Coding Techniques for Advanced Wireless Communication Systems

Chen Gong

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2012
Motivated by the ever increasing demand of wireless communication for larger capacity and higher quality, wireless communication system grows from a single-pair point-to-point communication system to a multiple-transceiver pair communication network. Various new communication techniques, for example, cooperative communication, interference management, multi-carrier communication, are employed to enhance the system capacity and improve the communication quality. Even for some single-pair communication scenarios, due to the different quality demands for different types of information messages, more advanced coding schemes should be designed to provide more protection for more important information messages, for example, the system emergency message.

This thesis proposes several coding schemes to address the above questions. More specifically, the proposed coding schemes are summarized as follows.

- Message-wise error protection is a new unequal error protection scheme where in a codebook some special messages are more protected than other ordinary messages. We propose the first practical coding scheme for message-wise error protection based on LDPC codes, where codeword flipping is employed to separate the special message codewords from the ordinary message codewords.

- We consider a half-duplex 4-node joint relay system with two sources, one relay, and one destination, where the relay combines the information from both sources and transmits it to the destination together with both sources. We propose joint network and channel coding schemes based on the superposition coding (SC) and the Raptor coding (RC), and design practical Raptor codes for the proposed coding schemes.

- We propose novel coding and decoding methods for a fully connected K-user Gaussian inter-
ference channel. Each transmitter encodes its information into multiple layers and transmits the superposition of those layers. Each receiver performs a twofold task by first identifying which interferers it should decode and then determining which layers of them should be decoded. We propose practical coding schemes that employ the quadrature amplitude modulations (QAM) and Raptor codes.

- We propose group decoding and the associated rate allocation schemes for the multi-relay assisted interference channels, where both the relays and the destinations employ constrained group decoding. We consider two types of relay systems, the hopping relay system with no direct source-destination links, and the inband relay system with direct source-destination links. For each relay type, our objective is to design the relay assignment and group decoding strategies at the relays and destinations, to maximize the minimum information rate among all source-destination pairs.

- We consider a distributed storage system employing some existing regenerate codes where the storage nodes are scattered in a wireless network. The existing full-downloading approach, where the data collector downloads all symbols from a subset of the storage nodes for data reconstruction, becomes less efficient in wireless networks. This is because that, due to fading, the wireless channels may not offer sufficient bandwidths for full downloading. We propose a partial downloading scheme that allows downloading a portion of the symbols from any storage node, and formulate a cross-layer wireless resource allocation problem for data reconstruction employing such partial downloading. We derive necessary and sufficient conditions for the data reconstructability for partial downloading, in terms of the numbers of downloaded symbols from the storage nodes. We also propose channel and power allocation schemes for partial downloading in wireless distributed storage systems.
# Table of Contents

1 Introduction 1

2 Message-wise Unequal Error Protection Based on Low-Density Parity-Check Codes 5

2.1 Background .......................................................... 6

  2.1.1 Performance Limits and Optimal Coding Schemes ................. 6

  2.1.2 Proposed Practical Coding Schemes .................................. 8

2.2 Single Special Message .............................................. 9

  2.2.1 Coding Schemes .................................................... 10

  2.2.2 Thresholds for Special Message Detection ......................... 11

  2.2.3 Flipping Detection Analysis ...................................... 12

  2.2.4 Error Exponent Analysis ....................................... 14

2.3 Multiple Special Messages ........................................... 16

  2.3.1 Coding Schemes .................................................... 16

  2.3.2 Analysis of BEC Decoding ....................................... 18

  2.3.3 Analysis of BSC and BIAWGNC Decoding .......................... 18

  2.3.4 Thresholds for the First Stage for BSC and BIAWGNC .............. 19

  2.3.5 Thresholds for the Second Stage for BSC and BIAWGNC .............. 20

2.4 Extensions to High-order QAM .................................... 21

  2.4.1 Single Special Message ........................................... 21

  2.4.2 Multiple Special Messages ....................................... 23

2.5 Numerical and Simulation Results .................................. 23
## 3 Joint Network and Channel Coding for a Joint Relay System

3.1 System Descriptions

3.1.1 Joint Relay System
3.1.2 Rateless Coding Scheme

3.2 Performance Analysis and Code Design

3.2.1 MI Evolution Approximation for MAC Detector
3.2.2 Overview of the Code Design
3.2.3 Code Design for the First Stage
3.2.4 Code Design for SC
3.2.5 Code Design for RC

3.3 Simulation Results

3.3.1 Achievable Rates
3.3.2 Performance of Optimized Codes for 2-user MAC at Relay
3.3.3 Performance of Optimized Codes for Joint Relay Systems

## 4 Constrained Partial Group Decoder for K-user Interference Channels

4.1 System Descriptions

4.1.1 Channel Model
4.1.2 Layered Encoding
4.1.3 Constrained Partial Group Decoding (CPGD)

4.2 Rate Allocation for CPGD

4.2.1 Problem Statement
4.2.2 Optimal Partitions and Rates
4.2.3 Iterative Rate Allocation
4.2.4 Numerical Results

4.3 Practical Transmission Scheme

4.3.1 Rate Selection Procedure
4.3.2 Parameter Design ............................................. 82
4.3.3 Numerical Results ............................................. 84

4.4 Rateless Code Design ............................................. 86
4.4.1 Transmission Using Raptor Codes ............................. 86
4.4.2 Code Design for $\mu = 1$ ..................................... 87
4.4.3 Simulation Results for System with Optimized Codes ........ 89

5 Group Decoding for Multi-relay Assisted Interference Channels 94
5.1 System Descriptions ............................................. 95
5.1.1 Transmission Model .......................................... 95
5.1.2 Relay Assignment and Relaying Modes ....................... 96
5.1.3 Group Decoding at Relays and Destinations ................. 97
5.1.4 Problem Statement ........................................... 98
5.2 Hopping Relay System ............................................ 99
5.2.1 Rate Allocation for Fixed Relay Assignment ................ 100
5.2.2 Dynamic Relay Assignment .................................. 105
5.3 Inband Relay System ............................................. 109
5.3.1 Destination Interference Channels ........................... 109
5.3.2 Rate Allocation for Fixed Relay Assignment ................. 110
5.3.3 Dynamic Relay Assignment .................................. 112
5.4 Numerical Results ................................................ 112
5.4.1 System Performance .......................................... 113
5.4.2 System Complexity Discussion .............................. 117

6 Partial Downloading and Resource Allocation for Wireless Distributed Storage Networks 119
6.1 System Description and Problem Formulation .................. 121
6.1.1 System Model ................................................ 121
6.1.2 Partial Downloading Scheme ................................ 122
6.1.3 Wireless Resource Allocation for Data Reconstruction .... 123
6.2 Reconstructability for Partial Downloading ...................... 125
6.2.1 \( \mu \)-Reconstructability for MSR Point ........................................ 126
6.2.2 A Partial Downloading Scheme for the MBR Point .......................... 128
6.2.3 \( \mu \)-reconstructability for MBR Point ........................................ 130

6.3 Channel and Power Allocation for Wireless Partial Downloading .......... 131
6.3.1 Optimal Channel and Power Allocation for the Relaxed Problem ........ 132
6.3.2 Local Adjustment ........................................................................ 133
6.3.3 Resource Allocation for Node Regeneration .................................... 134

6.4 Numerical Results for Resource Allocation ......................................... 135
6.4.1 Wireless Resource Allocation for Data Reconstruction ....................... 135
6.4.2 Comparison Between MSR and MBR Points .................................... 138
6.4.3 Comparison with the Flexible Coding in [1] ....................................... 139

6.5 Appendices ..................................................................................... 142
6.5.1 Proof of Theorem 6.1 .................................................................. 142
6.5.2 Symbol Selection Procedure for Partial Downloading at MSR Point... 142
6.5.3 Proof of Theorem 6.3 .................................................................. 143
6.5.4 Proof of Theorem 6.4 .................................................................. 144

7 Conclusions ...................................................................................... 146

Bibliography ....................................................................................... 146
List of Figures

2.1 Illustration of sphere-packing balls for the cavity coding and the proposed coding schemes. 9
2.2 MDR and FAR for the BSC and BIAWGN channel for different thresholds. Upper plot: BIAWGN channel; lower plot: BSC. 28
2.3 MDR and FAR for 16-QAM with different thresholds $\epsilon_{QAM}$. 29
2.4 BER performance of message-wise UEP in an AWGN channel with binary input. 30
2.5 BER performance of message-wise UEP in an AWGN channel with 16-QAM input. 31
2.6 Residue erasure probability performance of message-wise UEP in the BEC. 32
2.7 Percentage of detected message types in different decoding iterations for the multiple special message UEP. 33
3.1 A 4-node joint relay model. 41
3.2 Coding schemes for both sources and relay. 42
3.3 Factor graphs for the decoding of relay MAC (upper portion), SC (middle portion) and RC (lower portion). 52
3.4 EXIT charts for multiuser detectors based on LA and QA. 53
3.5 Performance of the optimized Raptor codes for the 2-user MAC. 56
3.6 Rate compatibility of the optimized Raptor codes for the 2-user MAC. 57
3.7 Performance of the optimized joint relay system in AWGN channels. 58
3.8 Throughput performance of the joint relay system in block fading channels employing the optimized profiles for AWGN channels for SC. 60
3.9 Throughput performance of the joint relay system in block fading channels employing the optimized profiles for AWGN channels for RC. 61
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>The sum-rate given by Algorithms 4.2 and 4.3 assuming Gaussian signaling and infinite-length random codes.</td>
<td>66</td>
</tr>
<tr>
<td>4.2</td>
<td>The sum-rate given by Algorithm 4.3 assuming Gaussian signaling and infinite-length random codes for different number of layers.</td>
<td>74</td>
</tr>
<tr>
<td>4.3</td>
<td>The sum-rate given by Algorithm 4.3 assuming Gaussian signaling and infinite-length random codes.</td>
<td>75</td>
</tr>
<tr>
<td>4.4</td>
<td>The averaged sum-rate $\bar{R}_{\text{sum}}(d_1)$ for different cutoff rate $d_1$.</td>
<td>76</td>
</tr>
<tr>
<td>4.5</td>
<td>Minimum rates of sources for the hopping relay system with fixed relay assignment.</td>
<td>83</td>
</tr>
<tr>
<td>4.6</td>
<td>Minimum rates of sources for the hopping relay system with fixed relay assignment.</td>
<td>85</td>
</tr>
<tr>
<td>4.7</td>
<td>Minimum rates of sources for the hopping relay system with dynamic relay assignment.</td>
<td>86</td>
</tr>
<tr>
<td>4.8</td>
<td>Minimum rates of sources for the inband relay system with fixed relay assignment.</td>
<td>87</td>
</tr>
<tr>
<td>4.9</td>
<td>Minimum rates of sources for the inband relay system with dynamic relay assignment.</td>
<td>88</td>
</tr>
<tr>
<td>4.10</td>
<td>The simulated throughput with QAM and optimized rateless codes.</td>
<td>91</td>
</tr>
<tr>
<td>4.11</td>
<td>The simulated throughput in MISO channel with beamforming.</td>
<td>92</td>
</tr>
<tr>
<td>5.1</td>
<td>The simulated multi-relay assisted interference channels.</td>
<td>93</td>
</tr>
<tr>
<td>5.2</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>98</td>
</tr>
<tr>
<td>5.3</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>99</td>
</tr>
<tr>
<td>5.4</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>100</td>
</tr>
<tr>
<td>5.5</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>101</td>
</tr>
<tr>
<td>5.6</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>102</td>
</tr>
<tr>
<td>6.1</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>103</td>
</tr>
<tr>
<td>6.2</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>104</td>
</tr>
<tr>
<td>6.3</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>105</td>
</tr>
<tr>
<td>6.4</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>106</td>
</tr>
<tr>
<td>6.5</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>107</td>
</tr>
<tr>
<td>6.6</td>
<td>The optimal predicted sum-rate under practical modulation and coding schemes.</td>
<td>108</td>
</tr>
</tbody>
</table>
6.7  Histogram of the power saving of the proposed partial downloading scheme over the full downloading scheme for the MBR point. ............................................. 140

6.8  Histogram of the power saving of the proposed partial downloading scheme over the downloading scheme constrained to $K = 4$ nodes for the MBR point. ............................................. 141
## List of Tables

2.1 Profiles for simulated codes. .................................................. 26  
2.2 Codeword length for simulations. ............................................ 27  
3.1 Simulated codes for the entire relay system. ............................ 58  
4.1 Modulation schemes for different rate values. .......................... 80  

viii
Acknowledgments

First and foremost, I would like to thank my Ph.D. advisor Prof. Xiaodong Wang, who has offered me the opportunity of the several years of Ph.D. study in Columbia University. These years of challenging and meaningful Ph.D. life has enlarged various aspects of my vision, not only in terms of the knowledge and the problem solving skills but also in terms of many things beyond studying and working. Moreover, Prof. Wang spent a lot of time and efforts advising my Ph.D. research all through these years. Wherever I will be, and whatever I will do, these years of Ph.D. study will become a treasure in my life.

I was fortunate to work with Dr. Guosen Yue and Dr. Ali Tajer in these years. Discussion with them has enlightened many new ideas for developing new techniques. Their research and working experiences has also helped me a lot. Moreover, working with Prof. Weiqiang Xu and Prof. Xinsong Hou has sparkled me a lot in the research in Columbia.

I would also like to thanks my M.S. advisors Prof. Huijian Cui and Prof. Kun Tang in Tsinghua University for offering me the opportunity and experiences of practical projects. These experiences provide not only another mode of thinking other than that for theoretical research, but also some basic skills for the coding technique research, which turn out to be of great help for the research here in Columbia.

Furthermore, I would like to thank many of my friends who have accompanied me in these years. These friends and colleagues are Prof. Gorge Moustakides, Dr. Guido Jajamovich, Dr. Xin Yan, Dr. Jun Wang, Dr. Chenghao Qi, Dr. Fanggang Wang, Wei Liu, Hua Xiao, Anna Choromanska, Shujuan Wang, Zhe Wang, and Shuying Li. Their encouragement helped me a lot when I was met difficulties in study and life.

I would like to thank the department academia coordinator Elsa for helping me when I was in difficulty and arranging the defense. I would also like to thanks Prof. Shih-fu Chang, Prof. John Wright, Dr. Doru Calin, and Dr. Guosen Yue, for spending time reviewing the thesis and attending
the defense.

As I write this, I am afraid that several important people are being accidently omitted. I apologize if you deserve to be here, but are not. I hope that you can forgive the oversight.

Last but not least, I would like to thank my parents, for their love and support.
Dedicated to all that I love and all that love me.
Chapter 1

Introduction

Motivated by the increasing demand of wireless communication for larger capacity and higher quality, wireless communication system grows from a single-pair point-to-point narrowband communication system to a multiple-transceiver pair wideband communication network. In such network, the challenges for more efficient and reliable communication quality come from not only the noise but also the interference from the co-channel transmitters. These difficulties motivate extensive research and development for more advanced communication techniques. The methods include not only employing multiple antennas and multiple transmission carriers, but also new transmission schemes based on cooperative communication and interference management.

Even for some single-carrier point-to-point communication system, for more advanced communication application, due to the different quality demands for different types of information messages, more advanced coding schemes should be designed to increase the protection levels for more important information messages. For example, in a remote sensing system the system emergency message should be much more protected than the ordinary information messages. This is different from the traditional unequal error protection scheme where in an information message some bits should be more protected than others. Prior research has been devoted to the information theoretical limits for such message-wise unequal error protection. However, a practical coding scheme that can approach the proposed information limits is still missing.

In this thesis, we propose several coding schemes for the message-wise unequal error protection, the joint network and channel coding for a 4-node relay system, $K$-user interference channel, multiple-relay assisted $K$-user interference channel, and a wireless distributed storage network. The
CHAPTER 1. INTRODUCTION

coding schemes proposed in this thesis are summarized as follows.

We design practical message-wise unequal error protection coding schemes. Message-wise error protection is a new unequal error protection scheme where in a codebook some special messages are more protected than other ordinary messages. We propose a practical coding scheme for message-wise error protection based on LDPC codes, where codeword flipping is employed to separate the special message codewords from the ordinary message codewords. Simulation results show that the proposed coding scheme can provide capacity-approach protection to both types of messages as if one type of messages is transmitted.

We propose joint channel and network coding schemes for a four node joint relay system with two sources, one relay, and one destination. In such a system, the relay combines the information from both sources and transmits it to the destination together with both sources. Two coding schemes are proposed based on the superposition coding (SC) and the Raptor coding (RC). We also design Raptor codes for the sources as well as the relay. For both additive white Gaussian noise (AWGN) and block fading channels, the joint relay system with optimized profiles exhibits significant performance gains over that employing the code profile optimized for either the single-user AWGN channel or the 2-user multiple-access channel.

We propose novel practical coding and decoding methods for a fully connected $K$-user Gaussian interference channel. Each transmitter encodes its information into multiple layers and transmits the superposition of those layers. Each receiver performs a twofold task by first identifying which interferers it should decode and then determining which layers of them should be decoded, called constrained partial group decoder (CPGD). We provide a distributed algorithm, which determines the transmission rate at each transmitter based on some optimality measure and also finds the order of the layers to be successively decoded at each receiver. We also consider practical design of a system that employs the quadrature amplitude modulations (QAM) and rateless codes. Numerical results show that the proposed multi-layer coding scheme with CPGD offers significant performance gain over the traditional un-layered transmission with single-user decoding.

We propose coding and rate allocation for a multi-relay assisted fully connected $K$-user Gaussian interference channel, where multiple relays assist the transmissions from the sources to destinations. Both the relays and the destinations employ constrained group decoding, where the desired messages are decoded jointly with some interferers’ messages in a successive manner. We consider two
types of relay systems, the hopping relay system with no direct source-destination links, and the inband relay system with direct source-destination links. For each relay type, our objective is to design the relay assignment and the group decoding strategies at the relays and destinations, to maximize the minimum information rate among all source-destination pairs. Numerical results demonstrate the significant performance improvement provided by the proposed group decoder over the traditional systems that employ single-user decoder at both the relays and the destinations where all interference is treated as noise, as well as those employing amplified-and-forward (AF)-based relaying schemes, as well as the effectiveness of the proposed dynamic relay assignment strategies.

We propose partial downloading and wireless resource allocation schemes for a distributed storage system employing some existing regenerate codes, where the storage nodes are scattered in a wireless network. The data collector connects to the storage nodes via orthogonal channels and downloads data symbols from these nodes. In the existing data reconstruction schemes for distributed storage systems, the data collector downloads all symbols from a subset of the storage nodes. Such a full-downloading approach becomes inefficient in wireless networks since due to fading, the wireless channels may not offer sufficient bandwidths for full downloading. Moreover, full-downloading is also less power efficient than partial downloading. Given a coding scheme employed by the wireless distributed storage system, we propose a partial downloading scheme that allows downloading a portion of the symbols from any storage node. We formulate a cross-layer wireless resource allocation problem for data reconstruction in distributed storage systems employing such partial downloading. To derive the fundamental properties of partial downloading as well as to reduce the complexity of wireless resource allocation, we derive necessary and sufficient conditions for data reconstructability for partial downloading, in terms of the numbers of downloaded symbols from the storage nodes. We also propose channel and power allocation schemes for partial downloading in wireless distributed storage systems. Simulation results are provided to demonstrate the significant power savings by the proposed partial downloading scheme compared with the full-downloading methods for distributed storage.

The above five coding schemes will be addressed in detail in Chapters Two to Six sequentially. Due to the relatively independent technique contents and complicated notation system for each coding scheme, the notations for each coding scheme are defined in the corresponding chapter.
Chapter 2

Message-wise Unequal Error Protection Based on Low-Density Parity-Check Codes

Classical communication theory assumes that all messages are created equal in terms of the protection against channel impairments. Based on this, various channel codes have been designed to approach the channel capacity $2^4$. The unequal error protection (UEP) usually refers to the scenario that in one codeword, some bits are better protected than others, i.e., bit-wise UEP. Several coding schemes for bit-wise UEP are proposed based on low-density parity-check (LDPC) codes and Luby transform (LT) codes. However, it is shown in [7] that without feedback the special bits cannot achieve a larger error exponent than other ordinary bits in the codeword. Therefore, the so-called message-wise UEP is further considered in [7], where one or more special messages are better protected than other ordinary messages, and the fundamental coding limits are studied in [8]. It is shown that without feedback the special messages can indeed achieve a larger error exponent. The message-wise UEP finds applications in various systems, e.g., that the system emergency messages and other important messages should be more protected than ordinary messages.

What is still lacking, however, is a practical coding scheme with performance approaching the information theoretic limits of the message-wise UEP. The cavity coding scheme proposed
in [7], which discards the codewords of ordinary messages that lie inside the sphere-packing balls of special message codewords, although optimal in the information theoretic sense, is less practical since it is difficult to determine which codewords of ordinary messages should be discarded. We propose a practical message-wise UEP scheme that employs LDPC codes and codeword flipping. In particular, we encode ordinary messages using an LDPC code, and flip the codewords whose weights are smaller than half of the codeword length. In case of a single special message, we encode it to the all-zero sequence. For multiple special messages, we encode them using another LDPC code, and flip the codewords whose weights are larger than half of the codeword length. The decoder performs message type detection and codeword flipping detection based on the unsatisfied check nodes in iterative decoding. We provide both theoretical analysis and simulation results for the proposed coding schemes in the binary symmetric channel (BSC), the additive white Gaussian noise (AWGN) channel, and the binary erasure channel (BEC). It is seen that the performance of the proposed practical message-wise UEP scheme is commensurate with that of the information theoretically optimal coding scheme.

2.1 Background

In this section, we give a brief summary of the information theoretic approaches to the message-wise UEP in [7] and a description of the proposed practical coding schemes. We first detect whether a special message codeword is transmitted, and then decode the message based on the corresponding codebook. We define the miss detection probability as the probability of a special message being detected as an ordinary message, and the false alarm probability as the probability of an ordinary message being detected as a special message.

2.1.1 Performance Limits and Optimal Coding Schemes

2.1.1.1 Single Special Message

Let $N$ be the codeword length, and $X$ and $Y$ be the channel input and output alphabets, respectively. Consider the coding scheme in which a single special message is much more protected than other ordinary messages. For a given coding scheme (codebook) $Q$, let $p_{md,Q}$ be the miss detection
probability for the special message and the corresponding error exponent is defined as

\[ E_{md,Q} \triangleq \lim \inf_{N \to \infty} -\frac{\ln p_{md,Q}}{N}. \] (2.1)

It is shown in [7] that the best achievable error exponent for the special message, among all possible codebooks, is given by

\[ E_{md} \triangleq \sup_{Q} E_{md,Q} = D(P^*_Y(y)||W_{Y|X}(y|x_r)), \quad \text{with } x_r = \arg \max_{x \in X} D(P^*_Y(y)||W_{Y|X}(y|x)), \] (2.2)

where \( P^*_Y(y) \) is the capacity achieving distribution of the channel output, \( W_{Y|X}(y|x) \) is the conditional distribution of the channel, and \( D(\cdot||\cdot) \) is the KL divergence [9].

We consider the \( E_{md} \) in (2.2) for the binary symmetric channel (BSC), the binary-input additive Gaussian white noise (BIAWGN) channel, and the binary erasure channel (BEC). For the BSC with the input alphabet \( X = \{0,1\} \) and crossover error probability \( p \), we have \( P^*_Y(y) = 0.5 \) for \( y = 0,1 \), and \( x_r \) can be either 0 or 1. Therefore from (2.2) we have the error component for BSC

\[ E_{md}^S = \ln \left( \frac{1}{2\sqrt{p(1-p)}} \right), \] (2.3)

which shows that for \( p < 1/2 \), the miss detection probability drops exponentially with the codeword length \( N \). For the BIAWGN channel with the input alphabet \( X = \{-1,+1\} \) and noise variance \( \sigma^2 \), we have \( P^*_Y(y) = \frac{1}{2\sqrt{2\pi}\sigma} \left( e^{-\frac{(y+1)^2}{2\sigma^2}} + e^{-\frac{(y-1)^2}{2\sigma^2}} \right) \), and \( x_r \) can be either \(-1\) or \(+1\). Therefore from (2.2) we have the error component for BIAWGN

\[ E_{md}^G = \frac{1}{2\sqrt{(2\pi)\sigma}} \int_{-\infty}^{+\infty} \left( e^{-\frac{(y+1)^2}{2\sigma^2}} + e^{-\frac{(y-1)^2}{2\sigma^2}} \right) \ln \left( 1 + e^{-\frac{2y}{\sigma^2}} \right) dy, \] (2.4)

which shows that for any \( \sigma > 0 \) the miss detection probability also drops exponentially with the codeword length \( N \). For the BEC with the input alphabet \( X = \{0,1\} \), the output alphabet \( Y = \{0,1,E\} \), and erasure probability \( p_E \), we have \( P^*_Y(0) = P^*_Y(1) = 0.5 - p_E/2 \) and \( P^*_Y(E) = p_E \). Therefore from (2.2) we have the error component for BEC

\[ E_{md}^E = +\infty, \] (2.5)

which shows that the miss detection probability can be reduced to zero.

The optimal coding scheme, as illustrated in Fig. 2.1, is as follows. The special message is encoded to the all-zero sequence for the BSC and all-one sequence for the BIAWGN channel.
The ordinary messages are encoded using a capacity-achieving codebook, in which \( \text{the low weight codewords are discarded} \). In the decoder, the special message is detected if the type of channel output sequence is not close to \( P_Y(y) \); otherwise the maximum-likelihood (ML) decoding is performed to decode the ordinary messages.

### 2.1.1.2 Multiple Special Messages

There are two conclusions on the performance limits for the UEP of multiple special messages. One is that the special messages can be protected to the same extent as if the ordinary messages are not transmitted, i.e., the existence of ordinary messages does not affect the protection of special messages. The other is that there is essentially no rate loss for ordinary messages if, in a codebook, the codewords outside the decoding balls of special codewords are employed for ordinary messages.

The optimal coding scheme, also illustrated in Fig. 2.1, is as follows. The special messages are encoded using a capacity-achieving codebook. The ordinary messages are encoded using another capacity-achieving codebook and cavity encoding, i.e., the codewords in the ordinary message codebook close to the special message codewords are discarded and the remaining codewords are employed to encode the ordinary messages. The decoder first detects whether a special message is transmitted by checking whether some special codeword is close to the received sequence. Then based on the detection outcome, it decodes the message using either the special or the ordinary codebook.

### 2.1.2 Proposed Practical Coding Schemes

The cavity coding schemes in [7], although optimal in the information theoretic sense, are difficult to implement in practice since there is not a systematic way to determine which codewords in the ordinary codebook are close to the special codewords. For example, let \( K \) be the information sequence length of the ordinary message encoder, and \( S \subset \{0,1\}^K \) be the set of information sequences whose codewords lie outside the decoding balls of special codewords. Let \( K' \) denote the length of ordinary messages. Since only a subset \( S \) of \( \{0,1\}^K \) is used for encoding, we have \( K' < K \). However, it is difficult to construct an efficient mapping from \( \{0,1\}^{K'} \) to \( S \).

We propose a practical coding scheme based on codeword flipping. As shown in Fig. 2.1, we split the codeword space \( \{0,1\}^N \) into two half spaces, one composed of the sequences of weight
CHAPTER 2. MESSAGE-WISE UNEQUAL ERROR PROTECTION BASED ON LOW-DENSITY PARITY-CHECK CODES

Figure 2.1: Illustration of sphere-packing balls for the cavity coding and the proposed coding schemes.

less than \( N/2 \) (lower half), and the other composed of the sequences of weight larger than \( N/2 \) (upper half). The codeword flipping makes the special and ordinary codewords dwell in the lower and upper half spaces, respectively, by folding the codewords in the upper and lower half spaces to their opposites. The decoder performs the detection of message types and codeword flipping by tracking the number of unsatisfied checks in iterative decoding. To better distinguish the flipped codewords from the original ones, we employ the LDPC codes with all-odd degree check nodes, for which no parity checks are satisfied for flipped codewords.

2.2 Single Special Message

In this section, we propose practical coding strategies in the BSC, the BIAWGN channel, and the BEC for the case of single special message, and provide performance analysis.
2.2.1 Coding Schemes

2.2.1.1 Encoding

The ordinary messages are encoded using a capacity-approaching LDPC \((N, M_o)\) code \(C_o\) with all-odd degree check nodes, and the codeword is flipped if its weight is smaller than \(N/2\). The single special message is encoded to the all-zero codeword. The reason all check nodes are of odd degree is that it can maximize the possibility of successfully codeword flipping detection.

2.2.1.2 Decoding

The decoding involves three steps: (a) detect whether a special message or an ordinary message is sent; (b) if an ordinary message is detected, detect whether the codeword is flipped; (c) decode the ordinary codeword.

First we present the decoding process for BSC and BIAWGN channel.

Detection of special message: Denote the channel output corresponding to a transmitted codeword as \(y_1, y_2, \ldots, y_N\). Define \(\bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i\). Then a special message is detected if

\[ \bar{y} < \frac{1}{2} - \epsilon_S \text{ for BSC}; \quad \bar{y} > \epsilon_G \text{ for BIAWGN}, \]

(2.6)

where \(\epsilon_S\) and \(\epsilon_G\) are some thresholds to be specified in Section 2.2.2. If an ordinary message is detected, we then perform the following codeword flipping detection.

Detection of codeword flipping: We first make hard decision on each coded bit based on its corresponding channel output \(y_i\). Let \(E_o^{(0)}\) be the number of unsatisfied check nodes based on the decoded bits, where \((0)\) indicates before the first LDPC decoding iteration. If \(E_o^{(0)} > M_o/2\) then the flipped codeword is detected; otherwise the original codeword is detected. If the codeword flipping is detected, we change the channel output \(y_i\) to \((1 - y_i)\) for BSC and to \(-y_i\) for BIAWGN channel.

Next we present the decoding process for BEC.

Detection of special message: We check the non-erased bits. If all non-erased bits are zero then a special message is detected.

Detection of codeword flipping: We look for a check node, of which none adjacent variable nodes are erased, and see if its check relation is violated. If so the codeword flipping is detected; otherwise
CHAPTER 2. MESSAGE-WISE UNEQUAL ERROR PROTECTION BASED ON LOW-DENSITY PARITY-CHECK CODES

the original codeword is detected. The miss decoding is declared if such a check nodes cannot be found. After that, we perform the LDPC decoding.

2.2.2 Thresholds for Special Message Detection

The thresholds $\epsilon_S$ and $\epsilon_G$ in (2.6) for detecting the special message are typically chosen to meet a certain requirement on either the miss detection probability $p_{md}$ or the false alarm probability $p_{fa}$. Hence in this subsection we provide a simple analysis on these two performance metrics.

2.2.2.1 Miss Detection Probability

For BSC, when the all-zero sequence is transmitted, $N\bar{y} = \sum_{i=1}^{N} y_i$ is the sum of $N$ independent Bernoulli $Bern(p)$ random variables and therefore has a binomial distribution. By the central limit theorem, for large $N$, the distribution of $\bar{y}$ is well approximated by Gaussian, i.e., $\bar{y} \sim N(p, \frac{p(1-p)}{N})$.

Then for the threshold $\epsilon_S$, the miss detection probability can be approximated by

$$p_{md}^S \approx Q\left(\frac{1/2 - \epsilon_S - p}{\sqrt{p(1-p)/N}}\right),$$

(2.7)

where $Q(x) = \frac{1}{\sqrt{2\pi}} \int_{x}^{\infty} e^{-t^2/2} dt$. On the other hand, for BIAWGN channel, when the all-one sequence is transmitted, we have $\bar{y} \sim N(1, \sigma^2/N)$. Hence the miss detection probability is given by

$$p_{md}^G = Q\left(\frac{1 - \epsilon_G}{\sigma} \sqrt{N}\right).$$

(2.8)

It can be easily seen that for BEC the miss detection probability is zero, because the erasure cannot flip the bit zero to bit one. Therefore we have

$$p_{md}^E = 0.$$  

(2.9)

2.2.2.2 False Alarm Probability

Denote $\{A_k\}$ as the weight spectrum of the ordinary code $C_o$, where $A_k$ is the fraction of weight-$k$ codeword. The false alarm probability is then given by

$$p_{fa} = \sum_k A_k p_{fa}(k),$$

(2.10)

where $p_{fa}(k)$ is the false alarm probability corresponding to the weight-$k$ ordinary codeword.
We first consider the BSC. Let \( \mu_y(k) \) and \( \sigma_y^2 \) denote the mean and variance of \( \bar{y} \), respectively, given that the weight of the message codeword is \( k \). Note that if \( k < N/2 \), then the codeword is first flipped before being transmitted and hence the actual weight of the sequence is \( N - k \). We then have
\[
\mu_y(k) = (2p - 1) \left| \frac{1}{2} - \frac{k}{N} \right| + \frac{1}{2}, \quad \text{and} \quad \sigma_y^2 = \frac{p(1-p)}{N}.
\] (2.11)

Again by invoking the Gaussian approximation, we have
\[
p_{fa}^S(k) \approx Q \left( \frac{\mu_y(k) - 1/2 + \epsilon_g}{\sigma_y} \right).
\] (2.12)

On the other hand, for the BIAWGN channel, by taking into account codeword flipping, it follows that \( \bar{y} \sim \mathcal{N}(-|N - 2k|/N, \sigma^2/N) \), given that the message codeword has a weight \( k \). Thus we have
\[
p_{fa}^G \approx Q \left( \frac{\sqrt{N}\epsilon_G}{\sigma} \right).
\] (2.13)

For the BEC, given a weight-\( k \) codeword, the false alarm occurs only when all the bits one are erased. Therefore, the false alarm probability is given by
\[
p_{fa}^E(k) = (p_E)^k.
\] (2.14)

From (2.13) we can see that for the BIAWGN channel
\[
p_{fa}(k) > p_{fa}(k-1) \text{ for } k \leq N/2, \text{ and } p_{fa}(k) < p_{fa}(k+1) \text{ for } k \geq N/2,
\] (2.15)

which also holds true for the BSC and the proof is given in Appendix 2.6.1. Hence an upper bound of the false alarm probability is given by \( p_{fa} \leq p_{fa}(N/2) \). For long LDPC codes, since the weights of almost all codewords are approximately \( N/2 \), the false alarm probability can be simply approximated using the dominating term, i.e., \( p_{fa} \approx p_{fa}(N/2) \).

### 2.2.3 Flipping Detection Analysis

We next justify the flipping detection method given in Section 2.2.1.2. Denote \( q \) as the bit error probability of the hard decision based on the channel output. We have \( q = p \) for the BSC and \( q = Q(1/\sigma) \) for the BIAWGN channel. If the codeword is not flipped, then the probability that the
any one hard decision is different from the original coded bits is $q$. On the other hand, for flipped codewords, since the channel changes the flipped bits back to the original bits with probability $q$, the probability that any one hard decision is different from the original coded bit is $1 - q$. We make the same assumption as that in [10] for the performance analysis of LDPC code ensembles based on density evolution, where a check node independently selects its degree $j$ with the probability $\tilde{\rho}_j$, and uniformly selects $j$ adjacent variable nodes among all variable nodes. For the long LDPC codes employed, this independent assumption works well, even after several iterations when the traversing messages become correlated. Here we hold similar assumption that for long codes all parity check relations are independent of each other.

Suppose that the check node profile is $\{\tilde{\rho}_j\}$ from the node perspective. The probabilities that a randomly selected check node is unsatisfied for the original and flipped codewords, are given respectively by

$$p_o = \frac{1 - \sum_j \tilde{\rho}_j (1 - 2q)^j}{2}, \quad \text{and} \quad p_f = \frac{1 - \sum_j \tilde{\rho}_j (2q - 1)^j}{2}. \quad (2.16)$$

We have that $p_f > p_o$ since $q < 1/2$.

Based on the assumption of independent parity check relations, the numbers of unsatisfied check nodes $E_o(0)$ satisfy the binomial distributions $\text{Bino}(p_o, M_o)$ and $\text{Bino}(p_f, M_o)$ for the original and flipped codewords, respectively. Hence the probability that the number of unsatisfied check nodes is $E_o(0)$ for these two cases are given respectively by

$$p_o(E_o(0)) = \binom{M_o}{E_o(0)} (p_o)^{E_o(0)} (1 - p_o)^{M_o - E_o(0)},$$

$$\text{and} \quad p_f(E_o(0)) = \binom{M_o}{E_o(0)} (p_f)^{E_o(0)} (1 - p_f)^{M_o - E_o(0)}. \quad (2.17)$$

Assume that the codeword flipping occurs with probability $1/2$, i.e., the probability that the codeword weight is larger than $N/2$ is $1/2$, which is valid for long codes. Then given the number of unsatisfied check nodes $E_o(0)$, the ML decision rule is to declare a flipping if the likelihood ratio $p_o(E_o(0))/p_f(E_o(0)) < 1$. Hence, it follows that flipping is detected if

$$\frac{E_o(0)}{M_o} > \frac{\ln (1 - p_f) - \ln (1 - p_o)}{\ln p_o - \ln (1 - p_o) + \ln (1 - p_f) - \ln p_f} = \frac{1}{2}, \quad (2.18)$$

where the last equality follows from the fact that, with all-odd degree check nodes we have $p_f + p_o = 1$, as shown below.
Next we give a justification on the assumption that all check node degrees are odd. Fixing $p^c_o$, from (2.16), we have

$$p^c_f = \frac{1 - \sum_j \tilde{\rho}_j (2q - 1)^j}{2}$$

$$= \frac{1 + \sum_{\text{odd } j} \tilde{\rho}_j (1 - 2q)^j - \sum_{\text{even } j} \tilde{\rho}_j (1 - 2q)^j}{2}$$

$$\leq \frac{1 + \sum_{\text{odd } j} \tilde{\rho}_j (1 - 2q)^j + \sum_{\text{even } j} \tilde{\rho}_j (1 - 2q)^j}{2}$$

$$= 1 - p^c_o,$$

(2.19)

where the equality is achieved only if all check node degrees are odd. For the two binomial distributions $Bino(p^c_o, M_o)$ and $Bino(p^c_f, M_o)$, the detection error will be minimized if they are most separated, i.e., if $p^c_f$ is maximized when all check node degrees are odd.

The error probability for flipping detection can again be obtained by the Gaussian approximation. In particular, the fraction $E_o^{(0)}/M_o$ follows $N(p^c_o, \eta^2)$ and $N(p^c_f, \eta^2)$ for the original and flipped codewords, respectively, where $\eta^2 = p^c_o p^c_f / M_o$. The miss detection and false alarm probabilities are then given by

$$p_{\text{flip,md}} = p_{\text{flip,fa}} \approx Q\left(\frac{1/2 - p^c_o}{\eta}\right) = Q\left(\frac{\sum_j \tilde{\rho}_j (1 - 2q)^j}{2\eta}\right).$$

(2.20)

For the BEC, we study the probability that every check node has at least one adjacent variable node erased by the channel. For a specific check node, the probability that all its adjacent nodes are not erased is given by

$$p^c_{\text{node,NE}} = \sum_j \tilde{\rho}_j (1 - p_E)^j$$

(2.21)

Again based on the assumption of independent parity check relations, the probability that all check nodes have one or more adjacent variable nodes erased is given by

$$p_{\text{flip,md}} = (1 - p^c_{\text{node,NE}})^M_o = \left(1 - \sum_j \tilde{\rho}_j (1 - p_E)^j\right)^{M_o},$$

(2.22)

which is virtually zero for any reasonable channel erasure probability $p_E$.

2.2.4 Error Exponent Analysis

In this subsection we compute the error exponents for the proposed practical coding schemes and reveal their relationship with the corresponding optimal error exponents from the information
CHAPTER 2. MESSAGE-WISE UNEQUAL ERROR PROTECTION BASED ON LOW-DENSITY PARITY-CHECK CODES

First we consider the miss detection probability. The analysis for the BEC is unnecessary since $p_{md}^E = 0$. For the BSC, it is the probability that the all-zero sequence is flipped to a sequence of weight larger than $(1/2 - \epsilon_S)N$. The channel output $y_i$ is i.i.d. and follows the distribution $P \sim Bern(p)$. The miss detection event corresponds to the type set $E = \{y_i, 1/N \sum_{i=1}^N y_i \geq 1/2 - \epsilon_S\}$ and $p_{md} = P(E)$. If $p < 1/2$, then for sufficiently small $\epsilon_S (\epsilon_S < 1/2 - p)$, by the Sanov’s theorem \[9\], the error exponent is given by

$$\lim_{N \to \infty} -\frac{1}{N} \ln p_{md} = D(P_0^*||P),$$

(2.23)

where $P_0^* = \arg \min_{P_0 \in E_0} D(P_0||P)$ and $E_0 = \{Bern(q), q \geq 1/2 - \epsilon_S\}$. Then we have $P_0^* \sim Bern(1/2 - \epsilon_S)$, and

$$D(P_0^*||P) = \left(\frac{1}{2} - \epsilon_S\right) \ln \left(\frac{1/2 - \epsilon_S}{p}\right) + \left(\frac{1}{2} + \epsilon_S\right) \ln \left(\frac{1/2 + \epsilon_S}{1 - p}\right),$$

(2.24)

which can be made arbitrarily close to the optimal error exponent given in (2.3) for sufficiently small $\epsilon_S$.

For the BIAWGN channel, we have $\bar{y} \sim N(1, \sigma^2/N)$. The miss detection probability is then $p_{md} = P(\bar{y} < \epsilon_G) = Q\left(\frac{1-\epsilon_G}{\sqrt{\sigma^2/N}}\right)$. Since $\left(\frac{1}{2} - \frac{1}{2\pi}\right) \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \leq Q(x) \leq \left(\frac{1}{2}\right) \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ \[12\], we have $\lim_{x \to +\infty} \ln Q(x)/x^2 = -1/2$ and the error exponent is given by

$$\lim_{N \to \infty} -\ln Q\left(\frac{1-\epsilon_G}{\sqrt{\sigma^2/N}}\right) = \frac{1}{2} \left(1 - \epsilon_G\right)^2,$$

(2.25)

where $\epsilon_G$ can be arbitrarily small. It is shown in Appendix 2.6.2 that the above error exponent coincides with the optimal error exponent $\mathcal{E}_{md}^G$ given in (2.4) in the low SNR region, i.e.,

$$\lim_{\sigma \to \infty} \frac{\mathcal{E}_{md}^G}{1/2} = 1.$$

(2.26)

Next we consider the false alarm probability, which is dominated by that for the weight-N/2 codewords (cf. (2.15)). For BSC, suppose that the channel flips a fraction $q_1$ of bit one and a fraction $q_2$ of bit zero. Then the set $S = \{(q_1, q_2)|(1 - q_1)/2 + q_2/2 \leq (1/2 - \epsilon_S)\}$ describes the flipping types that cause false alarms. Since $(p, p) \notin S$, by Sanov’s theorem the probability of the type set $S$ drops exponentially with the codeword length $N$; and the exponent is given by

$$\mathcal{E}_{fa}^S = \lim_{N \to \infty} -\ln p_{fa}^N = \min_{(q_1, q_2) \in S} \left(1/2D(q_1||p) + 1/2D(q_2||p)\right),$$

(2.27)
where we simply use \( p \), \( q_1 \), and \( q_2 \) to represent the distributions \( \text{Bern}(p) \), \( \text{Bern}(q_1) \), and \( \text{Bern}(q_2) \), respectively. For the BIAWGN channel, the false alarm probability for the weight-\( N/2 \) codeword is \( Q\left(\frac{\epsilon_G}{\sqrt{\sigma^2/N}}\right) \). Thus we have the error exponent

\[
\mathcal{E}_{fa}^G = \lim_{N \to \infty} -\frac{\ln p_{fa}}{N} = \frac{\epsilon_G^2}{2\sigma^2},
\]

(2.28)
since \( \lim_{x \to +\infty} \ln Q(x)/x^2 = -1/2 \), which again indicates that the false alarm probability drops exponentially with the codeword length \( N \). For the BEC, similarly the error component is that for the weight-\( N/2 \) codewords, which is given according to (2.14) as follows

\[
\mathcal{E}_{fa}^E = \lim_{N \to \infty} -\frac{\ln p_{fa}}{N} = -\ln \sqrt{P_E}.
\]

(2.29)

2.3 Multiple Special Messages

2.3.1 Coding Schemes

2.3.1.1 Encoding

The special and ordinary messages are encoded using the capacity-approaching \((N, M_s)\) code \( C_s \) and \((N, M_o)\) code \( C_o \), respectively. All the check nodes of \( C_s \) and \( C_o \) are of odd degrees. The codeword is flipped if the weight of the special codeword is larger than \( N/2 \), or the weight of the ordinary codeword is smaller than \( N/2 \).

2.3.1.2 Decoding for BSC and BIAWGN

The decoding for BSC and BIAWGN involves two stages. In the first stage, we detect whether the special or ordinary message is sent. If a decision is made, then the message is decoded using the corresponding decoder. On the other hand, if a decision cannot be reached in the first stage, then during the second stage, both the special and ordinary decoders are run in parallel. Based on the number of unsatisfied parity checks, the type of the message is determined and the corresponding decoder output gives the decoded message.

**First stage:** The special message is detected if

\[
\bar{y} < 1/2 - \eta^s_G \quad \text{for BSC}; \quad \bar{y} > \eta^s_G \quad \text{for BIAWGN}.
\]

(2.30)
The ordinary message is detected if
\[ y > \frac{1}{2} + \eta_o S \quad \text{for BSC}; \quad y < -\eta_o G \quad \text{for BIAWGN}. \] (2.31)

The choices of the thresholds \( \eta_S, \eta_o S, \eta_S G, \) and \( \eta_o G \) are discussed in Section 2.3.4. If the special or ordinary codeword is detected, we then perform the codeword flipping detection the same way as in Section 2.2.1.2 followed by the decoding of the corresponding codeword. Otherwise, if
\[ \frac{1}{2} - \eta_S \leq y \leq \frac{1}{2} + \eta_S \quad \text{for BSC}; \quad -\eta_o \leq y \leq \eta_o G \quad \text{for BIAWGN}, \] (2.32)
we run the second stage.

**Second stage:** In this stage, we first make hard decisions on the coded bits based on the channel outputs \( \{y_i\} \) and compute the number of unsatisfied parity checks for both the special and ordinary codes, denoted as \( E_s^{(0)} \) and \( E_o^{(0)} \), respectively. If \( E_s^{(0)} > M_o/2 \), we flip the channel outputs \( \{y_i\} \) for the special message decoder; and similarly for the ordinary message decoder.

Then, we proceed to perform iterative decoding for both \( C_s \) and \( C_o \), and compute the fraction of unsatisfied checks \( E_s^{(i)} / M_s \) and \( E_o^{(i)} / M_o \) in each iteration. Given some thresholds \( \theta_s \) and \( \theta_o \), if for some \( i \), \( E_s^{(i)} / M_s < \theta_s \) and \( E_o^{(i)} / M_o > \theta_o \), then a special codeword is detected; if \( E_s^{(i)} / M_s > \theta_s \) and \( E_o^{(i)} / M_o < \theta_o \), then an ordinary codeword is detected. If either the special or ordinary message is detected, we continue decoding the message of the detected type and terminate the other. If the message type is still not detected in the last iteration, the decoding failure is declared.

### 2.3.1.3 Decoding for BEC

Next we present the decoding schemes for BEC, which also involve two stages. The decoding is based on the standard erasure decoding scheme of LDPC codes. We decode using the decoders of both types of messages, and perform the wrong decoder detection along the decoding process, as follows.

**First stage:** We check if the right decoder is employed based on the parity checks, whose contributing variable bits are not erased. If the parity checks of these nodes are all satisfied, or all unsatisfied, then we use this decoder in the following stage; otherwise a wrong decoder is detected. If the check relations are all unsatisfied, we flip the codeword bits.

**Second stage:** We successively find a check node with only one erasure adjacent variable node and decode the erasure node, until all erasure nodes are decoded or decoding fails if the process...
stops but there are unrecovered erasure nodes. In the above procedure, if a variable node is the only unrecovered variable node for multiple check node and the recovery based on one of these check nodes will cause unsatisfied parity checks for others, we declare that a wrong decoder is detected.

2.3.2 Analysis of BEC Decoding

We present the analysis for the probability that the wrong decoder cannot be detected in the first stage. The analysis is based on the fact that for the special message decoder, the ordinary codeword can be viewed as a random binary sequence, and vice versa. If the special message codeword is transmitted, then the probability that the ordinary message decoder is used but not detected in the first stage is given by

\[ p_{\text{DetErr,1}}^{E} = 2 \sum_{n=0}^{M_o} \left( \frac{1}{2} \right)^n \binom{M_o}{n} \left( P_{\text{node,NE}}^c \right)^n \left( 1 - P_{\text{node,NE}}^c \right)^{M_o-n}, \]

where the sum is performed by considering all cases of \( n \) check nodes with all non-erased adjacent variable nodes, and that the probabilities that a check node is satisfied or unsatisfied are both 1/2. From (2.33), we have

\[ p_{\text{DetErr,1}}^{E} = 2 \left( 1 - P_{\text{node,NE}}^c \right)^{M_o}, \]

which shows that the probability that the ordinary message decoder cannot be detected in the first stage drops exponentially with \( M_o \). Similarly, the probability of undetected special message decoder applied for the ordinary codeword also drops exponentially with \( M_s \). For all simulated codewords, the wrong decoder can be detected in the first stage.

2.3.3 Analysis of BSC and BIAWGNC Decoding

Consider one codeword \( c^s = (c^s_1, c^s_2, ..., c^s_N) \) of weight \( \alpha N \) of \( C_s \) and the other \( c^o = (c^o_1, c^o_2, ..., c^o_N) \) of weight \( \beta N \) of \( C_o \). The two codewords are chosen equally random from the corresponding two codebooks. Since the two codewords are independent of each other, the probability that any one of the corresponding bits of the two codewords is different is given by

\[ P(c^s_i \neq c^o_i) = P(c^s_i = 0)P(c^o_i = 1) + P(c^s_i = 1)P(c^o_i = 0) = \alpha(1 - \beta) + \beta(1 - \alpha). \]
CHAPTER 2. MESSAGE-WISE UNEQUAL ERROR PROTECTION BASED ON LOW-DENSITY PARITY-CHECK CODES

The minimum of the probability $P(c_s \neq c_o)$ for all ordinary codewords in $C_o$ is given by $\min_{0 \leq \beta \leq 1} \{\alpha(1 - \beta) + \beta(1 - \alpha)\} = \min\{\alpha, 1 - \alpha\}$. Hence the codeword of weight $\alpha N$ from the codebook $C_s$ can be seen as the output of a virtual BSC with the input as a codeword of $C_o$ and the crossover probability

$$p_{equ} = \min\{\alpha, 1 - \alpha\}.$$  \hspace{1cm} (2.36)

Thus for a $C_s$ codeword of weight $\alpha N$ passing through the actual channel and then decoded by $C_o$, it can be viewed as a codeword of $C_o$ passing through a concatenated channel, with the first stage being a BSC with the crossover probability $p_{equ}$. For $C_o$ with the rate $r_o$, let $p_{th}^{o}$ be its BSC Shannon limit determined from $1 - H(p_{th}^{o}) = r_o$, where $H(p) = -p \log_2(p) - (1 - p) \log_2(1 - p)$. If $p_{equ} > p_{th}^{o}$, i.e., $p_{th}^{o} < \alpha < 1 - p_{th}^{o}$, then decoding a codeword of $C_s$ by the decoder of $C_o$ will fail, since the first component channel error exceeds the error-correcting capability of the code $C_o$. Hence for this range of $\alpha$ we can run the two decoders in parallel, as in the second-stage, the incorrect decoder will fail. On the other hand, if $\alpha < p_{th}^{o}$ or $\alpha > 1 - p_{th}^{o}$ (in the latter case the codeword is flipped), we perform the special codeword detection in the first stage before decoding. Similar arguments hold for ordinary messages.

2.3.4 Thresholds for the First Stage for BSC and BIAWGNC

As in the case of single special message, the detection thresholds $\eta_o^{S}$, $\eta_s^{S}$, $\eta_o^{G}$, and $\eta_s^{G}$ in the first stage are determined based on the miss detection and false alarm probabilities. First consider the BSC. As in Section 2.2.2.2 the probability that an ordinary codeword is misdetected as a special codeword (i.e., $\bar{y} < 1/2 - \eta_s^{S}$) is upper bounded by that for the weight $N/2$ codeword, given by

$$Q\left(\frac{\eta_s^{S}}{\sqrt{p(1 - p)}}\sqrt{N}\right).$$  \hspace{1cm} (2.37)

Next we consider the probability that a special codeword of weight smaller than $p_{th}^{o}N$ or larger than $(1 - p_{th}^{o})N$ is not detected as a special codeword in the first step, i.e., it is flipped by the channel to a sequence of weight larger than $(1/2 - \eta_s^{S})N$. It is upper bounded by the corresponding probability of the weight $p_{th}^{o}N$ codeword, for which $\bar{y}$ can be approximated by a Gaussian distribution with the mean $p(1 - p_{th}^{o}) + (1 - p)p_{th}^{o}$ and variance $p(1 - p)/N$. We consider $p < p_{th}^{o}$, because for $p > p_{th}^{o}$ the second component channel error exceeds the error-correcting capability of $C_o$ and the decoding
by \( C_o \) will fail. Since for \( p < p_o^{th} \) we have \( p(1 - p_o^{th}) + (1 - p)p_o^{th} < 2p_o^{th}(1 - p_o^{th}) \), this upper bound is given by

\[
Q\left( \frac{1/2 - \eta_s^s - p(1 - p_o^{th}) - (1 - p)p_o^{th}}{\sqrt{p(1 - p)}} \right) \leq Q\left( \frac{1/2 - \eta_s^s - 2p_o^{th}(1 - p_o^{th})}{\sqrt{p(1 - p)}} \right). \tag{2.38}
\]

We choose the threshold \( \eta_B^s \) to minimize the maximum of the two upper bounds given by (2.37) and (2.38), i.e.,

\[
\eta_B^s = \arg \max_\eta \min \{ \eta, 1/2 - \eta - 2p_o^{th}(1 - p_o^{th}) \} = 1/4 - p_o^{th}(1 - p_o^{th}). \tag{2.39}
\]

The analysis for the BIAWGN channel is similar. The probability that an ordinary message is misdetected as a special message (i.e., \( \bar{y} > \eta_G^s \)) is upper bounded by \( Q\left( \frac{\eta_G^s \sigma}{\sqrt{N}} \right) \). The probability that a special message of weight \( \alpha N \) satisfying \( \min\{\alpha, 1 - \alpha\} < p_o^{th} \) is not detected as a special message is upper bounded by \( Q\left( \frac{1-2p_o^{th} - \eta_G^s \sigma}{\sqrt{N}} \right) \). Similar to (2.39), we choose the threshold \( \eta_G^s \) to minimize the maximum of the above two upper bounds, i.e.,

\[
\eta_G^s = \arg \max_\eta \min \{ \eta \sigma \sqrt{N}, \frac{1 - 2p_o^{th} - \eta \sigma}{\sqrt{N}} \} = 1/2 - p_o^{th}. \tag{2.40}
\]

Similarly, we can obtain the thresholds for ordinary messages, given by

\[
\eta_G^o = 1/4 - p_s^{th}(1 - p_s^{th}), \quad \text{and} \quad \eta_G^o = 1/2 - p_s^{th}, \tag{2.41}
\]

where \( p_s^{th} \) is the BSC Shannon limit for the special message code.

### 2.3.5 Thresholds for the Second Stage for BSC and BIAWGN

We present a scheme to select the second-stage detection thresholds \( \theta_s \) and \( \theta_o \) for the fraction of unsatisfied check nodes to classify the correct and wrong decoders. Recall that the ordinary decoder can correct the BSC bit error of probability up to \( p_o^{th} \). If a special message codeword is sent and in order for the ordinary decoder to fail to decode it, the error probability with respect to the ordinary codewords should be greater than \( p_o^{th} \). Then the fraction of unsatisfied check nodes would be larger than

\[
f_o^c = \frac{1 - \sum_j \tilde{p}_o^j(1 - 2p_o^{th})^j}{2}, \tag{2.42}
\]
where \( \{ \tilde{\rho}_j^o \} \) is the check node profile for \( C_o \) from the node perspective. Similarly, if an ordinary codeword is sent, the fraction of unsatisfied check nodes by the special decoder would be larger than

\[
f^c_o = 1 - \sum j \tilde{\rho}_j^s(1 - 2p_{th}^s)^j,
\]

where \( \{ \tilde{\rho}_j^s \} \) is the check node profile for \( C_s \) from the node perspective.

In practice, since \( p_{th}^o \) and \( p_{th}^s \) are typically larger than the actual decoding thresholds for \( C_o \) and \( C_s \), the predicted fractions \( f^c_o \) and \( f^c_s \) will be larger than the fraction of unsatisfied check nodes.

We can choose the detection thresholds \( \theta_o \) and \( \theta_s \) somewhat smaller than the fractions \( f^c_o \) and \( f^c_s \) given in (2.42) and (2.43), e.g., \( \theta_s = 0.7f^c_s \) and \( \theta_o = 0.7f^c_o \). This makes sense since if the correct decoder is employed the fraction of unsatisfied check nodes will be zero or very small.

### 2.4 Extensions to High-order QAM

In this section we extend the message-wise UEP techniques to AWGN channels employing \( M \)-QAM with \( M = 2^{b_1+b_2} \). We use the natural bit-to-symbol mapping, where the first \( b_1 \) and last \( b_2 \) labeling bits of each signal point indicate its column and row indices, respectively. The reason of employing the natural mapping is that the signal point \((x, y)\) is flipped to \((-x, -y)\) when the codeword is flipped, which facilitates the separation of the signals for special and ordinary messages.

Let \( M_1 = 2^{b_1} \) and \( M_2 = 2^{b_2} \). In the following, we assume that the constellation signal points are \((2i - M_1 + 1, 2j - M_2 + 1)\) for \( 0 \leq i \leq M_1 - 1 \) and \( 0 \leq j \leq M_2 - 1 \). We employ a binary LDPC code of length \( N(b_1 + b_2) \), and map the \( N(b_1 + b_2) \) bits to the \( N \) signal points in the constellation.

#### 2.4.1 Single Special Message

The special message is encoded to the all-zero sequence, and the ordinary messages are encoded using a capacity-approaching LDPC code with all-odd degree check nodes. Let \( s^{(k)} \) be the constellation signal points, \( 0 \leq k \leq M - 1 \), where \( s^{(0)} \) is the signal point for the all-zero labeling bits. For the length-\( N \) symbol sequence \( \{r_i\} \), define

\[
L_u(\{r_i\}) \triangleq \frac{1}{MN} \sum_{i=1}^{N} \sum_{k=0}^{M-1} \|r_i - s^{(k)}\|^2 \quad \text{and} \quad L_s(\{r_i\}) \triangleq \frac{1}{N} \sum_{i=1}^{N} \|r_i - s^{(0)}\|^2 \tag{2.44}
\]
as the average distance between \( \{r_i\} \) and the coded symbol sequences of ordinary and special messages, respectively. The definition of \( L_u(\{r_i\}) \) follows that for long codes the ordinary message codeword bits are 0-1 equal probable and uniformly distributed over all signal points in the constellation.

For the encoded symbol sequence \( x_1, \ldots, x_N \), let \( x^I_i \) and \( x^Q_i \) be the in-phase (I) and quadrature (Q) components of \( x_i \), respectively. Define \( \bar{x}^I \triangleq \frac{1}{N} \sum_{i=1}^{N} x^I_i \) and \( \bar{x}^Q \triangleq \frac{1}{N} \sum_{i=1}^{N} x^Q_i \). We flip the codeword if \( L_s(\{x_i\}) < L_u(\{x_i\}) \), which is equivalent to \( \bar{x}^I + \bar{x}^Q > K \) for some constant \( K \) determined from (2.44). The rationale behind this flipping rule is that, if one encoded sequence is closer to the encoded sequence for the special message than the encoded sequences for ordinary messages, it should be flipped to be closer to those for ordinary messages. Note that the above scheme is a natural generalization of that for the binary case, in which for the constellation \{+1, −1\} the flipping condition \( L_s(\{y_i\}) < L_u(\{y_i\}) \) is equivalent to \( \sum_{i=1}^{N} y_i > 0 \), i.e., the codeword weight is smaller than \( N/2 \), which is the flipping condition given in Section 2.2.1.1.

At the decoder, the special codeword is detected if \( L_s(\{y_i\}) < L_u(\{y_i\}) - \lambda \) for some \( \lambda \). Denote the I and Q components of the received signals as \( y^I_i \) and \( y^Q_i \), \( 1 \leq i \leq N \), respectively. Define \( \bar{y}^I \triangleq \frac{1}{N} \sum_{i=1}^{N} y^I_i \) and \( \bar{y}^Q \triangleq \frac{1}{N} \sum_{i=1}^{N} y^Q_i \). Then this flipping detection rule is equivalent to

\[
\bar{y}^I + \bar{y}^Q > \epsilon_{QAM}
\]  

(2.45)

for some threshold \( \epsilon_{QAM} \). If the ordinary message is detected, we perform the codeword flipping detection in the same way as that for the binary case.

The miss detection and false alarm probabilities can be evaluated as follows. Assume the signal point for the special message is \( (s^{(0I)}, s^{(0Q)}) \). If the special message is sent, we have \( \bar{y}^I \sim \mathcal{N}(s^{(0I)}, \sigma/\sqrt{N}) \), \( \bar{y}^Q \sim \mathcal{N}(s^{(0Q)}, \sigma/\sqrt{N}) \), and \( \bar{y}^I + \bar{y}^Q \sim \mathcal{N}(s^{(0I)} + s^{(0Q)}, \sqrt{2\sigma}/\sqrt{N}) \). Thus the miss detection probability is given by

\[
p_{md} = Q \left( \frac{s^{(0I)} + s^{(0Q)} - \epsilon_{QAM}}{\sqrt{2\sigma}} \sqrt{N} \right).
\]  

(2.46)

The false alarm probability can be approximated by assuming that the symbols in each encoded sequence are uniformly distributed over all constellation points, i.e., \( \bar{x}^I = \bar{x}^Q = 0 \), and \( \bar{y}^I + \bar{y}^Q \sim \mathcal{N}(0, \sqrt{2\sigma}/\sqrt{N}) \). Thus the false alarm probability can be approximated as

\[
p_{fa} \approx Q \left( \frac{\epsilon_{QAM}}{\sqrt{2\sigma}} \sqrt{N} \right).
\]  

(2.47)
2.4.2 Multiple Special Messages

The encoding scheme for multiple special messages is a natural generalization of that for the binary case. For the binary case, denote the mean of the coded sequence \( \{x_i\} \) as \( \bar{x} = 1/N \sum_{i=1}^{N} x_i \). Then we have \( \bar{x} \geq 0 \) for special messages and \( \bar{x} \leq 0 \) for ordinary messages. For the general \( M \)-QAM case, we let the coded symbols for the ordinary and special messages dwell in the regions given by \( \bar{x}^I + \bar{x}^Q \geq 0 \) and \( \bar{x}^I + \bar{x}^Q \leq 0 \), respectively. Thus in the encoding, the special message codeword is flipped if \( \bar{x}^I + \bar{x}^Q < 0 \), and the ordinary messages codeword is flipped if \( \bar{x}^I + \bar{x}^Q > 0 \).

At the decoder, we make a first-stage detection of message type. A special message is detected if \( \bar{y}^I + \bar{y}^Q > \eta_{QAM} \) and an ordinary message is detected if \( \bar{y}^I + \bar{y}^Q < -\eta_{QAM} \). If the message type is not detected, then during the second stage detection we run the special and ordinary message decoders in parallel, flip the codewords if necessary, and track the number of unsatisfied check nodes in iterative decoding. The message type detection scheme is similar to that for the binary input, i.e., if the fraction of unsatisfied check nodes for one code becomes very small then we terminate the decoding of the other.

2.5 Numerical and Simulation Results

Code Optimizations

The LDPC codes employed in the following simulations are optimized based on the extrinsic transfer information charts [13] and differential evolution [14] under the constraint that the check node profile is concentrated to two consecutive odd degrees. Compared with the constraint that check node profile is concentrated to two consecutive degrees [12], this constraint incurs no performance degradation in terms of the profile threshold. We consider the BIAWGN channel, the AWGN channel with 16-QAM modulation, and the BEC. The ordinary messages are encoded using a rate-1/2 code. For the multiple special message case, the special messages are encoded using a rate-1/3 code. The optimized profiles for the rate-1/2 and -1/3 codes, as well as the corresponding thresholds (\( E_s/N_0 \) for the AWGN channel and \( p_E^* \) for the BEC), are shown in Table 2.1. The profiles \( \{\tilde{\lambda}_a^b\} \) and \( \{\tilde{\rho}_a^b\} \) denote the variable and check node profiles, respectively, from the node perspective. The superscript \( a = G, Q, E \) for the BIAWGN channel, the AWGN channel with 16-QAM modulation, and the BEC, respectively, and the subscript \( b = o, s \) for the special and
ordinary messages, respectively.

For 16-QAM signals, if we use the same optimization procedure as that for the binary-input case above, we obtain the flipping detection error probability around $10^{-3}$, which is too high. Therefore, we add an additional constraint in the optimization that the average check node degree should be smaller than some threshold $\bar{\rho}^*$, to increase the term $\sum_{j} \tilde{\rho}_j (1 - 2q)^j$ in (2.20). The profile threshold loss incurred by this constraint is less than 0.1dB.

Table 2.2 shows the codeword length, as well as the error probability of the codeword flipping detection $p_{\text{flip,md}}$ at the code thresholds, for the simulated codes. For the binary and 16-QAM cases, the detection error probability is computed based on (2.20), and for the BEC it is computed from (2.34). For the BEC, from (2.22), the probabilities that the special and ordinary decoders can escape the first stage decoder detection, if the opposite types of messages are transmitted, denoted as $p_{\text{esc}}$, are both smaller than $10^{-50}$.

Performance for Single Special Message

First we show the performance of the proposed single special message UEP coding scheme for the BSC (with crossover probability $p = 0.49$) and BIAWGN channel (with $E_s/N_0 = -37$dB). The miss detection rate (MDR) and false alarm rate (FAR) given by (2.7), (2.8), and (2.10) are plotted as a function of the threshold in Fig. 2.2, where the code weight spectrum is obtained from Monte Carlo simulations. The FAR upper bound $p_{fa}(N/2)$ is also plotted, which serves as a good approximation to the FAR. Note that the channel conditions considered here are extremely bad. For normal operating channel conditions, both the MDR and the FAR are virtually zero. Next we show the proposed single special message UEP performance in the AWGN channel with 16-QAM. Fig. 2.3 plots the MDR and FAR given by (2.46) and (2.47) as a function of the detection threshold for $E_s/N_0 = -33$dB.

The bit error rate (BER) performance of the optimized codes for the ordinary messages, for both binary and 16-QAM input signals, is shown in Fig. 2.4 and Fig. 2.5 respectively. In simulations no flipping detection error event is observed. Therefore, the codeword flipping does not incur any performance loss compared with the case without codeword flipping. Note that due to the perfect detection of the message type and codeword flipping, there is no performance loss incurred to the ordinary messages by the existence of the special message. The residue erasure probability of the
ordinary messages for the BEC is shown in Fig. 2.6 Since in simulations no flipping detection error occurs, there is no performance loss incurred to ordinary messages by the existence of special message.

Performance for Multiple Special Messages

We consider the AWGN channel with $E_s/N_0 = -2$dB and 7dB for the binary and 16-QAM input, respectively. The detection thresholds are computed according to (2.40) and (2.41) for the binary input, and $\eta_{QAM}^s = \eta_{QAM}^o = 1.0$ for the 16-QAM input. 100,000 codewords of each type are simulated, for which the information bits are generated uniformly over \{0, 1\}. No message type is detected in the first stage since for long codes almost all codewords have weights approximately equal to $N/2$. For $\theta_s = 0.7 f_c^s$ and $\theta_o = 0.7 f_c^o$, Fig. 2.7 shows the percentage of messages detected at different decoding iterations. It is seen that for the binary-input case, all special codewords are detected in the first iteration, and all ordinary codewords are detected in the second iteration. On the other hand, for the 16-QAM input, the special messages are also all detected in the first iteration; whereas 32.5% of the ordinary messages are detected in the first stage, and the rest are detected in the second stage. For the BEC, in simulations the decoder detection is performed correctly in the first stage.

The BER performance of the proposed message-wise UEP scheme under multiple special messages is shown in Fig. 2.4 and Fig. 2.5 for the binary input and 16-QAM, respectively. The residue erasure probability performance for the BEC is shown in Fig. 2.6. Again due to the perfect detection of message type and codeword flipping, the performance of each type of messages is the same as if only that type of message is transmitted.
Table 2.1: Profiles for simulated codes.

<table>
<thead>
<tr>
<th>Profile</th>
<th>$\tilde{\lambda}_G^C$</th>
<th>$\tilde{\lambda}_G^S$</th>
<th>$\tilde{\lambda}_Q^C$</th>
<th>$\tilde{\lambda}_Q^S$</th>
<th>$\tilde{\lambda}_E^C$</th>
<th>$\tilde{\lambda}_E^S$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.4632</td>
<td>0.5035</td>
<td>0.4761</td>
<td>0.5794</td>
<td>0.4027</td>
<td>0.4853</td>
</tr>
<tr>
<td>3</td>
<td>0.2934</td>
<td>0.2840</td>
<td>0.2896</td>
<td>0.1989</td>
<td>0.4503</td>
<td>0.3529</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>0.0037</td>
<td></td>
<td></td>
<td></td>
<td>0.0025</td>
</tr>
<tr>
<td>5</td>
<td>0.1058</td>
<td>0.0985</td>
<td>0.1013</td>
<td>0.1316</td>
<td>0.0294</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.0153</td>
<td>0.0027</td>
<td>0.0230</td>
<td></td>
<td></td>
<td>0.0720</td>
</tr>
<tr>
<td>7</td>
<td>0.0350</td>
<td>0.0385</td>
<td>0.0399</td>
<td>0.0419</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>0.0108</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td>0.0408</td>
<td>0.0342</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td>0.0778</td>
<td>0.0020</td>
<td>0.0023</td>
<td>0.0027</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0066</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0054</td>
</tr>
<tr>
<td>14</td>
<td>0.0040</td>
<td>0.0020</td>
<td>0.0043</td>
<td>0.0052</td>
<td>0.0010</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0.0126</td>
<td>0.0119</td>
<td>0.0092</td>
<td>0.0135</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.0295</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>0.0707</td>
<td>0.0535</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td>0.0107</td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td>0.0042</td>
<td>0.0255</td>
<td>0.0429</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Profile</th>
<th>$\tilde{\rho}_G^C$</th>
<th>$\tilde{\rho}_G^S$</th>
<th>$\tilde{\rho}_Q^C$</th>
<th>$\tilde{\rho}_Q^S$</th>
<th>$\tilde{\rho}_E^C$</th>
<th>$\tilde{\rho}_E^S$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>0.1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>0.6500</td>
<td></td>
<td>0.9000</td>
<td></td>
<td>0.4548</td>
</tr>
<tr>
<td>7</td>
<td>0.6193</td>
<td>0.3500</td>
<td>0.7500</td>
<td>0.3100</td>
<td>0.5452</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.3807</td>
<td>0.2500</td>
<td></td>
<td>0.2500</td>
<td>0.6900</td>
<td></td>
</tr>
</tbody>
</table>

| Threshold | -2.66dB | -5.10dB | 5.4dB | 2.0dB | 0.479 | 0.637 |
| Note      | -       | -       | $\tilde{\rho}^* = 6.5$ | $\tilde{\rho}^* = 3.8$ |
### Table 2.2: Codeword length for simulations.

<table>
<thead>
<tr>
<th>Type</th>
<th>N</th>
<th>$p_{flip,md}$</th>
<th>$p_{esc}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single - binary</td>
<td>100000</td>
<td>$&lt; 10^{-30}$</td>
<td>-</td>
</tr>
<tr>
<td>Multiple - binary</td>
<td>150000</td>
<td>$&lt; 10^{-30}$</td>
<td>-</td>
</tr>
<tr>
<td>Single - 16QAM</td>
<td>150000</td>
<td>$\approx 10^{-10}$</td>
<td>-</td>
</tr>
<tr>
<td>Multiple - 16QAM</td>
<td>150000</td>
<td>$\approx 10^{-10}$</td>
<td>-</td>
</tr>
<tr>
<td>Single - BEC</td>
<td>100000</td>
<td>$&lt; 10^{-100}$</td>
<td>-</td>
</tr>
<tr>
<td>Multiple - BEC</td>
<td>150000</td>
<td>$&lt; 10^{-100}$</td>
<td>$&lt; 10^{-50}$</td>
</tr>
</tbody>
</table>
Figure 2.2: MDR and FAR for the BSC and BIAWGN channel for different thresholds. Upper plot: BIAWGN channel; lower plot: BSC.
Figure 2.3: MDR and FAR for 16-QAM with different thresholds $\epsilon_{QAM}$. 
Figure 2.4: BER performance of message-wise UEP in an AWGN channel with binary input.
Figure 2.5: BER performance of message-wise UEP in an AWGN channel with 16-QAM input.
Figure 2.6: Residue erasure probability performance of message-wise UEP in the BEC.
Figure 2.7: Percentage of detected message types in different decoding iterations for the multiple special message UEP.
2.6 Appendix

2.6.1 Proof of (2.15) for BSC

It suffices to show that \( p_{fa}(i) < p_{fa}(i+1) \) for \( W \geq N/2 \) since for \( i < N/2 \), \( p_{fa}(i) = p_{fa}(N-i) \) due to codeword flipping.

Without loss of generality, we consider the two sequences with the first \( i \) and \( i+1 \) bits one and the remaining bits zero, denoted as \( s_i \) and \( s_{i+1} \), respectively. Let \( c = (c_1, c_2, ..., c_N) \) be a codeword sequence. Let \( S^c \) denote the set of sequences with weight less than \( (1/2 - \epsilon)N \). We partition \( S^c \) into two subsets \( S'_1 \) and \( S'_2 \) based on the transform \( T \) given by

\[
T : (c_1, ..., c_{i+1}, ...c_N) \rightarrow (c_1, ..., \overline{c_{i+1}}, ...c_N),
\]

where \( \overline{c_{i+1}} \) is the complement of the bit \( c_{i+1} \). Let \( S'_1 = \{ c \in S^c : T(c) \in S^c \} \) and \( S'_2 = S^c \setminus S'_1 \).

There are two properties for this partition. One is that for \( c \in S'_2 \), \( c_{i+1} = 0 \), otherwise the weight of \( T(c) \) is less than the weight of \( c \) so \( T(c) \in S^c \) and \( c \in S'_1 \). The other is that since \( T(T(c)) = c \), then for \( c \in S'_1 \), \( T(c) \in S'_1 \).

Let \( D_H(\cdot, \cdot) \) denote the Hamming distance between two sequences. For any \( s \), we have \( D_H(s, s_i) = D_H(T(s), s_{i+1}) \); and for \( s \in S'_2 \), we have \( D_H(s, s_i) = D_H(s, s_{i+1}) - 1 \). For a BSC with crossover probability \( p \), let \( p(r|s) \) denote the conditional probability that the sequence \( r \) is received given the transmitted sequence \( s \). We have \( p(r|s) = p^{D_H(r,s)}(1-p)^{N-D_H(r,s)} \), which decrease with \( D_H(r,s) \); and

\[
\begin{align*}
p_{fa}(i) &= \sum_{s \in S^c} p(s|s_i) = \sum_{s \in S'_1} p(s|s_i) + \sum_{s \in S'_2} p(s|s_i) \\
&= \sum_{s \in S'_1} p(T(s)|s_{i+1}) + \sum_{s \in S'_2} p(s|s_i) = \sum_{s \in S'_1} p(s|s_{i+1}) + \sum_{s \in S'_2} p(s|s_i) \\
&> \sum_{s \in S'_1} p(s|s_{i+1}) + \sum_{s \in S'_2} p(s|s_{i+1}) = \sum_{s \in S^c} p(s|s_{i+1}) = p_{fa}(i+1),
\end{align*}
\]

where (2.49) holds because the summation of \( T(s) \) over \( S'_1 \) is equivalent to the summation of \( s \) over \( S'_1 \).

2.6.2 Proof of (2.26)

Letting \( \mu = 1/\sigma \), we rewrite (2.41) as follows

\[
E_{md} = E_{md}(\mu) = \int_{-\infty}^{+\infty} \frac{1}{2\sqrt{2\pi}} \left( e^{-\frac{(x+\mu)^2}{2}} + e^{-\frac{(x-\mu)^2}{2}} \right) \ln \left( \frac{1 + e^{-2\mu x}}{2} \right) dx,
\]

(2.50)
and will prove that \( \lim_{\mu \to 0} \frac{E_{md}(\mu)}{\mu^{2}} = \frac{1}{2} \). Since
\[
\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} \left( e^{-\frac{(x+\mu)^{2}}{2}} + e^{-\frac{(x-\mu)^{2}}{2}} \right) \mu x dx = 0, \tag{2.50}
\]
we have
\[
\lim_{\mu \to 0} \frac{E_{md}(\mu)}{\mu^{2}} = \lim_{\mu \to 0} \frac{1}{\mu^{2}} \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^{2}+\mu^{2}}{2}} \left( \frac{e^{\mu x} + e^{-\mu x}}{2} \right) \ln \left( \frac{e^{\mu x} + e^{-\mu x}}{2} \right) dx
\]
\[
= \lim_{\mu \to 0} \frac{1}{\mu^{2}} \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^{2}}{2}} \left( \frac{e^{\mu x} + e^{-\mu x}}{2} \right) \ln \left( \frac{e^{\mu x} + e^{-\mu x}}{2} \right) dx \tag{2.51}
\]
\[
= \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^{2}}{2}} \lim_{\mu \to 0} \frac{1}{\mu^{2}} \left( \frac{e^{\mu x} + e^{-\mu x}}{2} \right) \ln \left( \frac{e^{\mu x} + e^{-\mu x}}{2} \right) dx \tag{2.52}
\]
\[
= \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^{2}}{2}} \frac{x^{2}}{2} dx = \frac{1}{2}.
\]
Note that \(2.52\) follows \(2.51\) and the Fubini-Tonelli theorem that the order of the limit \(\lim_{\mu \to 0}\) and the integral \(\int_{-\infty}^{+\infty}\) can be changed.
Cooperative communications \cite{15} offer performance improvements over the direct communications from both the information theoretical analysis \cite{16} and the practical code design \cite{17,20}. For practical considerations, the relay node is usually half-duplex so it cannot receive and transmit simultaneously. The design of low-density parity-check (LDPC) codes for a half-duplex relay has been investigated in \cite{18,19}. Among these works, two methods proposed in \cite{18,19} follow the information theoretic setups; and one simple code extending method proposed in \cite{20} offer good performance with low receiver complexity at the destination.

On the other hand, network coding was originally developed for information packet combining in the application layer \cite{21,22}, where there is no noise or interference. Recently, it has been applied to the physical layer with turbo \cite{23} and LDPC codes \cite{24,25}. In these works, orthogonal channels are assumed for the sources and relay, which however may incur significant loss in terms of the spectrum efficiency.

We consider the joint relay system with two sources, one half-duplex relay, and one destination. We assume non-orthogonal transmissions which can approach the achievable rates given by the information theoretical analysis. Due to the half-duplex constraint, the entire transmission is divided into two stages. In the first stage, both sources broadcast their information to the relay and the destination. We employ the decode-and-forward (DF) strategy where the relay tries to decode the information of both sources at the end of the first stage. In the second stage, both sources keep sending information to the destination, and the relay combines the information from both
sources and transmits the combined information to the destination. For information combining at
the relay, we consider two physical layer network coding schemes, namely, the superposition coding
(SC) \cite{26} and the Raptor coding (RC).

We employ Raptor codes \cite{4,27} for transmissions at both sources and the relay. The Raptor code
is a concatenated code with an inner Luby transform (LT) code to provide the rate compatibility
and a high-rate outer code to reduce the error floor of the LT code, and can approach the capacity
limit of the additive white Gaussian noise (AWGN) channel with the optimized profile \cite{4}. We fix
the outer codes to be some high-rate irregular repeat accumulate (IRA) codes \cite{28}, and consider
the design of parity generation profiles for LT encoding. To combat the mutual cancellation of two
signals of opposite signs in the multiple-access channel (MAC), we transmit the coded bits of both
the outer IRA codes and the inner LT codes (i.e., doping). The iterative \textit{a posteriori} probability
(APP) detection and soft Raptor decoding is employed at both the relay and destination receivers.
The extrinsic information transfer (EXIT) function-based performance analysis is presented, in
which the linear approximation (LA) and quadratic approximation (QA) are used to model the
mutual information (MI) evolution for the 2-user and 3-user detectors, respectively. We optimize
the code profiles for both SC and RC. Simulation results show that the optimized codes perform
close to the achievable rate limits, and outperform the codes employing various other benchmark
profiles.

3.1 System Descriptions

3.1.1 Joint Relay System

Fig. 3.1 shows the model for the 4-node cooperative communication system under consideration.
It consists of two source nodes $S_1$ and $S_2$, one half-duplex relay node $R$, and one destination node
$D$. We assume a unit time for the entire transmission. In the first stage of duration $t$, both $S_1$ and
$S_2$ broadcast to $R$ and $D$. In the second stage of duration $1 - t$, $S_1$, $S_2$, and $R$ transmit signals
to $D$. The transmitted signals, the received signals, and the channel gains are shown in Fig. 3.1.
With the exponential power decay, we have $h_{ij} = d_{ij}^{\alpha/2} g_{ij}$, $i, j \in \{1, 2, R, D\}$, where $g_{ij}$ denote the
the corresponding small-scale fading coefficient, and $\alpha \in (2, 5)$ is the power decay factor \cite{29}.

We consider binary coding and modulation, i.e., $x_k^{(q)} \in \{\pm 1\}$ for $k, q = 1, 2$ in Fig. 3.1. This
is because the performance gain provided by the relay node is significant in the low SNR region
where the binary modulation is near optimum. Note that the transmitted signals $w_{R}^{(2)}$ by the relay are not necessarily binary.

Assume that the average transmission power for the entire system is constrained by $P$, i.e.,
\[
t(P_1^{(1)} + P_2^{(1)}) + (1-t)(P_1^{(2)} + P_2^{(2)} + P_R^{(2)}) \leq P.
\]
We will consider the following two power allocation (PA) schemes.

**PA-1:** The sum power constraints for both stages are the same [19], i.e., $P_1^{(1)} + P_2^{(1)} \leq P$ and $P_1^{(2)} + P_2^{(2)} + P_R^{(2)} \leq P$.

**PA-2:** The powers of both sources remain the same for both stages, i.e., $P_1^{(1)} = P_2^{(1)}$ and $P_1^{(2)} = P_2^{(2)}$.

### 3.1.2 Rateless Coding Scheme

We employ rateless codes at both sources and the relay. Rateless codes have flexible rate compatibility [4,27] and they can be more easily optimized for the joint relay system under consideration than the LDPC codes [20], as will be detailed in Section 3.2. In the first stage, both sources keep sending coded bits to the relay, until the relay can successfully decode their information. In the second stage, both sources and the relay transmit additional parity bits to the destination, until the destination successfully decodes the information of both sources.

We employ the Raptor codes with the outer IRA codes for rateless encoding. Denote $\{\lambda_j^{(k)}\}$ as the profiles of the inner LT codes used at $S_k$ in the $q$th stage, and $\{\lambda_j^{R}\}$ as the profile of the code at $R$ for the information of $S_k$, for $k, q = 1, 2$. For the source $S_k$, the IRA code encodes $K_k$ information bits to $N_k$ LT input bits. For $q = 1, 2$, denote $M^{(q)}$ as the length of the transmitted sequence in the $q$th stage, which is equal to the length of coded bits under BPSK modulation. For the LT encoding with the degree profile $\{\lambda_j\}$, each coded bit is obtained by XORing $d$ IRA precoded bits that are uniformly selected from all precoded bits with the probability $P(d = j) = \lambda_j$.

#### 3.1.2.1 Stage 1: Doped Raptor Codes at Sources

The first stage corresponds to a 2-user MAC with BPSK modulation, where half of the received signals at the relay are the superpositions of two symbols of opposite signs. If the power difference of the two users is small, the extrinsic messages for these signals are very weak due to the mutual signal cancellation. In the optimized profile for the 2-user MAC, the portion of transmitted precoded bits,
i.e., \( \lambda_1 \), is small. Since the iterative decoding starts from only the strong extrinsic messages for the degree-1 output nodes, the enhancement of weak messages may get stuck for the optimized profile. The decoding fails sometimes even without noise. One solution is to make \( \lambda_1 \) large enough, which however significantly degrades the performance due to the large fraction of degree-1 output nodes connected to the same precoded bit.

To overcome this problem, we employ the doping method for concatenated codes \cite{30,31}, in which we transmit the coded bits of not only the inner code but also the outer code. The iterative decoding starts from the strong extrinsic messages for both the degree-1 LT coded bits and the precoded bits, so the weak messages can be enhanced through iterative decoding. The difference between doping and degree-1 LT coded bits is that with doping all the precoded bits are transmitted once, while with degree-1 LT coded bits, some precoded bits are transmitted multiple times while some are not transmitted due to the uniform selection of precoded bits for LT encoding. In both threshold evaluations and simulation results, the doped Raptor codes significantly outperform the Raptor codes without doping but with a large fraction of degree-1 coded bits. As shown in the upper portion of Fig. \ref{fig:3.2} in the first stage the source \( S_k \) first transmits the \( N_k \) IRA coded bits and then transmits the \((M^{(1)} - N_k) \) LT coded bits generated based on the profile \( \{\lambda_j^{k(1)}\} \).

### 3.1.2.2 Stage 2: Coding Schemes at Relay

In the second stage, the two sources generate \( M^{(2)} \) LT coded bits from the IRA precoded bits based on the profiles \( \{\lambda_j^{1(2)}\} \) and \( \{\lambda_j^{2(2)}\} \), and transmit them to the destination. No doping is needed now as the precoded bits have already been transmitted to the destination.

To combine and transmit the information from both sources at the relay, we consider the following two schemes, namely, the superposition coding (SC) and the Raptor coding (RC).

**Superposition coding (SC):** As shown in the middle portion of Fig. \ref{fig:3.2} the relay \( R \) generates exactly the same coded bits as \( S_1 \) and \( S_2 \), and transmits the superimposed signals, \( w_R^{(2)} = \sqrt{\beta_1} x_1^{(2)} + \sqrt{\beta_2} x_2^{(2)} \), to the destination, where \( \beta_k, k = 1, 2 \), denotes the power allocation factor and \( \beta_1 + \beta_2 = 1 \).

Assuming perfect synchronization, we can rewrite the received signals at \( D \) as

\[
y_D^{(2)} = (h_{1D} \sqrt{P_1^{(2)}} + h_{RD} \sqrt{\beta_1 P_R^{(2)}}) x_1^{(2)} + (h_{2D} \sqrt{P_2^{(2)}} + h_{RD} \sqrt{\beta_2 P_R^{(2)}}) x_2^{(2)} + n_D^{(2)}.
\] (3.1)
Hence, the signal model at $D$ in the second stage is still a 2-user MAC carrying two independent information sequences.

**Raptor coding (RC):** As shown in the bottom portion of Fig. 3.2, the relay $R$ generates new coded bits by jointly encoding the messages from $S_1$ and $S_2$, and sends the BPSK modulated signals $w^{(2)}_R$ to $D$. To generate each of the $M^{(2)}$ coded bits, we select $d_1$ bits uniformly from all the $N_1$ precoded bits of $S_1$, $d_2$ bits uniformly from all the $N_2$ precoded bits of $S_2$, and XOR all the $d_1 + d_2$ selected precoded bits, where $d_1$ and $d_2$ are independently generated with the probabilities $P(d_1 = j) = \lambda_1^{jR}$ and $P(d_2 = j) = \lambda_2^{jR}$. The signal model at $D$ in the second stage is then a 3-user MAC but carrying only two independent information streams.

### 3.1.2.3 Iterative Detection and Decoding

Since the signal models at both the relay and the destination are MAC, we employ iterative receivers at both nodes to perform the iterative APP detection and Raptor decoding. The extrinsic messages are iteratively exchanged between the APP multiuser detector and the soft Raptor decoders. At each iteration, the APP multiuser detection is followed by one round of soft Raptor decoding.
CHAPTER 3. JOINT NETWORK AND CHANNEL CODING FOR A JOINT RELAY SYSTEM

Distances between the sources, relay, and destination

\( d_{1D} \)
\( d_{1R} \)
\( d_{RD} \)
\( d_{2R} \)
\( d_{2D} \)

\( h_{1R} \)
\( h_{1D} \)
\( h_{2R} \)
\( h_{2D} \)

\( y_D^{(1)} = h_{1D} \sqrt{P_1^{(1)} x_1^{(1)}} + h_{2D} \sqrt{P_2^{(1)} x_2^{(1)}} + n_D^{(1)} \)
\( y_R^{(1)} = h_{1R} \sqrt{P_1^{(1)} x_1^{(1)}} + h_{2R} \sqrt{P_2^{(1)} x_2^{(1)}} + n_R^{(1)} \)

\( y_D^{(2)} = h_{1D} \sqrt{P_1^{(2)} x_1^{(2)}} + h_{2D} \sqrt{P_2^{(2)} x_2^{(2)}} + h_{RD} \sqrt{P_R^{(2)} w_R^{(2)}} + n_D^{(2)} \)

Figure 3.1: A 4-node joint relay model.
CHAPTER 3. JOINT NETWORK AND CHANNEL CODING FOR A JOINT RELAY SYSTEM

Figure 3.2: Coding schemes for both sources and relay.
3.2 Performance Analysis and Code Design

We perform the code ensemble analysis for the relay system under consideration based on the EXIT functions [13]. The evolutions of extrinsic mutual information (MI) are shown using the factor graphs in Fig. 3.3. The precode check nodes for $S_k$ are denoted as $\Gamma_k$; the LT input nodes (or IRA precoded bits) for $S_k$ are denoted as $\Omega_k$; the LT output nodes (or LT coded bits) in the $q$th stage for $S_k$ are denoted as $\Lambda_k^{(q)}$; and the LT output nodes for $R$ are denoted as $\Lambda_R$. The MAC channel in the $q$th stage is represented by the MAC factor nodes denoted as $\Phi^{(q)}$.

We now define the extrinsic MI for the iterative detection and decoding illustrated in Fig. 3.3. The subscripts $p$, $v$, $c$, $m$, and $d$ are used to represent $\Gamma_k$, $\Omega_k$ (treated as variable nodes in the iterative decoding), $\Lambda_k^{(q)}$ and $\Lambda_R$ (treated as check nodes), the mean of $\Omega_k$ and $\Lambda_k^{(1)}$, and $\Phi^{(q)}$, respectively. With these subscripts, for source $S_k$, we denote $I_{vc}^{(1)}$, $I_{vc}^{(2)}$, $I_{vc}^{kr}$, and $I_{vp}$ as the extrinsic MI outputs from $\Omega_k$ to $\Lambda_k^{(1)}$, to $\Lambda_k^{(2)}$, to $\Lambda_R$, and to $\Gamma_k$, respectively. Denote $I_{vd}^{(1)}$ and $I_{cd}^{(1)}$ as the extrinsic MI outputs to $\Phi^{(1)}$ from $\Omega_k$ and from $\Lambda_k^{(1)}$, respectively. Therefore, $I_{md}^{(1)}$ denotes the average extrinsic MI of $I_{vd}^{(1)}$ and $I_{cd}^{(1)}$. Similarly, we define $I_{cd}^{(2)}$ and $I_{rd}^{(2)}$ as the extrinsic MI from $\Lambda_k^{(2)}$ and $\Lambda_R$ to $\Phi^{(2)}$, respectively. Moreover, the reverse order in the subscript is used to represent the MI in opposite directions.

3.2.1 MI Evolution Approximation for MAC Detector

In [32] a low-complexity erasure channel approximation to the MI evolution for the multiuser detectors is proposed, which works only for the 2-user MAC with equal channel gains. Here we propose a low-complexity MI evolution approximation for multiuser detectors, which works for the 2- and 3-user channel with arbitrary channel gains.

In the 4-node joint relay system, the MAC model is given by

$$y = \sum_{j=1}^{J} G_j x_j + n,$$

where $x_j \in \{\pm 1\}$ denotes the BPSK modulated signal, $G_j$ denotes the combined channel gain including the signal power, $n \sim \mathcal{N}(0, \sigma^2)$ denotes the additive white noise, and $J = 2$ or $3$. The extrinsic MI output for one user is evaluated based on the extrinsic MI inputs from other users. One approach is to use Monte Carlo simulations. Let $L_j^a$ and $L_j^e$ denote the input and output LLRs for user $j$, respectively. Assuming that the extrinsic LLR inputs are symmetric Gaussian
CHAPTER 3. JOINT NETWORK AND CHANNEL CODING FOR A JOINT RELAY SYSTEM

distributed with the MI \( I_j^a = I(L_j^a; X_j) \) for \( j = 1, \cdots, J \), we can generate samples of the input LLR \( L_j^a \), and compute the extrinsic MI output \( I_j^e = I(L_j^e; X_j) \) using the histogram of the output LLR \( L_j^e \). However, the computational cost of such Monte Carlo approach is quite high. To reduce the complexity of evaluating the EXIT functions, we propose the following approximations.

### Extrinsic MI approximation for a 2-user detector

The linear approximation (LA) for \( I_1^e \) is given by

\[
I_1^e = a_1 I_2^a + b_1,
\]

(3.3)

where the coefficients \( a_1 \) and \( b_1 \) can be determined from the boundary conditions that \( I_1^e|_{I_2^a=0} = I(X_1; Y) \) and \( I_1^e|_{I_2^a=1} = I(X_1; Y|X_2) \). Similarly, we can obtain the LA for \( I_2^e \).

### Extrinsic MI approximation for a 3-user detector

The quadratic approximation (QA) for \( I_1^e \) is given by

\[
I_1^e = a_1 I_2^a I_3^a + b_1 I_2^a + c_1 I_3^a + d_1,
\]

(3.4)

where the four coefficients can be determined from the boundary conditions that \( I_1^e|_{I_2^a=0, I_3^a=0} = I(X_1; Y), I_1^e|_{I_2^a=1, I_3^a=0} = I(X_1; Y|X_2), I_1^e|_{I_2^a=0, I_3^a=1} = I(X_1; Y|X_3), \) and \( I_1^e|_{I_2^a=1, I_3^a=1} = I(X_1; Y|X_2, X_3) \). Similarly, we can obtain the QA for \( I_2^e \) and \( I_3^e \). Note that from the channel symmetry when \( G_2 = G_3 \) we have \( b_1 = c_1 \).

We now evaluate the performance of the above MI approximations. For the channel model given in (3.2), we compare the extrinsic MI evolutions from both the approximations and the Monte Carlo simulations. For the 2-user MAC, we set \( G_1 = 1.5, G_2 = 1.0, \) and \( \sigma = 0.85 \). The output extrinsic MI \( I_1^e \) as a function of \( I_2^a \) is shown in Fig. 3.4. For the 3-user MAC, we set \( G_1 = 1.5, G_2 = 1.2, G_3 = 1.0, \) and \( \sigma = 0.80 \). The extrinsic MI \( I_1^e \) as a function of \( I_2^a \) given \( I_2^a = 0.40 \), and as a function of \( I_3^a \) given \( I_2^a = 0.70 \) is also shown in Fig. 3.4. It is seen that both the LA and the QA give good approximations for the MI evolutions for 2-user and 3-user MAC, respectively.

### 3.2.2 Overview of the Code Design

We first introduce the following notations.

Denote \( \{\lambda_j^{k(q)}\} \) and \( \{\lambda_j^{kR}\}, k = 1, 2, q = 1, 2, \) as the profiles from the edge perspective corresponding to the profiles \( \{\lambda_j^{k(q)}\} \) and \( \{\lambda_j^{kR}\} \), respectively. Denote \( \lambda_j^{k(q)} \) and \( \lambda_j^{kR} \) as the average
degrees of \{λ_j^{k(q)}\} and \{λ_j^{kR}\}, respectively. Denote \{ρ_j^k\} and \{ρ̂_j^k\} as the profiles of the LT input nodes with respect to LT output nodes from the node and edge perspectives, respectively. Note that \{ρ_j^k\} can be approximated by the Poisson distribution with an average degree \overline{ρ}_j^k \[27\]. For the IRA precode, denote \{γ_j\} and \{γ̂_j\} as the degree profiles for all variable nodes from the node and edge perspectives, respectively. We use the rate 0.95 IRA codes with all information nodes of degree-3, then \{γ_2 = 0.05, γ_3 = 0.95\}, and \{γ̂_2 = 0.0339, γ̂_3 = 0.9661\}. The check nodes for the IRA code have the same degree \(d_c\). Denote \(η_k^{(1)}\) and \(η_k^{(2)}\) as the ratios of the LT coded parity bits over the IRA coded bits for source \(S_k\) in the first and the second stages, respectively. Let \(R_k\) be the overall code rate for source \(S_k\). We have

\[
η_k^{(1)} = \frac{M^{(1)} - N_k}{N_k} = \frac{0.95t}{R_k} - 1
\]

\[
η_k^{(2)} = \frac{M^{(2)}}{N_k} = \frac{0.95(1 - t)}{R_k}.
\]

(3.5)

For both SC and RC, we first optimize the profiles \{λ_j^{1(1)}\} and \{λ_j^{2(1)}\} for the first stage. Then, given the optimized profiles for the first stage, we optimize the profiles for the second stage. For SC, since the coded bits by the relay are the same as those by both sources, we only need to optimize \{λ_j^{2(2)}\} and \{λ_j^{2(2)}\}. For RC, since the relay performs joint encoding using its own profiles, we need to optimize the profiles \{λ_j^{k(2)}\} and \{λ_j^{kR}\} for \(k = 1, 2\).

3.2.3 Code Design for the First Stage

We present the code design for the 2-user MAC in the first stage. Instead of assuming the same channel gains for both users as in \[32\], we consider the code design for a more general 2-user MAC with arbitrary channel gains.
CHAPTER 3. JOINT NETWORK AND CHANNEL CODING FOR A JOINT RELAY SYSTEM

\[ I^{(1)}_{ec} = \sum_{i,j} \gamma_i \rho_j^k \left( \sqrt{(j-1)[J^{-1}(I^{(1)}_{cv})]^2 + [J^{-1}(I^{(1)}_{dv})]^2 + i[J^{-1}(I^{(1)}_{pv})]^2} \right) \]  
(3.6)

\[ I^k_{vp} = \sum_{i,j} \gamma_i \rho_j^k \left( \sqrt{j[J^{-1}(I^{(1)}_{cv})]^2 + [J^{-1}(I^{(1)}_{dv})]^2 + (i-1)[J^{-1}(I^{(1)}_{pv})]^2} \right) \]  
(3.7)

\[ I^{(1)}_{vd} = \sum_{i,j} \gamma_i \rho_j^k \left( \sqrt{j[J^{-1}(I^{(1)}_{cv})]^2 + i[J^{-1}(I^{(1)}_{pv})]^2} \right) \]  
(3.8)

\[ I^{(1)}_{cv} = 1 - \sum_j \lambda_j^{k(1)} \left( \sqrt{(j-1)[J^{-1}(1 - I^{(1)}_{dc})]^2} \right) \]  
(3.9)

\[ I^{(1)}_{cd} = 1 - \sum_j \lambda_j^{k(1)} \left( \sqrt{j[J^{-1}(1 - I^{(1)}_{dc})]^2} \right) \]  
(3.10)

\[ I^k_{pv} = 1 - J \left( \sqrt{(d_c - 1)[J^{-1}(1 - I^k_{dc})]^2} \right) \]  
(3.11)

\[ I_{md}^{(1)} = (I_{vd}^{(1)} + \eta_k^{(1)} I_{cd}^{(1)}) / (1 + \eta_k^{(1)}) \]  
(3.12)

Since only the LT output nodes in the first stage are connected to the LT input nodes, \( \{ \rho_j^k \} \) is Poisson distributed with the mean \( \rho^k = \lambda^{k \text{LT}} \eta_k \). The extrinsic MI outputs from the MAC detector, \( I^{(1)}_{dc} (= I^{(1)}_{dv}) \), can be obtained by the LA in (3.3) with the extrinsic MI input from the other user. Based on the factor graph in the upper portion of Fig. 3.3 the extrinsic MI outputs \( I^{(1)}_{ec} \), \( I^k_{vp} \), \( I^{(1)}_{vd} \), \( I^{(1)}_{cv} \), \( I^k_{cd} \), \( I^k_{pv} \), and \( I_{md}^{(1)} \), are given respectively by equations (3.6) to (3.12) shown in the following page, where \( I^{(1)}_{ec} \), \( I^k_{vp} \), and \( I^{(1)}_{vd} \) are obtained by the message updates at the LT input nodes; \( I^{(1)}_{cv} \) and \( I^k_{cd} \) are obtained by the message updates at the LT output nodes; \( I^k_{pv} \) is obtained by the message updates at the IRA check nodes; \( I_{md}^{(1)} \) is the average extrinsic MI for LT input and output nodes; and 

\[ J(\sigma) \triangleq 1 - \int_{-\infty}^{\infty} \frac{e^{-(\xi - \sigma^2/2)^2/2\sigma^2}}{\sqrt{2\pi \sigma^2}} \log_2(1 + e^{-\xi}) d\xi. \]  
(3.13)

Take (3.6) as an example. The LT input nodes are connected to both the IRA check nodes and the LT output nodes with the separately defined profiles \( \{ \gamma_i \} \) and \( \{ \rho_j^k \} \). The extrinsic MI output from the message update at an LT input node of degree-\( i \) (corresponding to the IRA code) along one edge of degree-\( j \) to the LT output node (corresponding to the LT code) is given by \( J \left( \sqrt{(j-1)[J^{-1}(I^{(1)}_{cv})]^2 + [J^{-1}(I^{(1)}_{dv})]^2 + i[J^{-1}(I^{(1)}_{pv})]^2} \right) \). In the mixture, the fraction for \( i[J^{-1}(I^{(1)}_{pv})]^2 \) from the precoder is \( \gamma_i \) from the node perspective, and the fraction for \( (j -
from the LT code is $\tilde{\rho}_j^k$ from the edge perspective. Similarly, we can obtain other extrinsic MI outputs by (3.7)–(3.12).

Given an SNR value (defined as $P/\sigma^2$), we track the extrinsic MI evolution and find the convergence point of the extrinsic MI outputs $I_{vd}^{1(1)}$ and $I_{vd}^{2(1)}$ as follows. The order of MI update is first the multiuser detector, then the LT input nodes, the LT output nodes, and finally the mixed MI from LT codes to multiuser detector.

Algorithm 3.1 [EXIT-based performance analysis of 2-user MAC]

(a) Initialize $I_{vc}^{k(1)} = I_{vp}^{k(1)} = I_{cv}^{k(1)} = I_{pv}^{k(1)} = I_{vd}^{k(1)} = I_{md}^{k(1)} = I_{dv}^{k(1)} = I_{dc}^{k(1)} = 0$, for $k = 1, 2$.

(b) Update $I_{dv}^{k(1)} = I_{dc}^{k(1)}$ based on the LA for 2-user MAC (3.3).

(c) Update $I_{vc}^{k(1)}$ and $I_{vp}^{k(1)}$ according to (3.6) and (3.7), respectively.

(d) Update $I_{cv}^{k(1)}$ and $I_{pv}^{k(1)}$ according to (3.9) and (3.11), respectively.

(e) Update $I_{vd}^{k(1)}$, $I_{cd}^{k(1)}$, and $I_{md}^{k(1)}$ according to (3.8), (3.10), and (3.12), respectively.

(f) Iterate steps (b)-(e) until $I_{vd}^{1(1)}$ and $I_{vd}^{2(1)}$ converge or the maximum number of iterations is reached. If both $I_{vd}^{1(1)}$ and $I_{vd}^{2(1)}$ approach one, we declare the decoding success; otherwise, we declare the decoding failure.

Given the code profiles and $P/\sigma^2$, we evaluate the performance of iterative decoding using Algorithm 3.1. Define $(P/\sigma^2)_{th}$ as the minimum SNR that supports the error-free decoding, which can be obtained by a bisection search. Given the rate constraints, the code optimization is to find the code profiles that minimize $(P/\sigma^2)_{th}$, given by

$$\min \{\lambda_j^{k(1)}\}_{k=1,2} \quad (P/\sigma^2)_{th}$$

s.t. $I_{vd}^{k(1)} \rightarrow 1$; Poisson distributed $\{\rho_j^k\}$,

$$\overline{\rho}^k = \lambda^{k(1)}_{j} \eta_j^{(1)}; \quad k = 1, 2.$$  

We solve the above code optimization using the differential evolution (DE) method [14,33]. Although strictly speaking the optimal profile for one code rate is no longer optimal for other code rates, due to the rate compatibility of the rateless codes, by generating different amount of rateless parity bits the optimized profile for one rate also performs well for other rates, as will be illustrated in Section 3.3.2.
3.2.4 Code Design for SC

The middle portion of Fig. 3.3 shows the extrinsic MI evolutions for SC. The extrinsic MIs for the LT output nodes in the two stages are treated separately due to the different SNRs of the received signals. Since more LT coded bits are generated in the second stage, the mean of the profile \( \{ \rho^k_j \} \) is given by

\[
\bar{\rho}^k = \frac{\lambda^k(1)\eta^k_{(1)} + \lambda^k(2)\eta^k_{(2)}}{\rho^k}.
\]  

Again, the extrinsic MI \( I^k_{dc}(= I^k_{dv}) \) and \( I^k_{dc} \) from the APP detectors for user \( k \) is obtained by the LA in (3.3).

\[
I^k_{vc}^{(1)} = \sum_{i,j} \gamma_i \rho^k_j (\sqrt{j \nu^k_{cv} - [J-1(I^k_{dv})^2] + [J-1(I^k_{dv})^2] + i[J-1(I^k_{pv})^2]}),
\]  

\[
I^k_{vc}^{(2)} = \sum_{i,j} \gamma_i \rho^k_j (\sqrt{j \nu^k_{cv} - [J-1(I^k_{dv})^2] + [J-1(I^k_{dv})^2] + i[J-1(I^k_{pv})^2]}),
\]  

\[
I^k_{vp} = \sum_{i,j} \gamma_i \rho^k_j (\sqrt{j \nu^k_{cv} + [J-1(I^k_{dv})^2] + [J-1(I^k_{pv})^2]}),
\]  

\[
I^k_{cv}^{(2)} = 1 - \sum_{j} \lambda_j^k (\sqrt{j \nu^k_{cv} + [J-1(I^k_{dv})^2] + [J-1(I^k_{pv})^2]}),
\]  

\[
I^k_{vd} = \sum_{i,j} \gamma_i \rho^k_j (\sqrt{j \nu^k_{cv} + i[J-1(I^k_{pv})^2]}),
\]  

\[
I^k_{cd}^{(2)} = 1 - \sum_{j} \lambda_j^k (\sqrt{j \nu^k_{cv} + i[J-1(I^k_{dv})^2]}).
\]  

Denote \( \kappa^k_{(1)} \) and \( \kappa^k_{(2)} \) as the fractions of edges connected to the LT output nodes in the first and second stages, respectively. We then have

\[
\kappa^k_{(1)} = \frac{\lambda^k(1)\eta^k_{(1)} + \lambda^k(2)\eta^k_{(2)}}{\rho^k},
\]

\[
\kappa^k_{(2)} = 1 - \kappa^k_{(1)}. \]  

The average variance \( \nu^k_{cv} \) of the extrinsic LLRs for all LT coded bits is then given by

\[
\nu^k_{cv} = \kappa^k_{(1)}[J-1(I^k_{cv})^2] + \kappa^k_{(2)}[J-1(I^k_{cv})^2].
\]  

Based on the factor graph in the middle portion of Fig. 3.3, we obtain the updates of \( I^k_{cv}^{(1)} \), \( I^k_{cd}^{(1)} \), \( I^k_{pv} \), and \( I^k_{md} \) by (3.9), (3.10), (3.11), and (3.12), respectively, using the profile \( \{ \rho^k_j \} \) with the average
degree in (3.15). The updates of $I^{(1)}_{vc}$, $I^{(2)}_{vc}$, $I^{(1)}_{vp}$, $I^{(2)}_{vd}$, and $I^{(2)}_{cd}$ are given respectively by equations (3.16) to (3.21) shown in this page. Take (3.17) as an example. Since the average variance of the LLRs from the LT output nodes to LT input nodes is given by $\nu^{k}_{cv}$, the variance of the LLRs from the LT input node of degree-$i$ (corresponding to the IRA code) along one edge of degree-$j$ (corresponding to the LT code) to the LT output node in the second stage is given by 

$$j\nu^{k}_{cv} - [J^{-1}(I^{(2)}_{dc})]^2 + [J^{-1}(I^{(2)}_{dv})]^2 + i[J^{-1}(I^{k}_{pe})]^2,$$

where we subtract the input variance from the LT output node in the second stage. In the MI mixture, the fraction for $j\nu^{k}_{cv} - [J^{-1}(I^{(2)}_{dc})]^2$ is $\rho^{k}_{j} j不了解^{k}_{j} (2)$ from the edge perspective, and the fraction for $i[J^{-1}(I^{k}_{pe})]^2$ is $\gamma_{i}$ from the node perspective.

The performance of the iterative receiver at the destination under SC can be evaluated similarly to that for the 2-user MAC. We update the output MI sequentially for the multiuser detector by (3.3), the LT input nodes by (3.23)–(3.18), the LT output nodes by (3.9), (3.11), and (3.19), and the mixed output MI to the channel by (3.10), (3.12), (3.20), and (3.21). The convergence criterion is the same as that for the 2-user MAC. The code design is similar to that for the 2-user MAC, except that the distribution $\{\rho^{k}_{j}\}$ is poisson distributed with the mean $\bar{\rho}^{k} = \lambda^{k(1)} \eta^{(1)}_{k} + \lambda^{k(2)} \eta^{(2)}_{k}$.

It can be seen that, unlike the code design for the LDPC extending as in [20], here the constraints on the degree profiles for two stages are very simple.

### 3.2.5 Code Design for RC

The bottom portion of Fig. 3.3 shows the MI evolution for RC. Besides the LT coded bits generated at the sources in both stages, the LT coded bits generated at the relay can be viewed as additional connections to the IRA precoded nodes. Therefore, the average degree of the profile $\{\rho^{k}_{j}\}$ is given by

$$\bar{\rho}^{k} = \lambda^{k(1)} \eta^{(1)}_{k} + \lambda^{k(2)} \eta^{(2)}_{k} + \lambda^{R} \eta^{(2)}_{k}. \tag{3.24}$$

First, we update the extrinsic MI $I^{(1)}_{dc} = I^{(1)}_{dv}$ and $I^{(2)}_{dc} = I^{(2)}_{dv}$ using the LA for the first stage, and update $I^{(1)}_{dc}$, $I^{(2)}_{dc}$, and $r^{(2)}_{dc}$ using the QA for the second stage. Denote $\kappa^{(1)}_{k}$, $\kappa^{(2)}_{k}$, and $\kappa^{R}_{k}$ as the fractions of edges connected to the LT output nodes in the first stage, the second stage, and...
the LT output nodes at the relay, respectively. We have

\[ \kappa_k^{(1)} = \frac{\lambda_k^{(1)} \eta_k^{(1)}}{\lambda_k^{(1)} \eta_k^{(1)} + \lambda_k^{(2)} \eta_k^{(2)} + \rho_k^R \eta_k}, \]

\[ \kappa_k^{(2)} = \frac{\lambda_k^{(2)} \eta_k}{\rho_k^R}, \]

\[ \kappa_k^R = 1 - \kappa_k^{(1)} - \kappa_k^{(2)}. \]  

The average variance from the extrinsic MIs of all LT output nodes is given by

\[ \nu_{cv}^k = \kappa_k^{(1)} [J^{-1}(I_{cv}^{k(1)})]^2 + \kappa_k^{(2)} [J^{-1}(I_{cv}^{k(2)})]^2 + \kappa_k^R [J^{-1}(I_{cv}^{r(2)})]^2. \]  

\[ I_{vc}^{kr} = \sum_{i,j} \gamma_i \rho_j^k \left( \sqrt{J^{-1}(I_{cv}^{kr})^2 + I^{-1}(I_{dc}^{k(i)})^2 + i[J^{-1}(I_{vc}^{k(i)})]^2} \right) \]  

\[ I_{cv}^{1r} = 1 - \sum_{j,i} \lambda_{ij}^r \lambda_i \left( \sqrt{(j-1)[J^{-1}(I_{vc}^{1r})^2 + i[J^{-1}(I_{vc}^{r(2)})]^2 + J^{-1}(1 - I_{dc}^{(2)})^2]} \right) \]  

\[ I_{vc}^{2r} = 1 - \sum_{j,i} \lambda_{ij}^r \lambda_i \left( \sqrt{(j-1)[J^{-1}(I_{vc}^{2r})^2 + i[J^{-1}(I_{vc}^{r(2)})]^2 + J^{-1}(1 - I_{vc}^{r(2)})^2]} \right) \]  

\[ I_{vc}^{r(2)} = 1 - \sum_{j,k} \lambda_{jk}^r \lambda_i \left( \sqrt{(j-1)[J^{-1}(1 - I_{vc}^{r(2)})]^2 + i[J^{-1}(1 - I_{vc}^{r(2)})]^2 + J^{-1}(1 - I_{vc}^{r(2)})^2]} \right) \]  

Based on the factor graph in the lower portion of Fig. 3.3, we obtain the updates of $I_{vc}^{k1}$, $I_{vc}^{k2}$, $I_{vc}^{k1}$, $I_{vc}^{k2}$, $I_{vc}^{k1}$, $I_{vc}^{k2}$, $I_{vc}^{k1}$, $I_{vc}^{k2}$, and $I_{vc}^{k2}$ by (3.9), (3.10), (3.11), (3.12), (3.13), (3.14), (3.15), (3.16), (3.17), (3.18), (3.19), (3.20), and (3.21), respectively, using the profile $\{\rho_j^k\}$ with the average degree given in (3.24). With $\nu_{cv}^k$ given in (3.26), we obtain the updates of $I_{vc}^{kr}$, $I_{vc}^{kr}$, and $I_{vc}^{r(2)}$, given by equations (3.27) to (3.30) in the following page. Take (3.28) as an example. The extrinsic MI output from the message update at the LT output nodes of degree-$i$ and -$j$ with respect to $S_1$ and $S_2$, respectively, to the LT input nodes of $S_1$ is given by (3.31) in the following page, where the MI evolution rule is the same as that for LDPC check nodes. In the MI mixture, since the MI is to the LT input nodes of $S_1$, the fraction of $(j-1)[J^{-1}(1 - I_{vc}^{r(2)})^2]$ is $\lambda_{ij}^r$ from the edge perspective, and the fraction of $i[J^{-1}(1 - I_{vc}^{r(2)})^2]$ is $\lambda_{ij}^r$ from the node perspective.
The performance of the iterative receiver at the destination under RC can be evaluated similarly. We update the output MI sequentially for the multiuser detector by the LA (3.3) and QA (3.4), the LT input nodes by (3.16), (3.17), (3.18), and (3.27), the LT output nodes by (3.9), (3.11), (3.19), (3.28), and (3.29), and the mixed output MI to the channel by (3.10), (3.12), (3.20), (3.21), and (3.30). The convergence criterion is the same as that for the 2-user MAC. The code design is similar to that for the 2-user MAC, except that the distribution \( \{ \rho_j^k \} \) is poisson distributed with the mean \( \bar{\rho}_j^k = \lambda_j^{(1)} \eta_j^{(1)} + (\lambda_j^{(2)} + \lambda_j^{(R)}) \eta_j^{(2)} \).
CHAPTER 3. JOINT NETWORK AND CHANNEL CODING FOR A JOINT RELAY SYSTEM

Figure 3.3: Factor graphs for the decoding of relay MAC (upper portion), SC (middle portion) and RC (lower portion).
Figure 3.4: EXIT charts for multiuser detectors based on LA and QA.
3.3 Simulation Results

3.3.1 Achievable Rates

The achievable rate region for the 4-node joint relay system is the union of the two achievable rate regions corresponding to the entire transmission with and without the relay assistance. Without the relay assistance, the achievable rate region is simply the 2-user MAC capacity region for both sources. Here we give the achievable rate region with the relay assistance, as follows.

\[
R_1 \leq R_1(t) = \min \{ tI(X_1^{(1)}; Y_R^{(1)}|X_2^{(1)}), tI(X_1^{(1)}; Y_D^{(1)}|X_2^{(1)}) + (1-t)I(Z_1; Y_R^{(2)}|X_2^{(2)}) \}
\]

\[
R_2 \leq R_2(t) = \min \{ tI(X_2^{(1)}; Y_R^{(1)}|X_1^{(1)}), tI(X_2^{(1)}; Y_D^{(1)}|X_1^{(1)}) + (1-t)I(Z_2; Y_R^{(2)}|X_1^{(2)}) \}
\]

\[
R_1 + R_2 \leq R_{12}(t) = \min \{ tI(X_1^{(1)}X_2^{(1)}; Y_R^{(1)}), tI(X_1^{(1)}X_2^{(1)}; Y_D^{(1)}) + (1-t)I(Z_{12}; Y_D^{(2)}) \} \quad (3.32)
\]

\[
I(W_R^{(2)}X_1^{(2)}; Y_D^{(2)}|X_2^{(2)}) = h(Y_D^{(2)}|X_2^{(2)}) - h(Y_D^{(2)}|X_1^{(2)}X_2^{(2)}W_R^{(2)})
\]

\[
= h(h_{1D}\sqrt{P_1^{(2)}X_1^{(2)} + h_{RD}\sqrt{P_R^{(2)}W_R^{(2)} + n_2^{(D)}}}) - h(n_2^{(D)}) \quad (3.33)
\]

We use the upper-case letters to denote the random variables corresponding to the signals in Fig. 3.1. Since the transmissions in the second stage do not contain any new information, following [19], we obtain the achievable rate region, given by (3.32), where \(Z_1 = X_1^{(2)}, Z_2 = X_2^{(2)}, Z_{12} = (X_1^{(2)}, X_2^{(2)})\) for SC, and \(Z_1 = (W_R^{(2)}, X_1^{(2)}), Z_2 = (W_R^{(2)}, X_2^{(2)}), Z_{12} = (W_R^{(2)}, X_1^{(2)}, X_2^{(2)})\) for RC. In the above region, the first and second terms in \(\min\{\}\) correspond to the successful decoding at the relay and the destination, respectively. Since the input signals for RC in the second stage are correlated, the terms \(Z_1, Z_2,\) and \(Z_{12}\) are derived from the achievable rate region of correlated sources for MAC [34].

The mutual information involved in (3.32) is computed based on the channel model in Fig. 3.1 under BPSK modulation. For example, for RC we have (3.33) shown in the following page, where the probability density function of \(h_{1D}\sqrt{P_1^{(2)}X_1^{(2)} + h_{RD}\sqrt{P_R^{(2)}W_R^{(2)} + n_2^{(D)}}}\) is evaluated assuming that the pair \((X_1^{(2)}, W_R^{(2)})\) is uniformly distributed on \(\{(\pm 1, \pm 1)\}\). The other mutual information terms in (3.32) can be similarly evaluated.
The partition of the two transmission stages which maximizes the sum rate of both sources, denoted as \( t_{\text{opt}} \), is given by

\[
t_{\text{opt}} = \arg \max_{0 \leq t \leq 1} \min\{R_1(t) + R_2(t), R_{12}(t)\}.
\] (3.34)

Given the overall code rates \( R_1 \) and \( R_2 \), the achievable limit of \( P/\sigma^2 \) is defined as the SNR where the point \((R_1,R_2)\) touches the boundary of the capacity region given in (3.32). It can be computed using a bisection search. For both SC and RC, the achievable limit of \( P/\sigma^2 \) is used as the benchmarks for the performance evaluations of optimized codes.

### 3.3.2 Performance of Optimized Codes for 2-user MAC at Relay

We present simulation results of the optimized Raptor codes for the 2-user MAC in the first stage. The performance of the optimized Raptor codes are compared with that of the optimized LDPC codes for 2-user MAC and the optimized Raptor codes for single-user channel in [4]. For simplicity, we consider the combined channel gains \( G_{kR} = h_{kR} \sqrt{P^{(1)}_k} \) for \( k = 1, 2 \) for the relay signal model in Fig. 3.1 and the AWGN \( n^{(1)}_R \sim \mathcal{N}(0, \sigma^2) \). The performance is plotted against \( P_n = 20 \log_{10}(1/\sigma) \).

Fig. 3.5 illustrates the performance of the optimized Raptor codes and LDPC codes for the 2-user MAC. To combat the mutual signal cancellation in the MAC, we design the check node profile and left the variable node profile to be concentrated. We consider the following two settings. In the first setting, \( G_{1R} = G_{2R} = 1.0 \), and the code length \( K_1 = K_2 = 38000 \), and \( M^{(1)} = 76000 \), denoted as \((0.5, 0.5)\). In the second, \( G_{1R} = 1.0 \), \( G_{2R} = 1.5 \), and the code length \( K_1 = 38000 \), \( K_2 = 76000 \), and \( M^{(1)} = 114000 \), denoted as \((1/3, 2/3)\). For both settings, the optimized Raptor codes perform the same as the optimized LDPC codes, within 1 dB away from the Shannon limit for 2-user MAC denoted as SL, and outperform by about 1 dB the Raptor codes from [4] denoted as AWGN. Without doping, for the case of \( G_{1R} = G_{2R} = 1.0 \), the code with optimized profiles cannot achieve the error-free decoding even if there is no channel noise.

We next evaluate the rate compatibility of the optimized Raptor codes for the 2-user MAC. The channel gains \( G_{1R} = G_{2R} = 1.0 \), and the information length \( K_1 = K_2 = 38000 \). Two rate settings, rate 0.4 and rate 0.5, are considered, for which \( M^{(1)} = 95000 \) and \( M^{(1)} = 76000 \), respectively. We use the optimized code profiles for rate 0.5 to generate more coded bits to reach rate 0.4, and use the optimized code profiles for rate 0.4 to generate fewer coded bits to reach rate 0.5. The performance is illustrated in Fig. 3.6. The optimized profile for rate 0.5 operating at rate 0.4 performs the
same as the optimized profile of rate 0.4. Operating at the code rate 0.5, the optimized profile for rate 0.4 exhibits only 0.15 dB performance loss compared with the optimized profile for rate 0.5. These results demonstrate the flexible rate compatibility of the designed Raptor codes for the 2-user MAC.

Since Raptor codes perform the same as LDPC codes for fixed code rate and exhibit flexible rate compatibility, we employ them for transmissions at both sources and the relay. The rate compatibility also makes them work well for block fading channels \([35]\).

![Figure 3.5: Performance of the optimized Raptor codes for the 2-user MAC.](image)

### 3.3.3 Performance of Optimized Codes for Joint Relay Systems

Consider the relay model where \(d_{1R} = d_{2R} = d_{RD} = 0.5\), \(d_{1D} = d_{2D} = 1.0\). Assume the power decay factor \(\alpha = 4\) and the unit noise power at \(R\) and \(D\). We consider both power allocation schemes. For PA-1, we set \(P_{11} = P_{21} = P/2\), \(P_{12} = P_{22} = \frac{2}{5}P\), and \(P_{R2} = P/5\). For PA-2, we set \(P_{11} = P_{21} = P_{12} = P_{22} = P_{R2}\). For SC, we set \(\beta_1 = \beta_2\). Based on the relay model and both power allocation schemes, we first optimize the code profiles for the entire relay system for
Figure 3.6: Rate compatibility of the optimized Raptor codes for the 2-user MAC.

AWGN channels, and then test the performance of the optimized profiles for AWGN channels in block fading channels. Note that for the block fading channel, although the tested profiles are only optimal for the average of all channel realizations, they still outperform the optimized profile for the direct-link channel.

### 3.3.3.1 AWGN Channel

Assume that all small-scale fading gains are one. We set the sum rate of two sources to be 0.45. Table 3.1 lists the designed power of the entire relay system and the optimal stage division $t_{opt}$ for both SC and RC considering both $PA-1$ and $PA-2$, denoted as “RC-1”, “RC-2”, “SC-1”, and “SC-2”, as well as the length of the optimized codes.

In simulations, we fix the unit channel noise power at both the relay and the destination, and increase the overall average power $P$ starting from the designed power in Table 3.1. The performance of the optimized codes is shown in Fig. 5.1. For SC, the optimized codes are compared with the codes using the optimized profiles of the first stage for both stages, denoted as “SC-i” and “SC-i
(Mac2)”, respectively, for $PA_i$. For RC, the optimized codes are compared with the codes using the optimized profile of single-user channel for all the six profiles, denoted as “RC-$i$” and “RC-$i$(AWGN)” for $PA_i$, respectively. The predicted thresholds are shown in the brackets following the four optimized profiles. The designed codes perform about 1.5dB away from the achievable rate limits based on (3.32), denoted as “Limit” in Fig. 5.1. It is seen that the code optimizations provide large performance gains (5 8 dB) for the entire system.

![Figure 3.7: Performance of the optimized joint relay system in AWGN channels.](image)
3.3.3.2 Block Fading Channel

Assume that the small-scale fading gains are Rayleigh distributed with unit power and remain constant within one transmission block. For each fading realizations, we compute the achievable rate regions of the relay system, and quantize the rate ratio of two sources to \((0.0, 1.0), (0.1, 0.9), \ldots, (0.9, 0.1), \text{ or } (1.0, 0.0)\) to maximize the sum rate of two sources. We fix the total number of information bits to be 7600, and dynamically adjust the number of information bits for both sources in different fading realizations. The first and second stages end when the relay and destination successfully decode the information of both sources, respectively. If the destination successfully decodes before the relay, we start the transmission of the next block. We set the maximum number of coded bits for each stage to be 38000. When the maximum number of coded bits is reached, we declare a block error if the decoding still fails.

In the simulations, we fix the power \(P\) to be the designed power in Table 3.1, and test the performance of the entire relay system for different noise power \(\sigma^2\) where \(n_R^{(1)}, n_D^{(1)}, n_D^{(2)} \sim \mathcal{N}(0, \sigma^2)\). The performance is plotted against \(P_n = 20 \log_{10}(1/\sigma)\). For SC, we set the two profiles for the first stage to be the optimized profile for the single-user channel and optimize the profiles for the second stage. In Fig. 3.8, the optimized codes for SC (denoted as Opt) provide about 3% throughput improvements over the codes employing the optimized single-user channel profile for both stages (denoted as AWGN). In Fig. 3.9, the optimized profiles for RC (denoted as Opt) significantly outperform (over 100%) those employing the optimized single-user channel profile for both stages (denoted as AWGN). Note that the RC is more practical than SC since the perfect synchronization between the relay and both sources is not required.
Figure 3.8: Throughput performance of the joint relay system in block fading channels employing the optimized profiles for AWGN channels for SC.
Figure 3.9: Throughput performance of the joint relay system in block fading channels employing the optimized profiles for AWGN channels for RC.
Chapter 4

Constrained Partial Group Decoder for $K$-user Interference Channels

Interference channel is a fundamental building block of the wireless networks. Due to the ever-shrinking network sizes and the increasing demands for achieving higher spectral efficiency, the emerging wireless networks operate in an interference-limited regime. Motivated by such demands, investigating different aspects of interference channel has resulted in various recent developments for further understanding the fundamental limits of these channels \cite{36,37}.

In the rich literature of interference channel it is well-understood that while a receiver is not ultimately interested in decoding the messages of the interferers, decoding them (fully or partially) is often advantageous for recovering its desired message \cite{38}. Motivated by this premise some recent developments for the $K$-user interference channels propose that each receiver should partition the interfering transmitters into two groups; one group to be fully decoded along with the designated transmitter and the other to be treated as Gaussian noise \cite{39,41}.

One major advantage of fully decoding a transmitter is that it suffices to assign only one codebook to that transmitter, which is appealing for practical purposes. One drawback of such decoding, on the other hand, is that the receivers will give up the freedom of decoding only a fraction of an interferer, which in some instances can be more beneficial than fully decoding it. Here we consider allocating more than one codebook to each transmitter and provide the receivers with the freedom of deciding about what interferers to decode as well as determining what fraction of such interferers need to be decoded.

Assigning multiple codebooks enables splitting the message of each transmitter to multiple
layers, each drawn from one separate codebook. Such rate splitting can be seen as the generalization of the Han-Kobayashi scheme \cite{38} for the 2-user interference channel, that splits the message of each transmitter into two layers. Splitting the messages into multiple layers provides the receiver with the freedom of partially decoding the interferers and consequently with the advantage of sustaining reliable communication at higher rates. Attaining such gains is at the cost of facing certain practical complexities and obstacles. In particular when each transmitter has multiple codebooks, each receiver has to identify the best subset of the codebooks to be decoded, which can be computationally prohibitive. Moreover, the interference channel is a distributed system which might allow only very limited coordination among different users, while multi-layer transmission schemes and decoding the interference by receivers necessitates an interplay among the design of the different transmitters and receivers. Finally, as the transmitters need to be decodable at multiple receivers designing the channel codes becomes considerably more complicated.

We first introduce and discuss the notion of constrained partial group decoder (CPGD) and then address the challenges pertinent to its implementation. We discuss these challenges through solving a fair rate allocation problem for the $K$-user interference channel. Specifically, in the first step we propose the so-called CPGD, and show that it can solve the rate allocation problem in a computationally efficient way and with limited coordination among different users. In the second step we focus on designing a practical coding scheme that can be used in conjunction with the proposed CPGD structure.

### 4.1 System Descriptions

#### 4.1.1 Channel Model

Consider a fully connected $K$-user interference channel. We denote the wireless channel from the $j^{th}$ transmitter to the $i^{th}$ receiver by $h_{i,j}$ for $i, j \in \{1, \ldots, K\}$. We assume quasi-static block fading channels such that the channel coefficients are fixed during the transmission of $N$ symbols and change to some other independent states afterwards. By defining $x_j[n]$ as the transmitted signal by the $j^{th}$ transmitter for $n \in \{1, \ldots, N\}$ the received signal by the $i^{th}$ receiver is given by

$$y_i[n] = \sum_{j=1}^{K} h_{i,j} \, x_j[n] + v_i[n] \, , \text{ for } i \in \{1, \ldots, K\} \, ,$$

(4.1)
where $v_i[n] \sim N(0, \sigma^2)$ accounts for the additive white Gaussian noise (AWGN). The term $h_{i,i} x_i[n]$ contains the intended signal for the $i^{th}$ receiver and the remaining summands constitute interference and noise. Also denote the transmission power of all transmitters as $P \triangleq \mathbb{E}(|x_j[n]|^2)$.

### 4.1.2 Layered Encoding

For allowing the receivers to decode the transmitters partially, the message of each transmitter is split into smaller layers each drawn from an independent codebook (rate splitting). Let us denote the number of codebooks (layers) that we allocate to transmitter $j$ by $L_j$. Also denote the set of codebooks of transmitter $j$ by $\mathcal{C}_j \triangleq \{C_{j,1}, \ldots, C_{j,L_j}\}$ for $j = 1, \ldots, K$. By denoting the codeword drawn from codebook $C_{j,k}$ by $x_{j,k}[n]$ from (4.1) we obtain

$$y_i[n] = \sum_{j=1}^{K} h_{i,j} \sum_{k=1}^{L_j} x_{j,k}[n] + v_i[n], \quad \text{for} \quad i \in \{1, \ldots, K\}.$$  

We adopt equal power allocation for all the layers $x_{j,k}$ at the transmitter $j$. By defining $R_{j,k}$ and $R_j$ as rates of codebook $C_{j,k}$ and transmitter $j$, respectively, we have $R_j = \sum_{k=1}^{L_j} R_{j,k}$. We use $(j, k)$ to denote the index of codebook $C_{j,k}$ for $1 \leq j \leq K$ and $1 \leq k \leq L_j$, and define $\mathcal{K}$ as the set of all such indices. For any set $\mathcal{A} \subseteq \mathcal{K}$ we define the rate vector $\mathbf{R}_A \triangleq [R_{j,k}]_{(j,k) \in \mathcal{A}}$. In this chapter all the rates are in bits/sec/Hz and all the logarithms are in base 2.

### 4.1.3 Constrained Partial Group Decoding (CPGD)

Motivated by the premise that decoding the interference (fully or partially) along with the desired signal is sometimes beneficial, we introduce the notion of constrained partial group decoding. This principle has been the basis of many developments in the rich literature on the interference channel. In particular, the Han-Kobayashi scheme for the 2-user interference channel uses the rate splitting technique in order to allow each receiver to decode a part of the message of the interfering transmitter. Here we consider the general $K$-user interference channel for any arbitrary $K \geq 2$.

Upon determining what codebooks to be decoded at each receiver, which hinges on the utility function that we seek to optimize for the network, each receiver employs a successive decoding procedure. In each stage a subset of the layers are jointly decoded via the maximum likelihood (ML) decoding, after subtracting the already decoded layers from the received signal, and by
treat the remaining layers as AWGN. In order to control the complexity of ML decoding we constrain the number of layers being jointly decoded at each stage to be at most $\mu$.

We say that a given ordered partition $Q_i \triangleq \{Q^i_1, \ldots, Q^i_{p_i}, Q^i_{p_i+1}\}$ of $\mathcal{K}$ (the set of the indices of all codebooks) is valid if all the following conditions are satisfied.

1. $|Q^i_m| \leq \mu$ for $m \in \{1, \ldots, p_i\}$;
2. All layers of transmitter $i$, i.e., $\{x_{i,k}\}_{k=1}^{L_i}$, are included in $\{Q^i_1, \ldots, Q^i_{p_i}\}$;
3. The rate vector $R_{Q^i_m}$ is decodable at the $m^{th}$ stage of the successive decoding procedure for $m \in \{1, \ldots, p_i\}$.

For a given valid partition $Q^i_k$ of $\mathcal{K}$, the $i^{th}$ receiver decodes the layers included in $\{Q^i_1, \ldots, Q^i_{p_i}\}$ successively in $p_i$ stages while those in $Q^i_{p_i+1}$ are always treated as AWGN. More specifically, in the $m^{th}$ stage, the $i^{th}$ receiver jointly decodes the layers in $Q^i_m$ via ML decoding assuming the AWGN variance is $\Sigma^i_m$.

1. Initialize $m = 1$.
2. Compute
   \[
   \Sigma^i_m = \sigma^2 + \sum_{n=m+1}^{p_i+1} \sum_{(j,k) \in Q^i_n} \frac{|h_{i,j}|^2 P}{L_j},
   \]
   and jointly decode the layers in $Q^i_m$ via ML decoding assuming the AWGN variance is $\Sigma^i_m$.
3. Update $y_i[n] \leftarrow y_i[n] - \sum_{j \in Q^i_m} h_{i,j}\hat{x}_j$ where $\hat{x}_j$ is the decision made on $x_j$ for $j \in Q^i_m$ and update $m \leftarrow m + 1$.
4. If $m = p_i+1$ stop and otherwise go to step 2.

Determining the optimal valid partition of $\mathcal{K}$ and the corresponding rate vectors supported by such partition is the task of the CPGD and is the subject of the next section. Whenever we are not using Gaussian codebooks, decodable pertains to the specific channel encoders and decoders deployed.
4.1.3.1 A Simple Example of CPGD

In Fig. 4.1, we show a simple illustrative example of the proposed CPGD for an interference channel with four users, where users 1 and 3 contain 2 layers and users 2 and 4 contain one layer. The set of all layers is given by \{((1, 1), (1, 2), (2, 1), (3, 1), (3, 2), (4, 1))\}. We show the CPGD at the receiver of user 3, which aims to decode layers (3, 1) and (3, 2). Shown in Fig. 4.1 layers (2, 1) and (3, 2) are decoded in the first stage; layers (4, 1) and (1, 2) are decoded in the second stage; and layer (3, 1) is decoded in the third stage. After the three stages, layers (3, 1) and (3, 2) are decoded and the decoding terminates. We have \( p_3 = 4 \), \( Q_3^1 = \{(2, 1), (3, 2)\} \), \( Q_3^2 = \{(4, 1), (1, 2)\} \), \( Q_3^3 = \{(3, 1)\} \), and \( Q_3^4 = \{(1, 1)\} \).

![Diagram](image)

Figure 4.1: An simple illustrative example for constrained partial group decoding.

4.2 Rate Allocation for CPGD

4.2.1 Problem Statement

Assume that the users in a \( K \)-user interference channel are operating at some decodable rate vector \( R_K = [R_{j,k}]_{(j,k) \in K} \), where \( R_{j,k} \) is the rate of the \( k^{th} \) layer of transmitter \( j \). \( R_K \) can be the rate vector achievable by some decoding scheme inferior to the CPGD, e.g., the single-user decoders where all interference is treated as noise. Employing CPGDs allows for further incrementing the rate vector beyond \( R_K \). Such further rate increment is of particular interest when they also satisfy a fairness criterion. We consider maximizing the sum-rate of the network such that the rates of all
layers are incremented equally, i.e.,

$$\begin{align*}
\text{max} & \quad x , \\
\text{s.t.} & \quad \tilde{R}_K = R_K + x \cdot 1_{1 \times L} \text{ is decodable} ,
\end{align*}$$

(4.3)

where $1_{1 \times L} \triangleq [1, \ldots, 1]$ and $L = \sum_{j=1}^{K} L_j$. Note that maximizing $x$ is equivalent to maximizing $\sum_{j=1}^{K} \sum_{k=1}^{L_j} (R_{j,k} + x)$, which is the sum-rate of the $K$-user interference channel. Parameter $x$ is an intermediate parameter that leverages solving the fair sum-rate optimization problem of interest. The physical meaning of $x$ is the amount of change in the rate of each codebook. Due to the imposed fairness constraint, we require that the rates of all codebooks are incremented/decremented equally and aim to maximize the sum-rate subject to this fairness constraint. As will be made clear later, the number of layers allocated to each transmitter depends on the strength of the channel for that user, such that users with stronger links are assigned more layers. Therefore, such freedom in assigning different numbers of layers to the transmitters along with incrementing the rates of all layers equally (as required by (4.3)) has the advantage that it facilitates assigning higher rates to the users with stronger links. Let us also define $\tilde{R}_{j,k}$ as the rate of the $k^{th}$ layer of the $j^{th}$ transmitter after solving the above rate allocation problem and further define $r_{j,k}$ as the corresponding rate increment, i.e., $r_{j,k} \triangleq \tilde{R}_{j,k} - R_{j,k}$. For any set $A \subseteq \mathcal{K}$, we denote the rate vectors $\tilde{R}_A \triangleq [\tilde{R}_{j,k}]_{(j,k) \in A}$ and $r_A \triangleq [r_{j,k}]_{(j,k) \in A}$.  

4.2.2 Optimal Partitions and Rates

We propose an algorithm for solving (4.3) when the receivers employ CPGDs. The underlying motivation for the proposed algorithms is to alleviate the complexities associated with obtaining the best decodable set on one hand, and to control the amount of information exchange among the users on the other hand. The algorithm is comprised of two sub-routines: the first one aims to find the optimal partition for each receiver in a computationally efficient way and is executed locally by each receiver (Algorithm 4.1); and the second combines such local optimal solutions to solve the global fair sum-rate optimization problem (Algorithm 4.2).

For any two disjoint sets $\mathcal{U}$ and $\mathcal{V}$ that are subsets of $\mathcal{K}$, let $\mathcal{C}_i(\mathcal{U}, \mathcal{V})$ denote the achievable rate region supported by the $i^{th}$ receiver for jointly decoding the layers in $\mathcal{U}$ via ML decoding while the layers $\{\mathcal{K}\setminus(\mathcal{U} \cup \mathcal{V})\}$ have already been successfully decoded and subtracted and those in $\mathcal{V}$ are being
treated as AWGN. Let us denote the channel that conveys the layer $x_{j,k}$ from the $j^{th}$ transmitter to the $i^{th}$ receiver by $h_{i,j}^{k}$, for which we clearly have $h_{i,j}^{k} = h_{i,j}$ for $k = 1, \ldots, L_j$. Therefore, $C_i(\mathcal{U}, \mathcal{V})$ can be characterized as

$$C_i(\mathcal{U}, \mathcal{V}) = \left\{ R \in \mathbb{R}^{[d]}_+ \mid \sum_{j \in \mathcal{D}} R_j \leq R_i(\mathcal{D}, \mathcal{V}), \forall \mathcal{D} \subseteq \mathcal{U} \right\}, \quad (4.4)$$

where upon defining $h_{i,A} \triangleq \left[ \sqrt{\frac{P}{L_j}} h_{i,j}^k \right]_{(j,k) \in A}$ for any $\mathcal{A} \subseteq \mathcal{K}$, the rate $R_i(\mathcal{D}, \mathcal{V})$ is given by

$$R_i(\mathcal{D}, \mathcal{V}) = \log_2 \left( 1 + \frac{\|h_{i,D}\|^2}{\sigma^2 + \|h_{i,V}\|^2} \right). \quad (4.5)$$

The normalizing factor $\sqrt{\frac{P}{L_j}}$ is to reflect equal power cross the layers of each transmitter. Let $Q_i = \{ Q_{i1}, \ldots, Q_{ip_i}, Q_{ip_i+1} \}$ be a valid partition of $\mathcal{K}$. Based on this partition all layers $x_{i,k}$ are decodable by the $i^{th}$ receiver if

$$\forall \ m \in \{1, \ldots, p_i\} : \ R_{Q_m} \in C_i(Q_{m1}, \cup_{j=m+1}^{p_i} Q_{j}). \quad (4.6)$$

For the $i^{th}$ receiver and the valid partition $Q_i$, after setting $r_{Q_m}^i = \left[ r_{j,k}^i \right]_{(j,k) \in Q_m}$, we define

$$\theta_i(Q_i) \triangleq \begin{cases} \max x, & \text{s.t. } r_{j,k}^i = x, \forall (j,k) \in \mathcal{K}, \\ \tilde{R}_{Q_m} = R_{Q_m} + r_{Q_m}^i \in C_i(Q_{m1}, \cup_{j=m+1}^{p_i} Q_{j}), & \forall m \in \{1, \ldots, p_i\}, \end{cases} \quad (4.7)$$

which suggests equal (fair) rate increment for all the layers included in $\{ Q_{i1}, \ldots, Q_{ip_i} \}$ such that they are all decodable at the $i^{th}$ receiver. There exists a delicate difference between (4.3) and (4.7). (4.3) characterizes the network-wide fair sum-rate optimization problem, whereas (4.7) solves the same problem selfishly for each receiver, in the sense that each receiver solves this problem with the purpose of maximizing the fair sum-rate with decodability guarantees only for itself. So each receiver solves problem (4.7) locally and selfishly. Once all the receivers have solved (4.7), they exchange some information, through which they collaboratively solve (4.3). By defining $\tilde{Q}_i$ as the ensemble of all valid partitions $Q_i$, the maximum layer rate increment with the fairness constraint from the viewpoint of the $i^{th}$ user is given by

$$\theta_i^* \triangleq \max_{Q_i \in \tilde{Q}_i} \theta_i(Q_i). \quad (4.8)$$
Clearly, solving (4.8) through the naive exhaustive search over all possible choices of \(Q^i\) has a prohibitive complexity. By defining
\[
\Delta_i(D, V) \triangleq R_i(D, V) - \sum_{(j,k) \in D} R_{j,k}, \tag{4.9}
\]
for any disjoint arbitrary \(D, V \subseteq \mathcal{K}\) we offer Algorithm 4.1 that yields two outputs for each receiver \(i\) and avoids exhaustive search for obtaining the optimal valid partition \(Q^i\) that solves (4.8) and also provides the corresponding optimal rate increment.

For brevity and also for focusing the attention on the more practical issues pertinent to the implementation of these decoders, we describe the steps involved in Algorithm 4.1 and provide their properties and skip the proofs. The proofs hold essentially along the same line of arguments as those provided in [40] and [42], albeit with significant differences. In contrast to the rate allocation schemes in [40] and [42] that allocate one codebook to each transmitter, we allocate multiple codebooks to each transmitter, where the number of the codebooks is determined by the strength of the channels corresponding to that transmitter. Moreover, in [40] and [42] there is no constraint on the size of the number of transmitters than can be jointly decoded, which gives rise to a prohibitive complexity due to ML decoding. In contrast, here we constrain the number of layers to be decoded jointly by \(\mu\) which is determined based on the computational complexity level that the receivers can afford.

It is noteworthy that the structure of Algorithm 4.1 rules finding the ordered partitions \(Q^i = \{Q^i_1, \ldots, Q^i_{p_i}, Q^i_{p_i+1}\}\), in a reverse order, i.e., it first identifies \(Q^i_{p_i+1}\) and \(Q^i_1\) is the last to be identified. \(Q^i_{p_i+1}\) contains the indices of the layers that will not be decoded in any stage of the successive decoding procedure of the CPGD algorithm. \(Q^i_1\) is the set of the indices of the layers that will tolerate the least amount of increase in their rates (subject to the fairness constraint). Subsequently, after decoding the layers in \(Q^i_1\), among the remaining layers in \(Q^i_2\) become the set of the indices of the layers that will tolerate the least amount of increase in their rates and the same property holds for all other sets \(Q^i_3, \ldots, Q^i_{p_i}\).

We remark that the complexity of Algorithm 4.1 is determined by those of the optimization problems in lines 3 and 4. As the size \(\mu\) is controlled to be small, even an exhaustive search for finding at most \(\mu\) layers among the (at most) \(L\) existing layers will suffice to obtain a computationally efficient (of the order at most \(L^\mu\)) way of solving these two problems. This is contrary to the
approaches in [40] and [42] that impose no constraint on decoding size, for which the complexity of the exhaustive search grows exponentially with |K| and solving them necessitates resorting to the submodular optimization tools. Also the algorithm has at most \( L = \sum_{j=1}^{K} L_j \) iterations, which consequently induced another level of computations with linear complexity in \( L \).

Algorithm 4.1 - The Optimal Valid Partition for the Receiver \( i \)

1: Initialize \( D = K, G = \emptyset, p_i = 0 \), and \( \ell = 1 \)
2: repeat
3: Find \( \delta_\ell = \min_{\forall \neq \emptyset, \forall \subseteq D, |V| \leq \mu} \frac{\Delta_i(V,G)}{|V|} \)
4: Find \( G_i^\ell = \arg \min_{\forall \neq \emptyset, \forall \subseteq D, |V| \leq \mu} \frac{\Delta_i(V,G)}{|V|} \)
5: \( D \leftarrow D \setminus G_i^\ell \) and \( G \leftarrow G \cup G_i^\ell \)
6: if \( (i,k) \in D \) for all \( k \in \{1, \ldots, L_i\} \)
7: \( \tilde{r}_{i,j,k} = +\infty \) for all \( (j,k) \in G_i^\ell \)
8: else
9: \( \tilde{r}_{i,j,k} = \delta_\ell \) for all \( (j,k) \in G_i^\ell \), \( p_i \leftarrow p_i + 1 \)
10: end if
11: \( \ell \leftarrow \ell + 1 \)
12: until \( D = \emptyset \)
13: Set \( Q_i^m \leftarrow G_i^{\ell-m} \) for \( 1 \leq m \leq p_i \), and \( Q_{p_i+1}^i \leftarrow \cup_{m>p_i} G_i^{\ell-m} \)
14: Output the rates \( \{\tilde{r}_{i,j,k}\} \) for \( (j,k) \in K \) and the partitions \( Q^i = \{Q_1^i, \ldots, Q_{p_i}^i, Q_{p_i+1}^i\} \).

Algorithm 4.2 - Fair Rate Allocation

1: Input \( R = [R_{j,k}]_{(j,k) \in K} \)
2: for \( i = 1, \ldots, K \) do
3: Run Algorithm 4.1 to determine \( \{\tilde{r}_{j,k}\} \) for \( (j,k) \in K \) and \( Q^i \)
4: end for
5: Obtain \( \theta^* = \min_i \theta_i^* \) where \( \theta_i^* = \min_{(j,k) \in K} \tilde{r}_{j,k}^i. \)
6: Update \( \tilde{R}_{j,k} \leftarrow R_{j,k} + \theta^* \) for \( (j,k) \in K \)
7: Output \( \tilde{R} = [\tilde{R}_{j,k}]_{(j,k) \in K} \) and the partitions \( \{Q^1, \ldots, Q^K\} \).

The main property of Algorithm 4.1 is formalized in the following theorem. The intuition behind Algorithm 4.1 and Theorem 4.1 is that it each receiver aims to identify the best set of interfering codebooks to decode that yields the highest sum-rate throughput. By focusing on each possible set of codebooks, these codebooks constitute a multiple access channel to the receiver. Therefore, the best set of codebooks is the one whose corresponding multiple access channel sustains a higher level
of rate increment (or lower level of rate decrement). A naive, and computationally complex approach is the exhaustive search. Alternatively, each receiver employ a successive decoding approach such that successively identifies and discards the codebooks that removing them allows for further increasing the sum-rate. Such successive identification of the codebooks resumes until discarding a codebook penalizes the sum-rate.

**Theorem 4.1:** Algorithm 4.1 identifies the partition that maximizes $\theta_i(Q^i)$ over all possible valid partitions $\tilde{Q}^i$ and obtains $\theta^*_i$, i.e.,

$$\{Q^i_1, \ldots, Q^i_{p_i+1}\} = \arg \max_{Q^i \in \tilde{Q}^i} \theta_i(Q^i) \quad \text{and} \quad \theta^*_i = \max_{Q^i \in \tilde{Q}^i} \theta_i(Q^i) = \min_{(j,k) \in K} r^i_{j,k}.$$

Through Algorithm 4.1 each receiver identifies its best partition in a distributed way by performing some local processing without any information exchange among different users. Such local processing can interestingly lead to solving the fair sum-rate optimization problem cast in (4.3) with limited information exchange. For this purpose, the $i^{th}$ receiver reports the scalar $\theta^*_i$, which is obtained in Algorithm 4.1 and is related to the maximum rate-increment possible for the $i^{th}$ receiver, to all other transmitters. The globally optimal rate increment is obtained by Algorithm 4.2, which is simply the smallest rate increment that all users suggest.

**Theorem 4.2:** The rate vector yielded by Algorithm 4.2 satisfies $\tilde{R}_K \succeq \hat{R}_K$, where $\hat{R}_K$ is any decodable rate-vector such that $\hat{R}_K = R_K + x \cdot 1_{1 \times L}$ for some $x \in \mathbb{R}$.

According to Theorem 4.1, by running Algorithm 4.1 each receiver specifies the best set of interferers to be decoded, which maximizes the sum-rate throughput. Given the different sum-rate throughputs that different users can sustain based on their local processing in Algorithm 4.1 and Theorem 4.1, the network-wide optimal sum-rate throughput is determined by the bottleneck user, which is the one that sustains the smallest sum-rate throughput.

### 4.2.3 Iterative Rate Allocation

Note that Algorithm 4.2 solves (4.3) by increasing the rate of each layer by an amount of $\theta^*$, which is the same for all layers. As stated in [39], this strict notion of fairness will penalize the system throughput in terms of the sum-rate. In this subsection, we propose Algorithm 4.3 to further increase the rates of some layers so that the sum-rate can be further increased.
Algorithm 4.3 - Iterative Rate Allocation

1: Input \( \mathbf{R} = [R_{j,k}]_{(j,k) \in \mathcal{K}} \)
2: repeat
3: for \( i = 1, \ldots, K \) do
4: Run Algorithm 4.1 to determine \( \{\tilde{r}_{j,k}^i\} \) for \( (j,k) \in \mathcal{K} \) and \( \mathcal{Q}^i \)
5: end for
6: Obtain \( \tilde{r}_{j,k} = \min_{i=1}^K \tilde{r}_{j,k}^i \) for \( (j,k) \in \mathcal{K} \).
7: Update \( \tilde{R}_{j,k} \leftarrow \tilde{R}_{j,k} + \tilde{r}_{j,k} \) for \( (j,k) \in \mathcal{K} \), \( \mathbf{R} \leftarrow \tilde{\mathbf{R}} \)
8: until \( \tilde{\mathbf{R}} \) converges
9: Output \( \tilde{\mathbf{R}} = [\tilde{R}_{j,k}]_{(j,k) \in \mathcal{K}} \) and the partitions \( \{\mathcal{Q}^1, \ldots, \mathcal{Q}^K\} \).

In contrast to Algorithm 4.2, Algorithm 4.3 iteratively makes a new rate increment recommendation for all layers based on the allocated rates obtained from the previous iteration, denoted as \( r_{j,k}^i \) for all receivers \( i \) in a distributive manner. The rate increment for the layer \( (j,k) \) is \( \tilde{r}_{j,k} = \min_i \tilde{r}_{j,k}^i \) which ensures that all layers are decodable after each iteration of rate increment recommendation.

The following Theorem 4.3 shows that for any layer the allocated rate vector from Algorithm 4.3 is larger than or equal to that from Algorithm 4.2.

**Theorem 4.3:** Denote the output rate vector yielded by Algorithms 4.2 and 4.3 as \( \tilde{\mathbf{R}}_{\mathcal{K},2} \) and \( \tilde{\mathbf{R}}_{\mathcal{K},3} \), respectively. We have \( \tilde{\mathbf{R}}_{\mathcal{K},3} \succeq \tilde{\mathbf{R}}_{\mathcal{K},2} \).

**Remark:** Algorithm 4.3 relaxes the rate constraint that the rate increments of all layers are not restricted to the minimal rate increment as recommended by Algorithm 4.2. Therefore, by running Algorithm 4.3 the rate increments of all layers are larger than or equal to those obtained from Algorithm 4.2. Based on Theorem 4.3, Algorithm 4.3 is employed for the design of practical coding scheme in the remainder of this chapter.

### 4.2.4 Numerical Results

Consider a single-antenna interference channel with \( K = 6 \) transceivers. For \( i, j \in \{1, \ldots, K\} \), the channel coefficients \( h_{i,j} \) are distributed as \( \mathcal{N}_C(0, 1) \). We run Algorithms 4.2 and 4.3 to assess the performance of the proposed coding scheme for Gaussian modulation and infinite-length random codes. The initial rate \( \mathbf{R}_\mathcal{K} \) is selected to be the rate vector achievable by using single-user decoders that treat all interference as noise. Fig. 4.2 shows the sum-rate obtained from Algorithms 4.2 and 4.3 for the un-layered coding scheme with the group size \( \mu = 1, 2, \) and 3. It is seen that the
The sum-rate obtained from Algorithm 4.3 is larger than that from Algorithm 4.2. In the remainder of this chapter, Algorithm 4.3 is employed to obtain all numerical and simulation results.

Fig. 4.3 shows the performance of the layered coding scheme with $\mu = 1$ for different number of layers $L_j = 2, 5, 10,$ and $15$ for all users. It is seen that the coding schemes with $L_j = 5$ significantly outperform those with $L_j < 5$, and further increasing $L_j$ brings little performance gain over $L_j = 5$.

Furthermore, we compare the sum-rate of users for different group sizes $\mu = 1, 2,$ and $3$, as illustrated in Fig. 4.4. For the layered coding scheme, we set $L_j = 5$ layers for all users. At each channel SNR value $P/\sigma^2$, one hundred channel realizations are simulated and the average sum-rate is plotted against the channel SNR. Also plotted is the performance of the single-user decoding, denoted by “MMSE decoding” (where “MMSE” stands for minimum mean square error). It is seen that CPGD schemes exhibit significant performance gains over the single-user decoding.

We have the following important observation from Fig. 4.4. Without layering, an increase in the group size $\mu$ leads to an increase in the sum-rate. Moreover, for a fixed group size, the layered coding can indeed provide higher sum-rate than the un-layered coding. More interestingly, it is seen that with layering, when the group size $\mu = 1$, there is a substantial gain in sum-rate compared with the un-layered coding scheme, and further increasing $\mu$ shows little performance gain over $\mu = 1$. Hence, the layered coding scheme with the group size $\mu = 1$ achieves a good tradeoff between the performance and complexity. It will be seen in the next section that this is also the case when practical modulations and channel codes are employed.

Remark: For $\mu = 1$, the group selection in Algorithm 4.1 can be simplified. In lines 3 and 4, given the selected layers $\mathcal{G}$, we find the optimal layer $(j^*, k^*)$ as follows,

\[
(j^*, k^*) = \arg \min_{(j,k) \in \mathcal{D}} \Delta_i((j,k), \mathcal{G}) = \arg \min_{(j,k) \in \mathcal{D}} \left( R_i((j,k), \mathcal{G}) - R_{j,k} \right),
\]

(4.10)

where the rate $R_i((j,k), \mathcal{G})$ is given by

\[
R_i((j,k), \mathcal{G}) = \log_2 \left( 1 + \frac{P}{L_j} \cdot \frac{\|h_{i,j}\|^2}{\sigma^2 + \|h_{i,G}\|^2} \right),
\]

(4.11)

and the rate increment is given by

\[
\delta^*_\ell = \arg \min_{(j,k) \in \mathcal{D}} \Delta_i((j,k), \mathcal{G}).
\]

(4.12)

Thus both steps involve linear search over the set $\mathcal{D}$, whose cardinality, in the worst case, is equal to the total number of layers $L$. The same will hold true when practical modulations and channel codes are employed.
Figure 4.2: The sum-rate given by Algorithms 4.2 and 4.3 assuming Gaussian signaling and infinite-length random codes.
Figure 4.3: The sum-rate given by Algorithm 4.3 assuming Gaussian signaling and infinite-length random codes for different number of layers.
Figure 4.4: The sum-rate given by Algorithm 4.3 assuming Gaussian signaling and infinite-length random codes.
4.3 Practical Transmission Scheme

The rate increment solutions obtained through the CPGD procedure in Algorithms 4.1 and 4.3 are not necessarily viable in practice. There are two major challenges involved. First in characterizing the rate regions in (4.4) and the achievable rates in (4.5) it is implicitly assumed that the codewords are drawn from Gaussian codebooks and we have ideal infinite-length random codes for achieving these rates. In practice, however, there is always a gap between the ideal Gaussian rates and the rates achieved by practical modulation and coding schemes, that we aim to minimize. More specifically, for practical rate allocation the rate $R_i(D, V)$ is given by

$$R_i(D, V) = \log_2 \left( 1 + \frac{\gamma \|h_i, D\|^2}{\sigma^2 + \|h_i, V\|^2} \right),$$

where $\gamma$ denotes the gap between the Gaussian modulation with infinite-length random codes and the practical QAM with finite-length codes. The other challenge is that, in contrast to the output rates of Algorithm 4.3 which can be set arbitrarily such that the system needs to perform online channel code construction which incurs unacceptable computational complexity, in practical transmission the channel codes are constructed offline and only a finite number of channel codes can be stored and selected according to current channel realizations.

4.3.1 Rate Selection Procedure

To address these challenges, we carry out a three-step procedure that selects a transmission rate for each user and its layers. In the first step, we examine whether all users can operate at rates well above zero that can be implemented in practice. In case there exist users that can operate only at very low rates we inactivate them. Then we determine how many layers each active transmitter should have. Subsequently, based on the number of layers obtained for each transmitter, by running Algorithm 4.3 we determine the rates for the layers of each transmitter, and inactivate the layers which operate at very low rates. In the second step, in order to find some implementable rates close to those yielded by Algorithm 4.3 we quantize the rates according to a quantization codebook that satisfies some optimality measure. These quantized rates are subsequently considered as coarse approximations for possible practical rates. As the quantized rates must be decodable, each quantized rate is smaller than its original counterpart which incurs some loss in spectral efficiency. In the final step, we compensate for such loss, through devising a fine tuning scheme for increasing
the rates beyond their quantized values and make them as close as possible to the original rates yielded by Algorithm 4.3 while being also practical.

We employ a table of spectrum rate $T = \{d_1, ..., d_T\}$ for the user and layer inactivation in the first step and the rate quantization in the second step. Each element $d_j \in T$ is associated with a pre-designed channel code with $N d_j$ information bits and a pre-designed modulation scheme which maps the coded bits to channel symbols. In the user (layer) inactivation, the users (layers) with allocated rates smaller than $d_1$ will be inactivated. If the rate of a layer is quantized to the rate $d_j$ for some $1 \leq j \leq T$, then we employ a codebook with the number of information bits $N d_j$ and the corresponding modulation scheme for that layer. To achieve the rate enhancement in the final step, we choose the number of transmitted symbols smaller than $N$ while keeping the number of information bits $N d_j$ unchanged. The rate enhancement could also be done by increasing the number of information bits for each layer while keeping the number of transmitted symbols unchanged. However, for a practical capacity-approaching codebook, changing the number of coded bits flexibly without regenerating the codebook can be achieved using rateless codes, while there is no practical capacity-approaching codebook which allows changing the number of information bits flexibly.

4.3.1.1 User Inactivation and Message Layering

For initializing the rate allocation procedure we start off with assigning one codebook to each transmitter, i.e., $\forall j$ we have $L_j = 1$. We run Algorithm 4.3 and inactivate user $j$ for which $\hat{R}_j < d_1$. Let us denote the set of users retained as active users by $\mathcal{H} \subseteq \{1, \ldots, K\}$. By inactivating some users, the rest are exposed to less interference and can possibly sustain higher rates. For this reason, after inactivating the users with rates smaller than $d_1$ we run Algorithm 4.3 again to obtain the new set of rate increment suggestions for all active users, denoted as $\{\tilde{R}_j^{(1)}\}_{j \in \mathcal{H}}$.

Now based on the rates $\{\tilde{R}_j^{(1)}\}_{j \in \mathcal{H}}$ we decide how many codebooks should be assigned to each transmitter as follows,

$$\forall j \in \mathcal{H} : \quad L_j = \max \left\{ 1, \left\lfloor \frac{\tilde{R}_j^{(1)}}{\Delta R} + 0.5 \right\rfloor \right\},$$

where $\Delta R$ is the unit rate to be sustained by each codebook. Given the number of layers, we can now assign multiple codebooks to each active transmitter and implement the CPGD in order to find the new rate for each layer by running Algorithm 4.3. Let us denote the rate of the $k^{th}$ layer
of the active transmitter $j \in \mathcal{H}$ by $\tilde{R}_{j,k}^{(2)}$. It is possible that the rates of some layers are not large enough to be implementable. We then inactivate the layers with rates smaller than $d_1$, and update the number of layers $L_j$ by the number of layers with rates larger than $d_1$, i.e.,

$$L_j \leftarrow \sum_{k=1}^{L_j} \mathbb{1}_{\{\tilde{R}_{j,k}^{(2)} \geq d_1\}}, \quad 1 \leq j \leq K.$$  \hspace{1cm} (4.15)

Based on the updated number of layers $L_j$, we run Algorithm 4.3 again to compute the rates $\tilde{R}_{j,k}^{(3)}$ for all layers. We define

$$R_{\text{sum}}^{(3)} = \sum_{j=1}^{K} \sum_{k=1}^{L_j} \tilde{R}_{j,k}^{(3)} \cdot \mathbb{1}_{\{\tilde{R}_{j,k}^{(3)} \geq d_1\}},$$  \hspace{1cm} (4.16)

We compare $R_{\text{sum}}^{(3)}$ with $R_{\text{sum}}^{(1)} = \sum_{j \in \mathcal{H}} \tilde{R}_{j,k}^{(1)}$. If the layered scheme provides larger rate, i.e., $R_{\text{sum}}^{(3)} > R_{\text{sum}}^{(1)}$, we employ the layered scheme; otherwise we employ the un-layered scheme. Hence we have the following sum rate

$$\tilde{R}_{\text{sum}}^{(3)} = \max \left\{ R_{\text{sum}}^{(3)}, R_{\text{sum}}^{(1)} \right\}.$$  \hspace{1cm} (4.17)

### 4.3.1.2 Rate Quantization and Modulation Selection (Coarse Tuning)

Assume that we get the rate $\hat{R}_{j,k}$ for the layer $(j, k)$ after the message layering and the rate allocation for each layer. Now we map the rate $\hat{R}_{j,k}$ to a pre-designed channel node and modulation scheme associated with some $\hat{R}_{j,k} \in T = \{d_1, \ldots, d_T\}$ as follows,

$$\hat{R}_{j,k} = \max_{d_q \in T, d_q \leq \hat{R}_{j,k}} d_q,$$  \hspace{1cm} (4.18)

and employ the channel code with $N\hat{R}_{j,k}$ information bits and the corresponding modulation scheme which maps $m_{j,k}$ coded bits to one symbol. To encode the layer $(j, k)$, we encode the $N\hat{R}_{j,k}$ information bits to $N\hat{m}_{j,k}$ coded bits, and map those coded bits to $N$ channel symbols.

The modulation scheme is selected from 4-, 16-, and 64-QAMs according to the rules in Table 4.1. Let us explain the rules in Table 4.1. Note that for the 4-, 16- and 64-QAMs, the capacity of bit-interleaved coded modulation with Gray mapping well approximates that of the coded modulation, and the capacity of coded modulation well approximates that of Gaussian modulation when the spectral efficiency is below 1, 2, and 3 bits per channel use, respectively [43]. Therefore in Table 4.1, when $\mu = 1$, for $d_j \leq 1.0$ we associate it with 4-QAM, for $1.0 \leq d_j \leq 2.0$ we associate it with 16-QAM, and for $d_j \geq 2.0$ we associate it with 64-QAM. For controlling the complexity of the ML
decoders, we constrain the number of layers to be decoded jointly in each iteration of the successive decoder of Algorithm 4.1 to be \( \mu \leq 3 \). For controlling the demodulation complexity when \( \mu = 3 \), i.e., three users can be jointly decoded, all users can employ only 4-QAM. For the same reason, for \( \mu = 2 \), all users only employ 4- and 16-QAMs, and the joint decoding of two users both employing 16-QAM is not allowed, since it incurs large complexity to the 2-user a posteriori probability (APP) detector.

<table>
<thead>
<tr>
<th>Group Size</th>
<th>( d_j \leq 1 )</th>
<th>( 1 \leq d_j \leq 2 )</th>
<th>( 2 &lt; d_j )</th>
<th>Symbol Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4-QAM</td>
<td>16-QAM</td>
<td>64-QAM</td>
<td>Gray mapping</td>
</tr>
<tr>
<td>2</td>
<td>4-QAM</td>
<td>16-QAM</td>
<td>16-QAM</td>
<td>Natural mapping</td>
</tr>
<tr>
<td>3</td>
<td>4-QAM</td>
<td>4-QAM</td>
<td>4-QAM</td>
<td>Natural mapping</td>
</tr>
</tbody>
</table>

Table 4.1: Modulation schemes for different rate values.

For \( \mu = 2 \), we provide a post-processing procedure (Algorithm 4.4) to avoid the joint decoding of two users (layers) both employing 16-QAM. For a decoding group \( \mathcal{Q}_i \) with \( |\mathcal{Q}_i| = 2 \), if both layers employs 16-QAM, we change the rate of smaller one to 4-QAM. The post-processing procedure is detailed in Algorithm 4.4.

**Algorithm 4.4 - Rate Post-processing for \( \mu = 2 \)**

1. for \( i \in \mathcal{H} \) and \( \mathcal{Q} \in \{ \mathcal{Q}_i^\ell \}_{1 \leq \ell \leq p_i} \);  
2. if \( |\mathcal{Q}| = \{(j_1, k_1), (j_2, k_2)\} = 2 \) and \( \hat{R}_{j_1, k_1} > 1.0, \hat{R}_{j_2, k_2} > 1.0 \)  
3. if \( \hat{R}_{j_1, k_1} < \hat{R}_{j_2, k_2} \)  
4. Let \( \hat{R}_{j_1, k_1} \leftarrow \max_{d_q \in T, d_q \leq 1.0} d_q \)  
5. else  
6. Let \( \hat{R}_{j_2, k_2} \leftarrow \max_{d_q \in T, d_q \leq 1.0} d_q \)  
7. end if  
8. end if  
9. end for

### 4.3.1.3 Rate Enhancement (Fine Tuning)

Recall that after rate quantization, there is a reduction in the rates of layers. Such reduction incurs a loss in terms of the spectral efficiency of the network. To compensate for such loss, we boost each rate by a factor \( \eta \), scaling up the quantized rate \( \hat{R}_{j,k} \) to \( \eta \cdot \hat{R}_{j,k} \). We need to determine the largest
possible $\eta$ denoted as $\eta^*$, such that the new scaled rates do not violate the Shannon limits. For each subset $\mathcal{U} \subseteq Q^*_m$ let us define the sum-rates

$$R^\text{sum}_{i,m}(\mathcal{U}) = \log_2 \left( 1 + \gamma \frac{\|h_{i,\mathcal{U}}\|^2}{\sigma^2 + \|h_{i,\mathcal{V}}\|^2} \right)$$

and $\hat{R}^\text{sum}_{i,m}(\mathcal{U}) = \sum_{(j,k) \in \mathcal{U}} \hat{R}_{j,k}$, \hspace{1cm} (4.19)

where $\mathcal{V} = \bigcup_{j=m+1}^{p_i} Q^i_j$. Given the subset $\mathcal{U}$ of $Q^*_m$, $R^\text{sum}_{i,m}(\mathcal{U})$ is Shannon limit of the sum-rate of the layers in $\mathcal{U}$, where the parameter $\gamma$ is incorporated to account for the loss due to practical constellation and coding schemes, and $\hat{R}^\text{sum}_{i,m}(\mathcal{U})$ is the sum of the quantized rates of the same layers obtained in Section 4.3.1.2. For $Q^*_m$, the rates $\hat{R}_{j,k}$ can be increased to $\eta_{i,m} \cdot \hat{R}_{j,k}$ for $(j,k) \in Q^*_m$ where $\eta_{i,m}$ is given by

$$\eta_{i,m} = \min_{\mathcal{U} \subseteq Q^*_m} \frac{R^\text{sum}_{i,m}(\mathcal{U})}{\hat{R}^\text{sum}_{i,m}(\mathcal{U})}. \hspace{1cm} (4.20)$$

Based on these definitions, the quantized rates $\{\hat{R}_{i,k}\}$ can be increased as much as the scaled sum-rate $\eta^* \cdot \hat{R}^\text{sum}_{i,m}$ does not exceed its limit $R^\text{sum}_{i,m}$ for all $i,m$, where

$$\eta^* = \min_{i,m} \eta_{i,m} = \min_{i,m} \min_{\mathcal{U} \subseteq Q^*_m} \frac{R^\text{sum}_{i,m}(\mathcal{U})}{\hat{R}^\text{sum}_{i,m}(\mathcal{U})}. \hspace{1cm} (4.21)$$

After the above three steps, the transmission data is assembled as follows. For the layer $(j,k)$, we choose the pre-designed channel code with the number of information bits $N \hat{R}_{j,k}$, encode those information bits to $N m_{j,k}$ coded bits using the rateless codes to be optimized in Section 4.4 and map these coded bits to $N$ symbols. The rate enhancement is implemented via transmitting

$$N_s = \Delta N \cdot \left\lfloor \frac{N}{\eta^* \Delta N} \right\rfloor \hspace{1cm} (4.22)$$

symbols, where $\Delta N$ is the number of symbols in each transmission burst.

**Remark:** In real transmission, the role of fine tuning is to set up a starting point after which the receiver is allowed to perform decoding and sending feedback. Doing so significantly reduces the feedback load compared to the transmission scheme in which the receivers begin decoding and sending feedback from the beginning of the transmission after each transmission burst of $\Delta N$ symbols.

### 4.3.1.4 An Illustrative Example of the Proposed Coding Scheme

In Fig. 4.5 we show an illustrative example for the procedure of the proposed coding scheme in an interference channel with four users. Let the spectrum rate table $T = \{d_1, d_2, d_3, d_4\}$. After the
user inactivation and message layering based on the threshold rate $d_1$, user 2 is inactivated and users 1 and 3 are divided into 2 layers. In the rate quantization and modulation selection, the rates of the five layers $(1, 1)$, $(1, 2)$, $(3, 1)$, $(3, 2)$, and $(4, 1)$ are quantized to be $d_1$, $d_2$, $d_3$, $d_1$, and $d_4$, respectively, i.e., the number of information bits assigned to these layers are $Nd_1$, $Nd_2$, $Nd_3$, $Nd_1$, and $Nd_4$, respectively, where $N$ is the nominal number of transmitted symbols. In the rate enhancement, the number of transmitted symbols is decreased to $\eta^*N$ for some $\eta^* < 1$, and thus the rate of the five layers are enhanced to $d_1/\eta^*$, $d_2/\eta^*$, $d_3/\eta^*$, $d_1/\eta^*$, and $d_4/\eta^*$, respectively.

The optimization of the table can be done in two steps. In the first step, we optimize $d_1$ based on the user inactivation and message layering, to maximize the sum rate of all layers; and in the second step, we optimize the remaining rates $d_2$, $d_3$, and $d_4$ based on optimal scalar quantization, to minimize the rate loss due to the rate quantization. The design of the rate table $T$ is addressed in the next subsection.

### 4.3.2 Parameter Design

In this section, we design the rate table $T$. In contrast to the above rate selection procedure which is performed online according to each channel realization, designing the rate table $T$ is done offline for the ensemble of channel realizations considered. We collect enough samples of channel realizations as a large training set, and design the rate table $T$ according to the training set. Assume that all elements $d_q$ in $T$ belongs to some candidate set $V = \{v_1, \ldots, v_S\}$, and we set $v_0 = 0$ and $v_{S+1} = +\infty$.

As we can see, given the channel realization, the sum-rate $\tilde{R}_{\text{sum}}^{(3)}$ in (4.17) is a function of the threshold rate $d_1$, denoted as $\tilde{R}_{\text{sum}}^{(3)}(d_1)$. Let

$$
\tilde{R}_{\text{sum}}^{(3)}(d_1) \triangleq \mathbb{E} \left[ \tilde{R}_{\text{sum}}^{(3)}(d_1) \right],
$$

where the expectation is over the ensemble of channel realizations, which is computed as the averaged $\tilde{R}_{\text{sum}}^{(3)}(d_1)$ based on all channel realizations in the training set. We select $d_1$ as follows,

$$
d_1 = \begin{cases} 
\max_{u \in V} u, \\
\text{s.t. } \tilde{R}_{\text{sum}}^{(3)}(u) \geq \max_{v \in V} \tilde{R}_{\text{sum}}^{(3)}(v) - \epsilon
\end{cases}
$$

where $\epsilon$ is to avoid the extreme low rate of $d_1$ when $\arg \max_v \tilde{R}_{\text{sum}}^{(3)}(v)$ is very small.
In the next step we find the optimal quantized rates $d_2, d_3, ..., d_T$ to minimize the rate loss due to the quantization operation (4.18). Based on the optimal $d_1$ from (4.24), we run the user inactivation and message layering for all the channel realizations in the training data set, and get the empirical distribution of the rate $\tilde{R}_{j,k}$ (unquantized version) for each layer $(j, k)$. Let $p_q$ be the fraction of rates $\tilde{R}_{j,k}$ that fall into the interval $[v_{q-1}, v_q)$ for $q \in \{1, \ldots, S + 1\}$. For any two consecutive quantization levels $d_q$ and $d_{q+1}$ we define $D(d_q, d_{q+1})$ as the rate distortion due to quantizing the rates that lie in the interval $[d_q, d_{q+1})$. As $d_q$ and $d_{q+1}$ are to be selected from $V$, then $\exists \ m < n \in \{1, \ldots, S\}$ such that $d_q = v_m$ and $d_{q+1} = v_n$. Therefore, the distortion of interest is the weighted sum of the distortions over the intervals $[v_m, v_{m+1}), \ldots, [v_{n-1}, v_n)$, i.e.,

$$D(d_q, d_{q+1}) = \sum_{i: \ d_q < v_i < d_{q+1}} (v_i - d_q)p_i.$$
Then given $d_1$ obtained from (4.24), we design the rest of $T$ by solving the following optimization problem,

$$\begin{align*}
\max_{\{d_2, \ldots, d_T\} \subset V} & \quad \sum_{q=1}^{T} D(d_q, d_{q+1}) \\
\text{s.t.} & \quad d_1 < d_2 < \cdots < d_T < d_{T+1} = +\infty.
\end{align*}$$

(4.25)

This problem can be solved using a dynamic programming approach as the distortion function can be solved recursively in $T$ stages where the optimal solution of each stage does not affect the optimal solution of later stages.

4.3.3 Numerical Results

We consider the same scenario as that in Section 4.2.4. Assume that the size of the rate quantization table is $|T| = 4$. From the code profile optimization results, we have $\gamma = 0.63$ for $\mu = 1$, $\gamma = 0.45$ for $\mu = 2$. We set $\gamma = 0.45$ for $\mu = 3$, which is an upper bound for the error free decoding, since we need to ensure the error-free decoding of any two users among the three.

We choose the maximum group size $\mu$ based on the sum-rate $\bar{R}_\text{sum}^{(3)}$ according to (4.23). Let the set of candidates $V = \{0.001, 0.002, 0.003, \ldots, 4.000\}$. In Fig. 4.6 we plot the sum-rates $\bar{R}_\text{sum}^{(3)}(d_1)$ for $\mu = 1, 2$ and 3 and the cutoff rate $d_1$ from 0.0 to 0.9 with and without layering. The average sum-rates $\bar{R}_\text{sum}(d_1)$ are computed for the SNR from 0dB to 9dB each with 1000 random channel realizations. It can be seen that at the optimum cutoff rate which maximizes $\bar{R}_\text{sum}^{(3)}(d_1)$, the coding schemes for $\mu = 1$ outperform those for $\mu = 2$ and 3.

Next we optimize the rate quantization table $T$ according to (4.25) in Section 4.3.2. The optimum cutoff rates $d_1$ are 0.45 and 0.60 for the layered and un-layered coding schemes with $\mu = 1$. We have that for $\mu = 1$, $T = \{0.45, 1.08, 1.87, 2.80\}$ for the layered coding scheme and $T = \{0.60, 1.28, 2.02, 2.88\}$ for the un-layered coding scheme. We also optimize the rate quantization table $T$ for $\mu = 2$ and $\mu = 3$ with and without layering. Fig. 4.7 shows the sum quantized rates with and without the fine turning for different coding schemes with group size $\mu = 1$ and 2. It is seen that the layered coding scheme significantly outperforms the un-layered scheme. Moreover, the layered scheme with $\mu = 1$ performs the best among all the schemes. This is because for $\mu = 1$ the performance gap ($\gamma = 0.63$) is significantly smaller than that for $\mu = 2$ ($\gamma = 0.45$). It is also seen that the fine tuning offers a rate enhancement about 20%. For the coding schemes with $\mu = 3$, with and without layering, we observe that they perform worse than $\mu = 1$, and the fine tuning
Figure 4.6: The averaged sum-rate $\bar{R}_{\text{sum}}^{(3)}(d_1)$ for different cutoff rate $d_1$.

offers significant rate enhancement.
4.4 Rateless Code Design

4.4.1 Transmission Using Raptor Codes

We employ the Raptor code, which is a concatenated code with an inner LT code and an outer linear code, to implement the rateless codebook. In particular, we use the doped Raptor code with a rate-0.95 IRA precode. The reason we employ the Raptor code is as follows. Firstly, it exhibits near-capacity performance for both the single-user channel and the two-user multiple-access channel. Secondly, the flexible online fine tuning can be easily implemented by fixing the precode and adjusting the number of LT coded bits.

For layer $(j, k)$, we first encode the $N\hat{R}_{j,k}$ information bits using a rate-0.95 IRA precode, and then perform the LT encoding to generate $m_{j,k}N_s$ coded bits ($N_s$ symbols). It may happen that the decoding of some layer based on the $N_s$ received symbols fails. In this case, we perform an
incremental transmission where each time each layer transmits additional $\Delta N$ parity symbols, until successful decoding. In most cases the decoding succeeds after transmitting $N_s$ symbols; and even if the incremental transmission happens, usually only one additional burst of $\Delta N$ symbols suffices.

In the remainder of this section we design the code profiles to increase the parameter $\gamma$ in (4.19) and reduce the loss in spectral efficiency due to the practical code design. We consider the code profile design for $\mu = 1$ and $\mu = 2$.

### 4.4.2 Code Design for $\mu = 1$

Since each time one user is decoded, we only need to consider the code profile optimization for the single-user AWGN channel.

#### 4.4.2.1 Code Profile Optimization

When $|Q^i_m| = 1$, only one layer is decoded by the $i^{th}$ receiver. Therefore, the employed profile should perform well for the single-user channel over a wide range of code rates. We find a profile that minimizes the maximum gap between the profile threshold and the capacity yielded by Gaussian signaling among the rates $R_0 = \{0.2, 0.3, 0.4, 0.5\}$. For each code rate $r \in R_0$, the threshold SNR for Gaussian signaling is given by

$$J^{(1)}_G (r) = 2r - 1.$$  \hfill (4.26)

We define $J^{(1)}_P (r, \{\lambda_j\})$ as the SNR threshold for the LT output node profile $\{\lambda_j\}$ (node perspective) for BPSK modulation, which is computed using the extrinsic information transfer (EXIT) function [13]. We optimize the profile via finding

$$\{\lambda^*_j\} = \arg\min_{\{\lambda_j\}} \max_{r \in R_0} \frac{J^{(1)}_P (r, \{\lambda_j\})}{J^{(1)}_G (r)},$$  \hfill (4.27)

using differential evolution (DE) method [14]. Although in our system 16- and 64-QAMs are also employed, since Gray mapping is employed, the optimized code profile for BPSK (QPSK) is also a good profile for the 16- and 64-QAMs. Let $R_0^{M_1}$ be a subset of the combination of the code rate and modulation schemes $R_0 \times \{4$-QAM, 16-QAM, 64-QAM$\}$ where each elements $(r, m) \in R_0^{M_1}$ satisfy the constraint of the spectrum efficiency and modulation schemes for $\mu = 1$ given in Table
CHAPTER 4. CONSTRAINED PARTIAL GROUP DECODER FOR K-USER INTERFERENCE CHANNELS

I. For the employed codes with the optimized profile \(\{\lambda_j^*\}\), we compute

\[
\gamma = \max_{(r,m) \in \mathcal{R}_0^{M_1}} \frac{\mathcal{J}_S^{(1)}(r,m,\{\lambda_j^*\})}{\mathcal{J}_G^{(1)}(r)},
\]

where \(\mathcal{J}_S^{(1)}(r,m,\{\lambda_j^*\})\) is the error-free decoding SNR for the profile \(\{\lambda_j^*\}\) from simulations for \((r,m) \in \mathcal{R}_0^{M_1}\). For each \(\mathcal{J}_S^{(1)}(r,m,\{\lambda_j^*\})\), one hundred channel realizations are simulated and the decoding is error-free if there is no decoding error.

4.4.2.2 EXIT Function

The threshold \(\mathcal{J}_P^{(1)}(r,\{\lambda_j\})\) is computed using EXIT functions as follows. Let \(\{\rho_j\}\) be the LT input node profile also from the node perspective, which can be derived from the profile \(\{\lambda_j\}\). Let \(\{\tilde{\lambda}_j\}\) and \(\{\tilde{\rho}_j\}\) be the corresponding profiles from the edge perspective. Let \(\{\gamma_i\}\) and \(\{\tilde{\gamma}_i\}\) denote the profiles of the outer IRA code from the node and edge perspectives, respectively, and \(d_c = 59\) denote the concentrated check node degree for the IRA code. We have that \(\{\rho_j\}\) satisfies a Poisson distribution with the average degree

\[
\bar{\rho} = 0.95 - \frac{r}{\bar{\lambda}},
\]

where \(\bar{\lambda}\) is the average degree of the profile \(\{\lambda_j\}\). Let \(I_{vc}\) and \(I_{cv}\) be the mutual information (MI) from the LT input nodes to the LT output nodes and the MI from the LT output nodes to the LT input nodes, respectively, and \(I_{vp}\) and \(I_{pv}\) be the MI from the LT input nodes to the IRA parity check nodes and the MI from the IRA parity check nodes to the LT input nodes, respectively. Let \(I_{dv}\) and \(I_{dc}\) be the MI from the channel to the LT input and output nodes, respectively. For the channel with AWGN of variance \(\sigma^2\), we have

\[
I_{dv} = I_{dc} = J(\sigma) \triangleq 1 - \int_{-\infty}^{\infty} \frac{e^{-(\xi - \sigma^2/2)/2\sigma^2}}{\sqrt{2\pi\sigma^2}} \log_2(1 + e^{-\xi})d\xi.
\]

Following the update of MI in [13], we present the update of the above MI as follows,

\[
I_{vc} = \sum_{i,j} \gamma_i \tilde{\rho}_j J\left(\sqrt{(j - 1)[J^{-1}(I_{cv})]^2 + [J^{-1}(I_{dv})]^2} + i[J^{-1}(I_{pv})]^2\right),
\]

\[
I_{vp} = \sum_{i,j} \tilde{\gamma}_i \rho_j J\left(\sqrt{j[J^{-1}(I_{cv})]^2 + [J^{-1}(I_{dv})]^2} + (i - 1)[J^{-1}(I_{pv})]^2\right),
\]

\[
I_{cv} = 1 - \sum_j \tilde{\lambda}_j J\left(\sqrt{(j - 1)[J^{-1}(1 - I_{vc})]^2 + [J^{-1}(1 - I_{dv})]^2}\right),
\]

and \(I_{pv} = 1 - J\left(\sqrt{(d_c - 1)[J^{-1}(1 - I_{vp})]^2}\right).

\]
Given an SNR value (defined as $P/\sigma^2$), the performance of iterative decoder can be evaluated as follows. First initialize $I_{dv} = I_{dc} = J(2/\sigma)$ according to (4.30) and all other MI to be zero, then iteratively update $I_{vc}$ and $I_{vp}$ according to (4.31) and (4.32), respectively, and then update $I_{cv}$ and $I_{pv}$ according to (4.33) and (4.34), respectively. The decoding is successful if $I_{vc}$ approaches one.

The threshold $J_p^{(1)}(r, \{\lambda_j\})$ is the minimum SNR that ensures successful decoding.

### 4.4.2.3 Code Profile Optimization Results

We show the code profile optimization results for $\mu = 1$. Fig. 4.8 shows the gap between the error free decoding SNR (no error decoding for 100 channel realizations) and the Gaussian signaling SNR (4.26) for both the optimized profile (4.27) and Luby’s profile, for different transmission rates. We fix the number of transmitted symbols to be 10000, and test the performance of the optimized profile for the code rates in $R_0$ for 4-, 16-, and 64-QAMs. Specifically, we simulate the optimized code profile and Luby’s profile for the code rates 0.2, 0.3, 0.4, and 0.5 for 4-QAM, the code rates 0.3, 0.4, and 0.5 for 16-QAM, and the code rates 0.4 and 0.5 for 64-QAM, which correspond to the transmission rates 0.4, 0.6, 0.8, 1.0, 1.2, 1.6, 2.0, 2.4, and 3.0. Fig. 4.8 shows that the maximal gaps for the optimized profile and Luby’s profile are 2.0dB and 3.5dB, corresponding to $\gamma = 0.63$ and $\gamma = 0.45$, respectively.

**Remark - Code Design for $\mu = 2$:** For $\mu = 2$, since one user may be decoded individually or jointly with another user, we need to find a good code profile for both the single-user decoding and the two-user joint decoding. Furthermore, for the two-user joint decoding, we need to find a good profile for the combinations of all channel gains, all modulation schemes, and all code rates for the two users. This makes the profile optimization rather complicated, and the gap of the optimized profile $\gamma = 0.45$ is significantly larger than that ($\gamma = 0.63$) for $\mu = 1$. This is the reason that practical coding schemes with group sizes $\mu = 2$ and 3 perform worse than that with group size $\mu = 1$.

### 4.4.3 Simulation Results for System with Optimized Codes

#### 4.4.3.1 SISO Channel

We use the same setup as that in Section 4.3.3 with group size $\mu = 1$ and employ the codes with the optimized profile and the optimized quantization tables corresponding to the layered and un-
layered coding. Fig. 4.9 shows the corresponding throughput, denoted as “layered, optimum” and “un-layered, optimum”, respectively, for the channel SNR $P/\sigma^2$ from 0dB to 9dB. The number of transmitted symbols is computed according (4.21) in Section 4.3.1.3. At each channel SNR, 1000 channel realizations are simulated and the throughput is the total number of information bits divided by the total number of transmitted symbols. Recall that based on the incremental transmission in Section 4.4.1 for each channel realization the number of transmitted symbols is given by $N_s + b\Delta N$ where $b$ is the number of additional bursts to guarantee successful decoding.

For comparison, we plot the throughput performance of the rate table $T = \{0.40, 0.80, 1.20, 1.60\}$ for the layered and un-layered coding schemes, denoted as “layered uniform” and “un-layered uniform”, respectively, in Fig. 4.9. It is seen that the optimized $T$ provides larger throughputs. Furthermore, we show the performance gain brought by the code profile optimization. We employ the same optimized rate quantization table $T$ and only substitute the optimized profile with Luby’s robust soliton distribution profile in [27] (c.f. Table I in [27], pp.2560) given by

$$
\Omega(x) = 0.008x + 0.494x^2 + 0.166x^3 + 0.073x^4 + 0.083x^5 \\
+ 0.056x^8 + 0.037x^9 + 0.056x^{19} + 0.025x^{65} + 0.003x^{66}.
$$

(4.35)

The throughput of Luby’s profile for the layered and un-layered coding schemes, denoted as “layered, Luby profile” and “un-layered, Luby profile”, respectively, is plotted in Fig. 4.9. It is seen that the optimized profile significantly outperforms Luby’s profile.

### 4.4.3.2 Extension to MISO Channel with Beamforming

Finally we consider a MISO interference channel where each transmitter employs three antennas and each receiver employs one antenna. The number of transmitter-receiver pairs is $K = 6$. The channel responses between any two antennas is an i.i.d. complex Gaussian random variable $\mathcal{CN}(0, 1)$. Assume that each user employs a transmission power $P$. Let $|T| = 4$.

We compare two schemes: one employs the simple channel matching (CM) beamforming together with the proposed CPGD with $\mu = 1$ and the optimized rate quantization and rateless code; the other scheme employs the linear MMSE beamforming together with the single-user decoding. For the latter, we design the corresponding practical coding scheme. We follow the procedure given in Section 4.3 to find the optimal cutoff rate $d_1$, which maximizes the sum-rate of all active users.
where the users with rates smaller than $d_1$ are inactivated, and then optimize the rate quantization set $T$. Fine tuning with incremental transmission is also employed with the initial number of transmitted symbols derived based on the same idea as that in Section 4.3.1.3. We also optimize the Raptor code profile according to Section 4.4.2.

Simulations are performed for both schemes at the SNR from 0dB to 9dB each with 1000 channel realizations. The throughput is plotted in Fig. 4.10 against the channel SNR $P/\sigma^2$. The CM beamforming with CPGD with and without layering, denoted as “CPGD layered” and “CPGD un-layered”, respectively, exhibits significant performance improvement over MMSE beamforming-based single-user coding scheme denoted as “MMSE single-user”. The CM beamforming which brings about stronger interference is more appropriate to the interference decoding since strong interference can be decoded before the decoding of intended messages.
Figure 4.9: The simulated throughput with QAM and optimized rateless codes.
Figure 4.10: The simulated throughput in MISO channel with beamforming.
Chapter 5

Group Decoding for Multi-relay Assisted Interference Channels

Interference channel is a fundamental building block for future wireless networks. The traditional decoding for interference channel is to treat interference as noise and suppress it [45,48]. The fundamental limits for such interference suppression schemes are well studied in [36,37,49,51]. However, due to the increasing demands for higher spectral efficiency, advanced decoding techniques are needed which exploit the structure of the interferers’ signals. It is shown in [38] that partial or full decoding of the interferers’ information is often helpful for recovering the desired information. Motivated by this premise some recent developments for the $K$-user interference channels propose that each receiver should partition the interfering transmitters into two groups, one group to be fully decoded along with the designated transmitter and the other to be treated as Gaussian noise [10]. In the decoding process, the users to be decoded are further partitioned into several smaller groups, whose sizes are smaller than some maximum group size, and in each step a group is decoded and subtracted from the received signal. We call this the constrained group decoder [39].

On the other hand, cooperative communications [16,52] can offer significant performance improvement over the direct-link communications through the relay assistance. For practical considerations, the relay is typically half-duplex and cannot receive and transmit simultaneously [20]. The relay-assisted transmission in interference channel has been studied in [53,54], where all interference is treated as noise. It remains a challenging problem to design coding and decoding strategies for relay-assisted interference channels that employ advanced receiver techniques such as the constrained group decoders.
We consider the design of a communication system with multiple transceivers and multiple relays, where the group decoders are employed at both relays and destinations. We classify the relays as either inband or hopping, where for inband relays there are direct source-destination links and for hopping relays there are no such links. We formulate the design problem as one to maximize the minimum information rate among all source-destination pairs, over all possible relay assignments, group decoding strategies at both the relays and the destinations, and rate vectors. This problem can be optimally solved for a hopping relay system with a fixed relay assignment. However, for general dynamic (i.e., channel-dependent) relay assignment, and for in-band relay systems, we propose efficient suboptimal algorithms to solve the above max-min design problem. Our results indicate that compared with the traditional systems that employ single-user receivers at the relays and destinations, as well as the AF-based multi-relay assisted system, the proposed systems that employ group decoding offer substantial increase in sustainable rate.

Different from the group decoder in [39, 40], where each receiver needs to decode only one transmitter, here we need to develop a group decoder and the associated rate allocation, where each receiver decodes possibly more than one transmitters. For example, for the hopping relay with dynamic relay assignment, each relay may be assigned to more than one transmitters and should decode all their information. Moreover, for dynamic relay assignment each relay also needs to identify the sources it assists, while in [39, 40] the source assisted by each relay is pre-specified. Therefore, following the ideas of group decoder given in [39], we propose an extended constrained group decoder where each receiver decodes a subset of transmitters instead of only one transmitter.

5.1 System Descriptions

5.1.1 Transmission Model

Consider a communication system with $K$ source-destination pairs, where the communication is assisted by $N$ intermediate relays. Denote the direct channel between source $j$ and destination $i$ by $h_{i,j}$ for $i, j \in K \triangleq \{1, \ldots, K\}$. Also, define $f_{n,j}$ as the channel from source $j$ to relay $n$ and $g_{i,n}$ as the channel from relay $n$ to destination $i$ for $i, j \in K$ and $n \in \mathcal{N} \triangleq \{1, \ldots, N\}$. We assume quasi-static block fading channels, i.e., the fading coefficients remain fixed during one transmission period and change to independent states afterwards.

We assume half-duplex relay transmission with synchronized relays that operate in the same
frequency band as the sources do. The transmissions are accomplished in two phases of durations $M_1$ and $M_2$ channel uses, respectively. In the first phase, the transmitters are active during the initial $M_1$ channel uses and the relays are in the listening mode. By denoting the transmitted signal of source $j$ by $x_j^1 \overset{\Delta}{=} [x_j^1(1), \ldots, x_j^1(M_1)]$ the received signal by relay $n \in \mathcal{N}$ during the first phase is

$$y_n^1 = \sum_{j=1}^{K} f_{n,j} x_j^1 + v_n^1,$$

(5.1)

where $v_n^1 \overset{\Delta}{=} [v_n^1(1), \ldots, v_n^1(M_1)]$ denotes the additive white Gaussian noise and its components are independently and identically distributed (i.i.d.) as $\mathcal{N}_C(0, \sigma^2)$. In the second phase, all sources and relays are active during the remaining $M_2$ channel uses. Similarly, by defining $x_j^2 \overset{\Delta}{=} [x_j^2(1), \ldots, x_j^2(M_2)]$ and $x_n^r \overset{\Delta}{=} [x_n^r(1), \ldots, x_n^r(M_2)]$ as the signals of source $j$ and relay $n$, respectively, during the second phase, the received signals at destination $i$ during the first and second phases are given respectively by

$$\text{Phase 1: } y_i^1 = \sum_{j=1}^{K} h_{i,j} x_j^1 + v_i^1,$$

$$\text{Phase 2: } y_i^2 = \sum_{j=1}^{K} h_{i,j} x_j^2 + \sum_{n=1}^{N} g_{i,n} x_n^r + v_i^2,$$

(5.2)

where $v_i^1$ and $v_i^2$ denote the additive white Gaussian noise with i.i.d. $\mathcal{N}_C(0, \sigma^2)$ components.

5.1.2 Relay Assignment and Relaying Modes

Each intermediate relay assists a group of source-destination pairs, such that each source-destination pair is assisted by at most one relay. We define the mapping $c : \mathcal{K} \rightarrow \{0\} \cup \mathcal{N}$ to characterize the assignments of the relays to the source-destination pairs, where

- $c(j) = n$ if source $j \in \mathcal{K}$ is assisted by relay $n \in \mathcal{N}$;
- $c(j) = 0$ if source $j \in \mathcal{K}$ is not assisted by any relay.

Let $\mathcal{S}_n$ be the set of sources assisted by relay $n \in \mathcal{N}$, and $\mathcal{S}_0$ be the set of sources not assisted by any relay.

$\mathcal{N}_C(a, b)$ and denotes a symmetric complex Gaussian distribution with mean $a$ and variance $b$. 
At the end of the first phase, each relay $n \in \mathcal{N}$ performs group decoding to decode the sources in $\mathcal{S}_n$; and in the second phase, relay $n$ re-encodes the message of each source $j \in \mathcal{S}_n$. By defining $\tilde{x}_j \triangleq [\tilde{x}_j(1), \ldots, \tilde{x}_j(M_2)]$ as the re-encoded message of source $j$, relay $n$ then employs superposition coding to combine the re-encoded signals of all sources in $\mathcal{S}_n$ and constructs $x'_n$ as

$$x'_n = \sum_{j \in \mathcal{S}_n} \tilde{x}_j, \text{ for } n \in \mathcal{N}. \quad (5.3)$$

From (5.2) and (5.3), the received signal at destination $i$ during the second phase is given by

$$y^2_i = \sum_{j=1}^{K} h_{i,j}x^2_j + \sum_{n=1}^{N} g_{i,n} \sum_{j \in \mathcal{S}_n} \tilde{x}_j + v^2_i \quad (5.4)$$

where $g_{i,c(j)}$ for $i, j \in \mathcal{K}$ denotes the gain from the relay that has re-encoded the message of source $j$ to destination $i$, and we define $g_{i,0} \triangleq 0$ for $i \in \mathcal{K}$. We assume that all the transmitted messages by the sources and their relay re-encoded versions satisfy the power constraint $P$, i.e., $\frac{1}{M_1} \mathbb{E}[\|x^1_j\|^2] \leq P$, $\frac{1}{M_2} \mathbb{E}[\|\tilde{x}_j\|^2] \leq P$, and $\frac{1}{M_2} \mathbb{E}[\|x^2_j\|^2] \leq P$ for all $j \in \mathcal{K}$.

We consider two types of relaying schemes. First, we consider hopping relays where there is no direct source-destination link ($\forall i, j, h_{i,j} = 0$) and destination $i$ decodes source $i$ through the received signals from the relays in the second phase. Secondly, we consider inband relays where the relays and sources share the same frequency band and destination $i$ decodes source $i$ by receiving both the direct and relayed transmissions.

### 5.1.3 Group Decoding at Relays and Destinations

Motivated by the premise that each destination, while not interested in decoding the messages of the interferers, can benefit from decoding them and improve its own communication, we allow each destination to decode a subset of the interferers along with the desired messages. Any interferer that is not decoded will be treated as Gaussian noise. More specifically, each destination $i \in \mathcal{K}$, although ultimately interested in decoding only its own signals $x^1_i$ and $x^2_i$, it can also decode some interference $x^1_j$ or $x^2_j$ for some $j \neq i$ in order to improve the transmission rates of $x^1_i$ and $x^2_i$. To achieve this, each destination $i \in \mathcal{K}$ performs multi-stage successive group decoding, by partitioning the user signals $\{x^1_j, x^2_j\}_{j \in \mathcal{K}}$ into several groups and decoding one group in each stage, until the
desired signals $\mathbf{x}_i^1$ and $\mathbf{x}_i^2$ are decoded. To control the decoding complexity, the size of each group should be smaller than some maximum value, hence the name constrained group decoder.

Similarly, we also allow the relays to perform constrained group decoding. Each relay $n \in \mathcal{N}$ partitions the sources signals $\{\mathbf{x}_j^1\}_{j \in \mathcal{K}}$ into several groups and decodes a group of sources in each stage in the successive group decoding, until the assigned sources $\mathcal{S}_n$ are decoded.

Determining the optimal choice of group partitioning at both the relays and the destinations depends on the objective function to be optimized. A typical objective is to maximize the minimum rate of sources. Usually a distributed group partitioning algorithm is favored, where each node locally finds its optimal group partition with limited information exchange between them. For example, for an interference channel with $K$ transceiver pairs, an optimal distributed group partitioning and rate allocation algorithm is developed in [39] to maximize the minimum rate of the transmitters, where each receiver locally partitions the groups and allocates the rates for the transmitters.

We employ the same criterion, which is to maximize the minimum rate of all sources. We aim to design a distributed group partitioning and rate allocation mechanism at both the relays and destinations, with limited information exchange among the sources, relays, and destinations. Note that in the proposed scheme we assume that each node only needs to know its incoming channels.

### 5.1.4 Problem Statement

We assume that the receivers at both the relays and destinations perform constrained group decoding, i.e., they decode a subset if the interferers along with the desired messages. Any interferer that is not decoded is treated as Gaussian noise. Denote $R_j^1$ and $R_j^2$ as the rates of the source messages in the first and second phases, respectively, and $t \triangleq \frac{M_j}{M_1 + M_2}$. Then, the overall rate of source $j \in \mathcal{K}$ is given by $R_j = tR_j^1 + (1-t)R_j^2$. Denote $\mathbf{R}^1 \triangleq [R_1^1, R_2^1, \ldots, R_K^1]$, $\mathbf{R}^2 \triangleq [R_1^2, R_2^2, \ldots, R_K^2]$, and $\mathbf{R} = t\mathbf{R}^1 + (1-t)\mathbf{R}^2$.

Let $\mathcal{C} \subseteq \{c : \mathcal{K} \rightarrow \{0\} \cup \mathcal{N}\}$ denote the set of valid relay assignment mappings. We are interested in the max-min rate allocation for the relay assisted interference channels, which maximizes the minimum rate among all sources, i.e., $\max_{j \in \mathcal{K}} \min_{c \in \mathcal{C}} R_j$, where the maximization is over all valid relay assignments, all possible group decoding strategies at the relays and the destinations, and all rate vectors $\mathbf{R}$ such that $\mathbf{R}^1$ is decodable by the relays, and $\mathbf{R}$ is decodable.
CHAPTER 5. GROUP DECODING FOR MULTI-RELAY ASSISTED INTERFERENCE CHANNELS

by the destinations.

5.2 Hopping Relay System

Since there is no direct source-destination link, the channel coefficient $h_{i,j} = 0$ for all $i, j \in K$. Therefore, the sources cannot send the messages in the second phase, i.e., $R_j^2 = 0$ for all $j \in K$. To ensure that the messages of all sources can be decoded by the desired destinations, each source must be assisted by one relay, i.e., $c(j) \neq 0, \forall j \in K$.

For each relay $n$, we partition the set of sources $K$ into $G^n = \{G^n_1, G^n_2, ..., G^n_{q_n+1}\}$. Relay $n$ employs a $q_n$-stage successive decoding, where in the $m^{th}$ stage, $1 \leq m \leq q_n$, the messages of $G^n_m$ are decoded by treating $\cup_{\ell=m+1}^{q_n+1} G^n_\ell$ as noise, and then subtracted from the received signal. To control the decoding complexity, we constrain that $|G^n_m| \leq \mu$ for all $1 \leq m \leq q_n$. Clearly the set of sources $S_n$ that should be decoded at relay $n$ satisfies $S_n \subseteq \cup_{m=1}^{q_n} G^n_m$, and $G^n_{q_n} \cap S_n \neq \emptyset$. We say that the partition $G^n$ is valid if

1. $|G^n_m| \leq \mu$ for all $m \in \{1, \ldots, q_n\}$;
2. $S_n \subseteq \cup_{m=1}^{q_n} G^n_m$, and $G^n_{q_n} \cap S_n \neq \emptyset$.

We define $G^n$ as the ensemble of all valid partitions of $K$ at relay $n$.

Similarly, each destination $i$ partitions $K$ into $Q^i = \{Q^i_1, Q^i_2, ..., Q^i_{p_i+1}\}$, and deploy a successive decoding procedure consisting of $p_i$ stages. In stage $m$, $1 \leq m \leq p_i$, the messages of $Q^i_m$ are decoded by treating $\cup_{\ell=m+1}^{p_i+1} Q^i_\ell$ as noise, and then subtracted from the received signal. We constrain that $|Q^i_m| \leq \mu$ for all $1 \leq m \leq p_i$, $i \in \cup_{m=1}^{p_i} Q^i_m$, and $Q^i_{p_i} \cap \{i\} \neq \emptyset$.

Recall that our objective is to jointly optimize the relay assignment, rate allocation and group decoding strategies at both the relay and destination sides, which in general is an intractable problem. In the proposed solution, we first obtain a good relay assignment and then optimize the rate allocation and group decoding for this assignment. In this section, we first solve the problem of rate allocation for any given relay mapping, and then discuss how to obtain a good relay mapping.
5.2.1 Rate Allocation for Fixed Relay Assignment

Given a relay mapping \( c(\cdot) \), the rate allocation for group decoding is formulated as

\[
\max_{\mathbf{R}, \{Q_i\}, \{G_n\}} \min_{j \in \mathcal{K}} R_j, \\
\text{s.t.} \quad \mathbf{R}^t \text{ is decodable by relays, } \mathbf{R} \text{ is decodable by destinations.}
\]  \tag{5.5}

The above problem can be optimally solved by combining solutions to the rate allocation problems for both source-relay and relay-destination interference channels, as stated in the following Theorem 5.1.

**Theorem 5.1:** Consider the following rate allocation problems for the source-relay and relay-destination interference channels, respectively,

\[
\max_{\mathbf{R}, \{G_n\}} \min_{j \in \mathcal{K}} R_j, \quad \text{s.t.} \quad \mathbf{R}^t \text{ is decodable by relays;}
\]  \tag{5.6}

\[
\max_{\mathbf{R}, \{Q_i\}} \min_{j \in \mathcal{K}} R_j, \quad \text{s.t.} \quad \mathbf{R} \text{ is decodable by destinations.}
\]  \tag{5.7}

Assume that \( \mathbf{R}^a \triangleq [R^a_i]_{i \in \mathcal{K}} \) and \( \mathbf{R}^b \triangleq [R^b_i]_{i \in \mathcal{K}} \) achieve the optimal solutions to (5.6) and (5.7), respectively. Then an optimal solution to (5.5), denoted as \( \hat{\mathbf{R}} \triangleq [\hat{R}_i]_{i \in \mathcal{K}} \), is given by \( \hat{R}_i = \min\{R^a_i, R^b_i\} \) for \( i \in \mathcal{K} \).

**Proof:** We prove Theorem 5.1 by contradiction. Assume that \( \tilde{\mathbf{R}}^a \) and \( \tilde{\mathbf{R}}^b \) are the solutions to (5.6) and (5.7), respectively, but \( \bar{\mathbf{R}} = [\bar{R}_j]_{j \in \mathcal{K}} \) where \( \bar{R}_j = \min\{\tilde{R}^a_j, \tilde{R}^b_j\} \) is not a solution to (5.5). Letting \( \hat{\mathbf{R}} = [\hat{R}_j]_{j \in \mathcal{K}} \) be a solution to (5.5), we have \( \min_{j \in \mathcal{K}} \hat{R}_j > \min_{j \in \mathcal{K}} \bar{R}_j \). Assume \( \hat{R}_{j'} = \min_{j' \in \mathcal{K}} \bar{R}_{j'} \) for some \( j' \in \mathcal{K} \), and without loss of generality let \( \hat{R}_{j'} = \hat{R}^a_{j'} \). We have

\[
\min_{j \in \mathcal{K}} \hat{R}_j > \min_{j \in \mathcal{K}} \bar{R}_j = \min_{j \in \mathcal{K}} \hat{R}^a_{j'},
\]  \tag{5.8}

which this contradicts with the fact that \( \hat{\mathbf{R}} \) is decodable for the source-relay interference channel where \( \hat{\mathbf{R}}^a \) is a solution to the max-min rate allocation problem. On the other hand, since the minimum rate of any solution to (5.5) should be smaller than those of both (5.6) and (5.7), and \( \hat{\mathbf{R}} \) achieves this bound, \( \hat{\mathbf{R}} \) is an optimal solution to (5.5). □

According to (5.4), the interference channel at the destinations is equivalent to a conventional \( K \)-user interference channel (i.e., there are \( K \) transmitter-receiver pairs and receiver \( i \) needs to decode the message from transmitter \( i \)), where the channel gain from transmitter \( j \) to receiver \( i \) is given by \( g_{i,c(j)} \). On the other hand, for the source-relay side, there are \( K \) transmitters and \( N \)
receivers, where \( K \) is not necessarily equal to \( N \) and each relay \( n \in \mathcal{N} \) is interested in decoding a subset \( \mathcal{S}_n \) of sources. In the sequel we propose a distributed algorithm to solve the rate allocation problem (5.6), which can also be used to solve (5.7).

For any subset \( \mathcal{A} \subseteq \mathcal{K} \), define \( \mathbf{R}_\mathcal{A} \triangleq [R_j]_{j \in \mathcal{A}} \). Also, for any disjoint \( \mathcal{D}, \mathcal{V} \subseteq \mathcal{K} \) and \( n \in \mathcal{N} \), define

\[
L_n(\mathcal{D}, \mathcal{V}) \triangleq \log \left( 1 + \frac{\sum_{j \in \mathcal{D}} |f_{n,j}|^2 P}{\sigma^2 + \sum_{j \in \mathcal{V}} |f_{n,j}|^2 P} \right),
\]

(5.9) as the achievable sum rate of the sources in \( \mathcal{D} \) at relay \( n \) where those in \( \mathcal{V} \) are treated as AWGN. Furthermore, for any disjoint \( \mathcal{U}, \mathcal{V} \subseteq \mathcal{K} \), we define

\[
C_n(\mathcal{U}, \mathcal{V}) \triangleq \{ \mathbf{R}_\mathcal{U} \in \mathbb{R}_+^{|\mathcal{U}|} : \sum_{j \in \mathcal{D}} R_j \leq L_n(\mathcal{D}, \mathcal{V}), \forall \mathcal{D} \subseteq \mathcal{U} \}
\]

(5.10) as the achievable rate region of sources in \( \mathcal{U} \) when treating those in \( \mathcal{V} \) as AWGN. Given the partition \( \mathcal{G}^n = \{ \mathcal{G}^{n}_m \}_{m=1}^{q_n+1} \) for relay \( n \), at the \( m \)th stage the sources in \( \mathcal{G}^{n}_m \) are decodable by treating those in \( \bigcup_{\ell>m} \mathcal{G}^{n}_\ell \) as AWGN, i.e., \( \mathbf{R}_{\mathcal{G}^{n}_m} \in C_n(\mathcal{G}^{n}_m, \bigcup_{\ell>m} \mathcal{G}^{n}_\ell) \) for \( m \in \{1, 2, ..., q_n\} \) and \( n \in \mathcal{N} \). Then, the rate allocation problem (5.11) can be reformulated explicitly as

\[
\max_{\mathbf{R}_{\mathcal{G}^n}} \min_{j \in \mathcal{K}} R_j \quad \text{s.t.} \quad \mathbf{R}_{\mathcal{G}^{n}_m} \in C_n(\mathcal{G}^{n}_m, \bigcup_{\ell>m} \mathcal{G}^{n}_\ell), \quad \forall m \in \{1, \ldots, q_n\}, n \in \mathcal{N}.
\]

(5.11)

We propose a distributed solution to the rate allocation problem (5.11) with limited information exchange between the sources and relays. To reduce the amount of information exchange, each relay \( n \in \mathcal{N} \) locally determines its group partition \( \mathcal{G}^n = \{ \mathcal{G}^{n}_m \}_{m=1}^{q_n+1} \) and performs a tentative rate allocation based on the strength of its incoming links. Then, each source combines its rates allocated by the relays to solve the rate allocation problem (5.11). In this way, the information exchange is the feedback of the allocated rates from the relays to the sources.

In the following we formulate a local group partitioning and rate allocation scheme at the relays. At each relay \( n \in \mathcal{N} \), given a valid partition \( \mathcal{G}^n = \{ \mathcal{G}^{n}_m \}_{m=1}^{q_n+1} \), we define

\[
\theta_n(\mathcal{G}^n) \triangleq \left\{ \begin{array}{ll}
\max_{\mathbf{R}} & \min_{j \in \mathcal{K}} R_j, \\
\text{s.t.} & \mathbf{R}_{\mathcal{G}^{n}_m} \in C_n(\mathcal{G}^{n}_m, \bigcup_{\ell>m} \mathcal{G}^{n}_\ell), \quad \forall m \in \{1, \ldots, q_n\}, n \in \mathcal{N}.
\end{array} \right.
\]

(5.12)

We maximize \( \theta_n(\mathcal{G}^n) \) over all possible valid partitions \( \mathcal{G}^n \in \mathcal{G}^n \), subject to the constraint that all
sources in $S_n$ can be decoded at relay $n$, as follows

$$\theta^{\ast}_n \triangleq \max_{G^n \in S_n} \theta_n(G^n). \quad (5.13)$$

Next, for each relay $n \in \mathcal{N}$, we propose a solution to each sub-problem (5.13) based on the strength of its incoming links $|f_{n,j}|^2$ for $1 \leq j \leq K$.

Before elaborating on Algorithm 5.1, the optimal solution to sub-problem (5.13), we first introduce the notion of the rate increment margin. For any decodable rate $\bar{R}$ and any disjoint $U, V \subseteq K$, define

$$\Delta_n(U, V, \bar{R}) \triangleq \min_{D \subseteq U, D \neq \emptyset} \frac{L_n(D, V) - \sum_{j \in D} \bar{R}_j |D|}{|D|} , \quad (5.14)$$

as the rate increment margin for the sources in $U$ when treating those in $V$ as noise at relay $n$. In other words, if each transmitter in $U$ increases its rate beyond $\Delta_n(U, V, \bar{R})$ then at relay $n$ the sources in $U$ become undecodable when treating those in $V$ as noise. Note that, for $\bar{R} = 0$ and the group partition $G^n = \{G^n_m\}_{m=1}^{q_n+1}$ for relay $n$, for $1 \leq m \leq q_n$, the maximum value of the minimum rate allocated rate to $G^n_m$ is given by $\Delta_n(G^n_m, \cup_{\ell > m} G^n_\ell, 0)$.

Algorithm 5.1 optimally solves the rate allocation problem (5.13) in a greedy manner, by successively partitioning the group that maximizes the rate increment margin [c.f. (5.14)] over the input decodable rate $\bar{R}$. Let $\mathcal{G}$ denote the set of decoded sources, which is initialized to be the empty set $\emptyset$; and let $\mathcal{D}$ denote the set of undecoded sources, which is initialized to be the set of all sources $\mathcal{K}$. In each iteration $\ell$, Algorithm 5.1 identifies a group of sources $Q \subseteq \mathcal{D}$, such that $|Q| \leq \mu$ and the minimum rate allocated to the sources $Q$ by treating $\mathcal{D} \setminus Q$ as noise, i.e., $\Delta_n(Q, \mathcal{D} \setminus Q, \bar{R})$, is maximized. Subsequently the undecoded source set $\mathcal{G}$ is updated to $\mathcal{G} \cup Q$ and the decoded source set $\mathcal{D}$ is updated to $\mathcal{D} \setminus Q$. The group of sources decoded in the $\ell^{th}$ stage, $\mathcal{G}^{\ell}_n$, is the optimal group $Q$ that maximize $\Delta_n(Q, \mathcal{D} \setminus Q, \bar{R})$, i.e.,

$$\mathcal{G}^{\ell}_n = \arg \max_{Q \neq \emptyset, Q \subseteq \mathcal{D}, |Q| \leq \mu} \Delta_n(Q, \mathcal{D} \setminus Q, \bar{R}); \quad (5.15)$$

and the rate increment of sources in $\mathcal{G}^{\ell}_n$, denoted as $r^n_j$ for $j \in \mathcal{G}^{\ell}_n$, is given by

$$r^n_j = \Delta_n(\mathcal{G}^{\ell}_n, \mathcal{D} \setminus \mathcal{G}^{\ell}_n, \bar{R}). \quad (5.16)$$

The sources $\mathcal{G}^{\ell}_n$ are then removed from the set of undecoded sources $\mathcal{D}$ and added to the set of decoded set $\mathcal{G}$. We iterate the above group partitioning process until $S_n \subseteq \mathcal{G}$, i.e., all desired sources
\(S_n\) are decoded. Then, since further decoding is not necessary, we terminate the group partitioning process and set the rate of undecoded sources to be positive infinite, i.e., \(r^n_j = +\infty\) for \(j \in D\).

Note that Algorithm 5.1 generalizes the distributed group partitioning and rate allocation algorithm proposed in \[39\]. By setting the number of relays equal to the number of sources, i.e., \(N = K\), and the assigned sources \(S_n = \{n\}\) for each relay \(n \in N\), Algorithm 5.1 becomes the distributed group partitioning and rate allocation algorithm in \[39\].

### Algorithm 5.1 - Group Partitioning and Rate Allocation for Extended Constrained Group Decoder

1: Initialize \(D = K, G = \emptyset, \ell = 1,\) and \(q_n = 0\).
2: repeat
3: Find \(G^n_\ell = \arg \max_{Q \neq \emptyset, Q \subseteq D, |Q| \leq \mu} \Delta_n(Q, D \setminus Q, \tilde{R});\)
4: Set \(r^n_j = \Delta_n(G^n_\ell, D \setminus G^n_\ell, \tilde{R})\) for all \(j \in G^n_\ell;\)
5: Update \(D \leftarrow D \setminus G^n_\ell, G \leftarrow G \cup G^n_\ell;\)
6: Update \(q_n \leftarrow q_n + 1\), and \(\ell \leftarrow \ell + 1;\)
7: if \(S_n \subseteq G\)
8: \(r^n_j = +\infty\) for all \(j \in D, Q^{n+1}_q \leftarrow D,\) and \(D \leftarrow \emptyset;\)
9: end if
10: until \(D = \emptyset\)
11: Output \(\{r^n_j\}_{j \in K}\) and the partition \(G^n = \{G^n_1, \ldots, G^n_{q_n}, G^n_{q_n+1}\}\).

### Algorithm 5.2 - Max-Min Fair Rate Allocation

1: Initialize \(R^{(0)} = 0, q = 0;\)
2: repeat
3: \(\tilde{R} \leftarrow R^{(q)};\)
4: for \(n \in N\) do
5: Run Algorithm 5.1;
6: end for
7: Update \(R^{(q+1)}_j = R^{(q)}_j + \min_{n \in N} r^n_j, \forall j \in K;\)
8: Update \(q \leftarrow q + 1;\)
9: until \(R^{(q)}\) converges
10: Output \(R^{(q)}\) and the partitions \(\{G^n\}_{n \in N}.\)

The optimality of Algorithm 5.1 is formally stated in Theorem 5.2. It is very interesting that Algorithm 5.1 optimally solves the rate allocation problem \[5.13\] in such a greedy manner. The
proof is similar to that of the optimality of the distributed rate allocation algorithm in [39]. The key ingredient is the same, the submodularity and transitivity of achievable rate function \( L_n(D, V) \) \[c.f. (5.9)\], as follows,

\[
L_n(D_1, V) + L_n(D_2, V) \geq L_n(D_1 \cap D_2, V) + L_n(D_1 \cup D_2, V),
\]

\[
L_n(U, V) + L_n(D, U \cup V) = L_n(D \cup U, V), \text{ for disjoint sets } U, V, \text{ and } D. \quad (5.17)
\]

We present Theorem 5.2 as follows, and omit the proof.

**Theorem 5.2:** The partition \( G^n = \{G^n_m\}^{q_n+1}_{m=1} \) yielded by Algorithm 5.1 for the setting \( \bar{R} = 0 \) is an optimal solution to (5.13). Moreover,

\[
\theta_n^* = \min_{j \in K} r_n^j,
\]

where \([r_1^n, r_2^n, ..., r_K^n] \) is yielded by Algorithm 5.1.

Algorithm 5.2 iteratively runs Algorithm 5.1 for solving the rate allocation of the source-relay interference channel. The input decodable rate \( \bar{R} \) to Algorithm 5.1 is initialized to be zero. In each iteration, each relay \( n \in \mathcal{N} \) performs the group partitioning and rate allocation according to Algorithm 5.1, and then feedbacks the allocated rate increment \( r_n^j \) to source \( j \), for \( j \in K \). Since the partitioned groups should be decodable at all relays, the rate increment for each source \( j \in K \) is the minimum of the rate increments feedbacked from all relays, i.e., \( \min_{n \in \mathcal{N}} r_n^j \). Then, each source broadcasts its updated rate to the relays, and in the next iteration the decodable rates \( \bar{R} \) for the relay group partitioning is set to be the updated source rates. The following three points illustrate the optimality of Algorithm 5.2. The proof is the same as that given in [39] and thus omitted here.

1. For all \( q \geq 1 \), \( R^{(q)} \) is decodable and max-min optimal, i.e., for any other arbitrary decodable rate vector \( \bar{R} \) we have

\[
\min_{j \in K} R_j^{(q)} \geq \min_{j \in K} \bar{R}_j, \quad \forall \ q \geq 1.
\]

2. The algorithm is monotonic in the sense that \( R^{(p+1)} \geq R^{(q)} \) for \( \forall \ q \geq 1 \) and is convergent.

3. The rate vector yielded by Algorithm 5.2 upon convergence is decodable, max-min fair, and pareto-optimal.
5.2.2 Dynamic Relay Assignment

In this subsection we focus on the selection of relay mapping for dynamic relay assignment. The selection is performed in two steps, namely, the coarse assignment and the fine adjustment. In the coarse assignment step, we find a relay assignment $c(\cdot)$ to maximize the minimum rate for the source-relay channels. It is possible that using such obtained relay assignment $c(\cdot)$, some destination $i \in K$ may suffer from strong interference when decoding source $i$, resulting in a very low rate for that source. Then in the fine adjustment step, we adjust the relay mapping to avoid such low-rate events.

Before presenting the above two steps for relay assignment, we first outline some main considerations in our design.

- **Size balance:** The coarse assignment step imposes that the number of sources forwarded by each relay should be approximately equal. Otherwise, if many sources are forwarded by one relay, then in the equivalent relay-destination channel they share the same channel gain. Such an interference channel with equally strong interferers as the desired source has a low achievable rate.

- **Identifying low-rate sources:** Using the relay assignment $c(\cdot)$ obtained in the coarse assignment step, we identify the low-rate sources coarsely corresponding to the two stages of the successive decoding, namely, the sources decoded with the strongest sources and those decoded with the desired sources. If a low-rate source is identified, we adjust the relay mapping accordingly. Usually one or two iterations of this procedure suffice to significantly increase the minimum rate of all sources.

- **Potential low-rate sources:** According to Algorithm 5.1, for the benchmark rate $\bar{R} = \mathbf{0}$, the group decoder decodes the sources in an order from those with strong links to those with weak links. For destination $i$, we denote the strongest incoming link $g_{i,n}^m \triangleq \max_{n, S_n \neq \emptyset} g_{i,n}$, and set the strong sources as those with the link gains larger than a threshold, say $\delta_1^i = 0.9 \cdot g_{i,n}^m$, given by

$$A_{i}^1 \triangleq \bigcup_{n:g_{i,n}^m > \delta_1^i} S_n.$$  \hspace{1cm} (5.20)

We assume that if the size $|A_{i}^1| \leq \mu$, then the sources decoded in the first stage is given by
\(D_1^1 = A_1^1\); otherwise \(D_1^1\) consists of the strongest \(\mu\) sources in \(A_1^1\). An estimate of the rate allocated to each source in \(D_1^1\) is given by

\[
r_1^1 = \frac{1 - t}{|D_1^1|} \log_2 \left( 1 + \frac{\sum_{j \in D_1^1} |g_{i,c(j)}|^2 P}{\sigma^2 + \sum_{j \in K \setminus D_1^1} |g_{i,c(j)}|^2 P} \right). \tag{5.21}
\]

Similarly, to identify the sources decoded with the desired source \(i\), each destination \(i \in K\) sets another threshold, say \(\delta_i^2 = 0.9 \cdot g_{i,c(i)}\), and stores the sources with the channel gain larger than \(\delta_i^2\) but smaller than \(g_{i,c(i)}\), given by

\[
A_2^i \triangleq \bigcup_{n: \delta_i^2 < g_{i,n} \leq g_{i,c(i)}} S_n. \tag{5.22}
\]

Since at each destination \(i \in K\) the group decoder decodes sources \(j\) with the channel gain \(g_{i,c(j)}\) in a decreasing order, we assume that when source \(i\) is decoded, the sources with link gains smaller than \(g_{i,c(i)}\), denoted as \(\bar{D}_2^i = \{j : g_{i,c(j)} \leq g_{i,c(i)}\}\), have not been decoded. We assume that the sources decoded with the desired source \(i\) is given by \(D_2^i = A_2^i\) if \(|A_2^i| \leq \mu\) and \(D_2^i\) consists of the \(\mu\) strongest sources in \(A_2^i\) if \(|A_2^i| > \mu\). An estimate of the rates allocated to each source in \(D_2^i\) is given by

\[
r_2^i = \frac{1}{|D_2^i|} \log_2 \left( 1 + \frac{\sum_{j \in D_2^i} |g_{i,c(j)}|^2 P}{\sigma^2 + \sum_{j \in D_2^i \setminus D_1^i} |g_{i,c(j)}|^2 P} \right). \tag{5.23}
\]

- **Feasible relay re-assignment:** We assume that when source \(i\) is re-assigned to relay \(n\), the new link gain from relay \(n\) to destination \(i\) should not be significantly smaller than the original one \(g_{i,c(i)}\). Based on this, we set a threshold, say \(\delta_i^3 = 0.8 \cdot g_{i,c(i)}\), and each destination \(i \in K\) finds a set of relays that source \(i\) can be re-assigned to, given by

\[
\mathcal{F}_i^D = \{n : n \in N, g_{i,n} > \delta_i^3\}. \tag{5.24}
\]

Based on the above considerations, in the following we present the two steps of our proposed relay assignment procedure.

### 5.2.2.1 Coarse Assignment

We consider the source-relay interference channel and find a good relay mapping \(c(\cdot)\) that maximizes the minimum rate of all sources, subject to the constraint that the each relay assists at most \([K/N]\) sources.
To do this, we start by forcing each relay to decode all sources, i.e., $S_n = K$ for all $n \in \mathcal{N}$, and run Algorithm 5.1 by setting the initial decodable rate $\bar{R} = 0$ [c.f. (5.14)] and obtain the partitions \( \{G^n_m\}_{m=1}^{\infty} \) \( (G^n_{n+1} = \emptyset) \) and the rates \([r^n_j]\) \( j \in \mathcal{K} \) for all sources. If a source \( j \in G^n_m \) is decoded by relay \( n \), the sources belonging to \( G^n_1, G^n_2, \ldots, G^n_m \) should be decoded before source \( j \), and thus the minimum rate yielded is given by

\[
R^n_j \triangleq \min_{p \in \cup_{1 \leq \ell \leq m} G^n_\ell} r^n_p, \quad \forall \ j \in G^n_m.
\]

The coarse assignment is performed in the following two steps:

- **Initial assignment**: Each source \( j \) ranks the received rates \( R^n_j \) for \( n \in \mathcal{N} \) in a descending order as follows,

\[
R^\sigma_j(1) \geq R^\sigma_j(2) \geq \ldots \geq R^\sigma_j(N),
\]

where \( \{\sigma_j(1), \sigma_j(2), \ldots, \sigma_j(N)\} \) is a permutation set of the index set \( \{1, 2, \ldots, N\} \). We initialize \( S_n = \{j : \sigma_j(1) = n\} \) for \( n \in \mathcal{N} \) and the rank index \( d_j = 1 \) for \( j \in \mathcal{K} \), which means that source \( j \) is assigned to relay \( \sigma_j(d_j) = \sigma_j(1) \).

- **Size balance**: By noting that we constrain each relay to assist at most \( \lceil K/N \rceil \) sources, we find a relay \( n \) with \( |S_n| > \lceil K/N \rceil \) (if any) and re-assign some of its assisted sources to other relays. We find the source \( j \in S_n \) that maximizes the next rate \( R^{\sigma_j(d_j+1)}_j \) if it is assigned to relay \( \sigma_j(d_j+1) \), given by

\[
k = \arg \max_{j \in S_n, d_j < N} R^{\sigma_j(d_j+1)}_j.
\]

Then we remove source \( k \) from \( S_n \) and add it to \( S_{\sigma_k(d_k+1)} \), and update \( d_k \leftarrow d_k + 1 \). We iterate the above process until \( |S_n| \leq \lceil K/N \rceil \) for all \( n \in \mathcal{N} \).

It can be proved that the size balance step terminates in finite number of iterations, and in practice less than five iterations suffice. Skipping the size balance step typically only causes a slight rate loss.

### 5.2.2.2 Fine Adjustment

Given the relay mapping \( c(\cdot) \) obtained from the coarse assignment step, each destination \( i \) identifies the sets \( \mathcal{A}^1_i \) and \( \mathcal{A}^2_i \) according to (5.20) and (5.22), respectively, as well as the set of feasible relays
\[ \mathcal{F}_i^D \] given by (5.24). Then, we perform the following adjustment of the relay mapping \( c(\cdot) \) by alternatively identifying the low-rate sources decoded with the strongest sources and those with the desired source.

- **Low-rate source decoded with the strongest sources:** For each destination \( i \in \mathcal{K} \), we compute \( r_i^1 \) according to (5.21), and find the destination that minimizes \( r_i^1 \) given by \( m^1 = \arg \min_{i \in \mathcal{K}} r_i^1 \).

  A low rate event is detected for destination \( m^1 \) if and only if \( |A^1_{m^1}| > \mu \), because during the decoding of \( D^1_{m^1} \), the undecoded sources in \( A^1_{m^1} \setminus D^1_{m^1} \) may cause strong interference.

  If a low-rate event is detected, we re-assign a source \( j \in A^1_{m^1} \setminus \{m^1\} \) to another relay. Specifically, for each source \( j \in A^1_{m^1} \setminus \{m^1\} \), we define its valid alternative relays as \( n \in \mathcal{F}_j^D \), such that \( R_{nj} > r_{m^1}^1 \), i.e., re-assigning source \( j \) to relay \( n \) will not yield a rate of source \( j \) in the source-relay channel smaller than \( r_{m^1}^1 \). Then for each relay \( n \), denote \( B^1_n \subseteq A^1_{m^1} \setminus \{m^1\} \) as the set of sources that it can serve as their valid alternative relay. Note that re-assigning a source \( j \in B^1_n \) to a relay \( n \) will cause an interference with the channel gain \( g_{m^1,n} \) to destination \( m^1 \).

  To minimize it, destination \( m^1 \) selects the relay

  \[
  n^1 = \arg \min_{n \in \mathcal{N}, B^1_n \neq \emptyset} |g_{m^1,n}|,
  \]

  and re-assigns a source \( j^1 \in B^1_n \) to relay \( n^1 \), i.e., updates \( c(j^1) = n^1 \).

- **Low-rate source decoded with the desired source:** For each destination \( i \in \mathcal{K} \) we compute \( r_i^2 \) according to (5.23), and find the destination that minimizes \( r_i^2 \) given by \( m^2 = \arg \min_{i \in \mathcal{K}} r_i^2 \).

  A low-rate event is detected if and only if \( |A^2_{m^2}| > \mu \).

  When a low-rate event is detected, for each relay \( n \), denote \( B^2_n = \{ j \in A^2_{m^2} \setminus \{m^2\} : n \in \mathcal{F}_j^D, R_{nj} > r_{m^2}^2 \} \) as the set of sources that it can serve as their valid alternative relay. Then destination \( m^2 \) selects the relay

  \[
  n^2 = \arg \min_{n \in \mathcal{N}, B^2_n \neq \emptyset} |g_{m^2,n}|,
  \]

  and re-assigns a source \( j^2 \in B^2_n \) to relay \( n^2 \), i.e., updates \( c(j^2) = n^2 \).

  After each relay re-assignment, we update the sets \( A^1_i, A^2_i, \) and \( \mathcal{F}_i^D \). In practice, only one or two iterations of the above fine adjustment steps suffice.
5.3 Inband Relay System

5.3.1 Destination Interference Channels

5.3.1.1 Group Decoder for the Destinations

Consider the signal model for the inband relay system in Section 5.1.2. From (5.2) and (5.4), it can be seen that the message intended to destination $i$ are transmitted by source $i$ and relay $c(i)$. Based on this, we consider an equivalent virtual interference channel at the $K$ destinations with $2K$ transmitters $\mathcal{T} \triangleq \{1, 2, \ldots, 2K\}$. For $i \in \mathcal{K}$, transmitter $i$ sends $x^1_i$ from source $i$ in the first phase [c.f.(5.2)] (link gain $h_{i,j}$ from source $j$ to destination $i$) and the re-encoded $\tilde{x}_i$ from relay $c(i)$ in the second phase [c.f.(5.4)] to the destinations (link gain $g_{i,c(j)}$ from relay $c(j)$ to destination $i$); and transmitter $i + K$ sends $x^2_{i}$ to the destinations in the second phase (link gain $h_{i,j}$ from source $j$ to destination $i$). Destination $i$ should decode transmitters $i$ and $i + K$.

For this purpose, destination $i$ partitions $\mathcal{T}$ into $\mathcal{Q}^i = \{Q^i_1, Q^i_2, \ldots, Q^i_{p_i+1}\}$, and deploys a successive decoding procedure consisting of $p_i$ stages. In stage $m$, $1 \leq m \leq p_i$, the messages of $Q^i_m$ are decoded by treating $\bigcup_{\ell=m+1}^{p_i+1} Q^i_{\ell}$ as noise, and then subtracted from the received signal. The same as those for the group decoder for the source-relay interference channel, we say that a partition of $\mathcal{T}$ at destination $i$ is valid if the following conditions are satisfied,

1. $|Q^i_m| \leq \mu$ for all $1 \leq m \leq p_i$;

2. the desired transmitters $\{i, i + K\} \subseteq \bigcup_{m=1}^{p_i} Q^i_m$, and $Q^i_{p_i} \cap \{i, i + K\} \neq \emptyset$.

5.3.1.2 Rate Allocation for the Destinations

We specify the achievable sum rate for the destination interference channel. For any $\mathcal{D}, \mathcal{V} \subseteq \mathcal{T}$, let $L^2_i(\mathcal{D}, \mathcal{V})$ be the sum rate of transmitters $\mathcal{D}$ when those in $\mathcal{V}$ are treated as noise. Since the destination interference channel is a time sharing of the sum rates corresponding to the two phases, we have

$$L^2_i(\mathcal{D}, \mathcal{V}) = tL^S_i(\mathcal{D} \cap \mathcal{K}, \mathcal{V} \cap \mathcal{K}) + (1-t)L^R_i(\mathcal{D}, \mathcal{V}), \tag{5.30}$$

where $L^S_i(\mathcal{D} \cap \mathcal{K}, \mathcal{V} \cap \mathcal{K})$ and $L^R_i(\mathcal{D}, \mathcal{V})$ are the sum rates corresponding to the first and second phases, respectively. The sum rate for the first phase is denoted as $L^S_i(\mathcal{D} \cap \mathcal{K}, \mathcal{V} \cap \mathcal{K})$ because in the first phase only transmitters $\mathcal{K} = \{1, 2, \ldots, K\}$ are transmitting. Let $\tilde{\mathcal{K}} = \{K + 1, K + 2, \ldots, 2K\}$. 
Based on the characterization of the equivalent virtual interference channel, the achievable sum rates corresponding to the two phases are given as follows:

\[ L_S^i(D \cap K, V \cap K) = \log_2 \left( 1 + \frac{\sum_{j \in D \cap K} |h_{i,j}|^2 P}{\sigma^2 + \sum_{j \in V \cap K} |h_{i,j}|^2 P} \right), \]

and

\[ L_R^i(D, V) = \log_2 \left( 1 + \frac{\sum_{j \in D \cap K} |g_{i,c}(j)|^2 P + \sum_{j \in D \cap K} |h_{i,j-K}|^2 P}{\sigma^2 + \sum_{j \in V \cap K} |g_{i,c}(j)|^2 P + \sum_{j \in V \cap K} |h_{i,j-K}|^2 P} \right). \] (5.31)

For each destination \( i \in K \), given an \( 2^K \)-tuple decodable rate \( \mathbf{R}_D = [R_D^1, R_D^2, \ldots, R_D^{2^K}] \) for the \( 2^K \) virtual transmitters at the destinations, we define

\[ \Delta^2_i(U, V, \mathbf{R}_D) \triangleq \min_{D \subseteq U, D \neq \emptyset} \frac{L_i^D(D, V) - \sum_{j \in D} R_D^j}{|D|}, \] (5.32)

as the minimum rate margin for the sources in \( U \) when treating those in \( V \) as noise at destination \( i \).

The group partitioning and rate allocation at each destination \( i \in K \) can also be performed through Algorithm 5.1 with some modifications. We initialize the set of undecoded transmitters \( D = T \) and the set of the desired transmitters \( S_i = \{i, i+K\} \), and substitute the rate margin \( \Delta_n(U, V, \mathbf{R}) \) by \( \Delta^2_i(U, V, \mathbf{R}_D) \) obtained from (5.32).

Recall that in the original rate allocation problem, we maximize the minimum rate for each source \( j \), i.e., maximize \( \min_{j \in K} \left( tR_j^1 + (1-t)R_j^2 \right) \). It is difficult to extend the rate allocation for the constrained group decoder, which is to maximize the minimum rate of the \( 2^K \) information sources, i.e., \( \min_{j \in K} \min\{tR_j^1, (1-t)R_j^2\} \), to maximizing the minimal sum rate of the two parts, i.e., \( \min_{j \in K} \left( tR_j^1 + (1-t)R_j^2 \right) \). Therefore, we change the optimization objective and maximize the minimum rate instead, i.e., \( \min_{j \in K} \min\{tR_j^1, (1-t)R_j^2\} \).

### 5.3.2 Rate Allocation for Fixed Relay Assignment

We assume a given relay assignment which assigns each source \( j \) to relay \( c_0(j) \). Since there are direct source-destination links, which may be stronger than the source-relay links, source \( j \) may not make use of its pre-assigned relay \( c_0(j) \), i.e., each source \( j \) is either assisted by relay \( c_0(j) \) or not assisted by any relay. In what follows, we first identify the set of sources \( S_0 \) that do not need the relay assistance, and then perform rate allocation for all sources.

In the first step, we perform rate allocation independently for the source-relay and the source-destination interference channels. More specifically, assuming that each source \( j \in K \) is decoded
by relay $c_0(j)$, we run Algorithms 5.1 and 5.2 to solve the following rate allocation problem for the source-relay interference channel

$$[\tilde{R}_j^R]_{j \in \mathcal{K}} \triangleq \left\{ \begin{array}{ll}
\arg \max R \min_{j \in \mathcal{K}} R_j \\
s.t. \ R \ is \ decodable \ at \ the \ relays.
\end{array} \right. \number{5.33}$$

Moreover, assuming that each destination $j \in \mathcal{K}$ is interested in decoding source $j$, we run Algorithms 5.1 and 5.2 to solve the following rate allocation problem for the source-destination interference channel

$$[\tilde{R}_j^D]_{j \in \mathcal{K}} \triangleq \left\{ \begin{array}{ll}
\arg \max R \min_{j \in \mathcal{K}} R_j \\
s.t. \ R \ is \ decodable \ at \ the \ destinations.
\end{array} \right. \number{5.34}$$

The set $S_0$ is given by the sources $j$ for which $\tilde{R}_j^R < \tilde{R}_j^D$, i.e., $S_0 = \{ j : j \in \mathcal{K}, \tilde{R}_j^R \leq \tilde{R}_j^D \}$. The relay assignment is given by $c(j) = c_0(j)$ for $j \not\in S_0$ and $c(j) = 0$ for $j \in S_0$.

In the next step, based on the relay assignment $c(\cdot)$ obtained in the first step, we perform rate allocation for the interference channels at both the relays and the destinations. Given the relay assignment $c(j)$ for each source $j \in \mathcal{K}$, we run Algorithms 5.1 and 5.2 to solve the following rate allocation problem for the source-relay interference channel

$$[\tilde{R}_j^1]_{j \in \mathcal{K}} \triangleq \left\{ \begin{array}{ll}
\arg \max R \min_{j \in \mathcal{K}} R_j \\
s.t. \ R \ is \ decodable \ by \ the \ relays.
\end{array} \right. \number{5.35}$$

For the interference channel at the destinations, letting $\mathbf{R}^D \triangleq [R_j^D]_1 \leq 2K = [R_j^1 \ R_j^2]_{j \in \mathcal{K}}$ be the split rates for the sources, the rate allocation problem is given by

$$[t\tilde{R}_j^1 \ (1-t)\tilde{R}_j^2]_{j \in \mathcal{K}} \triangleq \left\{ \begin{array}{ll}
\arg \max \mathbf{R} \min_{1 \leq j \leq 2K} R_j^D \\
s.t. \ \mathbf{R}^D \ is \ decodable \ by \ the \ destinations.
\end{array} \right. \number{5.36}$$

Note that the rate allocation \(\number{5.36}\) can be solved using the iterative rate allocation (Algorithm 5.2) with the group partitioning and rate allocation subroutine (the part of Algorithm 5.1) specified in Section 5.3.1.2.

Finally, a solution to the rate allocation is given by $R_j^1 = \min \{ \tilde{R}_j^1, \tilde{R}_j^1 \}$ and $R_j^2 = \tilde{R}_j^2$ for $j \in \mathcal{K}$. The rate of source $j$ is given by $R_j = tR_j^1 + (1-t)R_j^2$. 
5.3.3 Dynamic Relay Assignment

Next we consider joint relay assignment and rate allocation for the inband relay system. We propose a two-step heuristic solution which is a combination of the dynamic relay assignment for hopping relay [c.f. Section 5.2.2] and the rate allocation for inband relay with fixed relay assignment [c.f. Section 5.3.2]. In the first step, for each source we find its possible assisting relay based on the source-relay and relay-destination links. In the second step, assuming that the prefixed relay assignment is obtained from the first step, we perform rate allocation for the sources.

The first step is based on the relay assignment scheme for the hopping relay. We assume no direct source-destination channel, i.e., a hopping relay scenario where \( h_{i,j} = 0 \) for \( i, j \in \mathcal{K} \). We perform the initial assignment step of the coarse assignment in Section 5.2.2 to obtain the sources \( \{S_n\} \) assisted by each relay \( n \in \mathcal{N} \).

In the second step, let the prefixed relay assignment \( c_0(j) \) be the one obtained from the first step, we perform rate allocation for sources using the two-step rate allocation scheme for inband relay proposed in Section 5.3.2.

Remark: For the hopping relay system, the size balance step and the fine adjustment step further increase the minimum rates of the sources. However, for the inband relay system, since the direct source-destination link can compensate for the low rates of the relay-destination links, the relay assignment based on the initial assignment step suffices to provide good performance.

5.4 Numerical Results

We simulate the rate allocation for the proposed group decoder for the relay system with \( K = 6 \) sources and destinations. The node positions for the scenarios of fixed relay assignment and dynamic relay assignment is shown in Fig. 5.1. For fixed relay assignment, we assume that there are 6 relays and the position of relay \( i \) is \( (0, 0.5 \times i) \) for \( 1 \leq i \leq 6 \), where each relay \( i \) is assigned to assist source \( i \). For dynamic relay assignment, we assume that there are \( N = 3 \) relays and the position of relay \( i \) is \( (0, 0.75 + 0.5 \times i) \) for \( 1 \leq i \leq 3 \). The channel gain between two nodes is assumed to be complex Gaussian distributed with the mean zero and the variance \( 1/d^2 \) where \( d \) is the distance between the two nodes. We assume that the first time slot occupies half of the entire transmission period, i.e., \( t = 0.5 \), and consider the SNR values, \( P/\sigma^2 \), from 0dB to 9dB. For each SNR value, 1000 channel
realizations are simulated.

Figure 5.1: The simulated multi-relay assisted interference channels.

5.4.1 System Performance

For the hopping relay system with fixed relay assignment, we plot the minimum rates and sum rates of sources in Figs. 5.2 and 5.3, respectively, for group decoders with the group size \( \mu \) denoted as “GD, \( \mu = i \)” for \( 1 \leq i \leq 3 \), and for comparison the linear MMSE decoder where all interference is treated as noise. It is seen that the group decoder provides significantly larger minimum and sum rates over the linear MMSE decoder, and larger minimum and sum rates can be achieved by increasing the group size \( \mu \).

We compare the performance of the proposed DF scheme with the relay amplified and forward (AF) scheme where the group decoder is employed only at the receivers, denoted as “AF + GD”. Since the relays do not decode the source messages and thus superimpose their signals in the second time slot, for each channel realization we random select a relay to forward the received noised signals using a scaled power of \( KP \). It is seen that, due to the forward of noise, in the simulated SNR range, the proposed DF scheme significantly outperforms the AF scheme for the group sizes \( \mu = 1, 2, \) and 3. Moreover, we tailor the multi-relay AF scheme proposed in [55] for single source-destination pair to the multiple source-destination pair scenario using a time-division mode, where each source transmits in the \( 1/K \) of the total time slot (denoted as “Multi, AF, TDMA”). For fair comparison,
at each time each source transmits using a power $KP$ and each relay forwards using a power $P$. It is seen that the proposed group decoder significantly outperforms the multi-relay AF scheme.

For the hopping relay system with dynamic relay assignment, in Fig. 5.3 we show the performance of the proposed relay assignment scheme for the group decoder with group size $\mu = 2$. The proposed heuristic scheme, denoted as “Proposed heuristic”, achieves about 90% of the performance in terms of the minimum rate compared with the optimal solution obtained from an exhaustive search over all possible relay assignments denoted as “Exhaustive search”. To see the performance obtained from each step, we show the minimum rate of sources if the relay assignment terminates after only performing the initial assignment and the set size balance steps in the coarse assignment, denoted as “Initial assignment” and “Size balance”, respectively. It can be seen that balancing the size of $S_i$ significantly increases the minimum rate of sources in the high SNR region, and the fine adjustment further increases the minimum rate of sources in the low SNR region. We also compare the proposed scheme with the relay AF scheme where the group decoder is employed only at the

![Figure 5.2: Minimum rates of sources for the hopping relay system with fixed relay assignment.](image-url)
destinations, and the time-division multi-relay AF scheme with the same protocol as that for the fixed relay assignment scenario. It is also seen that the proposed heuristic method significantly outperforms the above two AF-based relay schemes.

For the inband relay system with fixed relay assignment, we plot the minimum rates of sources for the group sizes $1 \leq \mu \leq 3$ in Fig. 5.5, denoted as “GD, $\mu = i$, proposed” for $1 \leq i \leq 3$. Increasing the group size provides larger minimum rate of sources. We also compare the proposed relay activation scheme with the optimal solution obtained from exhaustive search for the group sizes $1 \leq \mu \leq 3$, denoted as “GD, $\mu = i$, exhaustive” for $1 \leq i \leq 3$. It is seen that the proposed heuristic solution performs close to the optimal relay activation obtained from exhaustive search. The minimum rate of sources based on the linear MMSE decoder is also plotted denoted as “MMSE decoder”. It is seen that the group decoder significantly outperforms the linear MMSE decoder.

For inband relay system with dynamic relay assignment, we plot the minimum rate of sources of the proposed relay assignment scheme in Fig. 5.6. The proposed dynamic relay assignment is
Figure 5.4: Minimum rates of sources for the hopping relay system with dynamic relay assignment.

denoted as “Initial assignment” since it only adopts the initial assignment step of the relay assignment for hopping relay with dynamic relay assignment. For comparison, we plot the minimum rate based on the relay assignment obtained after performing the size balance and the fine adjustment steps, denoted as “Size balance” and “Fine adjustment”, respectively. It is seen that due to the compensation of the direct source-destination links, further processing beyond the initial assignment is unnecessary. Moreover, the proposed relay assignment scheme performs close to the optimal solution obtained from exhaustive search denoted as “Exhaustive search”. We also plot the minimum rate obtained from the linear MMSE decoder assuming the same relay assignment as that of the “Initial assignment”, where at both relays and destinations all interference is treated as noise. It is seen that again the proposed group decoder significantly outperforms the linear MMSE decoder.
5.4.2 System Complexity Discussion

In the following we discuss the complexity of the proposed scheme using the number of message passing iterations between the sources and the relays, between the sources and the destinations, and between the relays and the destinations.

First we observe the hopping relay system. With fixed relay assignment, one message passing iteration between the sources and the relays and between the relays and the destinations suffices to maximize the minimum rate of the sources. With dynamic relay assignment, typically there are two to three source-relay message passing iterations in the coarse assignment and four relay-destination message passing iterations in the fine adjustment, and thus totally six to seven iterations. In the rate allocation step, we set two message passing iterations for the source-relay channel and the relay destinations channel.

Then we observe the inband relay system. With fixed relay assignment, in the step of identifying the set of sources assisted by relays, we set one message passing iteration for the tentative rate
allocations for both the source-relay channel and the source-destination channel; and in the step of rate allocation, we also set two message passing iterations for the interference channels at both the relays and the destinations. With dynamic relay assignment, in the relay assignment stage, the number of message passing iterations is the sum of that for the hopping relay case and that for the step of identifying the set of sources assisted by relays. In the rate allocation, we also set two message passing iterations for both the source-relay and the relay-destination channels.
Chapter 6

Partial Downloading and Resource Allocation for Wireless Distributed Storage Networks

The purpose of distributed storage is to store data in a distributed manner where the individual storage nodes may be unreliable. Recently, schemes based on regenerate codes have been proposed \cite{56, 57}, which perform data reconstruction and node regeneration. For data reconstruction, a data collector (DC) downloads the symbols in some storage nodes to reconstruct the data. For node regeneration, assuming that a storage node has failed, a new storage node downloads the symbols from some other storage nodes to regenerate the symbols in the failed node. An \((S, K, d, \alpha, \beta)\) regenerating code are described as follows.

1. There are \(S\) storage nodes, where each node stores \(\alpha\) symbols.

2. A DC can reconstruct the data via downloading all the \(K\alpha\) symbols from any \(K\) storage nodes.

3. A new storage node can regenerate the \(\alpha\) symbols in a failed node by downloading \(\beta\) symbols each from any \(d\) surviving nodes; and after that a DC can still reconstruct the data via downloading all the \(K\alpha\) symbols from any \(K\) storage nodes.

An optimal trade-off curve between the amount of downloaded symbols \(K\alpha\) for data reconstruction and \(\gamma = d\beta\) for node regeneration is reported in \cite{56, 57}, with two extremes called the minimum-storage regenerating (MSR) point and the minimum-bandwidth regenerating (MBR) point, cor-
responding to the coding schemes with the best efficiency $K\alpha = M$ for data reconstruction and $d\beta = \alpha$ for node regeneration, respectively. The explicit coding schemes for distributed storage systems are developed in [1],[58],[64]. Up to now, the best results are the coding schemes in [65], which achieve the optimal tradeoff at the MSR ($S \leq 2K$) and MBR (all $S$ and $K$) points using the product-matrix construction.

We consider wireless distributed storage systems where the storage nodes are connected to the DC through orthogonal wireless channels, as shown in Fig. 6.1. Due to the channel fading and the power and bandwidth constraints of the wireless links, the data collector may not be able to download all symbols from a storage node. In other words, when the downloading bandwidth from the DC to all storage nodes is limited, e.g., in a wireless distributed storage system, it could be more efficient to download a small number of symbols from all storage nodes, i.e., partial downloading, than to download all symbols from a few storage nodes, i.e., full downloading, due to the exponential nature of the power consumption as a function of the amount of information downloaded. Note that when designing such a partial downloading scheme for data reconstruction, we should maintain the efficiency of the node regeneration methods as in [65].

For a wireless distributed storage system, given the regeneration code employed and the channel state information of the wireless links, we are interested in designing a scheme for the DC to download symbols from the storage nodes with the minimum total power consumption, such that the whole data can be reconstructed. We will show that in order to formulate the above problem in an analytically tractable way, the key is to obtain a necessary and sufficient condition for reconstructing the original data in terms of the number of symbols downloaded from each storage node.

Specifically, we propose partial downloading schemes that allow downloading a portion of the symbols in a storage node for data reconstruction, at both the MSR and MBR points, while maintaining the node regeneration schemes in [58],[64],[65]. We give necessary and sufficient conditions for data reconstruction for the proposed partial downloading schemes. Such conditions effectively facilitate the solution to a cross-layer wireless resource allocation problem for data reconstruction. We then propose channel and power allocation schemes for data reconstruction in distributed wireless storage systems under partial downloading. We also briefly discuss the wireless resource allocation for node regeneration where a new storage node regenerates the symbols in a failed storage node by downloading symbols from some surviving nodes.
A related work is the flexible coding scheme proposed in [1], which downloads different numbers of symbols from storage nodes to perform data reconstruction and node regeneration. Compared with the explicit coding schemes in [1], the partial downloading scheme proposed in this chapter requires much less number of total downloaded symbols for node regeneration, which will be shown in Section 6.4.3. In particular, for the MBR point, using the proposed partial downloading scheme, the number of downloaded symbols required for data reconstruction is the total number of data symbols, which is the minimum possible.

We use the following notations throughout this chapter. For a family of matrices \( \{H^{(i)}\}_{i \in A} \) with the same number of rows, let \( HA \triangleq \left[H^{(i)} , i \in A \right] \) be the matrix obtained by horizontally concatenating \( H^{(i)} \) for \( i \in A \), e.g., \( HA = \left[H^{(1)} | H^{(2)} \right] \) for \( A = \{1, 2\} \). Let \( H_0 \subseteq H \) denote that \( H_0 \) is a submatrix of \( H \) by extracting columns of \( H \), and \( H_0 \subset H \) denote that \( H_0 \subseteq H \) and \( H_0 \neq H \). For \( H_0 \subseteq H \), let \( H \setminus H_0 \) be the submatrix of \( H \) that includes all columns of \( H \) that are not in \( H_0 \). Let \( \text{span}(H) \) be the space spanned by the columns of \( H \). For two spaces \( Q_0 \) and \( Q \), let \( Q_0 \subseteq Q \) denote that \( Q_0 \) is a subspace of \( Q \), and \( Q_0 \subset Q \) denote that \( Q_0 \subseteq Q \) and \( Q_0 \neq Q \). We have that if \( H_0 \subseteq H \) then \( \text{span}(H_0) \subseteq \text{span}(H) \). Let \( \text{rank}(H) \) be the column rank of the matrix \( H \) which equals to the dimension of space \( \text{span}(H) \).

### 6.1 System Description and Problem Formulation

#### 6.1.1 System Model

We store a data block of \( M \) symbols \( s = [s_1, s_2, ..., s_M]^T \) in a distributed storage system consisting of \( S \) storage nodes denoted as \( S = \{1, 2, ..., S\} \), where each node stores \( \alpha \) symbols. Each symbol is a packet of subsymbols in the field \( GF(q) \), and contains \( B \) bits. Each storage node \( i \) stores a linear combination of the data symbols denoted as \( m^{(i)} = [m_1^{(i)}, ..., m_\alpha^{(i)}] \), given by

\[
m_j^{(i)} = \sum_{k=1}^{M} h_{kj}^{(i)} s_k = s^T h_j^{(i)}, \quad 1 \leq j \leq \alpha, \tag{6.1}
\]

where the coefficients \( h_j^{(i)} = [h_{1j}^{(i)}, h_{2j}^{(i)}, ..., h_{Mj}^{(i)}]^T \in GF(q)^M \) for \( 1 \leq j \leq \alpha \). Denoting the encoding matrix \( H^{(i)} \triangleq [h_1^{(i)}, h_2^{(i)}, ..., h_\alpha^{(i)}] \in GF(q)^{M \times \alpha} \), we then have \( (m^{(i)})^T = s^T H^{(i)} \) for \( i \in S \). The DC downloads symbols from the storage nodes to reconstruct the data. Assume that the encoding matrices \( \{H^{(i)}\}_{i \in S} \) are known by the DC and they are such that the original data can be reconstructed.
via downloading all symbols from any $K$ storage nodes.

Assume that the DC and the storage nodes are connected via $N$ orthogonal wireless channels $\mathcal{N} = \{1, 2, ..., N\}$, where each channel is of bandwidth $W$ and duration $T$. For $i \in \mathcal{S}$ and $j \in \mathcal{N}$, let $g_j^{(i)}$ be the complex gain of channel $j$ from storage node $i$ to the DC, and $P_j$ be the transmission power of channel $j$. The number of symbols that can be transmitted from storage node $i$ to the DC over channel $j$ is then given by

$$c \left( |g_j^{(i)}|^2 P_j \right) = \frac{WT}{B} \log_2 \left( 1 + \kappa \frac{|g_j^{(i)}|^2 P_j}{\sigma^2} \right).$$

(6.2)

where $\sigma^2$ is the power of background noise and $\kappa < 1$ accounts for the rate loss due to the practical modulation and coding, compared with the ideal case of Gaussian signaling and infinite-length code. Assume that the minimum unit of data transmitted over a channel is a symbol, i.e., only one or several symbols can be transmitted over each channel. Our objective is to minimize the total transmission power $\sum_{j \in \mathcal{N}} P_j$ while guaranteeing successful data reconstruction.

6.1.2 Partial Downloading Scheme

We consider the reconstructability of the original data if the DC downloads a portion of the symbols stored in each storage node. For $i \in \mathcal{S}$, let $\mu_i$ be the number of symbols downloaded from storage node $i$. Since $\mu_i \leq \alpha$, we assume that the downloaded symbols are linear combinations of the symbols in node $i$ given by $s^T H^{(i)} A^{(i)}$ where $A^{(i)}$ is an $\alpha \times \mu_i$ matrix. From [1], the data $s$ can be reconstructed from the downloaded symbols $s^T \left[ H^{(i)} A^{(i)} \right]_{i \in \mathcal{S}}$ if and only if

$$\text{rank} \left( \left[ H^{(i)} A^{(i)} \right]_{i \in \mathcal{S}} \right) = M.$$

(6.3)

For each $i \in \mathcal{S}$ the matrix $A^{(i)}$ is assumed to be of full column rank, since otherwise at least one downloaded symbol can be expressed as a linear combination of other symbols downloaded from the same storage node, which means this symbol is redundant and should be removed to reduce the downloading bandwidth.

However, the search for the linear combination matrices $\{A^{(i)}\}_{i \in \mathcal{S}}$ that satisfy (6.3) can be computationally prohibitive. A simpler scheme is to directly download the stored symbols from each storage node, without performing linear combination. An immediate question is whether such a simpler approach may lose optimality, i.e., is it possible that for some $\{\mu_i\}_{i \in \mathcal{S}}$, (6.3) can
be satisfied by performing linear combination but it can not be satisfied by simply downloading the stored symbols without linear combination. The following result states that in terms of the numbers of symbols downloaded from the storage nodes, downloading the symbols stored in the storage nodes directly and downloading their linear combinations are equivalent. The proof is given in Appendix A.

**Theorem 6.1:** If there exist $\alpha \times \mu_i$ matrices $A^{(i)}$ for $i \in S$ such that (6.3) is satisfied, then there exist $M \times \mu_i$ submatrices $\bar{H}^{(i)} \subseteq H^{(i)}$, $i \in S$, such that $\bar{H}^S = \left[ \bar{H}^{(i)} \right]_{i\in S}$ is of rank $M$.

Based on Theorem 6.1, we provide the following definition of data reconstructability.

**Definition 1 (µ-reconstruction):** Given $\mu = \left[ \mu_1, \mu_2, \ldots, \mu_S \right]$, $\mu_i \in \{0,1,\ldots,\alpha\}$ for $1 \leq i \leq S$, the data is $\mu$-reconstructable if it can be reconstructed via downloading $\mu_i$ symbols from storage node $i$ for $i \in S$, which is equivalent to that there exist $M \times \mu_i$ submatrices $\bar{H}^{(i)} \subseteq H^{(i)}$, $i \in S$, such that the matrix $\bar{H}^S = \left[ \bar{H}^{(i)} \right]_{i\in S}$ is of rank $M$.

Note that $\mu$-reconstructability implies that $\mu_i \in \{0,1,\ldots,\alpha\}$ for $1 \leq i \leq S$.

### 6.1.3 Wireless Resource Allocation for Data Reconstruction

Given a coding scheme employed by the distributed storage system, in the following we describe the wireless setting and formulate a wireless resource allocation problem for downloading symbols from the storage nodes by the DC to reconstruct the data.

#### 6.1.3.1 Wireless Setting

The basic structure of the wireless distributed storage system is shown in Fig. 6.1. There are a command channel and a data channel from each storage node to the DC, and a feedback channel from the DC to each storage node. Each storage node $i$ knows only its own data symbols $m^{(i)}$ and linear combination coefficients $H^{(i)}$. Before downloading, each storage node $i$ sends its $H^{(i)}$ to the DC through the command channel, and some pilot symbols for channel estimation to the DC through the data channel. The DC then estimates the channel gains $\{g^{(i)}_j\}_{j \in \mathcal{N}}$ of the wireless link that connects itself to each storage node $i$. It then performs a wireless resource allocation to decide which symbols to download from each storage node, and the corresponding channels and powers for downloading these symbols; the results are feedbacked to the storage nodes through the feedback channel. Finally each storage node sends its chosen symbols to the DC through the data.
channel according to the resource allocation results.

Figure 6.1: Illustration of wireless distributed storage system.

6.1.3.2 Wireless Resource Allocation

For \( i \in S \) and \( j \in N \), let \( \beta_j^{(i)} = 1 \) if the DC downloads symbols from storage node \( i \) using channel \( j \) and \( \beta_j^{(i)} = 0 \) otherwise. For \( j \in N \), let \( P_j \) be the transmission power for channel \( j \). Then the number of symbols downloaded over channel \( j \) is given by

\[
X_j = c \left( P_j \sum_{i \in S} \beta_j^{(i)} |g_j^{(i)}|^2 \right),
\]

(6.4)
where \( c(\cdot) \) is given in (6.2). The number of symbols downloaded from storage node \( i \) is then,

\[
\mu_i = \sum_{j=1}^{N} \beta_{j}^{(i)} X_j, \; i \in S.
\]  

(6.5)

The wireless resource allocation problem is to minimize the total power across all \( N \) channels that guarantees data reconstruction, which can be formulated as follows,

\[
\min_{\{\beta_{j}^{(i)}, P_j\} \in S, j \in N} \sum_{j \in N} P_j
\]

s.t. the data is \( \mu \)-reconstructable;

(6.4) and (6.5), \( X_j \in \{0, 1, 2, \ldots, \alpha\} \) for \( j \in N \);

\[
\sum_{i \in S} \beta_{j}^{(i)} \leq 1, \; j \in N; \; \beta_{j}^{(i)} \in \{0, 1\}.
\]  

(6.6)

The last constraint dictates that each channel, if used, can only transmit symbols from one node. The constraint that the data is \( \mu \)-reconstructable significantly complicates the resource allocation problem because the variables \( \{P_j, \beta_{j}^{(i)}\} \) are constrained through (6.5) and the full rank condition in Definition 1, which makes the problem seemingly intractable. In the next section we provide the necessary and sufficient conditions for the \( \mu \)-reconstructability for the coding schemes at the MSR and MBR points in terms of simple linear constraints on \( \{\mu_i\} \). This will significantly simplify the above wireless resource allocation problem for data reconstruction in distributed storage systems.

The network equivalence theory [66] states that the demand of a network with noisy, independent, and memoryless links can be met if and only if it can be met on another network where each noisy link is replaced by another noiseless one with the same capacity. This means that the codes sent over the wireless channel can be independent of the codes used for data storage, which justifies the problem formulation using the noiseless channel capacity given by (6.4).

### 6.2 Reconstructability for Partial Downloading

In this section, we analyze the \( \mu \)-reconstructability for both the MSR point and the MBR point. In particular, for the MSR point, we prove a necessary and sufficient condition for the \( \mu \)-reconstructability for any MSR coding scheme. On the other hand, for the MBR point, we consider a specific MBR coding scheme and give the necessary and sufficient condition for the \( \mu \)-reconstructability for this particular scheme.
To analyze the $\mu$-reconstructability, we consider two problems. First, we determine the necessary and sufficient condition on $\{\mu_i\}_{i \in S}$ so that the DC can reconstruct the data via downloading $\mu_i$ symbols from storage node $i, i \in S$. Secondly, we provide a practical partial downloading scheme given a set $\{\mu_i\}_{i \in S}$ that satisfies this condition.

### 6.2.1 $\mu$-Reconstructability for MSR Point

We consider the $\mu$-reconstructability for all coding schemes satisfying the constraint of the MSR point that $K\alpha = M$. Since for any size-$K$ subset $R \subseteq S$, the data $s$ can be reconstructed from the $M$ downloaded symbols $s^T[H(i)]_{i \in R}$, it follows that the square matrix $[H(i)]_{i \in R}$ is of rank $M$. Thus for any subset $R \subseteq S$ with $|R| \leq K$, the columns of matrix $[H(i)]_{i \in R}$ are linearly independent.

Based on the above observation, we show that the trivial necessary condition $\sum_{i \in S} \mu_i \geq M$, i.e., that the number of downloaded symbols should be no less than the total number of data symbols to be reconstructed, is also a sufficient condition for $\mu$-reconstructability.

**Theorem 6.2:** For the MSR point the necessary and sufficient condition for the $\mu$-reconstructability is $\sum_{i \in S} \mu_i \geq M$.

**Proof:** We explicitly select $\mu_i$ symbols $s^T[H(i)]$ from storage node $i$ for any $\{\mu_i\}_{i \in S}$ satisfying $\sum_{i \in S} \mu_i = M$, where $H(i)$ is an $M \times \mu_i$ submatrix of $H(i), i \in S$, such that $H^S = [H(i)]_{i \in S}$ is of rank $M$. We first outline the basic idea of the proposed explicit construction. We keep selecting symbols that are linearly independent of the selected symbols, until $M$ symbols have been selected.

In each step, we say a storage node $i$ is feasible if the number of downloaded symbols from it is smaller than $\mu_i$. If all symbols in the feasible storage nodes are linearly dependent of the selected symbols, we pick a symbol from a feasible storage node, write it as a linear combination of the selected symbols. Then, based on the linear combination we replace a selected symbol by another symbol from the same storage node, such that the symbol from that feasible storage node can be selected to further increase the rank of $H^S$.

We initialize $H(i)$ as null matrices for $i \in S$. In the selection procedure, let $\lambda_i$ be the number of columns of $H(i)$, i.e., the number of symbols already selected from storage node $i$. Let $V = \{i \in S : \lambda_i < \mu_i\}$ be the set of storage nodes which do not satisfy the downloading requirement, $V^c = \{i : \lambda_i = \mu_i\}$, and $H(i) = H(i) \setminus H(i)$ for $i \in S$. We keep extracting a column of $H(i) = H(i) \setminus H(i)$
and adding it to $\bar{H}^{(i)}$ for some $i \in \mathcal{V} \triangleq \{i \in \mathcal{S} : \lambda_i < \mu_i\}$, until $\bar{H}^S = [\bar{H}^{(i)}]_{i \in S}$ is of rank $M$.

When $\text{rank} (\bar{H}^{(S)}) < M$, we check each column of $\bar{H}^\mathcal{V} = [\bar{H}^{(i)}]_{i \in \mathcal{V}}$ to see whether it is linear independent of the columns of $\bar{H}^S$ and if it is, we simply add it to $\bar{H}^\mathcal{V}$ and thus increase the rank by one; otherwise we perform the following procedure.

We randomly select some $i_0 \in \mathcal{V}$ and a column $h_{j_0}^{(i_0)}$ of $\bar{H}^{(i_0)}$, which can then be expressed as a linear combination of the columns in $\bar{H}^S$ as follows

$$h_{j_0}^{(i_0)} = \sum_{i \in \mathcal{W}} \sum_{j \in \mathcal{J}_i} \gamma_{ij}^{(i)} h_j^{(i)}, \quad \gamma_{ij}^{(i)} \neq 0,$$

where $\mathcal{W}$ is the set of storage nodes, and $\mathcal{J}_i \subseteq \{1, 2, ..., \alpha\}$, for each $i \in \mathcal{W}$, is the set of column indices of $\bar{H}^{(i)}$, that participate in the linear combination representation of $h_{j_0}^{(i_0)}$ in (6.7).

Let $\mathcal{W}_0 = \mathcal{W} \setminus \{i_0\}$ be the set of set of storage nodes other than $i_0$ that participates in the linear combination (6.7). Next we show that there exists a column $h_{j_0}^{(i_1)}$ of $\bar{H}^{(i_1)}$ that is linearly independent of the columns of $\bar{H}^S$. Recall the property of the MSR point that the columns of $\bar{H}^\mathcal{R}$ are linearly independent for any $|\mathcal{R}| \leq K$; and $\text{rank}(\bar{H}^\mathcal{R}) = M$, hence $\text{span}(\bar{H}^\mathcal{R}) = Q = GF(q)^M$ for any $|\mathcal{R}| \geq K$.

Since the linear combination (6.7) involves the columns from matrices $\bar{H}^{(i)}$ for $i \in \mathcal{W} \cup \{i_0\}$, we have $|\mathcal{W} \cup \{i_0\}| \geq K + 1$ and $|\mathcal{W}_0| = |\mathcal{W} \setminus \{i_0\}| \geq K$, and thus

$$Q = \text{span}(\bar{H}^{\mathcal{W}_0}) \subseteq \text{span}(\bar{H}^{S \setminus \mathcal{W}_0} | \bar{H}^{\mathcal{W}_0})$$

$$= \text{span}(\bar{H}^{S} | \bar{H}^{\mathcal{W}_0}).$$

Hence $\text{rank}(\bar{H}^{S} | \bar{H}^{\mathcal{W}_0}) = M$. Since by assumption that $\text{rank}(\bar{H}^{S}) < M$, it then follows that there exist a column $h_{j_1}^{(i_1)}$ of $\bar{H}^{\mathcal{W}_0}$ that is linear independent from the columns of $\bar{H}^S$.

Note that from (6.7) replacing a column $h_j^{(i)}$ for a $j \in \mathcal{J}_{i_1}$ with $h_{j_0}^{(i_0)}$ does not change $\text{span}(\bar{H}^S)$ but provides space for $h_{j_1}^{(i_1)}$, which increases the rank of $\bar{H}^S$ by one. We remove $h_{j_0}^{(i_0)}$ from $\bar{H}^S$, then add $h_{j_0}^{(i_0)}$ and $h_{j_1}^{(i_1)}$ to $\bar{H}^S$, increasing the rank of $\bar{H}^S$ by one. $\square$

One implementation issue related to the above symbol selection scheme is how to efficiently test whether an new column vector is linear independent of $\bar{H}^S$. In Appendix B we outline a recursive algorithm for efficient symbol selection for partial downloading.
6.2.2 A Partial Downloading Scheme for the MBR Point

To the best of our knowledge, up to now the best and also practical coding scheme for the MBR point is proposed in [65]. We next analyze the \(\mu\)-reconstructability of this coding scheme.

First we briefly outline the coding scheme given in [65]. Assume that a data block of \(M = \frac{K(K+1)}{2} + K(d-K)\) symbols is stored among \(S\) storage nodes, where each node stores \(\alpha = d\) symbols. The \(M\) symbols are represented using the following \(d \times d\) symmetric matrix

\[
B = \begin{bmatrix}
B^{(1)} & B^{(2)} \\
(B^{(2)})^T & 0
\end{bmatrix}
\]  

(6.10)

where \(B^{(1)}\) is a \(K \times K\) symmetric matrix storing \(K(K+1)/2\) symbols and \(B^{(2)}\) is a \(K \times (d-K)\) matrix storing \(K(d-K)\) symbols. For encoding, the matrix \(B\) is pre-multiplied by an \(S \times d\) Vandermonde matrix given by \(\Psi\), and each node \(i \in S\) stores the \(d\) symbols corresponding to the \(i^{th}\) row of \(\psi_i B\), where \(\psi_i\) is the \(i^{th}\) row of \(\Psi\). It is shown in [65] that the data can be reconstructed by downloading all symbols stored in any \(K\) storage nodes, and by downloading one symbol from each of any \(d\) surviving storage nodes the new storage node can regenerate the same symbols as those in a failed node.

Next we propose a partial downloading scheme for data reconstruction based on the above coding scheme. Since the matrix \(B^{(1)}\) is symmetric, from (6.10) we only need to decode the data symbols in \(B^{(2)}\) and in the upper triangular part of \(B^{(1)}\). As shown in Fig. 6.2, we divide the data symbol into \(d\) columns \(\{b_j\}_{1 \leq j \leq d}\), where \(b_j\) for \(1 \leq j \leq K\) are in the upper triangular part of \(B^{(1)}\), and \(b_j\) for \(K+1 \leq j \leq d\) are the columns of \(B^{(2)}\). We reconstruct the data in the backward order of \(b_d, b_{d-1}, \ldots, b_1\). Note that the number of symbols in \(b_j, 1 \leq j \leq d\), is given by \(\min\{j, K\} \triangleq \theta_j\). Let \((\psi_i B)_j\) be the \(j^{th}\) symbol in storage node \(i\) which is the product of \(\psi_i\) and the \(j^{th}\) column of \(B\). We now present the proposed partial downloading scheme to perform the above backward reconstruction.

1. First we reconstruct \(B^{(2)}\). For \(K+1 \leq j \leq d\), from Fig. 6.2 it is seen that to reconstruct \(b_j\) the DC needs to download the symbols \((\psi_i B)_j\) for \(i\) belonging to a subset \(R_j \subseteq S\), where
the size $|\mathcal{R}_j| \geq K = \theta_j$. Since $\Psi$ is a Vandermonde matrix, with $K$ downloaded symbols $b_j$ can be reconstructed.

2. Then we reconstruct $B^{(1)}$ via reconstructing $b_j$ for $1 \leq j \leq K$ in the order of $b_K, b_{K-1}, \ldots, b_1$. As shown in Fig. 6.2 when reconstructing $b_j$, since $b_l$ for $j < l \leq K$ have been reconstructed and $B^{(1)}$ is symmetric, the part of $B^{(1)}$ in the square shadow is known. Then, as $B^{(2)}$ has been reconstructed and thus is known, reconstructing $b_j$ amounts to downloading the symbols $(\psi_i B)_j$ for $i \in \mathcal{R}_j \subseteq S$, where the size $|\mathcal{R}_j| \geq j = \theta_j$. Again, since $\Psi$ is a Vandermonde matrix, with $j$ downloaded symbols $b_j$ can be reconstructed.

Based on the above partial downloading scheme, let $\eta_j^{(i)} = 1$ if the DC downloads $(\psi_i B)_j$ to reconstruct $b_j$ and $\eta_j^{(i)} = 0$ otherwise. The minimum requirement for data reconstruction is that

$$\sum_{i=1}^{S} \eta_j^{(i)} = \theta_j, \quad 1 \leq j \leq d. \quad (6.11)$$

The data is $\mu$-reconstructable if there exists $\eta_j^{(i)} \in \{0, 1\}$ for $1 \leq i \leq S$ and $1 \leq j \leq d$ such that

$$\mu_i = \sum_{j=1}^{d} \eta_j^{(i)}, \quad i \in S, \quad (6.12)$$

and (6.11) is satisfied. In the following subsection we provide a necessary and sufficient condition in terms of $\{\mu_i\}_{i \in S}$ for (6.11) and (6.12) to hold.
6.2.3 \( \mu \)-reconstructability for MBR Point

A straightforward necessary condition for (6.11) and (6.12) is that, for any subset \( A \subseteq S \), we have

\[
\sum_{i \in A} \mu_i = \sum_{i \in A} \sum_{j=1}^{d} \eta_j(i) = \sum_{j=1}^{d} \sum_{i \in A} \eta_j(i) \leq \sum_{j=1}^{d} \min\{\theta_j, |A|\},
\]

(6.13)

since \( \sum_{i \in A} \eta_j(i) \leq |A| \) and \( \sum_{i \in A} \eta_j(i) \leq \sum_{i \in S} \eta_j(i) = \theta_j \) for all \( 1 \leq j \leq d \). Denote the sorted \( \{\mu_i\}_{i \in S} \) in decreasing order as \( \mu^{(1)} \geq \mu^{(2)} \geq \ldots \geq \mu^{(S)} \), then we have

\[
\sum_{i=1}^{l} \mu^{(i)} \leq \sum_{j=1}^{d} \min\{\theta_j, l\}, \text{ for } 1 \leq l \leq d;
\]

and

\[
\sum_{i=1}^{S} \mu^{(i)} = \sum_{j=1}^{d} \theta_j.
\]

(6.14)

Since \( \theta_j = \min\{j, K\} \), (6.14) becomes

\[
\sum_{i=1}^{l} \mu^{(i)} \leq dl - \frac{l(l-1)}{2}, \text{ for } 1 \leq l \leq d;
\]

and

\[
\sum_{i=1}^{S} \mu^{(i)} = M.
\]

(6.15)

The following result shows that (6.14) or (6.15) is also sufficient for data reconstruction.

**Theorem 6.3:** Denote the sorted \( \{\mu_i\}_{i \in S} \) by \( \mu^{(1)} \geq \mu^{(2)} \geq \ldots \geq \mu^{(S)} \). Then (6.15) is a necessary and sufficient condition for the \( \mu \)-reconstructability of the proposed partial downloading scheme for the MBR point.

**Symbol Selection Scheme**

We prove the sufficiency via an explicit partial downloading scheme. More specifically, given \( \{\mu_i\}_{i \in S} \) satisfying (6.15), we compute \( \{\eta_j(i)\}_{i \in S, 1 \leq j \leq d} \) as follows.

- To initialize, we rank \( \{\mu_i\}_{i \in S} \) in decreasing order \( \mu_{i_1} \geq \mu_{i_2} \geq \ldots \geq \mu_{i_S} \), and let \( \theta_j = \min\{j, K\} \), \( j = 1, \ldots, K \).

- Then for \( k = 1, 2, \ldots \) to \( S \), we compute \( \{\eta_j^{(i_k)}\}_{1 \leq j \leq d} \) as follows.
  - Rank \( \{\theta_j\}_{1 \leq j \leq d} \) in decreasing order \( \theta_{j_1} \geq \theta_{j_2} \ldots \geq \theta_{j_d} \);
  - Let \( \eta_j^{(i_k)} = 1 \) for \( 1 \leq p \leq \mu_{i_k} \) and \( \eta_j^{(i_k)} = 0 \) for \( \mu_{i_k} + 1 \leq p \leq d \).
– Subtracting \( \{ \eta_{j_k}^{(i_k)} \}_{1 \leq j \leq d} \) from \( \theta_{jp} \), we update \( \theta_{jp} = \theta_{jp} - 1 \) for \( 1 \leq p \leq \mu_{ik} \).

In Appendix C we prove that if (6.14) holds, then using the above symbol selection scheme we can obtain \( \eta_{j}^{(i)} \), \( i \in \mathcal{S} \), \( 1 \leq j \leq d \), to reconstruct the data.

We remark that due to Theorem 6.1, condition (6.15) cannot be relaxed by allowing downloading the linear combinations of the symbols in the storage nodes.

Note that in the full downloading scheme proposed in [65], \( dK \) symbols have to be downloaded from \( K \) storage nodes for data reconstruction; whereas the above proposed partial downloading scheme downloads a total of \( M \) symbols, which is the minimum amount for reconstructing the original \( M \) data symbols. Since \( M = dK - K(K-1)/2 < dK \), the proposed partial downloading scheme can significantly reduce the bandwidth requirement for symbol downloading. For example, for \( d = 6 \) and \( K = 4 \), we have \( dK = 24 \) but \( M = 18 \), i.e., the number of downloaded symbols by the proposed partial downloading scheme is 75% of that by the scheme in [65].

Consider further the constraints in (6.15), which read \( \mu^{(1)} \leq d \), \( \mu^{(1)} + \mu^{(2)} \leq 2d - 1 \), and \( \mu^{(1)} + \mu^{(2)} + \mu^{(3)} \leq 3d - 3 \), etc. That is, downloading all symbols from any two storage nodes, or more than \( 3d - 3 \) symbols from any three storage nodes, is not allowed. Hence these constraints impose that the DC download evenly from all nodes. Such a requirement matches well with the characteristic of wireless channels. In particular, in general the optimal resource allocation in wireless systems tends to distribute the resource uniformly among the nodes, rather than giving the resources to only a few nodes and starving the other nodes.

### 6.3 Channel and Power Allocation for Wireless Partial Downloading

In this section we provide solutions to the channel and power allocation problem (6.6) for the MSR and MBR points based on the data reconstructability condition obtained in Theorems 6.2 and 6.3. First we drop the constraint \( \mu_i \leq \alpha \), \( i \in \mathcal{S} \) for the MSR point, and (6.15) for the MBR point, and minimize the sum power only with the constraint \( \sum_{j \in \mathcal{N}} X_j = \sum_{i \in \mathcal{S}} \mu_i = M \). Then we check whether the dropped original constraints are violated, and perform local adjustments if so. Otherwise, the solution to the relaxed problem obtained in the first step is also the optimal solution to the original one (i.e., problem (6.6)); and simulation results show that this is indeed the case.
most of the time.

Based on the channel estimation results, the DC runs the channel and power allocation to determine the number of symbols downloaded from each storage node. After that, according to the symbol selection schemes proposed for the MSR and MBR points, the DC determines which symbols to be downloaded from each storage node and notifies the storage nodes using the feedback channel. The storage nodes then send the data symbols to the data collector upon request.

### 6.3.1 Optimal Channel and Power Allocation for the Relaxed Problem

According to \((6.4)\), letting \(p(\cdot)\) be the inverse function of the capacity function \(c(\cdot)\) in \((6.2)\), we have

\[
P_j = \frac{p(X_j)}{\sum_{i \in S} \beta^{(i)}_j |g^{(i)}_j|^2}.
\]  

(6.16)

The relaxed version of problem \((6.6)\) is given by

\[
\min_{\{\beta^{(i)}, X_j\}_{i \in S, j \in N}} \sum_{j \in N} \frac{p(X_j)}{\sum_{i \in S} \beta^{(i)}_j |g^{(i)}_j|^2}
\]

s.t. \(\sum_{j = 1}^N X_j = M\); \(X_j \in \mathbb{Z}^+ \cup \{0\}\);

\[
\sum_{i \in S} \beta^{(i)}_j \leq 1, \ j \in N; \ \beta^{(i)}_j \in \{0, 1\}.
\]

(6.17)

Then the optimal strategy for channel assignment is to assign each channel to the node with the strongest channel, i.e., for each channel \(j \in N\),

\[
\beta^{(i)}_j = 1, \ \text{for } i_j = \arg \max |g^{(i)}_{ij}|; \ \text{and } \beta^{(i)}_j = 0, \ \text{otherwise}.
\]

(6.18)

Letting \(p_j(X_j) = \frac{1}{|g^{(i)}_{ij}|^2} p(X_j)\), the power allocation problem then becomes

\[
\min_{\{X_i\}_{i \in N}} \sum_{j = 1}^N p_j(X_j)
\]

s.t. \(\sum_{j = 1}^N X_j = M; \ X_j \in \mathbb{Z}^+ \cup \{0\}\).

(6.19)

We use the following greedy algorithm to optimally solve \((6.19)\).

- Initialize \(X_j = 0\) for \(j \in N\).
• While $\sum_{j \in \mathcal{N}} X_j < M$ do
  
  – For $j \in \mathcal{N}$, let $\Delta P_j = p_j(X_j + 1) - p_j(X_j)$ be the power increment for channel $j$.
  
  – Find the channel $j_0 = \arg \min_{j \in \mathcal{N}} \Delta P_j$ with the minimum power increment, and update $X_{j_0} \leftarrow X_{j_0} + 1$.

• Output $X_j$ for $j \in \mathcal{N}$.

We have the following result on the optimality of the above greedy approach to solving the relaxed problem in (6.19). The proof is given in Appendix D.

**Theorem 6.4:** The above greedy algorithm provides an optimal solution to the power allocation problem (6.19).

### 6.3.2 Local Adjustment

Based on the solution $\{X_j\}_{j \in \mathcal{N}}$ obtained by the greedy algorithm, we compute the number of symbols assigned to each storage node, given by $\mu_i = \sum_{j \in \mathcal{N}} \beta_j^{(i)} X_j$ for $i \in \mathcal{S}$. If $\{\mu_i\}_{i \in \mathcal{S}}$ violates the constraint that $\mu_i \leq \alpha, i \in \mathcal{S}$ for the MSR point, or (6.15) for the MBR point, we perform the following local adjustment.

#### 6.3.2.1 Local Adjustment for the MSR Point

The basic idea is to find a storage node $i$ with $\mu_i > \alpha$, and reassign one of its subcarrier to another storage node to decrease $\mu_i$, until $\mu_i \leq \alpha$ for all $i \in \mathcal{S}$. The local adjustment scheme is given as follows.

• While $\mu_i > \alpha$ for some $i \in \mathcal{S}$, do
  
  – Find a storage node $i$ with $\mu_i > \alpha$ and the set of assigned channel denoted as $\mathcal{N}_i = \{j : \beta_j^{(i)} = 1\}$;
  
  – Find the storage node $i' \in \mathcal{S} \setminus \{i\}$ and the channel $j \in \mathcal{N}_i$ that minimizes the power increment of reassigning the $X_j$ symbols in channel $j$ to node $i'$ such that $\mu_i' + X_j \leq \alpha$,
\\(i, j) = \arg \min_{(i', j) \in S' \times N: \mu_i' + X_j \leq \alpha} p^{(i')}_{j}(X_j) - p_{j}^{(i)}(X_j).\)

(6.20)

Reassign the \(X_{j_0}\) symbols in channel \(j_0\) to storage node \(i_0\), by letting \(\beta_{i_0}^{j_0} = 1\) and \(\beta_{j_0}^{(i)} = 0\).

6.3.2.2 Local Adjustment for the MBR Point

The local adjustment for the MBR point follows the same procedure as that for the MSR point. The only difference is that when selecting the storage node-channel pair in (6.20), we re-assign one symbol from the storage node with the maximum \(\mu_i\) to the storage node with the minimum \(\mu_i\), while minimizing the power increment of reassigning the symbol.

6.3.3 Resource Allocation for Node Regeneration

According to the coding scheme in [65], for the MSR and MBR points, the data can be regenerated via downloading one symbol from each of any \(\alpha + K - 1\) and \(\alpha\) storage nodes, respectively. Based on this explicit and practical coding scheme we can formulate the wireless resource allocation problem for node regeneration, which is similar to that for the data reconstruction based on partial downloading.

According to the node regeneration scheme proposed in [65], we have

- for the MSR point, as long as \(\alpha \geq K - 1\), the data in a failed storage node can be exactly regenerated via choosing any \(d = \alpha + K - 1\) storage nodes and downloading one symbol from each of the \(d\) nodes;

- for the MBR point, the data in a failed storage node can be exactly regenerated via selecting any \(d = \alpha\) nodes and downloading one symbol from each of them.

Then, the wireless resource allocation for both the MSR and MBR points can be similarly formulated as before, i.e., to minimize the power consumption of downloading \(d\) symbols from \(d\)
storage nodes. The power and channel allocation problem is similar to (6.17), which is to minimize the total power subject to that \( \sum_{i \in S} \mu_i = d \) and \( \mu_i \in \{0, 1\} \) for \( i \in S \). Thus, the problem can be solved similarly as that of the data reconstruction for the MSR point discussed in Section 6.3.

6.4 Numerical Results for Resource Allocation

We consider a wireless distributed storage system with \( S = 16 \) storage nodes communicating using \( N = 64 \) orthogonal channels with the noise power \( \sigma^2 = 0.25 \). Assume that in (6.2) the coefficients \( \frac{W^T B}{B^T} = 0.25 \) and \( \kappa = 0.5 \). Further assume that the distance between the DC and any storage node is 0.5; and the distance between the new storage node and any surviving storage nodes is also 0.5. The channel gain between the two nodes of distance \( d \) is modeled by a complex Gaussian random variable \( N_C(0, d^{-2}) \).

6.4.1 Wireless Resource Allocation for Data Reconstruction

6.4.1.1 MSR Point

Assume that each storage node stores \( \alpha = 4 \) symbols and the data can be reconstructed via downloading all symbols from \( K = 4 \) storage nodes. Thus the number of data symbols \( M = \alpha K = 16 \). Fig. 6.3 shows the sum power of the proposed resource allocation scheme in Section 6.3 for 100 channel realizations, where for 95 of them the proposed scheme (denoted as “proposed scheme”) attains the performance upper bound (denoted as “upper bound”), i.e., the optimal solution to the relaxed problem (c.f. Section 6.3.1), and thus is optimal. We further compare the proposed partial downloading scheme with the full downloading scheme (denoted as “full downloading”), where using exhaustive search the DC optimally selects 4 storage nodes and downloads all symbols from them. It is seen that the proposed partial downloading consumes less power than the full downloading scheme.

Fig. 6.4 shows the histogram of the power gap (in dB) between the proposed resource allocation scheme and the solution obtained from the first step, i.e., the upper bound, for 1000 channel realizations, which is an upper bound of the gap between the proposed scheme and the optimal solution. It is seen that for more than 950 channel realizations (95%), the proposed method attains the optimal solution. Moreover, Fig. 6.5 shows the histogram of the power gain (in dB) of the proposed partial downloading scheme over the optimal full downloading scheme from exhaustive
search. It is seen that for around 900 channel realizations (90%) the proposed scheme exhibits 0.6 to 1.2 dB performance gain over the full downloading scheme.

![Figure 6.3: Total power consumption under partial downloading and full downloading for data reconstruction at the MSR point.](image)

### 6.4.1.2 MBR Point

Assume that each storage node stores $d = \alpha = 6$ symbols and the data can be reconstructed via downloading all symbols from $K = 4$ storage nodes. According to (6.9), the number of data symbols $M = 18$.

Fig. 6.6 shows the sum power of the proposed wireless resource allocation scheme in Section 6.3 for 100 channel realizations. The proposed scheme (denoted as “proposed scheme”) attains the performance upper bound denoted as “upper bound”, i.e., the optimal solution to the relaxed problem (c.f. Section 6.3.1), and thus is optimal. We further compare the proposed partial downloading scheme with the full downloading scheme (denoted as “full downloading”), where again using exhaustive search the DC optimally selects 4 storage nodes and downloads all symbols from them.
The performance improvement of the proposed partial downloading scheme over the full downloading is even larger than that of the MSR point. This is because for the MBR point the total number of downloaded symbols for partial downloading is only 18 while that of the full downloading scheme is 24. We also test another downloading scheme where the DC downloads $M = 18$ symbols from $K = 4$ storage nodes where the numbers of downloaded symbols satisfy constraint (6.15), denoted as “$K$ nodes”, where similar exhaustive search method is employed to select the $K = 4$ storage nodes as that for the MSR point. It is seen that the proposed partial downloading scheme consumes less power than the above scheme, where all the downloaded symbols are constrained within $K = 4$ nodes.

We have simulated 1000 channel realizations for data reconstruction of the MBR point. For all simulated channel realizations, the proposed scheme shows zero gap to that obtained from the first step, and thus is optimal. Furthermore, in Fig. 6.7 we show the histogram of the sum power consumption gap between the proposed method and the upper bound for the MSR point.
gain (in dB) of the proposed partial downloading scheme over the full downloading scheme from exhaustive search. It is seen that for most channel realizations the proposed scheme exhibits 2.2 to 3.0 dB power saving over the full downloading scheme. Similarly, in Fig. 6.8 we show the histogram of the sum power gain (in dB) of the proposed partial downloading scheme over the downloading scheme where the DC downloads $M = 18$ symbols from $K = 4$ nodes. It is seen that for around 900 channel realizations (90%) the proposed scheme exhibits 0.4 to 1.2 dB performance gain over the full downloading scheme, which is similar to that for the MSR point.

### 6.4.2 Comparison Between MSR and MBR Points

We compare the number of downloaded symbols for data reconstruction and node regeneration for the MSR and MBR points. At both points, using on the proposed partial downloading schemes the DC needs to download $M$ symbols for data reconstruction. Since the constraint (6.15) matches well with the characteristics of wireless channels, for both MSR and MBR points the requirements

![Figure 6.5: Histogram of the power saving of the proposed partial downloading scheme over the full downloading scheme for the MSR point.](image-url)
for data reconstruction are virtually the same, which is that \( \sum_{i=1}^{S} \mu_i \geq M \), i.e, the total number of downloaded symbols is larger than or equal to the number of data symbols. However, for node regeneration, the number of downloaded symbols is \( \alpha + K - 1 \) for the MSR point but \( \alpha \) for the MBR point, which shows that MBR point is more efficient. Therefore, when employing partial downloading schemes the MBR point is preferred because the node regeneration is more efficient.

### 6.4.3 Comparison with the Flexible Coding in [1]

We compare the node regeneration bandwidth of the proposed partial downloading scheme with the flexible downloading schemes in [1]. For the method in [1], successful data reconstruction is guaranteed for any choice of \( \{\mu_i\}_{i \in S} \) if \( \sum_{i \in S} \mu_i \geq M \); and the data of a storage node \( \ell \) can be successfully regenerated as long as the number of downloaded symbols from storage node \( i \), denoted as \( \beta_i, i \neq \ell \), satisfies

\[
\sum_{i \in S, i \neq \ell} \beta_i \geq \gamma, \quad 0 \leq \beta_i \leq \beta_{\text{max}}, \tag{6.21}
\]

---

Figure 6.6: Total power consumption under partial downloading and full downloading for data reconstruction at the MBR point.
CHAPTER 6. PARTIAL DOWNLOADING AND RESOURCE ALLOCATION FOR WIRELESS DISTRIBUTED STORAGE NETWORKS

139

2.2 2.4 2.6 2.8 3 3.2 3.4 3.6
0 50 100 150 200 250 300
Power Consumption Saving (dB)
Number of Events

Figure 6.7: Histogram of the power saving of the proposed partial downloading scheme over the full downloading scheme for the MBR point.

where \( \beta_{\text{max}} \) is the upper bound on the number of downloaded symbols from the surviving storage node.

6.4.3.1 Achievable Bound

An achievable bound for the repair bandwidth \( \gamma \) proposed in [1] is given as

\[
\gamma \geq \max\{\alpha - \beta_{\text{max}}, M \mod \alpha\} + \left\lfloor \frac{M}{\alpha} \right\rfloor \beta_{\text{max}}. \tag{6.22}
\]

In our work, for data regeneration we set the maximum number of downloaded symbol from each storage node to be one, i.e., \( \beta_{\text{max}} = 1 \). Thus we have that \( M \mod \alpha \leq \alpha - 1 \), and thus \( \max\{\alpha - \beta_{\text{max}}, M \mod \alpha\} = \alpha - 1 \); and the bound (6.22) becomes

\[
\gamma \geq \alpha - 1 + \left\lfloor \frac{M}{\alpha} \right\rfloor. \tag{6.23}
\]

For the MSR and MBR points considered in the simulations, the above lower bound becomes \( \gamma \geq 7 \) \((M = 16 \text{ and } \alpha = 4)\) and \( \gamma \geq 8 \) \((M = 18 \text{ and } \alpha = 6)\), respectively.
Let us consider the partial downloading scheme. For the MSR point, downloading 7 symbols from 7 storage nodes suffices for the data regeneration of a failed storage node; which achieves the bound \((6.23)\). For the MBR point, downloading 6 symbols from 6 data storage node suffices; which is actually below this lower bound! This is because for data reconstruction the numbers of downloaded symbols \(\{\mu_i\}_{i \in S}\) should satisfy stronger requirement, i.e., \((6.14)\). However, for the wireless setting typically this stronger requirement can be satisfied if the numbers of downloaded symbols satisfy \(\sum_{i \in S} \mu_i = M\).

### 6.4.3.2 Explicit Coding Scheme

For the explicit coding scheme in \([\Pi]\), in order to guarantee the successful data reconstruction for any \(\sum_{i \in S} \mu_i \geq M\), the number of downloaded symbols for data regeneration is given by

\[
\gamma = (s + 1) \left\lfloor \frac{\alpha}{2} \right\rfloor + s(\alpha \mod 2), \quad s = \left\lfloor \frac{M}{\alpha} \right\rfloor.
\]

For the MSR point \((\alpha = 4, M = 16, s = 4)\) and the MBR point \((\alpha = 6, M = 18, s = 3)\) consid-
Ch. 6. Partial Downloading and Resource Allocation for Wireless Distributed Storage Networks

141

The numbers of downloaded symbols for node regeneration are 10 and 12, respectively; whereas using the proposed scheme the numbers of downloaded symbols are 7 and 6 for the MSR and MBR points, respectively. Moreover, for the MBR point the proposed partial downloading scheme requires downloading \( M \) symbols for data reconstruction and \( \alpha \) symbols for node regeneration, both are the minimum possible values.

6.5 Appendices

6.5.1 Proof of Theorem 6.1

Let \( Q = GF(q)^M \). For any \( M \times (M - \mu_i) \) matrix \( H_0 \), if \( \text{span}(\left[ H^{(i)} | A^{(i)} | H_0 \right]) = Q \), then \( \text{span}(\left[ H^{(i)} | H_0 \right]) = Q \) and thus there exists an \( M \times \mu_i \) submatrix \( \tilde{H}^{(i)} \subseteq H^{(i)} \) such that

\[
\text{span}(\left[ \tilde{H}^{(i)} | H_0 \right]) = Q. \tag{6.25}
\]

Then for \( i = 1 \), letting \( H_0 = \left[ H^{(2)} A^{(2)} | \ldots | H^{(S)} A^{(S)} \right] \), we can extract \( \tilde{H}^{(1)} \). For \( i = 2 \), letting \( H_0 = \left[ \tilde{H}^{(1)} \left| H^{(3)} A^{(3)} \right| \ldots \left| H^{(S)} A^{(S)} \right] \right) \), we can extract \( \tilde{H}^{(2)} \). In this way, we obtain \( \tilde{H}^{(i)} \subseteq H^{(i)} \) such that \( \text{span}(\left[ \tilde{H}^{(S)} \right]) = M \).

6.5.2 Symbol Selection Procedure for Partial Downloading at MSR Point

Let \( \Lambda \) be the number of columns of \( \tilde{H}^S \). Let \( \tilde{G}^S = \tilde{H}^S T \) be the Gaussian elimination representation of \( \tilde{H}^S \) via column transformation. Using the matrix \( G^S \) we can test whether a matrix is independent of the columns of \( \tilde{H}^S \).

- If \( h \) is linear independently of the columns of \( \tilde{H}^S \), we add \( h \) to \( \tilde{H}^S \), and accordingly update \( G^S \).
- Otherwise, we randomly select some \( i_0 \in \mathcal{V} \) and a column \( h^{(i_0)} \) of \( \tilde{H}^{(i_0)} \), and express it as a linear combination of the vectors in \( \tilde{G}^S \), given by

\[
h^{(i_0)} = \tilde{G}^S t_0. \tag{6.26}
\]

Let \( \tilde{H}^S = \tilde{G}^S T_0 \) represent the Gaussian elimination procedure for \( \tilde{H}^S \). We have \( h^{(i_0)} = \tilde{H}^S T_0^{-1} t_0 \) and thus an explicit representation of \( \tilde{H}^S \). We search for a column \( h^{(i_1)} \) that is linearly independent of \( \tilde{H}^S \) based on the Gaussian eliminated matrix \( \tilde{G}^S \). Since replacing
$h_{j_1}^{(i_1)}$ with $h_{j_0}^{(i_0)}$ does not change the spanned space and thus the Gaussian eliminated matrix $G^S$, we only need to update the Gaussian eliminated matrix $\bar{G}^S$ for adding $h_{j_1}^{(i_1)}$ to $\bar{H}^S$.

6.5.3 Proof of Theorem 6.3

The proof is by induction, that is, we show that if (6.14) holds for $S = L + 1$, then it also holds for $S = L$; and it holds for $S = 1$.

Let $\lfloor x \rfloor_l \triangleq \min\{x, l\}$. For $S = L + 1$, without loss of generality let $\mu_1 \geq \mu_2 \geq ... \geq \mu_{L+1}$ and $\theta_1 \geq \theta_2 \geq ... \geq \theta_d$. Letting $l = 1$ in (6.14), we have

$$\mu_1 \leq \sum_{j=1}^{d} \lfloor \theta_j \rfloor_1,$$  \hspace{1cm} (6.27)

which implies that $\theta_{\mu_1} \geq 1$; otherwise $\theta_j = 0$ for $j \geq \mu_1$ and thus $\sum_{j=1}^{d} \lfloor \theta_j \rfloor_1 \leq \mu_1 - 1$. Thus we can let $\eta_j^{(1)} = 1$ for $1 \leq j \leq \mu_1$ and $\eta_j^{(1)} = 0$ otherwise, and remove $\mu_1$ by updating $\theta_j \leftarrow \theta_j - 1$ for $1 \leq j \leq \mu_1$. It is easy to see that after subtracting,

$$\sum_{i=2}^{L+1} \mu^{(i)}_i = \sum_{i=1}^{L+1} \mu_i - \mu_1 = \sum_{j=1}^{d} \theta_j - \mu_1$$

$$= \sum_{j=1}^{\mu_1} (\theta_j - 1) + \sum_{k=\mu_1+1}^{d} \theta_j.$$  \hspace{1cm} (6.28)

Next we prove the following to show that (6.14) holds for $S = L$, i.e.,

$$\sum_{i=2}^{l+1} \mu_i \leq \sum_{k=1}^{\mu_1} \lfloor \theta_j - 1 \rfloor_l + \sum_{k=\mu_1+1}^{d} \lfloor \theta_j \rfloor_l,$$  \hspace{1cm} for $1 \leq l \leq L$.  \hspace{1cm} (6.29)

- If $\theta_{\mu_1} \leq l$, we have that $\theta_j \leq l$ for $j \geq \mu_1 + 1$. Thus $\lfloor \theta_j \rfloor_l = \lfloor \theta_j \rfloor_{l+1} = \theta_j$ for $j \geq \mu_{\mu_1} + 1$.

Since $\lfloor \theta_j \rfloor_{l+1} - \lfloor \theta_j - 1 \rfloor_l = 1$ for $1 \leq j \leq \mu_1$, we have

$$\sum_{i=2}^{l+1} \mu_i = \sum_{i=1}^{l+1} \mu_i - \mu_1 \leq \sum_{k=1}^{\mu_1} \lfloor \theta_j \rfloor_{l+1} - \mu_1$$

$$= \sum_{j=1}^{\mu_1} (\lfloor \theta_j \rfloor_{l+1} - 1) + \sum_{j=\mu_1+1}^{d} \lfloor \theta_j \rfloor_{l+1}$$

$$= \sum_{j=1}^{\mu_1} \lfloor \theta_j - 1 \rfloor_l + \sum_{j=\mu_1+1}^{d} \lfloor \theta_j \rfloor_l.$$  \hspace{1cm} (6.30)
• If \( \theta_{\mu_1} \geq l + 1 \), then \( \theta_j \geq l + 1 \) such that \( \theta_j - 1 \leq l \) for \( 1 \leq j \leq \mu_1 \), and thus we have

\[
\sum_{i=2}^{l+1} \mu_i \leq \sum_{i=1}^{l} \mu_i \leq \sum_{j=1}^{\mu_1} \left\lfloor \theta_j \right\rfloor + \sum_{j=\mu_1+1}^{d} \left\lfloor \theta_j \right\rfloor = \sum_{j=1}^{\mu_1} \left\lfloor \theta_j - 1 \right\rfloor + \sum_{j=\mu_1+1}^{d} \left\lfloor \theta_j \right\rfloor. 
\]

(6.31)

Then, (6.14) keeps being satisfied during the above subtracting process, until \( S = 1 \).

Finally for \( S = 1 \), since from (6.14) it follows that \( \mu_1 \leq \sum_{j=1}^{d} \left\lfloor \theta_j \right\rfloor \leq \sum_{j=1}^{d} \theta_j = \mu_1 \),

(6.32)

we have \( \theta_j = \left\lfloor \theta_j \right\rfloor \) and thus \( \theta_j \leq 1 \) for \( 1 \leq j \leq d \). We obtain \( \eta_{(1)}^j = \theta_j \) for \( 1 \leq j \leq d \).

### 6.5.4 Proof of Theorem 6.4

We follow Section 6.3.1 to use the notation \( p_j(X), j \in \mathcal{N} \) as the power consumption of transmitting \( X \) symbols in channel \( j \) by the strongest storage node. Let \( \Delta P_j^{(X)} = p_j(X+1) - p_j(X) \) denote the power increment of transmitting the \((X+1)^{th}\) symbol in channel \( j \). Letting \( X_j \) be the number of symbols transmitted in channel \( j \) for \( j \in \mathcal{N} \), the total power, denoted as \( P_T \), is given by

\[
P_T = \sum_{j \in \mathcal{N}} p_j(X_j) = \sum_{j \in \mathcal{N}} \sum_{X=0}^{X_j-1} \Delta P_j^{(X)},
\]

(6.33)

which shows that the total power \( P_T \) can be expressed as the sum of \( M \) power increments in the \( N \) channels. On the other hand, we can rank the power increments \( \{\Delta P_j^{(X)}\}_{j \in \mathcal{N}, 0 \leq X \leq \alpha-1} \) and find the \( M \) smallest ones according to the rank. In the following we show by induction that the greedy algorithm is optimal in the sense that it selects the \( M \) smallest power increments \( \Delta P_j^{(X)} \). The property used in the proof is that, for any \( j \in \mathcal{N} \), the power increment \( \Delta P_j^{(X)} \) increases with \( X \).

• First, the smallest power increment in the rank is given by

\[
\min_{0 \leq X \leq \alpha-1, j \in \mathcal{N}} \Delta P_j^{(X)} = \min_{j \in \mathcal{N}} \min_{0 \leq X \leq \alpha-1} \Delta P_j^{(X)} = \min_{j \in \mathcal{N}} \Delta P_j^{(0)},
\]

(6.34)

which is selected by the greedy algorithm in the first step.
Then, assuming that the $M'$ smallest power increments have been selected, we show that according to the greedy algorithm we will select the $(M' + 1)^{th}$ smallest one. Let $\mathcal{P}_{M'} = \{\Delta P_{j}^{(X)}\}$ denote the set of the $M'$ smallest power increments, and

$$X_j = \max\{X : \Delta P_{j}^{(X)} \in \mathcal{P}_{M'}\}. \tag{6.35}$$

Since $\Delta P_{j}^{(X)}$ increases with $X$, then for $j \in \mathcal{N}$ we have $\Delta P_{j}^{(X)} \in \mathcal{P}_{M'}$ for $X \leq X_j$, i.e., $X_j$ is the number of symbols allocated to channel $j$. Then according to the rank, the $(M' + 1)^{th}$ smallest power increment is given by

$$\min_{j \in \mathcal{N}} \min_{X \geq X_j} \Delta P_{j}^{(X)} = \min_{j \in \mathcal{N}} \Delta P_{j}^{(X_j)}, \tag{6.36}$$

which is selected by the greedy algorithm.
Chapter 7

Conclusions

We have proposed coding schemes for the following five applications for advanced wireless communication systems:

- message-wise unequal error protection;

- joint network and channel coding for a joint relay communication system;

- a full connected K-user interference channel;

- multi-relay assisted K-user interference channel;

- partial downloading and wireless resource allocation for a wireless distributed storage network.

Driven by the increasing demands for wireless communication systems, the mobile terminals may serve as not only communication terminals but also more intellectual terminals, with the capabilities of not only communication but also distributed sensing, control, recognition, and computing. The base stations will serve as the local centers for not only communication, but also many other information processing and control operations. This will motivate extensive research and development of new theories and techniques. It is difficult to say what is impossible, for the dream of yesterday is the hope of today and the reality of tomorrow. - Robert H. Goddard
Bibliography


[32] A. Roumy and D. Declercq, “Characterization and operation of LDPC codes for the 2-user

[33] K. Price and R. Storn, “Differential evolutiona simple and efficient heuristic for global opti-

[34] T. M. Cover, A. E. Gamal, and M. Salehi, “Multiple access channel with arbitrary correlated


Interference alignment, decomposition, and performance analysis,” IEEE Trans. Info. Theory,

[38] T. S. Han and K. Kobayashi, “A new achievable rate region for the interference channel,”

interference channels with successive group decoding,” IEEE Trans. Info. Theory, vol. 55,

[40] A. Tajer, N. Prasad, and X. Wang, “Beamforming and rate allocation in MISO cognitive radio

[41] A. Motahari and A. Khandani, “To decode the interference or to consider it as noise,” IEEE

[42] A. Tajer, N. Prasad, and X. Wang, “Fair rate adaptation in multiuser interference channels,”
in Proc. IEEE International Symposium on Information Theory (ISIT), Austin, TX, June
2010.


