UNIVERSITY OF CALIFORNIA
RIVERSIDE

Low Latency and Low Complexity Communication on High Noise Channels

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering

by

Navid Gharavi

December 2021

Dissertation Committee:

Professor Ilya Dumer, Chairperson
Professor Ertem Tuncel
Professor Leonid P. Pryadko

The Dissertation of Navid Gharavi is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

I would like to thank everyone who helped me through this journey. First of all, I would like to express my sincere gratitude to Prof. Ilya Dumer. I was immensely lucky to have him as my advisor and work under his supervision. I am extremely grateful for his guidance, insight, intuition and enthusiasm which inspired me through these years.

I would also like to express my gratitude to my dissertation committee members; professors Ertem Tuncel and Leonid Pryadko. I would like to thank Prof. Alexander N. Korotkov for allowing me to use his lab's computers and equipment during the last 6 years.

Finally, I would like to thank my parents and my brothers without whom I would not be able to continue during these years away from home.

The content of this thesis, is reflected in the following publications:

1. I. Dumer and N. Gharavi, "Combined polar-LDPC design for channels with high noise", 2021 IEEE Information Theory Workshop (ITW), Kanazawa, Japan,(Virtual Symposium), October 17-21, 2021, pp. 1-6

2. I. Dumer and N. Gharavi, "Codes approaching the Shannon limit with polynomial complexity per information bit," 2021 IEEE International Symposium on Information Theory (ISIT) (Virtual Symposium), July 12-20, 2021, paper D1-T2-S4, pp. 238-243.

3. I. Dumer and N. Gharavi, "Codes approaching the Shannon limit with polynomial complexity per information bit," Jan. 2021, http://arxiv.org/pdf/2101.10145, 18 pp.

4. I. Dumer and N. Gharavi, "Codes for high-noise memoryless channels," 2020 International Symposium on Information Theory and Its Applications (ISITA) (Virtual Symposium), October 25-27, 2020, paper A03-04, pp. 101-105. (Received best student paper award.)

To my loving parents and my brothers Kaveh and Amir.

ABSTRACT OF THE DISSERTATION

Low Latency and Low Complexity Communication on High Noise Channels

by

Navid Gharavi

Doctor of Philosophy, Graduate Program in Electrical Engineering
University of California, Riverside, December 2021
Professor Ilya Dumer, Chairperson

Low latency and low power communications have broad applications such as Internet of things (IoT), autonomous vehicles, industry automation (collaborative robots), health procedures (like robot assisted surgery requiring haptic feedback), satellite communication, radar applications, virtual reality headsets and millimeter Waves (mmWave). Low latency communications is in high demand for autonomous systems to be able to react swiftly to changes in the environment and to unexpected situations. Latency is tied to the technology used and even more to the overall architecture adopted. Enhanced mobile broadband, massive machine type communications and ultra-reliable and low latency communication are the most prominent promises of 5th generation mobile network (5G). The low power wide area networks communications is expected to grow exponentially from the 1.5B$ of 2018 to 65B$ in 2025 and communication services, asset tracking and smart buildings (installation and operation) are just some of its applications. Low power communications can use a number of different protocols and systems like Narrow-band (NB) IoT and Long Range Wide Area Network (LoRaWAN). Long range wide area networks use unlicensed spectrum and are more suitable for applications generated in low traffic volume (which is typically the case for IoT). All these applications of low power and low latency communication were the motivation behind our research.

In this thesis, we introduce a low-rate low-density parity-check (LDPC) code for channels with extreme noise and present a low latency and low complexity communication method for low power applications. We then show that this design has the ability of outperforming uncoded modulation for the signal-to-noise ratios (SNR) above -3 dB per information bit and achieve a 3 dB gain as SNR grows. We use belief propagation (BP) decoding only on information bits to decode these codes and by doing so the overall complexity of decoding would be log-linear in terms of block size. To improve code performance, information bits are further protected with a polar code. The combined design has low complexity of decoding, small latency and a vanishing bit error rate (BER).

We also prove upper and lower bounds on bit error rate of these algorithms at any SNR and study a combined scheme that splits the information block into b blocks and protects each with some polar code. Decoding moves back and forth between polar and LDPC codes, every time using a polar code of a higher rate. For a sufficiently large constant b and a large block size, this design yields a vanishing BER at any SNR that is arbitrarily close to the Shannon limit of -1.59 dB. This scheme also has very low complexity and decodes m information bits with complexity of order $\mathcal{O}(m \log m)$ per information bit.

In the subsequent chapters of this thesis, we combine polar and low-density parity-check (LDPC) with parity checks of small weight to achieve low latency and low complexity codes for high noise channels. Decoding of this codes also performs several iterations of the belief propagation (BP) algorithm. Partially corrected bits are then passed to a short polar code that uses successive cancellation list (SCL) decoder. The newly corrected bits then serve as the new inputs for the LDPC decoder. For codes of rate less than 0.05, the algorithm performs on a par with a cyclic redundancy check (CRC) aided successive cancellation list (CA-SCL) decoder, while substantially reducing its latency.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The ultimate goal of communication is to transmit information form source to the receiver with high reliability, low complexity, low latency and the least amount of power possible. However, these goals are hard to accomplish due to presence of noise in communication channel. In 1948 Claude E. Shannon [4] showed that reliable data transmission over a noisy channel, described in Figure (1.1), is possible. Shannon coding theorem demonstrated that maximum rate of transmission over a noisy channel is bounded by *channel capacity*, As long as the transmission rate is smaller than channel capacity, there are long codes that can achieve arbitrary low probability of output error. These *random codes* will obtain exponentially declining probability of error when *maximum likelihood* (ML) decoding is implemented but ML decoding is NP-hard [5].

It is an open problem to design the efficient capacity achieving codes with low complexity of encoding and decoding. In order to address this question there were a number of different

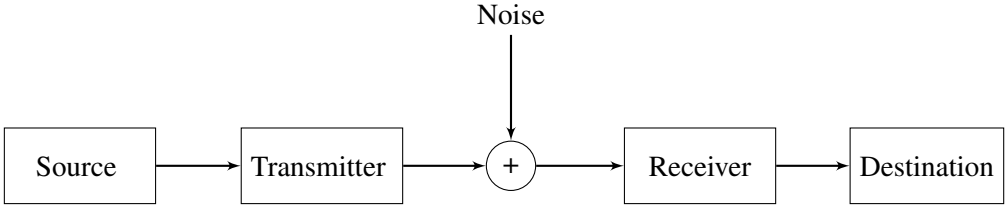

Figure 1.1: A general noisy communication system

codes introduced in the up coming years. In 1950, Hamming introduced Hamming distance [6] and formalized linear codes and lead to introduction of a number of important algebraic codes such as Reed-Muller codes [7, 8], Bose-Chaudhuri-Hocquenghem (BCH) codes [9, 10] and Reed-Solomon (RS) codes [11]. Even though these codes have lots of applications in real life, they did not achieve channel capacity on binary input additive white Gaussian noise (BI-AWGN) channels with low complexity.

In contrast to algebraic codes, probabilistic codes are mostly inspired by Shannon's random codes. Some of famous probabilistic codes are convolutional codes [12], product codes [13], concatenated codes [14], Turbo codes [15], low-density parity-check codes (LDPC) [16, 17], irregular LDPC codes [18, 19] and convolutional LDPC codes [20]. These code are practically important and have the ability of getting close to Shannon limit [21, 22].

Polar codes [23] are theoretically have the ability to achieve channel capacity over huge blocks. This capacity achieving property of polar codes has attracted the world to use this code on a wide variety of topics such as data compression [24, 25], broadcast channels [26, 27], multiple access channels [28, 29], physical layer security [30, 31], and coded modulation [32, 33]. This indicates the significance of polar codes and the need for comparison of any new capacity achieving codes with polar codes in terms of complexity and latency.

Low-capacity scenarios have become increasingly important in narrow-band and wide-band communications. Narrow-band communication is used in the technology of the Internet of Things (IoT) in cellular networks where a huge number of users need to be served [34] and wide-band communication is considered in millimeter-Wave (mmWave) which is crucial for the next generation of cellular networks (5G) [35]. In both of theses scenarios users need to work in very low signal-to-noise-ratios therefore we are interested to explore these schemes and introduce solutions for both asymptotic and practical cases.

Efficient code design protecting data from extreme levels of noise is central to many

low power applications. To efficiently employ Internet of things (IoT), prospective standards [36] are supposed to achieve a 20 dB reduction in *snr* per channel bit (we use notation *snr* for signal-to-noise ratios per input channel bits, while notation SNR will be applied to the output information bits).

There are a few different ways to address this important problem. One prospective approach is to design an efficient (possibly, capacity-reaching) sequence of codes of vanishing code rates. Practically, these codes should also achieve low decoding complexity and rapid decline of error rates. However, currently this problem is far from sound practical solutions.

From the theoretical standpoint, we consider binary linear codes $C(n,k)$ of length $n \to \infty$ and dimension k used on the BSC and BI-AWGN $\mathcal{N}(0,\sigma_n^2)$ with noise power $\sigma_n^2 \to \infty$. To achieve a fixed signal-to-noise ratio $SNR = 1/\left(2\sigma_n^2 R_n\right)$, these codes must have the vanishing code rates R_n that have an order of σ_n^{-2}. Moreover, the fundamental Shannon limit shows that any such code may achieve the vanishing BERs only if $SNR > \ln 2$ (equivalently, this limit corresponds to $10\log_{10}\ln 2 = -1.5917$ dB).

The central problem here is to design a capacity-achieving sequence of codes that have low decoding complexity and a rapidly declining BER. Currently, this problem is far from solution. To date, most existing capacity-achieving codes have code rates R_n that decline exponentially in code dimension m. In turn, this yields an exponential growth in bandwidth and decoding complexity, both proportional to R_n^{-1}.

For example, biorthogonal codes [37] $C(2^{m-1},m)$ achieve the Shannon limit; however, their code rate $R_n = m/2^{m-1}$ declines exponentially in m. By contrast, the output word error rates (WER) of these codes experience very slow decline, which is only polynomial in block-length n. In particular, for the low $SNR \in (\ln 2, 4\ln 2)$, codes $C(2^{m-1},m)$ have word error rate (WER) [38] bounded from above by

$$P_m = \exp\{-m(\sqrt{SNR} - \sqrt{\ln 2})^2\}$$

For a practically important range of $SNR \in [1,2]$ (which gives the range of $[0,3]$ dB), long codes C_m – up to billions of bits – still have very high error rates P_m. This is shown below for $m = 18, 21, 25$ and 30.

SNR (in dBs)	0	0.5	1	1.5	2	2.5	3
$m = 18$: P_{18} (in %)	60.3	39.6	22.1	10.2	3.8	1.1	0.2
$m = 21$: P_{21} (in %)	55.5	34.0	17.2	7.0	2.2	0.51	0.085
$m = 25$: P_{25} (in %)	49.6	27.7	12.3	4.2	1.1	0.19	0.02
$m = 30$: P_{30} (in %)	43.1	21.4	8.1	2.2	0.43	0.05	4E-3

Table 1.1: WER of Biorthogonal Codes

Further analysis shows that concatenations of codes $C(2^{m-1}, m)$ with the outer Reed–Solomon (RS) codes or algebraic geometry (AG) codes [39, 40] still have similar shortcomings, due to the fact that codes $C(2^{m-1}, m)$ should have length n proportional to $\sigma_n^2 \to \infty$. In summary, codes C_m or their concatenations fail to yield acceptable output error rates on the high-noise AWGN channels with SNR of [0,2] dB for the blocks of length $n < 10^8$.

As the second example, consider general RM codes or their bit-frozen sub-codes. Let W_m be a sequence of the binary symmetric channels (BSC$_p$) with transition error probabilities $p_m = (1 - \varepsilon_m)/2$ such that $\varepsilon_m \to 0$ as $m \to \infty$. It is well known that channels W_m yield a sequence of vanishing capacities

$$C_m \sim \varepsilon_m^2 / \ln 4, \quad m \to \infty$$

It was proven in [41, 42] that long low-rate RM codes $RM(m,r)$ of order $r = o(m)$ and length $n = 2^m$ approach the maximum possible code rates C_m on channels W_m under the maximum-likelihood (ML) decoding. Even in this case, code rates R_n decline exponentially as $m^r 2^{-m}$ and require exponential decoding complexity.

Consider also the existing low-complexity algorithms known for RM codes [43–45] or their bit-frozen sub-codes [46]. For low $SNR < 1$ dB, these algorithms yield high error rates above 10^{-3} or require unacceptably large lists under successive cancellation list (SCL) decoding.

Polar codes [23] of rate $R_n \to 0$ that operate under growing noise power $\sigma_n^2 \sim 1/(2SR_n)$ for a fixed SNR $= S$. One construction of such codes is considered in [47]. For $\sigma_n^2 \to \infty$, these codes begin with a growing number $\mu \sim \log_2 \sigma_n^2$ of upgrading channels and employ long repetition codes $B(2^\mu, 1)$ or RM codes $C(2^\mu, m+1)$. This design again results in a rapid complexity increase as $\sigma_n^2 \to \infty$. To advance polar design, it is important to analyze how polar codes of length $n \to \infty$ operate within a vanishing margin $\varepsilon_n \to 0$ to the Shannon limit. One important problem here is to find the minimal length $n(\varepsilon)$ which polar codes approach the Shannon limit within a gap ε. It is known [48] that such a length is a polynomial in ε^{-1}.

For moderate lengths, one efficient construction of [49, 50] concatenates repetition code of length 4 with a (2048,40) polar code. The resulting code has WER of .002 at the SNR of 2 dB and improves the NB-IoT standard [36] by 1 dB. Another recent design [51] yields WER of 0.0007 for the similar parameters. In this thesis, we present algorithms to improve asymptotic and practical performance of low rate designs with low complexity and low latency.

1.1 Contributions of this thesis

1.1.1 Codes for high-noise memoryless channels

We consider codes for channels with extreme noise that emerge in various low power applications such as IoT or sensor networks. To address this case, we design simple LDPC-type codes that have growing dimension m and length $m(m+1)/2$ where $m \to \infty$. These codes can be regarded as a "modulation-type" codes. We use belief propagation (BP) [16, 52] algorithm only on information bits to decode them. We show that this design has the ability to improve the original channel output for any signal-to-noise ratios (SNR) per information bit $SNR > -6$ dB and outperform uncoded modulation for SNRs above -3.7 dB. It also obtains a 3 dB gain over uncoded modulation as SNR grows. The proposed low rate LDPC design has an

overall complexity of order $\mathcal{O}(m^2 \log m)$ and a latency of order $\mathcal{O}(\log m)$ when BP algorithm is implemented in parallel. Similar to uncoded modulation, these codes also exhibit a floor on the output bit error rate (BER) for any m. To improve code performance, information bits are further protected with some polar code of length m. This design also has low complexity of order $\mathcal{O}(m^2 \log m)$, a latency of order $\mathcal{O}(m)$, a vanishing BER of order $\mathcal{O}(\exp\{-m^{1/2}\})$.

Tight lower and upper bounds for the maximum likelihood (ML) and BP algorithm, which are virtually identical to simulation results, are then obtained for BER at any SNR for binary symmetric channel (BSC) and binary-input additive white Gaussian noise (BI-AWGN) channels.

1.1.2 Codes approaching the Shannon limit with polynomial complexity per information bit

We introduce a combined scheme that splits m information bits into b blocks and protects each with some polar code. Decoding moves back and forth between polar and LDPC codes, every time using a polar code of a higher rate. We then present theoretical boundaries for BER of this algorithm with frozen information bits and compare them with simulation results and show that they are practically identical. We then show numerically that for a sufficiently large constant b and $m \to \infty$, this design yields a vanishing BER at any SNR that is arbitrarily close to the Shannon limit of -1.5917 dB. Unlike other existing designs, this scheme has a polynomial complexity of order $\mathcal{O}(m \ln m)$ per information bit. The latency of this capacity achieving design is of order $\mathcal{O}(m)$ which is significantly smaller than that of polar codes.

1.1.3 Combined polar-LDPC design using Gallager ensemble of LDPC codes

We combine polar and LDPC codes to address data correction for various low power applications. We use a combination of Gallager's design [16] and repetition codes to create long

low rate LDPC codes that have parity checks of a small weight w. Decoding first improves the information bits by using the small block of repeated information bits and then performs several iterations of the belief propagation (BP) algorithm that recalculates the information bits only. Partially corrected bits are then passed to a short polar code that uses successive cancellation list (SCL) decoder. The newly corrected bits then serve as the new inputs for an LDPC decoder. For codes of rate less than 0.1, the algorithm performs on a par with a CRC-Aided SCL polar decoder (CA-SCL), while substantially reducing its latency.

1.2 Outline

We start by introducing a low rate LDPC code in Chapter 2 and showcase the abilities of this code combined with repetition code and compare it to uncoded modulation. We then present precise lower and upper bounds for BER of this design. Then we combine this design with polar precoding and show the simulation results of this low rate and low latency code and compare it with state-of-the-art polar-repetition codes. In Chapter 3, we introduce a back and forth polar-LDPC scheme with the ability of approaching the the Shannon limit of -1.59 dB. In Chapter 4 we showcase an extension of these codes that uses parity check of higher weight and has the ability of introducing practical code design under moderate block lengths. The performance of this code is on a par with the best existing codes such as CA-Polar codes and hybrid non-binary repeated polar codes with much smaller latency. Finally, in Chapter 5 we conclude this work and discuss the potential future work.

Chapter 2

Codes for high-noise memoryless channels

2.1 Introduction

In this chapter, we introduce a low rate and low latency LDPC code and we discuss its properties, encoding and decoding algorithm. We present bounds for bit error rate (BER) of maximum likelihood (ML) and belief propagation (BP) algorithm and prove them on BSC and AWGN channels. Then we combine repetition codes with this low rate design and show that this design can outperform uncoded modulation for $SNR > -3$ dB. We also use polar codes as a precoding for this modulation LDPC code and present a low rate and low latency scheme that has the ability of outperforming polar-repetition codes with similar complexity and much smaller latency.

2.2 Basic construction

Our basic code - which we denote C_m - has generator matrix $G_m = [I_m|J_m]$, where I_m is an $m \times m$ identity matrix and J_m is an $m \times \binom{m}{2}$ matrix that includes all columns of weight 2. Clearly, $n = \binom{m+1}{2}$ and $k = m$. Let $a_{(s)}$ be any codeword generated by s rows of G_m. Note that every row in J_m has weight $m-1$, every two rows have a single common 1, and every $s \geq 2$ rows have $\binom{s}{2}$ common 1s. Any codeword $a_{(s)}$ that has weight s in I_m has overall weight

$$w_s = ms - 2\binom{s}{2} = s(m-s+1) \tag{2.1}$$

Thus, code C_m has distance m, which is achieved if $s = 1, m$. All other codewords have weight $2(m-1)$ or more. Note also that code C_m represents a heavily truncated Hadamard code [53] that leaves only positions of weight 1 and 2 and excludes other $m-2$ spherical layers formed by the full space of positions E_2^m. You can see the generator matrix G_4 in (2.2).

$$G_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{2.2}$$

2.3 Belief propagation decoding on BSC and AWGN channels

Tanner proposed to represent codes as bipartite graphs and to visualize iterative decoding as a message-passing algorithm on such a graph [54, 55]. Kim and Pearl introduced the belief propagation algorithm [56, 57] to solve statistical inference problems [58]. LDPC codes can also be represented as a bipartite graph and belief propagation algorithm can be used to decode them. Here we implement BP only on information bits.

2.3.1 Decoding on binary symmetric channels

In this section we consider communication on a BSC, described in Figure (2.1), and present their belief propagation decoding. Let $[i, j] = [j, i]$ denote code positions in G_m, where $0 \leq i \neq j \leq m$. Encoder aG_m receives a string $a = (a_{0,1}, ..., a_{0,m})$ of m information bits and adds $\binom{m}{2}$ parity bits $a_{1,2}, ..., a_{m-1,m}$ such that $a_{i,j} = a_{0,i} + a_{0,j}$. Note that encoding has complexity $\mathcal{O}(n)$. Let code C_m of rate $R = 2/(m+1)$ be used on a BSC channel described in Figure (2.3) . We use a map $\{0,1\} \to \{\pm 1\}$ for each transmitted symbol $a_{i,j}$, describe in Figure (2.2), where $0 \leq i \neq j \leq m$. Then the parity checks $a_{i,j}$ form the real-valued products

$$a_{0,i} = a_{0,j} a_{i,j} \tag{2.3}$$

It is important to note that when we send antipodal signals $s_i = \pm 1$ we can claim that error probability $p = (1 + u_0)/2$ where $u_0 \to 0$. In this thesis we assume that the communication rate is small, $r \to 0$ and because of that we can estimate $u_0 = \sqrt{c_b/m}$ where $c_b = 8SNR/\pi$. It can be shown that $\lim_{x \to 0} Q(x) = 0.5 - x/\sqrt{2\pi}$ where $Q(x)$ is:

$$n(x) = (2\pi)^{-1/2} \exp\{-x^2/2) \tag{2.4}$$
$$Q(x) = \int_x^{\infty} n(y) \partial y$$

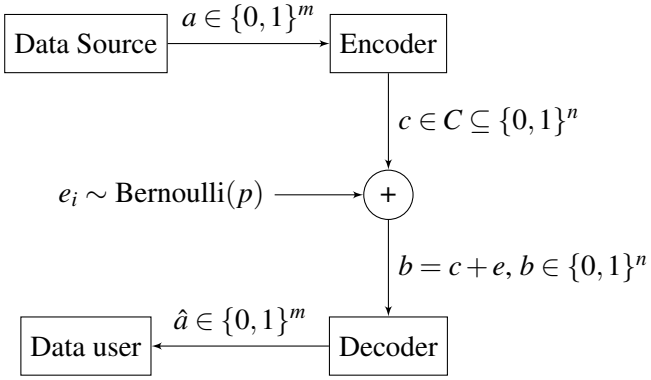

Figure 2.1: Communication system on a binary symmetric channel

$$\begin{cases} 0 \rightarrow 1 \\ 1 \rightarrow -1 \end{cases}$$

Figure 2.2: Antipodal Signal Representation of Binary Data

Our decoding algorithm Ψ_{BSC} performs several steps of belief propagation. However, unlike conventional algorithms, we estimate only information bits $a_{0,i}$. In the first iteration our estimate comes from the BSC channel but after the fist iteration this random variable will have a Gaussian distribution due to central limit theorem and independence of information bits and parity check bits. After the first iteration we assume that by removing intrinsic information in each step and having a large size, $m \rightarrow \infty$, there is a weak dependence between theses estimates. A number of sources show that summation of m dependent random variables can have a Gaussian distribution [59–64]. Our conjecture is that similar conditions hold for our partial log-likelihood random variables to the conditions that are mentioned in [65, 66].

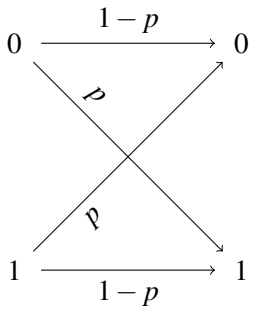

Figure 2.3: Binary symmetric channel

In our decoding algorithm we assume that the received parity checks have probability offsets $u_{i,j} = b_{i,j}u_0$, where their sign is the same as the received bits, $b_{i,j}$, of the BSC. Given some received bit $b_{i,j}$, an input $a_{i,j} = 1$ has posterior probability

$$q_{i,j} \triangleq \Pr\{1 \mid b_{i,j}\} = (1 + b_{i,j}u_0)/2.$$

Now we can describe the soft decision belief propagation decoding algorithm for a total of L iterations as follows:

For all $i, j \in \{1, ..., m\}$ and $j \neq i$:

A. Derive quantities $u_{i|\ell+1}(j) = u_{i,j} u_{i|\ell}(j)$

and $h_{i|\ell+1}(j) = 2\tanh^{-1}\left[u_{i|\ell+1}(j)\right]$.

B. Derive quantities $h_{i|\ell+1} = \sum_j h_{i|\ell+1}(j)$

and $h_{j|\ell+1}(i) = h_{i|\ell+1} - h_{i|\ell+1}(j)$

C. If $\ell < L$, find $u_{i|\ell+1}(j) = \tanh(h_{i|\ell+1}(j)/2)$.

Go to A with $\ell := \ell + 1$. If $\ell = L$:

estimate BER $\tau_L = \frac{1}{m}\sum_i \Pr\{h_{i|L} < 0\}$;

output numbers $h_{i|L}$ and $a_i = \text{sign}\,(h_{i|L})$. (2.5)

2.3.2 Decoding on AWGN Channels

Let us assume that the code C_m is transmitted over an AWGN channel with pdf. $\mathcal{N}(0, \sigma^2)$, described in 2.4, and constant $SNR = \left(2\sigma^2 R\right)^{-1}$ per information bit. In the sequel, it will be more convenient for us to use a constant $c = 4(SNR)$. We use the same mapping described in (2.2) for each transmitted symbol $a_{i,j}$, where $0 \leq i \neq j \leq m$. Then the parity checks $a_{i,j}$ form the real-valued products seen in (4.3). Let an all-one codeword 1^n be transmitted. Then the received symbols $y_{i,j} \equiv y_{j,i}$ form independent Gaussian random variables (r.v.) $\mathcal{N}(1, \sigma^2)$. We will use rescaled r.v. $z_{i,j} = \delta y_{i,j}$,where $\delta = 1/\left(\sigma^2 + 1\right) = c/\left(m + c + 1\right)$. It is easy to verify that this scaling gives power moments $x_0 = E(z_{i,j})$ and $\sigma_0^2 = E(z_{i,j}^2)$ such that

$$x_0 = \sigma_0^2 = \delta \tag{2.6}$$

Given some $z_{i,j}$, an input $a_{i,j} = 1$ has posterior probability:

$$q_{i,j} \triangleq \Pr\{1 \mid z_{i,j}\} = 1/(\exp(-2z_{i,j}) + 1). \tag{2.7}$$

Decoding algorithm $\Psi_{soft}(z)$ described below employs two closely related quantities, the log-likelihoods (l.l.h.) $h_{i,j}$ and the "probability offsets" $u_{i,j}$:

$$h_{i,j} = \ln[q_{i,j}] - \ln[1 - q_{i,j}] = 2z_{i,j} \tag{2.8}$$

$$u_{i,j} = 2q_{i,j} - 1 = \tanh(z_{i,j}) \tag{2.9}$$

Given the offsets $u_{0,j}$ and $u_{i,j}$ in (4.3), it is easy to verify that symbol $a_{0,i}$ has offset $u_{0,i} = u_{0,j}u_{i,j}$. Also, $u_{i,j} = \tanh(z_{i,j}) = \tanh(h_{i,j}/2)$. Function $\tanh(x)$ has derivatives $\tanh'(0) = 1$ and $\tanh''(0) = 0$ at $x = 0$. Thus, for the vanishing values of $z_{i,j} \to 0$,

$$u_{i,j} = z_{i,j} + o(z_{i,j}^2) = h_{i,j}/2 + o(h_{i,j}^2) \tag{2.10}$$

Algorithm Ψ_{soft} performs several steps of belief propagation. Unlike conventional algorithms, we estimate only information bits $a_{0,i}$. We will show that Ψ_{soft} requires $L \sim \ln m / \ln c$ iterations to achieve the best performance. For every step $\ell = 1, ..., L$ and every symbol $a_{0,i}$, consider its

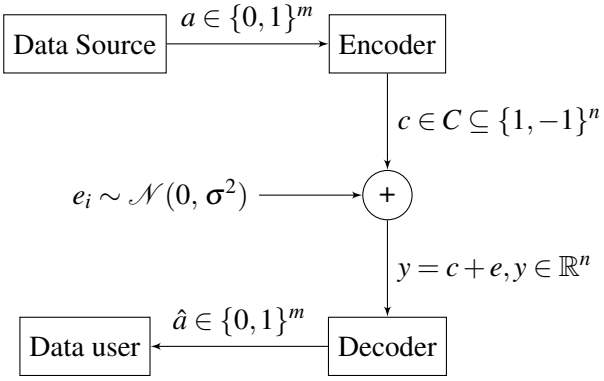

Figure 2.4: Communication system on an AWGN channel

j-th parity check $a_{0,i} = a_{0,j}a_{i,j}$ of (4.3). To re-evaluate $a_{0,i}$, we introduce the offset $u_{i|\ell}(j)$ of the symbol $a_{0,j}$ used in this parity check. Then the estimate $u_{i,j}u_{j|\ell}(j)$ re-evaluates symbol $a_{0,i}$ via the product $a_{0,j}a_{i,j}$. We then obtain the l.l.h. $h_{i|\ell+1}(j)$ of the j-th parity check using transforms (2.8) and (2.9). Next, the sum of l.l.h. $h_{i|\ell+1}(j)$ gives the compound estimate $h_{i|\ell+1}$ of the symbol $a_{0,i}$. Finally, we derive the partial l.l.h. $h_{j|\ell+1}(i)$ of the symbol $a_{0,i}$ that will be used in the next round to estimate $a_{0,j}$ via its i-th parity check $a_{0,j} = a_{0,i}a_{i,j}$. This excludes the intrinsic information $h_{i|\ell+1}(j)$ that symbol $a_{0,j}$ already used in round ℓ. Our recalculations begin with the original estimates $u_{i|0}(j) \triangleq u_{0,i}$. Round ℓ of Ψ_{soft} is done as follows.

For all $i, j \in \{1, ..., m\}$ and $j \neq i$:

A. Derive quantities $u_{i|\ell+1}(j) = u_{i,j}u_{i|\ell}(j)$

and $h_{i|\ell+1}(j) = 2\tanh^{-1}\left[u_{i|\ell+1}(j)\right]$.

B. Derive quantities $h_{i|\ell+1} = \sum_j h_{i|\ell+1}(j)$

and $h_{j|\ell+1}(i) = h_{i|\ell+1} - h_{i|\ell+1}(j)$

C. If $\ell < L$, find $u_{i|\ell+1}(j) = \tanh(h_{i|\ell+1}(j)/2)$.

Go to A with $\ell := \ell + 1$. If $\ell = L$:

estimate BER $\tau_L = \frac{1}{m}\sum_i \Pr\{h_{i|L} < 0\}$;

output numbers $h_{i|L}$ and $a_i = \text{sign}\,(h_{i|L})$. (2.11)

To estimate the complexity of Ψ_{soft}, note that Step A uses at most n multiplications and n two-way conversions $u \leftrightarrow h$. Step *B* calculates the sums $h_{i|\ell+1}$ using m operations for each i. It also requires $2n$ operations to derive the residual sums $h_{i|\ell+1}(j)$ and their offsets $u_{i|\ell+1}(j)$ for all pairs i, j. Then the overall complexity has the order $\mathscr{O}(n)$ for every iteration ℓ. Assuming that we have $L = \mathscr{O}(\log m)$ iterations, we obtain complexity $\mathscr{O}(n \log n)$.

2.4 Lower bounds for BER of codes C_m on AWGN channels

We will now study the output BER of codes C_m. We first show that long codes C_m fail to achieve BER $P_c \to 0$ for any $SNR = c/4$ even if they employ ML decoding. This is similar to the uncoded modulation (UM). Assume that an all-one codeword 1^n (formerly, a 0^n codeword in $\mathbb{F}_2^n$) is transmitted and $z = (z_{i,j})$ is received. Consider the sets of positions $I_0 = (0, j \,|\, j \neq 0, 1)$ and $I_1 = (0, j \,|\, j \neq 0, 1)$. For any vector z, we will define the corresponding r.v.

$$Y_0 = \sum\nolimits_{j \neq 0,1} z_{0,j}, \;\; Y_1 = \sum\nolimits_{j \neq 0,1} z_{1,j}$$

Below we use asymptotic probability density functions (pdf) as $m \to \infty$. Then r.v. $z_{i,j}$ have asymptotic pdf $\mathcal{N}(\delta, \delta)$. It is also easy to verify that r.v. $Z_i = \sum_j z_{i,j}$, Y_0, and Y_1 have asymptotic pdf $N(c,c)$. Code-words of minimum weight in C_m include m generator rows $g^{(p)}$, where $p = 1, ..., m$, of the generator matrix G_m and their sum $g^{(0)} = g^{(1)} + ... + g^{(m)}$. Under ML decoding, any two-word code $\{1^n, g^{(p)}\}$, has BER

$$P_c = \Pr\{Y_1 < 0\} = Q\left(\frac{m\delta - \delta}{\sqrt{m(\delta - \delta^2)}}\right) \sim Q\left(\sqrt{c}\right) \tag{2.12}$$

Here we write $f(m) \sim g(m)$ if $\lim f(m)/g(m) = 1$ as $m \to \infty$. Similarly, we use notation $f(m) \gtrsim g(m)$ if $\lim f(m)/g(m) \geq 1$.

Theorem 1 *Let codes C_m be used on an AWGN channel with an SNR of $c/4$ per information bit. Then for $m \to \infty$, ML decoding of codes C_m has BER*

$$p_{ML}(c) \gtrsim 2P_c(1 - P_c) = 2Q\left(\sqrt{c}\right) - 2Q^2\left(\sqrt{c}\right) \tag{2.13}$$

Proof. Without loss of generality, we consider BER of symbol $a_{0,1}$. In essence, we prove that

ML decoding gives $a_{0,1} = -1$ if so does one of the codes $\{1^n,\, g^{(p)}\}$ for $p = 0, 1$. All received vectors z form four disjoint subsets $U = A, B, C, D$, where

$$A = \{z \,|\, Y_0 < 0,\, Y_1 > 0\},\; B = \{z \,|\, Y_0 > 0,\, Y_1 < 0\} \tag{2.14}$$

$$C = \{z \,|\, Y_0 > 0,\, Y_1 > 0\},\; D = \{z \,|\, Y_0 < 0,\, Y_1 < 0\} \tag{2.15}$$

Clearly, $\Pr\{A\} = \Pr\{B\} = P_c(1 - P_c)$. We will prove that $p_{ML}(c) \gtrsim \Pr\{A\} + \Pr\{B\}$.

Two vectors $g^{(p)}$, $p = 0, 1$, have supports $J_p = \{(p, j)\}$, where $j \in \{0, ..., m\} \setminus \{p\}$. For any z, consider bitwise products $g^{(p)}z$ that flip symbols of z on the supports J_p. Then

$$g^{(0)}A = C,\; g^{(1)}A = D,\; g^{(0)}B = D,\; g^{(1)}B = C \tag{2.16}$$

Let z be decoded into some $a(z) \in C_m$ and let $a_{0,1}(z)$ be the first symbol of $a(z)$. We decompose each set U into

$$U_+ = \{z \in U : a_{0,1}(z) = 1\},\; U_- = \{z \in U : a_{0,1}(z) = -1\}$$

Note that $a\left(g^{(p)}z\right) = g^{(p)}a(z)$. Then

$$g^{(0)}A_+ = C_-,\;\; g^{(1)}A_+ = D_- \tag{2.17}$$

$$g^{(1)}B_+ = C_-,\;\; g^{(0)}B_+ = D_-$$

Conditions (2.16) and (2.16) show that maps $g^{(0)}$ and $g^{(1)}$ flip full sets U and there subsets U_+ and U_-.

In the next step, we remove the first symbol $a_{0,1}$ from each vector z and obtain four sets $U' = A', B', C', D'$ with a punctured symbol $a_{0,1}$. Let U'_+ and U'_- denote the punctured subsets of

U_+ and U_-. Below we show in Lemma *2* that the maps $g^{(0)}$ and $g^{(1)}$ cannot reduce the probability of the sets $A'+B'$. Namely,

$$\Pr\{C'_-\}+\Pr\{D'_-\}\geq 2\Pr\{A'_+\} \tag{2.18}$$

$$\Pr\{C'_-\}+\Pr\{D'_-\}\geq 2\Pr\{B'_+\} \tag{2.19}$$

Finally, consider $p_{ML}(c)\equiv\sum_U \Pr\{U_-\}$. We then prove in Lemma *3* that removing one bit $a_{0,1}$ has immaterial impact on $\Pr\{U\}$ as $m\to\infty$, so that $\Pr\{U\}\sim\Pr\{U'\}$. Then

$$p_{ML}(c)=\sum\nolimits_U \Pr\{U_-\}\sim\sum\nolimits_U \Pr\{U'_-\}$$

We can now use (2.18) and (2.19), which gives

$$\begin{aligned} p_{ML}(c) &\sim \Pr\{A'_-\}+\Pr\{B'_-\}+\Pr\{C'_-\}+\Pr\{D'_-\} \\ &\geq \Pr\{A'_-\}+\Pr\{B'_-\}+\Pr\{A'_+\}+\Pr\{B'_+\} \\ &= \Pr\{A'\}+\Pr\{B'\} \end{aligned}$$

Thus, we obtain (2.13). ■

Lemma 2 *Punctured sets $U'=A',B',C',D'$ satisfy inequalities (2.18) and (2.19).*

Proof. Recall that 1^n is the transmitted vector. In this case, the set C has the highest probability among all sets U, while D is the least likely. We now can establish stronger conditions. In essence, we show that the transition $A\mapsto C$ (or $B\mapsto C$) produces a greater increase $\Pr(C)-\Pr(A)$ than the drop $\Pr(A)-\Pr(D)$ required in transition $A\mapsto D$.

We say that any $x\in A',B'$ is a (θ,ρ) vector if $Y_0=\theta,\ \ Y_1=\rho$. According to (2.14), any $x\in A$ has $\theta<0$, $\rho>0$, whereas it is vice versa for $x\in B$.

Recall that r.v. Y_0, Y_1 have asymptotic pdf $\mathcal{N}(c,c)$. (The exact pdf is $\mathcal{N}(c\lambda, c\lambda - c\delta\lambda)$). Consider (θ,ρ)-vectors $x \in A$. On the subset $I_0 = \{(0,j)\}$, these vectors x have pdf

$$p(\theta) \sim (2\pi c)^{-1/2} e^{-(\theta-c)^2/2c}$$

For any x, the transform $g^{(0)}x$ only flips symbols $x_{0,j}$ thus replacing pdf. $p(\theta)$ on the set I_0 with $p(-\theta)$. This gives the ratio

$$r(\theta) = p(-\theta)/p(\theta) = e^{-2\theta}$$

The other transform $g^{(1)}x$ of any (θ,ρ)-vector x flips symbols $x_{1,j}$. Then we obtain the ratio

$$r(\rho) = p(\rho)/p(-\rho) = e^{-2\rho}$$

Now we consider two vectors from A_+, namely, $x = x(\theta,\rho)$ and $y = y(-\rho,-\theta)$. Then $g^{(0)}x \in C$ and $g^{(1)}x \in D$. The same inclusion holds for vector y. Also, both vectors x and y have the same pdf $p(x) = p(y) = p$ generated on the sets I_0 and I_1, since both r.v. Y_0 and Y_1 have the same distribution. We can now estimate the total pdf of vectors $g^{(p)}x$ and $g^{(p)}y$ as follows

$$p\left(g^{(0)}x\right) + p\left(g^{(1)}x\right) = \left(e^{-2\theta} + e^{-2\rho}\right)p$$
$$p\left(g^{(0)}y\right) + p\left(g^{(1)}y\right) = \left(e^{2\theta} + e^{2\rho}\right)p$$

Since $\exp\{-2a\} + \exp\{2a\} \geq 2$ for any a, we can reduce the latter equalities to

$$2\sum\nolimits_{p=1,2} p\left(g^{(p)}x\right) + p\left(g^{(p)}y\right) \geq 4p$$

This immediately leads to inequality (2.18). Inequality (2.19) is identical if we replace A_+ with B_+. Other inequalities of the same kind can be obtained if we consider subsets A', B' (or A'_-, B'_-).

■

We now prove that removing position $(0,1)$ is immaterial for our proof.

Lemma 3 *Any set U and its one-bit puncturing U' satisfy asymptotic equality* $\Pr\{U\} \sim \Pr\{U'\}$.

Proof. Note that r.v. $z_{0,1}$ has pdf $\mathcal{N}(\delta,\delta)$, where $\delta \sim c/m \to 0$ as $m \to \infty$, whereas r.v. Y_0 (or Y_1) has pdf $\mathcal{N}(c,c)$. Let $r = \sqrt{c/m}\ln m$ and $r' = r\ln m$. Then with probability tending to 1, we have the following conditions:

$$z_{0,1} \in [-r, r],\ Y_0 \notin [-r', r'] \tag{2.20}$$

Thus, $\Pr\{z_{0,1}/Y_0 \to 0\} \to 1$ as $m \to \infty$. Now we see that equalities $\Pr\{U\} \sim \Pr\{U'\}$ hold for any set U or U_+ or U_- as $m \to \infty$. ■

2.5 Probabilistic bounds for BP decoding on AWGN channels

Our next goal is to study BP algorithm Ψ_{soft} of (4.7). We first slightly expand on our notation. We say that events U_m hold with high probability P_m if $P_m \to 1$ as $m \to \infty$. Let $\mathcal{N}(a,b)$ denote the pdf of a Gaussian r.v. that has mean a, variance b, and the second power moment a^2+b. Consider a sequence of Gaussian r.v. x_m that have pdf $\mathcal{N}(a,b_m)$, where $b_m = b(1+\theta_m)$, $b>0$ is a constant, and $\theta_m \to 0$ as $m \to \infty$. Consider also any sequence t_m such that $t_m = o(\theta_m^{-1/2})$. Then $\Pr\{x_m > t_m\} \sim Q((t_m - a)\, b^{-1/2})$ and we write $\mathcal{N}(a,b_m) \sim \mathcal{N}(a,b)$.

Consider also r.v. $z_{i,j}$ that has pdf asymptotic $\mathcal{N}(\delta,\delta)$ as $m \to \infty$. Then restriction (2.20) shows that with high probability $z_{i,j} \to 0$. Then equality (2.10) shows that $u_{i,j} = z_{i,j} + o(z_{i,j}^2) \sim z_{i,j}$. Thus, we will replace r.v. $u_{i,j}$ in algorithm Ψ_{soft} with $z_{i,j}$.

To derive analytical bounds, we will slightly simplify algorithm Ψ_{soft} and assign the same value $h_{i|\ell+1}(j) = h_{i|\ell+1}$ for all j instead of different assignments $h_{i|\ell+1}(j) := h_{i|\ell+1} - h_{j|\ell+1}(i)$. It can be shown that this change is immaterial for our asymptotic analysis. It also makes very negligible changes even on the short blocks C. The simplified version of the algorithm Ψ_{soft} - described below - begins with the initial assignment $u_{j|0} = z_{0,j}$ in round $\ell = 0$. We will perform $L = 2\ln m/\ln c$ rounds. In round ℓ, Ψ_{soft} proceeds as follows.

$$\boxed{\begin{array}{l} A.\ \text{Derive quantities } u_{i|\ell+1}(j) = z_{i,j}u_{j|\ell} \\ \quad \text{and } h_{i|\ell+1}(j) = 2\tanh^{-1}\left[u_{i|\ell+1}(j)\right]. \\ B.\ \text{Derive quantities } h_{i|\ell+1} = \sum_j h_{i|\ell+1}(j) \\ C.\ \text{If } \ell < L,\ \text{find } u_{i|\ell+1} = \tanh(h_{i|\ell+1}/2). \\ \quad \text{Go to A with } \ell := \ell+1.\ \text{If } \ell = L: \\ \quad \text{estimate BER } \tau_L = \frac{1}{m}\sum_i \Pr\{h_{i|L} < 0\}; \\ \quad \text{output numbers } h_{i|L} \text{ and } a_{0,i} = \text{sign}\,(h_{i|L}). \end{array}} \tag{2.21}$$

To derive analytical bounds, what we need to show is that $z_{i,j}u_{j|\ell}$ for different values of j are weakly dependant. We know that $z_{i,j}$ for different values of j are i.i.d random variables but $u_{j|\ell}$ are not necessarily independent from each other. Namely, we call r.v. $\xi_1, ..., \xi_m$ weakly dependent if for $m \to \infty$, we have asymptotic equality

$$E(\xi_i \,|\, \xi_{j_1}, ..., \xi_{j_b}) \to E(\xi_i) \tag{2.22}$$

For any constant b, index i, and any subset $J = \{j_1, ..., j_b\}$ such that $i \notin J$. In particular, we will assume that the conditional moment $E(h_{i|\ell+1} \,|\, h_{j_1|\ell}, ..., h_{j_b|\ell})$ tends to the unconditional moment $E(h_{i|\ell+1})$. This assumption does not necessarily hold if b is a growing number. However, in our

case, r.v. $h_{i|\ell+1}$ includes $m-1$ different summands $h_{i|\ell+1}(j)$ for all $j \neq i$. On the other hand, only one related term $h_{j|\ell}(i)$ is included in each sum $h_{j|\ell}$ for any $j \in J$. (Both terms include the same factor $u_{i,j}$ used to evaluate symbols $a_{0,i}$ and $a_{0,j}$ in parity check (4.3)). We further assume that the sums of weakly dependent r.v. satisfy the central limit theorem (CLT). This has been proven in many settings [59–64], where "the future" r.v. have vanishing connection to the past (or "distant" past). In this case, our conjecture is that similar conditions to [65, 66] will hold for partial log-likelihood ratio random variables. The above assumption is also corroborated by the simulation results, which essentially coincide with the theoretical bounds derived below (see Figure 3.2, in particular).

Our goal is to derive BER $P_{soft}(c) = \lim \tau_{L}$ for Ψ_{soft} as $L, m \to \infty$. Given $c > 0$, consider the equation

$$x = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \tanh(t\sqrt{xc}) e^{-(t-\sqrt{xc})^2/2} dt \tag{2.23}$$

In Lemma 8, we will show that for $c \leq 1$ equation (2.23) has a single root $x = 0$. For $c > 1$, (2.23) has the root $x = 0$ and two other roots x_* and $-x_*$, where $x_* \in (0, 1)$.

For any $\ell = 0, 1, ..., L$ and any $m \to \infty$, we introduce parameter $c_\ell = c^{(\ell+1)/2}$. We then derive probabilities P_ℓ using recursion $P_{\ell+1} = (1 - P_\ell) S_\ell + P_\ell T_\ell$, where

$$S_\ell = (2\pi)^{-1/2} \int_{-\infty}^{\infty} Q(c_\ell t) e^{-(t-c_\ell)^2/2} dt \tag{2.24}$$

$$T_\ell = (2\pi)^{-1/2} \int_{-\infty}^{\infty} Q(c_\ell t) e^{-(t+c_\ell)^2/2} dt \tag{2.25}$$

and $P_0 = Q(\sqrt{c})$. For any ℓ, probabilities P_ℓ depend on c only. P_ℓ is representative of having a negative average for log-likelihood ratio random variables given the all-one codeword was sent at iteration ℓ. We will also show that quantities P_ℓ converge exponentially fast as $\ell \to \infty$. Let $P_\infty = \lim_{\ell \to \infty} P_\ell$. We can now establish the asymptotic value of BER as $m \to \infty$.

Theorem 4 *Let codes C_m be used on an AWGN channel with an SNR $c/4$ per information bit. For $m \to \infty$ and $c \leq 1$, algorithm Ψ_{soft} has BER $P_{soft}(c) \to 1/2$. For $c > 1$,*

$$P_{soft}(c) \sim (1 - P_\infty)\, Q\left(\sqrt{x_* c}\right) + P_\infty (1 - Q\left(\sqrt{x_* c}\right)) \tag{2.26}$$

In Figure 2.6 of this section, we will plot analytical bound (2.26) along with simulation results and the lower bound (2.13) of ML decoding. We will see that all three bounds of Figure 2.6 give very tight approximations.

We begin the proof of Theorem 4 with Lemma 5. Here we analyze the sums of r.v. z_j that have asymptotic pdf $\mathcal{N}(\delta, \delta)$ with a small bias $\delta \to 0$.

Lemma 5 *Consider m independent r.v. $z_1, ..., z_m$ with pdf $\mathcal{N}(\delta, \delta)$, where $\delta \sim c/m$. Let $Z = \sum_j z_j$ and $Y = \sum_j z_{i.j}^2$. Then for $m \to \infty$,*

$$E\,(Z\,|\,Y) \sim E\,(Z) \sim c \tag{2.27}$$

Proof. Consider r.v. $\varepsilon_j = z_j - \delta$ that has pdf $\mathcal{N}(0, \delta)$. Let $R = \sum_j \varepsilon_j^2$. This *r.v.* has $\aleph^2$ distribution that tends to $\mathcal{N}(c, 2\delta c)$ as $m \to \infty$. Next, note that r.v. z_j^2 and ε_j^2 are equivalent with high probability. Indeed,

$$z_j^2 = \varepsilon_j^2 + 2\delta\varepsilon_j + \delta^2 \sim \varepsilon_j^2 \tag{2.28}$$

Here with high probability we have two events. First, $\varepsilon_j^2 \geq \sqrt{\delta}/\ln m$, whereas the terms $\left|\delta\varepsilon_j\right|$ and δ^2 are bounded from above by $\delta^{3/2} \ln m = o\left(\sqrt{\delta}/\ln m\right)$. Thus, $z_j^2 \sim \varepsilon_j^2$ and $Y \sim R$ as $m \to \infty$. In turn, this implies that r.v. Y_i has asymptotic pdf $\mathcal{N}(c, 2\delta c)$.

To prove (2.27), we now may consider unbiased r.v. ε_j and prove asymptotic equality

$$E\left(\sum_j \varepsilon_j\,|\,R\right) \sim E\left(\sum_j \varepsilon_j\right) = 0 \tag{2.29}$$

Consider any subset $\mathscr{S}$ of 2^m unbiased vectors $(\pm\varepsilon_1,...,\pm\varepsilon_m)$ that give the same sum $R=\sum_j \varepsilon_j^2$. Then asymptotic equality (2.29) holds for each subset $\mathscr{S}$, which proves Lemma 5. ■

To prove Theorem 4, we will first study r.v. $u_{i\,|\,\ell}$ and their average *power* moments

$$x_\ell = E\sum\nolimits_i \left(u_{i\,|\,\ell}/m\right) \tag{2.30}$$

$$\sigma_\ell^2 = E\sum\nolimits_i \left(u_{i\,|\,\ell}^2/m\right) \tag{2.31}$$

Then r.v. $u_\ell=\sum_i u_{i\,|\,\ell}/m$ has power moments x_ℓ and σ_ℓ^2/m (here we assume that r.v. $u_{i\,|\,\ell}$ are weakly dependent).

In the following statements (Lemmas 6-8 and Theorem 4), we will show that r.v. u_ℓ undergo two different processes as $\ell\to\infty$. In the initial iterations $\ell=1,...,$ r.v. u_ℓ take vanishing values with high probability as $m\to\infty$. In these iterations, they also may take multiple random walks across the origin. For $c<1$ and $\ell\to\infty$, r.v. u_ℓ converge to 0. By contrast, for $c>1$, r.v. u_ℓ gradually move away from the origin in opposite directions, albeit with different probabilities. In the process, r.v. u_ℓ cross 0 with the rapidly declining probabilities as $\ell\to\infty$. They approach two end points, x_* and $-x_*$ with probabilities $1-P_\infty$ and P_∞, respectively, and converge to these points after $\ell\gtrsim \ln m/\ln c$ iterations. At this point, any r.v. $u_{i\,|\,\ell}$ (that represents a specific bit i) has BER of $Q\left(\sqrt{x_*c}\right)$ and $1-Q\left(\sqrt{x_*c}\right)$. This constitutes bound (2.26).

We first derive how quantities x_ℓ and σ_ℓ^2 change in consecutive iterations. Let $\sigma>0$ and $-\sigma\le x\le\sigma$. Below we use two functions $F_c(x,\sigma)$ and $G_c(x,\sigma)$ that are related to $E(u_{i|\ell+1})$ and $E(u_{i|\ell+1}^2)$

$$F_c(x,\sigma) = (2\pi)^{-1/2}\int_{-\infty}^{\infty} \tanh\left(\sigma t\sqrt{c}\right) e^{-(t-x\sqrt{c}/\sigma)^2/2}dt \tag{2.32}$$

$$G_c(x,\sigma) = (2\pi)^{-1/2}\int_{-\infty}^{\infty} \tanh^2\left(\sigma t\sqrt{c}\right) e^{-(t-x\sqrt{c}/\sigma)^2/2}dt \tag{2.33}$$

Lemma 6 *Let r.v. $u_{i|\ell}$, $i=1,..,m$, have average power moments x_ℓ and σ_ℓ^2 of (2.30) and (2.31). Then any r.v. $u_{i|\ell+1}$ has conditional power moments*

$$E\left(x_{\ell+1}\,|\,x_\ell,\sigma_\ell\right)=F_c\left(x_\ell,\sigma_\ell\right) \tag{2.34}$$

$$E\left(\sigma_{\ell+1}^2\,|\,x_\ell,\sigma_\ell\right)=G_c\left(x_\ell,\sigma_\ell\right) \tag{2.35}$$

Proof. Below we consider r.v. $z_{i,j}$, $Z_i=\sum_j z_{i,j}$ and $Y_i=\sum_j z_{i,j}^2$. The proof of Lemma 5 shows that these r.v. have pdfs $\mathcal{N}(\delta,\delta)$, $\mathcal{N}(c,c)$, and $\mathcal{N}(c,2\delta c)$, respectively. For $m\to\infty$, we will use three restrictions, all of which hold with high probability. Firstly, $\left|z_{i,j}\right|\le\Delta$, where $\Delta=2\sqrt{\delta}\ln m\to 0$. Indeed,

$$\Pr\left\{\left|z_{i,j}\right|>\Delta\right\}\le 2Q(2\ln m-\sqrt{\delta})=m^{-2\ln m+o(1)} \tag{2.36}$$

Also,

$$c-\sqrt{c\ln m}\le Z_i\le c+\sqrt{c\ln m} \tag{2.37}$$

$$Y_i\in(c-\Delta_1,c+\Delta_1),\ \Delta_1=m^{-1}c\ln m \tag{2.38}$$

Since $z_{i,j}\to 0$ for all i,j, algorithm Ψ_{soft} can use the following approximations

$$u_{i|\ell+1}(j)=u_{i,j}u_{j|\ell}\sim z_{i,j}u_{j|\ell} \tag{2.39}$$

$$h_{i|\ell+1}(j)=2\tanh^{-1}\left[z_{i,j}u_{j|\ell}\right]\sim 2z_{i,j}u_{j|\ell} \tag{2.40}$$

Here we assume that r.v. $z_{i,j}$ and $u_{j|\ell}$ are "weakly dependent". Indeed, any estimate of $u_{j|\ell}$ includes $m-1$ terms and only one term includes r.v. $z_{i,j}$. We then fix the sums $Z_i=\sum_j z_{i,j}$ and

consider conditional r.v. $z_{i,j}u_{j|\ell}|Z_i$. Given restrictions (2.37) and (2.38) we obtain the moments

$$E\left(z_{i,j}u_{j|\ell}|Z_i\right) = E\left(z_{i,j}\right)E(u_{j|\ell}) = x_\ell Z_i/m \tag{2.41}$$

$$\mathscr{D}\left(z_{i,j}u_{j|\ell}|Z_i\right) = E(z_{i,j}^2|Z_i)E(u_{j|\ell}^2) - (x_\ell Z_i/m)^2 \sim \delta\sigma_\ell^2 \tag{2.42}$$

and similarly to the proof of Lemma 5, we consider r.v. $z_{i,j}^2$ and the sums Z_i to be independent. We also remove the term $(x_\ell Z_i/m)^2$ in (2.42). Indeed, this term is immaterial since $x_\ell^2 \leq \sigma_\ell^2$ and $(Z_i/m)^2 \lesssim cm^{-2}\ln m = o(\delta)$, according to (2.37). In essence, here r.v. $z_{i,j}u_{j|\ell}$ have negligible means, which yield similar values of conditional variances $\mathscr{D}\left(z_{i,j}u_{j|\ell}|Z_i\right)$ and the second moments $E\left(z_{i,j}u_{j|\ell}|Z_i\right)^2$.

We can now proceed with r.v. $h_{i|\ell+1} = 2\sum_j z_{i,j}u_{j|\ell}$ that sums up independent r.v. $z_{i,j}u_{j|\ell}$ derived in Step B of Ψ_{soft}. Here we obtain

$$E\left(h_{i|\ell+1}|Z_i\right) = mE\left(z_{i,j}u_{j|\ell}|Z_i\right) \sim 2x_\ell Z_i \tag{2.43}$$

$$\mathscr{D}\left(h_{i|\ell+1}|Z_i\right) = m\mathscr{D}\left(z_{i,j}u_{j|\ell}|Z_i\right) \sim 4c\sigma_\ell^2 \tag{2.44}$$

We can now proceed with the r.v. $u_{i|\ell+1} \sim \tanh(h_{i|\ell+1}/2)$ used in Step C of Ψ_{soft}. For a given Z_i, r.v. $h_{i|\ell+1}$ has Gaussian pdf $\mathscr{N}(2x_\ell Z_i, 4c\sigma_\ell^2)$. By using the variables $z \equiv x_\ell Z_i$ and $t = z/\sigma_\ell\sqrt{c}$, we obtain (2.34):

$$E\left(u_{i|\ell+1}\right) \sim \left(2\pi\sigma_\ell^2 c\right)^{-1/2}\int_{-\infty}^{\infty}\tanh(z)e^{-(z-x_\ell c)^2/2c\sigma_\ell^2}dz$$

$$= (2\pi)^{-1/2}\int_{-\infty}^{\infty}\tanh(\sigma_\ell t\sqrt{c})e^{-\left(t-x_\ell\sqrt{c}/\sigma_\ell\right)^2/2}dt = F_c(x_\ell,\sigma_\ell) \tag{2.45}$$

Similarly, we obtain (2.35):

$$E\left(u_{i|\ell+1}^2\right) \sim G_c(x_\ell,\sigma_\ell) \tag{2.46}$$

which completes the proof. ■

Recall that the original r.v. $u_{i|0}$ have equal power moments $x_0 = \sigma_0^2$ of (2.6). The following lemma shows that nonlinear transformations (2.45) and (2.46) preserve this equality. It is for this reason that we rescaled the original r.v. $y_{i,j}$ into $z_{i,j}$ to achieve equality (2.6).

Consider function $F_c(x,\sigma)$ of (2.32) for $|x| = \sigma^2$. For any c, this gives the function

$$R_c(x) = (2\pi)^{-1/2} \int_{-\infty}^{\infty} \tanh(t\sqrt{|x|\,c}) e^{-\left(t-\sqrt{|x|c}\right)^2/2} dt \tag{2.47}$$

Lemma 7 *For any two quantities x, σ such that $|x| = \sigma^2$ and any $c > 0$, functions $F_c(x,\sigma)$ and $G_c(x,\sigma)$ satisfy relation*

$$\begin{aligned} F_c(x,\sigma) &= G_c(x,\sigma) = R_c(x), \qquad \text{if } x \geq 0 \\ F_c(x,\sigma) &= -G_c(x,\sigma) = -R_c(x), \quad \text{if } x < 0 \end{aligned} \tag{2.48}$$

Proof. Let $x = \sigma^2$ and $r = t\sqrt{xc}$. Then $e^{-(t-\sqrt{xc})^2/2} = e^r e^{-t^2/2} e^{-xc/2}$. Consider the function

$$f(r) = e^r \left(\tanh(r) - \tanh^2(r)\right) = \frac{e^r - e^{-r}}{1 + e^{2r} + e^{-2r}}$$

Clearly, $f(r)$ is an odd function of r. Then

$$F_c(x,\sigma) - G_c(x,\sigma) = (2\pi xc)^{-1/2} e^{-xc/2} \int_{-\infty}^{\infty} f(r) e^{-r^2/2xc} dr = 0$$

The case of $x < 0$ is similar. Note that $F_c(x,\sigma)$ is an odd function and $G_c(x,\sigma)$ is an even function. Then we proceed as above. ■

Lemma 8 *For $c \leq 1$, equation (2.23) has a single solution $x = 0$. For $c > 1$, equation (2.23) has three solutions: $x = 0$, $x_* \in (0,1)$ and $-x_*$.*

Proof. Let $x > 0$. Integration in (2.47) includes the pdf of $\mathcal{N}(\sqrt{xc}, 1)$, which gives negligible contribution beyond an interval $t \in (-x^{-1/4}, x^{-1/4})$. For $x \to 0$, we can now limit 2.47) to this interval. In this case, $t\sqrt{xc} \to 0$ for any c and $\tanh(t\sqrt{xc}) \sim t\sqrt{xc}$. Then

$$R_c(x) \sim (2\pi)^{-1/2} \int_{-\infty}^{\infty} t\sqrt{xc} e^{-(t-\sqrt{xc})^2/2} dt = xc \tag{2.49}$$

Thus, inequality $R_c(x) > x$ holds for sufficiently small x iff $c > 1$. On the other hand, $\tanh(t\sqrt{xc}) < 1$ and therefore $R_c(x) < 1$ for any x. Now we see that functions $y = R_c(x)$ and $y = x$ intersect at some point $x_* \in (0,1)$ for any $c > 1$. Finally, it can be verified that $R_c(x)$ has a declining positive derivative $R_c'(x)$, unlike the constant derivative 1 of the function $y = x$. Therefore, equation (2.23) has a single positive solution x_*. ■

In Figure 2.5, function $y = R_c(x)$ is shown for different values of $x \in [0,1]$ and $SNR = 10\log_{10}(c/4)$. The cross-point of functions $y = R_c(x)$ and $y = x$ represents the root x_*. Here the threshold $c = 1$ corresponds to $SNR = -6$ dB. Summarizing Lemmas 6-8, we have corollary 9.

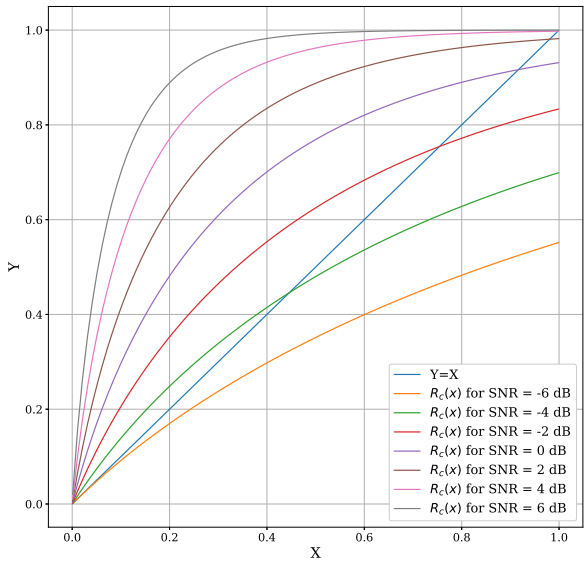

Figure 2.5: Functions $y = R_c(x)$ and $y = x$ for different values of $SNR = 10\log_{10}(c/4)$.

Corollary 9 *Let $m \to \infty$. Then r.v. $u_{i|\ell}$, $i = 1,..,m$, have power moments x_ℓ and σ_ℓ^2 that satisfy equality $|x_\ell| = \sigma_\ell^2$ for any iteration ℓ. Iteration ℓ transforms x_ℓ and σ_ℓ^2 into*

$$|x_{\ell+1}| = \sigma_{\ell+1}^2 = R_c(x_\ell) \tag{2.50}$$

Proof of Theorem 4.

1. Lemma 8 shows that for $c > 1$, function $R_c(x_\ell)$ grows for positive x_ℓ. Thus, equality $R_c(x_\ell) = x_\ell$ holds iff $x_\ell = x_*$, where x_* the root of (2.23). Next, consider initial iterations $\ell = 0,...$ Here r.v. u_0 has pdf $\mathcal{N}(\delta, \delta/m)$ and (with high probability) has vanishing values $|u_0| \le \sqrt{\delta/m}\ln m$. In further iterations ℓ, transform (2.49) performs simple scaling $x_{\ell+1} \sim cx_\ell$ as long as $x_\ell \to 0$ for $m \to \infty$. Thus, algorithm Ψ_{soft} fails for $c < 1$ since $x_\ell \to 0$ in this case.

2. Now let $c > 1$ and $L = \ln m / \ln c$. Note that $u_0 < 0$ with probability $Q(\sqrt{\delta m}) \sim Q(\sqrt{c})$. For iterations $\ell = o(L)$ and $m \to \infty$, we still obtain vanishing moments $|E(u_\ell)| \lesssim c^\ell \delta \to 0$ .It can also be verified that $E(u_\ell)$ moves away from 0 in $\mu = \alpha L$ iterations for some $\alpha > 0$.. Note also that r.v. u_ℓ has variance $\mathscr{D}(u_\ell) \le \mathscr{D}(u_{i|\ell})/m \le 1/m$. Thus, both cases, $u_\ell \to x_*$ or $u_\ell \to -x_*$, hold with high probability as $\ell \to \infty$.

3. We can now derive the BER for both cases. From (2.43) and (2.42), we see that the Gaussian random variable $h_{i|\ell+1}$ has the moments

$$E\left(h_{i|\ell+1}\right) \sim 2x_\ell E\left(Z_i\right) = 2x_\ell c, \quad \mathscr{D}\left(h_{i|\ell+1}\right) \sim 4c\sigma_\ell^2$$

For any iteration ℓ, we can now estimate BER $p_{i|\ell+1} = \Pr\{h_{i|\ell+1} < 0\}$ as

$$p_{i|\ell+1} = Q\left(x_\ell c/\sigma_\ell\sqrt{c}\right) = \begin{cases} Q\left(\sqrt{x_\ell c}\right), & \text{if } x_\ell > 0 \\ 1 - Q\left(\sqrt{-x_\ell c}\right), & \text{if } x_\ell < 0 \end{cases} \tag{2.51}$$

4. Consider the probabilities $P_\ell = \Pr\{x_\ell < 0\}$ and $1 - P_\ell = \Pr\{x_\ell > 0\}$, which define conditions of (2.51). We will now use two partial distributions of r.v. u_ℓ that have opposite means $\pm b_\ell$, where $b_\ell = |x_\ell|$. According to (2.50), r.v. $u_{i|\ell}$ have the second moment $E(u_{i|\ell}^2) = b_\ell$. Then r.v. $u_\ell = \sum_i \left(u_{i|\ell}/m\right)$ has the pdf $\mathcal{N}(\pm b_\ell, \eta_\ell)$ with the variance

$$\eta_\ell = \left(b_\ell - x_\ell^2\right)/m = b_\ell(1 - b_\ell)/m$$

Note that $b_\ell \to x_*$ for $\ell > L$, whereas $\eta_\ell \to 0$ as $\ell, m \to \infty$. Thus, r.v. u_ℓ cross 0 with a vanishing probability for any iteration $\ell > L$. On the other hand, r.v. u_ℓ may cross 0 multiple times if $\ell = o(L)$. From now on, we take $\ell = o(L)$. Then we will express $P_{\ell+1}$ via P_ℓ using the mean

$$b_\ell = c^\ell \delta$$

5. Consider both distributions $\mathcal{N}(x_\ell, \eta_\ell)$, where $x_\ell = \pm b_\ell = \pm c^\ell \delta$. Given some value u of r.v. u_ℓ, define r.v. $u_{\ell+1}\,|\,u = m^{-1}\sum_i \left(u_{i|\,\ell+1}\,|\,u\right)$. This r.v. has pdf

$$p(u) = \mathcal{N}(cu, c\eta_\ell) = (2\pi\eta_\ell)^{-1/2} e^{-(u-x_\ell)^2 m/2\eta_\ell}$$

First, let $E(u_\ell) = b_\ell$. Clearly $\Pr\{cu < 0\} = Q(u\sqrt{c/\eta_\ell})$. Then we average over all values u of u_ℓ and obtain the probability

$$\begin{aligned} S_\ell = \Pr\{u_{\ell+1} < 0\,|\,E(u_\ell) = b_\ell\} &= \int_{-\infty}^{\infty} Q(u\sqrt{c/\eta_\ell})p(u)du \\ &\sim (2\pi)^{-1/2}\int_{-\infty}^{\infty} Q(t\sqrt{c})e^{-(t-b_\ell/\sqrt{\eta_\ell})^2/2}dt \end{aligned}$$

Here we use variable $t = u/\sqrt{\eta_\ell}$. Next, we consider the initial iterations $\ell = o(\ln m/\ln c)$ and

introduce parameter

$$C_\ell = b_\ell/\sqrt{\eta_\ell} \sim \sqrt{c^{\ell+1}/(1-m^{-1}c^{\ell+1})} \sim c^{(\ell+1)/2} \tag{2.52}$$

Note that $b_\ell/\sqrt{\eta_\ell} = C_\ell \sim c_\ell$, which gives (2.24). Similarly, for $E(u_\ell) = -b_\ell$, we obtain the probability

$$Q_\ell = \Pr\{u_{\ell+1} < 0 \,|\, E(u_\ell) = -b_\ell\} = \int_{-\infty}^{\infty} Q(u/\sqrt{c/\eta_\ell}) p(-u) du$$

For $\ell < L = \ln m/\ln c$, this gives the probability

$$P_{\ell+1} = \Pr\{u_{\ell+1} < 0\} = (1-P_\ell)S_\ell + P_\ell Q_\ell = S_\ell + P_\ell T_\ell \tag{2.53}$$

where $T_\ell = Q_\ell - S_\ell$ is given by (2.25). We can also slightly tighten estimates (2.24) and (2.25), by using quantity C_ℓ of (2.52) instead of c_ℓ.

We can now proceed with iterations P_ℓ, which begin with $P_0 = Q(\sqrt{c})$. For any ℓ, quantities S_ℓ and T_ℓ depend on c only. Also, quantities $c_\ell = c^{(\ell+1)/2}$ grow exponentially, in which case $S_\ell \to 0$ and $Q_\ell \to 1$. Thus, quantities P_ℓ converge, since $P_{\ell+1} \sim P_\ell Q_\ell$ for sufficiently large $\ell \geq L$.

We can now evaluate P_{soft}. For $\ell \to \infty$, we replace P_ℓ with P_∞ in (2.53) and use x_* of (2.23). Finally, note that (2.26) is only an asymptotic estimate. Here we excluded the residual term $O(\ln m/\sqrt{m})$ used in approximations (2.36) and (2.38). ■

High-signal case. Consider functions S_ℓ and T_ℓ of (2.24) and (2.25) as $c \to \infty$. Then $S_\ell \to 0$, $T_\ell \to 1$, and $P_\infty \to P_0 = Q(\sqrt{c})$. In this case, $P_{soft} \sim 2Q(\sqrt{c}) \sim (2/\pi c)^{1/2} e^{-c/2}$. The latter represents a 3 dB gain over the uncoded modulation, whose BER has the order of $e^{-c/4}$.

Complexity. Given m information bits, algorithm Ψ_{soft} has complexity of order

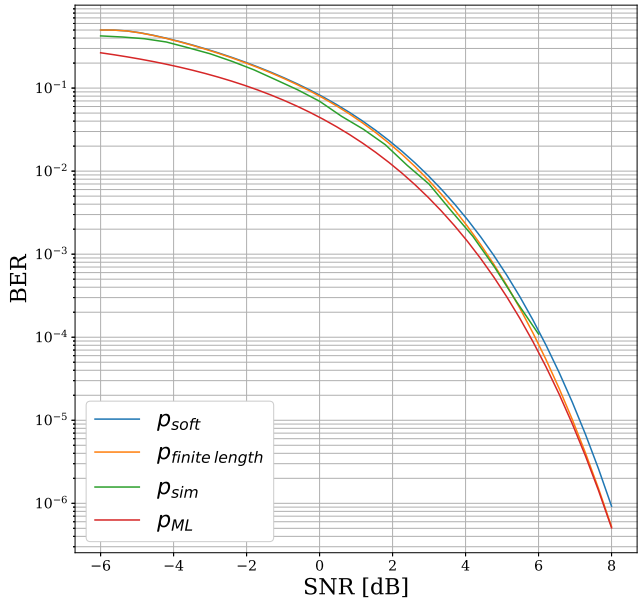

Figure 2.6: Simulation results and analytical bounds for the algorithm Ψ_{soft} applied to modulation-type codes C_{128} of length 8256.

$m^2 \log m$. Indeed, each iteration ℓ recalculates quantities $u_{i|\ell}(j)$ and $h_{i|\ell}(j)$ for all ordered pairs (i, j). This requires $O(m^2)$ operations. We also need $O(\log m / \log c)$ iterations ℓ to make the estimates $u_{i|\ell}$ bounded away from 0 as $m \to \infty$. Also, it can be shown that the stable point x_* can be reached within a margin $\varepsilon \to 0$ in $O\left(\ln \varepsilon^{-1} / \ln c\right)$ iterations. For $\varepsilon = m^{-1}$, this gives the overall complexity of $m^2 \ln m / \ln c$ operations.

2.6 BER bounds on binary symmetric channels

In this section, we use all the techniques introduced in Chapters 2.4 and 2.5 to come up with theoretical bounds of BER for ML algorithm and BP algorithm on the BSC. In order not to repeat the entire section, we try to reuse as many of those theorems as possible to explain the proofs on the BSC.

2.6.1 Lower bounds for BER of ML algorithm on BSC

Theorem 10 *Let codes C_m be used on a BSC with an SNR of s where $c_b = 8s/\pi$ per information bit. Then for $m \to \infty$, ML decoding of codes C_m has BER*

$$p_{ML}(s) \gtrsim 2P_{c_b}(1-P_{c_b}) = 2Q(\sqrt{c_b}) - 2Q^2(\sqrt{c_b}) \tag{2.54}$$

Proof. This theorem follows the proof of theorem 2.13 closely. We assume that the all-one codeword 1^n is sent and then we define r.v. B_i as follows.

$$B_0 = \sum_{j\neq 0,1} b_{0,j}, \;\; B_1 = \sum_{j\neq 0,1} b_{1,j}$$

Below we use the assumption that $m \to \infty$. It is easy to verify that r.v. B_0, and B_1 have asymptotic pdf $\mathcal{N}(\sqrt{mc_b}, m)$.

Code words of minimum weight in C_m include m generator rows $g^{(p)}$, $p = 1, ..., m$, of the generator matrix G_m and their sum $g^{(0)} = g^{(1)} + ... + g^{(m)}$. Under ML decoding, any two-word code $\{1^n, g^{(p)}\}$, has BER

$$P_{c_b} = \Pr\{Y_1 < 0\} \sim Q(\sqrt{c_b}) \tag{2.55}$$

Without loss of generality, we consider BER of symbol $a_{0,1}$. In essence, we use the same proof as theorem 2.13 to show that ML decoding gives $a_{0,1} = -1$ if so does one of the codes $\{1^n, g^{(p)}\}$ for $p = 0, 1$. All received vectors b form four disjoint subsets $U = V_1, V_2, V_3, V_4$, where

$$V_1 = \{z \,|\, B_0 < 0,\, B_1 > 0\},\; V_2 = \{z \,|\, B_0 > 0,\, B_1 < 0\} \tag{2.56}$$

$$V_3 = \{z \,|\, B_0 > 0,\, B_1 > 0\},\; V_4 = \{z \,|\, B_0 < 0,\, B_1 < 0\} \tag{2.57}$$

Clearly, $\Pr\{V_1\} = \Pr\{V_2\} = P_{c_b}(1-P_{c_b})$. We use the same technique used in theorem 2.13 and

show that $p_{ML}(s) \gtrsim \Pr\{V_1\} + \Pr\{V_2\}$.

2.6.2 Probabilistic bounds of BP algorithm on BSC

Now, in order to show BER of BP algorithm we first show the probability of high weight errors with P_∞ and then show BER of those cases by calculating the correct x_*. For any $\ell = 0, 1, ..., L$ and any $m \to \infty$, we introduce parameter $c_\ell = c_b^{(\ell+1)/2}$. We then derive probabilities P_ℓ using recursion $P_{\ell+1} = (1 - P_\ell)S_\ell + P_\ell T_\ell$, where

$$S_\ell = (2\pi)^{-1/2} \int_{-\infty}^{\infty} Q(c_\ell t) e^{-(t-c_\ell)^2/2} dt \tag{2.58}$$

$$T_\ell = (2\pi)^{-1/2} \int_{-\infty}^{\infty} Q(c_\ell t) \left(e^{-(t+c_\ell)^2/2} \right) dt \tag{2.59}$$

and $P_0 = Q(\sqrt{c_b})$. For any ℓ, probabilities P_ℓ depend on c_b only. P_ℓ is representative of high weight errors after ℓ iteration. Quantities P_ℓ converge exponentially fast as $\ell \to \infty$. Let $P_\infty = \lim_{\ell \to \infty} P_\ell$. We can now establish the asymptotic value of BER as $m \to \infty$.

Theorem 11 *Let codes C_m be used on a BSC channel with an SNR s per information bit and $c_b = 8s/\pi$. For $m \to \infty$ and $c_b \leq 1$, algorithm Ψ_{BSC} has BER $P_{BSC}(s) \to 1/2$. For $c_b > 1$,*

$$P_{BSC}(s) \sim (1 - P_\infty)\, Q\left(\sqrt{x_* c_b}\right) + P_\infty \left(1 - Q\left(\sqrt{x_* c_b}\right)\right) \tag{2.60}$$

where P_∞ can be calculated using equations (2.58), (2.59) and x_* is the positive answer to the equation $x = R_{c_b}(x)$ where $R_{c_b}(x)$ is (2.61) .

$$R_{c_b}(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \tanh(t\sqrt{x c_b}) e^{-\left(t - \sqrt{x c_b}\right)^2/2} dt \tag{2.61}$$

We begin the proof of this theorem by introducing to two random variable $h_{i|\ell}$ and $u_{i|\ell}$

where $h_{i|\ell}$ is half of the log likelihood of information bit i at iteration ℓ and $u_{i|\ell}$ is the probability offset of that information at iteration ℓ. Here we assume the all-one 1^n codeword is sent (without loss of generality we can later claim that BER of this one information bit is the same as all other information bits). It is important to mention that $h_{i|1}$ is a Gaussian random variable with distribution $\mathcal{N}(c_b u_0^2, c_b u_0^2)$ where $u_0 = \sqrt{c_b/m}$ is the probability offset of BSC.

$$u_{i|\ell} = \tanh(h_{i|\ell}) \tag{2.62}$$

$$h_{i|\ell+1} = \sum_k \tanh^{-1}(u_{k|\ell} b_{i,k} u_0) \tag{2.63}$$

Lemma 12 *Let us consider a r.v. V with a Gaussian distribution $\mathcal{N}(a,a)$ where $a \to 0$. We can claim that* $\tanh(v) = v$.

Proof. *We know that* $\lim_{x\to 0}\tanh(x) = x$ *and we can say that* $\Pr[|v-a| < c\sqrt{a}] \to 1$ *where c is any desired large constant, so we can say* $\tanh(v) = v$.

Theorem 13 *If $c_b^\ell u_0^2 \ll 1$ and $h_{i|\ell}$ has a Gaussian distribution $\mathcal{N}(c_b^\ell u_0^2, c_b^\ell u_0^2)$ we can say that $h_{i|\ell+1}$ will have a Gaussian distribution $\mathcal{N}(c_b^{\ell+1} u_0^2, c_b^{\ell+1} u_0^2)$.*

Proof. We use lemma (12) and weak dependence of these random variables to show that $h_{i|\ell+1}$ will have a Gaussian distribution $\mathcal{N}(c_b^{\ell+1} u_0^2, c_b^{\ell+1} u_0^2)$.

Theorem 14 *If $E(u_{i|\ell}) = a$ and $E(u_{i\ell}^2) = a$ then $h_{i|\ell+1}$ has a Gaussian distribution $\mathcal{N}(c_b a, c_b a)$.*

Proof. We use weak dependence of $u_{i\ell}$ and show

$$E(h_{i|\ell+1}) = m u_0^2 E(u_{i\ell}) = c_b a \tag{2.64}$$

$$E(h_{\ell+1}^2) = u_0^2(mE(u_{i\ell}^2) + m^2 u_0^E (u_{i|\ell})^2) = c_b a + (c_b a)^2 \tag{2.65}$$

Theorem 15 *If $h_{i|\ell}$ has a Gaussian distribution $\mathcal{N}(x,x)$ then $E(u_i) = \mu_x$ and $E(u_{i\ell}^2) = \sigma_x^2$ where μ_x and σ_x can be calculated using equations (2.66) and (2.67)*

$$\mu_x = (2x\pi)^{-1/2} \int_{-\infty}^{\infty} \tanh(t) e^{-((t-x)^2/2x)} dt \tag{2.66}$$

$$\sigma_x^2 = (2x\pi)^{-1/2} \int_{-\infty}^{\infty} \tanh^2(t) e^{-((t-x)^2/2x)} dt \tag{2.67}$$

Note. By using lemma 7 we can show that $|\mu_x| = \sigma_x^2$.

Now we can put together the proof of theorem (11) by stating that if $E(u_{i|\ell}) = E(u_{i|\ell+1})$ and $E(u_{i\ell}^2) = E(u_{i\ell+1}^2)$ the point x_* will satisfy equation (2.61) and BER of Ψ_{BSC} is going to be $P_{BSC}(s)$. It is interesting to point out that in Figure 2.7 for $SNR < -4.07$ dB our $x_* = 0$. In Figure 2.8 we can see that simulation result and theoretical bound P_{BSC} are virtually identical to each other.

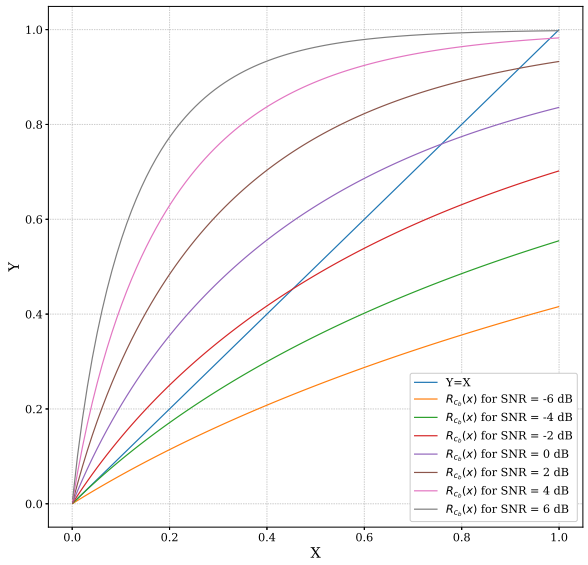

Figure 2.7: Functions $y = R_{c_b}(x)$ and $y = x$ for different values of SNR

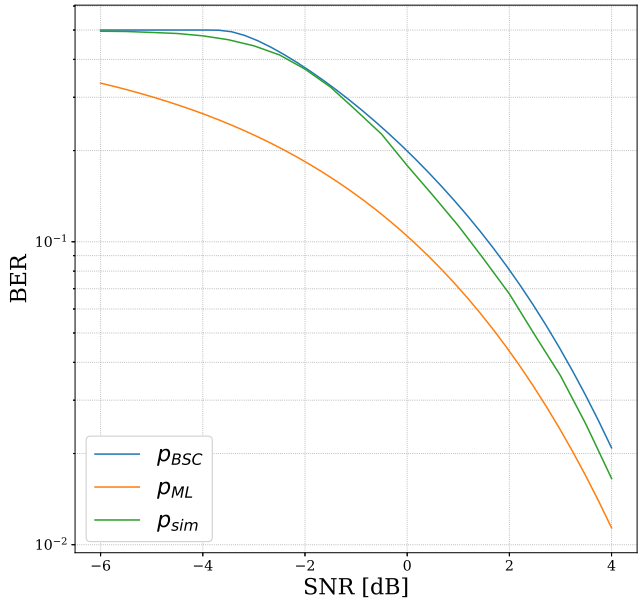

Figure 2.8: BER of BP algorithm on BSC for codes C_m for $m = 128$

2.7 Design improvements and simulation results

In Figure 2.6, we plot analytical bound P_{soft} of (2.26) along with simulation results P_{sim} and the lower bound P_{ML} of (2.13). Here we consider codes C_m of dimension $m = 128$ on the AWGN channels with various SNRs $10\log_{10}(c/4)$. We see that both bounds (2.26) and (2.13) tightly follow simulation results and each other. This also supports our main assumption that the algorithm Ψ_{soft} can be considered using independent random variables. For completeness, we also plot non-asymptotic bound $P_{finite\ length}$ obtained by using parameters C_ℓ of (2.52) in both formulas (2.24) and (2.25). Unexpectedly, this bound completely coincides with a much simpler lower bound P_{ML} for high SNR.

Simulation results of the BP algorithm Ψ_{soft} for different values of m are presented in Figure 2.9. These results show that for different values of m BER of these codes are similar and the assumption $m \to \infty$ can be relaxed to m being a large number and still the theoretical probabilistic bound (2.26) can describe BER of these Codes. It is also important to note that they

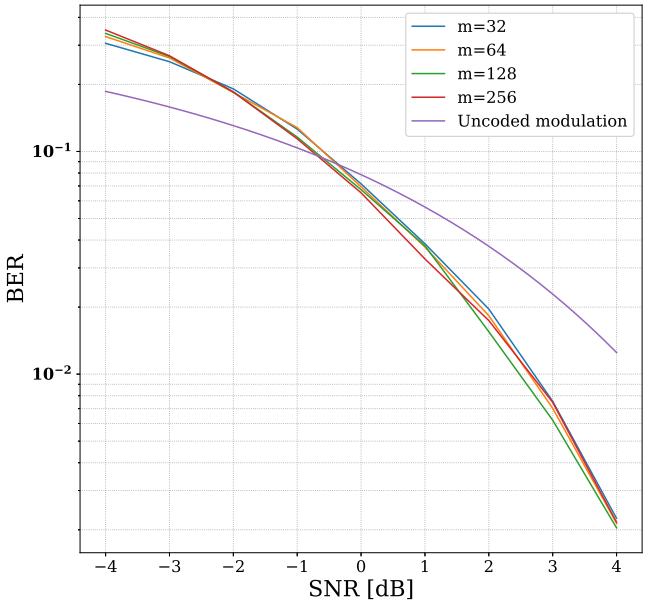

Figure 2.9: BER of BP algorithm Ψ_{soft} for codes C_m

improve on uncoded modulation for SNRs above 0 dB.

To improve performance of Ψ_{soft}, recall from (2.13) that ML decoding error $p_{ML} \sim 2Q(\sqrt{c})(1-Q(\sqrt{c}))$ of any symbol $a_{0,i}$ is defined by the two codewords of minimum weight, $g^{(i)}$ and $g^{(0)} = g^{(1)} + ... + g^{(m)}$. The latter also affects all m bits $a_{0,i}$, unlike $g^{(i)}$. We now repeat s times the information block I_m of code C_m using generator matrix $G_{m,s} = [I_m|...|I_m|J_m]$. This will also increase s times the weight of $g^{(0)}$. Then we obtain code $C_{m,s}$ with parameters

$$n_{m,s} = m(m+2s-1)/2, \quad k = m, \quad d_{m,s} = m+s-1$$

Decoding of code $C_{m,s}$ is almost identical to the algorithm Ψ_{soft}. The only difference arises in calculating the quantities $h^{\ell}_{0,i}$ in (4.6). With a new matrix $G_{m,s}$, we now have s copies $a^{(1)}_{0,i}$,...,$a^{(s)}_{0,i}$ of any symbol $a_{0,i}$ and can use s estimates of these symbols instead of a single estimate in (4.6):

$$h^{\ell}_{0,i} := h^{(1)}_{0,i} + ...h^{(s)}_{0,i} + S^{\ell}_{0,i}$$

For $s > 1$, code $C_{m,s}$ has a lower code rate $R_{m,s} = 2(m+2s-1)^{-1}$.

Given SNR c for a former code $A_{m,1}$, we now obtain a lower SNR $cR_{m,s}/R_{m,1}$. According to (2.12), then decoding into any vector $g^{(i)}$ of minimum weight d has probability $Q(\sqrt{c\eta_1})$, where $\eta_1 = (m+s-1)/(m+2s-1) < 1$. On the other hand, decoding into the vector $g^{(0)}$ of weight sm has a much lower BER of $Q(\sqrt{c\eta_0})$, where $\eta_0 = sm(m+2s-1)^{-1}$. In other words by repeating information bits a number of times we can reduce the probability of P_∞ in formula (2.26) considerably.

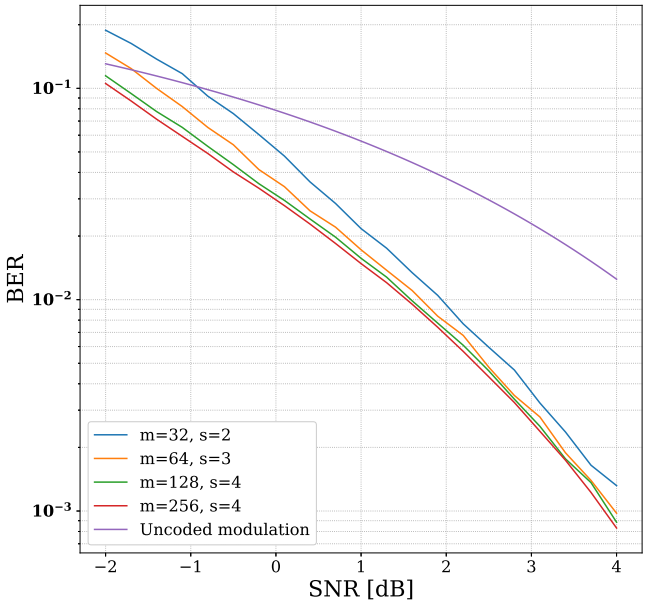

Figure 2.10: BER of BP algorithm Ψ_{soft} for codes $C_{m,s}$

For $s > 2$ and $m \to \infty$, we have $\eta_0 \sim s$ and $\eta_1 \to 1$. It is important to mention that for any large constant $s > 2$ we can say that $Q(\sqrt{cx_*\eta_1}) \gg Q(\sqrt{c\eta_0})$.

Corollary 16 *For $m \to \infty$ and a large constant s code $C_{m,s}$ has output BER where x_* can be calculated using the same recursive equations used to calculate (2.26)*

$$p_{m,s}(c) \sim Q(\sqrt{cx_*\eta_1}) + Q(\sqrt{c\eta_0}) \tag{2.68}$$

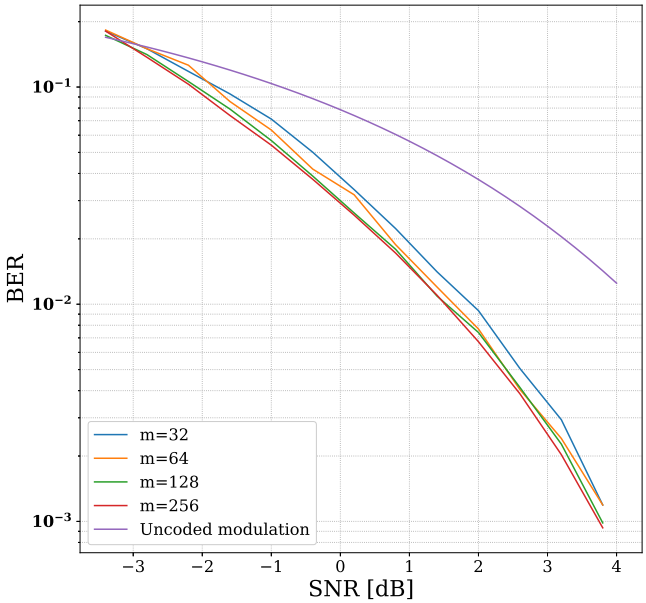

Figure 2.11: BER of algorithm Ψ_{soft} for codes $C_{m,s}$ with optimal parameter s.

Figures 2.10 and 2.11 we show BER achieved by codes $C_{m,s}$ on AWGN Channel. In Figure 2.12 we show BER of BSC for $m = 128$ and $s = \{1,\dots,6\}$ and in Figure 2.10, we have BER of AWGN channel for the case of $s = 2,3,4$ depending on m but keep the same s for all SNRs. Figure 2.11 presents the output BER when parameter s is also optimized depending on a given SNR. As SNR grows, parameter s declines and reaches $s = 1$ for $c \geq 3$. By contrast, codes $C_{m,s}$ outperform codes $C_{m,1}$ at the lower SNRs. Codes $C_{m,s}$ also outperform uncoded modulation at $SNR \geq -3$ dB and gain about 1.4 dB at the channel capacity $SNR = -1.59$ dB and about 2 dB at $SNR = 0$.

Finally, we combine modulation codes $C_{m,s}$ and results of BP algorithm Ψ_{soft} with some polar code of length m, which is formed by m information bits $a_{0,i}$, described in Figure (2.13). In decoding, the algorithm Ψ_{soft} outputs bits $a_{0,i}$ and passes them to a SCL decoder of the polar code that selects the optimal frozen bits [67, 68].

Note that adding a polar code of some rate $R < 1$ reduces the overall SNR by a factor R. To compare code performance at the given SNR, we will adjust the SNR and use BP algorithm

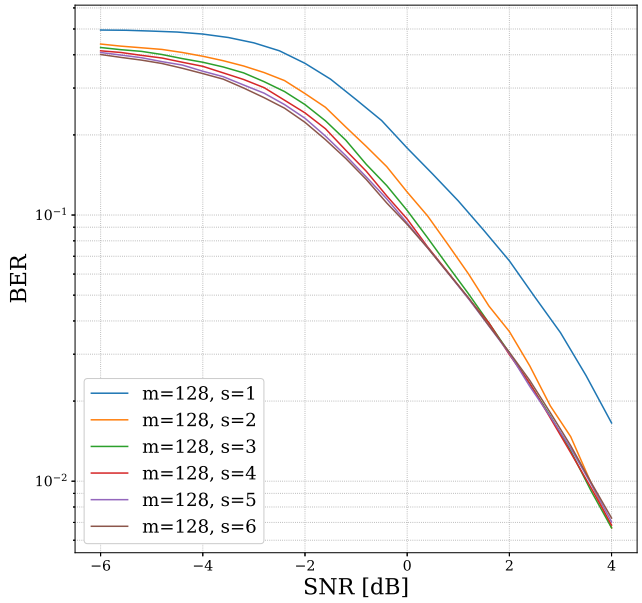

Figure 2.12: BER of BP algorithm Ψ_{BSC} for codes $C_{m,s}$ where $m = 128$ and $s = \{1 \dots 6\}$

Ψ_{soft} at the lower SNR. For each bit $a_{0,i}$, we then consider its output as a soft-decision AWGN channel and calculate the channel capacity. This way we can calculate the optimal precoding rate that allows the minimum SNR per information bit.

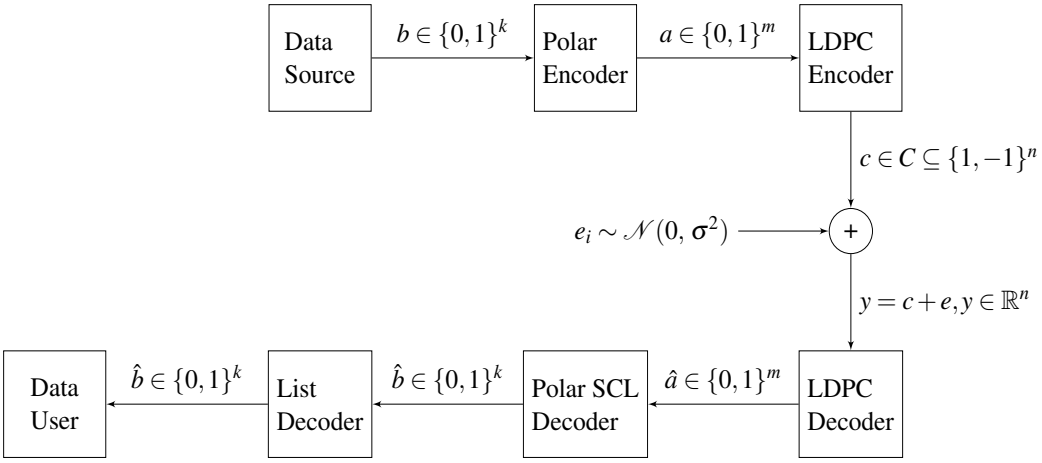

Figure 2.13: Polar-LDPC code design

For the non-asymptotic setting with parameters $m = 128, 256$, and precoding rate, $R = 0.64$ the results of this optimization are shown in Figure 2.14. Here we use SCL decoding with the list size $L = 32$ and after SCL we select the best candidate over the entire block size.

For $m = 128$, we reduce the output WER of [49] by a factor of 3 and also double the code rate. Selecting the best codeword over the entire block size will reduce the WER for non-asymptotic cases.

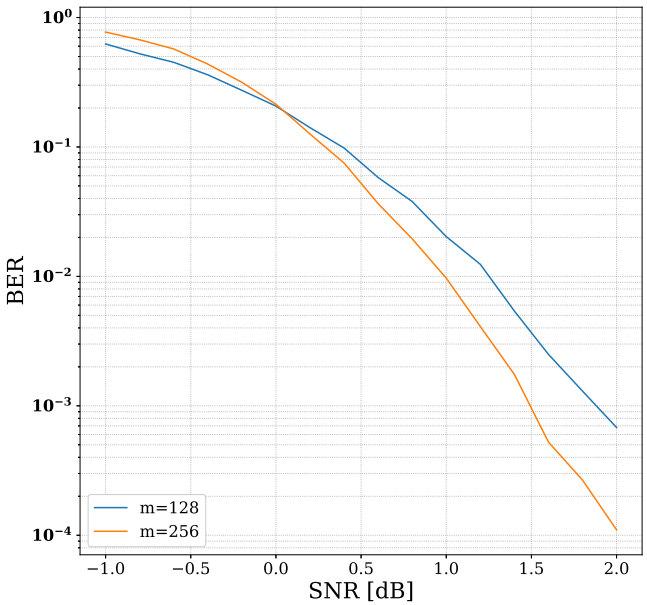

Figure 2.14: Block error rate of codes $C_{m,s}$ using BP decoding and polar-based precoding for $m = 128, 256$

2.8 Concluding remarks

In first part of this chapter, we introduced "modulation " binary codes C_m and we discussed their code properties, encoding and decoding algorithms. Then we described belief propagation (BP) decoding on BSC and AWGN channel and proved BER bounds for the maximum likelihood and belief propagation algorithms for BSC and AWGN channels. Then we showed that simulation results and theoretical boundaries are very close on both channels. Then we introduce a very easy approach to reduce BER of AWGN channel by repeating information bits s times and called theses codes $C_{m,s}$. We showed that encoding and decoding of these codes are very similar to C_m but they have the ability of outperforming uncoded modulation for any

channel SNR above -3 dB on AWGN channel for moderate lengths and for asymptotic conditions, $m \to \infty$, they can outperform uncoded modulation for any $SNR > -3.7$ dB per information bit. They also gain about 3 dB on UM for the AWGN channels with a higher SNR.

This ability of outperforming uncoded modulation for SNRs smaller than $-1.59\,dB$ allows us to use this design as an inner code for a two stage polar-LDPC code. In turn, codes $C_{m,s}$ - combined with polar codes - improve code performance on the high noise AWGN channels with a vanishing channel snr as $m \to \infty$. It can be proven that these modified codes can achieve exponentially declining BER for any $SNR > -1.0$ dB. The main advantage of theses codes is their latency which is of order $\mathcal{O}(\sqrt{n})$.

Chapter 3

Codes approaching the Shannon limit with polynomial complexity per information bit

3.1 Introduction

In this chapter, we present a multilevel protection scheme, described in Figure 3.1, that uses b different polar codes in parallel as the precoding for the low rate LDPC code. We then present tight bounds for BER of LDPC codes with frozen information bits and show that these bounds closely follow simulation results. Using these accurate bounds we then numerically show this design has the ability of achieving the Shannon limit of -1.5917 dB per information bits. It is also important to mention that this capacity achieving design achieves the channel capacity with a latency of order $\sqrt{n}$ if all operations of LDPC code decoding is done in parallel. The main claim of this chapter is mentioned in Statement 1 as follows.

Statement 1 *There exist codes $\widehat{C}_m$ of dimension $k \to \infty$ and length $O(k^2)$ that have complexity of*

order $\mathcal{O}(k^2 \log k)$ *and upper-bound BER to the order of* $\exp\{-c_{SNR}\sqrt{k}\}$*, where* $c_{SNR} > 0$ *depends on SNR and is positive for any SNR above the Shannon limit of* $\ln 2$.

Statement 1 is predicated on our "weak-independence" assumption discussed in section 2.5. From a practical point of view, this design heavily relies on the capacity achieving property of the small polar precoding blocks and this may result in really large block sizes for practical purposes.

3.2 Multilevel protection schemes

Let $B_i = B_i(\mu, \mu r_i)$ be a sequence of b capacity-achieving polar codes with rates $0 \le r_0 < ... < r_{b-1}$, that will be specified later. We first encode data block $\overline{\mathbf{a}_i}$ of length μr_i into some vector $A_i \in B_i$ and then form a compound block $A = (A_0, ..., A_{b-1})$ of length $m = \mu b$. Below $\mu \to \infty$ and b is a constant. Block A is further encoded by code C_m of rate $R_m = 2/(m+1)$ and length $n = \binom{m+1}{2}$. We use notation $\widehat{C}_m$ for the compound code of rate $R \sim R_m r$, where $r = \sum_i r_i / b$. Thus, code $\widehat{C}_m$ reduces code rate R_m by a factor of r, which gives SNR of $c/4r$ per information bit.

Let $I_s = \{\mu s + 1, ..., \mu(s+1)\}$ for any $s = 0, ..., b-1$. The received block $\widehat{C} = \widehat{C}(0)$ of length n is first decoded by the algorithm Ψ_{soft} using $L = O(\ln m)$ iterations. The result is some block $\widehat{A}(0)$ of length m. We then retrieve the first μ decoded bits in $\widehat{A}(0)$ that form the sub-block $\widehat{A}_0 = \left(\widehat{a}_1, ..., \widehat{a}_\mu\right)$ of length μ. Block $\widehat{A}_0$ is decoded by a polar code B_0 into some block $A_0 = \{a_1, ..., a_\mu\}$. We assume that the corrected block A_0 has $WER \to 0$ as $\mu \to \infty$. We then use A_0 to replace the first μ symbols of the block $\widehat{C}(0)$. The result is a new block $\widehat{C}(1)$ of length n. This completes round $s = 0$.

Round $s = 1$ is similar. Algorithm Ψ_{soft} now also employs block A_0 to recalculate the remaining $m - \mu$ information bits of $\widehat{C}(1)$. The obtained sub-block $\widehat{A}_1 = \left(\widehat{a}_{\mu+1}, ..., \widehat{a}_{2\mu}\right)$ is decoded into some vector $A_1 = \{a_{\mu+1}, ..., a_{2\mu}\}$ using code B_1. Then A_1 replaces $\widehat{A}_1$ in positions

$i \in I_1$ and yields a new block $\widehat{C}(2)$. Similarly, rounds $s = 2, ..., b-1$ only retrieve a block A_s on positions $i \in I_s$ Then we obtain block $\widehat{C}(s+1)$ that include corrected bits $a_1, ..., a_{(s+1)\mu}$.

In any round s, μs corrected information bits serve as frozen bits and aid the algorithm Ψ_{soft}. Indeed, with high probability, we use correct estimates $u_{j|\ell} = a_j$ for all $j \le \mu s$. Then the parity checks $u_{i|\ell+1}(j) = u_{i,j}u_{j|\ell}$ are reduced to the repetitions/inversions $u_{i|\ell+1}(j) = a_j u_{i,j}$ of symbols $u_{i,j}$. Also, recall that algorithm (4.7) outputs the likelihoods $h_{i|L}$ of all symbols a_i. Thus, we use $h_{i|L}$ as our bit estimates in every round s as follows.

For all $i \in \{\mu s+1, ..., m\}$ and $j \in \{1, ..., m\}$:

A. Use block $\widehat{C}(s)$. Derive $u_{i|\ell+1}(j) = u_{i,j}u_{j|\ell}$

and $h_{i|\ell+1}(j) = 2\tanh^{-1}(u_{i,j}u_{j|\ell})$

B. Derive $h_{i|\ell+1} = \sum_j h_{i|\ell+1}(j)$

C. If $\ell < L$, find $u_{i|\ell+1} = \tanh\left(h_{i|\ell+1}/2\right)$.

Goto A with $u_{i|\ell+1}$ and $\ell := \ell + 1$.

D. If $\ell = L$, use block $\widehat{A}_s = (h_{i|L},\ i \in I_s)$.

Decode it into $A_s \in B_s(\mu, \mu r_s)$.

E. Replace $\widehat{A}_s$ with A_s to form $\widehat{C}(s+1)$.

If $s < b-1$, let $s := s+1,\ \ell := 0$. Goto *A*.

If $s = b-1$, output bits $a_1, ..., a_m$.

Let an information block A consist of m zeros. We then use antipodal signaling and transmit a codeword 1^n over an AWGN channel. Round s includes μs correct information bits $u_{i|\ell} = a_i = 1$.

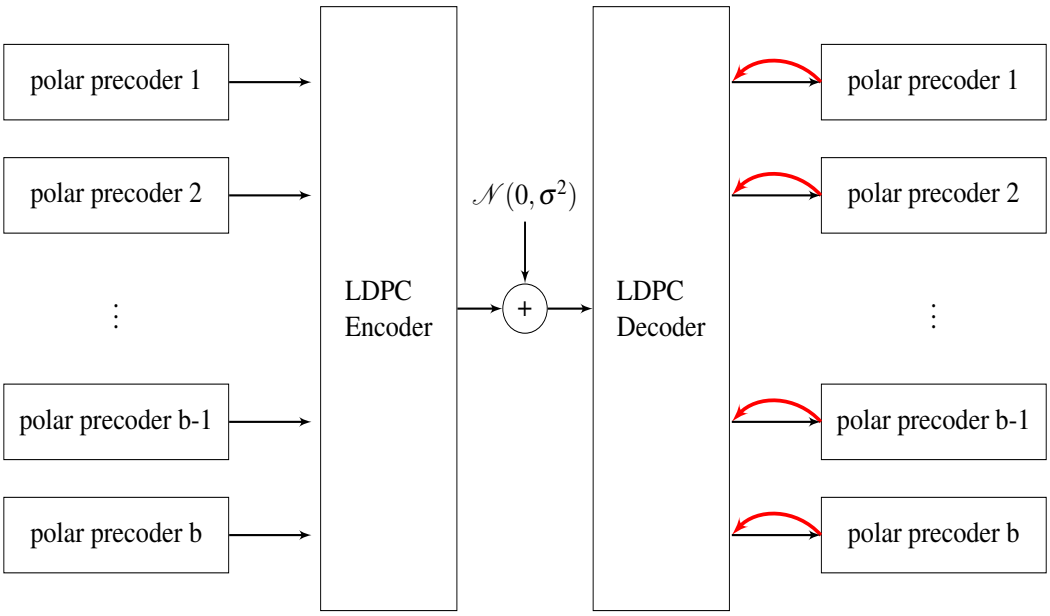

Figure 3.1: Multi level protection design on AWGN Channel

Let $\lambda_s = s/b$. Then the remaining $m - \mu s$ r.v. $u_{i|\ell}$, $i > \mu s$, have the average power moments

$$x_\ell = [m(1-\lambda_s)]^{-1} \sum_{i>\mu s} E u_{i|\ell} \tag{3.1}$$

$$\sigma_\ell^2 = [m(1-\lambda_s)]^{-1} \sum_{i>\mu s} E\left(u_{i|\ell}^2\right) \tag{3.2}$$

In particular, the initial setup with $\ell = 0$ employs the original r.v. $u_{i|0}$ that have asymptotic pdf $\mathcal{N}(\delta,\delta)$ for all $i > \mu s$ and satisfy equalities $x_0 = \sigma_0^2 = \delta$.

Theorem 17 *Let the algorithm Ψ_{soft} have λm correct information symbols $a_1 = \ldots = a_{\lambda m} = 1$, where $\lambda \in (0,1)$. Then the remaining $(1-\lambda)m$ symbols a_i have BER*

$$P_{soft}(\lambda, c) \sim Q\left(\sqrt{cX(\lambda)}\right) \tag{3.3}$$

where $X(\lambda)$ satisfies equations

$$X(\lambda) = \lambda + (1-\lambda)\,x(\lambda) \tag{3.4}$$

$$x(\lambda) = (2\pi)^{-1/2} \int_{-\infty}^{\infty} \tanh\left(t\sqrt{cX(\lambda)}\right) e^{-\left(t-\sqrt{cX(\lambda)}\right)^2/2} dt \tag{3.5}$$

Proof. In essence, we follow the proof of Theorem 4. The main difference - that simplifies the current proof - is that the former vanishing point $x_0 = \delta \to 0$ is now replaced with $X_0 \to \lambda$. This removes the random walks across 0 analyzed in parts 4 and 5 of the former proof. Thus, now we have the case of $P_\infty = 0$. The details are as follows.

For any $j \geq \mu s + 1$, we use approximations (2.39) and (2.40) and take $u_{j|\ell} = 1$ for $j \leq \mu s$. Then

$$h_{i|\ell+1}(j) \sim 2u_{i|\ell+1}(j) \sim \begin{cases} z_{i,j} u_{j|\ell}, & \text{if } j \geq \mu s + 1 \\ z_{i,j}, & \text{if } j \leq \mu s \end{cases}$$

For any given Z_i, consider the sums $Z_i' = \sum_{j \leq \mu s} z_{i,j}$ and $Z_i'' = \sum_{j \geq \mu s+1} z_{i,j}$. These sums have expected values $E(Z_i') = \lambda Z_i$ and $E(Z_i'') = (1-\lambda) Z_i$. Let

$$X_\ell = \lambda + (1-\lambda)\, x_\ell$$

$$\theta_\ell^2 = \lambda + (1-\lambda)\, \sigma_\ell^2$$

Then we define the moments

$$E\left(h_{i|\ell+1}\right) \sim 2x_\ell Z_i'' + 2Z_i' \sim 2Z_i\left[\lambda + x_\ell\,(1-\lambda)\right] = 2Z_i X_\ell \tag{3.6}$$

$$\mathscr{D}\left(h_{i|\ell+1}\right) \sim 4c\,(1-\lambda)\,\sigma_\ell^2 + 4c\lambda = 4c\theta_\ell^2 \tag{3.7}$$

Thus, r.v. $h_{i|\ell+1}/2$ has Gaussian pdf $\mathscr{N}(X_\ell c, \theta_\ell^2 c)$.

Next. consider r.v. $u_{i|\ell+1} \sim \tanh(h_{i|\ell+1}/2)$. Similarly to equalities (2.34) and (2.35), we have

$$E\left(u_{i|\ell+1}\right) \sim \left(2\pi\theta_\ell^2 c\right)^{-1/2} \int_{-\infty}^{\infty} \tanh(z) e^{-(z-X_\ell c)^2/2c\theta_\ell^2} dz = F_c(X_\ell, \theta_\ell) \tag{3.8}$$

$$E[u_{i|\ell+1}^2] \sim \left(2\pi\theta_\ell^2 c\right)^{-1/2} \int_{-\infty}^{\infty} \tanh^2(z) e^{-(z-X_\ell c)^2/2\theta_\ell^2 c} dz = G_c(X_\ell, \theta_\ell)$$

Any round $s = \lambda b$ begins with the initial values $X_0(\lambda)$ and $\theta_0^2(\lambda)$ that satisfy equalities

$$X_0(\lambda) = \theta_0^2(\lambda) = \lambda + \delta(1-\lambda) \sim \lambda \tag{3.9}$$

which are similar to the former equality $x_0 = \sigma_0^2$. Thus, we may follow the proof of Theorem 4 and obtain equality $F_c(X_\ell, \theta_\ell) = G_c(X_\ell, \theta_\ell)$ for any iteration ℓ. Now we see that $x_{\ell+1} = \sigma_{\ell+1}^2$ and $X_\ell = \theta_\ell^2$. Then for any λ and $\ell \to \infty$, we use variables $x(\lambda)$ and $X(\lambda) = \lambda + (1-\lambda)x(\lambda)$. Equalities (3.1) and (3.8) then give

$$x(\lambda) = E\left(u_i^\infty\right) = F_c(X(\lambda), \sqrt{X(\lambda)})$$

which can be rewritten as (3.5).

This also gives estimate (3.3). Indeed, iterations (3.6) and (3.7) show that the original iteration for $\ell = 0$ gives r.v. h_i^1 that has Gaussian pdf $\mathscr{N}(2\lambda c, 4\lambda c)$. Then for any round $s = \lambda b$, r.v. $u_1 = m^{-1}\sum_{i>\mu s} u_{i|1}$ has the mean $F_c(\lambda c, \lambda c) = R(\lambda c)$ and the vanishing variance $\mathscr{D} = R(\lambda c)/(1-\lambda)m$, where $R_c(x)$ is defined in (2.47). Thus, for any $\lambda > 0$, our iterations begin with the crossover probability $P_0 = \Pr\{u_1 \le 0\} \to 0$ as $m \to \infty$. The latter implies that $P_\ell \to 0$ for $\ell \to \infty$, as defined in (2.53). In turn, we can remove $P_\infty = 0$ from (2.26). Now we can use r.v.

$h_{i|\ell+1}$ that have pdf $\mathcal{N}(2X_\ell c, 4X_\ell c)$, according to (3.6) and (3.7). For $\ell \to \infty$, this gives (3.3) as

$$P_{soft}(\lambda, c) = \Pr\{h_{i|\infty} < 0\} \sim Q\left(\sqrt{X(\lambda)c}\right) \tag{3.10}$$

■

The absence of random walks in our current setup also makes bound (3.3) very tight. This is shown in Figure 3.2, where we plot analytical BER of (3.3) along with simulation results obtained for the algorithm $\Psi_{soft}(\lambda)$. Here we consider codes C_m with $m = 128$ and test various fractions of frozen bits $\lambda = s/m$ and different S/N ratios $10\log_{10}(c/4)$.

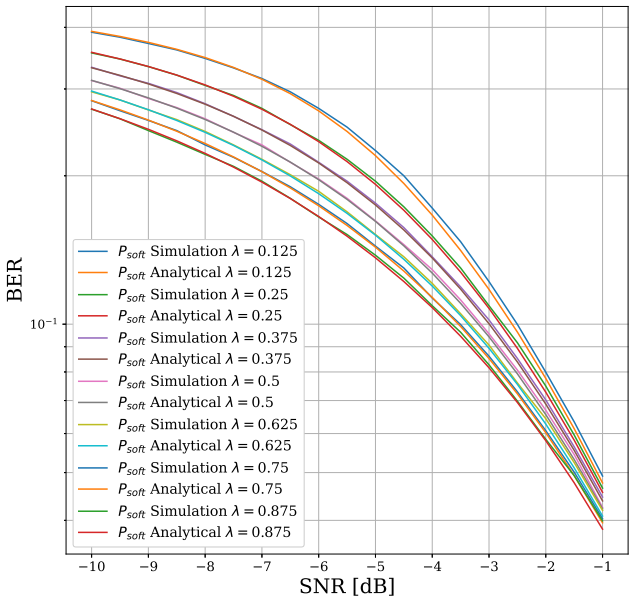

Figure 3.2: Simulation results and analytical bounds for the algorithm Ψ_{soft} applied to modulation-type codes C_{128} with a fraction λ of frozen bits.

It is also important to mention we can assume this frozen portion as a precoding for our LDPC code and estimate the asymptotic BER of the code for $m \to \infty$. Figure 3.2 and the bound (3.10) will allow us to have an accurate estimate of BER of this code a with a frozen portion λ (we assume λm bits out of m bits do not contain information and are frozen with value of 1). In Figure (3.3) we can see that these code can outperform uncoded modulation for any SNR larger

than -3.7 dB (this is the c where $x_* = 0.5$). In this case λ will affect the SNR and c and change it to $c_{new} = c(1-\lambda)$.

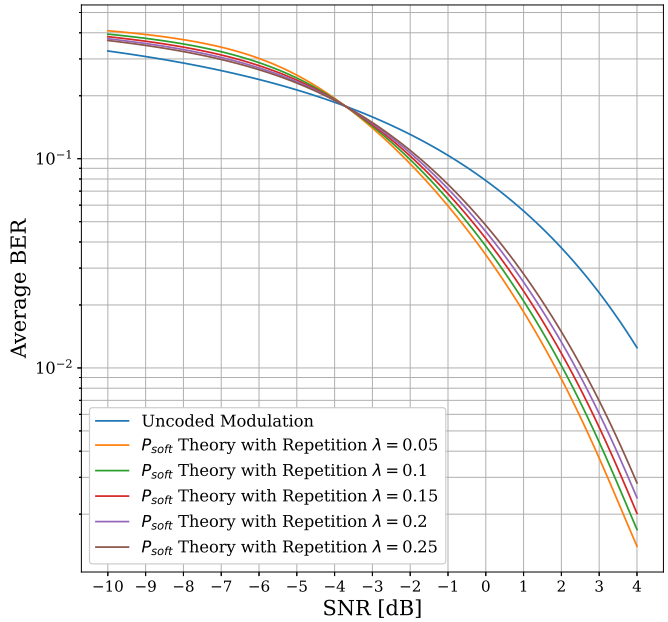

Figure 3.3: Analytical bounds for the algorithm with frozen information bits

Recall that the likelihoods $h_{i|L}(\lambda)$ give BER (3.3) in round $s = \lambda b$. We can now represent any Gaussian r.v. $h_{i|L}(\lambda)$ as a channel symbol that has pdf $\mathbb{N}(1,\sigma^2)$ and a BER $Q(1/\sigma)$. Thus, $\sigma^2 = 1/cX(\lambda)$. An important note is that codes $B_s(\mu,\mu r_s)$ now operate on the AWGN channels $\mathbb{N}(0,\sigma^2)$ that have a limited noise power $1/cX(\lambda)$. Unlike the original code C_m, we can now use codes $B_s(\mu,\mu r_s)$ of non-vanishing code rates that grow from r_0 to r_{b-1}.

Theorem 18 *Codes $\widehat{C}_m$ of dimension $k \to \infty$ and length $n = O(k^2)$ precoded with b polar codes have overall complexity of $O(n \ln n)$. For sufficiently large b, these codes achieve a vanishing BER if used arbitrarily close to the Shannon limit of -1.5917 dB per information bit.*

Proof. In round $s = \lambda b$, we use a capacity-achieving code $B_s(\mu,\mu r_s)$. The corresponding

BI-AWGN channel $\mathcal{N}_s(0,\sigma_s^2)$ has noise power $\sigma_s^2 = (X(\lambda)c)^{-1}$ and achieves capacity [69, 70]

$$\rho_c(\lambda) = \log_2 \sqrt{\frac{cX(\lambda)}{2\pi e}} - \int_{-\infty}^{\infty} f(y)\log_2 f(y)\, dy \tag{3.11}$$
$$f(y) = \sqrt{\frac{cX(\lambda)}{8\pi}}\left[e^{-(y+1)^2 cX(\lambda)/2} + e^{-(y-1)^2 cX(\lambda)/2}\right]$$

Here parameter λ changes from 0 to 1 in small increments $1/b$, which tend to 0 as $b \to \infty$. The average capacity for all AWGN channels $\mathcal{N}_s(0,\sigma_s^2)$ is $\rho_c = \int_0^1 \rho_c(\lambda)d\lambda$. Thus, for $m \to \infty$, code $\widehat{C}_m$ achieves a vanishing BER for any code rate $r < 2\rho_c/m$, which gives $SNR > c/4\rho_c$.

We now proceed with code complexity. For b polar codes $B_s(\mu, \mu r_s)$, design complexity has the order of $b\mu^2 \sim 2n/b$ or less. Their decoding requires the order of $b\mu\ln\mu < m\ln m$ operations. Algorithm Ψ_{soft} includes b rounds with $L = O(\ln m)$ iterations in each round. This gives complexity order of $n\ln n$ if b is a constant or $n\ln^2 n$ for growing $b < \ln m$. Thus, overall complexity has the order of $k^2 \ln k$, where $k \to \rho_c m$ is the number of information bits.

To calculate the minimum SNR $\varkappa = \min_c (c/4\rho_c)$, we select parameters c and b. Then we solve equation (3.4) for different values of $\lambda = s/b$, where $s = 0, ..., b-1$, and calculate ρ_c. The following table gives the highest value of code rate ρ_c, and the corresponding value of $\varkappa = \varkappa(c,b)$. Here we count $\varkappa$ in dB, as $10\log_{10}\varkappa$. The last line shows the gap $\varkappa/\ln 2 - 1$ to the Shannon limit of $\ln 2$.

b	10^2	10^3	10^4	25000
ρ_c	0.404	.3621	.3623	.3623
$\varkappa$ (in dB)	-1.5655	-1.5890	-1.5915	-1.5917
$\varkappa/\ln 2 - 1$	$6E-3$	$7E-4$	$6E-5$	$E-5$

Table 3.1: Relative gap to capacity as a function of number of divisions (b)

Finally, note that b is a constant for any $SNR > \ln 2$. Statement 1 now follows directly from the existing bounds [23] on BER for polar codes. Here polar codes B_i have length $\mu = m/b > 2k/b$.

3.3 Concluding remarks

In this chapter, we study new codes that can approach the Shannon limit on the BI-AWGN channels. We first employ "modulation " codes C_m that use parity checks of weight 3. These codes can be aided by other capacity achieving codes B_m via back-and-forth data recovery. Using BP algorithms that decode information bits only, codes C_m achieve complexity order of $n \ln n$. Then new analytical techniques give tight lower and upper bounds on the output BER, which are almost identical to simulation results. Finally, we employ multilevel codes of dimension $k \to \infty$ that approach the Shannon limit with complexity order of k^2. It is interesting that in the asymptotic case the number of overall information bits of the design approaches to $0.36k$. One open problem is to find out if there exists a close-form solution to the transcendental equations (3.4), which (unexpectedly) give the Shannon limit using numerical integration in (3.11).

It is important to highlight that if all the BP decoding part of LDPC code gets done in parallel, the latency of this capacity achieving design can be reduced to the latency of polar precoding stage which results in an overall latency of order ($\mathscr{O}(k)$). Latency of order ($\mathscr{O}(k)$) is much smaller than the latency of polar codes over the same block size which would be of order ($\mathscr{O}(n)$).

Chapter 4

Combined polar-LDPC design using Gallager ensemble of LDPC codes

4.1 Introduction

In this chapter, we consider code design for channels with high noise that can emerge in the Internet of things (IoT) for moderate block sizes. The design of [2] (multi-level design) becomes competitive only on the very large lengths $n \geq 2^{17}$. The main shortcoming is the very short length of m/b used in multilevel constructions even for small $b = 2,3$. Therefore, below we again take $b = 1$ and use LDPC codes with parity checks of weight w where $w \geq 2$. We show that a combination of repetition of information bits and parity checks of weight w have the ability of achieving low BER for low rate codes over moderate blocks. More importantly, the latency of this design is of order $\mathscr{O}(\|)$ where k is the number of information bits.

In section 4.2 we describes the basic sequence $C_{k,d,w}$ of the LDGM codes that have dimension k. These codes have systematic generator matrices, in which every parity-check column has fixed weight w. We only use $w \leq 4$ then we proceed with the basic decoding

algorithm in Chapter 4.3 and describe its enhancements in Chapter 4.4 and proceed with a two-stage design. Here we combine BP decoding of codes $C_{k,d,w}$ with a SCL decoding of the shorter polar codes. Simulation results are discussed in section 4.5. In particular, new codes of length 2^{13} achieve WER of about 10^{-4} at $SNR \sim 1$ dB using the lists of 32 candidates for polar codes. This yields a 0.75 dB gain over the designs of [1, 49]. However, the latter gain comes with a much higher dimension of about $k \approx 400$ that simplifies code design compared to [1, 49]. We propose 4 different algorithm and compare their complexity, latency and performance. Using simulation results we compare these 4 algorithms with each other and compare the best version of them with CA-Polar codes. The new two-stage codes presented below are also on par with the best CA-Polar codes that have similar parameters. Here we substantially reduce the latency of SCL decoding over the same block size and rate.

4.2 Basic construction

Let e_i denote a binary column of some length k' that has a single symbol one in the position $i \in \{1,...,k'\}$ only. We then consider a $k' \times k'$ identity matrix $I_{k'}$ and its cyclic permutation $P_{k'}^s = [e_{s+1}, \cdots e_{k'}, e_1, \cdots, e_s]$, where $s \in \{0,...,k'-1\} (\mathrm{mod}\, k')$. Here $P_{k'}^0 = I_{k'}$. Now consider integer parameters k', d, and $2 \leq w < \min\{k', d\}$. We then form the $(k'w \times k'd)$-matrix

$$J_{k,d,w} = \begin{bmatrix} I_{k'} & I_{k'} & \cdots & I_{k'} \\ P_{k'}^0 & P_{k'}^1 & \cdots & P_{k'}^{d-1} \\ P_{k'}^0 & P_{k'}^2 & \cdots & P_{k'}^{2d-2} \\ \cdots & \cdots & \cdots & \cdots \\ P_{k'}^0 & P_{k'}^{w-1} & \cdots & P_{k'}^{(w-1)(d-1)} \end{bmatrix} \tag{4.1}$$

Each column in matrix $J_{k,d,w}$ has weight w and each row has weight d. Note also that columns of the matrix $J_{k,d,w}$ are pairwise different if $d \le k'$ but some are repeated at most $\lceil d/k' \rceil$ times if $d > k'$. Next, let $C_{k,d,w}$ be a systematic LDPC code with the generator matrix $G_{k,d,w} = [I_k | J_{k,d,w}]$. This code has dimension $k = k'w$ and length $n = k + k'd$. Its parity check matrix $H_{k,d,w} = J_{k,d,w}^T \, | \, I_{n-k}$ has rows of weight $w+1$. Below we only consider low-rate codes $C_{k,d,w}$ that have small parameter $w = 2,3,4$ and a large parameter d.

Remarks. Systematic codes $C_{k,d,w}$ also form low-density generator matrix (LDGM) codes [71]. Polar-based LDGM codes of [72] may achieve the capacity of the binary memoryless channels if the columns of generator matrices have weight $(\log n)^{1.18}$ or more. Some high-rate LDGM codes that use circulant matrices P_j^i are considered in [73]. Our previous construction [2,51] differs from (4.1) and forms each parity bit as a binary sum of $p = 2$ information bits. It was shown that BP gives the output BERs that is equivalent to that of maximum likelihood (ML) decoding for large m. A more general construction of [74] analyzed ML performance of such codes for any $p \ge 2$.

Next, consider the parity check equations generated by matrix (4.1). Let $a = (a_1, \cdots, a_k)$ denote the string of k information bits a_i encoded by the generator matrix $I_k | J_{k,d,w}$ into the string $a_1, \cdots, a_n$. Consider the subset $J(i)$ of d columns that contain 1 in row i. For two rows $i, i' \in [1,k]$ and any column $j \in [1,n]$, we write $\delta_j(i,i') = 1$ if both rows of the generator matrix $G_{k,d,w}$ have common 1 in column j. Otherwise, $\delta_j(i,i') = 0$. Each column $j \in J(i)$ forms a parity check with respect to the information bit a_i. The first such column $j = i$ forms a trivial parity check $a_i = a_i$. For any $j > k$, the remaining $d-1$ parity checks include one parity bit a_j and $w-1$ information bits $a_{i'}$ such that $\delta_j(i,i') = 1$. Thus, each information bit a_i satisfies d parity checks

$$a_i = a_j + \sum_{i' \, : \, \delta_j(i,i')=1} a_{i'}, \quad \forall j \in J(i) \tag{4.2}$$

4.3 BP algorithm for parity check of weight w

Let code $C_{k,d,w}$ be used on an AWGN channel with a pdf. $\mathcal{N}(0,\sigma^2)$ and a constant $SNR = \left(2\sigma^2 R\right)^{-1}$ per information bit. We use a map $\{0,1\} \to \{\pm 1\}$ for each transmitted symbol a_i, $i \in [1,n]$. Then the parity checks (4.2) form the real-valued products

$$a_i = a_j \prod\nolimits_{i' \,:\, \delta_j(i,i')=1} a_{i'}, \quad j \in J(i) \tag{4.3}$$

Let an all-one codeword 1^n be transmitted. Then the received symbols y_j form independent Gaussian r.v. $\mathcal{N}(1,\sigma^2)$. In decoding, we will use rescaled r.v. $z_j = y_j/\sigma^2$. Given some z_j, an input $a_j = 1$ has posterior probability

$$q_j \triangleq \Pr\{1 \mid z_j\} = (e^{-2z_j}+1)^{-1}$$

Decoding algorithm $\Psi_{soft}(z)$ described below employs two closely related quantities, the log-likelihoods (l.l.h.) h_j and the "probability offsets" u_j.

$$\begin{aligned} h_j &= \ln[q_j] - \ln[1-q_j] = 2z_j \\ u_j &= 2q_j - 1 = \tanh(z_j) \end{aligned} \tag{4.4}$$

Algorithm Ψ_{soft} performs several rounds of belief propagation. We use the conventional BP algorithm of [16], with the only difference that our recalculations are limited to information symbols only. This is used to reduce complexity of the BP decoding. Our simulation results showed that this extra restriction caused only a slight degradation of code performance. Similarly to [2], we use $\mu = \mathcal{O}(\ln d)$ iterations to achieve the best performance.

For every step $\ell = 1,\ldots,\mu$ and every symbol a_i, consider its j-th parity check of (4.3). For each information bit $a_{i'}$, every decoding round ℓ will employ its offset $u_{i|\ell}(i')$, which will be

recalculated in the next round. For each $i = 1, ..., k$ and each $j \in J(i)$, we can now rewrite (4.3) using the offsets $u_{i|\ell}(i')$:

$$u_{i|\ell+1}(j) = u_j \prod_{i' \,:\, \delta_j(i,i')=1} u_{i|\ell}(i'), \quad j \in J(i) \tag{4.5}$$

Similarly to conventional BP decoding [16], we then obtain the l.l.h. $h_{i|\ell+1}(j)$ of the j-th parity check. Now we can derive the compound estimate

$$h_{i|\ell+1} = \sum_{j \neq i} h_{i|\ell+1}(j) \tag{4.6}$$

of the symbol a_i. The algorithm Ψ_{soft} begins with the original estimates $u_{i|0}(j) \triangleq u_j = \tanh(z_j)$ in iteration $\ell = 0$ and proceeds in round ℓ as follows.

For each $i \in [1,k]$ and each $j \in J(i)$:

A. Derive the offsets $u_{i|\ell+1}(j)$ of (4.5)

and l.l.h. $h_{i|\ell+1}(j) = 2\tanh^{-1}\left[u_{i|\ell+1}(j)\right]$.

B. Derive full l.l.h. $h_{i|\ell+1}$ of (4.6) and

partial l.l.h. $h_{j|\ell+1}(i) = h_{i|\ell+1} - h_{i|\ell+1}(j)$

C. If $\ell < \mu$, find $u_{i|\ell+1}(j) = \tanh(h_{i|\ell+1}(j)/2)$.

D. Go to A with $\ell := \ell + 1$. If $\ell = \mu$ go to B

and find BER $\tau_\mu = \frac{1}{m}\sum_i \Pr\{h_{i|\mu} < 0\}$;

output numbers $h_{i|\mu}$ and $a_i = \text{sign}\,(h_{i|\mu})$. (4.7)

To estimate the complexity $\Phi_{k,d,w}$ of the algorithm Ψ_{soft}, note that Step *A* uses $(d-1)(w-1)$ multiplications and d conversions into $h_{i|\ell+1}(j)$ for each of k information bits a_i.

For each i, step B uses d parity checks $j \in J(i)$. Here we use $d-1$ operations to find the full sums $h_{i|\ell+1}$ and $d-1$ operations to find the partial sums $h_{i|\ell+1}(j)$. Then step C uses d conversions into $u_{i|\ell+1}(j)$. Taking all k symbols for each iteration ℓ, we obtain complexity order $\Phi_{k,d,w} \sim kd(w+4)$. Given the length $n = kd/w + k$, we obtain $\Phi_{k,d,w} \sim (n-k)w(w+4)$. Following the analysis of [2], we further assume that we need $\mu \sim 2\log_2 k$ iterations. Then

$$\Phi_{k,d,w} \sim 2nw(w+4)\log_2 k \tag{4.8}$$

In Figure (4.1) we can see the results of this algorithm for a number of different parameters $k = \{256, 512, 1024\}$ and $w = 2$. These codes are similar to code C_m that are introduced in Chapter 2 in terms of weight of parity checks but are different due to their selection of parity checks. They basically select a portion of all possible parity checks and subsequently will have a higher rate, r, and lower distance, d, over the same block size. Higher rate of these codes will reduce the probability of high weight errors that was discussed in Chapter 2.

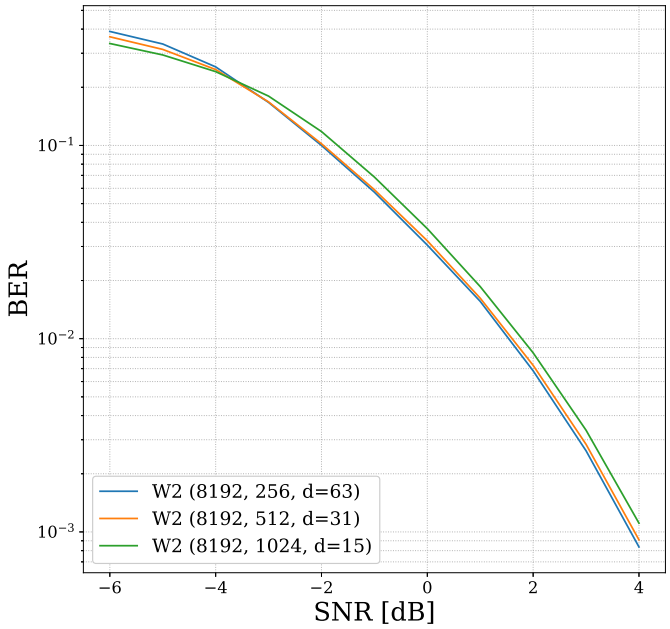

Figure 4.1: BER of BP algorithm for a $(8192, k)$ code

4.4 Joint polar-LDPC coding

One significant shortcoming of the above codes $C_{k,d,w}$ is their heavy reliance on information bits in the BP decoding Ψ_{soft}. Indeed, each parity check (4.5) forms a sequence of degrading channels that multiply $w-1$ information offsets $u_{i|\ell}(i')$ and only one parity-bit offset u_j. We will now reduce the BER of BP decoding by repeating all information bits s times, where we take small $s \leq 5$. Clearly, using multiple copies, we can make all received symbols z_k more reliable. The resulting code $C_{k,d,w}(s)$ has a generator matrix that includes s copies of I_k :

$$G_{k,d,w}(s) = [I_{k,1}|...|I_{k,s}|J_{k,d,w}] \tag{4.9}$$

This gives the length $n_w(s) = k'd + sk = k(d/w+s)$ and reduces the code rate R by a factor $n_w/n_w(s) = (d/w+1)(d/w+s)$. For the same SNR, we now need to handle a slight increase in noise power σ^2. Note, however, that we use parameter $d \gg ws$, in which case σ^2 undergoes only a slight increase. On the other hand, we can now begin BP decoding using s copies $a_i^{(1)}$,...,$a_i^{(s)}$ of any symbol a_i and combine s l.l.h. of the symbol a_i into a single estimate

$$h_i(s) := \sum_{p=1}^{s} h_i^{(p)}$$

We then convert $h_i(s)$ into $u_i(s) = \tanh(h_i(s)/2)$. Now the first round of our decoding $\Psi_{soft}(s)$ employs the s-fold estimates $u_{i|0}(j) \triangleq u_j(s)$ instead of the single estimates u_j. The following rounds are identical to the algorithm Ψ_{soft}.

Our simulation results in Figure 4.2 also indicate that the BER of BP decoding can be reduced 2 to 3 times by taking $s = 1,...,4$. To make codes $C_{k,d,w}(s)$ have the same length N, we will perform one more modification. Namely, we will keep s copies of identity matrix I_k in (4.9) but will reduce the number d of blocks $I_{k'}$ to keep the former length $n = k(d/w+1)$.

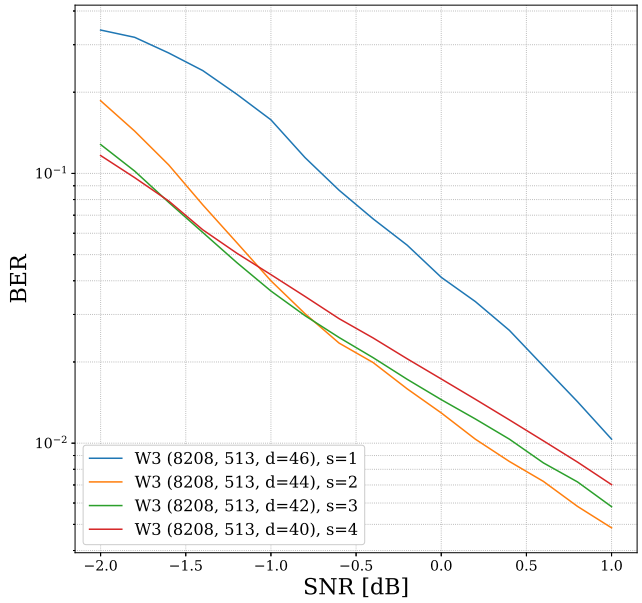

Figure 4.2: BER of BP algorithms for $w = 3$ and code $(8208, 513)$ with different number of repetitions $s = \{1, \ldots, 4\}$.

We will now combine any LDPC code $C_{k,d,w}$ with some polar code $P[k, kr]$ that has length k and code rate r, described in Figure (4.3). Recall that the algorithm Ψ_{soft} outputs l.l.h. $h_{i|\mu}$ of k bits a_i obtained after μ iterations. These l.l.h. will be passed to a SCL decoder of the polar code. Following the theoretical analysis of [2], we assume here that l.l.h. $h_{i|\mu}$ can be scaled as independent Gaussian random variables obtained by sending quantities ± 1 over a Gaussian channel with pdf. $\mathcal{N}(0, \sigma^2)$. Unlike [2], this assumption only follows simulation results without theoretical justification.

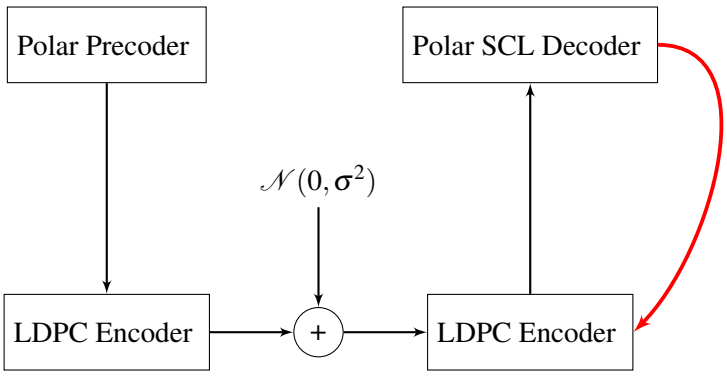

Figure 4.3: Polar-LDPC Design

Let $Q(x)$ denote the cumulative density function of the Gaussian pdf. $\mathcal{N}(0,1)$ in the interval $(-\infty,-x)$. In our model with a Gaussian pdf. $\mathcal{N}(0,\sigma^2)$, r.v. $h_{i|\mu}$ have some BER P that can be regarded as $Q(1/\sigma)$. Thus, BER P of a BP decoder defines the input noise power

$$\sigma^2 : Q(1/\sigma) = P$$

of the SCL decoder. Then we consider any polar code $P[k,\, kr]$ on the input channel $\mathcal{N}(0,\sigma^2)$. However, adding a polar code of some rate $r < 1$ reduces the overall SNR by a factor r. Thus, to keep the overall SNR at a given value c, a polar code $P[k,\, kr]$ must operate on the Gaussian channel $\mathcal{N}(0,\sigma^2/r)$.

Let $\mathrm{Pr}_P(k,r,\sigma^2)$ denote the BER of a polar code $P[k,\, kr]$ on a Gaussian channel $\mathcal{N}(0,\sigma^2/r)$. Then we can consider different codes $P[k,\, kr]$ and select the optimal rate

$$r_* = \min_r \mathrm{Pr}_P(k,r,\sigma^2)$$

Here we use one of the algorithms [67] or [75] to construct some polar code $P[k,\, kr]$. One drawback of this procedure is that these programs estimate codes $P[k,\, kr]$ using SC decoding with the list size $L = 1$, whereas we use larger lists, typically, $L = 32$. We will also use various CRC checks introduced in [76, 77].

Below we consider polar codes of length $k = 512$. We will then combine some polar code $P[k,\, kr]$ with the LDPC code $C_{k,d,w}$ of a larger length $n \approx 8192$. For these codes, we will compare four different algorithms that use joint BP-SCL decoding. Two of them, A and B, use no CRC checks, while two others use different CRCs. Also, two algorithms (A and C) will perform a single run of BP decoding, while two others (B and D) will make two runs.

Our basic algorithm A performs BP decoding and then proceeds with the SCL decoding

that processes $L = 32$ candidates. All L final candidates of length kr are encoded by the chosen polar code $P[k, kr]$ of length k. The obtained vectors $a^{(1)}, ..., a^{(L)}$ are then encoded to the full length n. Finally, we select the best (most probable) candidate $c \in C_{k,d,w}$. Thus, we replace inspection of codewords of length k in SCL decoding with ML decoding in the extended list of L candidates.

Decoding complexity of algorithm A is dominated by the complexity $\Phi_{k,d,w} \sim 2w(w+4)n\log_2 k$ of the BP decoding (4.8). Indeed, SCL decoding has much smaller complexity $\Phi_{k,L} \leq 4kL\log_2 k$ (SCL decoding of [78] meets this bound). Also, encoding to the length n requires about Lnw binary operations. Selection of the best candidate requires Ln operations with real numbers. Both terms are still small relative to $\Phi_{k,d,w}$. A slight reduction in complexity can also be obtained by encoding $L' < L$ of the best candidates to the full length n. Note also that the parallel BP decoding only depends on the number of iterations and has the latency of order $\log n$. The overall latency is then dominated by the latency k of SCL decoding. This compares favorably to the SCL decoding of polar codes of length n that have latency of order n.

Our second algorithm B trades off BER for the higher complexity by performing the second run of BP decoding. Here all L candidates $a^{(1)}, ..., a^{(L)}$ obtained by the SCL decoder are again encoded to the full length n. Let $c \in C_{k,d,w}$ be the most probable candidate and $a = (a_1, ..., a_k) \in P[k, kr]$ be its information set of length k. We then replace the received symbols $y_1, ..., y_k$ with the hard-decision inputs $a_1, ..., a_k$ and perform the second round of BP/SCL decoding. The best candidate $c' \in C_{k,d,w}$ is selected similarly to the first run. Finally, we select the better candidate of c and c'. One possible advantage of this approach is due to the fact that the second round of SCL decoding can further eliminate some errors left in first round. Simulation results show that the second round indeed reduces the output WER; however, the third round brings virtually no improvements. Algorithm B has complexity order of $2\Phi_{k,d,w}$.

Our third algorithm C is similar to algorithm A but also performs the CRC checks on

the length k. It proceeds with the final inspection of the best remaining candidates on the full length n. Given a CRC-q check of degree q, we use notation $P[k,kr,q]$ for a polar code that has kr information bits and uses q other bits for the CRC check. Below we select q=11, 7, and 4, and use polynomials $x^{11}+x^{10}+x^9+x^5+1$, x^7+x^4+1, and x^4+x+1. We first execute the BP-SCL decoding of Algorithm A, and then perform CRC-11 check for all 32 candidates. Let $a^{(1)},...,$ $a^{(L')}$ denote the list of $L' \leq 6$ most probable candidates left after the CRC-11 check. Here we keep the results of BP decoding if $L'=0$. Otherwise, we again encode all L' candidates to the full length n and select the best candidate $c \in C_{k,d,w}$. Note that CRC check has low complexity of order Lkq. Thus, algorithm C has complexity order $\Phi_{k,d,w}$.

The fourth algorithm D uses the following observation. Consider some polar code P. Let W^- denote some degrading channel of the SC decoding, W^+ be some upgrading channel, and $W=W^-,W^+$ be either of them. For any polar code, generic SC decoding is first done on all $k/2$ "one-bit downgraded" sequences $W(-)=W_1^-,W_2,...,W_t$ of depth $t=\log_2 k$. It then proceeds with $k/2$ "one-bit upgraded" sequences $W(+)=W_1^+,W_2,...,W_t$. In terms of the Plotkin $u,u+v$ construction, SC decoder first finds the vector v on the set $W(-)$ and then proceeds with vector u on the set $W(+)$. Here the two corrupted copies of vector u are combined together on the first channel W_1^+.

Let $P=P[k,kr,q_1,q_2]$ be a polar code that employs a separate CRC-q_1 check on $W(-)$ and then CRC-q_2 check on $W(+)$. Let $(u_*,u_*+v_*) \in P$ be the original vector encoded into the code $C_{k,d,w}$ and y be the received vector of length n. Algorithm D performs BP decoding of vector y. It proceeds with SCL decoding of code P but stops after processing all sequences $W(-)$. We then select $s \leq 6$ most probable candidates $v=v(1),...,v(s)$ that satisfy the CRC-q_1. Let $\widetilde{v}$ be the most probable among them. We then proceed with SCL decoding on the remaining set $W(+)$ starting with only s candidates $(0,v)$ left on $W(-)$. Upon completion, we apply CRC-q_2 to all obtained vectors u and form the list of $L' \leq 6$ most probable candidates $(u,u+v)$ on the length k.

These may include different vectors u and v. We again encode these candidates to the length n and select the best candidate c'. To this end, algorithm D is similar to B.

However, we will also apply extra processing to the most probable candidate $(0,\widetilde{v})$ obtained after CRC-q_1. Here we encode this vector into some code vector $c \in C_{k,d,w}$. First, let us assume that vector y is not corrupted by noise. Then we would obtain $\widetilde{v} = v_*$ and encode vector $(0,v_*)$ into the correct vector c. In this case, yc is formed by encoding vector u_*, u_* and includes k information bits $a_1,...,a_k$, which form 2-bit couples $a_j = a_{k/2+j}$ for all $j = 1,...,k/2$.

Next, consider the noisy versions of vectors y and yc. Let $\widetilde{a}_1,...,\widetilde{a}_k$ denote the first k l.l.h. of yc. We then replace these l.l.h. with k new l.l.h. $\widehat{a}_j = \widehat{a}_{k/2+j} = \widetilde{a}_j + \widetilde{a}_{k/2+j}$. Then BP decoding proceeds using the new vector yc. In each iteration $i = 2,...,\mu$, we repeat the coupling $\widehat{a}_j^{(i+1)} = \widehat{a}_{k/2+j}^{(i+1)} = \widehat{a}_j^{(i)} + \widehat{a}_{k/2+j}^{(i)}$. The obtained vector of k l.l.h. is then processed with SCL decoding on the sequences $W(+)$. All possible candidates undergo a CRC-q_2.

We again select $L'' \leq 6$ most probable candidates $\widetilde{u}''$. Then vectors $(\widetilde{u}'', \widetilde{u}'' + \widetilde{v})$ are encoded to the full length n. Let c'' be the most probable codeword among them. Finally, we select the most probable candidate among the two remaining vectors, c' and c''. Clearly, algorithm D has complexity order of $2\Phi_{k,d,w}$, since we perform two rounds of BP-SCL decoding.

Remark. All channels $W(+)$ can be further decomposed into two subsets: $W(+-) = W_1^+, W_2^-, ...$, and $W(++) = W_1^+, W_2^+, ...$ Thus, we can similarly return to BP decoding after SCL decoding on channels $W(+-)$. However, in our simulations multiple returns to BP decoding showed little improvement in BER while increasing complexity of algorithm D. We also avoided the second round of BP decoding for all but the best candidate $(0,\widetilde{v})$ due to the extra complexity.

4.5 Simulation results

The above algorithms A, B, C, and D were considered for two polar-LDPC codes. The first design uses an LDPC code $C_{k,d,w}(s)$ with parameters

$$k = 513,\ w = 3,\ s = 3,\ d = 39$$

Following (4.1) and (4.9), note that this code is formed using sub-matrix $J_{k,d,w}$ that consists of circulants $P^i_{k'}$ with $k' = 171$. This gives an $[8208, 513]$ code. The second design uses an LDPC code $C_{k,d,w}(s)$ with parameters

$$k = 512,\ w = 4,\ s = 5,\ d = 44$$

Here matrix $J_{k,d,w}$ consists of circulants $P^i_{k'}$ with $k' = 128$. This gives an $[8192, 512]$ code. We then select polar codes of length k using design of [67]. Here we also employ the CRC-11 in algorithm B and two CRC checks of degrees $q_1 = 7$ and $q_2 = 4$ in algorithm D. The selected polar codes have dimensions $k_1 = 400$ and $k_2 = 450$. This gives polar-LDPC codes A_1[8208,400] and A_2[8192,450]. In Figure (4.4) and (4.5) we present simulation results for these two codes. Interestingly, algorithms A and B slightly outperform algorithms C and D that employ CRC checks (both C and D) and half-way correction (D). Here the CA-Polar codes used in algorithms C and D improve the selection of the best candidates; however, our simulation shows that extending this process to the full length n also performs efficient selection in algorithms A and B. CA-Polar codes of algorithms C and D also have better weight spectra; however, extra parity checks used here cause a slight channel degradation.

Observe also that algorithm D does not improve on algorithms A and B despite a more powerful error correction of vectors u on the channel set $W(+)$. This is due to the fact that BP

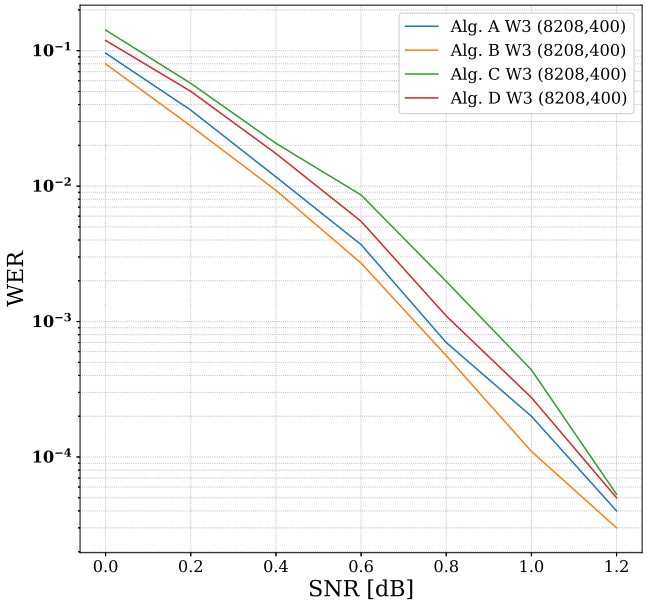

Figure 4.4: WER of algorithms A, B, C, and D of SCL-BP decoding for a (8192,400)-code with $w = 3$.

decoding - if incorrect - often keeps possible vectors v substantially corrupted. In this case, SCL decoding fails to keep the correct vector v_* among the survived candidates. Our simulation results confirmed the fact that incorrect decoding of vectors v in the $(u, u+v)$ construction is indeed the main source of the remaining errors.

In Figure 4.6, we also compare the WER of code A_1[8208,400] with three other recently constructed codes: two [8192,80]-codes, A_3 of Figure 4a in [1] and A_4 of [2], and a CA-Polar code A_5[8192,400] optimized using the software package [3]. Here codes A_1 and A_3 use polar codes of length 512 as their components, while code A_4 uses a polar code of length 128, which reduces decoding latency about 4 times. All four codes use the lists of size $L = 32$. We see that an LDPC-polar code A_1 achieves the $WER \sim 10^{-4}$ at the $SNR = 1$ dB, which is on par with the CA-Polar code A_5. This code also substantially reduces decoding latency of A_5, due to the much shorter polar code of length 512 used in A_1 instead of the length 8192 of code A_5.

We also see that code A_1 gains about 0.75 dB at the output WER of 10^{-4} compared to

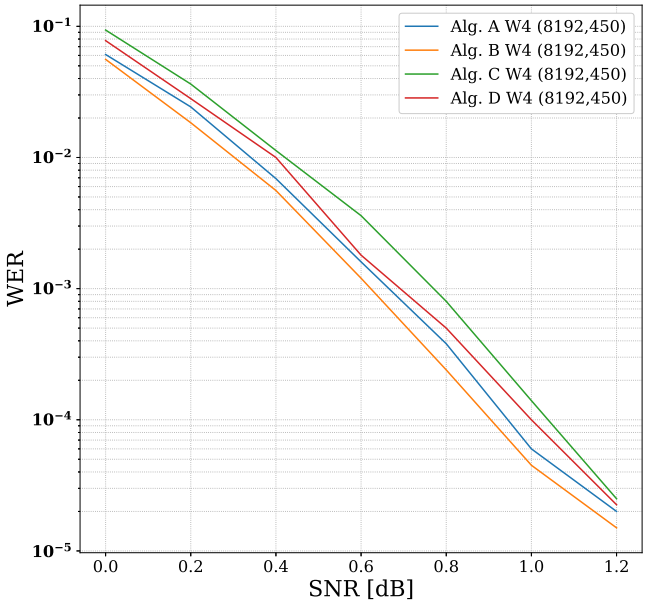

Figure 4.5: WER of algorithms A, B, C, and D of SCL-BP decoding for a (8192,450)-code with $w = 4$.

the code A_3 [1]. Note, however, that this improvement is achieved at a much higher code rate of about 0.05 instead of the rate below 0.01 used in [1, 2]. Our conjecture is that for a given length $n \sim 2^{13}$, codes of a rate 0.05 may achieve a much lower BER than the lower-rate codes. We also assume that this conjecture is valid for ML decoding or SC decoding or BP decoding when different codes are used on the AWGN channels with the same SNR.

4.6 Concluding remarks

We introduced a general low rate LDPC code $C_{k,d,w}$ with parity checks of weight w with flexible size and distance and presented simulation results for their BER performance. We studied joint decoding of long low rate LDPC codes $C_{k,d,w}$ combined with short polar codes. We used the results of polar SCL decoder to improve the estimate of LDPC BP decoder in the second run a subsequently improve the overall performance of this code . Two step decoding algorithms increase the complexity of decoding but reduce WER for these codes. This design

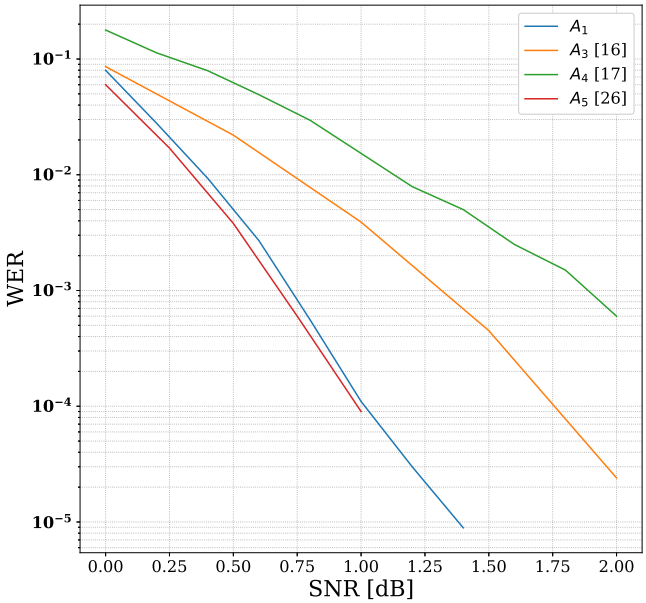

Figure 4.6: WER of four codes: (8208,400) polar-LDPC code A_1, (8192,80) codes A_3 [1] and A_4 [2], and (8192,400) polar code A_5 [3].

yielded performance that is similar to that of SCL decoding of polar codes, while substantially reducing decoding latency.

Chapter 5

Conclusions and future work

In this thesis, we introduced low latency and low rate communication schemes that can achieve the Shannon limit of a vanishing rate on the BI-AWGN channels. We first derived theoretical boundaries for the probability of error for these codes and then modified our design to consider practical cases of communication on limited block sizes and complexity. We demonstrated that these new designs are on a par with CA-Polar codes in terms of performance and complexity. For similar rates over the same block sizes, they achieved the same WER with much smaller latency.

In Chapter 2, we introduced the modulation binary codes $C_{m,s}$ and described their encoding and BP decoding algorithms on BSC and AWGN channels. We also presented boundaries for BER of ML and BP algorithm and compared them with simulation results. We showed that this simple design had the ability to outperform uncoded modulation for any $SNR > -3$ dB. Then we used a polar code as precoding to further protect information bits and showed that this design achieves exponential decline for WER on moderate $SNR < 2$ dB per information bit but is unable to achieve the Shannon limit of -1.59 dB.

In Chapter 3, we introduced a multilevel protection scheme that used a number of

polar precoders with different rates and combined them with modulation code C_m. Then we presented boundaries for BER of this design with frozen information bits and showed that they are practically identical to simulation results. Using these tight boundaries and capacity of the BI-AWGN channel we demonstrated that this back and forth decoding algorithm can get as close as we want to the Shannon limit of vanishing rates for a large number of polar precoders.

In Chapter 4, we proposed a design for low rate LDPC codes with parity checks of small weight. This design can achieve better WERs than the polar-modulation codes, described in Chapter 2, over the same block size. However, it is crucial to note that these codes have higher rates and bigger decoding complexity than the previous design. We also offered a novel decoding scheme for these designs by incorporating their polar-LDPC structure and provided simulation results that are better than or on a par with the state of the art algorithms such as polar coded repetition [1] codes and CA-Polar codes [3].

There are several open questions that could lead to interesting research problems. In order to prove bounds on BER we assumed that log-likelihoods of partial terms in each iteration are weakly dependent and used this assumption to prove our bounds. These bounds are practically identical to the results of simulations; however, we did not prove their weak independence. We also showed numerically that for a large number of capacity achieving polar precoders we can achieve the Shannon limit of vanishing rates but did not present closed-form formulas. Finally, we believe that the combination of parity checks of weight w where the majority of parity checks are of weight $1, 2$ and a small portion of the parity checks are of higher weights can actually achieve a much faster decline in WER on moderate blocks sizes.

Bibliography

[1] F. Abbasiy, H. Mahdavifarz, and E. Viterboy, "Polar coded repetition for low-capacity channels," in *2020 IEEE Information Theory Workshop (ITW)*, pp. 1–5, IEEE, 2021.

[2] I. Dumer and N. Gharavi, "Codes approaching the shannon limit with polynomial complexity per information bit," in *2021 IEEE International Symposium on Information Theory (ISIT)*, pp. 238–243, 2021.

[3] K. Shabunov, "Error correcting coding research tools." `https://github.com/kshabunov/ecclab`, 2019.

[4] C. E. Shannon, "A mathematical theory of communications," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[5] E. Berlekamp, R. McEliece, and H. van Tilborg, "On the inherent intractability of certain coding problems (corresp.)," *IEEE Transactions on Information Theory*, vol. 24, no. 3, pp. 384–386, 1978.

[6] R. W. Hamming, "Error detecting and error correcting codes," *The Bell System Technical Journal*, vol. 29, no. 2, pp. 147 – 160, 1950.

[7] I. Reed, "Class of multiple-error-correcting codes and the decoding scheme," *Transactions of the IRE Professional Group on Information Theory*, vol. 4, no. 4, pp. 38–49, 1954.

[8] D. E. Muller, "Application of boolean algebra to switching circuit design and to error detection," *Transactions of the I.R.E. Professional Group on Electronic Computers*, vol. 3, no. 3, pp. 6–12, 1954.

[9] R. Bose and D. Ray-Chaudhuri, "On a class of error correcting binary group codes," *Information and Control*, vol. 3, no. 1, pp. 68–79, 1960.

[10] A. Hocquenghem, "Codes correcteurs d'erreurs," *Chiffers*, vol. 2, pp. 147–156, 1959.

[11] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *Journal of The Society for Industrial and Applied Mathematics*, vol. 8, pp. 300–304, 1960.

[12] P. Elias, "Coding for noisy channels," *IRE Conv. Rec.*, vol. 3, pp. 37–46, 1955.

[13] P. Elias, "Error-free coding," *Transactions of the IRE Professional Group on Information Theory*, vol. 4, no. 4, pp. 29–37, 1954.

[14] G. Froney Jr, "Concatenated codes," tech. rep., MASSACHUSETTS INST OF TECH CAMBRIDGE RESEARCH LAB OF ELECTRONICS, 1965.

[15] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Proceedings of ICC '93 - IEEE International Conference on Communications*, vol. 2, pp. 1064–1070, 1993.

[16] R. Gallager, "Low-density parity-check codes," *IRE Transactions on information theory*, vol. 8, no. 1, pp. 21–28, 1962.

[17] D. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 399–431, 1999.

[18] M. Luby, M. Mitzenmacher, M. Shokrollahi, and D. Spielman, "Efficient erasure correcting codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 569–584, 2001.

[19] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 599–618, 2001.

[20] A. Jimenez Felstrom and K. Zigangirov, "Time-varying periodic convolutional codes with low-density parity-check matrix," *IEEE Transactions on Information Theory*, vol. 45, no. 6, pp. 2181–2191, 1999.

[21] S.-Y. Chung, G. Forney, T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 db of the shannon limit," *IEEE Communications Letters*, vol. 5, no. 2, pp. 58–60, 2001.

[22] D. J. MacKay and R. M. Neal, "Near shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 32, pp. 1645–1646, 1996.

[23] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.

[24] E. Arikan, "Source polarization," in *2010 IEEE International Symposium on Information Theory*, pp. 899–903, 2010.

[25] E. Abbe, "Polarization and randomness extraction," in *2011 IEEE International Symposium on Information Theory Proceedings*, pp. 184–188, 2011.

[26] M. Mondelli, S. H. Hassani, I. Sason, and R. L. Urbanke, "Achieving marton's region for broadcast channels using polar codes," *IEEE Transactions on Information Theory*, vol. 61, no. 2, pp. 783–800, 2015.

[27] N. Goela, E. Abbe, and M. Gastpar, "Polar codes for broadcast channels," *IEEE Transactions on Information Theory*, vol. 61, no. 2, pp. 758–782, 2015.

[28] H. Mahdavifar, M. El-Khamy, J. Lee, and I. Kang, "Achieving the uniform rate region of general multiple access channels by polar coding," *IEEE Transactions on Communications*, vol. 64, no. 2, pp. 467–478, 2016.

[29] E. Şaşoğlu, E. Telatar, and E. M. Yeh, "Polar codes for the two-user multiple-access channel," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6583–6592, 2013.

[30] Y.-P. Wei and S. Ulukus, "Polar coding for the general wiretap channel with extensions to multiuser scenarios," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 278–291, 2016.

[31] H. Mahdavifar and A. Vardy, "Achieving the secrecy capacity of wiretap channels using polar codes," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6428–6443, 2011.

[32] M. Seidl, A. Schenk, C. Stierstorfer, and J. B. Huber, "Polar-coded modulation," *IEEE Transactions on Communications*, vol. 61, no. 10, pp. 4108–4119, 2013.

[33] H. Mahdavifar, M. El-Khamy, J. Lee, and I. Kang, "Polar coding for bit-interleaved coded modulation," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 5, pp. 3115–3127, 2016.

[34] R. Ratasuk, N. Mangalvedhe, Y. Zhang, M. Robert, and J.-P. Koskinen, "Overview of narrowband iot in lte rel-13," in *2016 IEEE conference on standards for communications and networking (CSCN)*, pp. 1–7, IEEE, 2016.

[35] 3GPP, "5g; study on scenarios and requirements for next generation access technologies," Technical Report (TR) 38.913, 3rd Generation Partnership Project (3GPP), 2017. Version 14.3.0.

[36] Y. Z. M. R. R. Ratasuk, N. Mangalvedhe and J. Koskinen, "Overview of narrowband iot in lte rel-13," *2016 IEEE Conference on Standards for Communications and Networking (CSCN)*, pp. 1–7, 2016.

[37] C.-C. Chao, R. McEliece, L. Swanson, and E. Rodemich, "Performance of binary block codes at low signal-to-noise ratios," *IEEE Transactions on Information Theory*, vol. 38, no. 6, pp. 1677–1687, 1992.

[38] J. G. D. Forney and G. Ungerboeck, "Modulation and coding for linear gaussian channels," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2384–2415, 1998.

[39] V. D. Goppa, "A new class of linear correcting codes," *Probl. Peredachi Inf.*, vol. 6, no. 3, pp. 24–30, 1970.

[40] V. D. Goppa, "Codes on algebraic curves," *Sov. Math., Dokl.*, vol. 24, pp. 170–172, 1981.

[41] A. S. E. Abbe and A. Wigderson, "Reed-muller codes for random erasures and errors," *IEEE Transactions on Information Theory*, vol. 61, no. 10, pp. 5229–5252, 2015.

[42] R. Saptharishi, A. Shpilka, and B. L. Volk, "Efficiently decoding reed-muller codes from random errors," in *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 227–235, 2016.

[43] V. Sidel'nikov and A. Pershakov, "Decoding of reed-muller codes with a large number of errors," *Problems of Information Transmission*, vol. 28, no. 3, pp. 80–94, 92.

[44] P. Loidreau and B. Sakkour, "Modified version of sidelnikov-pershakov decoding algorithm for binary second order reed-muller codes," *International Workshop on Algebraic and Combinatorial Coding theory*, pp. 266–271, 2004.

[45] M. Ye and E. Abbe, "Recursive projection-aggregation decoding of reed-muller codes," *IEEE Transactions on Information Theory*, vol. 66, no. 8, pp. 4948–4965, 2020.

[46] I. Dumer and K. Shabunov, "Near-optimum decoding for subcodes of reed-muller codes," in *Proceedings. 2001 IEEE International Symposium on Information Theory (IEEE Cat. No. 01CH37252)*, p. 329, IEEE, 2001.

[47] I. Dumer, "Polar codes with a stepped boundary," in *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 2613–2617, 2017.

[48] V. Guruswami and P. Xia, "Polar codes: Speed of polarization and polynomial gap to capacity," *IEEE Transactions on Information Theory*, vol. 61, no. 1, pp. 3–16, 2015.

[49] M. Fereydounian, M. V. Jamali, H. Hassani, and H. Mahdavifar, "Channel coding at low capacity," in *2019 IEEE Information Theory Workshop (ITW)*, pp. 1–5, 2019.

[50] M. Fereydounian, H. Hassani, M. V. Jamali, and H. Mahdavifar, "Channel coding at low capacity," *arXiv preprint arXiv:1811.04322*, 2018.

[51] I. Dumer and N. Gharavi, "Codes for high-noise memoryless channels," in *2020 International Symposium on Information Theory and Its Applications (ISITA)*, pp. 101–105, 2020.

[52] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE transactions on information theory*, vol. 47, no. 2, pp. 619–637, 2001.

[53] J. Hadamard, "Résolution d'une question relative aux determinants," *Bulletin des Sciences Mathematiques*, vol. 17, pp. 240–246, 1893.

[54] R. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. 27, no. 5, pp. 533–547, 1981.

[55] N. Wiberg, H.-A. Loeliger, and R. Kotter, "Codes and iterative decoding on general graphs," in *Proceedings of 1995 IEEE International Symposium on Information Theory*, pp. 468–, 1995.

[56] J. H. Kim and J. Pearl, "A computational model for causal and diagnostic reasoning in inference systems," in *Proceedings of the Eighth International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'83, (San Francisco, CA, USA), p. 190–193, Morgan Kaufmann Publishers Inc., 1983.

[57] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1988.

[58] T. Richardson and R. Urbanke, *Modern Coding Theory*. USA: Cambridge University Press, 2008.

[59] A. Dvoretzky, "Asymptotic normality for sums of dependent random variables," in *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Volume 2: Probability Theory*, pp. 513–535, University of California Press, 1972.

[60] I. A. Ibragimov, "A note on the central limit theorems for dependent random variables," *Theory of Probability & Its Applications*, vol. 20, no. 1, pp. 135–141, 1975.

[61] M. Rosenblatt, "A central limit theorem and a strong mixing condition," *Proceedings of the National Academy of Sciences*, vol. 42, no. 1, pp. 43–47, 1956.

[62] W. Hoeffding and H. Robbins, "The central limit theorem for dependent random variables," *Duke Mathematical Journal*, vol. 15, no. 3, pp. 773 – 780, 1948.

[63] H. Bergström, "A comparison method for distribution functions of sums of independent and dependent random variables," *Theory of Probability & Its Applications*, vol. 15, no. 3, pp. 430–457, 1970.

[64] P. H. Diananda, "The central limit theorem for m-dependent variables," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, no. 1, p. 92–95, 1955.

[65] K. N. Berk, "A Central Limit Theorem for m-Dependent Random Variables with Unbounded m," *The Annals of Probability*, vol. 1, no. 2, pp. 352 – 354, 1973.

[66] A. DasGupta, *Central Limit Theorems for Dependent Sequences*, pp. 119–129. New York, NY: Springer New York, 2008.

[67] P. Trifonov, "Efficient design and decoding of polar codes," *IEEE Transactions on Communications*, vol. 60, no. 11, pp. 3221–3227, 2012.

[68] I. Tal and A. Vardy, "How to construct polar codes," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6562–6582, 2013.

[69] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience, 2006.

[70] S. J. Johnson, *Iterative Error Correction: Turbo, Low-Density Parity-Check and Repeat-Accumulate Codes*. Cambridge: Cambridge University Press, 2009.

[71] J. Garcia-Frias and W. Zhong, "Approaching shannon performance by iterative decoding of linear codes with low-density generator matrix," *IEEE Communications Letters*, vol. 7, no. 6, pp. 266–268, 2003.

[72] J. C.-J. Pang, H. Mahdavifar, and S. S. Pradhan, "Capacity-achieving polar-based ldgm codes with crowdsourcing applications," in *2020 IEEE International Symposium on Information Theory (ISIT)*, pp. 449–454, IEEE, 2020.

[73] P. Suthisopapan, M. Kupimai, R. Tongta, and V. Imtawil, "Design of high-rate ldgm codes," in *2009 Fourth International Conference on Communications and Networking in China*, pp. 1–4, IEEE, 2009.

[74] N. Sourlas, "Spin-glass models as error-correcting codes," *Nature*, vol. 339, no. 6227, pp. 693–695, 1989.

[75] H. Zhang, R. Li, J. Wang, S. Dai, G. Zhang, Y. Chen, H. Luo, and J. Wang, "Parity-check polar coding for 5g and beyond," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–7, IEEE, 2018.

[76] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2213–2226, 2005.

[77] K. Niu and K. Chen, "Crc-aided decoding of polar codes," *IEEE Communications Letters*, vol. 16, no. 10, pp. 1668–1671, 2012.

[78] I. Dumer and K. Shabunov, "Soft decision decoding of reed-muller codes: recursive lists," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1260–1266, 2006.

[79] I. Dumer and N. Gharavi, "Combined polar-ldpc design for channels with high noise," in *2021 IEEE Information Theory Workshop (ITW)*, pp. 1–6, 2021.

[80] E. Arikan and E. Telatar, "On the rate of channel polarization," in *2009 IEEE International Symposium on Information Theory*, pp. 1493–1495, IEEE, 2009.

[81] I. Dumer and N. Gharavi, "Codes approaching the shannon limit with polynomial complexity per information bit," *arXiv preprint arXiv:2101.10145*, 2021.

[82] A. Shokrollahi, "Raptor codes," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2551–2567, 2006.

[83] H. Pfister, "Private communication," July 2021.