Università di Padova  Facoltà di Ingegneria

Dipartimento di Ingegneria dell'Informazione

Scuola di Dottorato di Ricerca in Ingegneria dell'Informazione
Indirizzo: Scienza e Tecnologia dell'Informazione

Ciclo XXII

Tesi di Dottorato di Ricerca

Advanced Coding and Modulation for Digital Terrestrial Broadcasting and Cellular Systems

Direttore della Scuola: Ch.mo Prof. *Matteo Bertocco*

Supervisore: Ch.mo Prof. *Lorenzo Vangelista*

Dottorando: *Alberto Vigato*

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

While theoretically and technically television may be feasible, commercially and financially it is an impossibility, a development of which we need waste little time dreaming.

- Lee De Forest, 1926 -

Television won't last because people will soon get tired of staring at a plywood box every night.

- Darryl Zanuck, 1946 -

Television won't last. It's a flash in the pan.

- Mary Somerville, 1948 -

Contents

Acknowledgments **v**

Abstract **vii**

Sommario **ix**

List of Acronyms and Abbreviations **xi**

1 Introduction **1**
1.1 Thesis Outline 3

2 DVB–T2 Overview **5**
2.1 Background and Commercial Requirements 6
2.2 DVB–T2 System Overview 7
2.3 Error Protection Coding 9
2.4 Scheduling 10
2.5 Modulation Techniques 12
2.5.1 Rotated Constellations 13
2.5.2 PAPR Reduction 13
2.6 Synchronization and Channel Estimation 14
2.6.1 P1 Symbol 15
2.6.2 P2 Symbols 15
2.6.3 Scattered and Continual Pilots 16
2.6.4 Pilot Reference Sequence 16
2.7 Multiple–Antenna Techniques 17
2.8 Performance and Mode Comparison 18

3 Multiantenna Systems Overview **19**
3.1 ST Block Codes 20
3.1.1 STBC with Orthogonal Design 22
3.1.2 STBC with Cyclic Division Algebra Design 23
3.1.3 Rotation–based STBC 27

3.2 Delay Diversity 27

4 Hard and Soft Detection of STBC 29
4.1 Sphere Decoder 29
4.1.1 The Complex Sphere Decoder 33
4.2 Soft Decoding 37
4.2.1 Iterative MLM Complex Sphere Decoder 38
4.2.2 Group–wise MLM Complex Sphere Decoder 40
4.3 Performance Comparison 42
4.3.1 Methods Comparison 42
4.3.2 Performance on STBCs 44
4.4 Pseudo–Codes 48
4.4.1 Sphere Decoder 48
4.4.2 Complex SD 50
4.4.3 I–CSD 55
4.4.4 G–CSD 56

5 Modulation Diversity Schemes 61
5.1 Mapping Methods 62
5.1.1 Re–mapped Repetition 63
5.1.2 Rotational Multi–Carrier Modulation 65
5.1.3 Multidimensional Rotated QAM 67
5.2 Simulation Results 67
5.2.1 Uncoded Scenario 68
5.2.2 Coded Scenario 69
5.3 MR–QAM Matrices 72

6 Joint Performance of Interleaving and Mapping 73
6.1 DVB–T2 Simplified System Model 74
6.2 Interleavers and Demux from DVB 76
6.2.1 AWGN channel 76
6.2.2 Portable channel 77
6.3 Nonstandard Methods 79
6.3.1 Iterative Demapping 80
6.3.2 Re–mapped Repetition 81

7 Coded Decision Direct Demodulation 85
7.1 DVB–T2 System Model 86
7.1.1 Link Layer 86
7.1.2 Error Protection 87
7.1.3 Frame Builder 87

7.1.4 OFDM Modulation 88
7.2 The CD3 Principles 88
7.2.1 Channel Estimation 89
7.2.2 Reference Symbol in DVB–T2 92
7.3 MISO Techniques to Improve Portable Reception 93
7.3.1 Channel Estimate for MISO Transmissions with CD3 . 94
7.4 Numerical Results 94
7.4.1 Single Transmit Antenna 95
7.4.2 Multiple Transmit Antennas 98
7.4.3 Computational Complexity and Conclusions 98

8 Multiple Users Discovering 101
8.1 Overview 102
8.1.1 Wireless Environment and Typical Topology 104
8.2 Communication Model 105
8.2.1 System Level 105
8.2.2 Link Level 105
8.3 Receiver Algorithms: Coherent Case 106
8.3.1 Successive Interference Cancellation 107
8.3.2 Optimal Decoding Rule 107
8.3.3 Optimal Joint Decoding on the Trellis Product 108
8.3.4 Joint Iterative Decoding 110
8.3.5 JID using Convolutional Codes 112
8.3.6 Preliminary Comparison: ST vs JID vs SIC 114
8.3.7 Detailed Comparison: JID vs SIC Performance 115
8.4 Dense Scenario: Coherent Case 120
8.4.1 General Performance of JID 121
8.4.2 ST vs Conventional Single User Detection at Comparable Trellis Complexity 122
8.5 Decoding Algorithms: Non–Coherent Case 123
8.5.1 Optimal Decoding Rule 123
8.5.2 Hierarchical Coding 123
8.5.3 Flat Fading Channel Model 124
8.5.4 Improved Implementation: Different Interleavers 129
8.5.5 Frequency Offset Channel Model 130
8.6 Final Considerations 133

9 Conclusions 135

Bibliography 139

Acknowledgments

Padova, gennaio 2010

Sono trascorsi ormai tre anni da quando ho iniziato la scuola di dottorato presso il dipartimento di ingegneria dell'informazione all'università di Padova, tre anni vissuti tra esperienze ricche di sfide, impegno, curiosità, difficoltà ma soprattutto molte soddisfazioni.

Ringrazio per la fortuna di aver sempre potuto contare su una famiglia solida, aperta al confronto e benevola nell'assecondare le mie aspirazioni, presente e pronta nel sostenermi e confortarmi quando ho dovuto affrontare i momenti più ardui e ostili. Grazie papà *Fulvio*, mamma *Carla* e *Alessandro*.

Ai miei supervisori: rinnovo il mio ringraziamento al prof. *Gianfranco Cariolaro* che fin da quando ero studente magistrale ha creduto in me e nelle mie capacità, grazie per il forte incentivo a intraprendere la strada del dottorato, per gli utili suggerimenti impartiti in questo cammino e per avermi offerto l'opportunità di Qualcomm. Un sentito ringraziamento va al prof. *Lorenzo Vangelista*, grande figura di riferimento di profonda umanità e di gran cuore nella mia crescita professionale, grazie per l'enorme disponibilità dimostrata in questi anni, per ogni prezioso aiuto, ogni consiglio e la profonda fiducia sovente accompagnata da parole di incoraggiamento.

Grazie al prof. *Nevio Benvenuto* e al prof. *Stefano Tomasin*, la cui brillante esperienza accademica e la costante disponibilità hanno illuminato tutto il percorso di ricerca condotta in dipartimento sul DVB–T2 , offrendo sempre spunti interessanti che impreziosivano ogni argomento sviluppato durante il mio lavoro. Grazie alla dott.sa *Vittoria Mignone*, al dr *Alberto Morello* e allo staff dei laboratori di ricerca RAI di Torino, per avere aperto questa interessante e proficua collaborazione con il nostro dipartimento sul progetto DVB. Grazie inoltre ai docenti dei corsi didattici seguiti durante il dottorato, in particolare al prof. *Paolo Ciatti* e un ricordo al prof. *Lorenzo Brunetta*.

Per l'esperienza di Qualcomm oltre al professor Cariolaro debbo la mia gratitudine al dr *Roberto Padovani* che in prima persona ha accettato la mia candidatura alla scholarship, al dr *Thomas Richardson* al dr *Junyi Li* per avermi accolto a far ricerca nel loro staff, presso la sede Flarion, al dr *Cyril Méasson* che mi ha condotto e consigliato con costanza e pazienza durante il mio lavoro di ricerca in trasferta, al dr *Xinzhou Wu* e al dr *Saurabh Tavildar* per gli utili suggerimenti impartiti. Inoltre vorrei ringraziare gli amici conosciuti in Flarion: *Bilal Sadiq, Jaspreet Singh, Sundar Subramaninan, Mathieu Leconte* e *Mukul Agarwal* con cui ho condiviso tra l'altro molti piacevoli momenti extralavorativi.

Ringrazio la signora *Diane Litchko* delle risorse umane, per la solerte disponibiltà nell'aiutarmi ad ambientarmi nel New Jersey e a superare le difficoltà iniziali. Grazie inoltre al reverendo *David Noble*, padre *Arduino Puleo*, la famiglia *Cerati*, la famiglia *Reina* e tutte le persone incontrate negli States per aver reso indimenticabile questa esperienza extracontinentale.

A tutti gli amici che in questi anni mi sono stati vicini, con cui ho trascorso momenti felici e momenti tristi, condiviso gioie e fatiche, nel lavoro e fuori: voglio ringraziare *Ermanna Conte, Giovanni Vadalà, Francesco Renna, Marco Rotoloni, Emiliano Dall'Anese, Marco Maso, Davide Zennaro, Paolo Leoni, Nicola Wrachien, Federico Longhin, Ruggero Poletto, Marco Menegozzo, Martina Urbani, Chiara Panzetta, Annarita Felisatti* e molti altri.

A te *Sara*, che negli ultimi mesi sei stata la figura più presente nella mia vita, con me hai visto nascere e svilupparsi questo progetto di tesi, ma soprattuto con te ho condiviso sensazioni forti, sorrisi, lacrime, speranze, paure, progetti e sogni. Su di te ho potuto contare e confido di poter contare in futuro.

Alberto Vigato

Abstract

A new platform for HDTV transmissions, in the form of a new generation terrestrial digital video broadcasting standard (DVB–T2), has been developed by the DVB Project. A number of technical innovations have been included in DVB–T2 to boost throughput and ruggedness, enhance single–frequency network coverage, and ease both transmitter and receiver implementation. Several strategies have been investigated during the standard definition. This thesis present a research branch of this investigation starting from a multi–antenna overview: a new soft detector for space–time block codes (STBCs) is proposed, which is claim to be more efficient than the state of art techniques. Commercial and technical reasons have suggested the adoption of orthogonal frequency division multiplexing (OFDM). As multiple–antennas techniques are not backward compatible with DVB–T constellations, new modulation methods and iterative demapping schemes, suitable for OFDM in the presence of Rayleigh fading channels, are assessed in terms of diversity gain without spectral or power inefficiencies. A very important role is played by LDPC codes which are one of the principal innovation of the new digital terrestrial standard. We show how all these scenarios change once an LDPC encoding is applied to the systems. We also evaluate the performance of the coded decision directed demodulation (CD3) technique for channel estimation in DVB–T2. Since the insertion of pilot tones in the OFDM signal affects the spectral efficiency, CD3 is able to provide an accurate channel estimation with very few pilots, even in the presence of highly frequency dispersive and time–variant channels, increasing the spectrum efficiency. We give a comparison in terms of achievable throughput between DVB–T2 with CD3, even with multiple antennas, and current DVB–T.

A second part is related to a cellular network scenario. Assume a set of nodes drawn uniformly at random on a finite surface having a time slot where each node broadcasts its identifier. This problem becomes of practical and theoretical interest in a dense scenario where the number of available physical resources is much smaller than the total number of nodes in the system. Here we address link level strategies for single–hop peer discovery

assuming that peer identifiers are broadcasted via coded transmission and we are interested on simultaneous detection of peers which share the same physical resource then cause reciprocal interference. Several strategies are investigated through system simulations: it is shown that, in general, joint iterative decoding achieves impressive gains compared with interference cancellation even preserving almost the same complexity. Elementary modifications of broadcasted signals such as symbol permutations permit to further improve the system performance. Finally, in order to extend this work, we introduce some channel estimation procedures for primary channel models which suit with multiple detection goal.

Sommario

Nel progetto DVB è stato recentemente sviluppato uno standard di nuova generazione per la trasmissione terrestre del segnale video, conosciuto come DVB–T2. Sostanzialmente si tratta di una nuova piattaforma per la trasmissione della televisione digitale ad alta defizione. Il DVB–T2 include una serie di novità tecniche per migliorarne il throughput e la robustezza, estendere la copertura delle reti a singola frequenza, e semplificare la realizzazione del trasmettitore e del ricevitore. Per questa ragione, molteplici strategie sono state studiate nella fase di definizione e stesura dei protocolli del nuovo standard. In questa tesi è presentato un lavoro di ricerca inserito in questo contesto, partendo da una prima indagine sulle trasmissioni multiantenna: a tal fine è stato proposto un decodificatore 'soft' per i codici a blocco che operano nello spazio–tempo (codici STBC). Viene illustrato come sia maggiormente efficiente rispetto alle procedure presentate fino ad oggi. Le strategie commerciali e tecnologiche hanno favorito l'utilizzo della modulazione OFDM. Considerando la non retrocompatibilità dei sistemi ad antenna multipla con le attuali modulazioni del sistema corrente DVB–T, nuovi schemi di modulazione e demappatura iterativa, specifici per gli schemi OFDM in presenza di canali Rayleigh fading, sono stati valutati in termini di guadagno di diversità senza inefficienze spettrali o di potenza. Un ruolo fondamentale è attribuito ai codici LDPC che sono tra le principali innovazioni del nuovo standard per il digitale terrestre. Mostreremo come cambiano le performance di tutti gli scenari analizzati quando ad essi viene accostata una codifica LDPC. Inoltre daremo una valutazione del comportamento della tecnica CD3 per la stima di canale, anch'essa adottata nel DVB–T2. Poiché l'inserimento di toni pilota nel segnale OFDM indebolisce l'efficienza spettrale, il CD3 è in grado di garantire una stima di canale molto accura facendo ricorso ad un numero molto esiguo di toni pilota favorendo una maggiore efficienza dello spettro. Anche in presenza di canali tempo–varianti e altamente dispersivi in frequenza, il CD3 offre prestazioni ragguardevoli. Presenteremo un confronto tra DVB–T2 con stima di canale CD3 in singola e multipla antenna e l'attuale DVB–T in termini di throughput raggiungibile.

La seconda parte riguarda uno scenario relativo alle reti cellulari. Consideriamo un insieme di nodi sparsi uniformemente a caso su di un area geografica e ipotiziamo che debbano trasmettere il loro indirizzo identificativo. Il problema assume interesse sia teorico che pratico in uno scenario ad alta densità dove il numero di risorse fisiche è di molto inferiore al numero totale di dispositivi presenti nel sistema. In questa sede vengono proposte strategie a livello di collegamento per il riconoscimento diretto dei dispositivi assumendo che l'indirizzo identificativo sia trasmesso protetto da codifica e lo scopo sia di rilevare simultaneamente tutti i dispositivi che condividono la stessa risorsa fisica causando reciproca interferenza. Molteplici strategie sono state analizzate mediante simulazioni del sistema: si darà prova che, in generale, una decodifica iterativa congiunta comporta un guadagno rilevante confrontanto con la tecnica classica della cancellazione di interferenza, pur mantenedo una complessità computazionale pressoché simile. Ulteriori semplici modifiche sui segnali trasmessi come la permutazione dei simboli permettono un ulteriore miglioramento delle prestazioni di sistema. Come ultimo passo, per dare un estensione pratica al lavoro, vengono introdotte alcune procedure di stima di canale per modelli di canale elementari che si adattino alla filosofia del rilevamento multiplo.

List of Acronyms and Abbreviations

3GPP: 3rd Generation Partnership Project

ACE: Active Constellation Extension
AF+CD: Annihilating Filter and Cadzow iterative Denoising
ASO: Analogue Switch–Off
ATSC: Advanced Television Systems Committee
AWGN: Additive White Gaussian Noise

BB: Base Band
BBC: British Broadcasting Corporation
BCH: Bose Chaudhuri Hocquenghem code
BER: Bit/Block Error Ratio
BICM: Bit–Interleaved Coded Modulation
BICM–ID: Bit–Interleaved Coded Modulation Iterative Demapping
BPSK: Binary Phase–Shift Keying
BSA: Binary Switching Algorithm

CC: Convolutional Code
CCM: Constant Coding and Modulation
CDMA: Code Division Multiple Access
CD3: Coded Decision Directed Demodulation
CDD: Cyclic Delay Diversity
CENELEC: Comité Européen de Normalisation ELECtrotechnique
CMUX: Complex MUltiplication
C/N: Carrier to Noise ratio
COFDM: Coded Orthogonal Frequency Division Multiplexing
CP: Continual Pilots
CP: Cyclic Prefix
CRC: Cyclic Redundancy Check
CSD: Complex Sphere Decoder

CT: Computational Time

DAB: Digital Audio Broadcasting
DAC: Digital to Analog Converter
DD: Delay Diversity
DFT: Discrete Fourier Transform
DPSK: Differential Phase–Shift Keying
DRM: Digital Radio Mondiale
DVB: Digital Video Broadcasting
DVB–H: Digital Video Broadcasting Handheld
DVB–S: Digital Video Broadcasting Satellite
DVB–S2: Digital Video Broadcasting Satellite of 2nd generation
DVB–SH: Digital Video Broadcasting Satellite services to Handhelds
DVB–T: Digital Video Broadcasting Terrestrial
DVB–T2: Digital Video Broadcasting Terrestrial of 2nd generation

EBU: European Broadcasting Union
EIT: Event Information Table
ETSI: European Telecommunication Standard Institute
EXIT: EXtrinsic Information Transfer

FEC: Forward Error Correction
FEF: Future Extension Frame
FFT: Fast Fourier Transform

G–CSD: Group–wise (MLM) Complex Sphere Decoder
GI: Guard Interval
GS: Generic Stream
GSE: Generic Stream Encapsulation

HDTV: High Definition TeleVision

IC: Interference Cancellation
ICE: Ideal Channel Estimation
ICI: Inter–Channel Interference
I–CSD: Iterative (MLM) Complex Sphere Decoder
ID: Iterative Demapping
IDFT: Inverse Discrete Fourier Transform
IEEE: Institute of Electrical and Electronics Engineers
IFFT: Inverse Fast Fourier Transform
ISDB: Integrated Services Digital Broadcasting
ISDB–T: Integrated Services Digital Broadcasting Terrestrial

ISI: Inter–Symbol Interference

JID: Joint Iterative Decoder
JTC: Joint Technical Committee

LD: Linear Dispersion
LLR: Log–Likelihood Ratio
LTE: Long Term Evolution

MAC: Multiple–Access Channel
MAP: Maximum A Posteriori
MDDM: Multicarrier Delay–Diversity Modulation
MFN: Multi–Frequency Network
MIMO: Multiple In Multiple Out
MISO: Multiple In Single Out
ML: Maximum Likelihood
MLM: Max–Log–MAP
MMSE : Minimum Mean Square Error
MPEG: Moving Picture Experts Group
MRC: Maximum Ratio Combining
MR–QAM: Multidimensional Rotated QAM

OFDM: Orthogonal Frequency Division Multiplexing

PAM: Pulse Amplitude Modulation
PAPR: Peak–to–Average Power Ratio
PER: Packet Error Rate
PLP: Physical Layer Pipe
PN: Pseudo Noise
PP: Pilot Pattern configuration
PSK: Phase–Shift Keying

QAM: Quadrature Amplitude Modulation
QEF: Quasi Error Free
QFT: Qualcomm Flarion Technologies
QPSK: Quadrature Phase–Shift Keying

RAI: RAdiotelevisione Italiana
RF: Radio Frequency
R–MCM: Rotational Multi–Carrier Modulation
RR: Re–mapped Repetition
RS: Reed–Solomon code

SDTV: Standard Definition TeleVision
SER: Symbol Error Ratio
SFN: Single Frequency Network
SIC: Successive Interference Cancellation
SINR: Signal to Interference–plus–Noise Ratio
SISO: Single In Single Out
SD: Spatial Diversity
SD: Sphere Decoder
SM: Spatial Multiplexing
SNR: Signal–to–Noise Ratio
SOVA: Soft Output Viterbi Algorithm
SP: Scattered Pilots
ST: Super Trellis decoder
STBC: Space–Time Block Code
STTC: Space–Time Trellis Code
SVD: Singular Value Decomposition

TD–CDMA: Time Division Code Division Multiple Access
TFS: Time–Frequency Slicing
TS: Transport Stream
TU6: Typical Urban channel model with 6 taps

UHF: Ultra High Frequency

V–BLAST: Vertical Bell LAboratory Space Time
VHF: Very High Frequency

WCDMA: Wideband Code Division Multiple Access

Chapter 1

Introduction

Overall the world the broadcasting television systems are switching from analog to digital transmissions. This process is gradual and it has started some decades ago, with the birth of organizations, created and developed to this specific purpose. The most important are *Digital Video Broadcasting* (DVB) for Europe, *Advanced Television Systems Committee* (ATSC) for North America, and *Integrated Services Digital Broadcasting* (ISDB) for Japan and Brazil.

Our interests are clearly focused on DVB Project.[1] It is an industry–led consortium of over 280 broadcasters, manufacturers, network operators, software developers, regulatory bodies and others in over 35 countries committed to designing open interoperable standards for the global delivery of digital media services. They are published by a Joint Technical Committee (JTC) of European Telecommunications Standards Institute (ETSI)[2], European Committee for Electrotechnical Standardization – Comité Européen de Normalisation Electrotechnique (CENELEC)[3] and European Broadcasting Union (EBU).[4] Many aspects of DVB are patented, including elements of the MPEG video coding and audio coding. Services using DVB standards are available on every continent with more than 220 million DVB receivers deployed.

Among the set of different strategies, we are interested to the digital terrestrial broadcasting since my PhD principal topic has been working on technologies developing for the second generation of digital video broadcasting–terrestrial (DVB–T2). The first generation digital video broadcasting–terrestrial (DVB–T) standard has been defined in 1996 and published by the

[1] `http://www.dvb.org/`
[2] `http://www.etsi.org/WebSite/homepage.aspx`
[3] `http://www.cenelec.eu/Cenelec/Homepage.htm`
[4] `http://www.ebu.ch/`

European telecommunication standard institute (ETSI) in the 1997, to provide access to digital terrestrial television broadcasting through both roof–top and portable antennas. After about ten years, in 2006, DVB has launched an initiative (DVB–T2) to define a new standard for terrestrial reception of digital TV, whose main aim, as stated in the call for technologies [1], is to provide high definition TV (HDTV) to fixed receivers and standard definition TV (SDTV) to portable receivers. Possible extensions to mobile scenarios have been also envisioned. Many research groups answered to the *Call for Technologies* then thy have been involved in the development of DVB–T2 project during the last three years. In particular I belong to a group in my department which has started a collaboration with *RAI Centro Ricerche e Innovazione Tecnologica*[5] in Turin, Piedmont, Italy. Other important groups related to this project are in *BBC Research & Development*[6] in Kingswood, Surrey, UK, and *Philips*[7] in Amsterdam, North Holland, Netherlands.

DVB–T2 standard aims at providing high definition television and high rate services for the same spectrum allocation of current DVB–T standard. This goal is achieved by adopting new physical layer features that increase the spectrum efficiency. In this respect, the insertion of pilot tones in the orthogonal frequency division multiplexed (OFDM) signal, though allowing an efficient channel estimation, decreases the spectral efficiency. Several different technologies and strategies has been considered in the developing and description of this standard in order to achieve the required features. Most of them are not completely introduced in the final decision of the standard published by Ofcom on April 3, 2008. Any discarded solutions were related to multiple–in multiple–out (MIMO) antennas system, any different modulation schemes, new combining of mapping and interleaving patterns and so on.

In the second part of my PhD, after the DVB–T2 standard definition, I completed my PhD period with an abroad experience, spending nine month of internship in Qualcomm Flarion Technologies as a winner of *"Cariolaro Scholarship @ Qualcomm 2008"*. This branch of *Qualcomm*[8] deals with new cellular and wireless system technology definition working on simulations and prototypes. In specific, in order to give more continuity during my experience, I worked on physical layer topics such as channel estimation, encoding/decoding, joint channel estimation and detection using a factor graph and message–passing[9] approach, then joint decoding of multiple user in inter-

[5] `http://www.crit.rai.it/`
[6] `http://www.bbc.co.uk/rd/index.shtml`
[7] `http://www.research.philips.com/`
[8] `http://www.qualcomm.com/`
[9] Also known as *belief propagation.*

fering scenario. The most important results obtained during the internship are reported in this thesis.

1.1 Thesis Outline

The main part of this work concerns the research results carried out by our group during the DVB–T2 standard design divided into six chapters.

Chapter 2 presents a general overview of the principal features and key technologies, such as error protection, scheduling, modulations, synchronization, channel estimation, and multiple antenna, which are included in the next–generation terrestrial digital television standard DVB–T2, starting from the commercial requirements of the new standard and also pointing out a significant performance comparison between it and its predecessor DVB–T. Chapter 3 is a brief overview of multiantenna systems in terms of space–time block codes (STBC): following different design principles, we present some STBCs we have consider as possible candidates for a potential MIMO terrestrial broadcasting scenario. Chapter 4 meets the problem of soft detection for multiantenna systems which computational complexity is considerable. Soft detection is fundamental whenever a system is protected by a powerful channel coding which requires soft information to be decoded, such as turbo codes and low–density parity check (LDPC) block codes. A first analysis to the sphere decoder algorithm, considered the fundamental method for multidimensional hard detection in reasonable complexity, we derive an optimal end efficient algorithm for soft detection in MIMO systems. We also compare the STBCs performance even combined with LDPC. Chapter 5 compares three different modulation diversity schemes in terms of complexity, uncoded performance, and LDPC coded scenarios. Chapter 6 analyzes several interleaver, demultiplexer, and mapping patterns, used by other DVB standards, in order to find a good trade–off for DVB–T2 scenarios, such as AWGN and portable channel. Chapter 7 presents a good tutorial for the coded decision directed demodulation technique for DVB–T2 channel estimation with a performance analysis for single and multiple transmission antennas in different channel models and different constellation sizes in a LDPC/BCH and OFDM scenarios.

The second part of this thesis concerns the most important results obtained in QFT. They are dealt in Chapter 8: it provides a practical aproach to the problem of discovery of multiple users in high density cellular systems. Optimal and suboptimal decoding algorithms are derived in order to obtain significant solutions and results in different level of density for both coherent and incoherent detection. About the latter, any channel estimation ideas are

proposed.

Some final considerations are exposed in brief in Chapter 9.

Chapter 2

DVB–T2 Overview

Digital Video Broadcasting – Second Generation Terrestrial (DVB–T2) is a second–generation terrestrial transmission system for digital television broadcasting. The key motivation behind this standard was the desire in several European countries to offer high definition television (HDTV) services as efficiently and effectively as possible. It builds on the technologies used as part of the first–generation system, DVB–T, developed over a decade ago then it extends the range of most of the parameters of the previous standard and significantly reduces overhead to build a system with a throughput close to theoretical channel capacity, with the best possible ruggedness of transmission. A number of technical innovations have been included in DVB–T2 to boost throughput and ruggedness, enhance single–frequency network coverage, and ease both transmitter and receiver implementation. The move to HDTV inevitably brings with it a change of source coding, necessitating the introduction of new domestic reception equipment (set–top boxes and TV sets), and therefore offers an ideal opportunity to upgrade the transmission system simultaneously.

The system, which is currently standardized within ETSI, also provides many features that make it the most advanced terrestrial transmission technology, suitable for the coming decade and beyond, with some features (e.g., time–frequency slicing and time–interleaving) that may be of use for the next generation of the current version of the standard for handheld applications, DVB–H. Plans are already underway within the United Kingdom to launch a new terrestrial HDTV service using DVB–T2 from the end of 2009.

Starting from the motivations that led the DVB project to create the new standard, we surveys the key technologies behind DVB–T2, including the LDPC/BCH forward error correction scheme, transmission scheduling, orthogonal frequency–division multiplexing with huge block size, multiple–antenna transmissions, and synchronization techniques. In order to increase

the data rate of current DVB–T standard and allow HDTV transmission, the spectrum must be better exploited and the physical layer must be redesigned minimizing overheads, increasing constellation size and optimizing channel coding. Indeed, orthogonal frequency division multiplexing (OFDM), currently adopted for DVB–T [2] and nowadays easily implementable on chip, has proved to be an efficient modulation technique with a simple method for equalization of the broadband dispersive channel encountered by roof–top and portable antennas, as well as in mobile and handheld scenarios. Suitable configurations of OFDM allow also the deployment of single frequency networks (SFNs) [2], where all base stations transmit nation–wide on the same carrier frequency at the same time, under reasonable synchronization assumptions, allowing both a diversity reception gain and an easier network planning. Therefore, OFDM is the preferred modulation technique for DVB–T2. Other technologies used in previous DVB standards have been also considered suited for DVB–T2, e.g. the low density parity check (LDPC) codes, adopted in the recent DVB standard for satellite transmissions (DVB–S2) [3] and included in DVB–T2. At the same time, technologies not adopted in previous DVB standards are considered for DVB–T2 since they allow for a significant rate/performance improvement with a limited complexity. In this regard, the use of multiple antennas at the transmitter, has the potential of increasing the diversity gain and can be implemented in a distributed manner by SFNs: simple solutions as Alamouti [4] space–time codes, which require only signal combining at the receiver, are also well suited for low–cost user equipment.

In this chapter, after giving a first description of the key technologies, we present some first simulation results showing that all commercial requirements have been met and indeed, in some cases, substantially exceeded. A comparison with the current DVB–T standard is also provided, showing that DVB–T2 is able to increase the payload throughput and allows HDTV transmission with current network planning.

2.1 Background and Commercial Requirements

DVB–T is the most popular way to distribute digital TV in Europe and many other parts in the world. However, its success has been under pressure after the introduction of the MPEG–4 compression standard and the new digital video broadcasting standard for satellite transmissions, DVB–Satellite – Second Generation (DVB–S2) [5], which are driving the first adoption of HDTV on satellite and cable networks. Therefore, in 2006 the terrestrial TV broadcasting community felt the need for an improved system. While

increasing the throughput and ruggedness were the initial broad aims in building a second–generation system, the work followed the usual pattern within DVB, involving all parts of the broadcast value chain: broadcasters, network operators, equipment suppliers, chip providers, and receiver manufacturers. First – finishing in record time by March 2007 – a detailed set of commercial requirements was written for the system. In particular, DVB–T2 should provide:

- Reuse of both existing domestic receiving–antenna installations and existing transmitter infrastructures.[1]
- A minimum of 30 percent throughput increase over DVB–T, working within the same planning constraints and conditions as DVB–T.
- Improved single–frequency–network (SFN) performance compared with DVB–T.
- Service–specific robustness, for example, to offer, within a single 8 MHz channel, some services for rooftop reception and other services for portable reception.
- Flexibility in bandwidth and frequency allocation within the ultra high frequency (UHF) and very high frequency (VHF) bands, but with additional features/modes to support higher frequency bands.
- A mechanism to reduce the peak–to–average–power ratio of the transmitted signal in order to reduce transmission costs.

2.2 DVB–T2 System Overview

The DVB–T2 system specification, recently approved by the DVB project and sent to the European Telecommunications Standards Institute (ETSI) for standardization, includes the following key features:

- The same basic modulation technique as DVB–T: orthogonal frequency–division multiplexing (OFDM) with guard interval (GI), which provides a fundamentally resilient transmission system for the terrestrial channel.

[1] Despite the significant gains that multiple–input multiple–output (MIMO) techniques could offer in terrestrial transmissions, they would have forced replacement of existing rooftop antennas. Accordingly, they were not used.

- Extended range of fast Fourier transform (FFT) sizes of OFDM, to increase the single–frequency–network (SFN) performance and, together with an increased range of GIs, to provide significantly improved bandwidth efficiency.

- The same baseband framing and forward error correction (FEC) mechanisms included in DVB–S2 [5], adding the 256–quadrature amplitude modulation (QAM) constellation, to take full advantage of the efficiency of the error–correction technique, and introducing a concept called rotated constellation, which can significantly improve the system performance in frequency selective terrestrial channels.

- A method of transporting individual data services in separate logical channels, known as physical layer pipes (PLPs), within the physical layer, where the error–correction coding and interleaving are applied separately to each PLP. This allows for the implementation of service–specific robustness.

- A time interleaver of at least 70 ms for high–data–rate services, to provide increased immunity to impulsive interference.

- A very flexible frame structure, where the data can be either spread evenly across a whole frame for maximum time diversity or concentrated into bursts to allow power–saving techniques to be used in the receiver; the frame structure includes an efficient physical–layer signaling mechanism, called layer 1 (L1) signaling, which signals all of the parameters of the transmission system to the receiver.

- Extended range of reference signals – scattered pilots (SPs) and continual pilots (CPs) – to allow an optimum choice to be selected for any given channel.

- An optional mechanism for transmit diversity, based on the Alamouti scheme, to improve reception in areas where the coverage from two transmitters overlaps.

- Two separate mechanisms for reducing the peak–to–average–power ratio (PAPR) of the transmitted signal.

- Signaling is included to allow future backward–compatible standards to use parts of the T2 frame structure: time–frequency slicing (TFS) to allow several radio frequency (RF) channels to be used together for increased capacity and frequency diversity; and optional parts of

the frame whose content will be defined in the future, as the *future extension frame* (FEF) part.

Fig. 2.1 shows the block diagram of a DVB–T2 transmitter. A DVB–T2 transmitter has the capability of handling multiple PLPs, to provide multiple services, while the DVB–T2 receiver is only required to decode a single data PLP together with its associated common PLP (if any).

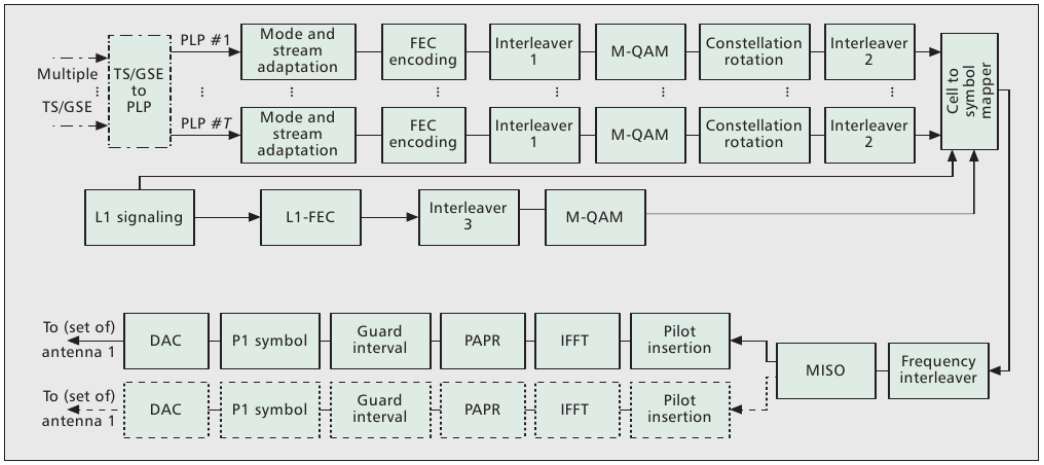

Figure 2.1: DVB–T2 block diagram.

DVB–T2 allows each PLP to carry its own independent service transport stream (TS) or generic stream encapsulation (GSE) stream. However, the DVB–T2 standard also defines a possible method to avoid transmitting the same information many times when handling multiple TSs (see Annex D of [2]). If multiple TSs share common packets (e.g., the event information table [EIT]), these can be removed from the TSs and mapped into the common PLP. The receiver will then merge the content of the common PLP and the user–selected data PLP to reconstruct the valid TS.

The *TS/GSE to PLP* block of Fig. 2.1 realizes the splitting and merging function, which ensures synchronization among the user–selected data PLPs and the common PLP, and provides full end–to–end TS transparency with improved bandwidth efficiency. The next *mode and stream adaptation* block maps the input stream into DVB–T2 blocks, with the compression of MPEG–2 null–packets and the insertion of cyclic redundancy check (CRC) bits.

2.3 Error Protection Coding

Following the philosophy of the DVB standard family, forward error correction of DVB–T2 includes the BCH and a subset of the low density parity

check (LDPC) codes of DVB–S2, with new bit interleavers. The target performance for the DVB systems (quasi error free [QEF]) is defined to be "less than 1 uncorrected error event per program per transmission hour," which for a 5 Mb/s service means having a bit error rate (BER) on the order of 10^{-10}. The LDPC code alone cannot always guarantee this target performance, so a BCH code has been cascaded to the LDPC to avoid undetected errors at low BER while still having a high code rate.

In order to reduce the encoding complexity, the LDPC parity check matrices are of the form $\mathbf{H} = [\mathbf{AB}]$, where $\mathbf{A}$, which is applied to the information part, is a sparse cyclic matrix and $\mathbf{B}$, which is applied to the parity part, has a staircase shape, with only its main and first lower diagonal containing ones. The cyclic nature of $\mathbf{A}$ reduces the storage requirements.

Two block lengths are available: 64,800 bits or 16,200 bits. The performance of the short codes is some tenths of dB worse than normal codes, but allows for low–bit–rate applications with shorter latency. LDPC code rates available in DVB–T2 are a selection of the code rates of the DVB–S2 code:[2] 1/2, 3/5, 2/3, 3/4, 4/5, and 5/6 are used for PLP protection; 1/4, for short code length only, is used in L1 signaling protection. The LDPC codes in DVB–T2 are irregular and the error protection level of each code bit is not uniform, but depends on the column weight of the parity check matrix. Hence, bit interleaved coded modulation (BICM) has been used to map the coded bits onto constellation symbols, by a cascade of an interleaver and a demultiplexer between the coder and the mapper, as shown in Fig. 2.1.

2.4 Scheduling

In order to offer service–specific robustness and optimize time–interleaving memory requirements, the DVB–T2 system can be described as a set of fully transparent PLPs, each one performing independent mode adaptation, FEC encoding, bitmapping onto constellation points (cells), and time interleaving. The scheduler/frame builder is a functional element that maps the data cells at the output of the time interleavers into OFDM symbols, also adding signaling information in order to construct DVB–T2 frames and superframes (Fig. 2.2).

Fig. 2.3 shows a simplified example of how cells coming from different PLPs, each identified by a different color, may be read from the time interleaving (TI) memories and mapped into OFDM symbols (vertical blocks).[3]

[2] Nominal code rate. The effective LDPC code rate is in some cases of short codes slightly lower.

[3] This is shown before the frequency interleaving is applied.

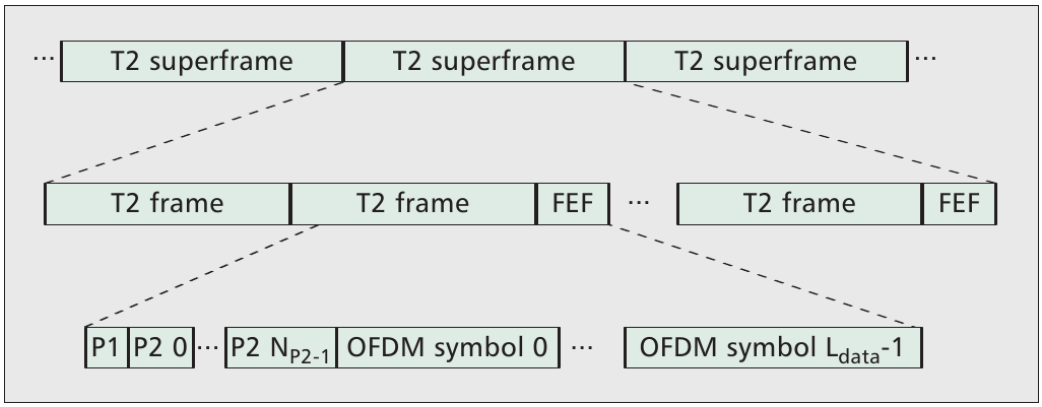

Figure 2.2: DVB–T2 frame structure.

The cell mapping strategy in time and frequency may be selected in a very flexible way. One possible target may be to achieve the maximum time diversity, thus spreading the cells from a PLP over all the OFDM symbols onto a frame, or even over multiple frames: to achieve this, the TI memory of a given PLP is split into several sub–slices, which are mapped in OFDM symbols alternated with the sub–slices of the other PLPs; this implies that, to receive the selected service, the receiver must operate continuously for all the OFDM symbols in a frame. A second target may be to obtain the maximum power saving in the receiver (e.g., for battery–powered portable devices) by switching the receiver on for only a small percentage of time: this can be achieved by concentrating cells of a PLP in time over a limited number of adjacent OFDM symbols, without sub–slicing, as shown in Fig. 2.3. Note that the figure shows two DVB–T2 frames where the data rates of the PLPs are constant. If the data rates are changing, the size of the slices would change from frame to frame accordingly.

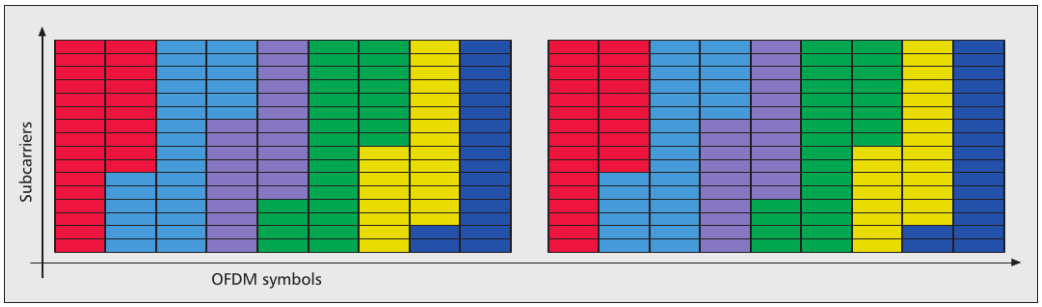

Figure 2.3: Different PLPs occupying different slices of individual modulation, code rate, and time interleaving. Two frames are shown.

The DVB–T2 frame structure is shown in Fig. 2.2. At the top level, the frame structure consists of superframes (maximum duration of 64 s when FEFs are not used), which are divided into DVB–T2 frames; and these are further divided into OFDM symbols. The number of DVB–T2 frames per superframe is such that every data PLP has an integer number of interleaving frames per superframe. In turn, the superframe may optionally carry FEF parts, which are time periods left unused by DVB–T2 signal, allowing for other future services as yet undefined. The frame starts with one reference symbol called P1 and one or more reference symbols called P2 (discussed in more detail in the section on synchronization and channel estimation), followed by a configurable number of data symbols. The frame duration is on the order of 100 to 250 ms. A data PLP does not have to be interleaved entirely within a single DVB–T2 frame, but may be spread across several frames.

The main purpose of the P2 symbols is to carry signaling data. As already said, the PLP throughput is time–variant; therefore, the position in time and frequency of the cells associated with a PLP changes frame by frame. Since the receiver must be able to extract at least the user selected data PLP and the common PLP (when present), it must be able to track the data cells' positions. The DVB–T2 system, even in case of static reception, may be affected by impulsive noise. Therefore, cell position signaling (the so called *dynamic* L1 information) has been given particular design care in DVB–T2, by including various transmission mechanisms based on error correction and detection, and repetition. L1 signaling is in fact transmitted in each frame within the P2 symbol, but the information relevant to the next frame may also be embedded in the PLP data. It is also possible to repeat this information by L1 signaling both the current and next frames.

2.5 Modulation Techniques

DVB–T2 uses coded OFDM (COFDM) [6], as used by the DVB–T, digital audio broadcasting (DAB), terrestrial integrated services digital broadcasting (ISDB–T), and digital radio mondiale (DRM) broadcast standards, and by other radio systems such as IEEE 802.11a/n and the Long Term Evolution (LTE) of 3GPP. A wider range of OFDM parameters is offered than for DVB–T, while coding is also changed (as discussed above). There are 1 K, 2 K, 4 K, 8 K, 16 K, and 32 K FFT sizes, and each subcarrier, in each symbol, is modulated using QAM constellations. A range of options is available for payload data: 4–, 16–, 64–, and 256–QAM. The combination of 256–QAM with the new LDPC error correction offers increased throughput with

performance roughly comparable with 64–QAM in DVB–T.

2.5.1 Rotated Constellations

The LDPC codes of DVB–T2 offer good performance in non–selective channels using a higher code rate than DVB–T – thus giving higher throughput. However, frequency selective channels need extra redundancy previously given by a lower–rate code.

DVB–T2 also includes *rotated constellation* as an optional feature to improve performance even for very frequency selective channels. Rotating the constellation by a suitable angle means that every constellation point maps onto a different point on each of the I and Q axes. So a 16–QAM constellation has 16 different values for both I and Q (Fig. 2.4). On its own, this would not change anything. However, we contrive that the I and Q values derived from the rotated constellation are separated by cyclically delaying the Q before time and frequency interleaving. The constellations actually transmitted after interleaving comprise unrelated I and Q values derived from different original rotated constellation (Fig. 2.4). When the I and Q values are reunited after deinterleaving at the receiver, they will have been affected differently by any frequency selective fading. Suppose, as an extreme example, that one has been lost altogether. The surviving axis still contains information about all of the possible points – it is less reliable, but no longer a complete erasure. Rotated constellation introduces further diversity as the same bit is mapped simultaneously in more subcarriers, thus achieving a higher degree of diversity. This would not be possible with denser constellations and lower code rate, as a single data bit would be mapped on a smaller number of subcarriers (due to the lower code rate). Simulations [7] show that rotated constellation provides up to 0.75 dB of advantage over conventional QAM on wireless channels at a very limited increase in implementation cost.

2.5.2 PAPR Reduction

OFDM has the disadvantage that the transmitted signal increasingly resembles Gaussian noise as the number of subcarriers increases, with the consequence that the peak–to–average power ratio is high. This places demands on the transmitter's power amplifier. DVB–T2 includes two optional features that can reduce PAPR.

Active constellation extension (ACE) modifies some of the transmitted constellations by selectively moving their outer points to positions having greater amplitude [8]. ACE reduces PAPR without throughput loss, but is not used together with rotated constellation.

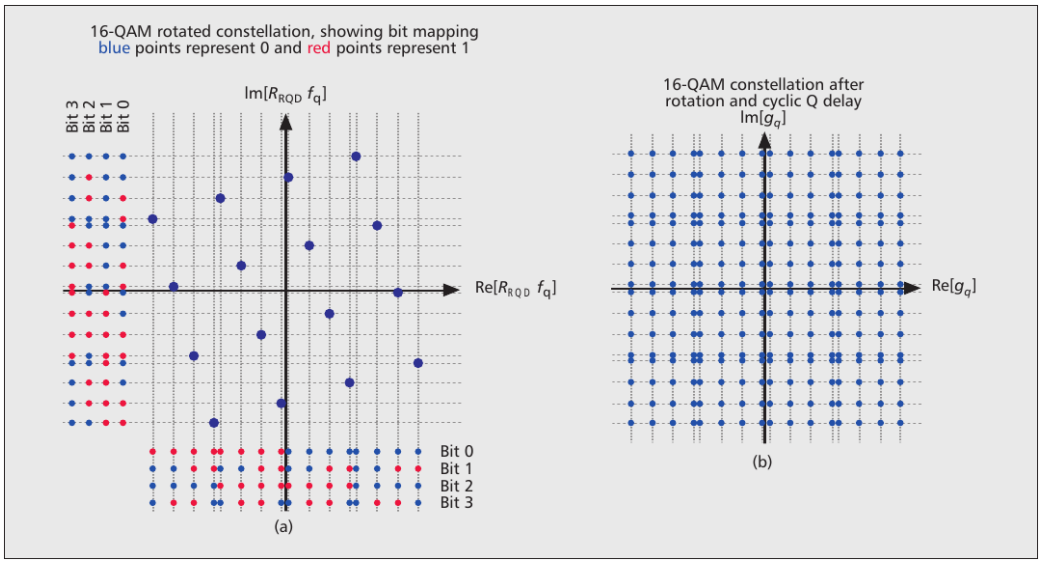

Figure 2.4: Rotated constellation with cyclic–Q delay: a) rotated 16–QAM constellation before cyclic Q delay; b) rotated 16–QAM constellation after cyclic Q delay, showing there are now $16^2 = 256$ possible states.

Reserved–carrier PAPR reduction sacrifices a small amount of throughput by reserving some subcarriers, which do not carry data [8]. They are used instead to carry arbitrary values, which permit the synthesis of a peak–canceling waveform.

2.6 Synchronization and Channel Estimation

The DVB–T2 standard includes particular design solutions to ease the time and frequency synchronization of the receiver. The most apparent one is the use of a frame made of a preamble and a payload, as schematically depicted in Fig. 2.2.

The preamble consists of a P1 symbol and a number of P2 symbols with the number depending on the chosen FFT size. For the 32 K and the 16 K sizes, there is only a single P2 symbol. For the 8 K, 4 K, 2 K, and 1 K sizes there are 2, 4, 8, and 16 P2 symbols, respectively. The payload follows the P2 symbols, although some of the data might already be carried within the P2 symbols, and consists of OFDM symbols of which subcarriers can be modulated by data or by known pilot values.

The use of the preamble significantly improves some synchronization steps, and it further allows for a much wider choice of transmitter parameters

without increasing the overall synchronization time.

2.6.1 P1 Symbol

The P1 symbol comprises an OFDM symbol with 1 K sub–carriers together with a special time–domain repetition structure as depicted in Fig. 2.5. The C part is a frequency shifted version of the first 542 samples of the OFDM symbol which forms the A part. The B part is the frequency shifted version of the last 482 samples of the A part. The frequency shift is equal to the subcarrier spacing of the OFDM symbol. Within the OFDM symbol, only 384 of the 1 K sub–carriers are differentially BPSK (DBPSK) modulated and used to transmit seven bits of information. The fixed P1 structure, together with the limited and highly protected signaling part, enables a fast scanning of the broadcast frequencies. The receiver can recognize the presence of a DVB–T2 transmission and store the key parameters (e.g., the FFT size or the presence of a FEF frame). The particular $C \rightarrow A \rightarrow B$ structure is designed to improve the robustness of the P1 detection in the presence of the most challenging channels such as a zero dB echo with opposite phase. The detection of the P1 symbol is also used to derive an initial time and frequency reference.

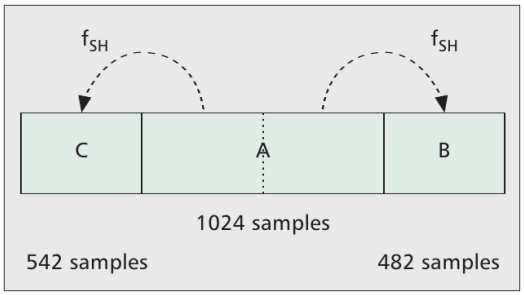

Figure 2.5: P1 symbol format.

2.6.2 P2 Symbols

The main role of the P2 symbol is carrying the L1 signaling, which can be quite large mainly because each PLP has its own transmission parameters. The L1 signaling is organized in an L1 presignaling part (where, e.g., the frame length is signaled) and an L1 post–signaling part. The protection of the bits of the L1 presignaling is based on a BCH code followed by a punctured LDPC code. The choice of the LDPC code might seem odd given the short length of the codeword. Nevertheless, it ensures no loss compared

to a convolutional code with the same rate and does not require a Viterbi decoder, which would only be used to decode the L1 signaling.

Another important role of the P2 symbol is to initiate the channel estimation process, and this is described further in the next section.

2.6.3 Scattered and Continual Pilots

A DVB–T2 receiver needs to estimate the channel experienced by the transmitted waveform to properly retrieve the transmitted information. To this end, the DVB–T2 standard defines conventional scattered pilot sequences that modulate a set of equally spaced subcarriers. The main novelty introduced by DVB–T2 is that it supports eight different SP patterns. The guiding principle in the design has been to match the pilot distance to the inverse of the GI length. While the SP are mainly designed to provide a reliable channel estimate, the continual pilots, which are matched to the FFT size, provide a means for fine frequency synchronization and common phase error correction.

The pilots in the P2 symbol are fixed and made to support the largest possible GI size, which is assumed to be acquired through conventional correlation–based methods. The pilot and data subcarriers' position in the P2 symbol are independent of other transmission parameters such as the bandwidth extension and the PAPR methods.

Some of the SP patterns require a channel estimate to be formed from several symbols, and the P2 pilots help to initiate this process. This is particularly the case with a very efficient mode mainly meant for fixed rooftop reception. The DVB–T2 standard defines an option where very few pilots are transmitted in the payload (PP8 in Table 2.1), and the channel estimation is based on the initial estimate offered by the P2 symbol followed by a data–aided channel estimation in which the decoded bits are fed back and used to refine the channel estimate [9]. This approach is only possible if an initial complete estimate is available, so without P2 it would not be feasible.

The different pilot distances also require adaptation of the pilot boosting factors (i.e., how much power is allocated to the pilots compared with the data). The DVB–T2 standard defines three boosting factors for SP and three for the CP.

2.6.4 Pilot Reference Sequence

The pilot values depend on the subcarrier index in the same way as for DVB–T. However, in DVB–T2, all the pilots (CP, SP, and P2) in each OFDM symbol are multiplied by plus or minus one according to a frame–level pseudo

	LDPC code rate	16–QAM	64–QAM	256–QAM
16 K	3/5	18.07	27.11	36.14
1/128 GI	2/3	20.11	30.17	40.21
PP7	3/4	22.62	33.93	45.24
32 K	3/5	17.05	25.63	34.23
1/16 GI	2/3	18.97	28.52	38.09
PP8	3/4	21.34	32.08	42.85
32 K	3/5	18.07	27.02	36.14
1/128 GI	2/3	20.11	30.06	40.21
PP7	3/4	22.62	33.82	45.24

Table 2.1: Achievable data rate (in Mb/s) for some DVB–T2 configurations.

noise (PN) sequence, and thus also depend on the OFDM symbol index. Although DVB–SH (digital video broadcasting – satellite services to handhelds) adopts freewheeling techniques for symbol counting, this signature on pilots provides an alternative and more robust frame synchronization approach that can indicate the current OFDM position within the frame if the preamble was lost, for instance, in the event of strong impulsive noise. Moreover, synchronization algorithms can exploit this frame level sequence to estimate and track the clock, symbol, frequency, and frame synchronization. This is achieved with no influence on the quality of the channel estimation.

2.7 Multiple–Antenna Techniques

The DVB–T standard allows simultaneous transmission on the same frequency of the same signal by multiple transmitters in order to implement a single–frequency network (SFN). By ensuring strict synchronization constraints, an SFN allows a simple network deployment where receivers see an equivalent channel obtained by the superposition of the channels relating to the multiple transmitters. However, when a receiver receives similar power levels from two transmitters, the channel frequency response will contain deep nulls due to destructive interference. For SFN with broadcasting stations equipped with a single antenna, by using a modified form of Alamouti [4] encoding, the new DVB–T2 standard provides an efficient means to exploit the presence of multiple transmitters. In other words, we obtain a distributed multiple–input single–output (MISO) system. In this configuration the data on the two transmitters are not identical but closely related, avoiding destructive interference. As a result, the SFN coverage is improved.

In this case (described as 2×1 MISO), the pilots have to provide two

independent channel estimations. Therefore, the number of pilots needs to be doubled. In this scenario the DVB–T2 standard uses the same pilot structure as for the single transmitter case (SISO), but half the corresponding size of the GI. The transmitters acting as antenna 1 use the exact same SISO pilot structure, while the transmitters acting as antenna 2 invert the pilots, modulating alternate pilot subcarriers.

2.8 Performance and Mode Comparison

The DVB–T2 standard provides a large set of transmitter configurations. The choice of some parameters is also dictated by the network deployment, an example being the GI duration and SP pattern configuration, which are related to the maximum tolerable delay channel spread, including possible SFN transmissions. For other parameters, such as the code rate and constellation size, the choice is guided by noise level and channel statistics. For various considered configurations (OFDM size, code rate, constellation, and pilot configuration) Table 2.1 shows the maximum throughput (in bits per second) that can be achieved by DVB–T2. In order to understand under which channel conditions these rates can be achieved with quasi error free reception, we simulated two relevant cases, and compared the performance of DVB–T and DVB–T2. The first case is a typical SFN deployment of the DVB–T standard (adopted, e.g., in Italy), which includes 8 K FFT, 1/4 GI, 64–QAM constellation, and 2/3 convolutional code rate. On a fixed Rician channel [2] DVB–T achieves a data rate of 19.91 Mb/s at a SNR of 17.1 dB with quasi error free reception (or equivalently a BER of 10^{-4} at the convolutional decoder output). For the same channel, DVB–T2 with extended bandwidth and parameters 32 K FFT, 1/16 GI, PP8, 256 QAM, and LDPC code rate 3/5 provides a data rate of 34.23 Mb/s (at BER of 10^{-4} at the LDPC decoder output), which is 72 percent higher. The second case is a multifrequency network DVB–T configuration (adopted in the United Kingdom), which includes 2K FFT, 1/32 GI, 64–QAM, and 2/3 code rate, yielding 24.13 Mb/s at an SNR of 18.9 dB in a fixed Rician channel for quasi error free conditions. Correspondingly, a DVB–T2 configuration with extended bandwidth, 32K FFT, 1/128 GI, PP7, 256 QAM, and an LDPC code rate of 3/5 provides a data rate of 36.14 Mb/s, which is about 50 percent higher. In both cases we observe that DVB–T2 allows HDTV MPEG–4 transmissions.

Chapter 3

Multiantenna Systems Overview

The use of multiple antennas at the transmit and/or at the receive side of a wireless communication system has the potential of greatly increase the capacity of the link [10]. Beyond the time and frequency dimensions that are commonly used in the design of communications systems, the use of multiple antennas provide a further dimension for transmission, denoted *space* dimension. Moreover, since a link with multiple antennas can be seen as a system with multiple input and multiple output (MIMO) signals, multiantenna systems are often referred to as MIMO systems.

In general, MIMO provides two gains:

- **Spatial diversity (SD) gain**: the reliability of a communication can be improved by exploiting the spatial diversity, i.e., the propagation of information through different paths.
- **Spatial multiplexing (SM) gain**: the spatial dimension in exploited to convey a higher bit rate.

Analytically, these two gains are defined as follows. Let us suppose that we aim at communicating at rate R when the average SNR of the communication system is Γ. Let $p_{\text{out}}(R)$ be the probability that the communication system does not provide the rate R, due to fading of the channel. Then, the channel achieves a multiplexing gain r and a diversity gain d when

$$R = r \log \Gamma \tag{3.1}$$

and

$$p_{\text{out}}(R) \approx \Gamma^{-d}, \tag{3.2}$$

respectively. We observe that the multiplexing gain dictates the achievable throughput, while the diversity gain dictates the probability for which the throughput is achieved.

On a plot showing the bit error rate (BER) as a function of Γ, the diversity gain determines an horizontal shift of the curve, while a multiplexing gain affects the slope of the curve, i.e.,

$$\text{BER} \propto f\left[(d\Gamma)^r\right] . \tag{3.3}$$

The number of transmit and receive antennas, together with the statistics of the channel limit the maximum diversity and multiplexing gains. Various techniques have been devised to exploit the benefits of MIMO, often with the aim of achieving either full diversity gain or full multiplexing gain. Among SD–oriented techniques we mention delay diversity (DD), space–time block codes (STBC) [4, 11] and space–time trellis codes (STTC) [12]. For the SM–oriented techniques we mention the vertical Bell laboratory space time (V–BLAST) system [13].

An alternative classification of MIMO techniques is related to the possibility of adapting the transmission to the channel conditions (precoding). In this case, the transmitter must know the particular channel on which transmission takes place and adapt the rate, the gain and the phase of the transmitted signal on each antenna, e.g., in order to maximize the capacity. Although precoding ensures the best performance of a MIMO system, it cannot be considered in a broadcast scenario, since each user experiences a different channel and a return channel is difficult to be implemented. Hence, in the following we focus on MIMO techniques that do not involve precoding.

3.1 ST Block Codes

The ST trellis encoder maps the information bits into N_T streams of complex symbols, that are transmitted simultaneously on the N_T antennas. Usually, STTC codes are designed to minimize the pairwise error probability, i.e., the probability of decoding the wrong code word [12]. The maximum likelihood (ML) ST trellis decoding has a complexity that grows exponentially with the diversity gain and the transmission rate [12]. Hence, ST block codes were originally introduced to reduce the decoding complexity.

Most of STBCs can be described as linear dispersion (LD) codes. With LD codes, a block of data bits is mapped into Q complex symbols s_q, $q = 1, 2, \ldots, Q$, taken from a real or complex constellation. The symbols transmitted by the N_T antennas for τ consecutive symbol periods are grouped into the codematrix $\mathbf{S}$ of size $N_T \times \tau$, where entry $S_{m,n}$ is the symbol transmitted at time m from antenna n. In general, a LD code is characterized by matrices $\mathbf{C}_q$ and $\mathbf{D}_q$, with $q = 1, 2, \ldots, Q$, each of size $N_T \times \tau$, and the codematrix is

generated as

$$\mathbf{S} = \sum_{q=1}^{Q} \left[s_q \mathbf{C}_q + s_q^* \mathbf{D}_q \right] . \tag{3.4}$$

Apart from an easier mathematical analysis and design, the linearity of LD codes enables the use of ML sphere decoding [14], achieving full performance of the code compared to other suboptimal decoders [15].

The general model of LD codes comprise many important examples: a) the Alamouti code, described in Section 3.1.1; b) orthogonal and non–orthogonal STBC and c) V–BLAST code. It is then clear that the criterion chosen for the design of matrices $\mathbf{C}_q$ and $\mathbf{D}_q$ determines the characteristics of the code, as well as its decoder implementation complexity.

The main criteria for the design of LD codes are:

- **Orthogonal design**: symbols are transmitted along orthogonal space–time dimensions. In this case, the ML receiver boils down to a simple linear receiver. Orthogonal design for complex data constellations exists only for two transmit antennas, while for real constellations can be derived for two, four and eight transmit antennas [4, 11]. Orthogonal design is possible for an arbitrary number of transmit antennas with the penalty of a rate loss [11].

- **Maximum mutual information** between the transmit and receive signal: can be used for an arbitrary number of transmit and receive antennas and can be decoded in a variety of ways, including successive nulling and cancelling (V–BLAST) and sphere decoding [16]. These codes maximize the ergodic capacity of the equivalent MIMO system but error probability performance may not be good.

- **Frame theory design**. In this case, the codes achieves still the maximum ergodic capacity but have also a better performance in terms of error probability than the maximum mutual information design [17].

- **Threaded algebraic design**: provides a systematic approach for the construction of ST codes that are full–diversity, full–rate and low–complexity for systems with an arbitrary number of transmit and receive antennas [18].

- **Cyclic division algebra criterion**: the aims with this design are: *a*) nonzero lower bound on the coding gain, which is independent of the spectral efficiency; *b*) shaping constraint that ensures the energy efficiency of the code and *c*) uniform average transmitted energy per

antenna. With this criterion, Golden codes are derived [19] for a system with two transmit and two receive antennas. A generalization for a larger number of transmit antennas has been derived in [20].

3.1.1 STBC with Orthogonal Design

A very important example of STBC with orthogonal design is the Alamouti code [4], which provides coding for a system with $N_T = 2$ and $N_R = 2$ antennas and complex constellation symbols. For this scheme $Q = 2$ symbols are transmitted over two symbol periods with the code matrix

$$\mathbf{S} = \begin{bmatrix} s_1 & -s_2^* \\ s_2 & s_1^* \end{bmatrix}. \tag{3.5}$$

The ML receiver for the Alamouti code can be easily implemented with a maximum ratio combining (MRC) receiver. Let $r_k^{(n)}$ be the sample received on antenna n at time k, we form the signals $\tilde{s}_1$ and $\tilde{s}_2$ as a replica of s_1 and s_2 respectively

$$\begin{aligned} \tilde{s}_1 &= h^{(1,1)*} r_1^{(1)} + h^{(1,2)} r_1^{(2)*} + h^{(2,1)*} r_2^{(1)} + h^{(2,2)} r_2^{(2)*}, \\ \tilde{s}_2 &= h^{(1,2)*} r_1^{(1)} - h^{(1,1)} r_1^{(2)*} + h^{(2,2)*} r_2^{(1)} - h^{(2,1)} r_2^{(2)*}. \end{aligned} \tag{3.6}$$

The simple decoder of the Alamouti code is among the reasons for its adoption in various standards, including WCDMA and CDMA2000. Moreover, the Alamouti code achieves full diversity for any complex constellation. Note however that simple decoding is possible only if the channel does not change in time, so that it can be considered time–invariant at least for two consecutive symbol periods.

As mentioned above, generalization of orthogonal design to other number of transmit antennas and complex constellations is difficult and can be achieved exactly only for a few antenna configurations.

Rotated and Scaled Alamouti It improves the standard Alamouti scheme with $N_T = 2$ as suggested by Willems [21]. In the rotated and scaled Alamouti code, we still transmit two symbols over two symbol periods but symbols transmitted by the second antenna are re–mapped by a function Φ to obtain the code matrix

$$\mathbf{S}_{RSA}(\mathbf{a}, \varphi) = \begin{bmatrix} a_1 e^{j\varphi} & -a_2^* \\ \Phi(a_2) & \Phi(a_1^*) \end{bmatrix}, \tag{3.7}$$

where φ is chosen as follows [21]

$$\varphi = \underset{\alpha}{\operatorname{argmax}} \min_{\mathbf{a}_1 \neq \mathbf{a}_2} |\det [\mathbf{S}_{RSA}(\mathbf{a}_1, \alpha) - \mathbf{S}_{RSA}(\mathbf{a}_2, \alpha)]|. \tag{3.8}$$

In particular, for a_i belonging to an integer 16–QAM constellation, the optimum value of φ turns out [21] to be $\varphi = 1.028$ and the re–mapping function is [22]

$$\Phi(a) = 2a - 5\left[\text{sign}\Re(a) + j\text{sign}\Im(a)\right], \tag{3.9}$$

where sign$\Re$ and sign$\Im$ are the sign of the real and imaginary part of a, respectively. Note that as $\Phi(a)$ is not a linear function.

3.1.2 STBC with Cyclic Division Algebra Design

As we mentioned above the aims of the design of LD codes with cyclic division algebra are: *a*) nonzero lower bound on the coding gain, which is independent of the spectral efficiency; *b*) shaping constraint that ensure the energy efficiency of the code and *c*) uniform average transmitted energy per antenna. Here we provide a relevant example of this design for the case of two transmit and two receive antennas ($N_T = 2$, $N_R = 2$), which comes under the name of Golden code [19], due to the use of the Golden number

$$\theta = \frac{1+\sqrt{5}}{2}. \tag{3.10}$$

For the Golden code, $Q = 4$ symbols are transmitted over two symbol periods ($\tau = 2$) by the codematrix

$$\mathbf{S} = \frac{1}{\sqrt{5}}\begin{bmatrix} \alpha(s_1 + s_2\theta) & \alpha(s_3 + s_4\theta) \\ \gamma\bar{\alpha}(s_3 + s_4\bar{\theta}) & \bar{\alpha}(s_1 + s_2\bar{\theta}) \end{bmatrix} \tag{3.11}$$

where

- $\bar{\theta} = 1 - \theta = \dfrac{1}{\theta} = \dfrac{1-\sqrt{5}}{2} \simeq 0.618 \in \mathbb{R}$;
- $\alpha = 1 + j - j\theta,\ \bar{\alpha} = 1 + j - j\bar{\theta} \in \mathbb{C}$;
- $\gamma = j$ (since $\gamma \in \mathbb{Z}[j]$ and $|\gamma| = 1$ then $\gamma \in \{1, -1, j, -j\}$. The choice comes from the fact that $\gamma \neq N_{\mathbb{K}/\mathbb{Q}(i)}(z)$ see [19])

The fundamental properties of Golden codes are [19]:

- **Full–rank**: matrix $\mathbf{S}$ is never singular.
- **Full–rate**: four source symbols generate four code symbols then the four degrees of freedom of the system are used.

- **Non–vanishing determinant for increasing rate**: whatever is the size of the data symbol constellation, the determinant of $\mathbf{S}$ is always larger than $\delta_{\min}(\mathcal{C}_g) = 1/5$.

- **Cubic shaping**: conversely to other space–time codes based on the theory of numbers, each layer of the Golden code is carved from a rotated version of the lattice $\mathbb{Z}[j]^2$.

- **Spectral efficiency**: The Golden code provides a spectral efficiency $2\log_2(M)$ bit/s/Hz where M is the cardinality of the symbol constellation.

Let us now consider the *peak–to–average power ratio* (PAPR), defined as follows

$$C_{\text{dB}} \triangleq 20\log_{10}\left(\frac{\max\limits_{s\in\mathcal{A}}|s|}{\sqrt{\text{E}[|s|^2]}}\right), \tag{3.12}$$

which measures the dispersion of the constellation $\mathcal{A}$. We evaluate the energy efficiency of the Golden code referring to four different source coding taken from standard modulations.

For BPSK data symbols, we obtain the Golden code with cardinality $2^3 = 8$. BPSK has $C = 0$ dB while the efficiency of the corresponding Golden codes is $C^G \simeq 2.78$ dB. For QPSK data symbols comprising $2^4 = 16$ points and in this case the efficiency is the same as for the BPSK case. For 8–PSK data symbols the Golden code constellation has $2^6 = 64$ points and its energy efficiency is still 2.7 dB. Finally for the 16–QAM data symbol constellation, the Golden code yields a constellation of $2^8 = 256$ points. All these coded constellations are illustrated in Fig. 3.1.

The energy efficiency for this constellation are $C \simeq 2.55$ dB for the original 16–QAM and $E^G \simeq 5.33$ dB for the corresponding Golden code. Tab. 3.1 summarizes all the results.

Constellation	$\lvert\mathcal{A}\rvert$	$\lvert\mathbf{G}(\mathcal{A}^4)\rvert$	C (dB)	C^G (dB)	$C^G - C$ (dB)
BPSK	$2^1 = 2$	$2^3 = 8$	0	2.78	2.78
QPSK	$2^2 = 4$	$2^4 = 16$	0	2.78	2.78
8–PSK	$2^3 = 8$	$2^6 = 64$	0	2.78	2.78
16–QAM	$2^4 = 16$	$2^8 = 256$	2.55	5.3275	2.78

Table 3.1: PAPR for Golden code.

Golden code should be seen as a particular case of a greater class of codes: the perfect STBCs. Already for the case of two transmit and two receive

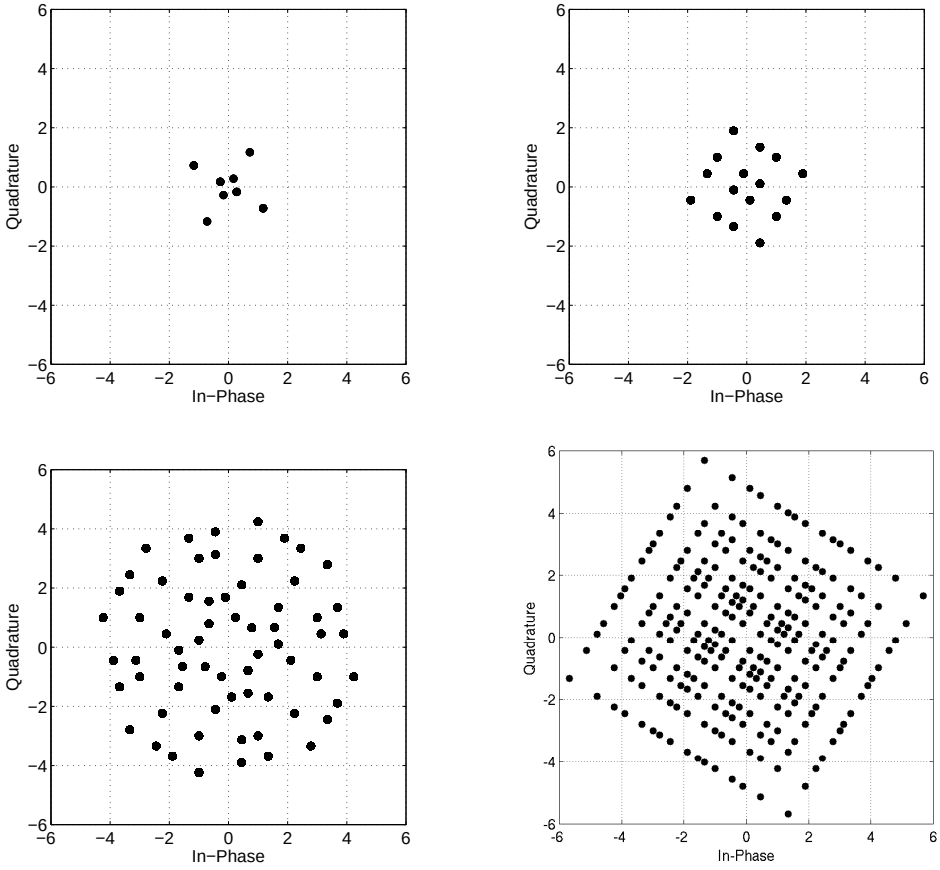

Figure 3.1: Constellation of Golden code with BPSK, QPSK, 8–PSK and 16–QAM symbols.

antennas, an infinite family of perfect STBCs, generalizing the Golden code construction, can be derived [20].

Furthermore, an explicit construction for other dimensions, such as 3×3, 4×4, and 6×6 MIMO systems has been derived in [20]. These codes are designed to satisfy the same criteria of the Golden code mentioned above. In particular, a square $M \times M$ STBC is called *perfect* code if and only if:

- it is a full–rate linear dispersion code using M^2–QAM information symbols;
- the minimum determinant of the infinite code is non zero (so that in particular the rank criterion is satisfied);
- the energy required to send the linear combination of the information symbols on each layer is similar to the energy used for sending the

symbols themselves (we do not increase the energy of the system in encoding the information symbols);

- it induces uniform average transmitted energy per antenna in all τ time slots, i.e., all the code symbols in the codematrix have the same average energy.

Lastly, we have compared Golden and Alamouti codes for BPSK and QPSK transmissions over random Rayleigh fading channels. We considered 500 realizations of 2×2 channels, which are time–invariant for 1000 symbols. The comparison between the two curves is depicted in Fig. 3.2, where the parameter σ is defined as

$$\sigma_{\mathrm{dB}} \triangleq 10 \log_{10} \left(r \frac{E_s}{N_0} \right) , \tag{3.13}$$

and r is the code rate.

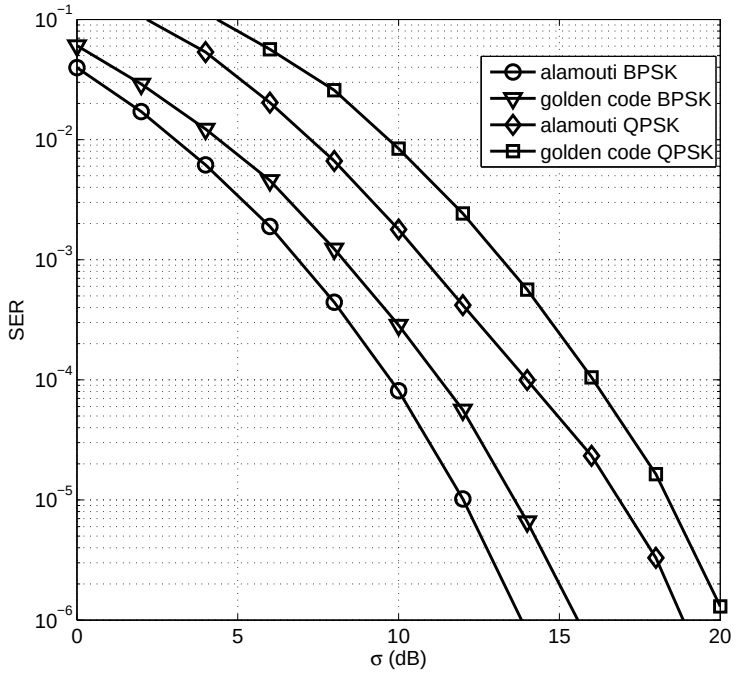

Figure 3.2: Performance comparison between Golden and Alamouti codes.

For the Alamouti code we have $r = 1/2$, while for the Golden code we have $r = 1$. We observe that the Alamouti code outperforms the Golden code by about 2 dB. Yet, we recall that the Golden code achieves a double rate with respect to the Alamouti codes.

3.1.3 Rotation–based STBC

We recall a STBC proposed by Yao and Wornell [23] also known as *Tilted QAM*. It is a full–rate STBC for $N_T = 2$ and its codewords are in the following form

$$\mathbf{S} = \begin{bmatrix} s_1 \cos\theta_1 - s_2 \sin\theta_1 & s_3 \sin\theta_2 + s_4 \cos\theta_2 \\ s_3 \cos\theta_2 - s_4 \sin\theta_2 & s_1 \sin\theta_1 + s_2 \cos\theta_1 \end{bmatrix} \tag{3.14}$$

where s_j is a symbol of a generic QAM constellation and

$$\theta_1 = \frac{1}{2}\arctan\left(\frac{1}{2}\right), \qquad \theta_2 = \frac{1}{2}\arctan(2). \tag{3.15}$$

The most important property of this method is that it is a unitary transformation that achieve the *full diversity–multiplexing frontier* [24].

3.2 Delay Diversity

The delay diversity (DD) technique allows to convert spatial diversity into multipath diversity. For transmissions over flat fading channels, the DD transmitter sends each data symbol sequentially on all the antennas [25–28]. Let s_k be the data signal and $h^{(m,n)}$ the channel gain between transmit antenna n and receive antenna m. The received signal on antenna m at time k can be written as

$$r_k^{(m)} = \sum_{n=1}^{N_T} h^{(m,n)} s_{k+n-1} + w_k^{(m)}, \tag{3.16}$$

where $w_k^{(m)}$ is the noise term. We observe that the received signal can be seen as a filtered version of the transmitted signal, where the filter taps are the spatial channel gains. At the receiver, equalizers plus threshold detectors or ML decoders can be used to recover the data sequence.

When the channel is frequency selective, full–diversity is achieved by delaying the transmission on each antenna by the length of the channel. In particular, let $h_\ell^{(m,n)}$ be the channel impulse response from antenna n to antenna m, with $\ell = 0, 1, \ldots, N_\tau - 1$. By delaying the sequential transmission of the data symbols by N_τ we obtain

$$r_k^{(m)} = \sum_{\ell=0}^{N_\tau - 1} \sum_{n=1}^{N_T} h_\ell^{(m,n)} s_{k+(n-1)N_\tau - \ell} + w_k^{(m)}, \tag{3.17}$$

and the received signal can be seen as a filtered version of the transmitted signal, where the filter taps are

$$[h_0^{(m,1)}, h_1^{(m,1)}, \ldots, h_{N_\tau-1}^{(m,1)}, h_0^{(m,2)}, h_1^{(m,2)}, \ldots, h_{N_\tau-1}^{(m,2)}, \ldots \\ \ldots, h_0^{(m,N_T)}, h_1^{(m,N_T)}, \ldots, h_{N_\tau-1}^{(m,N_T)}]\,. \quad (3.18)$$

Note that the combination of DD with frequency–domain equalizers (including multicarrier transmissions as OFDM) is convenient, as these systems capture multipath diversity, [29]. On the other hand, since the resulting channel (3.18) is N_T times longer than the single antenna (or single input single output, SISO) channel, a spectral inefficiency may be occurred. In fact, both CP and UW become $N_T N_\tau$ long. In order to overcome this problem, cyclic delay diversity (CDD) has been proposed in [30]. CDD is also named multicarrier delay–diversity modulation (MDDM). Fig. 3.3 shows a schematic idea of CDD.

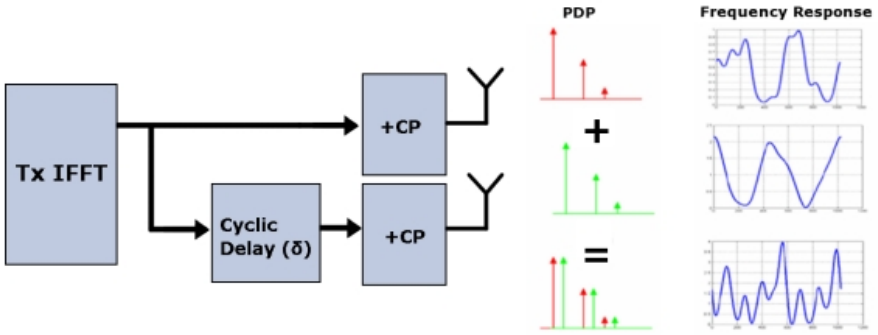

Figure 3.3: Cyclic Delay Diversity scheme.

The CDD modulates the block of M symbols $\mathbf{s} = [s_0, s_1, \ldots, s_{M-1}]$ by FFT into the block $\tilde{\mathbf{s}}$. For transmission on antenna n, the block $\tilde{\mathbf{s}}$ is cyclically shifted by nQ samples to obtain the block

$$\tilde{\mathbf{s}}^{(n)} = [\tilde{s}_{(nQ)_{\mathrm{mod}\,M}}, \tilde{s}_{(nQ+1)_{\mathrm{mod}\,M}}, \ldots, \tilde{s}_{(nQ+M)_{\mathrm{mod}\,M}}]\,, \quad (3.19)$$

where $(x)_{\mathrm{mod}\,M}$ is the reminder of division of x by M. Each cyclically shifted block is extended with CP and transmitted on the corresponding antenna. Note that the spectral efficiency of this modulation is the same as that of a SISO OFDM system. If $Q > N_\tau$ and $M > N_\tau N_T$, the equivalent channel seen at the receiver is as in (3.18).

Chapter 4

Hard and Soft Detection of STBC

In the previous chapter we have introduced the multiantenna systems also known as multiple input–multiple output (MIMO) wireless communication systems. These antenna configurations have been considered in the next generation of European digital terrestrial television DVB–T2. Space–time block codes (STBCs) exploit the benefits of MIMO and they can be designed following different criteria. Maximum–likelihood (ML) decoding of STBCs is one of the open problems due to its complexity: the exhaustive search increases exponentially with the size of the STBC matrix.

Linear STBCs belongs to a very common class of STBCs. Since they have an equivalent lattice sphere packing representation they can be decoded as arbitrary lattice codes. The very common tool for ML detection of lattice codes with a reasonable complexity is the sphere decoder. A good mathematical and analytical develop of lattice codes and sphere decoder is presented in [31]

In this chapter we start with an introduction to the sphere decoder and a more versatile version which suits better for different purposes. We show how this method can perform the hard ML detection for linear STBCs and also how it can be useful for soft detection. A new method to improve the soft detection is discussed and finally we show the performance comparison for different STBCs.

4.1 Sphere Decoder

The *sphere decoder* is a method to get a ML detection in lattice systems. The detection of most of the STBCs, such as the Golden Code, can be efficiently provided by the sphere decoder. Here we give an implementation according with [32].

Let $\mathcal{P}$ the finite alphabet of an L–PAM constellation, e.g. $\mathcal{P}_4 = \{-3, -1, +1, +3\}$, and let $\mathcal{C}$ the finite alphabet of an L^2–QAM. We define the finite lattice Λ generated by matrix $\mathbf{M} \in \mathbb{C}^{m\times m}$ and a column vector $\mathbf{a} = [a_1, a_2, \ldots, a_m]^{\mathrm{T}} \in \mathcal{C}^m$ as

$$\Lambda \triangleq \{\mathbf{Ma} : a_i \in \mathcal{C}, i = 1, 2, \ldots, m\}. \tag{4.1}$$

For the input–output relation

$$\mathbf{r} = \mathbf{Ma} + \mathbf{w}\,, \tag{4.2}$$

with $\mathbf{w}$ m–size column vector of Gaussian noise, the ML detection find $\mathbf{a}$ that minimizes the metric $\|\mathbf{r} - \mathbf{Ma}\|^2$, where $\mathbf{r}$ is the received vector and $\mathbf{Ma}$ is any valid point of the lattice. The cardinality of Λ, namely, total number of points in Λ, is exponential in m, with a base equal to the size of the complex signal constellation $\mathcal{C}$. For example, in case of 8×8 MIMO and 64QAM, the lattice has 64^8 points. A complete enumeration of the lattice points is not feasible.

The main idea of sphere decoding is to enumerate lattice points inside a hypersphere centered on $\mathbf{r}$ and reduce the radius d of the sphere each time a new point $\mathbf{a}'$ is found, as follows

$$d_{\text{new}} = \|\mathbf{r} - \mathbf{Ma}'\|\,. \tag{4.3}$$

This drastically decreases the number of enumerated points but still ensures the closest point criterion.

For computational reasons, we shall replace the complex–valued model by its real–valued equivalent. To this end, we define the vectors belonging to $\mathbb{R}^{2m}$ $\mathbf{r}_{\mathbb{R}}$, $\mathbf{a}_{\mathbb{R}}$ and $\mathbf{w}_{\mathbb{R}}$ as

$$\mathbf{r}_{\mathbb{R}} = \begin{bmatrix} \Re(\mathbf{r}) \\ \Im(\mathbf{r}) \end{bmatrix}, \qquad \mathbf{a}_{\mathbb{R}} = \begin{bmatrix} \Re(\mathbf{a}) \\ \Im(\mathbf{a}) \end{bmatrix}, \qquad \mathbf{w}_{\mathbb{R}} = \begin{bmatrix} \Re(\mathbf{w}) \\ \Im(\mathbf{w}) \end{bmatrix} \tag{4.4}$$

and the real $2m \times 2m$ matrix as

$$\mathbf{M}_{\mathbb{R}} = \begin{bmatrix} \Re(\mathbf{M}) & -\Im(\mathbf{M}) \\ \Im(\mathbf{M}) & \Re(\mathbf{M}) \end{bmatrix}. \tag{4.5}$$

Then the real–valued equivalent of model (4.2) is given by

$$\mathbf{r}_{\mathbb{R}} = \mathbf{M}_{\mathbb{R}}\mathbf{a}_{\mathbb{R}} + \mathbf{w}_{\mathbb{R}}. \tag{4.6}$$

By letting $n = 2m$, the ML detection criterion becomes

$$\hat{\mathbf{a}}_{\mathbb{R}} = \underset{\mathbf{b}\in\mathcal{P}^n}{\arg\min} \|\mathbf{r}_{\mathbb{R}} - \mathbf{M}_{\mathbb{R}}\mathbf{b}\|^2\,, \tag{4.7}$$

since $\mathcal{P}_L$ is obtained from $\mathbb{Z}_L = \{0, \ldots, L-1\}$ by applying a suitable translation and scaling $\mathcal{P}_L = 2\mathbb{Z}_L - (L-1)$, we observe that

$$\hat{\mathbf{a}}_{\mathbb{R}} = 2\hat{\mathbf{u}} - \mathbf{1}_L, \qquad \mathbf{a} = 2\mathbf{u} - \mathbf{1}_L, \qquad \hat{\mathbf{u}}, \mathbf{u} \in \mathbb{Z}_L^n. \tag{4.8}$$

Let $\mathbf{1}_L \triangleq (L-1)\,[1, \ldots, 1]^{\mathrm{T}}$. Then we can define the translated and rescaled version of the received vector as $\mathbf{y} \triangleq (\mathbf{r}_{\mathbb{R}} + \mathbf{M}_{\mathbb{R}}\mathbf{1}_L)/2$ and we can simplify (4.7) as

$$\hat{\mathbf{u}} = \underset{\mathbf{u} \in \mathbb{Z}_L^n}{\arg\min} \left\| \mathbf{y} - \mathbf{M}_{\mathbb{R}}\mathbf{u} \right\|^2 . \tag{4.9}$$

Next, we perform the QR decomposition of matrix $\mathbf{M}_{\mathbb{R}}$. It can be shown [33] that a real matrix can always be written as the product of two matrices, the first ($\mathbf{Q}$) orthogonal and the second ($\mathbf{R}$) upper triangular with non–negative diagonal elements r_{ii}

$$\mathbf{M}_{\mathbb{R}} = \mathbf{QR}: \quad \mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}_n \quad (r_{ii} \geq 0\ , \forall i). \tag{4.10}$$

Note that the orthogonal matrix $\mathbf{Q}$ maintains the Euclidean norm of any vector $\mathbf{v}$, i.e., $\|\mathbf{Qv}\|^2 = \|\mathbf{v}\|^2$. Hence, premultiplying (4.9) by $\mathbf{Q}^{\mathrm{T}}$ and defining

$$\mathbf{z} \triangleq \mathbf{Q}^{\mathrm{T}}\mathbf{y}, \tag{4.11}$$

minimization (4.9) becomes

$$\hat{\mathbf{u}} = \underset{\mathbf{u} \in \mathbb{Z}_L^n}{\arg\min} \left\| \mathbf{z} - \mathbf{Ru} \right\|^2 . \tag{4.12}$$

Due to the upper triangular shape of $\mathbf{R}$, (4.12) can also be written as

$$\hat{\mathbf{u}} = \underset{[u_1, \ldots, u_n]^{\mathrm{T}} \in \mathbb{Z}_L^n}{\arg\min} \sum_{i=1}^{n} \left(z_i - \sum_{j=i}^{n} r_{ij} u_j \right)^2 . \tag{4.13}$$

Now, if we consider a hypersphere $\mathcal{S}$ centered in $\mathbf{z}$ with square radius d^2 large enough to contain at least one lattice point, i.e., $\mathcal{S} \cap \Lambda \neq \varnothing$, there exists a n–size vector $\mathbf{u} \in \mathbb{Z}_L^n$ such that

$$\sum_{i=1}^{n} \left(z_i - \sum_{j=i}^{n} r_{ij} u_j \right)^2 < d^2 \tag{4.14}$$

and, to a greater extent, all the conditions

$$\sum_{i=k}^{n} \left(z_i - \sum_{j=i}^{n} r_{ij} u_j \right)^2 < d^2, \quad k = 1, \ldots, n \tag{4.15}$$

are satisfied. We observe that the k–th condition depends only on the last $n-k+1$ elements of $\mathbf{u}$, i.e., $\mathbf{u}_k = [u_k, \ldots, u_n]^{\mathrm{T}}$.

Let us define the elements $(\xi_1, \ldots, \xi_n)$ with $\xi_n = 0$ and

$$\xi_k = \sum_{j=k+1}^{n} r_{kj} u_j \,, \quad k = 1, 2, \ldots, n-1 \,, \tag{4.16}$$

and let T_k be the cumulative square distance of the found point $\mathbf{Ru}$ from the received vector $\mathbf{z}$, computed recursively, starting from $T_n = 0$, as follows:

$$T_{k-1} = T_k + |z_k - (r_{kk} u_k + \xi_k)|^2 \,. \tag{4.17}$$

Let also define the integer lower and upper bounds of the searched point as

$$A_k = \max \left\{ \left\lceil \frac{1}{r_{kk}} \left(z_k - \xi_k - \sqrt{d^2 - T_k} \right) \right\rceil , 0 \right\} , \tag{4.18}$$

$$B_k = \min \left\{ \left\lfloor \frac{1}{r_{kk}} \left(z_k - \xi_k + \sqrt{d^2 - T_k} \right) \right\rfloor , L-1 \right\} , \tag{4.19}$$

where d^2 is once again the square radius.

Now, if we solve each condition (4.15) with respect to u_k, we obtain

$$A_k \leq u_k \leq B_k \tag{4.20}$$

The Algorithm We describe the fundamental steps of the sphere decoder algorithm implemented as Algorithm I in [32], which is an improved variant of the classical Viterbo–Boutros algorithm [34].

The algorithm starts assuming $d^2 = +\infty$. When

- (*Step 1.*) Set $d^2 = +\infty$ and $k = n$.
- (*Step 2.*) Compute T_k.
- (*Step 3.*) If $d^2 > T_k$, we set $u_k = A_k$; we decrease k by one and go to Step 2.
- (*Step 4.*) If $d^2 < T_k$, we are going outside of the hypersphere and the point is not valid. We increase k by one and set $u_{k+1} = u_{k+1} + 1$. If $u_{k+1} \leq B_{k+1}$, i.e., the new point is inside the bound, we go to Step 2. Otherwise, we keep increasing k and trying to update u_k until the new point is inside the bound $u_k < B_k$. If we arrive at $k = n$, there are no points inside the hypersphere and the algorithm terminates.

- (*Step 5.*) When we reach the top level $k = 1$, we first compute the full square distance of point $\mathbf{u}$ as $d^2_{\text{new}} = \|\mathbf{z} - \mathbf{R}\mathbf{u}\|^2 = T_1 + |z_1 - (r_{11}u_1 + \xi_1)|^2$.

  If d^2_{new} is less than the actual distance, we have found a point inside the hypersphere and we set both i) the new square radius for the hypersphere as $d^2 = d^2_{\text{new}}$ and ii) $k = n$, and iterate from Step 2.

In order to obtain the desired vector $\hat{\mathbf{a}} \in \mathcal{C}^m$, we rescale and translate $\hat{\mathbf{u}}$ as in the first of (4.8) to obtain $\hat{\mathbf{a}}_{\mathbb{R}}$ and lastly from (4.4) we combine the first (last) components of $\hat{\mathbf{a}}_{\mathbb{R}}$ to get $\Re(\hat{\mathbf{a}})$ ($\Im(\hat{\mathbf{a}})$), and $\hat{\mathbf{a}} = \Re(\hat{\mathbf{a}}) + j\Im(\hat{\mathbf{a}})$.

4.1.1 The Complex Sphere Decoder

We introduce now a new complex sphere decoding algorithm (CSD [35]) whose computational cost is significantly lower than the above standard sphere decoder.

In early applications of the SD to multiple antenna systems, the complex system was decoupled into its real and imaginary parts to form a real–valued system, hence its dimension was twice that of the original system. This decoupling is permissible when the signal constellation has a lattice representation, such as in rectangular QAM. However, for other complex constellations, such as PSK, the real–valued SD is inefficient, since computational resources are wasted anytime infeasible candidates arise in the iterative process.

Recently in [36], a first version of the CSD was introduced, which eliminates the need to decouple the system, and thus solves the problem of infeasible candidates and it has computational advantage over the real–valued SD.

The new algorithm differs from the original SD in the following ways:

- the use of a new system ordering criterion to maximize the probability that the first feasible solution is optimal; the application of this idea, to reduced searching iterations, represents the most important step in the new algorithm;

- the use of a minimum mean square error criterion in the selection of candidates;

- the application of a "best–first" search strategy, which attempts to find new feasible solutions with as few number of steps as possible.

In [35] the idea is to represent the signal constellation as concentric rings, as in Fig. 4.1, each ring intersects points of the constellation. To obtain

the complete set of candidates, this process is applied to all rings in the constellation.

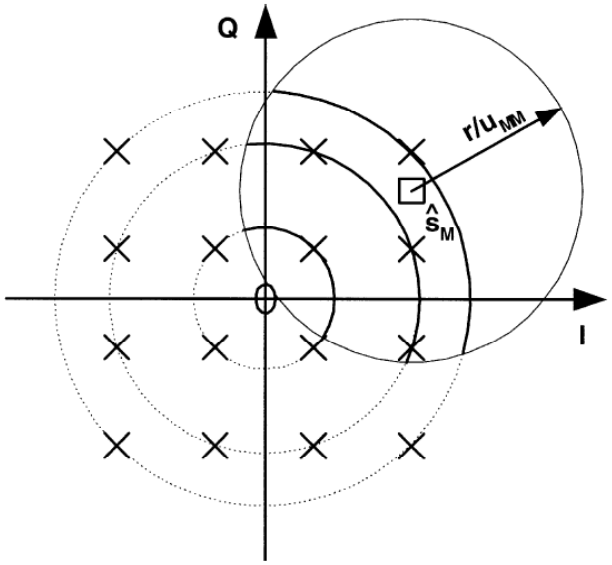

Figure 4.1: Intersection of search disk and a 16–QAM constellation can be obtained by considering the QAM constellation as three concentric rings of points.

In the original SD method, once a new feasible solution is obtained, the search process starts over from the root node in the tree using the new smaller radius. In the new algorithm, an alternate search strategy is applied. Rather than restarting the search from the root node, the search resumes with the path that is closest to completion. The motivation behind this is to form new feasible solutions as quickly as possible. We illustrate this with a short example, in Fig. 4.2

The Algorithm We refer to the transmission model (4.2) where $\mathbf{M} \in \mathbb{C}^{N_R \times N_T}$ and $\mathbf{w} \in \mathcal{CN}(0, N_0\mathbf{I}_{N_R})$. Multiplying both sides of the model from the left by $\mathbf{M}^H$, where "^{H}" denotes the conjugate transpose; let $\mathbf{y} = \mathbf{M}^H\mathbf{r}$, $\mathbf{G} = \mathbf{M}^H\mathbf{M}$ Hermitian and $\mathbf{n} = \mathbf{M}^H\mathbf{w}$, we obtain

$$\mathbf{y} = \mathbf{G}\mathbf{a} + \mathbf{n}, \quad \mathbf{G} \in \mathbb{C}^{N_T \times N_T}, \ \mathbf{n} \in \mathcal{CN}(0, N_0\mathbf{G}). \tag{4.21}$$

The Cholesky decomposition of an Hermitian matrix ensures that there exists a matrix $\mathbf{F}$ lower triangular such that $\mathbf{G} = \mathbf{F}^H\mathbf{F}$. For all the cases where $\mathbf{F}^H$ is non–singular we can multiply both sides of (4.21) from the left

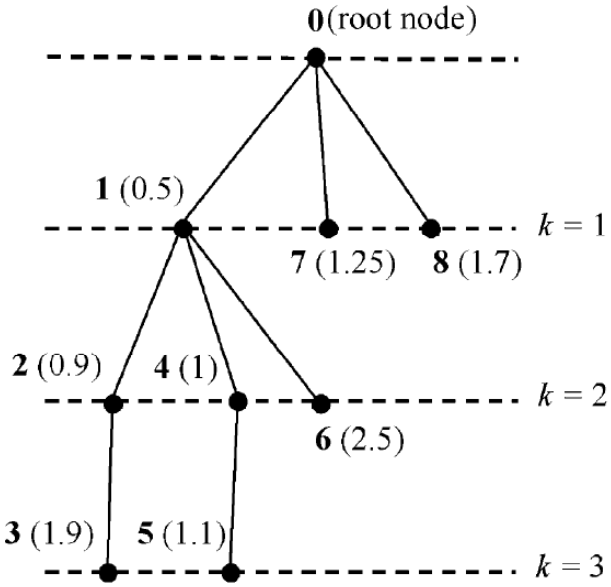

Figure 4.2: Example tree formed by new complex sphere decoder. In boldface the searching sequence number, in brackets the square distance from the detected point, dashed lines and the k values represent the searching levels.

by $(\mathbf{F}^H)^{-1}$; let $\tilde{\mathbf{y}} = (\mathbf{F}^H)^{-1}\mathbf{y}$ and $\mathbf{v} = (\mathbf{F}^H)^{-1}\mathbf{n}$, we obtain

$$\tilde{\mathbf{y}} = \mathbf{F}\mathbf{a} + \mathbf{v}, \quad \mathbf{v} \in \mathcal{CN}(0, N_0\mathbf{I}_{N_T}). \tag{4.22}$$

It can be proved that this set of operations maintains the Euclidean distance from the received point to each lattice point, i.e.

$$\|\mathbf{r} - \mathbf{M}\mathbf{a}\|^2 = \|\tilde{\mathbf{y}} - \mathbf{F}\mathbf{a}\|^2, \quad \forall \mathbf{a} \in \mathbb{C}^{N_T} \tag{4.23}$$

which is the fundamental question for the correct ML computation.

The algorithm for the new complex SD is given as follows:

- (*Step 1.*) Compute $\tilde{\mathbf{y}} = (\mathbf{F}^H)^{-1}\mathbf{H}^H\mathbf{r}$ and set the initial square searching radius $c = 10N_0$ that ensures $\mathrm{P}[\|\mathbf{v}\|^2 \leq c] \simeq 0.99$[1]. Set flag $= 0$.
- (*Step 2.*) Initialize $k = 0, \mathbf{z} = \tilde{\mathbf{y}}, \xi = 0$. Also, initialize the candidate sets $S_j = \varnothing$ for $j = 1, \ldots, N_T$, and set $\hat{\mathbf{a}} = \mathbf{0}$. If flag $= 1$, set $c = 1.2c$.
- (*Step 3.*) Set $k = k + 1$ and $\lambda_k = z_k/F_{kk}$. Obtain the candidates (if any) for a_k in the following way: for all ρ radius associated with a

[1] $\|\mathbf{v}\|^2/N_0 \sim \chi^2$ with $2N_T$ degrees of freedom.

particular ring of the k–th constellation we compute $\eta = (\rho^2 + |\lambda_k|^2 - cF_{kk}^{-2})/(2\rho|\lambda_k|)$. If $\eta < -1$, all the signal points on the ring are candidates for a_k. If $\eta > 1$, none of the points on the ring are candidates for a_k. For $-1 \leq \eta \leq 1$, only points which are located within the arc of the ring defined by the angle range $\arg(\lambda_k) + [-\cos^{-1}(\eta), \cos^{-1}(\eta)]$ are candidates for a_k.

- (*Step 4.*) If candidates are found, goto step 7. Otherwise, goto step 5.

- (*Step 5.*) If the candidate sets are all empty, goto step 6. Otherwise, goto step 13.

- (*Step 6.*) If no solution has been found yet, set flag = 1 and goto step 2. Otherwise, STOP and report the "current–best" solution.

- (*Step 7.*) For each candidate x found in step 3, compute $\delta = |\lambda_k - x|^2$. Denote the smallest δ as δ_k. Let the set S_k comprise the candidates with δ's not equal to δ_k and arrange them in increasing order of their δ's so that the first candidate in S_k is associated with the smallest δ and the last with the largest. For the set S_k, store the quantities $\mathbf{z}$, $\{\hat{a}_i\}_{i=1}^{k-1}$ and ξ. For each candidate in S_k, store their associated δ's.

- (*Step 8.*) Set $\hat{a}_k$ equal to the candidate associated with δ_k.

- (*Step 9.*) Set $\xi = \xi + F_{kk}^2 \delta_k$. If $\xi < c$, goto step 11. Otherwise, goto step 10.

- (*Step 10.*) If the candidate sets are all empty, goto step 6. Otherwise, goto step 13.

- (*Step 11.*) If $k < N_T$, set $z_{k+1} = z_{k+1} - \sum_{j=1}^{k} F_{k+1,j}\hat{a}_j$ and goto step 3.

- (*Step 12.*) If $k = N_T$, update the "current–best" solution with and let $c = \xi$. If the candidate sets are all empty, STOP and report the "current–best" solution. Otherwise, goto step 13.

- (*Step 13.*) Select the nonempty candidate set closest to the last tree level. From this candidate set, remove the first element and set the quantities $\mathbf{z}, \xi, \delta$, and $\{\hat{a}_i\}_{i=1}^{k}$ equal the values associated with the set and the chosen candidate. Set k equal to the level of the candidate set and $\delta_k = \delta$ and goto step 9.

4.2 Soft Decoding

More and more modern coding and decoding algorithms require soft bit metrics in place of the typical hard decision over detected symbols at the receiver. Soft information is then related to the bits involved in the sent symbols.

When we transmit a sequence of n symbols $\mathbf{a} = [a_1 \cdots a_n]^{\mathrm{T}}$ belonging to a M–ary alphabet, with $M = 2^b$, we are really transmitting a sequence of nb bits $\mathbf{c} = [c_1 \cdots c_{nb}]$ where $c_i \in \{0, 1\}$. Each symbol is obtained from a sequence of b bits by a suitable mapping $a_i = \mathrm{map}(c_{(i-1)b+1}, \ldots, c_{ib})$, typically with Gray configuration, i.e. for two symbols a_i and a_j with minimal euclidean distance, their corresponding bit sequences $\mathbf{c}_i$ and $\mathbf{c}_j$ have unitary Hamming distance. For example in Fig. 4.3 we have a Gray bit mapping of a 16–QAM constellation.

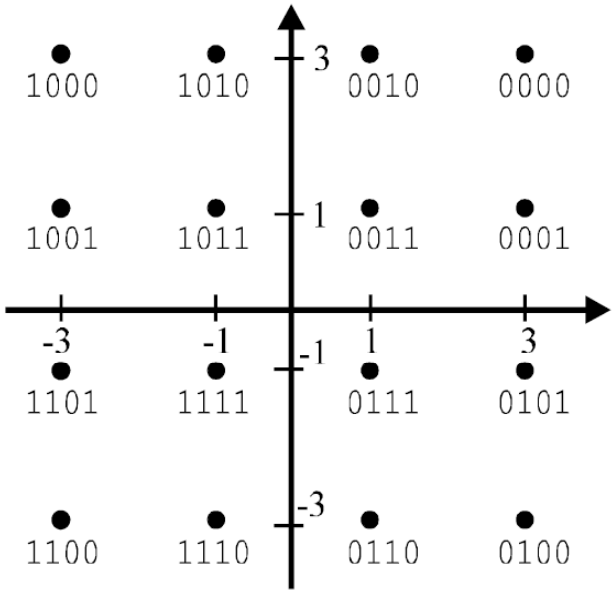

Figure 4.3: Gray bit mapping into a 16–QAM constellation.

At the receiver we have a sequence of samples, i.e. elements of the detection $\mathbf{r} = \mathbf{M}\mathbf{a} + \mathbf{w}$, and we are interested on computing the *a posteriori* log likelihood ratio (LLR) for each bit c_i involved in the transmission. LLR for the i–th bit is defined as

$$L(c_i) \triangleq \ln \frac{P(c_i = 1|\mathbf{r})}{P(c_i = 0|\mathbf{r})}. \tag{4.24}$$

In order to obtain a soft–output detector for STBCs, we resort to the max–log–MAP detector (see [37, pp. 677–678]). In particular, for the i–th

bit, let $\Lambda_i^{\pm 1}$ be the sets

$$\Lambda_i^{\pm 1} \triangleq \{\mathbf{Ma} | \mathbf{c} = \text{demap}(\mathbf{a}) \text{ and } c_i = \pm 1\}, \tag{4.25}$$

and define the lattice points at the shortest distance from $\mathbf{r}$ having bit $c_i = \pm 1$ as

$$\boldsymbol{\rho}_i^{\pm 1} \triangleq \underset{\boldsymbol{\rho} \in \Lambda_i^{\pm 1}}{\arg\min} \|\mathbf{r} - \boldsymbol{\rho}\|^2. \tag{4.26}$$

Then, the log–likelihood ratio (LLR) associated to bit c_i can be approximated as

$$L(c_i) \simeq -\frac{1}{N_0} \left(\|\mathbf{r} - \boldsymbol{\rho}_i^{+1}\|^2 - \|\mathbf{r} - \boldsymbol{\rho}_i^{-1}\|^2 \right) \tag{4.27}$$

whose computation requires the search of the two lattice points $\boldsymbol{\rho}_i^{+1}$ and $\boldsymbol{\rho}_i^{-1}$ that solve (4.26).

Here we present a method that computes (4.27) by the iterative Max–Log–MAP complex sphere decoder (I–CSD) proposed in [38]. Moreover we compare this method with a brand new solution. The solution we propose is called Group–wise Max–Log–MAP complex sphere decoder (G–CSD).

4.2.1 Iterative MLM Complex Sphere Decoder

The I–CSD proposed in [38] involves multiple stages of CSD to find the pair of candidates $\boldsymbol{\rho}_i^{+1}$ and $\boldsymbol{\rho}_i^{-1}$ for each LLR. For every received signal $\mathbf{r}$, the candidate with the shorter distance metric among the pair found from set Λ_i^{+1} and Λ_i^{-1} for every bit in the sequence $\{c_i\}, i = 1, \ldots, nb$, have the same minimum distance metrics d_{ML}^2 since one of the pair is the ML space time symbol estimate, $\mathbf{a}_{ML}$. The ML symbol $\mathbf{a}_{ML}$ is found by the sphere decoder at the first stage of the MLM sphere decoder architecture shown in Fig. 4.4. Then, the first stage finds the lattice point $\mathbf{M\hat{a}}$ and provides immediately half of candidates.

Therefore, having known the ML bit sequence $\{\hat{c}_i\}$ corresponding to $\mathbf{a}_{ML}$, the derivation of $L(c_i)$, only requires the search of a candidate and its corresponding distance metrics d_i^2 from the complementary set of $\Lambda_i^{\hat{c}_i}$, also noted as $\Lambda_i^{-\hat{c}_i}$, where the i–th bit is inverted to that of the ML solution. This is performed by the multiple sphere decoders at the second stage of the architecture shown in Fig. 4.4. Typically to perform a point searching inside a subset $\Lambda_i^{\hat{c}_i}$ cannot be done by the standard sphere decoder because $\Lambda_i^{\hat{c}_i}$ should not be decoupleable as well as a QAM. Hence this solution implementation requires the CSD at least for the second stage. Then CSD is applied nb times to lattices $\Lambda_i^{-\hat{c}_i}$ in order to find $\boldsymbol{\rho}_i^{-\hat{c}_i}$.

The method is described in steps in the following.

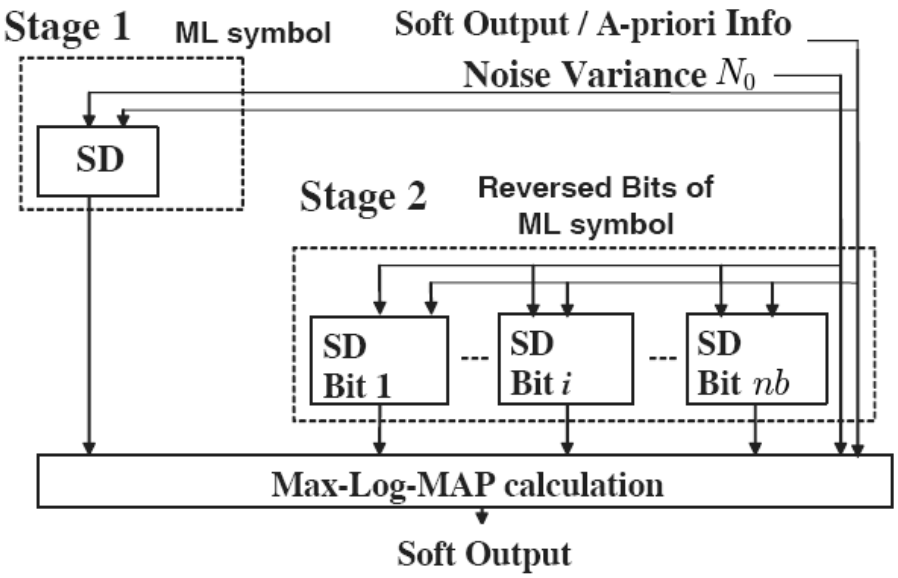

Figure 4.4: Architecture of the I–CSD.

- (*Input.*) Received vector $\mathbf{r}$, lattice generator $\mathbf{M}$, constellation $\mathcal{C}$ and noise variance N_0.

- (*Output.*) Metrics $L(c_i)$ for $i = 1, \ldots, nb$.

- (*Step 1.*) Set the searching set $\mathcal{S} = \mathcal{C}_1 \times \ldots \times \mathcal{C}_n$ with $\mathcal{C}_\ell = \mathcal{C}$ for $\ell = 1, \ldots, n$ then find the closest lattice point to $\mathbf{r}$ and its square distance $(\hat{\mathbf{a}}, d^2_{ML}) = \mathrm{CSD}(\mathbf{r}, \mathbf{M}, \mathcal{S}, 10N_0)$. Compute $\hat{\mathbf{c}} = \mathrm{demap}(\hat{\mathbf{a}})$ and set $\ell = 1$, $k = 1$.

- (*Step 2.*) Set $\mathcal{C}_\ell = \{a : \mathrm{demap}(a)_k \neq \mathrm{demap}(\hat{a}_\ell)_k\}$, let $\mathcal{C}_i = \mathcal{C}$ for $i \neq \ell$ and set the searching set $\mathcal{S} = \mathcal{C}_1 \times \ldots \times \mathcal{C}_n$. Find the new closest lattice point to $\mathbf{r}$ and its square distance $(\mathbf{a}, d^2) = \mathrm{CSD}(\mathbf{r}, \mathbf{M}, \mathcal{S}, 10N_0)$.

- (*Step 3.*) Set $i = (\ell - 1)b + k$ and compute

$$L(c_i) = \frac{\hat{c}_i}{N_0} \left(d^2 - d^2_{ML} \right). \tag{4.28}$$

  If $k < b$ then $k = k + 1$. Otherwise if $\ell < n$ then $\ell = \ell + 1$, $k = 1$, goto step 2. Otherwise STOP.

4.2.2 Group–wise MLM Complex Sphere Decoder

Here we propose an improved version of the I–CSD to compute the exact max–log–MAP value of the received point by trying to decrease the number of CSD applications in the second stage.

The original method calls the CSD method in order to compute only one LLR and another calling for the first stage, so the number of CSD applications is $N_{I-CSD} = nb + 1$. The new method starts again with the computation of $\mathbf{a}_{ML}$ and the related square distance d^2_{ML} and it can proceed with the second stage by observing that an ML element belonging to a complementary set of $\Lambda_i^{\hat{c}_i}$ should be ML for other complementary sets then we can evaluate more than one LLRs over a singular CSD calling. For this reason it is called *Group–wise max–log–MAP complex sphere decoder* (G–CSD). Since the proposed G–CSD provides at least one LLR for each CSD call, we have $N_{G-CSD} \leq N_{I-CSD}$ that implies $\mathrm{E}[N_{G-CSD}] \leq nb + 1$. However, in general, N_{G-CSD} depends on the lattice generator form.

Now we proceed with the detailed description of the algorithm. Let assume that $\mathcal{C}$ is the constellation set of the complex symbols $\mathbf{a}$. The G–CSD at the first stage applies the same strategy of I–CSD. It starts by applying a CSD on the search set $\mathcal{S}^{(0)} = \mathcal{C}_1^{(0)} \times \cdots \times \mathcal{C}_n^{(0)}$, where $\mathcal{C}_\ell^{(0)} = \mathcal{C}$. The CSD provides the vector of ML symbols $\hat{\mathbf{a}}$ at minimum square distance from $\mathbf{r}$, $d^2_{ML} = \|\mathbf{r} - \mathbf{M}\hat{\mathbf{a}}\|^2$, from which we can determinate the corresponding vector of bits $\hat{\mathbf{c}} = \mathrm{demap}(\hat{\mathbf{a}})$. Hence, the lattice point for bit i at the minimum distance from $\mathbf{r}$ is $\boldsymbol{\rho}_i^{\hat{c}_i} = \mathbf{M}\hat{\mathbf{a}}$.

The second stage change in the following way. We apply the CSD on the subset of $\mathcal{S}^{(0)}$ with $\hat{\mathbf{a}}$ removed, i.e. the search set is now $\mathcal{S}^{(1)} = \mathcal{C}_1^{(1)} \times \cdots \times \mathcal{C}_n^{(1)}$, where $\mathcal{C}_\ell^{(1)} = \mathcal{C}$, for $\ell = 2, \ldots, N$, and $\mathcal{C}_1^{(1)} = \mathcal{C}^{(0)} \setminus \{\hat{a}_1\}$. A significant reduction of computational time is achieved by wisely choosing the initial radius for the search. On applying the CSD on $\mathcal{S}^{(0)}$, we construct a list $\mathcal{L}$ with the explored lattice points before $\hat{\mathbf{a}}$. Then, the initial radius in the second CSD search on $\mathcal{S}^{(1)}$ is selected as the last inserted point in $\mathcal{L}$, which is the nearest point to $\mathbf{r}$ after $\hat{\mathbf{a}}$.

From the solution provided by the CSD, $\mathbf{a}^{(1)}$, and the corresponding codeword $\mathbf{c}^{(1)}$, we can determine some values $\boldsymbol{\rho}_i^{-\hat{c}_i}$ and hence some LLR values, as $\hat{\mathbf{c}}$ and $\mathbf{c}^{(1)}$ differ by at least one bit. The process is iterated by applying the CSD on the search set $\mathcal{S}^{(2)} = \mathcal{C}_1^{(2)} \times \cdots \times \mathcal{C}_n^{(2)}$, where $\mathcal{C}_\ell^{(2)} = \mathcal{C}$ for $\ell = 2, \ldots, N$, and $\mathcal{C}_1^{(2)}$ is the set of all constellation points having at least one bit whose LLR has not been computed yet. Also in this case, in order to reduce the search time, list $\mathcal{L}$ is extended with the explored points during the second search and the initial radius of the next CSD is the last inserted point. The entire process is iterated for all elements of $\hat{\mathbf{a}}$ until all LLRs have

been computed.

The step by step description of the method[2] is as follows.

- (*Input.*) Received vector $\mathbf{r}$, lattice generator $\mathbf{M}$, constellation $\mathcal{C}$ and noise variance N_0.
- (*Output.*) Metrics $L(c_i)$ for $i = 1, \ldots, nb$.
- (*Step 1.*) Set the flag bits $f_k = 0$ and the mask bits $\mu_k = 1$ for $k = 1, \ldots, b$. Set the searching set $\mathcal{S}^{(0)} = \mathcal{C}_1^{(0)} \times \ldots \times \mathcal{C}_n^{(0)}$ with $\mathcal{C}_\ell^{(0)} = \mathcal{C}$ for $\ell = 1, \ldots, n$.
- (*Step 2.*) Find the closest lattice point to $\mathbf{r}$ and its square distance $(\hat{\mathbf{a}}, d_{ML}^2, \mathcal{L}) = \mathrm{CSD}(\mathbf{r}, \mathbf{M}, \mathcal{S}^{(0)}, 10N_0)$. Compute $\hat{\mathbf{c}} = \mathrm{demap}(\hat{\mathbf{a}})$ and set $\ell = 1$, $p = 1$.
- (*Step 3.*) Set $\mathcal{C}_\ell^{(p)} = \mathcal{C}_\ell^{(p-1)} \setminus \{\hat{a}_\ell\}$, $\mathcal{C}_i^{(p)} = \mathcal{C}_i^{(p-1)}$ for $i \neq \ell$ and set the searching set $\mathcal{S}^{(p)} = \mathcal{C}_1^{(p)} \times \ldots \times \mathcal{C}_n^{(p)}$. Find $\boldsymbol{\sigma} \in \mathcal{L} \cap \mathcal{S}^{(p)}$ with minimum square distance; if $\mathcal{L} \cap \mathcal{S}^{(p)} = \varnothing$ then

  set $\sigma_\ell = \arg\min_{q \in \mathcal{C}_\ell^{(p)}} |q - \hat{a}_\ell|$ and $\sigma_i = \hat{a}_i$ for $i \neq \ell$.

  Set $d_\sigma^2 = \|\mathbf{r} - \mathbf{M}\boldsymbol{\sigma}\|^2$.
- (*Step 4.*) Find the new closest lattice point to $\mathbf{r}$ and its square distance $(\mathbf{a}^{(p)}, d^2, \mathcal{L}') = \mathrm{CSD}(\mathbf{r}, \mathbf{M}, \mathcal{S}^{(p)}, d_\sigma^2)$. Update $\mathcal{L} = \mathcal{L} \cup \mathcal{L}'$ and $\boldsymbol{\mu} = \boldsymbol{\mu} \wedge [\mathrm{demap}(a_\ell^{(p)}) \otimes \mathrm{demap}(\hat{a}_\ell)]$. For $k = 1$ to b:

  if $f_k = 0$ and $\mu_k = 0$ then

  set $i = (\ell - 1)b + k$, compute $L(c_i) = \dfrac{\hat{c}_i}{N_0}\left(d^2 - d_{ML}^2\right)$ and set $f_k = 1$.

  If $\boldsymbol{\mu} = \mathbf{0}$ then

  if $\ell < n$ then

  $\mathcal{C}_\ell^{(p)} = \mathcal{C}$, $\ell = \ell + 1$, $p = 1$, $\forall k$ $f_k = 0, \mu_k = 1$ and goto step 3

  else STOP.
- (*Step 5.*) Find the irrelevant symbols. Set $\mathcal{D} = \varnothing$, for each $q \in \mathcal{C}_\ell^{(p)}$:

  if $\boldsymbol{\mu} \wedge [\mathrm{demap}(q) \oplus \mathrm{demap}(\hat{a}_\ell)] = \mathbf{0}$ then

  $\mathcal{D} = \mathcal{D} \cup \{q\}$.

[2] Operators $\wedge$, $\otimes$ and $\oplus$ indicate the logical operators AND, XNOR and XOR, respectively, that are applied element–wise on bit arrays.

Find $\boldsymbol{\sigma} \in \mathcal{L} \cap \mathcal{S}$ with minimum square distance; if $\mathcal{L} \cap \mathcal{S}^{(p)} = \varnothing$ then

set $\sigma_\ell = \underset{q \in \mathcal{C}_\ell^{(p)}}{\arg\min} |q - \hat{a}_\ell|$ and $\sigma_i = \hat{a}_i$ for $i \neq \ell$.

Set $\mathcal{C}_\ell^{(p+1)} = \mathcal{C}_\ell^{(p)} \setminus \mathcal{D}$, $p = p + 1$ and $d_\sigma^2 = \|\mathbf{r} - \mathbf{M}\boldsymbol{\sigma}\|^2$ then goto step 4.

4.3 Performance Comparison

The introduction and the presentation of this new method for soft detection in STBC scenarios requires a motivation in terms of performance improvement. A detailed comparison between the two presented method I–CSD and G–CSD is presented in this section. Furthermore it follows a second part which shows how is the effects we get by combining LDPC coded information with STBCs in MIMO scenarios. Soft detection is provided by the methods presented so far.

4.3.1 Methods Comparison

We recall the two methods I–CSD and G–CSD, which achieve soft detection for STBCs, perform an exhaustive search in the lattice domain. There are many other methods to achieve soft detection we have not considered. Any of them are suboptimal as [39] which narrows the search in a spherical list centered in the ML point, or [40] which is optimal just for the case of two transmitter antennas and suboptimal for more than two. Another one [41] is optimal but it is specific for the Golden Code.

The first difference we want to point out is the number of CSD calls for the two methods. In our simulation we have choose a 4 dimension system $\mathbf{M} \in \mathbb{C}^{4\times4}$ with i.i.d. entries $M_{ij} \in \mathcal{CN}(0, \sigma^2)$. Each complex symbol a_j belongs to a 64–QAM constellation and the signal to noise ratio is fixed to $SNR = 0$. In Fig. 4.5 we plot the average number of CSD calls versus the variance σ^2 of the lattice matrix entries.

The number of CSD calls depends on the lattice form, i.e. the matrix $\mathbf{M}$. The improvement increase with the growing of σ^2 which means that the more stretched is lattice the higher is the probability to compute more than one LLR after a singular CSD calling. The experimental results also remark what we anticipate above about $N_{G-CSD} \leq N_{I-CSD}$ then $\mathrm{E}[N_{G-CSD}] \leq \mathrm{E}[N_{I-CSD}] = nb + 1$.

In a more specific scenario we consider a computational time comparison for three STBCs: the 2×2 Alamouti presented in Section 3.1.1, the Golden

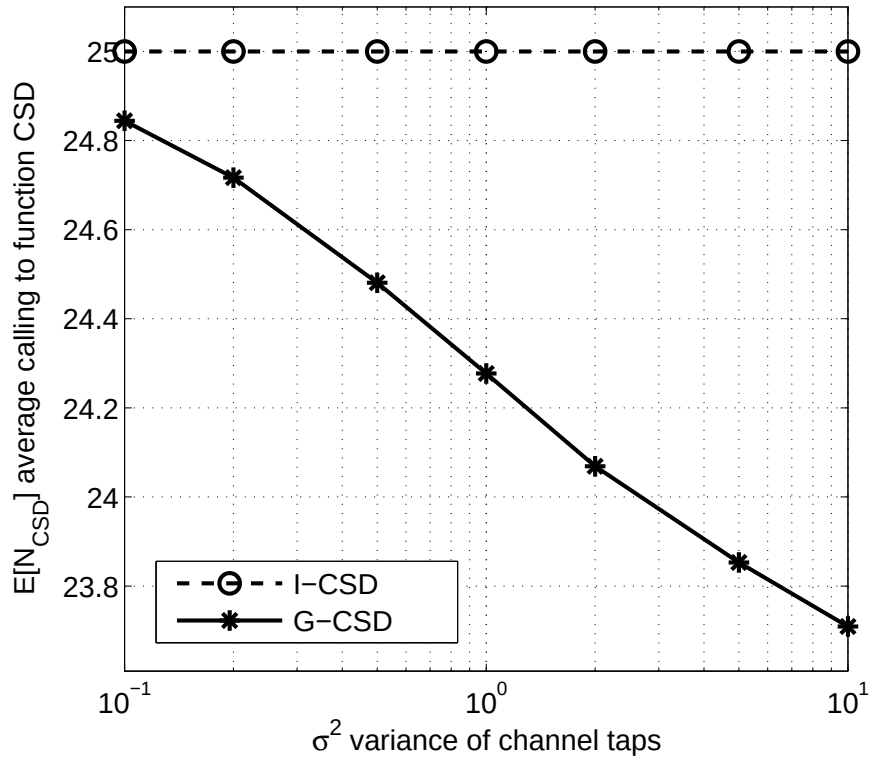

Figure 4.5: MLM sphere decoder performances comparison.

Code discussed in Section 3.1.2, and the Tilted QAM briefly exposed in Section 3.1.3.

The lattice generator matrix $\mathbf{M}$ for the three examined linear STBC is reported below. We use the compact notation h_{ij} and $\tilde{h}_{ij}$ to indicate the two consecutive channel matrix entries $h_{i,j}(2t)$ and $h_{i,j}(2t+1)$, respectively.

Golden code

$$\mathbf{M}_G = \frac{1}{\sqrt{5}} \begin{bmatrix} h_{11}\alpha & h_{11}\alpha\theta & h_{12}j\bar{\alpha} & h_{12}j\bar{\alpha}\bar{\theta} \\ h_{21}\alpha & h_{21}\alpha\theta & h_{22}j\bar{\alpha} & h_{22}j\bar{\alpha}\bar{\theta} \\ \tilde{h}_{12}\bar{\alpha} & \tilde{h}_{12}\bar{\alpha}\bar{\theta} & \tilde{h}_{11}\alpha & \tilde{h}_{11}\alpha\theta \\ \tilde{h}_{22}\bar{\alpha} & \tilde{h}_{22}\bar{\alpha}\bar{\theta} & \tilde{h}_{21}\alpha & \tilde{h}_{21}\alpha\theta \end{bmatrix}. \tag{4.29}$$

Tilted QAM

$$\mathbf{M}_T = \begin{bmatrix} h_{11}\cos\gamma_1 & h_{11}\sin\gamma_1 & h_{12}\cos\gamma_2 & h_{12}\sin\gamma_2 \\ h_{21}\cos\gamma_1 & h_{21}\sin\gamma_1 & h_{22}\cos\gamma_2 & h_{22}\sin\gamma_2 \\ \tilde{h}_{12}\sin\gamma_1 & \tilde{h}_{12}\cos\gamma_1 & \tilde{h}_{11}\sin\gamma_2 & \tilde{h}_{11}\cos\gamma_2 \\ \tilde{h}_{22}\sin\gamma_1 & \tilde{h}_{22}\cos\gamma_1 & \tilde{h}_{21}\sin\gamma_2 & \tilde{h}_{21}\cos\gamma_2 \end{bmatrix}. \tag{4.30}$$

Alamouti

$$\mathbf{M}_A = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ \tilde{h}_{12}^* & -\tilde{h}_{11}^* \\ \tilde{h}_{22}^* & -\tilde{h}_{21}^* \end{bmatrix}. \tag{4.31}$$

Table 4.1 shows the average computational time in our simulation environment for the considered techniques. Values of the average signal to noise ratio (SNR) per received antenna ρ ensure a bit error rate (BER) lower than 10^{-6}.

STBC and constellation	ρ [dB]	Average elapsed time [s] G–CSD	I–CSD
Golden 4–QAM	7.75	5.3×10^{-4}	4.9×10^{-4}
Tilted 4–QAM	7.80	5.7×10^{-4}	4.3×10^{-4}
Alamouti 16–QAM	8.30	4.1×10^{-4}	3.9×10^{-4}
Golden 16–QAM	15.35	2.0×10^{-3}	2.9×10^{-3}
Tilted 16–QAM	15.20	2.0×10^{-3}	2.9×10^{-3}
Alamouti 256–QAM	18.45	5.7×10^{-3}	6.8×10^{-3}
Golden 64–QAM	21.45	1.2×10^{-2}	2.0×10^{-2}
Tilted 64–QAM	21.25	1.2×10^{-2}	2.1×10^{-2}

Table 4.1: Performance comparison

We observe that G–CSD halves the computational time (CT) with respect to I–CSD for 64–QAM. For 16–QAM the CT reduction is of about 30%. For a small constellation size 4–QAM the CT of G–CSD is close or slightly higher than that for I–CSD. As higher constellation sizes provides a higher bit rate and hence are more demanding for the chip design, we conclude that G–CSD is highly valuable for practical implementation. We conclude that G–CSD is a valuable alternative to I–CSD for systems with higher bit–rates.

4.3.2 Performance on STBCs

The four STBCs presented in Section 3.1 have well known behavior in uncoded scenario. After a review of their performance we dedicate the last part of this section to analyze the combined performance of LDPC coded scenario with STBCs.

The system model we used in order to evaluate the performance on STBCs both in uncoded and coded scenarios is the following. We consider a communication system where a codeword of nb bits $\mathbf{c} = [c_1, \ldots, c_{nb}]$, with $c_i \in \{-1, +1\}$, is mapped into a vector of n complex symbols $\mathbf{a} = [a_1, \ldots, a_n]$

belonging to a 2^b–ary constellation $\mathcal{C}$. Typically Gray mapping is used such that binary codewords with unitary Hamming distance correspond to symbols at minimum Euclidean distance. For the sake of clarity, the relationship between $\mathbf{c}$ and $\mathbf{a}$ is denoted as $\mathbf{a} = \text{map}(\mathbf{c})$, $\mathbf{c} = \text{demap}(\mathbf{a})$. The transmitted symbol by the i–th antenna at time instant t is related to symbols in $\mathbf{a}$, and is denoted by $s_i(t)$. In this letter we focus on 2×2 MIMO configurations, due to their simplicity, with STBCs having a transmitted *code matrix* given by

$$\mathbf{S}(\mathbf{a}) = \begin{bmatrix} s_1(2t) & s_1(2t+1) \\ s_2(2t) & s_2(2t+1) \end{bmatrix}. \tag{4.32}$$

Moreover, we assume a flat fading channel hence the received vector at time t is given by

$$\begin{bmatrix} r_1(t) \\ r_2(t) \end{bmatrix} = \begin{bmatrix} h_{1,1}(t) & h_{1,2}(t) \\ h_{2,1}(t) & h_{2,2}(t) \end{bmatrix} \begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix} + \begin{bmatrix} w_1(t) \\ w_2(t) \end{bmatrix}, \tag{4.33}$$

where $w_i(t) \in \mathcal{CN}(0, N_0)$, i.i.d., are random variables representing AWGN and $h_{i,j}(t) \in \mathcal{CN}(0, 1)$, i.i.d., is the channel gain from antenna j to antenna i.

In the case where a STBC represents a linear mapping with respect to $\mathbf{a}$, the relationship (4.33) can be rewritten as

$$\mathbf{r} = \mathbf{M}\mathbf{a} + \mathbf{w}, \tag{4.34}$$

where

$$\mathbf{r} = [r_1(2t), r_2(2t), r_1'(2t+1), r_2'(2t+1)]^{\mathrm{T}}, \tag{4.35}$$

$$\mathbf{w} = [w_1(2t), w_2(2t), w_1'(2t+1), w_2'(2t+1)]^{\mathrm{T}}. \tag{4.36}$$

r_i', w_i' stand for either r_i, w_i or r_i^*, w_i^* in accordance with the considered STBC. The matrix $\mathbf{M}$ is a $4 \times n$ complex matrix, denoted *lattice generator matrix* as we have seen in (4.29),(4.30), and (4.31) for the three linear STBCs we have introduced.

Hard Detection We first consider an uncoded scenario where, in order to ensure the same information rate for all systems, we consider QPSK for the full–rate Golden code and tilted QAM and 16–QAM constellation for both Alamouti schemes. Then we have a codeword $\mathbf{c}$ is a group of 8–bit per unit of STBC. In all cases bit mapping is by Gray coding. Detection is performed by CSD with hard detection for linear STBCs. Since the rotated and scaled Alamouti is not linear, it cannot be decoded by using CSD. For ML hard detection it requires either an exhaustive search or a more elaborate detection method [21].

For the rotated and scaled Alamouti arises a problem in decoding because it is not a linear STBC and Fig. 4.6 compares the BER versus ρ, for the four STBCs.

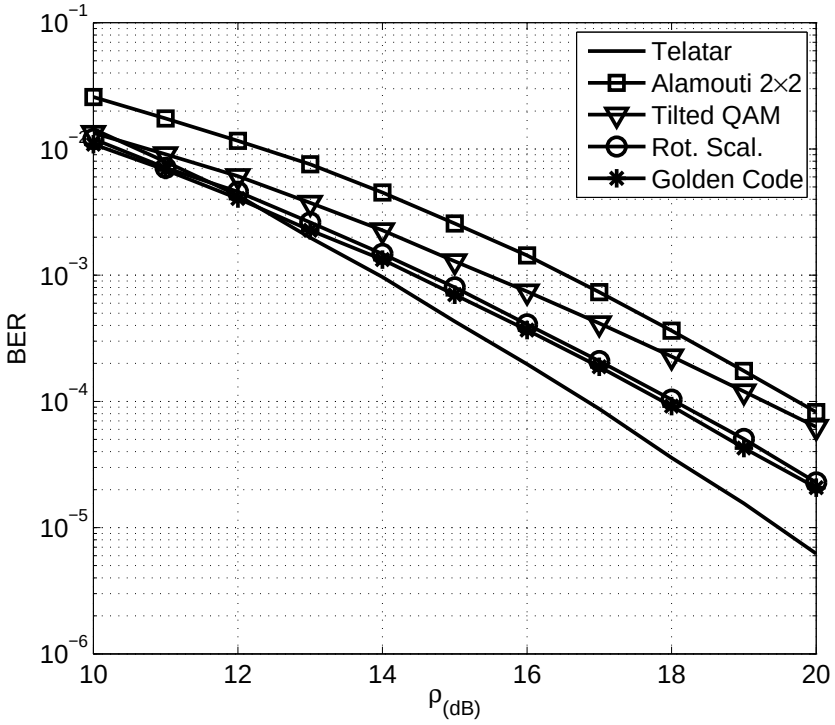

Figure 4.6: Bit error rate in the presence of hard detection for uncoded space–time codes.

This figure also contains a curve (unbroken line) which shows the Telatar capacity. It is computed according to [42]

$$C(\mathbf{H}) = \log_2 \det \left(\mathbf{I} + \frac{\rho}{N_0} \mathbf{H}\mathbf{H}^H \right) \text{ bit/s/Hz.} \tag{4.37}$$

We observe that in the uncoded scenario, the Golden code provides the lowest BER, outperforming the rotated and scaled Alamouti by about 0.2 dB. These results are in accordance with the literature [21].

Soft Detection We are interested on the performance of these STBCs in coded scenario using soft detection. In the coded scenario information bits are encoded by the low density parity check code (LDPC) adopted in the DVB–S2 standard [43]. The specification of this coding are

- codeword length = 64,800 bit/block (normal frame);
- code rate = 3/4;

We assume an independent channel matrix (H) for each STBC block. At the receiver, LLRs are computed by the G–CSD for linear STBCs and by the exhaustive method for the rotated and scaled Alamouti. LLR values are then passed to the LDPC decoder. The initial square radius for both I–CSD and G–CSD is set to $10N_0$ in order to ensure a probability to find a point inside the first sphere greater than 0.99, in agreement with [35].

Fig. 4.7 shows a comparison of four STBCs for three information rates.

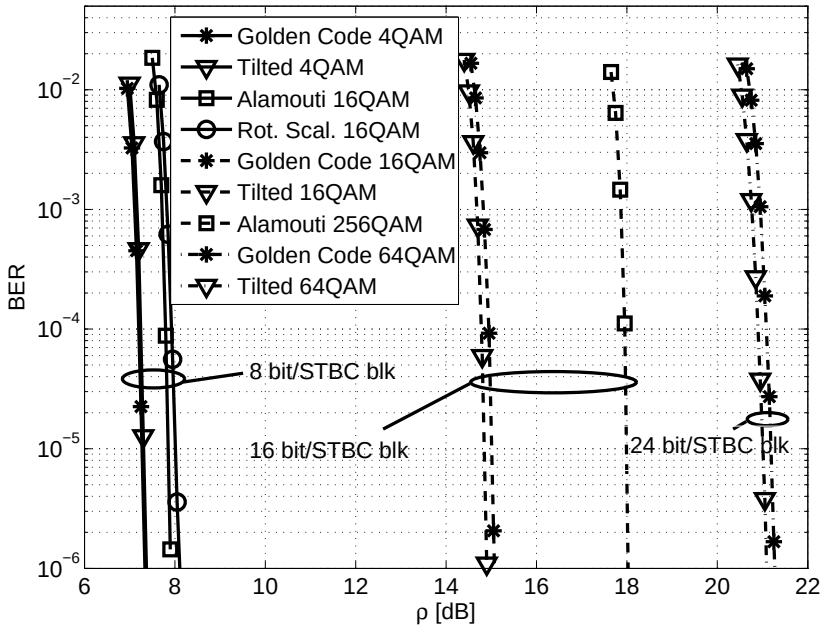

Figure 4.7: Bit error rate in the presence of soft detection of various space–time codes and LDPC decoding.

For **c** with 8 bit/codeword we compare the full–rate STBCs using 4–QAM and the Alamouti codes using 16–QAM. In this case the tilted QAM yields similar performance of the Golden code with a significant improvement over Alamouti codes. The 16 bit/codeword corresponds to 16–QAM for full–rate codes and 256–QAM for Alamouti codes. The rotated and scaled Alamouti code is not considered, since exhaustive detection becomes too complex. In this other case we observe that full–rate STBCs outperform Alamouti by

3 dB. Moreover, we note that although tilded QAM employs a simple real rotation matrix it outperforms the Golden code.

For an information rate of 24 bit/codeword, the Alamouti STBC is not considered because it would require a 4,096–QAM constellation. Also in this case we note that tilted QAM outperforms the Golden code.

In conclusion, for a given BER, the gain in signal to noise ratio of the Golden Code with other full–rate schemes, observed in the presence of hard detection, was not observed in the presence of soft detection using LDPC codes. Moreover, the gain of full–rate codes with Alamouti codes increases with the information rate.

4.4 Pseudo–Codes

We conclude this section with the pseudo–code for the main algorithms introduced here: the sphere decoder, the complex sphere decoder, the iterative and the group–wise max–log–MAP CSDs.

4.4.1 Sphere Decoder

Table 4.2 shows the pseudo–code of the SD.

```
u_d = SPHEREDECODER(z, R, L)
d = MAX_VAL;
k = n;
x[k] = 0;
T[k] = 0;
A[k] = MAX(CEIL((z[k] - x[k] -
           SQRT(d - T[k]))/R[k,k]), 0);
B[k] = MIN(FLOOR((z[k] - x[k] +
            SQRT(d - T[k]))/R[k,k]), L - 1);
u[k] = A[k] - 1;
WHILE (true)
   u[k] = u[k] + 1;
   IF (u[k] > B[k])
      IF (k = n)
         BREAK;
      ELSE
         k = k + 1;
      END IF
   ELSE IF (k > 0)
```

```
      x[k-1] = 0;
      FOR i = k TO n
         x[k-1] = x[k - 1] + R[k - 1,i] * u[i];
      END FOR
      T[k-1] = T[k] + (z[k] - x[k] - R[k,k] * u[k])^2;
      k = k - 1;
      IF (d < T[k])
         IF (k = m )
            BREAK;
         ELSE
            k = k + 1;
         END IF
      ELSE
         A[k] = MAX(CEIL((z[k] - x[k] -
                     SQRT(d - T[k]))/R[k,k]), 0);
         B[k] = MIN(FLOOR((z[k] - x[k] +
                     SQRT(d - T[k]))/R[k,k]), L - 1);
         u[k] = A[k] - 1;
      END IF
   ELSE
      d_new = T[1] + (z[1] - x[1] - R[1,1] * u[1])^2;
      IF (d_new < d)
         d = d_new;
         FOR i = 1 TO n
            u_d[i] = x[i];
            B[i] = MIN(FLOOR((z[i] - x[i] +
                        SQRT(d - T[i]))/R[i,i]), L - 1);
         END FOR
      END IF
   END IF
END WHILE
```

Table 4.2: The Sphere decoder algorithm.

In this pseudo–code the output `u_d` plays the rule of $\hat{\mathbf{u}}$, instead the inputs `z`, `R`, `L`, are, respectively, the vector $\mathbf{z}$, the upper triangular matrix $\mathbf{R}$ and the cardinality of the alphabet. The auxiliary variables `x[k]` and `T[k]`corresponds to ξ_k and T_k, respectively. The constant `MAX_VAL` indicates the greatest value represented in the computer, corresponding to $+\infty$. The function `CEIL(x)` computes the least integer greater than `x`, conversely

`FLOOR(x)` returns the greatest integer less than `x`; `SQRT(x)` computes the square root.

4.4.2 Complex SD

Table 4.3 is the pseudo–code for complex SD.

```
(a_ML, d2) = CSD(y, F, N0, L)
L1:
  N_T = SIZE(F);
  C = 10 * N0;
  PI = 3.1416;
  FOR k = 1 TO N_T
     sN[k] = 0;
     a[k] = 0;
  END FOR
  flag = false;
  flagfound = false;
L2:
  k = 0;
  FOR i = 1 TO N_T
     z[i] = y[i];
  END FOR
  xsi = 0;
  IF flag
     C = C * 1.2;
  END IF
L3:
  k = k + 1;
  lambda[k] = z[k] / F[k,k];
  FOR i = 1 TO L[k].Nrho
     eta = (L[k].rho[i]^2 + ABS(lambda[k])^2 - C / F[k,k]^2)
                        / (2 * L[k].rho[i] * ABS(lambda[k]));
     IF eta <= -1
        FOR l = 1 to L[k].rho[i].Nphi
           sN[k] = sN[k] + 1;
           S[k,sN[k]] = L[k].rho[i] *
                        EXP(j * L[k].rho[i].phi[l]);
        END FOR
     ELSE IF eta <= 1
        theta = PHASE(lambda[k]);
```

```
         aux = ACOS(eta);
         low = theta - aux;
         upp = theta + aux;
         IF upp > PI
            upp = upp - 2 * PI;
            FOR l = 1 TO L[k].rho[i].Nphi
               IF L[k].rho[i].phi[l] <= upp OR
                      L[k].rho[i].phi[l] >= low
                  sN[k] = sN[k] + 1;
                  S[k,sN[k]] = L[k].rho[i] *
                               EXP(j * L[k].rho[i].phi[l]);
               END IF
            END FOR
         ELSE IF  low < -PI
            low = low + * PI;
            FOR l = 1 TO L[k].rho[i].Nphi
               IF L[k].rho[i].phi[l] <= upp OR
                      L[k].rho[i].phi[l] >= low
                  sN[k] = sN[k] + 1;
                  S[k,sN[k]] = L[k].rho[i] *
                               EXP(j * L[k].rho[i].phi[l]);
               END IF
            END FOR
         ELSE
            FOR l = 1 TO L[k].rho[i].Nphi
               IF L[k].rho[i].phi[l] <= upp AND
                       L[k].rho[i].phi[l] >= low
                  sN[k] = sN[k] + 1;
                  S[k,sN[k]] = L[k].rho[i] *
                               EXP(j * L[k].rho[i].phi[l]);
               END IF
            END FOR
         END IF
      END IF
   END FOR
L4:
   IF S[k].N > 0
      GOTO L7;
   ELSE
      GOTO L5;
   END IF
```

```
L5:
  FOR i = N_T DOWNTO 1
     IF S[i].N > 1
        level = i;
        GOTO L13;
     END IF
  END FOR
  GOTO L6;
L6:
  IF NOT flagfound
     flag = true;
     GOTO L2;
  ELSE
     RETURN;
  END IF

L7:
  delta[k,1] = ABS(lambda[k] - S[k,1]);
  IF nS[k] > 1
     FOR i = 2 TO nS[k]
        delta[k,i] = ABS(lambda[k] - S[k,i]);
        FOR l = i DOWNTO 1
           IF delta[k,l] > delta[k,l - 1]
              aux = S[k,l - 1];
              S[k,l - 1] = S[k,l];
              S[k,l] = aux;
              aux = delta[k,l - 1];
              delta[k,l - 1] = delta[k,l];
              delta[k,l] = aux;
           ELSE
              BREAK;
           END IF
        END FOR
        FOR i = 1 TO N_T
           z_act[k,i] = z[i];
           a_act[k,i] = a[i];
        END FOR
        xsi_act = xsi;
     END FOR
     nS[k] = nS[k] - 1;
     delta_best = delta[k,nS[k]];
```

```
L8:
  a[k] = S[k,nS[k]];
L9:
  xsi = xsi + F[k,k]^2 * delta_best[k];
  IF xsi < C
     GOTO L11;
  ELSE
     GOTO L10;
  END IF;
L10:
  FOR i = N_T DOWNTO 1
     IF nS[i] > 0
        level = i;
        GOTO L13;
     END IF
  END FOR
  GOTO L6;
L11:
  IF k < N_T
     FOR i = 1 TO (k + 1)
        z[k + 1] = z[k + 1] - F[k + 1,i] * a[i];
     END FOR
     GOTO L3;
  END IF
L12:
  IF k = N_T
     FOR i = 1 TO N_T
        a_ML[i] = a[i];
     END FOR
     C = xsi;
     d2 = C;
     flagfound = true;
  END IF
  FOR i = N_T DOWNTO 1
     IF nS[i] > 0
        level = i;
        GOTO L13;
     END IF
  END FOR
  RETURN;
L13:
```

```
nS[level] = nS[level] - 1;
FOR i= 1 TO N_T
   z[i] = z_act[level,i];
   a[i] = a_act[level,i];
END FOR
xsi = xsi_act[level];
delta_best[level] = delta[level,nS[level]];
a[level] = S[level,nS[level]];
k = level;
GOTO L9;
```

Table 4.3: The complex sphere decoder algorithm.

Here the input `y`, `F`, and `N0` represent $\tilde{\mathbf{y}}$, $\mathbf{F}$, and N_0 of equation (4.22). A special description is necessary for the input `L`. It is a sort of data structure which contains the N_T constellations $\mathcal{C}_1, \ldots, \mathcal{C}_{N_T}$ which each symbol form a transmitted antenna belongs to. They are stored in a particular structure that fits for this algorithm: the k–th constellation can be described by a number of rings denoted by `L[k].Nrho` and the radius for the i–th ring is stored in `L[k].rho[i]`. In such ring `L[k].rho[i].Nphi` constellation points lie. The phase value ℓ–th point in such ring is stored in `L[k].rho[i].phi[l]`. In Fig. 4.1 we have an example of 16–QAM constellation characterized by `L[k].Nrho`$= 3$ rings, the number of points per rings are `L[k].rho[1].Nphi`=`L[k].rho[3].Nphi`$= 4$ for the first and the third ring and it is `L[k].rho[2].Nphi`$= 8$ for the second. The value of the phase for the first point is `L[k].rho[1].phi[1]`$= \pi/4$.

The output `a_ML` represent the point $\mathbf{a}_{ML}$ in the lattice which is at minimum square distance $\mathbf{a}_{ML} = \arg\min_{\mathbf{a}} \|\tilde{\mathbf{y}} - \mathbf{F}\mathbf{a}\|^2$ then `d2` is such distance $d^2 = \|\tilde{\mathbf{y}} - \mathbf{F}\mathbf{a}_{ML}\|^2$.

The function `SIZE(M)` applied to a $n \times n$ square matrix `M`returns the dimension n of the matrix. `ACOS(x)` returns the arc–cosine in a range between 0 and π of the value `x` which has to belong to the interval $[-1, +1]$. We remark that here we are using complex variables, not directly supported in most of program languages, then operators and functions act in complex environment. Here we have `ABS(z)` and `PHASE(z)` which return amplitude and phase in $[-\pi, +\pi]$ of the complex number `z`. `EXP(z)` calculate the exponential of `z`.

4.4.3 I–CSD

Now we present the pseudo–code for the two algorithms for soft detection in STBC. First we have the iterative max–log–MAP CSD in Table 4.4.

```
(LLR) = ICSD(y, F, N0, C)
N_T = SIZE(F);
C_S = SIZE(C)
n_bit = LOG2(C_S);
FOR l = 1 TO N_T
   L[l].STORE(C);
END FOR
(a_ML, d2) = CSD(y, F, N0, L);
FOR l = 1 to N_T
   IF l > 1
      L[l - 1].STORE(C);
   END IF
   bits_ML[] = DEMAP(C, a_ML[l]);
   FOR k = 1 TO n_bit
      n = 1;
      FOR m = 1 TO C_S
         bits[] = DEMAP(C, C[m]);
         IF bits[k] != bits_ML[k]
            C_tmp[n] = C[m];
            n = n + 1;
         END IF
      END FOR
      L[l].STORE(C_tmp);
      (a, d) = CSD(y, F, N0, L);
      LLR[l,k] = (bits_ML[k] / N0) * (d - d2);
   END FOR
END FOR
```

Table 4.4: The iterative max–log–MAP complex sphere decoder algorithm (I–CSD).

Once again the input `y`, `F`, and `N0` are $\tilde{\mathbf{y}}$, $\mathbf{F}$, and N_0 of equation (4.22). `C` is an array containing all the complex values for constellation points.

With `SIZE(C)` we indicate the number of elements in the array `C`. The function `LOG2(x)` computes the logarithm of `x` in base 2. In order to store

several constellations in a way that fits well for CSD algorithm we recur to the same data structure `L` we introduce previously. Now we introduce the method `L[k].STORE(C)` which stores the constellation complex points, contained in the array `C`, in the way we explained above. Hence we are allowed to call the method `CSD()` to our purpose. The method `bits[] = DEMAP(C,x)` returns the sequence of bits in the array `bits[]` belonging to the set $\{+1, -1\}$. This sequence is the one is mapped by the complex value in `C` which is at minimum Euclidean distance from the complex input `x`.

The output $L(c_i)$ are computed according the equation (4.28) and stored in the matrix `LLR[k,i]`.

4.4.4 G–CSD

In this last section we present the pseudo–code for the brand new group–wise max–log–MAP CSD in Table 4.5.

```
(LLR) = GCSD(y, F, N0, C)
N_T = SIZE(F);
C_S = SIZE(C)
n_bit = LOG2(C_S);
FOR k = 1 TO N_T
   L[k].STORE(C);
END FOR
(list_a, list_d2) = CSD(y, F, N0, L);
n_list = SIZE(list_a);
a_ML = list_a[n_list];
d2 = list_d2[n_list];
FOR l = 1 TO N_T
   IF l > 1
      L[l - 1].STORE(C);
   END IF
   bits_ML[] = DEMAP(C, a_ML[l]);
   FOR i = 1 TO n_bit
      f[i] = 0;
      msk[i] = 1;
   END FOR
   FOR i = 1 TO C_S
      n = 1;
      IF C[i] != a_ML[l]
         C_tmp[n] = C[i];
         n = n + 1;
```

```
      END IF
   END FOR
   start_d = 0;
   IF n_list > 1
      FOR i = n_list DOWNTO 1
         IF a_ML[l] != list_a[i,l]
            FOR q = 1 TO N_T
               start_a[q] = list_a[i,q];
            END FOR
            start_d = list_d2[i];
            BREAK;
         END IF
      END FOR
   END IF
   IF start_d = 0
      FOR i = 1 TO N_T
         start_a[i] = a_ML[i];
      END FOR
      start_a[l] = C_tmp[1];
      FOR i = 2 TO n
         IF ABS(start_a[l] - a_ML[l]) >
                  ABS(C_tmp[i] - a_ML[l])
            start_a[l] = C_tmp[i];
         END IF
      END FOR
      FOR i1 = 1 TO N_T
         d_part = y(i1);
         FOR i2 = 1 TO i1
            d_part = d_part - F[i1,i2] * start_a[i2];
         END FOR
         start_d = start_d + ABS(d_aprt)^2;
      END FOR
   END IF
   WHILE SUM(msk) > 0
      L[l].STORE(C_tmp)
      (list_a_1, list_d2_1) =
                  CSD(y, F, N0, L, start_a, start_d);
      n_list_1 = SIZE(list_a_1);
      a_1 = list_a[n_list_1];
      d2_1 = list_d2[n_list_1];
      list_a = UNION(list_a,list_a_1);
```

```
list_d2 = UNION(list_d2,list_d2_1);
(list_d2, ord) = SORT(list_d2);
n_list = SIZE(list_a);
bits[] = DEMAP(C, a_1[l]);
FOR i = 1 TO n_bit
   val = (bits_ML[i] * bits[i] + 1) / 2;
   msk[i] = msk[i] * val;
   IF (f[i] = 0) AND (msk[i] = 0)
      LLR[l,i] = (bits_ML[k] / N0) * (d2_1 - d2);
      f[i] = 1;
   END IF
END FOR
IF SUM(msk) > 0
   m = 1;
   FOR k = 1 TO n
      bits[] = DEMAP(C, C_tmp[k]);
      val = 0;
      FOR i = 1 TO n_bit
         tmp = (bits[i] * bits_ML[i] + 1) / 2;
         val = val + msk[i] * tmp;
      END FOR
      IF val > 0
         C_tmp_1[m] = C_tmp[k];
         m = m + 1;
      END IF
   END FOR
   FOR k = 1 TO m
      C_tmp[k] = C_tmp_1[k];
   END FOR
   n = m;
END IF
start_d = 0;
FOR i = 1 TO n_list
   FOR k = 1 TO n
      IF C_tmp[k] = list_a[ord[i],l]
         FOR q = 1 TO N_T
            start_a[q] = list_a[ord[i],q];
         END FOR
         start_d = list_d2[i];
         BREAK;
      END IF
```

```
         END FOR
      END FOR
      IF start_d = 0
         FOR i = 1 TO N_T
            start_a[i] = a_ML[i];
         END FOR
         start_a[l] = C_tmp[1];
         FOR i = 2 TO n
            IF ABS(start_a[l] - a_ML[l]) >
                    ABS(C_tmp[i] - a_ML[l])
               start_a[l] = C_tmp[i];
            END IF
         END FOR
         FOR i1 = 1 TO N_T
            d_part = y(i1);
            FOR i2 = 1 TO i1
               d_part = d_part - F[i1,i2] * start_a[i2];
            END FOR
            start_d = start_d + ABS(d_aprt)^2;
         END FOR
      END IF
   END WHILE
END FOR
```

Table 4.5: The group–wise max–log–MAP complex sphere decoder algorithm (G–CSD).

The input and output of this algorithm has the same format of the previous I–CSD.

What is changed here is the CSD method we are calling. This one does return not only the lattice point at minimum distance $\mathbf{a}_M L$ but all the points touched by the algorithm during its search which are stored in the array `list_a` then the related square Euclidean distances are stored in `list_d2`. When the method is called with two more arguments like this `CSD(y,F,N0,L,start_a,start_d)`, it means that sphere decoder starts its search from the point indicated in `start_a` in a sphere of square radius `start_d`. The method `UNION(set_A,set_B)` returns an array which contains the elements of both arrays `set_A` and `set_B` taken once, according to the union operator $\cup$ of the set theory. The method `(x_sort,positions)=SORT(x)`

applied to the array `x` returns all its elements in the array `x_sort` in a non–increasing order. `position` stores all the new positions of the elements in `x` in the array `x_sort`. This method is not standard because of the two variables returned, but it is common in very high level programming language such as MatLab.

Chapter 5

Modulation Diversity Schemes

The next generation of digital terrestrial television (DVB–T2) standard required to improve performance of the current DVB–T standard. The call for technologies for DVB–T2 standard [1] has fostered the investigation of solutions to improve the performance of the existing DVB–T standard regarding new modulation, equalization and channel estimation techniques, only to mention physical layer issues.

Orthogonal frequency division multiplexing (OFDM) with high number of subcarriers has been adopted as suggested by commercial and technical purposes. As multiple–input multiple–output (MIMO) antennas techniques are not backward compatible with current constellations, new modulation methods suitable for OFDM in the presence of Rayleigh fading channels could be developed in order to provide diversity gain without spectral or power inefficiencies.

Since one early application of any DVB–T2 specification would be for multi–channel high definition television (HDTV) broadcasting [1], an effort is necessary to improve significantly the spectral efficiency of transmission. To this end, modulations should evolve from current QAM to more elaborated modulation/coding schemes.

One option is to use coded modulation, e.g. trellis coded QAM [44] or bit interleaved (BICM) scheme [45], which provides code diversity at the expense of a quite high computational complexity. A second option is given by suitable modulation diversity techniques [46–48] that provide diversity with no spectral or power inefficiencies. All these techniques are well suited for a communication system using OFDM, that translates a frequency dispersive channel into a set of parallel flat fading channels with no inter symbol interference. Already included in the DVB–T standard, OFDM is also adopted in next generation DVB–T2 with longer block size to reduce the spectral inefficiency caused by the cyclic prefix. When modulation diversity is applied

to OFDM, a given bit is transmitted on multiple carriers, thus capturing frequency diversity.

With regards to modulation diversity techniques, in this chapter we compare three modulation methods, even in a coded scenario, namely,

1. re–mapped repetition,

2. rotational multi–carrier,

3. multidimensional rotated QAM.

The RR modulation [49] can be efficiently implemented by two pulse amplitude modulators and an interleaver. The R–MCM [50] requires instead the rotation of real symbol vectors by a simple rotation matrix and an interleaver. In MR–QAM [47] the rotation matrix is chosen according to algebraic number theory. In the literature the performance of these modulation schemes has been studied and evaluated in an uncoded scenario, i.e. by hard detection of the demodulated signal. Moreover, the DVB–T2 standard includes powerful coding schemes. However their interaction with soft detection and soft input decoding has rarely been considered, e.g. for turbo BICM [51] and DVB–S2 LDPC only for the code rate 5/6 [49]. We consider DVB–T2 low–density parity–check (LDPC) block codes, which are the same used in the digital satellite television standard DVB–S2 [43], and we evaluate the impact of coding on the above modulation techniques. In particular, when coding is considered, soft detection must be applied at the receiver in order to obtain the log likelihood ratio (LLR) relative to each bit. The soft information, forwarded to the LDPC decoder, is computed by the G–CSD method presented in Section 4.2.2. With respect to an uncoded scenario, we will see that soft detection and coding yield an interesting and quite different performance comparison between the considered modulation techniques. In literature there are further soft schemes on mapping bits to modulation symbols [52, 53] but they are not considered here.

5.1 Mapping Methods

We describe here the three mapping methods that will be compared next. The three mapping methods and the system model in which we are going simulate them are briefly introduced. This analysis is achieved in a classical wireless communication system model. The model we treat is very simple. An information bit stream $\{b_\ell\}$ is encoded into the codebit stream $\{c_m\}$, by a low–density parity–check (LDPC) block code [54]. The codebits are then

mapped into the complex symbol sequence $\{a_k\}$. We remark that the mapping is not restricted to usual QAM but comprises more elaborate methods, which are described in the following. In turn, sequence $\{a_k\}$ is OFDM modulated to generate the signal $\{s_k\}$, which is interpolated and transmitted on the wireless channel. Fig. 5.1 shows the transmitter scheme.

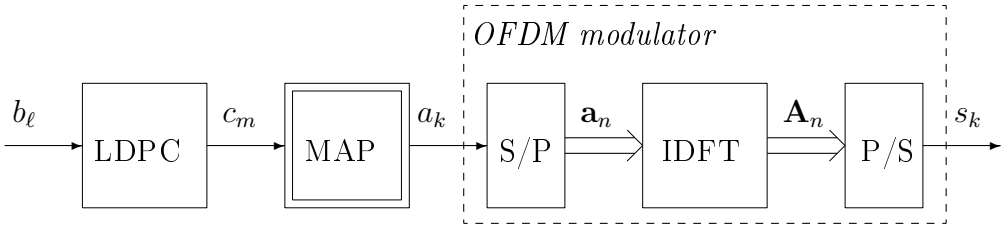

Figure 5.1: The transmitter scheme.

At the receiver, the signal is sampled. The relation between $\{s_k\}$ and the sampled received signal can be modeled as an equivalent discrete–time time–invariant (at least for the duration of one OFDM symbol) multipath fading channel of duration N_c with impulse response $\{h_i\}$, $i = 0, 1, \ldots, N_c - 1$. The received signal can be written as

$$r_k = \sum_{i=0}^{N_c-1} h_i s_{k-i} + w_k \,, \tag{5.1}$$

where $w_k \in \mathcal{CN}(0, N_0)$, i.i.d., represents the noise contribution.

The receiver structure, that extracts from $\{r_k\}$ the soft information (i.e. LLR) associated to codebits $\{c_m\}$, depends on the mapping method and will be outlined in the next section.

5.1.1 Re–mapped Repetition

The RR technique has been proposed in [49] to improve the performance of OFDM in an environment with severe frequency selectivity by exploiting the fact that a simultaneous deep fade on different subcarriers is unlikely. The basic idea of RR is of transmitting the same bit on two different OFDM subcarriers by a suitable remapping.

In particular, from the block of M consecutive codebits

$$\mathbf{c}_k = [c_{kM}, c_{kM+1}, \ldots, c_{kM+M-1}]^{\mathrm{T}} \tag{5.2}$$

we first obtain symbol d_k, taken from a real 2^M–size constellation, e.g. pulse amplitude modulation (PAM). The same set of bits $\mathbf{c}_k$ is mapped into a

different PAM symbol d'_k, by a different mapping which however depends only on d_k. The two PAM symbols d_k and d'_k form, respectively, the real and imaginary part of two symbols transmitted on two different subcarriers f_1 and f_2. Note that in this way the two symbols are affected by different fading. As shown in Fig. 5.2, mapping between PAM symbols $\{d_k\}$, $\{d'_k\}$ and $\{a_k\}$ is achieved by a block interleaver (Π in Fig. 5.2) of length K applied to symbols $\{d'_k\}$ to yield $\{d''_k\}$ then a_k is given by $a_k = d_k + jd''_k$.

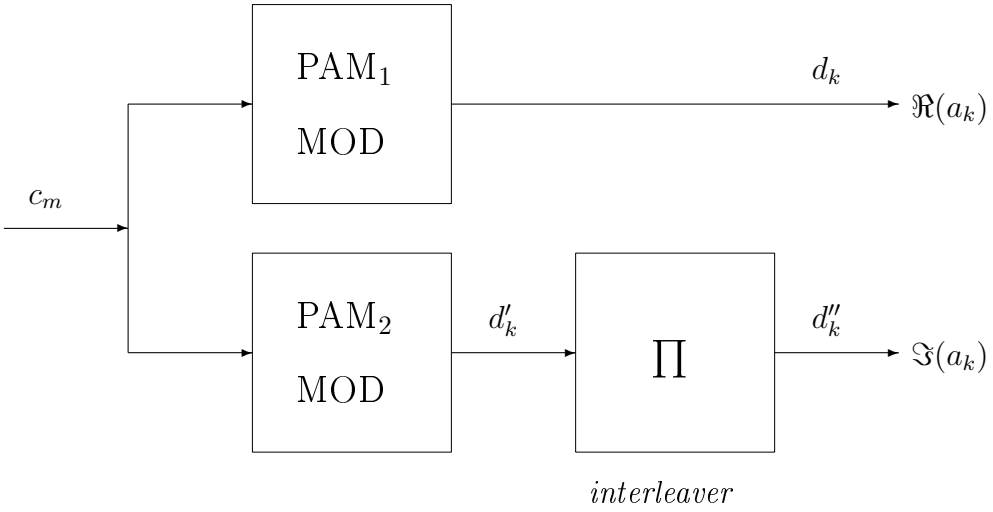

Figure 5.2: The re–mapped repetition modulation.

At the receiver, the phase distortion introduced by the channel is equalized on each subcarrier. Next, from the equalized signal $\{x_k\}$ associated to an OFDM symbol we extract the real $\{x_{I,k}\}$ and imaginary $\{x_{Q,k}\}$ parts. The latter signal is *de–interleaved* to get $\{x'_{Q,k}\}$. Hence, from the signal $\{x_{I,k} + jx'_{Q,k}\}$ we derive the a posteriori probabilities (APP) associated to symbols $\{d_k\}$ and then the LLRs associated to the codebits of $\mathbf{c}_k$.

We illustrate the novel mapping by an example using a 16–PAM constellation $\{\pm 1, \pm 3, \ldots, \pm 15\}$ for the real mappings. If we were to use the same mapping for both d_k and d'_k, the constellation would be as in Fig. 5.3.a. The RR scheme provides instead the use of two real mappings where the value of d'_k depends only on the value of d_k to provide the constellation shown in Fig. 5.3.b. Note also that now RR pairs assume a rotated QAM placement. As the minimum distance between any couple of constellation points of the RR scheme in increased by a factor $\sqrt{17/2}$ with respect to the scheme with the same mapping, we conclude that RR provides a better protection against noise.

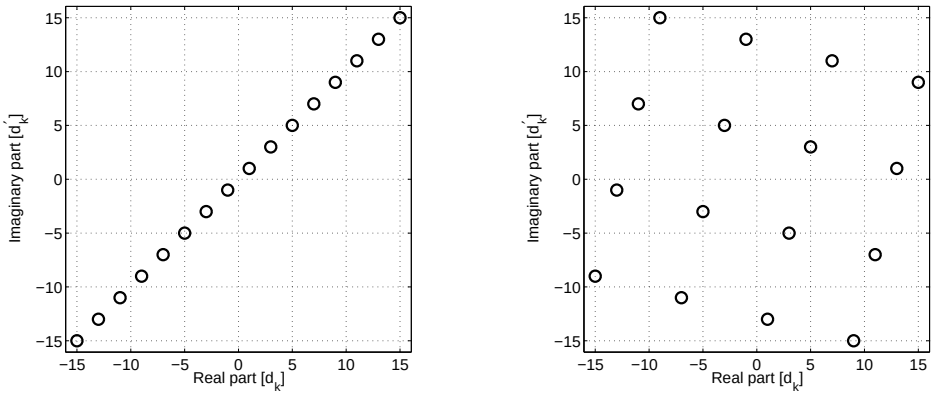

Figure 5.3: Two mappings using a 16PAM for the real and imaginary components of a 256–QAM: a) Repetition mapping for real and imaginary components and b) Repetition with re–mapping.

5.1.2 Rotational Multi–Carrier Modulation

In the R–MCM codebits $\{c_m\}$ are mapped into real PAM symbols $\{y_u\}$, which are grouped into vectors of $D = 2^N$ elements

$$\mathbf{y}_p = [y_{pD}, y_{pD+1}, \ldots, y_{pD+D-1}]^{\mathrm{T}} . \tag{5.3}$$

Each vector $\mathbf{y}_p$ is then multiplied by a $D \times D$ real square rotation matrix $\mathbf{R}$ to yield

$$\mathbf{z}_p = \mathbf{R}\mathbf{y}_p = [z_{pD}, z_{pD+1}, \ldots, z_{pD+D-1}]^{\mathrm{T}} . \tag{5.4}$$

A block interleaver Π' of length $2K$ is applied to sequence $\{z_u\}$ to yield the scrambled sequence $\{z'_u\}$. Afterward, $\{z'_u\}$ is split into two real sequences $\{z'_{2k}\}$ and $\{z'_{2k+1}\}$ to obtain $a_k = z'_{2k} + jz'_{2k+1}$. Fig. 5.4 shows the block diagram of the R–MCM method.

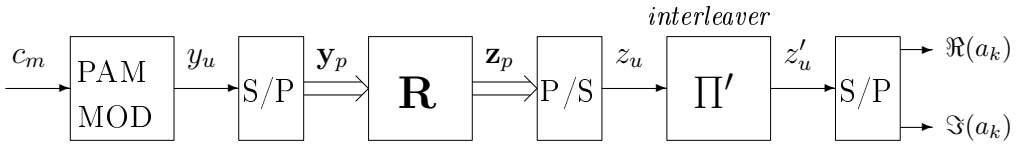

Figure 5.4: The R–MCM and MR–QAM.

At the receiver, after equalization, inverse operations of Fig. 5.4 are applied to yield the APP associated to $\{y_u\}$ and corresponding LLRs associated to codebits of $\{\mathbf{c}_k\}$.

The rotation matrix $\mathbf{R}$ is obtained in a recursive way, similarly to the rotation matrix for the rotational OFDM presented in the 3rd generation partnership project (3GPP) [50]. For a $D \times D$ rotation matrix $\mathbf{R}$, let $\{\theta_i\}$, $i = 1, \ldots, N$, be a sequence of real values in the range $(0, \pi/4)$. We define the orthogonal matrices

$$\mathbf{\Theta}_i \triangleq \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix}. \tag{5.5}$$

Let $\mathbf{R}_1 = \mathbf{\Theta}_1$ and define

$$\mathbf{R}_i \triangleq \mathbf{\Theta}_i \otimes \mathbf{R}_{i-1}, \quad i = 2, \ldots, N\,, \tag{5.6}$$

where $\otimes$ denotes the matrix Kronecker product [55]. We choose $\mathbf{R} = \mathbf{R}_N$. In the Appendix we show also that $\mathbf{R}$ is orthogonal and hence the statistical power of symbols is unchanged by the rotation.

Orthogonality of Rotation Matrix R

In order to show the orthogonality of $\mathbf{R}$ we recall two main properties of the matrix Kronecker product [55]:

1. *transposition* $(\mathbf{A} \otimes \mathbf{B})^{\mathrm{T}} = \mathbf{A}^{\mathrm{T}} \otimes \mathbf{B}^{\mathrm{T}}$;

2. *mixed–product* $(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = \mathbf{AC} \otimes \mathbf{BD}$.

From these properties we have

$$\begin{aligned} \mathbf{R}\mathbf{R}^{\mathrm{T}} &= (\mathbf{\Theta}_N \otimes \mathbf{R}_{N-1})(\mathbf{\Theta}_N \otimes \mathbf{R}_{N-1})^{\mathrm{T}} \\ &= (\mathbf{\Theta}_N \otimes \mathbf{R}_{N-1})(\mathbf{\Theta}_N^{\mathrm{T}} \otimes \mathbf{R}_{N-1}^{\mathrm{T}}) \\ &= \mathbf{\Theta}_N\mathbf{\Theta}_N^{\mathrm{T}} \otimes \mathbf{R}_{N-1}\mathbf{R}_{N-1}^{\mathrm{T}} \\ &= \mathbf{I}_2 \otimes \mathbf{R}_{N-1}\mathbf{R}_{N-1}^{\mathrm{T}} \end{aligned} \tag{5.7}$$

where $\mathbf{I}_2$ is the 2×2 identity matrix. We also obtain by recursion

$$\mathbf{R}\mathbf{R}^{\mathrm{T}} = \underbrace{\mathbf{I}_2 \otimes \mathbf{I}_2 \otimes \cdots \otimes \mathbf{I}_2}_{N \text{ times}} = \mathbf{I}_{2^N} \tag{5.8}$$

that proves the statement.

5.1.3 Multidimensional Rotated QAM

The MR–QAM design scheme is still represented by Fig. 5.4. Now the aim of MR–QAM is to increase the *modulation diversity order* L of $\mathbf{z}_p$, i.e. increase the minimum number of distinct components between any two constellation points. This improves the robustness of the system when the components of $\mathbf{z}_p$ are hit by independent fading.

In MR–QAM the criteria to design the rotation matrix $\mathbf{R}$ are [47],

1. minimize the average power of transmitted signal;
2. maximize the diversity order L;
3. maximize the *minimum L–product distance* defined as

$$\min_{\mathbf{z}\neq\tilde{\mathbf{z}}} \prod_{z_i\neq\tilde{z}_i} |z_i - \tilde{z}_i| \tag{5.9}$$

   where $\mathbf{z}$ and $\tilde{\mathbf{z}}$ are two multidimensional signals of the transmitted constellation;
4. minimize the total number of multidimensional signals at the minimum L–product distance.

The design of rotation matrix $\mathbf{R}$ to satisfy the above criteria is based on the algebraic number theory as discussed in [47], where is shown that as the size of $\mathbf{R}$ goes to infinity, the performance of an additive white Gaussian noise (AWGN) channel is achieved.

For the sake of brevity here their construction is omitted.

5.2 Simulation Results

We consider an OFDM system with $2^{14} = 16,384$ subcarriers, denoted 16k–OFDM, which is a solution for the next generation terrestrial digital video broadcasting (DVB–T2). A cyclic prefix of 1/8 of the OFDM block size is considered. This value also correspond to the largest cyclic prefix in the previous DVB–T standard.

The channel is a portable reception Rayleigh fading channel in its approximated format [2]. It comprises twelve taps spread on a channel duration of $N_c = 51$ samples. In Tab. 5.1 we report the values of the non–zero taps of the portable channel we have implemented for our simulations.

We observe that the channel is normalized, i.e. it has unitary energy $\sum_i |h_i|^2 = 1$. As mapping schemes, we are going to present various solutions.

Table 5.1: The portable channel taps value

i	h_i	i	h_i
1	-0.2083 + j0.1342	9	0.0510 - j0.1082
2	-0.0673 + j0.1097	13	0.1344 + j0.1410
4	0.2896 - j0.1073	18	0.4137 + j0.0644
5	0.3885 - j0.1729	25	-0.1906 + j0.2533
6	-0.4472 - j0.2005	30	-0.1961 - j0.0616
8	0.0048 + j0.0361	50	-0.1776 - j0.0506

The first is a classic 16–QAM mapping where each group of four codebits $\{c_m\}$ is mapped into a complex symbol a_k adopting a Gray coding. The second scheme is the RR modulation obtained from a 16PAM constellation. The interleaver acts on a group of $K = 2^{14}$ PAM symbols. For both R–MCM and MR–QAM a pair of codebits $\{c_m\}$ is mapped into a 4PAM symbol, in order to ensure the same spectral efficiency of 4 bit/s/Hz for all systems. The interleaver length is 2^{15}, since it operates on a single stream of combined real and imaginary components. For R–MCM we then consider three rotation matrices $\mathbf{R}$ having different dimension: a) 2×2 with $\theta_1 = 0.3\pi/4$, b) 4×4 with $\theta_1 = 0.3\pi/4, \theta_2 = 0.7\pi/4$ and c) 8×8 with $\theta_1 = 0.3\pi/4, \theta_2 = 0.5\pi/4, \theta_3 = 0.7\pi/4$. Finally, for MR–QAM we consider also three orthogonal rotation matrices for different dimensions. The selected matrices are chosen from the list presented in [56]. In Section 5.3 we report their numerical values. For both them we use again an interleaver of length 2^{15}. Because all employed rotation matrices are orthogonal, they maintain the average energy of each vector of symbols.

5.2.1 Uncoded Scenario

In the uncoded scenario the information bits $\{c_m\}$ are not protected by a channel code, as they are directly modulated. At the receiver, hard detection is used.

For both QAM and RR best hard detection is given by the maximum likelihood (ML) approach which corresponds to deciding for the constellation symbol nearest to the equalized signal. R–MCM and MR–QAM require instead a real lattice decoding algorithm such as the sphere decoder with fading [31, par. 4.2].

Fig. 5.5 compares the bit error rate (BER) versus the system signal to noise ratio (SNR) for the eight uncoded mapping schemes.

We see that the diversity gain is proportional to the size of the rotation matrix $\mathbf{R}$ [47]. Moreover, the behavior of RR is comparable to the 2×2

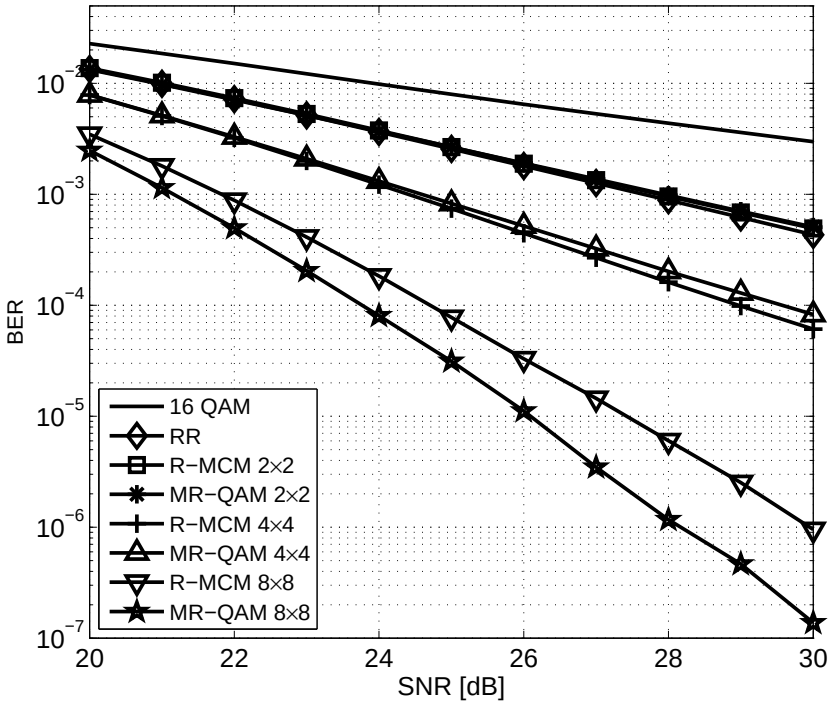

Figure 5.5: Bit error rate in the presence of hard detection for uncoded scenario. System rate 4 bit/s/Hz.

rotation methods. Nevertheless it performs slightly better than both MR–QAM and R–MCM.

For higher dimensions of the rotation matrix we find that MR–QAM outperforms R–MCM because MR–QAMs are designed to provide full diversity and large minimum product distance between any two points of the signal constellation. The gain of MR–QAM is roughly 2 dB at BER=10^{-6} for a 8×8 rotation matrix.

5.2.2 Coded Scenario

The coded scenario refers to the whole scheme in Fig. 5.1 where the source bits are encoded by the LDPC code adopted in the DVB–T2 standard with a codeword length of 64,800 bit/block. We have inserted exactly one LDPC block for each OFDM symbol, leaving 184 virtual subcarriers. Decoding is provided by the sum–product message passing algorithm [57] which stops after 30 iterations.

The LLR computation for soft detection is accomplished by the Group–wise max–log–map complex sphere decoder 4.2.2 based on several applications of the complex sphere decoder [35].

In [49] a comparison is provided between QAM and RR considering only code rate 5/6. Here we extend this comparison by including all the above mapping method and two different code rates: an high–rate of 9/10 and a low–rate of 3/4, both provided by the DVB–T2 standard.

Fig. 5.6 shows the BER vs SNR for code rate 9/10. For this scenario we note that the dimension of the rotation matrix plays the major role on performance. We observe an improvement of almost 1.5 dB going from QAM and RR modulations with 2×2 rotation matrices. Methods with rotation matrix of size 8×8 yield a gain of 0.5 dB over methods with rotation matrix 4×4, while in turn yield a further gain of 0.75 dB over methods with a rotation matrix of size 2×2. In general modulations with the same size rotation matrix yield similar performance.

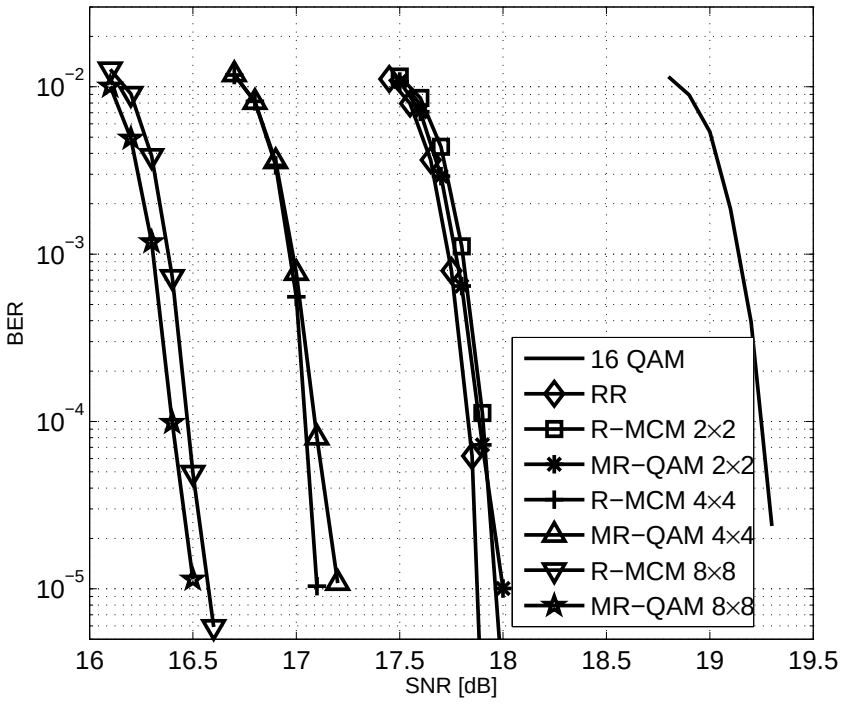

Figure 5.6: Bit error rate in the presence of soft detection and LDPC (code rate 9/10) decoding. System rate $\frac{9}{10}4$ bit/s/Hz.

Fig. 5.7 shows a comparison for the low–rate LDPC. Now we observe

that all methods yield similar performance. The surprising result is that the relative performance of the various modulation techniques is changed with respect to the previous results. In particular, the dimension of the rotation matrix is irrelevant and MR–QAM performs worse than QAM. Overall, best performance is achieved by RR, which outperforms QAM by 0.7 dB.

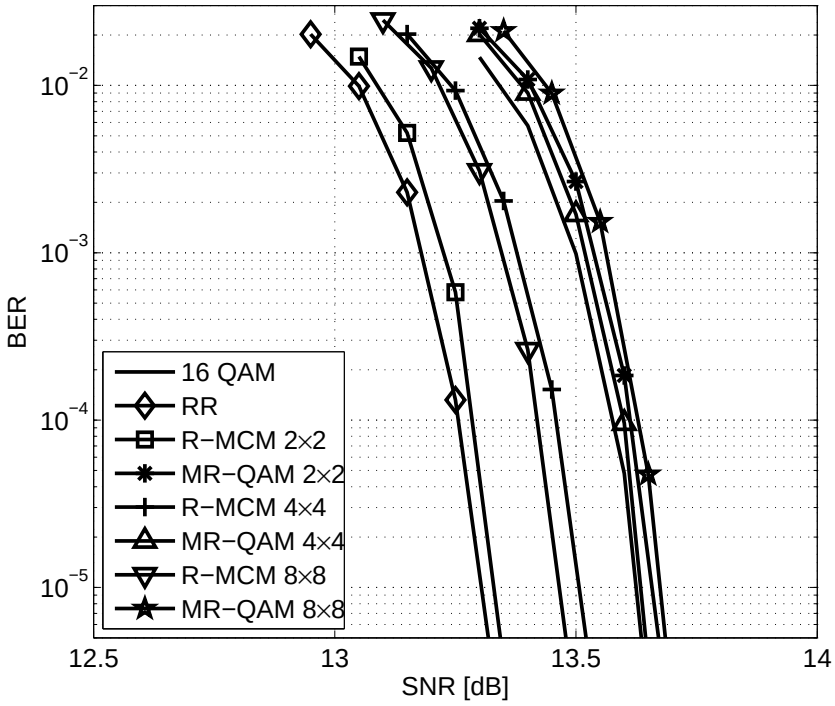

Figure 5.7: Bit error rate in the presence of soft detection and LDPC (code rate 3/4) decoding. System rate $\frac{3}{4}4$ bit/s/Hz.

Indeed it seems that a low–rate code completely retrieves the information associate to subcarriers affected by fading. Conversely, a high code rate is not be able to retrieve all the information and diversity provided by increasing the dimension of the rotation matrix can give a substantial performance improvement as seen in Fig. 5.6.

Results have shown that with a high–rate LDPC code, modulation diversity technique may play a significant role and can substantially improve performance. From our results the best performance is obtained by MR–QAM which outperforms R–MCM by a fraction of dB. However, for a low code rate, the performance of all methods are similar with a slight preference for the RR technique. This feature should come because LDPC gain

dominates the diversity gain provided by the presented modulation schemes.

5.3 MR–QAM Matrices

For the sake of completeness here we report the numerical value of the three rotational matrices regarding to the multidimensional rotated QAM which are used in our simulations.

MR–QAM 2×2

$$\mathbf{R} = \begin{bmatrix} -0.5257311121 & -0.8506508083 \\ -0.8506508083 & 0.5257311121 \end{bmatrix} \tag{5.10}$$

MR–QAM 4×4

$$\mathbf{R} = \begin{bmatrix} -0.3663925121 & -0.7677000238 & 0.4230815704 & 0.3120820187 \\ -0.2264430248 & -0.4744647078 & -0.6845603618 & -0.5049593142 \\ -0.4744647080 & 0.2264430248 & -0.5049593144 & 0.6845603618 \\ -0.7677000246 & 0.3663925106 & 0.3120820189 & -0.4230815707 \end{bmatrix} \tag{5.11}$$

MR–QAM 8×8

$$\mathbf{R} = \begin{bmatrix} -0.0891316083 & -0.1752279466 & -0.2553571073 & -0.3267903880 \\ -0.2553571073 & -0.4342179767 & -0.4830020216 & -0.3870952140 \\ -0.3870952140 & -0.4665539670 & -0.1752279466 & 0.2553571073 \\ -0.4665539670 & -0.2553571073 & 0.3267903880 & 0.4342179767 \\ -0.4830020216 & 0.0891316083 & 0.4665539670 & -0.1752279466 \\ -0.4342179767 & 0.3870952140 & 0.0891316083 & -0.4665539670 \\ -0.3267903880 & 0.4830020216 & -0.3870952140 & 0.0891316083 \\ -0.1752279466 & 0.3267903880 & -0.4342179767 & 0.4830020216 \end{bmatrix}$$
$$\begin{bmatrix} -0.3870952140 & -0.4342179767 & -0.4665539670 & -0.4830020216 \\ -0.1752279466 & 0.0891316083 & 0.3267903880 & 0.4665539670 \\ 0.4830020216 & 0.3267903880 & -0.0891316083 & -0.4342179767 \\ -0.0891316083 & -0.4830020216 & -0.1752279466 & 0.3870952140 \\ -0.4342179767 & 0.2553571073 & 0.3870952140 & -0.3267903880 \\ 0.3267903880 & 0.1752279465 & -0.4830020216 & 0.2553571073 \\ 0.2553571073 & -0.4665539670 & 0.4342179767 & -0.1752279466 \\ -0.4665539670 & 0.3870952140 & -0.2553571073 & 0.0891316083 \end{bmatrix} \tag{5.12}$$

Chapter 6

Joint Performance of Interleaving and Mapping

The new digital terrestrial television (DVB–T2) standard has been designed and defined [58] for use in a post–Analogue Switch–Off (ASO) environment introduces the latest modulation and coding techniques to enable highly efficient use of valuable terrestrial spectrum for the delivery of audio, video and data services to fixed, portable and mobile devices. DVB–T2 then increase the efficiencies in its use of spectrum compared to DVB–T [2].

Starting from the fact that the DVB–T2 standard includes powerful coding schemes which include low–density parity–check (LDPC) [54] block codes of the digital satellite television standard DVB–S2 [5] for their near Shannon limit performance [43], we investigated any techniques in order to improve the performance in terms of signal to noise ratio (SNR) to achieve a bit error rate (BER) target. In particular, when LDPC coding is considered, we recall that soft detection must be applied at the receiver in order to obtain the log likelihood ratio (LLR) relative to each bit as well as trellis–coded modulation with convolutional forward error correction (FEC) decoded by a soft output Viterbi algorithm (SOVA) [59].

A first strategy we considered is to combine methods implemented for other DVB standards, such as bit demultiplexing defined as a specific for inner interleaving of the old DVB–T standard and block interleavers used as bit interleaver present in DVB–S2 standard.

Another technique is to consider the fact that LDPC codes require LLRs to be decoded and adopt such schemes typically developed for other codes which require LLRs as well. In particular we refer to the bit–interleaved code modulation with iterative demapping (BIMC–ID). This method already analyzed for convolutional codes [52, 53, 60], here it is rearranged for LDPC. It consists of the concatenation of an LDPC encoder, an interleaver and a

symbol mapper at the transmitter and at the receiver it provides a feedback for improve the demapping and LLR computation by using the extrinsic information.

Finally, when fading channel occurs, other modulation diversity techniques that provide diversity with no spectral or power inefficiencies should be considered. In particular here we adopt the re–mapped repetition method explained previously in section 5.1.1 in order to verify its effect on a DVB–T2–*like* simplified scenario.

The aim of this chapter is to give an overview of several combinations of these methods and point out which are good and which are not.

6.1 DVB–T2 Simplified System Model

The whole DVB–T2 system model is very complex. In the aim of this analysis we can treat with a model with reduced complexity. The baseband discrete–time model of the considered broadcast wireless communication system, with the iterative decoding feedback, is shown in Fig. 6.1. It combines classical functional blocks taken from digital video broadcasting (DVB) standards.

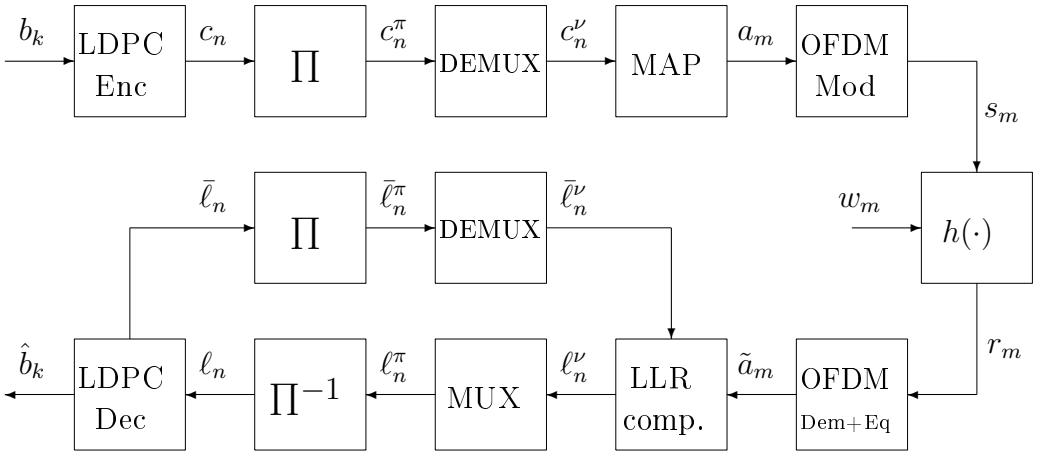

Figure 6.1: The complete system model.

Starting with an information stream $\{b_k\}$ of binary digits, $b_k \in \{0, 1\}$, we obtain a coded bit stream $\{c_n\}$ by coding blocks of source bit with a low–density parity–check (LDPC) block code suggested by DVB–S2 standard and reused by DVB–T2.

Each LDPC coded block can be interleaved by an either random interleaver [61] or a DVB–S2 interleaver [5] to obtain the bit stream $\{c_n^\pi\}$. The former is simply a random permutation which performs well for large length

of interleaved blocks as verified for Turbo codes. The latter is taken from second generation satellite digital video broadcasting standard and based on a matrix where data are serially written into the interleaver column–wise, and serially read out row–wise: from the first to the last column for the direct mode and from the last to the first in the reverse mode.

Before to be mapped into complex symbols of a M–ary constellation χ, interleaved coded bits can be demultiplexed in groups of $v = \log_2 M$ bits as well as in the non–hierarchical mode proposed in terrestrial digital video broadcasting standard (DVB–T) [2]. For each group if we enumerate the position i from 0 to $v-1$, the new demultiplexed position p_i is computed as

$$p_i = 2(i \mod v/2) + (i \text{ div } v/2) \tag{6.1}$$

where a div b is the result of the integer division a/b and $a \mod b$ denotes the remainder of a/b. For example for a constellation of 256 symbols the demultiplexer arrangement is

$$\begin{array}{lcccccccc} i: & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ & & & & \Downarrow & & & & \\ p_i: & 0 & 2 & 4 & 6 & 1 & 3 & 5 & 7 \end{array} \tag{6.2}$$

where only the first and the last bits preserve the original position.

The demultiplexed stream $\{c_n^\nu\}$ is now mapped into complex symbols stream $\{a_m\}$, i.e. $a_m = \mu(\mathbf{c}_m^\nu)$ where $\mathbf{c}_m^\nu = [c_{mv}^\nu, \ldots, c_{mv+v-1}^\nu]$. The constellation and mapping methods used are either the ordinary Gray QAM defined by DVB–T standard [2] or their rotated version revisited in section 5.1.1, here called *re–mapped repetition* (RR). Moreover, the latter includes a symbol interleaver that mixes to the imaginary parts of the symbols over the OFDM symbol.

In turn, sequence $\{a_m\}$ is OFDM modulate to obtain the signal $\{s_m\}$.

At the receiver the signal $\{r_m\}$ is obtained by filtering the transmitted signal with the equivalent discrete–time time–invariant channel $\{h_i\}$ of duration N_h and by adding the additive white Gaussian noise (AWGN) $\{w_m\}$, i.e. the samples of AWGN $w_m \in \mathcal{CN}(0, N_0)$ are i.i.d. so the received signal can be written as

$$r_m = \sum_{i=0}^{N_h-1} h_i s_{m-i} + w_m. \tag{6.3}$$

The signal is demodulated and equalized by the OFDM receiver to provide the symbols $\tilde{a}_m$.

Such symbols are used together the extrinsic information $\bar{\ell}_n^\mu$ to compute the log likelihood ratio (LLR) [52] necessary by the message passing algorithm

which decodes the LDPC. Following, the formula for LLR computation is

$$\ell^{\nu}_{mv+k} = \ln \frac{\sum\limits_{\mathbf{c}\in\{0,1\}^v, c_k=1} p(\tilde{a}_m|a_m=\mu(\mathbf{c}))\cdot \exp \sum\limits_{j\in\mathbb{J}^k_{\mathbf{c}}} \bar{\ell}^{\nu}_{mv+j}}{\sum\limits_{\mathbf{c}\in\{0,1\}^v, c_k=0} p(\tilde{a}_m|a_m=\mu(\mathbf{c}))\cdot \exp \sum\limits_{j\in\mathbb{J}^k_{\mathbf{c}}} \bar{\ell}^{\nu}_{mv+j}},$$
$$k = 0,\ldots,v-1 \quad (6.4)$$

where $\mathbb{J}^k_{\mathbf{c}} = \{j|j=0,\ldots,v-1, j\neq k, c_j=1\}$ is a subset of indexes. At the first iteration all values of extrinsic information ℓ^{ν}_m are set to zero, i.e. no extrinsic information.

After restructuring the order of LLR, we have the sequence $\{\ell_n\}$ that is decoded via LDPC belief propagation algorithm [54, 62] to provide the extrinsic information $\{\bar{\ell}_n\}$ for the next iteration and the estimated bits $\{\hat{b}_k\}$ obtained from the sign (hard detection) of $\{\bar{\ell}_n\}$.

6.2 Interleavers and Demux from DVB

In order to analyze how several DVB standard blocks perform we consider the ensemble of various combination in an hybrid DVB scenario in terms of constellation dimension M, channel model, number of OFDM carriers and so on.

LDPC codes are taken from DVB–S2 standard [5]. In our simulation we have used the so–called *normal fecframe* which provides coded word length $n_{\text{ldpc}} = 64,800$ bits for several rate from 1/4 to 9/10.

In this simulation the iterative demapping is not implemented because it will be considered apart in next section, so here the feedback of Fig. 6.1 has to be ignored.

6.2.1 AWGN channel

The first one use an AWGN channel model, i.e. $N_h = 1$ and $h_0 = 1$. This assumption allows us to avoid the OFDM blocks from the general system model. The choice of the interleaver falls among three possibilities:

1. no interleaving algorithm implemented (NO);
2. DVB–S2 direct mode (DVB–S2);
3. DVB–S2 reverse mode (rev S2).

The DVB–S2 interleaver matrix size is $(n_{\text{ldpc}}/v) \times v$ for both modes. For the demultiplexer we have consider two possibilities:

1. no demux (NO);
2. DVB–T non–hierarchical mode.

In Fig. 6.2 we report the six combinations of these interleaver and multiplexer for a system with LDPC code rate 1/2 and a 64–QAM mapping.

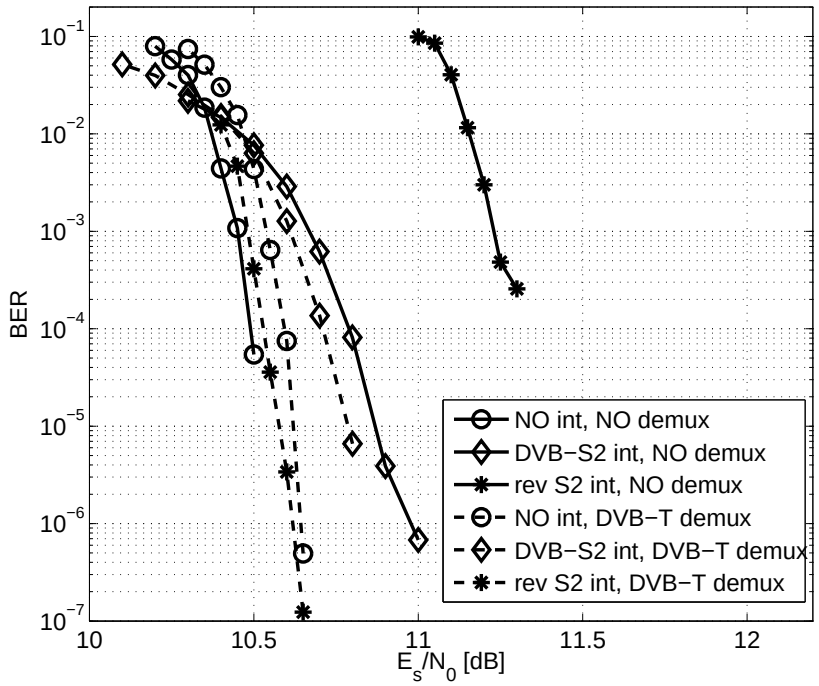

Figure 6.2: Implementation of DVB standard features on the system with LDPC code rate 1/2 and constellation 64–QAM over AWGN channel.

Apart form the case with reverse DVB–S2 interleaver mode without demux, which loses almost 0.8 dB, all the other cases have similar performance. Using DVB–S2 direct interleaver the slope of the curves appears little more smooth. Surprisingly we see that in this scenario it is better to do nothing, i.e. no interleaver and no demultiplexer, so we obtain the best performance.

6.2.2 Portable channel

We are also interested to investigate on the performance of these features in further scenarios which better represent a DVB–T2 requirements [58]. Now

we consider a different channel model that is a discrete–time approximation of the portable reception Rayleigh fading channel defined in [2] which does not provide the line of sight ray causing deep fading effect in several OFDM subcarriers in order to simulate a bad television receiving environment. The model, having $N_h = 51$, is the same used in the previous chapter. The non–zero tap values are reported in Table 5.1. The modulation mapping adopted is 256–QAM which is the greatest constellation provided by DVB–T2 standard. The OFDM operates in 32k mode, that means the 8 MHz band of a channel is fractioned in $32,768$ subcarriers. An integer number of LDPC blocks are inserted in a OFDM symbol then the unused subcarriers are considered as virtual carriers. The results of this configuration performance are depicted for the LDPC code rate $1/2$ in Fig. 6.3 and for the rate $9/10$ in Fig. 6.4.

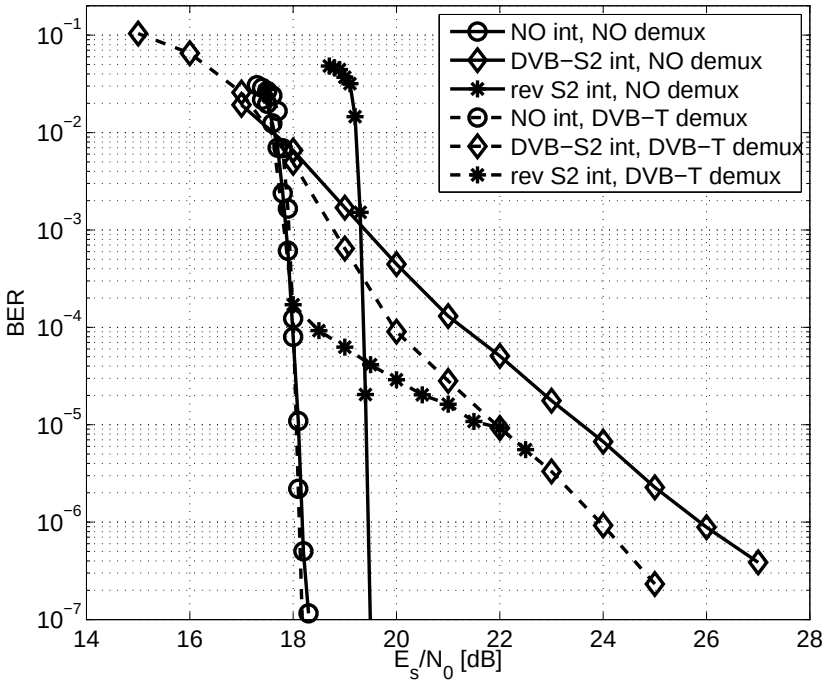

Figure 6.3: System with LDPC code rate $1/2$ and constellation 256–QAM over Portable channel.

In the first case we note how DVB–S2 direct interleaver degrades the slope trend. The same happens for the combination of DVB–S2 reverse interleaver with DVB–T demux. Again for DVB–S2 reverse interleaver without demux we have a loss of almost 1.5 dB.

In the second case, where the LDPC correction is less robust, we observe

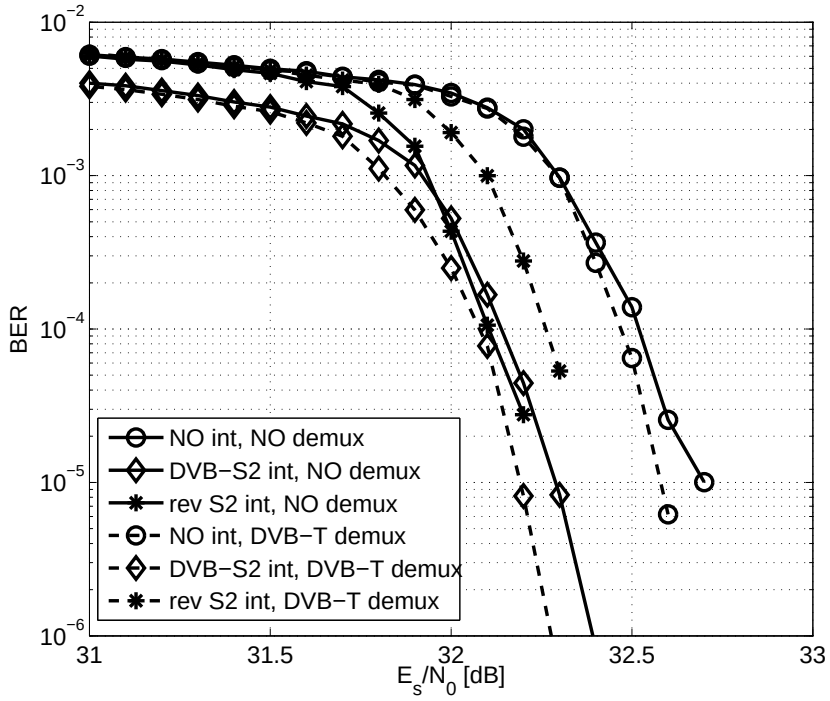

Figure 6.4: System with LDPC code rate 9/10 and constellation 256–QAM over Portable channel.

instead a reverse effect. Here DVB–S2 direct interleaver outperforms the other strategies. However all curves are grouped within 0.5 dB that is not comparable with the loss of the previous case.

From the analysis of the curves results that the DVB standard methods does not help a DVB–T2 system to improve its performance. This is not true for a singular case where we set the greatest LDPC code rate and QAM constellation size transmitted over a portable channel and we reach a gain of 0.5 dB. Nevertheless this scenario represents a very particular sort of extreme transmission environment. We conclude that the standard methods for DVB–S2 and DVB–T are not so reliable for DVB–T2 scenarios.

6.3 Nonstandard Methods

The above methods does not provide significant performance improvement. Hence, nonstandard methods are here investigated. They are analyzed in order to point out better solutions which could improve DVB–T2 systems.

6.3.1 Iterative Demapping

We consider now the iterative demapping (ID) method [60] that is based on the recalculation of the soft information on bits by using the extrinsic information at the receiver. This operation can be done more and more times by iterating the feedback loop. Here we limit the implementation to five iteration . The interleaver we employ in this scenario is a simple random permutation of length n_{ldpc}. This ensures that bits that belong to the same parity check equation are affected by uncorrelated noise.

In Fig. 6.5, apposed to the system without interleaver, we show the performance of the system with a random interleaver without iterative demapping and after five iteration. This is done with LDPC code rate 1/2 and with a 64–QAM constellation over AWGN channel.

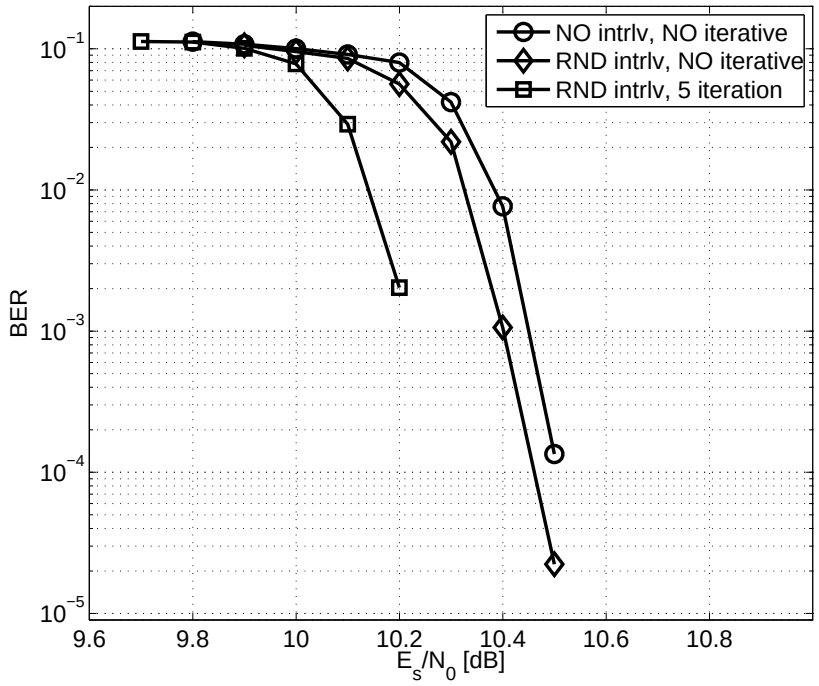

Figure 6.5: Iterative demapping on the system with LDPC code rate 1/2 and constellation 64–QAM over AWGN channel.

First of all we note that a random interleaver, rather than structured interleaver proposed by DVB–S2 standard, does not degrade the system performance but it provide a little improvement. After the iterations we obtain a global gain of 0.3 dB.

Moreover this very small gain comes only for small LDPC code rate. Actually we have verified that there is no gain for rate greater or equal to 4/5 neither by adding a random interleaver nor by applying any iteration.

Another technique we have investigate is to map coded bits on a QAM constellation with the optimized symbol mapping $\mu(\cdot)$ proposed in [53]. It shows that for BICM–ID scheme the Gray mapping is not optimal so it is replaced with a mapping optimized by the binary switching algorithm (BSA) [63] by using a cost function $W(\cdot)$ based on error bounds for AWGN channel. Let $\alpha = \mu(\mathbf{c})$ and $\alpha_i = \mu([c_1 \dots \bar{c}_i \dots c_v])$, the cost function related to the symbol α is computed as

$$W(\alpha) = \frac{1}{vM} \sum_{i=1}^{v} \exp\left\{ -\frac{E_s}{4N_0} |\alpha - \alpha_i|^2 \right\} \tag{6.5}$$

where E_s is the average energy of the constellation χ. The cost function of the whole constellation is given by the summation over all the symbol costs $W(\chi) = \sum_{\alpha \in \chi} W(\alpha)$.

The results is a mapping that tends to map as far apart as possible symbols that map bit sequence having hamming distance equal to 1, i.e. the reverse feature of the Gray.

In particular it can be shown that this optimum mapping for BIMC–ID degrades more than 5 dB compared to Gray mapping, of which only 0.5 dB are recovered after 5 iteration, revealing that it is not good for LDPC codes.

6.3.2 Re–mapped Repetition

When we return to consider portable channel, we are encouraged to adopt modulation techniques that avail the frequency diversity brought by the channel. Any modulation diversity techniques are presented in [49] which performance is even compared for LDPC coded scenarios. The best tradeoff is offered by the re–mapped repetition (RR) modulation, also presented in section 5.1.1, that ensure a gain for different LDPC code rates.

The RR technique has been proposed to improve the performance of OFDM in an environment with severe frequency selectivity by exploiting the fact that simultaneous deep fade on different subcarriers is unlikely. We recall, the basic idea of RR is to split the real and imaginary part of a symbol and transmit them on two different OFDM subcarriers to provide that the two parts are afflicted by independent fading. It can be shown that we improve the performance if we use in place of a traditional M–QAM constellation, a rotated version of it. The rotation angle of RR has been

calculated to be

$$\theta = -\arctan\left(\frac{1}{\sqrt{M}}\right) \tag{6.6}$$

which is the rotation angle which ensures the maximization of the minimum distance between two constellation points whenever one of the two constellation dimension collapse.

In this section we compare the performance of iterative demapping by considering two mapping methods which are the traditional QAM and the RR method. In Fig. 6.6 we compare the maps having constellation size $M = 64$ with LDPC code rate 1/2. In Fig. 6.7 we consider instead a constellation size $M = 256$ with LDPC code rate 9/10.

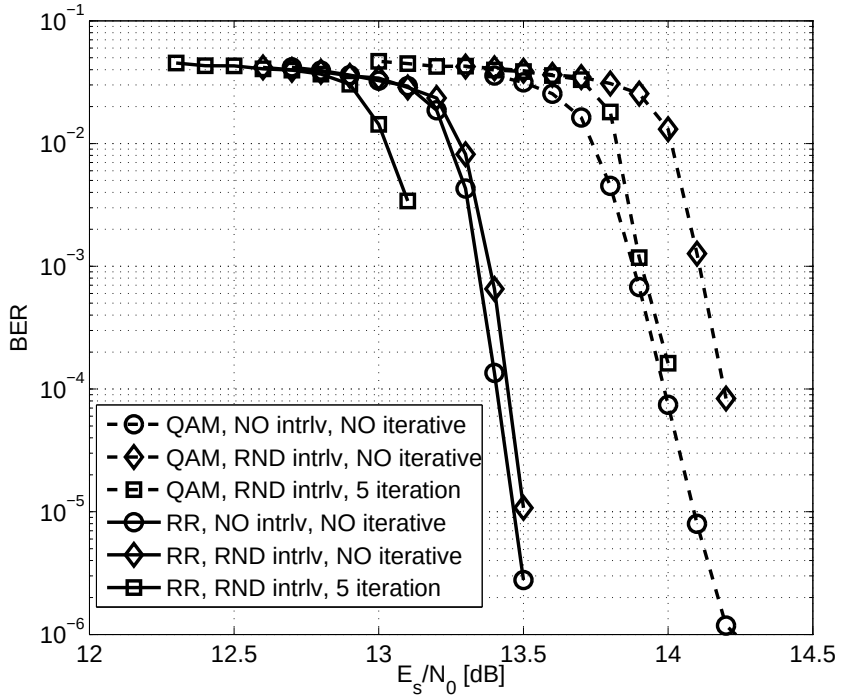

Figure 6.6: Comparison between 64–QAM and 64–RR on the system with LDPC code rate 1/2 over Portable channel.

For the scenario with rate 1/2 we note that random interleaver degrades of 0.3 dB on the QAM performance but they are absorbed by the iterative demapping. However if we apply the RR the system gain is 0.7 dB and a further gain of 0.3 dB is provided by the iterative demapping so we have a comprehensive improvement of 1 dB.

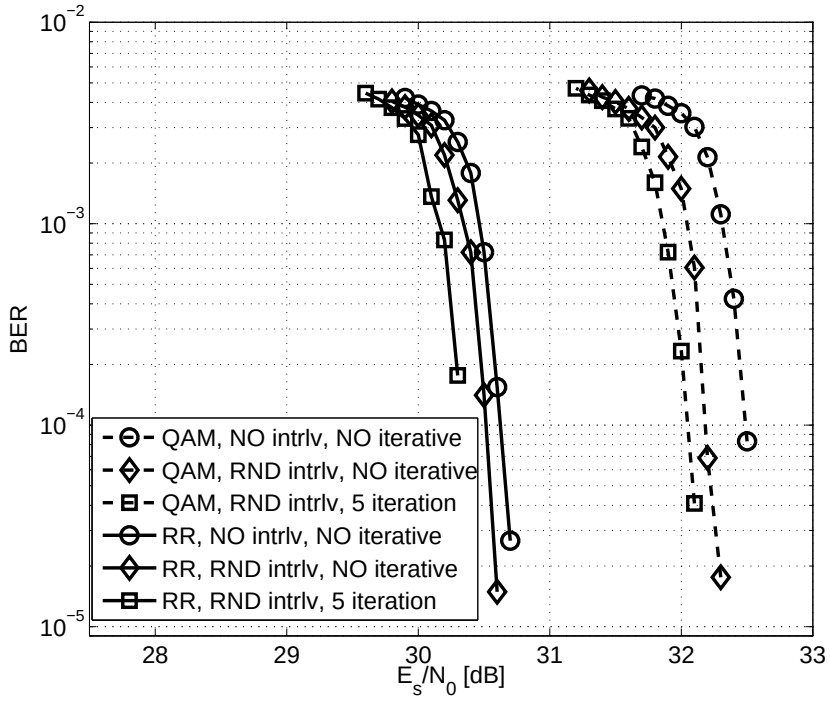

Figure 6.7: Comparison between 256–QAM and 256–RR on the system with LDPC code rate 9/10 over Portable channel.

Regarding the scenario with rate 9/10 we find a similar behavior for both QAM and RR by applying first a random interleaver and after the iterative demapping, saving almost 0.4 dB. Furthermore, it is interesting to note that here RR outperforms QAM of almost 2 dB that is an important results because we are operating at very high values of signal to noise ratio E_s/N_0.

In conclusion of our survey we try to really improve the performance by introducing other methods. The iterative demapping has shown that a further gain of 0.3 dB can be take out after few iterations. Nevertheless we have to consider that the LDPC decoding represents the bottleneck in terms of computational cost and this is a problem since this cost grows linearly with the number of the iterations. In portable environment a greater gain can be achieved by employing a RR mapping. The gain order is $0.7 \div 2$ dB for medium–high size constellations. The cost of this method is only a mapping latency dues to the symbol interleaver which serves to place real and imaginary part of symbols into different OFDM subcarriers.

Finally these results encourage strongly to use a modulation diversity scheme in portable–like scenarios, i.e. environment with severe frequency

selectivity caused by multipath.

Chapter 7

Coded Decision Direct Demodulation

In this chapter we evaluate performance of the coded decision directed demodulation (CD3) technique for channel estimation in DVB–T2. In fact, by iterating between decoding and channel estimation, with the latter obtained by using previously decoded symbols, CD3 is able to provide an accurate channel estimation with very few pilots, even in the presence of highly frequency dispersive and time–variant channels. Numerical results are provided for a comparison in terms of achievable throughput between DVB–T2 with CD3 and current DVB–T. Moreover, multiple antennas at the base station are also considered to provide spatial diversity and hence further improve system performance.

The DVB–T2 standard draft [58] included transmission of pilot symbols within each OFDM block, which are used at the receiver to obtain a channel estimate. Then, frequency interpolation is used to provide a channel estimate also on data carriers. However, the number of pilot symbols required for an accurate channel estimation increases with both channel dispersion and channel time variations, due to Doppler spread, thus reducing the spectral efficiency of the transmission. In recent years, a number of solutions have been proposed in the literature to improve the spectral efficiency by obtaining channel estimates with fewer or no pilots. Among others, we recall here the use of pilots superimposed to data symbols [64], iterative detection and channel estimation also for mobile channels [65], time–domain approaches for the channel estimate [66] and blind approaches [67]. As a mean to improve the DVB–T2 system throughput, in this chapter we analyze the performance of channel estimation by using decisions taken on the received signal and iterating between detection and channel estimation. This is the so called *coded decision directed demodulation* (CD3) [9]. At the beginning of a transmission,

a pilot OFDM symbol must be inserted to allow initialization of the channel estimate, while detection starts from the following symbol. This reference symbol is actually provided by the DVB–T2 standard, where each frame, comprising various OFDM symbols, is initiated with an OFDM symbol including additional pilots. It is seen that CD3 is bandwidth efficient, while still providing high performance in terms of achievable throughput even for very dispersive and time varying channels.

7.1 DVB–T2 System Model

In chapter 2 we have presented an overview of DVB–T2 system. Here we present an extract, as shown in Fig. 7.1, which is involved in our simulations. Then we recall the fundamental features of the standard focusing on the link layer, the error protection strategy, the frame builder an the OFDM modulation.

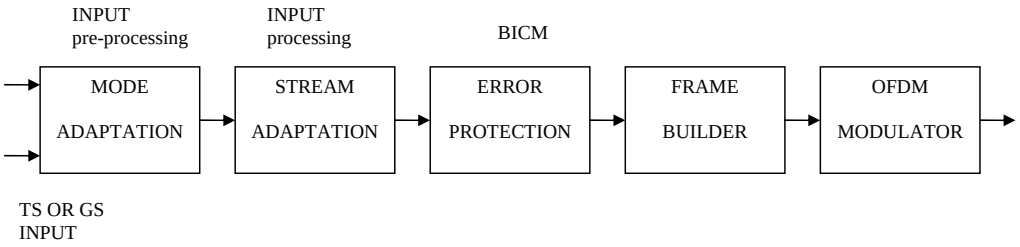

Figure 7.1: Baseline scheme of DVB–T2.

7.1.1 Link Layer

The transmitted signal may comprise many *services*, namely the main video signal, subtitling, multiple audio tracks, interactive content, etc. The link layer adapts the data associated to each service to the underlying layers. For example, the video signal (*stream*) formatted according to the Moving Picture Experts Group (MPEG–2) standard is allocated into a frame, which is transmitted by underlying layers. As for DVB–S2, DVB–T2 can operate with two types of input streams, namely transport stream (TS) and generic stream (GS), supporting both a single and a multiple input stream mode. The TS is a packetized stream, characterized by packets of 188 bytes, while GS can be either a continuous or a packetized stream with constant length

packets. We note that GS is directly compatible with the generic stream encapsulation (GSE) protocol developed by DVB to adapt an Internet protocol (IP) stream into a GS [68]. Input can be either single stream or multiple streams: in the first case all services are protected with the same reliability (modulation and coding) and a constant coding and modulation (CCM) is used on the stream, while in the second case each service may be protected by a different modulation and coding scheme, in other words we have a variable coding and modulation. Each input is denoted as a physical layer pipe (PLP). As from the original standard proposal, PLPs can be grouped when associated to the same modulation and coding.

The PLPs are processed first by a mode adaptation module, which performs synchronization of input streams (in the case of multiple input streams), cyclic redundancy check (CRC) insertion and base band (BB) header insertion. A stream adaptation module follows, which provides scheduling (for the multiple input streams), padding to complete a constant length frame and scrambling for energy dispersal.

7.1.2 Error Protection

The LDPC/BCH forward error correction scheme of DVB–S2 [3] is adopted, achieving quasi error free (QEF) operation at about 0.7dB to 1 dB from the Shannon limit, depending on the transmission mode. QEF is defined as less than one uncorrected error event per transmission hour, for a single TV source bit rate of 5 Mbit/s, corresponding to a packet error rate $PER = 10^{-7}$ at the input of the MPEG–2 demultiplexer.

Groups of bits (cells) obtained from LDPC/BCH are bit interleaved, demultiplexed and mapped with Gray mapping into QPSK, 16–QAM, 64–QAM or 256–QAM constellation symbols. Optional constellation rotation is also considered to improve system performance on deep frequency selective channels, typical of the portable reception.

To counteract the bursty nature of errors in the terrestrial channel, both in time (Doppler effect, impulsive noise) and in frequency (frequency selective fading), cell, time and frequency interleaving are then applied, to spread error burst before entering the FEC decoder, thus fully exploiting the FEC error correction capabilities.

7.1.3 Frame Builder

The frame builder organizes data symbols before modulation. In particular, as shown in Fig. 7.2, a super–frame is divided into frames, which are further

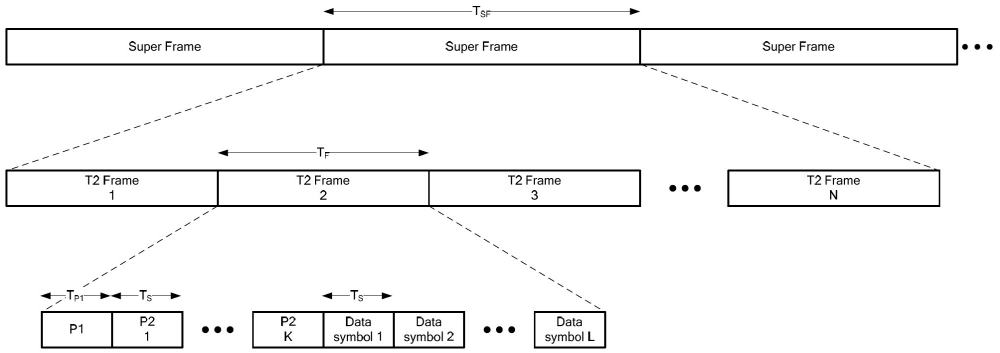

Figure 7.2: Super–frame structure (see [58]).

split into OFDM symbols. Each frame starts with a symbol P1 for synchronization purposes, followed by control symbols P2 and then data OFDM symbols.

7.1.4 OFDM Modulation

As in DVB–T, also for DVB–T2 channel estimation is obtained by inserting scattered pilot tones on a subset of OFDM subcarriers . At the receiver, channel estimation is performed on pilot carriers and then extended via interpolation on all OFDM subcarriers. Several pilot pattern configurations are available, matched to the scenarios for which the system is defined, in particular multi frequency networks (MFNs) and SFNs.

Control symbol P2 provides additional pilots for the initial channel estimate. In one pilot pattern configuration (PP8), there is only one pilot out of 96 data carriers per OFDM symbol. This mode allows a high–efficiency transmission but is feasible only for channels with low dispersion. However, by exploiting additional pilots in P2 symbol at the beginning of each frame and the CD3 iterative detection/decoding principle described in Section III, this efficient pilot configuration can be used also in the presence of very dispersive channels.

7.2 The CD3 Principles

OFDM transforms transmission over a dispersive channel into the parallel of N narrow–band channels. The OFDM symbol rate $1/T_s$ is usually very low.

A time guard interval (GI) with duration

$$T_g = MT_u, \quad \text{where} \quad T_u = \frac{T_s}{N+M}, \tag{7.1}$$

is inserted between adjacent OFDM symbols in order to avoid inter–symbol interference (ISI). The receiver samples the input signal at a frequency $f_c = 1/T_u$, and discards M samples of the GI from the $M+N$ complex samples corresponding to an OFDM symbol, so that echoes reaching the receiver with a delay shorter than T_g do not produce ISI. To achieve large T_g values while keeping high transmission efficiency, the OFDM symbol duration must be very long. Since the OFDM subcarrier spacing is proportional to $1/T_u$, an increase of the symbol duration corresponds to a proportional increase of the number of OFDM subcarriers allocated in a given bandwidth B_w. In addition to the GI, OFDM systems make use of powerful error correction codes, allowing the reconstruction of the information transported by subcarriers affected by deep fading.

A baseband complex envelope representation of the (sampled) signals will be used in the following, where n is the discrete time variable (index of the OFDM symbol) and k is the discrete frequency variable (index of the OFDM subcarrier). The elementary complex transmitted signal, over OFDM symbol n, subcarrier and k, is written as $x_{(n,k)}$.

The channel frequency response $H_{(n,k)}$, although variable throughout the total signal bandwidth (index k), generates a phase rotation and an amplitude variation of the corresponding transmitted signal $x_{(n,k)}$. $\{H_{(n,k)}\}$ shows also a time variation, depending on the moving obstacles around the receiver and on the receiver motion. In the following, it is assumed that $|H_{(n,k)}|$ is quasi–stationary during an OFDM symbol period, and that it is slowly changing over several OFDM symbol periods. The elementary complex received signal $y_{(n,k)}$ (after translation to base–band and OFDM demodulation) is a replica of the transmitted signal $x_{(n,k)}$ multiplied by the channel frequency response, plus a complex Gaussian noise component $w_{(n,k)}$, i.e.

$$y_{(n,k)} = x_{(n,k)} \cdot H_{(n,k)} + w_{(n,k)} \,. \tag{7.2}$$

7.2.1 Channel Estimation

Coherent demodulation requires estimate $\hat{H}_{(n,k)}$ of the channel frequency response $H_{(n,k)}$, so that the signal can be properly scaled as follows

$$z_{(n,k)} = \frac{y_{(n,k)}}{\hat{H}_{(n,k)}} \simeq x_{(n,k)} + \nu_{(n,k)} \tag{7.3}$$

where $\nu_{(n,k)} = w_{(n,k)}/\hat{H}_{(n,k)}$. If the transmitted OFDM symbol at time $n-1$ is known *a priori* by the receiver (reference sequence), the channel frequency response at time $(n-1)$ could be estimated by dividing the received signal $y_{(n-1,k)}$ by the transmitted signal $x_{(n-1,k)}$ as

$$\hat{H}_{(n-1,k)} = \frac{y_{(n-1,k)}}{x_{(n-1,k)}} = H_{(n-1,k)} + \epsilon(n-1,k) \tag{7.4}$$

where $\epsilon(n-1,k) = w_{(n-1,k)}/x_{(n-1,k)}$ is a Gaussian noise component, depending also on the amplitude of the transmitted signal $x_{(n-1,k)}$. Once $\hat{H}_{(n-1,k)}$ is derived, equalization of the successive symbol at time n can be easily obtained by (7.3), assuming that the channel frequency response is quasi–stationary over the two instants $n-1$ and n as

$$z_{(n,k)} = \frac{y_{(n,k)}}{\hat{H}_{(n-1,k)}} \simeq x_{(n,k)} + \nu_{(n,k)}\,. \tag{7.5}$$

In this case, the same BER versus C/N performance as in differential demodulation is achieved, since the channel estimate is as noisy as the signal $y_{(n-1,k)}$. Decoding and re–encoding of $z_{(n,k)}$ follows to obtain an estimate of the transmitted data symbol $\hat{x}_{(n,k)}$. Using this estimate, an updated channel estimate can be obtained similar to (7.4), to be used for equalization and decoding of the next OFDM symbol.

Note that $\{\hat{H}_{(n,k)}\}$ has been obtained on M subcarriers. However, by assuming that the channel duration in the time domain is at most N taps to fit the GI, we conclude that the estimate $\hat{H}_{(n,k)}$ can be described by fewer parameters. Indeed, various approaches have been proposed in the literature to improve channel estimate exploiting the redundancy of many subcarrier estimates (see e.g. [69, 70]). For example, the channel estimate $\hat{H}_{(n,k)}$ can be low–pass filtered in the frequency domain to average the noise component $\epsilon_{(n,k)}$, with a filter which should be flat in the time domain over an interval equal to T_g. If the ideal filter is used, a C/N improvement of $10\log_{10}(N/M)$ dB for the same BER is obtained. Assuming $N/M = 4$ and QPSK constellation, the C/N improvement is of 6 dB, corresponding to a C/N gain of the order of 2 dB compared to differential demodulation. With additional time–domain filtering (according to the channel variation speed), the BER vs C/N performance of ideal coherent demodulation can be almost approached.

To summarize, CD3, whose scheme is illustrated in Fig. 7.3, overcomes the need for transmission of a large percentage of pilot tones and can be described by the following steps:

1. start the decoding process from a reference sequence $x_{(n=1,k)}$;

2. perform channel estimation according to (7.4) and compute the C/N of each subcarrier to be passed to the decoder;

3. filter the channel estimate in the frequency and in the time domain to reduce the noise components;

4. equalize the next OFDM symbol through (7.5);

5. deliver the bit–stream after decoding to the user;

6. re–encode and re–modulate the bit–stream after decoding to yield a reliable detection of the transmitted OFDM symbols;

7. the detected symbol sequence after error correction is used in a feedback loop to perform the channel estimation relevant to symbol n, according to step 1, so that the process can continue on a symbol–by–symbol basis.

Using this process, the need to transmit pilot subcarriers, as well as additional training sequences, is in principle avoided.

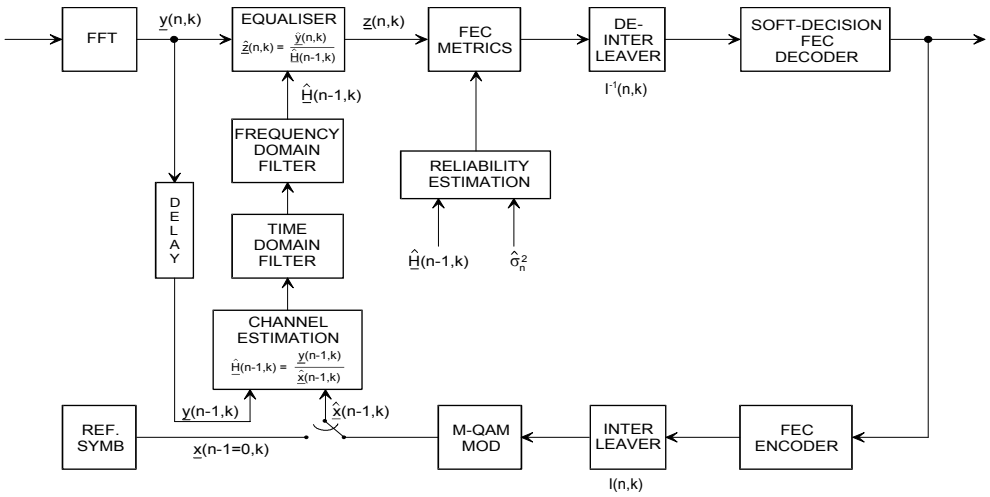

Figure 7.3: Basic scheme of a CD3 receiver.

Moreover, as the CD3 technique updates the channel estimate on a OFDM symbol basis, it allows a fast tracking of the channel variations over time. Reliability of the estimate is instead provided by the error correction capability of the code. Still, a drawback of CD3 is the additional delay introduced by decoding in the estimation process.

In the case of non–constant envelope constellations (i.e. 16–QAM and 64–QAM) the level of noise $\epsilon_{(n-1,k)}$ associated to the channel estimate significantly changes from sample to sample depending upon the selected constellation point. To reduce the degradation associated to lower amplitude constellation points, the following method is proposed: when, for a particular subcarrier, one of the N_P most internal points of the constellation is detected, the previous channel estimate is re–used (hold). This operation is anyway followed by frequency domain filtering. In the simulations, we choose $N_P = 16$ with 64–QAM constellations, as a compromise between good channel tracking and performance.

Moreover, the CD3 iterative mechanism could become unstable for low values of BER, when the code gain is reduced and errors are re–injected into the equalizer through the feedback loop of the channel estimate, providing in turn wrong soft–decision information to the decoder input. This could produce a rapid increase of the BER and a system break–down, until the reception of the next reference OFDM symbol. Instability is alleviated (better, shifted to very high BERs) by using powerful codes and frequency domain filtering, which attenuates the spikes produced by occasional errors in the feedback loop. In the case of severely corrupted symbols, since CRC provides a way to detect decoding errors, the channel estimate may be not updated, in order to prevent error propagation. This will still have an impact on fast fading channels, and in this case more sophisticated techniques could include time–domain filter over many past correctly decoded OFDM symbols. However, considering the low error rate that should anyhow characterize a good reception, in the cases of interest, error propagation has negligible effects. Numerical results provided in this chapter include CD3 with decoding and show that in the case of interest, error propagation is negligible.

In DVB–T2 with multiple input streams, when variable coding and modulation is adopted, each stream is associated to a given modulation and coding format. Moreover, within a super–frame, including multiple streams, the merger/slicer shall allocate the input–streams in decreasing order of robustness, e.g., first QPSK rate 1/2, last 256–QAM rate 9/10. In fact, to apply CD3 channel estimate correctly, the demodulation/decoding process must proceed continuously QEF from the beginning of the frame.

7.2.2 Reference Symbol in DVB–T2

In order to start the CD3 process, an initial channel estimate must be obtained. Indeed, DVB–T2 provides P2 symbols at the beginning of each frame, for signaling purposes. These OFDM symbols, of the same size as forthcoming data symbols, have additional pilots and provide accurate channel esti-

mate without the need of time interpolation over many OFDM symbols. The estimated channel on each pilot can then be interpolated on the frequency axis to data carriers and used to start CD3 process.

7.3 MISO Techniques to Improve Portable Reception

In order to improve system diversity, DVB–T2 provides the use of two antennas at the transmit station, while the receiver can still have a single antenna. A multiple input single output (MISO) Alamouti code [4] is used on the frequency axis. Let $x^{(a)}_{(n,k)}$ be the symbol on subcarrier k of OFDM symbol n transmitted on antenna $a = 1, 2$. From the data stream $x_{(n,k)}$ the transmitted symbols are organized as follows:

$$\begin{aligned} x^{(1)}_{(n,2k)} &= \frac{1}{\sqrt{2}} x_{(n,2k)}, \quad x^{(2)}_{(n,2k)} = \frac{1}{\sqrt{2}} x^{*}_{(n,2k+1)}, \\ x^{(1)}_{(n,2k+1)} &= \frac{1}{\sqrt{2}} x_{(n,2k+1)}, \quad x^{(2)}_{(n,2k+1)} = -\frac{1}{\sqrt{2}} x^{*}_{(n,2k)}, \end{aligned} \tag{7.6}$$

i.e. antenna 2 swaps odd and even cells of the OFDM symbol and introduces a complex conjugate and a sign change. Scaling by $\sqrt{2}$ ensures the same average power consumption with respect to the conventional one transmit antenna system. By indicating with $\{H^{(a)}_{(n,k)}\}$ the channel frequency response after OFDM demodulation relative to antenna a, the demodulated signal is

$$\begin{aligned} y_{(n,2k)} &= x_{(n,2k)} \cdot H^{(1)}_{(n,2k)} + x^{*}_{(n,2k+1)} \cdot H^{(2)}_{(n,2k)} + w_{(n,2k)}, \\ y_{(n,2k+1)} &= x_{(n,2k+1)} \cdot H^{(1)}_{(n,2k+1)} - x^{*}_{(n,2k)} \cdot H^{(2)}_{(n,2k+1)} + w_{(n,2k+1)}. \end{aligned} \tag{7.7}$$

At the receiver, combining of even and odd cells is performed, to obtain

$$\begin{aligned} z_{(n,2k)} &= G_{n,2k} \left[y_{(n,2k)} \cdot \hat{H}^{(1)*}_{(n,2k)} - y^{*}_{(n,2k+1)} \cdot \hat{H}^{(2)}_{(n,2k+1)} \right] \simeq x_{n,2k} + \nu'_{n,2k}\,, \\ z_{(n,2k+1)} &= G_{n,2k+1} \left[y_{(n,2k+1)} \cdot \hat{H}^{(1)*}_{(n,2k+1)} - y^{*}_{(n,2k)} \cdot \hat{H}^{(2)}_{(n,2k)} \right] \simeq x_{n,2k+1} + \nu'_{n,2k+1}\,, \end{aligned} \tag{7.8}$$

where $\nu'_{n,k}$ is the noise plus disturbance term and

$$G_{n,2k} = \frac{\sqrt{2}}{|\hat{H}^{(1)}_{(n,2k)}|^2 + |\hat{H}^{(2)}_{(n,2k+1)}|^2}, \quad G_{n,2k+1} = \frac{\sqrt{2}}{|\hat{H}^{(1)}_{(n,2k+1)}|^2 + |\hat{H}^{(2)}_{(n,2k)}|^2}. \tag{7.9}$$

If the channel is not changing on two adjacent cells of each OFDM symbols, then $\nu'_{n,k}$ depends only on noise.

7.3.1 Channel Estimate for MISO Transmissions with CD3

Decision directed channel estimation can be performed in the MISO case using (7.7), i.e.

$$\begin{aligned} \hat{H}^{(1)}_{(n,2k)} &= Y_{n,2k}\left[y_{(n,2k)} \cdot x^{*}_{(n,2k)} + y_{(n,2k+1)} \cdot x^{*}_{(n,2k+1)}\right], \\ \hat{H}^{(1)}_{(n,2k+1)} &= Y_{n,2k+1}\left[y_{(n,2k)} \cdot x_{(n,2k+1)} - y_{(n,2k+1)} \cdot x_{(n,2k)}\right], \end{aligned} \tag{7.10}$$

where

$$Y_{n,k} = \frac{2}{|\hat{x}_{(n,2k)}|^2 + |\hat{x}_{(n,2k+1)}|^2}, \quad Y_{n,2k+1} = \frac{2}{|\hat{x}_{(n,2k)}|^2 + |\hat{x}_{(n,2k+1)}|^2}. \tag{7.11}$$

Also in this case the assumption of slowly changing channel between two adjacent OFDM subcarriers is important to ensure a reliable channel estimate.

In correspondence of the reference symbol, in order to keep the overhead unchanged, symbols on odd subcarriers are transmitted at full power by the first antenna, while symbols on even subcarriers are transmitted by the second antenna. Channel estimation is carried out as in (7.4) and interpolation is needed to obtain channel estimate on all subcarriers for both channels.

7.4 Numerical Results

This section describes BER performance as a function of C/N, obtained through simulations of DVB–T2 in the presence of LDPC channel codes (long block length of 64,800 bits), for $N = 8,192$ (8k) and $N = 16,384$ (16k). The examined constellations are 16–QAM, 64–QAM and 256–QAM, and the code rates are 3/5, 2/3, 5/6, 8/9. Different (static) selective channel models have been simulated, according to [2]:

1. AWGN
2. Rice channel (fixed reception)
3. Rayleigh channel (portable reception).

We also considered the six–taps typically urban (TU6) channel model for the evaluation of multiantenna systems. These models were considered as they are the *de facto* standard channels in the digital video broadcasting community, in order to ease performance comparison. In our setting the reference symbol is inserted periodically once every 120 (60) useful OFDM

symbols for the 8k (16k) mode, i.e. in time by about 120 ms plus the GI. The reference OFDM symbol is a pseudo–random sequence of BPSK symbols. Channel estimation via CD3 is included in the simulations.

For DVB–T2, QEF condition is assumed to be achieved when $BER = 10^{-6}$ is obtained at the output of the LDPC decoder, assuming that a BCH code will further reduce the BER thereafter, while for DVB–T, QEF is achieved at $BER = 2 \cdot 10^{-4}$ at the output of the convolutional decoder [2], as Reed–Solomon code (RS) will remove most of the residual errors.

For fixed reception via directive antennas the reference available C/N is from 18 to 24 dB and the bandwidth efficiency of interest around 4–6 bit/s/Hz. Therefore the reported performance concentrates on 64–QAM and 256–QAM constellations.

7.4.1 Single Transmit Antenna

Fig.s 7.4 and 7.5 show the BER vs C/N at the output of the LDPC decoder for three code rates, 64–QAM and two channel models.

For both fixed and portable channels we observe the typical waterfall curves of LDPC codes and, as expected, the portable channel requires a higher C/N to achieve the same BER of the fixed channel. The regular behavior of the curve reveals also that error propagation due to CD3 channel estimation using past decoded bits has almost no impact on the system performance.

We compared also the performance of CD3 with a genie DVB–T2 system having ideal channel estimation (ICE) available. The comparison is reported in Table 7.1, where 64–QAM is considered and applied to various channels and code rates scenarios.

	code rate	C/N [dB] ICE	C/N [dB] CD3	Rate [Mbit/s] GI 1/8	Rate [Mbit/s] GI 1/64
AWGN	8/9	18.3	19.1	35.06	38.83
	5/6	16.9	17.6	32.84	36.38
	2/3	13.9	14.5	26.26	29.08
fixed	5/6	18.1	18.9	32.84	36.38
	2/3	14.8	15.5	26.26	29.08
portable	5/6	22.7	23.5	32.84	36.38

Table 7.1: Performance of DVB–T2 with $BER = 10^{-6}$ after the LDPC decoder. 64–QAM. Ideal channel estimation (ICE) and CD3.

In particular, we compare the required C/N in order to achieve QEF

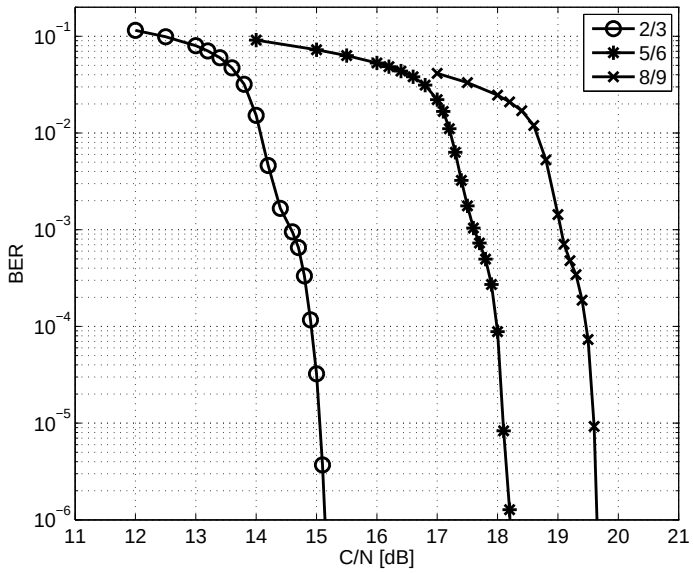

Figure 7.4: BER at the output of the LDPC decoder as a function of C/N for various code rates. Rice channel (fixed reception).

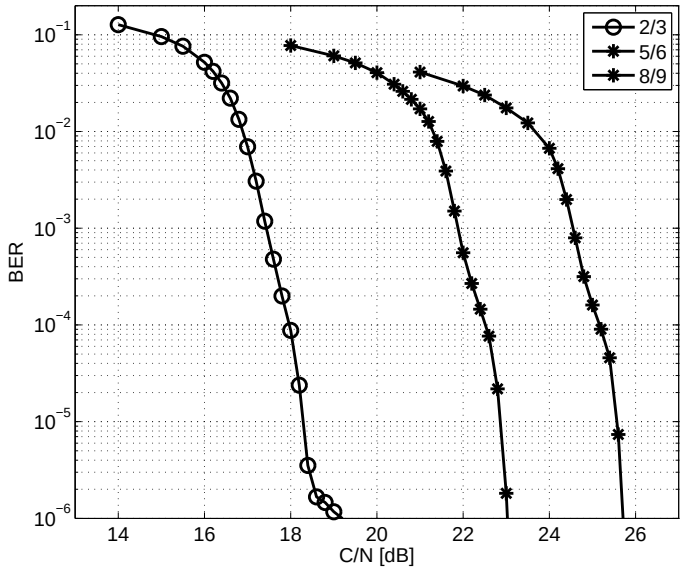

Figure 7.5: BER at the output of the LDPC decoder as a function of C/N for various code rates. Rayleigh channel (portable reception).

conditions. From the table we observe that, due to the residual noise in the channel estimate that increases the overall noise on the equalized signal, CD3 increases the C/N requirement of about 0.7 dB with respect to ICE. Hence, the advantage of CD3 is in using fewer pilots with a saving, in SFN, of about 8 % in achievable throughput for carrier which are otherwise dedicated to pilots boosted by 2.5 dB. For a comparison, Table 7.2 shows instead the required C/N to achieve QEF condition for a 64–QAM constellation and assuming ICE.

	code rate	C/N [dB]	Rate [Mbit/s] GI 1/4	Rate [Mbit/s] GI 1/32
AWGN	5/6	19.3	24.88	30.16
	2/3	16.5	19.91	24.13
fixed	5/6	20.0	24.88	30.16
	2/3	17.1	19.91	24.13
portable	5/6	25.3	24.88	30.16
	2/3	19.3	19.91	24.13

Table 7.2: Performance DVB–T with $BER = 2 \cdot 10^{-4}$ after the convolutional decoder. 64–QAM. Ideal channel estimation.

Since the OFDM block size of the proposed system is double with respect to that of DVB–T, for a fair comparison we halved the duration of the GI of the new system with respect to DVB–T. By comparing Tables 7.1 and 7.2 we note that for the same required C/N, DVB–T2 CD3 allows for higher code rates, obtaining a gain over DVB–T well above 30% in terms of bit rate. This is the result of i) a longer OFDM symbol duration for a given GI duration, ii) improved coding gain and iii) reduced number of pilots for channel estimation. Indeed, DVB–T2 with ideal channel estimate exhibits a C/N gain of about 3 dB with respect to DVB–T and the loss incurred by CD3 is well within the gain due to improved coding and higher spectral efficiency. Therefore, we can conclude that CD3 allows the deployment of DVB–T2 on existing DVB–T networks providing a higher rate while ensuring the same (or even better) coverage. See Table 7.3 for a summary of performance.

We also simulated the performance of the various systems with other code rates and constellation sizes (spanning from 16–QAM to 256–QAM) and the same comparative conclusions as before can be driven. In particular, for static portable reception via set–top antenna CD3 achieves 21.89 Mbit/s with a C/N of 16.6 dB with a GI of 1/8 and code rate 5/6.

Lastly, we have also investigated the possibility of trading off a lower coding gain for a higher constellation size. However, we have observed that

	config	Thr. (Mbit/s)		Thr. inc.
		DVB–T2 CD3	DVB–T	
Fixed indoor rec. (external ant.)	MFN	36	24	50%
	SFN	33	20	65%
Portable indoor rec. (Rayleigh ch.)	MFN	24	18	34%
	SFN	22	15	46%

Table 7.3: Summary of performance comparison

we do not obtain an advantage in terms of bit rate and C/N on the AWGN and fixed channels. For example on AWGN, 256–QAM 3/5 achieves 35.01 Mbit/s at C/N of 21.9 dB, which is similar to 64–QAM 5/6, as from Table 7.1. On the other side, a better performance is achieved for denser constellations on the Rayleigh channel, due to the more powerful code which allows better recovery in the presence of fading. In this case, 256–QAM 3/5 achieves 31.47 Mbit/s with a C/N of 21.9 dB.

7.4.2 Multiple Transmit Antennas

For performance with multiantenna systems, we considered two transmit antennas and a single receive antenna, with independent TU6 channels on each link. In our setting Alamouti code is applied on the time axis. Fig.s 7.6, 7.7, 7.8 show the average BER as a function of the average C/N, defined as

$$C/N = \mathsf{E}\left[\frac{\sigma_x^{(i)2}|H_{(n,k)}^{(i)}|^2}{\sigma_w^2}\right], \tag{7.12}$$

where $\sigma_w^2 = \mathsf{E}[|w_{(n,k)}|^2]$ is the noise power and $\sigma_x^{(i)2} = \mathsf{E}[|x_{(n,k)}^{(i)}|^2]$ is the power of the transmitted signal, satisfying $\sum_{i=1}^{N_T} \sigma_x^{(i)2} = 1$. Both single antenna and two antenna transmissions use the same transmit power. We observe a spatial diversity gain of the two antennas system with respect to the conventional single antenna system.

7.4.3 Computational Complexity and Conclusions

The complexity of CD3, when applied to DVB–T2, stems from the channel estimation performed at each OFDM symbol. For each active subcarrier, (7.4) requires a complex division, while subsequent frequency smoothing with a low pass filter requires L_f complex multiplications (CMUX) per active

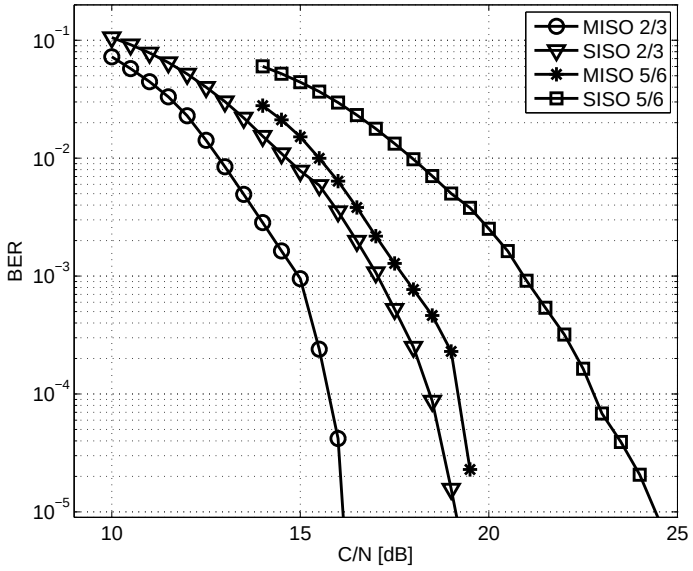

Figure 7.6: Average BER at the output of the LDPC decoder as a function of C/N for various code rates. Two transmit antennas and single receive antenna with Alamouti code. TU6 channel, 16–QAM.

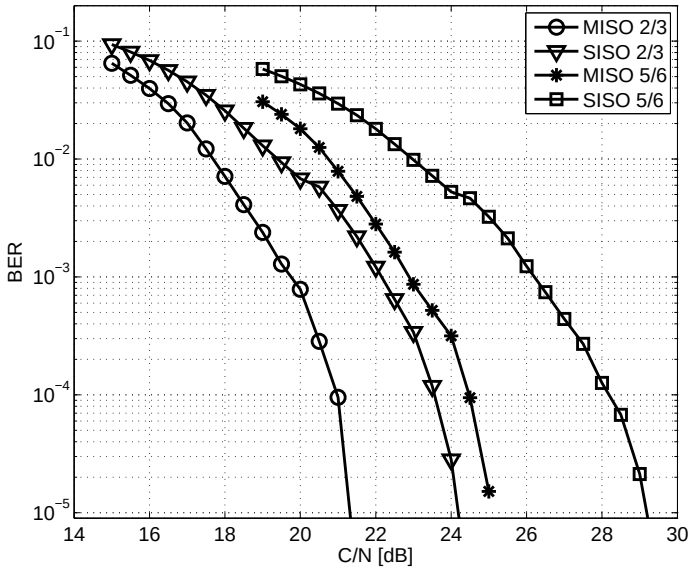

Figure 7.7: Average BER at the output of the LDPC decoder as a function of C/N for various code rates. Two transmit antennas and single receive antenna with Alamouti code. TU6 channel, 64–QAM.

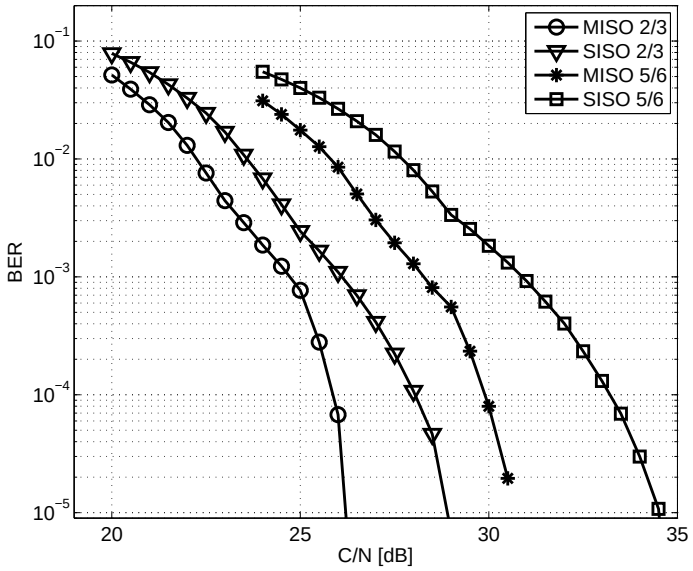

Figure 7.8: Average BER at the output of the LDPC decoder as a function of C/N for various code rates. Two transmit antennas and single receive antenna with Alamouti code. TU6 channel, 256–QAM.

subcarrier, where L_f is the number of filter taps. Assuming that a complex multiplication has the same complexity of a complex division, for a typical value of $L_f = 6$ we obtain a complexity of $L_f + 1 = 7$ CMUX per active subcarrier. For a conventional OFDM channel estimation, we need a) a complex division for each pilot carrier, b) time interpolation, c) frequency interpolation/smoothing. Considering a time interpolator with 4 taps, a frequency interpolator with 6 taps, and pilot spacing of 12 subcarriers, we obtain a complexity of $6 + 4/12 = 6.3$ CMUX per active carrier. Hence, we conclude that CD3 has a complexity similar to that of existing OFDM channel estimation techniques.

In conclusion, simulation results for the proposed technique both with a single transmit antenna and with two transmit antennas, show that CD3 allows to achieve high data rate transmission even for channel with fast time variations and severe frequency selective fading. The high data rate is made possible by both the efficient coding schemes of DVB–T2 and the ability of CD3 to track channel variations with a reduced overhead of pilot symbols with respect to conventional channel estimation techniques for OFDM. When compared to DVB–T, DVB–T2 CD3 yields a significant achievable throughput improvement allowing DVB–T2 to provide high definition TV.

Chapter 8

Multiple Users Discovering

All previous chapters present broadcast communication solutions. This chapter instead deals with cellular system scenario and present the most important research results obtained during my internship in Qualcomm Flarion Technologies (QFT). In QFT, I belonged to a research group involved to design and develop of new generation wireless technologies. My research topics in QFT were again related to the physical layer problems. Some interesting results has been obtained regarding multiple access scenarios, specifically on channel estimation and joint decoding on interferent signals. Here we present a summarize of the most interesting results.

The idea here concerns to find a good solution for multiple user discovering in high dense networks. Consider a wireless system composed by a synchronous network with nodes drawn uniformly at random on a finite surface. Each node or *peer* represents a wireless device. Assume that there is a time slot where each node broadcasts its *identifier*, i.e., an identification signal encoded transmitted using some *physical resource*. This classical synchronous *peer discovery* problem becomes of practical and theoretical interest in a dense scenario where the number of available physical resources is much smaller than the total number of node identifiers required by the system.

Driven by physical layer requirements for implementing a wireless network, this chapter addresses link level strategies for single–hop peer discovery. In a classical way we assume that peer identifiers are broadcasted via coded transmission. In order to obtain an operational measure of the system performance, we choose to evaluate the *average number of peers discovered by a typical node*. To this purpose, we present a multiuser joint decoding solution in an OFDM multiple access wireless channel based on the factor graph representation and the message–passing algorithm in order to discover the nodes in the network. This is a suboptimal solution based on a parallel system of independent decoders. In a particular case we derive a variation

which is optimal. This is based on a single superframe decoder (Super Trellis). We also present any channel estimation solution for both time–invariant and frequency offset time–variant channel that fit in a dense network scenario.

These strategies are investigated through system simulations: it is shown that, in general, joint iterative coding achieves impressive gains, in particular if compared with alternative strategies such as interference cancellation. Moreover, although identifiers use typically short codelengths, joint iterative receivers permit to achieve an overall system performance close to the one of joint optimal receivers while keeping the computational complexity low. Elementary modifications of broadcasted signals such as symbol permutations permit to further improve the system performance. Finally, this work suggest that the the overall system design shall be tailored in order to take full advantage of the joint decoding ability while using some form of congestion control to trace the number of interfering signals.

8.1 Overview

Recently, we have observed a strong growth of interests towards ad hoc wireless networks including wireless sensor networks and mesh networks in the unlicensed spectrum. Meanwhile, there have been various proposals to introduce direct mobile to mobile communications in the traditional cellular spectrum. In these networks, the discovery of neighboring information has been one of most important functions to have for various reasons. For example, in sensor networks, a node needs to exchange routing information with the one–hop neighbors to establish a good route to the sink. In addition, it is in general very important to know the identities of the neighboring node so that a node can know if there is anything interesting around in any ad hoc network, where no central authority can offer that information. We refer to this process as the *peer discovery* process.

In a synchronous system, one natural way to do peer discovery is to exploit the broadcast nature of the wireless channels. Consider a slotted time model and assume that there are a total of K time slots allocated for the purpose of peer discovery. Each user can pick a slot to transmit their identity information when they join the network and listen to the rest of the $K-1$ time slots to discover the peer information from its peers. Apparently, when more than 1 users pick the same slot to transmit, some nodes will see a collision and can not decode either of the transmitted signal in this time slot. In this chapter, we discuss ways to improve the decoding performance in this setup by introducing multi–user decoding capabilities to the nodes.

Multiuser detection [71] is a classical problem in wireless communications. In a multiple–access channel (MAC) scenario, where several devices transmit simultaneously, multiuser detection is accomplished under opportune hypothesis in [72]. A classical way to decode all the transmitted codewords is to do successive interference cancellation [73, 74], where a strong codeword is decoded first by taking the rest codewords as interference and removed from the received signal, then the second codeword can be decoded. It is well known that SIC combined with *time sharing* is capacity achieving for a Gaussian MAC channel. However, for the setup we are considering, since everyone is transmitting at a fixed rate, the concept of time sharing does not apply and SIC can be highly sub–optimal. However, SIC has its limitations as compared to optimal joint decoder:

(*i*) SIC is not robust in the presence of channel estimation error;

(*ii*) SIC has strong requirements on the difference between the strong user and the weak user.

In view of this, we consider multiuser joint decoding based on graphical models and belief propagation algorithms, which is a standard exercise in modern coding theory [62, 75–78]. A non–exhaustive list of works dealing with joint channel estimation and symbol decoding includes [79–90]. Iterative coding for interference channels is investigated and analyzed in details for the MAC with binary inputs and additive white Gaussian noise in [91, 92]: large blocklengths approach the channel capacity. In this chapter, we focus of short blocklengths and investigate the network performance assuming that each peer implements joint decoding. We consider multiple multi–user decoding techniques including SIC and joint iterative decoding (JID). We simulate the system impact of such joint decoding schemes on the performance of peer discovery in terms of the average of peers one can discover in the network. We show that significant system gains can be achieved with JID as compared to conventional single user decoders in a typical wireless ad hoc network deployment.

Although the core ideas are presented using a simplified framework, this work is driven by considerations that emerged from the implementation of a real peer–to–peer wireless network. We assume that distributed communications in the network follow certain protocols and that there is a key phase where each peer discovers neighboring nodes. The *peer discovery* phase is common with many wireless network implementations. It is particularly crucial and challenging in the high–density case, i.e., when the number of available physical resources is far less than the number of peers. Practical solutions consist first in *orthogonalizing* the identifier resources in a distributed

manner so that a *maximal number of peers can be typically discovered.* Several strategies can be devised at the level of the system: time, space, or position sharing; congestion control; beamforming; etc. *System level* techniques are generally complementary to *link level* techniques. Link level techniques form a second set of practical solutions: those are algorithms that are implemented at the physical layer of each node for identifier transmission and reception. In this chapter, we focus mainly of the link level and investigate joint detection/decoding of multiple peers. The information theoretic problem is non–coherent coding for the multiple–access channel. It shows a practical study: it is illustrated by simulations. Notice that the peer discovery performance is measured by the *typical number of discovered peers* because, in a system that is sufficiently distributed to be fair, it is the key operational performance measure.

8.1.1 Wireless Environment and Typical Topology

This work is directed towards the implementation of a real wireless network. Our design and model choices originate from practical but natural constraints issued from actual wireless techniques. The considered model remains fairly general. We think it brings new insights for peer discovery implementations.

Link Level: We assume that a given identifier is transmitted using a signal located on a single carrier frequency. The point–to–point channel is therefore modeled as a non–coherent standard Rayleigh fading channel with additive Gaussian noise. Two fading models are considered: time–invariant fading and frequency offset time–varying fading.

System Level: An important characteristic for the network behavior is the propagation model and the distribution law of the nodes on the plane. Recall that the peers are positioned uniformly at random through a square of a given size, and consider a standard propagation model of the electromagnetic waves with $\alpha \in [2, 8]$. Under those circumstances, simulations show that, unsurprisingly, the following remarkable topology appears: the network forms a regular lattice structure in the typical case. While general proofs and concentration statements for this phenomena, which are particularly interesting if the network is dynamic, are the subject of additional works,the regular lattice asymptotic average structure is an important justification for this chapter. It indicates that peer discovery can be reduced to a joint decoding of a *finite* number of neighboring peers whose signal is above a certain power threshold.

8.2 Communication Model

8.2.1 System Level

We consider a wireless network with N nodes and, initially, K physical resources available for the purpose of peer discovery. In a real system, K is chosen is such a manner such that

(i) the total number of resources occupy a small overhead of the system;

(ii) each resources contain enough degrees of freedom to transmit the typical identifier of a node. In the example presented here, each resource contains 72 symbols and each node can transit 64 bits over these symbols.

Every node picks a physical resource when it powers up and transmits its identifier on that resource. Each node also listens in the rest $K-1$ resources to discover its neighboring nodes. We assume that the same code book is used at all transmitters.

In this chapter, we mainly consider the deployment scenario where the number of nodes in a neighborhood exceeds the total number of resources K assigned for the purpose of peer discovery. In this case, multiple nodes may collide on a single physical resource. A direct consequence here is that for some nodes in the network, some of the resources become not decodeable and we see a performance degradation in the network. We are interested in the system level performance improvement after introducing joint (multi–user) decoding at the receivers. Particularly, we are interested the metric of *average number of discovered peers* across the network.

One important factor which have a direct impact on the system performance is how each node picks the resource when it joins the network. This is an interesting topic of its own and is not the main focus of this work. To provide a fair comparison between different decoding schemes, we use the *minimum energy* rule for picking the resources. That is, a mobile picks the resource with the smallest energy on it as its peer discovery resource at power up. This is a distributed *greedy* algorithm which tries to maximize the distance between two nodes sharing the same resource. All the system level simulations and comparisons will be based on this scheme for picking the resource, but with different decoding algorithms.

8.2.2 Link Level

We consider a discrete–time synchronous wireless system where we focus on a single tone narrow–band channel of a OFDM communication system. We

use modern wireless techniques and assume the channel to be memoryless and described by a single tap per time sample. We investigate a generic scenario characterized by single–antenna transmitters. Here we assume that n transmitters are sharing the same multiple–access channel and they have to be detected by a device with m receiving antennas. The link model is

$$\mathbf{y}(t) = \mathbf{H}(t)\mathbf{x}(t) + \boldsymbol{\nu}(t) \tag{8.1}$$

where $\mathbf{y}(t) \in \mathbb{C}^m$ is the signal vector received at time sample t, $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_n]$ is a complex matrix such that $\mathbf{h}_i = [h_{1i}, \ldots, h_{mi}]^{\mathrm{T}} \in \mathbb{C}^m$ represents the channel complex coefficient between the i–th transmitter and the m receiver antennas, $\mathbf{x} = [x_1, \ldots, x_n]^{\mathrm{T}}$ are the symbols transmitted by the n transmitters. They belong to an M–ary alphabet $x_i \in \mathcal{A}$. Finally $\boldsymbol{\nu} \sim \mathcal{CN}(\mathbf{0}, N_0\mathbf{I}_m)$ is the vector of the additive white Gaussian noise (AWGN) components at the receiver antennas.

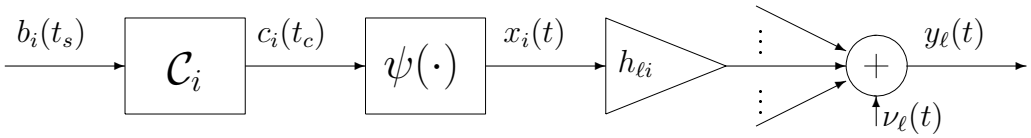

Figure 8.1: A transmitter–receiver branch of the system.

As depicted in Fig. 8.1 the encoder $\mathcal{C}_i$ provides a channel coding for the i–th transmitter. We assume that synchronization insures that the user codewords are aligned in time. For this purpose we consider all the codebooks having the same size parameters: given K_b source bits in input they return N_b coded bits: the *coding rate* is $R = K_b/N_b < 1$. The map function $\psi(\cdot)$ takes a group of $q = \log_2 M$ coded bits and map them in a complex value taken from the alphabet $\mathcal{A}$.

Each transmitter has a sequence of independent equidistributed source bits $\overline{b}_i$ at rate $t_s \in \mathbb{Z}(RT/q)$ that are coded into the bit sequence $\overline{c}_i$ at rate $t_c \in \mathbb{Z}(T/q)$. This sequence is mapped into the user codeword $\overline{x}_i$, of length $L = N/q$, and transmitted at the system rate $t \in \mathbb{Z}(T)$.

At the receiver two situations can happen: *coherent* detection when channel state is known ate receiver, *non–coherent* detection when the channel parameters have to be estimated from some additional information.

8.3 Receiver Algorithms: Coherent Case

In complement to resource orthogonalization, it may be of interest for each node to jointly discover several peers. In this section, we assume the multiple–

access channel to be coherent, i.e., the fading coefficients are known by the receiver.

8.3.1 Successive Interference Cancellation

Classical implementations are usually based on interference cancellation[1] (IC), see, e.g., [73, 74]. IC is motivated by information theoretic considerations and signal Gaussianity is assumed [72].

In general real systems do not fulfill this hypothesis and IC may not be fully rigorous. Although IC and successive IC (SIC) are not directly derived from optimal rules, their low–complex implementation works reasonably well in practice. IC is a very convenient information theoretic tool and also an attractive choice for multi–user decoding algorithm implementation due to its low complexity. However, IC has a strong constraint on the received signal strength disparity of the multiple codewords, which might lead to suboptimality in the scenario considered here. For example, we consider a convolutional codeword which requires a SINR of 4 dB to have a small enough decoding error probability. Applying IC requires the received codewords to be at least 4 dB apart in terms of received energy, so that the first codeword can be decoded by taking the second one as interference. However, in principle, an *optimal* joint decoder would just require both signals to be strong enough against noise without too much dependency on the signal disparity between the two codewords. In the sequel, we will mainly use use (S)IC as a reference against which we shall compare "more modern" alternatives such as iterative receivers [62, 78].

8.3.2 Optimal Decoding Rule

Before deriving low–complexity alternatives to SIC (such as the more efficient but still suboptimal JID), let us start with the optimal decoder. We shall later use it as a benchmark.

The optimal decision rule is the maximum a posteriori (MAP) decision rule. Without loss of generality and to make the connection with JID, let us present the MAP estimator in the context of local (i.e., *symbol* or *bit*) optimality. Let $\hat{x}_i(t)$ denote the MAP estimation associated with the particular

[1] Recall that IC is based on the detection of the strongest signal $x_i(t)$ considering the others as noise. The strongest signal is further subtracted from the received signal and the procedure can be iterated to become successive IC (SIC).

i–th transmitted for the time t. The optimal decision rule reads

$$\begin{aligned}\hat{x}_i(t) &= \arg\max_{x_i(t)\in\mathcal{A}} p(x_i(t)|\overline{\mathbf{y}}) \\ &= \arg\max_{x_i(t)\in\mathcal{A}} \sum_{\overline{\mathbf{x}}\sim x_i(t)} p(\overline{\mathbf{x}},\overline{\mathbf{y}}),\end{aligned} \tag{8.2}$$

where the notation $\sum_{\sim\alpha}$ means that we are summing over all variables but α. See also [62, 75].

Eq. (8.2) shows that brute force implementations of the optimal decoder require enumerating an exponential number of codewords. This is computationally prohibitively complex. When needed in practice, it is replaced by approximation methods such as SIC. There are however some cases for which joint MAP decoding is implementable with reasonable complexity. Those are mainly when the computation can be performed on a small trellis product.

(i) First, this may be a practical solution whenever the product of the code memory and the number of considered users is sufficiently small. The critical limitation of this method is the complexity. If we are using a CC with memory Γ, while a normal decoder has a trellis dimension equal to 2^Γ, the corresponding trellis product has $2^{n\Gamma}$ states. It means that the complexity grows exponentially with the number of simultaneous transmissions we are going to jointly decode. Adaptive coding combined with network density control could be implemented in a distributed way at the system level to guarantee this constraint. This makes this solution potentially valid in certain practical cases.

(ii) Second, this permits us to compare the optimal receiver with suboptimal receivers based on SIC or, more efficiently, on graphical models and JID.

8.3.3 Optimal Joint Decoding on the Trellis Product

For presentation simplicity, let us first assume that each node broadcasts words from the same codebook

$$\mathcal{C}_1 = \ldots = \mathcal{C}_n = \mathcal{C}. \tag{8.3}$$

With this aim each node use a convolutional encoder with rate $= 1/q$ with one source bit per transmitted symbol.[2] A standard exercise on probabilistic

[2]Notice that no symbol interleaving after the encoder is used in this simple scenario.

coding shows that each term of the sum in Eq. (8.2) factorizes as

$$p(\boldsymbol{\sigma}(0)) \prod_t p(\mathbf{y}(t)|\mathbf{x}(t)) p(\mathbf{x}(t)\boldsymbol{\sigma}(t)|\mathbf{b}(t)\boldsymbol{\sigma}(t-1)) p(\mathbf{b}(t)), \tag{8.4}$$

assuming that each peer broadcasts an independent signal chosen uniformly at random. Fig. 8.2 represents the corresponding *factor graph*. We further use standard sum–product message–passing rules from belief propagation to implement a BCJR–style decoding on this graphical model.

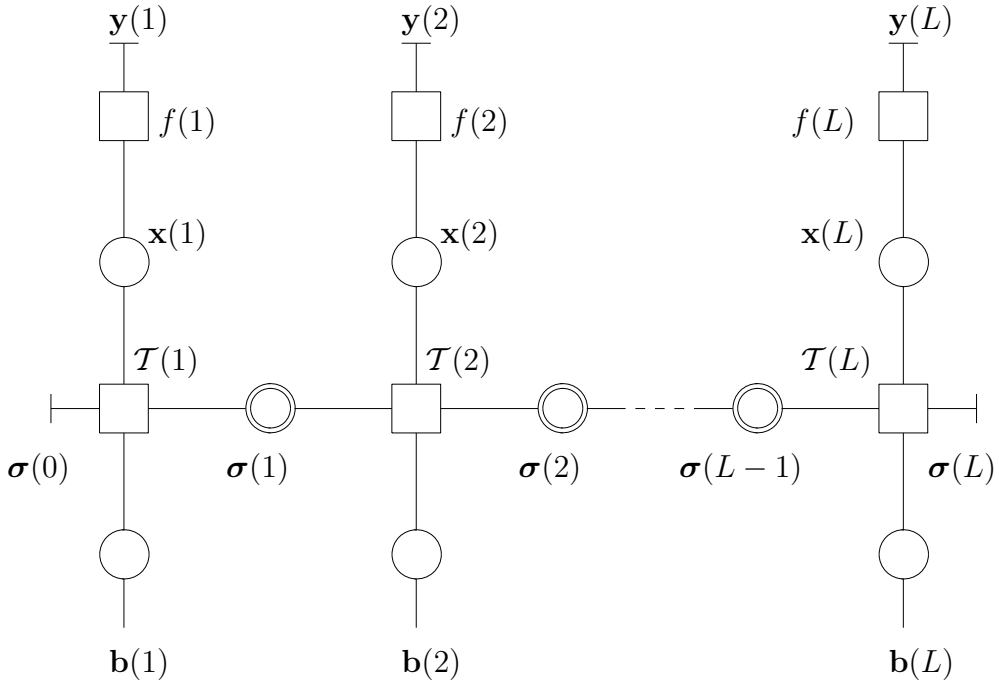

Figure 8.2: Factor graph for ST decoding.

In Fig. 8.2 MAP estimates on the bits, symbols, and corresponding joint estimates can be obtained from belief propagation. Here

$$f(t) = p(\mathbf{y}(t)|\mathbf{x}(t)) \propto \exp(-|\mathbf{y}(t) - \mathbf{H}(t)\mathbf{x}(t)|^2/N_0) \tag{8.5}$$

since the complex–valued noise is AWGN. The main characteristic of the trellis product represented here are the function nodes that involve the whole state space

$$\mathcal{T}(t) = p(\mathbf{x}(t)\boldsymbol{\sigma}(t)|\mathbf{b}(t)\boldsymbol{\sigma}(t-1)), \tag{8.6}$$

where the vector $\mathbf{x}(t)$ is represented as a single variable node in the factor graph and where the joint vector of the state variables

$$\boldsymbol{\sigma}(t) = [\sigma_1(t) \dots \sigma_n(t)]^{\mathrm{T}} \tag{8.7}$$

is represented by a single variable node as well the source bits

$$\mathbf{b}(t) = [b_1(t) \dots b_n(t)]^{\mathrm{T}}. \tag{8.8}$$

Recall that belief–propagation–like algorithms are optimal on cycle–free realizations. It is well–known, see [62,76], that the message–passing rules will translate into the BCJR algorithm [93] in the case of Fig. 8.2. Alternatively it can be translated into the Viterbi algorithm if we choose block optimality. In order to emphasize on the fact that we are processing the trellis product associated with n peers, we call *Super Trellis* (ST) decoder the iterative receiver associated with Fig. 8.2.

8.3.4 Joint Iterative Decoding

Let us further follow standard receipts from modern coding and derive a suboptimal but efficient turbo–like decoder. Consider now again n codes $\{\mathcal{C}_i\}_{i\in\{1,\cdots,n\}}$. Each term of the sum in Eq. (8.2) factorizes simply as

$$\begin{aligned}
\hat{x}_i(t)^{\mathrm{MAP}} &= \arg\max_{x_i(t)\in\mathcal{A}} \sum_{\sim x_i(t)} p(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \\
&= \arg\max_{x_i(t)\in\mathcal{A}} \sum_{\sim x_i(t)} p(\overline{\mathbf{y}}|\overline{\mathbf{x}}) p(\overline{\mathbf{x}}) \\
&= \arg\max_{x_i(t)\in\mathcal{A}} \sum_{\sim x_i(t)} \prod_t p(\mathbf{y}(t)|\mathbf{x}(t)) \prod_i p(\overline{x}_i \in \mathcal{C}_i).
\end{aligned} \tag{8.9}$$

As usual, we can use the sum–product message–passing rules to perform iterative decoding on this, in general non–cycle–free, factor graph. It is another standard exercise to derive a factor graph which is shown in Fig. 8.3.

The suboptimality of the message–passing algorithm can be seen from the factor graph since it presents several loops. Because this receiver is our main object of interest, let us go into more details and examine the different type of messages.The up–message $\mu_{\mathcal{C}_i\to t}(x_i(t))$ depends on the code we are using. In general it is the extrinsic symbol estimate $\mathcal{E}_{\mathcal{C}_i}$ provided by the soft decoding of $\mathcal{C}_i$

$$\mu_{\mathcal{C}_i\to t}(x_i(t)) = \mathcal{E}_{\mathcal{C}_i}(\{\mu_{f(\tau)\to i}(x_i(\tau))\}_{\tau\neq t}). \tag{8.10}$$

The down–message is

$$\mu_{f(t)\to i}(x_i(t)) = \sum_{\sim x_i(t)} p(\mathbf{y}(t)|\mathbf{x}(t)) \prod_{j\neq i} \mu_{\mathcal{C}_j\to t}(x_j(t)). \tag{8.11}$$

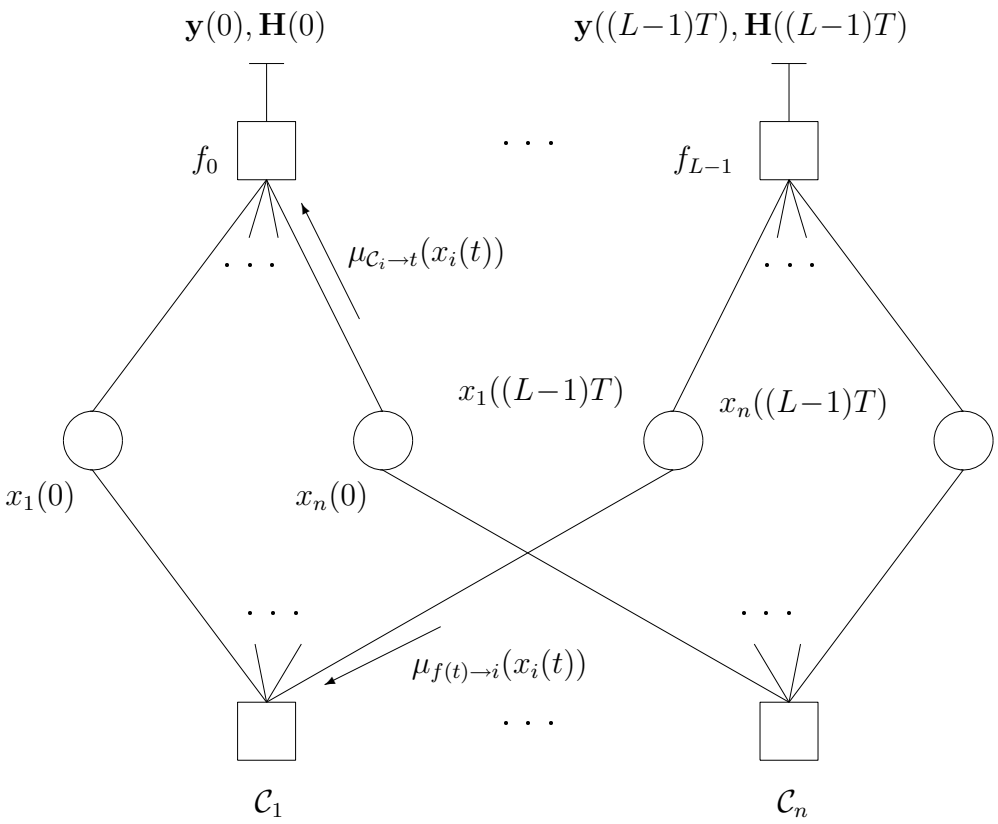

Figure 8.3: Factor graph for the Joint Iterative Decoding.

At this point the algorithm as to be implemented iteratively, for the sake of this reason it is called *Joint Iterative Decoding*. At the beginning we initialize the up–message to be uniform, i.e. no knowledge about the transmitted symbol statistic. For each transmitter i and for each time t we assume

$$\mu_{\mathcal{C}_i \to t}(x) = \frac{1}{L}, \quad \forall x \in \mathcal{A}. \tag{8.12}$$

We compute the down–message by the rule (8.11) then normalize such as

$$\sum_{x \in \mathcal{A}} \mu_{f(t) \to i}(x) = 1. \tag{8.13}$$

This message is used both to compute the source bit decoding, according to the actual decoding strategy, in order to get a decision over $\overline{b}_i$ and update the up–message for the next iteration.

It is intuitive to think that a proper choice of component codes $\mathcal{C}_i$ will increase the girth of the graphical model. This is implicit in construction with large codelengths presented in [91, 92, 94].

The different decoders associated with the n codes are represented by function nodes. For instance, if we use $n = 2$ and convolutional codes (followed by interleaving) for the two peers, the graphical model of the receiver is very similar to the one associated with classical turbo codes [94]. In this case the $n = 2$ function nodes represent 2 interconnected trellises. See [91,92] for the case of LDPC codes.

Recall now one of the main constraints of peer discovery scenarios. Peer identifiers are typically encoded using short to moderate blocklengths (e.g., 128 bits). Therefore, differently to the target of [91], we are not interested in the asymptotic regime of large blocklengths: standard techniques such as density evolution [95] or EXIT approximated analysis [96] are therefore not of first relevance here. Moreover the operational error probability point is typically 10% or 1% and asymptotic optimization is not of first importance. In this context it seems therefore reasonable to think of classical codes such as convolutional codes as a proper code design. Beside implementation simplicity, the induced joint coding system is similar to a turbo code, hence (in the case of convolutional codes concatenated with interleavers) it (relatively surprisingly) forms a strong joint code and turns out to be extremely efficient.

The implemented algorithm is called *joint iterative* decoding (JID). In our implementations, the typical number of iterations depends on the codebooks we are using. A typical number is 2 if we use CCs with no (or same) interleaver for all transmitters. If we use different codebooks (interleavers), JID typically requires at least 5 iterations.

8.3.5 JID using Convolutional Codes

While a "good" measure of code performance is the ability to perform close to Shannon limit, it is only valid for very large codelength. Pseudo–random structures such as LDPC or turbo codes [94,97], or recent academic constructions such as polar codes [98] would of course be extremely efficient in this context. For short to moderate blocklengths, algebraic codes or convolutional codes (CC) are still better adapted. In the context of iterative coding, CCs and their underlying Markov chains are common and convenient in practice.

We use CCs with rate $1/q$ where one source bit corresponds to q coded bits and 1 complex–valued transmitted symbol. In order to increase the code rate, a symbol puncturing can be applied such that $K_b/N_b > 1/q$. Typically, $q = 2$, and QPSK signaling is used. A simple way to obtain different codebook $\mathcal{C}_i$ starting from the same CC design is to apply different symbol interleavers Π_i to the complex–valued codewords obtained from the convolutional encoder.

The left side of Fig. 8.4 is related to a transmitted symbol and the right one is related to a punctured symbol where the symbol is completely

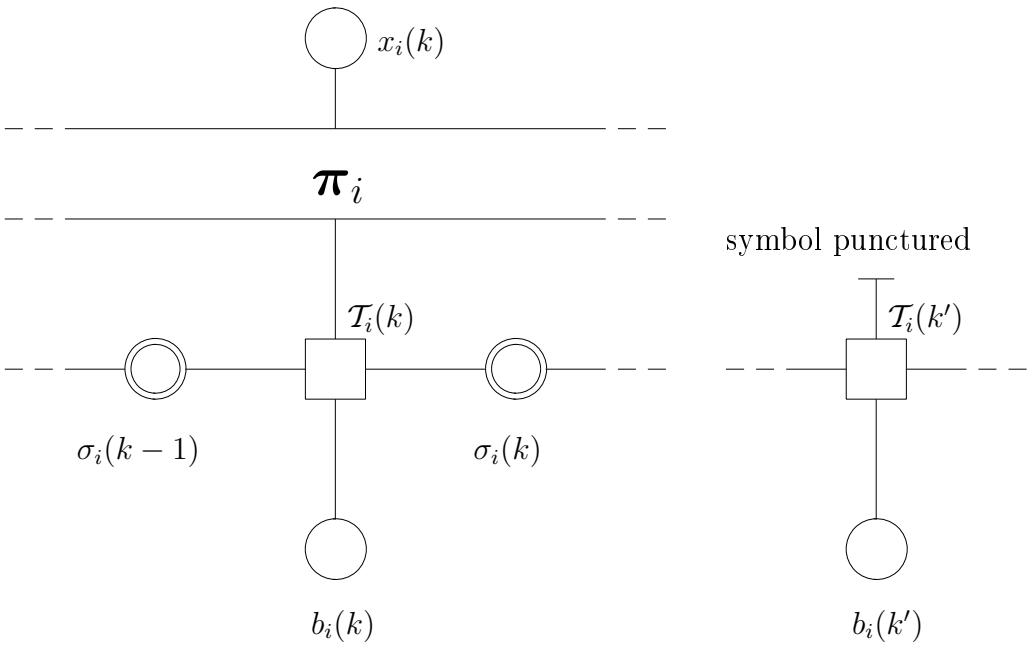

Figure 8.4: Factor graph element for local BCJR decoding.

unknown by the receiver then its *a priori* information is described by an uniform random variable. The node $\sigma_i(k)$ represents the state variable at time k of the i–th encoder. If the code $\mathcal{C}_i$ has memory Γ_i, this variables take 2^{Γ_i} values.

Eq. (8.9) further factorizes in the case of CCs and, in a standard manner, we get a turbo–style factor graph representing interconnected trellises. Fig. 8.4 depicts the sub–factorization of the factor nodes representing the trellis of CC $\mathcal{C}_i$. Similarly as in the trellis product case, the function $\mathcal{T}_i(k)$ represents the transition function from the time $k-1$ to k. It is the indicator function

$$\mathcal{T}_i(k) = p(x_i(k)\sigma_i(k)|b_i(k)\sigma_i(k-1)) \tag{8.14}$$

that is 1 if the four variables represent an allowed transition, otherwise is 0.

Notice that the complexity of each iteration of JID is given by the local BCJR decoding, i.e., it is roughly twice, i.e., comparable to, the complexity of one IC round implementing classical Viterbi [99] decoding.[3]

[3]Recall that standard Viterbi decoding returns the MAP or *maximum likelihood*, in case of equal priors, sequence and no further "soft" information.

8.3.6 Preliminary Comparison: ST vs JID vs SIC

It is first remarkable to observe that, even if the graphical model has many short cycles, JID performs extremely close to the optimal performance obtained through ST decoders. This is for example illustrated in Fig. 8.5 where the three methods are compared at medium blocklengths. Notice that SIC performs noticeably worst than JID whereas JID is close to the optimal[4] joint decoding.

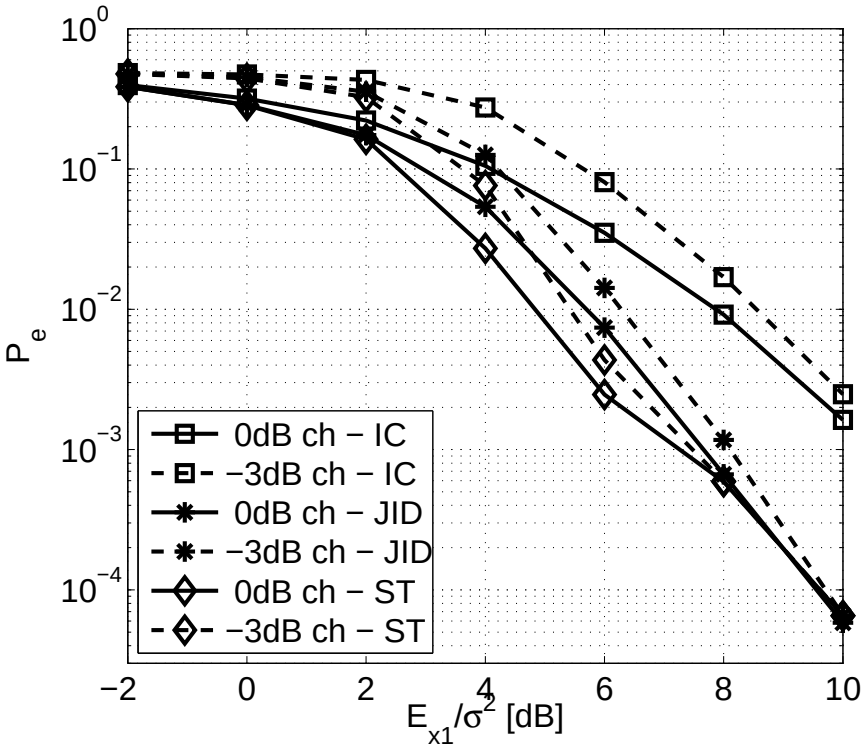

Figure 8.5: ST vs JID vs SIC in terms of bit error rate.

For the comparison in Fig. 8.5 the same codebook $\mathcal{C}$ is used for the two channels where the latter one is 3 dB weaker than the former. We use $q = 2$, $n = 2$, $m = 1$, and 998 bits are encoded into 1000 complex–valued QPSK symbols by using the CC (o2, o3) $\Gamma = 2$ and rate=1/2. The channel is assumed to be static, i.e., $\mathbf{H}(t) = \mathbf{H}$, and known at the receiver. The codebook we use is the same for every transmitter, i.e., *no* interleaver. The

[4]Block and symbol optimal performance curves are observed to be extremely close such cases.

number of iterations necessary for the convergence of JID is 2, which may be expected because of the very small loops.

8.3.7 Detailed Comparison: JID vs SIC Performance

We further study the case with two interferers, $n = 2$, and one or two receiving antennas, $m = 1, 2$. The CC is the same for both transmitters: it has two branches with memory $\Gamma = 6$ defined by the polynomials (o163, o135) [100]. The transmitted codeword is obtained by coding 72 source bits in 156 coded bit and contains 64 QPSK coded symbols, $q = 2$, after the mapping and uniform symbol puncturing. The comparison between JID and IC is shown in Fig.s 8.6, 8.7, 8.8, and 8.9 for both 1 and 2 receiver antennas case and for both *block error rate* target 10% and 1%.

We have

$$E_i = \mathsf{E}\left[\|\mathbf{h}_i x_i(t)\|^2\right]. \tag{8.15}$$

The pictures has to be read in this way: x–axis denotes the ratio between the energy of the strongest transmitter and the energy of the weakest E_1/E_2; y–axis is the scale of SNR for the strongest signal E_1/N_0. White area is the region where no signal is decoded; light gray means that the strongest signal alone is decode; then dark gray area means that both signals are decoded.

Fig.s 8.6–8.9 indicate the regime where JID outperforms IC: the white region is significantly smaller, in particular when E_2 is closer to E_1 the *similar energy* region. To understand this behavior, recall that IC decodes the signal considering the rest as noise,

$$\mathbf{y}(t) = \mathbf{h}_1 x_1(t) + \underbrace{\mathbf{h}_2 x_2(t) + \boldsymbol{\nu}(t)}_{\boldsymbol{\nu}'(t)}. \tag{8.16}$$

In short, if E_2 becomes significantly large, then the equivalent noise $\boldsymbol{\nu}'(t)$ is above the decoding ability of the receiver. When we consider two receiver antennas case, the *similar energy* region is significantly larger than for the one antenna case thanks to the additional diversity provided by the multiple antennas. Furthermore, in the complementary region we observe a classical gain of 3 dB since the receiver with two antennas catches double energy.

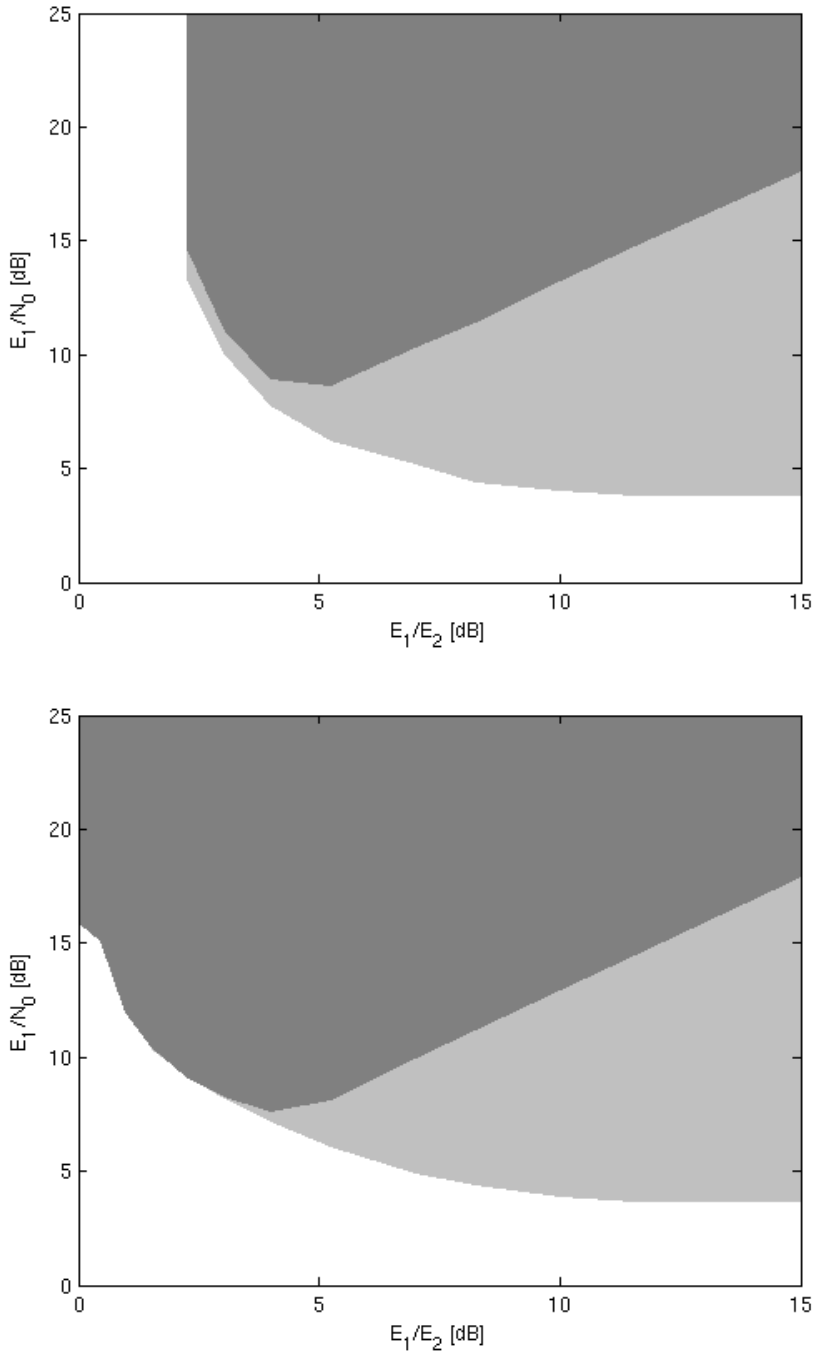

Figure 8.6: Comparison between IC (upper pictures) vs JID (lower pictures) for 2 transmitters: 1 receiver antenna, BER target 10%.

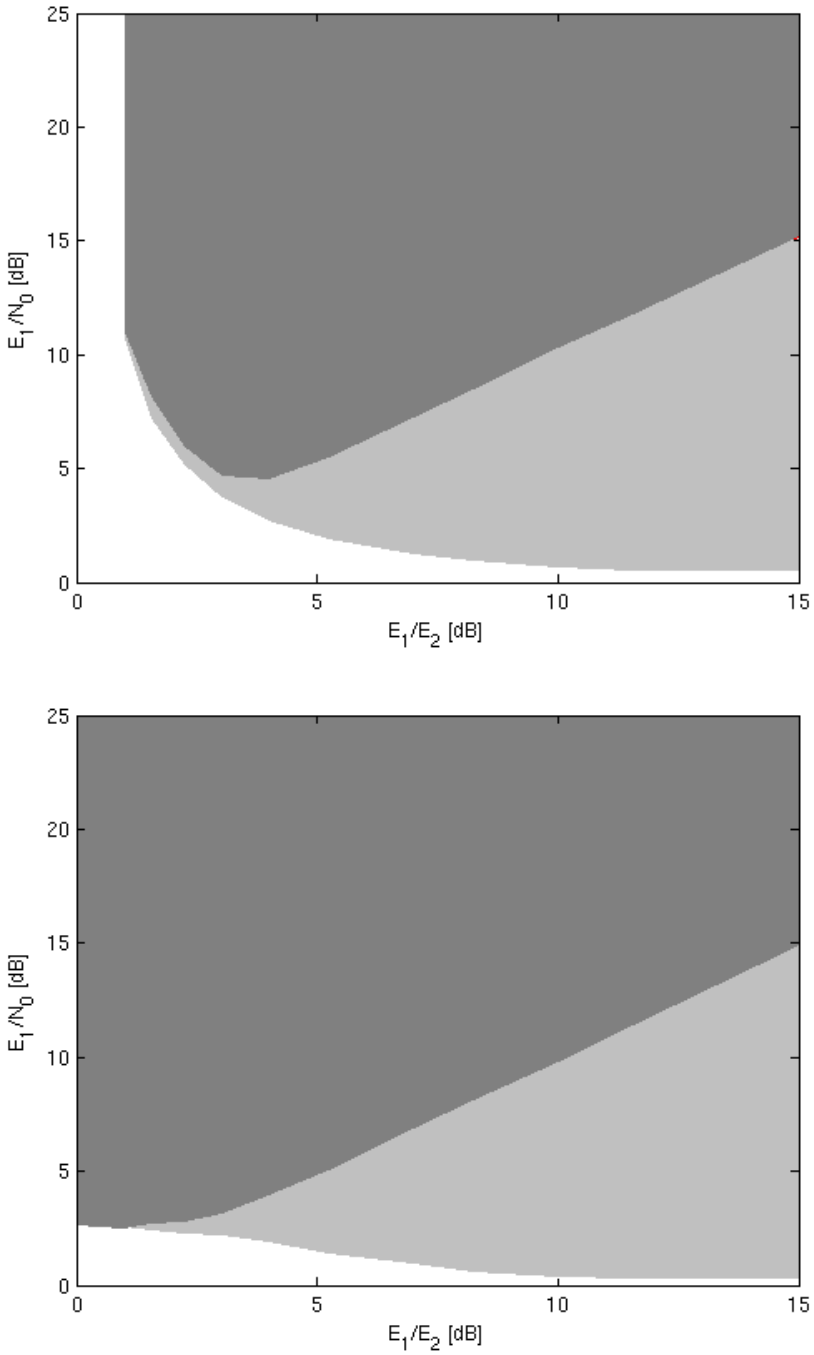

Figure 8.7: Comparison between IC (upper pictures) vs JID (lower pictures) for 2 transmitters: 2 receiver antenna, BER target 10%.

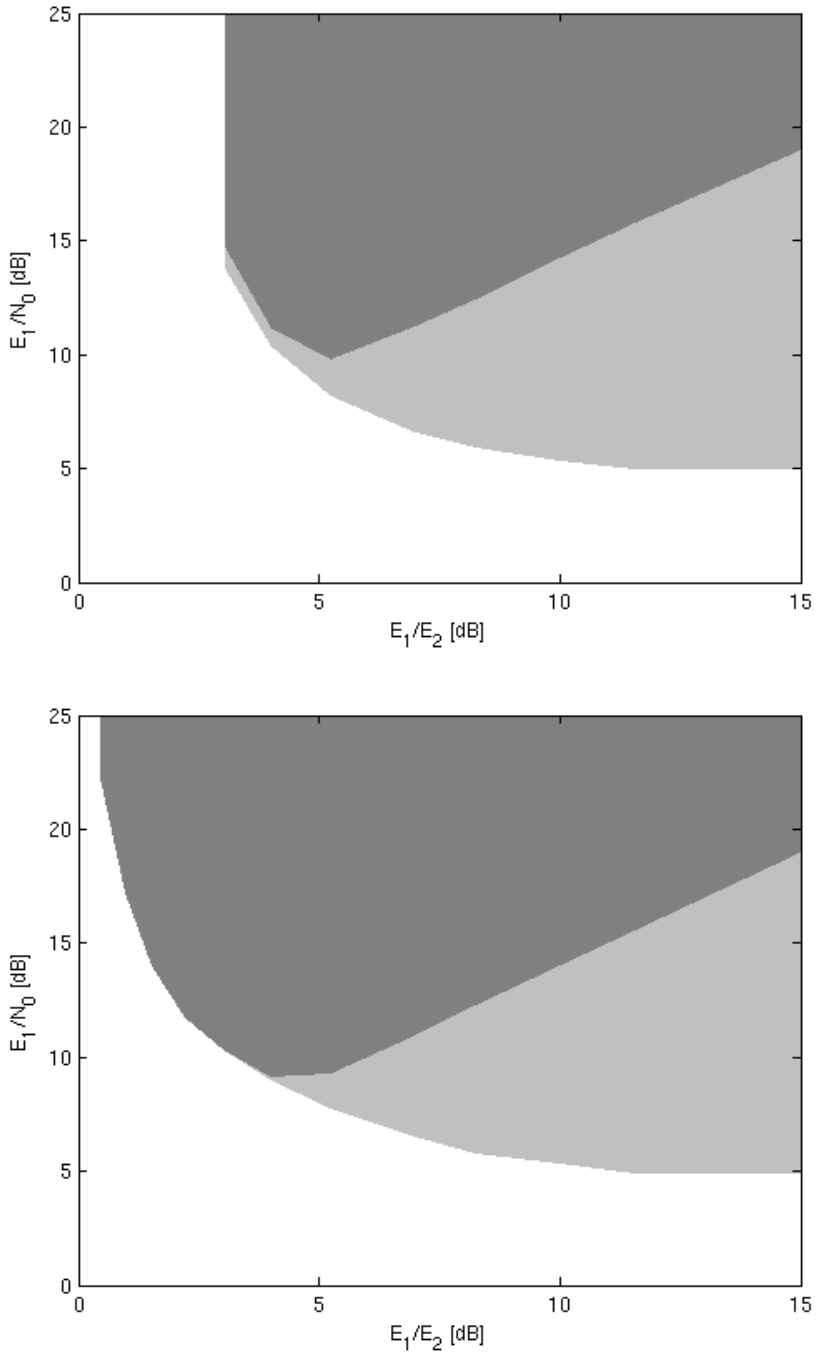

Figure 8.8: Comparison between IC (upper pictures) vs JID (lower pictures) for 2 transmitters: 1 receiver antenna, BER target 1%.

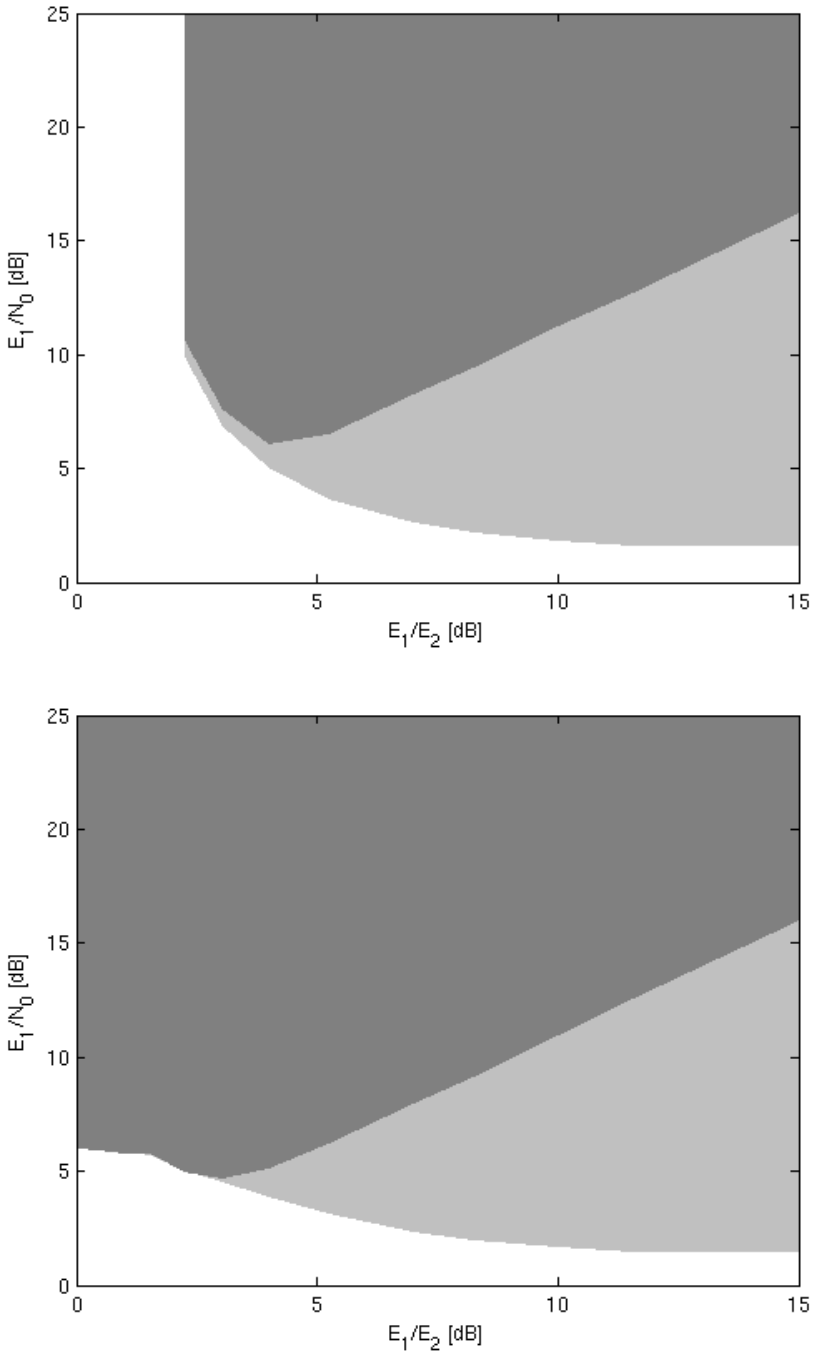

Figure 8.9: Comparison between IC (upper pictures) vs JID (lower pictures) for 2 transmitters: 2 receiver antenna, BER target 1%.

8.4 Dense Scenario: Coherent Case

Let us now simulate a typical peer–to–peer network. Consider a planar region as a square surface of 500 m × 500 m without obstacles over which devices are dropped uniformly at random. Assume that we are given $K = 50$ physical resources and that the number of devices N is either 5 or 10 times the number of physical resources. This forms a *high density* scenario. Fig. 8.10 shows an example where the number of devices is 5 times the number of resources.

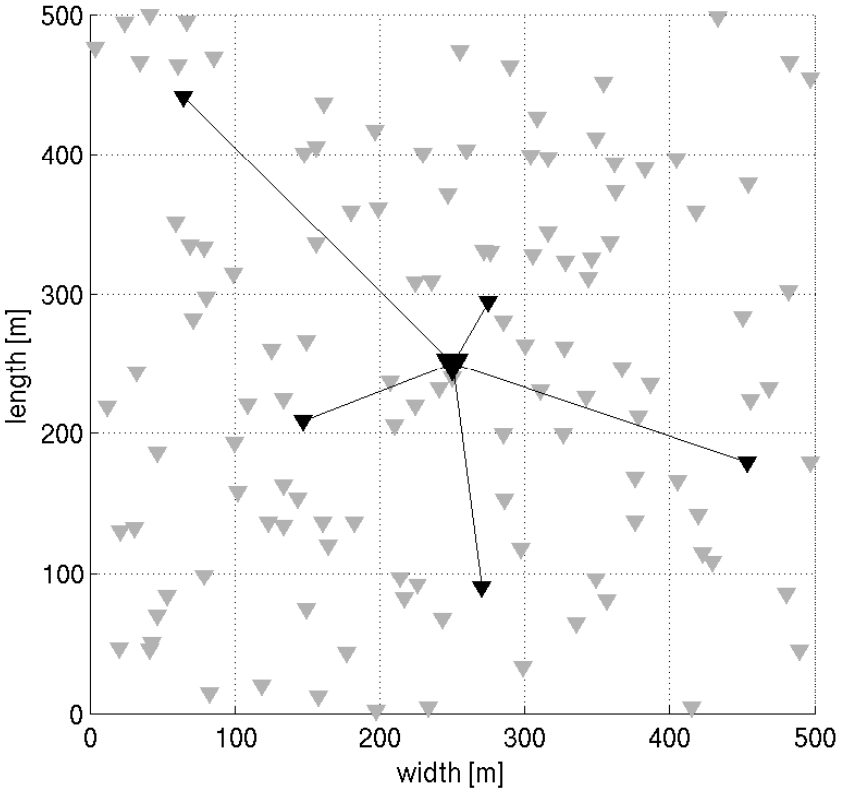

Figure 8.10: Dense scenario simulation with density degree $N/K = 5$.

In this picture, the big black device in the middle of the region represents the receiver node, the small black nodes are the transmitter devices that are sharing the same physical resource, finally all the reminder gray nodes are non–interfering nodes using different resources. In our simulations the channel gain is computed according to the relationship

$$E_i \propto d_i^{-\alpha} \tag{8.17}$$

where d_i is the distance between the i–th transmitter node and the receiver and α is the wave propagation attenuation. For most of the presented experiments we consider $\alpha = 3.5$.

The model is

$$\begin{aligned} \mathbf{y}(t) &= \sum_{i=1}^{n} \mathbf{h}_i x_i(t) + \boldsymbol{\nu}(t) \\ &\simeq \mathbf{h}_1 x_1(t) + \mathbf{h}_2 x_2(t) + \boldsymbol{\nu}'(t). \end{aligned} \tag{8.18}$$

Notice that the approximation in Eq. (8.18) is generally pessimistic because the bounded remaining transmitters as here considered as unbounded Gaussian noise

$$\nu'_k(t) \sim \mathcal{CN}(0, N'_0) \text{ with } N'_0 = N_0 + \sum_{i=3}^{n} E_i, \ \forall k. \tag{8.19}$$

8.4.1 General Performance of JID

Simulation results are summarized in Table 8.1. In a 5× scenario, observe that a receiver that performs joint decoding in association with two receiving antennas performs almost twice better than a conventional (std) receiver with one antenna and considering the interference as noise.

	1×		2×		5×		10×	
	avg dev	%	avg dev	%	avg dev	%	avg dev	%
1A std	1.0000	1.00	0.7717	1.00	0.5915	1.00	0.5427	1.00
1A IC	1.0000	1.00	1.5095	1.96	0.6620	1.12	0.5640	1.04
1A JID	1.0000	1.00	1.8770	2.43	0.6904	1.17	0.5810	1.07
2A std	1.0000	1.00	0.8856	1.15	0.7195	1.22	0.6805	1.25
2A IC	1.0000	1.00	1.6215	2.10	0.9864	1.67	0.8129	1.50
2A JID	1.0000	1.00	1.9647	2.55	1.1542	1.95	0.9323	1.72

Table 8.1: JID performance in dense scenarios.

In Table 8.1 '1A' means a single antenna receiver and '2A' means a receiver with 2 antennas. We denote by 'std' a standard or conventional decoding that detects a codeword from the strongest transmitted signal. The first column shows the average number of discovered nodes per receiver node and per physical resource. The second represents the gain in percentage when compared to the basic conventional solution using a single receiver antenna. Decoding target is the *block error rate* $= 10^{-2}$.

Notice that that increasing the density causes a natural degradation of the performance since $n = 2$ does not longer model the $n \gg 2$ interference channel well enough.

8.4.2 ST vs Conventional Single User Detection at Comparable Trellis Complexity

While dealing with the structure of the noise and its potential closeness with an AWGN, it is of interest to experimentally compare the following optimal receivers of comparable complexity: (i) The receiver decodes a single user (all peers use a strong CC) assuming the rest as noise (ii) The receiver decodes jointly two users (each peer use a weaker CC while the overall decoding complexity is similar as in (i))

For example, we compare the performance of a ST–based receiver with $n = 2$ when peers are using a CC with memory $\Gamma_{ST} = 4$ and a standard trellis–based decoder with $n = 1$ when peers are using a CC with memory $\Gamma = 8$. The two trellises have the same number of states (hence a similar[5] decoding complexity) since $2^{n\Gamma_{ST}} = 2^{\Gamma}$. The polynomial used are (o23, o35) for the CC with memory 4 and (o457, o755) for the CC with memory 8. Table 8.2 reports the performance for the single receiver antenna and four density scenarios in terms of number of devices in the square: 1, 2, 5, and 10 times the number of channels.

	std $\Gamma = 8$	ST $\Gamma_{ST} = 4$	% improve
1×	0.9286	0.8349	0.90
2×	0.7533	1.3419	1.78
5×	0.6159	0.6269	1.02
10×	0.5667	0.5286	0.93

Table 8.2: ST performance. The last column shows the improvement ratio.

These simulations are done with $\alpha = 5$ to stress better the behavior in different density. In the 1× case, the link channel is typically the standard AWGN channel with QPSK signaling and specific standard point–to–point coding performs obviously better. In such case there are no channel sharing then the performance of ST is comparable to a standard decoder for a CC with memory 4 that is weaker than memory 8. When the density becomes high, the interfering signals tends to jointly form an AWGN and the trade off with a better point–to–point coding scheme appears again. Unsurprisingly, in the 2× density case, the ST receiver is typically tailored for the link model: the gain of using it is about 78%. This suggests that a proper congestion control at the network level combined with the ability of joint decoding is extremely valuable and should translate into impressive performance gains. Hence ST outperforms a same complexity standard decoding just for small

[5]Notice that the ST may appear slightly more complex due to the fact that it has twice more edges.

density of devices between 2 and 5 times the number of channels. A final consideration is necessary for 10× case. Because of the high density, we have a very noisy scenario then the probability to have $E_2 \ll N_0'$ is non–negligible. Under this hypothesis we can assume the system close to a single user case

$$E_2 \ll N_0' \quad \Rightarrow \quad \mathbf{y}(t) \simeq \mathbf{h}_1 x_1(t) + \boldsymbol{\nu}'(t). \tag{8.20}$$

and the model can be approximated as a noisy 1×. That is the reason because ST fails in very high density scenarios.

8.5 Decoding Algorithms: Non–Coherent Case

In practice the wireless channel is unknown at the receiver. It may also be time–varying depending on physical variations or technical imperfections. We deal with non–coherent decoding for the multiple–access fading channel. From an engineering viewpoint this implies to somehow *estimate* the channels between the n transmitters and the receiver. Without loss of generality we still assume $n = 2$. Notice here that non–coherent MAC estimation is less trivial than standard point–to–point channel estimation. Fortunately, it is possible to model the channel time–variations to the first order and devise corresponding elementary estimation strategies.

8.5.1 Optimal Decoding Rule

The optimal decision rules reads

$$\begin{aligned} \hat{x}_i(t) &= \arg\max_{x_i(t)\in\mathcal{A}} p(x_i(t)|\overline{\mathbf{y}}) \\ &= \arg\max_{x_i(t)\in\mathcal{A}} \sum_{\overline{\mathbf{x}}\sim x_i(t)} \sum_{\mathbf{H}} p(\overline{\mathbf{x}}, \overline{\mathbf{y}}, \mathbf{H}). \end{aligned} \tag{8.21}$$

It is straightforward to see that the overall complexity can be extremely high depending on the possible channel quantization: the multi–user decoding problem with unknown channel is challenging to implement at reasonable computational and memory cost.

8.5.2 Hierarchical Coding

Our solution is to break the problem into two parts, in the same way we usually do for single user detection/decoding. We shall see that this suboptimal solution performs close to the optimal rule, at least in our framework. More

exactly, we will achieve performance in the non–coherent case that is close to the ideal coherent case.

We use

(*i*) a non–coherent channel code that is located on a small subset of the symbols pilots;

(*ii*) a coherent, typically larger, channel code that is located on the remaining symbols.

The first is also called pilot code while the second is referred to as data code.

From an operational viewpoint, the pilot code is used for channel estimation and sequence detection, and the data code is used for JID. The information encoded by the pilot code characterizes the type of data code we are using. Decoding is therefore *hierarchical*:[6]

(*i*) first, non–coherent decoding is performed for the pilot positions, which we also use for channel estimation;

(*ii*) second, coherent JID is performed for the remaining positions since the channel is known.

In other words, we simply use the data code developed for coherent JID in the previous sections, i.e. a CC followed by a random interleaver. It remains now to find a satisfying non–coherent pilot code. In a first approach, we may avoid using complex algebraic constructions. We aim at demonstrating the validity of our hierarchical method. Hence we use elementary codes that *orthogonalize* the information carried by the pilots coming from different users. We shall use simple Walsh or Fourier–like sequences depending on the channel model: this design choice is motivated by the existence of a practical and efficient decoder.

8.5.3 Flat Fading Channel Model

Consider first a time–invariant flat fading channel

$$h_{\ell i}(t) = h_{\ell i} = \sqrt{E_{\ell i}} e^{j\theta_{\ell i}} \tag{8.22}$$

where $\sum_\ell E_{\ell i} = mE_i$ and $\theta_{\ell i}$ is generated uniformly in $(0, 2\pi)$ for each word. Assume that each peer transmits an identifier encoded in a sequence of 72 QPSK symbols: 64 positions are reserved for the data code and 8 positions

[6] Generalizations of this technique include additional code layers, well–designed pilot codes, or refined information carried on the pilot codes.

for the pilots that are equally spaced (with period $T_{\mathrm{P}} = 9T$ while the symbol duration is assumed to be $T \simeq 20$ μs). The pilot codeword is chosen by taking uniformly at random a Walsh sequence of length 8, that is a row $\mathbf{w}_i$ of the 8×8 Hadamard matrix

$$\boldsymbol{\mathcal{H}} = \sqrt{E_q} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{w}_3 \\ \mathbf{w}_4 \\ \mathbf{w}_5 \\ \mathbf{w}_6 \\ \mathbf{w}_7 \\ \mathbf{w}_8 \end{bmatrix} \tag{8.23}$$

where E_q is the energy of QPSK symbols. The matrix $\boldsymbol{\mathcal{H}}$ is symmetric and orthogonal such that $\boldsymbol{\mathcal{H}}\boldsymbol{\mathcal{H}}^{\mathrm{T}} = 8E_q\mathbf{I}$. The choice of the sequence is arbitrary. Here we chose it according this rule: we take three bits of the transmitted source sequence, which form a simple non–coherent code. We map the value of these three bit on one row of $\boldsymbol{\mathcal{H}}$. This policy helps the decoding because it provides some information on the transmitted bits from the sequence detection. Let $\overline{\boldsymbol{\nu}}_{\mathrm{P}}$ the sequence of the equivalent noise vectors in the pilot positions $\{\boldsymbol{\nu}'(kT_{\mathrm{P}})\}_{k=1,\dots,8}$, the received vector on the pilot positions is the $m \times 8$ matrix

$$\overline{\mathbf{y}}_{\mathrm{P}} = \mathbf{h}_1\mathbf{w}_i + \mathbf{h}_2\mathbf{w}_j + \overline{\boldsymbol{\nu}}_{\mathrm{P}} \tag{8.24}$$

where $i, j \in \{1, \dots, 8\}$ label the two Walsh sequences. The receiver simply performs the Hadamard transform of $\overline{\mathbf{y}}_{\mathrm{P}}$ in order to estimate the sent sequences:

(i) with probability 7/8 we have $i \neq j$, i.e., no collision, and we get

$$\frac{1}{8E_q}\overline{\mathbf{y}}_{\mathrm{P}}\boldsymbol{\mathcal{H}} = [\boldsymbol{\xi}_1 \dots (\mathbf{h}_1 + \boldsymbol{\xi}_i) \dots (\mathbf{h}_2 + \boldsymbol{\xi}_j) \dots \boldsymbol{\xi}_8], \tag{8.25}$$

where

$$\boldsymbol{\xi}_k = \frac{1}{8E_q}\overline{\boldsymbol{\nu}}_{\mathrm{P}}\mathbf{w}_k^{\mathrm{T}} \sim \mathcal{CN}\left(\mathbf{0}, \frac{N_0'}{8E_q}\mathbf{I}_m\right), \quad k = 1, \dots, 8 \tag{8.26}$$

represents the equivalent noise. The sequence detection is then performed by considering the position of the two rows of the Hadamard transform with largest Euclidean norm. If

$$\|\mathbf{h}_{1,2} + \boldsymbol{\xi}_{i,j}\| > \|\boldsymbol{\xi}_k\|, \quad \forall k \neq i, j, \tag{8.27}$$

then the sequence detection is correct for both transmissions. Hence, the channel estimation is provided by taking the values of these rows and the estimation error is defined by $\boldsymbol{\xi}_{i,j}$.

(*ii*) With probability 1/8, collision occurs, i.e. $i = j$, we get

$$\frac{1}{8E_q}\overline{\mathbf{y}}_{\mathrm{P}}\boldsymbol{\mathcal{H}} = [\boldsymbol{\xi}_1 \dots (\mathbf{h}_1 + \mathbf{h}_2 + \boldsymbol{\xi}_i) \dots \boldsymbol{\xi}_8]. \tag{8.28}$$

In this case we can still detect the sent sequence but the channel estimation can only be accomplished for the strongest signal if

$$\|\mathbf{h}_1\| \gg \|\mathbf{h}_2\| \tag{8.29}$$

such as the weakest signal is contained in the noise.

Table 8.3 summarizes the high density performance considering that collision happens with probability 1/8.

	1×			2×		
	Avg	%	loss	Avg	%	loss
1A std	1.0000	1.00	1.00	0.7437	1.00	0.96
1A IC	1.0000	1.00	1.00	1.3805	1.86	0.91
1A JID	1.0000	1.00	1.00	1.6981	2.28	0.90
2A std	1.0000	1.00	1.00	0.7921	1.07	0.89
2A IC	1.0000	1.00	1.00	1.4832	1.99	0.91
2A JID	1.0000	1.00	1.00	1.7780	2.39	0.91
	5×			10×		
	Avg	%	loss	Avg	%	loss
1A std	0.5617	1.00	0.95	0.5129	1.00	0.95
1A IC	0.5972	1.06	0.90	0.5218	1.02	0.93
1A JID	0.6056	1.08	0.88	0.5249	1.02	0.90
2A std	0.6814	1.21	0.95	0.6420	1.25	0.94
2A IC	0.8332	1.48	0.85	0.7087	1.38	0.87
2A JID	0.9026	1.61	0.78	0.7455	1.45	0.80

Table 8.3: JID performance for unknown channel.

The 4–th and 7–th columns in this table represent the performance degradation caused by the non–coherent channel versus the coherent case of previous section. The worst case shows a 22% loss, which is quite low and very reasonable for practical purpose.

Fig.s 8.11 and 8.12 shows the JID performance in the case of one and two receiving antennas with Walsh sequences on the pilots. Decoding target is *block error* rate $= 10^{-2}$.

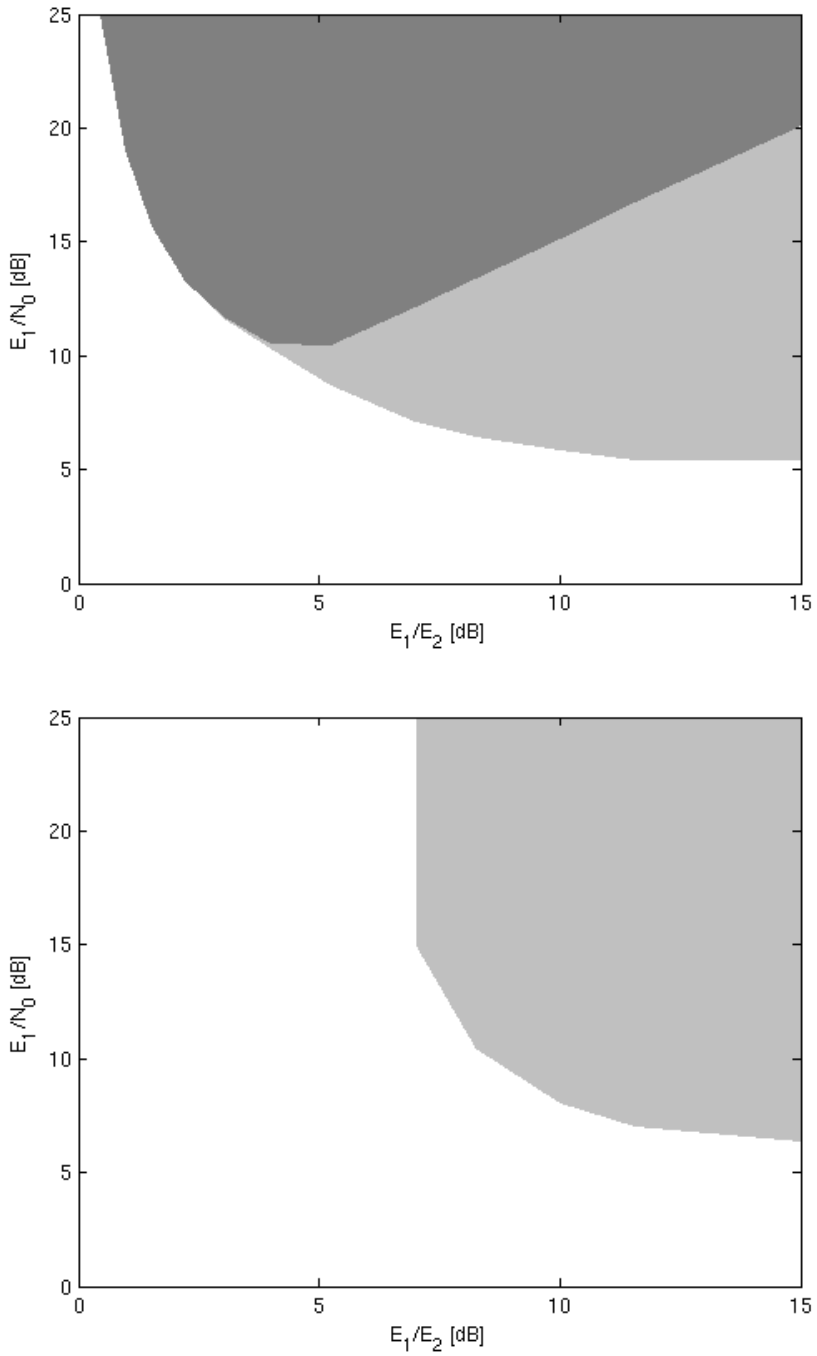

Figure 8.11: JID performance with channel estimation using Walsh sequences for 2 transmitters: 1 receiver antenna, BER target 1%. No–collision (upper pictures) vs collision (lower pictures).

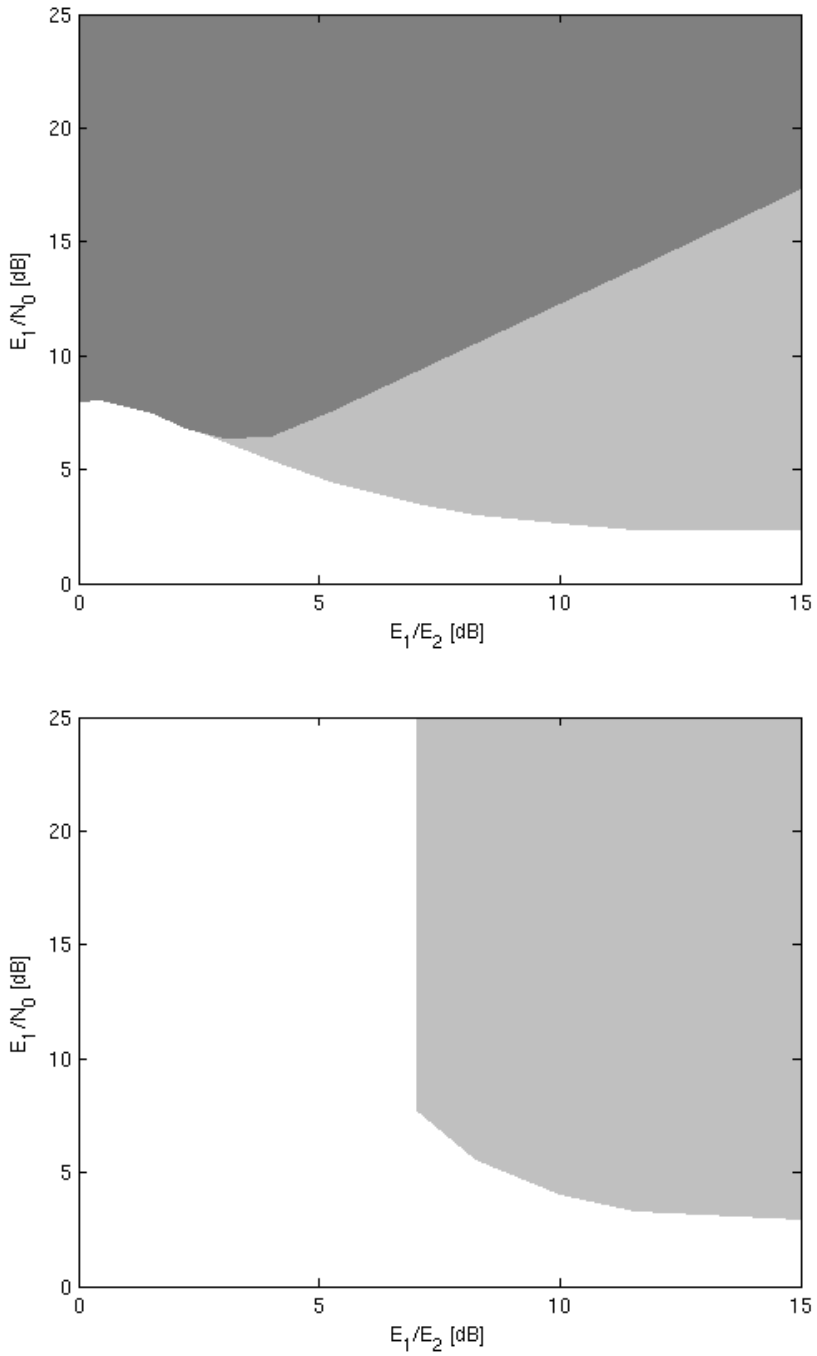

Figure 8.12: JID performance with channel estimation using Walsh sequences for 2 transmitters: 2 receiver antenna, BER target 1%. No–collision (upper pictures) vs collision (lower pictures).

The upper figures represent the non–collision case: comparing with the channel known case the degradation is about 1 dB. On the bottom we have the collision case where we can distinguish a region where the condition in Eq. (8.29) is satisfied and the algorithm decodes the strongest signal. The color legend is the same we used above.

8.5.4 Improved Implementation: Different Interleavers

The pilot code may further be used to provide information on the type of data code that is used. This turns out to be extremely efficient in conjunction with JID. As a simple example we shall use 3 information bits carried by the pilot code to characterize one among the 8 possible and different interleaver patterns $\{\boldsymbol{\pi}_i\}$ that each device picks at random. Assuming that the sequence detection is correct, the receiver uses the associated interleaving pattern to build the graphical model used for JID. The experimental benefits offered by our general approach as well as those of using different codebooks $\mathcal{C}_i \neq \mathcal{C}_j$ is shown in Fig. 8.13 where we compare the performance of JID and ST.

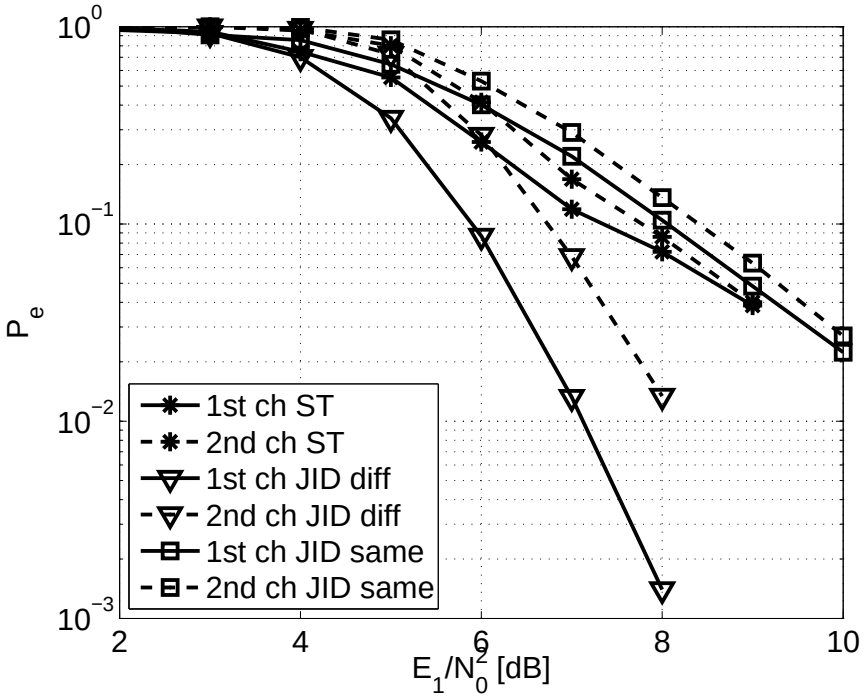

Figure 8.13: Performance comparison between JID and ST using same interleaver and different interleavers.

This picture shows the case $E_1/E_2 = 3$ dB. Since we are using different

codebooks, the number of iterations necessary to converge is 5. The motivation for using a CC followed by symbol interleaving at each peer should now be obvious. First, when the peers use different interleavers (no–collision case), the girth of the graphical model implemented at the receiver is typically larger than it would be without interleaving: this improves the message–passing performance). Second, and very important, the joint code formed by the graphical model is much stronger, mimicking the Turbo–principle, when, in short, information at a node arrive from nodes that are randomly far apart on the new graphical model.

In the dense scenario simulation performance, for the single receiver antenna case, it guarantees an improvement of 6% for the 5× dense scenario and 2% for the 10×.

8.5.5 Frequency Offset Channel Model

The previous static channel is unfortunately an optimistic model. Experimental measures show that some time variation has to be further incorporated in the model: it is mainly caused by

(*i*) the device mobility (Doppler);

(*ii*) the imperfect alignment between peer carrier frequencies.

The classical time–variant channel model has Doppler spectrum described by the Jakes model

$$\mathcal{D}(f) \propto \frac{1}{\sqrt{f_i^2 - f^2}}. \tag{8.30}$$

We use a first order approximation with the *Frequency Offset* channel model

$$h_{\ell i}(t) = h_{\ell i}(kT) = \sqrt{E_{\ell i}} e^{j\theta_{\ell i}} e^{j2\pi f_i kT} \tag{8.31}$$

where f_i is the new parameter that we have to estimate for the i–th transmitter. We assume f_i to be a uniform random variable in the range between -200 Hz and 200 Hz.

Because of the frequency offset model, it is more suitable to consider pilot orthogonality in the frequency domain: the new pilot code we use is formed by Fourier–like sequences that add an artificial frequency offset to the current frequency offset induced by the channel. Pilot sequence detection and channel estimation are jointly performed through parametric estimation using Prony's method. This method has been recently and extensively used for sparse sampling at the finite rate of innovation, see [101] and related works. These works motivate our pilot code design, a choice that is also reinforced

by the existence of low complex and very accurate implementations. In general, alternative parametric estimation methods can be implemented such as FFT–based methods using large FFT sizes. For now, our aim is mainly and purely to demonstrate the validity of JID for non–coherent peer discovery. Our non–coherent pilot decoder is basically described in [101] for the spike detection in the time domain. We use the shortcut AF+CD to name it because it is composed of two steps:

(*i*) *Cadzow iterative denoising* (CD);

(*ii*) the total least square method to solve *annihilating filter* (AF) equation.

The flow chart algorithm is shown in Fig. 8.14.

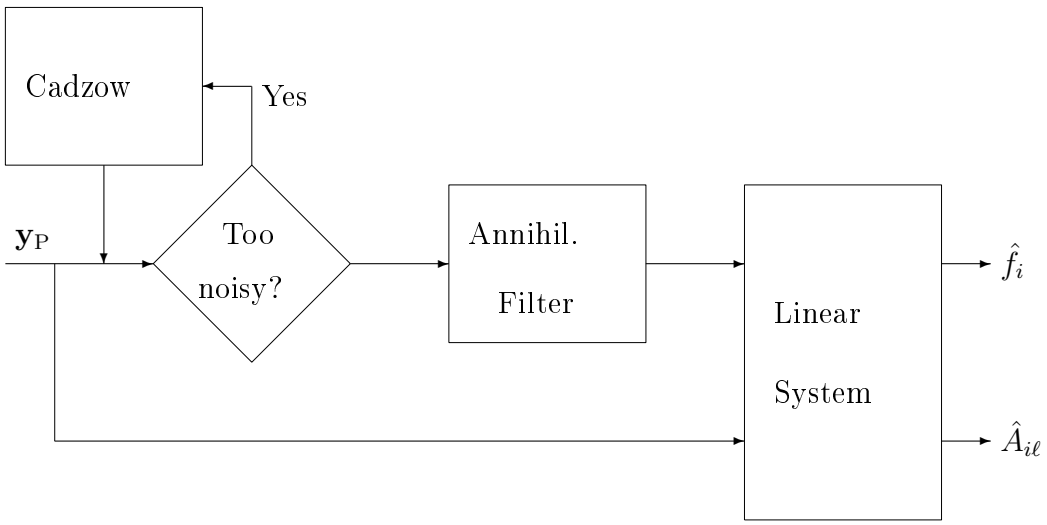

Figure 8.14: Schematic chart of the parametric channel estimation algorithm AF+CD.

The symbol $\hat{f}_i$ represents the estimation of the frequency offset f_i then $\hat{A}_{i\ell}$ is the estimation of the complex value $\sqrt{E_{\ell i}}e^{j\theta_{\ell i}}$.

The AF+CD parametric estimation for the position of spikes in the frequency domain, in the presence of noise, finds the best least square fit. Because AF+CD does not perform well in the noisy case when the spikes are close, we use 8 different pilot sequences supported by more than 8 pilots. In our case a good choice is to assign 12 pilots to the non–coherent code. This implies that $T_{\mathrm{P}} = 6T$ and the sequences are chosen from the rows of the

matrix

$$\boldsymbol{\mathcal{F}} = \sqrt{E_q} \begin{bmatrix} 1 & e^{j2\pi\Delta f_1 T_{\mathrm{P}}} & \dots & e^{j2\pi\Delta f_1 11 T_{\mathrm{P}}} \\ 1 & e^{j2\pi\Delta f_2 T_{\mathrm{P}}} & \dots & e^{j2\pi\Delta f_2 11 T_{\mathrm{P}}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{j2\pi\Delta f_8 T_{\mathrm{P}}} & \dots & e^{j2\pi\Delta f_8 11 T_{\mathrm{P}}} \end{bmatrix}, \tag{8.32}$$

where $\Delta f_k = (2k-1)/16T_{\mathrm{P}}$. To preserve the same number of transmitted symbols and source bits per codeword, we increase slightly the number of punctured symbols and use a (120, 72) CC.

At the receiver AF+CD estimates $\hat{f}_i \simeq \Delta f_k + f_i$, which is the sum of the channel frequency offset and the artificial frequency offset characterizing the pilot sequence. If $|f_i| \ll \Delta f_{k+1} - \Delta f_k$, the pilot sequence is detected by taking the artificial frequency offset that is the closest to the estimated frequency

$$\arg\min_{\Delta f_k} |\hat{f}_i - \Delta f_k|. \tag{8.33}$$

To improve the performance, the information from the pilot sequence detection can be used for labeling three source bits and eight different interleaver patterns $\boldsymbol{\pi}_i$ in the way we presented for the time–invariant channel model using Walsh sequences. Regarding computational complexity, CD computes one singular value decomposition (SVD) of a 7×6 matrix per iteration ($\sim$10 iterations); AF computes one SVD of a 10×3 matrix and the roots of a polynomial of degree equal to n. The AF+CD complexity turns out to be negligible compared with the JID, which justifies of our approach. Table 8.4 shows a dense scenario performance using $n = 2$ transmitters and a single antenna receiver.

Density	Hadamard	AF+CD	%
1×	1.0000	1.0000	1.00
2×	1.7633	1.7597	1.00
5×	0.6416	0.5533	0.86
10×	0.5355	0.4549	0.85

Table 8.4: Channel estimation and JID using different interleavers.

Last column of Table 8.4 shows the degradation of the performance in a time–varying channel compared to the static channel. AF+CD method ensures this degradation to be $\sim 15\%$.

8.6 Final Considerations

This chapter deals with the technical challenges coming from peer discovery in wireless networks. Distributed network strategies can be complementary to the ability for joint detection at each node. With this aim standard receipts from graphical models and message–passing algorithms (i.e., JID) turn out to be extremely efficient for practical purpose.

We first presented the optimal receiver based on the trellis product. The ST receiver complexity grows exponentially with the number of devices to be jointly decoded. Its practical advantage may therefore be restricted to some specific and mid–density scenarios, where n is small. It is therefore mainly a benchmark used to evaluate JID.We further presented JID, which can be easily implemented for both single and multiple receiving antennas. In terms of computational complexity, JID is comparable with SIC while JID outperforms SIC. In the non–coherent case we eventually used a frequency offset channel model to mimic the real time–variations of the channel up to the first order. Fourier–like sequences are example of a pilot code that can be easily decoded via practical algorithms such as the Cadzow denoising and annihilating filter method or more standard methods based on FFT. We experimentally validated the JID approach even in this non–coherent MAC case by using hierarchical decoding (first non–coherent pilot code, second coherent data code).

From a coding viewpoint, it is remarkable to observe that:

(i) interfering nodes that use different but very simple codebooks may jointly form a very efficient code, e.g. when different interleavers are used with CCs;

(ii) that multiple receiving antennas may bring significant gains.

Of course, more refined decoding algorithms could bring even better gain, in particular if some optimization is performed, and/or if channel estimation is jointly performed.

From a network viewpoint, one important observation we have here is that the system performance improvement with two–user JID is quite limited in the dense scenario. This is simply because that in a 2–dimension network, a receiver typically sees more than two strong codewords in the same resource and the capability of being able to decode two codewords can not help much in this case. A further careful analysis can show that a receiver typical sees *three* strong codewords rather than *two*. Apparently, to improve the performance in this deployment scenario, we can either consider a three–user JID

receiver or introduce system level algorithms to reduce the level of the congestion in the network, including distributed algorithms for avoiding collisions, controlling the allocation or orthogonalization of the physical resources and controlling the transmit power of each node to create a two–codewords interference scenario. Such system strategies are in general complementary to the coding used at each peer. We have seen that lot is gained if the code design is well adjusted to its local (i.e., controlled) environment so that adaptive strategies could be further imagined.

Chapter 9

Conclusions

This thesis gathers a study of analyzed eligible solutions for the second generation of European digital terrestrial broadcasting system DVB–T2. Clearly not all of them are included in the final standard definition but they also suggests some important and interesting results about this specific scenario. The key technologies included within the DVB–T2 standard has been presented with some first performance results. It is impressive to observe that commercial requirements have been met and in some cases substantially exceeded. We recall the most important fact who moved the creation of this new standard is to provide significantly improved performance for HDTV transmissions more and more demanded by clients and customers, but this is not the only important new feature that distinguishes the new standard. The first terrestrial HDTV service using DVB–T2 is expected for the end of 2009 in UK.

In our research on performance improvement for the standard we have firstly investigate a multiple antennas systems and the related space–time block codes. After a first performance evaluation of these methods, we have shown how deep is the impact of an LDPC code combined with the STBC. A significant problem related to this very complex system is the computation of soft information required by the LDPC decoder: it is the bottleneck of the system. To this aim we have developed a new soft detector for STBCs based on an efficient implementation on the complex sphere decoder algorithm. Compared to the canonical method, we have seen up to 50% of the time requirement reduction by the new algorithm on the interesting situations.

After the detection topic, we have moved our attention to modulation and mapping strategies. We have presented a performance comparison among three different modulation diversity techniques applied to OFDM systems in the presence of frequency dispersive channels in order to analyze their behavior in a coded transmission scenario. These are remapped–repetition

modulation, rotational multi–carrier modulation, and multidimensional rotated QAM. OFDM systems translate frequency dispersive channel into flat channels having different fading, thus the presented modulation techniques avail this diversity to improve the performance. One more time, we have involved LDPC codes at both high and low–rate so the comparison is performed both in an uncoded and LDPC coded scenario. Despite modulation diversity methods have the potential to improve the detection of data, FEC codes are able to correct strongly corrupted data due to deep fade channels, i.e. low–rate LDPC in a way that their effect dominates the diversity gain provided by the presented modulation schemes. Anyway they are sometimes able to give a very little improvement when are combined with LDPC codes.

Further methods have been considered. We have studied the joint performance of DVB–S2 interleavers and DVB–T demux, combined with LDPC codes, in order to find a good configuration eligible for the new standard DVB–T2. Other techniques, such as iterative demapping and re–mapped repetition, have been introduced, in particular thanks to their diversity which provides a performance gain in portable channels. Although this research has pointed out several limitations in any new scenarios, it has suggested very few good ideas applicable to a new broadcast scenario, since, most of the times, the small system performance gain does not justify the cost in terms of decoding complexity dues to the iterations and the mapping latency introduced by the symbol interleaver.

The final topic we have presented in this work is the proposition of coded decision directed demodulation (CD3) scheme to DVB–T2. This is actual part of the final DVB–T2 standard. Simulation results for the proposed technique have been seen even in multi–antennas scenario. We have show that CD3, thanks to the capability to track channel variations with a reduced overhead of pilot symbols with respect to conventional channel estimation techniques for OFDM, allows to achieve high data rate transmission even for channel with fast time variations and severe frequency selective fading. This method yields a significant achievable throughput improvement compared to DVB–T then it allows DVB–T2 to provide HDTV.

Parts of this research are published in [102–104].

The second part of this thesis reports a work of peer discovery in wireless cellular networks. The ability to get joint detection at each node in very high density scenario is investigated. The algorithms are derived by the fundamentals of graphical models and message–passing algorithms. We first presented the optimal receiver based on the trellis product but its complexity grows exponentially with the number of devices to be jointly decoded. We further presented a suboptimal joint iterative decoder whose computational complexity is comparable with the canonical successive interference cancel-

lation with better performance. Channel estimation solutions are proposed: they require Fourier–like sequences as a pilot code that fit for decoding algorithms such as the Cadzow denoising and annihilating filter method or the more standard methods based on FFT. From these strategies we have also recognized that further significant gain can be carried out by using different interleavers when it is possible and also multiple receiving antennas. This analysis is achieved for the case of two receiver but for a 2–dimension network more than two transmitters can be seen with a strong signal. The powerful of this research is that it can be extended to a general number of peers simultaneously discovered and since it is a link level solution, it can be endorsed by other system level algorithms to reduce the level of the congestion in the network, including distributed algorithms for avoiding collisions, controlling the allocation or orthogonalization of the physical resources and controlling the transmit power of each node to create a two–codewords interference scenario.

Bibliography

[1] ETSI, "DVB–T2 Call for Technologies," SB 1644r1, `http://www.dvb.org/technology/dvbt2/sb1644r1.01.T2_CfT.pdf`, Apr. 2007.

[2] ETSI, "Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television." ETSI EN 300 744 V1.5.1 (2004–11), Nov. 2004.

[3] ETSI "Digital Video Broadcasting (DVB) User guidelines for the second generation system for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB–S2)" ETSI TR 102 376 V1.1.1 (2005–02), Feb. 2005.

[4] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Trans. on Commun.*, vol. 16, pp. 1451–1458, Oct. 1998.

[5] ETSI, "Digital Video Broadcasting; Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering", ETSI EN 302 307 V1.1.1 (2004–06), June 2006.

[6] B. Le Floch, M. Alard, and C. Berrou, "Coded orthogonal frequency division multiplex," *Proc. IEEE,* vol. 83, no. 6, pp. 587–592, June 1986.

[7] C. Abdel Nour and C. Douillard, "Rotated QAM constellations to improve BICM performance for DVB–T2," *Proc. ISSSTA*, pp. 354–359, Aug. 2008.

[8] S. H. Han and J. H. Lee, "An overview of peak–to–average power ratio reduction techniques for multicarrier transmission," *IEEE Wireless Commun.,* vol. 12, no. 2, pp. 56–65, Apr. 2005.

[9] V. Mignone and A. Morello, "CD3–OFDM: a novel demodulation scheme for fixed and mobile receivers," *IEEE Trans. Commun.*, vol. 44, n. 9, Sep. 1996, pp. 1144–1151.

[10] G. J. Foschini and J. Salz, "Digital communications over fading radio channels," *Bell Syst. Tech. J.*, vol. 62, pp. 429–456, Feb. 1983.

[11] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space–time block codes from orthogonal design," *IEEE Trans. Info. Theory*, vol. 45, pp. 1456–1467, July 1999.

[12] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space–time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Trans. Info. Theory*, vol. 44, pp. 744–765, Mar. 1998.

[13] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "V–BLAST: an architecture for realizing very high data rates over the rich–scattering wireless channel," in *Proc. ISSSE*, 1998.

[14] B. Hassibi and H. Vikalo, "On sphere decoding algorithm I. Expected complexity," *IEEE Trans. Signal Proc.*, vol. 53, pp. 2806–2818, Aug. 2005.

[15] M. O. Damen, A. Chkief, and J.–C. Belfiore, "Lattice codes decoder for space–time codes," *IEEE Commun. Lett.*, vol. 4, pp. 161–163, May 2000.

[16] B. Hassibi and B. M. Hochwald, "High–rate codes that are linear in space and time," *IEEE Trans. on Info. Theory*, vol. 48, pp. 1804–1824, July 2002.

[17] R. W. Heath and A. J. Paulraj, "Linear dispersion codes for MIMO systems based on frame theory," *IEEE Trans. on Signal Proc.*, vol. 50, pp. 2429–2441, Oct. 2002.

[18] H. E. Gamal and M. O. Damen, "Universal space–time coding," *IEEE Trans. Info. Theory*, vol. 49, pp. 1097–1119, May 2003.

[19] J.–C. Belfiore, G. Rekaya, and E.Viterbo, "The Golden code: a 2×2 full–rate space–time code with non–vanishing determinants," *IEEE Trans. Info. Theory*, vol. 51, pp. 1432–1436, Apr. 2005.

[20] F. Oggier, G. Rekaya, J.–C. Belfiore, and E. Viterbo, "Perfect space–time block codes," *IEEE Trans. Info. Theory*, vol. 52, pp. 3885–3902, Sept. 2006.

[21] F. M. J. Willems, "Rotated and scaled Alamouti coding," *IEEE International Symposium on Information Theory (ISIT 2008)*, Toronto, Canada, July 2008.

[22] G. Benelli, "A new method for the integration of modulation and channel codingin an ARQ protocol," *IEEE Trans. Commun.*, vol. 41, pp. 1432–1436, Oct. 1992.

[23] H. S. Yao, G. W. Wornel, "Achieve to full MIMO diversity–multiplexing frontier with rotation–based space–time codes," in *Proc. Allerton Conf. Commun. Control, and Comput.*, Monticello, IL, Oct. 2003.

[24] L. Zheng and D. Tse, "Diversity and multiplexing: a fundamental tradeoff in multiple antenna channels," *IEEE Trans. Inform. Theory*, 49(5), pp. 1073–1096, May 2003.

[25] A. Wittneben, "A new bandwidth efficient transmit antenna modulation diversity scheme for linear digital modulation," in *Proc. IEEE Int. Conf. on Commun. (ICC)*, Geneva, 1993.

[26] N. Seshradri and J. H. Winters, "Two signaling schemes for improving the error performance of frequency–division–duplex (FDD) transmission systems using transmitter antenna diversity," *Int. J. Wireless Info. Networks*, vol. 1, pp. 24–47, Jan. 1994.

[27] J. H. Winters, "The diversity gain of transmit diversity in wireless systems with Rayleigh fading," in *Proc. IEEE Int. Conf. on Commun. (ICC)*, New Orleans, 1994.

[28] J. H. Winters, "The diversity gain of transmit diversity in wireless systems with Rayleigh fading," *IEEE Trans. on Vehic. Tech.*, vol. 47, pp. 119–123, 1998.

[29] S. Kaiser, "Spatial transmit diversity techniques for broadband OFDM systems," *in Proc. IEEE Globecom*, San Francisco, CA, 2000.

[30] A. Dammann and S. Kaiser, "Standard conformable antenna diversity techniques for OFDM and its application to the DVB–T system," in *Proc. IEEE Globecom*, San Antonio, TX, 2001.

[31] F. Oggier and E. Viterbo, "Algebraic Number Theory and Code Design for Rayleigh Fading Channel," *in Foundations and Trends in Communications and Information Theory*, vol. 1, pp. 333–415, 2004.

[32] M. O. Damen, H. E. Gamal, and G. Caire, "On maximum–likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, vol. 49, pp. 2389–2402, Oct. 2003.

[33] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge: University Press, 1985.

[34] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inf. Theory*, vol. 45, pp. 1639–1642, July 1999.

[35] D. Pham, K. R. Pattipati, P. K. Willett, J. Luo, "An improved complex sphere decoder for V–BLAST systems," *IEEE Sig. Proc. Lett.*, vol. 11, no. 9, pp. 748–751, Sept. 2004.

[36] B. Hochwald, S. ten Brink, "Achieving near–capacity on a multiple–antenna channel", *IEEE Trans. on Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[37] N. Benvenuto and G. Cherubini, *Algorithms for Communications Systems and their Applications.* Chichester: Wiley, 2002.

[38] M. S. Yee, "Max–log–map sphere decoder," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, Philadelphia, PA, Mar. 2005.

[39] J. Boutros, N. Gresset, L. Brunel, and M. Fossorier, "Soft–input soft–output lattice sphere decoder for linear channels," in *Proc. IEEE GLOBECOM San Francisco*, CA, pp. 1583–1587, Dec. 2003.

[40] M. Siti and M. P. Fitz, "A novel soft–output layered orthogonal lattice detector for multiple antenna communications," in *Proc. IEEE Int. Conf. Communications*, pp. 1686–1691, June 2006.

[41] C. Shen, M. P. Fitz, and M. Siti, "Generalized soft–output layered orthogonal lattice detector for golden code," in *Proc. IEEE Wireless Communications and Networking Conf.*, Mar. 2007.

[42] I. E. Telatar, "Capacity of multi–antenna Gaussian channels," technical memorandum, AT&T Bell Laboratories, June 1995.

[43] M. Eroz, F.–W. Sun and L.–N. Lee, "DVB–S2 low density parity check codes with near Shannon limit performance," in *Proc. Int. Journ. on Satellite Commun. Networks*, vol. 22, no. 3, May–June 2004.

[44] J. Du and B. Vucetic, "Trellis coded 16–QAM for fading channels," *European Trans. Telecom.*, vol. 4, no. 3, pp. 335–341, May–June 1993.

[45] G. Caire, G. Taricco and E. Biglieri, "Bit–interleaved coded modulation," *IEEE Trans. Inform. Theory*, vol. 44, pp. 927–946, May 1998.

[46] D. Rainish, "Diversity transform for fading channels," *IEEE Trans. Commun.*, vol. 44, pp. 1653–1661, Dec. 1996.

[47] J. Boutros and E. Viterbo, "Signal space diversity: a power and bandwidth efficient diversity technique for the Rayleigh fading channel," *IEEE Trans. on Info. Theory*, vol. 44, pp. 1453–1467, July 1998.

[48] R. Schober, L. H.–J. Lampe, W. H. Gerstacker and S. Pasupathy, "Modulation diversity for frequency–selective fading channels," *IEEE Trans. Info. Theory*, vol. 49, no. 9, pp. 2268–2276, Sept. 2003.

[49] J. Stott, "Re–mapped repetition," private communication, 4 June 2007.

[50] N. Miyazaki, Y. Hatakawa, T. Yamamoto, H. Ishikawa, T. Suzuki and K. Takeuchi, "A study on rotational OFDM transmission with multi–dimensional demodulator and twin turbo decoder," *VTC 2006 Fall*, TT–21 #3, Sept. 2006.

[51] C. Abdel Nour and C. Douillard, "On lowering the error floor of high order turbo BICM schemes over fading channels" *IEEE Global Telecommun. Conf.*, GLOBECOM, pp. 1–5, Nov. 2006.

[52] S. ten Brink, J. Speidel and R. Yan, "Iterative demapping and decoding for multilevel modulation," in *Proc. GLOBECOM*, pp. 579–584, Nov. 1998.

[53] F. Schreckenbach, N. Görtz, J. Hagenauer and G. Bauch, "Optimization of symbol mapping for bit–interleaved coded modulation with iterative decoding," *IEEE Commun. Lett.*, vol. 7, no. 12, pp. 593–595, Dec. 2003.

[54] R. G. Gallager, "Low density parity check codes," *IRE Trans. Inform. Theory*, vol. IT–8, pp. 21–28, Jan. 1962.

[55] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, 1991.

[56] Full diversity rotations, `http://www.telematica.polito.it/~viterbo/rotations/rotations.html`

[57] X.–Y. Hu, E. Eleftheriou, D.–M. Arnold, and A. Dholakia, "Efficient implementations of the sum–product algorithm for decoding LDPC codes," *in Proc. GLOBECOM 2001*, San Antonio, TX, pp. 1036–1040, Nov. 2001

[58] DVB, "A122: Framing structure, channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB–T2)," `http://www.dvb.org/technology/dvbt2/a122.tm3980r5.DVB-T2.pdf`, June 2008.

[59] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft–decision outputs and its applications," *Proc., IEEE Globecom Conf.* Dallas, TX, pp. 1680–1686, Nov. 1989.

[60] X. Li and J. A. Ritcey, "Bit–interleaved coded modulation with iterative decoding," in *Proc. Int. Conf. Communications*, pp. 858–862, June 1999.

[61] H. R. Sadjadpour, N. J. A. Sloane, M. Salehi and G. Nebe, "Interleaver design for turbo codes," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 831–837, May 2001.

[62] T. Richardson and R. Urbanke, *Modern Coding Theory.* Cambridge University Press, 2008.

[63] K. Zeger and A. Gersho, "Pseudo–Gray coding," *IEEE Trans. Commun.*, vol. 38, pp. 2147–2158, Dec. 1990.

[64] A. Goljahani, N. Benvenuto, S. Tomasin, and L. Vangelista, "Superimposed sequence versus pilot aided channel estimations for next generation DVB–T systems," *IEEE Trans. Broadcast.*, pp. 140–144, Mar. 2009.

[65] S. Tomasin, A. Gorokhov, H. Yang, and J.–P. Linnartz, "Iterative interference cancellation and channel estimation for mobile OFDM," *IEEE Trans. Wireless Commun.*, vol. 4, no. 1, pp. 238–245, Jan 2005.

[66] H. Minn and V. K. Bhargava, "An investigation into time–domain approach for OFDM channel estimation," *IEEE Trans. Broadcast.*, vol. 46, no. 4, pp. 240–248, Dec. 2000.

[67] H.–C. Wu, X. Huang, and D. Xu, "Novel semi–blind ICI equalization algorithm for wireless OFDM systems," *IEEE Trans. Broadcast.*, vol. 52, no. 2, pp. 211–218, June 2006.

[68] ETSI, "Generic Stream Encapsulation (GSE) Protocol," DVB Document A116, May 2007.

[69] O. Edfors, M. Sandell, J.–J. van de Beek, S. K. Wilson, and P. O. Borjesson, "OFDM channel estimation by singular value decomposition," *IEEE Trans. Commun.*, vol 46, no. 7, pp. 931–939, July 1998.

[70] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in OFDM systems," *IEEE Trans. Broadcast.*, vol. 48, no.3, Sep. 2002, pp. 223–229, Sept. 2002.

[71] S. Verdu, *Multiuser Detection.* Cambridge, UK: Cambridge University Press, 1998.

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

[73] M. Kobayashi, J. Boutros, and G. Caire, "Successive interference cancellation with SISO decoding and EM channel estimation," *IEEE J. Select. Areas Commun.,* vol. 19, pp. 1450–1460, Aug. 2001.

[74] J. G. Andrews, "Interference cancellation for cellular systems: a contemporary overview," *IEEE Wireless Communications*, vol. 12, no. 2, pp. 19–29, Apr. 2005.

[75] H. A. Loeliger, *On Euclidean space group codes*, Ph.D. dissertation Zurich, Switzerland: Swiss Federal Inst. Technol., 1992.

[76] N. Wiberg, *Codes and decoding on general graphs*, Ph.D. dissertation, Sweden: Linköping Univ., 1996.

[77] B. J. Frey, F. R. Kschischang, H.–A. Loeliger, and N. Wiberg, "Factor graphs and algorithms," in *Proc. 35th Allerton Conf. Communications, Control, and Computing* Monticello, IL: Allerton House, pp. 666–680, Sept. 29–Oct. 1 1997.

[78] F. R. Kschischang, B. J. Frey, and H.–A. Loeliger, "Factor graphs and the sum–product algorithm," *IEEE Trans. Inform. Theory,* vol. 47, pp. 498–519, Feb. 2001.

[79] C. Cozzo and B. L. Hughes, "Joint channel estimation and data symbol detection in space–time communications," *in Proc. Int. Conf. on Communications,* New Orleans, LA, June 2000.

[80] C. Komninakis and R. D. Wesel, "Joint iterative channel estimation and decoding in flat correlated Rayleigh fading," *IEEE J. Select. Areas Commun.,* vol. 19, pp. 1706–1717, Sept. 2001.

[81] J. Boutros and G. Caire, "Iterative multiuser joint decoding: Unified framework and asymptotic analysis," *IEEE Trans. Inform. Theory,* vol. 48, pp. 1772–1793, July 2002.

[82] M. Loncar, R. Muller, J. Wehinger, and T. Abe, "Iterative joint detection, decoding, and channel estimation for dual antenna arrays in frequency selective fading," *in Proc. 5th Int. Symp. Wireless Personal Multimedia Communication,* pp. 125–129, Honolulu, USA, Oct. 2002.

[83] I. Motedayen–Aval and A. Anastasopoulos, "Polynomial–complexity noncoherent symbol–by–symbol detection with application to adaptive iterative decoding of turbo–like codes," *IEEE Trans. Commun.,* vol. 51, no. 2, pp. 197–207, Feb. 2003.

[84] I. Sutskover, S. Shamai, and J. Ziv, "A Novel Approach to Iterative Joint Decoding and Phase Estimation," *Proc. 3rd International Symposium on Turbo Codes and Related Topics,* Brest, France, pp. 83–86, Sept. 1–5, 2003.

[85] J. Dauwels and H.-A. Loeliger, "Joint decoding and phase estimation: An exercise in factor graphs," *in Proc. IEEE Symp. Inf. Theory,* Yokohama, Japan, p. 231, Jul. 2003.

[86] J. Dauwels and H.-A. Loeliger, "Phase estimation by message passing," *in Proc. IEEE Int. Conf. Commun.* Paris, France, pp. 523–527, Jun. 2004.

[87] X. Jin, A. W. Eckford, and T. E. Fuja, "Analysis of joint channel estimation and LDPC decoding on block fading channels", *International Symposium on Information Theory and its Applications,* ISITA 04, Parma, Italy, October 10–13, 2004.

[88] H. Niu, M. Shen, J. A. Ritcey, and H. Liu, "A factor graph approach to iterative channel estimation and LDPC decoding over fading channels," *IEEE Trans. Wireless Commun.,* vol. 4, pp. 1345–1350, July 2005.

[89] Y. Zhu, D. Guo, and M. L. Honig, "A Message–Passing Approach for Joint Channel Estimation, Interference Mitigation and Decoding," *submitted*, Jan. 2009.

[90] N. Benvenuto and G. Sostrato, "Joint detection with low computational complexity for hybrid TD–CDMA systems," *IEEE J. Select. Areas Commun.,* vol. 19, no. 1, pp. 245–253, Jan. 2001.

[91] A. Amraoui, S. Dusad, and R. Urbanke, "Achieving general points in the 2–user Gaussian MAC without time–sharing or rate–splitting by means of iterative coding," in *Proc. IEEE Int. Symp. Information Theory* Lausanne, Switzerland, p. 334, Jun./Jul. 2002.

[92] P. Berlin and D. Tuninetti, "LDPC codes for fading Gaussian broadcast channels," *IEEE Trans. Inform. Theory*, vol. 51, pp. 2173–2182, June 2005.

[93] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory,* vol. IT–20, pp. 284–287, Mar. 1974.

[94] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error–correcting coding and decoding: turbocodes", *ICC '93,* Conference Record, Geneva, pp. 1064–1070, 1993.

[95] S.-Y. Chung, T. J. Richardson, and R. Urbanke, "Analysis of sum–product decoding of low–density parity–check codes using a Gaussian approximation," *IEEE Trans. Inform. Theory,* vol. 47, Feb. 2001.

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

[97] D. J. C. MacKay, and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electron. Lett.,* vol. 32, no. 18, pp. 1645–1646, Aug. 1996.

[98] E. Arikan, "Channel polarization: A method for constructing capacity–achieving codes for symmetric binary–input memoryless channels," *submitted to IEEE Trans. Inform. Theory,* 2008.

[99] J. D. Forney jr, "The Viterbi algorithm" *Proc. IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[100] R. Johannesson and K. S. Zigangirov, *Fundamentals of Convolutional Coding.* Piscataway, NJ: IEEE Press, 1999.

[101] T. Blu, P. L. Dragotti, M. Vetterli, P. Marziliano, and L. Coulot, "Sparse Sampling of Signal Innovations," *IEEE Signal Proc. Mag.*, pp. 31–40, March 2008.

[102] A. Vigato, S. Tomasin, L. Vangelista, N. Benvenuto, and V. Mignone, "Soft detection of modulation diversity schemes for next generation digital terrestrial television," *Proc. Int. Spread Spectrum Tech. and App. ISSSTA '08,* pp. 349–353, Bologna, Italy, Aug. 2008.

[103] A. Vigato, S. Tomasin, L. Vangelista, V. Mignone, N. Benvenuto, and A. Morello, "Coded decision direct demodulation for second generation digital video broadcasting standard," *IEEE Trans. Broadcast.*, vol. 55, no. 3, pp. 607–615, Sept. 2009.

[104] A. Vigato, N. Benvenuto, S. Tomasin, and L. Vangelista, "On hard and soft detection of space–time block codes by a novel soft output sphere decoder," *Proc. Int. Conf. Ultra Modern Telecom. ICUMT '09,* St.–Petersburg, Russia, Oct. 2009.