DISSERTATION

Design of Multiuser and Multi-Antenna Communications Systems Based on Graphical Models and Soft Information

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Doktors der technischen Wissenschaften

unter der Leitung von
Ao. Univ.-Prof. Dr. G. Matz
Institut für Nachrichtentechnik und Hochfrequenztechnik

eingereicht an der Technischen Universität Wien
Fakultät für Elektrotechnik

von
Clemens Novak
Hütteldorferstr. 168/20
1140 Wien

Wien, im Juni 2010

Die Begutachtung dieser Arbeit erfolgte durch:

1. Ao. Univ.-Prof. Dipl.-Ing. Dr. G. Matz

   Institut für Nachrichtentechnik und Hochfrequenztechnik
   Technische Universität Wien

2. Prof. Li Ping

   Department of Electronic Engineering
   City University of Kong Kong

Abstract

New products and services in the wireless market require higher data rates, improved link reliability, and the ability to cope with user interference. The use of multiple-input multiple-output (MIMO) communications has led to increased data rates and improved link reliability, and new multiuser transmission schemes such as interleave-division multiple access (IDMA) have been proposed. With IDMA, user separation is obtained via user-specific interleavers combined with low-rate channel coding. These new technologies and transmission schemes, however, require sophisticated and powerful receiver structures, which employ iterative algorithms.

In the first part of this thesis we use the factor graph framework to design iterative receiver algorithms for an IDMA multiuser uplink system employing MIMO orthogonal frequency-division multiplexing (OFDM). We derive the factor graph corresponding to the IDMA system, and use the sum-product algorithm to develop an iterative multiuser IDMA receiver that performs joint detection and channel estimation. Simulation results demonstrate large performance gains compared to classical receivers performing separate channel estimation and data detection. Suitable approximations to the messages passed by the sum-product algorithm yield a low complexity implementation that scales linearly with the number of users; further significant complexity reductions can be achieved by only updating a subset of messages in every iteration. We also perform an information theoretic performance analysis of IDMA and show that the performance of IDMA comes close to the information theoretic limit.

In the second part of the thesis we deal with quantization of messages in the factor graph. To this end we consider a bit-interleaved coded modulation (BICM) system and investigate the problem of how to quantize the log-likelihood ratios (LLRs) at the output of the demodulator. The optimal quantizer has previously been derived only for the special case of BPSK modulation over an AWGN channel. Extending this approach to other channels and modulations is difficult to implement in a practical system, and therefore we consider a different quantizer design which allows for simple implementation while only slightly degrading performance. We compare our quantizer design with the optimal quantizer in terms of maximum achievable rate and by means of BER simulations. We also propose a method for designing the different LLR quantizers during data transmission “on the fly”, i.e. without the need for lookup tables to store quantizer parameters. By designing demodulators, which directly calculate quantized LLRs, complexity reductions are possible. We demonstrate this effect by deriving a low-complexity soft-MMSE demodulator which outputs quantized LLRs.

In practical BICM systems, the channel between transmitter and receiver has to be estimated by means of pilot symbols. Because the pilot power is constrained, estimation errors occur. Recently, improved demodulators have been proposed, which take the channel uncertainty into account, thereby offering better performance. In the third part of this thesis, we provide a performance comparison of various demodulators for BICM with imperfect CSI in terms of the maximum achievable rate with a specific demodulator. In case of iterative decoded BICM (BICM-ID), we use EXIT charts to characterize the convergence behaviour of the MIMO-BICM-ID receivers. We also design LDPC codes that are matched to a specific demodulator in terms of their EXIT functions and provide BER comparisons. Finally, we investigate the impact of power allocation for pilots and data symbols on the maximum achievable rate and demonstrate the importance of correct power allocation.

Kurzfassung

Neue Produkte und Anwendungen in der drahtlosen Datenübertragung benötigen höhere Datenraten, Zuverlässigkeit und Robustheit gegenüber Interferenz von anderen Teilnehmern. Die Verwendung von *multiple-input multiple-output* (MIMO) Übertragung führte zur Steigerung der Datenrate und verbesserter Zuverlässigkeit, und neue Mehrbenutzer Übertragungsverfahren wie *interleave-division multiple access* (IDMA) wurden vorgeschlagen. Bei IDMA erfolgt die Separierung der Teilnehmer duch die Verwendung von userspezifischen Interleavern zusammen mit Kanalkodierung niederer Rate. Diese neuen Technologien und Übertragungsverfahren benötigen anspruchsvolle und leistungsfähige Empfängerstrukturen, welche iterative Algorithmen verwenden.

Im ersten Teil dieser Dissertation verwenden wir Faktorgraphen, um iterative Empfangsalgorithmen für ein IDMA Mehrbenutzersystem zu entwerfen, welches auf MIMO *orthogonal frequency-division multiplexing* (OFDM) basiert. Wir leiten den Faktorgraph des IDMA Systems her, und verwenden den *sum-product* Algorithmus, um einen iterativen IDMA Mehrbenutzerempfänger zu entwerfen, der gleichzeitig Detektion und Kanalschätzung durchführt. Simulationsergebnisse zeigen eine grosse Leistungsverbesserung gegenüber einem Empfänger, der zuerst den Kanal schätzt und dann die Daten detektiert. Eine Implementierung, deren Komplexität linear mit der Anzahl der Teilnehmer steigt, erhält man durch entsprechende Approximationen der Nachrichten, die im *sum-product* Algorithmus ausgetauscht werden. Eine weitere Reduktion der Komplexität wird durch Berechnen lediglich einer Teilmenge aller Nachrichten in jeder Iteration erreicht. Eine informationstheoretische Analyse des IDMA Systems zeigt, dass das System nahe des informationstheoretischen Limits arbeitet.

Der zweite Teil dieser Dissertation behandelt die Quantisierung von Nachrichten im Faktorgraphen. Wir betrachten ein BICM System und untersuchen das Problem der Quantisierung von *log-likelihood ratios* (LLRs) am Ausgang des Demodulators. Bisher wurde der optimale Quantisierer lediglich für den Spezialfall von BPSK Modulation über einen AWGN Kanal untersucht. Da eine Erweiterung dieser Lösung auf andere Kanälen und Modulationsformaten sehr kompliziert ist, schlagen wir eine alternative Quantisierungmethode vor. Diese lässt sich viel einfacher implementieren und liefert lediglich etwas schlechtere Ergebnisse. Wir vergleichen unsere Quantisierungsmethode mit dem optimalen Quantisierer bezüglich Transinformation und Bitfehlerrate. Weiters stellen wir eine Methode vor, mit der sich die Quanisierer während der Datenübertragung entwerfen lassen. Dadurch werden keine Tabellen zum Speichern der Parameter des Quantisierers benötigt. Die Verwendung von Demodulatoren, die direkt quantisierte LLRs berechnen und ausgeben, ermöglicht eine Verringerung der Komplexität des Demodulators. Wir demonstrieren diesen Effekt an einem *soft-MMSE* Demodulator, der direkt quantisierte LLRs berechnet.

In praktisch realisierten BICM Systemen wird der Kanal zwischen Sender und Empfänger mittels Übertragung von Pilotsymbolen geschätzt. Da die Sendeenergie begrenzt ist, treten Schätzfehler auf. In der Literatur wurden verbesserte Demodulatoren vorgestellt, die den Kanalschätzfehler berücksichtigen, und dadurch leistungsfähiger sind. Im dritten Teil dieser Dissertation vergleichen wir verschiedene Demodulatoren im Hinblick auf ihre maximal erreichbare Rate. Im Fall von iterativ demodulierter BICM Übertragung verwenden wir EXIT *charts*, um das Konvergenzverhalten des MIMO-BICM-ID Empfängers zu charakterisieren und entwerfen LDPC *codes*, die an den EXIT *chart* des entsprechenden Demodulators angepasst sind. Wir präsentieren einen Vergleich der Bitfehlerraten der verschiedenen Demodulatoren. Weiters untersuchen wir auch die Auswirkung der Leistungsaufteilung zwischen Pilot- und Datensymbolen auf die maximal erreichbare Rate und demonstrieren die Wichtigkeit der korrekten Aufteilung.

Acknowledgments

My time at the institute, and more specifically, in my research group, has been a very exciting one, both from a professional and personal point of view. I am very thankful to my supervisor Gerald Matz for his continuous support and guidance over the last several years. He provided me with a lot of freedom in my research and our numerous discussions have been essential for my professional and personal development.

It is also my pleasure to express my thanks to Li Ping who kindly agreed to act as a referee. I would also like to thank Franz Hlawatsch for introducing me to statistical signal processing and cosupervising me during the initial phase of my PhD.

I want to thank my colleagues at the institute, especially the people in my research group, for providing an exceptionally friendly and lively atmosphere. Our common activities have always been very fun and entertaining. I am particularly grateful to Peter Fertl, Johannes Maurer, Stefan Schwandter, and Valentin Schwarz for their collaboration and for many enlightening discussions from which I have benefited a lot.

Finally, I would like to express my deep gratitude to my parents. Their encouragement and confidence in me have been a great help. Very special thanks go to Martina for her love, patience and encouragement throughout the last years.

Contents

1

Introduction

WIRELESS communication has become ubiquitous in the last two decades. Initially, only high-end notebooks were equipped with connectivitiy to *Wireless Local Area Networks* (WLAN), which allowed for wireless broadband access to the internet. Due to increasing demand, WLAN connectivity is now common to all notebooks and many new consumer devices have been introduced to the market, which offer new services like multimedia streaming and video-on-demand. Many access points for WLAN have been installed in homes and offices, allowing for wireless internet access.

This development triggered customer demand for "wireless broadband access everywhere". Existing services, like surfing the internet or watching a video stream, are provided in a mobile environment. This creates new customers (e.g. people surfing the internet while commuting); extrapolating the success of wireless broadband by means of WLAN, this will be a huge market. In addition to providing existing services in a mobile fashion, also many new services have become available, which explicitly require mobile broadband access. One example for such a new service is "augmented reality", which enables users to obtain interactive information about sights near their current location by a clever combination of GPS enabled mobile devices, broadband internet access and a (user-generated) database accessible over the internet. Many of these new services are currently evolving, and will offer attractive opportunities and revenue models for service developers and network operators.

Current WLAN standards work reliably only up to 100 meters, making deployment in large areas, especially outdoors, difficult and costly. Given the infrastructure cellular network operators already have, it is only natural to extend current cellular communication technologies such that broadband internet access can be provided. Current second-generation cellular communication technologies like the *Global System for Mobile Communications* (GSM) are mostly tailored for voice-centric network traffic and perform poorly for data-centric applications. The third-generation cellular technology, *Universal Mobile Telecommunications System* (UMTS) and extensions, such as *High Speed Downlink Packet Access* (HSDPA) are better suited for data-centric network traffic. However, these new standards are not expected to be able to provide the increasing data rates and increased levels of quality of service (QoS) required in the future. Therefore new standards for the upcoming fourth-generation (4G) cellular communication systems have been proposed, which provide higher data rates and support QoS constraints. Most prominently, *Worldwide Interoperability for Microwave Access* (WiMAX) and *Long-Term Evolution* (LTE) of the 3GPP are currently considered or being already deployed.

Increased data rates and QoS constraints over the wireless channel pose three main challenges to system design and standard development:

(i) The spectrum usable for wireless communications is limited by various factors. Physical laws imply certain propagation conditions, thereby allowing only a portion of the available spectrum to be used for wireless communications. Also, process limitations in current semiconductor device technology limit the range of available frequencies. Spectrum regulation puts constraints on spectrum use, as frequency bands are sold to operators in auctions, and given the high prices for UMTS licenses paid in Europe, allocating more bandwidth poses a severe economical burden on operators.

(ii) The need for mobility poses a severe constraint on the energy consumption of wireless devices. Current rechargable batteries are highly complex and optimised components, but offer only limited capacity; novel energy sources like fuel cells are still being in the development phase.

(iii) Given the many mobile devices already deployed and the limited spectrum resources, interference between users is becoming stronger, thereby reducing data rates and limiting QoS. This implies that cooperation between different devices becomes more and more important.

Past strategies to overcome these challenges were increasing power levels or allocating more bandwidth to wireless devices, but these are not feasible anymore. Therefore new transmission schemes and algorithms are required, which allow for a much more efficient use of the limited resources available, promote cooperation between different wireless devices, and increase data rates and power efficiency to cope with tomorrows' communication requirements.

After the (re)discovery of low-density parity-check (LDPC) codes, transmission systems with iterative processing at the receivers received a lot of attention, as they promise large performance gains. Furthermore, they are reasonably simple to implement, because they consist of "standard" components, which are working together in an iterative manner. In the beginning, iterative systems were designed mainly based on intuition, but soon graphical models emerged as a very useful design tool.

1.1 Factor Graphs and Iterative Receivers

Graphical models allow a unified approach to many topics in coding, speech and signal processing, machine learning and statistical inference. Inference in graphical models is based on the actual graph, which describes the used model in an intuitive and simple manner, and on a generic message passing algorithm, which operates on the graph by exchanging messages between the graph's nodes. Initially, graphical models were mainly used in machine learning and statistical inference; [1] introduced Bayesian networks and the belief propagation algorithm, which operates on these graphs. Over time the range of applications widened, and many algorithms in seemingly different fields have been shown to be special cases of the generic message passing algorithm, known as the *sum-product algorithm*.

One of the first applications of graphical models in communications was coding. A class of very powerful codes are low-density parity-check (LDPC) codes, which were introduced by Gallager in [2]. LDPC codes, as many other codes, can be described by a particular form of graphical models, called “Tanner graphs” [3]. Tanner graphs are bipartite graphs, representing the code- and information bits and their dependencies induced by the code structure. It was also shown, that this graphical representation of LDPC codes allows to represent the decoding operation as a sum-product algorithm on the Tanner graph [3].

Factor graphs were introduced by Loeliger in [4, 5] and are generalizations of the Tanner graphs. Factor graphs are used to visualize a complicated “global” function (depending on many variables), which can be factored into a product of simpler “local” functions (depending on just a few variables). Factor graphs are bipartite graphs that express which variables are arguments of which “local” functions. In [6] it was argued, that problems dealing with such factorizable functions arise in many seemingly different fields, such as coding, speech and signal processing. The problems in these fields essentially require algorithms to calculate (approximate) marginal functions of the “global” function the factor graph is associated with. The factorization of the global function allows a computationally efficient implementation by means of the sum-product algorithm. The sum-product algorithm passes messages along the edges of the factor graph. At the nodes all the incoming messages from the neighbouring nodes are collected, and new messages are calculated by some message update rules. The newly updated messages are then propagated back to the neighbouring nodes. After some stopping criterion (which depends on the structure of the factor graph), the sum-product algorithm stops and the required marginal functions associated with every node can be approximately calculated from the messages received.

The idea of iterative processing at the receiver goes back to coding: In [7] turbo codes were introduced, which exhibit performance extremely close to the Shannon bound and were outperforming “classical codes” by large margins. Turbo codes consist of several convolutional *component* codes, each of which encodes the uniquely interleaved information bits. While the component codes can be decoded with low-complexity [8], large performance gains stem from the iterative decoding process, where the component decoders exchange information about the code bits. The success of turbo codes spurred enormous interest, and soon the underlying *turbo principle* [9] was applied to many other problems in communications, such as detection, channel estimation, and synchronization. In the context of bit-interleaved coded modulation (BICM) [10], iterative receivers were proposed in which demodulator and channel decoder work in an iterative fashion and exchange information about the transmitted bits [11]. Ideally, after each iteration more and more bits are correctly detected and - due to the *turbo principle* - help in correctly detecting the remaining unknown bits. These receiver algorithms can also be extended to incorporate channel estimation into the iterative processing; they obtain additional pilot symbols using already correctly detected bits [12], which can be used to estimate the channel. Ideally, these additional pilot symbols allow a more precise channel estimation, which helps in correctly detecting the remaining unknown bits.

One big advantage of all these receiver structures is that they do not need the design of a new transmission design; merely the receiver is redesigned and thereby gaining large improvements in performance. Due to the repeated application of e.g. the demodulator and the channel decoder, the complexity of iterative receivers is inherently higher than that of non iterative receivers. By designing the number of iterations performed at the receiver, a performance-complexity tradeoff can be achieved.

In the beginning, the *turbo principle* was used in an intuitive manner and many proposed algorithms were based on intuition. It soon became clear, however, that communication architectures can be expressed by means of factor graphs and that detection amounts to calculation of marginal functions of a complicated "global" function associated with the factor graph [13, 14]. The application of the factor graph framework to receiver design allows for a unified design and analysis. By approximating the messages to be exchanged between the nodes of the factor graph, the calculations involved in the message updates at the nodes can be considerably simplified. Furthermore, by updating and propagating only a subset of the messages further complexity reductions are possible. The factor graph framework allows a unified approach to these issues.

From a factor graph point of view, application of the sum-product algorithm to iterative receivers yield only an approximation of the marginal functions 8unless the factor graph is a tree), and considerable effort has been made to understand why iterative receivers are still offering such exceptional performance. Starting points for theoretical analysis were graph theory [15] and information geometry [16, 17] but a real breakthrough has not yet occured, thus making understanding of iterative systems an open research issue.

1.2 Scope of Work and Contributions

This thesis consists of three parts: The first part applies the factor graph framework to the design of an iterative receiver in multiuser systems. We consider an uplink scenario where several users transmit data to a base station and the users are separated by means of *interleave-division multiple access* (IDMA) [18]. Using the factor graph framework [4, 5], we can represent the IDMA system as a bipartite graph, consisting of nodes and connecting edges. By formulating the detection problem as marginalization, the sum-product algorithm can be used to develop an iterative multiuser MIMO-IDMA receiver that performs joint detection and channel estimation. The proposed algorithm calculates messages which are exchanged between the nodes of the factor graph.

The second part of this thesis discusses message quantization in the context of bit-interleaved coded modulation (BICM). We propose different quantization schemes and assess their impact on performance by information theoretic quantities. This is relevant in the context of receiver design in cooperative communication networks. Here, nodes can cooperate or use relays, in order to increase the total throughput of the network or enhance reliability of the transmissions. Cooperation schemes work by updating received messages from other nodes and transmitting the updated messages to other nodes or relays. Practical considerations require the quantization of messages to be exchanged between the nodes.

In the third part of this thesis, we consider the effects of imperfect CSI in case of BICM systems. In practical BICM systems, the channel between transmitter and receiver has to be estimated. Usually, this is done by means of pilot symbols, which are used by the receiver to estimate the channel. Because the pilot power is constrained, estimation errors occur, which result in the receiver having only imperfect CSI. We propose improved demodulator algorithms for this scenario and study their performance and compare these algorithms with conventional approaches.

Our contributions are as follows.

Graphical Models in Wireless Communications

- *Iterative MIMO-OFDM-IDMA receiver.* We propose a receiver for a MIMO-OFDM-IDMA system that can be viewed as an approximation to the maximum *a posteriori* (MAP) bit detector. The MAP bit detector is optimal in the sense of minimum probability of a bit error; its output is chosen such that the *a posteriori* probability of an information bit is maximized. This probability is obtained by a marginalization which, using the factor graph framework, is carried out efficiently by means of the sum-product algorithm. The complexity of this marginalization is exponential in the number of users, thus making it unrealizable even for moderate system sizes. We overcome this limitation by using approximations for some of the messages propagated in the factor graph, thereby obtaining a complexity that is linear in the number of users. Our receiver extends that of [18] to MIMO transmissions.

- *Higher-order modulation.* We furthermore extend the receiver of [18] to higher-order modulation formats, which results in a further increase of spectral efficiency (in addition to the MIMO multiplexing gain).

- *Selective message updating.* We propose a selective message update scheme where only certain messages are updated in each iteration. This yields a reduction of computational complexity and makes it possible to trade error performance against computational efficiency in a flexible manner.

- *Integrated channel estimation.* The iterative IDMA multiuser detector proposed in [18] assumes perfect channel state information (CSI) at the receiver. In practice, pilot-aided channel estimation is usually employed to obtain (imperfect) CSI, which is used by the multiuser detector instead of the true CSI. In our factor graph based receiver, on the other hand, pilot-aided channel estimation is an integral part of the iterative scheme. Thus, our receiver performs iterative joint multiuser data detection and channel estimation for pilot-assisted MIMO-IDMA (see [12, 19] for related work in the context of single-user systems).

- *Information-theoretic performance analysis.* Following the idea of [20], we consider the maximum achievable rate of IDMA as a fundamental performance limit and compare it with the information-theoretic capacity of the corresponding multiple access channel (MAC). To obtain the maximum achievable rate of IDMA we use the low-complexity IDMA receiver with an LDPC code as channel code and determine the achievable rate by means of BER simulations. It is shown that in our scenario, IDMA can perform within 2 dB of the information theoretic limit.

- *Numerical performance analysis.* Finally, we use numerical simulations to demonstrate the performance gains achieved with the proposed receiver relative to conventional IDMA receivers and the dependence of performance on certain system parameters. In particular, it is observed, that inclusion of channel estimation in the iterative detection/decoding scheme yields a dramatic improvement of reliability, and selective message updating results in a significant reduction of complexity.

Message Quantization in BICM Systems

- *BICM systems with quantized LLRs.* We consider a BICM system, where the LLRs calculated by the demodulator are quantized. We investigate the problem of quantizer design in terms of an equivalent discrete channel. The optimal quantizer for the special case of BPSK modulation over an AWGN channel has been proposed in [21], but extending this approach to other channels and modulations is very difficult to implement in a practical system. Thus, we consider a different quantizer design which allows for simple implementation while only slightly degrading information rate.

- *Performance evaluation.* To compare the optimum and proposed quantizer design in a *code-independent* manner, we propose to use the mutual information of the equivalent modulation channel of the BICM system [20]. In case of a fast-fading scenario, we use the ergodic rate, whereas for quasi-static fading we characterize the quantizer by means of outage probability. In case of BPSK-modulated single-input single-output (SISO) systems we use a semi-analytical approach, otherwise Monte-Carlo simulations.

- *On-the-fly quantizer design.* The quantizer parameters, namely quantizer intervals and quantizer outputs, depend on the system configuration and the SNR, requiring large lookup tables for storing the required parameters. We propose a method for designing the proposed LLR quantizer during data transmission, "on the fly" i.e. without the need for lookup tables to store quantizer parameters.

- *Numerical performance analysis.* We provide bit error rate (BER) simulations for BICM systems with LLR quantization using *low-density parity-check (LDPC)* codes, which compare different quantization schemes and demonstrate the effectiveness of our "on the fly" quantizer design.

- *Derivation of a low-complexity demodulator.* By designing demodulators which directly calculate quantized LLRs, complexity reductions are possible. We derive a low-complexity soft-MMSE demodualtor which outputs quantized LLRs, demonstrate its complexity savings and performance losses due to quantization.

Performance of BICM Systems with Imperfect CSI

- *Demodulator design for imperfect CSI.* We extend the results of the improved soft-MMSE demodulator [22] to arbitrary linear estimators and show that the improved soft-MMSE demodulator is independent of the actual linear estimator used. The improved BICM-ID demodulator of [23] is extended to MIMO channels with arbitrary spatial correlation.

- *Performance evaluation.* We again use the maximum rate achievable with a specific demodulator as a code-independent performance metric. These rates are measured by Monte-Carlo simulations of the equivalent BICM modulation channel [20]. We compare these maximum achievable rates of mismatched and improved low-complexity soft-MMSE receivers for correlated and uncorrelated MIMO channels. In case of BICM-ID, we use EXIT charts [24] to characterize the convergence behaviour of the MIMO-BICM-ID receivers employing the mismatched demodulator or the improved demodulator for different symbol mappings and different channel correlation models.

- *Impact of power allocation.* For BICM, we investigate the impact of power allocation between pilot and data symbol power on the maximum achievable rate and demonstrate the importance of correct power allocation.

- *Numerical performance analysis.* For BICM-ID, we design LDPC codes that are matched to a specific demodulator in terms of their EXIT functions, by using the approach from [25]. We also provide BER comparisons using the optimized LDPC codes and a "standard" (i.e., non-optimized) LDPC code.

1.3 Outline of the Thesis

- In **Chapter 2** we give an introduction to basic communication principles and give an overview of the state of the art. The chapter contains introductory material about MIMO, BICM, OFDM, and graphical models. The presented material serves as basis for the subsequent chapters.

- **Chapter 3** presents the design of iterative MIMO-OFDM-IDMA receivers using the factor graph framework. We construct the factor graph of the system and derive the messages to be propagated along the edges of the factor graph, and develop a selective message updating scheme. Finally, we provide information-theoretic performance limits of the proposed system and demonstrate the performance of the proposed receiver structures and algorithms by means of numerical simulations.

- **Chapter 4** investigates LLR quantization in BICM receivers. We present the system model and propose different quantizer designs. Next, we study the system capacity of SISO- and MIMO-BICM systems, respectively and propose the on-the-fly design of the quantizer. Finally, we propose a low-complexity soft-MMSE demodulator, which directly calculates quantized LLRs.

- In **Chapter 5** we develop demodulators for BICM systems with imperfect CSI. By taking the channel uncertainty into account, modified demodulator metrics are derived, which yield better performance. We also obtain these modified demodulators by means of the factor graph approach. We present numerical results, which demonstrate the performance difference between conventional and modified demodulators.

- Finally, **Chapter 6** summarizes the main contributions of this thesis and provides an outlook on future topics of interest.

2

Preliminaries

IN this chapter we introduce basic concepts of wireless communications and graphical models. We give an overview of MIMO wireless communications, BICM, and multiuser systems, and we introduce graphical models, especially factor graphs and the sum-product algorithm. Finally, we present some exemplary applications of graphical models to problems in signal processing and communications.

2.1 Wireless Communications

Wireless devices have become ubiquitous in the last decades, providing many benefits to its users. From an engineering point of view, the wireless channel introduces new challenges like fading and interference. These challenges have to be dealt with properly and in the following subsections we will explain some of the concepts used to overcome these challenges.

2.1.1 Wireless Propagation Channels

Wireless channels differ from wired channels by *multipath propagation*, i.e. by several propagation paths from the transmitter to the receiver, where the signal is reflected, diffracted, or scattered along the way. In addition, the propagation conditions change over time because of mobility of the receiver or the transmitter (or both) and because the environment undergoes changes as well. From a system-theoretic point of view, a wireless channel is modeled as a linear time-variant (LTV) system [26]. The system output (the received signal) $y(t)$ is given as

$$y(t) = \int h(t,\tau)x(t-\tau)d\tau \, ,$$

where $x(t)$ denotes the system input (the transmitted signal). The *time-variant* impulse response is denoted by $h(t,\tau)$; τ represents the multipath delay and t represents time. We take the Fourier transform of $h(t,\tau)$ with respect to t, and obtain the *deterministic spreading function*,

$$S(\nu,\tau) = \int h(t,\tau)e^{-j2\pi\nu t}dt \, ,$$

which characterizes the channel response in the Delay-Doppler-domain (ν denotes Doppler frequency). Given the random nature of the wireless channel, the impulse response $h(t,\tau)$ is characterized statistically by the autocorrelation function

$$\mathrm{E}\{h^\star(t_1,\tau_1)h(t_2,\tau_2)\} \, ,$$

where $\mathrm{E}\{\cdot\}$ denotes the expectation operation. Most channels fulfill the wide-sense stationarity (WSS) assumpion, so that the joint statistics of a channel measured at two different times t_1 and t_2 depends only on the difference $\Delta t = t_2 - t_1$. The autocorrelation function then becomes

$$R_h(\Delta t,\tau_1,\tau_2) = \mathrm{E}\{h^\star(t,\tau_1)h(t+\Delta t,\tau_2)\} \, .$$

In many wireless environments the different scatterers can be assumed to be uncorrelated. Therefore, the channel impulse response associated with a multipath component at delay τ_1 is uncorrelated with the impulse response associated with a different multipath component at delay $\tau_2 \neq \tau_1$. This assumption is called the *uncorrelated scaterers* (US) assumption. Combining it with the WSS assumptions yields the WSSUS assumption [27] which implies for the autocorrelation function

$$\mathrm{E}\{h^\star(t,\tau)h(t+\Delta t,\tau')\} = C_h(\Delta t,\tau)\delta(\tau-\tau') \, .$$

Here $C_h(\Delta t,\tau)$ captures the autocorrelation of the channel as a function of the multipath delay τ and the observation time difference Δt. Analogous to the definition of the deterministic scattering function, we define the *scattering function* $S_h(\nu,\tau)$ as the Fourier transform of the autocorrelation of the channel with respect to the observation time difference Δt,

$$S_h(\nu,\tau) = \int C_h(\Delta t,\tau)e^{-j2\pi\nu\Delta t}d\Delta t \, .$$

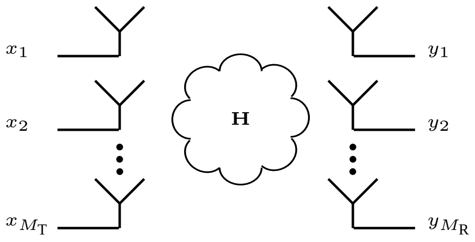


Figure 2.1: MIMO system model.

2.1.2 Multiple Antenna Systems

Multiple input multiple output (MIMO) systems have been introduced in [28]. They provide large gains in terms of rate, reliability and flexbility and thus spurred enormous research interest.

An $M_\mathrm{T} \times M_\mathrm{R}$ MIMO system is shown in Fig. 2.1, where transmitter and receiver are equipped with M_T and M_R antennas, respectively. Under ideal conditions, a MIMO system provides at least $M_\mathrm{T}M_\mathrm{R}$ independent paths between transmitter and receiver, which can be exploited in various ways. The first strategy is focused on increasing reliability, which is measured via *outage probability*. A wireless communication system is said to be in outage, if the channel is in a deep fade, and therefore no data can be transmitted. Given $M_\mathrm{T}M_\mathrm{R}$ independent paths in a MIMO system, the probability that all of them are in a deep fade (or in outage), is much lower than in a classical single-input single-output (SISO) system, leading to a much lower outage probability of the MIMO system. If a transmission scheme is designed to take advantage of these independent paths, this so called *diversity gain* of $M_\mathrm{T}M_\mathrm{R}$ can be exploited. Contrary to the first strategy, the second strategy focusses on increasing the throughput of the system. Here the transmitter sends a different data stream on every antenna, and if $M_\mathrm{R} \geq M_\mathrm{T}$, the receiver can separate these streams with high probability. The gain offered by this stratey is termed *multiplexing gain*. Under ideal conditions (and if $M_\mathrm{R} \geq M_\mathrm{T}$) it equals M_T and corresponds to an M_T-fold increase in data rate as compared to a SISO system. Both gains can only be provided by independent paths between transmitter and receiver. In rich scattering environments [29] this assumption is valid, but in certain scenarios (such as environments with a strong line of sight component or pinhole channels) the paths are no longer independent, thereby reducing the MIMO gains. Finally, we remark, that it is not possible to design transmission systems with maximum multiplexing and diversity gain; merely a tradeoff between the two gains can be achievd [30]. Space-time coded system [31–33] offer such a tradeoff.

System Model. At symbol time n the transmitter sends the vector $\mathbf{x}[n] = (x_1[n], \dots, x_{M_\mathrm{T}}[n])^T$. Its elements $x_k[n]$ are zero-mean, statistically independent, and uniformly drawn from the possibly complex symbol alphabet $\mathcal{A}$ of cardinality $|\mathcal{A}|$. The mean transmit energy of the transmit vector $\mathbf{x}[n]$ is given by $E\{\|\mathbf{x}[n]\|^2\} = E_x$.

The impulse response of the wireless channel between transmitter and receiver is modeled as a length-L_c sequence of $M_\mathrm{R} \times M_\mathrm{T}$ channel matrices $\mathbf{H}[n]$. In the complex base band, the receiver observes at symbol time n the length-M_R vector $\mathbf{y}[n] = (y_1[n], \dots, y_{M_\mathrm{R}}[n])^T$ according to

$$\mathbf{y}[n] = \sum_{k=0}^{L_c} \mathbf{H}[k]\mathbf{x}[n-k] + \mathbf{w}[n] \,, \tag{2.1}$$

where $\mathbf{w}[n]$ denotes zero-mean, white complex Gaussian noise, $\mathbf{w}[n] \sim \mathcal{CN}(\mathbf{0}, \sigma_w^2 \mathbf{I})$. If the impulse response of the wireless channel is very short compared to the symbol duration, we speak of a *narrow-band system* and the input-output relation (2.1) simplifies to the *flat-fading* model,

$$\mathbf{y}[n] = \mathbf{H}[n]\mathbf{x}[n] + \mathbf{w}[n] \,. \tag{2.2}$$

In case of many scatterers which are randomly distributed, a common model for a non line-of-sight connection between the transmitter and the receiver is the i.i.d. Rayleigh fading model. In this case the elements $h_{i,j}[n]$ of the matrix $\mathbf{H}[n]$ are i.i.d. circularly symmetric complex Gaussian random variables of unit variance, $\mathbf{h}[n] = \text{vec}\{\mathbf{H}[n]\} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$. If the antenna spacing at the transmitter and/or at the receiver becomes smaller than the wavelength[1], the i.i.d. assumption of the channel coefficients is not valid anymore, and the channel coefficients $h_{i,j}[n]$ become correlated. A widely used model for this scenario is the Kronecker model [34], where $\mathbf{h}[n] \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}_h)$ and the correlation matrix $\mathbf{C}_h$ is given by

$$\mathbf{C}_h = \mathbf{T}^{1/2} \otimes \mathbf{R}^{1/2} \,,$$

with the transmit and receive correlation matrices $\mathbf{T}$ and $\mathbf{R}$, respectively.

2.1.3 Bit-interleaved Coded Modulation

Traditional coding schemes for error correction, such as block or convolutional codes, provide coding gain at the expense of increased bandwidth or reduced data rate. Trellis coded modulation, introduced in [35], jointly optimizes both channel coding and modulation. This joint optimization yields significant coding gains without bandwidth expansion, thereby enabling spectrally efficient transmission. However, the extension of trellis coded modulation to fading channels proved to be difficult [36].

Bit-interleaved coded modulation [10], however, follows a different approach: By interleaving the coded bits before symbol mapping, the coded bits become de-facto independent, and channel code and modulation can be designed and optimized separately. This breaks the coded modulation paradigm of joint design of coding and modulation, but it provides better performance. This separate design furthermore allows an easy extension of BICM to MIMO, as only the modulator and demodulator parts of the system have to be adapted. BICM also opens up the possibility of iterative processing at the receiver, thereby offering even better performance [37]. Furthermore, analytical tools for evaluating the performance of BICM and design guidelines for good performance are known [10, 11].

System Model. A block diagram of a MIMO-BICM system is shown in Fig. 2.2. A sequence of N' information bits $b[n']$ $(n' = 1, \dots, N')$ is encoded using an error-correcting code with rate R, passed through a *bitwise interleaver* Π and then scrambled by a pseudo-random sequence $p[n']$. The interleaved and scrambled code bits $c[n']$ are uniformly distributed and demultiplexed into M_T antenna streams ("layers"), denoted by $c_l[n]$, $l = 1, \dots, M_\text{T}$. In each layer, groups of m code bits are mapped to (complex) data symbols $x_l[n] \in \mathcal{A}$, $l = 1, \dots, M_\text{T}$. The transmit vector at symbol time n is given by $\mathbf{x}[n] = (x_1[n] \dots x_{M_\text{T}}[n])^T$ and carries

[1] This might happen, for example, when small mobile devices are equipped with multiple antennas

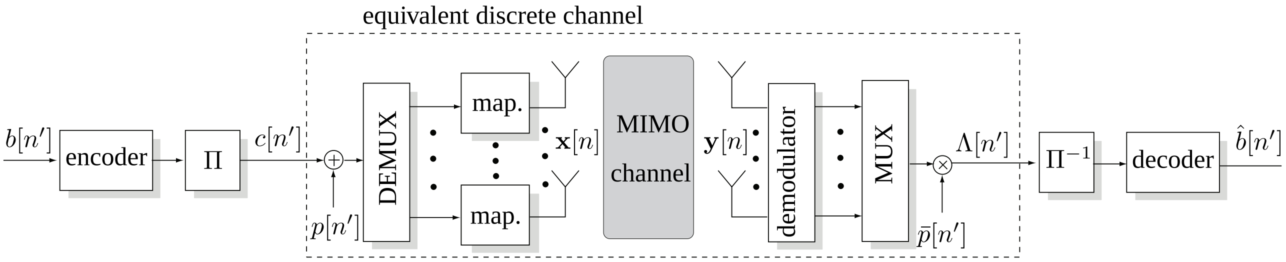


Figure 2.2: Block diagram of a MIMO-BICM system.

$R_0 = mM_\mathrm{T}$ interleaved code bits $c_l[n]$. The bits $c_l[n]$ corresponding to the symbol vector $\mathbf{x}[n]$ will also be denoted by $\mathbf{c}[n]$, and by using the symbol mapping function χ, we can write $\mathbf{x}[n] = \chi(\mathbf{c}[n])$. Finally, we have N'/R coded bits, and therefore a total number of $N \triangleq N'/(mRM_\mathrm{T})$ transmit symbols $\mathbf{x}[n]$.

Receiver. The receiver consists of a demodulator, a de-interleaver, and a channel decoder. The receive vector $\mathbf{y}[n]$ is given by (cf. (2.2))

$$\mathbf{y}[n] = \mathbf{H}[n]\mathbf{x}[n] + \mathbf{w}[n]\,.$$

The demodulator calculates (possibly approximate) log-likelihood ratios (LLRs) Λ_l for the code bits $c_l[n]$ according to [38]

$$\Lambda_l[n] = \log \frac{\Pr(c_l[n]{=}1|\mathbf{y}[n],\mathbf{H}[n])}{\Pr(c_l[n]{=}0|\mathbf{y}[n],\mathbf{H}[n])}\,, \tag{2.3}$$

where $\Pr(c_l[n]{=}c|\mathbf{y},\mathbf{H})$ denotes the posterior probability of the bit $c_l = c$ given the received vector $\mathbf{y}[n]$ and the channel matrix $\mathbf{H}[n]$. We can obtain this posterior probability as (we omit the time index n for notational convenience)

$$\Pr(c_l = u|\mathbf{y},\mathbf{H}) = \sum_{\mathbf{c}:c_l=u} \Pr(\mathbf{c}|\mathbf{y},\mathbf{H}) \propto \sum_{\mathbf{x}\in\chi_l^u} f(\mathbf{y}|\mathbf{x},\mathbf{H})f(\mathbf{x})\,, \tag{2.4}$$

where $f(\mathbf{y}|\mathbf{c},\mathbf{H})$ is the likelihood function of the channel, χ_l^u denotes the set of transmit vectors for which $c_l = u$, and $f(\mathbf{x})$ denotes the *a priori* probability of the symbol $\mathbf{x}$. In case of equally likely symbols, we further obtain

$$\Pr(c_l = u|\mathbf{y},\mathbf{H}) \propto \sum_{\mathbf{x}\in\chi_l^u} f(\mathbf{y}|\mathbf{x},\mathbf{H})\,. \tag{2.5}$$

From the MIMO system model (2.2), we obtain for the likelihood function of the channel

$$\mathbf{y}|\mathbf{x},\mathbf{H} \sim \mathcal{CN}(\mathbf{Hx},\sigma_w^2\mathbf{I})\,,$$

and the LLR expression (2.3) becomes

$$\Lambda_l = \log \frac{\sum\limits_{\mathbf{x}\in\chi_l^1} \exp\big(-\frac{1}{\sigma_w^2}\|\mathbf{y}-\mathbf{Hx}\|^2\big)}{\sum\limits_{\mathbf{x}\in\chi_l^0} \exp\big(-\frac{1}{\sigma_w^2}\|\mathbf{y}-\mathbf{Hx}\|^2\big)}\,. \tag{2.6}$$

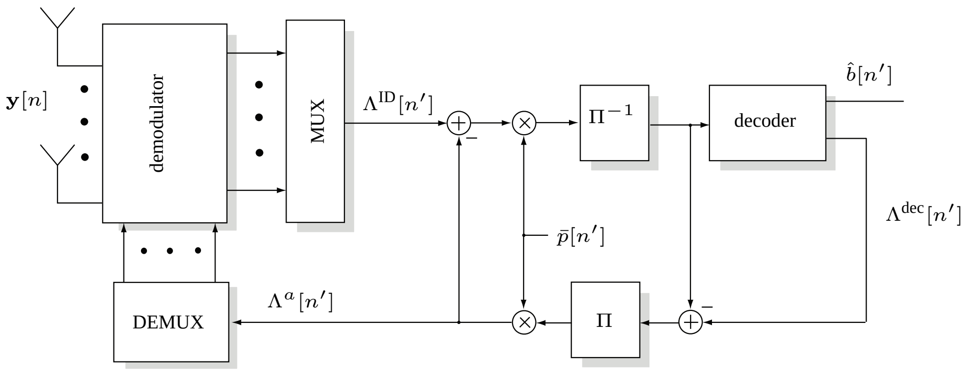


Figure 2.3: Block diagram of an iterative MIMO-BICM receiver.

Using the *max-log approximation*[2] we can simplify (2.6) to

$$\Lambda_l \approx \frac{1}{\sigma_w^2}\left[\min_{\mathbf{x}\in\chi_l^0}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2 - \min_{\mathbf{x}\in\chi_l^1}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2\right]. \tag{2.7}$$

In passing, we note that the sign of the LLR Λ holds information about which bit value (0 or 1) of c_l is more likely, while the magnitude $|\Lambda|$ measures the confidence the demodulator has about this decision. Therefore the LLRs can attain any value in the range $(-\infty,\infty)$.

Finally, the LLRs provided by the demodulator are de-scrambled by the sequence $\bar{p}_l[n] = 1-2p_l[n]$, deinterleaved and fed into the channel decoder. Based on the code structure, the channel decoder calculates LLRs for the information and code bits. The LLRs for the information bits are sliced and yield the bit estimates $\hat{b}[n']$.

Iteratively decoded BICM. Expression (2.4) holds the key to the design of iterative BICM receivers: Information from already decoded bits is reused as a priori information in the demodulator. After decoding, the so called *extrinsic* LLRs are obtained by subtracting the (interleaved) LLRs at the input of the channel decoder from the decoder LLRs $\Lambda^{\mathrm{dec}}[n']$. These extrinsic LLRs are interleaved, scrambled and fed back to the demodulator as the *a priori* LLRs $\Lambda^{a}[n']$ which are used to approximate the probability $f(\mathbf{x})$. Of course, these fed back LLRs are not “independent” new information, but in case of large block lengths and a random interleaver, their independence is a good approximation. Starting from (2.4), we have

$$\Pr(c_l = u|\mathbf{y},\mathbf{H}) \propto \sum_{\mathbf{x}\in\chi_l^u} f(\mathbf{y}|\mathbf{x},\mathbf{H}) f(\mathbf{x}) \approx \sum_{\mathbf{x}\in\chi_l^u} f(\mathbf{y}|\mathbf{x},\mathbf{H}) \prod_{k=1}^{mM_{\mathrm{T}}} p(c_k(\mathbf{x})), \tag{2.8}$$

where $c_k(\mathbf{x})$ denotes the kth bit label of the symbol vector $\mathbf{x}$ and we assumed that the interleaved code bits $c[n']$ are independent. The probability $p(c_k(\mathbf{x}))$ of a code bit is approximated from the *a priori* LLR Λ_k^a fed back by the decoder,

$$p(c_k(\mathbf{x})) \approx \frac{\exp(c_k(\mathbf{x})\Lambda_k^a)}{1+\exp(\Lambda_k^a)}. \tag{2.9}$$

[2] That is $\log(a+b) \approx \log(\max(a,b))$ for $a \ll b$ or $b \ll a$

The demodulator calculates LLRs in a similar manner as in (2.3), yielding

$$\Lambda_l^{\text{ID}} = \log \frac{\sum\limits_{\mathbf{x}\in\chi_l^1} f(\mathbf{y}|\mathbf{x},\mathbf{H}) \prod_{k=1}^{mM_{\text{T}}} \exp(c_k(\mathbf{x})\Lambda_k^a)}{\sum\limits_{\mathbf{x}\in\chi_l^0} f(\mathbf{y}|\mathbf{x},\mathbf{H}) \prod_{k=1}^{mM_{\text{T}}} \exp(c_k(\mathbf{x})\Lambda_k^a)} \,. \tag{2.10}$$

We caution the reader that this LLR is not a fraction of two probabilities as are the LLRs defined in (2.3), because of the approximation for $p(c_l(\mathbf{x}) = u)$ in (2.9). In a similar manner as before, the max-log approximation can be used to obtain the following LLR expression of the demodulator

$$\Lambda_l^{\text{ID}} \approx \min_{\mathbf{x}\in\chi_l^0} \left(\frac{1}{\sigma_w^2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \frac{1}{2} \sum_{l'} (2c_{l'}(\mathbf{x}) - 1)\Lambda_{l'}^a \right) - \min_{\mathbf{x}\in\chi_l^1} \left(\frac{1}{\sigma_w^2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \frac{1}{2} \sum_{l'} (2c_{l'}(\mathbf{x}) - 1)\Lambda_{l'}^a \right).$$

Closer inspection of above expression reveals, that Λ_l^{ID} can be expressed as

$$\begin{aligned} \Lambda_l^{\text{ID}} &= \Lambda_l^a + \min_{\mathbf{x}\in\chi_l^0} \left(\frac{1}{\sigma_w^2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \frac{1}{2} \sum_{l'\neq l} (2c_{l'}(\mathbf{x}) - 1)\Lambda_{l'}^a \right) \\ &\qquad - \min_{\mathbf{x}\in\chi_l^1} \left(\frac{1}{\sigma_w^2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 + \frac{1}{2} \sum_{l'\neq l} (2c_{l'}(\mathbf{x}) - 1)\Lambda_{l'}^a \right) \\ &\triangleq \Lambda_l^a + \Lambda_l^{\text{demod}} \,. \end{aligned}$$

This shows that the demodulator outputs the sum of the *a priori* LLR Λ_l^a and the *extrinsic* LLR Λ_l^{demod}. To avoid a positive feedback loop, only the *extrinsic* LLRs Λ_l^{demod} of the demodulator are fed into the deinterleaver, which are then decoded by the channel decoder [11]. This leads to the structure of the iterative BICM receiver as shown in Fig. 2.3: The *a priori* LLR Λ_l^a are subtracted from the LLRs calculated by the demodulator Λ_l^{ID} to yield the *extrinsic* LLRs Λ_l^{demod}. These LLRs are descrambled, deinterleaved, and decoded by the channel decoder, which outputs posterior LLRs $\Lambda^{\text{dec}}[n']$. By subtracting the input LLRs of the channel decoder, the extrinsic LLRs from the channel decoder are obtained. These are interleaved, scrambled and become the *a priori* LLRs Λ_l^a for the demodulator. The channel decoder outputs also bit decisions $\hat{b}[n']$ for the information bits, which are only used in the last iteration as final decisions of the iterative BICM receiver.

To demonstrate the performance improvement of iteratively decoded MIMO-BICM, we simulated the BER versus SNR performance of a 2×2 MIMO-BICM system, using a 16QAM symbol alphabet with Gray mapping. The channel code was a convolutional code with octal generator polynom $[13\,15]_8$. Fig. 2.4 shows the BER results for a non-iterative BICM receiver and an iterative receiver performing 10 iterations between demapping and decoding. It can be seen that the iterative receiver outperforms the non-iterative receiver by several dB: the SNR gap between the two receivers is about $5\,$dB at a BER of 10^{-4} and increases to about $7\,$dB at a BER of 10^{-5}. We also show the MIMO-BICM system capacity [10] as an absolute performance limit. At a BER of 10^{-4}, the iterative receiver operates about $2\,$dB away from this limit, whereas we observe a gap of almost $8\,$dB for the non-iterative BICM receiver. Employing more powerful codes can further reduce the gap of the iterative receiver to system capacity. In Chapter 5 we will demonstrate the performance gains of LDPC codes which have been matched to the demodulator characteristics.

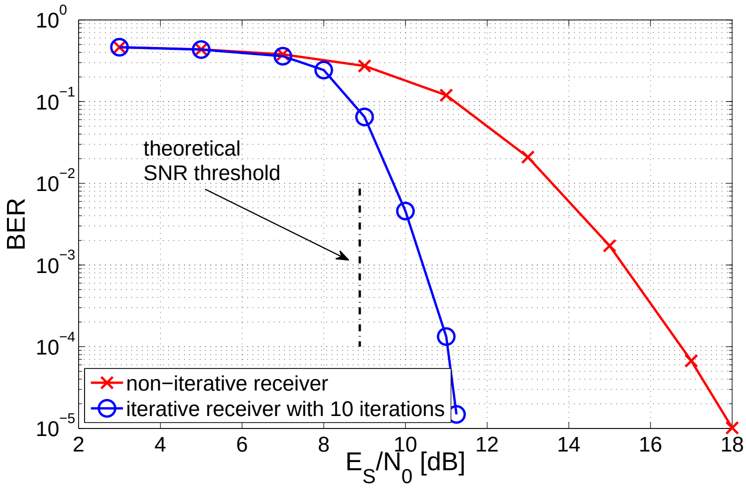


Figure 2.4: Performance of 2×2 MIMO-BICM receiver with different receiver architectures.

Capacity Measures. The performance of MIMO-BICM systems is often illustrated by means of BER versus SNR plots as was done in Fig. 2.4. However, these results can depend strongly on the characteristics of the outer channel code and on system parameters (number of antennas, symbol alphabet, etc.), thus making comparison of different demodulators very difficult.

We thus seek a *code-independent* performance measure, which is independent of the system parameters and the actual code used. To this end, [20] proposed to use the mutual information of the equivalent BICM modulation channel. The equivalent BICM modulation [10] channel has binary input c_l, and continuous output Λ_l and is characterized by the pdf $f(\Lambda_l|c_l)$. We caution the reader that this conditional density depends on the bit position l, whereas in most cases this information is not available to the channel decoder, i.e. the channel decoder is "‘blind"’ to the bit position in the symol label. We therefore randomly pick a bit position according to a uniform distribution to obtain the input bit c and the output LLR Λ. The mutual information between c and Λ is given by [39]

$$R \triangleq I(c;\Lambda) = R_0 - \frac{1}{2}\sum_{l=1}^{BM_\mathrm{T}}\sum_{b=0}^{1}\int f(\Lambda_l|c_l=b)\log_2\frac{2f(\Lambda_l)}{f(\Lambda_l|c_l=b)}d\Lambda_l\,,$$

where $f(\Lambda_l) = \frac{1}{2}\big(f(\Lambda_l|c_l=0)+f(\Lambda_l|c_l=1)\big)$. In case of a fast fading transission scenario, R is the maximum achievable rate achievable with the BICM system (cf. [40]). In case of quasi-static fading the maximum achievable rate R changes with every realization of the channel $\mathbf{H}$. Here, the probability

$$p_\mathrm{out}(r) = \Pr\{R \leq r\}\,, \quad 0 \leq r \leq R_0$$

characterizes the rate versus outage trade-off [40].

EXIT Charts. Exitrinsic information transfer (EXIT) charts [24] are a tool to analyze iterative systems. While BER simulations only assess the performance of the compound receiver, EXIT chart allow an independent analysis of the components (e.g. demoduator, channel decoder) of the iterative system and provide insights

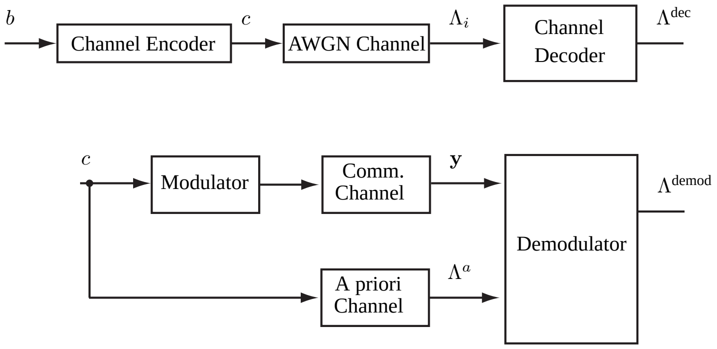


Figure 2.5: Setup for obtaining EXIT charts for the channel decoder and the demodulator.

into the performance and convergence properties of the receiver. EXIT charts can be obtained quite easily and with less simulation effort, than BER simulations of the complete system would require. The underlying idea of the analysis of iterative systems is to measure LLR distributions, track their changes over the course of iterations and thereby deduce performance and convergence properties of the system; this process is called density evolution [41]. However, LLR distributions are not easy to track due to their complicated form and therefore it is desirable to consider parameters of the LLR distribution instead. These parameters can be obtained from the LLR distribution quite easily and can be tracked with less effort. Different parameters of LLR distributions were considered and compared with respect to the ability to predict the iterative system's performance and in [24, 42] mutual information was chosen as the most robust and reliable measure.

EXIT charts measure the mutual information between bits at the input and the LLRs at the output of the component of an iterative receiver. The corresponding chart is then obtained by plotting the mutual information between the bits and the LLRs at the component's output versus the mutual information between the bits and the LLRs at the component's input. Intuitively, the component (e.g. the channel decoder) increases the mutual information about the bits by processing the LLRs at its input, and this increase in mutual information is plotted. To measure the mutual information, the LLR densities $f(\Lambda|c)$ are obtained (usually by means of Monte-Carlo simulations), and then the mutual information $I(c;\Lambda)$ is calculated according to [39]

$$I(c;\Lambda) = \frac{1}{2}\sum_{c}\int f(\Lambda|c)\log_2 \frac{2f(\Lambda|c)}{f(\Lambda|c=0)+f(\Lambda|c=1)}\, d\Lambda\,. \tag{2.11}$$

In the following we will describe how the EXIT charts of the channel decoder and the demapper are obtained in more detail. The corresponding measurement setups are shown in Fig. 2.5.

EXIT chart of the channel decoder: A random, uniform bit stream b is encoded by a channel encoder and the coded bits c are transmitted over an AWGN channel with noise variance σ_w^2. At the output of the AWGN channel, the LLRs Λ_i corresponding to the code bits are calculated according to

$$\Lambda_i = \frac{2y_i}{\sigma_w^2}\,,$$

where y_i denotes the observed output value from the channel at discrete time i. The LLRs Λ_i are fed into the channel decoder, which calculates LLRs of the information bits. The mutual information between the code bits c and the LLRs Λ_i at the input of the channel decoder is measured (cf. (2.11) with $f(\Lambda|c) = f(\Lambda_i|c)$) and will

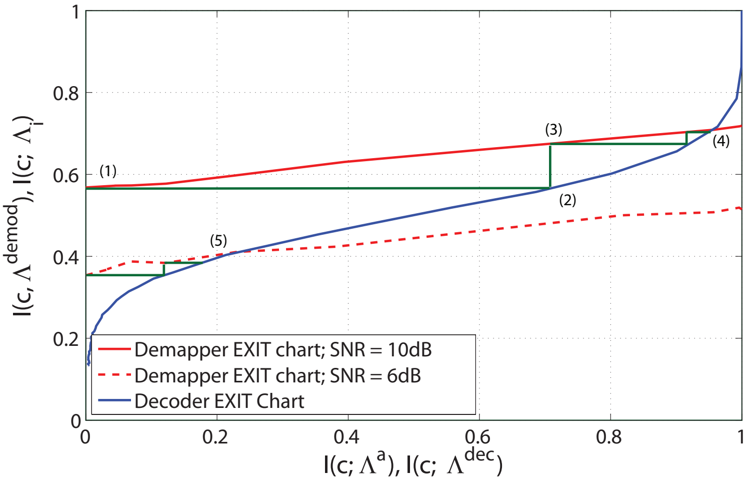


Figure 2.6: EXIT charts of the demodulator and the channel decoder of a 2 × 2 MIMO-BICM system with 16QAM Gray mapped alphabet.

be denoted as $I(c; \Lambda_i)$. Similarly, the mutual information between the information bits b and the LLRs Λ^{dec} at the output of the channel decoder is measured (cf. (2.11) with $f(\Lambda|c) = f(\Lambda^{\text{dec}}|c)$) and denoted by $I(b; \Lambda^{\text{dec}})$.

By using different values for the noise variance σ_w^2 of the AWGN channel, different values for the mutual information $I(c; \Lambda_i)$ are obtained, and the mutual information Λ^{dec} will also change. The EXIT chart of the decoder is then the plot of $I(b; \Lambda^{\text{dec}})$ versus $I(c; \Lambda_i)$. Intuitively, the EXIT chart describes how much information about the information bits is contained in Λ^{dec} for different values of information about the code bits c at the input of the channel decoder. The actual form of the EXIT chart depends on the code used and its rate.

EXIT chart of the demodulator: A random, uniform bit stream c is mapped to symbols and transmitted via the communication channel with noise variance σ_w^2, which outputs observed values $\mathbf{y}$. The same bit stream c is also transmitted over a so-called *a priori* channel (this is either an erasure channel with crossover probability ϵ or an AWGN channel with noise variance σ_a^2). At the output of the *a priori* channel, the LLRs Λ^a are calculated. From the observation of the channel output $\mathbf{y}$ and the a priori LLRs Λ^a, the demodulator calculates extrinsic LLRs Λ^{demod} according to (2.10). The mutual information between the bits c and the *a priori* LLRs Λ^a is measured (cf. (2.11) with $f(\Lambda|c) = f(\Lambda^a|c)$) and denoted by $I(c; \Lambda^a)$. Analogously, the mutual information between the bits c and the extrinsic LLRs Λ^{demod} is measured (cf. (2.11) with $f(\Lambda|c) = f(\Lambda^{\text{demod}}|c)$) and denoted by $I(c; \Lambda^{\text{demod}})$. By varying the parameter of the *a priori* channel (crossover probabiity ϵ or noise variance σ_a^2), different values for the mutual information $I(c; \Lambda^a)$ are obtained, resulting in different values for the mutual information $I(c; \Lambda^{\text{demod}})$. The EXIT chart of the demodulator is the plot of $I(c; \Lambda^{\text{demod}})$ versus $I(c; \Lambda^a)$. Intuitively, it describes how much information about the bits is contained in $I(c; \Lambda^{\text{demod}})$ for different values of information about the code bits at the input of the demodulator. The form of the EXIT chart depends on the noise variance σ_w^2 of the communication channel, the symbol mapping and demodulator.

Fig. 2.6 shows EXIT charts corresponding to the BICM-ID system considered before. The EXIT charts of the demodulator are shown for two different SNR values, SNR=4 dB and SNR=10 dB. The almost linear form with a small slope is typical for a Gray symbol mapping. The EXIT chart of the channel decoder is plotted

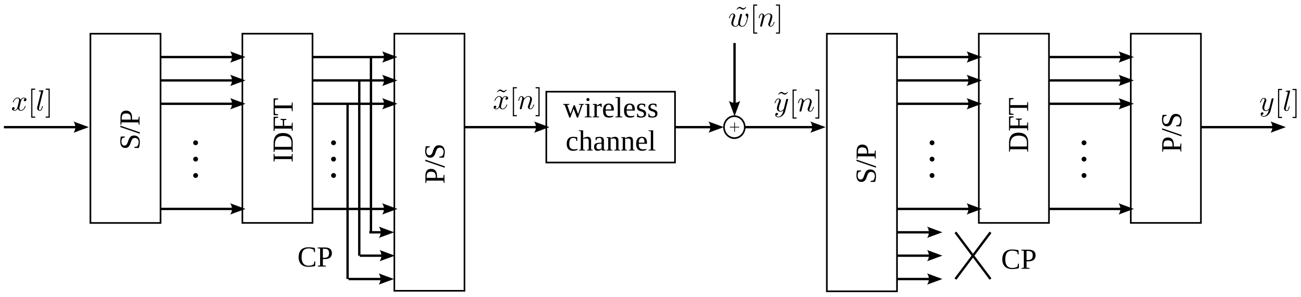


Figure 2.7: Block diagram of an OFDM system with a cyclic prefix of length $N_{\text{cp}} = 3$.

with exchanged axes; it shows $I(c;\Lambda_i)$ versus $I(c;\Lambda^{\text{dec}})$. The EXIT chart exhibits a thresholding behaviour: Below an input mutual information of $I(c;\Lambda_i) = 0.4$, the output mutual information $I(c;\Lambda^{\text{dec}})$ has only low values, for $I(c;\Lambda_i) > 0.6$ the output mutual information comes close to one. For more powerful codes (like LDPC codes), this threshold behaviour becomes even stronger. The iterative decoding process at an SNR of 10 dB is indicated by the green line, called a decoding trajectory. Initially, the demodulator calculates LLRs without any *a priori* information, therefore the decoding trajectory starts at the point labeled "(1)". The output LLRs of the demodulator are fed into the channel decoder (therefore the axis of the channel decoder EXIT chart is exchanged in Fig. 2.6), which yields an output mutual information of $I(c;\Lambda^{\text{dec}}) \approx 0.7$, corresponding to the point labelled "(2)". The output LLRs of the channel decoder are fed into the demodulator, which yields an output mutual information as inidicated by "(3)" in the EXIT chart. After several iterations, the decoding trajectory ends in point "(4)", which is the intersection of the two EXIT charts. Here, $I(c;\Lambda^{\text{dec}}) \approx 0.95$. This indicates a very good performance in terms of BER [43]. Comparing the EXIT chart of the demodulator at an SNR of 10 dB with the demodulator at an SNR of 6 dB, we see that in the latter case the decoding trajectory ends in "5", exhibiting a much lower value of $I(c;\Lambda^{\text{dec}})$, thereby indicating a much worse BER performance. Comparing these results with the actual BER in Fig. 2.4, we see that the EXIT chart analysis predicts the BER performance quite well: At SNR=6dB, the BER of the BICM-ID system is about $1/2$, whereas at SNR=10 dB, the BER is $5 \cdot 10^{-3}$.

2.1.4 Orthogonal Frequency Division Multiplexing

Historically, OFDM was introduced in the $1970'$s [44], but gained popularity in the last decade, because only current technologies make satisfactory implementation feasible. OFDM is a multi-carrier technique with specific orthogonality constraints between the subcarriers. It is robust to fading, and equalization of wireless channels is very simple. By appropriately choosing the subcarrier frequency spacing OFDM can be flexibly used in different mobile environments. This led to the adaptation of OFDM in many standards, such as WLAN (IEEE standards 802.11a and 802.11n), digital video broadcasing (DVB), and digital audio broadcasting (DAB).

A simplified model of an OFDM system is shown in Fig. 2.7. A block of data symbols $x[l]$, $l = 1,\dots,K$, is converted to a parallel vector. Being in the frequency domain, the index l is called subcarrier index. An inverse discrete Fourer transformation (IDFT) is applied to the vector, which can be efficiently implemented by means of an IFFT of length K. Next the *cyclic prefix* is added: The last N_{cp} elements of the vector are appended at its

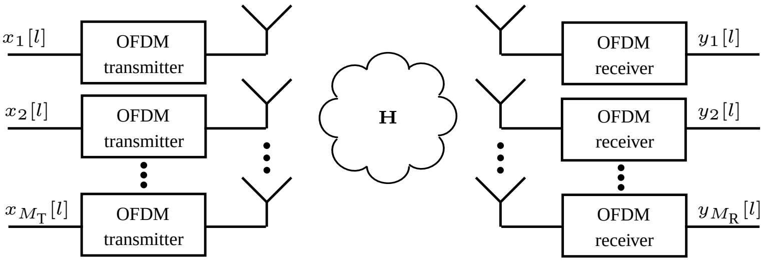


Figure 2.8: MIMO-OFDM system model.

beginning. Therefore, the resulting vector has length $K + N_{\text{cp}}$, which is converted into a serial data stream $\tilde{x}[n]$ and transmitted over the wireless channel which has a channel impulse response $\tilde{h}[n]\,, n = \tilde{h}[0], \ldots, \tilde{h}[L_c]$ of length-L_c.

The transmission introduces inter-symbol interference (ISI), and at the receiver, the signal $\tilde{y}[n]$ is observed,

$$\tilde{y}[n] = \sum_{n'=0}^{L_c} \tilde{h}[n']\tilde{x}[n-n'] + \tilde{w}[n]\,.$$

Here $\tilde{w}[n]$ denotes white Gaussian noise. The cyclic prefix (that is, the first N_{cp} samples) is discarded, and a discrete Fourier transform (implemented efficiently by means of a length-K FFT) followed by a parallel-serial conversion yields the sequence $y[l]$, $k = 1, \ldots, K$.

If the length of the cyclic prefix is larger than the channel impulse response length (that is $N_{\text{cp}} > L_c$), the cyclic prefix converts the (linear) convolution of the channel impulse response into a cyclic convolution [45]. After the demodulation at the receiver, this cyclic convolution corresponds to the multiplicative OFDM input-output relation

$$y[l] = h[l]x[l] + w[l]\,. \tag{2.12}$$

Here $h[l]$ and $w[l]$ denote the length-K DFT of the channel impulse response $\tilde{h}[n]$ and the noise sequence $\tilde{w}[n]$, respectively. From (2.12) follows that OFDM converts the frequency-selective wireless channel into a simple flat fading model.

The principle of OFDM transmission can be easily extended to a MIMO system as well, thereby making OFDM a very attractive choice for transmission. Such a MIMO-OFDM system is shown in Fig. 2.8. The symbols $x_i[l], i = 1, \ldots, M_{\text{T}}, l = 1, \ldots, K$ are transmitted over the ith antenna using OFDM modulation, i.e. the lth symbol $x_i[l]$ modulates a corresponding subcarrier. At symbol time n, the MIMO-OFDM transmitter sends the time-domain symbol $\tilde{\mathbf{x}}[n] = \big(\tilde{x}_1[n] \cdots \tilde{x}_{M_{\text{T}}}[n]\big)^T$ over the wireless channel, modeled as length-L_c sequence of $M_{\text{R}} \times M_{\text{T}}$ matrices $\tilde{\mathbf{H}}[n'], n' = 1, \ldots, L_c$.

After transmission over the frequency-selective wireless channel, the receiver observes the sequence

$$\tilde{\mathbf{y}}[n] = \sum_{n'=0}^{L_c} \tilde{\mathbf{H}}[n']\tilde{\mathbf{x}}[n-n'] + \tilde{\mathbf{w}}[n]\,,$$

where $\tilde{\mathbf{w}}[n]$ denotes white Gaussian noise. On every receive antenna an OFDM demodulator processes the received sequence, and combining the symbols from all antennas at subcarrier l into a vector $\mathbf{y}[l] =$

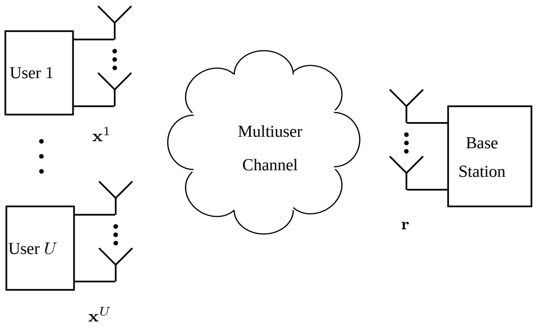


Figure 2.9: MIMO multiuser system.

$\left(y_1[l] \cdots y_{M_{\mathrm{R}}}[l]\right)^T$, the MIMO-OFDM input-output relation becomes

$$\mathbf{y}[l] = \mathbf{H}[l]\mathbf{x}[l] + \mathbf{w}[l]\,.$$

Analogously to the SISO case, $\mathbf{H}[l]$ and $\mathbf{w}[l]$ denote the length-K DFT of the channel impulse response $\tilde{\mathbf{H}}[n]$ and the noise sequence $\tilde{\mathbf{w}}[n]$, respectively.

2.1.5 Multiuser Systems

A multiuser system consists of U user terminals communicating with a common base station. Fig. 2.9 shows such a system. Every user terminal and the base station can be equipped with multiple antennas, in this case we speak of a multiuser MIMO system. Depending on the transmission direction, we speak of a *downlink* scenario, when we consider the transmission of data from the base station to the user terminals, and of an *uplink* scenario, when we consider the detection of the users data streams at the base station. In this thesis we will focus on uplink scenarios, in particular on the design of receiver algorithms for this case.

User Detection Strategies. The most intuitive way of dealing with multiuser uplink scenarios would be to separate the users as much as possible. By assigning users disjoint time slots (time division multiple access, TDMA), disjoint frequency band (frequency division multiple access, FDMA) or special spreading sequences (code-division multiple access, CDMA), interference between the users can be minimized or even totally avoided. This reduces the problem of multiuser detection to single-user detection. for which numerous schemes and algorithms exist. However, the *orthogonal* user separation strategies are suboptimal in terms of throughput, which will be shown by information theoretic arguments in the next paragraphs. In this thesis, we will therefore consider non-orthogonal schemes, with interference between users.

System Model. For simplicity we assume that every user terminal is equipped with the same number of transmit antennas, M_{T}. The uth terminal transmits the symbol vector $\mathbf{x}^u$ and the MIMO channel from this terminal to the base station is denoted by $\mathbf{H}^u$. At the receiver the symbol $\mathbf{y}$ is observed,

$$\mathbf{y} = \sum_{u=1}^{U} \mathbf{H}^u \mathbf{x}^u + \mathbf{w}\,, \tag{2.13}$$

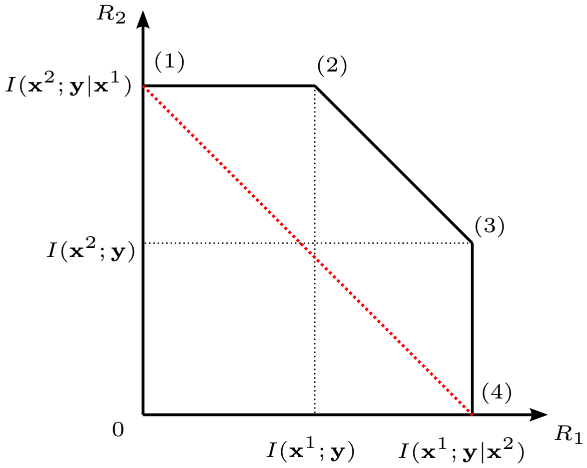


Figure 2.10: Achievable rate region of the multiple-access channel with $U = 2$ users.

where U denotes the total number of users and $\mathbf{w}$ denotes zero-mean, white complex Gaussian noise, $\mathbf{w} \sim \mathcal{CN}(\mathbf{0}, \sigma_w^2\mathbf{I})$.

Information-Theoretic Performance Limits. The multiple-access (MAC) channel (2.13) is a well studied model in information theory and can be characterized by the *rate region* [39]. For the special case of $U = 2$ users, the rate region becomes

$$\begin{aligned} R_1 &< I(\mathbf{x}^1;\mathbf{y}|\mathbf{x}^2)\,, \\ R_2 &< I(\mathbf{x}^2;\mathbf{y}|\mathbf{x}^1)\,, \\ R_1 + R_2 &< I(\mathbf{x}^1,\mathbf{x}^2;\mathbf{y})\,, \end{aligned} \tag{2.14}$$

and is depicted in Fig. 2.10 (R_i denotes the rates achievable by user i). The corner points of the rate region can be interpreted as follows: In point (1) only terminal 2 transmits and terminal 1 is idle (we have $R_1 = 0$), which is basically a single-user scenario. The maximum achievable rate of terminal 2 is given by $I(\mathbf{x}^2;\mathbf{y}|\mathbf{x}^1)$. Point (2) corresponds to the maximum rate at which terminal 1 can send as long as terminal 2 sends at its corresponding maximum rate $I(\mathbf{x}^2;\mathbf{y}|\mathbf{x}^1)$. For the channel from terminal 1 to the base station, the signal $\mathbf{x}^2$ is treated as noise. Using the results for single-user channels, terminal 1 can send at a rate of $I(\mathbf{x}^1;\mathbf{y})$, therefore the receiver can recover the signal $\mathbf{x}^1$ and perform *interference cancellation* to detect $\mathbf{x}^2$ from the received signal $\mathbf{y}$. The rate R_2 achievable in this case is then the rate $I(\mathbf{x}^2;\mathbf{y}|\mathbf{x}^1 = \mathbf{x})$, averaged over all symbols $\mathbf{x}^1$, namely

$$\sum_{\mathbf{x}} I(\mathbf{x}^2;\mathbf{y}|\mathbf{x}^1 = \mathbf{x})\, p(\mathbf{x}^1 = \mathbf{x}) = I(\mathbf{x}^2;\mathbf{y}|\mathbf{x}^1)\,.$$

The same arguments hold for the points (3) and (4), only with the role of terminal 1 and terminal 2 exchanged. The rate-pairs on the connection between points (1) and (2), (2) and (3), and (3) and (4) can be reached by time sharing. The red dotted line in Fig. 2.10 shows the achievable rate of orthogonal transmission schemes like TDMA and FDMA. Rate points (1) and (4) are trivially achievable, and by means of time sharing all rate-pairs on the red dotted line.

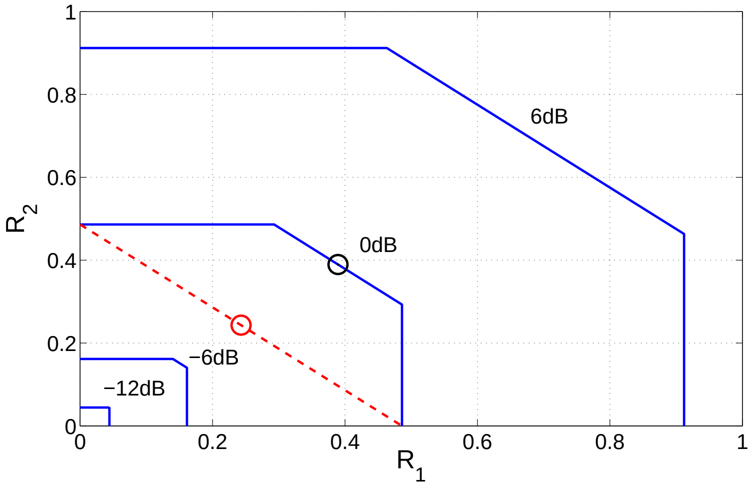


Figure 2.11: Achievable rate region of the Gaussian MAC with BPSK symbols.

We next show simulation results for a SISO system with $U = 2$ users using BPSK modulation over an AWGN channel. The system model (2.13) simplifies to

$$y = x^1 + x^2 + w\,,$$

with the noise distributed according to $w \sim \mathcal{N}(0, \sigma_w^2)$, and $x^u \in \{-1, +1\}$. Fig. 2.11 shows the capacity regions for an SNR between -12 dB and 6 dB (the SNR is increased in 6 dB steps). It can be seen that for $\sigma_w^2 \to 0$, the single-user rates (that is $I(\mathbf{x}^u; \mathbf{y}|\mathbf{x}^{3-u})$, $u = 1, 2$) converge to 1 bpcu, and the sum rate rate $I(\mathbf{x}^1, \mathbf{x}^2; \mathbf{y})$ converges to $3/2$ bpcu. In case of very small SNR, the rates $I(\mathbf{x}^u; \mathbf{y})$ converge to $I(\mathbf{x}^u; \mathbf{y}|\mathbf{x}^{3-u})$ $(u = 1, 2)$ and the points (2) and (3) in the rate region of Fig. 2.10 become one point.

As before the dotted red line shows the maximum achievable rate at an SNR of 0 dB of an orthogonal transmission schemes like TDMA and FDMA and the red circle represents the operation point where both terminals are assigned the same rate. In contrast, the black circle represents the operation point of an (non-orthogonal) information-theoretic optimal scheme at the same SNR, with the same rate assigned to both terminals. It can be seen, that there is a gap of about 0.15 bpcu in achievable rate per user. Stated in terms of SNR, an orthogonal scheme would require about 4 dB more SNR to achieve the same rate as an information-theoretic optimal transmission scheme. This demonstrates the poor performance of orthogonal transmission schemes quite clearly. Because the rate constraint $R_1 + R_2 < I(\mathbf{x}^1, \mathbf{x}^2; \mathbf{y})$ becomes inactive in the limit of low SNR (that is, the rate regions become rectangles), the relative performance loss of orthogonal transmission schemes increases with lower SNR.

2.2 Graphical Models

Graphical models provide a unifying framework for capturing dependencies among random variables, and building statistical models. In many applied fields, including signal processing for communications, image and speech processing, statistical models have been formulated by means of graphs, which provide a simple,

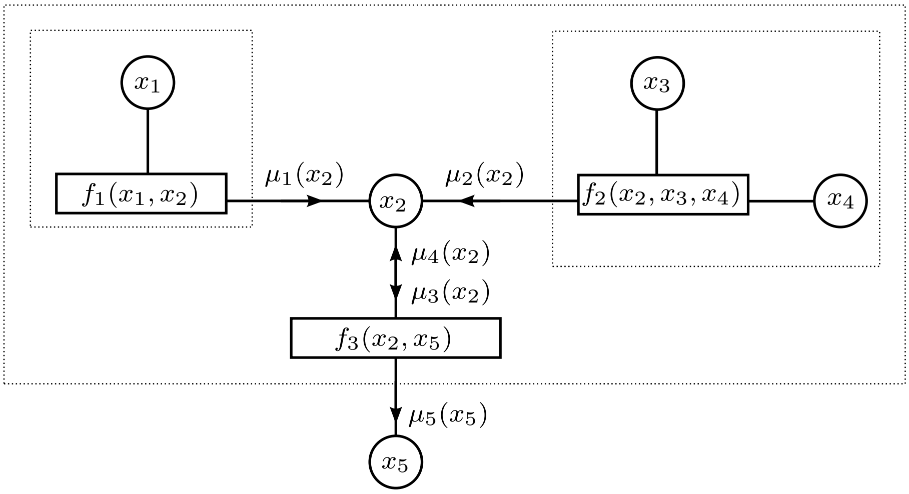


Figure 2.12: Factor graph for the function $f(\mathbf{x})$ in (2.15).

yet intuitive representation of the underlying constraints. Algorithms for computing statistical quantities such as likelihoods and marginals have often been formulated as recursions operating on these graphs. Using the concept of *message passing*, these recursions can be obtained by a simple "turn-the-crank procedure" [4, 5]. In this way, specific algorithms, for example the Kalman filter, the BCJR algorithm, and iterative decoding of low-density parity-check (LDPC) codes can be derived, understood, and unified with the concept of graphical models.

2.3 Factor Graphs and the Sum-Product Algorithm

Factor graphs [4, 5] are one particular family of graphical models and have found widespread use in signal processing and communications. Factor graphs represent functions of several variables which can be split into several factors. For example, consider the function

$$f(x_1, x_2, x_3, x_4, x_5) = f(\mathbf{x}) = f_1(x_1, x_2) f_2(x_2, x_3, x_4) f_3(x_2, x_5)\,, \tag{2.15}$$

with $\mathbf{x} = (x_1 \cdots x_5)^T$. The factors $f_i(\cdot)$, $i = 1, \ldots, 3$ are termed local functions, whereas their product is refered to as global function.

In Fig. 2.12 the factor graph corresponding to (2.15) is shown, which consists of *variable nodes* (depicted as circles) for the variables $x_1, \ldots, x_5$, and *function nodes* (depicted as rectangles) for the local functions $f_1(\cdot), f_2(\cdot), f_3(\cdot)$. If a local function depends on a variable node, the two corresponding graph vertices are connected with each other by an edge. For example, the function $f_1(\cdot)$ depends on the variables x_1 and x_2, and therefore these two variable nodes are connected with the function node $f_1(\cdot)$. A particular assignment of values to the variables nodes is termed a *configuration* of a factor graph. The *configuration space* is the set of all possible configurations, and therefore the domain of the global function $f(\cdot)$. If we assume that all variables of our example factor graph are binary, the configuration space is $\{0, 1\}^5$.

As stated before, a main area of application of factor graphs are statistical models. In this case the variables represent the random variables of the model, and the global function represents the probability density function, which can be factorized according to the statistical model.

Cut-Set Independence Theorem. This theorem [5] connects properties of the graph to the probability density function the graph represents.

Assume that some edges corresponding to variable nodes $Y_1, \ldots, Y_n$ form a cut-set of the graph (that is, removing these edges cuts the graph into two unconnected components with variables from the sets $\mathcal{X}_1$ and $\mathcal{X}_2$, respectively). It can be shown [4] that conditioned on $Y_1, \ldots, Y_n$, variables from the sets $\mathcal{X}1$ and $\mathcal{X}2$ become indepedent, i.e.

$$f(\mathcal{X}_1, \mathcal{X}_2 | Y_1, \ldots, Y_n) = f(\mathcal{X}_1 | Y_1, \ldots, Y_n) f(\mathcal{X}_2 | Y_1, \ldots, Y_n)\,.$$

In our example of Fig. 2.12, the edges connecting x_2 with $f_1(\cdot)$ and $f_2(\cdot)$ form a cut-set of the graph. Therefore, given x_2, the random variables x_1 and x_3 are independent, as are the random variable x_1 and the set $\{x_3, x_4, x_5\}$.

Marginalization. Statistical models are used for estimation and/or detection of some underlying parameters or random variables given observations. The process of estimation and detection often involves the calculation of marginals of the joint pdf $f(\mathbf{x})$, that is the functions

$$g_k(x_k) = \sum_{\sim x_k} f(\mathbf{x}), \quad k = 1, \ldots, \mathcal{X}. \tag{2.16}$$

In the whole thesis, the notation $\sum_{\sim x_k}$ denotes summation over all variables of $\mathbf{x}$ except x_k, and $\mathcal{X}$ denotes the length of the vector $\mathbf{x}$.

In our example we will consider calculation of the marginal function $g_5(x_5)$, defined as

$$g_5(x_5) = \sum_{\sim x_5} f(\mathbf{x})\,. \tag{2.17}$$

Naively, this function can be evaluated by calculating the values of $f(\mathbf{x})$ for all $2^5 = 32$ configurations, and then summing up these terms to obtain $g_5(x_5)$. A different strategy, which requires much less computations, is the application of the distributive law to (2.17), which yields

$$g_5(x_5) = \sum_{x_2} f_3(x_2, x_5) \sum_{x_1} f_1(x_1, x_2) \sum_{x_3, x_4} f_2(x_2, x_3, x_4)\,. \tag{2.18}$$

Using the factorization in the calculation of marginal functions yields a considerable complexity reduction: The first and second sum consist of two terms, while the third term consists of three terms. This is a much smaller number than directly calculating all 32 terms in (2.17).

Comparison of the factor graph in Fig. 2.12 and the expression in (2.18) reveals that calculating the sums in (2.18) is equivalent to “closing boxes” in the factor graph: The small dashed boxes correspond to the sums in brackets, and the “closing of the boxes” can be interpreted as summation over all variables inside the box. For example, the box enclosing the variable node x_1 and the local function $f_1(x_1, x_2)$ corresponds to the second sum in (2.18), $\sum_{x_1} f_1(x_1, x_2)$. The resulting expression of the “box-closing” operation is function depending solely on variable x_2 (the variable the box is connected with). It is denoted by $\mu_1(x_2)$ and is termed a *message*.

In a similar manner, the expression $\sum_{x_3,x_4} f_2(x_2,x_3,x_4)$ is identified with the small dashed box in the right upper part of Fig. 2.12, and the output of the box is given by the message $\mu_2(x_2)$. Replacing the sums with messages in (2.18), we can rewrite the marginal function $g_5(x_5)$ according to

$$g_5(x_5) = \sum_{x_2} f_3(x_2,x_5)\mu_1(x_2)\mu_2(x_2)\,. \tag{2.19}$$

The product of the two messages $\mu_1(x_2)$ and $\mu_2(x_2)$ can be interpreted as a new message,

$$\mu_3(x_2) = \mu_1(x_2)\mu_2(x_2)\,. \tag{2.20}$$

This message is emitted by the variable node x_2, and we obtain

$$g_5(x_5) = \sum_{x_2} f_3(x_2,x_5)\mu_3(x_2)\,. \tag{2.21}$$

Similarly, we can interpret $g_5(x_5)$ as a message obtained by the “box-closing” operation of the large dotted box in Fig. 2.12, and we have $\mu_5(x_5) = g_5(x_5)$.

The example demonstrates the important concept of *message updates*: Starting from the leave nodes of the factor graph, messages are passed along the edges of the factor graph. At function nodes, the local function associated with the function node and the incoming messages are involved in the message update process (cf. (2.21)). The updated messages are propagated to other nodes of the factor graph. At variable nodes, the outgoing messages depend on the incoming ones (cf. (2.20)). This concept of message updates makes the application of factor graphs popular: After formulation of the statistical model (and obtaining a graphical and intuitive representation of the model), the process of calculating the marginal functions becomes the task of updating and propagating messages.

When dealing with statistical models, often all marginal functions are of interest. Instead of running an instance of the sum-product algorithm for every marginal function, messages can be reused and thereby save computations. As an example, consider the calculation of another marginal function,

$$g_2(x_2) = \sum_{\sim x_2} f(\mathbf{x})\,.$$

Using a factorization similar to (2.18), we obtain

$$g_2(x_2) = \sum_{x_1} f_1(x_1,x_2) \sum_{x_3,x_4} f_2(x_2,x_3,x_4) \sum_{x_5} f_3(x_2,x_5)\,.$$

Note that here some of the summation variables are different as in (2.18). Using the previously calculated messages, we have

$$g_2(x_2) = \mu_1(x_2)\mu_2(x_2)\mu_4(x_2)\,,$$

where we introduced the new message $\mu_4(x_2) = \sum_{x_5} f_3(x_2,x_5)$.

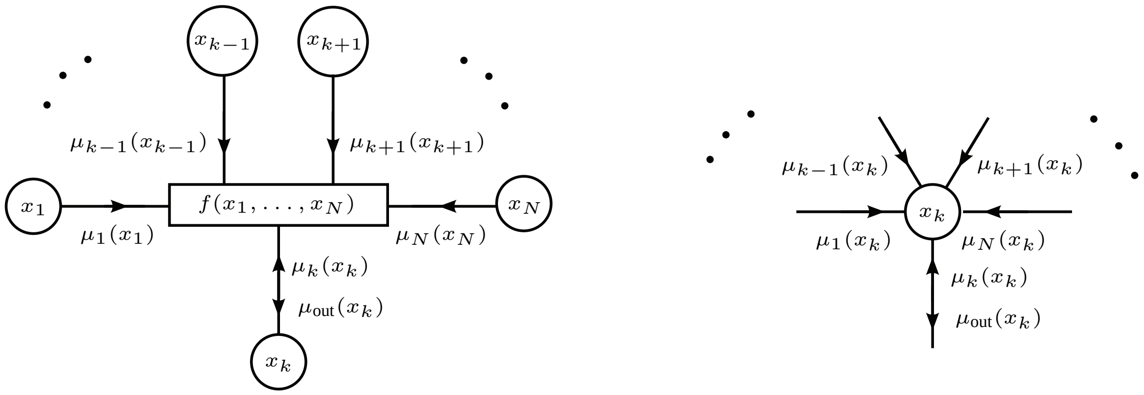


Figure 2.13: Message updates of the sum-product algorithm.

Message Updates. The message updates of the above example can be generalized to general graphical models and lead to two basic generic message update rules for (i) messages which leave a function node (and are directed towards a variable node), and (ii), messages which leave a variable node (and are directed towards a function node). It is important to note, that these updates yield exact results only for *cycle-free* factor graphs.

The first case is depicted in the left part Fig. 2.13: Messages $\mu_k(x_k),\ ,k = 1,\ldots,N$ arrive at the function node $f(x_1,\ldots,x_N)$, and the outgoing message $\mu_{\text{out}}(x_k)$ leaves the function node $f(\mathbf{x})$. It is given by

$$\mu_{\text{out}}(x_k) = \sum_{\sim x_k} f(x_1,\ldots,x_N) \prod_{n\neq k} \mu_n(x_n)\,. \tag{2.22}$$

The local function $f(\cdot)$ is multiplied with all incoming messages $\mu_n(x_n)$ except $\mu_k(x_k)$, and the resulting product is summed over all variables, except x_k. Obviously, the message $\mu_{\text{out}}(x_k)$ is a function only of the variable x_k and does not depend on the message $\mu_k(x_k)$. In the foregoing example, calculation of the message $\mu_5(x_5)$ in (2.21) was obtained this way. The second case deals with the arrival of messages from different factor nodes at a variable node and is depicted in the right part of Fig. 2.13. The incoming messages are denoted by $\mu_l(x_k),\ ,l = 1,\ldots,N$; because they are all connected with the variable node x_k they can only depend on this variable. The message $\mu_{\text{out}}(x_k)$ leaving the variable node x_k is given by

$$\mu_{\text{out}}(x_k) = \prod_{n\neq k} \mu_n(x_k)\,. \tag{2.23}$$

In this case the outgoing message is the product of all incoming messages, except the message $\mu_k(x_k)$; i.e. it is independent of the message $\mu_k(x_k)$. In the foregoing example, the message $\mu_3(x_3)$ in (2.20) was obtained in this manner.

Sum-Product Algorithm. The calculation of all marginal functions (in our example of $g_1(x_1),\ldots,g_7(x_7)$ can be performed concurrently, and the resulting algorithm is termed *sum-product algorithm*. Assuming that the factor graph is a tree, the algorithm starts at the leaves of the factor graphs. Leaves, which are variable nodes, send an initial message of value ’1’ to their respective neighbours, whereas leaves, which are function nodes, send the corresponding function value as initial value to their respective neighbours. When a neighbouring function node has received all required messages to calculate an outgoing message according to the update rule

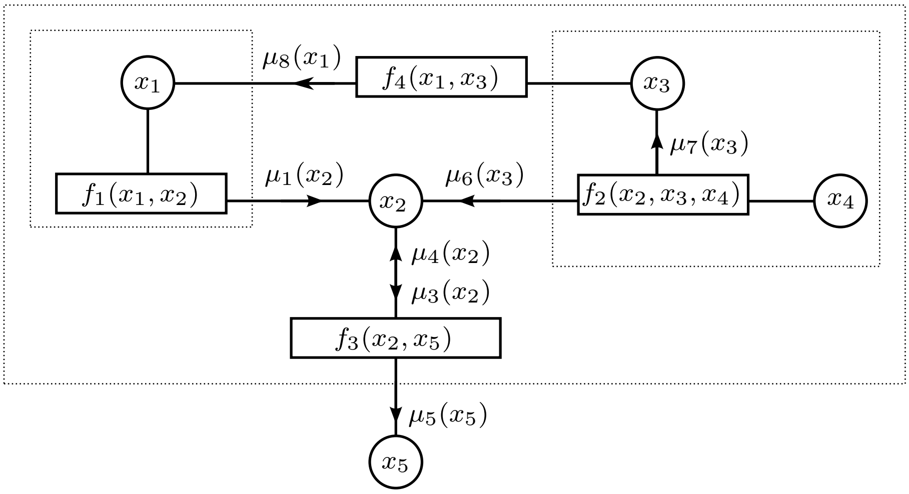


Figure 2.14: Factor graph example with cycles.

(2.22), it derives the updated message and propagates it along the corresponding edge of the factor graph. In an analgous manner, neighbouring variable nodes derive outgoing updated messages according to (2.23) and propagate these along the corresponding edges of the graph. In case of a cycle-free factor graph, this process is finished after a finite number of message updates. The marginal functions $g_k(x_k)$ at the corresponding variable nodes x_k can then be obtained as the product of all incoming messages. In case of Fig. 2.13, we would have

$$g_k(x_k) = \prod_n \mu_n(x_k)\,.$$

Factor Graphs with Cycles. Depending on the graphical model, the factor graph can have cycles. Yet, we can still apply the message updates (2.22) and (2.23) of the sum-product algorithm. Consider an example obtained by slightly modifying (2.15),

$$f(\mathbf{x}) = f_1(x_1,x_2)f_2(x_2,x_3,x_4)f_3(x_2,x_5)f_4(x_1,x_3)\,, \tag{2.24}$$

the corresponding factor graph is shown in Fig. 2.14. Compared to (2.15), an additional function $f_4(\cdot)$ has been added; also note that by the cut-set independence theorem, x_1 and x_3 are no longer independent given x_2.

Let us consider application of the sum-product algorithm to this modified model: Assume (somewhat arbitrarily) that the variable node x_1 emits the initial message '1', and the function node $f_1(\cdot)$ calculates the message $\mu_1(x_2)$. Together with the message $\mu_4(x_2)$, a new message is formed, which is received by the function node $f_2(\cdot)$. The message is updated and the new message $\mu_7(x_3)$ is sent to variable node x_3. This message is again updated at the function node $f_4(\cdot)$, yielding the message $\mu_8(x_1)$. This message would be updated at the function node $f_1(\cdot)$, giving an updated message $\mu_1(x_2)$. It can be seen that the sum-product algorithm does not terminate in finite time anymore, and that messages are sent in cirlces. Depending on the messages used, it is possible that some kind of positive feedback lets the messages grow without bound. Even in case of convergence of the messages, the final values do not represent the true marginals $g_k(x_k)$ anymore. Furthermore, initialization of the sum-product algorithm can influence the convergence properties of the sum-product algorithm.

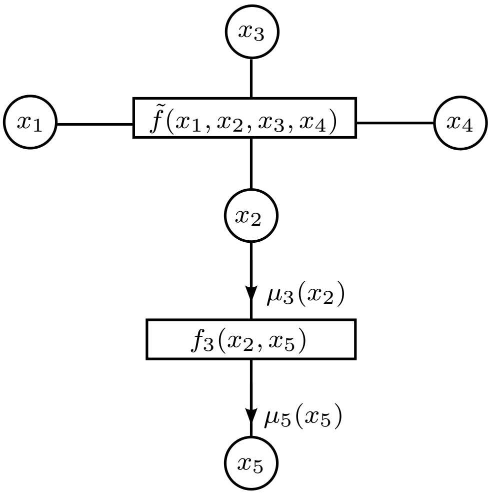


Figure 2.15: Factor graph example with cycles removed.

In practical systems, the cycles in the factor graph can be much longer, and in many cases the sum-product algorithm has been observed to give good results. For example, the factor graphs of LDPC codes and turbo codes have many cycles, giving rise to iterative algorithms for decoding. Although no theoretical results exist on the validity of the application of the sum-product algorithm to these factor graphs, the numerical performance of the algorithm in terms of bit error rates (BER) has been astonishing.

As a way to avoid cycles, we consider a different representation of (2.24),

$$f(\mathbf{x}) = \tilde{f}(x_1, x_2, x_3, x_4) f_3(x_2, x_5)\,,$$

with $\tilde{f}(x_1, x_2, x_3, x_4) = f_1(x_1, x_2) f_2(x_2, x_3, x_4) f_4(x_1, x_3)$. The corresponding factor graph is shown in Fig. 2.15. It is seen to be cycle-free and thereby all marginals can be calculated exactely by means of the sum-product algorithm. The drawback of this method is the increased computational complexity: The global function $f(\mathbf{x})$ is factorized into fewer local functions $f_i(\cdot)$, each of which depends on more variables x_i and therefore the sums for calculating messages consist of more terms. For example, the message $\mu_3(x_2)$ is given by

$$\mu_3(x_2) = \sum_{x_1, x_3, x_4} \tilde{f}(x_1, x_2, x_3, x_4)$$

and consists of 8 terms. Furthermore, an alternative factorization is often difficult to find for factor graphs in practical systems, therefore this method is not widely used in signal processing and communication applications.

Extension to other Semirings. The key step to an efficient calculation of marginals (2.16) is the application of the distributive law. The sum-product algorithm can be extended, if we consider semirings $\mathcal{D}$, on which two operations $+$ and $\cdot$ are defined and furthermore, the distributive law holds,

$$x \cdot (y + z) = x \cdot y + x \cdot z\,, \quad \forall x, y, z \in \mathcal{D}\,.$$

If the domain of the global function (and therefore also the domains of the local functions) is $\mathcal{D}$, then we can repeat all the steps in deriving the message update rules for the sum-product algorithm and obtain a sum-product algorithm suitable for the semiring $\mathcal{D}$ [4]. An important example for such a semiring is the "max-product" semiring, where real addition is replaced by the "max" operation. For nonnegative real-valued quantities $x, y, z > 0$, we have

$$x \max(y, z) = \max(xy, xz) \,.$$

By using this "distributive" law, all steps of this Subsection can be adapted and the resulting "max-product" algorithm could be used to efficiently find solutions for

$$g(x_k) = \arg\max_{\sim \mathbf{x}} f(\mathbf{x}) \,.$$

2.4 Codes on Graphs

We will next describe how decoding algorithms for convolutional codes and LDPC codes can be (re)derived by means of factor graphs and the application of the sum-product algorithm. The structure of the code can be visualized by factor graphs and it can be exploited to obtain efficient decoding algorithms.

2.4.1 Convolutional Codes

Basics. A simple block diagram of a convolutional encoder with rate $R = 1/2$ and memory length L_M is shown in Fig. 2.16. It consists of a shift-register with L_M stages, several modulo-2 adders and a multiplexer. The shift-register provides delayed versions $b_k, \dots, b_{k-L_M}$ of the information bits, some of which are modulo-2 added, yielding (in this example) two output sequences c_k^1 and c_k^2. The multiplexer serializes the sequences c_k^1 and c_k^2, yielding the bit sequence c_k. A length-N sequence of information bits is encoded, thus yielding a length N/R sequence of coded bits c_k. The past values of the information bits stored in the shift-register represent the state $S_k \in [0, 2^{L_M} - 1$ of the convolutional encoder at time k, which is defined as

$$S_k = \sum_{n=1}^{L_M} b_{k-n} 2^{n-1} \,.$$

Note that the number of states grows exponentially with the memory length, $|S_k| = 2^{L_M}$. At time k the information bit b_k induces a state transition from state S_k to state S_{k+1} and the generation of $1/R$ code bits, $c_k^1, \cdots, c_k^{1/R}$. We collect these code bits in a vector $\mathbf{c}_k = (c_k^1, \cdots, c_k^{1/R})$.

A standard form [46] to denote a convolutional code is by $1/R$ code polynomials, which describe which delayed information bits are modulo-2 added. Alternatively, the code can be described by two functions, a state transition function $S_k = T(S_{k-1}, b_k)$ and an output function $\mathbf{c}_k = O(S_k, b_k)$. The state transition function gives the next state of the encoder, when it is currently in state S_k with the information bit b_k at the input, while the output function gives the output sequence $\mathbf{c}_k$ the encoder emits in state S_k with information bit b_k at the input.

Decoding. The sequence of coded bits $\mathbf{c}_k$ is sent over a channel and the values $\mathbf{y}_k^i = (y_k^1 \dots y_k^{1/R})$ are observed. We assume the channel to be i.i.d. and described by the conditional pdf $f(y_k^i | c_k^i)$. Decoding of the

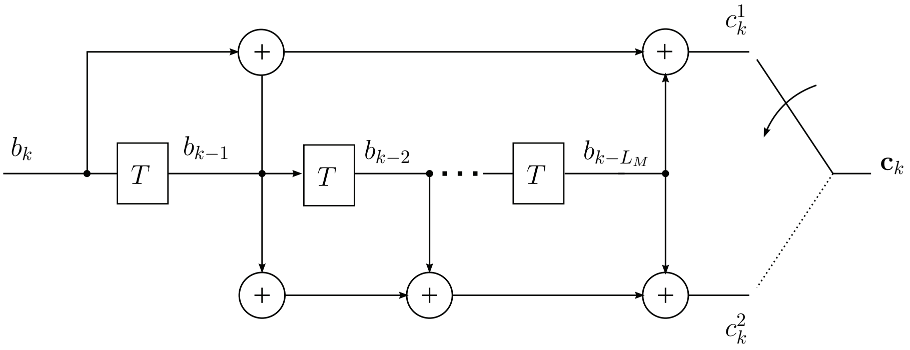


Figure 2.16: Convolutional encoder for rate-1/2 code; plus signs represent modulo-2 additions.

convolutional code can be based on the bitwise MAP criterion, that is bit-wise detection of the information bits $\hat{b}_k$ based on the received sequence $\mathbf{y}_k,\ k = 1, \ldots, N$, according to

$$\hat{b}_k = \arg\max_{b_k} p(b_k|\mathbf{y}_1, \cdots, \mathbf{y}_N)\,, \tag{2.25}$$

where $p(b_k|\mathbf{y}_1, \cdots, \mathbf{y}_N)$ denotes the posterior pmf of the information bit b_k given the observations $\mathbf{y}_1, \cdots, \mathbf{y}_N$. We can express this posterior density as a marginal [4]

$$P(b_k = b|\mathbf{y}_1, \cdots, \mathbf{y}_N) = \sum_{\sim b_k} p(b_1, \ldots, b_N, \mathbf{c}_1, \ldots, \mathbf{c}_N, S_0, \ldots, S_N|\mathbf{y}_1, \ldots, \mathbf{y}_N)\,. \tag{2.26}$$

This marginalization can be efficiently calculated by applying the sum-product algorithm to the factor graph of the convolutional code and results in the well-known BCJR algorithm [8].

Factor Graph. We will consider the factor graph corresponding to the density that needs to be marginalized according to (2.26). Using the Markov property of the convolutional code, we can rewrite $p(b_1, \ldots, b_N, \mathbf{c}_1, \ldots, \mathbf{c}_N, S_0, \ldots, S_N|\mathbf{y}_1, \ldots, \mathbf{y}_N)$ according to

$$p(b_1, \ldots, b_N, \mathbf{c}_1, \ldots, \mathbf{c}_N, S_0, \ldots, S_N|\mathbf{y}_1, \ldots, \mathbf{y}_N) = p(S_0) \prod_{k=1}^{N} p(S_k, \mathbf{c}_k|S_{k-1}, b_k) p(\mathbf{c}_k|\mathbf{y}_k)\,, \tag{2.27}$$

where $p(S_0)$ is the prior density of the decoder state S_0. The conditional density $p(S_k, \mathbf{c}_k|S_{k-1}, b_k)$ can be obtained from the state transition function $S_k = T(S_{k-1}, b_k)$ and the output function $O(S_k, b_k)$ according to

$$p(S_k, \mathbf{c}_k|S_{k-1}, b_k) = \begin{cases} 1 & \text{if } S_k = T(S_{k-1}, b_k) \text{ and } \mathbf{c}_k = O(S_k, b_k) \\ 0 & \text{otherwise}\,. \end{cases}$$

The density $p(\mathbf{c}_k|\mathbf{y}_k)$ of the channel can be obtained by application of Bayes theorem and using the independence of the channel,

$$p(\mathbf{c}_k|\mathbf{y}_k) \propto p(\mathbf{y}_k|\mathbf{c}_k) = \prod_{n=1}^{1/R} p(y_k^n|c_k^n)\,.$$

The factor graph corresponding to the factorization (2.27) is shown in Fig. 2.17. The state variables S_k, the information bits b_k and the code bits $\mathbf{c}_k$ correspond to variable nodes, whereas the conditional densities $f(\mathbf{c}_k|\mathbf{y}_k)$

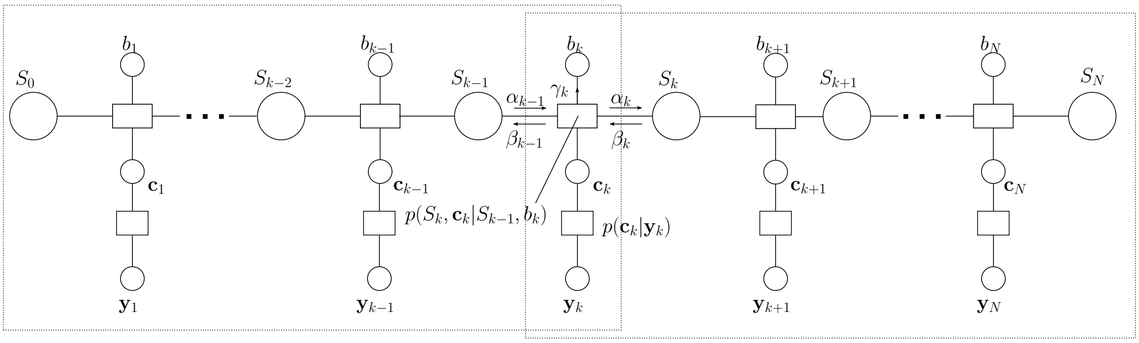


Figure 2.17: Factor graph of a convolutional code corresponding to the factorization in (2.27).

and $p(S_k, \mathbf{c}_k|S_{k-1}, b_k)$ correspond to function nodes. The Markov structure of the density in (2.27) can be seen intuitively. Also note that the factor graph has no cycles, i.e. the sum-product algorithm will calculate exactly the marginals of $p(b_1, \dots, b_N, \mathbf{c}_1, \dots, \mathbf{c}_N, S_0, \dots, S_N|\mathbf{y}_1, \dots, \mathbf{y}_N)$.

Sum-Product Algorithm. We define two message types: the messages $\alpha_k(S_k)$ are passed from the left to the right, while the messages $\beta_k(S_k)$ are passed from the right to the left of the factor graph. Initialization of the sum-product algorithm starts at the leaves, in our case these are the variable nodes S_0 and S_N. The variable node S_0 sends out an initial message $\alpha_0(S_0)$, furthermore we assume that the convolutional encoding starts in state $S_0 = 0$, and therefore the inital message becomes $\alpha_0(S_0) = \delta[S_0]$. The variable node S_N sends out an initial message $\beta_N(S_N)$. By means of zero-padding the information bit-sequence one can ensure, that the convolutional encoder stops in state $S_N = 0$; we therefore have $\beta_N(S_N) = \delta[S_N]$.

We now consider an arbitrary function node $p(S_k, \mathbf{c}_k|S_{k-1}, b_k)$, $1 \leq k \leq N-1$ and derive the message updates at this node. The message $\alpha_{k-1}(S_{k-1})$ is entering the node from the left. The variable node $\mathbf{y}_k$ emits the initial message '1', which causes the function node $f(\mathbf{c}_k|\mathbf{y}_k)$ to send out the message $f(\mathbf{c}_k|\mathbf{y}_k)$. Since this is the only message entering the variable node $\mathbf{c}_k$, it is simply forwarded to the function node $p(S_k, \mathbf{c}_k|S_{k-1}, b_k)$. Using the update rules (2.22) of the sum-product algorithm, we therefore obtain for the outgoing message $\alpha_k(S_k)$

$$\alpha_k(S_k) \propto \sum_{b_k, S_{k-1}, \mathbf{c}_k} p(S_k, \mathbf{c}_k|S_{k-1}, b_k) p(\mathbf{c}_k|\mathbf{y}_k) \alpha_{k-1}(S_{k-1}), \quad k = 1, \dots, N\,. \tag{2.28}$$

Using the interpretation of the "closing boxes" of [5] (here, the left box in Fig. 2.17 is closed), we can interpret the message $\alpha_k(S_k)$ as $\alpha_k(S_k) = p(S_k|\mathbf{y}_1, \dots, \mathbf{y}_k)$.

To show the validity of this interpretation, we derive the probability $p(S_k|\mathbf{y}_1, \dots, \mathbf{y}_k)$ from the system model (2.27). We have

$$p(S_k|\mathbf{y}_1, \dots, \mathbf{y}_k) = \sum_{\{b\}_1^k, \{S\}_1^{k-1}, \{\mathbf{c}\}_1^k} p(b_1, \dots, b_k, \mathbf{c}_1, \dots, \mathbf{c}_k, S_0, \dots, S_k|\mathbf{y}_1, \dots, \mathbf{y}_k)\,, \tag{2.29}$$

where the summation includes $b_1, \dots, b_k$, $S_0, \dots, S_k$, and $\mathbf{c}_1, \dots, \mathbf{c}_k$. Inserting (2.27), we obtain

$$p(S_k|\mathbf{y}_1, \dots, \mathbf{y}_k) = \sum_{\{b\}_1^k, \{S\}_1^{k-1}, \{\mathbf{c}\}_1^k} p(S_0) \prod_{l=1}^{N} p(S_l, \mathbf{c}_l|S_{l-1}, b_l)\,,$$

and by the distributive law we have

$$p(S_k|\mathbf{y}_1,\ldots,\mathbf{y}_k) = \sum_{b_k,S_{k-1},\mathbf{c}_k} p(S_k,\mathbf{c}_k|S_{k-1},b_k)p(\mathbf{c}_k|\mathbf{y}_k) \sum_{\{b\}_1^{k-1},\{S\}_1^{k-2},\{\mathbf{c}\}_1^{k-1}} \prod_{l=1}^{l-1} p(S_l,\mathbf{c}_l|S_{l-1},b_l)\,.$$

We recognize the second sum as $\alpha_{k-1}(S_{k-1})$, and finally obtain

$$p(S_k|\mathbf{y}_1,\ldots,\mathbf{y}_k) = \sum_{b_k,S_{k-1},\mathbf{c}_k} p(S_k,\mathbf{c}_k|S_{k-1},b_k)p(\mathbf{c}_k|\mathbf{y}_k)\alpha_{k-1}(S_{k-1})\,,$$

which is exactly the expression in (2.28), which we obtained by using the sum-product algorithm. In a similar manner, the message $\beta_{k-1}(S_{k-1})$ can be obtained as

$$\beta_{k-1}(S_{k-1}) = \sum_{b_k,S_k,\mathbf{c}_k} p(S_k,\mathbf{c}_k|S_{k-1},b_k)p(\mathbf{c}_k|\mathbf{y}_k)\beta_k(S_k), \quad k=1,\ldots,N\,.$$

This update can be interpreted as closing the right box in Fig. 2.17, and from the "closing boxes" interpretation again follows that $\beta_{k-1}(S_{k-1}) = p(S_{k-1}|\mathbf{y}_k,\ldots,\mathbf{y}_N)$.

As mentioned before, the sum-product algorithm starts at the variable nodes S_1 and S_N, and propagates the messages α and β concurrently from left to right and from right to left, respectively. After N such steps, all messages α and β have been obtained. To obtain the actual posterior density (2.26) required for the MAP decision in (2.25), we finally calculate messages γ_k according to

$$\gamma_k(b_k) = \sum_{S_{k-1},S_k,\mathbf{c}_k} p(S_k,\mathbf{c}_k|S_{k-1},b_k)p(\mathbf{c}_k|\mathbf{y}_k)\alpha_{k-1}(S_{k-1})\beta_k(S_k)\,.$$

This update can be interpretated as "box closing" around all parts of the factor graph except the variable node b_k. Therefore we have $\gamma_k(b_k) = p(b_k = b|\mathbf{y}_1,\cdots,\mathbf{y}_N)$, and the MAP decision rule (2.25) then becomes

$$\hat{b}_k = \arg\max_{b_k} \gamma_k(b_k)\,. \tag{2.30}$$

2.4.2 LDPC Codes

Low-density parity-check codes are linear binary block codes originally proposed by [47] and then rediscovered in [48]. LDPC codes offer extremely good performance; e.g. [49] presents LDPC code designs with a threshold less than 0.2 dB away from the Shannon limit, when a suitably large block length (> 100000 bits) is used. LDPC codes can be described by means of factor graphs and the decoding (and the encoding) is performed in an iterative manner by means of the sum-product algorithm.

In the following, the length-K information bit vector will be denoted by $\mathbf{u} = (u_1,\ldots,u_K)^T$ and it is mapped *linearly* to a length-N code bit-vector $\mathbf{c} = (c_1,\ldots,c_N)^T$. The code is described by the parity check matrix $\mathbf{P}$, and a bit vector $\mathbf{c}$ is a valid codeword if[3] $\mathbf{Pc} = \mathbf{0}$. Every row of the parity check matrix $\mathbf{P}$ describes a so called parity check equation, which any valid codeword $\mathbf{c}$ has to fullfill. The structure of the code can be intuitively described by means of a factor graph or a Tanner graph [3]. The factor graphs consists of variable

[3] Dealing with binary values, modulo-2 addition is used.

nodes and check nodes. The variable nodes represent the coded bits, and the function nodes represent the parity check equations as described by the rows of the parity check matrix $\mathbf{P}$ [50].

Fig. 2.18 shows a simple example of a factor graph with the parity check matrix $\mathbf{P}$ given by

$$\mathbf{P} = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix} .$$

For example, the first row of $\mathbf{P}$ corresponds to the first parity check given by $c_1 \oplus c_2 \oplus c_3 \oplus c_5 = 0$, where $\oplus$ denotes modulo-2 addition. In the factor graph graph every “1” represents an edge between a code bit and a check equation; the first row of $\mathbf{P}$ therefore represents the four connections between the first check equation and c_1, c_2, c_3, and c_5, respectively.

The term “low-density” in the name of the codes refers to the fact that the number of ones in $\mathbf{P}$ is small and grows only linearly with the block length N of the code.

Characterisation of LDPC Codes. LDPC codes can be characterized by the degrees of their variable and function nodes. *Regular codes* are those for which all nodes of the same type have the same degree. For example, a $(3, 6)$-regular LDPC code is a code in wich all variable nodes have degree 3 and all check nodes have degree 6. In case of *irregular* codes the degree of each sets of nodes is chosen according to some distribution. For example, an irregular LDPC code might have 40% variable nodes of degree 3 and 60% variable nodes of degree 5, whereas half the check nodes have degree 6 and 8, respectively. In case of irregular codes, these distributions are given by node degree distributions. The variable node degree distribution is denoted as

$$\lambda(x) = \sum_{i=2}^{d_v} \lambda_i x^{i-1}$$

and represents a code where λ_i is the fraction of variable nodes of degree i. Analogously, a check node degree distribution

$$\rho(x) = \sum_{i=2}^{d_c} \rho_i x^{i-1}$$

represents a code, where ρ_i is the fraction of check nodes of degree i. All specific LDPC codes with the same variable and check node distribution $\lambda(x)$ and $\rho(x)$ form a *code ensemble*. It can be shown [41], that in the limit of large blocklength $N \to \infty$, almost all codes of an ensemble behave alike. It therefore suffices to only consider the average behavior of the ensemble; the individual behaviour of almost all individual codes will be the same.

Decoding. In decoding, we seek a decision which minimizes the probability of a bit error. This leads to the MAP decision rule of the bit c_k and is given by

$$\hat{c}_k = \arg\max_{c_k} f(c_k | y_1, \ldots, y_N) , \tag{2.31}$$

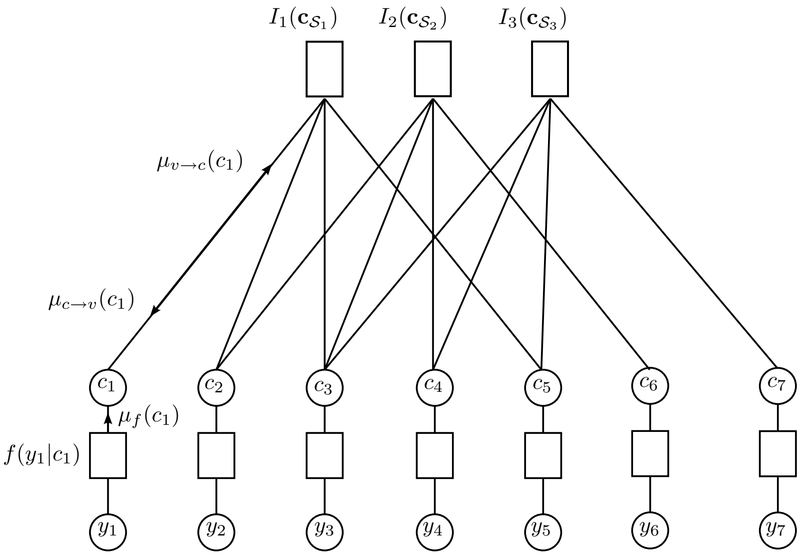


Figure 2.18: Factor graph of an LDPC code corresponding to the factorization (2.33).

where $f(c_k|y_1,\ldots,y_N)$ is the conditional probability of the bit c_k given the observation $y_1,\ldots,y_N$. We can obtain this probability by marginalization of the posterior probability

$$p(c_k|y_1,\ldots,y_N) = \sum_{\sim c_k} p(c_1,\ldots,c_N|y_1,\ldots,y_N)\,.$$

Using Bayes' theorem, the conditional probability $f(c_1,\ldots,c_N|y_1,\ldots,y_N)$ can be expressed as

$$p(c_1,\ldots,c_N|y_1,\ldots,y_N) \propto f(y_1,\ldots,y_N|c_1,\ldots,c_N)p(c_1,\ldots,c_N) = \prod_{n=1}^{N} f(y_n|c_n)I(\mathbf{c}), \qquad (2.32)$$

where we have assumed i.i.d. fading of the channel. The conditional density $f(y_n|c_n)$ describes the channel model, and $I(\mathbf{c})$ is an indicator function of the code, which is one, when the code bits $\mathbf{c}$ are a valid codeword, and zero otherwise, that is

$$I(\mathbf{c}) = \begin{cases} 1 & \text{if } \mathbf{P}\mathbf{c} = \mathbf{0} \\ 0 & \text{otherwise}\,. \end{cases}$$

For a codeword $\mathbf{c}$ to fullfill the condition $\mathbf{P}\mathbf{c} = \mathbf{0}$, the codeword $\mathbf{c}$ must satisfy *all* check-equations described by the rows of the parity check matrix $\mathbf{P}$. Therefore we can factorize the indicator function into a product of local indicator functions, each of which is associated with one row of $\mathbf{P}$. We define the set $\mathbf{c}_{\mathcal{S}_m}$ as the set of code bits involved in the mth parity check and the indicator function $I_m(\mathbf{c}_{\mathcal{S}_m})$ denotes the corresponding mth parity check node function. The indicator function $I(\mathbf{c})$ can then be rewritten as

$$I(\mathbf{c}) = \prod_{m=1}^{M} I_m(\mathbf{c}_{\mathcal{S}_m})\,.$$

Inserting this expression into (2.32), we obtain for the posterior density

$$f(c_1,\ldots,c_N|y_1,\ldots,y_N) = \prod_{n=1}^{N} f(y_n|c_n) \prod_{m=1}^{M} I_m(\mathbf{c}_{\mathcal{S}_m})\,. \qquad (2.33)$$

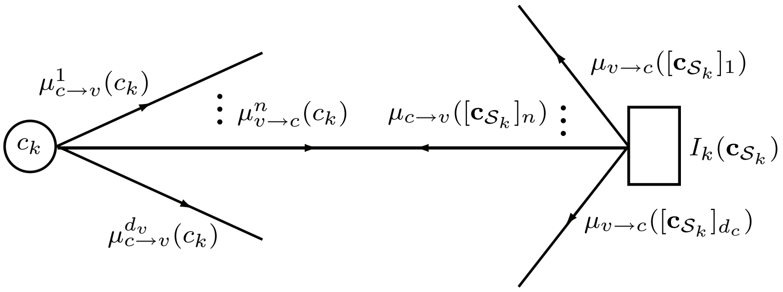

$\mu^1_{c\to v}(c_k)$
$\mu^n_{v\to c}(c_k)$
$\mu_{c\to v}([\mathbf{c}_{\mathcal{S}_k}]_n)$
$\mu_{v\to c}([\mathbf{c}_{\mathcal{S}_k}]_1)$
c_k
$I_k(\mathbf{c}_{\mathcal{S}_k})$
$\mu^{d_v}_{c\to v}(c_k)$
$\mu_{v\to c}([\mathbf{c}_{\mathcal{S}_k}]_{d_c})$

Figure 2.19: Variable- and check-node update.

In our example the set $\mathbf{c}_{\mathcal{S}_1}$ is $\mathbf{c}_{\mathcal{S}_1} = \{c_1, c_2, c_3, c_5\}$ and the corresponding first parity check function is

$$I_1(\mathbf{c}_{\mathcal{S}_1}) = I(c_1 \oplus c_2 \oplus c_3 \oplus c_5 = 0) = \begin{cases} 1 & \text{if } c_1 \oplus c_2 \oplus c_3 \oplus c_5 = 0 \\ 0 & \text{otherwise} \,. \end{cases}$$

Using this factorization, the sum-product algorithm can be used to calculate the marginal probabilites required for the MAP decision (2.31). Since the factor graph has cycles, the resulting decoding algorithm will become iterative and yields only approximations of the marginals. Numerical results, however, indicate very good performance. Next, we describe the messages and their updates in more detail.

Messages. We distinguish between two different types of messages: (i) messages from the variable nodes to the check nodes; the message on the ith edge from the variable nodes c_k to the check node is denoted by $\mu^i_{\rightarrow}(c_k)$, (ii) messages from the check nodes to the variable nodes c_k, which will be denoted by $\mu_{\leftarrow}(c_k)$. Initially, the variable nodes send out a '1', and the message $\mu_f(c_k)$ becomes $\mu_f(c_k) = f(y_1|c_1)$. In the first iteration, these messages are propagated to all check nodes.

In further iterations, the message updates are depicted in Fig. 2.19. At a variable node, the outgoing message $\mu^n_{\rightarrow}(c_k)$ is obtained as (cf. (2.22))

$$\mu^n_{\rightarrow}(c_k) = \prod_{n' \neq n} \mu^{n'}_{\leftarrow}(c_k) \,, \tag{2.34}$$

where the incoming messages $\mu^n_{\leftarrow}(c_k)$ will be defined below. Because $c_k \in \{0, 1\}$, it sufficies to use LLRs as representatives for the messages. We define

$$\tilde{\mu}^n_{\rightarrow}(c_k) \triangleq \log \frac{\mu^n_{\rightarrow}(c_k = 1)}{\mu^n_{\rightarrow}(c_k = 0)} \,,$$

and then (2.34) is replaced with

$$\tilde{\mu}^n_{\rightarrow}(c_k) = \sum_{n' \neq n} \tilde{\mu}^{n'}_{\leftarrow}(c_k) \,.$$

Here, $\tilde{\mu}^{n'}_{\leftarrow}(c_k)$ denotes the incoming LLR messages, which will be defined below. A check function node $I_k(\mathbf{c}_{\mathcal{S}_k})$ receives messages from all variable nodes in the set $\mathcal{S}_k$, and the nth variable node of this set will be denoted as $[\mathbf{c}_{\mathcal{S}_k}]_n$. On the edge connecting to this variable node, the function node outputs the message $\mu_{\leftarrow}([\mathbf{c}_{\mathcal{S}_k}]_n)$ according to the message update (2.23)

$$\mu_{\leftarrow}([\mathbf{c}_{\mathcal{S}_k}]_n) = \sum_{\sim [\mathbf{c}_{\mathcal{S}_k}]_n} I_k(\mathbf{c}_{\mathcal{S}_k}) \prod_{n' \neq n} \mu_{\rightarrow}([\mathbf{c}_{\mathcal{S}_k}]_{n'}) \,. \tag{2.35}$$

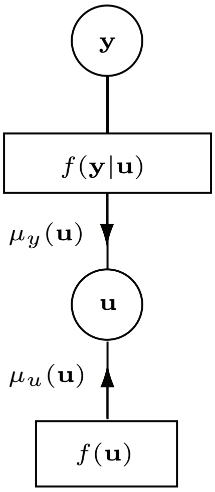


Figure 2.20: Factor graph with Gaussian messages corresponding to the model in (2.36).

We express this message by means of an LLR according to

$$\tilde{\mu}_{\leftarrow}([\mathbf{c}_{\mathcal{S}_k}]_n) \triangleq \log \frac{\mu_{\leftarrow}([\mathbf{c}_{\mathcal{S}_k}]_n = 1)}{\mu_{\leftarrow}([\mathbf{c}_{\mathcal{S}_k}]_n = 0)} \,,$$

and after some manipulations [50], the message update (2.35) can be shown to be equivalent to

$$\tilde{\mu}_{\leftarrow}([\mathbf{c}_{\mathcal{S}_k}]_n) = -2\tanh^{-1}\left(\prod_{n' \neq n} \tanh\big(\frac{1}{2}\tilde{\mu}_{\rightarrow}([\mathbf{c}_{\mathcal{S}_k}]_{n'})\big)\right) .$$

Different scheduling strategies for decoding of LDPC codes have been investigated, but the most common scheme is a flooding schedule: All variable nodes emit messages, which are processed concurrently at the check nodes. This allows for an easy parallelization. Typically, 50 to 100 iterations are performed at the decoder.

2.5 Factor Graphs with Gaussian Messages

In this Section we will consider statistical models where the exchanged messages are Gaussian functions [5]. In this case, the message update equations (2.22) and (2.23) take a particular simple form as the updated messages are again Gaussian functions. Therefore, the message update procedure reduces to calculate the updated mean and variance (or mean vector and covariance matrix in case of vector messages), and message passing amounts to exchanging mean and variance between nodes. This significantly reduces computational complexity and memory requirements for the storage of messages.

Factor graph example. To demonstrate the use of Gaussian messsages, we will consider a simple estimation problem. Of course, it is possible to solve the problem by standard techniques, but we will demonstrate the use of factor graphs and the sum-product algorithm. The system model is given by

$$\mathbf{y} = \mathbf{H}\mathbf{u} + \mathbf{w} \,. \tag{2.36}$$

Here, $\mathbf{u} = (u_1 \cdots u_N)^T$ denotes a Gaussian vector to be estimated with *a priori* density $\mathbf{u} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$, $\mathbf{H}$ is a known $M \times N$ observation matrix, and $\mathbf{w}$ denotes the Gaussian noise vector of length N ($\mathbf{w} \sim \mathcal{CN}(\mathbf{0}, \sigma_w^2 \mathbf{I})$). The vector $\mathbf{y} = (y_1 \cdots y_M)^T$ denotes the observation. Based on this observation the MMSE estimate $\mathbf{u}^{\text{MMSE}}$ of $\mathbf{u}$ shall be found. It is given by the posterior mean [51]

$$\mathbf{u}^{\text{MMSE}} = E\{\mathbf{u}|\mathbf{y}\} = \int \mathbf{u}\, f(\mathbf{u}|\mathbf{y}) d\mathbf{u} \,. \tag{2.37}$$

The posterior density $f(\mathbf{u}|\mathbf{y})$ can be expressed as [4]

$$f(\mathbf{u}|\mathbf{y}) = \frac{f(\mathbf{y}|\mathbf{u}) f(\mathbf{u})}{f(\mathbf{y})} \,,$$

where $f(\mathbf{y}|\mathbf{u})$ denotes the likelihood function, which can be obtained from the system model (2.36) as $f(\mathbf{y}|\mathbf{u}) = \mathcal{CN}(\mathbf{Hu}, \sigma_w^2 \mathbf{I})$, and $f(\mathbf{u})$ denotes the a priori density of $\mathbf{u}$. In the following we will calculate the mean vector and the covariance matrix of the Gaussian density

$$f(\mathbf{y}|\mathbf{u}) f(\mathbf{u}) \triangleq g_u(\mathbf{u})$$

by means of the factor graph framework using Gaussian message functions. The division of $g_u(\mathbf{u})$ by $f(\mathbf{y})$ to obtain $f(\mathbf{u}|\mathbf{y})$ is fairly standard, and can be found in e.g. [51]. The factor graph representing $f(\mathbf{y}|\mathbf{u}) f(\mathbf{u})$ is shown in Fig. 2.20 with variable nodes $\mathbf{u}$ and $\mathbf{y}$ and function nodes $f(\mathbf{y}|\mathbf{u})$ and $f(\mathbf{u})$.

Gaussian messages will be denoted as $\mathcal{CN}(u, \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(u-\mu)^2\right)$, following the notation of [4] (extension to multivariate Gaussian messages follows analogously). The sum-product algorithm starts with the variable node $\mathbf{y}$ emitting the initial message '1'. The message $\mu_y(\mathbf{u})$ then becomes [5]

$$\mu_y(\mathbf{u}) = f(\mathbf{y}|\mathbf{u}) \times 1 = \mathcal{CN}(\mathbf{y}, \mathbf{Hu}, \sigma_w^2 \mathbf{I}) \,.$$

For the next steps we need to reformulate this messages as Gaussian function in $\mathbf{u}$. Some simple manipulations yield

$$\mu_y(\mathbf{u}) = \mathcal{CN}\left(\mathbf{u}, \left(\mathbf{H}^H \mathbf{H}\right)^{-1} \mathbf{H}^H \mathbf{y}, \left(\frac{1}{\sigma_w^2} \mathbf{H}^H \mathbf{H}\right)^{-1}\right) .$$

The message $\mu_u(\mathbf{u})$ from the variable node $\mathbf{u}$ is given by

$$\mu_u(\mathbf{u}) = \mathcal{CN}\big(\mathbf{u}, \mathbf{0}, \mathbf{I}\big) \,.$$

The function $g_u(\mathbf{u})$ is then given by the product of the two messages $\mu_y(\mathbf{u})$ and $\mu_u(\mathbf{u})$. Using results from [5], it can be shown to be

$$g_u(\mathbf{u}) = \mu_y(\mathbf{u}) \mu_u(\mathbf{u}) \propto \mathcal{CN}\left(\mathbf{u}, \frac{1}{\sigma_w^2} \left(\frac{1}{\sigma_w^2} \mathbf{H}^H \mathbf{H} + \mathbf{I}\right)^{-1} \mathbf{H}^H \mathbf{y}, \left(\frac{1}{\sigma_w^2} \mathbf{H}^H \mathbf{H} + \mathbf{I}\right)^{-1}\right) .$$

Dividing $g_u(\mathbf{u})$ by $f(\mathbf{u})$, we finally obtain the posterior density $f(\mathbf{u}|\mathbf{y})$. The posterior mean (wich equals the MMSE estimator) $E\{\mathbf{u}|\mathbf{y}\} = \mathbf{u}^{\text{MMSE}}$ is then given by

$$\mathbf{u}^{\text{MMSE}} = E\{\mathbf{u}|\mathbf{y}\} = (\mathbf{H}^H \mathbf{H} + \sigma_w^2 \mathbf{I})^{-1} \mathbf{H}^H \mathbf{y} \,.$$

This result could have been easily obtained by standard methods (e.g. [51]), but extending the model demon-

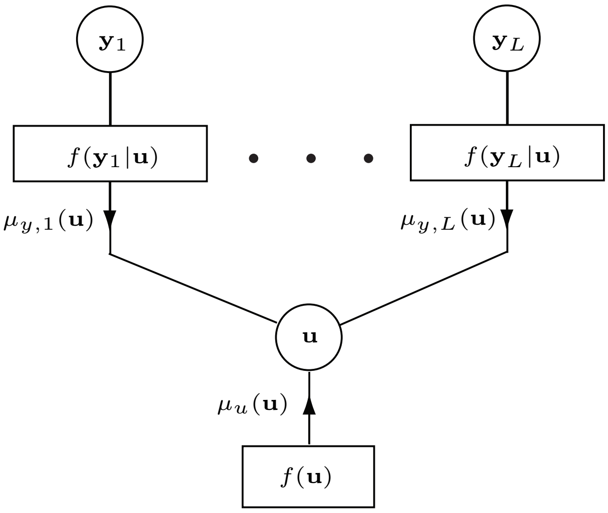


Figure 2.21: Factor Graphs with Gaussian messages corresponding to multiple observations $\mathbf{y}_1, \ldots, \mathbf{y}_L$.

strates the powerfulness of the factor graph framework. Consider a system, where L independent observations are made, that is

$$\mathbf{y}_n = \mathbf{H}\mathbf{u} + \mathbf{w}_n \, , \quad n = 1, \ldots, L \, .$$

Here, $\mathbf{y}_n$ denotes the observation at time instant n and $\mathbf{w}_n$ the noise at this time. Assuming independence of the noise over time n, the joint distribution of $\mathbf{u}$ and observations $\mathbf{y}_1, \ldots, \mathbf{y}_L$ can be written as

$$f(\mathbf{y}_1, \ldots, \mathbf{y}_L, \mathbf{u}) = f(\mathbf{u}) \prod_{n=1}^{L} f(\mathbf{y}_n|\mathbf{u}) \, .$$

The corresponding factor graph is shown in Fig. 2.21. Using the results from the example before, the marginal function $g_u(\mathbf{u})$ can be obtained from

$$g_u(\mathbf{u}) = \mu_u(\mathbf{u}) \prod_{n=1}^{L} \mu_{y,n}(\mathbf{u}) \, ,$$

where $\mu_{y,n}(\mathbf{u})$ denotes the message from the nth function node $f(\mathbf{y}_n|\mathbf{u})$ to the variable node $\mathbf{u}$. From this marginal function the MMSE estimate

$$\mathbf{u}^{\text{MMSE}} = E\{\mathbf{u}|\mathbf{y}_1, \ldots \mathbf{y}_L\}$$

can be obtained by dividing $g_u(\mathbf{u})$ by $f(\mathbf{y}_1, \cdots, \mathbf{y}_L)$. It is also possible to calculate the MMSE estimate given only a subset of L' observations $\mathbf{y}_1, \ldots, \mathbf{y}_{L'}$. In this case, only the messages $\mu_{y,n}(\mathbf{u}), n = 1, \ldots, L'$ are used in calculating the (partial) marginal function

$$g_u^{L'} = \mu_u(\mathbf{u}) \prod_{n=1}^{L'} \mu_{y,n}(\mathbf{u}) \, ,$$

yielding the MMSE estimate $\mathbf{u}^{\text{MMSE}} = E\{\mathbf{u}|\mathbf{y}_1, \ldots \mathbf{y}_{L'}\}$. Here, the factor graph approach can help with designing algorithms by providing intuitive insights into the considered problem.

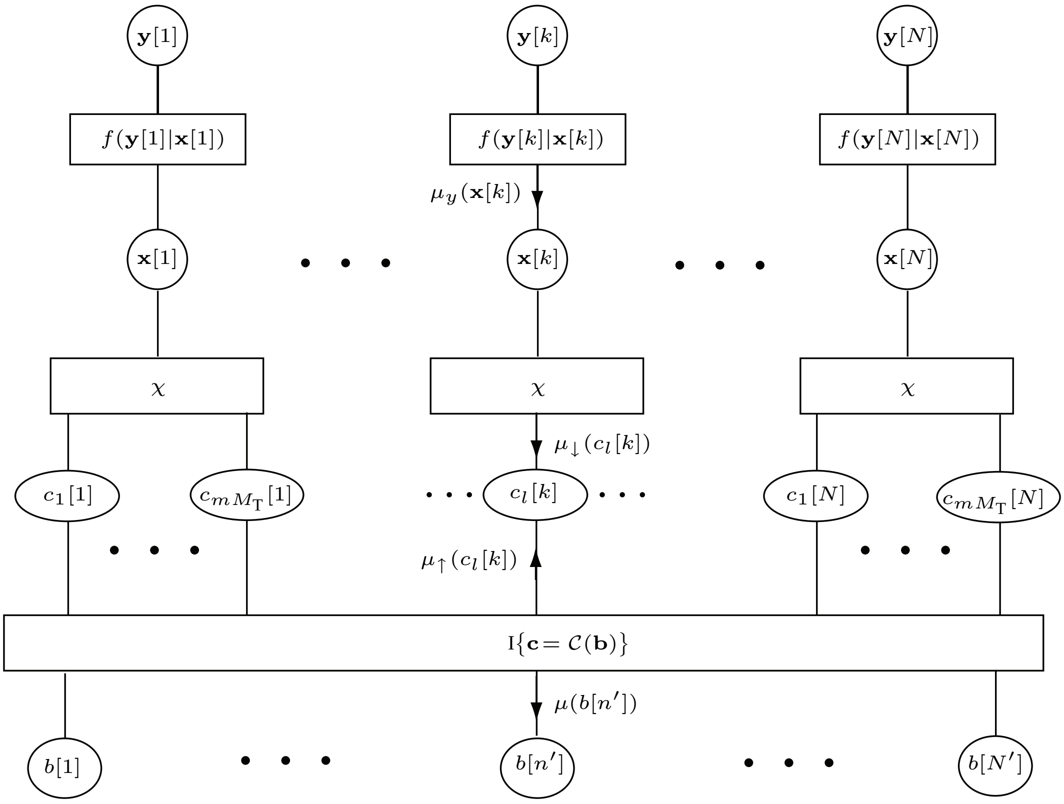


Figure 2.22: Factor graph representation of a BICM system.

2.6 Derivation of the BICM Receiver by Means of Factor Graphs

In this section we will demonstrate how the BICM receiver from Subsection 2.1.3 can be (re)obtained by means of the factor graph framework.

Derivation of factor graph. We consider a MAP detector, i.e. the receiver decides on the ith information bit $b[n']$ according to

$$\hat{b}[n'] = \arg \max_{b[n'] \in \{0,1\}} p(b[n']|\mathbf{r}) \,, \tag{2.38}$$

where $\mathbf{r} = \left(\mathbf{r}^T[1] \cdots \mathbf{r}^T[N]\right)^T$ is the received vector sequence and $p(b[n']|\mathbf{r})$ denotes the conditional probability of $b[n']$ given $\mathbf{r}$. To compute $p(b[n']|\mathbf{r})$ in (2.38), we first write it as a marginal of $p(\mathbf{b}|\mathbf{r})$ and apply Bayes' rule (assuming *a priori* equally likely information bit sequences $\mathbf{b}$):

$$p(b[n']|\mathbf{r}) \;=\; \sum_{\sim b[n']} p(\mathbf{b}|\mathbf{r}) \;\propto\; \sum_{\sim b[n']} f(\mathbf{r}|\mathbf{b}) \tag{2.39}$$

where $\sum_{\sim b[n']}$ denotes summation with respect to all components of $\mathbf{b}$ except $b[n']$, $f(\mathbf{r}|\mathbf{b})$ is the conditional probability density function of $\mathbf{r}$ given $\mathbf{b}$, and $\propto$ denotes equality up to factors irrelevant to the maximization in (2.38). We can further express the density $f(\mathbf{r}|\mathbf{b})$ according to $f(\mathbf{r}|\mathbf{b}) = \sum_{\mathbf{X},\mathbf{c}} f(\mathbf{r},\mathbf{X},\mathbf{c}|\mathbf{b}) =$

$\sum_{\mathbf{X},\mathbf{c}} f(\mathbf{r}|\mathbf{X})p(\mathbf{X}|\mathbf{c})p(\mathbf{c}|\mathbf{b})$, and can write (2.39) as

$$p(b[n']|\mathbf{r}) \propto \sum_{\sim b[n']} f(\mathbf{r}|\mathbf{X})p(\mathbf{X}|\mathbf{c})p(\mathbf{c}|\mathbf{b})\,. \tag{2.40}$$

Note that $f(\mathbf{r}|\mathbf{X})$ corresponds to the channel (cf. (2.2)), $p(\mathbf{X}|\mathbf{c})$ describes the modulator (symbol mappings $\mathbf{x}[n] = \chi\big(\mathbf{c}[n]\big)$), and $p(\mathbf{c}|\mathbf{b})$ represents the channel encoder and interleaver (one-to-one correspondences $\mathbf{c} = \mathcal{C}(\mathbf{b})$). There is $p(\mathbf{c}|\mathbf{b}) = 1$ if $\mathbf{c} = \mathcal{C}(\mathbf{b})$ and $p(\mathbf{c}|\mathbf{b}) = 0$ otherwise. We can rewrite this constraint by means of the indicator function $\mathrm{I}\{\cdot\}$, which equals 1 if its argument is true and 0 otherwise, according to

$$p(\mathbf{c}|\mathbf{b}) = \mathrm{I}\big\{\mathbf{c} = \mathcal{C}(\mathbf{b})\big\}\,. \tag{2.41}$$

We note that the code constraint $\mathrm{I}\big\{\mathbf{c} = \mathcal{C}(\mathbf{b})\big\}$ can be expressed in a more refined manner as was discussed in Subsection 2.4. Similar reasoning yields for the conditional density $p(\mathbf{X}|\mathbf{c})$ describing the mapping

$$p(\mathbf{X}|\mathbf{c}) = \prod_{n=1}^{N} \mathrm{I}\big\{\mathbf{x}[n] = \chi(\mathbf{c}[n])\big\}\,. \tag{2.42}$$

Finally, because the receive vectors $\mathbf{r}[n]$ are conditionally independent (cf. (2.2)) given the transmit vectors $\mathbf{x}[n]$ we have

$$f(\mathbf{r}|\mathbf{X}) = \prod_{n=1}^{N} f(\mathbf{r}[n]|\mathbf{x}[n])\,. \tag{2.43}$$

Inserting the expressions (2.41)–(2.43) into (2.40), we obtain the overall factorization

$$p(b[n']|\mathbf{r}) \propto \sum_{\sim b[n']} \prod_{n=1}^{N} \mathrm{I}\big\{\mathbf{x}[n] = \chi(\mathbf{c}[n])\big\} \prod_{n'=1}^{N} f(\mathbf{r}[n']|\mathbf{x}[n'])\mathrm{I}\big\{\mathbf{c} = \mathcal{C}(\mathbf{b})\big\} \tag{2.44}$$

which can be represented by the factor graph shown in Fig. 2.22. There are factor nodes for the channel, symbol mapper constraints, and code constraints, and variable nodes for the transmit symbols, code bits, and information bits.

Messages and sum-product algorithm. The factor graph in Fig. 2.22 has cycles, therefore the sum-product algorithm becomes iterative, yields only approximate marginal functions and requires appropriate scheduling of message updates. Depending on this scheduling different receiver algorithms can be obtained. The message $\mu_y(\mathbf{x}[k])$ is given as

$$\mu_y(\mathbf{x}[k]) = \mathcal{CN}(\mathbf{y}[k], \mathbf{H}[k]\mathbf{x}[k], \sigma_w^2\mathbf{I})\,.$$

Using the update rule (2.22) of the sum-product algorithm, we obtain for the message

$$\mu_\downarrow(c_l[k] = c) = \sum_{\mathbf{x}} \mathrm{I}\big\{\mathbf{x}[k] = \chi(\mathbf{c}[k])\big\}\, \mu_y(\mathbf{x}[k]) \prod_{l' \neq l} \mu_\uparrow(c_{l'}[k])\,.$$

In case of an non-iterative receiver and an iterative receiver in the first iteration, we have for the messages $\mu_\uparrow(c_{l'}[k]) = \text{const.}$. This simplifies the message $\mu_c(c_l[k])$ to

$$\mu_\downarrow(c_l[k] = c) = \sum_{\mathbf{x}:c_l[k]=c} \mu_y(\mathbf{x}[k])\,. \tag{2.45}$$

Converting this message into an LLR, we obtain the expression (2.5) for the soft-out BICM demodulator. In case of an iterative receiver (where the messages come from applying the sum-product algorithm to the code part I$\{\mathbf{c}=\mathcal{C}(\mathbf{b})\}$ of the factor graph), we have

$$\mu_{\downarrow}(c_l[k]=c)=\sum_{\mathbf{x}:c_l[k]=c}\mu_y(\mathbf{x}[k])\prod_{l'\neq l}\mu_{\uparrow}(c_{l'}[k])\,, \tag{2.46}$$

which is the same as (2.8). The messages $\mu_{\downarrow}(c_l[k]=c)$ are now used to decode the channel code as described in Subsection 2.4. Applying the sum-product algorithm to the factor graph part of the channel code (that is, the function node I$\{\mathbf{c}=\mathcal{C}(\mathbf{b})\}$)), yields the messages $\mu(b[n'])$ for the information bits $b[n'], n'=1,\ldots,N'$. Using these messages, the MAP decision (or an approximate decision in case of an iterative receiver) (2.38) can be made. If an iterative receiver is to be used, then the sum-product algorithm of the channel code also calculates the messages $\mu_{\uparrow}(c_l[k])$, which are then used in (2.46) in the next iteration.

These derivations demonstrate the powerfulness of the factor graph framework. By deriving an expression for the joint density describing the system model (2.44) and representing the detection problem as operating on the marginals of this joint density (2.38), different algorithms for the receiver design can be obtained. Some extensions are also possible: By approximating the message updates (2.45) and (2.46), several low-complexity algorithms for BICM demodulators can be obtained. It is also possible to consider a system with pilot-based estimation of the channel. By properly incorporating this extension in the expression for the joint density and applying the sum-product algorithm to the resulting factor graph, receivers which perform joint channel estimation and data detection can be obtained. These concepts will be explained in Chapter 3.

3

Interleave-Division Multiple Access Transmission Schemes

IN multiuser communications, large performance gains can be achieved by using a turbo-style interaction between multiuser detector and channel decoder [52]. Here, we consider an uplink scenario where U users transmit data to a common base station via multiple-input multiple-output (MIMO) channels. User separation is achieved by means of a recently introduced multiple-access technique known as *interleave-division multiple access* (IDMA) [18]. Using a factor graph framework [4, 5], we develop an iterative multiuser MIMO-IDMA receiver that performs joint multiuser data detection, channel decoding, and pilot-aided channel estimation. An orthogonal frequency-division multiplex (OFDM) modulation format is adopted to accommodate frequency-selective (time-dispersive) channels. The proposed receiver is suited to higher-order symbol alphabets for increased spectral efficiency, and it uses a selective message update scheme for reduced complexity.

3.1 Background and State of the Art

In a multiuser system employing code-division multiple access (CDMA), user separation is achieved by means of user-specific pseudorandom spreading sequences [53]. Recently, IDMA has been proposed as an alternative to CDMA [18]. With IDMA, user separation is obtained via user-specific interleavers combined with low-rate channel coding. Similarly to CDMA, IDMA offers diversity against fading and mitigation of inter-cell interference [18]. However, IDMA has some important advantages over CDMA: it enables the use of multiuser detectors that are significantly less complex than those required for CDMA, and it can outperform coded CDMA when iterative (turbo) receivers are used [18]. Furthermore, IDMA can be easily used in MIMO systems [54]. Just as for CDMA, frequency-selective channels can be accommodated by combining IDMA with an OFDM modulation format [55, 56].

An iterative receiver for IDMA based on a minimum mean square error (MMSE) equalization approach was described in [18] for single-antenna systems using binary (BPSK) modulation. OFDM-IDMA was introduced in [56]. An extension of IDMA to MIMO and to higher-order symbol alphabets, as well as receivers based on factor graphs and performing joint detection and channel estimation were proposed in [54, 57–59].

Factor graphs were introduced in Chapter 1 as a tool for dealing with functions of many variables that can be factored into "local" functions, each of which depends only on a subset of these variables. Such a factorization can be graphically represented by a factor graph. Marginals of the global function can be efficiently calculated (either exactly or approximately) by means of a message-passing algorithm known as the sum-product algorithm. The factor graph framework and the sum-product algorithm have been used in many different areas such as signal processing [4, 5], receiver design [13], and decoding of LDPC codes [49].

This chapter is organized as follows. The system model and IDMA transmitter are presented in Section 3.2. In Section 3.3, we construct the factor graph of our system and derive the messages to be propagated along the edges of the factor graph. Section 3.4 develops message approximations and a selective message updating scheme resulting in a complexity that is linear in the number of users. Information-theoretic performance limits of IDMA are determined in Section 3.5. Finally, Section 3.6 demonstrates the performance of the proposed receiver structures and algorithms by means of numerical simulations.

3.2 MIMO-OFDM-IDMA System Model

We consider a MIMO-OFDM-IDMA system for an uplink multiple-access scenario with U users. Each user employs M_{T} transmit antennas for spatial multiplexing [40], and the base station is equipped with M_{R} receive antennas. Frequency-selective Rayleigh fading channels are assumed, and the equivalent discrete-time complex baseband domain (with symbol-rate sampling) is considered throughout.

3.2.1 Transmitter

The MIMO-OFDM-IDMA transmitter for the uth user is shown in Fig. 3.1. This transmitter extends the BPSK-based MIMO-IDMA transmitter of [54] to OFDM modulation and higher-order symbol alphabets.

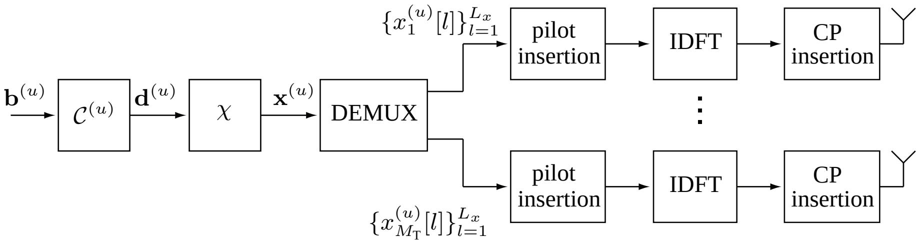


Figure 3.1: MIMO-OFDM-IDMA transmitter for the uth user.

Coding, interleaving, mapping. First, a length-J sequence of information bits of the uth user, denoted as $\mathbf{b}^{(u)} \triangleq \big(b_1^{(u)} \cdots b_J^{(u)}\big)^T$, is encoded into a length-L sequence of code bits, with $L > J$. The code rate is thus $R = J/L < 1$. The code is a serial concatenation of a terminated convolutional code and a low-rate repetition code. The code bit sequence is then passed through a user-specific interleaver $\pi^{(u)}$, which yields the interleaved bit sequence $\mathbf{c}^{(u)} \triangleq \big(c_1^{(u)} \cdots c_L^{(u)}\big)^T$. We can express $\mathbf{c}^{(u)}$ as

$$\mathbf{c}^{(u)} = \mathcal{C}^{(u)}(\mathbf{b}^{(u)})\,, \tag{3.1}$$

where the one-to-one function $\mathcal{C}^{(u)}(\cdot)$ denotes the combined effect of channel coding and interleaving. Different users employ identical codes but different interleavers. The low-rate repetition code together with the user-specific interleaver replaces the spreading employed in CDMA systems.

Next, the coded and interleaved bit sequence $\mathbf{c}^{(u)}$ is mapped to a sequence of L_x complex symbol vectors $\mathbf{x}^{(u)}[l] \triangleq \big(x_1^{(u)}[l] \cdots x_{M_\mathrm{T}}^{(u)}[l]\big)^T$, $l = 1, \ldots, L_x$ as follows. We use a complex symbol alphabet $\mathcal{S}$ of size $|\mathcal{S}| = 2^B$. Note that L_x is related to the number L of code bits as

$$L_x = L/(M_\mathrm{T}B)\,. \tag{3.2}$$

The transmit symbol $x_i^{(u)}[l] \in \mathcal{S}$ at the ith antenna of the uth user at symbol time l is obtained by mapping a group of B successive coded/interleaved bits $c_{\lambda(l,i)+1}^{(u)}, \ldots, c_{\lambda(l,i)+B}^{(u)}$, with $\lambda(l,i) \triangleq \big[(l-1)M_\mathrm{T} + i-1\big]B$, to a symbol from $\mathcal{S}$. This mapping operation will be denoted as

$$x_i^{(u)}[l] = \chi\big(\mathbf{c}_i^{(u)}[l]\big)\,,$$

with the one-to-one symbol mapping χ and the bit vector $\mathbf{c}_i^{(u)}[l] \triangleq \big(c_{\lambda(l,i)+1}^{(u)} \cdots c_{\lambda(l,i)+B}^{(u)}\big)^T$. We will refer to the bit vector $\mathbf{c}_i^{(u)}[l]$ as the *symbol label* associated with the symbol $x_i^{(u)}[l]$. With a slight abuse of notation, the transmit symbol vector (across the antennas) of the uth user will be similarly written as

$$\mathbf{x}^{(u)}[l] = \chi\big(\mathbf{c}^{(u)}[l]\big)\,, \tag{3.3}$$

where $\mathbf{c}^{(u)}[l] \triangleq \big(\mathbf{c}_1^{(u)T}[l] \cdots \mathbf{c}_{M_\mathrm{T}}^{(u)T}[l]\big)^T$.

OFDM modulation and pilot insertion. The symbols $x_i^{(u)}[l]$, $l = 1, \ldots, L_x$ are transmitted at the ith antenna of the uth user using OFDM modulation, i.e., the lth symbol $x_i^{(u)}[l]$ modulates a corresponding subcarrier. In addition, to enable channel estimation, K_pM_T pilot symbols are inserted for each user. The total number of subcarriers is thus

$$K \triangleq L_x + K_pUM_\mathrm{T}, \tag{3.4}$$

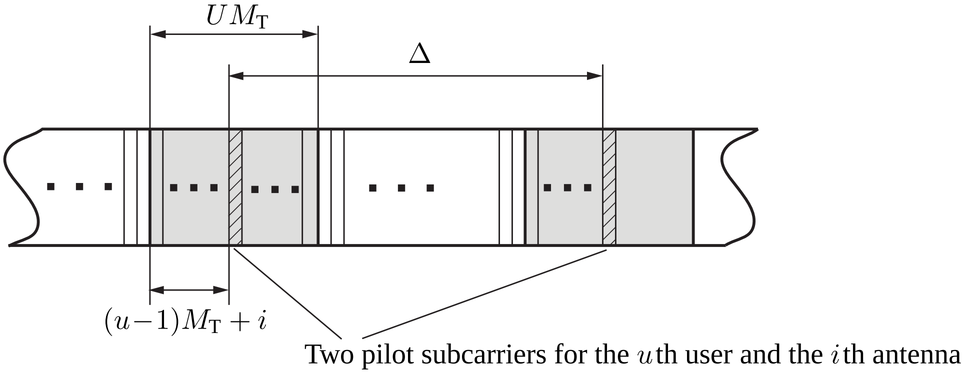

UM_{T}
Δ
$(u-1)M_{\mathrm{T}}+i$
Two pilot subcarriers for the uth user and the ith antenna

Figure 3.2: Arrangement of data subcarriers (white) and pilot subcarriers (grey). Two pilot blocks of length UM_{T} spaced Δ subcarriers apart are shown.

where K_pUM_{T} is the total number of pilot symbols (for all U users and all M_{T} transmit antennas). In what follows, $k \in \{0,\ldots,K-1\}$ will denote the subcarrier index. A symbol (data or pilot) of the uth user that is transmitted at the ith antenna and modulates the kth subcarrier is generically denoted as $s_i^{(u)}[k]$. The set $\{s_i^{(u)}[k]\}_{k=0}^{K-1}$ is composed of data symbols $\{x_i^{(u)}[l]\}_{l=1}^{L_x}$ and pilot symbols. Furthermore, the symbol vector (either data or pilot) across the antennas associated with subcarrier k will be denoted as $\mathbf{s}^{(u)}[k] \triangleq \big(s_1^{(u)}[k] \cdots s_{M_{\mathrm{T}}}^{(u)}[k]\big)^T$.

Finally, the time-domain signal of the uth user transmitted at the ith antenna is obtained by applying an inverse discrete Fourier transform (IDFT) of length K to the symbol sequence $\{s_i^{(u)}[k]\}_{k=0}^{K-1}$ and inserting a cyclic prefix of length N_{cp}. This results in a time-domain sequence (or block) of length $N \triangleq K + N_{\mathrm{cp}}$, which will be denoted as $\{\tilde{s}_i^{(u)}[n]\}_{n=1}^{N}$ (the tilde $\tilde{}$ indicates the time domain). The corresponding time-domain vector sequence will then be denoted as $\{\tilde{\mathbf{s}}^{(u)}[n]\}_{n=1}^{N}$, where $\tilde{\mathbf{s}}^{(u)}[n] \triangleq \big(\tilde{s}_1^{(u)}[n] \cdots \tilde{s}_{M_{\mathrm{T}}}^{(u)}[n]\big)^T$.

Pilot arrangement. We use the arrangement of data and pilot subcarriers shown in Fig. 3.2. The K_pUM_{T} pilot subcarriers are arranged in K_p blocks, each containing UM_{T} pilot subcarriers (one for each user and transmit antenna). Successive pilot subcarriers for a given user and transmit antenna are located in successive blocks and spaced Δ subcarriers apart (see Fig. 3.2). Note that whereas each data subcarrier is used jointly by all users and all transmit antennas, each pilot subcarrier is only used by one user for one transmit antenna; this amounts to an orthogonal pilot structure. The set of pilot subcarrier indices k employed by the uth user for the ith antenna is given by $\mathcal{P}_i^{(u)} \triangleq \{(u-1)M_{\mathrm{T}} + i + \nu\,\Delta - 1 \,|\, \nu = 0,\ldots,K_p-1\}$; note that $\mathcal{P}_i^{(u)} \subset \{0,\ldots,K-1\}$ and $|\mathcal{P}_i^{(u)}| = K_p$. The pilot symbols are chosen equal for all users and antennas, therefore $\mathbf{s}^{(u)}[k] = p\,\mathbf{e}_i$ for $k \in \mathcal{P}_i^{(u)}$, where p is the common pilot symbol and $\mathbf{e}_i$ denotes the ith unit vector of length M_{T}. Note that for $k \in \mathcal{P}_i^{(u)}$, the symbols at all other antennas, $s_{i'}^{(u)}[k]$ for $i' \neq i$, are zero. We have both $K = K_p\Delta$ and $K = L_x + K_pUM_{\mathrm{T}}$, whence $K_p(\Delta - UM_{\mathrm{T}}) = L_x$ or equivalently

$$K_p = \frac{L_x}{\Delta - UM_{\mathrm{T}}}. \tag{3.5}$$

Finally, the union of all the (disjoint) sets $\mathcal{P}_i^{(u)}$ will be denoted as $\mathcal{P}$. Note that $|\mathcal{P}| = K_pUM_{\mathrm{T}}$, which is the total number of pilot subcarriers.

Spectral efficiency. As a measure of spectral efficiency for one user, we calculate the ratio of the number of information bits J to the total number of subcarriers K. Using the definitions and relations provided previously, we obtain

$$\epsilon \triangleq \frac{J}{K} = \frac{M_{\mathrm{T}}BR\,(\Delta - UM_{\mathrm{T}})}{1+\frac{UM_{\mathrm{T}}}{\Delta - UM_{\mathrm{T}}}} = \frac{M_{\mathrm{T}}BR\,(\Delta - UM_{\mathrm{T}})^2}{\Delta}\,.$$

Most importantly, the spectral efficiency depends on the difference between the pilot block spacing Δ and the product of the number of users and transmit antennas UM_{T}. Assuming a fixed Δ, the spectral efficiency can become very small (even zero), if either the number of users or transmit antennas becomes large, because then the number of pilot subcarriers increases (we have UM_{T} pilot subcarriers in one block), which reduces spectral efficiency.

3.2.2 Channel

Considering the equivalent complex baseband after symbol-rate sampling, the received signal vector $\tilde{\mathbf{r}}[n] \triangleq \big(\tilde{r}_1[n] \cdots \tilde{r}_{M_{\mathrm{R}}}[n]\big)^T$ at symbol time n is given by

$$\tilde{\mathbf{r}}[n] = \sum_{u=1}^{U} \sum_{n'=0}^{L_{\mathrm{ch}}-1} \widetilde{\mathbf{H}}^{(u)}[n']\,\tilde{\mathbf{s}}^{(u)}[n-n'] + \tilde{\mathbf{w}}[n]\,, \qquad n = 1,\ldots,N\,. \tag{3.6}$$

Here, $L_{\mathrm{ch}} \leq N_{\mathrm{cp}}$ is the channel length, the $M_{\mathrm{R}} \times M_{\mathrm{T}}$ matrix sequence $\widetilde{\mathbf{H}}^{(u)}[n']$, $\nu = 0,\ldots,L_{\mathrm{ch}}-1$ is the MIMO channel impulse response from the uth user to the base station, the elements of the noise vector $\tilde{\mathbf{w}}[n]$ are i.i.d. complex Gaussian with noise variance σ_w^2, and $N = K + N_{\mathrm{cp}}$ is the number of symbols per temporal block.

The channels of all users are frequency-selective. Let $\tilde{h}_{j,i}^{(u)}[n']$ denote the (j,i)th element of $\widetilde{\mathbf{H}}^{(u)}[n']$, and let us collect the impulse response $\tilde{h}_{j,i}^{(u)}[n']$ of length L_{ch} in the length-K vector $\tilde{\mathbf{h}}_{j,i}^{(u)} \triangleq \big(\tilde{h}_{j,i}^{(u)}[0] \cdots \tilde{h}_{j,i}^{(u)}[L_{\mathrm{ch}}-1]\; 0 \cdots 0\big)^T$. The L_{ch} potentially nonzero channel taps are assumed uncorrelated and zero-mean complex Gaussian, i.e., $\tilde{\mathbf{h}}_{j,i}^{(u)} \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}_{\tilde{h}_p})$, where $\mathbf{C}_{\tilde{h}_p}$ (subscript p stands for "prior") is a diagonal $K \times K$ matrix in which only the first L_{ch} diagonal elements are nonzero. The corresponding frequency-domain channel coefficient vector is given by

$$\mathbf{h}_{j,i}^{(u)} \triangleq \big(h_{j,i}^{(u)}[0] \cdots h_{j,i}^{(u)}[K-1]\big)^T = \mathbf{F}\tilde{\mathbf{h}}_{j,i}^{(u)}\,, \tag{3.7}$$

where $\mathbf{F}$ is the unitary $K \times K$ DFT matrix with elements $(\mathbf{F})_{kl} = \frac{1}{\sqrt{K}} \exp\big(-j\frac{2\pi kl}{K}\big)$.

3.2.3 Receiver

The receiver processes the channel output sequence $\tilde{\mathbf{r}}[n]$, $n = 1,\ldots,N$ in (3.6). First the cyclic prefix is removed, resulting in a sequence of length K. Subsequently, a DFT with respect to n yields the frequency-domain sequence of receive vectors $\mathbf{r}[k]$, $k = 0,\ldots,K-1$. These K vectors of length M_{R} are stacked in the *(total) received vector* $\mathbf{r} = \big(\mathbf{r}^T[0] \cdots \mathbf{r}^T[K-1]\big)^T$ of length KM_{R}. The further processing steps (joint iterative data detection, channel decoding, and channel estimation) that ultimately result in the detected/decoded user bits will be discussed in Sections 3.3 and 3.4. This discussion will use the compact system channel representation described next.

3.2.4 System Channel

The frequency-domain sequence of receive vectors $\mathbf{r}[k]$ consists of L_x vectors at the data subcarriers $k \notin \mathcal{P}$ and $K_p U M_{\mathrm{T}}$ vectors at the pilot subcarriers $k \in \mathcal{P}$ or, more specifically, $k \in \mathcal{P}_i^{(u)}$, i.e.,

$$\mathbf{r}[k] \,=\, \begin{cases} \sum_{u=1}^{U} \mathbf{H}^{(u)}[k]\, \mathbf{s}^{(u)}[k] + \mathbf{w}[k]\,, & k \notin \mathcal{P}, \\ \mathbf{H}^{(u)}[k]\, \mathbf{s}^{(u)}[k] + \mathbf{w}[k]\,, & k \in \mathcal{P}_i^{(u)} \;(u = 1, \dots, U;\; i = 1, \dots, M_{\mathrm{T}})\,. \end{cases} \tag{3.8}$$

Here, $\mathbf{H}^{(u)}[k]$, $k \,=\, 0, \dots, K-1$ is the DFT of the time-domain channel impulse response matrix sequence $\widetilde{\mathbf{H}}^{(u)}[n']$, $n' = 0, \dots, L_{\mathrm{ch}}-1$ (zero-padded to a length-K sequence), and $\mathbf{w}[k]$ denotes white Gaussian noise of variance σ_w^2. For our later derivation and discussion of the iterative receiver, it will be convenient to represent the received vectors $\mathbf{r}[k]$ at *data* subcarriers $k \notin \mathcal{P}$ in terms of the transmitted data symbol vectors $\mathbf{x}^{(u)}[l]$, $l = 1, \dots, L_x$, i.e.,

$$\mathbf{r}_d[l] \,=\, \sum_{u=1}^{U} \mathbf{H}_d^{(u)}[l]\, \mathbf{x}^{(u)}[l] + \mathbf{w}[l]\,, \qquad l = 1, \dots, L_x\,, \tag{3.9}$$

where the subscript d indicates that the received symbol $\mathbf{r}_d[l]$ belongs to a data subcarrier, $\mathbf{H}_d^{(u)}[l]$ is the corresponding chanel matrix on this data subcarrier, and $l \in \{1, \dots, L_x\}$ is a re-mapped subcarrier index k. The received vectors $\mathbf{r}[k]$ at *pilot* subcarriers $k \in \mathcal{P}_i^{(u)}$ are given by

$$\mathbf{r}_p[k] \,=\, \mathbf{H}^{(u)}[k]\, \mathbf{s}^{(u)}[k] + \mathbf{w}[k] \,=\, p\mathbf{h}_i^{(u)}[k] + \mathbf{w}[k]\,, \qquad k \in \mathcal{P}_i^{(u)}, \tag{3.10}$$

where $\mathbf{h}_i^{(u)}[k]$ is the ith column of the channel matrix $\mathbf{H}^{(u)}[k]$ and the subscript p indicates that the received symbol $\mathbf{r}_p[k]$ belongs to a pilot subcarrier.

3.3 Factor Graph Framework for MIMO-OFDM-IDMA

In this section, we analyze the statistical structure of the MIMO-OFDM-IDMA system. We construct the corresponding factor graph and derive the messages to be propagated along its edges according to the sum-product algorithm. This leads quite naturally to an iterative technique for joint data detection, channel decoding, and pilot-based channel estimation.

3.3.1 Derivation of the Factor Graph

After the receiver frontend described in Section 3.2.3, the channel output is represented by the total received vector $\mathbf{r}$. The MIMO-OFDM-IDMA receiver operates on $\mathbf{r}$ and produces detected information bits $\hat{b}_m^{(u)}$, $m = 1, \dots, J$ for all users ($u = 1, \dots, U$). The receiver we propose is an approximation to the optimal (MAP) bit detector given by (cf. Subsection 2.6, [13, 60])

$$\hat{b}_m^{(u)} \,=\, \arg \max_{b_m^{(u)} \in \{0,1\}} \; p(b_m^{(u)}|\mathbf{r})\,, \qquad m = 1, \dots, J;\; u = 1, \dots, U\,. \tag{3.11}$$

Here, $p(b_m^{(u)}|\mathbf{r})$ denotes the conditional probability mass function (pmf) of the information bit $b_m^{(u)}$ given $\mathbf{r}$. In what follows, let $\mathbf{b} \,=\, \big(\mathbf{b}^{(1)T} \cdots \mathbf{b}^{(U)T}\big)^T$ and $\mathbf{c} \,=\, \big(\mathbf{c}^{(1)T} \cdots \mathbf{c}^{(U)T}\big)^T$ denote the vectors containing all information bits and code bits, respectively, of all users; furthermore, let $\mathbf{X} = (\mathbf{X}[1] \cdots \mathbf{X}[L_x])$ with $\mathbf{X}[l] = \big(\mathbf{x}^{(1)}[l] \cdots \mathbf{x}^{(U)}[l]\big)$ be the $M_{\mathrm{T}} \times UL_x$ matrix of all data vectors $\mathbf{x}^{(u)}[l]$, $l = 1, \ldots, L_x$, $u = 1, \ldots, U$. Note that there is a one-to-one correspondence between $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{X}$, due to the coding-interleaving mappings $\mathbf{c}^{(u)} = \mathcal{C}^{(u)}(\mathbf{b}^{(u)})$ in (3.1) and the modulation mapping $\mathbf{x}^{(u)}[l] = \chi\big(\mathbf{c}^{(u)}[l]\big)$ in (3.3). Thus, the quantities $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{X}$ are probabilistically equivalent. In what follows, we assume that all information bit sequences $\mathbf{b} \in \{0,1\}^{JU}$ are a priori equally likely.

To compute $p(b_m^{(u)}|\mathbf{r})$ in (3.11), we first write it as a marginal of $p(\mathbf{b}|\mathbf{r})$ and apply Bayes' rule:

$$p(b_m^{(u)}|\mathbf{r}) \,=\, \sum_{\sim b_m^{(u)}} p(\mathbf{b}|\mathbf{r}) \,\propto\, \sum_{\sim b_m^{(u)}} f(\mathbf{r}|\mathbf{b})\,. \tag{3.12}$$

Here, $\sum_{\sim b_m^{(u)}}$ denotes summation with respect to all components of $\mathbf{b}$ except $b_m^{(u)}$, $f(\mathbf{r}|\mathbf{b})$ is the conditional probability density function (pdf) of $\mathbf{r}$ given $\mathbf{b}$, and $\propto$ denotes equality up to factors irrelevant to the maximization in (3.11). Since $\mathbf{b} - \mathbf{c} - \mathbf{X}$ for a Markov chain, we have

$$f(\mathbf{r}|\mathbf{b}) \,=\, \sum_{\mathbf{X},\mathbf{c}} f(\mathbf{r},\mathbf{X},\mathbf{c}|\mathbf{b}) \,=\, \sum_{\mathbf{X},\mathbf{c}} f(\mathbf{r}|\mathbf{X},\mathbf{c},\mathbf{b})\, p(\mathbf{X}|\mathbf{c},\mathbf{b})\, p(\mathbf{c}|\mathbf{b}) \,=\, \sum_{\mathbf{X},\mathbf{c}} f(\mathbf{r}|\mathbf{X})\, p(\mathbf{X}|\mathbf{c})\, p(\mathbf{c}|\mathbf{b})\,,$$

and therefore we can write (3.12) as

$$p(b_m^{(u)}|\mathbf{r}) \,\propto\, \sum_{\sim b_m^{(u)}} f(\mathbf{r}|\mathbf{X})\, p(\mathbf{X}|\mathbf{c})\, p(\mathbf{c}|\mathbf{b})\,. \tag{3.13}$$

Here and subsequently, $\sum_{\sim b_m^{(u)}}$ denotes summation with respect to *all* unknown variables appearing in the summand except $b_m^{(u)}$ (in the present case, these variables are $\mathbf{X}$, $\mathbf{c}$, and all components of $\mathbf{b}$ except $b_m^{(u)}$).

We will now provide expressions for the conditional probability distributions appearing in (3.13). First, $p(\mathbf{c}|\mathbf{b})$ corresponds to the channel encoder and interleaver subsumed by the mappings $\mathbf{c}^{(u)} = \mathcal{C}^{(u)}(\mathbf{b}^{(u)})$, $u = 1, \ldots, U$; we thus have

$$p(\mathbf{c}|\mathbf{b}) \,=\, \begin{cases} 1, & \text{if } \mathbf{c}^{(u)} = \mathcal{C}^{(u)}(\mathbf{b}^{(u)}) \;\text{ for } u = 1, \ldots, U \\ 0, & \text{otherwise.} \end{cases} \tag{3.14}$$

Using the indicator function $\mathrm{I}\{\cdot\}$, which equals 1 if its argument is true and 0 otherwise, we can rewrite (3.14) as

$$p(\mathbf{c}|\mathbf{b}) \,=\, \prod_{u=1}^{U} \mathrm{I}\big\{\mathbf{c}^{(u)} \!=\! \mathcal{C}^{(u)}(\mathbf{b}^{(u)})\big\}\,. \tag{3.15}$$

We note that the code constraint $\mathrm{I}\big\{\mathbf{c}^{(u)} \!=\! \mathcal{C}^{(u)}(\mathbf{b}^{(u)})\big\}$ can be expressed in a more detailed manner by using the code structure as was presented in Chapter 2.

Next, the conditional pmf $p(\mathbf{X}|\mathbf{c})$ corresponds to the modulator defined by the mapping $\mathbf{x}^{(u)}[l] = \chi\big(\mathbf{c}^{(u)}[l]\big)$, $u = 1, \ldots, U$, i.e.,

$$p(\mathbf{X}|\mathbf{c}) \,=\, \begin{cases} 1, & \text{if } x_i^{(u)}[l] = \chi(\mathbf{c}_i^{(u)}[l]) \;\text{ for } u = 1, \ldots, U,\; i = 1, \ldots, M_{\mathrm{T}},\; l = 1, \ldots, L_x \\ 0, & \text{otherwise,} \end{cases} \tag{3.16}$$

or equivalently

$$p(\mathbf{X}|\mathbf{c}) \,=\, \prod_{u=1}^{U}\prod_{l=1}^{L_x} \mathrm{I}\big\{\mathbf{x}^{(u)}[l] = \chi(\mathbf{c}^{(u)}[l])\big\} \,=\, \prod_{u=1}^{U}\prod_{i=1}^{M_\mathrm{T}}\prod_{l=1}^{L_x} \mathrm{I}\big\{x_i^{(u)}[l] = \chi(\mathbf{c}_i^{(u)}[l])\big\}\,. \tag{3.17}$$

Finally, $p(\mathbf{r}|\mathbf{X})$ corresponds to the system channel given by (3.8)–(3.10). We will consider two cases.

1) Perfect channel state information at the receiver. For comparison purposes, it will be useful to consider a genie system with perfect CSI at the receiver, i.e., the receiver knows all channel matrices $\mathbf{H}_d^{(u)}[l]$. In this case, transmission of pilot symbols is not necessary; thus $K_p = 0$ and all sets $\mathcal{P}_i^{(u)}$ are empty. We then have

$$f(\mathbf{r}|\mathbf{X}) \,=\, \prod_{l=1}^{L_x} f(\mathbf{r}_d[l]\,|\,\mathbf{X}[l])\,. \tag{3.18}$$

From the system channel expression (3.9),

$$\mathbf{r}_d[l]\,|\,\mathbf{X}[l] \,\sim\, \mathcal{CN}\Bigg(\sum_{u=1}^{U}\mathbf{H}_d^{(u)}[l]\mathbf{x}^{(u)}[l], \sigma_w^2\mathbf{I}\Bigg)\,.$$

Inserting (3.18), (3.15), and (3.17) into (3.13), we obtain the following expression of the posterior pmf of $b_m^{(u)}$:

$$p(b_m^{(u)}|\mathbf{r}) \,\propto\, \sum_{\sim b_m^{(u)}}\prod_{l=1}^{L_x} f(\mathbf{r}_d[l]\,|\,\mathbf{X}[l]) \prod_{u'=1}^{U}\mathrm{I}\big\{\mathbf{c}^{(u')}\!=\mathcal{C}^{(u')}(\mathbf{b}^{(u')})\big\}\prod_{i=1}^{M_\mathrm{T}}\mathrm{I}\big\{x_i^{(u')}[l] = \chi(\mathbf{c}_i^{(u')}[l])\big\}\,. \tag{3.19}$$

2) Pilot-based channel estimation. Next, we consider the practically relevant case where the channel has to be estimated and, thus, pilot subcarriers are transmitted. We can write

$$f(\mathbf{r}|\mathbf{X}) \,=\, \int f(\mathbf{r}|\mathbf{X},\mathbf{H})\, f(\mathbf{H})\, d\mathbf{H}\,, \tag{3.20}$$

where $\mathbf{H}$ is a matrix combining the channels of all users. We will now find expressions for $f(\mathbf{H})$ and $f(\mathbf{r}|\mathbf{X},\mathbf{H})$. Assuming that the channels of different users and MIMO streams are independent, we have

$$f(\mathbf{H}) \,=\, \prod_{u=1}^{U}\prod_{i=1}^{M_\mathrm{T}}\prod_{j=1}^{M_\mathrm{R}} f(\mathbf{h}_{j,i}^{(u)})\,, \tag{3.21}$$

where

$$f(\mathbf{h}_{j,i}^{(u)}) = \int \delta(\mathbf{h}_{j,i}^{(u)} - \mathbf{F}\tilde{\mathbf{h}}_{j,i}^{(u)})\, f(\tilde{\mathbf{h}}_{j,i}^{(u)})\, d\tilde{\mathbf{h}}_{j,i}^{(u)} \tag{3.22}$$

according to (3.7). For $f(\mathbf{r}|\mathbf{X},\mathbf{H})$, we obtain

$$f(\mathbf{r}|\mathbf{X},\mathbf{H}) \,=\, \prod_{l=1}^{L_x} f(\mathbf{r}_d[l]|\mathbf{H}_d[l],\mathbf{X}[l]) \prod_{u=1}^{U}\prod_{i=1}^{M_\mathrm{T}}\prod_{k\in\mathcal{P}_i^{(u)}} f(\mathbf{r}_p[k]|\mathbf{h}_i^{(u)}[k])\,, \tag{3.23}$$

with $\mathbf{H}_d[l] \triangleq \big(\mathbf{H}_d^{(1)T}[l]\cdots\mathbf{H}_d^{(U)T}[l]\big)^T$. Expressions for $f(\mathbf{r}_d[l]|\mathbf{H}_d[l],\mathbf{X}[l])$ and $f(\mathbf{r}_p[k]|\mathbf{h}_i^{(u)}[k])$ are obtained from (3.9) and (3.10), respectively:

$$\mathbf{r}_d[l]\,|\,\mathbf{H}_d[l],\mathbf{X}[l] \,\sim\, \mathcal{CN}\Bigg(\sum_{u=1}^{U}\mathbf{H}_d^{(u)}[l]\mathbf{x}^{(u)}[l], \sigma_w^2\mathbf{I}\Bigg)\,,$$

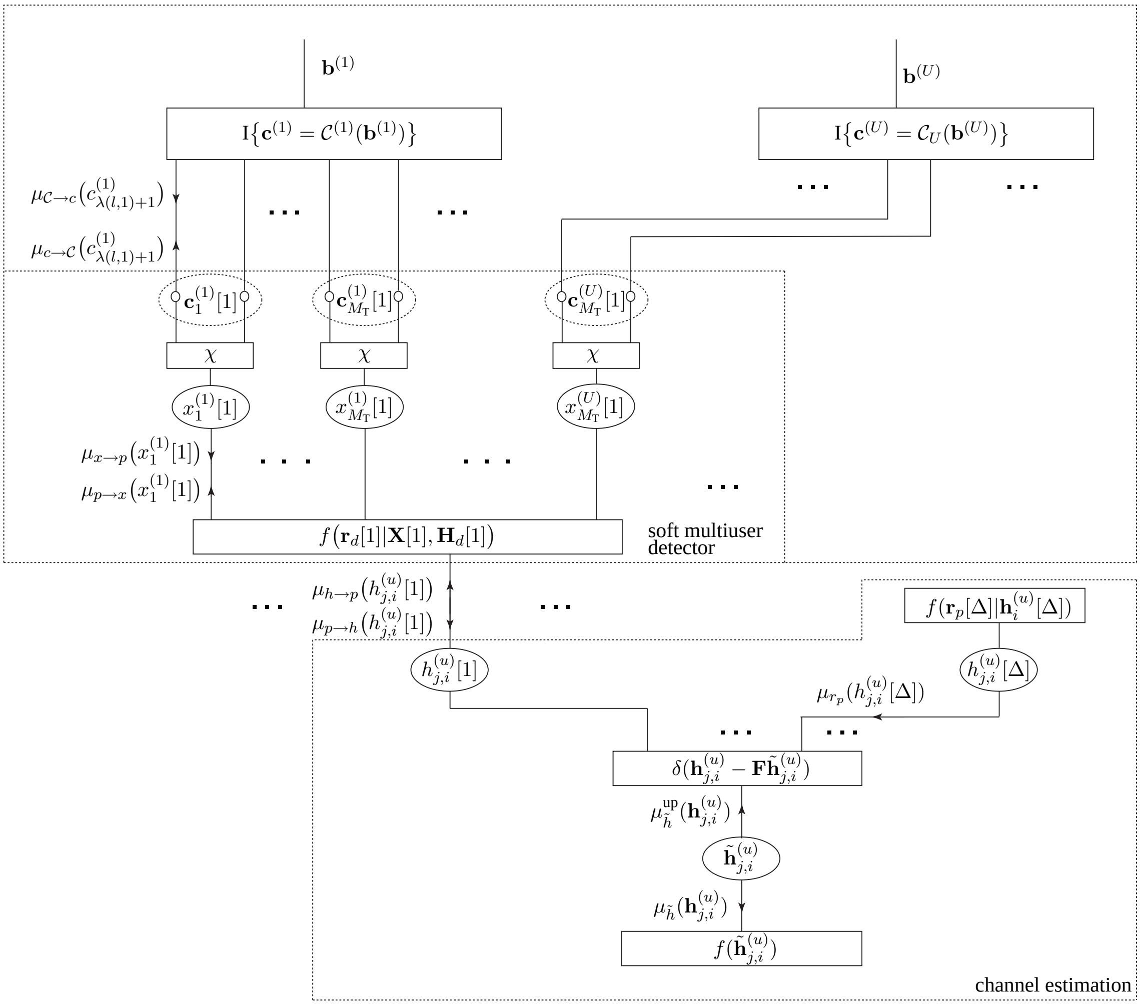

$\mathbf{b}^{(1)}$
$\mathbf{b}^{(U)}$
$\mathrm{I}\{\mathbf{c}^{(1)} = \mathcal{C}^{(1)}(\mathbf{b}^{(1)})\}$
$\mathrm{I}\{\mathbf{c}^{(U)} = \mathcal{C}_U(\mathbf{b}^{(U)})\}$
$\mu_{\mathcal{C}\to c}(c^{(1)}_{\lambda(l,1)+1})$
$\mu_{c\to\mathcal{C}}(c^{(1)}_{\lambda(l,1)+1})$
$\mathbf{c}_1^{(1)}[1]$
$\mathbf{c}_{M_\mathrm{T}}^{(1)}[1]$
$\mathbf{c}_{M_\mathrm{T}}^{(U)}[1]$
χ
χ
χ
$x_1^{(1)}[1]$
$x_{M_\mathrm{T}}^{(1)}[1]$
$x_{M_\mathrm{T}}^{(U)}[1]$
$\mu_{x\to p}(x_1^{(1)}[1])$
$\mu_{p\to x}(x_1^{(1)}[1])$
$f(\mathbf{r}_d[1]|\mathbf{X}[1], \mathbf{H}_d[1])$
soft multiuser detector
$\mu_{h\to p}(h^{(u)}_{j,i}[1])$
$\mu_{p\to h}(h^{(u)}_{j,i}[1])$
$f(\mathbf{r}_p[\Delta]|\mathbf{h}_i^{(u)}[\Delta])$
$h^{(u)}_{j,i}[1]$
$h^{(u)}_{j,i}[\Delta]$
$\mu_{r_p}(h^{(u)}_{j,i}[\Delta])$
$\delta(\mathbf{h}^{(u)}_{j,i} - \mathbf{F}\tilde{\mathbf{h}}^{(u)}_{j,i})$
$\mu^{\mathrm{up}}_{\tilde{h}}(\mathbf{h}^{(u)}_{j,i})$
$\tilde{\mathbf{h}}^{(u)}_{j,i}$
$\mu_{\tilde{h}}(\mathbf{h}^{(u)}_{j,i})$
$f(\tilde{\mathbf{h}}^{(u)}_{j,i})$
channel estimation

Figure 3.3: Factor graph for a pilot-aided MIMO-OFDM-IDMA system with channel coding and higher-order modulation.

$$\mathbf{r}_p[k]\,|\,\mathbf{h}_i^{(u)}[k] \;\sim\; \mathcal{CN}\big(p\,\mathbf{h}_i^{(u)}[k], \sigma_w^2\mathbf{I}\big)\,, \qquad k \in \mathcal{P}_i^{(u)}.$$

Inserting (3.20)–(3.23) into (3.13), the posterior pmf of $b_m^{(u)}$ is finally obtained as

$$p(b_m^{(u)}|\mathbf{r}) \;\propto \sum_{\sim b_m^{(u)}} \int f(\mathbf{X}, \mathbf{H}, \mathbf{r}|\mathbf{b})\, d\mathbf{H} \tag{3.24}$$

with

$$f(\mathbf{X}, \mathbf{H}, \mathbf{r}|\mathbf{b}) \;= \prod_{l=1}^{L_x} f(\mathbf{r}_d[l]|\mathbf{H}_d[l], \mathbf{X}[l]) \prod_{u=1}^{U} \prod_{i=1}^{M_\mathrm{T}} \prod_{k\in\mathcal{P}_i^{(u)}} f(\mathbf{r}_p[k]|\mathbf{h}_i^{(u)}[k]) \prod_{j=1}^{M_\mathrm{R}} \int \delta(\mathbf{h}^{(u)}_{j,i} - \mathbf{F}\tilde{\mathbf{h}}^{(u)}_{j,i})\, p(\tilde{\mathbf{h}}^{(u)}_{j,i})\, d\tilde{\mathbf{h}}^{(u)}_{j,i}$$
$$\times \prod_{u'=1}^{U} \mathrm{I}\big\{\mathbf{c}^{(u')} = \mathcal{C}_{u'}(\mathbf{b}^{(u')})\big\} \prod_{i'=1}^{M_\mathrm{T}} \mathrm{I}\big\{x_{i'}^{(u')}[k] = \chi(\mathbf{c}_{i'}^{(u')}[k])\big\}\,.$$

The factorizations in (3.19) and (3.24) can be represented by means of the factor graph [4, 13, 61] shown in Fig. 3.3. The part in the dotted box corresponds to (3.19) (receiver with perfect CSI), whereas the complete factor graph corresponds to (3.24) (receiver with pilot-based channel estimation). The probability distributions $f(\mathbf{r}_d[l]|\mathbf{H}_d[l],\mathbf{X}[l])$, $f(\mathbf{r}_p[k]|\mathbf{h}_i^{(u)}[k])$, and $f(\tilde{\mathbf{h}}_{j,i}^{(u)})$, the function $\delta(\mathbf{h}_{j,k}^{(u)} - \mathbf{F}\tilde{\mathbf{h}}_{j,k}^{(u)})$ expressing a DFT, and the constraint functions of the mapper, $\mathrm{I}\{x_i^{(u)}[k] = \chi(\mathbf{c}_i^{(u)}[k])\}$, and of the encoder/interleaver, $\mathrm{I}\{\mathbf{c}^{(u)} = \mathcal{C}_u(\mathbf{b}^{(u)})\}$, correspond to *function nodes*, which are represented by boxes. The information bits, code bits, transmit symbols, and channel coefficients correspond to *variable nodes*, which are represented by ellipses. It should be noted that the factor graph contains cycles.

3.3.2 Messages

For a factor graph without cycles, marginals like (3.13) can be determined exactly and efficiently by means of the sum-product algorithm [4]. Using these marginals, the MAP bit decisions in (3.11) can be easily determined. For a factor graph with cycles as in Fig. 3.3, the sum-product algorithm can still be used but it generally becomes iterative, yields only approximate results, and requires appropriate message scheduling.

We will next calculate the messages to be propagated along the edges of our factor graph according to the update rules of the sum-product algorithm [4]. The following discussion constantly refers to Fig. 3.3 regarding the notation used for the various messages and the position of the messages in the factor graph.

Because the code bit variable nodes $c_{\lambda(l,i)+j}^{(u)}$ and the transmit symbol variable nodes $x_i^{(u)}[l]$ are connected to only two neighboring function nodes, they are just transfer nodes passing the messages from one neighboring function node to the other. Thus, we only have to consider the message updates for the function nodes.

For the code function nodes $\mathrm{I}\{\mathbf{c}^{(u)} = \mathcal{C}^{(u)}(\mathbf{b}^{(u)})\}$, the sum-product algorithm amounts to the BCJR algorithm for soft-decoding the convolutional code [4, 8], and to a summation of the *a priori* log-likelihood ratios (LLRs) of successive bits (after interleaving) for soft-decoding the repetition code. The LLRs produced by the overall soft channel decoder are the sum of extrinsic LLRs and prior LLRs [52]. The extrinsic LLRs, denoted as $\xi_p^{(u)}$, correspond to messages (beliefs) [4, 13, 61]

$$\mu_{\mathcal{C}\to c}(c_q^{(u)}) \;=\; \frac{\exp(\xi_q^{(u)} c_q^{(u)})}{1+\exp(\xi_q^{(u)})}\,, \qquad c_q^{(u)} \in \{0,1\}\,. \tag{3.25}$$

These messages leave the code function node $\mathrm{I}\{\mathbf{c}^{(u)} = \mathcal{C}^{(u)}(\mathbf{b}^{(u)})\}$ and are propagated to the code variable nodes $\mathbf{c}_i^{(u)}[n]$ and further to the modulator function nodes χ.

Again invoking the sum-product algorithm, the message $\mu_{x\to p}(x_i^{(u)}[n])$ passed from the modulator function node χ to the variable node $x_i^{(u)}[n]$ and further to the channel function node $p(\mathbf{r}_d[l]|\mathbf{H}_d[l],\mathbf{X}[l])$ is obtained from the messages $\mu_{\mathcal{C}\to c}(c_{\lambda(l,i)+\nu}^{(u)})$ as

$$\mu_{x\to p}(x_i^{(u)}[l]) \;=\; \sum_{\mathbf{c}_i^{(u)}[l]} \mathrm{I}\{x_i^{(u)}[l] = \chi(\mathbf{c}_i^{(u)}[l])\} \prod_{\nu=1}^{B} \mu_{\mathcal{C}\to c}(c_{\lambda(l,i)+\nu}^{(u)}) \;=\; \bar{\mu}_{\mathcal{C}\to c}(\mathbf{c}_i^{(u)}[l])\big|_{\mathbf{c}_i^{(u)}[l]=\chi^{-1}(x_i^{(u)}[l])}\,, \tag{3.26}$$

where $\sum_{\mathbf{c}_i^{(u)}[l]}$ denotes summation over all the 2^B symbol labels and $\bar{\mu}_{\mathcal{C}\to c}(\mathbf{c}_i^{(u)}[l]) \triangleq \prod_{\nu=1}^{B} \mu_{\mathcal{C}\to c}(c_{\lambda(l,i)+\nu}^{(u)})$.

The message $\mu_{p\to x}\big(x_i^{(u)}[l]\big)$ passed from the channel function node $f(\mathbf{r}_d[l]|\mathbf{H}_d[l],\mathbf{X}[l])$ to the variable node

$x_i^{(u)}[l]$ and further to the modulator function node χ is obtained as follows. For a receiver with perfect CSI,

$$\mu_{p\to x}\big(x_i^{(u)}[l]\big) \;=\; \sum_{\sim x_i^{(u)}[l]} f(\mathbf{r}_d[l]\,|\mathbf{H}_d[l],\mathbf{X}[l]) \prod_{(i',u')\neq(i,u)} \mu_{x\to p}\big(x_{i'}^{(u')}[l]\big)\,, \tag{3.27}$$

and for a receiver with pilot-based channel estimation,

$$\mu_{p\to x}\big(x_i^{(u)}[l]\big) \;=\; \sum_{\sim x_i^{(u)}[l]} \int f(\mathbf{r}_d[l]\,|\mathbf{H}_d[l],\mathbf{X}[l]) \prod_{i',u'} \mu_{h\to p}(\mathbf{h}_{d,i'}^{(u')}[l]) \prod_{(i'',u'')\neq(i,u)} \mu_{x\to p}\big(x_{i''}^{(u'')}[l]\big)\, d\mathbf{H}_d[l]\,. \tag{3.28}$$

In (3.27) and (3.28), $\sum_{\sim x_i^{(u)}[l]}$ denotes summation with respect to all elements of $\mathbf{X}[l]$ except $x_i^{(u)}[l]$. The message $\mu_{h\to p}(\mathbf{h}_{d,i}^{(u)})$ involved in (3.28) will be determined in Section 3.4.

The message $\mu_{c\to\mathcal{C}}\big(c_{\lambda(l,i)+\nu}^{(u)}\big)$ passed from the modulator function node χ to the code variable node $\mathbf{c}_i^{(u)}[l]$ and further to the code function node $\mathrm{I}\big\{\mathbf{c}^{(u)} = \mathcal{C}^{(u)}(\mathbf{b}^{(u)})\big\}$ is obtained as

$$\begin{aligned}\mu_{c\to\mathcal{C}}\big(c_{\lambda(l,i)+\nu}^{(u)}\big) \;&=\; \sum_{\sim c_{\lambda(l,i)+\nu}^{(u)}} \mathrm{I}\big\{x_i^{(u)}[l] = \chi(\mathbf{c}_i^{(u)}[l])\big\}\, \mu_{p\to x}\big(x_i^{(u)}[l]\big) \prod_{\nu'\neq\nu} \mu_{\mathcal{C}\to c}\big(c_{\lambda(l,i)+\nu'}^{(m)}\big)\\ &=\; \sum_{\sim c_{\lambda(l,i)+\nu}^{(u)}} \mu_{p\to x}\big(\chi(\mathbf{c}_i^{(u)}[l])\big) \prod_{\nu'\neq\nu} \mu_{\mathcal{C}\to c}\big(c_{\lambda(l,i)+\nu'}^{(u)}\big)\,.\end{aligned} \tag{3.29}$$

From the message $\mu_{c\to\mathcal{C}}\big(c_q^{(u)}\big)$ the corresponding (extrinsic) LLR $\bar{\xi}_q^{(u)}$ can be obtained by [4]

$$\bar{\xi}_q^{(u)} = \log \frac{\mu_{c\to\mathcal{C}}\big(c_q^{(u)} = 1\big)}{\mu_{c\to\mathcal{C}}\big(c_q^{(u)} = 0\big)} \tag{3.30}$$

Combining (3.26) and (3.28) and inserting the result into (3.29) yields a message update that takes the code bit beliefs $\mu_{\mathcal{C}\to c}(c_q^{(u)})$ from the channel decoder and yields refined code bit beliefs $\mu_{c\to\mathcal{C}}(c_q^{(u)})$. Hence, these message updates taken together constitute a *soft-in/soft-out MIMO multiuser detector*.

Since (3.26) and (3.29) involve only one antenna of one user, the overall complexity of the sum-product algorithm is dominated by (3.28). Indeed, the sum in (3.28) involves $|\mathcal{S}|^{M_\mathrm{T}U-1}$ terms, so the complexity of calculating $\mu_{p\to x}\big(x_i^{(u)}[l]\big)$ is exponential in the number of transmit antennas M_T and in the number of users U. For example, $|\mathcal{S}|^{M_\mathrm{T}U-1} \approx 2.7\cdot 10^8$ for four users with two transmit antennas and 16-QAM modulation.

For the receiver with pilot-based channel estimation, the message $\mu_{p\to h}(\mathbf{h}_{d,i}^{(u)}[l])$ is obtained by means of the sum-product algorithm as

$$\mu_{p\to h}(\mathbf{h}_{d,i}^{(u)}[l]) \;=\; \sum_{\mathbf{X}[l]} \int f(\mathbf{r}_d[l]|\mathbf{H}_d[l],\mathbf{X}[l]) \prod_{(i',u')\neq(i,u)} \mu_{h\to p}(\mathbf{h}_{d,i'}^{(u')}[l]) \prod_{i'',u''} \mu_{x\to p}(x_{i''}^{(u'')})\, d\!\sim\!\mathbf{h}_{d,i}^{(u)}\,. \tag{3.31}$$

Here, $\mathbf{h}_{d,i}^{(u)}[l]$ denotes the ith column of $\mathbf{H}_d^{(u)}[l]$, and $\int\ldots d\sim\mathbf{h}_{d,i}^{(u)}$ denotes integration with respect to all elements of $\mathbf{H}_d[l]$ except those contained in $\mathbf{h}_{d,i}^{(u)}$.

The message $\mu_{r_p}(h_{j,i}^{(u)}[k])$, $k\in\mathcal{P}_i^{(u)}$ from the pilot symbols is given by

$$\mu_{r_p}(h_{j,i}^{(u)}[k]) \;\propto\; \exp\bigg(-\frac{1}{\sigma_w^2}\big|r_{p,j}[k] - h_{j,i}^{(u)}[k]p\big|^2\bigg)\,, \quad k\in\mathcal{P}_i^{(u)}.$$

This can be rewritten as a Gaussian in $h_{j,i}^{(u)}[k]$:

$$\mu_{r_p}(h_{j,i}^{(u)}[k]) \propto \exp\left(-\frac{|h_{j,i}^{(u)}[k] - r_{p,j}[k]/p|^2}{\sigma_w^2/|p|^2}\right). \tag{3.32}$$

Using the messages (3.31), (3.32), and the DFT relation (3.7), the messages $\mu_{h\to p}(\mathbf{h}_{d,i}^{(u)}[l])$ involved in (3.28) can be obtained. This derivation is deferred to Section 3.4.

3.4 Low-Complexity Receiver

In this section, we discuss two modifications of our iterative receiver that yield a significant reduction of computational complexity. First, we use Gaussian approximations for certain messages, so that closed-form integration becomes possible. As a result, the complexity of the multiuser detector is linear in the number of users, instead of exponential. An additional complexity reduction is achieved by a selective message update scheme where only certain messages are updated in each iteration.

3.4.1 Gaussian Approximations

To simplify the calculation of $\mu_{p\to x}\big(x_i^{(u)}[l]\big)$ in (3.27) or (3.28), we approximate the beliefs $\mu_{x\to p}\big(x_i^{(u)}[l]\big)$ in (3.26) by Gaussian distributions. That is, we set[1]

$$\mu_{x\to p}\big(x_i^{(u)}[l]\big) \propto \exp\left(-\frac{|x_i^{(u)}[l] - m_i^{(u)}[l]|^2}{\sigma_i^{(u)2}[l]}\right), \tag{3.33}$$

where the means $m_i^{(u)}[l]$ and variances $\sigma_i^{(u)2}[l]$ are chosen equal to those of the true $\mu_{\mathcal{C}\to c}\big(x_i^{(u)}[l]\big)$. Using (3.26), we obtain

$$\begin{aligned} m_i^{(u)}[l] &= \sum_{x_i^{(u)}[l]} x_i^{(u)}[l]\, \bar{\mu}_{\mathcal{C}\to c}\big(\chi^{-1}(x_i^{(u)}[l])\big), \\ \sigma_i^{(m)2}[l] &= \sum_{x_i^{(u)}[l]} \big|x_i^{(u)}[l] - m_i^{(u)}[l]\big|^2\, \bar{\mu}_{\mathcal{C}\to c}\big(\chi^{-1}(x_i^{(u)}[l])\big). \end{aligned}$$

For BPSK symbols with $|x_i^{(u)}[l]| = 1$, we have $|m_i^{(u)}[l]|^2 + \sigma_i^{(m)2}[l] = 1$. We will now use (3.33) for an approximate calculation of $\mu_{p\to x}\big(x_i^{(u)}[l]\big)$.

Perfect CSI. For the genie-aided case with perfect CSI at the receiver, we obtain from (3.27)

$$\mu_{p\to x}(x_i^{(u)}[l]) \propto \exp\Big(-\big(\mathbf{r}_d[l]-\mathbf{h}_{d,i}^{(u)}[l]x_i^{(u)}[l]-\mathbf{m}_i^{(u)}[l]\big)^H \mathbf{C}_i^{(u)-1}[l]\,\big(\mathbf{r}_d[l]-\mathbf{h}_{d,i}^{(u)}[l]x_i^{(u)}[l]-\mathbf{m}_i^{(u)}[l]\big)\Big), \tag{3.34}$$

with mean interference vector

$$\mathbf{m}_i^{(u)}[l] = \sum_{(i',u')\neq(i,u)} m_{i'}^{(u')}[l]\,\mathbf{h}_{d,i'}^{(u')}[l]$$

and interference plus noise covariance matrix

[1] To keep the notation simple, we use the symbols $\mu_{\mathcal{C}\to c}\big(x_i^{(u)}[l]\big)$ etc. also for the approximate quantities.

$$\mathbf{C}_i^{(u)}[l] \,=\, \sigma^2\mathbf{I} + \sum_{(i',u')\neq(i,u)} \sigma_{i'}^{(u')2}[l]\,\mathbf{h}_{d,i'}^{(u')}[l]\,\mathbf{h}_{d,i'}^{(u')H}[l]\,.$$

Hence, the (exponentially complex) computation of (3.27) is replaced with the computation of (3.34). The overall complexity of computing $\mu_{p\to x}(x_i^{(u)}[l])$ can be shown to scale linearly with the number of users and cubically with the number of transmit antennas.

Pilot-based channel estimation. For the receiver performing pilot-based channel estimation, we will use a Gaussian model also for the beliefs $\mu_{h\to p}(\mathbf{h}_{d,i}^{(u)}[l])$ i.e.,

$$\mu_{h\to p}(\mathbf{h}_{d,i}^{(u)}[l]) \,\propto\, \exp\Big(-\big(\mathbf{h}_{d,i}^{(u)}[l]-\mathbf{m}_{h_{d,i}^{(u)}}[l]\big)^H\,\mathbf{C}^{-1}_{h_{d,i}^{(u)}}[l]\,\big(\mathbf{h}_{d,i}^{(u)}[l]-\mathbf{m}_{h_{d,i}^{(u)}}[l]\big)\Big)\,, \tag{3.35}$$

where the mean vector $\mathbf{m}_{h_{d,i}^{(u)}}[l]$ and the covariance matrix $\mathbf{C}_{h_{d,i}^{(u)}}[l]$ will be determined at the end of this subsection.

Assuming that $x_i^{(u)}[l]$ and $\mathbf{h}_{d,i}^{(u)}[l]$ are mutually independent and distributed according to (3.33) and (3.35), respectively, the message $\mu_{p\to x}(x_i^{(u)}[l])$ in (3.28) equals the conditional pdf $f(\mathbf{r}_d[l]|x_i^{(u)}[l])$. This fact and the use of a Gaussian approximation allow for an approximation of the message $\mu_{p\to x}(x_i^{(u)}[l])$. We use the Gaussian approximation

$$\mu_{p\to x}(x_i^{(u)}[l]) \,=\, f_{\mathrm{G}}(\mathbf{r}_d[l]|x_i^{(u)}[l]) \,\propto\, \exp\Big(-\big(\mathbf{r}_d[l]-\mathbf{m}_{r_{d,i}^{(u)}[l]}\big)^H\,\mathbf{C}^{-1}_{r_{d,i}^{(u)}}[l]\,\big(\mathbf{r}_d[l]-\mathbf{m}_{r_{d,i}^{(u)}[l]}\big)\Big)\,, \tag{3.36}$$

with mean

$$\mathbf{m}_{r_{d,i}^{(u)}}[l] \,=\, x_i^{(u)}[l]\,\mathbf{m}_{h_{d,i}^{(u)}}[l] + \sum_{(i',u')\neq(i,u)} m_{i'}^{(u')}[l]\,\mathbf{m}_{h_{d,i'}^{(u')}}[l]$$

and covariance matrix

$$\mathbf{C}_{r_{d,i}^{(u)}}[l] \,=\, |x_i^{(u)}[l]|^2\,\mathbf{C}_{h_{d,i}^{(u)}}[l] + \sum_{(i',u')\neq(i,u)} \Big[\sigma_{i'}^{(u')2}[l]\,\mathbf{m}_{h_{d,i'}^{(u')}}[l]\,\mathbf{m}^H_{h_{d,i'}^{(u')}}[l] + \big(|m_{i'}^{(u')}[l]|^2+\sigma_{i'}^{(u')2}[l]\big)\,\mathbf{C}_{h_{d,i'}^{(u')}}[l]\Big] \\ +\, \sigma_w^2\mathbf{I}\,.$$

As in (3.34), the overall complexity of calculating $\mu_{p\to x}(x_i^{(u)}[l])$ scales linearly with the number of users and cubically with the number of transmit antennas.

In a similar manner, the message $\mu_{p\to h}(\mathbf{h}_{d,i}^{(u)}[l])$ in (3.31) can be approximated as

$$\mu_{p\to h}(\mathbf{h}_{d,i}^{(u)}[l]) \,\propto\, \exp\Big(-\big(\mathbf{h}_{d,i}[l]-\mathbf{m}_{h_{d,i}^{(u)}}[l]\big)^H\,|m_i^{(u)}|^2\mathbf{C}^{-1}_{h_{d,i}^{(u)}}[l]\,\big(\mathbf{h}_{d,i}[l]-\mathbf{m}_{h_{d,i}^{(u)}}[l]\big)\Big)\,,$$

with

$$\mathbf{m}_{h_{d,i}^{(u)}}[l] \,=\, \frac{1}{m_i^{(u)}[l]}\Bigg[\mathbf{r}[l] + \sum_{(i',u')\neq(i,u)} m_{i'}^{(u')}[l]\,\mathbf{m}_{h_{d,i'}^{(u')}}[l]\Bigg]$$

and

$$\mathbf{C}_{h_{d,i}^{(u)}}[l] \,=\, \sigma_i^{(u)2}[l]\,\mathbf{h}_{d,i}^{(u)}[l]\,\mathbf{h}_{d,i}^{(u)H}[l] + \sum_{(i',u')\neq(i,u)} \Big[\sigma_{i'}^{(u')2}[l]\,\mathbf{m}_{h_{d,i'}^{(u')}}[l]\,\mathbf{m}^H_{h_{d,i'}^{(u')}}[l] \\ + \big(|m_{i'}^{(u')}[l]|^2+\sigma_{i'}^{(u')2}[l]\big)\,\mathbf{C}_{h_{d,i'}^{(u')}}[l]\Big] + \sigma_w^2\mathbf{I}\,. \tag{3.37}$$

To obtain a message that is Gaussian in $\mathbf{h}_{d,i}^{(u)}[l]$, we approximate $\mathbf{C}_{h_{d,i}^{(u)}}[l]$ by neglecting the first term in (3.37). This is motivated by the fact that, typically, $\sigma_i^{(u)2}[l] \to 0$ in the course of the iterations. In particular, for the special case of BPSK modulation, this approximation yields

$$\mathbf{C}_{h_{d,i}^{(u)}}[l] \,=\, \sum_{(i',u')\neq(i,u)} \Big[\sigma_{i'}^{(u')2}[l]\,\mathbf{m}_{h_{d,i'}^{(u')}}[l]\,\mathbf{m}_{h_{d,i'}^{(u')}}^{H}[l] + \mathbf{C}_{h_{d,i'}^{(u')}}[l]\Big] \,+\, \sigma_w^2 \mathbf{I}\,.$$

Refinement of channel estimation. We next explain the calculation of the messages $\mu_{h\to p}(\mathbf{h}_{d,i}^{(u)}[l])$ from the messages $\mu_{p\to h}(\mathbf{h}_{d,i}^{(u)}[l])$. In the following, we consider all subcarriers (i.e., both data and pilot subcarriers) and define $\bar{\mu}_{p\to h}(h_{j,i}^{(u)}[k])$ as $\mu_{p\to h}(h_{j,i}^{(u)}[l])$ indexed with the true subcarrier position k. Similarly, we define $[\bar{\mathbf{m}}_{h_{j,i}^{(u)}}]_k$ and $[\bar{\mathbf{C}}_{h_{j,i}^{(u)}}]_{k,k}$ as the mean $[\mathbf{m}_{h_i^{(u)}}[l]]_j$ and variance $[\mathbf{C}_{h_i^{(u)}}[l]]_{j,j}$ indexed with the true subcarrier position, respectively.

Consider the vector $\mathbf{h}_{j,i}^{(u)}$ for given u, i, and j. From the factor graph in Fig. 3.3, it can be seen that the edges associated with the messages $\bar{\mu}_{p\to h}(h_{j,i}^{(u)}[k])$ and $\mu_{r_p}(h_{j,i}^{(u)}[k])$ are connected to the variable node $\mathbf{h}_{j,i}^{(u)}$. Let us collect the messages $\bar{\mu}_{p\to h}(h_{j,i}^{(u)}[k])$, $k \notin \mathcal{P}$ and $\mu_{r_p}(h_{j,i}^{(u)}[k])$, $k \in \mathcal{P}_i^{(u)}$ into a "vector message" $\mu_h(\mathbf{h}_{j,i}^{(u)})$. This message is Gaussian with some mean vector $\mathbf{m}_h$ and covariance matrix $\mathbf{C}_h$ (here and in the following, we omit the indices u, j, i for notational convenience). In the first several iterations, we use only the pilot symbols for channel estimation. We thus have

$$\mu_h(\mathbf{h}) \,=\, \prod_{k\in\mathcal{P}_i^{(u)}} \mu_{r_p}(h[k])\,,$$

whence

$$[\mathbf{m}_h]_k \,=\, \begin{cases} [\mathbf{r}_p[k]]_i/p\,, & k\in\mathcal{P}_i^{(u)} \\ 0\,, & \text{otherwise}\,, \end{cases} \qquad [\mathbf{C}_h]_{k,k} \,=\, \begin{cases} \sigma_w^2/|p|^2\,, & k\in\mathcal{P}_i^{(u)} \\ 0\,, & \text{otherwise}\,. \end{cases}$$

In later iterations, we also use the messages $\bar{\mu}_{p\to h}(h[k])$ for channel estimation, so

$$\mu_h(\mathbf{h}) \,=\, \prod_{k\in\mathcal{P}_i^{(u)}} \mu_{r_p}(h[k]) \prod_{k\notin\mathcal{P}} \bar{\mu}_{p\to h}(h[k])\,,$$

whence

$$[\mathbf{m}_h]_k \,=\, \begin{cases} [\mathbf{r}_p[k]]_i/p\,, & k\in\mathcal{P}_i^{(u)} \\ [\bar{\mathbf{m}}_{h_{j,i}^{(u)}}]_k\,, & k\notin\mathcal{P} \\ 0\,, & \text{otherwise}\,, \end{cases} \qquad [\mathbf{C}_h]_{k,k} \,=\, \begin{cases} \sigma_w^2/|p|^2\,, & k\in\mathcal{P}_i^{(u)} \\ [\bar{\mathbf{C}}_{h_{j,i}^{(u)}}]_{k,k}\,, & k\notin\mathcal{P} \\ 0\,, & \text{otherwise}\,. \end{cases}$$

Because $\tilde{\mathbf{h}} = \mathbf{F}^H\mathbf{h}$, the message $\mu_{\tilde{h}}(\tilde{\mathbf{h}})$ is Gaussian with mean $\mathbf{m}_{\tilde{h}} = \mathbf{F}^H\mathbf{m}_h$ and covariance $\mathbf{C}_{\tilde{h}} = \mathbf{F}^H\mathbf{C}_h\mathbf{F}$. The message $\mu_{\tilde{h}}(\tilde{\mathbf{h}})$ is multiplied by the *a priori* message $\mu_{f_{\tilde{h}}}(\tilde{\mathbf{h}})$, which is Gaussian with mean $\mathbf{m}_{\tilde{h}_p}$ and covariance $\mathbf{C}_{\tilde{h}_p}$ (cf. Section 3.2). The product message $\mu_{\tilde{h}}^{\text{up}}(\tilde{\mathbf{h}}) = \mu_{\tilde{h}}(\tilde{\mathbf{h}})\,\mu_{f_{\tilde{h}}}(\tilde{\mathbf{h}})$ is then again Gaussian [5], with mean and covariance to be denoted as $\mathbf{m}_{\tilde{h}}^{\text{up}}$ and $\mathbf{C}_{\tilde{h}}^{\text{up}}$, respectively. Because the time-domain channel $\tilde{\mathbf{h}}$ is assumed to have finite length L_{ch}, only the first L_{ch} elements of $\mathbf{m}_{\tilde{h}}^{\text{up}}$ (denoted by the vector $\mathbf{n}_{\tilde{h}}^{\text{up}}$) are nonzero. Similarly, only the top-left $L_{\text{ch}} \times L_{\text{ch}}$ submatrix of $\mathbf{C}_{\tilde{h}}^{\text{up}}$ (denoted as $\mathbf{D}_{\tilde{h}}^{\text{up}}$) has nonzero elements. Let $\mathbf{n}_{\tilde{h}}$ and $\mathbf{n}_{\tilde{h}_p}$

denote the vector of the first L_{ch} elements of $\mathbf{m}_{\tilde{h}}$ and $\mathbf{m}_{\tilde{h}_p}$, respectively, and let $\mathbf{D}_{\tilde{h}}$ and $\mathbf{D}_{\tilde{h}_p}$ denote the top-left $L_{\text{ch}} \times L_{\text{ch}}$ submatrix of $\mathbf{C}_{\tilde{h}}$ and $\mathbf{C}_{\tilde{h}_p}$, respectively. Then we have [5]

$$\mathbf{D}_{\tilde{h}}^{\text{up}} \,=\, \big(\mathbf{D}_{\tilde{h}}^{-1} + \mathbf{D}_{\tilde{h}_p}^{-1}\big)^{-1}, \qquad \mathbf{n}_{\tilde{h}}^{\text{up}} \,=\, \mathbf{D}_{\tilde{h}}^{\text{up}}\big(\mathbf{D}_{\tilde{h}}^{-1}\mathbf{n}_{\tilde{h}} + \mathbf{D}_{\tilde{h}_p}^{-1}\mathbf{n}_{\tilde{h}_p}\big)\,.$$

Because $\mathbf{h} = \mathbf{F}\tilde{\mathbf{h}}$, the message $\bar{\mu}_{h\to p}(\mathbf{h})$ is Gaussian with mean $\bar{\mathbf{m}}_h^{\text{up}} = \mathbf{F}\mathbf{m}_{\tilde{h}}^{\text{up}}$ and covariance $\bar{\mathbf{C}}_h^{\text{up}} = \mathbf{F}\mathbf{C}_{\tilde{h}}^{\text{up}}\mathbf{F}^H$ (cf. [5]). The mean $[\mathbf{m}_{h_{d,i}^{(u)}}[l]]_j$ and variance $[\mathbf{C}_{h_{d,i}^{(u)}}[l]]_{j,j}$ in (3.35) equal the element of $\mathbf{m}_h^{\text{up}}$ and the diagonal element of $\mathbf{C}_h^{\text{up}}$ at the corresponding data subcarrier position, respectively.

3.4.2 Selective Message Update

Since for higher-order symbol alphabets the calculation (update) of the messages $\mu_{c\to\mathcal{C}}(c_q^{(u)})$ in (3.29) is rather costly, we propose a scheme that yields a reduction of the number of updates performed. This reduction is achieved by calculating updated beliefs only for code bits with poor reliability. The reliability of a code bit $c_q^{(u)}$ is measured by its posterior LLR, which is given by

$$\tilde{\xi}_q^{(u)} \,=\, \log \frac{\mu_{\mathcal{C}\to c}(c_q^{(u)}\!=0)\,\mu_{c\to\mathcal{C}}(c_q^{(u)}\!=0)}{\mu_{\mathcal{C}\to c}(c_q^{(u)}\!=1)\,\mu_{c\to\mathcal{C}}(c_q^{(u)}\!=1)}\,. \tag{3.38}$$

If $|\tilde{\xi}_q^{(u)}|$ exceeds a prescribed threshold, the corresponding message $\mu_{c\to\mathcal{C}}(c_q^{(u)})$ is not updated, i.e., the value from the previous iteration is reused. In the course of the iterations of the sum-product algorithm, the code bit reliabilities improve and hence fewer and fewer message updates have to be performed. We note that this *selective message update* can be viewed as a specific scheduling [61] of the sum-product algorithm that dynamically adapts to the current bit reliabilities: Messages $\mu_{c\to\mathcal{C}}$ corresponding to unreliable decisions are updated earlier (and more often) than messages corresponding to reliable decisions.

The choice of the threshold affects both the number of message updates that have to be carried out and the performance of the sum-product algorithm (convergence behavior and final bit error rate). Since the LLRs generally increase with the signal-to-noise ratio (SNR), the threshold has to be adapted to the SNR. The impact of the LLR threshold on the performance and complexity of the receiver will be studied experimentally in Section 3.6. Generally, our experimental results show that the increase of BER caused by the selective message update scheme is rather small, thereby supporting the use of (3.38) as reliability measure.

3.4.3 Overall Receiver Structure

The message-passing algorithm developed above, together with appropriate scheduling, can be interpreted as the iterative turbo receiver structure shown in Fig. 3.4. The “OFDM receiver” block removes the cyclic prefix and calculates the frequency-domain sequence $\mathbf{r}$ by means of a DFT. The dotted box labeled “soft multiuser detector” in Fig. 3.3 corresponds to the soft-in/soft-out MIMO multiuser detector in Fig. 3.4, which exchanges messages with the U function nodes of the code constraints $\mathrm{I}\big\{\mathbf{c}^{(u)} = \mathcal{C}^{(u)}(\mathbf{b}^{(u)})\big\}$ located in the upper part of Fig. 3.3). The code constraint function nodes correspond to the U parallel single-user soft-in/soft-out channel decoders shown in Fig. 3.4. The dotted box “soft multiuser detector” in Fig. 3.3 also exchanges messages with the dotted box denoted “channel estimation”, which corresponds to the channel estimation block in Fig. 3.4.

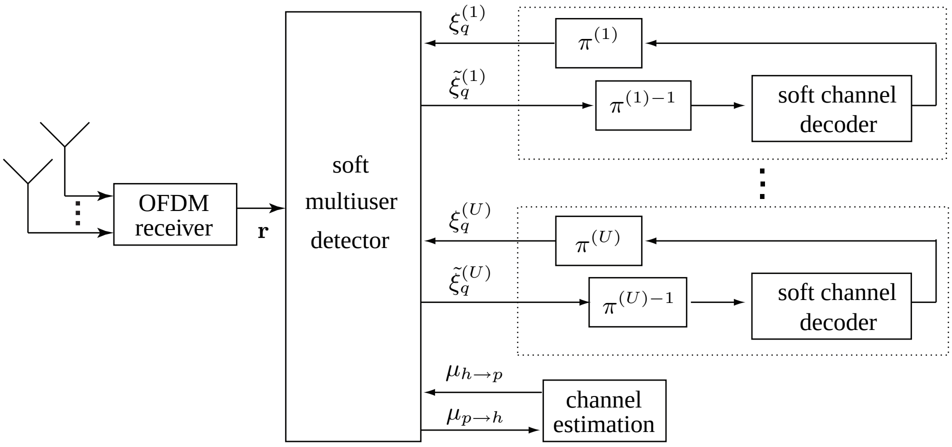


Figure 3.4: Structure of the MIMO-OFDM-IDMA receiver performing joint detection and channel estimation.

More specifically, the MIMO multiuser detector takes the code bit beliefs $\mu_{c\to\mathcal{C}}(c_q^{(u)})$ (they are obtained from the extrinsic LLRs $\xi_q^{(u)}$ by (3.25)) produced by the channel decoders as input and passes refined code bit beliefs $\mu_{c\to\mathcal{C}}(c_q^{(u)})$ (which are converted to LLRs $\bar{\xi}_q^{(u)}$ by means of (3.30)) back to the channel decoders. We distinguish between parallel and serial schedudling [61]: In case of parallel message scheduling the extrinsic LLRs $\xi_q^{(u)}$ for all users at the input of the multiuser detector are simultaneously updated by the channel decoders, and then used by the multiuser detector to calculate refined messages $\mu_{c\to\mathcal{C}}(c_q^{(u)})$ for all users concurrently. In a similar manner, the messages $\mu_{p\to h}$ obtained from the MIMO multiuser detector are used to calculate refined estimates $\mu_{h\to p}$ of the channel coefficients, which are used in the next iteration. One iteration consists of U decoding steps (i.e. calculating LLRs $\xi_q^{(u)}$), one multiuser detection step (i.e. calculating messages $\mu_{c\to\mathcal{C}}(c_q^{(u)})$), and one update of the messages $\mu_{h\to p}(h_{j,i}^{(u)}[l])$.

In case of serial message scheduling the user data is decoded sequentially. After decoding one user (yielding updated extrinsic LLRs $\xi_q^{(u)}$ of this user), the multiuser detector calculates refined messages $\mu_{c\to\mathcal{C}}(c_q^{(u)})$. These messages are then used by the channel deocder to decode the next user. After all U users have been decoded (and, correspondingly, U updates of the messages $\mu_{c\to\mathcal{C}}(c_q^{(u)})$ by the multiuser detector), one iteration is finished. Therefore one iteration consists of U decoding steps and U multiuser detection steps; it is therefore much more computationally complex than parallel scheduling.

When the message-passing algorithm is terminated after a predefined number of iterations, the signs of the *a posteriori* LLRs of the information bits calculated by the channel decoder (this calculation is not discussed here) provide the final bit decisions $\hat{b}_m^{(u)}$ approximating (3.11).

3.5 Performance Limits of IDMA

In Section 3.6 we will demonstrate the performance of the proposed low-complexity receiver derived in Section 3.4 by means of BER simulations. Being important from a system point of view, it was argued in [20] that this might be misleading as BER performance also depends on the channel code.

Therefore, we seek a code-independent performance assessment of the low-complexity receiver. To this end

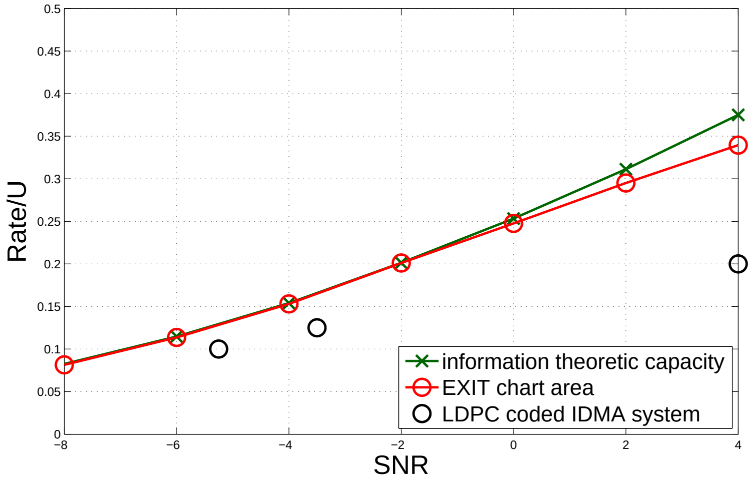


Figure 3.5: Information theoretic sum-rate of SISO-IDMA with $U = 4$ users and BPSK modulation compared with the area under the EXIT chart of the IDMA low-complexity multiuser detector and 3 operation points of an LDPC-coded IDMA system.

we compare the achievable rate of the IDMA receiver with the information theoretic capacity of the Gaussian multiple-access (MAC) channel given by (3.8) (or equivalently by (2.13)) restricted to finite symbol alphabet at its input. We consider only IDMA systems with perfect CSI at the receivers, as only in this case the information theoretic sum-rate is known.

The sum rate R_s of the MAC channel (3.8) is given by (2.14) (cf. [39])

$$R_s = \sum_{u=1}^{U} R_u = I(\mathbf{X};\mathbf{r})\,, \tag{3.39}$$

where R_u denotes the individual rate of the uth user and the mutual information [39] is given by

$$I(\mathbf{X};\mathbf{r}) = \frac{1}{2^{UM_\mathrm{T}B}} E_\mathbf{H}\left\{ \sum_{\mathbf{X}} \log \frac{2^{UM_\mathrm{T}B} f(\mathbf{r}|\mathbf{H},\mathbf{X})}{\sum_{\mathbf{X}'} f(\mathbf{r}|\mathbf{H},\mathbf{X}')} \right\}.$$

We assume independent user channel with i.i.d. Rayleigh fading; the expectation $E_\mathbf{H}$ has to be taken accordingly. For a symmetric MAC, the IDMA system considered here provides the same rate to every user, i.e. there is $R_u = \frac{1}{U} R_s$.

To obtain an approximation for the achievable rate of the low-complexity IDMA receiver presented in Section 3.4, we performed BER simulations, but used an LDPC code [49] for channel coding. Due to the strong performance of these codes, the waterfall region is close to the rate achievable with IDMA.

We first considered a SISO-IDMA system with $U = 4$ users and BPSK modulation and obtained sum-rate (normalized by the number of users) $\frac{1}{U} I(\mathbf{X};\mathbf{r})$ by means of Monte Carlo simulations. The results are shown in Fig. 3.5. We exchanged the serial concatenation of the convolutional code and the repetition code with a regular LDPC code[2] and performed BER simulations for code rates $R = 0.1,\ 0.125,\ 0.2$. The SNR values above which the BER dropped below 10^{-4} are also shown as operation points.

[2] The LDPC code was designed using the EPFL web-tool at `http://lthcwww.epfl.ch/research/ldpcopt`

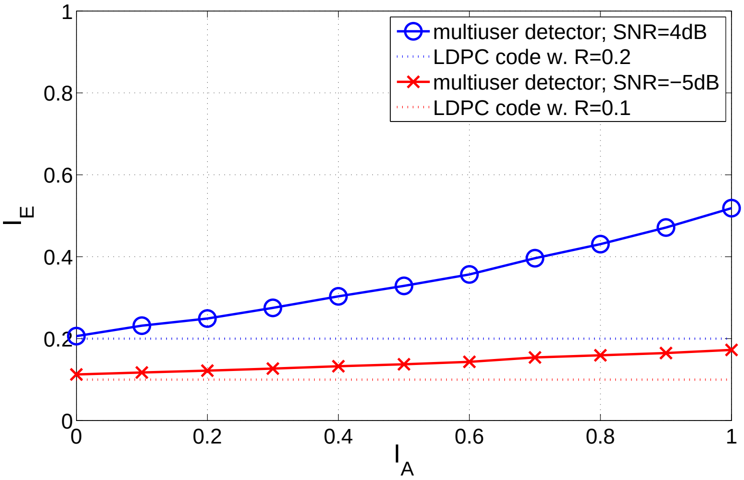


Figure 3.6: IDMA system with $U = 4$ users and BPSK modulation: EXIT charts of the low-complexity multiuser detector and of LDPC codes with different rates.

It can be seen that for code rates of $R = 0.1$ and $R = 0.125$ the IDMA system performs about 2 dB away from the information theoretic limit. For a code rate of $R = 0.2$ we observe a much larger gap of about 6 dB. This gap can partly be explained by the fact that the LDPC codes we used were optimized for AWGN channels and at higher SNR (which is equivalent to higher rates) there is a mismatch between the LDPC codes and the IDMA system [62].

We further investigated this mismatch by means of EXIT charts (see Chapter 2) of the low-complexity IDMA multiuser detector. The EXIT chart [42] is a plot of the mutual information $I_E(I_A) = I(c_q^{(u)}; \mu_{c\to\mathcal{C}}(c_q^{(u)}))$ versus $I_A = I(c_q^{(u)}; \mu_{\mathcal{C}\to c}(c_q^{(u)}))$. Fig. 3.6 shows EXIT charts of the low-complexity IDMA multiuser detector for 2 different SNR values, and of two LDPC codes with rates $R = 0.1$ and $R = 0.2$, respectively. At an SNR of -4 dB the EXIT chart of the multiuser detector matches the EXIT chart of the rate $R = 0.1$ LDPC code very well, and this explains the closeness of the corresponding operating point in Fig. 3.5 to the theoretical limit. At an SNR of 4 dB there is a significant mismatch of the multiuser EXIT chart and the EXIT chart of the rate $R = 0.2$ LDPC code, causing the large gap to the theoretical limit of this operating point.

A lot of experiments indicate that the area under the EXIT chart is approximately equal to the achievable rate. By perfectly matching the code to the EXIT chart of the multiuser detector [25, 63], the areas under the respective EXIT charts can be made equal, and therefore the area under the EXIT chart of the multiuser detector becomes an approximation of the achievable rate of the IDMA system. In Fig. 3.5 we additionally show the area under the EXIT chart of the multiuser detector. For an SNR below 0 dB the area is almost equal the information theoretic sum-rate, for higher SNR there is a small gap.

Finally, Fig. 3.7 shows the results for a 2×2 MIMO-IDMA system with $U = 2$ users and BPSK modulation. The used code rates for the BER simulations were $R = 0.125,\ 0.2,\ 0.25$ and all three operating points lie within 2 dB from the theoretical capacity limit. As in the SISO case, the area under the multiuser detector EXIT chart is very close to the information-theoretic limit for an SNR below 0 dB.

We observe that the low-complexity multiuser detector comes close to the absolute performance limit. This demonstrates the validity of the various assumptions and approximations involved in the derivation of

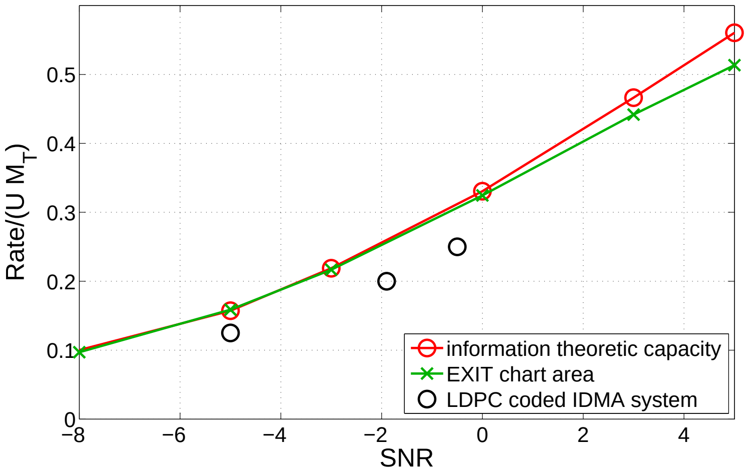


Figure 3.7: Sum-rate of 2×2 MIMO-IDMA with $U = 2$ users and BPSK modulation compared with the area under the EXIT chart of the IDMA low-complexity multiuser detector and 3 operation points of an LDPC-coded IDMA system.

the low-complexity multiuser detector. At higher SNR, the EXIT chart of the standard channel code (serial concatenation of the convolutional codes and a repetition code) is mismatched to the EXIT chart of the low-complexity multiuser detector. It is therefore important to employ optimized LDPC codes to achieve good performance.

3.6 Simulation Results

In this section, we present simulation results for the proposed MIMO-OFDM-IDMA receivers. We first consider receivers with perfect CSI and study the selective message update scheme. Then, we will investigate pilot-assisted receivers performing joint multiuser data detection and channel estimation. In all simulations, the channel code is a serial concatenation of a terminated rate-$1/2$ convolutional code with code polynomial given in octal notation by $[23\ 35]_8$ and a rate-$1/U$ repetition code, where U is the number of users. Thus, the overall code rate is $R = 1/(2U)$. The interleavers were generated randomly for each user.

3.6.1 Receivers with Perfect CSI

We first consider various versions of the genie-aided receiver with perfect CSI. We simulated a 2×2 MIMO-OFDM-IDMA system with $U = 2$ users, each transmitting $J = 512$ information bits, and 16-QAM modulation. The time-domain channel had $L_{\text{ch}} = 40$ independent taps, each with variance 1.

Basic low-complexity receiver. For the low-complexity receiver using the Gaussian message approximation (3.34), Fig. 3.8 shows the bit-error rate (BER) after 10 message-passing iterations, versus the SNR E_b/N_0. We can observe the typical turbo behavior, with an SNR of more than 8 dB required for convergence and a waterfall region above that SNR. For an SNR of $E_b/N_0 > 10$ dB, our receiver performs close to the single-user bound (i.e., only $U = 1$ user). We also show the BER of the receiver that calculates the exact message (3.29), i.e.,

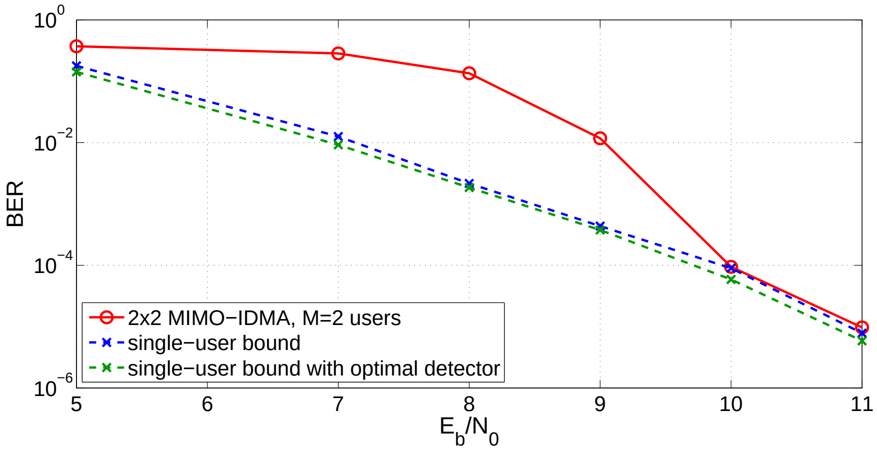


Figure 3.8: BER of different receivers versus SNR E_b/N_0 for a 2×2 MIMO system with 16-QAM modulation and $U=2$ users.

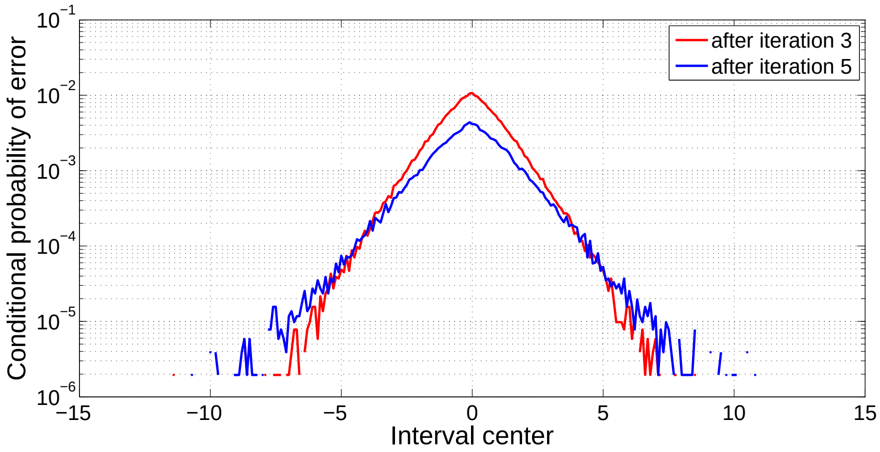


Figure 3.9: Conditional bit-error probability given that the posterior LLR lies in a small interval, versus the interval center, for a 2×2 MIMO system with 16-QAM modulation and $U=2$ users at an SNR of $9\,$dB.

without the Gaussian approximation, for $U=1$ user. It is seen that the proposed low-complexity receiver using (3.34) performs almost as well, which justifies the Gaussian approximation that led to (3.34).

Low-complexity receiver with selective message updates. Next, we present simulation results illustrating the selective message update scheme of Subsection 3.4.2. Based on the sign of the posterior LLR $\tilde{\xi}_q^{(u)}$ in (3.38), we can make a decision on the code bits $c_q^{(u)}$ and thereby compute the empirical probability of error $P\{\mathcal{E}\}$ for the code bits. In Fig. 3.9, we plot the conditional bit-error probability given that the LLR $\tilde{\xi}_q^{(u)}$ lies in an interval $\mathcal{I}_k$ of width 0.1, $P\{\mathcal{E}_q|\tilde{\xi}_q^{(u)}\in\mathcal{I}_k\}$, versus the center of $\mathcal{I}_k$. It can be seen that LLRs with a large absolute value yield a low probability of error. This experimental result provides a motivation for not updating the corresponding message $\mu_{c\to\mathcal{C}}(c_q^{(u)})$ when $|\tilde{\xi}_q^{(u)}|$ exceeds a certain threshold.

In Fig. 3.10, we study the BER versus complexity of the receiver using Gaussian message approximations, both with and without selective message updates, at an SNR of $E_b/N_0 = 11\,$dB. The complexity measure considered is the cumulative number of message updates divided by the block length, which can be interpreted

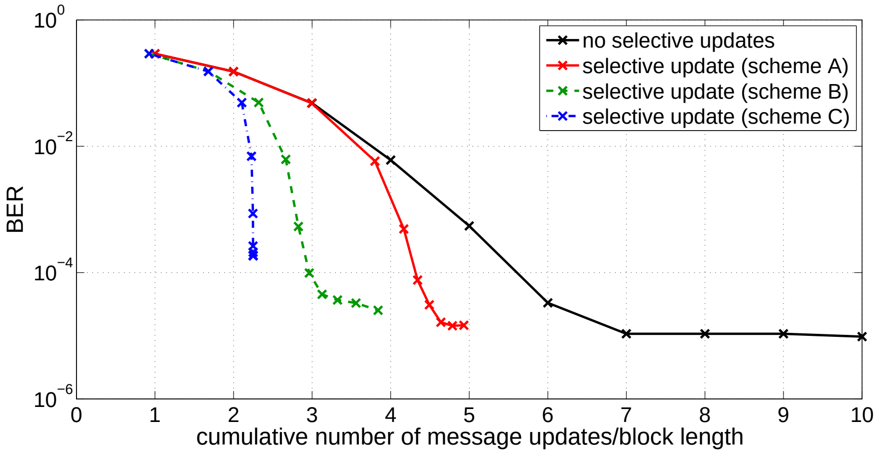


Figure 3.10: BER versus normalized cumulative number of message updates for receivers using different selective message update schemes and for the receiver without selective message updates, for a 2×2 MIMO system with 16-QAM modulation and $U = 2$ users at an SNR of 11 dB.

as the average number of message updates per code bit. We compare three different selective update schemes: schemes A and C use a constant LLR threshold of 30 and 5, respectively, which means that scheme C tends to perform fewer updates than scheme A. Scheme B uses an LLR threshold that decreases linearly from 30 (first iteration) to 5 (10th iteration); this is motivated by the fact that the LLRs tend to increase in the course of the iterations. From Fig. 3.10, we can draw the general conclusion that the selective message update strategy offers a very favorable performance–complexity tradeoff. Scheme A exhibits the quickest BER decrease with growing number of updates, but saturates at a BER slightly above 10^{-4} and a complexity of about 2.2 updates/block length; no updates are performed after that point. The last iterations reduce the BER only slightly but at the same time require only very few updates since most posterior LLR magnitudes are already larger than 5. The behavior of scheme C initially equals that observed without selective updates. Eventually, however, LLR thresholding sets in and the further BER decrease (down to below 10^{-5}) is achieved with significantly less complexity than without selective updates. The results of scheme B are intermediate between those of schemes A and C, with a quick initial BER decrease and saturation at a reasonably low BER. To achieve a target BER of 10^{-4}(or better), the method without selective updates requires six iterations with almost six updates/block length. Scheme B also requires only three updates/block length during six iterations, corresponding to computational savings of about 50%.

Low-complexity receiver with selective message updates and fixed complexity. In the selective message update scheme presented above, the actual number of updated messages depends on the LLR threshold and on system parameters like the SNR and the number of users. In practice, it may be more desirable to update a fixed number of messages in each iteration, so that a fixed complexity per iteration is obtained. This can be easily achieved by updating the messages $\mu_{c \to \mathcal{C}}(c_q^{(u)})$ corresponding to the pN posterior LLRs $\tilde{\xi}_q^{(u)}$ with smallest absolute value, where p $(0 \leq p \leq 1)$ controls the complexity (To obtain a high degree of flexibility, we allow the fixed number of message updates to change over the iterations, that is p changes over the iterations).

We consider three different schemes A, B, and C corresponding to a decreasing number of *total* message

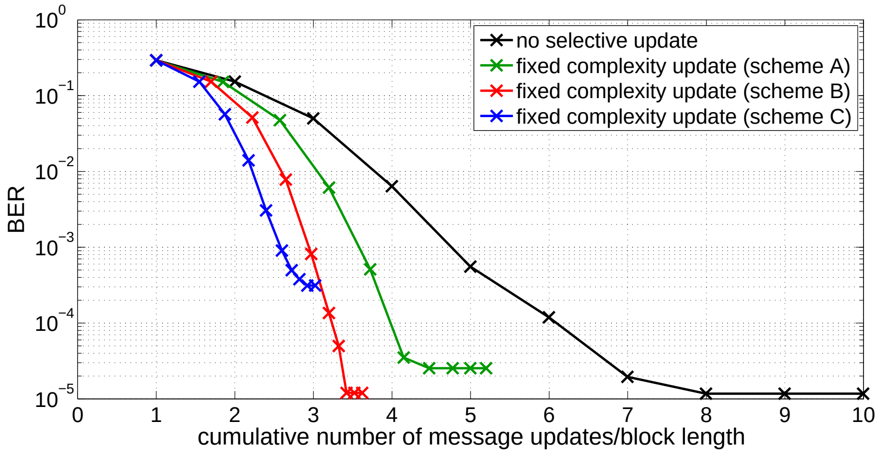


Figure 3.11: BER versus normalized cumulative number of message updates for receivers using different selective message update schemes with fixed complexity and for the receiver without selective message updates, for a 2×2 MIMO system with 16-QAM modulation and $U = 2$ users at an SNR of 11 dB.

updates, i.e., scheme A tends to update more messages than scheme B, etc. Fig. 3.11 shows the average BER versus complexity (cumulative number of updates/block length) for the three schemes and for the receiver without selective updates. The conclusions are essentially the same as before. For a target BER of 10^{-4}, computational savings of about 50% are possible (when scheme B is applied). The more aggressive thresholding scheme C attains a BER slightly above 10^{-4}. Schemes A and B as well as the receiver without selective updates attain smaller BERs with larger numbers of updates; however, after a certain point, additional updates no longer decrease the BER.

Next, we consider the fixed-complexity selective message update scheme for a system in which the two users have different channel gains. More specifically, the channel of the first user is stronger by 3 dB than that of the second user. We compare three different update schemes A, B, and C. In scheme A, the same number of updates per iteration is assigned to both users; in scheme B, the stronger user is assigned more updates than the weaker user, and in scheme C, the stronger user is assigned fewer updates than the weaker user. In schemes B and C, the total number of updates per iteration (summed over both users) is chosen to be the same as in scheme A. Fig. 3.12 shows the BER-versus-complexity curves for the three schemes and for the receiver without selective updates.

It can be seen that reducing the number of updates for the weaker user (as is done in scheme B) degrades performance, whereas schemes A and C yield almost the same performance. Compared to the receiver without selective updates, complexity savings on the order of 50% are possible.

We next compare the performance of different scheduling strategies. Fig. 3.13 shows the BER performance versus number of iterations of a 2×2 MIMO-IDMA system with 16QAM and $U = 2$ users, employing serial and parallel scheduling. It can be seen that the receiver employing serial scheduling converges faster (within 5 iterations), while the receiver employing parallel scheduling requires 10 iterations to reach the same BER performance. Considering the much higher complexity of the serial scheduling scheme in every iteration, it might still be favourable to use a receiver which employs parallel scheduling.

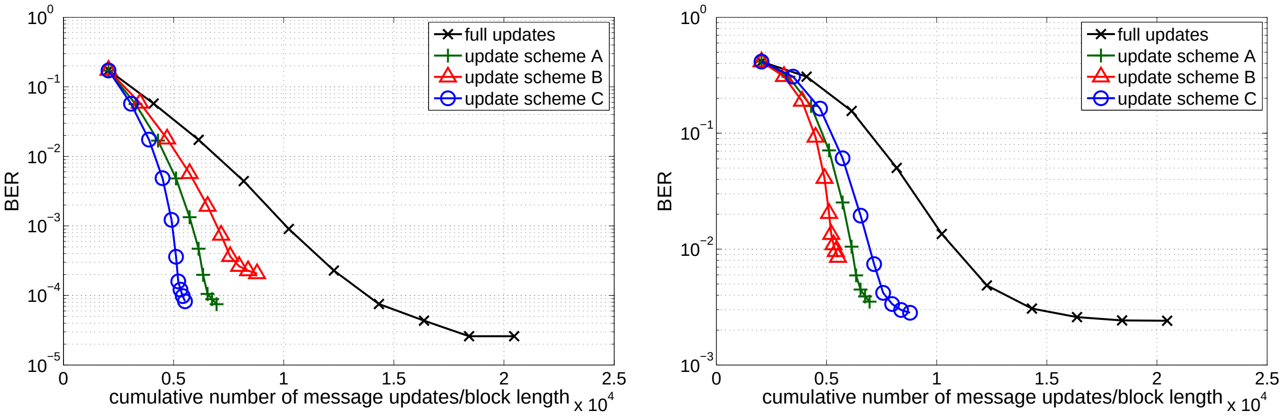


Figure 3.12: BER versus normalized cumulative number of message updates for receivers using different selective message update schemes with fixed user-dependent complexity, for a 2×2 MIMO system with 16-QAM modulation and $U = 2$ users at an SNR of 11 dB. The channel of user 1 is 3 dB stronger than the channel of user 2.

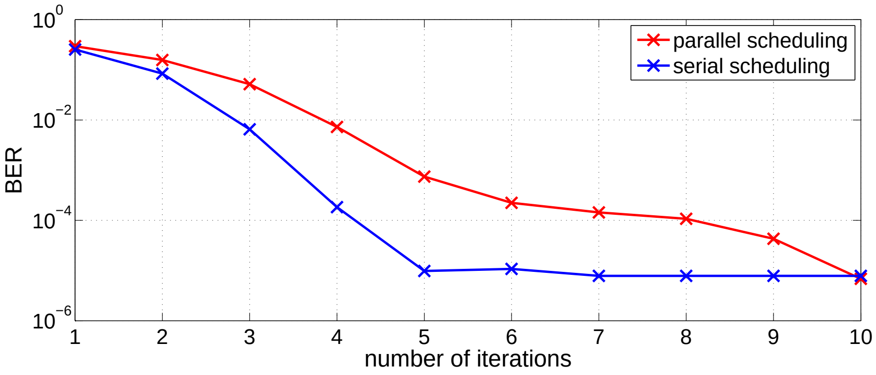


Figure 3.13: Performance comparison of scheduling strategies: BER versus cumulative number of message updates using parallel and serial scheduling for 2×2 MIMO-IDMA with 16QAM and $U = 2$ users at an SNR of 11 dB.

3.6.2 Receivers Performing Joint Channel Estimation and Data Detection

We now present simulation results for receivers performing joint data detection and pilot-aided channel estimation.

SISO system. First, we consider the single-input single-output (SISO) case, i.e., $M_\text{T} = M_\text{R} = 1$. We simulated a pilot-assisted SISO-OFDM-IDMA system with $U = 4$ users, each transmitting $J = 256$ information bits. The channel length was $L_\text{ch} = 40$ taps. The distance between the pilot blocks was chosen as $\Delta = 40$ subcarriers, and from (3.5) the number of pilot blocks is $K_p = 57$ and from (3.4) a total number of $K = 2276$ subcarriers was used. The modulation format was BPSK unless noted otherwise. The messages $\mu_{p \to h}$ were used for refining the channel estimate messages $\mu_{h \to p}$ only after the second iteration. Fig. 3.14 shows the average BER versus the SNR E_b/N_0 for the following receivers: (i) the proposed receiver using Gaussian message approximations; (ii) the genie-aided receiver with perfect CSI; (iii) a conventional receiver that separately estimates the channel

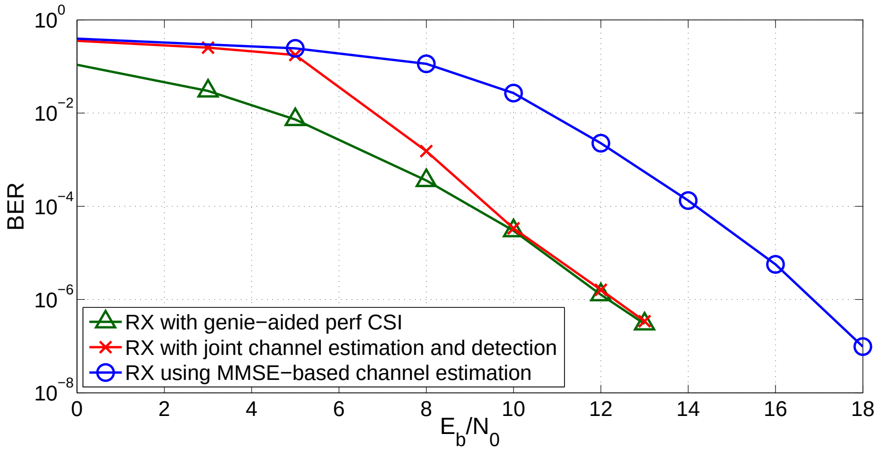


Figure 3.14: BER versus SNR E_b/N_0 for different iterative receivers, for a SISO system with BPSK modulation and $U = 4$ users.

coefficients by means of a pilot-based least-squares estimator and then uses the channel estimates for iterative data detection. The conventional receiver then uses the Gaussian approximations of (3.34). In all cases, 10 iterations were performed. It is seen that the receiver with integrated channel estimation gains about 5 dB of SNR compared to the conventional receiver and approaches the performance of the genie-aided receiver for SNRs larger than 10 dB.

For the proposed receiver using Gaussian message approximations, Fig. 3.15 depicts the mean square error (MSE) of the channel estimate versus the number of iterations, at an SNR of $E_b/N_0 = 13$ dB. As a reference, the channel estimation MSE of the conventional receiver with separate channel estimation is also shown (this is a horizontal line, as the MSE of the conventional receiver does not depend on the number of iterations). It is seen that the MSE of the proposed receiver starts decreasing significantly when the messages $\mu_{h\to p}$ are used (i.e., after the second iteration). After the seventh iteration, it is about 6 dB lower than the MSE of the conventional receiver. This decrease in MSE results in a better BER performance. This is demonstrated by Fig. 3.16, which shows the BER for the proposed and conventional receivers versus the number of iterations, again at $E_b/N_0 = 13$ dB. The BER of the conventional receiver saturates after four iterations, whereas the BER of the proposed receiver saturates after seven iterations to a value about three orders of magnitude smaller.

Next, we show results for a SISO-OFDM-IDMA system with $U = 2$ users and 16-QAM modulation. To avoid many short cycles in the factor graph (which would lead to poor performance), each user transmitted $J = 2048$ information bits. The pilot distance was $\Delta = 40$ subcarriers, resulting in $K_p = 54$ pilot blocks (cf. (3.5)) and a total number of $K = 2156$ subcarriers (c.f. (3.4)). The channel length was $L_{ch} = 20$. The receiver performed 15 iterations, and the messages $\mu_{p\to h}$ were used for refining the channel estimate messages $\mu_{h\to p}$ after the 8th iteration. Fig. 3.17 shows the average BER obtained with the proposed receiver, the conventional receiver, and the genie-aided receiver versus the SNR E_b/N_0. It is seen that the proposed receiver outperforms the conventional receiver for E_b/N_0 larger than about 22 dB, and it comes close to the genie-aided receiver for E_b/N_0 larger than about 25 dB.

We also considered the fixed-complexity selective message update scheme for this system. In every itera-

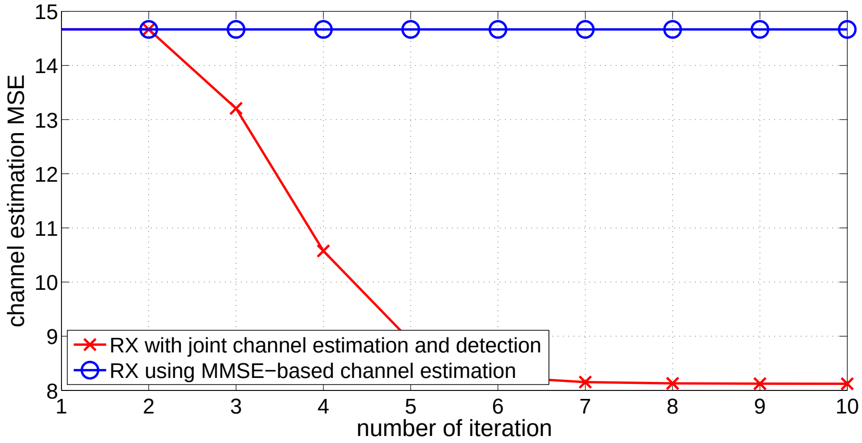


Figure 3.15: Channel estimation MSE versus number of iterations for different iterative receivers, for a SISO system with BPSK modulation and $U = 4$ users at an SNR of 13 dB.

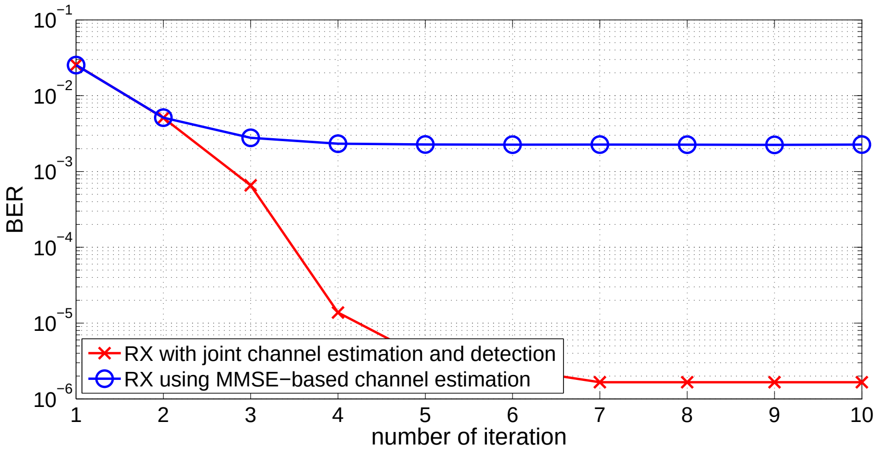


Figure 3.16: BER versus number of iterations for different iterative receivers, for a SISO system with BPSK modulation and $U = 4$ users at an SNR of 13 dB.

tion, the messages $\mu_{c \to \mathcal{C}}(c_q^{(u)})$ corresponding to the pN posterior LLRs $\tilde{\xi}_q^{(u)}$ with smallest absolute value are updated, whereas all messages $\mu_{p \to h}$ and $\mu_{h \to p}$ were updated.

Fig. 3.18 shows the average BER versus complexity (cumulative number of message updates/block length) for three different schemes and the receiver without selective message updates. The three different schemes A, B, and C correspond to decreasing number of message updates, i.e. scheme A updates more messages than scheme B, etc. Interestingly, the schemes A and B attain a lower BER than the receiver with full message updates, with about half the complexity. Scheme C performs slightly worse than the full complexity receiver but requires only about $1/3$ of the message updates. It seems that due to the system parameters, the many cycles of the factor graph decrease performance of the full complexity receiver, whereas the receivers with selective message updates (and therefore message updates) perform better.

In Fig. 3.17 we also plotted the BER performance of the selective message update scheme B, and it can be seen that it performs better than the full complexity receiver in spite of its lower complexity. Only in case of

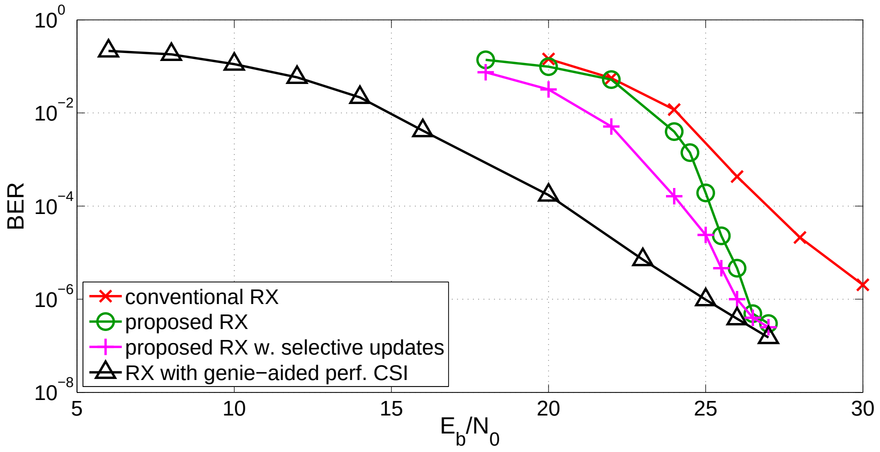


Figure 3.17: BER of different receivers versus SNR E_b/N_0 for a SISO system with 16-QAM modulation and $U = 2$ users.

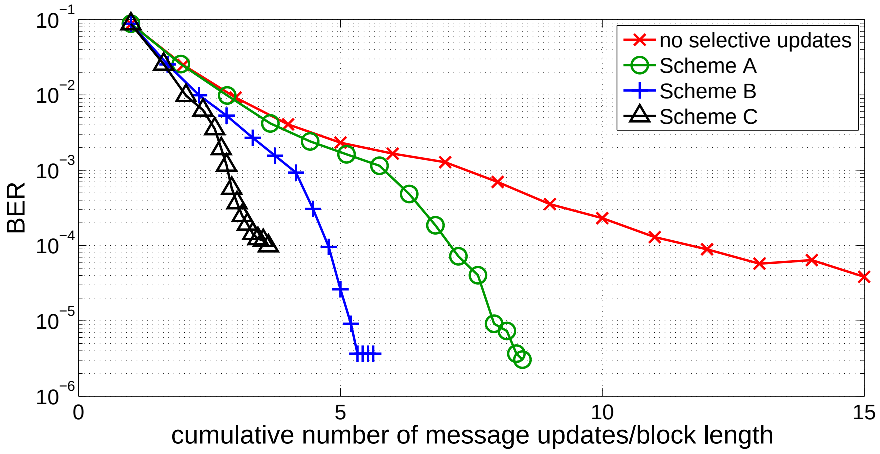


Figure 3.18: BER of different receivers versus SNR E_b/N_0 for a SISO system with 16-QAM modulation and $U = 2$ users.

high SNR (and convergence to the receiver with genie-aided CSI) both receivers perform the same.

MIMO system. Finally, we demonstrate the application of the proposed iterative receiver with integrated channel estimation to MIMO-OFDM-IDMA transmissions. We considered a 2×2 MIMO system with $U = 4$ users, each transmitting $J = 512$ information bits. The modulation format was BPSK. The distance between the pilot blocks was $\Delta = 30$ subcarriers. The number of pilot blocks was $K_p = 94$ and the total number of subcarriers $K = 2800$. The length of the time-domain channel was $L_{\text{ch}} = 30$ taps. Fig. 3.19 shows the BER versus the SNR. For an SNR larger than about 13 dB, the conventional receiver has an SNR gap of about 6 dB to the genie-aided receiver, and the proposed receiver comes within 0.5 dB of the genie-aided receiver.

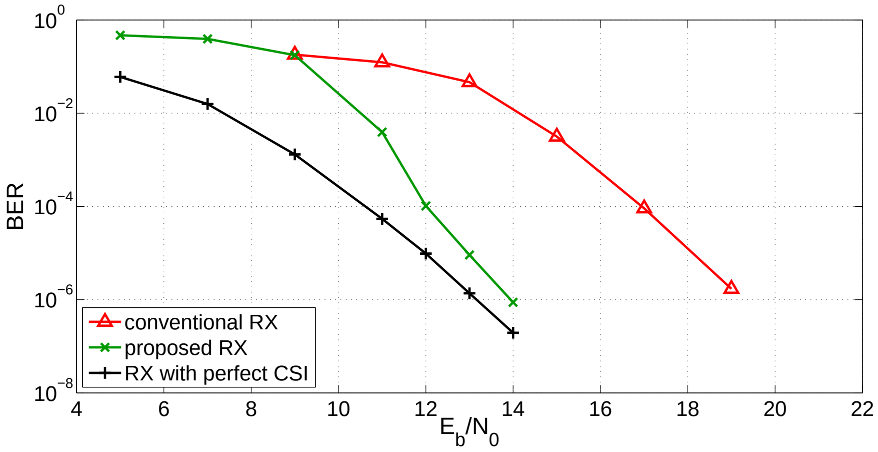


Figure 3.19: BER of different receivers versus SNR E_b/N_0 for a 2×2 MIMO system with BPSK modulation and $U = 4$ users.

4

MIMO-BICM with quantized LLRs

THE performance of bit-interleaved coded modulation (BICM) can be vastly improved by using soft detection, where the demodulator not only provides bit decisions, but also provides a reliability measure, usually a log-likelihood ratio (LLR), on these decisions. By appropriately using this reliability information in the channel decoder, significant performance gains can be achieved. In a variety of scenarios it is necessary or advantageous to quantize the LLRs before saving and/or further processing. This chapter studies how to best quantize LLRs, and which effects on performance LLR quantization has.

4.1 Introduction and Background

In a BICM system, a sequence of real-valued LLRs is calculated by the demodulator and needs to be stored, till a block of them can be deinterleaved and decoded by the channel decoder. Current error correcting channel codes are mostly block codes of large block length (LDPC codes operate with block length in the order of thousands to ten thousands of information bits). Therefore saving these real-valued LLRs for further processing requires a lot of memory. These memories require large chip sizes, which are unfavourable for reasons of cost, power consumption and reduced yields. Therefore, guidelines on how many bits are needed to efficiently store LLRs are required.

Quantization of soft information is also relevant in wireless (relay) networks [64]. Here, nodes can cooperate for transmission purposes or perform distributed turbo and network coding. This cooperation sometimes involves exchanging soft information between the nodes, which is usually represented by means of LLRs. It is undesirable to exchange LLRs by means of analog transmission, therefore the LLRs need to be quantized before they can be transmitted to cooperating nodes.

The system capacity of the equivalent BICM (modulation) channel between a code bit (at the input of the mapper) and its corresponding LLR (at the demodulator output) was studied in [10]. This work was extended and used for code-independent performance comparison of various sub-optimum soft-out demodulators for BICM in a *multiple-input multiple-output (MIMO)* context in [20]. Here we use the framework of the equivalent BICM modulation channel to propose different quantizer designs and compare their performance.

Optimal LLR quantization maximizing information rate for the special case of BPSK modulation over an AWGN channel was considered in [21], and in [65] we proposed a different quantizer design which allowed for simple implementation while only slightly degrading information rate. In this chapter we extend our proposed quantizer design to other modulation alphabets and system sizes and compare our approach with optimal LLR quantization.

This chapter is organized as follows: Section 4.2 presents the system model and Section 4.3 discusses LLR quantization based on an equivalent discrete channel. In Sections 4.3.1 and 4.4, we study the system capacity of SISO- and MIMO-BICM systems, respectively. The estimation of the quantizer parameters is addressed in Section 4.5 and BER results are provided in Section 4.6.

4.2 System Model

We consider a MIMO-BICM system with M_T transmit antennas and M_R receive antennas as was already described in Section 2.1.3. The system model is shown in Fig. 4.1; the only difference to the system model of Section 2.1.3 is the quantizer, whose operation will be explained below.

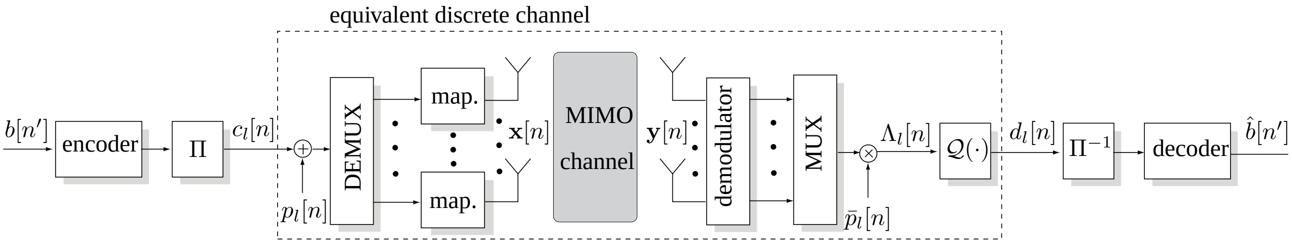


Figure 4.1: Block diagram of a MIMO-BICM system with LLR quantization.

4.2.1 Soft Demodulation

The MIMO-BICM system shown in Fig. 4.1 employs a max-log demodulator [38]. based on the received vector $\mathbf{y}$ it calculates LLRs Λ_l according to (2.7) (cf. Subsection 2.1.3)

$$\Lambda_l \approx \frac{1}{\sigma^2}\left[\min_{\mathbf{x}\in\mathcal{X}_l^0}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2 - \min_{\mathbf{x}\in\mathcal{X}_l^1}\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2\right]. \tag{4.1}$$

These LLRs (or approximate/quantized versions thereof) are de-scrambled by the sequence $\bar{p}_l[n] = 1-2p_l[n]$, de-interleaved and used by the channel decoder to obtain bit estimates $\hat{b}[n]$.

4.3 LLR Quantization

The LLRs in (4.1) can attain any real value, and we will next study how to quantize these LLRs. In practice the demodulator should directly deliver quantized LLRs (as an example of such a setup, we will study a soft-MMSE demodulator in Section 4.7).

The symmetric noise distribution and the use of the scrambler yield the symmetries $f_\Lambda(\xi) = f_\Lambda(-\xi)$ and $f_{\Lambda|c}(\xi|c{=}1) = f_{\Lambda|c}(-\xi|c{=}0)$ for the (un)conditional LLR distribution. Hence, knowledge of $f_{\Lambda|c}(\xi|c{=}1)$ is sufficient for characterizing Λ.

We consider a q-bit quantizer characterized by $K = 2^q$ bins $\mathcal{I}_k = [i_{k-1}, i_k]$, $k = 1, \ldots, K$. We use the convention $i_0 = -\infty$, $i_K = \infty$ and assume symmetric bins (this is motivated by the symmetry of the LLR distributions), with boundaries i_k sorted in ascending order. The quantizer $\mathcal{Q}(\cdot)$ maps the LLR Λ_l to a discrete LLR d_l according to

$$d_l = \mathcal{Q}(\Lambda_l) = \lambda_k \quad \text{if } \Lambda_l \in \mathcal{I}_k\,.$$

Here, $\lambda_k \in \mathcal{I}_k$ is the kth quantization level.

In the following, we consider the equivalent discrete channel with binary input $c \in \{0,1\}$ and K-ary output $d \in \{\lambda_1, \ldots, \lambda_K\}$ as shown in Fig. 4.2. Here, c and d are obtained by randomly picking a bit position $l = 1, \ldots, R_0$ according to a uniform distribution. This models a situation where the outer channel code is “blind” to the bit positions within the symbol labels. The crossover probabilities $p_{bk} = \Pr\{d = \lambda_k | c = b\} = \Pr\{\Lambda \in \mathcal{I}_k | c = b\}$ of this channel are given by

$$p_{bk} = \int_{\mathcal{I}_k} f_{\Lambda|c}(\xi|b)\, d\xi, \tag{4.2}$$

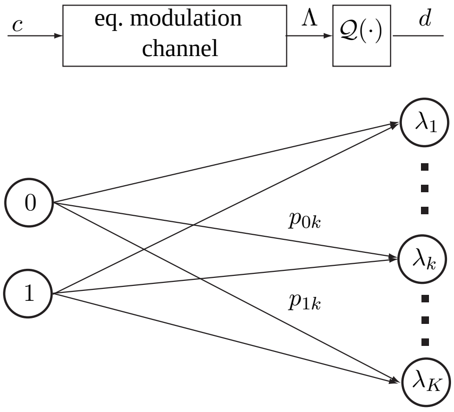


Figure 4.2: Block diagram and transition diagram of the equivalent discrete channel.

where $f_{\Lambda|c}(\xi|b)$ is the conditional probability density function (pdf) of the LLR Λ given that $c = b$ (averaged with respect to bit position l). Note that $\Pr\{d = \lambda_k\} = \Pr\{\Lambda \in \mathcal{I}_k\} = \frac{1}{2}(p_{0k} + p_{1k})$. The mutual information (capacity) $I = I(c\,; d)$ of this discrete channel is given by [39]

$$I = \frac{1}{2}\sum_{b=0}^{1}\sum_{k=1}^{K} p_{bk} \log_2 \frac{2p_{bk}}{p_{0k} + p_{1k}} \,. \tag{4.3}$$

If the LLR distribution $f_{\Lambda|c}(\xi|b)$ and hence the transition probabilities p_{bk} are averaged with respect to the statistics of the physical channel **H** (reflecting fast fading), the quantity I describes the ergodic rate achievable over the equivalent channel (cf. [40]). Otherwise (quasi-static fading), the transition probabilities p_{bk}, and thus the rate I, change with every realization of the channel matrix **H**. Here, the probability

$$p_{\text{out}}(r) = \Pr\{I \leq R\}\,, \quad 0 \leq R \leq R_0 \tag{4.4}$$

characterizes the rate (denoted R) versus outage trade-off [40].

The optimal quantizer maximizes the mutual information of the equivalent channel between the code bits c and the quantized LLRs d [21]. Let $\{i_k^{\text{opt}}\}_{k=1}^{K-1}$ denote the set of quantization bins of the optimal quantizer, we have

$$\{i_k^{\text{opt}}\}_{k=1}^{K-1} = \arg \max_{\{i_k\}_{k=1}^{K-1}} I(c; d)\,. \tag{4.5}$$

This optimization problem is infeasible to solve in practice, because the mutual information $I(c; d)$ depends on the crossover probabilities p_{bk} by (4.3), which by themselves depend on the quantization intervals (cf. (4.2)). Furthermore, the conditional probabilities $f_{\Lambda|c}(\xi|b)$ need to be known, so that the crossover probabilities p_{bk} can be calculated. In [21] the optimization was performed numerically for BPSK transmission over an AWGN channel, and in this work we will present results for more general channels, based on numerical optimization.

A much simpler (though suboptimal) approach to the quantizer design is the following: since $c - \Lambda - d$ is a Markov chain, the data processing inequality implies $I(c\,; d) \leq I(c\,; \Lambda)$. In order for $I(c\,; d)$ to be as close as possible to $I(c\,; \Lambda)$ (for fixed K), our proposed quantizer maximizes the mutual information $I(\Lambda; d)$, that is

$$\{i_k^{\star}\}_{k=1}^{K-1} = \arg \max_{\{i_k\}_{k=1}^{K-1}} I(\Lambda; d)\,.$$

With $H(\cdot)$ denoting entropy, it follows that $I(\Lambda;d) = H(d) - H(d|\Lambda)$ and $H(d|\Lambda) = 0$ because d is a deterministic function of Λ. $H(d)$ is maximized by a uniform distribution of d and therefore, the quantizer boundaries $i_k^\star$, $k = 1,\dots,K-1$, have to ensure that

$$\Pr\{d=\lambda_k\} = \frac{p_{0k}+p_{1k}}{2} = \frac{1}{K}, \quad k = 1,\dots,K\,. \tag{4.6}$$

Using the unconditional cumulative LLR distribution $F_\Lambda(\lambda) = \Pr\{\Lambda \le \lambda\} = \frac{1}{2}\int_{-\infty}^{\lambda}\big[f_{\Lambda|c}(\xi|c=0) + f_{\Lambda|c}(\xi|c=1)\big]\,d\xi$, the optimal boundaries can be obtained by finding the arguments for which $F_\Lambda(\lambda) = k/K$, i.e.,

$$i_k^\star = F_\Lambda^{-1}\Big(\frac{k}{K}\Big) \quad k = 1,\dots,K-1\,. \tag{4.7}$$

We note that for this approach the capacity of an equivalent modulation channel is notmaximized, only the quantizer intervals are chosen such that the quantizer output d represent Λ in an (information-theoretic) optimal manner.

For the capacity in (4.3) only the bins (boundaries) are relevant, i.e., the actual quantization levels λ_k do not influence the achievable rate. However, these values are important in order to provide the channel decoder (e.g., a belief propagation decoder) with correct reliability information [66]. In view of the equivalent discrete channel, we hence propose to choose the quantization levels as corresponding LLRs

$$\lambda_k^\star = \log\frac{\Pr\{c=1|d=\lambda_k\}}{\Pr\{c=0|d=\lambda_k\}} = \log\frac{p_{1k}}{p_{0k}}. \tag{4.8}$$

It can be shown that $\lambda_k^\star \in \mathcal{I}_k$.

4.3.1 SISO-BPSK over Rayleigh fading

We next study in more detail the case of a SISO system ($M_\mathrm{T} = M_\mathrm{R} = 1$) with BPSK modulation ($R_0 = 1\,$bpcu) in Rayleigh fading[1]. Here, the system model (2.1) becomes real-valued and simplifies to $y = hx + w$, with $h \sim \mathcal{N}(0,1)$, $w \sim \mathcal{N}(0,\sigma^2/2)$, and $x = 2c-1 \in \{-1,1\}$. Then, the LLR Λ can be calculated according to

$$\Lambda = \frac{hy}{\sigma^2} = \frac{1}{\sigma^2}h(hx+w). \tag{4.9}$$

4.3.2 Ergodic Capacity

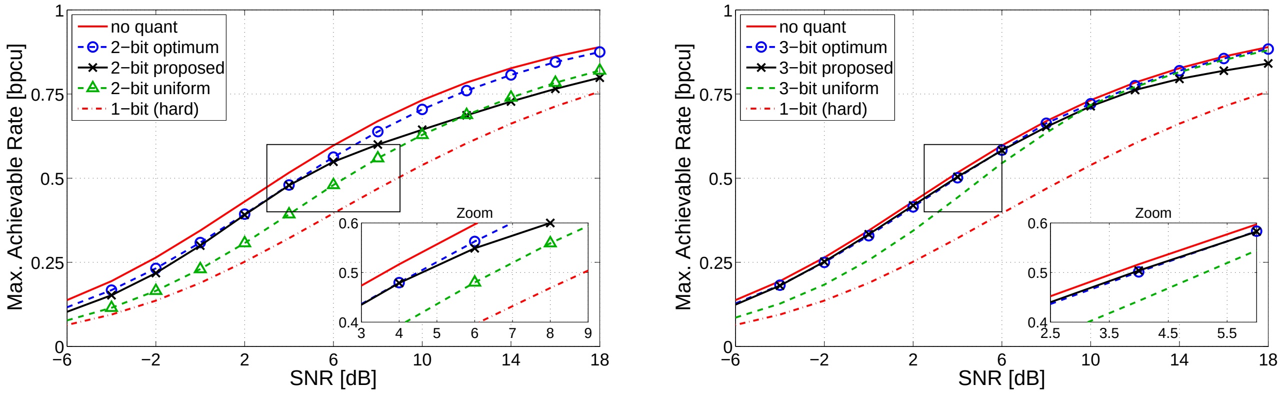
Conditioned on $c=x=1$, the LLR can be rewritten as $\Lambda = \frac{1}{\sigma^2}\,\mathbf{z}^T\mathbf{A}\,\mathbf{z}$, where $\mathbf{z} = \big(h\ \frac{\sqrt{2}w}{\sigma}\big)^T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ and

$$\mathbf{A} = \begin{pmatrix} 1 & \sigma/2 \\ \sigma/2 & 0 \end{pmatrix}.$$

Using the eigenvalue decomposition $\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^T$, with $\mathbf{U}$ orthogonal and $\boldsymbol{\Sigma} = \mathrm{diag}\{\sigma_1,\sigma_2\}$, where $\sigma_{1,2} = \frac{1\pm\sqrt{1+\sigma^2}}{2}$, we further obtain

$$\Lambda = \frac{1}{\sigma^2}\,\tilde{\mathbf{z}}^T\boldsymbol{\Sigma}\,\tilde{\mathbf{z}} = \frac{1}{\sigma^2}\Big[\sigma_1\tilde{z}_1^2 + \sigma_2\tilde{z}_2^2\Big]\,.$$

[1] The results in this section also apply to the inphase and quadrature phase of SISO systems with Gray-labeled QPSK and to the two layers of BPSK-modulated 2×2 MIMO systems.



(a)

(b)

Figure 4.3: Numerical capacity results for a SISO system with BPSK for various quantizer designs using (a) 2-bit quantization, and (b) 3-bit quantization (both with Gray labeling).

Here, $\tilde{\mathbf{z}} = \mathbf{U}^T\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ due to the orthogonality of $\mathbf{U}$. Thus, Λ is a linear combination of two independent chi-square random variables with one degree of freedom. The distribution $f_{\Lambda|c}(\xi|c = 1)$ can thus be shown to be given by (cf. [67])

$$f_{\Lambda|c}(\xi|c{=}1) = \frac{\sigma}{\pi} \exp\big(- \xi\sqrt{1+\sigma^2}\big)\, K_0(|\xi|), \tag{4.10}$$

where $K_0(\cdot)$ denotes the modified Bessel function of the second kind and order 0.

Using (4.10), one can determine the LLR distribution, the LLR quantization (cf. (4.5) and (4.7)), and the ergodic capacity of the equivalent channel given by (4.3).

Fig. 4.3(a),(b) show the ergodic capacity versus SNR for various quantizer designs with 2-bit and 3-bit quantization, respectively. We compare our proposed quantizer design (4.6) and the optimum quantizer (c.f. (4.5)). As a reference the ergodic capacity without quantization is also plotted (labeled ‘no quant’). Furthermore, we include a very simple quantizer design which uniformly quantizes the interval containing 70% of all LLRs (labeled ‘uniform’). Finally, the curve 1-bit is obtained by taking the sign of the LLRs (i.e. making a hard-decision).

It can be seen in Fig. 4.3(a) that for 2-bit quantization the optimum and the proposed quantizer perform almost the same up to rates of 0.5 bpcu. Compared to the non-quantized case the SNR loss at 0.5 bpcu is at most 1 dB, however, the performance advantage over 1 bit quantization is 4.4 dB. For higher rates, our proposed quantizer suffers from a significant performance degradation; at 0.75 bpcu, an SNR penalty of more than 3.5 dB can be observed compared to the optimum quantizer. Note that our proposed quantizer design performs significantly better than an uniform quantizer for low-to-medium rates, e.g., showing an SNR gap of 2 dB at 0.5 bpcu. Only at high rates (larger than 0.7 bpcu) uniform quantization starts to slightly outperform our proposed design. When spending an additional bit for quantization (i.e., 3 bits), both the optimum and the proposed quantizer closely approach the non-quantized performance curve for rates up to 0.75 bpcu. We can thus conclude that in this rate regime quantization with 3 bits is sufficient when using our proposed design. At higher rates our quantizer again suffers from a performance degradation; surprisingly, in this case 3-bit uniform quantization seems to be optimal in terms of performance.

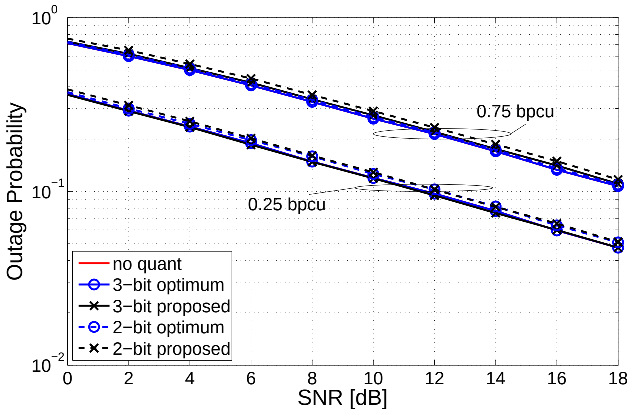


Figure 4.4: Quantizer performance for 2-bit and 3-bit quantization in quasi-static fading: outage probability versus SNR for $R = 0.25$ bpcu and $R = 0.75$ bpcu with (SISO, BPSK, Gray labeling).

4.3.3 Outage Probability

Additionally conditioning on the channel coefficient h, it follows straightforwardly that $\Lambda|c \sim \mathcal{N}(x\,\gamma, 2\gamma)$ with $\gamma = h^2/\sigma^2$. This allows to calculate the transition probabilities of the equivalent channel as

$$p_{bk} = Q\left(\frac{i_{k-1} - (2b-1)\gamma}{\sqrt{2\gamma}}\right) - Q\left(\frac{i_k - (2b-1)\gamma}{\sqrt{2\gamma}}\right).$$

The outage probability can thus be evaluated according to (4.4).

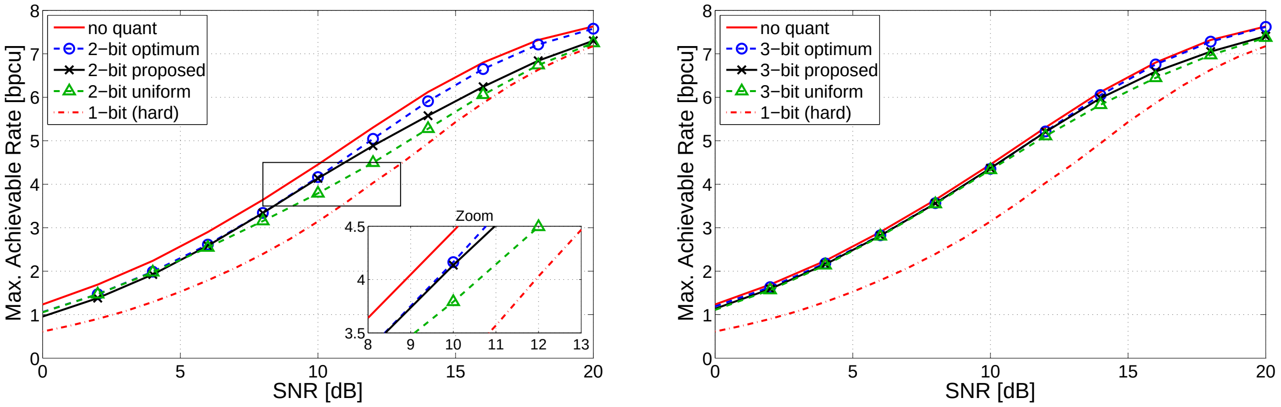
Numerical results of $p_{\text{out}}(r)$ versus SNR for quasi-static fading with rate $R = 0.25$ bpcu and with $R = 0.75$ bpcu are shown in Fig. 4.4. The difference between the proposed and the optimum quantizer is barely visible for both quantizer word-lengths and both target rates. Furthermore, the two quantizer designs closely approach the non-quantized case. Moreover, note that the asymptotic slopes of these curves show a diversity order of 1 and are thus independent of the quantizer word-lengths.

4.4 MIMO Systems and Higher-Order Modulation

In the following, we investigate LLR quantization for MIMO systems and higher-order constellations. Since in this case analytical expressions for the LLR distribution are hard to obtain in general, the remaining discussion is based exclusively on numerical results. For the capacity results in this section, we used empirical LLR distributions obtained from Monte-Carlo simulations (obtained with 10^5 fading realizations).

4.4.1 Ergodic Capacity

Fig. 4.5(a) shows the ergodic capacity (in bpcu) of a 2×2 MIMO system with a 16-QAM Gray mapped modulation alphabet (here, $R_0 = 8$) for the same quantizers as described in Subsection 4.3.2. As a reference we also plotted the capacity of the unquantized demodulator (labeled ’no quant’). In the following, in some of the plots we show insets that provide zooms of the capacity curves around a rate of $R_0/2$ bpcu.



(a) (b)

Figure 4.5: Numerical capacity results for 2×2 MIMO system with 16-QAM with (a) 2-bit quantization, and (b) 3-bit quantization (both with Gray labeling, $R_0 = 8$).

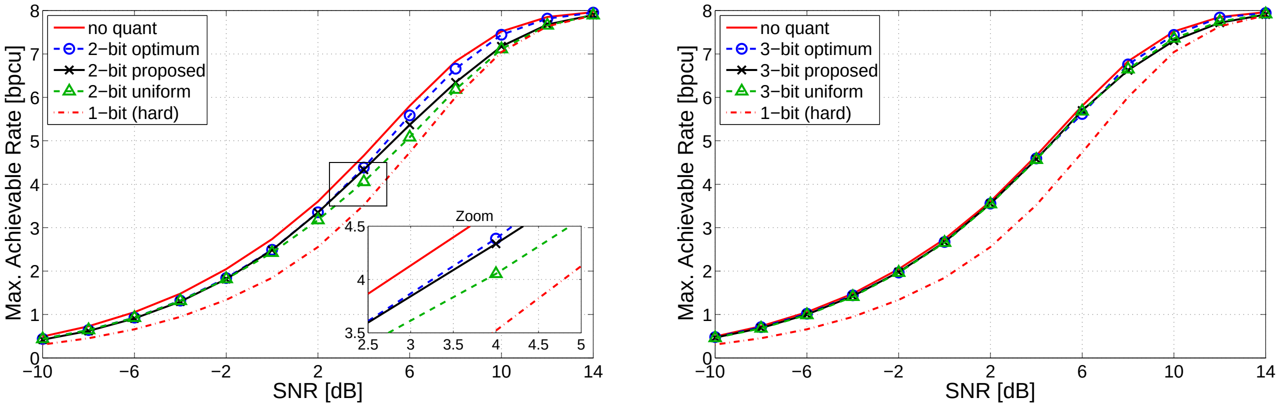
Up to code rates of $R = 1/2$ (this amounts to 4 bpcu), the 2-bit optimal and 2-bit equiprobable quantizer perform the same. Both exhibit an SNR of about 1 dB loss to the unquantized case at a rate of $R = 1/2$. At $R = 3/4$ (6 bpcu), the optimal 2-bit quantizer performs by more than 1 dB better than the equiprobable design. Up to rates of $R = 3/4$, there is almost no gap between the 3 bit optimal quantizer and the 3-bit equiprobable quantizer, both perform about 0.25 dB away from the unquantized demodulator. The 2-bit uniform quantizer performs as good as the optimal 2-bit quantizer only at low rates up to $R = 1/4$, at higher rates it performs worse, its performance comes close to the 1-bit (hard) demodulator. Finally, at a rate of $R = 1/2$, the 1-bit (hard) demodulator performs about 3 dB away from the unquantized demodulator.

In Fig. 4.6 we present our findings for the 4×4 MIMO case with 4QAM symbol alphabet. The results are very similar to the 2×2 16QAM case discussed before: For rates $R > 1/2$ there is a performance gap between the optimal and equiprobable 2 bit quantizers, which is about 1 dB at $R = 3/4$. At $R = 1/2$ the gap between 2 bit quantization (both optimal and equiprobable design) and the unquantized demodulator is about 0.75 dB. In case of 3 bit quantization, both designs perform identically, even in case of very high rates. The 1 bit (hard) demodulator has an SNR loss of about 3 dB compared to the unquantized case at a rate of $R = 1/2$.

4.4.2 Outage Capacity

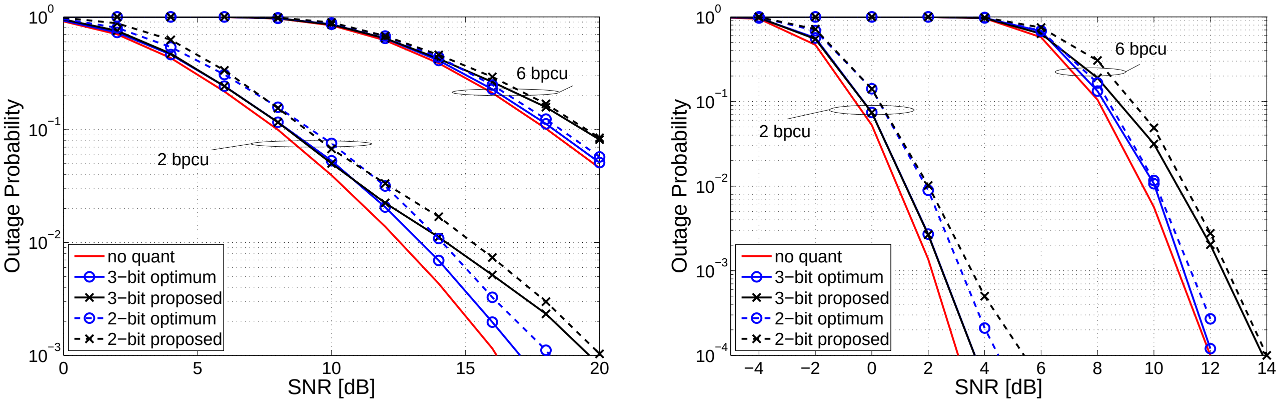
We next provide numerical results for the outage probability in (4.4) for both the 2×2 MIMO system with 16QAM symbol alphabet and the 4×4 MIMO system with 4QAM symbol alphabet. The outage probability P_{out} was obtained over 10^5 fading realizations and for each channel realization R in (4.3) was measured by transmitting a block of 10^4 symbol vectors. The quantization intervals were taken from a precomputed look-up table and have been computed for an i.i.d. fast Rayleigh fading channel using 10^5 fading realizations.

Fig. 4.7(a) shows our results for the 2×2 MIMO system with 16QAM symbol alphabet for target rates $R = 0.25$ and $R = 0.75$ (this amounts to 2bpcu and 6bpcu, respectively) for both the optimum and the proposed quantizer using a bit-width of 2 and 3 bits. At a target rate of $R = 0.25$, and for outage probabilities above $2 \cdot 10^{-1}$ no difference between the proposed and the optimal quantizer design can be seen; below this



(a) (b)

Figure 4.6: Numerical capacity results for 4×4 MIMO system with 4-QAM with (a) 2-bit quantization, and (b) 3-bit quantization (both with Gray labeling, $R_0 = 8$).



(a) (b)

Figure 4.7: Quantizer performance in quasi-static fading: outage probability versus SNR for $R = 2\,$bpcu and $R = 6\,$bpcu for (a) a 2×2 MIMO system with 16-QAM, and (b) a 4×4 MIMO system with 4-QAM (both with Gray labeling and $R_0 = 8$).

probability, there is a gap between the two quantizer designs (at an outage probability of 10^{-3}, this gap is about $2\,$dB for both 2 and 3-bit quantization). The optimal 2- and 3-bit quantizers respectively perform about $1\,$dB and $2\,$dB away from the unquantized case. At a target rate of $R = 0.75$, the gap between the optimal and the proposed quantizer appears at a much higher outage probability of $2 \cdot 10^{-1}$, and the gap between 2- and 3-bit quantization is much smaller.

Fig. 4.7(b) shows our results for the 4×4 MIMO system with 4QAM symbol alphabet for target rates $R = 0.25$ and $R = 0.75$ for the same quantizers as above. Here, the results are different: For the low target rate (corresponding to 2bpcu), there is no SNR gap between the 3-bit optimal and the 3-bit proposed quantizer (the gap to the unquantized case is about $1/2\,$dB), whereas for very low outage probabilities, the proposed quantizer with 2-bit performans worse than the optimal quantizer with 2-bit. For the high target rate ($R = 0.75$, corresponding to 6bpcu), there is no gap between the optimal quantizers with 2 and 3-bit, and also the 2- and 3-bit proposed quantizers perform the same. The optimum quantizers perform very close to the unquantized case, but there is a gap of about $1\,$dB between the proposed quantizers and the unquantized case.

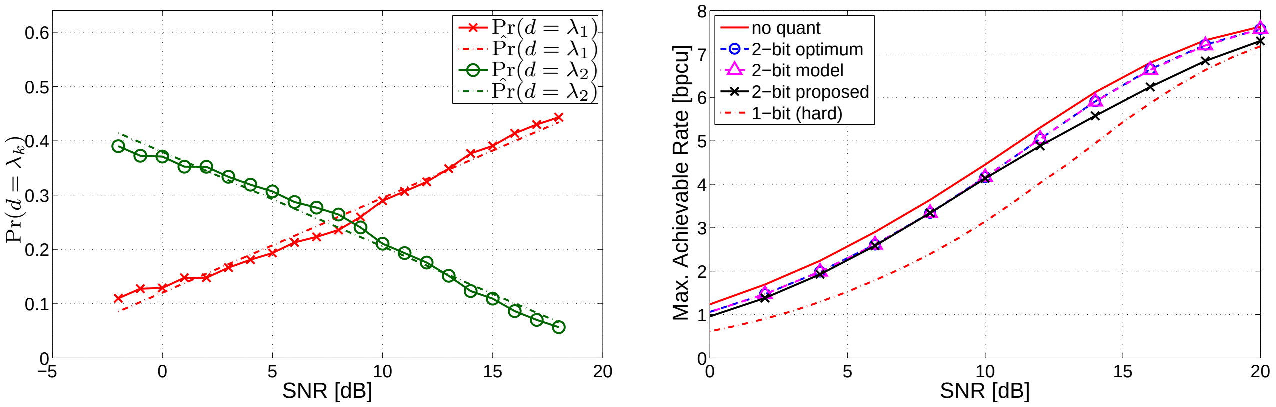
4.5 Estimation of Quantization Parameters

The proposed quantizers have been designed and optimized by means of extensive Monte Carlo simulations. The quantizer parameters (quantization intervals and quantizer outputs) depend on the system configuration and SNR. Storing these parameters for different system configurations and SNR values would require large lookup tables at the receiver and is therefore impractical. A solution to this problem is to design the quantizer on-the-fly, that is, during the data transmission and thereby without the need for lookup tables or Monte-Carlo simulations.

4.5.1 On-the-fly Design of Quantizer Intervals

The results of Section 4.4 for fast fading show that in case of 3 bit quantization, the equiprobable quantizer and the optimal quantizer exhibit the same performance, whereas in case of 2 bit quantization, this only holds true for rates $R < 3/4$. In general, the on-the-fly design of the equiprobable quantizer is much easier, therefore on-the-fly design of the equiprobable quantizer is sufficient for 3-bit quantization for all rates and for 2-bit quantization when $R < 3/4$. The boundaries of the equiprobable quantizer can be estimated by using an empirical estimate of the unconditional LLR distribution $F_\Lambda(\xi)$ and choosing the quantizer intervals such that (4.6) is fulfilled. The LLR distribution $F_\Lambda(\xi)$ can be obtained from the observation of a reasonable number of non-quantized LLRs.

In case of $R \geq 3/4$ and 2-bit quantization, the optimal quantizer performs significantly better,and thus its on-the-fly design needs to be addressed. For the design of the optimal quantizer, the conditional LLR distributions $f_{\Lambda|c}(\xi|b)$ have to be known, which strongly depend on the system parameters. Therefore, depending on the system parameters, different estimation procedures have to be followed. In the following we demonstrate this for the case of a 2×2 MIMO-system with 16QAM symbol alphabet and a 4×4 MIMO system with 4QAM symbol alphabet.



(a) (b)

Figure 4.8: 2×2 MIMO system with 16-QAM (a) Plot of $P(d = \lambda_k)$ of the optimal quantizer versus SNR. (b) Numerical capacity results obtained with the model (4.11).

2×2 **MIMO-system with 16QAM symbol alphabet.** Fig. 4.8(a) shows the probabilities $\Pr(d = \lambda_k)$ of the optimal quantizer output versus the SNR. It can be observed, that these probabilities can be modeled quite accurately as linear functions of the SNR, that is we propose to design the quantizer such that $\Pr(d = \lambda_k) = p_k^{\text{mod}}$ with

$$p_k^{\text{mod}} = a_k + b_k \rho \,. \tag{4.11}$$

The parameters a_k and b_k can be obtained e.g. using a least-squares approach. Using this model, we obtain the ergodic capacity of this quantizer design which is shown in Fig. 4.8(b) as the curve labeled '2-bit model'. It can be seen that this quantizer design performs as well as the numerically optimized 2-bit quantizer design (which is shown by the curve labeled '2-bit optimium').

4×4 **MIMO system with 4QAM symbol alphabet.** In this case the probabilities $\Pr(d = \lambda_k)$ of the optimal quantizer versus SNR are not easy to approximate as a linear function, nor a polynomial. A more successful approach is to directly model the quantizer boundary values versus SNR as a second-order polynomial,

$$i_3^{\text{mod}} = c + d\rho + e\rho^2 \,, \tag{4.12}$$

The parameters c, d, and e can be obtained from a least-squares match of the optimal quantizer boundaries (obtained by numerical optimization) for different SNR. The ergodic capacity of this quantizer design is plotted in Fig. 4.9 as the curve labeled '2-bit model'. Again, this quantizer design performs as well as the optimal 2-bit quantizer design (which is shown by the curve labeled '2-bit optimium').

4.5.2 Estimation of Quantization Levels

The quantization levels $\lambda_k^\star$ are calculated according to (cf. (4.8))

$$\lambda_k^\star = \log \frac{\Pr\{c = 1 | d = \lambda_k\}}{\Pr\{c = 0 | d = \lambda_k\}} = \log \frac{p_{1k}}{p_{0k}} \,.$$

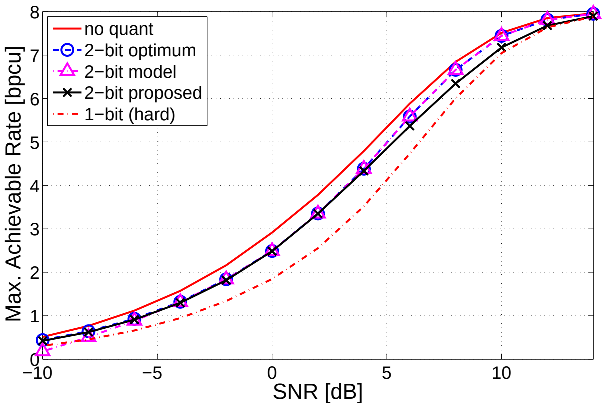


Figure 4.9: 4×4 MIMO system with 4-QAM: Performance of quantizer obtained with the model (4.12).

This requires the transition probabilities of the equivalent discrete channel and thus in turn the LLR distributions $f_{\Lambda|c}(\xi|c)$, which in general are unknown. Again, the quantization levels $\lambda_k^\star$ could be precalculated and stored in a lookup table, but this solution appears infeasible in practice, as the quantization levels depend on the system configuration and on the SNR. We thus address on-the-fly estimation of the quantization levels. Determining the quantization levels $\lambda_k^\star$ by estimating $f_{\Lambda|c}(\xi|c)$ is more difficult since the code bits are unknown at the receiver. Hence, we propose a method for estimating the $\lambda_k^\star$ from observations of the un-quantized LLRs Λ by use of a parametric model. We demonstrate this method for the case of a 2×2 MIMO-system with 16QAM symbol alphabet and a 4×4 MIMO system with 4QAM symbol alphabet. We note that other system parameters may require a different model.

2×2 **MIMO-system with 16QAM symbol alphabet.** In this case we use the following asymmetric-sided exponential model for the LLR distributions $f_{\Lambda|c}(\xi|c)$,

$$f_{\Lambda|c}(\xi|c\!=\!1) = \begin{cases} \frac{\alpha\beta}{\alpha+\beta}\exp(\alpha\xi) & \xi < 0\,, \\ \frac{\alpha\beta}{\alpha+\beta}\exp(-\beta\xi) & \xi \geq 0\,. \end{cases} \tag{4.13}$$

To estimate the two parameters $\alpha > 0$ and $\beta > 0$, we choose two bins $\bar{\mathcal{I}}_1$ and $\bar{\mathcal{I}}_2$ and use the non-quantized LLRs Λ to obtain empirical estimates $\hat{P}_i$, $i = 1, 2$, of the probabilities

$$P_i(\alpha, \beta) = \Pr\{\Lambda \in \bar{\mathcal{I}}_i\} = \int_{\bar{\mathcal{I}}_i} f_\Lambda(\xi)\, d\xi\,,$$

with $f_\Lambda(\xi) = \big[f_{\Lambda|c}(\xi|c\!=\!0) + f_{\Lambda|c}(\xi|c\!=\!1)\big]/2$. The system of equations $P_i(\alpha,\beta) = \hat{P}_i$ can then be solved numerically to obtain estimates of α and β. The transition probabilities of the equivalent channel and the quantization levels are then computed based on (4.13) using the estimates of α and β.

4×4 **MIMO system with 4QAM symbol alphabet.** In this case, a simple Gaussian parametric model for the LLR distributions $f_{\Lambda|c}(\xi|c)$ is used,

$$f_{\Lambda|c}(\xi|c\!=\!1) = \frac{1}{\sqrt{4\pi\mu^2}} \exp\Big(-\frac{1}{4\mu^2}|\Lambda - \mu|^2\Big)\,, \tag{4.14}$$

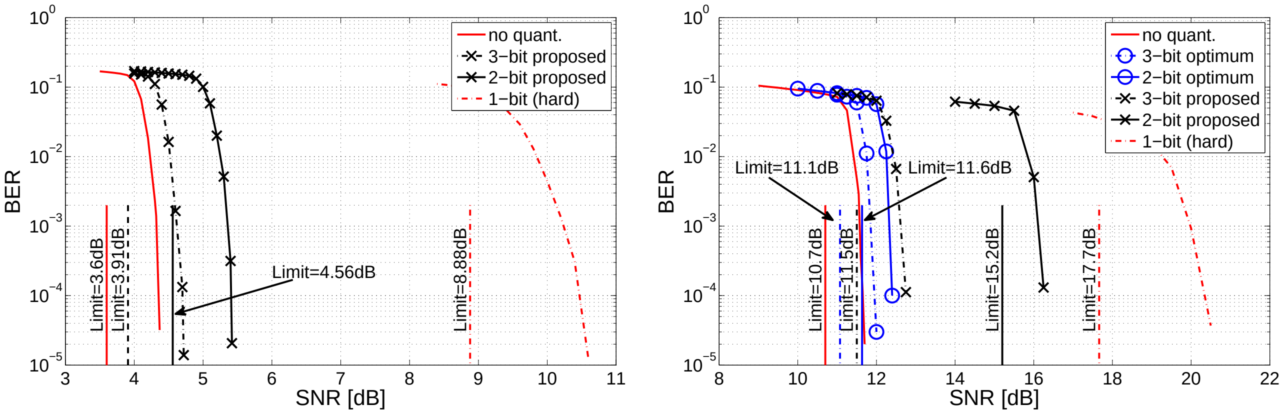


Figure 4.10: BER performance for a rate-1/2 LDPC coded SISO-BICM system with BPSK modulation and different LLR quantization word-lengths.

with $\sigma^2 = 2\mu^2$. The parameter μ can be obtained similar as before: We choose one bin $\bar{\mathcal{I}}$ and obtain the empirical estimate $\hat{P}$ from the non-quantized LLRs Λ. We then need to numerically solve

$$P(\mu) = \Pr\{\Lambda \in \bar{\mathcal{I}}\} = \int_{\bar{\mathcal{I}}} f_\Lambda(\xi)\, d\xi$$

to obtain the parameter μ. The transition probabilities of the equivalent channel and the quantization levels are then computed based on (4.14) using the estimate of μ.

4.6 Numerical BER Results

We finally provide BER simulations to verify the foregoing capacity results and to demonstrate the applicability of the proposed on-the-fly quantizer designs. For the simulations, we used a fast fading Rayleigh channel and the channel code was a regular LDPC code[2] with block length 64000.

4.6.1 SISO-BICM

We first consider the SISO-BICM system of Section 4.3.1. Fig. 4.10(a) shows the BER results for a SISO-BICM system with an LDPC of code rate $R = 1/2$ for proposed equiprobable LLR quantizers with different word-lengths together with the theoretical SNR thresholds (obtained from Fig. 4.3). At this rate, the optimal and proposed quantizers are effectively the same, so only the latter is shown. As a reference we also show the SNR thresholds (obtained from Fig. 4.3 and the BER achieved with the unquantized demodulator. All BER curves are reasonably close to the respective SNR thresholds. The gaps of 1-bit, 2-bit, and 3-bit LLR quantization to the non-quantized case respectively equal 6.2 dB, 1.1 dB, and 0.4 dB.

Fig. 4.10(b) shows the BER of the same system but with an LDPC of code rate $R = 3/4$ for the proposed and optimal quantizer design with different word-lengths. At this rate, there is a performance difference between optimal and proposed quantizer (compare with Fig. 4.3): With 3-bit quantization, the SNR gap between the

[2] The LDPC code was designed using the EPFL web-tool at `http://lthcwww.epfl.ch/research/ldpcopt`.

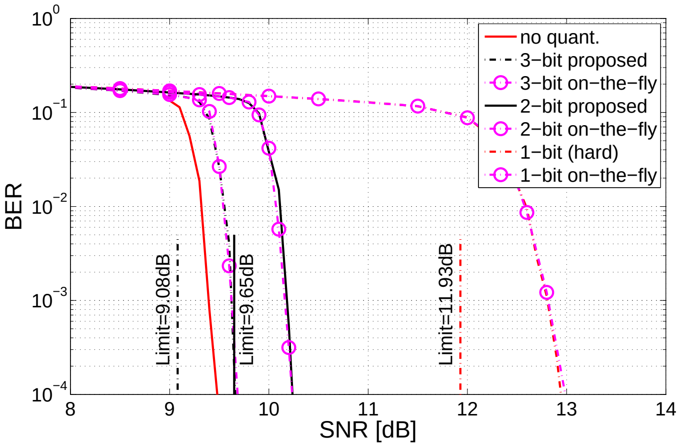


Figure 4.11: BER performance for a rate-1/2 LDPC coded 2×2 MIMO system with Gray-labeled 16-QAM and different LLR quantization word-lengths.

optimal and proposed quantizer is about 0.4 dB, this gap increases to $3.$ dB of 2-bit quantization. In general, the BER curves are quite close to the respective SNR thresholds, the only exception is that the SNR threshold of the 3-bit proposed quantizer is below the threshold of the 2-bit optimal quantizer, whereas the BER curves are reversed.

4.6.2 MIMO-BICM

$2{\times}2$ **MIMO case with** 16**QAM modulation.** Fig. 4.11 shows two (strongly overlapping) sets of BER curves for different LLR quantization word-lengths. One set of curves (labeled ‘proposed’) pertains to an offline design of the proposed LLR quantizer, whereas the other set (labeled ‘on-the-fly’) estimates the quantization parameters on-the-fly according to Section 4.5. The bins $\bar{\mathcal{I}}_1$ and $\bar{\mathcal{I}}_2$ were chosen heuristically. As reference the BER obtained with no quantization is also shown. Interestingly, there is a gap of 0.2 dB to the 3-bit quantized case which is not expected from the ergodic capacity curves in Fig. 4.5(b). The gap to the theoretical SNR thresholds (obtained from Fig. 4.5 and indicated by vertical lines) equals 0.6 dB for 3-bit and 2-bit quantization and 1 dB for 1-bit quantization (hard demodulation). Furthermore, the proposed on-the-fly estimator for the LLR quantizer parameters performs extremely well in this setup (virtually indistinguishable from the offline design). In case of $1-$bit quantization, the gap to the corresponding SNR limit is quite large. This is the case since an LDPC code optimized for an AWGN channel was used whereas the actual BICM modulation channel becomes a BSC.

To illustrate the importance of the correct choice of the LLR quantization levels, Fig. 4.12 shows BER versus quantizer output $\lambda_2 = -\lambda_1$ for the same MIMO system as before with 1-bit LLR quantization at an SNR of 12.8 dB. Here, the optimal quantizer level $\lambda_2^\star = 2.26$ (indicated in Fig. 4.12 by a dashed vertical line) achieves a BER of $4.5 \cdot 10^{-4}$. It is seen that the BER achieved by the belief propagation decoder is quite sensitive to the choice of the quantizer level λ_2: If the quantizer level is $\lambda_2 \leq 1.5$ or if $\lambda_2 \geq 4.3$, the BER of the overall system has deteriorated to about 10^{-1} (i.e., by more than 2 orders of magnitude).

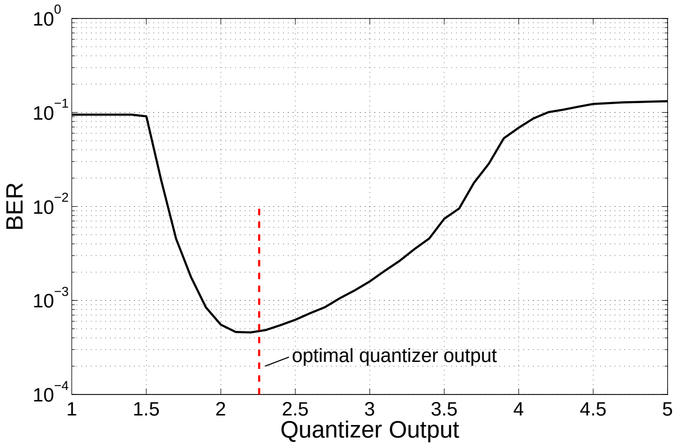


Figure 4.12: BER versus quantization level for 1-bit quantization at an SNR of 12.8 dB using a rate-$1/2$ LDPC code (2×2 MIMO, 16-QAM, Gray labeling).

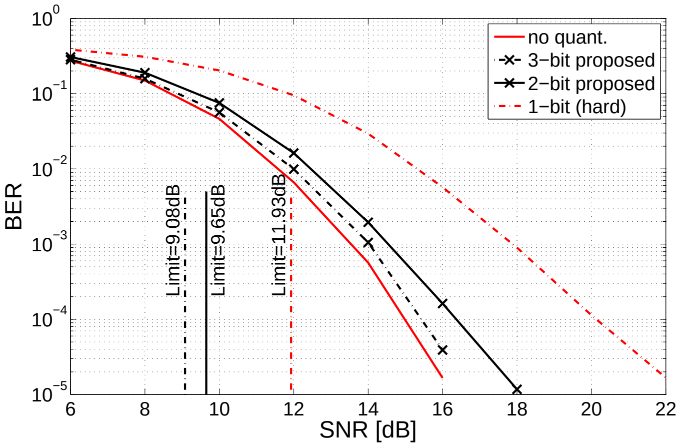


Figure 4.13: BER performance for a rate-1/2 convolutional coded 2×2 MIMO system with Gray-labeled 16-QAM and different LLR quantization word-lengths.

Convolutional Codes. Fig. 4.13 shows the same 2×2 MIMO system with 16QAM modulation employing a rate-$1/$ convolutional code. In this setup, the used channel decoder is a BCJR algorithm [8] delivering information bit LLRs. The code polynomial in octal notation is $[13, 15]$, and the block length was chosen to be 1024. It can be observed that there is a large gap to the theoretical SNR limits (in the order of several dB at a BER of 10^{-4}) and that the BER does not exhibit a threshold behavour as in the LDPC coded case; instead the BER curves exhibit a slope of about 6. The 3-bit case exhibits a slight performance loss of about 0.5 dB compared to the unquantized case. The gap between the 2-bit and the 3-bit quantized case is about 0.5 dB as is to be expected from the corresponding SNR limits. The 1-bit (hard) quantized case performs much worse.

We also show a plot of BER versus quantizer output $\lambda_2 = -\lambda_1$ with 1-bit LLR quantization at an SNR of 20 dB in Fig. 4.14. For values $\lambda_2 < 1$, the overall performance is about one order of magnitude worse than with the optimal quantizer output value of $\lambda_2^\star = 2.26$. On the other hand, for quantizer output values $\lambda_2 > \lambda_2^\star = 2.26$ no performance loss (as in the case of LDPC codes) is observed. This indicates that in case of

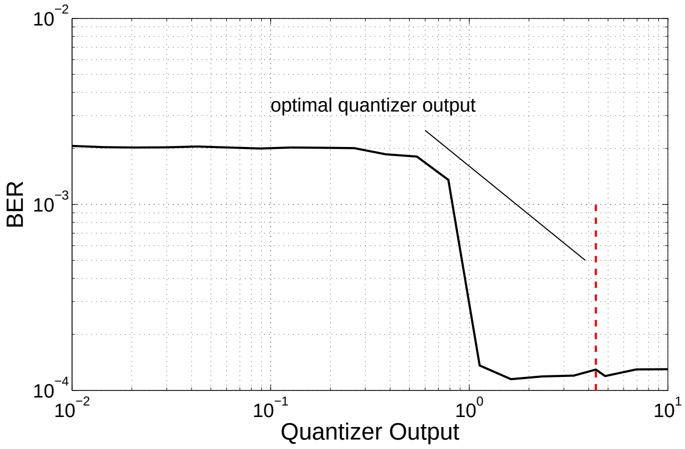


Figure 4.14: BER versus quantization level for 1-bit quantization at an SNR of $12.8\,$dB using a rate-$1/2$ convolutional code (2×2 MIMO, 16-QAM, Gray labeling).

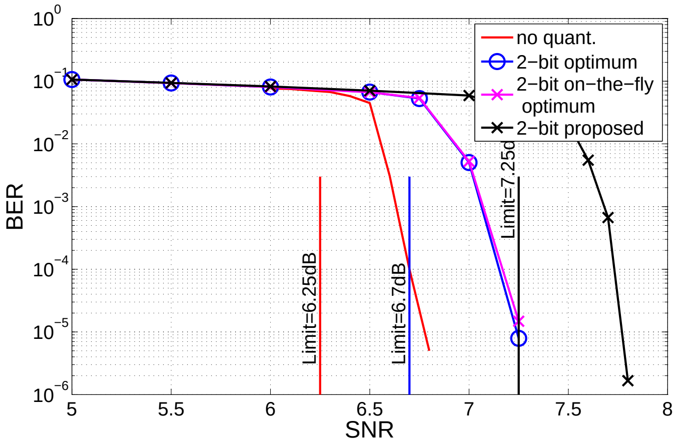


Figure 4.15: BER performance for a rate-3/4 LDPC coded 4×4 MIMO system with Gray labeled 4-QAM and different 2-bit quantizer designs.

convolutionally coded systems, the on-the-fly estimation of the quantizer output is not important as long as it is large enough. Choosing an SNR independent value of $\lambda_2 = 10$ should be sufficient for all relevant cases.

$4{\times}4$ **MIMO system with** 4**QAM modulation.** Fig. 4.15 shows results for this system with 4QAM modulation and 2 bit quantization. As can be seen in Fig. 4.6, at a rate of $R = 1/2$ almost no performance difference between the proposed and the optimal quantizer exists, therefore we chose the rate $R{=}3/4$. We show the BER of the system with three different quantizers: (i) the optimal quantizer, (ii) the proposed equiprobable quantizer, and (iii) the on-the-fly design of the optimal quantizer (denoted by '2-bit on-the-fly optimum'). As a reference we also plot the respective SNR thresholds. The proposed on-the-fly design of the optimal quantizer performs as good as the quantizer design using precomputed lookup tables. The SNR gaps between the proposed quantizer and the optimal quantizer and between the unquantized demodulator and the optimum quantizer are both about $0.5\,$dB; this results is consistent with the observations in Fig. 4.6.

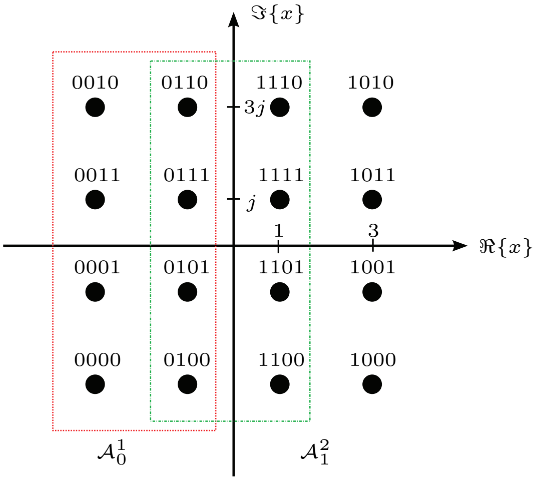


Figure 4.16: 16QAM constellation with Gray labeling; sets $\mathcal{A}_0^1$ and $\mathcal{A}_1^2$ are shown for illustration.

4.7 Low-Complexity Quantized Soft-MMSE Demodulator

4.7.1 Introduction

In [68], an efficient approach to compute LLRs by means of an MMSE filter was presented. The received vector is multiplied with an MMSE filter $\mathbf{W}$, yielding

$$\hat{\mathbf{x}}_{\text{MMSE}} = \mathbf{W}(\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H\mathbf{y}\,,$$

with the MMSE filter matrix $\mathbf{W} = \left[\mathbf{I} + \sigma^2(\mathbf{H}^H\mathbf{H})^{-1}\right]^{-1}$. The LLRs Λ_k^q corresponding to the bit c_k^q are then obtained by means of a *per-layer* max-log LLR calculation according to

$$\Lambda_k^q = \frac{1}{\sigma_k^2}\left[\min_{x\in\mathcal{A}_0^q}|\hat{x}_k - x|^2 - \min_{x\in\mathcal{A}_1^q}|\hat{x}_k - x|^2\right]. \tag{4.15}$$

Here, $\hat{x}_k = \frac{(\hat{\mathbf{x}}_{\text{MMSE}})_k}{(\mathbf{W})_{k,k}}$, $\sigma_k^2 = \frac{1-(\mathbf{W})_{k,k}}{(\mathbf{W})_{k,k}}$, and $\mathcal{A}_b^q$ denotes the set of symbols x_k for which $c_k^q = b$.

4.7.2 Efficient Calculation of Quantized LLRs

We next modify the MMSE detector in such a way that it directly delivers quantized LLRs. In Fig. 4.16 a 16QAM symbol alphabet with Gray mapping is shown together with two exemplary sets $\mathcal{A}_0^1$ and $\mathcal{A}_1^2$. It can be seen that each symbol x_k can be expressed as the sum of independently modulated real and imaginary parts. Furthermore, it can be observed that the first two bits of the bit label (that is c_k^1 and c_k^2) determine the real part of the symbol x_k, whereas the last two (that is c_k^3 and c_k^4) determine the imaginary part. Reversing the argument, the LLRs Λ_k^1 and Λ_k^2 depend on the real part of $\hat{x}_k$ only, whereas the other two LLRs Λ_k^3 and Λ_k^4 depend only on the imaginary part of $\hat{x}_k$.

In [69] it was shown that the LLRs Λ_k^q calculated by the soft-MMSE demodulator of (4.15) are piecewise linear functions of either the real or imaginary part of $\hat{x}_k$ and closed-form expressions were given. In the special case of 16QAM modulation and Gray mapping as in Fig. 4.16 we have $\Lambda_k^1 = \frac{1}{\sigma_k^2}\alpha(\Re(\hat{x}_k))$, $\Lambda_k^2 = \frac{1}{\sigma_k^2}\beta(\Re(\hat{x}_k))$,

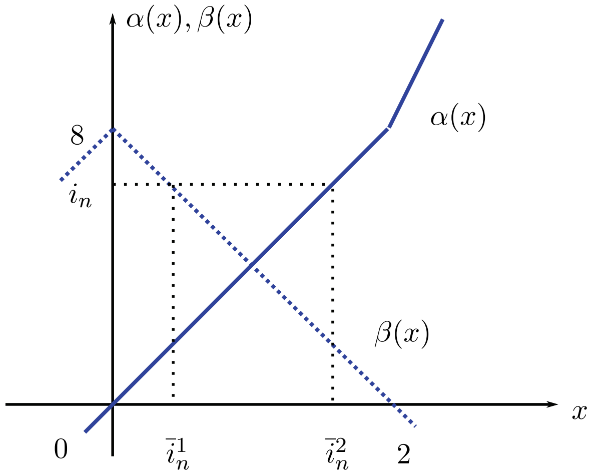


Figure 4.17: LLR functions for each of the 4 bits of a 16-QAM symbol in a SISO-BICM system.

$\Lambda_k^3 = \frac{1}{\sigma_k^2}\alpha(\Im(\hat{x}_k))$, and $\Lambda_k^4 = \frac{1}{\sigma_k^2}\beta(\Im(\hat{x}_k))$. The functions $\alpha(x)$ and $\beta(x)$ are defined as

$$\alpha(x) = \begin{cases} 4x\,, & |x| \leq 2 \\ -8 + 8x\,, & x > 2 \\ 8 + 8x\,, & x < -2\,, \end{cases}$$

$$\beta(x) = \begin{cases} 8 + 4x\,, & x \leq 0 \\ 8 - 4x\,, & x > 0\,, \end{cases} \tag{4.16}$$

and shown in Fig. 4.17. We note that $\alpha(\cdot)$ is odd symmetric, whereas $\beta(\cdot)$ is even symmetric. Using (4.16), the maximization in (4.15) can be avoided, and the LLRs Λ_k^q can be directly calculated from $\hat{x}_k$ via simple additions and multiplications. Furthermore, calculating continuous-valued LLRs which are quantized afterwards can be avoided; instead the quantized LLRs can be directly obtained by quantizing $\hat{x}_k$. This will be explained in the following.

We consider an L-bit quantizer characterized by $M = 2^L$ bins $\mathcal{I}_m = [i_{m-1}, i_m]$, $m = 1, \ldots, M$. We use the convention $i_0 = -\infty$, $i_M = \infty$ and assume symmetric bins (this is motivated by the symmetry of the LLR distributions), with boundaries i_m sorted in ascending order. The quantizer $\mathcal{Q}(\cdot)$ maps the continuous-valued LLR Λ_k^q to discrete LLRs d_k^q according to

$$d_k^q = \mathcal{Q}(\Lambda_k^q) = \lambda_m \quad \text{if } \Lambda_k^q \in \mathcal{I}_m\,,$$

where, $\lambda_m \in \mathcal{I}_m$ denotes the mth quantization level. Using (4.16), we can express the intervals $\mathcal{I}_m$ also in terms of the real or imaginary part of $\hat{x}_k$. In the following we restrict ourselves to obtaining the two LLRs Λ_k^1 and Λ_k^2 from $\Re(\hat{x}_k)$; the imaginary part follows analogously. In this case, the quantizer becomes

$$d_k^q = \mathcal{Q}^q\big(\Re(\hat{x}_k)\big) = \lambda_m \quad \text{if } \Re(\hat{x}_k) \in \bar{\mathcal{I}}_m^q\,,$$

with the transformed interval $\bar{\mathcal{I}}_m^q = [\bar{i}_{m-1}^q, \bar{i}_m^q]$. Note that the quantizer interval $\bar{\mathcal{I}}_m^q$ depends on the bit position q. Using (4.16) and Fig. 4.17 we can transform an interval boundary i_m from the interval $\mathcal{I}_m$ into the

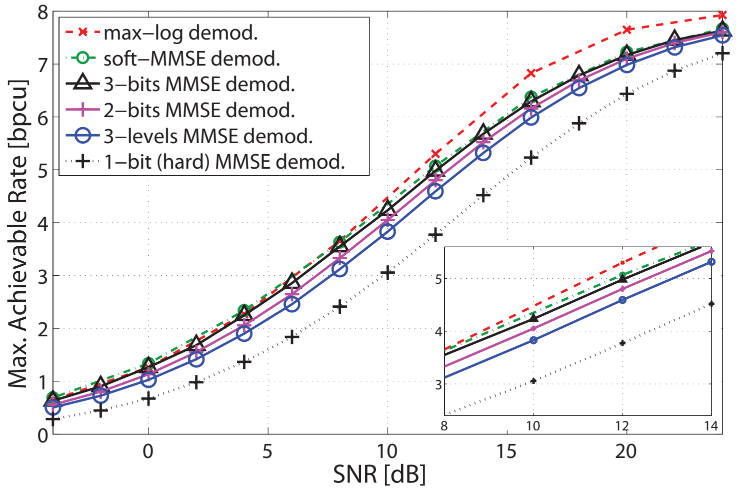


Figure 4.18: Ergodic capacity of a 2×2 MIMO-BICM system with Gray-labeled 16-QAM using a max-log demodulator and soft-MMSE demodulators for different LLR quantization word-lengths.

corresponding boundary

$$\bar{i}_m^1 = \alpha^{-1}(i_m) = \begin{cases} \frac{i_m}{4}, & |i_m| \le 8 \\ \frac{i_m+8}{8}, & i_m > 8 \\ \frac{i_m-8}{8}, & i_m < -8 \end{cases},$$

and

$$\bar{i}_m^2 = \beta^{-1}(i_m) = \begin{cases} \frac{i_m-8}{4}, & i_m \ge 0 \\ \frac{8-i_m}{4}, & i_m < 0 \end{cases},$$

respectively. Note that due to the even symmetry of $\beta(x)$, the interval $\mathcal{I}_m = [i_{m-1}, i_m]$ becomes the union of two intervals $\bar{\mathcal{I}}_m = [-\bar{i}_{m-1}, -\bar{i}_m] \cup [\bar{i}_{m-1}, \bar{i}_m]$.

Complexity. The complexity of the proposed soft-MMSE demodulator calculating quantized LLRs is given by calculation of the MMSE filter matrix $\mathbf{W} = \left[\mathbf{I} + \sigma^2(\mathbf{H}^H\mathbf{H})^{-1}\right]^{-1}$ (which has to be calculated anew for every channel realization) and the matrix-vector multiplication $\hat{\mathbf{x}}_{\text{MMSE}} = \mathbf{W}(\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H\mathbf{y}$. The quantization intervals $\mathcal{I}_m = [i_{m-1}, i_m]$ can be calculated offline (and stored in a lookup table) or designed on-the-fly as was presented in Section 4.5. Quantizing the real and imaginary parts of $\hat{\mathbf{x}}_{\text{MMSE}}$ yields the corresponding quantized LLRs, which can be implemented with minimal complexity.

4.7.3 Ergodic Capacity

We next provide numerical results for the ergodic capacity of a MIMO-BICM system with soft-MMSE demodulators and the optimal quantizer. We consider the equivalent discrete channel with binary input $c \in \{0, 1\}$ and M-ary output $d \in \{\lambda_1, \ldots, \lambda_M\}$, and calculate the mutual information according to (4.3). In general the conditional densities $f_{\Lambda|c}(\xi|b)$ are unknown, so they have to be estimated by means of Monte Carlo simulations. Based on these estimates, the maximization of $I(c; d)$ is then performed numerically.

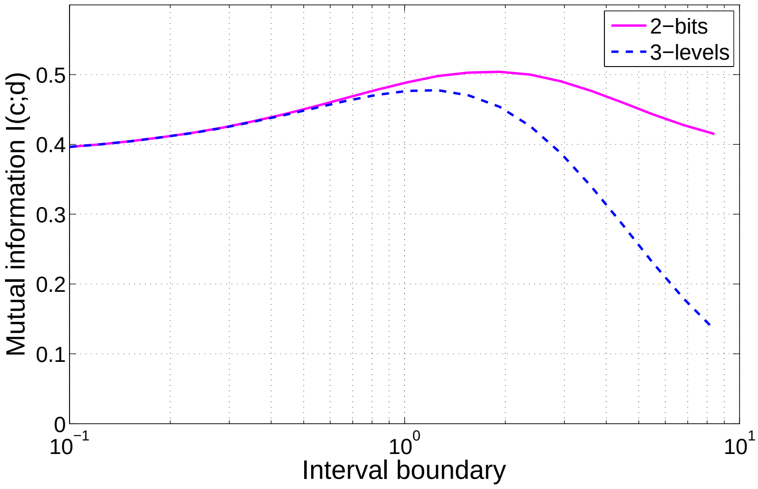


Figure 4.19: Impact of interval boundary on ergodic capacity of a 2×2 MIMO-BICM system with Gray-labeled 16-QAM at an SNR of 10 dB for 2-bit and 3-level quantization.

Uncorrelated MIMO-channel. Numerical results for the rate in bits per channel use (bpcu) versus SNR achievable with the quantized MMSE demodulator of different word-length are shown in Fig. 4.18 for a 2×2 uncorrelated MIMO system (that is, $\mathbf{C}_h = \mathbf{I}$) with 16QAM Gray-mapped modulation. As a reference, we show the capacity of non-quantized max-log demodulation (labeled 'max-log demod.') from (4.1), and the non-quantized soft-MMSE demodulator (labeled 'soft-MMSE demod.'). As stated in [20], the gap between max-log demodulation and soft-MMSE demodulation is very small for $R \leq 0.5$, the soft-MMSE demodulator even outperforms the max-log demodulator for small rates. The gaps of 3-bit, 2-bit, and 3-level quantization to the non-quantized MMSE demodulator equal 0.3 dB, 0.75 dB, 1.5 dB, respectively, at a rate of $R = 1/2$. Finally, the SNR loss with a hard-MMSE demodulator (which can be interpreted as 1-bit quantization) is about 3.5 dB, again at a rate of $R = 1/2$. This shows that increasing the number of quantization levels from 2 to 3 yields an SNR gain of 2 dB in this case.

In Fig. 4.19 we show the impact of the interval boundary on the ergodic capacity of the soft-out MMSE demodulator quantized with 3 levels and 2 bits, respectively. The system is the same considered before at an SNR of 10 dB. In both cases, one interval boundary is sufficient to index all quantizers in these cases. In case of quantization with 2 bits, the ergodic capacity changes only weakly with the interval boundary. In case of either very large or very small boundary values, the ergodic capacity approaches the capacity of 1-bit quantization. The ergodic capacity for the 3-level quantization depends much stronger on the interval boundary. In the limit of small boundary values, the center interval $\mathcal{I}_2 = [i_1, i_2]$ vanishes, and the ergodic capacity approaches the 1-bit case. In case of large interval boundaries, almost all LLRs fall into the middle interval and therefore the capacity quickly deteriorates.

Comparison with quantized max-log demodulator. Using results from the previous section and [20], we can compare the required SNR of quantized max-log demodulators and quantized soft-MMSE demodulators. At a rate of $R = 6$ bpcu there is an SNR gap of about 1 dB between the unquantized max-log demodulator and

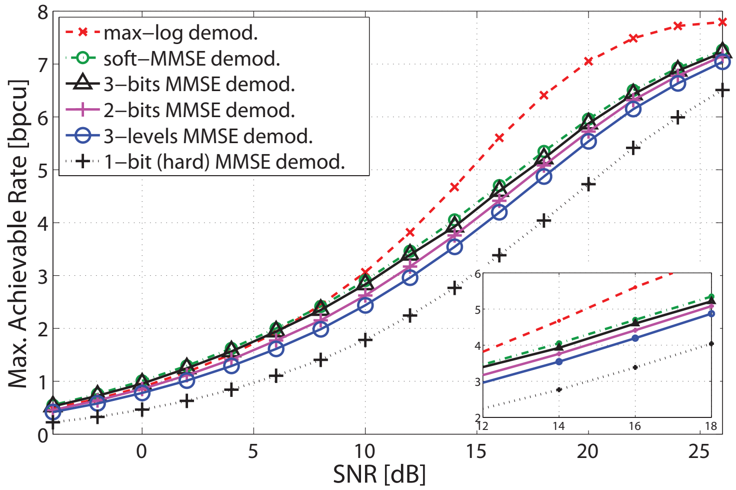


Figure 4.20: Ergodic capacity of a 2×2 MIMO-BICM system with correlated channel and a Gray-labeled 16-QAM symbol alphabet. Max-log demodulator and soft-MMSE demodulators using different LLR quantization word-lengths are shown.

the unquantized soft-MMSE demodulator. The max-log demodulator with 3 level quantization performs almost the same as the unquantized soft-MMSE demodulator, and the MMSE demodulator with 3-level quantization requires the same SNR as the 1-bit (hard) max-log demodulator. At a rate of $R=4$ bpcu the SNR gap between the unquantized max-log demodulator and the unquantized soft-MMSE demodulator is about 0.2 dB, and the 3-level quantized max-log demodulator performs only about 0.3 dB better than the 3-level quantized soft-MMSE demodulator. For rates $R = 4$ bpcu, there is only a very small gap in performance between the max-log demodulator and the soft-MMSE demodulator, with the latter being of significantly lower complexity. This gap between soft-MMSE demodulator and max-log demodulator increases for higher rates.

Correlated MIMO-channel. Fig. 4.20 shows similar results for the case of a channel with spatial correlation. We used a Kronecker model [34] for the correlation matrix of the channel, i.e., $\mathbf{C}_h = \mathbf{S}^{1/2} \otimes \mathbf{R}^{1/2}$, with the transmit and receive correlation matrices respectively chosen as

$$\mathbf{S} = \mathbf{R} = \begin{pmatrix} 1 & 0.7 \\ 0.7 & 1 \end{pmatrix}.$$

It is seen that all rates are lower than in the uncorrelated case, otherwise the results are quite similar: 3-bit quantization is sufficient for all rates, whereas 2-bit quantization leads to an SNR loss of about 0.7 dB, and quantization with 3 levels incurs an SNR loss of about 1.5 dB, at a rate of $R = 4$ bpcu, respectively. The hard-MMSE demodulator incurs an SNR loss of 4 dB, which is slightly higher than in the uncorrelated case. Comparing these results with results for the unquantized case shown in Fig. 4.18 reveals that correlation decreases ergodic capacity significantly: At a rate of $R=4$ bpcu the SNR loss due to correlation is about 4 dB.

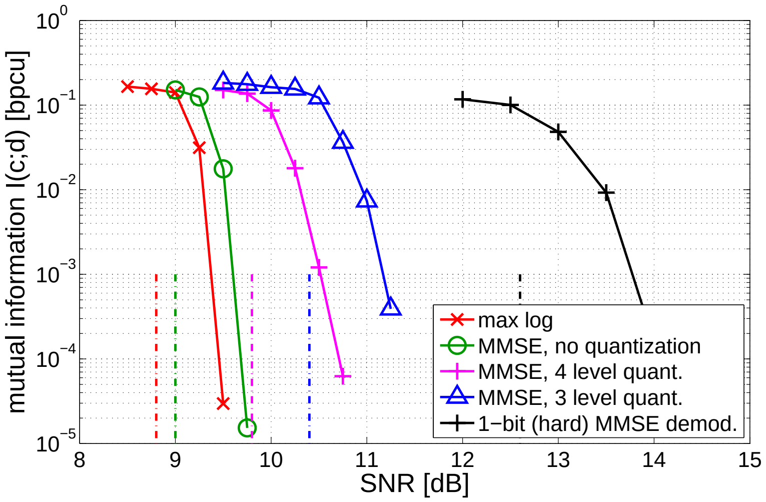


Figure 4.21: BER versus SNR of a 2×2 MIMO-BICM system with Gray-labeled 16-QAM using soft-MMSE demodulators with different quantization bit-width.

4.7.4 Numerical BER Results

To verify the capacity results, we performed BER simulations for the uncorrelated MIMO-BICM system in ergodic Rayleigh fast fading. The channel code was a regular LDPC code[3] with rate $1/2$ and block length 64000. Fig. 4.21 shows the BER for the various considered demodulators and the SNR thresholds obtained from Fig. 4.18 for $R = 4$ bpcu. All BER curves are reasonably close to the respective SNR thresholds. The gaps of 2-bit, 3-level, and 1-bit LLR quantization to the non-quantized MMSE demodulator respectively equal 0.8 dB, 1.4 dB, and 3.6 dB. Finally, the gap between the unquantized MMSE demodulator and the max-log demodulator of (4.1) is about 0.2 dB.

[3] The LDPC code was designed using the EPFL web-tool at `http://lthcwww.epfl.ch/research/ldpcopt`.

5

MIMO-BICM with imperfect CSI

THE receivers considered in the previous chapters either jointly estimate the channel and detect the data (Chapter 3) or assumed perfect CSI at the receiver (Chapter 4). Receivers performing joint channel estimation and data detection exhibit very good performance (almost as good as receivers with perfect CSI) at the cost of increased complexity. One approach to reduce complexity is to use receivers which estimate the channel by means of pilot symbols inserted at the transmitter but do not update their initial channel estimate. Demodulators for systems using pilot-based channel estimation are usually designed assuming perfect channel state information (CSI) and therefore such conventional designs yield only good results if this assumption is well fulfilled. Due to limited power budgets at the transmitters, the power available for the transmission of pilot symbols is limited, and this will result in imperfect CSI at the receiver. The resulting difference between the true and the estimated channel causes conventional demodulator designs to be mismatched, which results in performance degradation. In this chapter we investigate the effects of imperfect CSI on conventional receivers. We also propose optimal receiver designs, which take the uncertainty of the channel estimate into account and thereby provide better performance.

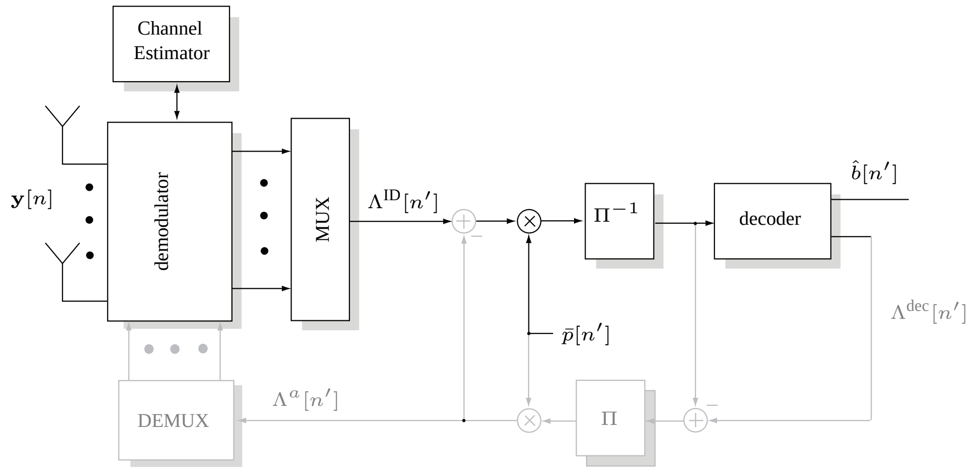


Figure 5.1: Block diagram of a MIMO-BICM receiver with channel estimation. Additional parts required for an iterative receiver structure are shown in gray.

5.1 Background and State of the Art

Soft-out demodulators provide the channel decoder not only with bit-decision, but also with reliability information regarding this bit decision. Usually, this reliability information is conveyed by means of LLRs (see Subsection 2.1.3). To reduce the complexity of the demodulator, in [68] a soft-MMSE demodulator has been proposed, and [20] demonstrated that performance of this soft-MMSE demodulator is very close to the *optimal* demodulator for most applications. BICM systems with iterative decoding (BICM-ID) have been observed to yield excellent performance (see Subsection 2.1.3 or [11]) and BICM-ID can be employed in MIMO systems as well. The convergence properties of BICM-ID receivers has been studied in [70, 71] using EXIT charts [24]. BICM-ID systems that utilize LDPC codes [41] have been shown to be able to operate close to capacity limits. In addition, the EXIT charts of LDPC codes can be optimized by appropriately adjusting their variable and check node degree distributions. Demodulation in MIMO system with imperfect CSI has been first adressed in [72]. Taking into account the statistics of the channel estimate, an optimal version of the max-log demodulator has been proposed in [23]. Its performance in the context of iteratively decoded MIMO-BICM was investigated in [73] by means of EXIT charts. In a similar spirit, a modified soft MMSE demodulator using CSI statistics has been presented [22]; its optimal performance was verified in terms of bit error rate (BER) using an off-the-shelf LDPC code.

The chapter is organized as follows: In Section 5.2 we present the system model, and in Section 5.3 the mismatched and optimal receivers are derived. We also provide a derivation of the receiver algorithms by means of the factor graph framework in Section 5.4. Finally, Section 5.5 illustrates our numerical results.

5.2 System Model

We consider a MIMO-BICM system with M_{T} transmit antennas and M_{R} receive antennas; the system model is shown in Fig. 2.2 and has been described in Section 2.1.3. Assuming block flat fading, the length-M_{R} receive vector at symbol time n is given by (for notational convenience, we omit the time index n)

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w}\,. \tag{5.1}$$

Here, $\mathbf{H}$ denotes the $M_\mathrm{R} \times M_\mathrm{T}$ MIMO channel matrix, $\mathbf{w} \sim \mathcal{CN}(\mathbf{0}, \sigma_w^2\mathbf{I})$ denotes i.i.d. complex Gaussian noise. By stacking the columns of the channel matrix $\mathbf{H}$ into a vector $\mathbf{h} = \mathrm{vec}\{\mathbf{H}\}$ and defining $\mathcal{X} = \mathbf{x}^T \otimes \mathbf{I}$, (5.1) can be rewritten as

$$\mathbf{y} = \mathcal{X}\mathbf{h} + \mathbf{w}\,, \qquad n = 1, \ldots, N. \tag{5.2}$$

The channel vector $\mathbf{h}$ is assumed zero-mean complex Gaussian with covariance matrix $\mathbf{C}_h$, $\mathbf{h} \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}_h)$.

For channel estimation, a length-N_p pilot sequence with corresponding $N_pM_\mathrm{R} \times M_\mathrm{T}M_\mathrm{R}$ matrix $\tilde{\mathcal{X}}_p = (\mathcal{X}_p^T[1] \cdots \mathcal{X}_p^T[N_p])^T$ is transmitted during a training phase. Each training symbol $\mathcal{X}_p[n]$ is assumed to be orthogonal $\mathcal{X}_p^H[n]\mathcal{X}_p[n] = P\mathbf{I}$; the total training power is denoted as $E_p \triangleq N_pP$. The received pilot sequence vector $\tilde{\mathbf{y}}_p[n] = (\mathbf{y}_{p,1}^T \cdots \mathbf{y}_{p,N_p}^T)^T$ (associated with the channel matrix at time n) has length N_pM_R and is given by (cf. (5.2))

$$\tilde{\mathbf{y}}_p[n] = \tilde{\mathcal{X}}_p\mathbf{h}[n] + \tilde{\mathbf{w}}[n], \tag{5.3}$$

with the stacked noise vector $\tilde{\mathbf{w}}[n] = (\mathbf{w}_1^T \cdots \mathbf{w}_{N_p}^T)^T$.

5.3 MIMO-BICM-ID Receivers with Imperfect CSI

The (possibly iterative) receiver structure employed is shown in Fig. 5.1. It is very similar to the BICM receiver shown in Fig. 2.2 and the BICM-ID receiver in Fig. 2.3. Basically, the receiver consists of three blocks: a channel estimator, a demodulator and a channel decoder. Initially, the channel estimator provides an estimate $\hat{\mathbf{h}}$ of the channel based on the known pilot symbols.

The channel decoder is connected to the demodulator by means of an interleaver; in case of perfect CSI the optimal demodulator calculates LLRs according to

$$\Lambda_l[n] = \log\frac{\Pr(c_l[n]{=}1|\mathbf{y}[n], \mathbf{H}[n])}{\Pr(c_l[n]{=}0|\mathbf{y}[n], \mathbf{H}[n])}\,, \tag{5.4}$$

(cf. (2.3)). Using the MIMO-BICM system model (5.1), this expression becomes

$$\Lambda_l[n] = \log\frac{\sum_{\mathbf{x}\in\chi_l^1}\exp\big\{-\frac{1}{\sigma_w^2}||\mathbf{y}-\mathbf{H}\mathbf{x}||^2\big\}}{\sum_{\mathbf{x}\in\chi_l^0}\exp\big\{-\frac{1}{\sigma_w^2}||\mathbf{y}-\mathbf{H}\mathbf{x}||^2\big\}} \tag{5.5}$$

In case of BICM-ID, the demodulator delivers LLRs according to

$$\Lambda_l^\mathrm{ID}[n] = \log\frac{\sum_{\mathbf{x}\in\chi_l^1}\exp\big\{-\frac{1}{\sigma_w^2}||\mathbf{y}-\mathbf{H}\mathbf{x}||^2\big\}\prod_{l=1}^{mM_\mathrm{T}}\exp(c_l(\mathbf{x})\Lambda_l^a)}{\sum_{\mathbf{x}\in\chi_l^0}\exp\big\{-\frac{1}{\sigma_w^2}||\mathbf{y}-\mathbf{H}\mathbf{x}||^2\big\}\prod_{l=1}^{mM_\mathrm{T}}\exp(c_l(\mathbf{x})\Lambda_l^a)}\,, \tag{5.6}$$

(cf. (2.10)), where Λ_l^a denotes the *a priori* information provided to the demodulator by the channel decoder.

5.3.1 Channel Estimation

The channel matrix $\mathbf{H}$ is estimated by means of the received pilot sequence $\tilde{\mathbf{y}}_p$. The general linear estimator is given by

$$\hat{\mathbf{h}} = \mathbf{A}\tilde{\mathbf{y}}_p = \mathbf{A}(\tilde{\mathcal{X}}_p\mathbf{h} + \tilde{\mathbf{w}}). \tag{5.7}$$

The channel estimation matrix $\mathbf{A}$ is an $M_\text{T}M_\text{R} \times N_pM_\text{R}$ full-rank matrix.

Least-Squares (LS) Estimator. The LS channel estimator [51] $\hat{\mathbf{h}}_\text{LS}$ is obtained by setting $\mathbf{A} = \tilde{\mathcal{X}}_p^{\#}$, and yields

$$\hat{\mathbf{h}}_\text{LS} = \tilde{\mathcal{X}}_p^{\#}\mathbf{y}_p = \mathbf{h} + \mathbf{e}.$$

Assuming orthogonal training sequences, the elements of the error vector $\mathbf{e}$ are distributed according to $\mathbf{e} \sim \mathcal{CN}(\mathbf{0}, \sigma_e^2)$, with variance $\sigma_e^2 = \frac{\sigma_w^2}{N_pP}$.

Minimum Mean Square Error (MMSE) Estimator. The MMSE estimator [51] $\hat{\mathbf{h}}_\text{MMSE}$ is obtained by choosing the estimator matrix according to

$$\mathbf{A} = \frac{1}{\sigma_w^2}\mathbf{\Sigma}\tilde{\mathcal{X}}_p^H,$$

with the matrix $\mathbf{\Sigma} = \left(\mathbf{C}_h^{-1} + \frac{N_pP}{\sigma_w^2}\mathbf{I}\right)^{-1}$.

The posterior density $f(\mathbf{h}|\hat{\mathbf{h}})$ can be obtained from

$$f(\mathbf{h}|\hat{\mathbf{h}}) = \frac{f(\hat{\mathbf{h}}|\mathbf{h})f(\mathbf{h})}{f(\hat{\mathbf{h}})},$$

with $f(\hat{\mathbf{h}}) = \int f(\hat{\mathbf{h}}|\mathbf{h})f(\mathbf{h})d\mathbf{h}$. From (5.7) follows $\hat{\mathbf{h}}|\mathbf{h} \sim \mathcal{CN}(\mathbf{A}\tilde{\mathcal{S}}_p\mathbf{h}, \sigma_w^2\mathbf{A}\mathbf{A}^H)$, and using the channel model of Section 5.2, the orthogonality of the training sequences $\tilde{\mathcal{S}}_p$ and the fact that $\mathbf{A}$ has full rank, $f(\mathbf{h}|\hat{\mathbf{h}})$ is complex Gaussian

$$\mathbf{h}|\hat{\mathbf{h}} \sim \mathcal{CN}\big(\hat{\mathbf{h}}_\text{MMSE}, \mathbf{\Sigma}\big). \tag{5.8}$$

Note that this density is independent of the actual linear estimator used, as long as the estimator matrix $\mathbf{A}$ is full rank.

5.3.2 Genie and Mismatched Demodulator

We fist consider a genie demodulator which is in possession of perfect CSI. Since max-log demodulation tends to be computationally expensive, [68] proposed a soft demodulator based on (linear) MMSE equalization and *per-layer* max-log LLR calculation. The MMSE equalizer output is given by

$$\hat{\mathbf{x}}^\text{MMSE} = \mathcal{E}\{\mathbf{x}\mathbf{y}^H|\mathbf{H}\}\left(\mathcal{E}\{\mathbf{y}\mathbf{y}^H|\mathbf{H}\}\right)^{-1}\mathbf{y} = \mathbf{W}\mathbf{y} \tag{5.9}$$

with the Wiener filter

$$\mathbf{W} = \mathbf{H}^H\Big(\mathbf{H}\mathbf{H}^H + \frac{\sigma_w^2}{E_s}\mathbf{I}\Big)^{-1}. \tag{5.10}$$

Assuming that the residual interference at the equalizer output is Gaussian, the approximate LLR for the ith bit in layer l is subsequently computed according to

$$\Lambda_{l,i}^{\text{MMSE}} = \frac{1}{\sigma_l^2}\bigg[\min_{x\in\mathcal{A}_i^0}|\hat{x}_l^{\text{MMSE}} - \mu_l x|^2 - \min_{x\in\mathcal{A}_i^1}|\hat{x}_l^{\text{MMSE}} - \mu_l x|^2\bigg].$$

Here $\mu_l = [\mathbf{WH}]_{l,l}$, $\sigma_l^2 = \mu_l - \mu_l^2$, and $\mathcal{A}_i^b$ denotes the set of transmit symbols whose bit label at position i equals b. The *genie MMSE* demodulator just described assumes perfect CSI. Practical implementations use a *mismatched MMSE* demodulator in which the true channel $\mathbf{H}$ in (5.10) is replaced with a channel estimate $\widehat{\mathbf{H}}$. We note that the performance of mismatched max-log and MMSE demodulation depends critically on the actual channel estimate.

In case of the BICM demodulator (5.5) we obtain the mismatched BICM demodulator, in case of the BICM-ID demodulator (5.6) we obtain the mismatched BICM-ID demodulator.

5.3.3 Optimal Demodulators

The mismatched demodulators do not exploit the statistical information about $\mathbf{h}$ conveyed by the channel estimate $\hat{\mathbf{h}}$ according to (5.8). Rather than replacing $\mathbf{H}$ with $\widehat{\mathbf{H}}$ in the final results (5.5), (5.6) and (5.10), this replacement should be made right in the beginning, i.e., the conditioning in (5.4) and (5.9) should be with respect to $\widehat{\mathbf{H}}$ instead of $\mathbf{H}$.

Optimal max-log demodulators. Using the relation

$$f(\mathbf{y}|\mathbf{x},\widehat{\mathbf{H}}) = \int f(\mathbf{y}|\mathbf{x},\mathbf{H})f(\mathbf{H}|\widehat{\mathbf{H}})\,d\mathbf{H}\,,$$

and the Gaussianity of the densities involved, we obtain

$$\mathbf{y}|\mathcal{X},\widehat{\mathbf{H}} \sim \mathcal{CN}\big(\widehat{\mathbf{H}}_{\text{MMSE}}\mathbf{x}, \mathcal{X}\Sigma\mathcal{X}^H + \sigma_w^2\mathbf{I}\big)\,, \tag{5.11}$$

with $\widehat{\mathbf{H}}_{\text{MMSE}} = \text{unvec}\{\widehat{\mathbf{H}}_{\text{MMSE}}\}$ (see Section 5.3.1). This distribution is again independent of the the actual channel estimator used and leads to the *optimal BICM* max-log demodulator, and the *optimal BICM-ID* max-log demodulator, both of which replace $\frac{1}{\sigma_w^2}\|\mathbf{y}-\mathbf{Hs}\|^2$ in (5.5) and (5.6), respectively, with the metric

$$(\mathbf{y}-\widehat{\mathbf{H}}_{\text{MMSE}}\mathbf{s})^H\big(\Sigma_{\mathbf{s}}+\sigma_w^2\mathbf{I}\big)^{-1}(\mathbf{y}-\widehat{\mathbf{H}}_{\text{MMSE}}\mathbf{s}) - \log\det\big(\Sigma_{\mathbf{s}}+\sigma_w^2\mathbf{I}\big)\,.$$

Here, $\Sigma_{\mathbf{s}} = (\mathbf{s}^T\otimes\mathbf{I})\boldsymbol{\Sigma}(\mathbf{s}\otimes\mathbf{I})$. This differs from the mismatched max-log demodulator in that the appropriate covariance matrix $\Sigma_{\mathbf{s}}+\sigma_w^2\mathbf{I}$ is used instead of $\mathbf{I}/\sigma_w^2$ and in that there is the additional log-det term that depends on the symbols.

Optimal soft-MMSE demodulator. In a similar spirit, [22] proposed an *optimal soft MMSE* demodulator given by

$$\tilde{\mathbf{s}}^{\text{MMSE}} = \mathcal{E}\{\mathbf{x}\mathbf{y}^H|\widehat{\mathbf{H}}\}\big(\mathcal{E}\{\mathbf{y}\mathbf{y}^H|\widehat{\mathbf{H}}\}\big)^{-1}\mathbf{y} = \widetilde{\mathbf{W}}\mathbf{y}\,. \tag{5.12}$$

Using (5.1) and (5.8), it is straightforward to show that the modified Wiener filter $\widetilde{\mathbf{W}}$ equals

$$\widetilde{\mathbf{W}} = \widehat{\mathbf{H}}_{\text{MMSE}}^H \Big(\widehat{\mathbf{H}}_{\text{MMSE}}\widehat{\mathbf{H}}_{\text{MMSE}}^H + \widetilde{\mathbf{\Sigma}} + \frac{\sigma_w^2}{E_s}\mathbf{I}\Big)^{-1}.$$

The matrix $\tilde{\mathbf{\Sigma}}$ equals the sum of $M_{\text{R}} \times M_{\text{R}}$ diagonal blocks of $\mathbf{\Sigma}$, i.e., $\widetilde{\mathbf{\Sigma}} = \sum_{l=1}^{M_{\text{T}}} \mathbf{\Sigma}_{l,l}$ with $\mathbf{\Sigma}_{l,l} = (\mathbf{e}_l^T \otimes \mathbf{I})\mathbf{\Sigma}(\mathbf{e}_l \otimes \mathbf{I})$ (here, $\mathbf{e}_l$ is the lth unit vector of length M_{T}). It is seen that the mismatched and optimal Wiener filter differ by $\widetilde{\mathbf{\Sigma}}$ which accounts for the additional "noise" caused by the channel estimation errors. The lth element of $\tilde{\mathbf{x}}^{\text{MMSE}}$ equals

$$\tilde{x}_l^{\text{MMSE}} = \tilde{\mu}_l x_l + z_l\,,$$

where $\tilde{\mu}_l = [\widetilde{\mathbf{W}}\widehat{\mathbf{H}}^{\text{MMSE}}]_{l,l}$ and z_l captures the residual interference whose power equals $\tilde{\sigma}_l^2 = \tilde{\mu}_l - \tilde{\mu}_l^2$. Assuming that the interference z_l is Gaussian, the *optimal MMSE* demodulator computes the per-layer LLRs

$$\tilde{\Lambda}_{l,i}^{\text{MMSE}} = \frac{1}{\tilde{\sigma}_l^2}\left[\min_{x\in\mathcal{A}_i^0} |\tilde{x}_l^{\text{MMSE}} - \tilde{\mu}_l x|^2 - \min_{x\in\mathcal{A}_i^1} |\tilde{x}_l^{\text{MMSE}} - \tilde{\mu}_l x|^2\right].$$

5.4 Factor Graph Interpretation of BICM Receivers

We can also derive the optimal max-log demodulators by means of the factor graph framework.

5.4.1 Derivation of Factor Graph

Our treatment builds on Section 2.6. We consider a MAP detector for the n'th information bit,

$$\hat{b}[n'] = \arg\max_{b[n']\in\{0,1\}} p(b[n']|\mathbf{y})\,,$$

where $\mathbf{y}$ denotes the sequence of received data symbols (cf. (5.2)) and the received pilot symbol sequences (cf. (5.3)). We consider the factorization (2.40),

$$p(b[n']|\mathbf{y}) \propto \sum_{\sim b[n']} p(\mathbf{y}|\mathbf{X})p(\mathbf{X}|\mathbf{c})p(\mathbf{c}|\mathbf{b})\,. \tag{5.13}$$

We can further develop the conditional density $p(\mathbf{y}|\mathbf{X})$ as

$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\mathbf{H},\mathbf{X})p(\mathbf{H})d\mathbf{H}\,,$$

where $\mathbf{H} = (\mathbf{H}[1]\cdots\mathbf{H}[N])$ denotes the sequence of channel matrices, and $p(\mathbf{H})$ denotes the prior density of the channel ($\mathbf{h} = \text{vec}\{\mathbf{H}\} \sim \mathcal{CN}(\mathbf{0}, \mathbf{C}_h)$). Assuming statistical independence, we have

$$p(\mathbf{y}|\mathbf{H},\mathbf{X}) \propto \prod_{n=1}^{N} p\big(\mathbf{y}[n]|\mathcal{X}[n],\mathbf{h}[n]\big)\, p\big(\tilde{\mathbf{y}}_p[n]|\mathbf{h}[n]\big)\,.$$

From the system model (5.2) we have $\mathbf{y}[n]|\mathcal{X}[n],\mathbf{h}[n] \sim \mathcal{CN}(\mathcal{X}[n]\mathbf{h}[n], \sigma_w^2\mathbf{I})$ and from (5.3) we obtain $\tilde{\mathbf{y}}_p[n]|\mathbf{h}[n] \sim \mathcal{CN}(\tilde{\mathcal{X}}_p\mathbf{h}[n], \sigma_w^2\mathbf{I})$. A segment of the factor graph representing the density in (5.13) is shown in Fig. 5.2.

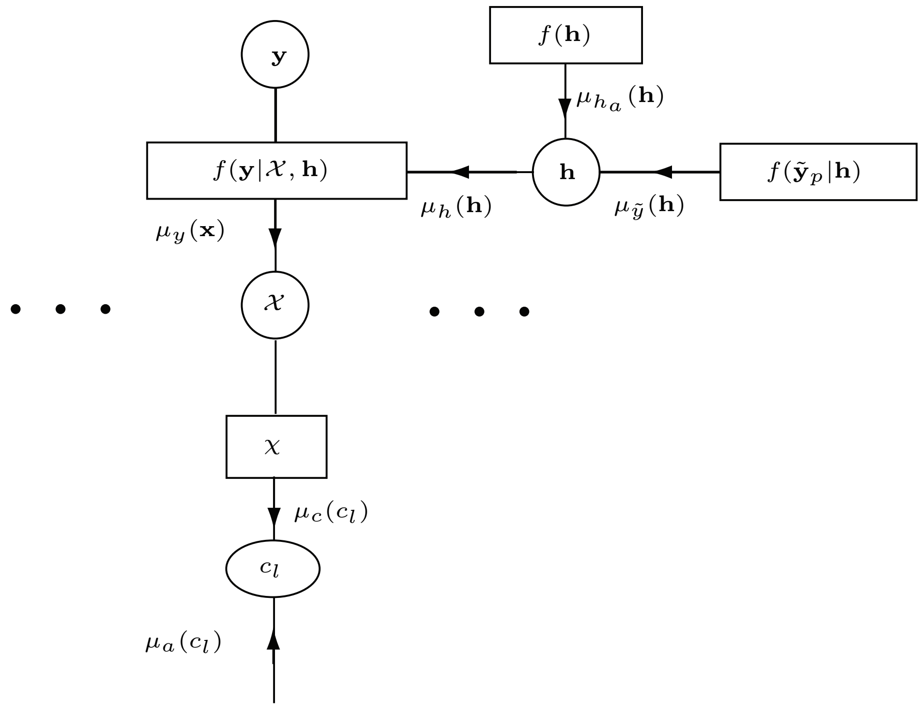


Figure 5.2: Factor graph representation of a BICM system with pilot based channel estimation.

5.4.2 Sum-Product Algorithm and Messages

In the following, we will omit the time index n for notational convenience. The message $\mu_a(\mathbf{h})$ from the *a priori* function block is Gaussian and is given by (cf. [5])

$$\mu_{h_a}(\mathbf{h}) = \mathcal{CN}(\mathbf{h}, \mathbf{0}, \mathbf{C}_h)\,,$$

and similarly we obtain for the message from the pilot blocks,

$$\mu_{\tilde{y}}(\mathbf{h}) = \mathcal{CN}\left(\mathbf{h}, \frac{1}{N_p P}\tilde{\mathcal{X}}_p^H \tilde{\mathbf{y}}_p, \frac{\sigma_w^2}{N_p P}\mathbf{I}\right).$$

The message $\mu_h(\mathbf{h})$ is then given by the product of the two messages $\mu_{h_a}(\mathbf{h})$ and $\mu_{\tilde{y}}(\mathbf{h}[k])$. It is Gaussian [5] and is given by

$$\mu_h(\mathbf{h}) = \mathcal{CN}\big(\mathbf{h}, \frac{1}{\sigma_w^2}\boldsymbol{\Sigma}\tilde{\mathcal{X}}_p^H \tilde{\mathbf{y}}, \boldsymbol{\Sigma}\big)\,.$$

This is exactely the density $f(\mathbf{h}|\hat{\mathbf{h}})$ of (5.8). Using the rules for message updates in the factor graph, we have

$$\mu_y(\mathbf{x}) = \int f(\mathbf{y}|\mathcal{X}, \mathbf{h})\, \mu_h(\mathbf{h})\, d\mathbf{h} = \mathcal{CN}(\mathbf{y}, \mathcal{X}\hat{\mathbf{h}}_{\text{MMSE}}, \mathcal{X}\mathbf{C}_h\mathcal{X}^H + \sigma_w^2\mathbf{I})\,,$$

which equals the density (5.11) derived for the modified demodulator. Now the messages $\mu_c(c_l[k])$ can be calculated according to (2.45) and (2.46) for the non-iterative and the iterative receiver, respectively. By calculating messages from the function node $f(\mathbf{y}[k]|\mathcal{X}[k], \mathbf{h}[k])$ to the variable node representing the channel, the channel estimate can be updated using information from other symbols by means of the message $\mu_a(c_l[k])$ as was done in Chapter 3. This leads to receivers with excellent performance but higher complexity and will not be considered here.

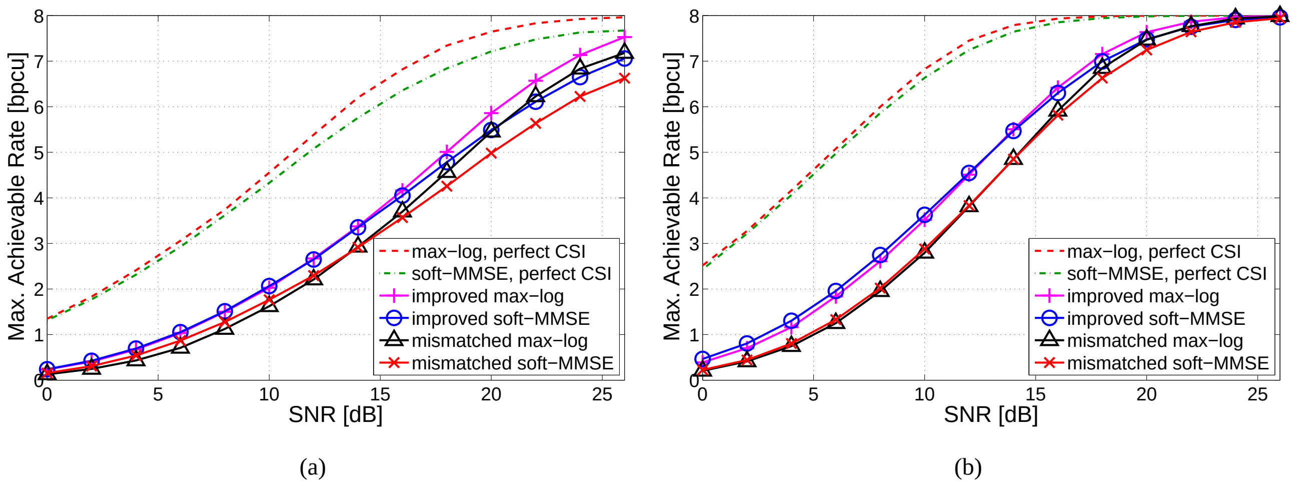


Figure 5.3: Demodulator comparison for (a) 2×2 and (b) 2×4 uncorrelated MIMO channel.

5.5 Numerical Results

In this section we present numerical results comparing the mismatched and the optimal demodulators. We first present results for BICM demodulators and then for BICM-ID.

5.5.1 Numerical Results for the BICM Demodulator

We consider an equivalent "modulation" channel (see Fig. 2.2) that comprises the space-time modulation, the actual fading channel, and the soft demodulator. The input of this equivalent channel is given by the interleaved code bits and its output is constituted by the (approximate) LLRs, generically denoted Λ (these LLRs are provided by the genie/mismatched/optimal max-log or MMSE demodulator). We adopt the approach from [20] which proposed to use the mutual information R of the equivalent modulation channel as a *code-independent* performance measure for MIMO soft demodulators. This mutual information can be interpreted as maximum rate that can be achieved with a given demodulator (in the sense of allowing asymptotically error-free communication). A mathematically precise justification of this interpretation was recently provided in [74].

For our setup, it can be shown that (recall that $R_0 = BM_\mathrm{T}$)

$$R = R_0 - \frac{1}{2}\sum_{k=1}^{BM_\mathrm{T}}\sum_{d_k=0}^{1}\int f(\hat{\Lambda}_k|d_k)\log_2\frac{2f(\hat{\Lambda})}{f(\hat{\Lambda}|d_k)}\,d\hat{\Lambda}_k, \tag{5.14}$$

where $f(\hat{\Lambda}) = \frac{1}{2}\sum_{d_k=0}^{1} f(\hat{\Lambda}_k|d_k)$. Analytical expressions for the conditional distributions $f(\Lambda_k|d_k)$ required for calculating the maximum achievable rate are unknown but for rare special cases. Hence, these distributions (and the capacity R) are generally determined numerically via Monte-Carlo simulations.

Capacity Results

Capacity for uncorrelated channel. Fig. 5.3 shows the capacity versus SNR for an i.i.d. MIMO channel, i.e., $\mathbf{C}_\mathbf{h} = \mathbf{I}$. For both the 2×2 and 2×4 system it is seen that both max-log and MMSE demodulation with

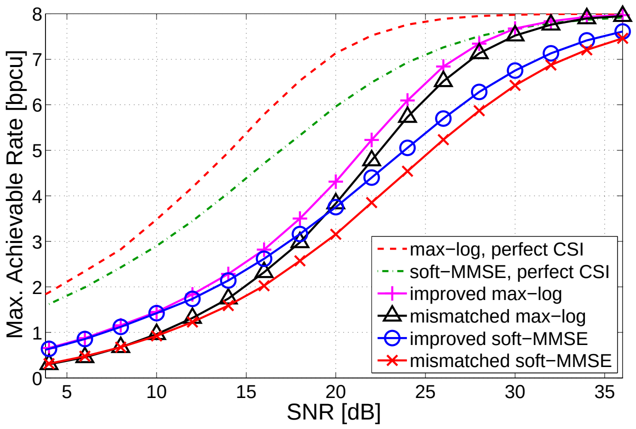


Figure 5.4: Demodulator comparison for 2 × 2 correlated MIMO channel.

imperfect CSI result in significant capacity losses compared to genie max-log and soft MMSE demodulation: at a rate of $R = 4$ bpcu, the SNR gap between the demodulators with perfect CSI and the demodulators with imperfect CSI equals 6 dB (with max-log) and 8 dB (with soft MMSE). However, optimal max-log and MMSE demodulation perform noticeably better than their mismatched counterparts. Specifically, for the 2×2 system shown in Fig. 5.3(a) the SNR gain of the optimal soft MMSE demodulator over mismatched MMSE ranges from 1 dB at $R = 2$ bpcu to about 1.5 dB at $R = 6$ bpcu. The SNR gain of the optimal max-log demodulator over mismatched max-log is about 1 dB for rates between $R = 2$ and $R = 6$ bpcu. For rates below $R \leq 4$ bpcu soft MMSE demodulation performs identically to or even better than max-log demodulation; at high rates, however, max-log is superior to MMSE demodulation.

Fig. 5.3(b) shows the results for the 2×4 system. Compared to the 2×2 case, all capacity curves are shifted to lower SNRs (by about 5 dB at $R = 4$ bpcu), despite the larger number of channel coefficients that have to be estimated (8 complex coefficients instead of 4). Apparently the larger number of receive antennas allows better spatial separation of the two data streams and outweighs the more difficult channel estimation. Max-log and MMSE demodulation (mismatched and optimal) perform almost identically in this scenario, with MMSE having a slight advantage a rates below 4 bpcu. The SNR gain of the optimal demodulators over their mismatched counterparts is about 1.5 dB at medium rates.

Capacity for correlated channel. We next consider the 2×2 system with a correlated MIMO channel that obeys the Kronecker model [34], i.e., $\mathbf{C_h} = \mathbf{T}^{1/2} \otimes \mathbf{R}^{1/2}$. The transmit and receive correlation matrices were chosen as

$$\mathbf{T} = \mathbf{R} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

with $\rho = 0.7$. Fig. 5.4 shows the capacity of the various demodulators versus SNR for this scenario. Compared to the uncorrelated case, all curves are shifted to the right by about 4 dB. Furthermore, max-log now outperforms soft MMSE for all rates above 0.5 bpcu: at $R = 4$ bpcu, the gap between optimal max-log and optimal soft MMSE is about 1.5 dB, and in case of the mismatched demodulators the gap is close to 2 dB. Furthermore, op-

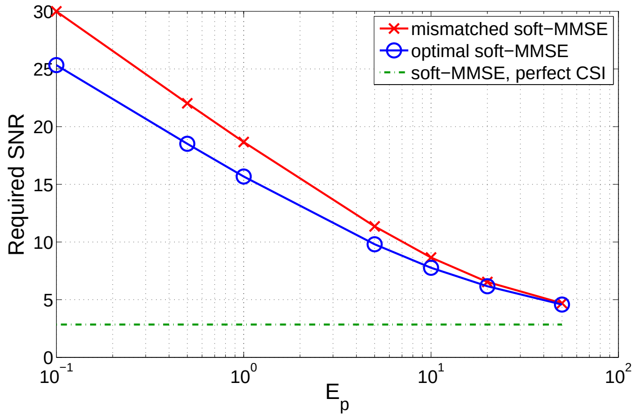


Figure 5.5: Required SNR for a target rate of $R = 4$ bpcu versus pilot power E_p of mismatched soft-MMSE and optimal soft-MMSE demodulator with 2×2 uncorrelated MIMO channel.

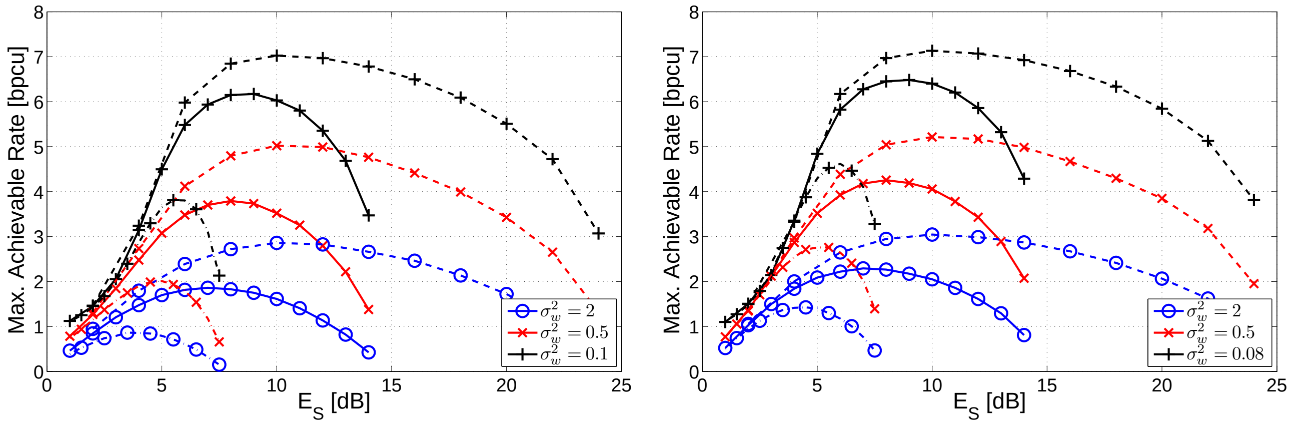
timal max-log and optimal soft MMSE gain 1 dB and 1.5 dB, respectively, over their mismatched counterparts. Apparently, the MMSE equalizer (5.12) performs worse in case of correlated channels.

Impact of Pilot Power

We next investigate the impact of the pilot power E_p on the performance of the mismatched and improved soft-MMSE demodulators. To this end we plot the required SNR to achieve a target rate of $R = 4$ bpcu versus the pilot power E_p in Fig. 5.5 for an uncorrelated 2×2 MIMO system. As a reference we also plot the required SNR for the same target rate in case of a demodulator having perfect CSI (which is of course independent of the pilot power). It can be observed that for very low pilot power, the gap between the mismatched and the optimal soft-MMSE demodulator can be up to 5 dB. In these scenarios the SNR gap to demodulators with perfect CSI is more than 20 dB. For larger pilot powers, the gap between mismatched and optimal demodulators decreases, as does the gap to the demodulator with perfect CSI.

Allocation of Pilot/Data Power

We next fix the *total* transmit power $E_{\text{tot}} = E_p + E_s$ and study how the allocation of power to pilots (i.e., E_p) and data symbols (i.e., E_s) impacts capacity. We reconsider the 2×2 system with uncorrelated MIMO channel. Fig. 5.6 shows the results obtained with mismatched (part (a)) and optimal (part (b)) soft MMSE demodulation for total power budgets of $E_{\text{tot}} = 8$ (dash-dotted line), $E_{\text{tot}} = 15$ (solid), $E_{\text{tot}} = 25$ (dashed), and for the three noise levels $\sigma_w^2 = 0.08$ (black '+'), $\sigma_w^2 = 0.5$ (red '×'), and $\sigma_w^2 = 2$ (blue '○'). It is seen that the power allocation has a strong impact on capacity: The capacity is very small for low pilot power (due to poor channel estimates) and for high pilot power (due to lack of resources for data transmissions). In between, there is an optimal choice of pilot power, roughly around $E_{\text{tot}}/2$. These results illustrate that improper power allocation can significantly deteriorate the overall performance.



(a)

(b)

Figure 5.6: Tradeoff between data andpilot symbol power. Demodulator comparison for (a) mismatched soft-MMSE, and (b) optimal soft-MMSE demodulator with 2×2 uncorrelated MIMO channel.

5.6 Numerical Results for the BICM-ID demodulator

In the following, we present numerical results for a 2×2 BICM system with 16QAM symbol alphabet (normalized to unit power) and $N_{\mathrm{p}}P = 0.4$. The number of coded bits per channel use is $L = 8$. We considered two different mappings: layer-wise Gray mapping and layer-wise m16a mapping [75], which is a mapping specifically optimized for BICM-ID with convolutional codes.

Demodulator EXIT Charts

The EXIT charts [24] of the demodulators were obtained by Monte Carlo simulations, using an AWGN channel for the a priori information. Fig. 5.6 shows the EXIT charts for the mismatched (with an LS channel estimator) and improved demodulator, both using Gray or m16a mapping. The SNRs have been chosen such that the area under the EXIT functions, which quantifies the maximum rate achievable with the respective demodulator [76], approximately equals $1/2$. With Gray mapping, the EXIT function of the mismatched demodulator at $10\,$dB SNR is almost identical to that of the improved detector at $9\,$dB. With m16a mapping, the EXIT functions of the two demodulators look quite different and the SNR required by mismatched and improved demodulation equals $11.75\,$dB and $9\,$dB, respectively. We conclude that the SNR threshold for the improved demodulator is identical under both mappings, even though different codes (matched to the respective EXIT function) are required to achieve this threshold. Furthermore, the SNR gain of the improved demodulator is about $1\,$dB for Gray mapping and $2.75\,$dB for m16a mapping.

Furthermore, codes designed for the mismatched demodulator with Gray mapping will also perform well for the improved detector with Gray mapping; however, with the improved demodulator the turbo cliff will occur at lower SNR. In contrast, with m16a mapping the pronounced difference between the EXIT functions of the mismatched and improved demodulator indicates that here the channel code should be matched to the respective demodulator used in order to avoid a large performance loss.

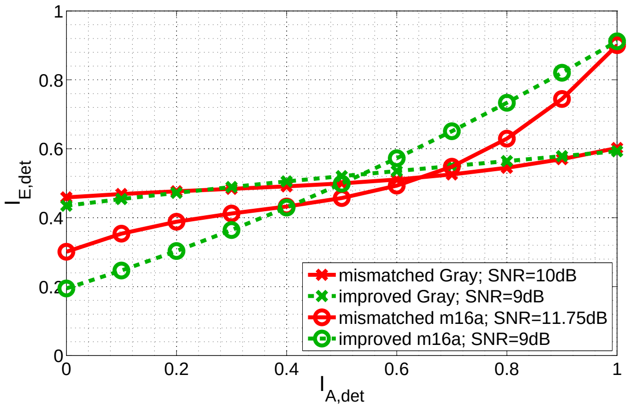


Figure 5.7: EXIT charts of mismatched and improved demodulator for Gray and m16a mapping.

Achievable Rates

A code-independent measure for the performance of the various demodulators with different mappings is the maximum rate they allow to achieve with vanishing error probability. This rate can be measured via the area under the demodulator's EXIT chart when the *a priori* channel is a binary erasure channel [76]. For reasons of numerical stability, we used an AWGN channel as *a priori* channel for obtaining the EXIT charts, in which case the area yields a good approximation for the achievable rates.

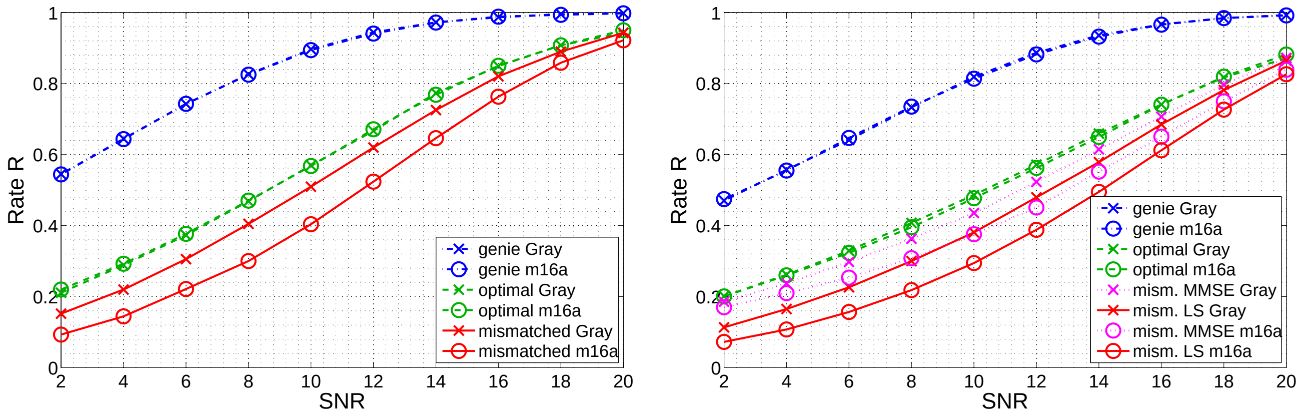
The resulting maximum rates achievable with the genie, mismatched, and improved demodulator are plotted versus SNR for a spatially uncorrelated Rayleigh fading channel in Fig. 5.8(a). It is seen that the improved demodulator is indeed superior to mismatched demodulation, even though there is still a significant gap to genie demodulation. For the genie and improved demodulator, the maximum rates are seen to be virtually the same for the two mappings used. Nevertheless, the corresponding EXIT charts (not shown) are different and achieving the maximum rates in an actual system thus requires matched code designs.

In contrast, for the mismatched demodulator the rate with m16a is lower than with Gray mapping. For a rate of $R = 1/2$, there is an SNR gap of about 2 dB between the mismatched demodulator with Gray and m16a. This indicates that the optimized m16a mapping is more sensitive to CSI inaccuracy than Gray mapping.

Fig. 5.8(b) shows similar results for the case of a channel with spatial correlation. We used a Kronecker model [34] for the correlation matrix of the channel, i.e., $\mathbf{C}_h = \mathbf{S}^{1/2} \otimes \mathbf{R}^{1/2}$, with the transmit and receive correlation matrices respectively chosen as

$$\mathbf{S} = \mathbf{R} = \begin{pmatrix} 1 & 0.7 \\ 0.7 & 1 \end{pmatrix}.$$

It is seen that the maximum achievable rates of the genie demodulator and modified demodulator are again almost independent of the mapping, albeit generaly smaller than in the uncorrelated case. The maximum rates achievable with the mismatched demodulator are shown for LS and MMSE channel estimation, both in conjunction with Gray and m16a mapping. Gray mapping is again preferable over m16a and in addition MMSE estimation is preferable over LS estimation due to its smaller channel estimation error. We conclude that m16a does not offer any advantage over Gray mapping in terms of maximum rates since the latter apparently is less sensitive to CSI inaccuracy.



(a)

(b)

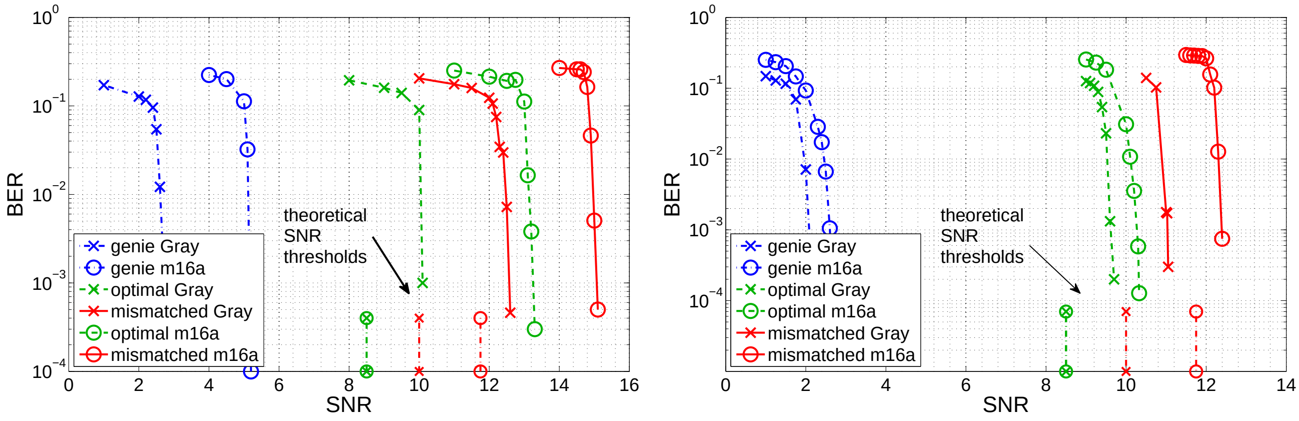
Figure 5.8: Rates achievable with genie, mismatched, and improved demodulator versus SNR for (a) uncorrelated Rayleigh fading and (b) correlated Rayleigh fading.

BER Performance

We next present bit error rate (BER) results for LDPC codes with $5 \cdot 10^4$ code bits and code rate $R=1/2$. With the same system parameters as before this amounts to 6250 transmit vectors. The MIMO channel was i.i.d. block fading, where the channel stays constant for 12 symbol periods (2 of which where used for training) and then a new, independent channel realization was drawn. The number of outer iterations (between demodulator and channel decoder) was 10, while the number of inner iterations (in the LDPC decoder) was 200. The mismatched demodulator was used with a LS channel estimator.

In Fig. 5.9(a), BER versus SNR for a non-optimized LDPC code with degree distribution $(3, 6)$ is shown together with the theoretical SNR thresholds. It is seen that there are significant gaps to the theoretical SNR thresholds, particularly for the genie and improved demodulator with m16a mapping. These gaps are caused by the mismatch between the EXIT functions of demodulator and code, which is significant when the m16a mapping is used. The EXIT charts of demodulators with Gray mapping are better matched to the code EXIT chart, therefore the gaps are smaller in this case. These BER results further confirm the superiority of the improved demodulator which outperforms the mismatched demodulator by about 2.4 dB (Gray) and 1.7 dB (m16a).

We further designed specifically optimized LDPC codes for each demodulator by matching the EXIT charts of the LDPC codes and of the demodulator according to [25]. The BER obtained with these optimized codes is plotted versus SNR in Fig. 5.9(b). All schemes are now much closer to the respective theoretical SNR thresholds. Furthermore, for genie and improved demodulation, Gray and m16a mapping now indeed feature approximately equal BER performance as predicted by Fig. 5.8(a) (m16a still performs slightly worse since here the code design does not achieve the theoretical optimum). The gain of improved demodulation over mismatched demodulation is about 1.5 dB (Gray mapping) and about 2 dB (m16a mapping).



(a)

(b)

Figure 5.9: BER versus SNR for MIMO-BICM-ID employing genie, mismatched, or improved demodulator and Gray or m16a mapping for (a) a non-optimized LDPC code and (b) optimized LDPC codes.

6

Conclusions and Outlook

FINALLY , we summarize the most important aspects and results of our work, present some conclusions, and provide suggestions for further research. In this thesis, we used the factor graph framework to design (possibly iterative) receiver algorithms. We first considered an IDMA system, derived a factor graph representation for it and used the factor graph to design a receiver performing joint data detection and channel estimation. Numerical results show that the proposed receiver operates close to the information theoretic limit and that joint data detection and channel estimation yields a dramatic performance gain.

We next considered BICM systems, where the demodulator calculates quantized LLRs. Using the information theoretic concept of the equivalent BICM modulation channel, we designed an optimal quantizer and also proposed a new quantization scheme, which is easier to design. We presented semi-analytical and numerical results, which show that only a few bits for LLR representation are needed. We also proposed an on-the-fly design of the quantizer parameters, thereby avoiding large lookup tables for storing the quantization parameters.

Finally, we focused on BICM systems with imperfect CSI at the receiver. We proposed optimal demodulator algorithms which take channel estimation errors into account, thereby offering improved performance. The improved demodulators can also be obtained by implementing the sum-product algorithm on the system's factor graph, thereby demonstrating the wide applicability of the factor graph approach. We demonstrated the performance gains possible with the proposed demodulators and also investigated the impact of allocation of pilot and data symbol power.

6.1 Conclusions

In the following we draw our conclusions based on the results presented in the previous chapters.

Graphical Models in Wireless Communications

- We proposed a receiver for a MIMO-OFDM-IDMA system that can be viewed as an approximation to the maximum *a posteriori* (MAP) bit detector. The MAP bit detector is optimal in the sense of minimum probability of a bit error; its output is chosen such that the *a posteriori* probability of an information bit is maximized. We obtained this probability by marginalization, which is carried out efficiently by means of the sum-product algorithm using the system's factor graph.

- The complexity of this marginalization is exponential in the number of users, thus making it unrealizable even for moderate system sizes. We overcame this limitation by using approximations for some of the messages propagated in the factor graph, thereby obtaining a complexity that is linear in the number of users. Our receiver extends that of [18] to MIMO transmissions.

- We extended the receiver of [18] to higher-order modulation formats, which results in a further increase of spectral efficiency (in addition to the MIMO multiplexing gain).

- We proposed a selective message update scheme where only certain messages are updated in each iteration. This yields a reduction of computational complexity and allows to trade error performance against computational efficiency in a flexible manner.

- Practical systems use pilot-aided channel estimation to obtain (imperfect) CSI. By using a factor graph based receiver, pilot-aided channel estimation becomes an integral part of the iterative scheme. Thus, the receiver performs iterative joint multiuser data detection and channel estimation for pilot-assisted MIMO-IDMA.

- Following the idea of [20], we considered the maximum achievable rate of IDMA as a fundamental performance limit and compared it with the information-theoretic capacity of the corresponding multiple access channel (MAC). To obtain the maximum achievable rate of IDMA we used the low-complexity IDMA receiver with an LDPC code as channel code and determined the achievable rate by means of BER simulations. It was shown that IDMA comes within 2 dB of the information theoretic limit.

- We used numerical simulations to demonstrate the performance gains achieved with the proposed receiver relative to conventional IDMA receivers and the dependence of performance on certain system parameters. It was observed, in particular, that inclusion of channel estimation in the iterative detection/decoding scheme yields a dramatic improvement of reliability, and selective message updating results in a significant reduction of complexity.

Message Quantization in BICM Systems

- We considered BICM systems with quantized LLRs and chose the quantization intervals by solving an optimization problem. Depending on the choice of the objective function we obtained an (information-theoretic) optimal quantizer or the proposed quantizer of [65].
- To compare the optimum and proposed quantizer design in a *code-independent* manner, we used the mutual information of the equivalent modulation channel of the BICM system [20]. In case of a fast-fading scenario, we used the ergodic rate, whereas for quasi-static fading we characterized the equivalent modulation channel by means of outage probability. In case of BPSK-modulated single-input single-output (SISO) systems a semi-analytical approach was employed, otherwise Monte-Carlo simulations.
- We provided a large number of numerical results for different system configuations and modulation alphabets. Our results demontrated, that only 3 bits are sufficient for quantizing LLRs with negligible loss in mutual information or outage probability. In case of low to medium rates, 2 bit quantization is sufficient.
- The quantizer parameters, namely quantizer intervals and quantizer outputs, depend on the system configuration and the SNR, requiring large lookup tables for storing the required parameters. We proposed a method for designing the proposed LLR quantizer during data transmission, “on the fly” i.e. without the need for precomputed lookup tables to store quantizer parameters.
- We provided bit error rate (BER) simulations for BICM systems with LLR quantization using *low-density parity-check (LDPC)* codes, which compare different quantization schemes. Our numerical results showed that the “on the fly” design of the quantizer intervals cause no loss in mutual information. The on-the-fly design of the quantizer outputs was demonstrated by BER simulations and performs as well as an offline design using precomputed lookup tables.
- By designing demodulators which directly calculate quantized LLRs, further complexity reductions are possible. We demonstrated these complexity savings for a low-complexity soft-MMSE demodulator. We also investigated the performance loss due to the use of a suboptimal demodulator and LLR quantization. For a wide range of scenarios, (quantized) soft-MMSE demodulators perform alsmost as well as the max-log demodulators, but with significantly lower complexity.

Performance of BICM Systems with Imperfect CSI

- We considered low-complexity soft-MMSE demodulators for BICM systems and the max-log demodulators for BICM-ID systems.
- The optimal BICM-ID demodulator of [23] was extended to MIMO channels with arbitrary spatial correlation. Furthermore, it was shown that the optimal demodulator is independent of the actual linear estimator used.

- We extended the results of the optimum soft-MMSE demodulator [22] to arbitrary linear estimators. Thereby we could demonstrate that the optimum soft-MMSE demodulator is independent of the actual linear channel estimator used.
- We proposed to use the maximum achievable rate with a specific demodulator as a code-independent performance metric. These rates were measured by Monte-Carlo simulations of the equivalent BICM modulation channel [20]. We compared these maximum achievable rates of mismatched and optimal low-complexity soft-MMSE receivers for correlated and uncorrelated MIMO channels. In case of BICM-ID, we used EXIT charts [24] to characterize the convergence behaviour of the MIMO-BICM-ID receivers employing the mismatched demodulator or the optimal demodulator for different symbol mappings and different channel correlation models.
- For BICM, we investigated the impact of power allocation between pilot and data symbol power on the maximum achievable rate and demonstrated the importance of correct power allocation. It turned out that this allocation is crucial for good performance.
- For BICM-ID, we used the approach from [25] to design LDPC codes that are matched to a specific demodulator in terms of their EXIT functions. We also provided BER comparisons using the optimized LDPC codes and a “standard” (i.e., non-optimized) LDPC code, thus showing the importance of matching LDPC codes to a specific demodulator.

6.2 Outlook

Several directions for future research seem to be interesting:

Graphical Models

- So far, we have only used Gaussian approximations in the proposed algorithms (especially in Chapter 3). While the Gaussian message approximations yield very good performance, a unified framework for message approximations would be desireable. Investigations of different message approximations, or whether performance can be improved by not approximating certain messages are interesting topics for further research.
- In almost all of the considered cases in this thesis, the factor graph has many cycles. In this case only very limited results on convergence properties for the sum-product algorithm are available and there are neither results about what the sum-product algorithm calculates (if it converges), either. Furthermore, many different schedulding schemes for the message updates are possible, and scheduling seems to have an impact on performance. Theoretical results on these issues would certainly help in designing efficient receivers using the factor graph approach.
- Other design approaches for iterative receivers have been proposed such as [77]. Also information geometry [16,17,78] has been used to gain insights into why and how iterative receivers obtain their impressive

performance. Using these theoretical methods, different strategies for iterative receiver design could be compared and further insights could be obtained.

Message Quantization

- At the end of Chapter 4 we presented a low-complexity soft-MMSE demodulator calculating quantized LLRs. Its performance is very close to the optimal demodulator in the low-rate regime, but in the high-rate regime, the max-log demodulator performs better. In practice, the max-log demodulator is implemented by means of the sphere-decoding algorithm [79]. Modifying the sphere-decoder such that it directly outputs quantized LLRs and investigating the tradeoff between quantizer bit-width and mean number of visited nodes in the sphere-decoding algorithm is a very interesting research topic.

- The channel decoder (either a belief propagation decoder for the LDPC code or the BCJR algorithm for a convolutional code) uses floating point variables for the decoding algorithm. In a practical communication system, however, an implementation with finite bit-width will be used, leading to a channel decoder working with quantized LLRs. It is an open research question, whether the quantization used for the LLRs can also be used in the channel decoder or whether a different number representation proves to be more advantageous. Further investigations into these issues seem to be promising.

- Message quantization can not only be used in BICM receivers, but can also be applied in distributed systems, like architectures using network coding or relaying schemes. In these systems, nodes exchange information which, for efficient and simple transmission, needs to be quantized. In the BICM systems considered in this thesis, the concept of the equivalent BICM modulation channel was used to design the optimal quantizer, but currently there are not many information theoretic results regarding distributed systems, thus making it impossible to design an optimal quantizer. Nevertheless, this is an interesting direction for future research.

BICM Receivers with Imperfect CSI

- While information theoretic bounds for systems with limited CSI exist [80, 81], no results are available on the capacity of coded modulation with imperfect CSI. Furthermore it is not clear how to measure the achievable rate of iterative demodulators. The EXIT chart approach provides some insights into the convergence process of iterative receivers and some approximations for achievable rates with low-complexity demodulators can be made, but exact results are still missing.

- In this thesis we used uncorrelated channel models and simple models for correlated channels. Using more realistic models for wireless channels or even testing the proposed demodulators with real channel data would allow a performance assessment under more realistic conditions. Furthermore, for the proposed demodulators, the channel statistics are required, which need be estimated by the receiver in a pracitcal system. Designing estimators for the channel statistics and assessing the performance these systems is an interesting research direction.

Bibliography

[1] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo: Morgan Kaufmann, 1988.

[2] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.

[3] N. Wiberg, *Codes and Decoding on General Graphs*. PhD thesis, Dept. of Electrical Engineering, Linköping, Sweden, 1996. Linköping studies in Science and Technology. Dissertation No. 440.

[4] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[5] H.-A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, and F. R. Kschischang, "The factor graph approach to model-based signal processing," *Proc. IEEE*, vol. 95, pp. 1295–1322, June 2007.

[6] S. M. Aji and R. J. McEliece, "A general algorithm for distributing information on a graph," in *Proc. IEEE ISIT-1997*, (Ulm, Germany), p. 6, 1997.

[7] C. Berrou, A. Glavieux, and P. Thitimajshime, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE ICC-1993*, (Geneva, Switzerland), pp. 1064–1070, May 1993.

[8] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. 20, pp. 284–287, March 1974.

[9] J. Hagenauer, "The turbo principle: Tutorial introduction and state of the art," in *Proc. 1st Int. Symp. on Turbo Codes and Related Topics*, (Brest, France), pp. 1–11, Sept. 1997.

[10] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inf. Theory*, vol. 44, pp. 927–945, May 1998.

[11] A. Chindapol and J. A. Ritcey, "Design, analysis, and performance evaluation for BICM-ID with square QAM constellations in Rayleigh fading channels," *IEEE J. Sel. Areas Comm.*, vol. 19, pp. 944–957, May 2001.

[12] N. Huaning, S. Manyuan, J. Ritcey, and L. Hui, "A factor graph approach to iterative channel estimation and LDPC decoding over fading channels," *IEEE Trans. Wireless Comm.*, vol. 4, pp. 1345–1350, July 2005.

[13] A. P. Worthen and W. E. Stark, "Unified design of iterative receivers using factor graphs," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 843–849, 2001.

[14] G. Colavolpe and G. Germi, "On the application of factor graphs and the sum-product algorithm to ISI channels," *IEEE Trans. Comm.*, vol. 53, pp. 818–825, May 2005.

[15] M. Wainwright, T. Jaakkola, and A. Willsky, "Tree-Based Reparameterization Framework for Analysis of Sum-Product and Related Algorithms," *IEEE Trans. Inf. Theory*, vol. 49, pp. 1120–1146, May 2003.

[16] S. Ikeda, T. Tanaka, and S. Amari, “Information geometry of turbo and low-density parity-check codes,” *IEEE Trans. Inf. Theory*, vol. 50, pp. 1097–1114, June 2004.

[17] S. Amari and H. Nagaoka, *Methods of Information Geometry*. New York: American Mathematical Society and Oxford University Press, 2000.

[18] L. Ping, L. Liu, K. Wu, and W. K. Leung, “Interleave-division multiple-access,” *IEEE Trans. Wireless Comm.*, vol. 5, pp. 938–947, Apr. 2006.

[19] M. C. Valenti and B. D. Woerner, “Iterative channel estimation and decoding of pilot symbol assisted turbo codes over flat-fading channels,” *IEEE J. Sel. Areas Comm.*, vol. 19, pp. 1697–1705, Sept. 2001.

[20] P. Fertl, J. Jaldén, and G. Matz, “Capacity-based performance comparison of MIMO-BICM demodulators,” in *Proc. IEEE SPAWC-08*, (Recife, Brazil), pp. 166–170, July 2008.

[21] W. Rave, “Quantization of log-likelihood ratios to maximize mutual information,” *IEEE Signal Processing Letters*, vol. 16, pp. 283–286, Apr. 2009.

[22] J. Wang, O. Wen, and S. Li, “Soft-Output MMSE MIMO detector under ML channel estimation and channel correlation,” *IEEE Signal Processing Letters*, vol. 16, pp. 667–670, Aug. 2009.

[23] S. Sadough, P. Piantanida, and P. Duhamel, “MIMO-OFDM optimal decoding and achievable information rates under imperfect channel estimation,” in *Proc. IEEE-SP Workshop on Signal Processing Advances in Wireless Communications*, pp. 1–5, 2007.

[24] S. ten Brink, “Convergence behavior of iteratively decoded parallel concatenated codes,” *IEEE Trans. Comm.*, vol. 49, pp. 1727–1737, Oct. 2001.

[25] G. Lechner, J. Sayir, and I. Land, “Optimization of LDPC Codes for Receiver Frontends,” *Proc. IEEE ISIT*, pp. 2388–2392, 2006.

[26] A. Goldsmith, *Wireless Communications*. Cambridge (UK): Cambridge Univ. Press, 2005.

[27] A. F. Molisch, ed., *Wideband Wireless Digital Communications*. Englewood Cliffs (NJ): Prentice Hall, 2001.

[28] I. E. Telatar, “Capacity of multi-antenna Gaussian channels,” *Europ. Trans. Telecomm.*, vol. 10, pp. 585–595, Nov./Dec. 1999.

[29] A. F. Molisch, ed., *Wireless Communications*. Chichester (UK): Wiley, 2005.

[30] L. Zheng and D. Tse, “Diversity and multiplexing: A fundamental tradeoff in multiple antenna channels,” *IEEE Trans. Inf. Theory*, vol. 49, pp. 1073–1096, May 2003.

[31] V. Tarokh, N. Seshadri, and A. R. Calderbank, “Space-time codes for high data rate wireless communications: Performance criterion and code construction,” *IEEE Trans. Inf. Theory*, vol. 44, pp. 744–765, March 1998.

[32] V. Tarokh, H. Jafarkhani, and R. A. Calderbank, “Space-time block codes from orthogonal designs,” *IEEE Trans.Inf.Theory*, vol. 45, pp. 1456–1467, July 1999.

[33] F. Oggier, G. Rekaya, J. Belfiore, and E. Viterbo, “Perfect space-time block codes,” *IEEE Trans. Inf. Theory*, vol. 52, pp. 3885–3902, Sept. 2006.

[34] C. Oestges, B. Clerckx, D. Vanhoenacker-Janvier, and A. Paulraj, “Impact of fading correlations on MIMO communication systems in geometry-based statistical channel models,” *IEEE Trans. Wireless Comm.*, vol. 4, pp. 1112–1120, May 2005.

[35] G. Ungerböck, “Channel coding with multilevel/phase signals,” *IEEE Trans. Inf. Theory*, vol. 28, pp. 56–67, Jan. 1982.

[36] C.-E. Sundberg and N. Seshadri, “Coded modulation for fading channels: An overview,” *European Trans. Telecomm.*, vol. 4, pp. 309–323, May/June 1993.

[37] X. Li and J. A. Ritcey, “Trellis-coded modulation with bit-interleaving and iterative decoding,” *IEEE J. Sel. Areas Comm.*, vol. 17, pp. 715–724, April 1999.

[38] S. H. Müller-Weinfurtner, “Coding approaches for multiple antenna transmission in fast fading and OFDM,” *IEEE Trans. Signal Processing*, vol. 50, pp. 2442–2450, Oct. 2002.

[39] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[40] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Boston (MA): Cambridge University Press, 2005.

[41] T. J. Richardson and R. L. Urbanke, “The capacity of low-density parity check codes under message-passing decoding,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, 2001.

[42] S. ten Brink, “Convergence of iterative decoding,” *Electronics Letters*, vol. 35, pp. 1117–1119, June 1999.

[43] T. Clevorn, S. Godtmann, and P. Vary, “BER prediction using EXIT charts for BICM with iterative decoding,” *IEEE Comm. Letters*, vol. 10, pp. 49 – 51, Jan 2006.

[44] S. B. Weinstein and P. M. Ebert, “Data transmission by frequency division multiplexing using the discrete Fourier transform,” *IEEE Trans. Comm. Technol.*, vol. 19, pp. 628–634, Oct. 1971.

[45] A. V. Oppenheim, *Signals and Systems*. Englewood Cliffs (NJ): Prentice Hall, 1983.

[46] S. Lin and D. J. Costello, *Error Control Coding*. Upper Saddle River (NJ): Prentice Hall, 2004.

[47] R. G. Gallager, “Low density parity check codes,” *IRE Trans. Info. Theory*, vol. 8, pp. 21–28, Jan. 1962.

[48] D. J. C. MacKay, “Good error correcting codes based on very sparse matrices,” *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, 1999.

[49] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, “Design of capacity-approaching irregular low-density parity check codes,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, 2001.

[50] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[51] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs (NJ): Prentice Hall, 1993.

[52] X. Wang and H. V. Poor, “Iterative (turbo) soft interference cancellation and decoding for coded CDMA,” *IEEE Trans. Comm.*, vol. 47, pp. 1046–1061, July 1999.

[53] S. Verdú, *Multiuser Detection*. Cambridge (UK): Cambridge Univ. Press, 1998.

[54] C. Novak, F. Hlawatsch, and G. Matz, “MIMO-IDMA: Uplink multiuser MIMO communications using interleave-division multiple access and low-complexity iterative receivers,” in *Proc. IEEE ICASSP-2007*, (Honolulu, Hawaii, USA), pp. 225–228, April 2007.

[55] I. Mahafeno, C. Langlais, and C. Jego, “OFDM-IDMA versus IDMA with ISI cancellation for quasi-static Rayleigh fading multipath channels,” in *Proc. 4th Int. Symp. on Turbo Codes & Related Topics*, (Munich, Germany), pp. 3–7, 2006.

[56] L. Ping, Q. Guo, and J. Tong, “The OFDM-IDMA approach to wireless communication systems,” *IEEE Wireless Communications*, vol. 14, no. 3, p. 18, 2007.

[57] C. Novak, G. Matz, and F. Hlawatsch, “A factor graph approach to joint iterative data detection and channel estimation in pilot-assisted IDMA transmissions,” in *Proc. IEEE ICASSP-2008*, (Las Vegas, NV), pp. 2697–2700, Apr. 2008.

[58] C. Novak, F. Hlawatsch, and G. Matz, “Low-complexity factor graph receivers for spectrally efficient MIMO-IDMA,” in *Proc. IEEE ICC-2008*, (Beijing, China), pp. 770–774, 2008.

[59] C. Novak, G. Matz, and Hlawatsch, "Factor graph based design of an OFDM-IDMA receiver performing joint data detection, channel estimation, and channel length selection," in *Proc. IEEE ICASSP-2009*, (Taipei, Taiwan), 2009.

[60] J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 3rd ed., 1995.

[61] J. Boutros and G. Caire, "Iterative multiuser joint decoding: Unified framework and asymptotic analysis," *IEEE Trans. Inf. Theory*, vol. 48, pp. 1772–1793, July 2002.

[62] A. Sanderovich, M. Peleg, and S. Shamai (Shitz), "LDPC coded MIMO multiple access with iterative joint decoding," *IEEE Trans. Inf. Theory*, vol. 51, pp. 1437–1450, Apr. 2005.

[63] A. A. Stephan ten Brink, Gerhard Kramer, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Comm.*, vol. 52, pp. 670–678, Apr 2004.

[64] S. Schwandter and G. Matz, "A practical forwarding scheme for wireless relay channels based on the quantization of log-likelihood ratios," in *Proc. IEEE ICASSP*, (Dallas, Texas), pp. 2502–2505, March 2010.

[65] C. Novak, P. Fertl, and G. Matz, "Quantization for soft-output demodulators in bit-interleaved coded modulation systems," in *Proc. IEEE ISIT-2009*, (Seoul, Korea), pp. 1070–1074, 2009.

[66] S. Schwandter, P. Fertl, Clemens Novak, and G. Matz, "Log-likelihood ratio clipping in MIMO-BICM systems: Information geometric analysis and impact on system capacity," in *Proc. IEEE ICASSP-2009*, pp. 2433–2436, 2009.

[67] M. K. Simon, *Probability Distributions Involving Gaussian Random Variables: A Handbook for Engineers and Scientists*. New York: Springer, 2002.

[68] D. Seethaler, G. Matz, and F. Hlawatsch, "An efficient MMSE-based demodulator for MIMO bit-interleaved coded modulation," in *Proc. IEEE Globecom 2004*, vol. IV, (Dallas, Texas), pp. 2455–2459, Dec. 2004.

[69] M. Benjillali, L. Szczecinski, and S. Aissa, "Probability density functions of logarithmic likelihood ratios in rectangular QAM," in *23rd Biennial Symposium on Communications*, (Kingston, Canada), pp. 283–286, May 2006.

[70] Y. Huang and J. Ritcey, "EXIT chart analysis of BICM-ID with imperfect channel state information," *IEEE Comm. Letters*, vol. 7, pp. 434–436, Sept. 2003.

[71] T. Clevorn, S. Godtmann, and P. Vary, "BER prediction using EXIT charts for BICM with iterative decoding," *IEEE Comm. Letters*, vol. 10, pp. 49–51, Jan. 2006.

[72] G. Taricco and E. Biglieri, "Space-time decoding with imperfect channel estimation," *IEEE Trans. Wireless Comm.*, vol. 4, no. 4, pp. 1874–1888, 2005.

[73] C. Novak, G. Lechner, and G. Matz, "MIMO-BICM with imperfect channel state information: EXIT chart analysis and LDPC code optimiaztion," in *Proc. Asilomar Conf. Signals, Systems, Computers*, (Pacific Grove, CA), pp. 443–447, Oct.–Nov. 2008.

[74] J. Jaldén, P. Fertl, and G. Matz, "On the generalized mutual information of BICM Systems with approximate demodulation," in *Proc. IEEE ITW 2010*, (Cairo, Egypt), Jan. 2010.

[75] F. Schreckenbach, N. Görtz, J. Hagenauer, and G. Bauch, "Optimization of symbol mappings for bit-interleaved coded modulation with iterative decoding," *IEEE Comm. Letters*, vol. 7, pp. 593–595, Dec. 2003.

[76] A. Ashikhmin, G. Kramer, and S. Brink, "Extrinsic information transfer functions: model and erasure channel properties," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2657–2673, 2004.

[77] B. Hu, I. Land, L. Rasmussen, R. Piton, and B. Fleury, "A divergence minimization approach to joint multiuser decoding for coded CDMA," *IEEE J. Sel. Areas Comm.*, vol. 26, p. 432, March 2008.

[78] S. Ikeda, T. Tanaka, and S. Amari, "Belief propagation and turbo code: Information geometrical view," in *Proc. 8th Int. Conf. Neural Info. Proc.*, (Shanghai, China), pp. 41–46, Nov. 2001.

[79] C. Studer, A. Burg, and H. Bölcskei, "Soft-output sphere decoding: Algorithms and VLSI implementation." submitted to *IEEE J. Sel. Areas Comm.*, Apr. 2007.

[80] M. Médard, “The effect upon channel capacity in wireless communications of perfect and imperfect knowledge of the channel,” *IEEE Trans. Inf. Theory*, vol. 46, pp. 933–946, May 2000.

[81] A. Lapidoth and S. Shamai, “Fading channels: How perfect need “perfect side information” be?,” *IEEE Trans. Inf. Theory*, vol. 48, pp. 1118–1134, May 2002.