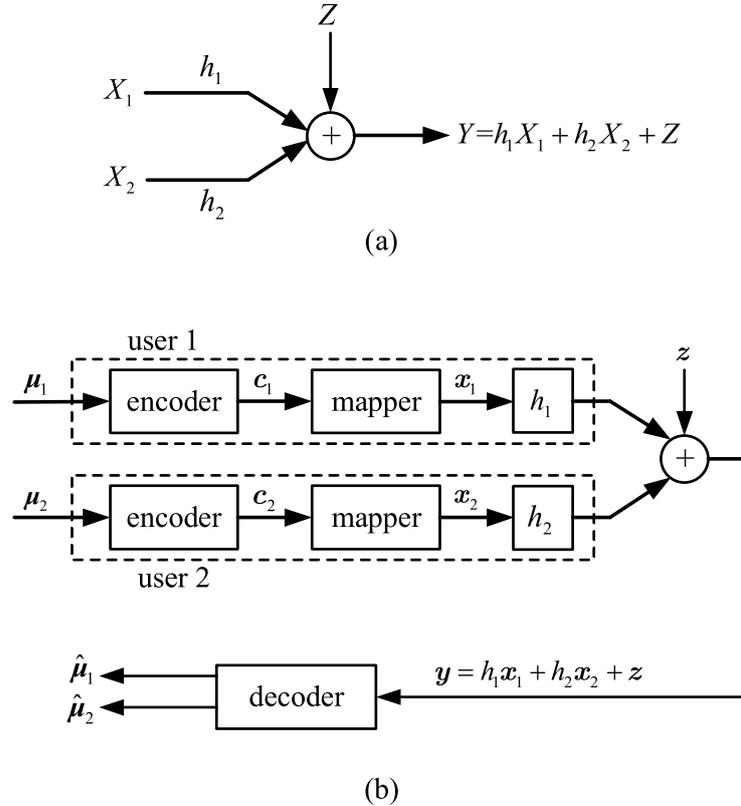


Figure 4.1: (a) 2-user Gaussian MAC model, and (b) system diagram of coding on Gaussian MAC.

The channel inputs  $X_1, X_2 \in \{\pm 1\}$  are uniformly distributed random variables. The variation in channel gains  $h_1, h_2 \in [0, \infty)$  can be explained by power constraint parameters. (Strictly speaking,  $h_1 = \sqrt{P_1}$  and  $h_2 = \sqrt{P_2}$ , where  $P_1$  and  $P_2$  are actual power gains of different users. For brevity's sake, we adopt parameters  $h_1$  and  $h_2$  to depict MAC model [31].) The noise  $Z$  is a random variable that follows Gaussian distribution  $\mathcal{N}(0, 1)$ .

Assume that the messages of two users are encoded by the same error correction codes with rate  $R$ . The system diagram of coding on 2-user Gaussian MAC is shown in Figure

4.1(b). Let vectors  $\boldsymbol{\mu}_1$  and  $\boldsymbol{\mu}_2$  be messages of users 1 and 2, respectively. Encoding messages of users 1 and 2, we have codewords  $\mathbf{c}_1$  and  $\mathbf{c}_2$ . Through mapping of  $\{0, 1\} \rightarrow \{+1, -1\}$ , we get symbol vectors  $\mathbf{x}_1$  and  $\mathbf{x}_2$ . Transmitting  $\mathbf{x}_1$  and  $\mathbf{x}_2$  on Gaussian MAC with power constraints  $h_1$  and  $h_2$ , the received signal vector is  $\mathbf{y} = h_1\mathbf{x}_1 + h_2\mathbf{x}_2 + \mathbf{z}$ , where each component of noise vector  $\mathbf{z}$  is a realization of  $Z$ . At receiver, a decoder is used to rebuild messages of users 1 and 2.

Given a power constraint tuples  $(h_1, h_2)$ , if the receiver can decode successfully, the power constraint tuples is achievable. With coding rate  $R$ , the MAC achievable power region (MAC-APR) is defined as the set of all achievable power tuples  $(h_1, h_2)$ , given by the equations

$$R \leq c(h_1^2) \quad (4.2)$$

$$R \leq c(h_2^2) \quad (4.3)$$

$$2R \leq c(h_1^2 + h_2^2) \quad (4.4)$$

where  $c(x) = 0.5 \log(1 + x)$  is the capacity function AWGNC [32]. A diagram of MAC-APR boundary is shown in Figure 4.2. The upper right area of the boundary is MAC-APR.

For specific code in MAC communication system, under BP decoding, there exist a BP achievable power region (BP-APR) [31]. The transmitted signal with arbitrary power tuples  $(h_1, h_2)$  in BP-APR can be error freely decoded. In the research of coding for MAC, the BP-APR is determined by calculating the BP thresholds for all possible power tuples.

Firstly, let us look at what a common code performs. For a common code, the diagram

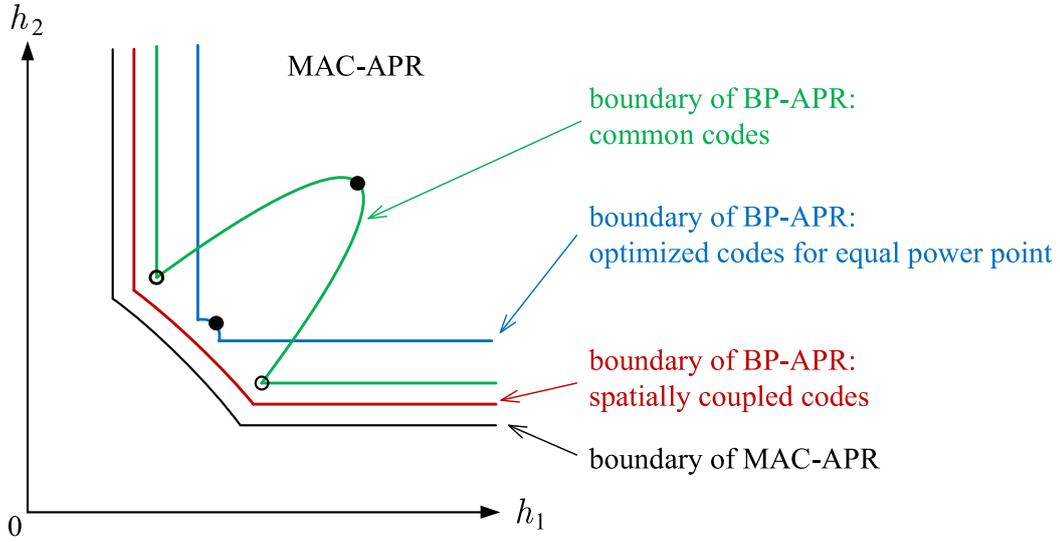


Figure 4.2: The boundary MAC-APR and boundaries of BP-APR for different codes.

of BP-APR boundary is illustrated in Figure 4.2. There are two types of important points on the BP-APR boundary. One type is the minimum sum power points that are labelled by blank small circles in Figure 4.2. Another type is the minimum equal power point that is labelled by filled small circle in Figure 4.2. Compared with the BP-APR and MAC-APR, there is a half ellipse unachievable region [33].

A typical MAC coding problem is that encode message of transmitters such that sum power of transmission is minimum. Obviously, the problem is equivalent to selecting coding scheme such that the minimum sum power points as near to the boundary of MAC-APR as possible. When transmitted signals with power allocation on the minimum sum power points, a successive cancellation decoding scheme can perfectly rebuild information of transmitters [34]. In fact, a capacity-approaching code in single user system can guarantees

the minimum sum power points near to the boundary of MAC-APR. However, a capacity-approaching single user code usually has a large half ellipse unachievable region.

Another typical MAC coding problem is based on an assumption that the transmission powers of all transmitters are equal. Thus, the problem of minimizing sum power is equivalent to selecting coding scheme such that the minimum equal power point as near to the boundary of MAC-APR as possible. Figure 4.2 also diagrams the BP-APR boundary of optimized codes for equal power point [35]. However, an optimized code with good minimum equal power point usually has BP-APR whose most part of boundary is away from the boundary of MAC-APR.

Whether one optimizes the minimum sum power points or equal power point, there is a considerable area of MAC-APR that cannot be achieved by MAC coding. In practical communication, the channel gains for different transmitters often randomly vary as time. This means that received power  $(h_1, h_2)$  could be located at an undecodable region with a large probability. Thus, an optimal solution to the MAC coding problem is to find codes such that their BP-APR is universally close to MAC-APR.

Spatially coupled codes are shown to universally approach MAC-APR on MAC [31] [36]. Figure 4.2 also diagrams the BP-APR boundary of spatially coupled codes. Although the BP-APR boundary is determined based on the assumption of infinite code length, spatially coupled codes may be the perfect solution to the MAC coding problem. All we need to do is find a good coding realization with as short a code length as possible for practical communication systems on MAC.

In this chapter, we focus on coding realization on Gaussian MAC. Due to excellent finite length performance, we employ our proposed SC-RCC codes, given in previous chapter, to encode two users' messages and investigate the iterative detective-decoding performance.

## 4.2 SC-RCC Codes on MAC

Consider SC-RCC code  $\mathcal{S}(q, L, \mathcal{C})$  for users 1 and 2. Figure 4.3(a) shows the factor graph of  $\mathcal{S}(3, 6, \mathcal{C})$  code at user  $t$ ,  $t = 1, 2$ . The filled circles, blank squares and squares with  $\mathcal{C}$  sign are equation constraint (EC), single-parity constraint (SPC), and convolutional code constraint (CCC), respectively. Review system diagram of coding on Gaussian MAC in Figure 4.1(b). For the  $t$ -th user, the message  $\boldsymbol{\mu}_t$ , codeword  $\mathbf{c}_t$ , and symbol vector  $\mathbf{x}_t$  can be obtained by encoding on factor graph in Figure 4.3(a).

Let  $\boldsymbol{\mu}_t = (\boldsymbol{\mu}_{t,0}, \boldsymbol{\mu}_{t,1}, \dots, \boldsymbol{\mu}_{t,L-1})$  be message bit vector of user  $t$ , where  $\boldsymbol{\mu}_{t,i} \in \{0, 1\}^M$  is message block at  $i$ -th coupling position,  $0 \leq i \leq L - 1$ . Each  $\boldsymbol{\mu}_{t,i}$  is input into an EC that is actually a repeater, and then is repeatedly output  $q$  times. Note that, for the purpose of spatial coupling, it should be that  $q \geq 3$ . Through interleavers, we get  $\boldsymbol{\alpha}_{t,i\tau} \in \{0, 1\}^M$ ,  $0 \leq \tau \leq q - 1$ . For an SPC that is actually a combiner, all of input bit vectors are mapped into one output bit vector that is bitwise modulo-2 sum of those input vectors. Through interleavers, we get  $\boldsymbol{\beta}_{t,i'} \in \{0, 1\}^M$ ,  $0 \leq i' \leq L + q - 2$ . Each  $\boldsymbol{\beta}_{t,i'}$  is input into a CCC that is a convolutional component code  $\mathcal{C}$  with rate  $R_{\mathcal{C}} = M/N$ . Then we obtain parity bit vector  $\boldsymbol{\nu}_t = (\boldsymbol{\nu}_{t,0}, \boldsymbol{\nu}_{t,1}, \dots, \boldsymbol{\nu}_{t,L+q-2})$ ,  $\boldsymbol{\nu}_{t,i'} \in \{0, 1\}^N$ . Message vector  $\boldsymbol{\mu}$  and parity vector  $\boldsymbol{\nu}$  will be

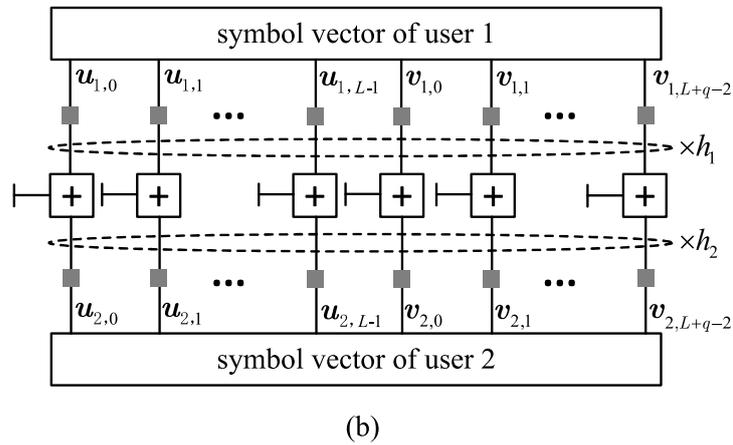
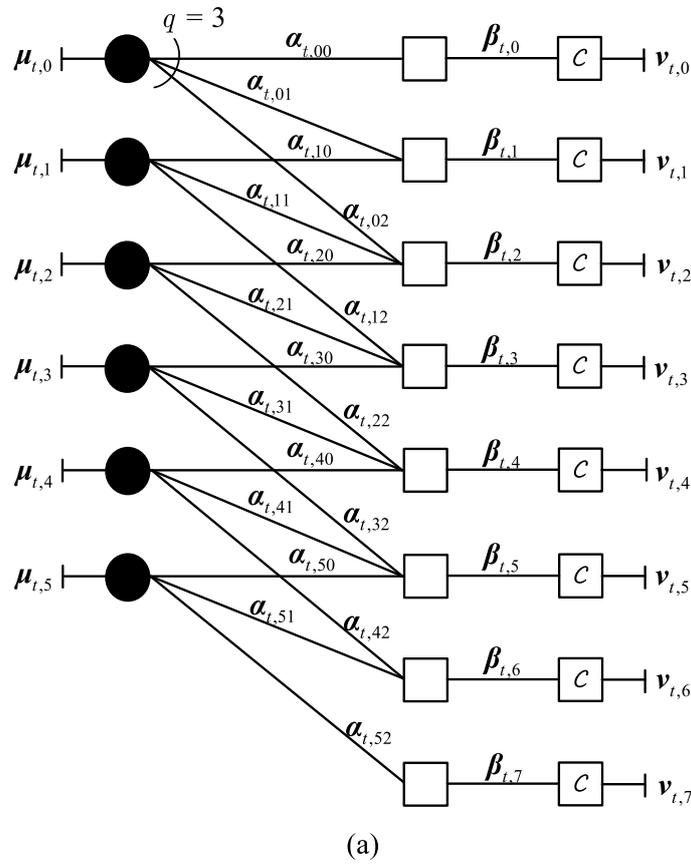


Figure 4.3: (a) Factor graph of SC-RCC code  $\mathcal{S}(q = 3, L = 6, \mathcal{C})$  at user  $t, t = 1, 2$  and (b) illustration of the permutation and transmission of two users' symbol vectors.

modulated and transmitted. We get the  $t$ -th user's codeword

$$\mathbf{c}_t = (\boldsymbol{\mu}_{t,0}, \boldsymbol{\mu}_{t,1}, \dots, \boldsymbol{\mu}_{t,L-1}, \boldsymbol{\nu}_{t,0}, \boldsymbol{\nu}_{t,1}, \dots, \boldsymbol{\nu}_{t,L+q-2}).$$

Through mapper of  $\{0, 1\} \rightarrow \{+1, -1\}$ , we get the  $t$ -th user's symbol vector

$$\mathbf{x}_t = \{\mathbf{u}_{t,0}, \mathbf{u}_{t,1}, \dots, \mathbf{u}_{t,L-1}, \mathbf{v}_{t,0}, \mathbf{v}_{t,1}, \dots, \mathbf{v}_{t,L+q-2}\}$$

where symbol blocks  $\mathbf{u}_{t,i} \in \{\pm 1\}^M$  and  $\mathbf{v}_{t,i'} \in \{\pm 1\}^N, t = 1, 2$ . Therefore, the receiver obtains a superimposed signal vector  $\mathbf{y} = h_1 \mathbf{x}_1 + h_2 \mathbf{x}_2 + \mathbf{z}, \mathbf{z} \in \mathcal{N}(0, 1)^{ML+N(L+q-1)}$ .

For guarantee excellent decoding performance, each user's symbol vector should be permuted before transmitting [37]. However, using an uniform permutation to each  $\mathbf{x}_t$  will lead to chaining decoding effect of spatial coupling invalidations. Thus, we have to respectively permute every message or parity blocks and transmit them in the same order. Figure 4.3(b) illustrates the permutation and transmission of two users' symbol vectors. In Figure 4.3(b), the small grey squares are interleavers, and the squares with plus sign are MAC constraint (MACC).

To rebuild the two users' messages at receiver, we using a joint iterative BP detection-decoding on the corresponding factor graph. The MACCs are processed by maximum posterior probability (MAP) detection, the ECs and SPCs are processed by MAP decoding similar to that of LDPC codes' variable and check nodes, and the CCCs are processed by a BCJR decoding.

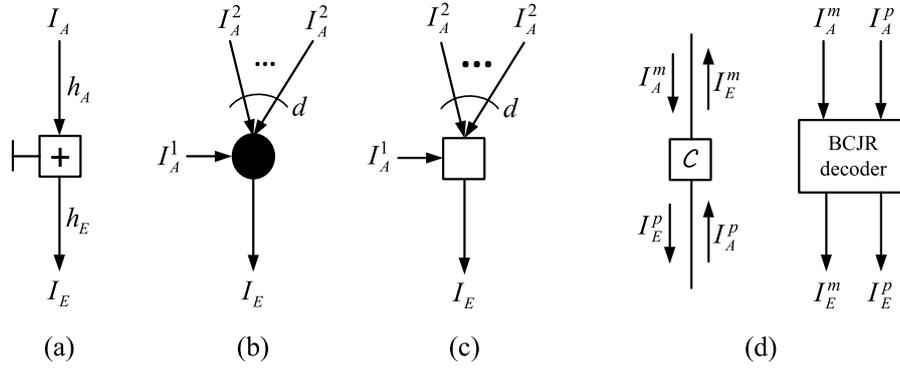


Figure 4.4: EXIT illustration of (a) MACC, (b) EC, (c) SPC, and (d) CCC.

### 4.3 Iterative Detection-Decoding Analysis

In this section, we employ EXIT function evolution to determine the BP threshold for SC-RCC code ensemble over Gaussian MAC.

#### 4.3.1 EXIT Function for Local Processing

Before analyzing proposed scheme, we first formulate the EXIT functions for the local detection at MACC and the local decoding at EC, SPC, and CCC in Figure 4.3.

##### 4.3.1.1 EXIT Function of MACC

Let  $\phi$  denote EXIT Function of MACC. For a general local MACC, as shown in Fig. 4.4(a), extrinsic information  $I_E$  depends on *a priori* information  $I_A$ , *a priori* message power

$h_A$ , and extrinsic message power  $h_2$ . The EXIT function of general MACC is defined by

$$I_E \triangleq \phi(I_A, h_A, h_E). \quad (4.5)$$

We employ a MAP detection at MACC. The  $\phi$ -function can be calculated by [38].

#### 4.3.1.2 EXIT Functions of EC and SPC

Let  $f$  and  $g$  denote EXIT Functions of EC and SPC, respectively. The decoding process of EC (or SPC) is similar to that of LDPC code's variable (or check) node. For a general local EC (or SPC) matching the proposed scheme, its extrinsic information  $I_E$  depends on a single *a priori* information  $I_A^1$  and  $d$  identical *a priori* information  $I_A^2$ , as shown in Fig. 4.4(b) (or Fig. 4.4(c)). From the computation of variable and check nodes' EXIT function, the EXIT function of the local EC and SPC are given by

$$\begin{aligned} I_E &\triangleq f(I_A^1, I_A^2, d) \\ &= J\left(\sqrt{[(J^{-1}(I_A^1))]^2 + d \cdot [(J^{-1}(I_A^2))]^2}\right) \end{aligned} \quad (4.6)$$

$$\begin{aligned} I_E &\triangleq g(I_A^1, I_A^2, d) \\ &= 1 - J\left(\sqrt{[(J^{-1}(1 - I_A^1))]^2 + d \cdot [(J^{-1}(1 - I_A^2))]^2}\right) \end{aligned} \quad (4.7)$$

where the  $J$ -function is given in Section 2.

### 4.3.1.3 EXIT Functions of CCC

A local CCC is actually a convolutional code. It is decoded by a BCJR decoder. As a component processor matching the proposed scheme, the BCJR decoder can rebuild both the message part (input of CCC) and parity part (output of CCC). Let *a priori* information of the two parts are  $I_A^m$  and  $I_A^p$ , corresponding extrinsic information are  $I_E^m$  and  $I_E^p$ . Their relation is illustrated in Fig. 4.4(d). The EXIT functions of the two parts are defined by

$$I_E^m \triangleq \lambda(I_A^m, I_A^p) \quad (4.8)$$

$$I_E^p \triangleq \rho(I_A^m, I_A^p) \quad (4.9)$$

where the function  $\lambda$  and  $\rho$  can be determined by the Monte Carlo method.

### 4.3.2 BP Thresholds

In order to simply analysis, we employ randomized ensemble  $\mathcal{S}(q, L, \mathcal{C}, w)$  to approximately determine the BP threshold of the iterative detection-decoding system of MAC.

Fig. 4.5 shows the factor graph representation of joint iterative detection-decoding analysis at the  $i$ -th coupling position. Note that, although the interleavers are not illustrated in Fig. 4.5 for brevity's sake, we analyze ensemble's BP threshold for all possible interleaver permutations illustrated in Fig. 4.3.

For ease of description, we label MACC sets connected to ECs and CCCs by  $V_1$  and

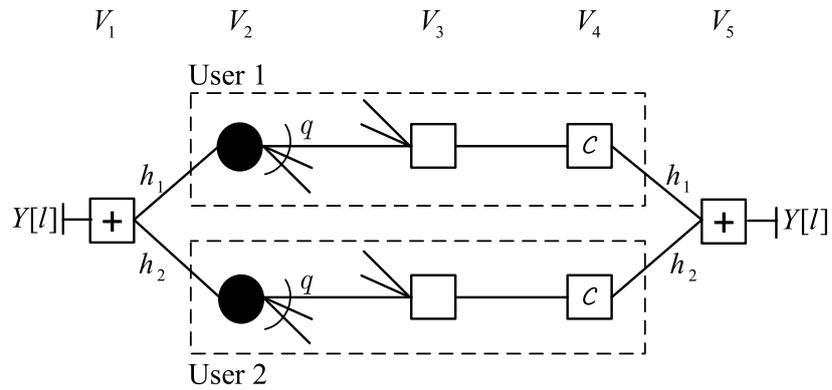


Figure 4.5: The Factor graph representation of joint iterative detection-decoding analysis at the  $l$ -th coupling position.

$V_5$ , respectively. Besides, we label EC, SPC, and CCC sets by  $V_2$ ,  $V_3$ , and  $V_4$ , respectively. Let  $I_{t,uv}[i]$  be the  $t$ -th user's extrinsic information passed from a constraint in set  $V_u$  to a constraint in set  $C_v$  at the  $i$ -th coupling position, where  $uv \in \{12, 23, 34, 45, 54, 43, 32, 21\}$ .

At  $l$ -th iteration, the  $t$ -th user's update equations of extrinsic information are given by

$$I_{t,12}^{(l+1)}[i] = \phi\left(I_{t',21}^{(l)}[i], h_{t'}, h_t\right) \quad (4.10)$$

$$I_{t,23}^{(l+1)}[i] = f\left(I_{t,12}^{(l)}[i], \frac{1}{w} \sum_{j=0}^{w-1} I_{t,32}^{(l)}[i+j], q-1\right) \quad (4.11)$$

$$I_{t,34}^{(l+1)}[i] = g\left(1, \frac{1}{w} \sum_{k=0}^{w-1} I_{t,23}^{(l)}[i-k], q\right) \quad (4.12)$$

$$I_{t,45}^{(l+1)}[i] = \rho\left(I_{t,34}^{(l)}[i], I_{t,54}^{(l)}[i]\right) \quad (4.13)$$

$$I_{t,54}^{(l+1)}[i] = \phi\left(I_{t',45}^{(l)}[i], h_{t'}, h_t\right) \quad (4.14)$$

$$I_{t,43}^{(l+1)}[i] = \lambda\left(I_{t,34}^{(l)}[i], I_{t,54}^{(l)}[i]\right) \quad (4.15)$$

$$I_{t,32}^{(l+1)}[i] = g\left(I_{t,43}^{(l)}[i], \frac{1}{w} \sum_{k=0}^{w-1} I_{t,23}^{(l)}[i-k], q-1\right) \quad (4.16)$$

$$I_{t,21}^{(l+1)}[i] = f\left(0, \frac{1}{w} \sum_{j=0}^{w-1} I_{t,32}^{(l)}[i+j], q\right) \quad (4.17)$$

where  $t = 1, 2$  and  $t' = 3 - t$ .

By updating (4.10) to (4.17) with given  $(h_1, h_2)$  and initial value  $I_{t,uv}^{(0)}[i] = 0$ , we can judge the pow constraint tuples  $(h_1, h_2)$  is achievable (all related  $I_{t,uv}^{(\infty)}[i] \rightarrow 1$ ) or unachievable (all related  $I_{t,uv}^{(\infty)}[i] \rightarrow 0$ ). Then, the BP-APR can be determined.

## 4.4 Numerical Results

Using analysis in Section 4.3, we can determine the BP-APR of our SC-RCC codes on Gaussian MAC. As an example, Figure 4.6 shows the BP-APR of  $\mathcal{S}(3, 30, \mathcal{C}(1/5))$ . We see that the BP-APR is universally close to MAC-APR with rate-0.5 coding.

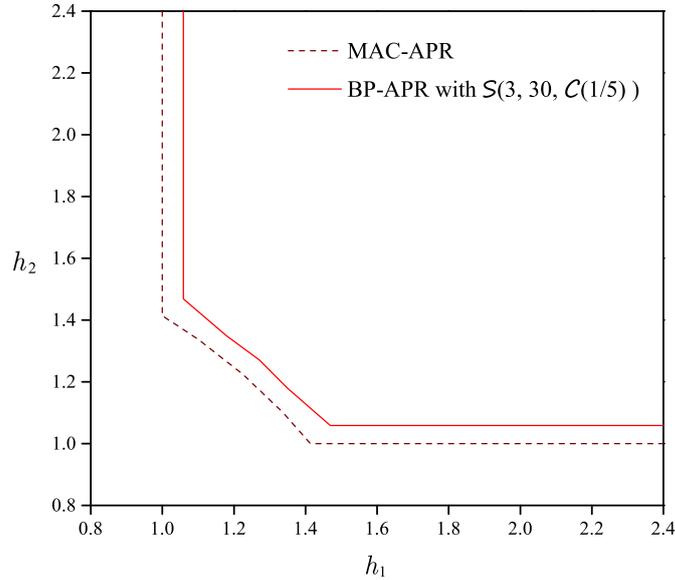


Figure 4.6: BP-APR of  $\mathcal{S}(3, 30, \mathcal{C}(1/5))$  on Gaussian MAC.

Observe the BP-APR of spatially coupled codes in Figure 4.2 again. In fact, the boundary lines paralleled with coordinate axes reflect the BP decoding performance of codes on single user AWGNC. Since we have shown the SC-RCC codes are capacity-approaching on AWGNC in previous chapter, we just need to investigate SC-RCC codes transmitted with equal power point. Let  $h = h_1 = h_2$ . The fact that required smallest power  $h$  with error-free decoding is close to theoretical limit of  $h^*$  can mean that the investigated code universally approach MAC-APR. With the definition of MAC-APR, we have  $h^* = \sqrt{[2^{4R} - 1]/2}$ .

Figures 4.7 and 4.8 show the BER versus  $h$  iterative detection-decoding performance of SC-RCC codes with  $M = 100, 200$  and code length  $n = 6200, 12400$ , respectively.

We see that convolutional component code  $\mathcal{C}(1/5)$  provides better decoding perfor-

mance than  $\mathcal{C}(1/3)$ ,  $\mathcal{C}(1/7)$  and  $\mathcal{C}(5/7)$  when BER is set as  $10^{-5}$ . Compared with conventional SC-LDPC code  $\mathcal{L}(dl = 3, dr = 6, L = 30)$ , the SC-RCC code  $\mathcal{S}(3, 30, \mathcal{C}(1/5))$  also performs better.

From Figures 4.7 and 4.8, we also see the  $\mathcal{S}(3, 30, \mathcal{C}(1/5))$  has decoding error floor when BER is about  $10^{-6}$ . The error floor performance better than that of  $\mathcal{S}(3, 30, \mathcal{C}(1/3))$  and worse than these of other considered codes. This is because the codeword weight distribution of  $\mathcal{S}(3, 30, \mathcal{C}(1/5))$  is better than  $\mathcal{S}(3, 30, \mathcal{C}(1/3))$  and worse than other codes [2].

## 4.5 Summary

In this chapter, we employed our proposed SC-RCC codes on Gaussian MAC. We analyzed the BP-APR and simulated several code realizations. Numerical results show that the BP-APR of SC-RCC codes universally approaches to MAC-APR and  $\mathcal{S}(3, 30, \mathcal{C}(1/5))$  provides better BER performance than conventional SC-LDPC codes at equal power case.

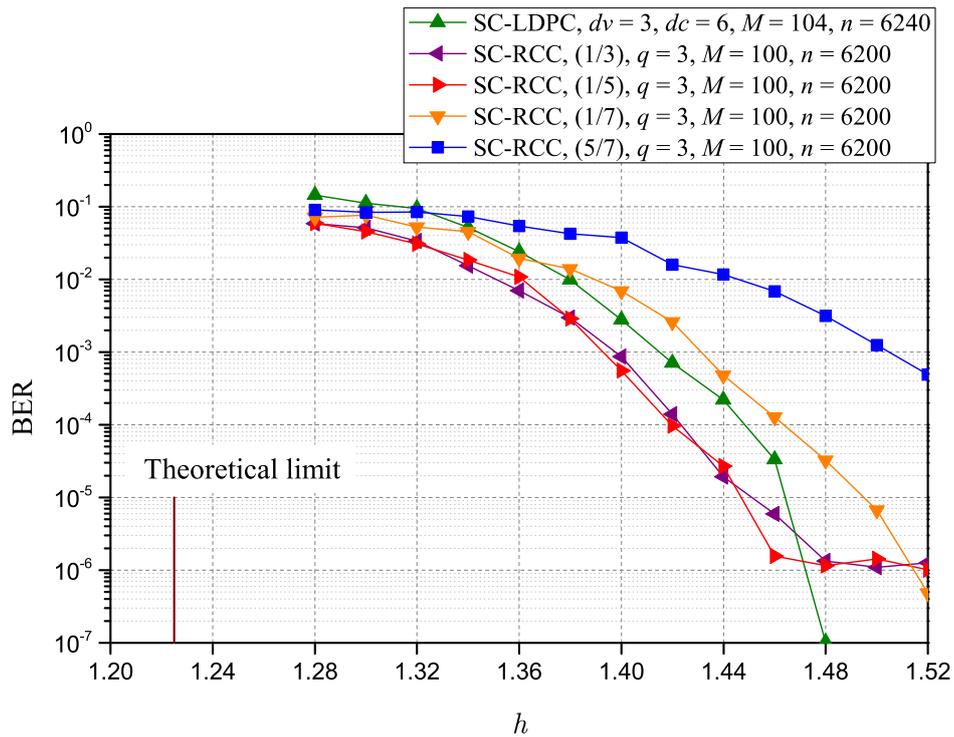


Figure 4.7: BER versus  $h$  of the SC-RCC codes  $\mathcal{S}(3, 30, \mathcal{C}(1/3))$ ,  $\mathcal{S}(3, 30, \mathcal{C}(1/5))$ ,  $\mathcal{S}(3, 30, \mathcal{C}(1/7))$ , and  $\mathcal{S}(3, 30, \mathcal{C}(5/7))$  with  $M = 100$ , and SC-LDPC codes  $\mathcal{L}(dl = 3, dr = 6, L = 30)$  with  $M = 104$ .

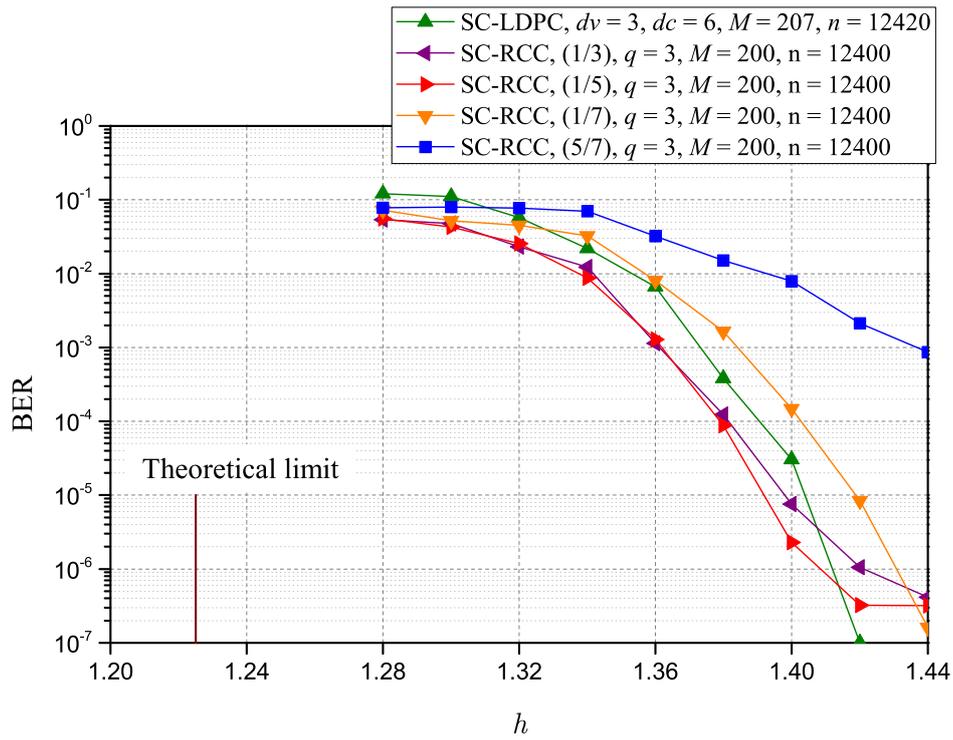


Figure 4.8: BER versus  $h$  of the SC-RCC codes  $\mathcal{S}(3, 30, \mathcal{C}(1/3))$ ,  $\mathcal{S}(3, 30, \mathcal{C}(1/5))$ ,  $\mathcal{S}(3, 30, \mathcal{C}(1/7))$ , and  $\mathcal{S}(3, 30, \mathcal{C}(5/7))$  with  $M = 200$ , and SC-LDPC codes  $\mathcal{L}(dl = 3, dr = 6, L = 30)$  with  $M = 207$ .



# Chapter 5

## Rate-Compatible RA-Extended SC-LDPC Codes

In this chapter, we propose a kind of rate-compatible spatially coupled serially-concatenated codes: RA-extended SC-LDPC codes. The RA-extended SC-LDPC codes are obtained by coupling rate-compatible base codes. We analyze the potential thresholds of base code ensembles, and use them to predict the BP threshold performance of the proposed code ensembles.

### 5.1 Introduction

The rate-compatible code family consists of a set of member codes with different rates, in which the higher rate member codes are embedded into the lower rate member codes and thus

all member codes can be processed by a single encoder and a single decoder [39]. There are two ways to generate rate-compatible code family. One is by puncturing, i.e. starting from a low rate code and then selectly removing some bits to obtain higher rate codes. Another is by extending, i.e. starting from a high rate code and then adding some bits accompanied with new parity-check relations to obtain lower rate codes.

A rate-compatible code family is good in the sense that all member codes are capacity-approaching. Since low-density parity-check (LDPC) codes can approach their corresponding capacities via degree distribution optimization, they are mostly used to construct rate-compatible codes by puncturing [40]- [42] or extending [42]- [44]. However, it is difficult to guarantee that all of the member codes are capacity-approaching, since degree distribution optimization of member codes is constrained due to their embedded construction.

Another approach for constructing good rate-compatible LDPC codes is spatial coupling. Without degree distribution optimization, a capacity-approaching code is obtained by coupling multiple identical non-optimized base codes [13], [14], [26], [50], [51]. Naturally, one can obtain a good family of rate-compatible spatially coupled LDPC (RC-SC-LDPC) codes by coupling multiple identical copies of each member code in a given rate-compatible base code family.

One scheme for constructing RC-SC-LDPC codes is to use spatial coupling to a family of *punctured* rate-compatible LDPC base codes, which are generated by randomly puncturing a regular LDPC code [52], [53]. The RC-SC-LDPC code ensembles achieve a wide and continuous rate range, and their belief propagation (BP) thresholds depend on that of the

lowest rate member code ensemble. When the lowest rate member code ensemble is capacity-approaching, all of the member code ensembles have BP thresholds close to Shannon limits. Like others punctured schemes, all the member codes in the rate-compatible code family employ a single decoder, and their decoding complexities are the same although some bits in higher rate member codes are removed randomly.

Another scheme is to use spatial coupling to a family of *extended* rate-compatible LDPC base codes, which are recursively formed by using three-edge-type (TET) graph extension [54]. More precisely, starting from  $(J, K)$ -regular LDPC code, where  $J$  and  $K$  are variable and check node degrees, a  $(J+i+1, K)$ -regular LDPC code is obtained by extending the graph of  $(J+i, K)$ -regular LDPC code. This provides a family of rate-compatible regular LDPC base codes with the rate increment  $1/K$ . Through spatial coupling, the RC-SC-LDPC codes are obtained, and they have BP thresholds near to Shannon limits at finite discrete rates. Although the larger  $K$  can provides the more discrete rates, the larger  $K$  also leads to the larger BP threshold gap to Shannon limit in low rate region for a given coupling length.

In this chapter, we propose an alternative scheme to construct RC-SC-LDPC codes based on repeat-accumulate (RA) extension, called RA-extended SC-LDPC codes. The rate-compatible base codes, called RA-extended LDPC base codes, for coupling are obtained by extending the variable nodes of a regular LDPC code with a simple and parameter-adjustable RA operation. More precisely, we repeatedly accumulate LDPC code's variable nodes by  $q+1$  times to obtain  $q+1$  blocks of accumulated parity bits. The whole accumulated bits in the first  $q$  blocks and  $\alpha$ -fractional accumulated bits in the last block are transmitted. The

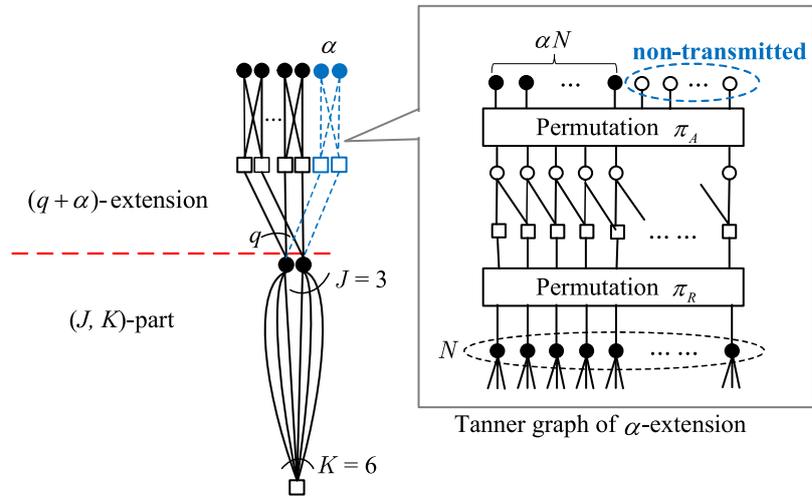
different  $q$  and  $\alpha$  provide different base codes that are rate-compatible. By coupling multiple identical copies of the above RA-extended LDPC base codes, we obtain the RA-extended SC-LDPC codes. The RA-extended SC-LDPC codes with different parameters  $q$  and  $\alpha$  are rate-compatible and achieve arbitrary continuous rates. Although all of the member codes in the proposed RA-extended SC-LDPC code family are decoded by a single decoder with the largest  $q_T$ , a high rate member code with smaller  $q_t$  has lower decoding cost than that with  $q_T$ , since  $q_T - q_t$  accumulated blocks of parity bits are not computed. This is different from conventional randomly punctured schemes.

To investigate iterative decoding performance of the proposed RA-extended SC-LDPC codes, we calculate potential thresholds of the RA-extended LDPC base code ensembles over binary erasure channel (BEC). The potential thresholds are capacity-approaching, which predict the proposed RA-extended SC-LDPC code ensembles are capacity-approaching. This fact is confirmed by BP thresholds of the proposed code ensembles with density evolution (DE) analysis. Compared with the conventional TET-extended scheme, the proposed rate-compatible codes have better BP threshold performance in low rate region.

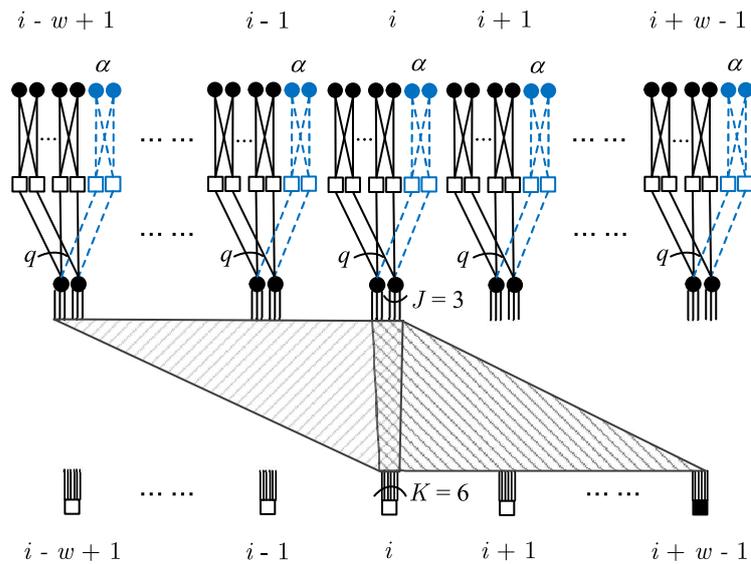
## 5.2 Construction of RA-Extended SC-LDPC Codes

We propose a kind of rate-compatible codes called RA-extended SC-LDPC codes. The proposed codes are obtained by spatially coupling multiple identical base codes.

Before proceeding, we describe the base code ensemble, i.e. RA-extended LDPC



(a)



(b)

Figure 5.1: (a) Protograph of  $\mathcal{B}(J, K, q, \alpha)$  and Tanner graph of  $\alpha$ -extension, and (b) connection diagram at  $i$ -th coupling position of  $\mathcal{C}(J, K, L, w, q, \alpha)$

code ensemble  $\mathcal{B}(J, K, q, \alpha)$ . Figure 5.1(a) illustrate protograph of  $\mathcal{B}(J = 3, K = 6, q, \alpha)$ . The filled circles are variable nodes to be transmitted to channel, and the blank squares are check nodes. RA-extended LDPC code ensemble's protograph consists of  $(J, K)$ -part and  $(q + \alpha)$ -extension part. The  $(J, K)$ -part represents a standard  $(J, K)$ -regular LDPC code ensemble. The  $(q + \alpha)$ -extension part represents a RA coding that repeatedly accumulate LDPC code's variable nodes by  $q + 1$  times to obtain  $q + 1$  blocks of accumulated parity bits. All accumulated bits in the first  $q$  blocks and  $\alpha$ -fractional accumulated bits in the last block are transmitted. Here,  $q$  is a nonnegative integer, and  $\alpha \in [0, 1)$  is a real number. The rate of  $\mathcal{B}(J, K, q, \alpha)$  is given by

$$R_{\mathcal{B}} = \frac{K - J}{K(1 + q + \alpha)}. \quad (5.1)$$

The details of  $(q + \alpha)$ -extension can be described by Tanner graph, which can be obtained by copying its corresponding protograph  $M$  ( $M$  is a positive integer) times and permuting the edges that connect corresponding variable nodes and check nodes in all copies [20]. A  $(q + \alpha)$ -extension includes  $q$  1-extensions and a single  $\alpha$ -extension. In Figure 5.1(a), we also illustrate Tanner graph of  $\alpha$ -extension, in which the blank circles are non-transmitted variable nodes. Let  $N$  be the number of variable nodes in  $(J, K)$ -part's Tanner graph. For example,  $N = 2M$  for  $\mathcal{B}(3, 6, q, \alpha)$ . We first permute the  $N$  variable nodes by  $\pi_R$  and accumulate the permuted ones. Then, through another permutation operation  $\pi_A$  to the  $N$  accumulated nodes, we transmit the first  $\alpha N$  ones (filled circles) and do not transmit the rest  $(1 - \alpha)N$  ones (blank circles). Notably, when  $\alpha = 1$ , the  $\alpha$ -extension is called 1-extension specially.

After describing base code ensemble  $\mathcal{B}(J, K, q, \alpha)$ , we are now ready to introduce its corresponding coupled version, the RA-extended SC-LDPC code ensemble  $\mathcal{C}(J, K, L, w, q, \alpha)$ . To construct  $\mathcal{C}(J, K, L, w, q, \alpha)$ , we copy the protograph of  $\mathcal{B}(J, K, q, \alpha)$  by  $L$  times and place them at  $L$  positions. Then we associate those  $(J, K)$ -parts of adjacent positions, while preserve  $(q + \alpha)$ -extension part of each position unchanged. More precisely, each of  $J$  edges proceeding from a variable node in  $(J, K)$ -part at position  $i$ ,  $0 \leq i \leq L - 1$ , is uniformly and independently connected to one of check nodes in  $(J, K)$ -parts from position  $i$  to  $i + w - 1$ , where coupling width  $w < L$ . Accordingly, each of  $K$  edges proceeding from a check node in  $(J, K)$ -part at position  $i$  is connected to one of variable nodes in  $(J, K)$ -parts from position  $i - w + 1$  to  $i$ . Figure 5.1(b) illustrate diagram of edge connections at coupling position  $i$ . For obtaining termination effect, check nodes are allowed to exist at positions  $L, L+1, \dots, L+w-2$ .

The proposed codes can also be seen as an SC-LDPC mother code  $\mathcal{C}_0(J, K, L, w, 0, 0)$  with  $(q + \alpha)$ -extensions at every coupling positions. The mother code has rate [13]

$$R_0 = \left(1 - \frac{J}{K}\right) - \frac{J}{K} \cdot \frac{w + 1 - 2 \sum_{i=0}^w \binom{i}{w}^K}{L}. \quad (5.2)$$

With  $(q + \alpha)$ -extension, the RA-extended SC-LDPC code ensemble  $\mathcal{C}(J, K, L, w, q, \alpha)$  has rate

$$R = \frac{R_0}{q + \alpha + 1} \quad (5.3)$$

and achieves an arbitrary rate over  $(0, R_0]$  by adjusting RA-extension parameters  $q$  and  $\alpha$ . Note that, when  $J$  is fixed, the larger  $K$  provides the larger  $R_0$  and thus the wider achievable rate interval.

A family of the proposed codes  $\mathcal{C}(J, K, L, w, q, \alpha)$  with various  $q$  and  $\alpha$  have rate compatibility. For nonnegative integer  $q_t$  and  $\alpha_t \in [0, 1)$ , let  $q_{t-1} + \alpha_{t-1} < q_t + \alpha_t$ ,  $t = 1, 2, \dots, T$ . Given mother code  $\mathcal{C}_0(J, K, L, w, q_0, \alpha_0)$  *a priori*, we obtain code  $\mathcal{C}_1(J, K, L, w, q_1, \alpha_1)$  by adjusting encoding operation such that  $(q_1 + \alpha_1)$ -extension is met. Recursively, we obtain code  $\mathcal{C}_t$  from  $\mathcal{C}_{t-1}$ ,  $t = 1, 2, \dots, T$ , and the codes of set  $\{\mathcal{C}_0, \mathcal{C}_1, \dots, \mathcal{C}_T\}$  are rate-compatible.

In the rate-compatible code family  $\{\mathcal{C}_0, \mathcal{C}_1, \dots, \mathcal{C}_T\}$ , all member codes can be decoded by a single decoder of  $\mathcal{C}_T$ . But a member code  $\mathcal{C}_t$ ,  $q_t < q_T$ , has lower decoding complexity than the lowest rate member code  $\mathcal{C}_T$ , since  $q_T - q_t$  1-extensions at each coupling position are not computed. This is different from conventional punctured rate-compatible codes, e.g. [52], where all member codes have the same decoding complexity due to randomly puncturing.

## 5.3 Iterative Decoding Analysis

In this section, we analyze the potential threshold of base code ensemble and BP threshold of the proposed RA-extended SC-LDPC ensemble over BEC with erasure probability  $\epsilon$ .

### 5.3.1 Potential Threshold Analysis

We briefly review the general concept of potential function and threshold.

**Definition 1** ([50]) A single iterative system  $(f, g)$  with parameter  $\epsilon \in [0, 1]$  that is defined

by

$$x^{(l+1)} \triangleq f\left(g(x^{(l)}), \epsilon\right) \quad (5.4)$$

where  $f : [0, 1] \times [0, 1] \rightarrow [0, 1]$  is strictly increasing in both arguments for  $y, \epsilon \in (0, 1]$ , and  $g : [0, 1] \rightarrow [0, 1]$  satisfies that its first-order derivative  $g'(x) > 0$  for  $x \in (0, 1)$ .  $\square$

**Definition 2** ([50]) The potential function  $U(x, \epsilon)$  and potential threshold  $\epsilon^*$  of a single iterative system are defined by

$$U(x, \epsilon) \triangleq \int_0^x \left(z - f(g(z), \epsilon)\right) \cdot g'(z) \, dz \quad (5.5)$$

and

$$\epsilon^* \triangleq \sup\{\epsilon \in [0, 1] \mid \min_{x \in [0, 1]} U(x, \epsilon) \geq 0\}. \quad (5.6)$$

$\square$

Corresponding to the single iterative system  $(f, g)$ , a spatially coupled iterative system  $(f, g, L, w)$  is defined by

$$x_i^{(l+1)} \triangleq \frac{1}{w} \sum_{k=0}^{w-1} f\left(\frac{1}{w} \sum_{j=0}^{w-1} g(x_{i+j-k}^{(l)}), \epsilon\right) \quad (5.7)$$

where  $x_i^{(0)} = \epsilon$  for  $i = 0, 1, \dots, L-1$  and  $x_i^{(0)} = 0$  for else  $i$ , and  $w < L$ . By [50, Theorem 1], if  $\epsilon < \epsilon^*$  and  $w$  is sufficiently large, then spatially coupled system  $(f, g, L, w)$  can converge to  $\mathbf{x} = \mathbf{0}$  for finite iterations. It is shown that several base code ensembles' DE can be described as  $(f, g)$ , and their potential thresholds is the maximum *a posteriori* (MAP) thresholds [50].

Thus, BP thresholds of corresponding coupled ensembles asymptotically saturate to potential thresholds (or MAP thresholds) of these base ensembles [13].

Based on above discussion, we conjecture that, for  $\mathcal{B}(J, K, q, \alpha)$ , its potential threshold is also MAP threshold. If the potential threshold is close to Shannon limit, then we predict that the proposed RA-extended ensemble  $\mathcal{C}(J, K, L, w, q, \alpha)$  has BP threshold close to Shannon limit. We calculate the potential threshold next.

### 5.3.2 Potential Thresholds of Base Code Ensembles

For the base code ensemble  $\mathcal{B}(J, K, q, \alpha)$ , we calculate its potential threshold  $\epsilon_{\mathcal{B}}^*$  by building its DE update as a single iterative system  $(f, g)$  with corresponding  $f$ -function and  $g$ -function.

Let us first observe the protograph in Figure 5.1(a). It is straightforward that if we merge the variable nodes and their  $(q+\alpha)$ -extension into modified (extension-variable) nodes, we can give the single iterative system description of  $\mathcal{B}(J, K, q, \alpha)$ , following that of  $(J, K)$ -regular LDPC code ensemble [50]. For analysis' sake, we draw out a virtual copy of each variable node in  $(J, K)$ -part for  $(q+\alpha)$ -extension. As shown in Figure 5.2, the blank circles are the virtual copies that do not change any of the base code ensemble of Figure 5.1(a). The two ensembles, represented by Figures 5.1(a) and 5.2, are equivalent.

In Figure 5.2, we label the variable node sets in  $(J, K)$ -part, 1-extensions, and  $\alpha$ -extension by  $V_1$ ,  $V_2$ , and  $V_3$ , respectively. Similarly, we label corresponding check node sets

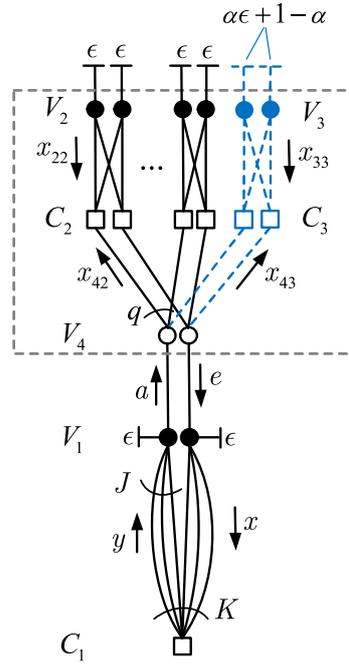


Figure 5.2: Protograph of  $\mathcal{B}(J, K, q, \alpha)$  with virtual copies of variable nodes in  $(J, K)$ -part by  $C_1$ ,  $C_2$ , and  $C_3$ . We also label the virtual copy of  $V_1$  by  $V_4$ . Let  $x_{uv}$  be the erasure probability of the message that is passed from a variable node in set  $V_u$  to a check node in set  $C_v$ ,  $uv \in \{22, 33, 42, 43\}$ . Moreover, let  $a$ ,  $e$ ,  $x$ , and  $y$  be the erasure probability messages outflow from  $V_1$  to  $V_4$ ,  $V_4$  to  $V_1$ ,  $V_1$  to  $C_1$ , and  $C_1$  to  $V_1$ , respectively.

The  $(q + \alpha)$ -extension with node sets  $\{V_2, C_2, V_3, C_3, V_4\}$  forms a sub-system with parameter  $\epsilon$ , input  $a$ , and output  $e$ . We define its *a posteriori* probability (APP) decoding function by

$$e \triangleq \gamma(a, \epsilon). \quad (5.8)$$

To calculate the  $\gamma$ -function, we analyze the messages update in the  $(q + \alpha)$ -extension. For

given  $\epsilon$  and  $a$ , the DE equations are given by

$$x_{42}^{(l+1)} = a \left( 1 - (1 - x_{22}^{(l)})^2 \right)^{q-1} \left( 1 - (1 - x_{33}^{(l)})^2 \right) \quad (5.9)$$

$$x_{43}^{(l+1)} = a \left( 1 - (1 - x_{22}^{(l)})^2 \right)^q \quad (5.10)$$

$$x_{22}^{(l+1)} = \epsilon \left( 1 - (1 - x_{42}^{(l)}) (1 - x_{22}^{(l)}) \right) \quad (5.11)$$

$$x_{33}^{(l+1)} = \left( \alpha \epsilon + 1 - \alpha \right) \left( 1 - (1 - x_{43}^{(l)}) (1 - x_{33}^{(l)}) \right). \quad (5.12)$$

When  $q > 0$  and  $\alpha > 0$ , we update equations (5.9) to (5.12) from  $x_{42}^{(0)} = x_{43}^{(0)} = a$  and  $x_{22}^{(0)} = x_{33}^{(0)} = \epsilon$  until all  $x_{uv}^{(l)}$  do not change with increasing  $l$ ,  $uv \in \{22, 33, 42, 43\}$ . Let these fixed points  $x_{uv}^{(l)} = x_{uv}^\circ$ . We have

$$\begin{aligned} e &= \gamma(a, \epsilon) \\ &= \left( 1 - (1 - x_{22}^\circ)^2 \right)^q \left( 1 - (1 - x_{33}^\circ)^2 \right) \end{aligned} \quad (5.13)$$

where  $x_{22}^\circ$  and  $x_{33}^\circ$  only depend on  $a$  and  $\epsilon$  for given  $q$  and  $\alpha$ . When  $q = 0$  and  $\alpha > 0$ , (5.9) and (5.11) do not exist since 1-extensions do not exist. When  $q = 0$  and  $\alpha = 0$ ,  $e = \gamma(a, \epsilon) = 1$ , since  $(q + \alpha)$ -extension does not exist.

After working out the  $\gamma$ -function, we turn to DE equations of base code ensemble  $\mathcal{B}(J, K, q, \alpha)$ . The equations are given by

$$y^{(l)} = 1 - (1 - x^{(l)})^{K-1} \quad (5.14)$$

$$a^{(l)} = \epsilon (y^{(l)})^J \quad (5.15)$$

$$e^{(l)} = \gamma(a^{(l)}, \epsilon) \quad (5.16)$$

$$x^{(l+1)} = \epsilon (y^{(l)})^{J-1} e^{(l)}. \quad (5.17)$$

Accordingly, the single iterative system  $(f, g)$  description is represented by

$$x^{(l+1)} = f\left(g(x^{(l)}), \epsilon\right) \quad (5.18)$$

where

$$g(x) = 1 - (1 - x)^{K-1} \quad (5.19)$$

$$f(y, \epsilon) = \epsilon y^{J-1} \gamma(\epsilon y^J, \epsilon). \quad (5.20)$$

When  $(q + \alpha)$ -extension does not exist,  $\gamma(\epsilon y^J, \epsilon) = 1$  and thus the  $f$ -function of (5.20) become  $f(y, \epsilon) = \epsilon y^{J-1}$  that is variable node update equation of  $(J, K)$ -regular ensemble [50]. When  $q + \alpha > 0$ , the  $f$ -function of (5.20) is update function of the modified (extension-variable) node, in which factor  $\gamma(\epsilon y^J, \epsilon)$  reflects the effect of  $(q + \alpha)$ -extension. Numerically, we find that  $\gamma$ -function is strictly increasing in  $y, \epsilon \in (0, 1]$  for all  $q$  and  $\alpha$  considered. This is consistent with normal APP decoding behavior. Thus, the  $f$ -function of (5.20) satisfies constraint in Definition 1. With (5.19) and (5.20), we can calculate potential threshold  $\epsilon_{\mathcal{B}}^*$  of  $\mathcal{B}(J, K, q, \alpha)$  by (5.6).

### 5.3.3 BP Thresholds of RA-Extended SC-LDPC Code Ensembles

For confirming the iterative decoding performance, we also use DE to determine BP threshold of the proposed ensemble  $\mathcal{C}(J, K, L, w, q, \alpha)$ . Different from base code ensemble, the message  $x$  in Figure 5.2 will be updated at all coupling positions. Denoted by these messages on the  $l$ -th iteration  $x_i^{(l)}$ ,  $0 \leq i \leq L - 1$ . The DE equation of  $\mathcal{C}(J, K, L, w, q, \alpha)$  is formed as (5.7),

where functions  $g$  and  $f$  are given in (5.19) and (5.20), respectively. By updating of (5.7), we can determine the BP threshold of  $\mathcal{C}(J, K, L, w, q, \alpha)$  by

$$\epsilon^{\text{BP}} \triangleq \sup\{\epsilon \in [0, 1] : \mathbf{x}^{(l)}(\epsilon) \xrightarrow{l \rightarrow \infty} \mathbf{0}\} \quad (5.21)$$

where vector  $\mathbf{x}^{(l)} \triangleq (x_0^{(l)}, x_1^{(l)}, \dots, x_{L-1}^{(l)})$ .

## 5.4 Numerical Results

In this section, we give numerical results of potential thresholds of  $\mathcal{B}(J, K, q, \alpha)$  and BP thresholds of  $\mathcal{C}(J, K, L, w, q, \alpha)$  over BEC. In the computation process of DE, we set  $10^5$  as the maximum iteration number of the variable nodes and halt the iteration when each  $x_{11,i}^{(l)}$  is smaller than the value of  $10^{-7}$ .

We first calculate potential threshold  $\epsilon_{\mathcal{B}}^*$  of the base code ensembles. Let  $\epsilon^{\text{Sh}}$  be the Shannon limit. Denote by potential gap  $G^{\text{P}} \triangleq \epsilon^{\text{Sh}} - \epsilon_{\mathcal{B}}^*$ . Figure 5.3 shows  $G^{\text{P}}$  versus  $q + \alpha$  curves of  $\mathcal{B}(3, K, q, \alpha)$ . We have the following observations.

- The investigated base code ensembles have very small potential gaps ( $\leq 0.053$ ) for all  $q + \alpha$ .
- When  $q + \alpha > 0$ , the smaller  $K$ , the smaller  $G^{\text{P}}$  to Shannon limit.

By discussion in Section 5.3.1, the fact that the base code ensembles have potential thresholds close to Shannon limits predict that the proposed RA-extended SC-LDPC code

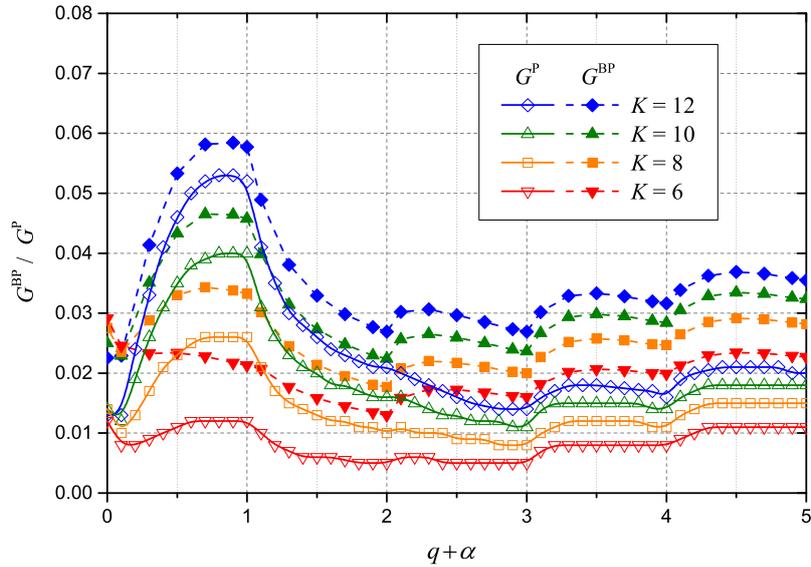


Figure 5.3: Potential gaps of  $\mathcal{B}(3, K, q, \alpha)$  and BP gaps of  $\mathcal{C}(3, K, 100, 5, q, \alpha)$  with  $K = 6, 8, 10$  and  $12$

ensembles, i.e. the coupled version of the base code ensembles, are also capacity-approaching when  $L$  is large, e.g.  $L = 100$ . The BP threshold gap  $G^{\text{BP}} \triangleq \epsilon^{\text{Sh}} - \epsilon^{\text{BP}}$  versus  $q + \alpha$  curves of  $\mathcal{C}(3, K, 100, 5, q, \alpha)$  in Fig. 5.3 verified the prediction.

Then, we turn to rate versus BP threshold performance of the proposed codes. As mentioned in Section 5.2, a family of codes  $\mathcal{C}(J, K, L, w, q, \alpha)$  with various  $q + \alpha$  are rate-compatible. The family has infinite member codes since  $q + \alpha$  can be arbitrary nonnegative real number. Figure 5.4 shows curves of two code families  $\mathcal{C}(3, K, 100, 5, q, \alpha)$  with  $K = 6$  and  $12$ . We see that the code family with  $K = 6$  has BP threshold performance nearer to Shannon limit, and the code family with  $K = 12$  has the wider achievable rate interval as we stated in Section 5.2. This is a trade-off between iterative decoding performance of

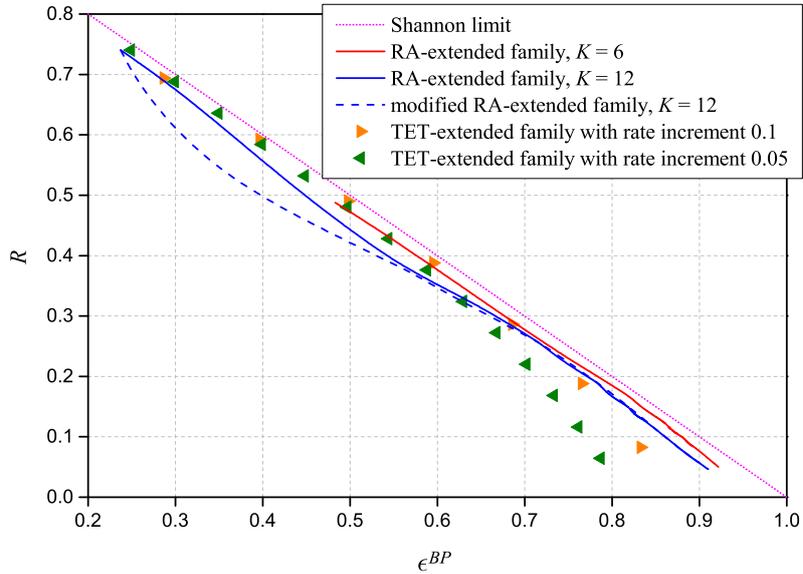


Figure 5.4: Rates versus BP thresholds of codes  $\mathcal{C}(3, K, 100, 5, q, \alpha)$  with  $K = 6$  and  $K = 12$ ,  $\mathcal{C}'(3, K = 12, 100, 5, q, \alpha)$ , and TET-extended rate-compatible code families with rate increments 0.1 and 0.05

code family and width of achievable rate interval. In addition, at the frequently-used specific rates, the above families' parameters, and their BP gaps are also listed in Tables 5.1 and 5.2.

For comparison, we also illustrate the performances of the rate-compatible TET-extended code families [54] as shown in Figure 5.4. From Figure 5.4, we see that there is a obvious loss in BP threshold performance in low rate region when the rate increment becomes small, e.g. from 0.1 to 0.05. On the contrary, our proposed rate-compatible families have better iterative decoding performance in low rate region and stable performance for whole achievable rate interval.

**Remark 1:** Let us turn to a modification of the  $\alpha$ -extension with focusing on the

Table 5.1: Rate-compatible code family of  $\mathcal{C}(3, 6, 100, 5, q, \alpha)$ 

$R$	$q$	$\alpha$	$\epsilon^{\text{Sh}}$	$\epsilon^{\text{BP}}$	$G^{\text{BP}}$
0.45	0	0.0736	0.55	0.5246	0.0254
0.40	0	0.2078	0.60	0.5767	0.0233
0.35	0	0.3803	0.65	0.6266	0.0234
0.30	0	0.6104	0.70	0.6769	0.0231
0.25	0	0.9324	0.75	0.7285	0.0215
0.20	1	0.4157	0.80	0.7835	0.0165
0.15	2	0.2207	0.85	0.8330	0.0170
0.10	3	0.8313	0.90	0.8798	0.0202
0.05	8	0.6620	0.95	0.9218	0.0282

Table 5.2: Rate-compatible code family of  $\mathcal{C}(3, 12, 100, 5, q, \alpha)$ 

$R$	$q$	$\alpha$	$\epsilon^{\text{Sh}}$	$\epsilon^{\text{BP}}$	$G^{\text{BP}}$
0.7	0	0.0577	0.3	0.2797	0.0203
0.6	0	0.2339	0.4	0.3643	0.0357
0.5	0	0.4807	0.5	0.4475	0.0525
0.4	0	0.8509	0.6	0.5413	0.0587
0.3	1	0.4679	0.7	0.6665	0.0335
0.2	2	0.7018	0.8	0.7715	0.0285
0.1	6	0.4036	0.9	0.8587	0.0413

permutation  $\pi_A$  in Figure 5.1(a). If the identity permutation matrix is chosen as  $\pi_A$ , the decoding complexity of the  $\alpha$ -extension will be decreased, compared with the normal random permutation. Obviously, the decoding complexity of the normal  $\alpha$ -extension is equal to that of 1-extension due to the random permutation. Thanks to the identity permutation, the non-transmitted accumulated nodes can be removed from decoding processing. The modified code family  $\mathcal{C}'(3, 12, 100, 5, q, \alpha)$  has almost the same decoding performance in the low rate region without much degradation in the high rate region (see Figure 5.4).

## 5.5 Summary

In this chapter, we proposed a family of capacity-approaching RC-SC-LDPC codes based on parameter-adjustable RA-extension. Arbitrary rates were achieved by simply adjusting RA-extension parameters  $q$  and  $\alpha$ . The potential thresholds were also calculated. The fact that the potential thresholds are close to the Shannon limits means that the proposed codes are capacity-approaching. Compared with conventional TET-extended RC-SC-LDPC code ensembles, the proposed code ensembles achieve continuous rates and perform better in the low rate region.



# Chapter 6

## Concluding Remarks

The spatial coupling is a new technique to structure capacity-approaching codes by associating multiple identical base codes. Without complicated optimization, we employ serially-concatenated codes, which include an outer code and an inner code, as base codes to structure more flexible spatially coupled codes. Since spatial coupling technique guarantees coupled codes to be capacity-approaching, the two component codes in base codes can be selected for some practical factors, such as simple encoding, effective decoding and rate adjustability, and so on.

The first kind of spatially coupled serially-concatenated codes is called SC-RCC codes. The SC-RCC codes are obtained by spatially coupling multiple identical base codes that consist of repeater-combiner outer code and convolutional inner code. In base code, both outer and inner component codes have simple encoder realization. With APP decoder, the

convolutional inner codes with IIR can provide more effective decoding. Thus, the SC-RCC codes also have simple encoder and effective decoder. The decoding analysis with infinite code length shows that the SC-RCC codes can be perfectly decoded  $E_b/N_0$  that are nearer to theoretical limit than conventional spatially coupled codes on AWGNC. With finite code length realizations, the SC-RCC codes perform better than conventional codes. In particular, when the rate is about 0.5, the proposed codes with a code length of 15628 have BER of  $10^{-5}$  at  $E_b/N_0$  of 1.13 dB, which is less than 1 dB away from the Shannon limit. This code length may therefore satisfy the practical requirements of communication systems.

The effect that use SC-RCC codes on Gaussian MAC is investigated. Since SC-RCC codes with finite code length have better decoding performance than conventional codes, we predict they also may be an excellent practical coding schemes on Gaussian MAC. We have analysed iterative detection-decoding with infinite code length and simulated code realization with finite code length . The numerical results show that the SC-RCC codes perform better than conventional spatially coupled codes on Gaussian MAC.

The second kind of spatially coupled serially-concatenated codes is called RA-extended SC-LDPC codes. The design is for rate-compatible communication problem. In some practical system, rate-compatible codes are usually used to adapt changing channel condition with various available coding rates. A family of rate-compatible codes consists of a set of member codes with different rates, in which the higher rate member codes are embedded into the lower rate codes, and all the member codes can be processed by a single encoder and a single decoder. The design problem of rate-compatible is to guarantee all of these embedded

member codes are capacity-approaching. The RA-extended SC-LDPC codes are obtained by coupling multiple identical serially-concatenated base codes, which consist of LDPC outer code and a simple rate-compatible repeat-accumulate (RA) inner code. Since the concatenated LDPC-RA base codes also can be considered as LDPC codes with RA-extension in graph representation, the spatially coupled codes are called RA-extended SC-LDPC codes. The base codes provide simple encoder construction and rate compatibility. Spatial coupling technique make all member codes are capacity-approaching. A potential threshold analysis is used to determine base codes potential thresholds. The fact that the potential thresholds with various rates are near to theoretical limits means the RA-extended SC-LDPC codes with various rate have decoding performance near to theoretical limit. This is confirmed by analysis results of the coupled codes. Compared with conventional rate-compatible coding schemes based on spatial coupling, our codes achieve arbitrary rates in a wide continuous real number interval and perform better than conventional rate-compatible SC-LDPC codes in the low rate region.



# Acknowledgments

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

I want to thank Prof. Ying Li, who is my advisor when I studied for Master Degree. Thanks for her drawing me to the field of channel codes and help me build the foundation of knowledge. I wish to express thanks to Prof. Li Ping of City University of Hong Kong for his guidance and suggestion.

I would like to thank the members in our laboratory, G. Song, S. Lu, M. He, Y. Ren, S. Liu, and H. Han. A number of discussions with them are acknowledged.

The scholarship supplied by Ministry of Education, Culture, Sports, Science and Technology (MEXT) is the financial support to me during this work.

Finally, I would like to thank my family. They always support and encourage me. My happy family is what drives me.



# Bibliography

- [1] C. E. Shannon, “A mathematical theory of communication,” *Bell System Technical Journal*, pp. 379-423 (Part I), pp. 623-656 (Part II), Jul. 1948.
- [2] W. Ryan and S. Lin, *Channel Codes: Classical and Modern*. Cambridge, UK: Cambridge Univ. Press, 2009.
- [3] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge, USA: Cambridge Univ. Press, 2008.
- [4] P. Elias, “Coding for noisy channels,” *IRE Conv. Rec.*, pt. 4, pp. 37-46, Mar. 1955.
- [5] R. Gallager, *Information Theory and Reliable Communication*, New York, Wiley, 1968.
- [6] T. Cover and J. Thomas, *Elements of Information Theory*, 2nd edn., New York, Wiley, 2006.
- [7] D. J. Costello and D. Forney, “Channel coding: the road to channel capacity,” *Proceedings of the IEEE*, vol. 95, no. 6, pp. 1150-1177, Jun. 2007.

- [8] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. IT-13, no. 4, pp. 260-269, Apr. 1967.
- [9] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. 1993 Int. Conf. on Communications (ICC)*, 1993, pp. 1064-1070.
- [10] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Trans. Commun.*, pp. 1261-1271, Oct. 1996.
- [11] R. G. Gallager, *Low-Density Parity-Check Codes*, Cambridge, MA, MIT Press, 1963.
- [12] T. Richardson and R. Urbanke, "Design of capacity-approaching irregular LDPC codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 619-637, Feb. 2001.
- [13] S. Kudekar, T. Richardson, and R. Urbanke, "Threshold saturation via spatial coupling: why convolutional LDPC ensembles perform so well over the BEC," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 803-834, Feb. 2011.
- [14] S. Kudekar, T. Richardson, and R. Urbanke, "Spatially coupled ensembles universally achieve capacity under belief propagation," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 7761-7813, Dec. 2013.
- [15] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Information Theory*, vol. 27, no. 9, pp. 533-547, Sep. 1981.

- [16] T. Richardson and R. Urbanke, "The capacity of LDPC codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599-618, Feb. 2001.
- [17] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727-1737, Oct. 2001.
- [18] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: model and erasure channel properties," *IEEE Trans. Inform. Theory*, vol. 50, no. 11, pp. 2657-2673, Nov. 2004.
- [19] S. Y. Chung, D. Forney, T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Letters*, vol. 5, no. 1, pp. 58-60, Jan. 2001.
- [20] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," Jet Propulsion Laboratory, Pasadena, CA, INP Progress Report pp. 42-154, Aug. 2003.
- [21] T. Richardson and R. Urbanke, "Multi-edge type LDPC codes," submitted to *IEEE Trans. Inform. Theory*, 2004.
- [22] R. Johannesson and K. S. Zigangirov, *Fundamentals of Convolutional Coding*, New York, IEEE Press, 1999.
- [23] G. D. Forney, "Codes on graph: normal realizations," *IEEE Trans. Information Theory*, vol. 47, no. 2, pp. 520-548, Feb. 2001.

- [24] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Information Theory*, vol. 20, no. 3, pp. 284-287, Mar., 1974.
- [25] S. J. Johnson, *Iterative Error Correction*, Cambridge, U.K.: Cambridge Univ. Press, 2010.
- [26] A. J. Felström and K. S. Zigangirov, "Time-varying periodic convolutional codes with low-density parity-check matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 6, pp. 2181-2191, Sep. 1999.
- [27] G. D. Forney, *Concatenated Codes*, Cambridge, MA: MIT Press, 1966.
- [28] S. Moloudi, M. Lentmaier, and A. Graell i Amat, "Spatially coupled turbo codes," in *Proc. 8th Int. Symp. on Turbo Codes and Iterative Information Processing (ISTC)*, pp. 82-86, Bremen, Germany, 2014.
- [29] S. Johnson and G. Lechner, "Spatially coupled repeat-accumulate codes," *IEEE Commun. Letters*, vol. 17, no. 2, pp. 373-376, Feb. 2013.
- [30] W. Hou, S. Lu, and J. Cheng, "Rate-compatible spatially coupled LDPC codes via repeat-accumulation extension," in *Proc. 8th Int. Symp. on Turbo Codes and Iterative Information Processing (ISTC)*, pp. 87-91, Bremen, Germany, 2014.

- [31] A. Yedla, P. Nguyen, H. Pfister, and K. Narayanan, "Universal codes for the Gaussian MAC via spatial coupling," in *Proc. 49th Annual Allerton Conf. on Commun., Control, and Comp.*, Monticello, IL, Sep. 2011, pp. 1801-1808.
- [32] A. El Gamal and Y. H. Kim *Network Information Theory*, Cambridge, U.K.: Cambridge Univ. Press, 2011.
- [33] C. Xu, L. Ping, P. Wang, S. Chan, and X. Lin, "Decentralized power control for random access with successive interference cancellation," *IEEE J. Sel. Areas Commu.*, col. 31, no. 11, pp. 2387-2396, Nov. 2013.
- [34] R. Palanki, A. Khandekar, and R. McEliece, "Graph-based codes for synchronous multiple access channels," in *Proc. 39th Annual Allerton Conf. on Commun., Control, and Comp.*, Monticello, IL, Sep. 2001, pp. 1263-1271.
- [35] A. Amraoui, S. Dusad, and R. Urbanke, "Achieving general points in the 2-user Gaussian MAC without time-sharing or rate-splitting by means of iterative coding," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Lausanne, Switzerland, Jun. 2002, p. 334.
- [36] S. Kudekar and K. Kasai, "Spatially coupled codes over the multiple access channel," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, St. Petersburg, Russia, Jul. 2011, pp. 2816-2820.
- [37] L. Ping, L. Liu, K. Wu, and W. Leung, "Interleaving-division multiple-access," *IEEE Trans. Wireless Commun.*, vol. 5, no. 4, pp. 938-947, Apr. 2006.

- [38] A. Roumy and D. Declercq, "Characterization and optimization of LDPC codes for the 2-user Gaussian multiple access channel," *EURASIP J. on Wireless Commu. and Networking*, vol. 2007, article ID 74890, pp. 1-10, 2007.
- [39] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. 36, no. 4, pp. 389-400, Apr. 1988.
- [40] J. Ha, J. Kim, and S. W. McLaughlin, "Rate-compatible puncturing of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2824-2836, Nov. 2004.
- [41] C. Hsu, and A. Anastasopoulos, "Capacity achieving LDPC codes through puncturing," *IEEE Trans. Inf. Theory*, vol. 54, no. 10, pp. 4698-4706, Oct. 2008.
- [42] G. Yue, X. Wang, and M. Madhian, "Design of rate-compatible irregular repeat accumulate codes", *IEEE Trans. Commun.*, vol. 55, no. 6, pp. 1153-1163, Jun. 2007.
- [43] N. Jacobsen and R. Soni, "Design of rate-compatible irregular LDPC codes based on edge growth and parity splitting," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, Baltimore, America, Sep. 2007, pp. 1052-1056.
- [44] T. V. Nguyen, A. Nosratinia, and D. Divsalar, "The design of rate-compatible protograph LDPC codes," *IEEE Trans. Commun.*, vol. 60, no. 10, pp. 2841-2850, Oct. 2012.
- [45] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja, and D. J. Costello, "LDPC block and convolutional codes based on circulant matrices," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 2966-2984, Dec. 2004.

- [46] M. Lentmaier, A. Sridharan, K. S. Zigangirov, and D. J. Costello, "Terminated LDPC convolutional codes with thresholds close to capacity," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Adelaide, Australia, Sep. 2005, pp. 1372-1376.
- [47] A. E. Pusane, A. J. Feltström, A. Sridharan, M. Lentmaier, K. S. Zigangirov, and D. J. Costello, "Implementation aspects of LDPC convolutional codes," *IEEE Trans. Commun.*, vol. 56, no. 7, pp. 1060-1069, Jul. 2008.
- [48] M. Lentmaier, A. Sridharan, D. J. Costello, and K. S. Zigangirov, "Iterative decoding threshold analysis for LDPC convolutional codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 5274-5289, Oct. 2010.
- [49] A. E. Pusane, R. Smarandache, P. O. Vontobel, and D. J. Costello, "Deriving good LDPC convolutional codes from LDPC block codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 835-857, Feb. 2011.
- [50] A. Yelda, Y. Jian, P. Nguyen, and H. Pfister, "A simple proof of threshold saturation for coupled scalar recursions," in *Proc. Int. Symp. Turbo Codes Iterative Inf. Processing (ISTC)*, Golthenburg, Sweden, Aug. 2012, pp. 51-55.
- [51] A. Yelda, Y. Jian, P. Nguyen, and H. Pfister, "A simple proof of Maxwell saturation for coupled scalar recursions," *IEEE Trans. Inf. Theory*, vol. 60, no. 11, pp. 6943-6965, Nov. 2014.

- [52] D. Mitchell, M. Lentmaier, A. Pusane, and D. J. Costello, "Randomly punctured spatially coupled LDPC codes," in *Proc. Int. Symp. on Turbo Codes and Iterative Information Processing (ISTC)*, Bremen, Germany, Aug. 2014, pp. 1-6.
- [53] D. Mitchell, M. Lentmaier, A. Pusane, and D. J. Costello, "Approximating decoding thresholds of punctured LDPC code ensembles on the AWGN channel," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Hong Kong, China, Jun. 2015, pp. 421-425.
- [54] Z. Si, R. Thobaben, and M. Skoglund, "Rate-compatible LDPC convolutional codes achieving the capacity of the BEC," *IEEE Trans. Inf. Theory*, vol. 58, no. 6, pp. 4021-4029, Jun. 2012.

# My Publications

1. W. Hou, S. Lu, and J. Cheng, “Spatially coupled repeater-combiner-convolutional codes,” *IEEE Communications Letter*. (accepted for publication)
2. W. Hou, S. Lu, and J. Cheng, “Rate-compatible spatially coupled LDPC code ensembles based on repeat-accumulate extensions,” *IET Communications*. (submitted)
3. S. Lu, W. Hou, and J. Cheng, “A family of  $(k+1)$ -ary signature code for noisy multiple-access adder channel,” *IEEE Trans. Information Theory*, vol. 65, no. 11, pp. 5848-5853, Nov. 2015.
4. W. Hou, S. Lu, and J. Cheng, “Rate-compatible spatially coupled LDPC codes via repeat-accumulation extension,” in *Proc. 8th Int. Symp. on Turbo Codes and Iterative Information Processing (ISTC)*, Bremen, Germany, Aug. 2014, pp. 87-91.
5. S. Lu, W. Hou, and J. Cheng, “Coding scheme for T-user noisy multiple-access adder channel,” in *Proc. Int. Symp. on Information Theory and its Applications (ISITA)*, Melbourne, Australia, Oct. 2014, pp. 536-540.

6. S. Lu, W. Hou, and J. Cheng, "Construction of error-correcting signature code on Hadamard matrix," in *Proc. IICREST Third Int. Symp. Radio System and Space Plasma (ISRSSP)*, Sofia, Bulgaria, Aug. 2013, pp. 57-63.
7. S. Lu, J. Cheng, W. Hou, and Y. Watanabe, "Generalized construction of signature code for T-user multiple-access adder channel," in *Proc. IEEE Int. Symp. on Information Theory (ISIT)*, Istanbul, Turkey, Jul. 2013. pp. 1655-1659.
8. S. Lu, W. Hou, and J. Cheng, "Ensemble weight enumerators for spatially coupled repeater-combiner-convolutional codes," in *Proc. the 38th Symp. on Information Theory and Its Applications (SITA)*, Kojima, Okayama, Japan, Nov. 2015, pp. 96-100.
9. W. Hou, S. Lu, and J. Cheng, "Concatenated codes by spatial coupling," in *Proc. the 37th Symp. on Information Theory and Its Applications (SITA)*, Unazuki, Toyama, Japan, Dec. 2014, pp. 477-481.
10. S. Lu, W. Hou, and J. Cheng, "A family of T-user codes for noisy multiple-access adder channel," in *Proc. the 37th Symp. on Information Theory and Its Applications (SITA)*, Unazuki, Toyama, Japan, Dec. 2014, pp. 645-649.
11. A. Kimura, W. Hou, M. Yoshida, and J. Cheng, "Rate-compatible spatially coupled repeat-accumulate codes," in *Proc. the 37th Symp. on Information Theory and Its Applications (SITA)*, Unazuki, Toyama, Japan, Dec. 2014, pp. 472-476.
12. W. Hou, S. Lu, and J. Cheng, "Rate compatibility of spatially coupled LDPC codes

- via repeat-accumulation extension,” in *Proc. the 36th Symp. on Information Theory and Its Applications (SITA)*, Ito, Shizuoka, Japan, Nov. 2013, pp. 244-248.
13. S. Lu, W. Hou, and J. Cheng, “Coding scheme of error-correcting signature codes from Hadamard matrix,” in *Proc. the 36th Symp. on Information Theory and Its Applications (SITA)*, Ito, Shizuoka, Japan, Nov. 2013, pp. 632-637.
  14. S. Lu, W. Hou, and J. Cheng, “Multiuser coding by difference matrices for multiple-access adder channel,” *IEICE Workshop on Error-Correcting Coding*, Okinawa, Japan, Sep. 2013.
  15. S. Lu, W. Hou, and J. Cheng, “A family of error-correcting signature code,” in *Proc. the 51st Joint Symp. on Research Centers, Science and Engineering Research Institute*, Doshisha University, Kyoto, Japan, Dec. 2013, pp. 121-126.
  16. W. Hou, S. Lu, J. Cheng, and Y. Watanabe, ”Preprocess scheme of errorcorrection code for physical layer security,” in *Proc. IEICE General Conf.*, Okayama, Japan, Mar. 2012, B-5-5, p. 404.
  17. Y. Li, W. Hou, and Y. Sun, “Low-density parity check code-based relay cooperative transmission method,” China Patent CN101826935, 2014.