SPACE-TIME CODED MODULATION DESIGN IN SLOW FADING

by

Akrum Elkhazin

A thesis submitted in conformity with the requirements for the degree of Doctorate of Philosophy in Applied Science
Graduate Department of Electrical and Computer Engineering
University of Toronto

Copyright © 2009 by Akrum Elkhazin
Abstract

Space-Time Coded Modulation Design in Slow Fading

Akrum Elkhazin

Doctorate of Philosophy in Applied Science

Graduate Department of Electrical and Computer Engineering

University of Toronto

2009

This dissertation examines multi-antenna transceiver design over flat-fading wireless channels. Bit Interleaved Coded Modulation (BICM) and MultiLevel Coded Modulation (MLCM) transmitter structures are considered, as well as the used of an optional spatial precoder under slow and quasi-static fading conditions. At the receiver, Multi-Stage Decoder (MSD) and Iterative Detection and Decoding (IDD) strategies are applied. Precoder, mapper and subcode designs are optimized for different receiver structures over the different antenna and fading scenarios.

Under slow and quasi-static channel conditions, fade resistant multi-antenna transmission is achieved through a combination of linear spatial precoding and non-linear multi-dimensional mapping. A time-varying random unitary precoder is proposed, with significant performance gains over spatial interleaving. The fade resistant properties of multidimensional random mapping are also analyzed. For MLCM architectures, a group random labelling strategy is proposed for large antenna systems.

The use of complexity constrained receivers in BICM and MLCM transmissions is explored. Two multi-antenna detectors are proposed based on a group detection strategy, whose complexity can be adjusted through the group size parameter. These detectors show performance gains over the the Minimum Mean Squared Error (MMSE) detector in spatially multiplexed systems having an excess number of transmitter antennas.

A class of irregular convolutional codes is proposed for use in BICM transmissions.
An irregular convolutional code is formed by encoding fractions of bits with different puncture patterns and mother codes of different memory. The code profile is designed with the aid of extrinsic information transfer charts, based on the channel and mapping function characteristics. In multi-antenna applications, these codes outperform convolutional turbo codes under independent and quasi-static fading conditions.

For finite length transmissions, MLCM-MSD performance is affected by the mapping function. Labelling schemes such as set partitioning and multidimensional random labelling generate a large spread of subcode rates. A class of generalized Low Density Parity Check (LDPC) codes is proposed, to improve low-rate subcode performance. For MLCM-MSD transmissions, the proposed generalized LDPC codes outperform conventional LDPC code construction over a wide range of channels and design rates.
Dedication

I would like to dedicate this to my lord, God almighty, the One, Lord of the heavens and earth, our creator and sustainer, the giver of knowledge and judge over all things. I present this humble contribution to the body of knowledge only through Allah will and guidance, and in spite of my own weaknesses and short comings along the way. There is not one novel contribution or seminal idea in the document the came solely from me, or my intelligence: All knowledge comes from God. This is truth.
Acknowledgements

I am grateful to all those people who have supported me directly or indirectly, morally or materially, during my Ph.D. years. At the top of the list are my co-supervisors Professor Subbarayan Pasupathy and Professor Konstantinos Plataniotis, who have been patient with their guidance, helping shape my research and writing skills over the years. My supervisors have been a spring board for ideas and offer great technical and moral support.

I would like to thank Professor Teng Joon Lim and Professor Dimitris Hatzinakos, who were on my defence committee and have given me suggestions about my research on different occasions. I am also grateful to the external examiner on my defence committee, Professor Tho Le-Ngoc, who has given me many constructive comments on my dissertation. I would like to thank my friend Milton Kooistra, who has helped me with proof reading.

I am grateful to my parents for their effort and guidance in raising me, and for their moral and financial support during my graduate studies. May Allah give them good in this world, and good in the next world, and protection from the punishment of the fire.

I have to acknowledge my wife, Jameelah, who has been a moral support through some difficult periods, and always behind me to the fullest of her ability.
Contents

Abstract ii

List of Acronyms viii

List of Tables xi

List of Figures xi

1 Introduction 1

1.1 Wireless Communications Background . . . . . . . . . . . . . . . . . . . 1
1.2 Coded Modulation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
1.3 Receiver Structures . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
1.4 Binary Codes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
1.5 Fading . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
1.6 Multi-Antenna Communications . . . . . . . . . . . . . . . . . . . . . . . 12
1.7 Scope and Overview . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14

2 Generalized Low Density Parity Check Codes 17

2.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
2.2 Multilevel Coded Modulation Design . . . . . . . . . . . . . . . . . . . . 19
2.3 GLDPC Code Structure . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
2.4 GLDPC Code Design and Analysis through EXIT Charts . . . . . . . . . 25
2.5 Generalized Check Node Design ........................................ 30
2.6 Code Graph Construction with Cycles .............................. 34
2.7 Results .................................................................... 36
2.8 Summary .................................................................... 48

3 Irregular Convolutional Coding ............................................ 49
 3.1 Introduction ............................................................. 49
 3.2 Trellis Termination ...................................................... 51
 3.3 Irregular Trellis Concatenation ....................................... 55
 3.4 Irregular Code Design using EXIT Charts ...................... 57
 3.5 Simulated Results ....................................................... 66
 3.6 Summary .................................................................... 73

4 Iterative Space-Time Detection ............................................. 75
 4.1 Introduction ............................................................. 75
 4.2 System Model ........................................................... 77
    4.2.1 Layered Space-Time Transmitter .............................. 77
    4.2.2 Channel Model ..................................................... 78
    4.2.3 IDD Layered Receiver ............................................ 79
 4.3 Detector Design ........................................................ 80
    4.3.1 Reduced Dimension MAP Detector ....................... 80
    4.3.2 Group MAP Detector ............................................ 83
 4.4 BER Analysis ............................................................ 85
 4.5 Simulated Results ....................................................... 91
 4.6 Summary .................................................................... 98

5 Space-Time Coding Over Slow Fading Channels .................... 100
 5.1 Introduction ............................................................. 100
 5.2 System Model ........................................................... 102
<table>
<thead>
<tr>
<th>abbrev.</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>APP</td>
<td>a posteriori probability</td>
</tr>
<tr>
<td>AWGN</td>
<td>Additive White Gaussian Noise</td>
</tr>
<tr>
<td>BCJR</td>
<td>Bahl-Cocke-Jelinek-Raviv</td>
</tr>
<tr>
<td>BICM</td>
<td>Bit-Interleaved Coded Modulation</td>
</tr>
<tr>
<td>BLAST</td>
<td>Bell Layered Space-Time</td>
</tr>
<tr>
<td>BER</td>
<td>Bit Error Rate</td>
</tr>
<tr>
<td>bps</td>
<td>Bits per second</td>
</tr>
<tr>
<td>CDMA</td>
<td>Code Division Multiple Access</td>
</tr>
<tr>
<td>CM</td>
<td>constrained modulation</td>
</tr>
<tr>
<td>CSI</td>
<td>Channel State Information</td>
</tr>
<tr>
<td>EXIT</td>
<td>Extrinsic Information Transfer</td>
</tr>
<tr>
<td>FER</td>
<td>Frame Error Rate</td>
</tr>
<tr>
<td>GLDPC</td>
<td>Generalize Low Density Parity Check</td>
</tr>
<tr>
<td>GMAP</td>
<td>Group Maximum a posteriori</td>
</tr>
<tr>
<td>IDD</td>
<td>Iterative Detection and Decoding</td>
</tr>
<tr>
<td>ISI</td>
<td>Intersymbol Interference</td>
</tr>
<tr>
<td>LDPC</td>
<td>Low Density Parity Check</td>
</tr>
<tr>
<td>LLR</td>
<td>Log Likelihood Ratio</td>
</tr>
<tr>
<td>Abbrev.</td>
<td>Name</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>MAC</td>
<td>Multiply and Accumulate</td>
</tr>
<tr>
<td>MAP</td>
<td>Maximum a posteriori</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple-Input Multiple-Output</td>
</tr>
<tr>
<td>MI</td>
<td>Mutual Information</td>
</tr>
<tr>
<td>ML</td>
<td>Maximum Likelihood</td>
</tr>
<tr>
<td>MLCM</td>
<td>Multilevel Coded Modulation</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean Square Error</td>
</tr>
<tr>
<td>MQAM</td>
<td>Multiple Quadrature Amplitude Modulation</td>
</tr>
<tr>
<td>MSD</td>
<td>Multi-Stage Decoder</td>
</tr>
<tr>
<td>OFDM</td>
<td>Orthogonal Frequency Division Multiplexing</td>
</tr>
<tr>
<td>PEG</td>
<td>Progressive Edge Growth</td>
</tr>
<tr>
<td>PSK</td>
<td>Phase Shift Keying</td>
</tr>
<tr>
<td>RSC</td>
<td>Recursive Systematic Convolutional</td>
</tr>
<tr>
<td>QAM</td>
<td>Quadrature Amplitude Modulation</td>
</tr>
<tr>
<td>RDMAP</td>
<td>Reduced Dimension Maximum a posteriori</td>
</tr>
<tr>
<td>SISO</td>
<td>Soft-Input Soft-Output</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>TCM</td>
<td>Trellis Coded Modulation</td>
</tr>
</tbody>
</table>
List of Tables

2.1 Optimal GLDPC check node degrees as function of the code rate and codeword length. .......................................................... 37
2.2 Optimal LDPC check node degree as function of the code rate and codeword length. .......................................................... 37
2.3 Threshold values $\frac{E_b}{N_0}(dB)$ for GLDPC codes, LDPC codes and LDPC codes with Hadamard constraints .................. 40
3.1 Generators for RSC mother codes ............................................. 61
3.2 Puncture Patterns for $m_0 = 1$ RSC mother codes (octal) .......... 62
3.3 Puncture Patterns for $m_0 = 2$ RSC mother codes (octal) .......... 62
3.4 Puncture Patterns for $m_0 = 3$ RSC mother codes (octal) .......... 63
3.5 Puncture Patterns for $m_0 = 4$ RSC mother codes (octal) .......... 63
3.6 Puncture Patterns for $m_0 = 5$ RSC mother codes (octal) .......... 63
3.7 Puncture Patterns for $m_0 = 6$ RSC mother codes (octal) .......... 65
3.8 Puncture Patterns for $m_0 = 7$ RSC mother codes (octal) .......... 65
3.9 Code SNR threshold values in decibels for different labelling functions, channels and constellations. ............................. 68
3.10 AWGN code coefficients for memory $M = 7$ design with random MQAM. ........ 70
4.1 Complexity of RDMAP, GMAP, MAP and MMSE, detectors per coded bit. .... 91
List of Figures

1.1 Bit-Interleaved Coded Modulation ................................................. 3
1.2 Multilevel Coded Modulation ......................................................... 4
1.3 Multistage Decoder ......................................................................... 5
1.4 Iterative Detection and Decoding ..................................................... 6
1.5 Regular LDPC graph (left) and EXIT chart (right) ............................ 8

2.1 Plot of the inverse MI function (top) and its derivative (bottom) as a function of rate for the binary AWGN channel ................................. 21
2.2 Graph of irregular LDPC code with generalized check node ............. 23
2.3 EXIT chart of GLDPC code over BICM channel ................................. 30
2.4 EXIT chart for super check nodes with $d_c = 7$ and $q_c = [0 \ldots 6]$ (left) and irregular repetition codes with $d_v = [2 \ldots 8]$ (right) ............... 33
2.5 BER of GLDPC codes, LDPC codes and LDPC codes with Hadamard constraints over AWGN channel at rates $r = \{0.05, 0.1, 0.2, 0.3\}$ for transmission of $N_u = 2000$ uncoded bits ......................................................... 38
2.6 FER of GLDPC codes, LDPC codes and LDPC codes with Hadamard constraints over AWGN channel at rates $r = \{0.05, 0.1, 0.2, 0.3\}$ for transmission of $N_u = 2000$ uncoded bits ......................................................... 39
2.7 Rate-loss of GLDPC and LDPC codes as a function of rate for $N_c = \{250, 4000\}$ ......................................................... 41
2.8 Power-loss of GLDPC and LDPC codes as a function of rate for $N_c = \{250, 4000\}$ ......................................................... 42
2.9 Rate-loss for 16-QAM MLCM-MSD transmission using GLDPC (top) and LDPC (bottom) subcodes and different labelling functions for packet lengths of $N_c = \{250, 4000\}$ .............................. 43

2.10 Rate-loss for 256-QAM MLCM-MSD transmission using GLDPC (top) and LDPC (bottom) subcodes and different labelling functions for packet lengths of $N_c = \{250, 4000\}$ .............................. 44

2.11 FER of LDPC and GLDPC coded MLCM-MSD systems over AWGN channel using 16-QAM ................................................................. 45

2.12 Conditional pdf of messages from the first subchannel using Gray and Anti-Natural labelling at transmission rates $r = \{1, 2\}$ .............................................. 46

2.13 Comparison of GLDPC coded BICM-IDD and MLCM-MSD transmissions at rates $r = \{1, 2, 3\}$ ................................................................. 47

3.1 Recursive systematic convolutional code with generator $g_{oct} = [7 \ 5]$, with corresponding trellis. Also shown is the projected trellis over $m_0 = 3$ space. .......................... 54

3.2 EXIT Curves for rate $\{1/2, 1/4\}$ convolutional outer codes for code memory $M = \{1, \ldots, 7\}$ ................................................................. 59

3.3 EXIT curves for punctured subcode set for code memory $M = \{1, \ldots, 7\}$ .... 64

3.4 Channel EXIT curves for different 16-QAM mapping functions over the AWGN and $2 \times 2$ MIMO fading channel. ........................................ 67

3.5 Optimized outer code EXIT curve for different mapping functions over the AWGN channel at a binary code rate of $1/2$ ........................................ 69

3.6 FER performance of 16-QAM BICM system over AWGN channel using fixed and variable sized trellises. ............................................ 71

3.7 AWGN channel performance of proposed codes for 2000 symbol transmissions. 72

3.8 AWGN channel performance of proposed codes for 250 symbol transmissions. 72

3.9 $2 \times 2$ MIMO channel performance using proposed codes for 1000 symbol transmissions. ................................................................. 73
3.10 2 × 2 MIMO channel performance using proposed codes for 125 symbol trans-
missions................................................................. 74

4.1 Layered Space-Time Transmitter ........................................ 78
4.2 IDD Layered Receiver ...................................................... 79
4.3 Uncoded BER performance of RDMAP and GMAP detectors for N = 6, M = 6
system using QPSK modulation with N_G = 1, 2, 3, 4. ......................... 92
4.4 Uncoded BER performance of RDMAP and GMAP detectors for QPSK system
with N = 6, N_G = 3 and different M. .......................................... 93
4.5 Coded BER comparison of RDMAP, GMAP, MMSE, and MAP detectors using
QPSK for N = 6, M = 3 system and N = 6, M = 5 system. Group size N_G = 4
for the RDMAP and GMAP detectors. ........................................ 94
4.6 Coded BER as a function of the number of turbo iterations for RDMAP and
GMAP detectors using QPSK for N = 6, M = 3 system with N_G = 4. ........ 96
4.7 Coded BER comparison of RDMAP and GMAP detectors for N = 10, M = 4
QPSK system with N_G = 1, 2, 4, 5. ........................................... 97
4.8 Coded BER comparison of MMSE, RDMAP and GMAP detectors for N =
4, M = 2 system using 16-QAM modulation with N_G = 2, 4. .................... 98

5.1 Space-Time MLCM Transmitter .......................................... 103
5.2 CM capacity as a function of the unconstrained capacity for unprecoded and
isotropic unitary precoded systems ............................................. 111
5.3 Subchannel capacities as a function of the CM Capacity for unprecoded system . 113
5.4 Subchannel capacities as a function of the CM Capacity for spatial interleaving
and isotropic unitary precoding ................................................. 114
5.5 Subchannel capacities as a function of the unconstrained capacity for random
labelling with and without random isotropic unitary precoding ............... 116
5.6 FER performance of BICM-IDD and MLCM-MSD systems using QPSK over
2 × 2 quasi-static Rayleigh fading channel for different mapping and precoder
choices ................................................................. 123
5.7 FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over
2 × 2 quasi-static Rayleigh fading channel for different mapping, precoder and
detector choices. Target error rate of FER=0.005 . . . . . . . . . . . . . . . . . . 124

5.8 FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over
2 × 2 quasi-static Rayleigh fading channel for different mapping, precoder and
detector choices. Target error rate of FER=0.1 . . . . . . . . . . . . . . . . . . . 125

5.9 FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over
2 × 2 quasi-static Rayleigh fading channel for different mapping, precoder and
detector choices and block size $N_s = 250$. . . . . . . . . . . . . . . . . . . 126

5.10 FER performance of BICM-IDD and MLCM-MSD systems using QPSK over
4 × 4 quasi-static Rayleigh fading channel for different mapping and precoder
choices . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 127

5.11 FER performance of BICM-IDD system using 16-QAM over 4 × 4 quasi-static
Rayleigh fading channel for different mapping, and detector choices, and block
size $N_s = 1000$. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 128

5.12 FER performance of MLCM-MSD system using 16-QAM over 4 × 4 quasi-static
Rayleigh fading channel for different mapping, precoder and detector choices,
and block size $N_s = 1000$. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 129

5.13 FER performance of BICM-IDD and MLCM-MSD system using 16-QAM over
4 × 4 quasi-static Rayleigh fading channel for different mapping, precoder and
detector choices, and block size $N_s = 250$. . . . . . . . . . . . . . . . . . . 130

5.14 FER performance of BICM-IDD and MLCM-MSD system using QPSK over 4 × 2
quasi-static Rayleigh fading channel for different mapping, precoder and detector
choices, and block size $N_s = 1000$. . . . . . . . . . . . . . . . . . . . . . 131

5.15 FER performance of BICM and MLCM system using 16-QAM over 4 × 2 quasi-
static Rayleigh fading channel for different mapping, precoder and detector
choices, and block size $N_s = 1000$. . . . . . . . . . . . . . . . . . . . 131
5.16 FER performance of BICM-IDD and MLCM-MSD system using 16-QAM over
2 × 2 doppler Rayleigh fading channel with $\Delta f_{dop} = 0.01$ for different mapping,
precoder and detector choices, and block size $N_s = 1000$. .............................. 133

5.17 FER performance of BICM-IDD and MLCM-MSD system using 16-QAM over
2 × 2 doppler Rayleigh fading channel with $\Delta f_{dop} = 0.01$ for different mapping,
precoder and detector choices, and block size $N_s = 250$. .............................. 133

5.18 FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over
2 × 2 Rician channel with unity k-factor for different mapping, precoder and
detector choices, and block size $N_s = 1000$. .............................. 134

5.19 FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over
2×2 correlated Rayleigh fading channel ($\alpha = 0.5$) for different mapping, precoder
and detector choices, and block size $N_s = 1000$. .............................. 135

5.20 FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over
2 × 2 urban microcell environment for different mapping, precoder and detector
choices, and block size $N_s = 1000$. .............................. 136

5.21 FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over
2 × 2 suburban macrocell environment for different mapping, precoder and de-
tector choices, and block size $N_s = 1000$. .............................. 137
Chapter 1

Introduction

1.1 Wireless Communications Background

Wireless personal communications has seen explosive growth in the last twenty years, finding application in a plethora of consumer devices. In the 1980s, personal wireless communications was in its infancy, relatively speaking. Passive paging networks and first generation analog cell phone networks were in place, although cell phones had a high mobile unit cost and power consumption that limited size, battery life and consumer accessibility. One also has to mention wireless phone technology that represented the last leg in the telephony network. In addition, citizens’ band radios continue to be ad hoc voice networks with low cost walkie talkies handsets that date back to the Second World War.

At present, a wide array of wireless communications devices play an increasingly larger role in everyday life. Third generation cell phone systems support an enriched range of services that include voice, email, internet connectivity, streamed multimedia, and video conversations. Wireless local area networks first grew with home networking products that have made their way into businesses and university networks with improved security and reliability. Wireless local area networks have permeated public places such
as hotels, cafes, malls and convention facilities. Broadband wireless is encroaching on the market that provides the last mile of service, traditionally dominated by cable and digital subscriber line providers. Finally, a wide range of products have been developed based on wireless personal area networks, such as Bluetooth. These products provide wireless connectivity to hand held, miniature and peripheral devices.

Next generation wireless systems are converging based on the demand for high-bandwidth multimedia applications over packet based networks. Transmission throughput comes at the cost of other system constraints such as transmission power that is limited by battery life, and spectral bandwidth that is often shared by a number of users. The capacity of the physical wireless channel is limited by channel fading characteristics. In a scattered fading environment where multiple reflections of the transmitter signal are received, random motion leads to periods where the reflected signals interfere destructively. Periods of destructive interference or deep fades make it difficult to transmit reliably with at a high throughput. Multi-antenna communications has emerged as a means to mitigate fading, because of the large number of wireless links between transmitter and receiver antenna pairs. It has been shown that the channel capacity grows with the minimum of the number of transmitter and receiver antennas in highly scattered environments [1, 2]

1.2 Coded Modulation

Most data communication systems involve some kind of error correcting code, and high throughput applications often require a non-binary modulation format. Many wireless transmissions can be thus classified as coded modulation systems, involving one or more binary codes connected to a mapping or modulation block. An early form of coded modulation is trellis coded modulation, which has been employed for a number of years in telephone modems [3]. Trellis coded modulation is a multilevel code that maps different
state transitions in a convolutional trellis to different constellation points. This scheme performs well for short packet transmissions, but does not support iterative decoding.

For longer packet transmissions on the order of hundreds of bits or longer, there is an advantage in introducing an interleaver between the outer code and the symbol mapper. Interleaving enables the use of near capacity achieving iterative decoding procedures, which is the basis for modern binary code design. Interleaving also serves another function in fading channels. Part of a packet transmission may undergo a deep fading, leading to a burst of poorly detected symbols. Interleaving spreads or randomizes detector errors, thus improving performance in time-fading channels [4, 5, 6].

There are a few types of interleaved coded modulation systems. The first is Bit-Interleaved Coded Modulation (BICM), whose structure is shown in Figure 1.1. A single outer encoder is used and coded bits are arbitrarily assigned to bit positions at the mapper input. The mapping or labelling function maps a binary vector to different points in a Phase-Shift Keying (PSK) or Quadrature Amplitude Modulation (QAM) symbol constellation. With the exception of the natural labelling format, the mapping function is a non-linear function. For multi-antenna applications, an optional precoder is shown that performs linear spatial spreading.

The second major coded modulation format is Multilevel Coded Modulation (MLCM) [7, 8] whose structure is shown in Figure 1.2. The main difference with the BICM structure is that a different interleaver and encoder is used for each bit in the symbol vector.
The encoders may be identical or have different rates and coding properties. The Bell Labs Space-Time (BLAST) architecture [9] and other layered architectures are MLCM transmissions with a structured mapping function. In the layered framework, a group of bits are mapped to the QAM or PSK symbol that is transmitted on each antenna. In slow fading, it is advantageous to introduce a time-varying mapping function, so that a deep fade on one antenna is averaged over different bits in the symbol vector. This is achieved by implementing the spatial precoder as a cyclicly rotating spatial interleaver in the Diagonal BLAST (D-BLAST) variant [9].

Another interleaved coded modulation structure that should be mention is turbo TCM [10], which consists of two TCM codes that are interleaved in parallel. This coded modulation format is often employed with higher order modulation formats.

1.3 Receiver Structures

There are a wide range of possible receiver structures and operations that can be applied to coded modulation transmissions. The majority of receiver side processing can be categorized as either detection or decoding operations. Detection produces information about the coded bits based on the received symbol, while decoding produced bit information based on the constraints of the underlying code or codes. These operations are performed by detector and decoder functions. Receiver structures can thus be understood in terms
of how the detector and decoder block are connected and interact.

The simplest receiver structure is the one-shot receiver that calls the detector once and passes coded bit information to the decoder, which then produces uncoded bit decisions. The one-shot receiver can be applied to either BICM or MLCM transmissions, although it is in general not capacity achieving for higher order modulation formats. Gray labelling is typically used with this structure so as to minimize detector message correlation. More correlated labelling functions require decoder feedback in order to avoid significant performance loss.

The next receiver structure is the MultiStage Decoder (MSD), shown in Figure 1.3. The MSD is based on sequential detection and decoding of bits in the symbol vector. At the $l^{th}$ stage, the $l^{th}$ bit in the symbol vector is detected and decoded, using previously decoded bit decisions in the detection operation. In order to perform complete decoding at each stage, each bit in the symbol vector must be encoded independently, thus the MSD can only be applied to MLCM transmissions. MLCM-MSD systems are near capacity achieving for many channels with appropriately constructed subcodes. They are less effective for shorter packet transmissions because of the block length reduction associated with using independent subcodes.

The last receiver structure is the Iterative Detection and Decoding (IDD) structure, shown in Figure 1.4. The IDD receiver passes soft bit information between the detector
and one or more decoders on each iteration. Unlike the MSD structure that uses hard bit decisions on the feedback path, both the detector and decoder produce soft outputs under IDD processing. This leads to a performance advantage over staged decoding for shorter transmissions and under certain fading conditions.

1.4 Binary Codes

Both BICM and MLCM transmission schemes are based on binary codes, with corresponding receiver side decoding structures. Turbo codes and Low Density Parity Check (LDPC) codes are two classes of binary random codes used in current and next generation wireless systems [11, 12]. Both are concatenated codes that can be decoded using an iterative decoding procedure, wherein subcode decoding elements exchange soft bit information. A turbo convolutional code [13, 14] is a parallel concatenation of two recursive systematic convolutional (RSC) codes, separated by a random interleaver. A LDPC code is a serial concatenation of an inner repetition code and an outer single parity check code, separated by a random interleaver [15, 16, 17]. Both the inner and outer codes may be irregular, meaning different repetition rates are used, and parity bits are generated using different numbers of information bits. An irregular LDPC code is specified by a
code profile, which is the relative fraction of different repetition rates and parity bits used to generate the code.

For long code construction, irregular LDPC codes can be designed to perform to within a fraction of a decibel of the channel capacity. Turbo codes have higher error floor, and incur significant decoding latency for longer code lengths because of the serial nature of encoding and decoding. For shorter transmissions, message correlation in the decoding process tends to limit performance. The LDPC code graphs is sparse and more sensitive to message correlation, whereas the convolutional trellis is more structured and less affected by message correlation. Accordingly, convolutional turbo codes tend to outperform LDPC codes at shorter block lengths.

The design and analysis of LDPC and convolutional turbo codes is well understood for binary channels such as the binary Additive White Gaussian Noise (AWGN) channel [13, 14, 16, 17]. Code design rules can be derived from the analysis of the iterative decoding process, typically under the assumption that the decoding graph is loop-free. Under the loop-free assumption, LDPC code performance can be predicted by analyzing the evolution of the check and repetition node message densities [18]. Within a density evolution framework, an irregular LDPC code profile can be found through optimization, although the process is computational intensive. In order to simplify analysis, the message densities are often assumed to be Gaussian. With the Gaussian assumption, the convergence process can be visualized in an Extrinsic Information Transfer (EXIT) chart, which can be constructed for convolutional turbo codes [19] and LDPC codes [20].

EXIT chart analysis is applied throughout this dissertation and is worth some discussion. Consider a regular LDPC code with $d_v$ parity checks per bit and $d_c$ bits per parity check. The bi-partite graph representing such a code is shown in Figure 1.5 on the left, where the variable and check nodes correspond to repetition and single parity check codes. Decoding is accomplished through an iterative exchange of messages between check and variable node elements. On each iteration, each node element receives a set
of *intrinsic* messages on connected edges, and responds with a set of *extrinsic* messages. The evolution of this iterative process can be analyzed by plotting the extrinsic information transfer properties of the check and variable nodes. The variable node extrinsic mutual information is plotted as a function of the intrinsic information on the right of Figure 1.5, where the intrinsic input is generated with a Gaussian density. The left most point represents the contribution of the channel observation with no intrinsic contribution, and the rightmost point is the output when the input messages are all correct. Also shown on the right of Figure 1.5 is the check node EXIT curve, except that the axes are swapped so that the intrinsic mutual information is on the y-axes and the extrinsic output is on the x-axes.

The decoding processes can be visualized through a stair-case decoding trajectory that is shown between the check and variable EXIT curves. The initial point at \((x, y) = (0, y_0)\) represents the initial variable node contribution, that is passed to the check node whose
output has extrinsic mutual information $x_0$, represented by the point $(x, y) = (x_0, y_0)$. Based on the check node output $x_0$, the variable node outputs $y_1$, followed in turn by the check node output $x_1$. The decoding process continues to the point where the variable and check node curves intersect. If this point is at $x = y = 1$, then the messages are all correct and successful decoding is predicted. If on the other hand the two curve intersect a some point before $x = y = 1$, some messages bits are in error causing a packet decoding error. One constraining factor in the decoding analysis is the shape of the detector and decoder curves. The region between the two curves needs to be minimized, as this area approximately represents the code’s capacity loss [21]. The check and variable node EXIT curves are generally chosen to approximately match, so that a decoding tunnel is opened with a small increase in the channel Signal-to-Noise power Ratio (SNR).

For non-binary channels, the code design problem depends on the receiver structure. When the iterative receiver is applied, EXIT chart analysis can be used to design either the outer code in a BICM transmission, or a set of identical subcodes in a MLCM transmission. A non-concatenated outer code may be used, since iterative processing is applied between the detector and decoder. In fact, using a concatenated outer code can increase receiver latency, since several decoder sub-iteration are required for each detector call. Most often a single outer code such as a convolutional code is employed. In this case, EXIT charts can be used to match the detector and decoder EXIT curves. The mapping function strongly influences the slope of the detector curve, while code properties determine the shape of the decoder curve.

The MSD receiver can only be applied to MLCM transmissions. The subcodes in an MLCM transmission are determined through a decomposition of the channel capacity into a set of virtual binary channels. The channel capacity $C = I(b; y)$ is the mutual information between the binary symbol vector $b$ and noisy channel observation $y$. This
capacity can be decomposed as

\[
I(B; y) = I(b_1; y) + I(b_2; y|b_1) + \ldots + I(b_L; y|b_1, \ldots, b_{L-1})
\]

\[= C_1 + C_2 + \ldots C_L \quad (1.1)\]

where \(C_l\) is the capacity of the \(l^{th}\) virtual channel. The \(l^{th}\) virtual subchannel is formed using a single information bit \(b_l\), assuming lower index bits \(b_i, i < l\) are known, and treating higher index bits \(b_i, i > l\) as a noise contribution. These channels are virtual since they require knowledge of lower index bits. Knowledge of lower index bits is only available in a multistage receiver, provided that previous decoded stages are error-free. Error-free decoding is ensured by encoding each symbol bit at rate slightly lower rate than the corresponding subchannel capacity. The distribution of subchannel capacities is largely determined by the mapping scheme used.

1.5 Fading

Fading generally refers to the fluctuation in the envelope of the transmitted signal. The classification of fading depends on the distances considered. Over a large distance scale, the gradual attenuation of the receiver signal as a function of physical distance is known as large-scale fading or path-loss. Over a small distances, rapid fluctuations in the signal envelope is referred to as small-scale fading. This phenomena stems from the fact that the received signal contains many reflections of the transmitted signal with different amplitudes and phases. Phase differences are far more critical than variations in amplitude, as the former can give rise to destructive interference which significantly limits channel capacity. The mitigation of small-scale fading is a central concern in wireless transceiver design.

Small-scale fading can be further classified based on the symbol interval used. If the multi-path components have a delay spread that is comparable or longer than the symbol
time, then the signal undergo frequency-selective fading that leads to Inter-Symbol Interference (ISI). Frequency-selective fading is maybe accommodated with a multi-carrier modulation such as Orthogonal Frequency Division Multiplexing (OFDM), which uses a set of orthogonal sub-carriers that each sees a frequency-flat channel. The symbol time is also key factor when considering time-varying channel characteristics. If the channel undergoes significant changes over a few symbols, then it is classified as a fast-fading channel, otherwise it is known as a slow-fading channel. In a fast-fading environment where channel estimation may be difficult, differential modulation is often used as it enables detection without explicit channel knowledge. A limiting case of slow-fading is quasi-static fading, wherein the channel remains constant for the duration of a packet transmission.

Channel fading characteristics significantly affect link capacity and reliability. High throughput reliable communications is particularly problematic in a frequency-flat quasi-static fading channel. The quasi-static fading condition means that if the channel is bad, it is bad for the whole packet, and that packet will likely be decoded erroneously. A time-varying channel will average good and bad portions of the channel, and it is unlikely the channel will remain bad for an entire packet. A frequency-flat channel is one where the entire transmission bandwidth experiences the same fading phenomenon, be it good or bad. In frequency selective fading, different parts of the transmission bandwidth undergo different fading phenomena, and it is improbable that the entire spectrum will be in a deep fade.

For multi-antenna communications, the spatial fading properties of the wireless channel become more significant. The simplest spatial fading channel is the Rayleigh channel, which assumes an infinite number of isotropically distributed scatters with no line-of-sight component. The addition of a line-of-sight component leads to Rician fading with a slight increases in channel capacity. Antenna correlation due to insufficient antenna spacing is often modelled with a correlated Rayleigh channel model. A number of more realistic channel models have been developed to model indoor and outdoor wireless chan-
channels [22, 23, 24, 25, 26, 27]. These models account for antenna geometries and polarization, as well as different environmental scenarios such as urban versus rural conditions. Particular fading scenarios such as the urban canyon [26, 27] and keyhole fading [24, 28] around building openings have also been modelled.

1.6 Multi-Antenna Communications

Multi-antenna communications can improve both the reliability and data throughput of a wireless link. Transmission reliability is improved through spatial diversity, wherein redundant copies of the transmitted signal are sent on different transmitter antennas. Transmission throughput can be increased through spatial multiplexing, which sends independent data on each transmitter antenna. In the high SNR region, transmission capacity grows with the minimum of the number transmitter and receiver antennas in a scattered environment [1, 2]. There is a tradeoff between diversity and multiplexing, such that increased capacity comes at the cost of reduced diversity. A broad range of multi-antenna coding strategies have been developed that exploit diversity and multiplexing gains to different degrees. These strategies can be classified based on the number of symbols spanned by the underlying code or coded modulation scheme.

Transmissions of a few symbols are best coded with space-time block codes. These codes are generally based on transmitting multiple copies of symbols on different transmitter antennas with appropriate scaling. The code structure and coefficients are often determined using rank and determinant criteria derived from the eigenvalue distribution of the combined code/channel correlations matrix [29]. The well known Alamouti code [30] is a rate-one orthogonal space-time block code for two transmitter antennas. This design has been generalized to larger systems in [31], however, there is a rate limitation with orthogonal designs. Higher rates are possible with quasi-orthogonal space-time block codes [32, 33], which relax the constraint of linear decoding complexity for ad-
ditional rate. Linear dispersion codes are a class of space-time block codes whose code coefficients are found through numerical optimization. Optimization techniques based on both mutual information and the determinant and rank criteria were initially proposed in [34], but better performance is possible by direct minimization of the block error rate through stochastic gradient search methods [35].

For transmissions longer than a few symbols, Maximum Likelihood (ML) decoding performance is not possible using space-time block codes alone because of the computational infeasibility. Space-time trellis coded modulation systems have been proposed [29] whose state mapping is determined through the rank and determinant criteria. For packet lengths on the order of hundreds of symbols, trellis coded modulation is outperformed by time-interleaving schemes such as BICM or MLCM.

BICM and MLCM architectures can be extended to space-time applications through the choice of the symbol mapper and precoder. The precoder is the linear component of the space-time coded modulation and the symbol mapper the non-linear component. Spatial precoders have been designed for BICM systems under different fading conditions [36, 37, 38, 39, 40, 41, 42]. In the MLCM framework, the D-BLAST variant of the layered architecture implements the spatial precoder as a spatial interleaver. This leads to a performance improvement under slow and quasi-static fading conditions. Precoders have also been designed correlated for fading channels [43, 44]

Multi-dimensional mapping is a particularly useful coding strategy under slow fading conditions. The most promising fade resistant mapping strategy is multi-dimensional random labelling [45, 46], although other coding and mapping strategies have been proposed [47]. Multi-dimensional random labelling generally uses a pseudo-random mapping function that changes between symbols and is known at the receiver. Random mapping leads to highly correlated detector symbol messages, whose statistical properties are less sensitive to the effects of fading. In a MLCM-MSD framework, near capacity achieving performance is possible using random labelling under quasi-static fading
Chapter 1. Introduction

Multi-dimensional random labelling poses a few design challenges. Since random labelling requires optimal detection, it may be computationally infeasible for large systems. In MLCM-MSD transmissions, random labelling leads to a large subchannel capacity spread that requires subcodes to be designed over a wide range of rates. The LDPC codes used in [45, 46] incur a significant capacity loss for low rate subcodes. In BICM-IDD transmissions, random labelling leads to steeply sloped channel EXIT curves. This is problematic for many existing code design procedures. To our knowledge, multi-dimensional random labelling has not been used in BICM systems to mitigate slow fading.

1.7 Scope and Overview

There are range of wireless communications systems that can benefit from multi-antenna solutions. Many of these systems involve multiple users sharing the communications channel, and others involve multi-carrier modulation formats, such as OFDM, that mitigate the effects of frequency selective fading. As the focus of this work is on the multi-antenna aspects of the system, the remaining design factors will be kept as simple as possible. To this end, the scope of this dissertation is limited to single-carrier single-user transmissions of an independent binary data source. Extensions to multimedia applications, multi-carrier OFDM, and multiuser systems are discussed as future work in Chapter 6. In addition, we only consider coded modulation designs with transmissions lengths in the range of hundreds-to-thousands of symbols. We also focus on high-rate designs that exploit the multiplexing gain of multi-antenna wireless channels.

The next major design considerations are the channel fading characteristics and how the channel is know at the transmitter and receiver. We consider frequency-flat fading channels, and focus on the slow fading scenario where channel coherence time is signif-
Chapter 1. Introduction

Significantly larger than the symbol time. Under slow-fading conditions, there is negligible ISI due to Doppler in the pulse shaping filter, and channel estimation is generally not a challenging problem. Channel estimation is avoided altogether in this dissertation by assuming perfect channel state information (CSI) at the receiver. It is further assumed that CSI is not available to the transmitter.

Different transmitter and receiver configurations are considered. Precoder and mapper designs are proposed for BICM and MLCM transmissions based on channel fading characteristics and receiver complexity constraints. Two suboptimal detectors are proposed based on a group detection strategy. Within a BICM-IDD framework, a class of irregular convolutional codes are proposed, which can be designed to suit different channel conditions. For MLCM-MSD systems, a class of generalized LDPC codes is proposed with performance gains over fading and non-fading channels.

Chapter 2 examines the use of LDPC codes in coded modulation systems. For finite length MLCM-MSD transmissions, set partitioning labelling minimizes capacity loss provided that efficient binary codes can be constructed over a wide rate range. Conventional LDPC codes perform well for moderate-to-high rate construction, but not in the low rate region. A class of generalized LDPC codes is proposed with improved low rate performance. This is achieved through a generalized check node that connects additional check node parity bits to the channel without repetition. The proposed generalized LDPC codes can also be designed for BICM-IDD systems.

Chapter 3 develops an irregular convolutional code construction procedure within a BICM-IDD system. A convolutional code can be made irregular using generator polynomials of different memory, and a set of puncture patterns that produce different subcode rates. The generator/puncture pattern combination makes a subcode, and the weighted sum of subcodes forms the overall code. An irregular code profile is determined through EXIT chart analysis, and different concatenation strategies are proposed to connect varying size trellises. The proposed codes outperform turbo codes and MLCM-MSD designs.
Chapter 1. Introduction

over multi-antenna channels.

Chapter 4 develops two detectors based on a group detection strategy that can operate within a IDD receiver. These detectors are useful in systems having a large constellation and number of transmitter antennas, such that the optimal detector becomes computationally infeasible. A suboptimal decision is made using a subset of the symbol vector, treating the remaining signal contribution as noise that is suppressed through filtering. The proposed detectors have a significant performance improvement over linear processing detectors in spatially multiplexed systems having fewer receiver antennas than transmitter antennas.

Chapter 5 tackles the problem of multi-antenna precoder and mapper design under slow fading. The BICM and MLCM coding strategies developed in Chapter 2 and Chapter 3 are extended to the slow fading conditions, and the detection strategies developed in Chapter 4 are applied where appropriate. Isotropic unitary random precoding is presented as an alternative to the spatial interleaving strategy used in layered multiplexing schemes. Unitary precoding does not affect receiver complexity and allows the use of a linear detector. Random multidimensional mapping is discussed as an effective non-linear spatial coding technique that requires a high complexity detector. Under optimal detection, the IDD receiver is shown to have a fundamental performance advantage over the MSD receiver in quasi-static fading, regardless of packet length. For large constellations, a group random mapping strategy is proposed for MLCM-MSD systems where optimal detection is not feasible.
Chapter 2

Generalized Low Density Parity Check Codes

Low density parity check codes are widely used in coded modulation systems in both bit-interleaved and multilevel coded modulation structures. In MLCM systems, set partition labelling has a rate advantage over other formats, but generates a wide range of sub code rates that may be difficult to implement. Conventional LDPC code construction incurs a significant capacity loss for low rate subcodes. A class of generalized LDPC codes is proposed in this chapter that improves conventional LDPC code performance in the low rate region. These codes are designed for MSD and IDD decoding.

2.1 Introduction

The interaction between the code and mapping function plays a large role in the design of coded modulation systems. For LDPC binary code, Gray mapping has traditionally been used in both MLCM and BICM configurations. Gray mapping minimizes capacity loss when the one-shot receiver is applied [5, 48, 49, 50]. Other labelling schemes have been proposed for BICM-IDD systems [6, 51, 52, 53], however Gray mapping is still a good choice with LDPC outer codes [20]. In the MLCM-MSD structure, set partitioning
labelling was originally proposed in [7] and was later shown to have better performance than other labelling formats [8] for finite length transmissions. Set partitioning has a capacity advantage over over schemes under MSD decoder, but generally requires efficient subcodes to be designed over a wide range of rates.

Low Density Parity Check (LDPC) codes [15] have been the subject of great interest since their rediscovery in [16, 17] because of their Shannon capacity approaching performance with a simple belief propagation (BP) decoder. A conventional LDPC code is based on a sparse parity check matrix whose rows represent single parity check codes and whose columns represent repetition codes. Several generalized LDPC codes have been proposed [54, 55, 56, 57, 58, 59, 60] that replace the single parity check codes with other block codes.

Existing LDPC and Generalized LDPC (GLDPC) designs, with the exception of [60], have limitations in the low rate region. Low code rates lead to variable node performance that is relatively insensitive to changes in the SNR, thus incurring a significant SNR penalty to open a decoding tunnel. In addition, low rate LDPC codes have parity check matrices that are almost square, for which it is difficult to find high girth graphs at moderate code lengths because of the large relative number of check equations. The solution in [60] is to link the check node performance to the channel SNR by connecting additional parity bits directly to the channel. The Hadamard check node used in [60] has limited design options, since it is only suited for the very low rate range $R \leq 0.05$.

This chapter develops a set of irregular LDPC codes with generalized check nodes having improved low rate $r < 1/3$ performance. These codes can improve MLCM-MSD performance over a wide range of design rates when set partitioning labelling is used. Set partitioning labelling generates a large spread of subcode rates, thereby benefitting from improved low rate construction. It will be shown that mapping schemes generating low and very high rate subcodes have a rate advantage over Gray mapping, which produces concentrated subcode rates.
An LDPC code is generalized by extending the single parity codes with additional parity bits that are connected directly to the channel without repetition. For binary channels or virtual binary channels in a staged decoder, irregular code profiles can be found through optimization over a conventional 2-dimensional EXIT chart. For BICM-IDD designs, the variable and check node EXIT curves are mutually dependent, requiring several 2-dimensional EXIT chart optimizations within a fixed point iteration framework. Only check node regular designs are considered. The check node generator matrix is found through area maximization, also within an EXIT chart framework.

### 2.2 Multilevel Coded Modulation Design

This section examines MLCM-MSD design for finite length construction. Consider a modulated symbol \( s \) that has a symbol alphabet of size \( 2^L \), that is formed according to \( s = m(b) \), where \( b \in \mathbb{B}^L \). The symbol is passed through an AWGN channel according to

\[
y = s + n
\]  

(2.1)

where \( n \) is a complex AWGN noise source with variance \( \sigma_n^2 \). The set subcode rates is determined through a decomposition of the constrained modulation (CM) capacity, which is given by

\[
C_{CM} = I(s; y) = s - E_{s,y} \left( \log_2 \frac{\sum_{s' \in S} p(y|s)}{p(y|s)} \right) 
\]  

(2.2)

\[
= L - E_{s,y} \left( \log_2 \frac{\sum_{s' \in S} \exp(||y - s||^2/(2\sigma_n^2))}{\exp(||y - s||^2/(2\sigma_n^2))} \right) 
\]  

(2.3)

where \( S \) is the set of possible transmitted symbols, and equiprobable transmitted symbols and perfect receiver CSI is assumed. The CM capacity can be decomposed into \( L \) parallel subchannels using the chain rule of mutual information as:

\[
I(b; y) = I(b_1; y) + I(b_2; y|b_1) + \ldots + I(b_L; y|b_1, \ldots, b_{L-1}) 
\]  

(2.4)
The capacity of each subchannel is \( C_l = I(b_l; y|b_1, \ldots, b_{l-1}) \), provided that lower level subchannel bits have been decoded correctly. The decomposition in (2.4) implies that if an MLCM encoding is used with code rates that are matched to the subchannel capacities \( r_l = C_l \), a multistage decoding strategy can be applied with capacity achieving performance [7, 8] for ideal subcodes. For practical codes, a subcode rate less than the corresponding capacity must be used to ensure the desired level of FER performance.

A detection error is declared if there is an error in any level, and detection errors appear relatively independently across subchannels under correct decision feedback. Let \( f^N_{\text{FER}}(r, C) \) be the FER of a GLDPC code of length \( N \) and rate \( r \) transmitted across a binary AWGN channel of capacity \( C \). The optimal set of rates is found by the condition

\[
\begin{align*}
{f^N_{\text{FER}}(r_i, C_i)} &= {f^N_{\text{FER}}(r_j, C_j)} \quad \forall i, j
\end{align*}
\]

whose optimality is proven in [8, 61].

The total capacity loss \( \sum_l (C_l - r_l) \) is affected by the mapping function that determines the set of subchannel rates, and the FER function \( f^N_{\text{FER}}(r, C) \). The effect of mapping on capacity loss was characterized in [8] using random error exponents, and set partition was found to outperform other labelling schemes. The analysis in [8] does not, however, give much insight into how capacity loss is affected by the properties of the mapping function. Consider the relationship between rate-loss \( \Delta r \), which is the rate penalty paid to sustain a desired level of performance, and power-loss \( \Delta P \), which is the addition transmission power required to sustain that desired level of performance. The entities are approximately related by derivative approximation \( \Delta P \approx \frac{\delta P}{\delta r} \Delta r \) or \( \Delta r \approx \Delta P/\frac{\delta P}{\delta r} \), where \( \frac{\delta P}{\delta r} \) is the derivative of the power threshold function \( P(r) \), which is the minimum SNR required to support at transmission rate \( r \). The threshold function and its derivative are shown in Figure 2.1. One can see that the derivative function is lowest in the region \( r = [0.4, 0.8] \) and increases sharply in the low rate region \( r < 0.25 \) and the very high rate region \( r > 0.9 \). In the low and very high rate regions, a small subchannel rate-loss leads to a large increase in the effective SNR of the corresponding virtual channel based on
Chapter 2. Generalized Low Density Parity Check Codes

Figure 2.1: Plot of the inverse MI function (top) and its derivative (bottom) as a function of rate for the binary AWGN channel

the derivative approximation. In other words, the power penalty incurred to sustain a certain level of FER performance translates into a smaller capacity loss in the low and very high rate regions. It is then advantageous to use mapping functions that generates subchannel capacities in these regions.

Labelling functions and detection orders that lead to a large spread of subchannel capacities have a smaller rate-loss than those yielding concentrated capacities. The set partitioning strategy in [7, 8] maximizes the minimum intra-set distance leading to a large spread of subcode rates. For an QAM modulation, set partitioning reduces to natural labelling with a least significant bit first detection order. For the AWGN channel considered in this chapter, a single code can be mapped to the same bit positions in the in-phase and quadrature-phase components, thereby doubling the code block length. For fading channels that are considered in later chapters, the in-phase and quadrature-phase signal components may experience different degrees of fading, and it is useful to map separate codes to each signal component.
2.3 GLDPC Code Structure

A GLDPC code is encoded by first encoding an LDPC codeword $c_L$ of length $N$ using a parity check matrix $H_{LDPC}$. Only check node regular designs are considered with $d_c$ ones in each column of $H_{LDPC}$. Each check equation forms a $(d_c, d_c - 1)$ single parity check code, where the designated position of the parity bit can be chosen arbitrarily. Choosing the last bit position for the parity bit, the check node generator matrix is $G = [I_k, 1_{k 	imes 1}]$. A GLDPC code is formed by extending the single parity check generator matrix with an additional $q_c$ columns. The new parity bits are connected directly to the channel without repetition through the LDPC code graph. A GLDPC codeword is structured as $c_G = [c_L, c_C]$, where $c_C = f_C(c_L, H_{LDPC})$ is the set of $N_C = Mq_c$ degree-1 channel bits generated from the check node constraints. The $m^{th}$ block of check node bit is produced according to $[c_{C1+q_c(m-1)}, \ldots, c_{Cq_c,m}] = c_L^{(i:h_{mi}=1)}[1 \ldots d_c - 1][g_{k_c+1}, \ldots, g_{n_c+1}]^T$, where $c_L^{(i:h_{mi}=1)}[1 \ldots j]$ represents the elements of $c_L$ corresponding to the first $j$ ones in the $m^{th}$ row of the parity check matrix $H_{LDPC}$, and $G = [g_1, \ldots, g_{k_c+1}]^T$ is the check node generator matrix. The generator is assumed to be in standard form $G = [I_k, P]$, with the first column of $P$ consisting of all ones so that the systematic bits and first parity check bit can be discarded, as they are already part of the LDPC codeword.

The Tanner graph of the proposed GLDPC code can be obtained from an LDPC code graph by replacing the original check node with a super check node. Consider the graph representation of the GLDPC code in Figure 2.2. The graph consists of degree-1 variable nodes, non-degree-1 variable nodes and super check nodes. The degree-1 variable nodes correspond to channel bits connected directly to the super check node without repetition, while the non-degree-1 variable nodes correspond to channel bits that are repeated in the LDPC code graph. Let $N$ be the number of variable nodes and $M$ be the number of super check nodes. The variable nodes are irregular with the $n^{th}$ variable node having $d_{vn}$ check node connections and the set of non-degree-1 variable nodes having a degree profile $\lambda_i = N_{dv}$. The regular super check nodes have $d_c$ variable node connections and
Chapter 2. Generalized Low Density Parity Check Codes

$q_c$ degree-1 channel connections. The addition of unrepeated super check node parity bits does not change the interconnection structure between the check and variable nodes, and the total number of edges between check and non-degree-1 nodes is $n_e = Md_c$. The total number of constraints is $N_C = \frac{n_c(q_c+1)}{d_c}$, and the total number of degree-1 and non-degree-1 variable nodes is $N_{VN} = \sum_{i=1}^{N_{dv}} \frac{n_c \lambda_i}{i} + \frac{n_c d_c}{d_c}$. Finally, the GLDPC code rate is given by

$$R = 1 - \frac{N_C}{N_{VN}} = 1 - \frac{q_c + 1}{d_c \sum_{i=1}^{N_{dv}} \frac{\lambda_i}{i} + q_c}$$

(2.6)

A non-zero $q_c$ adds degree-1 variable node connections to the super check node, bringing the second term in (2.6) closer to unity, and lowering the overall code rate.

GLDPC codes have the same bipartite Tanner graph as LDPC codes when the channel
nodes are omitted. It follows that GLDPC codes can be decoded using a BP decoder with different check node update rules to account for the addition channel connections. The GLDPC graph in Figure 2.2 has three kinds of connections: variable-to-check node, variable-to-channel node, and check-to-channel node. Let $U_{v\rightarrow c}$, $U_{ch\rightarrow v}$ and $U_{ch\rightarrow c}$ denote the set of edges between variable and check nodes, channel and variable nodes, and channel and check nodes, respectively. These three types of bidirectional edges can be further divided into six types of unidirectional edges denoted, $U_{v\rightarrow c}$, $U_{c\rightarrow v}$, $U_{ch\rightarrow v}$, $U_{v\rightarrow ch}$, $U_{ch\rightarrow c}$, and $U_{c\rightarrow ch}$. Associated with each type of edge is a message type and an update rule for that message type. For the type of $v \rightarrow c$ edges, let $v_n$ be the set of $n_{vn}$ extrinsic messages coming from variable node $n$ going to check nodes, and let $\tilde{v}_m$ be the set of $n_c$ variable node extrinsic messages going to check node $m$. Since there is only one type of edge from variable to check nodes, we have $\{v_n\} = \{\tilde{v}_m\}$. For the $c \rightarrow v$ edge type, let $u_m$, $\tilde{u}_n$ be the set of extrinsic messages coming from check node $m$ and going to variable node $n$. For the $U_{ch\rightarrow v}$ edge types, let $x_n$, $\tilde{w}_n$ be the extrinsic message coming from and going to the $n^{th}$ variable node. Similarly, for the $U_{ch\rightarrow c}$ edge types, let $z_m$, $\tilde{y}_m$ be extrinsic message vectors of size $q_c$ coming from and going to the $m^{th}$ check node.

The update rule for the channel nodes depends on the coded modulation structure. Based on a channel observation $y$, an extrinsic output is produced for the $l^{th}$ bit in the binary symbol vector $b$ according to

$$\lambda^E[b_l] = \log \frac{\sum_{b,b_l=1} \exp \left( -\frac{\|y-m(b)\|^2}{\sigma_n^2} \right) + \sum_m \log \frac{b_m e^{\lambda^I[b_m]} + b_m}{1 + e^{\lambda^I[b_m]}}}{\sum_{b,b_l=0} \exp \left( -\frac{\|y-m(b)\|^2}{\sigma_n^2} \right) + \sum_m \log \frac{b_m e^{\lambda^I[b_m]} + b_m}{1 + e^{\lambda^I[b_m]}}} - \lambda^I[b_l] \quad (2.7)$$

where $\lambda^I[b_l]$ is the intrinsic input for bit $b_l$. In an MLCM-MSD scheme, each bit is mapped to a separate code. During the decoding of the $l^{th}$ stage, a constant message is sent from corresponding channel nodes that is based on channel observations and previously decoded bits. In the case of the IDD receiver, the channel message is updated every iteration or at some larger period. In addition, the channel nodes have $L$ connections instead of the single channel node connections shown in Figure 2.2. These channel node
connections are randomly assigned to variable and check nodes. For either the MSD or IDD receiver structure, the channel updates \( \{ \tilde{w}_n \}, \tilde{y}_m \) are based on (2.7). Following the message-passing algorithm in [18], the \( l^{th} \) to \( (l+1)^{th} \) stage update equations for the four remaining message types are given as follows:

\[
\begin{align*}
    v_n^l &= \sum_{i=1}^{d_{vn}} \tilde{u}_{mi}^l + \tilde{w}_n - \tilde{u}_n^l \\
    x_n^l &= \sum_{i=1}^{d_{vn}} u_{ni}^l \\
    u_{mi}^{l+1} &= \frac{\sum_{b \in B^{k_{vn}}} \exp(-\frac{1}{2} b G[v_m^l, \tilde{y}_m^l])}{\sum_{b \in B^{k_{vn}}} \exp(-\frac{1}{2} b G[v_m^l, \tilde{y}_m^l])} - \tilde{v}_mi^l \\
    z_{mi}^{l+1} &= \frac{\sum_{b \in B^{k_{vn}}} \exp(-\frac{1}{2} b G[v_m^l, \tilde{y}_m^l])}{\sum_{b \in B^{k_{vn}}} \exp(-\frac{1}{2} b G[v_m^l, \tilde{y}_m^l])} - \tilde{y}_mi^l 
\end{align*}
\]

(2.8)

with \( v_n^0 = 0, x_n^0 = 0 \) \( \forall n \).

### 2.4 GLDPC Code Design and Analysis through EXIT Charts

There are six message densities whose evolution under iterative message passing needs to be analyzed. Let \( I_{a \rightarrow b} \) be the average mutual information (MI) conveyed by messages coming from node type \( a \) going to node type \( b \), which is referred to as an extrinsic MI \( I_{a \rightarrow b}^E \) in the update of \( a \) and an intrinsic MI \( I_{a \rightarrow b}^E \) in the update of \( b \). The MI for the six message densities are denoted, \( I_{ch \rightarrow v}, I_{ch \rightarrow c}, I_{v \rightarrow c}, I_{v \rightarrow ch}, I_{c \rightarrow v}, I_{c \rightarrow ch} \). The evolution of these message densities can be analyzed using a 2-dimensional EXIT chart [19, 20] in a fixed point iteration optimization involving the MI functions of all the densities. These MI functions can be derived by expressing the node update equations in terms of mutual information. For the channel nodes, it can be observed that the extrinsic MI is the same for channel-to-variable and channel-to-check node messages, provided that channel degree-1 bits and non-degree-1 bits are randomly distributed in the symbol vectors. The
extrinsic channel MI can be expressed as

\[ I_{\text{ch} \rightarrow v}^E = I_{\text{ch} \rightarrow c}^E = f_{\text{ch}}(\overline{I}_c^I) \tag{2.9} \]

where \( \overline{I}_c^I = (1 - \alpha D_1)I_{v \rightarrow c}^I + \alpha D_1 I_{c \rightarrow v}^I \) is the average detector intrinsic MI, and \( \alpha D_1 \) is the fraction for degree-1 bits in the GLDPC codeword. For the binary AWGN channel, the channel message MI can be found in terms of the noise variance as \( f_{\text{ch}}(\overline{I}_c^I) = J(4/\sigma_Z^2) \) \[19\], where

\[
J(\sigma_Z^2) = 1 - \int \frac{\exp\left(-\frac{(x - \sigma_Z^2/2)^2}{2\sigma_Z^2}\right)}{\sqrt{2\pi}\sigma_Z} \log_2[1 - e^{-x}] \, dx \tag{2.10}
\]

IN coded modulation systems, \( f_{\text{ch}}(\overline{I}_c^I) \) is determined through Monte Carlo simulations.

For the variable nodes, the MI transfer functions can be derived from the first two update equations in (2.8) as

\[
I_{\text{v} \rightarrow c}^E = f_{\text{v} \rightarrow c}(I_{\text{c} \rightarrow v}^I, I_{\text{ch} \rightarrow v}^I, \{\lambda_j\}) = \sum_j \lambda_j J((j - 1)J^{-1}(I_{v \rightarrow c}^I) + J^{-1}(I_{c \rightarrow v}^I))
\]

\[
I_{\text{v} \rightarrow \text{ch}}^E = f_{\text{v} \rightarrow \text{ch}}(I_{\text{c} \rightarrow v}^I, \{\lambda_j\}) = \sum_j \lambda_j J(jJ^{-1}(I_{\text{c} \rightarrow v}^I)) \tag{2.11}
\]

It will also be useful to represent the inverse functions MI function as \( I_{\text{c} \rightarrow v}^I = f_{\text{c} \rightarrow \text{v}}^{-1}(I_{\text{v} \rightarrow c}^E, I_{\text{ch} \rightarrow v}^I, \{\lambda_j\}) \).

The check node MI transfer functions are expressed as

\[
I_{\text{c} \rightarrow v}^E = f_{\text{c} \rightarrow \text{v}}(I_{\text{v} \rightarrow c}^I, I_{\text{ch} \rightarrow c}^I)
\]

\[
I_{\text{c} \rightarrow \text{ch}}^E = f_{\text{c} \rightarrow \text{ch}}(I_{\text{v} \rightarrow c}^I, I_{\text{ch} \rightarrow c}^I) \tag{2.12}
\]

Even though the analysis of the proposed GLDPC codes involves more message types than for LDPC codes, the optimization of the variable node degree profile remains the same when the remaining MI functions are fixed. Following the LDPC code EXIT chart analysis in [20], the variable degree is chosen to maximize the code rate constrained by the condition that the variable-to-check MI function is greater than the inverse check-to-variable MI function. This condition is graphically represented by plotting the variable-to-check MI function on the \( X - Y \) axis, plotting the check-to-variable MI function on the \( Y - X \) axis, and ensuring a gap or decoding tunnel exits between the two curves.
Chapter 2. Generalized Low Density Parity Check Codes

Under the assumption the message densities are Gaussian, the area under the variable-to-check MI function is the CM channel capacity, and the area above the check-to-variable MI function is the code rate. The Gaussian assumption often slightly overestimates the channel capacity, but greatly simplifies the density evolution analysis. A decoding tunnel between the two EXIT curves ensures error free transmission under the assumption that the combined LDPC and channel graph is loop free. The optimal degree distribution \( \{\lambda_i\}^* \) can be found as a solution to the following optimization problem:

\[
\arg \min_{\{\lambda_i\}} f_R(\{\lambda_i\}) \quad \text{s.t.} \quad \sum_i \lambda_i = 1, \quad f_{c\rightarrow v}(I_{v\rightarrow c}, I_{ch\rightarrow c}) > f_{v\rightarrow c}^{-1}(I_{v\rightarrow c}, I_{ch\rightarrow v}) \forall I_{v\rightarrow c} \quad (2.13)
\]

where \( f_R(\{\lambda_i\}) \) is the rate function in (2.6) evaluated as a function of \( \{\lambda_i\} \). The problem in (2.6) can be solved by one of several constrained cost minimization procedures [62], provided that \( I_{ch\rightarrow c}, I_{ch\rightarrow v} \) are constant or evaluated as a function of \( I_{v\rightarrow c} \). For the AWGN channel, \( I_{ch\rightarrow c} = I_{ch\rightarrow v} = J(4/\sigma_n^2) \) is constant, and the constraint curves \( f_{c\rightarrow v}(I_{v\rightarrow c}, I_{ch\rightarrow c}) \) and \( f_{v\rightarrow c}^{-1}(I_{v\rightarrow c}, I_{ch\rightarrow v}) \) can be evaluated independently of each other.

The optimization in (2.13) can be solved using a gradient search method, such as medium scale algorithm in the Matlab optimization toolbox. The Matlab implementation uses a sequential quadratic program to solve the Kuhn-Tucker equations, which balance the gradient of the objective and constraint functions at an optimal point [63]. Purely gradient based methods are fast, but may converge to local minima far from the optimal profile or have difficulty in flat regions of the objective function. One way to improve the convergence behaviour of gradient based solutions (2.13) is to observe that at the optimal degree profile, \( f_{c\rightarrow v} \) and \( f_{v\rightarrow c}^{-1} \) are almost equal, except for possibly the region of the EXIT curve near \( I_{v\rightarrow c} = 0 \). The search space of possible degree profiles can be significantly reduced by putting an additional upper bound constraint on (2.13) given by

\[
f_{c\rightarrow v}(I_{v\rightarrow c}, I_{ch\rightarrow c}) < f_{v\rightarrow c}^{-1}(I_{v\rightarrow c}, I_{ch\rightarrow v}) + \delta_{EXIT} \forall I_{v\rightarrow c} \in [\alpha_{EXIT}, 1] \quad (2.14)
\]

where \( \alpha_{EXIT}, \delta_{EXIT} > 0 \) are the range and tolerance parameters. We set \( \alpha_{EXIT} = 0.2 \) and evaluate the design rate for difference values of \( \delta_{EXIT} \), choosing the one that maximizes
the design rate while keeping the constraints below a certain threshold.

In BICM-IDD systems, the two sides of the second constraint in (2.13) are coupled by the channel node MI, and the variable degree profile must be jointly analyzed with the remaining MI functions. Let $y = I_{v-c}$ be the independent variable typically plotting on the y-axis of the EXIT chart and let the solution to (2.13) be represented in functional form as,

$$\{\lambda_i^*\} = f_{opt}(I_{v-c}, I_{c-v}, I_{ch-v})$$  \hspace{1cm} (2.15)

The optimal degree profile can be found from (2.15), provided that $I_{c-v}(y)$ and $I_{ch-v}(y)$ can be expressed in terms of $y$. The entities $I_{c-v}(y)$ and $I_{ch-v}(y)$ are, however, dependent on the degree profile through equations (2.11) and (2.9). The MI update equations (2.9), (2.11) and (2.12), and the degree profile optimization function can be iteratively applied with the objective of finding a steady-state relationship between the functions $I_{c-v}(y)$, $I_{c-v}(y)$, $I_{ch-v}(y)$, $I_{ch-c}(y)$ and $I_{v-ch}(y)$ after some number of iterations. The update equations for the MI functions for the $k^{th}$ iteration, $k \geq 1$ are given by,

$$\{\lambda_j^k\}^{k+1}(y) \leftarrow f_{opt}(y, I_{c-v}^k(y), I_{ch-v}^k(y))$$

$$I_{c-v}^{k+1}(y) \leftarrow f_{c-v}^{-1}(y, I_{ch-v}^k(y), \{\lambda_j^k\})$$

$$I_{c-ch}^{k+1}(y) \leftarrow f_{c-ch}(y, I_{ch-c}^k(y))$$

$$I_{v-ch}^k(y) \leftarrow f_{v-ch}(I_{c-v}^k(y), \{\lambda_j^k\})$$

$$I_{ch-v}^k(y) = I_{ch-c}^k(y) \leftarrow f_{ch}((1 - \alpha)I_{v-ch}^k(y) + \alpha I_{c-ch}^k(y))$$  \hspace{1cm} (2.16)

For the first iteration, the second and fifth update equations in (2.16) are dependent on the degree profile, which is not available at startup. The effectiveness of the fixed point update equations is dependent on the initial choices for the MI functions. In order to find a good initial estimate for the MI functions, we run the MI function update equations in (2.16) without updating $\{\lambda_j^k\}^k(y)$ for a few iterations, replacing the second and fifth
update equation in (2.16) with

\[
I_{c\rightarrow v}^{k+1}(y) \leftarrow f_{c\rightarrow v}(y, I_{ch\rightarrow v}^k(y), \{\lambda_j\}^k)
\]

\[
I_{v\rightarrow ch}^k(y) \leftarrow I_{c\rightarrow v}^k(y)
\]  

(2.17)

For the first equation in (2.17), we have used the fact that when a good code profile is found, \(f_{c\rightarrow v} = f_{v\rightarrow c}^{-1}\) because of the curve matching property, and in the second equations we have omitted the function \(f_{v\rightarrow ch}\). Although in general, \(f_{v\rightarrow ch}(y) \neq y\), channels that are well suited to LDPC codes have relatively flat EXIT curves, and the channel MI function is only slightly affected by a change in the \(f_{v\rightarrow ch}(y)\) function.

The operation of the MI function update equations in (2.16) can be visualized through an EXIT chart example. Consider a GLDPC coded BICM transmission over an AWGN channel using 16-QAM modulation, Gray labelling, and a GLDPC code rate of \(R = 0.25\), with check node parity check matrix

\[
P = \begin{bmatrix}
1 & 0 & 1 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 \\
0 & 1 & 1 & 0 & 0 & 1 \\
1 & 1 & 0 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 1
\end{bmatrix},
\]

(2.18)

and variable degree profile \(\lambda(x) = 0.156x^2 + 0.738x^3 + 0.0634x^4 + 0.0146x^9 + 0.0146x^{10} + 0.129x^{11}\) found using (2.13) that leads to threshold SNR of 0.48dB, which is the minimum SNR for which the EXIT constraints are satisfied at the given code rate. The channel extrinsic message MI function \(I_{ch\rightarrow v}(x)\) and the variable-to-check node MI function \(I_{v\rightarrow c}(x)\) are plotted in Figure 2.3 on the y-axis against \(x = I_{c\rightarrow v}\) on the x-axis. Also shown in Figure 2.3, is the channel extrinsic message MI \(I_{ch\rightarrow c}(y)\) and the variable-to-check node MI function \(I_{v\rightarrow c}(y)\) on the x-axis against \(y = I_{c\rightarrow v}\) on the y-axis. The channel MI functions seen by the variable and check nodes have the same start and end points, with the one
corresponding to the check node showing a steeper slope near \((x, y) = (0, 0.2)\) because of the discontinuous slope of the \(f^{\text{v}}_{\text{c} \rightarrow \text{v}}(y)\) near \(y = 0.2\). From a graphical perspective, \(I_{\text{ch} \rightarrow \text{v}}(x)\) is reflected into \(I_{\text{c} \rightarrow \text{v}}(x)\) to form \(I_{\text{ch} \rightarrow \text{c}}(y)\). This effect is more visible if a smaller \(q_c\) is used since this leads to more curvature in the \(I_{\text{c} \rightarrow \text{v}}(x)\) function. The similarity between the channel MI functions when plotted on the x-axis and y-axis indicates that fixed iteration procedure should not have difficulty converging.

\section*{2.5 Generalized Check Node Design}

Thus far a design procedure has been developed based on EXIT chart analysis to determine the variable node degree profile based on the type of super check node and the channel characteristic. The performance of the GLDPC code is then dependent on the number of degree-1 and non-degree-1 connections at each check node and the choice of the check node generator matrix. A super check node with \(d_c\) non-degree-1 connections
and \( q_c \) degree-1 connections is formed by a \((q_c + dc, d_c - 1)\) block code with a generator matrix \( G = [I_k, P] \), with the first column of \( P \) consisting of all ones. When the number of degree-1 and non-degree-1 connections has been fixed, what remains is determining the last \( q_c \) columns of the parity check matrix \( P \).

EXIT chart can be useful in finding a good super check node generator matrix, especially since the overall code performance is analyzed using EXIT charts in the previous section. There are a couple of factors that influence super check node design. The first is how closely the check node EXIT curve can be matched by repetition code curve. The mismatched area is direct rate-loss. The second factor has to do with how the check node EXIT curve shifts with a change in channel SNR and is determined by the number of channel connections \( q_c \). A large value of \( q_c \) gives rise to a check node MI function that is sensitive to the channel SNR, so a small power-loss is incurred to open a decoding tunnel of sufficient size to achieve error free transmission for finite packet lengths. Unfortunately, a large \( q_c \) value leads to curve matching problems that were pointed out in [60] and will be further discussed later in this section. In order to balance these two factors, it is necessary to determine \( P \) for different values of \( d_c \) and \( q_c \), simulate the FER performance for the packet length of interest, and choose the parity check matrix with the smallest FER.

The super check node in general has two MI functions, with \( I_{c\rightarrow v}^E = f_{c\rightarrow v}(I_{v\rightarrow c}^I, I_{ch\rightarrow c}^I) \) determining the MI going to the variable nodes and \( I_{c\rightarrow ch}^E = f_{c\rightarrow ch}(I_{v\rightarrow c}^I, I_{ch\rightarrow c}^I) \) determining the MI going to the channel nodes. For the purposes of designing the check node generator, only the output \( I_{c\rightarrow v}^E \) is considered and not \( I_{c\rightarrow ch}^E \). For the binary AWGN channel or MLCM-MSD system, \( I_{c\rightarrow ch}^E \) is not used entirely, and for the BICM-IDD system, the variable node EXIT curve is much steeper than the channel EXIT curve, thus \( I_{c\rightarrow v}^E \) has a significantly larger effect on system performance than \( I_{c\rightarrow ch}^E \). Another simplifying assumption employed for check node generator design is that the intrinsic channel and variable node messages are independent, which is strictly the case for the binary AWGN
channel. For the BICM-IDD system, the assumption of independent inputs should not significantly affect the resulting check node parity check matrix if the mapping function is chosen to minimize symbol message correlation.

A family of $K$ codes can be formed by extending a $(d_c, d_c - 1)$ single parity check code through the addition of columns to the parity check matrix in a recursive manner. The first code in the set has parity check matrix $P_1 = 1_{(d_c-1) \times 1}$. The $l^{th}$ matrix is formed from the $(l-1)^{th}$ matrix according to $P_l = [P_{l-1}, p_l]$, where $p_l \in \mathbb{B}^{d_c-1}$. What is required is a criterion to determine $p_l$ from $P_{l-1}$. From the EXIT chart in Figure 2.3, convergence is predicted if the check node curve stays below the variable node curve. This suggests that maximizing the area to the right of the check node curve should lead to a low convergence SNR. One complication is that as the design rate and $q_c$ increases, the check node curve tends to lift away from the y-axis at the origin and cannot be closely matched by the variable node curve. The area under the check node curve $f_{c-v}(y, I_{ch-c}^I)$ near $y = 0$ is not recoverable by a repetition code, and so the domain of $y$ can be limited to $y \in [\beta, 1]$, $0 < \beta < 1$. Let $f_{c-v}^E(I_{v-c}^I, I_{ch-c}^I, P]$ and $f_{c-v}^{APP}(I_{v-c}^I, I_{ch-c}^I, P]$ be the check node extrinsic and a posteriori output, respectively. For a previous parity check matrix $P_{l-1}$, the current parity vector $p_l$ can be found using the following selection criterion:

$$\arg\max_{p \in S_{Ad}(P_{l-1})} \int_{\beta}^{1} f_{c-v}^E(y, I_{ch-c}^I, [P_{l-1}, p]) + f_{c-v}^{APP}(y, I_{ch-c}^I, [P_{l-1}, p])dy \quad (2.19)$$

where $S_{Ad}(P) = \mathbb{B}^{d_c-1} - \text{col}(P) - \text{cols}(I_k)$ is the set of admissible $p$ vectors and col($X$) represent the set of column vectors forming $X$. Both the extrinsic and APP functions are averaged in (2.19). The gap between the APP and extrinsic functions is greatest near the middle of the $[0, 1]$ domain, thus the inclusion of the APP function in (2.19) places more weight on the area under the extrinsic curve near the center of the domain, which results in slightly better performance. For a channel message MI of $I_{ch-c}^I = 0.25$, the recursive application of (2.19) yields a family of check nodes whose transfer characteristics are shown in Figure 2.4 (left), for $d_c = 7$, $q_c = [0 \ldots 6]$ and $\beta = \{0, 0.5\}$. The parity check matrices used to generate the first six curves in Figure 2.4 (left) are a subset of the matrix
in (2.18) for the $\beta = 0.5$ case. Shown in Figure 2.4 (right) is the set of repetition code transfer functions for $d_v = [2\ldots8]$ generated at the same channel MI. One can see that the check and variable node curves have complementary shapes for the $\beta = 0.5$ case, but the check node curves lift away from the y-axis for the $\beta = 0$ case leading to a rate-loss.

In the $\beta = 0.5$ case, a super check node EXIT curve can be matched with a low rate repetition code, provided that the number of degree-1 check node bits is not too large.

The choice of the admissible set of vectors in (2.19) used to generate the $k^{th}$ code in the family is worth some discussion. Vectors corresponding to the non-degree-1 bits and previously selected degree-1 bits are removed from the admissible set in (2.19) in order to minimize the rate-loss in the region near $y = 0$. The repetition of a non-degree-1 bits as a degree-1 bits will cause the check node MI function to be more heavily influenced by $I_{ch\rightarrow c}^I$. Since $I_{ch\rightarrow c}^I$ is unchanged over $y$, the repetition of a non-degree-1 bit will raise the check node MI function all over, including the region near $y = 0$ that is unrecoverable. Allowing the repetition of degree-1 bits has a similar, but lesser effect.

The family of check node codes generated using (2.19) is dependent on the channel
SNR through $I_{ch \rightarrow c}$, and so different parity check matrices can be generated that are suited to various SNR ranges. The check node MI functions $f_{E}^{c \rightarrow v}(I_{v \rightarrow c}, I_{ch \rightarrow c}, P)$ and $f_{E}^{c \rightarrow ch}(I_{v \rightarrow c}, I_{ch \rightarrow c}, P)$ can be evaluated using Monte Carlo simulation and tabulated for different $P$ over the input space $I_{v \rightarrow c}, I_{ch \rightarrow c} \in [0, 1]$. Using the tabulated values, the check node MI functions can be efficiently evaluated for an arbitrary $I_{v \rightarrow c}, I_{ch \rightarrow c}$ using 2-dimensional interpolation.

2.6 Code Graph Construction with Cycles

Thus far, GLDPC code design has been discussed without consideration to how the LDPC code graph is constructed, or how cycles or loops in the graph affect performance. Loops in the GLDPC graph, or the combined channel and GLDPC code graph, have a negative effect on BP decoder performance. This is because loops lead to a correlated intrinsic input and an overconfident extrinsic output that will degrade decoding performance. For long codes, the code graphs can be efficiently built from randomly constructed graphs with near capacity achieving performance [16, 18], since the average minimum distance between nodes in the graph is large. The performance of random code construction is often improved by removing loops of length 4 in a post-processing step.

When constructing short-to-moderate length codes, more attention has to be paid to the distribution of loops within the code graph, especially short loops which should be avoided. The Progressive Edge Growth (PEG) algorithm constructs the code graph in an edge-by-edge manner with the objective of minimizing the graph girth, or shortest cycle in the graph [64]. The PEG algorithm maintains a set of connected nodes with minimum distance $d_{min}$. At each step, it forms a spanning tree for the set of connected nodes with the aim of finding a new connection whose minimum distance to the connected set is $d_{min}$ or more. If this is not possible, a connection is made that maximizes the new $d_{min}$. Variations on the PEG algorithm have been proposed [65, 66], that use slightly
different selection criteria when there are several candidate nodes with the same $d_{\text{min}}$. These modifications to the approach in [64] offer a slight performance improvement in the high SNR region. The PEG algorithm from [64] has been used to design the code graphs for the GLDPC codes developed in this chapter.

A graph’s girth is a good indication of the performance gap between finite length code performance and the theoretical EXIT chart predicted performance under the cycle free assumption. For a fixed code rate, graph girth for PEG and other construction methods increases with code length and decreases with both check and variable node degrees. For a fixed number of information bits, the maximum graph girth increases with the code rate. To see this, consider an irregular rate $r = 0.01$ LDPC code of length 1000, with 990 checks, which certainly containing loops of length 4. In comparison, the PEG algorithm can produce a graph of girth 10, for a $r = 0.5$ regular LDPC code of the same length. The relationship between girth and code rate is significant for GLDPC codes that are based on a higher rate LDPC subcode. The rate $r = 0.25$ GLDPC design example in Figure 2.3 is based on a rate $R = 0.678$ LDPC subcode. For a codeword length of 8000, the PEG algorithm produces a GLDPC code graph of girth 8, which compares to a girth of 4 for a similar LDPC code design. In comparison to low rate LDPC codes, GLDPC codes can be designed with larger girths, and enjoy a smaller performance loss from the predicted threshold.

The graph produced by the PEG algorithm does not account for loops induced by channel message correlation under iterative detection and decoding. Correlated extrinsic channel messages introduce smaller loops in the combined code and channel graph that cannot be easily avoided using a graph construction procedure like PEG. For example, consider a rate $R = 1/2$ LDPC code that is mapped to a symbol with two bits. If the symbol mapping is treated as a constraint, there are as many constraints as codeword bits, leading to linearly dependent parity check rows and loops of length four. Although loops induced by the channel graph cannot be avoided, its effect on the BP decoder
performance can be reduced using Gray mapping, which minimizes channel message correlation.

2.7 Results

The performance of the proposed GLDPC codes are evaluated in this section using Monte Carlo simulations. Trials were performed until 200 frame errors occurred, or until 10000 trails were completed. A maximum number of 200 iterations were performed to decode LDPC and GLDPC codes. The maximum variable degree was set based on the binary channel capacity $C$: $N_{dv} = 50, C < 0.7$ and $N_{dv} = 150, C \geq 0.7$. The maximum check node degree was set to $d_{c}^{max} \leq 8$, in order to limit overall complexity. An optimal set of check node generator matrices were produced at capacities $C = \{0.05, 0.1, 0.2, 0.3, 0.4\}$. If the channel capacity fell between two values, the generator corresponding to the larger capacity was used. The $\beta$ value used to find these generators were $\beta = 0$ for $C = 0.05$, $\beta = 0.25$ for $C = 0.1$ and $\beta = 0.5$ for the remaining capacities. The optimal parameters $d_{c}^{*}, q_{c}^{*}$ for the binary AWGN channel are shown in Table 2.1 for different codeword lengths $N_{c} = \{250, 1000, 4000\}$ and rates $r \in \{0.02, 0.05, \ldots, 0.35\}$. For rates in the range $r > .35$, the number of degree-1 connections was set to zero and the GLDPC code reduced to an LDPC code, with no restriction on $d_{c}$. The optimal LDPC check node degree $d_{c}^{LDPC*}$ is shown in Table 2.2 for $N_{c} = \{250, 1000, 4000\}$ and channel capacities $C \in \{0.05, \ldots, 0.95\}$.

Some analysis of the trends in Table 2.1 and Table 2.2 is needed. For LDPC codes, $d_{c}^{LDPC*}$ increases with rate and decreases slightly for shorter codeword lengths. The optimal $d_{c}^{*}$ parameter exhibits similar behaviour for GLDPC codes, except that the value of $d_{c}^{*}$ reaches a plateau at $d_{c}^{max} = 8$. The optimal number of degree-1 connections decreases with increasing rate and tends to zero at, or near, the rate where $d_{c}^{LDPC*} = d_{c}^{max}$. There is a slight anomaly in the range $r = \{0.4 \ldots 0.45\}$, where the conventional LDPC code with $d_{c} = 5$, marginally outperforms the GLDPC code with $d_{c} = 8, q_{c} = 1$. 
### Table 2.1: Optimal GLDPC check node degrees as function of the code rate and codeword length.

<table>
<thead>
<tr>
<th></th>
<th>( r )</th>
<th>0.02</th>
<th>0.05</th>
<th>0.1</th>
<th>0.15</th>
<th>0.2</th>
<th>0.25</th>
<th>0.3</th>
<th>0.35</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N_c = 250 )</td>
<td>( d_c )</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>( q_c )</td>
<td>21</td>
<td>21</td>
<td>11</td>
<td>8</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>( N_c = 1000 )</td>
<td>( d_c )</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>( q_c )</td>
<td>23</td>
<td>25</td>
<td>12</td>
<td>7</td>
<td>5</td>
<td>5</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>( N_c = 4000 )</td>
<td>( d_c )</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>( q_c )</td>
<td>25</td>
<td>25</td>
<td>11</td>
<td>7</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

### Table 2.2: Optimal LDPC check node degree as function of the code rate and codeword length.

<table>
<thead>
<tr>
<th></th>
<th>( r )</th>
<th>0.05</th>
<th>0.1</th>
<th>0.2</th>
<th>0.3</th>
<th>0.4</th>
<th>0.5</th>
<th>0.6</th>
<th>0.7</th>
<th>0.8</th>
<th>0.9</th>
<th>0.95</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N_c = 250 )</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>7</td>
<td>8</td>
<td>10</td>
<td>19</td>
<td>32</td>
<td>44</td>
<td></td>
</tr>
<tr>
<td>( N_c = 1000 )</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>9</td>
<td>12</td>
<td>20</td>
<td>32</td>
<td>68</td>
<td></td>
</tr>
<tr>
<td>( N_c = 4000 )</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>14</td>
<td>22</td>
<td>33</td>
<td>72</td>
<td></td>
</tr>
</tbody>
</table>
Figure 2.5: BER of GLDPC codes, LDPC codes and LDPC codes with Hadamard constraints over AWGN channel at rates $r = \{0.05, 0.1, 0.2, 0.3\}$ for transmission of $N_u = 2000$ uncoded bits

Since $d_{c}^{LDPC}$ increases monotonically with rate, increasing $d_{c}^{max}$ slightly expands the region where the check node generalization provides a performance advantage, while decreasing $d_{c}^{max}$ reduces this region. Unfortunately, the performance improvement that results from a larger $d_{c}^{max}$ comes at the cost of check node update complexity that grows exponentially with $d_{c}$.

The performance of the proposed GLDPC codes across the binary AWGN channel is shown in terms of BER in Figure 2.5 and FER in Figure 2.6, for different code rates using $N_u = 2000$ uncoded bits. Also shown for reference comparison is the performance of LDPC code at the same rates using the code construction method in [20], and a rate $r = 0.05$ LDPC code with Hadamard constraints [60]. Because of the wide range of rates, performance in Figures 2.5, 2.6 is measured in terms of the bit SNR, $E_b/N_0 = 2C(E_s/N_0)E_s/N_0$, instead of the symbol SNR $E_s/N_0$, where $C(E_s/N_0)$ is the unconstrained channel capacity function. This horizontally shifts the performance results...
Figure 2.6: FER of GLDPC codes, LDPC codes and LDPC codes with Hadamard constraints over AWGN channel at rates \( r = \{0.05, 0.1, 0.2, 0.3\} \) for transmission of \( N_u = 2000 \) uncoded bits by 1.75\( dB \) – 9.5\( dB \), based on the design rate. The performance advantage of the GLDPC codes over LDPC codes decrease with increasing rate. At a design rate of \( r = 0.05 \), the GLDPC code boasts a 2dB-2.5dB improvement over the LDPC codes depending on the error criteria, but this gain is cut down to about 0.5\( dB \) at a design rate of \( r = 0.3 \). This trend is not surprising since there are more degree-1 channel connections in the lower rate code. In fairness, it should also be mentioned that LDPC code performance in the very-low rate region could be improved with an irregular check node degree having some degree-2 nodes. The GLDPC codes also outperforms the LDPC code with Hadamard constraints in [60] at a design rate of \( r = 0.05 \). Higher rate LDPC code designs with Hadamard constraints were not presented in [60], and are likely of limited practical use.

The SNR threshold values for the different codes are shown in Table 2.3 at different design rates. Also shown is the SNR for which the design rate equals the channel capacity. The GLDPC code had lowest threshold values for all rates considered and the differences
Table 2.3: Threshold values $\frac{E_b}{N_0} (dB)$ for GLDPC codes, LDPC codes and LDPC codes with Hadamard constraints

<table>
<thead>
<tr>
<th>Rate</th>
<th>Chan</th>
<th>GLDPC</th>
<th>LDPC</th>
<th>LDPC Had</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>-1.44</td>
<td>-1.374</td>
<td>-1.124</td>
<td>-1.350</td>
</tr>
<tr>
<td>0.1</td>
<td>-1.32</td>
<td>-1.174</td>
<td>-1.043</td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>-0.96</td>
<td>-0.786</td>
<td>-0.670</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>-0.615</td>
<td>-0.374</td>
<td>-0.284</td>
<td></td>
</tr>
</tbody>
</table>

between the GLDPC and LDPC threshold diminishes with increasing rate. The relative difference between these thresholds does not accurately reflect performance differences. For example, the SNR gap between the LDPC code threshold and channel capacity is relatively independent of rate, whereas low rate performance is significantly hampered by problems opening a decoding tunnel and with low graph girth.

The effectiveness of using GLDPC and LDPC subcodes in MLCM systems is a function of the subchannel capacities and the rate-loss at these capacities. Since the number of coded bits is constant for all subchannels, it is necessary to analyze the rate performance of these codes for a fixed codeword length. The rate-loss for LDPC and GLDPC codes is shown in Figure 2.7 as a function of the binary codes rate $r \in [0.05, 0.95]$ for different codeword lengths $N_c = \{250, 1000, 4000\}$. The rate-loss of the LDPC codes stays relatively constant for rates $r < 0.5$ and then decreases with increasing rates. The slight irregularities in LDPC curves in the very low rate region are the result of having a regular check node. In addition, the slightly higher rate-loss at $r = 0.95$ for $N_c = 250$ indicates a performance limitation for short block length very high rate LDPC codes. The rate-loss of the GLDPC code tends to be small in the low rate region and largest around $r = 0.5$. Since Figure 2.7 is evaluated in terms of rate and not capacity, the rate below which the GLPDC code outperforms the LDPC code is influenced by the code length. The effective range of improvement for the GLDPC code is approximately $r < 0.4$ for $N_c = 4000$ and
Figure 2.7: Rate-loss of GLDPC and LDPC codes as a function of rate for $N_c = \{250, 4000\}$

$r < 0.45$ for $N_c = 250$.

The rate-loss can be converted into an equivalent power-loss and is plotted on a decibel scale in Figure 2.8. For a code length of $N_c = 250$, the power-loss generally decreases with rate, steeply in the boundary rate regions with a relatively flat middle region over $r = [0.25, 0.6]$. With increasing code length, the power-loss curve becomes flatter over a wider rate range. In the case $N_c = 4000$, there is a relatively constant power-loss of $1dB$ with a slight negative slope over $r = [0.1, 0.9]$.

The total rate-loss in an MLCM-MSD transmission is the sum of the subchannel losses, which can be evaluated by first determining the subchannel capacities through Monte Carlo simulation and then computing the rate-loss using tabulated data. The subchannel capacities sum to the CM capacity. Three ASK mapping schemes are considered: natural, Gray and set partitioning [8]. Over an AWGN channel, the total rate-loss for a 16-QAM scheme using GLDPC and LDPC subcodes is shown in Figure 2.9 for subcodes of length $N_c = 250$ on the top, and $N_c = 4000$ on the bottom. The rate loss
Figure 2.8: Power-loss of GLDPC and LDPC codes as a function of rate for $N_c = \{250, 4000\}$

for the 256-QAM case is shown in Figure 2.10. The target level of performance was set to $FER = 0.01$ and an SNR range of $[-10, 20]$ and $[-10, 25]$ was used to generate the capacities in Figure 2.9 and Figure 2.10, respectively.

For all mapping formats, the GLDPC coded MLCM-MSD system had a measurably lower rate-loss than the LDPC coded system for all capacities, except for the range within about 20% of the maximum transmission rate. This is in contrast to the binary AWGN channel where the GLDPC performance gains are only in the low rate region $r < 1/3$. The set partition mapping strategy consistently outperforms the other labelling schemes when the proposed GLDPC codes are used. With LDPC subcodes, the different labelling schemes have inconsistent performance in the low rate region because of problems with conventional LDPC code construction. For equiprobable QAM signaling, it is generally advantageous to choose the modulation so that system does not operate in the low rate region with respect to the constrained modulation capacity. At a capacity of $C = 1$, the 16-QAM and 256-QAM systems incur a capacity loss of 0.1 and the 0.12, respectively. The
Figure 2.9: Rate-loss for 16-QAM MLCM-MSD transmission using GLDPC (top) and LDPC (bottom) subcodes and different labelling functions for packet lengths of $N_c = \{250, 4000\}$
Figure 2.10: Rate-loss for 256-QAM MLCM-MSD transmission using GLDPC (top) and LDPC (bottom) subcodes and different labelling functions for packet lengths of $N_c = \{250, 4000\}$
Figure 2.11: FER of LDPC and GLDPC coded MLCM-MSD systems over AWGN channel using 16-QAM

larger loss in the 256-QAM case results from low capacity subchannels that lead to zero-rate subcodes. A zero-rate subcode is equivalent to encoding a pseudo-random binary sequence which is quite power inefficient, since this subchannel increases the symbol energy without providing any additional transmission rate.

The performance of the 16-QAM MLCM-MSD system using LDPC and GLDPC subcodes is shown in Figure 2.11 for transmissions of $N_s = 2000$ symbols with a throughput of $r = 1, 2$ bits per channel use. The use of GLDPC codes improved system performance by $0.7\, dB$ in the $r = 1$ system and $0.2\, dB$ in the $r = 2$ system. In the $r = 1$ case, system performance was improved when the rate of the first subcode was reduced by 0.012, shown with a label 'Mod' in Figure 2.11. Similar rate reduction did not produce appreciable performance improvement in the $r = 2$ system.

The need for a rate adjustment in the rate $r = 1$ system stems from non-Gaussianity in the channel node message densities when set partitioning labelling is used. Consider detecting the least significant bit first in a natural 4-ASK modulation, and let $\{-3, -1, 1, 3\}$
be the channel output constellation points. The channel node output is zero if a value of \( y = 0 \) is received, similar to the binary AWGN channel. There are also two other values \( y = \pm \alpha, 1 < \alpha < 3 \), for which the channel output is zero, that are not present in other labelling formats. The extra zeros lead to an LLR message density for the first subchannel that has peaks near zero. This can be seen in Figure 2.12, which shows the first subchannel conditional LLR probability density function (pdf) at the design SNR corresponding to rates \( r = \{1, 2\} \) for Gray and set partitioning labelling. The pdf of the Gray labelled LLR is quite Gaussian, whereas the set partitioning case has a single peak at zero for the \( r = 1 \) case, and multiple superimposed peaks at zero for the \( r = 2 \) case. These peaks hinder low rate LDPC code performance that has been optimized for a Gaussian message density. A simple approach to mitigate this mismatch is to reduce the rate of the first subcode in the lower rate system.
Figure 2.13: Comparison of GLDPC coded BICM-IDD and MLCM-MSD transmissions at rates $r = \{1, 2, 3\}$

The performance of the proposed GLDPC codes is shown for both the BICM-IDD and MLCM-MSD systems in Figure 2.13 at design rates $r = \{1, 2, 3\}$. In the BICM-IDD system, the channel node message was updated once every eight decoding iterations, producing slightly better results than if check, variable and channel nodes were all updated at the same rate. This behaviour is a result of lessened channel message correlation for the large updated period. For the $r = 1$ case, the 16-QAM BICM-IDD system outperformed the comparable MLCM-MSD configuration, but not the less complex QPSK BICM-IDD system. At the higher design rates, the MLCM-MSD system outperformed the BICM system under iterative detection by $0.2dB$ and $0.8dB$ at design rates of $r = 2$ and $r = 3$, respectively. The relatively poor performance of the higher rate BICM-IDD systems stems from greater channel message correlation associated with higher order modulation formats.
2.8 Summary

In this chapter, a set of generalized irregular LDPC codes is developed by extending single parity check codes with additional parity bits that are connected directly to the channel without repetition. These codes are aimed at the low rate $R < 1/3$ region where conventional LDPC code have low girth problems and require a large power loss to open a decoding tunnel. The generalization of the check node with degree-1 channel connections increases the area under the check node EXIT curve and the rate of the LDPC subcode. The use of a higher rate LDPC subcode mitigates adverse low rate effects. In coded modulation systems, the proposed codes improve MLCM-MSD performance over a wide range of rates, when set partitioning labelling is used. The GLDPC coded MLCM-MSD system also outperforms comparable BICM-IDD systems for higher order modulation formats.
Chapter 3

Irregular Convolutional Coding

The GLDPC codes developed in the last chapter were rendered ineffective under iterative
detection and decoding because of message correlation introduced by the multiple serially
concatenated structure. BICM transmissions are usually coded with a single outer code
such as a convolutional code, or a parallel concatenated outer code such as a turbo code.
This chapter develops an irregular convolutional coding structure for use as an outer code
in a BICM system. These codes can be designed to suit different mapping choices and
channel conditions.

3.1 Introduction

Since the discovery of turbo codes [13, 14], iterative decoding of concatenated convolu-
tional codes has been the subject of much research interest and a part of many existing
wireless standards. In comparison to LDPC codes, turbo codes perform better over short
blocks and slightly worse for longer transmissions. In coded modulation applications,
turbo codes and convolutional codes are widely used in BICM-IDD systems with itera-
tive detection and decoding [67, 68, 69, 70, 71, 52, 72]. The outer code design is influenced
by the channel, mapping, and modulation.

EXIT charts are a useful outer code design tool in BICM-IDD systems. Under IDD
processing, messages are passed between the outer decoder and the channel detector. The evolution of mutual information for these messages can be represented on an EXIT chart, and the code design problem becomes one of finding a code with an EXIT curve that matches the channel EXIT curve [72, 21]. Gray mapping has an almost horizontal EXIT curve for many channels and is generally well paired with a turbo code outer code. For other mapping formats, it is more difficult to find an off-the-shelf outer code with good EXIT curve matching properties.

There are two approaches to mitigate capacity loss due to EXIT curve mismatch that can be employed separately or together. The first to use an irregular outer code to influence the decoder EXIT curve, and the second is to control the shape of the decoder EXIT curve through a multi-dimensional mapping function. The latter approach is used in [67, 68, 69, 73, 74, 52, 40, 42], which applies the binary switching algorithm to optimize the detector EXIT curve. Alternatively, the decoder EXIT curve can be controlled through puncturing. In [72, 75, 76], an irregular convolutional code is formed using a set of subcodes produced from a convolutional mother code and a set of rate compatible puncture patterns. For turbo codes, EXIT curve shape can be manipulated to some extent using regular repetition and puncturing of the systematic or parity bits [77].

This chapter extends the work on irregular convolutional codes in [72] by introducing irregularity through generators of different memory. In [72], a single low-rate mother code is used with a rate compatible set of puncture patterns to produce a set of subcodes. These subcodes can be represented as a family of non-intersecting ‘S’ shaped EXIT curves, where each curve consists of two flat regions with a steeply sloped transition region in the middle. The set of EXIT curves are approximately a horizontal shift of each other. The use of irregular puncturing alone allow some flexibility to design a decoder EXIT to match the channel and mapper characteristic. More efficient codes can be designed with an expanded set of subcodes that differ in terms of both horizontal alignment and steepness of the transition region. The steepness of the transition region
is determined by the memory of the underlying convolutional mother code. A systematic procedure to recursively find mother codes and puncture patterns for a family of codes up to an arbitrary maximum code memory is developed in this chapter. Different memory convolutional subcodes gives rise to trellises of different size, that need to be connected in a way that ensures proper state transitions. Two trellis concatenation strategies are proposed, first strategy converts all trellises to the same size, and the second strategy uses puncture and termination bits to ensure proper state transitions.

### 3.2 Trellis Termination

A convolutional code is often analyzed through its trellis, which represents the possible states of the convolution encoder over time, along with the possible state transitions and output bits as a function of the input bit sequence. With the exception of the trellis sections near the beginning and end of the trellis, the trellis is regular and each section is identical. At the receiver, the evolution of state probabilities can be recursively evaluated as a function of the channel observation at time $n$, code constraints, and the set of state probabilities at time $n - 1$. This dynamic program is formalized in the Viterbi algorithm [78, 79] that produces hard decisions for uncoded bits after an appropriate decision delay. The BER performance of the Viterbi algorithm can be evaluated by applying the union bound on the transfer function representation of the infinite trellis [80].

In a concatenated coding scheme with iterative detection, the constituent decoder or decoders must produce soft outputs that are iteratively exchanged. In order to produce soft outputs, the decoder must recursively generate a set of backwards state probabilities in addition to the forward recursion used in the Viterbi algorithm. The a posteriori probability for a bit at time $n$ is evaluated by averaging possible state transitions using set of forward state probabilities at time $n$, and backward state probabilities at time $n + 1$. This forward-backward algorithm is referred to as the Bahl-Cocke-Jelinek-Raviv (BCJR) [81]
algorithm, and an implementation of this algorithm for concatenated convolutional codes can be found in [82]. The performance of concatenated subcodes is more tricky to analyze than single codes. Since the coded BER in iterative processing is typically quite high, the union bound cannot be applied to the individual subcodes directly. Bounding techniques can be applied to the graph of a concatenated code, where low-BER performance is dominated by the minimum free distance of the concatenated code [83, 84]. Maximizing the free distance is the key ingredient in designing a good interleaver, such as the S-random interleaver for the parallel concatenated configuration [85, 86, 87, 88]. Although the minimum free distance is a good design criterion from an interleaver design perspective, it does not provide many guidelines to design the constituent codes. There is also the question of whether an iterative decoder can achieve ML decoding performance or close to it. This problem is better answered using EXIT charts that graphically matches constituent subcode transfer properties.

Convolutional trellises have a regular structure except for possibly the sections near the start and end. A trellis starts in an initial state that is usually zero and ends at some final state. For a state encoder of memory $m_0$, a set of non-informative termination bits is encoded at the end of the block. There are several approaches that can be taken to terminate a trellis. The simplest is direct truncation, which does not use termination bits and incurs no rate-loss. The drawback is that the bits near the end of the trellis are more prone to error. Another strategy is zero termination, where $m_0$ bits are encoded after a block of $K$ information bits. If the code is recursive, the feedback tap is removed so that zeros are fed into the first register. The termination bits lead to a fractional rate-loss of $\frac{m_0}{K+m_0}$, that becomes significant for shorter block lengths. A third solution is tail-biting [89, 90, 91], which generates a circular trellis by making the initial and end states the same. There is no rate-loss due to termination bits, but a more complex two-pass encoding procedure is required. The tail-biting circular trellises outperform the other two structures, as it offers all bits the same level of protection without a fractional
rate-loss.

As tail-biting trellises perform better than terminated trellises, it is worth discussing the former in some detail. A convolutional code can be represented using state space equations:

\[
\begin{align*}
    x_{n+1} &= Ax_n + Bu_n \\
y_n &= Cx_n + Du_n
\end{align*}
\]

(3.1)

where \(x_n \in \mathbb{B}^{m_0}\) is the state vector at time \(n\), \(u_n \in \mathbb{B}^{k_0}\) is the input vector with \(k_0 = 1\) for binary convolutional subcodes, and parameters \(A, B, C, D\) are binary matrices that represent the convolutional code constraints. For a memory \(m_0 = 2\) convolutional code with recursive systematic generator \(g = [7 5]\) (Octal) shown in Figure 3.1, the state-space parameters take on values

\[
A = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, \quad B = 1, \quad C = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 1 \\ 0 \end{bmatrix}
\]

and generate a trellis also shown in the figure. The solution to (3.1) is evaluated at time \(n\) as

\[
x_n = x_n^{[zi]} + x_n^{[zs]} = A^T x_0 + \sum_{m=0}^{n-1} A^{n-1-m} B u_m^T u_m
\]

(3.2)

which is the superposition of a zero-input solution \(x_n^{[zi]}\) generated by the condition \(u_i = 0\ \forall i\), and a zero-state solution \(x_n^{[zs]}\) generated from the initial state \(x_0 = 0\). If a trellis circularity is imposed, the state after the last bit is encoded must be the same as the initial state. Enforcing the condition \(x_N = x_0\) for a block of \(N\) uncoded bits yields the constraint equation,

\[
(A^N + I_{m_0}) x_0 = x_n^{[zs]}
\]

(3.3)

The initial state can be evaluated as the solution to (3.3) provided that the generator is chosen so that the matrix \((A^N + I_{m_0})\) is invertible. Circular trellises cannot always be found at certain block lengths, such as lengths that are divisible by three [92, 93]. If the
Figure 3.1: Recursive systematic convolutional code with generator $g_{oct} = [7 \ 5]$, with corresponding trellis. Also shown is the projected trellis over $m_0 = 3$ space.

exact number of uncoded bits is fixed by external factors, it is always possible to add one or two punctured systematic bits to avoid block length that are divisible by three.

Another tail-biting strategy has been proposed in [94] that is based on code extension and puncturing. In this approach, $K$ information bits are encoded along with $m_0$ parity bits. The parity bits are chosen such that the trellis state at time $K + m_0 + 1$ is the same as the trellis state at time $m_0 + 1$. The trellis is hence circular over the interval $[m_0 + 1, K + m_0]$ and the first $m_0$ bits can be punctured, since these bits can be determined from the state at time $m_0 + 1$. Puncturing the first $m_0$ bits eliminates fractional rate-loss, but leads to less confident messages for bits near the punctured block. As a result, the extend and puncture tail-biting approach is outperformed by the tail-biting solutions in [90, 91] that maintains a uniform trellis.
3.3 Irregular Trellis Concatenation

In an irregular convolutional code, each convolution subcode generates different trellis sections with different numbers of states. We assume that all trellis sections generated from one mother code are grouped together in a contiguous block, and that there is up to one mother code of a given memory. As the decoder complexity is a function of code memory, let $M$ be the maximum code memory and let $\alpha_m$ be the number of trellis sections of memory $m$. Unless otherwise stated, a natural ordering of trellis blocks is assumed based on ascending code memory.

There are two approaches to connecting trellises of different sizes: convert all the trellises to the same size, or using termination and puncture bits to control trellis transitions. The constant trellis size strategy is an extension of the tail-biting solution in [90, 91], while the variable trellis size approach is based on the extend and puncture tail-biting solution in [94]. Each strategy has its relative strengths and weaknesses that depend on packet length.

A constant size trellis can be formed by recursively representing a trellis of size $2^m$ within a trellis of size $2^{m+1}$, until all trellises have the maximum size. Consider projecting the $g = [7 \ 5]_{oct}$, $m = 2$ convolutional code in Figure 3.1 onto an $m = 3$ state-space by adding an additional most significant bit. The output is unaffected by the additional bit, and the new state bit $s_{m-1}$ follows the most significant state bit $s_{m-2}$, as can be seen in the enlarged projected trellis, also shown in Figure 3.1. Subcodes can be converted to a common memory $M$ so that all the trellis sections have size $2^M$. Over this modified trellis, the BCJR algorithm from [82] can be applied with constraints that are based on the time-varying state-space equations. The state-space equations in (3.1) can be applied with vector dimensional agreement by replacing the constant parameters $A, B, C, D$ with time-varying parameters $\tilde{A}_n, \tilde{B}_n, \tilde{C}_n, \tilde{D}_n$ that are determined by the subcodes used to generate the $n^{th}$ trellis section. Also, let the parameters $\tilde{A}_n, \tilde{B}_n, \tilde{C}_n, \tilde{D}_n$ take on values $A_m, B_m, C_m, D_m$ if the $n^{th}$ trellis section is generated from a subcode having memory $m$. 
For an irregular convolutional code, enforcing the condition of trellis circularity $x_N = x_0$, yields a constraint equation

$$(\mathbf{A}_M^{\alpha M} \times \ldots \times \mathbf{A}_1^{\alpha_1} + \mathbf{I}_M)x_0 = x_0^{[z_s]}$$

(3.4)

where the $\mathbf{A}_m$ is defined as

$$\mathbf{A}_m = \begin{bmatrix} \mathbf{0}_{M \times 1} & \mathbf{I}_{M-1} \\ \mathbf{0}_{1 \times M-m} & \mathbf{f}_m \end{bmatrix}$$

with $\mathbf{f}_m$ being the binary feedback vector for the $m^{th}$ subcode. The correct value of the initial state $x_0$ can be found from (3.4) through matrix inversion, since the matrix on the right hand side of (3.4) is typically non-singular. If an integer solution to (3.3) is not found, then a circular trellis cannot be formed. Since the matrix $(\mathbf{A}_M^{\alpha M} \times \ldots \times \mathbf{A}_1^{\alpha_1} + \mathbf{I}_M)$ is typically singular, it is possible to have an integer solution to (3.3) for some uncoded bit sequences and not for others. Several candidate sequences must be tested before declaring a trellis circular.

If an irregular trellis is found to be non-circular, there are a couple of adjustments that can be made to bring about circularity. Since matrix multiplication is not commutative, the order of the trellis blocks influences the circularity of the trellis. If a natural trellis order does not produce a circular trellis, other orderings can be randomly generated, thereby changing the order of the $\mathbf{A}_m$ matrices in (3.3). If a circular trellis cannot be found by reordering, the number of trellis sections of one mother code can be incremented and another one decremented, choosing mother codes randomly. Reordering can then be applied on the modified trellis.

The variable sized trellis can be formed by connecting different size trellises with a judicious use of puncture and termination bits. An arbitrary state in a trellis of size $2^m$ can be reached from the 0 state after $m$ bits, and is determined by a uniquely bit sequence for all one-to-one codes. These bits can be punctured as is done in [94], since the bit values are represented in the state. Now consider two adjacent trellises, the first
of size $m_1$ and the second of size $m_2$ with $m_2 > m_1$. The first $m_2 - m_1$ bits in the second trellis can be punctured since the states in these sections only have one valid state transition. Consider ordering trellis in increasing order of memory with \{\(m_{\text{min}}, M\)\} being the minimum and maximum memory, and encoding \(K\) information bits from state-zero, and \(M\) termination bits. The first \(m_{\text{min}}\) are punctured and the last \(M\) termination bits are used to set the final state equal to the state at time \(m_{\text{min}}\). This ensures trellis circularity. There are additionally \(M - m_{\text{min}}\) bits at the beginning of each larger trellis block that are also punctured. This eliminates fractional rate-loss since a total of \(M\) bits are punctured, equaling number of terminations bits.

Comparing the variable and fixed trellis size construction approaches, each has its relative merit. Puncturing coded bits reduces the average channel message mutual information per bit and leads to a fractional capacity loss for short packets. The fixed trellis size solution has a weak bits near high-to-low memory transitions. This is because last few bits of the longer memory code are not part shorter memory code feedback loop, leading to an error floor and poorer longer packet performance.

### 3.4 Irregular Code Design using EXIT Charts

A convolutional outer code can be serially concatenated with a wide range of different types of inner codes, the most obvious being another convolutional code. A serially concatenated convolutional code can be designed to perform within 0.1dB of the Shannon capacity using a regular outer code and code doping strategy with two inner codes \[95\]. Unfortunately, serially concatenated convolutional codes do not outperform turbo codes for short block lengths, despite having a lower SNR convergence threshold. In addition, serially concatenated convolutional codes and LDPC codes have approximately the same performance for large block length, but the former typically incurs greater decoder latency because of the serial nature of trellis decoding.
The main application for serial concatenated convolutional codes is in BICM-IDD systems, where the channel detector acts as an inner decoder. One may be tempted to apply further a serially concatenated code to form a triple concatenated code with the channel acting as the innermost code. In order to avoid a capacity loss [72], the middle code must be of unity rate which significantly increases interleaver message correlation. Since the inner and outer interleavers are the same size, the inner interleaver cannot provide any additional spreading than the outer interleaver alone. The negative effect of the inner interleaver becomes more apparent for correlated symbol mappings, although the triple code concatenated structure should be avoided altogether if possible. A similar problem is discussed in Chapter 2 with respect to GLDPC coded BICM-IDD system, where the middle repetition code node is sensitive to channel message correlation.

The BICM channel has an EXIT curve shape that is influenced by the mapping and physical channel, and whose area approximates the CM channel capacity. The outer code is designed to match the reflected detector EXIT curve. As the outer code is composed of a set of convolutional subcodes, understanding the relationship between EXIT curve shape and the choice of generator polynomial is of key importance. Following the work in [72], we consider RSC mother codes. All one-to-one RSC code realizations have an EXIT curve that starts at $I_E = 0$ and terminates at $I_E = 1$, which is required for convergence.

The EXIT characteristics of unpunctured RSC codes are largely determined by the code memory and the number of generator polynomials. Consider a family of RSC rate $r = 1/4$ convolutional codes having generators $g_1 = [3 2 3 3]_{oct}$, $g_2 = [5 7 6 6]_{oct}$, $g_3 = [13 15 11 9]_{oct}$, $g_4 = [25 19 21 29]_{oct}$, $g_5 = [51 37 59 53]_{oct}$, $g_6 = [81 111 123 75]_{oct}$, $g_7 = [193 175 245 238]_{oct}$, where the first element is the feedback polynomial. A set of rate $r = 1/2$ codes are formed by omitting the last two polynomials in each case. The EXIT curves for these 14 codes are shown in Figure 3.2. All the curves have a ‘S’ shape with an initial region of gradually increase away from zero, followed by a steep
transition region and a final region of gradual increase towards unity. The area under the EXIT curve is approximately equal to the reciprocal of the code rate [21]. Adding more generators naturally increases this area. The effect of changing the code rate appears as a horizontal shift that can be seen between the $r = 1/4$ and $r = 1/2$ codes. A byproduct of the area constraint and curve symmetry is that all the curves of a certain rate intersect at crossover point near $I_E = 0.5$.

The memory of the code generator has a significant effect on the EXIT curve shape. From Figure 3.2, long memory codes have a steeper transition region and are flatter at the extrema. Of course, the codes depicted in Figure 3.2 are well chosen and an arbitrarily chosen code with long memory does not necessarily have a steeper transition region. Fixing the memory, however, limits the maximum slope in the transition regional and flatness in the extrema. This is a result of the fact that short memory codes perform better in the low SNR range and longer ones do better at high SNR. The initial flat region in Figure 3.2 corresponds to a low SNR range where the shorter memory code performs better, while the terminal flat region is a high SNR range where longer memory codes have the advantage.
Convolutional code generator polynomials are traditionally designed based on bounds, such as the union bound from above and the minimum distance bound from below. These bounds can be evaluated over a signal flow graph lifted from the trellis [79], and become tight at high SNR. Within the EXIT chart framework, the search for a good high SNR codes is one of finding a code with an EXIT curve that is a minimal distance away from $y = 1$ at a value of $x$ to the right of the crossover point. Because of the fixed nature of the crossover point, maximizing the flat region to the right of this point is equivalent to minimizing the flat region to the left and is also equivalent to maximizing the slope of the transition region. For a given memory, the goal is to find generators that maximize the transition slope or equivalent extrema properties. The set of codes produced at different memory leads to a family of EXIT curves that only intersects at the crossover point.

A set of convolutional mother codes can be recursively found as follows. Let $g_n^{(m)}$ be a length $n$ vector of generator polynomials that represents a rate $1/n$ RSC convolutional code of memory $m$. Rate $1/2$ codes can be found by examining the set of $2^{2(m+1)}$ possible memory $m$ codes and choosing generators that maximize the transition region slope. The subset of possible codes corresponding to lower memory codes can be eliminated, and codes that are not one-to-one can be quickly identified in simulations. There are typically several codes that have maximal slope and any one of them can be chosen. Finding lower rate codes proceeds in a recursive manner. The rate $1/(n+1)$ code can be found from the rate $1/n$ code by considering possible generators of the form $g_{n+1}^{(m)} = [g, g_n^{(m)}]$, and by selecting the $g$ that maximize the transition region slope. A set of convolutional mother codes found in this manner is shown in Table 3.1 for code memory $M = \{1\ldots7\}$ and rates of $r = \{1/2,\ldots,1/10\}$ in octal notation. The feedback polynomial and first generator polynomial are the same.

The use of multiple generator polynomials of different memory only affords limited degrees of freedom and cannot be used for high rate codes. Puncture patterns can be applied to fix the code rate thereby determining the horizontal alignment of the transition
region. The set of puncture patterns for a particular code should generate rates that uniformly span the EXIT space to ensure good curve matching properties. A particular puncture pattern and code pair may generate a subcode that is not one-to-one, having a terminal EXIT point below unity, or a low weight subcode having a less steeply sloped transition region. As such situations are undesirable, the puncture pattern should be chosen to match the underlying mother code. The maximal transition slope criteria can be used to choose between different puncture patterns of the same rate.

Consider a set of 17 uniformly spaced subcode rates \( r = \{0.1, 0.15, \ldots, 0.85, 0.9\} \) that can be generated using puncture patterns with corresponding periods \( N_p = \{1, 3, 1, 1, 3, 7, 2, 9, 1, 11, 3, 13, 7, 3, 4, 17, 9\} \). The puncture patterns are of minimal length to achieve the desired rate. A set of 200 random puncture patterns for each subcode rate and puncture period were generated, and the steepest transitional slope selection criterion was applied. Initially, only systematic puncture patterns were considered, which produced satisfactory patterns for all rates and generators except \( r = 0.8 \). For the \( r = 0.8 \) subcodes, a non-systematic puncture pattern was required for the \( m_0 = 2, 5, 6, 7 \) generators. A set of puncture patterns produced in this manner is shown in Tables 3.2 - 3.8 for trellis memory \( m_0 = \{1 \ldots 7\} \) using the mother codes shown in Table 3.1. The en-

<table>
<thead>
<tr>
<th>m</th>
<th>( f_k )</th>
<th>( g_1^{{m}} )</th>
<th>( g_2^{{m}} )</th>
<th>( g_3^{{m}} )</th>
<th>( g_4^{{m}} )</th>
<th>( g_5^{{m}} )</th>
<th>( g_6^{{m}} )</th>
<th>( g_7^{{m}} )</th>
<th>( g_8^{{m}} )</th>
<th>( g_9^{{m}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>7</td>
<td>6</td>
<td>6</td>
<td>5</td>
<td>7</td>
<td>5</td>
<td>7</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>15</td>
<td>11</td>
<td>9</td>
<td>7</td>
<td>11</td>
<td>5</td>
<td>13</td>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>25</td>
<td>19</td>
<td>21</td>
<td>29</td>
<td>25</td>
<td>31</td>
<td>25</td>
<td>21</td>
<td>29</td>
<td>28</td>
</tr>
<tr>
<td>5</td>
<td>51</td>
<td>37</td>
<td>59</td>
<td>53</td>
<td>55</td>
<td>39</td>
<td>55</td>
<td>63</td>
<td>22</td>
<td>45</td>
</tr>
<tr>
<td>6</td>
<td>81</td>
<td>111</td>
<td>123</td>
<td>75</td>
<td>125</td>
<td>89</td>
<td>97</td>
<td>49</td>
<td>115</td>
<td>25</td>
</tr>
<tr>
<td>7</td>
<td>193</td>
<td>175</td>
<td>245</td>
<td>238</td>
<td>43</td>
<td>188</td>
<td>227</td>
<td>6</td>
<td>31</td>
<td>250</td>
</tr>
</tbody>
</table>

Table 3.1: Generators for RSC mother codes
let \((m, p)\) be the fraction of coded bits produced by the memory \(m\) mother encoder and

<table>
<thead>
<tr>
<th>Table 3.2: Puncture Patterns for (m_0 = 1) RSC mother codes (octal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.1, 1, (1, 1, 1, 1, 1, 1, 1, 1)), (0.15, 3, (7, 7, 7, 7, 7, 3)), (0.20, 1, (1, 1, 1, 1, 1))</td>
</tr>
<tr>
<td>(0.25, 1, (1, 1, 1, 1)), (0.30, 3, (7, 7, 7, 1)), (0.35, 7, (177, 177, 175))</td>
</tr>
<tr>
<td>(0.40, 2, (3, 3, 1)), (0.45, 9, (777, 367, 245)), (0.50, 1, (1, 1)), (0.55, 11, (3777, 1577))</td>
</tr>
<tr>
<td>(0.60, 3, (7, 5)), (0.65, 13, (1777, 10663)), (0.70, 7, (177, 45))</td>
</tr>
<tr>
<td>(0.75, 3, (7, 1)), (0.80, 4, (17, 1)), (0.85, 17, (377777, 20101)), (0.90, 9, (777, 1))</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 3.3: Puncture Patterns for (m_0 = 2) RSC mother codes (octal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.1, 1, (1, 1, 1, 1, 1, 1, 1, 1, 1)), (0.15, 3, (7, 7, 7, 7, 7, 3)), (0.20, 1, (1, 1, 1, 1, 1, 1))</td>
</tr>
<tr>
<td>(0.25, 1, (1, 1, 1, 1)), (0.30, 3, (7, 7, 7, 1)), (0.35, 7, (177, 177, 175))</td>
</tr>
<tr>
<td>(0.40, 2, (3, 3, 1)), (0.45, 9, (777, 735, 454)), (0.50, 1, (1, 1)), (0.55, 11, (3777, 2776))</td>
</tr>
<tr>
<td>(0.60, 3, (7, 6)), (0.65, 13, (1777, 11631)), (0.70, 7, (177, 107))</td>
</tr>
<tr>
<td>(0.75, 3, (7, 1)), (0.80, 4, (15, 6)), (0.85, 17, (377777, 110040)), (0.90, 9, (777, 1))</td>
</tr>
</tbody>
</table>

try \(\{r, N_p, (p_1, p_2, \ldots, p_n)\}\) has rate \(r\), puncture period \(N_p\) with a puncture pattern \((p_1, p_2, \ldots, p_n)\) specified in octal notation. For example, the rate 0.45 entry for the \(m_0 = 1\) code generators in Table 3.2 has a puncture period of 9 and a puncture pattern of \((777, 367, 245)\) in octal or \((11111111, 011110111, 010100101)\) in binary. This is a systematic puncture pattern since none of the first generator bits are punctured.

The set of EXIT curves produced from the generators in Table 3.1 and puncture patterns from Tables 3.2 - 3.8 are shown in Figure 3.3. For a given memory, the set of rate compatible EXIT curves do not intersect. For a particular rate, codes of different memory only intersect at the terminal points and a crossover point near \(I_E = 0.5\), decreasing slightly for high rate codes.

The overall irregular convolutional code is constructed by encoding different fractions of bits with different mother codes, and by puncturing with the appropriate patterns. Let \(\lambda_{m,p}\) be the fraction of coded bits produced by the memory \(m\) mother encoder and
\{0.1, 1, (1, 1, 1, 1, 1, 1, 1, 1, 1, 1)\}, \{0.15, 3, (7, 7, 7, 7, 7, 3)\}, \{0.20, 1, (1, 1, 1, 1, 1)\} \\
\{0.25, 1, (1, 1, 1, 1)\}, \{0.30, 3, (7, 7, 7, 1)\}, \{0.35, 7, (177, 157, 177)\} \\
\{0.40, 2, (3, 3, 1)\}, \{0.45, 9, (777, 171, 746)\}, \{0.50, 1, (1, 1)\}, \{0.55, 11, (3777, 1767)\} \\
\{0.60, 3, (7, 6)\}, \{0.65, 13, (17777, 6554)\}, \{0.70, 7, (177, 26)\} \\
\{0.75, 3, (7, 1)\}, \{0.80, 4, (17, 1)\}, \{0.85, 17, (37777, 4060)\}, \{0.90, 9, (777, 1)\}

Table 3.4: Puncture Patterns for \(m_0 = 3\) RSC mother codes (octal)

\{0.1, 1, (1, 1, 1, 1, 1, 1, 1, 1, 1, 1)\}, \{0.15, 3, (7, 7, 7, 7, 7, 3)\}, \{0.20, 1, (1, 1, 1, 1, 1)\} \\
\{0.25, 1, (1, 1, 1, 1)\}, \{0.30, 3, (7, 7, 7, 1)\}, \{0.35, 7, (177, 77, 175)\} \\
\{0.40, 2, (3, 3, 1)\}, \{0.45, 9, (777, 176, 661)\}, \{0.50, 1, (1, 1)\}, \{0.55, 11, (3777, 3567)\} \\
\{0.60, 3, (7, 3)\}, \{0.65, 13, (17777, 1553)\}, \{0.70, 7, (177, 64)\} \\
\{0.75, 3, (7, 1)\}, \{0.80, 4, (17, 1)\}, \{0.85, 17, (37777, 220004)\}, \{0.90, 9, (777, 1)\}

Table 3.5: Puncture Patterns for \(m_0 = 4\) RSC mother codes (octal)

\{0.1, 1, (1, 1, 1, 1, 1, 1, 1, 1, 1, 1)\}, \{0.15, 3, (7, 7, 7, 7, 7, 3)\}, \{0.20, 1, (1, 1, 1, 1, 1)\} \\
\{0.25, 1, (1, 1, 1, 1)\}, \{0.30, 3, (7, 7, 7, 1)\}, \{0.35, 7, (177, 167, 175)\} \\
\{0.40, 2, (3, 3, 1)\}, \{0.45, 9, (777, 643, 535)\}, \{0.50, 1, (1, 1)\}, \{0.55, 11, (3777, 3753)\} \\
\{0.60, 3, (7, 5)\}, \{0.65, 13, (17777, 5037)\}, \{0.70, 7, (177, 45)\} \\
\{0.75, 3, (7, 1)\}, \{0.80, 4, (10, 17)\}, \{0.85, 17, (37777, 2420)\}, \{0.90, 9, (777, 1)\}

Table 3.6: Puncture Patterns for \(m_0 = 5\) RSC mother codes (octal)
Figure 3.3: EXIT curves for punctured subcode set for code memory $M = \{1, \ldots, 7\}$
\{0.1, 1, (1, 1, 1, 1, 1, 1, 1, 1, 1, 1), \{0.15, 3, (7, 7, 7, 7, 7, 7, 3), \{0.20, 1, (1, 1, 1, 1, 1, 1)\} \\
{0.25, 1, (1, 1, 1, 1)}, \{0.30, 3, (7, 7, 7, 1), \{0.35, 7, (177, 137, 177)\} \\
{0.40, 2, (3, 3, 3)}, \{0.45, 9, (777, 771, 452), \{0.50, 1, (1, 1)\}, \{0.55, 11, (3777, 1776)\} \\
{0.60, 3, (7, 1)}, \{0.65, 13, (17777, 15644), \{0.70, 7, (177, 31)\} \\
{0.75, 3, (7, 1)}, \{0.80, 4, (14, 7), \{0.85, 17, (377777, 124000), \{0.90, 9, (777, 1)\} \\

Table 3.7: Puncture Patterns for \(m_0 = 6\) RSC mother codes (octal)

\{0.1, 1, (1, 1, 1, 1, 1, 1, 1, 1, 1, 1), \{0.15, 3, (7, 7, 7, 7, 7, 7, 3), \{0.20, 1, (1, 1, 1, 1, 1, 1)\} \\
{0.25, 1, (1, 1, 1, 1)}, \{0.30, 3, (7, 7, 7, 1), \{0.35, 7, (177, 157, 177)\} \\
{0.40, 2, (3, 3, 1)}, \{0.45, 9, (777, 637, 113), \{0.50, 1, (1, 1)\}, \{0.55, 11, (3777, 3537)\} \\
{0.60, 3, (7, 6)}, \{0.65, 13, (17777, 6552), \{0.70, 7, (177, 46)\} \\
{0.75, 3, (7, 1)}, \{0.80, 4, (17, 1), \{0.85, 17, (377777, 202200), \{0.90, 9, (777, 1)\} \\

Table 3.8: Puncture Patterns for \(m_0 = 7\) RSC mother codes (octal)

punctured with the corresponding \(p^{th}\) puncture pattern in the \(m^{th}\) subset. It is convenient
to index over \(m\) and \(p\) using a single variable \(i = m \times P + p\), where \(P = 17\) is the number
of puncture patterns per code. The optimal degree distribution \(\{\lambda_i\}_i\) can be found as
the solution to the following constrained rate maximization problem:

\[
\arg \max_{\{\lambda_i\}} \sum_{i=1}^{Pm_0^\text{max}} \lambda_i r_i \quad \text{s.t.} \quad \sum \lambda_i = 1, \quad \sum \lambda_i f_c^{(i)}(I_I) > f_{ch}^{-1}(I_I) \forall I_I
\]

(3.5)

where \(r_i\) is the rate of the \(i^{th}\) subcode, \(f_c^{(i)}(I_I)\) is the \(i^{th}\) subcode EXIT curve, and \(f_{ch}^{-1}(I_I)\)
is the inverse channel EXIT function. The problem in (3.5) can be solved using a gradient
search technique discussed in Chapter 2, with additional upper bound constraints to aid
in convergence.
3.5 Simulated Results

This section examines how the proposed coding scheme performs with simulated examples over an AWGN channel and an ergodic fading MIMO channel with two transmitter and two receiver antennas. Monte Carlo simulation were performed until 300 frame errors occurred, or until 10000 trials were performed. For each trial, 40 turbo iteration were performed with the proposed code. For the reference turbo code design, 5 iterations were performed between the detector and decoder, and 20 turbo sub-iteration were performed during each global iteration. Finally, for the reference MLCM-MSD design using GLDPC subcodes, 200 decoder iterations were used at each MSD decoding stage.

We consider the performance of the proposed codes using different 16-QAM mapping functions. Gray and natural labelling are evaluated along with three kinds of random mapping functions: random ASK, random QAM and random MQAM. The random ASK and QAM schemes generate random assignments over a subset of constellation points, then merges these subsets to produce the overall mapping function. The random MQAM labelling functions applies a random mapping across two transmitter antennas in the MIMO channel, and across two timeslots in the AWGN channel, leading to an overall constellation of 256 points for a 16-QAM modulation format. A random MQAM symbol uses a pseudo-random mapping between binary vectors and constellation points, which is known at the transmitter and receiver, and changes between symbols. The five mapping functions differ in terms of the amount of message correlation they produce across symbol bits. Random MQAM labelling has the most detector message correlation, followed by random QAM, random ASK, natural and finally gray labelling.

The channel EXIT curves for the five labelling functions are shown in Figure 3.4 over the AWGN and MIMO channels. The EXIT curves were generated at a channel SNR of $\rho^* = 5.11 dB$ for the AWGN channel, and $\rho^* = 7.06 dB$ for the MIMO channel. These channel threshold SNR values yield a binary channel capacity of 1/2 for all mapping formats, when the optimal APP detector is applied. All the EXIT curves appear relatively
Figure 3.4: Channel EXIT curves for different 16-QAM mapping functions over the AWGN and $2 \times 2$ MIMO fading channel.

straight, with a slope that is strongly influenced by the mapping function. Gray labelling produces a nearly horizontal curve followed by natural labelling, random ASK, random QAM, and random MQAM labelling. The MIMO channel EXIT curves are consistently steeper than the corresponding AWGN channel curves. Also shown in Figure 3.4 is a modified channel EXIT curve for all the mapping functions except the random MQAM. The modified channel curves are produced by reducing the terminal EXIT point according to $f_{ch}(1) \leftarrow \max(0.95f_{ch}(1), f_{ch}(0.9))$, and interpolating. This modification can lead to improved error floor performance for some mapping functions.

Using the code optimization in (3.5), irregular code profiles were generated at different SNR values until a rate $1/2$ code was produced. The set of code threshold SNR’s that yielding rate $1/2$ codes is shown in Table 3.9, for a maximum convolution code memory of $M = \{4, 7\}$. For both the AWGN and MIMO channels, steeper channel EXIT curves lead to lower code threshold values. The EXIT chart predicted code threshold for the
Table 3.9: Code SNR threshold values in decibels for different labelling functions, channels and constellations.

<table>
<thead>
<tr>
<th></th>
<th>Gray</th>
<th>Natural</th>
<th>Rand ASK</th>
<th>Rand QAM</th>
<th>Rand MQAM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AWGN ($\rho^* = 5.11 dB$)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M = 4$</td>
<td>9.01</td>
<td>8.50</td>
<td>8.49</td>
<td>6.17</td>
<td>5.35</td>
</tr>
<tr>
<td>$M = 7$</td>
<td>8.48</td>
<td>7.08</td>
<td>6.98</td>
<td>5.64</td>
<td>5.23</td>
</tr>
<tr>
<td>$2 \times 2$ MIMO ($\rho^* = 7.06 dB$)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M = 4$</td>
<td>11.33</td>
<td>10.23</td>
<td>10.01</td>
<td>8.20</td>
<td>7.58</td>
</tr>
<tr>
<td>$M = 7$</td>
<td>9.70</td>
<td>8.59</td>
<td>8.07</td>
<td>7.65</td>
<td>7.16</td>
</tr>
</tbody>
</table>

MQAM mapping function is within about a tenth of a decibel from capacity, for the memory $M = 7$ design.

The trends in Table 3.9 can be explained by considering the match between the EXIT curve of irregular code profile, and the underlying detector EXIT curve. The detector EXIT curves of the five mapping functions are shown for the AWGN channel in Figure 3.5, along with the corresponding $M = 7$ irregular code curves. The memory $M = 7$ code optimizations for the three ASK mappings produced code profiles with non-zero coefficients of $\lambda_{1,3} = 0.0028$, $\lambda_{3,5} = 0.0008$, $\lambda_{6,7} = 0.0073$, $\lambda_{7,2...8} = \{0.0010, 0, 0, 0, 0, 0.0970, 0.8912\}$ for the Gray labelling, $\lambda_{1,5} = 0.0005$, $\lambda_{7,9...10} = \{0.9118, 0.0877\}$ for the natural labelling, and $\lambda_{5,9...10} = \{0.0002, 0.0001\}$, $\lambda_{6,8...9} = \{0.0040, 0.3001\}$, $\lambda_{7,9} = \{0.6956\}$ for the random ASK labelling. These profile generated almost identical code EXIT curves that are shown on the left of Figure 3.5. The random QAM labelling generated a profile specified by $\lambda_{3,3...5} = \{0.0016, 0, 0, 0.0311\}$, $\lambda_{4,5...9} = \{0.0413, 0.0379, 0, 0.0139\}$, $\lambda_{5,2...8} = \{0.0021, 0, 0, 0, 0.0297, 0.0137, 0.0353\}$, $\lambda_{6,6...10} = \{0.0295, 0.0319, 0.0291, 0.0749, 0.1090\}$, $\lambda_{7,5...11} = \{0.0067, 0.0031, 0.0040, 0.0287, 0.0756, 0.2005, 0.2004\}$, and the coefficients for random MQAM labelling case are specified in Table 3.10. From Figure 3.5, the gap between the detector and decoder EXIT curves decreases for the more correlated mapping
functions.

For all the mapping functions except the random MQAM function, the code EXIT curve intersects the channel EXIT curve near $I_I = 1$. This intersection can limit convergence and lead to an error-floor under non-ideal decoding. This error floor can be reduced by designing codes with the modified channel curves in Figure 3.4, which does not significantly affect the code SNR threshold. The channel curve modification has been applied in code design for all but the random MQAM case.

Practical detection and decoding performance for finite length transmissions falls short of the code threshold values in Table 3.9. This loss is influenced by a number of factors such as the packet length, mapping function, maximum code memory and the type of trellis concatenation. We first evaluate the performance of the fixed and variable size trellis concatenation strategies in Figure 3.6, for transmission of $N_{Sym} = \{125, 1000\}$. Random QAM labelling was used for the $N_{Sym} = 125$ system, and random MQAM labelling for the $N_{Sym} = 1000$ system. The fixed trellis size scheme performed better in
the short packet system, while the variable trellis size scheme performed better in the longer packet system. The short packet fixed size trellis system, and long packet variable size trellis system led to high error floors.

The performance of the proposed codes over the AWGN channel is shown for 1000 symbol transmissions in Figure 3.7, and for 125 symbol transmissions in Figure 3.8. A fixed size trellis was used for the 125 symbol packet and a variable size trellis for the 1000 symbol system. Shown for reference in Figures 3.7-3.8 is the performance of a turbo coded BICM-IDD system, a GLDPC coded MLCM-MSD system from Chapter 2 using set partitioning labelling. The turbo coded system used gray labelling, a S-random interleaver with $S_{parm} = 17$, and a binary turbo code with generator $g_{oct} = [37 \ 23]$. For the longer packet length, the performance of different mapping schemes follow the trends in Table 3.9, with the random MQAM labelled system performing slightly less than $2dB$ from capacity with a $M = 7$ memory code. At the shorter transmission length,

<table>
<thead>
<tr>
<th>(p)</th>
<th>$m = 1$</th>
<th>$m = 2$</th>
<th>$m = 3$</th>
<th>$m = 4$</th>
<th>$m = 5$</th>
<th>$m = 6$</th>
<th>$m = 7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0030</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0092</td>
<td>0.0560</td>
<td>0.0413</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0.0077</td>
<td>0.0017</td>
<td>0.0061</td>
<td>0.0023</td>
<td>0.0232</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0.0014</td>
<td>0.0086</td>
<td>0.0169</td>
<td>0.0302</td>
<td>0.0260</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0029</td>
<td>0</td>
<td>0.0046</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0.0030</td>
<td>0.0014</td>
<td>0.0033</td>
<td>0</td>
<td>0.0690</td>
</tr>
<tr>
<td>7</td>
<td>0.0755</td>
<td>0</td>
<td>0</td>
<td>0.0100</td>
<td>0.0077</td>
<td>0.0151</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.0148</td>
<td>0</td>
<td>0.0119</td>
<td>0.0005</td>
<td>0.0062</td>
<td>0.0222</td>
<td>0.0168</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0.0073</td>
<td>0.0014</td>
<td>0.0024</td>
<td>0.0173</td>
<td>0.0159</td>
<td>0.0057</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0193</td>
<td>0.0206</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0.0024</td>
<td>0</td>
<td>0</td>
<td>0.0221</td>
<td>0.0019</td>
<td>0.0103</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0.0044</td>
<td>0.0119</td>
<td>0.0322</td>
<td>0.0516</td>
<td>0.0284</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0016</td>
<td>0.0042</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0135</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.2031</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0230</td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3.10: AWGN code coefficients for memory $M = 7$ design with random MQAM.
the performance of the random MQAM scheme is limited by high detector message correlation, and the random ASK and random QAM designs perform better. For the random QAM mapping, there is minimal loss is using a less complex memory $M = 4$ code at frame error rates above $FER = 0.005$. In comparison to the reference designs, however, the proposed codes do not perform as well as either the turbo coded system or the GLDPC coded MLCM-MSD system.

The performance of the proposed codes over the $2 \times 2$ MIMO channel is shown in Figure 3.9 for 1000 symbol blocks and in Figure 3.10 for 125 symbol blocks. The performance trends across mapping functions and packet lengths generally follow the AWGN channel trends. Performance within $2dB - 2.5dB$ of capacity is possible with $M = 7$ designs using random QAM and random MQAM labelling for 1000 symbol packets, and random ASK and natural labelling for 125 symbol packets. At both packet lengths, the random MQAM labelled $M = 4$ design outperforms both the reference turbo coded sys-
Figure 3.7: AWGN channel performance of proposed codes for 2000 symbol transmissions.

Figure 3.8: AWGN channel performance of proposed codes for 250 symbol transmissions.
Figure 3.9: 2 × 2 MIMO channel performance using proposed codes for 1000 symbol transmissions.

The M = 4 design with 40 decoding iterations is less complex than either the $m_0 = 4$ turbo code with 80 inner iterations total, or the MLCM-MSD system with 1600 GLDPC decoder iterations for all stages. The more complex memory $M = 7$ codes outperform the reference designs over the MIMO channel, with significant gains in the shorter packet system. For the 125 symbol transmission, the proposed codes perform approximately $1dB$ and $2dB$ better than the reference turbo code and MLCM designs, respectively.

### 3.6 Summary

A framework for constructing and decoding irregular convolutional codes has been developed in this chapter. A convolutional code is made irregular using an irregular puncture pattern and multiple mother codes of different memory. Strategies based on fixed size and
variable size trellises have been proposed to connect different memory mother codes. The proposed irregular convolutional codes are designed within an EXIT chart framework for BICM-IDD systems. With an appropriate mapping choice, proposed codes outperform competing strategies over the $2 \times 2$ independent Rayleigh fading channel.

Figure 3.10: $2 \times 2$ MIMO channel performance using proposed codes for 125 symbol transmissions.
Chapter 4

Iterative Space-Time Detection

This chapter steps back from the coded modulation designs proposed in the previous two chapters and looks at receiver processing. In large systems, optimal detection becomes impractical so less complex designs must be employed. Two novel detectors are developed in this chapter based on a group strategy that can be used in an IDD receiver. As the focus in this chapter is receiver processing, simple convolutional outer-codes are used in a D-BLAST space-time architecture. The work in this is chapter has largely been published in [96, 97, 98].

4.1 Introduction

The use of multiple transmitter and receiver antennas can provide tremendous capacity increases in wireless fading environments [1]. The BLAST architecture [9] is a simple and efficient multiplexing structure that can take advantage of the multi-antenna channel capacity. It is often decoded using an MSD receiver with linear detector processing, and is effective in systems having an excess number of transmitter antennas [99].

There are several detection strategies that can be applied to systems having an excess number of transmitter antennas. An optimal solution is the ML detector. The ML detector unfortunately has exponential complexity that grows with $R^n$, where $R$ is the
constellation size and $n$ is the number of transmitter antennas. The ML detector can be implemented using the sphere decoding algorithm [100] with some complexity savings. Sphere decoding still has exponential worst-case complexity, although the average-case complexity can be polynomial at high SNR [101]. Suboptimal ML detectors have been applied to BLAST systems using tree-search algorithms [102] and group detection strategies [103, 104]. Group detection has also been extensively used in the context of multi-user detection [105, 106, 107, 108, 109, 110].

Iterative detection and decoding is an effective way to improve the performance of a BLAST receiver. Layered IDD receivers have been employed in [111, 112, 113], which consist of an iterative exchange of soft information between the detector and the decoders. The optimal soft-input soft-output (SISO) detector is the maximum a posteriori (MAP) detector, which, like the ML detector, has an exponential complexity that grows with $R^n$. For systems with a large number of transmitter antennas, a computationally more feasible minimum mean squared error (MMSE) detector has been proposed, first in the multiuser context [114, 115] and later applied to BLAST [116, 117]. The detector discussed in [116, 117] uses soft interference cancellation with an instantaneous MMSE filter that only suppresses residual interference that has not been cancelled. The performance of the SISO MMSE detector improves dramatically with successive iterations, but falls short of the MAP detector performance especially in systems having a significant excess number of transmitter antennas.

In this chapter, two novel detectors are proposed termed the Reduced Dimension MAP (RDMAP) detector and Group MAP (GMAP) detector, which are both based on a group detection strategy. These detectors bridge the performance gap between MAP and MMSE detectors in systems with an excess number of transmitter antennas and are developed for Quadrature Amplitude Modulation (QAM). The RDMAP and GMAP detectors operate by dividing the symbol vector into two groups: a MAP group and an interfering group. In the RDMAP detector, the MAP and interfering groups are formed
dynamically for each bit decision. The GMAP detector reduces the complexity of the RDMAP detector by using a static set of disjoint MAP groups and jointly produces a bit decision for each bit in the group. A novel greedy algorithm is proposed to form the MAP groups for the GMAP detector. For both detectors, the symbols in the interfering group are treated as an interfering noise source that is whitened by applying an appropriate filter. The prior probabilities for the interfering symbols are used to determine the mean of the interfering noise source. The size of the MAP group $N_G$ is an adjustable parameter that determines the detector complexity. Through the choice of this parameter, the RDMAP and GMAP detectors are generalizations of both the MAP detector and the MMSE detector in [116, 117].

An uncoded BER analysis of the RDMAP and GMAP detectors is also presented in this chapter for a quasi-static Rayleigh fading channel. By approximating the MAP decision rule with the ML decision rule, the union bound on the symbol error rate probability can be expressed as the sum of Hermitian quadratic forms in complex Gaussian variables. The BER is obtained from the pdf of summed Hermitian quadratic forms.

4.2 System Model

4.2.1 Layered Space-Time Transmitter

Consider the transmitter structure in Figure 4.1 for a layered space-time architecture [116] having $N$ transmitter antennas. Binary data is demultiplexed into $2NQ$ bit streams that are independently encoded and interleaved. An QAM symbol is generated by a group of $Q$ bit streams using a Gray labelling scheme. The $n^{th}$ modulated output or layer is given by

$$x_n(k) = M_{ary}(b_{(n-1)Q+1}(2k) \ldots b_{nQ}(2k)) + \sqrt{-1}M_{ary}(b_{(n-1)Q+1}(2k+1) \ldots b_{nQ}(2k+1)),$$

where $M_{ary}$ is the Gray coded M-ary modulating function whose output is $\{-2^{Q-1} - 1, -2^{Q-1} + 3, \ldots, 2^{Q-1} + 1\}$. For QPSK, the $n^{th}$ layer modulated output is given by

$$x_n(k) = \{2b_n(2k) - 1\} + \sqrt{-1}\{2b_n(2k + 1) - 1\}.$$
is passed through a modulo-$N$ shifter that changes every $\tau$ seconds and with no loss of
generality, $\tau$ is set equal to the symbol period. The transmitted signal on antenna $n$ is
given by $\tilde{s}_n(k) = \tilde{x}_\alpha(k)$, $\alpha = (n - k) \mod N$.

### 4.2.2 Channel Model

Consider a flat fading MIMO channel with $N$ transmitter antennas and $M$ receiver
antennas sampled at symbol rate. The channel output at the $m^{th}$ receiver antenna at
the $k^{th}$ time index is given by

$$\tilde{r}_m(k) = \sum_{n=1}^{N} \tilde{h}_{mn}\tilde{s}_n(k) + \tilde{v}_m(k) \tag{4.1}$$

where $\tilde{h}_{mn}$ is the channel gain between the $n^{th}$ transmitter antenna and the $m^{th}$ re-
ceiver antenna, and $\tilde{v}_m(k)$ is an AWGN noise source of variance $\sigma^2$. Define the signal-
to-noise power ratio (SNR) at the channel output as $\text{SNR} = \sum_{n=1}^{N} E|\tilde{h}_{mn}|^2 E_s/\sigma^2 = N E|\tilde{h}_{mn}|^2 E_s/\sigma^2$, where $E_s = E[|\tilde{s}_n(k)|^2] = 2$ for QPSK. Stacking measurements in the
spatial domain results in

$$\tilde{r}(k) \equiv [\tilde{r}_1(k), \ldots, \tilde{r}_M(k)]^T = \tilde{H}\tilde{s}(k) + \tilde{v}(k) \tag{4.2}$$

where $\{\tilde{H}\}_{mn} = \tilde{h}_{mn}$, $\tilde{s}(k) = [\tilde{s}_1(k), \ldots, \tilde{s}_N(k)]^T$ and $\tilde{v}(k) = [\tilde{v}_1(k), \ldots, \tilde{v}_M(k)]^T$. It is
convenient to transform the complex channel equation in (4.2) into real matrix equation

$$\mathbf{r}(k) = \mathbf{H}\mathbf{s}(k) + \mathbf{v}(k) \tag{4.3}$$
where \( r(k) = [\Re\{\tilde{r}^T(k)\}]\Im\{\tilde{r}^T(k)\}]^T \), \( s(k) = [\Re\{\tilde{s}^T(k)\}]\Im\{\tilde{s}^T(k)\}]^T \), \( v(k) = [\Re\{\tilde{v}^T(k)\}]\Im\{\tilde{v}^T(k)\}]^T \) and

\[
H = \begin{bmatrix}
\Re\{\tilde{H}\} & -\Im\{\tilde{H}\} \\
\Im\{\tilde{H}\} & \Re\{\tilde{H}\}
\end{bmatrix}
\]  \tag{4.4}

is the \( 2M \times 2N \) real channel matrix.

### 4.2.3 IDD Layered Receiver

The block diagram for the turbo processing BLAST receiver is shown in Figure 4.2. The receiver consists of a SISO symbol detector, a set of \( 2NQ \) SISO channel decoders, and an interleaver and deinterleaver between each decoder and the detector. There are modulo-N shifters at the input and output of the detector that have been omitted from Figure 4.2 for clarity. In each iteration, the detector produces an extrinsic output \( \lambda^E[b_n] \) for each bit in the symbol vector based on the channel observation and the set of intrinsic inputs \( \{\lambda'[b_n]\} \). After a sufficient number of iterations, estimates for the uncoded bits \( \{\hat{c}_n\} \) are obtained. The channel decoders can be efficiently implemented using the SISO APP module from [82].
4.3 Detector Design

From (4.3), the signal component of the real value channel observation $r(k)$ is determined by a symbol vector $s(k)$ that is formed from a bit vector $b = [b_1, \ldots, b_{NQ}]^T$. Let $S$ be the set of possible $s(k)$, and let $S^n_i$ be a subset of $S$ having $b_n = i$, $i \in \{0,1\}$. The APP for a coded bit $b_n$ can be evaluated in the log domain as

$$\lambda^E[b_n] = \log \frac{P(b_n = 1|r(k))}{P(b_n = 0|r(k))} = \log \frac{\sum_{s \in S^n_1} P(s) \exp \left( -\frac{\|r(k) - Hs\|^2}{2\sigma^2} \right)}{\sum_{s \in S^n_0} P(s) \exp \left( -\frac{\|r(k) - Hs\|^2}{2\sigma^2} \right)} - \lambda^T[b_n]$$

(4.5)

The evaluation of (4.5) can be computationally prohibitive for systems with a large number of transmitter antennas. There are a total $|S| = 2^{2NQ}$ terms to be summed in the numerator and denominator of (4.5). In the next two subsections, two reduced complexity detectors are derived using the principle of reducing the size of the symbol vector $s$.

4.3.1 Reduced Dimension MAP Detector

In order to reduce the number of terms to be summed in the MAP decision, a subset of the symbol vector $s(k)$ is used and the remaining signal contribution is treated as interference. For a signal vector $s = s(k)$, define the MAP group as the set $G = \{\alpha_1, \ldots, \alpha_{N_G}\}$ of $N_G$ integers corresponding to indices of elements in $s$ used in the MAP decision. Further define an interfering group as a complimentary set $\bar{G} = \{\beta_1, \ldots, \beta_{N_G}\}$ of $N_G$ integers corresponding to indices of elements in $s$ such that $G \cap \bar{G} = \emptyset$ and $G \cup \bar{G} = \{1, \ldots, 2N\}$. For a particular choice of $G$, the channel output can be expressed as

$$r = H_G s_G + H_{\bar{G}} s_{\bar{G}} + v$$

(4.6)

where $s_G = [s_{\alpha_1}, \ldots, s_{\alpha_{N_G}}]^T$ is the reduced dimension signal vector, $s_{\bar{G}} = [s_{\beta_1}, \ldots, s_{\beta_{N_G}}]^T$ is the interference vector, $H_G = [h_{\alpha_1}, \ldots, h_{\alpha_{N_G}}]$, $H_{\bar{G}} = [h_{\beta_1}, \ldots, h_{\beta_{N_G}}]$, and $h_i$ is the $i^{th}$ column of $H$. The time index $k$ has been omitted for clarity. The contribution
of the interference and Gaussian noise can be treated as a colored noise source. Let \( \mathbf{w} = \mathbf{H}_G \mathbf{s}_G + \mathbf{v} \) be the colored noise source whose mean is \( \mathbf{\bar{w}} = \mathbb{E}[\mathbf{w}] = \mathbf{H}_G \mathbf{\bar{s}}_G \), where \( \mathbf{\bar{s}}_G = [\hat{s}_{G1}, \ldots, \hat{s}_{GN_G}] \) and \( \hat{s}_{Gi} = \mathbb{E}[s_{Gi}] \) is evaluated using the prior probabilities from the channel decoders as

\[
\mathbb{E}[s_{Gi}] = \sum s_{Gi} P(s_{Gi}) \tag{4.7}
\]

For a symbol mapping \( s = M_{ary}(c_1 \ldots c_Q) \), the symbol probability is \( P(s) = \prod_{n=1}^{Q} P(b_n = c_n) \), with \( P(b_n = 0) = 1/(1 + \exp(\lambda^I[b_n])) \). For QPSK modulation (4.7) reduces to \( \mathbb{E}[s] = \tanh(\lambda^I[b]/2) \). The covariance of \( \mathbf{w} \) is given by

\[
\mathbf{R}_w = \mathbb{E}[(\mathbf{w} - \mathbf{\bar{w}})(\mathbf{w} - \mathbf{\bar{w}})^T] = \mathbf{H}_G \mathbf{\Omega} \mathbf{H}_G^T + \mathbf{I}\sigma^2/2 \tag{4.8}
\]

where \( \mathbf{\Omega} = \text{diag}(\omega^2_1, \ldots, \omega^2_{N_G}) \) and \( \omega^2_i = \mathbb{E}[|s_{Gi} - \hat{s}_{Gi}|^2] = \mathbb{E}[s_{Gi}^2] - \mathbb{E}[s_{Gi}]^2 \). For QPSK, \( \omega^2_i = 1 - \hat{s}^2_{Gi} \). Perfect interleaving is assumed in (4.8) so that the elements of \( \mathbf{\bar{s}}_G \) are independent. The noise \( \mathbf{w} \) can be whitened by first removing the mean \( \mathbf{\bar{w}} \) and then applying an appropriate noise whitening filter \( \mathbf{F} = \mathbf{\Sigma}^{-1/2} \mathbf{Q}^T \), where \( \mathbf{\Sigma} \) is a diagonal matrix and \( \mathbf{Q} \) is an orthogonal matrix, both obtained from the eigenvalue decomposition of \( \mathbf{R}_w = \mathbf{Q}\mathbf{\Sigma}\mathbf{Q}^T, \mathbf{Q}\mathbf{Q}^T = \mathbf{I} \). The whitened channel observation is given by

\[
\mathbf{y} = \mathbf{F}(\mathbf{r} - \mathbf{\bar{w}}) \tag{4.9}
\]

Under the assumption the noise at the filter output is Gaussian, the extrinsic output for the bit \( b_n \) in the mapping \( \mathbf{s}_G = [M_{ary}(b_1, \ldots, b_{N_G}), \ldots, M_{ary}(b_{1+(N_G-1)Q}, \ldots, b_{N_GQ})]^T \) can be evaluated as

\[
\lambda^E[b_n] = \log \frac{P(b_n = 1|\mathbf{y})}{P(b_n = 0|\mathbf{y})} = \log \frac{\sum_{\mathbf{s}_G \in S^1_{G1}} P(\mathbf{s}_G) \exp \left(-\frac{||\mathbf{y} - \mathbf{F}\mathbf{H}_G \mathbf{s}_G||^2}{2}\right)}{\sum_{\mathbf{s}_G \in S^0_{G0}} P(\mathbf{s}_G) \exp \left(-\frac{||\mathbf{y} - \mathbf{F}\mathbf{H}_G \mathbf{s}_G||^2}{2}\right)} - \lambda^I[b_n] \tag{4.10}
\]

where \( S^i_{Gi} \) is the set of possible vectors \( \mathbf{s}_G \) having \( b_n = i \). The noise at the filter output is uncorrelated, but it is not Gaussian since the interference has a non-Gaussian pdf. This is especially true at high SNR when the noise contribution is almost entirely from interference. The extrinsic output in (4.10) tends to be over confident, because of the
assumption of Gaussian noise at the filter output. This bias can cause the IDD receiver to diverge. In order to prevent divergence, the prior LLRs are prescaled according to $\lambda'[b] \leftarrow \text{sgn}(\lambda'[b])|\lambda'[b]|^\kappa$, where $0 < \kappa \leq 1$. Prescaling the prior LLRs leads to an over-estimation of the noise power from interfering symbols and compensates for the non-Gaussian noise pdf. The prescaling function acts as a soft clipping function, whose parameter value depends on the modulation format and will be specified in Section 4.5.

The choice of the groups $G$ and $\bar{G}$ is critical to the performance of the RDMAP detector. Consider forming $G$ and $\bar{G}$ for bit decisions corresponding to a symbol $s_i$. The biggest factor affecting the bit decision in (4.10) is the noise enhancement or signal attenuation associated with the noise whitening filter $F$. The noise whitening filter scales down the channel output in directions where the noise power is high and scales up the channel output in directions where the noise power is low, while maintaining an uncorrelated noise output. In order to minimize signal attenuation, the group $\bar{G}$ should be formed such that the noise power in the direction of the signal $h_i$ is minimized. In the channel observation space, the noise power from an interfering symbol $s_j, j \neq i$, projected onto the subspace of $s_i$ is given by

$$\varepsilon_j^2 = |h_i^T h_j|^2 \omega_j^2$$

(4.11)

The value of $\varepsilon_j^2$ is the projected noise variance caused by $s_j$, if it were considered in isolation. Although the total noise power in the subspace of $s_i$ is not simply the sum of $\varepsilon_j^2$'s, it still remains that symbols having a high $\varepsilon_j^2$ will lead to high noise enhancement if they are placed in the interfering group. Accordingly, $\bar{G}$ is formed by the $N_{\bar{G}}$ smallest $\varepsilon_j^2$'s and $G$ is formed by the $N_G$ largest $\varepsilon_j^2$'s.

The MAP detector and MMSE detector in [116, 117] are special cases of the RDMAP detector. Consider the following lemmas.

\textit{Lemma 4.1:} The RDMAP detector and MAP detector are equivalent for $N_G =$
2N. \footnote{The proofs for both lemmas can be found in the Appendix A.1.}

lemma 4.2: The RDMAP detector and MMSE detector in [116, 117] are equivalent for $N_G = 1$.

4.3.2 Group MAP Detector

The bulk of the RDMAP detector complexity lies in evaluating $r_G = FH_G s_G$ for all $s_G \in S_G$ in (4.10). The RDMAP detector uses each $r_G$ only once to produce a single bit decision. A bit decision could be produced for all the bits used to form $s_G$, leading to an approximate complexity reduction of $1/N_G$. This is the motivation behind the GMAP detector.

For a group $G$, the GMAP detector produces a log domain APP according to (4.10) for each bit $b_i, i = 1 \ldots N_G Q$ corresponding to the elements of $s_G$. With no loss of generality, it is assumed that the symbol vector $s$ can be divided into $N_\Psi$ disjoint groups of equal size denoted $\Psi = \{G_1, \ldots, G_{N_\Psi}\}$, such that $a \neq b \forall a \in G_i, b \in G_j, i \neq j$. At this point, it can be noted that RDMAP and GMAP detectors are equivalent in the limiting case when $N_G = 1$ and $N_G = 2N$. Based on lemmas 1 and 2, it can thus be concluded that the GMAP detector is also a generalization of both the MAP detector and MMSE detector in [116, 117].

What remains in the development of the GMAP detector is an algorithm to choose the groups in $\Psi$. Like the RDMAP detector, a correlation-based grouping algorithm is used, with a modified weighting criterion. Consider, for example, a situation in which there is a small angle between the vectors $h_i$ and $h_j$, but the magnitude of $h_j$ and/or $\omega_j^2$ is small. The RDMAP detector would likely include $s_i$ in the MAP decision group for $b_j$, but not include $s_j$ in the MAP decision group for $b_i$. In a joint detection structure, what is required is a cost metric that is symmetric to whether a decision on $b_i$ is made with interference from $s_j$ or vice-versa. To this end, let $R$ be the normalized correlation...
matrix whose entry in row \(i\) and column \(j\) is given by

\[
r_{ij} = \frac{|\mathbf{h}_i^T \mathbf{h}_j|}{\sqrt{||\mathbf{h}_i||^2||\mathbf{h}_j||^2}}
\]  

(4.12)

The element \(r_{ij}\) is the normalized correlation between the symbols \(s_i\) and \(s_j\) at the channel output. We note that if the \(\omega_j^2\) in (4.11) were used as weights, they would be cancelled in the normalization in (4.12).

Given a normalized correlation matrix \(\mathbf{R}\), consider forming \(\Psi\) using the following objective function

\[
\Theta = \arg\max_{G_1 \ldots G_{N_{\Psi}}} \sum_{k=1}^{N_{\Psi}} \arg\max_{j \in G_k} r_{ij}
\]  

(4.13)

which is equivalent to maximizing the maximum pairwise correlation among the members of each group, averaged over all groups. An exact solution to (4.13) requires an exhaustive search over all possible groupings and is computationally infeasible for large \(N\). An approximate solution to (4.13) can be found using the following greedy algorithm

\begin{verbatim}
1  R ← R − I;
2  Ψ ← \{G_1, \ldots, G_{2N}\},  G_i = \{i\};
3  while |Ψ| > N_{Ψ}
4    if \(\sum_{i=1}^{\midΨ\mid} I(|G_i| > 1) < N_{Ψ}\)
5      \(δ_{ij} = I(|G_i| + |G_j| ≤ N_G)\);
6    else
7      \(δ_{ij} = I(2 < |G_i| + |G_j| ≤ N_G)\);
8  end if
9  \(r_{kl} = \arg\max_{ij} r_{ij} δ_{ij}\);
10  G_k ← G_k \cup G_l;
11  Ψ ← Ψ \{G_l\};
12  for n = 1 \ldots |Ψ|, n ≠ k;
13    \(r_{nk} = r_{kn} = \max\{r_{nk}, r_{nl}\} \);
\end{verbatim}
end for
Remove $k^{th}$ row and column from $R$
end while

where $I(x)$ is an indicator function that evaluates to 1 if $x$ is true and 0 otherwise. The preceding algorithm starts with $2N$ groups with one element in each and finishes with $N_{\Psi}$ groups with $N_G$ elements in each group. At each iteration, two groups are merged. The list of possible groups mergers is determined in lines 4-8, and a greedy decision is made to merge the two groups having maximum pairwise correlation. The preceding algorithm is greedy as it attempts to form highly correlated groups first. Although this may not optimally satisfy the objective function in (4.13), it is advantageous to form highly correlated groups first, since the BER is more strongly influenced by groups with a high pairwise correlation instead of by the average pairwise correlation across all groups.

4.4 BER Analysis

This section presents an analysis of the uncoded BER for the RDMAP and GMAP detectors in a quasi-static Rayleigh fading environment. The channel coefficients $\{\tilde{h}_{mn}\}$ are assumed to have an independent Rayleigh fading distribution with unity variance, $\text{E}[\|\tilde{h}_{mn}\|^2] = 1$. Our analysis is based on expressing the symbol error rate as a Hermitian quadratic form in complex Gaussian random variables, and approximating the MAP detection rule with the ML detection rule. A similar approach has been used in the context of multiuser detection to analyze the performance of the maximum likelihood detector [118] and the optimal ratio combiner [119].

The RDMAP and GMAP detectors produce an $a$ posteriori bit probability in (4.10), with error probability $P_e = P(\lambda^A[b_n] < 0|b_n = 1)$, where $\lambda^A[b_n] = \lambda^I[b_n] + \lambda^E[b_n]$ is the $a$ posteriori output. The summations in the numerator and denominator of (4.10) are dominated by the term corresponding to the $s_G$ of closest Euclidean distance to
the channel observation in the noise whitened space. Approximating the summation operators in (4.10) with max operators yields the ML decision rule, whose decision error probability is given by

\[ P(\lambda^A(b_n) < 0| b_n = 1) \approx P \left( \max_{s \in S_{G1}^n} \left( \frac{-\|y - FH_Gs\|^2}{2} \right) - \log \left( \max_{s \in S_{G0}^n} \left( \frac{-\|y - FH_Gs\|^2}{2} \right) \right) < 0 \right) \]

(4.14)

\[ = P \left( \min_{s \in S_{G1}^n} \|y - Gs\|^2 > \min_{s \in S_{G0}^n} \|y - Gs\|^2 \right) \]

(4.15)

\[ = P \left( s \in S_{G0}^n | \min_{s \in S_{G1}^n} \|y - Gs\|^2 \right) \]

(4.16)

where \( G = FH_G \) and \( S_G \) is the set of all \( s_G \). The ML decision rule chooses the group vector that is closest to the received vector in the noise whitened space. For a transmitted group vector \( s_i \in S_{G1}^n \), an erroneous decision is made if the ML detector chooses an erroneous symbol vector \( s_j \in S_{G0}^n \), which will occur if \( C_j < C_i \), where \( C_i \) is the cost metric associated with a candidate group vector \( s_i \) defined by \( C_i \triangleq \|y - Gs_i\|^2 \). The probability of this pairwise error event is denoted \( P(D_{ij} < 0|s_i) \), where \( D_{ij} = C_j - C_i \).

The union bound on the probability of a detection error for a transmitted vector \( s_i \), is given by the sum of the pairwise error probability over the set of received vectors \( s_j \in S_{G0}^n \). Averaging over the set of transmitted vectors yields an upper bound of the error probability that is given by

\[ P_e = E_{s_i \in S_{G1}^n} \left[ \sum_{s_j \in S_{G0}^n} P(D_{ij} < 0| s_i) \right] \]

(4.17)

In order to evaluate (4.17), \( D_{ij} \) can be expressed as a hermitian quadratic form in complex Gaussian variables. Although it is easier to represent \( D_{ij} \) in a real valued hermitian quadratic form, such an expression does not allow for easy factorization, as will be discussed later. First, the group detector derived in Section 4.3 needs to be represented in complex variables.
The real channel matrix consists of orthogonal pairs of column vectors \( h_n, h_{n+N} \). Since both the RDMAP and GMAP detectors form the MAP group \( G \) based on the correlation between columns, it is unlikely that \( h_n \) and \( h_{n+N} \) will both be in \( \mathbf{H}_G \). Under this assumption, each column in \( \mathbf{H}_G \) is orthogonal to one column in \( \mathbf{H}_G \) and approximately independent of the remaining columns of \( \mathbf{H}_G, \mathbf{H}_G \). The independence would be exact if the MAP group was formed by randomly choosing independent columns of \( \mathbf{H} \). If the set of interfering sources \( \{ s_{\beta_n} h_{\beta_n} \} \) is projected onto a column \( h_{\alpha_i} \) of \( \mathbf{H}_G \), there will be one interfering source with no noise contribution and a noise contribution from the remaining \( 2N - N_G - 1 \) interfering sources, each of which has a signal subspace, defined by \( \{ h_{\beta_n} \} \), that is independent of \( h_{\alpha_i} \). If the same projection is applied to another column \( h_{\alpha_j} \) of \( \mathbf{H}_G \), there will be a different interfering source with no noise contribution and a noise contribution from the remaining \( 2N - N_G - 1 \) sources. In order to facilitate analysis, it is assumed that there are \( N_I = 2N - N_G - 1 \) interfering sources whose signal subspace is independent of \( \mathbf{H}_G \) leading to an approximate channel model

\[ \mathbf{r} \approx \mathbf{H}_G \mathbf{s}_G + \mathbf{H}_I \mathbf{s}_I + \mathbf{v} \quad (4.18) \]

where \( \mathbf{s}_I = [s_{I1}, \ldots, s_{IN_I}]^T \) is an interference vector that consists of independent elements of unity variance, and \( \mathbf{H}_I \) is a \( 2M \times N_I \) Gaussian matrix that is independent of \( \mathbf{H}_G \). Since the grouping criterion for the RDMAP and GMAP detectors is based on the assumption of \( N_I \) independent interfering sources, one should expect the uncoded BER of both detectors to be quite similar. Now consider the case where \( N_I \) is even, letting \( N_{Ic} = N_I/2 \) and impose a structure on \( \mathbf{H}_I \) of the form

\[ \mathbf{H}_I = \begin{bmatrix} \mathbf{H}_{Ir} & -\mathbf{H}_{Ii} \\ \mathbf{H}_{Ii} & \mathbf{H}_{Ir} \end{bmatrix} \quad (4.19) \]

where \( \mathbf{H}_{Ir}, \mathbf{H}_{Ii} \) are \( M \times N_{Ic} \) matrices of independent Gaussian entries with variance 1/2. The structure imposed in (4.19) is needed to represent the real-valued system in complex variables. One may note that the \( 2N - 2N_G \) columns of \( \mathbf{H}_G \) that are not orthogonal to
\( \mathbf{H}_G \) can be represented in a form similar to (4.19). Using (4.19), the real-valued channel model in (4.18) can be equivalently represented in complex variables as

\[
\mathbf{\tilde{r}} = \mathbf{H}_G \mathbf{s}_G + \mathbf{H}_I \mathbf{s}_I + \mathbf{\tilde{v}} 
\]  

(4.20)

where \( \mathbf{H}_{Gc} = \mathbf{H}_{Gr} + i \mathbf{H}_{Gi} \) such that \( \mathbf{H}_G = [\mathbf{H}_{Gr}^T, \mathbf{H}_{Gi}^T]^T \), \( \mathbf{H}_{Ic} = \mathbf{H}_{Ir} + i \mathbf{H}_{Ii} \) and \( \mathbf{s}_{Ic} = [s_{I1}, \ldots, s_{I(N_G)}]^T + i[s_{I(N_G+1)}, \ldots, s_{IN_G}]^T \). The complex noise correlation matrix is \( \mathbf{R}_{wc} = \mathbf{H}_{Ic}(2\mathbf{I}_{N_G})\mathbf{H}_{Ic}^H + \mathbf{I}_M \sigma^2 \), and the complex noise whitening filter is \( \mathbf{F}_c = \mathbf{\Sigma}_c^{-1/2} \mathbf{Q}_c^H \), where \( \mathbf{\Sigma}_c = \text{diag}[\lambda_1, \ldots, \lambda_M] \) is a real valued diagonal matrix and \( \mathbf{Q}_c \) is an orthogonal matrix, both obtained from the eigenvalue decomposition of \( \mathbf{R}_{wc} = \mathbf{Q}_c \mathbf{\Sigma}_c \mathbf{Q}_c^H ; \mathbf{Q}_c \mathbf{Q}_c^H = \mathbf{I} \). With no loss of generality, assume the elements on the diagonal of \( \mathbf{\Sigma}_c \) are in ascending order.

The noise whitened output is given by

\[
\mathbf{y}_c = \mathbf{\Sigma}_c^{-1/2} \mathbf{Q}_c^H \mathbf{H}_G \mathbf{s}_G + \mathbf{z}_c 
\]  

(4.21)

where \( \mathbf{z}_c \) is a length \( M \) vector of independent elements of unity variance. For the sake of analysis, further assume that the elements of \( \mathbf{z}_c \) have a complex Gaussian distribution. The matrix \( \mathbf{H}_{Gc} \) is rotationally invariant as it has independent Gaussian elements. Accordingly, multiplying \( \mathbf{H}_{Gc} \) by the orthonormal matrix \( \mathbf{Q}_c^H \) does not change its distribution. The combined channel and whitening filter response in (4.21) is thus a function of the random matrix \( \mathbf{H}_{Gc} \) and the random eigenvalue matrix \( \mathbf{\Sigma}_c \). The random eigenvalue matrix \( \mathbf{\Sigma}_c \) is approximated with its expectation \( \mathbf{\hat{\Sigma}}_c = \text{diag}[E[\lambda_1], \ldots, E[\lambda_M]] \) in (4.21), which leads to

\[
\mathbf{y}_c \approx \mathbf{G}_c \mathbf{s}_G + \mathbf{z}_c 
\]  

(4.22)

where \( \mathbf{G}_c \) is a \( M \times N_G \) matrix of zero mean independent complex Gaussian elements, whose element \( \{\mathbf{G}\}_{mn} \) has a variance of \( 1/E[\lambda_m] \). The mean eigenvalue matrix \( \mathbf{\hat{\Sigma}}_c \) can be obtained in one of several ways as discussed in Appendix A.2.

Thus far the complex noise whitened channel output in (4.22) has been developed under the assumption that \( N_I \) is even. The case where \( N_I \) is odd corresponds to having
purely real noise with variance 1/2 for one element of $s_{ic}$. Although the mean eigenvalue matrix $\hat{\Sigma}_c$ can be evaluated for $N_{ic} = N_I/2$ with an odd $N_I$, this may not correspond to having real noise with variance 1/2 in one dimension. The appropriate value of $N_{ic}$ is somewhere between $N_{ic} = \lfloor N_I/2 \rfloor$ and $N_{ic} = \lceil N_I/2 \rceil$. By considering many different values of $M, N$, and $N_G$, it was found that

$$N_{ic} = \begin{cases} N_I/2 & N_I \text{ even} \\ \lfloor N_I/2 \rfloor + 0.2 & N_I \text{ odd} \end{cases}$$ (4.23)

produced a BER curve that had approximately the same slope as the simulated BER at high SNR.

In terms of the complex noise whitened channel output in (4.22), the ML detector cost metric associated with a group vector $s_i \in S_G$ can be expressed as $C_i = \|y_c - G_c s_i\|^2$. The pairwise error metric $D_{ij}$ can be expressed as the sum of $M$ Hermitian quadratic forms

$$D_{ij} = \sum_{m=1}^{M} x_m^H F_{ij} x_m$$ (4.24)

where $x_m$ is a length $N_G + 1$ Gaussian vector defined by $x_m = [y_m; g_{m1}, \ldots, g_{mN_G}]^T$, $F_{ij}$ is a Hermitian matrix defined by $F_{ij} = u_j u_j^T - u_i u_i^T$, and $u_i = [1, -s_i^T]^T$. Each term $d_{ijm} = x_m^H F_{ij} x_m$ in (4.24) is a Hermitian quadratic form in $N_G + 1$ zero-mean complex Gaussian random variates, whose pdf has a two-sided Laplace transform given by $[120],[121, Appendix B]

$$\phi_{d_{ijm}}(s) = \frac{1}{\det(I + sR_m F_{ij})} = \frac{1}{\prod_{k=1}^{M+1} (1 + s\gamma_{mk})}$$ (4.25)

where $R_m = E[x_m x_m^H | s_i]$ is the covariance matrix given by

$$R_m = \begin{bmatrix} 1 + N_G/\lambda_m & s_{i1}/\lambda_m & s_{i2}/\lambda_m & \cdots & s_{iN_G}/\lambda_m \\ s_{i1}/\lambda_m & 1/\lambda_m & 0 & \cdots & 0 \\ s_{i2}/\lambda_m & 0 & 1/\lambda_m & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_{iN_G}/\lambda_m & 0 & 0 & \cdots & 1/\lambda_m \end{bmatrix}$$ (4.26)
and $\gamma_{mk}$ is the $k^{th}$ eigenvalue of $R_m F_{ij}$. As discussed in [118], $R_m$ is in general full rank and $F_{ij}$ has at most two non-zero eigenvalues, thus $R_m F_{ij}$ has in general two non-zero eigenvalues denoted by $\gamma_{m1}, \gamma_{m2}$. Since the Gaussian vectors $x_m$ are independent, the Laplace transform of the pdf of $D_{ij}$ can be expressed as the product of $\phi_{d_{ijm}}(s)$ given by

$$\Phi_{D_{ij}}(s) = \prod_{m=1}^{M} \phi_{d_{ijm}}(s) = \frac{1}{\prod_{m=1}^{M} (1 + s\gamma_{m1})(1 + s\gamma_{m2})} \tag{4.27}$$

with the region of convergence being the vertical strip enclosing the $j\omega$ axes bounded by the closest pole on either side. The function $\Phi_{D_{ij}}(s)$ has up to $2M$ distinct poles at points $p = -1/\gamma_{mk}$. Let $p_k, k = 1, \ldots, N_p$ denote the set of $N_p$ distinct poles of $\Phi_{D_{ij}}(s)$ and let $n_k$ be the multiplicity of pole $p_k$. Through a partial fraction expansion, $\Phi_{D_{ij}}(s)$ can be expressed as

$$\Phi_{D_{ij}}(s) = \sum_{k=1}^{N_p} \sum_{l=1}^{n_k} \frac{a_{kl}}{(s - p_k)^l} \tag{4.28}$$

where

$$a_{kl} = \frac{1}{l!} \lim_{s \to p_k} \left( \frac{d}{ds} \right)^k (\Phi_{D_{ij}}(s)(s - p_k)^n_k) \tag{4.29}$$

At this point it is worth justifying the need for a complex noise whitened channel output. If a real output were used, the $(1 + s\gamma_{mk})$ factors in the denominator of (4.27) would be replaced by factors of the form $\sqrt{1 + s\gamma_{mk}}$ [120], which would not allow for the expansion in (4.28). From tables of inverse Laplace transforms, the pdf of $D_{ij}$ is given by

$$P(x) = \sum_{k=1}^{N_p} \sum_{l=1}^{n_k} a_{kl} \text{sgn}(-p_k) \frac{x^{l-1}}{(l-1)!} \exp(p_k x) u(\text{sgn}(-p_k)t) \tag{4.30}$$

where $\text{sgn}(x)$ is 1 if $x \geq 0$ and $-1$ otherwise and $u(x)$ is 1 if $x \geq 0$ and 0 otherwise. Integrating (4.30) from $-\infty$ to 0 yields

$$P(D_{ij} < 0 | s_i) = \sum_{k=1}^{N_p} \sum_{l=1}^{n_k} \frac{a_{kl}}{(-p_k)^l(l-1)!} \Gamma(l) I(p_k > 0) \tag{4.31}$$

which can be substituted back into (4.17) to evaluate the BER.
Table 4.1: Complexity of RDMAP, GMAP, MAP and MMSE, detectors per coded bit.

4.5 Simulated Results

This section analyzes the coded and uncoded BER performance of the RDMAP detector, and GMAP detector along with the complexity of both detectors. The analysis in the previous section of the uncoded BER is also examined. The MAP detector and MMSE detector in [116, 117] are used as references for comparison in terms of complexity and performance. A complexity analysis for the RDMAP and GMAP detectors is shown in Table 4.1 along with the complexity of the MAP and MMSE [116, 117] detectors for reference. All operations are shown for a single bit decision in terms of the number of Multiply and Accumulate (MAC) operations. It was assumed that the noise free channel outputs \( r = Hs \) were precomputed for the MAP detector, but produced online for the RDMAP and GMAP detectors. The first two terms in the RDMAP complexity corresponds to evaluating \( E[w] \), followed by the complexity of evaluating \( R_w \), its eigenvalue value decomposition, and final the complexity of the applying the noise whitening filter in (4.9). For the GMAP detector, the number of operations is reduced by a factor \( N_G \), since \( N_G \) bits are produced with each group decision. The computational load of the MAP detector grows exponentially with \( N \), while the MMSE detector has cubic complexity based on the matrix inversion operation. The complexity of the proposed RDMAP and GMAP detectors grows exponentially with \( N_G \), and thus it is important to choose a mod-
Figure 4.3: Uncoded BER performance of RDMAP and GMAP detectors for $N = 6, M = 6$ system using QPSK modulation with $N_G = 1, 2, 3, 4$.

erately small $N_G$. For small $N_G$, the complexity of the RDMAP and GMAP detector is polynomial with respect to $N$ and $M$.

Simulations were performed for both QPSK and 16-QAM modulation using an quasi-static Rayleigh fading model with perfect receiver CSI. For the coded system, a rate 1/2 RSC code with generating polynomial $(7, 5)$ was used. For the QPSK simulations, bursts of 200 symbols were used, and final bit decision were produced after 10 turbo iterations with a prescaling constant of $\kappa = 1$. For the 16-QAM simulations, bursts of 400 symbols were used, with 15 turbo iterations and $\kappa = .65$. Increasing the number of turbo iterations lead to minimal addition performance gain. The 16-QAM system was more sensitive to correlated intrinsic input and thus required a non-unity $\kappa$.

The effect of the group size $N_G$ on the uncoded BER performance of the RDMAP and GMAP detectors is shown in Figure 4.3, for a system with $N = 6$ transmitting
antennas and $M = 6$ receiver antennas using QPSK modulation. From lemma 1, the MMSE detector in [116, 117] corresponds to the case of $N_G = 1$ for either detector. The uncoded BER performance of the both the RDMAP and GMAP detectors tends to improve with larger $N_G$, with the RDMAP detector performing slightly better than the GMAP detector especially for larger $N_G$. The RDMAP and GMAP detectors hence have a significant uncoded BER improvement over the MMSE detector [116, 117]. Looking at BER produced by the analysis in the previous section, it is almost exact in the $N_G = 1$ case. For larger $N_G$, the analysis tends to approach the simulated BER from above, with increasing SNR. In the low SNR region, the gap between the analysis and the simulated performance is largely the result of the union bound approximation, and is more pronounced for large $N_G$.

The effect of the number of receiver antennas on the uncoded BER performance is
Figure 4.5: Coded BER comparison of RDMAP, GMAP, MMSE, and MAP detectors using QPSK for $N = 6, M = 3$ system and $N = 6, M = 5$ system. Group size $N_G = 4$ for the RDMAP and GMAP detectors.

shown in Figure 4.4 for a system with $N = 6$ transmitter antennas, a group size of $N_G = 3$ and different number of receiver antennas. As expected, BER performance improves with more receiver antennas, and the RDMAP detector performs slightly better than the GMAP detector. A BER floor at $4 \times 10^{-2}$ exist in the $N = 6, M = 4$ system having fewer receiver antennas than transmitter antennas. The BER produced by analysis tended to provide an approximate upper bound on the BER, with relative tightness for the $N = 6, M = 6$ and $N = 6, M = 8$ systems. The analysis in the $N = 6, M = 4$ system tended to significantly overestimate the BER because of the error floor and the poorness of the union bound approximation when the BER is high. In the uncoded system with fewer receiver than transmitter antennas, both the RDMAP and GMAP detectors reached an error floor.

We now consider the coded BER performance of these detectors in a turbo processing
receiver in systems with fewer receiver antennas than transmitter antennas. For a group size $N_G = 4$, the BER performance of the RDMAP and GMAP detector is shown in Figure 4.5 for a $N = 6, M = 5$ system and a $N = 6, M = 3$ system. The BER performance of the MMSE detector from [116, 117] and MAP detector are also shown for reference. Unlike the uncoded case, there is no error floor, and the turbo processing receiver is able to equalize both systems that have fewer receiver than transmitter antennas. In both the $N = 6, M = 3$ and $N = 6, M = 5$ systems, the MAP detector performed the best, the MMSE detector performed the worst and the RDMAP detector preformed better than the GMAP detector. In the $N = 6, M = 5$ system, the performance gap between the MMSE and MAP detector is small, suggesting the MMSE has near optimal performance. In the $N = 6, M = 3$ system, however, there is a $2dB$ performance gap between the MMSE and MAP detectors, and the RDMAP and GMAP detectors have a performance improvement of approximately 1.3$dB$ and 1$dB$ respectively, over the MMSE detector.

The coded BER as a function of the number of turbo iteration is shown in Figure 4.6 for the RDMAP and GMAP detectors with $N_G = 4$ in a $N = 6, M = 3$ system using QPSK. The coded BER is produced after 1, 3, 5, and 10 turbo iterations is shown. For both detectors, there is a significant BER improvement after the first few turbo iterations, and the incremental improvement is less with each iteration. There is only a 0.3$dB$ performance improvement between the $5^{th}$ and $10^{th}$ iteration.

The effect of group size on the coded BER performance of the RDMAP and GMAP detector is shown in Figure 4.7 as a function of the group size $N_G = 1, 2, 4, 5$ for a system with $N = 10$ transmitter antennas and $M = 4$ receiver antennas using QPSK. The MAP detector is omitted as it is computational infeasible for this system and the MMSE detector [116, 117] is equivalent to the case of $N_G = 1$. The BER performance of the RDMAP and GMAP detectors improves with larger $N_G$, with the RDMAP performing better than the GMAP detector for a given $N_G$. For the case of $N_G = 5$, the RDMAP and
Figure 4.6: Coded BER as a function of the number of turbo iterations for RDMAP and GMAP detectors using QPSK for $N = 6, M = 3$ system with $N_G = 4$. 
Figure 4.7: Coded BER comparison of RDMAP and GMAP detectors for $N = 10, M = 4$ QPSK system with $N_G = 1, 2, 4, 5$.

GMAP detectors have a performance gain of $3dB$ and $2.5dB$ respectively, over the MMSE detector at a nominal BER of $10^{-3}$. The better performance of the RDMAP detector over the GMAP detector stems from the dynamic group selection, which incorporates the intrinsic symbol probabilities in the grouping criteria. As a result, the signal power at the whitening filter output is generally higher for the RDMAP detector.

The coded BER performance of the MMSE, RDMAP and GMAP detectors for 16-QAM modulation is shown in Figure 4.8 for a $N = 4, M = 2$ system for group sizes $N_G = 2, 4$. The MMSE detector was not able to equalize the channel having an error floor at $10^{-1}$. The RDMAP and GMAP detectors could equalize the channel although the GMAP detector with $N_G = 2$ reaches an error floor just above $10^{-3}$. At a nominal BER of $10^{-3}$, the RDMAP detector with $N_G = 4$ had the best performance, with the GMAP detector $N_G = 4$ and RDMAP detector $N_G = 2$ having a $2dB$ and $5dB$ performance loss, respectively. For systems having significantly fewer receiver antennas than
transmitter antennas, the RDMAP and GMAP detectors can provide a significant coded BER performance improvement over the MMSE detector at a computational cost that is significantly smaller than the MAP detector.

4.6 Summary

In this chapter, two novel iterative group detectors are developed that can be applied to layered space-time coding. Both detectors are based on a group detection strategy. The RDMAP detector uses a dynamically formed group for each bit decision, while the GMAP detector uses a static grouping. In both cases, a MAP decision is made using a group of transmitted symbols, treating the remaining signal contribution as interference. The interference is characterized as a non-zero mean colored noise source that is whitened before a decision is made. Both detectors allow a trade-off between complexity and
performance through the MAP group size, and include as special cases both the MAP and MMSE detectors. A novel grouping algorithm is developed for the GMAP detector. The uncoded BER performance of the RDMAP and GMAP detectors has been analyzed, and an approximate upper bound on the uncoded BER is provided. In the uncoded case, the RDMAP and GMAP detectors have a significant performance improvement over the MMSE detector for a moderately chosen group size. In the coded case, the RDMAP and GMAP detectors can have a large performance improvement over the MMSE detector in spatially multiplexed systems having fewer receiver antennas than transmitter antennas.
Chapter 5

Space-Time Coding Over Slow Fading Channels

This chapter examines the design of coded modulation systems in to slow and quasi-static fading scenarios. Mapping and precoding strategies are proposed to mitigate slow fading effects. The coded modulation and detection strategies developed in previous chapters are used where applicable.

5.1 Introduction

The adverse effects of slow and quasi-static fading can be mitigated through multi-dimensional mapping and by the introduction of a spatial precoder. The simplest precoder is a time-varying spatial interleaver, which is employed in the D-BLAST structure. A more general class of spatial precoders is based on unitary transformation matrices that rotate the symbol constellation at arbitrary angles [36, 37, 38, 39, 40, 41, 42]. Fade resistant unitary transformations are often determined through algebraic number theory [122]. Unitary precoding can be used in either MLCM or BICM transmissions and does not significantly affect receiver complexity sinceas it is a linear transformation. A wide range of detectors can thus be applied such as the MMSE detector, APP detector,
sphere detector, or one of the group detectors developed in Chapter 4.

An alternative and complement to linear precoding is multidimensional mapping, wherein each bit is spread across more than one transmitter antennas through a non-linear mapping function. Multidimensional mapping has been employed in coded modulation systems in a couple of ways. In BICM-IDD systems, multidimensional mapping functions have been used to control channel EXIT curve shape to match the outer code [67, 68]. Random multidimensional mapping has also been used to mitigate non-ergodic fading in multi-antenna MLCM-MSD systems [45, 46]. One drawback of multidimensional labelling is that the high complexity APP detector is typically required.

This chapter extends the coded modulation strategies developed in previous chapters to the slow fading conditions. This is achieved by controlling the mapping function and introducing a time-varying spatial precoder. A framework is developed to analyze and optimize system performance in slow and quasi-static fading. For MLCM-MSD transmissions, slow fading leads to fluctuations in the subchannel capacity distribution between channel realizations, causing a capacity loss. For BICM-IDD designs, the shape of the channel EXIT curve similarly varies with channel realizations. Variance introduced by slow fading can be minimized through mapper and precoder choices in both BICM-IDD and MLCM-MSD configurations. A class of time-varying random precoders is proposed as an improvement to spatial interleaving.

The multidimensional random mapping strategy from [45, 46] is considered in this chapter. The generalized LDPC codes and irregular convolutional codes developed in previous chapters are well suited to random mapping. The performance of MSD and IDD receivers is analyzed with random mapping under slow fading. With the APP detector, the IDD receiver is shown to have a performance advantage over the MSD receiver, which is independent of the binary codes employed. In large systems where optimal detection is infeasible, group random mapping is proposed, which can be processed with a group detection strategy similar to that in Chapter 4.
5.2 System Model

Consider a MIMO system with $N$ transmitter antennas and $M$ receiver antennas transmitting across a frequency nonselective channel. Assuming the fading rate is sufficiently slow that the combined channel and pulse shaping filters do not introduce ISI, the complex baseband system can be represented as

$$\tilde{y} = \sqrt{\frac{\rho}{N}} \tilde{H} \tilde{x} + \tilde{v}$$  \hspace{1cm} (5.1)

where $\tilde{y} \in \mathbb{C}^M$ is the received symbol vector, $\tilde{v} \in \mathbb{C}^M$ is the zero mean unity variance AWGN noise vector, $\tilde{x} \in \mathbb{C}^N$ is the transmitted symbol vector with unit power per antenna $||\tilde{x}_m|| = 1$, $\rho$ is the average signal-to-noise (SNR) power ratio at the receiver, and $\tilde{H} \in \mathbb{C}^{M \times N}$ is the complex channel matrix whose element $\tilde{h}_{mn}$ is the gain between the $n^{th}$ transmitter antenna and the $m^{th}$ receiver antenna. The transmit power has been normalized to be the same regardless of the number of transmitter antennas. The channel $\tilde{H}$ is assumed to be known at the receiver, but not the transmitter. Under slow-fading, the set of channel realization over a packet duration can be represented with a set of channel matrices $\{\tilde{H}\}$, which reduces to a single matric under quasi-static fading. It is convenient for receiver side processing to transform the complex channel equation in (5.1) into a real matrix equation

$$y = \sqrt{\frac{\rho}{N}} H x + v$$  \hspace{1cm} (5.2)

where $y = [\Re\{\tilde{y}^T\} \Im\{\tilde{y}^T\}]^T$, $x = [\Re\{\tilde{x}^T\} \Im\{\tilde{x}^T\}]^T$, $v = [\Re\{\tilde{v}^T\} \Im\{\tilde{v}^T\}]^T$ and

$$H = \begin{bmatrix} \Re\{\tilde{H}\} & -\Im\{\tilde{H}\} \\ \Im\{\tilde{H}\} & \Re\{\tilde{H}\} \end{bmatrix}$$  \hspace{1cm} (5.3)

is the $2M \times 2N$ real channel matrix.
5.2.1 Space-Time Coded Modulation Transmission

Consider the block diagram of space-time coded modulation transmitter in Figure 5.1. This is a generalized encoding structure that includes both BICM and MLCM as special cases, and admits hybrid cases. Uncoded bits are multiplexed into $L$ levels that are independently encoded using binary encoders of rates $\{R_i\}$. The coded binary output from the $i^{th}$ level is passed through a random interleaver and grouped into a binary vector $b_i$ of length $P = 2NQ/L$, where $Q$ is the number of bits used in a Q-ary ASK modulation. The mapping function stacks the $2NQ$ level inputs as $b = [b_1^T, \ldots, b_L^T]^T$ and produces a modulated symbol according to $s = M(b)$. QAM modulation is assumed. After applying a precoding matrix, the transmitter output is $x_k = G_k s_k$, where the time index $k$ is introduced to indicate the time-varying property of the precoding matrix $G_k$. Through the appropriate choice of $L$, $P$, $M(b)$ and $G_k$, the proposed structure encapsulates many existing space-time schemes:

- **BLAST**: $L = N$, $P = Q$, $M(b) = [m([b_1 \ldots b_Q]), \ldots, m([b_{(N-1)Q-1} \ldots b_{NQ}])]^T$, where $m([b_1 \ldots b_Q])$ is a ASK modulation function, also termed hybrid coded modulation [123]. Each level corresponds to a layer. For V-BLAST [124] set $G_k = I_N$ and for D-BLAST [1, 125] set $G_k = \Pi(I_N, k \mod N)$, where $\Pi(X, n)$ represents a cyclic shift of $n$ in the column order of $X$.

- **BICM** [20]: $L = 1$, $P = NQ$, $G_k = I_N$. A single code is mapped to all levels. The
mapping function $M(b)$ can be an ASK, or multidimensional QAM [67, 68].

- **MLCM**: $L = NQ$, $P = 1$, $G_k = I_N$. An independent code is used for each level.

The mapping function $M(b)$ can be ASK [123, 126] or multidimensional QAM [45, 46].

- **Multilevel BICM (ML-BICM)** [127]: $L = 2Q$, $P = N/2$, $G_k = I_N$, $M(b) = [m(c_1 \ldots c_{2Q}), \ldots, m(c_{(N/2-1)2Q-1} \ldots c_{NQ})]^T$, $c_n = b_{(N-1)mod(n,Q)+[n/Q]}$. Also known as multilevel coding with spatial multiplexing [128]. Each code is spatially multiplexed and mapped to a particular bit position in a QAM modulation. A slight variation of ML-BICM is Q-staged BICM [129], which maps bits from a common subcode to the same bit position in the in-phase and quadrature-phase signal components.

Random MQAM labelling is form multidimensional QAM labelling that is suited to slow-fading conditions [45, 46]. As discussed in Chapter 3, random MQAM is formed using a pseudo-random lookup table that assign an arbitrary mapping between binary vectors and constellation points. This mapping function changes between symbols and is known at the receiver.

### 5.2.2 Space-Time Receiver

The MSD and IDD receiver structures discussed in previous chapters will be considered. The use multi-dimensional mapping in this chapter puts some limitations on the detector choice, particularly with group detection. The group detectors in Chapter 4 were derived for ASK modulation without spatial precoding. Random precoding requires a group selection algorithm to be applied each symbol, adding additional complexity unless a fixed grouping is used. Multidimensional mapping schemes generally contain non-linear bit groupings that cannot be separated by linear filtering and must be grouped together in the detector processing. The group detector is derived as follows for non-ASK labelling.

Consider using a detector group size of $N_G$, and a possibly multi-dimensional mapping function $m(b)$ whose output spans $M_G$ complex dimensions. We assume $N_G$ is a multiple of $M_G$ and let $N_G = 2NQ - N_G$ be the size of the detector interfering group. Combining
the channel and precoder matrix $\mathbf{C} = \mathbf{HG}_k$, the channel output can be expressed as

$$
\mathbf{r} = \mathbf{C}_G\mathbf{s}_G + \mathbf{C}_G\tilde{\mathbf{s}}_G + \mathbf{v}
$$

(5.4)

similarly to (4.6). The contribution of the complementary group and Gaussian noise is treated as a colored noise source. Let $\mathbf{w} = \mathbf{C}_i\mathbf{s}_G + \mathbf{v}$ be a colored noise source whose mean and variance can be evaluated by partitioning the symbol interference vector as $\mathbf{s}_G = [\mathbf{s}_{G1}, \ldots, \mathbf{s}_{GN_G/M_G}]$ where $\mathbf{s}_{Gi} = \mathbf{m}(\mathbf{b})$. The noise mean is $\bar{\mathbf{w}} = \mathbb{E}[\mathbf{w}] = \sum_{j=1}^{N_G/M_G} \mathbf{C}_{Gj}\tilde{\mathbf{s}}_{Gj}$, where $\mathbf{C}_{Gj}$ are the columns of $\mathbf{C}_G$ corresponding to $\mathbf{s}_{Gj}$, and the covariance matrix is $\mathbf{R}_w = \sum_{j=1}^{N_G/M_G} \mathbf{C}_{Gj}\mathbf{\Omega}_j\mathbf{C}^T_{Gj} + \mathbf{I}\sigma^2/2$, where $\mathbf{\Omega}_j = \mathbb{E}[[\mathbf{s}_{Gj} - \tilde{\mathbf{s}}_{Gj}]^2] = \mathbb{E}[\mathbf{s}^2_{Gj}] - \mathbb{E}[\mathbf{s}_{Gj}]^2$. The expectation of the $n^{th}$ moment of $\mathbf{s}_{Gj}$ can be evaluated as

$$
\tilde{\mathbf{s}}^n_{Gj} = \mathbb{E}[\mathbf{s}^n_{Gj}] = \sum_{\mathbf{b}_j \in \mathbb{B}^{M_G}} \prod_{l=1}^{M_G} (\mathbf{m}(\mathbf{b}_j))^n b_l \exp(\lambda^l[b_l]) + \tilde{b}_l \frac{1}{1 + \exp(\lambda^l[b_l])}
$$

(5.5)

The noise $\mathbf{w}$ can be whitened by first removing the mean $\bar{\mathbf{w}}$ and then applying an appropriate noise whitening filter $\mathbf{F} = \mathbf{\Sigma}^{-1/2}\mathbf{Q}^T$, where $\mathbf{\Sigma}$ is a diagonal matrix and $\mathbf{Q}$ is an orthogonal matrix, both obtained from the eigenvalue decomposition of $\mathbf{R}_w = \mathbf{Q}\mathbf{\Sigma}\mathbf{Q}^T$, $\mathbf{QQ}^T = \mathbf{I}$. The noise whitened channel observation is given by $\mathbf{r} = \mathbf{F}(\mathbf{y} - \bar{\mathbf{w}})$ and the detector output for the $l^{th}$ bit in the binary group vector $\mathbf{b}_G$ is given by

$$
\lambda^E[b_{Gl}] = \log \frac{\sum_{\mathbf{b}_G:b_{Gl}=1} \exp \left( -\frac{||\mathbf{r} - \mathbf{C}_G\mathbf{m}_G(b_{Gl})||^2}{2} \right) + \sum_{\mathbf{m}_G} \log \frac{b_{Gl}e^{\lambda^l[b_{Gl}]} + b_{Gl}e^{\lambda^l[b_{Gl}]} - \lambda^l[b_{Gl}]}{1 + e^{\lambda^l[b_{Gl}]} - \lambda^l[b_{Gl}]} } {\sum_{\mathbf{b}_G:b_{Gl}=0} \exp \left( -\frac{||\mathbf{r} - \mathbf{C}_G\mathbf{m}_G(b_{Gl})||^2}{2} \right) + \sum_{\mathbf{m}_G} \log \frac{b_{Gl}e^{\lambda^l[b_{Gl}]} + b_{Gl}e^{\lambda^l[b_{Gl}]} - \lambda^l[b_{Gl}]}{1 + e^{\lambda^l[b_{Gl}]} - \lambda^l[b_{Gl}]} } - \lambda^l[b_{Gl}]
$$

(5.6)

where $\mathbf{m}_G$ is the arrogate mapping function for the symbols in the group vector.

The GMAP group selection algorithm from Chapter 4 can be applied with non-ASK modulation by modifying the channel correlation matrix $\mathbf{R}$ that is used for GMAP group selection. Let $\theta$ be a set of $M_G$ integers representing indices of the vector $\mathbf{s}$ that are spatially correlated with a group-wise mapping function. Setting all entries of $\mathbf{R}$ to zero, where both the row and column indices are in $\theta$, ensures that the mapping group takes precedence over other groupings. This preprocessing must be applied for all mapping groups in the symbol vector before the GMAP group selection algorithm can be applied.
5.3 Space-Time MLCM Design

MLCM was introduced in Chapter 2 using GLDPC component codes and MSD decoding over the AWGN channel. Over ergodic fading channels, the total capacity, and subchannel capacity distribution is relatively constant between channel uses. Under slow fading, the average channel capacity over a packet is not constant and therefore the subchannel capacities are random entities. A different rate optimization is required to determine subcode rates based on the underlying channel statistics.

The instantaneous unconstrained MIMO channel capacity is 
\[ \tilde{C}_U = \log \det [I_M + \frac{1}{N} \mathbf{HH}^H] \] bps/Hz [1]. For discrete modulation, the modulation constrained capacity of a MIMO fading channel is given by

\[ \tilde{C}_{CM} = I(s; r) = H(s) - E_{s,r} \left( \log_2 \frac{\sum_{s \in S} p(r|s)}{p(r|s)} \right) \]

\[ = NQ - E_{s,r} \left( \log_2 \frac{\sum_{s \in S} \exp(||r - \sqrt{\frac{N}{S}} \mathbf{Hs}'||^2/2)}{\exp(||r - \sqrt{\frac{N}{S}} \mathbf{Hs}||^2/2)} \right) \] (5.8)

where \( S \) is the set of possible transmitted symbols, and equiprobable transmitted symbols and perfect receiver CSI are assumed. For packet based transmission, it is also useful to define the packet averaged channel capacities \( C_U = \mathbb{E}_H[\tilde{C}_U] \) and \( C_{CM} = \mathbb{E}_H[\tilde{C}_{CM}] \), where the expectation is taken over the set channel realizations seen over a packet. In quasi-static fading, the instantaneous and packet averaged capacities have the same statistics, since the channel remains constant for the packet duration. The instantaneous CM capacity can be decomposed into \( L \) parallel subchannels using the chain rule of mutual information as:

\[ I(b; y) = I(b_1; y) + I(b_2; y|b_1) + \ldots + I(b_L; y|b_1, \ldots, b_{L-1}) \] (5.9)

where \( \tilde{C}_l = I(b_l; y|b_1, \ldots, b_{l-1}) \) is the channel capacity associated with the \( l \)th symbol bit, assuming lower level bits are known. Averaging subchannel capacities over a packet yields \( C_l = \mathbb{E}[\tilde{C}_l] \). Over a slow fading channel, the subchannel capacities are random entities whose statistics are influenced by the packet length and fading rates.
For the slow fading channel, the decomposition in (5.9) produces subchannel rates that vary with channel realizations, thus a set of rates that achieves capacity for one channel realization may not for another. Since there is no one rate distribution that achieves capacity for all channel realizations, we choose a distribution that leads to error-free transmission on as many channel realizations as possible. Let $f_{ER}(\{r_l\}, \rho)$ represent FER as a function of the code rates $\{r_l\}$ and SNR $\rho$, and let $\hat{P}_{ER}$ be the design error rate. A frame or packet error is declared if there is an error in any of the levels. The optimal set of rates can be found by solving the following optimization problem:

$$\max \sum_{l=1}^{L} r_l \quad \text{s.t.} \quad f_{ER}(\{r_l\}, \rho) \leq \hat{P}_{ER}, \quad r_l > 0 \quad \forall \ l$$

In [61], the solution to (5.10) is found by assuming that level errors occur independently, in which case, the optimal set of rates $\{r_l^*\}$ is determined by the condition, $P_{ER}^l(\rho, r_l^*) = P_{ER}^1(\rho, r_1^*)$, $\forall \ l$, where is $P_{ER}^l(\rho, r_l^*)$ is the error rate for the $l^{th}$ level. Unfortunately, level errors are not independent since an error on one level indicates the channel may be in deep fade, which would also tend to cause errors in other levels. It follows that solving (5.10) directly leads to better performance than assuming independent level errors. Since (5.10) is usually evaluated over a finite set of channel realizations, $f_{ER}(\{r_l\}, \rho)$ is not a smooth function and so gradient based optimization techniques lead to unstable results. A steepest decent optimization procedure not based on gradients can be used to solve (5.10) by controlling the step-size, and noting that $f_{ER}(\{r_l\}, \rho)$ is monotonically increasing function of $r_l$ and thus has no local minima or maxima. A general approach to solving (5.10) is to increase the rate which least affects the FER for a fixed step size; if none of the rates can be further incremented, reduce the step-size and all the rates by the old step size, and continue. An algorithm to solve (5.10) based on this strategy can be found in Appendix A.3.

The optimization problem in (5.10) is useful for finding an optimal set of rates for different design parameters such as precoders, mapping functions and code structures. It does not, however, give insight into why certain design choices work better than others.
One way to visualize the performance of the multi-antenna MLCM transmissions is to look at the subchannel capacities \( \{c_i\} \) as a function of the CM capacity \( C_{CM}(\rho, \{H\}) \). It is argued in [45, 46] that for an MLCM-MSD transmission to achieve the channel capacity over a quasi-static fading channel, the subchannel capacities must be a deterministic function of the CM capacity as

\[
C_i(\rho, \{H\}) = g_i(C_{CM}(\rho, \{H\}))
\] (5.11)

The proof for (5.11) is found in [46]. Equation (5.11) essentially states that there should be no variation in the distribution of subchannel capacities once the overall capacity is fixed. Although random mapping was found to minimizes the variance in the \( g_i(C) \) functions, these functions are in general probabilistic, with variations that depend on the particular channel realization. A small variance in the \( \{g_i(C)\} \) functions means that different levels are tightly correlated and so either none of the subchannels are in outage, or all or most of the subchannels are in outage. A large variance in the \( \{g_i(C)\} \) functions leads to poor performance since no one set of component code rates will operate close to capacity for different channel realizations.

### 5.3.1 Precoder Design

The precoder in a space-time MLCM transmission serves two purposes. Spectral precoders are traditionally used to minimize degradation of the CM capacity due to a poor alignment between the constellation and channel realization that may arise under slow and quasi-static fading conditions. A second and ostensibly more important precoder function in MLCM-MSD transmissions is to spread each bit across the signal space. The CM capacity is decomposed into a set of random subchannel capacities under staged decoding. Precoding can be used to reduce variance in these subchannel capacity functions thereby improving performance.

As a natural starting point, consider how the unconstrained channel capacity is af-
fected by the insertion of a precoder before the channel. Let $C$ be the unconstrained channel capacity without the precoder averaged over a packet duration. The channel capacity with a precoder matrix $G$ is given by,

$$C_G = E \left[ \log \det \left( I_M + \frac{\rho}{N} H_n G_n G_n^H H_n \right) \right] \leq C$$

(5.12)

where $H_n$ is the channel for the $n^{th}$ symbol. Equality in (5.12) indicates that the precoder incurs no capacity loss and a sufficient condition for equality in (5.12) is $G_n G_n^H = I$, implying that $G$ is unitary. A unitary precoder rotates the modulated symbols across different antennas.

Although the unconstrained capacity is unaffected by unitary precoding, the CM capacity is a function of the constellation orientation and channel realization. Since the channel is itself random, it is difficult to find a single fixed precoder that performs well on all channels. In fact, for a large class of \textit{rotationally invariant} channel such as the independent Rayleigh fading channel [130], the average CM capacity is unaffected by a fixed precoder matrix. Rotationally invariant channels have spatially symmetric fading statistics and are defined by the property that $p(H = \tilde{H}) = p(H = G \tilde{H})$ for some known unitary matrix $G$. Even if a channel is not rotationally invariant due to polarization or antenna geometry, a fixed precoder can only mitigate known asymmetries in the spatial fading model. The alternative to a fixed precoder is a time-varying precoder that rotates the constellation a different angle for each symbol. Even if the constellation for some symbols is aligned with deep fades, other symbols will be better aligned thereby averaging the loss.

One precoding strategy is to spread each modulated symbol over different phases and spatial angles as uniformly as possible. The set of complex random unitary matrices having a uniform prior density are known as \textit{isotropically distributed} unitary matrices. An isotropically distributed unitary matrix has a pdf that is unchanged when a unitary transformation is applied. A discussion of the properties of random unitary matrices can be found in [130]. Such matrices have a density that uniformly spans the space of
possible rotations, and can easily be produced by applying a QR (Gram-Schmidt) factorization to randomly generated complex Gaussian matrices with independently identically distributed entries.

The advantage of isotropic unitary precoding can be understood from the relationship between the unconstrained and CM capacities. Consider a sample $2 \times 2$ quasi-static Rayleigh fading channel that will be discussed in this section and assume a 16-QAM constellation. A scatter plot of the CM capacity versus unconstrained capacity with and without the precoder is shown in Figure 5.2. The CM capacity without precoding falls short of the precoder case for some channels whose instantaneous capacity is close to the maximum transmission rates. As discussed in [131], isotropic unitary precoding maximizes the CM capacity, making it independent of the constellation alignment and only a function of the channel matrix singular values. This result holds for all rotationally invariant channels, and is independent of the coded modulation format.

In addition to maximizing the CM capacity, spatial precoding enables coded modulation schemes with binary codes to exploit this capacity. In slow fading, the most significant factor affecting coded modulation performance is variance in either the sub-channel capacity distribution or detector EXIT curves. This variance can be reduced with precoding and multi-dimensional mapping. When ASK labelling is used, the precoder then becomes the only means to suppress subchannel capacity variance. The simplest and maybe most widely used precoder is the spatial interleaver that is used in the D-BLAST structure.

Spatial interleaving improves slow and quasi-static fading performance by effectively creating a cyclically time-varying channel with $N$ realizations. Each ASK symbol sees $N$ different channel realizations which tends to reduce subchannel capacity variance. This can be seen in Figure 5.3 and Figure 5.4 that shows the MSD subchannel capacity distribution as a function of the CM capacity with and without spatial interleaving. These plots were produced by generating approximately 200 sample channel realizations over
Figure 5.2: CM capacity as a function of the unconstrained capacity for unprecoded and isotropic unitary precoded systems
an SNR range of $[-5, 25\, \text{dB}]$ and plotting the subchannel rates against the CM capacity. Across all levels, the unprecoded case in Figure 5.3 had a significantly higher subchannel capacity variance than the unprecoded case in Figure 5.4.

The spatial interleaver precoder, however, has a couple of limitations. Firstly, only produces $N$ different effective channel realizations and so each bit is spread in $N$ different directions with ASK modulation. Using a more uniform spatial spreading function would tend to reduce subchannel capacity variance. Secondly, the spatial interleaver does not affect the phase of the modulated signal. Since the in-phase and quadrature-phase components may experience different fading conditions, a time-varying phase rotation would tend to average the variation across both component. The aforementioned limitations can be mitigated by introducing random spatial and phase rotations. Isotropic unitary matrices are a good choice because they have a uniform spreading quality. This can be seen in Figure 5.3, which shows the subchannel capacity function for both the spatial interleaving and the isotropic unitary precoder cases. The isotropic unitary precoder has a smaller subchannel capacity variance on all levels.

### 5.3.2 Space-Time Mapping

The precoder and mapper both reduce subchannel capacity variance by introducing correlation between the subchannel capacities. The main difference between these operations is that the latter allows a non-linear transformation between bits and symbols. A non-linear mapping that spans more than one complex dimensions can spread a single coded bit across multiple dimensions within the same symbol. An ASK mapping function only affects the symbol output in a single direction, so multi-dimensional spreading requires multiple timeslots with precoding alone. Non-linear mapping hence affords a greater reduction in subchannel capacity variances than precoding alone.

Random mappings have been shown to perform well over quasi-static fading channels [45, 46]. A random MQAM mapping function spreads symbol bits across all trans-
Figure 5.3: Subchannel capacities as a function of the CM Capacity for unprecoded system
Figure 5.4: Subchannel capacities as a function of the CM Capacity for spatial interleaving and isotropic unitary precoding
mitter antennas. Different random mapping functions can be used on each symbol to ensure further randomness across a packet transmission. The effectiveness of random labelling in reducing subchannel capacity variance can be seen in Figure 5.5, which shows a scatter plot of the subchannel capacities as a function of the unconstrained capacity for 20 different mappings over the sample Rayleigh channel. In comparison to Figure 5.4, there is less subchannel capacity variation for the random mapped system than the ASK mapped system. This improvement comes at the cost of receiver complexity since the randomly labelled transmission requires an APP detector, whereas ASK labelling can be processed with a linear detector.

Variance in the subchannel capacity functions can be further reduced by combining random labelling and isotropic unitary precoding. Also shown in Figure 5.5 is the subchannel capacities for the random labelling with unitary precoding additionally applied. The subchannel capacities are essentially a deterministic function of the unconstrained capacity when random precoding and mapping are used. The slight spread of points in the unprecoded case is due to fluctuation in the CM capacity and not irregularities in the random mapping function. If the subchannel capacities were plotted against the CM capacity, there would be no difference between the two cases. The random labelling strategy proposed in [45, 46] minimizes coded modulation capacity loss with respect to the CM capacity, while the precoder serves to maximizes the CM capacity.

For systems with large constellations or many transmitter antennas, random labelling is not feasible because of detector complexity. In such systems, there are many groupwise random labelling functions, where the bits within a group have random labelling and can be detected using a less complex group detector. The effect of group configurations and size on performance will be explored in Section 5.5 using simulated results.
Figure 5.5: Subchannel capacities as a function of the unconstrained capacity for random labelling with and without random isotropic unitary precoding
5.4 Space-Time Iterative Receiver Design

Thus far design parameters such as the binary codes, precoder and symbol mapping have been analyzed for MLCM-MSD systems. In this section, these design parameters will be analyze under IDD receiver processing.

5.4.1 MLCM Rate Allocation

An IDD strategy uses iterative message passing to approximate the Maximum Likelihood (ML) decoder. It is shown in [8] that if an ML decoder is used, an MLCM-MSD scheme can achieve error-free transmission for a set of rates other than those in (5.10), if the following conditions are satisfied:

\[
\sum_{l=1}^{L} r_l < C_{CM} \\
\sum_{l=1}^{L} r_{\alpha_l} + \cdots + r_{\alpha_n} \leq I(y; \{b_{\alpha_i}\}|\{b_{\bar{\alpha}_j}\}) \ \forall \ n, \ \{\alpha_n\} \in S^{(n)} \tag{5.13}
\]

where \(S^{(n)}\) is a subset of \(\{1, \ldots, L\}\) having \(n\) unique elements and \(\{\bar{\alpha}_j\}\) is the complementary set of \(\{\alpha_i\}\) such that \(\{\alpha_i\} \cup \{\bar{\alpha}_j\} = \{1, \ldots, L\}\) and \(\{\alpha_i\} \cap \{\bar{\alpha}_j\} = \emptyset\). In order to choose these rates, consider the following theorem:

**Theorem 1:** For any channel realizations whose CM capacity is greater than or equal to the transmission rate, an asymptotically long MLCM-MSD transmission with random mapping can achieve error-free transmission for an equal rate assignment if the ML decoder is applied.

**proof:** This is proven using the set of conditions in (5.13). If a particular channel realization is not in outage, then the first condition in (5.13) is satisfied. Expanding the
remaining set of conditions for random labelling yields after some simplification

\[ \begin{align*}
C_1 & \leq r_i \leq C_N - (C_{CM} - R) \quad \forall i \in S^{(1)} \\
C_1 + C_2 & \leq r_i + r_j \leq C_N + C_{N-1} - (C_{CM} - R) \quad \forall i, j \in S^{(2)} \\
& \vdots \\
\sum_{l=1}^{L/2} C_l & \leq r_{\alpha_1} + \ldots + r_{\alpha_{L/2}} \leq \sum_{l=1+L/2}^{L} C_l - (C_{CM} - R) \quad \forall \{\alpha_i\} \in S^{(L/2)}
\end{align*} \]  

(5.14)

where \( R = \sum_{l=1}^{L} r_l \) and \( S^{(n)} \subset S \) is a set of \( n \) distinct indices. We have used the fact that random labelling is invariant to the bit order, thus \( I(y; b_i|b_{\alpha_1}, \ldots, b_{\alpha_{n-1}}) = C_n \forall \{\alpha_i\} \in S^{(n-1)}, i \neq \{\alpha_i\} \). The first inequality in (5.14) implies that the smallest and largest rates are bounded by the smallest and largest subchannel capacities, and the \( n^{th} \) inequality implies that the sum of the largest and smallest \( n \) rates are bounded by the sum of the smallest and largest \( n \) capacities. One can verify that the set of uniform rates \( \{r_l = C_l\} \) satisfies the conditions in (5.14), with equality in each case for at least one value of \( \{\alpha_i\} \). Taking the expectation over \( S^{(n)} \) in (5.14) does not change the inequalities, and yields a set of constraints

\[ \sum_{l=1}^{n} C_l \leq E_{\alpha_i \in S^{(n)}} [r_{\alpha_1} + \ldots + r_{\alpha_n}] \leq \sum_{l=L-n+1}^{L} C_l - (C_{CM} - R) \quad \forall n, \{\alpha_i\} \in S^{(n)} \]  

(5.15)

Since \( E_{\alpha_i \in S^{(n)}} [r_{\alpha_1} + \ldots + r_{\alpha_n}] = nR/L \), it follows that the set of rates defined by \( \sum_{l=1}^{n} r_l = nR/L \quad \forall \quad n \) or \( r_l = R/L \) satisfies (5.15). Thus, the equal rate assignment for random labelling can asymptotically achieve error-free transmission on all channel realizations not in outage, regardless of the subchannel capacity distribution. This completes the proof. ■

Since the ML decoder performance is only an upper bound on iterative receiver performance, Theorem 1 cannot be directly applied to the iterative receiver. Theorem 1 tells us that for each viable channel realization, there exists a set of equal rate codes that can achieve error-free transmission; it does not, however, imply the same code can achieve error-free transmission on all viable channel realizations or how to design such a code.
An equal rate assignment is a good design rule since it facilitates a simple 2-dimensional EXIT chart analysis. Multi-rate assignments require multi-dimensional EXIT charts that are more difficult to analyze in practice [127]. For an equal rate assignment using identical codes, the code design procedure for MLCM and BICM systems is identical under IDD processing.

5.4.2 EXIT Chart Analysis

Over the AWGN and ergodic Rayleigh fading channels, the detector EXIT curve is constant between channel realizations, leading to a code profile that is asymptotically capacity achieving for all channel realizations. Under slow fading, the detector EXIT curve is not constant and varies with individual channel realizations. In addition, the distribution of detector EXIT curves is dependent on the number of symbols used to evaluate the channel message mutual information. A decode EXIT curve may intersect the detector curve for a certain fraction of channel realizations: this fraction becomes the predicted FER. Thus, the outer code rate needs to be maximized while maintaining a minimal level of FER performance. This optimization can be divided into two steps. The first step is to determine a design channel EXIT curve $I_{\text{ch}}^E(I_{\text{ch}})$ irrespective of the binary code. The area under the design channel curve is maximized such that the fraction of channel realizations leading to intersections does not exceed the prescribed FER. The second step is to construct the outer code based on the design channel EXIT curve. The EXIT chart code design procedures developed in Chapter 2 and Chapter 3 can be directly applied to the design curve produced from the first step.

A design channel curve that minimizes the FER can be found by evaluating $I_{\text{ch}}^E$ at points $x_i \in [0, 1]$ and solving the following optimization problem:

$$\arg \max_{I_{\text{ch}}^E((x_i))} \sum_i I_{\text{ch}}^E(x_i) \text{ s.t. } f_{\text{ER}}(\{x_i\}, \rho) \leq \hat{P}_{\text{ER}}, \ I_{\text{ch}}^E(x_i) > I_{\text{ch}}^E(x_j) > 0 \ \forall \ x_i > x_j,$$

(5.16)

where $f_{\text{ER}}(\{x_i\}, \rho) = E_H[\prod_{x_i} I(I_{\text{ch}}^E(x_i) < I_{\text{ch}}^E(x_i, H, \rho))]$, and $I(x)$ is an indicator function.
that is 1 if $x$ is true and 0 otherwise. The monotonic constraint $I_{ch}^{x^*}(x_i) > I_{ch}^{x^*}(x_j) \forall x_i > x_j$ in (5.16) can be omitted since the solution of (5.16) without this constraint tends to be monotonic in $x_i$ because $I_{ch}^{x}(x_i, H, \rho)$ is monotonic. Without the monotonic assumption, the EXIT chart optimization problem for the quasi-static fading channel in (5.16) is a slight generalization of the MLCM optimization in (5.10). It can also be solved using the procedure in Appendix A.3.

### 5.4.3 Comparison with MSD

The similarity between the EXIT chart optimization in (5.16) and MLCM-MSD rate optimization in (5.10) is worth some discussion. The MLCM-MSD system decomposes the channel capacity into $L = 2NQ$ levels assuming previous levels have been decoded correctly. The EXIT chart analysis decomposes the channel into a set of vertical strips, and the decoding trajectory only passed from one region to the next if the extrinsic mutual information has increased the require amount over the previous regions. The natural question then arises as to which decomposition method produces less variation over channel realizations and how this translates into performance.

We first compare the MSD and IDD receiver structure when the optimal APP detector is applied. For the IDD structure, the available channel capacity is represented by the area under the detector EXIT curve. This detector EXIT curve is necessarily monotonic [19], with a smooth shape that is almost a straight line for many mapping functions and channels. Although the channel EXIT curve may be evaluated at an arbitrary number of points, its shape and the corresponding decoding trajectory are determined by a few degrees of freedom, such as the area, left most point $I_{ch}^E(0)$ and right most point $I_{ch}^E(1)$. These two extrema points are related to the capacities of the first and last sub-channels under multistage decoding. The left point $I_{ch}^E(0)$ is the capacity for the first MSD subchannel, averaged over different bit choices for that subchannel. Similarly, the right most EXIT point $I_{ch}^E(1)$ is equal to the capacity of the last MSD subchannel, av-
eraged over possible bit positions for that subchannel. Conditioned on the CM capacity, the endpoints of the channel EXIT curve have a smaller variance than two subchannels in a MSD receiver. This indicates that the IDD receiver should produce fewer frame errors than the MSD receiver when only the first and last subchannels are considered. Considering errors in the remaining subchannels in only increases the number of frame errors counted in the MSD receiver. It then follows that the when the APP detector is used, the IDD receiver has a performance advantage over the MSD receiver under slow fading.

When a suboptimal detector is employed, there is a reduction in the available channel capacity for either the multistage or EXIT chart decomposition. This capacity loss is a function of the receiver structure, detector type, and mapping choice. The group detector from Section 5.2.2 can be applied in an IDD or MSD receivers with some restrictions on the mapping function. In the MSD structure, the detector uses hard bit values from previously decoded levels whereas the IDD receiver performs partial noise suppression using soft bits values. When a correlated mapping such as group random labelling is employed, partial noise suppression through linear filtering is not very effective. This is because the constellation of the interfering group spans multiple dimensions that cannot be individually suppressed. Thus for IDD processing receivers, ASK labelling should be used, and group random labelling should be employed with the MSD structure. Since group random mappings is more fading resistant than ASK labelling, neither the MSD nor IDD receiver has a systematic advantage in all scenarios. As a general rule, however, the IDD receiver tends to perform better in the low SNR region and for short packet transmissions.

5.5 Simulated Results

The performance of the various coded modulation schemes discussed in this chapter is evaluated in this section using simulated results. Different antenna configurations are
considered with appropriate transceiver design choices that focus on high throughput applications. A variety of theoretical and practical wireless spatial channel models are considered, as well as the effect of doppler fading and channel modelling error. The GLDPC codes from Chapter 2 are used in the MLCM-MSD systems, while the irregular convolutional codes from Chapter 3 are employed in BICM-IDD systems. The group detection and selection strategies from Chapter 4 are also evaluated.

Error rate curves were generated by performing Monte Carlo trials. A minimum of 300 channels were generated and trials were continued until 150 frame errors occurred or 10000 trials were performed. In some cases further trials were performed and results averaged. A maximum of 200 decoder iterations were performed for the GLDPC codes and 30 iterations for the irregular convolutional codes.

In order to perform Monte Carlo trials, a pre-simulation run was required to determine code design parameters. A set of 2000 test channel realizations was generated and transmission blocks of the appropriate length were simulated in order to perform a sub-channel or EXIT chart decomposition. Simulations were performed at reference SNR values spaced by $2dB$. The optimal set of rates, or design EXIT curve, was then calculated at each SNR value for the desired level of FER performance. In the EXIT design, irregular code profiles were generated using the analysis in Chapter 3 at different SNR values until resulting code rate equaled the design rate. For the MSD receiver design, the rate-loss for a finite length code was determined through the subchannel rate function $f_{ER}(\{r_i\}, \rho)$, that was then used in the rate optimization in (5.10). This function can be empirically determined by simulating a GLDPC code of the prescribed length and rate over a binary AWGN channel at different SNR values.

The first channel considered is the sample quasi-static Rayleigh fading channel with two transmitter and receiver antennas. The performance of the BICM-IDD and MLCM-MSD systems is shown in Figure 5.6 for a rate $r = 2$ bits per channel use transmission using QPSK modulation and the APP detector. Blocks of 1000 symbols were used and
Figure 5.6: FER performance of BICM-IDD and MLCM-MSD systems using QPSK over 2 × 2 quasi-static Rayleigh fading channel for different mapping and precoder choices. The design FER was set to $FER = 0.005$. Random labelling was evaluated for both coded modulation schemes with and without isotropic unitary precoding. Natural and Gray labelling were considered for the MLCM-MSD transmission as well as the spatial interleaver precoder. With Random labelling, the BICM-IDD system outperformed the MLCM-MSD system by about 0.3 dB at the design FER for reasons discussed in the last section. The unitary precoder only marginally improved performance by about 0.1 dB for either coded modulation format, suggesting that the CM capacity loss from a fixed constellation is minimal. Gray and Natural labelling with precoding incur a performance loss of approximately 0.3 dB and 0.2 dB from the random labelling case due to greater subchannel capacity variance. Gray labelling slightly outperforms natural labelling with QPSK modulation because the former is a non-linear mapping over two dimensions. In comparison to the spatial interleaver, unitary precoding provided a performance improvement of 0.7 dB.

A pragmatic observation that can be made about the relative performance of the
systems shown in Figure 5.6 is that the poorest performing spatial interleaver performs within just over $2dB$ from the channel cutoff capacity. This may be an acceptable performance loss for many wireless systems, and there is only about $1dB$ additional improvement with the best BICM-IDD configuration. The situation is quite different when 16QAM modulation is used. The performance of the different rate $r = 4$ systems is shown for 1000 symbol blocks in Figure 5.7 for a target level of performance of $FER = 0.005$.

The spatial interleaver with ASK labelling incurs a capacity loss of over $10dB$ at the target FER, while the BICM-IDD and MLCM-MSD systems with random labelling and precoding incurs a loss of about $1.5dB$ and $1.9dB$, respectively. The three ASK labelling schemes essentially had the same performance. In comparison to the random mapped MLCM-MSD system, the precoded ASK labelled system incurred a performance loss of approximately $0.5dB$ and $1.5dB$ with the APP and linear detectors, respectively.

We next consider how performance is affected by the transmission length and choice
Figure 5.8: FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over $2 \times 2$ quasi-static Rayleigh fading channel for different mapping, precoder and detector choices. Target error rate of FER=0.1 of the target FER. The performance of the designs shown in Figure 5.7 are shown in Figure 5.8 for a target level of performance of $FER = 0.1$, and in Figure 5.9 for a shorter transmission block length of 250 symbols. Increasing the target FER did not significantly affect the relative performance of the different designs. The BICM-IDD system performed closer to the cutoff capacity, about $1dB$ at $FER = 0.1$, and the ASK system reached an error floor when the linear detector was used. For the 250 symbol system, performance trends were similar to the longer block length case with a slightly greater horizontal spread. The one exception is the set partitioning labelling, which outperformsed natural and Gray labelling by about $1.2dB$. Set partition labelling has a performance advantage over other ASK labelling schemes for shorter transmissions.

Thus far in this chapter, the number transmitter and receiver antennas has been fixed at two. The performance of a larger $4 \times 4$ system is shown in Figure 5.10, transmitting at a rate of $r = 4$ bits per channel use over a quasi-static Rayleigh fading channel with QPSK.
When random labelling was used, precoding did not measurably affect performance for either coded modulation format, indicating that a fixed constellation has minimal effect on the CM capacity for larger antenna systems. With random labelling, the BICM-IDD system performed about 1\,dB from the cutoff capacity and the MLCM-MSD system incurred an additional 0.4\,dB loss. Gray and Natural labelling had approximately the same performance that was 0.4\,dB and 0.8\,dB worse than the random labelling case for the APP and linear detectors, respectively. Since the Gray mapped QPSK constellation is a non-linear mapping over two complex dimensions, the linear detector was implemented as a group detector over two dimensions. Finally, the precoder provided 0.25\,dB improvement over the spatial interleaver.

Going to the higher rate 16-QAM modulation gives rise to certain computational complexity issues in the 4×4 system, since it is difficult to use the APP detector in Monte Carlo simulations. It follows that random labelling with APP detection is likely too
Figure 5.10: FER performance of BICM-IDD and MLCM-MSD systems using QPSK over $4 \times 4$ quasi-static Rayleigh fading channel for different mapping and precoder choices.

complex for realtime communications hardware. Group random labelling is possible with detector complexity that can be tuned through the group size. There are two strategies to forming groups for isotropic channels. The first is to ensure mapping groups include the in-phase and quadrature-phase components from the same QAM symbol, and second is to group these components in separate QAM symbols. Simulated results indicate that grouping the in-phase and quadrature-phase components separately produces slightly better results for both coded modulation formats. The next question is how large should the labelling groups be, if they should be used at all. The answer depends on the coded modulation.

The performance of the BICM-IDD system is shown in Figure 5.11 for different mapping and detector group sizes over the four antenna Rayleigh channel at rate $r = 8$ bits per channel use. For ASK random mapping, there was $0.25\,dB$ improvement in using a receiver groups size of four, and an additional $0.05\,dB$ improvement if the GMAP group selection strategy from Chapter 4 was used. The choice of ASK labelling has minimal
impact on performance in this system, and random ASK labelling was used for consistency with other cases. For a fixed detector group size, increasing the label group size actually degrades performance. For a detector groups size of \( GS = 4 \), the ASK random labelling system outperformed the size two and four random labelled systems by 0.3 dB and 0.7 dB, respectively. The situation is different for the MLCM systems whose performance is shown in Figure 5.12. Random labelling over four dimensions outperformed set partitioning by 0.25 dB and 0.5 dB for a detector group sizes of \( GS = 4 \) and \( GS = 1 \), respectively. Set partitioning labelling had the best performance of the ASK mapping schemes considered. Finally, unitary precoding provided a 0.9 dB performance improvement over spatial interleaving for Gray labelling, and approximately the same improvement over other ASK labelling schemes, although the results are not shown.

The performance trends of the BICM-IDD and MLCM-MSD systems over the \( 4 \times 4 \) channel are affected by the transmission block length. The best BICM-IDD and MLCM-
Figure 5.12: FER performance of MLCM-MSD system using 16-QAM over 4 × 4 quasi-static Rayleigh fading channel for different mapping, precoder and detector choices, and block size $N_s = 1000$.

MSD systems both performed about $2dB$ from the cutoff capacity, with the BICM-IDD system doing slightly better for a transmission length of 1000 symbols. Shortening the block size to 250 symbols gives the BICM-IDD system a $1dB$ performance gain over comparable MLCM-MSD configurations. It is somewhat surprising that the GMAP group selection was outperformed by the fixed detector groupings in the BICM-IDD system. The reason for this is that group selection strategies introduce more detector message correlation, and this becomes a critical factor for the shorter block lengths. For the MLCM configuration, set partitioning with linear detection performed particularly well, $0.5dB$ worse than the group random mapped system.

In many wireless applications, there may be a different number of transmitter and receiver antennas. Systems with more receiver antennas are easy to equalize, while it has been argued in Chapter 4 that the detector is a key design element in the overloaded scenario with a greater number of transmitter antennas. The performance of different
Figure 5.13: FER performance of BICM-IDD and MLCM-MSD system using 16-QAM over 4 × 4 quasi-static Rayleigh fading channel for different mapping, precoder and detector choices, and block size $N_s = 250$.

designs is shown in Figure 5.14 for the $r = 4$ QPSK case and Figure 5.15 for the $r = 8$ 16-QAM case. In comparison to the other antenna configurations, the detector has a greater influence on performance in overload system, especially under iterative decoding. The APP detector buys a $1.5dB$ improvement over the linear detector in the QPSK modulated MLCM-MSD system, and the group detector $GS = 4$ improves the linear detector performance by $5dB$ in the 16-QAM BICM-IDD system. The performance of the 16-QAM MLCM configuration is less dependent on the detector.

All the examples thus far have assumed quasi-static fading in which the channel is constant over the packet duration. A more realistic scenario accounts for some degree of channel variation over a packet duration due to ambient motion in the environment. For the Rayleigh fading channel model having an infinite number of scatters, velocity in the environment is often characterized using a doppler fading model in terms of the normalized fading rate $\Delta f_{dop} = \frac{f_c v_{max}}{c} T$, where $f_c$ is the carrier frequency, $v_{max}$ is the
Figure 5.14: FER performance of BICM-IDD and MLCM-MSD system using QPSK over $4 \times 2$ quasi-static Rayleigh fading channel for different mapping, precoder and detector choices, and block size $N_s = 1000$.

Figure 5.15: FER performance of BICM and MLCM system using 16-QAM over $4 \times 2$ quasi-static Rayleigh fading channel for different mapping, precoder and detector choices, and block size $N_s = 1000$. 
maximum velocity and $c$ is the speed of light, and $T$ is the symbol time. In an effort to find a working upper bound on $\Delta f_{dop}$, consider choosing parameter value $f_c = 5GHz$, $v_{\text{max}} = 120km/h$ and $f_s = 1/T = 50kHz$. Higher carrier frequencies tend to require a line-of-sight component, the velocity is limited by local driving speeding limits, and the spectral bandwidth is chosen in light of transmission bandwidth and latency demands in next generation systems. These parameter values give a normalized fading rate of about $\Delta f_{dop} = 0.01$. Choosing a normalized fading rate much higher may start to undermine assumptions of perfect receiver CSI and negligible ISI due to the doppler spread in the combined channel and pulse shaping filter response.

The performance of the various BICM-IDD and MLCM-MSD systems is shown for the $2 \times 2$ doppler Rayleigh fading channel in Figure 5.16 for packets of 1000 symbols, and in Figure 5.17 for packets of 250 symbols. A normalized fading rate of $\Delta f_{dop} = 0.01$ and design rate of $r = 4$ was used. An independent Rayleigh fading spatial model was used with time correlation generated through a modified Jake’s model [132]. One may observe that the channel cutoff capacity is roughly $2dB$ farther to the right in the 250 symbol systems. This is because as the transmission window expands to infinity, channel statistics become stationary and performance approaches the independent fading scenario. Other performance trends were similar to previously discussed systems.

The final set of simulations examine different spatial fading phenomena. The Rayleigh fading model is based on an infinite set of isotropic scatter with no line-of-sight component or antenna correlation. The performance of the different systems is shown for the $2 \times 2$ Rican channel with k-factor $k_{Ric} = 1$ in Figure 5.18. The performance is shown for the correlated Rayleigh fading channel in Figure 5.19, with a transmitter and receiver antenna correlation coefficient of $\alpha = 0.5$. In comparison to the Rayleigh channel, antenna correlation reduces performance and the line-of-sight component improves performance at the same SNR. The BICM-IDD and MLCM-MSD code parameters were designed for both the actual spatial fading model as well as the generic Rayleigh fading model. Code
Figure 5.16: FER performance of BICM-IDD and MLCM-MSD system using 16-QAM over $2 \times 2$ doppler Rayleigh fading channel with $\Delta f_{dop} = 0.01$ for different mapping, precoder and detector choices, and block size $N_s = 1000$.

Figure 5.17: FER performance of BICM-IDD and MLCM-MSD system using 16-QAM over $2 \times 2$ doppler Rayleigh fading channel with $\Delta f_{dop} = 0.01$ for different mapping, precoder and detector choices, and block size $N_s = 250$. 
optimization based on the Rayleigh channel incurs no performance loss if random labelling is used, and minimal loss for other labelling formats. In the MLCM-MSD system with set partitioning labelling, the Rayleigh channel optimization outperformed the actual channel optimization. This phenomenon stems from non-Gaussianity in the message density for set partitioning labelling, and has been discussed for non-fading channels in Chapter 2.

Several spatial channel models have been proposed that model different real world wireless channels [24, 23, 25, 26, 27]. The 3rd Generation Partnership Project Group has developed a wireless spatial fading model [26], which classifies cellular wireless channels into scenarios such as suburban macro cell, urban macro cell, or urban micro cell. Antenna properties such as positioning, polarization and mutual coupling can also be modelled. A slightly modified version of this model has been proposed in [27], with a publicly available Matlab implementation. The performance of different BICM and
Figure 5.19: FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over $2 \times 2$ correlated Rayleigh fading channel ($\alpha = 0.5$) for different mapping, precoder and detector choices, and block size $N_s = 1000$.

MLCM systems has been evaluated using the model from [27]. The set of default parameters from [27] was used, except that the symbol frequency was set to $f_s = 400kHz$. At this symbol rate, the urban macro cell environment gives rise to non-negligible ISI due to multipath and will not be considered. Any residual ISI due to multipath for the other two environments is omitted.

The performance of different BICM-IDD and MLCM-MSD designs is shown for the urban microcell environment in Figure 5.20 and for the suburban macrocell environment in Figure 5.21. Both environments have a lower capacity than the Rayleigh fading channel at the same SNR. For the urban microcell environment, the code optimizations for the Rayleigh and actual channel models produced similar results for the cases considered. For the suburban environment, the Rayleigh and actual channel model produced essentially identical results for the BICM-IDD system, but the Rayleigh optimization performed worse in the MLCM-MSD system. The performance loss in the MLCM-MSD Rayleigh
optimization stems from the fact that the uplink of a suburban cell has few scatters, and is not well represented by an infinite scatterer model. The IDD receiver performance was, however, unaffected by channel mismatch when random mapping was used.

5.6 Summary and Conclusions

This chapter has designed different multi-antenna transceivers for slow fading environments. Good coded modulation performance is possible through the use multidimensional random labelling and the introduction of a time-varying spatial precoder. Isotropic random unitary precoding is proposed for both BICM-IDD and MLCM-MSD systems. This precoding strategy significantly outperforms spatial interleaving in layered architectures and does not increase system complexity. If the optimal detector is available, the combination of random unitary precoding and random multidimensional mapping is shown to be the capacity maximizing solution. For larger systems where suboptimal detection
Figure 5.21: FER performance of BICM-IDD and MLCM-MSD systems using 16-QAM over $2 \times 2$ suburban macrocell environment for different mapping, precoder and detector choices, and block size $N_s = 1000$.

is required, group random mapping is proposed with group detection for MLCM-MSD systems.
Chapter 6

Summary and Conclusions

This dissertation has analyzed the design of a single-carrier multi-antenna wireless system under slow-fading conditions. Multi-antenna systems hold great capacity gains in a highly scattered environment that are exploited through an appropriate design coded modulation scheme and receiver structure. Novel contributions are made in several aspects of space-time transceiver design.

6.1 Key Contributions

Under slow and quasi-static fading conditions, coded modulation performance depends largely on precoder and mapper design. Random isotropic unitary precoding is proposed for both MLCM and BICM transmissions. Random precoding allows the use of a low complexity MMSE detector, and provides significant performance gains over spatially interleaving in MLCM-MSD systems. When the APP detector is available, a strategy of applying both random unitary precoding and random multidimensional mapping maximizes fading performance in both BICM-IDD and MLCM-MSD systems. Although multidimensional random labelling in MLCM-MSD transmissions has been discussed elsewhere, their use in BICM-IDD transmissions to combat slow fading is unique, to the best of our knowledge. We also show that under quasi-static fading conditions, the BICM-
IDD structure has a performance advantage over the MLCM-MSD structure, even for long packet transmissions.

The problem of multi-antenna detection has been examined in high dimensional systems where the optimal detector is computational infeasible. Two detectors based on a group detection strategy have been proposed for either MSD or IDD receivers. These detectors bridge the performance/complexity gap between the MMSE and APP detectors. For MLCM-MSD transmissions, group detection is combined with group random labelling yielding improved performance. The work on group detection in Chapter 4 has been published in [96, 97, 98].

Coded modulation design using LDPC codes has been considered for BICM-IDD and MLCM-MSD structures. MLCM-MSD transmissions require subcodes to be design over a wide range of rates when capacity efficient mapping functions are used. A generalized LDPC code structure has been proposed that improves the low-rate performance of conventional LDPC codes. The check node is generalized by adding additional check node parity bit that are connected directly to the channel. This reduces the rate of the LDPC subcode, leading a performance gain in the low rate $R < 1/3$ region. In MLCM-MSD transmissions, the proposed generalized LDPC codes outperform conventional LDPC code construction over a wide range of transmission rates and channels.

Concatenated convolutional codes are another class of widely used binary codes in coded modulation applications. A convolutional code can be made irregular by encoding fractions of bits with different puncture patterns and mother codes of different memory. Fixed size and variable size trellis have been proposed to connected different memory mother codes. In multi-antenna BICM-IDD transmissions, irregular convolutional codes outperform comparable turbo coded and GLDPC coded systems.
6.2 Future Directions

The future directions of this dissertation can be divided into two broad categories. The first relates to improvements in the analysis procedures and component design that would improve over-all system performance. The second category stems from expanding the application scope beyond a single-carrier single-user system transmitting an uncorrelated binary data source.

6.2.1 Methods

There are a number of possible improvements to the analysis and design procedures discussed in this work. Some of these advancements are outlined as follows:

1. The analysis of iterative message passing through EXIT chart is central to the work in this dissertation. One limitation of EXIT chart analysis is the assumption that the message densities are Gaussian, which greatly simplifies the analysis. The Gaussian assumption leads to certain performance limitations. For example, experiments were conducted with parallel concatenated irregular convolutional codes, but the non-Gaussian message density of the irregular code profile led to a mismatch between the actual and predicted code threshold values. One way to improve the accuracy of the EXIT chart design procedure is to use the EXIT chart decoding trajectory in the analysis [133, 134]. If a convergence failure is consistently observed at some early point, then an additional gap can be imposed between the two EXIT curves in the affected region. This strategy was applied to some extent in this work to mitigate error floor problems, although not in any systematic manner. Another way to improve the accuracy of the EXIT chart analysis to assume the message density takes on a parametric distribution. For LDPC codes, EXIT charts have been developed using mix-Gaussian message densities [135] and through finite order moment generation functions [136].
A second major limitation of EXIT charts analysis is the assumption of uncorrelated message densities. Message correlations is particularly problematic for short block length transmissions, making the EXIT chart prediction inaccurate. In this dissertation, the effects of message correlation could only be assessed though brute force Monte Carlo simulations. A systematic analysis of correlation from code, mapper and interleaver components may lead to an improved EXIT model and better overall system performance. Interleaver design has not been considered in this work, and the problem of interleaver design with irregular codes and with IDD processing is worth further study.

2. The generalized LDPC and irregular convolutional codes developed in the dissertation are based on binary component codes. Non-binary convolutional turbo codes [137] and LDPC codes [138] have been developed, with better performance than their binary counter parts for the same block length. Non-binary subcodes can also offer more degrees of freedom in designing an irregular code profile.

3. Modulation has not been a primary focus of this dissertation, as only QAM modulation formats have been considered. One limitation of QAM modulation is that for equiprobable constellation points, there is a capacity loss in the low rate region. This is a large part of the reason why in Chapter 2, the rate \( r = 1 \) 16-QAM GLDPC coded BICM-IDD system did not outperform a comparable QPSK modulated system. QAM modulation formats with irregular level spacing have been proposed [139], leading to a constellation shaping gain for the AWGN channel. This modulation strategy may expand the use of proposed GLDPC codes in low-rate BICM-IDD applications.

Another problem that is related to modulation is the appearance of zero-rate subcodes in MLCM transmissions. In Chapter 5, the \( 4 \times 2 \) 16-QAM MLCM-MSD system with group random labelling had two zero-rate subcodes encoded with a
pseudo-random sequence. There were also other cases of zero-rate subcode. One strategy to avoid zero-rate subcodes is to use an irregular modulation scheme that consists of an aggregate of QPSK and 16-QAM symbols on different antenna. This would facilitate greater control of the subcode rate distribution so that zero-rate subcodes could be avoided.

6.2.2 Applications

By expanding the scope of application of this research, there are a number of possible directions of future study that may lead useful designs. Some of these applications are described as follows:

1. Multimedia applications are a major driving force behind high-bandwidth communication systems. Treating image and video stream data as an uncorrelated binary data source, may not address end-to-end system goals. For example, an image may be divided into different layers, where course and fine image data are mapped to different layers with different levels of protection. Much of the work herein on MLCM-MSD transmissions can be extend to applications involving unequal error protection. Another consideration is how source and channel coding should be applied together, since their separation is only optimal for long bursts over stationary ergodic channel. Shorter packets and slow-fading conditions lead to additional gain from joint source-channel encoding and decoding structures. Some of the work herein on IDD and MSD receiver processing can be expanded to include source decoding.

2. This work has only considered single-carrier transmissions, although multi-carrier modulation is a part of current and next generation wireless networking and cellular standards. OFDM divides a frequency selective channel into a set of parallel frequency-flat channels. The correlation between two channels is a function of the
frequency spacing between them, thus tones with sufficient frequency separation undergo approximately independent fading. A virtual MIMO channel can be constructed over independently fading OFDM tones instead of over multiple antennas, which would produce a block diagonal channel matrix. Over quasi-static and slow fading channels using OFDM, the unitary precoding and multidimensional mapping strategies of Chapter 5 can be applied across tones. One would expect similar performance improvements at the cost of increased detector complexity.

3. The work herein is limited to a simple single-user point-to-point communications systems. More realistic wireless networks have several users that share the physical channel or access to it. Current multiuser systems are either based on OFDM multiple access or Code Division Multiple Access (CDMA). MIMO CDMA may hold some interest developments, as there are many parallels between multi-user and spatially multiplexed systems. Layered architecture have been combined with CDMA spreading codes [106] with group detection [106] over the uplink channel. There is also a similarity between random chip sequences, random precoding and random mapping that is worth some investigations.

Another interesting scenario is where there are several network nodes cooperate to enhance the transmission rate and reliability. Through cooperative coding [140], higher throughput is possible by exploiting the additional spatial diversity of multiple wireless links. There are a number of coding and media access control strategies that may be applied to increase transmission rate. Some possible physical layer designs may include group detection and spatial coding strategies similar to those discussed in this dissertation.
Appendix A

Appendix

A.1 Relationship between RDMAP/GMAP detectors and MAP/MMSE detector

In this appendix, lemmas 4.1, 4.2 from Chapter 4 are proven showing that the RDMAP/GMAP detector is a generalization of both the MAP detector and MMSE detector in [116, 117].

lemma 4.1: The RDMAP detector and MAP detector are equivalent for $N_G = 2N$.

proof: For $N_G = 2N$, $s_G = Ps$ and $H_G = HP^T$, where $P$ is a permutation matrix. Also, $w = v$ is a zero mean Gaussian scaler that has a whitening filter vector $F = 1/\sigma^2$.

The RDMAP decision in (4.10) for a bit $b_i$ corresponds to a symbol $s_i = s_{Gj}$ that can be expressed as

$$
\Lambda_1[b_i] = \log \frac{\sum_{s_{Gj} \in S_G, s_{Gj} = +1} P(s_G) \exp \left( -\frac{||\sigma^{-1} r - \sigma^{-1} HP^T s_G||^2}{2} \right)}{\sum_{s_{Gj} \in S_G, s_{Gj} = -1} P(s_G) \exp \left( -\frac{||\sigma^{-1} r - \sigma^{-1} HP^T s_G||^2}{2} \right)}
$$

(A.1)

$$
= \log \frac{\sum_{s \in S, s_i = +1} P(s) \exp \left( -\frac{||r - Hs||^2}{2\sigma^2} \right)}{\sum_{s \in S, s_i = -1} P(s) \exp \left( -\frac{||r - Hs||^2}{2\sigma^2} \right)}
$$

(A.2)

which is the MAP decision rule in (4.5).

lemma 4.2: The RDMAP detector and MMSE detector in [116, 117] are equivalent for $N_G = 1$.  

144
**proof**: The MMSE detector in [116, 117] uses a combination of soft interference cancellation and a MMSE residual suppression filter. To make a decision for a bit $b_i$, the channel observation after interference cancellation is given by

$$\tilde{r}_i = r - H\hat{s}_i$$ (A.3)

where $\hat{s}_i = [\hat{s}_1, \ldots, \hat{s}_{i-1}, 0, \hat{s}_{i+1}, \ldots, \hat{s}_N]^T$, $\hat{s}_j = \tanh(\lambda_2^b|b_j|/2)$, and QPSK is assumed.

The output $\tilde{r}_i$ is filtered with a residual MMSE filter given by $w_i^T = 2h_i^T R_i^{-1} \tilde{r}$, with $R_i = [H\Omega_i H^T + \sigma^2 I]$, $\Omega_i = \text{diag}(\omega_1, \ldots, \omega_{i-1}, 1, \omega_{i+1}, \omega_N)$ and $\omega_j = E[|s_i - \hat{s}_i|^2] = 1 - \hat{s}_i^2$. The filter output is approximated as a Gaussian distribution to produce an extrinsic LLR given by

$$\lambda_1[b_i] = \frac{2h_i^T R_i^{-1} \tilde{r}_i}{1 - h_i^T R_i^{-1} h_i}$$ (A.4)

For the case where $N_G = 1$, the RDMAP decision in (4.10) can be simplified as

$$\Lambda[b_i] = \log \frac{P(s_i = +1)}{P(s_i = -1)} - \frac{||y - Fh_i||^2}{2} + \frac{||y + Fh_i||^2}{2} = \lambda_2^b[b_i] + 2h_i^T F^T y$$ (A.5)

where the second term in (A.6) is the extrinsic output that can be further simplified as

$$\lambda_1[b_i] = 2h_i^T F^T F(r - \bar{w}) = 2h_i^T R_w^{-1} (r - \bar{w})$$ (A.7)

What is now required to be shown is that (A.7) and (A.4) are equivalent. First note that $\tilde{r}_i = r - \bar{w}$, as it can be readily seen that $H\hat{s}_i = H_G s_G$. What remains is to show that

$$R_w^{-1} = R_i^{-1} (1 - h_i^T R_i^{-1} h_i)^{-1}$$ (A.8)

Substituting $R_w = R_i + h_i h_i^T$ on the l.h.s. of (A.8) and using the matrix inversion lemma \(^1\) with $A = R_i$, $B = -1$ and $X = h_i$ gives the expression on the r.h.s of (A.8) after some simplification. ■

\(^1\)(A + XBX^T)^{-1} = A^{-1} - A^{-1}X(B^{-1} + X^T A^{-1} X)^{-1} X^T A^{-1}
A.2 Evaluation of Noise Correlation Eigenvalues

This appendix discusses several methods to evaluate the expectation of the ordered eigenvalues of the noise correlation matrix $\hat{\Sigma}_c$ from Chapter 4. A direct method to determine $\hat{\Sigma}_c$ is through Monte Carlo simulations. Such a method can be, however, computationally intensive. Alternatively, the eigenvalues of $\hat{\Sigma}_c$ can be expressed as

$$\text{eig} [\hat{\Sigma}_c] = 2 \mathbb{E} \left[ \mathbf{W}(M, N_{\text{ic}}) \right] + \text{diag} [\mathbf{I} \sigma^2]$$  \hspace{1cm} (A.9)

where $\mathbf{W}(m, n)$ is a complex Wishart matrix of the form $\mathbf{W}(m, n) = \mathbf{H}(m, n) \mathbf{H}(m, n)^H$ and $\mathbf{H}(m, n)$ is a $m \times n$ matrix of independent complex zero-mean Gaussian entries with unity variance. The pdf of the ordered eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_m$ of $\mathbf{W}(m, n)$ is given by [141]

$$P(\lambda_1, \lambda_2, \ldots, \lambda_m) = \frac{\pi^{m(m-1)/2}}{\Gamma_m(n) \Gamma_m(m)} \exp \left[ \sum_{i=1}^{m} \lambda_i \right] \prod_{i=1}^{m} \lambda_i^{n-m} \prod_{i<j} (\lambda_i - \lambda_j)^2$$  \hspace{1cm} (A.10)

where $\Gamma_m(a) = \pi^{m(m-1)/2} \prod_{i=1}^{m} \Gamma(a - i + 1)$ is the multivariate gamma function. The expectation of the Wishart matrix eigenvalues can be directly evaluated by marginalizing the joint pdf in (A.10). The number of factors on the right hand side of (A.10) grows with $m!$ and quickly becomes computationally infeasible for moderately large $m$. A computationally more efficient method to find the expected eigenvalues can be adapted from [119, appendix B]. This method exploits the intrinsic symmetry in eigenvalue pdf and converts the $m$ nested integrals (required to find the expectation) into $m$ separate integrals.

The method used in this appendix to find the expected eigenvalues of the Wishart matrix is simpler than the methods previously discussed and it is based on the the asymptotic eigenvalue distribution as $n, m \to \infty$. Letting $m, n$ increase without bound such that $m/n \to \beta$ asymptotically, the empirical distribution of $F(x) = P(\lambda < x)$ of $\mathbf{W}(m, n)/n$ converges to the non-random limit defined by [141]

$$dF(x) = \frac{\sqrt{(x - a(\beta))(b(\beta) - x)}}{2\pi \beta x} + \delta(x) I(\beta > 0) (1 - 1/\beta)$$  \hspace{1cm} (A.11)
for $\beta \neq 1$ where $a(\beta) = (1 - \sqrt{\beta})^2$, $b(\beta) = (1 + \sqrt{\beta})^2$. In the case of $\beta = 1$, one has $dF(x) = (2\pi)^{-1}((4 - x)/x)^{1/2}$ for $0 \leq x \leq 4$. The expectation of the eigenvalues of $W(m, n)$ can be found from (A.11), by dividing the domain of $dF(x)$ into $m$ adjacent regions, such that each region has the same area under $dF(x)$. Each region can be represented by its center of mass under $dF(x)$, denoted $x_i, i = 1 \ldots m$ for the $i^{th}$ region. The $x_i$s can be easily found by numerical integrating (A.11). The expected eigenvalues of $W(m, n)$ can then be approximated as $E[eig[W(m, n)]] \approx ndiag[x_1, \ldots, x_m]$. The use of (A.11) to approximate the eigenvalues of $W(m, n)$ tends to be accurate even for small $m$, especially when $n \geq m$. For the simulated examples in Chapter 5, there was negligible difference in the estimated BER whether the mean eigenvalue matrix $\hat{\Sigma}_c$ was determined through Monte Carlo simulations or by using the asymptotic distribution in (A.11).

A.3 MLCM-MSD rate and EXIT curve optimization over slow fading

This appendix describes a method to solve the MSD rate allocation problem in (5.10) and EXIT chart curve matching problem in (5.16). Both of these problems can be expressed as follows:

$$\arg \max_{\{x_i\}} \sum_i g(x_i) \quad s.t. \quad f_{ER}(\{x_i\}, \rho) \leq \hat{P}_{ER},$$

(A.12)

where $g(x_i)$ and $f_{ER}(\{x_i\}, \rho)$ are monotonically increasing functions. For the MLCM-MSD rate optimization, $\{x_i\}$ are the subchannel rates and $g(x) = x$, and for the EXIT chart optimization $\{x_i\}$ are points on the EXIT graph and $g(x) = I_E^{thv}(x)$. For the EXIT chart optimization, there is additional monotonic constraint $g(x_i) > g(x_j) > 0 \ \forall \ x_i > x_j$ that can be omitted because the design EXIT curve is constrained by channel realization EXIT curves that are monotonic. In order to solve A.12, one can use the fact that $g(x_i)$ and $f_{ER}(\{x_i\}, \rho)$ are monotonically increasing functions. Increasing a value $x_i$ can only increase the overall rate and possibly the constraint function, while reducing $x_i$ will
reduce the rate and possibly the constraint function. The zero-rate or zero-valued EXIT curve solution is always feasible and it is possible to start from this point and choose to increment the $x_i$ that has minimal effect on the error function. The choice of $x_i$ to increment can be found based on subchannel error function $f_{ER}^i(x_i, \rho)$, which is the error rate considering the $i^{th}$ subchannel or EXIT point only. The increment value also needs to be adjusted as the current solution approaches the optimal solution. A program that can be used to solve A.12 is shown as follows:

1. \( \{r_i\} \leftarrow 0; \)
2. \textbf{while } \delta > tol \textbf{ do}
3. \hspace{1em} \( n = \arg \min_i f_{ER}^i(\rho, x_i + \delta); \)
4. \hspace{1em} \textbf{if } f_{ER}^i(\rho, x_n + \delta) < P_{ER} \textbf{ then}
5. \hspace{2em} \( x_n \leftarrow x_n + \delta; \)
6. \hspace{1em} \textbf{else}
7. \hspace{2em} \( \{x_i\} \leftarrow \max(\{x_i\} - \delta, 0), \delta \leftarrow \delta/2; \)
8. \hspace{1em} \textbf{end if}
9. \textbf{end while}

The above procedure iteratively increases the set of points for a fixed increment size while maintaining the target FER. If no further feasible increments can be made, the increment value is reduced and the rates are backed off by an appropriate amount before continuing.
Bibliography


Gallager codes,” IEEE Transactions on Communications, vol. 47, no. 10, pp. 1449

irregular low-density parity-check codes,” IEEE Transactions on Information Theory,

codes,” IEEE Transactions on Communications, vol. 49, no. 10, pp. 1727 – 1737,

for modulation and detection,” IEEE Transactions on Communications, vol. 52,

[21] A. Ashikhmin, G. Kramer, and S. ten Brink, “Extrinsic information transfer func-
tions: Model and erasure channel property,” IEEE Transactions on Information Theory,


channel model with joint correlation of both link ends,” IEEE Transactions on

correlation, double scattering, and keyhole,” IEEE Transactions on Information Theory,


