TECHNISCHE UNIVERSITÄT MÜNCHEN

Lehrstuhl für Nachrichtentechnik

Coding for Higher-Order Modulation and Probabilistic Shaping

Fabian Steiner

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor–Ingenieurs

genehmigten Dissertation.

Vorsitzender:	Prof. Dr.-Ing. Wolfgang Kellerer
Prüfer der Dissertation:	1. Prof. Dr. sc. techn. Gerhard Kramer
	2. Prof. Richard Wesel, Ph.D.
	3. Prof. Rüdiger Urbanke, Ph.D

Die Dissertation wurde am 02.10.2019 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 06.12.2019 angenommen.

Acknowledgment

Many people have contributed to this thesis and I am thankful to all of them. Nevertheless, I would like to mention the support of some of them in more detail.

First, I would like to thank my doctoral advisor Prof. Gerhard Kramer for giving me the opportunity to conduct research at his chair and offering a work environment that was highly motivating. I really appreciate all the opportunities to attend conferences and spend research stays abroad.

I would also like to express my deepest gratitude to Georg Böcherer and Gianluigi Liva. I learnt so much from them in the past years and their advice and approach to scientific work had a huge impact on my work and thinking. I will always remember our trips to the DLR, discussing, drinking cappuccinos and espressos, and bargaining who is going to pay for them. Unfortunately, Gianluigi always won by exploiting his relation with the Italian barista. Working with both of them was a real pleasure and above all, we became close friends, which is even better.

In addition to that, my research benefitted a lot from my colleagues at LNT. I would like to thank Patrick Schulte for countless discussions on any topic related (and not related) to probabilistic shaping. Exchanging ideas, thoughts, and writing papers with you was always fun. Besides, I will definitely miss your cooking skills. You made sure that we never starved during lunch. I also want to thank my “coding office” mates, Tobias Prinz, Peihong Yuan and Thomas Wiegart. Working with you was great fun and I really appreciate that you guys did not complain about my desk, which was usually covered with huge piles of papers, books as well as empty coffee mugs.

Apart from my office mates and Patrick, there was also Lars Palzer who I liked to have a drink with or two, discuss about passive investment strategies and talk about the latest machine learning stuff. Special thanks to Andrei Nedelcu, who was able to keep my enthusiasm for coarsely quantized MIMO research going, even after leaving the field for a while. Finally, I want to thank Diego Lentner and Emna ben Yacoub. It was a real pleasure to work with you on parts of your Master Thesis.

Last but not least, I would like to thank my parents Rosmarie and Roland Steiner and my girlfriend Lisa Broska for their overwhelming support in every stage of this thesis. Lisa, I know, I owe you a lot of weekends and I will do my best to compensate for them in the years to come.

Munich, February 2020 Fabian Steiner

Contents

3. Probabilistic Shaping 39

Zusammenfassung

Informationstheoretische Werkzeuge werden genutzt, um ein Kommunikationssystem mit kodierter Modulation and Wahrscheinlichkeitsanpassung (PS) zu entwerfen, das nahe der theoretischen Shannon Grenze operiert.

Das Prinzip der Amplituden-Wahrscheinlichkeitsanpassung (PAS) wird für Kanäle mit einer symmetrischen kapazitätserreichenden Verteilung eingeführt. Es werden erreichbare Raten für Dekodiermetriken auf Symbol- und Bitebene analysiert und innerhalb einer geschichteteten Architektur bestehend aus einer Vorwärtsfehlerkorrektur- und Wahrscheinlichkeitsanpassungsschicht untersucht. Ferner wird eine Verteilungsanpassungskomponente eingeführt.

Eine geometrische Anpassung des Modulationsformats (GS) wird diskutiert und es wird ein Optimierungsalgorithmus auf der Basis von differentieller Evolution (differential evolution) vorgeschlagen, um geometrisch optimierte Konstellationen zu erhalten. PS und GS werden auf der Basis von erreichbaren Raten und Fehlerkurven mit kodierten Daten verglichen. Hierbei dient der ATSC 3.0 Standard als repräsentative Fallstudie. Die Ergebnisse lassen den Schluss zu, dass GS gegenüber PS für gleich große Konstellationen generell unterlegen ist, insbesondere wenn Dekodiermetriken auf Bitebene herangezogen werden.

Zudem werden PS Strategien für den Fall entwickelt, dass jedes Bitlevel unabhängig von den anderen in seiner Wahrscheinlichkeit angepasst wird. Dieser Ansatz wird als Produktverteilungsanpassung bezeichnet und wird zum Beispiel im Falle von parallelen Kanälen mit unterschiedlichen Modulationsformaten angewandt. Hierdurch können die Verteilungsanpassungskomponenten für die niedrigeren Bitlevel geteilt werden, was die Komplexität verringert. Ein Verfahren zur Wahrscheinlichkeitsanpassung auf der Basis von Zeitaufteilung wird für On-Off Keying gezeigt – hier kann PAS aufgrund der fehlenden Symmetrie in der Eingangsverteilung nicht angewandt werden.

Zudem werden Ansätze zum Entwickeln von binären und nicht-binären Low-density parity-check (LDPC) Codes hergeleitet, die es erlauben, optimierte Codes für verschiedenste Vorgaben und spektrale Effizienzen zu konstruieren. Die Zuordnung von Bitleveln zu den Variablenknoten des LDPC Codes werden für Dekodiermetriken auf Bitebene und Modulationen höherer Ordnung optimiert. Um die Optimierung zu vereinfachen und bestehende Ansätze zur Berechnung der Dekodierschwelle (z.B. EXIT oder P-EXIT) wiederverwenden zu können, wird das Konzept von Ersatzkanälen eingeführt. Die erhaltenen Ergebnisse werden durch Verteilungsdichteevolution (density evolution) und Simulationen verifiziert. Für spektrale Effizienzen von 1.5 und 2.5 Bits/Kanalbenutzung werden konkrete Codes entworfen. Es zeigt sich, dass jene Codes für die optimierten spektralen Effizienzen nahe der theoretischen Grenzen operieren, jedoch ein suboptimales Ergebnis liefern, wenn sie bei anderen Operationspunkten betrieben werden. Aus diesem Grund wird zusätzlich ein „robuster Ansatz“ entwickelt, durch den die Codes ein gutes Verhalten über eine große Breite an spektralen Effizienzen zeigen. Es werden ebenso quantisierte LDPC Dekodieralgorithmen untersucht und auf räumlich gekoppelte LDPC Codes angewandt. Nicht binäre LDPC Codes werden mit Symbol- und Bitdekodiermetriken kombiniert und für kurze Blocklängen simuliert. Hierbei zeigen sie einen klaren Vorteil gegenüber binären Codes.

Schließlich werden Anwendungen von Wahrscheinlichkeitsanpassung im Bereich der optischen Datenübertragung diskutiert. Ein Erwartungs-Maximierungs-Algorithmus wird zum blinden Schätzen der PAS Parameter vorgeschlagen, wodurch keine zusätzlichen Pilotsymbole und Kontrollinformationen mehr benötigt werden. Zudem wird PAS für Konstellationen in höheren Dimensionen erweitert. Dieses neue Verfahren wird Quadrantenwahrscheinlichkeitsanpassung genannt und stellt die Grundlage für ein neues Verfahren dar, das einen Kompromiss zwischen der Komplexität der Verteilungsanapassungskomponente sowie der Komplexität der Soft-Informationsberechnung erlaubt.

Abstract

Information theoretic tools are used to design communication systems with coded modulation and probabilistic shaping (PS) that operate close to the Shannon limit.

Probabilistic amplitude shaping (PAS) is introduced for channels for which the capacity-achieving distribution is symmetric. Achievable rates for symbol-metric decoding (SMD) and bit-metric decoding (BMD) are analyzed in the context of a layered PS architecture consisting of a forward error correction (FEC) and shaping layer. Distribution matching is explained and shown to be a central building block for PAS.

Geometric shaping (GS) is discussed and an optimization algorithm based on differential evolution is proposed to obtain optimized geometrically shaped constellations. PS and GS are compared by means of achievable rates and finite blocklength coded results. The ATSC 3.0 standard serves as a representative case study. It is concluded that GS is generally inferior to PS for the same constellation size, especially if BMD is considered. As extensions, PS strategies are developed where each bit level is shaped individually. This approach is referred to as product distribution matching and is applied for instance to parallel channels operated with different modulation formats. As a result, distribution matchers for lower bit levels may be shared among all channels, which decreases complexity. Further, achievable rates for hard-decision decoding and PAS are derived and compared to in coded simulations by means of product codes. Shaping via time-sharing is demonstrated for on-off keying, an example where PAS can not be applied because of the non-symmetric input distribution.

Binary and non-binary low-density parity-check (LDPC) codes are designed considering various constraints and target spectral efficiencies (SEs). Bit mapping optimization is discussed for BMD and higher-order modulation. The concept of surrogate channels is introduced to facilitate the code optimization and to reuse existing approaches such as EXIT and P-EXIT analysis to determine decoding thresholds. The results are verified by

density evolution and justified by finite length simulations. Specific code designs are given for SEs of 1.5 and 2.5 bits/channel use. It is shown that these specifically designed codes perform poorly when operated over a broad range of SEs for seamless rate adaptation. Consequently, a tailored optimization approach for a robust code design is proposed as well. Quantized LDPC decoding approaches are investigated and applied to spatially coupled LDPC codes. Non-binary LDPC codes are combined with SMD and BMD and simulated for short blocklength scenarios where they show superior performance compared to binary codes.

Finally, applications of PS are discussed for optical communications. An expectation maximization formulation is proposed for blindly estimating the PAS signaling parameters, which avoids the need for additional pilots and control information. Further, PAS is extended to higher dimensions by introducing the concept of quadrant shaping (QS). QS is used to show case a shaping scheme that allows a trade-off between distribution matching and demapping complexity.

1

Introduction

1.1. Motivation

Shannon's seminal work [1] provided the blueprint for capacity achieving communications and laid the theoretical foundation for all wired, wireless and optical communication systems that now constitute the backbone of our globalized world.

Although the underlying concept and proof technique are fairly easy to grasp, the non-constructive proof by random coding arguments makes it difficult to implement the optimal signaling strategy in practice. In general, communication at the Shannon limit requires two components: First, we need good forward error correction (FEC) codes and, second, the channel inputs should have the optimal input distribution. As history shows, accomplishing these two goals took major efforts over the years and the brightest minds in information theory contributed to its solution over the course of the past seventy years.

Needless to say, resignation and doubts concerning further progress was a constant trait of this endeavor. For example, the 1971 IEEE Communications Theory Workshop in St. Petersburg, Florida, became famous as the "coding is dead" workshop, as many participants could not foresee any further significant improvements in coding at the time [2, pp. 243–245]. Following R. Lucky's rhetorical question "Why are we technologists so bad at predicting the future of technology?" [2, p. 244], we now know that this sentiment was not right and the coding gain was about to increase: The invention of the Viterbi algorithm [3] emphasized the importance of soft decision inputs for the FEC decoder and the advent of iterative decoding as used by Turbo codes [4] and low-density parity-check (LDPC) codes [5] showed how the decoding of smaller component codes may result in an improved performance with manageable complexity. Recently, the invention of polar codes [6, 7] has given a constructive approach to design capacity achieving codes.

As pointed out before, a good FEC code is only one prerequisite to operate at the Shannon limit. The other one is optimized signaling adjusted to the respective channel and input constraints. For many practically relevant transmission systems, the underlying channel is well modelled by additive white Gaussian noise (AWGN). By Shannon's channel coding theorem, we know that the optimized input signal should be Gaussian distributed. With the rise of voiceband modems facing bandwidth limited channels in the 1960s (see Sec. 3.1), the need for adopting optimized signaling strategies became apparent. However, the combination with FEC turned out to be challenging.

In 2014, a new approach to combine coding and optimized signaling, called probabilistic amplitude shaping (PAS) was developed [8, 9] and was quickly adopted in practice as it circumvents many of the previous difficulties. We refer to Sec. 3.2 for a detailed discussion. This thesis focuses on this approach and investigates various aspects around PAS. In particular, we provide a comprehensive picture of the interaction of optimized coding and signaling using different FEC architectures. We focus on an information theoretic treatment as well as practical finite length and simulation based validations.

Since its invention, PAS has been investigated for various standards and applications. For wireless communications, it was considered during 5G standardization [10]. For wired access, it is investigated for new versions of digital subscriber line (DSL) standards, see, e.g., [11]. The most significant impact of PAS is in coherent optical communications. After experiments in [12, 13] that validated the theoretical gains for the AWGN channel also for fiber optical links, many other experiments and field trials followed – on dark fiber [14] and even transatlantic [15, 16, 17] and transpacific distances [18]. Steve Grubb, global optical architect at Facebook, considers probabilistic shaping (PS) "[...] as the best technique that will closely approach the Shannon limit and achieve the highest capacities possible on a submarine cable" [19].

While the increase of spectral efficiency (SE) by PS is important for optical communication, the benefits of increased flexibility are of even greater practical importance. By changing the signal distribution, the transmission rate can be finely adjusted to tune the reach and data rate. Previous transceivers did not offer such flexibility and required many different modulation and code implementations.

Nowadays, several major optical equipment manufacturers offer products that implement PS in their digital signal processors: In March 2018, Nokia presented the Photonics Service Engine 3 (PSE3)[1], claiming 65% increased capacity, while the power usage per bit is reduced by 60% and any reach from 10 km to 10 000 km is supported. Acacia implements a variant of PS by fractional quadrature amplitude modulation (QAM) constellations[2] and Ciena offers PS in their WaveLogic 5 DSP chip[3], see also [20].

The demand for increased data rates will continue and even though we may hit the Shannon limit for point-to-point links, much is still to be gained and explored for multi-user and multi-antenna setups and their equivalents for optical communications (e.g., multi-core fiber). To quote [2] again: "Fundamental studies in math should be supported. Do not be shortsighted; they will pay off in the long run."

[1] https://www.nokia.com/networks/technologies/photonic-service-engine/

[2] https://acacia-inc.com/product/ac1200/

[3] https://www.ciena.com/wavelogic/wavelogic-5

The remaining parts of the thesis are structured as follows:

▷ Chapter 2 provides the information theoretic foundations for the subsequent chapters. We start with a brief summary of results from probability theory and use these to derive the channel coding theorem, introduce the concept of achievable rates and explain the basic building blocks of a communication system. To put our simulation results into perspective, we also introduce finite length coding bounds. Further, we give a short introduction into graphs and the sum-product algorithm for inference on graphical models.

▷ Chapter 3 introduces the concept of layered PS and explains the difficulty of combining FEC with optimized signaling. We discuss PAS as one implementation of layered PS and compare it to geometric shaping which places the constellation points non-uniformly over the real axis. We then extend the principle of PAS to parallel channels and show why product distribution matching is beneficial for multi-carrier systems. We also discuss gains of PS for hard decision decoding. Finally, we consider signaling with on-off keying which has a non-symmetric capacity achieving distribution and prevents the application of PAS. Instead, we resort to a time sharing based scheme.

▷ Chapter 4 deals with the design of LDPC codes for PAS with bit-metric decoding. We review basic tools to design binary LDPC codes (density evolution, extrinsic information transfer charts) and discuss their application for the bit mapping optimization of off-the-shelf codes and for the joint code and bit mapping design of new codes. In addition, we discuss quantized LDPC decoding and introduce binary message passing, ternary messsage passing and quaternary message passing. All three approaches exploit channel soft information but pass only one or two bit messages during the iterative decoding schedule.

▷ Chapter 5 discusses decoding for PAS with non-binary low-density parity-check (NB-LDPC) codes. We investigate symbol-metric and bit-metric decoding, derive expressions for the decoder soft information and perform numerical comparisons.

▷ Chapter 6 presents applications of the previous topics to optical communications. We show how signaling parameters (e.g., the employed distribution, signal-to-noise ratio) may be estimated in a blind fashion from the received noisy signals and apply the approach to measured simulation data. Finally, we extend PAS to higher dimensions and introduce the concept of quadrant shaping. Transmission experiments validate the theoretical findings.

1.2. Contributions

Most results in this thesis appeared in the following conference proceedings and journal publications.

- ▷ 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.

- ▷ 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.

- ▷ 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,” *IEEE/OSA J. Lightw. Technol.*, vol. 34, no. 7, pp. 1599–1609, Apr. 2016.

- ▷ F. Steiner and P. Schulte, “Design of robust, protograph based LDPC codes for rate-adaptation via probabilistic shaping,” *Proc. Int. Symp. Turbo Codes and Iterative Inf. Process. (ISTC)*, 2016, pp. 56–60.

- ▷ F. Steiner and G. Böcherer, “Comparison of Geometric and Probabilistic Shaping with Application to ATSC 3.0,” in *Proc. Int. ITG Conf. Source Channel Coding (SCC)*, Hamburg, Germany, 2017.

- ▷ P. Schulte, F. Steiner, and G. Böcherer, “Four Dimensional Probabilistic Shaping for Fiber-Optic Communication,” in Proc. Advanced Photonics, Paper SpM2F.5, 2017.

- ▷ F. Steiner, G. Liva, and G. Böcherer, “Ultra-Sparse Non-Binary LDPC Codes for Probabilistic Amplitude Shaping,” in *IEEE Global Telecommun. Conf. (GLOBECOM)*, 2017, pp. 1–5.

- ▷ F. Steiner, P. Schulte, and G. Böcherer, “Approaching Waterfilling Capacity of Parallel Channels by Higher Order Modulation and Probabilistic Amplitude Shaping,” in *Proc. Ann. Conf. Inf. Sci. Syst. (CISS)*, Princeton, USA, 2018. (INVITED)

- ▷ F. Steiner, G. Kramer, “Optimization of Bit Mapping and Quantized Decoding for Off-the-Shelf Protograph LDPC Codes with Application to IEEE 802.3ca,” in *Proc. Int. Symp. Turbo Codes and Iterative Inf. Process. (ISTC)*, Hong Kong, 2018. (INVITED)

- ▷ F. Steiner, F. Da Ros, M. P. Yankov, G. Böcherer, P. Schulte, S. Forchhammer and G. Kramer, “Experimental Verification of Rate Flexibility and Probabilistic Shaping by 4D Signaling,” in *Proc. Optical Fiber Commun. Conf. (OFC)*, San Diego, USA, 2018.

- ▷ F. Steiner, P. Schulte, and G. Böcherer, “Blind Decoding-Metric Estimation for Probabilistic Shaping via Expectation Maximization,” in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Rome, Italy, 2018.

- ▷ F. Steiner, G. Böcherer, and G. Liva, “Bit-Metric Decoding of Non-Binary LDPC Codes with Probabilistic Amplitude Shaping,” *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2210-2213, 2018.

- ▷ G. Böcherer, P. Schulte, and F. Steiner, “Probabilistic Shaping and Forward Error Correction for Fiber-Optic Communication Systems,” *IEEE/OSA J. Lightw. Technol.*, vol. 37, no. 2, pp. 230–244, Jan. 2019.

- ▷ A. Git, B. Matuz, and F. Steiner, “Protograph-Based LDPC Code Design for Probabilistic Shaping with On-Off Keying,” in *Proc. Ann. Conf. Inf. Sci. Syst. (CISS)*, Baltimore, USA, 2019. (INVITED)

- ▷ M. Coşkun, G. Durisi, T. Jerkovits, G. Liva, W. Ryan, B. Stein, F. Steiner, “Efficient Error-Correcting Codes in the Short Blocklength Regime”, *Elsevier Physical Communication*, vol. 34, pp. 66-79, June 2019.

- ▷ F. Steiner, E. ben Yacoub, B. Matuz, G. Liva, A. Graell i Amat, “One and Two Bit Message Passing for SC-LDPC Codes with Higher-Order Modulation,” *IEEE/OSA J. Lightw. Technol.*, vol. 37, no. 23, pp. 5914-5925, Dec. 2019.

2

Preliminaries

2.1. Notation

We refer to the set of natural numbers as $\mathbb{N}$; if zero is included, the set is denoted as $\mathbb{N}_0$. The set of all integers is $\mathbb{Z}$. The set of real numbers is $\mathbb{R}$, the set of positive real numbers is $\mathbb{R}^+$ and the set of non-negative real numbers is $\mathbb{R}_0^+$. The set of complex numbers is $\mathbb{C}$ and j is the imaginary unit. General sets are denoted by calligraphic letters such as $\mathcal{A}$.

Depending on the context, we use different notations for vectors. In general, vectors are written in bold font and are assumed to be row vectors, i.e., $\boldsymbol{x} = (x_1, x_2, \dots, x_n)$. If the dimension is important, it is denoted as a superscript, i.e., $\boldsymbol{x} = x^n$. We use the notation $[\boldsymbol{x}]_i = x_i$ to index the i-th component of $\boldsymbol{x}$. A matrix is written in uppercase with bold font, e.g., $\boldsymbol{X}$. As for a vector, the notation $[\boldsymbol{X}]_{ij} = x_{ij}$ refers to the component in the i-th row and j-th column of $\boldsymbol{X}$.

The discrete, linear convolution of two vectors $\boldsymbol{a} = (a_1, a_2, \dots, a_k)$ and $\boldsymbol{b} = (b_1, b_2, \dots, b_l)$ is denoted by $\boldsymbol{c} = \boldsymbol{a} * \boldsymbol{b}$. The i-th entry of the $k + l - 1$ dimensional vector $\boldsymbol{c}$ is

$$c_i = \sum_j a_i b_{i-j}, \quad \forall i \in \{1, \dots, k + l - 1\}.$$

The notation $\boldsymbol{a}^{*b}$ denotes a b-fold convolution of $\boldsymbol{a}$ with itself, i.e.,

$$\boldsymbol{a}^{*b} = \underbrace{\boldsymbol{a} * \dots * \boldsymbol{a}}_{b \text{ times}}.$$

Random variables are denoted in uppercase letters; the corresponding realizations have lowercase. The probability mass function (PMF) of the random variable (RV) X is referred to as P_X, while the probability density function (PDF) is written as p_X. The function $\mathbb{1}(\cdot)$ is the indicator function and returns one if its argument is true and zero otherwise.

2.2. Probability Theory

2.2.1. Probability Space

A probability space is defined by the triple $(\Omega, \mathcal{F}, \Pr)$. The set $\Omega = \{\omega_1, \omega_2, \dots\}$ is the sample space and the set of all possible outcomes of a random experiment. These outcomes are also called *elementary events*. Sometimes, one is interested not only in the occurrence of a certain elementary event, but rather a general *event* that may consist of several elementary events $\{\omega_i\} \subseteq \Omega$. The set $\mathcal{F}$ is a σ-algebra, fulfilling the properties:

$$\Omega \in \mathcal{F} \tag{2.1}$$

$$\mathcal{A} \in \mathcal{F} \Rightarrow \mathcal{A}^{\mathrm{c}} \in \mathcal{F} \tag{2.2}$$

$$\mathcal{A}_1, \mathcal{A}_2, \dots \in \mathcal{F} \Rightarrow \bigcup_{i=1}^{\infty} \mathcal{A}_i \in \mathcal{F} \tag{2.3}$$

The probability measure $\Pr : \mathcal{F} \to [0,1]$ is a function that assigns to each element in $\mathcal{F}$ a number in the interval $[0,1]$ and fulfills the following properties for any $\mathcal{A} \in \mathcal{F}$:

$$\Pr(\mathcal{A}) \geq 0 \tag{2.4}$$

$$\Pr(\Omega) = 1 \tag{2.5}$$

$$\Pr\left(\bigcup_{i=1}^{\infty} \mathcal{A}_i\right) = \sum_{i=1}^{\infty} \Pr(\mathcal{A}_i), \quad \text{if } \mathcal{A}_i \cap \mathcal{A}_j = \emptyset, \forall i \neq j. \tag{2.6}$$

2.2.2. Conditional Probability and Stochastic Independence

The events $\mathcal{A}_i, i = 1, \dots, n$, are stochastically independent if

$$\Pr(\mathcal{A}_1 \cap \mathcal{A}_2 \cap \dots \cap \mathcal{A}_n) = \prod_{i=1}^{n} \Pr(\mathcal{A}_i). \tag{2.7}$$

The probability of the event $\mathcal{B}$ conditioned on the occurrence of event $\mathcal{A}$ with $\Pr(\mathcal{A}) > 0$ is

$$\Pr(\mathcal{B}|\mathcal{A}) = \frac{\Pr(\mathcal{A} \cap \mathcal{B})}{\Pr(\mathcal{A})}. \tag{2.8}$$

If the events $\mathcal{A}$ and $\mathcal{B}$ are independent, then (2.7) and (2.8) give

$$\Pr(\mathcal{B}|\mathcal{A}) = \Pr(\mathcal{B}). \tag{2.9}$$

Let the events $\mathcal{B}_i, i = 1, \dots, n$, *partition* Ω, i.e., $\cup_{i=1}^{n} \mathcal{B}_i = \Omega$ and $\mathcal{B}_i \cap \mathcal{B}_j = \emptyset$, $\forall i \neq j$. The *law of total probability* states that for any $\mathcal{A} \subseteq \mathcal{F}$ we have

$$\Pr(\mathcal{A}) = \sum_{i=1}^{n} \Pr(\mathcal{A}|\mathcal{B}_i) \Pr(\mathcal{B}_i). \tag{2.10}$$

Further, we can upper bound the union of events as

$$\Pr\left(\bigcup_{i=1}^{n} \mathcal{A}_i\right) \leq \sum_{i=1}^{n} \Pr(\mathcal{A}_i). \tag{2.11}$$

This bound is called a *union bound* and holds with equality if the events $\mathcal{A}_i$, $i = 1, \dots, n$, are disjoint.

2.2.3. Random Variables

For a given probability space $(\Omega, \mathcal{F}, \Pr)$, we can define a RV X as a mapping from the sample space Ω to another measurable space $\mathcal{S}$, i.e., $X : \Omega \to \mathcal{S}$. In the following, we consider sets $\mathcal{S}$ which have an *order relation.* Depending on the further properties of $\mathcal{S}$ we can distinguish two important types of RVs. If $\mathcal{S} = \mathbb{R}$, X is called a real or continuous RV, which is defined via its *cumulative distribution function (CDF)* $F_X(x)$ as

$$F_X(x) = \Pr(\{\omega \in \Omega : X(\omega) \leq x\}). \tag{2.12}$$

If the CDF is differentiable (except for a finite number of points) and continuous, we further define the PDF of RV X as

$$p_X(x) = \frac{\mathrm{d}F_X(x)}{\mathrm{d}x}. \tag{2.13}$$

If $\mathcal{S} = \mathcal{X}$ is a finite set, then the RV $X : \Omega \to \mathcal{X}$ is said to be a discrete RV and is defined as

$$P_X(x) = \Pr(\{\omega \in \Omega : X(\omega) = x\}). \tag{2.14}$$

We refer to (2.14) as the PMF of the RV X. Similarly to the continuous case, a discrete RV also has a distribution function

$$F_X(x) = \sum_{a \in \mathcal{X} : a \leq x} P_X(a). \tag{2.15}$$

From (2.4) and (2.5), we have

$$p_X(a) \geq 0 \qquad\qquad P_X(a) \geq 0 \tag{2.16}$$

$$\int_{-\infty}^{\infty} p_X(x)\,\mathrm{d}x = 1 \qquad\qquad \sum_{x \in \mathcal{X}} P_X(x) = 1. \tag{2.17}$$

Note that the requirement for the set $\mathcal{S}$ to have an order relation is important, e.g., for discrete RVs defined on finite fields (Sec. 2.3.5). Here, no ordering exists such that no CDF can be defined.

The definitions for joint and conditional density and mass functions for RVs follow in the same spirit.

The support of a PMF P_X is defined as

$$\operatorname{supp}(P_X) = \{x \in \mathcal{X} : P_X(x) > 0\}. \tag{2.18}$$

2.2.4. Moments of Random Variables

The expectation of a transformation $f(X)$ of a discrete RV X with $f : \mathcal{X} \to \mathbb{R}$ is given by

$$\mathrm{E}\left[f(X)\right] = \sum_{x\in\mathcal{X}} f(x)P_X(x). \tag{2.19}$$

For a real valued RV we have

$$\mathrm{E}\left[f(X)\right] = \int_{\mathbb{R}} f(x)p_X(x)\,\mathrm{d}x. \tag{2.20}$$

The k-th moment ($k \in \mathbb{N}$) of a RV defined on $\mathcal{X} \subseteq \mathbb{R}$ is the expected value of the k-th power of the RV X, and we have

$$\mathrm{E}\left[X^k\right] = \sum_{x\in\mathcal{X}} x^k P_X(x) \qquad \text{and} \qquad \mathrm{E}\left[X^k\right] = \int_{\mathbb{R}} x^k p_X(x)\,\mathrm{d}x. \tag{2.21}$$

The variance of X can be related to the first and second moment of X as

$$\mathrm{Var}\left[X\right] = \mathrm{E}\left[(X - \mathrm{E}\left[X\right])^2\right] = \mathrm{E}\left[X^2\right] - \mathrm{E}\left[X\right]^2. \tag{2.22}$$

For a complex valued RV the variance is defined as

$$\mathrm{Var}\left[X\right] = \mathrm{E}\left[|X - \mathrm{E}\left[X\right]|^2\right] = \mathrm{E}\left[|X|^2\right] - |\mathrm{E}\left[X\right]|^2. \tag{2.23}$$

If higher moments of a real valued RV should be calculated, *moment generating functions* are helpful. The moment generating function of the RV X is

$$M_X(r) = \mathrm{E}\left[\mathrm{e}^{rX}\right], \quad r \in \mathbb{R}. \tag{2.24}$$

The k-th moment of X is given as

$$\mathrm{E}\left[X^k\right] = \left.\frac{\mathrm{d}^k M_X(r)}{\mathrm{d}^k r}\right|_{r=0}. \tag{2.25}$$

The equality (2.25) can be proven by using the series expansion of the exponential function.

2.2.5. Functions of Random Variables

We often have to deal with functions of RVs and are interested in the resulting PDFs. Consider $Y = g(X)$, where X is a real valued RV and $g : \mathbb{R} \to \mathbb{R}$ is a monotonic, differentiable function. We have

$$\begin{aligned}F_Y(y) &= \Pr(Y \leq y) = \Pr(g(X) \leq y)\\ &= \begin{cases}\Pr(X \leq g^{-1}(y)) = F_X(g^{-1}(y)), & g \text{ is monotonically increasing}\\ \Pr(X \geq g^{-1}(y)) = 1 - F_X(g^{-1}(y)), & g \text{ is monotonically decreasing.}\end{cases}\end{aligned} \tag{2.26}$$

Therefore, the PDF of Y is

$$p_Y(y) = \frac{\mathrm{d}F_Y(y)}{\mathrm{d}y} = p_X(g^{-1}(y)) \cdot \left|\frac{\mathrm{d}g^{-1}(y)}{\mathrm{d}y}\right| = p_X(g^{-1}(y)) \cdot \left|\frac{1}{g'(g^{-1}(y))}\right| \tag{2.27}$$

where $g'(x)$ is the first derivative of $g(x)$. If the function g is not monotonic (i.e., there is no inverse function on $\mathbb{R}$), then we determine intervals on which g is monotonic and treat them separately. For this, we calculate all zeros of $g(x_i) - y = 0$, $i = 1, \ldots, N$, and get

$$p_Y(y) = \sum_{i=1}^{N} p_X(x_i) \cdot \left|\frac{1}{g'(x_i)}\right|. \tag{2.28}$$

2.2.6. Important Inequalities

The *Markov inequality* states that for a non-negative RV X, we have

$$\begin{aligned}\Pr(X \geq a) &= \int_a^\infty p_X(x)\,\mathrm{d}x \leq \int_a^\infty \frac{x}{a} p_X(x)\,\mathrm{d}x\\ &\leq \int_0^a \frac{x}{a} p_X(x)\,\mathrm{d}x + \int_a^\infty \frac{x}{a} p_X(x)\,\mathrm{d}x = \frac{\mathrm{E}\left[X\right]}{a}\end{aligned} \tag{2.29}$$

A generalized version of (2.29) is

$$\Pr(X \geq a) \leq \frac{\mathrm{E}\left[f(X)\right]}{f(a)} \tag{2.30}$$

for any monotonically increasing, non-negative function f. The proof of (2.30) follows the same steps as the previous derivation.

Using the Markov inequality (2.29), we can state *Tchebycheff's inequality*

$$\Pr(|X - \mathrm{E}\left[X\right]| \geq a) = \Pr(|X - \mathrm{E}\left[X\right]|^2 \geq a^2) \leq \frac{\mathrm{E}\left[(X - \mathrm{E}\left[X\right])^2\right]}{a^2} = \frac{\mathrm{Var}\left[X\right]}{a^2} \tag{2.31}$$

which relates the probability of the deviation of an RV from its mean to its variance. Obviously, the Markov inequality can be applied here as the RV $|X - \mathrm{E}\left[X\right]|$ is non-negative.

2.2.7. Weak Law of Large Numbers

Let $X_1, X_2, \dots, X_n$ be independent and identically distributed (iid) real-valued RVs. The sample mean S_n is

$$S_n = \frac{1}{n}\sum_{i=1}^{n} X_i. \tag{2.32}$$

We have $\mathrm{E}[S_n] = \mathrm{E}[X]$ and $\mathrm{Var}[S_n] = \mathrm{Var}[X]/n$. Using Tchebycheff's inequality (2.31), we have

$$\Pr(|S_n - \mathrm{E}[X]| > \varepsilon) = \Pr(|S_n - \mathrm{E}[X]|^2 > \varepsilon^2) \leq \frac{\mathrm{Var}[S_n]}{\varepsilon^2} = \frac{\mathrm{Var}[X]}{n\varepsilon^2} \tag{2.33}$$

for any $\varepsilon > 0$. The weak law of large numbers (WLLN) states that the sample mean S_n converges to the true mean in probability for an increasing number of samples, i.e.,

$$\Pr(|S_n - \mathrm{E}[X]| > \varepsilon) \to 0 \quad \text{for } n \to \infty. \tag{2.34}$$

2.3. Information Theory

2.3.1. Information Theoretic Quantities and Their Properties

We will first consider discrete RVs. The *self-information* of a realization of a discrete RV X with PMF P_X is defined as $-\log_2(P_X(x))$. As P_X is a PMF (2.16), it follows that $-\log_2(P_X(x))$ is always non-negative. We define the entropy of the discrete RV X as the average self-information

$$\mathrm{H}(X) = \mathrm{E}[-\log_2(P_X(X))] = -\sum_{x\in\mathrm{supp}(P_X)} P_X(x)\log_2(P_X(x)). \tag{2.35}$$

We can bound the entropy by

$$0 \leq \mathrm{H}(X) \leq \log_2(|\mathcal{X}|) \tag{2.36}$$

where the left hand side follows from the non-negativity of the self-information and the right hand side follows from

$$\mathrm{E}\left[\log_2\left(\frac{1}{P_X(X)\,|\mathcal{X}|}\right)\right] + \log_2(|\mathcal{X}|) \leq \left(\frac{|\mathrm{supp}(P_X)|}{|\mathcal{X}|} - 1\right)\frac{1}{\log(2)} + \log_2(|\mathcal{X}|) \tag{2.37}$$

where we used the inequality $\log_2(x) \leq (x-1)/\log(2)$. We have equality on the left-hand side of (2.36) by a degenerate distribution P_X and equality on the right hand side by a uniform distribution over $\mathcal{X}$.

For a binary RV X with PMF $P_X(0) = p, P_X(1) = 1-p$ and $0 \leq p \leq 1$, we introduce the *binary entropy function* $\mathrm{H}_2(p)$ as

$$\mathrm{H}_2(p) = -p\log_2(p) - (1-p)\log_2(1-p). \tag{2.38}$$

We write $\mathrm{H}_2(0) = \mathrm{H}_2(1) = 0$. $\mathrm{H}_2(p)$ is invertible on the interval $[0, 0.5]$ and we denote its inverse by ${\mathrm{H}_2}^{-1}$.

The *cross entropy* $\mathrm{X}(P_X \parallel P_Y)$ between P_X and P_Y with $\mathrm{supp}(P_X) \subseteq \mathrm{supp}(P_Y)$ is defined as

$$\mathrm{X}(P_X \parallel P_Y) = -\sum_{x\in\mathrm{supp}(P_X)} P_X(x)\log_2(P_Y(x)). \tag{2.39}$$

The *conditional entropy* of the RV $X|\{Y = y\}$ is

$$\mathrm{H}(X|Y=y) = \sum_{x\in\mathrm{supp}(P_{X|Y}(\cdot|y))} -P_{X|Y}(x|y)\log_2(P_{X|Y}(x|y)) = \mathrm{E}\left[-\log_2(P_{X|Y}(X|y)\right]. \tag{2.40}$$

Averaging (2.40) over P_Y yields the conditional entropy $\mathrm{H}(X|Y)$, i.e.,

$$\mathrm{H}(X|Y) = \sum_{y\in\mathcal{Y}} P_Y(y)\,\mathrm{H}(X|Y=y) = \mathrm{E}\left[-\log_2(P_{X|Y}(X|Y))\right]. \tag{2.41}$$

We have that $\mathrm{H}(X|Y) \le \mathrm{H}(X)$ and equality is achieved if X and Y are stochastically independent.

The *Kullback-Leibler divergence* $\mathrm{D}(P_X \parallel P_Y)$ for two distributions P_X and P_Y with $\mathrm{supp}(P_X) \subseteq \mathrm{supp}(P_Y)$ is defined as

$$\mathrm{D}(P_X \parallel P_Y) = \sum_{x\in\mathrm{supp}(P_X)} P_X(x)\log_2\left(\frac{P_X(x)}{P_Y(x)}\right) = \mathrm{E}\left[\log_2\left(\frac{P_X(X)}{P_Y(X)}\right)\right]. \tag{2.42}$$

The definition shows that the Kullback-Leibler divergence is not symmetric in its arguments. As before, we can use the inequality $\log_2(x) \le (x-1)/\log(2)$ to show that the divergence is non-negative:

$$\begin{aligned}
-\mathrm{D}(P_X \parallel P_Y) &= \mathrm{H}(X) - \mathrm{X}(P_X \parallel P_Y) && (2.43)\\
&= \sum_{x\in\mathrm{supp}(P_X)} P_X(x)\log_2\left(\frac{P_Y(x)}{P_X(x)}\right) && (2.44)\\
&\le \frac{1}{\log(2)} \sum_{x\in\mathrm{supp}(P_X)} P_X(x)\left(\frac{P_Y(x)}{P_X(x)} - 1\right) && (2.45)\\
&= \frac{1}{\log(2)} \sum_{x\in\mathrm{supp}(P_X)} P_Y(x) - 1 \le 0. && (2.46)
\end{aligned}$$

From the previous result we see that the solution to optimization problems of the form

$$\min_{P_Y} \quad \mathrm{X}(P_X \parallel P_Y) \tag{2.47}$$

is given by $P_Y = P_X$.

The *mutual information (MI)* is defined as

$$\mathrm{I}(X;Y) = \mathrm{H}(X) - \mathrm{H}(X|Y) = \mathrm{H}(Y) - \mathrm{H}(Y|X) \tag{2.48}$$

$$= \mathrm{E}\left[\log_2\left(\frac{P_{Y|X}(Y|X)}{P_Y(Y)}\right)\right] = \mathrm{D}(P_{XY} \parallel P_X P_Y) \tag{2.49}$$

The reformulation in the last step as a divergence shows that the MI is non-negative and zero if and only if X and Y are stochastically dependent. Further, from (2.36) and (2.48), we can establish the following bounds for the MI

$$0 \leq \mathrm{I}(X;Y) \leq \min(\mathrm{H}(X), \mathrm{H}(Y)). \tag{2.50}$$

The term inside the expectation in (2.49) is often referred to as *information density*

$$i(x;y) = \log_2\left(\frac{P_{Y|X}(y|x)}{\sum_{a\in\mathcal{X}} P_{Y|X}(y|a)P_X(a)}\right). \tag{2.51}$$

For a continuous RV, the concept of entropies does not exist. However, we can define the *differential entropy* $\mathrm{h}(X)$ as

$$h(X) = \mathrm{E}\left[-\log_2\left(p_X(X)\right)\right] = -\int\limits_{x\in\mathrm{supp}(p_X)} p_X(x)\log_2(p_X(x))\,\mathrm{d}x. \tag{2.52}$$

In contrast to the entropy of a discrete RV, no bounds on (2.52) can be given. In particular, the differential entropy can be negative, e.g., for a uniformly distributed RV between $[0, A]$ with $A < 1$. Most concepts introduced above can also be formulated in terms of the differential entropy.

In practice, calculating the expectations of (2.35), (2.49) and (2.52) may not be feasible if the involved RVs are high dimensional. However, if sampling from the respective distributions is possible, we can approximate them by employing the WLLN (2.34), i.e., we obtain for $n \to \infty$

$$\mathrm{H}(X) \approx -\frac{1}{n}\sum_{i=1}^{n}\log_2(P_X(x_i)) \tag{2.53}$$

and

$$\mathrm{I}(X;Y) \approx -\frac{1}{n}\sum_{i=1}^{n}\log_2\left(\frac{P_{Y|X}(y_i|x_i)}{P_Y(y_i)}\right) \tag{2.54}$$

where the samples $x_i, i = 1, \dots, n$ and sample pairs $(x_i, y_i), i = 1, \dots, n$ are distributed as P_X and $P_{Y|X}P_X$, respectively. Alternatively, numerical quadrature rules such as Gauss-Hermite quadratures can be used (see Appendix A.3).

2.3.2. Channel Coding Theorem

In this section, we briefly review the steps of the noisy channel coding theorem as stated by Gallager [21, Ch. 5]. We also discuss *mismatched decoding metrics* that were treated by Gallager in exercise 5.22[1] of [21]. Mismatched decoding metrics were also discussed by Kaplan and Shamai [22].

Problem Setting

Following James L. Massey's "Basic Information-Theoretic Model of a Digital Communication System" [23], a system model for the following investigation is shown in Fig. 2.1.

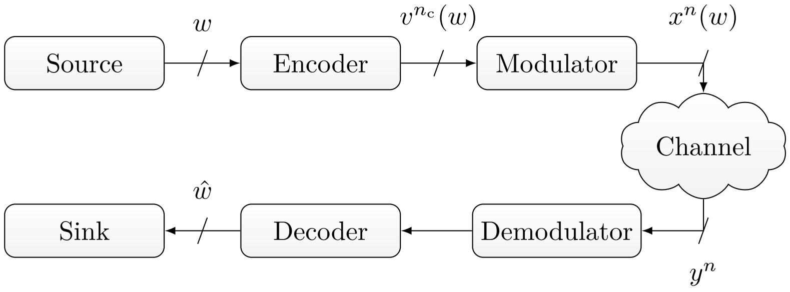


Figure 2.1.: Adaption of Massey's "Basic Information-Theoretic Model of a Digital Communication System".

We are interested in transmitting one of 2^{nR} messages $w \in \mathcal{W} = \{1, \dots, 2^{nR}\}$. We associate each message $w \in \mathcal{W}$ with a codeword of the codebook $\mathcal{C}$ with cardinality $|\mathcal{C}| = 2^{nR}$. The codeword for the message w reads $v^{n_c}(w) \in \mathcal{C}$ with the codeword symbols $v_i(w)$, $i = 1, \dots, n_c$, taken from a set $\mathcal{V}$. The modulator takes the codeword as input and returns the modulated codeword $x^n(w)$ which is a string of length n with entries taken from a set $\mathcal{X}$. The modulated codeword $x^n(w)$ is transmitted over a channel $p_{Y^n|X^n}$ and the receiver obtains a noisy estimate y^n of the original message. The decoder has the task to estimate the transmitted codeword $\hat{x}^n$ from which the estimate of the original message $\hat{w}$ can be obtained. The noisy channel coding theorem characterizes the probability $\Pr(\hat{W} \neq W)$.

[1] "A discrete memoryless channel has the transition probabilities $P(j|k)$. Unfortunately, the decoder for the channel is a maximum likelihood decoder designed under the mistaken impression that the transition probabilities are $P'(j|k)$."

Decoding Metrics

To determine the transmitted codeword from the observation y^n, the decoder uses a *decoding metric*, i.e., a function $q : \mathcal{X}^n \times \mathcal{Y}^n \to \mathbb{R}^+$ that assigns a score to each modulated codeword $x^n(w)$. The decoder chooses its estimate by selecting the message/codeword which gets the highest score, i.e.,

$$\hat{w} = \operatorname*{argmax}_{w \in \mathcal{W}} q(x^n(w), y^n). \tag{2.55}$$

For implementation reasons, most practical decoding metrics are memoryless, i.e., with a slight abuse of notation, we have

$$\hat{w} = \operatorname*{argmax}_{w \in \mathcal{W}} \prod_{i=1}^{n} q(x_i(w), y_i) \quad \text{where } x_i(w) = [x^n(w)]_i. \tag{2.56}$$

There are different types of decoding metrics. In this thesis, we distinguish between:

- ▷ Decoding metrics based on *symbol-metric decoding (SMD)*: SMD operates directly on the modulation symbols $x \in \mathcal{X}$. Hence, they can be employed with non-binary (NB) codes, where the cardinality $|\mathcal{V}|$ of the codeword alphabet is the same as the cardinality $|\mathcal{X}|$ of the modulation set[2].

- ▷ Decoding metrics based on *bit-metric decoding (BMD)*: BMD is commonly used when binary FEC codes are combined with higher-order modulation formats, i.e., $|\mathcal{X}| > 2$. To calculate a metric for each codeword bit, a marginalization step over all possibly transmitted modulation symbols is needed. BMD is used for instance in bit-interleaved coded modulation (BICM) [24].

Both approaches can be further classified into hard decision (HD) and soft decision (SD) based metrics. We characterize HD based decoding metrics as those that use a Hamming distance and do not exploit reliability information, see Sec. 3.8. Instead, SD metrics use reliability information and generally outperform HD based schemes.

An important instance of an SD SMD decoding metric is $q(x^n, y^n) = p_{Y^n|X^n}(y^n|x^n)$. It represents a maximum likelihood (ML) decoder that was analyzed by Gallager in [21]. In [22], Kaplan and Shamai use Gallager's derivation of a *mismatched decoder* [21, Exercise 5.22] and formalize the setting. In this mismatched setting, the decoder does not know (or does not use) $p_{Y^n|X^n}(y^n|x^n)$ because it has no access to it or only knows it approximately, e.g., because it does not have instantaneous channel state information (CSI), but only a time-average one. Another reason not to use $p_{Y^n|X^n}(y^n|x^n)$ is because the calculations may be too complex [25, §4]. Most practical decoding metrics are mismatched.

[2] More generally, the cardinality of the codeword alphabet may also be a power of the cardinality of the modulation set for SMD, also see Sec. 5.2

Derivation of the Decoding Error Probability via Random Coding

Calculating the probability $\Pr(\hat{W} \neq W)$ for a given code and decoding metric is very difficult in general, and it may require exhaustive Monte Carlo (MC) simulations. Instead, Shannon had the idea to resort to a *random coding* argument to investigate the average decoding error probability of an ensemble of codes. Instead of looking at a single code $\mathcal{C}$, we investigate a large set of codes, the so called code ensemble, and try to determine its properties. Given the average decoding error performance, we know that there must exist a code that is at least as good as the ensemble average. We will follow this line of thought in the following.

Without loss of generality we assume a setup where the codeword and modulation symbol set coincide, i.e., $\mathcal{V} = \mathcal{X}$. The random coding experiment to construct the codebook $\mathcal{C}$ has the following form: For each of the 2^{nR} messages in $\mathcal{W}$, choose a codeword x^n at random by sampling it from P_{X^n}. All codewords are iid. The RV X^n is defined on $\mathcal{X}^n$, i.e., the n-fold Cartesian product of $\mathcal{X}$.

The law of total probability (2.10) gives

$$\Pr(W \neq \hat{W}) = \sum_{w \in \mathcal{W}} \Pr(\hat{W} \neq w | W = w) P_W(w). \tag{2.57}$$

Consider a particular $W = w_0$ to examine the probability $\Pr(\hat{W} \neq w_0 | W = w_0)$. As the decision for $\hat{W}$ is based on maximizing the decoding metric, we decide erroneously for $\tilde{w}$ if $q(x^n(\tilde{w}), y^n) > q(x^n(w_0), y^n)$. We can write the corresponding error event as

$$\mathcal{E}(\tilde{w}) = \left\{ \frac{q(X^n(\tilde{w}), Y^n)}{q(X^n(w_0), Y^n)} \geq 1 \right\}. \tag{2.58}$$

We have

$$\begin{aligned}
&\Pr(\hat{W} \neq w_0 | W = w_0) \\
&= \sum_{x^n(w_0) \in \mathcal{X}^n} P_{X^n}(x^n(w_0)) \Pr(\hat{W} \neq w_0 | W = w_0, X^n = x^n(w_0)) \\
&= \sum_{x^n(w_0) \in \mathcal{X}^n} P_{X^n}(x^n(w_0)) \Pr\left(\bigcup_{\hat{w} \neq w_0} \mathcal{E}(\tilde{w}) \middle| W = w_0, X^n = x^n(w_0) \right) \\
&= \sum_{x^n(w_0) \in \mathcal{X}^n} P_{X^n}(x^n(w_0)) \int_{\mathbb{R}^n} p_{Y^n|X^n}(y^n | x^n(w_0)) \\
&\times \Pr\left(\bigcup_{\hat{w} \neq w_0} \left\{ \frac{q(X^n(\tilde{w}), y^n)}{q(x^n(w_0), y^n)} \geq 1 \right\} \middle| W = w_0, X^n = x^n(w_0), Y^n = y^n \right) \mathrm{d}y^n
\end{aligned} \tag{2.59}$$

where we first average over all possible codewords that may have been chosen for message w_0, and then we average over all possible noisy channel observations y^n given $x^n(w_0)$ was chosen. Let us further examine the inner term in (2.59), for which we get:

$$\Pr\left(\bigcup_{\tilde{w}\neq w_0}\left\{\frac{q(X^n(\tilde{w}),y^n)}{q(x^n(w_0),y^n)}\geq 1\right\}\middle| W=w_0, X^n=x^n(w_0), Y^n=y^n\right) \tag{2.60}$$

$$\leq \sum_{\tilde{w}\neq w_0}\Pr\left(\left\{\frac{q(X^n(\tilde{w}),y^n)}{q(x^n(w_0),y^n)}\geq 1\right\}\middle| W=w_0, X^n=x^n(w_0), Y^n=y^n\right) \tag{2.61}$$

$$\leq \sum_{\tilde{w}\neq w_0}\left(\Pr\left(\left\{\frac{q(X^n(\tilde{w}),y^n)}{q(x^n(w_0),y^n)}\geq 1\right\}\middle| W=w_0, X^n=x^n(w_0), Y^n=y^n\right)\right)^{\rho} \tag{2.62}$$

$$\leq \left(\sum_{\tilde{w}\neq w_0}\frac{\mathrm{E}\left[q(X^n(\tilde{w}),y^n)^s|W=w_0, X^n=x^n(w_0), Y^n=y^n\right]}{q(x^n(w_0),y^n)^s}\right)^{\rho} \tag{2.63}$$

$$= \left(\sum_{\tilde{w}\neq w_0}\frac{\mathrm{E}\left[q(X^n(\tilde{w}),y^n)^s\right]}{q(x_n(w_0),y^n)^s}\right)^{\rho} \tag{2.64}$$

$$= \left((2^{nR}-1)\frac{\mathrm{E}\left[q(X^n,y^n)^s\right]}{q(x^n(w_0),y^n)^s}\right)^{\rho} \tag{2.65}$$

The step in (2.61) applies the union bound (2.11) and step (2.62) introduces a parameter ρ with $0\leq\rho\leq 1$, see [21, Ch. 5.6]. In (2.63) we applied the generalized version of the Markov inequality (2.30) with $s>0$ and (2.65) follows because $\mathrm{E}\left[q(X^n(\tilde{w}),y^n)\right]$ is the same for all $\tilde{w}\neq w_0$.

We now restrict attention to a memoryless channel $p_{Y^n|X^n}(y^n|x^n)=\prod_{i=1}^n p_{Y|X}(y_i|x_i)$, a memoryless decoding metric $q(x^n,y^n)=\prod_{i=1}^n q(x_i,y_i)$ and $P_X(x^n)=\prod_{i=1}^n P_X(x_i)$. The inner part of (2.65) simplifies as

$$\begin{aligned}\frac{\mathrm{E}\left[q(X^n,y^n)^s\right]}{q(x^n(w_0),y^n)^s} &= \frac{\sum_{a^n\in\mathcal{X}^n}P_{X^n}(a^n)q(a^n,y^n)^s}{q(x^n(w_0),y^n)^s} = \frac{\sum_{a^n\in\mathcal{X}^n}\prod_{i=1}^n P_X(a_i)q(a,y_i)^s}{\prod_{i=1}^n q(x_i(w_0),y_i)^s}\\ &= \frac{\prod_{i=1}^n\sum_{a\in\mathcal{X}}P_X(a)q(a,y_i)^s}{\prod_{i=1}^n q(x_i(w_0),y_i)^s} = \prod_{i=1}^n\left(\frac{\sum_{a\in\mathcal{X}}P_X(a)q(a,y_i)^s}{q(x_i(w_0),y_i)^s}\right).\end{aligned} \tag{2.66}$$

We use (2.66) to rewrite (2.65) as

$$\begin{aligned}\Pr(\hat{W}\neq W|W=w_0) &\leq 2^{nR\rho}\cdot\prod_{i=1}^n\int_{\mathbb{R}}\sum_{x\in\mathcal{X}}p_{Y|X}(y_i|x)P_X(x)\left(\frac{\sum_{a\in\mathcal{X}}P_X(a)q(a,y_i)^s}{q(x,y_i)^s}\right)^{\rho}\mathrm{d}y_i\\ &= 2^{nR\rho}\cdot\left(\int_{\mathbb{R}}\sum_{x\in\mathcal{X}}p_{Y|X}(y|x)P_X(x)\left(\frac{\sum_{a\in\mathcal{X}}P_X(a)q(a,y)^s}{q(x,y)^s}\right)^{\rho}\mathrm{d}y\right)^n\\ &= 2^{nR\rho}2^{-nE_0(q,P_X,\rho,s)} = 2^{-n(E_0(q,P_X,\rho,s)-R\rho)}\end{aligned} \tag{2.67}$$

where we introduced the shorthand notation

$$E_0(q, P_X, \rho, s) = -\log_2 \left(\int_{\mathbb{R}} \sum_{x\in\mathcal{X}} p_{Y|X}(y|x) P_X(x) \left(\frac{\sum_{a\in\mathcal{X}} P_X(a) q(a,y)^s}{q(x,y)^s} \right)^{\rho} \mathrm{d}y \right). \tag{2.68}$$

The expression (2.67) appears in the solution manual[3] for the exercises of [21] and is discussed in further detail in [22]. The error exponent for a decoding metric q and rate R is

$$E(q, R) = \max_{0\le\rho\le 1} \max_{s\ge 0} \max_{P_X} \quad (E_0(q, P_X, \rho, s) - R\rho). \tag{2.69}$$

Gallager's Error Exponent We can obtain Gallager's random coding exponent of [21] from (2.67) by instantiating it with $q(x, y) = p_{Y|X}(y|x)$:

$$\begin{aligned}
&\Pr(\hat{W} \neq w_0 | W = w_0) \\
&= 2^{nR\rho} \left(\int_{\mathbb{R}} \sum_{x\in\mathcal{X}} P_X(x) p_{Y|X}(y|x) \frac{\left(\sum_{a\in\mathcal{X}} P_X(a) p_{Y|X}(y_i|a)^s\right)^{\rho}}{p_{Y|X}(y_i|x_i(w_0))^{s\rho}} \mathrm{d}y \right)^n \\
&= 2^{nR\rho} \left(\int_{\mathbb{R}} \sum_{x\in\mathcal{X}} P_X(x) p_{Y|X}(y|x)^{1-\rho s} \left(\sum_{a\in\mathcal{X}} P_X(a) p_{Y|X}(y_i|a)^s \right)^{\rho} \mathrm{d}y \right)^n && (2.70) \\
&= 2^{nR\rho} \left(\int_{\mathbb{R}} \left(\sum_{x\in\mathcal{X}} P_X(x) p_{Y|X}(y|x)^{\frac{1}{1+\rho}} \right)^{1+\rho} \mathrm{d}y \right)^n && (2.71) \\
&= 2^{nR\rho} 2^{-nE_{0,\mathrm{Gal}}(P_X,\rho)} = 2^{-n\left(-R\rho + E_{0,\mathrm{Gal}}(P_X,\rho)\right)} && (2.72)
\end{aligned}$$

where we set $s = 1/(1+\rho)$ from (2.70) to (2.71) and we defined

$$E_{0,\mathrm{Gal}}(P_X, \rho) = -\log_2 \left(\int_{\mathbb{R}} \left(\sum_{x\in\mathcal{X}} P_X(x) p_{Y|X}(y|x)^{\frac{1}{1+\rho}} \right)^{1+\rho} \mathrm{d}y \right). \tag{2.73}$$

The above specific choice for s is also the optimal one for this setting. In general, one must optimize over s. Using the expression for the information density (2.51), we see that (2.73) can be written as

$$E_{0,\mathrm{Gal}}(P_X, \rho) = -\log_2 \mathrm{E}\left[\exp\left(-i_{\frac{1}{1+\rho}}(X;Y) \right) \right]. \tag{2.74}$$

The form (2.74) is useful for a numerical implementation by means of Gauss Hermite quadrature rules, see Appendix A.3. We define Gallager's coding exponent as

$$E_{\mathrm{Gal}}(R) = \max_{0\le\rho\le 1} \max_{P_X} \quad (E_{0,\mathrm{Gal}}(P_X, \rho) - \rho R). \tag{2.75}$$

[3] The solution manual discusses the case for a discrete memoryless channel, but the structure is the same.

Error Exponent for BMD and a Product Input Distribution We now investigate a mismatched case with the bitwise decoding metric (see Sec. 2.3.6 for the employed notation)

$$q(x,y) = \prod_{k=1}^{m} p_{Y|B_k}(y|b_k), \quad \boldsymbol{b} = (b_1, b_2, \dots, b_m) = \chi(x). \tag{2.76}$$

The random coding experiment samples from P_X, which factors as

$$P_X(x) = P_{\boldsymbol{B}}(\chi(x)) = \prod_{k=1}^{m} P_{B_k}(b_k).$$

Now (2.68) gives:

$$\begin{aligned}
&E_0^{\text{BMD}}(P_X, \rho, s)\\
&= -\log_2\left(\sum_{x\in\mathcal{X}} \prod_{k=1}^{m} P_{B_k}([\chi(x)]_k) \int_{\mathbb{R}} p_{Y|X}(y|x) \left(\frac{\sum_{a\in\mathcal{X}} P_X(a) \prod_{k=1}^{m} p_{Y|B_k}(y|[\chi(a)]_k)^s}{\prod_{k=1}^{m} p_{Y|B_k}(y|[\chi(x)]_k)^s}\right)^{\rho} \mathrm{d}y\right)\\
&= -\log_2\left(\sum_{x\in\mathcal{X}} \prod_{k=1}^{m} P_{B_k}([\chi(x)]_k) \int_{\mathbb{R}} p_{Y|X}(y|x) \left(\frac{\sum_{a\in\mathcal{X}} \prod_{k=1}^{m} p_{Y|B_k}(y|[\chi(a)]_k)^s P_{B_k}([\chi(a)]_k)}{\prod_{k=1}^{m} p_{Y|B_k}(y|[\chi(x)]_k)^s}\right)^{\rho} \mathrm{d}y\right)\\
&= -\log_2\left(\sum_{x\in\mathcal{X}} \prod_{k=1}^{m} P_{B_k}([\chi(x)]_k) \int_{\mathbb{R}} p_{Y|X}(y|x) \left(\prod_{k=1}^{m} \frac{\sum_{b\in\{0,1\}} p_{Y|B_k}(y|b)^s P_{B_k}(b)}{p_{Y|B_k}(y|[\chi(x)]_k)^s}\right)^{\rho} \mathrm{d}y\right).
\end{aligned} \tag{2.77}$$

We define the random coding exponent for BMD with a product input distribution as

$$E^{\text{BMD}}(R) = \max_{0\le\rho\le1} \max_{s\ge0} \max_{P_X} \quad \left(E_0^{\text{BMD}}(P_X, \rho, s) - \rho R\right). \tag{2.78}$$

2.3.3. Information Rates

We want to define error free transmission by requiring that the average probability of error approaches zero as $n \to \infty$. Any rate R for which the random coding error exponent is non-negative is an achievable rate. As shown in Appendix A.1, an achievable rate is

$$R = \left.\frac{\mathrm{d}E_0(q, P_X, \rho, s)}{\mathrm{d}\rho}\right|_{\rho=0}. \tag{2.79}$$

To calculate the derivative, we write (2.68) as

$$\begin{aligned}
E_0(q, P_X, \rho, s) &= -\log_2\left(\mathrm{E}\left[\left(\underbrace{\frac{\sum_{a\in\mathcal{X}} P_X(a) q(a,Y)^s}{q(X,Y)^s}}_{Z}\right)^{\rho}\right]\right) = -\log_2\left(\mathrm{E}\left[Z^{\rho}\right]\right)\\
&= -\log_2\left(\mathrm{E}\left[2^{\log_2(Z^{\rho})}\right]\right)
\end{aligned} \tag{2.80}$$

and apply the differentiation laws of the moment generating function (2.25). We obtain

$$\left.\frac{\mathrm{d}E_0(q, P_X, \rho, s)}{\mathrm{d}\rho}\right|_{\rho=0} = -\operatorname{E}\left[\log_2(Z)\right] = \operatorname{E}\left[\log_2\left(\frac{q(X,Y)^s}{\sum_{a\in\mathcal{X}} q(a,Y)^s P_X(a)}\right)\right] \tag{2.81}$$

and define the *generalized mutual information (GMI)* [22]

$$R_{\mathrm{GMI}} = \max_{s\geq 0} \quad \operatorname{E}\left[\log_2\left(\frac{q(X,Y)^s}{\sum_{a\in\mathcal{X}} q(a,Y)^s P_X(a)}\right)\right]. \tag{2.82}$$

Instantiating with $q(x,y) = p_{Y|X}(y|x)$, we get

$$R = \mathrm{I}(X;Y) = \operatorname{E}\left[\log_2\left(\frac{p_{Y|X}(Y|X)}{p_Y(Y)}\right)\right]. \tag{2.83}$$

Similarly, for BMD as shown in (2.76), we have

$$R_{\mathrm{BICM}} = \sum_{k=1}^{m} \mathrm{I}(B_k;Y) \tag{2.84}$$

which is known as the “BICM capacity” [26]. Subsequent works [27, 28, 29] extended the framework and notion of the GMI. For both (2.83) and (2.84), the optimization over s results in $s = 1$. This can be seen by noting that after inserting the respective metrics, the resulting expressions can be understood as an instance of (2.47).

2.3.4. Important Channel Models and Their Capacities

Symmetric Channels

A discrete input, discrete output channel $P_{Y|X}$ is said to be *symmetric* if the columns of the corresponding channel transition probability matrix are permutations of each other.[4] As a result, we have $\mathrm{H}(Y|X) = \mathrm{H}(Y|X = x)$ and the conditional entropy does not depend on the distribution P_X of the channel input X.

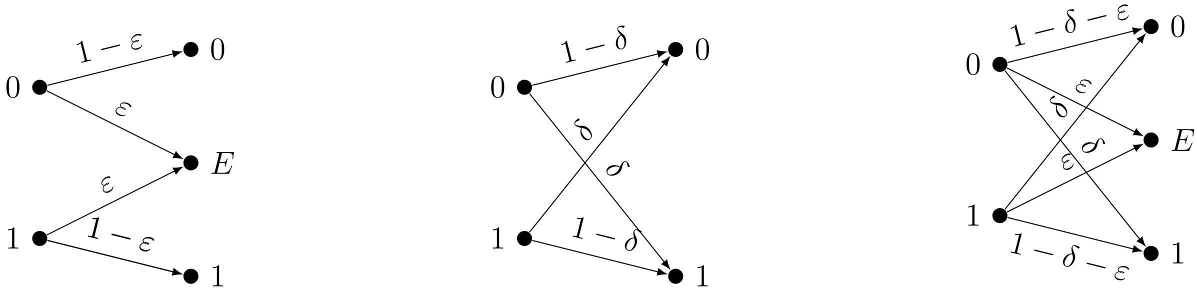
For channels with a continuous output alphabet $y \in \mathbb{R}$, we define symmetry by

$$p_{Y|X}(y|x) = p_{Y|X}(-y| - x). \tag{2.85}$$

Binary Erasure Channel (BEC)

The binary erasure channel (BEC) has a ternary output alphabet $\mathcal{Y} = \{0, 1, E\}$ and its model is shown in Fig. 2.2a. The transmitted symbols are either correctly received or completely unknown, which is denoted by the erasure symbol E. The channel transition

[4] This assumes that the channel transition probability matrix is defined such that the probabilities $P_{Y|X}(\cdot|x)$ are arranged as columns.



(a) BEC (b) BSC (c) BEEC

Figure 2.2.: Channel models for important binary input channels.

probabilities are

$$\begin{aligned} P_{Y|X}(0|0) = P_{Y|X}(1|1) &= 1 - \varepsilon \\ P_{Y|X}(E|0) = P_{Y|X}(E|1) &= \varepsilon. \end{aligned} \tag{2.86}$$

As the channel is symmetric, a uniform input distribution is capacity achieving and we calculate

$$C_{\text{BEC}} = 1 - \varepsilon. \tag{2.87}$$

Binary Symmetric Channel (BSC)

The binary symmetric channel (BSC) has a binary output alphabet $\mathcal{Y} = \{-1, +1\}$ and its model is shown in Fig. 2.2b. The transmitted symbols are received either correctly or incorrectly. The channel transition probability is

$$\begin{aligned} P_{Y|X}(0|0) = P_{Y|X}(1|1) &= 1 - \delta \\ P_{Y|X}(1|0) = P_{Y|X}(0|1) &= \delta. \end{aligned} \tag{2.88}$$

As for the BEC, the BSC is symmetric and a uniform input distribution is capacity achieving. We have

$$C_{\text{BSC}} = 1 - \mathrm{H}_2(\delta). \tag{2.89}$$

Binary Error and Erasure Channel (BEEC)

The binary error and erasure channel (BEEC) is a combination of the BEC and BSC, such that the channel output is ternary $\mathcal{Y} = \{0, 1, E\}$ and both errors and erasures may occur. The channel transition probability is

$$\begin{aligned} P_{Y|X}(0|0) = P_{Y|X}(1|1) &= 1 - \delta - \varepsilon \\ P_{Y|X}(1|0) = P_{Y|X}(0|1) &= \delta \\ P_{Y|X}(E|0) = P_{Y|X}(E|1) &= \varepsilon \end{aligned} \tag{2.90}$$

The channel defined by (2.90) is again symmetric and we obtain

$$C_{\text{BEEC}} = (1-\varepsilon) \cdot \left(1 - \mathrm{H}_2\left(\frac{\delta}{1-\varepsilon}\right)\right). \tag{2.91}$$

Additive White Gaussian Noise Channel (AWGNC)

In contrast to the previously considered channels, the AWGN channel has a continuous input and output, i.e., $\mathcal{X} = \mathcal{Y} = \mathbb{R}$. The model is

$$Y = X + N. \tag{2.92}$$

The RV N is Gaussian distributed with zero mean and variance σ^2, i.e., $N \sim \mathcal{N}(0, \sigma^2)$. The channel is characterized by the PDF

$$p_{Y|X}(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} \mathrm{e}^{-\frac{(y-x)^2}{2\sigma^2}}. \tag{2.93}$$

We obtain the MI

$$\mathrm{I}(X;Y) = \mathrm{h}(Y) - \mathrm{h}(Y|X) = \mathrm{h}(Y) - \mathrm{h}(N) = \mathrm{h}(Y) - \frac{1}{2}\log_2(2\pi\mathrm{e}\sigma^2). \tag{2.94}$$

To find the capacity of the AWGN channel with an average power constraint on the channel input, i.e., $\mathrm{E}\left[X^2\right] \leq P$, we maximize (2.94) over p_X. This implies we have to solve

$$\max_{p_X} \quad \mathrm{h}(Y) \qquad \text{subject to } \mathrm{E}\left[X^2\right] \leq P. \tag{2.95}$$

The average power constraint on X implies an average power constraint on Y such that (assuming the channel input X and the noise N to be stochastically independent) $\mathrm{E}\left[Y^2\right] = \mathrm{E}\left[X^2\right] + \mathrm{E}\left[N^2\right] \leq P + \sigma^2$. For an average power constraint, the differential entropy is maximized by a Gaussian distribution [30, Sec. 2.5.3] which implies that Y must be Gaussian with zero mean and variance $P + \sigma^2$. If X is Gaussian, Y is Gaussian as well[5]. Therefore, we achieve capacity by choosing $X \sim \mathcal{N}(0, P)$ and the capacity is

$$\begin{aligned} C_{\text{AWGN}} &= \mathrm{h}(Y) - \frac{1}{2}\log_2(2\pi\mathrm{e}\sigma^2) = \frac{1}{2}\log_2(2\pi\mathrm{e}(P+\sigma^2)) - \frac{1}{2}\log_2(2\pi\mathrm{e}\sigma^2) \\ &= \frac{1}{2}\log_2\left(1 + \frac{P}{\sigma^2}\right). \end{aligned} \tag{2.96}$$

The capacity C_{AWGN} is the central quantity that we want to approach in this thesis. It serves as the fundamental benchmark to assess our coding schemes. The ratio P/σ^2 is called the signal-to-noise ratio (SNR). The capacity (2.96) is strictly increasing in the

[5] The PDF of a sum of independent RVs is given by the convolution of the PDFs of the summands. The convolution of two Gaussian PDFs is again Gaussian.

SNR so that an inverse exists. We denote this inverse by C_{AWGN}^{-1} in the following.

We now investigate the impact of a non-optimal, i.e., non Gaussian, distribution on the channel input. Let $\tilde{X}$ denote the zero mean RV $\tilde{X}$ with $\mathrm{E}\left[\tilde{X}^2\right] = P$ for the suboptimal channel input and let $\tilde{Y}$ be the respective channel output. The MI $\mathrm{I}(\tilde{X};\tilde{Y})$ is

$$\mathrm{I}(\tilde{X};\tilde{Y}) = C_{\mathrm{AWGN}} - \mathrm{D}(p_{\tilde{Y}} \parallel p_Y). \tag{2.97}$$

Hence, the "penalty" from not using the optimal distribution is characterized by the Kullback Leibler divergence of the respective channel output PDFs

$$p_{\tilde{Y}}(y) = \int_{\mathbb{R}} p_{Y|X}(y|x) p_{\tilde{X}}(x)\,\mathrm{d}x \qquad \text{and} \qquad p_Y(y) = \int_{\mathbb{R}} p_{Y|X}(y|x) p_X(x)\,\mathrm{d}x. \tag{2.98}$$

This follows from the property that

$$\begin{aligned}\mathrm{D}(p_{\tilde{Y}} \parallel p_Y) &= \int_{\mathbb{R}} p_{\tilde{Y}}(y) \log_2\left(\frac{p_{\tilde{Y}}(y)}{p_Y(y)}\right)\mathrm{d}y = -\,\mathrm{h}(p_{\tilde{Y}}) - \int_{\mathbb{R}} p_{\tilde{Y}}(y)\log_2(p_Y(y))\,\mathrm{d}y && (2.99)\\ &= -\,\mathrm{h}(\tilde{Y}) + \frac{1}{2}\log_2(2\pi\mathrm{e}(P+\sigma^2)) = -\,\mathrm{h}(\tilde{Y}) + \mathrm{h}(Y). && (2.100)\end{aligned}$$

The result (2.97) has a practical implication as we will see in subsequent sections: Even though a system uses a non-optimal input distribution, it may still operate close to capacity if the Kullback-Leibler distance between the implied output distributions is small.

2.3.5. Finite Fields and Linear Block Codes

Finite Fields

To impose structure on codes, the concepts of *groups* and *finite fields* are beneficial [31, Ch. 2]. A group is an algebraic structure $(\mathbb{G},+)$ with a set of elements $\mathbb{G}$ and an operation $+$ such that the following four properties hold:

- ▷ Closure: For any $a, b \in \mathbb{G}$, it must hold that $a + b \in \mathbb{G}$.
- ▷ Associativity: It must hold that $a + (b + c) = (a + b) + c, \quad \forall a, b, c \in \mathbb{G}$.
- ▷ Neutral element: $\exists 0 \in \mathbb{G} : a + 0 = a, \quad \forall a \in \mathbb{G}$.
- ▷ Inverse element: $\exists(-a) \in \mathbb{G} : a + (-a) = 0, \quad \forall a \in \mathbb{G}$.

If commutativity holds, i.e., $a + b = b + a \in \mathbb{G}$, then the tuple $(\mathbb{G},+)$ is called an *Abelian group.*

Example 1. The set $\mathbb{F}_2 = \{0, 1\}$ with operation $+$ which is defined via mod 2 addition is an Abelian group. The addition table is

$+$	0	1
0	0	1
1	1	0

The neutral element is 0 and the elements of $\mathbb{F}_2$ are self inverse.

A *field* is an algebraic structure $(\mathbb{G}, +, \cdot)$ with the set $\mathbb{G}$ and two operations $+$ and $\cdot$ such that

- $(\mathbb{G}, +)$ is an Abelian group with respect to $+$ and neutral element 0. $(\mathbb{G}, +)$ is also referred to as the additive group of $\mathbb{G}$.
- $(\mathbb{G} \setminus \{0\}, \cdot)$ is an Abelian group with respect to $\cdot$. $(\mathbb{G} \setminus \{0\}, \cdot)$ is referred to as the multiplicative group of $\mathbb{G}$.
- The *distributive law* $a \cdot (b + c) = a \cdot b + a \cdot c$ holds for any $a, b, c \in \mathbb{G}$.

For coding applications, *finite fields*, i.e., fields where the respective set contains a finite number of elements are of major importance. To construct finite fields, we resort to a construction based on polynomials. The set of all polynomials with coefficients in $\mathbb{G}$ (with $|\mathbb{G}| = p$ and p being a prime number) is denoted as $\mathbb{G}[x]$. Let $f(x) = \sum_{i=1}^{o} f_i x^i$ be a polynomial over $\mathbb{G}$ of degree o, i.e., $\deg(f(x)) = o$. The polynomial $f(x)$ is called a *prime polynomial*, if $f_o = 1$ and if $f(x)$ is irreducible, meaning that $f(x)$ can not be decomposed into a product of two or more polynomials over $\mathbb{G}[x]$ with both degrees larger or equal than one. The polynomials

$$\mathbb{F}_q = \mathbb{G}[x] \mod f(x) \tag{2.101}$$

then form a finite field with $q = p^o$ elements.

Example 2. We want to construct a finite field with 8 elements over $\mathbb{G} = \mathbb{F}_2$. According to the previous definition, we need a prime polynomial of degree $o = 3$. We find that two such polynomials exist and choose $f(x) = 1 + x + x^3$ as prime polynomial (The other one is $f(x) = 1 + x^2 + x^3$.). The finite field is $\mathbb{F}_8 = \{0, 1, x, x^2, 1 + x, 1 + x^2, x + x^2, 1 + x + x^2\}$.

While the construction of the addition table is straightforward, the construction of the multiplication table turns out to be tedious. We exploit the fact that the multiplicative group $\mathbb{F}_{p^o} \setminus \{0\}$ is cyclic, i.e., each element can be written as a power of the *primitive element* $\alpha \in \mathbb{F}_{p^o} \setminus \{0\}$, i.e., $\mathbb{F}_{p^o} = \left\{0, \alpha^0, \alpha^1, \ldots, \alpha^{p^o - 2}\right\}$ where $\alpha^i \cdot \alpha^j = \alpha^{(i+j) \mod (p^o - 1)}$. To establish the addition table based on the primitive element, we need the notion of a *minimal* and *primitive polynomial*: The minimal polynomial of an element $\beta \in \mathbb{F}_{p^o}$ is the monic polynomial in $\mathbb{F}_p[x]$ of smallest degree that has β as its root. The primitive polynomial is the minimal polynomial of the primitive element α.

Linear Block Codes

A $(n_\mathrm{c}, k_\mathrm{c})$ linear block code $\mathcal{C}$ over a finite field $\mathbb{F}_q$ is a k_c dimensional subspace of $\mathbb{F}_q^n$. We define its rate R_c as

$$R_\mathrm{c} = \frac{\log_2(|\mathcal{C}|)}{n_\mathrm{c}} = \frac{\log_2(q^{k_\mathrm{c}})}{n_\mathrm{c}} = \frac{k_\mathrm{c}}{n_\mathrm{c}} \log_2(q). \tag{2.102}$$

A linear block code can be defined in two ways. First, the code $\mathcal{C}$ may be specified by its generator matrix $\boldsymbol{G} \in \mathbb{F}_q^{k_\mathrm{c} \times n_\mathrm{c}}$ as

$$\mathcal{C} = \left\{ \boldsymbol{v} \in \mathbb{F}_q^{n_\mathrm{c}} : \boldsymbol{v} = \boldsymbol{u}\boldsymbol{G}, \boldsymbol{u} \in \mathbb{F}_q^{k_\mathrm{c}} \right\} \tag{2.103}$$

where the generator matrix consists of k_c linearly independent row vectors $\boldsymbol{g}_i, i = 1, \ldots, k_\mathrm{c}$, that form the basis of the code $\mathcal{C}$. Therefore, the code consists of all vectors $\boldsymbol{v}$ that can be represented as a linear combination of the basis vectors.

Alternatively, the code is specified by its full rank parity-check matrix $\boldsymbol{H} \in \mathbb{F}_q^{m_\mathrm{c} \times n_\mathrm{c}}$ with $m_\mathrm{c} = n_\mathrm{c} - k_\mathrm{c}$ via

$$\mathcal{C} = \left\{ \boldsymbol{v} \in \mathbb{F}_q^{n_\mathrm{c}} : \boldsymbol{v}\boldsymbol{H}^\mathrm{T} = \boldsymbol{0} \right\}. \tag{2.104}$$

The dual code of $\mathcal{C}$ is denoted as $\mathcal{C}^\perp$ and is defined as

$$\mathcal{C}^\perp = \left\{ \boldsymbol{x} \in \mathbb{F}_q^{n_\mathrm{c}} : \boldsymbol{x}\boldsymbol{v}^\mathrm{T} = 0, \quad \forall \boldsymbol{v} \in \mathcal{C} \right\}. \tag{2.105}$$

Combining (2.103) and (2.104), it follows that $\boldsymbol{H}$ is a generator matrix of $\mathcal{C}^\perp$ and $\boldsymbol{G}$ is a parity-check matrix of $\mathcal{C}^\perp$. Consequently, the dimension of the dual code $\mathcal{C}^\perp$ is $(n_\mathrm{c} - k_\mathrm{c})$.

For practical purposes, *systematic encoding* is beneficial, where the generator matrix is decomposed into an identity matrix of dimension $k_\mathrm{c} \times k_\mathrm{c}$ and a parity-forming matrix $\boldsymbol{P}$ of size $k_\mathrm{c} \times (n_\mathrm{c} - k_\mathrm{c})$, i.e.,

$$\boldsymbol{G} = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{P} \end{pmatrix}. \tag{2.106}$$

After encoding the information vector $\boldsymbol{u}$, it appears as the first part of the codeword $\boldsymbol{v}$ again since

$$\boldsymbol{v} = \boldsymbol{u}\boldsymbol{G} = \boldsymbol{u} \begin{pmatrix} \boldsymbol{I} & \boldsymbol{P} \end{pmatrix} = \begin{pmatrix} \boldsymbol{u} & \boldsymbol{u}\boldsymbol{P} \end{pmatrix}.$$

Any generator matrix $\boldsymbol{G}$ can be brought into systematic form by Gaussian elimination[6]. To define a linear code, we specify its generator matrix $\boldsymbol{G}$ or parity-check matrix $\boldsymbol{H}$.

The *Hamming weight* $w_\mathrm{H}(\boldsymbol{v})$ of a binary vector $\boldsymbol{v} = (v_1, \ldots, v_{n_\mathrm{c}})$ is defined as

$$w_\mathrm{H}(\boldsymbol{v}) = \sum_{i=1}^{n_\mathrm{c}} \mathbb{1}\left(v_i \neq 0\right). \tag{2.107}$$

[6] Sometimes, additional column permutations are necessary to obtain the form in (2.106).

We use (2.107) to define the *minimum distance* of a linear code $\mathcal{C}$ as

$$d_{\min} = \min_{\substack{\boldsymbol{v}_1, \boldsymbol{v}_2 \in \mathcal{C} \\ \boldsymbol{v}_1 \neq \boldsymbol{v}_2}} w_{\mathrm{H}}(\boldsymbol{v}_1 - \boldsymbol{v}_2) = \min_{\boldsymbol{v} \in \mathcal{C} \setminus \{\boldsymbol{0}\}} w_{\mathrm{H}}(\boldsymbol{v}) \tag{2.108}$$

where the last step exploits that the sum of two codewords is again a codeword. Hence, the codeword with the lowest Hamming weight determines the minimum distance of the code. For many classical algebraic codes, the minimum distance is directly related to their (guaranteed) *error correction capability* t as

$$t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor. \tag{2.109}$$

We evaluate the performance of a blockcode by its frame error rate (FER) or bit error rate (BER), which is given as

$$\mathrm{FER} = \Pr(W \neq \hat{W}) = \Pr(\boldsymbol{U} \neq \hat{\boldsymbol{U}}), \tag{2.110}$$

$$\mathrm{BER} = \frac{1}{k_{\mathrm{c}}} \sum_{i=1}^{k_{\mathrm{c}}} \Pr(U_i \neq \hat{U}_i). \tag{2.111}$$

The information sequence associated with the message W is denoted by the RV $\boldsymbol{U} = (U_1, U_2, \ldots, U_{k_{\mathrm{c}}})$. The RV $\hat{\boldsymbol{U}} = (\hat{U}_1, \hat{U}_2, \ldots, \hat{U}_{k_{\mathrm{c}}})$ is the decision for $\boldsymbol{U}$ at the receiver after FEC decoding. These metrics are usually evaluated by means of MC simulations.

Cyclic and Quasi-Cyclic Codes

A cyclic code $\mathcal{C}$ [31, Ch. 5] has the property that each cyclic rotation of a codeword is again a codeword. A cyclic shift of the codeword $\boldsymbol{v} = (v_1, v_2, \ldots, v_{n_{\mathrm{c}}}) \in \mathcal{C}$ *to the right by one position* is defined as the operation

$$(v_1, v_2, \ldots, v_{n_{\mathrm{c}}}) \mapsto (v_{n_{\mathrm{c}}}, v_1, \ldots, v_{n_{\mathrm{c}}-1}). \tag{2.112}$$

Each cyclic code is also a linear code. Compared to linear codes, cyclic codes impose further structural properties on the codewords which can be exploited for encoding and decoding with lower complexity.

The codewords of a quasi-cylic (QC) code $\mathcal{C}$ are sectioned into t parts of length Q, i.e., we have

$$\boldsymbol{v} = (v_1, v_2, \ldots, v_{n_{\mathrm{c}}}) = (\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_t) \tag{2.113}$$

where $\boldsymbol{v}_i = (v_{(i-1)Q+1}, \ldots, v_{iQ})$. If all sections $\boldsymbol{v}_i$, $i = 1, \ldots, t$ are shifted by the same offset, the resulting vector is again a codeword. All codes in Chapters 4 and 5 are based on QC constructions, as they have a smaller description complexity than random LDPC codes and emerge naturally by building a code from a protograph ensemble.

Bose-Chaudhuri-Hocquenghem Codes

Bose-Chaudhuri-Hocquenghem (BCH) codes [31, Ch. 6] are cyclic codes and binary subcodes of Reed Solomon codes with a guaranteed error correction capability of t. For a BCH code with blocklength $n_{\mathrm{c}} = 2^o - 1$ and error correction capability t, the generator polynomial is defined as

$$g(x) = \mathrm{lcm}\left(\Phi_{\alpha}(x), \Phi_{\alpha^2}(x), \dots, \Phi_{\alpha^{2t}}(x)\right) \tag{2.114}$$

where $\Phi_{\alpha^j}(x)$ is the minimal polynomial of the element $\alpha^j \in \mathbb{F}_{2^o}$. The resulting code dimension is $k_{\mathrm{c}} = 2^o - 1 - \deg(g(x))$. BCH codes have a minimum distance of $d_{\mathrm{min}} \geq d_{\mathrm{min,d}} = 2t + 1$ and are usually decoded by the Berlekamp-Massey algorithm [32].

Product Codes

Product codes [31, Ch. 4.7] can be understood as two dimensional blockcodes, where the codeword is a two dimensional matrix with its rows and columns being formed by the constraints of two linear blockcodes as component codes. Assuming a row code with parameters $(n_{\mathrm{c}}^{\mathrm{r}}, k_{\mathrm{c}}^{\mathrm{r}})$ and a column code with parameters $(n_{\mathrm{c}}^{\mathrm{c}}, k_{\mathrm{c}}^{\mathrm{c}})$, the product code has the parameters $(n_{\mathrm{c}}^{\mathrm{r}} \cdot n_{\mathrm{c}}^{\mathrm{c}}, k_{\mathrm{c}}^{\mathrm{c}} \cdot k_{\mathrm{c}}^{\mathrm{r}})$. Further, it can be shown that its minimum distance is given by the product of the minimum distances of the respective component codes.

Product codes can be decoded iteratively in a SD or HD manner. For the former, two dimensional product codes can be understood as generalized LDPC codes (see Sec. 4) with degree two variable nodes (VNs) and are decoded by the sum-product algorithm (SPA). For HD decoding, the component codes are commonly decoded via syndrome decoding (for high rate component codes) or by the Berlekamp Massey algorithm [32].

Example 3. We consider a (9,4) product code with a (3,2) single-parity check (SPC) code for the row and column component codes. The codeword array $\boldsymbol{V}$ is

$$\boldsymbol{V} = \begin{array}{|c|c|c|} \hline v_1 & v_2 & v_1 + v_2 \\ \hline v_3 & v_4 & v_3 + v_4 \\ \hline v_1 + v_3 & v_2 + v_4 & \begin{array}{c} v_1 + v_2 \\ + \\ v_3 + v_4 \end{array} \\ \hline \end{array}$$

Here, v_1, v_2, v_3, v_4 denote the bits of the systematic information part and all other bits (shaded in dark gray) are calculated as the modulo 2 sum of the rows and columns, respectively. The bit in the lower east corner is a “check on a check”.

For the HD case, the component codes are often BCH codes. We use product codes in Sec. 3.8.4 to evaluate the performance of a shaped coded modulation setup with HD.

2.3.6. Modulation Formats and Labeling

After FEC encoding, the bits are mapped to constellation symbols. An M-ary constellation is a set $\mathcal{X}$ of M points that are used for the pulse shaping. Common constellations are M-amplitude shift keying (ASK), M-QAM, M-phase-shift keying (PSK) and M-amplitude phase-shift keying (APSK). M-ASK is also known as bipolar pulse-amplitude modulation (PAM). The number M of points is usually chosen as a power of two, i.e., $M = 2^m, m \in \mathbb{N}$. We distinguish between coherent and non-coherent modulation formats, where the latter do not convey information in the phase of a constellation point, i.e., only the amplitude carries information. Examples for non-coherent modulation formats include on-off keying (OOK) or unipolar PAM formats used with direct detection (DD)/intensity modulation (IM) transceivers. An overview of different coherent modulation formats is shown in Fig. 2.3.

In this thesis, ASK and QAM modulation formats are used in most cases. We define the normalized M-ASK signaling set as $\mathcal{X}_{\text{ASK}} = \{\pm 1, \pm 3, \dots, \pm(M-1)\}$, where M is even. The extension to a two-dimensional M^2-QAM constellation is straightforward: take the Cartesian product of two real-valued ASK constellations, i.e., $\mathcal{X}_{\text{QAM}} = \mathcal{X}_{\text{ASK}} \times \mathcal{X}_{\text{ASK}}$.

For optical communications, higher dimensional modulations formats are of interest. For instance, a four-dimensional dual polarized QAM (DP-QAM) constellation is the Cartesian product of two QAM constellations, or equivalently, the Cartesian product of four ASK constellations.

To use binary FEC codes, we introduce a binary interface for the transmitted constellation points. We define the binary labeling function $\chi\colon \mathcal{X} \to \{0,1\}^m$

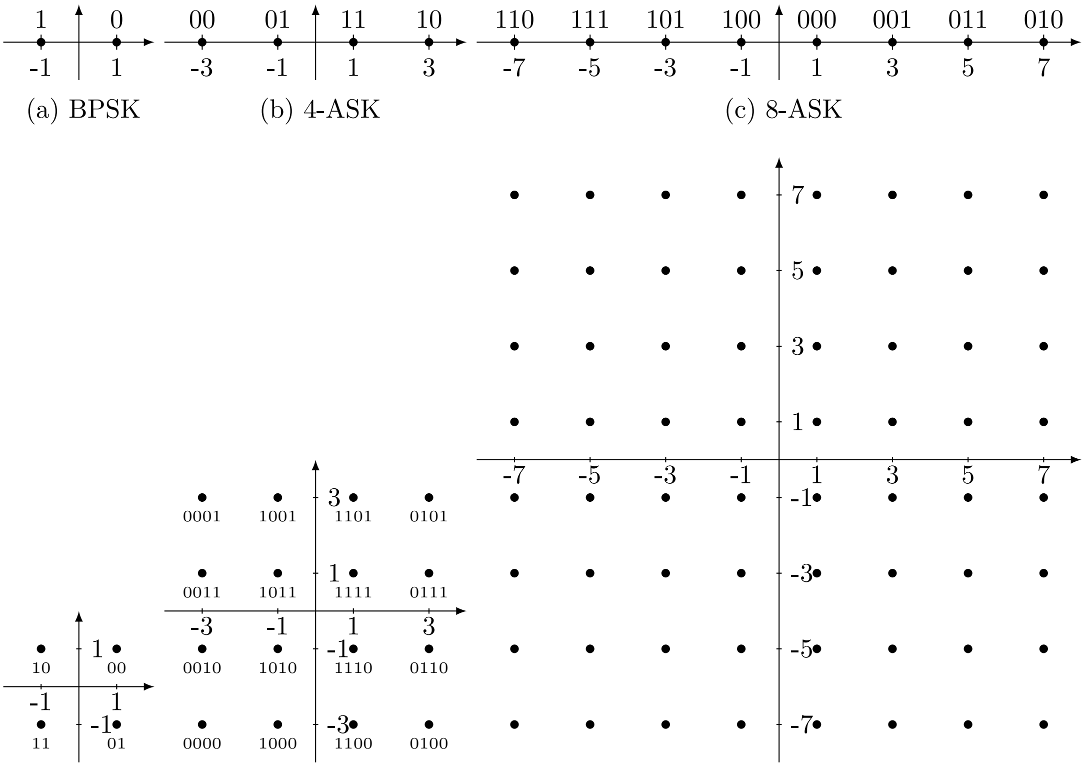
$$\chi(x) = \boldsymbol{b} = b_1 b_2 \dots b_m \tag{2.115}$$

that assigns an m-bit binary label $\boldsymbol{b}$ to each constellation point. A binary reflected Gray code (BRGC) [33] usually works well in practice. It is depicted in Fig. 2.3.

2.3.7. Finite Length Coding Bounds

To evaluate the performance in the finite blocklength regime and to guide the design of practical communication systems, finite length coding bounds are an important tool. Their significance has increased with the advent of ultra-reliable low-latency communication (URLLC) in 5G. The low latency aspect refers mainly to low processing latency, and hence, small blocklengths. While results on finite blocklength information theory date back to works by Feinstein [34], Shannon [35], Strassen [36], Gallager [37] and others, broad interest was sparked by Polyanskiy's seminal work [38] and many subsequent papers that provided additional details on how to compute the presented bounds in a numerically and computationally feasible manner.

In the following, we review some of these finite length bounds in detail. They will serve as benchmark curves in the following chapters on tailored code designs. First, we introduce Shannon's sphere packing bound (SPB) of 1959. It provides a lower bound on



(a) BPSK (b) 4-ASK (c) 8-ASK

(d) 4-QAM (e) 16-QAM (f) 64-QAM

Figure 2.3.: Overview of typical coherent modulation formats. A binary reflected Gray code is used for the label of each signal point.

the FER of any spherical code, i.e., a code whose codewords lie on a spherical shell. We then describe Gallager's random coding bound (RCB) that was derived in Sec. 2.3.2 as part of the channel coding theorem. A tighter version of the RCB is the so called random coding union bound (RCUB) of [38, Sec. III-B]. The latter two bounds provide achievability results. Finally, we present a bound based on the normal approximation (NA).

Sphere Packing Bound

The SPB appears in [39] and assumes a spherical codebook, i.e., $\mathcal{C} = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x}\|^2 = P\}$ with $|\mathcal{C}| = 2^{nR}$. The bound owes its name to the steps used in its derivation. With each codeword $\boldsymbol{x}_i$, we associate a Voronoi region, defined as the convex set of points that are closer to $\boldsymbol{x}_i$ than to any of the other $2^{nR} - 1$ codewords of $\mathcal{C}$. Each Voronoi cell is formed by at most $2^{nR} - 1$ hyperplanes that go through the origin, as all codewords have the same distance from the origin. Hence, each Voronoi cell is a polyhedric cone, the cells subdivide the space $\mathbb{R}^n$, and the sum of their solid angles $\Omega_i, i = 1, \dots, 2^{nR}$ equals the surface area of the n-dimensional spherical shell.

An ML decoder identifies for a given receive sequence $\boldsymbol{y}$ the Voronoi region from which it originated. An error occurs if the received sequence falls outside the Voronoi region which corresponds to the transmitted signal point. To lower bound this probability, Shannon introduces the sphere packing argument, saying that a randomly picked Voronoi cell does not exhibit a better probability of error than a circular cone of the same solid angle. This claim is based on propositions stating that among the cones of a given solid angle, the circular one provides the lowest probability of error and it is best to share the total solid angle evenly between all Voronoi cells. The corresponding solid angle θ is therefore given as

$$\Omega(\theta) = 2^{-nR} \tag{2.116}$$

where $\Omega(\theta)$ is the solid angle of a spherical cap in n dimensions with half-angle θ

$$\Omega(\theta) = \frac{1}{2} I_{\sin^2(\theta)}\left(\frac{n-1}{2}, \frac{1}{2}\right) \tag{2.117}$$

and the function $I_x(a,b)$ is the regularized, incomplete beta function

$$I_x(a,b) \triangleq \frac{\int_0^x t^a (1-t)^b \, \mathrm{d}t}{\int_0^1 t^a (1-t)^b \, \mathrm{d}t}. \tag{2.118}$$

For numerical evaluations and determining θ, rewriting (2.116) as done in [40, Sec. II] is beneficial. Having the half angle θ, the probability can be lower bounded as

$$\Pr(\hat{W} \neq W) \geq \Pr\left(\frac{Z + \sqrt{P/\sigma^2}}{\sqrt{V/(n-1)}} \leq \sqrt{n-1}\cot(\theta)\right) \tag{2.119}$$

where Z is a Gaussian normal distributed RV and V is a χ^2 distributed RV with $(n-1)$ degrees of freedom. Further, the RV V is stochastically independent of Z. The CDF is implemented in many numerical libraries[7].

We note that the SPB as presented here does not take modulation constraints into account. Refined versions of the SPB can be found in [40, 41].

Random Coding Bound

The RCB can be obtained from the channel coding theorem of Sec. 2.3.2. It gives an upper bound on the FER as

$$\Pr(W \neq \hat{W}) \leq 2^{-nE(R)} \tag{2.120}$$

where $E(R)$ denotes the respective error exponent, e.g., (2.69) or (2.75).

[7] The resulting RV has a non-central t distribution. For instance, in Matlab, the CDF is implemented by the function `nctcdf`.

Random Coding Union Bound

As the name suggests, the RCUB is based on the same random coding arguments as Gallager's matched or mismatched RCB (see Sec. 2.3.2). For a general decoding metric $q : \mathcal{X}^n \times \mathbb{R}^n \to \mathbb{R}^+$, this bound is [38, Theorem 16][8]

$$\Pr(\hat{W} \neq W) \leq \mathrm{E}\left[\min\left(1, \Pr\left(\frac{q(X^n(\tilde{w}), Y^n)}{q(X^n, Y^n)} \geq 1 \middle| X^n, Y^n\right)\right)\right]. \tag{2.121}$$

The derivation starts from (2.60), applies the union bound (without introducing the tightening parameter ρ as in (2.62)) and uses the generalized Markov inequality. As the resulting expression represents a probability, we know that it is upper bounded by one, so that we have

$$\begin{aligned}
&\Pr\left(\bigcup_{\hat{w}\neq w_0}\left\{\frac{q(X^n(\tilde{w}), y^n)}{q(x^n(w_0), y^n)} \geq 1\right\}\middle| W = w_0, X^n = x^n(w_0), Y^n = y^n\right) && (2.122)\\
&\leq \min\left(1, (2^{nR}-1)\frac{\mathrm{E}\left[q(X^n(\tilde{w}), y^n)^s\right]}{q(x^n(w_0), y^n)^s}\right). && (2.123)
\end{aligned}$$

The improved tightening of the RCUB follows from the trivial upper bound 1. Unfortunately, this modification couples all n channel uses and the integrals for the averaging over $p_{Y|X}(y_i|x)$, $i = 1, \ldots, n$, can not be calculated separately. We use the saddlepoint approximation of [42] to obtain an approximation to the tail probability of the respective RVs.

Normal Approximation

The starting point for the derivation of the NA is (2.121) and (2.123). Additionally, we assume a memoryless decoding metric and set $s = 1$. It is

$$\begin{aligned}
\Pr(\hat{W} \neq W) &\leq \mathrm{E}\left[\min\left(1, 2^{nR-\sum_{i=1}^{n}\log_2\left(\frac{q(X_i(w_0),Y_i)}{\sum_{a\in\mathcal{X}} P_X(a)q(a,Y_i)}\right)}\right)\right] && (2.124)\\
&\leq 2^{-n\delta}\cdot\Pr\left(2^{nR-Z_n} \leq 2^{-n\delta}\right) + 1\cdot\Pr\left(2^{nR-Z_n} \geq 2^{-n\delta}\right) && (2.125)\\
&\leq 2^{-n\delta}\cdot 1 + \Pr\left(nR - Z_n \geq 2^{-n\delta}\right) && (2.126)\\
&\leq 2^{-n\delta} + \Pr\left(Z_n \leq nR - 2^{-n\delta}\right) && (2.127)
\end{aligned}$$

[8] We present a generalized version for the expression in Polyanskiy's paper for an arbitrary decoding metric q.

where we introduced $\delta > 0$, the RV

$$Z_n = \sum_{i=1}^{n} \log_2 \left(\frac{q(X_i(w_0), Y_i)}{\sum_{a \in \mathcal{X}} P_X(a) q(a, Y_i)} \right)$$

and used the bound

$$\mathrm{E}\left[U\right] = \int_{u \le a} u \cdot p_U(u)\, \mathrm{d}u + \int_{u \ge a} u \cdot p_U(u)\, \mathrm{d}u \le a \cdot \Pr(U \le a) + 1 \cdot \Pr(U \ge a)$$

for an RV U with $\mathrm{supp}(p_U) = [0, 1]$. To continue, we assume that Z_n is Gaussian distributed (*normal approximation*) which holds for $n \to \infty$ by the central limit theorem. By choosing δ appropriately (δ has to decrease slower than $1/n$ and faster than $1/\sqrt{n}$), we get

$$\Pr(\hat{W} \neq W) \approx Q\left(\frac{\mathrm{E}\left[Z_n\right] - nR}{\sqrt{\mathrm{Var}\left[Z_n\right]}} \right). \tag{2.128}$$

For instance, if we consider SMD and have $q(x, y) = p_{Y|X}(y|x)$, the mean and variance of Z_n becomes

$$\mathrm{E}\left[Z_n\right] = n\, \mathrm{E}\left[i(X;Y)\right] = n\, \mathrm{I}(X;Y) \tag{2.129}$$

$$\mathrm{Var}\left[Z_n\right] = n \left(\mathrm{E}\left[i(X;Y)^2 \right] - \mathrm{E}\left[i(X;Y)\right]^2 \right). \tag{2.130}$$

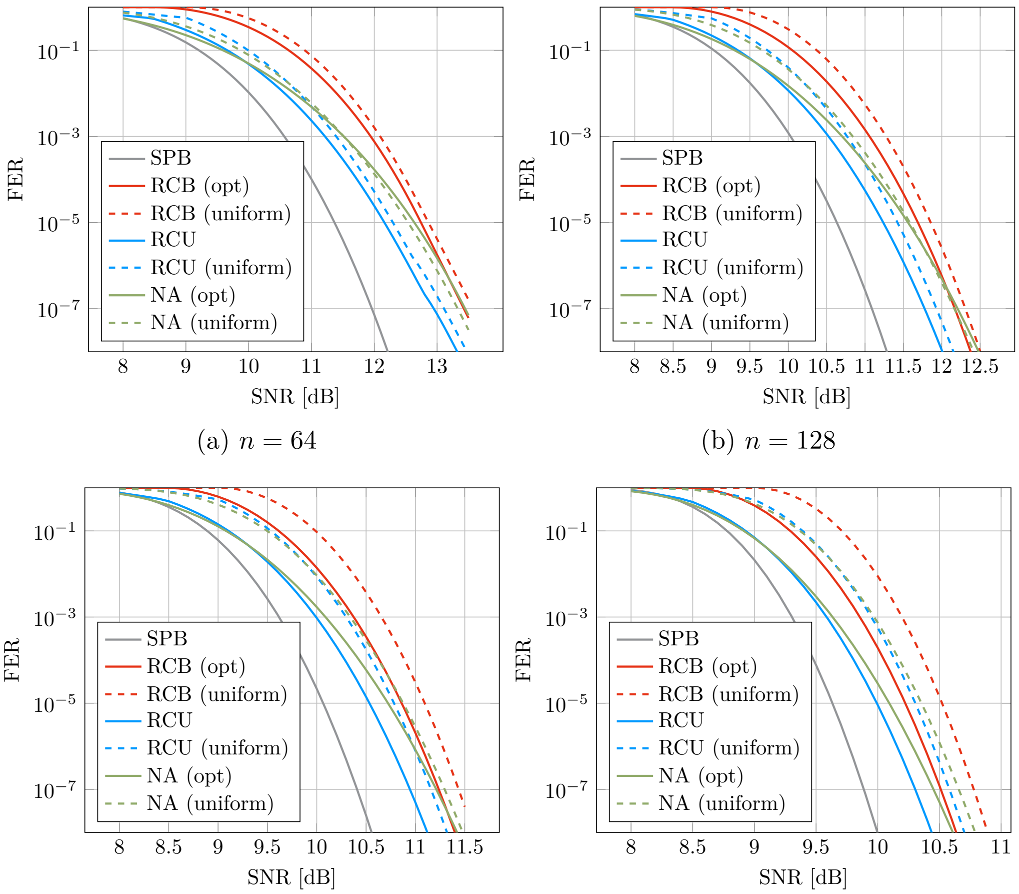
The latter term is called *dispersion*. The bound (2.128) can be tightened as shown in [29, 38].

Numerical Evaluations of the Finite Length Bounds

In Fig. 2.4 we show results of the numerical evaluation of the finite length coding bounds for 8-ASK, an SE of 1.5 bpcu and $n \in \{64, 128, 256\}$ channel uses. For all plots, the SMD decoding metric $q(x, y) = p_{Y|X}(y|x)$ is considered. We observe that the RCUB provides the tightest achievability bound for all considered blocklengths – it also outperforms the NA for low FERs.

In Fig. 2.5, we depict the same scenario as before for $n = 21\,600$ channel uses. As expected from theory, the converse and achievability bounds become tighter. The realized shaping gains according to the RCB is what the asymptotic information rate analysis predicts.

In Fig. 2.6, we compare Gallager's RCB for 4 and 8-ASK and an SE of 1.5 bpcu. We consider SMD with $q(x, y) = p_{Y|X}(y|x)$ and BMD with $q(x, y) = \prod_{k=1}^{m} p_{Y|B_k}(y|[\chi(x)]_k)$. For the latter, a BRGC label is employed. We see that the loss due to BMD for 8-ASK is reflected for both short ($n = 64$) and long blocks ($n = 21\,600$). The asymptotic BMD loss is 0.43 dB for 8-ASK and negligible for 4-ASK. The latter is because the target SE of 1.5 bpcu is close to the saturation region of 4-ASK (i.e., its high SNR regime). This is the regime where SMD and BMD become similar in their performance, see Fig. 3.1.



(a) $n = 64$

(b) $n = 128$

(c) $n = 256$

(d) $n = 512$

Figure 2.4.: Comparison of finite length bounds for 8-ASK with SMD, $R_{\text{tx}} = 1.5\,$bpcu and different numbers of channel uses. Note that the x-axis ranges reduce as n increases.

This suggests that the operation points of BMD transceivers must be chosen carefully for optimal performance. We further added the NA for 8-ASK for SMD and BMD. While we observe significant differences for the RCB and the NA for a small number of channel uses, both become more similar for large n. The relative differences between SMD and BMD are also revealed for the NA.

2.4. Graph Theory

2.4.1. Undirected and Bipartite Graphs

Many mathematical concepts and relations can be stated as graphs. An *undirected graph* G is a tuple $(\mathcal{U}, \mathcal{E})$ consisting of a set $\mathcal{U} = \{\mathtt{u}_1, \mathtt{u}_2, \ldots, \mathtt{u}_k\}$ of *nodes* (or vertices) and a set

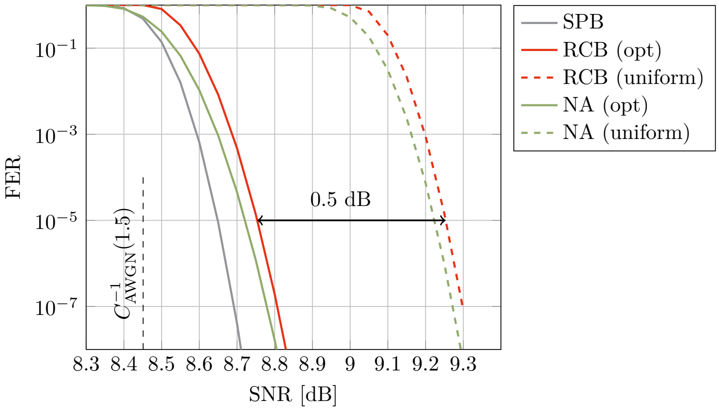
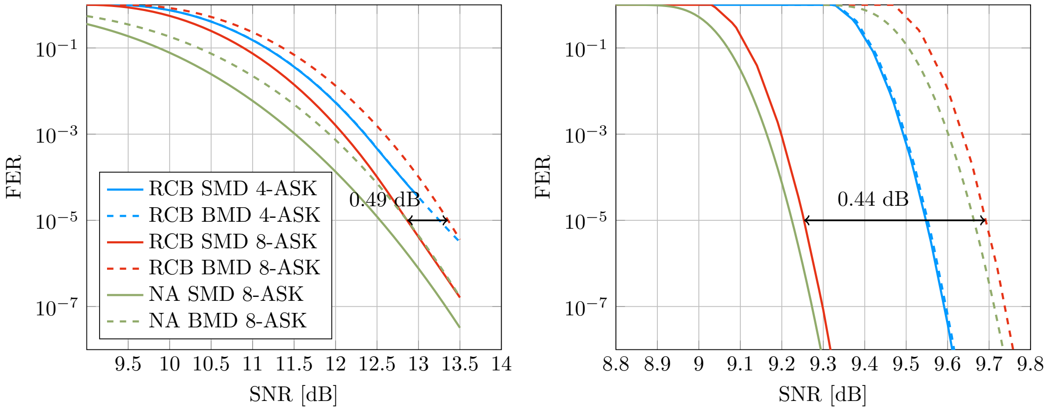


Figure 2.5.: Comparison of finite length bounds for 8-ASK with SMD, $R_{\mathrm{tx}} = 1.5$ bpcu and $n = 21\,600$ channel uses.



(a) $n = 64$

(b) $n = 21\,600$

Figure 2.6.: Comparison of the RCB and the NA for SMD and BMD for 4-ASK and 8-ASK with $R_{\mathrm{tx}} = 1.5$ bpcu.

$\mathcal{E} = \{\mathsf{e}_{ij}\}$ of *edges*. The set $\mathcal{E}$ contains e_{ij} if there is a connection between node u_i and node u_j. As the graph is undirected, the order of the indices $i, j \in \{1, 2, \dots, k\}$ does not matter. Further, each pairing of i, j is distinct and the requirement $i \neq j$ prevents loops.

A length ℓ *walk* is a sequence of ℓ nodes $\mathsf{u}_{i_1}, \mathsf{u}_{i_2}, \mathsf{u}_{i_3}, \dots, \mathsf{u}_{i_{\ell+1}}$ such that the edges $\mathsf{e}_{i_1 i_2}, \mathsf{e}_{i_2 i_3}, \mathsf{e}_{i_3 i_4}, \dots$ are in $\mathcal{E}$. A *path* is a walk where each node $\mathsf{u}_{i_1}, \dots, \mathsf{u}_{i_\ell}$ appears at most once. A length ℓ *cycle* is a path where the starting and ending nodes coincide, i.e., $\mathsf{u}_{i_1} = \mathsf{u}_{i_{\ell+1}}$. The length of the shortest cycle in a graph determines its *girth*.

Bipartite graphs belong to a special class of graphs, whose set $\mathcal{U}$ of nodes is split into two disjoint subsets $\mathcal{V}$ and $\mathcal{C}$ such that $\mathcal{U} = \mathcal{V} \cup \mathcal{C}$ and $\mathcal{V} \cap \mathcal{C} = \emptyset$, and edges are allowed to connect nodes from different node sets only, i.e., if $\mathsf{e}_{ij} \in \mathcal{E}$ then $\mathsf{v}_j \in \mathcal{V}$ and $\mathsf{c}_i \in \mathcal{C}$, $i = 1, \dots, |\mathcal{C}|$, $j = 1, \dots, |\mathcal{V}|$. We refer to $\mathcal{V}$ and $\mathcal{C}$ as the sets of VNs and factor nodes (FNs), respectively. The set $\mathcal{N}(\mathtt{v}_j)$ ($\mathcal{N}(\mathtt{c}_i)$) denotes the neighbors of VN $\mathtt{v}_j$ (FN $\mathtt{c}_i$), i.e.,

$$\mathcal{N}(\mathtt{v}_j) = \{\mathtt{c}_i \in \mathcal{C} : \mathtt{e}_{ij} \in \mathcal{E}\}, \tag{2.131}$$

$$\mathcal{N}(\mathtt{c}_i) = \{\mathtt{v}_j \in \mathcal{V} : \mathtt{e}_{ij} \in \mathcal{E}\}. \tag{2.132}$$

The girth of a bipartite graph is always an even number.

2.4.2. The Sum-Product Algorithm on Factor Graphs

Factor graphs are a special class of bipartite graphs that describe how a “global” function of many variables decouples into the product of many “local” functions [43]. For instance, the global function

$$f(v_1, v_2, v_3, v_4, v_5) = f_1(v_1, v_2) f_2(v_2, v_3) f_3(v_3, v_4) f_4(v_3, v_5) \tag{2.133}$$

is described by the factor graph of Fig. 2.7. The FNs $\mathtt{c}_1, \mathtt{c}_2, \mathtt{c}_3, \mathtt{c}_4$ are denoted by rectangular boxes and represent the local functions (factors) $f_1, f_2, \dots, f_4$. The VNs $\mathtt{v}_1, \mathtt{v}_2, \dots, \mathtt{v}_5$ have circular boxes and represent the arguments of the local functions. In all our examples, the values of the VNs come from a finite set $\mathbb{F}$.

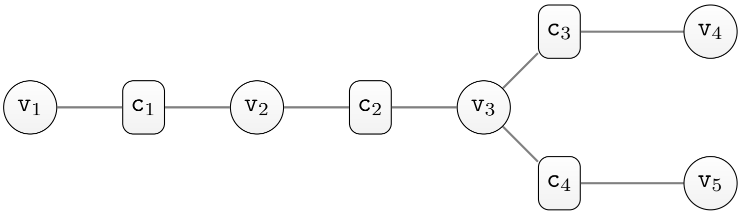


Figure 2.7.: Factor graph for the global function of (2.133).

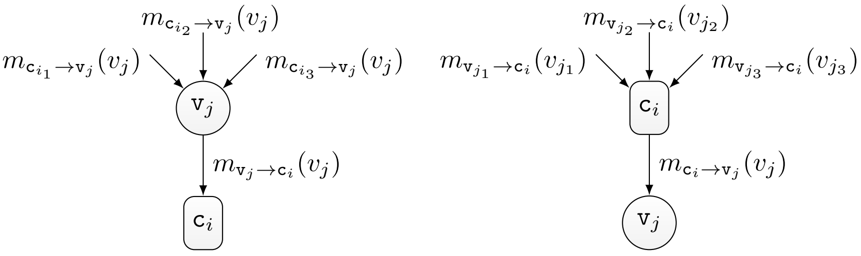
Suppose that we want to solve the marginalization problem

$$f(v_i) = \sum_{v_1 \in \mathbb{F}} \cdots \sum_{v_{i-1} \in \mathbb{F}} \sum_{v_{i+1} \in \mathbb{F}} \cdots \sum_{v_{|\mathcal{V}|} \in \mathbb{F}} f(v_1, v_2, \dots, v_{|\mathcal{V}|}) = \sum_{\sim v_i} f(v_1, v_2, \dots, v_{|\mathcal{V}|}) \tag{2.134}$$

where the notation $\sim v_i$ is short hand for the summation over the values of all VNs except for the i-th one. Naively, this implies summing over $2^{|\mathcal{V}|-1}$ values if the alphabet $\mathbb{F}$ of each VN is binary. By exploiting the factorization (2.133), the complexity can be reduced significantly by applying the *distributive law*. More precisely, we use the *Cartesian product distributive law* [44].

The *sum-product algorithm* (SPA) formalizes the application of the distributive law and uses the following building blocks at the VN and FN: For the VN update (Fig. 2.8a) we have

$$m_{\mathtt{v}_j \to \mathtt{c}_i}(v_j) = \prod_{\mathtt{c} \in \mathcal{N}(\mathtt{v}_j) \setminus \{\mathtt{c}_i\}} m_{\mathtt{c} \to \mathtt{v}_j}(v_j) \tag{2.135}$$


$m_{\mathtt{c}_{i_2}\to\mathtt{v}_j}(v_j)$
$m_{\mathtt{c}_{i_1}\to\mathtt{v}_j}(v_j)$
$m_{\mathtt{c}_{i_3}\to\mathtt{v}_j}(v_j)$
$\mathtt{v}_j$
$m_{\mathtt{v}_j\to\mathtt{c}_i}(v_j)$
$\mathtt{c}_i$
$m_{\mathtt{v}_{j_2}\to\mathtt{c}_i}(v_{j_2})$
$m_{\mathtt{v}_{j_1}\to\mathtt{c}_i}(v_{j_1})$
$m_{\mathtt{v}_{j_3}\to\mathtt{c}_i}(v_{j_3})$
$\mathtt{c}_i$
$m_{\mathtt{c}_i\to\mathtt{v}_j}(v_j)$
$\mathtt{v}_j$

(a) VN to FN

(b) FN to VN

Figure 2.8.: Update rules of the SPA.

while the FN update is given by (Fig. 2.8b)

$$m_{\mathtt{c}_i\to\mathtt{v}_j}(v_j) = \sum_{\sim v_j} f_i(\mathcal{N}(\mathtt{c}_i)) \prod_{\mathtt{v}\in\mathcal{N}(\mathtt{c}_i)\setminus\{\mathtt{v}_j\}} m_{\mathtt{v}\to\mathtt{c}_i}(v_j). \tag{2.136}$$

Example 4. We want to calculate $f(v_1)$ for the factor graph in Fig. 2.7 and use the update rules of (2.135) and (2.136) to obtain:

$$\begin{aligned}
m_{\mathtt{c}_3\to\mathtt{v}_3}(v_3) &= \sum_{v_4\in\mathbb{F}} f_3(v_3,v_4)\cdot m_{\mathtt{v}_4\to\mathtt{c}_3}(v_4)\\
m_{\mathtt{c}_4\to\mathtt{v}_3}(v_3) &= \sum_{v_5\in\mathbb{F}} f_4(v_3,v_5)\cdot m_{\mathtt{v}_5\to\mathtt{c}_3}(v_5)\\
m_{\mathtt{v}_3\to\mathtt{c}_2}(v_3) &= m_{\mathtt{c}_3\to\mathtt{v}_3}(v_3)\cdot m_{\mathtt{c}_4\to\mathtt{v}_3}(v_3)\\
m_{\mathtt{c}_2\to\mathtt{v}_2}(v_2) &= \sum_{v_3\in\mathbb{F}} f_2(v_2,v_3)\cdot m_{\mathtt{v}_3\to\mathtt{c}_2}(v_3)\\
m_{\mathtt{v}_2\to\mathtt{c}_1}(v_2) &= m_{\mathtt{c}_2\to\mathtt{v}_2}(v_2)\\
m_{\mathtt{c}_1\to\mathtt{v}_1}(v_1) &= \sum_{v_2\in\mathbb{F}} f_2(v_1,v_2)\cdot m_{\mathtt{v}_2\to\mathtt{c}_1}(v_2).
\end{aligned}$$

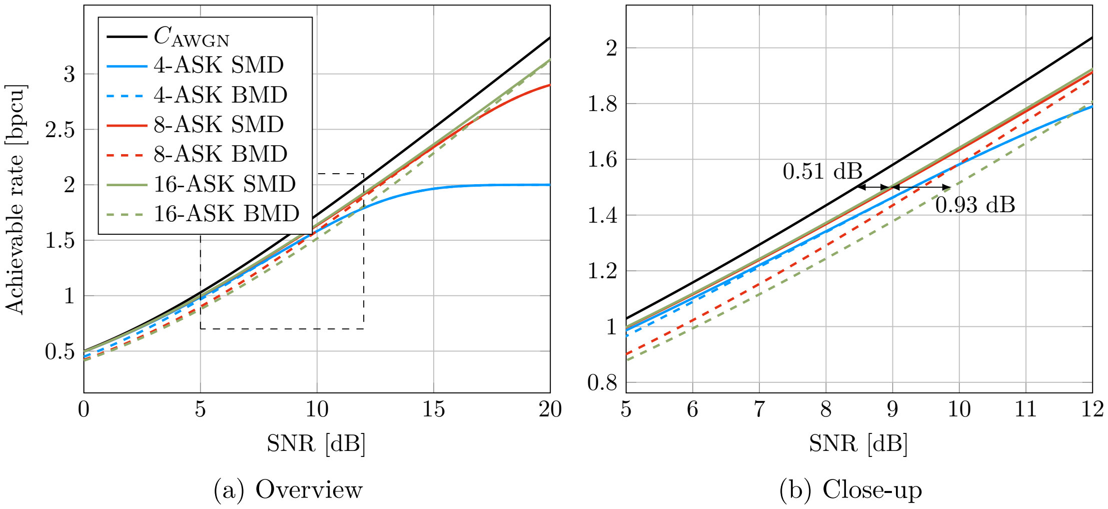
Overall, we get

$$\begin{aligned}
f(v_1) = m_{\mathtt{c}_1\to\mathtt{v}_1}(v_1) &= \sum_{v_2\in\mathbb{F}} f_2(v_1,v_2) m_{\mathtt{v}_2\to\mathtt{c}_1}(v_2)\\
&= \sum_{v_2\in\mathbb{F}} f_2(v_1,v_2)\left(\sum_{v_3\in\mathbb{F}} f_2(v_2,v_3)\left(\sum_{v_4\in\mathbb{F}} f_3(v_3,v_4) m_{\mathtt{v}_4\to\mathtt{c}_3}(v_4)\right)\cdot\right.\\
&\qquad\left.\left(\sum_{v_5\in\mathbb{F}} f_4(v_3,v_5) m_{\mathtt{v}_5\to\mathtt{c}_3}(v_5)\right)\right).
\end{aligned}$$

3

Probabilistic Shaping

3.1. Introduction and Historic Overview



(a) Overview

(b) Close-up

Figure 3.1.: Information rates of uniform signaling with 4-ASK, 8-ASK and 16-ASK and their comparison to the AWGN capacity. The solid curves are SMD rates, the dashed ones are BMD rates.

In what follows, the AWGN channel with discrete inputs and an average power constraint is of central interest. Its information theoretic model for n channel uses is described by

$$Y_i = \Delta X_i + N_i, \qquad \mathrm{E}\left[(\Delta X_i)^2\right] \leq P, \qquad i = 1, \ldots, n \tag{3.1}$$

where $X_i \in \mathcal{X}$ is the discrete channel input, $\Delta \in \mathbb{R}^+$ is the constellation scaling and N_i is zero mean Gaussian noise with variance σ^2 and PDF given in (2.93). We define the SNR as $\mathrm{E}\left[X^2\right]/\sigma^2$ and drop the time index i whenever possible.

The model (3.1) has been studied in detail in the literature, as it can be used to model various practical communication scenarios. To achieve the capacity of the AWGN channel, a codebook with Gaussian distributed signal points is necessary, see Sec. 2.3.4. For practical implementations, such a codebook is not feasible because of its complexity regarding storage and analog-to-digital converter (ADC)/digital-to-analog converter (DAC) requirements. While this was not much of an issue in the early days of digital communications, where channels were mostly power-limited and had a large bandwidth, things changed when applications were facing band-limited channels and demanded higher SEs. One of the first examples were voice band modems for telephone channels [45], which had only 300 Hz to 3000 Hz of usable spectrum, but offered SNRs of up to 28 dB. While PSK constellations were used mostly in the early 60s, QAM formats started to emerge in 1971, where the Codex 9600 C used 16-QAM in the V.29 standard to transmit 4 bpcu uncoded. Hereby, each constellation point was used with the same probability.

From the very beginning, researchers were well aware that this kind of signaling was not optimal and incurred a loss compared to the capacity of the channel because of the non-Gaussian signaling. This is illustrated in Fig. 3.1, where the achievable rates of 4-ASK, 8-ASK and 16-ASK are shown and compared to the Shannon limit C_{AWGN} (2.96). For high SNRs and for $M \to \infty$, the loss between signaling with discrete equi-spaced and uniformly distributed constellation points and a Gaussian codebook amounts to 1.53 dB [45].

To mitigate this problem, researchers developed *PS* approaches, that use constellation points with different probability. However, these approaches were not considered practical. For instance, in [45], the authors suggest to use a prefix-free code (e.g., inverse Huffmann coding) to parse the data bits into chunks and map those to respective constellation points, but point out that the variable length of the outputs may lead to error propagation, overflow and delay. It is further important to mention that shaping was not considered as crucial at this point, as much larger gains could be obtained through improved coding schemes.

One of those was Gottfried Ungerböck's trellis coded modulation (TCM). In his seminal work "Channel Coding with Multilevel/Phase Signals" [46], Ungerböck showed that the efficiency of practical transceivers can be improved substantially if the FEC is designed jointly with the modulation scheme. This allowed to achieve coding gains for higher-order modulation that were comparable to those obtained previously for the power-limited case [47]. TCM built on the notion that FEC needs to consider the Euclidean distance of signal points and should enlarge it. This led to the notion of "coding by set partitioning", where the coded bits selected a partition of the constellation and the uncoded bits selected the points within this partition.

To operate at the ultimate limit, PS and FEC should be combined. However, as many works showed, the combination of PS and FEC was perceived as difficult – especially when modern FEC codes (e.g., LDPC) should be considered.

3.2. Probabilistic Shaping with Forward Error Correction

Predominantly, two approaches have been considered in the literature to combine FEC and PS. They are depicted in Fig. 3.2 and Fig. 3.3 and will be discussed in the following two subsections.

3.2.1. Shaping as an Inner Code

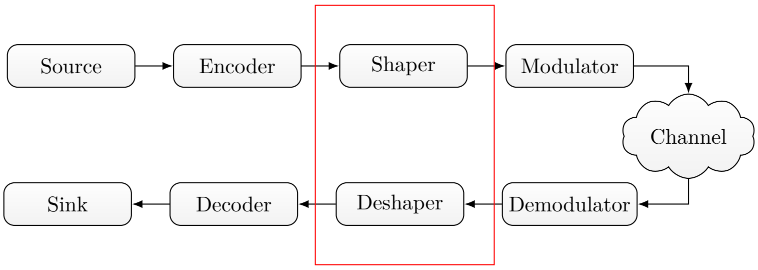


Figure 3.2.: Shaping as an inner code.

The optimal distribution must be realized at the channel input where the power constraint applies. Therefore, a natural choice is to place the shaping as an inner code (i.e., after FEC encoding) as shown in Fig. 3.2. However, this approach has an important drawback: The “inverse” shaping operation must be performed before (or jointly with) the FEC decoder. This may lead to severe error propagation and large complexity.

A practical approach was proposed by Gallager in [21, Sec. 6.2] and is now commonly referred to as *many-to-one-mapping*. The idea is to use a deterministic mapping function to assign several binary sequences of length m bits (after FEC encoding) to one channel input symbol. A comprehensive summary is given in [48]. While this solves the PS problem at the transmitter side, the receiver now has to deal with both the decoding and deshaping. Gallager notes that “Unfortunately, the problem of finding decoding algorithms is not so simple” [21, Sec. 6.2]. Still, many recent works [49, 50, 51, 52, 53] picked up this scheme and employ iterative demapping, i.e., iterations between the SD demapper and SD FEC decoder to resolve the ambiguities. Another disadvantage is that this scheme is inflexible in terms of the realized output distribution and therefore transmission rate, as only distributions of the form $P_X(x) \propto 1/2^m$ can be realized and additional steps are needed to match $P_X(x)$ to the capacity achieving distribution [54]. Additionally, the need for joint demapping/decoding comes at the price of reduced flexibility and complicated rate adaptation. Last but not least, shaping as an inner code generally requires a lower FEC code rate, which is undesired for high data rates – the throughput that needs to be supported by the digital signal processing (DSP) chip must be significantly larger in this case.

Another type of shaping is Trellis shaping [55], [56, Sec. IV]. This method overcomes the problem of error propagation and solves the decoding issues raised by Gallager. For this, a dedicated shaping code uses energy minimizing sequences only. The sequences are found via a modified Viterbi algorithm in the shaping code trellis. A simple four state convolutional shaping code may achieve shaping gains of up to 1 dB. Trellis shaping was also considered in the context of the V.34 modem standard, but was perceived as too complex compared to shell mapping (SM) [56, Ch. 4], [57].

3.2.2. Shaping as an Outer Code

An alternative strategy is pursued in Fig. 3.3. Here, the shaping is performed as an outer code (i.e., before the FEC encoding) in such a way that the desired properties of the shaping are not destroyed. Schemes of this kind are said to employ *reverse concatenation* and originate from requirements in magnetic and optical data storage where certain sequences are forbidden [58, 59, 60, 61]. In [62], the authors built on this principle and introduce the concept of *sparse-dense transmission*. The term "sparse-dense" hereby reflects the composition of a FEC codeword with sparse (ones and zeros are not equally distributed) and dense parts (zeros and ones are approximately equally distributed). The sparse part is realized with appropriate mapping techniques (e.g., look-up tables) and maintained in the FEC codeword by systematic encoding. In general, any communication scheme using this approach operates in a time sharing (TS) fashion as only a fraction of the codeword symbols are shaped.

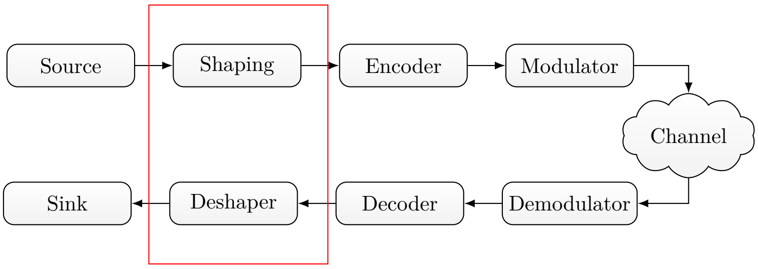


Figure 3.3.: Shaping as an outer code.

3.3. Layered Probabilistic Shaping

In the classical random coding setup of Sec. 2.3.2, the FEC codebook was created at random using the distribution P_X so that all transmitted codewords have the optimal distribution for the considered channel. Practical codes, e.g., linear block codes as introduced in Sec. 2.3.5, however, have code symbols that are (approximately) uniformly distributed. To achieve a shaping gain, we should transmit codewords from a subset of the code, which is chosen such that the code symbols of these codewords have the desired distribution. The problem of reliably transmitting information over a noisy channel in a power efficient manner is thereby decomposed into *layers* which can be tackled independently from each other.

First, the *shaping layer* has the task to encode into a subset of the FEC code. The *FEC layer* has the task to recover the transmitted sequence from the noisy channel observations. For this, the decoder uses a decoding metric that evaluates all codewords in the codebook and exploits the knowledge about the prior of the possibly transmitted codeword symbol.

For the analysis of achievable rates for a layered PS scheme, we first consider both layers as being independent of each other.

3.3.1. Forward Error Correction Layer

To analyze the FEC layer, we consider the following random coding experiment [63, Appendix B]: We create a codebook $\mathcal{C}$ with 2^{nmR_c} codewords and $0 \leq R_\text{c} \leq 1$. Each entry of a codeword is chosen at random and uniformly from the set $\mathcal{X}$, i.e., $P_X(x) = 1/|\mathcal{X}|$, with $|\mathcal{X}| = 2^m$.

The decoder uses the decoding metric $q : \mathcal{X} \times \mathbb{R} \to \mathbb{R}^+$ to determine a decision for the transmitted sequence. We follow similar steps as in Sec. 2.3.2 to obtain an upper bound on $\Pr(\hat{W} \neq W | W = w_0, X^n = x^n, Y^n = y^n)$, that is

$$\begin{aligned}\Pr(\hat{W} \neq W | W = w_0, X^n = x^n, Y^n = y^n) &\leq 2^{nmR_\text{c}} \frac{\mathrm{E}\left[q(X^n, y^n)\right]}{q(x^n, y^n)} &(3.2)\\ &= 2^{nmR_\text{c}} 2^{\log_2\left(\frac{\mathrm{E}[q(X^n,y^n)]}{q(x^n,y^n)}\right)} &(3.3)\\ &= 2^{nmR_\text{c}} 2^{-\log_2\left(\frac{q(x^n,y^n)}{\mathrm{E}[q(X^n,y^n)]}\right)}. &(3.4)\end{aligned}$$

For a memoryless metric, the probability (3.4) goes to zero if

$$R_\text{c} < \frac{1}{mn}\sum_{i=1}^{n} \log_2\left(\frac{q(x_i, y_i)}{\mathrm{E}\left[q(X, y_i)\right]}\right) = \frac{1}{mn}\sum_{i=1}^{n} \log_2\left(\frac{q(x_i, y_i)}{\sum_{a\in\mathcal{X}} q(a, y_i)\frac{1}{|\mathcal{X}|}}\right). \tag{3.5}$$

For $n \to \infty$ the right hand side of (3.5) converges to

$$\frac{1}{m}\mathrm{E}\left[\log_2\left(\frac{q(X,Y)}{\sum_{a\in\mathcal{X}} q(a,Y)\frac{1}{|\mathcal{X}|}}\right)\right] = 1 - \frac{1}{m}\underbrace{\mathrm{E}\left[-\log_2\left(\frac{q(X,Y)}{\sum_{a\in\mathcal{X}} q(a,Y)}\right)\right]}_{\mathrm{U}(q)} = 1 - \frac{1}{m}\mathrm{U}(q) \tag{3.6}$$

by the WLLN (2.34). We refer to the underbraced term as the *uncertainty* $\mathrm{U}(q)$ and note that it takes the form of a cross entropy (2.39). We can now instantiate (3.6) for SMD and BMD. In contrast to before, we explicitly include the knowledge about the distribution that is used for the shaping layer in the decoding metric. The SMD metric is

$$q_\text{SMD}(x, y) = P_{X|Y}(x|y) \propto p_{Y|X}(y|x)P_X(x) \tag{3.7}$$

and we have

$$q_{\text{BMD}}(x,y) = \prod_{k=1}^{m} P_{B_k|Y}([\chi(x)]_k|y) \propto \prod_{k=1}^{m} p_{Y|B_k}(y|b_k)P_{B_k}(b_k) \tag{3.8}$$

for BMD where

$$P_{B_k|Y}(b|y) = \frac{P_{B_kY}(b,y)}{p_Y(y)} = \frac{\sum_{x\in\mathcal{X}_k^b} p_{Y|X}(y|x)P_X(x)}{p_Y(y)} \tag{3.9}$$

and $\mathcal{X}_k^b = \{x \in \mathcal{X} : [\chi(x)]_k = b\}$. The resulting mismatched uncertainty expressions are

$$\mathrm{U}(q_{\text{SMD}}) = \mathrm{E}\left[-\log_2\left(\frac{P_{X|Y}(X|Y)}{\sum_{a\in\mathcal{X}} P_{X|Y}(a|Y)}\right)\right] = \mathrm{H}(X|Y) \tag{3.10}$$

$$\mathrm{U}(q_{\text{BMD}}) = \mathrm{E}\left[-\log_2\left(\frac{\prod_{k=1}^{m} P_{B_k|Y}(b_k|y)}{\sum_{a\in\mathcal{X}} \prod_{k=1}^{m} P_{B_k|Y}([\chi(a)]_k|y)}\right)\right] = \sum_{k=1}^{m} \mathrm{H}(B_k|Y). \tag{3.11}$$

In (3.10), the mismatched uncertainty becomes a *matched uncertainty*, i.e., a conditional entropy.

For BMD, one often prefers a representation in the logarithmic domain and defines the value

$$l_k = \log\left(\frac{P_{B_k|Y}(0|y)}{P_{B_k|Y}(1|y)}\right) \quad \text{such that} \quad P_{B_k|Y}(b|y) = \frac{\mathrm{e}^{l_k(1-b)}}{1+\mathrm{e}^{l_k}}. \tag{3.12}$$

Using this, we can rewrite (3.11) as

$$\begin{aligned}\mathrm{U}(q_{\text{BMD}}) &= \sum_{k=1}^{m} \mathrm{E}\left[-\log_2\left(\frac{\mathrm{e}^{L_k(1-B_k)}}{1+\mathrm{e}^{L_k}}\right)\right] = \sum_{k=1}^{m} \mathrm{E}\left[\log_2\left(\frac{1+\mathrm{e}^{L_k}}{\mathrm{e}^{L_k(1-B_k)}}\right)\right] \\ &= \sum_{k=1}^{m} \mathrm{E}\left[\log_2\left(1+\mathrm{e}^{-L_k\cdot(1-2B_k)}\right)\right]. \end{aligned} \tag{3.13}$$

3.3.2. Shaping Layer

The task of the shaping layer is to encode into a *shaping set* $\mathcal{S}$ which contains all sequences with the desired properties, i.e., for the AWGN channel, the shaping set may contain sequences with the lowest energy. For the FEC encoded codeword $\boldsymbol{v}$, we have to ensure that $\boldsymbol{v} \in \mathcal{S} \cap \mathcal{C}$. As outlined in detail in [63, Sec. IV-A][1], successful shaping set encoding is possible if

$$R_{\text{tx}} < \left[\frac{\log_2(|\mathcal{S}|)}{n} - m(1-R_{\text{c}})\right]^{+} \tag{3.14}$$

[1] The paper introduces additional rates, namely the shaping rate R_{ps} with $0 \leq R_{\text{ps}} \leq 1$ and the FEC code rate R_{fec} which corresponds to R_{c} in this thesis. We have $R_{\text{tx}} = mR_{\text{ps}}R_{\text{fec}}$.

where $[\cdot]^+ = \max(0, \cdot)$. For a *constant-composition shaping set* $\mathcal{S}$ (see Sec. 3.4.2) we have

$$\lim_{n\to\infty} \frac{\log_2(|\mathcal{S}|)}{n} = \mathrm{H}(X) \tag{3.15}$$

and (3.14) becomes

$$R_{\text{tx}} < [\mathrm{H}(X) - m(1 - R_{\text{c}})]^+ = [mR_{\text{c}} - \mathrm{D}(P_X \parallel P_U)]^+ . \tag{3.16}$$

where P_U is the discrete, uniform distribution on $\mathcal{X}$, i.e., $P_U(u) = 1/|\mathcal{X}|, \forall u \in \mathcal{X}$. Using (3.5) and (3.6) in (3.16), we have

$$R_{\text{tx}} < [\mathrm{H}(X) - \mathrm{U}(q)]^+ . \tag{3.17}$$

For SMD, we instantiate (3.17) with (3.10) and obtain

$$R_{\text{tx}} < R_{\text{SMD}} = \mathrm{I}(X;Y). \tag{3.18}$$

For BMD, we instantiate (3.17) with (3.11) and obtain

$$R_{\text{tx}} < R_{\text{BMD}} = \left[\mathrm{H}(X) - \sum_{k=1}^{m} \mathrm{H}(B_k|Y)\right]^+ . \tag{3.19}$$

The expression (3.19) was first stated in [8] and was derived by a random coding argument and a typicality decoder. We denote the inverses of (3.18) and (3.19) by R_{SMD}^{-1} and R_{BMD}^{-1}, respectively. For many scenarios, a practically feasible implementation of the shaping layer constitutes the difficult part for a shaping scheme based on reverse concatenation.

3.3.3. Optimum Input Distribution

To find the optimal PMF P_X, we solve the following optimization problem:

$$R^\star_{\text{SMD/BMD}} = \max_{P_X, \Delta} \quad R_{\text{SMD/BMD}} \quad \text{subject to} \quad \mathrm{E}\left[(\Delta X)^2\right] \leq P. \tag{3.20}$$

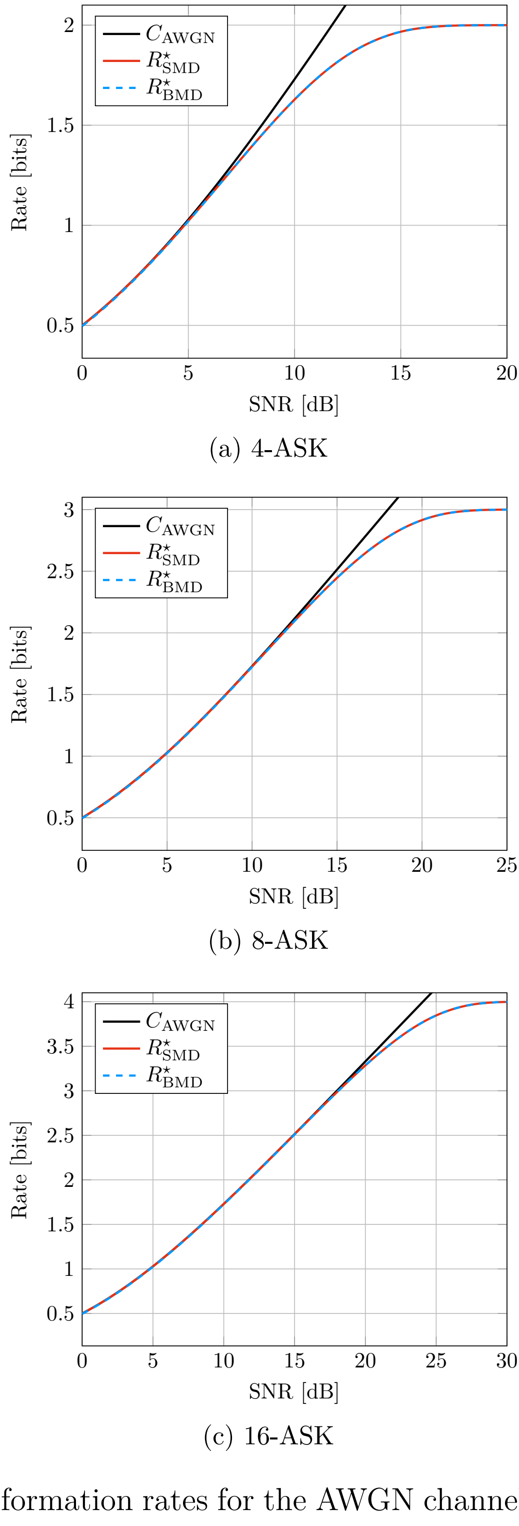
For practical implementations, a parametric description of P_X is desirable. In [64], the authors introduced the family of Maxwell-Boltzmann (MB) distributions of the form

$$P_X(x;\nu) = \frac{\exp(-\nu x^2)}{\sum_{a\in\mathcal{X}} \exp(-\nu a^2)} \tag{3.21}$$

where $\nu \in \mathbb{R}$. For $\nu = 0$, the MB distribution degrades to the uniform distribution on $\mathcal{X}$ and converges to a distribution with a support of two points for $\nu \to \infty$.

The MB distribution arises naturally for power-efficient communication[2] as it is the

[2] This can be seen if (3.20) is considered for $R_{\text{SMD}} = \mathrm{I}(X;Y)$ for high SNRs, when $\mathrm{H}(X|Y)$ vanishes.



(a) 4-ASK

(b) 8-ASK

(c) 16-ASK

Figure 3.4.: Optimized information rates for the AWGN channel and SMD and BMD metrics. $R^{\star}_{\text{SMD}}$ and $R^{\star}_{\text{BMD}}$ basically lie on top of each other.

solution of the optimization problem

$$\max_{P_X} \quad \mathrm{H}(X) \quad \text{subject to} \quad \mathrm{E}\left[X^2\right] \leq P. \tag{3.22}$$

Therefore, the MB distribution maximizes the entropy of the channel input subject to an average power constraint. In a dual formulation, one can show that the solution of (3.22) is the same as for

$$\min_{P_X} \quad \mathrm{E}\left[X^2\right] \quad \text{subject to} \quad \mathrm{H}(X) \geq R. \tag{3.23}$$

That is, the MB distribution minimizes the average energy subject to an entropy constraint. If we optimize over the family of MB distributions only, we have

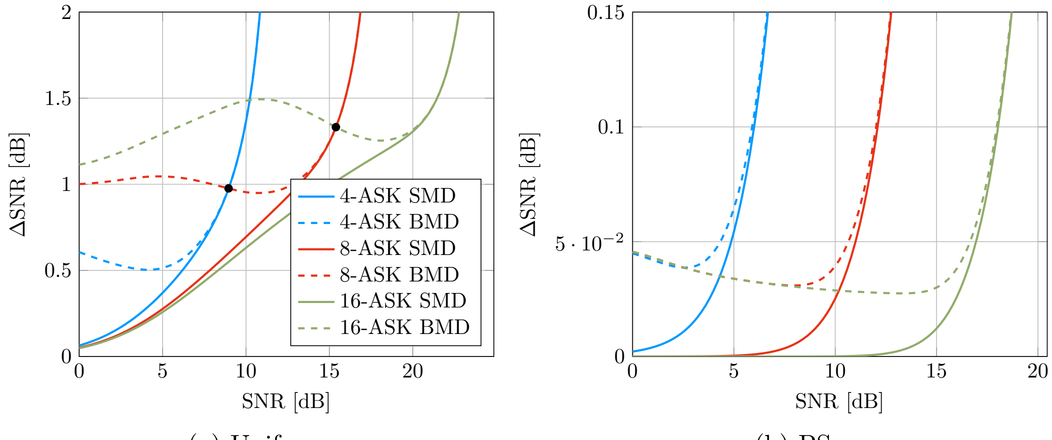
$$R^{\mathrm{MB},\star}_{\mathrm{SMD/BMD}} = \max_{\nu,\Delta} \quad R_{\mathrm{SMD/BMD}} \quad \text{subject to} \quad \mathrm{E}\left[(\Delta X)^2\right] \leq P. \tag{3.24}$$

The result of the optimization (3.20) is shown in Fig. 3.4. First, we see that the optimized information rates are virtually the same as the AWGN capacity (dashed, black curve). Surprisingly, there is no obvious gap between BMD and SMD, whereas we could observe significant losses for BMD in the uniform case, see Fig. 3.1.

Another perspective on these results is shown in Fig. 3.5 where the gap to the AWGN capacity is shown. The gap is defined as

$$\Delta\mathrm{SNR} = R^{\star,-1}_{\mathrm{SMD/BMD}}(C_{\mathrm{AWGN}}(\mathrm{SNR})) - \mathrm{SNR}. \tag{3.25}$$

For PS and SMD, we observe in Fig. 3.5b that the gap to capacity is vanishingly small. For BMD, it is smaller than 0.05 dB for meaningful operating regimes. In contrast, Fig. 3.5a shows the same scenario for uniform signaling. Here, the results for BMD are particularly



(a) Uniform

(b) PS

Figure 3.5.: Gap to AWGN capacity for SMD and BMD.

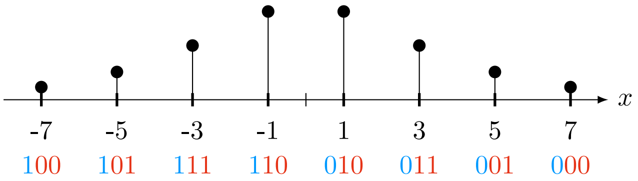


Figure 3.6.: 8-ASK constellation with a BRGC labeling. Bit-level one (blue) represents sign bits (distinguishing between the negative and positive side), whereas bit levels two and three (red) denote the bits representing the amplitude values 1, 3, 5 and 7.

interesting as they indicate clear operating regions for each constellation size. The SNRs points where a switch to the next higher constellation size should be performed are indicated with a dot. We see that 8-ASK should be used for SNRs higher than about 9 dB and 16-ASK for SNRs higher than 15.4 dB. For SMD, we see that increasing the constellation order generally decreases the gap to capacity for a given SNR. However, this requires to use a lower rate FEC code to operate at the same SE, which is often not desired.

3.4. Implementation of Layered PS: Probabilistic Amplitude Shaping

3.4.1. Foundations of Probabilistic Amplitude Shaping

PAS is a practically relevant instance of a layered PS scheme suited for the average power constrained AWGN channel. It exploits the symmetry property of the optimal input distribution such that the suboptimality of sparse-dense transmission is circumvented, see Sec. 3.2.2. PAS relies on the following three requirements and principles:

1. The capacity achieving distribution P_X^* is symmetric, i.e., it allows a factorization as $X = A \cdot S$, where the RVs A and S denote the amplitude and sign parts, respectively, and we have
$$P_X(x) = P_A(|x|) \cdot P_S(\text{sign}(x)) \tag{3.26}$$
where the alphabet of X is $\mathcal{X} = \mathcal{A} \times \mathcal{S}$ with $\mathcal{A} = \{1, 3, \ldots, M-1\}$ and $\mathcal{S} = \{-1, +1\}$.

   This property is fulfilled for the AWGN channel with an average power constraint (see [65, Proposition 2.3]). The PMF P_A is non-uniform on $\mathcal{A}$, whereas P_S is uniform on its binary support $\mathcal{S}$.

2. A systematic generator matrix $\boldsymbol{G} = (\boldsymbol{I} \quad \boldsymbol{P})$ is used for encoding.

3. A distribution matcher (DM) [66] generates a sequence of symbols with a specified distribution.

The transmitter component side of PAS is illustrated in Fig. 3.7 and is summarized as follows: A number k_{dm} of uniformly distributed source bits are converted by a one-to-one, fixed-to-fixed length DM of rate k_{dm}/n to a sequence of n shaped amplitude values. A bit mapping function

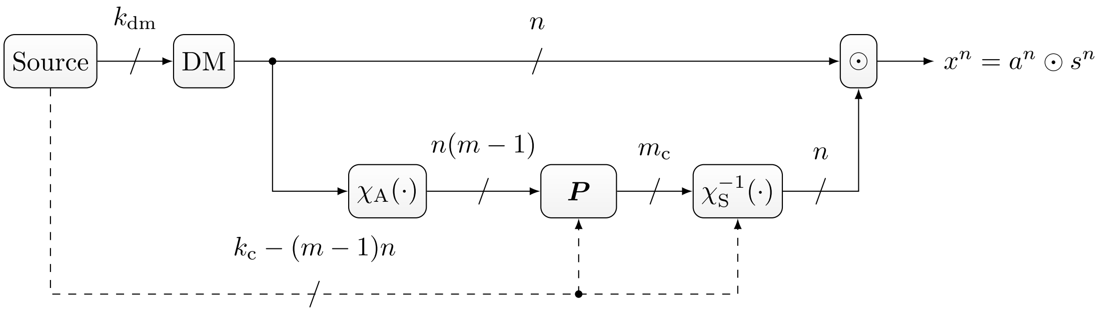


Figure 3.7.: Encoding procedure for PAS.

$$\chi_{\mathrm{A}} : \mathcal{A} \to \{0,1\}^{m-1} \tag{3.27}$$

maps each amplitude a_i of the amplitude sequence $a^n = (a_1, a_2, \dots, a_n)$ to its binary representation. For example, in Fig. 3.6, we have $\chi_{\mathrm{A}}(3) = (1,1)$. The binary representation of a^n has length $n(m-1)$ and is encoded by a systematic generator matrix of a code with rate $R_{\mathrm{c}} = (m-1)/m$ such that n parity bits are generated. The distribution in the systematic part is left unchanged, whereas the n parity bits are approximately uniformly distributed (as a modulo-2 sum of many bits). We refer to this as the *uniform check bit assumption* [9, Fig. 2]. Consequently, the parity bits can be used as sign bits using the inverse of the sign mapping function

$$\chi_{\mathrm{S}} : \{-1,+1\} \to \{1,0\}. \tag{3.28}$$

The final transmit sequence x^n is obtained after a componentwise multiplication of the amplitude sequence with the sign sequence.

A generalization of the scheme for code rates $R_{\mathrm{c}} > (m-1)/m$ is indicated by the dashed line in Fig. 3.7. If the code rate is higher than $(m-1)/m$, the encoding produces less than n parity (sign) bits. To compensate, we can use some of the systematically encoded information bits as additional signs bits. This procedure is referred to as *extended PAS*.

The transmission rate of the extended PAS scheme is

$$\begin{aligned} R_{\mathrm{tx}} &= \frac{k_{\mathrm{dm}} + k_{\mathrm{c}} - (m-1)n}{n} = \frac{k_{\mathrm{dm}}}{n} + \frac{R_{\mathrm{c}} n_{\mathrm{c}} - (m-1)n}{n} \\ &= \frac{k_{\mathrm{dm}}}{n} + 1 - (1 - R_{\mathrm{c}}) \cdot m \qquad \text{[bits/channel use (bpcu)]}. \end{aligned} \tag{3.29}$$

In the last step we assumed $n_{\mathrm{c}} = n \cdot m$. The term

$$\gamma = 1 - (1 - R_{\mathrm{c}}) \cdot m \tag{3.30}$$

denotes the fraction of sign bits (i.e., those of bit level one) that are used as additional information bits in the extended PAS scheme.

3.4.2. Distribution Matcher Algorithms

Blackbox Description

A fixed-to-fixed length DM is a function $f_{\mathrm{dm}} : \{0,1\}^{k_{\mathrm{dm}}} \to \mathcal{C}_{\mathrm{dm}} \subseteq \mathcal{A}^n$ that takes an input sequence of k_{dm} bits and maps those in a one-to-one fashion[3] to a length n output sequence. For PAS, the shaping set $\mathcal{S}$ (see Sec. 3.3) is

$$\mathcal{S} = \mathcal{C}_{\mathrm{dm}} \times \{-1,+1\}^n \tag{3.31}$$

with cardinality $|\mathcal{S}| = 2^{k_{\mathrm{dm}}} \cdot 2^n = 2^{k_{\mathrm{dm}}+n}$. The symbols of the output sequence are taken from a set $\mathcal{A}$. Let $n_{a_i}(a^n), i = 1, \dots, |\mathcal{A}|$ denote the number of occurrences of the symbol $a_i \in \mathcal{A}$ in the sequence a^n. The symbols in the set $\mathcal{C}_{\mathrm{dm}}$ have the empirical distribution

$$P_{\hat{A}}(a_i) = \frac{1}{|\mathcal{C}_{\mathrm{dm}}|} \sum_{a^n \in \mathcal{C}_{\mathrm{dm}}} \frac{n_{a_i}(a^n)}{n}. \tag{3.32}$$

Depending on the concrete implementation, the output distribution is a parameter of the DM.

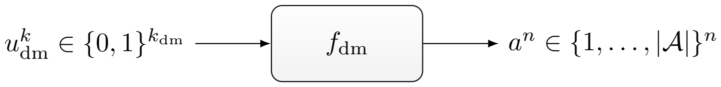

$u_{\mathrm{dm}}^k \in \{0,1\}^{k_{\mathrm{dm}}}$
f_{dm}
$a^n \in \{1,\dots,|\mathcal{A}|\}^n$

Figure 3.8.: Blackbox description of a DM.

The DM rate is given by

$$R_{\mathrm{dm}} = \frac{k_{\mathrm{dm}}}{n}. \tag{3.33}$$

For small and moderate blocklengths, an important performance metric of a DM is its rate loss

$$R_{\mathrm{loss}} = \mathrm{H}(P_{\hat{A}}) - R_{\mathrm{dm}}. \tag{3.34}$$

Constant Composition Distribution Matching

Constant composition distribution matching (CCDM) was introduced in [66] and implements a DM interface based on types, i.e., all output sequences a^n have the same number of occurrences $n_{a_i}(a^n), i = 1, \dots, |\mathcal{A}|$. We refer to a type configuration for a given output

[3] The function f_{dm} is injective.

alphabet $\mathcal{A}$ and output length n via the type vector

$$\boldsymbol{t}_{\mathcal{A}}^{n} = (n_{a_1}(a^n), n_{a_2}(a^n), \dots, n_{a_{|\mathcal{A}|}}(a^n)). \tag{3.35}$$

The set of all sequences of type $\boldsymbol{t}_{\mathcal{A}}^{n}$ is

$$\mathcal{T}^{\boldsymbol{t}_{\mathcal{A}}^{n}} = \{a^n \in \mathcal{A}^n \,|n_{a_i}(a^n) = [\boldsymbol{t}_{\mathcal{A}}^{n}]_i\ , \quad i = 1, \dots, |\mathcal{A}|\} \tag{3.36}$$

and its cardinality can be calculated by the multinomial

$$\left|\mathcal{T}^{\boldsymbol{t}_{\mathcal{A}}^{n}}\right| = \frac{n!}{\prod_{i=1}^{|\mathcal{A}|}[\boldsymbol{t}_{\mathcal{A}}^{n}]_i!}. \tag{3.37}$$

The maximum number k_{dm} of bits that can be encoded with the codebook $\mathcal{C}_{\mathrm{ccdm}}^{\boldsymbol{t}_{\mathcal{A}}^{n}} \subseteq \mathcal{T}^{\boldsymbol{t}_{\mathcal{A}}^{n}}$ is

$$k_{\mathrm{dm}} = \left\lfloor \log_2\left(\left|\mathcal{T}^{\boldsymbol{t}_{\mathcal{A}}^{n}}\right|\right)\right\rfloor. \tag{3.38}$$

CCDM realizes an n-type distribution on the output sequence a^n such that (3.32) becomes

$$P_{\hat{A}}(a_i) = \frac{n_{a_i}(a^n)}{n}. \tag{3.39}$$

For a given distribution P_A, its n-type approximation $P_{\hat{A}}$ (with the Kullback-Leibler divergence as underlying similarity metric) can be calculated with the approach shown in [67].

A CCDM can be implemented by arithmetic coding, where the arithmetic decoder serves as a DM encoding device and the arithmetic encoder implements the DM decoding. For binary output alphabets, an efficient approach is given in [68]. Further, the CCDM rate loss (3.34) vanishes for long output blocklengths [69], i.e., we have

$$\frac{k_{\mathrm{dm}}}{n} \xrightarrow{n\to\infty} \mathrm{H}(\hat{A}) \quad \text{or, equivalently, } R_{\mathrm{loss}} \xrightarrow{n\to\infty} 0. \tag{3.40}$$

Shell Mapping for Distribution Matching

Shell mapping for distribution matching (SMDM) is based on the SM algorithm [57] that was used in the V.34 modem standard [70]. The SM algorithm allows an efficient indexing of the lowest weight sequences for a given alphabet $\mathcal{A}$ and output length n. Traditionally, the indexing builds on a divide and conquer approach. Recently, approaches based on enumerative coding [71, 72] have emerged [73].

SMDM depends on the weight function $W^n : \mathcal{A}^n \to \mathbb{N}_0$ which assigns a weight to each sequence $a^n \in \mathcal{A}^n$, where we commonly have $W^n(a^n) = \sum_{i=1}^{n} W(a_i)$ with a slight abuse of notation. For instance, $W^n(a^n)$ may represent the "power cost" for the transmission of a^n. SM orders the sequences $a^n \in \mathcal{A}^n$ according to the sequence weights $W^n(a^n)$. As permutations of a^n have the same weight, they are assigned the same cost. SM creates one of these ordered lists, e.g., by lexicographical ordering. Overall, the SMDM codebook

$\mathcal{C}_{\text{smdm}}^{W^n}$ is the solution to the problem

$$\min_{\substack{\mathcal{C}_{\text{smdm}}^{W^n}\subseteq\mathcal{A}^n\\ \left|\mathcal{C}_{\text{smdm}}^{W^n}\right|=2^{k_{\text{dm}}}}} \sum_{a^n\in\mathcal{C}_{\text{smdm}}^{W^n}} W^n(a^n) = \min_{\substack{\mathcal{C}_{\text{smdm}}^{W^n}\subseteq\mathcal{A}^n\\ \left|\mathcal{C}_{\text{smdm}}^{W^n}\right|=2^{k_{\text{dm}}}}} \sum_{a^n\in\mathcal{C}_{\text{smdm}}^{W^n}} \sum_{i=1}^{n} W(a_i). \tag{3.41}$$

It was shown in [74] that SM implements the optimum block-to-block distribution matcher for the divergence metric $\mathrm{D}(P_{\tilde{A}^n} \parallel P_A^n)$, where $P_{\tilde{A}^n}$ is the realized output distribution on the sequences in $\mathcal{C}_{\text{smdm}}$ and P_A^n is the desired distribution. However, because of its implementation complexity, only moderate output lengths are practically feasible. For a software implementation, big integer libraries (e.g., GNU Multiple Precision Arithmetic Library, GMP[4] or libNTL[5]) are required to address input sequences with $k_{\text{dm}} > 64$ bits.

For our particular problem of power efficient signaling, we typically choose $W(a) = a^2$. In this case, the SMDM codebook contains sequences of minimum energy. The empirical symbol output distribution $P_{\hat{A}}$ is calculated as in [75].

3.4.3. Optimal Code Rate and Constellation Size

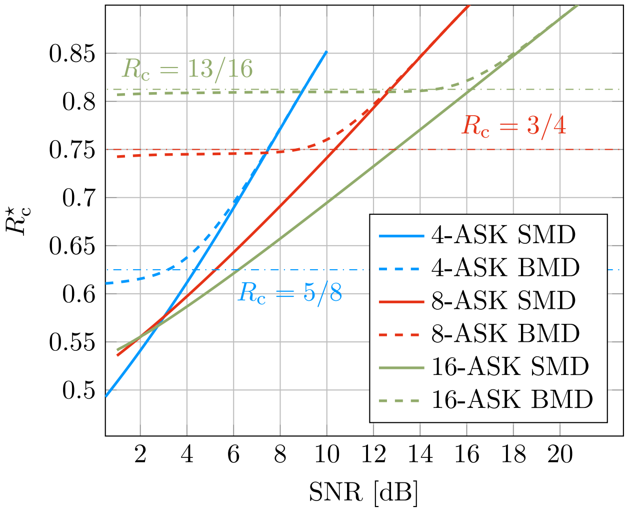


Figure 3.9.: Optimal R_c for 4-ASK, 8-ASK and 16-ASK and BMD and SMD.

As seen from (3.29), a desired transmission rate can be obtained by PAS with different code rates R_c by adjusting the DM rate accordingly. This is in contrast to uniform signaling, where $R_{\text{tx}} = R_c \cdot m$ such that for a given constellation cardinality of 2^m points the transmission rate is determined only by the code rate.

[4] `https://gmplib.org/`

[5] `https://www.shoup.net/ntl/`

This motivates to revisit the question for the optimal FEC code rate with PAS. In Sec. 3.3.1, the condition for error-free decoding is derived. Asymptotically, the optimal FEC code rate $R_\mathrm{c}^\star$ is therefore given by

$$R_\mathrm{c}^\star = 1 - \frac{1}{m}\,\mathrm{U}(q) \tag{3.42}$$

when the decoder uses the decoding metric q. For SMD, we have

$$R_\mathrm{c}^\star = 1 - \frac{1}{m}\,\mathrm{U}(q_\mathrm{SMD}) = 1 - \frac{1}{m}\,\mathrm{H}(X|Y) \tag{3.43}$$

and for BMD

$$R_\mathrm{c}^\star = 1 - \frac{1}{m}\,\mathrm{U}(q_\mathrm{BMD}) = 1 - \frac{1}{m}\sum_{k=1}^{m}\mathrm{H}(B_k|Y). \tag{3.44}$$

We show the respective optimal FEC rates for different SNRs in Fig. 3.9. The numerical computation uses an MB distribution. We see that the optimal FEC code rate $R_\mathrm{c}^\star$ for BMD is almost constant over a wide range for all three modulation formats. This suggests to operate PAS for BMD with the close-to-optimal code rates shown in Table 3.1. The pairing reflects code rates which are also commonly available for off-the-shelf codes in standards.

M-ASK	R_c
4	5/8
8	3/4
16	13/16

Table 3.1.: Close-to-optimal FEC code rates for PAS with BMD and a given M-ASK constellation.

In Fig. 3.10, we show the gap to the AWGN capacity for a given rate using the FEC rates of Table 3.1. We also add the SMD/BMD curves for 16-ASK from Fig. 3.5 for the optimal FEC code rate to have an insight for the incurred suboptimality. The observed gaps are smaller than 0.05 dB for the entire operating region. Interestingly, the loss due to a non-optimal FEC is even negligible for SMD (compare the solid gray and green curve).

3.4.4. Error Exponents for Probabilistic Amplitude Shaping

In [76], the error exponent for PAS was derived for a memoryless decoding metric. It is given as

$$E_\mathrm{PAS}(P_A, q) = \max_{0\le\rho\le 1}\quad \left(E_{0,\mathrm{PAS}}(\rho, P_A, q) - \rho m R_\mathrm{c}\right) \tag{3.45}$$

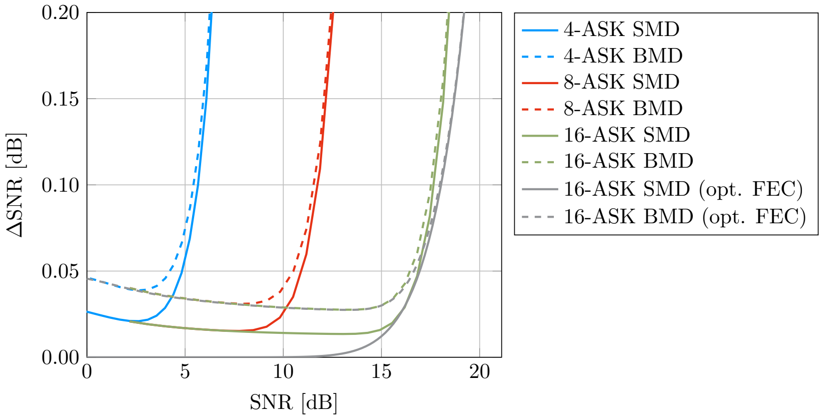


Figure 3.10.: Gap to AWGN capacity for PAS signaling with SMD and BMD using the close-to-optimal FEC code rates of Table 3.1.

where

$$E_{0,\mathrm{PAS}}(\rho, P_A, q) = \\ -\sum_{a\in\mathcal{A}} P_A(a) \log_2 \left(\int_{y\in\mathbb{R}} \sum_{s\in\{\pm1\}} p_{Y|X}(y|sa) P_S(s) \left(\frac{\sum_{x\in\mathcal{X}} q(x,y)\,|\mathcal{X}|}{q(sa,y)} \right)^{\rho} \mathrm{d}y \right) \quad (3.46)$$

Instantiating (3.46) with the metric (3.7) for SMD and (3.8) for BMD, we obtain

$$E_{0,\mathrm{PAS}}^{\mathrm{SMD}}(\rho, P_A) = \\ -\sum_{a\in\mathcal{A}} P_A(a) \log_2 \left(\int_{\mathbb{R}} \sum_{s\in\{\pm1\}} p_{Y|X}(y|sa) P_S(s) \left(\frac{p_Y(y)\,|\mathcal{X}|}{p_{Y|X}(y|sa) P_X(sa)} \right)^{\rho} \mathrm{d}y \right) \quad (3.47)$$

$$E_{0,\mathrm{PAS}}^{\mathrm{BMD}}(\rho, P_A) = \\ -\sum_{a\in\mathcal{A}} P_A(a) \log_2 \left(\int_{\mathbb{R}} \sum_{s\in\{\pm1\}} p_{Y|X}(y|sa) P_S(s) \left(\frac{p_Y(y)^m\,|\mathcal{X}|}{\prod_{k=1}^{m} p_{Y|B_k}(y|b_k) P_{B_k}(b_k)} \right)^{\rho} \mathrm{d}y \right) \quad (3.48)$$

where $b_1 = \chi_{\mathrm{S}}(s)$ and $b_k = [\chi_{\mathrm{A}}(a)]_{k-1}$ for $k = 2, \dots, m$ in (3.48). We refer to the corresponding error exponents of (3.47) and (3.48) as $E_{\mathrm{PAS}}^{\mathrm{SMD}}(P_A)$ and $E_{\mathrm{PAS}}^{\mathrm{BMD}}(P_A)$. In [77], the author derived PAS error exponents using the framework of joint source and channel coding.

In Fig. 3.11, we compare the RCBs based on the PAS error exponents for SMD and BMD decoding for 8-ASK and an SE of 1.5 bpcu. We observe that BMD entails almost no loss, while uniform signaling with BMD had a loss of 0.44 dB in Fig. 2.6. For the computation of the PAS RCB, we take the CCDM constraint into account, i.e., we use a

21 600-type distribution and use the same distribution for each evaluated SNR. Instead, the distribution for Gallager's RCB is optimized for each SNR.

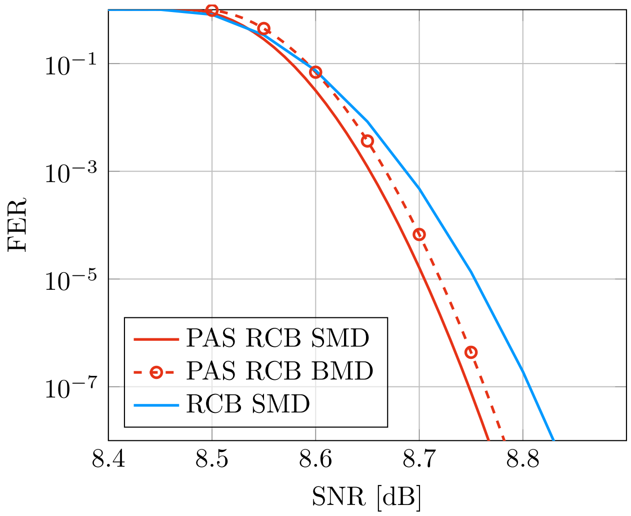


Figure 3.11.: Comparison of the PAS RCB for SMD and BMD for 8-ASK with $R_{\text{tx}} = 1.5$ bpcu.

3.5. Geometric Constellation Shaping

As an alternative to PS, geometric shaping (GS) can be employed to mimic a "Gaussian-like" shape of the input distribution. While PS imposes a non-uniform distribution on a set of equidistant constellation points, GS employs a uniform distribution on non-equidistant constellation points. The authors of [78] show that this approach achieves the capacity of the AWGN channel for SMD if the number of constellation points goes to infinity. In [79], the achievable rate of GS is investigated when both SMD and BMD are employed on one-dimensional constellations. Numerical results indicate that both optimization criteria lead to different constellations. Recently, GS constellations were included in the DVB-NGH [80, 81] and ATSC 3.0 standards [82, 83], where they are referred to as non-uniform constellations (NUCs). Besides, GS constellations are also considered for optical communications [84, 85, 86]. Most of these works use tailored optimization procedures that take potential non-linearities of the optical channel into account.

In the following, we compare PS and GS in terms of their information theoretic achievable rates for SMD and BMD. To this end, we propose a differential evolution [87] based optimization approach to optimize GS constellations for the AWGN channel. The results show that GS has a gap to capacity of about 0.4 dB when BMD is used. In contrast, PAS with BMD virtually achieves capacity. Further, we compare a selection of ATSC 3.0 modcods to a PAS system operating with a single modcod. FEC simulations show that the information theoretic gains predict the coded performance improvements accurately.

3.5.1. Optimized Constellations

A GS optimized constellation depends on the SNR and the employed decoding metric. In the following, we investigate the design of GS optimized constellations for two kinds of signaling with inphase and quadrature components: First, we construct constellations that can be constructed as the Cartesian product of two one-dimensional GS optimized constellations. Obviously, this results in less degrees of freedom, but facilitates optimization. As an additional benefit, the inphase and quadrature components of the constellation can be demapped independently of each other. Second, we design two-dimensional GS constellations where the full design space is exploited. Usually, this results in inphase and quadrature components that are correlated such that a two dimensional demapping is necessary for optimal performance. In this case, we have the system model

$$Y = X + N, \qquad \mathrm{E}\left[|X|^2\right] \leq P \tag{3.49}$$

where the discrete input X comes from a complex-valued constellation $\mathcal{X}$ and N is a circularly symmetric Gaussian RV with PDF

$$p_N(n) = \frac{1}{\pi\sigma^2}\mathrm{e}^{-\frac{|n|^2}{\sigma^2}}. \tag{3.50}$$

The SNR is defined as $\mathrm{E}\left[|X|^2\right]/\sigma^2$. For simplicity, we consider $\mathrm{E}\left[|X|^2\right] = 1$. The GS optimization problem can be formulated as

$$\mathcal{X}^* = \operatorname*{argmax}_{\substack{\mathcal{X}:\mathrm{E}\left[|X|^2\right]\leq 1\\ |\mathcal{X}|=M}} R_{\{\mathrm{BMD/SMD}\}}. \tag{3.51}$$

For both metrics, the optimization problem (3.51) in $\mathcal{X}$ is non-convex. The works [79, 88] employed “constrained non-linear optimization algorithms” without providing details on the employed optimization procedure. In [89], simulated annealing is used to optimize APSK constellations. Initial investigations by using standard, black box interior point algorithms like Matlab’s `fmincon` showed that the optimization depends on the initialization, which suggests that only locally optimal solutions are found.

We propose an optimization based on differential evolution [87], which is a genetic algorithm that appears to find the global optimum, i.e., differential evolution recovered previously reported results from any valid starting point.

The differential evolution approach is summarized in Algorithm 1. It starts with an initial population $\{\tilde{\mathcal{X}}_p^{(0)}\}_{p=1}^{P}$ of candidate constellations (see line 1 of Algorithm 1). In each generation, a population member experiences a mutation. For this, differential evolution randomly selects three distinct population members and combines them as shown in line 6 (*mutation*). The result of this operation may violate the feasible set and the function `map`$(\cdot)$ implements a bounce back strategy. Eventually, the new candidate constellation is generated by replacing each component of $\tilde{\mathcal{X}}_p^{(g-1)}$ with probability p_c by the corresponding entry of $\boldsymbol{T}$ (*crossing*). If the metric for the new candidate $\boldsymbol{T}$ has improved we keep

Algorithm 1 Genetic algorithm to find the best GS constellation for a given SNR.

```
INPUT: SNR, constellation size $M$, candidate set size $P$, number of generations $G$, crossover probability $p_c$, amplification factor $F$.
 1: Choose feasible initial population set $\{\tilde{\mathcal{X}}_p^{(0)}\}_{p=1}^{P}$ at random.
 2: Evaluate $\mathsf{R}_{\{\mathrm{BMD},\mathrm{SMD}\}}$ for each population member.
 3: for $g = 1, \dots, G$ do
 4:     for $p = 1, \dots, P$ do
 5:         Choose $r_1 \neq r_2 \neq r_3$ randomly from $\{1, \dots, P\}$.
 6:         $\boldsymbol{T} = \mathtt{map}(\tilde{\mathcal{X}}_{r_1}^{(g-1)} + F \cdot (\tilde{\mathcal{X}}_{r_2}^{(g-1)} - \tilde{\mathcal{X}}_{r_3}^{(g-1)}))$
 7:         $\boldsymbol{T} = \mathtt{mutate}(\boldsymbol{T}, \tilde{\mathcal{X}}_p^{(g-1)}, p_\mathrm{c})$
 8:         Evaluate metric of new candidate $\boldsymbol{T}$.
 9:         Set $\tilde{\mathcal{X}}_p^{(g)} = \boldsymbol{T}$, if metric has improved.
10:     end for
11:     if all population members have the same metric then
12:         Stop.
13:     end if
14: end for
```

it, otherwise we set $\tilde{\mathcal{X}}_p^{(g)} = \tilde{\mathcal{X}}_p^{(g-1)}$ (*selection*). We stop after G generations or once all population members have the same objective function value.

As discussed before, we distinguish between one-dimensional GS (1D-GS) and two-dimensional GS (2D-GS) and exploit symmetry to decrease the number of optimization parameters.

For 1D-GS and an M-ary 1D constellation (1D-GS 1D-NUC), each of the $M/2$ components of $\tilde{\mathcal{X}}_p$ is constrained to the non-negative real axis and the augmented, final constellation $\mathcal{X}_p$ with the negative part must fulfill the power constraint. A two-dimensional 1D-GS M-ary constellation (1D-GS 2D-NUC) can be obtained by the Cartesian product of two copies of 1D-GS $\sqrt{M}$-ary 1D-NUCs.

For 2D-GS, the population members are restricted to the first quadrant of the complex plane and $(M/4) \cdot 2$ real variables must be optimized ($M/4$ for the real and $M/4$ for the imaginary parts). This introduces additional degrees of freedom and leads to larger achievable rates.

To remain in the feasible set, i.e., the real non-negative axis for 1D-GS and the first quadrant for 2D-GS, the `map` function (see line 6 of Algorithm 1) replaces any negative real or imaginary part by its absolute value and rescales it to meet the power constraint.

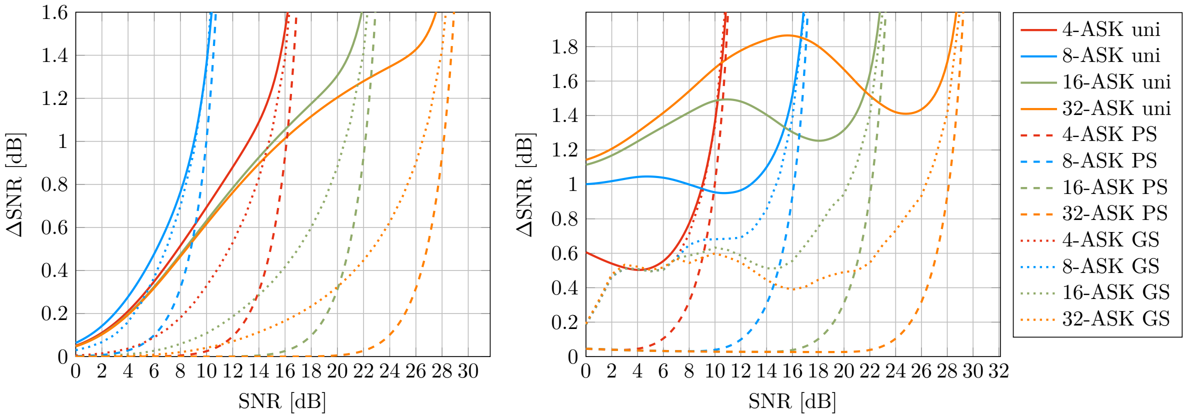
We used an amplification factor $F = 0.5$ and a crossover probability $p_\mathrm{c} = 0.88$. The number of generations is set to $G = 10\,000$ and the population size was chosen depending on the number of degree of freedoms (DOFs) as $P = 5 \cdot \mathrm{DOF}$. Choosing this parameter accurately turned out to be crucial in our experiments: Setting it too small, the optimum may not be found and setting it too large, the number of generations would not suffice. Usually 100 to 1000 generations are enough to observe convergence at low and medium SNR.

If the optimization metric targets BMD rates, the bit labeling must be taken into account as well. For 1D-GS, we randomly assign each component of $\tilde{\mathcal{X}}_p^{(0)}$ a $\log_2(M) - 1$ bit label.

The labels for the augmented constellation $\mathcal{X}_p$ are then obtained by first replicating and then prefixing each half with a zero and one, respectively. For 2D-GS, the same approach applies, however, each quadrant in the augmented constellation is prefixed by one of the four two-bit labels 00, 10, 11 and 01 in an ordered manner, which is consistent with ATSC 3.0.

3.5.2. Achievable Rate Comparison of PS and GS

In the following, we compare the BMD and SMD achievable rates for both GS and PS constellations. As a performance metric we employ the SNR gap to capacity of (3.25).



(a) SMD

(b) BMD

Figure 3.12.: Gap to capacity for 1D-GS $\{4, 8, 16, 32\}$-ASK constellations and PAS.

Fig. 3.12a shows the gap to capacity for optimized one-dimensional $\{4, 8, 16, 32\}$-ASK constellations with SMD. As a reference, we also plot the gaps for uniform, equidistant constellations. As derived in [78], the shaping gain of GS constellations increases with the constellation size.

Fig. 3.12b provides the same evaluation for BMD. Here, the gap to capacity does not exhibit a monotonic behavior and is larger in the low to medium SNR regime, as there is an additional BMD penalty. PS is better than GS over the whole range of constellations and SNR values. In particular, the gap to capacity remains almost constant at about 10^{-2} dB, which improves upon GS by more than 0.4 dB and vanishes for SMD.

In Fig. 3.13, we show the gap to capacity for 1D-GS $\{16, 64\}$-ary 2D-NUCs and 2D-GS $\{16, 32, 64\}$-ary constellations. The benefits of the additional degrees of freedom are clearly visible.

Summarizing, both approaches improve the SE of a communication system compared to uniform, equidistant signaling. PS is better than GS for the considered constellation sizes for both BMD and SMD. Hence, the statement of [79], claiming that "any gain in capacity which can be found via probabilistic shaping can also be achieved or exceeded solely

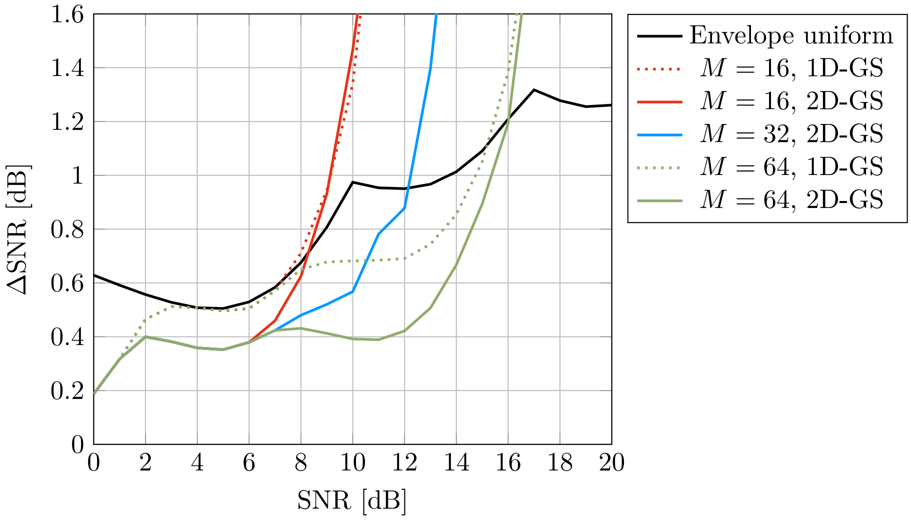


Figure 3.13.: BMD gap to capacity for 2D-GS constellations.

through geometric shaping" should be considered with caution, as it implicitly assumes a very large constellation size for GS. We illustrate this with two examples for SMD.

Example 5. In [78], the authors provide a signal set construction where the constellation points are chosen as the centroids of equiprobable quantiles of the Gaussian distribution and show that it is capacity-achieving for $M \to \infty$. An equidistant 8-ASK constellation with optimized distribution using the procedure of Sec. 3.4 for 10 dB yields $R_{\text{SMD}} = 1.726$ bpcu. To achieve the same rate, $M = 50$ constellation points must be used for this GS approach.

Example 6. In [90], the author describes a practical scheme to construct capacity-approaching APSK constellations with n rings having n constellation points each. We consider the same case as in Example 5. The rate gap $C_{\text{AWGN}} - \mathsf{R}_{\text{SMD}}$ for an equidistant 8-ASK constellation and optimized distribution at 10 dB is 0.0037 bits per real dimension. According to [90, Fig. 2b], this requires an APSK constellation with more than $35^2 = 1225$ points and two-dimensional demapping.

Similar observations can be found in [91], where the authors investigate the impact of constellation cardinality on the effect of approaching the AWGN channel capacity. They show that the convergence speed of methods like [78] is only $\mathcal{O}(1/M^2)$ (and thus require large constellation sizes), whereas using Gauss quadratures that involves both geometrical and probabilistic shaping approaches capacity exponentially fast in the constellation size.

3.5.3. Case Study: ATSC 3.0

We now compare the coded performance of PS and GS. The GS constellations of the recent ATSC 3.0 standard serve as reference designs.

For GS, the combination with FEC is straightforward and does not require any modifications. However, note that 2D-GS 2D-NUCs require two-dimensional demapping, so that the soft information calculation has increased complexity compared to QAM constellations where each component can be be demapped independently.

ATSC 3.0 defines 6 constellations (QPSK, {16, 64, 256, 1024, 4096}-NUCs), The smaller ones $(16, 64, 256)$ are 2D-GS 2D-NUCs, whereas the larger ones $(1024, 4096)$ are 1D-GS 2D-NUCs. The standard also defines LDPC codes with blocklengths 16 200 and 64 800 bits for code rates from 2/15 to 13/15 [92], giving rise to 46 modcods for the long blocklengths and 29 modcods for the short blocklengths [93].

For each modcod, the standard provides a constellation that has been designed to perform well with the associated code. In Fig. 3.14, we depict the operating points of all mandatory ATSC 3.0 modcods [82, Table 6.12] involving the $\{16, 64, 256\}$-ary 2D-GS 2D-NUCs by considering their gap to capacity. For each modcod, we calculate the required SNR to operate at an SE of $R_{\text{tx}} = \log_2(M) \cdot R_{\text{c}}$ bpcu, i.e., SNR $= R_{\text{BMD}}^{-1}(R_{\text{tx}})$. For PAS we consider a 256-QAM constellation that is constructed by the Cartesian product of two equidistant 16-ASK constellations and operated with a 5/6 rate code. We emphasize that only one modcod is necessary for PAS to operate within the targeted SE range of 1.0 bpcu to 5.33 bpcu within 0.06 dB.

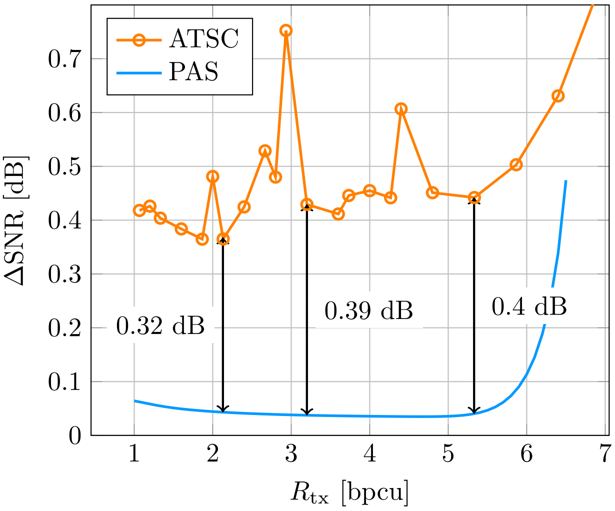


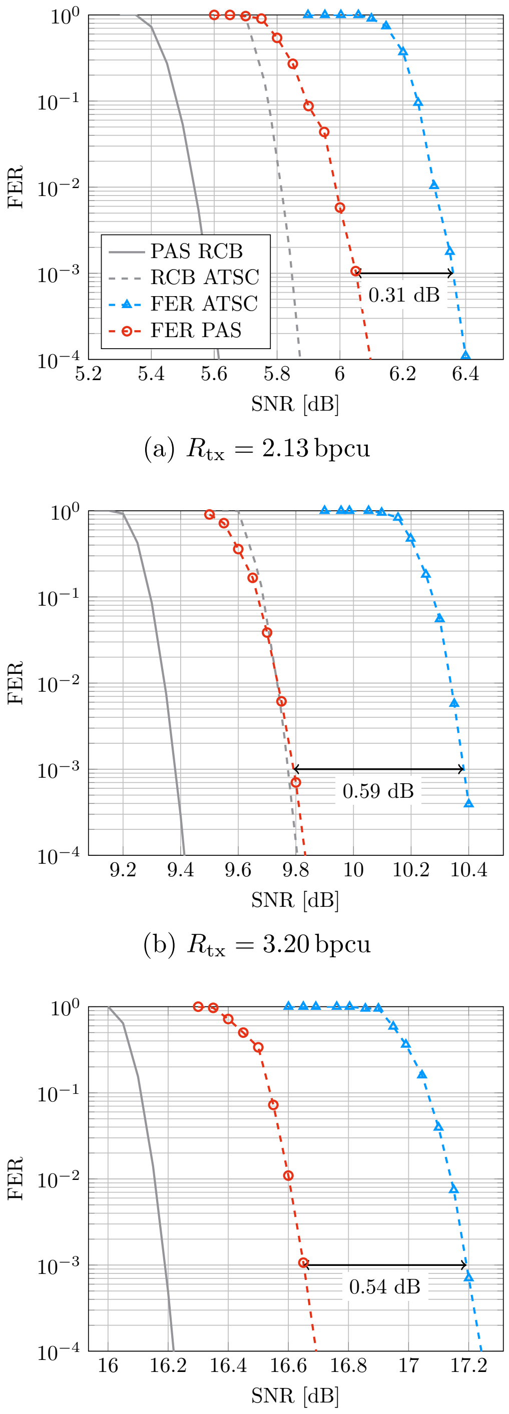
Figure 3.14.: SNR gap to capacity for ATSC 3.0 operating points comprising 2D-GS $\{16, 64, 256\}$ 2D-NUCs and allowed code rates compared to a single PAS modcod of 256-QAM and a 5/6 code.

In the following, we compare the coded performance of a selection of modcods which are summarized in Table 3.2. The comparison shows that the asymptotic gains of Fig. 3.14 translate into practice.

R_{tx} [bpcu]	Modcod	$R_{\text{BMD}}^{-1}(R_{\text{tx}})$ [dB]	ΔSNR [dB]
2.13	PAS 256-QAM, 5/6	5.34	0.043
	ATSC 16 2D-GS, 8/15	5.66	0.37
3.20	PAS 256-QAM, 5/6	9.17	0.038
	ATSC 64 2D-GS, 8/15	9.56	0.43
5.33	PAS 256-QAM, 5/6	15.99	0.040
	ATSC 256 2D-GS, 10/15	16.38	0.44

Table 3.2.: Considered modcods for SEs of 2.13, 3.2 and 5.33 bpcu.

The rate 8/15 and 10/15 LDPC codes for the ATSC 3.0 constellations are irregular repeat accumulate (IRA) LDPC codes [94] with blocklength 64 800. For each constellation, a different interleaving and bit-mapping is used according to the standard [82, Table 6.8]. PAS is operated with one single off-the-shelf 5/6 IRA LDPC code from the DVB-S2 standard of the same blocklength with an optimized bit-mapper of [9, Sec. VII-B]. In both cases, we used 50 belief propagation (BP) iterations with full sum-product update rule at the check nodes. Fig. 3.15 shows that the predicted asymptotic performance gains are reflected in the coded results. For SEs of 3.2 bpcu and 5.33 bpcu, the gains even exceed the predicted ones (0.59 dB vs. 0.39 dB and 0.54 dB vs. 0.4 dB).



(a) $R_{\text{tx}} = 2.13\,\text{bpcu}$

(b) $R_{\text{tx}} = 3.20\,\text{bpcu}$

(c) $R_{\text{tx}} = 5.33\,\text{bpcu}$

Figure 3.15.: Comparison of the coded performance of different ATSC 3.0 modcods and PAS using a single modcod.

3.6. Product Distribution Matching

In many practical settings, the data link is well modeled by a set of non-interacting parallel channels. Examples include multi-carrier transmission such as orthogonal frequency division multiplexing (OFDM), discrete multitone (DMT), and multi-antenna transceivers when the singular value decomposition (SVD) of the channel matrix is used to orthogonalize the system. Employing current DM algorithms in such scenarios is challenging, as techniques like bit-loading partition the transmitted sequence in several short segments, each with an individual constellation size and distribution, which potentially causes a significant rate loss.

For such applications, a tailored, hierarchical DM scheme is beneficial. Therefore, this section proposes product distribution matching (PDM), which internally uses a collection of parallel DMs with smaller output alphabets to synthesize the desired distribution as a product distribution. A preferable implementation uses binary output alphabets for the individual DMs. This approach both facilitates high-throughput applications by parallelization and reduces the rate loss for short output lengths, which makes the PDM particularly amenable for large constellations and high-throughput. A similar approach was developed independently in [95] and further investigated in [96].

3.6.1. Principles of Product Distribution Matching

The architecture for PDM is shown in Fig. 3.16. A number k_{dm} of binary data bits are demultiplexed into $m-1$ parallel blocks of lengths k_{dm_2} to k_{dm_m}. Note that we start counting by two as bit level one is the sign bit and has a uniform distribution. No binary matcher is required for this level. The $m-1$ parallel binary DMs output $m-1$ shaped binary sequences of length n. We introduce the labeling function $\chi_{\mathrm{A}}^{\mathrm{dm}} : \mathcal{A} \to \{0,1\}^{m-1}$. Its inverse recombines the $m-1$ DM output bits into an amplitude sequence of length n.

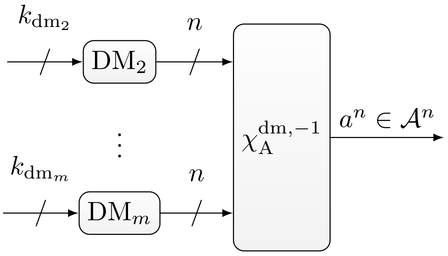


Figure 3.16.: The PDM architecture.

We further define the labeling function $\chi^{\mathrm{dm}}(x) = (\chi_{\mathrm{S}}(\mathrm{sign}(x)), \chi_{\mathrm{A}}^{\mathrm{dm}}(|x|)) = \boldsymbol{b}^{\mathrm{dm}}$. For PDM, we require that the distribution P_X factors for each bit level, i.e.,

$$P_X(x) = \prod_{k=1}^{m} P_{B_k^{\mathrm{dm}}}([\chi^{\mathrm{dm}}(x)]_k). \tag{3.52}$$

At this point, it is not clear how χ^{dm} should be chosen. However, from the shaping layer considerations in Sec. 3.3.2, we have

$$\mathrm{D}(P_X \parallel P_U) = \left[\sum_{k=1}^{m} \mathrm{H}(B_k^{\text{dm}})\right] - \log_2(|\mathcal{X}|) \tag{3.53}$$

when P_X has the form (3.52). Analog to (3.15), the first term in (3.53) refers to the asymptotic shaping set size. Therefore, an achievable rate is

$$R_{\text{BMD}}^{\Pi} = \left[\sum_{k=1}^{m} \mathrm{H}(B_k^{\text{dm}}) - \mathrm{U}(q_{\text{BMD}})\right]^{+} = \left[\sum_{k=1}^{m} \mathrm{H}(B_k^{\text{dm}}) - \sum_{k=1}^{m} \mathrm{H}(B_k^{\text{fec}}|Y)\right]^{+}. \tag{3.54}$$

Note that (3.54) already takes into account that the mapping χ^{dm} for constructing the amplitude sequence may be different from χ^{fec} used for modulation and demapping, i.e., the one that is used to calculate the bit metric. In general, the bit levels of $\boldsymbol{B}^{\text{fec}}$ are stochastically dependent, while those of $\boldsymbol{B}^{\text{dm}}$ are not.

3.6.2. Optimal Input PMF

To find the optimum input distribution for PDM, we solve the following optimization problem:

$$\max_{P_X,\Delta,\chi^{\text{dm}},\chi^{\text{fec}}} \quad R_{\text{BMD}}^{\Pi} \quad \text{subject to} \quad \mathrm{E}\left[(\Delta X)^2\right] \le P. \tag{3.55}$$

Numerical simulations for the average power constrained additive white Gaussian noise channel (AWGNC) suggest to use the natural based binary code (NBBC) for χ^{dm} (see Table 3.3) and the BRGC for χ^{fec} – for smaller constellation sizes, an exhaustive search over all labeling functions is possible.

Similar to (3.23), a good heuristic for the optimal P_X of (3.55) for the average power constrained AWGNC is the solution of the following optimization problem:

$$\begin{aligned} \min_{P_{B_2},\dots,P_{B_m}} \quad & \mathrm{E}\left[X^2\right] \\ \text{subject to} \quad & \sum_{j=2}^{m} \mathrm{H}(B_k^{\text{dm}}) = R_{\text{dm}} \\ & \boldsymbol{B} = \chi^{\text{dm}}(X). \end{aligned} \tag{3.56}$$

Remark 1. In [97], the authors considered the case when $\boldsymbol{B}^{\text{dm}} = \boldsymbol{B}^{\text{fec}}$, in which case (3.54) becomes

$$R_{\text{BICM}} = \sum_{k=1}^{m} \mathrm{I}(B_k^{\text{fec}}; Y) \tag{3.57}$$

	-7	-5	-3	-1	1	3	5	7
BRGC	000	001	011	010	**110**	**111**	**101**	**100**
NBBC	000	001	010	011	**111**	**110**	**101**	**100**

Table 3.3.: Two labels for 8-ASK. The amplitude label of NBBC is NBC and the amplitude label of BRGC is also BRGC.

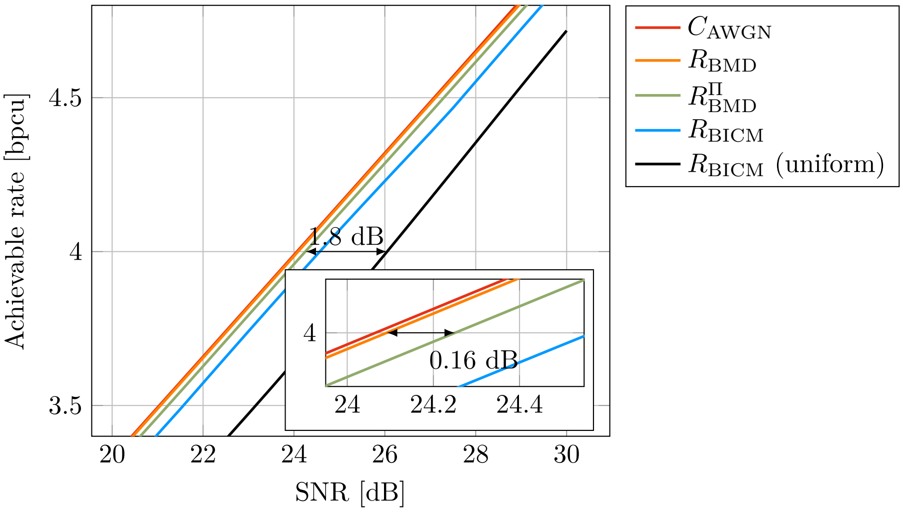


Figure 3.17.: Achievable rates for 64-ASK and different bit-metric decoding schemes.

which is the "BICM capacity" and derived earlier in (2.84). The rate expression of (3.19) is more general and allows to capture the effect of different labeling strategies at the transmitter and receiver ($\boldsymbol{B}^{\mathrm{dm}}$ vs. $\boldsymbol{B}^{\mathrm{fec}}$).

In Fig. 3.17, we display the achievable rates for 64-ASK and different shaping schemes. Note that the input distribution has been optimized for the shaped cases of R_{BMD} and R^{Π}_{BMD} for each SNR. We observe that the product constraint (3.52) in combination with the adjusted labeling at the transmitter ($\boldsymbol{B}^{\mathrm{dm}}$) and receiver ($\boldsymbol{B}^{\mathrm{fec}}$) leads to a performance loss of only 0.16 dB compared to R_{BMD} with a 32-ary DM at an SE of 4 bpcu. At the same time, the energy efficiency is improved by 1.8 dB over R_{BICM} with a uniform distribution.

3.6.3. Numerical Simulation Results

We numerically assess the different DM implementations by using 64-ASK, a DM amplitude distribution with $R_{\mathrm{dm}} = 4.1$ bits and $R_{\mathrm{c}} = 9/10$. This scenario therefore targets an SE of $R_{\mathrm{tx}} = 4.5$ bpcu with $\gamma = 0.4$. We employ a 32-ary CCDM as a reference. The performance of this system is compared to a PDM setup with 1 (B_2^{dm}), 2 ($B_2^{\mathrm{dm}}, B_3^{\mathrm{dm}}$), 3 ($B_2^{\mathrm{dm}}, B_3^{\mathrm{dm}}, B_4^{\mathrm{dm}}$), 4 ($B_2^{\mathrm{dm}}, B_3^{\mathrm{dm}}, B_4^{\mathrm{dm}}, B_5^{\mathrm{dm}}$) and 5 ($B_2^{\mathrm{dm}}, B_3^{\mathrm{dm}}, B_4^{\mathrm{dm}}, B_5^{\mathrm{dm}}, B_6^{\mathrm{dm}}$) individually shaped bit levels and corresponding binary CCDMs.

DM configuration	Required SNR [dB]	Gap to capacity [dB]
32-ary DM	27.13	0.05
PDM 1 Bit shaped	28.29	1.21
PDM 2 Bits shaped	27.48	0.40
PDM 3 Bits shaped	27.35	0.27
PDM 4 Bits shaped	27.32	0.24
PDM 5 Bits shaped	27.31	0.23

Table 3.4.: Required SNRs for different DM configurations and a target SE of 4.5 bpcu ($C_{\mathrm{AWGN}}^{-1}(4.5) = 27.08\,\mathrm{dB}$).

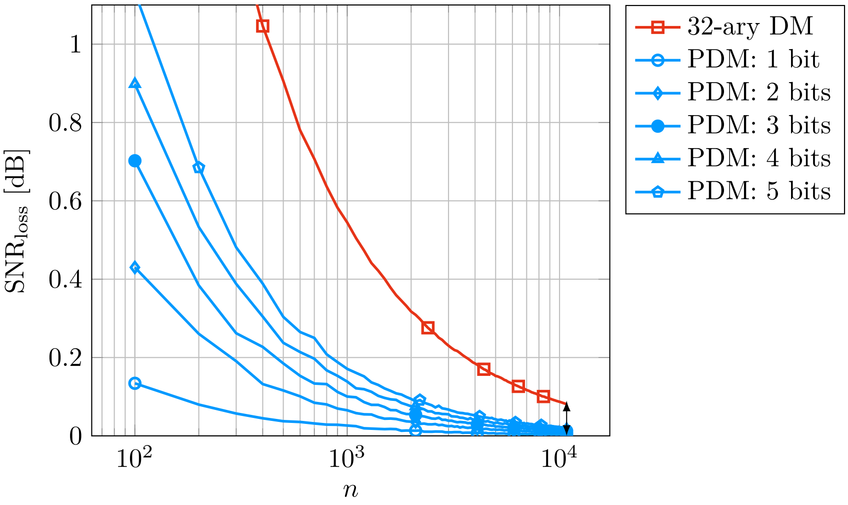


Figure 3.18.: SNR loss comparison for 64-ASK and $\mathrm{H}(A) = 4.1$ bits.

The capacity analysis of Table 3.4 provides insights into the asymptotic performance of the considered schemes: While the gap to capacity to achieve an SE of 4.5 bpcu is very similar for 3, 4 and 5 shaped bit levels, larger gaps can be observed when only 1 or 2 bit levels are shaped. In all cases, the channel input distributions were chosen according to (3.22) and (3.56) for $R_{\mathrm{dm}} = 4.1$ bits. For PDM, a uniform distribution is imposed on the unshaped bit levels.

However, the capacity analysis does not take the finite length implementation penalty of the DMs into account. To assess this influence, we consider the rate loss of the PDM scheme similar to (3.34). The rate and the output distribution of the k-th DM, $k = 2, \ldots, m$, is k_{dm_k}/n and $P_{B_k^{\mathrm{dm}}}$, respectively. If the PDM is realized by parallel, binary CCDMs then k_{dm_k} can be computed analogously to (3.38) via

$$k_{\mathrm{dm}_k} = \left\lfloor \log_2 \binom{n}{n \cdot P_{B_k^{\mathrm{dm}}}(0)} \right\rfloor . \tag{3.58}$$

The DM rate of PDM is

$$\frac{k_{\mathrm{dm}}}{n} = \frac{k_{\mathrm{dm}_2} + \cdots + k_{\mathrm{dm}_m}}{n} \tag{3.59}$$

and the total rate loss of PDM is the sum of the individual rate losses, i.e.,

$$R_{\mathrm{loss}} = \sum_{k=2}^{m} \left[\mathrm{H}(B_k^{\mathrm{dm}}) - \frac{k_{\mathrm{dm}_k}}{n} \right]. \tag{3.60}$$

We evaluate (3.60) for output blocklengths ranging from 100 to 10 800 symbols in Fig. 3.18. To allow an easier comparison, the rate loss is converted to an "SNR loss" via

$$\mathrm{SNR}_{\mathrm{loss}} = 10 \log_{10} \left(\frac{R_{\mathrm{BMD}}^{-1}(R_{\mathrm{tx}} + R_{\mathrm{loss}})}{R_{\mathrm{BMD}}^{-1}(R_{\mathrm{tx}})} \right). \tag{3.61}$$

As a rule of thumb, the following expression is useful as a rough estimate:

$$\begin{aligned} \mathrm{SNR}_{\mathrm{loss,awgn}} &= 10 \log_{10} \left(\frac{2^{2(R_{\mathrm{tx}} + R_{\mathrm{loss}})} - 1}{2^{2R_{\mathrm{tx}}} - 1} \right) \\ &\approx R_{\mathrm{loss}} \cdot 20 \log_{10} 2 \approx R_{\mathrm{loss}} \cdot 6\,\mathrm{dB}. \end{aligned} \tag{3.62}$$

We observe that the PDMs have an aggregated rate loss that is significantly lower than the rate loss of the 32-ary DM.

For the case of factorizable amplitude distributions, a simple combinatoric argument shows that symbol-wise DMs have a larger rate loss than PDMs.

Example 7. Consider the output distribution $[\frac{4}{9}, \frac{2}{9}, \frac{2}{9}, \frac{1}{9}]$, which is a product distribution of $[\frac{2}{3}, \frac{1}{3}]$ and $[\frac{2}{3}, \frac{1}{3}]$. The CCDM creates a codebook with 3780 entries, consisting of all permutations of $[1, 1, 1, 1, 2, 2, 3, 3, 4]$. The PDM with binary CCDMs creates a codebook with all permutations of $[1, 1, 1, 1, 1, 1, 4, 4, 4]$, $[1, 1, 1, 1, 1, 2, 3, 4, 4]$, $[1, 1, 1, 1, 2, 2, 3, 3, 4]$ and $[1, 1, 1, 2, 2, 2, 3, 3, 3]$ which corresponds to 84, 1512, 3780 and 1680 possible sequences. The number of potential sequences has increased nearly by a factor of 2. According to (3.33) and (3.59), this leads to matcher rates of $\frac{11}{9}$ and $\frac{12}{9}$ for CCDM and PDM, respectively.

To investigate both finite and asymptotic effects, we consider a coded scenario with a rate 9/10 LDPC block code from the DVB-S2 standard [98] of block length 64 800 bits and a corresponding DM output length of 10 800 symbols. One hundred iterations are used for the BP decoding.

For an output length of 10 800 symbols (marked by an arrow in Fig. 3.18), the 32-ary DM has a SNR loss of 0.1 dB, whereas the loss for the PDM implementations is smaller than 0.02 dB. At the same time, the 32-ary DM gains asymptotically, so that both effects start to outweigh each other. This can be seen in particular for PDM with 4 and 5 shaped bit levels, which show similar performance as the 32-ary DM. The large gap in the Shannon

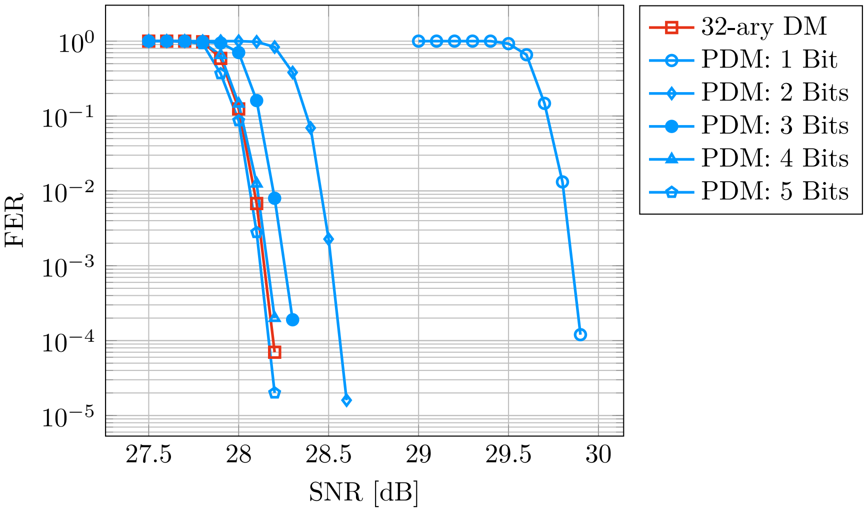


Figure 3.19.: Performance comparison of the proposed PDM for 64-ASK and a target SE of 4.5 bpcu and different numbers of shaped bits.

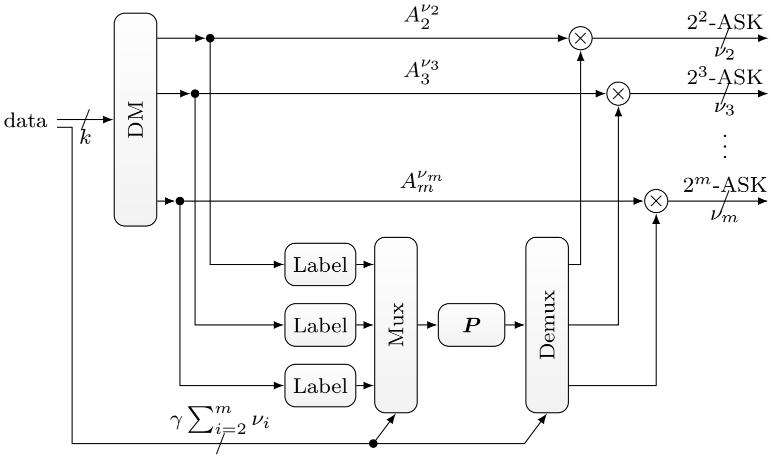


Figure 3.20.: Illustration of PAS for $L = \sum_{i=2}^{m} \nu_i$ parallel channels. Note that the power control can still be applied individually.

limit for only 1 or 2 shaped bits can also be observed in the plot. This notion allows a trade-off in practice: If a certain performance degradation is tolerable, a certain number of DMs can be saved. For instance, if only 2 bit levels are shaped, then the loss in energy efficiency is 0.4 dB at a target FER of 10^{-3}.

3.7. Probabilistic Amplitude Shaping for Parallel Channels

In this section, we build upon the previous PDM scheme and apply it to a set of parallel channels, which shares the component DMs for lower bit levels among different sub-carriers.

We consider L parallel AWGN channels of the form

$$Y_\ell = h_\ell X_\ell + N_\ell, \quad \ell = 1, 2, \dots, L. \tag{3.63}$$

The noise terms N_ℓ are zero mean Gaussian with unit variance. The coefficients h_ℓ model the channel gains and we assume that both the receiver and transmitter have full channel state information, i.e., they both know the channel gains h_ℓ and the noise variance.

3.7.1. Waterfilling Benchmark

The transmitter has an average power budget P, i.e., the inputs are subject to the sum power constraint

$$\frac{1}{L}\sum_{\ell=1}^{L} \mathrm{E}\left[X_\ell^2\right] \leq P. \tag{3.64}$$

The average SE

$$\frac{1}{L}\sum_{\ell=1}^{L} \frac{1}{2}\log_2(1 + h_\ell^2 P_\ell) \tag{3.65}$$

is achievable with the channel inputs X_ℓ being independent zero mean Gaussian with variance P_ℓ. The average SE is maximized by waterfilling, i.e.,

$$P_\ell^* = \left[\frac{1}{\lambda} - \frac{1}{h_\ell^2}\right]^+, \quad \lambda\colon \frac{1}{L}\sum_{\ell=1}^{L} P_\ell^* = P. \tag{3.66}$$

Suppose that P_ℓ^* is positive. The SE allocated to channel ℓ is then $C_\ell = \frac{1}{2}\log_2(h_\ell^2/\lambda)$ and we have

$$\mathsf{C}_{\mathrm{WF}}(P) = \frac{1}{L}\sum_{\ell=1}^{L} C_\ell = \frac{1}{L}\sum_{\ell=1}^{L} \frac{1}{2}\log_2 \frac{h_\ell^2}{\lambda}. \tag{3.67}$$

The function $\mathsf{C}_{\mathrm{WF}}(P)$ is the maximum achievable SE under the sum power constraint P and it serves in the following as our benchmark. For discrete inputs, the power allocation follows the mercury-waterfilling principle [99]. In the following, we develop a heuristic that uses PDM and operates closely to $\mathsf{C}_{\mathrm{WF}}(P)$.

3.7.2. Bit-Loading Strategy

Since the per-channel SEs can differ by several bits, we need to support several constellations in parallel. This is important for, e.g., DSL systems where some good channels may support up to 32 768-QAM [100], whereas the majority of channels needs to be operated with smaller modulation formats. Next, we have to decide which constellation size is used for which channel, an approach known as bit-loading. We employ the following heuristic: We calculate the waterfilling solution for the given channel coefficients and obtain the optimal rate assignment $C_\ell, \ell = 1, \dots, L$ from (3.67). Then, we use Ungerböck's rule-of-thumb [46] to choose a constellation size $M_\ell = 2^{m_\ell}$ for channel ℓ such that $m_\ell \approx C_\ell + 1$. This avoids a reduced SE because of too small constellation sizes. We assume the largest constellation size is 2^m, i.e., $m = \max_\ell m_\ell$. Further, the smallest constellation size is 2^2-ASK for the ease of exposure. An extension to channels using binary phase shift keying (BPSK) is straightforward.

3.7.3. Product Distribution Matching for Parallel Channels

PAS can be combined with parallel channels as illustrated in Fig. 3.7. A DM device transforms data bits into a sequence of amplitudes for each channel, which are then combined with sign bits originating from a common encoding device. In its simplest form, this DM device internally uses individual DMs, each with its output alphabet size matched to the corresponding constellation size.

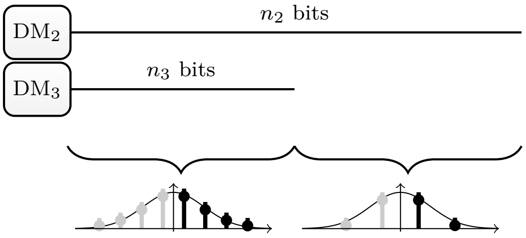


Figure 3.21.: Simultaneously generating two Gaussian-like amplitude distributions for 4-ASK and 8-ASK by reusing the DM of bit level 2.

PDM allows to jointly generate a length L amplitude sequence with different constellation sizes. For example, suppose we have $L = \nu_2 + \nu_3$ possibly different channels where ν_2 channels use 4-ASK and ν_3 channels use 8-ASK. The PDM needs one binary DM for 4-ASK and two binary DMs for 8-ASK. As illustrated in the top part of Fig. 3.21, the idea is now to use for the first amplitude bit level B_2 of 4-ASK and 8-ASK a single binary DM with output length $n_2 = \nu_2 + \nu_3$, and to generate the second amplitude bit level B_3 for 8-ASK by a second binary DM with output length $n_3 = \nu_3$. This approach allows the DMs to operate over a longer blocklength, causing the DM rate to reach its asymptotic limit faster. The illustration in the bottom part of Fig. 3.21 shows this scheme.

We state how the system can be parameterized to operate at a given SE. For the considered case we assume ν_i channel uses of a 2^i-ASK constellation for $i = 2, \dots, m$ within one FEC frame. The blocklength of the FEC code is

$$n_\mathrm{c} = L + \sum_{i=2}^{m} n_i \tag{3.68}$$

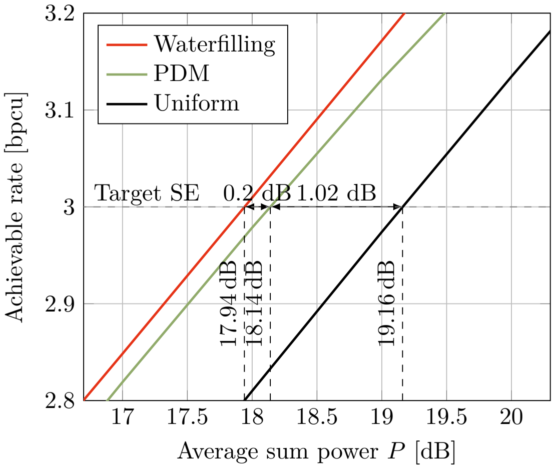


Figure 3.22.: Achievable rates of the considered example.

where we have $L = \sum_{i=2}^{m} \nu_i$ and the parameters n_i denote the DM output lengths

$$n_i = \sum_{\ell=1}^{L} \mathbb{1}(m_\ell \geq i) \cdot \nu_i, \quad i = 2, 3, \ldots, m. \tag{3.69}$$

The corresponding DM input lengths are $k_2, k_3, \ldots, k_m$. The average SE of the overall system is now

$$R_{\text{tx}} = \frac{\sum_{i=2}^{m} k_i}{\sum_{i=2}^{m} \nu_i} + \gamma. \tag{3.70}$$

and converges to $(\sum_{i=2}^{m} \mathrm{H}(B_i^{\text{dm}}) n_i)/(\sum_{i=2}^{m} \nu_i) + \gamma$ for large L. The formula (3.30) generalizes to

$$\gamma = 1 - (1 - R_{\text{c}}) \frac{\sum_{i=2}^{m} \nu_i \cdot i}{\sum_{i=2}^{m} \nu_i}. \tag{3.71}$$

3.7.4. Simulation Results

To evaluate the performance of parallel PAS with PDM, we employ the following example of three different constellation sizes which are used equally often. The coefficients are chosen such that the channel quality varies significantly (over a range of 12 dB) and requires three different modulation formats. The waterfilling solution (3.67) for a target SE of 3.0 bpcu yields the following rate allocation

$$\begin{aligned} Y_1 &= 2.0 \cdot X_1 + Z_1, \quad & C_1 &= 4.0 \\ Y_2 &= 1.0 \cdot X_2 + Z_2, \quad & C_2 &= 3.0 \\ Y_3 &= 0.5 \cdot X_3 + Z_3, \quad & C_3 &= 2.0 \end{aligned}$$

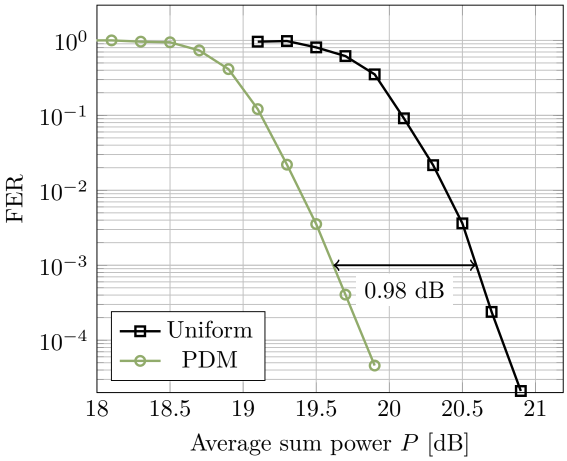


Figure 3.23.: Coded performance comparison of PDM and uniform scheme for parallel channels (LDPC code with block length 3600 bits).

which is achieved for an average sum-power of 17.94 dB. We select constellation sizes of

DM_i	ν_i	n_i	$P_{B_i^{\mathrm{dm}}}(0)$	$\mathrm{H}(B_i^{\mathrm{dm}})$
2	300	900	0.1995	0.7208
3	300	900	0.3736	0.9534
4	300	600	0.4408	0.9898
5	300	300	0.4709	0.9976

Table 3.5.: PDM properties for the considered example.

$2^{m_1} = 32$, $2^{m_2} = 16$ and $2^{m_3} = 8$ points according to our bit-loading strategy of Sec. 3.7.2. The achievable rates are plotted against the average sum power in Fig. 3.22. Our proposed heuristic scheme exhibits a gap of 0.2 dB to the waterfilling benchmark of (3.67) for the target SE of 3.0 bpcu. The uniform reference curve is shown in black and has a gap of 1.22 dB to the waterfilling solution. The employed bit distributions are summarized in Table 3.5 and have been chosen as the solution of the following heuristic optimization problem (see also (3.56)) for $R_{\mathrm{dm}} = 3.0 - \gamma$ with $\gamma = 1/3$:

$$\min_{P_{B_2^{\mathrm{dm}}},\dots,P_{B_m^{\mathrm{dm}}}} \sum_{\ell=1}^{3} \frac{1}{h_\ell^2} \mathrm{E}\left[X_\ell^2\right] \quad \text{subject to} \quad \frac{1}{L}\sum_{i=2}^{5} \mathrm{H}(B_i^{\mathrm{dm}}) n_i = R_{\mathrm{dm}}. \tag{3.72}$$

In Fig. 3.23, we consider the same scenario with finite length LDPC codes from the 5G standard [101] (basegraph BG1). The uniform reference uses a $R_{\mathrm{c}} = 3/4$, while the shaped case has a $R_{\mathrm{c}} = 5/6$ code ($\gamma = 1/3$). In both cases the number of transmitted bits is 3600. The asymptotic gains are reflected in the coded results. We perform 100 BP iterations.

3.8. Probabilistic Amplitude Shaping for Hard-Decision Decoding

This section investigates achievable rates for PAS with HD decoding metrics. While HD decoding metrics are generally outperformed by their SD counterparts, they play an important role for high throughput and low complexity optical receivers. There is no common definition of HD metrics in the literature. In the following, we refer to a HD decoding metric as one that obeys the following two principles:

1. The decoding metric does not exploit reliability information.
2. The decoding metric is based on the Hamming distance between the binary representation of the received value and the transmitted constellation symbol.

We note that this definition is different from, e.g., [102, Sec. III-C] where the authors derive achievable rates for “HD coded modulation decoders”, but allow to exploit soft-information associated with the probability of the individual channel transition probabilities.

3.8.1. Demapping Strategies

We consider a coherent receiver architecture. As a result, we have to determine the hard estimate for a given receive symbol $y \in \mathbb{R}$ first. We distinguish between symbol-metric and bit-metric based approaches. A symbol-wise HD demapper obtains the estimate

$$\hat{x} = Q_{\text{HD}}^{\text{SW}}(y) = \operatorname*{argmax}_{x \in \mathcal{X}} P_{X|Y}(x|y) = \operatorname*{argmax}_{x \in \mathcal{X}} p_{Y|X}(y|x) P_X(x) \tag{3.73}$$

whereas a bit-wise HD demapper obtains the estimate for the k-th bit as

$$\hat{b}_k = Q_{\text{HD}}^{\text{BW},k}(y) = \operatorname*{argmax}_{b \in \{0,1\}} P_{B_k|Y}(b|y) = \operatorname*{argmax}_{b \in \{0,1\}} p_{Y|B_k}(y|b_k) P_{B_k}(b). \tag{3.74}$$

Both expressions represent the maximum a posteriori (MAP) estimate of the respective symbol. Similarly, an ML estimate is possible when the respective prior (P_X or P_{B_k}) is neglected. With a slight abuse of notation, the Voronoi regions are given as

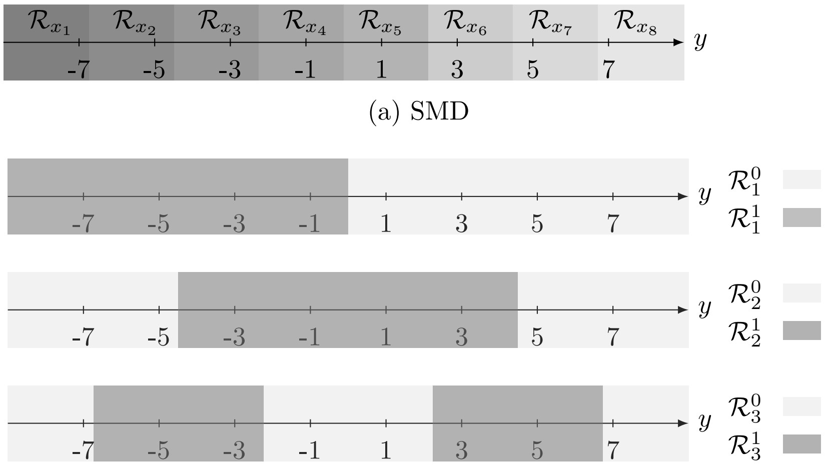
$$\mathcal{R}_x = \left\{ y \in \mathbb{R} : Q_{\text{HD}}^{\text{SW}}(y) = x \right\} \tag{3.75}$$

and

$$\mathcal{R}_k^b = \left\{ y \in \mathbb{R} : Q_{\text{HD}}^{\text{BW},k}(y) = b \right\}. \tag{3.76}$$

We exemplarily depict the respective Voronoi regions in Fig. 3.24 for 8-ASK with PS and an MB distribution with $\mathrm{H}(X) = 2.5$ bits. The SNR is 9 dB.

Further, we show the influence of the respective demapping strategy on the uncoded BER for 8-ASK with uniform and shaped signaling and a BRGC label in Fig. 3.25. The



(a) SMD

(b) BMD

Figure 3.24.: Voronoi regions for HD demapping.

PS case uses the same MB distribution as before. As seen from the plot, the chosen demapping strategy has almost no influence numerically. Similar conclusions were drawn in [103].

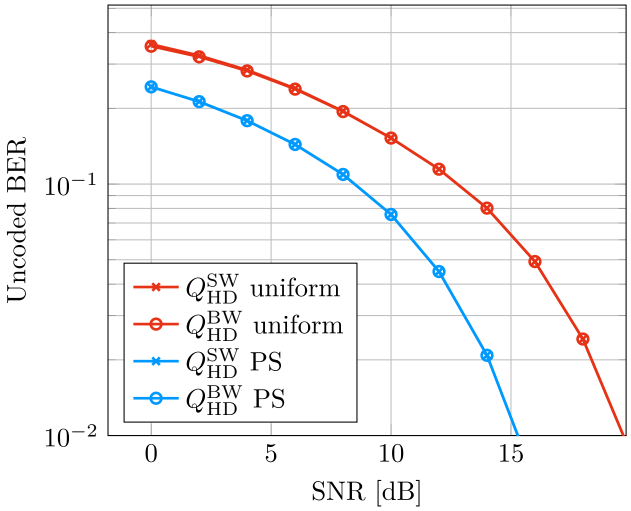


Figure 3.25.: Uncoded BER of uniform and shaped 8-ASK with different demapping metrics for HD decoding.

3.8.2. Achievable Rate Derivation

To derive achievable rates for HD schemes, we first state the decoding metrics and then use these to calculate the respective mismatched uncertainty expressions according to (3.6).

Following our definition for HD based metrics, we choose

$$q_{\text{SMD}}^{\text{HD}}(x,y) = \varepsilon^{\mathbb{1}\left(x \neq Q_{\text{HD}}^{\text{SW}}(y)\right)} \tag{3.77}$$

$$q_{\text{BMD}}^{\text{HD}}(x,y) = \prod_{k=1}^{m} \varepsilon^{\mathbb{1}\left([\chi(x)]_k \neq Q_{\text{HD}}^{\text{BW},k}(y)\right)} = \varepsilon^{\sum_{k=1}^{m} \mathbb{1}\left([\chi(x)]_k \neq Q_{\text{HD}}^{\text{BW},k}(y)\right)} \tag{3.78}$$

where $\varepsilon \in [0,1)$ is a constant. The mismatched uncertainty expressions are given by

$$\text{U}\left(q_{\text{SMD}}^{\text{HD}}\right) = \min_{s \geq 0} \text{E}\left[-\log_2\left(\frac{\varepsilon^{s \cdot \mathbb{1}\left(X \neq Q_{\text{HD}}^{\text{SW}}(Y)\right)}}{\sum_{a \in \mathcal{X}} \varepsilon^{s \cdot \mathbb{1}\left(a \neq Q_{\text{HD}}^{\text{SW}}(Y)\right)}}\right)\right], \tag{3.79}$$

$$\text{U}\left(q_{\text{BMD}}^{\text{HD}}\right) = \min_{s \geq 0} \text{E}\left[-\log_2\left(\frac{\varepsilon^{s \cdot \sum_{k=1}^{m} \mathbb{1}\left([\chi(X)]_k \neq Q_{\text{HD}}^{\text{BW},k}(Y)\right)}}{\sum_{a \in \mathcal{X}} \varepsilon^{s \cdot \sum_{k=1}^{m} \mathbb{1}\left([\chi(a)]_k \neq Q_{\text{HD}}^{\text{BW},k}(Y)\right)}}\right)\right] \tag{3.80}$$

such that we obtain the achievable rates according to (3.14) as

$$R_{\text{SMD}}^{\text{HD}} = \left[\text{H}(X) - \text{U}\left(q_{\text{SMD}}^{\text{HD}}\right)\right]^{+} \tag{3.81}$$

$$R_{\text{BMD}}^{\text{HD}} = \left[\text{H}(X) - \text{U}\left(q_{\text{BMD}}^{\text{HD}}\right)\right]^{+}. \tag{3.82}$$

The previous expressions can be simplified further and we can give an interpretation of the optimization of (3.79) and (3.80). As shown in Appendix A.2, we have

$$R_{\text{SMD}}^{\text{HD}} = [\text{H}(X) - \text{H}_2(\delta_{\text{SMD}}) - \delta_{\text{SMD}} \log_2(M-1)]^{+} \tag{3.83}$$

$$R_{\text{BMD}}^{\text{HD}} = [\text{H}(X) - m\,\text{H}_2(\delta_{\text{BMD}})]^{+} \tag{3.84}$$

where

$$\delta_{\text{SMD}} = \Pr(\hat{X} \neq X) \tag{3.85}$$

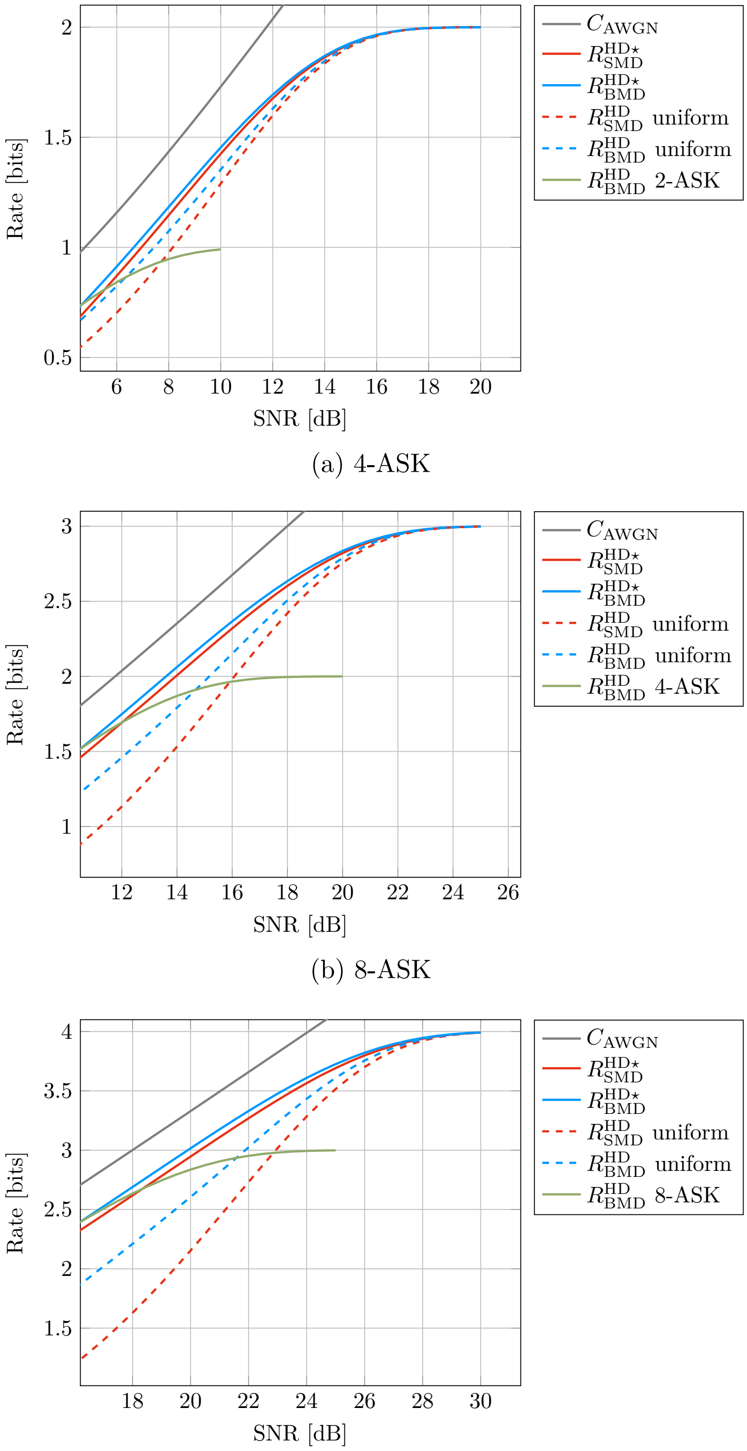
$$\delta_{\text{BMD}} = \frac{1}{m} \sum_{k=1}^{m} \Pr(\hat{B}_k \neq B_k). \tag{3.86}$$

Remark 2. For uniform bit levels, (3.84) becomes

$$R_{\text{BMD}}^{\text{HD}} = m(1 - \text{H}_2(\delta_{\text{BMD}})) \tag{3.87}$$

which is an achievable rate for transmitting over m parallel BSC channels with error probability δ_{BMD}.

The numerical evaluation of the achievable rates is shown in Fig. 3.26 for 4-ASK, 8-ASK and 16-ASK. Note that the BMD rates for HD are better than their SMD counterparts for both uniform and shaped signaling. This result was also observed in [104] and is because no reliability information associated with the individual transition probabilities may be exploited.



(a) 4-ASK

(b) 8-ASK

(c) 16-ASK

Figure 3.26.: Comparison of HD achievable rates for 4, 8 and 16-ASK.

Another perspective is provided in Fig. 3.27, where the gap to the AWGN capacity is plotted. In contrast to the SD scenario of Fig. 3.5, we observe that each constellation size should only be operated for a certain SNR regime. Again, this is because the decoder can not make use of any reliability information in the HD context. For shaped signaling the gap to AWGN capacity can be reduced to about 1.8 dB over the whole SNR range with the optimal signaling.

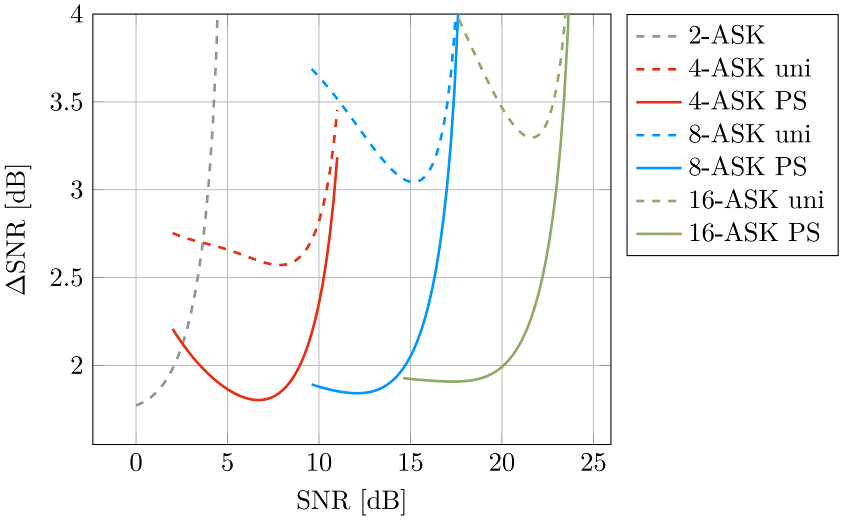


Figure 3.27.: Comparison of HD BMD gaps to AWGN capacity.

3.8.3. Optimal Code Rate and Constellation Size

As in Sec. 3.4.3, we investigate the optimal FEC rates $R_{\mathrm{c}}^{\star}$ for HD BMD. We show the numerical evaluations of

$$R_{\mathrm{c}}^{\star} = 1 - \frac{1}{m}\,\mathrm{U}\left(q_{\mathrm{BMD}}^{\mathrm{HD}}\right)$$

in Fig. 3.28 where the optimal distribution is used for each SNR. For 8-ASK and 16-ASK, we identify operating regimes with an almost constant mismatched uncertainty suggesting that using a fixed FEC code rate is close-to-optimal here. We summarize those FEC rates in Table 3.6 and observe that those close-to-optimal, fixed FEC code rates are larger than in the SD case of Table 3.1.

3.8.4. Coded Performance with Product Codes

In this section, we investigate how the predicted asymptotic gains translate into real gains for a coded system. For FEC, we resort to a product code (PC) with BCH component codes, see Sec. 2.3.5. The component codes are decoded iteratively by bounded distance decoding (BDD). In our case, BDD is implemented by the Berlekamp-Massey algorithm [32].

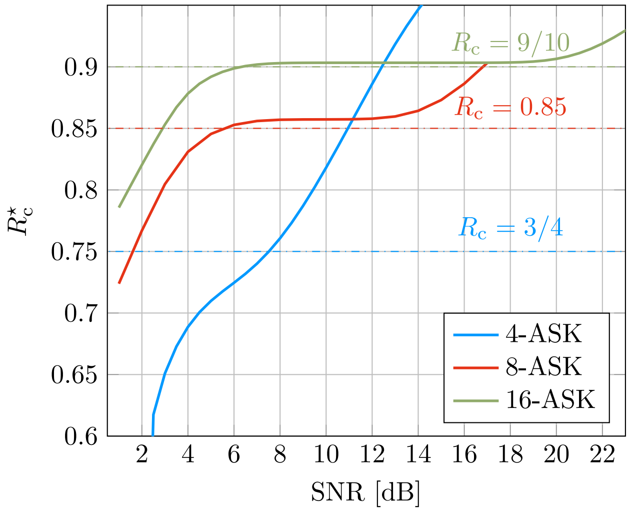


Figure 3.28.: Optimal $R_c^\star$ for PAS with 4-ASK, 8-ASK and 16-ASK with HD and BMD.

M-ASK	R_c
4	0.75
8	0.85
16	0.90

Table 3.6.: Close-to-optimal, fixed FEC code rates for HD BMD and a given constellation order.

In Fig. 3.29, we depict a scenario for 8-ASK. The BCH code has parameters $(255, 239, 2)$[6] such that the PC has an overall rate of $R_c = (239/255)^2 = 0.8784$, which is close to the optimal one with 8-ASK (see Fig. 3.28) for the desired operating range. The blocklength is $n_c = 65\,025$ bits. We perform 20 iterations between the row and column component codes. The target FER is 10^{-3}. We obtain the different operating points by varying the DM rate between 0.3647 bits/symbol to 1.8647 bits/symbol ($\gamma = 0.6353$). We also show results for uniform scenarios (using the component codes BCH(255, 179) with $R_c = 0.4927$, BCH(255, 207) with $R_c = 0.6590$ and BCH(255, 231) with $R_c = 0.8206$) and observe that the PAS operating points have a significantly reduced gap to the achievable rate over the whole range of transmission rates. This is somewhat surprising as the uniform signaling modes use codes of much lower code rate which also have a larger error correction capability.

[6] This code was also proposed as part of the G.975 standard for forward error correction for submarine systems [105].

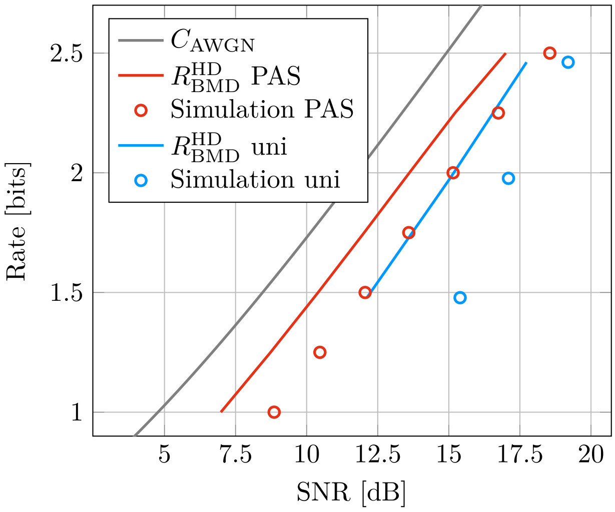


Figure 3.29.: Comparison of $R_{\mathrm{BMD}}^{\mathrm{HD}}$ and coded performance of a PC with rate $R_{\mathrm{c}} = 0.8784$ and blocklength 65 025 for 8-ASK. The target FER is 10^{-3}.

3.9. Performance Comparison of DM Algorithms for Short Blocklengths

In Sec. 3.4.2 different DM architectures and their properties were discussed. For short blocklength scenarios, their rate loss (3.34) must be taken into account, as any potential gains from PS may vanish for short blocks. We illustrate the rate loss of CCDM and SMDM in Fig. 3.30. SMDM improves upon CCDM for short blocklengths and the insights

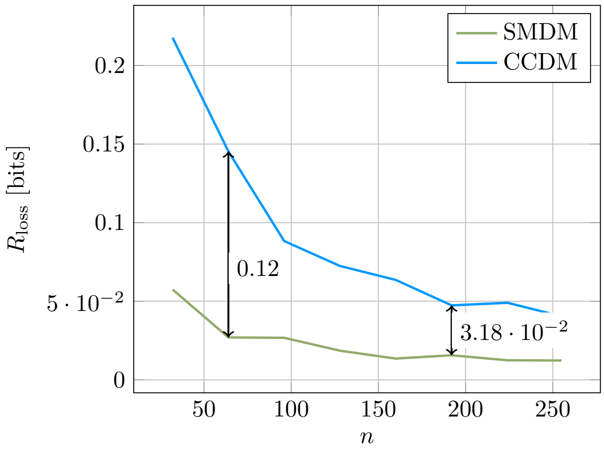


Figure 3.30.: Performance comparison of CCDM and SMDM for the considered setting. The DM rate is 1.25 bpcu, the output alphabet is 4-ary.

of this investigation allow a quick assessment of possible gains in a coded scenario. For this, we will review two different short blocklength setups for $n = 64$ and $n = 192$ channel

Algorithm 2 Ordered Statistics Decoding.

```
INPUT: Generator matrix $\boldsymbol{G}$, soft information $\boldsymbol{l}_{\mathrm{dec}} = (l_{11}l_{12}\dots l_{1m}\dots l_{n(m-1)}l_{nm})$.
Sort according to reliability: $\boldsymbol{l}_1 = \mathsf{sort}(|\boldsymbol{l}_{\mathrm{dec}}|, \texttt{'descend'}) \rightarrow$ permutation $\boldsymbol{\pi}_1$
Reorder columns of $\boldsymbol{G}$: $\boldsymbol{G}_1 = \boldsymbol{G}(:, \boldsymbol{\pi}_1)$.
if rank$(\boldsymbol{G}_1(:, 1 : k_{\mathrm{c}})) < k_{\mathrm{c}}$ then
    Apply additional permutation $\boldsymbol{\pi}_2$ such that $\boldsymbol{G}_2$ has full rank: $\boldsymbol{G}_2 = \boldsymbol{G}_1(:, \boldsymbol{\pi}_2)$.
    $\boldsymbol{l}_2 = \boldsymbol{l}_1(\boldsymbol{\pi}_2)$.
end if
Determine information bits of most reliable basis: $\boldsymbol{u} = (\boldsymbol{l}_2(1 : k_{\mathrm{c}}) \leq 0)$.
Build set of error patterns up to weight $t$:
    $$\mathcal{E}_t = \left\{\boldsymbol{e} \in \{0,1\}^{k_{\mathrm{c}}} : w_{\mathrm{H}}(\boldsymbol{e}) \leq t\right\}.$$
Build list of possible codewords as:
    $$\mathcal{L} = \left\{\boldsymbol{c} \in \{0,1\}^{n_{\mathrm{c}}} : \boldsymbol{c} = (\boldsymbol{u} + \boldsymbol{e})\boldsymbol{G}_2, \quad \forall \boldsymbol{e} \in \mathcal{E}_t\right\}.$$
Determine (permuted) codeword estimate $\hat{\boldsymbol{c}}_2$ as:
    $$\hat{\boldsymbol{c}}_2 = \operatorname*{argmax}_{c \in \mathcal{L}} \sum_{i=1}^{n_{\mathrm{c}}} [\boldsymbol{l}_2]_i \cdot (1 - 2c_i).$$
Codeword estimate is $\hat{\boldsymbol{c}}(\boldsymbol{\pi}_1(\boldsymbol{\pi}_2)) = \hat{\boldsymbol{c}}_2$.
```

uses. Finite blocklength bounds from Sec. 2.3.7 will complement this investigation.

To focus on the DM properties and put aside the influence of the FEC decoder, the receiver for the case $n = 64$ employs ordered statistics decoding (OSD). OSD is a SD decoding algorithm for any linear blockcode that was conceived by Dorsch in 1974 [106] and later rediscovered by Fossorier and Lin [107]. The algorithm aims at finding the most reliable information basis and adds small weight error patterns up to Hamming weight t to find the most likely codeword. If the OSD order t is chosen large enough, ML decoding performance can be achieved. The procedure is summarized in Algorithm 2.

In Fig. 3.31, we present the results for a target SE of 1.5 bpcu with 8-ASK. The parameters of the employed codes $\mathcal{C}_1$ and $\mathcal{C}_2$ (derived from BCH codes) are summarized in Table 3.7a and 3.7b. The best known linear codes for the chosen parameters have $d_{\min} = 14$ $(\mathcal{C}_1)$[7] and $d_{\min} = 28$ $(\mathcal{C}_2)$[8], respectively. In [107], the authors derive a condition for the binary-input additive white Gaussian noise (biAWGN) channel that relates the minimum distance of the code to the required OSD parameter t_{ML} to obtain near ML performance

[7] `http://codetables.markus-grassl.de/BKLC/BKLC.php?q=2&n=192&k=144`

[8] `http://codetables.markus-grassl.de/BKLC/BKLC.php?q=2&n=192&k=96`

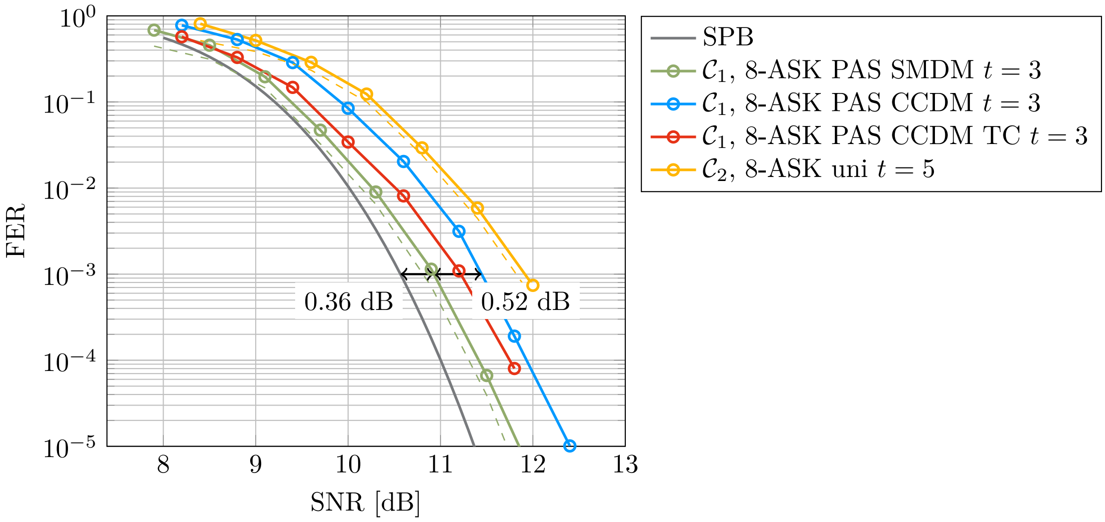


Figure 3.31.: Performance of OSD for 8-ASK with $R_{\mathrm{tx}} = 1.5\,\mathrm{bpcu}$ and $n = 64$. The respective ML lower bounds are shown with dashed lines.

in the high SNR regime, which reads as

$$t_{\mathrm{ML}} \approx \left\lceil \min\left(\frac{d_{\min}}{4} - 1, k_{\mathrm{c}}\right)\right\rceil . \tag{3.88}$$

From numerical evaluations, we conjecture that this rule of thumb also holds for BMD. Consequently, we choose $t = 3$ for $\mathcal{C}_1$ and $t = 5$ for $\mathcal{C}_2$ such that the depicted results correspond to the ML decoding performance of the respective codes.

For the shaped case, we compare two DM scenarios. The first employs a CCDM with parameters summarized in Table 3.8a, whereas the second uses an SMDM with parameters shown in Table 3.8b.

At an FER of 10^{-3} , PAS with SMDM performs within a gap of 0.3 dB to the SPB, while CCDM is 0.5 dB less power efficient due to its significant rate loss at this short blocklength. We can already observe this from the rate loss analysis shown in Tables 3.8a and 3.8b. Using our rule of thumb of (3.62) to convert the rate loss to an SNR loss, we can expect CCDM to be $0.69\,\mathrm{dB} - 0.24\,\mathrm{dB} \approx 0.45\,\mathrm{dB}$ less power efficient, which is reflected in the coded performance.

We further note that the SPB (2.119) is only a converse for the CCDM setting, as the shaping set of CCDM is indeed a spherical code, whereas SMDM uses points within the $n = 64$ dimensional sphere. The impact of the rate loss is alleviated to some extent by exploiting a type check (TC) during the evaluation of the OSD list candidates in $\mathcal{L}$: If the codeword with the highest score does not pass the type check imposed by the type $\boldsymbol{t}^n_{\mathcal{A}}$, we choose the candidate with the next highest score fulfilling the TC. If there is no candidate at all in $\mathcal{L}$, we randomly return a FEC codeword, which fulfills the TC. We see that the TC is particularly helpful at low SNRs.

	Parameter	Value
BCH code	n_c	255
	k_c	147
	$d_\mathrm{min,d}$	29
$\mathcal{C}_1$	n_c	192
	k_c	144
	R_c	3/4
	d_min	14

(a) Parameters of code $\mathcal{C}_1$

	Parameter	Value
BCH code	n_c	255
	k_c	99
	$d_\mathrm{min,d}$	47
$\mathcal{C}_2$	n_c	192
	k_c	96
	R_c	1/2
	d_min	23

(b) Parameters of code $\mathcal{C}_2$

Table 3.7.: Code parameters of the employed codes for OSD.

Parameter	Value
k_dm	80
n	64
R_dm	1.25
$\boldsymbol{t}_\mathcal{A}^n$	$\{37, 21, 5, 1\}$
P_A	$(0.578, 0.328, 0.078, 0.016)$
R_loss	0.1157 bits

(a) CCDM

Parameter	Value
k_dm	80
n	64
R_dm	1.25
$W(a)$	a^2
P_A	$(0.612, 0.306, 0.074, 0.008)$
R_loss	0.0400 bits

(b) SMDM

Table 3.8.: DM parameters for $n = 64$.

In Fig. 3.32, we again show a scenario for 8-ASK and an SE of 1.5 bpcu, but with a larger number of $n = 192$ channel uses and LDPC codes. In contrast to before, we only compare SMDM and CCDM without a TC, as the latter is not easily integrated in the BP decoding process. The DM parameters are summarized for both cases in Tables 3.9a and 3.9b.

The LDPC codes are taken from the 5G standard [101], see also Sec. 4.1.2. The codes for PAS have a rate of $R_\mathrm{c} = 3/4$ and are derived from basegraph 1. The code for the uniform scenario has rate $R_\mathrm{c} = 1/2$ and is obtained from basegraph 2. 200 BP iterations are performed. At an FER of 10^{-3}, the performance gap between SMDM and CCDM is less pronounced than before and about 0.2 dB. Again, this is expected from the rate loss analysis in Fig. 3.30 and Table 3.9, which predicts an SMDM improvement of 0.19 dB. The gain over uniform signaling is about 1 dB.

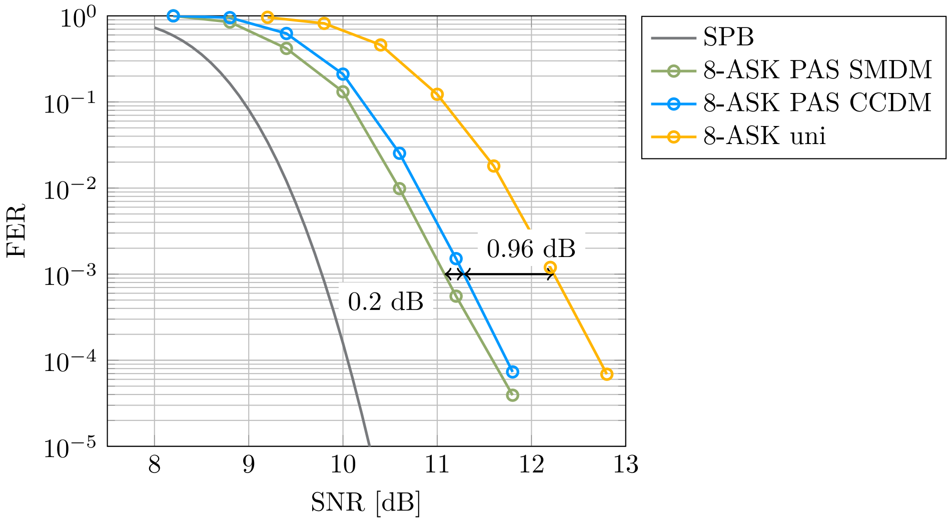


Figure 3.32.: Performance of 5G LDPC codes for 8-ASK with $R_{\text{tx}} = 1.5$ bpcu and $n = 192$.

Parameter	Value
k_{dm}	240
n	192
R_{dm}	1.25
$\boldsymbol{t}^n_{\mathcal{A}}$	$\{118, 58, 14, 2\}$
P_A	$(0.615, 0.302, 0.073, 0.010)$
R_{loss}	0.0474 bits

(a) CCDM

Parameter	Value
k_{dm}	240
n	192
R_{dm}	1.25
$W(a)$	a^2
P_A	$(0.622, 0.301, 0.070, 0.007)$
R_{loss}	0.0156 bits

(b) SMDM

Table 3.9.: DM parameters for $n = 192$.

3.10. Channels with a Non-Symmetric, Capacity Achieving Input Distribution

In this section, we examine a simple example where PAS cannot synthesize the optimal input distribution because it is not symmetric. We consider OOK, a non-coherent modulation scheme that is important for optical communications such as free space optical communication (FSO) communications where simple transceiver architectures are required.

OOK with a uniform distribution has a significant energy loss compared to optimal signaling. Therefore, many practical implementations resort to pulse position modulation (PPM). However, PPM requires SMD for good performance, i.e., the FEC decoder must operate on the whole PPM symbol. If binary codes with BMD are considered, a significant performance loss is observed. This is illustrated in Fig. 3.33a, where a gap of 1.66 dB between OOK with a capacity achieving input distribution and 8-PPM with BMD at a rate of 0.2 bpcu is visible.

3.10.1. System Model and Optimal Signaling for On-Off Keying

We consider the system model of (3.1) with the channel input $\mathcal{X} = \{0, A\}$. The average signal power is $\mathrm{E}[X^2] = A^2 P_X(A)$. An achievable rate is given by the MI $\mathrm{I}(X;Y)$ and the maximum achievable rate is the solution of the following optimization problem

$$C_{\mathrm{OOK}} = \max_{P_X} \quad \mathrm{I}(X;Y) \quad \text{subject to} \quad A^2 P_X(A) \leq P. \tag{3.89}$$

We refer to (3.89) as the "OOK capacity", which is shown in Fig. 3.33a. Note that average power constraint is always active, so that the amplitude is $A = \sqrt{P/P_X(A)}$. If a uniform distribution is chosen, i.e., $P_X(0) = P_X(A) = 0.5$, we observe a significant degradation in power efficiency. In Fig. 3.33b, we show the optimal pulse probability $P_X(A)$ and pulse amplitude A as a result of the optimization in (3.89). We observe that the optimal distribution is heavily biased in the low and medium SNR range, which gives an insight why uniform signaling is particularly harmful here.

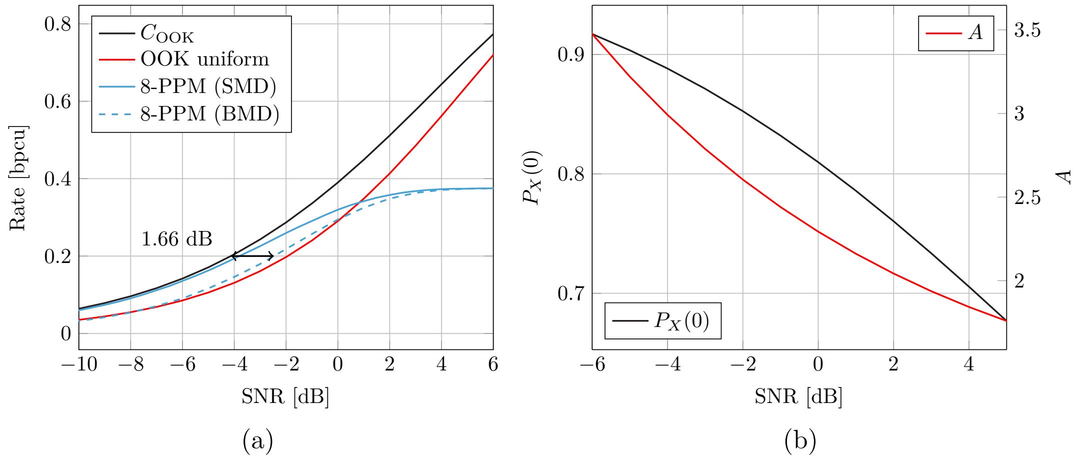


Figure 3.33.: Achievable rates for OOK and PPM. The right figure shows the optimal probability $P_X(0)$ and pulse position A to achieve the OOK capacity.

3.10.2. Probabilistic Shaping via Time Sharing

In the following, we employ the sparse-dense scheme of Sec. 3.2.2 with a $(n_\mathrm{c}, k_\mathrm{c})$ linear blockcode. We distinguish between a modulated information symbol X_S and a modulated parity symbol X_U. We use the signal set $\mathcal{X}_\mathrm{S} = \{0, A_\mathrm{S}\}$ for the information part, i.e., for a number of $k_\mathrm{c} = R_\mathrm{c} \cdot n_\mathrm{c}$ channel uses. For the remaining $(1 - R_\mathrm{c}) \cdot n_\mathrm{c}$ channel uses involving the parity bits, the signal set is $\mathcal{X}_\mathrm{U} = \{0, A_\mathrm{U}\}$.

A CCDM realizes the non-uniformly distributed symbols. It encodes k_{dm} uniformly distributed bits into a length k_{c} shaped information bit sequence which is then FEC encoded. The DM has rate $R_{\mathrm{dm}} = k_{\mathrm{dm}}/k_{\mathrm{c}}$ which approaches the entropy of the DM output distribution (see Sec. 3.4.2) for long k_{c}. Therefore, we have $R_{\mathrm{dm}} = \mathrm{H}(X_{\mathrm{S}})$ and the overall transmission rate is

$$R_{\mathrm{tx}} = \mathrm{H}(X_{\mathrm{S}}) \cdot R_{\mathrm{c}}. \tag{3.90}$$

Thus, R_{tx} is directly related to $P_{X_{\mathrm{S}}}(A_{\mathrm{S}})$ via

$$P_{X_{\mathrm{S}}}(A_{\mathrm{S}}) = \mathrm{H}^{-1}\left(\frac{R_{\mathrm{tx}}}{R_{\mathrm{c}}}\right). \tag{3.91}$$

An achievable rate of the TS scheme is given by

$$R_{\mathrm{TS}} = R_{\mathrm{c}}\,\mathrm{I}(X_{\mathrm{S}};Y_{\mathrm{S}}) + (1-R_{\mathrm{c}})\,\mathrm{I}(X_{\mathrm{U}};Y_{\mathrm{U}}). \tag{3.92}$$

From (3.90), reliable communication is guaranteed as long as $R_{\mathrm{tx}} \leq R_{\mathrm{TS}}$. In the following, we distinguish two cases.

Case 1: Same Pulse Amplitudes Consider the case where both pulse amplitudes are the same, i.e., $A_{\mathrm{S}} = A_{\mathrm{U}} = A$. The average signal power is

$$\mathrm{E}\left[R_{\mathrm{c}}X_{\mathrm{S}}^2 + (1-R_{\mathrm{c}})X_{\mathrm{U}}^2\right] = \left(R_{\mathrm{c}}P_{X_{\mathrm{S}}}(A) + (1-R_{\mathrm{c}})\frac{1}{2}\right)A^2 \tag{3.93}$$

and the optimization problem for (3.92) is

$$R^*_{\mathrm{TS}_1} = \max_{P_{X_{\mathrm{S}}},A} R_{\mathrm{TS}} \quad \text{subject to} \quad \left(R_{\mathrm{c}}P_{X_{\mathrm{S}}}(A) + (1-R_{\mathrm{c}})\frac{1}{2}\right)A^2 \leq P. \tag{3.94}$$

As for (3.89), the power constraint is always active. Thus, for a fixed $P_{X_{\mathrm{S}}}(A)$ we have $A = P/\sqrt{R_{\mathrm{c}}P_{X_{\mathrm{S}}}(A) + (1-R_{\mathrm{c}})/2}$.

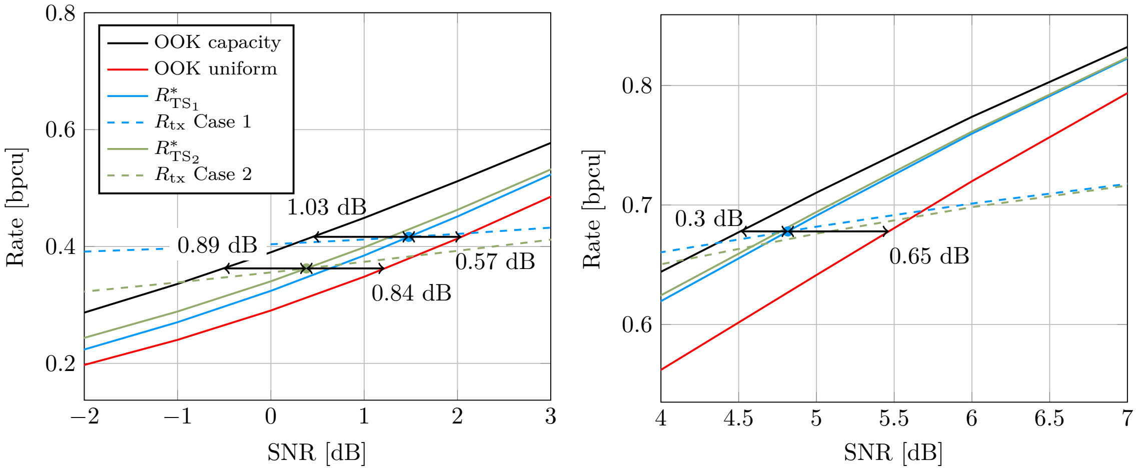
Case 2: Individual Pulse Amplitudes We now permit different pulse amplitudes A_{S} and A_{U}. The average signal power is

$$\mathrm{E}\left[R_{\mathrm{c}}X_{\mathrm{S}}^2 + (1-R_{\mathrm{c}})X_{\mathrm{U}}^2\right] = R_{\mathrm{c}}P_{X_{\mathrm{S}}}(A_S)A_{\mathrm{S}}^2 + (1-R_{\mathrm{c}})\frac{1}{2}A_{\mathrm{U}}^2. \tag{3.95}$$

Similar to the first case, we have

$$R^*_{\mathrm{TS}_2} = \max_{P_{X_{\mathrm{S}}},A_{\mathrm{S}},A_{\mathrm{U}}} R_{\mathrm{TS}} \quad \text{subject to} \quad R_{\mathrm{c}}P_{X_{\mathrm{S}}}(A_S)A_{\mathrm{S}}^2 + (1-R_{\mathrm{c}})\frac{1}{2}A_{\mathrm{U}}^2 \leq P. \tag{3.96}$$

Again, the average power constraint is always active.



(a) $R_{\mathrm{c}} = 0.5$

(b) $R_{\mathrm{c}} = 0.75$

Figure 3.34.: Achievable rates for the TS schemes with different code rates R_{c}.

3.10.3. Comparison of Achievable Rates via Time Sharing

We plot the achievable rates for both TS schemes in Fig. 3.34 for the code rates $R_{\mathrm{c}} = 0.5$ and $R_{\mathrm{c}} = 0.75$. The dashed curves show the transmission rates (3.90) with the optimized pulse probabilities $P_{X_{\mathrm{S}}}(A)$ and $P_{X_{\mathrm{S}}}(A_{\mathrm{S}})$ according to (3.94) and (3.96), respectively. The crossing of the R_{TS} and R_{tx} curves indicates the optimal operating points for the chosen code rates. Comparing (3.89) and (3.92), we observe that using a low code rate, e.g., $R_{\mathrm{c}} = 0.5$ in Fig. 3.34, increases the gap to C_{OOK} as the fraction of transmission symbols with a uniform distribution also increases with lower R_{c}. The gap to the OOK capacity is about 1.0 dB for $R_{\mathrm{c}} = 0.5$, while it reduces to 0.3 dB for a rate $R_{\mathrm{c}} = 0.75$ code at the respective optimal transmission rates given by (3.90). These results motivate using a high rate code, even for low transmission rates. This requires using a pulse probability different from the optimal one from (3.94) or (3.96). For example, consider the first TS scheme. In order to operate at $R_{\mathrm{tx}} = 0.25$ bpcu as in Fig. 3.34 (a), instead of $R_{\mathrm{c}} = 0.5$ we may use $R_{\mathrm{c}} = 0.75$ with $P_{X_{\mathrm{S}}}$ directly given by (3.91). We can also see that the gains of the TS scheme 2 vanish for transmission rates larger than about 0.5 bpcu.

3.10.4. Signaling for Fixed Transmission and FEC Code Rates

As pointed out in Sec. 3.10.2, for a target transmission rate R_{tx} and fixed code rate R_{c}, the probability $P_{X_{\mathrm{S}}}(A_{\mathrm{S}})$ is given by (3.91). Thus, for the first TS scheme, the average power constraint in (3.93) determines A and there are no additional degrees of freedom for the optimization in (3.94). The second TS scheme has an additional degree of freedom by optimizing over either A_{S} or A_{U}. A practical communication scheme uses a family of channel codes of different rates. For each target rate R_{tx} we choose the code rate such that the required SNR is minimized. We proceed as follows.

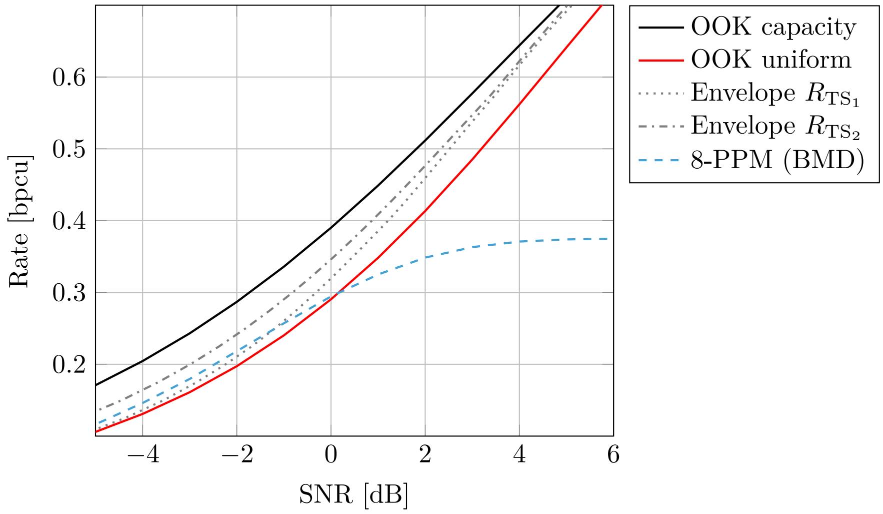


Figure 3.35.: Achievable rates for TS schemes with fixed code rates from the set $\mathcal{R}_\mathrm{c}$.

1. Consider a set $\mathcal{R}_\mathrm{c}$ of code rates.

2. For a target R_tx, determine the required SNR for all possible $R_\mathrm{c} \in \mathcal{R}_\mathrm{c}$, such that $R_\mathrm{tx} = R^*_{\mathrm{TS}_i}, i \in \{1,2\}$. Since R_tx is fixed, for a specified R_c the pulse probabilities $P_{X_\mathrm{S}}(A)$ (for R_{TS_1}) and $P_{X_\mathrm{S}}(A_S)$ (for R_{TS_2}) are obtained from (3.91).

3. Among all $R_\mathrm{c} \in \mathcal{R}_\mathrm{c}$, use the code rate R^*_c that requires the smallest SNR.

As an example, consider the set of code rates $\mathcal{R}_\mathrm{c} = \{0.25, 0.33, 0.5, 0.67, 0.75, 0.8, 0.9\}$. For different transmission rates in the range $0.2\,\mathrm{bpcu} \leq R_\mathrm{tx} \leq 0.85\,\mathrm{bpcu}$ we determine the required SNR for the code rates in $\mathcal{R}_\mathrm{c}$, and choose for each R_tx the code rate R^*_c with the lowest SNR requirement. Table 3.10 gives an overview of the code rates R^*_c for some R_tx. The gray colored curves in Fig. 3.35 represent the corresponding achievable rates versus SNR for the first and second TS schemes using code rates from Tab. 3.10. Observe from the table that for the second TS scheme it is beneficial to use high code rates, even if low transmission rates are targeted. In Sec. 4.9, we optimize LDPC codes for the proposed TS schemes.

3.11. Overview over Decoding Metrics and Achievable Rates

In Fig. 3.36 we summarize practically relevant FEC decoding metrics, corresponding achievable rates, their estimators for MC based evaluations, and implementation examples by means of different code classes. The overview discusses all examples in the previous sections. Apart from SMD and BMD, the class of multistage decoding metrics (in combination with multilevel coding) [108] is also important in practice. For instance, for

R_{tx}	R_{c}^* TS$_1$	R_{c}^* TS$_2$
0.20	0.33	0.67
0.25	0.50	0.67
0.33	0.5	0.67
0.50	0.67	0.67
0.67	0.75	0.80
0.75	0.80	0.80
0.85	0.90	0.90

Table 3.10.: Code rates R_{c}^* for some R_{tx}.

polar codes [6, 7] a multilevel coding/multistage decoding approach is natural for higher-order modulation because of the successive cancelation decoding [109, 110]. Multilevel coding/multistage decoding is possible for SD and HD decoding metrics.

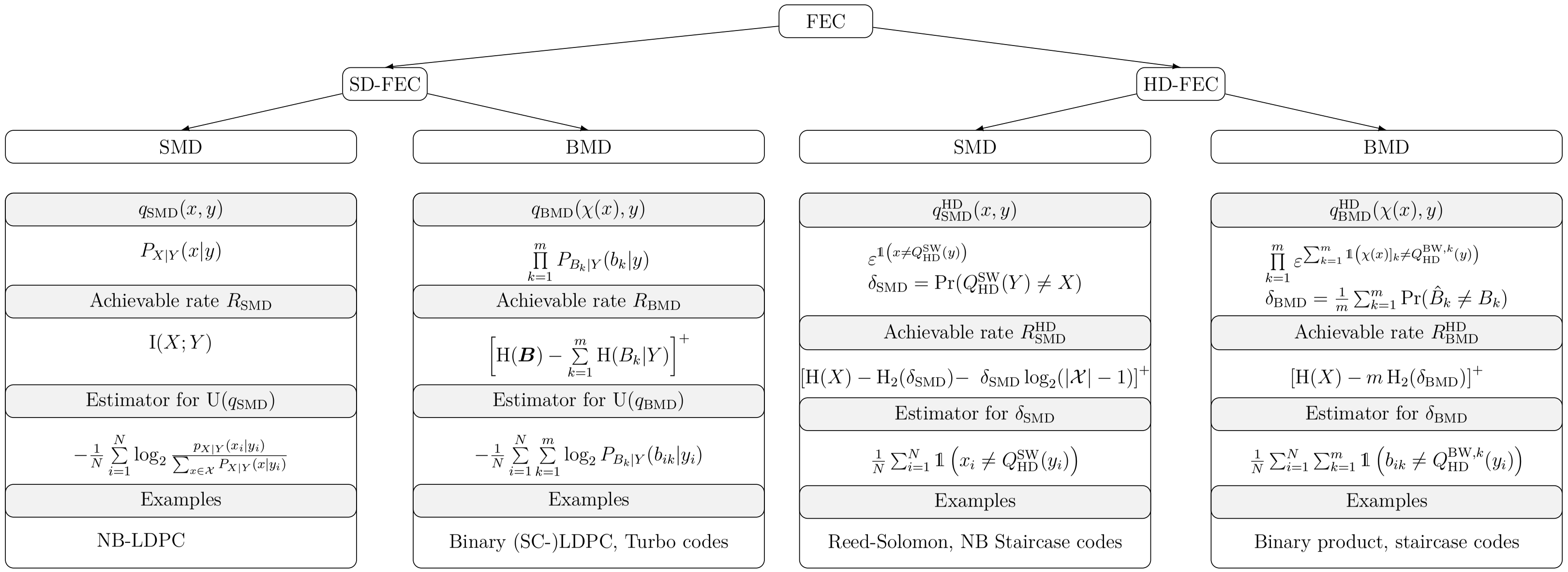

FEC
SD-FEC
HD-FEC
SMD
BMD
SMD
BMD
$q_{\text{SMD}}(x,y)$
$P_{X|Y}(x|y)$
Achievable rate R_{SMD}
$\text{I}(X;Y)$
Estimator for $\text{U}(q_{\text{SMD}})$
$-\frac{1}{N}\sum_{i=1}^{N}\log_2\frac{p_{X|Y}(x_i|y_i)}{\sum_{x\in\mathcal{X}}P_{X|Y}(x|y_i)}$
Examples
NB-LDPC
$q_{\text{BMD}}(\chi(x),y)$
$\prod_{k=1}^{m}P_{B_k|Y}(b_k|y)$
Achievable rate R_{BMD}
$\left[\text{H}(\boldsymbol{B})-\sum_{k=1}^{m}\text{H}(B_k|Y)\right]^+$
Estimator for $\text{U}(q_{\text{BMD}})$
$-\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{m}\log_2 P_{B_k|Y}(b_{ik}|y_i)$
Examples
Binary (SC-)LDPC, Turbo codes
$q_{\text{SMD}}^{\text{HD}}(x,y)$
$\varepsilon^{\mathbb{1}\left(x\neq Q_{\text{HD}}^{\text{SW}}(y)\right)}$
$\delta_{\text{SMD}}=\Pr(Q_{\text{HD}}^{\text{SW}}(Y)\neq X)$
Achievable rate $R_{\text{SMD}}^{\text{HD}}$
$[\text{H}(X)-\text{H}_2(\delta_{\text{SMD}})-\ \delta_{\text{SMD}}\log_2(|\mathcal{X}|-1)]^+$
Estimator for δ_{SMD}
$\frac{1}{N}\sum_{i=1}^{N}\mathbb{1}\left(x_i\neq Q_{\text{HD}}^{\text{SW}}(y_i)\right)$
Examples
Reed-Solomon, NB Staircase codes
$q_{\text{BMD}}^{\text{HD}}(\chi(x),y)$
$\prod_{k=1}^{m}\varepsilon^{\sum_{k=1}^{m}\mathbb{1}\left(\chi(x)]_k\neq Q_{\text{HD}}^{\text{BW},k}(y)\right)}$
$\delta_{\text{BMD}}=\frac{1}{m}\sum_{k=1}^{m}\Pr(\hat{B}_k\neq B_k)$
Achievable rate $R_{\text{BMD}}^{\text{HD}}$
$[\text{H}(X)-m\,\text{H}_2(\delta_{\text{BMD}})]^+$
Estimator for δ_{BMD}
$\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{m}\mathbb{1}\left(b_{ik}\neq Q_{\text{HD}}^{\text{BW},k}(y_i)\right)$
Examples
Binary product, staircase codes

Figure 3.36.: Overview over decoding metrics and achievable rates.

4

Code Design for Binary Low-Density Parity-Check Codes

4.1. Introduction

Binary LDPC codes are binary linear block codes defined by an $m_\mathrm{c} \times n_\mathrm{c}$ sparse parity-check matrix $\boldsymbol{H}$. The code dimension[1] is $k_\mathrm{c} = n_\mathrm{c} - \mathrm{rank}(\boldsymbol{H})$. An LDPC code can also be described by its Tanner graph [111], which is a bipartite graph $G = (\mathcal{V} \cup \mathcal{C}, \mathcal{E})$ having n_c VNs $\mathsf{v}_j \in \mathcal{V}$ and m_c FNs $\mathsf{c}_i \in \mathcal{C}$ (see Sec. 2.4.1). In the context of LDPC codes, the FNs are also referred to as check nodes (CNs). The set $\mathcal{E}$ of edges contains the element e_{ij} if and only if the parity-check matrix element h_{ij} (entry in the i-th row and j-th column of $\boldsymbol{H}$) is equal to 1. The *degree* of a VN v_j is denoted by d_{v_j} and it is the cardinality of the set $\mathcal{N}(\mathsf{v}_j)$. Similarly, the degree of a CN c_i is denoted by d_{c_i} and it is the cardinality of the set $\mathcal{N}(\mathsf{c}_i)$. The Tanner graph of an exemplary LDPC is shown in Fig. 4.1.

To analyze the properties and characteristics of LDPC codes in an asymptotic setting, Gallager introduced the notion of *ensembles*. In the following, we analyze two different ensemble types.

4.1.1. Unstructured Ensembles

Unstructured LDPC code ensembles are characterized by their VN and CN degree distributions. We distinguish two perspectives, namely the *node and edge perspective*. For the

[1] This definition is important for regular LDPC codes (see Sec. 4.1.1), which are rank deficient by construction, such that $\mathrm{rank}(\boldsymbol{H}) < m_\mathrm{c}$.

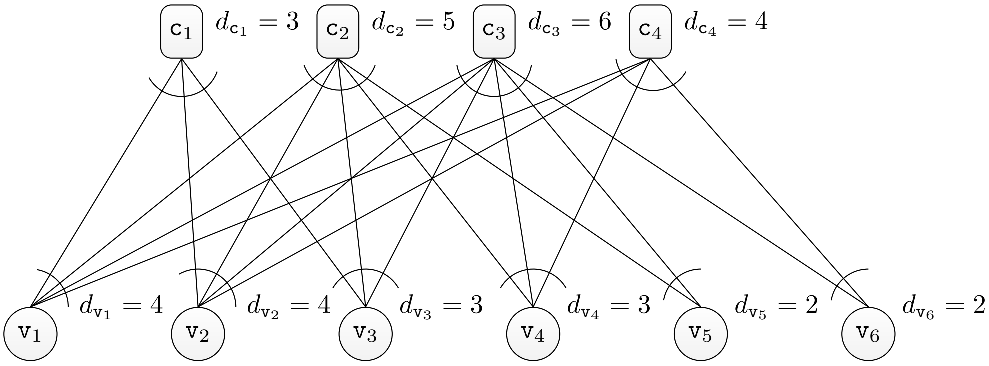


Figure 4.1.: Illustration of a Tanner graph with six VNs and four CNs.

node perspective, we have the VN and CN *degree polynomials*

$$\Lambda(x) = \sum_{i=1}^{d_{\mathrm{v,max}}} \Lambda_i x^i \qquad R(x) = \sum_{i=1}^{d_{\mathrm{c,max}}} R_i x^i. \tag{4.1}$$

The coefficients Λ_i and R_i denote the fraction of VNs in $\mathcal{V}$ and CNs in $\mathcal{C}$ that have degree i. Similarly, for the edge perspective, we have

$$\lambda(x) = \sum_{i=1}^{d_{\mathrm{v,max}}} \lambda_i x^{i-1} \qquad \rho(x) = \sum_{i=1}^{d_{\mathrm{c,max}}} \rho_i x^{i-1}. \tag{4.2}$$

Here, the coefficients λ_i and ρ_i denote the fraction of edges in $\mathcal{E}$ connected to degree i VNs and CNs, respectively. We note that the above descriptions are in their most general form. For practical LDPC codes, more constraints have to be imposed to get meaningful codes. For instance, we allow degree one VNs above, but they have to be used with care. In an unstructured ensemble, two degree one VNs may be connected to the same degree two CN, giving rise to a code with minimum distance of two.

Example 8. For the Tanner graph in Fig. 4.1, we have

$$\Lambda(x) = \frac{2}{6}x^2 + \frac{2}{6}x^3 + \frac{2}{6}x^4 \qquad R(x) = \frac{1}{4}x^3 + \frac{1}{4}x^4 + \frac{1}{4}x^5 + \frac{1}{4}x^6. \tag{4.3}$$

The edge perspective degree distributions are

$$\lambda(x) = \frac{4}{18}x + \frac{6}{18}x^2 + \frac{8}{18}x^3 \qquad \rho(x) = \frac{3}{18}x^2 + \frac{4}{18}x^3 + \frac{5}{18}x^4 + \frac{6}{18}x^5. \tag{4.4}$$

Both descriptions can be transformed into each other. For example, we have

$$\lambda(x) = \frac{1}{L'(1)}L'(x) \qquad \rho(x) = \frac{1}{R'(1)}R'(x). \tag{4.5}$$

The design rate $R_{\mathrm{c,d}}$ of an LDPC code is given by

$$R_{\mathrm{c,d}} = 1 - \frac{L'(1)}{R'(1)} = 1 - \frac{\bar{d}_{\mathtt{v}}}{\bar{d}_{\mathtt{c}}} \tag{4.6}$$

where $\bar{d}_{\mathtt{v}}$ and $\bar{d}_{\mathtt{c}}$ denote the average VN and CN degree, respectively.

Decoding Complexity

To evaluate the decoding complexity of a given LDPC code, the internal decoder data flow F is an important metric. It is defined as the number of bits that are processed in each BP iteration and can be calculated as

$$F = \frac{2 \cdot k_{\mathrm{c}} \cdot q \cdot \bar{d}_{\mathtt{v}}}{R_{\mathrm{c}}} = 2 \cdot n_{\mathrm{c}} \cdot q \cdot \bar{d}_{\mathtt{v}} \tag{4.7}$$

where q is the number of bits used to represent a message sent on a given edge.

Role of Degree Two Variable Nodes

The number of degree two VNs plays an important role for the design of good LDPC codes. On the one hand, a certain fraction of degree two VNs is necessary for capacity approaching decoding thresholds [112, 113]. On the other hand, a too large fraction is harmful for practical codes:

- ▷ A large fraction of degree two VNs may violate the stability condition, which is a necessary condition for the convergence of the BP algorithm and ensures that the minimum distance of the specified ensemble grows linearly with the blocklength [114] for unstructured, irregular codes.
- ▷ Degree two VNs which form a cycle constitute the support of a codeword. Hence, a lot of degree two VNs may lead to a small minimum distance if they are not placed and connected judiciously[2]. For instance, a length 8 cycle between degree two VNs results in a codeword with $d_{\mathrm{min}} = 4$.

Any tailored code design needs to take these conflicting requirements into account.

4.1.2. Structured Ensembles and Protographs

For practical purposes, it is beneficial to impose more structure on an LDPC code ensemble. Examples of structured LDPC code ensembles are multi-edge type (MET) [116] and protograph based ensembles [117, 118]. Protograph based ensembles are defined via a (typically small) basematrix $\boldsymbol{B}$ of dimension $m_{\mathrm{p}} \times n_{\mathrm{p}}$ with elements $b_{ij} \in [0, 1, \ldots, b_{\mathrm{max}}]$. A

[2] One way to avoid cycles between degree two VNs is to place them in the form of a double diagonal in the parity-check matrix. *Repeat-accumulate* codes [115] constitute an important LDPC code class which adopts this property. This structure is also beneficial for low complexity encoding.

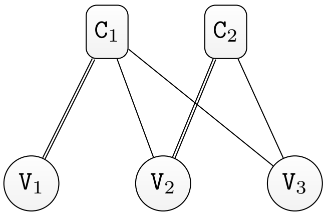


Figure 4.2.: Tanner graph of a protograph.

basematrix may also be represented as a bipartite graph, called a protograph. An example is shown in Fig. 4.2 for the basematrix

$$\boldsymbol{B} = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 2 & 1 \end{pmatrix}.$$

Since the elements of the basematrix are not strictly binary, parallel edges are allowed and their numbers correspond to the respective entries in the basematrix. The Tanner graph of an LDPC code can be obtained by *lifting* a protograph: through copy-and-permute operations a number of copies of the protograph is generated and their edges are permuted such that connectivity constraints imposed by the basematrix are maintained [117]. A protograph-based LDPC (P-LDPC) code ensemble is defined by the set of all length-n_c LDPC codes whose Tanner graph is obtained by lifting $\boldsymbol{B}$ by a factor of Q such that $n_\mathrm{c} = Q \cdot n_\mathrm{p}$.

The copy-and-permute operation can also be interpreted as follows: Each entry b_{ij} in the basematrix is replaced by a sum of b_{ij} distinct permutation matrices of size $Q \times Q$. This approach allows a straightforward way to generate realizations of a protograph ensemble. If cyclicly shifted identity matrices are used as permutation matrices, the resulting parity-check matrices have a QC structure, see Sec. 2.3.5.

To distinguish the VNs and CNs in the protograph from those in the lifted parity-check matrix, we introduce the protograph VN set $\mathcal{V}_\mathrm{p} = \{\mathtt{V}_1, \mathtt{V}_2, \dots, \mathtt{V}_{n_\mathrm{p}}\}$ and CN set $\mathcal{C}_\mathrm{p} = \{\mathtt{C}_1, \mathtt{C}_2, \dots, \mathtt{C}_{m_\mathrm{p}}\}$. Every protograph VN (CN) identifies a VN (CN) type. We use the wording "a type $\mathtt{V}_k$ VN" with $k \in \{1, \dots, n_\mathrm{P}\}$ to identify a VN in the lifted Tanner graph of type $\mathtt{V}_k$. We also use the convention that a VN $\mathtt{v}_j$ in the Tanner graph is of type $\mathtt{V}_k$ if $\lceil j/Q \rceil = k$, i.e., consecutive blocks of Q VNs are associated to a given type. The same applies to CNs.

P-LDPC codes introduce structure for an LDPC code ensemble and facilitate the design of optimized LDPC codes. For instance, the notion of a VN type can be used to designate the association of a given BMD bit level to a VN in the protograph.

In the following, we review three well known protograph families, which will serve as reference designs.

Accumulate-Repeat-Jagged-4-Accumulate (AR4JA) AR4JA codes [119, Sec. 7.4] constitute a class of protographs with code rates $(k-1)/k, k \in \{2, 3, \dots\}$ and protograph

Protograph	$\omega^\star$
$\boldsymbol{B}_{\mathrm{AR4JA}-1/2}$	0.0144
$\boldsymbol{B}_{\mathrm{AR4JA}-2/3}$	0.0058
$\boldsymbol{B}_{\mathrm{AR4JA}-3/4}$	0.0032
$\boldsymbol{B}_{\mathrm{AR4JA}-4/5}$	0.0021
$\boldsymbol{B}_{\mathrm{AR4JA}-5/6}$	0.0015

Table 4.1.: Relative minimum distance of different AR4JA codes.

dimensions of $m_{\mathrm{p}} \times n_{\mathrm{p}} = 3 \times (k+3)$, i.e., there is one punctured VN. The basematrix of the rate 1/2 code is

$$\boldsymbol{B}_{\text{AR4JA-1/2}} = \begin{pmatrix} 0 & 0 & 1 & 0 & 2 \\ 1 & 1 & 0 & 1 & 3 \\ 1 & 2 & 0 & 2 & 1 \end{pmatrix} \tag{4.8}$$

where the last column is punctured. The first two columns correspond to information bits, the remaining ones are parity bits or punctured. Higher rate codes are obtained by adding a pair of columns to the beginning of (4.8), e.g., for a rate 2/3 code, we have

$$\boldsymbol{B}_{\text{AR4JA-2/3}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 2 \\ 3 & 1 & 1 & 1 & 0 & 1 & 3 \\ 1 & 3 & 1 & 2 & 0 & 2 & 1 \end{pmatrix}. \tag{4.9}$$

Realizations of this ensemble have a minimum distance that grows linearly with the blocklength. The relative minimum distance was computed with the algorithm of [120]. We show the corresponding $\omega^\star$ in Table 4.1.

Protograph Based Raptor Like (PBRL) PBRL codes were introduced in [121][3] as one approach for rate-compatible LDPC codes that can be derived from a common basematrix. Their common structure is

$$\boldsymbol{B} = \begin{pmatrix} \boldsymbol{B}^{\mathrm{HR}} & \boldsymbol{0} \\ \boldsymbol{B} & \boldsymbol{I} \end{pmatrix} \tag{4.10}$$

and can therefore be understood as a concatenation of an high rate code with basematrix $\boldsymbol{B}^{\mathrm{HR}}$ and an low-density generator matrix (LDGM) code with basematrix $\begin{pmatrix} \boldsymbol{B} & \boldsymbol{I} \end{pmatrix}$. This code class is inherently rate-adaptive by adding rows to the LDGM code and thereby decreasing its overall rate. For its construction, usually a good high rate (e.g., 2/3, 3/4) basematrix $\boldsymbol{B}^{\mathrm{HR}}$ is chosen and the rows of the LDGM part are selected in a greedy manner one after the other such that the decoding threshold of the respective code is minimized.

[3] An earlier patent showing similar ideas has priority date January 24, 2007, see `https://patents.google.com/patent/US8578249`.

PBRL codes have found their way into the 5G standard [122, 123] for enhanced mobile broadband (eMBB). This choice is motivated by the requirements for 5G, which involve the support of a large range of code rates, a fine granularity in blocklengths, support for hybrid automated repeat request (HARQ), high throughput decoders and a simple code description for a facilitated hardware implementation.

4.1.3. Sum-Product Algorithm for Single-Parity-Check Constraints

Finding the bit-MAP estimate can be interpreted as a special instance of an inference problem, where the marginal $P_{V|\boldsymbol{Y}}(v_j|\boldsymbol{y})$ should be calculated from $P_{\boldsymbol{V}|\boldsymbol{Y}}(\boldsymbol{v}|\boldsymbol{y})$. In the context of factor graphs we can identify the global function $f(v_1, v_2, \ldots, v_{n_\mathsf{c}})$ with $P_{\boldsymbol{V}|\boldsymbol{Y}}(\boldsymbol{v}|\boldsymbol{y})$. Assuming this factor graph is cycle-free, we can use the SPA to calculate the marginal with low complexity.

Classical LDPC codes[4] use SPC codes as constraints at the CNs. Consider a CN c_i with degree d_c, neighbors $\mathcal{N}(\mathsf{c}_i) = \{\mathsf{v}_{j_1}, \mathsf{v}_{j_2}, \ldots, \mathsf{v}_{j_{d_\mathsf{c}}}\}$ and the local function

$$f_i(v_{j_1}, v_{j_2}, \ldots, v_{j_{d_\mathsf{c}}}) = \mathbb{1}\left(v_{j_1} + v_{j_2} + \ldots + v_{j_{d_\mathsf{c}}} = 0\right). \tag{4.11}$$

We first assume the exchanged messages to be in the *probability domain.* In this particular case, the CN update (2.136) for $k \in \{1, \ldots, d_\mathsf{c}\}$ has a closed form expression [5]

$$\begin{aligned} m_{\mathsf{c}_i \to \mathsf{v}_{j_k}}(v) &= \sum_{\sim v_{j_k}} f_i(v_{j_1}, v_{j_2}, \ldots, v, \ldots, v_{j_{d_\mathsf{c}}}) \prod_{\mathsf{v} \in \mathcal{N}(\mathsf{c}_i) \setminus \{\mathsf{v}_{j_k}\}} m_{\mathsf{v} \to \mathsf{c}_i}(v) \\ &= \sum_{\sim v_{j_k}} \mathbb{1}\left(v_{j_1} + v_{j_2} + \ldots + v + \ldots + v_{j_{d_\mathsf{c}}} = 0\right) \prod_{\mathsf{v} \in \mathcal{N}(\mathsf{c}_i) \setminus \{\mathsf{v}_{j_k}\}} m_{\mathsf{v} \to \mathsf{c}_i}(v) \\ &= \frac{1}{2} + (-1)^v \frac{1}{2} \prod_{\mathsf{v} \in \mathcal{N}(\mathsf{c}_i) \setminus \{\mathsf{v}_{j_k}\}} (1 - 2 \cdot m_{\mathsf{v} \to \mathsf{c}_i}(1)) \end{aligned} \tag{4.12}$$

If the VN alphabet is binary, i.e., $v_j \in \mathbb{F}_2$, we prefer a representation of the messages which avoids a renormalization of probabilities and calculating products. Therefore, we reformulate the SPA in the log-domain and consider the messages

$$l_{\mathsf{v}_j \to \mathsf{c}_i} = \log\left(\frac{m_{\mathsf{v}_j \to \mathsf{c}_i}(0)}{m_{\mathsf{v}_j \to \mathsf{c}_i}(1)}\right) \qquad l_{\mathsf{c}_i \to \mathsf{v}_j} = \log\left(\frac{m_{\mathsf{c}_i \to \mathsf{v}_j}(0)}{m_{\mathsf{c}_i \to \mathsf{v}_j}(1)}\right). \tag{4.13}$$

Using the VN update rule of (2.135), we get

$$l_{\mathsf{v}_j \to \mathsf{c}_i} = \log\left(\frac{\prod_{\mathsf{c} \in \mathcal{N}(\mathsf{v}_j) \setminus \{\mathsf{c}_i\}} m_{\mathsf{c} \to \mathsf{v}_j}(0)}{\prod_{\mathsf{c} \in \mathcal{N}(\mathsf{v}_j) \setminus \{\mathsf{c}_i\}} m_{\mathsf{c} \to \mathsf{v}_j}(1)}\right) = \sum_{\mathsf{c} \in \mathcal{N}(\mathsf{v}_j) \setminus \{\mathsf{c}_i\}} \log\left(\frac{m_{\mathsf{c} \to \mathsf{v}_j}(0)}{m_{\mathsf{c} \to \mathsf{v}_j}(1)}\right) = \sum_{\mathsf{c} \in \mathcal{N}(\mathsf{v}_j) \setminus \{\mathsf{c}_i\}} l_{\mathsf{c} \to \mathsf{v}_j}. \tag{4.14}$$

[4] "Classical" in the sense of LDPC codes as introduced by Gallager [5]. *Generalized LDPC codes* [111] replace the SPC code by a general blockcode, e.g., a Hamming code.

Similarly, for the CN update rule of (2.136), we get

$$l_{\mathtt{c}_i \to \mathtt{v}_j} = \log\left(\frac{m_{\mathtt{c}_i \to \mathtt{v}_j}(0)}{m_{\mathtt{c}_i \to \mathtt{v}_j}(1)}\right) = \log\left(\frac{\frac{1}{2} + \frac{1}{2}\prod_{\mathtt{v} \in \mathcal{N}(\mathtt{c}_i)\setminus\{\mathtt{v}_j\}}(1 - 2m_{\mathtt{v}\to\mathtt{c}_i}(1))}{\frac{1}{2} - \frac{1}{2}\prod_{\mathtt{v} \in \mathcal{N}(\mathtt{c}_i)\setminus\{\mathtt{v}_j\}}(1 - 2m_{\mathtt{v}\to\mathtt{c}_i}(1))}\right). \tag{4.15}$$

To simplify the previous expression, we need two intermediate results. First, from (4.13), we have

$$m_{\mathtt{v}_j \to \mathtt{c}_i}(1) = \frac{1}{1 + \mathrm{e}^{l_{\mathtt{v}_j \to \mathtt{c}_i}}} \tag{4.16}$$

such that

$$1 - 2m_{\mathtt{v}_j \to \mathtt{c}_i}(1) = \frac{\mathrm{e}^{l_{\mathtt{v}_j \to \mathtt{c}_i}} - 1}{\mathrm{e}^{l_{\mathtt{v}_j \to \mathtt{c}_i}} + 1} = \tanh\left(\frac{l_{\mathtt{v}_j \to \mathtt{c}_i}}{2}\right).$$

Second, we note that the inverse of the $\tanh(\cdot)$, $\operatorname{atanh}(\cdot)$, can be expressed as $\operatorname{atanh}(x) = 0.5 \cdot \ln((1-x)/(1+x))$. Hence, (4.15) becomes

$$l_{\mathtt{c}_i \to \mathtt{v}_j} = 2 \operatorname{atanh}\left(\prod_{\mathtt{v} \in \mathcal{N}(\mathtt{c}_i)\setminus\{\mathtt{v}_j\}} \tanh\left(\frac{l_{\mathtt{v}\to\mathtt{c}_i}}{2}\right)\right). \tag{4.17}$$

As the exchanged messages in the current formulation of the SPA are *beliefs* (i.e., probability of a coded bit taking a certain value), the formulation is also known as *belief propagation*.

4.1.4. Decoding of LDPC Codes

Practical LDPC codes usually do not have a cycle-free factor graph[5] such that we can only hope for an approximated value of the a posteriori distribution. Therefore we apply the SPA iteratively and hope for convergence. In the following, we apply the previously developed log-domain version of the SPA for the decoding of binary LDPC codes.

In contrast to Sec. 4.1.3, the exchanged messages on the Tanner graph have superscripts (denoted by (ℓ)) that indicate the iteration number. The overall procedure is summarized in Algorithm 3. We initialize the algorithm by

$$l^{(0)}_{\mathtt{v}_j \to \mathtt{c}_i} = l_{\mathrm{dec},j} = \log\left(\frac{P_{V|Y}(v_j = 0|y)}{P_{V|Y}(v_j = 1|y)}\right). \tag{4.18}$$

[5] While a cycle-free graph is desirable from the perspective of getting exact results for the marginalization, this property is harmful from a minimum distance perspective [124] in the sense that any cycle-free Tanner graph with SPC CN constraints and $R_\mathrm{c} \geq 1/2$ has $d_\mathrm{min} \leq 2$. For general CN constraints see comments in [124, Sec. V-C].

Algorithm 3 Sum-Product Decoding of LDPC codes.

```
INPUT: $l^{(0)}_{\mathtt{v}_j \to \mathtt{c}_i} = l_{\mathrm{dec},j}, \forall j = 1, \dots, n_\mathrm{c}, \mathtt{c}_i \in \mathcal{N}(\mathtt{v}_j)$; max. iterations $\ell_\mathrm{max}$
$\ell = 1$
while $\ell \leq \ell_\mathrm{max}$ do
    // CN update
    for $i = 1, \dots, m_\mathrm{c}$ do
        for $\mathtt{v}_j \in \mathcal{N}(\mathtt{c}_i)$ do
            $l^{(\ell)}_{\mathtt{c}_i \to \mathtt{v}_j} = 2 \,\mathrm{atanh}\left( \prod_{\mathtt{v} \in \mathcal{N}(\mathtt{c}_i) \setminus \{\mathtt{v}_j\}} \tanh\left( \frac{l^{(\ell-1)}_{\mathtt{v} \to \mathtt{c}_i}}{2} \right) \right)$
        end for
    end for
    // VN update
    for $j = 1, \dots, n_\mathrm{c}$ do
        for $\mathtt{c}_i \in \mathcal{N}(\mathtt{v}_i)$ do
            $l^{(\ell)}_{\mathtt{v}_j \to \mathtt{c}_i} = \sum_{\mathtt{c} \in \mathcal{N}(\mathtt{v}_j) \setminus \{\mathtt{c}_i\}} l^{(\ell)}_{\mathtt{c} \to \mathtt{v}_j} + l_{\mathrm{dec},j}$
        end for
    end for
    $\ell = \ell + 1$
end while
// Final codeword bit estimate
for $j = 1, \dots, n_\mathrm{c}$ do
    $l_{\mathrm{app},j} = \sum_{\mathtt{c} \in \mathcal{N}(\mathtt{v}_j)} l^{(\ell_\mathrm{max})}_{\mathtt{c} \to \mathtt{v}_j} + l_{\mathrm{dec},j}$
    $\hat{v}_j = \frac{1}{2} - \frac{1}{2} \,\mathrm{sign}\left( l_{\mathrm{app},j} \right)$
end for
```

In a practical implementation of Algorithm 3, two important simplifications can be employed:

1. The CN update is calculated based on the *Jacobian logarithm*, i.e.,

$$\ln(\mathrm{e}^x + \mathrm{e}^y) = \max(x, y) + \log\left(1 + \mathrm{e}^{-|x+y|}\right).$$

   For a degree $d_\mathtt{c} = 3$ CN $\mathtt{c}_i$ with neighbors $\mathcal{N}(\mathtt{c}_i) = \{\mathtt{v}_{j_1}, \mathtt{v}_{j_2}, \mathtt{v}_{j_3}\}$, (4.17) becomes

$$\begin{aligned} l_{\mathtt{c}_i \to \mathtt{v}_{j_3}} &= f_\mathrm{CN}(l_{\mathtt{v}_{j_1} \to \mathtt{c}_i}, l_{\mathtt{v}_{j_2} \to \mathtt{c}_i}) \\ &= \mathrm{sign}(l_{\mathtt{v}_{j_1} \to \mathtt{c}_i}, l_{\mathtt{v}_{j_2} \to \mathtt{c}_i}) \cdot \min\left(\left| l_{\mathtt{v}_{j_1} \to \mathtt{c}_i} \right|, \left| l_{\mathtt{v}_{j_2} \to \mathtt{c}_i} \right| \right) \\ &\quad + \ln\left(\frac{1 + \mathrm{e}^{-\left| l_{\mathtt{v}_{j_1} \to \mathtt{c}_i} + l_{\mathtt{v}_{j_2} \to \mathtt{c}_i} \right|}}{1 + \mathrm{e}^{-\left| l_{\mathtt{v}_{j_1} \to \mathtt{c}_i} - l_{\mathtt{v}_{j_2} \to \mathtt{c}_i} \right|}} \right). \end{aligned} \tag{4.19}$$

   For CNs with degree $d_\mathrm{c} > 3$, we apply (4.19) recursively:

$$l_{\mathtt{c}_i \to \mathtt{v}_{j_{d_\mathrm{c}}}} = f_\mathrm{CN}(l_{\mathtt{v}_{j_1} \to \mathtt{c}_i}, f_\mathrm{CN}(l_{\mathtt{v}_{j_2} \to \mathtt{c}_i}, f_\mathrm{CN}(\dots, l_{\mathtt{v}_{j_{d_\mathrm{c}-1}} \to \mathtt{c}_i}))). \tag{4.20}$$

   Additionally, to reduce the complexity of the CN operation, the trellis representation of an SPC code [125] avoids the re-calculation of interim expressions. The implementation of the CN update has a significant influence on the error floor [126]. If not stated otherwise, all simulation results in this thesis use a non-saturating VN and CN implementation.

2. The naive formulation of the VN update in Algorithm 3 calculates the sum of incoming messages many times. Instead, a practical implementation calculates a temporary value first
$$l^{(\ell)}_{\mathrm{tmp},j} = \sum_{\mathtt{c}\in\mathcal{N}(\mathtt{v}_j)} l^{(\ell)}_{\mathtt{c}\to\mathtt{v}_j} + l_{\mathrm{dec},j}$$
and then subtracts the corresponding incoming message to get the right extrinsic value
$$l^{(\ell)}_{\mathtt{v}_j\to\mathtt{c}_i} = l^{(\ell)}_{\mathrm{tmp},j} - l^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}.$$

3. The message schedule, i.e., the order of how messages are computed and used for the calculation of other messages, can be modified to achieve faster convergence speeds. One prominent approach is the so called *layered* schedule [127], which commonly saves half the number of iterations compared to the *flooding* schedule of Algorithm 3. We will use the usual flooding schedule in all simulations.

4. Algorithm 3 can be extended by an early stopping criterion. For this, we calculate the HD estimate $\hat{\boldsymbol{v}}$ after each iteration and check whether the syndrome $\hat{\boldsymbol{v}}\boldsymbol{H}^{\mathrm{T}}$ is zero.

Variants of the sum-product algorithm are the *min-sum algorithm* [128], which neglects the second summand in (4.19) or the *offset min-sum algorithm*, which replaces the second summand by a constant value that needs to be optimized for each code and current iteration number. The *scaled offset min-sum algorithm* additionally scales the result of the minimum operator in (4.19) by a scalar.

4.2. Asymptotic Decoding Threshold Analysis

4.2.1. Density Evolution

As shown in [129] the asymptotic analysis of LDPC codes via density evolution (DE) is based on the *concentration theorem* stating that the decoding performance for an individual member of the ensemble will concentrate asymptotically around its average performance and that the average performance will concentrate around the performance of a cycle-free graph for a large blocklength. To facilitate analysis, the *symmetry condition* allows to constrain the DE to the *all-zero codeword* only.

Let $L_{\mathtt{v}\to\mathtt{c}}, L_{\mathtt{c}\to\mathtt{v}}, L_{\mathrm{dec}}$ be the RVs associated with the messages (4.14), (4.17) and (4.18), respectively. The symmetry condition requires that

$$p_{L_{\mathtt{v}\to\mathtt{c}}|V}(l|0) = p_{L_{\mathtt{v}\to\mathtt{c}}|V}(-l|1) \tag{4.21}$$

$$p_{L_{\mathsf{c}\to\mathsf{v}}|V}(l|0) = p_{L_{\mathsf{c}\to\mathsf{v}}|V}(-l|1) \tag{4.22}$$

$$p_{L_{\mathrm{dec}}|V}(l|0) = p_{L_{\mathrm{dec}}|V}(-l|1). \tag{4.23}$$

As analysis reveals, the requirements (4.21), (4.22) hold for the BP algorithm as presented in Sec. 4.1.4. Whether (4.23) is fulfilled or not depends on the channel $p_{Y|X}$, the modulation/demapping and binary labeling.

Example 9. For the AWGNC with BPSK signaling, i.e., $\mathcal{X} = \{-1, +1\}$, $\chi(+1) = 0, \chi(-1) = 1$, the channel law $p_{Y|X}$ is given by (2.93) and (4.18) becomes

$$l_{\mathrm{dec}} = \log\left(\frac{\mathrm{e}^{-(y-1)^2/(2\sigma^2)}}{\mathrm{e}^{-(y+1)^2/(2\sigma^2)}}\right) = \log\left(\mathrm{e}^{4y/(2\sigma^2)}\right) = \frac{2}{\sigma^2}y. \tag{4.24}$$

The above statement can be interpreted as a transformation of the RV Y. Conditioning on $X = -1$ (i.e., $V = 1$) and $X = 1$ (i.e., $V = 0$), we have

$$(L_{\mathrm{dec}}|\{V = 0\}) \sim \mathcal{N}(2/\sigma^2, 4/\sigma^2) \qquad (L_{\mathrm{dec}}|\{V = 1\}) \sim \mathcal{N}(-2/\sigma^2, 4/\sigma^2). \tag{4.25}$$

(4.23) is fulfilled for this channel and modulation setting. Note that the density $p_{L_{\mathrm{dec}}|V}(l|0)$ exhibits an interesting property, namely the *consistency condition* $f(x) = f(-x)\mathrm{e}^x$. A Gaussian distribution (2.93) is *consistent*, if its mean μ and variance σ^2 are related as $\sigma^2 = 2\mu$. A consistent Gaussian distribution is therefore characterized by a single parameter.

Density evolution tracks the densities $p_{L_{\mathsf{v}\to\mathsf{c}}|V}(l|0)$, $p_{L_{\mathsf{c}\to\mathsf{v}}|V}(l|0)$ and $p_{L_{\mathrm{app}}|V}(l|0)$ over the course of iterations. To simplify notation, we drop the conditioning on $V = 0$ and use the all-zero codeword assumption. We declare successful decoding when

$$\int_{-\infty}^{0} p_{L_{\mathrm{app}}^{(\ell)}}(l)\,\mathrm{d}l \to 0 \quad \text{for } \ell \to \infty. \tag{4.26}$$

We now consider channels that are characterized by a single parameter ξ (e.g., ε for the BEC, δ for the BSC, σ for the AWGNC). The set of channel parameters for which we observe convergence (4.26) defines the convergence region

$$\Upsilon = \left\{\xi : \int_{-\infty}^{0} p_{L_{\mathrm{app}}^{(\ell)}}(l;\xi)\,\mathrm{d}l \to 0 \quad \text{for } \ell \to \infty\right\}. \tag{4.27}$$

Here, $p_{L_{\mathrm{app}}^{(\ell)}}(l;\xi)$ denotes the PDF of the a posteriori information, when the BP iterations were initialized by l_{dec} having a channel law with parameter ξ. The *decoding threshold* ξ_{th}

is defined to be the supremum[6] of all values in Υ:

$$\xi_{\text{th}} = \sup_{\xi} \Upsilon. \tag{4.28}$$

Density Evolution for Unstructured, Regular Ensembles

We first analyze $(d_{\mathtt{v}}, d_{\mathtt{c}})$ unstructured, regular ensembles. Following (4.14), the outgoing message $l_{\mathtt{v}\to\mathtt{c}}$ of a VN update, is given as the summation of independent RVs. Hence, its PDF is given by a convolution of the incoming PDFs:

$$p^{(\ell)}_{L_{\mathtt{v}\to\mathtt{c}}}(l) = p_{L_{\text{dec}}}(l) * \left(p^{(\ell)}_{L_{\mathtt{c}\to\mathtt{v}}}(l)\right)^{*(d_{\mathtt{v}}-1)} \tag{4.29}$$

Similarly, we have

$$p^{(\ell)}_{L_{\text{app}}}(l) = p_{L_{\text{dec}}}(l) * \left(p^{(\ell)}_{L_{\mathtt{c}\to\mathtt{v}}}(l)\right)^{*d_{\mathtt{v}}}. \tag{4.30}$$

The CN update is more involved. The update rule (4.17) is a transformation of the RV $L_{\mathtt{v}\to\mathtt{c}}$, see Sec. 2.2.5. Unfortunately, no closed form expression can be given such that we simply denote this transformation by the function $f_{\text{CN}}(\cdot,\cdot)$, which takes the PDF of the incoming message $p^{(\ell)}_{L_{\mathtt{v}\to\mathtt{c}}}$ and the CN degree $d_{\mathtt{c}}$ as arguments:

$$p^{(\ell)}_{L_{\mathtt{c}\to\mathtt{v}}}(l) = f_{\text{CN}}(p^{(\ell)}_{L_{\mathtt{v}\to\mathtt{c}}}(l), d_{\mathtt{c}}) \tag{4.31}$$

We explain an approximate method in Sec. 4.2.2 to circumvent this problem. Alternatively, Chung developed an alternative in his PhD thesis [130] that approximates $p_{L_{\mathtt{c}\to\mathtt{v}}}(l)$ by a Gaussian distribution, whose mean and variance are derived from $L^{(\ell)}_{\mathtt{v}\to\mathtt{c}}$ and the CN degree $d_{\mathtt{c}}$.

Example 10. We derive the DE equations for a $(d_{\mathtt{v}}, d_{\mathtt{c}})$ regular LDPC code for the BEC with erasure probability ε. We have

$$l_{\text{dec}} = \log\left(\frac{P_{Y|X}(y|0)}{P_{Y|X}(y|1)}\right) = \begin{cases} \infty, & y = 0 \\ 0, & y = E \\ -\infty, & y = 1. \end{cases} \tag{4.32}$$

The density $p_{L_{\text{dec}}}(l)$ degrades to a PMF with two mass points, one being at 0 with probability ε and the other at $+\infty$ with probability $1-\varepsilon$. Therefore, DE for the BEC is characterized by tracking the evolution of the a posteriori erasure probability $\varepsilon^{(\ell)}_{\text{app}}$. We obtain:

[6] This assumes that the degradation of the channel becomes more significant with a larger value of ξ. This is the case for the BSC, BEC and AWGNC. If a transformation of the channel parameter is considered, e.g., the SNR in case of the AWGNC, this definition must be adjusted.

$$\varepsilon^{(\ell)}_{\mathtt{v}\to\mathtt{c}} = \varepsilon \cdot \left(\varepsilon^{(\ell)}_{\mathtt{c}\to\mathtt{v}}\right)^{d_{\mathtt{v}}-1} \tag{4.33}$$

$$\varepsilon^{(\ell)}_{\mathtt{c}\to\mathtt{v}} = 1 - \left(1 - \varepsilon^{(\ell)}_{\mathtt{v}\to\mathtt{c}}\right)^{d_{\mathtt{c}}-1} \tag{4.34}$$

$$\varepsilon^{(\ell)}_{\mathrm{app}} = \varepsilon \cdot \left(\varepsilon^{(\ell)}_{\mathtt{c}\to\mathtt{v}}\right)^{d_{\mathtt{v}}}. \tag{4.35}$$

Density Evolution for Protograph Based Ensembles

For protograph based ensembles, we have to take each VN ($\mathtt{V}_j, j = 1, \dots, n_{\mathrm{p}}$) and CN ($\mathtt{C}_i, i = 1, \dots, m_{\mathrm{p}}$) type into account. Let L_{dec_j} represent the RV describing the decoder soft information at the j-th VN $\mathtt{V}_j$. Correspondingly, we get

$$p^{(\ell)}_{L_{\mathtt{V}_j\to\mathtt{C}_i}}(l) = p_{L_{\mathrm{dec},j}}(l) \mathop{\circledast}_{\mathtt{C}_{i'}\in\mathcal{N}(\mathtt{V}_j)} p^{(\ell)}_{L_{\mathtt{C}_{i'}\to\mathtt{V}_j}}(l)^{*(b_{ij}-\mathbb{1}(i'=i))} \tag{4.36}$$

$$p^{(\ell)}_{L_{\mathrm{app},j}}(l) = p_{L_{\mathrm{dec},j}}(l) \mathop{\circledast}_{\mathtt{C}_{i}\in\mathcal{N}(\mathtt{V}_j)} p^{(\ell)}_{L_{\mathtt{C}_{i}\to\mathtt{V}_j}}(l)^{*(b_{ij})}. \tag{4.37}$$

For the CN update, assuming that CN $\mathtt{C}_i$ has neighbors $\mathcal{N}(\mathtt{C}_i) = \{\mathtt{V}_{j_1}, \mathtt{V}_{j_2}, \dots, \mathtt{V}_{j_{d_{\mathtt{c}}}}\}$ we have

$$p^{(\ell)}_{L_{\mathtt{C}_i\to\mathtt{V}_j}}(l) = f_{\mathrm{CN}}\left(\underbrace{p^{(\ell)}_{L_{\mathtt{V}_{j_1}\to\mathtt{C}_i}}(l), \dots, p^{(\ell)}_{L_{\mathtt{V}_{j_1}\to\mathtt{C}_i}}(l)}_{b_{ij_1}-\mathbb{1}(j=j_1)\text{ times}}, \dots, \underbrace{p^{(\ell)}_{L_{\mathtt{V}_{j_{d_{\mathtt{c}}}}\to\mathtt{C}_i}}(l), \dots, p^{(\ell)}_{L_{\mathtt{V}_{j_{d_{\mathtt{c}}}}\to\mathtt{C}_i}}(l)}_{b_{ij_{d_{\mathtt{c}}}}-\mathbb{1}(j=j_{d_{\mathtt{c}}})\text{ times}}\right) \tag{4.38}$$

where the function $f_{\mathrm{CN}}(\cdot)$ represents the CN transformation of the incoming PDFs. Again, we refer to Sec. 4.2.2 for the discussion of a practical implementation of this transformation.

We extend the notion of a convergence region for protographs. Let $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_{n_{\mathrm{p}}})$ denote the parameter vector which defines the (potentially) n_{p} different channels associated with each protograph VN type. We declare convergence for the protograph DE and parameter vector $\boldsymbol{\xi}$ if

$$\int_{-\infty}^{0} p_{L^{(\ell)}_{\mathrm{app},j}}(l;\xi_j)\,\mathrm{d}l \to 0 \quad \text{for } \ell \to \infty, \quad \forall j = 1, \dots, n_{\mathrm{p}}. \tag{4.39}$$

Correspondingly, we have the convergence region

$$\Upsilon_{\mathrm{p}} = \left\{\boldsymbol{\xi} : \int_{-\infty}^{0} p_{L^{(\ell)}_{\mathrm{app},j}}(l;\xi_j)\,\mathrm{d}l \to 0 \quad \text{for } \ell \to \infty, \quad \forall j = 1, \dots, n_{\mathrm{p}}\right\}. \tag{4.40}$$

Example 11. As before, we instantiate (4.36), (4.37) and (4.38) for the BEC. Let ε_j denote the erasure probability at the j-th VN $\mathtt{V}_j$, i.e., the j-th VN is connected to a BEC with erasure probability ε_j. We obtain

$$\varepsilon_{\mathtt{V}_j \to \mathtt{C}_i} = \varepsilon_j \cdot \prod_{\mathtt{C}_{i'} \in \mathcal{N}(\mathtt{V}_j)} \left(\varepsilon^{(\ell)}_{\mathtt{C}_{i'} \to \mathtt{V}_j}\right)^{(b_{i'j} - \mathbb{1}(i=i'))} \tag{4.41}$$

$$\varepsilon_{\mathtt{C}_i \to \mathtt{V}_j} = 1 - \prod_{\mathtt{V}_{j'} \in \mathcal{N}(\mathtt{C}_j)} \left(1 - \varepsilon^{(\ell)}_{\mathtt{V}_{j'} \to \mathtt{C}_i}\right)^{(b_{ij'} - \mathbb{1}(j=j'))} \tag{4.42}$$

$$\varepsilon_{\mathrm{app},j} = \varepsilon_j \cdot \prod_{\mathtt{C}_i \in \mathcal{N}(\mathtt{V}_j)} \left(\varepsilon^{(\ell)}_{\mathtt{C}_i \to \mathtt{V}_j}\right)^{b_{ij}} . \tag{4.43}$$

4.2.2. Discretized Density Evolution

Discretized density evolution (DDE) approximates the real DE expressions (4.29) – (4.31) by discretizing the PDF of the involved BP messages. It was used to design capacity approaching LDPC codes in [112] and quantizes the decoder soft-information (4.18) with a b bit ($b \in \mathbb{N}$) quantization function, which first clips its input to $B \in \mathbb{R}^+$ or $-B$ via

$$\mathsf{clip}(l) = \begin{cases} +B, & l \geq +B \\ l, & -B < l < +B \\ -B, & l \leq -B \end{cases} \tag{4.44}$$

and maps the result to $q = 2^b - 1$ quantization levels. We define the quantization function as $\mathsf{Q}(\cdot) : \mathbb{R} \to \mathcal{Q}$, where $\mathcal{Q} = \{-(q-1)/2, \ldots, 0, \ldots, (q-1)/2\}$, $\Delta = (2B)/(q-1)$, and

$$\mathsf{Q}(l) = \begin{cases} \left\lfloor \mathsf{clip}(l)/\Delta + \frac{1}{2} \right\rfloor, & l > \frac{\Delta}{2} \\ \left\lceil \mathsf{clip}(l)/\Delta - \frac{1}{2} \right\rceil, & l < -\frac{\Delta}{2} \\ 0, & \text{otherwise.} \end{cases} \tag{4.45}$$

We use this type of quantization to represent $l = 0$ without quantization error. This is important for punctured VNs. In the following, we describe DDE for protographs.

Using channel adapters [131] the PDFs of the m BMD bit channels are symmetrized such that the all-zero codeword assumption can be used. We quantize the symmetrized RV $L_{\mathrm{dec},j}$ of the decoder soft information (4.18) at the j-th VN by (4.45) and represent the PMF of the discrete RV by the vector $\boldsymbol{l}_{\mathrm{dec},j}$ of length q, where the entries $l_{\mathrm{dec},jk}, k \in \mathcal{Q}$ correspond to

$$l_{\mathrm{dec},jk} = \int_{k\Delta}^{(k+1)\Delta} p_{L_{\mathrm{dec},j}}(l)\, \mathrm{d}l. \tag{4.46}$$

We define the DDE VN update rule as

$$\boldsymbol{l}_{\mathtt{V}_j\to\mathtt{C}_i} = \mathsf{T}\left(\boldsymbol{l}_{\mathrm{dec},j} * \underset{\mathtt{C}_{i'}\in\mathcal{N}(\mathtt{V}_j)}{\circledast} \boldsymbol{l}_{\mathtt{C}_{i'}\to\mathtt{V}_j}^{*\left(b_{i'j}-\mathbb{1}(i=i')\right)}\right). \tag{4.47}$$

The truncation operator $\mathsf{T}(\cdot)$ shrinks the dimensions of the vectors resulting from the convolution. For a length $2q-1$ vector $\boldsymbol{c}=(c_{-(q-1)},\dots,c_{q-1})$, we have

$$\mathsf{T}(\boldsymbol{c}) = \left(\sum_{k=-(q-1)}^{-(q-1)/2} c_k, c_{-(q-1)/2+1},\dots,c_{(q-1)/2-1}, \sum_{k=(q-1)/2}^{(q-1)} c_k\right). \tag{4.48}$$

For punctured VNs, we have $l_{\mathrm{dec},jk}=0, \forall k\in\mathcal{Q}\setminus\{0\}$ and $l_{\mathrm{dec},j0}=1$.

For the CN operation, we first consider a degree three CN with two incoming messages $\boldsymbol{a}$ and $\boldsymbol{b}$. The outgoing message $\boldsymbol{c}$ is given as $\boldsymbol{c}=\mathsf{R}(\boldsymbol{a},\boldsymbol{b})$, where each entry c_k of $\boldsymbol{c}$ is given by

$$c_k = \sum_{(k_1,k_2)\in\mathcal{Q}\times\mathcal{Q}:\ \mathsf{LUT}(k_1,k_2)=k} a_{k_1}b_{k_2}. \tag{4.49}$$

The two-dimensional data structure $\mathsf{LUT}(\cdot,\cdot):\mathcal{Q}\times\mathcal{Q}\to\mathcal{Q}$ is a look-up table (LUT) of the quantized CN operation for the SPA (4.17), i.e., for $a,b\in\mathcal{Q}$, we have

$$\mathsf{LUT}(a,b) = \mathsf{Q}(2\,\mathrm{atanh}(\tanh(a/2)\cdot\tanh(b/2)). \tag{4.50}$$

The CN update rule can be stated as the nested application of the pairwise operator $\mathsf{R}(\cdot,\cdot)$

$$\boldsymbol{l}_{\mathtt{C}_i\to\mathtt{V}_j} = \mathsf{R}\left(\boldsymbol{l}_{\mathtt{V}_{j'}\to\mathtt{C}_i}, \mathsf{R}(\dots,\dots)\right) \tag{4.51}$$

to the set of messages

$$\{\quad \underbrace{\boldsymbol{l}_{\mathtt{V}'_j\to\mathtt{C}_i}}_{(b_{ij'}-\mathbb{1}(j'=j))\text{ times}} \quad\},\quad \mathtt{V}_{j'}\in\mathcal{N}(\mathtt{C}_i). \tag{4.52}$$

The a posteriori message is

$$\boldsymbol{l}_{\mathrm{app},j} = \mathsf{T}\left(\boldsymbol{l}_{\mathrm{dec},j} * \underset{\mathtt{C}_{i'}\in\mathcal{N}(\mathtt{V}_j)}{\circledast} \boldsymbol{l}_{\mathtt{C}_{i'}\to\mathtt{V}_j}^{*b_{i'j}}\right). \tag{4.53}$$

We summarize DDE in Algorithm 4.

Algorithm 4 Algorithmic description of DDE

```
INPUT: $\boldsymbol{l}^{(0)}_{\mathtt{V}_j \to \mathtt{C}_i} = \boldsymbol{l}_{\mathrm{dec},j}$, $\forall j = 1, \ldots, n_{\mathrm{p}}, \mathtt{C}_i \in \mathcal{N}(\mathtt{V}_j)$, MIN_BER, max. iterations $\ell^{\mathrm{DDE}}_{\max}$
converged ← 0, $\ell \leftarrow 1$
while $\ell \leq \ell^{\mathrm{DDE}}_{\max}$ do
    for $i = 1, \ldots, m_{\mathrm{p}}$ do
        for $j = 1, \ldots, n_{\mathrm{p}}$ do
            if $b_{ij} \neq 0$ then
                Calculate $\boldsymbol{l}^{(\ell)}_{\mathtt{C}_i \to \mathtt{V}_j}$ according to (4.51).
            end if
        end for
    end for
    for $j = 1, \ldots, n_{\mathrm{p}}$ do
        for $i = 1, \ldots, m_{\mathrm{p}}$ do
            if $b_{ij} \neq 0$ then
                Calculate $\boldsymbol{l}^{(\ell)}_{\mathtt{V}_j \to \mathtt{C}_i}$ according to (4.47).
            end if
        end for
    end for
    for $j = 1, \ldots, n_{\mathrm{p}}$ do
        Calculate $\boldsymbol{l}^{(\ell)}_{\mathrm{app},j}$ according to (4.53)
    end for
    if $\sum_{k=1}^{(q-1)/2} l_{\mathrm{app},jk} \leq$ MIN_BER, $\forall j = 1, \ldots, n_{\mathrm{p}}$ then
        converged ← 1
        Break.
    end if
    $\ell \leftarrow \ell + 1$
end while
```

4.2.3. EXIT and P-EXIT Analysis

The previously discussed approaches for calculating the decoding thresholds of LDPC codes involve a significant computational burden, as the VN and CN operations have to deal with PDFs or, in the discretized case, with PMFs. As any optimization of LDPC codes for a particular setting involves evaluating the decoding threshold of different ensembles many times, we need to come up with an efficient method. One approach to accomplish this is by introducing a parameterized model of the decoding process so that a single parameter (instead of an entire PDF or PMF) is tracked over the course of iterations. In [132], it was shown empirically that the MI is one choice that yields accurate results. This gave rise to extrinsic information transfer (EXIT) charts.

EXIT Analysis for Unstructured, Regular Ensembles

We first consider the case of unstructured, $(d_{\mathtt{v}}, d_{\mathtt{c}})$ regular ensembles again. Instead of the PDFs (4.21)–(4.23), we now consider the MI expressions

$$I_{\mathtt{v} \to \mathtt{c}} = \mathrm{I}(V; L_{\mathtt{v} \to \mathtt{c}}) \tag{4.54}$$

$$I_{\mathtt{c} \to \mathtt{v}} = \mathrm{I}(V; L_{\mathtt{c} \to \mathtt{v}}) \tag{4.55}$$

$$I_{\text{dec}} = \mathrm{I}(V; L_{\text{dec}}). \tag{4.56}$$

EXIT analysis requires to find functions $f_{\text{EXIT}-\text{CN}}$ and $f_{\text{EXIT}-\text{VN}}$ that return the extrinsic MI $I_{\mathtt{v}\to\mathtt{c}}$ ($I_{\mathtt{c}\to\mathtt{v}}$) given the respective input MI values, i.e.,

$$I_{\mathtt{v}\to\mathtt{c}} = f_{\text{EXIT}-\text{VN}}(I_{\mathtt{c}\to\mathtt{v}}, I_{\text{dec}}) \tag{4.57}$$

$$I_{\mathtt{c}\to\mathtt{v}} = f_{\text{EXIT}-\text{CN}}(I_{\mathtt{v}\to\mathtt{c}}). \tag{4.58}$$

In many cases, no closed form expressions can be given, e.g., for turbo codes when the parallel concatenated convolutional codes are decoded by component BCJR decoders. Here we need to perform MC simulations first to come up with interpolations for (4.57) and (4.58). For the BEC and component codes based on SPC codes and repetition codes (RCs), the derivation is straightforward and is even equivalent to DE. The reason is that the extrinsic channels are also BECs so that one parameter fully characterizes the setup.

Example 12. Starting from Example 10 and using (2.87), we find the EXIT formulas for the BEC:

$$I^{(\ell)}_{\mathtt{v}\to\mathtt{c}} = f_{\text{EXIT}-\text{VN}}(I_{\mathtt{c}\to\mathtt{v}}, I_{\text{dec}}) = I_{\text{dec}} \cdot \left(1 - I^{(\ell)}_{\mathtt{c}\to\mathtt{v}}\right)^{d_{\mathtt{v}}-1} \tag{4.59}$$

$$I^{(\ell)}_{\mathtt{c}\to\mathtt{v}} = f_{\text{EXIT}-\text{CN}}(I_{\mathtt{v}\to\mathtt{c}}) = \left(I^{(\ell)}_{\mathtt{v}\to\mathtt{c}}\right)^{d_{\mathtt{c}}-1} \tag{4.60}$$

$$I^{(\ell)}_{\text{app}} = \mathrm{I}_{\text{dec}} \cdot \left(1 - I^{(\ell)}_{\mathtt{c}\to\mathtt{v}}\right)^{d_{\mathtt{v}}}. \tag{4.61}$$

Another important channel is the binary-input additive white Gaussian noise channel (biAWGNC). Unfortunately, finding the closed form expressions for (4.57) and (4.58) is not possible. Instead, we aim for approximations based on the following observations:

▷ In [129], the authors showed that the SPA preserves the symmetry of the message distributions and that the consistency condition holds (see Example 9). Now consider a degree $d_{\mathtt{v}}$ VN with neighbors $\mathcal{N}(\mathtt{v}_j) = \{\mathtt{c}_{i_1}, \ldots, \mathtt{c}_{i_{d_{\mathtt{v}}}}\}$. If we assume that the incoming and iid. messages $L_{\mathtt{c}_{i_1}}, \ldots, L_{\mathtt{c}_{i_{d_{\mathtt{v}}}}}$ are Gaussian distributed with variance $\sigma^2_{L_{\mathtt{c}\to\mathtt{v}}}$, then $L_{\mathtt{v}\to\mathtt{c}}$ is also Gaussian distributed with variance

$$\mathrm{Var}\left[L_{\mathtt{v}\to\mathtt{c}_i}\right] = \mathrm{Var}\left[\sum_{\mathtt{c}_{i'}\in\mathcal{N}(\mathtt{v})\setminus\{\mathtt{c}_i\}} L_{\mathtt{c}_{i'}\to\mathtt{v}} + L_{\text{dec}}\right] = (d_{\mathtt{v}}-1)\sigma^2_{L_{\mathtt{c}\to\mathtt{v}}} + \sigma^2_{L_{\text{dec}}} \tag{4.62}$$

and its mean follows by the consistency condition as $2\,\mathrm{Var}\left[L_{\mathtt{v}\to\mathtt{c}_i}\right]$. For convenience, we introduce the function

$$\mathrm{I}(X; L) = J(\sigma_L) = 1 - \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma_L^2}} \mathrm{e}^{-\frac{\left(z-\sigma_L^2/2\right)^2}{2\sigma_L^2}} \log_2\left(1+\mathrm{e}^{-z}\right) \mathrm{d}z \tag{4.63}$$

which denotes the MI of a biAWGNC that fulfills the consistency condition, i.e., of

$L = X + N$ where $X \in \{-\sigma_L^2/2, +\sigma_L^2/2\}$ and $N \sim \mathcal{N}(0, \sigma_L^2)$. Numerical approximations for (4.63) were given in [133] and [134]. The latter is more accurate, but also more complicated to compute efficiently. The VN EXIT function is therefore given by

$$I_{\mathtt{v}\to\mathtt{c}} = f_{\mathrm{EXIT-VN}}(I_{\mathtt{c}\to\mathtt{v}}, I_{\mathrm{dec}}) = J\left(\sqrt{(d_{\mathtt{v}}-1)\cdot J^{-1}(I_{\mathtt{c}\to\mathtt{v}})^2 + J^{-1}(I_{\mathrm{dec}})^2}\right). \tag{4.64}$$

▷ For the CNs, we exploit the duality property of EXIT functions on the BEC. Let $f(x)$ be the EXIT function for a given code on the BEC. It is shown in [135] that the EXIT function for the dual code is given by $1 - f(1-x)$. As RCs and SPC codes are dual to each other and we have the EXIT function for the VNs (4.64), we can hope that this property also yields accurate results for the biAWGNC. The duality of the CN and VN operations has been noted before [130].

Example 13. Using the previous derivations, we can now formulate the EXIT analysis for a $(d_{\mathtt{v}}, d_{\mathtt{c}})$ regular LDPC code ensemble for transmission over the biAWGNC with parameter σ^2. We have

$$I_{\mathtt{v}\to\mathtt{c}} = f_{\mathrm{EXIT-VN}}(I_{\mathtt{c}\to\mathtt{v}}, I_{\mathrm{dec}}) = J\left(\sqrt{(d_{\mathtt{v}}-1)\cdot J^{-1}(I_{\mathtt{c}\to\mathtt{v}})^2 + \sigma_L^2}\right) \tag{4.65}$$

$$I_{\mathtt{c}\to\mathtt{v}} = f_{\mathrm{EXIT-CN}}(I_{\mathtt{v}\to\mathtt{c}}) = 1 - J\left(\sqrt{(d_{\mathtt{c}}-1)\cdot J^{-1}(1 - I_{\mathtt{v}\to\mathtt{c}})}\right) \tag{4.66}$$

$$I_{\mathrm{app}} = J\left(\sqrt{d_{\mathtt{v}}\cdot J^{-1}(I_{\mathtt{c}\to\mathtt{v}})^2 + \sigma_L^2}\right) \tag{4.67}$$

where $\sigma_L^2 = 4/\sigma^2$ according to (4.25).

P-EXIT Analysis for Protograph Based Ensembles

The previous EXIT analysis was adopted for protograph based ensembles in [136] and will be referred to as P-EXIT in the following. It will serve as our primary tool in the subsequent code design approaches. The main difference compared to the previous description is that we need to track the MI of each edge individually.

In the following, we denote by $\mathrm{I}^{(\ell)}_{\mathtt{v}_j\to\mathtt{c}_i}$ the MI between the message sent at iteration ℓ by the j-th VN to the i-th CN and the corresponding codeword bit. Similarly, $\mathrm{I}^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}$ denotes the MI between the message sent at iteration ℓ by the i-th CN to the j-th VN and the corresponding codeword bit. We further express the MI between the j-th channel output and input as $\mathrm{I}_{\mathrm{dec},j}$. The evolution of the MI can be tracked by applying the recursion

$$\mathrm{I}^{(\ell)}_{\mathtt{v}_j\to\mathtt{c}_i} = f_{\text{P-EXIT-}\mathtt{v}_j-\mathtt{c}_i}\left(\boldsymbol{I}^{(\ell)}_{\mathtt{c}\to\mathtt{v}_j}, \mathrm{I}_{\mathrm{dec},j}\right) \qquad \mathrm{I}^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j} = f_{\text{P-EXIT-}\mathtt{c}_i-\mathtt{v}_j}\left(\boldsymbol{I}^{(\ell-1)}_{\mathtt{v}\to\mathtt{c}_i}\right) \tag{4.68}$$

with

$$\boldsymbol{I}^{(\ell)}_{\mathtt{C}\to\mathtt{V}_j} = \left(\mathrm{I}^{(\ell)}_{\mathtt{C}_1\to\mathtt{V}_j}, \mathrm{I}^{(\ell)}_{\mathtt{C}_2\to\mathtt{V}_j}, \dots, \mathrm{I}^{(\ell)}_{\mathtt{C}_{m_\mathrm{p}}\to\mathtt{V}_j}\right) \qquad \boldsymbol{I}^{(\ell)}_{\mathtt{V}\to\mathtt{C}_i} = \left(\mathrm{I}^{(\ell)}_{\mathtt{V}_1\to\mathtt{C}_i}, \mathrm{I}^{(\ell)}_{\mathtt{V}_2\to\mathtt{C}_i}, \dots, \mathrm{I}^{(\ell)}_{\mathtt{V}_{n_\mathrm{p}}\to\mathtt{C}_i}\right) \tag{4.69}$$

where we set $\mathrm{I}^{(\ell)}_{\mathtt{V}_j\to\mathtt{C}_i} = \mathrm{I}^{(\ell)}_{\mathtt{C}_i\to\mathtt{V}_j} = 0$ if $\mathtt{C}_i \notin \mathcal{N}(\mathtt{V}_j)$. In (4.68) we introduced the VN and CN P-EXIT functions $f_{\text{P-EXIT-}\mathtt{V}_j-\mathtt{C}_i}$ and $f_{\text{P-EXIT-}\mathtt{C}_i-\mathtt{V}_j}$, which depend on the underlying channel model and usually can not be given in closed form. We moreover denote by $\mathrm{I}^{(\ell)}_{\mathrm{app},j}$ the MI between the logarithmic a posteriori ratio message computed at the j-th VN in the ℓ-th iteration, and the corresponding codeword bit. Note that $\mathrm{I}^{(\ell)}_{\mathrm{app},j}$ is a function of $\boldsymbol{I}^{(\ell)}_{\mathtt{C}\to\mathtt{V}_i}$ and $\mathrm{I}_{\mathrm{dec},j}$. We define the protograph convergence region $\Upsilon^{\mathrm{EXIT}}_{\mathrm{p}}$ as the set of channel MI vectors $\boldsymbol{I}_{\mathrm{ch}} = \left(\mathrm{I}_{\mathrm{dec},1}, \mathrm{I}_{\mathrm{dec},2}, \dots, \mathrm{I}_{\mathrm{dec},n_\mathrm{p}}\right)$ for which $\mathrm{I}^{(\ell)}_{\mathrm{app},j}$ converges to 1, i.e.,

$$\Upsilon^{\mathrm{EXIT}}_{\mathrm{p}} = \left\{\boldsymbol{I}_{\mathrm{ch}} \,\middle|\, \mathrm{I}^{(\ell)}_{\mathrm{app},j} \to 1,\ \forall j = 1,\dots,n_\mathrm{p},\ \ell\to\infty\right\}.$$

Example 14. Adopting Example 10 for protograph ensembles, we obtain the following P-EXIT equations when the j-th protograph VN is connected to a BEC with erasure probability ε_j, $j = 1,\dots,n_\mathrm{p}$:

$$\mathrm{I}^{(\ell)}_{\mathtt{V}_j\to\mathtt{C}_i} = 1 - \varepsilon_j \prod_{\mathtt{C}_{i'}\in\mathcal{N}(\mathtt{V}_j)} \left(1 - \mathrm{I}^{(\ell-1)}_{\mathtt{C}_{i'}\to\mathtt{V}_j}\right)^{b_{i'j} - \mathbb{1}(i'=i)} \tag{4.70}$$

$$\mathrm{I}^{(\ell)}_{\mathtt{C}_i\to\mathtt{V}_j} = \prod_{\mathtt{V}_{j'}\in\mathcal{N}(\mathtt{C}_i)} \left(\mathrm{I}^{(\ell)}_{\mathtt{V}_{j'}\to\mathtt{C}_i}\right)^{b_{ij'} - \mathbb{1}(j=j')} \tag{4.71}$$

$$\mathrm{I}^{(\ell)}_{\mathrm{app},j} = 1 - \varepsilon_j \prod_{\mathtt{C}_i\in\mathcal{N}(\mathtt{V}_j)} \left(1 - \mathrm{I}^{(\ell-1)}_{\mathtt{C}_i\to\mathtt{V}_j}\right)^{b_{ij}}. \tag{4.72}$$

Example 15. Adopting Example 13 for protograph ensembles, we obtain the following P-EXIT equations when the j-th protograph VN is connected to a biAWGNC with variance σ_j^2, $j = 1,\dots,n_\mathrm{p}$:

$$\mathrm{I}^{(\ell)}_{\mathtt{V}_j\to\mathtt{C}_i} = J\left(\sqrt{\sum_{\mathtt{C}_{i'}\in\mathcal{N}(\mathtt{V}_j)} \left(b_{i'j} - \mathbb{1}\left(i=i'\right)\right)\cdot J^{-1}\left(\mathrm{I}^{(\ell-1)}_{\mathtt{C}_{i'}\to\mathtt{V}_j}\right)^2 + \sigma_j^2/4}\right) \tag{4.73}$$

$$\mathrm{I}^{(\ell)}_{\mathtt{C}_i\to\mathtt{V}_j} = 1 - J\left(\sqrt{\sum_{\mathtt{V}_{j'}\in\mathcal{N}(\mathtt{C}_i)} \left(b_{ij'} - \mathbb{1}\left(j'\neq j\right)\right)\cdot J^{-1}\left(1 - \mathrm{I}^{(\ell)}_{\mathtt{V}_i\to\mathtt{C}_{j'}}\right)^2}\right) \tag{4.74}$$

$$\mathrm{I}_{\mathrm{app},j}^{(\ell)} = J\left(\sqrt{\sum_{\mathtt{C}_i \in \mathcal{N}(\mathtt{V}_j)} b_{ij} \cdot J^{-1}\left(\mathrm{I}_{\mathtt{C}_i \to \mathtt{V}_j}^{(\ell)}\right)^2 + \sigma_j^2/4}\right). \tag{4.75}$$

4.3. Optimizing LDPC Codes for Higher-Order Modulation

4.3.1. Bit Level Uncertainties for Bit Metric Decoding

To operate LDPC codes with higher order modulation, usually BMD is employed, where the demapper calculates a bit-wise soft information for each of the $m = \log_2(M)$ bits indexing a constellation symbol[7]. As numerical investigations show, the associated bit channels have a different "reliability". In the following, this reliability is expressed in terms of the bit uncertainty $\mathrm{H}(B_k|Y), k = 1, \ldots, m$[8]. From (3.11), we recall that the sum of all m bit uncertainties determines the mismatched uncertainty $\mathrm{U}(q_{\mathrm{BMD}})$ for BMD and layered PS. We depict the bit uncertainties for uniform and shaped signaling in Fig. 4.3 for $\{4, 8, 16\}$-ASK and a BRGC labeling.

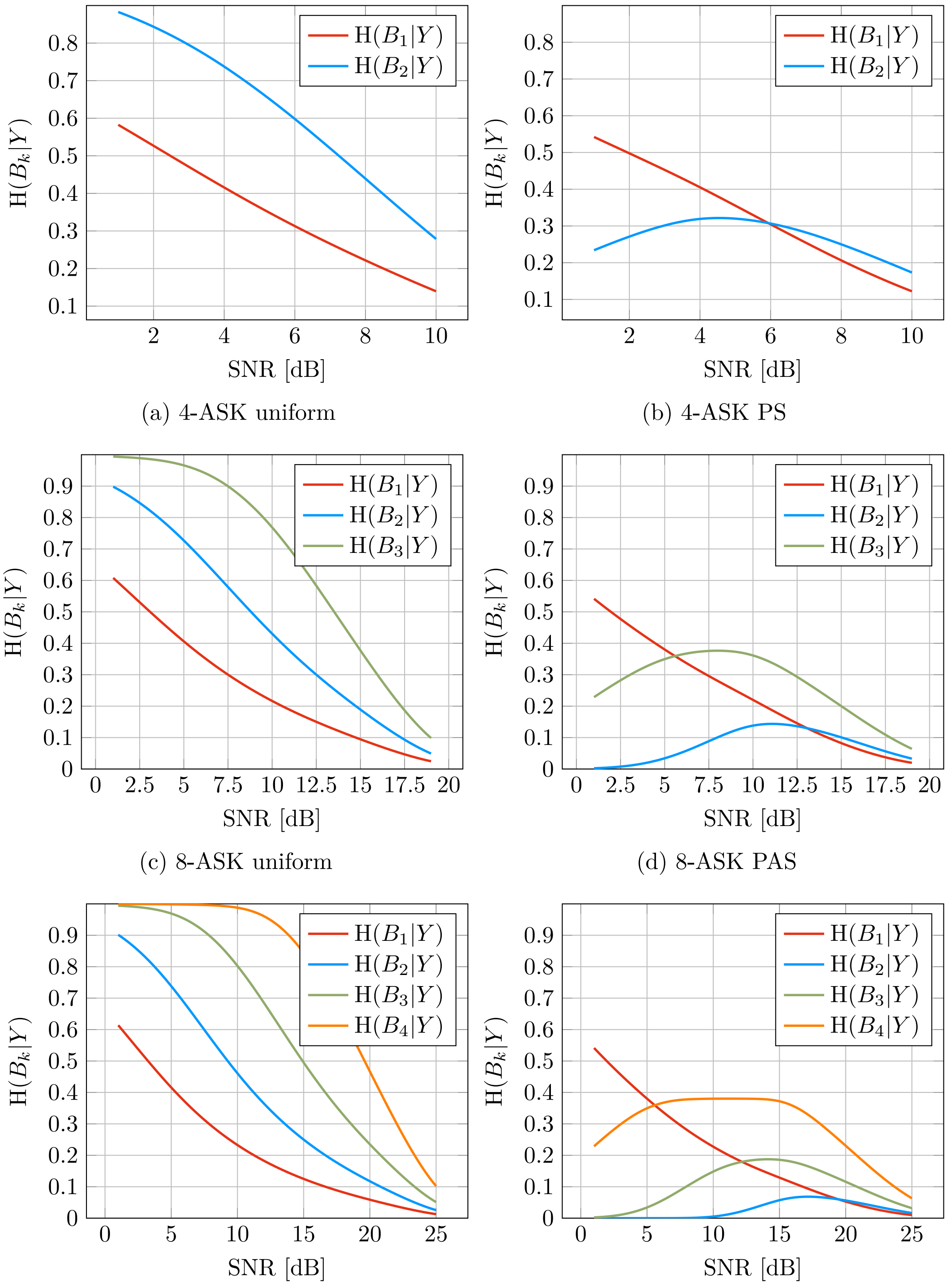
For uniform signaling, the $\mathrm{H}(B_k|Y)$ curves are monotonous and decrease for increasing SNR. Further, the bit levels are ordered in the sense that $\mathrm{H}(B_k|Y) \leq \mathrm{H}(B_{k+1}|Y)$, $k = 1, \ldots, m-1$. For PAS, the picture changes for the bit levels representing the amplitude, i.e., $B_2, \ldots, B_m$. While the uncertainty of bit level one still shows a monotonous behavior (bit level one is uniform), the uncertainties for bit levels two and higher first increase and then decrease again for higher SNRs. This behavior is related to the optimal input distribution being different for each SNR. For low SNRs, P_X is shaped significantly, while it becomes more and more uniform for higher SNRs. This observation is crucial for the design of optimized LDPC codes with an irregular degree profile, as the mapping of the bit channels to VNs with different degrees (for general unstructured irregular ensembles) or to different VN types (for protographs) matters.

4.3.2. Review of Existing Approaches

Various optimization techniques have been proposed to improve LDPC codes for higher order modulation with BMD. Two approaches can be distinguished in the literature and are summarized in Fig. 4.4. The red frames indicate the parts which are subject to optimization.

[7] Alternatively, multilevel coding with multistage decoding is possible as well (see also Sec. 3.11). However, this requires the design of individual codes for each bit level and the blocklength of each code is smaller for a given number of channel uses. As the performance of LDPC codes improves significantly with their blocklength, BMD with one code for all bit levels is generally preferred.

[8] In the uniform case, the mutual information $\mathrm{I}(B_k; Y)$ can be used, too, as $\mathrm{I}(B_k; Y) = \mathrm{H}(B_k) - \mathrm{H}(B_k|Y) = 1 - \mathrm{H}(B_k|Y)$ which is then only an affine transformation of the uncertainty.



(a) 4-ASK uniform

(b) 4-ASK PS

(c) 8-ASK uniform

(d) 8-ASK PAS

(e) 16-ASK uniform

(f) 16-ASK PAS

Figure 4.3.: H($B_k|Y$) for $\{4, 8, 16\}$-ASK with uniform and MB distributions optimized for each SNR. A BRGC code is used for the constellation labeling $\chi(\cdot)$.

Bit Mapping Optimization for Off-the-Shelf Codes

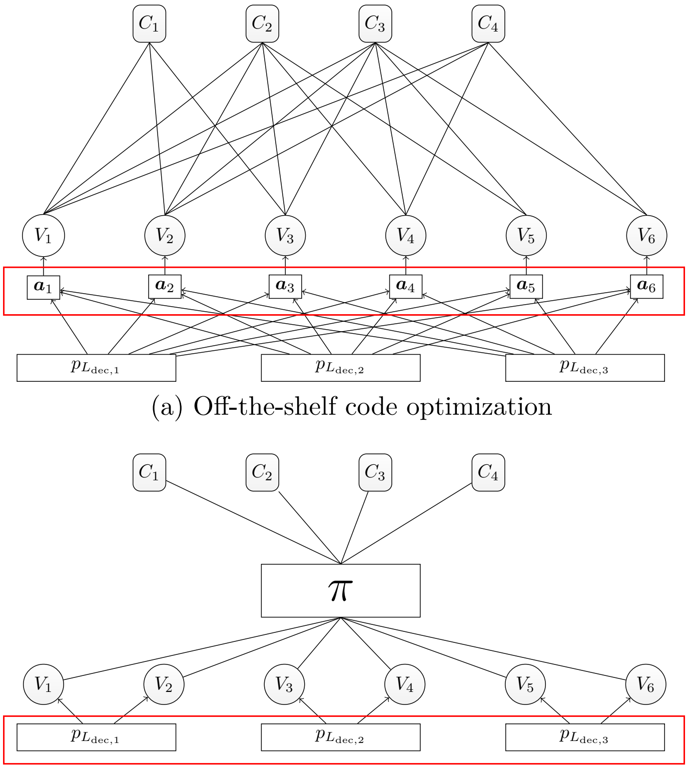
The first approach (Fig. 4.4a) considers the bit mapping optimization for an off-the-shelf LDPC code. In [137], the authors introduce the concept of variable degree matched mapping (VDMM) to optimize the association of the AR4JA VN types (see Sec. 4.1.2) to the two distinct bit channels of 16-QAM. Two schemes are investigated numerically: A *waterfilling scheme* assigns the bit channels of highest mutual information $\mathrm{I}(B_k; Y)$ to high degree VNs, whereas a *reverse waterfilling scheme* assigns the bit channels of lowest mutual information $\mathrm{I}(B_k; Y)$ to low-degree VNs. A gain of $0.15\,\mathrm{dB}$ is observed between the two schemes, but a random assignment is shown to perform at least as well as the waterfilling scheme. We revisit this example later in Sec. 4.4.2. In [138, 139], the authors use MET descriptions to optimize extended irregular repeat-accumulate (eIRA) codes for 8-PSK and 16-QAM using EXIT analysis and DDE.

The authors of [140] build upon the previous work of [137] and use P-EXIT to find optimized bit mappings by trying out all possible permutations ($4! = 24$ for four transmitted VNs types in a rate 1/2 AR4JA protograph) and deriving rules to find equivalent ones. In [141], the authors discuss the inflexibility of the previous approaches, as they require that the number of transmitted VNs types is an integer multiple of the number of bit levels m. To circumvent this, they first lift the protograph by a factor of m such that it can be combined with any modulation format. Still, only plain waterfilling VDMM mapping is used. The authors of [142] then introduce generalized variable degree matched mapping (GVDMM), which uses the lifting technique of [141], and they perform an exhaustive search over all possible permutations for various APSK and QAM constellations and the AR4JA codes. Improved decoding thresholds over the waterfilling principle are reported.

In [143], the authors aim to improve the bit mapping for DVB-S2 LDPC codes for 64- and 256-APSK constellations. They introduce an assignment matrix that defines what fraction of a certain bit level is assigned to each VN degree and optimize its entries by a multigrid search algorithm. The subsequent works [144, 145] adopt the notion of an assignment matrix to establish a relation of the fraction of the bit levels assigned to a VN type for structured codes.

Joint LDPC Code Design and Bit Mapping Optimization

The second approach (Fig. 4.4b) designs the code and the bit mapping jointly to exploit additional degrees of freedom. The first attempt in this regard was [146], where the authors used MET ensembles and modified the EXIT analysis to design optimized LDPC codes for higher order modulations using linear programming. In [147, 148], we find optimized protograph ensembles for uniform and shaped signaling. The approach builds heavily on the surrogate channel approach which is outlined in the following.



(a) Off-the-shelf code optimization

(b) Joint LDPC code design and bit mapping optimization

Figure 4.4.: Strategies for optimizing the bit mapping for LDPC codes and BMD.

4.3.3. Surrogate Based LDPC Code Design for Protographs

Previous work [149, 150, 151] has shown that LDPC codes tend to exhibit a *universal behavior*. Universality refers to the fact that decoding performance is similar over various channels for a common metric, e.g., the MI. In [149], this property was explicitly used for code design, i.e., a code is designed specifically for one channel, but operated on another. Such an approach is called *surrogate channel* based design. It is of great practical interest, as the optimization for a specific channel model might not be feasible or is computationally too cumbersome. A surrogate channel based design consists of two steps:

1. Choose the surrogate channel (e.g., BEC, biAWGNC).

2. Establish equivalence between the real and the surrogate channel.

The choice of the surrogate channel determines the complexity of the threshold analysis and the accuracy of the obtained result. Further, we still have to find out how the actual channel and the surrogate one should be matched, i.e., in what metric equivalence should be established.

In the following, we investigate two surrogate channels, namely the BEC and the biAWGNC and use the conditional entropy $\mathrm{H}(B|Y)$ to establish equivalence. The validity of the proposed techniques is verified by DDE.

The BEC as Surrogate Channel

The code design uses BECs for the m bit channels. The BECs have the input $\breve{X}_k$ and channel output $\breve{Y}_k$. The erasure probability (2.86) for the k-th surrogate channel is $\breve{\epsilon}_k$. As $\mathrm{H}(\breve{X}_k|\breve{Y}_k) = \breve{\epsilon}_k$, we have

$$\breve{\epsilon}_k = \mathrm{H}(B_k|Y), \quad k = 1, \dots, m. \tag{4.76}$$

The biAWGNC as Surrogate Channel

The code design uses biAWGN channels of the form $\breve{Y}_k = \breve{X}_k + \breve{N}_k$ where $\breve{X}_k \in \{-1, +1\}$ and $\breve{N}_k \sim \mathcal{N}(0, \breve{\sigma}_k^2)$. Unfortunately, no closed form expression can be given, such that the surrogate channel parameters $\breve{\sigma}_k^2$ must be determined numerically:

$$\breve{\sigma}_k^2 : \mathrm{H}(\breve{B}_k|\breve{Y}) = \mathrm{H}(B_k|Y), \quad k = 1, \dots, m. \tag{4.77}$$

Comparison of the Surrogate Thresholds with Discretized Density Evolution

In Table 4.2 (a) and (b), we compare the thresholds obtained via P-EXIT using biAWGN and BEC surrogate channels to the results obtained by DDE. We consider uniform and shaped signaling for SEs of 1.5 bpcu and 2.5 bpcu.

	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{DDE}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{EXIT}}$ (biAWGN)	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{EXIT}}$ (BEC)
4-ASK uni, reg. LDPC ($d_{\mathrm{v}} = 3, R_{\mathrm{c}} = 3/4$)	10.04	10.03	9.98
8-ASK uni, reg. LDPC ($d_{\mathrm{v}} = 3, R_{\mathrm{c}} = 1/2$)	10.81	10.85	10.84
8-ASK PAS, reg. LDPC ($d_{\mathrm{v}} = 3, R_{\mathrm{c}} = 3/4$)	9.37	9.36	9.30
8-ASK PAS, opt. protograph	8.80	8.77	9.00

(a) Decoding thresholds in dB for $R_{\mathrm{tx}} = 1.5$ bpcu.

	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{DDE}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{EXIT}}$ (biAWGN)	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{EXIT}}$ (BEC)
16-ASK uni,reg. LDPC ($d_{\mathrm{v}} = 3, R_{\mathrm{c}} = 5/8$)	17.41	17.39	17.43
16-ASK PAS,reg. LDPC ($d_{\mathrm{v}} = 3, R_{\mathrm{c}} = 13/16$)	15.74	15.78	15.68

(b) Decoding thresholds in dB for $R_{\mathrm{tx}} = 2.5$ bpcu.

Table 4.2.: Comparison of decoding thresholds obtained via P-EXIT (biAWGNC and BEC surrogates) and DDE.

We observe that the P-EXIT thresholds based on the biAWGN surrogate channels provide close approximations of the real DDE thresholds, while the BEC surrogates exhibit larger discrepancies, especially for the optimized protograph code with an irregular degree profile, where a gap of 0.2 dB in the respective decoding thresholds is visible. Therefore we will resort to biAWGN surrogate channels in all subsequent sections.

These results are validated in the finite length simulations of Figs. 4.5 and 4.6 for 200 and 20 BP iterations, respectively. The curves were obtained as follows: For each SNR, we derived the surrogate parameters of the $m = 2$ or $m = 3$ bit channels by (4.76) and (4.77). We then simulate the transmission of a FEC codeword over the actual channel with BMD and the parallel biAWGNCs and BECs with respective channel parameters. We see that the biAWGNC surrogates capture the finite length scaling accurately (for both 20 and 200 BP iterations), while the BEC surrogates show larger discrepancies and are usually off by 0.1 dB to 0.3 dB. The beginning of the waterfall regions in Fig. 4.6 is well reflected by the decoding thresholds of Table 4.2.

4.4. Optimizing Off-the-Shelf Protograph Based LDPC Codes

None of the previously suggested optimization approaches is tailored to the code defined in IEEE 802.3ca and also other standards (e.g., IEEE 802.11, G.hn). Some methods assume *unstructured* LDPC codes (e.g., [139, 143]) and others are prohibitively complex to work with the protograph sizes in standards. For example, the authors of [144] use differential evolution to optimize the bit mapping for protograph based spatially coupled LDPC codes with window decoding and exploit the periodicity imposed by window decoding to limit the optimization space. Furthermore, sampling from the high-dimensional polytope to generate populations for differential evolution becomes rather time consuming.

In this work, we propose an algorithm that optimizes the bit mapping of P-LDPC codes one level after the other. We use the surrogate approach of Sec. 4.3.3 and P-EXIT analysis to determine the decoding threshold for a given mapping and use the *patternsearch* algorithm [152] to find the best bit mapping. We validate this approach by comparing the predicted P-EXIT thresholds with DDE.

In the following, we use the ideas of [143, 144] to formulate an optimization procedure that optimizes the assignment of the m bit channels to the $n_{\mathrm{p},t}$ transmitted VN types of a given protograph basematrix. This bit mapping can be expressed as a non-negative matrix $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{n_{\mathrm{p},t}}]$ of dimension $m \times n_{\mathrm{p},t}$ where the entry $a_{kj} = [\boldsymbol{A}]_{kj}$ denotes the fraction of bit level k that is assigned to the j-th transmitted VN type. The matrix $\boldsymbol{A}$ needs to fulfill the constraints

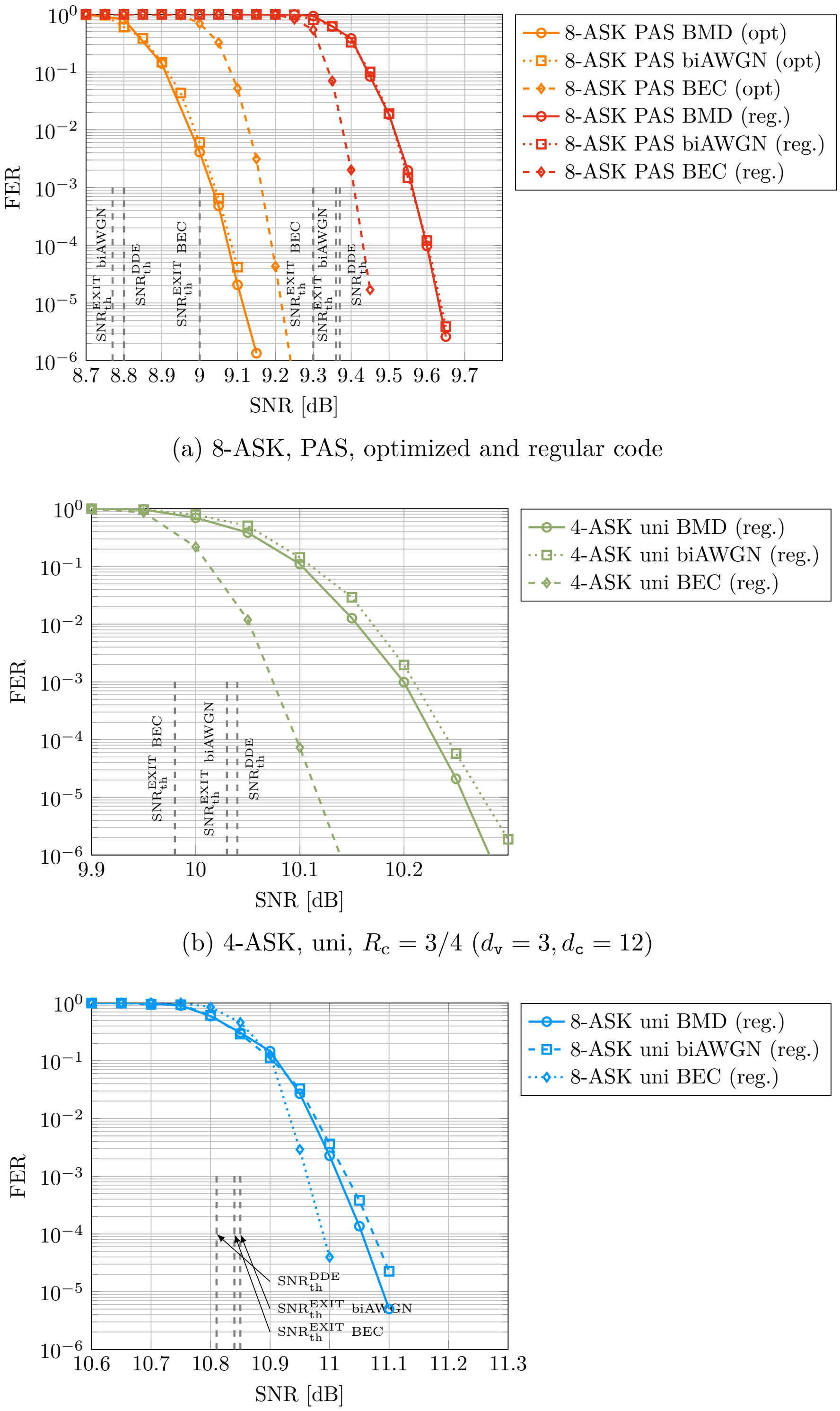
$$\sum_{j=1}^{n_{\mathrm{p},t}} a_{kj} \frac{1}{n_{\mathrm{p},t}} = \frac{1}{m}, \qquad \sum_{k=1}^{m} a_{kj} = 1, \tag{4.78}$$

for all $k \in \{1, 2, \ldots, m\}$ and $j \in \{1, 2, \ldots, n_{\mathrm{p},t}\}$. We denote the set of matrices $\boldsymbol{A}$ which fulfill the above constraints as $\mathcal{A}$. For PAS, we further impose the constraints

$$a_{1j} = 1, \qquad a_{kj} = 0, \quad j \in \{2, \ldots, m\}, \quad j \in \mathcal{V}_{\mathrm{p}}^{\mathrm{par}} \tag{4.79}$$

where $\mathcal{V}_{\mathrm{p}}^{\mathrm{par}} \subseteq \mathcal{V}_{\mathrm{p}}$ is the set of transmitted parity VNs in the protograph, as the parity VNs have to be mapped to bit level one. The set $\mathcal{A}$ is adjusted accordingly in this case.

Let the BP decoding threshold for a given basematrix $\boldsymbol{B}$, bit mapping $\boldsymbol{A}$ and signaling

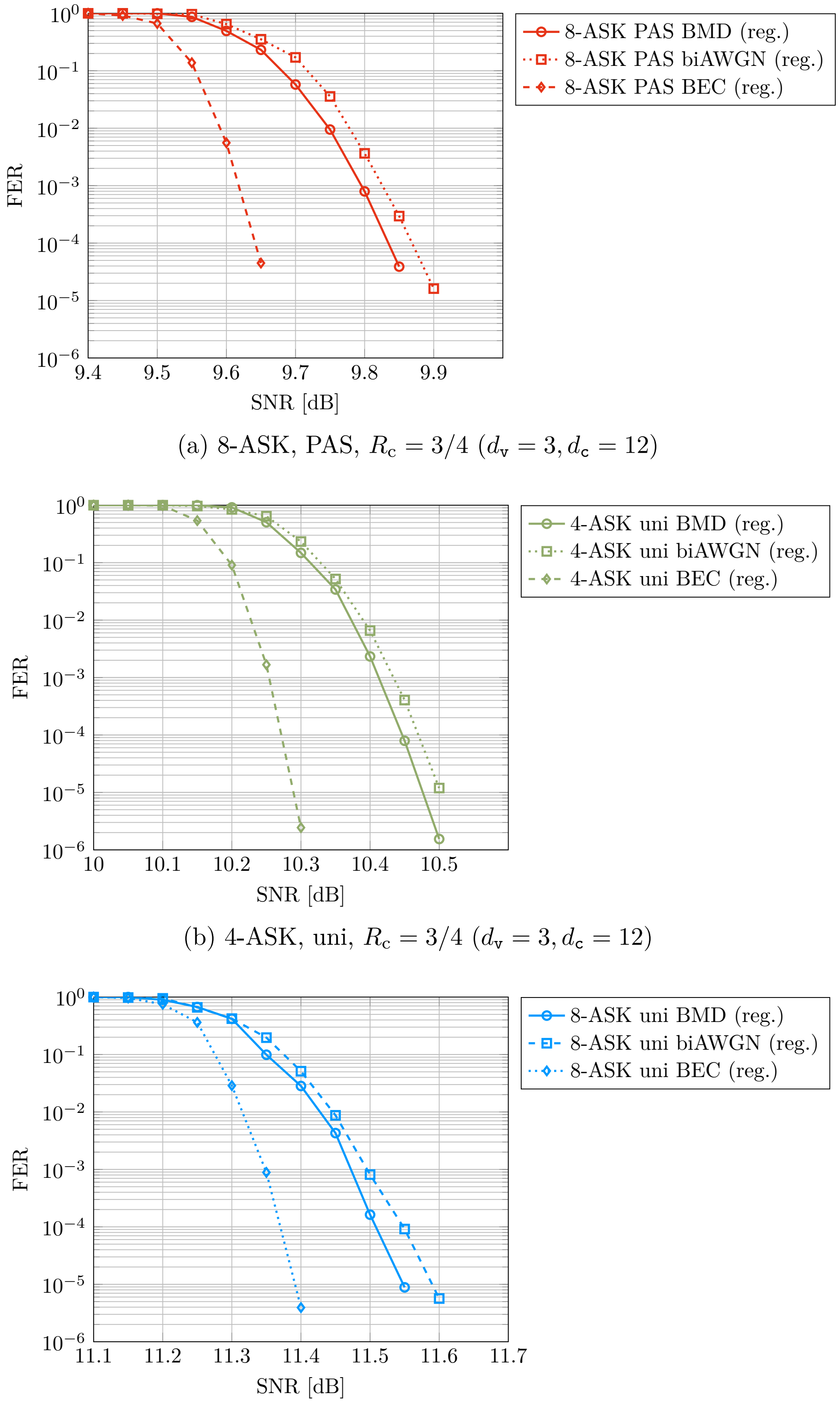


(a) 8-ASK, PAS, optimized and regular code

(b) 4-ASK, uni, $R_{\mathsf{c}} = 3/4$ $(d_{\mathtt{v}} = 3, d_{\mathtt{c}} = 12)$

(c) 8-ASK, uni, $R_{\mathsf{c}} = 1/2$ $(d_{\mathtt{v}} = 3, d_{\mathtt{c}} = 6)$

Figure 4.5.: Finite length decoding performance of various surrogate approaches with for 1.5 bpcu and 200 BP iterations.



(a) 8-ASK, PAS, $R_\mathrm{c} = 3/4$ $(d_\mathrm{v} = 3, d_\mathrm{c} = 12)$

(b) 4-ASK, uni, $R_\mathrm{c} = 3/4$ $(d_\mathrm{v} = 3, d_\mathrm{c} = 12)$

(c) 8-ASK, uni, $R_\mathrm{c} = 1/2$ $(d_\mathrm{v} = 3, d_\mathrm{c} = 6)$

Figure 4.6.: Finite length decoding performance of various surrogate approaches with for 1.5 bpcu and 20 BP iterations.

mode P_X be $\mathrm{SNR}_{\mathrm{th}}(\boldsymbol{A};\boldsymbol{B},P_X)$. The optimization problem is

$$\min_{\boldsymbol{A}} \quad \mathrm{SNR}_{\mathrm{th}}(\boldsymbol{A};\boldsymbol{B},P_X) \qquad \text{subject to } \boldsymbol{A}\in\mathcal{A}. \tag{4.80}$$

The obvious choice for calculating the decoding threshold is DDE. However, DDE takes a couple of seconds for the considered protograph sizes. Its use as part of an optimization algorithm which evaluates the objective many times is therefore limited. Instead, we use the surrogate approach of Sec. 4.3.3 and P-EXIT. We find the biAWGN surrogate with parameter $\breve{\sigma}_j^2$ for the j-th VN type as

$$\breve{\sigma}_j^2 : \mathrm{H}(\breve{X}_j|\breve{Y}_j) = \sum_{k=1}^{m} a_{kj}\,\mathrm{H}(B_k|Y), \quad j=1,\dots,n_{\mathrm{p},t}. \tag{4.81}$$

4.4.1. Successive Bit Mapping Optimization

Performing the optimization (4.80) jointly over all bit levels is a complicated task, as it involves a large number of optimization variables for large constellation sizes and protograph dimensions. Instead, we propose a successive method that optimizes each bit level one at a time while leaving the mappings of the other bit levels fixed. As a consequence, we do not optimize over the whole bit mapping matrix $\boldsymbol{A}$, but only over one row of $\boldsymbol{A}$, where the ordering is chosen as a parameter. All other bit levels are assigned uniformly. The algorithm for uniform signaling is summarized in Algorithm 5. For PAS, the function MAKE_A is modified accordingly to account for the additional constraints (4.79). For the optimization in line 3, we use *patternsearch* [152], a derivative free optimization approach that starts from a feasible initial point $\boldsymbol{x}$ (i.e., one that fulfills the constraints) and then performs a search with a set of vectors to find a direction in which the objective value improves. For our setting, we use a so-called $2N$ basis which consists of the $2N$ canonical unit vectors $\boldsymbol{e}_i, i=1,\dots,N$ of $\mathbb{R}^N$ and their negative counterparts, where N is the number of independent optimization variables. The algorithm then polls all possible new points $\boldsymbol{x}\pm s\cdot\boldsymbol{e}_i$ after an appropriate scaling ($s\in\mathbb{R}^+$) of the basis vectors and selects the one with the best objective value as the starting point for the next iteration.

4.4.2. Case Study: AR4JA

In this case study, we revisit the scenario of [137], which optimizes the bit mapping for the rate 1/2 AR4JA basematrix (4.8) for 4-ASK and uniform signaling. Fig. 4.7 shows the simulation results of various mappings, as well as our optimized mapping based on the approach in Sec. 4.4. The optimized assignment matrix is

$$\boldsymbol{A}_{\mathrm{opt}} = \begin{pmatrix} 0 & 0 & 1 & 1\\ 1 & 1 & 0 & 0\end{pmatrix} \tag{4.82}$$

Algorithm 5 Algorithmic description of the successive bit mapping optimization.

```
INPUT: Protograph $\boldsymbol{B}$, Distribution $P_X$, Ordering $\mathcal{O}$, Set of fixed row indices $\mathcal{I}$
$\boldsymbol{A}_\mathrm{F} \leftarrow [], \mathcal{I} \leftarrow []$
for $j \in \mathcal{O}$ do
    $\boldsymbol{a}_\mathrm{opt} = \mathrm{argmin}_{\boldsymbol{a}}\, \mathrm{SNR}^*(\textsc{make\_A}(\boldsymbol{a}, j, \boldsymbol{A}_\mathrm{F}, \mathcal{I}); \boldsymbol{B}, P_X)$ subject to $0 \leq a_i \leq 1 - \mathrm{sum}(\boldsymbol{A}_\mathrm{F}(:,i),1)), \forall i \in \{1, \dots, n_{\mathrm{p},t}\}$
    $\boldsymbol{A}_\mathrm{F} \leftarrow \begin{bmatrix} \boldsymbol{A}_\mathrm{F} \\ \boldsymbol{a}_\mathrm{opt} \end{bmatrix}, \mathcal{I} \leftarrow \mathcal{I} \cup \{j\}$
end for
$\boldsymbol{A}_\mathrm{opt} = \textsc{make\_A}(\{\}, \{\}, \boldsymbol{A}_\mathrm{F}, \mathcal{I})$
function $\textsc{make\_A}(\boldsymbol{a}, j, \boldsymbol{A}_\mathrm{F}, \mathcal{I})$
    $\boldsymbol{A}(j,:) \leftarrow \boldsymbol{a}$
    $\boldsymbol{A}(\mathcal{I},:) \leftarrow \boldsymbol{A}_\mathrm{F}$
    $\boldsymbol{A}([1:m] \setminus \mathcal{I},:) = (1/(m - |\mathcal{I}|)) \cdot (1 - \mathrm{sum}(\boldsymbol{A}_\mathrm{F}, 1))$
    return $\boldsymbol{A}$
end function
```

which obtains a decoding threshold of $\mathrm{SNR}_\mathrm{th}^\mathrm{DDE} = 5.77\,\mathrm{dB}$. Obviously, this does not correspond to the waterfilling solution, which corresponds to

$$\boldsymbol{A}_\mathrm{WF} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \tag{4.83}$$

and has a decoding threshold of $\mathrm{SNR}_\mathrm{th}^\mathrm{DDE} = 5.93\,\mathrm{dB}$. The random mapping corresponds to $\boldsymbol{A}_\mathrm{rnd} = 0.5 \cdot \boldsymbol{I}$ with $\mathrm{SNR}_\mathrm{th}^\mathrm{DDE} = 5.88\,\mathrm{dB}$. The parity-check matrix has dimensions $m_\mathrm{c} \times n_\mathrm{c} = 6144 \times 10240$ and is taken from [119]. Because of the puncturing, the number of transmitted bits is $n_\mathrm{c,t} = 8192$. 200 BP iterations are performed.

In Fig. 4.8 we depict the same scenario, but only 20 BP iterations are performed. Interestingly, the optimized bit mapping from before is not optimal any more, but is about 0.2 dB worse than the the waterfilling one. If we repeat the optimization from above, but only allow $\ell_\mathrm{max}^\mathrm{EXIT} = 20$ iterations for P-EXIT, we obtain the mapping

$$\boldsymbol{A}_\mathrm{opt} = \begin{pmatrix} 0.16 & 0.95 & 0 & 0.88 \\ 0.84 & 0.05 & 1 & 0.12 \end{pmatrix} \tag{4.84}$$

which essentially represents the waterfilling one of (4.83). We also observe that its finite length performance closely matches the waterfilling one. These results indicate that the number of iterations must be taken into account when the bit mapping is optimized.

4.4.3. Case Study: IEEE 802.3ca

Recently, the standardization consortium for next generation passive optical network (PON) finalized the IEEE 802.3ca standard [153], which uses an LDPC code for FEC. The proposed code has a basematrix with dimensions $m_\mathrm{p} \times n_\mathrm{p} = 12 \times 69$ and an irregular degree profile of degree three, six, eleven and twelve VNs. The circulant size is $Q = 256$, re-

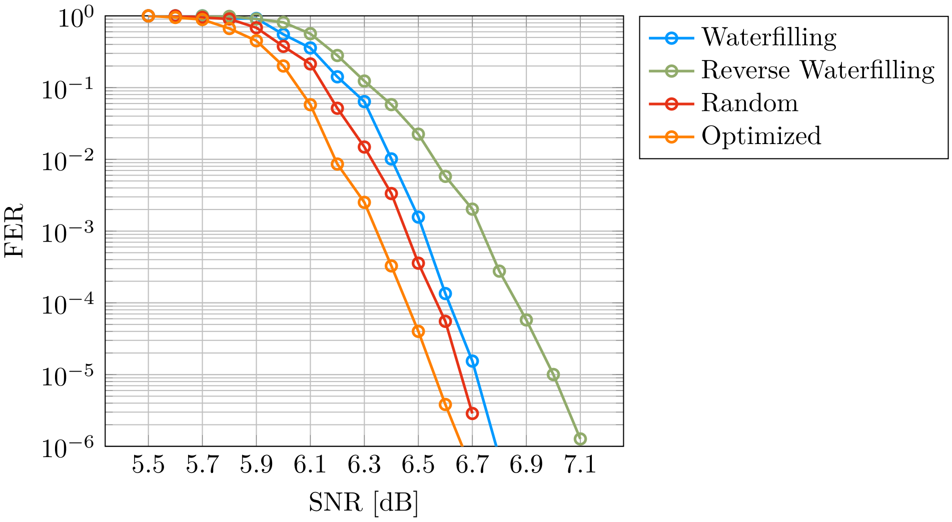


Figure 4.7.: Performance of different bit mappings for an AR4JA rate 1/2 LDPC code with 4-ASK and uniform signaling. 200 BP iterations are performed. The SE is $R_{\text{tx}} = 1.0$ bpcu.

sulting in a final parity-check matrix size of $m_{\text{c}} = 3072$ and $n_{\text{c}} = 17664$. The final graph has a girth of 6. The degree 11 and 12 VNs of the protograph are punctured. While writing this manuscript, the exact shortening pattern for the last information VN is still being discussed. In the following, we assume this VN to be shortened completely. The number of transmitted bits is therefore $n_{\text{c,t}} = 16896$ with the overall code rate of $R_{\text{c}} = 14336/16896 \approx 0.8485$.

We focus on a scenario with an SE of $R_{\text{tx}} = 2.545$ bpcu. Uniform signaling uses 8-ASK, whereas PAS uses 16-ASK with an appropriately chosen MB input distribution. The different constellation sizes are chosen such that the best performance for both signaling modes is ensured. The DM rate is $R_{\text{dm}} = 2.152$ bits.

We validate the P-EXIT thresholds by DDE and use a 8-bit quantization ($q = 255$) with $B = 15$. These values are motivated by numerical observations (e.g. in Fig. 4.17), which show that a decoder with these parameters operates with almost no loss as compared to a full resolution, floating point implementation. The obtained decoding thresholds are summarized in Table 4.3. As a reference we choose a bit mapping $\boldsymbol{A}_{\text{ref}}$ which assigns each bit level uniformly to each VN type, i.e., $\boldsymbol{A}_{\text{ref}} = 1/m \cdot \mathbf{1}$, where $\mathbf{1}$ is the all-ones matrix of size $m \times n_{\text{p},t}$. For PAS, $\boldsymbol{A}_{\text{ref}}$ is additionally adjusted to meet the constraints of (4.79).

We observe that the P-EXIT and DDE values are in good agreement with a maximum difference of 0.12 dB, which is caused by the surrogate analysis and the mixing of the bit channels. For uniform signaling the gain is 0.23 dB, and for PAS the gain is 0.37 dB based on the DDE thresholds. In both cases, the optimization yields a bit mapping $\boldsymbol{A}_{\text{opt}}$ which favors the assignment of the most reliable bit-channel (i.e., the one with the smallest $\mathrm{H}(B_k|Y)$) to the degree six VNs in the protograph. For uniform signaling, this means bit level one is mapped to the degree 6 VN types, whereas for PAS bit level two (which has the

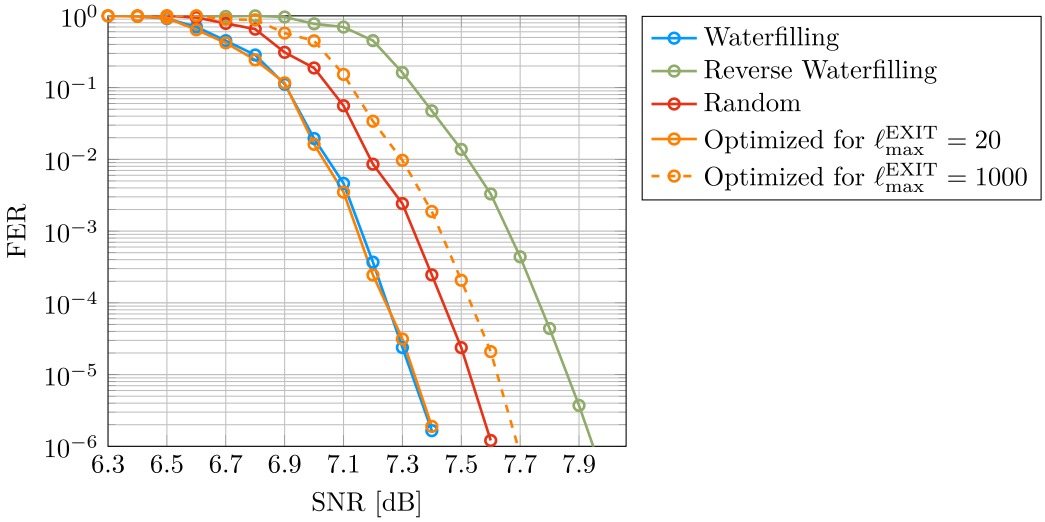


Figure 4.8.: Performance of different bit mappings for an AR4JA rate 1/2 LDPC code with 4-ASK and uniform signaling. 20 BP iterations are performed. The SE is $R_{\mathrm{tx}} = 1.0$ bpcu.

Signaling	Bit mapping	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{EXIT}}$ [dB]	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{DDE}}$ [dB]
PAS	reference	16.00	16.12
	optimized	15.67	15.75
uniform	reference	17.12	17.21
	optimized	16.90	16.98

Table 4.3.: Comparison of decoding thresholds with P-EXIT and DDE for PAS and uniform signaling.

largest prior $\log(P_{B_k}(0)/P_{B_k}(1))$ is the most reliable one, see Fig. 4.3. Empirical studies show that the ordering $\mathcal{O}$ (cf. the input of Algorithm 5) plays an important role and that the best decoding threshold is achieved by starting with the bit channel having the smallest uncertainty. This result is intuitive as the first bit channel has the largest degree of freedom for the bit mapping optimization. We validate the asymptotic results by finite length simulations in Fig. 4.9 with 100 BP iterations.

4.4.4. Case Study: DVB-S2

We now focus on the LDPC codes from the DVB-S2 standard, which defines codes of various rates from 1/4 to 9/10 and blocklength $n_{\mathrm{c}} = 64\,800$ bits. Although it is not directly visible from the parity-check matrix, these codes have a protograph representation from which the final code is derived after lifting by $Q = 360$. In the following, we describe the optimization for two signaling modes targeting $R_{\mathrm{tx}} = 1.5$ bpcu with 8-ASK (using a rate 3/4 code) and $R_{\mathrm{tx}} = 2.5$ bpcu with 16-ASK (using the rate 5/6 code). The DM parameters

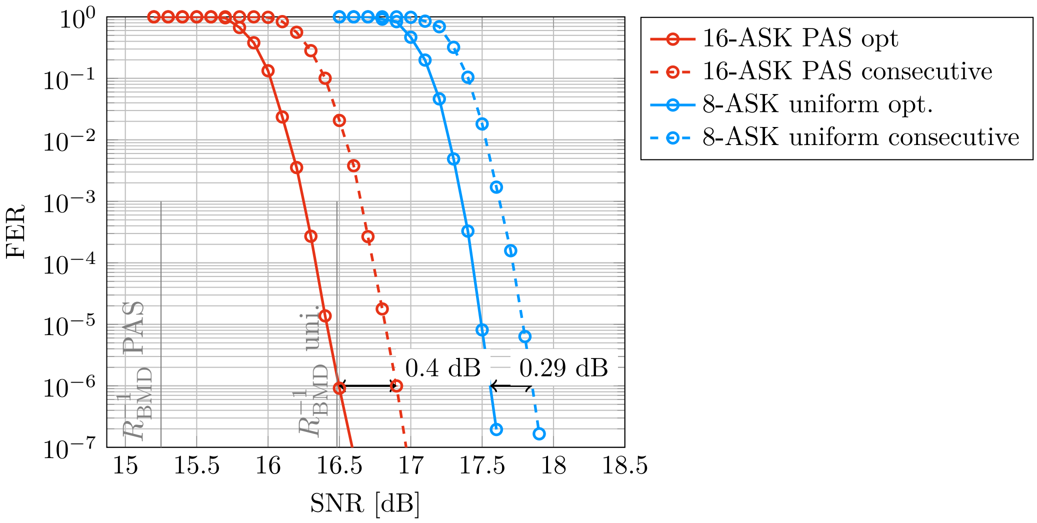


Figure 4.9.: Comparison of uniform and PAS signaling for the 802.3ca LDPC code for a target SE of 2.545 bpcu.

are summarized in Table 4.4. The considered codes have the VNs degree profile shown in

Parameter	$R_{\text{tx}} = 1.5\,\text{bpcu}$	$R_{\text{tx}} = 2.5\,\text{bpcu}$
$\vert\mathcal{A}\vert$	4	8
k_{dm}	27 000	35 100
n	21 600	16 200
R_{dm}	1.25	2.167
$\boldsymbol{t}_{\mathcal{A}}^{n}$	(13590, 6423, 1435, 152)	(5776, 4680, 3072, 1635, 705, 246, 70, 16)

Table 4.4.: CCDM parameters for the target signaling modes.

Table 4.5.

Using the approach of Sec. 4.4.1, we obtain bit mappings with decoding thresholds of $\text{SNR}_{\text{th}}^{\text{EXIT}} = 8.82\,\text{dB}$ and $\text{SNR}_{\text{th}}^{\text{EXIT}} = 15.15\,\text{dB}$. For space reasons, we show only the assignment of each bit level to the respective VN degree, i.e., each table entry denotes

$$\sum_{i \in \mathcal{V}_{\text{p},d_{\text{v}}}} a_{ji}, \quad j \in \{1, 2, \ldots, m\}$$

where $\mathcal{V}_{\text{p},d_{\text{v}}} \subseteq \mathcal{V}_{\text{p}}$ denotes the subset of protograph VNs with degree d_{v}. Table 4.6a shows the results for the $R_{\text{tx}} = 1.5\,\text{bpcu}$ case, while Table 4.6b highlights the $R_{\text{tx}} = 2.5\,\text{bpcu}$ case. In both cases, bit level one is completely assigned to the degree one and degree two VNs because of PAS. For the $R_{\text{x}} = 1.5\,\text{bpcu}$ case, bit level three (which is the least reliable one according to Fig. 4.3) is completely assigned to the degree 12 VNs. A similar observation can also be observed in $R_{\text{tx}} = 2.5\,\text{bpcu}$ case, where bit level four (again, the least reliable one) is assigned to the degree 13 VNs.

	VN degrees				
R_c	Λ_{13}	Λ_{12}	Λ_3	Λ_2	Λ_1
3/4		5400	43 200	16 199	1
5/6	5400		48 600	10 799	1

Table 4.5.: Number of VNs with the respective degrees for the considered DVB-S2 LDPC codes.

	$\mathcal{V}_{\mathrm{p},12}$	$\mathcal{V}_{\mathrm{p},3}$	$\mathcal{V}_{\mathrm{p},1,2}$
B_1	0	0.083	0.25
B_2	0	0.333	0
B_3	0.083	0.25	0

(a) $R_\mathrm{c} = 3/4$

	$\mathcal{V}_{\mathrm{p},13}$	$\mathcal{V}_{\mathrm{p},3}$	$\mathcal{V}_{\mathrm{p},1,2}$
B_1	0.0119	0.074	0.1667
B_2	0	0.25	0
B_3	0.0357	0.2143	0
B_4	0.0357	0.2143	0

(b) $R_\mathrm{c} = 5/6$

Table 4.6.: Optimized assignment of bit levels to different VN degrees for two DVB-S2 codes.

To verify the asymptotic findings, Fig. 4.10 shows the finite length simulations. As a reference scheme, we use a consecutive bit mapping. We observe that the optimized bit mapping improves the decoding performance by 0.21 dB and 0.24 dB at an FER of 10^{-4}, respectively.

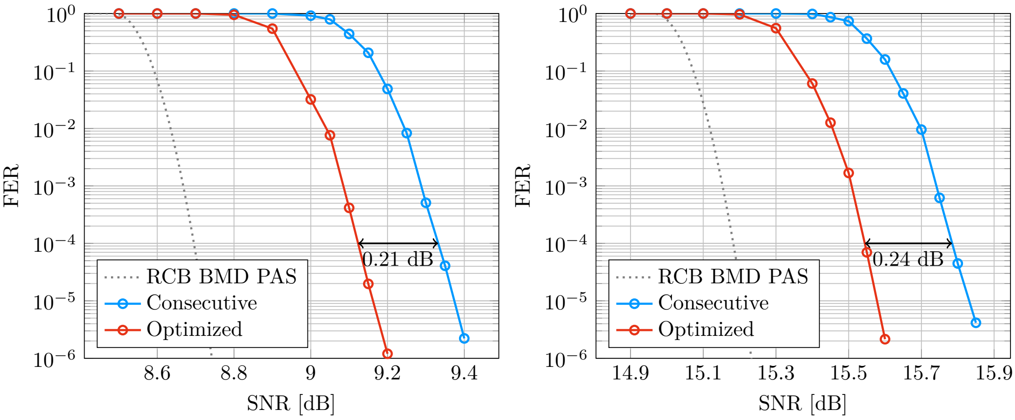
4.5. Protograph Based LDPC Code Design Examples

4.5.1. General Principles

In this section, we design optimized P-LDPC codes for different SEs using the previously introduced tool chain. The design of tailored LDPC codes can usually be decomposed into two steps:

1. Find a protograph ensemble with a good decoding threshold.
2. Construct a realization of the protograph ensemble (i.e., the parity-check matrix) by using girth optimization tools.

To accomplish step one, we use differential evolution, a genetic algorithm. Its general principle follows the one outlined in Algorithm 1 for the optimization of GS constellations, however the combination and selection steps are modified (see Appendix A.5) to take the discrete search space (entries of the basematrix) into account. For the optimization we impose restrictions that facilitate the finite length code construction in step two. For instance, we limit the maximum number of parallel edges in the basematrix to upper



(a) $R_{\text{tx}} = 1.5\,\text{bpcu}$

(b) $R_{\text{tx}} = 2.5\,\text{bpcu}$

Figure 4.10.: Performance of different bit mappings for DVB-S2 codes.

bound the maximum VN degree or limit the number of degree two VNs to ensure minimum distance growth properties.

For step two, we first lift the protograph randomly by a factor of $Q_1 \geq b_{\max}$ to remove all parallel edges and then use an adapted version of the progressive edge-growth (PEG) algorithm [154] to lift the binary basematrix in a series of liftings with lifting factors $Q_2, Q_3, \ldots, Q_{L_{\max}}$ to the final parity-check matrix of blocklength $n_c = n_p Q$ with $Q = Q_1 \cdot Q_2 \cdot \ldots Q_{L_{\max}}$, while a desired target girth is ensured. The lifting is performed in several steps as all code designs in this section are QC codes and lifting the binary basematrix directly to its final size would result in a poor minimum distance and a large multiplicity of potentially harmful structures. Numerical evaluations have shown that two additional liftings, i.e., $L_{\max} = 3$ result in good codes for the considered setup.

4.5.2. Example Designs

We concentrate on SEs of 1.5 bpcu and 2.5 bpcu and consider shaped and uniform scenarios with a blocklength of $n_c = 64\,800$ bits. The DM parameters for $R_{\text{tx}} = 1.5\,\text{bpcu}$ with a rate 3/4 code are given in Table 4.4. The parameters for $R_{\text{tx}} = 2.5\,\text{bpcu}$ and a rate 13/16 code are given in Table 4.7.

Parameter	$R_{\text{tx}} = 2.5\,\text{bpcu}$
$\lvert\mathcal{A}\rvert$	8
k_{dm}	36 450
n	16 200
R_{dm}	2.25
$\boldsymbol{t}^n_{\mathcal{A}}$	(5457, 4528, 3118, 1782, 845, 332, 109, 29)

Table 4.7.: CCDM parameters for $R_{\text{tx}} = 2.5\,\text{bpcu}$ and a rate 13/16 code.

To limit the design space[9], we impose further restrictions on the set of valid basematrices. These restrictions help to avoid harmful structures. The considered designs constraints are summarized in Tables 4.8 and 4.9. The general design guidelines were as follows:

- ▷ The selected basematrix dimensions are chosen as a trade-off to provide sufficient degrees of freedom and maintain a manageable search space. Besides, the number n_p of VNs must be an integer multiple of the number of bit levels m.
- ▷ The maximum VN degree is limited to 12 which limits the decoding complexity of the final code, see the discussion of decoder data flow (4.7).
- ▷ The number of degree two VNs is limited to $m_\mathrm{p}-1$. This constitutes a necessary condition for the protograph ensemble to have a linear minimum distance growth [155]. Further, if more than one degree 2 VN is allowed, we impose the constraint that they are placed in a staircase fashion, i.e.,

$$\begin{pmatrix} 1 & 0 & 0 & 0 & \dots \\ 1 & 1 & 0 & 0 & \dots \\ 0 & 1 & 1 & 0 & \dots \\ 0 & 0 & 1 & 1 & \dots \\ 0 & 0 & 0 & 1 & \dots \end{pmatrix}.$$

  As a result, we avoid cycles among degree 2 VNs already by construction and numerical evaluations show that these designs behave favorably compared to others from an error floor perspective. Further, if no constraints regarding the degree 2 VNs were imposed, we usually observe that those are placed on the most unreliable bit level, i.e., the one having the largest $\mathrm{H}(B_k|Y)$, $k = 1, \dots, m$. Constructing realizations of these codes and simulating them resulted in high error floors. Instead, we intentionally place them on bit level one. This usually implies a minor penalty for the decoding threshold (in the order of 0.05 dB to 0.1 dB), but the codes show steep waterfall behavior and error floors are not visible down to at least 10^{-6} in FER.

For differential evolution, we choose $G = 5000$ generations and $P = 500$ population members per generation, see Appendix A.5. The basematrices can be found in Appendix A.4.1.

The numerical results are shown in Fig. 4.11. The blue, green and red colors of the curves refer to the three different signaling modes: 8-ASK with PAS and a rate 3/4 code, 4-ASK uniform with a rate 3/4 code and 8-ASK uniform with a rate 1/2 code. All modes have have an SE of 1.5 bpcu. We show the results of regular LDPC codes with degree 3 VNs in dashed lines as references. At an FER of 10^{-3}, the optimized code gains 0.53 dB over the regular one and operates 0.37 dB from the PAS RCB for BMD. It also becomes apparent that 4-ASK with a higher rate FEC code is better than using 8-ASK with a lower

[9] Theoretically, if an exhaustive search over all basematrices was performed, the decoding threshold of up to $b_{\max}^{m_\mathrm{p} n_\mathrm{p}}$ basematrices would need to be calculated. For a rate 3/4 basematrix of dimension $m_\mathrm{p} \times = n_\mathrm{p} = 3 \times 12$ and a maximum number of $b_{\max} = 3$ parallel edges, this amounts to more than $3^{36} \approx 1.5 \times 10^{17}$ possibilities.

Parameter	$\boldsymbol{B}_{\text{8-PAS-3/4-}\Lambda_2=2}$	$\boldsymbol{B}_{\text{8-PAS-3/4-}\Lambda_2=0}$	$\boldsymbol{B}_{\text{4-uni-3/4}}$	$\boldsymbol{B}_{\text{8-uni-1/2}}$
R_c	3/4	3/4	3/4	1/2
$m_\text{p} \times n_\text{p}$	3×12	3×12	2×8	6×12
b_max	4	4	6	1
$d_{\text{v,max}}$	12	12	12	6
Λ_2	2	0	1	5
$\text{SNR}_\text{th}^\text{EXIT}$ [dB]	8.77	8.88	9.70	9.88
$\text{SNR}_\text{th}^\text{DDE}$ [dB]	8.82	8.88	9.71	9.97
ΔSNR [dB]	0.34	0.40	0.40	0.54
ω^*	4.58×10^{-4}	3.97×10^{-3}	1.01×10^{-3}	1.48×10^{-3}

Table 4.8.: Overview of optimized P-LDPC ensembles for an SE of 1.5 bpcu.

rate FEC code. For the PAS case, we also show an optimized code $\boldsymbol{B}_{\text{8-PAS-3/4-}\Lambda_2=0}$ that does not have any degree 2 VNs. It loses 0.1 dB.

For an SE of 2.5 bpcu we optimize two codes for PAS. Its parameters are shown in Table 4.9 and the simulation results are depicted in Fig. 4.12.

All protograph ensembles have a minimum distance that grows linearly with the blocklength. Comparing $\omega^\star$ for $\boldsymbol{B}_{\text{8-PAS-3/4-}\Lambda_2=2}$ and $\boldsymbol{B}_{\text{8-PAS-3/4-}\Lambda_2=0}$ we notice the huge impact of degree 2 VNs. We further note that this property is only given for informative reasons and generally does not apply to our constructed finite length codes as they have a QC design – the concept and derivation of the relative minimum distance assumes a random design, i.e., a lifting by means of arbitrary permutation matrices.

4.6. Robust LDPC Codes for Flexible Rate Adaptation

4.6.1. Need for Flexible Rate Adaptation

Practical communication systems need to adapt the SE to the channel quality. For instance, in optical systems a transceiver that operates on a short network segment with high SNR should achieve a high spectral efficiency to maximize the net data rate over this segment. Similarly, a transceiver operating on a long network segment (e.g., an intercontinental route) with low SNR should use a lower order modulation format and/or a FEC code with low code rate to ensure reliable transmission. For wireless systems, rate adaptation is important because the channel quality changes rapidly with the user's mobility or fading conditions.

As pointed out before, many existing transceivers implement rate adaptation by supporting several *modcods*, i.e., combinations of modulation formats and coding rates. For instance, LTE chooses from a set of 29 different modcods [156, Table 7.1.7.1-1] and DVB-

Parameter	$\boldsymbol{B}_{\text{16-PAS-13/16-}\Lambda_2=0}$	$\boldsymbol{B}_{\text{16-PAS-13/16-}\Lambda_2=2}$
R_c	13/16	13/16
$m_\mathrm{p} \times n_\mathrm{p}$	3×16	3×16
$b_{\max}$	3	4
$d_{\mathrm{v},\max}$	6	12
Λ_2	0	2
$\mathrm{SNR}_\mathrm{th}^\mathrm{EXIT}$ [dB]	15.36	15.26
$\mathrm{SNR}_\mathrm{th}^\mathrm{DDE}$ [dB]	15.37	15.29
$\Delta\mathrm{SNR}$ [dB]	0.43	0.35
ω^*	1.19×10^{-3}	3.07×10^{-4}

Table 4.9.: Overview of optimized P-LDPC ensembles for an SE of 2.5 bpcu.

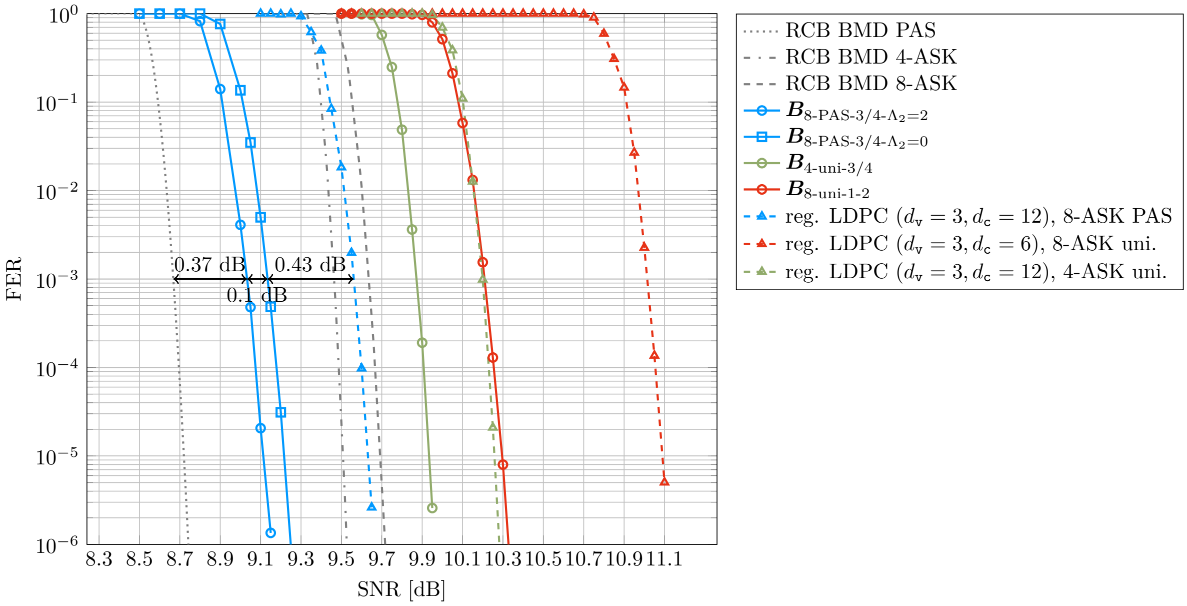


Figure 4.11.: Performance of optimized LDPC Codes for $R_\mathrm{tx} = 1.5$ bpcu and $n_\mathrm{c} = 64\,800$.

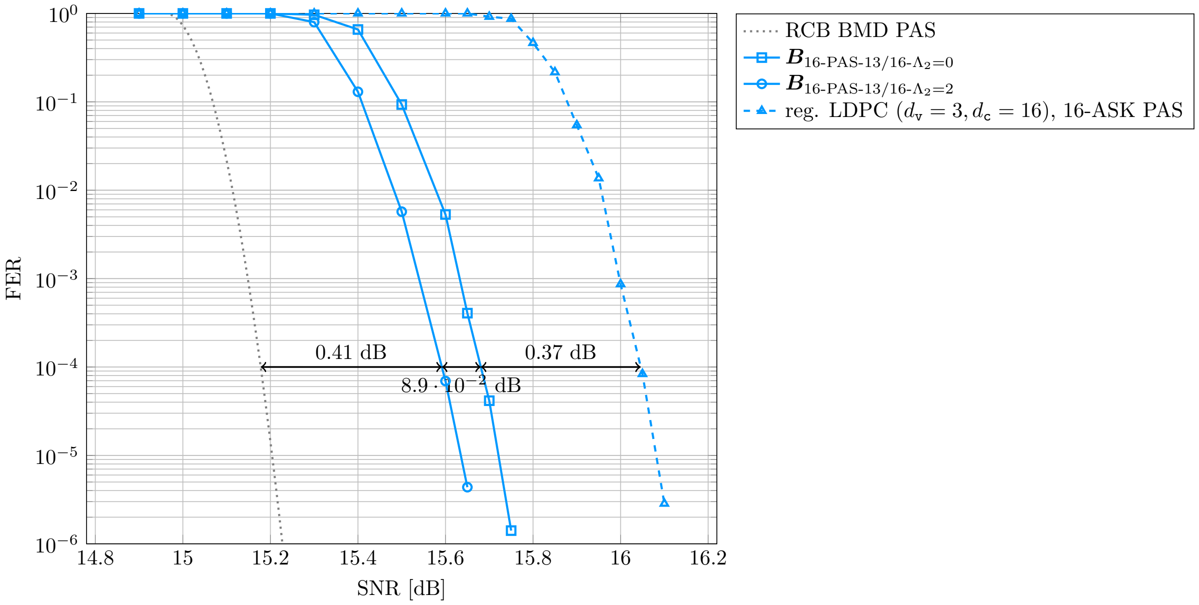


Figure 4.12.: Performance of optimized LDPC Codes for $R_{\text{tx}} = 2.5$ bpcu and $n_{\text{c}} = 64\,800$.

S2X [157] defines 116 modcods [157, Table 1], which extend the 40 modcods of DVB-S2 [98]. Here, flexibility comes at the price of increased complexity and implementation overhead. In [9], seamless rate adaptation from 2 to 10 bits per QAM symbol was demonstrated by PAS with only five modcods. Its practical applicability was shown in optical experiments in [12, 13].

We have seen in the previous sections that the performance of LDPC codes for higher order modulation with BMD depends significantly on the bit mapping. The codes in Sec. 4.5 are optimized for one particular SE and their applicability to rate adaptive transceivers is limited. We illustrate this by operating the optimized codes for $R_{\text{tx}} = 1.5$ bpcu and $R_{\text{tx}} = 2.5$ bpcu over different SEs in Fig. 4.13.

Instead, in this section, we design robust LDPC codes for rate adaptive transceivers. We exemplarily design a rate 13/16 P-LDPC code for a 16-ASK constellation to operate over the AWGN channel with any SE in the range from 0.7 bpcu to 2.7 bpcu.

In order to find the protograph ensemble with the best decoding threshold, we resort to differential evolution, see Appendix A.5. The asymptotic decoding threshold is used as a metric to select new population members, but we modify the previous evaluation step of Algorithm 10 such that

$$\boldsymbol{B}_p^{(g)} = \underset{\boldsymbol{B}\in\left\{\boldsymbol{B}_p^{(g-1)},\tilde{\boldsymbol{B}}\right\}}{\operatorname{argmin}} \; \max_{R_{\text{tx}}\in\mathcal{R}} \; \text{SNR}_{\text{th}}^{\text{EXIT}}(\boldsymbol{B}, R_{\text{tx}}) - R_{\text{BMD}}^{-1}(R_{\text{tx}}), \quad p = 1, \dots, P \tag{4.85}$$

where $\mathcal{R} \subseteq [0.7; 2.7]$ is the set of all considered operating points for which the code should be optimized and $\text{SNR}_{\text{th}}^{\text{EXIT}}(\boldsymbol{B}, R_{\text{tx}})$ denotes the decoding threshold (obtained via P-EXIT

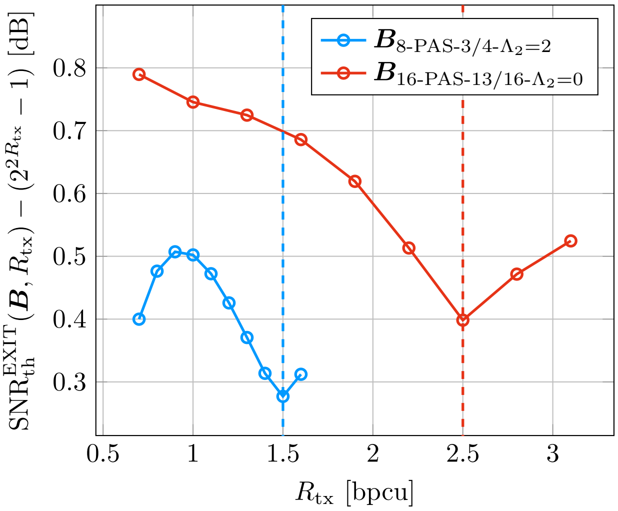


Figure 4.13.: Illustration of the gap to capacity of the decoding thresholds for different operating modes and optimized basematrices.

and a biAWGNC surrogate) of the protograph $\boldsymbol{B}$ with signaling adjusted for a target SE of R_{tx}.

4.6.2. Simulation Results

We design a rate 13/16 code which allows up to four different VN degrees per bit level such that the resulting base matrices have dimensions $m_{\mathrm{p}} \times n_{\mathrm{p}} = 3 \times 16$. The number of parallel edges is limited to 3, which results in a maximum VN degree of 9. The number of degree 2 VNs is limited to one column and all other nodes must have a degree of at least 3 to ensure a linear growth of the minimum distance [158]. We optimize basematrices for five scenarios. The first four target specific SEs of 0.7 bpcu, 1.1 bpcu, 2.1 bpcu and 2.7 bpcu, respectively. The last protograph $\boldsymbol{B}_{\mathrm{rob}}$ represents the robust approach that targets all rates in the interval $[0.7; 2.7]$ jointly and is given as

$$\boldsymbol{B}_{\mathrm{rob}} = \begin{pmatrix} 3 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 0 & 1 & 1 & 1 & 3 & 1 & 1 & 1 \\ 3 & 2 & 2 & 0 & 2 & 2 & 2 & 2 & 2 & 0 & 1 & 1 & 3 & 2 & 1 & 2 \\ 3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 3 & 3 & 0 & 0 & 0 \end{pmatrix}. \tag{4.86}$$

For optimization, we chose $\mathcal{R} = \{0.7, 1.1, 2.1, 2.7\}$. Including further rates did not improve the asymptotic decoding thresholds considerably – using less or other operating points resulted in inferior performance. We observed good results by including the boundaries of the desired operating region and pursuing the following heuristic approach: For each target SE R_{tx}, consider the entropies $\mathrm{H}(B_k|Y)$, $k = 1, \ldots, m$, e.g., in Fig. 4.3f. For different SEs, the entropies may change their ordering. If this happens, the respective rate should be added to $\mathcal{R}$. In the considered example, the ordering of $\mathrm{H}(B_1|Y)$ and $\mathrm{H}(B_4|Y)$ changes around $R_{\mathrm{tx}} \approx 1.1$ bpcu, and again the ordering of $\mathrm{H}(B_1|Y)$ and $\mathrm{H}(B_3|Y)$ changes at around $R_{\mathrm{tx}} \approx 2.1\,$bpcu. For small protograph sizes and limited degrees of freedom, it may suffice to consider the boundary operating points only.

Fig. 4.14 depicts the SNR gap of the decoding thresholds to AWGN capacity, i.e., $\mathrm{SNR}_{\mathrm{th}}^{\mathrm{EXIT}}(\boldsymbol{B}, R_{\mathrm{tx}}) - (2^{2R_{\mathrm{tx}}} - 1)$, for the considered range of SEs. Protographs which are optimized for one particular SE tend to perform poorly if operated at other SEs. This is especially the case for the codes optimized for high SEs, where the gap increases up to 1.3 dB when operated at lower SNR. The robust protograph design exhibits the desired feature of minimizing the maximum gap for each operating point in $\mathcal{R}$ and therefore achieves a balanced behavior.

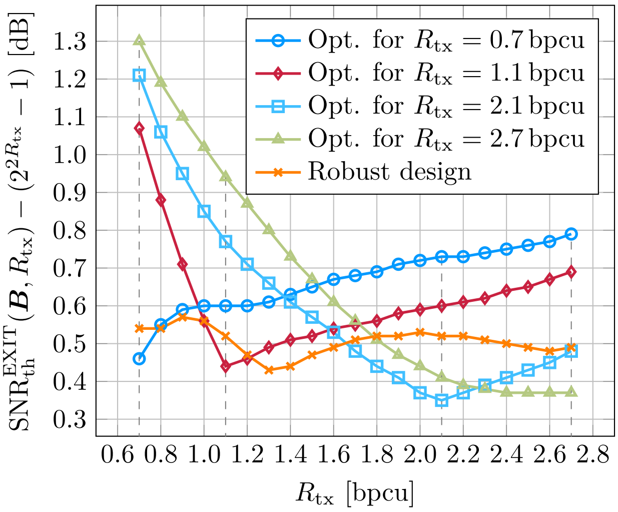


Figure 4.14.: Gap in dB to $C_{\mathrm{AWGN}}(\mathrm{SNR})$ of the asymptotic protograph BP decoding thresholds over the range of considered SEs.

For the finite length comparison in Fig. 4.15, the protographs have first been lifted by a factor of three to remove parallel edges and then by a factor of 338 to yield parity-check matrices of size 2705×16224. As a baseline for performance comparison, we choose the rate 5/6 DVB-S2 code for short frame sizes, which has a blocklength of $n = 16\,200$ [98]. For the bit mapping bit levels two to four are assigned consecutively to the first 12 150 VNs, whereas bit level one is assigned to the remaining ones. As the information part of the parity-check matrix has mostly degree three VNs and only a small number of 360 degree 13 VNs, optimizing the bit-mapper did not improve performance. In addition to the DVB-S2 reference, we also plot the PAS RCB, see Sec. 3.4.4. In all cases, 100 BP iterations with a full sum-product update rule are performed. We observe that the predictions of the asymptotic decoding thresholds are well reflected in the finite-length performance as well.

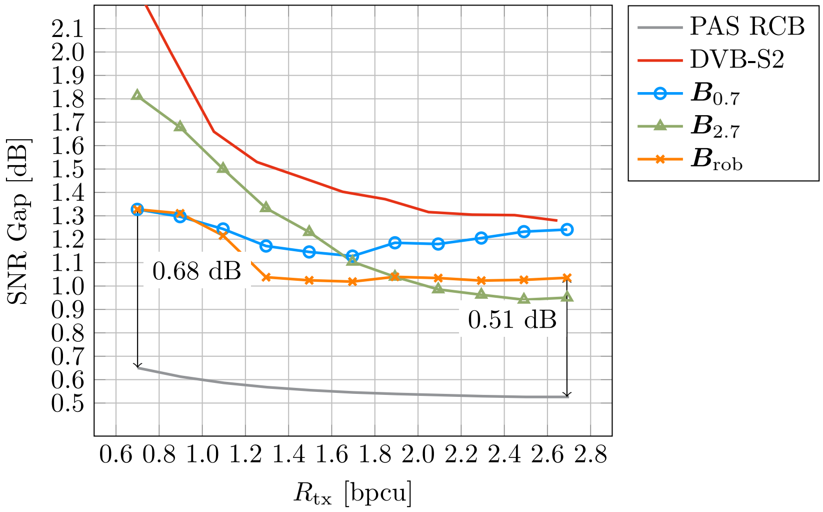


Figure 4.15.: Gap in dB to $C_{\mathrm{AWGN}}(\mathrm{SNR})$ for different protograph designs over the range of considered SEs at a target FER of 10^{-3}.

4.7. Clipping Optimization for Quantized LDPC Decoders

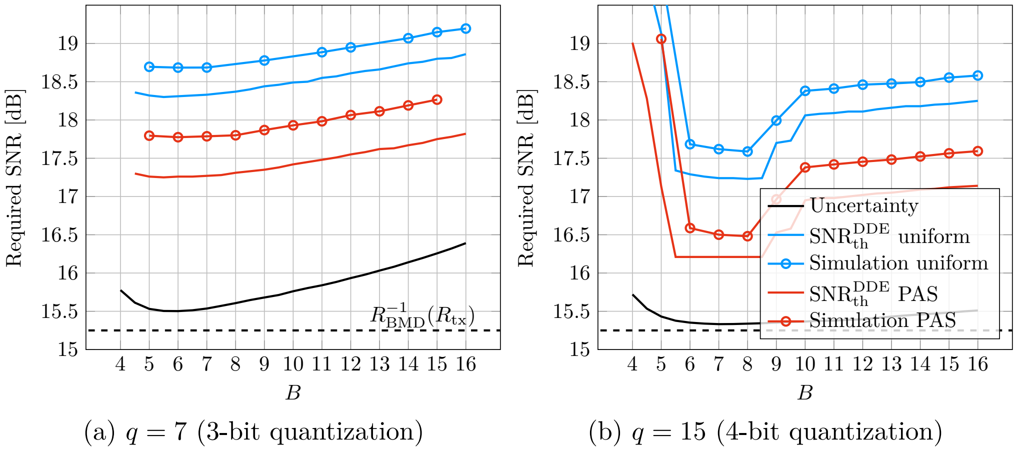
Practical LDPC decoders quantize the exchanged messages with a finite number of bits. This is particularly important for optical communication with its high throughput requirements [159, Sec. III]. Previous works [160, 161] noted that the performance of quantized decoders depends on the clipping of messages and found the optimal clipping by time consuming finite length simulations.

Instead, we optimize the clipping of a quantized sum-product LDPC decoder exemplarily for three and four bits resolution with DDE. We use the decoding threshold of an ensemble as the objective and show that DDE accurately predicts the finite length performance, making it an important tool to facilitate the design process. We use a quantized LDPC decoder as shown in [162, Sec. VI].

First we investigate the influence of the number of quantization levels q and the clipping B. As noted in [161], clipping the soft information can greatly influence the performance of the decoder and depends on the considered code ensemble.

We examine two scenarios with $b = 3$ and $b = 4$ bits resolution and investigate different approaches to find the best B. The first approach considers the mismatched uncertainty expression in (3.13). We evaluate the metric by generating soft information values according to (4.18), quantizing them (4.45) and approximating the expectation by its empirical mean in a MC manner. The second approach uses DDE of Sec. 4.2.2, and determines the decoding threshold of the LDPC code ensemble given the selected quantization and clipping parameters.

We depict the results of this analysis in Fig. 4.16 for the setting of Sec. 4.4.3. The optimized clipping is given by $B \approx 6$ for three bits and by $B \approx 8$ for four bits. The lines without markers represent the DDE thresholds, whereas the lines with markers are



(a) $q = 7$ (3-bit quantization)

(b) $q = 15$ (4-bit quantization)

Figure 4.16.: Optimal value of B based on the uncertainty, DDE decoding thresholds and finite length simulation results. The target rate is $R_{\mathrm{tx}} = 2.545$ bpcu and we depict the required SNR to achieve this SE for PAS and uniform signaling. The black curves are based on the uncertainty. The curves without markers denote the DDE decoding thresholds for uniform and PAS signaling. The curves with markers are the corresponding finite length simulation results for a frame error rate of 10^{-3}. In all cases, an optimized bit mapping is used.

finite length simulation results and denote the required SNR to obtain a target FER of 10^{-3}. Observe that the simulation results closely follow the DDE thresholds. Observe also that the mismatched uncertainty (3.13) provides a good indication for the optimal clipping value, but does not reflect the overall qualitative behavior.

In Fig. 4.17, we show the simulation results for the quantized decoders discussed in this section. We see that the loss due to quantization is about 0.25 dB for 4 bits and 1.50 dB for 3 bits compared to the unquantized case. A quantized decoder with $B = 15$ and $q = 8$ operates with almost no loss as compared to a floating point implementation with full double resolution.

4.8. Quantized Message Passing Decoders

Optical coherent transceivers with data rates of 400 Gbps are about to be installed in the field [163] and research already considers 1 Tbps. These data rates require sophisticated optical components, improved digital signal processing algorithms, and FEC solutions that can cope with the high speed. While SD decoders are superior in terms of the net coding gain (NCG), HD decoders are appealing when low power consumption and high throughputs are of paramount importance. HD-FEC for optical communications is usually based on product-like codes with Reed-Solomon (RS) or BCH component codes of high rate, which can be efficiently decoded via BDD (e.g., based on the syndrome). Spatially coupled HD-FEC constructions, such as staircase codes [159] or braided codes [164] achieve

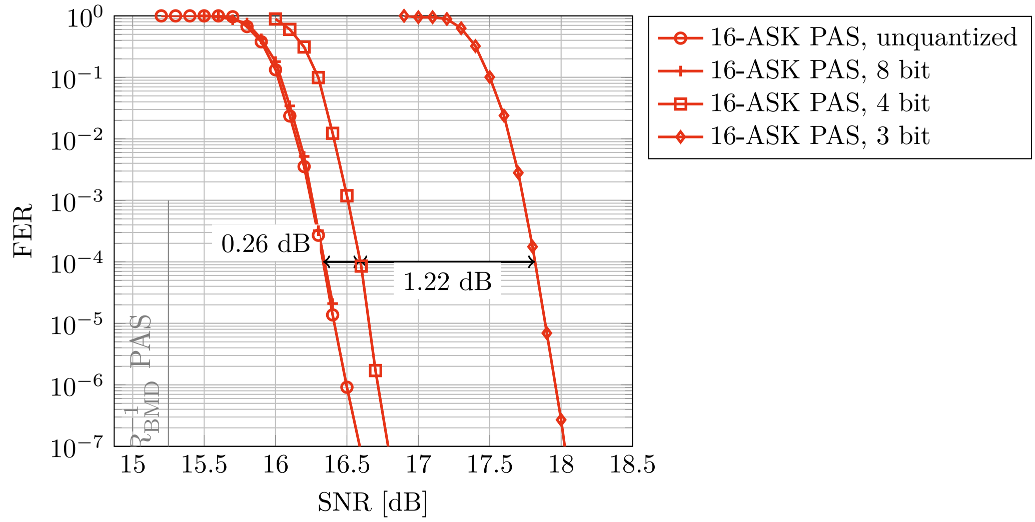


Figure 4.17.: Decoding performance of the 802.3ca LDPC code with 3, 4 and 8 bits quantization.

additional gains.

Recently, hybrid approaches based on concatenating an inner SD-FEC and an outer HD-FEC have received attention [165, 166]. These ideas have found their way into standards: the optical internet working forum established the 400G ZR standard, a specification to transmit at 400 Gbps over data center interconnect links up to 100 km, and agreed on an FEC solution consisting of an inner Hamming code and an outer staircase code, where the inner code decoder is SD and the outer code decoder is HD, with a total of 14.8% overhead and a NCG of 10.8 dB.

To exploit the soft-information from the channel, while still only exchanging binary messages during the iterations of BDD, the authors of [167] weight the HD output of the component decoders and recombine it with the soft-information from the channel, after which another HD is made. Similar approaches were also considered in [168, 169], where soft information from the channel is used to exploit particularly reliable and unreliable bits to improve the miscorrection-detection capability of the BDD decoder.

In [170], the authors present a one-bit binary message passing (BMP) algorithm for LDPC codes. In particular, the VN processor combines the soft channel message with scaled binary messages from the CNs, followed by a HD step. The idea of passing binary messages dates back to the seminal work of Gallager [5], where he presented algorithms that are now called Gallager A and Gallager B.

Thus, a BMP decoder ($q = 1$) allows to reduce the data flow F (4.7) by a factor of q compared to a decoder using q bits to represent messages.

The work in [171] extends the BMP algorithm to ternary messages. The third message is an erasure that denotes complete uncertainty about the respective bit value. The algorithm, dubbed ternary message passing (TMP) decoding, closely resembles algorithm E from [129], except that it exploits soft-information at the VNs.

4.8.1. Protograph Based Spatially Coupled LDPC Codes

Consider a spatially coupled LDPC (SC-LDPC) code with a right-unterminated parity-check matrix [172]

$$\boldsymbol{H} = \begin{pmatrix} \boldsymbol{H}_0(0) & & & \\ \boldsymbol{H}_1(0) & \boldsymbol{H}_0(1) & & \\ \vdots & \boldsymbol{H}_1(1) & \boldsymbol{H}_0(2) & \\ \boldsymbol{H}_\mu(0) & \vdots & \boldsymbol{H}_1(2) & \\ & \boldsymbol{H}_\mu(1) & \vdots & \ddots \\ & & \boldsymbol{H}_\mu(2) & \ddots \\ & & & \ddots \end{pmatrix}. \tag{4.87}$$

In (4.87), μ denotes the syndrome former memory of the SC-LDPC code. The index in brackets denotes the *spatial position*. If the matrices $\boldsymbol{H}_i(s), i \in \{0, \ldots, \mu\}$, are the same for all spatial positions $s \in \{0, \ldots, S-1\}$, the SC-LDPC code is called time-invariant and the index s can be dropped. The dimension of the matrices $\boldsymbol{H}_i(s)$ is $m_\mathrm{c}^\mathrm{SC} \times n_\mathrm{c}^\mathrm{SC}$.

Because of the diagonal structure of $\boldsymbol{H}$, a CN is connected to at most $(\mu+1)n_\mathrm{c}^\mathrm{SC}$ VNs. This allows using a window decoding approach [173] that reduces latency, increases throughput, and makes SC-LDPC codes particularly interesting for optical communications [172].

SC-LDPC codes are known to exhibit a phenomenon known as *threshold saturation* [174] that allows to approach the bit-wise MAP decoding threshold of the underlying block code with (unquantized) BP decoding.

SC-LDPC codes can be constructed from protographs and have the structure

$$\boldsymbol{B} = \begin{pmatrix} \boldsymbol{B}_0 & & & \\ \boldsymbol{B}_1 & \boldsymbol{B}_0 & & \\ \vdots & \boldsymbol{B}_1 & \boldsymbol{B}_0 & \\ \boldsymbol{B}_\mu & \vdots & \boldsymbol{B}_1 & \\ & \boldsymbol{B}_\mu & \vdots & \ddots \\ & & \boldsymbol{B}_\mu & \ddots \\ & & & \ddots \end{pmatrix}. \tag{4.88}$$

The protograph in (4.88) is then lifted by a factor of Q to obtain the final parity-check matrix $\boldsymbol{H}$.

For practical operation, the SC-LDPC code is commonly terminated after a number of S spatial positions. Due to this termination, a rate loss occurs that vanishes for large S. The resulting code rate is

$$R_\mathrm{c} = 1 - \frac{\mu + S}{S}\frac{m_\mathrm{p}^\mathrm{SC}}{n_\mathrm{p}^\mathrm{SC}} = 1 - \left(1 + \frac{\mu}{S}\right)\frac{m_\mathrm{p}^\mathrm{SC}}{n_\mathrm{p}^\mathrm{SC}} \tag{4.89}$$

where the base matrices $\boldsymbol{B}_0, \ldots, \boldsymbol{B}_\mu$ have dimensions $m_\mathrm{p}^\mathrm{SC} \times n_\mathrm{p}^\mathrm{SC}$. The overall size of the matrix $\boldsymbol{B}$ is $m_\mathrm{p} \times n_\mathrm{p} = (\mu + S)m_\mathrm{p}^\mathrm{SC} \times n_\mathrm{p}^\mathrm{SC} \cdot S$.

4.8.2. Decoding Algorithms for One And Two Bit Message Passing

In this section, we first review the BMP and TMP decoding algorithms introduced in [170] and [171, Sec. 4.8.3]. In Sec. 4.8.4, we then present a new decoding algorithm that takes full advantage of 2-bit messages, which we dub quaternary message passing (QMP).

For the described algorithms, we denote by $m_{\mathtt{c}\to\mathtt{v}}^{(\ell)}$ the message sent from CN $\mathtt{c}$ to its neighboring VN $\mathtt{v}$ at the ℓ-th iteration. Similarly, $m_{\mathtt{v}\to\mathtt{c}}^{(\ell)}$ is the message sent from VN $\mathtt{v}$ to CN $\mathtt{c}$. The soft information at the input of the decoder for the j-th coded bit is denoted by $l_{\mathrm{dec},j}$ and calculated according to (4.18).

4.8.3. Binary and Ternary Message Passing

For BMP, the exchanged messages are binary, i.e., $m_{\mathtt{v}\to\mathtt{c}}^{(\ell)}, m_{\mathtt{c}\to\mathtt{v}}^{(\ell)} \in \mathcal{M}_\mathrm{BMP} \triangleq \{-1, +1\}$. For TMP, the exchanged messages are ternary and we have $m_{\mathtt{v}\to\mathtt{c}}^{(\ell)}, m_{\mathtt{c}\to\mathtt{v}}^{(\ell)} \in \mathcal{M}_\mathrm{TMP} \triangleq \{-1, 0, +1\}$. A message value of zero indicates complete uncertainty about the respective bit.

In every decoding iteration, each VN and CN computes extrinsic messages that are forwarded to the neighboring nodes. Specifically, the message from VN $\mathtt{v}$ to CN $\mathtt{c}$ is obtained by combining the channel soft-information l_dec with a weighted version of all other incoming CN messages. Finally, a quantization function $\Psi\colon \mathbb{R} \to \mathcal{M}$ is applied to turn the result into binary and ternary messages for BMP and TMP, respectively. The weighting factors $w_{ij}^{(\ell)}$ are real valued and depend on the current iteration number. They can be obtained from the DE analysis of the respective decoding algorithm. The quantization function is

$$\Psi(x) = \begin{cases} +1, & x > 0 \\ -1, & x \leq 0 \end{cases} \tag{4.90}$$

for BMP and

$$\Psi(x) = \begin{cases} +1, & x > T \\ 0, & -T \leq x \leq T \\ -1, & x < -T \end{cases} \tag{4.91}$$

for TMP. The equality signs in (4.90) and (4.91) are chosen such that ties are broken. Note that the threshold parameter $T \in \mathbb{R}_0^+$ in (4.91) depends on the SNR and needs to be chosen for each signaling mode and iteration individually to minimize the decoding threshold. However, numerical studies reveal that a single value that is kept constant over the iterations entails almost no loss in performance. Therefore, we resort to this setting in the following.

For the CN to VN update, a CN sends the product of incoming messages from the other neighboring VNs. In the last iteration ℓ_max, the a-posteriori estimate of each codeword

Algorithm 6 BMP and TMP decoding.

```
Set $m^{(0)}_{\mathtt{v}_j \to \mathtt{c}_i} = \Psi(l_{\mathrm{dec},j}), \forall j = 1, \dots, n_{\mathrm{c}}, \forall \mathtt{c}_i \in \mathcal{N}(\mathtt{v}_j)$.
$\ell = 0$
while $\ell \le \ell_{\max}$ do
    // CN update
    for $i = 1, \dots, m_{\mathrm{c}}$ do
        for $\mathtt{v}_j \in \mathcal{N}(\mathtt{c}_i)$ do
            $m^{(\ell)}_{\mathtt{c}_i \to \mathtt{v}_j} = \prod_{\mathtt{v}_{j'} \in \mathcal{N}(\mathtt{c}_i) \setminus \{\mathtt{v}_j\}} m^{(\ell-1)}_{\mathtt{v}_{j'} \to \mathtt{c}_i}$
        end for
    end for
    // VN update
    for $j = 1, \dots, n_{\mathrm{c}}$ do
        for $\mathtt{c}_i \in \mathcal{N}(\mathtt{v}_j)$ do
            $m^{(\ell)}_{\mathtt{v}_j \to \mathtt{c}_i} = \Psi\left( l_{\mathrm{dec},j} + \sum_{\mathtt{c}_{i'} \in \mathcal{N}(\mathtt{v}_j) \setminus \{\mathtt{c}_i\}} w^{(\ell)}_{i'j} m^{(\ell)}_{\mathtt{c}_{i'} \to \mathtt{v}_j} \right)$
        end for
    end for
    $\ell = \ell + 1$
end while
// Final codeword bit estimate
for $j = 1, \dots, n_{\mathrm{c}}$ do
    $\hat{c}_j = \frac{1}{2} - \frac{1}{2} \operatorname{sign}\left( l_{\mathrm{dec},j} + \sum_{\mathtt{c}_i \in \mathcal{N}(\mathtt{v}_j)} w^{(\ell_{\max})}_{ij} m^{(\ell_{\max})}_{\mathtt{c}_i \to \mathtt{v}_j} \right)$
end for
```

bit is calculated by making a hard decision on the combined soft-information from all CN neighbors and the channel. The algorithmic procedure for BMP and TMP decoding is summarized in Algorithm 6. The weighting factors $w^{(\ell)}_{ij}$ have been derived as part of the DE for BMP and TMP in [171].

4.8.4. Quaternary Message Passing

The TMP algorithm of Sec. 4.8.3 requires two bits per exchanged message. We now introduce a QMP decoding algorithm that requires the same number of bits per exchanged message, but allows a more granular quantization of the associated reliability soft-information.

The key idea of QMP is to distinguish between low and high reliability messages. The VN to CN and CN to VN messages, $m^{(\ell)}_{\mathtt{v} \to \mathtt{c}}$ and $m^{(\ell)}_{\mathtt{c} \to \mathtt{v}}$, respectively, take values in the quaternary alphabet $\mathcal{M}_{\mathrm{QMP}} \triangleq \{-\mathtt{H}, -\mathtt{L}, +\mathtt{L}, +\mathtt{H}\}$ and $\mathtt{L}$ and $\mathtt{H}$ correspond to messages with low and high reliability, respectively. The quantization function is

$$\Psi(x) = \begin{cases} -\mathtt{H}, & x \le -T \\ -\mathtt{L}, & -T < x < 0 \\ +\mathtt{L}, & 0 \le x < T \\ +\mathtt{H}, & x \ge T. \end{cases} \tag{4.92}$$

Algorithm 7 QMP decoding.

```
Set $m^{(0)}_{\mathtt{v}_j\to\mathtt{c}_i} = \Psi(l_{\mathrm{dec},j}), \forall j = 1,\dots,n_{\mathrm{c}}, \forall \mathtt{c}_i \in \mathcal{N}(\mathtt{v}_j)$.
$\ell = 0$
while $\ell \leq \ell_{\max}$ do
    // CN update
    for $i = 1,\dots,m_{\mathrm{c}}$ do
        for $\mathtt{v}_j \in \mathcal{N}(\mathtt{c}_i)$ do
            $m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j} = \min_{\mathtt{v}_{j'}\in\mathcal{N}(\mathtt{c}_i)\setminus\{\mathtt{v}_j\}} |m^{(\ell-1)}_{\mathtt{v}_{j'}\to\mathtt{c}_i}| \cdot$
                $\prod_{\mathtt{v}_{j'}\in\mathcal{N}(\mathtt{c}_i)\setminus\{\mathtt{v}_j\}} \mathrm{sign}\left(m^{(\ell-1)}_{\mathtt{v}_{j'}\to\mathtt{c}_i}\right)$
        end for
    end for
    // VN update
    for $j = 1,\dots,n_{\mathrm{c}}$ do
        for $\mathtt{c}_i \in \mathcal{N}(\mathtt{v}_j)$ do
            $l^{(\ell)}_{\mathrm{av}} = \sum_{\mathtt{c}_{i'}\in\mathcal{N}(\mathtt{v}_j)\setminus\{\mathtt{c}_i\}} \mathrm{sign}(m^{(\ell)}_{\mathtt{c}_{i'}\to\mathtt{v}_j}) w^{(\ell)}_{i'j,|m^{(\ell)}_{\mathtt{c}_{i'}\to\mathtt{v}_j}|}$
            $m^{(\ell)}_{\mathtt{v}_j\to\mathtt{c}_i} = \Psi\left(l_{\mathrm{dec},j} + l^{(\ell)}_{\mathrm{av}}\right)$
        end for
    end for
    $\ell = \ell + 1$
end while
// Final codeword bit estimate
for $j = 1,\dots,n_{\mathrm{c}}$ do
    $l_{\mathrm{in}} = \sum_{\mathtt{c}_i\in\mathcal{N}(\mathtt{v}_j)} \mathrm{sign}(m^{(\ell_{\max})}_{\mathtt{c}_i\to\mathtt{v}_j}) w^{(\ell_{\max})}_{ij,|m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}|}$
    $\hat{c}_j = \frac{1}{2} - \frac{1}{2}\,\mathrm{sign}\left(l_{\mathrm{dec},j} + l_{\mathrm{in}}\right)$
end for
```

The QMP decoding algorithm is summarized in Algorithm 7. At the CNs, a min-sum decoding rule is employed. At the VNs, the incoming messages are weighted and combined with the channel soft-information. In contrast to BMP and TMP, two sets of weighting factors are needed for QMP depending on the magnitude of the received message. The weights $w^{(\ell)}_{ij,\mathtt{L}}$ are used for messages with low reliability (i.e., $m_{\mathtt{c}\to\mathtt{v}} \in \{-\mathtt{L}, +\mathtt{L}\}$), whereas $w^{(\ell)}_{ij,\mathtt{H}}$ are used for messages with high reliability (i.e., $m_{\mathtt{c}\to\mathtt{v}} \in \{-\mathtt{H}, +\mathtt{H}\}$).

4.8.5. Initialization of Density Evolution for Different Bit Channels

We associate each VN type with a bit level. Let $\phi(j)$ be the bit level on which the VNs of type $\mathtt{V}_j$ are mapped and let $\mathcal{V}^{(k)}_{\mathrm{p}}$ be the subset of protograph VNs that are mapped to the k-th bit level. We assume that the number n_{p} of VNs in the protograph is an integer multiple of m, such that each bit level is assigned to the same number of VNs.

Let $p^{(\ell)}_{\mathtt{m}}(i,j)$ be the probability that the message sent from $\mathtt{V}_j$ to $\mathtt{C}_i$ at the ℓ-th iteration on one of the b_{ij} edges connecting $\mathtt{V}_j$ to $\mathtt{C}_i$ is equal to $\mathtt{m} \in \{-\mathtt{H}, -\mathtt{L}, +\mathtt{L}\}$. To initialize DE, we calculate the initial message probabilities as

$$p^{(0)}_{-\mathtt{H}}(i,j) = \int_{-\infty}^{-T} p_{\tilde{L}_{\phi(j)}|B_{\phi(j)}}(l|0)\,\mathrm{d}l \tag{4.93}$$

$$p^{(0)}_{-\mathtt{L}}(i,j) = \int_{-T}^{0} p_{\tilde{L}_{\phi(j)}|B_{\phi(j)}}(l|0)\,\mathrm{d}l \tag{4.94}$$

$$p^{(0)}_{+\mathtt{L}}(i,j) = \int_{0}^{T} p_{\tilde{L}_{\phi(j)}|B_{\phi(j)}}(l|0)\,\mathrm{d}l. \tag{4.95}$$

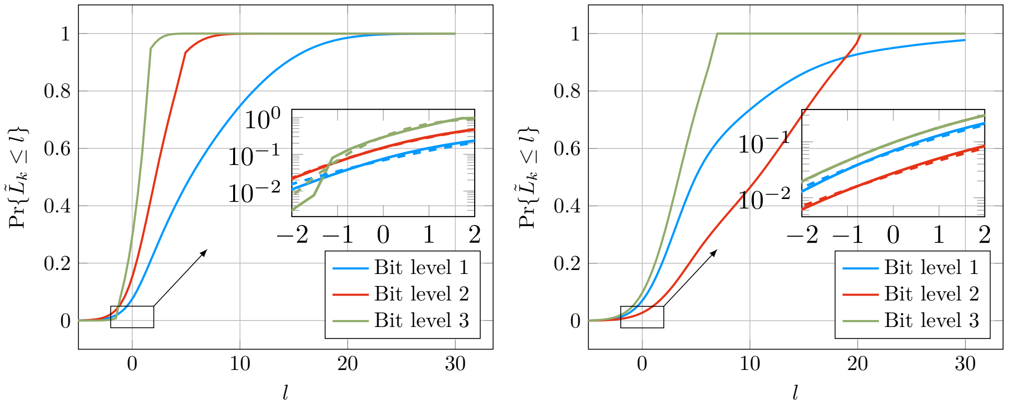
The integrals in (4.93)–(4.95) do not allow a closed form solution, but can be calculated by means of Monte Carlo simulations or transformations of RVs. Note that the above calculations need to be performed only once.

In Fig. 4.18, we show the CDFs $\Pr(\tilde{L}_k \leq l)$ for 8-ASK ($k = 1, \dots, 3$) with uniform and PS signaling obtained via Monte Carlo simulations. The CDFs can be used to calculate (4.93)–(4.95) as

$$p^{(0)}_{-\mathtt{H}}(i,j) = \Pr\{\tilde{L}_{\Phi(j)} \leq -T\} \tag{4.96}$$

$$p^{(0)}_{-\mathtt{L}}(i,j) = \Pr\{\tilde{L}_{\Phi(j)} \leq 0\} - \Pr\{\tilde{L}_{\Phi(j)} \leq -T\} \tag{4.97}$$

$$p^{(0)}_{+\mathtt{L}}(i,j) = \Pr\{\tilde{L}_{\Phi(j)} \leq T\} - \Pr\{\tilde{L}_{\Phi(j)} \leq 0\}. \tag{4.98}$$



(a) Uniform

(b) PS

Figure 4.18.: Comparison of CDF plots for 8-ASK with uniform and PS signaling. Both scenarios are for SNR = 9 dB. PS signaling uses an MB distribution with entropy H(X) = 2.25 bits. The dashed lines in the insets denote the CDFs obtained via the surrogate approach.

4.8.6. Surrogate Channels for the Initialization of Density Evolution

As an alternative to Monte Carlo simulations, we can use a *surrogate channel* approach [149, 150, 151] to approximate the required input probabilities (4.93)–(4.95). For this, the bit-channels $p_{\tilde{L}_k|B_k}$ are replaced by "equivalent" AWGN channels with uniform binary inputs for which the derivation of the CDFs is easier. We establish their "equivalence"[10] by requiring that the channel and its surrogate have the same channel uncertainty. Let the surrogate be $\breve{Y}_k = \breve{X}_k + \breve{N}_k$ with $\breve{X}_k \in \{-1, +1\}$ and $\breve{N}_k \sim \mathcal{N}(0, \breve{\sigma}_k^2)$ for $k = 1, \ldots, m$. For each SNR, we calculate the set of equivalent channel parameters

$$\breve{\sigma}_k^2 : \mathrm{H}(\breve{B}_k|\breve{Y}) = \mathrm{H}(B_k|Y), \quad k = 1, \ldots, m. \tag{4.99}$$

For QMP we obtain the expressions

$$p_{-\mathtt{H}}^{(0)}(i,j) = Q\left(\frac{T + \mu_{\mathrm{ch},\phi(j)}}{\sigma_{\mathrm{ch},\phi(j)}}\right) \tag{4.100}$$

$$p_{-\mathtt{L}}^{(0)}(i,j) = Q\left(\frac{\mu_{\mathrm{ch},\phi(j)}}{\sigma_{\mathrm{ch},\phi(j)}}\right) - Q\left(\frac{T + \mu_{\mathrm{ch},\phi(j)}}{\sigma_{\mathrm{ch},\phi(j)}}\right) \tag{4.101}$$

$$p_{+\mathtt{L}}^{(0)}(i,j) = Q\left(\frac{-T + \mu_{\mathrm{ch},\phi(j)}}{\sigma_{\mathrm{ch},\phi(j)}}\right) - Q\left(\frac{\mu_{\mathrm{ch},\phi(j)}}{\sigma_{\mathrm{ch},\phi(j)}}\right) \tag{4.102}$$

where $\mu_{\mathrm{ch},k} = 2/\breve{\sigma}_k^2$, $\sigma_{\mathrm{ch},k}^2 = 4/\breve{\sigma}_k^2$, and $Q(\cdot)$ is the standard normal Gaussian tail probability, i.e.,

$$Q(x) = \int_x^\infty (1/\sqrt{2\pi}) \exp(-\tau^2/2)\,\mathrm{d}\tau. \tag{4.103}$$

In Fig. 4.18, we show the approximations of the true CDFs by the surrogate approach (dashed lines). A close match of the true CDFs and their approximations is observed.

4.8.7. Density Evolution for Window Decoding

We follow the approach of [175] to determine the decoding threshold of protograph-based SC-LDPC code ensembles for window decoding. For this, we apply the DE analysis of [171, 176] for the respective decoding algorithm on a protograph matrix $\boldsymbol{B}_{[1:W,1:W]}$ that has been derived from (4.88) for a given decoding window size of W with $\mu + 1 \le W \le L$. The notation $\boldsymbol{B}_{[1:W,1:W]}$ denotes the block matrix of size $W \times W$ that is formed from the first

[10] The term "equivalence" is not meant to have a strict information theoretic meaning in this context. Rather, this term refers to the observation that both types of threshold evaluations yield similar results numerically.

Mode	R_{tx} [bpcu]	$R_{\mathrm{BMD}}^{-1}(R_{\mathrm{tx}})$ [dB]
4U-0.50	1.0	5.2803
4U-0.75	1.5	9.3084
8PS-0.67	1.5	8.5334
8PS-0.83	1.5	8.5606

Table 4.10.: Operating modes and their capacities for SE = 1.5 bpcu.

W block rows and W block columns of $\boldsymbol{B}$. For instance, for $\mu = 2$ and $W = 4$ we have

$$\boldsymbol{B}_{[1:4,1:4]} = \begin{pmatrix} \boldsymbol{B}_0 & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{B}_1 & \boldsymbol{B}_0 & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{B}_2 & \boldsymbol{B}_1 & \boldsymbol{B}_0 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{B}_2 & \boldsymbol{B}_1 & \boldsymbol{B}_0 \end{pmatrix}. \tag{4.104}$$

Convergence of the window decoder is declared when the probability of decoding error for the VNs in the first block column is (approximately) zero. The respective decoding threshold is referred to $\mathrm{SNR}_{\mathrm{th}}^{\mathrm{BMP}}$ for BMP, $\mathrm{SNR}_{\mathrm{th}}^{\mathrm{TMP}}$ for TMP, and $\mathrm{SNR}_{\mathrm{th}}^{\mathrm{QMP}}$ for QMP, respectively.

4.8.8. Numerical Results

We investigate the following signaling modes. The first one operates at 1.0 bpcu, whereas the others operate at a SE of 1.5 bpcu:

1. 4U-0.50: 4-ASK uniform with $R_{\mathrm{c}} = 0.50$
2. 4U-0.75: 4-ASK uniform with $R_{\mathrm{c}} = 0.75$
3. 8PS-0.67: 8-ASK PAS with $R_{\mathrm{c}} = 0.67$
4. 8PS-0.83: 8-ASK PAS with $R_{\mathrm{c}} = 0.83$.

The required SNRs to operate at this SE are summarized in Table 4.10.

As FEC codes, we consider (asymptotically) regular, protograph-based SC-LDPC codes with VN degrees $d_{\mathrm{v}} = 4$ and $d_{\mathrm{v}} = 6$, and design rates $R_{\mathrm{c}} \in \{2/3, 3/4, 5/6\}$.

The submatrices $\boldsymbol{B}_i$ in (4.88) are given by

$$\boldsymbol{B}_i = \underbrace{\begin{pmatrix} 1 & 1 & \dots & 1 \end{pmatrix}}_{d_{\mathrm{c}}}, \quad i = 0, \dots, \mu, \tag{4.105}$$

where $\mu = d_{\mathrm{v}} - 1$. The corresponding right-unterminated ensembles are referred to via their base matrices as $\boldsymbol{B}^{d_{\mathrm{v}},d_{\mathrm{c}}}$.

$\boldsymbol{B}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{full}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{BMP}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{TMP}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{QMP}}$
$\boldsymbol{B}^{4,8}$	5.36	7.75	6.50	6.26

Table 4.11.: Decoding thresholds in dB for 4-ASK uniform and an SE of 1.0 bpcu.

$\boldsymbol{B}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{full}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{BMP}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{TMP}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{QMP}}$
$\boldsymbol{B}^{4,16}$	9.41	10.89	10.11	10.00
$\boldsymbol{B}^{6,24}$	9.34	10.72	10.0	9.88

Table 4.12.: Decoding thresholds in dB for 4-ASK uniform and an SE of 1.5 bpcu.

Asymptotic Decoding Thresholds

The decoding thresholds in Tables 4.11, 4.12 and 4.13 were obtained for window decoding and a window size of $W = 15$ spatial positions, using the procedure presented in Sec. 4.8.7. We use $T = 1.3$ as a threshold parameter for BMP, TMP, and QMP in all numerical evaluations. A maximum number of 1000 iterations per window are performed. These parameters were chosen to depict the absolute performance limits. Increasing the window size did not further affect the numerical results. We conclude that the performance of a block-based decoder is similar. For uniform signaling, we use a consecutive bit mapping of the BMD bit channel to each protograph VN, i.e., for 2^m-ASK we have

$$\mathcal{V}_{\mathrm{p}}^{(1)} = \{\mathtt{V}_1, \mathtt{V}_{1+m}, \mathtt{V}_{1+2m}, \dots, \mathtt{V}_{n_{\mathrm{p}}-(m-1)}\} \tag{4.106}$$

$$\vdots$$

$$\mathcal{V}_{\mathrm{p}}^{(m)} = \{\mathtt{V}_m, \mathtt{V}_{2m}, \mathtt{V}_{3m}, \dots, \mathtt{V}_{n_{\mathrm{p}}}\}. \tag{4.107}$$

For PAS, we must take into account that bit level one (representing the sign of the constellation points [9]) is mainly formed by parity bits and has to be placed accordingly. We choose

$$\mathcal{V}_{\mathrm{p}}^{(1)} = \{\mathtt{V}_{(n_{\mathrm{p}}/m)\cdot(m-1)+1}, \mathtt{V}_{(n_{\mathrm{p}}/m)\cdot(m-1)+2}, \dots, \mathtt{V}_{n_{\mathrm{p}}}\} \tag{4.108}$$

$$\mathcal{V}_{\mathrm{p}}^{(2)} = \{\mathtt{V}_1, \mathtt{V}_m, \mathtt{V}_{2m-1}, \dots, \mathtt{V}_{(n_{\mathrm{p}}/m-1)\cdot(m-1)+1}\} \tag{4.109}$$

$$\vdots$$

$$\mathcal{V}_{\mathrm{p}}^{(m)} = \{\mathtt{V}_{m-1}, \mathtt{V}_{2(m-1)}, \mathtt{V}_{3(m-1)}, \dots, \mathtt{V}_{(n_{\mathrm{p}}/m)\cdot(m-1)}\}. \tag{4.110}$$

These mappings are repeated for each spatial position.

The decoding threshold for full BP decoding is obtained via discretized DE [112] with 8-bit quantization and a dynamic range of the soft-information of $[-16, +16]$. Increasing the resolution had no further effect. The DM rate for the PS modes was chosen according to (3.29) and the output symbols have an MB distribution [64] with corresponding entropy.

As expected from [174], we see in Tables 4.11–4.13 that the regular ensembles under

$\boldsymbol{B}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{full}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{BMP}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{TMP}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{QMP}}$
$\boldsymbol{B}^{4,12}$	8.65	10.81	9.68	9.50
$\boldsymbol{B}^{4,24}$	8.67	10.06	9.33	9.23
$\boldsymbol{B}^{6,18}$	8.57	10.62	9.55	9.37
$\boldsymbol{B}^{6,36}$	8.59	9.88	9.21	9.10

Table 4.13.: Decoding thresholds in dB for 8-ASK PS and an SE of 1.5 bpcu.

$\boldsymbol{B}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{BMP}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{TMP}}$	$\mathrm{SNR}_{\mathrm{th}}^{\mathrm{QMP}}$
$\boldsymbol{B}^{4,16}$	10.89	10.11	10.0
$\boldsymbol{B}^{4,12}$	10.81	9.68	9.50

Table 4.14.: Decoding thresholds in dB via surrogates of selected ensembles of Table III and Table IV.

full BP decoding come close (within a few hundred of a dB) to the theoretic limits for the specific signaling modes. In previous works [170, 171], the authors observed that quantized message passing decoders have a smaller gap to the achievable rate limit for high rate codes, even when codes are specifically designed for low code rates. This is also reflected in the following results.

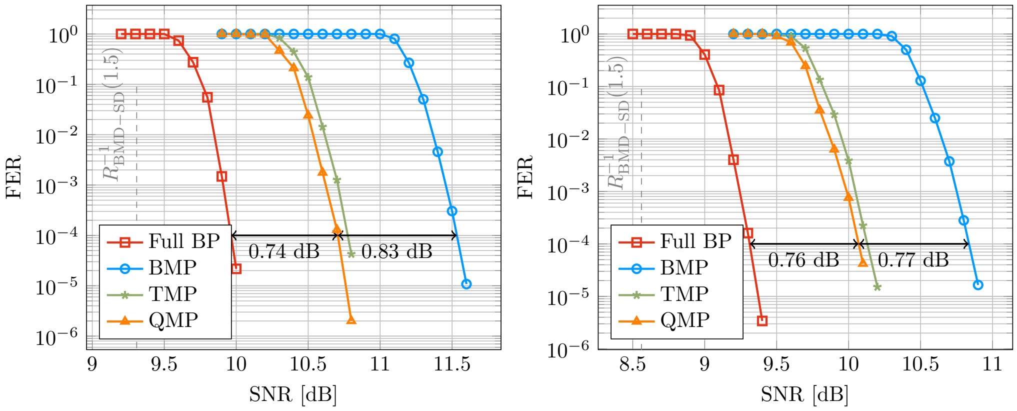
While BMP, TMP, and QMP have gaps of 2.39 dB, 1.14 dB, and 0.9 dB to the unquantized BP threshold for 4U-0.50 (i.e., for $R_{\mathrm{c}} = 1/2$), the gaps are only 1.48 dB, 0.70 dB, and 0.59 dB for 4U-0.75 (i.e., for $R_{\mathrm{c}} = 3/4$). The gain of TMP over BMP, i.e., using two bits instead of one, is significant and ranges from 0.7 dB to 1.25 dB depending on the signaling mode and code ensemble. The gain of QMP over TMP is particularly pronounced for low code rates (0.24 dB for 4U-0.50) and decreases for higher code rates to about 0.1 dB. We note that these gains can be obtained at no increase in data flow. This observation has a particular implication for PAS, where the same transmission rate can be obtained with different FEC code rates by adjusting the signaling distribution: Going from a rate 2/3 to a rate 5/6 code ($\boldsymbol{B}^{4,12}$ vs. $\boldsymbol{B}^{4,24}$) decreases the decoding threshold by 0.75 dB (BMP), 0.35 dB (TMP) and 0.27 dB (QMP). This is in contrast to full BP, where the decoding threshold even slightly deteriorates. Uniform signaling does not allow this flexibility, as the constellation order and FEC code rate directly determine the transmission rate.

In Table 4.14 we show the thresholds for the $\boldsymbol{B}^{4,16}$ (uniform) and $\boldsymbol{B}^{4,16}$ (PS) ensembles obtained via the surrogate approach of Sec. 4.8.5. We observe that the decoding thresholds numerically coincide.

Finite Length Simulations

We validate our asymptotic findings by finite length simulations with a block-based decoder for the 4U-0.75 and 8PS-0.83 signaling modes in Fig. 4.19. We use terminated SC-LDPC codes with $S = 50$ spatial positions and an overall blocklength of $n_\mathrm{c} = 60\,000$ bits. The resulting code rates are 0.735 ($\boldsymbol{B}^{4,16}$) and 0.8233 ($\boldsymbol{B}^{4,24}$) according to (4.89) with lifting factors of $Q = 300$ and $Q = 200$, respectively. We used cyclic liftings and girth optimization techniques to ensure a minimum girth of eight. Because of the termination, the effective SE is 1.47 bpcu. The weighting factors were chosen as calculated by the DE analysis at the respective decoding threshold.

For both cases, QMP gains about 0.8 dB compared to BMP. As predicted by DE, the performance of QMP improves over TMP in the order of about 0.1 dB. The gap of QMP to full BP decoding is about 0.75 dB at a FER of 10^{-4}.



(a) Uniform: 4U-0.75

(b) PAS: 8PS-0.83

Figure 4.19.: FER simulation results for uniform (a) and PAS (b) signaling and an SE of 1.5 bpcu.

4.9. Protograph Based LDPC Code Design for On-Off Keying

We now discuss the design of P-LDPC codes for the two TS schemes of Sec. 3.10. As in the previous sections, the PDF of the decoder soft information is not symmetric for OOK. Instantiating (4.18) for OOK, we have

$$l_{\mathrm{dec}} = \underbrace{\frac{A}{\sigma^2}y - \frac{A^2}{2\sigma^2}}_{\text{channel}} + \underbrace{\log\left(\frac{P_X(A)}{P_X(0)}\right)}_{\text{prior}}. \tag{4.111}$$

To find optimized protograph ensembles for the TS schemes, we resort to the surrogate approach of Sec. 4.3.3. We determine the parameters of the surrogate channels for the shaped and uniform parts as

$$\breve{\sigma}_{\mathrm{S}}^2 : \mathrm{H}(\breve{X}_{\mathrm{S}}|\breve{Y}_{\mathrm{S}}) = \mathrm{H}(X_{\mathrm{S}}|Y_{\mathrm{S}}), \tag{4.112}$$

$$\breve{\sigma}_{\mathrm{U}}^2 : \mathrm{H}(\breve{X}_{\mathrm{U}}|\breve{Y}_{\mathrm{U}}) = \mathrm{H}(X_{\mathrm{U}}|Y_{\mathrm{U}}). \tag{4.113}$$

The RVs $\breve{X}_{\mathrm{S}}$, $\breve{X}_{\mathrm{U}}$ and $\breve{Y}_{\mathrm{S}}$, $\breve{Y}_{\mathrm{U}}$ denote the input and output of the surrogate channels for the shaped and uniform part, respectively. Equation (4.112) must be solved numerically, and we obtain $\breve{\sigma}_{\mathrm{U}}^2 = (4\sigma^2)/A_{\mathrm{U}}^2$.

For the optimization with differential evolution and P-EXIT, we assume that the first $(n_{\mathrm{p}} - m_{\mathrm{p}})$ protograph VNs are associated with a biAWGN channel with variance $\breve{\sigma}_{\mathrm{S}}^2$, while the remaining m_{p} nodes are connected to biAWGN channel with variance $\breve{\sigma}_{\mathrm{U}}^2$.

As constraints, we allow for a maximum number of $m_{\mathrm{p}} - 1$ VNs of degree 2 and set the maximum number of parallel edges to four. We design three optimized base matrices, where two target an SE of 0.25 bpcu with schemes TS1 ($\boldsymbol{B}_{\mathrm{OOK-0.25-TS1}}$) and TS2 ($\boldsymbol{B}_{\mathrm{OOK-0.25-TS2}}$) and the third targets an SE of 0.67 bpcu with TS1 ($\boldsymbol{B}_{\mathrm{OOK-0.67-TS1}}$). The signaling parameters are summarized in Table 4.15 and the thresholds of the obtained basematrices are shown in Table 4.16. The basematrices can be found in Appendix A.4.2. As before, we observe that the thresholds obtained via the surrogate approach are close to the ones obtained by DDE.

	Parameters	$R_{\mathrm{tx}} = 0.25$ bpcu	$R_{\mathrm{tx}} = 0.67$ bpcu
TS1	$R_{\mathrm{TS}_1}^{-1}$ [dB]	-1.1591	4.6588
	R_{dm}	0.5	0.89
	A	1.8107	1.6790
	$P_{X_{\mathrm{S}}}(A)$	0.11	0.3063
TS2	$R_{\mathrm{TS}_2}^{-1}$ [dB]	-1.8094	–
	R_{dm}	0.3750	–
	A_{S}	3.4264	–
	A_{U}	1.6118	–
	$P_{X_{\mathrm{S}}}(A_{\mathrm{S}})$	0.0724	–

Table 4.15.: OOK signaling parameters for the considered protograph optimization.

To verify our asymptotic findings, we construct finite length codes from the ensembles given by $\boldsymbol{B}_{\mathrm{OOK-0.25-TS1}}$, $\boldsymbol{B}_{\mathrm{OOK-0.25-TS2}}$ and $\boldsymbol{B}_{\mathrm{OOK-0.67-TS1}}$ and compare to state-of-the-art off-the-shelf codes. Fig. 4.20 shows the scenario for an SE of 0.25 bpcu. The blocklength is $n = 64\,800$. For $R_{\mathrm{c}} = 0.5$ we consider TS scheme one while for $R_{\mathrm{c}} = 0.67$ we use TS scheme two. For comparison, the performance of an off-the-shelf DVB-S2 code [98] with uniform signaling with $R_{\mathrm{c}} = 0.25$ is shown. Also, the performance of two off-the-shelf DVB-S2 codes with shaping (i.e., for $R_{\mathrm{c}} = 0.5$ and $R_{\mathrm{c}} = 0.67$) is shown. We observe that in the waterfall region shaping gains 0.1 dB for case 1 and 0.35 dB for case 2, using codes from the DVB-S2 standard. However, the DVB-S2 LDPC codes show visible error floors.

Parameter	$\boldsymbol{B}_{\text{OOK}-0.25-\text{TS1}}$	$\boldsymbol{B}_{\text{OOK}-0.25-\text{TS2}}$	$\boldsymbol{B}_{\text{OOK}-0.67-\text{TS1}}$
R_{c}	$1/2$	$2/3$	$3/4$
$m_{\text{p}} \times n_{\text{p}}$	4×7	3×9	3×12
$b_{\max}$	4	4	4
$d_{\text{v,max}}$	12	12	12
$\text{SNR}_{\text{th}}^{\text{EXIT}}$ [dB]	-0.70	-1.43	4.98
$\text{SNR}_{\text{th}}^{\text{DDE}}$ [dB]	-0.68	-1.42	4.99
ΔSNR [dB]	0.47	0.39	0.33

Table 4.16.: Overview of optimized P-LDPC ensembles for OOK with TS1 and TS2.

Our designs gain 0.35 dB for case 1 and 1.06 dB for case 2, respectively.

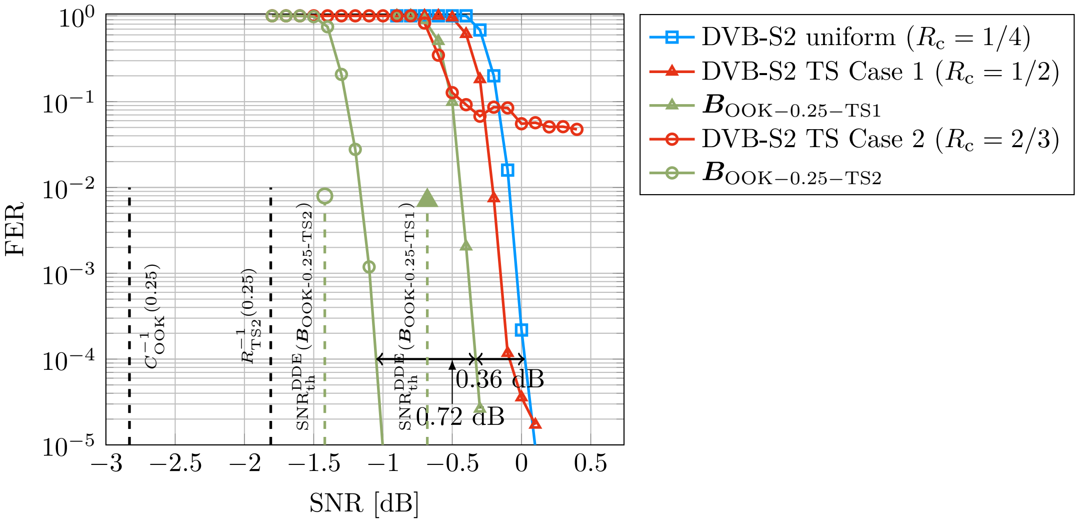


Figure 4.20.: Performance comparison of uniform and shaped modulation formats using TS for an SE of $R_{\text{tx}} = 0.25$ bpcu.

Fig. 4.21 depicts the scenario for $R_{\text{tx}} = 0.67$ bpcu and TS scheme one. Here we did not consider TS scheme two, since the achievable rate curves in Fig. 3.35 suggest only small gains. Let $n = 64\,800$ and $R_{\text{c}} = 0.75$ for $\mathcal{C}_3$. With shaping, the DVB-S2 code of $R_{\text{c}} = 0.75$ gains 0.8 dB with respect to a DVB-S2 code of $R_{\text{c}} = 0.67$ with uniform signaling. A dedicated P-LDPC code shows gains 0.92 dB with respect to the uniform case.

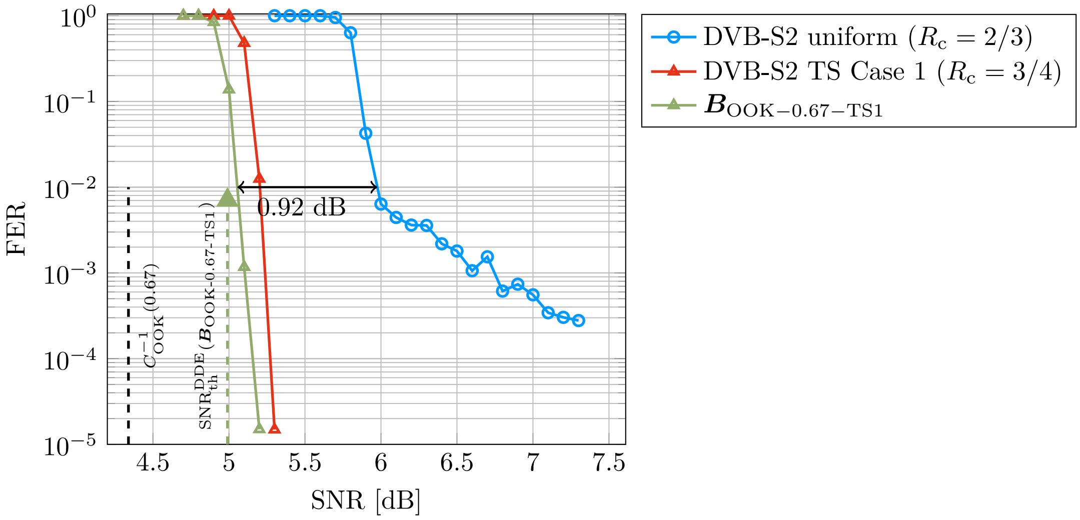


Figure 4.21.: Performance comparison of uniform and shaped modulation formats using TS for an SE of $R_{\mathrm{tx}} = 0.67$ bpcu.

5

Code Design for Non-Binary Low-Density Parity-Check Codes

5.1. Introduction

5.1.1. Non-Binary Low-Density Parity-Check Codes

Non-binary LDPC (NB-LDPC) codes are block codes defined by an $m_\mathrm{c} \times n_\mathrm{c}$ sparse parity-check matrix $\boldsymbol{H}$, where the non-zero entries of $\boldsymbol{H}$ are taken from the finite field $\mathbb{F}_q$ with $q > 2$. In this chapter, we consider only NB-LDPC codes over binary extension fields with $q = 2^o, o \in \mathbb{N}$. As for binary LDPC codes, the parity-check matrix $\boldsymbol{H}$ can be represented by a bipartite graph and each codeword symbol $v_j \in \mathbb{F}_{2^o}$ is represented by one of the n_c VNs v_j, $j = 1, \ldots, n_\mathrm{c}$, in the graph. The m_c linear constraints are represented by CNs c_i, $i = 1, \ldots, m_\mathrm{c}$. If the edge label $h_{ij} \in \mathbb{F}_{2^o}$ is non-zero, then there is an edge between v_j and c_i. Note that for binary LDPC codes, the edge labels h_{ij} were allowed to take values in $\mathbb{F}_2$ only.

We concentrate on a special class of NB-LDPC codes, namely *ultra-sparse* regular LDPC codes, which have a constant VN degree of $d_\mathsf{v} = 2$ and a constant CN degree d_c. In graph theory, ultra-sparse NB-LDPC are also known as *cycle codes* [177]. Their design rate is $R_\mathrm{c,d} = 1 - 2/d_\mathsf{c}$.

Previous works have shown that the ultra-sparse structure facilitates the design of graphs with a large girth [178] even for small blocklengths[1]. As a result, they clearly outperform the binary counterparts for short blocklengths. All discussed codes in this section further have a QC structure. It was shown in [179, Corollary 2.1] that QC ultra-sparse NB-LDPC

[1] The choice for ultra-sparse codes with $d_\mathsf{v} = 2$ is also motivated by DE results for NB-LDPC codes over large field orders, e.g., $\mathbb{F}_{64}$ and above. Here, an optimization of the degree distributions shows a dominating fraction of degree two VNs.

codes always have a girth of the form $g = 4 \cdot i, i \in \mathbb{N}$.

While the girth properties of ultra-sparse NB-LDPC codes are beneficial, their minimum distance scales only logarithmically with the blocklength [180, Sec. IV-E] so that a careful design and selection of the non-zero coefficients h_{ij} in the parity-check matrix is needed. We discuss this aspect in Sec. 5.1.3.

5.1.2. Decoding of Non-Binary Low-Density Parity-Check Codes

NB-LDPC codes are decoded by an NB version of the SPA of Sec. 4.1.4 to calculate an approximation of the symbol-MAP probability $P_{V|\boldsymbol{Y}}(v_j|\boldsymbol{y})$, $j \in 1, \ldots, n_{\mathrm{c}}$, where $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ is the vector of channel observations. The variable α denotes the primitive element of $\mathbb{F}_q$ in the following, see Sec. 2.3.5.

We define the VN to CN message vector for the ℓ-th iteration as

$$\boldsymbol{m}^{(\ell)}_{\mathtt{v}_j \to \mathtt{c}_i} = \begin{pmatrix} m^{(\ell)}_{\mathtt{v}_j \to \mathtt{c}_i}(0) \\ m^{(\ell)}_{\mathtt{v}_j \to \mathtt{c}_i}(\alpha^0) \\ \vdots \\ m^{(\ell)}_{\mathtt{v}_j \to \mathtt{c}_i}(\alpha^{q-2}) \end{pmatrix}. \tag{5.1}$$

We discuss the initialization of (5.1) for $\ell = 0$ and the derivation of the decoder soft information in greater detail in Sec. 5.2 and Sec. 5.3.

The CN to VN vector for the ℓ-th iteration is

$$\boldsymbol{m}^{(\ell)}_{\mathtt{c}_i \to \mathtt{v}_j} = \begin{pmatrix} m^{(\ell)}_{\mathtt{c}_i \to \mathtt{v}_j}(0) \\ m^{(\ell)}_{\mathtt{c}_i \to \mathtt{v}_j}(\alpha^0) \\ \vdots \\ m^{(\ell)}_{\mathtt{c}_i \to \mathtt{v}_j}(\alpha^{q-2}) \end{pmatrix}. \tag{5.2}$$

We now derive the expressions for the respective CN and VN updates. The CNs implement an NB SPC code, i.e., for the i-th CN with degree $d_{\mathtt{c}}$ and neighbors $\mathcal{N}(\mathtt{c}_i) = \{\mathtt{v}_{j_1}, \ldots, \mathtt{v}_{j_{d_{\mathtt{c}}}}\}$ the incoming messages must fulfill

$$\sum_{l=1}^{d_{\mathtt{c}}} h_{ij_l} v_{j_l} = 0. \tag{5.3}$$

Hence, we have for $\beta \in \mathbb{F}_{2^o}$

$$m_{\mathtt{c}_i \to \mathtt{v}_j}(\beta) = \Pr\Big(h_{ij_1} M_{\mathtt{v}_{j_1} \to \mathtt{c}_i} + \ldots + h_{ij}\beta + \ldots h_{ij_{d_{\mathtt{c}}}} M_{\mathtt{v}_{j_{d_{\mathtt{c}}}} \to \mathtt{c}_i} = 0\Big) \tag{5.4}$$

where $M_{\mathtt{v}_j \to \mathtt{c}_i}$ denotes the RV of the message from the j-th VN to the i-th CN. Their PMF is given by (5.1). Regarding (5.3), we introduce the short hand notation $v_j^{\pi} = h_{ij} v_j$. The dependence of v_j^{π} on the i-th CN is assumed from the context. Now, (5.3) can be written as

$$\sum_{l=1}^{d_{\mathtt{c}}} v_{j_l}^{\pi} = 0 \tag{5.5}$$

and the PMFs of the RV $M_{\mathtt{v}_j^{\pi} \to \mathtt{c}_i}$ are obtained by cyclicly shifting the entries of the PMF vector (5.1) except for the first one, i.e., for $h_{ij} = \alpha^l$, we have

$$P_{M_{\mathtt{v}_j^{\pi} \to \mathtt{c}_i}}(\alpha^k) = P_{M_{\mathtt{v}_j \to \mathtt{c}_i}}(\alpha^{k-l}), \quad k = 0, \ldots, 2^o - 2. \tag{5.6}$$

For instance, we get

$$\boldsymbol{m}_{\mathtt{v}_j^{\pi} \to \mathtt{c}_i}^{(\ell)} = \begin{pmatrix} m_{\mathtt{v}_j \to \mathtt{c}_i}^{(\ell)}(0) \\ m_{\mathtt{v}_j \to \mathtt{c}_i}^{(\ell)}(\alpha^{q-2}) \\ \vdots \\ m_{\mathtt{v}_j \to \mathtt{c}_i}^{(\ell)}(\alpha^{q-4}) \\ m_{\mathtt{v}_j \to \mathtt{c}_i}^{(\ell)}(\alpha^{q-3}) \end{pmatrix} \quad \text{for } h_{ij} = \alpha^1 \tag{5.7}$$

and

$$\boldsymbol{m}_{\mathtt{v}_j^{\pi} \to \mathtt{c}_i}^{(\ell)} = \begin{pmatrix} m_{\mathtt{v}_j \to \mathtt{c}_i}^{(\ell)}(0) \\ m_{\mathtt{v}_j \to \mathtt{c}_i}^{(\ell)}(\alpha^{q-3}) \\ \vdots \\ m_{\mathtt{v}_j \to \mathtt{c}_i}^{(\ell)}(\alpha^{q-5}) \\ m_{\mathtt{v}_j \to \mathtt{c}_i}^{(\ell)}(\alpha^{q-4}) \end{pmatrix} \quad \text{for } h_{ij} = \alpha^2. \tag{5.8}$$

Correspondingly, we have

$$m_{\mathtt{c}_i \to \mathtt{v}_j}(\beta) = \Pr\left(M_{\mathtt{v}_{j_1}^{\pi} \to \mathtt{c}_i} + \ldots + h_{ij}\beta + \ldots M_{\mathtt{v}_{j_{d_\mathtt{c}}}^{\pi} \to \mathtt{c}_i} = 0 \right). \tag{5.9}$$

Unfortunately, no closed form expression can be given for (5.9) as it was possible for the binary case, see (4.12). Instead, the probability is determined by a convolution which has complexity $\mathcal{O}(d_{\mathtt{c}}^2)$, i.e.,

$$\boldsymbol{m}_{\mathtt{c}_i \to \mathtt{v}_j} = \circledast_{\mathtt{v}_{j'} \in \mathcal{N}(\mathtt{c}_i) \setminus \{\mathtt{v}_j\}} \boldsymbol{m}_{\mathtt{v}_{j'}^{\pi} \to \mathtt{c}_i}. \tag{5.10}$$

However, for NB codes over binary extension fields computational savings are possible by implementing the convolution as a componentwise multiplication by the Hadamard transform (HT) [181]. As a result (5.10) can be written as

$$\boldsymbol{m}_{\mathtt{c}_i\to\mathtt{v}_j} = \mathcal{H}\left(\bigodot_{\mathtt{v}_{j'}\in\mathcal{N}(\mathtt{c}_i)\setminus\{\mathtt{v}_j\}} \mathcal{H}\left(\boldsymbol{m}_{\mathtt{v}^{\pi}_{j'}\to\mathtt{c}_i}\right)\right) \tag{5.11}$$

where the operator $\mathcal{H}(\cdot)$ denotes the self-inverse HT given by

$$\mathcal{H}(\boldsymbol{a}) = \frac{1}{\sqrt{2}}\begin{pmatrix}+1 & +1\\ +1 & -1\end{pmatrix}^{\otimes o}\boldsymbol{a}. \tag{5.12}$$

Efficient HT implementations reduce the complexity of calculating (5.10) to $\mathcal{O}(d_{\mathtt{c}}\log(d_{\mathtt{c}}))$.

The VNs implement an RC such that

$$\boldsymbol{m}_{\mathtt{v}_j\to\mathtt{c}_i} = \left(\bigodot_{\mathtt{c}_{i'}\in\mathcal{N}(\mathtt{v}_j)\setminus\{\mathtt{c}_i\}} \boldsymbol{m}_{\mathtt{c}^{\pi}_{i'}\to\mathtt{v}_j}\right)\odot\boldsymbol{m}_{\mathrm{dec},j} \tag{5.13}$$

where $\boldsymbol{m}_{\mathrm{dec},j}$ is the decoder soft information and $\boldsymbol{m}_{\mathtt{c}^{\pi}_i\to\mathtt{v}_j}$ denotes the PMF after the reverse cyclic shift when going from the CNs to the VNs, i.e., for $h_{ij}=\alpha^k$, we have

$$P_{M_{\mathtt{c}^{\pi}_i\to\mathtt{v}_j}}(\alpha^k) = P_{M_{\mathtt{c}_i\to\mathtt{v}_j}}(\alpha^{k+l}). \tag{5.14}$$

For instance, we get

$$\boldsymbol{m}^{(\ell)}_{\mathtt{c}^{\pi}_i\to\mathtt{v}_j} = \begin{pmatrix} m^{(\ell)}_{\mathtt{c}_i\to\mathtt{c}_j}(0)\\ m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}(\alpha^1)\\ \vdots\\ m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}(\alpha^{q-2})\\ m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}(\alpha^0)\end{pmatrix} \quad \text{for } h_{ij}=\alpha \tag{5.15}$$

and

$$\boldsymbol{m}^{(\ell)}_{\mathtt{c}^{\pi}_i\to\mathtt{v}_j} = \begin{pmatrix} m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}(0)\\ m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}(\alpha^2)\\ \vdots\\ m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}(\alpha^{0})\\ m^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j}(\alpha^1)\end{pmatrix} \quad \text{for } h_{ij}=\alpha^2. \tag{5.16}$$

If a certain number of iterations has been performed or if a stopping criteria has been reached, then the a posteriori probability vector of the j-th VN is

$$\boldsymbol{m}_{\mathrm{app},j} = \left(\bigodot_{\mathtt{c}\in\mathcal{N}(\mathtt{v}_j)} \boldsymbol{m}_{\mathtt{c}^{\pi}\to\mathtt{v}_j}\right)\odot\boldsymbol{m}_{\mathrm{dec},j}. \tag{5.17}$$

Algorithm 8 Sum-Product Decoding of NB-LDPC codes in the Probability Domain.

```
Given: $\boldsymbol{m}^{(0)}_{\mathtt{v}_j\to\mathtt{c}_i} = \boldsymbol{m}_{\mathrm{dec},j}, \forall j = 1,\dots,n_{\mathrm{c}}, i \in \mathcal{N}(\mathtt{v}_j)$.
$\ell = 0$
while $\ell \le \ell_{\max}$ do
    // CN update
    for $i = 1,\dots,m_{\mathrm{c}}$ do
        for $j \in \mathcal{N}(\mathtt{c}_i)$ do
            $\boldsymbol{m}^{(\ell)}_{\mathtt{c}_i\to\mathtt{v}_j} = \circledast_{\mathtt{v}_{j'}\in\mathcal{N}(\mathtt{c}_i)\setminus\{\mathtt{v}_j\}} \boldsymbol{m}^{(\ell)}_{\mathtt{v}^{\pi}_{j'}\to\mathtt{c}_i}$
        end for
    end for
    // VN update
    for $j = 1,\dots,n_{\mathrm{c}}$ do
        for $i \in \mathcal{N}(\mathtt{v}_j)$ do
            $\boldsymbol{m}^{(\ell)}_{\mathtt{v}_j\to\mathtt{c}_i} = \left(\bigodot_{\mathtt{c}_{i'}\in\mathcal{N}(\mathtt{v}_j)\setminus\{\mathtt{c}_i\}} \boldsymbol{m}^{(\ell)}_{\mathtt{c}^{\pi}_{i'}\to\mathtt{v}_j}\right) \odot \boldsymbol{m}_{\mathrm{dec},j}$
            Normalize $\boldsymbol{m}^{(\ell)}_{\mathtt{v}_j\to\mathtt{c}_i}$.
        end for
    end for
    $\ell = \ell + 1$
end while
// Final codeword symbol estimate
for $j = 1,\dots,n_{\mathrm{c}}$ do
    $\boldsymbol{m}_{\mathrm{app},j} = \left(\bigodot_{i'\in\mathcal{N}(\mathtt{v}_j)} \boldsymbol{m}^{(\ell)}_{\mathtt{c}_{i'}\to\mathtt{v}_j}\right) \odot \boldsymbol{m}_{\mathrm{dec},j}$
    $\hat{v}_j = \mathrm{argmax}_{v\in\mathbb{F}_q}\, m_{\mathrm{app},j}(v)$
end for
```

The decoding algorithm in the probability domain is summarized in Algorithm 8.

5.1.3. Optimization of the Non-Zero Parity-Check Matrix Entries

In contrast to binary LDPC codes, NB-LDPC codes offer an additional degree of freedom in their design by choosing the non-zero coefficients in the parity-check matrix. Even in early works [182] it was noted that a deliberate choice may yield a better performance (both in terms of the error floor and waterfall behavior) compared to a purely random selection. In [180] the authors propose several strategies to mitigate these issues.

We illustrate the problem by means of a $(d_{\mathsf{v}} = 2, d_{\mathsf{c}} = 8)$ code with $n_{\mathrm{c}} = 192$ for 8-ASK and an SE of 1.5 bpcu in Fig. 5.1. A random choice of the non-zero coefficients yields a high error floor which is due to low weight NB codewords with $d_{\min} = 4$. We also plot the undetected error rate and see that it coincides with the FER, which suggest that all frame errors are caused by wrong codewords.

To circumvent this, we choose the coefficients such that the minimum distance of the code formed by the binary image of the non-zero coefficients in a row is maximized [180]. For a NB-LDPC code over $\mathbb{F}_{2^o}$ with a CN degree of d_{c}, the code of the binary image of a row has parameters $(d_{\mathsf{c}} \cdot o, (d_{\mathsf{c}} - 1) \cdot o)$. The best set of coefficients can be found by exhaustive search for moderate field sizes and CN degrees. For a degree d_{c} CN and a field

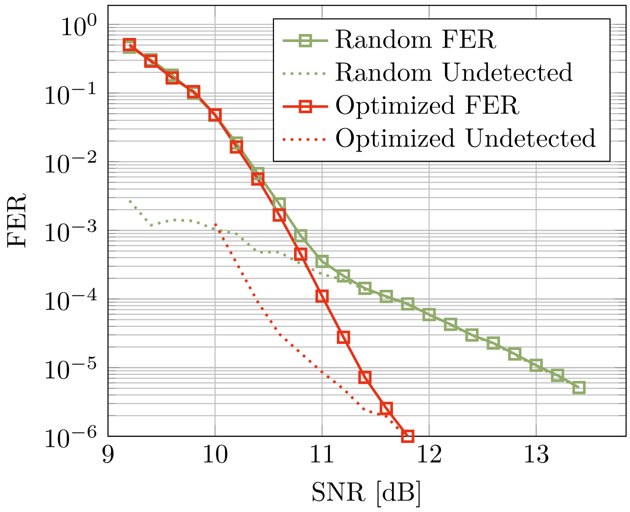


Figure 5.1.: Influence of the choice of the non-zero coefficients in the parity-check matrix of a $(d_{\mathtt{v}} = 2, d_{\mathtt{c}} = 8)$ NB-LDPC code over $\mathbb{F}_{64}$ on the decoding performance.

size of 2^o, we have to evaluate the minimum distance of $(2^o - 1)^{d_{\mathtt{c}}}$ different binary codes. As these codes usually have a high rate, this computation should be performed in the dual domain by means of the MacWilliams identity [183]. Recently, approximate methods for large field sizes and large $d_{\mathtt{c}}$ were considered as well [184]. For the considered code parameters of Fig. 5.1, we find the best choice of coefficients as

$$\begin{pmatrix} \alpha^0 & \alpha^6 & \alpha^{13} & \alpha^{21} & \alpha^{28} & \alpha^{36} & \alpha^{44} & \alpha^{54} \end{pmatrix} \tag{5.18}$$

where the underlying primitive polynomial is $1 + x + x^6$. For the optimized code we use permutations of (5.18) for each row. The code of the binary image associated with (5.18) has $d_{\min} = 3$ with a multiplicity of 276. In contrast, the binary codes associated with the random choice of coefficients in Fig. 5.1 had all $d_{\min} = 2$ with multiplicities ranging from 6 to 25.

5.1.4. Non-Binary Codes and Probabilistic Amplitude Shaping

The combination of PAS and NB codes was suggested in [185]. Herein, the authors show that the parity symbols after encoding are distributed uniformly asymptotically, even if the information symbols follow a non-uniform distribution. This property enables a straightforward application of PAS, as it extends the uniform check bit assumption [9, Fig. 2] to higher order fields. Further, the authors propose a new design for circular QAM constellations that can be used with NB codes over prime fields of order larger than two.

In what follows, we propose a different strategy and consider only NB codes over the extension field $\mathbb{F}_q$ with $q = 2^o$. This specific choice enables low complexity decoding by using the HT at the CNs while benefitting from the excellent performance for short blocklengths. Two approaches are discussed in Sec. 5.2 and Sec. 5.3, respectively. The first approach employs SMD and requires that the field and constellation order are matched. The second approach uses BMD and allows any combination of field and constellation order. At the same time, the loss of BMD is limited to a small value because of shaping.

5.2. Symbol-Metric Decoding of Non-Binary LDPC Codes

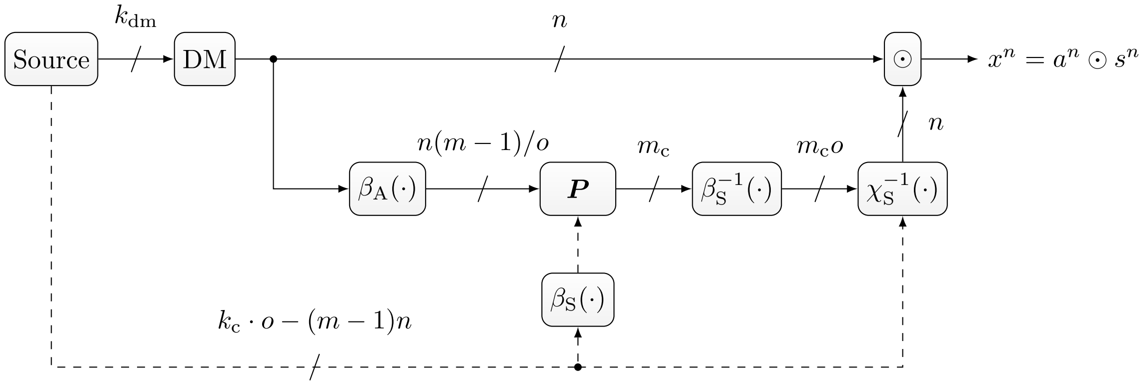

Source
k_{dm}
DM
n
$\odot$
$x^n = a^n \odot s^n$
n
$n(m-1)/o$
m_{c}
$m_{\mathrm{c}}o$
$\beta_{\mathrm{A}}(\cdot)$
$\boldsymbol{P}$
$\beta_{\mathrm{S}}^{-1}(\cdot)$
$\chi_{\mathrm{S}}^{-1}(\cdot)$
$\beta_{\mathrm{S}}(\cdot)$
$k_{\mathrm{c}} \cdot o - (m-1)n$

Figure 5.2.: Transmitter system model for PAS with SMD and NB $\mathbb{F}_q$ codes and $q = 2^o$.

The PAS system model for NB codes is depicted in Fig. 5.2 and is a natural extension of the binary case shown in Fig. 3.7. Again, we exploit the symmetry property of the optimal input distribution P_X to factorize it into independent RVs referring to the amplitude and sign (3.26). We consider a 2^m-ASK constellation. The sign distribution P_S is uniform on $\mathcal{S} = \{-1, +1\}$, while P_A is non-uniform on the amplitude set $\mathcal{A} = \{1, 3, \dots, 2^m - 1\}$.

We first use $R_{\mathrm{c}} = (m-1)/m$ codes. In Fig. 5.2, this corresponds to the absence of the dashed lines. The DM maps k_{dm} data bits to n amplitudes. The FEC encoder generates redundancy, which is mapped to the n signs. FEC encoding is systematic to preserve the amplitude distribution imposed by the DM. The combination of an amplitude and a sign results in one channel input symbol. The n channel input symbols can be represented by $m \cdot n$ bits, which requires an NB code with blocklength $n_{\mathrm{c}} = (nm)/o$. Each amplitude requires $(m-1)$ bits for its representation and we require $o = \lambda(m-1)$ when 2^m-ASK PAS is combined with NB codes over $\mathbb{F}_{2^o}$. The variable $\lambda \in \mathbb{N}$ defines the number of amplitudes in $\mathcal{A}$ which are mapped to one $\mathbb{F}_{2^o}$ symbol. We denote this mapping as

$$\beta_{\mathrm{A}} : \mathcal{A}^{\lambda} \to \mathbb{F}_{2^o}. \tag{5.19}$$

The amplitude part has a size of $k_{\mathrm{c}} = n/\lambda$ symbols and is collected in the vector $\boldsymbol{u} \in \mathbb{F}_{2^o}^{k_{\mathrm{c}}}$. Systematic encoding with $\boldsymbol{G} = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{P} \end{pmatrix}$ yields the parity part $\boldsymbol{p} = \boldsymbol{u}\boldsymbol{P}$ of $(1-R_{\mathrm{c}})n_{\mathrm{c}}$ symbols that are approximately uniformly distributed [185, Theorem I]. The decoder assumes that the signs are uniformly distributed. Using the inverse of the mapping

$$\beta_{\mathrm{S}} : \{0,1\}^o \to \mathbb{F}_{2^o} \tag{5.20}$$

we relate each parity symbol to a sign sequence.

For the decoder input we calculate the vectors

$$\boldsymbol{m}_{\mathrm{dec},j} = \begin{pmatrix} P_{V|\boldsymbol{Y}}(0|\boldsymbol{y}) \\ P_{V|\boldsymbol{Y}}(\alpha^0|\boldsymbol{y}) \\ \vdots \\ P_{V|\boldsymbol{Y}}\left(\alpha^{q-2}|\boldsymbol{y}\right) \end{pmatrix}, \quad j = 1, \dots, n_{\mathrm{c}}. \tag{5.21}$$

The expression $P_{V|\boldsymbol{Y}}(v|\boldsymbol{y}_j)$ denotes the probability that the j-th codeword symbol is $v \in \mathbb{F}_{2^o}$ when $\boldsymbol{y}_j$ was received.

We distinguish two cases for the decoder soft information vectors $\boldsymbol{m}_{\mathrm{dec},j}$ depending on whether the codeword symbol v refers to an amplitude (5.19) or sign mapping (5.20). Let $\boldsymbol{y}_j^{\mathrm{A}} = (y_{j1}, \dots, y_{j\lambda})$ be the vector of all received symbols that resulted from the transmission of the amplitudes associated with the j-th codeword symbol. Similarly, the vector $\boldsymbol{y}_j^{\mathrm{S}} = (y_{j1}, \dots, y_{jo})$ refers to the received symbols that resulted from the transmission of the signs associated with the j-th codeword symbol.

Amplitude Mappings For $j = 1, \dots, k_{\mathrm{c}}$ and $\boldsymbol{a} = (a_1, \dots, a_\lambda) = \beta_{\mathrm{A}}^{-1}(v)$, assuming uniform signs, the demapper calculates the metric

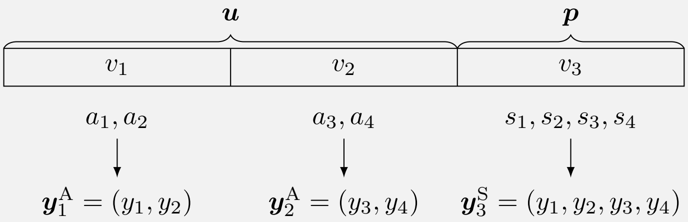
$$\begin{aligned} P_{V|\boldsymbol{Y}}(v|\boldsymbol{y}_j^{\mathrm{A}}) \propto P_{V,\boldsymbol{Y}}(v, \boldsymbol{y}_j^{\mathrm{A}}) &= P_{\boldsymbol{AY}}(\beta_{\mathrm{A}}^{-1}(v), \boldsymbol{y}_j^{\mathrm{A}}) \\ &= \prod_{l=1}^{\lambda} P_{AY}(a_l, y_{jl}) \\ &= \prod_{l=1}^{\lambda} \sum_{s\in\{\pm1\}} P_{XY}(a_l \cdot s, y_{jl}) \\ &= \prod_{l=1}^{\lambda} \frac{1}{2} P_A(a_l) \sum_{s\in\{\pm1\}} p_{Y|X}(y_{jl}|a_l \cdot s). \end{aligned} \tag{5.22}$$

Sign Mappings For the parity part $j = k_{\mathrm{c}} + 1, \dots, n_{\mathrm{c}}$ and $\boldsymbol{s} = \left(s_1, \dots, s_o\right) = \beta_{\mathrm{S}}^{-1}(v)$, assuming uniform signs, the demapper calculates the metric

$$\begin{aligned} P_{V|\boldsymbol{Y}}(v|\boldsymbol{y}_j^{\mathrm{S}}) \propto P_{V\boldsymbol{Y}}(v, \boldsymbol{y}_j^{\mathrm{S}}) &= P_{\boldsymbol{SY}}(\beta_{\mathrm{S}}^{-1}(v), \boldsymbol{y}_j^{\mathrm{S}}) \\ &= \prod_{l=1}^{o} P_{SY}(s_l, y_{jl}) \\ &= \prod_{l=1}^{o} \sum_{\substack{x\in\mathcal{X}: \\ \mathrm{sign}(x)=s_l}} P_{XY}(x, y_{jl}) \end{aligned}$$

$$= \prod_{l=1}^{o} \frac{1}{2} \sum_{a \in \mathcal{A}} P_{Y|X}(y_{jl}|a \cdot s_l) P_A(a). \tag{5.23}$$

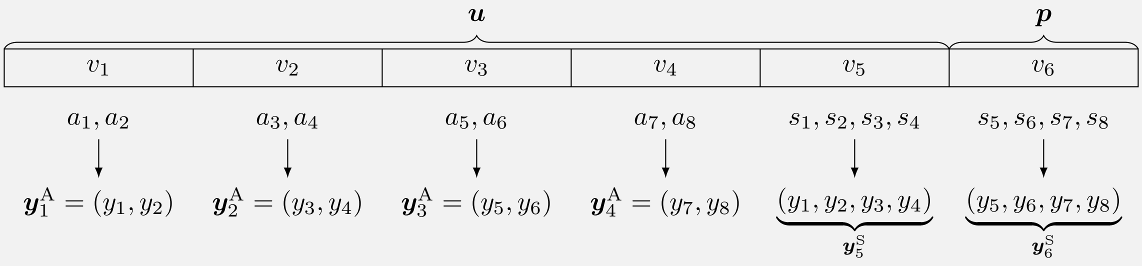
Example 16. We illustrate the setting for 8-ASK, i.e., $m = 3$, and a rate $R_\mathrm{c} = 2/3$ code over $\mathbb{F}_{16}$ ($o = 4$) with blocklength $n_\mathrm{c} = 3$. Using these parameters, we have $n = (n_\mathrm{c} \cdot o)/m = 4$ channel uses. Each of the two symbols in the information part (v_1, v_2) represents $\lambda = o/(m-1) = 4/(3-1) = 2$ amplitudes. The last codeword symbol forms the parity part and is mapped to four sign bits.



$\boldsymbol{u}$
$\boldsymbol{p}$
v_1
v_2
v_3
a_1, a_2
a_3, a_4
s_1, s_2, s_3, s_4
$\boldsymbol{y}_1^\mathrm{A} = (y_1, y_2)$
$\boldsymbol{y}_2^\mathrm{A} = (y_3, y_4)$
$\boldsymbol{y}_3^\mathrm{S} = (y_1, y_2, y_3, y_4)$

As for the binary case, PAS can also be operated with NB codes of rates larger than $(m-1)/m$. In this case, $(\gamma n)/o$ information symbols are used as signs, see (3.30). This means (5.22) must be applied only for the first n/λ variables nodes (nodes associated with amplitude mappings) and the remaining n/o variable nodes (nodes associated with sign mappings) are initialized with (5.23).

Example 17. We illustrate this setting for $m = 3$ and a rate $R_\mathrm{c} = 5/6$ code over $\mathbb{F}_{16}$ ($o = 4$) with blocklength $n_\mathrm{c} = 6$. Using these parameters, we have a number of $n = (n_\mathrm{c} \cdot o)/m = 8$ channel uses. Four of the five codeword symbols in the information part (v_1, v_2, v_3, v_4) represent amplitudes. The last information symbol (v_5) as well as the parity symbol (v_6) originate from sign mappings.



$\boldsymbol{u}$
$\boldsymbol{p}$
v_1
v_2
v_3
v_4
v_5
v_6
a_1, a_2
a_3, a_4
a_5, a_6
a_7, a_8
s_1, s_2, s_3, s_4
s_5, s_6, s_7, s_8
$\boldsymbol{y}_1^\mathrm{A} = (y_1, y_2)$
$\boldsymbol{y}_2^\mathrm{A} = (y_3, y_4)$
$\boldsymbol{y}_3^\mathrm{A} = (y_5, y_6)$
$\boldsymbol{y}_4^\mathrm{A} = (y_7, y_8)$
$\underbrace{(y_1, y_2, y_3, y_4)}_{\boldsymbol{y}_5^\mathrm{S}}$
$\underbrace{(y_5, y_6, y_7, y_8)}_{\boldsymbol{y}_6^\mathrm{S}}$

5.3. Bit-Metric Decoding of Non-Binary LDPC Codes

5.3.1. Bit-Metric Decoding for Uniform Constellations

We now describe how a NB-LDPC code can be operated with BMD and uniform signaling. The blockwise application of (5.20) maps a length $k_\mathrm{c} \cdot o$ vector of uniformly distributed bits to k_c symbols of $\mathbb{F}_{2^o}$. This sequence is encoded into a length n_c symbols codeword $\boldsymbol{v}$ with binary representation $\boldsymbol{v}_\mathrm{bin} = (v_{\mathrm{bin},1}, v_{\mathrm{bin},2}, \ldots, v_{\mathrm{bin},n_\mathrm{c}\cdot o})$. We assume $n_\mathrm{c} \cdot o = m \cdot n$ for simplicity in the following. Eventually, the modulator maps blocks of m bits to one 2^m-ASK symbol

$$x_i = \chi^{-1}(v_{\mathrm{bin},(i-1)\cdot m+1}, \ldots, v_{\mathrm{bin},i\cdot m}), \quad i = 1, \ldots, n.$$

At the receiver side, the received sequence is demodulated by calculating the entries of the soft information vector $\boldsymbol{l}_\mathrm{dec} = (l_{\mathrm{dec},1}, l_{\mathrm{dec},2}, \ldots, l_{\mathrm{dec},m\cdot n})$ where

$$l_{\mathrm{dec},(i-1)m+k} = \log\left(\frac{P_{B_k|Y}(0|y_i)}{P_{B_k|Y}(1|y_i)}\right) \tag{5.24}$$

for $i = 1, \ldots, n$ and $k = 1, \ldots, m$. The distribution $P_{B_k|Y}$ was derived in (3.9). The input (5.21) to the NB-LDPC decoder is calculated for $\boldsymbol{m}_{\mathrm{dec},j} = (m_{\mathrm{dec},j}(0), \ldots, m_{\mathrm{dec},j}(\alpha^{q-2}))$ as

$$m_{\mathrm{dec},j}(c) = \frac{\tilde{P}_j(c)}{\sum_{c'\in\mathbb{F}_q} \tilde{P}_j(c')} \quad \text{with} \quad \tilde{P}_j(c) = \prod_{l=1}^{o} \tilde{P}_{jl}, c \in \mathbb{F}_q \tag{5.25}$$

for $j = 1, \ldots, n_\mathrm{c}$ and $l = 1, \ldots, o$, where

$$\tilde{P}_{j,l} = \begin{cases} \frac{\exp(l_{\mathrm{dec},(j-1)\cdot o+l})}{1+\exp(l_{\mathrm{dec},(j-1)\cdot o+l})}, & \text{if } [\beta_{\mathbb{F}_q}^{-1}(c)]_l = 0, \\ \frac{1}{1+\exp(l_{\mathrm{dec},(j-1)\cdot o+l})}, & \text{if } [\beta_{\mathbb{F}_q}^{-1}(c)]_l = 1. \end{cases} \tag{5.26}$$

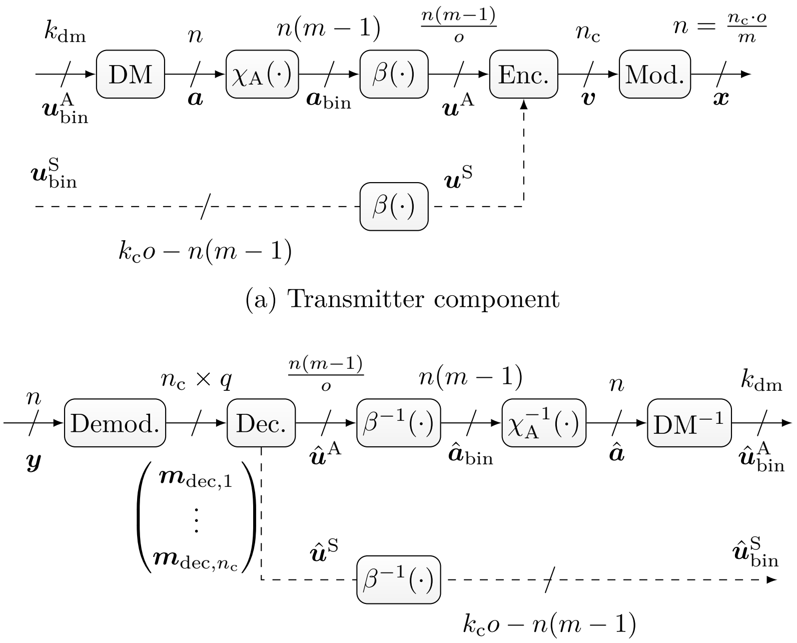
Of course, an interleaver can be included in the setup above, e.g., for fading channels.

5.3.2. Bit-Metric Decoding for PAS

The same principle as shown in Sec. 5.3.1 can also be applied to PAS and is shown in Fig. 5.3. A number k_dm of uniformly distributed information bits are matched to n amplitudes following a specified distribution. Using the amplitude mapping χ_A (3.27) the amplitudes are mapped to a length $n \cdot (m-1)$ bit string, mapped to $\mathbb{F}_{2^o}$ symbols and encoded into the codeword $\boldsymbol{v}$. A modulator then maps the binary image of $\boldsymbol{v}$ to channel inputs $x \in \mathcal{X}$ via a consecutive application of χ^{-1}, while taking the position of amplitude and sign bits into account.

At the receiver side, the demapper calculates a soft information vector as shown in (5.24), (5.25) and (5.26) for the uniform scenario.

Example 18. Consider a length $n_\mathrm{c} = 3$, rate $R_\mathrm{c} = 2/3$ code over $\mathbb{F}_{32}$ ($o = 5$), while using an 8-ASK constellation ($m = 3$) such that the channel is used $n = (n_\mathrm{c} \cdot o)/m = 5$ times with constellation symbols x_1, x_2, x_3, x_4, x_5. The length m binary label of the i-th channel symbol is referred to as $b_{i,1} \ldots b_{i,m}$. That is, for the given scenario, we


k_{dm}
n
$n(m-1)$
$\frac{n(m-1)}{o}$
n_{c}
$n = \frac{n_{\mathrm{c}} \cdot o}{m}$
DM
$\chi_{\mathrm{A}}(\cdot)$
$\beta(\cdot)$
Enc.
Mod.
$\boldsymbol{u}_{\mathrm{bin}}^{\mathrm{A}}$
$\boldsymbol{a}$
$\boldsymbol{a}_{\mathrm{bin}}$
$\boldsymbol{u}^{\mathrm{A}}$
$\boldsymbol{v}$
$\boldsymbol{x}$
$\boldsymbol{u}_{\mathrm{bin}}^{\mathrm{S}}$
$\boldsymbol{u}^{\mathrm{S}}$
$\beta(\cdot)$
$k_{\mathrm{c}}o - n(m-1)$

(a) Transmitter component

n
$n_{\mathrm{c}} \times q$
$\frac{n(m-1)}{o}$
$n(m-1)$
n
k_{dm}
Demod.
Dec.
$\beta^{-1}(\cdot)$
$\chi_{\mathrm{A}}^{-1}(\cdot)$
DM^{-1}
$\boldsymbol{y}$
$\hat{\boldsymbol{u}}^{\mathrm{A}}$
$\hat{\boldsymbol{a}}_{\mathrm{bin}}$
$\hat{\boldsymbol{a}}$
$\hat{\boldsymbol{u}}_{\mathrm{bin}}^{\mathrm{A}}$
$\begin{pmatrix} \boldsymbol{m}_{\mathrm{dec},1} \\ \vdots \\ \boldsymbol{m}_{\mathrm{dec},n_{\mathrm{c}}} \end{pmatrix}$
$\hat{\boldsymbol{u}}^{\mathrm{S}}$
$\beta^{-1}(\cdot)$
$\hat{\boldsymbol{u}}_{\mathrm{bin}}^{\mathrm{S}}$
$k_{\mathrm{c}}o - n(m-1)$

(b) Receiver component

Figure 5.3.: Operating a rate $R_{\mathrm{c}} = k_{\mathrm{c}}/n_{\mathrm{c}}$ NB-LDPC code with PAS and BMD. The dashed lines are needed for code rates $R_{\mathrm{c}} > (m-1)/m$. The functions $\chi_{\mathrm{A}}(\cdot)$ and $\beta(\cdot)$ are applied to each amplitude in the vector $\boldsymbol{a}$ and chunks of o consecutive bits in $\boldsymbol{a}_{\mathrm{bin}}$, respectively.

have $\chi(x_i) = b_{i,1}b_{i,2}b_{i,3}$. Conventional PAS with NB codes and SMD (see Sec. 5.2) is not possible for these parameters, as $o = 5$ is not an integer multiple of $m-1 = 2$. After encoding, the binary image of the codeword $\boldsymbol{v} = (v_1, v_2, v_3)$ is

$$\boldsymbol{v}_{\mathrm{bin}} = (\underbrace{b_{1,2}b_{1,3}b_{2,2}b_{2,3}b_{3,2}}_{\beta_{\mathrm{S}}^{-1}(c_1)}, \underbrace{b_{3,3}b_{4,2}b_{4,3}b_{5,2}b_{5,3}}_{\beta_{\mathrm{S}}^{-1}(v_2)}, \underbrace{b_{1,1}b_{2,1}b_{3,1}b_{4,1}b_{5,1}}_{\beta_{\mathrm{S}}^{-1}(v_3)}).$$

The binary image of the parity symbol $v_3 \in \mathbb{F}_{32}$, i.e., $\beta_{\mathrm{S}}^{-1}(v_3)$, provides the signs for the five channel uses and the soft information vector reads

$$\boldsymbol{l}_{\mathrm{dec}} = (l_{1,2}l_{1,3}l_{2,2}l_{2,3}l_{3,2}l_{3,3}l_{4,2}l_{4,3}l_{5,2}l_{5,3}l_{1,1}l_{2,1}l_{3,1}l_{4,1}l_{5,1})\,. \tag{5.27}$$

Eventually, the vector $\boldsymbol{l}_{\mathrm{dec}}$ is combined as shown in (5.25) and (5.26) to form the decoder a-priori soft-information.

	8-ASK SE = 1.5 bpcu		16-ASK SE = 3 bpcu	
R_c Mode	1/2 uni.	3/4 PAS	3/4 uni.	5/6 PAS
R_BMD^{-1} [dB]	9.44	8.48	19.25	18.11
R_SMD^{-1} [dB]	9.00	8.46	19.17	18.10
$\mathbb{F}_{64}$, BMD [dB]	9.93	8.90	–	–
$\mathbb{F}_{64}$, SMD [dB]	9.53	8.92	–	–
$\mathbb{F}_{256}$, BMD [dB]	9.91	8.93	19.79	18.54
$\mathbb{F}_{256}$, SMD [dB]	–	8.93	19.85	–

Table 5.1.: MCDE thresholds and required asymptotic SNR values in dB.

5.4. Asymptotic Decoding Thresholds

We investigate the asymptotic decoding thresholds for SMD and BMD with NB-LDPC codes. As shown in [186], DE for NB-LDPC codes can exploit symmetry of the involved messages as well as the all-zero codeword assumption. Despite these properties, DE still turns out to be challenging for NB-LDPC codes, as it requires to track a density for each field element. To circumvent this difficulty, we resort to Monte Carlo Density Evolution (MCDE) which is explained in detail in Appendix A.6.

The results of the MCDE analysis are shown in Table 5.1. For the $R_\mathrm{tx} = 1.5$ bpcu case, we observe a gap of 0.4 dB between uniform signaling with SMD and BMD, while there is no significant loss for the shaped case. Further, we note that going from $\mathbb{F}_{64}$ to $\mathbb{F}_{256}$ does not give significant performance gains with respect to the decoding threshold.

For $R_\mathrm{tx} = 3.0$ bpcu, the gap between SMD and BMD in the uniform case decreases to 0.06 dB as predicted by the achievable rate analysis. The gain of about 1.1 dB for shaped signaling is well reflected in the decoding thresholds as well.

5.5. Finite Length Simulation Results

In this section, we compare both approaches by means of finite length simulations for 8-ASK with $R_\mathrm{tx} = 1.5$ bpcu ($n = 192$) and 16-ASK with $R_\mathrm{tx} = 3.0$ bpcu ($n = 288$). The DM parameters for the 8-ASK scenario and $n = 192$ are the same as in Table 3.9. For the 16-ASK case, we summarize the parameters in Table 5.2.

The NB-LDPC codes were constructed from protographs of the form

$$[\underbrace{2 \quad 2 \quad \dots \quad 2}_{d_c/2}]$$

via cyclic liftings and a PEG-like algorithm [154]. All constructed matrices have girth 8. The coefficients were optimized according to the approach of Sec. 5.1.3. We performed a maximum of 200 BP iterations for decoding.

Parameter	Value
k_{dm}	768
n	288
R_{dm}	2.667
$\boldsymbol{t}_{\mathcal{A}}^{n}$	$\{64, 60, 51, 41, 30, 21, 13, 8\}$
P_A	$(0.2222, 0.2083, 0.1771, 0.1424, 0.1042, 0.0729, 0.0451, 0.0278)$
R_{loss}	0.0903 bits

(a) CCDM

Parameter	Value
k_{dm}	768
n	288
R_{dm}	2.667
$W(a)$	a^2
P_A	$(0.2413, 0.2205, 0.1841, 0.1403, 0.0976, 0.0618, 0.0357, 0.0187)$
R_{loss}	0.011 bits

(b) SMDM

Table 5.2.: DM parameters for the 16-ASK, $R_{\mathrm{tx}} = 3.0$ bpcu setup.

In Fig. 5.4, we show the performance for 8-ASK and a target SE of 1.5 bpcu. We consider codes over $\mathbb{F}_{64}$ and $\mathbb{F}_{256}$. The PAS setting uses a rate $R_{\mathrm{c}} = 3/4$ code ($d_{\mathtt{v}} = 2, d_{\mathtt{c}} = 8$), while uniform signaling employs a rate $R_{\mathrm{c}} = 1/2$ code ($d_{\mathtt{v}} = 2, d_{\mathtt{c}} = 4$) As suggested from the decoding thresholds in Table 5.1, the codes over both fields perform very similar to each other. For uniform signaling, we see a significant gain of 0.47 dB of SMD over BMD. We note that the plot does not include any SMD results for the $\mathbb{F}_{256}$ code and uniform signaling, as such an operation is not possible ($m = 3$ is not an integer multiple of $o = 8$), see Sec. 5.2. The PAS results use SMDM and achieve a gain of 0.72 dB over uniform signaling with SMD. Further, SMD and BMD practically coincide in the shaped case as could be expected from Table 5.1. We complement the NB results with the binary ones from Fig. 3.32 in which the 5G LDPC codes are used. We see that the NB codes clearly outperform the binary codes for low FERs. Further, to put the results into perspective, we also include the SPB and RCUB from Sec. 2.3.7, where the latter is evaluated for SMD and the distribution realized by SMDM (see Table 3.9). At an FER of 10^{-4}, we operate 0.93 dB from the SPB and 0.3 dB from the RCUB.

In Fig. 5.5, we show the average number of iterations until convergence (i.e., until the syndrome is zero) for all of the considered signaling options of Fig. 5.4. For high SNRs, NB codes usually require only two to three iterations. We also observe that the shaped modes require less iterations for the same FER.

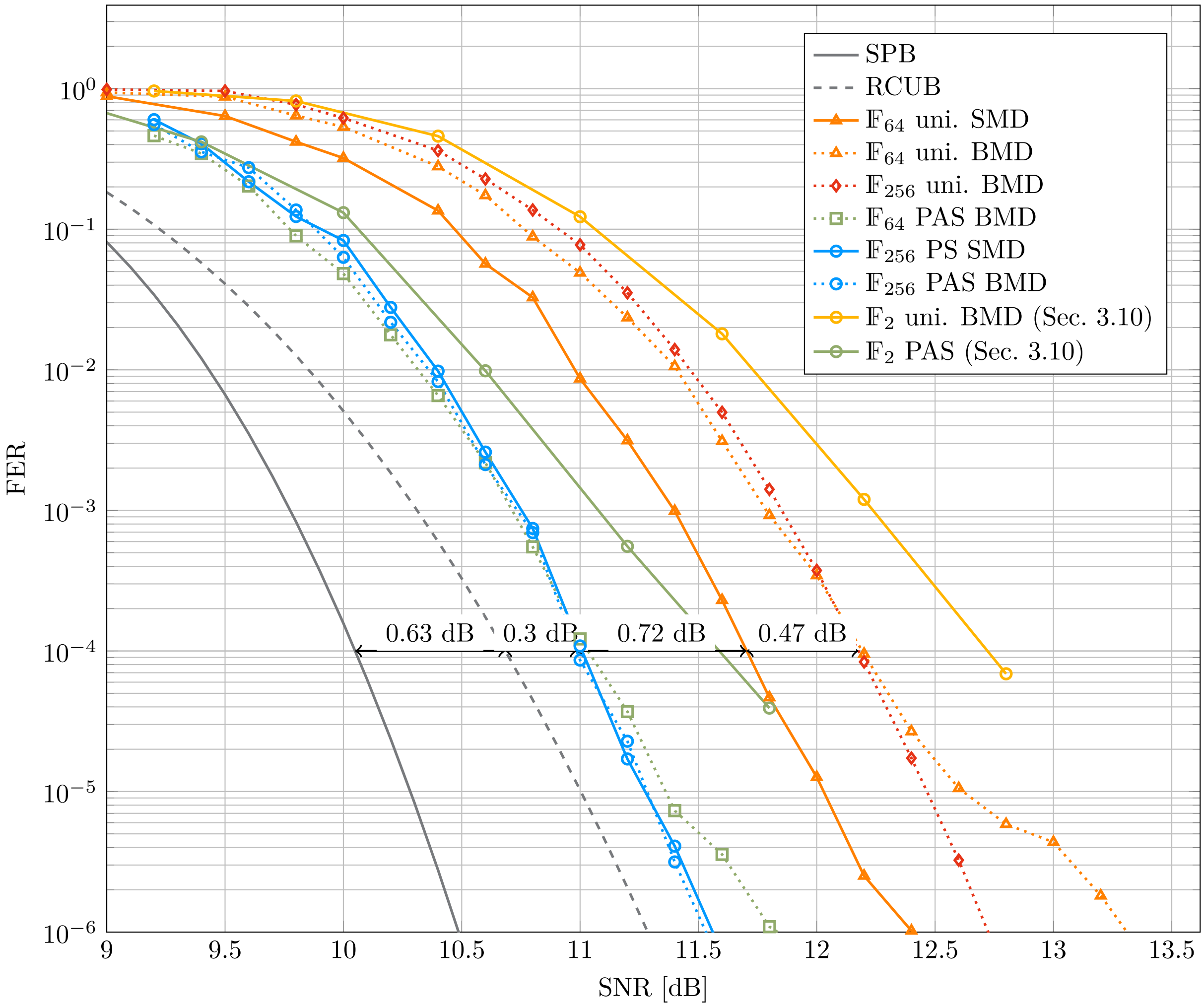


Figure 5.4.: Coded performance of NB-LDPC codes for an SE of 1.5 bpcu and $n = 192$.

In Fig. 5.6 we show the performance for 16-ASK and a target SE of 3.0 bpcu. The PAS setting uses BMD with a rate $R_\mathsf{c} = 5/6$ code ($d_\mathsf{v} = 2, d_\mathsf{c} = 12$) over $\mathbb{F}_{256}$. SMD with the setup of Sec. 5.2 is not possible here, as the extension order (8) is not an integer multiple of the number of amplitude bits (3). Uniform signaling employs a rate $R_\mathsf{c} = 3/4$ code ($d_\mathsf{v} = 2, d_\mathsf{c} = 8$) over $\mathbb{F}_{256}$, while both SMD and BMD is used.

We observe a significant performance gain of 0.53 dB of SMDM over CCDM, which matches the one predicted by comparing the respective rate loss values $((0.0903 - 0.011) \cdot 6\,\text{dB} \approx 0.48\,\text{dB})$ of Table 5.2. Having the results of Table 3.9 in mind (which even considers a smaller output blocklength of $n = 192$ channel uses, but only predicts an improvement by 0.19 dB) this may look surprising, but it is due to the larger 8-ary output alphabet of the DMs. The rate loss also depends on the cardinality of the output alphabet.

The gain over uniform signaling is 1.25 dB at an FER of 10^{-4} Further, we see that the

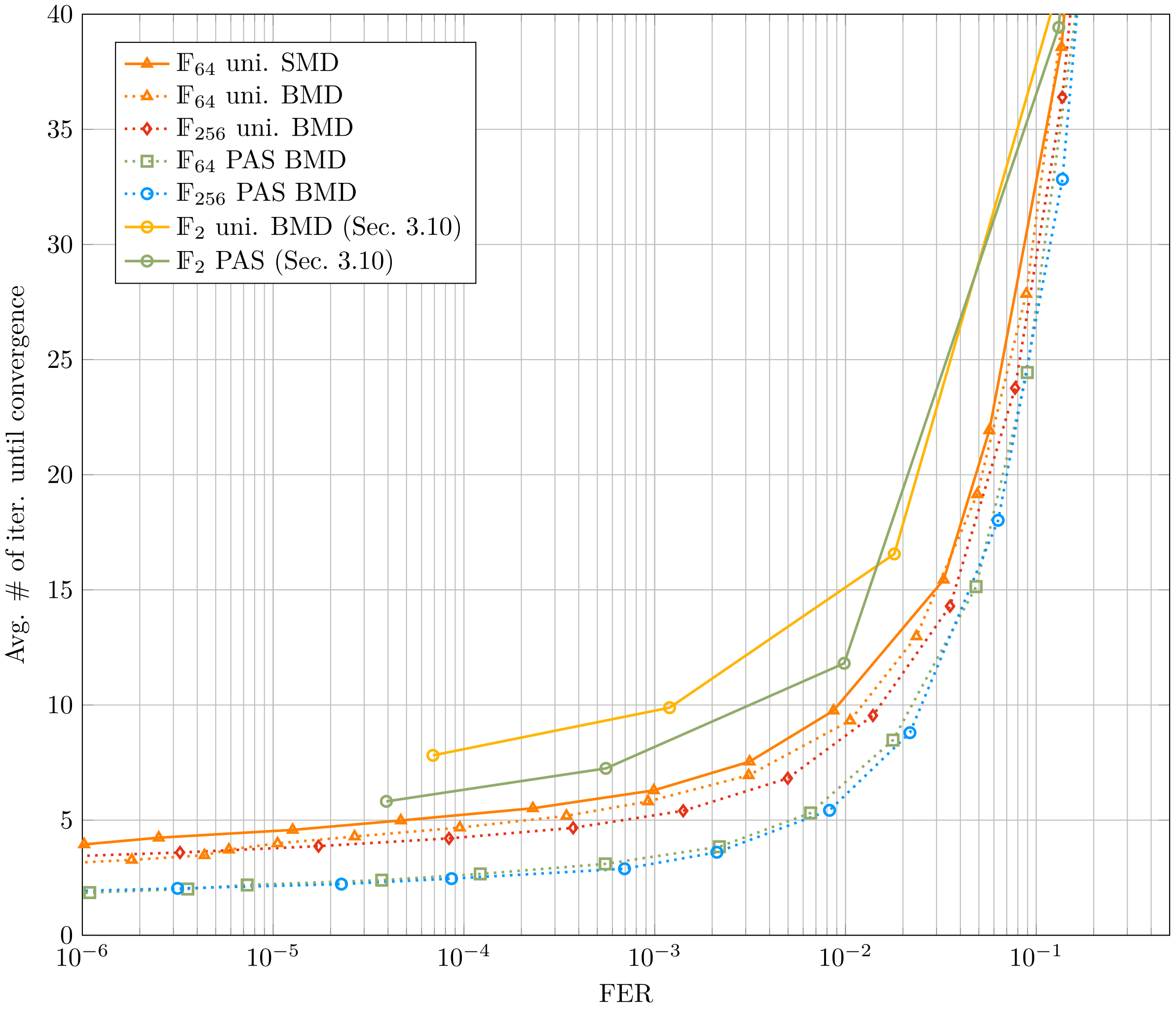


Figure 5.5.: Comparison of the number of required decoding iterations for the setup of Fig. 5.4.

performance of the uniform setup is very similar for SMD and BMD. At an FER of 10^{-4}, the gap to the SPB is 0.91 dB and 0.4 dB to the RCUB. As before, we evaluate the latter with the SMDM output distribution of Table 5.2b.

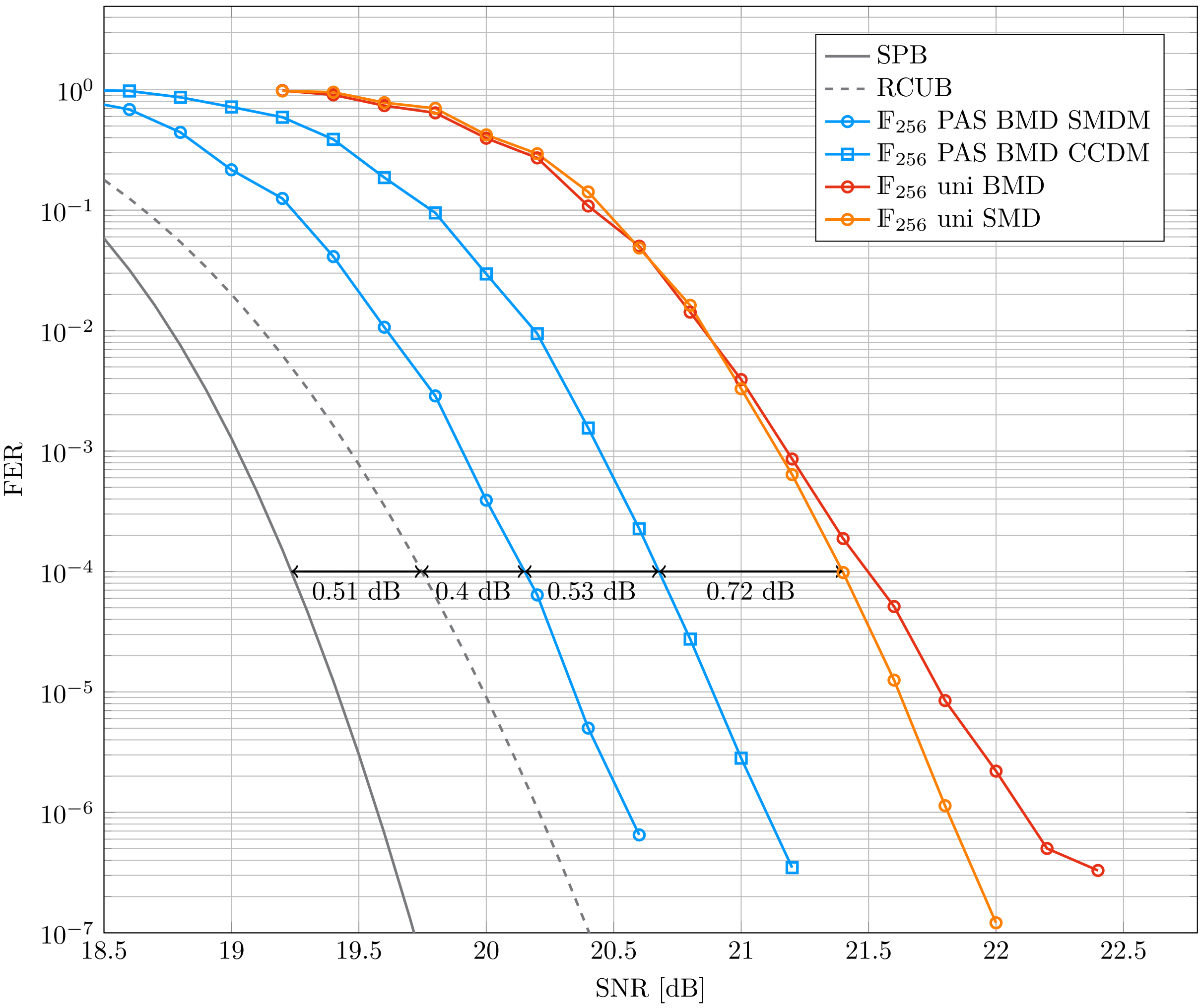


Figure 5.6.: Coded performance of NB-LDPC codes for an SE of 3.0 bpcu and $n = 288$.

6 Applications to Optical Communications

To fully characterize the performance of optical transceivers, one must conduct transmission experiments. In the following, we consider the DSP (e.g., sampling, chromatic dispersion compensation, equalization, phase noise compensation) and the optical channel as a black box which provides the discrete time receive samples.

For a given experimental setup (e.g., input power, transmission distance, DSP and modulation setting), we transmit a test sequence x^n and measure a noisy observation y^n. The n entries $x_1 \dots x_n$ correspond to n real-valued ASK symbols. In case of 2D modulation, two successive entries of x^n correspond to the in-phase and quadrature components of a QAM symbol; if in addition polarization division multiplexing is applied, four successive entries of x^n correspond to the in-phase and quadrature components of the two polarizations.

6.1. Blind Estimation of Shaping Parameters

PAS gained much interest in the field of optical communications to increase the SE and flexibility of transceivers. Several metrics have been suggested to characterize the performance limits of a coded modulation system, e.g., GMI [187], normalized generalized mutual information (NGMI) [188, 189] and cross entropy based (mismatched) uncertainty [63], see also Sec. 3.3. The essential component of these metrics is a simple and tractable model of the optical communication channel as seen by the FEC decoder.

A pragmatic and empirically accurate model to describe the accumulated noise at the receiver after transmission over a long, dispersion-uncompensated fiber is an AWGN channel [190, 191]. As usual, its parameters (e.g., noise variance, gain) must be estimated at the receiver. For PS, the receiver also needs to know the distribution of the transmit symbols to achieve the best performance [192]. Usually, these parameters are either known or obtained by data-aided (DA) ML estimation using both the transmitted and received data [13, Sec. III-B]. In general, the latter approach should be used with caution, as it may overestimate the achievable rates.

In this section, we propose an unsupervised learning approach based on expectation maximization (EM) that estimates all model parameters for a decoding metric suitable for PAS and SD-FEC schemes in a blind fashion. The solution uses only the channel outputs and learns all relevant parameters on the fly. This is particularly important for PAS which is inherently rate flexible and a separate signaling of the modulation parameters is undesirable. We validate the approach by using recorded data from transmission experiments [13] and compare to DA estimation.

6.1.1. Maximum-Likelihood Estimation

We first consider ML estimation to find the parameters $\boldsymbol{\theta} \in \mathbb{R}^n$ of a model that "explains" the observations[1] $y_i, i = 1, \dots, N$. The model is commonly given by a PDF $p_{Y^N}(\cdot; \boldsymbol{\theta})$ or PMF $P_{Y^N}(\cdot; \boldsymbol{\theta})$. We assume a memoryless model for the observations y^N such that

$$p_{Y^N}(y^N; \boldsymbol{\theta}) = \prod_{i=1}^{N} p_{Y_i}(y_i; \boldsymbol{\theta}). \tag{6.1}$$

The ML problem is

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{i=1}^{N} p_{Y_i}(y_i; \boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \underbrace{\sum_{i=1}^{N} \ln\left(p_{Y_i}(y_i; \boldsymbol{\theta})\right)}_{L(\boldsymbol{\theta})}. \tag{6.2}$$

We refer to the factors in (6.2) as *likelihoods*, and the summands as *log-likelihoods*. The function $L(\boldsymbol{\theta})$ is called a log-likelihood function and it follows by introducing the logarithm. Taking logarithms is useful when dealing with exponential models [193] that often appear in practice.

We assume that the transmitter sends a sequence of N *pilot symbols* x_i, $i = 1, \dots, N$, that the receiver knows, and the AWGN model (3.1) becomes

$$Y_i = \Delta x_i + N_i, \quad i = 1, \dots, N \tag{6.3}$$

where the N_i are independent and identically distributed as $\mathcal{N}(0, \sigma^2)$. We collect the model parameters $\theta_1 = \Delta$ and $\theta_2 = \sigma^2$ into the vector $\boldsymbol{\theta} = (\theta_1, \theta_2)$ and use (6.2) (recognizing that $Y_i \sim \mathcal{N}(\Delta x_i, \sigma^2)$) to obtain

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{N} \ln\left(p_{Y_i}(y_i; \boldsymbol{\theta})\right) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \underbrace{\sum_{i=1}^{N} -\frac{1}{2}\ln(2\pi\theta_2) - \frac{(y_i - \theta_1 x_i)^2}{2\theta_2}}_{L(\boldsymbol{\theta})}. \tag{6.4}$$

[1] In the context of supervised/unsupervised learning, the model observations are also called *labels*.

Taking derivatives and equating to zero, we have

$$\nabla L(\boldsymbol{\theta}_{\mathrm{ML}}) = \begin{pmatrix} \sum_{i=1}^{N} \frac{(y_i - \theta_{\mathrm{ML},1} x_i) x_i}{\theta_{\mathrm{ML},2}} \\ \sum_{i=1}^{N} \frac{(y_i - \theta_{\mathrm{ML},1} x_i)^2}{2\theta_{\mathrm{ML},2}^2} - \frac{N}{2\theta_{\mathrm{ML},2}} \end{pmatrix} = \mathbf{0} \tag{6.5}$$

so that

$$\hat{\theta}_{\mathrm{ML},1} = \hat{\Delta} = \frac{\sum_{i=1}^{N} y_i x_i}{\sum_{i=1}^{N} x_i^2} \tag{6.6}$$

$$\hat{\theta}_{\mathrm{ML},2} = \hat{\sigma}^2 = \frac{1}{N} \sum_{i=1}^{N} (y_i - x_i \hat{\theta}_1)^2. \tag{6.7}$$

These are the ML estimates because the problem is convex, as can be checked via the Hessian. The receiver uses these parameter estimates for detection and decoding, e.g., for calculating soft information for the FEC decoder (4.18).

6.1.2. Expectation Maximization

Introduction

We recall the definition of the Kullback-Leibler divergence (2.46) and use the *log-sum identity* [30, Sec. 1.9.1]. Consider positive a_k and non-negative b_k for $k = 1, \ldots, K$, and suppose that at least one of the b_k is positive. Let $S_a = \sum_{k=1}^{K} a_k$ and $S_b = \sum_{k=1}^{K} b_k$, and define $P_A(k) = a_k / S_a$ and $P_B(k) = b_k / S_b$ for $k = 1, \ldots, K$. We have

$$\sum_{k=1}^{K} a_k \log\left(\frac{a_k}{b_k}\right) = S_a \log\left(\frac{S_a}{S_b}\right) + S_a \, \mathrm{D}(P_A || P_B). \tag{6.8}$$

Now suppose that we wish to perform the same task as before but *without* sending pilots. Instead, we know only that the X_i are taken from a discrete and finite set $\mathcal{X}$ and have the respective distributions P_{X_i}, $i = 1, \ldots, N$. The receiver has access only to the labels $y_i, i = 1, \ldots, N$ and the ML rule is

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{N} \ln\left(p_{Y_i}(y_i; \boldsymbol{\theta})\right) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \underbrace{\sum_{i=1}^{N} \ln\left(\sum_{x \in \mathcal{X}} p_{Y_i X_i}(y_i, x; \boldsymbol{\theta})\right)}_{L(\boldsymbol{\theta})}. \tag{6.9}$$

Compared to (6.4), solving for the stationary points of (6.9) is challenging and no closed form solution can be given for most cases because of the marginalizing over the *latent* or *hidden* variables X_i. If we knew the latent variables X_i, we could group the observations based on the originally transmitted constellation point, and apply ML estimation with the pilots. To simplify the problem and come up with a solution of (6.9), the idea of EM is the following: Break the ML estimation in two parts, namely:

1. First, calculate a *"soft" assignment* to the latent variables.

2. Second, perform the desired *parameter optimization.*

The EM algorithm was described by Dempster, Laird and Rubin [194] in 1977.

Evidence Lower Bound (ELBO)

Consider the log-likelihood function from (6.9):

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{N} \ln\left(\sum_{x\in\mathcal{X}} p_{Y_i X_i}(y_i, x; \boldsymbol{\theta})\right) \tag{6.10}$$

where $p_{Y_i X_i}(y, x; \boldsymbol{\theta}) = P_{X_i}(x; \boldsymbol{\theta}) p_{Y|X}(y|x; \boldsymbol{\theta})$ and where we abuse notation by writing $p_{Y_i X_i}(\cdot; \boldsymbol{\theta})$ as a density. We artificially augment $L(\boldsymbol{\theta})$ by N auxiliary probability distributions $Q_{X_i|Y_i}(\cdot|y_i)$, $i = 1, \ldots, N$ and exploit that $\sum_{x\in\mathcal{X}} Q_{X_i|Y_i}(x|y_i) = 1$:

$$\begin{aligned} L(\boldsymbol{\theta}) &= \sum_{i=1}^{N} \left(\sum_{x\in\mathcal{X}} Q_{X_i|Y_i}(x|y_i)\right) \ln\left(\frac{\sum_{x\in\mathcal{X}} p_{Y_i X_i}(y_i, x; \boldsymbol{\theta})}{\sum_{x\in\mathcal{X}} Q_{X_i|Y_i}(x|y_i)}\right) \\ &= \sum_{i=1}^{N} \left[\sum_{x\in\mathcal{X}} Q_{X_i|Y_i}(x|y_i) \ln\left(\frac{p_{Y_i X_i}(y_i, x; \boldsymbol{\theta})}{Q_{X_i|Y_i}(x|y_i)}\right)\right] + \mathrm{D}\left(Q_{X_i|Y_i}(\cdot|y_i) || P_{X_i|Y_i}(\cdot|y_i; \boldsymbol{\theta})\right) \end{aligned} \tag{6.11}$$

where we have applied the log-sum identity (6.8) with

$$a_x \coloneqq Q_{X_i|Y_i}(x|y_i) \text{ and } b_x \coloneqq p_{Y_i X_i}(y_i, x; \boldsymbol{\theta}) \tag{6.12}$$

so that $S_a = 1$ and $S_b = p_Y(y_i; \boldsymbol{\theta})$. We thus have

$$L(\boldsymbol{\theta}) \geq E(\boldsymbol{\theta}) \coloneqq \sum_{i=1}^{N} \sum_{x\in\mathcal{X}} Q_{X_i|Y_i}(x|y_i) \ln\left(\frac{p_{Y_i X_i}(y_i, x; \boldsymbol{\theta})}{Q_{X_i|Y_i}(x|y_i)}\right) \tag{6.13}$$

where $E(\boldsymbol{\theta})$ is referred to as the *evidence lower bound (ELBO)*. Moreover, equality holds in (6.13) if and only if

$$Q_{X_i|Y_i}(x|y_i) = P_{X_i|Y_i}(x|y_i; \boldsymbol{\theta}), \quad \forall x \in \mathcal{X}, \quad \forall i = 1, \ldots, N. \tag{6.14}$$

The ELBO is the fundamental building block of the EM algorithm summarized in Algorithm 9.

We make several remarks.

- ▷ Various approaches can be used to check convergence. For instance, the algorithm can be stopped after a certain number of iterations if the value $\|\boldsymbol{\theta}^{(t)} - \boldsymbol{\theta}^{(t-1)}\|$ is small, or if $L(\boldsymbol{\theta}^{(t)}) - L(\boldsymbol{\theta}^{(t-1)})$ is small.

Algorithm 9 Expectation Maximization

```
t = 1
Initialize $\boldsymbol{\theta}^{(1)}$ with a good starting value.
while convergence criteria is not met do
    E-step: Compute
        $Q^{(t)}_{X_i|Y_i}(x|y_i) = P_{X_i|Y_i}\left(x|y_i;\boldsymbol{\theta}^{(t)}\right), \quad \forall x \in \mathcal{X}, \quad \forall i = 1,\dots,N$
    M-step: Compute
        $\boldsymbol{\theta}^{(t+1)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \quad \sum_{i=1}^{N}\sum_{x\in\mathcal{X}} Q^{(t)}_{X_i|Y_i}(x|y_i)\ln\left(p_{Y_iX_i}\left(y_i,x;\boldsymbol{\theta}\right)\right)$
    t = t + 1
end while
```

- ▷ Initializing the EM algorithm with different starting values $\boldsymbol{\theta}^{(1)}$ and choosing the outcome with the highest objective function value may improve performance.

- ▷ The name *E-step* (for “expectation step”) is not self-explanatory in the above formulation. The terminology comes from [194] where the *E-step* is written as

$$Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(t)}) = \sum_{i=1}^{N} \mathrm{E}_{Q^{(t)}_{X_i|Y_i}(\cdot|y_i)}\left[\ln(p_{YX}(y_i,X;\boldsymbol{\theta}))\right] \tag{6.15}$$

where $Q^{(t)}_{X_i|Y_i}(x|y_i) = P_{X|Y}(x|y_i;\boldsymbol{\theta}^{(t)})$, and the *M-step* as

$$\boldsymbol{\theta}^{(t+1)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(t)}). \tag{6.16}$$

We prefer the presentation in Algorithm 9 as the required computational steps, i.e., computing the $Q^{(t)}_{X_i|Y_i}(\cdot|y_i)$ and the maxima, are specified separately.

6.1.3. K-Means

An algorithm closely related to the EM algorithm is K-Means, a name coined by MacQueen [195] in 1967, though the concept was originally formulated by Steinhaus [196] in 1957. While at Bell Labs, Llyod also formulated K-Means to find an optimal quantizer for pulse code modulation (PCM), but the approach was not published until 1982[2].

We begin with the K-means problem formulation and then relate it to our approach in the previous section. As a *clustering algorithm*, K-Means aims at solving the following problem: Given a set of N points $\boldsymbol{y}_i \in \mathbb{R}^n$, $i = 1,\dots,N$, we want to find K centers $\boldsymbol{x}_j \in \mathbb{R}^n, j = 1,\dots,K$ and corresponding assignments $\delta_{ji} \in \{0,1\}$ such that the distance

[2] `https://en.wikipedia.org/wiki/K-means_clustering#History`

of each point to its (representative) center is minimized. We have

$$\min_{\substack{\boldsymbol{x}_j,\, j=1,\ldots,K \\ \delta_{ji},\, j=1,\ldots,K,\, i=1,\ldots,N}} \sum_{i=1}^{N}\sum_{j=1}^{K} \delta_{ji} \left\|\boldsymbol{x}_j - \boldsymbol{y}_i\right\|^2 . \tag{6.17}$$

Because of the discrete nature of the assignment variables δ_{ji}, solving (6.17) directly with convex optimization techniques is not possible. However, we can pursue a two step approach that first finds the optimal centers and then updates the assignment:

1. (*E-step* replaced by *Decision-step*) Given the cluster centers $\boldsymbol{x}_j, j = 1, \ldots, K$, the optimal assignment δ_{j^*i} for the i-th point $\boldsymbol{y}_i$ is found via

$$j^* = \operatorname*{argmin}_{j\in\{1,2,\ldots,K\}} \left\|\boldsymbol{x}_j - \boldsymbol{y}_i\right\|^2 . \tag{6.18}$$

2. (*M-step*) Given an assignment δ_{ij}, we can find the optimal centers by solving for a stationary point $\boldsymbol{x}_j$ as

$$\frac{\partial}{\partial \boldsymbol{x}_j} \sum_{i=1}^{N}\sum_{j=1}^{K} \delta_{ji} \left\|\boldsymbol{x}_j - \delta_{ji}\boldsymbol{y}_i\right\|^2 = \sum_{i=1}^{N} 2\boldsymbol{x}_j\delta_{ji} - 2\delta_{ji}\boldsymbol{y}_i = 0$$

such that

$$\boldsymbol{x}_j = \frac{\sum_{i=1}^{N} \delta_{ji}\boldsymbol{y}_i}{\sum_{i=1}^{N} \delta_{ji}}. \tag{6.19}$$

6.1.4. Numerical Results

We consider the model (3.1) with circularly symmetric, complex Gaussian noise with zero mean and variance σ^2. We use the developed EM algorithm to estimate the signaling parameters for an optical transmission experiment with PAS. The model parameters are specified by the parameter vector $\boldsymbol{\theta} = (\Delta, \sigma^2, \boldsymbol{p})$ for a general P_X or by $\boldsymbol{\theta} = (\Delta, \sigma^2, \nu)$ for an MB distribution P_X (3.21), respectively. The entries of the vector $\boldsymbol{p} = (p_1, p_2, \ldots, p_M)$ are $p_i = P_X(x_i)$.

The stationary points of the optimization in the M-step of Algorithm 9 are given as

$$\Delta_{\text{opt}}^{\text{EM}} = \frac{\sum_{i=1}^{N}\sum_{x\in\mathcal{X}} Q_{X_i}^{(t)}(x)\Re(y_i x^*)}{\sum_{i=1}^{n}\sum_{x\in\mathcal{X}} Q_{X_i}^{(t)}(x)\left|x\right|^2} \tag{6.20}$$

$$\sigma_{\text{opt}}^{2,\text{EM}} = \frac{1}{N}\sum_{i=1}^{n}\sum_{x\in\mathcal{X}} Q_{X_i}^{(t)}(x)\left|y_i - \Delta_{\text{opt}}^{\text{EM}} x\right|^2 \tag{6.21}$$

$$p_{j,\text{opt}}^{\text{EM}} = \frac{1}{N}\sum_{i=1}^{n} Q_{X_i}^{(t)}(x_j) \tag{6.22}$$

where $\boldsymbol{p}_{\text{opt}}^{\text{EM}} = (p_{1,\text{opt}}^{\text{EM}}, \ldots, p_{M,\text{opt}}^{\text{EM}})$. If we assume an MB distribution on $\mathcal{X}$, the value for ν_{opt}

is the solution of the non-linear equation

$$\frac{1}{n}\sum_{i=1}^{N} \mathrm{E}_{X\sim Q_{X_i}^{(t)}}\left[X^2\right] = \mathrm{E}_{X\sim P_X^{\nu_{\text{opt}}}}\left[X^2\right]. \tag{6.23}$$

Remark 3. The EM estimation can be considered in the framework of decoding metrics as follows: First, note that blind estimation calculates the model parameters from the observation y^n, i.e., $\Delta(y^n)$, $\sigma^2(y^n)$, $P_X(y^n)$. Second, note that once the parameters are estimated, a memoryless metric is used. Together, we have

$$q(x^n, y^n) = \prod_{i=1}^{n} q(x_i, y_i; \Delta(y^n), \sigma^2(y^n), P_X(y^n)) \tag{6.24}$$

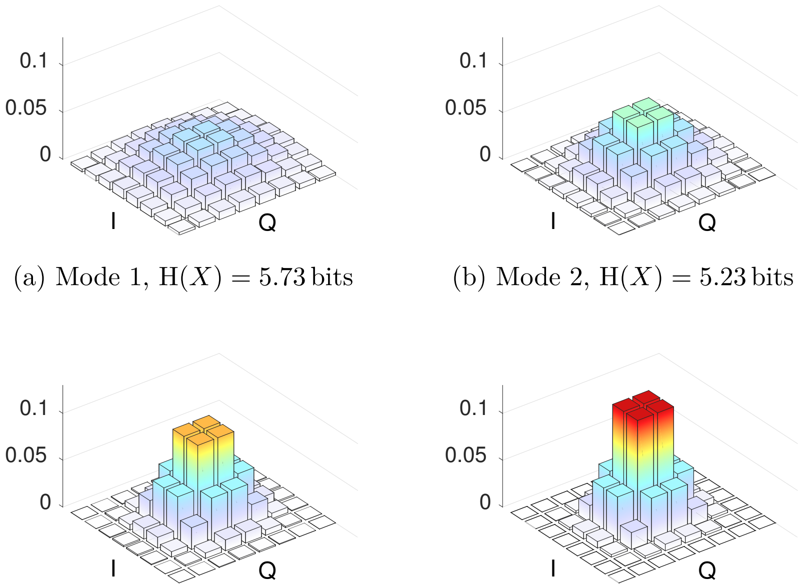
that is, blind parameter estimation is included in the general framework of decoding metrics $q(x^n, y^n)$. In particular, the derivations of Secs. 3.3.1 and 3.3.2 hold and no potential rate overestimation as in the case of DA estimation can occur.

We evaluate the DA and EM approaches by comparing their achievable rate estimates. For this, we use the obtained parameters and calculate the decoder soft information (4.18) l_{ik} for all $i = 1, \ldots, N$ samples and $k = 1, \ldots, m$ bit levels to evaluate (3.13) empirically (see also (2.53) and (2.54)). The achievable rate estimate is

$$\hat{R}_{\text{BMD}} = \left[\mathrm{H}(X) - \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{m} \log_2\left(1 + \mathrm{e}^{-(1-2b_{ik})l_{ik}}\right)\right]^{+}. \tag{6.25}$$

As shown in [194], EM does not necessarily converge to the globally optimal solution of (6.9), but usually only to a local optimum. To guarantee convergence to good parameter values, the EM algorithm needs to be initialized carefully. For this, we choose the initial parameter $\boldsymbol{\theta}^{(0)}$ by a modified version of K-Means [197], which uses only the channel outputs and includes a constraint on the equi-spaced constellation points. The number of initial clusters for K-Means is an important parameter, as not all of the M constellation points might have been transmitted. We circumvent this problem by initializing K-Means with a smaller number of clusters. For instance, for 64-QAM, we run the EM algorithm with initializations obtained from K-Means with the number of clusters set to $\{4, 16, 36, 64\}$ and choose the parameter set that yields the largest $\hat{R}_{\text{BMD}}$. This corresponds to the common practice of initializing EM with different random starting points and using the set of parameters maximizing the objective (6.9).

To assess the performance of both schemes, we use the recorded sequences of one of our previous transmission experiments [13], in which four shaping modes with different entropies were investigated. The PMFs are depicted in Fig. 6.1. The length of each sequence was $N \approx 20\,000$ QAM symbols. Mode 4 effectively corresponds to a 36-QAM constellation. The results are shown in Fig. 6.2. We observe that both the DA and EM approaches achieve the same achievable rates, showing that EM can accurately estimate the parameters for all considered modes. EM converged in less than 30 iterations.



(a) Mode 1, $\mathrm{H}(X) = 5.73\,\mathrm{bits}$

(b) Mode 2, $\mathrm{H}(X) = 5.23\,\mathrm{bits}$

(c) Mode 3, $\mathrm{H}(X) = 4.60\,\mathrm{bits}$

(d) Mode 4, $\mathrm{H}(X) = 4.13\,\mathrm{bits}$

Figure 6.1.: Employed distributions for evaluation of the EM approach.

6.2. Extension of Probabilistic Amplitude Shaping to Higher Dimensions

In the previous chapters PAS was used only for one and two dimensional constellations. For optical communications, higher dimensional constellations are of interest, too, as polarization division multiplexing allows to modulate the inphase and quadrature components of both polarizations with, e.g., one four dimensional (4D) symbol. The combination of 4D shaping with forward error correction (FEC) is discussed in [198]. The main conclusion of [198] is that when BMD is used, conventional QAM modulations perform better than 4D shaping. This is partially attributed to the absence of good binary labels for shaped 4D constellations. In this section, we discuss the concept of *quadrant shaping (QS)* [199] which extends PAS to higher dimensions. We also show that finding good binary labels is possible with the proposed approach as a BRGC is used overall.

6.2.1. Quadrant Shaping

We consider the AWGN model

$$\boldsymbol{Y} = \Delta \boldsymbol{X} + \boldsymbol{N} \tag{6.26}$$

where the channel inputs $\boldsymbol{X} \in \mathcal{X} \subset \mathbb{R}^4$ are taken from the 4-fold Cartesian product of an M-ASK constellation. The signaling set thus has cardinality $|\mathcal{X}| = M^4$. As a binary labeling, we use the BRGC for M-ASK in each dimension, i.e., we assign a length

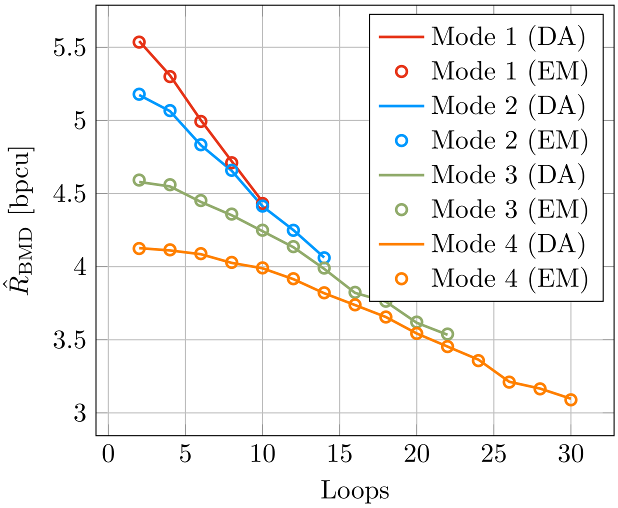
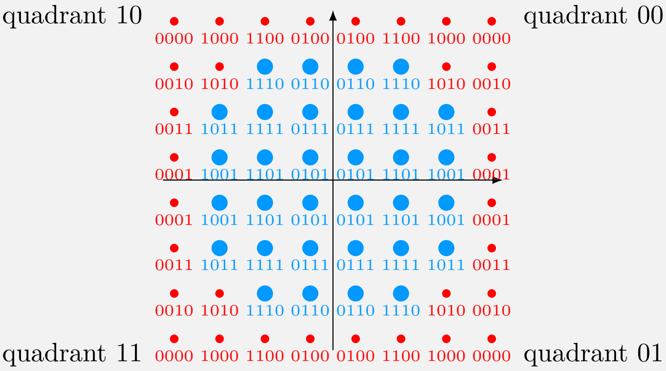


Figure 6.2.: Achievable rates obtained via DA and blind (EM) parameter estimation. The sequences are taken from the transmission experiment in [13]. One loop corresponds to 240 km.

$m = 4\log_2(M)$ binary label for each $\boldsymbol{x} \in \mathcal{X}$ via $\chi : \mathcal{X} \to \{0,1\}^m$ and $\chi(\boldsymbol{x}) = b_1 b_2 \dots b_m = \boldsymbol{b}$. The noise $\boldsymbol{N}$ is a multivariate Gaussian RV with stochastically independent entries that have variance σ^2. The PDF is given in (A.19). The channel output is as $\boldsymbol{Y} \in \mathbb{R}^4$.

To generate the desired distribution on the constellation symbols, PAS uses a DM to create non-uniformly distributed amplitudes and it uses the approximately uniformly distributed parity bits of a FEC code to generate the corresponding signs. This idea can be extended to four (and higher) dimensions. The 1D amplitudes of the original PAS become 4D constellation points in the positive quadrant $\mathcal{Q} = \{\boldsymbol{x} \in \mathcal{X} | x_i \geq 0, i = 1,2,3,4\}$ with $|\mathcal{Q}| = 2^{m-4}$. Four sign bits choose the 4D symbol's quadrant.

Example 19. We show an example for QS in two dimensions in the figure below. The DM can be implemented as a LUT with eight entries.



$\boldsymbol{u}_{\text{dm}}$	quadrant bits
000	1001
001	1011
010	1101
011	1111
100	1110
101	0101
110	0111
111	0110

6.2.2. Experimental Investigation

A key enabler for PAS is the DM and most work so far considered CCDM (see Sec. 3.4.2), which has excellent performance but introduces additional complexity because of the arithmetic coding. For low-complexity applications, we may also use a simple LUT as DM and select low power sequences from a (4D) hypercube consisting of the 4-fold Cartesian product of M-ary ASK constellations. The combination with FEC is then realized via quadrant shaping.

We show experimentally that rate adaptation with steps of 0.5 bits/QAM symbol (bpQs) can be achieved with the scheme of the previous section and a simple LUT. Further, we compare this method to conventional PAS with CCDM and uniform constellations.

System Setup

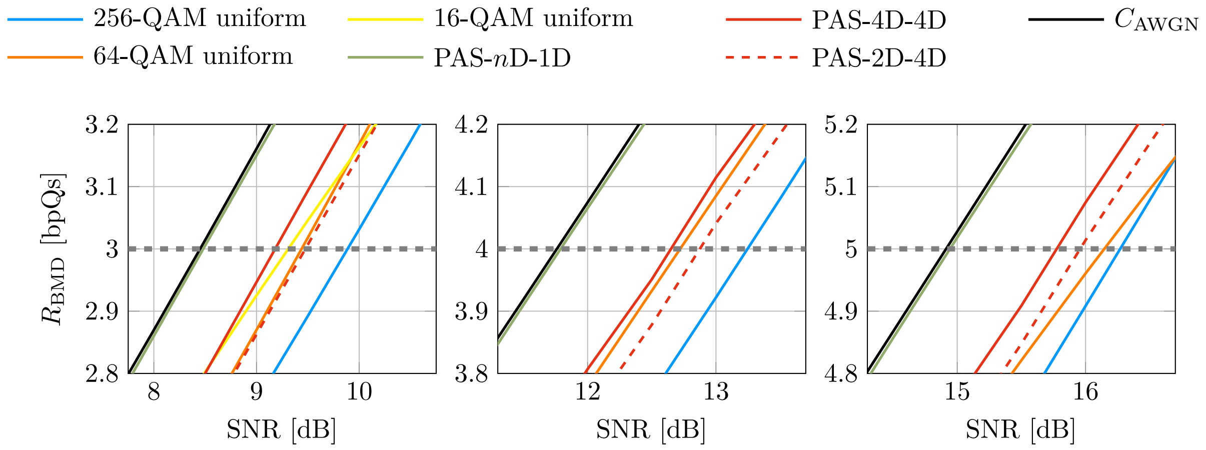
We distinguish three signaling and receiver schemes. PAS-nD-1D is the conventional setting using a n-dimensional (nD) DM and 1D demapping. PAS-4D-4D uses a 4D DM and also demaps in 4D. A variant of the latter is PAS-4D-2D, where a sub-optimal and less complex demapping in 2D is used. The PAS-4D schemes employ 4D signaling with an M-ASK constellation in each dimension. For the binary labeling of the constellation points, we use an $m = \log_2(M^4)$ bit BRGC. Following the principle of QS, $m_\mathrm{Q} = m - 4$ bits determine a point in a quadrant and $m_\mathrm{S} = 4$ bits represent the 4 sign bits. A rate adaptation can be realized by a DM based on a LUT with at most 2^{m_Q} entries, that selects a subset of 2^{k_dm} low-energy points from the 2^{m_Q} points in a quadrant. The corresponding DM rate is $R_\mathrm{dm} = k_\mathrm{dm}$ bits/4D-symbol. We refer to the resulting constellation as $\mathcal{X}$. The SE is $R_\mathrm{tx} = R_\mathrm{dm} + 4 \cdot \gamma$ bits/4D-symbol, where γ is the fraction of sign bits per dimension that carry additional information bits. For the FEC code rate R_c we have $\gamma = 1 - (1 - R_\mathrm{c}) \cdot \log_2(M)$. By choosing k, a rate adaptation with a granularity of 1 bit/4D-symbol, i.e., 0.5 bpQs is possible so that any SE within the set $\{0.5, 1.0, 1.5, \ldots, 2\log_2(M) - 2\} + 2 \cdot \gamma$ bpQs can be realized with the *same* FEC overhead (OH). If the scheme is extended to N-dimensions (ND), a granularity of $1/(N/2)$ bpQs is possible. The exemplary transmission modes of this work target SEs of 3 bpQs, 4 bpQs and 5 bpQs and are summarized in Table 6.1. We assume a *single* FEC code with OH = 23% ($R_\mathrm{c} = 13/16$).

The calculated achievable rates for BMD and the *linear AWGN channel* are shown in Fig. 6.3. We observe that the PAS-4D modes are superior to their uniform $\{16, 64, 256\}$-QAM counterparts for all three target SEs. The loss in power efficiency due to 2D demapping is at most 0.2 dB for BMD. The PAS-nD-1D modes virtually achieve Gaussian capacity.

We experimentally investigate the 4D signaling scheme for a short-reach scenario and unrepeated transmission for 100 km, 140 km and 180 km, where the four dimensions are transmitted as two complex dimensions in two subsequent time slots. The CCDM for PAS-nD-1D operates in $n = 6000$ dimensions and the input distributions are taken from the MB family. The setup is shown in Fig. 6.4. The data symbols are interleaved with quadrature

Signaling mode	R_{tx}[bpQs]	$\lvert\mathcal{X}\rvert$	R_{dm}[bpQs]	OH
PAS-4D-4D-4.5	5.0	8182	4.5	23%
PAS-4D-4D-3.5	4.0	2048	3.5	23%
PAS-4D-4D-2.5	3.0	512	2.5	23%
PAS-nD-1D-4.5	5.0	256	4.5	23%
PAS-nD-1D-3.5	4.0	256	3.5	23%
PAS-nD-1D-2.5	3.0	256	2.5	23%
16-QAM uniform	3.0	16	–	33%
64-QAM uniform	4.0	64	–	50%
256-QAM uniform	5.0	256	–	60%

Table 6.1.: Investigated signaling modes for the experiment.



(a) $R_{\mathrm{tx}} = 3\,\mathrm{bpQs}$ (b) $R_{\mathrm{tx}} = 4\,\mathrm{bpQs}$ (c) $R_{\mathrm{tx}} = 5\,\mathrm{bpQs}$

Figure 6.3.: Simulated achievable rates of several signaling strategies for the linear AWGN channel.

phase-shift keying (QPSK) pilots at a pilot rate of 10%. A frame alignment sequence is added at the beginning of the sequence and square root raised cosine (RRC) pulse shaping with a roll-off factor of 0.1 is applied [200]. We employ a wave-division multiplexing (WDM) setup with 5 external cavity lasers (ECLs, 10 kHz linewidth) on a 25 GHz grid. An arbitrary waveform generator (AWG, 20 GHz) drives the two IQ modulators. The four interferers (IQ mod 1) are combined with the central channel (IQ mod 2), a delay-and-add polarization emulator generates a dual-polarization signal and the channels are individually decorrelated. The transmission link consists of an erbium doped fiber amplifier (EDFA) followed by a variable optical attenuator (VOA) that sets the total power launched into the standard single-mode fiber (SSMF) of lengths 100 km, 140 km and 180 km. After transmission, the central channel is demodulated using a standard preamplified coherent receiver followed by a digital storage oscilloscope (DSO, 80 GSa/s and 33 GHz analog bandwidth). The receiver DSP is performed offline [200].

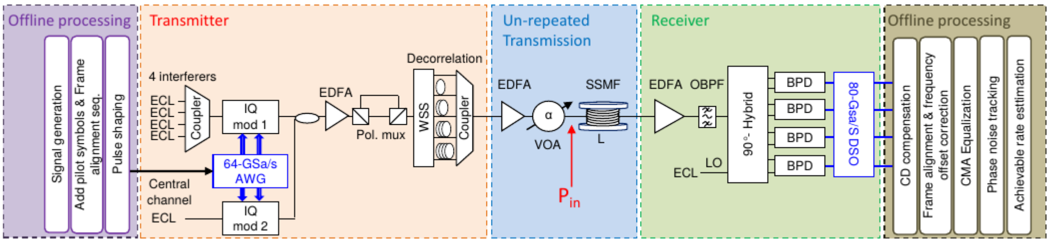
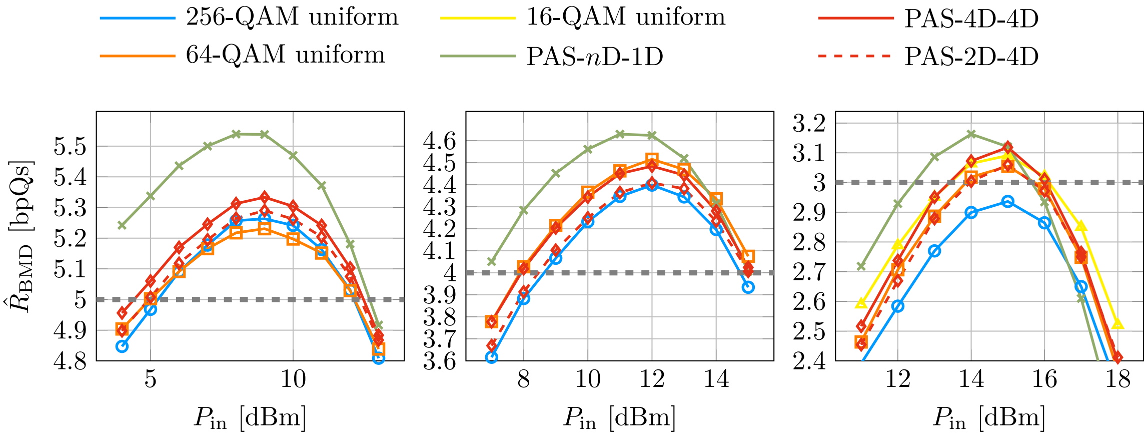


Figure 6.4.: Experimental setup of the optical experiment.

In back-to-back experiments, we validated that all modulation formats exhibit the same implementation penalty in the operating regime of interest to ensure a fair comparison. The achievable rates for the transmission experiments for link lengths of 100 km, 140 km and 180 km are shown in Fig. 6.5. To operate efficiently for all three distances (corresponding to optimal launch powers of 9 dBm, 12 dBm and 15 dBm) at the desired SEs (dashed gray line), the three different uniform constellations (16-, 64- and 256-QAM) would need to be operated with three different FEC codes (see Table 6.1). The PAS modes allow a flexible operation with a single FEC. For increased transmission lengths, and therefore launch powers P_{in}, we observe that the PAS-nD-1D modes are penalized due to their increased higher order moments and their severe impact on the non-linear interference noise (NLIN) [201, 200]. At the respective optimal launch powers, the shaping gain of PAS-nD-1D is 0.2 bpQs, 0.15 bpQs, and 0.04 bpQs as compared to PAS-4D-4D. However, the PAS-4D DM is a simple LUT with at most $2^{k_{\text{dm}}} = 2^9 = 512$ entries. Moreover, 2D demapping (PAS-4D-2D) loses at most 0.1 bpQs for a target SE of 4 bpQs.



(a) 100 km (b) 140 km (c) 180 km

Figure 6.5.: Achievable rates of the considered signaling strategies for the transmission experiment.

The results show that 4D signal shaping and rate adaptation with a simple LUT based DM allows a good trade-off between shaping gain and DM complexity. The suggested schemes are promising for transmission links where full fledged PAS is too complex or penalized because of NLIN. Future work should investigate the trade-off between higher dimensional constellations with $N > 4$, rate flexibility, and demapping complexity.

7 Conclusions and Outlook

In this thesis we investigated PS for higher-order modulation and developed tailored code constructions for binary and non-binary LDPC codes. The results were presented by means of information theoretic quantities (achievable rates and finite length bounds) as well as simulations using various FEC codes and decoders, which validated the developed theory and stressed its importance for system design. As an outlook, this thesis gives rise to questions of theoretical and practical relevance that deserve attention and require further study.

For PS we considered OOK as one example where the capacity achieving input distribution is not symmetric and PAS can not be applied. The class of channels and modulation constraints that require such an asymmetric constellation also comprises the case of unipolar PAM constellations, which play an important role for transceivers based on IM and DD, e.g., for low cost data center applications. While shaping via TS shows significant gains over uniform signaling, there is still a non-negligible gap motivating a new scheme that closes this gap. For polar codes, the scheme by Honda and Yamamoto [202] provides such a solution. In addition, studying peak-power constraints originating, e.g., from limited extinction ratios of Mach-Zehnder modulators, is interesting from an information theoretic perspective. Further, PS was found to be beneficial for multi user information theory scenarios such as dirty paper coding [203] and investigating related setups for wiretap channels is likely to yield new insights.

In the LDPC code design part, we used the decoding threshold as our primary design target. However, the decoding threshold is a quantity that is *defined asymptotically*, while we aim at designing *good practical finite length codes*. This raises the question whether codes can be designed specifically for the desired target blocklength. Machine learning approaches [204] or genetic algorithm formulations [205, 206] may provide good frameworks for this task. Similarly, DE assumes an infinite number of iterations, while practical decoders often perform only 5 to 10 iterations or even less, e.g., for window decoding [172]. Recent work [207] has shown that simply adjusting the number of DE iterations may not yield accurate results. Another interesting aspect is the investigation of error floors for different bit mappings for the same parity-check matrix. Numerical evidence suggests that certain mappings yield a higher undetected error rate than others. Further, the code design in Sec. 4.5 did not take the need for efficient encoding into consideration. It is interesting to integrate this requirement as well.

For high throughput decoding applications, quantized LDPC decoders will become increasingly important. Our studies concentrated on their performance and investigated losses compared to the unquantized case. Future work should take the effects of a limited number of iterations and error floors into account.

For NB-LDPC codes, the design of codes over intermediate field sizes, e.g., $\mathbb{F}_{32}$, is interesting to find a trade-off between improved performance at short block lengths (compared to binary LDPC codes) and decoding complexity. As for the binary case, one needs to find a good optimization metric for finite length code design that is sufficiently easy to evaluate. We remark that our numerical investigations showed significant deviations between the real decoding thresholds and those obtained by existing P-EXIT approaches [208] for NB-LDPC codes.

A

Appendix

A.1. Derivation of Achievable Rates

In Sec. 2.3.3, we define the achievable rate for a certain random coding experiment as a rate R for which the random coding exponent $E(R)$ is positive. Hence, we need to analyze the function

$$E(R) = \max_{0 \leq \rho \leq 1} (E_0(\rho) - \rho R)$$

This is an optimization problem with affine inequality constraints, for which we can use the Karush-Kuhn-Tucker (KKT) conditions [209] to obtain insights. The Lagrangian function is

$$L(\rho, \mu, \lambda) = E_0(\rho) - \rho R + \mu \rho + \lambda(1 - \rho).$$

Primal feasibility (PF):

- ▷ $\rho \geq 0$
- ▷ $\rho \leq 1$

Dual feasibility (DF):

- ▷ $E_0'(\rho) - R + \mu - \lambda = 0$
- ▷ $\mu \geq 0$
- ▷ $\lambda \geq 0$

Complementary slackness (CS):

- ▷ $\mu\rho = 0$
- ▷ $\lambda(1-\rho) = 0$

Depending on the Lagrangian variables and the CS constraints we can analyze different cases:

1. Case 1: $\mu \neq 0, \lambda \neq 0$.

   This assumption leads to a contradiction. $\mu \neq 0$ implies $\rho = 0$, but at the same time $\lambda \neq 0 \Rightarrow \rho = 1$, which is not possible.

2. Case 2: $\mu \neq 0, \lambda = 0$. $\mu \neq 0$ implies $\rho = 0$ and $E_0'(0) - R + \mu = 0$, i.e., for $R \geq E_0'(0)$.

3. Case 3: $\mu = 0, \lambda \neq 0$. $\lambda \neq 0$ implies $\rho = 1$ and $E_0'(1) - R - \lambda = 0$, i.e., for $R \leq E_0'(1)$.

4. Case 4: $\mu = 0, \lambda = 0$. We have $0 < \rho < 1$ and $E_0'(\rho) - R = 0$.

From the four cases above, we can draw the following conclusions:

- ▷ For $0 \leq R < E_0'(1) : \rho = 1$ and $E(R) = E_0(1) - R$ is linear with slope -1.
- ▷ For $E_0'(1) \leq R \leq E_0'(0) : 0 < \rho < 1$.
- ▷ For $R > E_0'(0)$: $\rho = 0$, $E(R) = E_0(0) = 0$.

The value of $E_0'(1)$ is commonly referred to as critical rate R_{crit}. Correspondingly, an achievable rate is given by

$$R = \left.\frac{\partial E_0(\rho)}{\partial \rho}\right|_{\rho=0}. \tag{A.1}$$

A.2. Derivation of Achievable Rates for Hard-Decision Decoding

In (3.79) and (3.80) we derived the mismatched uncertainty expressions

$$\mathrm{U}\left(q_{\text{SMD}}^{\text{HD}}\right) = \min_{s\geq 0} \mathrm{E}\left[-\log_2\left(\frac{\varepsilon^{s\cdot\mathbb{1}\left(X\neq Q_{\text{HD}}^{\text{SW}}(Y)\right)}}{\sum_{a\in\mathcal{X}}\varepsilon^{s\cdot\mathbb{1}\left(a\neq Q_{\text{HD}}^{\text{SW}}(Y)\right)}}\right)\right]$$

$$\mathrm{U}\left(q_{\text{BMD}}^{\text{HD}}\right) = \min_{s\geq 0} \mathrm{E}\left[-\log_2\left(\frac{\varepsilon^{s\cdot\sum_{k=1}^{m}\mathbb{1}\left([\chi(X)]_k\neq Q_{\text{HD}}^{\text{BW},k}(Y)\right)}}{\sum_{a\in\mathcal{X}}\varepsilon^{s\cdot\sum_{k=1}^{m}\mathbb{1}\left([\chi(a)]_k\neq Q_{\text{HD}}^{\text{BW},k}(Y)\right)}}\right)\right].$$

Starting with the SMD case, we have

$$\begin{aligned}
&\mathrm{E}\left[-\log_2\left(\frac{\epsilon^{s\cdot\mathbb{1}\left(X\neq Q_{\mathrm{HD}}^{\mathrm{SW}}(Y)\right)}}{\sum_{a\in\mathcal{X}}\varepsilon^{s\cdot\mathbb{1}\left(a\neq Q_{\mathrm{HD}}^{\mathrm{SW}}(Y)\right)}}\right)\right]\\
&=-\int_{\mathbb{R}}\sum_{x\in\mathcal{X}}p_{Y|X}(y|x)P_X(x)\log_2\left(\frac{\varepsilon^{s\cdot\mathbb{1}\left(x\neq Q_{\mathrm{HD}}^{\mathrm{SW}}(y)\right)}}{\sum_{a\in\mathcal{X}}\varepsilon^{s\cdot\mathbb{1}\left(a\neq Q_{\mathrm{HD}}^{\mathrm{SW}}(y)\right)}}\right)\mathrm{d}y\\
&=-\sum_{\hat{x}\in\mathcal{X}}\sum_{x\in\mathcal{X}}P_{\hat{X}|X}(\hat{x}|x)P_X(x)\log_2\left(\frac{\epsilon^{s\cdot\mathbb{1}(x\neq\hat{x})}}{\sum_{a\in\mathcal{X}}\epsilon^{s\cdot\mathbb{1}(a\neq\hat{x})}}\right) \qquad \text{(A.2)}
\end{aligned}$$

where we introduced

$$P_{\hat{X}|X}(\hat{x}|x)=\int_{y\in\mathcal{R}_{\hat{x}}}p_{Y|X}(y|x)\,\mathrm{d}y,\quad \hat{x}\in\mathcal{X}. \qquad \text{(A.3)}$$

Let us now investigate the term within the logarithm of (A.2) for which we have

$$\frac{\epsilon^{s\cdot\mathbb{1}(x\neq\hat{x})}}{\sum_{a\in\mathcal{X}}\epsilon^{s\cdot\mathbb{1}(a\neq\hat{x})}}=\begin{cases}\frac{\varepsilon^s}{(M-1)\varepsilon^s+1}, & x\neq\hat{x},\\ \frac{1}{(M-1)\varepsilon^s+1}, & x=\hat{x}.\end{cases} \qquad \text{(A.4)}$$

Therefore, we may write (A.2) as

$$-\sum_{\substack{x,\hat{x}\in\mathcal{X}:\\ \hat{x}=x}}P_{\hat{X}|X}(\hat{x}|x)P_X(x)\log_2\left(\frac{1}{(M-1)\varepsilon^s+1}\right)-\sum_{\substack{x,\hat{x}\in\mathcal{X}:\\ \hat{x}\neq x}}P_{\hat{X}|X}(\hat{x}|x)P_X(x)\log_2\left(\frac{\varepsilon^s}{(M-1)\varepsilon^s+1}\right). \qquad \text{(A.5)}$$

Introducing

$$\begin{aligned}
\delta_{\mathrm{SMD}}&=\sum_{\hat{x}\neq x}P_{\hat{X}|X}(\hat{x}|x)P_X(x) \qquad \text{(A.6)}\\
1-\delta_{\mathrm{SMD}}&=\sum_{\hat{x}=x}P_{\hat{X}|X}(\hat{x}|x)P_X(x) \qquad \text{(A.7)}
\end{aligned}$$

we can reformulate (A.5) as

$$\begin{aligned}
&-(1-\delta_{\mathrm{SMD}})\log_2\left(\frac{1}{(M-1)\varepsilon^s+1}\right)-\delta_{\mathrm{SMD}}\log_2\left(\frac{\varepsilon^s}{(M-1)\varepsilon^s+1}\right) \qquad \text{(A.8)}\\
&-(1-\delta_{\mathrm{SMD}})\log_2\left(\frac{1}{(M-1)\varepsilon^s+1}\right)-(M-1)\frac{\delta_{\mathrm{SMD}}}{M-1}\log_2\left(\frac{\varepsilon^s}{(M-1)\varepsilon^s+1}\right) \qquad \text{(A.9)}
\end{aligned}$$

and we recognize its representation as a cross entropy (2.39). This expression is minimized over s if we choose

$$\frac{\delta_{\mathrm{SMD}}}{M-1} = \frac{\varepsilon^s}{(M-1)\varepsilon^s + 1}$$

according to (2.47). Finally, we get

$$s = \log_\varepsilon \left(\frac{\delta_{\mathrm{SMD}}}{(1-\delta_{\mathrm{SMD}})(M-1)} \right)$$

and

$$\mathrm{U}\left(q_{\mathrm{SMD}}^{\mathrm{HD}}\right) = \mathrm{H}_2(\delta_{\mathrm{SMD}}) + \delta_{\mathrm{SMD}} \log_2(M-1). \tag{A.10}$$

The derivation for BMD follows similar steps:

$$\begin{aligned}
&\mathrm{E}\left[-\log_2\left(\frac{\varepsilon^{s\cdot\sum_{k=1}^m \mathbb{1}\left([\chi(X)]_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(Y)\right)}}{\sum_{a\in\mathcal{X}} \varepsilon^{s\cdot\sum_{k=1}^m \mathbb{1}\left([\chi(x)]_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(Y)\right)}}\right)\right] \\
&= -\int_{\mathbb{R}} \sum_{x\in\mathcal{X}} p_{Y|X}(y|x) P_X(x) \log_2\left(\frac{\varepsilon^{s\cdot\sum_{k=1}^m \mathbb{1}\left([\chi(x)]_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}{\sum_{a\in\mathcal{X}} \varepsilon^{s\cdot\sum_{k=1}^m \mathbb{1}\left([\chi(a)]_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}\right) \mathrm{d}y \\
&= -\int_{\mathbb{R}} \sum_{x\in\mathcal{X}} p_{Y|X}(y|x) P_X(x) \log_2\left(\frac{\prod_{k=1}^m \varepsilon^{s\cdot\mathbb{1}\left([\chi(x)]_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}{\sum_{a\in\mathcal{X}} \prod_{k=1}^m \varepsilon^{s\cdot\mathbb{1}\left([\chi(a)]_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}\right) \mathrm{d}y \\
&= -\int_{\mathbb{R}} \sum_{\boldsymbol{b}\in\{0,1\}^m} p_{Y|\boldsymbol{B}}(y|\boldsymbol{b}) P_{\boldsymbol{B}}(\boldsymbol{b}) \log_2\left(\frac{\prod_{k=1}^m \varepsilon^{s\cdot\mathbb{1}\left(b_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}{\prod_{k=1}^m \sum_{b\in\{0,1\}} \varepsilon^{s\cdot\mathbb{1}\left(b \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}\right) \mathrm{d}y \\
&= -\int_{\mathbb{R}} \sum_{\boldsymbol{b}\in\{0,1\}^m} p_{Y|\boldsymbol{B}}(y|\boldsymbol{b}) P_{\boldsymbol{B}}(\boldsymbol{b}) \left(\sum_{k=1}^m \log_2\left(\frac{\varepsilon^{s\cdot\mathbb{1}\left(b_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}{\sum_{b\in\{0,1\}} \varepsilon^{s\cdot\mathbb{1}\left(b \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}\right)\right) \mathrm{d}y \\
&= -\sum_{k=1}^m \int_{\mathbb{R}} \sum_{b_k\in\{0,1\}} p_{Y|B_k}(y|b_k) P_{B_k}(b_k) \log_2\left(\frac{\varepsilon^{s\cdot\mathbb{1}\left(b_k \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}{\sum_{b\in\{0,1\}} \varepsilon^{s\cdot\mathbb{1}\left(b \neq Q_{\mathrm{HD}}^{\mathrm{BW},k}(y)\right)}}\right) \mathrm{d}y \\
&= -\sum_{k=1}^m \sum_{\hat{b}_k, b_k\in\{0,1\}} P_{\hat{B}_k|B_k}(\hat{b}_k|b_k) P_{B_k}(b_k) \log_2\left(\frac{\varepsilon^{s\cdot\mathbb{1}\left(b_k \neq \hat{b}_k\right)}}{\sum_{b\in\{0,1\}} \varepsilon^{s\cdot\mathbb{1}\left(b \neq \hat{b}_k\right)}}\right).
\end{aligned}$$

In the last step, we introduced

$$P_{\hat{B}_k|B_k}(\hat{b}|b) = \int_{\mathcal{R}_k^{\hat{b}}} p_{Y|B_k}(y|b)\,\mathrm{d}y. \tag{A.11}$$

Again, we investigate the inner term of the logarithm for which we have

$$\frac{\varepsilon^{s\cdot\mathbb{1}\left(b\neq\hat{b}\right)}}{\sum_{b\in\{0,1\}}\varepsilon^{s\cdot\mathbb{1}\left(b\neq\hat{b}\right)}}=\begin{cases}\frac{1}{1+\varepsilon^s}, & b=\hat{b},\\ \frac{\varepsilon^s}{1+\varepsilon^s}, & b\neq\hat{b}.\end{cases}\tag{A.12}$$

The setting in (A.11) is an instance of a general problem where

$$\sum_{k=1}^{m}\mathrm{E}\left[f(X_k)\right]\tag{A.13}$$

should be calculated with $f:\mathbb{R}\to\mathbb{R}$ and where the RVs X_k have possibly different PMFs but share the same support $\mathcal{X}$. It is straightforward to show that

$$\sum_{k=1}^{m}\mathrm{E}\left[f(X_k)\right]=m\,\mathrm{E}\left[f(Z)\right]\tag{A.14}$$

where the PMF of the RV Z is given as

$$P_Z(a)=\frac{1}{m}\sum_{k=1}^{m}P_{X_k}(a).\tag{A.15}$$

The normalization with $1/m$ is required to obtain a valid PMF. Exploiting this insight, we choose

$$P_{\hat{A}A}(\hat{a},a)=\frac{1}{m}\sum_{k=1}^{m}P_{\hat{B}_k|B_k}(\hat{a}|a)P_{B_k}(a)$$

and rewrite (A.11) as

$$\begin{aligned}&-m\sum_{\hat{a},a\in\{0,1\}}P_{\hat{A}A}(\hat{a},a)\log_2\left(\frac{\varepsilon^{s\cdot\mathbb{1}(a\neq\hat{a})}}{\sum_{b\in\{0,1\}}\varepsilon^{s\cdot\mathbb{1}(b\neq\hat{a})}}\right)\\&=-m\left(\sum_{\substack{a,\hat{a}\in\{0,1\}:\\a=\hat{a}}}P_{\hat{A}A}(\hat{a},a)\log_2\left(\frac{1}{1+\varepsilon^s}\right)+\sum_{\substack{a,\hat{a}\in\{0,1\}:\\a\neq\hat{a}}}P_{\hat{A}A}(\hat{a},a)\log_2\left(\frac{\varepsilon^s}{1+\varepsilon^s}\right)\right).\end{aligned}$$

Substituting

$$\delta_{\mathrm{BMD}}=\sum_{\substack{a,\hat{a}\in\{0,1\}:\\a\neq\hat{a}}}P_{\hat{A}A}(\hat{a},a)$$

we have

$$-m\left((1-\delta_{\mathrm{BMD}})\log_2\left(\frac{1}{1+\varepsilon^s}\right)+\delta_{\mathrm{BMD}}\log_2\left(\frac{\varepsilon^s}{1+\varepsilon^s}\right)\right)\tag{A.16}$$

which is minimized over s by

$$\delta_{\mathrm{BMD}} = \frac{\varepsilon^s}{1+\varepsilon^s}$$

according to (2.47) such that

$$s = \log_\varepsilon\left(\frac{\delta_{\mathrm{BMD}}}{1-\delta_{\mathrm{BMD}}}\right).$$

We obtain

$$\mathrm{U}\left(q_{\mathrm{BMD}}^{\mathrm{HD}}\right) = m\,\mathrm{H}_2(\delta_{\mathrm{BMD}}). \tag{A.17}$$

A.3. Gauss-Hermite Quadrature Rule

The K-th order Gauss-Hermite quadrature approximate integrals of the form

$$\int\limits_{-\infty}^{\infty} \mathrm{e}^{-z^2} f(z)\,\mathrm{d}z \approx \sum_{i=1}^{K} w_i f(\xi_i) \tag{A.18}$$

by a weighted sum [210, §3.5(v)]. The samples points $\xi_i, i = 1, \ldots, K$ are the roots of the Hermite polynomial $H_K(x)$ [210, §18.3] and the weights $w_i, i = 1, \ldots, K$ are given by

$$w_i = \frac{2^{K-1} K! \sqrt{\pi}}{K^2 H_{K-1}(\xi_i)^2}.$$

Gauss-Hermite quadratures are useful when a differential entropy or mutual information should be calculated and the channel law is Gaussian. For many scenarios, however, we encounter expressions that have a slightly different form than (A.18). In this case, we use integration by substitution. In the following, we illustrate the procedure by calculating the mutual information $\mathrm{I}(\boldsymbol{X};\boldsymbol{Y})$ when the channel law $p_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{x})$ is a N-dimensional Gaussian RV, i.e.,

$$p_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{x}) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left(-\frac{\|\boldsymbol{y}-\boldsymbol{x}\|^2}{2\sigma^2}\right). \tag{A.19}$$

We have

$$\begin{aligned}
\mathrm{I}(\boldsymbol{X};\boldsymbol{Y}) &= \int\limits_{\mathbb{R}^N} \sum_{\boldsymbol{a}\in\mathcal{X}} p_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{a}) P_{\boldsymbol{X}}(a) \underbrace{\log_2\left(\frac{p_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{a})}{\sum_{\boldsymbol{b}\in\mathcal{X}} p_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{b}) P_{\boldsymbol{X}}(\boldsymbol{b})}\right)}_{f(\boldsymbol{y},\boldsymbol{a})} \mathrm{d}\boldsymbol{y} \\
&= \sum_{\boldsymbol{a}\in\mathcal{X}} P_{\boldsymbol{X}}(\boldsymbol{a}) \int\limits_{\mathbb{R}^N} p_{\boldsymbol{Y}|\boldsymbol{X}}(\boldsymbol{y}|\boldsymbol{a}) f(\boldsymbol{y},\boldsymbol{a})\,\mathrm{d}\boldsymbol{y}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{\boldsymbol{a}\in\mathcal{X}} P_{\boldsymbol{X}}(\boldsymbol{a}) \int\limits_{\mathbb{R}^N} A\mathrm{e}^{-\frac{\|\boldsymbol{y}-\boldsymbol{a}\|^2}{B}} f(\boldsymbol{y},\boldsymbol{a})\,\mathrm{d}\boldsymbol{y} \\
&= A\sqrt{B} \sum_{\boldsymbol{a}\in\mathcal{X}} P_{\boldsymbol{X}}(\boldsymbol{a}) \int\limits_{\mathbb{R}^N} \mathrm{e}^{-\|\boldsymbol{z}\|^2} f(\sqrt{B}\boldsymbol{z}+\boldsymbol{a},\boldsymbol{a})\,\mathrm{d}\boldsymbol{z} \\
&= A\sqrt{B} \sum_{\boldsymbol{a}\in\mathcal{X}} P_{\boldsymbol{X}}(\boldsymbol{a}) \left(\sum_{i_1=1}^{K} \cdots \sum_{i_N=1}^{K} w_{i_1}\cdot\ldots\cdot w_{i_N} f(\sqrt{B}(\xi_{i_1},\xi_{i_2},\ldots,\xi_{i_N})^{\mathrm{T}}+\boldsymbol{a},\boldsymbol{a}) \right)
\end{aligned}
$$

where we introduced the abbreviations $A = 1/(2\pi\sigma^2)^{N/2}$ and $B = 2\sigma^2$. Gauss-Hermite quadratures can also be used to calculate integrals involving circularly symmetric Gaussian PDFs (3.50).

A.4. Collection of Optimized Basematrices

A.4.1. Optimized Protographs for Higher-Order Modulation

$$
\boldsymbol{B}_{\text{8-PAS-3/4-}\Lambda_2=2} = \begin{pmatrix} 1 & 0 & 0 & 4 & 2 & 2 & 2 & 2 & 0 & 0 & 0 & 4 \\ 1 & 1 & 2 & 3 & 1 & 1 & 2 & 0 & 2 & 2 & 2 & 3 \\ 0 & 1 & 1 & 4 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 4 \end{pmatrix} \tag{A.20}
$$

$$
\boldsymbol{B}_{\text{8-PAS-3/4-}\Lambda_2=0} = \begin{pmatrix} 0 & 1 & 4 & 4 & 2 & 2 & 2 & 4 & 0 & 0 & 0 & 0 \\ 2 & 0 & 4 & 4 & 0 & 1 & 0 & 4 & 1 & 1 & 1 & 1 \\ 1 & 4 & 3 & 3 & 1 & 4 & 3 & 4 & 2 & 2 & 2 & 2 \end{pmatrix} \tag{A.21}
$$

$$
\boldsymbol{B}_{\text{4-uni-3/4}} = \begin{pmatrix} 6 & 1 & 1 & 6 & 1 & 1 & 1 & 1 \\ 6 & 2 & 2 & 6 & 2 & 1 & 2 & 2 \end{pmatrix} \tag{A.22}
$$

$$
\boldsymbol{B}_{\text{8-uni-1/2}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \end{pmatrix} \tag{A.23}
$$

$$
\boldsymbol{B}_{\text{16-PAS-13/16-}\Lambda_2=0} = \begin{pmatrix} 3 & 2 & 2 & 1 & 3 & 2 & 3 & 2 & 0 & 2 & 1 & 2 & 0 & 2 & 2 & 2 \\ 1 & 0 & 0 & 3 & 2 & 1 & 2 & 2 & 3 & 0 & 3 & 0 & 3 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 1 & 3 & 1 & 2 & 3 & 1 & 2 & 1 & 3 & 1 & 1 & 1 \end{pmatrix} \tag{A.24}
$$

$$
\boldsymbol{B}_{\text{16-PAS-13/16-}\Lambda_2=2} = \begin{pmatrix} 1 & 0 & 4 & 0 & 0 & 0 & 0 & 1 & 4 & 4 & 4 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 2 & 2 & 1 & 1 & 2 & 4 & 4 & 3 & 2 & 2 & 2 & 2 & 2 \\ 0 & 1 & 4 & 1 & 4 & 2 & 2 & 2 & 4 & 4 & 4 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{A.25}
$$

For all protographs in this section, the optimal bit mapping assignment is

$$
\Phi(j) = \left((j-1) \mod \left(\frac{n_{\mathrm{p}}}{m} \right) \right) + 1, \quad j = 1, \ldots, n_{\mathrm{p}}. \tag{A.26}
$$

For instance, for $\boldsymbol{B}_{8\text{-PAS-}3/4\text{-}\Lambda_2=2}$, we have $m = 3$ (8-ASK), $n_\mathrm{p} = 12$ such that the first four VNs need to be mapped to bit level one, the next four to bit level two and and the last four to bit level three.

A.4.2. Optimized Protographs for On-Off Keying

In the following, we provide the optimized base matrices for OOK with shaping via TS as introduced in Sec. 4.9. For $\boldsymbol{B}_{\mathrm{OOK}-0.25-\mathrm{TS1}}$ the first column is punctured. The first k_p VNs of each basematrix are associated with the shaped part, the remaining ones with the unshaped part.

$$\boldsymbol{B}_{\mathrm{OOK}-0.25-\mathrm{TS1}} = \begin{pmatrix} 3 & 0 & 0 & 1 & 2 & 0 & 0 \\ 1 & 0 & 2 & 0 & 0 & 1 & 2 \\ 3 & 0 & 1 & 2 & 2 & 1 & 1 \\ 2 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{B}_{\mathrm{OOK}-0.25-\mathrm{TS2}} = \begin{pmatrix} 1 & 0 & 1 & 0 & 2 & 0 & 0 & 0 & 3 \\ 4 & 2 & 3 & 2 & 4 & 2 & 1 & 1 & 3 \\ 3 & 1 & 4 & 1 & 1 & 1 & 2 & 1 & 1 \end{pmatrix}$$

$$\boldsymbol{B}_{\mathrm{OOK}-0.67-\mathrm{TS2}} = \begin{pmatrix} 4 & 0 & 1 & 4 & 0 & 3 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 & 1 & 1 & 1 & 2 & 2 & 1 & 3 & 2 \\ 3 & 2 & 1 & 4 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 1 \end{pmatrix}$$

A.5. Differential Evolution for the Optimization of Basematrices

In the following, we describe the optimization approach based on differential evolution [87] to design basematrices for a desired operating mode.

In line 1, the initial candidates are obtained by uniformly sampling integers in $[0, b_\mathrm{max}]$, where $b_\mathrm{max} \in \mathbb{N}_0$ indicates the maximum allowed number of parallel edges and is specified in the design constraints. If the resulting basematrix does not fulfill the design constrains, it is rejected and we start over. The function `combine_and_mutate()` in line 7 is implemented as shown in Algorithm 11. Its main task is to ensure that the obtained basematrix has only non-negative integer entries after the recombination with other population members. In particular, we round each entry to the next integer in line 3.

The inner loop of Algorithm 10 is easily parallelizable, which can be used to speed-up the optimization if multiple CPUs are available.

Algorithm 10 Genetic algorithm to find the best basematrix for a given signaling mode.

```
INPUT: Protograph dimensions $m_\mathrm{p} \times n_\mathrm{p}$, design constraints, candidate set size $P$, number of generations
    $G$, crossover probability $p_c$, amplification factor $F$.
Choose feasible initial population set $\left\{\boldsymbol{B}_p^{(0)}\right\}_{p=1}^{P}$ at random.
Evaluate $\mathrm{SNR}_\mathrm{th}^\mathrm{EXIT}$ for each population member.
for $g = 1, \dots, G$ do
    for $p = 1, \dots, P$ do
        repeat
            Choose $r_1 \neq r_2 \neq r_3$ randomly from $\{1, \dots, P\}$.
            $\tilde{\boldsymbol{B}} = \text{combine\_and\_mutate}\left(\boldsymbol{B}_p^{(g-1)}, \left\{\boldsymbol{B}_p^{(g-1)}\right\}_{p=1}^{P}, r_1, r_2, r_3, F, p_\mathrm{c}\right)$.
        until $\tilde{\boldsymbol{B}}$ fulfills design constraints
        Evaluate $\mathrm{SNR}_\mathrm{th}^\mathrm{EXIT}$ of new candidate $\tilde{\boldsymbol{B}}$.
        Set $\boldsymbol{B}_p^{(g)} = \tilde{\boldsymbol{B}}$, if decoding threshold has improved.
    end for
    if all population members have the same metric then
        Stop.
    end if
end for
```

A.6. Monte-Carlo Density Evolution

MCDE was introduced in [211] and is based on assumptions that also underly the SPA: the incoming messages of a VN or CN are assumed to be stochastically independent. In the following, we describe the MCDE algorithm for protographs and NB-LDPC codes using PMFs as messages. Its application to binary codes is straightforward. As usual for protographs, we track each edge of the underlying Tanner graph separately and it turns out beneficial to consider an "edge centered" perspective in the following. We illustrate this principle with the protograph

$$\boldsymbol{B} = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 2 & 1 \end{pmatrix}$$

of the introductory example in Fig. 4.2. We first enumerate each edge as shown in Fig. A.1 and introduce the sets $\mathcal{E}_\mathtt{V}(i)$ ($\mathcal{E}_\mathtt{C}(i)$) of edges that connect to the same VN (CN) as the i-th

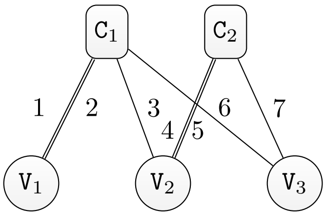


Figure A.1.: Tanner graph of a protograph with enumerated edges.

edge. For instance, we have $\mathcal{E}_\mathtt{V}(3) = \{4, 5\}$ and $\mathcal{E}_\mathtt{C}(2) = \{1, 3, 6\}$. Further, we need to keep

Algorithm 11 Implementation of the `combine_and_mutate()` function to obtain a new protograph basematrix based on the current population.

```
function COMBINE__AND__MUTATE($\boldsymbol{B}_p^{(g-1)}$, $\left\{\boldsymbol{B}_p^{(g-1)}\right\}_{p=1}^{P}$, $r_1$, $r_2$, $r_3$, $F$, $p_\mathrm{c}$)
    $\tilde{\boldsymbol{B}} = \boldsymbol{B}_{r_1}^{(g-1)} + F \cdot \left(\boldsymbol{B}_{r_2}^{(g-1)} - \boldsymbol{B}_{r_3}^{(g-1)}\right)$
    $\tilde{\boldsymbol{B}} = \mathrm{round}(\tilde{\boldsymbol{B}})$
    // find entries of $\tilde{\boldsymbol{B}}$ which are not within the allowed set
    $\mathrm{idx} = \tilde{\boldsymbol{B}}(:) < 0 \quad | \quad \tilde{\boldsymbol{B}} > b_\mathrm{max}$
    // and replace them by choosing the respective entry at random
    $\tilde{\boldsymbol{B}}(\mathrm{idx}) = \mathrm{randi}([0, b_\mathrm{max}], \mathrm{sum}(\mathrm{idx}))$
    for $i = 1 : \mathrm{numel}(\tilde{\boldsymbol{B}})$ do
        if $\mathrm{rand}() < p_\mathrm{c}$ then
            continue
        else
            $\tilde{\boldsymbol{B}}(i) = \boldsymbol{B}(i)^{(g-1)}$
        end if
    end for
    return $\tilde{\boldsymbol{B}}$
end function
```

track from which VN each edge i originates. We denote this VN as $\mathcal{V}(i)$.

The decoding threshold is found by performing a bisection search over a given start interval $(\mathrm{SNR}_\mathrm{l}, \mathrm{SNR}_\mathrm{h})$ and testing whether the SPA converges for the SNR value in the center of this interval (lines 4 – 48). To test for convergence of the SPA, we imitate the decoding algorithm for NB-LDPC codes (Algorithm 8) and use S samples per edge in protograph. A sample is defined as one PMF vector representing the belief about a certain codeword symbol. To obtain accurate results, we choose $S = 10^4$. In each iteration a new length q decoder soft information vector $\boldsymbol{m}_\mathrm{dec}$ is generated for each of the S edge samples (line 7) in an iid. fashion. For instance, this is done according to (5.22), (5.23) or (5.25). For the permutation of the message vectors (line 12), we randomly choose an element from $\mathbb{F}_q$ and use the same element for the de-permutation (line 20). If a certain construction allows only coefficients from a restricted set (see Sec. 5.1.3), this can be reflected here as well. After ℓ_max iterations we calculate the number of occurred symbol errors in lines 31 – 39. An error is declared if the zero element is not the most likely one (line 35). If no error is observed, we declare convergence for the SNR under consideration and adjust the bounds of the new interval.

Algorithm 12 MCDE for NB-LDPC codes.

```
INPUT: Basematrix $\boldsymbol{B}$, Number of samples per edge $S$, field size $q$, SNR range $(\mathrm{SNR_l}, \mathrm{SNR_h})$ in which
    the decoding threshold is expected, desired accuracy $\mathrm{SNR_{acc}}$, $\ell_{\max}$ maximum number of iterations
$E = \mathrm{sum}(\boldsymbol{B}(:))$
$\boldsymbol{m}_{V\to C} = \mathrm{zeros}(E, S, q)$
$\boldsymbol{m}_{C\to V} = \mathrm{zeros}(E, S, q)$
while $(\mathrm{SNR_h} - \mathrm{SNR_l}) > \mathrm{SNR_{acc}}$ do
    converged = 0, $\ell = 0$
    $\mathrm{SNR_m} = \mathrm{SNR_l} + (\mathrm{SNR_h} - \mathrm{SNR_l})/2$
    Generate decoder soft information $\boldsymbol{m}_{\mathrm{dec}}(i, j, :), i \in \{1, \ldots, E\}, j \in \{1, \ldots, S\}$ for $\mathrm{SNR} = \mathrm{SNR_m}$.
    while $\ell < \ell_{\max}$ do
        // CN operation
        for $i = 1, \ldots, E$ do
            for $j = 1, \ldots, S$ do
                Perform random permutation of $\boldsymbol{m}_{V\to C}(i, j, :)$ to obtain $\boldsymbol{m}^{\pi}_{V\to C}(i, j, :)$, see (5.7).
                $\boldsymbol{m}_{C\to V}(i, j, :) = \circledast_{k\in\mathcal{E}_{\mathrm{c}}(i)} \boldsymbol{m}^{\pi}_{V\to C}(k, j, :)$
            end for
        end for
        // VN operation
        Generate new iid. $\boldsymbol{m}_{\mathrm{dec}}$
        for $i = 1, \ldots, E$ do
            for $j = 1, \ldots, S$ do
                Perform de-permutation of $\boldsymbol{m}_{C\to V}(i, j, :)$ to obtain $\boldsymbol{m}^{\pi}_{C\to V}(i, j, :)$, see (5.15).
                $\boldsymbol{m}_{V\to C}(i, j, :) = \boldsymbol{m}_{\mathrm{dec}}(i, j, :) \odot \left(\bigodot_{k\in\mathcal{E}_{\mathrm{v}}(i)} \boldsymbol{m}^{\pi}_{C\to V}(k, j, :)\right)$
            end for
        end for
    end while
    // APP operation
    for $i = 1, \ldots, E$ do
        for $j = 1, \ldots, S$ do
            $\boldsymbol{m}_{\mathrm{app}}(i, j, :) = \boldsymbol{m}_{\mathrm{dec}}(i, j, :) \odot \left(\bigodot_{k\in\mathcal{V}(i)} \boldsymbol{m}_{C\to V}(k, j, :)\right)$
        end for
    end for
    err_ctr = 0
    for $i = 1, \ldots, E$ do
        for $j = 1, \ldots, S$ do
            [∼,idx_max] = max($\boldsymbol{m}_{\mathrm{app}}(i, j, :)$)
            if idx_max $\neq 0$ then
                err_ctr += 1
            end if
        end for
    end for
    if err_ctr == 0 then
        converged = 1
    end if
    if converged then
        $\mathrm{SNR_h} = \mathrm{SNR_m}$
    else
        $\mathrm{SNR_l} = \mathrm{SNR_m}$
    end if
end while
return $\mathrm{SNR_m}$
```

B

Acronyms

ADC analog-to-digital converter

APSK amplitude phase-shift keying

AR4JA accumulate-repeat-jagged-4-accumulate

ASK amplitude shift keying

AWGN additive white Gaussian noise

AWGNC additive white Gaussian noise channel

BCH Bose-Chaudhuri-Hocquenghem

BDD bounded distance decoding

BEC binary erasure channel

BEEC binary error and erasure channel

BER bit error rate

biAWGN binary-input additive white Gaussian noise

biAWGNC binary-input additive white Gaussian noise channel

BICM bit-interleaved coded modulation

BMD bit-metric decoding

BMP binary message passing

BP belief propagation

BPSK binary phase shift keying

BRGC binary reflected Gray code

BSC binary symmetric channel

CCDM constant composition distribution matching

CDF cumulative distribution function

CN check node

CSI channel state information

DA data-aided

DAC digital-to-analog converter

DD direct detection

DDE discretized density evolution

DE density evolution

DM distribution matcher

DMT discrete multitone

DOF degree of freedom

DSL digital subscriber line

DSP digital signal processing

eIRA extended irregular repeat-accumulate

EM expectation maximization

eMBB enhanced mobile broadband

EXIT extrinsic information transfer

FEC forward error correction

FER frame error rate

FN factor node

FSO free space optical communication

GMI generalized mutual information

GS geometric shaping

GVDMM generalized variable degree matched mapping

HARQ hybrid automated repeat request

HD hard decision

HT Hadamard transform

iid independent and identically distributed

IM intensity modulation

IRA irregular repeat accumulate

KKT Karush-Kuhn-Tucker

LDGM low-density generator matrix

LDPC low-density parity-check

LUT look-up table

MAP maximum a posteriori

MB Maxwell-Boltzmann

MC Monte Carlo

MCDE Monte Carlo Density Evolution

MET multi-edge type

MI mutual information

ML maximum likelihood

NA normal approximation

NB-LDPC non-binary low-density parity-check

NB non-binary

NBBC natural based binary code

NCG net coding gain

NGMI normalized generalized mutual information

NLIN non-linear interference noise

NUC non-uniform constellation

OFDM orthogonal frequency division multiplexing

OH overhead

OOK on-off keying

OSD ordered statistics decoding

P-LDPC protograph-based LDPC

PAM pulse-amplitude modulation

PAS probabilistic amplitude shaping

PBRL protograph-based Rapter-Like

PC product code

PDF probability density function

PDM product distribution matching

PEG progressive edge-growth

PMF probability mass function

PON passive optical network

PPM pulse position modulation

PS probabilistic shaping

PSK phase-shift keying

QAM quadrature amplitude modulation

QC quasi-cylic

QMP quaternary message passing

QS quadrant shaping

RC repetition code

RCB random coding bound

RCUB random coding union bound

RS Reed-Solomon

RV random variable

SC-LDPC spatially coupled low-density parity-check code

SD soft decision

SE spectral efficiency

SM shell mapping

SMD symbol-metric decoding

SMDM shell mapping as distribution matcher

SNR signal-to-noise ratio

SPA sum-product algorithm

SPB sphere packing bound

SPC single-parity check

SVD singular value decomposition

TC type check

TCM trellis coded modulation

TMP ternary message passing

TS time sharing

URLLC ultra-reliable low-latency communication

VDMM variable degree matched mapping

VN variable node

WLLN weak law of large numbers

Bibliography

[1] C. E. Shannon, “A mathematical theory of communication,” *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[2] R. Lucky, *Lucky Strikes...Again: (Feats and Foibles of Engineers).* Wiley, 1993.

[3] A. Viterbi, “Error bounds for convolutional codes and an asymptotically optimum decoding algorithm,” *IEEE Trans. Inf. Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[4] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes.” in *Proc. IEEE Int. Conf. Commun. (ICC)*, vol. 2, May 1993, pp. 1064–1070.

[5] R. G. Gallager, “Low-density parity-check codes,” *IRE Trans. Inform. Theory*, vol. 8, no. 1, pp. 21–28, 1962.

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

[7] 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.

[8] G. Böcherer, “Probabilistic signal shaping for bit-metric decoding,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2014, pp. 431–435.

[9] 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.

[10] Huawei, “Signal Shaping for QAM Constellations,” Huawei, Athens, Greece, Tech. Rep., Feb. 2018, 3GPP TSG–RAN no. 88 R1-1705061.

[11] P. Iannone, Y. Lefevre, W. Coomans, V. van Veen, and J. Cho, “Increasing Cable Bandwidth Through Probabilistic Constellation Shaping,” in *Proc. SCTE ISBE*, Oct. 2018, pp. 1–14.

[12] F. Buchali, G. Böcherer, W. Idler, L. Schmalen, P. Schulte, and F. Steiner, “Experimental demonstration of capacity increase and rate-adaptation by probabilistically

shaped 64-QAM," in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2015, paper PDP3.4.

[13] 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.

[14] W. Idler, F. Buchali, L. Schmalen, E. Lach, R. P. Braun, G. Böcherer, P. Schulte, and F. Steiner, "Field Trial of a 1 Tbit/s Super-Channel Network Using Probabilistically Shaped Constellations," *J. Lightw. Technol.*, vol. 35, no. 8, pp. 1399–1406, 2017.

[15] S. Chandrasekhar, B. Li, J. Cho, X. Chen, E. Burrows, G. Raybon, and P. Winzer, "High-spectral-efficiency transmission of PDM 256-QAM with Parallel Probabilistic Shaping at Record Rate-Reach Trade-offs," in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2016, pp. 1–3.

[16] J. Cho, X. Chen, S. Chandrasekhar, G. Raybon, R. Dar, L. Schmalen, E. Burrows, A. Adamiecki, S. Corteselli, Y. Pan, D. Correa, B. McKay, S. Zsigmond, P. Winzer, and S. Grubb, "Trans-Atlantic Field Trial Using Probabilistically Shaped 64-QAM at High Spectral Efficiencies and Single-Carrier Real-Time 250-Gb/s 16-QAM," in *Proc. Optical Fiber Commun. Conf. (OFC)*, Mar. 2017, paper Th5B.3.

[17] A. Ghazisaeidi, I. F. de Jauregui Ruiz, R. Rios-Müller, L. Schmalen, P. Tran, P. Brindel, A. C. Meseguer, Q. Hu, F. Buchali, G. Charlet, and J. Renaudier, "Advanced C+L-Band Transoceanic Transmission Systems Based on Probabilistically Shaped PDM-64QAM," *J. Lightw. Technol.*, vol. 35, no. 7, pp. 1291–1299, Apr. 2017.

[18] I. F. de Jauregui Ruiz, A. Ghazisaeidi, O. A. Sab, P. Plantady, A. Calsat, S. Dubost, L. Schmalen, V. Letellier, and J. Renaudier, "25.4-Tb/s Transmission Over Transpacific Distances Using Truncated Probabilistically Shaped PDM-64QAM," *J. Lightw. Technol.*, vol. 36, no. 6, pp. 1354–1361, Mar. 2018.

[19] S. Grubb, "Submarine Cables: Deployment, Evolution, and Perspectives," in *Proc. Optical Fiber Commun. Conf. (OFC)*, Mar. 2018, paper M1D.1.

[20] K. Roberts, Q. Zhuge, I. Monga, S. Gareau, and C. Laperle, "Beyond 100 Gb/s: Capacity, Flexibility, and Network Optimization," *J. Optical Commun. Netw.*, vol. 9, no. 4, pp. C12–C24, Apr. 2017.

[21] R. G. Gallager, *Information Theory and Reliable Communication.* John Wiley & Sons, Inc., 1968.

[22] G. Kaplan and S. Shamai, "Information rates and error exponents of compound channels with application to antipodal signaling in a fading environment," *AEU. Archiv für Elektronik und Übertragungstechnik*, vol. 47, no. 4, pp. 228–239, 1993.

[23] J. L. Massey, “Coding and modulation in digital communications,” in *Proc. Int. Zurich Seminar*, 1974, pp. E2(1)–E2(4).

[24] E. Zehavi, “8-PSK trellis codes for a Rayleigh channel,” *IEEE Trans. Commun.*, vol. 40, no. 5, pp. 873–884, May 1992.

[25] R. G. Gallager, “Capacity and coding for degraded broadcast channels,” *Problemy Peredachi Informatsii*, vol. 10, no. 3, pp. 3–14, 1974.

[26] G. Caire, G. Taricco, and E. Biglieri, “Bit-interleaved coded modulation,” *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.

[27] N. Merhav, G. Kaplan, A. Lapidoth, and S. Shamai Shitz, “On information rates for mismatched decoders,” *IEEE Trans. Inf. Theory*, vol. 40, no. 6, pp. 1953–1967, Nov. 1994.

[28] A. Ganti, A. Lapidoth, and I. E. Telatar, “Mismatched decoding revisited: General alphabets, channels with memory, and the wide-band limit,” *IEEE Trans. Inf. Theory*, vol. 46, no. 7, pp. 2315–2328, Nov. 2000.

[29] J. Scarlett, A. Martinez, and A. G. i Fabregas, “Mismatched Decoding: Error Exponents, Second-Order Rates and Saddlepoint Approximations,” *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2647–2666, May 2014.

[30] G. Kramer, “Lecture notes in Information Theory,” Technical University of Munich, Tech. Rep., Oct. 2018.

[31] S. Lin and D. Costello, *Error Control Coding: Fundamentals and Applications.* Pearson-Prentice Hall, 2004.

[32] J. Massey, “Shift-register synthesis and BCH decoding,” *IEEE Trans. Inf. Theory*, vol. 15, no. 1, pp. 122–127, Jan. 1969.

[33] F. Gray, “Pulse code communication,” U. S. Patent 2 632 058, 1953.

[34] A. Feinstein, “A new basic theorem of information theory,” *IRE Trans. Inf. Theory*, vol. 4, no. 4, pp. 2–22, Sep. 1954.

[35] C. E. Shannon, “Certain results in coding theory for noisy channels,” *Information and Control*, vol. 1, no. 1, pp. 6–25, Sep. 1957.

[36] V. Strassen, “Asymptotische Abschätzugen in Shannon’s Informationstheorie,” in *Trans. Prague Conf. Inf. Theory Etc, 1962. Czech. Academy of Sc., Prague*, 1962, pp. 689–723.

[37] R. Gallager, “A simple derivation of the coding theorem and some applications,” *IEEE Trans. Inf. Theory*, vol. 11, no. 1, pp. 3–18, Jan. 1965.

[38] 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.

[39] C. E. Shannon, “Probability of error for optimal codes in a Gaussian channel,” *Bell Syst. Tech. J.*, vol. 38, no. 3, pp. 611–656, May 1959.

[40] A. Valembois and M. P. C. Fossorier, “Sphere-packing bounds revisited for moderate block lengths,” *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 2998–3014, Dec. 2004.

[41] C. E. Shannon, R. G. Gallager, and E. R. Berlekamp, “Lower bounds to error probability for coding on discrete memoryless channels. I,” *Information and Control*, vol. 10, no. 1, pp. 65–103, Jan. 1967.

[42] J. Font-Segura, G. Vazquez-Vilar, A. Martinez, A. Guillén i Fàbregas, and A. Lancho, “Saddlepoint approximations of lower and upper bounds to the error probability in channel coding,” in *Proc. Ann. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2018, pp. 1–6.

[43] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[44] S. M. Aji and R. J. McEliece, “The generalized distributive law,” *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 325–343, Mar. 2000.

[45] G. Forney, R. Gallager, G. Lang, F. Longstaff, and S. Qureshi, “Efficient Modulation for Band-Limited Channels,” *IEEE J. Sel. Areas Commun.*, vol. 2, no. 5, pp. 632–647, Sep. 1984.

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

[47] G. D. Forney Jr, “The Viterbi Algorithm: A Personal History,” *arXiv:cs/0504020*, Apr. 2005.

[48] Robert J. McEliece, “Are turbo-like codes effective on nonstandard channels?” *IEEE Inf. Theo. Society Newsletter*, pp. 3–8, Dec. 2001.

[49] D. Raphaeli and A. Gurevitz, “Constellation shaping for pragmatic turbo-coded modulation with high spectral efficiency,” *IEEE Trans. Commun.*, vol. 52, no. 3, pp. 341–345, Mar. 2004.

[50] M. Yankov, D. Zibar, K. Larsen, L. Christensen, and S. Forchhammer, “Constellation Shaping for Fiber-Optic Channels With QAM and High Spectral Efficiency,” *IEEE Photon. Technol. Lett.*, vol. 26, no. 23, pp. 2407–2410, Dec. 2014.

[51] D. Feng, Q. Li, B. Bai, and X. Ma, “Gallager mapping based constellation shaping for LDPC-coded modulation systems,” in *Proc. Int. Workshop on High Mobility Wireless Commun. (HMWC)*, Oct. 2015, pp. 116–120.

[52] C. Pan and F. R. Kschischang, “Probabilistic 16-QAM Shaping in WDM Systems,” *J. Lightw. Technol.*, vol. 34, no. 18, pp. 4285–4292, Sep. 2016.

[53] M. P. Yankov, F. Da Ros, E. P. da Silva, S. Forchhammer, K. J. Larsen, L. K. Oxenløwe, M. Galili, and D. Zibar, “Constellation Shaping for WDM Systems Using 256QAM/1024QAM With Probabilistic Optimization,” *J. Lightw. Technol.*, vol. 34, no. 22, pp. 5146–5156, Nov. 2016.

[54] G. Böcherer and R. Mathar, “Matching Dyadic Distributions to Channels,” in *Proc. Data Compression Conf. (DCC)*, Mar. 2011, pp. 23–32.

[55] G. D. Forney, “Trellis shaping,” *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 281–300, Mar. 1992.

[56] S. A. Tretter, *Constellation Shaping, Nonlinear Precoding, and Trellis Coding for Voiceband Telephone Channel Modems: With Emphasis on ITU-T Recommendation V.34*, ser. The Springer International Series in Engineering and Computer Science. Springer Science, 2002.

[57] R. Laroia, N. Farvardin, and S. A. Tretter, “On optimal shaping of multidimensional constellations,” *IEEE Trans. Inf. Theory*, vol. 40, no. 4, pp. 1044–1056, Jul. 1994.

[58] W. G. Bliss, “Circuitry for performing error correction calculations on baseband encoded data to eliminate error propagation,” *IBM Tech. Discl. Bull.*, vol. 23, pp. 4633–4634, 1981.

[59] M. Mansuripur, “Enumerative modulation coding with arbitrary constraints and postmodulation error correction coding for data storage systems,” in *Optical Data Storage ’91*, vol. 1499, Jul. 1991, pp. 72–87.

[60] K. A. S. Immink, “A practical method for approaching the channel capacity of constrained channels,” *IEEE Trans. Inf. Theory*, vol. 43, no. 5, pp. 1389–1399, Sep. 1997.

[61] M. Blaum, R. D. Cideciyan, E. Eleftheriou, R. Galbraith, K. Lakovic, T. Mittelholzer, T. Oenning, and B. Wilson, “High-Rate Modulation Codes for Reverse Concatenation,” *IEEE Trans. Magn.*, vol. 43, no. 2, pp. 740–743, Feb. 2007.

[62] E. Ratzer, “Error-Correction on Non-Standard Communication Channels,” Ph.D. Thesis, University of Cambridge, 2003.

[63] 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.

[64] F. Kschischang and S. Pasupathy, “Optimal nonuniform signaling for Gaussian channels,” *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 913–929, May 1993.

[65] J. Huang and S. P. Meyn, "Characterization and computation of optimal distributions for channel coding," *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2336–2351, Jul. 2005.

[66] P. Schulte and G. Böcherer, "Constant Composition Distribution Matching," *IEEE Trans. Inf. Theory*, vol. 62, no. 1, pp. 430–434, Jan. 2016.

[67] G. Böcherer and B. C. Geiger, "Optimal Quantization for Distribution Synthesis," *IEEE Trans. Inf. Theory*, vol. 62, no. 11, pp. 6162–6172, Nov. 2016.

[68] T. V. Ramabadran, "A coding scheme for m-out-of-n codes," *IEEE Trans. Commun.*, vol. 38, no. 8, pp. 1156–1163, Aug. 1990.

[69] P. Schulte and B. C. Geiger, "Divergence scaling of fixed-length, binary-output, one-to-one distribution matching," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017, pp. 3075–3079.

[70] ITU-T Recommendation V.34, "A modem operating at data signalling rates of up to 33 600 bit/s for use on the general switched telephone network and on leased point-to-point 2-wire elephone-type circuits," Feb. 1998.

[71] J. Schalkwijk, "An algorithm for source coding," *IEEE Trans. Inf. Theory*, vol. 18, no. 3, pp. 395–399, May 1972.

[72] T. Cover, "Enumerative source encoding," *IEEE Trans. Inf. Theory*, vol. 19, no. 1, pp. 73–77, Jan. 1973.

[73] Y. C. Gültekin, F. M. J. Willems, W. J. van Houtum, and S. Şerbetli, "Approximate Enumerative Sphere Shaping," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2018, pp. 676–680.

[74] 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.

[75] R. F. H. Fischer, "Calculation of shell frequency distributions obtained with shell-mapping schemes," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1631–1639, Jul. 1999.

[76] G. Böcherer, "Achievable Rates for Probabilistic Shaping," *arXiv:1707.01134v5*, May 2018.

[77] R. A. Amjad, "Information Rates and Error Exponents for Probabilistic Amplitude Shaping," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Nov. 2018, pp. 1–5.

[78] F.-W. Sun and H. C. A. van Tilborg, "Approaching capacity by equiprobable signaling on the Gaussian channel," *IEEE Trans. Inf. Theory*, vol. 39, no. 5, pp. 1714–1716, Sep. 1993.

[79] M. F. Barsoum, C. Jones, and M. Fitz, “Constellation Design via Capacity Maximization,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2007, pp. 1821–1825.

[80] “Digital Video Broadcasting (DVB); Next Generation broadcasting system to Handheld, physical layer specification (DVB-NGH),” no. A160, Nov. 2013.

[81] D. Gómez-Barquero, C. Douillard, P. Moss, and V. Mignone, “DVB-NGH: The Next Generation of Digital Broadcast Services to Handheld Devices,” *IEEE Trans. Broadcast.*, vol. 60, no. 2, pp. 246–257, Jun. 2014.

[82] “ATSC Proposed Standard: Physical Layer Protocol (A/322),” no. S32-230r56, Jun. 2016.

[83] N. S. Loghin, J. Zöllner, B. Mouhouche, D. Ansorregui, J. Kim, and S. I. Park, “Non-Uniform Constellations for ATSC 3.0,” *IEEE Trans. Broadcast.*, vol. 62, no. 1, pp. 197–203, Mar. 2016.

[84] Z. Qu and I. B. Djordjevic, “Geometrically Shaped 16QAM Outperforming Probabilistically Shaped 16QaM,” in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2017, pp. 1–3.

[85] B. Chen, C. Okonkwo, H. Hafermann, and A. Alvarado, “Increasing Achievable Information Rates via Geometric Shaping,” in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2018, pp. 1–3.

[86] R. T. Jones, T. A. Eriksson, M. P. Yankov, and D. Zibar, “Deep Learning of Geometric Constellation Shaping Including Fiber Nonlinearities,” in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2018, pp. 1–3.

[87] R. Storn and K. Price, “Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces,” *J. Global Optimization*, vol. 11, no. 4, pp. 341–359, Dec. 1997.

[88] J. Zoellner and N. Loghin, “Optimization of high-order non-uniform QAM constellations,” in *Proc. IEEE Int. Symp. Broadband Multim. Syst. Broadc. (BMSB)*, Jun. 2013, pp. 1–6.

[89] F. Kayhan and G. Montorsi, “Constellation design for transmission over nonlinear satellite channels,” in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2012, pp. 3401–3406.

[90] H. Méric, “Approaching the Gaussian Channel Capacity With APSK Constellations,” *IEEE Commun. Lett.*, vol. 19, no. 7, pp. 1125–1128, Jul. 2015.

[91] Y. Wu and S. Verdú, “The impact of constellation cardinality on Gaussian channel capacity,” in *Proc. Allerton Conf. Commun., Contr., Comput.*, Sep. 2010, pp. 620–628.

[92] K. J. Kim, S. Myung, S. I. Park, J. Y. Lee, M. Kan, Y. Shinohara, J. W. Shin, and J. Kim, “Low-Density Parity-Check Codes for ATSC 3.0,” *IEEE Trans. Broadcast.*, vol. 62, no. 1, pp. 189–196, Mar. 2016.

[93] L. Michael and D. Gómez-Barquero, “Bit-Interleaved Coded Modulation (BICM) for ATSC 3.0,” *IEEE Trans. Broadcast.*, vol. 62, no. 1, pp. 181–188, Mar. 2016.

[94] H. Jin, A. Khandekar, and R. McEliece, “Irregular repeat-accumulate codes,” in *Proc. Int. Symp. Turbo Codes Iter. Inf. Process. (ISTC)*, 2000, pp. 1–8.

[95] M. Pikus and W. Xu, “Bit-Level Probabilistically Shaped Coded Modulation,” *IEEE Commun. Lett.*, vol. 21, no. 9, pp. 1929–1932, Sep. 2017.

[96] ——, “Applying bit-level probabilistically shaped coded modulation for high-throughput communications,” in *Proc. IEEE Int. Symp. Personal, Indoor, Mobile Radio Commun. (PIMRC)*, Montreal, Canada, Oct. 2017, pp. 1–6.

[97] A. Guillén i Fàbregas and A. Martinez, “Bit-interleaved coded modulation with shaping,” in *IEEE Inf. Theory Workshop (ITW)*, 2010.

[98] “Digital Video Broadcasting (DVB); 2nd Generation Framing Structure, Channel Coding and Modulation Systems for Broadcasting, Interactive Services, News Gathering and Other Broadband Satellite Applications (DVB-S2),” no. EN 302 307, 2009.

[99] A. Lozano, A. M. Tulino, and S. Verdu, “Mercury/waterfilling: Optimum power allocation with arbitrary input constellations,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Sep. 2005, pp. 1773–1777.

[100] W. Coomans, R. B. Moraes, K. Hooghe, A. Duque, J. Galaro, M. Timmers, A. J. van Wijngaarden, M. Guenach, and J. Maes, “XG-fast: The 5th generation broadband,” *IEEE Commun. Mag.*, vol. 53, no. 12, pp. 83–88, Dec. 2015.

[101] 3GPP, “3GPP TS 38.212 V15.0.0: Multiplexing and channel coding,” 3GPP, Tech. Rep., Dec. 2017.

[102] 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.

[103] M. Ivanov, F. Brännstrom, A. Alvarado, and E. Agrell, “On the Exact BER of Bit-Wise Demodulators for One-Dimensional Constellations,” *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1450–1459, Apr. 2013.

[104] A. Sheikh, A. Graell 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.

[105] ITU, “G.975 forward error correction for submarine systems,” ITU, Tech. Rep., Oct. 2010.

[106] B. Dorsch, “A decoding algorithm for binary block codes and J-ary output channels (corresp.),” *IEEE Trans. Inf. Theory*, vol. 20, no. 3, pp. 391–394, May 1974.

[107] M. P. C. Fossorier and S. Lin, “Soft-decision decoding of linear block codes based on ordered statistics,” *IEEE Trans. Inf. Theory*, vol. 41, no. 5, pp. 1379–1396, Sep. 1995.

[108] 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.

[109] M. Seidl, A. Schenk, C. Stierstorfer, and J. B. Huber, “Polar-Coded Modulation,” *IEEE Trans. Commun.*, vol. 61, no. 10, pp. 4108–4119, Oct. 2013.

[110] 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)*, San Francisco, USA, Mar. 2017.

[111] R. Tanner, “A recursive approach to low complexity codes,” *IEEE Trans. Inf. Theory*, vol. 27, no. 5, pp. 533–547, Sep. 1981.

[112] S.-Y. Chung, G. D. Forney Jr, T. J. Richardson, and R. Urbanke, “On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit,” *IEEE Commun. Lett.*, vol. 5, no. 2, pp. 58–60, 2001.

[113] Wen-Yen Weng, A. Ramamoorthy, and R. D. Wesel, “Lowering the error floors of irregular high-rate LDPC codes by graph conditioning,” in *Proc. IEEE Veh. Technol. Conf. (VTC)*, Sep. 2004, pp. 2549–2553.

[114] C. Di, T. Richardson, and R. Urbanke, “Weight Distribution of Low-Density Parity-Check Codes,” *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 4839–4855, Nov. 2006.

[115] D. Divsalar, H. Jin, and R. J. McEliece, “Coding theorems for "turbo-like" codes,” in *Proc. Allerton Conf. Commun., Contr., Comput.*, 1998, pp. 201–210.

[116] T. Richardson and R. Urbanke, “Multi-edge type LDPC codes,” *Workshop honoring Prof. Bob McEliece on his 60th birthday, California Institute of Technology, Pasadena, California*, pp. 24–25, 2002.

[117] J. Thorpe, “Low-density parity-check (LDPC) codes constructed from protographs,” *IPN progress report*, vol. 42, no. 154, pp. 42–154, 2003.

[118] D. Divsalar, S. Dolinar, C. R. Jones, and K. Andrews, “Capacity-approaching protograph codes,” *IEEE J. Sel. Areas Commun.*, vol. 27, no. 6, pp. 876–888, Aug. 2009.

[119] The Consultative Committee for Space Data Systems (CCSDS), “CCSDS 131.0-B-3: TM Synchronization and Channel Coding,” Tech. Rep., Sep. 2017.

[120] E. Paolini and M. F. Flanagan, “Efficient and Exact Evaluation of the Weight Spectral Shape and Typical Minimum Distance of Protograph LDPC Codes,” *IEEE Commun. Lett.*, vol. 20, no. 11, pp. 2141–2144, Nov. 2016.

[121] T. Y. Chen, K. Vakilinia, D. Divsalar, and R. D. Wesel, “Protograph-Based Raptor-Like LDPC Codes,” *IEEE Trans. Commun.*, vol. 63, no. 5, pp. 1522–1532, May 2015.

[122] T. Richardson and S. Kudekar, “Design of Low-Density Parity Check Codes for 5G New Radio,” *IEEE Commun. Mag.*, vol. 56, no. 3, pp. 28–34, Mar. 2018.

[123] 3GPP, “3GPP TS 38.211 V15.0.0: Physical channels and modulation,” 3GPP, Tech. Rep., Dec. 2017.

[124] T. Etzion, A. Trachtenberg, and A. Vardy, “Which codes have cycle-free Tanner graphs?” *IEEE Trans. Inf. Theory*, vol. 45, no. 6, pp. 2173–2181, Sep. 1999.

[125] J. Chen, A. Dholakia, E. Eleftheriou, M. P. C. Fossorier, and X.-Y. Hu, “Reduced-Complexity Decoding of LDPC Codes,” *IEEE Trans. Commun.*, vol. 53, no. 8, pp. 1288–1299, Aug. 2005.

[126] B. K. Butler and P. H. Siegel, “Error Floor Approximation for LDPC Codes in the AWGN Channel,” *IEEE Trans. Inf. Theory*, vol. 60, no. 12, pp. 7416–7441, Dec. 2014.

[127] D. Hocevar, “A reduced complexity decoder architecture via layered decoding of LDPC codes,” in *Proc. IEEE Workshop Signal Process. Syst. (SIPS)*, Oct. 2004, pp. 107–112.

[128] Jinghu Chen and M. P. C. Fossorier, “Near optimum universal belief propagation based decoding of low-density parity check codes,” *IEEE Trans. Commun.*, vol. 50, no. 3, pp. 406–414, Mar. 2002.

[129] T. J. Richardson and R. L. Urbanke, “The capacity of low-density parity-check codes under message-passing decoding,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, 2001.

[130] S. Y. Chung, “On the Construction of Some Capacity-Approaching Coding Schemes,” Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts, Sep. 2000.

[131] J. Hou, P. H. Siegel, L. B. Milstein, and H. D. Pfister, “Capacity-approaching bandwidth-efficient coded modulation schemes based on low-density parity-check codes,” *IEEE Trans. Inf. Theory*, vol. 49, no. 9, pp. 2141–2155, Sep. 2003.

[132] S. ten Brink, “Convergence behavior of iteratively decoded parallel concatenated codes,” *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, 2001.

[133] S. ten Brink, G. Kramer, and A. Ashikhmin, “Design of low-density parity-check codes for modulation and detection,” *IEEE Commun. Lett.*, vol. 52, no. 4, pp. 670–678, 2004.

[134] F. Brännstrom, 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.

[135] A. Ashikhmin, G. Kramer, and S. ten Brink, “Extrinsic information transfer functions: Model and erasure channel properties,” *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2657–2673, 2004.

[136] G. Liva and M. Chiani, “Protograph LDPC Codes Design Based on EXIT Analysis,” in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov. 2007, pp. 3250–3254.

[137] D. Divsalar and C. Jones, “Protograph based low error floor LDPC coded modulation,” in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Oct. 2005, pp. 378–385.

[138] G. Durisi, L. Dinoi, and S. Benedetto, “eIRA Codes for Coded Modulation Systems,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2006, pp. 1125–1130.

[139] G. Richter, A. Hof, and M. Bossert, “On the Mapping of Low-Density Parity-Check Codes for Bit-Interleaved Coded Modulation,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2007, pp. 2146–2150.

[140] Y. Jin, M. Jiang, and C. Zhao, “Optimized variable degree matched mapping for protograph LDPC coded modulation with 16QAM,” in *Proc. Int. Symp. Turbo Codes Iter. Inf. Process. (ISTC)*, Sep. 2010, pp. 161–165.

[141] T. V. Nguyen, A. Nosratinia, and D. Divsalar, “Threshold of Protograph-Based LDPC Coded BICM for Rayleigh Fading,” in *IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2011, pp. 1–5.

[142] C. Tang, H. Shen, M. Jiang, and C. Zhao, “Optimization of Generalized VDMM for Protograph-Based LDPC Coded BICM,” *IEEE Commun. Lett.*, vol. 18, no. 5, pp. 853–856, May 2014.

[143] T. Cheng, K. Peng, J. Song, and K. Yan, “EXIT-Aided Bit Mapping Design for LDPC Coded Modulation with APSK Constellations,” *IEEE Commun. Lett.*, vol. 16, no. 6, pp. 777–780, Jun. 2012.

[144] C. Häger, A. Graell i Amat, A. Alvarado, F. Brännström, and E. Agrell, “Optimized Bit Mappings for Spatially Coupled LDPC Codes over Parallel Binary Erasure Channels,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 2064–2069.

[145] C. Häger, A. Graell i Amat, F. Brännström, A. Alvarado, and E. Agrell, “Improving soft FEC performance for higher-order modulations via optimized bit channel mappings,” *Optics Express*, vol. 22, no. 12, pp. 14 544–14 558, Jun. 2014.

[146] L. Zhang and F. Kschischang, “Multi-Edge-Type Low-Density Parity-Check Codes for Bandwidth-Efficient Modulation,” *IEEE Trans. Commun.*, vol. 61, no. 1, pp. 43–52, Jan. 2013.

[147] F. Steiner, G. Böcherer, and G. Liva, “Protograph-based LDPC code design for bit-metric decoding,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2015, pp. 1089–1093.

[148] ——, “Protograph-Based LDPC Code Design for Shaped Bit-Metric Decoding,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 2, pp. 397–407, Feb. 2016.

[149] F. Peng, W. Ryan, and R. Wesel, “Surrogate-channel design of universal LDPC codes,” *IEEE Commun. Lett.*, vol. 10, no. 6, pp. 480–482, Jun. 2006.

[150] M. Franceschini, G. Ferrari, and R. Raheli, “Does the Performance of LDPC Codes Depend on the Channel?” *IEEE Trans. Commun.*, vol. 54, no. 12, pp. 2129–2132, Dec. 2006.

[151] I. Sason, “On Universal Properties of Capacity-Approaching LDPC Code Ensembles,” *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 2956–2990, Jul. 2009.

[152] R. Hooke and T. A. Jeeves, “Direct Search Solution of Numerical and Statistical Problems,” *J. ACM*, vol. 8, no. 2, pp. 212–229, Apr. 1961.

[153] IEEE LAN/MAN Standards Committee, “Draft Standard for Ethernet Amendment: Physical Layer Specifications and Management Parameters for 25 Gb/s, 50 Gb/s, and 100 Gb/s Passive Optical Networks,” IEEE, Tech. Rep., Mar. 2018, IEEE P802.3ca/D1.0.

[154] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, “Regular and irregular progressive edge-growth Tanner graphs,” *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 386–398, Jan. 2005.

[155] D. Divsalar, S. Dolinar, and C. Jones, “Construction of Protograph LDPC Codes with Linear Minimum Distance,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2006, pp. 664–668.

[156] “LTE; evolved universal terrestrial radio access (E-UTRA); physical layer procedures,” no. TS 136 213, 2013.

[157] “Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 2: DVB-S2 Extensions (DVB-S2X),” no. EN 302 307-2, 2014.

[158] T. V. Nguyen, A. Nosratinia, and D. Divsalar, "The Design of Rate-Compatible Protograph LDPC Codes," *IEEE Trans. Commun.*, vol. 60, no. 10, pp. 2841–2850, Oct. 2012.

[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] J. Chen and P. M. C. Fossorier, "Density evolution for BP-based decoding algorithms of LDPC codes and their quantized versions," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, vol. 2, Nov. 2002, pp. 1378–1382 vol.2.

[161] J. Zhao, F. Zarkeshvari, and A. H. Banihashemi, "On implementation of min-sum algorithm and its modifications for decoding low-density parity-check (LDPC) codes," *IEEE Trans. Commun.*, vol. 53, no. 4, pp. 549–554, Apr. 2005.

[162] J. Hamkins, "Performance of Low-Density Parity-Check Coded Modulation," *IPN progress report*, vol. 42, no. 184, pp. 1–36, Feb. 2011.

[163] A. Schmitt, "OFC 2018: Post-Show Report," 2018.

[164] A. J. Feltström, D. Truhachev, M. Lentmaier, and K. S. Zigangirov, "Braided Block Codes," *IEEE Trans. Inf. Theory*, vol. 55, no. 6, pp. 2640–2658, Jun. 2009.

[165] M. Magarini, R. 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.

[166] L. M. Zhang and F. R. Kschischang, "Low-Complexity Soft-Decision Concatenated LDGM-Staircase FEC for High-Bit-Rate Fiber-Optic Communication," *J. Lightw. Technol.*, vol. 35, no. 18, pp. 3991–3999, Sep. 2017.

[167] A. Sheikh, A. Graell i Amat, and G. Liva, "Binary Message Passing Decoding of Product-like Codes," *arXiv:1902.03575*, Feb. 2019.

[168] G. Liga, A. Sheikh, and A. Alvarado, "A novel soft-aided bit-marking decoder for product codes," *arXiv:1906.09792*, Jun. 2019.

[169] Y. Lei, B. Chen, G. Liga, X. Deng, Z. Cao, J. Li, K. Xu, and A. Alvarado, "Improved Decoding of Staircase Codes: The Soft-aided Bit-marking (SABM) Algorithm," *arXiv:1902.01178*, Feb. 2019.

[170] G. Lechner, T. Pedersen, and G. Kramer, "Analysis and Design of Binary Message Passing Decoders," *IEEE Trans. Commun.*, vol. 60, no. 3, pp. 601–607, Mar. 2012.

[171] E. Ben Yacoub, F. Steiner, B. Matuz, and G. Liva, "Protograph-Based LDPC Code Design for Ternary Message Passing Decoding," in *Proc. Int. ITG Conf. Syst. Commun. Coding (SCC)*, Rostock, Germany, Feb. 2019.

[172] L. Schmalen, V. Aref, J. Cho, D. Suikat, D. Rösener, and A. Leven, "Spatially Coupled Soft-Decision Error Correction for Future Lightwave Systems," *J. Lightw. Technol.*, vol. 33, no. 5, pp. 1109–1116, Mar. 2015.

[173] A. R. Iyengar, P. H. Siegel, R. L. Urbanke, and J. K. Wolf, "Windowed Decoding of Spatially Coupled Codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 4, pp. 2277–2292, Apr. 2013.

[174] S. Kudekar, T. J. Richardson, and R. L. Urbanke, "Threshold Saturation via Spatial Coupling: Why Convolutional LDPC Ensembles Perform So Well over the BEC," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 803–834, Feb. 2011.

[175] A. R. Iyengar, M. Papaleo, P. H. Siegel, J. K. Wolf, A. Vanelli-Coralli, and G. E. Corazza, "Windowed Decoding of Protograph-Based LDPC Convolutional Codes Over Erasure Channels," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2303–2320, Apr. 2012.

[176] F. Steiner, E. Ben Yacoub, B. Matuz, G. Liva, and A. G. i Amat, "One and Two Bit Message Passing for SC-LDPC Codes With Higher-Order Modulation," *J. Lightw. Technol.*, vol. 37, no. 23, pp. 5914–5925, Dec. 2019.

[177] X. Y. Hu and E. Eleftheriou, "Cycle tanner-graph codes over GF(2^b)," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2003, p. 87.

[178] A. Venkiah, D. Declercq, and C. Poulliat, "Design of cages with a randomized progressive edge-growth algorithm," *IEEE Commun. Lett.*, vol. 12, no. 4, pp. 301–303, Apr. 2008.

[179] M. P. C. Fossorier, "Quasicyclic low-density parity-check codes from circulant permutation matrices," *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1788–1793, Aug. 2004.

[180] C. Poulliat, M. Fossorier, and D. Declercq, "Design of regular (2,d_c)-LDPC codes over GF(q) using their binary images," *IEEE Trans. Commun.*, vol. 56, no. 10, pp. 1626–1635, Oct. 2008.

[181] L. Barnault and D. Declercq, "Fast decoding algorithm for LDPC over GF(2^q)," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Paris, Mar. 2003, pp. 70–73.

[182] M. C. Davey and D. MacKay, "Low-density parity check codes over GF(q)," *IEEE Commun. Lett.*, vol. 2, no. 6, pp. 165–167, Jun. 1998.

[183] J. MacWilliams, "A theorem on the distribution of weights in a systematic code," *Bell Sys. Tech. J.*, vol. 42, no. 1, pp. 79–94, Jan. 1963.

[184] E. Boutillon, "Optimization of Non Binary Parity Check Coefficients," *IEEE Trans. Inf. Theory*, vol. 65, no. 4, pp. 2092–2100, Apr. 2019.

[185] J. J. Boutros, F. Jardel, and C. Méasson, “Probabilistic shaping and non-binary codes,” in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017, pp. 2308–2312.

[186] A. Bennatan and D. Burshtein, “Design and analysis of nonbinary LDPC codes for arbitrary discrete-memoryless channels,” *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 549–583, Feb. 2006.

[187] A. Alvarado, E. Agrell, D. Lavery, R. Maher, and P. Bayvel, “Replacing the Soft-Decision FEC Limit Paradigm in the Design of Optical Communication Systems,” *J. Lightw. Technol.*, vol. 33, no. 20, pp. 4338–4352, Oct. 2015.

[188] J. Cho, X. Chen, S. Chandrasekhar, and P. Winzer, “On line rates, information rates, and spectral efficiencies in probabilistically shaped QAM systems,” *Optics Express*, vol. 26, no. 8, pp. 9784–9791, Apr. 2018.

[189] J. Cho, L. Schmalen, and P. J. Winzer, “Normalized generalized mutual information as a forward error correction threshold for probabilistically shaped qam,” in *Proc. Eur. Conf. Optical Commun. (ECOC)*, Sep. 2017, Paper M.2.D.2.

[190] R. Dar, M. Feder, A. Mecozzi, and M. Shtaif, “Properties of nonlinear noise in long, dispersion-uncompensated fiber links,” *Optics Express*, vol. 21, no. 22, pp. 25 685–25 699, Nov. 2013.

[191] P. Poggiolini, G. Bosco, A. Carena, V. Curri, Y. Jiang, and F. Forghieri, “The GN-Model of Fiber Non-Linear Propagation and its Applications,” *J. Lightw. Technol.*, vol. 32, no. 4, pp. 694–721, Feb. 2014.

[192] L. Schmalen, “Probabilistic constellation shaping: Challenges and opportunities for forward error correction,” in *Proc. Optical Fiber Commun. Conf. (OFC)*, Mar. 2018, Paper M3C.1.

[193] M. J. Wainwright and M. I. Jordan, “Graphical Models, Exponential Families, and Variational Inference,” *Found. Trends Mach. Learn.*, vol. 1, no. 1-2, pp. 1–305, Jan. 2008.

[194] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[195] J. MacQueen, “Some methods for classification and analysis of multivariate observations,” in *Proc. Fifth Berkeley Symp. Math. Statistics and Probability*, vol. 1: Statistics, 1967.

[196] H. Steinhaus, “Sur la division des corps matériels en parties,” *Bulletin de l’Académie Polonaise des Sciences, Classe 3*, vol. 4, pp. 801–804, 1957.

[197] O. Simeone, “A Brief Introduction to Machine Learning for Engineers,” *arXiv:1709.02840*, Sep. 2017.

[198] A. Alvarado and E. Agrell, “Four-Dimensional Coded Modulation with Bit-Wise Decoders for Future Optical Communications,” *J. Lightw. Technol.*, vol. 33, no. 10, pp. 1993–2003, May 2015.

[199] P. Schulte, F. Steiner, and G. Böcherer, “Four dimensional probabilistic shaping for fiber-optic communication,” in *Proc. Advanced Photonics*, Jul. 2017, Paper SpM2F.5.

[200] J. Renner, T. Fehenberger, M. P. Yankov, F. D. Ros, S. Forchhammer, G. Böcherer, and N. Hanik, “Experimental Comparison of Probabilistic Shaping Methods for Unrepeated Fiber Transmission,” *J. Lightw. Technol.*, vol. 35, no. 22, pp. 4871–4879, Nov. 2017.

[201] T. Fehenberger, A. Alvarado, G. Böcherer, and N. Hanik, “On Probabilistic Shaping of Quadrature Amplitude Modulation for the Nonlinear Fiber Channel,” *J. Lightw. Technol.*, vol. 34, no. 21, pp. 5063–5073, Nov. 2016.

[202] 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.

[203] G. Böcherer, D. Lentner, A. Cirino, and F. Steiner, “Probabilistic Parity Shaping for Linear Codes,” *arXiv:1902.10648*, Feb. 2019.

[204] M. Ebada, S. Cammerer, A. Elkelesh, and S. ten Brink, “Deep Learning-based Polar Code Design,” *arXiv:1909.12035*, Sep. 2019.

[205] A. Elkelesh, M. Ebada, S. Cammerer, L. Schmalen, and S. ten Brink, “Decoder-in-the-Loop: Genetic Optimization-based LDPC Code Design,” *accepted for IEEE Access*, 2019.

[206] A. Elkelesh, M. Ebada, S. Cammerer, and S. t Brink, “Decoder-Tailored Polar Code Design Using the Genetic Algorithm,” *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 4521–4534, Jul. 2019.

[207] I. P. Mulholland, E. Paolini, and M. F. Flanagan, “Design of Protograph-based LDPC Code Ensembles with Fast Convergence Properties,” in *Proc. Int. ITG Conf. Source Channel Coding (SCC)*, Feb. 2017, pp. 1–6.

[208] B.-Y. Chang, L. Dolecek, and D. Divsalar, “EXIT chart analysis and design of non-binary protograph-based LDPC codes,” in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Nov. 2011, pp. 566–571.

[209] M. Bazaraa, H. Sherali, and C. Shetty, *Nonlinear Programming: Theory and Algorithms.* John Wiley & Sons, 2013.

[210] F. W. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark, *NIST Handbook of Mathematical Functions*, 1st ed. New York, NY, USA: Cambridge University Press, 2010.

[211] D. J. C. MacKay, *Information Theory, Inference & Learning Algorithms.* New York, NY, USA: Cambridge University Press, 2002.