2017

Efficient Transmission Techniques in Cooperative Networks: Forwarding Strategies and Distributed Coding Schemes

Xuanxuan Lu
Lehigh University

Efficient Transmission Techniques in Cooperative Networks: Forwarding Strategies and Distributed Coding Schemes

by

Xuanxuan Lu

Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy for the Degree of

Doctor of Philosophy

in

Electrical Engineering

Lehigh University

January 2017

Approved and recommended for acceptance as a dissertation in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

_____

Date

_____

Dissertation Advisor

_____

Accepted Date

Committee Members:

_____

Prof. Tiffany Jing Li(Chair)

_____

Prof. Zhiyuan Yan

_____

Prof. Parvathinathan Venkitasubramaniam

_____

Prof. Ping-Shi Wu

Acknowledgements

Pursuing Ph.D degree is a long and tedious process. I could not imagine finishing it without the financial and spiritual support from numerous people.

First, my heartily acknowledgment goes to my advisor Tiffany Jing Li, who provided valuable advice, patience and encouragement in all the time of my research. Her diligence and passion have strongly influenced me and inspired me to devote myself to my studies in the past years.

Second, I like to express my great appreciation to Prof. Zhiyuan Yan, Prof. Parvathinathan Venkitasubramaniam, and Prof. Ping-Shi Wu, for serving on my doctoral committee and spending their precious time on reviewing my dissertation. I show my gratitude to Dr. Kapil Chawla and Dr. Shuping Gong in Broadcom Inc., who gave me valuable advice and suggestion during my internship in Broadcom Inc.

My appreciation also goes to all my fellow students Chuanming Wei, Yang Yang, Peiyu Tan, Kai Xie, Xingkai Bao, Chen Chen, Hongmei Xie, Feng Shi, Chenrong Xiong, Jun Lin, Yang Liu, and other friends in Packard Lab 401, who help me when I meet obstacle in research, support me when I feel frustrated in life.

I would like to give my special thanks to my parents, Maolin and Yingchun, on whose constant loving support I have relied throughout my time to complete my Ph.D. study. Last but not least, I am very grateful to my husband Jiangfan, for his love and patience during the past five years. This period has more special meaning to us because of the birth of our daughter Claire Siran Zhang, who provided an additional joyful dimension to our life. Her Chinese name “Siran" comes from the maximum likelihood ratio rule, which is widely used in the area of signal processing.

This dissertation is dedicated to my advisor, my parents, my husband and my daughter.

Contents

Acknowledgements iv

List of Tables x

List of Figures xi

Abstract 1

1 Introduction **3**

1.1 Background and Motivation 3

1.2 Outline of the Dissertation 7

2 Analog-Encode-Forward (AEF) Strategy for Single Relay Systems **9**

2.1 Introduction and Motivation 9

2.2 Efficient and Practical Analog Code 13

2.2.1 Encoding of Mirrored Baker's Map Codes 14

2.2.2 Decoding of Mirrored Baker's Map Codes 17

2.3 System Model for User Cooperation 20

2.3.1 Relay System Model 20

2.3.2 Traditional Relaying Schemes 21

2.4 AEF and AEF-DF Scheme 24

2.4.1 Analog-Encode-Forward (AEF) with ML Analog Decoder . . . 24

2.4.2 Hybrid AEF-DF 25
2.4.3 Simulation Results 26
2.5 A New Analog-Encode-Forward (AEF) Strategy and MAP Analog Decoder 27
2.5.1 AEF with MAP Analog Decoder 28
2.5.2 Simulation Results 33

3 Z-Forward Strategy for Parallel Relay Systems 36
3.1 Introduction and Motivation 36
3.2 System Model 39
3.3 Representation of Soft Messages 41
3.4 Traditional Strategies and Z-forward Strategy 45
3.5 Threshold Selection in Single-Relay Systems 49
3.6 Threshold Selection in Multi-Relay Systems 54
3.6.1 BER Performance in Two-relay Systems 55
3.6.2 Sub-optimal Z-forward in Multi-relay Systems 62
3.7 Estimation at Destination 63
3.8 Numerical Results 65

4 New Soft-Encoding Relay (SoER) Mechanisms for Parallel Relay Systems: Convolutional and Turbo Constructions 72
4.1 Introduction and Motivation 72
4.2 System Model 76
4.3 Proposed Distributed Soft-Encoding Codes 78
4.3.1 General Idea of Soft Encoding 78
4.3.2 Choice of Soft Messages 80

4.4 SISO Convolutional Codes 85
4.4.1 SISO Convolutional Encoder 85
4.4.2 SISO Viterbi Decoder using Gaussian Approximation 86
4.4.3 SISO Viterbi Decoder Using A More Accurate PDF 90
4.5 SISO Turbo Codes 99
4.5.1 Recursive SISO Convolutional Codes and Distributed SISO Turbo Codes 100
4.5.2 BCJR Convolutional Decoder and Iterative Turbo Decoder . . . 100
4.6 Analysis 102
4.6.1 Diversity Order Analysis 102
4.6.2 Code Selection for Feed-forward Soft-Encoding 104
4.6.3 Threshold Selection 105
4.7 Numerical Results 107

5 Cooperative Forward Strategy through Signal-Superposition-Based Braid Coding 115

5.1 Introduction and Motivation 115
5.1.1 Related Work 117
5.1.2 Novelty and Contributions 119
5.2 Braid Coding Cooperative Scheme 122
5.3 Code Optimization 127
5.4 Decoding Algorithm 131
5.4.1 Viterbi Decoding of Regenerative Code 131
5.4.2 Linear Detector of Nonregenerative Code 133
5.5 Theoretical Performance Analysis 135
5.5.1 Free distance d 135

5.5.2 Diversity Order 137
5.5.3 Bit Error Rate Performance of Adaptive Transmission Scheme . 141
5.6 Forward and Backward Message Passing Iterative Decoding 146
5.7 Simulation Results 150

6 Conclusions **157**

Bibliography **159**

References **159**

Vita **170**

List of Tables

3.1 Optimal LLR thresholds θ vs the SNR (dB) of the channels SR and RD in Z-forward strategy in a single-relay system 51

3.2 Optimal normalized thresholds $\beta\theta$ vs the SNR (dB) of the channels SR and RD in Z-forward strategy in a single-relay system 52

3.3 Optimal LLR thresholds of Z-forward in a double-relay system, $\text{SNR}_{SR_1} = \text{SNR}_{SR_2}$, $\text{SNR}_{R_1D} = \text{SNR}_{R_2D}$, $\theta_1 = \theta_2$ 60

3.4 Optimal LLR thresholds of Z-forward in a double-relay system, $\text{SNR}_{SR_2} = \text{SNR}_{SR_1} - 3 = \text{SNR}_{R_2D} = \text{SNR}_{R_1D} - 3$, $\theta_1 \neq \theta_2$ 60

3.5 Optimal LLR thresholds of Z-forward a double-relay system, $\text{SNR}_{SR_2} = \text{SNR}_{SR_1} - 3$, $\text{SNR}_{R_2D} = \text{SNR}_{R_1D} - 3$, $\theta_1 = \theta_2$ 60

3.6 Simulated forward schemes 65

List of Figures

2.1 A 2-hop system, where the source must rely on the relay to deliver the message to the destination. 13

2.2 The baker's map. 14

2.3 Mirrored baker's map codes. 16

2.4 The proposed analog-encode-forward (AEF) scheme. 24

2.5 BER of different schemes at S-R SNR of 2dB 28

2.6 BER of different schemes at S-R SNR of 3dB 29

2.7 Analog-Encode-Forward with a MAP decoder. 30

2.8 Observed decoder-LLRs at the relay and its Gaussian approximation,α=0.54 34

2.9 Observed decoder-LLRs at the relay and its Gaussian approximation,α=0.4 34

2.10 BLER comparison of different schemes at S-R SNR of 28dB on block rayleigh fading channels 35

3.1 System model 36

3.2 Relaying function in different forward strategies 47

3.3 Optimal LLR thresholds of Z-forward strategy with different SNR_{SR} and SNR_{RD} in a single-relay system 55

3.4 Optimal LLR thresholds of Z-forward in a double-relay system 61

3.5 BER performance of Z-forward with different thresholds under AWGN channel with 2 relay nodes, MRC estimate, all the average channel SNRs are the same. 68
3.6 BER performance of different schemes under block Rayleigh fading channel with 1-3 relay nodes, MRC estimate, all the average channel SNRs are the same. 69
3.7 BER performance of different schemes under block Rayleigh fading channel with 2 or 3 relay nodes, MRC estimate, $\overline{\text{SNR}}_{SR_1} = \overline{\text{SNR}}_{R_1D} = \overline{\text{SNR}}_{SR_2} + 3 = \overline{\text{SNR}}_{R_2D} + 3 = \overline{\text{SNR}}_{SR_3} = \overline{\text{SNR}}_{R_3D}$. 69
3.8 BER performance of different schemes under block Rayleigh fading channel with 2 or 3 relay nodes, ML and MRC estimate, all the average channel SNRs are the same 70
3.9 BER performance of different schemes under block Rayleigh fading channel with 2 or 3 relay nodes, ML and MRC estimate, $\overline{\text{SNR}}_{SR_1} = \overline{\text{SNR}}_{R_1D} = \overline{\text{SNR}}_{SR_2} + 3 = \overline{\text{SNR}}_{R_2D} + 3 = \overline{\text{SNR}}_{SR_3} = \overline{\text{SNR}}_{R_3D}$. 70
3.10 BER performance comparison of Z-forward schemes and different adaptive AF-DF schemes under block Rayleigh fading channel with 2 relay nodes, all the average channel SNRs are the same. 71
3.11 BER performance comparison of Z-forward schemes and different relay selective schemes under block Rayleigh fading channel with 2 relay nodes, all the average channel SNRs are the same. 71

4.1 A two-hop parallel-relay system 76
4.2 Curves of $tanh(L_x/2)$ and rLLR functions with $\theta = 2, 8, 20$. 83
4.3 Illustration of encoding process 85

4.4 PDF of y_{R_iD} with comparison to Gaussian distribution, $SNR_{SR_i} = SNR_{R_iD} = 6$ (dB), $\theta = 2, 8, 20$, $g = (111)$. Dashed lines correspond to $\theta = 2$, solid lines correspond to $\theta = 8$, dot-dash lines correspond to $\theta = 20$. In each case, dark red line depicts the PDF using Gaussian approximation, and light blue line denotes the actual PDF of y_{R_iD}. . . . 91

4.5 PDF of y_{R_1D} and y_{R_2D} with comparison to the Gaussian distribution and the theoretical PDF in (37). 98

4.6 Illustration of encoding process 101

4.7 BER comparison of conventional schemes and limited-LLR based soft coding with different thresholds under AWGN channel, $(5, 7)$ distributed code, all source-relay and relay-destination channels are of the same SNR, $SNR_{SR_1} = SNR_{SR_2} = SNR_{R_1D} = SNR_{R_2D}$ (dB). 109

4.8 BER comparison of conventional schemes and limited-LLR based soft coding with different thresholds under AWGN channel, one source-relay channel is 3 dB better than others, $SNR_{SR_1} = SNR_{SR_2} - 3 = SNR_{R_1D} = SNR_{R_2D}$ (dB). 110

4.9 FER comparison of different schemes under Rayleigh block fading channel, $(5, 7)$ or $(15, 17)$ distributed code, all source-relay and relay-destination channels are of the same average SNR, $SNR_{SR_1} = SNR_{SR_2} = SNR_{R_1D} = SNR_{R_2D}$ (dB). 111

4.10 FER comparison of different schemes under Rayleigh block fading channel, $(5, 7)$ or $(15, 17)$ distributed code, the average SNRs of source-relay channels are 5 dB better than the relay-destination channels, $SNR_{SR_1} - 5 = SNR_{SR_2} - 5 = SNR_{R_1D} = SNR_{R_2D}$ (dB). 112

4.11 FER comparison of distributed turbo code under Rayleigh block fading channel, generated by $(1, \frac{1}{1+D})$, $SNR_{SR_1} = SNR_{SR_2} = SNR_{R_1D} = SNR_{R_2D}$ (dB). 113

5.1 LSR structure of nonregenerative (left) and regenerative $m = 2$ (right) braid coding. 126

5.2 (a) ASK with a non-uniform constellation under a given power constraint; (b) ASK with a uniform constellation without power normalization; (c) Normalized ASK with a uniform constellation. 128

5.3 Trellis for regenerative code $(b_0+b_1D+b_2D^2)$ (solid lines are associated with input -1 and dashed lines are associated with input $+1$) 132

5.4 State diagram of the transmitted signal at each user 142

5.5 An elementary decoder structure 148

5.6 BER vs. SNR (dB) of S_1-D channel for 2-user braid coding cooperative systems with different weights, $m = 1, 2, 3$, $N = 4$, Rayleigh fading, $\text{SNR}_{\mathsf{S}_1\mathsf{D}} = \text{SNR}_{\mathsf{S}_2\mathsf{D}}$, $\text{SNR}_{\mathsf{S}_1\mathsf{S}_2} = \text{SNR}_{\mathsf{S}_1\mathsf{D}} + 20$dB. 151

5.7 BER vs. SNR (dB) of S_1-D channel for 2-user braid coding cooperative systems with different weights, $m = 1, 2, 3$, $N = 4$, AWGN, $\text{SNR}_{\mathsf{S}_1\mathsf{D}} = \text{SNR}_{\mathsf{S}_2\mathsf{D}} + 4$dB, $\text{SNR}_{\mathsf{S}_1\mathsf{S}_2} = \text{SNR}_{\mathsf{S}_2\mathsf{D}} + 6$dB. 152

5.8 BER vs. SNR (dB) of S_1-D channel for 2-user braid coding cooperative systems with different block lengths, $m = 1, 2, 3$, Rayleigh fading, $\text{SNR}_{\mathsf{S}_1\mathsf{D}} = \text{SNR}_{\mathsf{S}_2\mathsf{D}}$, $\text{SNR}_{\mathsf{S}_1\mathsf{S}_2} = \text{SNR}_{\mathsf{S}_1\mathsf{D}} + 20$dB. 153

5.9 BER vs. SNR (dB) of S_1-D channel for 3-user braid coding cooperative systems with different weights, $m = 1, 2$, Rayleigh fading, $\text{SNR}_{\mathsf{S}_1\mathsf{D}} = \text{SNR}_{\mathsf{S}_2\mathsf{D}} = \text{SNR}_{\mathsf{S}_3\mathsf{D}}$, $\text{SNR}_{\mathsf{S}_1\mathsf{S}_2} = \text{SNR}_{\mathsf{S}_1\mathsf{S}_3} = \text{SNR}_{\mathsf{S}_2\mathsf{S}_3} = \text{SNR}_{\mathsf{S}_1\mathsf{D}} + 15$dB. 154

5.10 BER vs. SNR (dB) of S_1-D channel for 4-user braid coding cooperative systems with different weights, $m = 2, 3$, Rayleigh fading, SNRs of all channels $\mathsf{S_i D}$ are equal, SNR of the inter-user channels $\text{SNR}_{\mathsf{S_i S_j}} = \text{SNR}_{\mathsf{S_i D}} + 15dB$. 155

5.11 BER vs. SNR (dB) of S_1-D channel for the 2-user nonregenerative braid coding cooperative systems with ML, Linear, MMSE, and PE detectors, $N = 4$, Rayleigh fading, $\text{SNR}_{\mathsf{S_1 D}} = \text{SNR}_{\mathsf{S_2 D}}$, $\text{SNR}_{\mathsf{S_1 S_2}} = \text{SNR}_{\mathsf{S_1 D}} + 20$dB. 156

5.12 BER vs. SNR (dB) of S_1-D channel for 2-user braid coding cooperative systems with different decoding methods and numbers of iterations, source encoded by $(1, 1/1 + D)$, $m = 2$, $N = 4$, Rayleigh fading, $\text{SNR}_{\mathsf{S_1 D}} = \text{SNR}_{\mathsf{S_2 D}}$, $\text{SNR}_{\mathsf{S_1 S_2}} = \text{SNR}_{\mathsf{S_1 D}} + 15$dB. 156

Abstract

This dissertation focuses on transmission and estimation schemes in wireless relay network, which involves a set of source nodes, a set of destination nodes, and a set of nodes helps communication between source nodes and destination nodes, called relay nodes. It is noted that the overall performance of the wireless relay systems would be impacted by the relay methods adopted by relay nodes. In this dissertation, efficient forwarding strategies and channel coding involved relaying schemes in various relay network topology are studied.

First we study a simple structure of relay systems, with one source, one destination and one relay node. By exploiting "analog codes" – a special class of error correction codes that can directly encode and protect real-valued data, a soft forwarding strategy – "analog-encode-forward (AEF)" scheme is proposed. The relay node first soft-decodes the packet from the source, then re-encodes this soft decoder output (Log Likelihood Ratio) using an appropriate analog code, and forwards it to the destination. At the receiver, both a maximum-likelihood (ML) decoder and a maximum a posterior (MAP) decoder are specially designed for the AEF scheme.

The work is then extended to parallel relay networks, which is consisted of one source, one destination and multiple relay nodes. The first question confronted with us is which kind of soft information to be relayed at the relay nodes. We analyze a set of prevailing soft information for relaying considered by researchers in this field. A truncated LLR is proved to be the best choice, we thus derive another soft forwarding strategy – "Z" forwarding strategy. The main parameter effecting the overall performance in this scheme is the threshold selected to cut the LLR information. We analyze the threshold selection at the relay nodes, and derive the exact ML estimation at the

destination node.

To circumvent the catastrophic error propagation in digital distributed coding scheme, a distributed soft coding scheme is proposed for the parallel relay networks. The key idea is the exploitation of a rate-1 soft convolutional encoder at each of the parallel relays, to collaboratively form a simple but powerful distributed analog coding scheme. Because of the linearity of the truncated LLR information, a nearly optimal ML decoder is derived for the distributed coding scheme.

In the last part, a cooperative transmission scheme for a multi-source single-destination system through superposition modulation is investigated. The source nodes take turns to transmit, and each time, a source "overlays" its new data together with (some or all of) what it overhears from its partner(s), in a way similar to French-braiding the hair. We introduce two subclasses of braid coding, the nonregenerative and the regenerative cases, and, using the pairwise error probability (PEP) as a figure of merit, derive the optimal weight parameters for each one. By exploiting the structure relevance of braid codes with trellis codes, we propose a Viterbi maximum-likelihood (ML) decoding method of linear-complexity for the regenerative case. We also present a soft-iterative joint channel-network decoding. The overall decoding process is divided into the forward message passing and the backward message passing, which makes effective use of the available reliability information from all the received signals. We show that the proposed "braid coding" cooperative scheme benefits not only from the cooperative diversity but also from the bit error rate (BER) performance gain.

Chapter 1

Introduction

1.1 Background and Motivation

The modern communication system is undergoing a profound paradigm shift from point-to-point to multi-terminal communication due to exploding demand for high spectrum efficiency and low error rate. By employing supportive relays, large overall system gain could be achieved because of pathloss gain and diversity gain. This merit can also be translated into lower transmission power, and better coverage. It is analyzed that the cooperative gain in wireless relay networks depends well on the transmission strategy of the relay nodes and the decoding method at the destination nodes under the same physical condition.

A good number of practical signal relaying strategies have been proposed, including *amplify-forward* (AF) for the non-regenerative strategies and *decode-forward* (DF) for the regenerative strategies [17]. Geometric analysis and channel-metric based studies

show that AF and DF each has its advantages with respect to different relative locations [18] and different signal-to-noise ratios (SNR) [19]. One way to combine the merits of AF and DF is via opportunistic switching and time sharing, and several useful switching criteria based on SNR and CRC (cyclic redundancy check) have been proposed [20] [21]. For multi-relay systems, there is also the choice for opportunistic selection using relay selection strategies based on, for example, SNR [22] [23] and log-likelihood ratio (LLR) [24] [25]. However, the more challenging case is when all the (instantaneous) relay channels are in fairly bad conditions, in which case neither opportunistic switching (between AF or DF) nor opportunistic selection (among the relays) does much help. To solve this challenge or achieve the so-called "channel recycling" goal of user cooperation – namely, to combine individually useless channels and make them useful again – requires more sophisticated operations. To this end, researchers have attempted *estimate-forward* (EF) [19] [26] that blends the key aspects of DF and AF: signal processing of DF and soft-forward of AF. Practical strategies include, for example, *decode-amplify-forward* (DAF) [3], and *soft-decoding-forward* (SDF) [27] [28] [29], and *soft-encoding-forward* [30]. These strategies generalize the conventional DF practice by allowing the relay to soft-decode the received signal, and then generate a function, rather than a pure replication, of the source signal, where the function may either reflect a level of reliability estimate of the signal, or be a transformation of the signal in some signal or codeword space. Such generalization promises additional gains in many scenarios and especially when channels are less than desirable. For example, it is shown in [3] that in the low source-relay SNR region, decode-amplify-forward (which instructs the relay to soft-decode the reception, amplify the decoder-LLRs, and then forward them to the destination) can double the capacity of AF and DF.

This dissertation is primarily interested in developing an optimal way to achieve signal forwarding in a 2-hop relay network. We focus on non-bandwidth-reduction forwarding (i.e. not compress-forward). The questions that confront us include: What type of information should be forwarded, what function best captures this information, and how should the relay(s) and destination operate. A new, practical soft relaying scheme is developed by us in one chapter.

To exploit the channel coding gain in wireless relay network, we also consider how to best perform channel coding at the relay nodes and decode it at the destination. Traditional approaches to relay signals start with quantization, followed by a digital error-correction code, so that the quantized signals can be recovered with a desired probability of error. Low-order quantization is simple but tends to introduce severe granularity error that is irreversible at the destination; whereas high-order quantization inevitably causes a considerable increase in complexity, data volume and bandwidth consumption. An alternative to the legacy quantize-and-code approach is to encode the signals directly in the analog domain, resulting in a soft-in soft-out mapping that completely circumvents quantization. Such strategies, known either as analog coding or joint source-channel coding, dated back to when Shannon established the separation theorem. It is well-known that performing source coding (quantization) and channel coding separately in tandem does not necessarily cause performance sub-optimality, and this is what underpins the practice of quantize-and-code. However, separate coding does suffer from a serious practicality drawback: the *robustness* issue. Namely, a separate coding approach performs well only near the "designed point"; as soon as the channel condition slides away from the designed point, the system performance drastically degrades. In comparison, a direct analog coding approach has the potential to perform well in a wide range of channel conditions. One chapter thus considers to perform analog coding at the relay

nodes in single-relay system.

In multi-relay network, we consider how to design a good distributed code. A pioneering paper, [6] was the first to propose a soft distributed code, which uses the soft estimate information (hyperbolic tangent function) to circumvent the error propagation caused by digital codes. A highly-efficient shift register encoder was then developed which uses the unbounded LLRs as the soft input [52]. There is also the proposal of soft-input soft-output (SISO) encoder based on the modified BCJR algorithm [46], and the study of its applicability to higher-order modulation systems [**?**]. A complexity comparison is made between the shift-register encoder [52] and the BCJR-based encoder [46], and analysis favors the former [54].

To assist the design of practical decoders for these SISO codes and to evaluate the decoder performance, [55] explored a wide-spread tool of Gaussian assumption, which approximates the LLR (the soft information for the relay-destination transmission) as Gaussian distributed. This assumption is fairly accurate at low source-relay SNRs, but generates an undesirable discrepancy at in the high SNR region. Another study [56], which targets the case of minimum mean square error (MMSE) soft information forwarding (i.e. uses tanh function to describe the soft messages), proposes to divide the soft information into two components that involve hard errors and soft errors, respectively. The work is further extended by [58], where a practical SISO encoder and decoder is designed for the tanh function. Again, Gaussian approximation is used in the study to approximate the probability density function (PDF) of the output from the SISO encoder. However, it is demonstrated by us that the Gaussian approximation is not accurate, which would degrade the overall BER performance. This motives us to consider which is the best soft information employed by the encoder at each relay and how to

best decode them.

We also consider the efficient transmission for the multi-source single-destination cooperative systems. The traditional store-and-forward is easy to implement, but hard to achieve the network capacity. However, the inception of network coding changed the situation. Noteworthy realization of network coding includes analog network coding (superposition in signal domain), and digital network coding (superposition in analog domain). The traditional digital network coding would bring in severe error propagation like DF schemes. We are interested in developing practical cooperation mechanisms that allow the system to tap the gains promised by network coding (both diversity gain and BER performance gain) without having to sacrifice the time/bandwidth efficiency or data rate for M-to-1 systems. To achieve diversity gain, BER performance gain and a full rate (i.e. rate 1) at the same time can be quite challenging. It is shown that effective diversity gain and coding gain from superposition in code domain [66] can only be achieved with a judiciously-designed code-book, which requires a higher complexity than superposition modulation [69]. We are particularly interested in efficient cooperative transmission schemes through signal superposition in the chapter.

1.2 Outline of the Dissertation

This dissertation is organized as follows:

In chapter 2, we study a mirrored baker's map code, which is a kind of analog code. By leveraging the cleverly designed mirrored baker's map code, we propose a new soft forwarding strategy, termed analog-encode-forward (AEF) strategy for the single source,

single relay and single destination relay system. Both ML and MAP decoding algorithm for AEF strategy are derived.

In chapter 3, we consider the relay system with one source, one destination and parallel relay nodes. What soft information should be forwarded is analyzed first. We thus propose a new "Z" forwarding strategy, which is simple to operate but delivers the best BER performance among different prevailing forwarding strategy. Parameter selection is discussed for the forwarding strategy, on which the overall performance depends. A exact ML estimation algorithm is also proposed.

In chapter 4, a new soft distributed coding scheme is proposed for parallel relay system. We still argue the range-limited LLR serves as the best soft input for the SISO encoder. Another advantage to encoder the range-limited LLR at relay nodes is the linearity of the soft information, so that we can derive a nearly optimal ML decoder at the destination node.

In chapter 5, a general class of superposition-based coding strategy is proposed to match to the network topology of a M-to-1 multi-user single-destination cooperative system. The proposed *braid code* is capable of simultaneously achieving diversity gain, coding gain and a full rate. Efficient ML decoders are derived for both source-uncoded and source-coded cases.

Chapter 2

Analog-Encode-Forward (AEF) Strategy for Single Relay Systems

2.1 Introduction and Motivation

The key idea of user cooperation in relay systems is to introduce one or multiple cooperating nodes, termed the *relay(s)* (R), which help(s) to retransmit or forward part of or all of the signals received from the *source(s)* (S) to the *destination(s)* (D). In this chapter, a soft forwarding strategy in a simple relay system with one source node, one destination node, and one relay node is investigated, shown in Fig. 2.1.

In retrospect, the notion of signal relaying first appeared in the early seventies [1], and fundamental results on possible relaying strategies and information-theoretic properties over Gaussian noise channels were developed in the late nineties [2]. However, it was not until the past decade that signal relaying received serious attention as a feasible

and potentially very fruitful strategy for modern-day communication systems. Considerable amount of research has since been launched to study and improve the reliability and efficiency of relay systems (e.g. [3]- [7]). Despite the tremendous advances in theoretical results and the proposition of various systems and network models, practical strategies have not evolved much far from *amplify-forward* (AF) and *decode-forward* (DF), the two basic modes that were developed in the seventies [2].

Amplify-forward and decode-forward both have advantages and disadvantages, which appear to be complementary. It is possible to improve these strategies by joining the merit of AF (i.e. soft-forwarding, operational at all times) with that of the DF (i.e. coding gain), and at the same time, fixing each other's short-coming. For example, a *decode-amplify-forward* (DAF) scheme proposed in [3] presents a simple but effective solution. In the DAF scheme, the relay first soft-decodes the packet (to obtain the coding gain); Should the CRC fail, the relay can amplify and forward the soft reliability information produced by the decoder, thereafter referred to as the *decoder-LLR*, to the destination. As such, the system has not only exploited the coding gain, but also achieved soft-forwarding as well as all-time-operation ability. Capacity analysis and simulations show that the DAF scheme is capable of very rewarding performance gains compared to either AF or DF, especially when the S-R channel is weak [3].

It is common sense that the worst case tends to dominate the performance of a communication system. When all the channels are good, any strategy will likely deliver plausible performance. The challenge is when some of or all of the channels are weak or happen to experience unlucky deep fades. Consider a case when the S-D channel is very poor or practically non-existent, i.e. the source and the destination are not directly reachable, and must therefore rely on the relay to forward the message (see. Fig. 2.1).

For AF and DF to work well, it is expected that the S-R channel be in a reasonable condition. The DAF scheme has improved the situation by delivering desirable performance in the case of weak S-R channel and decent R-D channel [3]. When the R-D channel is also poor, the DAF scheme does not render much help, as the soft-reliability message forwarded by the relay, which is not being protected, will likely become badly-corrupted and near-useless when it reaches the destination.

To solve this problem, consider extending the DAF scheme by launching effective protection on the R-D transmission also. This idea, although conceptually simple, appears rather difficult in practice. This is because the data thereof is soft, or, real-valued, whereas the conventional channel codes work only on discrete/digital data. To protect real-valued data with digital codes requires the data be first of all quantized; but quantization tends to significantly increases the data volume, and inevitably causes irrecoverable granularity error. The cost and overhead of quantizing the soft decoder-LLRs and then encoding and transmitting them, could easily overweigh the potential gain, making the entire process meaningless. A smarter approach is to directly protect and transmit decoder-LLRs (or other real-valued probability values) using some soft-input encoding scheme on the relay hop, and to decode and deduce the corresponding binary bits at the destination [4–6]. For example, [6] proposed a novel and concrete "soft-input" coding scheme that confirmed not only the feasibility but also the high benefits of transmitting protected soft information.

This dissertation considers a somewhat different strategy by leveraging the recent advances of analog coding technology. Introduced independently by Marshall and Wolf in the early eighties [8] [9], this coding concept, termed either *analog code* (by Wolf [9]) or *real value code* (by Marshall [8]), represents a generalization of digital channel

codes, by directly encoding real-/complex-valued source sequences to real-/complex-valued codewords. Despite its thirty-year-old concept, however, effective analog coding schemes that can serve practical purposes only appeared in the last few years.

In this part of the dissertation, we will present an efficient analog code based on chaotic functions. We will detail the underpinning principle, the simple encoding algorithm, and the maximum-likelihood (ML) decoding algorithm that produces a desirable mean square error (MSE) performance. It should be noted that there is a major difference between our coding strategy and the soft-input coding strategies proposed previously (e.g. [4, 6]). In the latter, the real-valued information at the input to the code must be some type of bit reliability information (e.g. probability, (log) likelihood ratio), both the encoding and decoding processes are specifically designed for this type of input, and the decoding process ultimately targets the underlying binary bits. In comparison, in our proposed coding scheme (and analog coding schemes in general), the input source can be general and arbitrary real or complex values, the encoding and decoding processes do not make any assumption on the meaning of the source, and the decoding process works its best to minimize MSE.

It is also worth noting that, unlike the early-day analog codes [8,9] that were a natural extension of linear digital codes in high-dimensional finite fields, our analog code here is nonlinear and is constructed by cleverly exploiting the *chaos theory* and the *turbo coding paradigm* [11]. We will detail (in Section 2.2) how the prominent feature of chaos – the butterfly effect – can be enacted to construct error correction codes, and how the renowned parallel concatenation structure of digital turbo codes can be leveraged to achieve good performance.

Next, we will demonstrate how our chaos-based analog code can be employed to

extend and improve decode-amplify-forward. The result is a new signal relaying scheme called *analog-encode-forward (AEF)*, in which the relay will first soft-decode the packet from the source, next "soft-encode" the decoder-LLR using an appropriate analog code, and finally forwards the "soft codeword" to the destination. We further present a hybrid AEF and DF scheme, where the relay performs AEF upon unsuccessful decoding, and switches to DF otherwise. We show that the new strategies, AEF and hybrid AEF-DF, can achieve impressive gains even when S-R and R-D channels are both weak.

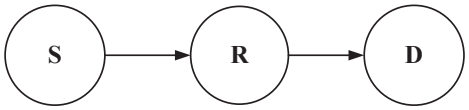


Figure 2.1: A 2-hop system, where the source must rely on the relay to deliver the message to the destination.

2.2 Efficient and Practical Analog Code

We first discuss the chaos-based analog code, before putting it in the context of signal relaying. Throughout this proposal, we will use bold fonts to represent vectors and matrices, and regular fonts to represent scalars. The notation $\mathbf{A}_m^n$ represent the vector $(A_m, A_{m+1}, \cdots, A_n)$.

The key idea of chaos-based error correction is to exploit the *butterfly effect* (i.e. *sensitivity to initial condition*) of a chaotic system to serve the *distance expansion* condition required by a good channel code. Since a small difference (distance) in the initial state will lead to huge differences (distance) later on, a chaotic function can act like an encoder, taking in the source symbols as the initial state, and producing a sequence of succeeding states as the codeword.

However, directly exploiting this idea, such as what is shown in [10], has not been effective. Part of the reason is that, because of the high-sensitivity of chaotic functions and hence the potential complication in estimating its initial state, simple chaotic functions are preferred, to ensure affordable decoding complexity. However, simple functions tend to offer relatively simple and weak relation between the time-evolving states.

Recall from the success of digital turbo codes that, in general, it is not only feasible but also highly beneficial to construct longer, stronger codes using shorter, weaker codes. Exploiting a parallel concatenation structure similar in flavor to that of the digital turbo code, we present a *mirrored baker's map code*.

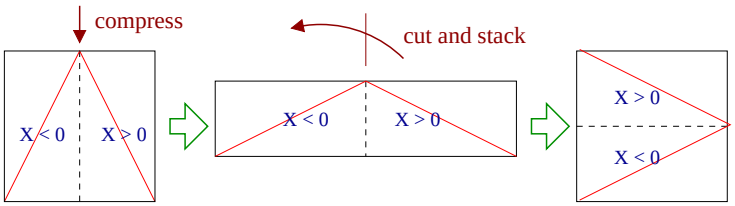


Figure 2.2: The baker's map.

2.2.1 Encoding of Mirrored Baker's Map Codes

The baker's map, a chaotic function that maps a unit square to itself, is named after a kneading process that bakers operate on dough (see fig. 2.2. Consider a rate $1/N$ chaotic code based on a single baker's map. A pair of source symbols $\{u, v\}$, namely, the systematic symbols, are taken in as the initial state $\{x[0], y[0]\} = \{u, v\}$, and $(N-1)$ succeeding states $\{x[1], y[1]\}, \cdots, \{x[N-1], y[N-1]\}$ are collected as the parity

symbols $F(\{x, y\})$:

$$
\begin{aligned}
&\{x[k], y[k]\} \\
=&F\big(\{x[k-1], y[k-1]\}\big) \qquad (2.1)\\
=&\begin{cases} \big\{2x[k-1]+1,\ \frac{y[k-1]}{2}-\frac{1}{2}\big\}, & \text{if } x[k-1]<0; \\ \big\{1-2x[k-1],\ \frac{1}{2}-\frac{y[k-1]}{2}\big\}, & \text{otherwise;} \end{cases} \qquad (2.2)
\end{aligned}
$$

where $-1 \leq x[0] \leq 1$ and $-1 \leq y[0] \leq 1$.

From (2.2), we see that $y[k]$ is only a function of both $x[k-1]$ and $y[k-1]$, but $x[k]$ is a function of only $x[k-1]$. What this implies is that a previous state of x is incident with and protected by all the future states of x and y, whereas a previous state of y is protected by the future states of y only, causing a weaker protection. A parallel concatenation though a mirrored replication, as depicted in Fig. 2.3, can quickly and effectively solve this issue. Specifically, one can feed u, v as the initial state to the first baker's map, collecting $(N-1)$ succeeding states thereof: $\{x_1[0] = u, y_1[0] = v\}, \{x_1[1], y_1[1]\}, \cdots, \{x_1[N-1], y_1[N-1]\}$, and feed the mirrored pair $\{v, u\}$ to the second baker's map, and collect another $(N-1)$ succeeding states $\{x_2[0] = v, y_2[0] = u\}, \{x_2[1], y_2[1]\}, \cdots, \{x_2[N-1], y_2[N-1]\}$. Mathematically, encoding is performed

in the following recursive manner:

$$\begin{cases} \{x_1[k], y_1[k]\} &= F(\{x_1[k-1], y_1[k-1]\}) \\ &= F^k(\{x_1[0], y_1[0]\}) \\ &= F^k(\{u, v\}) \\ \{x_2[k], y_2[k]\} &= F(\{x_2[k-1], y_2[k-1]\}) \\ &= F^k(\{x_2[0], y_2[0]\}) \\ &= F^k(\{v, u\}) \end{cases} \tag{2.3}$$

where F is the baker's map defined in (2.2).

If the systematic symbols u and v are transmitted only once, then we have a code rate $\frac{1}{2N-1}$; if they are transmitted twice (from both branches), then the code rate becomes $\frac{1}{2N}$.

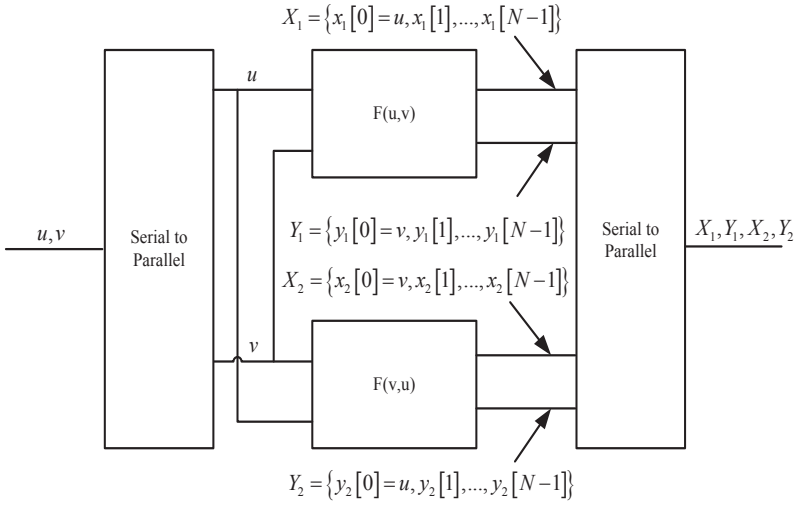


Figure 2.3: Mirrored baker's map codes.

2.2.2 Decoding of Mirrored Baker's Map Codes

Consider transmitting the codewords $\{\mathbf{X_1}_0^{N-1}, \mathbf{Y_1}_0^{N-1}\}$, $\{\mathbf{X_2}_0^{N-1}, \mathbf{Y_2}_0^{N-1}\}$ through a block fading channel with fading coefficients h_{x1}, h_{y1}, h_{x2} and h_{y2}. The receiver gets the noisy codeword $\{\mathbf{Rx_1}_0^{N-1}, \mathbf{Ry_1}_0^{N-1}\}$, $\{\mathbf{Rx_2}_0^{N-1}, \mathbf{Ry_2}_0^{N-1}\}$:

$$\mathbf{Rx_1}_0^{N-1} = h_{x1}\mathbf{X_1}_0^{N-1} + \mathbf{W_1}_0^{N-1}, \tag{2.4}$$

$$\mathbf{Ry_1}_0^{N-1} = h_{y1}\mathbf{Y_1}_0^{N-1} + \mathbf{W_2}_0^{N-1}, \tag{2.5}$$

$$\mathbf{Rx_2}_0^{N-1} = h_{x2}\mathbf{X_2}_0^{N-1} + \mathbf{W_3}_0^{N-1}, \tag{2.6}$$

$$\mathbf{Ry_2}_0^{N-1} = h_{y2}\mathbf{Y_2}_0^{N-1} + \mathbf{W_4}_0^{N-1}, \tag{2.7}$$

where h denotes the fading coefficient and $\mathbf{W}$ denoted the independent Gaussian noise. Consider the ML decoder:

$$\begin{aligned}
&\{\tilde{u}, \tilde{v}\} \\
&= \arg\max_{-1\le u,v\le 1} \Pr\left(\{\mathbf{Rx_1}, \mathbf{Ry_1}, \mathbf{Rx}_2, \mathbf{Ry}_2\}_0^{N-1} \middle| \{u, v\} \right), \qquad (2.8) \\
&= \arg\min_{-1\le u,v\le 1} \sum_{i=0}^{N-1} \Big\{ (Rx_1[i] - h_{x1}x_1[i])^2 + (Ry_1[i] - h_{y1}y_1[i])^2 \\
&\qquad + (Rx_2[i] - h_{x2}x_2[i])^2 + (Ry_2[i] - h_{y2}y_2[i])^2 \Big\}. \qquad (2.9)
\end{aligned}$$

Recall that the baker's map F is a nonlinear but piece-wise linear function, and so is its multi-fold recursion F^i. Since parallel concatenation preserves linearity, the entire mirrored baker's map code is essentially a piece-wise linear function. It is then natural to divide the entire support region to several sections, each of which presents $x_1[i], y_1[i], x_2[i], y_2[i]$ as a linear function of u and v. For a baker's map of the $(N-1)$th

order, it is sufficient to divide the region into 2^{N-1} linear and differentiable sections, such that in each section, the ML decoder formulated in (2.9) reduces to minimizing a bi-variable quadratic function, which can be efficiently solved by taking partial derivatives on u and v, respectively. Each section is uniquely specified by the signs of the x sequences, $\mathbf{S}_{\mathbf{1}_0}^{N-2}$ and $\mathbf{S}_{\mathbf{2}_0}^{N-2}$, where $s_1[i] = \text{sign}(x_1[i])$ and $s_2[i] = \text{sign}(x_2[i])$.

The overall decoding process can proceed in the following simple recursive manner.

Step 1: Uniformly divide the entire 2-dimensional square of $[-1, 1] \times [-1, 1]$ into a total of 4^{N-1} sub-squares, each uniquely specified by a sign sequence of length $2(N-1)$, $\{\mathbf{S}_{\mathbf{1}_0}^{N-2}, \mathbf{S}_{\mathbf{2}_0}^{N-2}\}$.

Step 2: In each linear section, recursively compute the following parameters:

$$\text{branch 1} \begin{cases} a_1[n] = -2a_1[n-1]s_1[n-1] \\ b_1[n] = 1 - 2s_1[n-1]b_1[n-1] \\ a_2[n] = -0.5a_2[n-1]s_1[n-1] \\ b_2[n] = 0.5s_1[n-1] - 0.5b_2[n-1]s_1[n-1] \end{cases} \tag{2.10}$$

$$\text{branch 2} \begin{cases} a_3[n] = -2a_3[n-1]s_2[n-1] \\ b_3[n] = 1 - 2s_2[n-1]b_3[n-1] \\ a_4[n] = -0.5a_4[n-1]s_2[n-1] \\ b_4[n] = 0.5s_2[n-1] - 0.5b_4[n-1]s_1[n-1] \end{cases} . \tag{2.11}$$

such that the coded symbols relate to the source, u and v, in the following simple, linear

forms:

$$\text{branch 1} \begin{cases} x_1[n] = a_1[n]u + b_1[n], \\ y_1[n] = a_2[n]v + b_2[n], \end{cases} \tag{2.12}$$

$$\text{branch 2} \begin{cases} x_2[n] = a_3[n]v + b_3[n], \\ y_2[n] = a_4[n]u + b_4[n], \end{cases} \tag{2.13}$$

where $n = 0, 1, 2, \cdots, N-1$ is the time index.

Step 3: Insert (2.12) and (2.13) in (2.9), and take partial derivatives on u and v to obtain the "locally-optimal" solution:

$$\tilde{u} = \underset{\mathbf{s_{10}}^{N-1}, \mathbf{s_{20}}^{N-1}}{\arg\min} \frac{T_1}{\sum_{i=0}^{N-1} \left(a_1^{*2}[i] + a_4^{*2}[i] \right)}, \tag{2.14}$$

$$\tilde{v} = \underset{\mathbf{s_{10}}^{N-1}, \mathbf{s_{20}}^{N-1}}{\arg\min} \frac{T_2}{\sum_{i=0}^{N-1} \left(a_2^{*2}[i] + a_3^{*2}[i] \right)}, \tag{2.15}$$

where

$$T_1 = \sum_{i=0}^{N-1} \Big\{ Rx_1[i]a_1^*[i] - a_1^*[i]b_1^*[i] + Ry_2[i]a_4^*[i] - a_4^*[i]b_4^*[i] \Big\},$$
$$T_2 = \sum_{i=0}^{N-1} \Big\{ Ry_1[i]a_2^*[i] - a_2^*[i]b_2^*[i] + Rx_2[i]a_3^*[i] - a_3^*[i]b_3^*[i] \Big\}.$$

where $a_1^*[i] = a_1[i]h_{x1}$, $b_1^*[i] = b_1[i]h_{x1}$, $a_2^*[i] = a_2[i]h_{y1}$, $b_2^*[i] = b_2[i]h_{y1}$, $a_3^*[i] = a_3[i]h_{x2}$, $b_3^*[i] = b_3[i]h_{x2}$, $a_4^*[i] = a_4[i]h_{y2}$, $b_4^*[i] = b_4[i]h_{y2}$:

Since the locally-optimal solutions may actually fall outside the designated region, they should also be compared to the respective region boundaries. The ones that fall out

should be replaced by the appropriate boundary point.

Step 4: All the 4^{N-1} local solutions are compared, and the one that minimizes the ML target function in (2.9) becomes the ultimate ML solution.

For a $(4N, 2)$ or $(4N-2, 2)$ mirrored baker's map code, the complexity of the decoder increases exponentially with N. In practice, considering the code rate $\frac{1}{2N}$ or $\frac{1}{2N-1}$, N is generally chosen to be relatively small, such as 2 and 3; so the complexity is rather reasonable.

2.3 System Model for User Cooperation

We now discuss our cooperative schemes and demonstrate how analog codes may be employed to enhance soft-forwarding.

2.3.1 Relay System Model

We consider a simple 3-node 2-hop system depicted in Fig. 2.1, where the source communicates to the destination via the help of the relay. The communication process is straightforward: in the first phase, the source node transmits a packet of data to the relay; in the second phase, the relay node processes and forwards the observations to the destination, which makes a best estimation of the original message. Subscripts $_S$, $_D$, $_{SR}$ and $_{RD}$ are used to denote the quantities associated with the source S, the relay R, the S-R channel and the R-D channel, respectively.

$$y_{SR}(i) = \alpha_{SR}(i)x_S(i) + w_{SR}(i) \tag{2.16}$$

$$y_{RD}(i) = \alpha_{RD}(i)x_R(i) + w_{RD}(i) \tag{2.17}$$

where x is the transmitted signal, y is the received signal and α is the channel state information (CSI). In the case of AWGN, α is a constant of 1. The addictive white Gaussian noises, w_{SR} and w_{RD}, have zero-mean and variances σ^2_{SR} and σ^2_{RD}, respectively. $x_S \in \{-1, +1\}$ is always modulated using binary phrase shift keying (BPSK), and x_R depends on the respective forward strategy.

We assume that the S-R and R-D channels are spatially independent, and that the instantaneous CSI is known to the receiver, but not the transmitter. Let the signals being transmitted have unit energy on average. The signal-to-noise ratio (SNR) of each channel is defined as $\gamma \stackrel{\Delta}{=} \frac{1}{N_0} = \frac{1}{2\sigma^2}$, where σ^2 is the corresponding noise variance.

2.3.2 Traditional Relaying Schemes

We first discuss three existing relay strategies: amplify-forward, decode-forward, and decode-amplify-forward [3], before proceeding to the new analog-encode-forward strategy.

• Amplify-Forward (AF)

A simple cooperative strategy, the amplify-forward scheme lets the relay scale (amplify) the S-R (real-valued) reception, and puts it on the R-D channel. Let i be the time index, and $\bar{P}_{SR} = \mathbf{E}[\,|y_{SR}|^2\,]$ be the average power for the S-R reception. Here we consider the case that the channel code rate at the source node is 1/2. The signal forwarded

at the relay node takes the form of

$$x_R(i) = \frac{y_{SR}(i)}{\sqrt{\bar{P}_{SR}}}, \;\; i = 1, 2, ..., 2N \tag{2.18}$$

where $2N$ is the packet size, N is the length of the original message bits at the source node.

$$\bar{P}_{SR} = \frac{1}{2N} \sum_{i=1}^{2N} |(y_{SR}(i))^2| \overset{N\to\infty}{\longrightarrow} \sigma_{SR}^2 + \alpha_{SR}^2 \tag{2.19}$$

The destination observes $y_{RD}(i)$ from the R-D channel:

$$y_{RD}(i) = \frac{\alpha_{SR}(i)\alpha_{RD}(i)}{\sqrt{\bar{P}_{SR}}} x_S(i) + \left(\frac{\alpha_{RD}(i) n_{SR}(i)}{\sqrt{\bar{P}_{SR}}} + n_{RD}(i) \right) \tag{2.20}$$

where the last two terms denote the combined noise, which is not necessarily Gaussian. In the case of quasi-static fading (block fading), where the fading coefficient α_{RD} remains a constant for the entire block, then combined noise (in this block) follows a Gaussian distribution with variance $(\alpha_{RD}^2 \sigma_{SR}^2)/(\alpha_{SR}^2 + \sigma_{SR}^2) + \sigma_{RD}^2$.

The destination can compute the log-likelihood ratios from the channel (assuming all the CSI's are known):

$$L_{AF}(i) = \frac{2\sqrt{\bar{P}_{SR}}\alpha_{SR}(i)\alpha_{RD}(i)}{\alpha_{RD}^2(i)\sigma_{SR}^2 + \bar{P}_{SR}\sigma_{RD}^2} y_{RD}(i) \tag{2.21}$$

• Decode-Forward (DF)

In a DF scheme, the relay node attempts to decode the source bits from the S-R reception, and retransmit the detected bits (and usually re-encode them using a same or different channel code before putting them through the R-D channel). The destination

can extract the following channel-LLRs:

$$L_{DF}(i) = \frac{2\alpha_{RD}}{\sigma_{RD}^2} y_{RD}(i) \tag{2.22}$$

Since the DF scheme cleans up the noise and regenerates the signals at the relay, the destination can usually expect better reception than AF. The downside, however, is when the relay node fails to decode the S-R reception successfully, in which case signal relaying must be aborted in order to avoid disastrous error propagation. It has been shown in [12] that such S-R "outage" cases may actually occur on the order of a couple of percent (over slow-varying fading channels), which means that it is not as rare an event as can be safely ignored.

• **Decode-Amplify-Forward (DAF)**

The decode-amplify-forward scheme attempts to combine the merit of both AF and DF: soft-forwarding and all-time-operatability of AF, and S-R coding gain of DF [3]. In DAF, the relay node first soft-decodes the S-R codeword to extract the decoder-LLRs of the source bits, and then scales (amplifies) these decoder-LLRs (rather than direct S-R channel reception) and forwards them to the destination.

By exploiting the channel code in the S-R transmission, the DAF scheme is like the AF scheme operating on an "improved" S-R channel. Analysis and simulations show that it promises considerable capacity enhancement as well as practical gains [3]. Aiming at "improving" the S-R channel also, we propose to incorporate (analog) code on the second hop, which gives rise to the proposed AEF scheme.

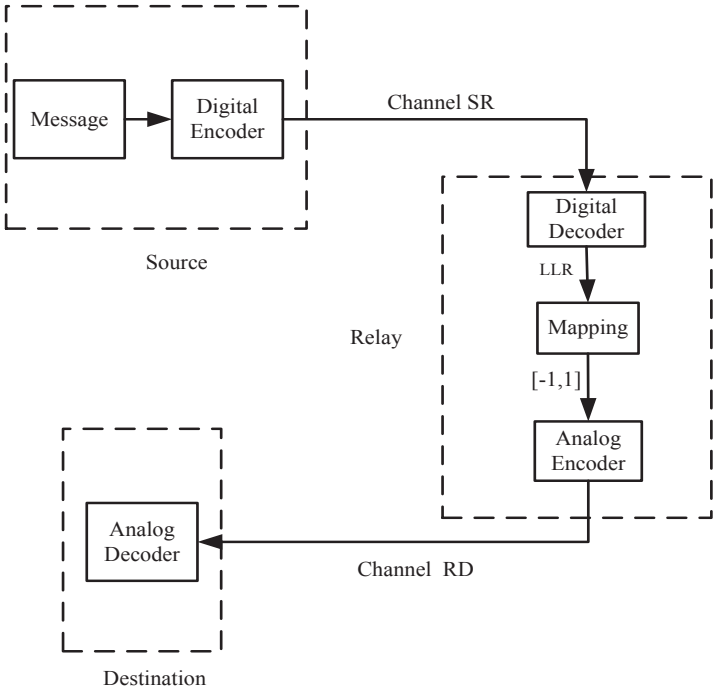


Figure 2.4: The proposed analog-encode-forward (AEF) scheme.

2.4 AEF and AEF-DF Scheme

2.4.1 Analog-Encode-Forward (AEF) with ML Analog Decoder

The proposed analog-encode-forward scheme is a generalization of the DAF scheme by allowing analog coding on the R-D channel. Fig. 2.4 illustrates the integrated system diagram for AEF. The source node encodes the message bits using a digital code, same as all the other relaying strategies. The relay takes the following action on the S-R reception:

First, it decodes the S-R codeword, and extracts the real-valued decoder-LLRs of the message bits.

Next, it prepares the soft message in a form suitable for encoding by the subject analog code. Since our mirrored baker's map code takes in real-valued sources in the range of $[-1, 1]$, instead of truncating and scaling decoder-LLRs (which ranges from $-\infty$ to $+\infty$), we let the relay compute the probabilities (which ranges from 0 to 1), and scales and shifts it to $[-1, 1]$. Recall that the LLR is defined as

$$L_{SR}(i) = \ln \frac{P(x_s(i) = +1)}{P(x_s(i) = -1)} = \ln \frac{P\left(\tilde{x}_s\left(i\right) = +1\right)}{1 - P\left(\tilde{x}_s\left(i\right) = +1\right)}; \tag{2.23}$$

so the probability of the bit being $+1$ can be computed as:

$$P(x_s(i) = +1) = \frac{e^{L_{SR}(i)}}{1 + e^{L_{SR}(i)}} \tag{2.24}$$

Finally, the relay encodes the real-valued messages using the analog code, and transmits the resultant real-valued codeword. For a fair comparison with all the other schemes, the average power of the analog codeword should be normalized to unity.

The destination, upon receiving R-D analog codeword, will perform analog decoding to extract the analog source, which denotes the probability of message bits being $+1$. Recall this probability was actually the probability after S-R decoding, so a simple comparison of it to the threshold (0.5) readily generates the hard-decision of the original message bit.

2.4.2 Hybrid AEF-DF

It is apparent that clean signal regeneration at the relay, if possible, results the best R-D reception. Hence, a time-sharing hybrid AEF-DF scheme can be arranged, in the

same way as the hybrid DAF-DF scheme discussed in [3]: When the S-R reception is reasonably good such that the decoder can correctly extract all the message bits (as verified by CRC), the relay should proceed with a decode-forward scheme (by re-encoding the binary message bits using an appropriate digital code). Otherwise, the relay resumes AEF, by re-encoding the soft decoder outputs using the analog code and forwarding the real-valued analog codeword.

2.4.3 Simulation Results

We conduct simulations to compare the performance of five schemes: AF, DF, DAF [3], and the proposed AEF and hybrid AEF-DF scheme. Both AWGN and block fading channels are considered. In the S-R transmission, a $(2000, 1000)$ recursive systematic convolutional (RSC) code with generator polynomial $(1, 23/35)_{oct}$ is used as the channel code. The BCJR algorithm is used for the schemes that involve decoding at the relay. The R-D transmission also involves 2000 symbols,w where AF and DAF forward the channel-LLR's and decoder-LLR's of the S-R systematic and parity bits, DF always re-encodes the decoded bits using the same $(2000, 1000)$ RSC, and AEF encodes the decoder-LLR's of the S-R systematic bits using a rate 1/2 mirrored baker's map code.

We fix the S-R channel SNR, and evaluate the performance as a function of the R-D SNR. Fig. 2.5 presents the AWGN case, at S-R SNR of 2dB. We see that AEF consistently outperform AF at low and high R-D SNRs. The AEF scheme also performs better than DAF and DF at low R-D SNRs, but falls short at high R-D SNRs. As expected, the hybrid AEF-DF scheme performs the best. Fig. 2.10 represents the block fading channel case, at S-R SNR of 30dB. Here, AEF appears to be slightly outperform AF,

DF and ADF on high SNRs. We note all the schemes appear to hit the same error floor, which was limited by the S-R channel quality.

Comments: As a stand-alone analog code, our mirrored baker's map code can noticeably outperform uncoded systems in transmitting uniformly-distributed analog sources. The fact that the new AEF scheme shows only a small advantage and only at low R-D SNRs may be attributed to the following factors: (i) The analog message at the relay (i.e. the probabilities of bits being $+1$) may not be uniformly distributed, but the decoder treats them as uniform. (ii) Probabilities may not be the best message type for soft-forwarding, because they have different levels of sensibility and importance in different value regions. For example, consider the probability of a particular bit being $+1$ that is distorted from 0.8 to 0.9, versus a same-scale distortion from 0.45 to 0.55. Both result in the same squared error, but the latter, being around the threshold value, lead to a harmful sign change and hence a bit error. (iii) gains in MSE do not necessarily correspond in similar scales to gains in BER. These issues wait to be investigated, and we believe there is (considerable) room for AEF to improve.

2.5 A New Analog-Encode-Forward (AEF) Strategy and MAP Analog Decoder

The analog-encode-forward strategy described above generalizes the decode-amplify-forward scheme [3] by employing analog coding in the $\mathtt{R}-\mathtt{D}$ transmission. Upon receiving the S-R codeword, the relay decodes and extracts the real-valued decoder-LLRs of the message bits, prepares them in a form suitable for encoding by the analog code, and

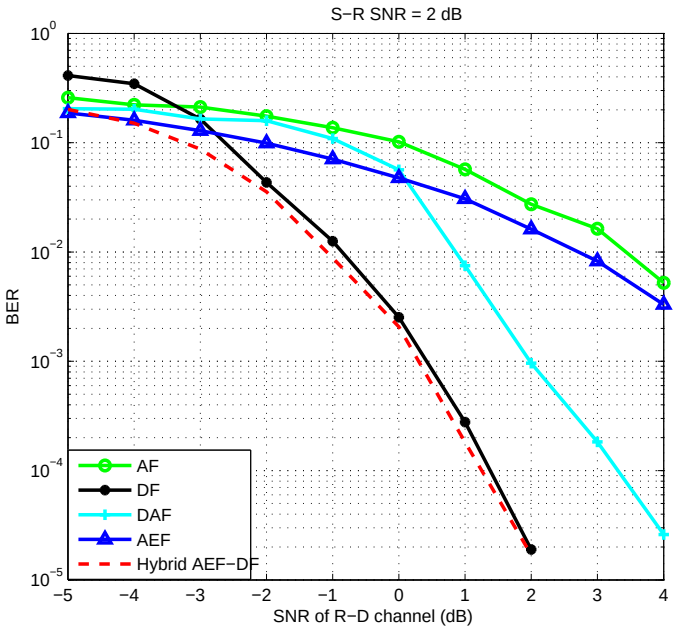


Figure 2.5: BER of different schemes at S-R SNR of 2dB

forwards the analog-coded soft messages to the destination. Since the proposed analog code takes in values in $[-1, 1]$, it appears natural to convert the decoder-LLRs (which is unbounded) to the probability values (which is bounded) and shifts and scales them to $[-1, 1]$. The destination, after receiving R-D analog codewords, uses, for example, a general-purpose ML decoder to retrieve the decoder probabilities. Then, by comparing them to the threshold of 0.5, hard-decisions of the source bits are obtained.

2.5.1 AEF with MAP Analog Decoder

A new analog-encode-forward strategy is proposed in this section to cleverly solve the problems by: a) exploiting a different mapping method at the relay, which encodes and forwards the (truncated) soft decoder-LLRs, rather than directly employing the probabilities of the bits; and b) designing a binary-output MAP decoder at the destina-

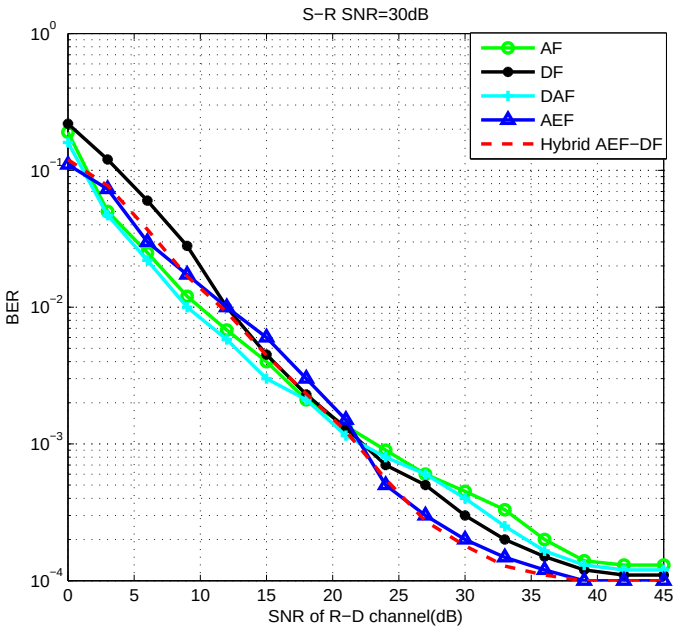


Figure 2.6: BER of different schemes at S-R SNR of 3dB

tion. We note that although the data the relay node forwards is soft (i.e. decoder LLRs) and spans the entire support region of the mirrored baker's map code (i.e. $[-1, 1]$), ultimately we are only concerned about the data originated at the source node, which has only two binary values. Hence, the $\mathtt{R} - \mathtt{D}$ messages should not be treated as having a uniform *a priori* distribution over $[-1, 1]$ (as is what the general-purpose ML decoder does); rather, there are only two points within $[-1, 1]$ that have non-zero *a priori* probabilities, and they correspond to the original bit (at the source node) being +1 or -1. Exploiting this information leads to an extremely simple but effective MAP decoder.

Fig. 2.7 illustrates the integrated system structure for the new AEF. The source encodes the message bits using a digital channel code, similar to all the other strategies. Next, we have:

Step 1: Upon receiving the signals from the source, the relay decodes the S-R code-

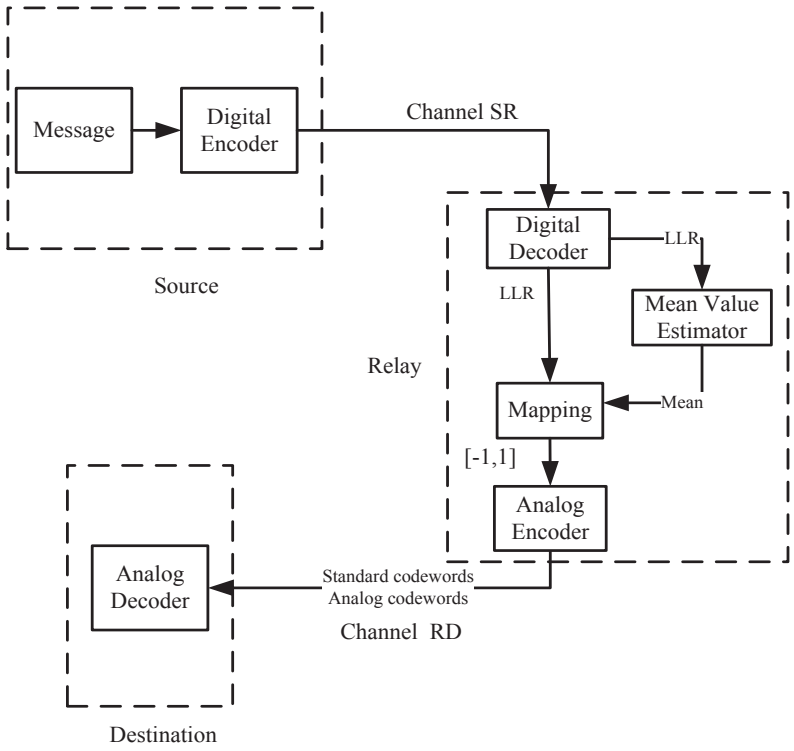


Figure 2.7: Analog-Encode-Forward with a MAP decoder.

word, and extracts the real-valued decoder-LLRs l_{SR}^{DEC} of the message bits.

Step 2: The relay prepares the soft message to be analogly-encoded and forwarded to the destination. Note the the soft message should be limited to $[-1, 1]$ in order to feed to the mirrored baker's map code. The Gaussian approximation states that decoder-LLRs follow an approximated Gaussian distribution on AWGN and block fading channels [13] [14]. Let $m > 0$ be the mean of the decoder-LLRs when the information bit is $+1$, and $-m$ be the mean of the decoder-LLRs when the information bit is -1. Theoretically, the precise mean value of the decoder-LLRs satisfies [14]

$$m = \frac{\sum_{i=1}^{N} l_{SR}^{DEC}(i)\, x_s(i)}{N}, i = 1, 2, ..., N \tag{2.25}$$

Since $x_s(i)$ is unknown to the relay, it is not possible to compute the true mean, we propose to approximate it using the "mean magnitude value":

$$m \approx \frac{\sum_{i=1}^{N} \left| l_{SR}^{DEC}(i) \right|}{N}, i = 1, 2, ..., N \tag{2.26}$$

where $|x|$ denotes the absolute value of x. The unbiased standard derivation can be subsequently obtained:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N} \left(|l_{SR}^{DEC}(i)| - m \right)^2}{N-1}}. \tag{2.27}$$

In light of the fact that about 95% probability mass of a Gaussian distribution lies within the range of $[m - 2\sigma, m + 2\sigma]$, we clip decoder-LLR values to this range,

$$\begin{aligned} &\text{if } (l_{SR}^{DEC}(i) > 0) \text{ and } (l_{SR}^{DEC}(i) > m + 2\sigma), \\ &\qquad l_{SR}^{DEC}(i) = m + 2\sigma, \end{aligned} \tag{2.28}$$

$$\begin{aligned} &\text{if } (l_{SR}^{DEC}(i) < 0) \text{ and } (l_{SR}^{DEC}(i) < -m - 2\sigma), \\ &\qquad l_{SR}^{DEC}(i) = -m - 2\sigma, \end{aligned} \tag{2.29}$$

and then linearly scale them to $[-1, 1]$, the input region for our mirrored baker's map code:

$$l_{analog-input} = \frac{l_{SR}^{DEC}(i)}{m + 2\sigma}, i = 1, 2, ..., N \tag{2.30}$$

Step 3: The clipped and scaled decoder-LLRs have mean values $m' = \frac{m}{m+2\sigma}$ and $-m'$ will be encoded using an $(4N, 2)$ mirrored baker's map code and transmitted to the destination. In our simulations, we consider an (4,2) code, where every 2 symbols

at the input produce 4 symbols at the output. We emphasize that the input sequences, being scaled decoder-LLRs, are like noisy data – although they take values across the entire region of $[-1, 1]^2$, they should be regarded as the noisy versions of only four possible input sequences $\{m', m'\}$, $\{m', -m'\}$, $\{-m', m'\}$, and $\{-m', -m'\}$. Let $\mathbf{A}_0^{4N-1}$, $\mathbf{B}_0^{4N-1}$, $\mathbf{C}_0^{4N-1}$, and $\mathbf{D}_0^{4N-1}$ be the four corresponding codewords ($N = 1$), which are forwarded by the relay to the destination.

Step 4: The destination employs a new MAP decoder that accounts for the fact there are essentially only four valid input sequences (with equal probability). This is drastically different from the conventional ML decoder presented in Subsection III C, which considers a continuum of input $[-1, 1]^2$ with uniform distribution. Since there are only four possibilities, it is straightforward and very efficient to perform an exhaustive search, to locate the one that has the minimum Euclidean distance from the $\mathtt{R} - \mathtt{D}$ reception.

Let $\mathbf{y}_{RD}$ be the codeword the destination received from the relay. Mathematically, the MAP decoder computes the four Euclidean distance:

$$\Delta_A = ||\mathbf{A}_0^{4N-1} - \mathbf{y}_{RD}||, \ \Delta_B = ||\mathbf{B}_0^{4N-1} - \mathbf{y}_{RD}||, \tag{2.31}$$

$$\Delta_C = ||\mathbf{C}_0^{4N-1} - \mathbf{y}_{RD}||, \ \Delta_D = ||\mathbf{D}_0^{4N-1} - \mathbf{y}_{RD}||. \tag{2.32}$$

Then, it decides on $\{m', m'\}$ if Δ_A is the smallest of all, on $\{m', -m'\}$ if Δ_B is the smallest, on $\{-m', m'\}$ if Δ_C is the smallest, and on $\{-m', -m'\}$ if Δ_D is the smallest. Since m' and $-m'$ are decoder-LLRs, m' corresponds to the data bit $+1$ at the source $\mathtt{R}$ and $-m'$ corresponds to the data bit -1. Since the new MAP decoder cleverly exploits the nature of the relay application, it outwins the conventional ML decoder not only in complexity but also in performance.

2.5.2 Simulation Results

Simulation results are presented to compare the performance of five schemes: (i) AF, (ii) DF, (iii) DAF through soft repetition code [3], (iv) AEF with an ML decoder, and (v) the proposed AEF with the new MAP decoder. Both $\mathtt{S}-\mathtt{R}$ and $\mathtt{R}-\mathtt{D}$ channels are assumed to have block fading. In the S-R transmission, a $(1000, 500)$ recursive systematic convolutional (RSC) code with generator polynomial $(1, 35/23)_{oct}$ is used together with binary phase shift keying (BPSK) at the source. The relay uses the BCJR algorithm to decode the RSC and obtain decoder-LLRs. The R-D transmission also consists of 2000 symbols: (i) in AF, the relay forwards the channel-LLRs of the S-R systematic and parity bits; (ii) in DF, the relay makes hard decisions on the decoded bits, and re-encodes them using the same $(1000, 500)$ RSC; (iii) in soft repetition DAF, the relay repeats the decoder-LLRs of the systematic bits twice, and the destination combines the two decoder-LLRs corresponding to the same bit, and compares the value with the threshold 0 to decide on bit +1 or -1; (iv)(v) in the proposed AEF, the relay clips and scales the decoder-LLRs, and encodes them using a (4,2) rate-1/2 mirrored baker's map code, and the destination may decode it using either ML or MAP decoding.

Fig. 2.8 and Fig. 2.9 shows the probability density function (PDF) of the decoder-LLRs at the relay with $\mathtt{S}-\mathtt{R}$ SNR equaling 28 dB, α=0.54 and α=0.4, respectively. We see that the analytical Gaussian curve does not match the histogram exactly, but is nevertheless a reasonable approximation.

Fig. 2.10 presents the block error rates (BLER) of the five schemes over block fading channels. S-R SNR is fixed to 28 dB. As expected, our new AEF strategy with the MAP decoder performs the best among all. DAF with soft repetition and hard-decision (MAP)

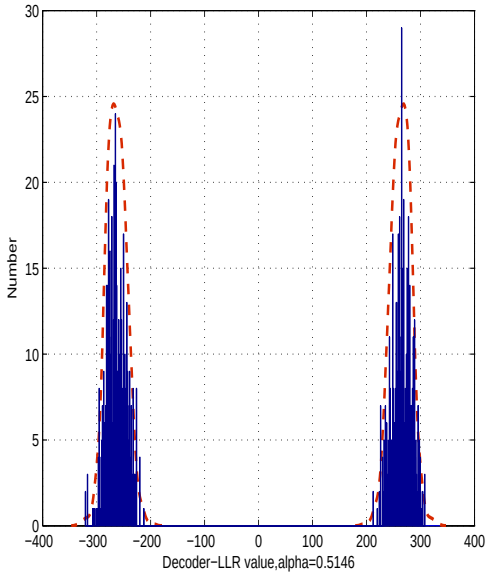


Figure 2.8: Observed decoder-LLRs at the relay and its Gaussian approximation,α=0.54

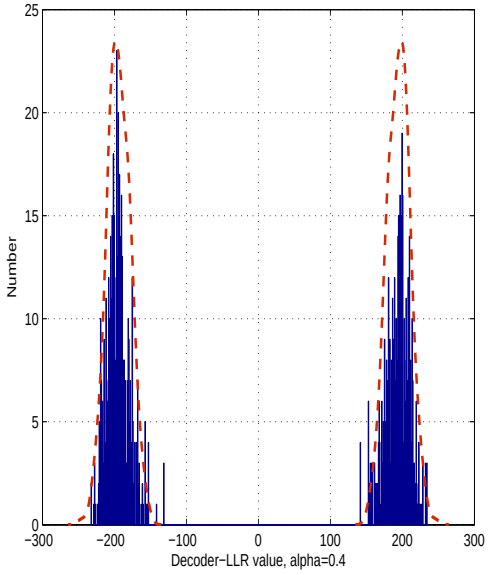


Figure 2.9: Observed decoder-LLRs at the relay and its Gaussian approximation,α=0.4

performs the second best. All the schemes hit an error floor of around block error rate of 4×10^{-4}, an artifact due to the limitation of the S-R channel quality.

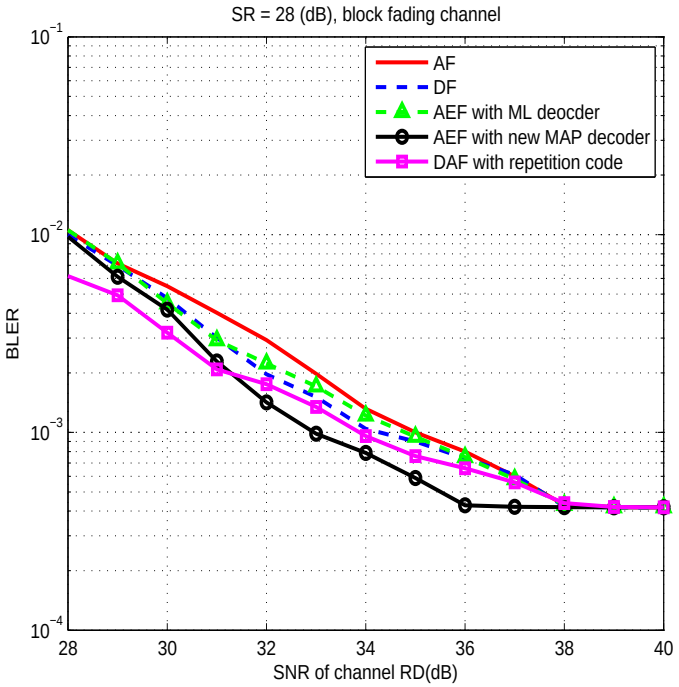


Figure 2.10: BLER comparison of different schemes at S-R SNR of 28dB on block rayleigh fading channels

Chapter 3

Z-Forward Strategy for Parallel Relay Systems

3.1 Introduction and Motivation

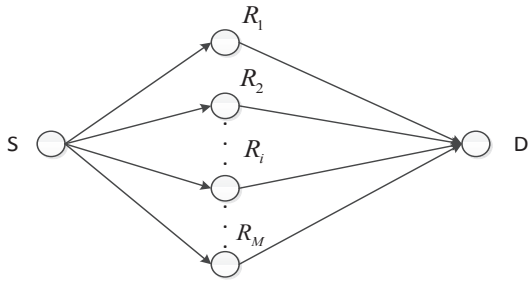


Figure 3.1: System model

Consider a wireless relay network shown in Fig. 3.1, where a single source S communicates to a single destination D via the help of a set of parallel relays R_i, $i = 1, 2, \cdots, M$. By judiciously employing the supportive relays, a higher end-to-end data rate can be achieved from pathloss gains and/or diversity gains. The advantage can

also be translated into a lower transmission power, or a better communication coverage [15] [16].

This chapter is primarily interested in developing an optimal way to achieve signal forwarding in a 2-hop relay network. We focus on non-bandwidth-reduction forwarding (i.e. not compress-forward). The questions that confront us include: What type of information should be forwarded, what function best captures this information, and how should the relay(s) and destination operate. Assessing several possible message representations, we identified LLR [3] as a very desirable form to represent the soft messages, because LLRs represent the reliability of the received signals, are very simple to calculate and conveniently addable, and take a Gaussian distribution (for additive white Gaussian noise (AWGN) channels). We discovered that by using range-limited LLR, we could reap most the benefits out of LLRs, without having to deal with infinite or excessively large values (which may cause numerical overflow issues). We further show in this paper that range-limited LLR is not only simple and effective, but also analytically tractable, thus enabling us to find the optimal thresholds that promise the smallest BER. It is worth noting that, for single-relay systems, an efficient $\tanh$-forward strategy/EF is proposed [19], where the relay forwards $\tanh(LLR(x)/2)$, the hyperbolic function of half the LLR of the reception at the relay. The optimality of $\tanh$-forward in a single-relay setting, in terms of maximizing the SNR at the destination, is also established [19]. However, it fails to achieve the full diversity order [31] and, hence, in a multi-relay setting, it will eventually fall behind AF and DF (at sufficiently high SNRs). Another interesting study [31] proposed to simplify the nonlinearity of $\tanh$-forward via a pre-determined, three-segment, *piecewise approximation*. The resulting *piecewise-forward (PF)* scheme was shown to be not only simple, but also capable of reaching a full diversity order; however, the method, being fixed and not specifically

tuned to multiple relays, did not achieve performance optimality. As one would expect, a truly efficient forward strategy must account for the collaborative effect of the multiple parallel S-R$_i$-D channels to minimize the overall bit error rate (BER), whereas $\tanh$-forward has clearly no relationship with other S-R$_i$-D segments.

Towards an end-to-end optimality, this chapter proposes a new, practical, soft forwarding strategy termed "Z-forward[1]", where the relays forward a θ-truncated version of the LLRs, where θ is a non-negative LLR threshold that needs to be optimized. If $\theta = 0$, the relay will forward the hard decision (sign of the received signal). Otherwise, the LLR of the received signal will be truncated to θ ($-\theta$), if its value is greater than θ (smaller than $-\theta$). The key in the design is the choice of the threshold θ_i for relay R$_i$. We show that Z-forward subsumes AF, DF and PF as its special case, and that a judicious selection of θ_i's can achieve an overall minimal BER for multi-relay systems (as well as for single-relay systems). Specifically, our contributions include:

1. We evaluate what soft messages the relay(s) should forward, and propose Z-forward as a class of efficient signal forwarding strategy for 2-hop relay systems. The forwarded message takes the form of a 3-segment piece-wise linear function of the received signal, and is rather simple to compute. We show that it is possible to determine the optimal thresholds θ_i's ($i = 1, 2, \cdots, M$) based on the specific channel conditions, to deliver the smallest overall BER for an arbitrary number of relays (arbitrary $M \geq 1$).

2. For Z-forward systems with a single or multiple active relays, we develop both the maximum ratio combining (MRC) decoding and the maximum-likelihood (ML)

[1]This is so named, because the transform function performed by the relay is a piece-wise linear function taking zig-zag form.

decoding for the destination. We derive the exact pdf of the received signals at the destination, formulate the overall BER as a function of the thresholds θ_i (and the channel conditions) based on MRC decoding, and compute the best values for θ_i to minimize the BER. We show that in the single-relay case, Z-forward delivers practically almost the same BER performance as the previously-proposed $\tanh$-forward, and both are optimal; but in multiple-relay case, Z-forward clearly outperforms $\tanh$-forward, PF, AF/DF adaptively switching schemes, and relay-selection AF, DF, EF schemes. Further, since the proposed Z-forward scheme is always better than AF and DF, and AF and DF are shown to attain the full diversity order [20] [32], Z-forward is therefore guaranteed of a full diversity also.

3. To ease the computation of θ_i, we propose to simplify the original Z-forward scheme, by adopting a single θ for all the relays instead of different ones for different relays. We provide a rather simple rule-of-thumb formulation for θ, which may or may not involve a search for correction term (in a small confined region). Extensive simulations demonstrate that the simplified strategy can still outperform the previous schemes including AF, DF, $\tanh$-forward, and PF.

3.2 System Model

The 2-hop relay system model is shown in Fig. 3.1. It consists of a source S, a destination D, and M parallel relay nodes R_i, $i = 1, 2...M$, with no direct S-D link. (Relay selection is a worthy topic of its own, but it not the subject of this paper; hence all the relays are assumed to be active at all times.)

Following the convention, we consider binary phase shift keying[2] (BPSK) with coherent detection. We assume all the links are block Rayleigh fading channels, as this is the case where time diversity is hard to achieve and hence user cooperation is most useful. We further assume that accurate channel state information (CSI) is available for all the links as [25] [31], such that the optimal thresholds for the proposed forward scheme can be computed based on the complete (instantaneous) channel condition. This may be expensive to achieve in practice, but it sheds useful insight into what optimal strategies are like, as well as provides an error rate lower bound of what can be achieved.

For ease of the discussion, we will focus on uncoded systems (with some mentioning of the coded system). From a performance perspective, a channel code acts much like a "channel booster" which helps boost the "effective" channel quality. In other words, an AWGN communication channel with a channel coding may be modeled, to the eye of the respective receiver, as a "virtual channel with an improved channel SNR". The proposed strategy applies to the coded systems also.

The proposed Z-forward strategy follows the same 2-phase operation procedure as the conventional AF and DF strategies. In the first phase, the source S broadcasts the signal x_S to all the relays R_i. Let E_S denote the energy per bit used by the source. The received signal at relay R_i is given by

$$y_{SR_i} = \sqrt{E_S} h_{SR_i} x_S + n_{SR_i}, \tag{3.1}$$

where h_{SR_i} is the respective Rayleigh CSI with mean zero and unit variance, $n_{SR_i} \sim$

[2]We used BPSK, primarily because it has been a tradition for a line of coding/modulation related study to first start with this basic scheme, so as to help reveal the fundamental essence, and to pave the way for subsequent extension to more sophisticated schemes. Please also note that, in general, the results of BPSK scheme can be directly applied to QPSK or 4QAM.

$\mathcal{N}(0, \sigma^2_{SR_i})$ is the AWGN at the relay R_i. The equivalent instantaneous SNR for the channel S-R_i during each frame is defined as SNR_{SR_i}, and measured in $10\log_{10}\frac{E_S h^2_{SR_i}}{2\sigma^2_{SR_i}}$ (dB). The average SNR for channel is S-R_i is $\overline{\text{SNR}}_{SR_i}$, and measured in $10\log_{10}\frac{E_S}{2\sigma^2_{SR_i}}$ (dB).

In the second phase, each relay R_i sends the processed signals l_i to the destination through (mutually orthogonal) channel R_i-D. The destination receives, respectively,

$$y_{R_iD} = h_{R_iD}\beta_i l_i + n_{R_iD}, \tag{3.2}$$

where $i = 1, 2, \cdots, M$, $n_{R_iD} \sim \mathcal{N}(0, \sigma^2_{R_iD})$ is the AWGN at the destination, and β_i is the energy normalization/amplification factor. Let $E_{R_i} = E(|\beta l_i|^2)$ be the energy per bit used by the relay R_i, and the instantaneous R_i-D channel SNR during each frame is defined as SNR_{R_iD}, and measured in $10\log_{10}\frac{E_{R_i} h^2_{R_iD}}{2\sigma^2_{R_iD}}$ (dB). The average SNR for channel R_i-D is $\overline{\text{SNR}}_{R_iD}$, and measured in$10\log_{10}\frac{E_{R_i}}{2\sigma^2_{R_iD}}$ (dB).

The destination collects all the signals y_{R_iD} and performs appropriate decoding to get x_D, an estimate of the source x_S.

3.3 Representation of Soft Messages

We now discuss what are the best messages to forward to the relay. From the information theory perspective, if the R_i-D channels can be modeled as lossless processes, then as long as each relay is forwarding some "sufficient statistics" of the received signal y_{SR_i}, forwarding is done optimally. However, since each R_i-D channel is lossy in its own way, the choice of the sufficient statistics will make a difference.

We consider two cases: (i) When relay R_i can successfully decode and demodulate the data from S-R_i transmission (as indicated by the CRC), then these correctly-decoded bits constitute the simplest form of the sufficient statistics for the source data, and should therefore be forwarded to the destination (possibly employing additional error protection). (ii) Suppose that the CRC check does not pass, such that the relay is equipped with only the compromised data. Clearly, the relay should defer the hard decisions to the destination, and for that to happen, it is expected to do its best to pass along soft messages indicating the reliability of the reception.

There are a variety of choices for the sufficient statistics of y_{SR_i}, and all of them can be viewed as some representation of the probabilistic nature of the estimates. Popular examples include the probability $P(x = 0|y)$, the likelihood ratio $P(x = 0|y)/P(x = 1|y)$, the log-likelihood ratio $\log(P(x = 0|y)/P(x = 1|y)$, and the hyperbolic tangent of one half of the LLR $\tanh(\frac{1}{2}log\frac{P(x)=0|y}{P(x)=1|y})$ (where y is short for y_{SR_i} and x is short for x_S). In the case of independent channels with individual fading and noisy uncertainty, different choices will likely result in drastically different performances (in addition to different complexity). Below we evaluate and compare these choices of soft messages. (We assume proper scaling is always used to satisfy the energy constraint.)

(i) The probability of a bit being 0 [30], $P(x=0|y)$: Since $P(x=0|y)$ takes positive values only, transmitting it directly leads to a one-sided signal space that is energy-inefficient. Instead, a scaled-and-shifted version, such as $2P(x = 0|y) - 1$, makes for good antipodal signaling. However, we argue that $2P(x = 0|y) - 1$ is not a good choice either, because it can be rather sensitive to additive noise and that the impact of the noise is dependent on the value of $2P(x = 0|y) - 1$. Consider, for example, the two cases of $P(x = 0|y) = 0.95$ and $P(x = 0|y) = 0.54$, both of which encounter

the same additive noise of -0.10. The former represents a harmless case where the noise causes $2P(x=0|y)-1$ to change value from 0.90 to 0.80 (or for $P(x=0|y)$ to change from 0.95 to 0.90), which preserves the same confident and correct judgment of $x=0$. However, the latter becomes a harmful case as $2P(x=0|y)-1$ changes from 0.08 to -0.02, causing a preference change from x being "0" to x being "1". Such is particularly undesirable, because exactly when the soft messages need to be protected the most (i.e. $P(x=0|y)$ around 0.5), is when they are most vulnerable to noise.

(ii) The likelihood ratio [33], $\frac{P(x=0|y)}{P(x=1|y)}$: The likelihood ratio takes a value from 0 to ∞ with the tie set at 1. This is clearly not a good choice, not only because it is asymmetric and numerically unstable (when $P(x=1|y)$ is close to 0), but also because the value approaches infinity very quickly as $P(x=0|y) \to 1$, making it extremely difficult to normalize the transmit energy.

(iii) The log-likelihood ratio [3], $\log \frac{P(x=0|y)}{P(x=1|y)}$: Log-domain representations are in general more numerically stable than otherwise, with a far less chance for numerical overflow/underflow. The LLRs take symmetric values centered at 0, and have a desirable property of being "addable": namely, two or more LLR values of the same bit (assuming coming from different transmissions) can be directly added together to form the combined reliability of the bit. Further, the renowned Gaussianity property [34] states that, the LLRs extracted directly from a Gaussian channel follow an exact Gaussian distribution whose variance equals twice the mean. However, the drawback of LLRs is also obvious: unbounded value range which makes amplification and modulation difficult.

(iv) The hyperbolic tangent form [19], $\tanh(\frac{1}{2}\log \frac{P(x=0|y)}{P(x=1|y)})$: One of the biggest motivation for using the hyperbolic tangent is its optimality in signal relaying, in term of achieving the maximal SNR and the minimal BER at the destination in the relay system

with a single relay node. However, a big pitfall of the hyperbolic tangent value lies in the fact that

$$\tanh\left(\frac{1}{2}\log\frac{P(x=0|y)}{P(x=1|y)}\right) = \frac{1-e^{-\log\left(\frac{P(x=0|y)}{P(x=1|y)}\right)}}{1+e^{-\log\left(\frac{P(x=0|y)}{P(x=1|y)}\right)}} = 2P(x=0|y)-1, \tag{3.3}$$

and hence, for the reasons stated in the above, these soft messages are highly susceptible to noise when they are weak with rather small absolute values, making the worst case even worse. Since communications are all about rare events (such as an error event probability of once in a million), the worst case tends to dominate the performance.

(v) Range-limited LLR: In this paper, we propose range-limited LLR values as a very efficient choice for soft messages:

$$l_i = \begin{cases} \theta_i, & \log\frac{P(x_S=0|y_{SR_i})}{P(x_S=1|y_{SR_i})} \geq \theta_i \\ -\theta_i, & \log\frac{P(x_S=0|y_{SR_i})}{P(x_S=1|y_{SR_i})} \leq -\theta_i \\ \log\frac{P(x_S=0|y_{SR_i})}{P(x_S=1|y_{SR_i})}, & \text{otherwise}, \end{cases} \tag{3.4}$$

where the positive value θ_i sets the cap for the absolute LLR value.

When $\theta_i = 0$,

$$l_i = \begin{cases} 1, & \log\frac{P(x_S=0|y_{SR_i})}{P(x_S=1|y_{SR_i})} \geq 0 \\ -1, & \log\frac{P(x_S=0|y_{SR_i})}{P(x_S=1|y_{SR_i})} < 0, \end{cases} \tag{3.5}$$

By judiciously limiting the LLRs to a symmetric bounded range, we can still reap most the benefits of LLRs, without having to deal with infinite or excessively large

values. This would considerably reduce the peak-to-average-power ratio (PAPR) and ease the way to control the average transmit power.

3.4 Traditional Strategies and Z-forward Strategy

Having decided to use the range-limited LLRs as the soft-message, we now detail the operations of the proposed Z-forward strategy. We will first briefly describe some practical traditional relaying schemes and our new strategy. Despite the evolution of the relaying strategies, either linear or nonlinear one, either soft forward methods or hard forward strategies, three basic and practical strategies still most widely used are amplify-forward, decode-forward, and estimate-forward:

- Amplify-forward (AF): Relay nodes scale the (real-valued) S-R receptions in accordance to individual power constraint, and resend these scaled waveforms to the destination. AF can be regarded as a special case of the Z-forward with $\theta_i = +\infty$ for all i's.

- Decode-forward (DF): Relay nodes decode and demodulate the S-R receptions, and transmit their hard decisions (possibly in a channel-coded form) to the destination. DF is like Z-forward with $\theta_i = 0$ for all i's.

- Estimate-forward (EF, i.e. $\tanh$-forward) [19]: Relay nodes compute the minimal mean square error (MMSE) estimate of their S-R receptions, which results in the forwarded messages to take an $\tanh$ form: $\tanh(LLR/2)$. For a single-relay system, the scheme is shown to maximize the equivalent SNR at the destination.

- Piecewise-forward (PF) [31]: PF presents a simplification of $\tanh$-forward by approximating $\tanh(LLR/2)$ using a pre-determined, three-segment, piece-wise linear function. It is a special case of Z-forward with a fixed, non-optimized, single threshold satisfying $\tanh((\theta/2)/2) = 1/2$, which leads to $\theta = 2\ln(3)$.

The $\tanh$-forward scheme is derived based on maximizing the end-to-end SNR[3], the new Z-forward scheme is proposed with an aim to minimizing the end-to-end BER using adaptive thresholds. It is a rather general relaying scheme that subsumes AF, DF, and PF as its special case. The technical details of the proposed Z-forward scheme are given below:

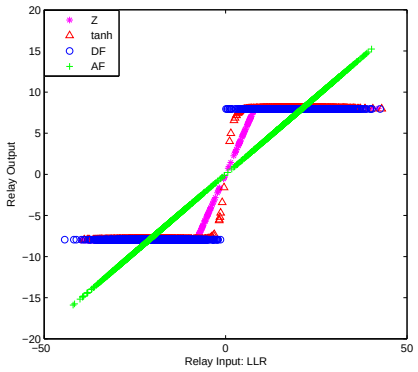
In the 2-hop S-R_i-D parallel relay model shown in Fig. 3.1, each relay R_i computes the LLR from the S-R_i channel reception:

$$\begin{aligned} LLR_i &\triangleq \log \frac{P(x_S = 0|y_{SR_i})}{P(x_S = 1|y_{SR_i})} \\ &= \log \frac{\frac{1}{\sqrt{2\pi\sigma_{SR_i}^2}} e^{-\frac{(y_{SR_i} - \sqrt{E_S} h_{SR_i})^2}{2\sigma_{SR_i}^2}}}{\frac{1}{\sqrt{2\pi\sigma_{SR_i}^2}} e^{-\frac{(y_{SR_i} + \sqrt{E_S} h_{SR_i})^2}{2\sigma_{SR_i}^2}}} = \frac{2\sqrt{E_S} h_{SR_i}}{\sigma_{SR_i}^2} y_{SR_i}. \end{aligned} \tag{3.6}$$

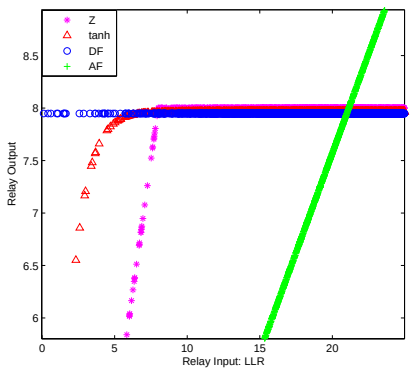
Inserting (3.1) into (3.6) leads to:

$$\begin{aligned} LLR_i &= \frac{2\sqrt{E_S} h_{SR_i}}{\sigma_S^2} (\sqrt{E_S} h_{SR_i} x_S + n_{SR_i}) \\ &= \frac{2E_S h_{SR_i}^2}{\sigma_{SR_i}^2} x_S + \frac{2\sqrt{E_S} h_{SR_i}}{\sigma_{SR_i}^2} n_{SR_i} \\ &= m_i x_S + n_{1i}, \end{aligned} \tag{3.7}$$

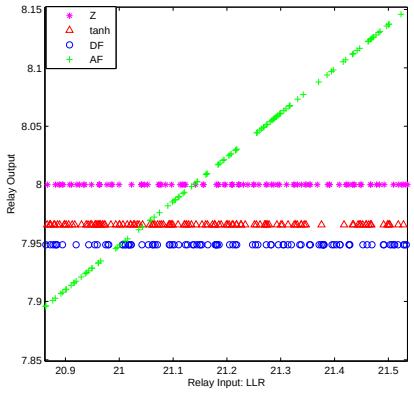
[3]Maximal SNR will guarantee minimum BER on AWGN channels, but not necessarily so on other channels.



(a)



(b)



(c)

Figure 3.2: Relaying function in different forward strategies

where $m_i = \frac{2E_S h_{SR_i}^2}{\sigma_{SR_i}^2}$, and $n_{1i} \sim \mathcal{N}(0, \sigma_{1i}^2) = \mathcal{N}(0, \frac{4E_S h_{SR_i}^2}{\sigma_{SR_i}^2})$.

These exact LLR values are then truncated according to (5.28) and (3.5) before being forwarded to the destination. Fig. 3.2(a) illustrates the ensemble statistics of the messages used in various forward schemes conforming to a normalized power constraint at the relay. The x-axis denotes the LLR calculated from the channel reception using (3.6). The y-axis represents the different message representations (as a function of LLR). We used an ensemble size of 5000, and the S-R channel SNR of 7 dB. Here the threshold of the proposed Z-forward strategy is set as $\theta = 8$. (The optimal value of θ will be discussed in the next Section.) Fig. 3.2(b) and 3.2(c) are enlarged version of fig. 3.2(a) with respect to different LLR's regions. The DF, $\tanh$ and Z curves all reach a ceiling floor, and almost overlap when the absolute value of the LLR is larger than 8. Since the thresholds as described at the beginning of this section are different in these schemes, the ceiling of DF and $\tanh$ scheme is slightly lower than that in Z-forward.

The overall performance depends heavily on the choice of the thresholds θ_i at each relay node R_i. Different thresholds would essentially lead to different forwarding strategies, as well as different BER performances. Two extreme cases of our Z-forward strategy is AF, which has a sufficiently high threshold, and DF, whose threshold equals zero. Next we will formulate the threshold selection problem as an optimization problem, and solve it for the single-relay and the double-relay systems.

A number of metrics are available to optimize θ_i. Here we consider minimizing the

end-to-end BER P_e.

$$\begin{aligned} &\min\ P_e(\theta_1, ..., \theta_M) \\ &\text{s.t. } \beta_i^2 E[|l_i|^2] = E_{R_i}, \\ &\quad \theta_i \geq 0, \quad i = 1, ...M. \end{aligned} \tag{3.8}$$

where $E[|l_i^2|]$ is computed by (3.9) when $\theta_i > 0$, $E[|l_i^2|] = 1$ when $\theta_i = 0$, $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{u^2}{2}} du$.

$$\begin{aligned} E[|l_i^2|] &= \theta_i^2 (\int_{-\infty}^{-\theta_i} \frac{1}{\sqrt{2\pi}\sigma_{1i}} e^{\frac{-(l_i - m_i)^2}{2\sigma_{1i}^2}} dl_i \\ &+ \int_{\theta_i}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_{1i}} e^{\frac{-(l_i - m_i)^2}{2\sigma_{1i}^2}} dl_i) + \int_{-\theta_i}^{+\theta_i} \frac{l_i^2}{\sqrt{2\pi}\sigma_{1i}} e^{\frac{-(l_i - m_i)^2}{2\sigma_{1i}^2}} dl_i \\ &= \theta_i^2 \left(Q\left(\frac{\theta_i + m_i}{\sigma_{1i}}\right) + Q\left(\frac{\theta_i - m_i}{\sigma_{1i}}\right) \right) \\ &+ (m_i^2 + \sigma_{1i}^2) \left(1 - \left(Q\left(\frac{\theta_i + m_i}{\sigma_{1i}}\right) + Q\left(\frac{\theta_i - m_i}{\sigma_{1i}}\right) \right) \right) \\ &+ \frac{\sigma_{1i}}{\sqrt{2\pi}} \left((m_i - \theta_i) e^{\frac{-(m_i + \theta_i)^2}{2\sigma_{1i}^2}} - (m_i + \theta_i) e^{\frac{-(m_i - \theta_i)^2}{2\sigma_{1i}^2}} \right) \\ &= (\theta_i^2 - m_i^2 - \sigma_{1i}^2) \left(Q\left(\frac{\theta_i + m_i}{\sigma_{1i}}\right) + Q\left(\frac{\theta_i - m_i}{\sigma_{1i}}\right) \right) \\ &+ m_i^2 + \sigma_{1i}^2 \\ &+ \frac{\sigma_{1i}}{\sqrt{2\pi}} \left((m_i - \theta_i) e^{\frac{-(m_i + \theta_i)^2}{2\sigma_{1i}^2}} - (m_i + \theta_i) e^{\frac{-(m_i - \theta_i)^2}{2\sigma_{1i}^2}} \right). \end{aligned} \tag{3.9}$$

3.5 Threshold Selection in Single-Relay Systems

We first consider the single-relay system. Since there is only one active relay in the system, the relay node index i is conveniently dropped.

The S-R-D transmission forms a Markov chain, and hence the average end-to-end BER of the overall system P_e can be expressed by

$$\begin{aligned} P_e =& P(x_s = 1)(P(l|x_s = 1)P(x_R = -1|l)) \\ &+ P(x_s = -1)(P(l|x_s = -1)P(x_R = 1|l)) \\ =& P(l|x_s = 1)P(x_R = -1|l). \end{aligned} \tag{3.10}$$

The second equality comes from the fact that the channel is symmetric and the signal space satisfies the geometric uniformity, and hence, without loss of generality, we assume $x_s = +1$ is transmitted.

Given that the relay performs "sign preserving" relaying, the optimal decision rule at the destination is to decide on $x_R = 1$ if $y_{RD} \geq 0$ and on $x_R = -1$ otherwise [26]. The end-to-end BER can be then written as

$$P_e = P(l|x_s = 1)P(y_{RD} < 0|l). \tag{3.11}$$

The soft message l to be forwarded to the destination can be characterized in three sections:

$$l = \begin{cases} \theta, & \text{with probability } p_\theta = Q(\frac{\theta - m}{\sigma_1}) \\ LLR, & \text{with pdf } f_l = \frac{1}{\sqrt{2\pi}\sigma_1} e^{\frac{-(l-m)^2}{2\sigma_1^2}} \\ -\theta, & \text{with probability } p_{-\theta} = Q(\frac{\theta + m}{\sigma_1}). \end{cases} \tag{3.12}$$

Table 3.1: Optimal LLR thresholds θ vs the SNR (dB) of the channels SR and RD in Z-forward strategy in a single-relay system

$SR \setminus RD$	0	1	2	3	4	5	6	7	8	9	10
0	1.25	1.05	0.85	0.70	0.50	0.35	0.20	0.1	0.05	0	0
1	1.30	1.10	0.90	0.70	0.50	0.35	0.20	0.1	0.05	0	0
2	1.35	1.15	0.95	0.75	0.55	0.40	0.20	0.1	0.05	0	0
3	1.35	1.20	1.00	0.75	0.55	0.40	0.20	0.1	0.05	0	0
4	1.40	1.20	1.00	0.80	0.60	0.40	0.25	0.1	0.05	0	0
5	1.40	1.20	1.00	0.80	0.60	0.40	0.25	0.1	0.05	0	0
6	1.40	1.20	1.00	0.80	0.60	0.40	0.25	0.1	0.05	0	0
7	1.45	1.20	1.00	0.80	0.60	0.40	0.25	0.1	0.05	0	0
8	1.45	1.20	1.00	0.80	0.60	0.40	0.25	0.1	0.05	0	0
9	1.45	1.25	1.05	0.80	0.60	0.40	0.25	0.1	0.05	0	0
10	1.45	1.25	1.05	0.80	0.60	0.40	0.25	0.1	0.05	0	0

Since the destination receives a Gaussian signal $y_{RD} \sim \mathcal{N}(\beta h_{RD} l, \sigma_{RD}^2)$, we have

$$\begin{aligned}
&P(l=\theta|x_s=1)P(y_{RD}<0|l=\theta)\\
=&\int_{-\infty}^{-\theta}\frac{1}{\sqrt{2\pi}\sigma_1}e^{\frac{-(l-m)^2}{2\sigma_1^2}}dl\int_{-\infty}^{0}\frac{1}{\sqrt{2\pi}\sigma_{RD}}e^{\frac{-(y+\beta h_{RD}\theta)^2}{2\sigma_{RD}^2}}dy\\
=&Q\left(\frac{\theta+m}{\sigma_1}\right)\left(1-Q\left(\frac{\beta h_{RD}\theta}{\sigma_{RD}}\right)\right).
\end{aligned} \tag{3.13}$$

$$\begin{aligned}
&P(-\theta<l<\theta|x_s=1)P(y_{RD}<0|-\theta<l<\theta)\\
=&\int_{-\theta}^{\theta}\frac{1}{\sqrt{2\pi}\sigma_1}e^{\frac{-(l-m)^2}{2\sigma_1^2}}\left(\int_{-\infty}^{0}\frac{1}{\sqrt{2\pi}\sigma_{RD}}e^{\frac{-(y-\beta h_{RD}l)^2}{2\sigma_{RD}^2}}dy\right)dl.
\end{aligned} \tag{3.14}$$

$$\begin{aligned}
&P(l=-\theta|x_s=1)P(y_{RD}<0|l=-\theta)\\
=&\int_{\theta}^{+\infty}\frac{1}{\sqrt{2\pi}\sigma_1}e^{\frac{-(l-m)^2}{2\sigma_1^2}}dl\int_{-\infty}^{0}\frac{1}{\sqrt{2\pi}\sigma_{RD}}e^{\frac{-(y-\beta h_{RD}\theta)^2}{2\sigma_{RD}^2}}dy\\
=&Q\left(\frac{\theta-m}{\sigma_1}\right)Q\left(\frac{\beta h_{RD}\theta}{\sigma_{RD}}\right).
\end{aligned} \tag{3.15}$$

Table 3.2: Optimal normalized thresholds $\beta\theta$ vs the SNR (dB) of the channels SR and RD in Z-forward strategy in a single-relay system

$SR \setminus RD$	0	1	2	3	4	5	6	7	8	9	10
0	1.05	1.04	1.03	1.03	1.02	1.01	1.01	1.00	1.00	1.00	1.00
1	1.03	1.03	1.02	1.02	1.01	1.01	1.00	1.00	1.00	1.00	1.00
2	1.02	1.02	1.02	1.01	1.01	1.01	1.00	1.00	1.00	1.00	1.00
3	1.01	1.01	1.01	1.01	1.01	1.00	1.00	1.00	1.00	1.00	1.00
4	1.01	1.01	1.01	1.01	1.01	1.00	1.00	1.00	1.00	1.00	1.00
5	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
6	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
7	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
8	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
9	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
10	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Gathering (3.13), (3.14), (3.15) and (3.10), we can get the expression of the end-to-end BER as a function of the threshold θ, $P_e(\theta)$:

$$\begin{aligned}
P_e(\theta) =& Q\left(\frac{\theta+m}{\sigma_1}\right)\left(1-Q\left(\frac{\beta h_{RD}\theta}{\sigma_{RD}}\right)\right)+Q\left(\frac{\theta-m}{\sigma_1}\right)Q\left(\frac{\beta h_{RD}\theta}{\sigma_{RD}}\right) \\
&+\int_{-\theta}^{\theta}\frac{1}{\sqrt{2\pi}\sigma_1}e^{\frac{-(l-m)^2}{2\sigma_1^2}}\left(\int_{-\infty}^{0}\frac{1}{\sqrt{2\pi}\sigma_{RD}}e^{\frac{-(y-\beta h_{RD}l)^2}{2\sigma_{RD}^2}}dy\right)dl \\
=& Q\left(\frac{\theta+m}{\sigma_1}\right)\left(1-Q\left(\frac{\beta h_{RD}\theta}{\sigma_{RD}}\right)\right) \\
&+Q\left(\frac{\theta-m}{\sigma_1}\right)Q\left(\frac{\beta h_{RD}\theta}{\sigma_{RD}}\right)\int_{-\theta}^{\theta}\frac{1}{\sqrt{2\pi}\sigma_1}e^{\frac{-(l-m)^2}{2\sigma_1^2}}Q\left(\frac{\beta h_{RD}l}{\sigma_{RD}}\right)dl.
\end{aligned} \tag{3.16}$$

After simplification, we get

$$P_e(\theta)=\frac{1}{2}+\left(Q\left(\frac{\theta+m}{\sigma_1}\right)-Q\left(\frac{\theta-m}{\sigma_1}\right)\right)\left(\frac{1}{2}-Q\left(\frac{\beta h_{RD}\theta}{\sigma_{RD}}\right)\right)- \frac{1}{\sqrt{2}\pi\sigma_1}I\left(\frac{\beta h_{RD}l}{\sigma_{RD}},\frac{m}{\sqrt{2}\sigma_{RD}},\frac{\theta-m}{\sqrt{2}\sigma_1}\right) - \frac{1}{\sqrt{2}\pi\sigma_1}I\left(\frac{-\beta h_{RD}l}{\sigma_{RD}},\frac{m}{\sqrt{2}\sigma_{RD}},\frac{\theta+m}{\sqrt{2}\sigma_1}\right). \quad (3.17)$$

where $I(a,b,x)$ [35] is defined as

$$I(a,b,x) = \int_0^x e^{-t^2}\int_0^{ax+b} e^{-s^2}dsdt = \frac{\sqrt{\pi}}{2}\int_0^x e^{-t^2}(1-2Q(\sqrt{2}(ax+b)))dt. \quad (3.18)$$

This BER formulation thus completes formulation of the optimization problem in (3.8) (for the single-relay case).

To better illustrate the problem, we consider a simple case with unit transmission energy at the source and at the relay, and unit fading coefficient for channel S-R and R-D. The solutions to the optimization problem can be obtained using the exhaustive grid search method [36]. The grid search method proceeds as follows: Set the search range as $\theta = 0 : \Delta\theta : \Theta$, where $\Delta\theta$ is the unit search step or the search precision (0.05 for the single-relay case), and Θ is a very large value (1000 for the single-relay case). Fig. 3.3 presents the optimal thresholds as a function of SNR_{SR} and SNR_{RD}. In the figure, the SNR of the equivalent AWGN channels varies from 0 to 10 dB. The general results in Fig. 3.3 work for the AWGN and the block fading channel, as the block Rayleigh fading channel is composed of a full spectrum of instantaneous Gaussian channel with SNR $|h_{SR}|^2/2\sigma_{SR}^2$ and $|h_{RD}|^2/2\sigma_{RD}^2$. (Given h_{SR} and h_{RD} perfectly known, the block

fading channel becomes an instantaneous AWGN channel.)

Table. 3.1 lists the searched values of the optimal thresholds, which will be used in Section 5.7 for simulation. It shows that in the SNR region of interested, all the optimal LLR thresholds take a fairly small value. As one can observe, for a fixed R-D SNR, the threshold θ of the range-limited LLR would generally increase with the increase of the S-R SNR; and for a fixed S-R SNR, the threshold θ would instead decrease as the R-D SNR increases. The former is due to the fact that as the S-R SNR increases, the mean of the received LLR from the S-R channel (shown in Eq. (7)) tends to increase, and hence the threshold increases along with it. The latter may be attributed to the fact that as the R-D channel gets increasingly better, it would inject less negative impact on the transmission, and hence it would make sense for the relay node to perform DF (the smaller the threshold, the more the Z-forward resembles DF). To shed further insight into the optimal thresholds, Table. 3.2 lists the value of $\beta\theta$ (and when $\theta = 0$, it reduces to DF, so $\beta\theta = 1$ even is shown). It reveals that with the fixed R-D SNR, the normalized threshold would increase as the S-R SNR decreases. This suggests that when the channel is in a poor condition, Z-forward behaves more like AF. On the other hand, when both channel R-D and S-R have high SNR, $\beta\theta \rightarrow 1.00$, which suggests that Z-forward behaves like DF.

3.6 Threshold Selection in Multi-Relay Systems

The more intriguing case is when multiple active relays are involved in the system, allowing the system to reap off beneficial diversity gain, at the cost of a higher complexity. It should also be cautioned, however, that the optimal threshold derived for the

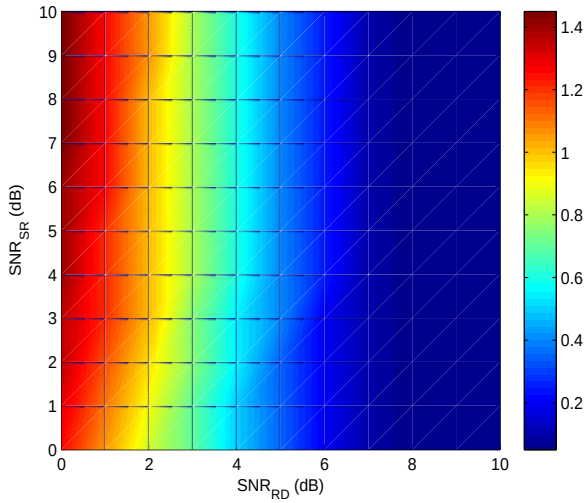


Figure 3.3: Optimal LLR thresholds of Z-forward strategy with different SNR_{SR} and SNR_{RD} in a single-relay system

single-relay systems is no longer optimal here, and the performance of the entire system must depend on the quality of all the channel segments. As the number of relay increases, the searching for the optimal threshold becomes increasingly harder. Below, we first consider the case of a double-relay diamond network, and propose two sub-optimal methods to search for the thresholds.

3.6.1 BER Performance in Two-relay Systems

Before we derive the end-to-end BER, we briefly discuss the decoding strategy at the destination. Given the availability of the channel CSI, the destination can perform MRC [37]. Though MRC is not the optimal estimation method, performance analysis with ML estimator is quite complicated. With MRC, the signals from every relay will be co-phased and their amplitudes appropriately weighted, before being combined. In our calculation below, the average transmission power at each relay is assumed unit power,

i.e. $E_{R_1} = E_{R_2} = 1$.

Let α_i be the combining weight for the signal from the S-R_i-D transmission, $i = 1, 2$. We note that the range-limited LLR l_i's are no longer Gaussian, and although we are able to calculate the exact pdfs for them (see below), it is convenient to simply apply a Gaussian approximation [38] $l_i \sim \mathcal{N}(\mu_i, \tilde{\sigma}_i^2)$ in the calculation of MRC weights α_i. The Gaussian approximation allows us to approximate the soft-messages l_i as

$$l_i = \mu_i x_S + \tilde{n}_i, \tag{3.19}$$

where

$$\begin{aligned}\mu_i = & \theta_i \left(Q\left(\frac{\theta_i - m_i}{\sigma_{1i}}\right) - Q\left(\frac{\theta_i + m_i}{\sigma_{1i}}\right)\right) \\ & + \int_{-\theta_i}^{\theta_i} l_i \frac{1}{\sqrt{2\pi}\sigma_{1i}} e^{\frac{-(l_i - m_i)^2}{2\sigma_{1i}^2}} dl_i,\end{aligned} \tag{3.20}$$

$$\tilde{\sigma}_i^2 = E\left[|l_i|^2\right] - u_i^2. \tag{3.21}$$

It then follows that the signal received from the R_i-D channel can be written as

$$\begin{aligned} y_{R_iD} &= h_{R_iD}\beta_i(\mu_i x_S + \tilde{n}_i) + n_{R_iD} \\ &= h_{R_iD}\beta_i\mu_i x_S + h_{R_iD}\beta_i\tilde{n}_i + n_{R_iD}, \end{aligned} \tag{3.22}$$

which leads to the combining weights

$$\alpha_i = \frac{\beta_i \mu_i h_{R_iD}}{\beta_i^2 h_{R_iD}^2 \tilde{\sigma}_i^2 + \sigma_{R_iD}^2}. \tag{3.23}$$

After MRC, the destination can simply make a binary decision on the original source x_S via sign detection.

Now that we have an efficient decoder, we proceed to the calculation of the end-to-end BER. The destination obtains the signal y_D (via MRC),

$$y_D = \alpha_1 y_{R_1 D} + \alpha_2 y_{R_2 D}, \tag{3.24}$$

and makes a binary decision (via sign detection).

From eq. (3.2), we have $y_{R_i D} = \beta_i h_{R_i D} l_i + n_{R_i D}$. Let $f(l_i|x_S = +1)$ and $f(l_i|x_S = -1)$ denote the conditional pdf of l_i, conditioned on $x_s = +1$ and $x_s = -1$ being transmitted, respectively. Consider that l_i take the form of a piece-wise linear function of LLR, we get to compute the exact pdf as

$$\begin{aligned} f(l_i|x_S = +1) &= \delta(l_i - \theta_i) Q\left(\frac{\theta_i - m_i}{\sigma_{1i}}\right) \\ &\quad + \frac{1}{\sqrt{2\pi}\sigma_{1i}} e^{\frac{-(l_i - m_i)^2}{2\sigma_{1i}^2}} + \delta(l_i + \theta_i) Q\left(\frac{\theta_i + m_i}{\sigma_{1i}}\right), \\ f(l_i|x_S = -1) &= \delta(l_i - \theta_i) Q\left(\frac{\theta_i + m_i}{\sigma_{1i}}\right) \\ &\quad + \frac{1}{\sqrt{2\pi}\sigma_{1i}} e^{\frac{-(l_i + m_i)^2}{2\sigma_{1i}^2}} + \delta(l_i + \theta_i) Q\left(\frac{\theta_i - m_i}{\sigma_{1i}}\right), \end{aligned} \tag{3.25}$$

where $\delta(\bullet)$ is the Dirac delta function.

We use $f_{n_{R_i D}}$ to denote the pdf of the noise $n_{R_i D}$:

$$f(n_{R_i D}) = \frac{1}{\sqrt{2\pi}\sigma_{R_i D}} e^{\frac{-\left(n_{R_i D}\right)^2}{2\sigma_{R_i D}^2}}. \tag{3.26}$$

$$f(y_{R_iD}|x_S=+1)=Q\left(\frac{\theta_i-m_i}{\sigma_{1i}}\right)\mathcal{N}\left(\beta_i h_{R_iD}\theta_i;\sigma_{R_iD}\right)+Q\left(\frac{\theta_i+m_i}{\sigma_{1i}}\right)$$
$$\mathcal{N}\left(-\beta_i h_{R_iD}\theta_i;\sigma_{R_iD}\right)+\mathcal{N}\left(\beta_i h_{R_iD}m_i;\phi\right)\left(Q\left(\frac{A-B}{\sigma_{R_iD}\sigma_{1i}\phi}\right)-Q\left(\frac{A+B}{\sigma_{R_iD}\sigma_{1i}\phi}\right)\right), \tag{3.28}$$

$$f(y_{R_iD}|x_S=-1)=Q\left(\frac{\theta_i-m_i}{\sigma_{1i}}\right)\mathcal{N}\left(-\beta_i h_{R_iD}\theta_i;\sigma_{R_iD}\right)+Q\left(\frac{\theta_i+m_i}{\sigma_{1i}}\right)$$
$$\mathcal{N}\left(\beta_i h_{R_iD}\theta_i;\sigma_{R_iD}\right)+\mathcal{N}\left(-\beta_i h_{R_iD}m_i;\phi\right)\left(Q\left(\frac{C-D}{\sigma_{R_iD}\sigma_{1i}\phi}\right)-Q\left(\frac{C+D}{\sigma_{R_iD}\sigma_{1i}\phi}\right)\right), \tag{3.29}$$

$$A=\sigma^2_{R_iD}m_i+\beta_i h_{R_iD}\sigma^2_{1i}y_{R_iD},\quad B=\sigma^2_{R_iD}\theta_i+\beta_i h_{R_iD}\sigma^2_{1i}\beta_i h_{R_iD}\theta_i$$
$$C=\sigma^2_{R_iD}m_i-\beta_i h_{R_iD}\sigma^2_{1i}\beta_i h_{R_iD}\theta_i, D=\sigma^2_{R_iD}\theta_i+\beta_i h_{R_iD}\sigma^2_{1i}y_{R_iD}$$
$$\phi=\sqrt{\sigma^2_{R_iD}+\left(\beta_i h_{R_iD}\right)^2\sigma^2_{1i}},\qquad \mathcal{N}\left(a,b\right)=\frac{1}{\sqrt{2\pi}b}e^{\frac{-\left(y_{R_iD}-a\right)^2}{2b^2}}$$

Thus the pdf of the signal received from each S-R_i-D transmission can be expressed by

$$f(y_{R_iD}|x_S=+1)=\frac{1}{\beta_i h_{R_iD}}f\left(\frac{l_i}{\beta_i h_{R_iD}}|x_S=+1\right)\otimes f\left(n_{R_iD}\right),$$
$$f(y_{R_iD}|x_S=-1)=\frac{1}{\beta_i h_{R_iD}}f\left(\frac{l_i}{\beta_i h_{R_iD}}|x_S=-1\right)\otimes f\left(n_{R_iD}\right), \tag{3.27}$$

where $\otimes$ indicates the convolutional operation. The analytical results can be found in (3.28) and (3.29).

Again, without loss of generality, we assume that $x_S=+1$ was transmitted. The

overall instant end-to-end BER can be calculated as

$$P_e(\theta_1,\theta_2) = P(y_D<0) = P(\alpha_1 y_{SR_1} + \alpha_2 y_{SR_2} < 0)$$
$$= \int_{-\infty}^{+\infty} \int_{-\infty}^{-\frac{\alpha_1}{\alpha_2} y_{R_1D}} f(y_{R_2D}|x_S{=}{+}1) f(y_{R_1D}|x_S{=}{+}1) dy_{R_2D} dy_{R_1D}. \tag{3.30}$$

This BER formulation thus completes formulation of the optimization problem in (3.8) for the double-relay case.

The BER of M-relay system ($M \geq 2$) can be computed by the similar way.

$$P_e(\theta_1, ...\theta_M)$$
$$= P(y_D{<}0){=}P(\sum_{i=1}^{M} \alpha_i y_{R_iD}{<}0){=}P(y_{R_MD}{<}-\frac{\sum_{i=1}^{M-1} \alpha_i y_{R_iD}}{\alpha_M})$$
$$= \int_{-\infty}^{+\infty} ... \int_{-\infty}^{+\infty} \int_{-\infty}^{-\frac{\sum_{i=1}^{M-1} \alpha_i y_{R_iD}}{\alpha_M}} \prod_{i=1}^{M} f(y_{R_iD}|x=+1) dy_{R_MD} ... dy_{R_1D}. \tag{3.31}$$

To help illustrate, we demonstrate in Fig. 3.4 and Table 3.3 the optimal threshold results for the 2-relay AWGN channels with $\text{SNR}_{SR_1} = \text{SNR}_{SR_2}$, and $\text{SNR}_{R_1D} = \text{SNR}_{R_2D}$. The search range is as $\theta_1 = 0 : \Delta\theta_1 : \Theta_1$ and $\theta_2 = 0 : \Delta\theta_2 : \Theta_2$, and the grid parameters are set as $\Delta\theta_1 = \Delta\theta_2 = 0.1$ and $\Theta_1 = \Theta_2 = 1000$. Since the two parallel channel are statistically identical, they have the same MRC weights and the same thresholds θ. It is worth pointing out that, even though the second relay channel is like an exact replica of the first, when both are employed, the optimal thresholds (optimized in the double-relay context) are drastically different from those optimized in the

Table 3.3: Optimal LLR thresholds of Z-forward in a double-relay system, $\text{SNR}_{SR_1} = \text{SNR}_{SR_2}$, $\text{SNR}_{R_1D} = \text{SNR}_{R_2D}$, $\theta_1 = \theta_2$

$SNR_{SR} \backslash SNR_{RD}$	0	1	2	3	4	5	6	7	8	9	10
0	2.7	2.9	3.2	3.5	3.7	3.9	4.1	4.3	4.4	4.5	4.6
1	2.8	3.1	3.4	3.7	4.0	4.3	4.5	4.7	4.9	5.0	5.2
2	2.9	3.2	3.5	3.9	4.3	4.7	5.0	5.2	5.4	5.6	5.8
3	2.9	3.2	3.6	4.1	4.7	5.1	5.6	5.9	6.1	6.3	6.5
4	3.0	3.3	3.7	4.3	4.9	5.6	6.2	6.6	6.9	7.2	7.4
5	3.0	3.3	3.8	4.4	5.1	6.0	6.8	7.4	7.9	8.2	8.5
6	3.0	3.3	3.8	4.5	5.3	6.3	7.4	8.3	9.0	9.5	9.8
7	3.0	3.4	3.8	4.5	5.4	6.5	7.8	9.2	10.3	11.1	11.5
8	3.0	3.4	3.8	4.5	5.4	6.6	8.0	9.8	11.5	12.8	13.6
9	3.0	3.4	3.8	4.5	5.4	6.6	8.1	10.1	12.4	14.6	16.1
10	3.0	3.4	3.9	4.5	5.4	6.6	8.1	10.1	12.6	15.6	18.5

Table 3.4: Optimal LLR thresholds of Z-forward in a double-relay system, $\text{SNR}_{SR_2} = \text{SNR}_{SR_1} - 3 = \text{SNR}_{R_2D} = \text{SNR}_{R_1D} - 3$, $\theta_1 \neq \theta_2$

SNR_{SR_1}	0	1	2	3	4	5	6	7	8	9
(t_1,t_2)	(2,3)	(2,3)	(2,3)	(2,4)	(3,4)	(4,4)	(4,5)	(6,7)	(6,8)	(7,9)

Table 3.5: Optimal LLR thresholds of Z-forward a double-relay system, $\text{SNR}_{SR_2} = \text{SNR}_{SR_1} - 3$, $\text{SNR}_{R_2D} = \text{SNR}_{R_1D} - 3$, $\theta_1 = \theta_2$

$SNR_{SR_1} \backslash SNR_{R_1D}$	0	1	2	3	4	5	6	7	8	9	10
0	2.0	2.0	2.0	3.0	3.0	3.0	3.0	4.0	4.0	5.0	6.0
1	2.0	2.0	2.0	3.0	3.0	3.0	3.0	4.0	5.0	6.0	7.0
2	2.0	2.0	2.0	3.0	3.0	3.0	4.0	4.0	6.0	7.0	8.0
3	2.0	2.0	3.0	3.0	3.0	4.0	4.0	5.0	6.0	7.0	8.0
4	2.0	2.0	3.0	3.0	3.0	4.0	4.0	5.0	6.0	7.0	8.0
5	2.0	2.0	3.0	3.0	4.0	4.0	4.0	5.0	6.0	7.0	8.0
6	2.0	3.0	3.0	3.0	4.0	4.0	5.0	5.0	6.0	7.0	8.0
7	3.0	3.0	3.0	3.0	4.0	5.0	5.0	6.0	6.0	7.0	8.0
8	3.0	4.0	4.0	4.0	4.0	5.0	6.0	7.0	8.0	7.0	9.0
9	4.0	5.0	4.0	4.0	4.0	5.0	6.0	8.0	8.0	9.0	9.0
10	5.0	6.0	6.0	6.0	6.0	7.0	7.0	9.0	11.0	12.0	14.0

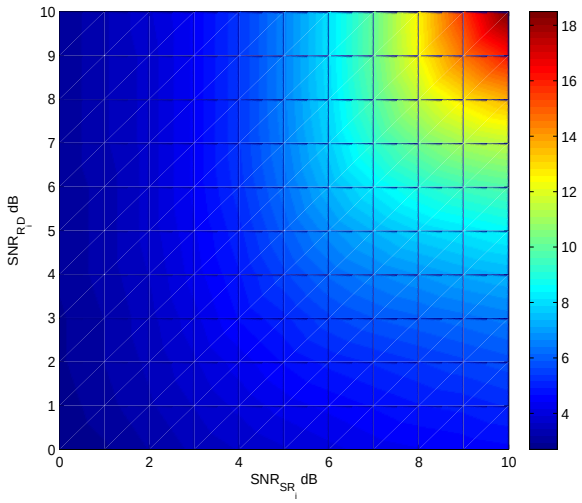


Figure 3.4: Optimal LLR thresholds of Z-forward in a double-relay system

single-relay context in Fig. 3.3 and Table 3.1. This suggests that the results obtained from single-relay systems may not serve the multi-relay systems as well as one would hope. The value of the thresholds are much larger than they are in Table 3.1. When the threshold is large, the message forwarded by the relay(s) (which are the Z-truncated LLR values) will preserve more reliability information from the original S-R LLR values, compared to smaller thresholds. The reason that it is beneficial for the 2-user case to preserve more "details" in reliability information, is that LLR details can be helpful when the two relaying branches combine their results. For example, it may be helpful to the common destination to know that one branch has a (normalized) LLR of $+3.75$ and the other has a (normalized) LLR of 3.19, rather than have two branches truncated to the same absolute value but with opposite signs (as in the case of a small threshold). In comparison, the single-user case does not have a second relaying branch to help it, and hence may not benefit much from preserving LLR details.

Table 3.4 shows the optimal threshold results with $SNR_{SR_2} = SNR_{R_2D} = SNR_{SR_1} -$

$3 = SNR_{R_1D} - 3$. The search range is as $\theta_1 = 0 : \Delta\theta_1 : \Theta_1$ and $\theta_2 = 0 : \Delta\theta_2 : \Theta_2$, and the grid parameters are set as $\Delta\theta_1 = \Delta\theta_2 = 1$ and $\Theta_1 = \Theta_2 = 500$. Since the two S-R-D are different, with the weaker one picking up a slightly larger value, and the difference between the optimal thresholds θ_1 and θ_2 is fairly small, we also evaluated the option of using the same threshold θ for both relays (even though the two channels are quite different). The results are shown in Table V. We see that the optimal thresholds increase with the increase of either the S-R SNR or the R-D SNR, meaning that the system favors the preservation of more soft information to the next stage. This is because the system now has two parallel relays that can help each other, and hence preserving the details can help the relays to collectively determine the overall data reliability in a finer manner.

3.6.2 Sub-optimal Z-forward in Multi-relay Systems

Clearly, the complexity of finding the optimal thresholds increases with the number of relays, and can become tedious when many active relays are present in the system. We thus propose to simplify the design by adopting a single threshold θ for all the relays in $M \geq 2$ parallel-relay systems.

Extensive experiments show that the thresholds will in general increase with the channel SNRs, and that the optimal value is usually greater than 2. Hence, we propose the following rule-of-thumb for the single "unified" threshold:

$$\theta = 2 + \frac{\max(\sum_{j=1}^{M}(\text{SNR}_{SR_j} + \text{SNR}_{R_jD}), 0)}{2M} + \varepsilon, \tag{3.32}$$

where SNR_{SR_i} and SNR_{R_iD} are the instantaneous SNR of channel S-R_i and R_i-D, ε is a modification coefficient.

We then perform neighborhood optimization of ϵ over its search region $[a, b]$, where the optimality is indicated by the largest corresponding BER whose expression is given in (3.31). Clearly, when the search range continues to increase (i.e. $a \to -\infty$ and $b \to \infty$), the neighborhood optimization eventually becomes a global, exhaustive grid search. In general, we found that a search region of $[-3, 3]$ seems sufficient. We term this simplified version *Z-suboptimal 1* scheme.

We can further simplify the mechanism by dropping the correction term ϵ and inserting in (3.32) the average SNR_{SR_i} and SNR_{R_iD} (rather than instantaneous SNR_{SR_i} and SNR_{R_iD}). This results in an extremely simple scheme, which we call *Z-suboptimal 2* scheme.

3.7 Estimation at Destination

In the multi-relay 2-hop network, we assume that the destination also knows the threshold that are used by the relays, so it can match its decoding process with the relay process. Two decoding methods are available at the destination. The first is to combine the signals from different relay nodes using MRC as in (3.23), and to make a hard decision on the combined signals. This estimation method is easy to carry out for

destination. Mathematically, the estimated bit can be expressed by

$$x_D = \begin{cases} +1, & y_D \geq 0 \\ -1, & y_D < 0. \end{cases} \tag{3.33}$$

In addition to the MRC, the destination may also perform a ML estimation. In section VI, we applied the Gaussian approximation to characterize the pdf of the received signal at the destination. Since l_i does not follow the Gaussian distribution exactly, the MRC decoding does not yield the true optimal estimate.

ML estimator can expect to produce (slightly) better results. We have already formulated the end-to-end BER in the previous section, and subsequently computed the thresholds θ_i for Z-forward. Based on these θ_i, we can derive the conditional pdf expression of the l_i, the soft-message that is to be forwarded by the ith relay, as shown in (3.28) and (3.29).

The ML estimator then makes the hard decision based on the following rule:

$$x_D = \begin{cases} +1, & \prod_{i=1}^{M} f(y_{R_iD}|x_S = +1) \geq \prod_{i=1}^{M} f(y_{R_iD}|x_S = +1) \\ -1, & \text{otherwise.} \end{cases} \tag{3.34}$$

We note that ML decoding is optimal in all cases, and can generate both soft and hard results. The downside, however, is the high complexity (in evaluating the conditional pdf in (3.28) and (3.29). In comparison, MRC is simpler, but could only generate optimal hard decisions in a single-relay system.

Table 3.6: Simulated forward schemes

	Z-forward	**AF**	**DF**	**EF/tanh-forwward**	**PF**
MRC	⁕	$\surd$	$\triangle$ [32]	• [19]	N/A
ML	⁕	$\surd$	$\triangle$ [39]	N/A	∇ [31]
AF/DF-adaptive	N/A	$\surd$	$\triangle$	N/A	N/A
Relay-selective	N/A	$\surd$	$\triangle$	•	N/A

⁕ Z-forward: Both MRC and ML are simulated in this section.
$\surd$ AF: MRC performs the same as ML, the simulation of MRC is provided.
$\triangle$ DF: Regular MRC with Gaussian approximation performs very bad. DF with ML [39] performs practically the same as C-MRC [32]. DF with C-MRC is simulated.
• EF: EF with ML is intractable due to nonlinearity of $\tanh$. EF with MRC is simulated.
∇ PF: MRC for PF performs the same as MRC for EF/$\tanh$-forward. PF with ML is simulated, which performs better than PF with MRC.

3.8 Numerical Results

This section evaluates the numerical results of the proposed forward scheme under both the AWGN and the block Rayleigh fading channels. The frame size in a block Rayleigh fading channel is set to be 500 bits. All the relays will individually follow the constraint of unit average transmission power (for each frame).

We evaluate the proposed Z-forward scheme, as well as a variety of existing schemes serving as the benchmark. The profiles of these schemes are described in Table VI. We also compared Z-forward with two kinds of adaptive schemes: 1) Adaptive AF/DF schemes in a double relay system, in which each relay node switches between AF and DF according to appropriate criterion, such as the SNR of the S-R channel, the theoretically estimated performance of the S-R-D channel, and the decoding result of the S-R channel. 2) Relay-selective AF, DF, and EF schemes in a double-relay system, in which the better relay (i.e., the better S-R-D channel) is instantaneously selected. Repetition code is used to ensure for a fair comparison in bandwidth and power among all the schemes.

We first test the Z-forward scheme with different fixed thresholds over AWGN channels in a diamond network. Suppose all the 4 channel segments have the same SNR. The SNR value varies from 0 to 10 dB. As shown in Fig. 3.5, the BER performance reveals obvious difference with different thresholds. When SNR_{R_iD} is low, smaller thresholds tend to achieve better performance; as SNR increases, the optimal threshold value also increases. This is consistent with the numerical results in Fig. 3.4.

Figures 3.6 and 3.7 evaluate the performance of Z-forward, AF, DF, and EF/$\tanh$ using MRC decoding on block Rayleigh fading channels, with statically-identical S-R-D channels in Fig. 3.6 and different S-R-D channels in Fig. 3.7. Both the optimal thresholds, and the two suboptimal simplifications, are evaluated with Z-forward. Several observations can be made. 1) In a single-relay system, AF performs the worst while all the others perform similar. 2) In a multi-relay system, AF performs the work at low SNRs, but catches up and outperforms DF and EF at high SNRs. Z-forward always yields the best results, with its simplified versions (which have very low complexity) performing on par with the other conventional schemes, and the one with the optimal thresholds perform strictly better. 3) The gains of Z-forward become considerably larger as the number of parallel relays increases. With 3 parallel relays, Z-forward suboptimal 2, suboptimal 1, and optimal have demonstrated about 0.5 dB, 0.8 dB and 1.5 dB gain over AF, respectively.

Figures 3.8 and 3.9 evaluate the performance of Z-forward, and PF using ML decoding on block Rayleigh fading channels, with statically-identical S-R-D channels in Fig. 3.8 and different S-R-D channels in Fig. 3.9. Z-forward with MRC is also shown for comparison purpose. We see that Z-forward suboptimal 1 and suboptimal 2 have an advantage of some 1 dB and 0.5 dB gain over PF.

In Fig. 3.10, Z-forward is compared to three AF/DF switching schemes based on the S-R channel condition [40], the theoretical performance under the current S-R-D channel condition, and the S-R decoding results at the relays [25]. In the first adaptively switching scheme (legend "S-R SNR"), AF is used if the instantaneous SNR of channel S-R is larger than the average SNR of the channel, and DF is used otherwise. In the second scheme (legend "S-R-D BER"), the theoretical S-R-D performance is calculated for both AF and DF, and the better one is picked. In the last scheme (legend "S-R decoding"), the relays will try to first make a hard-decision of the S-R transmission, if there is no error, then the system proceeds with DF. Otherwise, it goes with AF. In all the schemes, the destination knows exactly which one of AF and DF is employed by each relay, so that appropriate combining coefficients would be used. It can be observed that both AF/DF adaptive schemes based on S-R-D BER and S-R decoding results can reach the full diversity. We see that Z-forward always performs the best, because the adaptive schemes are like switching between Z-forward with $\theta = 0$ and $\theta = +\infty$, rather than using the optimal θ at all times.

We also compare Z-forward with relay selective schemes in a double-relay system in Fig. 3.11. In the relay-selective AF, DF, and EF schemes, the CSI of all the channels will be utilized to pick the best relay, such that the instantaneously better S-R-D channel is always selected and used. A rate 1/2 repetition code is adopted in the selective schemes for fairness in bandwidth and energy. We see DF relay-selective scheme performs better than the AF and EF selective ones. However, it falls behind the Z-forward scheme by almost 1.2 dB at the BER of 10^{-4}. Z-forward still produces the best performance of all.

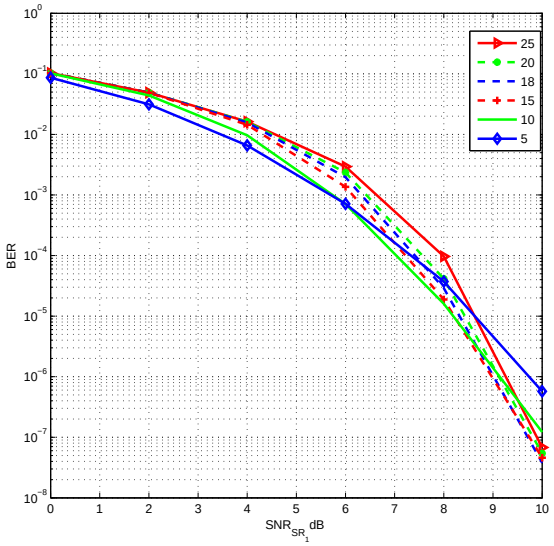


Figure 3.5: BER performance of Z-forward with different thresholds under AWGN channel with 2 relay nodes, MRC estimate, all the average channel SNRs are the same.

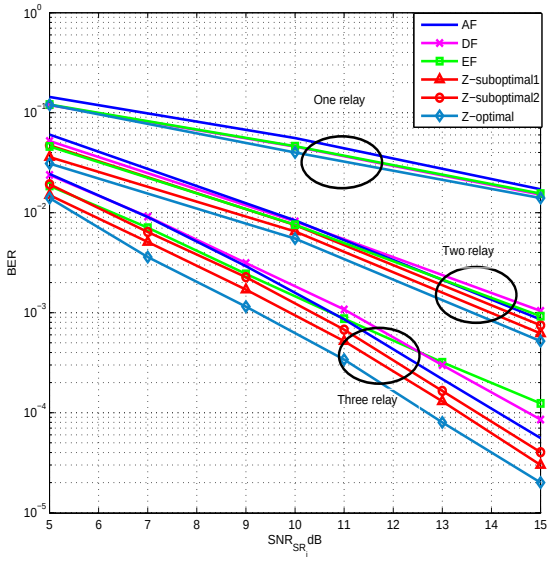


Figure 3.6: BER performance of different schemes under block Rayleigh fading channel with 1-3 relay nodes, MRC estimate, all the average channel SNRs are the same.

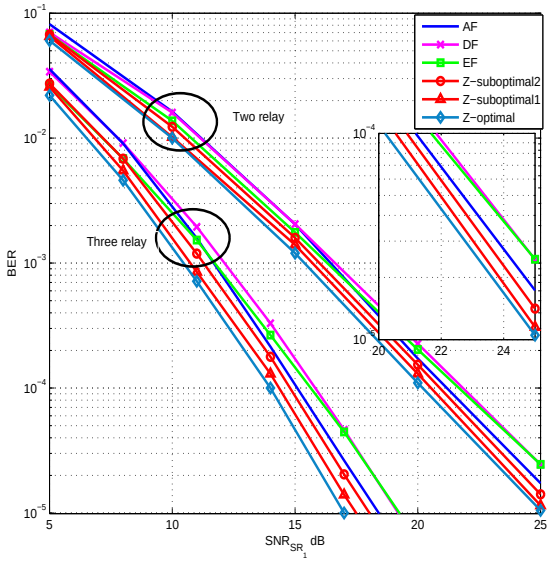


Figure 3.7: BER performance of different schemes under block Rayleigh fading channel with 2 or 3 relay nodes, MRC estimate, $\overline{\text{SNR}}_{SR_1} = \overline{\text{SNR}}_{R_1D} = \overline{\text{SNR}}_{SR_2} + 3 = \overline{\text{SNR}}_{R_2D} + 3 = \overline{\text{SNR}}_{SR_3} = \overline{\text{SNR}}_{R_3D}$.

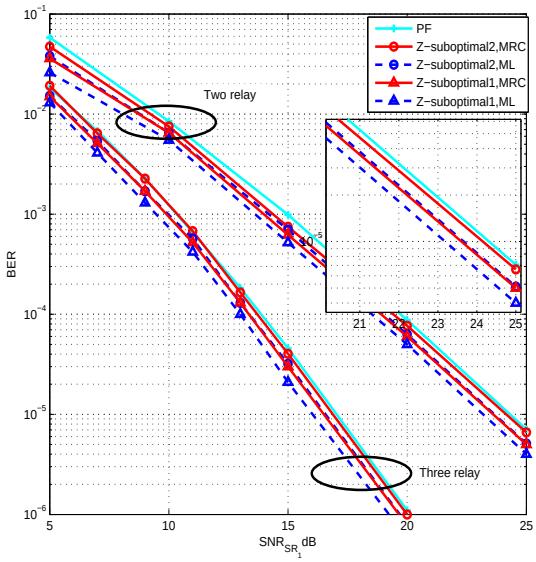


Figure 3.8: BER performance of different schemes under block Rayleigh fading channel with 2 or 3 relay nodes, ML and MRC estimate, all the average channel SNRs are the same

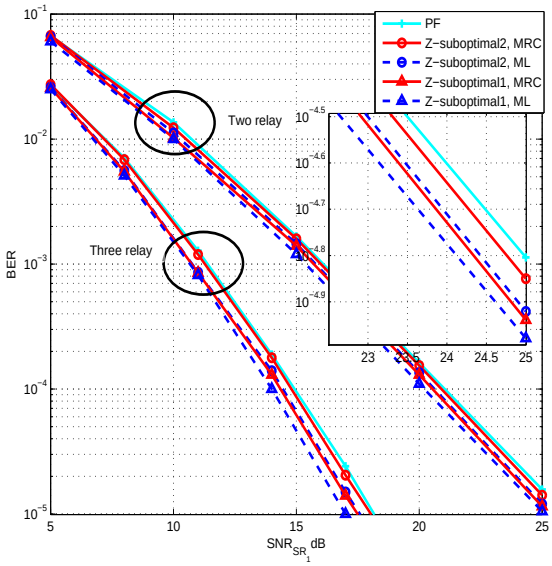


Figure 3.9: BER performance of different schemes under block Rayleigh fading channel with 2 or 3 relay nodes, ML and MRC estimate, $\overline{\text{SNR}}_{SR_1} = \overline{\text{SNR}}_{R_1D} = \overline{\text{SNR}}_{SR_2} + 3 = \overline{\text{SNR}}_{R_2D} + 3 = \overline{\text{SNR}}_{SR_3} = \overline{\text{SNR}}_{R_3D}$.

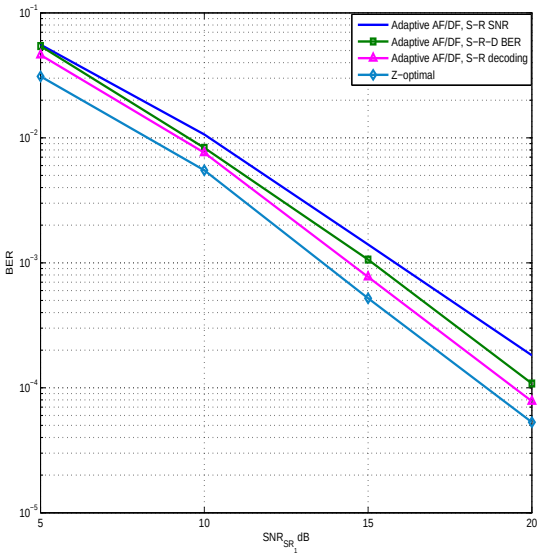


Figure 3.10: BER performance comparison of Z-forward schemes and different adaptive AF-DF schemes under block Rayleigh fading channel with 2 relay nodes, all the average channel SNRs are the same.

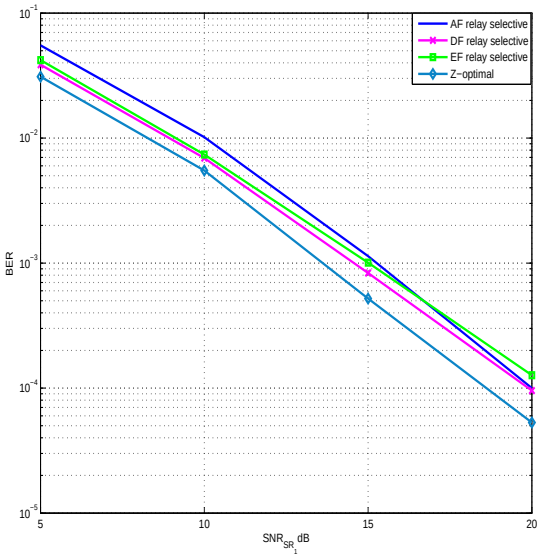


Figure 3.11: BER performance comparison of Z-forward schemes and different relay selective schemes under block Rayleigh fading channel with 2 relay nodes, all the average channel SNRs are the same.

Chapter 4

New Soft-Encoding Relay (SoER) Mechanisms for Parallel Relay Systems: Convolutional and Turbo Constructions

4.1 Introduction and Motivation

Wireless relay systems, by involving one or multiple relay nodes in message forwarding, promise several benefits including lower transmit power, higher date rate, and extended transmit range, compared to a single-hop point-to-point system [41]. The efficiency of a relay system is heavily dependent on the relaying strategies being employed [17]. Among the variety of practical relaying strategies, amplify-forward (AF)

and decode-forward (DF) remain the most popular due to their simplicity and good performance [32]. It is shown that AF can deliver near-optimal performance at low transmit power (at the source node), but its performance at high signal-to-noise ratio (SNR) region is yet to be desired. In comparison, DF is capable of superb performance at high SNRs, but deteriorates sharply as the source-relay channel quality drops below some threshold, causing severe error propagation.

From Information Theory, we know that processing reduces entropy (i.e., $H(x) \geq H(f(x))$, where $H(x)$ stands for the entropy of a random variable x, $f(x)$ is a function of x). In other words, it instructs that an intermediate processor – which, in the context of a relay network, corresponds to an intermediate relay node – should delay making a hard decision (and other forms of quantization) as much as possible, unless it is absolutely sure of the hard decisions (such as being confirmed by the cyclic redundancy check (CRC) code). This naturally leads to the so-called *soft-information relaying* (SIR) [42] [26] [3] [43] [44] [38]. These SIR strategies blends elements from both AF and DF by having the relay(s) decode, extract, and forward soft reliability information instead of hard-quantized bits decisions.

One big aspect of SIR research focuses on ways to *represent* the soft messages. Proposals include pure log-likelihood ratio (LLR) [3], symbol-wise mutual information [43] [44], hyperbolic tangent function [19] [45] [46], and truncated versions of LLRs [38] [47] [31]. A more interesting aspect of SIR is to exploit possible coding gains at the relay-destination transmission (in addition to the coding gains at the source-relay transmission). Unlike the hard-information relaying (HIR) schemes (i.e. conventional decode-forward), where a conventional digital code such as the turbo code [48] [49] and the random sparse-graph code [7] [50] [51], can be directly applied in the second

hop, here in SIR, a "soft-input" code must be employed to effectively protect the soft messages generated by the relay.

This chapter studies new ways of soft message forwarding and protecting for wireless relay networks. We consider the two-hop parallel-relay system. Of particular interest are the questions of what and how: which soft messages carry the best information in the second leg, how to encode and protect them, how to effectively decode them, and what is the performance. In addressing these questions, we succeeded in developing two types of soft-encoding relay (SoER) strategies: the SoER convolutional codes and the SoER turbo codes. Our contributions are summarized below:

- In the choice of soft message representation, we show that the hyperbolic tangent form (tanh) [19], which is similar to the Lambert-W function [26], and which is shown to be SNR-optimal in uncoded relay systems, has several complexity issues that forbid the derivation of the PDFs necessary for ML estimation. Specifically, through the analysis of a general process of soft-encoding, and through the evaluation of the probabilistic distribution of the general soft message (i.e. LLR values), we determine that the *range-limited LLR* (rLLR) can be a simple and effective presentation for the soft message forwarding and protecting in the second hop (and hence the input to the SISO encoder at the relay).

- We propose a distributed SISO convolutional encoder and two effective decoders. The new code is close in spirit to the digital convolutional code and renders a very similar encoding and decoding complexity as its digital counter part, but takes in the rLLRs as the soft input. Both the non-recursive code and the recursive code are investigated. Two modified Viterbi decoders are designed to produce maximum likelihood (ML) decoding results. In the calculation of the branch metrics, the

first decoder applies the Gaussian assumption to the PDF of the rLLRs, while the second carefully characterizes a more accurate PDF of the rLLRs. We show that the Gaussian assumption has sacrificed some 0.5 dB of performance loss (at the frame error rate of 10^{-3} on a block fading channel), in exchange for simplicity. We also show that our SISO convolutional codes can outperform their peer SISO coding schemes in the literature [58]. We attribute the gain to the use of a better representation for soft messages, and the judicious design of the encoding and decoding algorithms to handle them optimally.

- Exploiting the turbo principle, we further propose a distributed SISO turbo encoder employing two SISO recursive systematic convolutional codes. A modified BCJR decoder is developed for the recursive component codes, and an iterative decoder performs the overall soft-output *a posteriori* decoding for the distributed turbo code. This new SISO turbo code can deliver even superb performance, and is almost 1 dB better (at the FER of 10^{-3} on a block fading channel) than the best-known scheme in the literature [6].

Notation: (i) Unless otherwise stated, we use boldface lower-case letters to denote vectors, and use regular letters to denote scalars and random variables. (ii) $\mathcal{N}(m, \sigma^2)$ represents the Gaussian distribution with mean m and variance σ^2. (iii) The subscripts S, R ($\mathsf{R_i}$), and D are used to denote the quantities pertaining to the source, the relay (the ith relay), and the destination, respectively.

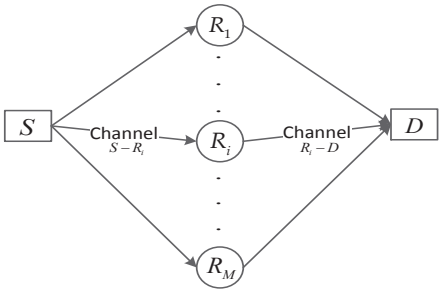


Figure 4.1: A two-hop parallel-relay system

4.2 System Model

We consider a cooperative communication system model shown in Fig. 4.1, where a set of parallel relays (R_i, $i = 1, 2, \cdots, M$) are employed to assist the communication between the source S and destination D. The source and the relay nodes work in a time division manner in accordance with a half-duplex mode. The relays are assumed to participate in every communication session in a collaborative and trustworthy manner. We assume binary phase shift keying (BPSK) is adopted at the source, mapping 0 and 1 to $+1$ and -1. All the channels are quasi-static fading channels, where the fading coefficients are fixed over the course of each communication session, but change independently from session to session. The channel state information (CSI) is known to the receivers in each transmission.

Each communication session consists of two phases. For simplicity, suppose the source is uncoded (the scheme we propose can be applied to channel coded case directly). In phase 1, source S broadcasts information $\mathbf{x}_S = (x_S(1), x_S(2), ...x_S(N))$ with an average energy E_S, and all the relay nodes hear it. Let the signal received at relay

node R_i be $\mathbf{y}_{SR_i}$, which can be expressed by

$$y_{SR_i}(j) = \sqrt{E_S} h_{SR_i} x_S(j) + n_{SR_i}(j), \tag{4.1}$$

at jth communication session, where $i = 1, 2, ..., M$, $j = 1, 2, ..., N$, h_{SR_i} is the fading coefficient of channel S-$\mathsf{R_i}$, and n_{SR_i} is a Gaussian white noise with zero mean and a variance of $\sigma^2_{SR_i}$.

Then, each relay nodes R_i extracts the appropriate soft information $\mathbf{l}_i = (l_i(1), l_i(2), \cdots, l_i(N))$ from the received signal $\mathbf{y}_{SR_i}$, either directly (when source-relay packets are uncoded) or via channel decoding (when source-relay packets are encoded). Then the soft information at each relay node would be fed into a specially-designed rate-1 soft encoder, and each relay node transmits the output $\mathbf{c}_i = (c_i(1), c_i(2), ..., c_i(N))$ to the destination after power normalization, generating a distributed channel code. Let $\mathbf{y}_{R_iD} = (y_{R_iD}(1), y_{R_iD}(2), ..., y_{R_iD}(N))$ denote the corresponding signals the destination receives from these channels. We have:

$$y_{R_iD}(j) = h_{R_iD} \beta_i c_i(j) + n_{R_iD}(j), \tag{4.2}$$

where $j = 1, 2, ..., N$, n_{R_iD} denotes the white noise with Gaussian distribution with mean 0 and variance $\sigma^2_{R_iD}$, and β_i is the normalization factor (introduced by the power amplifier) satisfying

$$E(|\beta_i c_i|^2) = E_{R_i} \tag{4.3}$$

where E_{R_i} is the energy per bit used by the relay R_i.

Finally, after receiving all the signals $\mathbf{y}_{R_iD}$, the destination performs appropriate

soft-input decoding, and makes a hard decision, $\mathbf{x}_D$, for the original source bits $\mathbf{x}_S$.

4.3 Proposed Distributed Soft-Encoding Codes

4.3.1 General Idea of Soft Encoding

In what follows, we will focus on the relay-destination transmission, and especially the soft-message preparation, that is suitable for carry on essential information for the destination and for SISO coding. Before detailing the specific code structure and coding algorithms, we first briefly introduce the fundamental concept of soft encoding.

In the broad sense, SISO (channel) encoding refers to a channel code that takes in real-valued data as the input and produces real-valued codeword at the output. The input data may take values from an arbitrary domain, follow an arbitrary distribution, or have arbitrary meanings. The decoder will take in the (noisy) soft reception, and produce the best estimates of the original soft data or some function of them (such as a two-level quantized version). Performance is usually evaluated through mean square error (MSE), but can be other distortion metrics as well.

Rather than the general-purpose SISO encoding, here we consider a type that is specifically designed for the relay(s) in a two-hop or multi-hop system. The real-valued data at the input to the encoder are some probabilistic form of binary bits, and the decoder is only interested in the accuracy of the binary decisions (as measured by bit error rate or BER) rather than the accuracy of the soft probabilistic data (MSE). The entire code may be regarded as an outgrowth of the conventional linear binary code, where

the binary bits are replaced by their probabilistic values and the binary parity checks are similarly replaced by some appropriate constraints.

Take an $(N, N-1)$ binary single parity check code, for example. As a hard-encoding code, the parity bit p is computed via the binary addition (or exclusive-OR) of the source bits:

$$p = x(1) \oplus x(2) \oplus \cdots \oplus x(N-1). \tag{4.4}$$

The same code, when viewed as an SISO code, possesses the following encoding function:

$$\begin{aligned} &\tanh\left(\frac{1}{2}\log\frac{P(p=0)}{P(p=1)}\right) = \\ &\tanh\left(\frac{1}{2}\log\frac{P(x(1)=0)}{P(x(1)=1)}\right)\tanh\left(\frac{1}{2}\log\frac{P(x(2)=0)}{P(x(2)=1)}\right) \\ &\cdots\tanh\left(\frac{1}{2}\log\frac{P(x(N-1)=0)}{P(x(N-1)=1)}\right), \end{aligned} \tag{4.5}$$

where the logarithm has base e. Note that $\tanh()$ and $\log()$ are both one-to-one functions, and that $P(x(i)=0)$ relates to $P(x(i)=1)$ via $P(x(i)=0)+P(x(i)=1)=1$. Hence, for any soft input that takes the probabilistic form or its equivalence, the soft encoder will be able to generate a probabilistic soft output corresponding to the parity bit.

Since any linear binary code is essentially a collection of single parity check codes operated on different subsets of the source bits, the soft encoding process described in (4.5) therefore generalizes to an arbitrary linear binary code.

4.3.2 Choice of Soft Messages

We now get back to cooperative systems. When the relay can successfully decode and demodulate the data from the source-relay transmission, it can safely forward the correctly re-generated binary data and/or their encoded versions (for a better protection). However, in the case when the CRC does not pass, the relay only has compromised data. To avoid disastrous error propagation, it therefore makes sense for the relay to defer the hard decisions to the destination, by passing along the soft messages (that indicate the reliability of the data) [47].

As discussed in [57], the choice of the soft messages actually makes a difference in terms of end-to-end communication efficiency and reliability. Specifically, it was shown [57] that among a variety of message representations, including the probability of a bit being 0, $P(x=0)$, the likelihood ratio, $\frac{P(x=0)}{P(x=1)}$, the log-likelihood ratio, $log\frac{P(x=0)}{P(x=1)}$, the hyperbolic tangent form, $\tanh(\frac{1}{2}log\frac{P(x=0)}{P(x=1)})$ (equivalent to $P(x=0)-P(x=1)$), and the range-limited LLR (rLLR) l_x, the last stands out as the best-performing choice especially in an uncoded parallel-relay system.

Let $L_x \triangleq \log\frac{P(x=0)}{P(x=1)}$ be the conventional LLR, which is a linear function of x. The range-limited LLR is defined as a truncated version of LLR, and hence takes the form of a 3-segment piece-wise linear function of x:

$$l_x \triangleq \begin{cases} \theta, & \log\frac{P(x=0)}{P(x=1)} \geq \theta, \\ -\theta, & \log\frac{P(x=0)}{P(x=1)} \leq -\theta, \\ \log\frac{P(x=0)}{P(x=1)}, & \text{otherwise}, \end{cases} \tag{4.6}$$

where the positive value θ[1] is the threshold used to truncate the LLR.

The motivation for adopting rLLR l_x instead of the hyperbolic tangent form [58] in our soft-coding system is several-fold. We start by first noting some important facts about the tanh representation: (i) tanh is the full and undistorted representation of the reliability information, and in a single-relay uncoded Gaussian environment, this representation is SNR-optimal (see [19]). (ii) The involvement of channel coding (in a single-relay system) should not change the optimality of tanh in theory, because channel coding in general only acts to enhance the channel (i.e. the combination of the channel coding and the original channel together presents a better "effective" channel). However, the involvement of channel coding does bring in an important practicality issue: the availability of a practical, optimal decoder. Clearly, to derive an optimal/accurate decoder requires the knowledge of the pdf of the received signal. The nonlinearity of tanh, adding to the channel coding operation, makes the pdf of the signal at the final destination intractable. (iii) The optimality of tanh in the single-relay system does not carry automatically to the multi-relay environment. [47] showed, through analysis and simulations, that the tanh representation does not produce the smallest overall error probability at the destination when it comes to multiple relays, and that range-limited LLRs (with optimized thresholds) can do better. From an intuitive perspective, one can expect that the choice of the optimal message representation will in general depend on the number of channel segments and their individual conditions; it is therefore the tanh representation, being only a function of the source-relay segment(s) and not of the relay-destination segment(s), may fall short especially in a multi-relay environment.

Second, by setting an appropriate cap value, the rLLRs can actually approximate the

[1]When $\theta = 0$, i.e. the truncating threshold equals 0, rLLR degenerates to binary hard decision (no longer a soft message relaying strategy).

more important part of the hyperbolic tangent values. As shown in Fig. 4.2, we may roughly divide the hyper-tangent curve into three sections, the two ends with values very close to $+1$ and -1 which represent the very confident estimates, and the middle section that appears to increase (linear-like) with the LLR value. When we limit the LLR values L_x to be within a finite region such as $[-\theta, \theta]$, then $\tanh(L_x/2) \approx l_x/\theta$, meaning that a scaled form of rLLR may be used to approximate $\tanh(L_x/2)$. Fig. 4.2 plots the curves of $\tanh(L_x/2)$ and l_x/θ with different values of θ, as a function of L_x.

Third, as the tanh form naturally lends itself to a simple but meaningful soft-encoding process, the resemblance of rLLR to tanh allows for the adoption of the same simple encoding operation[2]. l_x/θ not only captures the key characteristics of $\tanh(L_x/2)$, but also provides the much-needed simplicity for encoding and especially decoding. With this approximation, the soft-encoding rule in (4.5) can be simplified to:

$$l_p = \theta \frac{l_x(1)}{\theta} \frac{l_x(2)}{\theta} \cdots \frac{l_x(N-1)}{\theta} = \frac{1}{\theta^{N-2}} \prod_{i=1}^{N-1} l_x(i), \tag{4.7}$$

where $l(\cdot)$ represents the rLLR described in (5.28) with a truncating threshold θ. In practice, since all the nodes must satisfy its specific power constraint during transmission (i.e. any symbol that is to be put on the channel will be scaled by the power amplifier), we can thus conveniently drop the scalar in the soft encoding process and simplify (4.7) to

$$l_p = l_x(1) l_x(2) \cdots l_x(N-1), \tag{4.8}$$

[2]It is noted here, in theory, any one-to-one function defined on the input sequence can serve as the encoding process, but in practice, it is rather clueless as what function would lead to a good distance separation in the code space and to the availability of a good-performing decoder with feasible complexity.

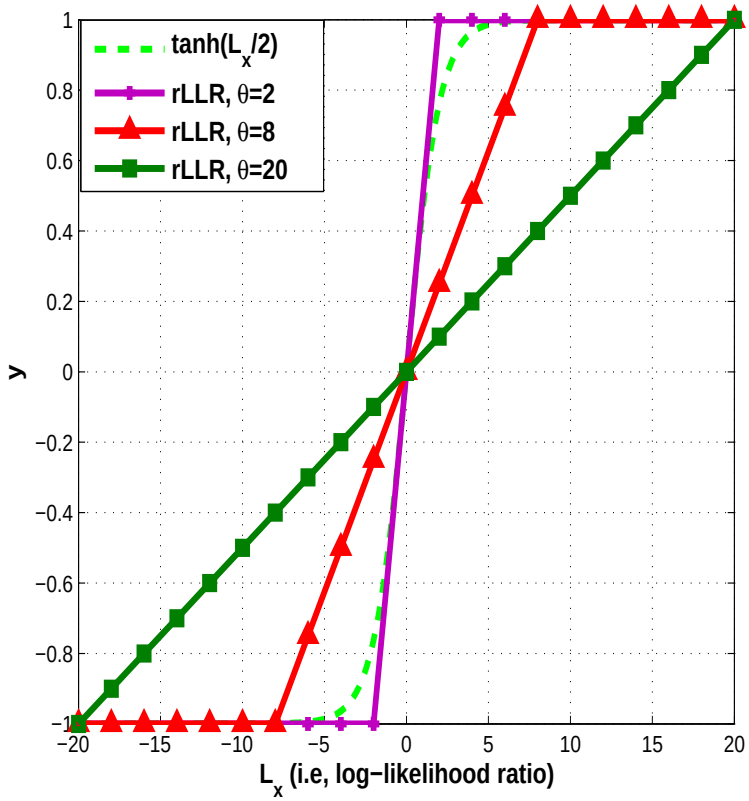


Figure 4.2: Curves of $tanh(L_x/2)$ and rLLR functions with $\theta = 2, 8, 20$.

Further, the rLLR form (with proper choice of the thresholds) actually causes the final signals received at the destination to behave closer to the Gaussian distribution than does the original tanh form (will be discussed in Section. 4.4.2), making the Gaussian-approximated decoder to perform better. The piece-wise linearity of rLLR also makes it possible for us to derive a more accurate PDF and hence an improved decoder, instead of barely using Gaussian approximation, which brings in additional decoding gains. And lastly, rLLRs are numerically more stable.

We now summarize the steps to prepare the soft messages:

(i) The case of uncoded source-relay transmission: The ith relay extracts the LLR's $L_i(j)$ directly from the channel receptions, thereafter referred to as *channel-LLR*, as

follows:

$$L_i(j) = \frac{2h_{SR_i}}{\sigma^2_{SR_i}} y_{SR_i}(j), j = 1, 2, \cdots, N, \tag{4.9}$$

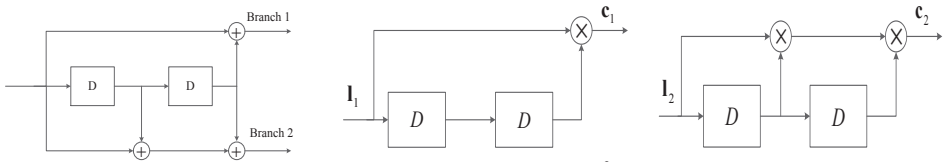
where i is the relay ID and j is the bit index. Since the reception y_{SR_i} follows a Gaussian distribution, so does its LLR value L_i.

(ii) The case when the source-relay transmission is protected by some soft-decodable channel code: the paradigm does not change except that the relay must first (soft-)decode the channel code. In today's systems, pretty much all the error correction codes that are used for the wireless channels are soft-decodable [3] (e.g. convolutional codes, LDPC codes, turbo codes, and turbo product codes), the soft output of the soft-decoder (at the relay), referred to as the *decoder-LLR*, is nothing much different with the *channel-LLR* extracted directly from the channel (i.e. decoder-LLR follows an approximated Gaussian distribution whose variance equals twice the mean for a Gaussian or block fading channel). In other words, the existence of the error correction code at the source only helps make the source-relay channel a better channel with a higher "effective" SNR. The relay, in order to take advantage of the enhanced source-relay channel, must perform soft-decoding of channel code to extract decoder-LLRs, but other than that, the relay may proceed the same way as it treats an uncoded source-relay channel.

Hence, either case, the LLR available at the relay node R_i follows a Gaussian distribution. Generally, we can express this LLR value as

$$L_i(j) = m_i x_S(j) + n_{L_i} \tag{4.10}$$

where m_i represents the mean of the LLR at the ith relay, n_{L_i} denotes a Gaussian noise with distribution $\mathcal{N}(0, \sigma^2_{L_i})$.



(a) The conventional digital $(5,7)_{oct}$ convolutional code.

(b) Rate-1 soft encoding structure for relay R_1.

(c) Rate-1 soft encoding structure for relay R_2

Figure 4.3: Illustration of encoding process

We next apply a truncating threshold θ to these Gaussian-distributed channel-LLR or decoder-LLR values according to (5.28) to obtain rLLR, $l_i(j)$, the form we choose to represent our soft messages.

4.4 SISO Convolutional Codes

4.4.1 SISO Convolutional Encoder

Having prepared the soft messages, the relay nodes will then protect them by feeding them into the SISO encoder to generate encoded soft messages. The fundamental idea of soft encoding has been illuminated in section 4.3.1 and the general expressions are provided in (4.5) and (4.8). Below we present the proposed SISO convolutional code via an illustrating example.

For simplicity, consider a relay system involving two active parallel relays, which collaborate to form a distributed $(5,7)_{oct}$ convolutional code. In the conventional digital coding scheme, each relay will decode and demodulate the source-relay transmission, and, assuming the decoding is all correct, make hard binary decisions, and then feed them to the convolutional code depicted in Fig. 4.3(a), with one relay handles one rate-1 branch respectively. The re-generated and re-encoded data from both relays will be

forwarded to the destination (using BPSK modulation) through their respective channels to form a a rate-1/2 $(5, 7)_{oct}$ digital code. This hard-encoding method is very simple, and works well when source-relay decoding is near perfect (for both relays). Otherwise, there is not only danger for severe error propagation, but also the missed opportunity of weighing the different SNR_{SR_1} and SNR_{SR_2} in the source-relay hop.

The proposed soft-encoding mechanism effectively circumvents these problems by allowing the relays to preserve and protect soft rLLR values, in the case of imperfect first-hop detection/decoding. The soft-encoding process is illustrated in Fig. 4.3(b)(c), where the binary input is replaced by the rLLRs, and the binary addition is replaced by real-valued multiplication. This philosophy works for both the non-recursive codes (feed-forward shift registers) and the recursive codes (feed-backward shift registers). (More discussion of the recursive case can also be found in Section 4.5.)

A quick summary of the soft-encoding steps, including power normalization, is provided in Algorithm 1. We next proceed to the decoder design.

4.4.2 SISO Viterbi Decoder using Gaussian Approximation

ML decoding requires the knowledge of the PDF of the reception at the destination D. We propose two methods to evaluate the PDF of the rLLR at the output of the SISO convolutional code – a simple method with Gaussian approximation (this subsection) and a more sophisticated method that characterizes a more accurate PDF (next subsection) – and subsequently derive the Viterbi decoding algorithm.

Previous studies have established the Gaussian distribution as a convenient and fairly-accurate approximation for the PDF of LLR values. Although rLLR is more like

Algorithm 1: Soft encoding scheme

Input: Range-limited LLR values $l_i(j)$ with cap θ, deduced from the source-relay transmission, where $i = 1, 2$, and $j = 1, 2, \cdots, N$.
Output: Encoded versions of rLLR values.
Initialization: Flush the memory of the convolutional code with θ:

$$l_1(-1) = l_1(0) = l_2(-1) = l_2(0) = \theta. \tag{4.11}$$

Step 1: Soft encoding: As demonstrated in Fig. 4.3(B) and (C), the two relays take in their respective truncated LLRs', feed them into their respective (rate-1) soft encoder, and computes the soft codeword via multiplication:

$$c_1(j) = l_1(j)\, l_1(j-2), \tag{4.12}$$
$$c_2(j) = l_2(j)\, l_2(j-1)\, l_2(j-2), \tag{4.13}$$

where $j = 1, 2, \cdots, N$.
Step 2: Power normalization: Power normalization is perform via eq. (4.3). Then after power scaling $x_{R_i}(j) = \beta_i c_i(j)$, the encoded soft messages are transmitted to destination.

truncated Gaussians, we nevertheless approximate them as some Gaussian with mean μ_i and variance $\sigma_{l_i}^2$.

To ease the illustration, let us take the nonrecursive convolutional code shown in Fig. 4.3 as an example.

Let $\mathbf{l}_i = (l_i(1), l_i(2), ..., l_i(N))$ be the decoder-rLLR's at the ith relay node R_i. We have

$$l_i(j) = \mu_i x_S(j) + n_{l_i} \tag{4.14}$$

where $\mu_i = \frac{\sum_{j=1}^{N} l_i(j) x_S(j)}{N}$ represents the mean value of the decoder-LLRs, and n_{l_i} represents a Gaussian noise with mean zero and variance $\sigma_{l_i}^2 = \frac{\sum_{j=1}^{N} (l_i(j) - \mu_i x_S(j))^2}{N}$.

The soft codewords can be expressed as:

$$c_1(j) =(\mu_1 x_S(j)+n_{l_1}(j))(\mu_1 x_S(j–2) + n_{l_1}(j–2))=\mu_1^2 x_S(j)x_S(j–2)+\tilde{n}_{l_1}=\mu_1^2 x_{c_1}(j) + \tilde{n}_{l_1}, \tag{4.15}$$

$$\begin{aligned} c_2(j) &= (\mu_2 x_S(j) + n_{l_2}(j))(\mu_2 x_S(j-1) + n_{l_2}(j-1))(\mu_2 x_S(j-2) + n_{l_2}(j-2)) \\ &= \mu_2^3 x_S(j)x_S(j-1)x_S(j-2) + \tilde{n}_{l_2} = \mu_2^3 x_{c_2}(j) + \tilde{n}_{l_2}. \end{aligned} \tag{4.16}$$

where $\tilde{n}_{l_1} \sim \mathcal{N}(0, \left(\mu_1^2 + \sigma_{l_1}^2\right)^2 - \mu_1^4)$, and $\tilde{n}_{l_2} \sim \mathcal{N}(0, \left(\mu_2^2 + \sigma_{l_2}^2\right)^3 - \mu_2^6)$, assuming that the virtual noise at each time instant is uncorrelated. $x_{c_i}(j)$ is the digital codeword corresponding to soft codeword $c_i(j)$.

The destination gathers the receptions from all the active relays and performs joint decoding. We take the same $(5, 7)_{oct}$ distributed convolutional code as an example, and explain how ML-optimal Viterbi decoding can be efficiently achieved.

As discussed previously, each of the two active relays transmits $\beta_i c_i(j)$. The destination receives

$$y_{R_iD}(j) = h_{R_iD}\beta_i c_i(j) + n_{R_iD}, \;\; i = 1, 2. \tag{4.17}$$

Substituting (4.16) into (4.17), we rewrite the signals received at the destination as

$$y_{R_1D}(j) =h_{R_1D}\beta_1 \left(\mu_1^2 x_S(j)x_S(j-2) + \tilde{n}_{l_1}\right)+n_{R_1D}=h_{R_1D}\beta_1\mu_1^2 x_{c_1}(j)+h_{R_1D}\beta_1\tilde{n}_{l_1}+n_{R_1D}, \tag{4.18}$$

$$y_{R_2D}(j) =h_{R_2D}\beta_2 \left(\mu_2^3 x_S(j)x_S(j-1)x_S(j-2)+\tilde{n}_{l_2}\right)+n_{R_2D}=h_{R_2D}\beta_2\mu_2^3 x_{c_2}(j)+h_{R_2D}\beta_2\tilde{n}_2+n_{R_2D}. \tag{4.19}$$

We now revisit the legacy Viterbi algorithm, and make necessary modifications to make it work properly for our SISO code. First, evaluate the PDF of the signals received at the destination:

$$f(y_{R_iD}(j)|x_c(j)=\pm 1)=\frac{1}{\sqrt{2\pi}\tilde{\sigma}_{R_iD}}e^{-\frac{\left(y_{R_iD}(j)\mp\mu_{R_i}\right)^2}{2\tilde{\sigma}^2_{R_iD}}}, \tag{4.20}$$

where $\mu_{R_1}=h_{R_1D}\beta_1\mu_1^2x_{c_1}(j)$, $\tilde{\sigma}^2_{R_1D}=h^2_{R_1D}\beta_1^2\left(\left(\mu_1^2+\sigma_1^2\right)^2-\mu_1^4\right)+\sigma^2_{R_1D}$, $\mu_{R_2}=h_{R_2D}\beta_2\mu_2^3x_{c_2}(j)$,
$\tilde{\sigma}^2_{R_2D}=h^2_{R_2D}\beta_2^2\left(\left(\mu_2^2+\sigma_2^2\right)^3-\mu_2^6\right)+\sigma^2_{R_2D}$.

The branch metric $m(j)$ in the Viterbi algorithm should be adjusted to

$$m(j)=\frac{\left(y_{R_1D}(j)-h_{R_1D}\beta_1\mu_1^2x_{c_1}(j)\right)^2}{\tilde{\sigma}^2_{R_1D}}+\frac{\left(y_{R_2D}(j)-h_{R_2D}\beta_2\mu_2^3x_{c_2}(j)\right)^2}{\tilde{\sigma}^2_{R_2D}}. \tag{4.21}$$

The path metric is calculated by summing together (i.e. real-value addition) all the brach metrics along the way. The algorithm is otherwise the same as the conventional Viterbi algorithm for digital codes.

For a general code $g=(g_1,...,g_M)$ with $g_i=\sum_{k=0}^{K-1}b_{i,k}D^k+D^K$, $b_{i,k}\in\{0,1\}$, $i=1,2,...M$ the branch metric of the Viterbi decoder can be computed as follows:

Theorem 1. *Let* $w_i=\sum_{k=0}^{K-1}b_{i,k}+1$ *be the weight of the generator polynomial of the rate-1 convolutional code employed by the relay node* $\mathsf{R_i}$. *The branch metric at time* j *can be computed by*

$$m(j) = \sum_{i=1}^{M} \frac{\left(y_{R_iD}(j) - h_{R_iD}\beta_i \mu_i^{w_i} x_{c_i}(j)\right)^2}{\tilde{\sigma}^2_{R_iD}}, \tag{4.22}$$

where $\tilde{\sigma}^2_{R_iD} = h^2_{R_iD}\beta_i^2((\mu_i^2 + \sigma_i^2)^{w_i} - \mu_i^{2w_i}) + \sigma^2_{R_iD}$.

Proof. The proof follows directly from (4.16) and (4.21). □

It is known that the PDF of the noise at the destination does not follow the exact Gaussian distribution. Fig. 4 shows the PDF of the received signal with different θ's at the destination from the relay R_i, assuming information bit $+1$ is transmitted. The curves are obtained by averaging over 10000 frames, each consisting of 1500 symbols. As indicated by the figure, with threshold 8 the actual distribution gets a closer resemblance to the Gaussian distribution than the others[3].

A short summary of the ML Viterbi decoding process using the Gaussian approximation goes in ML decoding algorithm I.

4.4.3 SISO Viterbi Decoder Using A More Accurate PDF

The previous subsection presents the Viterbi decoder with the Gaussian approximation. Here we try to avoid the Gaussian approximation by evaluating a more accurate PDF of rLLR values. To ease illustration, we take the first branch in Fig. 4.3 as an illustrating example. For simplicity, the relay node index i is conveniently dropped for the example illustration. To start, note that the input to the SISO decoder, channel-rLLR

[3]The PDF curves of the actual received signal and the distribution calculated from Gaussian approximation in tanh scheme are almost overlapped with our case with threshold set to 2.

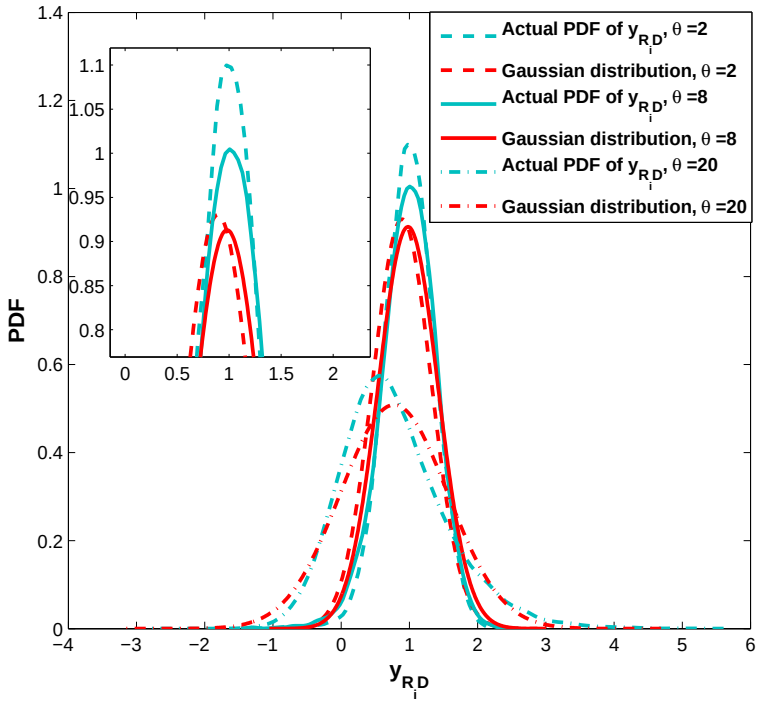


Figure 4.4: PDF of y_{R_iD} with comparison to Gaussian distribution, $SNR_{SR_i} = SNR_{R_iD} = 6$ (dB), $\theta = 2, 8, 20$, $g = (111)$. Dashed lines correspond to $\theta = 2$, solid lines correspond to $\theta = 8$, dot-dash lines correspond to $\theta = 20$. In each case, dark red line depicts the PDF using Gaussian approximation, and light blue line denotes the actual PDF of y_{R_iD}.

or decoder-rLLR from the source-relay transmission, follows a truncated Gaussian distribution:

$$f(l(j)|x_S(j) = \pm 1) = \delta(l(j) - \theta)Q(\frac{\theta \mp m}{\sigma_L}) + \frac{1}{\sqrt{2\pi}\sigma_L} e^{\frac{-(l(j)\mp m)^2}{2\sigma_L^2}} + \delta(l(j) + \theta)Q(\frac{\theta \pm m}{\sigma_L}), \tag{4.23}$$

where $-\theta \leq l(j) \leq \theta$, $j = 1, 2, ..., N$, $\delta(x)$ is Dirac delta function of x, and $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{(-\frac{u^2}{2})} dx$.

We first derive the PDF of the soft codeword $c(j)$ in R_1, where $c(j) = l(j)l(j-2)$ (which corresponds to $x_c(j) = x_S(j)x_S(j-2)$, where $x_c \in \{+1, -1\}$ and $x_S \in \{+1, -1\}$). The conditional PDF of $c(j)$ can be expressed by $f(c(j)|x_c(j)) = f(l(j)l(j-$

Algorithm 2: ML decoding algorithm I.

Input: Reception from relay-destination transmission, (4.17).
Output: Binary decisions of the original source data $\mathbf{x_S}$.
Initialization:
A 4-state trellis corresponding to the $(5,7)_{oct}$ convolutional code is set up.
Each branch is marked with a binary input bit $x_S(j) = +1$ or $x_S(j) = -1$, and two output signals $x_S(j)x_S(j-2)$ and $x_S(j)x_S(j-1)x_S(j-2)$.
All the state metrics are reset to zero.
Step 1: Trellis Decoding:
The decoder proceeds through the trellis by computing each branch metric using (4.41),
The decoder accumulate branch metric along the (survival) paths to generate the state metrics for all the states from 1 to N.
The survival path leading into any state is the one that provides the smaller cumulative metric so far, and the other competing path with a larger cumulative metric is eliminated (random choice in case of a tie).
Step 2: Tracing back and sequence detection:
After all the state metrics are computed, the state at time N with the smallest state metric is declared as the final survivor,
The binary input bits corresponding to this survival path are declared as the decoding decision.

$2)|x_S(j)x_S(j-2))$. For simplicity of exposition, let us use a_1 and a_2 to denote $l(j)$ and $l(j-2)$, and use b_1 and b_2 to denote $x_S(j)$ and $x_S(j-2)$. Thus we have $f(c(j)|x_c(j)) = f(a_1a_2|b_1b_2)$. When $x_c = 1$, we have $b_1 = b_2 = 1$ or $b_1 = b_2 = -1$. Since $l(j)$'s $(j = 1, 2, 3...N)$ are independent and identically distributed (i.i.d) with the PDF expressed as (4.23), we get

$$f(a_1, a_2|b_1 = \pm 1, b_2 = \pm 1) = f(a_1|b_1 = \pm 1)f(a_2|b_2 = \pm 1). \tag{4.24}$$

Since $\theta > 0$, $m > 0$, and σ_L is rather small when the channel SNR is large, we have

$$Q\left(\frac{\theta - m}{\sigma_L}\right) \gg Q\left(\frac{\theta + m}{\sigma_L}\right). \tag{4.25}$$

We can then simplify (4.23) to ($i=1,2$):

$$f(a_i|b_i=\pm 1) \approx \delta(a_i \mp \theta) Q\left(\frac{\theta - m}{\sigma_L}\right) + \frac{1}{\sqrt{2\pi}\sigma_L} e^{\frac{-(a_i \mp m)^2}{2\sigma_L^2}}, \tag{4.26}$$

Substituting (4.26) into (4.24), we obtain

$$\begin{aligned}
&f(a_1, a_2|b_1=+1, b_2=+1) = f(a_1|b_1=1) f(a_2|b_2=1) \\
&= \delta(a_1-\theta)\delta(a_2-\theta)\alpha^2 + \alpha\delta(a_1-\theta)N(a_2; m, \sigma_L) \\
&+\alpha\delta(a_2-\theta)N(a_1; m, \sigma_L) + N(a_2; m, \sigma_L)N(a_1; m, \sigma_1), \\
&f(a_1, a_2|b_1=-1, b_2=-1) = f(a_1|b_1=-1) f(a_2|b_2=-1) \\
&= \delta(a_1+\theta)\delta(a_2+\theta)\alpha^2 + \alpha\delta(a_1+\theta)N(a_2; -m, \sigma_L) \\
&+\alpha\delta(a_2+\theta)N(a_1; -m, \sigma_L) + N(a_2; -m, \sigma_L)N(a_1; -m, \sigma_L).
\end{aligned} \tag{4.27}$$

where $\alpha = Q\left(\frac{\theta - m}{\sigma_L}\right)$, and $N(a_i; m, \sigma_L)$ is the PDF of variable a_i, which is a Gaussian distribution with mean m and variance σ_L^2.

Thus we can calculate $f(c|x_c = 1)$ by (the time index j in $x_c(j)$ is dropped for simplicity):

$$\begin{aligned}
f(c|x_c=1) &= \int_{-\theta}^{+\theta} f\Big(a_1, \frac{c}{a_1}\Big|x_c=1\Big)\frac{1}{|a_1|}da_1 = \int_{-\theta}^{+\theta} f\Big(a_1, \frac{c}{a_1}\Big|x_c=-1\Big)\frac{1}{|a_1|}da_1, \\
&= \delta(c-\theta^2)\alpha^2 + \alpha\Big(\frac{1}{\sqrt{2\pi}\sigma_L} e^{\frac{-(\frac{c}{\theta}-m)^2}{2\sigma_L^2}}\Big)\big(\frac{1}{\theta} + \frac{1}{\theta}\big) + \varepsilon, \\
&\approx \delta(c-\theta^2)\alpha^2 + \alpha\Big(\frac{2}{\sqrt{2\pi}\sigma_L\theta} e^{\frac{-(\frac{c}{\theta}-m)^2}{2\sigma_L^2}}\Big) \\
&= \delta(c-\theta^2)\alpha^2 + 2\alpha N(c; \theta m, \theta\sigma_L),
\end{aligned} \tag{4.28}$$

where ε follows the "product-normal"-like distribution [59] defined by

$$\varepsilon = \int_{-\theta}^{+\theta} N\big(\frac{c}{a_1}; m, \sigma_L\big) N(a_1; m, \sigma_L) \frac{1}{|a_1|} da_1. \tag{4.29}$$

With an appropriate choice for the value of θ, ε becomes fairly small compared to the other terms in (4.28) especially when the S-R SNR is high (see Fig. 4.5 for a visual verification). Hence we have safely ignored the value of ϵ in (4.28).

Similarly, when $x_c = -1$, we have

$$f(c|x_c = -1) \approx \delta(c + \theta^2)\alpha^2 + 2\alpha N(c; -\theta m, \theta\sigma_L). \tag{4.30}$$

In the more general case, let us use c^w to denote the product (i.e. the codeword bit) of w soft rLLR inputs with the PDF expressed in (4.23), i.e.

$$c^w = a_1 a_2 ... a_w, \tag{4.31}$$

where w is the weight of the generator polynomial at a relay node, and $-\theta^w \leq c^w \leq \theta^w$. We have the following result about the PDF of c^w.

Theorem 2. *The PDF of c^w conditioned on x_c takes the form of*

$$f(c^w|x_c{=}{\pm}1){\approx}\delta(c^w{\mp}\theta^w)\alpha^w{+}w\alpha^{w-1}N(c^w; {\pm}\theta^{w-1}m, \theta^{w-1}\sigma_L). \tag{4.32}$$

Proof. (Proof by induction.) When $w = 2$, (4.28) and (4.30) apparently satisfy Theorem

2.

Suppose Theorem 2 holds for $w=i$. Consider $w = i + 1$, we have

$$\begin{aligned} f(c^{i+1}|x_c = +1) = f(c^i a_{i+1}|x_c = +1) &= \int_{-\theta}^{+\theta} f(a_{i+1}, \frac{c^{i+1}}{a_{i+1}}|x_c = +1)\frac{1}{|a_{i+1}|}da_{i+1}, \\ &= \int_{-\theta}^{+\theta} f(a_{i+1}|x_c = +1)f(\frac{c^{i+1}}{a_{i+1}}|x_c = +1)\frac{1}{|a_{i+1}|}da_{i+1}, \\ &\approx \delta(c^{i+1} - \theta^{i+1})\alpha^{i+1} + \alpha^i(i+1)N(c^{i+1}; \theta^i m, \theta^i \sigma_L). \end{aligned} \tag{4.33}$$

This concludes: $f(c^w|x_c{=}{+}1){\approx}\delta(c^w{-}\theta^w)\alpha^w{+}w\alpha^{w-1}N(c^w; \theta^{w-1}m, \theta^{w-1}\sigma_L)$. Similarly proof goes to$f(c^w|x_c = -1) \approx \delta(c^w + \theta^w)\alpha^w + w\alpha^{w-1}N(c^w; -\theta^{w-1}m, \theta^{w-1}\sigma_L)$. □

With the above theoretical results in hand, we can easily calculate the PDF of the reception from each relay-destination branch. Suppose the destination D receives signal $y_{R_iD} = h_{R_iD}\beta_i c_i + n_{R_iD}$ from the relay R_i. Let $\tilde{\beta}_i \triangleq h_{R_iD}\beta_i$, we can rewrite y_{R_iD} as

$$y_{R_iD} = \tilde{\beta}_i c_i + n_{R_iD}, \tag{4.34}$$

whose conditional PDF becomes:

$$f(y_{R_1D}|x_c = \pm 1) = \int_{-\theta^2}^{\theta^2} f(c_1|x_c = \pm 1)f(n_{R_iD})dc_1, \tag{4.35}$$

Substituting (4.28) into (4.35), we obtain:

$$f(y_{R_1D}|x_c=\pm 1)=\alpha^2 N(y_{R_1D};\pm\tilde{\beta}_1\theta^2,\sigma_{R_1D}) \\ +2\alpha N(y_{R_1D};\pm\tilde{\beta}_1 m_1\theta,A)\left(Q\Big(\frac{B_1^{\pm}-C}{\sigma_{L_1}\sigma_{R_1D}A}\Big)-Q\Big(\frac{B_1^{\pm}+C}{\sigma_{L_1}\sigma_{R_1D}A}\Big)\right), \tag{4.36}$$

where $A=\sqrt{\sigma_{L_1}^2\tilde{\beta}_1^2\theta^2+\sigma_{R_1D}^2}$, $B_1^{+}=-\sigma_{L_1}^2\tilde{\beta}_1\theta y_{R_1D}-\sigma_{R_1D}^2 m_1$, $B_1^{-}=\sigma_{L_1}^2\tilde{\beta}_1\theta y_{R_1D}-\sigma_{R_1D}^2 m_1$, $C=\sigma_{L_1}^2\tilde{\beta}_1^2\theta^3+\sigma_{R_1D}^2\theta$.

Theorem 3. *Suppose the relay node* R_i *employs a (rate-1) SISO nonrecursive convolutional code described in Section 4.4.1. Let* w *be the weight of the generator polynomial of this SISO convolutional code. The (conditional) PDF of the soft message* y_{R_iD} *it forwards to the destination can be (approximately) given by (4.37).*

$$f(y_{R_iD}|x_c=\pm 1)=\alpha^w N(y_{R_iD};\pm\tilde{\beta}_i\theta^w,\sigma_{R_iD}) \\ +\alpha^{w-1}wN(y_{R_iD};\pm\tilde{\beta}_i m_i\theta^{w-1},A)\left(Q\Big(\frac{B_i^{\pm}-C}{\sigma_{L_i}\sigma_{R_iD}A}\Big)-Q\Big(\frac{B_i^{\pm}+C}{\sigma_{L_i}\sigma_{R_iD}A}\Big)\right). \tag{4.37}$$

where $A=\sqrt{\sigma_{L_i}^2\tilde{\beta}_i^2\theta^{2(w-1)}+\sigma_{R_iD}^2}$, $B_i^{+}=-\sigma_{L_i}^2\tilde{\beta}_i\theta^{w-1}y_{R_iD}-\sigma_{R_iD}^2 m_i$, $B_i^{-}=\sigma_{L_i}^2\tilde{\beta}_i\theta^{w-1}y_{R_iD}-\sigma_{R_iD}^2 m_i$, $C=\sigma_{L_i}^2\tilde{\beta}_i^2\theta^{(2w-1)}+\sigma_{R_iD}^2\theta$.

Proof. The proof for this general PDF result is straightforward, which follows the same line of derivation as we did for the illustration example. It is therefore omitted. □

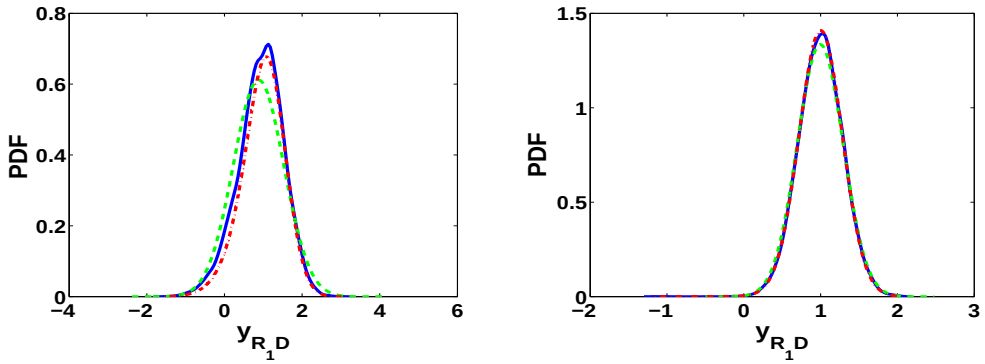
We note that Theorem 3 does not force Gaussian assumption to the PDF of the soft

messages. Nevertheless, it has dropped a couple of very small terms in the calculation of the true PDF. To see the accuracy of the derived PDF, we compare the theoretical PDF of $f(y_{R_iD}|x_c)$ given in (4.37), the PDF calculated from the Gaussian approximation and the histogram of $f(y_{R_iD}|x_c = +1)$ we have collected using the Monte Carlo simulations ($\theta = 8$ is used as the truncating threshold.) in Fig. 4.5. It can be observed that the PDF calculated from the more accurate approach bridges the gap between the actual PDF (histogram of the received signals) and the Gaussian approximated PDF. Our theoretical PDF expression in (4.37) provides a rather precise (and simple) characterization of the true PDF. When the channel SNR increases, the theoretical PDF with the threshold 8 gets even more accurate.

Given the fairly accurate PDF of the receptions in (4.37), the branch metric for the Viterbi decoder, which involves M relay branches, can then be calculated by

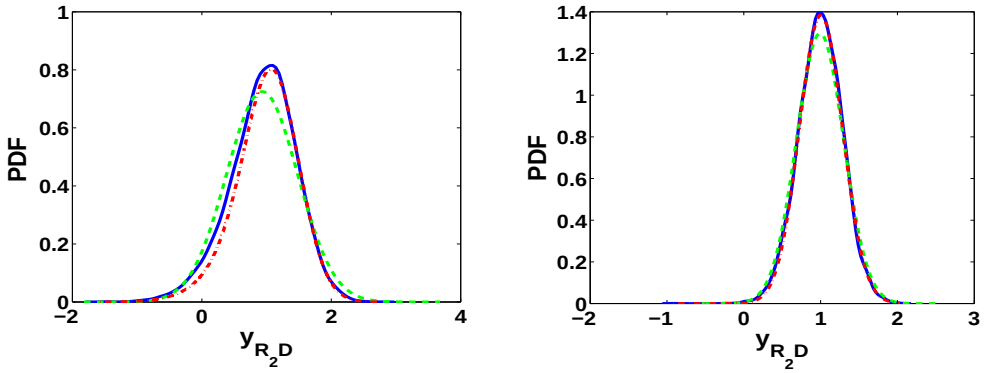
$$log\left(\prod_{i=1}^{M} f(y_{R_iD}|x_c = \pm 1)\right), \tag{4.38}$$

at each stage of the trellis. Here the logarithm function is applied to provide a good numerical stability in the calculation. The path metric at each stage is calculated by accumulating all the branch metrics along the way, i.e. by adding the new branch metric to the (survival) path metric of the previous stage. The survival path is the one with the biggest path metric. The rest of the decoding process remains the same as the usual Viterbi algorithm.



(a) $SNR_{SR} = SNR_{RD} = 4$ (dB), $\theta = 8$, $g_1 =$ (101)

(b) $SNR_{SR} = SNR_{RD} = 8$ (dB), $\theta = 8$, $g_1 =$ (101)



(c) $SNR_{SR} = SNR_{RD} = 5$ (dB), $\theta = 8$, $g_2 =$ (111)

(d) $SNR_{SR} = SNR_{RD} = 8$ (dB), $\theta = 8$, $g_2 =$ (111)

Figure 4.5: PDF of y_{R_1D} and y_{R_2D} with comparison to the Gaussian distribution and the theoretical PDF in (37).

4.5 SISO Turbo Codes

Digital convolutional codes include both feed-forward (i.e. non-recursive) and feed-backward (i.e. recursive) forms, with the former taking the flavor of finite impulse response (FIR) filters and the latter infinite impulse response (IIR) filters. When two digital recursive convolutional codes are used in parallel concatenation, and combined with an appropriate interleaver in-between, a powerful digital turbo code results. Turbo code is known to achieve remarkable performance close to the Shannon limit, and the gain is attributed to the fact that the two branches provide complementary support for each other, namely, when one recursive component codes produces a low-weight (i.e. weak) sub-codeword, the other will, with a high probability, produce a high-weight (i.e. strong) sub-codeword, thus strengthening the entire codeword.

The clever idea of applying a distributed digital turbo code to a parallel-relay decode-forward system was proposed by Valenti and Cheng in [48]. In the context of soft information relaying, [6] was the first to construct a SISO turbo code (based on the the a posteriori probability of the information symbols, which is equivalent to the tanh representation [57] of the soft message), and to apply it to the relay system. The research in the field of SISO turbo codes has not evolved much from that. In what follows, we will extend the SISO nonrecursive convolution codes discussed previously to the recursive case, derive the corresponding BCJR decoding algorithm, and develop a new (distributed) SISO turbo code for use in parallel-relay systems.

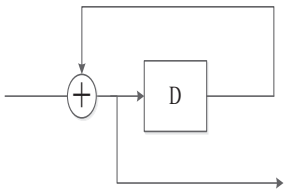
4.5.1 Recursive SISO Convolutional Codes and Distributed SISO Turbo Codes

Again, we assume that the relays will use rLLR to represent the soft messages. As shown in Fig. 4.6, a digital recursive convolutional code involves nothing more than parity check operations. Previous discussion already showed that, corresponding to the binary parity check operation in (4.4), is the SISO operation in (4.5), which is the general form, and (4.8), when rLLR is used to approximate the hyperbolic tangent value.

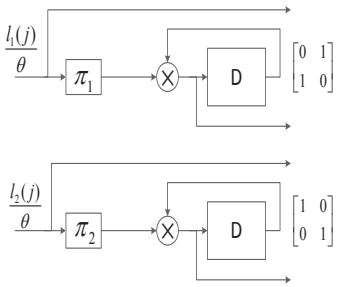
It then follows that a distributed SISO turbo code can be constructed to assist the soft information relaying. Fig. 4.6 depicts a two-relay case. Each relay proceeds with the following operations: First, demodulate and (soft) decode what it receives from the source-relay transmission, to get the LLR values L_x for each source bit x; Next truncate L_x with the pre-determined threshold θ to get rLLR l_x; Then, interleave the sequence of $\frac{l_x}{\theta}$'s with its distinctive interleaver, and pass the scrambled sequence to the SISO recursive convolutional encoder to get the soft coded bits; Finally, perform proper puncturing (if necessary) and power normalization, and send the data over to the destination. In general, different relays can use the same recursive code , but the interleavers they use must be different from each other. Algebraic interleaves, random interleaves and better yet, S-random interleavers, are all good choices.

4.5.2 BCJR Convolutional Decoder and Iterative Turbo Decoder

We can apply the proposed soft coding method directly to recursive convolutional codes, thus generating distributed Turbo code. From discussion in section. 4.3.1, the soft



(a) The digital encoder of $(1, 1/1 + D)$ convolutional code.



(b) Component Turbo encoders using $(1, 1/1+D)$ convolutional code.

Figure 4.6: Illustration of encoding process

information LLR of each binary addition of two bits can be roughly approximated by the product of their LLR's. Take recursive convolutional code $(1, 1/1 + D)$ as example, shown in fig. 4.6. Fig. 4.6(A) show the digital coding structure for $(1, 1/1 + D)$ convolutional code; (B) is its corresponding soft encoder.

We use $m(j-1)$ denote the information stored in the register in time unit $j-1$, $j = 1, 1, ...N$. $m(0)$ is initialed to be cap θ. The output codeword at time instant j is

$$c(j) = \frac{l(j)}{\theta} m(j-1), m(j-1) = m(j-2)\frac{l(j-1)}{\theta}. \tag{4.39}$$

By iterative calculation, we can further express the output codeword at time instant j as

$$c(j) = \prod_{k=1}^{j} \frac{l(k)}{\theta}. \tag{4.40}$$

Consider soft turbo code generated by $(1, 1/1 + D)$. Suppose two relay nodes are involved in the system. One relay node would soft encode the range-limited LLR's from the source using coding scheme proposed above; the other would encode the interleaved

version of the range-limited LLR's. At the receiver side, we use Gaussian estimation described in section. 4.4.2 and BCJR algorithm to decode the distributed Turbo code. Since the mean and variance of $c(j)$ is time-variant, the calculation of γ in BCJR algorithm [60] would change accordingly at each time unit, which can be expressed as

$$\gamma(j) = \exp\left(-\frac{\left(y_{R_iD}(j) - h_{R_iD}\beta_i(\frac{\mu_i}{\theta})^j x_{c_i}(j)\right)^2}{\tilde{\sigma}^2_{R_iD}(j)}\right), \tag{4.41}$$

where $\tilde{\sigma}^2_{R_iD}(j) = h^2_{R_iD}\beta_i^2(((\frac{\mu_i}{\theta})^2 + (\frac{\sigma_i}{\theta})^2)^j - (\frac{\mu_i}{\theta})^{2j}) + \sigma^2_{R_iD}$, i is the relay index, j is the time index.

4.6 Analysis

4.6.1 Diversity Order Analysis

We now discuss the theoretical error probability of the proposed coding scheme. Let d_{free} be the free distance (minimum distance) of the proposed soft-encoding convolutional code, and let $B_{d_{free}}$ be its multiplicity, i.e. $B_{d_{free}}$ denotes the total number of valid codewords having weight d_{free}. At the high SNR region of the block fading channel, the (instantaneous) codeword error [58] can be approximated by

$$P_e \approx B_{d_{free}} Q\left(\sqrt{2\sum_{i=1}^{M} w_i\gamma_i}\right) \approx 0.5 B_{d_{free}} \exp(-\sum_{i=1}^{M} w_i\gamma_i), \tag{4.42}$$

where w_i is the weight of the generator polynomial (of the convolutional code) at Relay R_i, and γ_i denotes the equivalent SNR of the 2-hop channel S-R_i-D.

From the previous discussion in Section 4.4.2, we see that γ_i can be computed by

$$\gamma_i = \frac{h_{R_iD}^2 \beta_i^2 \mu_i^{2w}}{\sigma_{R_iD}^2 + h_{R_iD}^2 \beta_i^2 ((\mu_i^2 + \sigma_{l_i}^2)^w - \mu_i^{2w})}. \tag{4.43}$$

Define $\gamma_{in,i} \triangleq \frac{\mu_i^2}{\sigma_{l_i}^2}$, and $\gamma_{R_iD} \triangleq \frac{h_{R_iD}^2 E_{R_i}}{\sigma_{R_iD}^2} = \frac{h_{R_iD}^2 \beta_i^2 (\mu_i^2 + \sigma_{l_i}^2)^w}{\sigma_{R_iD}^2}$. Combining (4.43) and (4.42), we can simplify the error probability:

$$P_e \approx 0.5 B_{d_{free}} \exp\left(- \sum_{i=1}^{M} \frac{w_i \gamma_{R_iD} \gamma_{in,i}}{w_i \gamma_{R_iD} + \gamma_{in,i}} \right), \tag{4.44}$$

where $\gamma_{in,i}$ is given in (4.45).

$$\gamma_{in,i} = \frac{\mu_i^2}{\sigma_{l_i}^2} = \frac{\mu_i^2}{E[l_i^2] - \mu_i^2} = \frac{1}{\frac{E[l_i^2]}{\mu_i^2} - 1} = \frac{1}{\frac{\theta_i^2 (Q(\frac{\theta_i - m_i}{\sigma_{L_i}}) + Q(\frac{\theta_i + m_i}{\sigma_{L_i}})) + \int_{-\theta_i}^{+\theta_i} l_i^2 N(l_i; m_i, \sigma_{L_i}) dl_i}{(\theta_i (Q(\frac{\theta_i - m_i}{\sigma_{L_i}}) - Q(\frac{\theta_i + m_i}{\sigma_{L_i}})) + \int_{-\theta_i}^{+\theta_i} l_i N(l_i; m_i, \sigma_{L_i}) dl_i)^2} - 1}. \tag{4.45}$$

From both theoretical and experimental aspects (in Section 5.7), the overall performance is directly affected by the threshold selected at each relay node. For example, if we choose $\theta = 0$, then the SoER is essentially a digital distributed codes. However, it is difficult to optimize this value with channel coding involved. From (4.45), for any given threshold θ_i, $\gamma_{in,i}$ goes to infinity with the increase of γ_{SR_i}. Then (4.44) can be further simplified to

$$P_e \approx 0.5 B_{d_{free}} \exp\left(- \sum_{i=1}^{M} w_i \gamma_{R_iD} \right). \tag{4.46}$$

Without loss of generality, we assume $E_{R_i} = 1$, and $\sigma_{R_iD}^2 = \sigma^2$, where $i =$

$1, 2, ..., M$. The average error probability over the Rayleigh fading channel is calculated by

$$\begin{aligned}
\bar{P}_e &\approx 0.5B_{d_{free}} \prod_{i=1}^{M} \int_0^{+\infty} \exp(-w_i \frac{h_{R_iD}^2}{\sigma^2}) f_{h_{R_iD}}(h_{R_iD}) dh_{R_iD} \\
&= 0.5B_{d_{free}} \prod_{i=1}^{M} \int_0^{+\infty} \exp(-w_i \frac{h_{R_iD}^2}{\sigma^2}) 2h_{R_iD} \exp(-h_{R_iD}^2) dh_{R_iD} \\
&< 0.5B_{d_{free}} \prod_{i=1}^{M} \int_0^{+\infty} \exp(-\frac{h_{R_iD}^2}{\sigma^2}) 2h_{R_iD} \exp(-h_{R_iD}^2) dh_{R_iD} \\
&= 0.5B_{d_{free}} \prod_{i=1}^{M} \frac{1}{\left(\frac{1}{\sigma^2} + 1\right)}.
\end{aligned} \tag{4.47}$$

We can then compute the diversity order of the proposed scheme, D, as

$$D \triangleq -\lim_{\sigma^2 \to 0} \frac{\log \bar{P}_e}{\log \frac{1}{\sigma^2}} = M, \tag{4.48}$$

which theoretically proves that the proposed scheme can achieve the full diversity order.

4.6.2 Code Selection for Feed-forward Soft-Encoding

Just as in the digital systems, the choice of the base convolutional codes makes a difference in the system performance. There has been extensive research, based either on free-distance analysis or union bounds or computer-assisted exhaustive search, on the best generator polynomials for digital convolutional codes. From (4.46), we see that the proposed soft-encoding scheme is much like an analog extension of the digital coding schemes; thus, we can leverage the results developed in the digital field to assist our code selection in the analog field. These well-established rules include: increas-

ing the memory size of the code to enhance the performance, selecting codes that are non-catastrophic, and that the good generator polynomials in the digital domain tend to perform well in the analog domain also (These hypotheses are consistent with the simulation results shown in Section. 5.7). To further verify these hypotheses, we also evaluate all the rate-1/2 feed-forward (i.e. non-recursive) convolutional codes with a memory size of up to 2. The component codes are selected from the following set of generator polynomials 1, $1+D$, $1+D^2$ and $1+D+D^2$, and all possible combinations (of two branches) are evaluated through Monte Carlo simulations. As expected, $(1+D^2, 1+D+D^2)$ (i.e. $(5,7)_{oct}$), the best digital code, remain the best of the soft-encoding codes. They result in the choices of the constituent codes for our SISO encoder. A related issue is the judicious assignment of constituent codes to different relays to achieve a lower error rate. The overall error probability P_e of the proposed system can be evaluated by $P_e \approx 0.5 B_{d_{free}} \exp\left(-\sum_{i=1}^{M} \frac{1}{w_i^{-1}\gamma_{R_iD}^{-1}+\gamma_{in,i}^{-1}}\right)$. According to Lemma 2 in [58], given the positive real numbers $v_1 > v_2 > 0$ and $\theta_1 > \theta_2 > 0$, we have $\frac{1}{v_1+\theta_1} + \frac{1}{v_2+\theta_2} > \frac{1}{v_1+\theta_2} + \frac{1}{v_2+\theta_1}$. It then follows that, for the two-relay case, if $\gamma_{R_1D} = \gamma_{R_2D}$, the constituent codes with a larger weight w_i should be paired with a better source-relay channel (i.e. a higher $\gamma_{in,i}$). This paring rule is similar to that in [58].

4.6.3 Threshold Selection

The threshold plays an important role in real practice and directly affects the end performance. Just like the overall error rate is a function of all the channel segments, the threshold should be optimized for different number of relays and different channel conditions. For a multi-relay uncoded system, [47] carried out a rather rigorous and extensive analysis of the threshold choice for rLLRs. By formulating the problem as one

that minimizes the overall error probability, [47] showed that the optimal LLR thresholds (for uncoded systems) may be numerically computed, but the computation is very tedious involving all the channel SNRs and multiple levels of integrals. Specifically, the key results about the optimal threshold are [47]: (i) The optimal thresholds are different for different relays, and for an M-relay system, each one of the M thresholds is a function of the $2M$ channel SNRs and involves an M-level integral; (ii) the optimal threshold value tends to increase with the channel condition; (iii) the tanh form may be viewed as a special case of the proposed range-limited LLR with threshold $2\ln 3$ or 2.197, and this value is close to optimal only when the channel conditions are fairly poor (see Table III in [47] and Fig. 4.7); (iv) For 2-relay uncoded systems, the optimal threshold tends to take a value between 2 to 19, and when the channel condition is reasonable (medium to high SNRs for all the segments), a threshold value of 8 appears to provide uniformly good performance; (v) For systems with 3 relays or more, a threshold of 8 to 10 appears to also deliver a consistently good performance, which far exceeds the performance of tanh.

In this chapter, since our focus is on soft-encoding rather than pure message representation, we leveraged the previous results and used an adequate threshold value of 8, rather than going all the lengths to figure out the individual values for each different SNR. This rule-of-thumb value is obtained from numerous experiments, and it is believed that it strikes a great balance between complexity and performance. First, complexity-wise, recall that an M-parallel-relay system involves altogether $2M$ channel segments with $2M$ channel SNRs. [47] showed the that the optimal threshold value is different for each of the M relays, and they are all fairly complicated functions of the $2M$ SNRs (involving multiple levels of integrals). Hence it takes tremendous complexity to compute all the optimal threshold values even for a single channel profile, and

to let the threshold values change with the channel, the computation involved would be even more daunting. Second, performance-wise, since we consider coded systems, the "effective channel quality tends to be fairly reasonable (i.e. the combination of the error correction code and the channel can together enact a "virtual channel" with a relatively decent channel quality), in which case the rule-of-thumb threshold value of 8 appears to deliver consistently good performance. From our extensive analysis and simulations, we believe that the trouble involved in finding and changing the optimal threshold with the change of the channel outweighs the gain it brings in, and that the rule-of-thumb value 8 presents a simple and decent solution for the coded systems. This choice is validated by many Monte Carlo simulation curves shown in the paper (as well as many un-shown).

4.7 Numerical Results

To verify the efficiency of the proposed coding schemes, we now present Monte Carlo simulation results, and compare them with those of the existing distributed coding schemes. We consider BPSK at the source, and either AWGN or block fading for each communication link. The block length for the source bits is fixed to 300, and $M = 2$ parallel relays are considered, each generating a rate-1 convolutional codeword and transmitting it orthogonally (e.g. time orthogonality) to the receiver to collaboratively form a rate-1/2 distributed code. Four systems are evaluated and compared, all of which use the same average transmitting power per block to ensure a fair comparison.

a) Digital distributed coding

Reference system 1 (legend "digital"): a conventional digital distributed code, in

which each relay forces a binary hard decision ($+1$ or -1) to the received signal and encodes them via a conventional rate-1 digital encoder, and the destination performs the conventional Viterbi decoding.

b) Soft distributed coding generated by tanh

Reference system 2 (legend "tanh"): a soft-encoding distributed code recently proposed in [58] and distributed Turbo codes in [**?**], in which the soft-encoders (at the relays) take in the hyperbolic tangent function of the received signals, and the decoder (at the destination) performs a modified BCJR algorithm specifically designed for the individual convolutional code.

c) Soft distributed coding generated by range-limited LLR

The new system (legend "rLLR"): the SoER scheme proposed in this paper, in which the soft-encoders (at the relays) take in judiciously-truncated LLRs of the received signals, and the decoder (at the destination) performs modified ML decoding algorithm proposed in Subsections 4.4.2 (ML1) and 4.4.3 (ML2), matched to the individual convolutional code. For Turbo distributed code, the modified BCJR algorithm in Section 4.5 is employed. Unless otherwise stated, a threshold value $\theta = 8$ is used, which is about the best threshold we obtained experimentally (among a wide range of thresholds we tested).

We perform a comparative evaluation of the afore-mentioned systems in a variety of scenarios.

AWGN channels and distributed feed-forward convolutional codes: We first test the the proposed rLLR-based soft-coding schemes with different threshold values θ. For

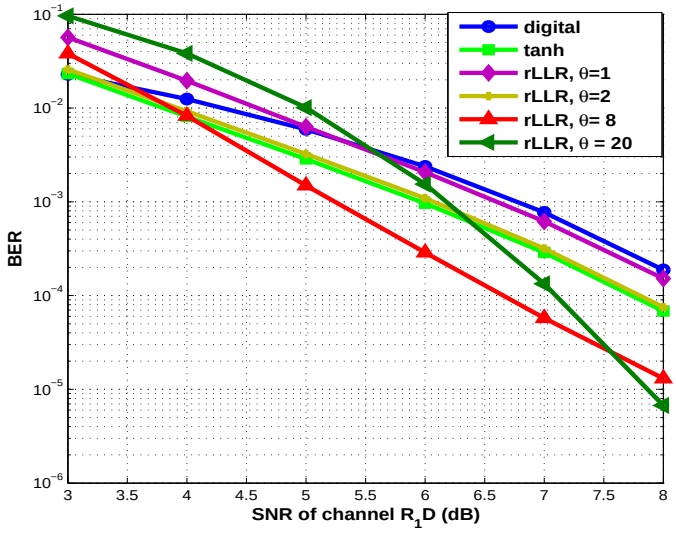


Figure 4.7: BER comparison of conventional schemes and limited-LLR based soft coding with different thresholds under AWGN channel, $(5, 7)$ distributed code, all source-relay and relay-destination channels are of the same SNR, $SNR_{SR_1} = SNR_{SR_2} = SNR_{R_1D} = SNR_{R_2D}$ (dB).

comparison purpose, the digital scheme a) and the tanh scheme b) are also plotted. Fig. 4.7 demonstrates the bit error rate of all of these cases over homogeneous AWGN channels (where all the four channel segments have the same SNR). For simplicity, no channel code is employed in the source-relay transmission, and the relay-destination transmission uses the distributed convolutional code $(5, 7)_{oct}$ depicted in Fig. 4.3. ML decoding algorithm I is adopted by the destination. By enabling the protected transmission of soft-reliability information and therefore effectively suppressing error propagation, the "tanh" soft-encoding scheme in [58] is able to achieve about 0.8 dB gain over the conventional digital coding scheme at an BER of 10^{-3}, and the proposed new scheme with $\theta = 8$ achieves an additional 0.7 dB. We attribute the additional gain of the new scheme over the previous scheme in [58] to the more appropriate forms of the soft message (i.e. carefully truncated LLRs). We also observe that the choice of the threshold value θ directly affects the end-to-end performance. When θ is set to a fairly

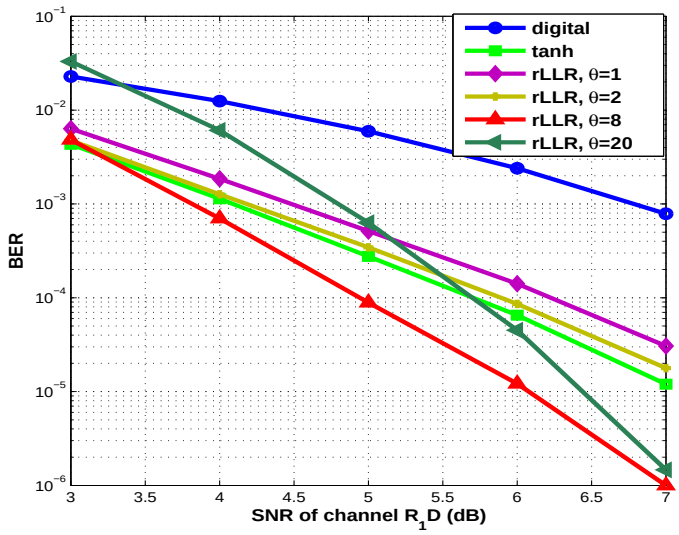


Figure 4.8: BER comparison of conventional schemes and limited-LLR based soft coding with different thresholds under AWGN channel, one source-relay channel is 3 dB better than others, $SNR_{SR_1} = SNR_{SR_2} - 3 = SNR_{R_1D} = SNR_{R_2D}$ (dB).

small value such as 1, the system performs much like a digital coding system. When θ increases to 2, the system exhibits a performance similar to that of the "tanh" soft-encoding scheme, which shows better performance at low SNR region. As θ continues to increase, the system performance also improves (to a point), and we found $\theta = 8$ to be about the best choice overall. This is because, as detailed in the paper [47], the optimal threshold is actually a changing value that increases with the channel SNRs. While value 8 appears to be delivering consistently good performance in the medium to high SNR region, the optimal threshold value can be as small as 2 in the low SNR region. Since the tanh scheme can be essentially well approximated by a special case of the range-limited LLR scheme with threshold set to $2 \ln 3$ or 2.197 [47], it should therefore not be surprising that tanh may outperform the threshold-8 range-limited LLR scheme at some point of the low SNR region. Further increasing the threshold (such as $\theta = 20$) will degrade the performance at the low-to-medium SNR region but improve it at the very high SNR region.

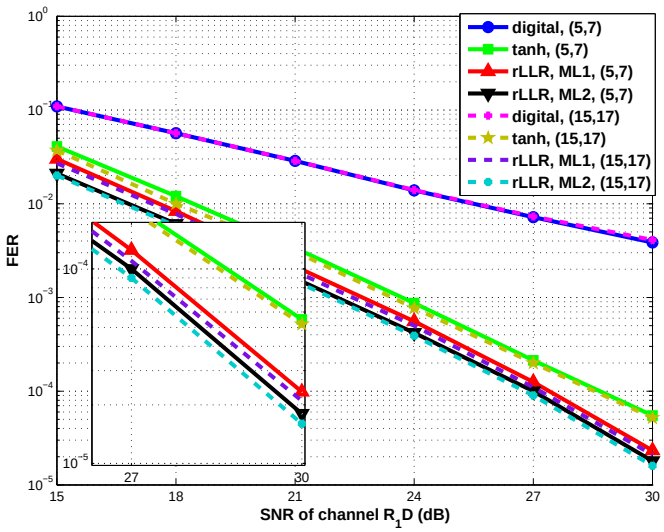


Figure 4.9: FER comparison of different schemes under Rayleigh block fading channel, $(5, 7)$ or $(15, 17)$ distributed code, all source-relay and relay-destination channels are of the same average SNR, $SNR_{SR_1} = SNR_{SR_2} = SNR_{R_1D} = SNR_{R_2D}$ (dB).

A slightly different situation is evaluated in Fig. 4.8 with heterogeneous AWGN channels, where $SNR_{SR_1} = SNR_{SR_2} - 3 = SNR_{R_1D} = SNR_{R_2D}$ (dB). Similar observations are made here: the choice of θ makes a difference (to the proposed new schemes), the tanh scheme is only slightly advantageous in a very narrow low-SNR region that corresponds to BER of $10^{-1} \sim 10^{-2}$, and the case of threshold-2 appears to deliver similar performance to tanh, and with an appropriate choice such as $\theta = 8$, the new scheme can drastically outperform the existing schemes.

Rayleigh block fading channels and distributed feed-forward convolutional codes: Fig. 4.9 presents the frame error rate of the new system (with either decoding scheme) and the two reference systems over block Rayleigh fading channels with $SNR_{SR_1} = SNR_{SR_2} = SNR_{R_1D} = SNR_{R_2D}$ (dB). No coding is deployed in the source-relay transmission, and either the $(5, 7)_{oct}$ or the $(15, 17)_{oct}$ distributed convolutional code is employed in the relay-destination leg. A wide range of channel conditions is tested from

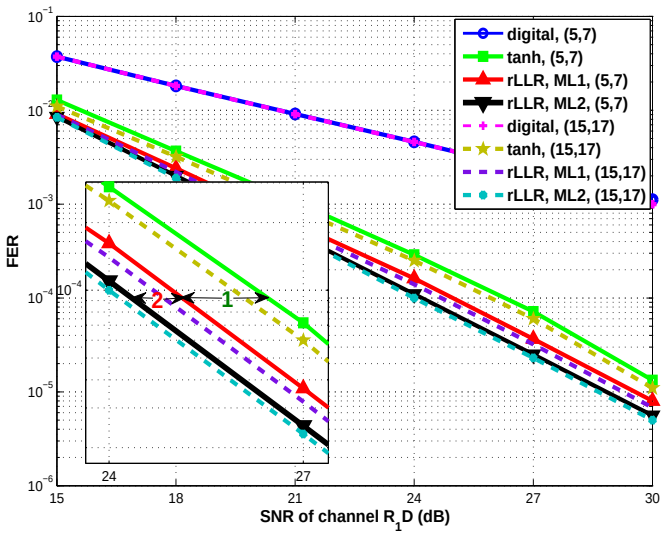


Figure 4.10: FER comparison of different schemes under Rayleigh block fading channel, $(5, 7)$ or $(15, 17)$ distributed code, the average SNRs of source-relay channels are 5 dB better than the relay-destination channels, $SNR_{SR_1} - 5 = SNR_{SR_2} - 5 = SNR_{R_1D} = SNR_{R_2D}$ (dB).

SNR_{R_1D} from 15 dB to 30 dB. As expected, code $(15, 17)_{oct}$ promises a larger gain than code $(5, 7)_{oct}$, but the gain is fairly marginal. In each case, the two soft-encoding schemes clearly exhibit far better coding gains and diversity gains over the conventional digital schemes. The reason the digital case does not (yet) exhibit as good a slope as the soft-encoding cases can be attributed to the following facts: (i) In the digital case, each relay forces a hard decision on the reception, and encodes it with a rate-1 digital code, and because the channels are relatively weak, there is severe error propagation that degrades the performance of the digital system (much more than it does to the soft-encoding cases). (ii) The diversity order, which is defined as the asymptotic slope of the error rate curve, is relevant and meaningful only in the fairly high SNR region. In the figure, the SNRs are far from being high for the digital case, as the BER of the digital case is $10^{-1} \sim 10^{-3}$. It also can be observed that the proposed new soft-encoding scheme with ML decoding algorithm I (legend "rLLR, ML1") leads the way by more

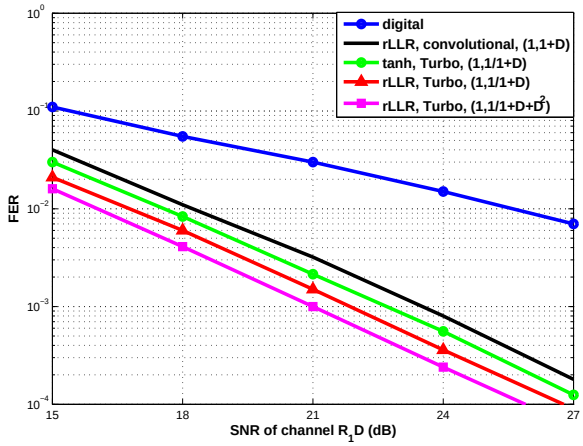


Figure 4.11: FER comparison of distributed turbo code under Rayleigh block fading channel, generated by $(1, \frac{1}{1+D})$, $SNR_{SR_1} = SNR_{SR_2} = SNR_{R_1D} = SNR_{R_2D}$ (dB).

than 1 dB gain over the previous soft-encoding one. The proposed new scheme with ML decoding algorithm II (legend "rLLR, ML2") performs the best.

In Fig. 4.10, we test these schemes over heterogeneous block fading channels with $SNR_{SR_1} - 5 = SNR_{SR_2} - 5 = SNR_{R_1D} = SNR_{R_2D}$ (dB). These simulation results again confirm the efficiency of the proposed new scheme. It is shown in Fig. 4.10 that the performance gains of our proposed schemes can be attributed to two parts: part 1 (based on range-limited LLR representation with Gaussian-approximated decoder) denotes the gain coming from the proposed message representation, and part 2 (based on the revised decoder) denotes the gain coming from the enhancement of the decoder. This is because the tanh function is nonlinear, the resulting signal at the destination becomes hard to characterize, and to assist decoding, the scheme had to resort to the popular treatment of Gaussian assumption, the PDF curves of the signals received at the destination actually deviates fairly noticeably from the true Gaussian PDF. In contrast, the piece-wise linearity of the range-limited LLRs not only makes it possible to derive a more accurate decoder, but the appropriate choice of the threshold also makes it possi-

ble for a more accurate Gaussian assumption (see Fig. 4) and hence a better-performing Gaussian-approximated decoder. In summary, the tanh scheme suffers from at least two disadvantages: the unavailability of an accurate decoder and the suboptimality in message representation in a multi-relay environment. In comparison, the proposed scheme with the range-limited LLRs and the revised decoder have advanced in both directions.

Rayleigh block fading channels and distributed turbo codes: As we have discussed, when the relays at both branches employ feed-backward convolutional codes (with appropriate interleaving performed before encoding) (see Fig. 4.6), a distributed turbo code can be formed, which is capable of even higher gains than distributed convolutional codes. Fig. 4.11 tests the concept of soft-encoding distributed turbo code. The same puncturing pattern as shown in Fig. 4.6 is employed, and the first relay uses an identity interleaver (π_1) while the send relay uses a random interleaver (π_2). Block Rayleigh fading channels with homogeneous channel SNRs are tested. We observe that the new soft-encoding turbo scheme significantly outperforms the conventional hard-encoded turbo scheme and the tanh-encoded (soft-estimate-encoded) scheme proposed in [6], as well as the soft-encoding convolutional scheme. By adopting a component code $(1, 1/(1 + D + D^2))$ with a larger memory than $(1, 1/(1 + D))$, an additional 1 dB gain is also attainable.

Chapter 5

Cooperative Forward Strategy through Signal-Superposition-Based Braid Coding

5.1 Introduction and Motivation

Consider a multi-source single-destination M-to-1 cooperative system where two or more sources (i.e. users) communicate, and at the same time help one another communicate, to the common destination [15]. Conventional relay systems are based on the practice of *store-and-forward*, where the intermediate relays receive packets from the upstream(s), buffer them, and forward them to the downstream(s) one by one *in a best effort to avoid colliding these packets with one another*. Store-and-forward is easy to implement, but fails to achieve the network capacity. To realize the full potential promised

by the theory, researchers have also looked into more sophisticated techniques.

The inception of network coding [61] completely revised the legacy philosophy by equipping all the nodes with coding capabilities, such that not only the transmitter(s) and the destination(s), but all the intermediate relays are also allowed to decode/demodulate packets and perform inter-packet encoding. When a relay node performs network coding (e.g. by bit-wise mixing the two packets) and forwards it, it has, in a sense, purposefully introduced "controlled packet collision" of these packets in the downstream. As such, network coding is considered by many as an overarching term which defines the myriad strategies that enable inter-packet coding during the course of message routing.

Noteworthy practices of network coding include, for example, physical-layer network coding (e.g. analog network coding in two-way relay systems [62], superposition modulation (superposition in signal domain) [63] [64] [65]), random-mixing coding (e.g. random bit-wise XORing packets [61], superposition in code domain [66]), and so on. These strategies applied in different system models have considerably benefited the communication systems by increasing the diversity order, improving the throughput, reducing the bit error rate (BER), and/or extending the transmit range. It should be noted, however, that a vast majority of the schemes available in the literature inevitably employ the idea of time division of some kind (or its bandwidth counterpart, frequency division) to perform cooperation communication, sometimes in combination with coded cooperation. For example, in the case of two or more sources transmitting data to a common destination, a popular framework is to divide the transmission session into two phases, where all the sources take terms to transmit its own data in the first phase, and then help one another by forwarding the data in the second phase (e.g. [17] [67] [68]). Such a practice is simple, but requires additional time resource (or frequency resource),

compared to the network-coded systems.

In contrast with previous studies of superposition modulation which mainly focus on exploring the diversity gain for the 2-user system, we have developed a rather elegant strategy that enables the resulting network code, thereafter referred to as the *braid code*[1], to simultaneously achieve diversity gain, coding gain and a full rate for the multi-user systems! The key to our success lies firstly in the operation domain – rather than perform coding in the conventional digital domain, we are able to design a real-valued superposition code (i.e. analog domain) that matches right to the underlying network topology. Additionally, we have carefully optimized the code parameters (to maximize the encoding performance), and derived a maximum-likelihood (ML) decoding algorithm (to maximize the decoding performance), all of which contribute to the excellent performance of the proposed cooperation strategy based on the braid coding.

5.1.1 Related Work

Before elaborating our cooperation strategy, we first provide a quick literature overview. Despite the existence of a myriad network-coded cooperative schemes in the literature, only a handful considered achieving M-to-1 cooperative gains without allocating additional time slots for cooperation for the M collaborating users.

A pioneering full-rate scheme for M-to-1 systems dates back to 2005 [65], in which each user in the cooperative cohort transmits a superposed signal comprising of its own data and the relaying data from the previous time slot, thus achieving a rate of 1. The

[1]While preparing for this manuscript, we realize that the term "braid coding" has also been used in the literature [70] to refer to a kind of digital turbo-like code. The braid code proposed in this paper is different from that in [70]. Our braid code is essentially an analog convolutional code that belongs to the class of physical network coding, specifically superposition modulation.

diversity-multiplexing tradeoff is analyzed in [65]; however, the superposition coefficients, which signify how much energy is allocated for the self data and for the relaying data, respectively, are not optimized. It does not provide any BER performance analysis, nor BER simulation curves.

A variation of the scheme discussed in [65] is later proposed, which considers cleaning up some of the relaying data (e.g. via decode-and-forward), before constructing the new superposition signal [63] for the 2-user system. The coefficients are subsequently optimized under various channel qualities in [72]. The optimization is based on a numerical method, where an upper bound of the packet error rate (PER) is firstly formulated, but since the closed-form expression is intractable, the researchers then resort to numerical exhaustive search to determine appropriate coefficients. The outage probability is analyzed in [71]. All of the work focus on the 2-user system, because the numerical searching task becomes intractable when considering a system with more users. Apparently, by cleaning up some of the previous information, [63] is able to concentrate transmit power only on the fresh data of itself and the most recent correct-decoded data belonging to the partner, which can be decided by cyclic redundancy check (CRC) codes. [73] proposed an opportunistic scheme where a user would forward a superposed signal only if the achievable inter-user channel capacity is larger than the required transmission rate, and would otherwise revert to the non-cooperative case. Then it analyzes the outage probability under this case for the symmetric 2-to-1 system, in which the two user-destination channels are of the same quality. To decrease the decoding complexity, a suboptimal decoder for the superposition modulation is developed in [74]; [75] considers a forward message passing for the 2-user system. The strategy in [76] pulls in the feedback channel from the destination to facilitate a higher gain.

The above-mentioned studies all considered binary phase shift keying (BPSK), which equates to 1-dimensional (1-D) superposition modulation. The work of [77] investigated quadrature phase-shift keying (QPSK) or 2-dimensional (2-D) superposition for a 2-to-1 system, but the scheme takes up additional time slots (i.e. 3 slots for 2 data packets) for cooperation, resulting in a loss of code rate.

Some of these superposition operations are also adopted in networks other than the M-to-1 topology. For example, [78] considered a one-source, multiple-relay, and one-destination 2-hop system, while [79] considered a two-source, one-relay, and two-destination 2-hop system.

5.1.2 Novelty and Contributions

The primary interest of this chapter is to develop a practical network-coded cooperative strategy for M-to-1 networks that can simultaneously achieve a full diversity gain, a desirable coding gain, and a full rate. Our proposal is a progressive real-domain coding strategy termed the *braid coding*. The code is named after a combining process similar to how a girl French braids her hair: As each user takes its turn to transmit, it combines its own data (fresh data to be transmitted) with what it hears from the system in the preceding time slot(s) (previously-transmitted data to be relayed), by applying appropriate processing on and assigning appropriate weights for each. The proposed braid coding structure is a particular realization of physical-layer network coding, and more specifically, a subclass of superposition modulation[2]. The proposed coding structure is advantageous in that it is well-defined (being connected with convolutional codes of

[2]It is noted that the analog network coding in [62] [80] [81] focuses on two-way relay channels, in which two, or several users exchange information with the help of one relay node. Our paper is interested in M-to-1 systems, in which the users serve as both the source and the relay nodes.

different structures), quite broad (which subsumes several previous conventional superposition modulation schemes such as [63] and [65] as its special cases), equipped with a rather simple encoding and an ML-optimal decoding algorithm, and, best of all, achieves impressive performance gain without scarifying any rate loss. We classify braid coding into the regenerative and the nonregenerative types, characterize their respective properties, and show that the key lies in the judicious choice of the weights and the constraint length of the braid code.

In the case of *nonregenerative* braid coding, each user takes in the relay data without any decoding (or detection) effort, and blends it right into the fresh data. The advantage is the operational simplicity on the user end and the ability to achieve a full diversity gain [65], but the decoding complexity at the common destination can be high, and there is a chance for dispersive error when the inter-user channels are less than desirable. We formulate the scheme as a real-domain recursive convolutional code $1/(a_0 + a_1 D)$, where D stands for the delay, and a_0 and a_1 are the weights for the fresh data and the relay data, respectively. Using the pair-wise error probability (PEP) as a figure of merit, we theoretically derive the globally optimal values for a_0 and a_1, which maximize the coding gain for every individual transmission and for the entire session. While diversity gain is always attainable regardless of the weight assignment (provided $a_0 > 0$ and $a_1 > 0$), simulations show that the optimal coefficients can bring considerable additional coding gains (than otherwise).

In comparison, *regenerative* braid coding requires each user to decode and clean up the relay data, before reassembling some of them together with its fresh data. We show that the scheme can be formulated as a general real-domain non-recursive convolutional code $(b_0 + b_1 D + \cdots + b_m D^m)$ of memory m. When $m = 1$, the regenerative braid code im-

plements the same superposition modulation discussed in [63]. We derive the globally optimal values for the weights b_j that achieve the PEP optimality in every transmission and for arbitrary m, and demonstrate a modified Viterbi algorithm for the common destination to optimally decode all the data in linear time. The key in the design of regenerative code is a balance between coding gain (which may favor a larger m) and complexity (which favors a smaller m). Through free-distance analysis and computer simulations, we recommend $m = 2$ and 3 for the 2-to-1 system, and demonstrate their performance advantages over the nonregenerative case (including [65]) and the previous superposition modulation cooperative schemes (including [63]). We recommend the regenerative braid code with $m = M - 1$ as a particularly attractive candidate for the M-to-1 systems when M is large for its excellent performance and simple decoding strategy at the destination.

The main contribution of the paper is summarized as follows:

- A general class of superposition-based coding strategy is proposed to match to the network topology of a M-to-1 multi-user single-destination cooperative system. The proposed *braid code* is capable of simultaneously achieving diversity gain, coding gain and a full rate. Optimal choices of the code parameters (including the weights and the constraint length) optimizing the PEP is analytically derived. We prove that the regenerative-braid-coding-based M-to-1 cooperative scheme can reach a full diversity order, and derive the theoretical BER performance for the $m=2$ regenerative braid coding.[3]

- By exploiting the structure relevance of braid codes with trellis codes, we are

[3]Recall that $m = 2$ strikes the best tradeoff between performance and complexity. It is also possible to derive the BER performance for other values of m using the same method we discuss in the paper, but the expression gets tediously complicated as m increases.

also able to leverage several excellent coding ideas (such as the linear-complexity maximum likelihood trellis-based decoding algorithm for the regenerative braid coding) to achieve both low complexity and good performance.

- In addition to non-channel-coded systems, we have also considered systems with soft-decodable channel codes. In particular, with regenerative braid codes, we present a soft-iterative joint channel-network decoding strategy which makes effective use of the log-likelihood ratio (LLR) reliability information from all the received signals via a forward message passing and a backward message passing. The algorithm is shown to deliver gains compared to the conventional forward message passing decoder.

Notation: (i) Unless otherwise stated, we use lowercase boldface and uppercase boldface letters to denote vectors and matrices, respectively, and use regular letters to denote scalars and random variables. (ii) $\mathcal{N}(m, \sigma^2)$ represents the Gaussian distribution with mean m and variance σ^2.

5.2 Braid Coding Cooperative Scheme

The proposed braid code works for general M-to-1 cooperative systems, but for ease of proposition, our discussion below focuses on $M = 2$.

Let S_1 and S_2 be the two sources taking turns to communicate to the common destination D. Suppose each communication session consists of N equal-length time slices, and each time slice consists of two equal halves assigned to S_1 and S_2 respectively. Since user cooperation is most useful where time and space diversity is hard to get, we

consider slow fading such that all the channel state information (CSI) remains invariant within each communication session (but changes independently between sessions). Following the convention [63] [65] [71] [74], we assume that all the CSIs are known to the respective receiving users and the common destination, which means CSI can be estimated with fairly high accuracy.

Let subscript $i \in \{1, 2\}$ be the user index and subscript $k \in \{1, 2, \cdots N\}$ be the time index. Let $s_{i,k} \in \{\pm 1\}$ and $x_{i,k} \in \mathcal{R}$ be the fresh data and the transmitted signal from S_i at time k, respectively, and let $y_{i,k}$ and $r_{i,k}$ be the corresponding reception at the other user and at the destination, respectively. The idea to achieve full diversity and power gain is to have each user superpose its fresh data with the relay data, using appropriate braiding schemes and weights.

• **Nonregenerative Braid Coding** $1/(a_0 + a_1 D)$.

Here, each user takes in what it hears from the other user as it is (without any decoding or signal processing), and blends it with its fresh data via signal superposition. Mathematically, we have

$$\begin{aligned}
k=1 : \mathsf{S}_1 &: x_{1,1} = a_0 s_{1,1}, \\
\mathsf{S}_2 &: x_{2,1} = a_0 s_{2,1} + a_1' y_{1,1} = a_0 s_{2,1} + a_1' (h_0 x_{1,1} + z_{1,1}), \\
k=2 : \mathsf{S}_1 &: x_{1,2} = a_0 s_{1,2} + a_1' y_{2,1} = a_0 s_{1,2} + a_1' (h_0 x_{2,1} + z_{2,1}), \\
\mathsf{S}_2 &: x_{2,2} = a_0 s_{2,2} + a_1' y_{1,2} = a_0 s_{2,2} + a_1' (h_0 x_{1,2} + z_{1,2}),
\end{aligned}$$

and so on, where h_0 is the Rayleigh distributed with mean zero and unit variance CSI for the inter-user channel, $z_{i,k}$ is the zero mean additive white Gaussian noise (AWGN) with variance σ_0^2 for the inter-user channel (assuming channel reciprocity), and a_0 and a_1' are

the weights assigned to the fresh data and the relay data, respectively. Correspondingly, the common destination receives

$$r_{i,k} = h_i x_{i,k} + n_{i,k},$$

for $i = 1, 2$ and $k = 1, 2, \cdots, N$, where h_i is Rayleigh distributed CSI with zero mean and unit variance, $n_{i,k} \sim \mathcal{N}(0, \sigma_i^2)$ is AWGN for the S_i-D channel. Denote a per-transmission power constraint as P, the instantaneous and average signal-to-noise ratio (SNR) of each channel is $\gamma_i = \frac{|h_i|^2 P}{2\sigma_i^2}$ and $\bar{\gamma}_i = \frac{P}{2\sigma_i^2}$, respectively.

Let $a_1 = a_1' h_0$ be the channel-adjusted weight for the relay data. The signals transmitted by each user with a size N, $\mathbf{x} = [x_{1,1}, x_{2,1}, \cdots, x_{1,N}, x_{2,N}]^T$, can then be rewritten in a compact matrix form as:

$$\mathbf{x} = a_0 \begin{bmatrix} 1, & 0, & 0, & \cdots & 0 \\ a_1, & 1, & 0, & \cdots & 0 \\ a_1^2, & a_1, & 1, & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1^{2N-1}, & a_1^{2N-2}, & a_1^{2N-3}, & \cdots & 1 \end{bmatrix} \begin{bmatrix} s_{1,1} \\ s_{2,1} \\ s_{1,2} \\ \vdots \\ s_{2,N} \end{bmatrix} = \mathbf{G}\mathbf{s}, \tag{5.1}$$

and the corresponding reception at the destination becomes

$$\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{n}, \tag{5.2}$$

where $\mathbf{n}$ is the noise vector following $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, $\mathbf{\Sigma}$ and $\mathbf{H}$ are a $2N$-by-$2N$ cyclic-2 diagonal square matrix.

From the coding perspective, the nonregenerative scheme is similar in spirit to the

recursive code $1/(a_0 + a_1 D)$ followed by a cyclic-2 fading channel, see the linear shift register (LSR) representation in Fig. 5.1-left. It requires minimal effort on the user side, but since the resultant real-domain trellis has a growing number of states with time, the overall code is not linear-time decodable at the destination.

• **Regenerative Braid Coding** $(b_0 + b_1 D + \cdots + b_m D^m)$**.**

In the regenerative case, we consider the adaptive transmission scheme which means one source node would only help another when it correctly decodes the new information sent by another node. Or a new session will start. In the desired case when user decoding is all successful (i.e. good inter-user channel), the braid code can last all the way to the end of the session time $2N$; but if at some point a user fails to decode, then braid code and the session terminate early, and a new code (session) will start. Each user performs progressive decoding on what it hears from the system, and re-packs some of them together with its fresh data using appropriate combining weight. For example, if each user superposes its fresh data with m previous source data (of which $\lfloor \frac{m}{2} \rfloor$ belong to itself and $\lceil \frac{m}{2} \rceil$ belong to the other user), then the signals that is successively transmitted by the two users will take the following matrix form:

$$\mathbf{x} = \begin{bmatrix} b_0, & 0, & 0, & 0, & 0, & \cdots & 0 \\ b_1, & b_0, & 0, & 0, & 0, & \cdots & 0 \\ b_2, & b_1, & b_0, & 0, & 0, & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & 0 \\ b_m, & \cdots & b_2, & b_1, & b_0, & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0, & \cdots & b_m & \cdots & b_2 & b_1 & b_0 \end{bmatrix} \begin{bmatrix} s_{1,1} \\ s_{2,1} \\ s_{1,2} \\ \vdots \\ \cdot \\ \vdots \\ s_{2,N} \end{bmatrix} = \mathbf{G}\mathbf{s}. \tag{5.3}$$

In general, the braid code seen by the common destination takes the form of a real-domain nonrecursive convolutional code $(b_0+b_1D+\cdots+b_mD^m)$. Consider the power constraint of each transmission as P, each time the users normalize the transmission power by using factor β. After power normalization, $\mathbf{x}$ is transmitted to the destination through the block fading channel as described above. An example LSR for $m=2$ is shown in Fig. 5.1-right. Comparing to the nonregenerative case, here the code has a fixed number of states (2^m) in the trellis, and the destination can therefore resort to the Viterbi algorithm to decode all the data efficiently and optimally. (The nonregenerative code must use a higher-complexity algorithm such as the list decoding.) Further, although the users have also performed decoding in each of their cooperation stages, their decoding involves only data subtraction (signal cancellation) – provided that each user is provisioned with m memories to store the historic source data – and hence has an extremely low complexity.

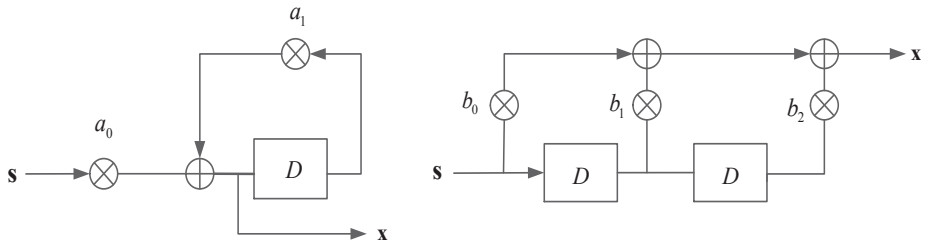


Figure 5.1: LSR structure of nonregenerative (left) and regenerative $m=2$ (right) braid coding.

Remark: Extension to 2-D signal superposition is considered when the source bits $s_{i,k}$ are QPSK modulated, instead of BPSK. Assume $s_{i,k}\in\{\pm1\pm i\}$. Since $s_{i,k}$ consists of two orthogonal signal spaces, the transmitted signal can be written as a two-dimensional superposed signal. All the analysis of the 1-D superposition in this paper can be applied to the 2-D signal superposition directly.

5.3 Code Optimization

The performance of the braid coding and hence the cooperative gain closely depend on the choice of the weights. [65] was the first to demonstrate an example of nonregenerative braid coding, [63] [71] [72] presented a regenerative case with memory $m = 1$, and [77] focused on two-dimensional superposition modulation. These papers also suggested a few empirical weight choices, but lack analytical results. Below we provide a rigorous derivation of the optimal weights that simultaneously achieve per-transmission optimality, where the optimality is measured in terms of the pairwise error probability of the two nearest neighbors in the signal constellation (worst-case PEP).

Theorem 4. *Under a given power constraint, the optimal amplitude shift keying (ASK) that minimizes the worst-case PEP (or, equivalently, maximizes the minimum distance) is one that has a uniform constellation.*

Proof. (Proof by contradiction) It is known that the energy-efficient ASK signal space is two-sided, symmetric, and have the centroid in the origin [82]. Now suppose that the optimal ASK which minimizes the worse-case PEP under a given power constraint P does not have a uniform constellation; see Fig. 5.2 (a) for an illustration of an 2^M-ASK. Without loss of generality, suppose this 2^M-ASK has a minimum distance d_{min} between $x_{1,1}$ and $x_{1,2}$ (as well as between $x_{0,1}$ and $x_{0,2}$). Using $d_{min} = d(x_{1,1}, x_{1,2})$ as the distance, we can construct a *uniform* 2^M-ASK by moving the constellation points of the original 2^M-ASK closer towards the origin. As shown in Fig. 5.2 (b), the resulting 2^M-ASK has a uniform constellation, a minimum distance of d_{min}, and an overall average power smaller than P. We can then uniformly expand all the signal points in the constellation in Fig. 5.2 (b), such that the new constellation remains a uniform 2^M-ASK,

but has an overall average power of P. This new constellation, shown in Fig. 5.2 (c), clearly has a minimum distance larger than d_{min}, and hence contradicts our assumption. In this, we have shown that the optimal 2^M-ASK that maximizes the minimum distance must be a uniform constellation. In fact, it can be calculated that $d_{min} \leq \sqrt{\frac{3P}{2^{M-2}(2^{2M}-1)}}$ for 2^M-ASK, and the equality is achieved only with a uniform, symmetric constellation.

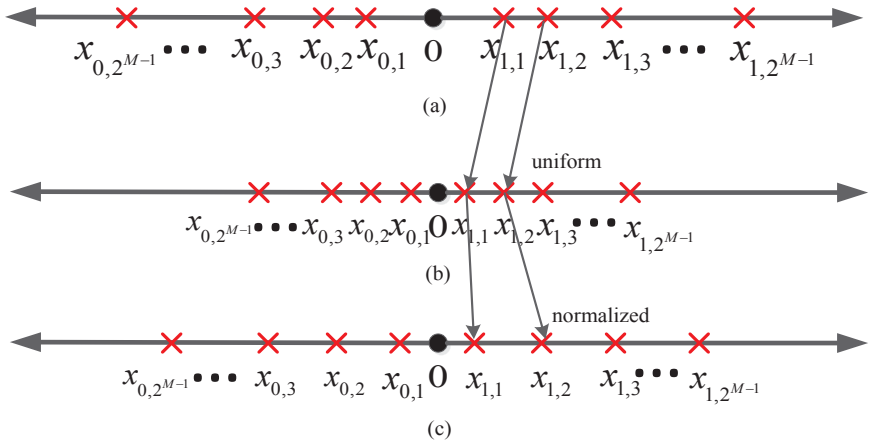


Figure 5.2: (a) ASK with a non-uniform constellation under a given power constraint; (b) ASK with a uniform constellation without power normalization; (c) Normalized ASK with a uniform constellation.

□

In 1-D superposition modulation, each user essentially transmits an ASK-modulated signal – possibly with a different constellation size – every time. The question then is whether it is possible or how to find appropriate values of a_i's and b_i's such that the nonregenerative/regenerative code will achieve a uniform signal constellation *every time* of the transmission.

Theorem 5. *Consider nonregenerative braid coding, the choice $a_1 = 1/2$ (and arbitrary nonzero a_0) will guarantee a uniform ASK constellation in every transmission.*

Proof. In the nonregenerative case, the users take turns to transmit an ever-increasing

ASK constellation – each time the size doubles that of the previous one. Specifically, the signals transmitted by S_1 and S_2 at time k are

$$x_{1,k} = a_0\big(s_{1,k} + a_1 s_{2,k-1} + a_1^2 s_{1,k-1} + ... + a_1^{2k-2} s_{1,1}\big), \tag{5.4}$$

$$x_{2,k} = a_0\big(s_{2,k} + a_1 s_{1,k} + a_1^2 s_{2,k-1} + \ ... \ + a_1^{2k-1} s_{1,1}\big), \tag{5.5}$$

where $s_{i,j} \in \{+1, -1\}$ for $i = 1, 2$ and $j = 1, 2, \cdots, k$. Clearly, a_0 is just a scalar that does not affect the signal spacing whatsoever. To show $a_1 = 1/2$ will consistently produce a uniform constellation, it is sufficient to show that the set $\mathcal{X}_n = \{s_0 + \frac{1}{2}s_1 + \frac{1}{2^2}s_2 + \cdots + \frac{1}{2^n}s_n : s_i \in \{+1, -1\}, i = 1, 2, \cdots, n\}$ is uniform for all non-negative integer n.

Let $s_i' = \frac{1-s_i}{2} \in \{0, 1\}$. To show that $\mathcal{X}_n$ is uniform is equivalent to show that $\mathcal{X}_n' = \{s_0' + \frac{1}{2}s_1' + \frac{1}{2^2}s_2' + \cdots + \frac{1}{2^n}s_n' : s_i' \in \{0, 1\}, i = 1, 2, \cdots, n\}$ is uniform. The latter comes directly from the fact that $\mathcal{X}_n'$ is essentially the base-2 numeral system $(s_0'.s_1's_2' \cdots s_n')_2$. (Alternatively, the uniformity of $\mathcal{X}_n$ can be proven using mathematical induction.) □

Remark: (i) Recall that $a_1 = a_1' H_0$, where H_0 is the inter-user channel fading coefficient. This suggests that it is enough for the respective receiving user (and no need for the common destination) to know the inter-user CSI H_0. For ease of discussion, we have assumed that the fading coefficients remain unchanged during a session; but H_0 does not have to be invariant (nor does H_1 or H_2). As long as the respective user compensates for H_0 by choosing the right weight $a_1' = \frac{1}{2H_0}$, the signals are bounded between $-2a_0$ and $2a_0$, and the common destination can guarantee to receive optimal signal every time throughout the session. (ii) It can be easily verified that $a_1 = \frac{1}{2}$ is the only valid choice that will guarantee unanimously uniform constellations with bounded energy. The only other choice $a_1 = 2$ that makes uniform constellations will lead to ever increasing and

hence infinite transmit energy.

Theorem 6. *Consider memory-m regenerative braid coding among users. The choice $b_i = \frac{1}{2}b_{i-1}$ ($i = 1, 2, \cdots, m$, and arbitrary positive b_0) will guarantee a uniform ASK constellation in every transmission.*

Proof. In memory-m regenerative coding, the signals transmitted by S_1 and S_2 are given in (5.3). The choice $b_i = \frac{1}{2}b_{i-1}$ will lead to

$$\mathbf{x} = b_0 \begin{bmatrix} 1, & 0, & 0, & 0, & 0, & \cdots & 0 \\ \frac{1}{2}, & 1, & 0, & 0, & 0, & \cdots & 0 \\ \frac{1}{2^2}, & \frac{1}{2}, & 1, & 0, & 0, & \cdots & 0 \\ \vdots & \ddots & \ddots, & \ddots, & 0, & \vdots & 0 \\ \frac{1}{2^m}, & \cdots & \frac{1}{2^2}, & \frac{1}{2}, & 1, & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0, & \cdots, & \frac{1}{2^m}, & \cdots & \frac{1}{2^2}\cdot & \frac{1}{2}\cdot & 1 \end{bmatrix} \begin{bmatrix} s_{1,1} \\ s_{2,1} \\ s_{1,2} \\ \vdots \\ \cdot \\ \vdots \\ s_{2,2N} \end{bmatrix},$$

where each row of $\mathbf{x}$ constitutes a transmission. To see each transmission corresponds to a uniform ASK, it is enough to show $\mathcal{X}_n = \{s_0 + \frac{1}{2}s_1 + \frac{1}{2^2}s_2 + \cdots + \frac{1}{2^n}s_n : s_i \in \{+1, -1\}, i = 1, 2, \cdots, n\}$ is uniform for $n = 0, 1, \cdots, m$; and in the proof of Theorem 5, we have shown this for $n = 0, 1, \cdots, \infty$. □

Remark: (i) The nonregenerative braid code is like an infinite impulse response (IIR) filter, and the regenerative code is like a finite impulse response (FIR) filter. Since the regenerative code has a finite memory size m, the choice $b_i = 2b_{i-1}$ for $i = 1, 2, \cdots, m$ will also lead to uniform constellations with bounded energy, but then the transmit energy of the first $(m-1)$ transmissions (and especially the first transmission) will be

significantly smaller than that of the later transmissions. If we follow the communication convention of keeping per-transmission energy as uniform as possible, then the solution of $b_i = \frac{1}{2} b_{i-1}$ becomes unique. (ii) 2-D superposition cases just extend one dimension compared to 1-D cases. Though the above discussion focuses on 1-D cases, the code optimization also works for 2-D cases.

5.4 Decoding Algorithm

Upon receiving the braid codes collaboratively generated by both source nodes, the destination node performs a decoding process. We consider a maximum-likelihood decoder that produces the most probable codeword. Recall that the general ML-optimal decoding algorithm has a rather high complexity that increases exponentially with the block size. For regenerative braid coding, we are able to leverage from the digital coding concepts and tools, and develop a trellis-based sequence decoding algorithm that achieves ML optimality with linear complexity. For nonregenerative braid coding, whose ML decoding complexity increases rather quickly with the block size, we also present a decorrelating (DC) detector, a minimum mean square error (MMSE) detector and a polynomial expansion (PE) detector, that gives suboptimal performance in exchange of lower complexity.

5.4.1 Viterbi Decoding of Regenerative Code

The finite memory of regenerative braid codes not only allows the users to perform simple cancellation-based decode-and-forward (and can therefore clean up the inter-

user channel noise), but also allows the common destination to perform efficient Viterbi decoding on a trellis of 2^m states. Fig. 5.3 demonstrates an example of such a trellis with $m=2$, where each branch from the state j to the state i is associated with inputs of ± 1 and outputs of $c_{j,i} = \pm b_0 \pm b_1 \pm b_2$. The initial two time stages, where the branches are associated with outputs $\pm b_0$ and $\pm b_0 \pm b_1$, respectively, are not shown. For the 1-D superposition modulation, the branch metric $\alpha_{j,i}^t$ from the state j to the state i at the stage t is calculated by

$$\alpha_{j,i}^t = \frac{(r_{t',t} - h_{t'} c_{j,i})^2}{\sigma_{t'}^2}. \tag{5.6}$$

where $t' = 1$ when t is odd; $t' = 2$ when t is even.

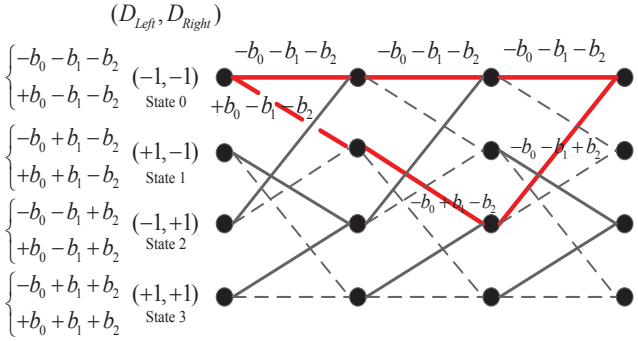


Figure 5.3: Trellis for regenerative code $(b_0 + b_1 D + b_2 D^2)$ (solid lines are associated with input -1 and dashed lines are associated with input $+1$)

The branch metric is accumulated to form the path/state metric. It is worth noting that the overall does not necessarily end in the all-zero state, and the code is therefore a "non-terminating" code. The complexity of the decoder is $O(2N2^m)$ for a communication session with N cooperative rounds, which is linear to session length. 2-D superposition modulation can be decoded by two parallel 1-D Viterbi decoders described above.

A short summary of this ML Viterbi decoding process goes in Algorithm 3.

Algorithm 3: ML decoding algorithm.

Input: Reception from the relay-destination transmission $\mathbf{r}$.
Output: Binary decisions of the original source data $\mathbf{s}$.
Initialization:
A trellis corresponding to the regenerative code $(b_0+b_1D+b_2D^2)$ is constructed, as shown in Fig. 5.3.
Stage $t=1$, $c^1_{0,0}=-b_0$, $c^1_{0,1}=+b_0$, calculate the branch metric $\alpha^1_{0,0}$ and $\alpha^1_{0,1}$, all other branch metrics are infinity; all the state metrics $\beta^1_i=0$, $i=0,1,2,3$.
Stage $t=2$, $c^2_{0,0}=-b_0-b_1$, $c^2_{0,1}=b_0-b_1$, $c^2_{1,2}=-b_o+b_1$, $c^2_{1,3}=+b_0+b_1$.
Stage $t\geq 2$, the branch labels are shown as Fig. 5.3.
Trellis Decoding:
for stage t from **2 to 2N do**
  for state i from **0 to 3 do**
    for each branch entering state i **do**
      $\beta^t_i=\beta^{t-1}_j+\alpha^{t-1}_{j,i}$
      $\alpha^t_{j,i}=\frac{\left(r_{t',t}-h_{t'}c_{j,i}\right)^2}{\sigma^2_{t'}}$
      choose the branch with the smaller β^t_i.
Stage $t=2N$, trace back the survival path; the binary input bits corresponding to the survival path are declared as $\mathbf{s}$.

5.4.2 Linear Detector of Nonregenerative Code

As the coding matrix of the nonregenerative code is a low-triangle matrix, the number of states of the Viterbi decoder will increase exponentially with the session length N, which disables the usage of the Viterbi decoding algorithm. Several sub-optimal detectors for the nonregenrative codes are discussed in this section. For every session, the received signal at the destination can be expressed as

$$\mathbf{r}=\mathbf{H}\mathbf{x}+\mathbf{n}=\mathbf{H}\mathbf{G}\mathbf{s}+\mathbf{n}. \tag{5.7}$$

Let $\mathbf{L}=\mathbf{H}\mathbf{G}$, the decorrelating detector can be employed at the destination. The

estimate of $\mathbf{s}$ is given by

$$\hat{\mathbf{s}}_{DC} = \mathbf{L}^{-1}\mathbf{r}. \tag{5.8}$$

The detector is easy to implement, but the main drawback of this detector is the noise enhancement effect. MMSE detector is able to solve this problem, which yields the estimate

$$\hat{\mathbf{s}}_{MMSE} = (\mathbf{L} + \mathbf{\Sigma})^{-1}\mathbf{r}. \tag{5.9}$$

Both the two detectors require the computation of the inverse matrices, which is difficult to implement when $\mathbf{L}$ is large. An alternative detector adopted for our nonregenerative braid coding is a polynomial expansion (PE) detector [83]. We use polynomials to approximate the corresponding matrix inverse, so as to mitigate the complexity increase caused by the matrix inversion operation. The estimate of $\mathbf{s}$ is calculated by

$$\hat{\mathbf{s}}_{PE} = \sum_{i=1}^{K} w_i \mathbf{L}^i \mathbf{r}, \tag{5.10}$$

where w_i, $i = 1, 2, ...K$, is the coefficient to be optimized subject to a cost function for a given $\mathbf{G}$ and K. By choosing different cost functions, PE detector can approximate the decorrelating detector and the MMSE detector.

It can be seen from eq. (5.10) that the estimate of $\mathbf{s}$ is a linear combination of the vectors $\mathbf{p}_i = \mathbf{L}^i\mathbf{r}$, $i = 1, 2, ..., K$. Let $\mathbf{w} = [w_1, w_2, ..., w_K]^T$, and $\mathbf{P} = [\mathbf{p_1}, \mathbf{p_2}, ..., \mathbf{p_K}]$.

The MMSE operation is equivalent to optimize the cost function

$$E[|\mathbf{s} - \mathbf{P}\mathbf{w}|^2], \tag{5.11}$$

with respect to $\mathbf{w}$. By minimizing eq. (11), we can obtain

$$\mathbf{w} = (E[\mathbf{P}^T\mathbf{P}])^{-1}E[\mathbf{P}^T\mathbf{s}], \tag{5.12}$$

where

$$\begin{aligned} E[\mathbf{P}^T\mathbf{P}](i,j) &= \text{Tr}[\mathbf{L}^{i+j+2}] + 2\text{Tr}[\mathbf{\Sigma}\mathbf{L}^{i+j}], \\ E[\mathbf{P}^T\mathbf{s}](i) &= \text{Tr}[\mathbf{L}^{i+1}]. \end{aligned} \tag{5.13}$$

The complexity of inversing matrix is thus reduced, which leads to a low complexity detector at the destination. When K is large, the performance of the PE detector can approach the performance of the MMSE detector.

5.5 Theoretical Performance Analysis

5.5.1 Free distance d

Since the performance analysis of the nonregenerative braid coding is well investigated by [65], we focus on the theoretical performance of the progressive regenerative braid coding in this section. Not only do the weights b_i's, but the memory size m also directly affects the code performance (as well as the complexity). We now identify the

optimal m that leads to the best overall regenerative braid code, and we do so by evaluating the free distance of the corresponding trellis. We calculate the free distance d first.

Theorem 7. *The free distance for a regenerative braid code* $(b_0+b_1D+\cdots+b_mD^m)$*with optimal coefficient is* $d_{free}(m) = m + 1$.

Proof. Let $\mathbf{s} = [\cdots, s_t, s_{t+1}, \cdots]$ be the source sequence that was transmitted (the correct path), $s_i \in \{\pm 1\}$, $i = 1, 2...$; and let the competing path $\tilde{\mathbf{s}} = [\cdots, \tilde{s}_t, \tilde{s}_{t+1}, \cdots]$ diverge from at time stage t (i.e. $s_t \neq \tilde{s}_t$). Consider encoding $\mathbf{s}$ and $\tilde{\mathbf{s}}$ using the linear shift register. Let $v_1, v_2, \cdots, v_m$ be the values of the registers $D^1, D^2, \cdots, D^m$ at time t, respectively. We have the following codeword for $\mathbf{s}$ (starting at time t):

$$\mathbf{c}(\mathbf{s}) = \underbrace{\begin{bmatrix} b_0, b_1, b_2, \cdots, b_m \end{bmatrix}}_{\mathbf{b}} \underbrace{\begin{bmatrix} s_t, & s_{t+1}, & s_{t+2}, & \cdots \\ v_1, & s_t, & s_{t+1}, & \cdots \\ v_2, & v_1, & s_t, & \cdots \\ v_3, & v_2, & v_1, & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ v_m, & v_{m-1}, & v_{m-2}, & \cdots \end{bmatrix}}_{\mathbf{S}}.$$

Similarly, we have the codeword $\mathbf{c}(\tilde{\mathbf{s}}) = \mathbf{b}\tilde{\mathbf{S}}$ for source sequence $\tilde{\mathbf{s}}$, and the Euclidean

distance between them is:

$$d_E = |\mathbf{b}(\mathbf{S}-\tilde{\mathbf{S}})| = |\mathbf{b}\begin{bmatrix} |s_t-\tilde{s}_t|, & |s_{t+1}-\tilde{s}_{t+1}|, & |s_{t+2}-\tilde{s}_{t+2}|, & \cdots \\ 0, & |s_t-\tilde{s}_t|, & |s_{t+1}-\tilde{s}_{t+1}|, & \cdots \\ 0, & 0, & |s_t-\tilde{s}_t|, & \cdots \\ 0, & 0, & 0, & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ 0, & 0, & 0, & \cdots \end{bmatrix}.|$$

Clearly, d_E is minimized when $\tilde{s}_t \neq s_t$, but $\tilde{s}_j = s_j$, $\forall j \neq t$, in which case $(\mathbf{S}-\tilde{\mathbf{S}}) = [diag(|s_t-\tilde{s}_t|), \mathbf{0}] = [diag(2), \mathbf{0}]$, and the free distance becomes $m+1$. □

5.5.2 Diversity Order

In a desired case, for which the SNR of the inter-user channel is high enough, the session length of the braid coding can always last to $2N$ in the 2-user system (let N be arbitrary large). We assume the system is in continuous operation. Let $\mathbf{x}$ be the transmitted codeword sequence. The destination decodes the output as $\mathbf{x}'$. Given the power constraint $P = 1$ per transmission, we can calculate the first event error probability [60] by

$$\begin{aligned} P_d &= \Pr\{m(\mathbf{r}|\mathbf{x}') > m(\mathbf{r}|\mathbf{x})\} \\ &= \Pr\{\sum_{l=1}^{t}(r_l - h_k x'_l)^2 < \sum_{l=1}^{t}(r_l - h_k x_l)^2\} \\ &= \Pr\{\sum_{l=1}^{t}(-2r_l h_k x'_l + h_k^2 x'^2_l) < \sum_{l=1}^{t}(-2r_l h_k x_l + h_k^2 x_l^2)\}, \end{aligned} \tag{5.14}$$

where $k = ((l+1) \bmod 2) + 1$, $m(\mathbf{r}|\mathbf{x})$ denotes the path metric.

Suppose $\mathbf{x}'$ and $\mathbf{x}$ differ on d stages $p[1], p[2], ..., p[d]$, P_d is simplified to

$$P_d = \Pr\{\sum_{l=1}^{d} 2h_{p[k]}r_{p[l]}(x_{p[l]} - x'_{p[l]}) < \sum_{l=1}^{d} h_{p[k]}^2(x_{p[l]}^2 - x_{p[l]}'^2)\}. \tag{5.15}$$

where $p[k] = ((p[l]+1) \bmod 2) + 1$.

Without loss of generality, let $\mathbf{s}$ be $(-1, -1, \cdots, -1)$. Since r_l follows a Gaussian distribution with mean $h_1(-b_0-b_1-b_2)$ and variance σ_1^2 at odd time instances, and mean $h_2(-b_0-b_1-b_2)$ and variance σ_2^2 at even time instances, $\rho = \sum_{l=1}^{d} 2h_{p[k]}r_{p[l]}(x_{p[l]} - x'_{p[l]})$ follows the Gaussian distribution as shown below:

$$\rho \sim \mathcal{N}(\sum_{l=1}^{d} 2h_{p[k]}^2 x_{p[l]}(x_{p[l]} - x'_{p[l]}), \sum_{l=1}^{d} 4h_{p[k]}^2\sigma_{p[k]}^2(x_{p[l]} - x'_{p[l]})^2). \tag{5.16}$$

Consequently, the first event probability can be written as

$$\begin{aligned} P_d &= \Pr\{\rho < \sum_{l=1}^{d} h_{p[k]}^2(x_{p[l]}^2 - x_{p[l]}'^2)\} \\ &= \frac{1}{\sqrt{2\pi\sigma_\rho^2}} \int_{-\infty}^{\sum_{l=1}^{d} h_{p[k]}^2(x_{p[l]}^2 - x_{p[l]}'^2)} e^{-\frac{(\rho - m_\rho)^2}{2\sigma_\rho^2}} d\rho, \end{aligned} \tag{5.17}$$

where

$$\begin{aligned} m_\rho &= \sum_{l=1}^{d} 2h_{p[k]}^2 x_{p[l]}(x_{p[l]} - x'_{p[l]}), \\ \sigma_\rho^2 &= \sum_{l=1}^{d} 4h_{p[k]}^2\sigma_{p[k]}^2(x_{p[l]} - x'_{p[l]})^2. \end{aligned}$$

From Theorem 7, $d = m+1$, $\sum_{l=1}^{m+1}(x_{p[l]} - x'_{p[l]})^2 = P$. Let $\rho' = \frac{\rho - m_\rho}{\sigma_\rho}$, we can rewrite P_d as

$$
\begin{aligned}
P_d &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\frac{\sum_{l=1}^{d} h^2_{p[k]}(x^2_{p[l]} - x'^2_{p[l]}) - m_\rho}{\sigma_\rho}} e^{-\frac{\rho'^2}{2}} d\rho' \\
&= Q\left(\frac{\sum_{l=1}^{m+1} h^2_{p[k]}(x_{p[l]} - x'_{p[l]})^2}{2\sqrt{\sum_{l=1}^{m+1} \sigma^2_{p[k]} h^2_{p[k]}(x_{p[l]} - x'_{p[l]})^2}} \right).
\end{aligned} \tag{5.18}
$$

We use $B_{d_{free}}$ to denote the number of competing paths of the correct path for s. Particularly for $m = 2$, it can be identified from the trellis in Fig. 5.3, $d = 3$ and $B_{d_{free}} = 1$. At high SNR region, the instantaneous bit error probability P_b for the 2-user system can thus be approximated by

$$
\begin{aligned}
P_b &\approx B_{d_{free}} P_d \\
&= Q\left(\frac{h_1^2 b_0^2 + h_2^2 b_1^2 + h_1^2 b_2^2}{\sqrt{h_1^2(b_0^2 + b_2^2)\sigma_1^2 + h_2^2 b_1^2 \sigma_2^2}}\right) \\
&\approx \frac{1}{2} e^{-\frac{1}{2}\frac{(h_1^2(b_0^2+b_2^2) + h_2^2 b_1^2)^2}{h_1^2(b_0^2+b_2^2)\sigma_1^2 + h_2^2 b_1^2 \sigma_2^2}} < \frac{1}{2} e^{-\frac{1}{2}\frac{h_1^2(b_0^2+b_2^2) + h_2^2 b_1^2}{\tilde{\sigma}^2}}.
\end{aligned} \tag{5.19}
$$

where $\tilde{\sigma} = \max(\sigma_1, \sigma_2)$.

The average bit error probability over the Rayleigh fading channel is upper bounded

by eq. (5.20).

$$\begin{aligned}\bar{P}_b &< \int_0^{+\infty}\int_0^{+\infty}\frac{1}{2}e^{-\frac{1}{2}\frac{(h_1^2(b_0^2+b_2^2)+h_2^2b_1^2)}{\tilde{\sigma}^2}}f_{h_1}(h_1)f_{h_2}(h_2)dh_1dh_2 \\ &= \frac{1}{2}\int_0^{+\infty}e^{-\frac{1}{2}\frac{h_1^2(b_0^2+b_2^2)}{\tilde{\sigma}^2}}f_{h_1}(h_1)dh_1\int_0^{+\infty}e^{-\frac{1}{2}\frac{h_2^2b_1^2}{\tilde{\sigma}^2}}f_{h_2}(h_2)dh_2 \\ &= \frac{2}{\left(\frac{(b_0^2+b_2^2)}{\tilde{\sigma}^2}+1\right)\left(\frac{b_1^2}{\tilde{\sigma}^2}+1\right)} < \frac{2\tilde{\sigma}^4}{(b_0^2+b_2^2)b_1^2}.\end{aligned} \tag{5.20}$$

We suppose $P = 1$. The diversity order D of the proposed scheme for the 2-user system can be calculated by

$$D \triangleq -\lim_{\sigma^2\to 0}\frac{\log \bar{P}_b}{\log\frac{1}{\tilde{\sigma}^2}} = 2, \tag{5.21}$$

which theoretically proves the proposed scheme can achieve the full diversity.

For the two-user system with m-regenerative braid codes, when two S-D channels are of the same average SNR, the average bit error probability can be approximated by

$$\bar{P}_b \approx \frac{2\tilde{\sigma}^4}{(\sum\limits_{\substack{i\in even\\ \leq i\leq m}} b_i^2+\tilde{\sigma}^2)(\sum\limits_{\substack{i\in odd\\ \leq i\leq m}} b_i^2+\tilde{\sigma}^2)}. \tag{5.22}$$

It is straightforward that under the same power constraint, the average bit error probability would decrease with m increases. Thus the full diversity order can be achieved for all $m \geq 1$[4].

For the M-user system, we consider the regenerative braid coding with memory size

[4]The $m = 1$ regenerative braid codes, which is the conventional superposition modulation scheme proposed by [63], can also achieve the diversity order of 2, as proved by [72].

m, and $m \geq M-1$. By the same mean, when $m = M-1$, the instantaneous bit error probability is expressed as

$$P_b \approx B_{d_{free}} P_d < \frac{1}{2} B_{d_{free}} e^{-\frac{1}{2}\frac{\sum_{i=0}^{m} b_i^2 h_{i+1}^2}{\tilde{\sigma}^2}}. \tag{5.23}$$

By integrating P_b on all the h_i's, we can obtain

$$D = -\lim_{\tilde{\sigma}^2 \to 0} \frac{\log \bar{P}_b}{\log \frac{1}{\tilde{\sigma}^2}} = M. \tag{5.24}$$

Therefore, the full diversity order can be achieved for the M-user system by using the $m \geq M-1$ regenerative codes. However, considering the coding gain and decoding complexity, we recommend to use a $m = M, M+1$ regenerative code for the M-user system; When M is large, a $m = M-1$ regenerative code is recommended.

5.5.3 Bit Error Rate Performance of Adaptive Transmission Scheme

In a general case, the transmission session would terminate early (the braid coding can not last to the prescribed N), and a new code session will start. We show the BER performance of the $m = 2$ regenerative code for the 2-user system in this section. We note here the theoretical BER performance of larger m cases can be derived by the same method used here, but the expression gets tediously complicated. For the $m = 2$ braid coding, the signal transmitted by each user may be composed by one, or two, or three information symbols. Let Z be the number of information symbols involved in the superposed signals transmitted by the source nodes, $Z = 1, 2, 3$. We use η_Z to indicate the stationary probability of transmitting a signal being composed by Z information

symbols at each user. We assume $P = 1$, the coefficients thus satisfy the equation $b_0^2 + b_1^2 + b_2^2 = 1$.

Fig. 5.4 shows the transition relationship between different states of the transmitted signals. Without loss of generality, we assume S_1 initiates the transmission, and sends $s_{1,1}$ to S_2. If $s_{1,1}$ is correctly decoded at S_2 with full power $P = 1$, S_2 transmits $\beta(b_1 s_{1,1} + b_0 s_{2,1})$ to D and S_1, where β is the power normalization factor. Otherwise, S_2 broadcasts $s_{2,1}$. In the former case, if S_1 can also decode the desired information $s_{2,1}$ with transmission power βb_0 successfully upon receiving the composed signals from S_2, it then transmits the three information symbols composed signal $(b_2 s_{1,1} + b_1 s_{2,1} + b_0 s_{1,2})$ to other nodes; Or it returns to the state $Z = 1$, which means it just broadcasts $s_{1,2}$. When one user works in the state $Z = 3$, the other one would continue to work in the state $Z = 3$ if it decodes the desired information with power b_0. If it fails to decode the new information, it restarts a transmission session.

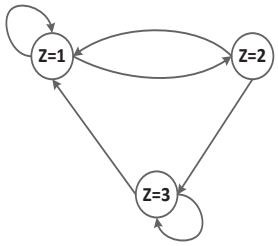


Figure 5.4: State diagram of the transmitted signal at each user

We assume the average error probability of decoding S_1's information at S_2 is equal to the error probability of decoding S_2's information at S_1. Hence, we can obtain

$$\begin{aligned} \eta_1 &= \eta_1 P_{S,1} + \eta_2 P_{S,2} + \eta_3 P_{S,3}, \\ \eta_2 &= \eta_1 (1 - P_{S,1}), \\ \eta_1 + \eta_2 + \eta_3 &= 1. \end{aligned} \tag{5.25}$$

where $P_{S,Z}$ is the decoding error probability at one user when the other one transmits signals superposed by Z information symbols.

Let $\bar{\gamma}_0 = \frac{1}{2\sigma_0^2}$. Solving these equations, we have

$$\begin{aligned}
\eta_1 &= \frac{P_{S,3}}{P_{S,3}(2-P_{S,1}) + (1-P_{S,1})(1-P_{S,2})}, \\
\eta_2 &= \frac{P_{S,3}(1-P_{S,1})}{P_{S,3}(2-P_{S,1}) + (1-P_{S,1})(1-P_{S,2})}, \\
\eta_3 &= \frac{(1-P_{S,1})(1-P_{S,2})}{P_{S,3}(2-P_{S,1}) + (1-P_{S,1})(1-P_{S,2})},
\end{aligned} \tag{5.26}$$

where

$$\begin{aligned}
P_{S,1} &= \frac{1}{2}(1 - \sqrt{\frac{\bar{\gamma}_0}{\bar{\gamma}_0 + 1}}), \\
P_{S,2} &= \frac{1}{2}(1 - \sqrt{\frac{\bar{\gamma}_0 \beta^2 b_0^2}{\bar{\gamma}_0 \beta^2 b_0^2 + 1}}), \\
P_{S,3} &= \frac{1}{2}(1 - \sqrt{\frac{\bar{\gamma}_0 b_0^2}{\bar{\gamma}_0 b_0^2 + 1}}).
\end{aligned} \tag{5.27}$$

The occurrence probability $P_{O,N'}$ of a code with a session size N' can be given by

$$P_{O,N'} = \begin{cases} \eta_1 P_{S,1}, & N' = 1, \\ \eta_2 P_{S,2}, & N' = 2, \\ \eta_2 (1-P_{S,2})(1-P_{S,3})^{(N'-3)}, & N' > 2, \end{cases} \tag{5.28}$$

The BER can be computed as the sum of the bit error probability $P_{b,N'}$ weighted by

the occurrence probability, which is expressed as

$$P_b = \sum_{N'=1}^{2N} P_{O,N'} P_{b,N'}. \tag{5.29}$$

For $N' = 1$, the destination only receives one signal containing the transmitted information symbol, which means the braid coding is on the initialization stage. The corresponding BER at the destination is

$$P_{b,1} = Q(\sqrt{2\gamma_1}) = Q(h_1/\sigma_1). \tag{5.30}$$

For $N' = 2$, $s_{1,1}$ and $s_{2,1}$ are transmitted. Let $\tilde{\sigma}_2^2 = h_2^2\beta^2 b_0^2 + \sigma_2^2$. Then the error probability can be expressed as

$$\begin{aligned} P_{b,2} = \Pr\{&(r_{1,1} - h_1 s'_{1,1})^2 + (r_{2,1} - h_2(\beta b_1 s'_{1,1} + \beta b_0 s'_{2,1}))^2 \\ &< (r_{1,1} - h_1 s_{1,1})^2 + (r_{2,1} - h_2(\beta b_1 s_{1,1} + \beta b_0 s_{2,1}))^2\}. \end{aligned} \tag{5.31}$$

Case I: The error probability of $s_{1,1}$ can be approximated by

$$P_I = Q\left(\sqrt{\frac{\frac{h_1^4}{\sigma_1^4} + \frac{h_2^4\beta^4 b_1^4}{\tilde{\sigma}_2^4}}{\frac{h_1^2}{\sigma_1^2} + \frac{h_2^2\beta^2 b_1^2}{\tilde{\sigma}_2^2}}}\right) = Q\left(\sqrt{\frac{h_1^4\tilde{\sigma}_2^4 + h_2^4\beta^4 b_1^4\sigma_1^4}{h_1^2\sigma_1^2\tilde{\sigma}_2^4 + h_2^2\beta^2 b_1^2\sigma_1^2\tilde{\sigma}_2^2}}\right) \tag{5.32}$$

Case II: We calculate the error probability of $s_{2,1}$

1: If $s_{1,1}$ is successfully decoded, $s_{1,1}$ can be subtracted from $r_{2,1}$. It is equivalent to decode $s_{2,1}$ upon having $r_{2,1} - h_2\beta_1 b_1 s_{1,2}$. Thus the BER in this scenario is $P_{II,1} =$

$Q(\frac{h_2\beta b_0}{\sigma_2})$.

2: The destination fails to decode both $s_{1,1}$ and $s_{2,1}$. Under this case, we simplify eq. (5.31), and obtain

$$\begin{aligned} P_{II,2} &= \Pr\{r' = h_1 r_{1,1} + h_2 r_{2,1}\beta b_1 < 0\} \\ &= Q\left(\frac{h_1^2 + h_2^2\beta^2 b_1(b_1+b_0)}{\sqrt{h_1^2\sigma_1^2 + h_2^2\beta^2 b_1^2\sigma_2^2}}\right). \end{aligned} \tag{5.33}$$

When the SNR's of the source to destination channels are large, the decoding error probability $P_{b,2}$ can be approximated by summing the probabilities of all the different cases,

$$P_{b,2} = P_I(1 - P_{II,2}) + (1 - P_I)P_{II,1} + 2P_I P_{II,2} \tag{5.34}$$

For $N' > 2$, the braid coding will terminate when $N' = 2N$. We approximate $P_{b,N}$ using the BER of the inter-user channel of good quality case in eq. (5.19) .

The instantaneous BER is thus approximated by

$$P_b(h_1,h_2) \approx P_{O,1}Q\left(\frac{h_1}{\sigma_1}\right) + P_{O,2}P_{b,2} + (1 - P_{O,1} - P_{O,2})Q\left(\frac{h_1^2 b_0^2 + h_2^2 b_1^2 + h_1^2 b_2^2}{\sqrt{h_1^2(b_0^2+b_2^2)\sigma_1^2 + h_2^2 b_1^2\sigma_2^2}}\right). \tag{5.35}$$

The average BER under Rayleigh fading channel can be calculated using the numer-

ical integration as

$$\bar{P}_b = \int_0^{+\infty} \int_0^{+\infty} P_b(h_1, h_2) f(h_1) f(h_2) dh_1 dh_2, \tag{5.36}$$

where

$$f(h_1) = 2h_1 e^{-h_1^2}, f(h_2) = 2h_2 e^{-h_2^2}. \tag{5.37}$$

5.6 Forward and Backward Message Passing Iterative Decoding

In this section, we consider a scenario, where both the source nodes are equipped with soft-decodable channel codes. A forward and backward message passing iterative decoder is designed in this section for superposition modulation based on the regenerative braid coding. The extrinsic information is not only exchanged between signal detectors and channel decoders in every information block, but also between different information blocks in the proposed decoding method. The iterative decoder proposed here is different from that in [75], which considers the $m = 1$ regenerative code, and just performs the forward iterative decoding. By applying both the forward and the backward message passing iterative decoding, we remedy the knowledge imbalance of the a prior LLR information of the transmitted codewords, which contributes to the performance improvement.

Since each source node is encoded, the system setting is rephrased here. We use a codeword package $\mathbf{s}_{i,k} = (s_{i,k}^1, s_{i,k}^2, ...s_{i,k}^{N_C})$ of length N_C to denote the signals transmitted

from the source node S_i at k th time slice. Correspondingly, the destination node receives the signal $\mathbf{r}_{i,k}$. For simplicity, we consider the decoding of the $\mathbf{s}_{1,k}$ for the $m = 2$ regenerative braid coding. Suppose it is not one of the last m transmitted information in one transmission session, then it is involved in three successively transmitted signals $\mathbf{r}_{1,k}$, $\mathbf{r}_{2,k}$ and $\mathbf{r}_{1,k+1}$, where

$$\begin{aligned}
\mathbf{r}_{1,k} &= h_{1,k}(b_2\mathbf{s}_{1,k-1} + b_1\mathbf{s}_{2,k-1} + b_0\mathbf{s}_{1,k}) + \mathbf{z}_{1,k}, \\
\mathbf{r}_{2,k} &= h_{2,k}(b_2\mathbf{s}_{2,k-1} + b_1\mathbf{s}_{1,k} + b_0\mathbf{s}_{2,k}) + \mathbf{z}_{2,k}, \\
\mathbf{r}_{1,k+1} &= h_{1,k}(b_2\mathbf{s}_{1,k} + b_1\mathbf{s}_{2,k} + b_0\mathbf{s}_{1,k+1}) + \mathbf{z}_{1,k+1}.
\end{aligned} \tag{5.38}$$

Upon receiving the signals $\mathbf{r}_{1,k}$, $\mathbf{r}_{2,k}$ and $\mathbf{r}_{1,k+1}$, D performs a forward message passing iterative decoding. The destination first calculates the LLR's of $\mathbf{s}_{1,k}$ and $\mathbf{s}_{2,k-1}$ using the elementary iterative decoder shown in Fig. 5.5. It exchanges information between signal detectors and channel decoders. Then the LLR of $\mathbf{s}_{1,k}$ and the renewed LLR of $\mathbf{s}_{2,k-1}$ are fed to the input of the elementary iterative decoder of the next codeword $\mathbf{s}_{2,k}$, as the a prior information; after decoding of one session ends, D performs the backward message passing iterative decoding. It decodes from the signal received in the last time slice, and forwards the a prior information of signals to the elementary decoder of the previous transmitted codeword. The details of the decoding steps are described below:

Step 1: forward message passing

In the forward message passing decoding, $L^a_{1,k-1}$ and $L^a_{2,k-1}$, the a prior information of $\mathbf{s}_{1,k-1}$ and $\mathbf{s}_{2,k-1}$ (red dash input) from the previous forward decoding step, are fed into detector 1. Though the received signal $\mathbf{r}_{2,k}$ contains the information of $\mathbf{s}_{2,k-1}$, $\mathbf{s}_{1,k}$ and $\mathbf{s}_{2,k}$, just $L^a_{2,k-1}$ is known. The a prior information of $\mathbf{s}_{1,k}$ and $\mathbf{s}_{2,k}$ is set to zero for

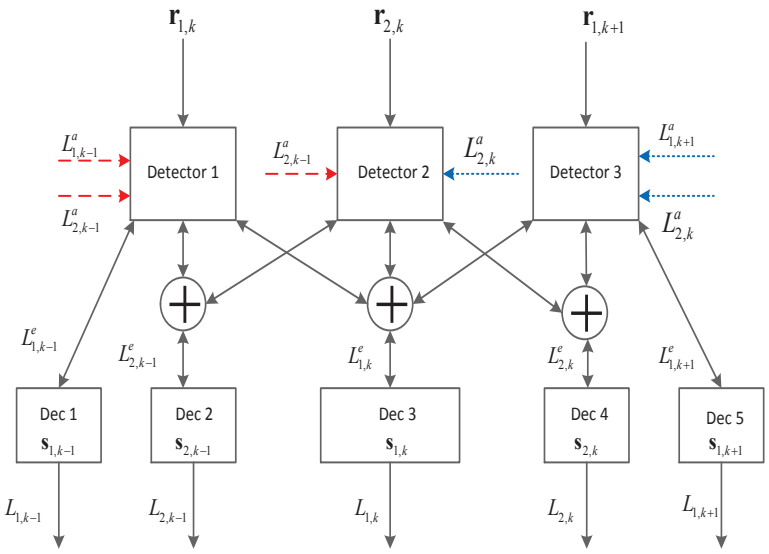


Figure 5.5: An elementary decoder structure

detector 2. For the same reason, all the a prior information is set to zero for detector 3.

Assuming $\eta_{1,k} = h_{1,k}(b_2\mathbf{s}_{1,k-1} + b_1\mathbf{s}_{2,k-1}) + \mathbf{z}_{1,k}$, we can express the LLR of $\mathbf{s}_{1,k}$ from $\mathbf{r}_{1,k}$ as

$$L^{D_1}_{1,k} = \frac{2h_{1,k}b_0}{\text{var}(\eta_{1,k})}[\mathbf{r}_{1,k} - E(\eta_{1,k})]. \tag{5.39}$$

where $E(\eta_{1,k})$ and var$(\eta_{1,k})$ can be calculated by

$$\begin{aligned} E(\eta_{1,k}) &= h_{1,k}(b_2E(\mathbf{s}_{1,k-1}) + b_1E(\mathbf{s}_{2,k-1})) \\ &\approx h_{1,k}(b_2tanh(L^a_{1,k-1}/2) + b_1tanh(L^a_{2,k-1}/2)), \\ \text{var}(\eta_{1,k}) &\approx h^2_{1,k}(b_2^2(1 - tanh^2(L^a_{1,k-1}/2)) + b_1^2(1 - tanh^2(L^a_{2,k-1}/2))) + \sigma^2_{1,D}. \end{aligned} \tag{5.40}$$

$L^{D_2}_{1,k}$ and $L^{D_3}_{1,k}$, the LLR's of $\mathbf{s}_{1,k}$ from $\mathbf{r}_{2,k}$ and $\mathbf{r}_{1,k+1}$, can be computed by the same mean. The sum of all the LLR's from the three signals, denoted by $L^D_{1,k}$, is fed into the

decoder of $\mathbf{s}_{1,k}$ (Dec 3). After the soft decoding at decoder 3, we get the refreshed LLR $L_{1,k}$ of $\mathbf{s}_{1,k}$. Then the extrinsic information $L_{1,k}^{e_i}$, which can be expressed as

$$L_{1,k}^{e_i} = L_{1,k} - L_{1,k}^{D_i}, \tag{5.41}$$

where $i = 1, 2, 3$, is fed back to detector i to improve the a prior information of the detectors for the next detection.

At the same time, the LLR's of $\mathbf{s}_{1,k-1}$, $\mathbf{s}_{2,k-1}$, $\mathbf{s}_{2,k}$ and $\mathbf{s}_{1,k+1}$ are also forwarded to the corresponding decoders from detectors 1-3. The decoders calculate the extrinsic information of the corresponding codewords. Then, the a prior information of all the codewords is updated. After several iterations, the extrinsic information $L_{2,k-1}^{e}$ and $L_{1,k}^{e}$ are forwarded to the elementary decoder as the a prior information for decoding the next codeword.

It should be mentioned here a flag bit [63] is embedded in each codeword to indicate the cooperation state. If a new session starts, the flag bit is set to zero. The decoding of a codeword involved in three successive transmissions is shown here, but the decoding of a codeword involved in just one, or two transmission is just a simpler case of the decoding process described above.

Step 2: backward message passing

The drawback of the forward iterative decoding is the lack of the a prior information of codewords from the next half time slices. For instance, $L_{2,k}^{a}$ and $L_{1,k+1}^{a}$ are unknown in the previous forward message passing decoder. The decoding process does not make effective use of all the LLR's from the received signals after $\mathbf{s}_{1,k}$. To remedy this drawback, a backward message passing decoder is used here, in which the decoding process

is performed from the last received codeword to the first codeword after the forward decoding is completed. As shown in Fig. 5.5, the a prior information $L^a_{2,k}$ and $L^a_{1,k+1}$ (blue dot line) is forwarded back to the elementary decoder to estimate the LLR of $L^a_{1,k}$, with the updated a prior information of $\mathbf{s}_{1,k-1}$ and $\mathbf{s}_{2,k-1}$ (red dash input). The elementary decoding procedure is the same as that in forward iterative decoding, but with more a prior information.

The complexity of the overall iterative decoding method is linear to the complexity of just performing forward decoding. But the performance will be improved by taking a full advantage of all the LLR's of the received signals.

5.7 Simulation Results

In this section we evaluate the BER performance of the proposed braid coding cooperative scheme under various scenarios. Our simulation employs independent quasi-static flat Rayleigh fading, or AWGN, for all the channels. CSI is assumed to be available at the respective receivers. For fair comparison, all the performance evaluation is tested with the same average transmission power. Unless otherwise stated, the braid codes use the derived optimal weights.

Test case 1 (Optimal weight and constraint length): We assume the source nodes adopt BPSK modulation [5]. Four types of proposed braid codes, nonregenerative and regenerative with $m = 1, 2, 3$ with different weights, are compared with the superposi-

[5]When the source nodes adopt QPSK modulation (2-D modulation), the same performance as BPSK modulation can be observed, since QPSK is two-dimensional BPSK. We do not show the performance of QPSK here.

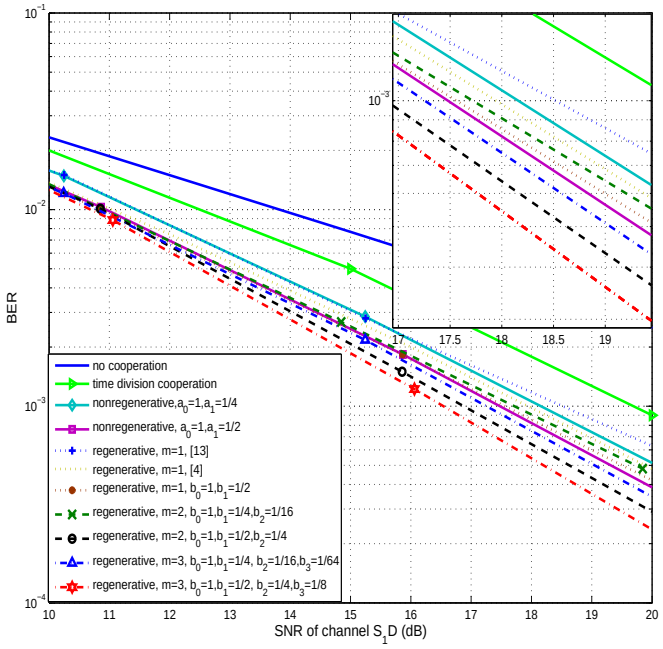


Figure 5.6: BER vs. SNR (dB) of S_1-D channel for 2-user braid coding cooperative systems with different weights, $m = 1, 2, 3$, $N = 4$, Rayleigh fading, $\text{SNR}_{\mathsf{S}_1\mathsf{D}} = \text{SNR}_{\mathsf{S}_2\mathsf{D}}$, $\text{SNR}_{\mathsf{S}_1\mathsf{S}_2} = \text{SNR}_{\mathsf{S}_1\mathsf{D}} + 20\text{dB}$.

tion modulation scheme using the best coefficients[6] from [63] and [71] under Rayleigh fading channel in Fig. 5.6. The BER of non-cooperative scheme and time-division cooperation (where each user uses 4ASK and spares half of its time to relay the other user's data) are also included. Obviously, non-cooperative scheme shows no diversity gain. Since channel CSIs remain constant in a session, time-division achieves the same diversity order of 2 as braid coding, but as we will see from simulations, it falls short in power gain. Though all the braid coding schemes can reach full diversity gain 2, braid coding schemes with our optimal weight coefficients and larger constrain length performs better than the original proposed superposition modulation scheme in [63] [71]. As indicated by Theorem 7 and Corollary 1, a larger m leads to a larger d_{free}, but the coding gain

[6]The performance of [72] is similar to [63]. For the sake of figure clarity, the curve simulated using coefficients in [72] is not shown here.

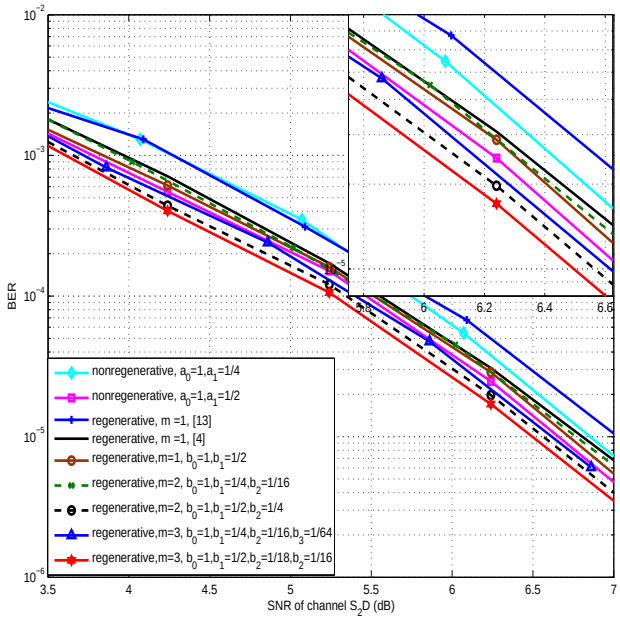


Figure 5.7: BER vs. SNR (dB) of S_1-D channel for 2-user braid coding cooperative systems with different weights, $m = 1, 2, 3$, $N = 4$, AWGN, $\text{SNR}_{\mathsf{S}_1\mathsf{D}} = \text{SNR}_{\mathsf{S}_2\mathsf{D}} + 4$dB, $\text{SNR}_{\mathsf{S}_1\mathsf{S}_2} = \text{SNR}_{\mathsf{S}_2\mathsf{D}} + 6$dB.

quickly hits a diminishing return for $m \geq 3$. Considering the decoding complexity, we recommend $m = 2$ or $m = 3$ in practical 2-user cooperative systems. We also test the proposed scheme with different m and weights under AWGN channel in Fig. 5.7. The two source-destination channels are of different quality. There is still performance gain, enabled by braid coding with our proposed optimized weights, but the gain is smaller than the fading channel case. This is because cooperative coding brings in diversity gain for block fading channels, whereas each AWGN channel already provides a diversity order of $+\infty$, and hence cooperation cannot bring in any more diversity gain.

Test case 2(Constraint length and session length): We evaluate the impact of session length N for braid coding in Fig. 5.8. In all the four braid codes tested, we have found that the system improves as N increases from 2 to 4, but beyond 4 the gain is simply not

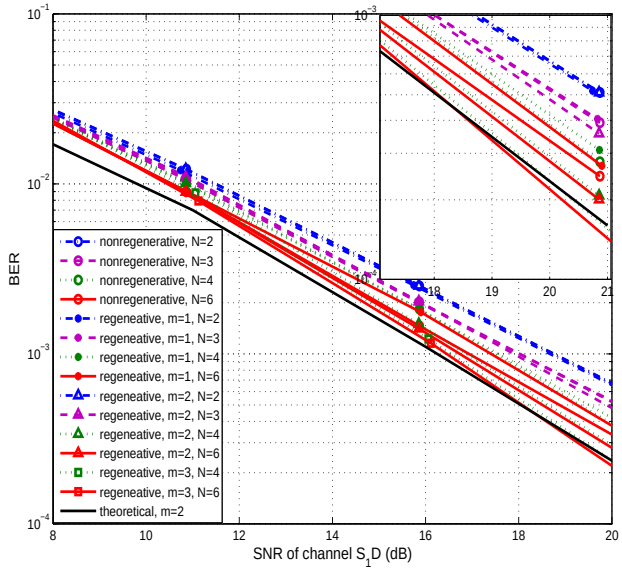


Figure 5.8: BER vs. SNR (dB) of S_1-D channel for 2-user braid coding cooperative systems with different block lengths, $m = 1, 2, 3$, Rayleigh fading, $\text{SNR}_{\mathsf{S}_1\mathsf{D}} = \text{SNR}_{\mathsf{S}_2\mathsf{D}}$, $\text{SNR}_{\mathsf{S}_1\mathsf{S}_2} = \text{SNR}_{\mathsf{S}_1\mathsf{D}} + 20\text{dB}$.

perceivable. Theoretical BER derived in Section 5.5 is also plotted in this figure. We see that the theoretical BER is lower than the simulated performance, because for $Z > 3$ we approximate the BER by the BER with large N. The gap between the theoretical and simulated results become smaller when the SNR of the inter-user channel increases.

Test case 3 (Braid coding in multiuser systems): The BER performance of the braid coding cooperative scheme applied to the 3-to-1 system and the 4-to-1 system is depicted in Fig. 5.9 and Fig. 5.10, respectively. Assume all the inter-user channels have the same average SNR. Round robin scheduling scheme is used by all the source nodes. In Fig. 5.9, it is shown that the $m = 2$ braid coding cooperative scheme is able to achieve more diversity gain than the conventional $m = 1$ scheme in [63]. The curve of the BER performance of the $m = 2$ braid coding has almost the same slope with the curve of diversity order 3 when the channel SNR is high. In addition, we also evaluate several other choices of weights for braid coding, and in each case, it is clear that the cases

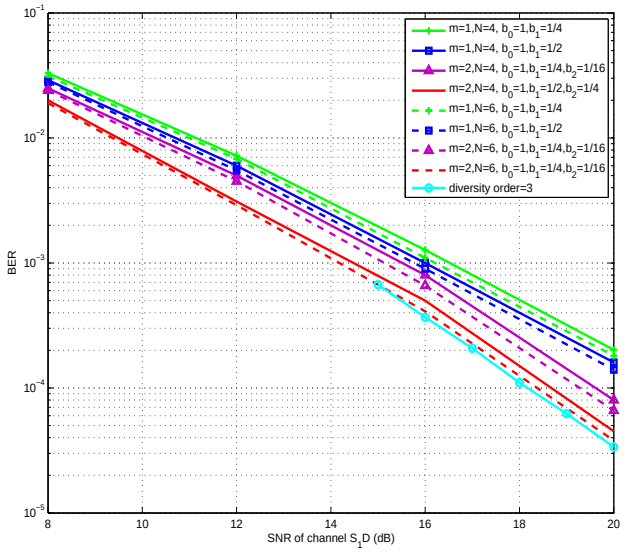


Figure 5.9: BER vs. SNR (dB) of S_1-D channel for 3-user braid coding cooperative systems with different weights, $m = 1, 2$, Rayleigh fading, $\text{SNR}_{\mathsf{S}_1\mathsf{D}} = \text{SNR}_{\mathsf{S}_2\mathsf{D}} = \text{SNR}_{\mathsf{S}_3\mathsf{D}}$, $\text{SNR}_{\mathsf{S}_1\mathsf{S}_2} = \text{SNR}_{\mathsf{S}_1\mathsf{S}_3} = \text{SNR}_{\mathsf{S}_2\mathsf{S}_3} = \text{SNR}_{\mathsf{S}_1\mathsf{D}} + 15\text{dB}$.

with the optimal weights we derived outperforms the others. When the session length grows from 4 to 6, there is still some performance gain, but not much. The same trend is also identified in the four-user system. The Fig. 5.9 and Fig. 5.10 show the achievable diversity order increases as M increases. Since the decoding complexity increases exponentially with the memory size m, the best performance-complexity trade-off is to choose $m = M - 1$ when M is large. For regenerative codes, the decoding complexity is linear with N, and hence per-bit complexity is constant, irrespective of N. For an M-user system, increasing N from 1 up to M would noticeably improve the performance, but the gain becomes marginal or almost imperceivable as N continues to increase.

Test case 4 (Detectors for the nonregenerative braid coding): Performance of different detectors for the nonregenerative braid coding is shown in Fig. 5.11. The Linear detector, MMSE detector, and PE detector are compared with the ML detector. It shows that the MMSE detector performs much better than the linear detector. Since inversing

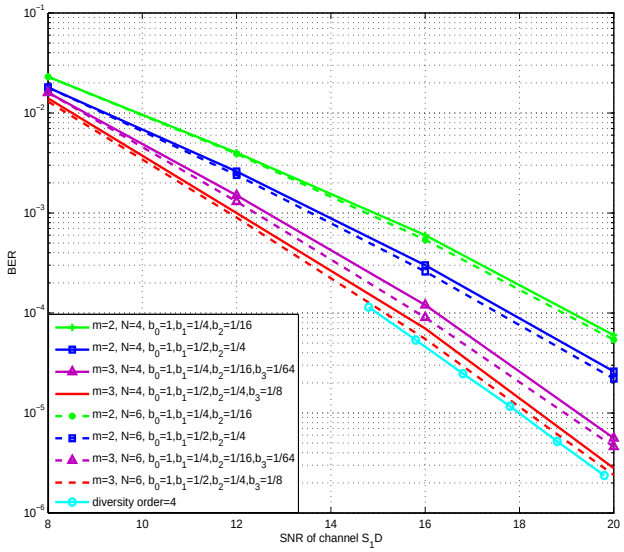


Figure 5.10: BER vs. SNR (dB) of S_1-D channel for 4-user braid coding cooperative systems with different weights, $m = 2, 3$, Rayleigh fading, SNRs of all channels $S_i D$ are equal, SNR of the inter-user channels $\text{SNR}_{S_i S_j} = \text{SNR}_{S_i D} + 15dB$.

matrix would bring many computation complexity, the BER performance of the PE detector is also evaluated. By increasing K, the performance of the PE detector approaches that of the MMSE detector. The ML detector performs the best, but the decoding complexity of the ML detector grows exponentially with the session length.

Test case 5 (Iterative decoding for the channel-coded system): Finally we compare the proposed iterative decoding scheme with the forward decoding scheme proposed in [75]. Suppose source nodes are encoded by convolutional code $(1, 1/1 + D)$. We simulate the $m = 2$ braid coding case here. The number of iterations of the elementary decoder is set to 5, or 8. The simulation result shows that, when the number of iterations increases, the BER performance of our proposed scheme improves. The proposed iterative scheme outperforms the conventional one by almost 1 dB with 8 iterations .

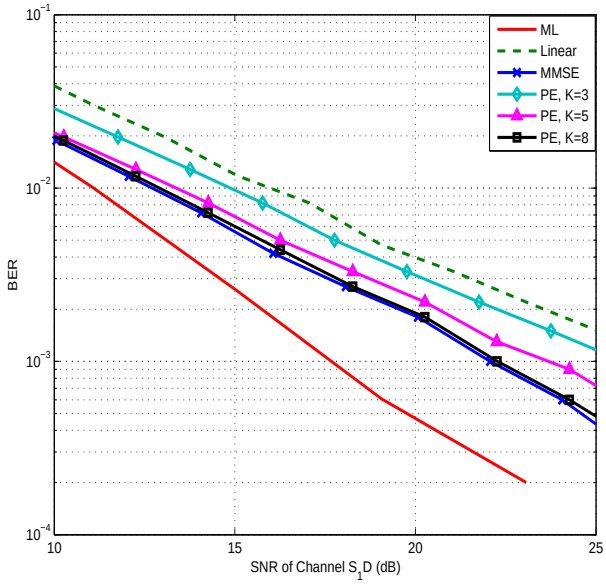


Figure 5.11: BER vs. SNR (dB) of S_1-D channel for the 2-user nonregenerative braid coding cooperative systems with ML, Linear, MMSE, and PE detectors, $N = 4$, Rayleigh fading, $\text{SNR}_{S_1D} = \text{SNR}_{S_2D}$, $\text{SNR}_{S_1S_2} = \text{SNR}_{S_1D} + 20\text{dB}$.

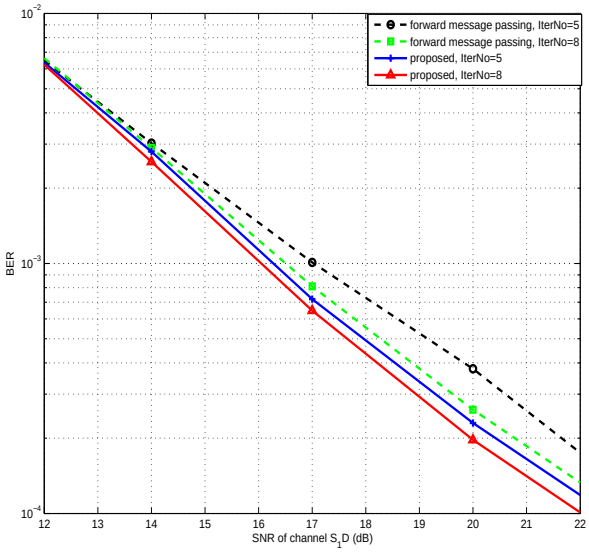


Figure 5.12: BER vs. SNR (dB) of S_1-D channel for 2-user braid coding cooperative systems with different decoding methods and numbers of iterations, source encoded by $(1, 1/1 + D)$, $m = 2$, $N = 4$, Rayleigh fading, $\text{SNR}_{S_1D} = \text{SNR}_{S_2D}$, $\text{SNR}_{S_1S_2} = \text{SNR}_{S_1D} + 15\text{dB}$.

Chapter 6

Conclusions

This dissertation presents our research on transmission and estimation problems in wireless relay network. Efficient forward strategies and distributed coding schemes are proposed.

In Chapter 2, a new forwarding strategy, termed analog-encode-forward (AEF), with ML decoder and MAP decoder is proposed in this chapter. The key idea is to leverage the recent advances in analog codes to perform soft-message forwarding at the R-D transmission. A mirrored baker's map code is discussed in detail, including its encoder, decoder and utilization in user cooperation. The destination uses the ML analog decoder to retrieve the message. In AEF with MAP decoder scheme, a MAP analog decoding algorithm is designed specifically for the relay systems. The MAP decoder turns out to be extremely simple, and brings additional coding gain. Our new soft message relaying strategies get some improvement when compared to former practical relaying schemes.

In Chapter 3, we have studied the problem of soft-forward for 2-hop relay networks

with multiple parallel relays. The primary contribution is the proposition of a new relaying strategy termed Z-forward. The forwarded soft messages form a three-segment piece-wise linear function of the signal LLRs, whose simplicity allows us to compute its exact pdf and to subsequently formulate the end-to-end BER. We show that Z-forward subsumes the traditional AF, DF, and PF schemes as its special case, and with optimal thresholds, the performance can be considerably improved. Two sub-optimal Z-forward schemes are also proposed to reduce the complexity. In a single relay network, optimized Z-forward performs on par with the previously-proposed $\tanh$-forward/EF scheme, and both are practically optimal. However, in a multi-relay network, the new scheme is noticeably better by around 1 dB on block fading channels, and the gain is more prominent with more relays. All the Z-forward schemes can reach a full diversity order, with either MRC and ML estimation. The techniques developed in this paper are for BPSK systems. An interesting future direction would be to consider networks with possible high-order modulation, and to develop extensions of the Z-forward for models equipped with powerful channel coding.

In Chapter 4, we proposed a new soft-encoding distributed coding scheme for parallel relay systems. Unlike the previous work that favors the hyperbolic tangent of one half of the LLR values (soft estimate of the received signals), here we argue that range-limited LLR serves as a better soft message representation in general, and is particularly suitable for soft-encoding at each relay node. Based on this, we specifically developed a simple but rather powerful framework of soft-input soft-output encoding scheme for parallel relay systems. We presented the general idea of encoding and protecting these soft messages, and discussed in detail the application of distributed convolutional codes and distributed turbo codes. For the former, two ML-Viterbi decoding algorithms were developed (one with Gaussian approximation), and for the latter, a BCJR decoding al-

gorithm was developed. Comparison of our new codes with the existing hard-encoding digital codes and a previously proposed soft-encoding "tanh" code reveals an encouraging performance gain on both AWGN and fading channels.

In Chapter 5, we propose a generalization of cooperative scheme through superposition modulation based on braid coding. Two subclasses, regenerative and nonregenerative braid codes, are considered, and the optimal weights and optimal memory-sizes are designed to achieve the power gain, as well as the diversity gain, with a full rate. Although our discussion here focuses on the 2-user case, the proposed scheme easily generalizes to the M-user cases. We theoretically prove that our proposed scheme is able to reach the full diversity order for the M-user system. At the destination, a Viterbi decoder of linear complexity to the length of the information length is especially designed for the regenerative braid coding; several linear detectors are compared for the nonregenerative braid coding. An iterative decoder for coded systems based on the regenerative braid coding is designed, which performs better than the conventional iterative decoder, by making effective use of all the LLR information of the received signals.

Bibliography

[1] E. C. van der Meulen, "Three-terminal communication channels," *Adv. Appl. Prob.*, vol. 3, pp. 120-154, 1971.

[2] T. M. Cover and A. A. El Gamal, "Capacity theorems for the relay channel," *IEEE Trans. Info. Theory*, vol. IT-25, pp. 572-584, Sept. 1979

[3] X. Bao and J. Li, "Efficient Message Relaying for Wireless User Cooperation: Decode-Amplify-Forward (DAF) and Hybrid DAF and Coded-Cooperation," *IEEE Trans. Wireless Commun.*, pp.3975-3984, Oct. 2004.

[4] P. Weitkemper, D. Wubben, and K.-D. Kammeyer, "Minimum MSE relaying in coded networks," *Intl. ITG Workshop Smart Antennas*, pp 96-103, Vienna, Feb 2008.

[5] R. Hoshyar, and R. Tafazolli, "Soft decode and forward of MQAM modulations for cooperative relay Channels," *IEEE Vehicular Tech. Conf (VTC-Spring)*, pp 639-643, May 2008

[6] Y. Li, B. Vucetic, T. F. Wong, and M. Dohler,"Distributed turbo coding with soft information relaying in multihop relay networks," *IEEE J. Selected Areas Commun.*, pp 2040-2050, Oct. 2006.

[7] X. Bao, and J. Li, "Adaptive network coded cooperation for wireless relay networks: Matching code-on-graph with network-on-graph," *IEEE Trans Wireless Commun.*, pp 574-583, Feb. 2008.

[8] T. G. Marshall, Jr.. "Real number transform and convolutional codes," *Proc. 24th Midwest Symp. Circuits Sys.*, Editor: S. Kame, Albuquerque, NM, June 29-30, 1981

[9] J. K. Wolf, "Analog codes," *IEEE Intl. Conf. Comm,* Boston, MA, USA, Vol. 1,June, 1983, pp. 310-312.

[10] B. Chen and G. W. Wornell, "Analog error-correcting codes based on chaotic dynamical systems," *IEEE Trans Commun.*, vol.46, pp.881-890, Jul. 1998.

[11] X. Kai and J. Li, "Chaotic analog error correction codes: The mirrored baker's codes," *IEEE Global Commun. Conf. (GLOBECOM)* , pp.1-5, Oct. 2010.

[12] M. Yu and J. Li, "Is amplify-and-forward practically better than decode-and-forward or vice versa?" *IEEE Intl. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 1520-6149, March 2005.doi:10.1109/ICASSP.2005.1415722

[13] M. Fu, "Stochastic analysis of turbo decoding," *IEEE Trans. Info. Theory*, vol. 51, pp. 81-100, Jan. 2005

[14] X. Bao, and J. Li, "On accuracy of Gaussian assumption in iterative analysis for LDPC codes," *Proc. IEEE Intl. Symp. Info. Theory.*,2006.

[15] A. Sendonaris, E. Erkip and B. Aazhang, "User cooperation diversity-Part I: system description", *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1927-1938, Nov. 2003.

[16] A. Sendonaris, E. Erkip and B. Aazhang, "User cooperation diversity-Part II: implementation aspects and performance analysis", *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1939-1948, Nov. 2003.

[17] J. N. Laneman, D. Tse and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Trans. Info. Theory*, vol.50, pp.3062-3080, Dec. 2004.

[18] M. Yu, J. Li and H. Sadjadpour, "A geometry-inclusive analysis for single-relay systems", *International J. of Digital Multimedia Broadcasting*, March 2009.

[19] K. S. Gomadam and S. A. Jafar, "Optimal relay functionality for SNR maximization in memoryless relay networks," *IEEE J. Selected Areas Commun.*, vol. 25, pp. 390-401, Feb. 2007.

[20] A. Ribeiro, X. Cai and G. B. Giannakis, "Symbol error probabilities for general cooperative links", *IEEE Trans. Wireless Commun.*, vol. 4, no. 3, pp. 1264-1273, May 2005.

[21] T. Liu, L. Song, Y. Li, Q. Huo and B. Jiao , "Performance analysis of hybrid relay selection in cooperative wireless system", *IEEE Trans. Commun.*,vol. 60, no. 3, pp. 779-788, Mar. 2012.

[22] F. A. Onat, A. Adinoyi, Y. Fan, H. Yanikomeroglu, J. Thompson, and I. Marsland, "Threshold selection for SNR-based selective digital relaying in cooperative wireless networks", *IEEE Trans. Wireless Commun.*, vol. 7, no. 11, pp. 4226-4237, Nov. 2008.

[23] F. A. Onat, Y. Fan, H. Yanikomeroglu and J. Thompson , "Asymptotic BER Analysis of Threshold Digital Relaying Schemes in Cooperative Wireless Systems", *IEEE Trans. Wireless Commun.*,vol. 7, no. 12, pp. 4938-4247, Dec. 2008.

[24] T. Kim and D. I. Kim, "Hybrid hard/soft decode-and-forward relaying protocol with distributed turbo code", *Springer Computational Sci. and Its App. (ICCSA)*, Berlin, 2009.

[25] X. Zhang, M. Hasna and A. Ghrayed, "Performance analysis of relay assignment schemes for cooperative networks with multiple source-destination pairs", *IEEE Trans. Wireless Commun.*, vol. 11, no. 1, pp. 166-177, Jan. 2012.

[26] I. Abou-Faycal, M. Medard, "Optimal uncoded regeneration for binary antipodal signaling", in *Proc. IEEE Int. conf. Comm. (ICC)*, Paris, France, June, 2004.

[27] H H Sneesens, L Vandendorpe, "Soft decode and forward improves cooperative communications", in *IEEE Int. Workshop on Com. Adv. in Multi-Sensor Adap. Processing*, Puerto Vallarta, Dec., 2005.

[28] P. Weitkemper, D. Wubben, V. Kuhn and K. Kammeyer, "Soft information relaying for wireless networks with error-prone source-relay link", in *Int. ITG Conf. on SCC*, Germany, Jan., 2008.

[29] M. H. Azmi, J. Li, J. Yuan, and R. Malaney, "LDPC codes for soft decode-and-forward in half-duplex relay channels," *IEEE J. Selected Areas Commun.*, vol. 31, no. 8, pp. 1402-1413, Aug., 2013.

[30] X. Lu, J. Li and Y. Liu, "A New Forwarding strategy for Wireless Relay Channels: Analog-Encode-and-Forward(AEF)," *Information Sciences and Systems (CISS), Annual Conference on*, Princeton, USA, March. 2012.

[31] S. Tian, Y. Li, and B. Vucetic, "Piecewise-and-forward relaying in wireless relay networks", *IEEE Signal Process. Lett.* ,vol. 18, no. 5, pp. 323-326, May 2011.

[32] T. Wang, A. Cano, G. B. Giannakis, and J. N. Laneman, "High performance cooperative demodulation with decode-and-forward relays", *IEEE Trans. Commun.*, vol. 55, no. 7, pp. 1427-1438, July 2007.

[33] H. Poor, *An Introduction to Signal Detection and Estimation*, 2nd ed., New York: Springer, 1998, pp. 173-184.

[34] H. El Gamal, and A. R. Hammons, "Analyzing the turbo decoder using the gaussian approximation", *IEEE Trans. Info. Theory*, vol. 47, no. 2, pp. 671-686, Feb. 2001.

[35] H. Fayed, and A. Atiya, "An evaluation of the integral of the product of the error function and the normal probability density, with application to the bivariate normal intergral", *Math. of computation*.

[36] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press 2004.

[37] J. N. Laneman and G. W. Wornell "Energy-Efficient antenna sharing and relaying for wireless networks", *IEEE Wireless Communications and Networking Conf. (WCNC)*, New Orleans, USA, Mar. 2005.

[38] X. Lu, J. Li, and Y. Liu, "A parametric approach to optimal soft signal relaying in wireless parallel-relay systems", *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May. 2014.

[39] D. Chen and J. N. Laneman, "Modulation and demodulation for cooperative diversity in wireless systems," *IEEE Trans. Wireless Commun.*, vol. 5, no. 7, pp.1785-1794, July 2006.

[40] S. Bouanen, H. Boujemaa, and W. Ajib, "Threshold-based adaptive decode-amplify-forward relaying protocol for cooperative systems", *IEEE Wireless Communications and Mobile Computing Conference (IWCMC)*, Istanbul, Turkey, July 2011.

[41] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity. Part I and Part II," *IEEE Trans. on Comm.*,vol. 51, no. 11, pp. 1927-1948, Nov. 2003.

[42] H.H. Sneessens and L. Vandendorpe, "Soft decode and forward improves cooperative communications," *1st IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, Puerto Vallarta, Mexico, Dec. 2005, pp. 157 - 160.

[43] M. A. Karim, T. Yang, J. Yuan, Z. Chen, and I. Land, "A Novel Soft Forwarding Technique for Memoryless Relay Channels Based on Symbol-wise Mutual Information," *IEEE Commun. Lett.*, vol. 14, no. 10, pp. 927-929, Oct.2010.

[44] M. A. Karim, J. Yuan, Z. Chen, and J. Li, "Soft Information Relaying in Fading Channels," *IEEE Commun. Lett.*, vol. 1, no. 3, pp. 233-236, Jun.2012.

[45] P. Weitkemper and G. Dietl, "Maximum Likelihood Receiver for MMSE Relaying," *Proc. IEEE ICC*, 2011.

[46] P. Weitkemper, D. Wubben and K. Kammeyer, "Minimum MSE Relaying for Arbitrary Signal Constellations in Coded Relay Networks," *Proc. IEEE VTC*, Spring, 2009.

[47] X. Lu, J. Li, and Y. Liu, "Soft parallel wireless relay via Z-forward," *IEEE Trans Wireless Commun.*, vol. 14, no. 11, pp. 6339-6352, Nov. 2015.

[48] M.C. Valenti and S. Cheng, "Iterative demodulation and decoding of turbo coded M-ary noncoherent orthogonal modulation," *IEEE Journal on Sel. Areas in Comm.*, vol. 23, no. 9, pp. 1738-1747, Sept. 2005.

[49] Y. Li, "Distributed coding for cooperative wireless networks: An overview and recent advances," *IEEE Communications Magazine*,vol. 47, no. 8, pp.71-77, Aug. 2008.

[50] X. N Zeng, A. Ghrayeb, and M. Hasna, "Joint Optimal Threshold-Based Relaying and ML Detection in Network-Coded Two-Way Relay Channels," *IEEE Trans Commun.*, vol. 60, no. 9, pp. 2657-2666, Sep. 2012.

[51] Y. Peng, M. Wu, H. Zhao, and W. Wang, "Cooperative Network Coding with Soft Information Relaying in Two-Way Relay Channels," *Journal of Commun.*, vol. 4, no. 11, pp. 849-855, Dec. 2009.

[52] R. Thobaben, "On Distributed Codes with Noisy Relays," *Proc. IEEE Asilomar*, 2008.

[53] F. Hu and J. Lilleberg, "Novel Soft Symbol Estimate and Forward Scheme in Cooperative Relaying Networks," *Proc. IEEE PIMRC*, 2009.

[54] A. Winkelbauer and G. Matz, "On efficient Soft-Input Soft-Output Encoding of Convolutional Codes," *Proc. IEEE ICASSP*, 2011.

[55] M.S. Rahman, Y. Li and B. Vucetic, "Distributed Analog Channel Coding for Wireless Relay Networks," *Proc. IEEE Globecom*, 2010.

[56] K. Lee, and L. Hanzo, "MIMO-assisted Hard Versus Soft Decoding-and-Forwarding for Network Coding Aided Relaying System," *IEEE Trans Wireless Commun.*, vol. 8, no. 1, pp. 376-385, Jan. 2009.

[57] X. Lu, J. Li and Y. Liu, "Soft-Encoding Distributed Coding for Parallel Relay Systems," *Proc. IEEE ISIT*, Istanbul, Turkey, July 2013.

[58] Y. Li, M. S. Rahman, S. X. Ng, and B. Vucetic, "Distributed soft coding with a soft input soft output (SISO) relay encoder in parallel relay channels," *IEEE Trans Commun.*, vol. 61, no. 9, pp. 3660-3672, Sep. 2013.

[59] J. A. Rice, "Mathematical Statistics and Data Analysis," *Third edition, Thomson Brooks/Cole* , 2007.

[60] S. Lin and D. J.Costello, Jr, "Error Control Coding," *Second edition, Pearson Education, Inc.*, 2004.

[61] S.-Y. R. Li, R. W. Yeung and N. Cai, "Linear network coding," *IEEE Trans. Info. Theory*, vol.49, pp. 371-381, Feb. 2003.

[62] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard and J. Crowcroft, "XORs in the air: practical wireless network coding," *IEEE/ACM Trans. Networking*, vol.16, no. 3, pp. 497-510, Jun. 2008.

[63] E. Larsson and B. Vojcic, "Cooperative transmit diversity based on superposition modulation," *IEEE Commun. Lett.*, vol. 9, no. 9, pp. 778-780, Sep. 2005.

[64] P. Hoeher, and T. Wo, "Superposition modulation: myths and facts," *IEEE Commun. Magazine*, vol. 49, no. 12, pp. 110-116, Dec. 2011.

[65] K. Azarian, H. El Gamal, and P. Schniter, "The three node wireless network: achievable rates and cooperation strategies," *IEEE Trans. Info. Theory*, vol. 51, no. 12, pp. 4152-4172, Dec. 2005.

[66] L. Xiao, T. E. Fuja, J. Kliewer, and D. J. Costello, Jr., "A network coding approach to cooperative diversity ," *IEEE Trans. Info. Theory*, vol. 53, no. 10, pp. 3714-3722, Oct. 2007.

[67] T. E. Hunter and A. Nosratinia, "Diversity through coded cooperation," *IEEE Trans. Wireless Comm.*, vol. 5, no. 2, pp. 283-289, Feb. 2006.

[68] V. Mahinthan, J. W. Mark, and X. Shen, "Performance analysis and power allocation for M-QAM cooperative diversity systems," *IEEE Trans. Wireless Commun.*, vol. 9, no. 3, pp. 1237-1247, Mar. 2010.

[69] S. Vanka, S. Srinivasa, Z. Gong, Vizi, P., K. Stamatiou, and M. Haenggi, "Superposition coding strategies: design and experimental evaluation ," *IEEE Trans. Wireless Comm.*, vol. 11, no. 7, pp. 2628-2639, July 2012.

[70] W. Zhang, M. Lentmaier, K. Zigangirov and D. J. Costello Jr, "Braided convolutional codes: a new class of turbo-like codes," *IEEE Trans. Info. Theory*, vol. 56, no. 1, pp. 316-331, Jan. 2010.

[71] Y. Wu, M. Zheng, Z. Fei, E. Larsson and J. Kuang, "Outage probability analysis for superposition coded symmetric relaying," *Science China, Springer*, vol. 56, Feb. 2013.

[72] L. Xiao, T. E. Fuja, J. Kliewer, and D. J. Costello, Jr, 'Error performance analysis of signal superposition coded cooperative diversity", *IEEE Trans. Comm.*, vol. 57, no. 10, pp. 3123-3131, Oct. 2009.

[73] I. Krikidis, "Analysis and optimization issues for superposition modulation in cooperative networks ," *IEEE Trans. Vehicular. Tech.*, vol. 58, no. 9, pp. 4837-4847, Nov. 2009.

[74] Y. Hu, K. Li and K. Teh, "Performance analysis of two-user cooperative multiple access systems with DF relaying and superposition modulation," *IEEE Trans. Vehicular. Tech.*, vol. 60, no. 7, pp. 3118-3126, Sep. 2011.

[75] T. Yang and J. Yuan, "Performance of iterative decoding for superposition modulation-based cooperative transmission," *IEEE Trans. Wireless Commun.*, vol. 9, no. 1, pp. 51-59, Jan. 2010.

[76] T. V. K. Chaitanya and E. Larsson, "Superposition modulation-based symmetric relaying with hybrid ARQ: analysis and optimization," *IEEE Trans. Vehicular. Tech.*, vol. 60, no. 8, pp. 3667-3683, Oct. 2011.

[77] C. Chiang, U. Phuyal, and V. K. Bhargava, "Power allocation in two-dimensional superposition modulation based cooperative wireless communication system," *IEEE Trans. Commun.*, vol. 60, no. 12, pp. 3662-3670, Dec. 2012.

[78] Y. Hu, K. Li and K. Teh, "An efficient successive relaying protocol for multiple-relay cooperative networks," *IEEE Trans. Wireless Comm.*, vol. 11, no. 5, pp. 1892-1899, May 2012.

[79] Z. Fei, A. Yang, C. Xing, J. Yuan and J. Kuang, "Performance of superposition coding for downlink coordinated two-point system," *IEEE Trans. Vehicular. Tech.*, vol. 62, no. 8, pp. 4057-4064, Oct. 2013.

[80] H. Zhang, H. Xing, X. Chu, A. Nallanathan, W. Zheng, and X. Wen, "Secure resource allocation for OFDMA two-way relay networks," in *Proc. 2012 IEEE Globecom*.

[81] J. Zhao, Z. Lu, X. Wen, H. Zhang, S. He, W. Jing, "Resource management based on security satisfaction ratio with fairness-aware in two-way relay networks," *International Journal of Distributed Sensor Networks*, vol.2015 (2015), Article ID 819195.

[82] A. Goldsmith, *Wireless communications*, chapter 5, Cambridge University Press, 2005.

[83] S. Moshavi, E. Kanterakis and D. L. Schilling, "A new multiuser detection scheme for DS-CDMA systems ," *International Journal of wireless info. networks*, vol. 3, no. 1, Jan. 1996.

Vita

Xuanxuan Lu received the B.E. degree from the School of Telecommunication Engineering, Beijing University of Posts and Telecommunications, Beijing, China, in 2008 and the M.E. degree in information and communication engineering from Zhejiang University, Hangzhou, China in 2011.

Since 2011, she has been currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, Lehigh University, Bethlehem, PA under the supervision of Prof. Tiffany Jing Li.

Her primary research interests include signal processing for multiuser system, cooperative communications, and visible light communications.