Technische Universität München
TUM School of Computation, Information and Technology

Polarizing Codes for Higher-Order Modulation and Direct Detection Channels

Tobias Prinz

Vollständiger Abdruck der von der TUM School of Computation, Information and Technology der Technischen Universität München zur Erlangung eines

Doktors der Ingenieurwissenschaften (Dr.-Ing.)

genehmigten Dissertation.

Vorsitz:	Prof. Dr.-Ing. Wolfgang Kellerer
Prüfer der Dissertation:	1. Prof. Dr. sc. techn. Gerhard Kramer
	2. Prof. Dr.-Ing. Stephan ten Brink

Die Dissertation wurde am 03.04.2025 bei der Technischen Universität München eingereicht und durch die TUM School of Computation, Information and Technology am 15.07.2025 angenommen.

Acknowledgment

This doctoral thesis is based on my work at the Institute for Communication Engineering (LNT) at the Technical University of Munich (TUM) and would not have been possible without the support of many people. I would like to express my gratitude to all these people.

First and foremost, I would like to thank my doctoral advisor Gerhard Kramer for his support and guidance throughout my time at the institute. I am grateful for the freedom he gave me in choosing and working on various topics. The trust in my independence allowed me to grow both professionally and personally. Whenever input was needed, he had plenty of ideas and perspectives that were always insightful and helpful. I also want to thank for the flexibility that allowed me to balance my work with my private life and family. Furthermore, I am grateful for being involved in many teaching activities. This provided me the chance to discover how much joy teaching and sharing knowledge with students brings.

Next, I would like to thank all the colleagues at the institute that supported my work. At the beginning of my time at the institute, Georg Böcherer gave me a lot of input about coded modulation and probabilistic shaping. Together with my coding mates Fabian Steiner, Peihong Yuan and Thomas Wiegart, I explored various aspects of coding and I still remember the spirit when optimizing all of our codes for the Heinrich Hecht award. Special thanks go to my colleague Daniel Plabst for the collaborative work on Direct Detection Channels. Together, we spent countless hours at the whiteboard, working through every aspect of the problem and brainstorming solutions. The teamwork was really amazing.

I want to thank all of my colleagues for creating a friendly and relaxed atmosphere at the institute throughout my time here. I remember a lot of precious moments including playing table football, making and drinking coffee, cooking ”Allgäuer Käsespätzle“ and many more.

Last but not least, I would like to thank my family for their ongoing support. Thanks to my parents and my father in law for helping us with our two children whenever I needed some time for writing my thesis. The biggest thanks go to my wife, Anna, for her unwavering support and, most of all, for constantly encouraging me. I am glad for how we managed, together, to care for our two children, founding an orchestra and still being able for both of us to pursue our professional paths. It wouldn’t have been possible without you, and I am thankful for everything we’ve achieved together.

Munich, October 2025 Tobias Prinz

Contents

Abstract

This thesis studies polarizing codes for higher-order modulation and direct detection (DD) channels. Two probabilistic shaping schemes based on polarizing codes are developed. The first scheme combines probabilistic amplitude shaping (PAS) with polar coded modulation (PCM) that is based on multilevel coding (MLC) and multistage decoding (MSD). This method is called polar coded PAS (PC-PAS). To improve performance, a multistage list decoding algorithm is introduced that takes advantage of the PCM structure. Moreover, when using a constant composition distribution matcher (DM) for PAS, a type check at the receiver further improves performance, especially for short block lengths. The second scheme uses a polarizing code, namely a polar code or a branching multi-scale entanglement renormalization ansatz (BMERA) code, to perform probabilistic shaping and forward error correction (FEC) simultaneously.

PAS is applied to transmit 6-ary pulse amplitude modulation (6-PAM) for optical communication systems with intensity modulation (IM) and peak power constrained DD channels. PAS is used to map bits to symbols from the 6-ary alphabet. Alternatively, two-dimensional thirty-two quadrature amplitude modulation (32-QAM) constellations are used and mapped to one-dimensional 6-PAM constellations where the shape of the 32-QAM constellation is optimized. Simulations and optical experiments show that the new constellations significantly outperform existing methods.

PCM is also used to improve communication over bandlimited channels with DD. For such channels, the transmitter should use bipolar or complex-valued constellations and the receiver should sample faster than the symbol rate. Practical receiver architectures based on separate detection and decoding (SDD) and successive interference cancellation (SIC) are developed that approach the information rates of joint detection and decoding (JDD). Two different detector types are studied: the forward-backward algorithm, which computes the a posteriori probabilities (APPs), and a Gibbs sampling detector, which approximates the APPs. Simulations with multilevel PCM and multistage detection and decoding confirm that bipolar and complex-valued constellations achieve higher rates than IM.

Zusammenfassung

Diese Arbeit beschäftigt sich mit polarisierenden Codes für höherwertige Modulationsformate und *Direct Detection* (DD) Kanäle. Zwei Methoden zur Wahrscheinlichkeitsanpassung mit polarisierenden Codes werden entwickelt. Die erste Methode kombiniert *Probabilistic Amplitude Shaping* (PAS) mit *Polar Coded Modulation* (PCM). PCM basiert auf einer mehrstufigen Codierung (MLC) und einer mehrstufigen Decodierung (MSD). Die Kombination aus PAS und PCM wird PC-PAS genannt. Ein mehrstufiger Listen-Algorithmus zur Decodierung nutzt die mehrstufige Struktur von PCM und verbessert die Fehlerraten dieser Methode. Zudem kann bei der Nutzung eines *Constant Composition Distribution Matchers* (CCDM) bei PC-PAS ein *Type Check* die Fehlerraten weiter reduzieren, vor allem bei der Übertragung von kurzen Blöcken. Bei der zweiten Methode werden sowohl die Wahrscheinlichkeitsanpassung als auch die Vorwärtsfehlerkorrektur gleichzeitig von einem polarisierenden Code realisiert. Diese Methode wird entweder durch *Polar Codes* oder durch *Branching Multi-Scale Entanglement Renormalization Ansatz* (BMERA) Codes realisiert.

PAS ermöglicht die Übertragung von sechsstufiger Pulsamplitudenmodulation (6-PAM) in optischen Kommunikationssystemen mit Intensitätsmodulation (IM) und DD Empfängern mit einer Begrenzung der maximalen Sendeleistung. PAS bildet dabei Bits auf die sechsstufige Konstellation ab. Alternativ dazu wird eine zweidimensionale 32-stufige Quadraturamplitudenmodulation (32-QAM) verwendet, welche dann auf eine eindimensionale 6-PAM Konstellation abgebildet wird. Die Form der 32-QAM Konstellationen wird dabei optimiert. Simulationen und optische Experimente zeigen, dass die neuen Konstellationen bisher bestehende Methoden übertreffen.

PCM wird auch für die Verbesserung der Übertragungsraten bei bandbegrenzten DD Kanälen verwendet. Für derartige Kanäle sollten bipolare und komplexwertige Konstellationen verwendet werden. Dabei muss am Empfänger schneller als mit der Symbolrate abgetastet werden. Es werden praktische Empfängerstrukturen mit getrennter Detektion und Decodierung (SDD) und sukzessiver Interferenzauslöschung entwickelt. Die gezeigten Empfänger nähern sich den Raten an, die mit gleichzeitiger Detektion und Decodierung möglich wären. Zwei unterschiedliche Detektoren werden untersucht: Der Vorwärts-Rückwärts Algorithmus, der die *A Posteriori* Wahrscheinlichkeiten (APPs) berechnet und ein *Gibbs Sampling* Detektor, der die APPs näherungsweise bestimmt. Simulationen mit mehrstufiger PCM und mehrstufiger Detektion und Decodierung bestätigen, dass bipolare und komplexwertige Konstellationen höhere Raten ermöglichen als IM.

1 Introduction

The invention of polar codes and the discovery of the polarization principle in [1] are two major breakthroughs in coding theory. Polar codes have been included in the 5G new radio standard [2] within less than a decade after they were published. Channel polarization occurs under successive cancellation (SC) decoding, where the codeword bits are successively decoded based on previously decoded bits. A hard problem is thereby divided into easier problems that are solved successively.

This simple principle is used repeatedly in this thesis. An important example is coded modulation with multilevel coding (MLC) [3–5] which extends polar codes [6, 7] to higher-order alphabets. The code is divided into several stages or levels representing bits that label the transmitted symbols. Multistage decoding (MSD) successively decodes the bits of a stage based on the previously decoded stages.

Successive interference cancellation (SIC) is another example that exhibits the idea. SIC is known in multi-user scenarios, where signals from multiple users interfere [8, 9]. A receiver decodes the signals successively, treating the unknown signals as noise and subtracting the decoded signals from the received signals. Usually, the receiver starts with the strongest signal which is easiest to decode. In fact, SIC can also be used to treat intersymbol interference (ISI) where symbols within one sequence interfere with each other [10–12].

To simplify the receiver, we harness a method from sampling theory called Gibbs sampling [13]. It is often difficult to draw samples from a high-dimensional probability density function (PDF). Gibbs sampling addresses this problem by sampling from the marginal conditional distributions such that many properties of the high-dimensional density function are preserved. The algorithm conditions on previously drawn samples from the other dimensions.

1.1. Related Work

The idea of polar codes decoded with SC decoding appeared in [14]. In [1], Arıkan proved that polar codes achieve the symmetric capacity of discrete memoryless channels (DMCs). The performance for reasonable block lengths under SC decoding was poor, however. Tal and Vardy improved the performance by introducing successive cancellation list (SCL) decoding with outer cyclic redundancy check (CRC) codes [15]. Polar coded modulation (PCM) is discussed in [6, 7, 16] as the extension of binary polar codes to coded modulation with a multilevel structure. Similar schemes are presented in [17, 18].

The paper [19] shows that channel polarization is a general phenomenon that is not limited to the polar transform. For example, an alternative transform inspired by quantum many-body physics is presented in [20, 21] where it is called the branching multi-scale entanglement renormalization ansatz (BMERA) transform. The resulting BMERA codes have a better error exponent than polar codes [22]; the distance properties and weight spectra of BMERA codes are discussed in [23–25]. SC decoders for BMERA codes are presented in [26–28].

Polarizing codes are also useful for shaping. Honda and Yamamoto [29] introduced a scheme to transmit non-uniformly distributed symbols on binary input channels based on polar codes. This idea is taken up in [30] to transmit shaped on-off keying (OOK), and in [31] where one polar code is used for distribution matching and one for forward error correction (FEC). Extensions to higher-order modulation can be found in [32–37]. In [37], it is proved that a multilevel scheme based on the Honda-Yamamoto scheme achieves the capacity of DMCs. Further examples of probabilistic shaping are Gallager's many-to-one mappings [38], trellis shaping [39], and probabilistic amplitude shaping (PAS) [40].

PAS was used with low-density parity-check (LDPC) codes in fiber-optic communication [41]. Early fiber-optic communication systems were based on direct detection (DD) receivers [42], where a photodiode (PD) measures the amplitude or intensity of the received signal. DD receivers are usually paired with intensity modulation (IM) because the transmitted data can be recovered by DD with symbol-rate sampling [43–45]. Systems with IM/DD result in cheap hardware, low power consumption, low latency and reduce hardware complexity and form factors as compared to coherent receivers [42,46,47]. Today, IM/DD is still used in short-reach optical links, e.g., in intra-data-center communication up to several dozens of kilometers [42]. Usually, unipolar modulation with several amplitude levels, i.e., M-ary pulse amplitude modulation (PAM) with $M \geq 2$, is used. The receiver applies classic signal processing such as linear equalization [48] or modern machine learning methods [49, 50].

Short-reach systems often operate without optical amplifiers, and the optical input power to the fiber is limited by the maximum laser power, resulting in a peak power constraint [51–53]. The capacity-achieving distribution of an additive white Gaussian noise (AWGN) channel under a peak power constraint is discrete [54] and there is an optimal number of signal points for each signal-to-noise ratio (SNR), i.e., there are SNR regions where the optimal number of signal points is not a power of two. Therefore, [55, 56] suggest transmitting 6-ary alphabets based on 32-quadrature amplitude modulation (QAM) constellations.

It is known that oversampling at the receiver of a DD system increases capacity [57] and that DD systems can operate within 1 bit/s/Hz of the capacity with coherent detection. This suggests using bipolar or even complex-valued constellations for DD [57–61]. The first example of a DD system not restricted to IM used a Kramers-Kronig receiver that required minimum phase signals obtained by inserting an optical carrier. However, the optical carrier reduces the spectral efficiency (SE) [58, 62, 63]. The paper [64] shows how rates for channels with ISI can be computed based on the forward-backward algorithm (FBA) [65]. There are many examples where the FBA is replaced by Gibbs sampling [13]. For example, Gibbs sampling was applied to code-division multiple access [66, 67], multi-input multi-output systems [68, 69], and channels with ISI [70–73]. Bit-wise Gibbs sampling for SIC is described in [74, 75].

1.2. Contributions

This section lists the main contributions of the thesis. Most results appear in [28, 76–81]

We combine polar codes with PAS to obtain polar coded PAS (PC-PAS). PAS was originally described for bit-metric decoding (BMD) and the extension to MSD for polar codes requires modifications. We encode based on dynamically frozen bits to perform systematic encoding for PAS. We propose a multistage list decoding algorithm for PCM and PC-PAS, where the list of possible codewords is passed across decoding stages. If PC-PAS is used with a constant composition distribution matcher (CCDM), we propose a type check in addition to a CRC check during decoding to compensate for the CCDM rate loss.

We describe SC decoding of BMERA codes and provide results for BMERA coded modulation. Further, we use BMERA codes instead of polar codes in the shaping schemes from [30] and [82], and compare the results with those of polar codes.

We suggest transmitting 6-PAM via PAS or by other 32-QAM constellations as in [55] that are better suited for peak power constrained channels. The solutions are supported by information rate comparisons and simulations with FEC. We conduct optical experiments with DD to confirm the performance.

We study bandlimited optical channels with DD and oversampling at the receiver. We show that bipolar and complex-valued signals can be detected by DD receivers. The information rates are compared to practical receivers that use separate detection and decoding (SDD). SIC based on SDD is shown to approach the joint detection and decoding (JDD) rates. Different SIC receivers based on symbol or bit perspectives are presented. The FBA is used as a detector and Gibbs sampling is used to reduce complexity. Coded results with multilevel polar codes demonstrate the feasibility of SIC.

1.3. Outline

The thesis is organized as follows.

- ▷ Chapter 2 introduces notation, channel models, and modulation formats. The principles of coded modulation with BMD and MSD are reviewed. PAS is also reviewed; see [40].
- ▷ Chapter 3 reviews the basics of polar and BMERA codes, including the polar and BMERA transforms, the channel polarization principle, encoding and decoding, and design methods.
- ▷ Chapter 4 extends polar and BMERA codes to polar and BMERA coded modulation, i.e., how polar and BMERA codes can be used to transmit symbols of constellations with more than two symbols. Design methods based on surrogate channels are introduced and a multistage list decoding algorithm is presented.
- ▷ Chapter 5 combines PCM with PAS to obtain PC-PAS. It is shown how PC-PAS can be used with bipolar amplitude shift keying (ASK) and unipolar PAM constellations. The results are compared to PAS with LDPC codes from the 5G new radio standard [2].
- ▷ Chapter 6 reviews polar shaping based on [29] and provides results for OOK, ASK, and PAM transmission using BMERA codes.
- ▷ Chapter 7 discusses transmission of symbols from 6-ary alphabets over peak power constrained channels. Two schemes are introduced and compared, one based on PAS and the other on 32-QAM constellations.
- ▷ Chapter 8 studies SIC for bandlimited optical channels with DD. We discuss different detectors based on symbols and bits and their information rates. Detectors based on the FBA and Gibbs sampling are proposed.
- ▷ Chapter 9 concludes the thesis.

2

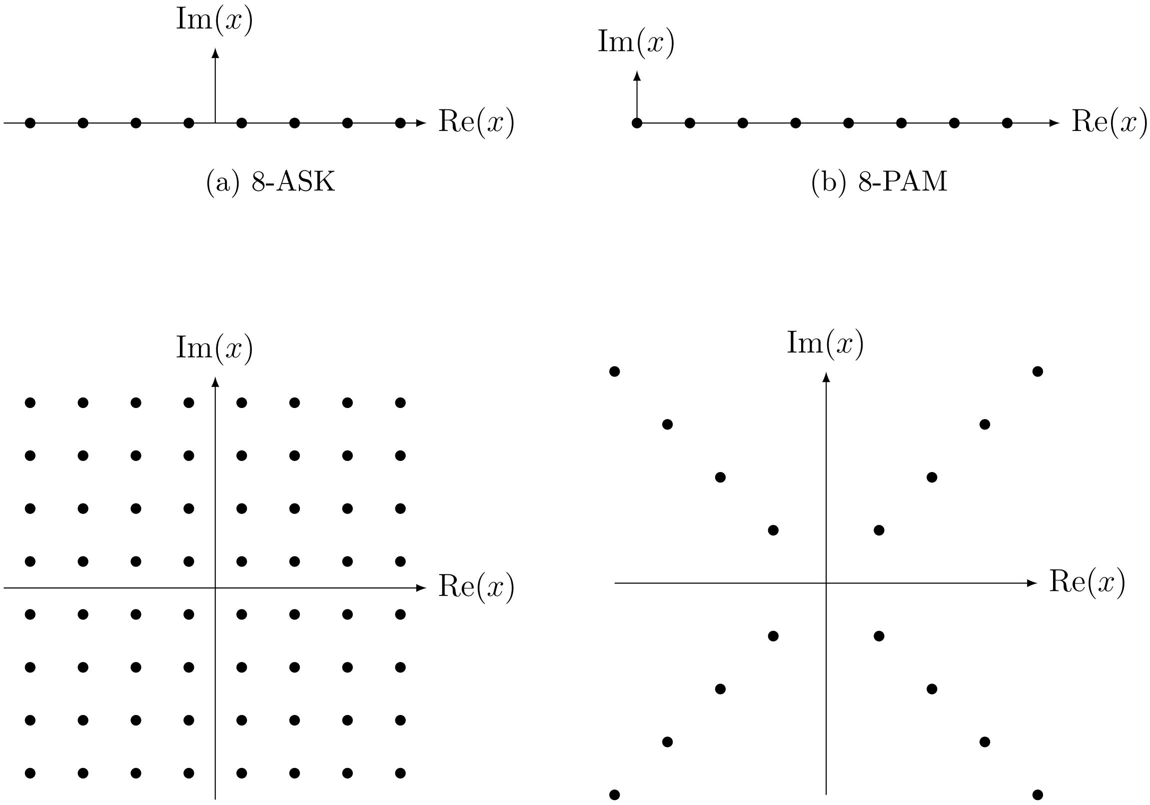
Preliminaries

2.1. Notation and Definitions

Vectors are written as lowercase italic bold letters $\boldsymbol{a}$ and matrices are written as uppercase bold letters $\mathbf{A}$. If not stated otherwise, a vector $\boldsymbol{a}$ has dimension n. A vector with indices from k to l is denoted $a_k^l = (a_k, a_{k+1}, \dots, a_l)$ and a vector with indices from 1 to k is denoted $a_1^k = a^k = (a_1, a_2, \dots, a_k)$. The same notation is used for a vector of vectors, e.g., $\boldsymbol{a}_k^l = (\boldsymbol{a}_k, \dots, \boldsymbol{a}_l)$. The vector $\boldsymbol{a}_{[k]} = (a_1, \dots, a_{k-1}, a_{k+1}, \dots, a_n)$ has the entry a_k removed. The transpose of a vector or matrix is written as $\boldsymbol{a}^{\mathrm{T}}$ and $\mathbf{A}^{\mathrm{T}}$, respectively. The $n \times 1$ all-zeros vector is $\mathbf{0}_n$ and the $n \times n$ identity matrix is $\mathbf{I}_n$. The diagonal matrix with entries taken from $\boldsymbol{a}$ is written as $\mathrm{diag}(\boldsymbol{a})$. The notation $\boldsymbol{a} \otimes \boldsymbol{b}$ refers to the Kronecker product of $\boldsymbol{a}$ and $\boldsymbol{b}$. Consider the sets of indices denoted as $\mathcal{A}$ and $\mathcal{B}$. The matrix $\mathbf{A}_{\mathcal{A}}$ consists of the rows of $\mathbf{A}$ chosen according to $\mathcal{A}$ and the matrix $\mathbf{A}_{\mathcal{A},\mathcal{B}}$ consists of the rows and columns of $\mathbf{A}$ chosen according to $\mathcal{A}$ and $\mathcal{B}$, respectively. $\mathcal{A} \setminus \mathcal{B}$ denotes the relative complement of set $\mathcal{B}$ to $\mathcal{A}$. We define element-wise absolute values and squaring by $|\boldsymbol{x}|^{\circ 2} = [|x_1|^2, \dots, |x_n|^2]$. Additions and multiplications related to bits are always modulo 2. The addition $a \oplus b$ explicitly denotes modulo-2 additions.

Random variables (RVs) are written as uppercase italic letters X and a vector of RVs as $\boldsymbol{X}$. The realization of a RV X or $\boldsymbol{X}$ is written as the lowercase letters x and $\boldsymbol{x}$, respectively. The PDF of a continuous RV is denoted $p_X(x)$ and the probability mass function (PMF) of a discrete RV is denoted $P_X(x)$. For convenience, we often discard the indices of the PDFs and PMFs and write $p(x)$ and $P(x)$, respectively. The joint distribution of a continuous RV Y and a discrete RV X is written as $P_X(x)p_{Y|X}(y|x)$, but for simplicity we sometimes abbreviate as $p(x, y)$.

We write entropy and mutual information (MI) as $\mathbb{H}\left(.\right)$ and $\mathbb{I}\left(.\right)$, respectively. The binary entropy function is denoted $\mathbb{H}_2\left(.\right)$. The differential entropy of a continuous RV is denoted $h(.)$. We define the entropy rate, conditional entropy rate, and mutual information



(a) 8-ASK (b) 8-PAM
(c) 64-QAM (d) 16-SQAM

Figure 2.1.: Signal constellations used in this thesis.

rate in bits per transmitted symbol as the respective

$$\mathbb{H}_n(\boldsymbol{X}) = \frac{1}{n}\mathbb{H}(\boldsymbol{X}) \tag{2.1}$$

$$\mathbb{H}_n(\boldsymbol{X}|\boldsymbol{Y}) = \frac{1}{n}\mathbb{H}(\boldsymbol{X}|\boldsymbol{Y}) \tag{2.2}$$

$$\mathbb{I}_n(\boldsymbol{X};\boldsymbol{Y}) = \mathbb{H}_n(\boldsymbol{X}) - \mathbb{H}_n(\boldsymbol{X}|\boldsymbol{Y}) \tag{2.3}$$

2.2. Signal Constellations

We transmit symbols X from a finite alphabet $\mathcal{X}$ over a channel. The distribution of the symbols of the constellation $\mathcal{X}$ is $P_X(.)$. Fig. 2.1 shows the four types of constellations used in this thesis:

▷ Bipolar M-ary ASK in Fig. 2.1a with

$$\mathcal{X}_{\text{ASK}} = \{\pm\Delta, \pm 3\Delta, \ldots, \pm(M-1)\Delta\}. \tag{2.4}$$

▷ Unipolar M-ary PAM in Fig. 2.1b with

$$\mathcal{X}_{\text{PAM}} = \{0, \Delta, 2\Delta, \ldots, (M-1)\Delta\}\,. \tag{2.5}$$

▷ Complex M-ary QAM in Fig. 2.1c with

$$\mathcal{X}_{\text{QAM}} = \left\{\pm a\Delta \pm \mathrm{j}b\Delta \;\middle|\; a = 1, 3 \ldots, \sqrt{M}-1,\; b = 1, 3, \ldots, \sqrt{M}-1\right\}. \tag{2.6}$$

▷ Complex M-ary star-QAM (SQAM) in Fig. 2.1d with

$$\mathcal{X}_{\text{SQAM}} = \{\pm a\Delta \pm \mathrm{j}a\Delta \mid a = 1, \ldots, M/4\}\,. \tag{2.7}$$

The factor Δ modifies the average power. Note that binary phase shift keying (BPSK) is the same as 2-ASK and OOK is the same as 2-PAM.

2.3. Channel Models

This section defines channel models and reviews their capacities. The capacity of a memoryless channel with input X and output Y described by $P_{Y|X}$ is [83]

$$C = \max_{P_X} \mathbb{I}\left(X; Y\right). \tag{2.8}$$

For many channels it makes sense to define a constrained capacity:

$$C = \max_{P_X} \mathbb{I}\left(X; Y\right) \text{ s.t. } \mathbb{E}\left[|X|^2\right] \le P_{\text{tx}} \tag{2.9}$$

where we have an average power constraint. For block length $n \to \infty$, there exist codes with rate $R < C$ such that the block error probability vanishes.

2.3.1. Binary Erasure Channel

Consider a binary input alphabet $\mathcal{X} = \{0, 1\}$ and output alphabet $\mathcal{Y} = \{0, 1, \mathcal{E}\}$. The binary erasure channel (BEC) has PMF

$$P_{Y|X}(0|0) = P_{Y|X}(1|1) = 1 - \varepsilon \tag{2.10}$$

$$P_{Y|X}(\mathcal{E}|0) = P_{Y|X}(\mathcal{E}|1) = \varepsilon \tag{2.11}$$

$$P_{Y|X}(1|0) = P_{Y|X}(0|1) = 0 \tag{2.12}$$

where ε is the erasure probability. The *a posteriori* probabilities (APP) of uniform, independent, and identically distributed (u.i.i.d.) channel inputs X is

$$P_{X|Y}(0|0) = P_{X|Y}(1|1) = 1 \tag{2.13}$$

$$P_{X|Y}(0|\mathcal{E}) = P_{X|Y}(1|\mathcal{E}) = 0.5, \tag{2.14}$$

i.e., we either know the channel input or we have to guess it. The capacity of the BEC [84, Sec. 8.5.1] is

$$C_{\text{BEC}} = 1 - \varepsilon. \tag{2.15}$$

We will use BECs to construct polar and BMERA codes in section 3.6.

2.3.2. AWGN Channel

The output of an AWGN channel is

$$Y = X + \sigma N \tag{2.16}$$

where X is the channel input with distribution P_X and N is zero mean Gaussian noise with unit variance. We define the SNR as

$$\text{SNR} = \frac{\mathrm{E}\left[|X|^2\right]}{\sigma^2} \tag{2.17}$$

and the peak signal-to-noise ratio (PSNR) as

$$\text{PSNR} = \frac{\max_{x\in\mathcal{X}} |X|^2}{\sigma^2}. \tag{2.18}$$

The SNR refers to an average power constraint and the PSNR to a maximum or peak power constraint.

The channel capacity under an average power constraint for real-valued inputs is [83]

$$C(\text{SNR}) = \frac{1}{2}\log_2\left(1 + \text{SNR}\right). \tag{2.19}$$

The maximizing distribution P_X is Gaussian. The capacity of the AWGN channel under a peak power constraint or amplitude constraint is derived in [54]. The maximizing distribution is a discrete distribution where the number of signal points grows with the PSNR. At low PSNR values, the maximizing distribution has only two signal points.

Optimal ASK Signaling Under an Average Power Constraint

The optimal P_X for ASK and PAM constellations can be obtained by the Blahut-Arimoto algorithm [85,86]. As described in [40], we use sub-optimal sampled Gaussian distributions

$$P_X = Ae^{-\nu x^2} \tag{2.20}$$

where A is a normalization constant. Note that the distribution is one-sided in case of PAM. The distribution (2.20) is sometimes referred to as a Maxwell-Boltzmann (MB) distribution [87].

2.4. Coded Modulation

In most systems, the cardinality $M = |\mathcal{X}|$ is a power of two to interface with binary FEC codes. One exception where M is not a power of two is discussed in chapter 7.

2.4.1. Labeling

If M is a power of two, we can label each symbol $x \in \mathcal{X}$ by a sequence of $m = \log_2 M$ bits. We denote the labeling function for symbol x as

$$b(x) = (b_1(x), \ldots, b_m(x)), \quad \forall x \in \mathcal{X} \tag{2.21}$$

and refer to the bits $l, \ldots, k$ via

$$b_l^k(x) = (b_l(x), \ldots, b_k(x)), \quad \forall l, k = 1, \ldots, m \ \wedge \ k > l. \tag{2.22}$$

We define the labeling functions for a sequence $\boldsymbol{x} = (x_1, \ldots, x_n)$ of symbols as

$$b(\boldsymbol{x}) = (b(x_1), \ldots, b(x_n)) \tag{2.23}$$

$$b_l(\boldsymbol{x}) = (b_l(x_1), \ldots, b_l(x_n)) \tag{2.24}$$

$$b_l^k(\boldsymbol{x}) = \left(b_l^k(x_1), \ldots, b_l^k(x_n)\right) \tag{2.25}$$

Fig. 2.3 shows a basic coded modulation scheme. A source outputs k uniformly distributed data bits $\boldsymbol{D}$. The k data bits are encoded to a length nm codeword $\boldsymbol{C}$. The bits of the codeword are interpreted as the bit labels of a sequence $\boldsymbol{X}$. The mapper maps the sequence $\boldsymbol{C}$ to a sequence of n symbols $\boldsymbol{X}$. The sequence $\boldsymbol{X}$ is transmitted over a DMC described by $P_{Y|X}$. The transmission rate in bits per channel use (bpcu) of the scheme is

$$R = \frac{k}{n} \text{ [bpcu]}. \tag{2.26}$$

The supremum of rates at which data can be reliably transmitted over the channel is

$$R_{\text{CM}} = \mathbb{I}\,(X; Y)\,. \tag{2.27}$$

2.4.2. Bit-metric and Multistage Decoding

There are two main options to detect and decode information in a coded modulation system, namely BMD [88, 89] and MSD [3, 5].

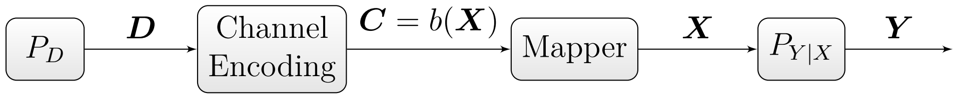


Figure 2.2.: Basic Coded Modulation Scheme.

Consider the the coded modulation rate (2.27) and the equations

$$\begin{aligned} R_{\text{CM}} &= \mathbb{I}\left(X;Y\right) && (2.28)\\ &= \mathbb{H}\left(X\right) - \mathbb{H}\left(X|Y\right) && (2.29)\\ &\overset{(a)}{=} \mathbb{H}\left(b(X)\right) - \mathbb{H}\left(b(X)|Y\right) && (2.30)\\ &\overset{(b)}{=} \mathbb{H}\left(b(X)\right) - \sum_{l=1}^{m} \mathbb{H}\left(b_l(X)|Y, b_1^{l-1}(X)\right) && (2.31)\\ &\overset{(c)}{\geq} \mathbb{H}\left(b(X)\right) - \sum_{l=1}^{m} \mathbb{H}\left(b_l(X)|Y\right) && (2.32) \end{aligned}$$

where (a) follows from the 1-to-1 labeling function, (b) from the chain rule of entropy, and (c) because conditioning cannot increase entropy.

BMD works as follows. At the receiver, a detector computes the metrics $P_{b_l(X)|Y}(b_l(x_i)|y_i)$ for $l = 1, \ldots, m$ and $i = 1, \ldots, n$. A decoder estimates the transmitted data based on these metrics. The dependency of $b_l(X)$ and $b_1^{l-1}(X)$ is neglected in the bit-metrics, so BMD achieves the rate (2.32) given by

$$R_{\text{BMD}} = \mathbb{H}\left(X\right) - \sum_{l=1}^{m} \mathbb{H}\left(b_l(X)|Y\right). \tag{2.33}$$

The rate (2.31) can be achieved by MSD. The m bit-levels are successively decoded based on the channel outputs and the previously decoded bit-levels, i.e., a detector computes

$$P_{b_l(X)|Y b_1^{l-1}(X)}(b_l(x_i)|y_i, \hat{b}_1^{l-1}(x_i)). \tag{2.34}$$

MSD requires m detection and decoding stages. MSD is usually combined with MLC, i.e., each bit-level is encoded separately at the transmitter.

2.4.3. Labeling Strategies

We discuss two labeling strategies, namely the binary reflected Gray code (BRGC) [90] and the natural binary code (NBC). The BRGC is often used in bit-interleaved coded modulation (BICM) systems with BMD as it maximizes the BMD rate (2.33) [91]. The NBC labeling follows the set partitioning properties described by [4] and is a good choice for MSD. Table 2.1 shows both labeling strategies for 8-ary ASK. BRGC and NBC are

8-ASK symbols	-7	-5	-3	-1	1	3	5	7
BRGC	000	100	110	010	011	111	101	001
NBC	000	100	010	110	001	101	011	111

Table 2.1.: BRGC and NBC labeling for 8-ASK symbols.

related [92] by the following $(m \times m)$ transformation matrix

$$\mathbf{T}_{\text{NBC}\to\text{BRGC}} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 1 \end{bmatrix} \tag{2.35}$$

that has only ones on the diagonal and sub-diagonal.

2.5. Bhattacharyya Parameter

The source Bhattacharyya parameter is (see [93])

$$Z(X|Y) = 2\sum_{y\in\mathcal{Y}} P_Y(y)\sqrt{P_{X|Y}(0|y)P_{X|Y}(1|1)} \tag{2.36}$$

and

$$Z(X|Y) = 2\int_{y\in\mathcal{Y}} P_Y(y)\sqrt{P_{X|Y}(0|y)P_{X|Y}(1|1)}\,\mathrm{d}y \tag{2.37}$$

for binary-input discrete and continuous memoryless channels, respectively. According to [93, Proposition 2] and [29, Proposition 1] we have the bounds

$$Z(X|Y)^2 \le \mathbb{H}\left(X|Y\right) \le \log_2\left(1+Z(X|Y)\right). \tag{2.38}$$

The property

$$Z(X|Y,S) \le Z(X|Y) \tag{2.39}$$

is shown in [94, Lemma 6].

2.6. Probabilistic Amplitude Shaping

PAS was introduced in [40] as a shaping scheme that combines distribution matching and FEC. The aim of PAS is to transmit a coded sequence $X_1, \dots, X_n$, where the X_i are distributed according to a specified $P_X(x)$. The symbols are taken from an alphabet $\mathcal{X}$ with $|\mathcal{X}| = M$. The alphabet size is often a power of two but we also consider $M = 6$ in chapter 7.

PAS requires dividing the constellation into two sub-constellations that have the same distribution, i.e., we divide $\mathcal{X}$ into two subsets $\mathcal{X}_1$ and $\mathcal{X}_2$ with corresponding distributions P_{X_1} and P_{X_2} such that $\mathcal{X}_1 \cup \mathcal{X}_2 = \mathcal{X}$, $\mathcal{X}_1 \cap \mathcal{X}_2 = \emptyset$, $|\mathcal{X}_1| = |\mathcal{X}_2| = M/2$, and

$$\exists x_1 \in \mathcal{X}_1 : P_{X_1}(x_1) = P_{X_2}(x_2), \quad \forall x_2 \in \mathcal{X}_2. \tag{2.40}$$

The distributions P_{X_1} and P_{X_2} thus satisfy

$$P_{X_1}(x) = \frac{1}{2} P_X(x), \quad \forall x \in \mathcal{X}_1 \tag{2.41}$$

$$P_{X_2}(x) = \frac{1}{2} P_X(x), \quad \forall x \in \mathcal{X}_2. \tag{2.42}$$

In [40], PAS is used for ASK and QAM constellations on the AWGN channel where the real- and imaginary parts of the distributions are symmetric around zero and satisfy (2.40). To build pairs of symbols from $\mathcal{X}_1$ and $\mathcal{X}_2$ with the same probability mass, we define an injective and surjective mapping

$$T : \ \mathcal{X}_1 \mapsto \mathcal{X}_2 \tag{2.43}$$

and its inverse

$$T^{-1} : \ \mathcal{X}_2 \mapsto \mathcal{X}_1 \tag{2.44}$$

such that

$$P_{X_2}(T(x)) = P_{X_1}(x) \ \forall x \in \mathcal{X}_1 \tag{2.45}$$

$$P_{X_1}(T(x)) = P_{X_2}(x) \ \forall x \in \mathcal{X}_2. \tag{2.46}$$

The PAS architecture is depicted in Fig. 2.3. A source P_D outputs $k + \gamma n$ u.i.i.d. data bits, where $0 \leq \gamma \leq 1$ is chosen such that $\gamma n \in \mathbb{N}_0$. A fixed-to-fixed length distribution matcher (DM) maps k data bits $D_1, \dots, D_k$ to a length-n sequence of symbols (amplitudes in case of ASK) $A_1, \dots, A_n$ according to the distribution P_{X_1}, where $A_i \in \mathcal{X}_1$. The symbols A_i are labeled with $m - 1$ bits by a labeling function $b_1^{m-1}(A_i)$ and we obtain the bit sequence $b_1^{m-1}(A_1^n)$ of length $(m-1)n$. This bit sequence is concatenated with additional γn data bits $D_{k+1}, \dots, D_{k+\gamma n}$ and the resulting sequence $\left(b_1^{m-1}(A_1^n), D_{k+1}, \dots, D_{k+\gamma n}\right)$ is fed to the FEC encoder. The FEC encoder is represented by the $(m - 1 + \gamma)n \times (1 - \gamma)n$ parity matrix $\mathbf{P}$ that is part of the systematic generator matrix

$$\mathbf{G} = [\mathbf{I}, \mathbf{P}] \tag{2.47}$$

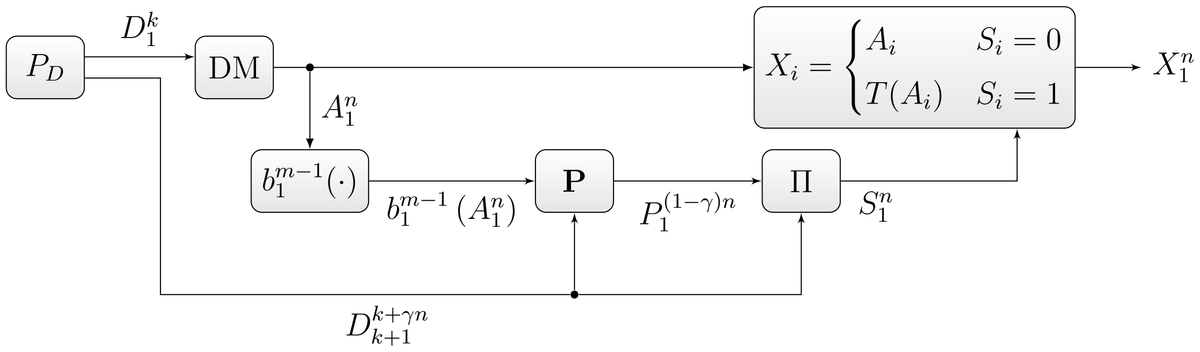


Figure 2.3.: PAS architecture.

with **I** being the identity matrix of dimension $(m+\gamma-1)n$.

The parity bits $P_1^{(1-\gamma)n}$ are assumed to be uniformly distributed [40] and we combine them with the uniformly distributed data bits $D_{k+1}^{k+\gamma n}$ to obtain a sequence of n uniformly distributed bits. We introduce an interleaver in Fig. 2.3 that assigns the parity and data bits to the bit sequence S_1^n. In [40], the bits in S_1^n are used as sign bits, i.e., the bit S_i decides whether we send $+A_i$ or $-A_i$. In our more general description, the bit S_i decides whether we transmit $A_i \in \mathcal{X}_1$ or $T(A_i) \in \mathcal{X}_2$. Usually the bits S_1^n are interpreted as the m-th bit-level in the labeling of the symbols X_i, i.e.,

$$S_i = 0 \Rightarrow b_m(X_i) = 0 \tag{2.48}$$

$$S_i = 1 \Rightarrow b_m(X_i) = 1. \tag{2.49}$$

This means the m-th bit of the label decides whether the symbol X_i is from $\mathcal{X}_1$ or $\mathcal{X}_2$ and the first $m-1$ bits of the label indicate the symbol within $\mathcal{X}_1$ or $\mathcal{X}_2$, respectively. This means we can use any label $b_1^m(X_i)$ that can be decomposed into the two described parts. An example for such a label is the BRGC label for ASK constellations.

In this thesis, we consider PAS with polar codes in chapter 5 and use PAS to transmit 6-ary constellations in chapter 7. Two commonly-used fixed-to-fixed length DMs are the CCDM [95] and the shell-mapping distribution matcher (SMDM) [96].

3

Polarizing Codes

Polar codes were described in [1, 14, 97] and shown to achieve the symmetric capacity for binary input DMCs in [1]. Polar codes are based on the polar transform and a phenomenon called channel polarization. [19] shows that polarization is a general phenomenon that is not restricted to the polar transform. For example, we will study BMERA codes [20, 21] that are based on the BMERA transform. In this chapter, we review polar codes and BMERA codes, and discuss the basics of channel polarization, decoding algorithms and code design.

3.1. Polar Transform

Consider the binary $(1 \times n)$ vectors $\boldsymbol{U}$ and $\boldsymbol{C}$. The polar transform of length n is

$$\boldsymbol{C} = \boldsymbol{U}\mathbf{G}_n \tag{3.1}$$

with the polar transformation matrix

$$\mathbf{G}_n = \mathbf{F}^{\otimes \log_2 n} \tag{3.2}$$

where $\mathbf{F}^{\otimes \log_2 n}$ is the $\log_2 n$-fold Kronecker power of the polar kernel

$$\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}. \tag{3.3}$$

The length n of the polar transform is restricted to powers of 2.

We can represent the polar transform in the recursive form

$$\mathbf{G}_n = \left(\mathbf{I}_{n/2} \otimes \mathbf{F}\right) \left(\mathbf{G}_{n/2} \otimes \mathbf{I}_2\right) \tag{3.4}$$

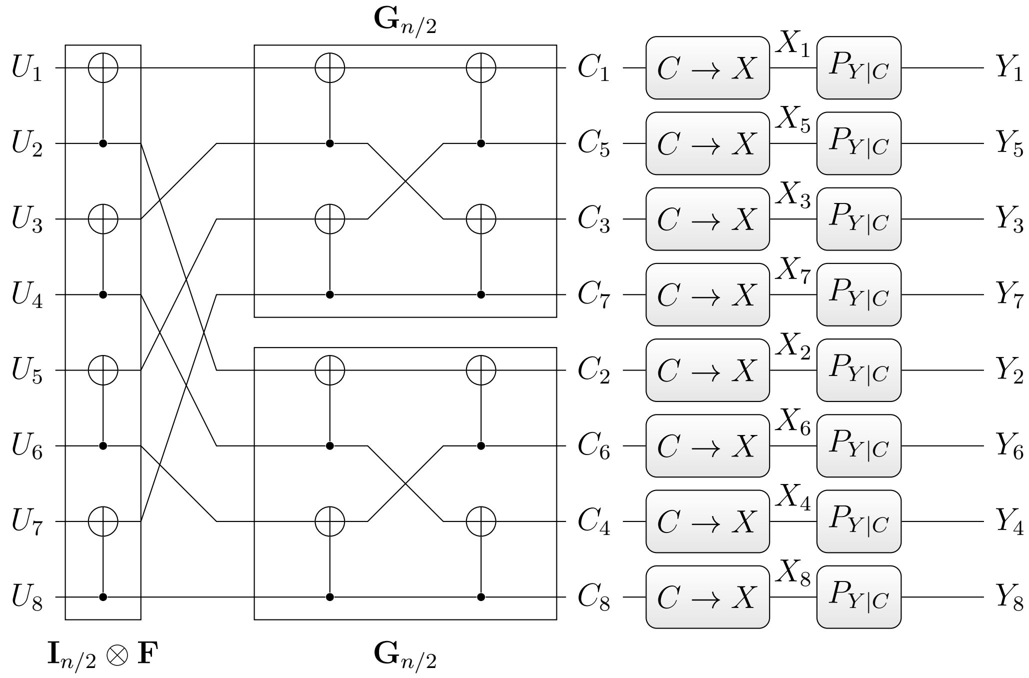


Figure 3.1.: Graphical representation of a polar transform of length $n = 8$.

with $\mathbf{G}_2 = \mathbf{F}$. We sometimes write $\mathbf{G}_n = \mathbb{G}$ and drop the index n.

Fig. 3.1 shows the polar transform for $n = 8$ in the bit-reversal permutation. This permutation is often used to highlight the recursive structure. The result $\boldsymbol{C}$ of the transform is transmitted over a binary-input DMC described by $P_{Y|C}$.

3.2. BMERA Transform

Inspired by contractible tensor networks in quantum physics [98], the authors of [20–22] came up with a new polarizing transform called the BMERA transform resulting in BMERA codes. Later in [22], the codes are renamed convolutional polar codes. We use the name BMERA since the codes are block and not convolutional codes.

The BMERA transform is based on the polar kernel from (3.3). The transform of size n, where n is a power of two, is recursively defined as

$$\mathbf{G}_n = \left[\mathbf{S}_n \left(\mathbf{I}_{n/2} \otimes \mathbf{F}\right) \mathbf{S}_n^{\mathrm{T}} \left(\mathbf{I}_{n/2} \otimes \mathbf{F}\right)\right] \cdot \left(\mathbf{G}_{n/2} \otimes \mathbf{I}_2\right) \tag{3.5}$$

where $\mathbf{G}_2 = \mathbf{S}_2 \mathbf{F} \mathbf{S}_2^{\mathrm{T}}$ and $\mathbf{S}_n$ results from an identity matrix of size n with its columns

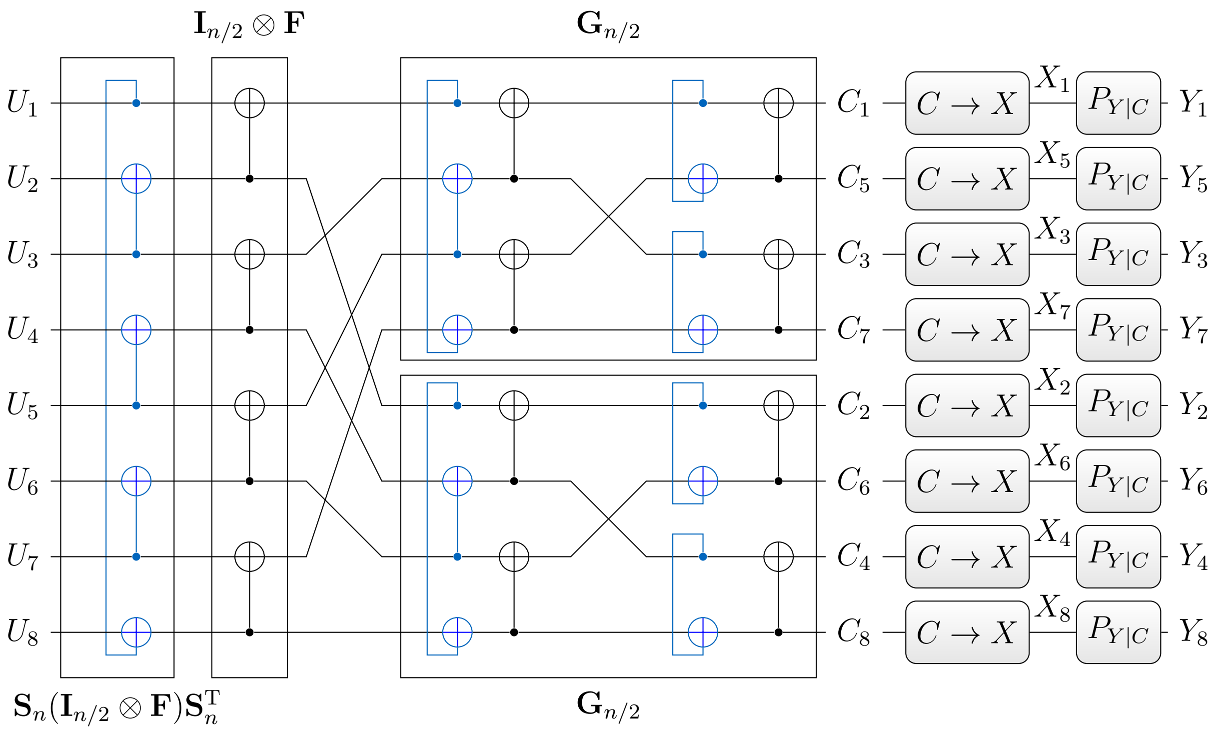


Figure 3.2.: Graphical representation of a BMERA transform of length $n = 8$.

circularly shifted to the left by one, e.g., for $n = 4$ we have

$$\mathbf{S}_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{3.6}$$

Fig. 3.2 shows the BMERA transform of length $n = 8$. Without the blue lines and sums, it represents a polar code.

3.3. Channel Polarization

We describe channel polarization for the polar transform as in section 3.1. Most of the results also hold for the BMERA transform.

The concept of polarization is presented in [1, Theorem 1] and the following Theorem is proved.

Theorem 3.1 (Theorem 1 in [1])**.** For any binary-input discrete memoryless channel (B-DMC) $P_{Y|X}$, the metrics $\mathbb{I}\left(U_i; \boldsymbol{Y}|U_1^{i-1}\right)$ with $i = 1, \ldots, n$ polarize in the sense that, for any fixed $\delta \in (0,1)$, as $n \to \infty$ through powers of two, we have

$$\lim_{n\to\infty} \frac{1}{n} \left| \left\{ i \in 1, \ldots, n : \mathbb{I}\left(U_i; \boldsymbol{Y} | U_1^{i-1}\right) \in (1-\delta, 1] \right\} \right| = 1 - \mathbb{H}\left(X|Y\right) \tag{3.7}$$

$$\lim_{n\to\infty} \frac{1}{n} \left| \left\{ i \in 1, \ldots, n : \mathbb{I}\left(U_i; \boldsymbol{Y} | U_1^{i-1}\right) \in [0, \delta) \right\} \right| = \mathbb{H}\left(X|Y\right). \tag{3.8}$$

Note that $1 - \mathbb{H}(X|Y)$ is the symmetric capacity of the channel, i.e., the capacity under the constraint that the input X is uniformly distributed. The same theorem is stated and proved for the BMERA transform in [22, Sec. IV.B-IV.C]. The proof uses the techniques from [99] that are easier to follow than the arguments in [1].

The theorem states that there are two kinds of index positions. If $\mathbb{I}\left(U_i; \boldsymbol{Y}|U_1^{i-1}\right)$ is close to one, we can reliably determine the value of U_i based on the channel observations $\boldsymbol{Y}$ and the previous symbols U_1^{i-1}. We can reliably transmit information over these positions. If $\mathbb{I}\left(U_i; \boldsymbol{Y}|U_1^{i-1}\right)$ is close to zero, then we cannot reliably estimate U_i.

3.4. Polar and BMERA Coding

Channel polarization shows we have either reliable or unreliable bit positions for large n. The paper [1] suggests to use the reliable positions for data transmission and set bits at the unreliable positions to a fixed value, usually zero. These bits are called frozen bits.

We define a polar or BMERA code by the block length n, the code rate $R = k/n$ and the set $\mathcal{F} \subseteq \{1, \ldots, n\}$ with the indices of the frozen bits, i.e., $|\mathcal{F}| = n - k$. We write the complementary set with the indices of the reliable bits as $\mathcal{U} = \mathcal{F}^{\mathsf{c}}$.

Encoding of a binary data vector $\boldsymbol{d}$ of length k works as follows. The frozen bits of $\boldsymbol{u}$ are set to zero, i.e., $\boldsymbol{u}_{\mathcal{F}} = \boldsymbol{0}$. The unfrozen bits are used to transmit data, i.e., $\boldsymbol{u}_{\mathcal{U}} = \boldsymbol{d}$. The resulting vector $\boldsymbol{u}$ is transformed by the polar or BMERA transform to obtain the codeword $\boldsymbol{c}$.

3.4.1. Systematic Encoding of Polar Codes

Polar coding as described above is non-systematic, i.e., the data bits do not appear directly in the codeword. Systematic encoding of polar codes is first described in [100]. For systematic encoding, the length k data vector $\boldsymbol{d}$ appears in the codeword, i.e., there is a subset $\mathcal{A} \subseteq \{i : i = 1, \ldots, n\}$ with $|\mathcal{A}| = k$ such that $\boldsymbol{c}_{\mathcal{A}} = \boldsymbol{d}$.

We split the transform into frozen and non-frozen parts as

$$\boldsymbol{c} = \boldsymbol{u}\mathbf{G} = \boldsymbol{u}_{\mathcal{U}}\mathbf{G}_{\mathcal{U}} + \boldsymbol{u}_{\mathcal{F}}\mathbf{G}_{\mathcal{F}}. \tag{3.9}$$

We split the codeword (3.9) into a systematic part and a part with only parity bits:

$$\boldsymbol{c}_{\mathcal{A}} = \boldsymbol{u}_{\mathcal{U}}\mathbf{G}_{\mathcal{U},\mathcal{A}} + \boldsymbol{u}_{\mathcal{F}}\mathbf{G}_{\mathcal{F},\mathcal{A}} \tag{3.10}$$

$$\boldsymbol{c}_{\mathcal{A}^{\mathsf{c}}} = \boldsymbol{u}_{\mathcal{U}}\mathbf{G}_{\mathcal{U},\mathcal{A}^{\mathsf{c}}} + \boldsymbol{u}_{\mathcal{F}}\mathbf{G}_{\mathcal{F},\mathcal{A}^{\mathsf{c}}}. \tag{3.11}$$

To have $\boldsymbol{c}_\mathcal{A} = \boldsymbol{d}$, we must set

$$\boldsymbol{u}_\mathcal{U} = (\boldsymbol{d} - \boldsymbol{u}_\mathcal{F}\mathbf{G}_{\mathcal{F},\mathcal{A}})\,\mathbf{G}_{\mathcal{U},\mathcal{A}}^{-1} \tag{3.12}$$

and can express systematic encoding by

$$\boldsymbol{c} = (\boldsymbol{d} - \boldsymbol{u}_\mathcal{F}\mathbf{G}_{\mathcal{F},\mathcal{A}})\,\mathbf{G}_{\mathcal{U},\mathcal{A}}^{-1}\mathbf{G}_\mathcal{U} + \boldsymbol{u}_\mathcal{F}\mathbf{G}\mathcal{F}. \tag{3.13}$$

In case of zero-frozen bits, (3.13) simplifies to

$$\boldsymbol{c} = \boldsymbol{d}\mathbf{G}_{\mathcal{U},\mathcal{A}}^{-1}\mathbf{G}_\mathcal{U}. \tag{3.14}$$

We see that the set $\mathcal{A}$ must be chosen so the matrix $\mathbf{G}_{\mathcal{U},\mathcal{A}}$ has full rank and is invertible. The straightforward choice of $\mathcal{A}$ for polar codes is $\mathcal{A} = \mathcal{U}$ [100]. For polar codes, $\mathbf{G}$ is a lower triangular matrix with only ones on the diagonal. With $\mathcal{A} = \mathcal{U}$, we have $\mathbf{G}_{\mathcal{U},\mathcal{A}} = \mathbf{G}_{\mathcal{U},\mathcal{U}}$ which is again lower triangular with only ones on the diagonal. Thus, $\mathbf{G}_{\mathcal{U},\mathcal{U}}$ is invertible and a good choice for polar codes. Efficient implementations of systematic encoding for polar codes with $\mathbf{G}_{\mathcal{U},\mathcal{U}}$ without performing the matrix inversion explicitly can be found in [101–103].

The transformation matrix of BMERA codes is not lower-triangular and we cannot simply choose $\mathcal{A} = \mathcal{U}$.

3.5. Polar and BMERA Decoding

3.5.1. Successive Cancellation Decoding

Theorem 3.1 suggests the decoding algorithm for polarizing codes. The bits U_i are decoded successively based on the channel outputs $\boldsymbol{y}$ and the previously decoded bits $\hat{u}_1^{i-1}$. This approach is called SC decoding [1]. If $i \in \mathcal{F}$ is a frozen position, we set the value of $\hat{u}_i$ to the frozen value, usually zero. If $i \in \mathcal{U}$ is a non-frozen position, we evaluate the decoding metrics $P_{U_i|\boldsymbol{Y}U_1^{i-1}}(u_i|\boldsymbol{y}, \hat{u}_1^{i-1})$, i.e.,

$$\hat{u}_i = \begin{cases} \text{frozen value} & i \in \mathcal{F} \\ 0 & i \in \mathcal{U} \quad \wedge \quad P_{U_i|\boldsymbol{Y}U_1^{i-1}}(0|\boldsymbol{y}, \hat{u}_1^{i-1}) \geq P_{U_i|\boldsymbol{Y}U_1^{i-1}}(1|\boldsymbol{y}, \hat{u}_1^{i-1}) \\ 1 & \text{otherwise.} \end{cases} \tag{3.15}$$

The metrics $P_{U_i|\boldsymbol{Y}U_1^{i-1}}(u_i|\boldsymbol{y}, \hat{u}_1^{i-1})$ can be efficiently computed via the structure of the polar transform. Fig. 3.3 shows a cutout from the polar transform. The decoding metric for the bits with odd indices can be computed as

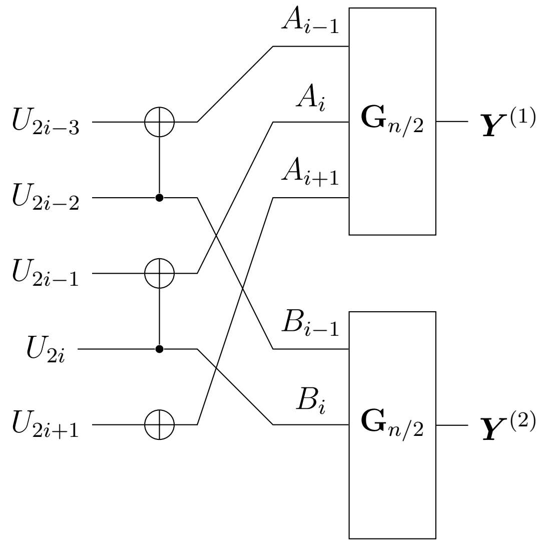


Figure 3.3.: Computation of the decoding metrics.

$$P_{U_{2i-1}|\boldsymbol{Y}U_1^{2i-2}}\left(u_{2i-1}|\boldsymbol{y},u_1^{2i-2}\right) \tag{3.16}$$
$$\stackrel{(a)}{=}\sum_{u_{2i}} P_{U_{2i-1}U_{2i}|\boldsymbol{Y}U_1^{2i-2}}\left(u_{2i-1},u_{2i}|\boldsymbol{y},u_1^{2i-2}\right) \tag{3.17}$$
$$\stackrel{(b)}{=}\sum_{u_{2i}} P_{A_iB_i|\boldsymbol{Y}A_1^{i-1}B_1^{i-1}}\left(u_{2i-1}\oplus u_{2i},u_{2i}|\boldsymbol{y},a_1^{i-1}(u_1^{2i-2}),b_1^{i-1}(u_1^{2i-2})\right) \tag{3.18}$$
$$\stackrel{(c)}{=}\sum_{u_{2i}} P_{A_i|\boldsymbol{Y}^{(1)}A_1^{i-1}}\left(u_{2i-1}\oplus u_{2i}|\boldsymbol{y}^{(1)},a_1^{i-1}(u_1^{2i-2})\right) P_{B_i|\boldsymbol{Y}^{(2)}B_1^{i-1}}\left(u_{2i}|\boldsymbol{y}^{(2)},b_1^{i-1}(u_1^{2i-2})\right) \tag{3.19}$$

where (a) is a marginalization and (b) is a transform of variables where $a_1^{i-1}(u_1^{2i-2})$ and $b_1^{i-1}(u_1^{2i-2})$ specify that a_1^{i-1} and b_1^{i-1} are functions of the previous bits u_1^{2i-2}. Step (c) follows because the two parts are independent, as can be seen in Fig. 3.3.

For bits with even indices, the decoding metric can be computed as

$$P_{U_{2i}|\boldsymbol{Y}U_1^{2i-1}}\left(u_{2i}|\boldsymbol{y},u_1^{2i-1}\right) \tag{3.20}$$
$$\stackrel{(a)}{=}\frac{1}{c}P_{U_{2i-1}U_{2i}|\boldsymbol{Y}U_1^{2i-2}}\left(u_{2i-1},u_{2i}|\boldsymbol{y},u_1^{2i-2}\right) \tag{3.21}$$
$$\stackrel{(b)}{=}P_{A_i|\boldsymbol{Y}^{(1)}A_1^{i-1}}\left(u_{2i-1}\oplus u_{2i}|\boldsymbol{y}^{(1)},a_1^{i-1}(u_1^{2i-2})\right) P_{B_i|\boldsymbol{Y}^{(2)}B_1^{i-1}}\left(u_{2i}|\boldsymbol{y}^{(2)},b_1^{i-1}(u_1^{2i-2})\right) \tag{3.22}$$

where (a) follows from Bayes' rule with a constant normalization term c and (b) follows from independence. The metrics $P_{A_i|\boldsymbol{Y}^{(1)}A_1^{i-1}}$ and $P_{B_i|\boldsymbol{Y}^{(1)}B_1^{i-1}}$ needed in (3.19) and (3.22) can be computed in the same way. The recursion stops when $\boldsymbol{y}^{(1)}$ and $\boldsymbol{y}^{(2)}$ are scalars and we can directly use the channel APPs $P_{X|Y}$.

An efficient implementation of the recursive calculations is given in [15]. The paper [104] provides an implementation based on log-likelihood ratios (LLRs) and [105] suggests using the min-sum approximation for LLR based decoders. Efficient calculations of these metrics for BMERA codes are presented in [26–28] and we have added the same derivations as above for polar codes in Appendix A for BMERA codes. Polar and BMERA codes asymptotically achieve capacity under SC decoding [1, 22]. The frame error probability for polar and BMERA codes of length n and with coderate R under SC decoding satisfies

$$P_e(n, R) = \mathcal{O}\left(2^{-n^\beta}\right) \tag{3.23}$$

with $\beta < \frac{1}{2}$ for polar codes [106, Theorem 2] and $\beta < \frac{1}{2}\log_2 3$ for BMERA codes [22, Theorem 2]. The higher error exponent β of BMERA codes results in a better SC performance of BMERA codes as compared to polar codes. The performance for short block lengths n suffers under SC decoding due to error propagation in the successive decoding approach.

3.5.2. Successive Cancellation List Decoding

SCL decoding [15] improves the performance of polar codes. Instead of hard-decisions for every bit U_i for $i \in \mathcal{U}$, a list of possible estimates $\hat{\boldsymbol{u}}$ is created. For every $i \in \mathcal{U}$, we split decoding into two paths for $\hat{u}_i = 0$ and $\hat{u}_i = 1$ and assign a path metric to each path. The path metric is proportional to the metric $P_{U_1^i|\boldsymbol{Y}}(\hat{u}_1^i|\boldsymbol{y})$. When the number of paths exceeds a predefined list size L, the paths with the lowest path metrics are discarded and decoding is continued with the remaining L paths. SCL decoding results in a list of L possible estimates $\hat{\boldsymbol{u}}(1), \ldots, \hat{\boldsymbol{u}}(L)$. Finally, the estimate with the highest path metric is chosen.

The performance of SCL decoding is close to the performance of the maximum-likelihood decoder [15] at high SNR. An implementation of SCL decoding for polar codes based on LLRs is presented in [104]. An efficient implementation of SCL decoding for BMERA codes is presented in [107].

3.5.3. CRC-aided Successive Cancellation Decoding

Polar and BMERA codes with SCL decoding cannot compete with state-of-the art codes such as LDPC codes in terms of frame error rate (FER). Since SCL decoding achieves maximum-likelihood decoding performance, this must be due to poor code properties such as the minimum distance or distance profile. Outer CRC codes are suggested in [15] to improve the code properties. The CRC code can be integrated in the SCL decoding process. Every codeword from the final list that does not pass the cyclic redundancy check is removed from the list. The most likely codeword from the remaining list is then chosen as the estimate. Polar codes and BMERA codes with an outer CRC code can compete with state-of-the art codes, e.g., polar codes with an outer CRC code were chosen for the 5G new radio standard [2].

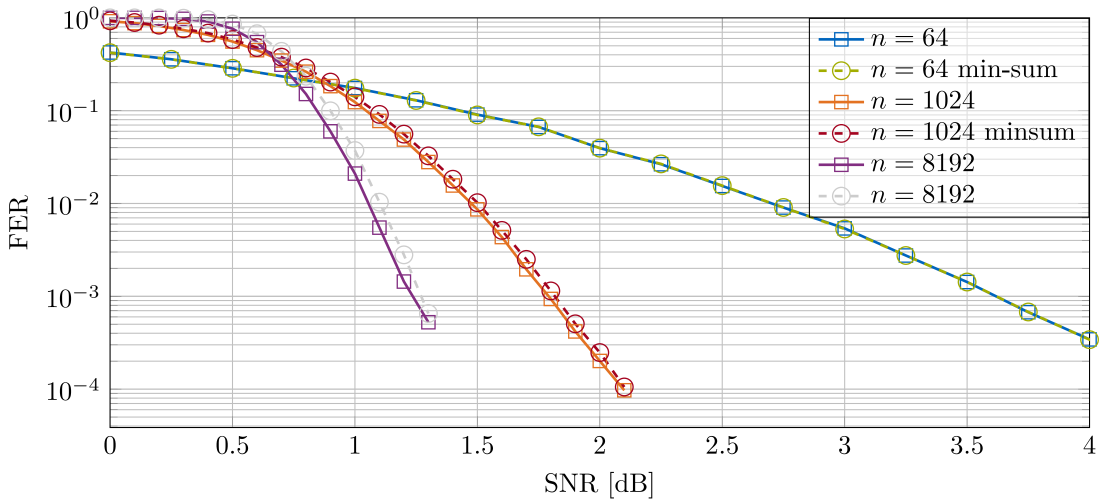


Figure 3.4.: Loss due to min-sum approximations in CRC-aided list decoding of polar codes with list size $L = 32$ for $n = 64, 1024, 8192$ at a rate of 0.5 bpcu over an AWGN channel.

3.5.4. Impact of the Min-sum Approximation on Frame Error Rate

This section presents numerical results showing the impact of the min-sum approximation on the FER for CRC-aided list decoding of polar codes. During the simulation, each channel output is decoded twice. First, the SCL algorithm is used with exact computation of all metrics. Second, the min-sum approximation is used within the SCL algorithm. Fig. 3.4 shows the FERs for both methods for a rate of 0.5 bpcu. A 5-bit CRC was used for $n = 64$ and a 8-bit CRC for $n = 1024$ and $n = 8192$. For $n = 64$, no FER degradation can be observed when using the min-sum approximation. As n grows, the loss in terms of FER grows when using the min-sum approximation. Nevertheless, the losses are negligible for most practical applications. In this thesis, all the results with polar and BMERA decoders are performed with the min-sum approximation.

3.6. Code Design

By code design, we mean the choice of the sets $\mathcal{F}$ and $\mathcal{U}$ with the frozen and information bit indexes, respectively. Consider a (n, k) code, i.e., we need to find a set $\mathcal{F}$ with $|\mathcal{F}| = n - k$ and the complementary set $\mathcal{U}$ such that

$$\mathbb{I}\left(U_i; \boldsymbol{Y}|U_1^{i-1}\right) \geq \mathbb{I}\left(U_j; \boldsymbol{Y}|U_1^{j-1}\right) \quad \forall i, j \ : \ i \in \mathcal{U}, j \in \mathcal{F}. \tag{3.24}$$

We sort the k indices associated with the largest MI terms into the set $\mathcal{U}$. The challenge of code design is computing or approximating the MI metrics $\mathbb{I}\left(U_i; \boldsymbol{Y}|U_1^{i-1}\right)$. We discuss

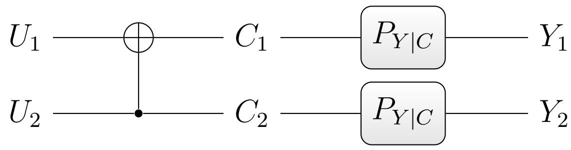


Figure 3.5.: Density evolution with erasure channels for a polar code of size $n = 2$.

some common methods in the following subsections. Note that the resulting sets $\mathcal{F}$ and $\mathcal{U}$ are optimal under SC decoding but not necessarily under SCL decoding or CRC-aided SCL decoding. Methods for designing polar codes for SCL decoding are discussed in [108].

3.6.1. Monte Carlo Design

The design of polarizing codes with Monte Carlo methods works for all channels where we can provide metrics of the form $P_{C|Y}(c_i|y_i)$ for $i = 1, \ldots, n$. Consider the mutual information

$$\mathbb{I}\left(U_i; \boldsymbol{Y}|U_1^{i-1}\right) = \mathbb{H}\left(U_i|U_1^{i-1}\right) - \mathbb{H}\left(U_i|\boldsymbol{Y}, U_1^{i-1}\right) \tag{3.25}$$

$$\stackrel{(a)}{=} 1 - \mathbb{H}\left(U_i|\boldsymbol{Y}, U_1^{i-1}\right) \tag{3.26}$$

where (a) follows because the U_i are u.i.i.d.. The term $\mathbb{H}\left(U_i|\boldsymbol{Y}, U_1^{i-1}\right)$ can be approximated as follows. We randomly generate N_{MC} realizations of the vector $\boldsymbol{U}$, i.e., $\boldsymbol{u}(1), \ldots, \boldsymbol{u}(N_{\mathrm{MC}})$. We transform each of these vectors using the polar or BMERA transform and transmit the resulting codewords $\boldsymbol{c}(1), \ldots, \boldsymbol{c}(N_{\mathrm{MC}})$ over the channel to obtain N_{MC} observation vectors $\boldsymbol{y}_1, \ldots, \boldsymbol{y}(N_{\mathrm{MC}})$. From sampling theory (c.f. [109]), we can perform the approximation

$$\mathbb{H}\left(U_i|\boldsymbol{Y}, U_1^{i-1}\right) = \sum_{u_1^{i-1}} \int_{\boldsymbol{y}} P_{U_1^{i-1}}(u_1^{i-1}) p_{\boldsymbol{Y}|U_1^{i-1}}(\boldsymbol{y}|u_1^{i-1}) \mathbb{H}\left(U_i|\boldsymbol{Y} = \boldsymbol{y}, U_1^{i-1} = u_1^{i-1}\right) \tag{3.27}$$

$$\approx \frac{1}{N_{\mathrm{MC}}} \sum_{k=1}^{N_{\mathrm{MC}}} \mathbb{H}_2\left(P_{U_i|\boldsymbol{Y}U_1^{i-1}}\left(0|\boldsymbol{y}(k), u_1^{i-1}(k)\right)\right). \tag{3.28}$$

The metrics $P_{U_i|\boldsymbol{Y}U_1^{i-1}}$ needed for the approximation can be obtained by running a SC decoder.

3.6.2. Density Evolution with Erasure Channels

Polar Codes

The probability of an erasure is $P_{Y|C}(\mathcal{E}|c_i) = \epsilon$ for $i = 1, 2$. The symbol U_1 can be reconstructed only if C_1 and C_2 are both known, i.e., the probability that U_1 cannot be reconstructed is

$$\varepsilon^- = 1 - (1 - \varepsilon)^2 = 2\epsilon - \varepsilon^2. \tag{3.29}$$

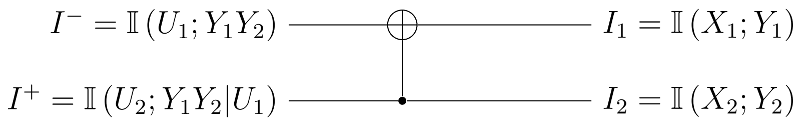


Figure 3.6.: MIs of the basic polar transform.

When decoding symbol U_2, we know $\mathcal{U}_1$. We cannot reconstruct U_2 if C_1 and C_2 are both erased, i.e., the probability that U_2 can not be reconstructed when knowing U_1 is

$$\varepsilon^+ = \varepsilon^2. \tag{3.30}$$

For $n > 2$, we recursively apply (3.30) and (3.29) to get the erasure probabilities for all symbols $U_1, \ldots, U_n$.

BMERA Codes

Density evolution for BMERA codes needs more than two update equations. The complete derivation of the BMERA density evolution for erasure channels is outside the scope of this thesis. A derivation and all 32 update equations can be found in the appendix of [22]

3.6.3. Approximate Density Evolution Based on Gaussian Approximations

Polar code design for binary-input additive white Gaussian noise (biAWGN) channels based on density evolution is presented in [110]. We study an approximate density evolution suggested in [111–113]. Fig. 3.6 shows the basic polar transform and displays the MI expressions. According to [114], the MIs I^- and I^+ are calculated from I_1 and I_2 as

$$I^- = 1 - J\left(\sqrt{\left[J^{-1}(1-I_1)\right]^2 + \left[J^{-1}(1-I_2)\right]^2}\right) \tag{3.31}$$

$$I^+ = J\left(\sqrt{\left[J^{-1}(I_1)\right]^2 + \left[J^{-1}(I_2)\right]^2}\right) \tag{3.32}$$

where the J-function is

$$J(\sigma) = 1 - \int_{-\infty}^{+\infty} \frac{e^{-\left(\left(\xi-\sigma^2/2\right)^2/2\sigma^2\right)}}{\sqrt{2\pi}\sigma} \cdot \log_2\left(1 + e^{-\xi}\right) \mathrm{d}\xi. \tag{3.33}$$

The authors of [111, 112] approximate the J-function as in [115]. In [113], it is shown that the approximation from [116, Chapter 4.2] is a better choice for polar codes. We use the approximations

$$J(\sigma) \approx \left(1 - 2^{-H_1\sigma^{2H_2}}\right)^{H_3} \tag{3.34}$$

$$J^{-1}(I) \approx \left(-\frac{1}{H_1}\log_2\left(1 - I^{\frac{1}{H_3}}\right)\right)^{\frac{1}{2H_2}} \tag{3.35}$$

from [116, Eq. (9)-(10)] with $H_1 = 0.3073$, $H_2 = 0.8935$ and $H_3 = 1.1064$.

4

Polar and BMERA Coded Modulation

The previous chapter discussed polar and BMERA codes for B-DMCs. Polar codes for higher-order modulation schemes, i.e., ASK and QAM, were presented in [7]. In principle, there are two possible choices for coded modulation, as mentioned in Sec. 2.4.2:

1. BMD;
2. MSD with MLC;

MSD requires successive detection and decoding of the bit-levels. This fits well with polarizing codes due to the successive decoding methods for polar and BMERA codes. We mostly use polar codes or PCM for the description. All of the methods are similar for BMERA codes.

4.1. Polar Mapper and Demapper

Fig. 4.1 shows the multilevel structure of PCM. Consider m input vectors $\boldsymbol{u}_l$ of length n for $l = 1, \ldots, m$. Each of the m vectors is transformed by a polar transform to obtain m vectors $\boldsymbol{c}_l$, where $\boldsymbol{c}_l = b_l(\boldsymbol{x})$ has the l-th bit-level of the symbols $x_1, \ldots, x_n$. A mapper maps the bits to the corresponding symbols $x_1, \ldots, x_n$ that are transmitted over the channel, and the receiver sees a vector $\boldsymbol{y}$ of dimension n.

Fig. 4.2 shows the principle of multistage demapping and decoding. The bit-levels are successively demapped and decoded based on the channel outputs y_i and previously decoded bit-levels. The demapper of the l-th stage computes the APPs

$$P_{b_l(X)|Y b_1^{l-1}(X)}\left(b_l(x_i)|y_i, \hat{b}_1^{l-1}(x_i)\right) \quad \forall i = 1, \ldots, n \ \wedge \ b_l(x_i) = 0, 1. \tag{4.1}$$

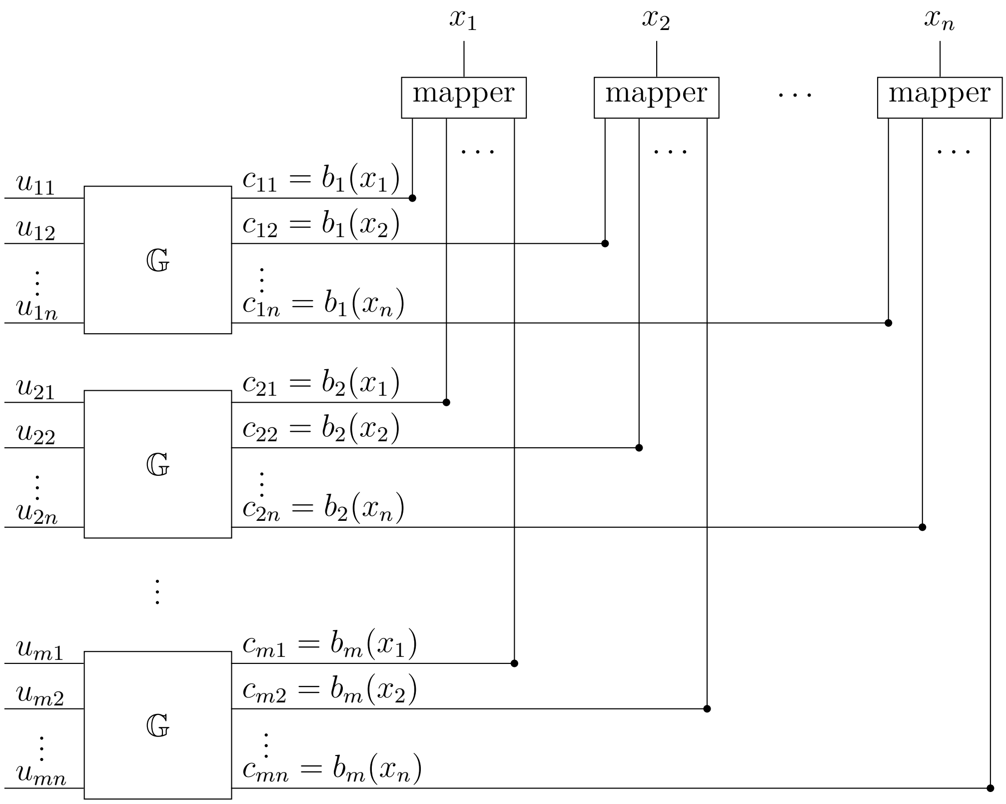


Figure 4.1.: Multilevel structure of PCM [7].

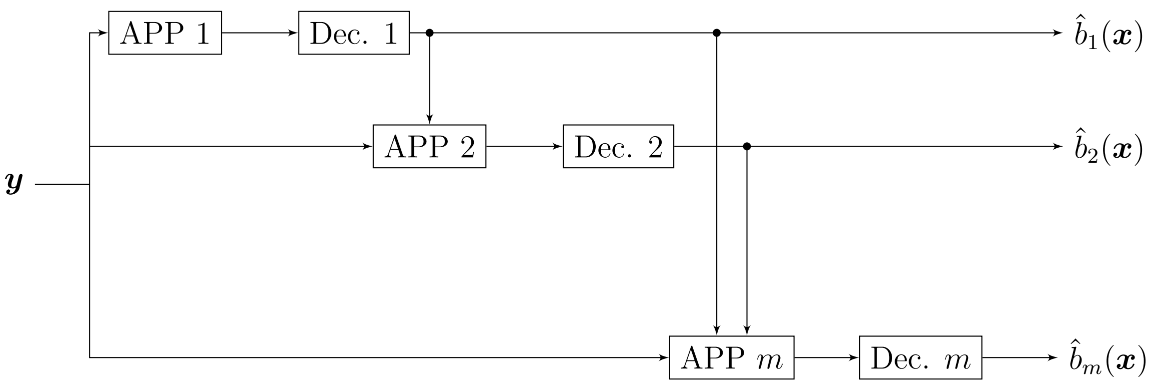


Figure 4.2.: Generic multistage demapping and decoding.

The term $\hat{b}_1^{l-1}(x_i)$ specifies that the bit-levels $1, \ldots, l-1$ are known from previous stages. The metrics or LLR values obtained from the metrics are passed to the decoder of the l-th stage that computes an estimate $\hat{b}_l(\boldsymbol{x}) = \hat{\boldsymbol{c}}_l$.

4.2. Design of Codes for Polar and BMERA Coded Modulation

We can design the codes for polar and BMERA coded modulation using the Monte Carlo methods described in section 3.6.1. If the number of Monte Carlo iterations is large enough, the results are nearly optimal for SC decoding. This comes at the cost of high computational complexity. In the following, we present a more efficient method based on surrogate channels [77].

4.2.1. Design Based on Surrogate Channels

Designs based on surrogate channels are presented in [77, 117]. We define biAWGN and BEC surrogate channels per bit-level $l = 1, \ldots, m$ with input $\tilde{X}_l$ and output $\tilde{Y}_l$. The biAWGN surrogate channel is characterized by its noise variance $\tilde{\sigma}_l^2$ and the BEC surrogate channel by its erasure probability $\tilde{\varepsilon}_l$. The rates of the surrogate channels are

$$R_{\text{biAWGN}} = \mathbb{I}\left(\tilde{X}; \tilde{X} + \sigma_l N\right) \tag{4.2}$$

$$R_{\text{BEC}} = 1 - \tilde{\varepsilon}_l. \tag{4.3}$$

N denotes unit variance AWGN. The MI of the real channel can be split into bit-wise components as

$$\mathbb{I}(X;Y) = \sum_{l=1}^{m} \mathbb{I}\left(b_l(X); Y \middle| b_1^{l-1}(X)\right) \tag{4.4}$$

$$= \sum_{l=1}^{m} \mathbb{H}\left(b_l(X) \middle| b_1^{l-1}(X)\right) - \mathbb{H}\left(b_l(X) \middle| Y, b_1^{l-1}(X)\right) \tag{4.5}$$

We consider two methods to find the variance or erasure probability of the surrogate channel.

1. The surrogate channel is chosen so its rate is the MI term in (4.4); see [77]. The resulting variance and erasure probability are

$$\tilde{\sigma}_l^2 : \quad R_{\text{biAWGN}} = \mathbb{I}\left(b_l(X); Y \middle| b_1^{l-1}(X)\right) \tag{4.6}$$

$$\tilde{\varepsilon}_l : \quad R_{\text{BEC}} = \mathbb{I}\left(b_l(X); Y \middle| b_1^{l-1}(X)\right). \tag{4.7}$$

For the BEC surrogate channel, we can write

$$\tilde{\varepsilon}_l = 1 - \mathbb{I}\left(b_l(X); Y \middle| b_1^{l-1}(X)\right). \tag{4.8}$$

2. The surrogate channel is chosen so its entropy is $\mathbb{H}\left(b_l(X)\middle|Y, b_1^{l-1}(X)\right)$ in (4.5). The resulting variance of the biAWGN surrogate channel and the erasure probability of the BEC surrogate channel are

$$\tilde{\sigma}_l^2: \quad R_{\text{biAWGN}} = 1 - \mathbb{H}\left(\tilde{X}|\tilde{Y}\right) = 1 - \mathbb{H}\left(b_l(X)\middle|Y, b_1^{l-1}(X)\right) \tag{4.9}$$

$$\tilde{\varepsilon}_l: \quad R_{\text{BEC}} = 1 - \mathbb{H}\left(\tilde{X}|\tilde{Y}\right) = 1 - \mathbb{H}\left(b_l(X)\middle|Y, b_1^{l-1}(X)\right). \tag{4.10}$$

For the BEC surrogate channel, we can write

$$\tilde{\varepsilon}_l = \mathbb{H}\left(b_l(X)\middle|Y, b_1^{l-1}(X)\right). \tag{4.11}$$

Both methods are identical for uniformly distributed input symbols X, since we have $\mathbb{H}\left(b_l(X)\middle|b_1^{l-1}(X)\right) = 1$ in that case.

Once we have defined the surrogate channels for each bit-level, we perform density evolution using the BEC surrogate channel as described in section 3.6.2, or approximate density evolution for the biAWGN surrogate channel as in 3.6.3. Density evolution is executed separately for each of the m bit levels but the frozen bits are chosen jointly for all bit-levels. For each bit-level, density evolution outputs n metrics related to the reliability of decoding the corresponding bits under SC decoding. We combine the metrics of all bit-levels to obtain mn metrics in total. Of these mn bits, we choose the k most reliable indices for a code with coderate $k/(mn)$. This simultaneously optimizes the rate allocation.

Fig. 4.3 compares the surrogate methods. We simulate polar codes with non-uniform input distributions P_X using PAS as described in chapter 5. We design and simulate 8-ASK transmission with $n = 512$ and transmit at rate $R = 2$ bpcu. The rate of the underlying code is $k/(mn) = 0.7$. No outer CRC code is used and we use SC decoding and no list decoding. The codes are designed for each SNR point and not for a particular design SNR.

Observe that the design based on entropy is better for non-uniform input distributions than the design based on MI. Observe also that one can use either the BEC or biAWGN surrogate channels. The design with biAWGN seems to be slightly better but the difference is marginal. In this thesis, we use biAWGN surrogates for polar codes and BEC surrogates for BMERA codes.

4.3. Multistage List Decoding of Polar Coded Modulation

Decoding multilevel polar codes based on BMD is described in [6, 7, 16–18]. Numerical results with SCL decoding based on BMD are presented in [118, 119]. Results for multilevel

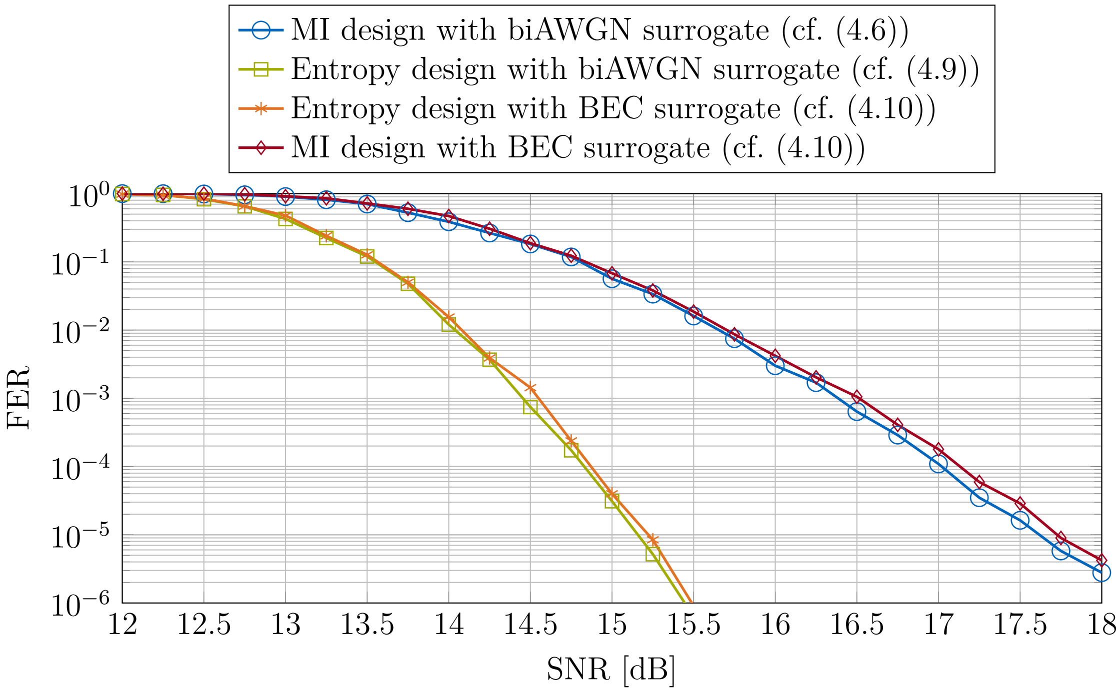


Figure 4.3.: FER comparisons for polar-coded modulation with non-uniform input distributions and with surrogate channels for $n = 512$ and $R = 2$ bpcu.

polar codes with SCL are presented on [120]. However, it is not stated if MSD or BMD was used, nor how the list is handled between the stages.

Multistage SC decoding works as follows. An SC decoder successively decodes the m stages. After decoding stage l, the estimates of the first l stages are passed to a demapper that computes the decoding metrics for the $(l + 1)$-th stage as described in section 2.4.2. Multistage SC decoding is described in [6, 7, 16, 121]. The paper [122] mentions that for the extension of list decoding, the multilevel structure should be viewed as one code rather than m short codes. An outer CRC code that comprised all m levels could be used in this case, but no results for list decoding are presented.

First results for multistage SCL decoding were presented in [76, 77], and more detail was provided in [123]. Multistage SCL decoding works as follows. At the transmitter, the data bits of all bit-levels are encoded jointly by a single CRC code. The data bits and CRC parity are encoded by the PCM scheme as described in section 4.1 and transmitted over the channel. The demapper calculates the metrics for the first bit-level. The first decoder is a SCL decoder, as described in [15], and outputs a list of codewords. For each codeword in the list, the demapper computes the metrics for the second bit-level. The second decoder is thus initialized by a list of decoding metrics and again provides a list of possible codewords. We continue until we have decoded all m bit-levels. The m-th decoder outputs a list of possible codewords of the last stage. Each list element of the last decoder is then concatenated with the corresponding outputs of the first $m - 1$ decoders. We check

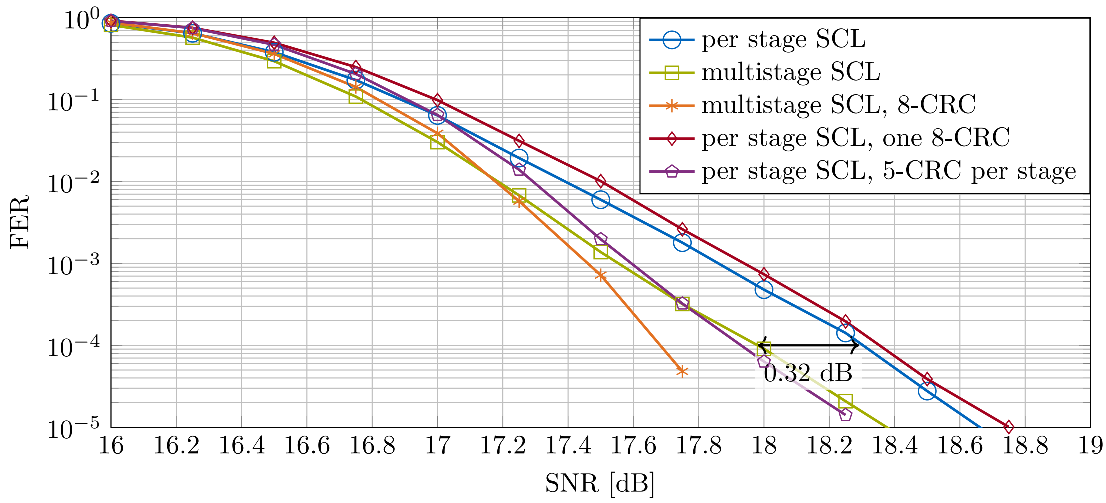


Figure 4.4.: FER of multistage list decoding with list size 32 for 8-ASK with $n = 512$ and rate $R = 2.5$ bpcu.

the CRC and delete all list elements that do not pass the check. From the reduced list, we choose the most likely codeword as our final estimate.

Fig. 4.4 shows FER curves with multistage list decoding for 8-ASK with $n = 512$ and $R = 2.5$ bpcu. We compare multistage list decoding as described above with per-stage list decoding where a hard decision of the most likely codeword from a list is made after each stage. Without outer CRC codes, multistage SCL decoding is about 0.32 dB more power efficient than per-stage SCL at an FER of 10^{-4}.

We compare two CRC strategies for per-stage SCL decoding. We use either one CRC of length 8 across all bit-levels or one CRC of length 5 per stage where the latter is based on a simple line search. We observe that one CRC across all bit-levels performs worse than pure per-stage SCL. The reason is that the CRC can be used only during decoding of the last stage and cannot correct errors in the first $m-1$ stages. Usually the last stage is the most reliable one and the CRC cannot help that much in this stage. One CRC per stage performs better since we also use CRC codes for the first $m-1$ stages.

The CRC in multistage SCL combines all bit-levels of the code to one overall code. Often the small code lengths of the component codes is listed as a disadvantage of multilevel schemes. Multistage SCL decoding with one outer CRC code partially compensates this loss. Especially at high SNR, we observe a larger slope in the FER curves with CRC-aided multistage SCL decoding compared to per stage SCL decoding.

5

Polar Coded Probabilistic Amplitude Shaping

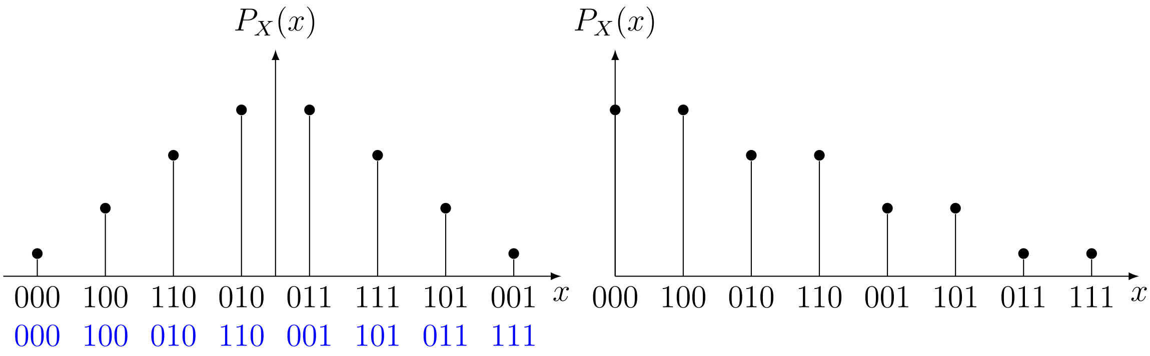
PAS was introduced in [40] in combination with LDPC codes and BMD. MSD with MLC is not considered in [40] since it requires modifying the PAS scheme. In this chapter, we show how PAS can be combined with MSD with multilevel PCM. The results of this chapter were published in [76]. The results in this chapter apply only to polar codes and not to BMERA codes.

5.1. Labeling Transform

PAS divides the constellation $\mathcal{X}$ into two subsets $\mathcal{X}_1$ and $\mathcal{X}_2$, as described in section 2.6 where the symbols in $\mathcal{X}_1$ and $\mathcal{X}_2$ have the same distribution, i.e., for every point in $\mathcal{X}_1$ there is a point in $\mathcal{X}_2$ with the same probability mass. The label of the symbols $b(X_i)$ must be decomposable into two parts such that one bit indicates whether the symbol is from $\mathcal{X}_1$ or $\mathcal{X}_2$, and the remaining $m-1$ bits indicate the symbol within $\mathcal{X}_1$ and $\mathcal{X}_2$. We call the bits that indicate whether the symbol is from $\mathcal{X}_1$ or $\mathcal{X}_2$ sign-bits as in [40] for ASK constellations. For constellations other than ASK these bits are not necessarily sign bits but we will use the name anyway. We consider ASK and PAM in this chapter.

Fig. 5.1a shows a non-uniform distribution for 8-ASK. We can define the sets $\mathcal{X}_1$ and $\mathcal{X}_2$ of positive and negative points, respectively. We show two labelings in Fig. 5.1a: a BRGC label and a NBC label. For the BRGC label, the m-th bit is the sign-bit. The NBC label, which performs well under MSD [7, 77], does not provide a bit that separates the set into $\mathcal{X}_1$ and $\mathcal{X}_2$. The BRGC label provides this bit, but the performance is worse under MSD.

The optimal distribution of PAM under an average power constraint cannot be divided into two sets $\mathcal{X}_1$ and $\mathcal{X}_2$, except for the uniform distribution. We use the distribution shown in Fig. 5.1b and assign the same probability to pairs of points. Fig. 5.2 shows rates for 4-PAM and 8-PAM, where we use MB distributions for P_X. We compare a normal



(a) 8-ASK distribution with BRGC label and NBC label in blue.

(b) 8-PAM distribution with NBC label.

Figure 5.1.: Example of two non-uniform distributions and their corresponding labels.

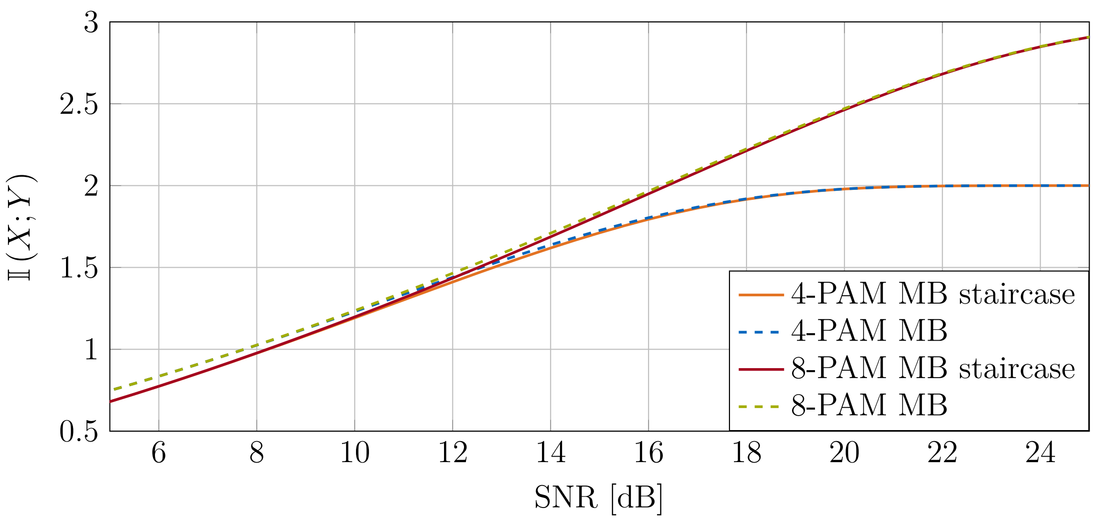


Figure 5.2.: Mutual information $\mathbb{I}(X;Y)$ for PAM constellations.

MB distribution, where we have M different probability masses, with a staircase MB distribution. For the staircase MB distribution, the probabilities of the "steps" follow a MB distribution. The parameter ν of the MB distribution (cf. (2.20)) is chosen such that $\mathbb{I}(X;Y)$ is maximized for every SNR point. The sub-optimality of the staircase distribution is seen at small SNR values. At high SNR, the staircase distribution is almost optimal. For PAM, we can directly use the NBC label. Note that the first bit-level of the NBC labeling is the "sign-bit" in case of PAM.

To use PAS with ASK as in [40], but performing MSD based on the NBC label, we

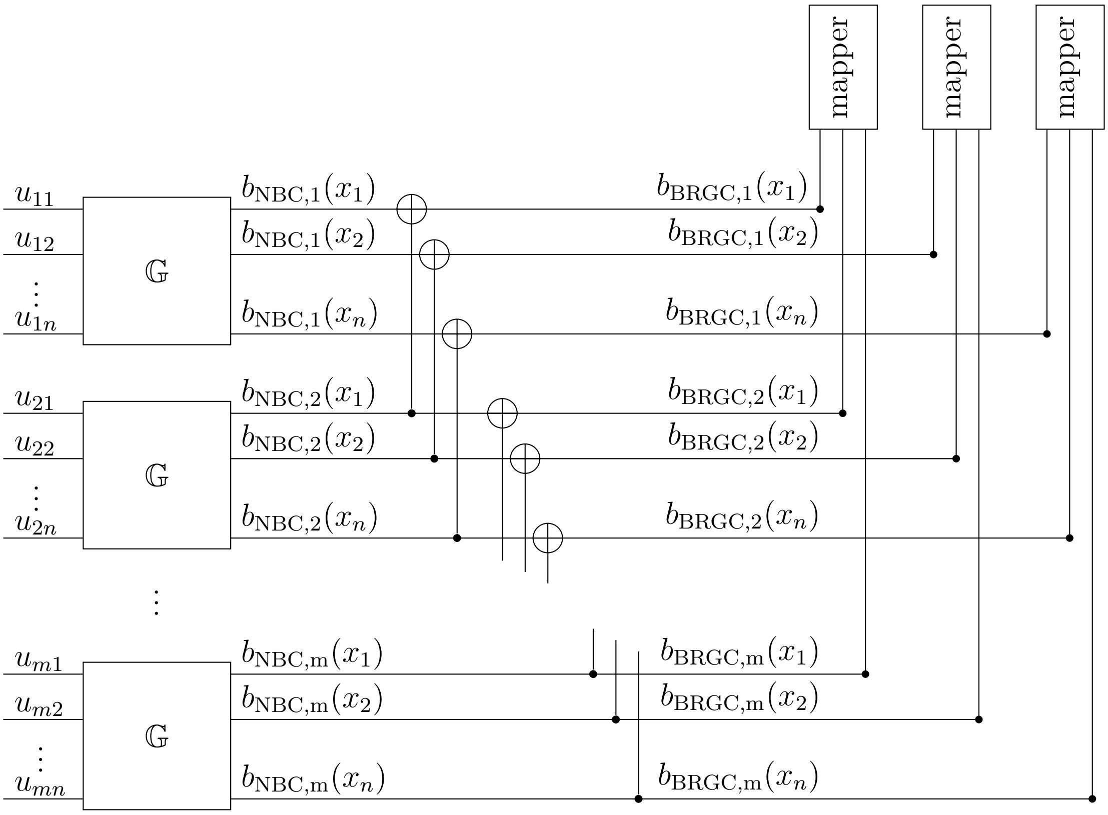


Figure 5.3.: Full PCM transform including the labeling transform from NBC to BRGC.

introduce a labeling transform. NBC and BRGC relate to each other via

$$b_{\text{BRGC}}(X) = b_{\text{NBC}}(X) \underbrace{\begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 1 \end{bmatrix}}_{:=\mathbf{L}_{\text{NBC}\to\text{BRGC}}} \tag{5.1}$$

where $\mathbf{L}_{\text{NBC}\to\text{BRGC}}$ is a $m \times m$ matrix. The full PCM transformation matrix for ASK including the labeling transform is

$$\mathbf{G} = \mathbf{L}_{\text{NBC}\to\text{BRGC}} \otimes \mathbb{G} \tag{5.2}$$

which is illustrated in Fig. 5.3. The overall transform for PAM is

$$\mathbf{G} = \mathbf{I}_m \otimes \mathbb{G}. \tag{5.3}$$

5.2. Systematic Encoding with Dynamic Frozen Bits

We can find the optimal positions of the frozen bits for a given channel, constellation and distribution with one of the methods described in Sec. 4.2, i.e., the optimal sets of frozen bits $\mathcal{F}$ and unfrozen bits $\mathcal{U}$ are fixed for a given PAS scenario. In chapter 3.4.1, we described systematic encoding for polarizing codes. We need to find a set $\mathcal{A}$ such that the sub-matrix $\mathbf{G}_{\mathcal{U}\mathcal{A}}$ from matrix $\mathbf{G}$ as defined in (5.2) and (5.3) has full rank. The set $\mathcal{A}$ has the indices of the systematic bits. Usually, we choose $\mathcal{A} = \mathcal{U}$ for polar codes. For PC-PAS this is in general not possible, as we show in Sec. 5.2.1 and 5.2.2. In [76], dynamic frozen bits are used to perform systematic encoding when $\mathcal{A} \neq \mathcal{U}$. The principle of dynamic frozen bits was described in [124] to improve the minimum distance of polar codes. Instead of setting the frozen bits to predefined values, e.g., setting them to zero, the value of the frozen bit u_i is a function of one or more unfrozen bits u_j with $j < i$. Since $j < i$, these bits are already decoded and assumed known at the decoder when the value of u_i is needed. In the following, we show the general idea of systematic encoding with $\mathcal{U} \neq \mathcal{A}$ as described in [76].

Consider two sets $\mathcal{U}$ and $\mathcal{A}$ with $\mathcal{U} \neq \mathcal{A}$, $|\mathcal{U}| = |\mathcal{A}| = k$ and $\operatorname{rank}(\mathbf{G}_{\mathcal{U}\mathcal{A}}) < k$. For PAS we need to choose $k = (m-1)n + \gamma n$; see Sec. 2.6. We define an encoding set $\mathcal{U}_{\text{enc}}$ with $|\mathcal{U}_{\text{enc}}| = k$ and $\mathcal{U}_{\text{enc}} \neq \mathcal{U}$. Since $\mathcal{U}_{\text{enc}} \neq \mathcal{U}$, the set $\mathcal{U}_{\text{enc}}$ contains frozen bits. We define these frozen bits within $\mathcal{U}_{\text{enc}}$ as

$$\mathcal{F}_{\text{dyn}} = \mathcal{U}_{\text{enc}} \setminus \mathcal{U} \tag{5.4}$$

and call them dynamically frozen bits. We denote the unfrozen bits that are not within the set $\mathcal{U}_{\text{enc}}$ as

$$\mathcal{U}_{\text{cp}} = \mathcal{U} \setminus \mathcal{U}_{\text{enc}}. \tag{5.5}$$

Consider transmitting the data $\boldsymbol{d}$ of dimension k. The codeword $\boldsymbol{c}$ must satisfy

$$\boldsymbol{c}_{\mathcal{A}} = \boldsymbol{d} \tag{5.6}$$

for systematic encoding. The idea of systematic encoding with dynamically frozen bits is to choose $\boldsymbol{u}_{\mathcal{U}_{\text{enc}}}$ so that (5.6) is satisfied. The actual frozen bits are set to zero:

$$\boldsymbol{u}_{\mathcal{F}\setminus\mathcal{F}_{\text{dyn}}} = \mathbf{0}. \tag{5.7}$$

Since the dynamically frozen bits satisfy $\mathcal{F}_{\text{dyn}} \in \mathcal{U}_{\text{enc}}$, we copy these bits to the unfrozen bits $\mathcal{U}_{\text{cp}}$, i.e., we choose $\boldsymbol{u}_{\mathcal{U}_{\text{cp}}}$ such that

$$\boldsymbol{u}_{\mathcal{U}_{\text{cp}}} = \boldsymbol{u}_{\mathcal{F}_{\text{dyn}}}. \tag{5.8}$$

Fig. 5.4 illustrates the sets to facilitate further derivations.

Consider u_i where $i \in \mathcal{F}_{\text{dyn}}$. We have an index $j \in \mathcal{U}_{\text{cp}}$ such that $u_i = u_j$. The value of

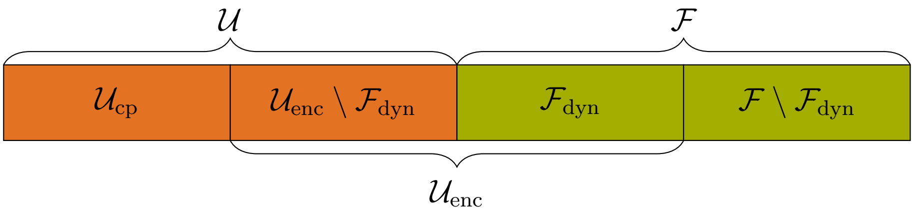


Figure 5.4.: Illustration of index sets in PC-PAS.

the dynamically frozen bit u_i must be known when the SC decoder is processing u_i, i.e., the bit u_j must be decoded before u_i and thus $j < i$. As a result, the first condition that must be satisfied for all dynamically frozen bits is

$$|\{j \in \mathcal{F}_{\mathrm{dyn}} \;:\; j \leq i\}| \leq |\{j \in \mathcal{U}_{\mathrm{cp}} \;:\; j \leq i\}| \, \forall i \in \mathcal{F}_{\mathrm{dyn}}. \tag{5.9}$$

Consider $i \in \mathcal{F}_{\mathrm{dyn}}$. We have $|\{j \in \mathcal{F}_{\mathrm{dyn}} \;:\; j \leq i\}|$ dynamically frozen bits with index less or equal than i. We need at most the same number of bits in the set $\mathcal{U}_{\mathrm{cp}}$ with an index smaller or equal than i since the value of all dynamically frozen bits must be known before.

We can write encoding as

$$\boldsymbol{c} = \boldsymbol{u}\mathbf{G} \tag{5.10}$$

$$\stackrel{(a)}{=} \boldsymbol{u}_{\mathcal{F}\setminus\mathcal{F}_{\mathrm{dyn}}}\mathbf{G}_{\mathcal{F}\setminus\mathcal{F}_{\mathrm{dyn}}} + \boldsymbol{u}_{\mathcal{U}_{\mathrm{cp}}}\mathbf{G}_{\mathcal{U}_{\mathrm{cp}}} + \boldsymbol{u}_{\mathcal{U}_{\mathrm{enc}}}\mathbf{G}_{\mathcal{U}_{\mathrm{enc}}} \tag{5.11}$$

$$\stackrel{(b)}{=} \boldsymbol{u}_{\mathcal{U}_{\mathrm{cp}}}\mathbf{G}_{\mathcal{U}_{\mathrm{cp}}} + \boldsymbol{u}_{\mathcal{U}_{\mathrm{enc}}}\mathbf{G}_{\mathcal{U}_{\mathrm{enc}}} \tag{5.12}$$

$$\stackrel{(c)}{=} \boldsymbol{u}_{\mathcal{U}_{\mathrm{cp}}}\mathbf{G}_{\mathcal{U}_{\mathrm{cp}}} + \begin{bmatrix} \boldsymbol{u}_{\mathcal{F}_{\mathrm{dyn}}} & \boldsymbol{u}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}}} \end{bmatrix} \begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\mathrm{dyn}}} \\ \mathbf{G}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}}} \end{bmatrix} \tag{5.13}$$

$$\stackrel{(d)}{=} \begin{bmatrix} \boldsymbol{u}_{\mathcal{F}_{\mathrm{dyn}}} & \boldsymbol{u}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}}} \end{bmatrix} \begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\mathrm{dyn}}} + \mathbf{G}_{\mathcal{U}_{\mathrm{cp}}} \\ \mathbf{G}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}}} \end{bmatrix}. \tag{5.14}$$

We separate the vector according to the three sets defined above in (a) and set the actual frozen bits to zero in (b). In (c), we separate $\boldsymbol{u}_{\mathcal{U}_{\mathrm{enc}}}$ into the dynamically frozen bits and the remaining bits. We use $\boldsymbol{u}_{\mathcal{U}_{\mathrm{cp}}} = \boldsymbol{u}_{\mathcal{F}_{\mathrm{dyn}}}$ from (5.8) in (d).

The systematic part of the codeword $\boldsymbol{c}_{\mathcal{A}}$ can be expressed as

$$\boldsymbol{c}_{\mathcal{A}} = \begin{bmatrix} \boldsymbol{u}_{\mathcal{F}_{\mathrm{dyn}}} & \boldsymbol{u}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}}} \end{bmatrix} \begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\mathrm{dyn}},\mathcal{A}} + \mathbf{G}_{\mathcal{U}_{\mathrm{cp}},\mathcal{A}} \\ \mathbf{G}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}},\mathcal{A}} \end{bmatrix} \tag{5.15}$$

and we obtain $\boldsymbol{c}_{\mathcal{A}} = \boldsymbol{d}$ from (5.6) by choosing

$$\begin{bmatrix} \boldsymbol{u}_{\mathcal{F}_{\mathrm{dyn}}} & \boldsymbol{u}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}}} \end{bmatrix} = \Pi(\boldsymbol{u}_{\mathrm{enc}}) = \boldsymbol{d} \begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\mathrm{dyn}},\mathcal{A}} + \mathbf{G}_{\mathcal{U}_{\mathrm{cp}},\mathcal{A}} \\ \mathbf{G}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}},\mathcal{A}} \end{bmatrix}^{-1}. \tag{5.16}$$

where the rank of the matrix has to be k to be invertible. Π denotes a permutation of the vector elements.

We conclude with the following Theorem; see (5.9) and (5.16).

Theorem 5.1. We can perform systematic encoding with dynamically frozen bits as described above for a given set of frozen bits $\mathcal{F}$ and unfrozen bits $\mathcal{U}$ if the sets $\mathcal{A}$ and $\mathcal{U}_{\text{enc}}$ are chosen such that

$$|\{j \in \mathcal{F}_{\text{dyn}} \ : \ j \leq i\}| \leq |\{j \in \mathcal{U}_{\text{cp}} \ : \ j \leq i\}|, \quad \forall i \in \mathcal{F}_{\text{dyn}} \tag{5.17}$$

and

$$\text{rank}\left(\begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\text{dyn}},\mathcal{A}} + \mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{A}} \\ \mathbf{G}_{\mathcal{U}_{\text{enc}} \setminus \mathcal{F}_{\text{dyn}},\mathcal{A}} \end{bmatrix}\right) = k. \tag{5.18}$$

We show in the following two sections how these sets can be chosen in case of NBC labeling for PAM transmission, and BRGC labeling for ASK transmission. To simplify notation in the following derivations, we define the sets

$$\mathcal{Z}_1 = \{i \in \mathbb{N} \ : \ 1 \leq i \leq n\} \tag{5.19}$$
$$\mathcal{Z}_2^m = \{i \in \mathbb{N} \ : \ n+1 \leq i \leq mn\} \tag{5.20}$$
$$\mathcal{Z}_1^{m-1} = \{i \in \mathbb{N} \ : \ 1 \leq i \leq (m-1)n\}. \tag{5.21}$$

5.2.1. NBC Labeling

For NBC labeling, the first bit-level is the "sign"-bit and the bit-levels $2, \ldots, m$ are the "amplitude"-bits that must be systematic in the codeword, as shown in Fig. 5.1b. This implies the indices $n+1, \ldots, mn$ must be part of the set $\mathcal{A}$. Recall from Sec. 2.6 that the number of unfrozen bits for PAS must be

$$|\mathcal{U}| = (m-1)n + \gamma n. \tag{5.22}$$

We choose all frozen bits within the bit-levels $2, \ldots, m$ as dynamically frozen bits:

$$\mathcal{F}_{\text{dyn}} = \{i \in \mathcal{F} \ : \ i \geq n\}. \tag{5.23}$$

It follows that the number of unfrozen bits within the bit-levels $2, \ldots, m$ is $(m-1)n - |\mathcal{F}_{\text{dyn}}|$ and the number of unfrozen bits within the first bit-level is $|\mathcal{F}_{\text{dyn}}| + \gamma n$ to obtain the number from (5.22). The set of unfrozen bits within the first bit-level is

$$\mathcal{U}_1 = \{i \in \mathcal{U} \ : \ i \leq n\}. \tag{5.24}$$

We separate the set $\mathcal{U}_1$ into the two sets $\mathcal{U}_{\text{cp}} \subseteq \mathcal{U}_1$ and $\mathcal{U}' \subseteq \mathcal{U}_1$ such that $|\mathcal{U}_{\text{cp}}| = |\mathcal{F}_{\text{dyn}}|$, $|\mathcal{U}'| = \gamma n$ and $\mathcal{U}_{\text{cp}} \cap \mathcal{U}' = \emptyset$.

We now define $\mathcal{A} = \mathcal{U}_{\text{enc}}$ for NBC labeling with

$$\mathcal{A} = \mathcal{U}_{\text{enc}} = \mathcal{U}' \cup \mathcal{Z}_2^m \tag{5.25}$$

Thus, the encoding set $\mathcal{U}_{\text{enc}}$ consists of all indices corresponding to the bit-levels $2, \dots, m$ and the unfrozen bit set $\mathcal{U}'$ from the first bit-level. In the following, we show that our choice satisfies the conditions of Theorem 5.1. We start by showing (5.17). Since $\mathcal{F}_{\text{dyn}} \subseteq \mathcal{Z}_2^m$ and $\mathcal{U}_{\text{cp}} \subseteq \mathcal{Z}_1$, we have

$$|\{j \in \mathcal{U}_{\text{cp}} \ : \ j \leq i\}| = |\mathcal{F}_{\text{dyn}}|, \quad \forall i \in \mathcal{F}_{\text{dyn}} \tag{5.26}$$

and (5.17) is satisfied.

We now show the choice of $\mathcal{A}$ and $\mathcal{U}_{\text{enc}}$ in (5.25) satisfies (5.18). Using an NBC labeling, we have

$$\mathbf{G} = \mathbf{I}_m \otimes \mathbb{G} = \begin{bmatrix} \mathbb{G} & & & \mathbf{0} \\ & \mathbb{G} & & \\ & & \ddots & \\ \mathbf{0} & & & \mathbb{G}, \end{bmatrix} \tag{5.27}$$

i.e., $\mathbf{G}$ is a lower triangular matrix with only ones on the diagonal since $\mathbf{G}_n$ is lower triangular for polar codes. We have

$$\begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\text{dyn}},\mathcal{A}} + \mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{A}} \\ \mathbf{G}_{\mathcal{U}_{\text{enc}} \setminus \mathcal{F}_{\text{dyn}},\mathcal{A}} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\text{dyn}},\mathcal{A}} \\ \mathbf{G}_{\mathcal{U}_{\text{enc}} \setminus \mathcal{F}_{\text{dyn}},\mathcal{A}} \end{bmatrix} + \begin{bmatrix} \mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{A}} \\ \mathbf{0} \end{bmatrix} = \mathbf{P}\mathbf{G}_{\mathcal{U}_{\text{enc}},\mathcal{A}} + \begin{bmatrix} \mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{A}} \\ \mathbf{0} \end{bmatrix} \tag{5.28}$$

where $\mathbf{P}$ is a row permutation matrix and $\mathbf{G}_{\mathcal{U}_{\text{enc}},\mathcal{A}}$ is a lower triangular matrix with only ones on the diagonal since we have chosen $\mathcal{U}_{\text{enc}} = \mathcal{A}$. The term $\mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{A}}$ is added to the rows of $\mathbf{G}_{\mathcal{U}_{\text{enc}},\mathcal{A}}$ that correspond to the dynamically frozen bits $\mathcal{F}_{\text{dyn}}$. We write this part as

$$\mathbf{G}_{\mathcal{F}_{\text{dyn}},\mathcal{A}} + \mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{A}} \overset{(a)}{=} \begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\text{dyn}},\mathcal{U}'} & \mathbf{G}_{\mathcal{F}_{\text{dyn}},\mathcal{Z}_2^m} \end{bmatrix} + \begin{bmatrix} \mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{U}'} & \mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{Z}_2^m} \end{bmatrix} \tag{5.29}$$

$$= \begin{bmatrix} \mathbf{0} & \underbrace{\mathbf{G}_{\mathcal{F}_{\text{dyn}},\mathcal{Z}_2^m}}_{\text{holds diagonal elements of } \mathbf{G}_{\mathcal{U}_{\text{enc}},\mathcal{A}}} \end{bmatrix} + \begin{bmatrix} \mathbf{G}_{\mathcal{U}_{\text{cp}},\mathcal{U}'} & \mathbf{0} \end{bmatrix} \tag{5.30}$$

where we split the set $\mathcal{A}$ into $\mathcal{U}'$ and $\mathcal{Z}_2^m$ in (a). The terms that are added to $\mathbf{G}_{\mathcal{F}_{\text{dyn}},\mathcal{A}}$ affect the matrix $\mathbf{G}_{\mathcal{U}_{\text{enc}},\mathcal{A}}$ only to the left of the diagonal elements. The resulting matrix is therefore a lower triangular matrix with ones on the diagonal and with full rank, and (5.18) is satisfied for the choice of $\mathcal{A}$ and $\mathcal{U}_{\text{enc}}$ in (5.25). We illustrate this in Fig. 5.5. $\mathcal{U}_{\text{cp}}$ has only indices from the first block depicted by the red rows. We add these rows to the rows that correspond to the dynamically frozen bits that can occur only in the second to last block. These modifications do not change the lower triangular structure of the matrix, and the resulting matrix has full rank.

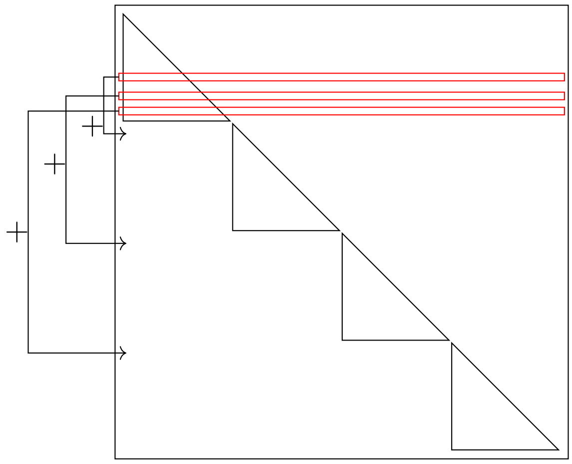


Figure 5.5.: Schematic representation of the matrix $\mathbf{G}$ for NBC labeling with $m = 4$ and the modifications for systematic encoding.

5.2.2. BRGC Labeling

For ASK with BRGC labeling, the last bit-level is the "sign"-bit and the levels $1, \ldots, m-1$ are the "amplitude"-bits that need to be systematic; see Fig. 5.1a. This implies the indices $1, \ldots, (m-1)n$ must be part of the set $\mathcal{A}$.

We again define the sets $\mathcal{U}_1$, $\mathcal{U}_{\text{cp}}$ and $\mathcal{U}'$ as described in section 5.2.1 for NBC labeling. We choose

$$\mathcal{U}_{\text{enc}} = \mathcal{U}' \cup \mathcal{Z}_2^m \tag{5.31}$$

and

$$\mathcal{A} = \mathcal{Z}_1^{m-1} \cup \{(m-1)n + i \ : \ i \in \mathcal{U}'\}. \tag{5.32}$$

Additionally, we make the following assumption.

Assumption 5.1. *If $i \in \mathcal{U}_1$ is an unfrozen bit within the first component block, then the i-th bits in blocks $2, \ldots, m$ are also unfrozen bits. We can write this condition as*

$$\{j \in \mathcal{F}_{\text{dyn}} \ : \ ((j-1) \mod n) + 1 = i\} = \emptyset \quad \forall i \in \mathcal{U}_1. \tag{5.33}$$

For the channels considered in this thesis and under MSD, this assumption is usually valid since the quality of the bit-channels gets better from stage to stage, i.e., if a bit is unfrozen in the first block then it will also be unfrozen in the preceding blocks.

We must show that (5.17) and (5.18) are satisfied for the choice of $\mathcal{U}_{\text{enc}}$ and $\mathcal{A}$ in (5.31) and (5.32), respectively. The choice of $\mathcal{A}$ and $\mathcal{U}_{\text{enc}}$ results in the same properties of $\mathcal{U}'$ and $\mathcal{F}_{\text{dyn}}$ that are used in 5.2.1 to show that (5.17) is satisfied. Therefore, condition (5.17) is also satisfied in this case.

We now show that the choice also satisfies (5.18). For BRGC labeling, we have

$$\mathbf{G} = \mathbf{L}_{\mathrm{NBC}\to\mathrm{BRGC}} \otimes \mathbb{G} = \begin{bmatrix} \mathbb{G} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbb{G} & \mathbb{G} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbb{G} & \mathbb{G} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbb{G} & \mathbb{G} \end{bmatrix}. \tag{5.34}$$

We must show the matrix

$$\begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\mathrm{dyn}},\mathcal{A}} + \mathbf{G}_{\mathcal{U}_{\mathrm{cp}},\mathcal{A}} \\ \mathbf{G}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}},\mathcal{A}} \end{bmatrix} \overset{(a)}{=} \begin{bmatrix} \mathbf{G}_{\mathcal{F}_{\mathrm{dyn}},\mathcal{A}} + [\mathbb{G}_{\mathcal{U}_{\mathrm{cp}}}, \mathbf{0}] \\ \mathbf{G}_{\mathcal{U}_{\mathrm{enc}}\setminus\mathcal{F}_{\mathrm{dyn}},\mathcal{A}} \end{bmatrix} \tag{5.35}$$

$$\overset{(b)}{=} \mathbf{P} \begin{bmatrix} \mathbb{G}_{\mathcal{U}'} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbb{G} + \mathbf{M}^{(2)} & \mathbb{G} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{M}^{(3)} & \mathbb{G} & \mathbb{G} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{M}^{(m)} & \mathbf{0} & \mathbf{0} & \dots & \mathbb{G} & \mathbb{G}_{\{1,\dots,n\},\mathcal{U}'} \end{bmatrix} \tag{5.36}$$

$$\mathbf{G}'$$

has full rank. Since $\mathcal{U}_{\mathrm{cp}} \subseteq \mathcal{Z}_1$, we have $\mathbf{G}_{\mathcal{U}_{\mathrm{cp}}} = [\mathbb{G}_{\mathcal{U}_{\mathrm{cp}}}, \mathbf{0}]$ in (a); see the first row in (5.34). The matrix $\mathbf{P}$ in (b) is a row permutation matrix that sorts the rows as they appear in $\mathbf{G}$. The addition of $\mathbf{M}^{(l)}$ for $l = 2, \dots, m$ corresponds to the term $\mathbb{G}_{\mathcal{U}_{\mathrm{cp}}}$ in (5.35). The rows of $\mathbb{G}_{\mathcal{U}_{\mathrm{pc}}}$ are distributed to the matrices $\mathbf{M}^{(l)}$ such that the equation above holds. The rows of $\mathbf{M}^{(l)}$ that do not correspond to a dynamically frozen bit are all-zero rows.

We first show that the sub-matrix $\mathbf{G}'$ from (5.36) is invertible. We perform the following elementary row operations

$$\mathbf{P}_1\mathbf{G}' = \begin{bmatrix} \mathbf{1}_n & \mathbf{1}_n & \dots & \mathbf{1}_n & \mathbf{1}_n \\ \mathbf{1}_n & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_n & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{1}_n & \mathbf{0} \end{bmatrix} \mathbf{G}' \tag{5.37}$$

$$= \begin{bmatrix} \mathbb{G} + \sum_{l=2}^{m} \mathbf{M}^{(l)} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbb{G} + \mathbf{M}^{(2)} & \mathbb{G} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{M}^{(3)} & \mathbb{G} & \mathbb{G} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}^{(m-1)} & \mathbf{0} & \mathbf{0} & \dots & \mathbb{G} \end{bmatrix} \tag{5.38}$$

and obtain a block lower triangular matrix. This matrix is invertible if all block matrices on the diagonal are invertible. All matrices $\mathbb{G}$ are polar transforms and therefore have full rank. It remains to show that the matrix

$$\mathbb{G} + \sum_{l=2}^{m} \mathbf{M}^{(l)} \tag{5.39}$$

has full rank.

Consider index $i \in \mathcal{F}_{\text{dyn}}$ of a dynamically frozen bit. We can write

$$i = (l-1)n + j \tag{5.40}$$

so i is the j-th bit of the $(l+1)$-th component code. Note that $2 \leq l < m$ since there are no dynamically frozen bits within the first component code. We now have

$$\mathbf{M}_j^{(l)} = \mathbb{G}_k \tag{5.41}$$

with $k \in \mathcal{U}_{\text{cp}}$, i.e., we add row $\mathbb{G}_k$ to row $\mathbb{G}_j$ in (5.39). If $j \neq k$, this is a linear combination of linear independent rows and does not change the rank of the matrix. Index j is a dynamically frozen bit within the l-th component code. Due to assumption 5.1, we have $j \neq k$ and conclude that the matrix $\mathbf{G}'$ from (5.36) has full rank.

Consider now the matrix from (5.36):

$$\mathbf{P} \begin{bmatrix} [\mathbb{G}_{\mathcal{U}'}, \mathbf{0}] & \mathbf{0} \\ \mathbf{G}' & \begin{bmatrix} \mathbf{0} \\ \mathbb{G}_{\{1,\dots,n\},\mathcal{U}'} \end{bmatrix} \end{bmatrix} \tag{5.42}$$

where $[\mathbb{G}_{\mathcal{U}'}, \mathbf{0}]$ is a $\gamma n \times (m-1)n$ matrix with rank γn and $[\mathbf{0}, \mathbb{G}_{\{1,\dots,n\},\mathcal{U}'}]^{\mathrm{T}}$ is a $(m-1)n \times \gamma n$ matrix with rank γn. The inverse of this matrix can be computed if the Schur complement

$$\mathbf{S} = -\begin{bmatrix} \mathbb{G}_{\mathcal{U}'} & \mathbf{0} \end{bmatrix} \mathbf{G}' \begin{bmatrix} \mathbf{0} \\ \mathbb{G}_{\{1,\dots,n\},\mathcal{U}} \end{bmatrix} \tag{5.43}$$

has full rank. We have shown that the $(m-1)n \times (m-1)n$ matrix $\mathbf{G}'$ has full rank. The matrices $[\mathbb{G}_{\mathcal{U}'}, \mathbf{0}]$ and γn and $[\mathbf{0}, \mathbb{G}_{\{1,\dots,n\},\mathcal{U}'}]^{\mathrm{T}}$ are both full-rank matrices with rank γn. It follows that the Schur complement $\mathbf{S}$ has also full rank, i.e., $\operatorname{rank} \mathbf{S} = \gamma n$. Summarizing, the conditions (5.17) and (5.18) are satisfied for the presented choice of the sets $\mathcal{A}$ and $\mathcal{U}_{\text{enc}}$ for BRGC labeling of ASK transmission.

5.3. Decoding PC-PAS

Decoding PC-PAS is done by the multistage list decoding algorithm described in chapter 4.3. An outer CRC-code selects valid codewords from the resulting list of codewords.

If a CCDM is used for PC-PAS, we can use a type check in addition to the CRC-code to choose valid codewords from the list [76, 125]. The output of a CCDM has constant type property, i.e., each possible output has the same empirical distribution. Checking this empirical distribution in list decoding is denoted as type check.

5.4. Numerical Results

To compare the performance of PC-PAS with uniform PCM, we simulate PC-PAS with a CCDM [95] and show results with and without the type check.

5.4.1. Code and Distribution Optimization

To compare the performance, we heuristically optimize the following parameters:

- ▷ n_{CRC}: number of CRC-bits;
- ▷ $\mathrm{SNR}_{\mathrm{code}}$: the SNR used for constructing the codes according to chapter 4.2.1;
- ▷ SNR_{P_X}: the SNR used to optimize the distribution.

We optimize these parameters via a grid search such that the FER at an optimization SNR $\mathrm{SNR}_{\mathrm{opt}}$ is minimized. We build the grid in steps of 0.25 dB for the SNR values and $n_{\mathrm{CRC}} = 3, 4, \dots, 24$. The corresponding CRC polynomials are taken from [126] and are listed in Table A.1. We use different SNR values $\mathrm{SNR}_{\mathrm{code}}$ than $\mathrm{SNR}_{\mathrm{code}}$ for code construction since the resulting codes can have better performance under list decoding with outer CRC-codes than the ones constructed at the optimization SNR; see [108].

The distribution is chosen as follows. For each SNR_{P_X}, we choose the MB distribution P_X that maximizes the rate at SNR_{P_X}; see [40, Sec. III-C]. For PAM constellations, we use staircase distributions as in Fig. 5.1b. The distribution of the steps is chosen as a MB distribution as well. The corresponding amplitude distribution is then quantized for the CCDM as described in [95, Sec. III-A].

Table 5.1 summarizes the simulated scenarios and lists the parameters that resulted from the grid search optimization. We perform multistage list decoding with list size 32 in all PCM and PC-PAS scenarios. For the ASK scenarios we include LDPC codes as defined the 5G new radio standard [2] for comparison. LDPC codes with PAS are based on BMD as described in [40]. For LDPC codes, we only optimized the parameter $\mathrm{SNR}_{\mathrm{P_X}}$ since the other parameters are not relevant.

5.4.2. Results for 8-ASK Transmission

Fig. 5.6-5.8 show results for ASK transmission with block lengths $n = 64, 128, 1024$, respectively. The first two scenarios are with 8-ASK and the latter with 16-ASK. Observe the effect of the type check for decoding PC-PAS. The gain of performing the type check decreases as n increases. We have gains of 0.31 dB, 0.16 dB and 0.05 dB for $n = 64, 128, 1024$,

Scenario	n [symbols]	rate [bpcu]	$\mathrm{SNR}_{\mathrm{opt}}$ [dB]	n_{CRC} [bits]	$\mathrm{SNR}_{\mathrm{code}}$ [dB]	SNR_{P_X} [dB]	type check
PCM 4-PAM	64	1.25	14.5	7	16.75		
PC-PAS 4-PAM	64	1.25	13.5	5	14.25	11	no
PC-PAS 4-PAM	64	1.25	13.5	3	15.75	9.75	yes
PCM 8-ASK	64	1.75	13.5	5	17.75		
PC-PAS 8-ASK	64	1.75	13	4	16.75	12.25	no
PC-PAS 8-ASK	64	1.75	13	3	14.5	12.75	yes
LDPC-PAS 8-ASK	64	1.75	14.5			5.5	
8-ASK uniform	128	2	14.5	7	16.25		
PC-PAS 8-ASK	128	2	14	7	15.25	11.25	no
PC-PAS 8-ASK	128	2	13.5	3	15	11	yes
LDPC-PAS 8-ASK	128	2	15			11.25	
PCM 16-ASK	1024	3	20.2	6	20.45		
PC-PAS 16-ASK	1024	3	19.25	7	19.75	18.75	no
PC-PAS 16-ASK	1024	3	19.25	3	19.75	18.25	yes
LDPC-PAS 16-ASK	1014	3	19.5			18.5	

Table 5.1.: Code parameters resulting from the optimization.

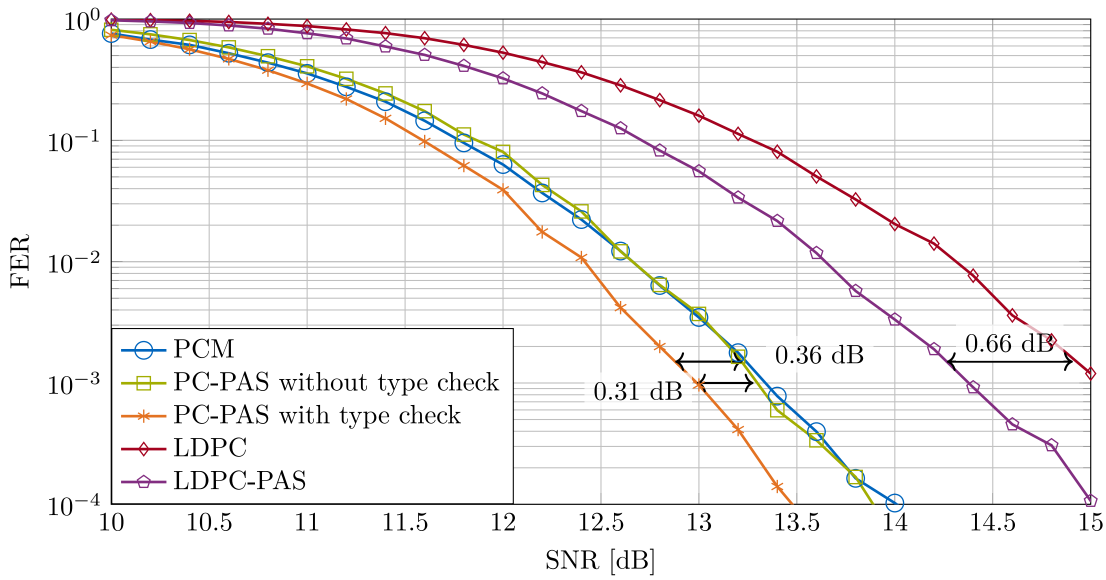


Figure 5.6.: FER of PC-PAS with 8-ASK for $n = 64$ and rate 1.75 bpcu.

respectively. As the type check partially compensates for the CCDM rate loss, this is in line with the results from [95] where the CCDM rate loss decreases as n grows. We also observe from Table 5.1 that the type check partially takes over the task of the CRC check. The optimal CRC length is reduced in all scenarios where a type check is used.

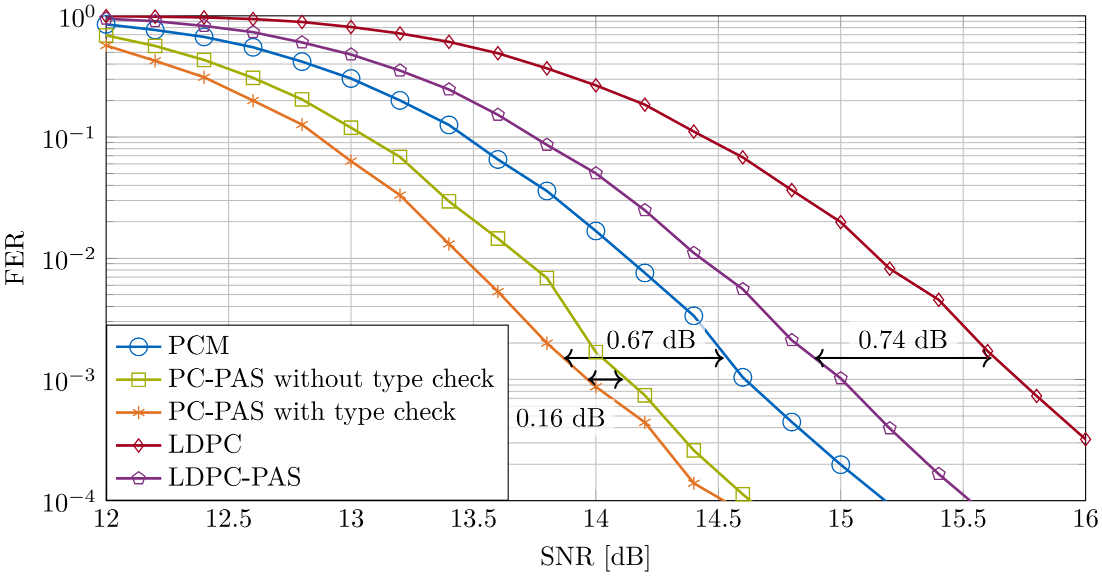


Figure 5.7.: FER of PC-PAS with 8-ASK for $n = 128$ and rate 2 bpcu.

Observe that PCM and PC-PAS are good choices at short block lengths, where they outperform the LDPC codes by more than 1 dB. As n grows, the performance of the LDPC schemes approaches the performance of the polar schemes. But for 16-ASK with $n = 1024$, the polar schemes are still more energy efficient. The gains obtained by using non-uniform input distributions (shaping gain) are in the expected range (cf. [40, Table I]) except for the shaping gain of PC-PAS for $n = 64$ that is smaller than expected. This might result from different small block length effects and is not further investigated here.

5.4.3. Results for 4-PAM Transmission

Fig. 5.9 shows the FER performance of PCM and PC-PAS for 4-PAM transmission with $n = 64$ and an average power constrained AWGN channel. A shaping gain of 1 dB is obtained by using the sub-optimal staircase MB distribution with PC-PAS instead of uniformly distributed PCM.

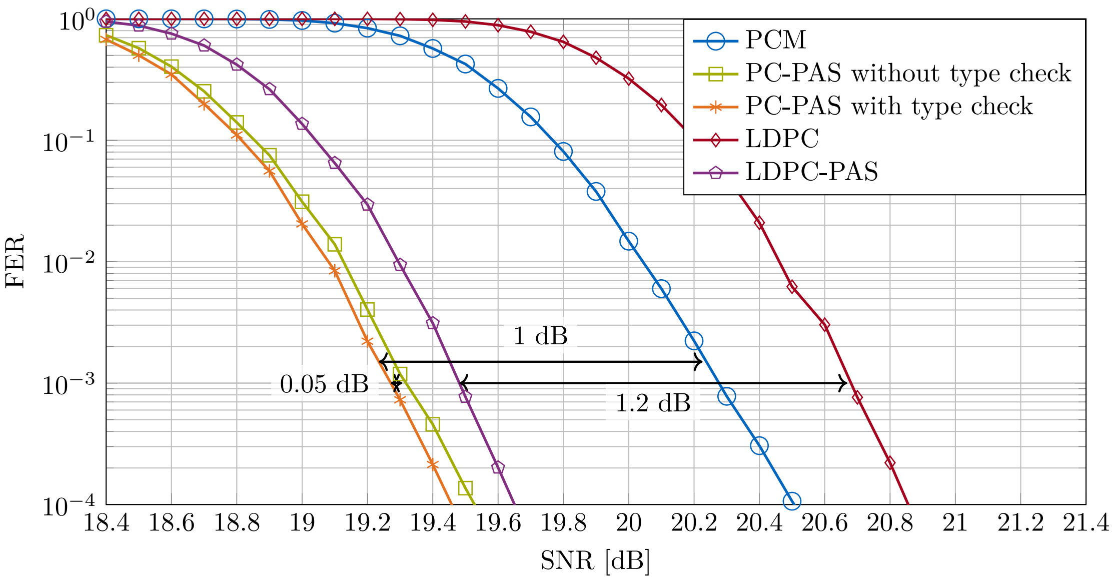


Figure 5.8.: FER of PC-PAS with 16-ASK for $n = 1024$ and rate 3 bpcu.

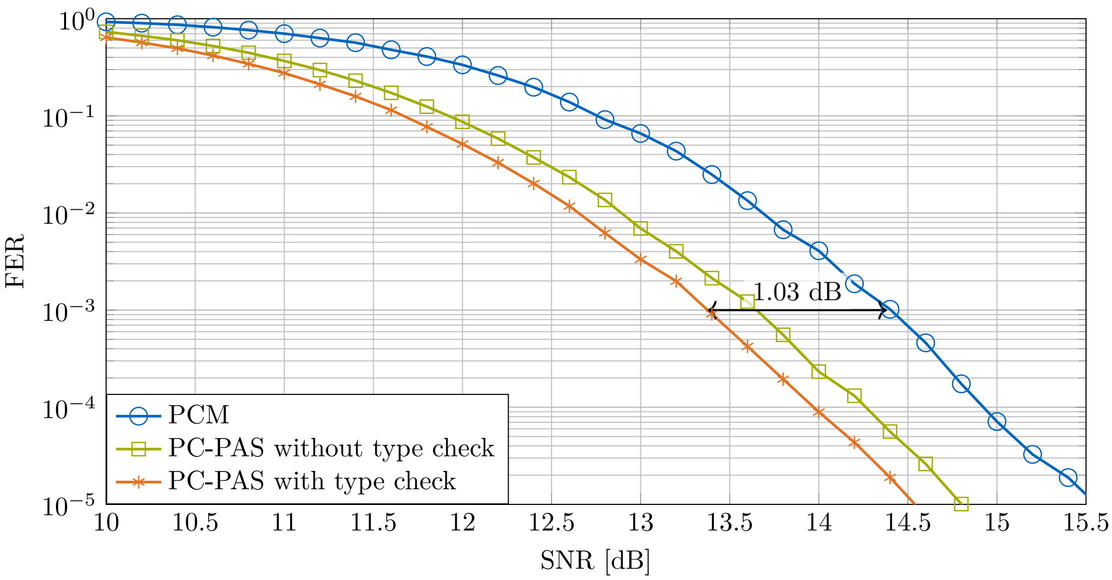


Figure 5.9.: FER of PC-PAS with 4-PAM for $n = 64$ and rate 1.25 bpcu.

6

Probabilistic Shaping with Polar and BMERA codes

The previous chapter showed how polar codes can be combined with PAS. A DM is used to generate non-uniform distributions. The paper [29] shows that polar codes can produce non-uniform distributions by themselves. Two separate polar codes are used in [31], one for FEC and the other to realize a non-uniform distribution. A multilevel DM based on polar codes is presented in [36]. Schemes where one polar code performs both FEC and distribution matching are presented in [30, 32, 33, 35, 82, 127, 128] In [82], multilevel coded modulation with multistage decoding is combined with the scheme of [29] and it is proved that the capacity of DMCs is achieved. The contribution of this thesis, in addition to [82], are results for BMERA codes using the same scheme. We first recap the scheme from [82] and then present results for BMERA and polar codes and compare them with PC-PAS.

6.1. Binary Shaping with Polar and BMERA Codes

6.1.1. Polarization

We described channel polarization in Sec. 3.3 by the MI terms $\mathbb{I}\left(U_i; \boldsymbol{Y}|U_1^{i-1}\right)$. In [29], the conditional Bhattacharyya parameter [93] (cf. Sec. 2.5) is used to show polarization. One can show the terms $Z(U_i|U_1^{i-1}, \boldsymbol{Y})$ polarize, i.e., the two sets

$$\mathcal{U} = \left\{ i \in 1, \ldots, n : Z(U_i|U_1^{i-1}, \boldsymbol{Y}) < 2^{-n^{\beta}} \right\} \tag{6.1}$$

$$\mathcal{F} = \left\{ i \in 1, \ldots, n : Z(U_i|U_1^{i-1}, \boldsymbol{Y}) \geq 1 - 2^{-n^{\beta}} \right\} \tag{6.2}$$

with any $\beta < \frac{1}{2}$ ($\beta < \frac{1}{2}\log_2 3$ for BMERA codes) have size

$$\lim_{n\to\infty} \frac{1}{n} |\mathcal{U}| = 1 - \mathbb{H}(X|Y) \tag{6.3}$$

$$\lim_{n\to\infty} \frac{1}{n} |\mathcal{F}| = \mathbb{H}(X|Y). \tag{6.4}$$

Polar code can thus achieve the symmetric capacity of a channel, i.e., the capacity that can be achieved by uniformly distributed inputs. For capacity, one must show that the MI $\mathbb{I}(X;Y) = \mathbb{H}(X) - \mathbb{H}(X|Y)$ can be achieved instead of $1 - \mathbb{H}(X|Y)$.

In [29], it is shown that the terms $Z(U_i|U_1^{i-1})$ polarize as well. We define the two sets

$$\mathcal{S} = \left\{i \in 1,\dots,n : Z(U_i|U_1^{i-1}) < 2^{-n^\beta}\right\} \tag{6.5}$$

$$\mathcal{U}_S = \left\{i \in 1,\dots,n : Z(U_i|U_1^{i-1}) \geq 1 - 2^{-n^\beta}\right\} \tag{6.6}$$

which have the properties (see [29])

$$\lim_{n\to\infty} \frac{1}{n} |\mathcal{S}| = 1 - \mathbb{H}(X) \tag{6.7}$$

$$\lim_{n\to\infty} \frac{1}{n} |\mathcal{U}_S| = \mathbb{H}(X). \tag{6.8}$$

These properties can be derived from (6.3) and (6.4) by choosing a channel where the output Y takes the same value with probability one. We call $\mathcal{S}$ the shaping set.

In the following, we describe how [29] combines channel coding and distribution matching. Using property (2.39), we have $\mathcal{S} \subseteq \mathcal{U}$ and $\mathcal{S} \cap \mathcal{F} = \emptyset$. Fig. 6.1 illustrates the relation between the sets $\mathcal{U}$, $\mathcal{F}$, $\mathcal{S}$ and $\mathcal{U}_s$.

We observe the following [29, Thm. 1]:

$$\lim_{n\to\infty} \frac{1}{n} |\mathcal{U} \cap \mathcal{U}_s| = \mathbb{I}(X;Y) \tag{6.9}$$

$$\lim_{n\to\infty} \frac{1}{n} |\mathcal{F} \cup \mathcal{S}| = 1 - \mathbb{I}(X;Y) \tag{6.10}$$

The result is proved in [29].

6.1.2. Code Design

The parameters n and k represent the block length of the code and the dimension, i.e., the number of data bits transmitted per block. Additionally, let s be the cardinality of the shaping set $\mathcal{S}$.

Code design for shaped polar and BMERA codes involves finding good sets $\mathcal{U}$, $\mathcal{F}$, $\mathcal{S}$ and $\mathcal{U}_s$ for the parameters n, k and s and a target distribution $P_X(x)$. The cardinality of the

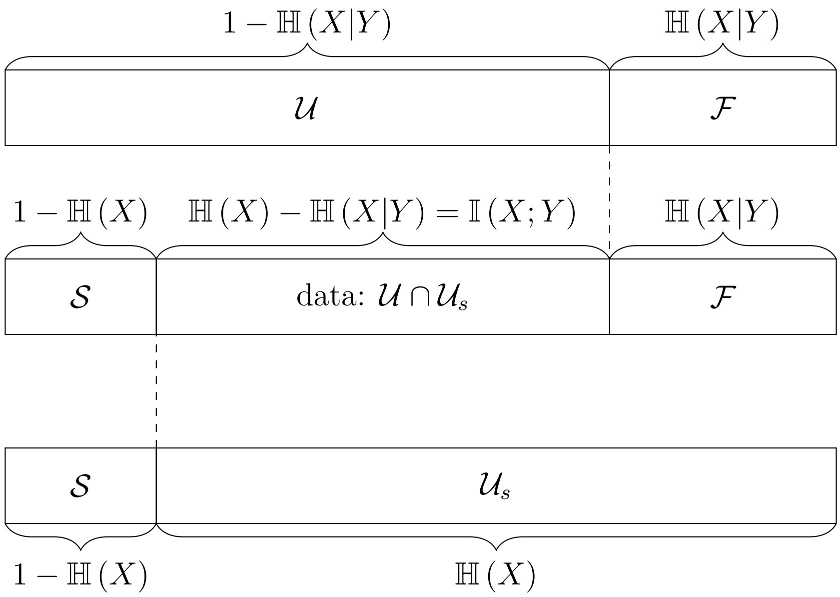


Figure 6.1.: Relation between the sets $\mathcal{U}$, $\mathcal{F}$, $\mathcal{S}$ and $\mathcal{U}_s$.

sets is:

$$\mathcal{U} = k + s \Rightarrow \mathcal{F} = n - (k+s) \tag{6.11}$$

$$\mathcal{S} = s \Rightarrow \mathcal{U}_s = n - s. \tag{6.12}$$

Sets $\mathcal{U}$ and $\mathcal{F}$ can be found using one of the methods described in chapter 4.2 based on Monte Carlo methods or (approximate) density evolution.

The sets $\mathcal{S}$ and $\mathcal{U}_s$ can be found via Monte Carlo simulations. We approximate the metrics $\mathbb{H}\left(U_i|U_1^{i-1}\right)$ instead of the Bhattacharyya parameters $Z(U_i|U_1^{i-1})$ used in (6.5) and (6.6). The conditional entropy and the Bhattacharyya parameter are closely related; see Eq. (2.38). In the k-th Monte Carlo iteration, a codeword $\boldsymbol{x}(k)$ is generated randomly from the desired distribution P_X. The vector $\boldsymbol{u}(k)$ is obtained from $\boldsymbol{x}(k)$ by the inverse polar or BMERA transform. From sampling theory, we can approximate

$$\mathbb{H}\left(U_i|U_1^{i-1}\right) = \sum_{u_1^{i-1}} P_{U_1^{i-1}}\left(u_1^{i-1}\right)\mathbb{H}\left(U_i|U_1^{i-1}=u_1^{i-1}\right) \tag{6.13}$$

$$\approx \frac{1}{N_{\mathrm{MC}}}\sum_{k=1}^{N_{\mathrm{MC}}}\mathbb{H}_2\left(P_{U_i|U_1^{i-1}}\left(0|u_1^{i-1}(k)\right)\right). \tag{6.14}$$

The metrics $P_{U_i|U_1^{i-1}}$ are obtained by running a SC decoder that knows the values u_1^{i-1}, i.e., the decoder uses the true values of the previous bits u_1^{i-1} instead of estimates $\hat{u}_i^{i-1}$. We obtain the set $\mathcal{S}$ by searching for the s indices with the smallest $\mathbb{H}\left(U_i|U_1^{i-1}\right)$.

6.1.3. Encoding

Consider transmitting $\boldsymbol{d}$ of dimension k. For encoding, we set the frozen bits $\boldsymbol{u}_{\mathcal{F}}$ to zero. The indices in $\boldsymbol{u}$ that correspond to $\mathcal{U} \cap \mathcal{U}_s$ are used to transmit data, i.e., $\boldsymbol{u}_{\mathcal{U} \cap \mathcal{U}_s} = \boldsymbol{d}$. According to (6.5), a bit U_i with $i \in \mathcal{S}$ is almost deterministic given the previous U_1^{i-1}, i.e., we can determine their values by an SC or SCL decoder. The SC decoder runs with $\mathcal{U}_s$ as the frozen set containing the actual frozen set $\mathcal{F}$ and data set $\mathcal{U} \cap \mathcal{U}_s$. When the decoder arrives at decoding step $i \in \mathcal{S}$, the decoder sets

$$u_i = \begin{cases} 0 & \text{if } P_{U_i|U_1^{i-1}}(0|u_1^{i-1}) > P_{U_i|U_1^{i-1}}(1|u_1^{i-1}) \\ 1 & \text{else.} \end{cases} \tag{6.15}$$

As input we use the probabilities $P_{X_i}(x_i)$ instead of $P_{X_i|Y_i}(x_i|y_i)$ as for channel decoding. We transform the input vector $\boldsymbol{U}$ and obtain $\boldsymbol{X}$ that can be transmitted over the channel. The accuracy of the distribution can be improved by using a SCL decoder for encoding.

6.1.4. Decoding

We can use standard SC or SCL decoders for shaped polar and BMERA codes with the unfrozen bits $\mathcal{U}$ and the frozen bits $\mathcal{F}$. Note that the input to the decoders must be the APPs $P_{X|Y}$ or corresponding LLR values, due to the non-uniform distribution of X. Outer-CRC codes can be added and used during decoding to reduce the number of codewords in the list.

6.1.5. Extension to BMERA Codes

The polarization in [29] is only proved for polar codes and the results in [30, 32, 33, 35, 36, 82, 127, 128] are based on polar codes, as well. The extension to BMERA codes is straightforward. The proof of (6.5) and (6.6) in [29] is based on (6.3)-(6.4) and no additional structure of the polar code is taken into account. In [22], (6.3)-(6.4) is proved for BMERA codes with a different factor β. All the remaining steps in [29] are not polar transform specific and therefore the same for BMERA codes.

6.2. Multilevel Honda-Yamamoto Scheme

The paper [82] extends the binary shaping scheme described above to a multilevel scheme that is called multilevel Honda-Yamamoto (MLHY) scheme. The MLHY scheme uses the same multilevel structure described in chapter 4 for PCM. At the decoder of the MLHY scheme, a multistage list decoder as described in 4.3 is used. Encoding for MLHY requires a SC or SCL decoder as described in Sec. 6.1.3. For the MLHY scheme, a multistage list decoder as described in 4.3 is used for encoding. The sets $\mathcal{F}$ and $\mathcal{U}$ are obtained as described for PCM in Sec. 4.2. The sets $\mathcal{S}$ and $\mathcal{U}_S$ are obtained with Monte Carlo simulations as described in Sec. 6.1.2 where the set indices are chosen jointly across all bit-levels as described in 6.1.2. It is proved in [82] that this scheme achieves the capacity of DMCs.

6.3. Numerical Results

We next present numerical results for MLHY polar and BMERA codes. The sets $\mathcal{U}$ and $\mathcal{F}$ for polar codes are designed with biAWGN surrogate channels, as described in chapter 4.2.1. For BMERA codes, we use BECs as surrogate channels. The sets $\mathcal{S}$ and $\mathcal{U}_s$ are obtained via Monte-Carlo approximation of the entropy terms as described above.

6.3.1. Rate Loss Comparison of Polar and BMERA based DMs

A polar or BMERA code can be used as a pure DM, i.e., we have $\mathcal{U} = \{1, \ldots, mn\}$ and $\mathcal{F} = \emptyset$. We compare polar and BMERA DMs with various encoding list sizes to CCDM and SMDM in terms of the rate loss [40]

$$\Delta_R = \mathbb{H}\left(\hat{P}_X\right) - R = \mathbb{H}\left(\hat{P}_X\right) - \frac{mn-s}{n}. \tag{6.16}$$

Table 6.1 lists the parameters that we use for rate loss comparisons. We use one-sided alphabets of the form $\mathcal{X} = \{0, 1, \ldots, M-1\}$ and choose the MB distribution that results in the desired rate R.

M	R [bpcu]
2	0.5
4	1.625
8	2.375
16	3.250

Table 6.1.: Parameters of DMs for rate loss comparisons.

Rate Loss with and without Min-Sum Approximation

Fig. 3.4 shows the FERs of the min-sum approximation for decoding of polar codes. Fig. 6.2 shows the effect of the min-sum approximation on the rate loss of polar and BMERA DMs using standard SC decoding at the encoder. Similar to the effect on the FER in 3.4, the rate loss due to the min-sum approximation grows as the block length n grows.

Fig. 6.3 shows the same rate loss curves but with a list encoder. For large n the degradation by the min-sum approximation seems to have even more effect than with $L_{\text{enc}} = 1$. The rate loss seems to saturate, especially for $M = 4, 8, 16$, i.e., for the MLHY scenarios. The influence of the min-sum approximation for BMERA codes is similar and not shown here. Even though it seems the min-sum approximation affects the rate loss more

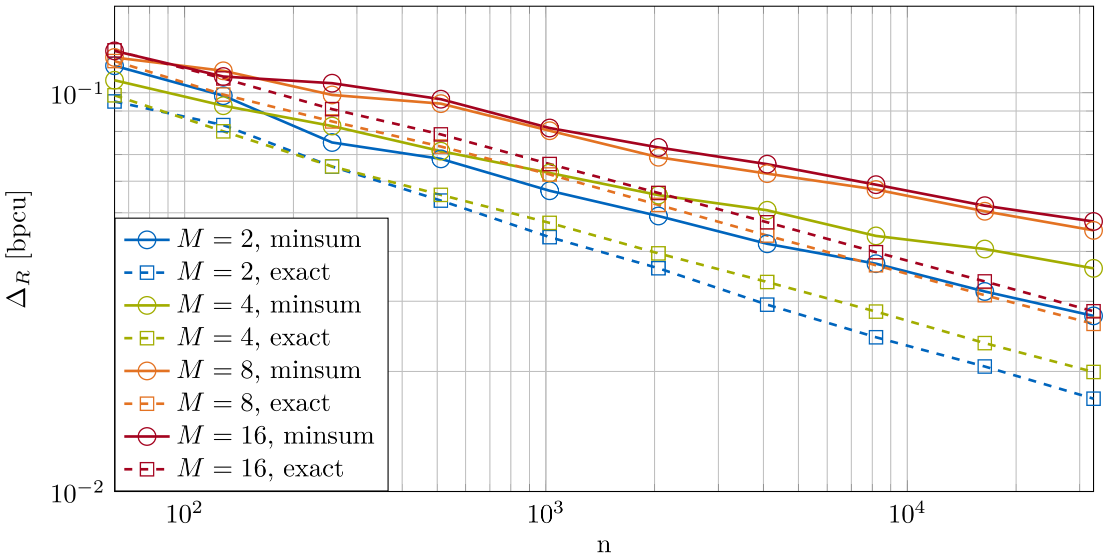


Figure 6.2.: Effect of the min-sum approximation on the rate loss of polar DMs with encoding list size $L_{\text{enc}} = 1$.

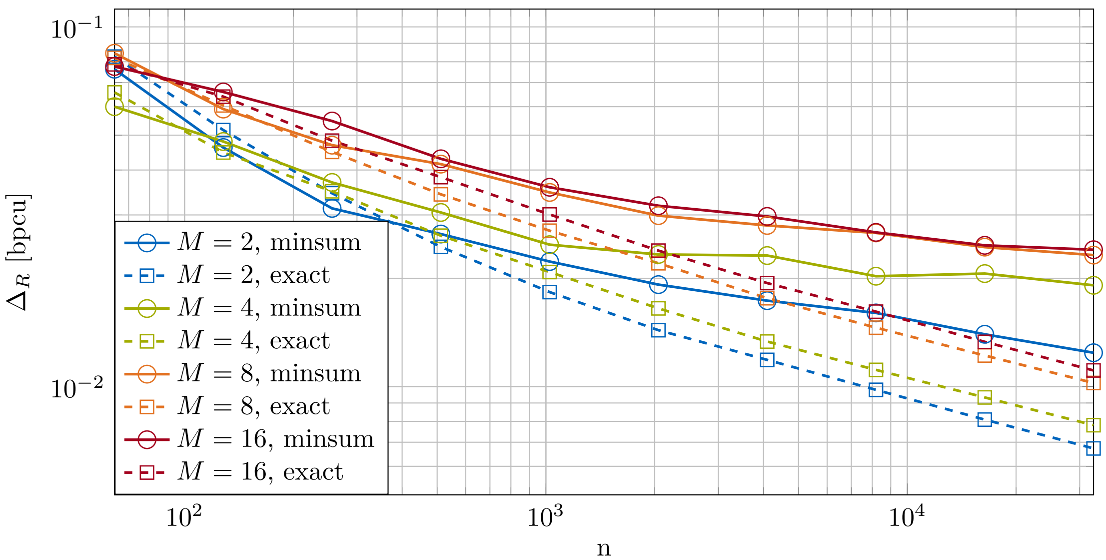


Figure 6.3.: Effect of the min-sum approximation on the rate loss of polar DMs with encoding list size $L_{\text{enc}} = 32$.

than the FEC during channel decoding, the loss in energy efficiency is comparable. We approximate the slope of the AWGN channel capacity with 0.15 bpcu/dB. Then a rate loss of 0.02 bpcu that we have for large n in Fig. 6.2-6.3 corresponds to an energy loss of approximately 0.13 dB that is similar to the losses in Fig. 3.4. Therefore, we do not use the min-sum approximation for the FER simulations in section 6.3.2. To compare DMs, we use the exact results (without using the min-sum approximation) to see the full potential of polar and BMERA DMs.

Rate Loss Comparison of Polar Matcher, BMERA Matcher, CCDM and SMDM

Fig. 6.4 compares the rate loss of binary polar and BMERA DMs as described in section 6.1 and [29, 30] with CCDM [95] and SMDM [96]. Observe the rate loss of BMERA codes is less than for polar codes, i.e., we save a factor of approximately 2 in block length when using BMERA codes instead of polar codes. This factor grows to 2.6 under list encoding with $L_{\mathrm{enc}} = 32$. The accuracy of polar and BMERA DMs can be improved by using list encoding. Under list encoding with L_{enc}, we save a factor of approximately 21.9 in block length compared to normal encoding with $L_{\mathrm{enc}} = 1$ for polar codes. The factor for BMERA codes is slightly higher.

The rate loss of polar and BMERA codes with list encoding is comparable to those of CCDM if the block length is approximately $n = 128$. As the block length grows, the rate loss of CCDM gets smaller than that of polar and BMERA DMs. In [96, Fig. 1], it is shown that a SMDM has good rate loss properties for small block length. Even with list encoding, polar and BMERA DMs are outperformed by the SMDM.

Fig. 6.5 compares the rate loss curves of CCDM, SMDM, polar and BMERA MLHY DMs for $M = 4, 8, 16$. The same plot without BMERA and SMDM is shown in [82, Fig. 2]. The best rate loss results can be obtained by the SMDM. Neither CCDM nor polar or BMERA DMs can compete with the SMDM. The comparison of binary DMs in Fig. 6.4 already showed that polar and BMERA DMs are comparable to a CCDM if n is small. For $M > 2$, the MLHY polar and BMERA DMs even outperform CCDM at short block lengths. Especially when M is large, the rate loss of polar and BMERA DMs is considerably less than that of a CCDM.

6.3.2. FER Simulations

Optimization

For the simulations of MLHY polar and BMERA codes, we heuristically optimize the following parameters:

- ▷ n_{CRC}: number of CRC-bits;
- ▷ $\mathrm{SNR}_{\mathrm{code}}$: the SNR used to construct the codes according to chapter 4.2.1;
- ▷ SNR_{P_X}: the SNR used to optimize the distribution;
- ▷ s: size of the shaping set $\mathcal{S}$.

We perform a grid search over these parameters as described in chapter 5.4.1 for PC-PAS. In addition, we optimize the size of the shaping set $s = |\mathcal{S}|$. We make a gird search where

$$(\log_2(M) - \mathbb{H}(X))\, n - 20 \leq s \leq (\log_2(M) - \mathbb{H}(X))\, n + 20. \tag{6.17}$$

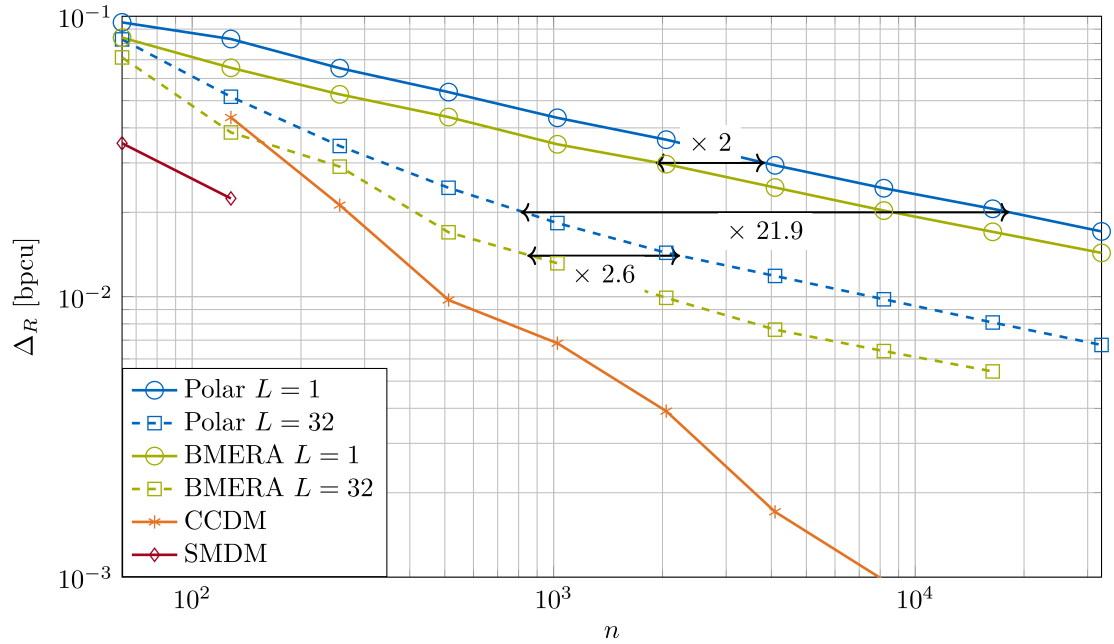


Figure 6.4.: Rate loss comparison of binary polar and BMERA DMs with CCDM and SMDM.

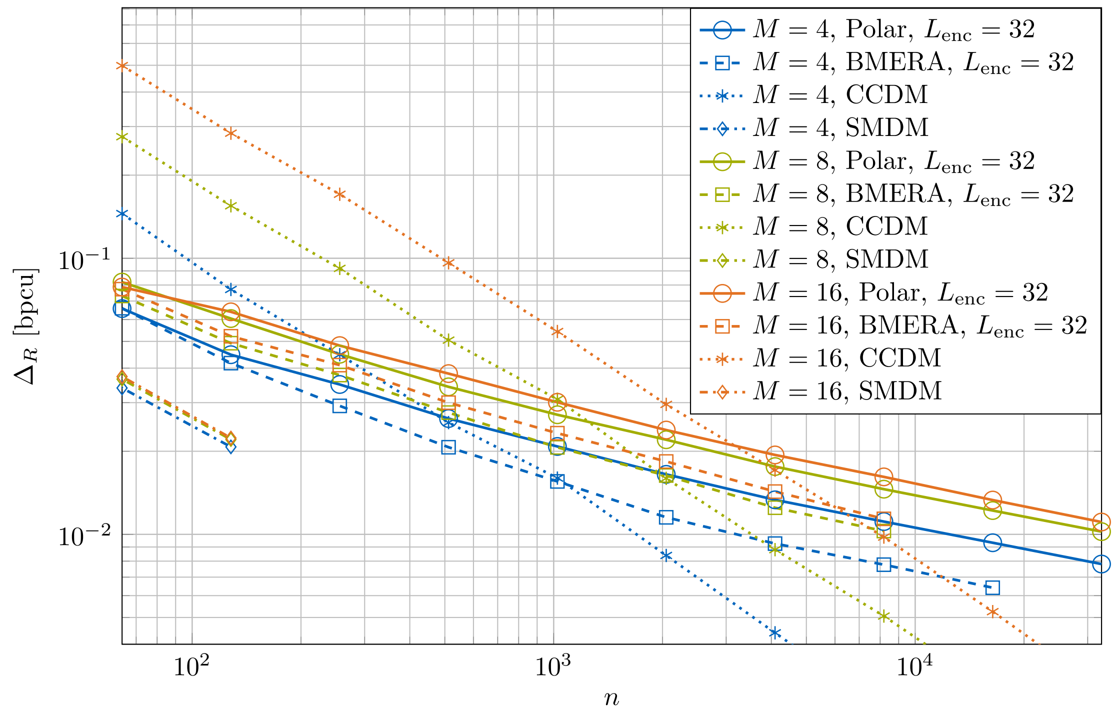


Figure 6.5.: Rate loss comparison for $M > 2$ of CCDM, SMDM, polar MLHY and BMERA MLHY.

Scenario	n	rate [bpcu]	$\mathrm{SNR}_{\mathrm{opt}}$ [dB]	n_{CRC} [bits]	$\mathrm{SNR}_{\mathrm{code}}$ [dB]	SNR_{P_X} [dB]	s [bits]
OOK, polar	1024	2/3	6.75	5	7.75		
OOK, BMERA	1024	2/3	6.75	10	7.25		
OOK, polar shaped	1024	2/3	6	11	6.5	4.5	116
OOK, BMERA shaped	1024	2/3	5.8	9	5.8	3.8	126
Polar MLHY 4-PAM	64	1.25	13.2	6	15.2	11.7	21
BMERA MLHY 4-PAM	64	1.25	13.2	3	15.2	10.2	24
Polar MLHY 8-ASK	64	1.75	12.8	5	14.8	12.3	23
BMERA MLHY 8-ASK	64	1.75	12.7	3	13.7	12.2	23

Table 6.2.: Code parameters resulting from the optimization.

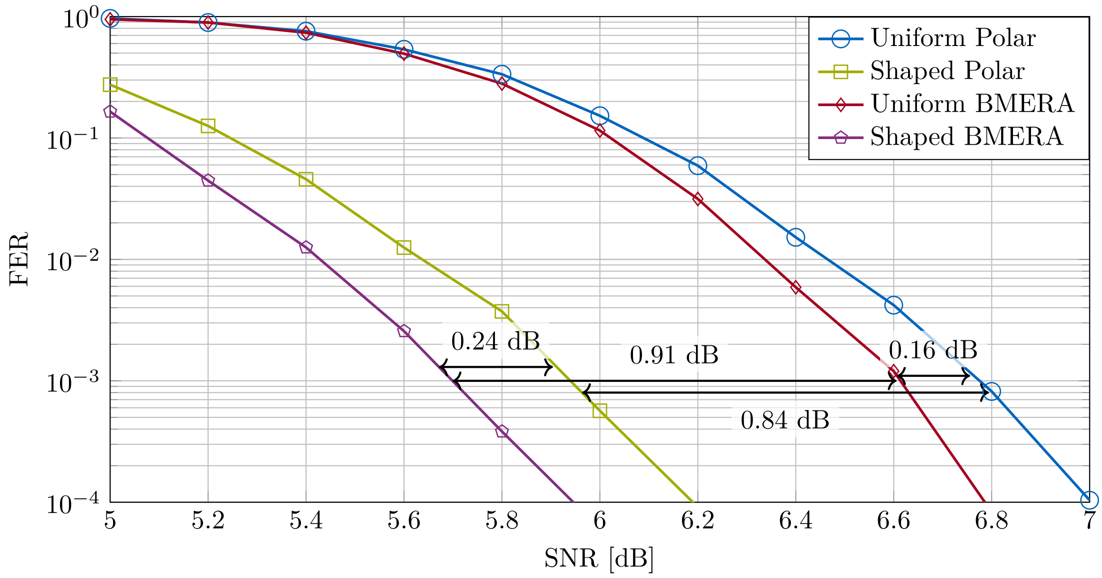


Figure 6.6.: FER of OOK with $n = 1024$ and rate 2/3 bpcu; see [30, Fig. 4]).

Table 6.2 summarizes the simulated scenarios and the resulting parameters from the grid-search optimization. The parameters of the uniform PCM and PC-PAS schemes are listed in Table 5.1. The PC-PAS reference curves are always with type check.

Binary OOK Transmission using Shaped BMERA and Polar Codes

Fig. 6.6 shows the results for OOK transmission with shaped and uniform polar and BMERA codes. This is an example for non-uniform input distributions that cannot be achieved by PAS. The same scenario is also simulated in [30] but without BMERA codes. Observe the large shaping gains of 0.91 and 0.84 for BMERA and polar codes, respectively. The results suggest that BMERA codes profit slightly more from shaping than polar codes

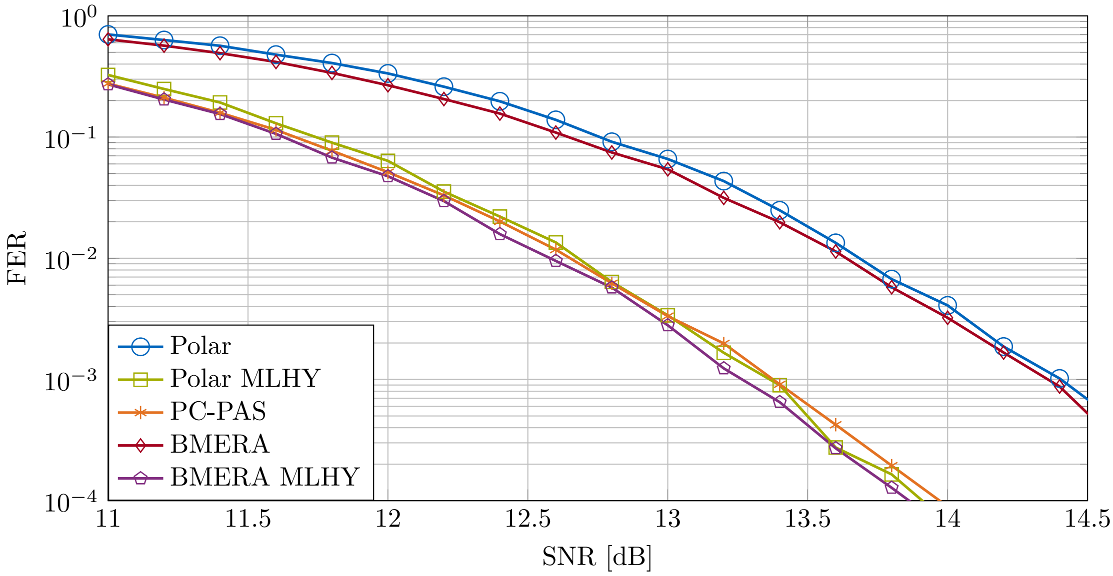


Figure 6.7.: FER of MLHY polar and BMERA codes with 4-PAM for $n = 64$ and rate 1.25 bpcu; see [82, Fig. 4].

due to the higher error exponent β of BMERA codes. For uniform polar and BMERA codes the difference of the higher error exponent is partially compensated by the outer CRC code. For list encoding of polar and BMERA codes, there is nothing corresponding to an outer CRC code and we can slightly see the effect of the higher error exponent. The improved performance of BMERA codes compared to polar codes (0.24 dB for the shaped case) comes at the cost of higher complexity. A complexity comparison of polar and BMERA codes is provided in [25].

6.3.3. MLHY Polar and BMERA Codes with ASK and PAM

Fig. 6.7 shows FER results for 4-PAM transmission with $n = 64$ and rate 1.25 bpcu. The results are similar to the results in Fig. 5.9. The FER of the MLHY polar and BMERA codes is similar to the FER of PC-PAS. BMERA codes seem to have a slightly better FER than polar codes but the difference is small and comes with more complexity.

Fig. 6.8 shows FER results for 8-ASK for $n = 64$ and rate 1.75 bpcu. Uniform polar and BMERA codes perform similarly here. PC-PAS cannot achieve the full shaping gain. One reason might be the higher rate loss of CCDM compared to polar and BMERA codes at short block lengths, as seen in Fig. 6.5. The shaping gain of BMERA MLHY codes seems to be slightly higher than that of polar MLHY codes. Again, the higher error exponent of BMERA codes might explain this behaviour.

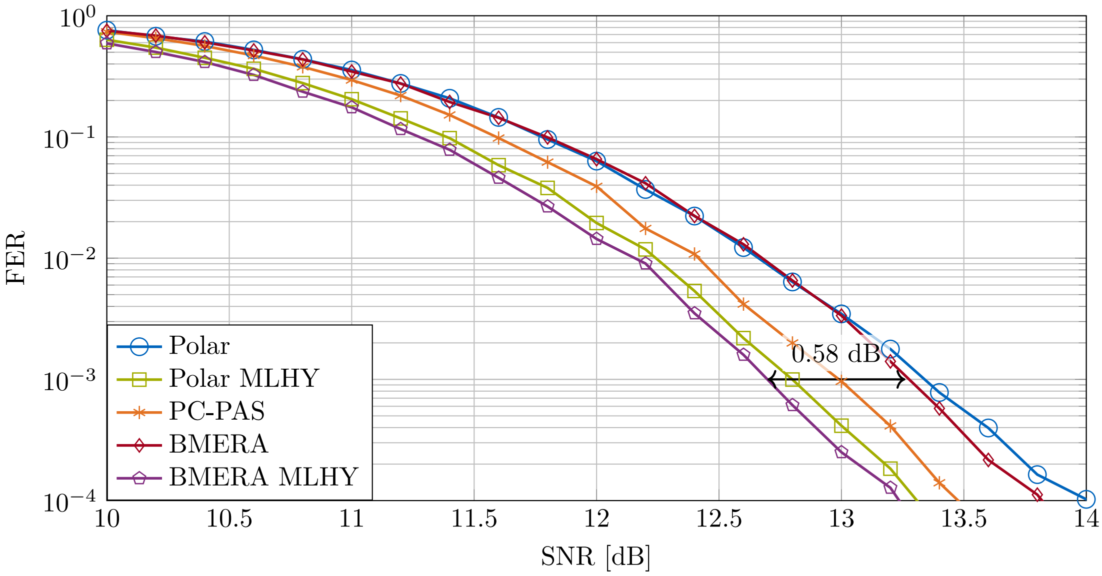


Figure 6.8.: FER of MLHY polar and BMERA codes with 8-ASK for $n = 64$ and rate 1.75 bpcu; see [82, Fig. 4].

7

6-PAM for Peak-power Constrained Channels

Many of today's short-reach optical communication systems operate without optical amplifiers and the input power to the fiber is limited by the maximum power of the laser. This dictates a peak power constraint [51]. Usually IM and DD are used with OOK or unipolar 4-ary and 8-ary PAM modulation. Increasing the modulation size results in higher complexity, especially at the receiver.

Usually, the number of symbols in a modulation alphabet is a power of two. This allows for an easy integration with binary data and binary FEC since the bits can be directly mapped to symbols. However, under a peak power constraint, the optimal input distribution is discrete [54] and there are SNR regions where a modulation format with six points outperforms modulation formats with eight or four points. Also, the complexity at the receiver can be reduced with six instead of eight points. Fig. 7.1 shows the rates of M-PAM with $M = 4, 6, 8$ under soft decoding (SD) with symbol-metric decoding (SMD) that can be achieved over a peak-power constraint AWGN channel. Observe that 6-PAM outperforms 4-PAM and 8-PAM at PSNR values between 20.5 dB and 24 dB. This result motivates using 6-ary modulations. The main challenge is the mapping of binary data to the signal points.

In [55, 56], the authors suggest to use 32-QAM, or more precisely a cross shaped QAM-32 (32-C-QAM) constellation as shown in Fig. 7.2 with the constellation shifted to the first quadrant of the complex plane. The idea is to map 5 bits to 32 two-dimensional signal points with 6 levels per dimension. The two dimensions of a signal point are then transmitted consecutively as two real signal points. We observe in Fig. 7.11 that this scheme does not outperform 6-PAM or 8-PAM.

In this chapter, we study three schemes to transmit 6-PAM. First, we review the benchmark 32-C-QAM. Second, we propose a modified 32-QAM constellation with a framed-cross shape that improves information rates. We call this constellation framed-cross shaped

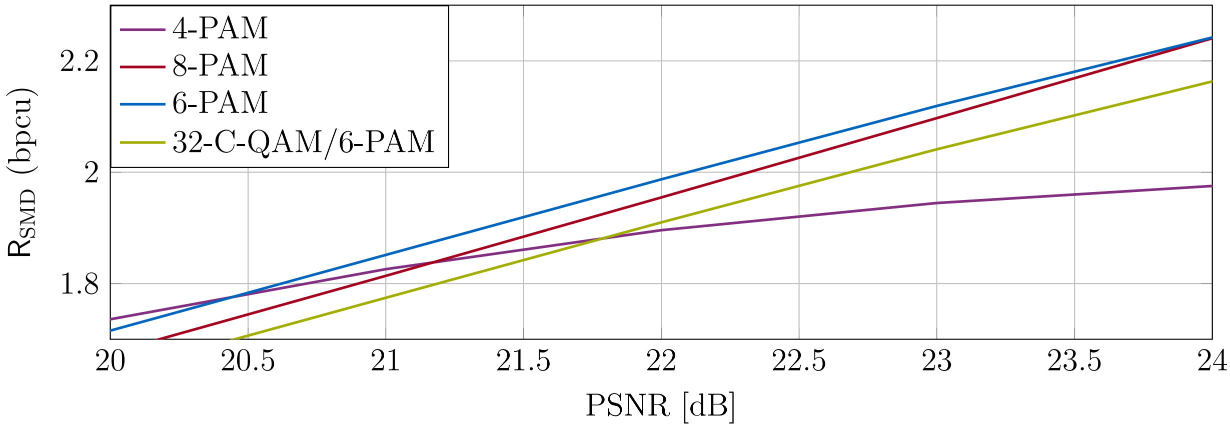


Figure 7.1.: Soft SMD rates of M-PAM with $M = 4, 6, 8$ and 32-C-QAM; see Fig. 7.2.

32-QAM (32-FC-QAM). Third, we use PAS [40] to map bits to a 6-ary constellation. The PAS scheme is not restricted to 6-ary constellations and can be used for any constellation whose number of points M is even but not a power of two. The idea is that a DM creates a block of $(M/2)$-PAM symbols that is combined with parity bits from a FEC code to obtain a block of M-PAM symbols. Of course, PAS can also generate non-uniform distributions. We compare the different 6-PAM strategies in terms of achievable rates and FERs over peak-power constrained AWGN channels. Finally, we conduct experiments to confirm the results for fiber-optic channels with DD.

This chapter is mainly based on the publications [78, 79].

7.1. Achievable Rates for 6-PAM Constellations

Consider a peak-power constrained AWGN channel as described in Sec. 2.3.2. An important metric to compare different constellations is achievable rate. We study the rates $\mathsf{R}_{\mathrm{SMD}} = I_{\mathrm{CM}}$ and $\mathsf{R}_{\mathrm{BMD}}$ defined in Sec. 2.4.2 under SMD and BMD. These rates can be achieved with SD, i.e., the detector outputs soft metrics like probabilities or LLRs that are used by a soft decoder. In addition, we also consider the rates $\mathsf{R}_{\mathrm{HD,SMD}}$ and $\mathsf{R}_{\mathrm{HD,BMD}}$ under hard-decision decoding (HD), i.e., the detector outputs hard decisions of the bits or symbols, respectively. We refer to [129, Eq. (1)], [130, Chap. 8] and [131, Sec. III-IV] for detailed derivations and descriptions of these rates. Unless stated otherwise, we use uniformly distributed input symbols with probability $P_X(x) = 1/M, \forall\, x \in \mathcal{X}$ in this chapter.

For symbol-wise HD [131, Sec. III-C], [130, Eq. (8.53)], one can achieve the rate

$$\mathsf{R}_{\mathrm{HD,SMD}} = \max\left(0,\ \mathbb{H}\left(X\right) - \left(\mathbb{H}_2\left(\delta\right) + \delta \log_2(M-1)\right)\right) \tag{7.1}$$

where $\delta = \Pr[\hat{X} \neq X]$ is the symbol error probability with $\hat{X}$ being the symbol-wise hard

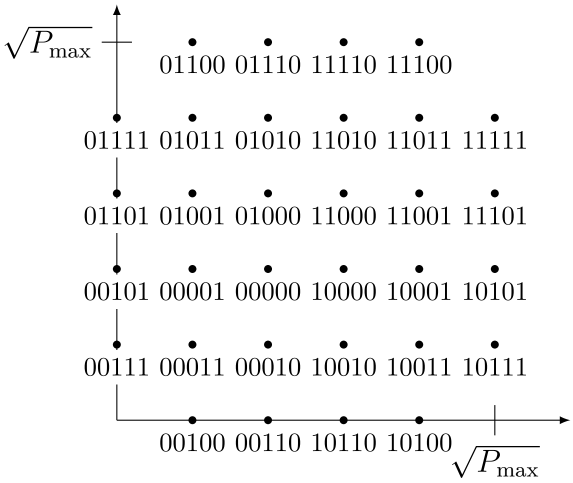


Figure 7.2.: 32-C-QAM with quasi-SU labeling [132].

decisions fed to the decoder. The BMD rate under HD [131, Sec. IV], [130, Eq. (8.66)] is

$$\mathsf{R}_{\mathrm{HD,BMD}} = \max\left(0,\ \mathbb{H}\left(X\right) - m\mathbb{H}_2\left(\varepsilon\right)\right) \tag{7.2}$$

where $\varepsilon = 1/m \cdot \sum_{k=1}^{m} \Pr[\hat{B}_k \neq B_k]$ is the average bit error probability where $\hat{B}_k$ is the bit-wise hard decision of the k^{th} bit.

7.2. Strategies to Transmit 6-PAM

7.2.1. 6-PAM using 32-QAM Constellations

Cross 32-QAM Constellation

One-dimensional 6-PAM symbols can be created from a two-dimensional 32-C-QAM constellation [55, 56], as depicted in Fig. 7.2. 32-QAM is a simple choice to transmit 6-ary alphabets since 5 bits are directly mapped to the 32-QAM symbols. We use a quasi symmetric-ultracomposite (SU) labeling structure introduced in [132]. Fig. 7.1 shows that 32-C-QAM is not a good choice to transmit 6-PAM under a peak power constraint. Fig. 7.3 shows the optimal distribution of 6-PAM under a peak power constraint at a rate of 1.85 bpcu for SD-SMD. Observe the outer constellation points have a larger probability than the inner points. This is the opposite of the distribution induced by the 32-C-QAM constellation. Fig. 7.4 shows the one-dimensional distributions of the real and imaginary parts of the 32-C-QAM constellation. The outer points have the smallest probabilities, which explains the poor performance of 32-C-QAM in Fig. 7.1. We formulate an optimization problem in the next section to obtain 32-QAM constellations that are better suited to a peak power constraint.

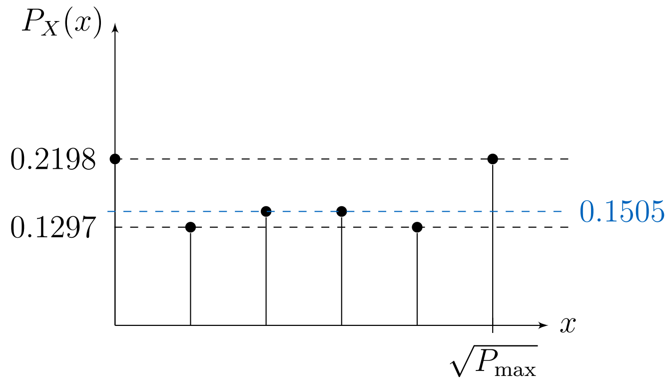


Figure 7.3.: Optimal distribution $P_X(x)$ of 6-PAM for SD-SMD at a rate of 1.85 bpcu.

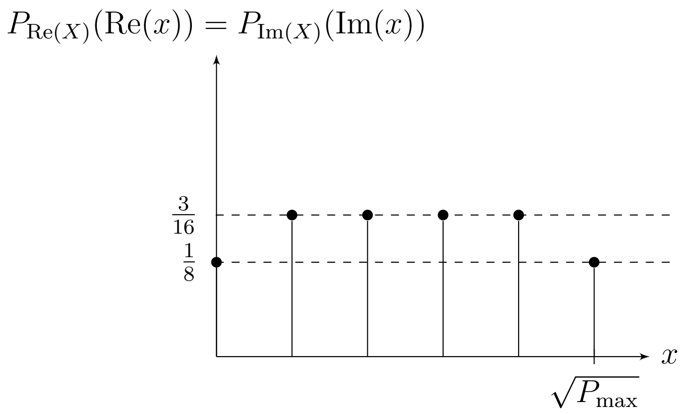


Figure 7.4.: One-dimensional distribution of 32-C-QAM.

Framed-cross 32-QAM Constellation

We design 32-QAM constellations by removing 4 constellation points from the full 6×6 36-QAM constellation

$$\mathcal{X}_{36} = \{a + \mathrm{j}b \mid a, b \in \mathcal{X}\} \tag{7.3}$$

where we use $\mathcal{X}$ from (2.5) with $M = 6$. Now maximize the SMD-SD rate over all 32-symbol subsets $\tilde{\mathcal{X}}$ of $\mathcal{X}_{36}$:

$$\mathcal{X}^{\star}_{32,\mathrm{SMD}} = \underset{\substack{\tilde{\mathcal{X}} \subset \mathcal{X}_{36} \\ |\tilde{\mathcal{X}}|=32}}{\operatorname{argmax}} \mathsf{R}_{\mathrm{SMD}}. \tag{7.4}$$

The optimal constellation $\mathcal{X}^{\star}_{32,\mathrm{SMD}}$ according to (7.4) at PSNR values around 22 dB is depicted in Fig. 7.5. This constellation does not allow Gray labeling and finding the best bit labeling under BMD seems difficult. Instead, we modify (7.4) by taking the BMD rate from (2.33) as the new objective function. Furthermore, we impose a Gray labeling con-

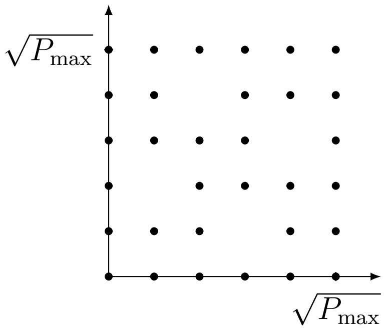


Figure 7.5.: Optimal 32-QAM constellation for SD-SMD under a peak power constraint.

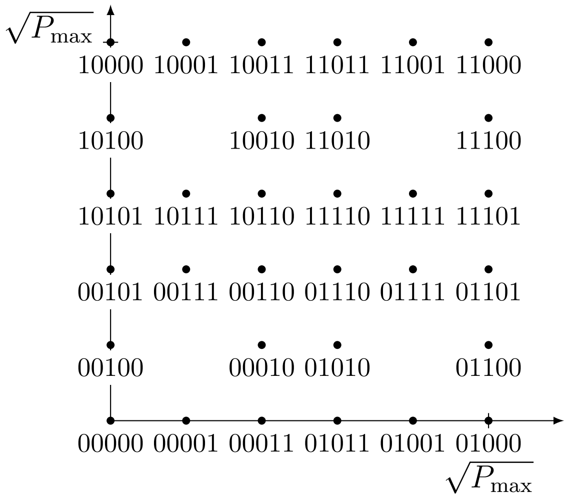


Figure 7.6.: 32-FC-QAM constellation with Gray labeling.

straint were the first two bits of the label correspond to the quadrant of the constellation. The result is shown in Fig. 7.6 and we denote this constellation as 32-FC-QAM.

7.2.2. M-PAM using PAS

PAS was designed for distributions P_X that are symmetric around zero [40]. However, PAS can be extended to distributions with a symmetry as described in Sec. 2.6. We propose a PAS scheme for M-PAM distributions for even integers $M > 2$. The number of bits needed to label the constellation points is $m = \lceil \log_2 M \rceil$. The same idea is used in [133] to construct flexible QAM constellations.

In Sec. 2.6, we introduced the two PAS sets $\mathcal{X}_1$ and $\mathcal{X}_2$. We define the set $\mathcal{X}_1$ as the set of the first $M/2$ symbols of the M-PAM constellation $\mathcal{X}$:

$$\mathcal{X}_1 = \left\{ i \cdot \frac{\sqrt{P_{\max}}}{M-1} \right\}_{i=0}^{M/2-1}. \tag{7.5}$$

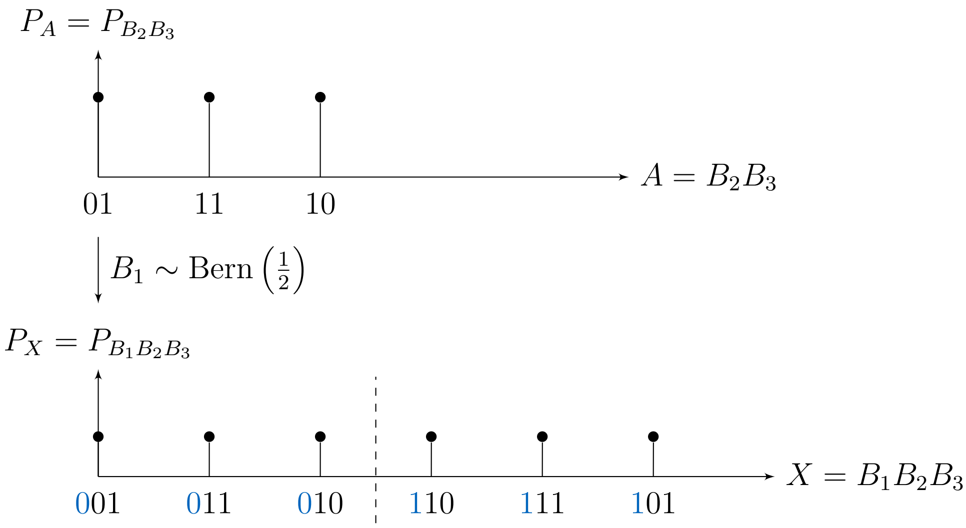


Figure 7.7.: Generating 6-PAM symbols using PAS: 3-PAM symbols are generated using a DM and labelled with two bits. Uniform parity bits are used to extend the 3-PAM symbols to 6-PAM symbols with a three bit label.

Every point $a \in \mathcal{X}_1$ has a binary label $b_2^m(x)$ of length $m-1$. We use subsets of BRGC [90] of length $m-1$, where we discard the first $2^{m-1} - M/2$ codewords from the usual Gray code (compare the labeling of $A = B_2B_3$ in Fig. 7.7). The mapping $T : \mathcal{X}_1 \mapsto \mathcal{X}_2$ as introduced in Sec. 2.6, is

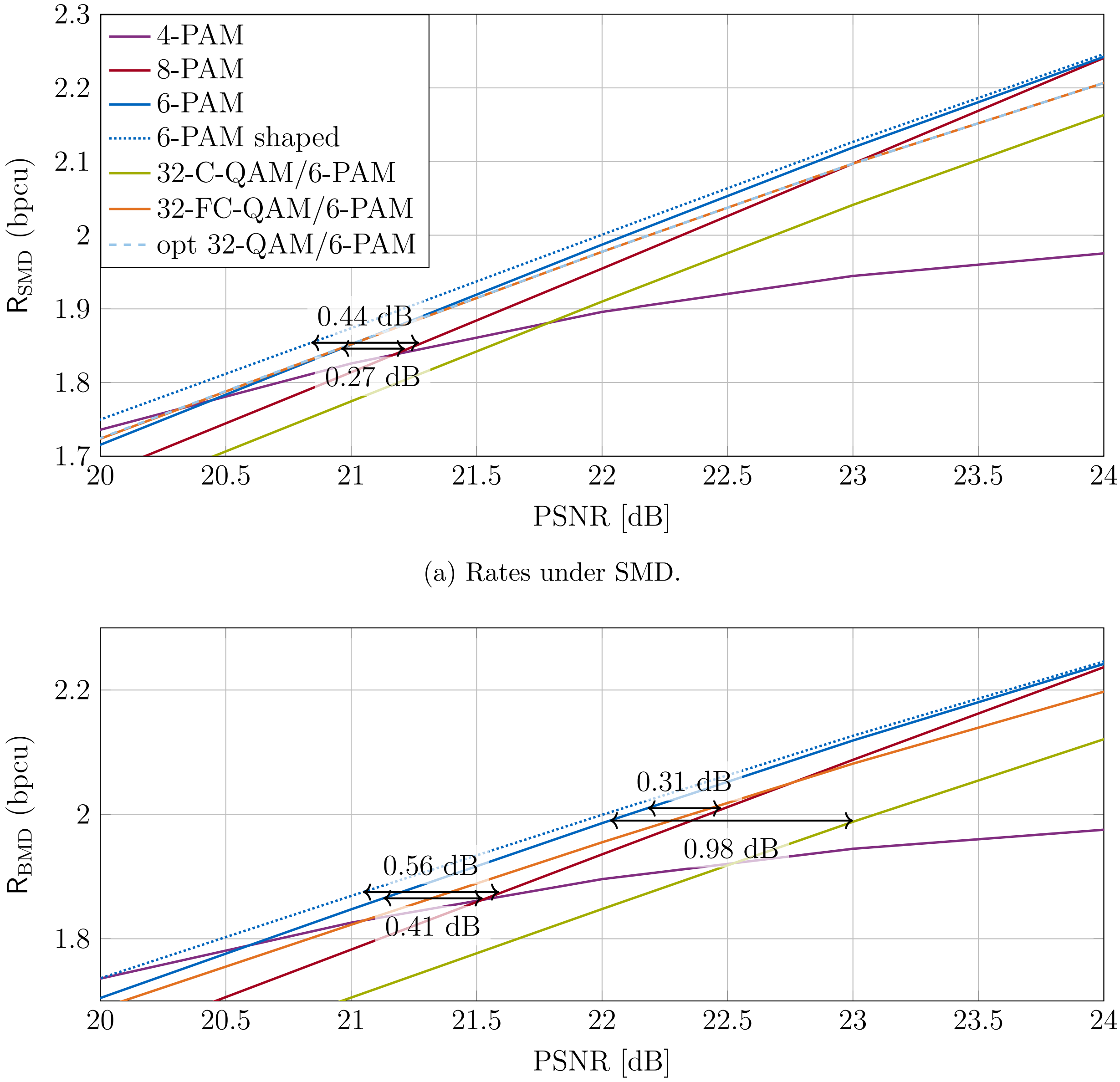
$$T(x) = \sqrt{P_{\max}} - x, \qquad \forall x \in \mathcal{X}_1. \tag{7.6}$$

Generation of 6-PAM via PAS is denoted as PAS-6-PAM.

7.3. 6-PAM Rates for AWGN Channels

We compare the 6-PAM strategies in terms of achievable rates for bit- and symbol-wise HD and SD. Fig. 7.8 shows the SD rates for transmission over a peak-power constrained AWGN channel. We assume the rate loss of the DM is zero, which is true for most DMs as the block length grows to infinity. The curve labeled “6-PAM shaped” shows that we can improve the rates by using a non-uniform distribution.

Fig. 7.9 shows the rates under hard decision decoding. The curves in Fig. 7.8 and Fig. 7.9 show that each decoding method has a PSNR region where 6-PAM outperforms 4-PAM and 8-PAM. The gains of 6-PAM over 8-PAM are significant for hard decision decoding: 1.32 dB for HD-SMD and 0.75 dB for HD-BMD at a rate of 2 bpcu. The SD gains are smaller: 6-PAM gains 0.27 dB over 8-PAM for soft SMD at a rate of 1.85 bpcu. The gain for soft BMD is 0.41 dB at 1.875 bpcu. The gains under SD can be increased to



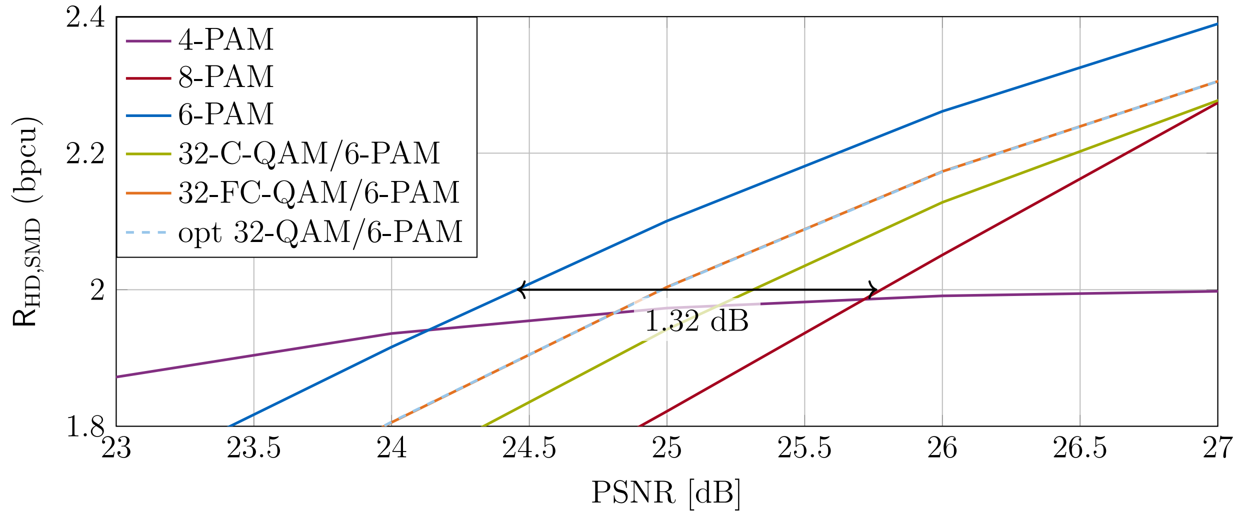
(a) Rates under SMD.

(b) Rates under BMD.

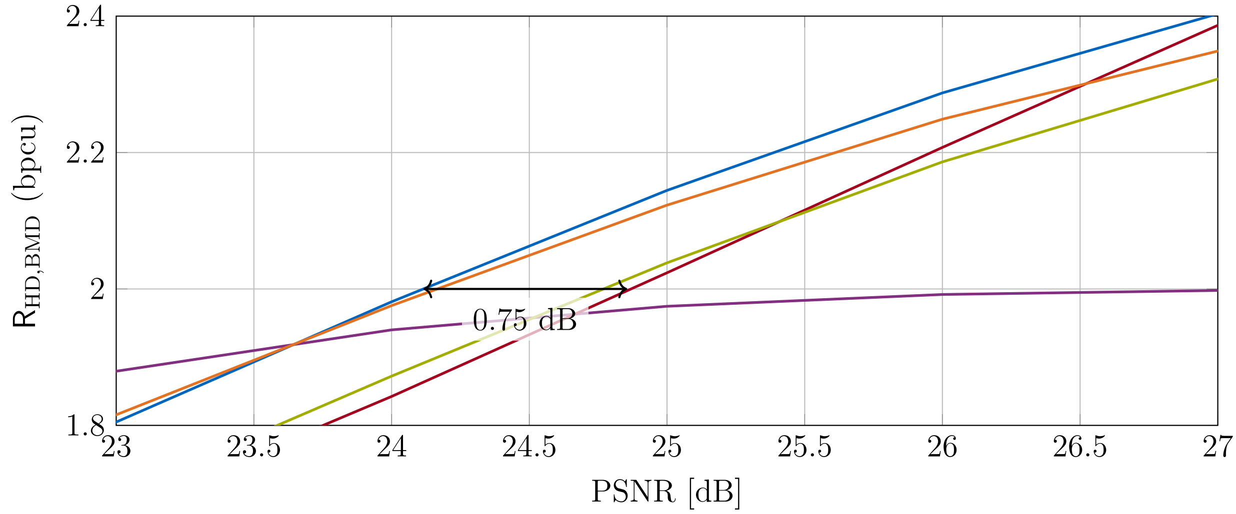
Figure 7.8.: Rates of M-PAM with $M = 4, 6, 8$ and 32-QAM for the peak-power constrained AWGN channel under SD.

0.44 dB and 0.56 dB when using non-uniform distributions. Fig. 7.8 and Fig. 7.9 show that 32-C-QAM loses significant PSNR compared to 6-PAM under a peak power constraint for all scenarios. This shows that 32-C-QAM as proposed in [55,56] is not a good choice under a peak power constraint.

Our proposed 32-FC-QAM constellation outperforms 32-C-QAM in every scenario. For



(a) Rates under SMD.



(b) Rates under BMD.

Figure 7.9.: Rates of M-PAM with $M = 4, 6, 8$ and 32-QAM for the peak-power constrained AWGN channel under HD.

SD-SMD and HD-BMD, 32-FC-QAM achieves similar rates as 6-PAM if the rates are low. At higher PSNR values, the performance of 6-PAM is better than 32-FC-QAM because the optimal distribution approaches a uniform distribution. For HD-SMD and SD-BMD, the rate of 32-FC-QAM lies between that of 6-PAM and 32-C-QAM.

Fig. 7.8a and 7.9a show the performance of 32-FC-QAM is almost optimal for 32-QAM constellations. The PSNR loss compared to the SMD optimized 32-QAM constellation from Fig. 7.5 is not visible.

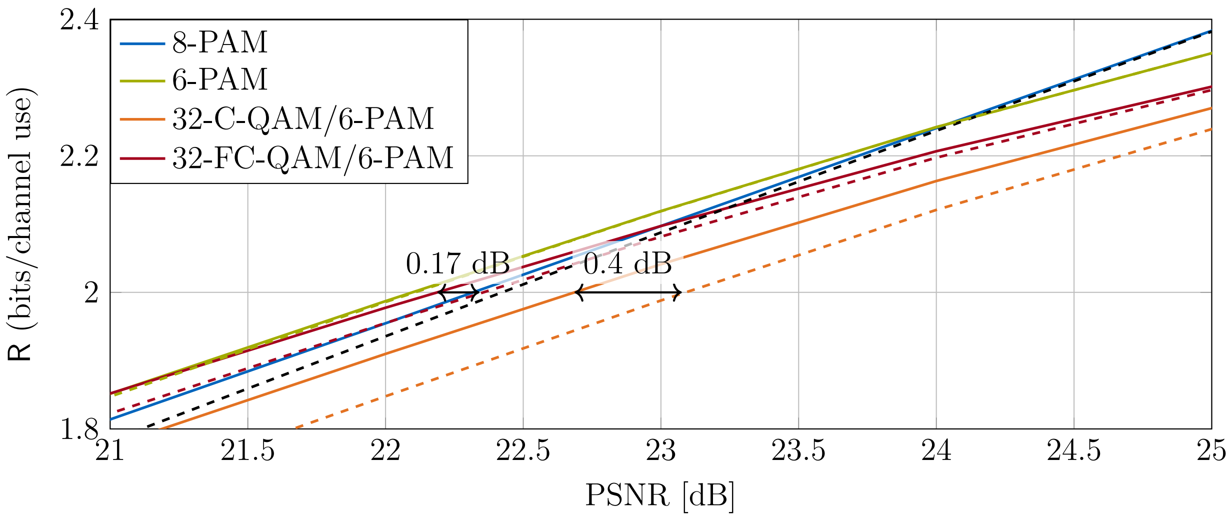


Figure 7.10.: SEs under BMD (dashed) and SMD (solid) soft-decoding.

Fig. 7.10 shows BMD and SMD rates for soft-decoding. BMD exhibits a loss because it uses a mismatched decoding metric instead of $P_{X|Y}$. The BMD loss of 6-PAM is negligible in the PSNR region of interest and the rate loss of 32-FC-QAM is only 0.17 dB, whereas there is a 0.4 dB loss between SMD and BMD for 32-C-QAM.

We conclude that the benefits of PAS-6-PAM and 32-FC-QAM over 32-C-QAM are:

1. The uniform (or optimized) distribution of PAS-6-PAM and the non-uniform distribution implied by 32-FC-QAM are closer to the optimal distribution (cf. Fig. 7.3) under a peak-power constraint than the distribution implied by 32-C-QAM (cf. Fig. 7.4).

2. The BMD loss is minimized due to the Gray labeling of PAS-6-PAM and 32-FC-QAM. 32-C-QAM does not allow a Gray labeling.

PAS-6-PAM has an additional gain: the entropy of 6-PAM is $\log_2(6) \approx 2.58$ bpcu compared to 2.5 bpcu for the 32-QAM constellations, i.e. the rate saturation occurs at a higher value.

7.4. FER Comparisons of 6-PAM Constellations

To confirm the results, we perform simulations over an AWGN channel including FEC. We use 5G new radio LDPC codes; see [2]. We evaluate 6-PAM with PAS and compare it to 6-PAM generated by 32-QAM, and to 8-PAM. We simulate at a SE of 2 bpcu. We use a block length of 3000 channel uses per code frame. The code parameters are summarized in Table 7.1. For 6-PAM with PAS we use a CCDM [95]. The rate loss of CCDM is here

$$\Delta_R = \log_2(3) - \frac{4743}{3000} \approx 0.004\,\text{bpcu}. \tag{7.7}$$

Scenario	code length	coderate R_{FEC}	γn
PAS 6-PAM	9000 bits	$7257/9000 \approx 0.8$	1257 bits
8-PAM	9000 bits	2/3	
32-QAM/6-PAM	7500 bits	0.8	

Table 7.1.: Code parameters.

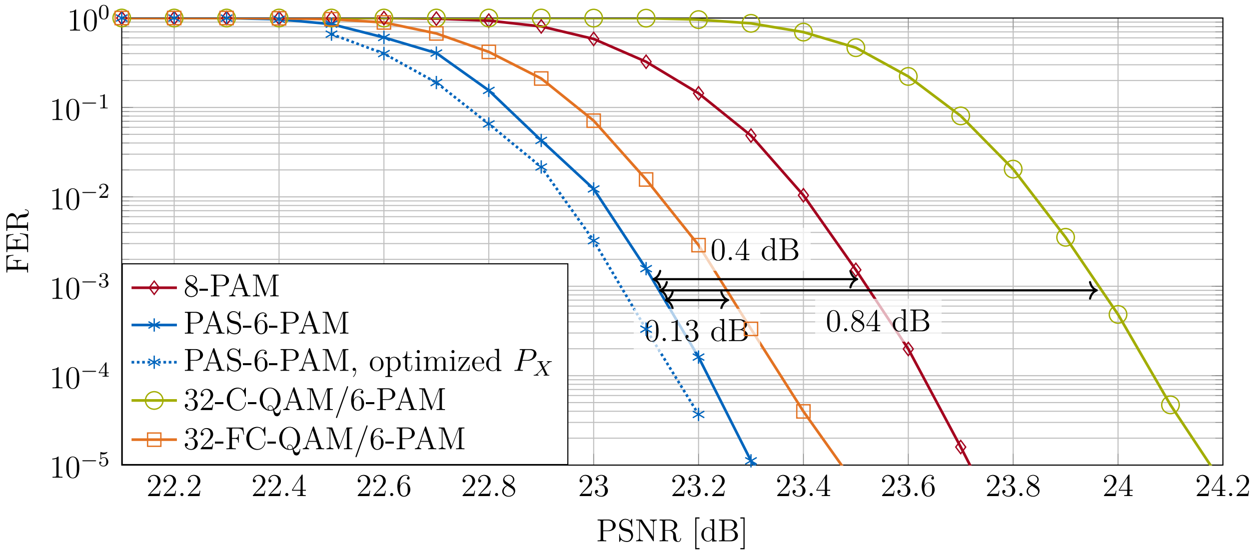


Figure 7.11.: Performance comparison of 6-PAM schemes at a SE of 2 bpcu under SD-BMD with $n = 3000$ channel uses per frame.

At the decoder, we perform 200 and 20 belief propagation iterations with a full sum-product update rule for SD and HD, respectively.

7.4.1. SD-BMD using 5G LDPC Codes

Fig. 7.11 shows the FER curves for soft BMD. At a FER of 10^{-3}, 6-PAM using PAS gains 0.4 dB and 0.84 dB compared to 8-PAM and 32-C-QAM, respectively. The achievable rate curves in Fig. 7.8b showed similar gains (0.31 dB and 0.98 dB, respectively). The FER of 32-FC-QAM is within 0.13 dB of 6-PAM. We gain another 0.13 dB by using an optimized P_X. Using a non-uniform distribution comes with no extra costs compared to 6-PAM with PAS with a uniform distribution. The results are in line with the rate results from Fig. 7.8b.

7.4.2. HD using 5G LDPC Codes

SD FEC codes may be too complex for high-throughput applications such as IM/DD for short-reach fiber-optic links. Instead, one often uses HD. Achievable rates for HD are

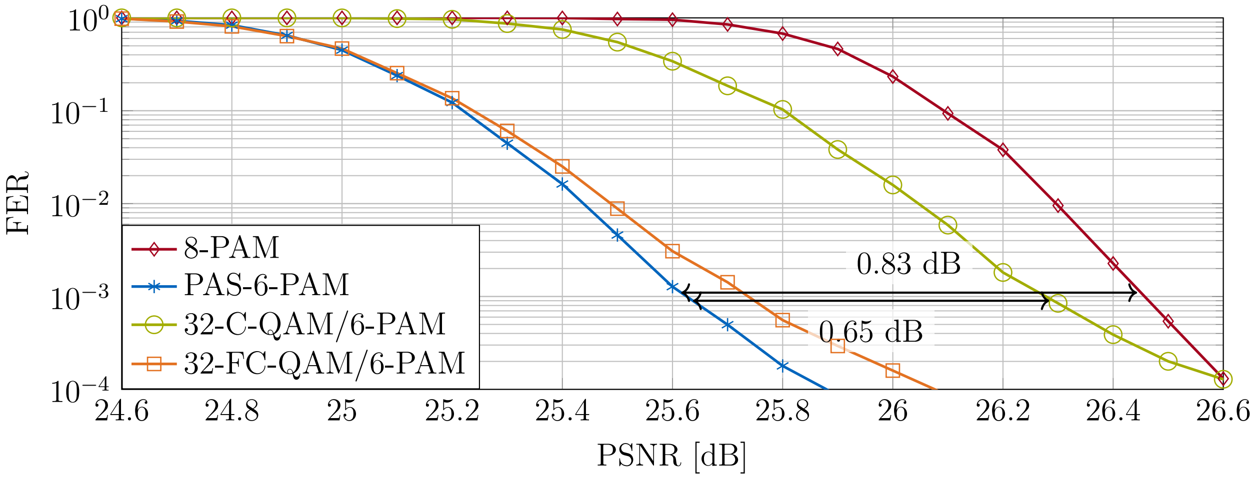


Figure 7.12.: Performance comparison of 6-PAM schemes at a SE of 2 bpcu under HD-BMD using 5G LDPC codes with $n = 3000$ channel uses per frame.

depicted in Fig. 7.9. Observe that BMD can improve on SMD under HD, see Figs. 7.9a and 7.9b, see [131]. The reason is that BMD works with the binary Hamming distance that preserves part of the Euclidean distance, while the M-ary Hamming metric for SMD does not.

To validate the results, Fig. 7.12 shows the performance of 5G LDPC codes with HD. We make bit-wise hard decisions on the channel outputs and assign the values $+a$ and $-a$ to the LLRs if the decision is 0 or 1, respectively. The value a is the same for all bit-levels and symbols and the optimal value of a is found by line search. We use the same simulation parameters as for SD, i.e., $n = 3000$ channel uses and a target SE of 2 bpcu. PAS-6-PAM gains 0.83 dB and 0.65 dB over 8-PAM and 32-C-QAM based 6-PAM, respectively. The FER performance of 32-FC-QAM is close to the performance of PAS-6-PAM. These results are very similar to the achievable rate results from Fig. 7.9b.

We remark there are codes that perform better under HD-BMD than 5G LDPC codes. The aim here is to show that information rate gains predict the real coding gains, and 5G LDPC codes allow flexible rate adaptation to approach the rates in Table 7.1. The choice of these sub-optimal codes for HD decoding might also explain the onset of error "floors" for all constellations except 8-PAM in Fig. 7.12.

7.5. Experiments

Sec. 7.3 and Sec. 7.4 provide achievable rate and FER results for peak power constrained AWGN channels. To further verify the results, we conduct optical experiments in the C-band. Similar experiments are conducted in [52] and [53] for 8-PAM in the O-band.

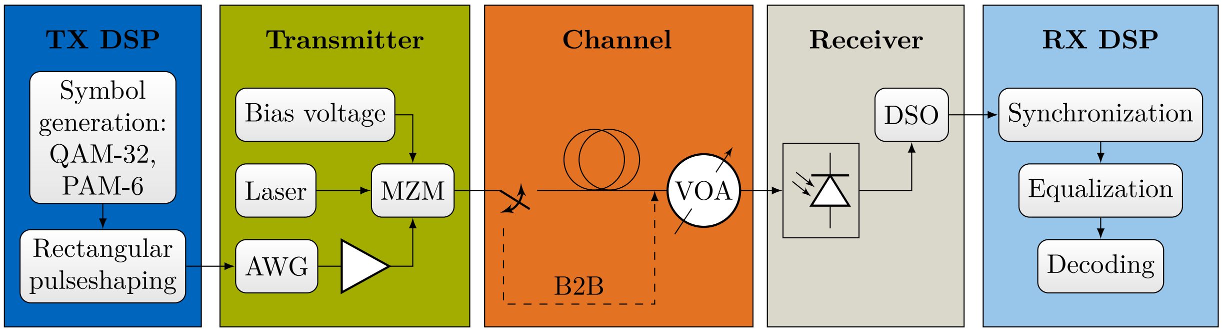


Figure 7.13.: Experimental setup.

7.5.1. Experimental Setup

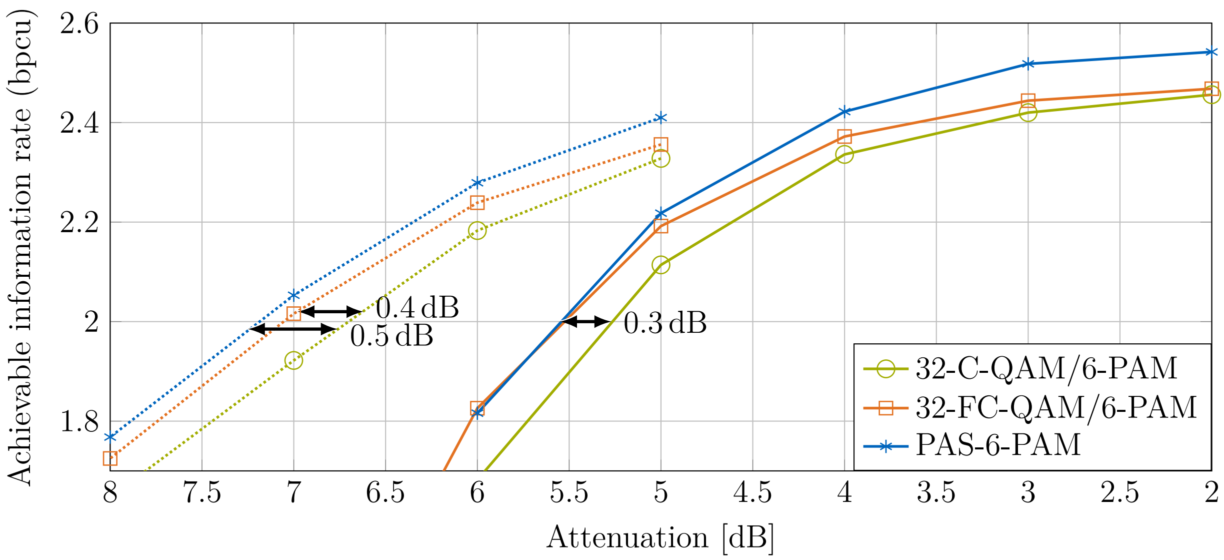
The experimental setup is depicted in Fig. 7.13. The transmitter generates constellation symbols according to one of the strategies described in Sec. 7.2 and performs rectangular pulse shaping. The arbitrary waveform generator (AWG) operates at sampling rates 95.6 GSamples/s and 120 GSamples/s. The first sampling rate is used for 95.6 GBaud transmission with 1 SPS and the latter for 60 GBaud transmission with 2 SPS. The AWG signal is electrically amplified and drives a Mach-Zehnder modulator (MZM) that modulates a 1550 nm external cavity laser. The MZM is operated at the quadrature point. We consider either transmission over 1 km of standard single mode fiber (SSMF) or an optical back-to-back (B2B) configuration. Both setups are connected to a variable optical attenuator (VOA).

After the VOA, a PD converts the optical intensity to an electrical current that is sampled by a digital storage oscilloscope (DSO) operating at 256 GSamples/s and the samples are stored for offline processing. We synchronize and re-sample to 2 SPS before equalization with a Volterra equalizer, using 41 first-order, 7 second-order and 5 third-order taps. The equalizer is trained based on a mean square error objective. The equalized sequence is down-sampled to the symbol rate to retrieve the symbols. For the setup with and without fiber, measurements are taken for different attenuation factors of the VOA. For 95.6 GBaud symbol rate transmission, we use an additional 3-tap noise whitening filter [48] and compute the symbol-wise APPs through the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm [65].

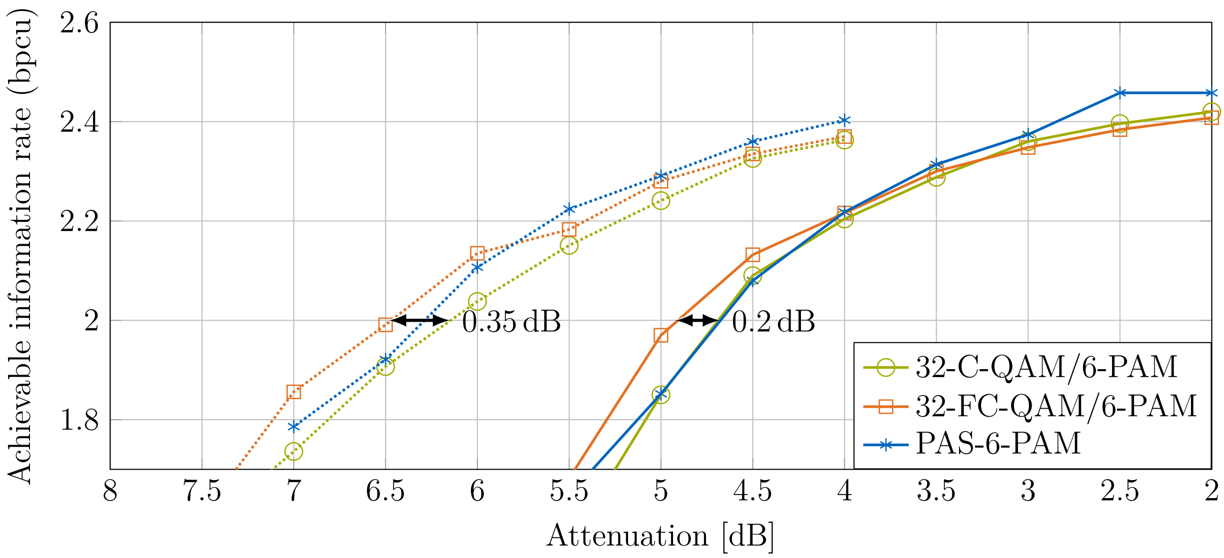
7.5.2. Experimental Results

We consider blocks of $n = 1000$ symbols. This results in a binary code length of 2500 bits and 3000 bits for 32-QAM and PAS-6-PAM, respectively. We use LDPC codes from the 5G standard for soft-decoding [2] and shortened Bose–Chaudhuri–Hocquenghem (BCH) codes [134, 135] with a mother code length of 4095 bits for HD. The transmission rate is adjusted by changing the code dimension. For PAS-6-PAM, we use a CCDM. The rate loss of the CCDM that outputs uniformly distributed symbols from the 3-ary alphabet is approximately 0.011 bpcu for $n = 1000$ symbols. Since the transmitted sequences from the experiments are usually not codewords of a FEC code, we apply the method of channel adapters and scrambling sequences [129].

Fig. 7.14 shows achievable information rates for transmissions at 60 GBaud symbol rate over 1 km SSMF, and at 95.6 GBaud symbol rate in an optical B2B setup. For each attenuation factor, we adjust the transmission rate so the FER is 10^{-3}. The results show that 32-FC-QAM gains between 0.2 dB and 0.4 dB over conventional 32-C-QAM for each scenario when operating at a rate of 2 bpcu. The largest gains are for soft BMD with LDPC codes. The 32-FC-QAM gains are especially noteworthy since the complexity is the same as for 32-C-QAM. PAS-6-PAM gains up to 0.5 dB over conventional 32-C-QAM. For 95.6 GBaud, PAS-6-PAM shows smaller gains, especially for high attenuation.



(a) Symbol rate: 60 GBaud, 1 km.



(b) Symbol rate: 95.6 GBaud, optical B2B.

Figure 7.14.: Achievable rates using BCH (solid lines) with HD and LDPC codes (dotted lines) with SD with block length $n = 1000$ real-valued channel uses evaluated at an FER of 10^{-3}.

8 Successive Interference Cancellation for Direct Detection

Receivers with DD measure the amplitude or intensity of signals by a PD. Usually such receivers are combined with IM at the transmitter, i.e., one modulates the intensity. DD allows easy reconstruction of the transmitted data under symbol-rate sampling [43–45]. Such a system is described in Chapter 7 where 6-PAM is used in a system with DD.

The paper [57] showed that the DD capacity increases by oversampling at the receiver, i.e., sampling faster than the symbol rate. In fact, if the dominant noise is before DD, then the capacity is within 1 bit/s/Hz of the capacity with coherent detection. This means one can retrieve at least parts of the phase of the transmitted signal after DD, which motivates using bipolar modulation or even complex-valued modulation formats [57–61, 80].

Fig. 8.1 shows a motivating toy example with BPSK and DD. We use triangular pulses as outlined by the grey lines. The green line shows the transmitted signal $X(t)$. DD is modeled by a square-law device (SLD) device that outputs the signal $|X(t)|^2$. Observe one cannot reconstruct the transmitted data with symbol-rate sampling outlined by the red dots in the plot. However, one can reconstruct the transmitted data by the blue samples in between that are obtained by two-fold oversampling.

As in [80], we study bandlimited systems with DD and compute achievable information rates using frequency-domain raised-cosine (FD-RC) pulses. We propose SIC with the FBA or Gibbs sampling for detection to approach the achievable information rates in practical systems. [136] extends the scheme where a neural network serves as detector. This chapter is mainly based on the papers [80, 81].

8.1. System Model

Propagation in fiber is described by the Nonlinear Schrödinger Equation [137, p. 65], [138, Part 2] that models attenuation and chromatic dispersion (CD), both linear effects, and a

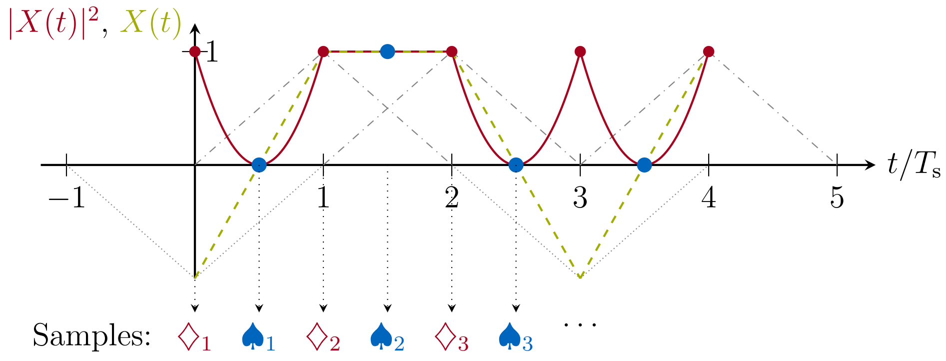


Figure 8.1.: Principle of phase retrieval in DD systems using a triangular pulse shape.

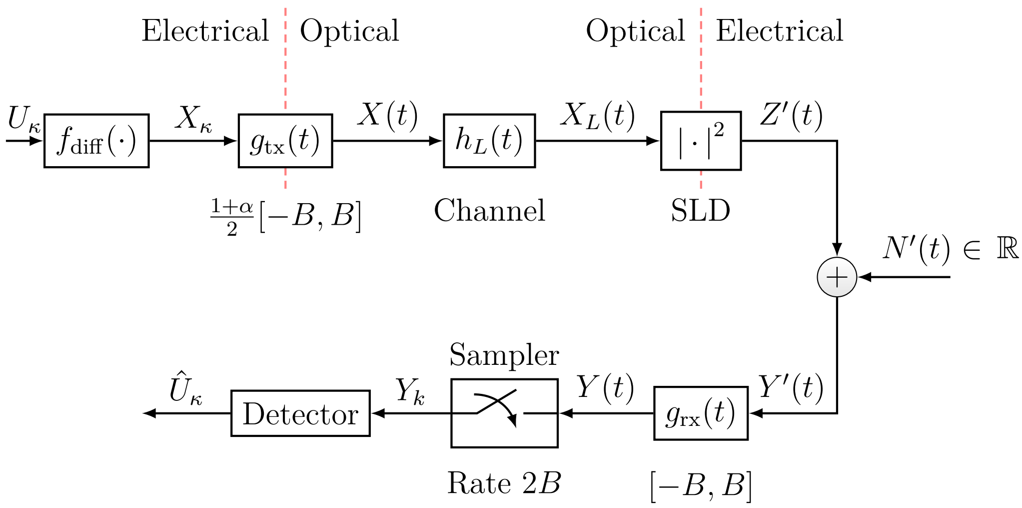


Figure 8.2.: System model with DD and two-fold oversampling [80].

Kerr non-linearity. As we consider short-reach links without optical amplifiers, the noise from the SLD is dominant and we neglect optical noise. We assume sufficiently small launch power that we can neglect the Kerr non-linearity [137].

8.1.1. Continuous-Time Model

The system model is shown in Fig. 8.2. The symbol alphabet is $\mathcal{A} = \{a_1, \ldots, a_M\}$ with $M = 2^m$. A source puts out u.i.i.d. symbols $(U_\kappa)_{\kappa\in\mathbb{Z}}$ with $U_\kappa \in \mathcal{A}$. A differential phase mapper $f_{\text{diff}}(\cdot)$ maps these symbols to the transmit symbols $(X_\kappa)_{\kappa\in\mathbb{Z}}$, $X_\kappa \in \mathcal{A}$, using $|X_k| = |U_k|$ and $\angle X_k = \angle X_{k-1} + \angle U_k$ for phase modulation. Additionally, we use the notation

$$X_\kappa = f_{\text{diff}}(U_\kappa, X_{\kappa-1}) \tag{8.1}$$

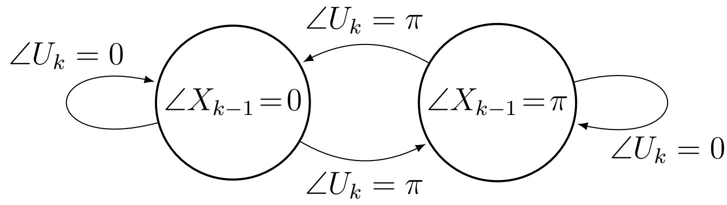


Figure 8.3.: Differential encoding of BPSK.

and

$$U_\kappa = f_{\text{diff}}^{-1}\left(X_\kappa, X_{\kappa-1}\right) \tag{8.2}$$

for the inverse operation of differential encoding. Differential coding helps to resolve phase ambiguities; see [80, Sec. IV] and the Appendix A.2. An example of differential phase encoding for BPSK is shown in Fig. 8.3.

We obtain the continuous-time baseband waveform

$$X(t) = \sum_\kappa X_\kappa \cdot g_{\text{tx}}(t - \kappa T_{\text{s}}) \tag{8.3}$$

with symbol-rate $B = 1/T_{\text{s}}$. We use a FD-RC pulse $g_{\text{tx}}(t)$ which has the spectrum [139, Eq. (6.17)]

$$G_{\text{tx}}(f) = \begin{cases} 1, & |f| \leq \frac{1-\alpha}{2T_{\text{s}}} \\ \frac{1}{2}\left[1 + \cos\left(\frac{\pi T_{\text{s}}}{\alpha}\left[|f| - \frac{1-\alpha}{2T_{\text{s}}}\right]\right)\right], & \frac{1-\alpha}{2T_{\text{s}}} < |f| \leq \frac{1+\alpha}{2T_{\text{s}}} \\ 0, & \text{otherwise} \end{cases} \tag{8.4}$$

with roll-off factor α. Note that one often uses root-raised cosine pulses in memoryless channels, where the pulse shape in combination with a matched filter gives a raised-cosine. For our system with DD and CD, no matched filter is applied and we use a raised-cosine directly for pulse shaping. A roll-off factor $\alpha = 0$ corresponds to the sinc pulse.

The channel exhibits CD with response [50, Sec. II.B]

$$h_L(t) \circ\!\!-\!\!\bullet\, H_L(f) = e^{\mathrm{j}(\beta_2/2)\omega^2 L} \tag{8.5}$$

where β_2 is the group velocity dispersion, $\omega = 2\pi f$, and L is the fiber length. The convolution of $X(t)$ and $h_L(t)$ results in $X_L(t)$. The receiver PD is modeled as a SLD that outputs the intensity $Z'(t) = |X_L(t)|^2$ of the optical signal and the PD noise $N'(t)$ is modeled as a real-valued white Gaussian random process with two-sided power spectral density $N_0/2$.

The resulting signal $Y'(t)$ is filtered by a lowpass filter $g_{\mathrm{rx}}(t)$ with impulse response

$$g_{\mathrm{rx}}(t) = 2B\,\mathrm{sinc}(2Bt) \circ\!\!-\!\!\bullet\, G_{\mathrm{rx}}(f) = \begin{cases} 1, & |f| \leq B \\ 0, & \text{otherwise.} \end{cases} \tag{8.6}$$

The receiver low-pass filter rejects out-of-band noise and avoids aliasing. The sampler converts the analog signal to digital samples with the sampling rate $1/T_{\mathrm{s}}' = 2B$, i.e., we sample with a oversampling factor of $N_{\mathrm{os}} = T_{\mathrm{s}}/T_{\mathrm{s}}' = 2$ samples per transmit symbol. The SLD doubles the bandwidth of the received signal as the squaring corresponds to a self-convolution in frequency, i.e., the bandwidth of the FD-RC signal doubles from $(1+\alpha)B$ to $2(1+\alpha)B$. Note that the bandwidth of the lowpass filter and the received signal are equal for $\alpha = 0$. In case of $\alpha > 0$, the filter removes useful signal components at the band edges. We use small rolloff factors up to $\alpha = 0.2$ to minimize the effect of this operation.

The filtered noise, i.e., the noise present before the sampling device, is a zero-mean stationary Gaussian process with auto-correlation function

$$\frac{N_0}{2}\left(g_{\mathrm{rx}}(tau) * g_{\mathrm{rx}}(\tau)\right) = \frac{N_0}{2} g_{\mathrm{rx}}(\tau). \tag{8.7}$$

8.1.2. Discrete-Time Model

Suppose $g_{\mathrm{tx}}(t)$ is a sinc pulse, i.e., $\alpha = 0$ and the filter $g_{\mathrm{rx}}(t)$ and two-fold oversampling provide sufficient statistics. The channel response up to $X_L(t)$ is $\psi(t) = g_{\mathrm{tx}}(t) * h_L(t)$ and the samples are $\psi_k = \psi(kT_{\mathrm{s}}')$, $k \in \mathbb{Z}$. For simplicity, suppose $\psi_k = 0$ for $k \notin [0, K-1]$, where K is an odd integer. We write the channel Toeplitz matrix as $\boldsymbol{\Psi} \in \mathbb{C}^{2n\times(2n+K-1)}$, which is constructed from the over-sampled channel response

$$\boldsymbol{\psi} = [\psi_{K-1}, \ldots, \psi_0] \in \mathbb{C}^K. \tag{8.8}$$

Consider the vector of noise-free samples

$$\boldsymbol{z} = |\boldsymbol{\Psi}\tilde{\mathbf{x}}'|^{\circ 2} = \begin{bmatrix} z_1, & z_2, & z_3, & \ldots & z_{2n} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{2n\times 1} \tag{8.9}$$

with input $\tilde{\mathbf{x}}' = [\boldsymbol{s}_0^{\mathrm{T}}, (\boldsymbol{x}')^{\mathrm{T}}]^{\mathrm{T}}$ and up-sampled symbols

$$\boldsymbol{x}' = \begin{bmatrix} 0, & x_1, & 0, & x_2, & \ldots, & 0, & x_n \end{bmatrix}^{\mathrm{T}} \in \mathbb{C}^{2n\times 1} \tag{8.10}$$

where the initial state vector is

$$\boldsymbol{s}_0 = \left[0,\ x_{1-\widetilde{K}},\ 0,\ x_{2-\widetilde{K}},\quad \ldots,\quad 0,\ x_0\right]^{\mathrm{T}} \in \mathbb{C}^{(K-1)\times 1} \tag{8.11}$$

and the channel memory is $\widetilde{K} = (K-1)/2$. The discrete-time channel is Gaussian with conditional density

$$p(\boldsymbol{y}|\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{y} - |\boldsymbol{\Psi}\tilde{\boldsymbol{x}}'|^{\circ 2}\ ;\ \mathbf{0}_{2n}, N_0 B \mathbf{I}_{2n}\right). \tag{8.12}$$

Detector	Information Rate	Mismatched Rate	Reference
JDD	I_{JDD}	$I_{q,\text{JDD}}$	(8.15), (8.64)
SDD	I_{SDD}	$I_{q,\text{SDD}}$	(8.19), Sec. 8.4.3
SIC / MSD	$I_{\text{SIC}} = I_{\text{SIC-MSD}}$	$I_{q,\text{SIC}} = I_{q,\text{SIC-MSD}}$	(8.24), Sec. 8.4.3
SIC, bit-wise	$I_{\text{b-SIC}}$	$I_{q,\text{b-SIC}}$	(8.38), Sec. 8.4.3

Table 8.1.: Information rates studied in this chapter.

We introduce the additional notation

$$\boldsymbol{y} = y_1^{2n} = [y_1, y_2, \dots, y_{2n}] = \left[y_1^{\diamondsuit}, y_1^{\spadesuit}, y_2^{\diamondsuit}, \dots, y_n^{\spadesuit}\right] \tag{8.13}$$

to distinguish between odd and even samples.

For $\alpha > 0$, the bandwidth of the signal $\psi(t) = g_{\text{tx}}(t) * h_L(t)$ is larger then $2B$ and double oversampling does not give sufficient statistics. Therefore, we use four-fold over-sampling during our simulations followed by low-pass filtering and down-sampling to 2-fold oversampling (c.f. [80, Eq. (60)]). As a statistical model of the channel, we still use the density from (8.12) with the double over-sampled Toeplitz matrix $\boldsymbol{\Psi}$. This results in a mismatch between the actual model used for simulation and the channel model that is used to compute densities.

8.2. Detection and Decoding Strategies and their Rates

An extensive rate analysis for the described scenario is presented in [80] and is based on JDD. JDD is usually infeasible in practical systems. We discuss SDD and detection based on SIC to reduce the complexity. Another approach would be turbo detection and decoding (TDD), where SDD is used in a turbo loop to exchange extrinsic information between the detector and the decoder [140, 141]. TDD requires specific code designs that are tailored to the TDD approach. We therefore focus on SDD and SIC where "off-the-shelf" binary codes can be used.

We consider rates for a fixed block length n and for $n \to \infty$. The resulting rates are listed in Table 8.1 for convenience.

8.2.1. Joint Detection and Decoding

We start by considering JDD which gives bounds for practical systems. Usually detection and decoding are separated, i.e., a detector computes either a hard decision or some soft metric for a symbol or bit and passes the results to a decoder. In JDD, detection and decoding is done jointly. We obtain the JDD rate

$$I_{n,\mathrm{JDD}} = \mathbb{I}_n\left(\boldsymbol{U};\boldsymbol{Y}\right) = \mathbb{H}_n\left(\boldsymbol{U}\right) - \mathbb{H}_n\left(\boldsymbol{U}|\boldsymbol{Y}\right) \tag{8.14}$$

with the entropy and MI rate as defined in Sec. 2.1. The limiting rate is

$$I_{\mathrm{JDD}} = \lim_{n\to\infty} I_{n,\mathrm{JDD}}. \tag{8.15}$$

8.2.2. Separate Detection and Decoding

In symbol-wise SDD, a detector computes symbol-wise APPs $P(u_\kappa, \boldsymbol{y})$, $\kappa = 1,\ldots,n$. These APPs are forwarded to a decoder and treated as independent of each other (c.f. [131,142]). To get the rates, we must replace the joint entropy expression $\mathbb{H}_n\left(\boldsymbol{U}|\boldsymbol{Y}\right)$ by the marginal entropy expressions. We obtain the bound

$$\begin{aligned}
\mathbb{I}_n\left(\boldsymbol{U};\boldsymbol{Y}\right) &= \mathbb{H}_n\left(\boldsymbol{U}\right) - \mathbb{H}_n\left(\boldsymbol{U}|\boldsymbol{Y}\right) && (8.16)\\
&= \mathbb{H}_n\left(\boldsymbol{U}\right) - \sum_{\kappa=1}^{n} \mathbb{H}\left(U_\kappa|\boldsymbol{Y}, U_1^{\kappa-1}\right) && (8.17)\\
&\geq \mathbb{H}_n\left(\boldsymbol{U}\right) - \frac{1}{n}\sum_{\kappa=1}^{n} \mathbb{H}\left(U_\kappa|\boldsymbol{Y}\right) := I_{n,\mathrm{SDD}} && (8.18)
\end{aligned}$$

and the SDD limiting rate is

$$I_{\mathrm{SDD}} = \lim_{n\to\infty} I_{n,\mathrm{SDD}}. \tag{8.19}$$

Equation (8.18) implies $I_{\mathrm{SDD}} \leq I_{\mathrm{JDD}}$, i.e., there is an information loss due to neglecting $U_1^{\kappa-1}$ in (8.17) that grows with the channel memory (c.f. [11, Fig. 5], [143, Figs. 7-9], [144, Fig. 12]).

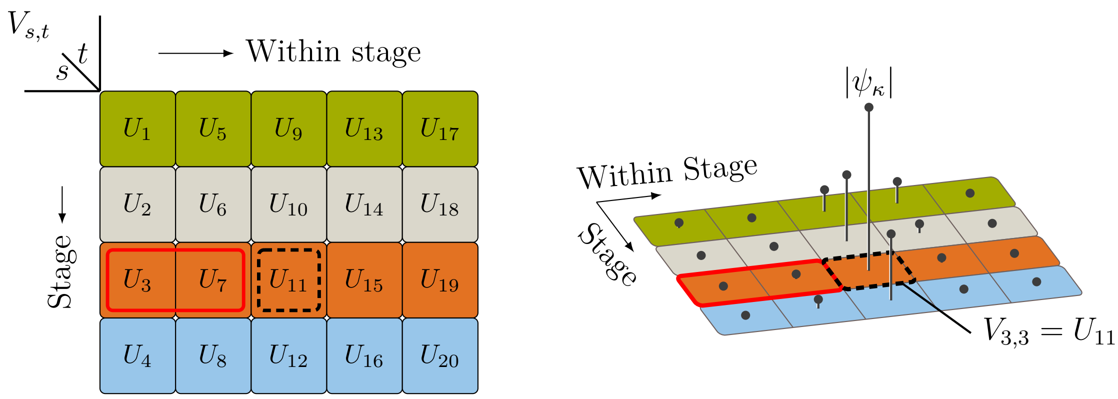
8.2.3. Successive Interference Cancellation

SIC combines SDD with MLC and multistage detection/decoding (MSD) so the SDD loss within one stage is reduced [5, 11, 12]. SIC performs a serial-to-parallel (S/P) conversion on the information $\boldsymbol{U}$, i.e., down-sample $\boldsymbol{U}$ to create S shorter strings of length $N = n/S$ (assume $N \in \mathbb{Z}$). We write

$$\boldsymbol{V}_s = \left(V_{s,t}\right)_{t=1}^{N} = \left(U_s, U_{s+S}, \ldots, U_{s+(N-1)S}\right) \tag{8.20}$$

and form the vector $\boldsymbol{V} = (\boldsymbol{V}_s)_{s=1}^{S}$. Fig. 8.4a illustrates the S/P conversion for an input string U^{20} with $S = 4$ and $N = 5$. The four rows in Fig. 8.4a represent $\boldsymbol{V}_1, \ldots, \boldsymbol{V}_4$.

We relate the JDD, SDD, and SIC rates; see [11, 12]. Since $\boldsymbol{V}$ is a resorted version of



(a) S/P conversion of an input string U^{20} to $V_{s,t}$, $s = 1, \ldots, 4$ and $t = 1, \ldots, 5$.

(b) ISI with SIC.

Figure 8.4.: SIC example.

$\boldsymbol{U}$, we have $I_n(\boldsymbol{U};\boldsymbol{Y}) = I_n(\boldsymbol{V};\boldsymbol{Y})$ and

$$\mathbb{I}_n\left(\boldsymbol{V};\boldsymbol{Y}\right) \overset{(a)}{=} \mathbb{H}_n\left(\boldsymbol{V}\right) - \frac{1}{n}\sum_{s=1}^{S} \mathbb{H}\left(\boldsymbol{V}_s|\boldsymbol{Y},\boldsymbol{V}^{s-1}\right) \tag{8.21}$$

$$\overset{(b)}{\geq} \underbrace{\mathbb{H}_n\left(\boldsymbol{V}\right) - \frac{1}{n}\sum_{s=1}^{S}\sum_{t=1}^{N} \mathbb{H}\left(V_{s,t}|\boldsymbol{Y},\boldsymbol{V}^{s-1}\right)}_{:=\ I_{n,\mathrm{SIC}}} \tag{8.22}$$

$$\overset{(c)}{\geq} \underbrace{\mathbb{H}_n\left(\boldsymbol{V}\right) - \frac{1}{n}\sum_{s=1}^{S}\sum_{t=1}^{N} \mathbb{H}\left(V_{s,t}|\boldsymbol{Y}\right)}_{=\ I_{n,\mathrm{SDD}}} \tag{8.23}$$

where step (a) follows from the chain rule of entropy; steps (b)-(c) follow because conditioning cannot increase entropy; and step (c) recovers the SDD rate. The limiting SIC rate is

$$I_{\mathrm{SIC}} = \lim_{n\to\infty} I_{n,\mathrm{SIC}}. \tag{8.24}$$

Fig. 8.4b shows how SIC reduces interference. The colored strips are the same as in Fig. 8.4a and the points show the magnitudes $|\psi_k|$ of the channel response samples due to the symbol $V_{3,3} = U_{11}$. SIC successively detects and decodes the rows in Fig. 8.4a with the vectors $\boldsymbol{V}_1, \ldots, \boldsymbol{V}_4$. The rates (8.21) and (8.22) are the same if $V_{s,1}, \ldots, V_{s,N}$ are independent given $\boldsymbol{Y}$ and $\boldsymbol{V}^{s-1}$ for each s. For example, consider $\boldsymbol{V}_3$ and in particular $V_{3,3}$. The channel outputs $\boldsymbol{Y}$ that are relevant for detecting $V_{3,3} = U_{11}$ are hardly affected by $V_{3,1} = U_3$ and $V_{3,2} = U_7$ that are neglected in SIC.

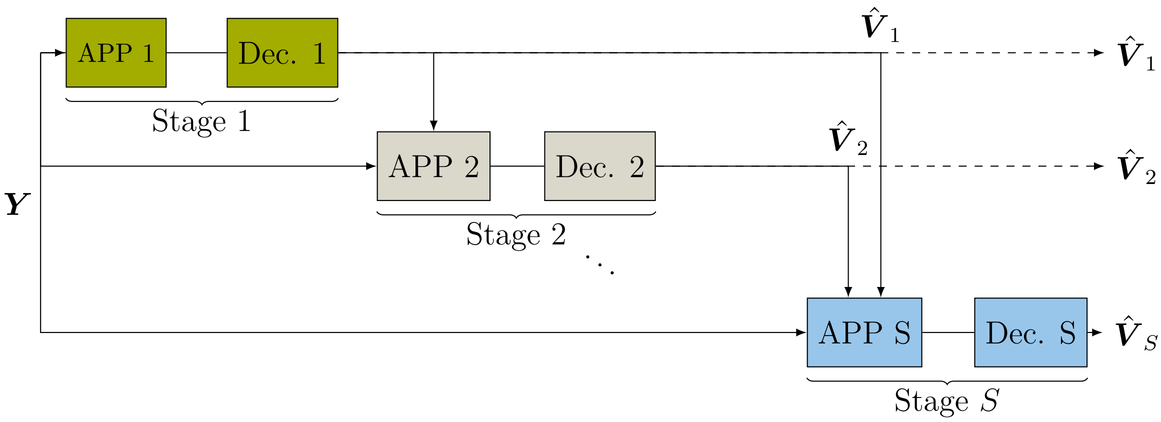


Figure 8.5.: SIC receiver structure.

We describe properties of the SIC rates. First, rewrite $I_{n,\mathrm{SIC}}$ in (8.22) as

$$I_{n,\mathrm{SIC}} = \frac{1}{S}\sum_{s=1}^{S} \underbrace{\frac{1}{N}\sum_{t=1}^{N} I\left(V_{s,t}; \boldsymbol{Y}, \boldsymbol{V}^{s-1}\right)}_{:= I_{N,\mathrm{SIC}}^{(s)}} \tag{8.25}$$

i.e., write the SIC rate as a sum of S rates of each stage. The expression (8.25) suggests the receiver in Fig. 8.5 with S stages, each with a symbol-wise APP detector and decoder. In SIC stage s, the detector computes the APPs

$$P\left(v_{s,t}|\boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1}\right), \quad t = 1, \ldots, N \tag{8.26}$$

where $\hat{\boldsymbol{V}}^{s-1}$ is the vector of decoded symbols from previous stages.

Next, suppose $\boldsymbol{V}_s$ is encoded with a rate less than the limiting rate [11, Sec. III a)]

$$I_{\mathrm{SIC}}^{(s)} = \lim_{N\to\infty} I_{N,\mathrm{SIC}}^{(s)}. \tag{8.27}$$

This allows reliable decoding as the block length grows, and we may assume $\boldsymbol{V}_s = \hat{\boldsymbol{V}}_s$ and the rates of the individual stages are non-decreasing

$$I_{\mathrm{SIC}}^{(1)} \leq I_{\mathrm{SIC}}^{(2)} \leq \ldots I_{\mathrm{SIC}}^{(S)}. \tag{8.28}$$

By (8.21)–(8.23) we have $I_{\mathrm{SDD}} \leq I_{\mathrm{SIC}} \leq I_{\mathrm{JDD}}$. We also have

$$\lim_{S\to\infty} I_{\mathrm{SIC}} = I_{\mathrm{JDD}}. \tag{8.29}$$

We remark that the length of the component codes in a SIC scheme is given as n/S in contrast to the full length n in a SDD scheme. We know from information theory (c.f. [145, 146]) that decreasing the block length of a code deteriorates the achievable rate. Alternatively, one could increase the block length in a SIC scheme by a factor of S as compared to a SDD scheme at the cost of also increasing the decoding latency by a factor of S. This leads to a trade-off for the number of SIC stage, even though the rate expressions above suggest to choose S as large as possible.

To mitigate the effect of shorter component block lengths, we use polar codes with multistage list decoding as described in Sec. 4.3 where a list of possible codewords is passed across decoding stages.

8.3. Binary Codes for SIC

The previous section derived rates achieved by JDD, SDD, and SIC. The rates and their expressions are based on symbols and would require channel codes based on symbols. Most practical codes are binary. This section combines SIC with binary codes and describes the corresponding rates. We use the notations for bit labelings of symbols as defined in Sec. 2.4.1. We use the same constellation alphabet $\mathcal{A}$ with $|\mathcal{A}| = M$ across all SIC stages. The number of bits per symbol is then $m = \log_2 M$. Note that it is also possible to use different constellation sizes, e.g., a smaller constellation size in the first SIC stages.

8.3.1. SIC Receiver

The structure of a SIC receiver based on symbols is depicted in Fig. 8.5. A demapper calculates APPs for all symbols of the current decoding stage and passes them to a symbol-wise decoder. We extend this scheme to work with binary codes. Consider the following inequalities that result in three different binary schemes:

$$
\begin{aligned}
I_{n,\mathrm{JDD}} &= \mathbb{I}_n\left(\boldsymbol{V};\boldsymbol{Y}\right) && (8.30)\\
&= \mathbb{H}_n\left(\boldsymbol{V}\right) - \mathbb{H}_n\left(\boldsymbol{V}|\boldsymbol{Y}\right) && (8.31)\\
&= \mathbb{H}_n\left(\boldsymbol{V}\right) - \frac{1}{n}\sum_{s=1}^{S}\mathbb{H}\left(\boldsymbol{V}_s|\boldsymbol{Y},\boldsymbol{V}^{s-1}\right) && (8.32)\\
&= \mathbb{H}_n\left(\boldsymbol{V}\right) - \frac{1}{n}\sum_{s=1}^{S}\sum_{l=1}^{m}\mathbb{H}\left(b_l(\boldsymbol{V}_s)|\boldsymbol{Y},\boldsymbol{V}^{s-1},b_1^{l-1}(\boldsymbol{V}_s)\right) && (8.33)\\
&\geq \underbrace{\mathbb{H}_n\left(\boldsymbol{V}\right) - \frac{1}{n}\sum_{s=1}^{S}\sum_{l=1}^{m}\sum_{t=1}^{N}\mathbb{H}\left(b_l(V_{s,t})|\boldsymbol{Y},\boldsymbol{V}^{s-1},b_1^{l-1}(\boldsymbol{V}_s)\right)}_{=I_{n,\text{b-SIC}}} && (8.34)\\
&\geq \underbrace{\mathbb{H}_n\left(\boldsymbol{V}\right) - \frac{1}{n}\sum_{s=1}^{S}\sum_{l=1}^{m}\sum_{t=1}^{N}\mathbb{H}\left(b_l(V_{s,t})|\boldsymbol{Y},\boldsymbol{V}^{s-1},b_1^{l-1}(V_{s,t})\right)}_{=I_{n,\text{SIC}}=I_{n,\text{SIC-MSD}}} && (8.35)
\end{aligned}
$$

$$\underbrace{\geq \mathbb{H}_n\left(\boldsymbol{V}\right) - \frac{1}{n}\sum_{s=1}^{S}\sum_{l=1}^{m}\sum_{t=1}^{N}\mathbb{H}\left(b_l(V_{s,t})|\boldsymbol{Y},\boldsymbol{V}^{s-1}\right)}_{=I_{n,\text{SIC-BMD}}} \tag{8.36}$$

b-SIC Receiver

We have the bit-wise successive interference cancellation (b-SIC) rate $I_{n,\text{b-SIC}}$ in (8.34) that is at least the SIC-rate $I_{n,\text{SIC}}$ in (8.25) based on symbols. The idea is to extend the SIC scheme to work with bits instead of symbols. We have another m detecting and decoding stages within each of the S SIC decoding stages that are successively detected and decoded. The bit-wise detectors compute the APPs

$$P\left(b_l(v_{s,t})|\boldsymbol{y},\hat{\boldsymbol{v}}^{s-1},\hat{b}_1^{l-1}(\boldsymbol{v}_s)\right), \tag{8.37}$$

i.e., the detector computes the probability of the l-th bit of symbol $V_{s,t}$ based on knowing all the symbols $\hat{\boldsymbol{v}}^{s-1}$ from previous SIC stages and based on knowing bit-level one to $l-1$ of all symbols $\boldsymbol{v}_s$ of the current SIC stage. The APPs are passed to a decoder and the resulting maximum a posteriori probabilit (MAP) bit estimates $\hat{b}_l(\boldsymbol{v}_s)$ are then passed to the next detector.

We rewrite the rate (8.34) and obtain

$$I_{n,\text{b-SIC}} = \frac{1}{S}\sum_{s=1}^{S}\sum_{l=1}^{m}\underbrace{\frac{1}{N}\sum_{t=1}^{N}\mathbb{I}\left(b_l(V_{s,t});\boldsymbol{Y},\boldsymbol{V}^{s-1},b_1^{l-1}(\boldsymbol{V}_s)\right)}_{:=I_{N,\text{b-SIC}}^{(s,l)}} \tag{8.38}$$

where $I_{N,\text{b-SIC}}^{(s,l)}$ is the information rate of bit level l in SIC stage s. Encoding the bit level l of SIC stage s with rate less than

$$I_{\text{b-SIC}}^{(s,l)} = \lim_{N\to\infty} I_{N,\text{b-SIC}}^{(s,l)} \tag{8.39}$$

allows reliable decoding of $b_l(\boldsymbol{V}_s)$ as the block length grows. By successively decoding over the bit-levels $l=1,\ldots,m$ and SIC stages $s=1,\ldots,S$, we assume that $\hat{b}_l(\boldsymbol{V}_s) = b_l(\boldsymbol{V}_s)$ and $\hat{\boldsymbol{V}}^{s-1} = \boldsymbol{V}^{s-1}$ and thus b-SIC achieves the limiting rate

$$I_{\text{b-SIC}} = \lim_{S\to\infty} I_{n,\text{b}-\text{SIC}} \geq I_{\text{SIC}}. \tag{8.40}$$

SIC-MSD Receiver

For the SIC-MSD receiver, instead of conditioning on the previous bit-levels $1,\ldots,l-1$ of all symbols $\boldsymbol{V}_s$ of the current SIC stage as in (8.37), we limit the conditioning to previous bit-levels of the current symbol $V_{s,t}$, i.e., we calculate the APPs

$$P\left(b_l(v_{s,t})|\boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1}, \hat{b}_1^{l-1}(v_{s,t})\right). \tag{8.41}$$

One advantage of this scheme is that conditioning on one symbol does not introduce memory of the other symbols of the current stage into the APP, resulting in a lower complexity to compute the APPs. Basically, one can calculate the symbol-wise APPs $P\left(v_{s,t}|\boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1}\right)$ and marginalize them after each decoding step to obtain the bit-wise APPs. This does not require additional runs of a complex detector per bit stage.

We rewrite (8.35) in terms of MI and obtain

$$I_{n,\text{SIC-MSD}} = \frac{1}{S}\sum_{s=1}^{S}\sum_{l=1}^{m}\underbrace{\frac{1}{N}\sum_{t=1}^{N}\mathbb{I}\left(b_l(V_{s,t}); \boldsymbol{Y}, \boldsymbol{V}^{s-1}, b_1^{l-1}(V_{s,t})\right)}_{:=I_{N,\text{SIC-MSD}^{(s,l)}}}. \tag{8.42}$$

By the same arguments as for the b-SIC receiver, we obtain the limiting rate

$$I_{\text{SIC-MSD}} = \lim_{n\to\infty} I_{n,\text{SIC-MSD}}. \tag{8.43}$$

SIC-BMD Receiver

For the SIC-BMD receiver, we neglect the dependencies between the bit-levels. This results in fewer detecting and decoding stages, as we jointly detect all bit-levels of one SIC stage and jointly decode them. The detector computes the APPs

$$P\left(b_l(v_{s,t})|\boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1}\right). \tag{8.44}$$

We rewrite (8.36) in terms of MI and obtain

$$I_{n,\text{SIC-BMD}} = \frac{1}{S}\sum_{s=1}^{S}\sum_{l=1}^{m}\frac{1}{N}\sum_{t=1}^{N}\mathbb{I}\left(b_l(V_{s,t}); \boldsymbol{Y}, \boldsymbol{V}^{s-1}\right). \tag{8.45}$$

By the same arguments as for the b-SIC receiver, we obtain the limiting rate

$$I_{\text{SIC-BMD}} = \lim_{n\to\infty} I_{n,\text{SIC-BMD}}. \tag{8.46}$$

We do not provide rate results for SIC-BMD in this thesis but mention them here for the sake of completeness.

8.3.2. SIC Transmitter

Fig. 8.6 shows the basic encoder structure for SIC. A binary source provides $k \leq n \cdot m$ u.i.i.d. bits $\boldsymbol{D}$. The S/P converter partitions $\boldsymbol{D}$ into S strings $\boldsymbol{D}_1, \dots, \boldsymbol{D}_S$ for each stage where $\boldsymbol{D}_s$ has length k_s. Each $\boldsymbol{D}_s$ is encoded to a length Nm bit string $\boldsymbol{B}_s$, so the code

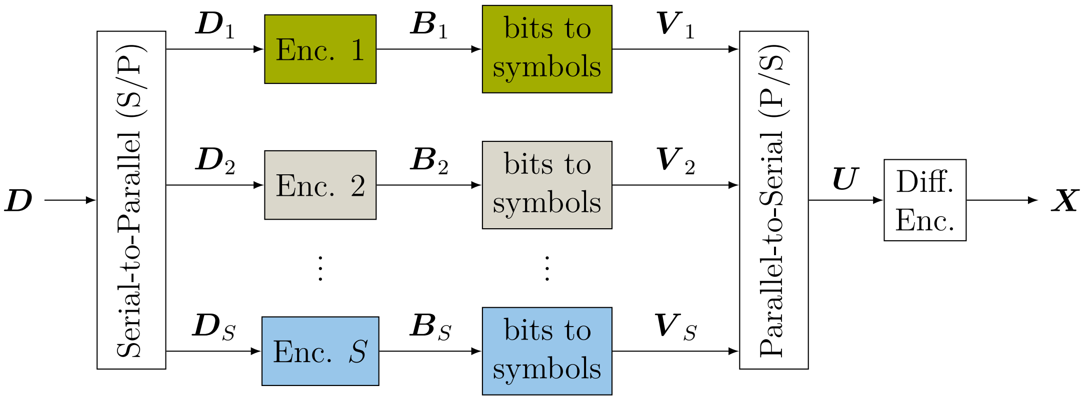


Figure 8.6.: SIC encoder for binary codes.

rate per stage is $k_s/(Nm)$. The bits $\boldsymbol{B}_s$ are mapped to the length N string $\boldsymbol{V}_s$ of symbols from the modulation alphabet $\mathcal{A}$. The information rate in bpcu for SIC stage s is

$$R_s = k_s/\,N \text{ [bpcu]} \tag{8.47}$$

and the overall information rate is

$$R_{\text{SIC}} = \frac{1}{S}\sum_{s=1}^{S} R_s = \frac{k}{n} \text{ [bpcu]}. \tag{8.48}$$

Finally, the strings $\boldsymbol{V}_1, \ldots, \boldsymbol{V}_S$ are passed to a parallel-to-serial (P/S) converter that reverses (8.20) and produces $\boldsymbol{U}$.

In case of b-SIC and SIC-MSD, we have M encoders per SIC stage, i.e., each $\boldsymbol{D}_s$ is split into M strings $\boldsymbol{D}_{s,l}$ for $l = 1, \ldots, M$. The block length in bits of the underlying channel codes is $n/S = N$ in case of b-SIC and SIC-MSD and $nm/S = Nm$ for SIC-BMD.

8.4. Achievable Rate Computations

In this section, we show how the rates from the last sections can be calculated.

8.4.1. JDD Rate

For JDD, we must calculate the MI

$$\begin{aligned}
\mathbb{I}_n(\boldsymbol{U};\boldsymbol{Y}) &\overset{(a)}{=} \mathbb{I}_n(\boldsymbol{X};\boldsymbol{Y}) && (8.49)\\
&= h_n(\boldsymbol{Y}) - h_n(\boldsymbol{Y}|\boldsymbol{X}) && (8.50)\\
&\overset{(b)}{=} h_n(\boldsymbol{Y}) - h_n(\boldsymbol{N}) && (8.51)
\end{aligned}$$

where (a) follows from $\boldsymbol{X}$ being a function of $\boldsymbol{U}$ given the initial state and (b) holds since the statistics of $\boldsymbol{Y}$ given the channel input $\boldsymbol{X}$ depend only on the noise vector $\boldsymbol{N}$. The term $h_n(\boldsymbol{N})$ is easy to compute for AWGN. It remains to compute the term $h_n(\boldsymbol{Y})$. We can approximate this term by Monte-Carlo simulations as described in [64]:

$$h_n(\boldsymbol{Y}) \approx -\frac{1}{n} \log_2 p(\boldsymbol{y}) \tag{8.52}$$

where $\boldsymbol{y}$ is a realization of $\boldsymbol{Y}$ obtained in the Monte-Carlo simulation. The approximation becomes exact in the limit $n \to \infty$ since $\boldsymbol{Y}$ comes from an ergodic and cyclostationary process with period $N_{\text{os}} = 2$.

We calculate the density $p(\boldsymbol{y})$ by the BCJR or FBA [65, 80]. Define the channel state at time κ as

$$s_{\kappa-1} = \left(x_{\kappa-\widetilde{K}}, \ldots, x_{\kappa-1} \right). \tag{8.53}$$

The density $p(\boldsymbol{y})$ can be computed from the state metrics in the forward path of the FBA as

$$p(\boldsymbol{y}) = \sum_{s_n} \sum_{\substack{u_n \\ s_{n-1}}} \cdots \overbrace{\sum_{\substack{u_2 \\ s_1}} \underbrace{\sum_{\substack{u_1 \\ s_0}} p(s_0) \cdot p(x_1, y_1^{\diamondsuit}, y_1^{\spadesuit}, s_1|s_0)}_{\vec{\mu}(s_1)} \cdot p(x_2, y_2^{\diamondsuit}, y_2^{\spadesuit}, s_2|s_1)}^{\vec{\mu}(s_2)} \cdot \tag{8.54}$$

with the state metrics $\vec{\mu}(s_\kappa) = p(y_\kappa^{\diamondsuit}, y_\kappa^{\spadesuit}, s_\kappa)$ Since the state metrics quickly approach zero as κ grows, we normalize as described in [64]. The product terms from (8.54) can be decomposed as

$$p\left(x_\kappa, y_\kappa^{\diamondsuit}, y_\kappa^{\spadesuit}, s_\kappa|s_{\kappa-1}\right) = \underbrace{p\left(y_\kappa^{\diamondsuit}, y_\kappa^{\spadesuit}|x_\kappa, s_{\kappa-1}\right)}_{(a)} \cdot \underbrace{p\left(s_\kappa|x_\kappa, s_{\kappa-1}\right)}_{(b)} \cdot \underbrace{p\left(x_\kappa|s_{\kappa-1}\right)}_{(c)}. \tag{8.55}$$

The term (a) is

$$p\left(y_\kappa^{\diamondsuit}, y_\kappa^{\spadesuit}|x_\kappa, s_{\kappa-1}\right) = p\left(y_\kappa^{\diamondsuit}|s_{\kappa-1}\right) \cdot p\left(y_\kappa^{\spadesuit}|x_\kappa, s_{\kappa-1}\right) \tag{8.56}$$

with

$$p\left(y_\kappa^{\diamondsuit}|s_{\kappa-1}\right) = p_N\left(y_\kappa^{\diamondsuit} - \left|\left[0, x_{\kappa-\widetilde{K}}, 0, \ldots, x_{\kappa-1}, 0\right] \cdot \boldsymbol{\psi}^{\mathrm{T}}\right|^2\right) \tag{8.57}$$

and

$$p\left(y_\kappa^{\spadesuit}|x_\kappa, s_{\kappa-1}\right) = p_N\left(y_\kappa^{\diamondsuit} - \left|\left[x_{\kappa-\widetilde{K}}, 0, \ldots, x_{\kappa-1}, 0, x_\kappa\right] \cdot \boldsymbol{\psi}^{\mathrm{T}}\right|^2\right) \tag{8.58}$$

with the channel vector $\boldsymbol{\psi}$ from (8.8).

The term (b) describes if there is a state transition between state $s_{\kappa-1}$ and s_κ for the channel input $X_\kappa = x_\kappa$. The term (b) takes the value one if the state transmission exists and zero otherwise. The evaluation is straightforward with the state definition in (8.53). The term (c) evaluates to

$$p\left(x_\kappa | s_{\kappa-1}\right) = p(x_\kappa) \tag{8.59}$$

due to independent and identically distributed (i.i.d.) signaling.

8.4.2. Rates for SIC and SDD

To compute achievable rates for SDD and the described variants of SIC, we must compute the entropy terms in (8.23), (8.34), (8.35), and (8.36). Since our channel model represents an ergodic and stationary process, the entropy terms are independent of the time index, e.g., for SDD, we have $\mathbb{H}\left(U_\kappa|\boldsymbol{Y}\right) = \mathbb{H}\left(U_\tau|\boldsymbol{Y}\right) = \mathbb{H}\left(U|\boldsymbol{Y}\right)$ with $\kappa \neq \tau$. We can approximate this term with samples from a Monte-Carlo simulation as

$$\mathbb{H}\left(U|\boldsymbol{Y}\right) \approx \frac{1}{n}\sum_{\kappa=1}^{n} -\log_2 P(u_\kappa|\boldsymbol{y}). \tag{8.60}$$

For the variants of SIC, we have a similar behaviour. The entropy terms within one SIC-stage and bit-level are independent of the time index t, e.g., we have

$$\mathbb{H}\left(b_l(V_{s,t})|\boldsymbol{Y}, \boldsymbol{V}^{s-1}, b_1^{l-1}(\boldsymbol{V}_s)\right) = \mathbb{H}\left(b_l(V_{s,\tau})|\boldsymbol{Y}, \boldsymbol{V}^{s-1}, b_1^{l-1}(\boldsymbol{V}_s)\right) \tag{8.61}$$

for b-SIC.

It remains to show how the APPs needed for the approximations can be calculated. In Sec. 8.5, we propose a detector based on the FBA and in Sec. 8.6, we suggest a detector based on Gibbs sampling that can approximate the APPs.

8.4.3. Mutual Information Rates under Mismatched Decoding

Calculating the forward metrics in (8.54) requires marginalizing over all states s_κ. The number of states for an alphabet of size M and a channel with length $\widetilde{K}$ is $M^{\widetilde{K}}$, which grows exponentially and is often not feasible. To reduce complexity, we use the concept of mismatched decoding; see [38, Ex. 5.22], [147–150]. We replace the true channel model (8.12) with

$$q(\boldsymbol{y}|\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{y} - \left|\boldsymbol{\Psi}'\tilde{\boldsymbol{x}}'\right|^{\circ 2}\ ;\ \boldsymbol{\mu}_{\mathbf{Q}}, \mathbf{C}_{\mathbf{QQ}}\right). \tag{8.62}$$

where the Toeplitz matrix $\boldsymbol{\Psi}'$ is constructed the same way as $\boldsymbol{\Psi}$ in Sec. 8.1.2 but with a channel memory $\widetilde{K}_q$ smaller than the actual channel memory $\tilde{K}$, i.e., we remove samples from the edges of the channel vector (8.8). The mean $\boldsymbol{\mu}_{\mathbf{Q}}$ and covariance matrix $\mathbf{C}_{\mathbf{QQ}}$ are optimized as described in [80, Sec. III.C] with a diagonal matrix $\mathbf{C}_{\mathbf{QQ}}$.

Let $q(\boldsymbol{y}) = \sum_{\boldsymbol{u}} P(\boldsymbol{u})q(\boldsymbol{y}|\boldsymbol{u})$ and consider the mismatched rates

$$\mathbb{I}_{q,n}\left(\boldsymbol{U};\boldsymbol{Y}\right) := \frac{1}{n}\mathbb{E}\left[\log_2 \frac{q(\boldsymbol{Y}|\boldsymbol{U})}{q(\boldsymbol{Y})}\right] \tag{8.63}$$

$$I_{q,\text{JDD}} := \lim_{n\to\infty} \frac{1}{n}\mathbb{I}_{q,n}\left(\boldsymbol{U};\boldsymbol{Y}\right) \tag{8.64}$$

where the expectation is over the actual $P(\boldsymbol{u})p(\boldsymbol{y}|\boldsymbol{u})$. We have (see [80, Sec. III.B] and [64])

$$\mathbb{I}_{q,n}\left(\boldsymbol{U};\boldsymbol{Y}\right) \leq \mathbb{I}_{n}\left(\boldsymbol{U};\boldsymbol{Y}\right), \quad I_{q,\text{JDD}} \leq I_{\mathbf{JDD}}. \tag{8.65}$$

To approximate the JDD rate, we do the same as in (8.52) but with $q(\boldsymbol{y})$ instead of $p(\boldsymbol{y})$, i.e., we have

$$h_{q,n}(\boldsymbol{Y}) \approx -\frac{1}{n}\log_2 q(\boldsymbol{y}) \tag{8.66}$$

where the realizations $\boldsymbol{y}$ must be drawn from the actual channel model $P(\boldsymbol{u})p(\boldsymbol{y}|\boldsymbol{u})$ since the expectation in (8.63) is also over the actual channel $P(\boldsymbol{u})p(\boldsymbol{y}|\boldsymbol{u})$. In the same way, we compute the limiting rates $I_{q,\text{SDD}}$, $I_{q,\text{SIC}} = I_{q,\text{SIC-MSD}}$ and $I_{q,\text{b-SIC}}$ as lower bounds on the respective I_{SDD}, $I_{\text{SIC}} = I_{\text{SIC-MSD}}$, and $I_{\text{b-SIC}}$. We refer to Table 8.1 for the rates discussed in this chapter.

8.5. Forward-Backward Detector

In Sec. 8.4.1, we show how the forward path of the FBA can compute $p(\boldsymbol{y})$. We can use the full FBA to calculate the APPs needed for SDD, SIC, b-SIC, and SIC-MSD. We describe the method in this section based on SDD.

For SDD, we need to calculate the symbol APPs $P(u_\kappa|\boldsymbol{y})$. We have

$$P(u_\kappa|\boldsymbol{y}) \sim p(u_\kappa, \boldsymbol{y}) \tag{8.67}$$

because $p(\boldsymbol{y})$ is a constant with respect to the decisions on u_κ. We can obtain $p(u_\kappa, \boldsymbol{y})$ by the marginalization

$$p(u_\kappa, \boldsymbol{y}) = \sum_{s_0}\sum_{\boldsymbol{u}\setminus u_\kappa} p(s_0, \boldsymbol{u}, \boldsymbol{y}) \tag{8.68}$$

where we marginalize over all possible initial states and all possible input vectors $\boldsymbol{u}$ with the κ-th entry being fixed to u_κ. This marginalization can be efficiently calculated by the FBA [65], [151, Sec. IV.A]. We compute the metrics on the forward (FW) path as described in Sec. 8.4.1:

$$\vec{\mu}(s_\kappa) = \sum_{u_\kappa, s_{\kappa-1}} p(x_\kappa, y_\kappa^\diamondsuit, y_\kappa^\spadesuit, s_\kappa|s_{\kappa-1}) \cdot \vec{\mu}(s_{\kappa-1}) \tag{8.69}$$

where x_κ is a function of u_κ and $s_{\kappa-1}$ from differential encoding. Additionally, we have the metrics of the backward (BW) path:

$$\overleftarrow{\mu}(s_{\kappa-1}) = \sum_{u_\kappa, s_\kappa} p(x_\kappa, y_\kappa^\diamondsuit, y_\kappa^\spadesuit, s_\kappa | s_{\kappa-1}) \cdot \overleftarrow{\mu}(s_\kappa). \tag{8.70}$$

The forward and backward metrics are initialized with $\vec{\mu}(s_0) = P(s_0)$ and $\overleftarrow{\mu}(s_\kappa) = P(s_\kappa)$. The FW and BW path metrics are normalized after every iteration to avoid numerical problems.

The SDD APPs can be calculated from the FW and BW metrics as follows [151]:

$$p(u_\kappa, \boldsymbol{y}) = \sum_{s_{\kappa-1}} \sum_{s_\kappa} \vec{\mu}(s_{\kappa-1}) \cdot p(x_\kappa, y_\kappa^\diamondsuit, y_\kappa^\spadesuit, s_\kappa | s_{\kappa-1}) \cdot \overleftarrow{\mu}(s_\kappa). \tag{8.71}$$

The APPs for SIC and the binary variants can be obtained by the FBA by applying changes to the underlying trellis. Fixing values from previous stages corresponds to removing edges from the underlying trellis or setting the corresponding (b) from (8.55) to zero. The same can be done based on bits for b-SIC.

8.6. Gibbs Sampling Detector

The complexity of the FBA is exponential in the channel memory $\widetilde{K}$ and must be greatly reduced in the mismatched model to be feasible to perform the APP calculations. Gibbs sampling reduces the complexity to quadratic in $\widetilde{K}$, see Table 8.2, by generating samples from the joint PMFs

- ▷ $P(\boldsymbol{v}|\boldsymbol{y})$
- ▷ $P(\boldsymbol{v}_s|\boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1})$
- ▷ $P(b_l(\boldsymbol{v}_s)|\boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1}, \hat{b}_1^{l-1}(\boldsymbol{v}_s))$
- ▷ $P(b_l(\boldsymbol{v}_s)|\boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1})$

for SDD, SIC, b-SIC, and SIC-BMD, respectively.

Using Gibbs sampling for SIC-MSD requires to sample from the joint densities

$$P(b_l(\boldsymbol{v}_s)|\boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1}, \hat{b}_1^{l-1}(v_{s,t})), \quad \forall t = 1, \dots, N, \tag{8.72}$$

i.e., we must sample from a different large density for each $t = 1, \dots, N$. We therefore focus on b-SIC with Gibbs sampling in the following.

8.6.1. Gibbs Sampling with b-SIC

During Gibbs sampling, we sample all bits that have not yet been decoded in one of the previous SIC stages or bit-levels of the current SIC stage. Consider SIC stage s and bit

Algorithm	Multiplications per Iteration	Iterations
FBA	$\mathcal{O}\left(n \cdot S \cdot \lvert\mathcal{A}\rvert^{\widetilde{K}+1}\right)$	1
Gibbs (b-SIC)	$\mathcal{O}\left(n \cdot S \cdot m \cdot \widetilde{K}^2 \cdot N_{\text{par}}\right)$	60

Table 8.2.: Algorithmic complexity for channel memory $\widetilde{K}$. The parameter N_{par} refers to the number of parallel samplers.

stage l for $t = 1, \ldots, N$. To approximate the APPs (8.37), we sample from the joint posterior PMF of the bits that are not yet decoded:

$$P\left(\widetilde{\boldsymbol{b}}_{s,l} \,\middle|\, \boldsymbol{y}, b(\hat{\boldsymbol{v}}^{s-1}), \hat{b}_1^{l-1}(\boldsymbol{v}_s)\right) \tag{8.73}$$

where

$$\widetilde{\boldsymbol{b}}_{s,l} = \pi\left(b_l^m(\boldsymbol{v}_s), b(\boldsymbol{v}_{s+1}^S)\right) \tag{8.74}$$

has a total of $W = nm - (s-1)Nm - N(l-1)$ bits, namely the $(m-l+1)N$ bits of bit-levels $l, \ldots, m$ of SIC stage s and the $(S-s)mN$ bits of SIC stages $s+1, \ldots, S$. The permutation π sorts the bits in the order they are transmitted over the channel, and this reflects our choice of sampling order; see (8.76). Empirically, this ordering results in faster convergence than other orderings.

To simplify notation, assume that s, l are fixed and replace $\widetilde{\boldsymbol{b}}_{s,l}$ with $\widetilde{\boldsymbol{b}}$. We further assume the conditional values are fixed and define

$$\Phi(\widetilde{\boldsymbol{b}}) := P(\widetilde{\boldsymbol{b}} \,\middle|\, \boldsymbol{y}, \hat{\boldsymbol{v}}^{s-1}, \hat{b}_1^{l-1}(\boldsymbol{v}_s)). \tag{8.75}$$

At iteration $i = 0$, we initialize with a realization $\widetilde{\boldsymbol{b}}(0)$ of u.i.i.d. bits. At iteration i, $i \geq 1$, the sampler successively draws realizations from the distributions

$$\widetilde{b}_w(i) \sim \Phi(\widetilde{b}_w | \widetilde{b}^{w-1}(i), \widetilde{b}_{w+1}^W(i-1)) \tag{8.76}$$

for $w = 1, \ldots, W$. Observe that we use samples from iteration i for $\widetilde{b}^{w-1}(i)$ and samples from iteration $i-1$ for $\widetilde{b}_{w+1}^W(i-1)$ when sampling $\widetilde{b}_w(i)$. This sampling structure is the reason for choosing the permutation π in (8.74). The samples of $\widetilde{b}^{w-1}(i), \widetilde{b}_{w+1}^W(i-1)$ are ordered as the bits affect the channel. The algorithm stops after N_{iter} iterations. Note that we sample from the marginal distributions of $\Phi(\widetilde{\boldsymbol{b}})$ in (8.76). Nevertheless, for $N_{\text{iter}} \to \infty$ the distribution of $\widetilde{\boldsymbol{b}}(N_{\text{iter}})$ tends to the joint density $\Phi(\widetilde{\boldsymbol{b}})$ [109, Ch. 29.5].

In the following, we show how to compute the sampling probabilities (8.76). We start with an example illustrated in Fig. 8.7 since a general description would require many index transformations resulting in complex notations. We consider $n = 12$, $m = 3$ bit-levels and $S = 2$ SIC stages and a channel $\boldsymbol{\psi} \in \mathbb{C}^K$ of length $K = 2\widetilde{K} + 1 = 5$. The first SIC stage and the first bit-level of the second SIC stage are already decoded (highlighted

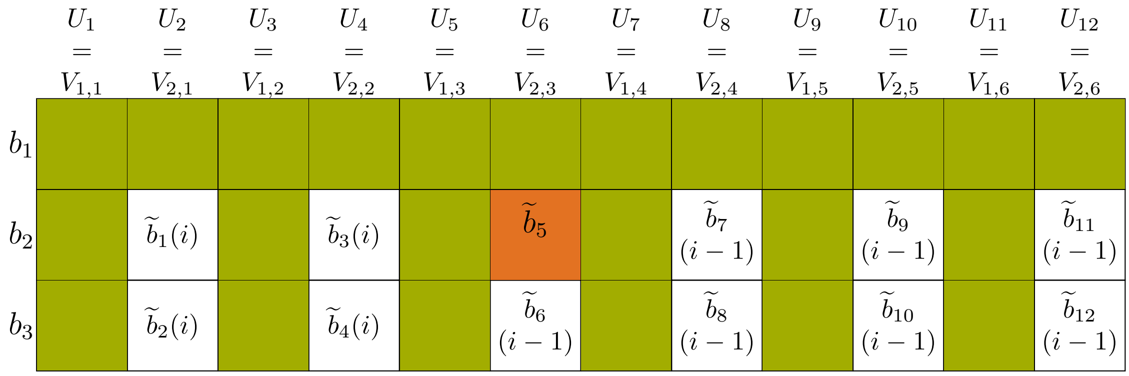


Figure 8.7.: Gibbs sampling example.

in green). Currently, the second bit-level of the second SIC stage is sampled. We are in iteration i and within this iteration the bits $b_2^3(V_{2,1})$ and $b_2^3(V_{2,2})$ are already sampled ((i) in Fig. 8.7). The samples marked with $(i-1)$ are taken from the last iteration. We now sample the bit $b_2(V_{2,3}) = \widetilde{b}_5$.

We have the vector $\widetilde{\boldsymbol{b}}$ of length $W = 12$ with all bits that are not yet decoded with the bit ordering depicted in Fig. 8.7. We draw a sample from the distribution

$$\Phi(\widetilde{b}_5|\widetilde{b}^4(i), \widetilde{b}_6^{12}(i-1)) \tag{8.77}$$

$$\stackrel{(a)}{=} P\left(\widetilde{b}_5|\boldsymbol{y}, \hat{\boldsymbol{v}}_1, b_1(\boldsymbol{v}_2), \widetilde{b}_1^4(i), \widetilde{b}_6^{12}(i-1)\right) \tag{8.78}$$

$$\stackrel{(b)}{=} \frac{1}{A} p\left(\boldsymbol{y}|\hat{\boldsymbol{v}}_1, b_1(\boldsymbol{v}_2), \widetilde{b}_1^4(i), \widetilde{b}_5, \widetilde{b}_6^{12}(i-1)\right) \tag{8.79}$$

$$\stackrel{(c)}{=} \frac{1}{A} p\left(\boldsymbol{y}|\hat{u}_1, \tilde{u}_2(i), \hat{u}_3, \tilde{u}_4(i), \hat{u}_5, \tilde{u}_6(\widetilde{b}_5), \hat{u}_7, \tilde{u}_8(i-1), \dots\right) \tag{8.80}$$

$$\stackrel{(d)}{=} \frac{1}{A} p\left(\boldsymbol{y}|x_1(i), x_2(i), \dots, x_5(i), \tilde{x}_6(\widetilde{b}_5), e^{j\varphi(\widetilde{b}_5)}x_7(i-1), e^{j\varphi(\widetilde{b}_5)}x_8(i-1), \dots,\right) \tag{8.81}$$

We add conditions that are neglected in the definition of $\Phi(\widetilde{\boldsymbol{b}})$ in (8.75) in (a). We use Bayes' rule in (b) where A is a normalization constant that is independent of the value of $\widetilde{b}_5$. In (c), we transform the bits or symbols into the corresponding u_κ, where $\hat{u}_\kappa$ are already fixed from the last SIC stage and do not change their value in the Gibbs sampling process. $\tilde{u}_\kappa(i)$ and $\tilde{u}_\kappa(i-1)$ are defined from the i-th or $(i-1)$-th iteration and still undergo a sampling process, but their values are the same for $\widetilde{b}_5 = 0, 1$ in the current sampling step. The value of $\tilde{u}_6(\widetilde{b}_5)$ has different values for $\widetilde{b}_5 = 0, 1$. Differential encoding is applied in (d) where $x_\kappa(i)$ and $x_\kappa(i-1)$ denote the value of the transmitted symbol in the i-th and $(i-1)$-th iteration. We assume the initial state is known for applying differential encoding. Note that different values of $\widetilde{b}_5$ might change the phase of $\tilde{u}_6(\widetilde{b}_5)$. This phase change affects the phase of all subsequent transmitted symbols denoted by the phase change $e^{j\varphi(\widetilde{b}_5)}$, i.e., to compute the probabilities in (8.76) for $\widetilde{b}_5 = 0, 1$, it seems that we must consider different values for all transmitted symbols that are transmitted after the symbol that is currently sampled, resulting in massive complexity.

In the following, we return to a general notation by denoting the bit that is currently sampled by $\widetilde{b}_w$ and the index of the symbol in the sequence u_1^n that corresponds to this bit as ν, i.e., changing the bit $\widetilde{b}_w$ also changes the value of u_ν. We show that we need not consider the full sequence of transmitted symbols to calculate (8.81) for $\widetilde{b}_w = 0, 1$.

We consider the quotient of the probabilities from (8.81) where $\widetilde{b}_w = 0$ and $\widetilde{b}_w = 1$ in the nominator and denominator, respectively. We have

$$\frac{\Phi(0|\widetilde{b}^{w-1}(i), \widetilde{b}_{w+1}^W(i-1))}{\Phi(1|\widetilde{b}^{w-1}(i), \widetilde{b}_{w+1}^W(i-1))} = \frac{p\left(\boldsymbol{y}|x_1^{\nu-1}(i), \tilde{x}_\nu(\widetilde{b}_w = 0), e^{j\varphi_0} x_{\nu+1}^n(i-1)\right)}{p\left(\boldsymbol{y}|x_1^{\nu-1}(i), \tilde{x}_\nu(\widetilde{b}_w = 1), e^{j\varphi_1} x_{\nu+1}^n(i-1)\right)} \tag{8.82}$$

$$\overset{(a)}{=} \frac{p\left(\boldsymbol{y}|\tilde{x}_1^0, \tilde{x}_2^0, \ldots, \tilde{x}_n^0\right)}{p\left(\boldsymbol{y}|\tilde{x}_1^1, \tilde{x}_2^1, \ldots, \tilde{x}_n^1\right)} \tag{8.83}$$

$$\overset{(b)}{=} \prod_{\kappa=1}^{n} \frac{p_N\left(y_\kappa^\diamondsuit|\tilde{x}_1^0, \tilde{x}_2^0, \ldots, \tilde{x}_n^0\right) p_N\left(y_\kappa^\spadesuit|\tilde{x}_1^0, \tilde{x}_2^0, \ldots, \tilde{x}_n^0\right)}{p_N\left(y_\kappa^\diamondsuit|\tilde{x}_1^1, \tilde{x}_2^1, \ldots, \tilde{x}_n^1\right) p_N\left(y_\kappa^\spadesuit|\tilde{x}_1^1, \tilde{x}_2^1, \ldots, \tilde{x}_n^1\right)} \tag{8.84}$$

where we use a new notation in (a) to denote the sequence of transmitted symbols in case of $\widetilde{b}_w = 0, 1$ and use that the noise samples are independent in (b). Similar to (8.57) and (8.58) in the FBA, the noise densities in (8.84) can be written as

$$p_N\left(y_\kappa^\diamondsuit|x_1^{\widetilde{b}_w}, \ldots, x_n^{\widetilde{b}_w}\right) = p_N\left(y_\kappa^\diamondsuit - \left|\left[x^{\widetilde{b}_w}_{\kappa-\widetilde{K}/2}, 0, \ldots, x_\kappa^{\widetilde{b}_2}, \ldots, 0, x^{\widetilde{b}_b}_{\kappa+\widetilde{K}/2}\right] \cdot \boldsymbol{\psi}^{\mathrm{T}}\right|^2\right) \tag{8.85}$$

$$p_N\left(y_\kappa^\spadesuit|x_1^{\widetilde{b}_w}, \ldots, x_n^{\widetilde{b}_w}\right) = p_N\left(y_\kappa^\spadesuit - \left|\left[0, x^{\widetilde{b}_w}_{\kappa-\widetilde{K}/2+1}, 0, \ldots, x_\kappa^{\widetilde{b}_w}, \ldots, x^{\widetilde{b}_w}_{\kappa+\widetilde{K}/2}, 0\right] \cdot \boldsymbol{\psi}^{\mathrm{T}}\right|^2\right). \tag{8.86}$$

We can distinguish three cases for the index κ. For $\kappa < \nu - \widetilde{K}/2$, we have

$$\left[x^0_{\kappa-\widetilde{K}/2}, 0, \ldots, x_\kappa^0, \ldots, 0, x^0_{\kappa+\widetilde{K}/2}\right] = \left[x^1_{\kappa-\widetilde{K}/2}, 0, \ldots, x_\kappa^1, \ldots, 0, x^1_{\kappa+\widetilde{K}/2}\right] \tag{8.87}$$

$$\left[0, x^0_{\kappa-\widetilde{K}/2+1}, 0, \ldots, x_\kappa^0, \ldots, x^0_{\kappa+\widetilde{K}/2}, 0\right] = \left[0, x^1_{\kappa-\widetilde{K}/2+1}, 0, \ldots, x_\kappa^1, \ldots, x^1_{\kappa+\widetilde{K}/2}, 0\right], \tag{8.88}$$

i.e., terms in the nominator and denominator of (8.84) are the same and can be removed. For $\kappa > \nu + \widetilde{K}/2$, we have

$$\left[x^0_{\kappa-\widetilde{K}/2}, 0, \ldots, x_\kappa^0, \ldots, 0, x^0_{\kappa+\widetilde{K}/2}\right] = e^{j(\varphi_0 - \varphi_1)}\left[x^1_{\kappa-\widetilde{K}/2}, 0, \ldots, x_\kappa^1, \ldots, 0, x^1_{\kappa+\widetilde{K}/2}\right] \tag{8.89}$$

$$\left[0, x^0_{\kappa-\widetilde{K}/2+1}, \ldots, x_\kappa^0, \ldots, x^0_{\kappa+\widetilde{K}/2}, 0\right] = e^{j(\varphi_0 - \varphi_1)}\left[0, x^1_{\kappa-\widetilde{K}/2+1}, \ldots, x_\kappa^1, \ldots, x^1_{\kappa+\widetilde{K}/2}, 0\right]. \tag{8.90}$$

Since we take the absolute value in (8.85) and (8.86), the resulting density has the same value for both, $\widetilde{b}_w = 0$ and $\widetilde{b}_w = 1$ and the corresponding terms can be removed from (8.84).

To conclude, we have

$$
\begin{aligned}
&\Phi(\widetilde{b}_w|\widetilde{b}^{w-1}(i), \widetilde{b}_{w+1}^{W}(i-1)) = \\
&\quad \frac{1}{A_1} \prod_{\kappa=\nu-\widetilde{K}/2}^{\nu+\widetilde{K}/2} \left[p_N \left(y_\kappa^{\diamondsuit} - \left| \left[x_{\kappa-\widetilde{K}/2}^{\widetilde{b}_w}, 0, \ldots, x_\kappa^{\widetilde{b}_2}, \ldots, 0, x_{\kappa+\widetilde{K}/2}^{\widetilde{b}_b} \right] \cdot \boldsymbol{\psi}^{\mathrm{T}} \right|^2 \right) \cdot \right. \\
&\qquad\qquad \left. p_N \left(y_\kappa^{\spadesuit} - \left| \left[0, x_{\kappa-\widetilde{K}/2+1}^{\widetilde{b}_w}, 0, \ldots, x_\kappa^{\widetilde{b}_w}, \ldots, x_{\kappa+\widetilde{K}/2}^{\widetilde{b}_w}, 0 \right] \cdot \boldsymbol{\psi}^{\mathrm{T}} \right|^2 \right) \right]
\end{aligned}
\tag{8.91}
$$

where A_1 is a normalization constant chosen so that

$$
\Phi\left(\widetilde{b}_w = 0|\widetilde{b}^{w-1}(i), \widetilde{b}_{w+1}^{W}(i-1)\right) + \Phi\left(\widetilde{b}_w = 1|\widetilde{b}^{w-1}(i), \widetilde{b}_{w+1}^{W}(i-1)\right) = 1. \tag{8.92}
$$

8.6.2. APP Estimation and Stalling

The bit-wise APPs may be written as

$$
\Phi\left(\widetilde{b}_w\right) = \sum_{\widetilde{\boldsymbol{b}}_{[w]}} \Phi(\widetilde{\boldsymbol{b}}) = \mathbb{E}\left[\Phi(\widetilde{b}_w|\widetilde{\boldsymbol{B}}_{[w]})\right]. \tag{8.93}
$$

One can approximate the expectation from the sampling strings $\widetilde{\boldsymbol{b}}(0), \ldots, \widetilde{\boldsymbol{b}}(N_{\text{iter}})$ via empirical PMF estimation or importance sampling [71]. A simple estimate is

$$
\Phi\left(\widetilde{b}_w\right) \approx \frac{1}{N_{\text{iter}}} \sum_{i=1}^{N_{\text{iter}}} \Phi\left(\widetilde{b}_w|\widetilde{\boldsymbol{b}}_{[w]}(i)\right). \tag{8.94}
$$

The estimate (8.94) is accurate at low SNR, and decoding achieves near-MAP performance [71]. However, the estimate can be inaccurate at high SNR if the sampling stalls because the conditionals (8.76) are overconfident, e.g., due to poor initialization or a bad current state of the Gibbs sampler [71, 152–154]. For instance, consider $\Phi(\widetilde{b}_w = 0) \approx 1$ and $\Phi(\widetilde{b}_w = 1) \approx 0$ so the iterations in (8.76) are nearly deterministic. In this case, a large N_{iter} is required to obtain good APP estimates.

To address stalling, one may use several samplers in parallel that are initialized with different bit strings. This makes stalling less likely but increases computational complexity. Alternatively, one can reduce confidence by increasing the variance of the channel likelihood; see [152, Sec. III] and [71, 155, 156]. Consider sampling from a modified posterior

$$
\Omega(\widetilde{\boldsymbol{b}}) = \frac{1}{A_2} p(\boldsymbol{y}|\widetilde{\boldsymbol{b}}, b(\boldsymbol{v}^{\widehat{s-1}}), \hat{b}_1^{l-1}(\boldsymbol{v}_s))^{\frac{1}{\eta}} \tag{8.95}
$$

where $\eta \in [1, \infty)$ is a confidence parameter and A_2 is a normalization constant. For Gaussian channels, η is a multiplicative factor for the noise variance.

The conditionals of (8.95) for Gibbs sampling are

$$\Omega\Big(\widetilde{b}_w \big| \, \widetilde{b}^{w-1}(i), \widetilde{b}_{w+1}^{W}(i-1)\Big) = \frac{1}{A_3}\left(p\left(\boldsymbol{y}\big| \, \boldsymbol{x}(\widetilde{b}_w)\right)\right)^{\frac{1}{\eta}} \tag{8.96}$$

where A_3 is again a normalization constant. We may insert the modified conditionals (8.96) into (8.76). As $N_{\text{iter}} \to \infty$ the distribution of $\widetilde{\boldsymbol{b}}(N_{\text{iter}}), \widetilde{\boldsymbol{b}}(N_{\text{iter}+1}), \ldots$ tends to $\Omega(\widetilde{\boldsymbol{b}})$. We use importance sampling to estimate (8.93) as

$$\Phi(\widetilde{b}_w) \approx \frac{1}{N_{\text{iter}}} \sum_{i=1}^{N_{\text{iter}}} \underbrace{\frac{\Phi(\widetilde{\boldsymbol{b}}_{[w]}(i))}{\Omega(\widetilde{\boldsymbol{b}}_{[w]}(i))}}_{\text{Importance weights}} \Phi(\widetilde{b}_w | \widetilde{\boldsymbol{b}}_{[w]}(i)) \tag{8.97}$$

where the $\widetilde{\boldsymbol{b}}(i)$ are provided by the sampler and approximate $\Omega(\widetilde{\boldsymbol{b}})$ and not the true posterior PMF. The importance weights resolve the mismatch. For $\eta = 1$, (8.97) becomes (8.94). For $\eta > 1$, one can avoid stalling at high SNR and the estimate (8.97) is significantly better than (8.94) in general [152, Fig. 2]. Finally, one may optimize η for every SNR by a line search with the corresponding rate expression as the cost function.

8.7. Numerical Results

If not stated otherwise, we simulate blocks with $n \geq 20 \times 10^3$ transmit symbols over $L = 0\,\text{km}$ or $L = 30\,\text{km}$ of SSMF without attenuation (0 dB/km). The symbol rate is $B = 35\,\text{GBaud}$ in the C band at carrier wavelength 1550 nm. The group velocity dispersion is $\beta_2 = -2.168 \times 10^{-23}\,\text{s}^2/\text{km}$. The average transmit power is

$$P_{\text{tx}} = \frac{\mathbb{E}\left[\|X(t)\|^2\right]}{n \cdot T_{\text{s}}} \tag{8.98}$$

and the variance of the noise samples after filtering is $N_0 B = 1$ so we define $\text{SNR} = P_{\text{tx}}$; see [80, Sec. V]. The SE of FD-RC pulses is

$$\text{SE} = R/(1+\alpha) \quad [\text{bit/s/Hz}] \tag{8.99}$$

where R is the information rate in bpcu. We estimate R using $I_{q,\text{JDD}}$, $I_{q,\text{SIC}}$, and $I_{q,\text{b-SIC}}$, that we compute via Monte Carlo simulation. The channel state $\boldsymbol{s}_0'$ (cf. Sec. 8.4.3) is known at the receiver.

We use the M-ary modulation formats M-PAM, M-ASK and M-SQAM as defined in Sec. 2.2. We approximate the combined filter $\psi(t)$ with a discrete filter ψ_k with $K = N_{\text{os}} \cdot 101 + 1$ taps, resulting in a system memory of $\widetilde{K} = 101$. This choice has ψ_k containing 99.8% of the energy of $\psi(t)$.

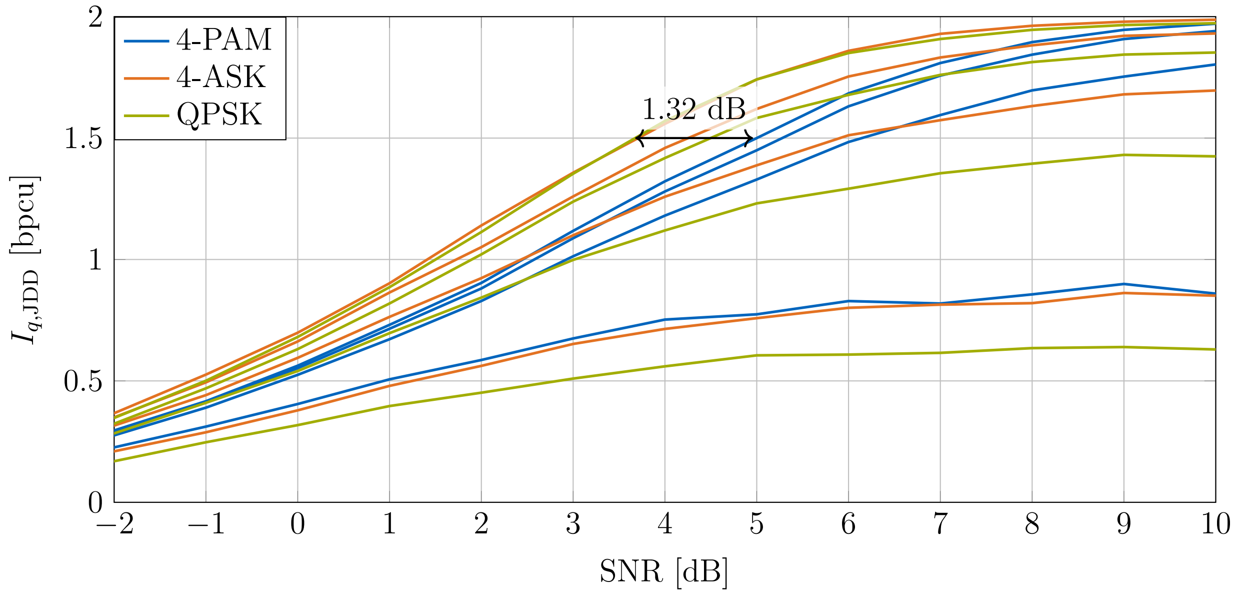


Figure 8.8.: Mismatched JDD rates $I_{q,\mathrm{JDD}}$ with $L = 30\,\mathrm{km}$, $\alpha = 0$, and $\widetilde{K}_q = [3, 5, 7, 9]$.

8.7.1. JDD Rates for Different Channel Sizes

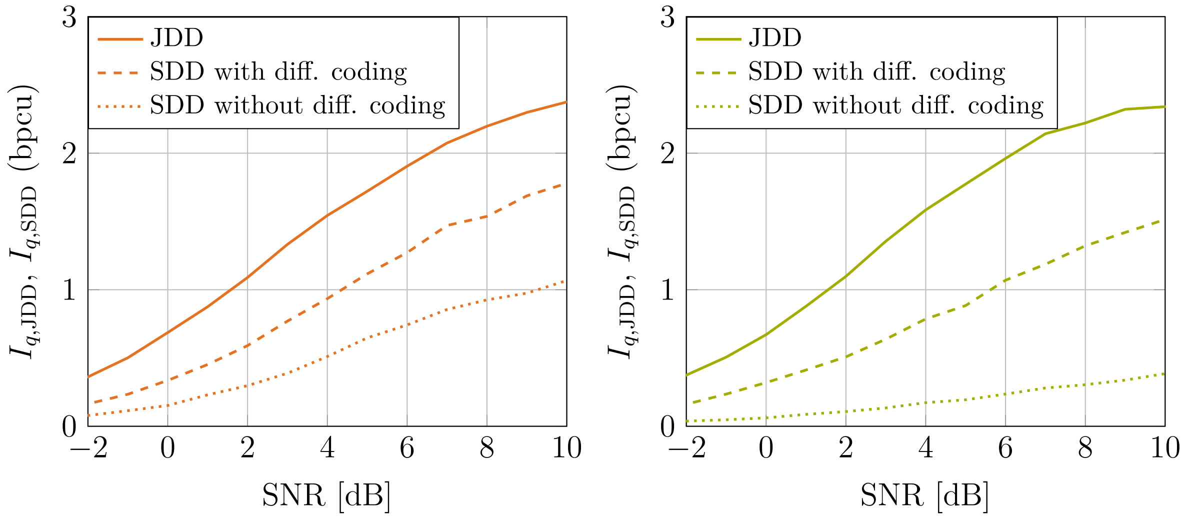
First, we compare JDD rates for different $\widetilde{K}_q$ of the auxiliary channel model in Fig. 8.8. We simulate a fiber of length 30 km and transmit sinc pulses ($\alpha = 0$). We make three observations. First, increasing $\widetilde{K}_q$ improves the mismatched rates $I_{q,\mathrm{JDD}}$ and should approach I_{JDD} for large $\widetilde{K}_q$. Second, if $\widetilde{K}_q$ is small, i.e., we have a large model mismatch, unipolar 4-PAM performs best since bipolar and complex constellations need a more accurate model to retrieve the phase of the transmitted symbols. Third, bipolar and complex constellations outperform unipolar PAM by around 1.32 dB at a rate of 1.5 bpcu. The distances between the different PAM curves suggest that PAM is already approaching I_{JDD} whereas it seems that the ASK and quadrature phase shift keying (QPSK) rates can be further increased by increasing $\widetilde{K}_q$.

8.7.2. SDD Rates with and without Differential Encoding

We introduced differential coding in Sec. 8.1 to avoid phase ambiguities. Phase ambiguities are also discussed in Appendix A.2. Fig. 8.9 shows SDD rates $I_{q,\mathrm{SDD}}$ with and without differential coding. We observe that differential coding is necessary for SDD.

8.7.3. SDD, SIC, JDD with FBA

Fig. 8.10 shows the rates $I_{q,\mathrm{JDD}}$ and $I_{q,\mathrm{SIC}}$ for $L = 30\,\mathrm{km}$ of fiber and with sinc pulses. Note that SDD corresponds to SIC with only $S = 1$ SIC stage. The rates increase with the number S of SIC stages. Observe that $\widetilde{K}_q = 9$ suffices to achieve nearly 2 bpcu for all



(a) 8-ASK

(b) 8-SQAM

Figure 8.9.: JDD rates $I_{q,\mathrm{JDD}}$ and SDD rates $I_{q,\mathrm{SDD}}$ with and without differential coding with $L = 30\,\mathrm{km}$, $\alpha = 0$, and $\widetilde{K}_q = 7$.

4-ary alphabets and large SNR; see also [80, Fig. 8a]. For $M = 4$, SDD achieves rates close to 2 bpcu for large SNR values and especially for PAM. The losses of SDD compared to JDD in the medium SNR range are relatively small for unipolar PAM but around 2 dB or more for bipolar and complex signaling.

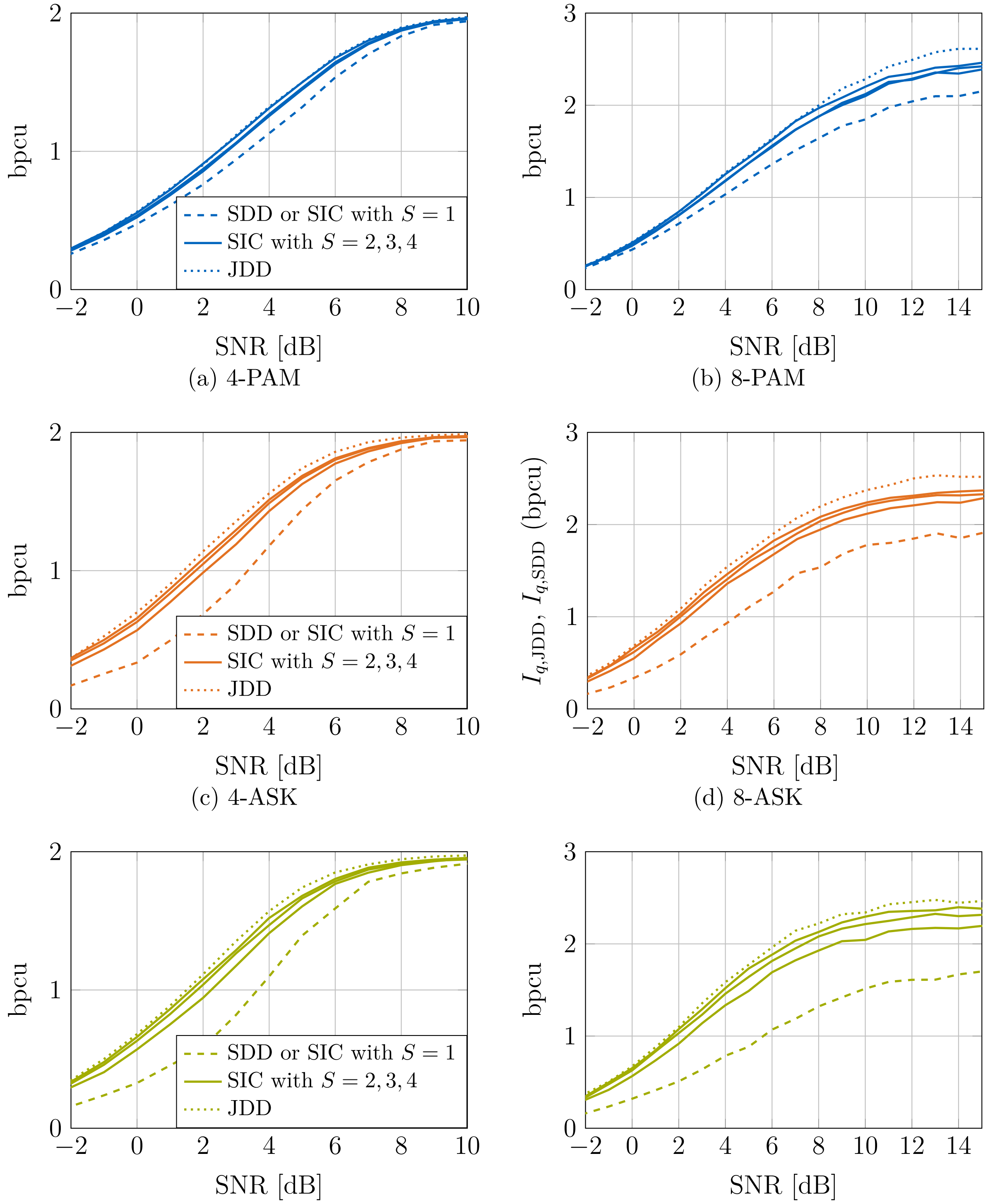
For $M = 8$ and $\widetilde{K}_q = 7$, the rates are limited by interference and saturate before 3 bpcu. The interference cannot be resolved due to the inexact auxiliary channel model. Increasing $\widetilde{K}_q$ would thus achieve higher rates. Likewise, SDD shows large losses for ASK and 8-SQAM for medium SNR due to the small auxiliary memory.

Fig. 8.10 shows that SIC can close the gap between JDD and SDD and approach the JDD rates as the number of SIC stages S grows. The results confirm the inequalities in (8.22). The performance with $S = 4$ agrees with JDD at low-to-medium SNR. For large S, the energy gains for ASK, PAM, and SQAM are similar to those in [80].

Fig. 8.11 shows the difference between $I_{q,\mathrm{SIC}}$ and $I_{q,\text{b-SIC}}$ as described by the inequality in (8.35). The rates are simulated using 4-ASK constellations with sinc pulses, $L = 30\,\mathrm{km}$ and $\widetilde{K}_q = 9$. The rates are computed with the FBA. The curves in Fig. 8.11 confirm the inequality in (8.34), i.e., the b-SIC with S SIC stages are higher than the SIC rates with S SIC stages. The difference is higher at $S = 1$ SIC stage and decreases as the number of SIC stages increases.

8.7.4. SIC Rate Approximation Using Gibbs Sampling

This section presents results for $I_{q,\text{b-SIC}}$ where the APPs are approximated by Gibbs sampling. The results are compared to $I_{q,\text{b-SIC}}$ where the APPs are computed via the FBA.



(a) 4-PAM (b) 8-PAM (c) 4-ASK (d) 8-ASK (e) QPSK (f) 8-SQAM

Figure 8.10.: Comparison of JDD, SIC and SDD ($I_{q,\mathrm{JDD}}$, $I_{q,\mathrm{SIC}}$, $I_{q,\mathrm{SDD}}$) rates for $S = 1, 2, 3, 4$ stages, $L = 30\,\mathrm{km}$, $\widetilde{K}_q = 9$ and $\widetilde{K}_q = 7$ for $M = 4$ and $M = 8$, respectively.

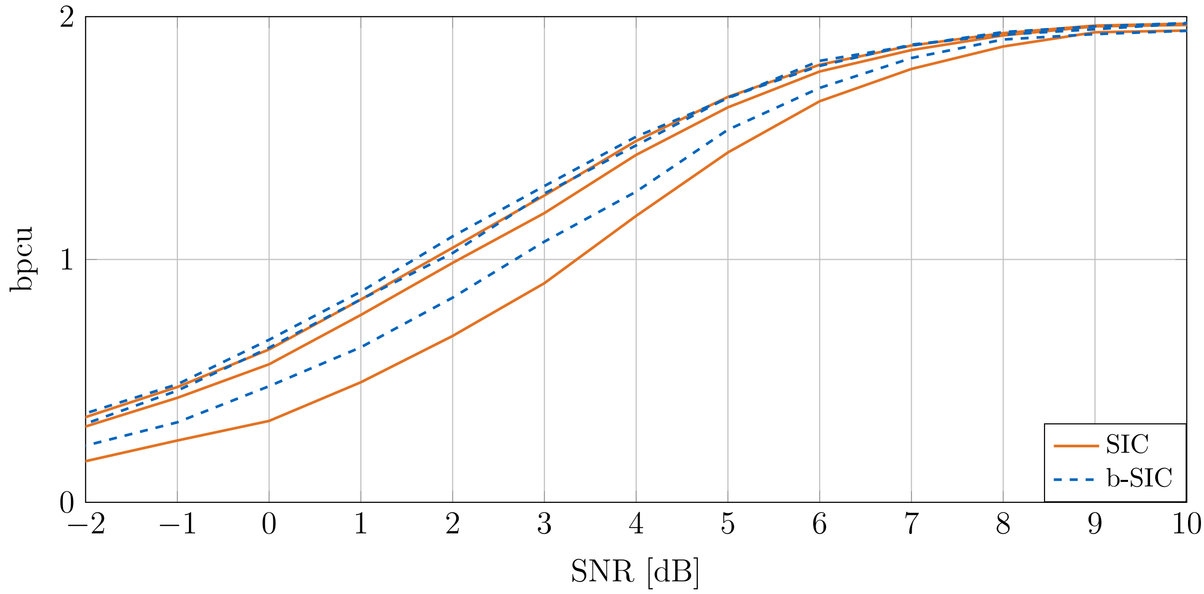


Figure 8.11.: Comparison of $I_{q,\text{SIC}}$ and $I_{q,\text{b-SIC}}$ for 4-ASK via FBA with sinc pulses, $L = 30\,\text{km}$, $\widetilde{K}_q = 9$, and $S = 1, 2, 3$.

To avoid stalling at high SNR, we use $N_{\text{par}} = 20$ parallel samplers and perform $N_{\text{iter}} = 60$ iterations per sampler. The first ten iterations per sampler are discarded (the so-called burn-in period). For every SNR, we use a line search with a training set of 10×10^3 symbols to optimize the confidence parameter η with respect to $I_{q,\text{b-SIC}}$. We use a binary reflected Gray code to label the elements in the alphabet $\mathcal{A}$.

Fig. 8.12 shows results for $L = 30\,\text{km}$ and sinc pulses and plots the SIC rates $I_{q,\text{b-SIC}}$ obtained via Gibbs sampling and FBA for 4-ASK and $S = 1, \ldots, 4$. The Gibbs sampler approximates the APPs. In terms of information theory this is another model mismatch. Thus, the rates obtained via Gibbs sampling are at most the FBA rates. This can be observed in Fig. 8.12 where the FBA rates (solid lines) are slightly higher than the Gibbs rates (dashed lines). At low SNR values, the rates obtained by Gibbs sampling are close to the FBA rates. At high SNR values, the Gibbs sampler suffers from stalling. As S is chosen large enough, the rates $I_{q,\text{b-SIC}}$ via Gibbs sampling can approach the JDD rates at low and medium SNR values, but not at high SNR. Remedies for this behavior include more iterations and parallel samplers.

Fig. 8.13 shows the b-SIC rates $I_{q,\text{b-SIC}}$ obtained by Gibbs sampling with $\widetilde{K}_q = 9$ for M-ary constellations with $M = 8, 16, 32$. We use FD-RC pulses with $\alpha = 0.2$. Complexity requirements, especially RAM limitations, prohibits using the FBA to create reference plots. ASK and PAM constellations achieve the maximum SE of $\log_2 M/(1 + \alpha)$ for $M = 8, 16$. For $M = 32$, both curves saturate due to ISI; this can be resolved using $\widetilde{K}_q > 9$ and possibly more iterations to avoid stalling. ASK gains up to 2.6 dB over PAM for $M = 32$. M-SQAM saturates at lower rates, likely because it is difficult to distinguish the phases. Two options to improve performance are increasing $\widetilde{K}_q$ and using modulations that avoid

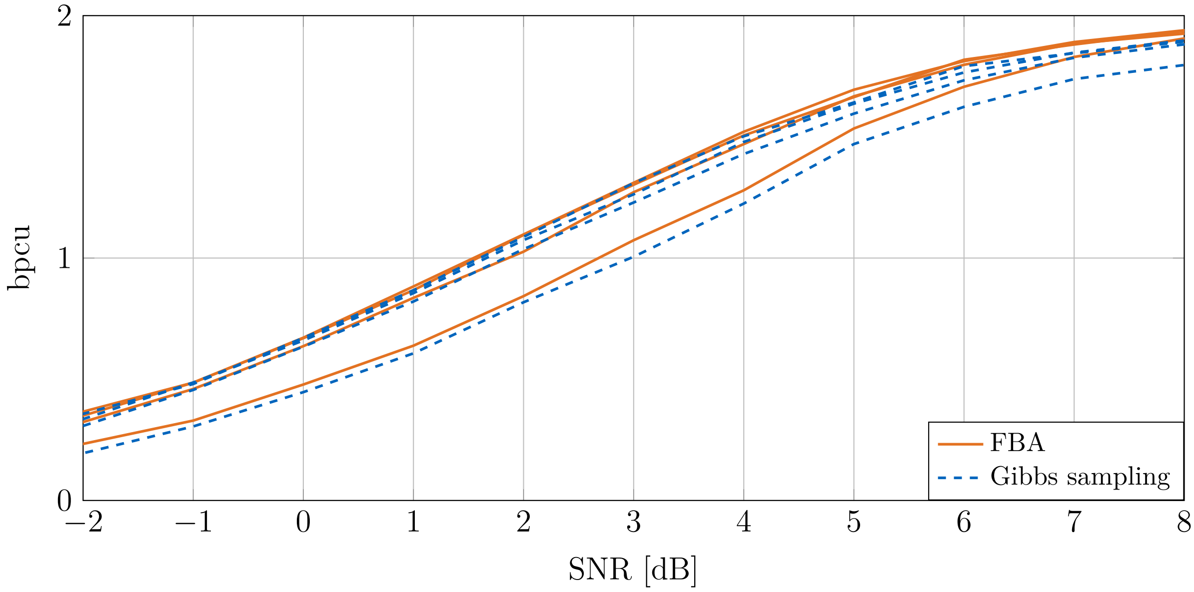


Figure 8.12.: Comparison of $I_{q,\text{b-SIC}}$ via Gibbs sampling and via FBA for 4-ASK with sinc pulses, $L = 30\,\text{km}$, and $\widetilde{K}_q = 9$.

phase ambiguities.

Fig. 8.14 shows b-SIC rates for $S = 4$, $\widetilde{K}_q = 9$, and 4-ASK when using Gibbs sampling with $N_{\text{iter}} = 1, 2, 5, 10, 20, 50$ iterations after the burn-in period. Observe that only a few iterations are needed to get good results in the low SNR regime. At high SNR values, more iterations are needed. We conclude that the number of iterations N_{iter} can be used to adopt the complexity of detector at the cost of lower rates.

8.7.5. Polar-Coded SDD and SIC with the FBA

Multilevel polar codes with MSD for memoryless channels are discussed in chapter 4. The multilevel structure of polar codes fits the SIC scenarios discussed in this chapter. We show that the rates presented in the previous sections can be achieved with polar codes. We consider the SIC-MSD scheme but polar codes may also be used with the b-SIC scheme or the SIC-BMD scheme.

Consider transmitting 4-ASK/PAM with $n = 1024$, $S = 1$ (SDD), and $S = 2, 4$. In general, we need Sm component codes, one for each SIC and bit-level. The length of the component codes is n/S bits. For our 4-ary example, this results in component codes of length 1024, 512 and 256 for $S = 1, 2, 4$, respectively. We use multistage list decoding as described in Sec. 4.3 with a maximum list size of eight and an outer 16-bit CRC code. The polar codes are designed using the Monte Carlo method described in Sec. 3.6.1, and the code rate of all component codes (S SIC levels, m bit levels) is simultaneously optimized. We simulate 10^6 frames that each have 1024 4-PAM/ASK symbols with FD-RC pulses with $\alpha = 0.2$ over a fiber of length 30 km. We use the FBA to compute the APPs needed

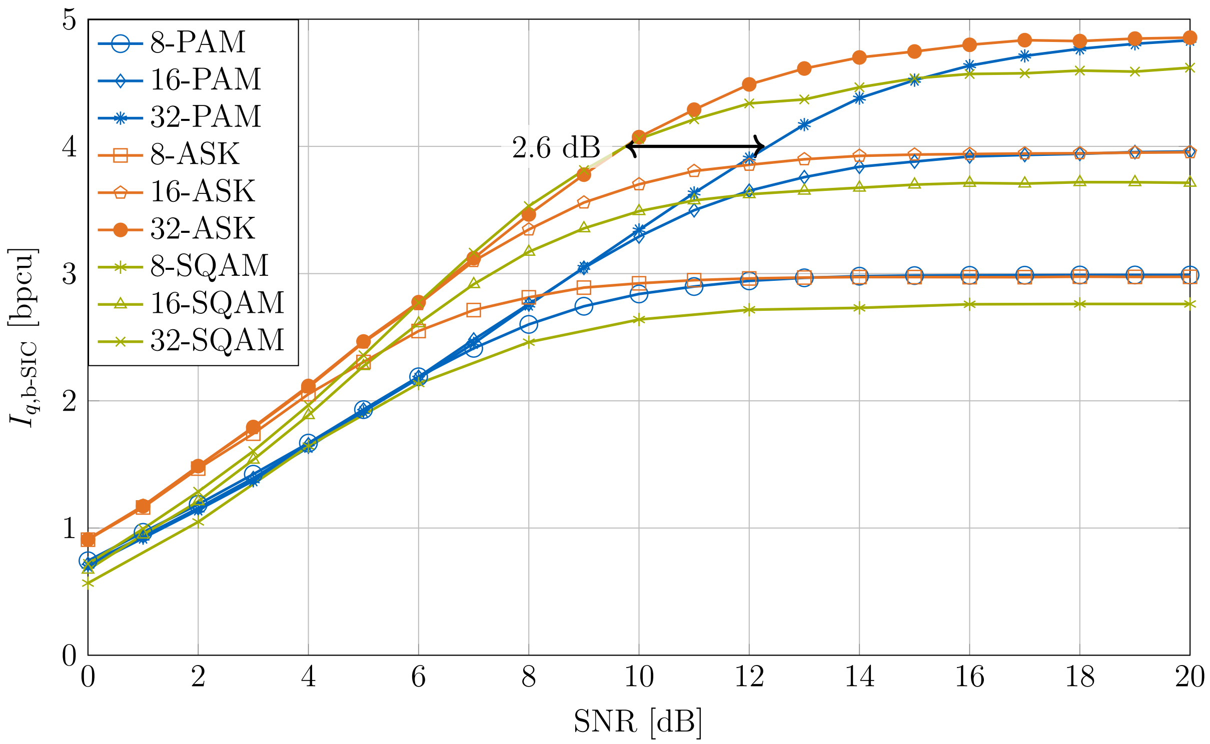


Figure 8.13.: b-SIC rates $I_{q,\text{b-SIC}}$ approximated by Gibbs sampling using FD-RC pulses with $\alpha = 0.2$, $L = 0\,\text{km}$, $\widetilde{K}_q = 9$, and $S = 4$.

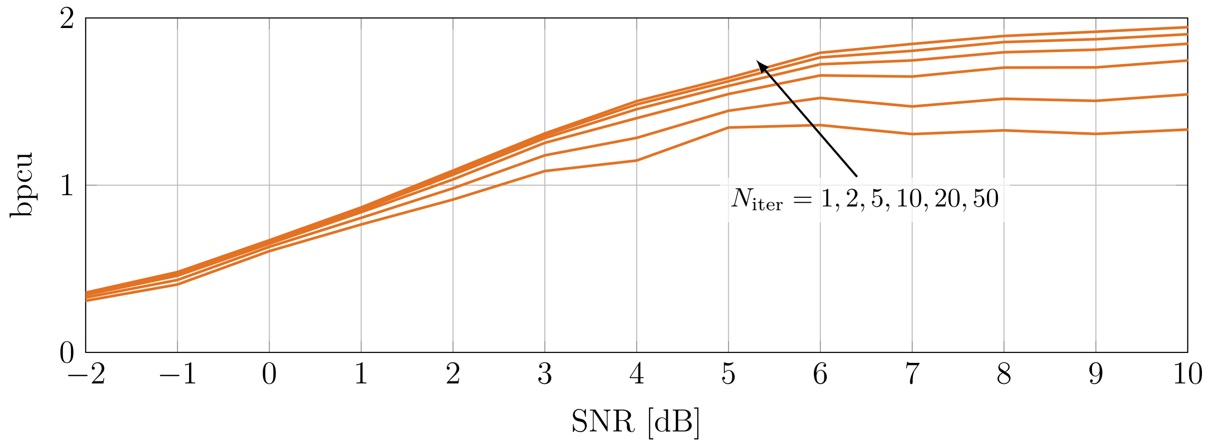
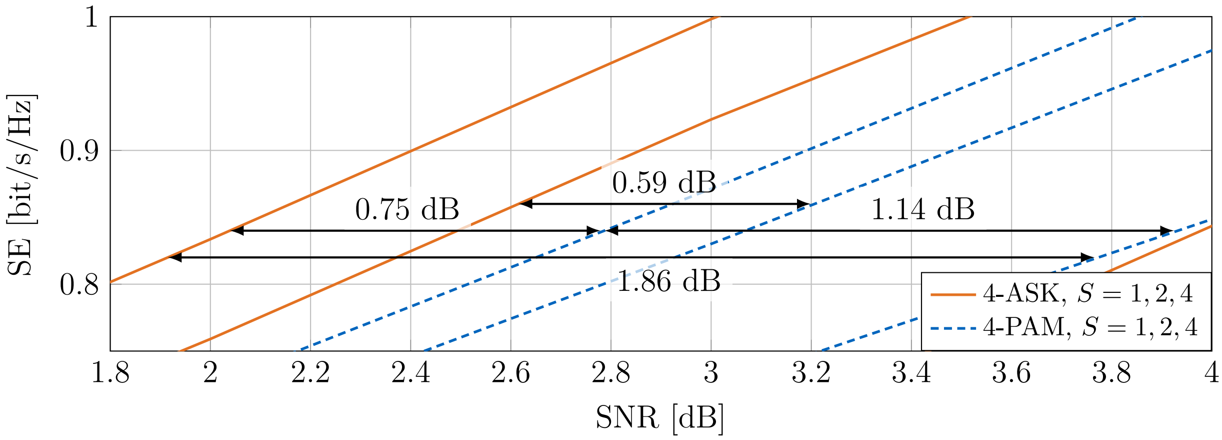
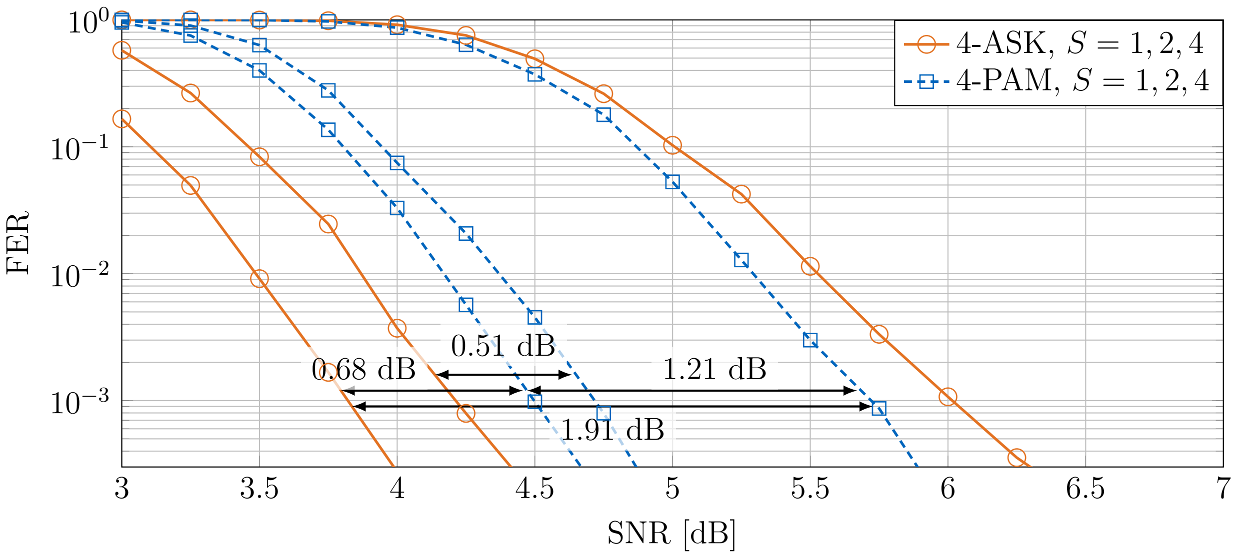


Figure 8.14.: SIC rates $I_{q,\text{b-SIC}}$ via Gibbs sampling vs. number of Gibbs sampler iterations for 4-ASK, $L = 30\,\text{km}$, $\alpha = 0$, $\widetilde{K}_q = 9$, $S = 4$ and $N_{\text{par}} = 20$ parallel samplers.



(a) Achievable rates around SE ≈ 0.83 bit/s/Hz.



(b) FER at a SE ≈ 0.83 bit/s/Hz for PCM.

Figure 8.15.: SIC rates and polar-coded FER for 4-ASK/PAM, $\widetilde{K}_q = 5$, FD-RC ($\alpha = 0.2$) and $L = 30$ km.

by the polar decoders. We use a mismatched model with $\widetilde{K}_q = 5$. We transmit at a rate of one bit per channel use resulting in a SE of 0.83 bit/s/Hz.

Fig. 8.15a shows $I_{q,\text{SIC-MSD}}$ around the target SE. For SDD, the 4-ASK and 4-PAM rates almost coincide. Using SIC with $S = 2, 4$, ASK gains approximately 0.59 dB and 0.75 dB in SNR over PAM, respectively, at 0.83 bit/s/Hz. Comparing SIC with $S = 4$ to SDD ($S = 1$), PAM and ASK gain 1.14 dB and 1.86 dB in SNR, respectively. Fig. 8.15b shows the FERs for which the SNR gains closely match those of Fig. 8.9a at FER 10^{-3}.

We remark that fiber-optic applications often require end-to-end FERs below 10^{-15}. One can achieve this by serially concatenating a soft-decision inner code, such as a LDPC or trellis code, with an interleaver and a powerful outer code to reduce the FER to the desired level, e.g., a Reed-Solomon code, a BCH code, or a staircase code; see [157–159]. Serial concatenation with a soft-decision inner code is also part of the 5G wireless standard, e.g., with a polar inner code and a CRC outer code as described above; see [160].

9 Conclusions

We extended PAS [40] to MLC and MSD to enable the combination of PCM [7] and PAS. We derived a coding scheme for polar codes based on dynamically frozen bits such that systematic encoding of polar codes meets the requirements of PAS. A multistage list decoding algorithm is proposed where lists of possible codewords are passed across decoding stages. Additional outer CRC codes [15] and a type check [125] in case of PAS with a CCDM [95] soften the multistage structure. The multilevel code structure can also be interpreted as one large code. Our PC-PAS schemes exhibit excellent FER performance for short block lengths and outperform state-of-the-art LDPC codes.

We presented results for BMERA coded modulation and for MLHY [29] based probabilistic shaping with BMERA codes instead of polar codes as in [30, 37]. We showed that BMERA codes for distribution matching result in lower rate losses than polar codes. Nevertheless, when used with list decoding and outer CRC-codes, the performance difference between polar and BMERA codes are small and come at the cost of increased complexity.

We showed that 6-PAM exhibits PSNR gains compared to 4-PAM and 8-PAM in certain SNR regions when transmitting over AWGN channels with a peak power constraint. We proposed using PAS to obtain 6-ary constellations. In addition, we suggested a suboptimal scheme based on 32-QAM constellations. The scheme is based on mapping the two-dimensional points of the 32-QAM constellation to 6 one-dimensional PAM symbols. We optimized the shape of the 32-QAM constellation and obtained a framed cross shaped constellation that outperforms classical 32-QAM as proposed in [55, 56]. Experiments on IM/DD links confirmed the gains.

We suggested several receiver architectures for DD receivers based on JDD, SDD, and SIC. The JDD rates show that one can transmit signals where the phase carries information like in ASK or QAM. The rates are higher than the rates obtained with traditional IM. Practical receivers based on SDD with oversampling cannot approach these rates. We suggested SIC with MLC and MSD. SIC approaches the JDD performance within only a few SIC stages. We proposed Gibbs sampling as a detector that approximates the metrics needed for SIC. Gibbs sampling reduces the complexity as compared to detectors like the FBA. Gibbs sampling also allows adapting the complexity by modifying the number of iterations. Simulations showed that only a few iterations are required at low SNR to perform close to the JDD rates. Simulations with PCM and FBA confirm the gains of ASK transmission as compared to PAM transmission in over-sampled DD systems.

A

Appendix

A.1. Decoding Metrics for BMERA codes

This section shows how the decoding metrics for BMERA codes can be computed. Fig. A.1 shows a cutout of the BMERA transform.

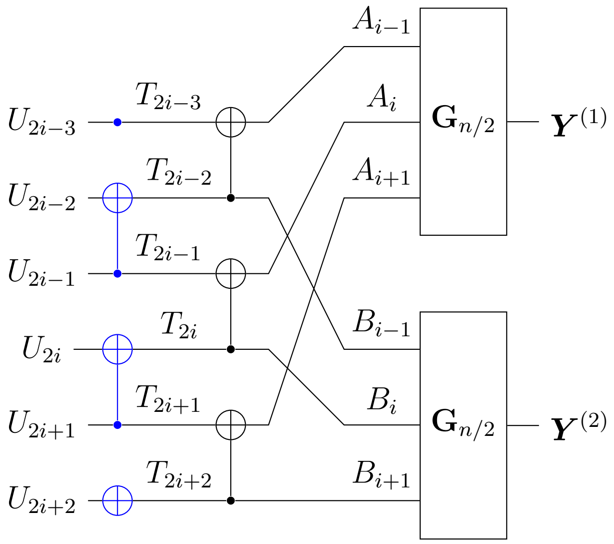


Figure A.1.: BMERA Decoding

A.1.1. Polar Decoding Recursion

We attempt to develop the same recursions as for polar decoding. Consider decoding U_{2i-2}, i.e., a bit with an even index. We assume that U_1^{2i-3} is known. From U_1^{2i-3}, we can compute T_1^{2i-3}, A_1^{i-2} and B_1^{i-2}, i.e., they are also known. We do not explicitly state in the following derivations that they are a function of U_1^{2i-3} and assume they are known. We derive the APP of U_{2i-2} as

$$P_{U_{2i-2}|\boldsymbol{Y}U_1^{2i-3}}\left(u_{2i-2}\middle|\boldsymbol{y},u_1^{2i-3}\right) \tag{A.1}$$

$$= \sum_{u_{2i-1}} P_{U_{2i-2}^{2i-1}|\boldsymbol{Y}U_1^{2i-3}}\left(u_{2i-2},u_{2i-1}\middle|\boldsymbol{y},u_1^{2i-3}\right) \tag{A.2}$$

$$= \frac{1}{c_1}\sum_{u_{2i-1}} P_{T_{2i-3}^{2i-1}|\boldsymbol{Y}T_1^{2i-4}}\left(u_{2i-3},u_{2i-2}\oplus u_{2i-1},u_{2i-1}\middle|\boldsymbol{y},t_1^{2i-4}\right) \tag{A.3}$$

$$= \frac{1}{c_1}\sum_{t_{2i}}\sum_{u_{2i-1}} P_{T_{2i-3}^{2i}|\boldsymbol{Y}T_1^{2i-4}}\left(u_{2i-3},u_{2i-2}\oplus u_{2i-1},u_{2i-1},t_{2i}\middle|\boldsymbol{y},t_1^{2i-4}\right) \tag{A.4}$$

$$= \frac{1}{c_1}\sum_{t_{2i}}\sum_{u_{2i-1}} P_{A_{i-1}^{i}|\boldsymbol{Y}^{(1)}A_1^{i-2}}\left(u_{2i-3}\oplus u_{2i-2}\oplus u_{2i-1},u_{2i-1}\oplus t_{2i}\middle|\boldsymbol{y}^{(1)},a_1^{i-2}\right) \tag{A.5}$$

$$P_{B_{i-1}^{i}|\boldsymbol{Y}^{(2)}B_1^{i-2}}\left(u_{2i-2}\oplus u_{2i-1},t_{2i}\middle|\boldsymbol{y}^{(2)},b_1^{i-2}\right). \tag{A.6}$$

For polar codes, we can recursively compute the APP of a bit U_{2i} from the APP of a single bit of the upper code and from the APP of a single bit of the lower code. In (A.6), we observe a different situation for BMERA codes. The APP of an even bit in $\boldsymbol{U}$ is a function of the APPs of two bits of the upper and lower code. This means we cannot follow the same recursion as for polar codes. The same can be shown for decoding of U_{2i-1}, i.e., bits with an odd index.

One can show that to compute the two-bit APPs in (A.6), we need 3-bit APPs from the next upper and lower code. Luckily, one can compute the three-bit APPs from other three-bit APPs, as we show next.

A.1.2. Recursion for Three-bit APPs

We show how 3-bit APPs can be computed from 3-bit APPs from the previous layer. We first consider the 3-bit APP of U_{2i-2}^{2i}, i.e., the triplet starts with an even index.

$$P_{U_{2i-2}^{2i}|\boldsymbol{Y}U_1^{2i-3}}\left(u_{2i-2}^{2i}\middle|\boldsymbol{y},u_1^{2i-3}\right) \tag{A.7}$$

$$= \sum_{u_{2i+1}} P_{U_{2i-2}^{2i+1}|\boldsymbol{Y}U_1^{2i-3}}\left(u_{2i-2}^{2i+1}\middle|\boldsymbol{y},u_1^{2i-3}\right) \tag{A.8}$$

$$= \frac{1}{c_1}\sum_{u_{2i+1}} P_{T_{2i-3}^{2i+1}|\boldsymbol{Y}T_1^{2i-4}}\left(u_{2i-3},u_{2i-2}\oplus u_{2i-1},u_{2i-1},u_{2i}\oplus u_{2i+1},u_{2i+1}\middle|\boldsymbol{y},u_1^{2i-4}\right) \tag{A.9}$$

$$= \frac{1}{c_1}\sum_{t_{2i+2}}\sum_{u_{2i+1}} P_{T_{2i-3}^{2i+2}|\boldsymbol{Y}T_1^{2i-4}}\left(u_{2i-3},u_{2i-2}\oplus u_{2i-1},u_{2i-1},u_{2i}\oplus u_{2i+1},u_{2i+1},t_{2i+2}\middle|\boldsymbol{y},u_1^{2i-4}\right) \tag{A.10}$$

$$= \frac{1}{c_1}\sum_{t_{2i+2}}\sum_{u_{2i+1}}\Bigg[P_{A_{i-1}^{i+1}|\boldsymbol{Y}^{(1)}A_1^{i-2}}\Big(u_{2i-3}\oplus u_{2i-2}\oplus u_{2i-1},u_{2i-1}\oplus u_{2i}\oplus u_{2i+1},$$

$$u_{2i+1} \oplus t_{2i+2} \Big| \boldsymbol{y}^{(1)}, a_1^{i-2} \Big)$$
$$P_{B_{i-1}^{i+1}|\boldsymbol{Y}^{(2)} B_1^{i-2}} \left(u_{2i-2} \oplus u_{2i-1}, u_{2i} \oplus u_{2i+1}, t_{2i+2} \Big| \boldsymbol{y}^{(2)}, b_1^{i-2} \right) \Bigg] \tag{A.11}$$

This equation if valid for $i \in \mathbb{Z} \wedge 2 \leq i \leq n/2 - 1$. The special case with $i = n/2$ is derived in section A.1.3.

Next, consider the 3-bit APPs of U_{2i-1}^{2i+1}, i.e., the triplet starts with an odd index:

$$P_{U_{2i-1}^{2i+1}|\boldsymbol{Y} U_1^{2i-2}} \left(u_{2i-1}^{2i+1} \Big| \boldsymbol{y}, u_1^{2i-2} \right) \tag{A.12}$$
$$= \frac{1}{c_1} P_{T_{2i-3}^{2i+1}|\boldsymbol{Y} T_1^{2i-4}} \left(u_{2i-3}, u_{2i-2} \oplus u_{2i-1}, u_{2i-1}, u_{2i} \oplus u_{2i+1}, u_{2i+1} \Big| \boldsymbol{y}, t_1^{2i-4} \right) \tag{A.13}$$
$$= \frac{1}{c_1} \sum_{t_{2i+2}} P_{T_{2i-3}^{2i+2}|\boldsymbol{Y} T_1^{2i-4}} \left(u_{2i-3}, u_{2i-2} \oplus u_{2i-1}, u_{2i-1}, u_{2i} \oplus u_{2i+1}, u_{2i+1}, t_{2i+2} \Big| \boldsymbol{y}, t_1^{2i-4} \right) \tag{A.14}$$
$$= \frac{1}{c_1} \sum_{t_{2i+2}} \Bigg[P_{A_{i-1}^{i+1}|\boldsymbol{Y}^{(1)} A_1^{i-2}} \Big(u_{2i-3} \oplus u_{2i-2} \oplus u_{2i-1}, u_{2i-1} \oplus u_{2i} \oplus u_{2i+1},$$
$$u_{2i+1} \oplus t_{2i+2} \Big| \boldsymbol{y}^{(1)}, a_1^{i-2} \Big)$$
$$P_{B_{i-1}^{i+1}|\boldsymbol{Y}^{(2)} B_1^{i-2}} \left(u_{2i-2} \oplus u_{2i-1}, u_{2i} \oplus u_{2i+1}, t_{2i+2} \Big| \boldsymbol{y}^{(2)}, b_1^{i-2} \right) \Bigg] \tag{A.15}$$

This equation is valid for $i \in \mathbb{Z} \wedge 2 \leq i \leq n/2 - 1$. The special case with $i = 1$ is discussed in section A.1.3.

A.1.3. 3-bit APPs at the borders of the transform

Consider (A.11) with $i = n/2$. We have $(U_{2i-2}, U_{2i-1}, U_{2i}) = (U_{n-2}, Un - 1, U_n)$, i.e., the last three bits in that layer. Fig. A.2 illustrates this scenario. In this case, we drop the two marginalizations from (A.8) and (A.10) and obtain

$$P_{U_{n-2}^{n}|\boldsymbol{Y} U_1^{n-3}} \left(u_{n-2}^{n} \Big| \boldsymbol{y}, u_1^{n-3} \right) \tag{A.16}$$
$$= \frac{1}{c} P_{T_{n-3}^{n}|\boldsymbol{Y} T_1^{n-4}} \left(u_{n-3}, u_{n-2} \oplus u_{n-1}, u_{n-1}, u_n \oplus u_1 \Big| \boldsymbol{y}, t_1^{n-4} \right) \tag{A.17}$$
$$= \frac{1}{c} P_{A_{n/2-1}^{n/2}|\boldsymbol{Y}^{(1)} A_1^{n/2-2}} \left(u_{n-3} \oplus u_{n-2} \oplus u_{n-1}, u_{n-1} \oplus u_n \oplus u_1 \Big| \boldsymbol{y}^{(1)}, a_1^{n/2-2} \right)$$
$$P_{B_{n/2-1}^{n/2}|\boldsymbol{Y}^{(2)} B_1^{n/2-2}} \left(u_{n-2} \oplus u_{n-1}, u_n \oplus u_1 \Big| \boldsymbol{y}^{(2)}, b_1^{n/2-2} \right). \tag{A.18}$$

Consider (A.15) with $i = 1$. We have $(U_{2i-1}, U_{2i}, U_{2i+1}) = (U_1, U_2, U_3)$, i.e., the first three bits in that layer. Fig. A.3 illustrates this scenario. Since there are no previous bits to

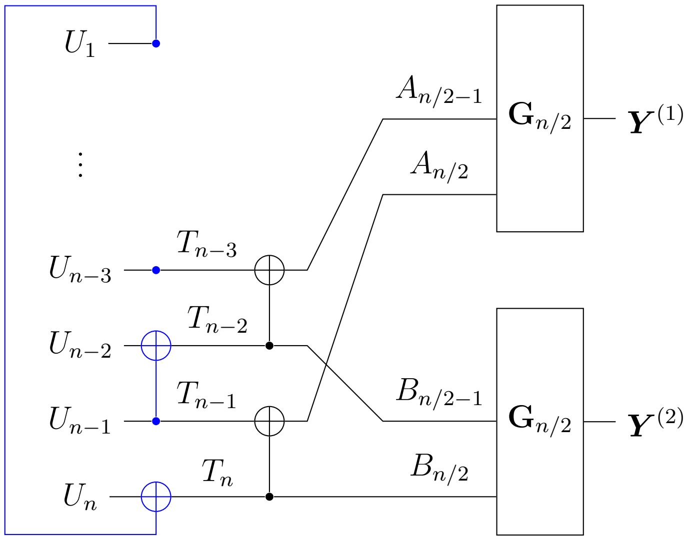


Figure A.2.: Cutout from the BMERA transform for the last bits

consider, we skip the application of Bayes' rule in (A.13) and obtain

$$P_{U_1^3|\boldsymbol{Y}}\left(u_1^3\middle|\boldsymbol{y}\right) \tag{A.19}$$

$$P_{T_1^3|\boldsymbol{Y}}\left(t_1^3\middle|\boldsymbol{y}\right) \tag{A.20}$$

$$= \sum_{t_4} P_{T_1^4|\boldsymbol{Y}}\left(u_1, u_2 \oplus u_3, u_3, t_4|\boldsymbol{y}\right) \tag{A.21}$$

$$= \sum_{t_4} P_{A_1^2|\boldsymbol{Y}^{(1)}}\left(u_1 \oplus u_2 \oplus u_2, u_3 \oplus t_4\middle|\boldsymbol{y}^{(1)}\right) P_{B_1^2|\boldsymbol{Y}^{(2)}}\left(u_2 \oplus u_2, t_4\middle|\boldsymbol{y}^{(2)}\right). \tag{A.22}$$

The 2-bit APPs needed for (A.18) and (A.22) can be obtained by marginalizing 3-bit APPs, except in the case where the size of the layer is only $n = 2$. We discuss this initialization case in the next section.

A.1.4. Initialization with $n = 2$

For the initialization with $n = 2$ at the beginning of the decoding algorithm, we have

$$P_{U_1U_2|\boldsymbol{Y}}\left(u_1, u_2|\boldsymbol{y}\right) = P_{C_1|Y_1}(u_2|y_1)P_{C_2|Y_2}(u_1 \oplus u_2|y_2) \tag{A.23}$$

as can be seen from Fig. A.4.

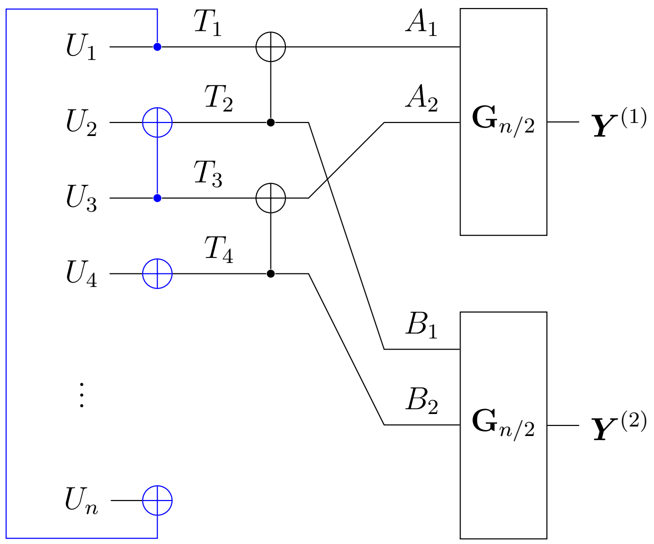


Figure A.3.: Cutout from the BMERA transform for the first bits

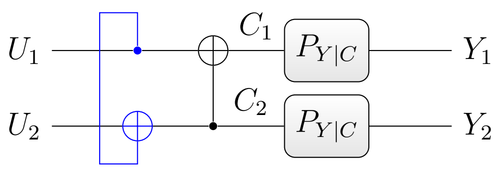


Figure A.4.: BMERA transform of size $n = 2$

A.2. Phase Ambiguities for DD channels

This appendix treats phase ambiguities for pulses symmetric about $t = 0$, such as FD-RC with CD, and symbol alphabets $\mathcal{A}$ with phase symmetries, such as M-ASK [80, Appendix]. A similar discussion can be found in [58, Sec. II]. Our main interest is high SNR, so consider the noise-free SLD output

$$y'(t) = \left| \sum_{\kappa} X_\kappa \, g_{\text{tx}}(t - \kappa T_s) \right|^2 . \tag{A.24}$$

At $t = 0$, we have

$$y'(0) = |X_0|^2 \cdot |g_{\text{tx}}(0)|^2 \tag{A.25}$$

and similarly for all even samples; at $t = T_s/2$ we have

$$\begin{aligned} y'(T_s/2) &= \left| \sum_{\kappa} X_\kappa \, g_{\text{tx}}(T_s/2 - \kappa T_s) \right|^2 \\ &= \left| \sum_{\kappa \geq 1} \left(X_\kappa + X_{1-\kappa} \right) \, g_{\text{tx}}(T_s/2 - \kappa T_s) \right|^2 \end{aligned} \tag{A.26}$$

and similarly for all odd samples. One loses the phase information in the even samples (A.25) and there is symmetric ISI in the odd samples. This symmetry can be problematic if $\mathcal{A}$ has a ± 1 phase symmetry, for example.

To illustrate, consider BPSK with $\mathcal{A} = \{\pm 1\}$ and a mismatched receiver that accounts for only the first pulse values $g_{\text{tx}}(\pm T_s/2)$ in (A.26). Suppose the initial and final states are x_0 and x_{n+1}, respectively. The resulting approximation of (A.26) is

$$y'(T_s/2) \approx 2 \left(1 + X_0 X_1\right) |g_{\text{tx}}(T_s/2)|^2 \tag{A.27}$$

and we obtain the string of normalized odd samples

$$\left\{ \frac{y'(\kappa T_s + T_s/2)}{2 |g_{\text{tx}}(T_s/2)|^2} - 1 \right\}_{\kappa=0}^{n+1} \approx x_0 X_1, X_1 X_2, \ldots, X_n x_{n+1}. \tag{A.28}$$

The expression (A.28) shows that the SLD operates similarly to a differential detector on the odd samples, which explains why differential coding is useful. In fact, without differential coding, SDD is useless if $x_0 = x_{n+1} = 0$ because $P_{X_\kappa|\boldsymbol{Y}}(a|\mathbf{y}) = 1/2$ for $a = \pm 1$. Moreover, SDD is hampered for large n if $x_0 \in \{\pm 1\}$ and $x_{n+1} \in \{\pm 1\}$, since SDD must first detect X_1 or X_n, and then successively the remaining symbols.

A.3. CRC-Polynomials

Table A.1 lists the CRC polynomials used in this thesis for different CRC sizes in bits. The notation of the polynomials is in implicit +1 notation, i.e., the corresponding polynomial for the 8 bit CRC is $x^8 + x^7 + x^6 + x^3 + x^2 + x^1 + 1$ with the +1 not explicitly written in `0xe7`.

CRC size (bits)	CRC polynomial
3	`0x5`
4	`0x9`
5	`0x12`
6	`0x33`
7	`0x65`
8	`0xe7`
9	`0x119`
10	`0x327`
11	`0x5db`
12	`0x987`
13	`0x1abf`
14	`0x27cf`
15	`0x4f23`
16	`0x8d95`
17	`0x16fa7`
18	`0x23979`
19	`0x6fb57`
20	`0xb5827`
21	`0x1707ea`
22	`0x308fd3`
23	`0x540df0`
24	`0x8f90e3`

Table A.1.: CRC polynomials from [126].

B

List of Acronyms

32-C-QAM cross shaped QAM-32
32-FC-QAM framed-cross shaped 32-QAM
APP *a posteriori* probabilities
ASK amplitude shift keying
AWG arbitrary waveform generator
AWGN additive white Gaussian noise
B-DMC binary-input discrete memoryless channel
b-SIC bit-wise successive interference cancellation
B2B back-to-back
BCJR Bahl-Cocke-Jelinek-Raviv
BCH Bose–Chaudhuri–Hocquenghem
BEC binary erasure channel
biAWGN binary-input additive white Gaussian noise
BICM bit-interleaved coded modulation
BMD bit-metric decoding
BMERA branching multi-scale entanglement renormalization ansatz
bpcu bits per channel use
BPSK binary phase shift keying
BRGC binary reflected Gray code
BW backward
CCDM constant composition distribution matcher
CD chromatic dispersion
CRC cyclic redundancy check
DD direct detection
DM distribution matcher
DMC discrete memoryless channel
DSO digital storage oscilloscope

FBA	forward-backward algorithm
FD-RC	frequency-domain raised-cosine
FEC	forward error correction
FER	frame error rate
FW	forward
HD	hard-decision decoding
i.i.d.	independent and identically distributed
IM	intensity modulation
ISI	intersymbol interference
JDD	joint detection and decoding
LDPC	low-density parity-check
LLR	log-likelihood ratio
MAP	maximum a posteriori probabilit
MB	Maxwell-Boltzmann
MI	mutual information
MLC	multilevel coding
MLHY	multilevel Honda-Yamamoto
MSD	multistage decoding
MZM	Mach-Zehnder modulator
NBC	natural binary code
OOK	on-off keying
P/S	parallel-to-serial
PAM	pulse amplitude modulation
PAS	probabilistic amplitude shaping
PC-PAS	polar coded PAS
PCM	polar coded modulation
PD	photodiode
PDF	probability density function
PMF	probability mass function
PSNR	peak signal-to-noise ratio
QAM	quadrature amplitude modulation
QPSK	quadrature phase shift keying
RV	random variable
S/P	serial-to-parallel
SC	successive cancellation
SCL	successive cancellation list
SD	soft decoding
SDD	separate detection and decoding
SE	spectral efficiency
SIC	successive interference cancellation
SLD	square-law device
SMD	symbol-metric decoding
SMDM	shell-mapping distribution matcher

SNR	signal-to-noise ratio
SPS	sample per symbol
SQAM	star-QAM
SSMF	standard single mode fiber
SU	symmetric-ultracomposite
TDD	turbo detection and decoding
u.i.i.d.	uniform, independent, and identically distributed
VOA	variable optical attenuator

Bibliography

[1] E. Arıkan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.

[2] 3GPP, "Multiplexing and channel coding," 3GPP TS 38.212, V15.0.0, Dec. 2017.

[3] H. Imai and S. Hirakawa, "A new multilevel coding method using error-correcting codes," *IEEE Trans. Inf. Theory*, vol. 23, no. 3, pp. 371–377, May 1977.

[4] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inf. Theory*, vol. 28, no. 1, pp. 55–67, Jan. 1982.

[5] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel codes: Theoretical concepts and practical design rules," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1361–1391, Jul. 1999.

[6] M. Seidl, A. Schenk, C. Stierstorfer, and J. B. Huber, "Multilevel polar-coded modulation," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2013, pp. 1302–1306.

[7] ——, "Polar-Coded Modulation," *IEEE Trans. Commun.*, vol. 61, no. 10, pp. 4108–4119, Oct. 2013.

[8] T. M. Cover, "Some Advances in Broadcast Channels," in *Advances in Communication Systems.* Elsevier, Jan. 1975, vol. 4, pp. 229–260.

[9] S. Verdú, *Multiuser Detection.* Cambridge University Press, Aug. 1998.

[10] T. Guess and M. K. Varanasi, "A new successively decodable coding technique for intersymbol-interference channels," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2000, p. 102.

[11] H. D. Pfister, J. B. Soriaga, and P. H. Siegel, "On the achievable information rates of finite state ISI channels," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Nov. 2001, pp. 2992–2996.

[12] J. B. Soriaga, H. D. Pfister, and P. H. Siegel, "Determining and Approaching Achievable Rates of Binary Intersymbol Interference Channels Using Multistage Decoding," *IEEE Trans. Inf. Theory*, vol. 53, no. 4, pp. 1416–1429, Apr. 2007.

[13] S. Geman and D. Geman, “Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 6, pp. 721–741, Nov. 1984.

[14] N. Stolte, “Rekursive Codes mit der Plotkin-Konstruktion und ihre Decodierung,” PhD thesis, Technische Universität Darmstadt, Jan. 2002.

[15] I. Tal and A. Vardy, “List Decoding of Polar Codes,” *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.

[16] M. Seidl, A. Schenk, C. Stierstorfer, and J. B. Huber, “Aspects of Polar-Coded Modulation,” in *Proc. Int. ITG Conf. Syst. Commun. Coding (SCC)*, Jan. 2013, pp. 1–6.

[17] H. Mahdavifar, M. El-Khamy, J. Lee, and I. Kang, “Compound polar codes,” in *2013 Information Theory and Applications Workshop (ITA)*, Feb. 2013, pp. 1–6.

[18] ——, “Polar Coding for Bit-Interleaved Coded Modulation,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 5, pp. 3115–3127, May 2016.

[19] S. B. Korada, E. Şaşoğlu, and R. Urbanke, “Polar Codes: Characterization of Exponent, Bounds, and Constructions,” *IEEE Trans. Inf. Theory*, vol. 56, no. 12, pp. 6253–6264, Dec. 2010.

[20] A. J. Ferris and D. Poulin, “Branching MERA codes: A natural extension of polar codes,” *arXiv:1312.4575*, Dec. 2013.

[21] ——, “Branching MERA codes: A natural extension of classical and quantum polar codes,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2014, pp. 1081–1085.

[22] A. J. Ferris, C. Hirche, and D. Poulin, “Convolutional Polar Codes,” *arXiv:1704.00715*, Apr. 2017.

[23] R. Morozov and P. Trifonov, “A Lower Bound on Minimum Distance of Convolutional Polar Codes,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2019, pp. 2104–2108.

[24] ——, “On Distance Properties of Convolutional Polar Codes,” *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 4585–4592, Jul. 2019.

[25] R. A. Morozov, “Convolutional Polar Kernels,” *IEEE Trans. Commun.*, vol. 68, no. 12, pp. 7352–7361, Dec. 2020.

[26] H. Saber, Y. Ge, R. Zhang, W. Shi, and W. Tong, “Convolutional Polar Codes: LLR-based Successive Cancellation Decoder and List Decoding Performance,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2018, pp. 1480–1484.

[27] R. Morozov and P. Trifonov, “Efficient SC Decoding of Convolutional Polar Codes,” in *Int. Symp. Inf. Theory and its Applicat. (ISITA)*, Oct. 2018, pp. 442–446.

[28] T. Prinz and P. Yuan, "Successive Cancellation List Decoding of BMERA Codes with Application to Higher-Order Modulation," in *Proc. IEEE Int. Symp. Turbo Codes Iter. Inf. Process. (ISTC)*, Dec. 2018, pp. 1–5.

[29] J. Honda and H. Yamamoto, "Polar Coding Without Alphabet Extension for Asymmetric Models," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 7829–7838, Dec. 2013.

[30] T. Wiegart, F. Steiner, P. Schulte, and P. Yuan, "Shaped On–Off Keying Using Polar Codes," *IEEE Commun. Lett.*, vol. 23, no. 11, pp. 1922–1926, Nov. 2019.

[31] D. Sutter, J. M. Renes, F. Dupuis, and R. Renner, "Achieving the capacity of any DMC using only polar codes," *Proc. IEEE Inf. Theory Workshop (ITW)*, pp. 114–118, Sep. 2012.

[32] O. İşcan and W. Xu, "Polar Codes with Integrated Probabilistic Shaping for 5G New Radio," in *Proc. IEEE Veh. Technol. Conf. Fall (VTC-Fall)*, Aug. 2018, pp. 1–5.

[33] O. İşcan, R. Böhnke, and W. Xu, "Probabilistically Shaped Multi-Level Coding with Polar Codes for Fading Channels," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–5.

[34] ——, "Shaped Polar Codes for Higher Order Modulation," *IEEE Commun. Lett.*, vol. 22, no. 2, pp. 252–255, Feb. 2018.

[35] T. Matsumine, T. Koike-Akino, D. S. Millar, K. Kojima, and K. Parsons, "Polar-Coded Modulation for Joint Channel Coding and Probabilistic Shaping," in *Proc. Optical Fiber Commun. Conf.*, Mar. 2019, pp. 1–3.

[36] R. Böhnke, O. İşcan, and W. Xu, "Multi-Level Distribution Matching," *IEEE Commun. Lett.*, vol. 24, no. 9, pp. 2015–2019, Sep. 2020.

[37] C. Runge, "Higher-Order Polar-Coded Modulation for Assymmetric Channels," Master's thesis, Technical University of Munich, Nov. 2021.

[38] R. G. Gallager, *Information Theory and Reliable Communication.* New York: John Wiley & Sons, 1968.

[39] G. Forney, "Trellis shaping," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 281–300, Mar. 1992.

[40] G. Böcherer, F. Steiner, and P. Schulte, "Bandwidth Efficient and Rate-Matched Low-Density Parity-Check Coded Modulation," *IEEE Trans. Commun.*, vol. 63, no. 12, pp. 4651–4665, Dec. 2015.

[41] F. Buchali, F. Steiner, G. Böcherer, L. Schmalen, P. Schulte, and W. Idler, "Rate Adaptation and Reach Increase by Probabilistically Shaped 64-QAM: An Experimental Demonstration," *J. Lightw. Technol.*, vol. 34, no. 7, pp. 1599–1609, Apr. 2016.

[42] M. Chagnon, "Optical Communications for Short Reach," *J. Lightw. Technol.*, vol. 37, no. 8, pp. 1779–1797, Apr. 2019.

[43] S. D. Dissanayake and J. Armstrong, "Comparison of ACO-OFDM, DCO-OFDM and ADO-OFDM in IM/DD Systems," *J. Lightw. Technol.*, vol. 31, no. 7, pp. 1063–1072, Apr. 2013.

[44] L. Chen, B. Krongold, and J. Evans, "Performance Analysis for Optical OFDM Transmission in Short-Range IM/DD Systems," *J. Lightw. Technol.*, vol. 30, no. 7, pp. 974–983, Apr. 2012.

[45] A. Mecozzi and M. Shtaif, "On the capacity of intensity modulated systems using optical amplifiers," *IEEE Photon. Technol. Lett.*, vol. 13, no. 9, pp. 1029–1031, Sep. 2001.

[46] K. Zhong, X. Zhou, J. Huo, C. Yu, C. Lu, and A. P. T. Lau, "Digital Signal Processing for Short-Reach Optical Communications: A Review of Current Technologies and Future Trends," *J. Lightw. Technol.*, vol. 36, no. 2, pp. 377–400, Jan. 2018.

[47] Q. Hu, M. Chagnon, K. Schuh, F. Buchali, and H. Bülow, "IM/DD Beyond Bandwidth Limitation for Data Center Optical Interconnects," *J. Lightw. Technol.*, vol. 37, no. 19, pp. 4940–4946, Oct. 2019.

[48] T. Wettlin, S. Calabrò, T. Rahman, J. Wei, N. Stojanović, and S. Pachnicke, "DSP for High-Speed Short-Reach IM/DD Systems Using PAM," *J. Lightw. Technol.*, vol. 38, no. 24, pp. 6771–6778, Dec. 2020.

[49] B. Karanov, M. Chagnon, F. Thouin, T. A. Eriksson, H. Bülow, D. Lavery, P. Bayvel, and L. Schmalen, "End-to-End Deep Learning of Optical Fiber Communications," *J. Lightw. Technol.*, vol. 36, no. 20, pp. 4843–4855, Oct. 2018.

[50] D. Plabst, F. J. G. Gómez, T. Wiegart, and N. Hanik, "Wiener Filter for Short-Reach Fiber-Optic Links," *IEEE Commun. Lett.*, vol. 24, no. 11, pp. 2546–2550, Nov. 2020.

[51] T. Wiegart, F. Da Ros, M. P. Yankov, F. Steiner, S. Gaiarin, and R. D. Wesel, "Probabilistically Shaped 4-PAM for Short-Reach IM/DD Links With a Peak Power Constraint," *J. Lightw. Technol.*, vol. 39, no. 2, pp. 400–405, Jan. 2021.

[52] M. S.-B. Hossain, G. Böcherer, T. Rahman, T. Wettlin, N. Stojanović, S. Calabrò, and S. Pachnicke, "Probabilistic Shaping for High-Speed Unamplified IM/DD Systems With an O-Band EML," *J. Lightw. Technol.*, vol. 41, no. 16, pp. 5373–5382, Aug. 2023.

[53] M. S.-B. Hossain, G. Böcherer, T. Rahman, N. Stojanović, P. Schulte, S. Calabrò, J. Wei, C. Bluemm, T. Wettlin, C. Xie, M. Kuschnerov, and S. Pachnicke, "Experimental Comparison of Cap and Cup Probabilistically Shaped PAM for O-Band

IM/DD Transmission System," in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2021, pp. 1–4.

[54] J. G. Smith, "The information capacity of amplitude- and variance-constrained sclar gaussian channels," *Information and Control*, vol. 18, no. 3, pp. 203–219, Apr. 1971.

[55] A. Ghiasi, Z. Wang, V. Telang, and B. Welch, "Investigation of PAM-4/6/8 signaling and FEC for 100 Gb/s serial transmission," IEEE 802.3bm Task Force, San Antonio, 2012.

[56] L. Chorchos, "PAM-6 generation using 32-QAM constellation," in *Metro and Data Center Optical Networks and Short-Reach Links II*, vol. 10946. International Society for Optics and Photonics, Feb. 2019.

[57] A. Mecozzi and M. Shtaif, "Information Capacity of Direct Detection Optical Transmission Systems," *J. Lightw. Technol.*, vol. 36, no. 3, pp. 689–694, Feb. 2018.

[58] M. Secondini and E. Forestieri, "Direct Detection of Bipolar Pulse Amplitude Modulation," *J. Lightw. Technol.*, vol. 38, no. 21, pp. 5981–5990, Nov. 2020.

[59] A. Tasbihi and F. R. Kschischang, "On the Capacity of Waveform Channels Under Square-Law Detection of Time-Limited Signals," *IEEE Trans. Inf. Theory*, vol. 66, no. 11, pp. 6682–6687, Nov. 2020.

[60] ——, "Direct Detection Under Tukey Signalling," *J. Lightw. Technol.*, vol. 39, no. 21, pp. 6845–6857, Nov. 2021.

[61] A. Mecozzi, C. Antonelli, and M. Shtaif, "Kramers–Kronig coherent receiver," *Optica*, vol. 3, no. 11, pp. 1220–1227, Nov. 2016.

[62] T. Bo and H. Kim, "Toward Practical Kramers-Kronig Receiver: Resampling, Performance, and Implementation," *J. Lightw. Technol.*, vol. 37, no. 2, pp. 461–469, Jan. 2019.

[63] Z. Li, M. Erkilinç, K. Shi, E. Sillekens, L. Galdino, B. Thomsen, P. Bayvel, and R. Killey, "Joint Optimisation of Resampling Rate and Carrier-to-Signal Power Ratio in Direct-Detection Kramers-Kronig Receivers," in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2017, pp. 1–3.

[64] D. M. Arnold, H.-A. Loeliger, P. O. Vontobel, A. Kavcic, and W. Zeng, "Simulation-Based Computation of Information Rates for Channels With Memory," *IEEE Trans. Inf. Theory*, vol. 52, no. 8, pp. 3498–3508, Aug. 2006.

[65] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[66] V. Buchoux, O. Cappe, and E. Moulines, "Turbo multiuser detection for coded DS-CDMA systems: A Gibbs sampling approach," in *Asilomar Conf. Signals, Syst., Comput.*, vol. 2, Oct. 2000, pp. 1426–1430.

[67] X. Wang and R. Chen, "Adaptive Bayesian multiuser detection for synchronous CDMA with Gaussian and impulsive noise," *IEEE Trans. Signal Process.*, vol. 48, no. 7, pp. 2013–2028, Jul. 2000.

[68] T. M. Schmidl, A. Gatherer, X. Wang, and R. Chen, "Interference cancellation using the Gibbs sampler," in *Proc. IEEE Veh. Technol. Conf. Fall (VTC-Fall)*, vol. 1, Sep. 2000, pp. 429–433.

[69] Z. Shi, H. Zhu, and B. Farhang-Boroujeny, "Markov chain Monte Carlo techniques in iterative detectors: A novel approach based on Monte Carlo integration," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, vol. 1, Nov. 2004, pp. 325–329 Vol.1.

[70] Z. Yang and X. Wang, "Turbo equalization for GMSK signaling over multipath channels based on the Gibbs sampler," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 9, pp. 1753–1763, Sep. 2001.

[71] B. Farhang-Boroujeny, H. Zhu, and Z. Shi, "Markov chain Monte Carlo algorithms for CDMA and MIMO communication systems," *IEEE Trans. Signal Process.*, vol. 54, no. 5, pp. 1896–1909, May 2006.

[72] R.-H. Peng, R.-R. Chen, and B. Farhang-Boroujeny, "Low Complexity Markov Chain Monte Carlo Detector for Channels with Intersymbol Interference," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2009, pp. 1–5.

[73] F. M. Kashif, H. Wymeersch, and M. Z. Win, "Monte carlo equalization for nonlinear dispersive satellite channels," *IEEE J. Sel. Topics Signal Process.*, vol. 26, no. 2, pp. 245–255, Feb. 2008.

[74] H. Zhu, B. Farhang-Boroujeny, and R.-R. Chen, "On performance of sphere decoding and Markov chain Monte Carlo detection methods," *IEEE Signal Process. Lett.*, vol. 12, no. 10, pp. 669–672, Oct. 2005.

[75] X. Mao, P. Amini, and B. Farhang-Boroujeny, "Markov Chain Monte Carlo MIMO Detection Methods for High Signal-to-Noise Ratio Regimes," in *IEEE Global Commun. Conf. (GLOBECOM)*, Nov. 2007, pp. 3979–3983.

[76] T. Prinz, P. Yuan, G. Böcherer, F. Steiner, O. İşcan, R. Böhnke, and W. Xu, "Polar coded probabilistic amplitude shaping for short packets," in *Proc. IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Jul. 2017, pp. 1–5.

[77] G. Böcherer, T. Prinz, P. Yuan, and F. Steiner, "Efficient Polar Code Construction for Higher-Order Modulation," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC) Workshop*, Mar. 2017, pp. 1–6.

[78] T. Prinz, T. Wiegart, D. Plabst, S. Calabrò, G. Böcherer, N. Stojanović, and T. Rahman, "PAM-6 Coded Modulation for IM/DD Channels with a Peak-Power Constraint," in *Proc. IEEE Int. Symp. Topics Coding (ISTC)*, Aug. 2021, pp. 1–5.

[79] T. Prinz, T. Wiegart, D. Plabst, T. Rahman, M. S.-B. Hossain, N. Stojanović, S. Calabrò, N. Hanik, and G. Kramer, "Comparison of PAM-6 Modulations for Short-Reach Fiber-Optic Links with Intensity Modulation and Direct Detection," in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2022, pp. 1–4.

[80] D. Plabst, T. Prinz, T. Wiegart, T. Rahman, N. Stojanović, S. Calabrò, N. Hanik, and G. Kramer, "Achievable Rates for Short-Reach Fiber-Optic Channels With Direct Detection," *J. Lightw. Technol.*, vol. 40, no. 12, pp. 3602–3613, Jun. 2022.

[81] T. Prinz, D. Plabst, T. Wiegart, S. Calabrò, N. Hanik, and G. Kramer, "Successive Interference Cancellation for Bandlimited Channels With Direct Detection," *IEEE Trans. Commun.*, vol. 72, no. 3, pp. 1330–1340, Mar. 2024.

[82] C. Runge, T. Wiegart, D. Lentner, and T. Prinz, "Multilevel Binary Polar-Coded Modulation Achieving the Capacity of Asymmetric Channels," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2022, pp. 2595–2600.

[83] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[84] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* New Jersey: John Wiley & Sons, Inc., 2006.

[85] R. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Inf. Theory*, vol. 18, no. 4, pp. 460–473, Jul. 1972.

[86] S. Arimoto, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 14–20, Jan. 1972.

[87] F. R. Kschischang and S. Pasupathy, "Optimal nonuniform signaling for Gaussian channels," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 913–929, May 1993.

[88] A. Martinez, A. Guillen i Fabregas, G. Caire, and F. M. J. Willems, "Bit-Interleaved Coded Modulation Revisited: A Mismatched Decoding Perspective," *IEEE Trans. Inf. Theory*, vol. 55, no. 6, pp. 2756–2765, Jun. 2009.

[89] E. Zehavi, "8-PSK trellis codes for a Rayleigh channel," *IEEE Trans. Commun.*, vol. 40, no. 5, pp. 873–884, May 1992.

[90] G. Frank, "Pulse code communication," US Patent 2,632,058, Mar., 1953.

[91] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.

[92] A. Alvarado, A. Graell i Amat, F. Brännström, and E. Agrell, "On the equivalence of TCM encoders," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2012, pp. 2401–2405.

[93] E. Arıkan, "Source polarization," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2010, pp. 899–903.

[94] L. Liu, Y. Yan, C. Ling, and X. Wu, "Construction of Capacity-Achieving Lattice Codes: Polar Lattices," *IEEE Trans. Commun.*, vol. 67, no. 2, pp. 915–928, Feb. 2019.

[95] P. Schulte and G. Böcherer, "Constant Composition Distribution Matching," *IEEE Trans. Inf. Theory*, vol. 62, no. 1, pp. 430–434, Jan. 2016.

[96] P. Schulte and F. Steiner, "Divergence-Optimal Fixed-to-Fixed Length Distribution Matching With Shell Mapping," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 620–623, Apr. 2019.

[97] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2008, pp. 1173–1177.

[98] G. Evenbly and G. Vidal, "A class of highly entangled many-body states that can be efficiently simulated," *Phys. Rev. Lett.*, vol. 112, no. 24, p. 240502, Jun. 2014.

[99] M. Alsan and E. Telatar, "A Simple Proof of Polarization and Polarization for Non-Stationary Memoryless Channels," *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 4873–4878, Sep. 2016.

[100] E. Arıkan, "Systematic Polar Coding," *IEEE Commun. Lett.*, vol. 15, no. 8, pp. 860–862, Aug. 2011.

[101] G. T. Chen, Z. Zhang, C. Zhong, and L. Zhang, "A Low Complexity Encoding Algorithm for Systematic Polar Codes," *IEEE Commun. Lett.*, vol. 20, no. 7, pp. 1277–1280, Jul. 2016.

[102] G. Sarkis, I. Tal, P. Giard, A. Vardy, C. Thibeault, and W. J. Gross, "Flexible and Low-Complexity Encoding and Decoding of Systematic Polar Codes," *IEEE Trans. Commun.*, vol. 64, no. 7, pp. 2732–2745, Jul. 2016.

[103] H. Vangala, Y. Hong, and E. Viterbo, "Efficient Algorithms for Systematic Polar Encoding," *IEEE Commun. Lett.*, vol. 20, no. 1, pp. 17–20, Jan. 2016.

[104] A. Balatsoukas-Stimming, M. B. Parizi, and A. Burg, "LLR-Based Successive Cancellation List Decoding of Polar Codes," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5165–5179, Oct. 2015.

[105] C. Leroux, I. Tal, A. Vardy, and W. J. Gross, "Hardware architectures for successive cancellation decoding of polar codes," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, May 2011, pp. 1665–1668.

[106] E. Arıkan and E. Telatar, "On the rate of channel polarization," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2009, pp. 1493–1495.

[107] R. Morozov, "Efficient List Decoding of Convolutional Polar Codes," *arXiv:2007.05811*, Jul. 2020.

[108] P. Yuan, T. Prinz, G. Böcherer, O. İşcan, R. Böhnke, and W. Xu, "Polar Code Construction for List Decoding," in *Proc. Int. ITG Conf. Syst. Commun. Coding (SCC)*, Feb. 2019, pp. 1–6.

[109] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms.* Cambridge University Press, Sep. 2003.

[110] R. Mori and T. Tanaka, "Performance and construction of polar codes on symmetric binary-input memoryless channels," in *IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2009, pp. 1496–1500.

[111] P. Trifonov, "Efficient Design and Decoding of Polar Codes," *IEEE Trans. Commun.*, vol. 60, no. 11, pp. 3221–3227, Nov. 2012.

[112] H. Vangala, E. Viterbo, and Y. Hong, "A Comparative Study of Polar Code Constructions for the AWGN Channel," *arXiv:1501.02473*, Jan. 2015.

[113] D. Dosio, "Polar codes for error correction: Analysis and decoding algorithms," Master's thesis, University of Bologna, 2016.

[114] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 670–678, Apr. 2004.

[115] S.-Y. Chung, T. J. Richardson, and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 657–670, Feb. 2001.

[116] F. Brannstrom, L. K. Rasmussen, and A. J. Grant, "Convergence analysis and optimal scheduling for multiple concatenated codes," *IEEE Trans. Inf. Theory*, vol. 51, no. 9, pp. 3354–3364, Sep. 2005.

[117] F. Steiner, G. Böcherer, and G. Liva, "Protograph-Based LDPC Code Design for Shaped Bit-Metric Decoding," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 2, pp. 397–407, Feb. 2016.

[118] H. Afşer, N. Tirpan, H. Deliç, and M. Koca, “Bit-interleaved polar-coded modulation,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2014, pp. 480–484.

[119] P. Chen, M. Xu, B. Bai, and X. Ma, “Design of polar coded 64-QAM,” in *Proc. IEEE Int. Symp. Turbo Codes Iter. Inf. Process. (ISTC)*, Sep. 2016, pp. 251–255.

[120] K. Chen, K. Niu, and J. Lin, “Polar coded modulation with optimal constellation labeling,” in *National Doctoral Academic Forum on Information and Communications Technology 2013*, Aug. 2013, pp. 1–5.

[121] C. I. Ionita, M. Mansour, J. C. Roh, and S. Hosur, “On the design of binary polar codes for high-order modulation,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2014, pp. 3507–3512.

[122] S. R. Tavildar, “Bit-Permuted Coded Modulation for Polar Codes,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC) Workshop*, Mar. 2017, pp. 1–6.

[123] L. Karakchieva and P. Trifonov, “Joint list multistage decoding with sphere detection for polar coded SCMA systems,” in *Int. ITG Conf. Syst. Commun. Coding (SCC)*, Feb. 2019, pp. 1–6.

[124] P. Trifonov and V. Miloslavskaya, “Polar codes with dynamic frozen symbols and their decoding by directed search,” in *Proc. IEEE Inf. Theory Workshop (ITW)*, Sep. 2013, pp. 1–5.

[125] P. Yuan, G. Böcherer, P. Schulte, G. Kramer, R. Böhnke, and W. Xu, “Error detection using symbol distribution in a system with distribution matching and probabilistic amplitude shaping,” US Patent US10 880 037B2, Dec., 2020.

[126] P. Koopman, “Best CRC Polynomials,” https://users.ece.cmu.edu/~koopman/crc/.

[127] O. İşcan, R. Böhnke, and W. Xu, “Probabilistic Shaping Using 5G New Radio Polar Codes,” *IEEE Access*, vol. 7, pp. 22 579–22 587, 2019.

[128] ——, “Sign-bit shaping using polar codes,” *Trans. Emerg. Telecommun. Tech.*, vol. 31, no. 10, p. e4058, Jul. 2020.

[129] G. Böcherer, P. Schulte, and F. Steiner, “Probabilistic Shaping and Forward Error Correction for Fiber-Optic Communication Systems,” *J. Lightw. Technol.*, vol. 37, no. 2, pp. 230–244, Jan. 2019.

[130] G. Böcherer, “Principles of Coded Modulation,” Habilitation Thesis, Technical University of Munich, Munich, 2018.

[131] A. Sheikh, A. G. i Amat, and G. Liva, “Achievable Information Rates for Coded Modulation With Hard Decision Decoding for Coherent Fiber-Optic Systems,” *J. Lightw. Technol.*, vol. 35, no. 23, pp. 5069–5078, Dec. 2017.

[132] R. D. Wesel, X. Liu, J. M. Cioffi, and C. Komninakis, “Constellation labeling for linear encoders,” *IEEE Trans. Inf. Theory*, vol. 47, no. 6, pp. 2417–2431, Sep. 2001.

[133] F. Buchali, Q. Hu, M. Chagnon, K. Schuh, L. Schmalen, and S. Makovejs, “Flexible Transmission Enabled by Novel M2-QAM Formats with Record Distance - Spectral Efficiency Tuneability,” in *Proc. Optical Fiber Commun. Conf.*, Mar. 2018, pp. 1–3.

[134] A. Hocquenghem, “Codes correcteurs d’erreurs,” *Chiffers*, vol. 2, pp. 147–156, 1959.

[135] R. C. Bose and D. K. Ray-Chaudhuri, “On a class of error correcting binary group codes,” *Information and Control*, vol. 3, no. 1, pp. 68–79, Mar. 1960.

[136] D. Plabst, T. Prinz, F. Diedolo, T. Wiegart, G. Böcherer, N. Hanik, and G. Kramer, “Neural Network-Based Successive Interference Cancellation for Non-Linear Bandlimited Channels,” *IEEE Trans. Commun.*, vol. 73, no. 3, pp. 1847–1861, Mar. 2025.

[137] G. P. Agrawal, *Fiber-Optic Communication Systems.* John Wiley & Sons, Feb. 2012.

[138] R.-J. Essiambre, G. Kramer, P. J. Winzer, G. J. Foschini, and B. Goebel, “Capacity Limits of Optical Fiber Networks,” *J. Lightw. Technol.*, vol. 28, no. 4, pp. 662–701, Feb. 2010.

[139] R. G. Gallager, *Principles of Digital Communication.* Cambridge University Press Cambridge, UK, 2008.

[140] C. Douillard, M. Jézéquel, C. Berrou, D. Electronique, A. Picart, P. Didier, and A. Glavieux, “Iterative correction of intersymbol interference: Turbo-equalization,” *Eur. Trans. Telecommun.*, vol. 6, no. 5, pp. 507–511, 1995.

[141] X. Wang and H. V. Poor, “Iterative (turbo) soft interference cancellation and decoding for coded CDMA,” *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, Jul. 1999.

[142] G. Liga, A. Alvarado, E. Agrell, and P. Bayvel, “Information Rates of Next-Generation Long-Haul Optical Fiber Systems Using Coded Modulation,” *J. Lightw. Technol.*, vol. 35, no. 1, pp. 113–123, Jan. 2017.

[143] R. R. Muller and W. H. Gerstacker, “On the capacity loss due to separation of detection and decoding,” *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1769–1778, Aug. 2004.

[144] A. Kavcic, X. Ma, and M. Mitzenmacher, “Binary intersymbol interference channels: Gallager codes, density evolution, and code performance bounds,” *IEEE Trans. Inf. Theory*, vol. 49, no. 7, pp. 1636–1652, Jul. 2003.

[145] A. Feinstein, "A new basic theorem of information theory," *Transactions of the IRE Professional Group on Information Theory*, vol. 4, no. 4, pp. 2–22, 1954.

[146] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Channel Coding Rate in the Finite Blocklength Regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[147] D. Divsalar, "Performance of Mismatched Receivers on Bandlimited Channels." Ph.D. dissertation, University of California, Los Angeles, 1978.

[148] G. Kaplan and S. Shamai, "Information rates and error exponents of compound channels with application to antipodal signaling in a fading environment," *Archiv für Elektronik und Übertragungstechnik (AEU)*, vol. 47, no. 4, pp. 228–239, 1993.

[149] J. Scarlett, A. G. i Fàbregas, A. Somekh-Baruch, and A. Martinez, "Information-theoretic foundations of mismatched decoding," *Foundations and Trends® in Commun. Inf. Theory*, vol. 17, no. 2–3, pp. 149–401, 2020.

[150] G. Kramer, "Information rates for channels with fading, side information and adaptive codewords," *Entropy*, vol. 25, no. 5, p. 728, 2023.

[151] 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, Feb. 2001.

[152] M. Senst and G. Ascheid, "A Rao-Blackwellized Markov Chain Monte Carlo Algorithm for Efficient MIMO Detection," in *IEEE Int. Conf. Commun. (ICC)*, Jun. 2011, pp. 1–6.

[153] B. Hassibi, M. Hansen, A. G. Dimakis, H. A. J. Alshamary, and W. Xu, "Optimized Markov Chain Monte Carlo for Signal Detection in MIMO Systems: An Analysis of the Stationary Distribution and Mixing Time," *IEEE Trans. Signal Process.*, vol. 62, no. 17, pp. 4436–4450, Sep. 2014.

[154] J. C. Hedstrom, C. H. Yuen, R.-R. Chen, and B. Farhang-Boroujeny, "Achieving Near MAP Performance With an Excited Markov Chain Monte Carlo MIMO Detector," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 7718–7732, Dec. 2017.

[155] M. Hansen, B. Hassibi, A. G. Dimakis, and W. Xu, "Near-Optimal Detection in MIMO Systems Using Gibbs Sampling," in *IEEE Global Commun. Conf. (GLOBECOM)*, Nov. 2009, pp. 1–6.

[156] D. Auras, U. Deidersen, R. Leupers, and G. Ascheid, "VLSI design of a parallel MCMC-based MIMO detector with multiplier-free Gibbs samplers," in *2014 22nd International Conference on Very Large Scale Integration (VLSI-SoC)*, Oct. 2014, pp. 1–6.

[157] T. Mizuochi, Y. Konishi, Y. Miyata, T. Inoue, K. Onohara, S. Kametani, T. Sugihara, K. Kubo, H. Yoshida, T. Kobayashi, and T. Ichikawa, "Experimental Demonstration of Concatenated LDPC and RS Codes by FPGAs Emulation," *IEEE Photon. Technol. Lett.*, vol. 21, no. 18, pp. 1302–1304, Sep. 2009.

[158] M. Magarini, R.-J. Essiambre, B. E. Basch, A. Ashikhmin, G. Kramer, and A. J. de Lind van Wijngaarden, "Concatenated Coded Modulation for Optical Communications Systems," *IEEE Photon. Technol. Lett.*, vol. 22, no. 16, pp. 1244–1246, Aug. 2010.

[159] B. P. Smith, A. Farhood, A. Hunt, F. R. Kschischang, and J. Lodge, "Staircase Codes: FEC for 100 Gb/s OTN," *J. Lightw. Technol.*, vol. 30, no. 1, pp. 110–117, Jan. 2012.

[160] N. Kumar, D. Kedia, and G. Purohit, "A review of channel coding schemes in the 5G standard," *Telecommunication Systems*, vol. 83, no. 4, pp. 423–448, Aug. 2023.