Polar Codes with Near-Optimal Convergence to Channel Capacity

Andrii Riazanov

CMU-CS-22-102

May 2022

Computer Science Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Venkatesan Guruswami, Chair
Pravesh Kothari
Ryan O'Donnell
Alexander Barg, University of Maryland

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

This research was sponsored by the National Science Foundation under awards CCF-1422045, CCF-1563742, CCF-1814603, and CCF-1908125. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government or any other entity.

Keywords: Coding Theory, Capacity-achieving Codes, Polar Codes, Scaling Exponent

Abstract

Reliable transmission of data is a central topic in coding theory and information theory. Both of these fields were founded by Claude E. Shannon in his seminal work, where he formalized the problems of communicating information and established their limits. It has been a major problem since then to find explicit coding schemes that achieve these limits.

For channel coding this corresponds to finding codes that achieve channel (Shannon) capacity. Channel polarization is a novel approach to code construction introduced by Arıkan, which he used to construct polar codes that provably achieve capacity for any memoryless symmetric channel and have low encoding and decoding complexities.

The focus of this thesis is on constructing a variant of polar codes with an almost optimal speed of convergence to capacity. Let W be a binary-input memoryless symmetric (BMS) channel with Shannon capacity $I(W)$. Shannon's noisy coding theorem established the *existence* of capacity-achieving codes (without efficient construction or decoding) which have rate $R = I(W) - \delta$ and blocklength $N = O(1/\delta^2)$. This quadratic scaling of blocklength N on the gap δ to capacity is known to be the best possible.

We construct, for any sufficiently small $\delta > 0$, a variant of polar codes with rate $R = I(W) - \delta$ and almost-optimal block length $N = O(1/\delta^{2+\alpha})$, which enables reliable communication on W with quasi-linear time encoding and decoding. This result thus yields a constructive version of Shannon's theorem with near-optimal convergence to capacity as a function of the block length, which resolves a central theoretical challenge associated with the attainment of Shannon capacity.

The codes constructed in this dissertation are a variant of Arıkan's polar codes based on multiple carefully constructed local kernels, one for each intermediate channel that arises in the decoding. A crucial ingredient in the analysis is a strong converse of the noisy coding theorem when communicating using random linear codes on arbitrary BMS channels. Our converse theorem shows extreme unpredictability of even a single message bit for random coding at rates slightly above capacity.

Acknowledgments

First and foremost, I would like to thank my parents who set me on the path for learning and development from my early life. Their desire to provide me and my brother with the best education we could get (and handle) is what ultimately made my Ph.D. study possible, and I am grateful for their unconditional support and trust in me.

I have been incredibly fortunate to have Venkat Guruswami as my advisor during the last five years. He has been extremely flexible and supportive with the directions where I was taking my research, and was an incredible source of ideas and questions. I was lucky to use Venkat's broad expertise and insight to work on a variety of problems, one of which eventually led to writing this dissertation. I am very thankful for his support and guidance during my program.

Many results in this thesis were obtained in collaboration with Min Ye. He has been great to work with and I have learned a lot from him during his visit to CMU, which I greatly enjoyed.

I would like to thank everyone involved in creating and running the Algorithms, Combinatorics, and Optimization (ACO) program which I was fortunate to be a part of. I find the exposure to a variety of topics in computer science and math that this program gives tremendously useful and exciting, and definitely worth taking additional qualifying exams (of which I was trying to convince all interested incoming students ever since I started the program).

I thank the committee members: Venkat, Ryan O'Donnell, Pravesh Kothari, and Alexander Barg, for taking their time and effort to give their valuable insight and feedback on this thesis.

I would like to thank Deb Cavlovich and all the staff at CSD for making the administrative aspects of my Ph.D. life a smooth sailing.

I am also grateful for the fun times and memorable experiences I had with my peers during my stay at CMU. This includes, but is not limited to: climbing with Vijay Bhattiprolu (special thanks for introducing me to climbing in general), Michael Rudow, Roie Levin, Arjun Teh, and many more; playing volleyball with Alex Wang and Sasha Rudenko; biking with Sai Sandeep; skiing/snowboarding with Paul Gölz, Sasha, and Alex; and doing coding competitions to distract ourselves from research with Sasha, Paul, and Da Qi Chen.

I would like to express my gratitude to my undergraduate mentors who have greatly influenced my path as a researcher. Michael Vyalyi inspired me through his lectures and introduced me to theoretical computer science (which

I stubbornly referred to as "math" for several years), and was a great source of knowledge during our collaboration. Yury Maximov was tremendously helpful and supportive during the hardest times, and I was very lucky to be a part of his research group.

Finally, I am deeply grateful to my wife Svetlana for her endless support and love, and to my son Max for being a source of pure joy, inspiration, and motivation. I have been fortunate to have my family by my side all these years and was very lucky to have a flexible schedule of a graduate student which allowed me to be by Max's side as he developed from his very birth. I thank Svetlana for believing in me from the very beginning, and for always being there for me.

Contents

Chapter 1

Introduction

The problem of transmission of information through a noisy channel lies at the heart of any communication system. This problem is a central topic of information theory and coding theory, where the goal is to find coding schemes which allow efficient, reliable, and fast communication.

To reliably transmit a message through a noisy communication channel, the common approach is to add some redundancy to the message. If the redundancy is introduced in a clever way, the receiving side might be able to recover the initial message, even though it was perturbed by noise during the communication. The task is then to try to add as little additional information as possible, while still keeping the communication reliable.

This problem was first formalized and studied by Shannon in his seminal work [Sha48]. For any probabilistic communication channel $W : \mathcal{X} \to \mathcal{Y}$ which takes an input from alphabet $\mathcal{X}$ and outputs a symbol from alphabet $\mathcal{Y}$, he considered the maximal rate at which information can be sent through W in a reliable way. Shannon showed how to compute this maximal rate, called the capacity of the channel $I(W)$, and proved the existence of codes with rates arbitrarily close to $I(W)$. However, this result only proves the existence of such codes and leaves open a challenge of finding such good codes constructively.

In more detail, the performance of a code consists of its rate R, block error probability P_e, and encoding and decoding algorithms complexities. While Shannon's result proves the existence of codes for which $R \to I(W)$ and $P_e \to 0$ simultaneously, his noisy coding theorem is based on the probabilistic method and does not describe any explicit construction for codes that approach capacity, and does not have a way to decode from errors efficiently. It has since been a major challenge to find capacity-achieving codes for which there exist explicit polynomial-time construction procedures and efficient encoding and decoding algorithms.

Polar codes, introduced by Arıkan in his breakthrough paper [Arı09], formed the first family of codes that *provably* resolved this challenge, achieving capacity for any binary-input memoryless symmetric channel while having low encoding and decoding complexities. Prior to this discovery, the capacity-achieving property was not proven even for bestperforming codes used in practical applications (except for the case of the binary erasure channel).

The next challenge then is finding codes that not only achieve capacity, but also do it (almost) optimally fast. This means that one wants the rate R to approach channel capacity $I(W)$ as fast as possible, in terms of the increasing blocklength N. At the same time, one might also aim for the best possible convergence of the decoding error probability P_e to 0. Finally, we desire codes with efficient algorithms.

The ultimate goal in this thesis then is to construct the codes for which:

(1) $R \to I(W)$, and the convergence is almost optimally fast;

(2) $P_e \to 0$, and the convergence is almost optimally fast;

(3) construction, encoding and decoding algorithms are fast (polynomial in N).

The main contribution of this dissertation is constructing a variant of polar codes which resolves this challenge, for any binary-input memoryless symmetric channel. We now explain in more detail what "fast" means for the convergences above.

1.1 Scaling exponent

Consider the binary symmetric channel (BSC), which is one of the most fundamental and well-studied noise models in coding theory. The BSC with crossover probability $p \in (0, 1/2)$ (BSC_p) flips each transmitted bit independently with probability p. By Shannon's noisy coding theorem [Sha48], we know that the capacity of BSC_p is $I(W) = 1 - h(p)$, where $h(\cdot)$ is the binary entropy function. More precisely, the theorem showed that for any $\delta > 0$, there *exist* codes of rate $I(W) - \delta$ using which one can achieve miscommunication probability at most $2^{-\Omega(\delta^2 N)}$, where N is the blocklength of the code. In fact, random linear codes under the maximum likelihood decoding offer this guarantee with high probability. Thus Shannon's theorem implies the existence of codes of blocklength $N = O(1/\delta^2)$ that can achieve small error probability on BSC_p at rates within δ of capacity. Conversely, by several classical results [Wol57, Str62, Str09, PPV10], it is known that the blocklength N has to be at least $\Omega(1/\delta^2)$ in order to approach capacity within δ. We refer to δ, which is equal to $I(W) - R$, as the *gap to capacity.*

If the blocklength N of the code scales as $O(1/\delta^\mu)$ as a function of the gap δ to capacity, we say that μ is the *scaling exponent.* This is equivalent to saying that the gap to capacity δ scales as $O(N^{-1/\mu})$, and so the scaling exponent captures the convergence of the code rate to capacity, as the blocklength increases. The previous paragraph shows that the fastest convergence to capacity corresponds to the minimal value of the scaling exponent $\mu = 2$, and is achieved by random linear codes.

As this result was only proven for random codes, it is not constructive. The theoretical challenge of constructing codes of rate $1 - h(p) - \delta$ with blocklength N and construction/decoding complexity scaling polynomially in $1/\delta$ remained wide open for a long time. This is not surprising, as we already know that even the problem of designing codes that achieve capacity in general is highly non-trivial, and only a few families of codes provably achieve this. Arıkan's original analysis [Arı09] established the convergence to capacity for polar codes as the blocklength grows to infinity, but did not quantify the speed of this convergence. Later, around 2013, two independent works [GX15, HAU14] gave an effective finite-length analysis of polar codes, deriving codes with the blocklength, construction, and decoding complexity all bounded by a polynomial in $1/\delta$, for any binary-input memoryless symmetric channel. These results established that the scaling exponent μ of polar codes is finite. The polar codes are up to this day the *only* known efficiently decodable capacity-achieving family of codes proven to have a finite scaling exponent. The work [GX15] did not give an explicit upper bound on the scaling exponent of polar codes, whereas [HAU14] showed the bound $\mu \leq 6$. The upper bound was then improved to $\mu \leq 4.714$ for a general BMS channel in subsequent works [GB14, MHU16]. On the other hand, for original Arıkan's polar codes it was shown in [HAU14] that the scaling exponent is *lower bounded* by $\mu \geq 3.579$.

This is quite far from the optimal speed of convergence with $N = O(1/\delta^2)$ achieved by random codes. But these recent results raise the intriguing challenge of constructing codes with the scaling exponent close to 2, a goal we could not even dream of till the recent successes of polar codes. The main contribution of this dissertation is closing this gap, and building a variant of polar codes with an almost optimal convergence to channel capacity. Specifically, for arbitrarily small positive α, we obtain a scaling of $N = O(1/\delta^{2+\alpha})$, or, in other words, get the scaling exponent $\mu = 2+\alpha$. As we are interested in constructive results, our codes have polynomial-time construction and efficient (quasi-linear time) encoding and decoding algorithms.

1.2 Decoding error probability

In addition to looking at the speed of convergence to capacity, we are also interested in how fast the block error probability P_e tends to 0. Several different regimes can be considered in terms of scalings of the rate R and the decoding error probability P_e as N increases:

- In the *error exponent regime*, the rate $R < I(W)$ is fixed, and the scaling of P_e as a function of N is studied.
- In the *scaling exponent regime*, the error probability P_e is fixed, and the scaling of $\delta = I(W) - R$ is considered.
- Finally, in the *moderate deviations regime*, neither of the parameters is fixed, and instead the joint scaling of $I(W) - R$ and P_e is studied.

For the standard polar codes in the *error exponent regime*, Arıkan and Telatar [AT09] first proved that the decoding error probability scales with blocklength N as $\exp(-\sqrt{N})$. This behaviour was later extended in [KSU10] for a more general version of polar codes which use larger $\ell \times \ell$ kernels G, where the decoding error was estimated as $P_E \approx \exp(-N^{E_c(G)})$, where $E_c(G) < 1$ is a constant which depends on the matrix G.

The *scaling exponent regime* is what we already briefly discussed in Section 1.1.

For the *moderate deviations regime*, several works [AW10, AW14, PPV10] studied the joint behaviour of R and P_e for general codes, and proved the fundamental tradeoff between these parameters. Namely, they proved $\dfrac{-\ln P_e}{N(I(W)-R)^2} \to \dfrac{1}{2V}$, where V is a parameter of a channel called channel dispersion ($V > 0$ for non-trivial binary-input channels so we always assume this). It then follows that for the code with rate $R \geq I(W) - N^{-1/\mu}$ (i.e. scaling exponent μ) and decoding error probability $P_e \leq \exp(-N^{\varphi})$, these scalings must satisfy the tradeoff $\frac{2}{\mu} + \varphi \leq 1$.

As our primary focus is getting the almost-optimal scaling exponent $\mu = 2 + \alpha$, we see that the best scaling of the block error probability we can hope to achieve is $P_e \leq \exp(-N^{\Omega(\alpha)})$. This can be viewed as achieving the optimal tradeoff curve $\frac{2}{\mu} + \varphi \leq 1$ from one side, near the point $(\mu = 2, \varphi = 0)$. This is exactly the scaling we prove for our codes by applying the framework from [WD18a] for studying the tradeoff between μ and φ.

1.3 Summary of contributions

The main contribution of this thesis work consists in resolving the challenges we described for an arbitrary binary-input memoryless symmetric channel. Specifically, we present a variant of polar codes with a near-optimal convergence to capacity, almost optimal scaling of the decoding error probability, and which also can be constructed in polynomial time and have quasi-linear encoding and decoding algorithms. This “constructivizes” the quantitative finite-length version of Shannon’s theorem with a small slack in the speed of convergence to capacity. Below we outline several ideas and technical contributions that all lead to this result.

1.3.1 Mixed-kernel construction

Arıkan’s original polar coding construction is based on a large tensor power of a simple 2×2 matrix, which is called the *kernel* of the code construction. For this construction, it was shown in [HAU14] that the scaling exponent μ for Arıkan’s original polar code construction is *lower bounded* by 3.579 (when successive cancellation decoding is used). Given this limitation, one approach to improve μ is to consider polar codes based on larger $\ell \times \ell$ kernels for some $\ell > 2$. This approach did in fact lead to improvements in scaling exponents, but only for the special case of erasure channels. For the BEC, using large kernels, polar codes with the scaling exponent $(2 + \alpha)$ for any desired $\alpha > 0$ were given in [FHMV17], where it was shown that a random kernel of large enough size works. However, no improvement was proven for the general BMS channel using large matrices.

One of the crucial ideas that lead to our code construction is the use of mixed kernels in the construction of polar codes, where the kernel matrix is not fixed throughout the recursive construction. This can be viewed as the next natural extension of kernel usage, after increasing the kernel size from 2 to arbitrary ℓ. Although this approach is not new and it was shown that using different internal kernels can also lead to polarization

([PSL15, YB15, GBLB17, BBGL17, WD18a]), no gains in parameter scalings were previously analytically proven. We show how to utilize the mixed-kernel (also referred to as a *dynamic kernel*) approach, and describe a way to choose the kernels during the construction procedure which results in obtaining the almost-optimal scaling exponent, while also keeping the other desired properties of the code.

1.3.2 Strong converse for bit-decoding

One of the main technical results that made our code construction possible is formulated in a context not related to polar codes, and concerns bit-decoding of random linear codes. The strong converse of Shannon's theorem implies that if a code has rate greater than the capacity of the channel, then the probability of error in decoding *the entire message* goes to 1. More specifically, Wolfowitz's classic result [Wol57] implies that if $R \geq I(W) + O(1/\sqrt{N})$, then P_e is close to 1, which means that decoding the whole input message cannot be done reliably using this code.

In our version of the strong converse we show that for *random linear codes* with rates slightly above capacity, decoding even one bit of the message is not possible reliably. Specifically, if $R \geq I(W) + O(\log^3(N)/\sqrt{N})$, then even when one only needs to recover *a single bit* of the input message X given the output Y, this is still not possible to do even slightly better than a coin flip (for random linear codes). The specific very strong lower bound on the entropy $H(X_1 \,|\, Y_1^N)$ is what makes this statement technically challenging. As this contribution can be formulated independently of the polar codes context, it is presented separately in Chapter 3 before we talk about polarization.

1.3.3 Inverse sub-exponential decoding error probability

Decoding error probability scaling as $P_e \leq \exp(N^{-\varphi})$ for some φ was previously proven for polar codes ([AT09, KSU10, HMTU13, MT14, GX15]) and the moderate deviations regime with both finite scaling exponent and good scaling of the decoding error probability was proven for general BMS channels in [MHU16] and further improved in [WD18a, WD18b]. However, the challenge of getting the *near-optimal* scaling exponent and sub-exponentially small decoding error probability simultaneously was open for some time. The challenge was resolved in [WD19, WD21], where the authors prove that for any pair of parameters (μ, φ) satisfying $\varphi + \frac{2}{\mu} < 1$, codes with $R \geq I(W) - N^{-1/\mu}$ and $P_e \leq \exp(N^{-\varphi})$ are achievable, however their codes are lacking polynomial time construction as presented.

In [GRY20] we first showed how to obtain the almost-optimal scaling exponent but only with inverse-polynomial error probability $P_e \leq N^{-\Omega(1)}$. After that in [GRY22] we show that the analysis of polarization from [WD19] can be applied to our codes to improve the decoding error probability to be sub-exponentially small, while keeping the near-optimal scaling exponent and polynomial-time construction. In this thesis, we further show that the framework from [WD19] can also be used to achieve any pair of parameters (φ, μ) satisfying a certain tradeoff (not optimal), also without sacrificing poly(N) construction, for any BMS channel.

1.4 Organization

In Chapter 2 we briefly establish the background for channel coding and go over the notations, definitions, and several useful technical facts.

In Chapter 3 we prove the strong converse theorem for single-bit decoding for random linear codes, a self-contained result outside of the context of polar codes.

Chapter 4 gives an overview of polar codes and covers the state of the art.

In Chapter 5 we show for any binary-input symmetric channel how to construct polar codes with the near-optimal scaling exponent $\mu = 2 + \alpha$ for any small α, but with only inverse-polynomial decoding error probability P_e. This is the result from [GRY20].

Finally, Chapter 6 improves the codes to have sub-exponentially small decoding error probability, the result from [GRY22] obtained using the tools from [WD18a, WD19].

Chapter 2

Preliminaries

2.1 Notations

We use bold notations to denote vectors, while keeping the notations for the coordinates regular font, for instance $\boldsymbol{U} \in \mathcal{U}^n$ corresponds to $\boldsymbol{U} = (U_1, U_2, \dots, U_n)$. We denote a slice of a vector as $\boldsymbol{U}_i^j = (U_i, U_{i+1}, \dots, U_j)$, and in some places $\boldsymbol{U}_{<i} = (U_1, \dots, U_{i-1})$ is used.

Calligraphic capital letters $(\mathcal{X}, \mathcal{Y})$ generally correspond to sets/alphabets.

Uppercase letters X, Y, U, and V, and their bold vector notations generally denote random variables (inputs and outputs of coding channels), while their lowercase versions usually denote realizations of these random variables.

2.2 Coding channel and its parameters

In this thesis we only work with channel coding, so we first define the channel that describes the communication.

Definition 2.1. *A discrete memoryless channel $W : \mathcal{X} \to \mathcal{Y}$ is described by a finite input alphabet $\mathcal{X}$, a finite output alphabet $\mathcal{Y}$, and a conditional probability distribution $W(y \,|\, x)$, such that if the input $\mathbf{X} \in \mathcal{X}^n$ is fed to n copies of the channel, the output $\mathbf{Y} \in \mathcal{Y}^n$ satisfies $\mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \,|\, \boldsymbol{X} = \mathbf{x}] = \prod_{i=1}^n W(y_i \,|\, x_i)$.*

We will mostly consider binary-input channels in this thesis, for which $\mathcal{X} = \mathbb{F}_2$ is a binary field. Moreover, our focus will only be on symmetric channels:

Definition 2.2. *Binary-input discrete memoryless channel is said to be **symmetric** is there exists an permutation $\sigma : \mathcal{Y} \to \mathcal{Y}$ on the output alphabet such that $\sigma^{-1} = \sigma$ (involution) and $W(y \,|\, 0) = W(\sigma(y) \,|\, 1)$ for all $y \in \mathcal{Y}$.*

We abbreviate a binary-input discrete memoryless symmetric channel as a *BMS channel*, and we only consider such channels in this thesis. So assume $\mathcal{X} = \mathbb{F}_2$ unless stated otherwise.

The simplest example of a BMS channel is called a *binary symmetric channel (BSC)* for the case $\mathcal{Y} = \mathbb{F}_2$. The BSC_p is described by a crossover (flip) probability $p \in [0, 1]$, such that $W(1 \,|\, 0) = W(0 \,|\, 1) = p$ and $W(0 \,|\, 0) = W(1 \,|\, 1) = 1 - p$.

Another basic example of a BMS channel is a *binary erasure channel* BEC, where $\mathcal{Y} = \{0, 1, ?\}$. Each input is either erased with some probability ε, or remains unchanged with probability $(1 - \varepsilon)$.

Denote by Q the uniform distribution over the input alphabet $\mathbb{F}_2$, and treat it as a distribution of the input to the channel W. Then by $W(x, y)$ we denote the joint distribution $W(x, y) = Q(x) \cdot W(y \,|\, x)$ of input X and output Y. Further, we abuse the notation and use $W(x \,|\, y)$ as a posterior distribution of the input x given the output of the channel y. It will always be clear from the context and difference in notations if we are considering a transmission probability or a posterior probability.

2.3 Channel parameters

All logarithms in this document are to the base 2, unless explicitly specified.

Definition 2.3. *The entropy $H(W)$ of a BMS channel W is the conditional entropy $H(X \,|\, Y)$, where $X \sim Q$ is an input to W and Y is output of W:*

$$H(W) = H(X \,|\, Y) = -\sum_{x \in \mathbb{F}_2} \sum_{y \in \mathcal{Y}} W(x, y) \log_2 W(x \,|\, y).$$

The entropy of a channel corresponds to the level of uncertainty about the input X if the output of the channel is known, and represents the amount of noise the channel W introduces into communication.

For example, the entropy of $\mathrm{BSC}(p)$ is $H(\mathrm{BSC}(p)\ ) = -p \log p - (1-p) \log(1-p) = h(p)$ for $p \in [0, 1]$, where $h(p)$ is called the *binary entropy function.* Notice how the entropy of $\mathrm{BSC}(p)$ is 0 for $p = 0$ or $p = 1$, as the channel is deterministic for these cases and does not introduce any noise.

Definition 2.4. *For a BMS channel W, the* channel capacity *$I(W)$ is equal to the* mutual information *$I(X : Y)$ of the input X and the output Y of the channel:*

$$I(W) = H(Q) - H(W) = 1 - H(W).$$

Definition 2.5. *The Bhattacharyya parameter of a BMS channel W is*

$$Z(W) = \sum_{y \in \mathcal{Y}} \sqrt{W(y \,|\, 0) W(y \,|\, 1)}.$$

The Bhattacharyya parameter $Z(W)$ also measures the amount of noise W introduced, since $H(W)$ is close to 0 if and only if $Z(W)$ is close to 0, and the same with the other end of the interval $[0, 1]$. This can be formalized in the following inequalities:

Proposition 2.6 ([Arı09, Kor09])**.** *For any BMS channel W,*

$$\begin{aligned} I(W) + Z(W) &\geq 1, \\ I(W)^2 + Z(W)^2 &\leq 1. \end{aligned}$$

2.4 Codes

The communication protocol in channel coding looks as follows. We are trying to send a message of k bits, $\boldsymbol{U} = (U_1, U_2, \dots U_k) \in \mathbb{F}_2^k$. The *encoder* function $Enc : \mathbb{F}_2^k \to \mathbb{F}_2^N$ introduces the redundancy into the message, producing a *codeword* $\boldsymbol{X} = Enc(U_1^k) \in \mathbb{F}_2^N$. The message is then sent through N copies of the communication BMS channel W, which produce an output vector $\boldsymbol{Y} \in \mathcal{Y}^N$. As the channel is memoryless, this means that each coordinate X_i is transferred independently to obtain Y_i according to transition probabilities $W(y \,|\, x)$ of the channel. Our task is to try and recover the initial message $\boldsymbol{U}$, so at the last stage the decoding function $Dec : \mathcal{Y}^N \to \mathbb{F}_2^k$ takes the output vector $\boldsymbol{Y}$ and produces a guess $\hat{\boldsymbol{U}}$ for the initial message.

$$\boldsymbol{U} \in \mathbb{F}_2^k \xrightarrow{\quad Enc(\cdot) \quad} \boldsymbol{X} \in \mathbb{F}_2^N \longrightarrow \boxed{W^N} \longrightarrow \boldsymbol{Y} \in \mathcal{Y}^N \xrightarrow{\quad Dec(\cdot) \quad} \hat{\boldsymbol{U}} \in \mathbb{F}_2^k$$

The set of codewords $\boldsymbol{X}$ that can be sent through the channel, i.e. the range of $Enc(\cdot)$, is called the *code* C, which is a subset of $\mathbb{F}_2^N$.

The length of a codeword N is called a *blocklength*, and we call the *rate* of the code $R = \frac{k}{N}$. This is an effective rate at which the communication is happening, meaning that we are trying to send R bits of information per one usage of the channel W. The channel capacity $I(W)$ determines the maximal possible rate, for a given BMS channel W, for which such communication can be reliable. The reliability of the communication is described by the probability of decoding the message incorrectly, $P_e = \mathbb{P}[\hat{\boldsymbol{U}} \neq \boldsymbol{U}]$, where again $\hat{\boldsymbol{U}} = Dec(W^N(Enc(\boldsymbol{U})))$. We call P_e to be the decoding error probability or the block error probability.

Linear codes

A binary code C is a *linear code* if C is a linear subspace of $\mathbb{F}_2^N$. A binary linear code C with blocklength N and rate R (so the message length is $k = RN$) can always be described by a generator matrix $G \in \mathbb{F}_2^{k \times N}$: $C = \{\mathbf{x}G \,:\, \mathbf{x} \in \mathbb{F}_2^k\}$. For binary linear codes that are described by a generator matrix the encoding procedure $Enc(\cdot)$ is straightforward – this is just a matrix multiplication, and can always be done in $O(nk)$ time. Polar codes, as we will see, are in fact linear codes.

2.5 Useful facts

Binary entropy function

The binary entropy function is defined as $h(x) = -x\log x - (1-x)\log(1-x)$, where $0\log 0$ is taken to be 0. We will use the following simple fact several times in the proofs:

Proposition 2.7. $h(x) \leq 2x\log\frac{1}{x}$ *for* $x \in [0, 1/2]$.

Proof. Consider the function $f(x) = 2x\log\frac{1}{x} - h(x) = x\log\frac{1}{x} - (1-x)\log\frac{1}{1-x}$ on $[0, 1/2]$. We have $f''(x) = \dfrac{2x-1}{x(1-x)\ln 2} < 0$ on $(0, 1/2)$, so f is strictly concave on this interval, and further $f(0) = f(1/2) = 0$. Therefore, $f(x)$ is positive on $(0, 1/2)$. □

The following proposition follows from the facts that $h(x)$ is concave, increasing for $x \in [0, 1/2)$, and symmetric around $1/2$, i.e. $h(x) = h(1-x)$ for $x \in [0, 1]$.

Proposition 2.8. *For any* $x, y \in [0, 1]$, $|h(x) - h(y)| \leq h(|x-y|)$.

Proof. The inequality is trivial when x or y is equal to 0 or $h(x) = h(y)$. Without loss of generality, assume $x > y$. Further, consider first the case $h(x) > h(y)$. We have two cases:

(a) $0 < y \leq (x-y) < x$. By the mean value theorem, we can write $(h(x) - h(x-y)) = h'(\xi_1)y$ for some $\xi_1 \in (x-y, x)$, and $h(y) = h(y) - h(0) = h'(\xi_2)y$ for some $\xi_2 \in (0, y)$. Then $\xi_2 \leq \xi_1$, and since h is concave, it follows that $h'(\xi_2) \geq h'(\xi_1)$, thus $h(x) - h(x-y) \leq h(y)$. Rearranging, obtain the desired inequality.

(b) $0 < (x-y) \leq y < x$. By the same argument, one has $(h(x) - h(y)) = h'(\xi_1)(x-y)$ for $\xi_1 \in (y, x)$ and $h(x-y) = h(x-y) - h(0) = h'(\xi_2)(x-y)$ for some $\xi_2 \in (0, x-y)$, and so $\xi_2 \leq \xi_1$, therefore $h'(\xi_2) \geq h'(\xi_1)$ by concavity. Thus $h(x) - h(y) \leq h(x-y)$.

Next, if $h(x) < h(y)$, define $x' = 1-y$ and $y' = 1-x$. It follows that $x' > y'$ and $h(x') = h(y) > h(x) = h(y')$, so the inequality in the proposition holds for x' and y' by the cases (a)-(b) above. But clearly $|h(x) - h(y)| = |h(x') - h(y')| \leq h(|x'-y'|) = h(|x-y|)$ by symmetry of h around $\frac{1}{2}$. □

Channel degradation

Definition 2.9. *Let* $W : \{0,1\} \to \mathcal{Y}$ *and* $\widetilde{W} : \{0,1\} \to \widetilde{\mathcal{Y}}$ *be two BMS channels. We say that* $\widetilde{W}$ *is* degraded *with respect to* W*, or, correspondingly,* W *is* upgraded *with respect to* $\widetilde{W}$*, denoted as* $\widetilde{W} \preceq W$*, if there exists a discrete memoryless channel* $W_1 : \mathcal{Y} \to \widetilde{\mathcal{Y}}$ *such that*

$$\widetilde{W}(\widetilde{y}\,|\,x) = \sum_{y\in\mathcal{Y}} W(y\,|\,x)W_1(\widetilde{y}\,|\,y) \qquad \forall\ x \in \{0,1\},\ \widetilde{y} \in \widetilde{\mathcal{Y}}.$$

This is equivalent to saying that $\widetilde{W}(x)$ and $W_1(W(x))$ are identically distributed for any $x \in \{0,1\}$. In other words, one can simulate the usage of $\widetilde{W}$ by first using the channel

W and then applying some other channel W_1 to the output of W. We utilize the following fact from [TV13, Lemma 3]:

Proposition 2.10. *Let W and $\widetilde{W}$ be two BMS channels, such that $\widetilde{W} \preceq W$. Then $H(\widetilde{W}) \geq H(W)$ and $Z(\widetilde{W}) \geq Z(W)$.*

Chapter 3

Strong Converse for Bit-Decoding

In this chapter we describe and prove the strong converse theorem for bit-decoding for random linear codes. This is one of the main ingredients used in our construction for polar codes (and probably the most technically challenging one).

We look at the Shannon noisy-channel coding theorem, specifically at its converse, which sets the upper bound for the achievable rates of communication through the channel. The original theorem by Shannon [Sha48] showed that if the rate R of a family of codes is larger than the capacity $C(W)$ of the channel W, then, as the blocklength N increases, the block error probability is bounded from 0.

An improvement on this result by Wolfowitz [Wol57] is called a *strong converse* for Shannon's theorem, it shows that the channel capacity is a sharp threshold for reliable communication. Specifically, it proves that if the rate for the family of codes scales as $R \geq I(W) + O(1/\sqrt{N})$ as N increases, then the block error probability tends to 1. In other words, the communication is very unreliable even if the rate of the code is just slightly above capacity.

For the reasons that will become apparent in the later chapters of this thesis, here we will be considering the strong converse for *bit-decoding*. That is, while we have the same communication protocol for channel coding as before, we are only interested in decoding a single bit of the initial message, given the output vector. Without loss of generality, say we are trying to decode the first bit. We show that if one uses a random linear code with the rate slightly above the capacity of the channel, specifically $R \geq I(W) + O\left(\log^3 N/\sqrt{N}\right)$, then predicting the first bit of a message with even a tiny advantage over a uniform guess is not possible, with high probability over the randomness of the code.

Notice that if we only want to decode a specific message bit and we do not put any constraints on the code, then we can easily construct codes with rates substantially above capacity that still allow us to decode this specific message bit with high probability. All we need to do is to repeat the message bit sufficiently many times in the codeword, decode each copy based on the corresponding channel output, and then take a majority vote. The overall code rate does not even figure in this argument. Therefore, one can only hope that the converse theorem for bit-decoding holds for certain code ensembles, and for the purposes of our polar codes construction, we are interested in the random linear code ensemble. While the converse for bit-decoding in this case is surely intuitive, establishing it in the strong quantitative form that we need, and also for all BMS channels, turns out to be a challenging task.

Below we formulate the main theorem that we prove in this chapter. Notice that in this chapter the notation ℓ is used for a blocklength, instead of the usual N in all other places in this document.

Theorem 3.1. *Let W be any BMS channel, and ℓ and k be integers that satisfy $\ell \geq k \geq \ell(1 - H(W)) + 14\ell^{1/2}\log^3 \ell$, and let ℓ be large enough so that $\log \ell \geq 20$. Let G be a random binary matrix uniform over $\{0,1\}^{k\times \ell}$. Suppose a message $\boldsymbol{V} \cdot G$ is transmitted through ℓ copies of the channel W, where $\boldsymbol{V}$ is uniformly random over $\{0,1\}^k$, and let $\boldsymbol{Y}$ be the output vector, i.e. $\boldsymbol{Y} = W^{\ell}(\boldsymbol{V}\cdot G)$. Then, with probability at least $1 - \ell^{-(\log \ell)/20}$ over the choice of G it holds $H\left(V_1 \,\middle|\, \boldsymbol{Y}\right) \geq 1 - \ell^{-(\log\ell)/20}$.*

We want to point out two quantitative features of the above theorem. First, it applies at rates really close to the capacity of the channels, almost meeting the second-order scaling of the standard strong converse, up to polylog factors. Second, it rules out predicting the bit V_1 with advantage $\ell^{-\omega(1)}$ over random guessing. Both of these features are important for our code construction in the future chapters.

3.1 Proof outline

In this section we describe the plan of the proof for Theorem 3.1, but only restricted to the case of a binary symmetric channel (BSC$_p$) instead of a general BMS channel. The proof for the general BMS channel case follows the same blueprint by using the fact that a BMS channel can be represented as a convex combination of BSC subchannels, but executing it involves overcoming several additional technical hurdles. We believe that this outline will help the reader to navigate through the proofs for both BSC and BMS channel cases.

Proof plan for BSC$_p$. We prove the lower bound on $H\left(V_1 \,\middle|\, \boldsymbol{Y}\right)$ by lower bounding $\mathop{\mathbb{E}}_{g\sim G}\left[H\left(V_1 \,\middle|\, \boldsymbol{Y}\right)\right]$ and using Markov's inequality. Thus we write

$$\begin{aligned}\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right] &= \sum_g \mathbb{P}(G=g)H^{(g)}(V_1|\boldsymbol{Y}) \\ &= \sum_g \mathbb{P}(G=g)\left(\sum_{\boldsymbol{y}\in\mathcal{Y}^\ell}\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y})H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})\right),\end{aligned}$$

where the summation of g is over $\{0,1\}^{k\times\ell}$, and by $\mathbb{P}^{(g)}(\cdot)$ and $H^{(g)}(\cdot)$ we denote probability and entropy over the randomness of the message $\boldsymbol{V}$ and channel noise *for a fixed matrix g*.

1: Restrict to zero-input. The first step is to use the linearity of the (random linear) code and the additive structure of BSC to prove that we can change $\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y})$ to

$\mathbb{P}^{(g)}(\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{V} = \mathbf{0})$ in the above summation, where $\mathbf{0}$ is the all-zero vector. This observation is crucial for our arguments, since it allows us to only consider the outputs which are "typical" for the all-zero codeword, and there is no dependence on g in this case. Formally, we prove Lemma 3.5.1:

$$\mathop{\mathbb{E}}_{g \sim G}\left[H^{(g)}(V_1 | \boldsymbol{Y})\right] = \sum_{\boldsymbol{y} \in \mathcal{Y}^\ell} \mathbb{P}(\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{V} = \mathbf{0}) \cdot \mathop{\mathbb{E}}_{g \sim G}\left[H^{(g)}(V_1 | \boldsymbol{Y} = \boldsymbol{y})\right].$$

2: Define a typical set of outputs. We define a typical output set for the zero-input for BSC$_p$ as $\mathcal{F} \coloneqq \left\{\boldsymbol{y} \in \mathcal{Y}^\ell \ : \ |wt(\boldsymbol{y}) - \ell p| \leq 2\sqrt{\ell}\log\ell\right\}$. It is clear that if zero-vector is transmitted through the channel, the output will be a vector from $\mathcal{F}$ with high probability. It means that we do not lose too much in terms of accuracy if we restrict our attention only to this typical set, so the following inequality suffices as a good lower bound on the expectation.

$$\mathop{\mathbb{E}}_{g \sim G}\left[H^{(g)}(V_1 | \boldsymbol{Y})\right] \geq \sum_{\boldsymbol{y} \in \mathcal{F}} \mathbb{P}(\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{V} = \mathbf{0}) \cdot \mathop{\mathbb{E}}_{g \sim G}\left[H^{(g)}(V_1 | \boldsymbol{Y} = \boldsymbol{y})\right]. \tag{3.1}$$

3: Fix a typical output $\boldsymbol{y} \in \mathcal{F}$. For a fixed choice of an output vector $\boldsymbol{y} \in \mathcal{F}$, we can write $H^{(g)}(V_1 | \boldsymbol{Y} = \boldsymbol{y}) = h(\mathbb{P}^{(g)}(V_1 = 0 | \boldsymbol{Y} = \boldsymbol{y})) = h\left(\frac{\mathbb{P}^{(g)}(V_1 = 0, \boldsymbol{Y} = \boldsymbol{y})}{\mathbb{P}^{(g)}(\boldsymbol{Y} = \boldsymbol{y})}\right)$. It suffices to show that the ratio of these probabilities is very close to $1/2$ with high probability. In order to do this, we will show that both denominator and numerator are highly concentrated around their respective means for $g \sim G$, and that the means have a ratio of nearly $1/2$. Focusing on the denominator (the argument for the numerator is almost identical), we have:

$$2^k \cdot \mathbb{P}^{(g)}(\boldsymbol{Y} = \boldsymbol{y}) = \mathbb{P}(\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{V} = \mathbf{0}) + \sum_{d=0}^{\ell} B_g(d, \boldsymbol{y}) p^d (1-p)^{\ell - d}, \tag{3.2}$$

where $B_g(d, \boldsymbol{y})$ is defined as the number of nonzero codewords in the code spanned by the rows of g at Hamming distance d from $\boldsymbol{y}$. We proceed with proving concentration on the summation above by splitting it into two parts.

> **3a: Negligible part.** If $\boldsymbol{y}$ was received as the output of the channel, it is very unlikely that an input codeword $\boldsymbol{x}$ such that $|\text{dist}(\boldsymbol{x}, \boldsymbol{y}) - \ell p| \geq 6\sqrt{\ell}\log\ell$ was transmitted. It is then possible to show that the expectation (over $g \sim G$) of $\sum_{d \,:\, |d - \ell p| \geq 6\sqrt{\ell}\log\ell} B_g(d, \boldsymbol{y}) p^d (1-p)^{\ell - d}$ is negligible with respect to the expectation of the whole summation. Markov's inequality implies then that this sum is negligible with high probability over $g \sim G$.
>
> **3b: Substantial part.** On the other hand, for any d such that $|d - \ell p| \leq 6\sqrt{\ell}\log\ell$, the expectation of $B_g(d, \boldsymbol{y})$ is going to be extremely large for the above-capacity regime. We can apply Chebyshev's inequality to prove concentration on every single weight coefficient $B_g(d, \boldsymbol{y})$ with d in such a range. A union bound then implies that they are all concentrated around their means simultaneously.

This proves that the summation over d is concentrated around its mean in (3.2). Finally, since $|wt(\boldsymbol{y}) - \ell p| \leq 2\sqrt{\ell}\log\ell$ for $\boldsymbol{y} \in \mathcal{F}$ and we leave enough room above the capacity of the channel, w.h.p. over choice of g we have $B_g(wt(\boldsymbol{y}), \boldsymbol{y}) \gg 1$, and consequently $\mathbb{P}(\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{V} = \boldsymbol{0}) = p^{wt(\boldsymbol{y})}(1-p)^{\ell - wt(\boldsymbol{y})}$ is negligible compared to the second sum term in (3.2).

4: Concentration of entropy. Proving in the same way concentration on the numerator $\mathbb{P}^{(g)}(V_1 = 0, \boldsymbol{Y} = \boldsymbol{y})$, we derive that $\frac{\mathbb{P}^{(g)}(V_1=0,\boldsymbol{Y}=\boldsymbol{y})}{\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y})}$ is close to $\frac{1}{2}$ with high probability for any typical $\boldsymbol{y} \in \mathcal{F}$, and thus $\mathbb{E}_{g\sim G}[H^{(g)}(V_1 | \boldsymbol{Y} = \boldsymbol{y})]$ is close to 1 with high probability for such $\boldsymbol{y}$. Recalling that the probability to receive $\boldsymbol{y} \in \mathcal{F}$ is overwhelming for zero-vector input, out of (3.1) obtain the desired lower bound on $\underset{g\sim G}{\mathbb{E}}\left[H^{(g)}(V_1 | \boldsymbol{Y})\right]$.

The full proof for the BSC case is presented in Section 3.2. In order to generalize the proof to general BMS channels, we need to track and prove concentration bounds for many more parameters (in the BSC case, we had a single parameter d that was crucial). More specifically, in the BSC case we have to deal with a single binomial distribution when trying to estimate the expectation of $B_g(d, \boldsymbol{y})$. For general BMS channels, however, we have to cope with a multinomial distribution and an ensemble of binomially distributed variables that depend on the particular realization of that multinomial distribution. Moreover, we emphasize that Theorem 3.1 must hold in the non-asymptotic regime independent of the underlying channel W. (In contrast, in typical coding theorems in information theory one fixes the channel and lets the block length grow to infinity.) We show how to overcome all these technical challenges for the general BMS case in the following sections.

The rest of the chapter is organized as follows. In Section 3.2 we first fully prove Theorem 3.1 for BSC$_p$. Even though it is a partial case of the full proof, we believe it helps to understand the structure and main ideas behind the proof, and provides the roadmap for the argument for the general case. Next, in Section 3.3 we prove Theorem 3.1 for the case when the output alphabet size of a BMS channel W is bounded by $2\sqrt{\ell}$. The proof mimics the approach for the BSC case to some extent. Finally, in Section 3.4, we show how the case of a general BMS channel can be reduced to the case of the channel with a bounded alphabet via "upgraded binning" to merge output symbols.

3.2 Strong converse for BSC$_p$

Throughout this section consider the channel W to be BSC with the crossover probability $p \leq \frac{1}{2}$. Denote $H = H(W) = h(p)$, where $h(\cdot)$ is the binary entropy function. For the BSC$_p$ case we actually only require $k \geq \ell(1-H) + 8\sqrt{\ell}\log^2 \ell$ in the condition of the Theorem 3.1, and it suffices to have $\ell \geq 8$. The bound on the conditional entropy will also be stronger for this case, we prove $H\left(V_1 \;\middle|\; \boldsymbol{Y}\right) \geq 1 - \ell^{-\log \ell}$.

Proof of Theorem 3.1 *(BSC case).* We follow the plan described in Section 3.1. As we discussed there, we prove that $H(V_1 \mid \boldsymbol{Y})$ is very close to 1 with high probability over G by showing that its expectation over G is already very close to 1 and then using Markov inequality. So we want to prove a lower bound on

$$\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right] = \sum_g \mathbb{P}(G=g)H^{(g)}(V_1|\boldsymbol{Y}),$$

where $H^{(g)}(V_1|\boldsymbol{Y})$ is the conditional entropy for the fixed matrix g. Similarly, in the remainder of this section, $\mathbb{P}^{(g)}(\cdot)$ denotes probabilities of certain events *for a fixed matrix* g. By $\sum_g$ we denote the summation over all binary matrices from $\{0,1\}^{k\times\ell}$.

Restrict to zero-input. We rewrite

$$\begin{aligned}\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right] &= \sum_g \mathbb{P}(G=g)\left(\sum_{\boldsymbol{y}\in\mathcal{Y}^\ell}\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y})H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})\right)\\ &= \sum_{\boldsymbol{y}\in\mathcal{Y}^\ell}\sum_g \mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y})\cdot\mathbb{P}(G=g)H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y}).\end{aligned}$$

Our first step is to prove that in the above summation we can change $\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y})$ to $\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{V}=\boldsymbol{0})$, where $\boldsymbol{0}$ is the all-zero vector. This observation is crucial for our arguments, since it allows us to only consider the outputs $\boldsymbol{y}$ which are "typical" for the all-zero codeword when approximating $\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right]$. Precisely, we prove

Lemma 3.2. *Let W be a BMS channel, ℓ and k be integers such that $k\le\ell$. Let G be a random binary matrix uniform over $\{0,1\}^{k\times\ell}$. Suppose a message $\boldsymbol{V}\cdot G$ is transmitted through ℓ copies of W, where $\boldsymbol{V}$ is uniformly random over $\{0,1\}^k$, and let $\boldsymbol{Y}$ be the output vector $\boldsymbol{Y}=W^\ell(\boldsymbol{V}\cdot G)$. Then*

$$\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right] = \sum_{\boldsymbol{y}\in\mathcal{Y}^\ell}\sum_g \mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{V}=\boldsymbol{0})\cdot\mathbb{P}(G=g)H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y}). \tag{3.3}$$

The above lemma is formulated for any BMS channel, and we will also use it for the proof of the general case in Sections 3.3-3.4. The proof of this lemma uses the symmetry of linear codes with respect to shifting by a codeword and additive structure of BSC, together with the fact that a BMS channel can be represented as a convex combination of several BSC subchannels. The proof is deferred to Section 3.5.1 at the end of this chapter.

Note that $\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{V}=\boldsymbol{0})$ does not in fact depend on the matrix g, since $\boldsymbol{0}\cdot g=\boldsymbol{0}$, and so randomness here only comes from the usage of the channel W. Specifically, $\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{V}=\boldsymbol{0}) = p^{wt(\boldsymbol{y})}(1-p)^{\ell-wt(\boldsymbol{y})}$, where we denote by $wt(\boldsymbol{y})$ the Hamming weight of $\boldsymbol{y}$. Then in (3.3) we obtain

$$\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right] = \sum_{\boldsymbol{y}\in\mathcal{Y}^\ell} p^{wt(\boldsymbol{y})}(1-p)^{\ell-wt(\boldsymbol{y})}\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})\right].$$

Define a typical set. The above expression allows us to only consider "typical" outputs $\boldsymbol{y}$ for the all-zero input while approximating $\mathbb{E}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right]$. For the BSC case, we consider $\boldsymbol{y}$ to be typical when $|wt(\boldsymbol{y}) - \ell p| \leq 2\sqrt{\ell}\log\ell$. Then we can write:

$$\mathop{\mathbb{E}}_{g\sim G}\Big[H^{(g)}(V_1|\boldsymbol{Y})\Big] \geq \sum_{|wt(\boldsymbol{y})-\ell p|\leq 2\sqrt{\ell}\log\ell} p^{wt(\boldsymbol{y})}(1-p)^{\ell-wt(\boldsymbol{y})}\mathop{\mathbb{E}}_{g\sim G}\Big[H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})\Big]. \quad (3.4)$$

Fix a typical output. Let us fix any typical $\boldsymbol{y} \in \mathcal{Y}^\ell$ such that $|wt(\boldsymbol{y}) - \ell p| \leq 2\sqrt{\ell}\log\ell$, and show that $\mathbb{E}_{g\sim G}[H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})]$ is very close to 1. To do this, we first notice that

$$H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y}) = h\left(\frac{\mathbb{P}^{(g)}(V_1=0,\boldsymbol{Y}=\boldsymbol{y})}{\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y})}\right). \quad (3.5)$$

Denote $\widetilde{\boldsymbol{V}} = \boldsymbol{V}^{[2:k]}$ to be bits 2 to k of vector $\boldsymbol{V}$, and by $\widetilde{g} = g[2:k]$ the matrix g without its first row. Next we define the shifted weight distributions of the codebooks generated by g and $\widetilde{g}$:

$$\begin{aligned} B_g(d,\boldsymbol{y}) &:= |\{\mathbf{v} \in \{0,1\}^k \setminus \mathbf{0} \quad : wt(\mathbf{v}g+\boldsymbol{y}) = d\}|, \\ \widetilde{B}_g(d,\boldsymbol{y}) &:= |\{\widetilde{\mathbf{v}} \in \{0,1\}^{k-1} \setminus \mathbf{0} : wt(\widetilde{\mathbf{v}}\widetilde{g}+\boldsymbol{y}) = d\}|. \end{aligned}$$

Therefore,

$$\begin{aligned} \frac{\mathbb{P}^{(g)}(V_1=0,\boldsymbol{Y}=\boldsymbol{y})}{\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y})} &= \frac{\sum_{\widetilde{\boldsymbol{u}}}\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y}\big|V_1=0,\widetilde{\boldsymbol{V}}=\widetilde{\boldsymbol{u}})}{\sum_{\boldsymbol{u}}\mathbb{P}^{(g)}(\boldsymbol{Y}=\boldsymbol{y}\big|\boldsymbol{V}=\boldsymbol{u})} \\ &= \frac{p^{wt(\boldsymbol{y})}(1-p)^{\ell-wt(\boldsymbol{y})} + \sum_{d=0}^{\ell}\widetilde{B}_g(d,\boldsymbol{y})p^d(1-p)^{\ell-d}}{p^{wt(\boldsymbol{y})}(1-p)^{\ell-wt(\boldsymbol{y})} + \sum_{d=0}^{\ell}B_g(d,\boldsymbol{y})p^d(1-p)^{\ell-d}}. \quad (3.6) \end{aligned}$$

We will prove a concentration of the above expression around 1/2, which will then imply that $H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})$ is close to 1 with high probability by (3.5). To do this, we will prove concentrations around means for both numerator and denominator of the above ratio. Since the following arguments work in exactly the same way, let us only consider the denominator for now.

By definition,

$$B_g(d,\boldsymbol{y}) = \sum_{\mathbf{v}\neq\mathbf{0}}\mathbb{1}[wt(\mathbf{v}g+\boldsymbol{y})=d]. \quad (3.7)$$

The expectation and variance of each summand is

$$\mathop{\mathrm{Var}}_{g\sim G}\mathbb{1}\Big[wt(\mathbf{v}g+\boldsymbol{y})=d\Big] \leq \mathop{\mathbb{E}}_{g\sim G}\mathbb{1}\Big[wt(\mathbf{v}g+\boldsymbol{y})=d\Big] = \binom{\ell}{d}2^{-\ell} \qquad \forall\mathbf{v}\in\{0,1\}^k\setminus\mathbf{0}.$$

Clearly, the summands in (3.7) are pairwise independent. Therefore,

$$\mathop{\mathrm{Var}}_{g\sim G}\Big[B_g(d,\boldsymbol{y})\Big] \leq \mathop{\mathbb{E}}_{g\sim G}\Big[B_g(d,\boldsymbol{y})\Big] = (2^k-1)\binom{\ell}{d}2^{-\ell}, \quad (3.8)$$

and then

$$\mathop{\mathbb{E}}_{g\sim G}\left[\sum_{d=0}^{\ell} B_g(d,\boldsymbol{y})p^d(1-p)^{\ell-d}\right] = (2^k-1)2^{-\ell}\left(\sum_{d=0}^{\ell}\binom{\ell}{d}p^d(1-p)^{\ell-d}\right) = (2^k-1)2^{-\ell}.$$

Let us now show that $\sum_{d=0}^{\ell} B_g(d,\boldsymbol{y})p^d(1-p)^{\ell-d}$ is tightly concentrated around its mean for $g\sim G$. To do this, we split the range of d into two parts: when $|d-\ell p| > 6\sqrt{\ell}\log\ell$, and when $|d-\ell p|\le 6\sqrt{\ell}\log\ell$:

$$\sum_{d=0}^{\ell} B_g(d,\boldsymbol{y})p^d(1-p)^{\ell-d} = \sum_{|d-\ell p|>6\sqrt{\ell}\log\ell} B_g(d,\boldsymbol{y})p^d(1-p)^{\ell-d} + \sum_{|d-\ell p|\le 6\sqrt{\ell}\log\ell} B_g(d,\boldsymbol{y})p^d(1-p)^{\ell-d}.$$

In the proof below we will use the following multiplicative form of Chernoff bound applied to a binomial random variable:

$$\mathop{\mathbb{P}}_{X\sim\mathrm{Binom}(\ell,p)}\left[|X-\ell p|\ge\delta\ell p\right]\le 2e^{-\ell p\delta^2/3} \qquad \text{for all } 0\le\delta\le 1. \tag{3.9}$$

Applying this for $\delta=\frac{6\log\ell}{p\ell^{1/2}}$, we have

$$\mathop{\mathbb{P}}_{X\sim\mathrm{Binom}(\ell,p)}\left[|X-\ell p|\ge 6\sqrt{\ell}\log\ell\right] = \sum_{|d-\ell p|\ge 6\sqrt{\ell}\log\ell}\binom{\ell}{d}p^d(1-p)^{\ell-d}\le 2e^{\frac{-12\log^2\ell}{p}} < 2\ell^{-12\log\ell}. \tag{3.10}$$

Negligible part. Denote $Z_g(\boldsymbol{y})=\sum\limits_{|d-\ell p|>6\sqrt{\ell}\log\ell} B_g(d,\boldsymbol{y})p^d(1-p)^{\ell-d}$, and notice that

$$\mathop{\mathbb{E}}_{g\sim G}[Z_g(\boldsymbol{y})] = (2^k-1)2^{-\ell}\sum_{|d-\ell p|>6\sqrt{\ell}\log\ell}\binom{\ell}{d}p^d(1-p)^{\ell-d}\le(2^k-1)2^{-\ell}\cdot 2\ell^{-12\log\ell},$$

where we used (3.8) and (3.10). Then Markov's inequality gives

$$\mathop{\mathbb{P}}_{g\sim G}\left[Z_g(\boldsymbol{y})\ge\mathop{\mathbb{E}}_{g\sim G}[Z_g(\boldsymbol{y})]\ell^{2\log\ell}\right]\le\ell^{-2\log\ell},$$

and so

$$\mathbb{P}\left[Z_g(\boldsymbol{y})<2(2^k-1)2^{-\ell}\ell^{-10\ell\log\ell}\right]\ge 1-\ell^{-2\log\ell}.$$

Define the set of matrices for which $Z_g(\boldsymbol{y})$ is indeed negligible as

$$\mathcal{G}_1 := \{g\in\{0,1\}^{k\times\ell} \;:\; Z_g(\boldsymbol{y})<2(2^k-1)2^{-\ell}\ell^{-10\ell\log\ell}\}, \tag{3.11}$$

and then $\mathop{\mathbb{P}}_{g\sim G}[g\in\mathcal{G}_1]\ge 1-\ell^{-2\log\ell}$.

Substantial part. Now we deal with the part when $|d-\ell p| \le 6\sqrt{\ell}\log \ell$. For now, let us fix any d in this interval, and use Chebyshev's inequality together with (3.8):

$$\begin{aligned}\mathop{\mathbb{P}}_{g\sim G}\left[\Big|B_g(d,\boldsymbol{y})-\mathbb{E}[B_g(d,\boldsymbol{y})]\Big| \ge \ell^{-2\log \ell}\,\mathbb{E}[B_g(d,\boldsymbol{y})]\right] &\le \frac{\mathrm{Var}[B_g(d,\boldsymbol{y})]}{\ell^{-4\log\ell}\,\mathbb{E}^2[B_g(d,\boldsymbol{y})]} \\ &\le \frac{\ell^{4\log\ell}}{\mathop{\mathbb{E}}_{g\sim G}[B_g(d,\boldsymbol{y})]} \le \ell^{4\log\ell}\,\frac{2^{\ell-k+1}}{\binom{\ell}{d}}.\end{aligned} \tag{3.12}$$

We use the following bound on the binomial coefficients

Fact 3.3 ([MS77], Chapter 10, Lemma 7)**.** *For any integer* $0 \le d \le \ell$*,*

$$\frac{1}{\sqrt{2\ell}}2^{\ell h(d/\ell)} \le \binom{\ell}{d} \le 2^{\ell h(d/\ell)} \tag{3.13}$$

Since we fixed $|d-\ell p| \le 6\sqrt{\ell}\log\ell$, Propositions 2.8 and 2.7 imply

$$\left|h(p)-h\left(\frac{d}{\ell}\right)\right| \le h(6\ell^{-1/2}\log\ell) \le 12\ell^{-1/2}\log\ell\cdot\log\frac{\ell^{1/2}}{6\log\ell} \le 6\ell^{-1/2}\log^2\ell. \tag{3.14}$$

Recalling that we consider the above-capacity regime with $k \ge \ell(1-h(p))+8\sqrt{\ell}\log^2\ell$, we derive from (3.13) and (3.14)

$$\frac{2^{\ell-k+1}}{\binom{\ell}{d}} \le \sqrt{2\ell}\cdot 2^{\ell\left[h(p)-h\left(\frac{d}{\ell}\right)-8\ell^{-1/2}\log^2\ell\right]} \le \sqrt{2\ell}\cdot 2^{-2\ell^{1/2}\log^2\ell}.$$

Therefore, we get in (3.12):

$$\mathop{\mathbb{P}}_{g\sim G}\left[\Big|B_g(d,\boldsymbol{y})-\mathbb{E}[B_g(d,\boldsymbol{y})]\Big| \ge \ell^{-2\log\ell}\,\mathbb{E}[B_g(d,\boldsymbol{y})]\right] \le \sqrt{2\ell}\cdot\ell^{4\log\ell}\,2^{-2\ell^{1/2}\log^2\ell} \le \ell^{-\sqrt{\ell}-1}, \tag{3.15}$$

where the last inequality holds since $\ell \ge 8$. Finally, we denote by $\mathcal{G}_2$ the set of binary matrices from $\{0,1\}^{k\times\ell}$ for which $B_g(d,\boldsymbol{y})$ does not deviate much from its expectation for all d such that $|d-\ell p| \le 6\sqrt{\ell}\log\ell$:

$$\mathcal{G}_2 := \left\{g\ :\ \Big|B_g(d,\boldsymbol{y})-\mathbb{E}[B_g(d,\boldsymbol{y})]\Big| \le \ell^{-2\log\ell}\,\mathbb{E}[B_g(d,\boldsymbol{y})]\ \text{ for all } |d-\ell p| \le 6\sqrt{\ell}\log\ell\right\}. \tag{3.16}$$

Then by a simple union bound applied to (3.15) for all d such that $|d-\ell p| \le 6\sqrt{\ell}\log\ell$ we obtain

$$\mathop{\mathbb{P}}_{g\sim G}[g\in\mathcal{G}_2] \ge 1-\ell^{-\sqrt{\ell}}.$$

We are now ready to combine these bounds to get the needed concentration.

Lemma 3.4. *Fix* $\boldsymbol{y}$*. With probability at least* $1 - 2\ell^{-2\log\ell}$ *over the choice of* $g \sim G$*,*

$$(2^k - 1)2^{-\ell}(1 - 2\ell^{-2\log\ell}) \leq \sum_{d=0}^{\ell} B_g(d, \boldsymbol{y})p^d(1-p)^{\ell-d} \leq (2^k - 1)2^{-\ell}(1 + 2\ell^{-2\log\ell}). \quad (3.17)$$

Proof. Indeed, by union bound $\mathbb{P}_{g\sim G}[g \in \mathcal{G}_1 \cap \mathcal{G}_2] \geq 1 - \ell^{-2\log\ell} - \ell^{-\sqrt{\ell}} \geq 1 - 2\ell^{-2\log\ell}$. But for any $g \in \mathcal{G}_1 \cap \mathcal{G}_2$ we derive

$$\begin{aligned}\sum_{d=0}^{\ell} B_g(d, \boldsymbol{y})p^d(1-p)^{\ell-d} \geq & \sum_{|d-\ell p|\leq 6\sqrt{\ell}\log\ell} B_g(d, \boldsymbol{y})p^d(1-p)^{\ell-d} \\ \overset{(a)}{\geq} & (2^k - 1)2^{-\ell}(1 - \ell^{-2\log\ell}) \sum_{|d-\ell p|\leq 6\sqrt{\ell}\log\ell} \binom{\ell}{d} p^d(1-p)^{\ell-d} \\ \overset{(b)}{\geq} & (2^k - 1)2^{-\ell}(1 - \ell^{-2\log\ell})(1 - 2\ell^{-12\log\ell}) \\ \geq & (2^k - 1)2^{-\ell}(1 - 2\ell^{-2\log\ell}),\end{aligned}$$

where (a) follows from (3.16) (since $g \in \mathcal{G}_2$) and the expression in (3.8) for $\mathbb{E}[B_g(d, \boldsymbol{y})]$, and (b) uses the concentration inequality for binomial random variable from (3.10). On the other hand, we can upper bound this expression as

$$\begin{aligned}&\sum_{d=0}^{\ell} B_g(d, \boldsymbol{y})p^d(1-p)^{\ell-d} \\ &= \sum_{|d-\ell p|\leq 6\sqrt{\ell}\log\ell} B_g(d, \boldsymbol{y})p^d(1-p)^{\ell-d} + \sum_{|d-\ell p|> 6\sqrt{\ell}\log\ell} B_g(d, \boldsymbol{y})p^d(1-p)^{\ell-d} \\ &\overset{(a)}{\leq} (2^k - 1)2^{-\ell}(1 + \ell^{-2\log\ell}) \sum_{|d-\ell p|\leq 6\sqrt{\ell}\log\ell} \binom{\ell}{d} p^d(1-p)^{\ell-d} + Z_g(\boldsymbol{y}) \\ &\overset{(b)}{\leq} (2^k - 1)2^{-\ell}(1 + \ell^{-2\log\ell}) + 2(2^k - 1)2^{-\ell}\ell^{-10\log\ell} \\ &\leq (2^k - 1)2^{-\ell}(1 + 2\ell^{-2\log\ell}),\end{aligned}$$

where (a) is again from (3.16) and (3.8) and the notation $Z_g(\boldsymbol{y})$ for the negligible part, and (b) is from (3.11) (as g is in $\mathcal{G}_1$). □

We similarly obtain the concentration for the sum in the numerator of (3.6): with probability at least $1 - 2\ell^{-2\log\ell}$ over the choice of g, it holds

$$(2^{k-1} - 1)2^{-\ell}(1 - 2\ell^{-2\log\ell}) \leq \sum_{d=0}^{\ell} \widetilde{B}_g(d, \boldsymbol{y})p^d(1-p)^{\ell-d} \leq (2^{k-1} - 1)2^{-\ell}(1 + 2\ell^{-2\log\ell}). \quad (3.18)$$

Next, let us use the fact that we took a typical output $\boldsymbol{y}$ with $|wt(\boldsymbol{y}) - \ell p| \leq 2\sqrt{\ell}\log\ell$ to show that the terms $p^{wt(\boldsymbol{y})}(1-p)^{\ell - wt(\boldsymbol{y})}$ are negligible in both numerator and denominator of (3.6). We have

$$p^{wt(\boldsymbol{y})}(1-p)^{\ell-wt(\boldsymbol{y})} = \left(\frac{1-p}{p}\right)^{\ell p - wt(\boldsymbol{y})} \cdot p^{\ell p}(1-p)^{\ell-\ell p} = 2^{(\ell p - wt(\boldsymbol{y}))\cdot\log\left(\frac{1-p}{p}\right)} \cdot 2^{-\ell h(p)}. \quad (3.19)$$

Simple case analysis gives us:

(a) If $p < \frac{1}{\sqrt{\ell}}$, then $(\ell p - wt(\boldsymbol{y})) \cdot \log\left(\frac{1-p}{p}\right) \leq \ell p \log \frac{1}{p} < \ell \frac{1}{\sqrt{\ell}} \log \sqrt{\ell} < \sqrt{\ell} \log^2 \ell$;

(b) In case $p \geq \frac{1}{\sqrt{\ell}}$, obtain $(\ell p - wt(\boldsymbol{y})) \cdot \log\left(\frac{1-p}{p}\right) \leq 2\sqrt{\ell} \log \ell \cdot \log \frac{1}{p} \leq \sqrt{\ell} \log^2 \ell$.

Using the above in (3.19) we derive for $k \geq \ell(1 - h(p)) + 8\sqrt{\ell} \log^2 \ell$

$$p^{wt(\boldsymbol{y})}(1-p)^{\ell - wt(\boldsymbol{y})} \leq 2^{\sqrt{\ell}\log^2 \ell - \ell h(p)} \leq 2^{2\sqrt{\ell}\log^2 \ell - \ell h(p) - 2\log^2 \ell - 2} \leq \ell^{-2\log \ell}\left(2^{k-1} - 1\right)2^{-\ell}.$$

Combining this with (3.17) and (3.18) and using a union bound we derive that with probability at least $1 - 4\ell^{-2\log \ell}$ it holds

$$\left|\left(p^{wt(\boldsymbol{y})}(1-p)^{\ell - wt(\boldsymbol{y})} + \sum_{d=0}^{\ell} B_g(d, \boldsymbol{y}) p^d (1-p)^{\ell-d}\right) - (2^k - 1)2^{-\ell}\right| \leq 3\ell^{-2\log \ell} \cdot (2^k - 1)2^{-\ell},$$

$$\left|\left(p^{wt(\boldsymbol{y})}(1-p)^{\ell - wt(\boldsymbol{y})} + \sum_{d=0}^{\ell} \widetilde{B}_g(d, \boldsymbol{y}) p^d (1-p)^{\ell-d}\right) - (2^{k-1} - 1)2^{-\ell}\right| \leq 3\ell^{-2\log \ell} \cdot (2^{k-1} - 1)2^{-\ell}.$$

Therefore, with probability at least $1 - 4\ell^{-2\log \ell}$ the expression in (3.6) is bounded as

$$\frac{(1 - 3\ell^{-2\log \ell})(2^{k-1} - 1)2^{-\ell}}{(1 + 3\ell^{-2\log \ell})(2^k - 1)2^{-\ell}} \leq \frac{\mathbb{P}^{(g)}(V_1 = 0, \boldsymbol{Y} = \boldsymbol{y})}{\mathbb{P}^{(g)}(\boldsymbol{Y} = \boldsymbol{y})} \leq \frac{(1 + 3\ell^{-2\log \ell})(2^{k-1} - 1)2^{-\ell}}{(1 - 3\ell^{-2\log \ell})(2^k - 1)2^{-\ell}}. \tag{3.20}$$

We can finally derive:

$$\begin{aligned} \frac{(1 - 3\ell^{-2\log \ell})(2^{k-1} - 1)}{(1 + 3\ell^{-2\log \ell})(2^k - 1)} &\geq (1 - 6\ell^{-2\log \ell})\left(\frac{1}{2} - 2^{-k}\right) \geq (1 - 6\ell^{-2\log \ell})\left(\frac{1}{2} - \ell^{-8\sqrt{\ell}\log \ell}\right) \\ &\geq \frac{1}{2} - \ell^{-\log \ell}, \end{aligned} \tag{3.21}$$

$$\frac{(1 + 3\ell^{-2\log \ell})(2^{k-1} - 1)}{(1 - 3\ell^{-2\log \ell})(2^k - 1)} \leq (1 + 9\ell^{-2\log \ell})\frac{1}{2} \leq \frac{1}{2} + \ell^{-\log \ell}.$$

Therefore, with probability at least $1 - 4\ell^{-2\log \ell}$ over $g \sim G$ it holds

$$\left|\frac{\mathbb{P}^{(g)}(V_1 = 0, \boldsymbol{Y} = \boldsymbol{y})}{\mathbb{P}^{(g)}(\boldsymbol{Y} = \boldsymbol{y})} - \frac{1}{2}\right| \leq \ell^{-\log \ell}. \tag{3.22}$$

Since $h(1/2 + x) \geq 1 - 4x^2$ for any $x \in [-1/2, 1/2]$ ([Top01, Theorem 1.2]), we then derive:

$$\begin{aligned} \mathop{\mathbb{E}}_{g \sim G}\left[H^{(g)}(V_1 | \boldsymbol{Y} = \boldsymbol{y})\right] = \mathop{\mathbb{E}}_{g \sim G}\left[h\left(\frac{\mathbb{P}^{(g)}(V_1 = 0, \boldsymbol{Y} = \boldsymbol{y})}{\mathbb{P}^{(g)}(\boldsymbol{Y} = \boldsymbol{y})}\right)\right] &\geq (1 - 4\ell^{-2\log \ell})(1 - 4\ell^{-2\log \ell}) \\ &\geq 1 - 8\ell^{-2\log \ell}. \end{aligned}$$

Concentration of entropy. We are now ready to plug this into (3.4):

$$\begin{aligned}
\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right] &\geq (1-8\ell^{-2\log\ell}) \sum_{|wt(\boldsymbol{y})-\ell p|\leq 2\sqrt{\ell}\log\ell} p^{wt(\boldsymbol{y})}(1-p)^{\ell-wt(\boldsymbol{y})} \\
&= (1-8\ell^{-2\log\ell}) \sum_{|d-\ell p|\leq 2\sqrt{\ell}\log\ell} \binom{\ell}{d} p^d(1-p)^{\ell-d} \\
&= (1-8\ell^{-2\log\ell}) \mathop{\mathbb{P}}_{X\sim\text{Binom}(\ell,p)}\left[|X-\ell p|\leq 2\sqrt{\ell}\log\ell\right] \\
&\geq (1-8\ell^{-2\log\ell})(1-2e^{-(4\log^2\ell)/3p}) \\
&\geq (1-8\ell^{-2\log\ell})(1-2\ell^{-2\log\ell}) \\
&\geq 1-10\ell^{-2\log\ell}, \qquad (3.23)
\end{aligned}$$

where the second inequality is obtained from the Chernoff bound (3.9) with $\delta = \frac{2\log\ell}{p\ell^{1/2}}$, and the third inequality follows from $p \leq 1/2$ and $e^{-8/3} < 2^{-2}$. Finally, using the fact that $H^{(g)}(V_1|\boldsymbol{Y}) \leq 1$, Markov's inequality, and (3.23), we get

$$\begin{aligned}
\mathop{\mathbb{P}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y}) \leq 1-\ell^{-\log\ell}\right] &= \mathop{\mathbb{P}}_{g\sim G}\left[1-H^{(g)}(V_1|\boldsymbol{Y}) \geq \ell^{-\log\ell}\right] \\
&\leq \frac{\mathop{\mathbb{E}}_{g\sim G}\left[1-H^{(g)}(V_1|\boldsymbol{Y})\right]}{\ell^{-\log\ell}} \leq 10\ell^{-\log\ell}.
\end{aligned}$$

Thus we conclude that with probability at least $1-10\ell^{-\log\ell}$ over the choice of the kernel G it holds that $H(V_1 \mid \boldsymbol{Y}) \geq 1-\ell^{-\log\ell}$ when $k \geq \ell(1-h(p)) + 8\sqrt{\ell}\log^2\ell$ and the underlying channel is BSC. This completes the proof of Theorem 3.1 for the BSC case. □

3.3 Strong converse for BMS channels with bounded alphabet size

This section is devoted to proving Theorem 3.1 for the case when $W : \{0,1\} \to \mathcal{Y}$ is a BMS channel which has a bounded output alphabet size, specifically $|\mathcal{Y}| \leq 2\sqrt{\ell}$.

3.3.1 Notations and settings

We will use the fact that any BMS can be viewed as a convex combination of BSCs (see for example [LH06, Kor09]), and generalize the ideas of the previous section. One can think of the channel W as follows: it consists of m possible underlying BSC subchannels $W^{(1)}, W^{(2)}, \ldots, W^{(m)}$. On any input, W randomly chooses one of the subchannels it is going to use with probabilities $q_1, q_2, \ldots, q_m$ respectively. The subchannel $W^{(j)}$ has crossover probability p_j, and without loss of generality $0 \leq p_1 \leq p_2 \leq \cdots \leq p_m \leq \frac{1}{2}$. The subchannel $W^{(j)}$ has two possible output symbols $z_j^{(0)}$ or $z_j^{(1)}$, corresponding to 0 and 1, respectively (i.e. 0 goes to $z_j^{(0)}$ with probability $1-p_j$, or to $z_j^{(1)}$ with probability p_j under $W^{(j)}$). Then the whole output alphabet is $\mathcal{Y} = \{z_1^{(0)}, z_1^{(1)}, z_2^{(0)}, z_2^{(1)}, \dots, z_m^{(0)}, z_m^{(1)}\}$, $|\mathcal{Y}| = 2m \le 2\sqrt{\ell}$. The conditional entropy of the BMS channel W can be expressed as $H(W) = \sum\limits_{i=1}^{m} q_i h(p_i)$, i.e. it is a convex combination of entropies of the subchannels $W^{(1)}, W^{(2)}, \dots, W^{(m)}$ with the corresponding coefficients $q_1, q_2, \dots, q_m$.

Remark 3.5. *Above we ignored the case when some of the subchannels have only one output (i.e. BEC subchannels). See [TV13, Lemma 4] for a proof that we can do this without loss of generality.*

In this section the expectation is only going to be taken over the randomness of the matrix $g \sim G$, so we omit this in some places. As in the BSC case, by $\mathbb{P}^{(g)}[\cdot]$ and $H^{(g)}(\cdot)$ we denote the probability and entropy only over the randomness of the channel and the message, *for a fixed kernel g*.

For any possible output $\boldsymbol{y} \in \mathcal{Y}^\ell$ we denote by d_i the number of symbols from $\{z_i^{(0)}, z_i^{(1)}\}$ it has (i.e. the number of uses of the $W^{(i)}$ subchannel), so $\sum_{i=1}^m d_i = \ell$. Let also t_i be the number of symbols $z_i^{(1)}$ in $\boldsymbol{y}$. Then

$$\mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{V} = \boldsymbol{0}] = \prod_{i=1}^{m} q_i^{d_i} p_i^{t_i} (1 - p_i)^{d_i - t_i}. \tag{3.24}$$

Remark 3.6. *For this case of bounded output alphabet size, we will consider the above-capacity regime when $k \ge \ell(1 - H(W)) + 13\ell^{1/2} \log^3 \ell$ (note that this is made intentionally weaker than the condition in Theorem* 3.1*).*

We will follow the same blueprint of the proof for BSC from Section 3.1, however all the technicalities along the way are going to be more challenging. In particular, while we were dealing with one binomial distribution in Section 3.2, here we will face a multinomial distribution of $(d_1, d_2, \dots, d_m)$ as a choice of which subchannels to use, as well as binomial distributions $t_i \sim \text{Binom}(d_i, p_i)$ which correspond to "flips" within one subchannel.

3.3.2 Proof of Theorem 3.1 begins

As in the BSC case, we are going to lower bound the expectation of $H^{(g)}(V_1 | \boldsymbol{Y})$ and use Markov's inequality afterwards.

Restrict to zero-input. We use Lemma 3.2 to write

$$\mathop{\mathbb{E}}_{g \sim G} \left[H^{(g)}(V_1 | \boldsymbol{Y}) \right] = \sum_{\boldsymbol{y} \in \mathcal{Y}^\ell} \mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{V} = \boldsymbol{0}] \mathop{\mathbb{E}}_{g \sim G} \left[H^{(g)}(V_1 | \boldsymbol{Y} = \boldsymbol{y}) \right]. \tag{3.25}$$

Notice that there is no dependence of $\mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{V} = \boldsymbol{0}]$ on the kernel g, since the output for the zero-input depends only on the randomness of the channel.

Typical output set. As for the binary case, we would like to consider the set of "typical" outputs (for input $\mathbf{0}$) from $\mathcal{Y}^\ell$. We define $\boldsymbol{y} \in \mathcal{Y}^\ell$ to be *typical* if

$$\sum_{i=1}^{m}(\ell \cdot q_i - d_i)h(p_i) \le 2\sqrt{\ell}\log \ell, \tag{3.26}$$

$$\sum_{i=1}^{m}(p_i d_i - t_i)\log\left(\frac{1-p_i}{p_i}\right) \le 3\sqrt{\ell}\log^2 \ell. \tag{3.27}$$

By typicality of this set we mean the following

Lemma 3.7. $\sum\limits_{\boldsymbol{y}\ \text{typical}} \mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{V} = \mathbf{0}] \ge 1 - \ell^{-\log \ell}$. *In other words, on input* $\mathbf{0}$*, the probability to get the output string which is not typical (for which* (3.26) *or* (3.27) *is not satisfied) is at most* $\ell^{-\log \ell}$.

We defer the proof of this lemma until Section 3.3.5, until after we see why we are actually interested in these conditions on $\boldsymbol{y}$.

3.3.3 Fix a typical output

For this part, let us fix one $\boldsymbol{y} \in \mathcal{Y}^\ell$ which is typical and prove that $\mathbb{E}_g\left[H^{(g)}(V_1|\boldsymbol{Y})\right]$ is very close to 1. We have

$$H^{(g)}(V_1|\boldsymbol{Y}) = h\left(\frac{\mathbb{P}^{(g)}\left[V_1 = 0, \boldsymbol{Y} = \boldsymbol{y}\right]}{\mathbb{P}^{(g)}\left[\boldsymbol{Y} = \boldsymbol{y}\right]}\right). \tag{3.28}$$

Similar to the BSC case, we will prove that both the denominator and numerator of the fraction inside the entropy function above are tightly concentrated around their means. The arguments for the denominator and the numerator are almost exactly the same, so we only consider the denominator for now.

Concentration for $\mathbb{P}^{(g)}\left[\boldsymbol{Y} = \boldsymbol{y}\right]$

Define now the shifted weight distributions for the codebook g with respect to m different underlying BSC subchannels. First, for any $x \in \{0,1\}^\ell$ and $i = 1, 2, \dots, m$, define

$$\text{dist}_i(x, \boldsymbol{y}) = |\{\text{positions } j \text{ such that } (x_j = 0, \boldsymbol{y}_j = z_i^{(1)}) \text{ or } (x_j = 1, \boldsymbol{y}_j = z_i^{(0)})\}|.$$

That is, if you send x through W^ℓ and receive $\boldsymbol{y}$, then $\text{dist}_i(x, \boldsymbol{y})$ is just the number of coordinates where the subchannel i was chosen, and the bit was flipped.

In our settings, we now need to think of "distance" between some binary vector $\mathbf{x} \in \{0,1\}^\ell$ and $\boldsymbol{y}$ as of an integer vector $\boldsymbol{s} = (s_1, s_2, \dots, s_m)$, where $0 \le s_i \le d_i$ for $i \in [m]$, where $s_i = \text{dist}_i(\mathbf{x}, \boldsymbol{y})$ is just the number of flips that occurred in the usage of i^{th} subchannel when going from $\mathbf{x}$ to $\boldsymbol{y}$. In other words, s_i is just the Hamming distance between the parts of $\mathbf{x}$ and $\boldsymbol{y}$ which correspond to coordinates j where $\boldsymbol{y}_j$ is $z_i^{(0)}$ or $z_i^{(1)}$ (outputs obtained from the subchannel $W^{(i)}$).

Now we can formally define shifted weight distributions for our fixed typical $\boldsymbol{y}$. For an integer vector $\boldsymbol{s} = (s_1, s_2, \ldots, s_m)$, where $0 \leq s_i \leq d_i$ define

$$B_g(\boldsymbol{s}, \boldsymbol{y}) = \left| \mathbf{v} \in \{0,1\}^k \setminus \mathbf{0} \ : \ \mathrm{dist}_i(\mathbf{v} \cdot g, \boldsymbol{y}) = s_i \quad \text{for } i = 1, 2, \ldots, m \right|.$$

We can express $\mathbb{P}^{(g)}[\mathcal{Y} = \boldsymbol{y}]$ in terms of $B_g(\boldsymbol{s}, \boldsymbol{y})$ as follows:

$$2^k \cdot \mathbb{P}^{(g)}[\mathcal{Y} = \boldsymbol{y}] = \mathbb{P}[\mathcal{Y} = \boldsymbol{y} | \mathbf{v} = \mathbf{0}] + \sum_{\substack{0 \leq s_j \leq d_j \\ j=1,2,\ldots,m}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1 - p_i)^{d_i - s_i}, \tag{3.29}$$

because $\prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1 - p_i)^{d_i - s_i}$ is exactly the probability to get output $\boldsymbol{y}$ if a $\mathbf{v}$ is sent that satisfies $\mathrm{dist}_i(\mathbf{v} \cdot g, \boldsymbol{y}) = s_i \quad$ for $i = 1, 2, \ldots, m$.

We have:

$$B_g(\boldsymbol{s}, \boldsymbol{y}) = \sum_{\mathbf{v} \neq \mathbf{0}} \mathbb{1}\Big[\mathrm{dist}_i(\mathbf{v} \cdot g, \boldsymbol{y}) = s_i, \quad \forall i = 1, 2, \ldots, m\Big]. \tag{3.30}$$

For a fixed non-zero $\mathbf{v}$ but uniformly random binary matrix g, the vector $\mathbf{v} \cdot g$ is just a uniformly random vector from $\{0,1\}^\ell$. Now, the number of vectors $\mathbf{x}$ in $\{0,1\}^\ell$ such that $\mathrm{dist}_i(\mathbf{x}, \boldsymbol{y}) = s_i \ \ \forall i = 1, 2, \ldots, m$, is $\prod_{i=1}^{m} \binom{d_i}{s_i}$, since for any $i = 1, 2, \ldots, m$, we need to choose which of the s_i coordinates amongst the d_i uses of the subchannel $W^{(i)}$, got flipped. Therefore

$$\mathop{\mathbb{P}}_{g \sim G}\Big[\mathrm{dist}_i(\mathbf{v} \cdot g, \boldsymbol{y}) = s_i, \ \ \forall i = 1, 2, \ldots, m\Big] = 2^{-\ell} \prod_{i=1}^{m} \binom{d_i}{s_i}.$$

Then for the expectation of the shifted weight distributions we obtain

$$\mathop{\mathbb{E}}_{g \sim G}[B_g(\boldsymbol{s}, \boldsymbol{y})] = \sum_{\mathbf{v} \neq \mathbf{0}} \mathop{\mathbb{P}}_{g \sim G}\Big[\mathrm{dist}_i(\mathbf{v} \cdot g, \boldsymbol{y}) = s_i, \ \ \forall i = 1, 2, \ldots, m\Big] = \frac{2^k - 1}{2^\ell} \prod_{i=1}^{m} \binom{d_i}{s_i}. \tag{3.31}$$

For the expectation of the summation in the RHS of (3.29), which we denote as E, we derive:

$$\begin{aligned}
E &:= \mathop{\mathbb{E}}_{g \sim G}\left[\sum_{\substack{0 \leq s_j \leq d_j \\ j=1,2,\ldots,m}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1 - p_i)^{d_i - s_i} \right] \\
&= \left(\prod_{i=1}^{m} q_i^{d_i} \right) \cdot \sum_{\substack{0 \leq s_j \leq d_j \\ j=1,2,\ldots,m}} \left(\mathop{\mathbb{E}}_{g \sim G}\Big[B_g(\boldsymbol{s}, \boldsymbol{y})\Big] \cdot \prod_{i=1}^{m} p_i^{s_i} (1 - p_i)^{d_i - s_i} \right) \\
&= \frac{2^k - 1}{2^\ell} \left(\prod_{i=1}^{m} q_i^{d_i} \right) \cdot \sum_{\substack{0 \leq s_j \leq d_j \\ j=1,2,\ldots,m}} \prod_{i=1}^{m} \binom{d_i}{s_i} p_i^{s_i} (1 - p_i)^{d_i - s_i} \\
&= \frac{2^k - 1}{2^\ell} \prod_{i=1}^{m} q_i^{d_i} \cdot \prod_{i=1}^{m} \left(\underbrace{\sum_{s_i=0}^{d_i} \binom{d_i}{s_i} p_i^{s_i} (1 - p_i)^{d_i - s_i}}_{=1} \right) = \frac{2^k - 1}{2^\ell} \prod_{i=1}^{m} q_i^{d_i}.
\end{aligned} \tag{3.32}$$

Next, by (3.30) we can see that $B_g(\boldsymbol{s}, \boldsymbol{y})$ is a sum of pairwise independent indicator random variables, since $\boldsymbol{v}_1 \cdot g$ and $\boldsymbol{v}_2 \cdot g$ are independent for distinct and non-zero $\boldsymbol{v}_1, \boldsymbol{v}_2$. Therefore

$$\operatorname*{Var}_{g\sim G}[B_g(\boldsymbol{s}, \boldsymbol{y})] \leq \operatorname*{\mathbb{E}}_{g\sim G}[B_g(\boldsymbol{s}, \boldsymbol{y})]. \tag{3.33}$$

Splitting the summation in (3.29)

We will split the summation in (3.29) into two parts: for the first part, we will show that the expectation of each term is very large, and then use Chebyshev's inequality to argue that each term is concentrated around its expectation. For the second part, its expectation is going to be very small, and Markov's inequality will imply that this part also does not deviate from its expectation too much with high probability (over the random kernel $g \sim G$). Putting these two arguments together, we will obtain that the sum in the RHS of (3.29) is concentrated around its mean.

To proceed, define a distribution $\Omega = \text{Binom}(d_1, p_1) \times \text{Binom}(d_2, p_2) \times \cdots \times \text{Binom}(d_m, p_m)$, and consider a random vector $\chi \sim \Omega$. In other words, χ has m independent coordinates $\chi_i,\ i = 1, \ldots, m$, where χ_i is a binomial random variable with parameters d_i and p_i. Note that by definition then for any vector $\boldsymbol{s} = (s_1, s_2, \ldots, s_m)$, where $0 \leq s_i \leq d_i$ and s_i is integer for any i, we have

$$\operatorname*{\mathbb{P}}_{\chi}[\chi = \boldsymbol{s}] = \prod_{i=1}^{m} \operatorname*{\mathbb{P}}_{\chi}[\chi_i = s_i] = \prod_{i=1}^{m} \binom{d_i}{s_i} p_i^{s_i}(1 - p_i)^{d_i - s_i}.$$

Let now $\mathcal{T}$ be some subset of $\mathcal{S} = [0 : d_1] \times [0 : d_2] \times \cdots \times [0 : d_m]$, where $[0 : d] = \{0, 1, 2, \ldots, (d-1), d\}$ for integer d. Let also $\mathcal{N}$ be $\mathcal{S} \setminus \mathcal{T}$. Then the summation in the RHS of (3.29) we can write as

$$\sum_{\boldsymbol{s}\in\mathcal{S}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i}(1 - p_i)^{d_i - s_i} = \left(\sum_{\boldsymbol{s}\in\mathcal{T}} + \sum_{\boldsymbol{s}\in\mathcal{N}}\right) B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i}(1 - p_i)^{d_i - s_i}. \tag{3.34}$$

In the next section we describe how we choose $\mathcal{T}$.

3.3.3.1 Substantial part

Exactly as in the binary case, using (3.33) and Chebyshev's inequality, we have for any $s \in \mathcal{S}$

$$\begin{aligned}\operatorname*{\mathbb{P}}_{g\sim G}\left[\Big|B_g(\boldsymbol{s}, \boldsymbol{y}) - \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})]\Big| \geq \ell^{-2\log\ell}\, \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})]\right] &\leq \frac{\text{Var}[B_g(\boldsymbol{s}, \boldsymbol{y})]}{\ell^{-4\log\ell}\, \mathbb{E}^2[B_g(\boldsymbol{s}, \boldsymbol{y})]} \\ &\leq \frac{\ell^{4\log\ell}}{\mathbb{E}_{g\sim G}[B_g(\boldsymbol{s}, \boldsymbol{y})]} \leq \ell^{4\log\ell}\, \frac{2^{\ell-k+1}}{\prod_{i=1}^{m}\binom{d_i}{s_i}}.\end{aligned} \tag{3.35}$$

We need the above to be upper bounded by $\ell^{-2\sqrt{\ell}}$ to be able to use union bound for all $\boldsymbol{s} \in \mathcal{T} \subset \mathcal{S}$, since $|\mathcal{S}| \leq \ell^{O(\sqrt{\ell})}$. Recall that we have $k \geq \ell(1 - H(W)) + 13\ell^{1/2}\log^3\ell$, and then using a lower bound for binomial coefficients from Fact 3.3 we obtain for the RHS of (3.35)

$$\ell^{4\log\ell}\,\frac{2^{\ell-k+1}}{\prod_{i=1}^{m}\binom{d_i}{s_i}} \le \ell^{4\log\ell}\cdot\left(2\prod_{i=1}^{m}\sqrt{2d_i}\right)\cdot 2^{\ell H(W)-\sum_{i=1}^{m} d_i h\left(\frac{s_i}{d_i}\right)-13\ell^{1/2}\log^3\ell}. \tag{3.36}$$

We want to show that the term $2^{-\Omega(\ell^{1/2}\log^3\ell)}$ is the dominant one. First, it is easy to see that $\ell^{4\log\ell}=2^{4\log^2\ell}\le 2^{\ell^{1/2}\log^3\ell}$ for $\ell\ge 4$. To deal with the factor $2\prod_{i=1}^{m}\sqrt{2d_i}$, recall that $\sum_{i=1}^{m} d_i=\ell$ and $m\le\sqrt{\ell}$ in this section (recall discussion at the beginning of Section 3.3), then AM-GM inequality gives us

$$2\prod_{i=1}^{m}\sqrt{2d_i}\le 2\cdot 2^{m/2}\cdot\sqrt{\left(\frac{\sum_{i=1}^{m} d_i}{m}\right)^{m}}=2\cdot\left(\frac{2\ell}{m}\right)^{m/2}\le 2\cdot(2\sqrt{\ell})^{\sqrt{\ell}/2}\le 2^{\ell^{1/2}\log^3\ell}, \tag{3.37}$$

where we used the fact that $(a/x)^x$ is increasing while $x\le a/e$ and the condition $\ell\ge 4$. For the last factor of (3.36) we formulate a lemma.

Lemma 3.8. *There exists a set $\mathcal{T}\subseteq\mathcal{S}=[0:d_1]\times[0:d_2]\times\cdots\times[0:d_m]$, such that $\mathbb{P}_{\chi\sim\Omega}[\chi\in\mathcal{T}]\ge 1-\ell^{-(\log\ell)/4}$, and for any $\boldsymbol{s}\in\mathcal{T}$ it holds that*

$$\ell H(W)-\sum_{i=1}^{m} d_i h\left(\frac{s_i}{d_i}\right)\le 8\,\ell^{1/2}\log^3\ell.$$

($\Omega=Binom(d_1,p_1)\times Binom(d_2,p_2)\times\cdots\times Binom(d_m,p_m)$)

Proof. Rearrange the above summation as follows:

$$\begin{aligned}\ell H(W)-\sum_{i=1}^{m} d_i h\left(\frac{s_i}{d_i}\right) &=\sum_{i=1}^{m}\left(\ell q_i h(p_i)-d_i h\left(\frac{s_i}{d_i}\right)\right)\\ &=\sum_{i=1}^{m}\Big(\ell q_i-d_i\Big)h(p_i)+\sum_{i=1}^{m} d_i\left(h(p_i)-h\left(\frac{s_i}{d_i}\right)\right).\end{aligned}$$

Now recall that we took typical $\boldsymbol{y}$ for now, so by inequality (3.26) from the definition of the typicality of $\boldsymbol{y}$ we already have that the first part of the above sum is upper bounded by $\ell^{1/2}\log^3\ell$.

To deal with the second part, which is $\sum_{i=1}^{m} d_i\left(h(p_i)-h\left(\frac{s_i}{d_i}\right)\right)$, we use a separate Lemma 3.17, since the proof will be almost identical to another concentration inequality we will need later. Lemma 3.17 claims that $\sum_{i=1}^{m} d_i\left(h(p_i)-h\left(\frac{\chi_i}{d_i}\right)\right)\le 7\ell^{1/2}\log^3\ell$ with probability at least $1-\ell^{-(\log\ell)/4}$ over $\chi\sim\Omega$. Then the result of the current lemma follows by taking $\mathcal{T}$ to be the subset of $\mathcal{S}$ where this inequality holds. □

Fix now a set $\mathcal{T}\subseteq\mathcal{S}$ as in Lemma 3.8. Then using the arguments above we conclude that the RHS in (3.36), and therefore (3.35), is bounded above by $2^{-3\ell^{1/2}\log^3\ell}$ for any $\boldsymbol{s}\in\mathcal{T}$. Thus we can apply union bound over $\boldsymbol{s}\in\mathcal{T}$ for (3.35), since $|\mathcal{T}|\le|\mathcal{S}|=\prod_{i=1}^{m}(d_i+1)\le\left(2\sqrt{\ell}\right)^{\sqrt{\ell}}\le 2^{\ell^{1/2}\log^3\ell}$ for $\ell\ge 4$, similarly to (3.37). Therefore, we derive

Corollary 3.9. *With probability at least* $1 - 2^{-2\ell^{1/2}\log^3\ell}$ *(over the random kernel* $g \sim G$*) it holds simultaneously for all* $\boldsymbol{s} \in \mathcal{T}$ *that*

$$\Big| B_g(\boldsymbol{s}, \boldsymbol{y}) - \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})] \Big| \le \ell^{-2\log\ell}\, \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})].$$

Moreover, the set $\mathcal{N} = \mathcal{S} \setminus \mathcal{T}$ satisfies $\mathbb{P}_{\chi\sim\Omega}[\chi \in \mathcal{N}] \le \ell^{-(\log\ell)/4}$, which we will use next section to bound the second part of (3.34).

3.3.3.2 Negligible part

Denote for convenience $Z_g(\boldsymbol{y}) = \sum_{s\in\mathcal{N}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^m q_i^{d_i} p_i^{s_i}(1-p_i)^{d_i-s_i}$, the second part of the RHS of (3.34). Recall the value of $\mathbb{E}_{g\sim G}[B_g(\mathbf{s}, \boldsymbol{Y})]$ from (3.31) and notation of E in (3.32). Then for the expectation of $Z_g(\boldsymbol{y})$ derive

$$\begin{aligned}
\mathop{\mathbb{E}}_{g\sim G}[Z_g(\boldsymbol{y})] &= \left(\prod_{i=1}^m q_i^{d_i}\right) \cdot \sum_{s\in\mathcal{N}} \left(\mathop{\mathbb{E}}_{g\sim G}\Big[B_g(\mathbf{s}, \boldsymbol{y})\Big] \prod_{i=1}^m p_i^{s_i}(1-p_i)^{d_i-s_i} \right) \\
&= \frac{2^k - 1}{2^\ell} \left(\prod_{i=1}^m q_i^{d_i}\right) \cdot \sum_{s\in\mathcal{N}} \prod_{i=1}^m \binom{d_i}{s_i} p_i^{s_i}(1-p_i)^{d_i-s_i} \\
&= E \cdot \mathop{\mathbb{P}}_{\chi\sim\Omega}\Big[\chi \in \mathcal{N}\Big] \\
&\le E \cdot \ell^{-(\log\ell)/4}.
\end{aligned}$$

Markov's inequality implies

Corollary 3.10. *With probability at least* $1 - \ell^{-(\log\ell)/8}$ *(over the random kernel* $g \sim G$*) it holds*

$$Z_g(\boldsymbol{y}) \le \ell^{(\log\ell)/8}\, \mathbb{E}[Z_g(\boldsymbol{y})] \le E \cdot \ell^{-(\log\ell)/8}.$$

3.3.3.3 Putting it together

Combining the Corollaries 3.9 and 3.10 together and using the union bound, derive

Corollary 3.11. *With probability at least* $1 - \ell^{-(\log\ell)/8} - 2^{-2\ell^{1/2}\log^3\ell} \ge 1 - 2\ell^{-(\log\ell)/8}$ *over the randomness of the kernel* $g \sim G$ *it simultaneously holds*

$$\begin{aligned}
&\Big| B_g(\boldsymbol{s}, \boldsymbol{y}) - \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})] \Big| \le \ell^{-2\log\ell}\, \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})], \qquad \textit{for all } \boldsymbol{s} \in \mathcal{T}, \\
&\sum_{s\in\mathcal{N}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^m q_i^{d_i} p_i^{s_i}(1-p_i)^{d_i-s_i} \le E \cdot \ell^{-(\log\ell)/8}.
\end{aligned} \tag{3.38}$$

We are finally ready to formulate the concentration result we need. The following lemma is an analogue of Lemma 3.4 from the BSC case:

Lemma 3.12. *With probability at least* $1 - 2\ell^{-(\log\ell)/8}$ *over the choice of* $g \sim G$ *it holds*

$$\left| \sum_{s\in\mathcal{S}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^m q_i^{d_i} p_i^{s_i}(1-p_i)^{d_i-s_i} - E \right| \le 2\ell^{-(\log\ell)/8} \cdot E.$$

Proof. Let us consider a kernel g such that the conditions (3.38) hold, which happens with probability at least $1 - 2\ell^{-(\log \ell)/8}$ according to Corollary 3.11. Then

$$\begin{aligned}
\sum_{\boldsymbol{s} \in \mathcal{S}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1-p_i)^{d_i - s_i} &\geq \sum_{\boldsymbol{s} \in \mathcal{T}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1-p_i)^{d_i - s_i} \\
&\overset{(3.38)}{\geq} \sum_{\boldsymbol{s} \in \mathcal{T}} \left(1 - \ell^{-2\log \ell}\right) \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})] \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1-p_i)^{d_i - s_i} \\
&\overset{(3.31)}{=} \left(1 - \ell^{-2\log \ell}\right) \frac{2^k - 1}{2^\ell} \prod_{i=1}^{m} q_i^{d_i} \cdot \sum_{\boldsymbol{s} \in \mathcal{T}} \prod_{i=1}^{m} \binom{d_i}{s_i} p_i^{s_i} (1-p_i)^{d_i - s_i} \\
&= \left(1 - \ell^{-2\log \ell}\right) \cdot E \cdot \underset{\chi \sim \Omega}{\mathbb{P}} \left[\chi \in \mathcal{T}\right] \\
&\geq \left(1 - \ell^{-2\log \ell}\right) \left(1 - \ell^{-(\log \ell)/8}\right) E \\
&\geq \left(1 - 2\ell^{-(\log \ell)/8}\right) E.
\end{aligned}$$

For the other direction, we derive for such g

$$\begin{aligned}
\sum_{\boldsymbol{s} \in \mathcal{S}} B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1-p_i)^{d_i - s_i} &= \left(\sum_{\boldsymbol{s} \in \mathcal{T}} + \sum_{\boldsymbol{s} \in \mathcal{N}}\right) B_g(\boldsymbol{s}, \boldsymbol{y}) \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1-p_i)^{d_i - s_i} \\
&\overset{(3.38)}{\leq} \sum_{\boldsymbol{s} \in \mathcal{T}} \left(1 + \ell^{-2\log \ell}\right) \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})] \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1-p_i)^{d_i - s_i} + E \cdot \ell^{-(\log \ell)/8} \\
&\leq \left(1 + \ell^{-2\log \ell}\right) \underbrace{\sum_{\boldsymbol{s} \in \mathcal{S}} \mathbb{E}[B_g(\boldsymbol{s}, \boldsymbol{y})] \prod_{i=1}^{m} q_i^{d_i} p_i^{s_i} (1-p_i)^{d_i - s_i}}_{E} + E \cdot \ell^{-(\log \ell)/8} \\
&= \left(1 + \ell^{-2\log \ell} + \ell^{-(\log \ell)/8}\right) E \\
&\leq \left(1 + 2\ell^{-(\log \ell)/8}\right) E.
\end{aligned}$$

□

3.3.4 Concentration of entropy

We can now get a tight concentration for $\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y}]$ using the relation (3.29). We already showed that the sum in RHS of (3.29) is tightly concentrated around its expectation, so it only remains to show that $\mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{v} = \boldsymbol{0}]$ is tiny compared to E. Here we will use that we picked $\boldsymbol{y}$ to be "typical" from the start so that (3.26) and (3.27) hold, and that we consider here the above-capacity regime. Recall (3.24), as well the the conditions (3.26) and (3.27) on $\boldsymbol{y}$ being typical. We derive

$$\begin{aligned}
\mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} | \boldsymbol{V} = \boldsymbol{0}] = \prod_{i=1}^{m} q_i^{d_i} p_i^{t_i} (1-p_i)^{d_i - t_i} &= \prod_{i=1}^{m} \left[q_i^{d_i} \cdot p_i^{d_i p_i} (1-p_i)^{d_i(1-p_i)} \cdot \left(\frac{1-p_i}{p_i}\right)^{d_i p_i - t_i} \right] \\
&= \prod_{i=1}^{m} q_i^{d_i} \cdot \prod_{i=1}^{m} 2^{-d_i h(p_i)} \cdot \prod_{i=1}^{m} 2^{(d_i p_i - t_i) \log\left(\frac{1-p_i}{p_i}\right)}
\end{aligned}$$

$$\begin{aligned}
&= \prod_{i=1}^{m} q_i^{d_i} \cdot 2^{\sum_{i=1}^{m}(-\ell q_i h(p_i)+(\ell q_i - d_i)h(p_i))} \cdot 2^{\sum_{i=1}^{m}(d_i p_i - t_i)\log\left(\frac{1-p_i}{p_i}\right)} \\
&\overset{(3.26),(3.27)}{\leq} \prod_{i=1}^{m} q_i^{d_i} \cdot 2^{-\ell H(W)+2\ell^{1/2}\log\ell+3\ell^{1/2}\log^2\ell} \\
&\leq \prod_{i=1}^{m} q_i^{d_i} \cdot \frac{2^k-1}{2^\ell} \cdot \ell^{-\log\ell} \;=\; E\cdot\ell^{-\log\ell}, \qquad (3.39)
\end{aligned}$$

where the last inequality follows from $k \geq \ell(1-H(W)) + 13\ell^{1/2}\log^3\ell$.

Now, combining this with Lemma 3.12, we obtain a concentration for (3.29):

Corollary 3.13. *With probability at least* $1 - 2\ell^{-(\log\ell)/8}$ *over the choice of kernel* $g \sim G$ *and for any typical* $\boldsymbol{y}$

$$\left|2^k \cdot \mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}] - E\right| \leq 3\ell^{-(\log\ell)/8}\cdot E,$$

where $E = \dfrac{2^k-1}{2^\ell}\prod_{i=1}^{m} q_i^{d_i}$.

Next, completely analogously we derive the concentration for $\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}|V_1=0]$, which is the numerator inside the entropy in (3.28). The only thing that changes is that we will have dimension $k-1$ instead of k for this case. We can state

Corollary 3.13′. *With probability at least* $1 - 2\ell^{-(\log\ell)/8}$ *over the choice of kernel* $g \sim G$ *and for any typical* $\boldsymbol{y}$

$$\left|2^k \cdot \mathbb{P}^{(g)}[V_1 = 0, \boldsymbol{Y}=\boldsymbol{y}] - \widetilde{E}\right| \leq 3\ell^{-(\log\ell)/8}\cdot \widetilde{E},$$

where $\widetilde{E} = \dfrac{2^{k-1}-1}{2^\ell}\prod_{i=1}^{m} q_i^{d_i}$.

Combining these two together and skipping the simple math, identical to that of the BSC case in (3.20)–(3.22), we derive

Corollary 3.14. *With probability at least* $1 - 4\ell^{-(\log\ell)/8}$ *over the choice of kernel* $g \sim G$ *and for any typical* $\boldsymbol{y}$*,*

$$\left|\frac{\mathbb{P}^{(g)}[V_1=0,\boldsymbol{Y}=\boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}]} - \frac{1}{2}\right| \leq \ell^{-(\log\ell)/9}.$$

Since $h(1/2+x) \geq 1-4x^2$ for any $x \in [-1/2, 1/2]$ ([Top01, Theorem 1.2]), we then derive for a typical $\boldsymbol{y}$:

$$\begin{aligned}
\mathop{\mathbb{E}}_{g}\left[H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})\right] = \mathop{\mathbb{E}}_{g}\left[h\left(\frac{\mathbb{P}^{(g)}[V_1=0,\boldsymbol{Y}=\boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}]}\right)\right] &\geq (1-4\ell^{-(\log\ell)/8})\cdot(1-4\ell^{-(\log\ell)/9}) \\
&\geq 1 - 8\ell^{-(\log\ell)/9}.
\end{aligned}$$

Then in (3.25) we have

$$\begin{aligned}\mathop{\mathbb{E}}_{g}\left[H^{(g)}(V_1|\boldsymbol{Y})\right] &= \sum_{\boldsymbol{y}\in\mathcal{Y}^\ell}\mathbb{P}[\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{V}=\boldsymbol{0}]\mathop{\mathbb{E}}_{g}\left[H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})\right]\\ &\geq \sum_{\boldsymbol{y}\text{ typical}}\mathbb{P}[\boldsymbol{Y}=\boldsymbol{y}|\boldsymbol{V}=\boldsymbol{0}]\mathop{\mathbb{E}}_{g}\left[H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})\right]\\ &\geq (1-\ell^{-\log\ell})\cdot(1-8\ell^{-(\log\ell)/9}).\\ &\geq 1-9\ell^{-(\log\ell)/9}\geq 1-\ell^{-(\log\ell)/10},\end{aligned}\tag{3.40}$$

where we used that the probability to get a typical output on a zero input is at least $1-\ell^{-\log\ell}$ by Lemma 3.7, as well as the condition $\log\ell\geq 20$.

Finally, using the fact that $H^{(g)}(V_1|\boldsymbol{Y})\leq 1$, Markov's inequality, and (3.40), we get

$$\begin{aligned}\mathop{\mathbb{P}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\leq 1-\ell^{-\frac{\log\ell}{20}}\right] &= \mathbb{P}\left[1-H^{(g)}(V_1|\boldsymbol{Y})\geq \ell^{-\frac{\log\ell}{20}}\right]\\ &\leq \frac{\mathbb{E}\left[1-H^{(g)}(V_1|\boldsymbol{Y})\right]}{\ell^{-(\log\ell)/20}}\leq \ell^{-(\log\ell)/20}.\end{aligned}$$

This completes the proof of Theorem 3.1 for the case of BMS channel with bounded output alphabet size, assuming the typicality Lemma 3.7 and concentration Lemma 3.17 which we used in Lemma 3.8. We now proceed to prove these.

3.3.5 Proof that the typical set is indeed typical

Proof of Lemma 3.7. We start with proving that (3.26) is satisfied with high probability (over the randomness of the channel). Notice that $(d_1, d_2, \dots, d_m)$ are multinomially distributed by construction, since for each of the ℓ bits transitioned, we choose independently the subchannel $W^{(i)}$ to use with probability q_i, for $i = 1, 2, \dots, m$, and d_i represents the number of times the channel $W^{(i)}$ was chosen. So indeed $(d_1, d_2, \dots, d_m) \sim \text{Mult}(\ell, q_1, q_2, \dots, q_m)$. The crucial property of multinomial random variables we are going to use is *negative association* ([JDP83], [DR96]). The (simplified version of the) fact we are going to use about negatively associated random variables can be formulated as follows:

Lemma 3.15 ([JDP83], Property P$_2$)**.** *Let $X_1, X_2, \dots, X_m$ be negatively associated random variables. Then for every set of m positive monotone non-decreasing functions $f_1, \dots, f_m$,*

$$\mathbb{E}\left[\prod_{i=1}^m f_i(X_i)\right]\leq\prod_{i=1}^m\mathbb{E}[f_i(X_i)].$$

We also use the fact that since $(d_1, d_2, \dots, d_m)$ are negatively associated, then after applying decreasing functions $g_i(x) = \ell q_i - x$ coordinate-wise to these random variables we also obtain negatively associated random variables ([DR96], Proposition 7). In other words, we argue that $(\ell q_1 - d_1, \ell q_2 - d_2, \dots, \ell q_m - d_m)$ are also negatively associated, thus we can apply Lemma 3.15 to these random variables.

Let us now denote for convenience $\alpha_i = h(p_i)$ for $i = 1, 2, \dots, m$, and so $0 \le \alpha_i \le 1$. Let also $X = \sum_{i=1}^m (\ell \cdot q_i - d_i)\alpha_i$, and we now can start with simple exponentiation and Markov's inequality: for any a and any $t > 0$

$$\mathbb{P}[X \ge a] = \mathbb{P}[e^{tX} \ge e^{ta}] \le e^{-ta}\,\mathbb{E}\left[e^{tX}\right] = e^{-ta}\,\mathbb{E}\left[\prod_{i=1}^m e^{t\cdot\alpha_i(\ell q_i - d_i)}\right] \le e^{-ta}\prod_{i=1}^m \mathbb{E}\left[e^{t\cdot\alpha_i(\ell q_i - d_i)}\right], \tag{3.41}$$

where in the last inequality Lemma 3.15 is applied for negatively associated random variables $(\ell q_1 - d_1, \ell q_2 - d_2, \dots, \ell q_m - d_m)$, as discussed above, and positive non-decreasing functions $f_i(x) = e^{t\cdot\alpha_i\cdot x}$, since $\alpha_i, t \ge 0$.

Next, consider the following claim, which follows from standard Chernoff-type arguments:

Claim 3.16. *Let $Z \sim$ Binom(n,p), and let $b > 0$. Then $\mathbb{E}[e^{-b\cdot Z}] \le e^{np\cdot(e^{-b}-1)}$.*

Proof. We can write $Z = \sum\limits_{j=1}^n Z_j$, where $Z_j \sim$ Bern(p) are independent Bernoulli random variables. Then

$$\mathbb{E}\left[e^{-b\cdot Z}\right] = \mathbb{E}\left[\prod_{j=1}^n e^{-b\cdot Z_j}\right] = \prod_{j=1}^n \mathbb{E}\left[e^{-b\cdot Z_j}\right] = \left((1-p) + p\cdot e^{-b}\right)^n \le e^{np(e^{-b}-1)}, \tag{3.42}$$

where the only inequality uses the fact that $1 + x \le e^x$ for any x. □

Turning back to (3.41), we are going to bound the terms $\mathbb{E}\left[e^{t\cdot\alpha_i(\ell q_i - d_i)}\right]$ individually. It is clear that the marginal distribution of d_i is just Binom(ℓ, q_i), so we are able to use Claim 3.16 for it. Derive:

$$\begin{aligned}\mathbb{E}\left[e^{t\cdot\alpha_i(\ell q_i - d_i)}\right] = e^{t\alpha_i\ell q_i}\cdot\mathbb{E}\left[e^{-t\alpha_i\cdot d_i}\right] &\overset{(3.42)}{\le} e^{t\cdot\alpha_i\ell q_i}\cdot e^{\ell q_i\left(e^{-t\alpha_i}-1\right)} \\ &= e^{\ell q_i\left(t\alpha_i + e^{-t\alpha_i}-1\right)} \le e^{\ell q_i\left(t+e^{-t}-1\right)},\end{aligned} \tag{3.43}$$

where the last inequality uses that $x + e^{-x}$ is increasing for $x \ge 0$ together with $0 \le t\alpha_i \le t$, as $t > 0$ and $0 \le \alpha_i \le 1$. Plugging the above into (3.41) and using $\sum_{i=1}^m q_i = 1$, we obtain

$$\mathbb{P}[X \ge a] \le e^{-ta}\prod_{i=1}^m e^{\ell q_i\left(t+e^{-t}-1\right)} = e^{-ta}\cdot e^{\ell\left(t+e^{-t}-1\right)} \le e^{-ta+\ell\frac{t^2}{2}}, \tag{3.44}$$

where we use $x + e^{-x} - 1 \le \frac{x^2}{2}$ for any $x \ge 0$. Finally, by taking $a = 2\sqrt{\ell}\log\ell$, setting $t = a/\ell$, and recalling what we denoted by X and α_i above, we immediately deduce

$$\mathbb{P}\left[\sum_{i=1}^m (\ell\cdot q_i - d_i)h(p_i) \ge 2\sqrt{\ell}\log\ell\right] \le e^{-\frac{a^2}{2\ell}} = e^{-2\log^2\ell} \le \ell^{-2\log\ell}.$$

This means that the first typicality requirement (3.26) holds with very high probability (over the randomness of the channel).

Let us now prove that the second typicality condition (3.27) holds with high probability. For that, we condition on the values of $d_1, d_2, \dots, d_m$. We will see that (3.27) holds with high probability for any fixed values of $d_1, d_2, \dots, d_m$, and then it is clear that is will imply that it also holds with high probability overall.

So, fix the values of $d_1, d_2, \dots, d_m$. Denote $Y = \sum_{i=1}^m (p_i d_i - t_i) \log\left(\frac{1-p_i}{p_i}\right)$, and then our goal it to show that Y is bounded above by $O(\sqrt{\ell}\log^2 \ell)$ with high probability (over the randomness of t_i's). Given the conditioning on $d_1, d_2, \dots, d_m$, it is clear that $t_i \sim \mathrm{Binom}(d_i, p_i)$ for all $i = 1, 2, \dots, m$, and they are all independent (recall that d_i corresponds to the number of times subchannel $W^{(i)}$ is chosen, while t_i corresponds to the number of "flips" within this subchannel).

We split the summation in Y into two parts: let $T_1 = \{i : p_i \le \frac{1}{\ell}\}$ and $T_2 = [m] \setminus T_1$. Then for any realization of t_i's, we have $\sum\limits_{i \in T_1} (p_i d_i - t_i) \log\left(\frac{1-p_i}{p_i}\right) \le \sum\limits_{i \in T_1} p_i d_i \log\left(\frac{1}{p_i}\right) \le \sum\limits_{i \in T_1} \frac{d_i \log \ell}{\ell} \le \log \ell$.

Denote the second part of the summation as $Y_2 = \sum_{i \in T_2} (p_i d_i - t_i) \log\left(\frac{1-p_i}{p_i}\right)$. Notice that $\log\left(\frac{1-p_i}{p_i}\right) \le \log\left(\frac{1}{p_i}\right) \le \log \ell$ for $i \in T_2$. Denote then $\gamma_i = \log\left(\frac{1-p_i}{p_i}\right) / \log \ell$, and so $0 \le \gamma_i \le 1$ for $i \in T_2$. Finally, let $\widetilde{Y_2} = Y_2 / \log \ell = \sum_{i \in T_2} (p_i d_i - t_i) \cdot \gamma_i$.

We now prove the concentration on $\widetilde{Y_2}$ in almost exactly the same way as we did for X above. Similarly to (3.41) obtain

$$\begin{aligned} \mathbb{P}\left[\widetilde{Y_2} > a\right] = \mathbb{P}\left[e^{t\widetilde{Y_2}} > e^{ta}\right] \le e^{-ta}\, \mathbb{E}\left[e^{t\widetilde{Y_2}}\right] &= e^{-ta} \cdot \mathbb{E}\left[\prod_{i=1}^m e^{t \cdot \gamma_i \cdot (p_i d_i - t_i)}\right] \\ &= e^{-ta} \cdot \prod_{i=1}^m \mathbb{E}\left[e^{t \cdot \gamma_i \cdot (p_i d_i - t_i)}\right], \end{aligned} \tag{3.45}$$

where the last equality holds because we conditioned on $d_1, d_2, \dots, d_m$, and so $t_1, t_2, \dots, t_m$ are independent, as discussed above. Next, Claim 3.16 applied for $t_i \sim \mathrm{Binom}(d_i, p_i)$ and $t \cdot \gamma_i > 0$ for any $t > 0$ gives $\mathbb{E}\left[e^{-t\gamma_i \cdot t_i}\right] \le e^{d_i p_i (e^{-t\gamma_i} - 1)}$, and so similarly to (3.42)–(3.44) derive from (3.45)

$$\begin{aligned} \mathbb{P}\left[\widetilde{Y_2} > a\right] \le e^{-ta} \cdot \prod_{i \in T_2} e^{p_i d_i \left(t\gamma_i + e^{-t\gamma_i} - 1\right)} &\le e^{-ta} \cdot \prod_{i \in T_2} e^{p_i d_i \left(t + e^{-t} - 1\right)} \\ &\le e^{-ta + \sum_{i \in T_2} p_i d_i \cdot t^2/2} \le e^{-ta + \ell t^2/2} \end{aligned}$$

for any $t > 0$, where we used $0 \le \gamma_i \le 1$ for $i \in T_2$, $p_i < 1$, and $\sum_{i \in T_2} d_i \le \ell$. Therefore, by taking again $a = 2\sqrt{\ell}\log \ell$ and $t = a/\ell$, obtain

$$\mathbb{P}\left[Y_2 \ge 2\sqrt{\ell}\log^2 \ell\right] = \mathbb{P}\left[\widetilde{Y_2} \ge 2\sqrt{\ell}\log \ell\right] \le \ell^{-2\log \ell}.$$

Since $Y \le \log \ell + Y_2$, we conclude that $Y \le 3\sqrt{\ell}\log^2 \ell$ with probability at least $1 - \ell^{-2\log \ell}$ over the randomness of the channel.

Since both (3.26) and (3.27) hold with probability at least $1-\ell^{-2\log\ell}$, the union bound implies that these two conditions hold simultaneously with probability at least $1-2\ell^{-2\log\ell}\geq 1-\ell^{-\log\ell}$. □

3.3.6 Concentration lemma

Lemma 3.17. *Let $\chi\sim\Omega=Binom(d_1,p_1)\times Binom(d_2,p_2)\times\cdots\times Binom(d_m,p_m)$, where d_i's are nonnegative integers for $i\in[m]$, $p_i\leq 1/2$, $\sum_{i=1}^m d_i=\ell$, and $m\leq\sqrt{\ell}$. Let also ℓ be large so that* $\log\ell\geq 8$. *Then the following holds with probability at least* $1-\ell^{-(\log\ell)/4}$*:*

$$\sum_{i=1}^m d_i\left(h(p_i)-h\left(\frac{\chi_i}{d_i}\right)\right)\leq 7\ell^{1/2}\log^3\ell. \tag{3.46}$$

Proof. First, notice that we can disregard all the indices i for which $d_i=0$, as they do not contribute anything to the LHS of (3.46). So from now on, we assume for simplicity that $d_i\geq 1$ for all $i=1,2,\ldots,m$.

Next, we split the interval $[1:m]$ into two parts. In the first part the value of $d_i\cdot p_i$ is going to be small, and the sum of $d_ih(p_i)$ will also be small and can be upper bounded. For the second part, when $d_i\cdot p_i$ is large enough, we will be able to apply some concentration arguments. Denote:

$$F_1 \coloneqq \left\{i\ :\ p_i\leq\frac{4\log^2\ell}{d_i}\right\},$$
$$F_2 \coloneqq \{1,2,\ldots,m\}\setminus F_1.$$

Then

$$\sum_{i=1}^m d_i\left(h(p_i)-h\left(\frac{\chi_i}{d_i}\right)\right)\leq\sum_{i\in F_1}d_ih(p_i)+\sum_{i\in F_2}d_i\left(h(p_i)-h\left(\frac{\chi_i}{d_i}\right)\right). \tag{3.47}$$

Let us deal with the summation over F_1 first. Split this set even further: $F_1^{(1)}=\{i\in F_1\ :\ d_i\geq 8\log^2\ell\}$, and $F_1^{(2)}=F_1\setminus F_1^{(1)}$. Then for any $i\in F_1^{(1)}$ we use $h(p_i)\leq 2p_i\log\frac{1}{p_i}$ from Proposition 2.7, since $p_i\leq 1/2$. For any $i\in F_1^{(2)}$ we just use $h(p_i)\leq 1$. Combining these, obtain

$$\begin{aligned}\sum_{i\in F_1}d_ih(p_i)\leq\sum_{i\in F_1^{(1)}}2d_ip_i\log\frac{1}{p_i}+\sum_{i\in F_1^{(2)}}d_i\leq\sum_{i\in F_1^{(1)}}8\log^2\ell\cdot\log\left(\frac{d_i}{4\log^2\ell}\right)+\left|F_1^{(2)}\right|\cdot 8\log^2\ell\\ \leq 8\log^2\ell\cdot\sum_{i\in F_1^{(1)}}\log d_i+\left|F_1^{(2)}\right|\cdot 8\log^2\ell. \end{aligned}\tag{3.48}$$

For the second summand in the RHS above, we will just use $\left|F_1^{(2)}\right|\leq m\leq\ell^{1/2}$. For the first summand, we use Jensen's inequality, the fact that $\sum\limits_{i=1}^m d_i=\ell$, and $\left|F_1^{(1)}\right|\leq m\leq\ell^{1/2}$

to derive

$$\sum_{i\in F_1^{(1)}} \log d_i \le \left|F_1^{(1)}\right| \cdot \log\left(\frac{\sum_{i\in F_1^{(1)}} d_i}{\left|F_1^{(1)}\right|}\right) \le \left|F_1^{(1)}\right| \cdot \log\left(\frac{\ell}{\left|F_1^{(1)}\right|}\right) \le \ell^{1/2}\log\left(\ell^{1/2}\right) = \frac{1}{2}\ell^{1/2}\log\ell,$$

where the last inequality uses that $x\log(\ell/x)$ is increasing while $x \le \ell/e$. Therefore, in (3.48) obtain

$$\sum_{i\in F_1} d_i h(p_i) \le 8\log^2\ell \cdot \sum_{i\in F_1^{(1)}} \log d_i + \left|F_1^{(2)}\right| \cdot 8\log^2\ell \le 5\ell^{1/2}\log^3\ell, \tag{3.49}$$

where we also used $8 \le \log\ell$.

Therefore, the first part of the RHS of (3.47) is always bounded by $5\ell^{1/2}\log^3\ell$. We will now deal with the remaining summations over $i \in F_2$.

For any $i \in F_2$, by definition $d_ip_i \ge 4\log^2\ell$. Now apply the multiplicative Chernoff bound (3.9) for $\chi_i \sim \text{Binom}(d_i, p_i)$ and $\delta = \frac{\log\ell}{\sqrt{d_ip_i}}$ to get

$$\mathop{\mathbb{P}}_{\chi_i}\left[|\chi_i - d_ip_i| \ge \sqrt{d_ip_i}\log\ell\right] \le 2e^{-\log^2\ell/3} \le \ell^{-(\log\ell)/3} \qquad \text{if } \log\ell \le \sqrt{d_ip_i}, \tag{3.50}$$

where the last inequality holds for $\log\ell > 3$ because the log in the exponent is to base 2. The condition $\log\ell \le \sqrt{d_ip_i}$ is required in order to have $\delta \le 1$ for the multiplicative Chernoff bound (3.9) to be applicable, and it is satisfied for $i \in F_2$.

Then, by the union bound, we derive

$$\mathop{\mathbb{P}}_{\chi\sim\Omega}\left[|\chi_i - d_ip_i| \ge \sqrt{d_ip_i}\log\ell \text{ for some } i \in F_2\right] \le |F_2| \cdot \ell^{-(\log\ell)/3} \le \ell^{-(\log\ell)/3+1/2}. \tag{3.51}$$

Define the sets $\mathcal{T}_1^{(i)}$ for all $i = 1, 2, \dots, m$ as follows:

$$\begin{aligned} \mathcal{T}_1^{(i)} &:= \left\{s_i \in [0:d_i] \;:\; |s_i - d_ip_i| \le \sqrt{d_ip_i}\log\ell\right\}, && \text{for } i \in F_2; \\ \mathcal{T}_1^{(i)} &:= [0:d_i], && \text{for } i \notin F_2, \end{aligned} \tag{3.52}$$

and let

$$\theta_i := \mathbb{P}[\chi_i \in \mathcal{T}_1^{(i)}]. \tag{3.53}$$

Then by (3.50) we have

$$\begin{aligned} \theta_i &\ge 1 - \ell^{-(\log\ell)/3}, && \text{for } i \in F_2; \\ \theta_i &= 1, && \text{for } i \notin F_2. \end{aligned}$$

Finally, define

$$\theta := \prod_{i=1}^m \theta_i = \prod_{i\in F_2} \theta_i = \prod_{i\in F_2} \mathbb{P}[\chi_i \in \mathcal{T}_1^{(i)}] = \mathop{\mathbb{P}}_{\chi\sim\Omega}[\chi_i \in \mathcal{T}_1^{(i)} \text{ for all } i \in F_2] \ge 1 - \ell^{-(\log\ell)/3+1/2}, \tag{3.54}$$

where the last inequality is a direct implication of (3.51).

We will now define a set of new probability distributions $\mathcal{D}_i$ for all $i = 1, 2, \ldots, m$, as binomial distributions $\mathrm{Binom}(d_i, p_i)$ restricted to intervals $\mathcal{T}_1^{(i)}$. Formally, let us write

$$\mathop{\mathbb{P}}_{\eta_i \sim \mathcal{D}_i}\left[\eta_i = x\right] = \begin{cases} 0, & \text{if } x \notin \mathcal{T}_1^{(i)}; \\ \mathbb{P}_{\chi_i \sim \mathrm{Binom}(d_i, p_i)}\left[\chi_i = x\right] \cdot \theta_i^{-1}, & \text{if } x \in \mathcal{T}_1^{(i)}. \end{cases} \tag{3.55}$$

(So to get $\mathcal{D}_i$ we just took a distribution $\mathrm{Binom}(d_i, p_i)$, truncated it so it does not have any mass outside of $\mathcal{T}_1^{(i)}$, and rescaled appropriately.)

Next, define a product distribution $\mathcal{D} \coloneqq \bigtimes_{i=1}^m \mathcal{D}_i$ on the set $\mathcal{T}_1 \coloneqq \bigtimes_{i=1}^m \mathcal{T}_1^{(i)}$. Notice now that it is trivial that for any subset $\mathcal{R} \subseteq \mathcal{T}_1$ it holds

$$\mathop{\mathbb{P}}_{\chi \sim \Omega}[\chi \in \mathcal{R}] = \mathop{\mathbb{P}}_{\eta \sim \mathcal{D}}[\eta \in \mathcal{R}] \cdot \theta. \tag{3.56}$$

Since θ is very close to 1, it suffices to prove the claims for $\mathcal{D}$ instead of Ω.

Recall that our goal was to show that $\sum_{i \in F_2} d_i \left(h(p_i) - h\left(\frac{\chi_i}{d_i}\right)\right)$ (the second part from (3.47)) is bounded above by $O(\ell^{1/2} \log^3 \ell)$ with high probability, when $\chi \sim \Omega$. Instead now let us show that this summation is small with high probability when $\chi \sim \mathcal{D}$, and then use the arguments above to see that there is not much of a difference when $\chi \sim \Omega$.

Claim 3.18. *Let $i \in F_2$ and $\chi_i \sim \mathcal{D}_i$. Then*

$$\left| d_i \left(h(p_i) - h\left(\frac{\chi_i}{d_i}\right)\right)\right| \le \sqrt{d_i p_i} \log^2 \ell. \tag{3.57}$$

Proof. First, $\left|\frac{\chi_i}{d_i} - p_i\right| \le \sqrt{\frac{p_i}{d_i}} \log \ell$ for $\chi_i \sim \mathcal{D}_i$ by definition of the distribution $\mathcal{D}_i$. Now, for $i \in F_2$, $p_i \ge \frac{4 \log^2 \ell}{d_i}$, from which it follows that $\frac{p_i}{2} \ge \sqrt{\frac{p_i}{d_i}} \log \ell$, and therefore $\frac{p_i}{2} \le \frac{\chi_i}{d_i} \le \frac{3p_i}{2}$. We then use the concavity of the binary entropy function on $[0, 1]$. For a concave differentiable function f on an interval $[a, b]$, one has $|f(b) - f(a)| \le |b - a| \cdot \max\{|f'(a)|, |f'(b)|\}$, which follows from a standard inequality $f(y) \le f(x) + f'(x)(y - x)$ applied for (a, b) or (b, a), depending on which of $f(a)$ and $f(b)$ is larger. We apply this for the binary entropy function $h(\cdot)$ and one of the intervals $\left[\frac{\chi_i}{d_i}, p_i\right]$ and $\left[p_i, \frac{\chi_i}{d_i}\right]$, depending on which of $\frac{\chi_i}{d_i}$ and p_i is smaller:

$$\left| h\left(\frac{\chi_i}{d_i}\right) - h(p_i)\right| \le \left|\frac{\chi_i}{d_i} - p_i\right| \cdot \max\left\{ \left|\frac{dh}{dx}(p_i)\right|, \left|\frac{dh}{dx}\left(\frac{\chi_i}{d_i}\right)\right| \right\}.$$

Now, both p_i and $\frac{\chi_i}{d_i}$ lie in the interval $\left[\frac{p_i}{2}, \frac{3p_i}{2}\right]$, which is contained in $\left[\frac{p_i}{2}, 1 - \frac{p_i}{2}\right]$, as $p_i < 1/2$. Out of symmetry of h around $1/2$, it follows that the maximal value of $\left|\frac{dh}{dx}(\cdot)\right|$ on the interval $\left[\frac{p_i}{2}, 1 - \frac{p_i}{2}\right]$ is attained at $\frac{p_i}{2}$. Therefore, we have

$$\begin{aligned}\left|h\left(\frac{\chi_i}{d_i}\right)-h(p_i)\right| &\le \left|\frac{\chi_i}{d_i}-p_i\right|\cdot\max\left\{\left|\frac{dh}{dx}(p_i)\right|,\left|\frac{dh}{dx}\left(\frac{\chi_i}{d_i}\right)\right|\right\}\\ &\le \sqrt{\frac{p_i}{d_i}}\log\ell\cdot\left|\frac{dh}{dx}\left(\frac{p_i}{2}\right)\right| = \sqrt{\frac{p_i}{d_i}}\log\ell\cdot\log\frac{1-p_i/2}{p_i/2}\\ &\le \sqrt{\frac{p_i}{d_i}}\log\ell\cdot\log\frac{2}{p_i} \le \sqrt{\frac{p_i}{d_i}}\log\ell\cdot\log\left(\frac{d_i}{2\log^2\ell}\right) \le \sqrt{\frac{p_i}{d_i}}\log^2\ell,\end{aligned}$$

where the penultimate inequality follows from $p_i \ge \frac{4\log^2\ell}{d_i}$ for $i\in F_2$, and the last inequality uses $\frac{d_i}{2\log^2\ell}\le\ell$, as $\sum_{i=1}^m d_i=\ell$ and d_i's are nonnegative. Therefore, (3.57) follows. □

Let $\chi\sim\mathcal{D}$ here and further. Denote for convenience new random variables $X_i = d_i\left(h(p_i)-h\left(\frac{\chi_i}{d_i}\right)\right)$ for all $i\in F_2$, and let also $X=\sum_{i\in F_2}X_i=\sum_{i\in F_2}d_i\left(h(p_i)-h\left(\frac{\chi_i}{d_i}\right)\right)$.

Claim 3.19. *With probability at least* $1-\ell^{-\log\ell}$,

$$X-\mathbb{E}[X]\le\ell^{1/2}\log^3\ell.$$

Proof. Obviously all the X_i's are independent, and also $X_i\in\left[-\sqrt{d_ip_i}\log^2\ell,\sqrt{d_ip_i}\log^2\ell\right]$ by Claim 3.18. Then we can apply Hoeffding's inequality for the sum of bounded independent random variables ([Hoe63, Theorem 2]), and obtain

$$\begin{aligned}\underset{\chi\sim\mathcal{D}}{\mathbb{P}}\left[X-\mathbb{E}[X]\ge\ell^{1/2}\log^3\ell\right] &\le \exp\left(-\frac{2\ell\log^6\ell}{\sum\limits_{i\in F_2}(2\sqrt{d_ip_i}\log^2\ell)^2}\right)\\ &= \exp\left(-\frac{2\ell\log^6\ell}{\log^4\ell\cdot\sum\limits_{i\in F_2}(4d_ip_i)}\right)\le\exp\left(-\frac{\ell\log^2\ell}{\sum\limits_{i\in F_2}d_i}\right)\\ &\le e^{-\log^2\ell}\le\ell^{-\log\ell},\end{aligned}$$

where we use $p_i\le 1/2$ and $\sum\limits_{i\in F_2}d_i\le\sum\limits_{i=1}^m d_i=\ell$ in the second and third inequalities, respectively. □

So by now we proved that $X=\sum_{i\in F_2}d_i\left(h(p_i)-h\left(\frac{\chi_i}{d_i}\right)\right)$ does not deviate much from its expectation, when $\chi\sim\mathcal{D}$. What we are left to show now is that $\mathbb{E}[X]$ is not very large by itself.

The following two claims show that the first moment and mean absolute deviation of the distribution $\mathcal{D}_i$ are close to those of Ω_i, for $i\in F_2$. This easily follows from the definition (3.55) of $\mathcal{D}_i$, and the proofs are deferred to Section 3.5.2 at the end of this chapter.

Claim 3.20. *Let $i \in F_2$. Then* $\left|\underset{\chi_i \sim \mathcal{D}_i}{\mathbb{E}}\left[\frac{\chi_i}{d_i}\right] - p_i\right| \leq \frac{1}{d_i}$.

Claim 3.21. *Let $\chi_i \sim \mathcal{D}_i$ and $\eta_i \sim \Omega_i$ for $i \in F_2$. Then* $\mathbb{E}\left|\chi_i - \mathbb{E}[\chi_i]\right| \leq \mathbb{E}\left|\eta_i - \mathbb{E}\left[\eta_i\right]\right| + 1$.

These observations allow us we prove the following

Claim 3.22. *Let $i \in F_2$, and $\chi_i \sim \mathcal{D}_i$. Then* $h\left(\mathbb{E}\left[\frac{\chi_i}{d_i}\right]\right) - \mathbb{E}\left[h\left(\frac{\chi_i}{d_i}\right)\right] \leq \frac{5\log \ell}{d_i}$.

Proof. Unfortunately, Jensen's inequality works in the opposite direction for us here. However, we use some form of converse Jensen's from [Dra11], which says the following:

Lemma 3.23 (Converse Jensen's inequality, [Dra11], Corollary 1.8)**.** *Let f be a concave differentiable function on an interval $[a, b]$, and let Z be a (discrete) random variable, taking values in $[a, b]$. Then*

$$0 \leq f(\mathbb{E}[Z]) - \mathbb{E}[f(Z)] \leq \frac{1}{2}\left(f'(a) - f'(b)\right) \cdot \mathbb{E}\left|Z - \mathbb{E}[Z]\right|.$$

We apply it here for the concave binary entropy function h and random variable $Z = \frac{\chi_i}{d_i}$ for $\chi_i \sim \mathcal{D}_i$, which takes values in $[a, b] := \left[p_i - \sqrt{\frac{p_i}{d_i}}\log \ell, p_i + \sqrt{\frac{p_i}{d_i}}\log \ell\right]$. Recall also that for $i \in F_2$, $p_i \geq \frac{4\log^2 \ell}{d_i}$ and then $\frac{p_i}{2} \geq \sqrt{\frac{p_i}{d_i}}\log \ell$, therefore $a = p_i - \sqrt{\frac{p_i}{d_i}}\log \ell \geq \frac{p_i}{2}$, and also $b = p_i + \sqrt{\frac{p_i}{d_i}}\log \ell \leq \frac{3p_i}{2}$. Using the mean value theorem, for some $c \in [a, b] \subseteq \left[\frac{p_i}{2}, \frac{3p_i}{2}\right]$ we have

$$h'(a) - h'(b) = (b - a) \cdot (-h''(c)) \leq 2\sqrt{\frac{p_i}{d_i}}\log \ell \cdot (-h''(c)).$$

Now we look at $(-h''(c)) = \frac{1}{c(1-c)\ln 2}$ for some $c \in \left[\frac{p_i}{2}, \frac{3p_i}{2}\right]$. As $p_i < 1/2$, it follows $\left[\frac{p_i}{2}, \frac{3p_i}{2}\right] \subseteq \left[\frac{p_i}{2}, 1 - \frac{p_i}{2}\right]$. Using the symmetry of a function $x(1-x)$ around $1/2$, we conclude that its minimal value over the interval $\left[\frac{p_i}{2}, \frac{3p_i}{2}\right]$ is attained at $p_i/2$. Thus derive $c(1-c) \geq \frac{p_i}{2}\left(1 - \frac{p_i}{2}\right) \geq \frac{3p_i}{8}$, since $p_i < 1/2$. And so $(-h''(c)) = \frac{1}{c(1-c)\ln 2} \leq \frac{8}{p_i \cdot 3\ln 2} \leq \frac{4}{p_i}$. Therefore

$$h'(a) - h'(b) \leq \frac{8\log \ell}{\sqrt{d_i p_i}}.$$

Finally, Claim 3.21 gives $\mathbb{E}\left|Z - \mathbb{E}[Z]\right| \leq \mathbb{E}\left|\frac{Z_2}{d_i} - \mathbb{E}\left[\frac{Z_2}{d_i}\right]\right| + \frac{1}{d_i}$ for $Z_2 \sim \mathrm{Binom}(d_i, p_i)$, thus

$$\begin{aligned}\mathbb{E}\left|Z - \mathbb{E}[Z]\right| \leq \frac{1}{d_i}\mathbb{E}\left|Z_2 - \mathbb{E}[Z_2]\right| + \frac{1}{d_i} &\leq \frac{1}{d_i}\sqrt{\mathbb{E}[(Z_2 - \mathbb{E}[Z_2])^2]} + \frac{1}{d_i} \\ &= \sqrt{\frac{p_i(1-p_i)}{d_i}} + \frac{1}{d_i} \leq \sqrt{\frac{p_i}{d_i}} + \frac{1}{d_i}.\end{aligned}$$

Putting all this together, Lemma 3.23 gives us

$$0 \leq h\left(\mathbb{E}\left[\frac{\chi_i}{d_i}\right]\right) - \mathbb{E}\left[h\left(\frac{\chi_i}{d_i}\right)\right] \leq \frac{1}{2} \cdot \frac{8\log \ell}{\sqrt{d_i p_i}} \cdot \left(\sqrt{\frac{p_i}{d_i}} + \frac{1}{d_i}\right) = \frac{4\log\ell}{d_i} + \frac{4\log\ell}{d_i\sqrt{d_i p_i}} \leq \frac{5\log\ell}{d_i},$$

where the last step uses $\sqrt{p_i d_i} \geq 2\log\ell$ for $i \in F_2$. □

We can now use the above claims and Proposition 2.8 to bound the expectation of X:

$$\begin{aligned}
\mathbb{E}[X] = \sum_{i\in F_2} d_i\left(h(p_i) - \mathbb{E}\left[h\left(\frac{\chi_i}{d_i}\right)\right]\right) &\leq \sum_{i\in F_2} d_i\left(h(p_i) - h\left(\mathbb{E}\left[\frac{\chi_i}{d_i}\right]\right) + \frac{5\log\ell}{d_i}\right)\\
&\leq \sum_{i\in F_2} d_i\left(h\left(\left|p_i - \mathbb{E}\left[\frac{\chi_i}{d_i}\right]\right|\right) + \frac{5\log\ell}{d_i}\right)\\
&\leq \sum_{i\in F_2} d_i\left(h\left(\frac{1}{d_i}\right) + \frac{5\log\ell}{d_i}\right) \qquad (3.58)\\
&\leq \sum_{i\in F_2} d_i\left(\frac{2}{d_i}\log d_i + \frac{5\log\ell}{d_i}\right)\\
&\leq 7\ell^{1/2}\log\ell \leq \ell^{1/2}\log^3\ell,
\end{aligned}$$

where the first inequality is from Claim 3.22, the second is by Proposition 2.8, the third one follows from Claim 3.20, the fourth inequality is from Proposition 2.7, and the next ones follow from $d_i \leq \ell$, $|F_2| \leq m \leq \ell^{1/2}$, and $\log\ell > 8$ by the conditions for this Lemma 3.17.

So we showed in Claim 3.19 that X does not exceed its expectations by more than $\ell^{1/2}\log^3\ell$ with high probability (over $\chi \sim \mathcal{D}$), and also that $E[X]$ is bounded by $\ell^{1/2}\log^3\ell$ in (3.58), and therefore X does not exceed $2\ell^{1/2}\log^3\ell$ with high probability. Specifically, it means that there exists $\mathcal{T} \subseteq \mathcal{T}_1$, such that $\mathbb{P}_{\chi\sim\mathcal{D}}[\chi \in \mathcal{T}] \geq 1 - \ell^{-\log\ell}$, and that for any $\boldsymbol{s} \in \mathcal{T}$ it holds $\sum\limits_{i\in F_2} d_i\left(h(p_i) - h\left(\frac{s_i}{d_i}\right)\right) \leq 2\ell^{1/2}\log^3\ell$. Recall that $\sum\limits_{i\in F_1} d_i h(p_i) \leq 5\ell^{1/2}\log^3\ell$ as we showed in (3.49). Thus, by summing these two inequalities, we conclude from (3.47) that $\sum_{i=1}^m d_i\left(h(p_i) - h\left(\frac{s_i}{d_i}\right)\right) \leq 7\ell^{1/2}\log^3\ell$ for any $\boldsymbol{s} \in \mathcal{T}$.

Finally, the last step is to return back from the product of "truncated binomials" $\mathcal{D}$ to the original product of binomials Ω. As we defined the set $\mathcal{T}$ above, we have $\mathbb{P}_{\chi\sim\mathcal{D}}[\chi \in \mathcal{T}] \geq 1 - \ell^{-\log\ell}$. But by (3.56) the distributions Ω and $\mathcal{D}$ are very close to each other, and therefore we obtain:

$$\mathbb{P}_{\chi\sim\Omega}[\chi \in \mathcal{T}] = \mathbb{P}_{\chi\sim\mathcal{D}}[\chi \in \mathcal{T}] \cdot \theta \geq \left(1 - \ell^{-\log\ell}\right)\left(1 - \ell^{-(\log\ell)/3 + 1/2}\right) \geq 1 - \ell^{-(\log\ell)/4},$$

where we used the bound (3.54) on θ for the first inequality and $\log\ell \geq 8$ for the second one. □

This concludes the proof of Theorem 3.1 for the case of a BMS channel with a bounded size of an output alphabet (modulo some technical lemmas, proofs for which are deferred to Section 3.5).

3.4 Strong converse for BMS channels with arbitrary alphabet size

In this section we finish the proof of Theorem 3.1 for the general BMS channel using the results from the previous section.

For BMS channels with large output alphabet size we will use binning of the output, however we will do it in a way that *upgrades* the channel, rather than degrades it (recall Definition 2.9). Specifically, we will employ the following statement:

Proposition 3.24. *Let W be any BMS channel. Then there exists another BMS channel $\widetilde{W}$ with the following properties:*

(i) Output alphabet size of $\widetilde{W}$ is at most $2\sqrt{\ell}$;

(ii) $\widetilde{W}$ is upgraded with respect to W, i.e. $W \preceq \widetilde{W}$;

(iii) $H(\widetilde{W}) \geq H(W) - \dfrac{\log \ell}{\ell^{1/2}}$.

Before proving this proposition, we first show how we can finish a proof of Theorem 3.1 using it. So, consider any BMS channel W with output alphabet size larger than $2\sqrt{\ell}$, and consider the channel $\widetilde{W}$ which satisfies properties (i)-(iii) from Proposition 3.24 with respect to W. First of all, notice that $k \geq \ell(1 - H(W)) + 14\ell^{1/2}\log^3 \ell \geq \ell\left(1 - H(\widetilde{W}) - \frac{\log \ell}{\ell^{1/2}}\right) + 14\ell^{1/2}\log^3 \ell$, and thus $k \geq \ell(1 - H(\widetilde{W})) + 13\ell^{1/2}\log^3 \ell$. Taking the property (i) into consideration, it follows that the channel $\widetilde{W}$ satisfies all the conditions for the arguments in the Section 3.3 to be applied (see remark 3.6), i.e. the statement of Theorem 3.1 holds for $\widetilde{W}$. Therefore, we can argue that with probability at least $1 - \ell^{-(\log \ell)/20}$ over a random kernel G it holds $H(V_1 \,|\, \widetilde{\boldsymbol{Y}}) \geq 1 - \ell^{-(\log \ell)/20}$, where $\widetilde{\boldsymbol{Y}} = \widetilde{W}^\ell(\boldsymbol{V} \cdot G)$ is the output vector if one would use the channel $\widetilde{W}$ instead of W, for $\boldsymbol{V} \sim \{0,1\}^k$.

Now, let W_1 be the channel which "proves" that $\widetilde{W}$ is upgraded with respect to W, i.e. $W_1\left(\widetilde{W}(x)\right)$ and $W(x)$ are identically distributed for any $x \in \{0,1\}$. Trivially then, $W_1^\ell\left(\widetilde{W}^\ell(X)\right)$ and $W^\ell(X)$ are identically distributed for any random variable X supported on $\{0,1\}^\ell$.

Next, observe that the following forms a Markov chain

$$V_1 \to \boldsymbol{V} \to \boldsymbol{V} \cdot G \to \widetilde{W}^\ell(\boldsymbol{V}G) \to W_1^\ell\left(\widetilde{W}^\ell(\boldsymbol{V}G)\right),$$

where $\boldsymbol{V}$ is distributed uniformly over $\{0,1\}^k$. But then the data-processing inequality gives

$$I\left(V_1\,;\, W_1^\ell\left(\widetilde{W}^\ell(\boldsymbol{V}G)\right)\right) \leq I\left(V_1\,;\, \widetilde{W}^\ell(\boldsymbol{V}G)\right).$$

However, as we discussed above, $W_1^\ell\left(\widetilde{W}^\ell(\boldsymbol{V}G)\right)$ and $W^\ell(\boldsymbol{V}G)$ are identically distributed, and so

$$I(V_1\,;\,\boldsymbol{Y}) = I\left(V_1\,;\, W^\ell(\boldsymbol{V}G)\right) = I\left(V_1\,;\, W_1^\ell\left(\widetilde{W}^\ell(\boldsymbol{V}G)\right)\right) \leq I\left(V_1\,;\, \widetilde{W}^\ell(\boldsymbol{V}G)\right) = I(V_1\,;\,\widetilde{\boldsymbol{Y}}).$$

Therefore using $H(X|Y) = H(X) - I(X;Y)$ we derive that

$$H(V_1 \mid \boldsymbol{Y}) \geq H(V_1 \mid \widetilde{\boldsymbol{Y}}) \geq 1 - \ell^{-(\log \ell)/20}$$

with probability at least $1 - \ell^{-(\log \ell)/20}$. This concludes the proof of Theorem 3.1. $\square$

Proof of Proposition 3.24. We describe how to construct such an upgraded channel $\widetilde{W}$. We again are going to look at W as a convex combination of BSCs, as we discussed in Section 3.3.1: let W consist of m underlying BSC subchannels $W^{(1)}, W^{(2)} \dots, W^{(m)}$, each has probability q_j to be chosen. The subchannel $W^{(j)}$ has crossover probability p_j, and $0 \leq p_1 \leq \dots \leq p_m \leq \frac{1}{2}$. The subchannel $W^{(j)}$ can output $z_j^{(0)}$ or $z_j^{(1)}$, and the whole output alphabet is then $\mathcal{Y} = \{z_1^{(0)}, z_1^{(1)}, z_2^{(0)}, z_2^{(1)}, \dots, z_m^{(0)}, z_m^{(1)}\}$, $|\mathcal{Y}| = 2m$. It will be convenient to write the transmission probabilities of W explicitly: for any $k \in [m]$, $c, x \in \{0,1\}$:

$$W\left(z_k^{(c)} \,\middle|\, x\right) = \begin{cases} q_k \cdot (1 - p_k), & x = c, \\ q_k \cdot p_k, & x \neq c. \end{cases} \tag{3.59}$$

The key ideas behind the construction of $\widetilde{W}$ are the following:

- decreasing a crossover probability in any BSC (sub)channel always upgrades the channel, i.e. $\mathrm{BSC}_{p_1} \preceq \mathrm{BSC}_{p_2}$ for any $0 \leq p_2 \leq p_1 \leq \frac{1}{2}$ ([TV13, Lemma 9]). Indeed, one can simulate a flip of coin with bias p_1 by first flipping a coin with bias p_2, and then flipping the result one more time with probability $q = \frac{p_1 - p_2}{1 - 2p_2}$. In other words, $\mathrm{BSC}_{p_1}(x)$ and $\mathrm{BSC}_q\left(\mathrm{BSC}_{p_2}(x)\right)$ are identically distributed for $x \in \{0,1\}$.
- "binning" two BSC subchannels with the same crossover probability doesn't change the channel ([TV13, Corollary 10]).

Let us finally describe how to construct $\widetilde{W}$. Split the interval $[0, 1/2]$ into $\sqrt{\ell}$ parts evenly, i.e. let $\theta_j = \frac{j-1}{2\sqrt{\ell}}$ for $j = 1, 2, \dots, \sqrt{\ell} + 1$, and consider intevals $[\theta_j, \theta_{j+1})$ for $j = 1, 2, \dots, \sqrt{\ell}$ (include $1/2$ into the last interval). Now, to get $\widetilde{W}$, we first slightly decrease the crossover probabilities in all the BSC subchannels $W^{(1)}, W^{(2)} \dots, W^{(m)}$ so that they all become one of $\theta_1, \theta_2, \dots, \theta_{\sqrt{\ell}}$. After that we bin together the subchannels with the same crossover probabilities and let the resulting channel be $\widetilde{W}$. Formally, we define

$$\begin{aligned} T_j &\coloneqq \left\{ i \in [m] \;:\; p_i \in \Big[\theta_j, \theta_{j+1}\Big) \right\}, \qquad\qquad j = 1, 2, \dots, \sqrt{\ell} - 1, \\ T_{\sqrt{\ell}} &\coloneqq \left\{ i \in [m] \;:\; p_i \in \Big[\theta_{\sqrt{\ell}}, \theta_{\sqrt{\ell}+1}\Big] \right\}. \end{aligned}$$

So, T_j is going to be the set of indices of subchannels of W for which we decrease the crossover probability to be equal to θ_j. Then the probability distribution over the new, binned, BSC subchannels $\widetilde{W^{(1)}}, \widetilde{W^{(2)}} \dots, \widetilde{W^{(\sqrt{\ell})}}$ in the channel $\widetilde{W}$ is going to be

$(\widetilde{q_1}, \widetilde{q_2}, \dots, \widetilde{q_{\sqrt{\ell}}})$, where $\widetilde{q_j} \coloneqq \sum\limits_{i \in T_j} q_i$. The subchannel $\widetilde{W^{(j)}}$ has crossover probability θ_j, and it can output one of two new symbols $\widetilde{z_j^{(0)}}$ or $\widetilde{z_j^{(1)}}$. The whole output alphabet is then $\widetilde{\mathcal{Y}} = \{\widetilde{z_1^{(0)}}, \widetilde{z_1^{(1)}}, \widetilde{z_2^{(0)}}, \widetilde{z_2^{(1)}}, \dots, \widetilde{z_{\sqrt{\ell}}^{(0)}}, \widetilde{z_{\sqrt{\ell}}^{(1)}}\}$, $|\widetilde{\mathcal{Y}}| = 2\sqrt{\ell}$. To be more specific, we describe $\widetilde{W} : \{0,1\} \to \widetilde{\mathcal{Y}}$, as follows: for any $j \in [\sqrt{\ell}]$ and any $b, x \in \{0,1\}$

$$\widetilde{W}\left(\widetilde{z_j^{(b)}} \,\middle|\, x\right) = \begin{cases} \sum\limits_{i \in T_j} q_i \cdot (1 - \theta_j), & x = b, \\ \sum\limits_{i \in T_j} q_i \cdot \theta_j, & x \neq b. \end{cases} \tag{3.60}$$

Property (i) on the output alphabet size for $\widetilde{W}$ then holds immediately. Let us verify (ii) by showing that $\widetilde{W}$ is indeed upgraded with respect to W.

One can imitate the usage of W using $\widetilde{W}$ as follows: on input $x \in \{0,1\}$, feed it through $\widetilde{W}$ to get output $\widetilde{z_j^{(b)}}$ for some $b \in \{0,1\}$ and $j \in [\sqrt{\ell}]$. We then know that the subchannel $\widetilde{W^{(j)}}$ was used, which by construction corresponds to the usage of a subchannel $W^{(i)}$ for some $i \in T_j$. Then we randomly choose an index k from T_j with probability of $i \in T_j$ being chosen equal to $\dfrac{q_i}{\widetilde{q_j}}$. This determines that we are going to use the subchannel $W^{(k)}$ while imitating the usage of W. By now we flipped the input with probability θ_j (since we used the subchannel $\widetilde{W^{(j)}}$), while we want it to be flipped with probability $p_k \geq \theta_j$ overall, since we decided to use $W^{(k)}$. So the only thing we need to do it to "flip" b to $(1-b)$ with probability $\frac{p_k - \theta_j}{1 - 2\theta_j}$, and then output $z_k^{(b)}$ or $z_k^{(1-b)}$ correspondingly.

Formally, we just describe the channel $W_1 : \widetilde{\mathcal{Y}} \to \mathcal{Y}$ which proves that $\widetilde{W}$ is upgraded with respect to W by all of its transmission probabilities: for all $k \in [m]$, $j \in [\sqrt{\ell}]$, $b, c \in \{0,1\}$ set

$$W_1\left(z_k^c \,\middle|\, \widetilde{z_j^{(b)}}\right) = \begin{cases} 0, & k \notin T_j \\ \dfrac{q_k}{\sum\limits_{i \in T_j} q_i} \cdot \left(1 - \dfrac{p_k - \theta_j}{1 - 2\theta_j}\right), & k \in T_j,\ b = c, \\ \dfrac{q_k}{\sum\limits_{i \in T_j} q_i} \cdot \left(\dfrac{p_k - \theta_j}{1 - 2\theta_j}\right), & k \in T_j,\ b \neq c. \end{cases} \tag{3.61}$$

It is easy to check that W_1 is a valid channel, and that it holds for any $k \in [m]$ and $c, x \in \{0,1\}$

$$\sum_{j \in [\sqrt{\ell}],\, b \in \{0,1\}} \widetilde{W}\left(\widetilde{z_j^{(b)}} \,\middle|\, x\right) W_1\left(z_k^{(c)} \,\middle|\, \widetilde{z_j^{(b)}}\right) = W\left(z_k^{(c)} \,\middle|\, x\right), \tag{3.62}$$

which proves that $\widetilde{W}$ is indeed upgraded to W. For completeness, we prove the above equality in Section 3.5.3.

It only remains to check that the property (iii) also holds, i.e. that the entropy did not decrease too much after we upgrade the channel W to $\widetilde{W}$. We have

$$H\left(\widetilde{W}\right) = \sum_{j\in[\sqrt{\ell}]} \widetilde{q}_j h(\theta_j) = \sum_{j\in[\sqrt{\ell}]} \left(\sum_{i\in T_j} q_i\right) h(\theta_j) = \sum_{k\in[m]} q_k h(\theta_{j_k}),$$

where we again denoted by j_k the index from $[\sqrt{\ell}]$ for which $k \in T_{j_k}$. Therefore

$$H(W) - H\left(\widetilde{W}\right) = \sum_{k\in[m]} q_k\Big(h(p_k) - h(\theta_{j_k})\Big) \le \sum_{k\in[m]} q_k\Big(h(\theta_{j_k+1}) - h(\theta_{j_k})\Big),$$

since $p_k \in [\theta_{j_k}, \theta_{j_k+1}]$ as $k \in T_{j_k}$. Finally, since $\theta_{j+1} - \theta_j = \frac{1}{2\sqrt{\ell}}$, Proposition 2.8 gives

$$H(W) - H\left(\widetilde{W}\right) \le \sum_{k\in[m]} q_k\Big(h(\theta_{j_k+1}) - h(\theta_{j_k})\Big) \le h\left(\frac{1}{2\sqrt{\ell}}\right) \le 2\cdot\frac{1}{2\sqrt{\ell}}\log\left(2\sqrt{\ell}\right) \le \frac{\log\ell}{\sqrt{\ell}}.$$

This marks the end of the proof for Proposition 3.24, and therefore of the general case of Theorem 3.1, modulo deferred technical proofs presented in the next section. □

3.5 Deferred proofs

3.5.1 Proofs of entropic lemma for BMS channels

In the following proof we use the representation of BMS channel W as a convex combination of several BSC subchannels $W^{(1)}$, $W^{(2)}$, …, $W^{(m)}$, see the beginning of Section 3.3.1 for details. Each subchannel $W^{(j)}$ can output one of two symbols $z_j^{(0)}, z_j^{(1)}$, and $W^{(j)}(z_j^{(0)}|0) = W^{(j)}(z_j^{(1)}|1)$, $W^{(j)}(z_j^{(1)}|0) = W^{(j)}(z_j^{(0)}|1)$. The output alphabet for W is thus $\mathcal{Y} = \{z_1^{(0)}, z_1^{(1)}, z_2^{(0)}, z_2^{(1)}, \dots, z_m^{(0)}, z_m^{(1)}\}$. Define for these proofs the "flip" operator $\oplus : \mathcal{Y}\times\{0,1\} \to \mathcal{Y}$ as follows: $z_j^{(c)} \oplus b = z_j^{(b+c)}$, where $b, c \in \{0,1\}$, and $(b+c)$ is addition mod 2. In other words, $z_j^{(c)} \oplus 0$ doesn't change anything, and $z_j^{(c)} \oplus 1$ flips the output of the subchannel $W^{(j)}$ to the opposite symbol. Note then that $W^{(j)}(z_j^{(c)}\,|\,b) = W^{(j)}(z_j^{(c)} \oplus b\,|\,0)$. Finally, we overload the operator to also work on $\mathcal{Y}^\ell \times \{0,1\}^\ell \to \mathcal{Y}^\ell$ by applying it coordinate-wise. It then easily follows that $W^\ell\left(\boldsymbol{y}\,|\,\boldsymbol{x}\right) = W^\ell\left(\boldsymbol{y}\oplus\boldsymbol{x}\,|\,\boldsymbol{0}\right)$ for any $\boldsymbol{y} \in \mathcal{Y}^\ell$ and $\boldsymbol{x} \in \{0,1\}^\ell$.

Proof of Lemma 3.2. We can write

$$\begin{aligned}\mathop{\mathbb{E}}_{g\sim G}\left[H^{(g)}(V_1|\boldsymbol{Y})\right] &= \sum_g \mathbb{P}(G=g)\left(\sum_{\boldsymbol{y}\in\mathcal{Y}^\ell} \mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}]H^{(g)}(V_1|\boldsymbol{Y}=\boldsymbol{y})\right)\\ &= \sum_g \mathbb{P}(G=g)\left(\sum_{\boldsymbol{y}\in\mathcal{Y}^\ell}\left(\sum_{\boldsymbol{v}\in\{0,1\}^k}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y},\,\boldsymbol{V}=\boldsymbol{v}]\right)h\left(\frac{\mathbb{P}^{(g)}[V_1=0,\,\boldsymbol{Y}=\boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}]}\right)\right)\\ &= \frac{1}{2^k}\sum_{\boldsymbol{v}\in\{0,1\}^k}\sum_g \mathbb{P}(G=g)\sum_{\boldsymbol{y}\in\mathcal{Y}^\ell}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\Big|\boldsymbol{V}=\boldsymbol{v}]h\left(\frac{\mathbb{P}^{(g)}[V_1=0,\,\boldsymbol{Y}=\boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}]}\right),\quad(3.63)\end{aligned}$$

where $h(x) := -x\log_2 x - (1-x)\log_2(1-x)$ is the binary entropy function. Next, we show that for any fixed matrix g and any fixed $\boldsymbol{v} \in \{0,1\}^k$ it holds

$$\sum_{\boldsymbol{y}\in\mathcal{Y}^\ell} \mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\Big|\boldsymbol{V}=\boldsymbol{v}]h\left(\frac{\mathbb{P}^{(g)}[V_1=0,\boldsymbol{Y}=\boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}]}\right) = \\ = \sum_{\boldsymbol{y}\in\mathcal{Y}^\ell} \mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\Big|\boldsymbol{V}=\boldsymbol{0}]h\left(\frac{\mathbb{P}^{(g)}[V_1=0,\boldsymbol{Y}=\boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}]}\right), \quad (3.64)$$

where $\boldsymbol{0}$ is the all-zero vector.

First of all, we know that

$$\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\Big|\boldsymbol{V}=\boldsymbol{v}] = W^\ell(\boldsymbol{y}\mid\boldsymbol{v}G) = W^\ell(\boldsymbol{y}\oplus\boldsymbol{v}G\mid\boldsymbol{0}) = \mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G\mid\boldsymbol{V}=\boldsymbol{0}], \quad (3.65)$$

as was discussed at the beginning of this appendix. In the same way, it's easy to see

$$\begin{aligned}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}] &= \frac{1}{2^k}\sum_{\boldsymbol{u}\in\{0,1\}^k}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\Big|\boldsymbol{V}=\boldsymbol{u}] \\ &= \frac{1}{2^k}\sum_{\boldsymbol{u}\in\{0,1\}^k}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G\Big|\boldsymbol{V}=\boldsymbol{u}+\boldsymbol{v}] \\ &= \frac{1}{2^k}\sum_{\boldsymbol{u}+\boldsymbol{v}\in\{0,1\}^k}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G\Big|\boldsymbol{V}=\boldsymbol{u}+\boldsymbol{v}] = \mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G].\end{aligned} \quad (3.66)$$

The above equality uses the fact that we are considering linear codes, and $\boldsymbol{v}G$ is an arbitrary codeword. It follows from the symmetry of linear codes that "shifting" the output by a codeword does not change anything. Shifting here means the usual shifting for the BSC case, though for a general BMS channel this is actually flipping the outputs or appropriate BSC subchannels, without changing which subchannel was used for which bit.

Denote now $\widetilde{\boldsymbol{V}} = \boldsymbol{V}_{>1}$, and recall that we are considering fixed $\boldsymbol{v}$ for now. Denote then also v_1 as the first coordinate of $\boldsymbol{v}$ and $\tilde{\boldsymbol{v}} = \boldsymbol{v}_{>1}$. Then we derive similarly

$$\begin{aligned}\mathbb{P}^{(g)}[V_1=0,\boldsymbol{Y}=\boldsymbol{y}] &= \frac{1}{2^k}\sum_{\tilde{\boldsymbol{u}}\in\{0,1\}^{k-1}}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\Big|V_1=0,\widetilde{\boldsymbol{V}}=\tilde{\boldsymbol{u}}] \\ &= \frac{1}{2^k}\sum_{\tilde{\boldsymbol{u}}\in\{0,1\}^{k-1}}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G\mid V_1=v_1,\widetilde{\boldsymbol{V}}=\tilde{\boldsymbol{u}}+\tilde{\boldsymbol{v}}] \\ &= \frac{1}{2^k}\sum_{\tilde{\boldsymbol{u}}+\tilde{\boldsymbol{v}}\in\{0,1\}^{k-1}}\mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G\Big|V_1=v_1,\widetilde{\boldsymbol{V}}=\tilde{\boldsymbol{u}}+\tilde{\boldsymbol{v}}] \\ &= \mathbb{P}^{(g)}[V_1=v_1,\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G].\end{aligned} \quad (3.67)$$

Notice that

$$\mathbb{P}^{(g)}[V_1=v_1,\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G] + \mathbb{P}^{(g)}[V_1=1-v_1,\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G] = \mathbb{P}^{(g)}[\boldsymbol{Y}=\boldsymbol{y}\oplus\boldsymbol{v}G],$$

and thus using the symmetry of the binary entropy function around 1/2 obtain

$$h\left(\frac{\mathbb{P}^{(g)}[V_1 = v_1, \boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{v}G]}{\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{v}G]}\right) = h\left(\frac{\mathbb{P}^{(g)}[V_1 = 1 - v_1, \boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{v}G]}{\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{v}G]}\right).$$

Using this and (3.65)–(3.67) derive

$$\begin{aligned}
&\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y} \Big| \boldsymbol{V} = \boldsymbol{v}] h\left(\frac{\mathbb{P}^{(g)}[V_1 = 0, \boldsymbol{Y} = \boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y}]}\right) \\
=&\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{v}G \mid \boldsymbol{V} = \boldsymbol{0}] h\left(\frac{\mathbb{P}^{(g)}[V_1 = 0, \boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{v}G]}{\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{v}G]}\right).
\end{aligned}$$

Finally, summing both parts over $\boldsymbol{y} \in \mathcal{Y}^\ell$ and noticing that $\boldsymbol{y} \oplus \boldsymbol{v}G$ will also range through all $\mathcal{Y}^\ell$ in this case, we establish (3.64). Then in (3.63) deduce

$$\begin{aligned}
\mathop{\mathbb{E}}_{g \sim G}\left[H^{(g)}(V_1 | \boldsymbol{Y})\right] &= \frac{1}{2^k} \sum_{\boldsymbol{v} \in \{0,1\}^k} \sum_g \mathbb{P}(G = g) \sum_{\boldsymbol{y} \in \mathcal{Y}^\ell} \mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y} \Big| \boldsymbol{V} = \boldsymbol{0}] h\left(\frac{\mathbb{P}^{(g)}[V_1 = 0, \boldsymbol{Y} = \boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y}]}\right) \\
&= \sum_{\boldsymbol{y} \in \mathcal{Y}^\ell} \sum_g \mathbb{P}(G = g)\, \mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y} \Big| \boldsymbol{V} = \boldsymbol{0}] h\left(\frac{\mathbb{P}^{(g)}[V_1 = 0, \boldsymbol{Y} = \boldsymbol{y}]}{\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y}]}\right) \\
&= \sum_{\boldsymbol{y} \in \mathcal{Y}^\ell} \mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \Big| \boldsymbol{V} = \boldsymbol{0}] \mathop{\mathbb{E}}_{g \sim G}\left[H^{(g)}(V_1 | \boldsymbol{Y} = \boldsymbol{y})\right],
\end{aligned}$$

since $\mathbb{P}^{(g)}[\boldsymbol{Y} = \boldsymbol{y} \Big| \boldsymbol{V} = \boldsymbol{0}]$ does not depend on the matrix g. □

3.5.2 Proofs for lemmas in Section 3.3.6

Proof of Claim 3.20. Denote for convenience the distribution $\Omega_i := \text{Binom}(d_i, p_i)$. Note that $\mathbb{E}_{\chi_i \sim \Omega_i}\left[\frac{\chi_i}{d_i}\right] = p_i$. Then we derive

$$\begin{aligned}
\left|\mathop{\mathbb{E}}_{\chi_i \sim \mathcal{D}_i}\left[\frac{\chi_i}{d_i}\right] - p_i\right| &= \left|\mathop{\mathbb{E}}_{\chi_i \sim \mathcal{D}_i}\left[\frac{\chi_i}{d_i}\right] - \mathop{\mathbb{E}}_{\chi_i \sim \Omega_i}\left[\frac{\chi_i}{d_i}\right]\right| \\
&= \left|\sum_{s \in [0:d_i]} \frac{s}{d_i} \mathop{\mathbb{P}}_{\chi_i \sim \mathcal{D}_i}[\chi_i = s] - \sum_{s \in [0:d_i]} \frac{s}{d_i} \mathop{\mathbb{P}}_{\chi_i \sim \Omega_i}[\chi_i = s]\right| \\
&\overset{(3.55)}{=} \left|\sum_{s \in \mathcal{T}_1^{(i)}} \frac{s}{d_i} \mathop{\mathbb{P}}_{\chi_i \sim \Omega_i}[\chi_i = s] \cdot \theta_i^{-1} - \sum_{s \in [0:d_i]} \frac{s}{d_i} \mathop{\mathbb{P}}_{\chi_i \sim \Omega_i}[\chi_i = s]\right| \\
&= \left|\sum_{s \in \mathcal{T}_1^{(i)}} \frac{s}{d_i} \mathop{\mathbb{P}}_{\chi_i \sim \Omega_i}[\chi_i = s] \cdot \left(\theta_i^{-1} - 1\right) - \sum_{s \notin \mathcal{T}_1^{(i)}} \frac{s}{d_i} \mathop{\mathbb{P}}_{\chi_i \sim \Omega_i}[\chi_i = s]\right| \\
&\le \sum_{s \in \mathcal{T}_1^{(i)}} \frac{s}{d_i} \mathop{\mathbb{P}}_{\chi_i \sim \Omega_i}[\chi_i = s] \cdot \left(\theta_i^{-1} - 1\right) + \sum_{s \notin \mathcal{T}_1^{(i)}} \frac{s}{d_i} \mathop{\mathbb{P}}_{\chi_i \sim \Omega_i}[\chi_i = s].
\end{aligned}$$

We have $\sum\limits_{s\notin\mathcal{T}_1^{(i)}} \frac{s}{d_i}\,\mathbb{P}_{\chi_i\sim\Omega_i}[\chi_i = s] \leq \sum\limits_{s\notin\mathcal{T}_1^{(i)}} \mathbb{P}_{\chi_i\sim\Omega_i}[\chi_i = s] \overset{(3.53)}{=} (1-\theta_i) \overset{(3.50)}{\leq} 2\ell^{-(\log\ell)/3}$.

Next, $\sum_{s\in\mathcal{T}_1^{(i)}} \frac{s}{d_i}\,\mathbb{P}_{\chi_i\sim\Omega_i}[\chi_i = s] \leq \mathbb{E}_{\chi_i\sim\Omega_i}\left[\frac{\chi_i}{d_i}\right] \leq 1$, and $\theta_i^{-1} - 1 = \frac{1-\theta_i}{\theta_i} \leq 2(1-\theta_i) \leq 4\ell^{-(\log\ell)/3}$.

Combining the above together, conclude $\left|\mathbb{E}\left[\frac{\chi_i}{d_i}\right] - p_i\right| \leq 6\ell^{-(\log\ell)/3} \leq \frac{1}{\ell} \leq \frac{1}{d_i}$. □

Proof of Claim 3.21. Using the result of Claim 3.20 derive

$$\mathbb{E}\left|\chi_i - \mathbb{E}[\chi_i]\right| \leq \mathbb{E}\left|\chi_i - p_i d_i\right| + \mathbb{E}\left|p_i d_i - \mathbb{E}[\chi_i]\right| \leq \mathbb{E}\left|\chi_i - p_i d_i\right| + 1. \tag{3.68}$$

From (3.53), (3.56), and definition (3.52) of $\mathcal{T}_1^{(i)}$ for $i \in F_2$ observe also the following:

$$\begin{aligned}
\mathop{\mathbb{E}}_{\chi_i\sim\mathcal{D}_i}\left|\chi_i - p_i d_i\right| &= \sum_{s\in\mathcal{T}_1^{(i)}} \left|s - p_i d_i\right| \cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s]\cdot\theta_i^{-1} \\
&= \sum_{s\in\mathcal{T}_1^{(i)}} \left|s - p_i d_i\right| \cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s] + \sum_{s\in\mathcal{T}_1^{(i)}} \left|s - p_i d_i\right| \cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s]\cdot(\theta_i^{-1} - 1) \\
&\leq \sum_{s\in\mathcal{T}_1^{(i)}} \left|s - p_i d_i\right| \cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s] + \sqrt{d_i p_i}\log\ell\cdot \underbrace{\sum_{s\in\mathcal{T}_1^{(i)}} \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s]}_{\theta_i}\cdot\left(\frac{1-\theta_i}{\theta_i}\right) \\
&= \sum_{s\in\mathcal{T}_1^{(i)}} \left|s - p_i d_i\right| \cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s] + \sqrt{d_i p_i}\log\ell\cdot(1-\theta_i) \\
&= \sum_{s\in\mathcal{T}_1^{(i)}} \left|s - p_i d_i\right| \cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s] + \sum_{s\notin\mathcal{T}_1^{(i)}} \sqrt{d_i p_i}\log\ell\cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s] \\
&\leq \sum_{s\in\mathcal{T}_1^{(i)}} \left|s - p_i d_i\right| \cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s] + \sum_{s\notin\mathcal{T}_1^{(i)}} \left|s - p_i d_i\right| \cdot \mathop{\mathbb{P}}_{\eta_i\sim\Omega_i}[s] = \mathop{\mathbb{E}}_{\eta_i\sim\Omega_i}\left|\eta_i - p_i d_i\right|.
\end{aligned}$$

Combining this with (3.68), obtain the needed. □

3.5.3 Proof in Section 3.4

Here we explicitly show that the channel $\widetilde{W}$ we constructed in Section 3.4 is indeed upgraded with respect to W. Recall that W, $\widetilde{W}$, and W_1 are defined in (3.59), (3.60), and (3.61) correspondingly, and our goal is to prove (3.62). First, to check that W_1 is a valid channel, observe

$$\sum_{k\in[m],\, c\in\{0,1\}} W_1\left(z_k^{(c)}\;\middle|\;\widetilde{z_j^{(b)}}\right) = \sum_{k\in T_j}\left(W_1\left(z_k^0\;\middle|\;\widetilde{z_j^{(b)}}\right) + W_1\left(z_k^1\;\middle|\;\widetilde{z_j^{(b)}}\right)\right) = \sum_{k\in T_j}\frac{q_k}{\sum\limits_{i\in T_j} q_i} = 1.$$

Finally, for any $k \in [m]$, $c \in \{0,1\}$, let j_k be such that $k \in T_{j_k}$. Then we have for any $x \in \{0,1\}$

$$\sum_{j\in[\sqrt{\ell}],\, b\in\{0,1\}} \widetilde{W}\left(\widetilde{z_j^{(b)}} \,\middle|\, x\right) W_1\left(z_k^{(c)} \,\middle|\, \widetilde{z_j^{(b)}}\right) = \sum_{b\in\{0,1\}} \widetilde{W}\left(\widetilde{z_{j_k}^{(b)}} \,\middle|\, x\right) W_1\left(z_k^{(c)} \,\middle|\, \widetilde{z_{j_k}^{(b)}}\right).$$

Now, if $x = c$, we derive

$$\begin{aligned}
&\sum_{b\in\{0,1\}} \widetilde{W}\left(\widetilde{z_{j_k}^{(b)}} \,\middle|\, x\right) W_1\left(z_k^{(c)} \,\middle|\, \widetilde{z_{j_k}^{(b)}}\right)\\
&\qquad= \widetilde{W}\left(\widetilde{z_{j_k}^{(x)}} \,\middle|\, x\right) W_1\left(z_k^{(x)} \,\middle|\, \widetilde{z_{j_k}^{(x)}}\right) + \widetilde{W}\left(\widetilde{z_{j_k}^{(1-x)}} \,\middle|\, x\right) W_1\left(z_k^{(x)} \,\middle|\, \widetilde{z_{j_k}^{(1-x)}}\right)\\
&\qquad= \sum_{i\in T_{j_k}} q_i \cdot (1-\theta_{j_k}) \cdot \frac{q_k}{\sum\limits_{i\in T_{j_k}} q_i} \cdot \left(1 - \frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right) + \sum_{i\in T_{j_k}} q_i \cdot \theta_{j_k} \cdot \frac{q_k}{\sum\limits_{i\in T_{j_k}} q_i} \cdot \left(\frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right)\\
&\qquad= q_k\left(1 - \theta_{j_k} - (1-\theta_{j_k}) \cdot \left(\frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right) + \theta_{k_j} \cdot \left(\frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right)\right)\\
&\qquad= q_k\left(1 - \theta_{j_k} - (1-2\theta_{j_k}) \cdot \left(\frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right)\right) = q_k \cdot (1-p_k).
\end{aligned}$$

Otherwise, then $x = 1-c$, obtain

$$\begin{aligned}
&\sum_{b\in\{0,1\}} \widetilde{W}\left(\widetilde{z_{j_k}^{(b)}} \,\middle|\, x\right) W_1\left(z_k^{(c)} \,\middle|\, \widetilde{z_{j_k}^{(b)}}\right)\\
&\qquad= \widetilde{W}\left(\widetilde{z_{j_k}^{(x)}} \,\middle|\, x\right) W_1\left(z_k^{(1-x)} \,\middle|\, \widetilde{z_{j_k}^{(x)}}\right) + \widetilde{W}\left(\widetilde{z_{j_k}^{(1-x)}} \,\middle|\, x\right) W_1\left(z_k^{(1-x)} \,\middle|\, \widetilde{z_{j_k}^{(1-x)}}\right)\\
&\qquad= \sum_{i\in T_{j_k}} q_i \cdot (1-\theta_{j_k}) \cdot \frac{q_k}{\sum\limits_{i\in T_{j_k}} q_i} \cdot \left(\frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right) + \sum_{i\in T_{j_k}} q_i \cdot \theta_{j_k} \cdot \frac{q_k}{\sum\limits_{i\in T_{j_k}} q_i} \cdot \left(1 - \frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right)\\
&\qquad= q_k\left((1-\theta_{j_k}) \cdot \left(\frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right) + \theta_{j_k} - \theta_{j_k} \cdot \left(\frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right)\right)\\
&\qquad= q_k\left((1-2\theta_{j_k}) \cdot \left(\frac{p_k - \theta_{j_k}}{1-2\theta_{j_k}}\right) + \theta_{j_k}\right) = q_k \cdot p_k.
\end{aligned}$$

Therefore, for any $k \in [m]$ and $c, x \in \{0,1\}$

$$\sum_{j\in[\sqrt{\ell}],\, b\in\{0,1\}} \widetilde{W}\left(\widetilde{z_j^{(b)}} \,\middle|\, x\right) W_1\left(z_k^{(c)} \,\middle|\, \widetilde{z_j^{(b)}}\right) = W\left(z_k^{(c)} \,\middle|\, x\right).$$

Chapter 4

Polar Codes Overview

In this chapter we describe the context and background of the phenomenon of channel polarization that lies at the heart of Arıkan's polar coding approach. We start with describing the original Arıkan's construction from [Arı09], where the author introduced the first-ever family of codes that provably achieves capacity for any BMS channel and has low-complexity encoding and decoding procedures. This invention of polar codes is considered to be one of the most important major advances in modern coding theory, from both theoretical and practical standpoints. Further we review the state of the art on polar codes and discuss our results.

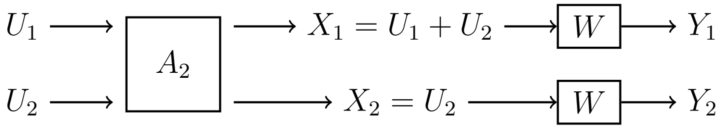
4.1 Original polar codes

4.1.1 Polar transformation

The idea of channel polarization introduced by Arıkan is based on a small linear transformation. Define a binary matrix

$$A_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix},$$

which we call a *basic Arıkan's kernel.* Let $\boldsymbol{U} = (U_1, U_2) \in \mathbb{F}^2$ be a vector of two uniformly random bits. Suppose we are transmitting the vector $\boldsymbol{U}$ by first encoding it as $\boldsymbol{X} = \boldsymbol{U} \cdot A_2$, and then sending $\boldsymbol{X}$ through two independent copies of the channel W, obtaining the output vector $\boldsymbol{Y} = (Y_1, Y_2)$.



Denote by $W_{(2)}$ the channel that sends $\boldsymbol{U}$ to $\boldsymbol{Y}$, i.e.

$$W_{(2)}(\boldsymbol{y} \mid \boldsymbol{u}) = W^2(\boldsymbol{y} \mid \boldsymbol{x}) = W^2(\boldsymbol{y} \mid \boldsymbol{u} \cdot A_2) = \prod_{i=1}^{2} W(y_i \mid (\boldsymbol{u}A_2)_i).$$

So, $W_{(2)}$ describes a combination of two copies of channels W that sends $\boldsymbol{U}$ to $\boldsymbol{Y}$, this step is referred to as *channel combining* in [Arı09]. Since the linear transformation A_2 is invertible, the mutual information between the output $\boldsymbol{Y}$ and either of $\boldsymbol{U}$, $\boldsymbol{X}$ is the same,

$$I(\boldsymbol{U}\,;\,\boldsymbol{Y}) = I(\boldsymbol{X}\,;\,\boldsymbol{Y}) = I(X_1\,;\,Y_1) + I(X_2\,;\,Y_2) = 2I(W). \tag{4.1}$$

Using the chain rule for the mutual information, we can also write

$$I(\boldsymbol{U}\,;\,\boldsymbol{Y}) = I(U_0\,;\,\boldsymbol{Y}) + I(U_1\,;\,\boldsymbol{Y}, U_0). \tag{4.2}$$

We can interpret the summands in this equation in the following way. Suppose we are trying to guess the message $\boldsymbol{U}$ based on the output vector $\boldsymbol{Y}$, and we are doing it in a specific way, in a successive fashion. We first decode the first bit U_1 based on $\boldsymbol{Y}$, producing a guess $\hat{U_1}$. And after that, we are trying to guess U_2 based on both $\boldsymbol{Y}$ and $\hat{U_1}$, where we are “pretending” (or assuming) that our first guess was correct.

Define then two synthetic channels, which correspond to these two guessing procedures. The channel $W^{(0)}\ : \mathbb{F}_2 \to \mathcal{Y}^2$ is the channel that sends U_1 to $\boldsymbol{Y}$, and it views U_2 as a random noise. The second channel $W^{(1)}\ : \mathbb{F}_2 \to \boldsymbol{Y}^2 \times \mathbb{F}_2$ sends the bit U_2 to $\boldsymbol{Y}$ and U_0. They can be formally described as:

$$\begin{aligned} W^{(0)}(\boldsymbol{y} \,|\, u_1) &= \frac{1}{2}\sum_{u_1 \in \mathbb{F}_2} W(y_1 \,|\, u_1 + u_2) W(y_2 \,|\, u_2), \\ W^{(1)}(\boldsymbol{y}, u_1 \,|\, u_2) &= \frac{1}{2} W(y_1 \,|\, u_1 + u_2) W(y_2 \,|\, u_2). \end{aligned} \tag{4.3}$$

The capacities of these channels are exactly the summands that appear in the RHS of (4.2). Then, using also the preservation of mutual information from (4.1), and rewriting it in the form of channel entropies, we observe the entropy preservation property:

$$H(W^{(0)}) + H(W^{(1)}) = 2H(W). \tag{4.4}$$

We transformed two copies of the channel W into a pair of channels $(W^{(0)}, W^{(1)})$ with preservation of total entropy, however, these channels are not identical anymore (except in some trivial cases). The following relation on the entropies is straightforward:

$$H(W^{(1)}) \le H(W) \le H(W^{(0)}).$$

Indeed, this follows from $H(W^{(1)}) = H(U_2 \,|\, \boldsymbol{Y}, U_1) \le H(U_2 \,|\, Y_2) = H(X_2 \,|\, Y_2) = H(W)$. Moreover, for the Bhattacharyya parameter on the new channels we have the relations

$$\begin{aligned} Z(W^{(0)}) &\le 2Z(W) - Z(W)^2, \\ Z(W^{(1)}) &= Z(W)^2. \end{aligned} \tag{4.5}$$

In other words, instead of a pair of identical channels (W, W), after this transformation we end up with a worse channel $W^{(0)}$ and a better channel $W^{(1)}$.

4.1.2 Recursive application

The next step is to apply this transformation recursively. We will combine two pairs of channels $(W^{(0)}, W^{(0)})$ and $(W^{(1)}, W^{(1)})$, and apply the same transformation to them, producing 4 channels $\left\{(W^{(0)})^{(0)}, (W^{(0)})^{(1)}, (W^{(1)})^{(0)}, (W^{(1)})^{(1)}\right\}$. Let us now understand why these channels are important.

For that, consider now sending 4 bits $\boldsymbol{U} \in \mathbb{F}_2^4$, which are transformed linearly into $\boldsymbol{X} = \boldsymbol{U} \cdot B_4 \cdot A_2^{\otimes 2}$, where $\otimes$ denotes a Kronecker product/powering, and B_n is a certain bit-reversal permutation[1]. After that $\boldsymbol{X}$ is sent through 4 copies of the channel W. Consider then again the process of guessing $\boldsymbol{U}$ based on $\boldsymbol{Y}$ in a successive manner: for $i = 1, 2, 3, 4$, we are producing a guess $\hat{U}_i$ for U_i, based on $\boldsymbol{Y}$ and our previous guesses $\hat{\boldsymbol{U}}_1^{i-1}$. Similarly, this can be viewed as trying to decode the communication through four synthetic bit-channels, where the channel $W_i : \mathbb{F}_2 \to \mathcal{Y}^4 \times \mathbb{F}_2^{i-1}$ sends the bit U_i to $\boldsymbol{Y}$ and $\boldsymbol{U}_1^{i-1}$, for all $i \in [4]$. Arıkan showed that these four synthetic bit-channels are exactly the channels $\left\{(W^{(0)})^{(0)}, (W^{(0)})^{(1)}, (W^{(1)})^{(0)}, (W^{(1)})^{(1)}\right\}$.

More generally (but analogically), say we are now sending $N = 2^n$ bits $\boldsymbol{U} \in \mathbb{F}_2^n$, by computing $\boldsymbol{X} = \boldsymbol{U} \cdot B_N \cdot A_2^{\otimes n}$ and sending $\boldsymbol{X}$ through N copies of W. Again, we are trying to decode $\boldsymbol{U}$ in the same successive fashion, which leads to N synthetic bit-channels

$$W_i : \mathbb{F}_2 \to \mathcal{Y}^N \times \mathbb{F}_2^{i-1}, \qquad W_i(\boldsymbol{y}, \boldsymbol{u}_1^{i-1} \,|\, u_i) \coloneqq \frac{1}{2^{N-1}} \sum_{\boldsymbol{u}_{i+1}^N \in \mathbb{F}_2^{N-i}} W^N(\boldsymbol{y} \,|\, \boldsymbol{u} \cdot B_N A_2^{\otimes n}). \tag{4.6}$$

Arıkan showed that the choice of linear transformation as a Kronecker product $A_2^{\otimes n}$, together with the bit-reversal permutation B_N, gives a recursive representation for these bit-channels, such that the set $\{W_i\}_{i=1}^N$ is equal to the set $\left\{\left(\dots\left(W^{(i_1)}\right)^{(i_2)}\dots\right)^{(i_n)}\right\}$, where $\boldsymbol{i} = (i_1, i_2, \dots, i_n) \in \{0,1\}^n$ ranges through all possible binary vectors of length n. This means that all N bit-channels can be obtained by recursively applying the basic polar transformations, defined in (4.3), n times.

4.1.3 Code construction

Let us describe how we construct and use (polar) codes given such a set of synthetic bit-channels $\{W_i\}_{i=1}^N$. To construct a code of rate R, select a set $\mathcal{I}$ of $k = RN$ indices from $[N]$, and use the bits in $\boldsymbol{U}$ on the corresponding positions, i.e. $\boldsymbol{U}_{\mathcal{I}}$, to send a message of k bits. Denote the remaining positions $\mathcal{F} = [N] \backslash \mathcal{I}$, and set a value for the remaining positions, i.e. $\boldsymbol{U}_{\mathcal{F}}$, to be equal to 0 (these are called *frozen bits*). This establishes the encoding, and the actual code C (set of codewords) is then going to be a set of all possible linear combinations of rows of $B_N A_2^{\otimes n}$ with indices from $\mathcal{I}$. Now, let us use the *successive cancellation decoding*

[1] The bit-reversal operation is not necessary for polarization and just makes the analysis easier. Even though it can be ignored for the pure purpose of this overview, for completeness: $\boldsymbol{V} = \boldsymbol{U} B_N$ for $N = 2^n$ for such $\boldsymbol{V}$ that $V_{(b_1, b_2, \dots, b_n)} = U_{(b_n, b_{n-1}, \dots, b_2, b_1)}$, where $(i-1) = (b_1, b_2, \dots, b_n)$ is the base-2 representation $(i-1)$, for all $i \in [N]$.

algorithm, the idea for which we actually considered these bit-channels. That is, for $i = 1, 2, \dots, N$, produce a guess $\hat{U}_i$ for U_i, based on $\boldsymbol{Y}$ and the previous guesses $\hat{\boldsymbol{U}}_1^{i-1}$. If $i \in \mathcal{I}$, this is done by comparing which one of $W_i(\boldsymbol{Y}, \hat{\boldsymbol{U}}_1^{i-1} \,|\, 0)$ and $W_i(\boldsymbol{Y}, \hat{\boldsymbol{U}}_1^{i-1} \,|\, 1)$ is larger. The probability to make an error at this step, if we guessed all the previous bits correctly and $\hat{\boldsymbol{U}}_1^{i-1} = \boldsymbol{U}_1^{i-1}$, is exactly the decoding error probability of W_i under the uniform input[2] , i.e. $P_e(W_i)$, for which we can use inequality $P_e(W_i) \le Z(W_i)$. Otherwise, for $i \in \mathcal{F}$, we guess $\hat{U}_i = 0$, as this is a frozen bit, and we cannot make a mistake at this step. Therefore, union bound implies that the overall decoding error probability is bounded by

$$P_e \le \sum_{i \in \mathcal{I}} Z(W_i). \tag{4.7}$$

To construct such codes which would allow reliable communication one would take the set of indices $\mathcal{I}$ for which $Z(W_i)$ is small. Therefore, we want to prove that there are a lot of channels, specifically as close to $I(W)N$ as possible, for which $Z(W_i)$ is small. Finally, we want $Z(W_i)$ to be sufficiently small to get the decoding error probability as low as possible. This is exactly what is called to be *channel polarization* – we want to show that out of these N channels, approximately $I(W)N$ will be almost noiseless. This would mean, because of the conservation of entropy property (4.4), that at the same time around $(1 - I(W))N$ channels should become very noisy.

All this leads us to the task of analyzing the bit-channels W_i and their parameters, in order to construct good (polar) codes. This is where the recursive way to describe these bit-channels comes in handy.

4.1.4 Stochastic processes

A convenient way to analyze the bit-channels is using a stochastic process of coding channels, where at each step we take one of the transformations in (4.3) with equal probability. Formally, let $\mathsf{W}_0 = W$ (the initial channel), and $\mathsf{W}_{i+1} = (\mathsf{W}_i)^{(0)}$ or $\mathsf{W}_{i+1} = (\mathsf{W}_i)^{(1)}$ with probability 1/2 each. We equivalently can define a random sequence $B_1, B_2, \dots$ of i.i.d Bernoulli(1/2) random variables, i.e. random 0/1 coin flips. In such settings we can define the channel process as $\mathsf{W}_{i+1} = (\mathsf{W}_i)^{(B_{i+1})}$.

Another useful way to look at it is by looking at the *channel tree*. For convenience, denote for $b_1, b_2, \dots, b_j \in \{0,1\}^j$ recursively $W_{b_1, b_2, \dots, b_j} = \left(W_{b_1, b_2, \dots, b_{j-1}}\right)^{(b_i)}$, i.e. this is the channel which is obtained after applying the corresponding transformation $\bullet^{(0)}$ an $\bullet^{(1)}$ in the order that the sequence $b_1, b_2, \dots, b_j$ dictates. The binary channel tree then is constructed in a natural manner – the root is W. After that, every channel W' in the tree has two children $(W')^{(0)}$ and $(W')^{(1)}$. Then the channels at level j of this tree are exactly the channels $\{W_{b_1, b_2, \dots, b_j}\}$ for all $b_1, b_2, \dots, b_j \in \{0,1\}^j$. Define the random process W as a random walk down this tree, which starts at the root, and at each steps moves to the child of the current channel, choosing either one with equal probability. Clearly, this

[2] technically, this is true on average over all possible values of frozen bits $\boldsymbol{U}_{\mathcal{F}}$, however [Arı09] also proves the inequality between $P_e(W_i)$ and $Z(W_i)$ for any fixed value of frozen bits if W is symmetric.

defines the same stochastic process W_n of channels. Notice that for a fixed n, the marginal distribution of W_n is uniform over all 2^n bit-channels at the level n of this tree.

Further, define two stochastic processes for channel parameters as $\mathsf{H}_n = H(\mathsf{W}_n)$ and $\mathsf{Z}_n = Z(\mathsf{W}_n)$. Then the entropy conservation property (4.4) and the relations for Bhattacharyya parameter (4.5) immediately imply the following

Proposition 4.1 ([Arı09], Prop. 8 and 9)**.** *H_n is a bounded martingale. Z_n is a bounded super-martingale.*

4.1.5 Channel polarization

The last proposition and evolution of the Bhattacharyya parameter (4.5) can be used to prove (a weak form of) polarization

Lemma 4.2 (Channel Polarization [Arı09, Prop. 8-10])**.**

- *The process $\{\mathsf{H}_n\}$ converges almost surely to a random variable H_∞ such that $H_\infty = 1$ w.p $H(W)$ and $H_\infty = 0$ w.p. $1 - H(W)$.*
- *The process $\{\mathsf{Z}_n\}$ converges almost surely to a random variable H_∞ such that $Z_\infty = 1$ w.p $H(W)$ and $Z_\infty = 0$ w.p. $1 - H(W)$.*

This implies the following *polarization behaviour*: for any $\delta > 0$

$$\begin{aligned}
&\mathop{\mathbb{P}}_{i\sim[N]}\left[H(W_i) \in (\delta, 1-\delta)\right] && \longrightarrow 0 \quad \text{as} \quad N \to \infty, \\
&\mathop{\mathbb{P}}_{i\sim[N]}\left[H(W_i) \in [0, \delta)\right] && \longrightarrow I(W) \quad \text{as} \quad N \to \infty, \\
&\mathop{\mathbb{P}}_{i\sim[N]}\left[H(W_i) \in (1-\delta, 1]\right] && \longrightarrow 1 - I(W) \quad \text{as} \quad N \to \infty,
\end{aligned}$$

meaning that as N increases, almost all of the bit-channels become either very noisy (entropy $> \delta$) or almost noiseless (entropy $< \delta$), and the fraction of *unpolarized* channels tends to 0. The same holds if we change the entropy to the Bhattacharyya parameter.

This does not quite give us capacity-achieving codes yet, as in (4.7) we want the sum of Bhattacharyya parameters $Z(W_i)$ for good channels to be $o(1/N)$ for decaying decoding error probability. What Arıkan originally proved in [Arı09] is the following:

Theorem 4.3 ([Arı09, Theorem 2])**.** *For any binary-input discrete memoryless channel W*

$$\lim_{n\to\infty} \mathbb{P}\left[\mathsf{Z}_n < 2^{-5n/4}\right] = I(W). \tag{4.8}$$

This allows to construct polar codes which have rates arbitrarily close to capacity with decaying decoding error probability:

Theorem 4.4 ([Arı09, Theorem 4])**.** *Consider any binary-input discrete memoryless channel W, and fix a rate $R < I(W)$, where $I(W)$ is a symmetric capacity of the channel*[3]*. Then there exists a family of polar codes with increasing blocklength N, with rates $R_N > R$ and decoding error probability $P_N = O(N^{1/4})$.*

[3] For symmetric channels, symmetric capacity matches the Definition 2.4 we considered.

Proof. Fix any $0 < \varepsilon < I(W) - R$, then from Theorem 4.3 have $\mathbb{P}\left[\mathsf{Z}_n < 2^{-5n/4}\right] \geq I(W) - \varepsilon$ for sufficiently large n. Recall that the marginal distribution of Z_n is uniform over $N = 2^n$ parameters of bit-channels $\{Z(W_1), Z(W_2), \dots, Z(W_N)\}$. As we described in Section 4.1.3, fix the set of *information indices*

$$\mathcal{I} = \left\{ i \in [N] \,:\, Z(W_i) < 2^{-5n/4} = N^{-5/4} \right\},$$

therefore the rate for such code is $R_N = \frac{|\mathcal{I}|}{N} \geq (I(W) - \varepsilon) > R$. Finally, under successive cancellation decoding the decoding error probability is bounded by (4.7), therefore

$$P_N \leq \sum_{i \in \mathcal{I}} Z(W_i) \leq N \cdot N^{-5/4} = N^{-1/4}. \qquad \square$$

This shows how to translate the limiting behavior of the parameters of the bit-channels as in (4.8) into constructing capacity-achieving codes. Soon after the original introduction of polar codes, the rate at which Z_n decays, when tends to 0, was improved:

Theorem 4.5 ([AT09, Theorem 2])**.** $\lim_{n\to\infty} \mathbb{P}\left[\mathsf{Z}_n < 2^{-2^{\beta n}}\right] = I(W)$ *for any* $\beta < 1/2$. *On the contrary, for any* $\beta' > 1/2$, $\lim_{n\to\infty} \mathbb{P}\left[\mathsf{Z}_n > 2^{-2^{\beta' n}}\right] = 1$ *if* $I(W) < 1$.

This established that polar codes have the decoding error probability scaling as $2^{-N^{1/2}}$. Furthermore, Arıkan presented in [Arı09] quasi-linear time $O(N \log N)$ algorithms for decoding and decoding procedures, which rely on a recursive (FFT-like) structure of the linear transformation $A_2^{\otimes n}$.

Notice, however, that this does not yield any results as to *how fast* the codes approach capacity in terms of blocklength N. That is, no estimation on how fast $\mathbb{P}\left[\mathsf{Z}_n < 2^{-2^{\beta n}}\right]$ tends to $I(W)$, in terms of n, is yet given, whereas recall that the main focus of this thesis in on the speed of this convergence. We discuss this after we do a slight detour and discuss two ways how the construction of polar codes can be generalized.

4.2 Larger kernels and general alphabets

While the original polar codes are based on a recursive application of a small 2×2 linear transform A_2, which is called the *kernel* of the transformation $A_2^{\otimes n}$, Arıkan inferred that polarization behavior is a general phenomenon, and is not restricted to this particular kernel. It was indeed shown in [KSU10] how to prove the polarization for $\ell \times \ell$ kernels for any $\ell \geq 2$, and a simple criterion for a matrix to be polarizing was given. For completeness (and since our results concern large kernels), let us write down the polarization transformation for this case explicitly, although it is very similar to the 2×2 case.

Consider an arbitrary BMS channel $W : \mathbb{F}_2 \to \mathcal{Y}$, and an $\ell \times \ell$ invertible binary matrix (kernel) K. Suppose we are transmitting a binary vector $\boldsymbol{U} = (U_1, U_2, \dots, U_\ell)$ uniformly chosen from $\{0,1\}^\ell$ in the following way: first, it is transformed into $\boldsymbol{X} = \boldsymbol{U}K$, which is then sent through ℓ copies of the channel W to get the output $\boldsymbol{Y} = W^\ell(\boldsymbol{X}) \in \mathcal{Y}^\ell$.

Now imagine decoding the input bits U_i successively in the order of increasing i. This naturally leads to a binary-input channel $W_i : \mathbb{F}_2 \to \mathcal{Y}^\ell \times \mathbb{F}_2^{i-1}$, for each $i \in [\ell]$, which is the channel "seen" by the bit U_i when all the previous bits $\boldsymbol{U}_{<i}$ and all the channel outputs $\boldsymbol{Y} \in \mathcal{Y}^\ell$ are known. Formally, the transition probabilities of this channel are

$$W_i(\boldsymbol{Y}, \boldsymbol{U}_{<i} \,|\, U_i) = \frac{1}{2^{\ell-1}} \sum_{\boldsymbol{V} \in \mathbb{F}_2^{\ell-i}} W^\ell\Big(\boldsymbol{Y} \,|\, (\boldsymbol{U}_{<i}, U_i, \boldsymbol{V})K\Big), \tag{4.9}$$

where $\boldsymbol{U}_{<i} \in \mathbb{F}_2^{i-1}$ are the first $(i-1)$ bits of $\boldsymbol{U}$, and the sum is over all possible values $\boldsymbol{V} \in \mathbb{F}_2^{\ell-i}$ that the last $(\ell - i)$ bits of $\boldsymbol{U}$ can take. This is a direct generalization of the bit-channels in (4.6). In this document we call the channel W_i as *"Arıkan's i^{th} bit-channel of W with respect to K"*, where some parts of this naming might be skipped and implied from the context.

A *polarization transform* associated with the kernel K is then defined as a transformation that maps ℓ copies of the channel W to the bit-channels W_1, $W_2, \dots, W_\ell$. Since K is invertible, a direct implication of the chain rule for entropy also gives the entropy conservation property, which is

$$\ell \cdot H(W) = H(\boldsymbol{X}|\boldsymbol{Y}) = H(\boldsymbol{U}|\boldsymbol{Y}) = \sum_{i=1}^{\ell} H(U_i|\boldsymbol{U}_{<i}, \boldsymbol{Y}) = \sum_{i=1}^{\ell} H(W_i). \tag{4.10}$$

Definition 4.6. *An invertible binary matrix is called* mixing*, or* polarizing*, if it is not upper-triangular under any column permutation.*

In [KSU10] the authors proved that any mixing matrix K polarizes any BMS channel, i.e. the Arıkan's bit-channels (with respect to K) $W_1, W_2, \dots, W_\ell$ start polarizing – some of them become better than W (have smaller entropy), and some become worse. Applying this kernel recursively, similar to the idea of original polar codes, naturally leads to an ℓ-ary tree of bit-channels. The t'th level of the tree corresponds to the linear transformation $K^{\otimes t}$, the t-fold Kronecker product of K.[4]. The results in [KSU10] then actually show the same polarization as in Lemma 4.2 for any mixing matrix G and BMS channel W. Moreover, they also prove sub-exponentially small decay rate of the decoding error probability $P_e < 2^{-N^{E(K)}}$, similarly to Theorem 4.5, but with a matrix-dependent constant $E(K) < 1$ instead of $\beta < 1/2$.

Another way to generalize the problem is to consider coding channels with non-binary input alphabets. It is not surprising that the same polarization ideas and construction work for other alphabet sizes. Initially, in [STA09] the results from [Arı09] and [AT09] were generalized to arbitrary alphabets, with prime-sized alphabets being the most straightforward case, and with certain restrictions for other cases. The construction was further studied under various regimes in subsequent works, but since the focus of this thesis is primarily on the binary-input channels, we omit a detailed discussion here. However, we will review the advances in the scaling exponent for the case of non-binary alphabets in the next section.

[4] As before, the analysis is more convenient if one applies a bit-reversal permutation of the U_i's, and indeed we do so also in our construction, but we skip this detail here for simplicity.

4.3 Prior work

Let us recall the context of Section 1.1, where we explained that the main goal of this dissertation is to find the codes which achieve capacity optimally fast. That is, for a channel W we want to construct the codes for which the gap to capacity $\delta = I(W) - R$ scales almost optimally, as the blocklength N increases (or we can talk about scaling of N with respect to δ, equivalently). Of course, we also want these codes to enjoy low block error probability and have efficient encoding and decoding probability. It is known that the best scaling of blocklength is of the form $N = O(1/\delta^{\mu})$, where μ is called a *scaling exponent*. The optimal scaling exponent is $\mu = 2$, and is achieved by random codes (and even by random linear codes), which is implied by Shannon's noisy coding theorem [Sha48].

4.3.1 Capacity-achieving codes

Before we talk about achieving capacity fast, let us first briefly review what we know about achieving capacity in general. A classical example of a capacity-achieving family is Forney's concatenated codes [For65], however, their decoding complexity scales exponentially in the gap to capacity δ. Turbo codes invention [BGT93] was a breakthrough in practical coding theory, and even though empirically they operate with rates close to capacity, they were not proven to achieve capacity. Low-density parity-check (LDPC) codes, introduced in [Gal65] in the 1960s and later rediscovered [Mac99, Spi96] in the 1990s, also enjoyed a very good practical performance, and eventually were proven to achieve capacity for the binary erasure channel (BEC) in [LMSS98, LMSS01]. However, nothing rigorous was proven for their scaling exponent.

Recently, spatially-coupled LDPC codes were proven to achieve capacity for arbitrary BMS channel in [KRU13] using density evolution and low-complexity message-passing algorithms. A heuristic argument suggests that the scaling exponent for such codes has a value close to 3, however, this has not been rigorously proven and appears to be technically challenging.

In another recent breakthrough [KKM$^+$17] the authors prove that Reed-Muller codes achieve capacity over BEC under the maximum-likelihood decoding, but there are no known bounds on the scaling exponents. In [AY20] it is shown that Reed-Muller codes have polarization property for an arbitrary BMS channel, however, this does not lead (at least yet) to the argument that they achieve capacity.

Finally, there are polar codes, discussed in the next section.

4.3.2 Scaling exponent for polar codes

As we saw in Section 4.1, the original Arıkan's polar codes achieve symmetric capacity for any binary-input discrete memoryless channel, with the block error scaling as $P_e < \exp(-N^{1/2})$ as the blocklength N increases, when the rate $R < I(W)$ is fixed. We also discussed that this was generalized in [KSU10] to larger $\ell \times \ell$ binary kernels with $P_e <$

$\exp(-N^{E(K)})$, where the exponent depends on the kernel K. These results were further generalized for channels with prime power input alphabet size in [MT14].

As for the scaling exponent, empirical bounds for BEC and original polar codes were presented in [KMTU10], suggesting $\mu_{BEC} \approx 3.627$, based on a certain "scaling assumption". The first rigorous proofs of finite scaling exponents for polar codes were (independently) presented in [GX13] and [HAU14]. In [GX13, GX15] the authors showed that there exists a finite μ such that $I(W) - R$ scales as $N^{-1/\mu}$ and $P_e \leq \exp(-N^{0.49})$ at the same time, while also keeping poly(N) construction time for such codes, along with $O(N \log N)$ encoding and decoding complexities, since the proof was for the standard Arıkan's polar codes. No explicit upper bound on μ was provided though, so although convergence with poly$(1/\delta)$ speed was proven, the degree of this polynomial was not determined.

In [HAU14] concrete lower and upper bounds on the scaling exponent μ for standard Arıkan's polar code were presented, for a fixed decoding error probability P_e. The authors proved $3.579 \leq \mu \leq 6$ for any BMS channel, where they relied on studying the largest eigenvalue of a *polar operator* (similarly to [KMTU10]). This idea is still the main tool for getting estimates or bounds on the scaling exponent for polar codes, and we will learn more about it in Section 5.1. The upper bound was then improved to $\mu \leq 5.702$ in [GB14], and further to $\mu \leq 4.714$ for any BMS channel W and to $\mu_{BEC} \leq 3.639$ for the BEC in [MHU16]. In the same paper, the authors discussed the moderate deviations regime, where the codes have both good scaling exponent μ and block error probability $P_e \leq \exp(-N^{\varphi})$, and they showed that any pair of parameters can be achieved if the point $(\varphi, 1/\mu)$ lies under a certain curve connecting points $(0, 1/(\mu^* + 1))$ and $(1/2, 0)$, μ^* being the best provably achievable exponent. This curve was later improved in [WD18a] and [WD18b].

We can already see, however, that the lower bound $3.579 \leq \mu$ shows that the original polar codes (with a 2×2 kernel A_2) fall short of achieving the optimal scaling of $N = O(1/\delta^2)$, at least when the standard successive-cancellation decoding is used. So we need to consider larger kernels if we hope to get this scaling with polar codes.

The results of [GX15] were generalized significantly in [BGN+18], proving that the entire class of polar codes, based on arbitrary $\ell \times \ell$ mixing matrices over any prime field as kernels, has a finite scaling exponent. This was initially done with only inverse-polynomial decoding error probability $P_e = O(N^{-\Omega(1)})$, and was improved to $P_e < \exp(-N^{\varphi})$ for any desired $\varphi < 1$ in [BGS18], however the claim was only that the scaling exponent is finite.

For the binary erasure channel (BEC), several studies showed that larger kernels can benefit the scaling exponent. Binary $\ell \times \ell$ kernels for powers of two $\ell \leq 64$ optimized for the binary erasure channel appear in [MT12, FV14, YFV19]; a 64×64 kernel achieving $\mu < 3$ is reported in [YFV19]. Pfister and Urbanke proved in [PU16] that the optimal scaling exponent $\mu = 2$ can be approached if one considers transmission with $q \times q$ kernels over the q-ary erasure channel, as the input alphabet size q grows.

Finally, Fazeli, Hassani, Mondelli, and Vardy [FHMV17, FHMV18] showed how to achieve any near-optimal scaling exponent $\mu_{BEC} > 2$ for the BEC, using large binary $\ell \times \ell$ kernels. Specifically, they show how to get any scaling exponent $\mu_{BEC} = 2 + \alpha$, for arbitrarily small $\alpha > 0$, using kernel size $\ell \geq \ell_0(\alpha) = \exp\left(\Omega(\alpha^{-1.01})\right)$. The decoding error probability P_e is fixed here. This is a significant result, which constructs, for the first time ever, a family of codes which achieve capacity of a channel near-optimally fast and have low encoding/decoding complexities.

4.4 Current and parallel work

In this section we briefly describe the contribution of this thesis and how it relates to the relevant recent work that came out while this dissertation was in writing. The brevity is because a) our results were discussed in some detail in Section 1.3; and b) they will be described in even more detail and formalized soon in the next chapter.

Our main result can be viewed as an extension to [FHMV18] to the general case of BMS channels. Specifically, in [GRY20] we construct a variant of polar codes which has scaling exponent arbitrarily close to 2, i.e. $\mu = 2 + \alpha$, *for an arbitrary BMS channel* W, also using large $\ell \times \ell$ kernels (the scaling is similar to $\ell_0(\alpha)$ from before). This is a much more general family of coding channels, and the generalization is by no means straightforward. Some of the inherent difficulties when coming from the BEC to BMS channels are described in Sections 5.1.1 and 5.1.2. For the codes in [GRY20] we obtained inverse-polynomial decoding error probability $P_e = O\left(N^{-\Omega(1)}\right)$, polynomial-time construction and $O(N \log N)$ encoding and decoding.

At nearly the same time, Wang and Duursma in [WD19] also presented the construction of polar codes with scaling exponent arbitrarily close to 2, but for even more general settings – they prove it for an arbitrary discrete memoryless channel (DMC). This includes asymmetric channels and arbitrary input alphabet size. Moreover, their result covers moderate deviations regime, where both convergences of R to $I(W)$ and of P_e to 0 are studied at the same time. The authors showed that for each (φ, μ) such that $\varphi + 2/\mu < 1$ (which is the best achievable region for any possible code for channels with positive dispersion V), the polar code with $R \geq I(W) - N^{-1/\mu}$ and $P_e \leq \exp(-N^{\varphi})$ can be constructed. However, to the best of our understanding, the construction complexity is not addressed in [WD19, WD21], and seems to be exponential in N.

In [GRY22] we show that we can apply the convergence analysis from [WD19] to our construction to also get sub-exponentially small $P_e \leq \exp(-N^{\Omega(\alpha)})$ when $\mu = 2 + \alpha$, while keeping polynomial-time construction, for any BMS channel. The same machinery also appeared previously in [WD18a] and [WD18b], see also a dissertation [Wan21]. In this thesis we further show how to use the same approach to get any pair of parameters (φ, μ) which satisfy a certain condition, similar to the curve for moderate deviations regime from [MHU16] and [WD18b], while still preserving poly(N) time construction complexity.

Chapter 5

Near-Optimal Convergence to Capacity

This chapter contains the construction of polar codes with the near-optimal convergence to capacity for any BMS channel. Specifically, we prove

Theorem 5.1. *Let W be an arbitrary BMS channel with Shannon capacity $I(W)$ and fix any $c > 0$. For any desired $\alpha \in \left(0, \frac{1}{12+2c}\right)$, if we choose a large enough constant $\ell \geq \ell_0(\alpha)$ to be a power of 2, then there is a code $\mathcal{C}$ generated by the polar coding construction using kernels of size $\ell \times \ell$ such that the following four properties hold when N is the code length:*

1. *the code construction has $N^{O_\alpha(1)}$ complexity;*
2. *both encoding and decoding have $O_\alpha(N \log N)$ complexity;*
3. *the rate of $\mathcal{C}$ is $I(W) - O(N^{-1/2+(c+6)\alpha})$; and*
4. *the decoding error probability of $\mathcal{C}$ is $O_\alpha(\log N/N^c)$ under successive cancellation decoding when $\mathcal{C}$ is used for channel coding over W.*

The value for $\ell_0(\alpha)$ is the smallest number which satisfies $\log \ell_0 \geq \frac{11}{\alpha}$ and $\frac{\log \ell_0}{\log\log \ell_0 + 2} \geq \frac{3}{\alpha}$, and does not depend on the channel W.

Notice that the above theorem only gets the inverse-polynomial decoding error probability. This will be improved to sub-exponentially small decay in the next chapter.

The notation $O_\alpha(\bullet)$ hides the constant factors which depend only on α and ℓ (since the size ℓ that we must choose only depends on α itself, we ignore it in this notation). Note that these constants might be arbitrarily large in terms of α, but are still constants, and treated as such.

A similar lower bound on the required kernel size ℓ also appears in [FHMV17], where polar codes with the near-optimal convergence to channel capacity for the BEC are constructed. Due to this requirement of extremely large ℓ, we want to point out that this result is primarily theoretical in nature, and meant to illustrate that the polar coding framework is powerful enough to achieve an asymptotically optimal rate of convergence to Shannon capacity with efficient algorithms. The choice of (very) large constants which is required to prove this convergence makes this construction unsuitable for practical applications. To convince the reader even more and avoid the attempts to try this at home, the constants in the theorem hide 2^ℓ factors, since ℓ is treated as a constant, which means it actually hides doubly-exponential in $1/\alpha$ factors, i.e. $\exp(2^{1/\alpha})$. So we are paying a stiff (but constant in terms of N) price if we want to get the codes with near-optimal scaling exponent.

The rest of the chapter is structured as follows. In Section 5.1 we give an overview of how our construction works. We start by explaining the standard (in the polar coding literature) way to obtain bounds on the scaling exponents, and then proceed with explaining the ideas we introduce to get the scaling exponent arbitrarily close to 2. After that, in Sections 5.2-5.5 we show our construction and prove that is works.

5.1 Approach overview

5.1.1 Analysis of polarization via recursive potential function

For any fixed BMS channel W, recall the random process of bit-channels W_t defined for recursive polar codes construction in Section 4.1.4. We defined it as a random walk down the binary tree for the standard kernel A_2; however, it is defined in an identical way for $\ell \times \ell$ kernels by walking down the ℓ-ary tree. That is, define $\mathsf{W}_0 = W$ and $\mathsf{W}_{j+1} = (\mathsf{W}_j)_i$ for i uniformly chosen from $[\ell]$, where $(\mathsf{W}_j)_i$ is the i^{th} Arıkan's bit-channel of W_j with respect to the kernel K. On the tree of channels, where each node W' has exactly ℓ children marked as Arıkan's bit channels $(W')_1, (W')_2, \dots, (W')_\ell$, this corresponds to starting with $\mathsf{W}_0 = W$ and walking down to a uniformly random child of the current node at each step. The entropy and Bhattacharyya processes are defined similarly as $\mathsf{H}_j \coloneqq H(\mathsf{W}_j)$ and $\mathsf{Z}_j \coloneqq Z(\mathsf{W}_j)$. Since every kernel in the construction is chosen to be invertible, H_j is a martingale due to the conservation of entropy property (4.10). It is clear that W_j marginally is a uniformly random channel of the j^{th} level of the channel tree, and then H_j marginally is the entropy of such a randomly chosen channel.

The principle behind polarization is that for large enough t, almost all of the channels on the t-th level of the channel tree will be close to either a useless or a noiseless channel, i.e. their entropy will be very close to 1 or 0. To estimate how fast such polarization actually happens, one needs to bound the fraction of unpolarized channels (which have entropies not close to neither 0 nor 1) on the t-th level, i.e., $\delta_t = \mathbb{P}\Big[\mathsf{H}_t \in (\zeta_t, 1-\zeta_t)\Big]$ for some tiny threshold ζ_t, and show that this fraction vanishes rapidly with increasing t. Notice that this is essentially what we did in Section 4.1.5 for the A_2 kernel case, however, we looked at Z_t instead and considered the event that it is small, but it is not hard to see that one can switch back and forth between these two arguments. We also were only interested in $\delta_t \to 0$, without considering its speed, as we merely wanted to obtain capacity-achieving property.

With this notation for the fraction of unpolarized codes, one (roughly) gets codes of length ℓ^t, rate $I(W) - \delta_t - \zeta_t$, for which the SC decoder achieves decoding error probability $\zeta_t \ell^t$ for communication over W. While the gap to capacity is (roughly) $\delta_t + \zeta_t$, the quantity ζ_t is not *that* hard to get to be very small (we already know that we can get it sub-exponentially small in $N = \ell^t$ for some cases), so the fraction of unpolarized channels δ_t is what actually governs the gap to capacity of polar codes. Therefore, one would need to show $\delta_t \leq O(\ell^{-t/\mu})$ to establish a scaling exponent of μ, since the blocklength is $N = \ell^t$.

Being more precise, we have the following result (stated explicitly in [BGN+18, Theorem A.3]): if for all t

$$\mathbb{P}[\mathsf{H}_t \in (p\ell^{-t}, 1 - p\ell^{-t})] \leq D \cdot \beta^t, \tag{5.1}$$

then this corresponds to a polar code with block length $N = \ell^t$, rate $(D \cdot \beta^t + p\ell^{-t})$-close to the capacity of the channel, and decoding error probability at most p under the successive cancellation decoder. The reader should think of p as being inverse polynomial (of fixed degree) in N for discussion in this chapter.

To track the fractions of polarized and non-polarized channels at each level of the construction, we use a potential function on channel entropy

$$g_\alpha(h) = (h(1-h))^\alpha, \tag{5.2}$$

where $\alpha > 0$ is some small fixed parameter. This α corresponds to the gap to the scaling exponent in Theorem 5.1, and in this document we always consider $\alpha < \frac{1}{12}$ (and smaller bounds in some cases). Such a potential function was also used for example in [MHU16] and [FHMV17], and in general the use of some kind of concave potential function on channel parameters was used since [KMTU10] to obtain all known results for scaling exponents, to the best of our knowledge. Such a potential function just needs to "punish" the channels when they have entropy far from the endpoints on $[0, 1]$ interval. We are going to track the expected value $\mathbb{E}[g_\alpha(\mathsf{H}_t)]$ as t increases, and then use Markov's inequality to get

$$\mathbb{P}[\mathsf{H}_t \in (p\ell^{-t}, 1 - p\ell^{-t})] = \mathbb{P}[g_\alpha(\mathsf{H}_t) \geq g_\alpha(p\ell^{-t})] \leq \frac{\mathbb{E}[g_\alpha(\mathsf{H}_t)]}{g_\alpha(p\ell^{-t})} \leq 2 \left(\ell^t/p\right)^\alpha \cdot \mathbb{E}[g_\alpha(\mathsf{H}_t)]. \tag{5.3}$$

To give an upper bound on $\mathbb{E}[g_\alpha(\mathsf{H}_t)]$, we desire to prove that the average of the potential function of all the children of any channel in the tree decreases significantly with respect to the potential function of this channel. Specifically, we want to obtain for some small λ_α and any channel W' in the bit-channel tree the inequality

$$\mathop{\mathbb{E}}_{i \sim [\ell]} \left[g_\alpha\left(H(W_i')\right) \right] \leq \lambda_\alpha \cdot g_\alpha\left(H(W')\right), \tag{5.4}$$

where W_i' are the children of W' in the construction tree for $i \in [\ell]$, and the constant λ_α should only depends on the potential function g_α (in our particular case (5.2), just on α) and ℓ, but should be universal for all the channels in the tree. If one can guarantee that (5.4) holds throughout the construction process, then for the martingale process H_t

obtain

$$\begin{aligned}\mathbb{E}\left[g_\alpha\left(\mathsf{H}_t\right)\right] &= \mathbb{E}\left[\mathop{\mathbb{E}}_{j\sim[\ell]}\left[g_\alpha\left(H\left((\mathsf{W}_{t-1})_j\right)\right)\right]\,\middle|\,\mathsf{W}_{t-1}\right] \\ &= \mathbb{E}\left[\frac{1}{\ell}\frac{\sum_{j=1}^{\ell} g_\alpha\left(H\left((\mathsf{W}_{t-1})_j\right)\right)}{g_\alpha\left(H(\mathsf{W}_{t-1})\right)}\cdot g_\alpha\left(H(\mathsf{W}_{t-1})\right)\,\middle|\,\mathsf{W}_{t-1}\right] \\ &\le \lambda_\alpha\cdot\mathbb{E}\left[g_\alpha\left(\mathsf{H}_{t-1}\right)\right],\end{aligned}\tag{5.5}$$

and inductively

$$\mathbb{E}\left[g_\alpha\left(\mathsf{H}_t\right)\right] \le \lambda_\alpha\cdot\mathbb{E}\left[g_\alpha\left(\mathsf{H}_{t-1}\right)\right] \le \lambda_\alpha^2\cdot\mathbb{E}\left[g_\alpha\left(\mathsf{H}_{t-2}\right)\right] \le \cdots \le \lambda_\alpha^t\mathsf{H}_0 \le \lambda_\alpha^t.\tag{5.6}$$

Then (5.3) and (5.1) imply the existence of codes with rate $O\left((N/p)^\alpha\cdot\lambda_\alpha^t\right)$-close to capacity of the channel. It is then clear that estimating λ_α, which one can represent as $\lambda_\alpha = \ell^{-1/\mu}$, directly leads to statements about the scaling exponents. Since our main task is to achieve a gap which is close to $N^{-1/2} = \ell^{-t/2}$, we want to construct the codes in such a way that we can prove multiplicative decrease (5.4) for some $\lambda_\alpha \approx \ell^{-1/2}$ at each step t.

This is exactly what was achieved in [FHMV17] for the binary erasure channel (BEC) case. The polarization process for BEC has a particularly nice structure. If the initial channel W is the binary erasure channel with erasure probability z (denoted BEC(z)), then the Arıkan's bit-channels W_i, $i \in [\ell]$, with respect to any kernel K are also binary erasure channels. Moreover, they can be written as BEC($p_i^{(K)}(z)$), where $p_i^{(K)}(\cdot)$ are some polynomials of degree at most ℓ, which are fully determined by the choice of K. Crucially, *all* the channels in the recursive tree remain BEC. Therefore it suffices to prove the existence of a good polarizing kernel for the class of binary erasure channels, which is parameterized by a single number, the erasure probability, which also equals the entropy of the channel. In [FHMV17] it is proven that a random kernel works with good probability for all BEC universally. Fundamentally the calculations for BEC revolve around the rank of various random subspaces, as decoding under the BEC is a linear-algebraic task. Moving beyond the BEC takes us outside the realm of linear algebra into information-theoretic settings where tight quantitative results are much harder to establish.

5.1.2 The road to BMS channels: using multiple kernels

For the case when the initial channel W is a BSC, a fundamental difficulty (among others) is that the channels in the recursion tree will no longer remain BSC (even after the first step). Further, to the best of our knowledge, the various channels that arise do not share a nice common exploitable structure. Therefore, we have to think of the intermediate channels as arbitrary BMS channels, a very large and diverse class of channels. It is not clear (to us) if there exists a single kernel K to universally polarize *all* BMS channels at a rapid rate, i.e. for which we can prove (5.4) for small λ_α *universally for all* BMS channels W', where W'_i are Arıkan's bit-channels of W' w.r.t. K. Even if such a kernel exists, proving so seems out of reach of current techniques. Finally, even for a specific BMS, proving that a random kernel polarizes it fast enough requires very strong quantitative bounds on the performance and limitations of random linear codes for channel coding.

Since we are not able to establish that a single kernel would work for the whole construction universally, the idea behind our codes is to use different kernels, which are picked individually for each bit-channel that we face in the channel tree. That way, by choosing a suitable kernel for each channel in the tree, we can ensure that polarization is fast enough throughout the whole process. Notice that while we required the decay rate λ_α in (5.4) to be fixed for the whole process (tree), we did not say that the kernel with respect to which the Arıkan's transformation is taken should be the same. As long as (5.4) holds at each node of the channel tree, the speed of convergence to capacity follows. The idea of using several distinct internal kernels (it is called *dynamic kernels* in some places) is not new and previously appeared in [YB15, PSL15, GBLB17, BBGL17] in some variations, but was not used to improve the scaling exponent of polar codes before [WD19] and our result in [GRY20]. Below we describe the mixed-kernel construction slightly more formally.

Though we use different kernels in the code construction, all of them have the same size $\ell \times \ell$ (mixed-sized constructions also do enjoy polarization behavior). We say that a kernel is *good* if all but a $\widetilde{O}(\ell^{-1/2})$ fraction of the bit-channels obtained after polar transform by this kernel have entropy $\ell^{-\Omega(\log \ell)}$-close to either 0 or 1. Given a BMS channel W, the code construction consists of t steps, from Step 0 to Step $t-1$. At Step 0, we find a good kernel $K_1^{(0)}$ for the original channel W. After the polar transform of W using kernel $K_1^{(0)}$, we obtain ℓ bit-channels $W_1, \dots, W_\ell$. Then in Step 1, we find good kernels for each of these ℓ bit-channels. More precisely, the good kernel for W_i is denoted as $K_i^{(1)}$, and the bit-channels obtained by polar transforms of W_i using kernel $K_i^{(1)}$ are denoted as $\{W_{i,j} : j \in [\ell]\}$; see Figure 5.1 for an illustration. At Step j, we will have ℓ^j bit-channels $\{W_{i_1,\dots,i_j} : i_1, \dots, i_j \in [\ell]\}$. For each of them, we find a good kernel $K_{i_1,\dots,i_j}^{(j)}$. After polar transform of $\{W_{i_1,\dots,i_j} : i_1, \dots, i_j \in [\ell]\}$ using these kernels, we will obtain ℓ^{j+1} bit-channels. Finally, after the last step (Step $t-1$), we will obtain $N = \ell^t$ bit-channels $\{W_{i_1,\dots,i_t} : i_1, \dots, i_t \in [\ell]\}$. Using the good kernels we obtained in this code construction procedure, we can build an $N \times N$ matrix (or we can view it as a large single kernel) $M^{(t)}$ such that the N bit-channels resulting from the polar transform of the original channel W using this large kernel $M^{(t)}$ are exactly $\{W_{i_1,\dots,i_t} : i_1, \dots, i_t \in [\ell]\}$. We will say a few more words about this in Section 5.1.4 and provide all the details in Section 5.4.

5.1.3 Sharp transition in polarization

The main technical challenge then consists in showing that if ℓ is large enough, it is possible to choose kernels in the construction process for which λ_α is close to $\ell^{-1/2}$. Specifically, we prove that if ℓ is a power of 2 such that $\log \ell = \Omega\left(\frac{1}{\alpha^{1.01}}\right)$, then it is possible to achieve

$$\lambda_\alpha \le \ell^{-1/2+5\alpha} \tag{5.7}$$

by individually choosing suitable kernels in the code construction. To obtain such behavior, while choosing the kernel for the current channel W' during the recursive process we

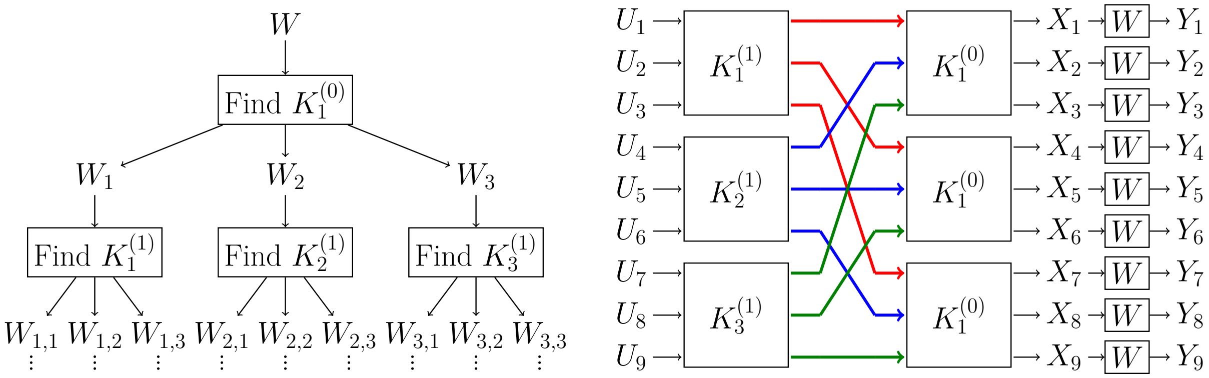

W
Find $K_1^{(0)}$
W_1
W_2
W_3
Find $K_1^{(1)}$
Find $K_2^{(1)}$
Find $K_3^{(1)}$
$W_{1,1}$ $W_{1,2}$ $W_{1,3}$ $W_{2,1}$ $W_{2,2}$ $W_{2,3}$ $W_{3,1}$ $W_{3,2}$ $W_{3,3}$
U_1 U_2 U_3 U_4 U_5 U_6 U_7 U_8 U_9
$K_1^{(1)}$
$K_2^{(1)}$
$K_3^{(1)}$
$K_1^{(0)}$
$K_1^{(0)}$
$K_1^{(0)}$
X_1 X_2 X_3 X_4 X_5 X_6 X_7 X_8 X_9
W
Y_1 Y_2 Y_3 Y_4 Y_5 Y_6 Y_7 Y_8 Y_9

Figure 5.1: The left figure illustrates the code construction, and the right figure shows the encoding procedure for the special case of $\ell = 3$ and $t = 2$. All the kernels in this figure have size 3×3. One can show that the bit-channel $W_{i,j}$ in the left figure is exactly the channel mapping from $U_{3(i-1)+j}$ to $(\boldsymbol{U}_{[1:3(i-1)+j-1]}, \mathbf{Y}_{[1:9]})$ in the right figure.

distinguish two cases:

Case 1: W' is already very noisy or almost noiseless. We call this regime *suction at the ends* (following [BGN$^+$18]), and it is well understood that polarization happens very fast for this case, even for standard choice of a kernel. So in this case we take a power of standard Arıkan's polarization kernel $K = \left[\begin{smallmatrix}1&0\\1&1\end{smallmatrix}\right]^{\otimes \log \ell}$ and prove (5.4) with a geometric decrease factor $\lambda_\alpha \le \ell^{-1/2}$.

Case 2: W' is neither very noisy nor almost noiseless. We refer to this case as *variance in the middle* regime (following [BGN$^+$18] again). For such a channel we adopt the framework from [FHMV17] and show a *sharp transition in polarization* for a random kernel K, with respect to W'. Specifically, we prove that with high probability over $K \sim \{0,1\}^{\ell\times\ell}$ (for ℓ large enough) it holds

$$\begin{aligned} H(W_i'(K)) &\le \ell^{-\Omega(\log \ell)} &&\text{for} \quad i \ge \ell \cdot H(W') + \Omega(\ell^{1/2}\log^3 \ell), \\ H(W_i'(K)) &\ge 1 - \ell^{-\Omega(\log \ell)} &&\text{for} \quad i \le \ell \cdot H(W') - \Omega(\ell^{1/2}\log^3 \ell). \end{aligned} \tag{5.8}$$

It then follows that only about $\widetilde{O}(\ell^{-1/2})$ fraction of bit-channels are not polarized for some kernel K, which then easily translates into the bound (5.7) on λ_α that we desire. Note that we can always ensure that we take an invertible kernel K since a random binary matrix is invertible with at least some constant probability.

Proving such a sharp transition constitutes the bulk of the technical work for our construction, however, the majority of it was already done in Chapter 3. This is because in Section 5.2.2 we show that inequalities in (5.8) correspond to decoding a single bit of a message which is transmitted through W' using a random linear code, which is exactly the setting for the strong converse theorem we proved. The first set of inequalities in (5.8) correspond to saying that one can decode this single bit with low error probability with high probability over the randomness of the code, if the rate of the code is at least $\widetilde{O}(\ell^{-1/2})$

smaller than the capacity of the channel (where ℓ is the blocklength of the code). This follows from the well-studied fact that random linear codes achieve Shannon's capacity over any BMS channel ([Gal65, BF02]).

The second set of inequalities, on the other hand, corresponds to saying that for a random linear code with rate exceeding capacity by at least $\widetilde{O}(\ell^{-1/2})$, even predicting a single bit of the message with a tiny advantage over a uniform guess is not possible. This is exactly what we proved in Theorem 3.1.

5.1.4 Construction, encoding and decoding

Once we have obtained the kernels in the code construction, the encoding procedure is pretty standard; see [PSL15, YB15, GBLB17, BBGL17, WD18a] for discussions on polar codes using multiple kernels. As mentioned in Section 5.1.2, we can build an $N \times N$ matrix $M^{(t)} := D^{(t-1)}Q^{(t-1)}D^{(t-2)}Q^{(t-2)} \dots D^{(1)}Q^{(1)}D^{(0)}$, where the matrices $Q^{(1)}, Q^{(2)}, \dots, Q^{(t-1)}$ are some permutation matrices, and $D^{(0)}, D^{(1)}, \dots, D^{(t-1)}$ are block diagonal matrices. In particular, all the blocks on the diagonal of $D^{(j)}$ are the kernels that we obtained in Step j of the code construction. We illustrate the special case of $\ell = 3$ and $t = 2$ in Figure 5.1. We take a random vector $\boldsymbol{U}_{[1:N]}$ consisting of $N = \ell^t$ i.i.d. Bernoulli-1/2 random variables and we transmit the random vector $\boldsymbol{X}_{[1:N]}$ through N independent copies of W. Denote the output vector as $\boldsymbol{Y}_{[1:N]}$. Then for every $i \in [N]$, the bit-channel mapping from U_i to $(\boldsymbol{U}_{[1:i-1]}, \boldsymbol{Y}_{[1:N]})$ is exactly $W_{i_1,\dots,i_t}$, where $(i_1, \dots, i_t)$ is ℓ-ary expansion of i.

We have shown that almost all of the N bit-channels $\{W_{i_1,\dots,i_t} : i_1, \dots, i_t \in [\ell]\}$ become either noiseless or completely noisy. In the code construction, we can track $H(W_{i_1,\dots,i_t})$ for every $(i_1, \dots, i_t) \in [\ell]^t$, and this allows us to identify which U_i's are noiseless under successive decoding. Then in the encoding procedure, we only put information in these noiseless U_i's and set all the other U_i's to be some "frozen" value, e.g., 0. This is equivalent to saying that the generator matrix of our code is the submatrix of $M^{(t)}$ consisting of rows corresponding to indices i of the noiseless U_i's. In Section 5.4, we will show that similarly to the classical polar codes, both the encoding and decoding of our code also have $O(N \log N)$ complexity.

To achieve polynomial-time construction for the codes, we need to quantize every bit-channel we obtain in every step of the code construction. The idea was used in [TV13] and subsequent polar codes papers in which careful estimation of construction complexity was present. More precisely, we merge the output symbols whose log-likelihood ratios are close to each other, so that after the quantization, the output alphabet size of every bit-channel is always polynomial in N. This allows us to construct the code in polynomial time. Without quantization, the output alphabet size would eventually be exponential in N. However, notice that approximating the channels in such a way can only give us inverse-polynomial approximations on channel parameters, and thus on decoding error probability. We provide more details about this aspect, and how it affects the code construction and the analysis of decoding, in Section 5.2.1 and Section 5.4. This problem is further addressed (and is the main focus of) Chapter 6.

5.2 Give me a channel, I'll give you a kernel

In this section we prove that a suitable kernel can be chosen for any bit-channel which appears in the channel tree during the construction. Specifically, we show that for any given binary-input memoryless symmetric (BMS) channel W we can find a kernel K of size $\ell \times \ell$, such that the Arıkan bit-channels of W with respect to this kernel will be highly polarized. By this we mean that the multiplicative decrease λ_α defined in (5.4) will be sufficiently close to $\ell^{-1/2}$. The algorithm (Algorithm A) to find such a kernel is as follows: if the channel is already almost noiseless or too noisy (entropy is very close to 0 or 1), we take this kernel to be a tensor power of original Arıkan's kernel for polar codes, $A_2 = \left[\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right]$. Otherwise, the algorithm will just try out all the possible invertible kernels in $\{0,1\}^{\ell\times\ell}$, until a "good" kernel is found, which means that conditions (5.9) should be satisfied. Before proving that Algorithm A achieves our goals of bringing λ_α close to $\ell^{-1/2}$, we discuss several details about it.

5.2.1 Local kernel construction

Algorithm A: Kernel search

```
Input: BMS channel $\widetilde{W}$ with output size $\le \mathsf{Q}$, error parameter $\Delta$, and number $\ell$
Output: invertible kernel $K \in \{0,1\}^{\ell\times\ell}$
if $H(\widetilde{W}) < \ell^{-4}$ or $H(\widetilde{W}) > 1 - \ell^{-4} + \Delta$ then
    return $K = A_2^{\otimes \log \ell}$
else
    for $K \in \{0,1\}^{\ell\times\ell}$, if $K$ is invertible do
        Compute Arıkan's bit-channels $\widetilde{W}_i(K)$ of $\widetilde{W}$ with respect to the kernel $K$,
         as in (4.9)
        if

            $H(\widetilde{W}_i(K)) \le \ell^{-(\log \ell)/4}$        for  $i \ge \ell \cdot H(\widetilde{W}) + \ell^{1/2} \log^3 \ell$
                                                                                              (5.9)
            $H(\widetilde{W}_i(K)) \ge 1 - \ell^{-(\log \ell)/20}$   for  $i \le \ell \cdot H(\widetilde{W}) - 14\ell^{1/2} \log^3 \ell$

         then
            return $K$
        end
    end
end
```

As briefly discussed at the end of Section 5.1.4, we are unable to efficiently track all the bit-channels in the ℓ-ary recursive tree *exactly*. This is because the size of the output alphabet of the channels increases *exponentially* after each step deeper into the tree (this simply follows from the definition of bit-channels (4.9)). Thus computing all the channels (and their entropies) cannot be done in poly(N) time. To overcome this issue we follow the approach of [TV13], with subsequent simplification in [GX15], of approximating the channels in the tree by degrading (see Definition 2.9) them. Degradation is achieved via the procedure of merging the output symbols, which (a) decreases the output alphabet size, and (b) does not change the entropy of the channel too much. This implies (with all the details worked out in Section 5.4) that we can substitute all the channels in the tree of depth t by their *degraded approximations*, such that all the channels have output alphabet size at most Q (a parameter depending on $N = \ell^t$ to be chosen), and that if $\widetilde{W}$ is a degraded approximation of the channel W in the tree, then $H(W) \le H(\widetilde{W}) \le H(W) + \Delta$ for some Δ depending on Q. Moreover, in Theorem 5.2 which we formulate and prove shortly, we show that when we apply Algorithm A to a degraded approximation $\widetilde{W}$ of W with small enough Δ, then, even though conditions (5.9) only dictate a sharp transition for $\widetilde{W}$, the same kernel will induce a sharp transition in polarization for W.

The second issue which such degraded approximation resolves is the running time of Algorithm A. Notice that we are only going to apply it for channels with output size bounded by Q, and recall ℓ is treated as a constant (though very large). First of all, trying out all the possible kernels will then also take a constant number of iterations. Finally, within each iteration, calculating all the Arıkan's bit-channels and their entropies in a straightforward way will take $\text{poly}(\mathsf{Q}^\ell)$ time, which is $\text{poly}(\mathsf{Q})$ since ℓ is a constant. Therefore by choosing Q to be polynomial in N, the algorithm indeed works in $\text{poly}(N)$ time.

We now leave the full details concerning the complexity of the algorithm to be handled in Section 5.4, and proceed with showing that Algorithm A always returns a kernel which makes λ_α from (5.4) close to $\ell^{-1/2}$.

Theorem 5.2. *Let $\alpha \in \left(0, \frac{1}{12}\right)$ be a small fixed constant. Let ℓ be an even power of 2 such that $\log \ell \ge \frac{11}{\alpha}$ and $\frac{\log \ell}{\log\log \ell + 2} \ge \frac{3}{\alpha}$. Let $W : \{0,1\} \to \mathcal{Y}$ and $\widetilde{W} : \{0,1\} \to \widetilde{\mathcal{Y}}$ be two BMS channels, such that $\widetilde{W} \preceq W$, $H(\widetilde{W}) - \Delta \le H(W) \le H(\widetilde{W})$ for some $0 \le \Delta \le \ell^{-\log \ell}$, and $|\widetilde{\mathcal{Y}}| \le \mathsf{Q}$. Then Algorithm A on inputs $\widetilde{W}$, Δ, and ℓ returns a kernel $K \in \{0,1\}^{\ell \times \ell}$ that satisfies*

$$\frac{1}{\ell \cdot g_\alpha(H(W))} \sum_{i=1}^{\ell} g_\alpha\left(H(W_i)\right) \le \ell^{-\frac{1}{2}+5\alpha}, \tag{5.10}$$

where $W_1, W_2, \ldots, W_\ell$ are the Arıkan's bit-channels of W with respect to the kernel K, and $g_\alpha(\cdot)$ is the potential function $g_\alpha(h) = (h(1-h))^\alpha$ for any $h \in [0,1]$, as defined in (5.2).

Proof. As we discussed above, we consider two cases:

Suction at the ends. If $H(\widetilde{W}) \notin (\ell^{-4}, 1 - \ell^{-4} + \Delta)$, Algorithm A returns a standard Arıkan's kernel $K = A_2^{\otimes \log \ell}$ on input $\widetilde{W}$ and Δ. For this case $H(W) \notin (\ell^{-4}, 1 - \ell^{-4})$, and fairly standard arguments imply that the polarization under such a kernel is much faster when the entropy is close to 0 or 1. For completeness, we present the full proofs for this case in a deferred Section 5.3. Specifically, Lemma 5.10 immediately implies the result of the theorem for this regime, as we pick $\log \ell \ge \frac{1}{\alpha}$.

Variance in the middle. Otherwise, if $H(\widetilde{W}) \in (\ell^{-4}, 1-\ell^{-4}+\Delta)$, it holds $H(W) \in (\ell^{-4}-\Delta, 1-\ell^{-4}+\Delta)$, thus $H(W) \in (\ell^{-4}/2, 1-\ell^{-4}/2)$ since $0 \le \Delta \le \ell^{-\log \ell}$ and $\log \ell$ is large by the conditions of the theorem.

We first need to argue that the algorithm will at least return some kernel. We formulate this as a separate Theorem 5.5 in Section 5.2.2. The theorem essentially claims that for any $\widetilde{W}$ an overwhelming fraction of possible kernels $K \in \{0,1\}^{\ell\times\ell}$ satisfies the conditions in (5.9) for $\widetilde{W}$ and K (note that we do not use any conditions on the size of $\widetilde{\mathcal{Y}}$ or the entropy $H(\widetilde{W})$ at all at this point). Clearly then, there is a decent fraction of *invertible* kernels from $\{0,1\}^{\ell\times\ell}$ which also satisfy these conditions. Therefore, the algorithm will indeed terminate and return such a good kernel. Moreover, the theorem claims that a random kernel from $\{0,1\}^{\ell\times\ell}$ will satisfy (5.9) with high probability, and it is also known that it will be invertible with at least some constant probability. It means that instead of iterating through all possible kernels in step 4 of Algorithm A, we could take a random kernel and check it, and then the number of iterations needed to find a good kernel would be very small with high probability. In four words, *random dynamic kernels work*. However, to keep everything deterministic, we stick to the current approach.

Suppose now the algorithm returned an invertible kernel $K \in \{0,1\}^{\ell\times\ell}$, which means that relations (5.9) hold for $\widetilde{W}$ and Arıkan's bit-channels $\widetilde{W}_1, \widetilde{W}_2, \dots, \widetilde{W}_\ell$ (we omit dependence on K from now on). Denote also $W_i = W_i(K)$ as Arıkan's bit-channels of W with respect to K. Now we use the property that degradation is preserved under Arıkan's transformation:

Proposition 5.3 ([YB15, Lemma IV.1])**.** *Let W and $\widetilde{W}$ be BMS channels, such that $\widetilde{W} \preceq W$, and $K \in \{0,1\}^{\ell\times\ell}$ be any invertible matrix. Denote by W_i, $\widetilde{W}_i$ the Arıkan's bit-channels of W and $\widetilde{W}$ with respect to the kernel K for any $i \in [\ell]$. Then for any $i \in [\ell]$, we have $\widetilde{W}_i \preceq W_i$, and consequently $H(\widetilde{W}_i) \ge H(W_i)$.*

This fact was proved in [KU10, Lemma 21] for the special case of Arıkan's kernel and then generalized in [YB15, Lemma IV.1] to general kernels. Apply this directly to our situation, thus $H(W_i) \le H(\widetilde{W}_i)$ for all $i \in [\ell]$. Now, since K is invertible, conservation of entropy implies $\sum_{i=1}^{\ell}\left(H(\widetilde{W}_i) - H(W_i)\right) = \ell\left(H(\widetilde{W}) - H(W)\right) \le \ell\cdot\Delta$, therefore derive $H(W_i) \le H(\widetilde{W}_i) \le H(W_i) + \ell\cdot\Delta$ for any $i \in [\ell]$. Then deduce from (5.9)

$$\begin{aligned} &H(W_i) \le H(\widetilde{W}_i) \le \ell^{-(\log \ell)/4} && \text{for} \quad i \ge \ell\cdot H(\widetilde{W}) + \ell^{1/2}\log^3 \ell \\ &H(W_i) \ge H(\widetilde{W}_i) - \ell\cdot\Delta \ge 1 - \ell^{-(\log\ell)/21} && \text{for} \quad i \le \ell\cdot H(\widetilde{W}) - 14\cdot \ell^{1/2}\log^3 \ell, \end{aligned} \tag{5.11}$$

where we used that $\Delta \le \ell^{-\log \ell}$ and ℓ is large in the condition of the theorem.

Recall that $H(W) \in (\ell^{-4}/2, 1-\ell^{-4}/2)$ for variance in the middle regime, and note that this implies

$$g_\alpha(H(W)) \ge g_\alpha(\ell^{-4}/2) = \left(\frac{1}{2}\cdot(1-\ell^{-4}/2)\right)^\alpha \cdot \ell^{-4\alpha} \ge \left(\frac{1}{4}\right)^\alpha \ell^{-4\alpha} \ge \frac{1}{2}\ell^{-4\alpha}, \tag{5.12}$$

since g_α is increasing on $(0, 1/2)$ and $\alpha < 1/2$. Using (5.11) and the trivial bound $g_\alpha(x) \le 1$ for all the indices i close to $\ell \cdot H(\widetilde{W})$ obtain that the LHS of the desired inequality (5.10) is at most

$$\begin{aligned}
&\frac{1}{\ell \cdot g_\alpha(H(W))}\Bigg(\sum_{i=1}^{\ell \cdot H(\widetilde{W}) - 14\cdot \ell^{1/2}\log^3 \ell} g_\alpha\left(1 - \ell^{-(\log \ell)/21}\right) + 15\ell^{1/2}\log^3 \ell \\
&\qquad\qquad + \sum_{i=\ell\cdot H(\widetilde{W}) + \ell^{1/2}\log^3\ell}^{\ell} g_\alpha\left(\ell^{-(\log\ell)/4}\right)\Bigg) \\
&\qquad \overset{(a)}{<} 2\ell^{4\alpha-1}\left(15\ell^{1/2}\log^3\ell + \ell\cdot H(\widetilde{W})\cdot \ell^{-(\alpha\log\ell)/21} + (\ell - \ell\cdot H(\widetilde{W}))\cdot \ell^{-(\alpha\log\ell)/4}\right) \\
&\qquad < 30\ell^{-\frac{1}{2}+4\alpha}\log^3\ell + 2\ell^{-(\alpha\log\ell)/21+4\alpha} \\
&\qquad \overset{(b)}{\le} \ell^{-\frac{1}{2}+4\alpha}\left(30\log^3\ell + 2\ell^{-1/42}\right) < \ell^{-\frac{1}{2}+4\alpha}\cdot 32\log^3\ell \\
&\qquad \overset{(c)}{\le} \ell^{-\frac{1}{2}+5\alpha},
\end{aligned}$$

where (a) follows from (5.12) and the fact that $g_\alpha(x) = g_\alpha(1-x) \le x^\alpha$ for $x \in (0,1)$; (b) uses the condition $\log \ell \ge \dfrac{11}{\alpha}$, and (c) uses $\dfrac{\log\ell}{\log\log\ell + 2} \ge \dfrac{3}{\alpha}$ from the requirements that we have on ℓ in the conditions of this theorem. □

Remark 5.4. *We are mostly interested in the cases where α is close to* 0*, as our goal is to approach the optimal scaling exponent* 2*. For such α, we can absorb the two conditions on ℓ in Theorem* 5.2 *into one condition* $\log\ell \ge \Omega(\alpha^{-1.01})$ *for convenience of notation.*

5.2.2 Strong channel coding and converse theorems

In this section we will show that Algorithm A, which is used to prove the multiplicative decrease of almost $\ell^{-1/2}$ as in (5.10) in the settings of Theorem 5.2, indeed always returns some kernel. While the analysis of suction at the ends regime, deferred to Section 5.3, follows standard methods in the literature and only relies on the fact that polarization becomes much faster when the channel is noiseless or useless, in this section we follow the ideas from [FHMV17] and prove a *sharp transition in the polarization behavior*, when we use a random and sufficiently large kernel.

The sharp transition stems from the fact that when the kernel K is large enough, with high probability (over the randomness of K) all the Arıkan's bit-channel with respect to K, except for approximately $\ell^{1/2}$ of them in the middle, are guaranteed to be either very noisy or almost noiseless. We formulate the following theorem, which was used in the proof of Theorem 5.2:

Theorem 5.5. *Let W be any BMS channel. Let $W_1, W_2, \dots, W_\ell$ be the Arıkan's bit-channels defined in* (4.9) *with respect to the kernel K chosen uniformly at random from $\{0,1\}^{\ell\times\ell}$, where ℓ is a large integer such that* $\log\ell > 40$*. Then for the following inequalities all hold with probability* $(1 - o_\ell(1))$ *over the choice of K:*

(a) $H(W_i) \leq \ell^{-(\log \ell)/4}$ *for* $i \geq \ell \cdot H(W) + \ell^{1/2} \log^3 \ell$;

(b) $H(W_i) \geq 1 - \ell^{-(\log \ell)/20}$ *for* $i \leq \ell \cdot H(W) - 14 \cdot \ell^{1/2} \log^3 \ell$.

Remark 5.6. *One can notice that the above theorem is stated for any BMS channel W, independent of the value of $H(W)$. So in terms of sharp polarization, a random kernel also works for the suction at the ends regime. The technical difficulty though arises because we need the lower bound* (5.12) *on $g_\alpha(H(W))$ in order to get the desired inequality* (5.10)*, which is why we distinguish the suction at the ends regime separately.*

The proof of this theorem relies on the strong converse for bit-decoding which we proved in Chapter 3. The following proposition shows how to connect Arıkan's bit-channels to this context.

Proposition 5.7. *Let W be a BMS channel, $K \in \{0,1\}^{\ell \times \ell}$ be an invertible matrix, and $i \in [\ell]$. Set $k = \ell - i + 1$, and let G be a matrix which is formed by the last k rows of K. Let $\boldsymbol{U}$ be a random vector uniformly distributed over $\{0,1\}^\ell$, and $\boldsymbol{V}$ be a random vector uniformly distributed over $\{0,1\}^k$. Then*

$$H\left(U_i \,\middle|\, W^\ell(\boldsymbol{U} \cdot K), \boldsymbol{U}_{<i}\right) = H\left(V_1 \,\middle|\, W^\ell(\boldsymbol{V} \cdot G)\right). \tag{5.13}$$

We are implicitly using the concept of coset codes [Gal68, Section 6.2] in this proposition, and the proof technique here is quite standard in the polar coding literature. For example, the same proof technique is used to show that the values of the frozen bits do not matter for polar codes [Arı09, KU10] when W is a symmetric channel. The proof of this proposition only uses basic properties of BMS channels and linear codes, we present it here for completeness, as it is an important tool that connects the context of polar codes to the bit-decoding of linear codes.

Proof of Proposition 5.7. Let us unfold the conditioning in the LHS as follows

$$H\left(U_i \,\middle|\, W^\ell(\boldsymbol{U} \cdot K), \boldsymbol{U}_{<i}\right) = \mathop{\mathbb{E}}_{\mathbf{w} \sim \{0,1\}^{i-1}} \left[H\left(U_i \,\middle|\, W^\ell(\boldsymbol{U} \cdot K), \boldsymbol{U}_{<i} = \mathbf{w}\right)\right]. \tag{5.14}$$

We are going to show that the conditional entropy inside the expectation doesn't depend on the choice of $\boldsymbol{w}$, which will allow us to restrict to $\boldsymbol{w} = \mathbf{0}$.

Denote the (random) output of the channel $\boldsymbol{Y} = W^\ell(\boldsymbol{U} \cdot K)$. Let us fix some $\boldsymbol{w} \in \{0,1\}^{i-1}$ and consider $H\left(U_i \,\middle|\, \boldsymbol{Y}, \boldsymbol{U}_{<i} = \boldsymbol{w}\right)$. Unfolding the conditional entropy even more, derive

$$H\left(U_i \,\middle|\, \boldsymbol{Y}, \boldsymbol{U}_{<i} = \boldsymbol{w}\right) = \sum_{\boldsymbol{y} \in \mathcal{Y}^\ell} \mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{U}_{<i} = \boldsymbol{w}] \cdot H\left(U_i \,\middle|\, \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{U}_{<i} = \boldsymbol{w}\right). \tag{5.15}$$

Denote by B the first $(i-1)$ rows of K, and thus $\boldsymbol{Y} = W^\ell(\boldsymbol{U} \cdot K) = W^\ell(\boldsymbol{U}_{<i} \cdot B + \boldsymbol{U}_{\geq i} \cdot G)$.

We then have

$$\begin{aligned}
\mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{U}_{<i} = \boldsymbol{w}] &= \sum_{\mathbf{v} \in \{0,1\}^k} \frac{1}{2^k}\, \mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{U}_{<i} = \boldsymbol{w}, \boldsymbol{U}_{\geq i} = \mathbf{v}] \\
&= \sum_{\mathbf{v} \in \{0,1\}^k} \frac{1}{2^k} W^{\ell}\Big(\boldsymbol{y} \,\Big|\, \mathbf{w} \cdot B + \mathbf{v} \cdot G\Big) \\
&= \sum_{\mathbf{v} \in \{0,1\}^k} \frac{1}{2^k} W^{\ell}\Big(\boldsymbol{y} \oplus \boldsymbol{w}B \,\Big|\, \mathbf{v} \cdot G\Big) \\
&= \sum_{\mathbf{v} \in \{0,1\}^k} \frac{1}{2^k}\, \mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{w}B \mid \boldsymbol{U}_{<i} = \mathbf{0}, \boldsymbol{U}_{\geq i} = \mathbf{v}] \\
&= \mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{w}B \mid \boldsymbol{U}_{<i} = \mathbf{0}]. \qquad (5.16)
\end{aligned}$$

For the entropy in the RHS of (5.15), observe

$$H\left(U_i \,\middle|\, \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{U}_{<i} = \boldsymbol{w}\right) = h\Big(\, \mathbb{P}\left[U_i = 0 \mid \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{U}_{<i} = \boldsymbol{w}\right] \Big),$$

where $h(\cdot)$ is a binary entropy function. Out of the definition of conditional probability, obtain

$$\begin{aligned}
\mathbb{P}\left[U_i = 0 \mid \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{U}_{<i} = \boldsymbol{w}\right] &= \frac{\mathbb{P}[U_i = 0, \boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{U}_{<i} = \boldsymbol{w}]}{\mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{U}_{<i} = \boldsymbol{w}]} \\
&= \frac{\mathbb{P}[U_i = 0, \boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{w}B \mid \boldsymbol{U}_{<i} = \mathbf{0}]}{\mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{w}B \mid \boldsymbol{U}_{<i} = \mathbf{0}]} \\
&= \mathbb{P}\left[U_i = 0 \mid \boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{w}B, \boldsymbol{U}_{<i} = \mathbf{0}\right],
\end{aligned}$$

where the second equality also uses (5.16) (and similar equality with $U_i = 0$ inside the probability, which is almost identical to (5.16)). Therefore, deduce in (5.15)

$$\begin{aligned}
H\left(U_i \,\middle|\, \boldsymbol{Y}, \boldsymbol{U}_{<i} = \boldsymbol{w}\right) &= \sum_{\boldsymbol{y} \in \mathcal{Y}^{\ell}} \mathbb{P}[\boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{w}B \mid \boldsymbol{U}_{<i} = \mathbf{0}] \cdot H\left(U_i \,\middle|\, \boldsymbol{Y} = \boldsymbol{y} \oplus \boldsymbol{w}B, \boldsymbol{U}_{<i} = \mathbf{0}\right) \\
&= \sum_{\boldsymbol{z} \in \mathcal{Y}^{\ell}} \mathbb{P}[\boldsymbol{Y} = \boldsymbol{z} \mid \boldsymbol{U}_{<i} = \mathbf{0}] \cdot H\left(U_i \,\middle|\, \boldsymbol{Y} = \boldsymbol{z}, \boldsymbol{U}_{<i} = \mathbf{0}\right) \\
&= H\left(U_i \,\middle|\, \boldsymbol{Y}, \boldsymbol{U}_{<i} = \mathbf{0}\right),
\end{aligned}$$

since $\boldsymbol{z} = \boldsymbol{y} \oplus \boldsymbol{w}B$ ranges over all $\mathcal{Y}^{\ell}$ for $\boldsymbol{y} \in \mathcal{Y}^{\ell}$. Therefore, in (5.14) there is no actual dependence on $\boldsymbol{w}$ under the expectation in the RHS, and thus

$$H\left(U_i \,\middle|\, W^{\ell}(\boldsymbol{U} \cdot K), \boldsymbol{U}_{<i}\right) = H\left(U_i \,\middle|\, W^{\ell}(\boldsymbol{U} \cdot K), \boldsymbol{U}_{<i} = \mathbf{0}\right).$$

Finally, note that we can take $\boldsymbol{V} = \boldsymbol{U}_{\geq i}$, since it is uniformly distributed over $\{0,1\}^k$, and then $V_1 = U_i$. Since $\boldsymbol{U} \cdot K = \boldsymbol{U}_{\geq i} \cdot G = \boldsymbol{V} \cdot G$ when $\boldsymbol{U}_{<i} = \mathbf{0}$, we indeed obtain

$$H\left(U_i \,\middle|\, W^{\ell}(\boldsymbol{U} \cdot K), \boldsymbol{U}_{<i}\right) = H\left(U_i \,\middle|\, W^{\ell}(\boldsymbol{U} \cdot K), \boldsymbol{U}_{<i} = \mathbf{0}\right) = H\left(V_1 \,\middle|\, W^{\ell}(\boldsymbol{V} \cdot G)\right). \quad \square$$

This brings us to the settings of bit-decoding for random linear codes. Notice that the LHS of (5.13) is exactly the entropy $H(W_i)$ of the i-th Arıkan's bit-channel of W with respect to the kernel K, by definition of this bit-channel. On the other hand, one can think of the RHS of (5.13) in the following way: look at G as a generator matrix for a linear code of blocklength ℓ and dimension k, which is transmitted through the channel W. Then $H\left(V_1 \,\middle|\, W^\ell(\boldsymbol{V}\cdot G)\right)$ corresponds to how well one can decode the first bit of the message, given the output of the channel. Since in Theorem 5.5 we are interested in random kernels, the generator matrix G is also random, and thus we are indeed interested in understanding the bit-decoding of random linear codes.

5.2.2.1 Part (a): channel capacity theorem

Part (a) of Theorem 5.5 corresponds to transmitting through W random linear codes with rates *below* the capacity of the channel. For this regime, it turns out that we can use the classical result that random linear codes achieve the capacity of the channel with *low error decoding probability.* Trivially, the bit-decoding error probability is even smaller, making the corresponding conditional entropy also very small. We just need to formally show the strong quantitative form we require in Theorem 5.5. Therefore, the following theorem follows from classical Shannon's theory:

Theorem 5.8. *Let W be any BMS channel and $k \leq \ell(1-H(W)) - \ell^{1/2}\log^3\ell$, where $\ell \geq 4$. Let G be a random binary matrix uniform over $\{0,1\}^{k\times\ell}$. Suppose a codeword $\boldsymbol{V}\cdot G$ is transmitted through ℓ copies of the channel W, where $\boldsymbol{V}$ is uniformly random over $\{0,1\}^k$, and let $\boldsymbol{Y}$ be the output vector, i.e. $\boldsymbol{Y} = W^\ell(\boldsymbol{V}\cdot G)$. Then with high probability over the choice of G it holds $H\left(V_1 \,\middle|\, \boldsymbol{Y}\right) \leq \ell^{-(\log\ell)/4}$.*

Proof. The described communication is just a transmission of a random linear code $C = \{\boldsymbol{v}G,\ \boldsymbol{v}\in\{0,1\}^k\}$ through W^ℓ, where the rate of the code is $R = \frac{k}{\ell} \leq I(W) - \ell^{-1/2}\log^3\ell$, so it is separated from the capacity of the channel. It is a well-studied fact that random (linear) codes achieve capacity for BMS, and moreover a tight error exponent was described by Gallager in [Gal65] and analyzed further in [BF02], [For05], [DZF16]. Specifically, one can show $\overline{P_e} \leq \exp(-\ell E_r(R,W))$, where $\overline{P_e}$ is the probability of decoding error, averaged over the ensemble of all linear codes of rate R, and $E_r(R,W)$ is the so-called *random coding exponent.* It is proven in [iFLM11, Theorem 2.3] that for any BMS channel W, one has $E_r(R,W) \geq E_r^{\mathrm{BSC}}(R, I(W))$ where the latter is the error exponent for the BSC channel with the same capacity $I(W)$ as W. But the optimal scaling exponent for BSC channels for the regime when the rate is close to the capacity of the channel is given by the so-called sphere-packing exponent $E_r^{\mathrm{BSC}}(R,I) = E_{\mathrm{sp}}(R,I)$ (see, for instance, [For05, Section 1.2], which is easily shown to be almost quadratic in $(I-R)$. Specifically, we use the following

Lemma 5.9. *$E_{sp}(R,I) \geq \frac{2\log^4\ell}{\ell}$ for $R \leq I - \ell^{-1/2}\log^3\ell$.*

Proof. For the sphere-packing exponent we use the expression from [For05, eq (1.4)]

$$E_{\mathrm{sp}}(R, I) = D_{\mathrm{KL}}\left(\delta_{\mathrm{GV}}(R) \,\middle|\middle|\, p\right),$$

where $I = I(W) = 1 - H(W) = 1 - h(p)$ is the capacity of the BSC_p channel (with $p < \frac{1}{2}$), D_{KL} stands for the Kullback–Leibler divergence, and $\delta_{GV}(R)$ is the relative Gilbert-Varshamov distance, which is defined as the solution to $1 - h(\delta) = R$ for $\delta \in \left(0, \frac{1}{2}\right)$. For convenience, we will just write δ instead of $\delta_{\mathrm{GV}}(R)$ below.

For $R \leq I - \ell^{-1/2}\log^3 \ell = 1 - h(p) - \ell^{-1/2}\log^3 \ell$, we then have $1 - h(\delta) \leq 1 - h(p) - \ell^{-1/2}\log^3 \ell$, and so $h(\delta) - h(p) \geq \ell^{-1/2}\log^3 \ell$. Using Proposition 2.8, obtain $h(\delta - p) \geq h(\delta) - h(p) \geq \ell^{-1/2}\log^3 \ell$. Next, since $h(x)$ in increasing on $\left(0, \frac{1}{2}\right)$ and by Proposition 2.7

$$h(\ell^{-1/2}\log^2 \ell) \leq 2\ell^{-1/2}\log^2 \ell \cdot \log \frac{\ell^{1/2}}{\log^2 \ell} \leq 2\ell^{-1/2}\log^2 \ell \cdot \frac{1}{2}\log \ell = \ell^{-1/2}\log^3 \ell,$$

we conclude that $\delta - p \geq \ell^{-1/2}\log^2 \ell$.

Finally, we use Pinsker's inequality $D_{\mathrm{KL}}\left(P \,||\, Q\right) \geq 2\Delta^2(P, Q)$ between the KL divergence and the total variation distance $\Delta(P, Q) = \frac{1}{2}||P - Q||_1$ of two distributions P and Q over the same probability space. Abusing the notation and denoting $\Delta(\delta, p)$ as the distance between $\mathrm{Bern}(\delta)$ and $\mathrm{Bern}(p)$, we have $\Delta(\delta, p) = |\delta - p|$, and so obtain

$$E_{\mathrm{sp}}(R, I) = D_{\mathrm{KL}}\left(\delta \,||\, p\right) \geq 2\Delta^2(\delta, p) = 2(\delta - p)^2 \geq \frac{2\log^4 \ell}{\ell}. \qquad \square$$

Therefore using this lemma we have $\overline{P_e} \leq \exp(-\ell E_r(R, W)) \leq \exp(-\ell E_{\mathrm{sp}}(R, I(W))) \leq \exp(-2\log^4 \ell)$. Then Markov's inequality implies that if we take a random linear code (i.e. choose a random binary matrix G), then with probability at least $1 - \ell^{-2}$ the decoding error is going to be at most $\ell^2 \exp(-2\log^4 \ell) \leq \exp(-\log^4 \ell) \leq \ell^{-\log \ell}$. Consider such a good linear code (matrix G), and then $\boldsymbol{V}$ can be decoded from $\boldsymbol{Y}$ with high probability, thus, clearly, V_1 can be recovered from $\boldsymbol{Y}$ with at least the same probability. Then Fano's inequality and Proposition 2.7 gives us:

$$\begin{aligned} H(V_1 \mid \boldsymbol{Y}) \leq h_2(\ell^{-\log \ell}) &\leq 2\ell^{-\log \ell} \cdot \log\left(\frac{1}{\ell^{-\log \ell}}\right) \\ &= 2\ell^{-\log \ell} \cdot \log^2 \ell \quad \leq \quad \ell^{-(\log \ell)/4}, \end{aligned}$$

where the last inequality follows from $2\log^2 \ell \leq 2^{\frac{3\log^2 \ell}{4}}$, which holds for $\ell \geq 4$. Thus we indeed obtain that the above holds with high probability (at least $1 - \ell^{-2}$, though this is very loose) over the random choice of G. □

5.2.2.2 Part (b): strong converse for bit-decoding under noisy channel coding

On the other hand, part (b) of Theorem 5.5 concerns bit-decoding of linear codes with rates *above* the capacity of the channel. This is exactly the strong converse in Theorem 3.1

we proved in Chapter 3, where we proved that with high probability, for a random linear code with rate slightly above the capacity of a BMS channel, any single bit of the input message is highly unpredictable based on the outputs of the channel on the transmitted codeword.

The above statements make the proof of Theorem 5.5 immediate:

Proof of Theorem 5.5. Fix i and denote $k = \ell - i + 1$, then by Proposition 5.7 $H(W_i) = H\left(V_1 \,\middle|\, W^\ell(\boldsymbol{V} \cdot G_k)\right)$, where $\boldsymbol{V} \sim \{0,1\}^k$ and G_k is formed by the last k rows of K. Note that since K is uniform over $\{0,1\}^{\ell\times\ell}$, this makes G_k uniform over $\{0,1\}^{k\times\ell}$ for any k. Then:

(a) For any $i \geq \ell \cdot H(W) + \ell^{1/2}\log^3\ell$, we have $k \leq \ell(1 - H(W)) - \ell^{1/2}\log^3\ell$, and therefore Theorem 5.8 applies, giving $H(W_i) \leq \ell^{-(\log\ell)/4}$ with probability at least $1 - \ell^{-2}$ over K.

(b) Analogically, if $i \leq \ell \cdot H(W) - 14 \cdot \ell^{1/2}\log^3\ell$, then $k \geq \ell(1 - H(W)) + 14\ell^{1/2}\log^3\ell$, and Theorem 3.1 gives $H(W_i) \geq 1 - \ell^{-(\log\ell)/20}$ with probability at least $1 - \ell^{-(\log\ell)/20}$ over K.

It only remains to take the union bound over all indices i in (a) and (b) and recall that we took ℓ large enough so that $\log\ell > 40$. This implies that all of the bounds on the entropies will hold simultaneously with probability at least $1 - \ell\cdot\ell^{-2} \geq 1 - \ell^{-1}$ over the random kernel K. □

5.3 Suction at the ends

In this section we present the proof for Theorem 5.2 in the case the standard Arıkan's kernel was chosen in Algorithm A – the so-called suction at the ends regime. Recall that, as we discussed in section 5.2.1, this regime applies when the entropy of the channel W falls outside of the interval $(\ell^{-4}, 1 - \ell^{-4})$, and the algorithm directly takes a kernel $K = A_2^{\otimes\log\ell}$, where $A_2 = \left[\begin{smallmatrix}1 & 0\\ 1 & 1\end{smallmatrix}\right]$ is the kernel of Arıkan's original polarizing transform, instead of trying out all the possible $\ell\times\ell$ matrices (or picking it at random). Note that multiplying by such a kernel K is equivalent to applying the Arıkan's 2×2 transform recursively $\log\ell$ times. Suppose we have a BMS channel W with $H(W)$ very close to 0 or 1. For Arıkan's basic transform, by working with the channel Bhattacharyya parameter $Z(W)$ instead of the entropy $H(W)$, it is well known that one of the two Arıkan bit-channels has Z value getting much closer (quadratically closer) to the boundary of the interval $(0,1)$ [Arı09, Kor09]. Using these ideas, we prove in this section that basic transform decreases the average of the potential function $g_\alpha(\cdot)$ of entropy at least by a factor of $\ell^{-1/2}$ after $\log\ell$ iterations for large enough ℓ.

The basic Arıkan's transform takes one channel W and splits it into a slightly worse channel W^- and a slightly better channel W^+ (we used slightly different notations $W^{(0)}$ and $W^{(1)}$ in Section 4.1 for the same channels). Then the transform is applied recursively to W^- and W^+, creating channels W^{--}, W^{-+}, W^{+-}, and W^{++}. One can think of the process as of a complete binary tree of depth $\log \ell$, with the root node W, and any node at the level i is of form W^{B_i} for some $B_i \in \{-,+\}^i$, with two children W^{B_i-} and W^{B_i+}. Denote $r = \log \ell$, then the channels at the leaves $\{W^{B_r}\}$, for all $B_r \in \{-,+\}^r$ are exactly[1] the Arıkan's subchannels of W with respect to the kernel $K = A_2^{\otimes \log \ell}$. We prove the following

Lemma 5.10. *Let W be a BMS channel with $H(W) \notin (\ell^{-4}, 1-\ell^{-4})$, and $\alpha \in \left(0, \frac{1}{12}\right)$ be some constant. Let ℓ be a power of two and denote $r = \log \ell$. Then for ℓ large enough such that $r \geq \max\left\{\dfrac{1}{\alpha}, 128\right\}$*

$$\sum_{B \in \{-,+\}^r} g_\alpha\left(H\left(W^B\right)\right) \leq \ell^{1/2} g_\alpha\left(H(W)\right), \tag{5.17}$$

where $g_\alpha(\cdot)$ is the potential function defined in (5.2).

Clearly, the above lemma implies the suction at the end case of Theorem 5.2, as the inequality $\log \ell \geq \frac{1}{\alpha}$ holds by the conditions of this theorem.

For the analysis below, apart from the entropy of the channel, we will also use the Bhattacharyya parameter $Z(W)$ (Definition 2.5) and the inequalities which connect it to the entropy:

$$Z(W)^2 \leq H(W) \leq Z(W), \tag{5.18}$$

for any BMS channel W ([Kor09, Lemma 1.5], [Arı10, Proposition 2]). The reason we use this parameter is because of the following relations, which show how the Bhattacharrya parameter changes after the basic transform ([Arı09, Proposition 5] [RU08], [HAU14, eq (13)]):

$$Z(W^+) = Z(W)^2, \tag{5.19}$$

$$Z(W)\sqrt{2 - Z(W)^2} \leq Z(W^-) \leq 2Z(W). \tag{5.20}$$

The fact that Z squares for the better channel and is at worst multiplied by a constant in the worse channel is what enables to prove strong suction at the ends, and this is what lies at the root of sub-exponentially small block error probability for polar codes.

We will also use the conservation of conditional entropy (4.4)

$$H(W^+) + H(W^-) = 2H(W). \tag{5.21}$$

Proof of Lemma 5.10. The proof is presented in the next two sections, as it is divided into two parts: the case when $H(W) \leq \ell^{-4}$ (suction at the lower end), and when $H(W) \geq 1-\ell^{-4}$ (suction at the upper end).

[1] We ignore bit-reversal permutation here, as is doesn't change the proofs.

5.3.1 Suction at the lower end

Suppose $H(W) \leq \ell^{-4}$ for this case, thus $Z(W) \leq \ell^{-2} = 2^{-2r}$.

First, recursive application of (5.21) gives

$$\sum_{B \in \{-,+\}^r} H\left(W^B\right) = 2^r H(W), \tag{5.22}$$

and since entropy is always nonnegative, this implies for any $B \in \{-,+\}^r$

$$H\left(W^B\right) \leq 2^r H(W). \tag{5.23}$$

Denote now $k = \left\lceil \log \frac{1}{\alpha} \right\rceil$, and notice that $\log r \geq k-1$ since $r \geq \frac{1}{\alpha}$. For $B \in \{-,+\}^r$, define $wt_+(B)$ to be number of $+$'s in B. We will split the summation in (5.17) into two parts: the part with $wt_+(B) < k$, and when $wt_+(B) \geq k$.

First part. Out of (5.23) derive

$$\begin{aligned} \sum_{wt_+(B)<k} g_\alpha\left(H\left(W^B\right)\right) \leq \sum_{j=0}^{k-1} \binom{r}{j} g_\alpha\left(2^r H(W)\right) &\leq \log r \cdot \binom{r}{\log r} \cdot 2^{r\alpha} H(W)^\alpha \\ &\leq 2^{\log^2 r + r\alpha} \cdot H(W)^\alpha, \end{aligned} \tag{5.24}$$

where we used $\binom{r}{\log r} \leq \frac{r^{\log r}}{(\log r)!}$; the fact the g_α is increasing on $\left(0, \frac{1}{2}\right)$ together with $2^r H(W) \leq \ell^{-3} < \frac{1}{2}$, and that $g_\alpha(x) \leq x^\alpha$ for $x \in (0,1)$.

Second part. We are going to use the following observation, which was established in [AT09, Lemma 1] and can be proved by induction based on (5.19) and (5.20):

Claim 5.11. *Let $B \in \{-,+\}^r$, such that number of $+$'s in B is equal to s. Then*

$$Z\left(W^B\right) \leq \left(2^{r-s} \cdot Z(W)\right)^{2^s}.$$

This corresponds to first using the upper bound in (5.20) *$(r-s)$ times, and after that using* (5.19) *s times while walking* ***down*** *the recursive binary tree of channels.*

Then, using Claim 5.11 along with (5.18) and the fact that $Z(W) \leq \ell^{-2} = 2^{-2r}$, we obtain the following for any $B \in \{-,+\}^r$ with $wt_+(B) = s \geq k$:

$$\begin{aligned} H\left(W^B\right) \leq Z\left(W^B\right) \leq \left(2^{r-s} \cdot Z(W)\right)^{2^s} &\leq 2^{(r-s)2^s} \cdot Z(W)^{2^s-2} \cdot H(W) \\ &\leq 2^{(r-s)2^s} \cdot 2^{-2r \cdot 2^s + 4r} \cdot H(W) \\ &= 2^{-r2^s - s2^s + 4r} \cdot H(W) \\ &\leq 2^{-r2^k - k2^k + 4r} \cdot H(W). \end{aligned}$$

Therefore

$$\sum_{wt_+(B)\geq k} g_\alpha\left(H\left(W^B\right)\right) \leq \sum_{wt_+(B)\geq k} H\left(W^B\right)^\alpha \leq 2^r \cdot 2^{\alpha(-r2^k-k2^k+4r)} \cdot H(W)^\alpha. \tag{5.25}$$

Observe now the following chain of inequalities

$$\frac{r}{2} + 4r\alpha + 2 \leq r \leq r \cdot 2^k\alpha \leq r \cdot 2^k\alpha + k \cdot 2^k\alpha,$$

which trivially holds for $\alpha \leq \dfrac{1}{12}$. Therefore

$$r + \alpha(-r2^k - k2^k + 4r) \leq \frac{r}{2} - 2,$$

and thus in (5.25) obtain

$$\sum_{wt_+(B)\geq k} g_\alpha\left(H\left(W^B\right)\right) \leq 2^{r/2-2} \cdot H(W)^\alpha. \tag{5.26}$$

Overall bound. Combining (5.24) and (5.26) we derive

$$\begin{aligned}\sum_{B\in\{-,+\}^r} g_\alpha\left(H\left(W^B\right)\right) &\leq \left(2^{\log^2 r+r\alpha} + 2^{r/2-2}\right) \cdot H(W)^\alpha \\ &\leq 2^{r/2} \cdot \frac{H(W)^\alpha}{2} \\ &\leq \ell^{1/2} g_\alpha(H(W)),\end{aligned}$$

where we used $\log^2 r + r\alpha \leq \frac{r}{2} - 2$ for $r \geq 128$, and $\frac{1}{2} \leq (1-x)^\alpha$ for any $x \leq \frac{1}{2}$. This proves Lemma 5.10 for the lower end case $H(W) \leq \ell^{-4}$.

5.3.2 Suction at the upper end

Now consider the case $H(W) \geq 1 - \ell^{-4}$. The proof is quite similar to the previous case, but we are going to track the distance from $H(W)$ (and $Z(W)$) to 1 now. Specifically, denote

$$I(W) = 1 - H(W),$$
$$S(W) = 1 - Z(W),$$

where $I(W)$ is actually the (symmetric) capacity of the channel, and $S(W)$ is a notation for a parameter we use in this proof[2]. Notice that $g_\alpha(x) = g_\alpha(1-x)$, therefore it suffices to prove (5.17) with capacities of the channels instead of entropies in the inequality. Also notice that $I(W) \leq \ell^{-4}$ for the current case of suction at the upper end.

[2] for analyzing suction at the end for channels with non-binary input alphabet, a more complex parameter $S(W)$ is usually studied, see e.g [MT14, WD19].

Let us now derive the relations between $I(W)$, $S(W)$, as well as evolution of $S(\cdot)$ for W^+ and W^-, similar to (5.18), (5.19), (5.20), and (5.21). Inequalities in (5.18) imply

$$S(W) = 1 - Z(W) \le 1 - H(W) = I(W),$$
$$I(W) = 1 - H(W) \le 1 - Z(W)^2 \le 2(1 - Z(W)) = 2S(W),$$

so let us combine this to write

$$S(W) \le I(W) \le 2S(W). \tag{5.27}$$

Next, (5.19) and (5.20) give

$$S(W^+) = 1 - Z(W)^2 \le 2(1 - Z(W)) \le 2S(W), \tag{5.28}$$
$$S(W^-) \le 1 - Z(W)\sqrt{2 - Z(W)^2} \le 2(1 - Z(W))^2 = 2S(W)^2, \tag{5.29}$$

where we used $1 - x\sqrt{2 - x^2} \le 2(1 - x)^2$ for any $x \in (0, 1)$, which can be proven easily by showing that equality holds at $x = 1$ and that the derivative of RHS minus LHS is negative on $(0, 1)$.

Finally, it easily follows from (5.22) that

$$\sum_{B \in \{-,+\}^r} I\left(W^B\right) = 2^r I(W),$$

and since capacity is nonnegative as well, we also obtain for any $B \in \{-,+\}^r$

$$I\left(W^B\right) \le 2^r I(W). \tag{5.30}$$

We now proceed with a very similar approach to the suction at the lower end case in Section 5.3.1: denote $k = \left\lceil \log \frac{1}{\alpha} \right\rceil$, and notice that $\log r \ge k - 1$ since $r \ge \frac{1}{\alpha}$. For $B \in \{-,+\}^r$, define $wt_-(B)$ to be number of $-$'s in B. We will split the summation in (5.17) (but with capacities of channels instead of entropies) into two parts: the part with $wt_-(B) < k$, and when $wt_-(B) \ge k$.

First part. Out of (5.30) derive, similarly to (5.24)

$$\sum_{wt_-(B) < k} g_\alpha\left(I\left(W^B\right)\right) \le \sum_{j=0}^{k-1} \binom{r}{j} g_\alpha\left(2^r I(W)\right) \le \log r \binom{r}{\log r} 2^{r\alpha} I(W)^\alpha \le 2^{\log^2 r + r\alpha} \cdot I(W)^\alpha. \tag{5.31}$$

Second part. Similarly to Claim 5.11, one can show via induction using (5.28) and (5.29) the following

Claim 5.12. *Let $B \in \{-,+\}^r$, such that number of $-$'s in B is equal to s. Then*

$$S\left(W^B\right) \le 2^{2^s - 1}\left(2^{r-s} \cdot S(W)\right)^{2^s}.$$

This corresponds to first using equality (5.28) $(r-s)$ times, and after that using bound (5.29) s times while walking ***down*** *the recursive binary tree of channels.*

Using this claim with (5.27) and the fact that $S(W) \leq I(W) \leq \ell^{-4} \leq 2^{-4r}$ obtain for any $B \in \{-,+\}^r$ with $wt_-(B) = s \geq k$

$$\begin{aligned} I\left(W^B\right) \leq 2S\left(W^B\right) &\leq 2^{2^s} \cdot \left(2^{r-s} \cdot S(W)\right)^{2^s} && \leq 2^{(r-s+1)2^s} \cdot S(W)^{2^s-1} \cdot I(W) \\ &\leq 2^{(r-s+1)2^s-4r2^s+4r} \cdot I(W) && = 2^{-2^s(3r+s-1)+4r} \cdot I(W) \\ &\leq 2^{-2^k(3r+k-1)+4r} \cdot I(W) && \leq 2^{-r2^k} \cdot I(W), \end{aligned}$$

where the last inequality uses $4r \leq 2^k(2t+k-1)$, which holds trivially for $k \geq 1$. Therefore

$$\sum_{wt_-(B)\geq k} g_\alpha\left(I\left(W^B\right)\right) \leq \sum_{wt_-(B)\geq k} I\left(W^B\right)^\alpha \leq 2^r \cdot 2^{-\alpha r 2^k} \cdot I(W)^\alpha \leq I(W)^\alpha, \tag{5.32}$$

since $\alpha \cdot 2^k \geq 1$ by the choice of k.

Overall bound. The bounds (5.31) and (5.32) give us

$$\sum_{B\in\{-,+\}^r} g_\alpha\left(H\left(W^B\right)\right) = \sum_{B\in\{-,+\}^r} g_\alpha\left(I\left(W^B\right)\right) \leq \left(2^{\log^2 r + r\alpha} + 1\right) \cdot I(W)^\alpha \leq \ell^{1/2} g_\alpha(H(W))$$

for large enough r when $H(W) \geq 1 - \ell^{-4}$. This completes the proof of Lemma 5.10. □

This also marks the end of the complete proof for Theorem 5.2. So we know know that we indeed can find a suitable kernel for each bit-channel in the tree, such that the total fraction of unpolarized channels decays fast enough to ensure the near-optimal scaling exponent. In the remaining part of this chapter, we formally describe our construction and wrap up the proof of Theorem 5.1.

5.4 Code construction, encoding and decoding procedures

Before presenting our code construction and encoding/decoding procedures, we first distinguish the difference between the code construction and the encoding procedure. The objectives of code construction for polar-type codes are two-fold: First, find the $N \times N$ encoding matrix; second, find the set of noiseless bits under the successive cancellation decoder, which will carry the message bits. On the other hand, by encoding we simply mean the procedure of obtaining the codeword $\boldsymbol{X}_{[1:N]}$ by multiplying the information vector $\boldsymbol{U}_{[1:N]}$ with the encoding matrix, where we only put information in the noiseless bits in $\boldsymbol{U}_{[1:N]}$ and set all the frozen bits to be 0. As we will see at the end of this section, while the code construction has complexity polynomial in N, the encoding procedure only has complexity $O_\ell(N \log N)$.

For polar codes with a fixed invertible kernel $K \in \{0,1\}^{\ell\times\ell}$, the polarization process works as follows: We start with some BMS channel W. After applying the polar transform

Algorithm B: Degraded binning algorithm

```
Input: $W : \{0,1\} \to \mathcal{Y}$, bound $\mathsf{Q}$ on the output alphabet size after binning
Output: $\widetilde{W} : \{0,1\} \to \widetilde{\mathcal{Y}}$, where $|\widetilde{\mathcal{Y}}| \le \mathsf{Q}$
Initialize the new channel $\widetilde{W}$ with output symbols $\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_{\mathsf{Q}}$ by setting
  $\widetilde{W}(\tilde{y}_i|x) = 0$ for all $i \in [\mathsf{Q}]$ and $x \in \{0,1\}$
for $y \in \mathcal{Y}$ do
    $p(0|y) \leftarrow \frac{W(y|0)}{W(y|0)+W(y|1)}$
    $i \leftarrow \lceil \mathsf{Q} \cdot p(0|y) \rceil$
    if $i = 0$ then
        $i \leftarrow 1$    // $i = 0$ if and only if $p(0|y) = 0$; we merge this single point into
                              the next bin
    end
    $\widetilde{W}(\tilde{y}_i|0) \leftarrow \widetilde{W}(\tilde{y}_i|0) + W(y|0)$
    $\widetilde{W}(\tilde{y}_i|1) \leftarrow \widetilde{W}(\tilde{y}_i|1) + W(y|1)$
end
return $\widetilde{W}$
```

to W using kernel K, we obtain ℓ bit-channels $\{W_i : i \in [\ell]\}$ as defined in (4.9). Next, we apply the polar transform using kernel K to each of these ℓ bit-channels, and we write the polar transform of W_i as $\{W_{ij} : j \in [\ell]\}$. Then we apply the polar transform to each of the ℓ^2 bit channels $\{W_{i_1,i_2} : i_1, i_2 \in [\ell]\}$ and obtain $\{W_{i_1,i_2,i_3} : i_1, i_2, i_3 \in [\ell]\}$, so on and so forth. After t rounds of polar transforms, we obtain ℓ^t bit-channels $\{W_{i_1,\dots,i_t} : i_1, \dots, i_t \in [\ell]\}$, and one can show that these are the bit-channels seen by the successive cancellation decoder when decoding the corresponding polar codes constructed from kernel K.

For our purpose, we need to use polar codes with mixed kernels, and we need to search for a “good” kernel at each step of polarization. We will also introduce a new notation for the bit-channels in order to indicate the usage of different kernels for different bit-channels. As mentioned in Sections 5.1.4 and 5.2.1, we need to use a binning algorithm (Algorithm B) to quantize all the bit-channels we obtain in the code construction procedure. As long as we choose the parameter Q in Algorithm B to be a large enough polynomial of N, the quantized channel can be used as a very good approximation of the original channel. This is made precise by [GX15, Proposition 13]: For W and $\widetilde{W}$ in Algorithm B, we have[3]

$$H(W) \le H(\widetilde{W}) \le H(W) + \frac{2\log \mathsf{Q}}{\mathsf{Q}}. \tag{5.33}$$

Given a BMS channel W, our code construction works as follows:

1. **Step 0:** We first use Algorithm B to quantize/bin the output alphabet of W such that the resulting (degraded) channel has at most N^3 outputs, i.e., we set $\mathsf{Q} = N^3$

[3] The binning algorithm (Algorithm 2) in [GX15] has one minor difference from the binning algorithm (Algorithm B) here, but one can easily check that this minor difference does not affect the proof at all.

in Algorithm B. Note that the parameter Q can be chosen as any polynomial of N. By changing the value of Q, we obtain a tradeoff between the decoding error probability and the gap to capacity; see Section 5.5.2. Here we choose the special case of $\mathsf{Q} = N^3$ to give a concrete example of code construction. Next we use Algorithm A in Section 5.2 to find a good kernel[4] for the quantized channel and denote it as $K_1^{(0)}$. Recall from Section 5.1.2 that a kernel is good if all but a $\tilde{O}(\ell^{-1/2})$ fraction of the bit-channels obtained after polar transform by this kernel have entropy $\ell^{-\Omega(\log \ell)}$-close to either 0 or 1. The superscript (0) in $K_1^{(0)}$ indicates that this is the kernel used in Step 0 of polarization. In this case, we use $\{W_i(B, K_1^{(0)}) : i \in [\ell]\}$ to denote the ℓ bit-channels resulting from the polar transform of the quantized version of W using kernel $K_1^{(0)}$. Here B stands for the binning operation, and the arguments in the brackets are the operations to obtain the bit-channel $W_i(B, K_1^{(0)})$ from W: first bin the outputs of W and then perform the polar transform using kernel $K_1^{(0)}$. For each $i \in [\ell]$, we again use Algorithm B to quantize/bin the output alphabet of $W_i(B, K_1^{(0)})$ such that the resulting (degraded) bit-channel $W_i(B, K_1^{(0)}, B)$ has at most N^3 outputs.

2. **Step 1:** For each $i_1 \in [\ell]$, we use Algorithm A to find a good kernel for the quantized bit-channel $W_{i_1}(B, K_1^{(0)}, B)$ and denote it as $K_{i_1}^{(1)}$. The ℓ bit-channels resulting from the polar transform of $W_{i_1}(B, K_1^{(0)}, B)$ using kernel $K_{i_1}^{(1)}$ are denoted as $\{W_{i_1,i_2}(B, K_1^{(0)}, B, K_{i_1}^{(1)}) : i_2 \in [\ell]\}$. In this step, we will obtain ℓ^2 bit-channels $\{W_{i_1,i_2}(B, K_1^{(0)}, B, K_{i_1}^{(1)}) : i_1, i_2 \in [\ell]\}$. For each of them, we use Algorithm B to quantize/bin its output alphabet such that the resulting (degraded) bit-channels $\{W_{i_1,i_2}(B, K_1^{(0)}, B, K_{i_1}^{(1)}, B) : i_1, i_2 \in [\ell]\}$ has at most N^3 outputs. See Fig. 5.2 for an illustration of this procedure for the special case of $\ell = 3$.

3. We repeat the polar transforms and binning operations at each step of the code construction. More precisely, at **Step** j we have ℓ^j bit-channels

$$\{W_{i_1,i_2,\dots,i_j}(B, K_1^{(0)}, B, K_{i_1}^{(1)}, B, \dots, K_{i_1,\dots,i_{j-1}}^{(j-1)}, B) : i_1, i_2, \dots, i_j \in [\ell]\}.$$

This notation is a bit messy, so we introduce some simplified notation for the bit-channels obtained with and without binning operations: We still use

$$W_{i_1,i_2,\dots,i_j}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K_{i_1,\dots,i_{j-1}}^{(j-1)})$$

to denote the bit-channel obtained without the binning operations at all, and we use

$$W_{i_1,i_2,\dots,i_j}^{\text{bin}}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K_{i_1,\dots,i_{j-1}}^{(j-1)})$$

to denote the bit-channel obtained with binning operations performed at every step from Step 0 to Step $j-1$, i.e.,

$$W_{i_1,i_2,\dots,i_j}^{\text{bin}}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K_{i_1,\dots,i_{j-1}}^{(j-1)}) := W_{i_1,i_2,\dots,i_j}(B, K_1^{(0)}, B, K_{i_1}^{(1)}, B, \dots, K_{i_1,\dots,i_{j-1}}^{(j-1)}, B).$$

[4] We will prove in Proposition 5.15 that the error parameter Δ in Algorithm A can be chosen as $\Delta = \frac{6\ell \log N}{N^2}$ when we set $\mathsf{Q} = N^3$.

Moreover, we use $W^{\mathrm{bin}*}_{i_1,i_2,\dots,i_j}(K_1^{(0)},K_{i_1}^{(1)},\dots,K_{i_1,\dots,i_{j-1}}^{(j-1)})$ to denote the bit-channel obtained with binning operations performed at every step except for the last step, i.e.,

$$W^{\mathrm{bin}*}_{i_1,i_2,\dots,i_j}(K_1^{(0)},K_{i_1}^{(1)},\dots,K_{i_1,\dots,i_{j-1}}^{(j-1)}) := W_{i_1,i_2,\dots,i_j}(B,K_1^{(0)},B,K_{i_1}^{(1)},B,\dots,B,K_{i_1,\dots,i_{j-1}}^{(j-1)}).$$

Next we use Algorithm A to find a good kernel for each of them and denote the kernel as $K^{(j)}_{i_1,\dots,i_j}$. After applying polar transforms using these kernels, we obtain ℓ^{j+1} bit-channels

$$\{W^{\mathrm{bin}*}_{i_1,\dots,i_{j+1}}(K_1^{(0)},K_{i_1}^{(1)},\dots,K_{i_1,\dots,i_j}^{(j)}) : i_1,\dots,i_{j+1}\in[\ell]\}.$$

Then we quantize/bin the output alphabets of these bit-channels using Algorithm B and obtain the following ℓ^{j+1} quantized bit-channels

$$\{W^{\mathrm{bin}}_{i_1,\dots,i_{j+1}}(K_1^{(0)},K_{i_1}^{(1)},\dots,K_{i_1,\dots,i_j}^{(j)}) : i_1,\dots,i_{j+1}\in[\ell]\}.$$

4. After **step** $t-1$, we obtain $N=\ell^t$ quantized bit-channels

$$\{W^{\mathrm{bin}}_{i_1,\dots,i_t}(K_1^{(0)},K_{i_1}^{(1)},\dots,K_{i_1,\dots,i_{t-1}}^{(t-1)}) : i_1,i_2,\dots,i_j\in[\ell]\},$$

and we have also obtained all the kernels in each step of polarization. More precisely, we have ℓ^i kernels in step i, so from step 0 to step $t-1$, we have $1+\ell+\dots+\ell^{t-1}=\frac{N-1}{\ell-1}$ kernels in total.

5. Find the set of good (noiseless) indices. More precisely, we use the shorthand notation[5]

$$\begin{aligned} H_{i_1,\dots,i_t}(W) &:= H(W_{i_1,\dots,i_t}(K_1^{(0)},K_{i_1}^{(1)},\dots,K_{i_1,\dots,i_{t-1}}^{(t-1)})) \\ H^{\mathrm{bin}}_{i_1,\dots,i_t}(W) &:= H(W^{\mathrm{bin}}_{i_1,\dots,i_t}(K_1^{(0)},K_{i_1}^{(1)},\dots,K_{i_1,\dots,i_{t-1}}^{(t-1)})) \end{aligned} \tag{5.34}$$

and define the set of good indices as

$$\mathcal{S}_{\mathrm{good}} := \left\{(i_1,i_2,\dots,i_t)\in[\ell]^t : H^{\mathrm{bin}}_{i_1,\dots,i_t}(W) \le \frac{7\ell\log N}{N^2}\right\}. \tag{5.35}$$

6. Finally, we need to construct the encoding matrix from these $\frac{N-1}{\ell-1}$ kernels. The kernels we obtained in step j are

$$\{K^{(j)}_{i_1,\dots,i_j} : i_1,\dots,i_j\in[\ell]\}.$$

For an integer $i\in[\ell^j]$, we write the j-digit ℓ-ary expansion of $i-1$ as $(\tilde{i}_1,\tilde{i}_2,\dots,\tilde{i}_j)$, where $\tilde{i}_j$ is the least significant digit and $\tilde{i}_1$ is the most significant digit, and each digit takes value in $\{0,1,\dots,\ell-1\}$. Let $(i_1,i_2,\dots,i_j) := (\tilde{i}_1+1,\tilde{i}_2+1,\dots,\tilde{i}_j+1)$, and define the mapping $\tau_j : [\ell^j]\to[\ell]^j$ as

$$\tau_j(i) := (i_1,i_2,\dots,i_j) \quad \text{for } i\in[\ell^j]. \tag{5.36}$$

[5] We omit the reference to the kernels in the notation $H_{i_1,\dots,i_t}(W)$ and $H^{\mathrm{bin}}_{i_1,\dots,i_t}(W)$.

This is a one-to-one mapping between $[\ell^j]$ and $[\ell]^j$, and we use the shorthand notation $K_i^{(j)}$ to denote $K_{\tau_j(i)}^{(j)}$ for $i \in [\ell^j]$. For each $j \in \{0, 1, \dots, t-1\}$, we define the block diagonal matrices $\overline{D}^{(j)}$ with size $\ell^{j+1} \times \ell^{j+1}$ and $D^{(j)}$ with size $N \times N$ as

$$\overline{D}^{(j)} := \mathrm{Diag}(K_1^{(j)}, K_2^{(j)}, \dots, K_{\ell^j}^{(j)}), \qquad D^{(j)} := \underbrace{\{\overline{D}^{(j)}, \overline{D}^{(j)}, \dots, \overline{D}^{(j)}\}}_{\text{number of } \overline{D}^{(j)} \text{ is } \ell^{t-j-1}}. \tag{5.37}$$

For $i \in [\ell^t]$, we have $\tau_t(i) = (i_1, \dots, i_t)$. For $j \in [t-1]$, we define the permutation $\varphi^{(j)}$ on the set $[\ell^t]$ as

$$\varphi^{(j)}(i) := \tau_t^{-1}(i_1, \dots, i_{t-j-1}, i_t, i_{t-j}, i_{t-j+1}, \dots, i_{t-1}) \quad \forall i \in [\ell^t]. \tag{5.38}$$

By this definition, $\varphi^{(j)}$ simply keeps the first $t-j-1$ digits of i to be the same and performs a cyclic shift on the last $j+1$ digits. Here we give some concrete examples:

$$\begin{aligned}
\varphi^{(1)}(i) &= \tau_t^{-1}(i_1, \dots, i_{t-2}, i_t, i_{t-1}), \\
\varphi^{(2)}(i) &= \tau_t^{-1}(i_1, \dots, i_{t-3}, i_t, i_{t-2}, i_{t-1}), \\
\varphi^{(3)}(i) &= \tau_t^{-1}(i_1, \dots, i_{t-4}, i_t, i_{t-3}, i_{t-2}, i_{t-1}), \\
\varphi^{(t-1)}(i) &= \tau_t^{-1}(i_t, i_1, i_2, \dots, i_{t-1}).
\end{aligned}$$

For each $j \in [t-1]$, let $Q^{(j)}$ be the $\ell^t \times \ell^t$ permutation matrix corresponding to the permutation $\varphi^{(j)}$, i.e., $Q^{(j)}$ is the permutation matrix such that

$$(U_1, U_2, \dots, U_{\ell^t}) Q^{(j)} = (U_{\varphi^{(j)}(1)}, U_{\varphi^{(j)}(2)}, \dots, U_{\varphi^{(j)}(\ell^t)}). \tag{5.39}$$

Finally, for each $j \in [t]$, we define the $N \times N$ matrix

$$M^{(j)} := D^{(j-1)} Q^{(j-1)} D^{(j-2)} Q^{(j-2)} \dots D^{(1)} Q^{(1)} D^{(0)}. \tag{5.40}$$

Therefore, $M^{(j)}, j \in [t]$ satisfy the following recursive relation:

$$M^{(1)} = D^{(0)}, \qquad M^{(j+1)} = D^{(j)} Q^{(j)} M^{(j)}.$$

Our encoding matrix for code length $N = \ell^t$ is the submatrix of $M^{(t)}$ consisting of all the row vectors with indices belonging to the set $\mathcal{S}_{\text{good}}$ defined in (5.35); see the next paragraph for a detailed description of the encoding procedure.

Once we obtain the matrix $M^{(t)}$ and the set $\mathcal{S}_{\text{good}}$ in the code construction, the encoding procedure is standard; it is essentially the same as the original polar codes [Arı09]. Let $\boldsymbol{U}_{[1:N]}$ be a random vector consisting of N i.i.d. Bernoulli-$1/2$ random variables, and let $\boldsymbol{X}_{[1:N]} = \boldsymbol{U}_{[1:N]} M^{(t)}$. Recall that we use $\{W_i(M^{(t)}) : i \in [\ell^t]\}$ to denote the ℓ^t bit-channels resulting from the polar transform of W using matrix $M^{(t)}$. If we transmit the random vector $\boldsymbol{X}_{[1:N]}$ through N independent copies of W and denote the channel outputs as $\boldsymbol{Y}_{[1:N]}$, then by definition, the bit-channel mapping from U_i to $(\boldsymbol{U}_{[1:i-1]}, \boldsymbol{Y}_{[1:N]})$ is exactly $W_i(M^{(t)})$. Therefore, if we use a successive cancellation decoder to decode the input

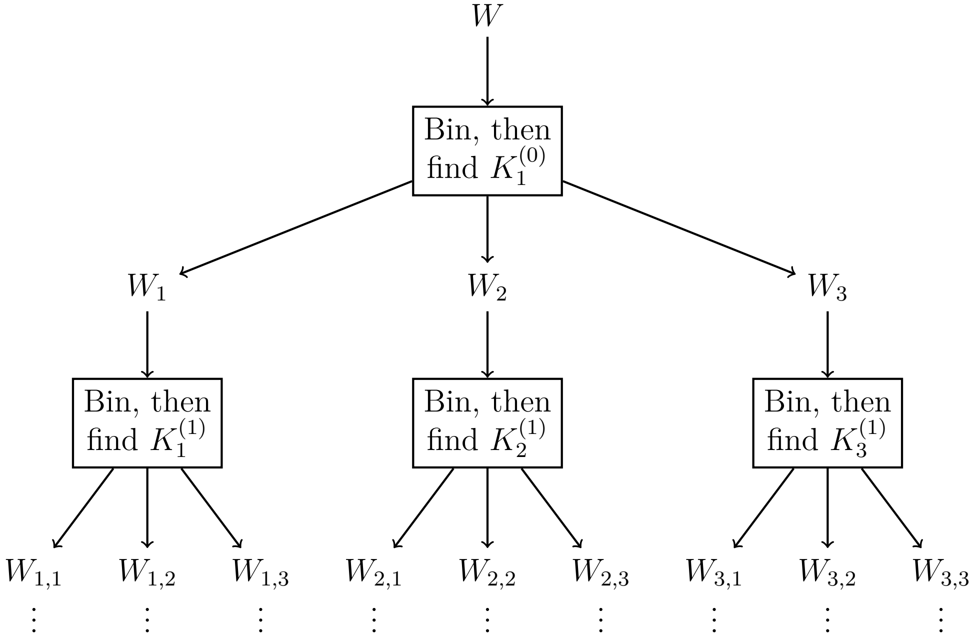

W
Bin, then find $K_1^{(0)}$
W_1
W_2
W_3
Bin, then find $K_1^{(1)}$
Bin, then find $K_2^{(1)}$
Bin, then find $K_3^{(1)}$
$W_{1,1}$
$W_{1,2}$
$W_{1,3}$
$W_{2,1}$
$W_{2,2}$
$W_{2,3}$
$W_{3,1}$
$W_{3,2}$
$W_{3,3}$

Figure 5.2: Illustration of code construction for the special case of $\ell = 3$.

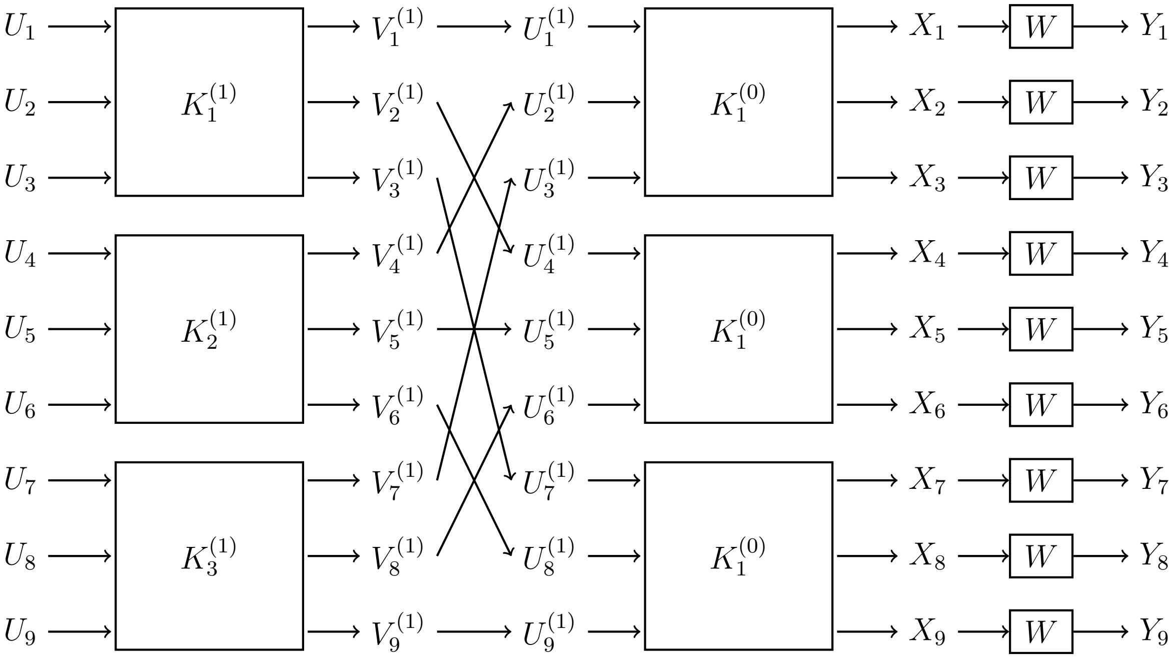


Figure 5.3: Illustration of the encoding process $\boldsymbol{X}_{[1:N]} = \boldsymbol{U}_{[1:N]}M^{(t)}$ for the special case of $\ell = 3$ and $t = 2$. Here $\boldsymbol{X}_{[1:N]}$ and $\boldsymbol{U}_{[1:N]}$ are row vectors. All four kernels in this figure $K_1^{(0)}, K_1^{(1)}, K_2^{(1)}, K_3^{(1)}$ have size 3×3, and the outputs of each kernel is obtained by multiplying the inputs with the kernel, e.g. $\boldsymbol{V}_{[1:3]}^{(1)} = \boldsymbol{U}_{[1:3]}K_1^{(1)}$.

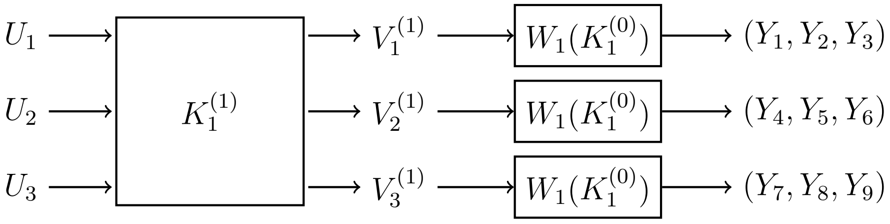


Figure 5.4: The (stochastic) mapping from $\boldsymbol{U}_{[1:3]}$ to $\boldsymbol{Y}_{[1:9]}$

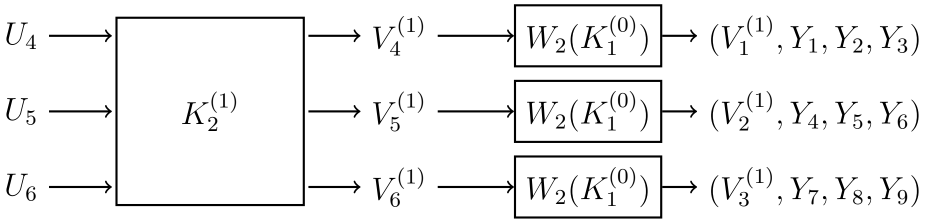


Figure 5.5: The (stochastic) mapping from $\boldsymbol{U}_{[4:6]}$ to $(\boldsymbol{V}_{[1:3]}^{(1)}, \boldsymbol{Y}_{[1:9]})$

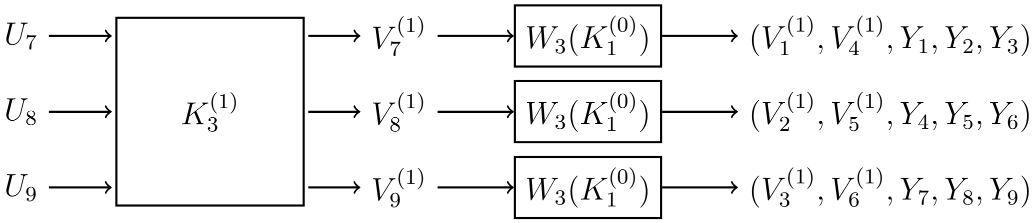

U_7
U_8
U_9
$K_3^{(1)}$
$V_7^{(1)}$
$V_8^{(1)}$
$V_9^{(1)}$
$W_3(K_1^{(0)})$
$W_3(K_1^{(0)})$
$W_3(K_1^{(0)})$
$(V_1^{(1)}, V_4^{(1)}, Y_1, Y_2, Y_3)$
$(V_2^{(1)}, V_5^{(1)}, Y_4, Y_5, Y_6)$
$(V_3^{(1)}, V_6^{(1)}, Y_7, Y_8, Y_9)$

Figure 5.6: The (stochastic) mapping from $\boldsymbol{U}_{[7:9]}$ to $(\boldsymbol{V}_{[1:6]}^{(1)}, \boldsymbol{Y}_{[1:9]})$

vector $\boldsymbol{U}_{[1:N]}$ bit by bit from all the channel outputs $\boldsymbol{Y}_{[1:N]}$ and all the previous input bits $\boldsymbol{U}_{[1:i-1]}$, then $W_i(M^{(t)})$ is the channel seen by the successive cancellation decoder when it decodes U_i. Clearly, $H(W_i(M^{(t)})) \approx 0$ means that the successive cancellation decoder can decode U_i correctly with high probability. For every $i \in \ell^t$, we write $\tau_t(i) = (i_1, i_2, \dots, i_t)$. In Proposition 5.13 below, we will show that $H(W_i(M^{(t)})) = H_{i_1,\dots,i_t}(W)$. Then in Proposition 5.15, we further show that $H_{i_1,\dots,i_t}(W) \approx H^{\text{bin}}_{i_1,\dots,i_t}(W)$. Therefore, $H(W_i(M^{(t)})) \approx H^{\text{bin}}_{i_1,\dots,i_t}(W)$. By definition (5.35), the set $\mathcal{S}_{\text{good}}$ contains all the indices $(i_1, \dots, i_t)$ for which $H^{\text{bin}}_{i_1,\dots,i_t}(W) \approx 0$, so for all i such that $\tau_t(i) \in \mathcal{S}_{\text{good}}$, we also have $H(W_i(M^{(t)})) \approx 0$, meaning that the successive cancellation decoder can decode all the bits $\{U_i : \tau_t(i) \in \mathcal{S}_{\text{good}}\}$ correctly with high probability. In the encoding procedure, we put all the information in the set of good bits $\{U_i : \tau_t(i) \in \mathcal{S}_{\text{good}}\}$, and we set all the other bits to be some pre-determined value, e.g., set all of them to be 0. It is clear that the generator matrix of this code is the submatrix of $M^{(t)}$ consisting of all the row vectors with indices belonging to the set $\mathcal{S}_{\text{good}}$.

5.4.1 Analysis of bit-channels

We say that two channels $W_1 : \{0,1\} \to \mathcal{Y}_1$ and $W_2 : \{0,1\} \to \mathcal{Y}_2$ are equivalent if there is a one-to-one mapping φ between $\mathcal{Y}_1$ and $\mathcal{Y}_2$ such that $W_1(y_1|x) = W_2(\varphi(y_1)|x)$ for all $y_1 \in \mathcal{Y}_1$ and $x \in \{0,1\}$. Denote this equivalence relation as $W_1 \equiv W_2$. Then we have the following result.

Proposition 5.13. *For every $i \in \ell^t$, we write $\tau_t(i) = (i_1, i_2, \dots, i_t)$. Then we always have*

$$W_i(M^{(t)}) \equiv W_{i_1,\dots,i_t}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K_{i_1,\dots,i_{t-1}}^{(t-1)}).$$

Before formally proving this proposition, we first use the special case of $t = 2$ and $\ell = 3$ to illustrate the main idea behind the proof. In this case, we obtained one kernel $K_1^{(0)}$ in step 0 and three kernels $K_1^{(1)}, K_2^{(1)}, K_3^{(1)}$ in step 1. See Fig. 5.3 for an illustration of the encoding process $\boldsymbol{X}_{[1:9]} = \boldsymbol{U}_{[1:9]} M^{(2)}$. In particular, we can see that

$$\boldsymbol{V}_{[1:9]}^{(1)} = \boldsymbol{U}_{[1:9]} D^{(1)}, \qquad \boldsymbol{U}_{[1:9]}^{(1)} = \boldsymbol{V}_{[1:9]}^{(1)} Q^{(1)}, \qquad X_{[1:9]} = \boldsymbol{U}_{[1:9]}^{(1)} D^{(0)}.$$

Therefore, we indeed have $\boldsymbol{X}_{[1:9]} = \boldsymbol{U}_{[1:9]} D^{(1)} Q^{(1)} D^{(0)} = \boldsymbol{U}_{[1:9]} M^{(2)}$. Assume that $\boldsymbol{U}_{[1:9]}$ consists of 9 i.i.d. Bernoulli-1/2 random variables. Since $D^{(1)}, Q^{(1)}, D^{(0)}$ are all invertible matrices, the random vectors $\boldsymbol{V}_{[1:9]}^{(1)}, \boldsymbol{U}_{[1:9]}^{(1)}$ and $\boldsymbol{X}_{[1:9]}$ also consist of i.i.d. Bernoulli-1/2 random variables.

In order to analyze the bit-channels, we view Fig. 5.3 from the right side to the left side. First, observe that the following three vectors

$$(U_1^{(1)}, U_2^{(1)}, U_3^{(1)}, Y_1, Y_2, Y_3), \qquad (U_4^{(1)}, U_5^{(1)}, U_6^{(1)}, Y_4, Y_5, Y_6), \qquad (U_7^{(1)}, U_8^{(1)}, U_9^{(1)}, Y_7, Y_8, Y_9)$$

are independent and identically distributed (i.i.d.).

Given a channel $W_1 : \mathcal{X} \to \mathcal{Y}$ and a pair of random variables (X, Y) that take values in $\mathcal{X}$ and $\mathcal{Y}$ respectively, we write

$$\mathbb{P}(X \to Y) \equiv W_1$$

if $\mathbb{P}(Y = y|X = x) = W(y|x)$ for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$, where $\mathbb{P}(X \to Y)$ means the channel that takes X as input and gives Y as output. By this definition, we have

$$\mathbb{P}(U_1^{(1)} \to \boldsymbol{Y}_{[1:3]}) \equiv \mathbb{P}(U_4^{(1)} \to \boldsymbol{Y}_{[4:6]}) \equiv \mathbb{P}(U_7^{(1)} \to \boldsymbol{Y}_{[7:9]}) \equiv W_1(K_1^{(0)}).$$

Since $V_1^{(1)} = U_1^{(1)}, V_2^{(1)} = U_4^{(1)}, V_3^{(1)} = U_7^{(1)}$, we also have

$$\mathbb{P}(V_1^{(1)} \to \boldsymbol{Y}_{[1:3]}) \equiv \mathbb{P}(V_2^{(1)} \to \boldsymbol{Y}_{[4:6]}) \equiv \mathbb{P}(V_3^{(1)} \to \boldsymbol{Y}_{[7:9]}) \equiv W_1(K_1^{(0)}).$$

Moreover, the following three vectors

$$(V_1^{(1)}, \boldsymbol{Y}_{[1:3]}), \qquad (V_2^{(1)}, \boldsymbol{Y}_{[4:6]}), \qquad (V_3^{(1)}, \boldsymbol{Y}_{[7:9]})$$

are independent. Therefore, the (stochastic) mapping from $U_{[1:3]}$ to $Y_{[1:9]}$ in Fig. 5.3 can be represented in a more compact form in Fig. 5.4. From Fig. 5.4, we can see that

$$\begin{aligned}
&W_1(M^{(2)}) \equiv \mathbb{P}(U_1 \to \mathbf{Y}_{[1:9]}) \equiv W_{1,1}(K_1^{(0)}, K_1^{(1)}),\\
&W_2(M^{(2)}) \equiv \mathbb{P}(U_2 \to (U_1, \mathbf{Y}_{[1:9]})) \equiv W_{1,2}(K_1^{(0)}, K_1^{(1)}),\\
&W_3(M^{(2)}) \equiv \mathbb{P}(U_3 \to (U_1, U_2, \mathbf{Y}_{[1:9]})) \equiv W_{1,3}(K_1^{(0)}, K_1^{(1)}).
\end{aligned}$$

Next we investigate $W_4(M^{(2)}), W_5(M^{(2)}), W_6(M^{(2)})$. Observe that

$$\mathbb{P}(U_2^{(1)} \to (U_1^{(1)}, \mathbf{Y}_{[1:3]})) \equiv \mathbb{P}(U_5^{(1)} \to (U_4^{(1)}, \mathbf{Y}_{[4:6]})) \equiv \mathbb{P}(U_8^{(1)} \to (U_7^{(1)}, \mathbf{Y}_{[7:9]})) \equiv W_2(K_1^{(0)}).$$

Therefore,

$$\mathbb{P}(V_4^{(1)} \to (V_1^{(1)}, \mathbf{Y}_{[1:3]})) \equiv \mathbb{P}(V_5^{(1)} \to (V_2^{(1)}, \mathbf{Y}_{[4:6]})) \equiv \mathbb{P}(V_6^{(1)} \to (V_3^{(1)}, \mathbf{Y}_{[7:9]})) \equiv W_2(K_1^{(0)}).$$

Moreover, since

$$(V_1^{(1)}, V_4^{(1)}, \mathbf{Y}_{[1:3]}), \qquad (V_2^{(1)}, V_5^{(1)}, \mathbf{Y}_{[4:6]}), \qquad (V_3^{(1)}, V_6^{(1)}, \mathbf{Y}_{[7:9]})$$

are independent, the (stochastic) mapping from $\boldsymbol{U}_{[4:6]}$ to $(\boldsymbol{V}_{[1:3]}^{(1)}, \mathbf{Y}_{[1:9]})$ in Fig. 5.3 can be represented in a more compact form in Fig. 5.5. Notice that there is a bijection between $\boldsymbol{U}_{[1:3]}$ and $\boldsymbol{V}_{[1:3]}^{(1)}$. Thus we can conclude from Fig. 5.5 that

$$\begin{aligned}
&W_4(M^{(2)}) \equiv \mathbb{P}(U_4 \to (\boldsymbol{U}_{[1:3]}, \mathbf{Y}_{[1:9]})) \equiv \mathbb{P}(U_4 \to (\boldsymbol{V}_{[1:3]}^{(1)}, \mathbf{Y}_{[1:9]})) \equiv W_{2,1}(K_1^{(0)}, K_2^{(1)}),\\
&W_5(M^{(2)}) \equiv \mathbb{P}(U_5 \to (\boldsymbol{U}_{[1:4]}, \mathbf{Y}_{[1:9]})) \equiv \mathbb{P}(U_5 \to (U_4, \boldsymbol{V}_{[1:3]}^{(1)}, \mathbf{Y}_{[1:9]})) \equiv W_{2,2}(K_1^{(0)}, K_2^{(1)}),\\
&W_6(M^{(2)}) \equiv \mathbb{P}(U_6 \to (\boldsymbol{U}_{[1:5]}, \mathbf{Y}_{[1:9]})) \equiv \mathbb{P}(U_6 \to (U_4, U_5, \boldsymbol{V}_{[1:3]}^{(1)}, \mathbf{Y}_{[1:9]})) \equiv W_{2,3}(K_1^{(0)}, K_2^{(1)}).
\end{aligned}$$

Finally, we can use the same method to show that

$$\begin{aligned}
&\mathbb{P}(V_7^{(1)} \to (V_1^{(1)}, V_4^{(1)}, \mathbf{Y}_{[1:3]})) \equiv \mathbb{P}(V_8^{(1)} \to (V_2^{(1)}, V_5^{(1)}, \mathbf{Y}_{[4:6]}))\\
\equiv &\mathbb{P}(V_9^{(1)} \to (V_3^{(1)}, V_6^{(1)}, \mathbf{Y}_{[7:9]})) \equiv W_3(K_1^{(0)}).
\end{aligned}$$

Therefore, the (stochastic) mapping from $\boldsymbol{U}_{[7:9]}$ to $(\boldsymbol{V}_{[1:6]}^{(1)}, \mathbf{Y}_{[1:9]})$ in Fig. 5.3 can be represented in a more compact form in Fig. 5.6. Notice that there is a bijection between $\boldsymbol{U}_{[1:6]}$ and $\boldsymbol{V}_{[1:6]}^{(1)}$. Thus we can conclude from Fig. 5.6 that

$$\begin{aligned}
&W_7(M^{(2)}) \equiv \mathbb{P}(U_7 \to (\boldsymbol{U}_{[1:6]}, \mathbf{Y}_{[1:9]})) \equiv \mathbb{P}(U_7 \to (\boldsymbol{V}_{[1:6]}^{(1)}, \mathbf{Y}_{[1:9]})) \equiv W_{3,1}(K_1^{(0)}, K_3^{(1)}),\\
&W_8(M^{(2)}) \equiv \mathbb{P}(U_8 \to (\boldsymbol{U}_{[1:7]}, \mathbf{Y}_{[1:9]})) \equiv \mathbb{P}(U_8 \to (U_7, \boldsymbol{V}_{[1:6]}^{(1)}, \mathbf{Y}_{[1:9]})) \equiv W_{3,2}(K_1^{(0)}, K_3^{(1)}),\\
&W_9(M^{(2)}) \equiv \mathbb{P}(U_9 \to (\boldsymbol{U}_{[1:8]}, \mathbf{Y}_{[1:9]})) \equiv \mathbb{P}(U_9 \to (U_7, U_8, \boldsymbol{V}_{[1:6]}^{(1)}, \mathbf{Y}_{[1:9]})) \equiv W_{3,3}(K_1^{(0)}, K_3^{(1)}).
\end{aligned}$$

Now we have proved Proposition 5.13 for the special case of $\ell = 3$ and $t = 2$. The proof for the general case follows the same idea, and we defer it to Section 5.6 at the end of this chapter.

5.4.2 Complexity of code construction, encoding and decoding

Proposition 5.14. *The code construction has $N^{O_\ell(1)}$ complexity. Both the encoding and successive decoding procedures have $O_\ell(N \log N)$ complexity.*

Proof. The key in our proof is that we consider ℓ as a (possibly very large) constant. We start with the code construction and we first show that both Algorithm A and Algorithm B have poly(N) time complexity. In the worst case, we need to check all 2^{ℓ^2} possible kernels in Algorithm A, and for each kernel we need to calculate the conditional entropy of the ℓ subchannels. Since we always work with the quantized channel with output size upper bounded by N^3, each subchannel of the quantized channels has no more than $2^\ell N^{3\ell}$ outputs. Therefore, the conditional entropy of these subchannels can be calculated in poly(N) time, so Algorithm A also has poly(N) complexity. After finding the good kernels, we need to use Algorithm B to quantize/bin the output alphabet of the subchannels produced by these good kernels. As mentioned above, the original alphabet size of these subchannels is no more than $2^\ell N^{3\ell}$. Therefore, Algorithm B also has poly(N) complexity. At Step i, we use Algorithm A ℓ^i times to find good kernels, and then we use Algorithm B ℓ^{i+1} times to quantize the bit-channels produced by these kernels, so in total we use Algorithm A $\frac{N-1}{\ell-1}$ times and we use Algorithm B $\frac{\ell(N-1)}{\ell-1}$ times. Finally, finding the set $\mathcal{S}_{\text{good}}$ only requires calculating the conditional entropy of the bit-channels in the last step, so this can also be done in polynomial time. Thus we conclude that the code construction has poly(N) complexity, albeit the degree in poly(N) complexity depends on ℓ.

In the encoding procedure, we first form the vector $\boldsymbol{U}_{[1:N]}$ by putting all the information in the bits $\{U_i : \tau_t(i) \in \mathcal{S}_{\text{good}}\}$ and setting all the other bits $\{U_i : \tau_t(i) \notin \mathcal{S}_{\text{good}}\}$ to be 0. Then we multiply $\boldsymbol{U}_{[1:N]}$ with the encoding matrix $M^{(t)}$ and obtain the codeword $\boldsymbol{X}_{[1:N]} = \boldsymbol{U}_{[1:N]} M^{(t)}$. Since the matrix $M^{(t)}$ has size $N \times N$, a naive implementation of the encoding procedure would require $O(N^2)$ operations. Fortunately, we can use (5.40) to accelerate the encoding procedure. Namely, we first multiply $\boldsymbol{U}_{[1:N]}$ with $D^{(t-1)}$, then multiply the result with $Q^{(t-1)}$, then multiply by $D^{(t-2)}$, so on and so forth. As mentioned above, for $j = 0, 1, \dots, t-1$, each $D^{(j)}$ is a block diagonal matrix with N/ℓ blocks on the diagonal, where each block has size $\ell \times \ell$. Therefore, multiplication with $D^{(j)}$ only requires $N\ell$ operations. By definition, $Q^{(j)}, j \in [t-1]$ are permutation matrices, so multiplication with them only requires N operations. In total, we multiply with $2t - 1 = 2\log_\ell N - 1$ matrices. Therefore, the encoding procedure can be computed in $O_\ell(N \log N)$ time, where O_ℓ means that the constant in big-O depends on ℓ.

The decoding algorithm uses exactly the same idea as the algorithm in Arıkan's original paper [Arı09, Section VIII-B]. Here we only use the special case of $\ell = 3$ and $t = 2$ in Fig. 5.3 to explain how Arıkan's decoding algorithm works for large (and mixed) kernels, and we omit the proof for general parameters. We start with the decoding of U_1, U_2, U_3 in Fig. 5.3. It is clear that decoding U_1, U_2, U_3 is equivalent to decoding $U_1^{(1)}, U_4^{(1)}, U_7^{(1)}$. Then the log-likelihood ratio (LLR) of each of these three bits can be calculated locally from only three output symbols. More precisely, the LLR of $U_1^{(1)}$ can be computed from $\mathbf{Y}_{[1:3]}$, the LLR of $U_4^{(1)}$ can be computed from $\mathbf{Y}_{[4:6]}$, and the LLR of $U_7^{(1)}$ can be computed from $\mathbf{Y}_{[7:9]}$.

Therefore, the complexity of calculating each LLR only depends on the value of ℓ. Since ℓ is considered as a constant, the calculation of each LLR also has constant time complexity (although the complexity is exponential in ℓ). The next step is to decode $\boldsymbol{U}_{[4:6]}$ from $\boldsymbol{Y}_{[1:9]}$ together with $\boldsymbol{U}_{[1:3]}$. This is equivalent to calculating the LLRs of $U_2^{(1)}, U_5^{(1)}, U_8^{(1)}$ given $\boldsymbol{Y}_{[1:9]}$ and $U_1^{(1)}, U_4^{(1)}, U_7^{(1)}$. This again can be done locally: To compute the LLR of $U_2^{(1)}$, we only need the values of $\boldsymbol{Y}_{[1:3]}$ and $U_1^{(1)}$; to compute the LLR of $U_5^{(1)}$, we only need the values of $\boldsymbol{Y}_{[4:6]}$ and $U_4^{(1)}$; to compute the LLR of $U_8^{(1)}$, we only need the values of $\boldsymbol{Y}_{[7:9]}$ and $U_7^{(1)}$. Finally, the decoding of $\boldsymbol{U}_{[7:9]}$ from $\boldsymbol{Y}_{[1:9]}$ and $\boldsymbol{U}_{[1:6]}$ can be decomposed into local computations in a similar way. Using this idea, one can show that for general values of ℓ and t, the decoding can also be decomposed into $t = \log_\ell N$ stages, and in each stage, the decoding can further be decomposed into N/ℓ local tasks, each of which has constant time complexity (although the complexity is exponential in ℓ). Therefore, the decoding complexity at each stage is $O_\ell(N)$, and the overall decoding complexity is $O_\ell(N \log N)$. As a final remark, we mention that after calculating the LLRs of all U_i's, we will only use the LLRs of the bits $\{U_i : \tau_t(i) \in \mathcal{S}_{\text{good}}\}$. For these bits, we decode U_i as 0 if its LLR is larger than 0 and decode it 1 otherwise. Recall that in the encoding procedure, we have set all the other bits $\{U_i : \tau_t(i) \notin \mathcal{S}_{\text{good}}\}$ to be 0, so for these bits we simply decode them as 0. □

5.5 Putting everything together

5.5.1 Code rate and decoding error probability

In (5.34), we have defined the conditional entropy for all the bit-channels obtained in the last step (Step $t-1$). Here we also define the conditional entropy for the bit-channels obtained in the previous steps. More precisely, for every $j \in [t]$ and every $(i_1, i_2, \dots, i_j) \in [\ell]^j$, we use the following short-hand notation:

$$\begin{aligned}
H_{i_1,\dots,i_j}(W) &:= H(W_{i_1,\dots,i_j}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K_{i_1,\dots,i_{j-1}}^{(j-1)})) \\
H^{\text{bin}}_{i_1,\dots,i_j}(W) &:= H(W^{\text{bin}}_{i_1,\dots,i_j}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K_{i_1,\dots,i_{j-1}}^{(j-1)})) \\
H^{\text{bin}*}_{i_1,\dots,i_j}(W) &:= H(W^{\text{bin}*}_{i_1,\dots,i_j}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K_{i_1,\dots,i_{j-1}}^{(j-1)})).
\end{aligned}$$

According to (5.33), we have

$$H^{\text{bin}*}_{i_1,\dots,i_j}(W) \le H^{\text{bin}}_{i_1,\dots,i_j}(W) \le H^{\text{bin}*}_{i_1,\dots,i_j}(W) + \frac{6 \log N}{N^3} \tag{5.41}$$

for every $j \in [t]$ and every $(i_1, i_2, \dots, i_j) \in [\ell]^j$.

Proposition 5.15. *For every $j \in [t]$ and $(i_1, i_2, \dots, i_j) \in [\ell]^j$, the conditional entropy $H_{i_1,\dots,i_j}(W)$ and $H^{\text{bin}}_{i_1,\dots,i_j}(W)$ satisfy the following inequality*

$$H_{i_1,\dots,i_j}(W) \le H^{\text{bin}}_{i_1,\dots,i_j}(W) \le H_{i_1,\dots,i_j}(W) + \frac{6\ell \log N}{N^2} \tag{5.42}$$

Proof. Since the binning algorithm (Algorithm B) always produces a channel that is degraded with respect to the original channel, the first inequality in (5.42) follows immediately by applying Proposition 5.3 recursively in our t-step code construction.

Now we prove the second inequality in (5.42). We will prove the following inequality by induction on j:

$$H^{\text{bin}}_{i_1,\dots,i_j}(W) \le H_{i_1,\dots,i_j}(W) + \frac{6\log N}{N^3}(1+\ell+\ell^2+\cdots+\ell^j) \qquad \forall (i_1,i_2,\dots,i_j)\in[\ell]^j. \quad (5.43)$$

The base case of $j=0$ is trivial. Now assume that this inequality holds for j and we prove it for $j+1$. By the chain rule, we know that

$$\sum_{i_{j+1}=1}^{\ell} H^{\text{bin}\,*}_{i_1,\dots,i_j,i_{j+1}}(W) = \ell H^{\text{bin}}_{i_1,\dots,i_j}(W), \qquad \sum_{i_{j+1}=1}^{\ell} H_{i_1,\dots,i_j,i_{j+1}}(W) = \ell H_{i_1,\dots,i_j}(W).$$

Therefore,

$$\sum_{i_{j+1}=1}^{\ell}\Big(H^{\text{bin}\,*}_{i_1,\dots,i_j,i_{j+1}}(W) - H_{i_1,\dots,i_j,i_{j+1}}(W)\Big) = \ell\Big(H^{\text{bin}}_{i_1,\dots,i_j}(W) - H_{i_1,\dots,i_j}(W)\Big).$$

Since every summand on the left-hand side is non-negative, we have

$$H^{\text{bin}\,*}_{i_1,\dots,i_j,i_{j+1}}(W) - H_{i_1,\dots,i_j,i_{j+1}}(W) \le \ell\Big(H^{\text{bin}}_{i_1,\dots,i_j}(W) - H_{i_1,\dots,i_j}(W)\Big) \le \frac{6\log N}{N^3}(\ell+\ell^2+\cdots+\ell^{j+1}),$$

where the second inequality follows from the induction hypothesis. Combining this with (5.41), we obtain that

$$H^{\text{bin}}_{i_1,\dots,i_j,i_{j+1}}(W) \le H_{i_1,\dots,i_j,i_{j+1}}(W) + \frac{6\log N}{N^3}(1+\ell+\ell^2+\cdots+\ell^{j+1}).$$

This establishes the inductive step and completes the proof of (5.43). The inequality (5.42) then follows directly from (5.43) by using the fact that $1+\ell+\cdots+\ell^j < \ell N$ for all $j \le t$. □

Recall that in Remark 5.4 we denoted by $\ell \ge \exp(\Omega(\alpha^{-1.01}))$ the conditions on ℓ to be large enough so that $\log \ell \ge \frac{11}{\alpha}$ and $\frac{\log \ell}{\log\log \ell + 2} \ge \frac{3}{\alpha}$. In the theorems below, even though the statements hold for any $\alpha \in (0, 1/12)$, we modify the intervals of α so that the rate appears positive in the formulations. This is also why in the formulation of the Theorem 5.1 we take α from $(0, 1/36)$.

We now can formulate

Theorem 5.16. *For arbitrarily small $\alpha \in \left(0, \frac{1}{14}\right)$, if we choose a large enough constant $\ell \ge \exp(\Omega(\alpha^{-1.01}))$ to be a power of 2 and let $t = \log_\ell N$ grow, then the codes constructed from the above procedure have decoding error probability $O_\alpha(\log N/N)$ under successive decoding and code rate $I(W) - N^{-1/2+7\alpha}$, where $N = \ell^t$ is the code length.*

Proof. By (5.42) and the definition of $\mathcal{S}_{\text{good}}$ in (5.35), we know that for every $(i_1, \ldots, i_t) \in \mathcal{S}_{\text{good}}$, we have $H_{i_1,\ldots,i_t}(W) \le H^{\text{bin}}_{i_1,\ldots,i_t}(W) \le \frac{7\ell \log N}{N^2}$. Then by Lemma 2.2 in [BGN+18], we know that the decoding error probability of the channel $W_{i_1,\ldots,i_t}(K_1^{(0)}, K_{i_1}^{(1)}, \ldots, K_{i_1,\ldots,i_{t-1}}^{(t-1)})$ is also upper bounded by $\frac{7\ell \log N}{N^2}$ (under ML decoding). Since the cardinality of $\mathcal{S}_{\text{good}}$ is at most N, we can conclude that the overall decoding error probability under the successive cancellation decoder is $O_\alpha(\log N/N)$ using the union bound.

Notice that $|\mathcal{S}_{\text{good}}|$ is the code dimension. Therefore, we only need to lower bound $|\mathcal{S}_{\text{good}}|$ in order to get the lower bound on the code rate. Define another set

$$\mathcal{S}'_{\text{good}} := \left\{ (i_1, i_2, \ldots, i_t) \in [\ell]^t : H_{i_1,\ldots,i_t}(W) \le \frac{\ell \log N}{N^2} \right\}. \tag{5.44}$$

According to (5.42), if $H_{i_1,\ldots,i_t}(W) < \frac{\ell \log N}{N^2}$, then $H^{\text{bin}}_{i_1,\ldots,i_t}(W) \le \frac{7\ell \log N}{N^2}$. Therefore, $\mathcal{S}'_{\text{good}} \subseteq \mathcal{S}_{\text{good}}$, so $|\mathcal{S}_{\text{good}}| \ge |\mathcal{S}'_{\text{good}}|$. In Lemma 5.17 below, we will prove that $|\mathcal{S}'_{\text{good}}| \ge N(I(W) - N^{-1/2+7\alpha})$. Therefore, $|\mathcal{S}_{\text{good}}| \ge N(I(W) - N^{-1/2+7\alpha})$. This completes the proof of the theorem. □

Lemma 5.17. *If $\alpha \in \left(0, \frac{1}{14}\right)$ and ℓ is large enough so that $\log \ell \ge \frac{11}{\alpha}$ and $\frac{\log \ell}{\log\log \ell + 2} \ge \frac{3}{\alpha}$, then the set $\mathcal{S}'_{\text{good}}$ defined in (5.44) satisfies the following inequality*

$$\left|\mathcal{S}'_{\text{good}}\right| \ge N\left(I(W) - N^{-\frac{1}{2}+7\alpha}\right).$$

Proof. Recall that we proved in (5.3)–(5.6)

$$\mathbb{P}\left[\mathsf{H}_t \in \left(\frac{\ell \log N}{N^2}, 1 - \frac{\ell \log N}{N^2}\right)\right] \le 2\frac{N^{2\alpha}}{(\ell \log N)^\alpha} \cdot \lambda_\alpha^t,$$

where H_t is (marginally) the entropy of the random channel at the last level of construction, i.e. H_t is uniformly distributed over $H_{i_1,\ldots,i_t}(W)$ for all possible $(i_1, i_2, \ldots, i_t) \in [\ell]^t$, and λ_α is such that (5.4) holds for any channel W' throughout the construction. By Proposition 5.15, we can choose the error parameter Δ in Algorithm A to be $\Delta = \frac{6\ell \log N}{N^2}$, which satisfies the condition $\Delta \le \ell^{-\log \ell}$ in Theorem 5.2. Then Theorem 5.2 tells us that as long as the conditions on ℓ and α specified in this lemma hold, Algorithm A allows us to choose kernels such that $\lambda_\alpha \le \ell^{-1/2+5\alpha}$, which gives

$$\mathbb{P}\left[\mathsf{H}_t \in \left(\frac{\ell \log N}{N^2}, 1 - \frac{\ell \log N}{N^2}\right)\right] \le \frac{2N^{-1/2+7\alpha}}{(\ell \log N)^\alpha}. \tag{5.45}$$

On the other hand, conservation of entropy throughout the process implies $E\left[\mathsf{H}_t\right] = H(W)$, therefore by Markov's inequality

$$\mathbb{P}\left[\mathsf{H}_t \ge 1 - \frac{\ell \log N}{N^2}\right] \le \frac{H(W)}{1 - \frac{\ell \log N}{N^2}} \le H(W) + \frac{2\ell \log N}{N^2}.$$

Since $H(W) = 1 - I(W)$ for symmetric channels and $\left|\mathcal{S}'_{\text{good}}\right| = N \cdot \mathbb{P}\left[\mathsf{H}_t \le \frac{\ell \log N}{N^2}\right]$, we have

$$\begin{aligned}\left|\mathcal{S}'_{\text{good}}\right| &\ge N\left(1 - \frac{2N^{-1/2+7\alpha}}{(\ell \log N)^\alpha} - H(W) - \frac{2\ell \log N}{N^2}\right)\\ &\ge N\left(I(W) - \frac{3N^{-1/2+7\alpha}}{(\ell \log N)^\alpha}\right)\\ &\ge N\Big(I(W) - N^{-1/2+7\alpha}\Big).\end{aligned}$$ □

5.5.2 Main theorem

As we mentioned at the beginning of this section, the code construction presented above only takes the special case of $\mathsf{Q} = N^3$ as a concrete example, where Q is the upper bound on the output alphabet size after binning; see Algorithm B. In fact, we can change the value of Q to be any polynomial of N, and this allows us to obtain a trade-off between the decoding error probability and the gap to capacity while maintaining the polynomial-time code construction as well as the $O_\alpha(N \log N)$ encoding and decoding complexity. That is, we prove the Theorem 5.1 teased at the beginning of this chapter, restated below for convenience:

Theorem 5.1. *Let W be an arbitrary BMS channel with Shannon capacity $I(W)$ and fix any $c > 0$. For any desired $\alpha \in \left(0, \frac{1}{12+2c}\right)$, if we choose a large enough constant $\ell \ge \ell_0(\alpha)$ to be a power of 2, then there is a code $\mathcal{C}$ generated by the polar coding construction using kernels of size $\ell \times \ell$ such that the following four properties hold when N is the code length:*

1. *the code construction has $N^{O_\alpha(1)}$ complexity;*
2. *both encoding and decoding have $O_\alpha(N \log N)$ complexity;*
3. *the rate of $\mathcal{C}$ is $I(W) - O(N^{-1/2+(c+6)\alpha})$; and*
4. *the decoding error probability of $\mathcal{C}$ is $O_\alpha(\log N / N^c)$ under successive cancellation decoding when $\mathcal{C}$ is used for channel coding over W.*

The value for $\ell_0(\alpha)$ is the smallest number which satisfies $\log \ell_0 \ge \frac{11}{\alpha}$ and $\frac{\log \ell_0}{\log\log \ell_0 + 2} \ge \frac{3}{\alpha}$, and does not depend on the channel W.

Proof. We set $\mathsf{Q} = N^{c+2}$ (instead of N^3 as before). Properties (1) and (2) follow from Proposition 5.14. Here we explain how to adjust the proof of Theorem 5.16 to show properties (3) and (4). First, we change the definitions of $\mathcal{S}_{\text{good}}$ and $\mathcal{S}'_{\text{good}}$ to

$$\mathcal{S}_{\text{good}} := \left\{(i_1, i_2, \ldots, i_t) \in [\ell]^t : H^{\text{bin}}_{i_1,\ldots,i_t}(W) \le \frac{(2c+3)\ell \log N}{N^{c+1}}\right\},$$

$$\mathcal{S}'_{\text{good}} := \left\{(i_1, i_2, \ldots, i_t) \in [\ell]^t : H_{i_1,\ldots,i_t}(W) \le \frac{\ell \log N}{N^{c+1}}\right\}.$$

The definition of $\mathcal{S}_{\text{good}}$ immediately implies property (4). Next we prove property (3).

Since we change Q from N^3 to N^{c+2}, inequality (5.41) becomes

$$H^{\mathrm{bin}\,*}_{i_1,\dots,i_j}(W) \le H^{\mathrm{bin}}_{i_1,\dots,i_j}(W) \le H^{\mathrm{bin}\,*}_{i_1,\dots,i_j}(W) + \frac{2(c+2)\log N}{N^{c+2}}.$$

As a consequence, inequality (5.42) in Proposition 5.15 becomes

$$H_{i_1,\dots,i_j}(W) \le H^{\mathrm{bin}}_{i_1,\dots,i_j}(W) \le H_{i_1,\dots,i_j}(W) + \frac{2(c+2)\ell\log N}{N^{c+1}}.$$

This inequality tells us that $\mathcal{S}'_{\mathrm{good}} \subseteq \mathcal{S}_{\mathrm{good}}$, so $|\mathcal{S}_{\mathrm{good}}| \ge |\mathcal{S}'_{\mathrm{good}}|$. Then we follow Lemma 5.17 to lower bound $|\mathcal{S}'_{\mathrm{good}}|$. Inequality (5.45) now becomes

$$\mathbb{P}\left[\mathsf{H}_t \in \left(\frac{\ell\log N}{N^{c+1}}, 1-\frac{\ell\log N}{N^{c+1}}\right)\right] \le \frac{2N^{-1/2+(c+6)\alpha}}{(\ell\log N)^{\alpha}}.$$

Therefore, we obtain that

$$|\mathcal{S}_{\mathrm{good}}| \ge |\mathcal{S}'_{\mathrm{good}}| \ge N\Big(I(W) - N^{-1/2+(c+6)\alpha}\Big).$$

This completes the proof of the main theorem of this chapter. □

5.6 Deferred proof of Proposition 5.13

We still use $\boldsymbol{U}_{[1:N]}$ to denote the information vector and use $\boldsymbol{X}_{[1:N]} = \boldsymbol{U}_{[1:N]}M^{(t)}$ to denote the encoded vector. Assume that $\boldsymbol{U}_{[1:N]}$ consists of N i.i.d. Bernoulli-1/2 random variables. Similarly to the example in Section 5.4.1, we define the random vectors $\boldsymbol{V}^{(j)}_{[1:N]}, \boldsymbol{U}^{(j)}_{[1:N]}$ for $j = t-1, t-2, \dots, 1$ recursively

$$\begin{aligned}
\boldsymbol{V}^{(t-1)}_{[1:N]} &= \boldsymbol{U}_{[1:N]}D^{(t-1)}, \\
\boldsymbol{U}^{(j)}_{[1:N]} &= \boldsymbol{V}^{(j)}_{[1:N]}Q^{(j)} \text{ for } j = t-1, t-2, \dots, 1, \\
\boldsymbol{V}^{(j)}_{[1:N]} &= \boldsymbol{U}^{(j+1)}_{[1:N]}D^{(j)} \text{ for } j = t-2, t-3, \dots, 1, \\
\boldsymbol{X}_{[1:N]} &= \boldsymbol{U}^{(1)}_{[1:N]}D^{(0)}.
\end{aligned} \tag{5.46}$$

Moreover, let $\boldsymbol{U}^{(t)}_{[1:N]} := \boldsymbol{U}_{[1:N]}$. We will prove the following two claims:

1. For every $a = 1, 2, \dots, t$, the following ℓ^{t-a} random vectors

   $$(\boldsymbol{U}^{(a)}_{[h\ell^a+1:h\ell^a+\ell^a]}, \boldsymbol{Y}_{[h\ell^a+1:h\ell^a+\ell^a]}), \quad h = 0, 1, \dots, \ell^{t-a}-1$$

   are i.i.d.

2. For every $a = 1, 2, \dots, t$ and every $i \in [\ell^a]$, we write $\tau_a(i) = (i_1, i_2, \dots, i_a)$, where τ_a is the a-digit expansion function defined in (5.36). Then for every $h = 0, 1, \dots, \ell^{t-a}-1$ and every $i \in [\ell^a]$, we have

   $$\mathbb{P}(U^{(a)}_{h\ell^a+i} \to (\boldsymbol{U}^{(a)}_{[h\ell^a+1:h\ell^a+i-1]}, \boldsymbol{Y}_{[h\ell^a+1:h\ell^a+\ell^a]})) \equiv W_{i_1,\dots,i_a}(K^{(0)}_1, K^{(1)}_{i_1}, \dots, K^{(a-1)}_{i_1,\dots,i_{a-1}}). \tag{5.47}$$

Note that Proposition 5.13 follows immediately from taking $a = t$ in (5.47). Therefore, we only need to prove these two claims.

We start with the first claim. By (5.37), for every $j = 0, 1, \dots, t-1$, the matrix $D^{(j)}$ is a block diagonal matrix with ℓ^{t-j-1} blocks on the diagonal, where each block has size $\ell^{j+1} \times \ell^{j+1}$, and all the ℓ^{t-j-1} blocks are the same. According to (5.38)–(5.39), the permutation matrix $Q^{(j)}$ keeps the first $t-j-1$ digits of the ℓ-ary expansion to be the same and performs a cyclic shift on the last $j+1$ digits. Therefore, for every $j = 1, \dots, t-1$, the permutation matrix $Q^{(j)}$ is also a block diagonal matrix with ℓ^{t-j-1} blocks on the diagonal, where each block has size $\ell^{j+1} \times \ell^{j+1}$, and all the ℓ^{t-j-1} blocks are the same. Therefore, for every $j \in [t]$, the matrix $M^{(j)}$ defined in (5.40) can be written in the following block diagonal form

$$M^{(j)} := \underbrace{\{\overline{M}^{(j)}, \overline{M}^{(j)}, \dots, \overline{M}^{(j)}\}}_{\text{number of } \overline{M}^{(j)} \text{ is } \ell^{t-j}}, \tag{5.48}$$

where the size of $\overline{M}^{(j)}$ is $\ell^j \times \ell^j$. By the recursive definition (5.46), one can show that for every $j \in [t]$, we have

$$\boldsymbol{X}_{[1:N]} = \boldsymbol{U}_{[1:N]}^{(j)} M^{(j)}.$$

Combining this with (5.48), we obtain that for every $a \in [t]$ and every $h = 0, 1, \dots, \ell^{t-a} - 1$,

$$\boldsymbol{X}_{[h\ell^a+1:h\ell^a+\ell^a]} = \boldsymbol{U}_{[h\ell^a+1:h\ell^a+\ell^a]}^{(a)} \overline{M}^{(a)}. \tag{5.49}$$

Since $\boldsymbol{X}_{[1:N]}$ consists of N i.i.d. Bernoulli-1/2 random variables, the following ℓ^{t-a} random vectors

$$(\boldsymbol{X}_{[h\ell^a+1:h\ell^a+\ell^a]}, \boldsymbol{Y}_{[h\ell^a+1:h\ell^a+\ell^a]}), \quad h = 0, 1, \dots, \ell^{t-a} - 1$$

are i.i.d. Combining this with (5.49), we conclude that the random vectors

$$(\boldsymbol{U}_{[h\ell^a+1:h\ell^a+\ell^a]}^{(a)}, \boldsymbol{Y}_{[h\ell^a+1:h\ell^a+\ell^a]}), \quad h = 0, 1, \dots, \ell^{t-a} - 1$$

are also i.i.d. This proves claim 1.

Next, we prove claim 2 by induction. The case of $a = 1$ is trivial. Now we assume that (5.47) holds for a and prove it for $a+1$. In light of claim 1, we only need to prove (5.47) for the special case of $h = 0$ because the distributions for different values of h are identical, i.e. we only need to prove that

$$\mathbb{P}(U_i^{(a+1)} \to (\boldsymbol{U}_{[1:i-1]}^{(a+1)}, \boldsymbol{Y}_{[1:\ell^{a+1}]})) \equiv W_{i_1,\dots,i_{a+1}}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K_{i_1,\dots,i_a}^{(a)}) \qquad \forall i \in [\ell^{a+1}]. \tag{5.50}$$

For a given $i \in [\ell^{a+1}]$, we write its $(a+1)$-digit expansion as $\tau_{a+1}(i) = (i_1, i_2, \dots, i_{a+1})$. By (5.46), we know that $\boldsymbol{V}_{[1:N]}^{(a)} = \boldsymbol{U}_{[1:N]}^{(a+1)} D^{(a)}$. By (5.37), the matrix $D^{(a)}$ is a block diagonal matrix with ℓ^{t-1} blocks on the diagonal, where each block has size $\ell \times \ell$. (Note that these ℓ^{t-1} blocks are not necessarily all the same unless $a = 0$.) Therefore, for every $h = 0, 1, \dots, \ell^{t-1} - 1$, there is a bijection between the two vectors $\boldsymbol{V}_{[h\ell+1:h\ell+\ell]}^{(a)}$ and

$\boldsymbol{U}^{(a+1)}_{[h\ell+1:h\ell+\ell]}$. Consequently, there is a bijection between the two vectors $\boldsymbol{U}^{(a+1)}_{[1:i-i_{a+1}]}$ and $\boldsymbol{V}^{(a)}_{[1:i-i_{a+1}]}$, so we have

$$\mathbb{P}(U_i^{(a+1)} \to (\boldsymbol{U}^{(a+1)}_{[1:i-1]}, \boldsymbol{Y}_{[1:\ell^{a+1}]})) \equiv \mathbb{P}(U_i^{(a+1)} \to (\boldsymbol{U}^{(a+1)}_{[i-i_{a+1}+1:i-1]}, \boldsymbol{V}^{(a)}_{[1:i-i_{a+1}]}, \boldsymbol{Y}_{[1:\ell^{a+1}]})). \tag{5.51}$$

By (5.37), we also have that

$$\boldsymbol{V}^{(a)}_{[i-i_{a+1}+1:i-i_{a+1}+\ell]} = \boldsymbol{U}^{(a+1)}_{[i-i_{a+1}+1:i-i_{a+1}+\ell]} K^{(a)}_{i_1,i_2,\dots,i_a}. \tag{5.52}$$

Let $\hat{i} := (i - i_{a+1})/\ell$, so $\tau_a(\hat{i}) = (i_1, i_2, \dots, i_a)$. According to the induction hypothesis,

$$\mathbb{P}(U_{\hat{i}}^{(a)} \to (\boldsymbol{U}^{(a)}_{[1:\hat{i}-1]}, \boldsymbol{Y}_{[1:\ell^a]})) \equiv W_{i_1,\dots,i_a}(K_1^{(0)}, K_{i_1}^{(1)}, \dots, K^{(a-1)}_{i_1,\dots,i_{a-1}}).$$

Combining this with the relation $\boldsymbol{U}^{(a)}_{[1:N]} = \boldsymbol{V}^{(a)}_{[1:N]} Q^{(a)}$ and (5.51)–(5.52), we can prove (5.50) with the ideas illustrated in Fig. 5.4–5.6. This completes the proof of claim 2 as well as Proposition 5.13.

Chapter 6

Near-Optimal Decoding Error Probability

In this section we show how to adjust our construction to obtain inverse sub-exponential $\exp(-N^{\alpha})$ probability of error decoding, while still having $\mathrm{poly}(N)$ time complexity of construction. Note that up to this point we only claimed inverse-polynomial decoding error probability in Theorem 5.1. This restriction came from the fact that we need to approximate the channels we see in the tree during the construction phase (recall the discussion at the beginning of Sections 5.2.1 and 5.4), and to get a polynomial-time construction we need the binning parameter Q to be $\mathrm{poly}(N)$ itself. But this means that we are only able to track the parameters (entropies, for instance) of the bit-channels approximately, with an additive error which is inverse-polynomial in N, see (5.41). Since the decoding error probability relates directly to the upper bound on the entropies of the "good" bit-channels we choose, this leads to only being able to claim inverse-polynomial decoding error probability.

It was proved in a recent work [WD19] that it is possible to achieve a fast scaling of polar codes (good scaling exponent) and a good decoding error probability (inverse sub-exponential instead of inverse-polynomial in N) simultaneously, also using the idea of multiple (dynamic) random kernels in the construction. Specifically, for any constants $\varphi, \mu > 0$ such that $\varphi + 2/\mu < 1$, it is shown that one can construct a polar code with the rate $N^{-1/\mu}$ close to capacity of the channel (i.e. the scaling exponent μ) and the decoding error probability $\exp(-N^{\varphi})$, as $N \to \infty$. Moreover, it is known that this is an optimal scaling of these two parameters one can obtain for *any* (not just polar) codes (for coding over non-trivial BMS channels). However, the construction phase in [WD19] tracked the *true* bit-channels that are obtained in the ℓ-ary tree of channels, which makes the construction intractable. This is because (most of) the true bit-channels cannot even be described in a tractable way, since they have exponential sizes of output alphabet.

In what follows we show that we can apply a very strong analysis of polarization from [WD18a, WD19] to our codes to overcome this issue of intractable construction. A nuance of our codes that we utilize is that we use a fixed kernel A_2 for suction at the ends regime. We can then ensure that even though we track only *approximations* (binned versions) of the bit-channels in the tree, we are still able to prove very strong polarization if we use the analysis from [WD19]. This comes from the fact that we know very well how Arıkan's basic 2×2 kernel evolves the parameters of the bit-channels. This allows us to get very strong bounds on the parameters of the *true* bit-channels (which leads to good decoding error probability), while still only tracking their *approximations* (which keeps the construction time polynomial). This approach is in fact quite standard for the standard ($A_2^{\otimes t}$) polar codes, when one aims at sub-exponentially small decoding error probability but wishes to keep the poly-time construction, see for example [TV13, GX13]. However, the challenge here consists in not losing the quality of the scaling exponent while we are trying to capture better P_e. Somewhat surprisingly, the construction phase for our codes where the local kernels are chosen is exactly the same as it was before in Section 5.4, and the difference lies in a much tighter analysis of how to choose a set of "good" indices to actually construct a polar code.

Without stating the theorem, here is what we prove in this section: for any BMS channel W and $\alpha \in \left(0, \frac{1}{36}\right)$, for large enough constant $\ell = 2^s$, we can build the codes in $\text{poly}(N)$ time such that $R \geq I(W) - N^{-1/2+18\alpha}$ and $P_e \leq \exp(-N^{\alpha})$, and for which we have $O_\alpha(N \log N)$ complexity encoding and decoding algorithms. Essentially we get the scaling exponent close to 2, but improve the decoding error probability to sub-exponentially small in N, keeping the construction polynomial-time.

6.1 Preliminaries

6.1.1 Notations

We fix a small positive parameter $\alpha > 0$ from the statement of Theorem 5.1, which corresponds to how close the scaling exponent will be to 1/2. Specifically, we will have the scaling exponent $\mu = 2 + O(\alpha)$. As before, the size of the kernel is denoted by $\ell = 2^s$, where ℓ is large enough in terms of α (specifically, the bounds from the statement of the Theorem 5.1 must hold).

We are going to work with the complete ℓ-ary tree of bit-channels, as described in Section 5.1.2. Let t be the depth of this tree, then there are $N = \ell^t$ bit-channels at the last level, denoted as W_i for $i \in [\ell^t]$ (these notations depend on the depth t of the tree at which we are looking, but it will always be clear from the context). Throughout this section we will denote such a tree of depth t as $\mathcal{T}_t$.

We will work with the same random process W_i of walking down the tree $\mathcal{T}_t$, starting from a root and taking a uniformly random child at each step. As before, define the random processes $\mathsf{Z}_j = Z(\mathsf{W}_j)$ and $\mathsf{H}_j = H(\mathsf{W}_j)$. Further, we will also look at random processes $\mathsf{W}_j^{\text{bin}}, \mathsf{H}_j^{\text{bin}}, \mathsf{Z}_j^{\text{bin}}$, which means that we also do the binning procedure as described in the construction phase in Section 5.4. Note that $\mathsf{W}_j^{\text{bin}}$ are the channels that we actually track during the construction of the code, while W_j are the *true* bit-channels in the tree.

Finally, by $\exp(\bullet)$ we will denote $2^{\bullet}$ in this section, and we denote by $x^+ = \max\{x, 0\}$

the positive part of x.

6.1.2 Plan

First, notice that building the tree $\mathcal{T}_t$ of bit-channels is itself a part of the construction of our polar codes. This includes tracking the binned versions of the bit-channels and picking the kernels using Algorithm A. This part will stay exactly the same as it is described in Section 5.4, with the binning parameter $\mathsf{Q} = N^3$, and the same threshold of ℓ^{-4} in the Algorithm A. The only part of the construction that is going to change is how we pick the set of good indices which we use to transmit information.

We will follow the analysis from [WD19, Appendices B, C], which also appeared before that in [WD18a] under the name of "recyclable recruit-train-retain template". An experienced reader might notice that the proof we provide is a careful application of this technique, slightly modified for our purposes to ensure polynomial-time construction.

The proof consists of three steps, where at each step we improve the decoding error probability while keeping the scaling exponent close to 2 (recall that $s = \log_2 \ell$):

1) $\mathbb{P}\left[\mathsf{Z}_t \le \exp(-2st)\right] \ge I(W) - \ell^{-(1/2-10\alpha)t}$,

2) $\mathbb{P}\left[\mathsf{Z}_t \le \exp\left(-2^{t^{1/3}}\right)\right] \ge I(W) - \ell^{-(1/2-11\alpha)t+\sqrt{t}}$,

3) $\mathbb{P}\left[\mathsf{Z}_t \le \exp\left(-st \cdot \ell^{\alpha \cdot t}\right)\right] \ge I(W) - \ell^{-(1/2-16\alpha)t+2\sqrt{t}}$ for $t = \Omega(\log^6 s)$.

Moreover, the polarization at each step is *poly-time constructible*:

Definition 6.1. *We call the polarization* $\mathbb{P}[\mathsf{Z}_t \le p(t)] \ge R(t)$ *to be* poly-time constructible *if one can find at least* $N \cdot R(t)$ *indexes* $i \in [N]$ *such that* $Z(W_i) \le p(t)$*, where* $N = \ell^t$*, in time polynomial in* N*.*

Notice that poly-time constructible polarization immediately implies polar codes with polynomial-time construction. Therefore, the polarization behavior from Step 3 with $t \ge \frac{1}{\alpha^2}$ will correspond to polar codes with rate $I(W) - N^{-1/2+18\alpha}$ (i.e. codes with scaling exponent $(2 + O(\alpha))$ and sub-exponentially small decoding error probability $N \cdot \exp\left(-st \cdot \ell^{\alpha \cdot t}\right) = \exp(-N^{\alpha})$, with poly$(N)$ construction time.

6.2 Getting $\exp(-N^{\alpha})$ decoding error probability

6.2.1 Step 1

Lemma 6.2. $\mathbb{P}\left[\mathsf{Z}_t \le \exp(-2st)\right] \ge I(W) - \ell^{-(1/2-10\alpha)t}$*. Moreover, this polarization is poly-time constructible.*

Proof. This follows from the analysis of the construction we already have in the previous sections. Fix some t and let $N = \ell^t$. Then the following is implied from Section 5.5.2 if one takes $\mathsf{Q} = N^3$, i.e. $c = 3$:

$$\mathop{\mathbb{P}}_{i \sim [N]} \left[H(W_i^{\text{bin}}) \leq \frac{1}{N^4} \right] \geq I(W) - N^{-(1/2 - 10\alpha)}.$$

Note here that $H(W_i^{\text{bin}})$ are the entropies of the binned bit-channels that we are actually tracking during the construction phase, so they are computable in polynomial time. This means that there is poly(N)-time procedure which returns all the indices i for which $H(W_i^{\text{bin}}) \leq \frac{1}{N^4}$. Then $Z(W_i^{\text{bin}}) < \sqrt{H(W_i^{\text{bin}})} \leq \frac{1}{N^2}$ for these indices, so we have for the random process $\mathsf{Z}_t^{\text{bin}}$:

$$\mathbb{P}\left[\mathsf{Z}_t^{\text{bin}} \leq \ell^{-2t}\right] = \mathbb{P}\left[\mathsf{Z}_t^{\text{bin}} \leq 2^{-2st}\right] = \mathbb{P}\left[\mathsf{Z}_t^{\text{bin}} \leq \exp\left(-2st\right)\right] \geq I(W) - N^{-(1/2-10\alpha)},$$

and moreover, one can find at least $N(I(W) - N^{-(1/2-10\alpha)})$ indexes within $i \in [N]$ for which the inequality $Z(W_i^{\text{bin}}) \leq \exp\left(-2st\right)$ holds in poly(N) time (just by returning the indices for which $H(W_i^{\text{bin}}) \leq \frac{1}{N^4}$). Since it always holds $\mathsf{Z}_t \leq \mathsf{Z}_t^{\text{bin}}$, the statement of the lemma follows. □

6.2.2 Step 2

Next, we are going to strengthen the polarization of the construction, using the result of Lemma 6.2. Specifically, we prove

Lemma 6.3. $\mathbb{P}\left[\mathsf{Z}_n \leq \exp\left(-2^{n^{1/3}}\right)\right] \geq I(W) - \ell^{-(1/2-11\alpha)n+\sqrt{n}}$. *Moreover, this polarization is poly-time constructible.*

Proof. For this lemma, we fix n to be the total depth of the tree (instead of t), and we want to prove the speed of polarization at level n. To do this, we will divide the tree into $\sqrt{n}$ stages[1], each of depth $\sqrt{n}$, and apply the polarization we obtained at Step 1 at each stage. So, we look at m being $\sqrt{n}, 2\sqrt{n}, \dots, n - \sqrt{n}$. Define the following events, starting with $E_0^{(0)} = \emptyset$ (again, closely following [WD19]):

$$\begin{aligned}
A_m &= \left\{\mathsf{Z}_m^{\text{bin}} < \exp(-2sm)\right\} \setminus E_0^{(m-\sqrt{n})} \\
B_m &= A_m \bigcap \left\{\sum_{i=1}^{s\sqrt{n}} g_{sm+i} \leq \beta \cdot s\sqrt{n}\right\} \\
E_m &= A_m \setminus B_m \\
E_0^{(m)} &= E_0^{(m-\sqrt{n})} \cup E_m,
\end{aligned}$$

[1] In this chapter, we always implicitly round square roots to the nearest integer. Such approximation only introduces negligible inaccuracies, which we can ignore

where for now one can think of g_j's as of independent Bern(1/2) random variables for all $j \in [s \cdot n]$. In the following several paragraphs we explain what these events are going to correspond to. First of all, the actual random variable we are tracking here is W_n, and its realizations are ℓ^n bit-channels W_i for $i \in [\ell^n]$ at the last level of the tree. We can then think of events and subsets of bit-channels at level n interchangeably.

Notice that each bit-channel W_i for $i \in [\ell^n]$ corresponds to a unique path in the tree $\mathcal{T}_n$ from the root W (the initial channel) to the leaf W_i on the n^{th} level. We will be interested in the bit-channels on these path, their binned versions, and the parameters of both versions (true and binned) of these channels during the ensuing arguments. We denote this path of true bit-channels as $W_i^{(0)} = W, W_i^{(1)}, \dots, W_i^{(n-1)}, W_i^{(n)} = W_i$. Clearly, this path is just a realization of a random walk $\mathsf{W}_0, \mathsf{W}_1, \dots, \mathsf{W}_n$, when W_n ends up being W_i. In the same way, we will denote by $W_i^{(k),\text{bin}}$, for $k = 0, 1, \dots, n$ the binned version of the bit-channel along this path, and by $H_i^{(k)}$, $H_i^{(k),\text{bin}}$, $Z_i^{(k)}$, and $Z_i^{(k),\text{bin}}$ the corresponding parameters of these channels.

We are going to construct a set of "good" bit-channels $E_0^{(n-\sqrt{n})}$ incrementally, by inspecting the tree from top to bottom. We start with the set $E_0^{(0)} = \emptyset$. Then, at each stage $m = \sqrt{n}$, $2\sqrt{n}, \dots, n - \sqrt{n}$, we find a set E_m of bit-channels which we mark to be "good" at level m. Precisely, the channel W_i, for some $i \in [\ell^n]$, is going to be in E_m, if: a) it was not marked as good before that (i.e. it is not in $E_0^{(m-\sqrt{n})}$); b) the Bhattacharyya parameter $Z_i^{(m),\text{bin}}$ is small, specifically smaller then $\exp(-2sm)$; and c) a certain condition holds for how the branches are chosen in the path for W_i between levels m and $m + \sqrt{n}$ in the tree (more details on this later). Here conditions a) and b) correspond together to the event A_m, while condition c) further defines the event B_m. Then the set $E_0^{(m)}$ will be the set of all bit-channels that we marked to be good up to the level m in the tree, and in the end, by collecting all the bit-channels that we marked as good at the stages $m = \sqrt{n}, 2\sqrt{n}, \dots, n - \sqrt{n}$, we obtain the final set $E_0^{(n-\sqrt{n})}$.

Denote by corresponding lowercase letters the probabilities of the events described before, i.e. $a_m := \mathbb{P}[A_m]$, etc. Finally, let $q_m = I(W) - e_0^{(m)}$, i.e. q_m is the gap between the capacity and the fraction of the channels which we marked as "good" up to level m.

To begin the formal analysis, let us first consider what happens in the case of the event A_m. First, it means that $\mathsf{Z}_m^{\text{bin}} < \exp(-2sm)$. But then we know that we are going to apply Arıkan's kernel $A_2^{\otimes s}$ to this bit-channel at level m, since the threshold for picking Arıkan's kernel in Algorithm A, which we use in the construction phase, is $\ell^{-4} = \exp(-4s)$. This means that, conditioned on A_m, we have $\mathsf{Z}_{m+1} \le \mathsf{Z}_m \cdot 2^s \le \mathsf{Z}_m^{\text{bin}} \cdot 2^s < 2^s \cdot \exp(-2sm)$, where the first inequality follows from that we know how the Bhattacharyya parameter evolves when we use basic Arıkan's transforms. Precisely, using the kernel $A_2^{\otimes s}$ is equivalent to using the basic 2×2 kernel A_2 for s times, and the kernel A_2 in the worst case doubles the Bhattacharyya parameter. Thus s applications of A_2 can increase the Bhattacharyya parameter by at most a factor of 2^s.

Then it is easy to see that even after we apply Arıkan's kernel $A_2^{\otimes s}$ a total of $\sqrt{n}$ times, the Bhattacharyya parameter will still be below the threshold ℓ^{-4}: conditioned on A_m, one has $\mathsf{Z}_{m+\sqrt{n}} \le \mathsf{Z}_m \cdot (2^s)^{\sqrt{n}} < \exp(-2sm) \cdot \exp(s\sqrt{n}) < \exp(-sm) < \ell^{-4}$, as $m \ge \sqrt{n}$. It is easy to verify, using Proposition 5.15 and the relation (5.18) between the entropy and Bhattacharyya parameter of the bit-channel, that the binned parameter $\mathsf{H}^{\text{bin}}_{m+j}$ will also be below ℓ^{-4} for $j = 1, 2, \ldots, \sqrt{n}$. This means that indeed for these $\sqrt{n}$ levels, the Arıkan's kernel was taken in the construction phase. Therefore, we know that only the kernel $A_2^{\otimes s}$ was applied at levels between m and $m + \sqrt{n}$, which can also be viewed as applying the basic 2×2 kernel A_2 for $s\sqrt{n}$ levels in the tree. Further this can be viewed as taking $s\sqrt{n}$ "good" or "bad" branches while going down the tree, where the good branch corresponds to squaring the Bhattachryya parameter, and the bad branch at most doubles it. Denote then by bits $g_{sm+i} \in \{0, 1\}$, for $i \in [s\sqrt{n}]$, the indicators of these branches being good or bad, where $g_{sm+i} = 0$ means the branch is bad, and $g_{sm+i} = 1$ means the branch is good. It is clear then that since we consider the random process of going down the tree choosing the next child randomly, then all g_{sm+i}'s are independent Bern(1/2) random variables. These are exactly the random variables appearing in the definition of B_m.

Notice then that

$$\frac{b_m}{a_m} = \mathbb{P}\left[\sum_{i=1}^{s\sqrt{n}} g_{sm+i} \le \beta \cdot s\sqrt{n}\right] \le 2^{-s\sqrt{n}(1-h_2(\beta))} \le 2^{-\gamma s\sqrt{n}} ,$$

where we can take, for instance, $\beta = 1/20$ and $\gamma = 0.85$. The inequality follows from entropic bound on the sum of binomial coefficients (one could also just use the Chernoff bound).

Recall that we defined $q_m = I(W) - e_0^{(m)}$. We then can write $q_{m-\sqrt{n}} - a_m = I(W) - (e_0^{(m-\sqrt{n})} + a_m)$. But note that by definition, the event $\left\{\mathsf{Z}^{\text{bin}}_m < \exp(-2sm)\right\}$ is a subevent of $A_m \cup E_0^{(m-\sqrt{n})}$, and thus using the bound from Lemma 6.2 (applied for the depth m) we know that

$$(e_0^{(m-\sqrt{n})} + a_m) \ge \mathbb{P}[A_m \cup E_0^{(m-\sqrt{n})}] \ge \mathbb{P}[\mathsf{Z}^{\text{bin}}_m < \exp(-2sm)] \ge I(W) - 2^{(-1/2+10\alpha)sm} .$$

Therefore we conclude

$$(q_{m-\sqrt{n}} - a_m)^+ \le 2^{(-1/2+10\alpha)sm}.$$

We can then derive

$$\begin{aligned} q_m = I(W) - e_0^{(m)} &= I(W) - (e_0^{(m-\sqrt{n})} + e_m) = q_{m-\sqrt{n}} - e_m \\ &= q_{m-\sqrt{n}}\left(1 - \frac{e_m}{a_m}\right) + \frac{e_m}{a_m}(q_{m-\sqrt{n}} - a_m) \\ &\le q^+_{m-\sqrt{n}} \cdot \frac{b_m}{a_m} + (q_{m-\sqrt{n}} - a_m)^+ \\ &\le q^+_{m-\sqrt{n}} \cdot 2^{-\gamma s\sqrt{n}} + 2^{(-1/2+10\alpha)sm}. \end{aligned}$$

Thus we end up we the following recurrence on q^+_m (recall that $\ell = 2^s$):

$$\begin{aligned} &q^+_{\sqrt{n}} \le 1 \\ &q^+_m \le q^+_{m-\sqrt{n}} \cdot \ell^{-\gamma\sqrt{n}} + \ell^{-\frac{m}{2}+10\alpha m}. \end{aligned}$$

Solving this recurrence gives us $q^+_{n-\sqrt{n}} \leq \ell^{-\frac{n}{2}+11\alpha n+\sqrt{n}}$, since $\gamma > 1/2$. Therefore we can conclude

$$e_0^{(n-\sqrt{n})} \geq I(W) - \ell^{-\frac{n}{2}+11\alpha n+\sqrt{n}}. \tag{6.1}$$

Next, let us look at an arbitrary bit-channel (realization of Z_n) for which the event $E_0^{(n-\sqrt{n})}$ happens, and prove that such a bit-channel is indeed "good." Since $E_0^{(n-\sqrt{n})}$ happened, it means that E_m happened at some stage, thus $\mathsf{Z}_m^{\text{bin}} < \exp(-2sm)$ and $\sum_{i=1}^{s\sqrt{n}} g_{sm+i} \geq \beta \cdot s\sqrt{n}$, where g_{sm+i} for $i \in [s\sqrt{n}]$ correspond to taking bad or good branches in the basic 2×2 Arıkan's kernel. Similarly to Claim 5.11, we then can bound

$$\mathsf{Z}_{m+\sqrt{n}} < \left(2^{s\sqrt{n}}\mathsf{Z}_m\right)^{2^{\beta\cdot s\sqrt{n}}} < \left(2^{sm}\exp(-2sm)\right)^{2^{\beta\cdot s\sqrt{n}}} \leq \exp\left(-sm\cdot 2^{\beta\cdot s\sqrt{n}}\right).$$

Then for the remaining $(n-m-\sqrt{n})$ levels of the tree, it is easy to see that the Bhattacharyaa parameter will also not ever be above the threshold of picking Arıkan's kernel in Algorithm A, thus, similarly as before, we can argue that the Bhattacharyya parameter increases by at most a factor of 2^s at each level. Therefore, we derive

$$\mathsf{Z}_n < 2^{s(n-m-\sqrt{n})} Z_{m+\sqrt{n}} \leq 2^{sn}\exp\left(-sm\cdot 2^{\beta\cdot s\sqrt{n}}\right) < \exp\left(-2^{n^{1/3}}\right),$$

where the last inequality follows from $m \geq \sqrt{n}$, $\beta = \frac{1}{20}$, and the condition $s \geq \frac{11}{\alpha}$ from Theorem 5.2 combined with the fact that α is small.

Since we proved that the event $E_0^{(n-\sqrt{n})}$ implies $\mathsf{Z}_n < \exp(-2^{n^{1/3}})$, we conclude, using (6.1):

$$\mathbb{P}[\mathsf{Z}_n < \exp(-2^{n^{1/3}})] \geq e_0^{(n-\sqrt{n})} \geq I(W) - \ell^{-\frac{n}{2}+11\alpha n+\sqrt{n}},$$

which precisely proves the polarization that was stated in the lemma.

The only thing left to prove then is that this polarization is poly-time constructible. To do this, we show that one can find the set $E_0^{(n-\sqrt{n})}$ of bit-channels in poly-time (recall here the equivalence between events and subsets of the bit-channel at the level n of the tree $\mathcal{T}_n$). But one can see that checking if a particular bit-channel W_i, for some $i \in [\ell^n]$, is easy. Indeed, to check if W_i is in $E_0^{(n-\sqrt{n})}$, it suffices to check if W_i is in E_m for any $m = \sqrt{n}, 2\sqrt{n}, \ldots, n-\sqrt{n}$. But this corresponds to looking at a Bhattacharyya parameter $Z_i^{(m),\text{bin}}$ and checking if it is smaller than $\exp(-2sm)$, and, if this is the case, also looking at how many "good" branches (in the basic 2×2 Arıkan's transforms) there were within the next stage ($\sqrt{n}$ levels) in the tree $\mathcal{T}_n$. The latter can be done easily since this information is essentially given by the index i of the bit-channel W_i (by its binary representation, to be precise). The former is actually also straightforward, since $Z_i^{(m),\text{bin}}$ is the parameter of the binned bit-channel $W_i^{(m),\text{bin}}$ that we are *actually tracking* during the construction phase, so we have this channel written down explicitly, and thus calculating its Bhattacharyya parameter is simple. Therefore all this can be done in time polynomial in ℓ^n, and then the whole set $E_0^{(n-\sqrt{n})}$ can be found in polynomial time (we can also say that the event $E_0^{(n-\sqrt{n})}$ is poly-time checkable). This finishes the proof of this lemma. □

For the following step, we will use the event $E_0^{(n-\sqrt{n})}$ as was defined in the proof of the above lemma. For convenience, we denote it as $R_n = E_0^{(n-\sqrt{n})}$, for any integer n. What we will use is that $\mathbb{P}[R_n] \geq I(W) - \ell^{-\frac{n}{2}+11\alpha n+\sqrt{n}}$; if R_n happens, then $\mathsf{Z}_n < \exp\left(-2^{n^{1/3}}\right)$; and that for any bit-channel it can be checked in poly-time if R_n happened, all of which is proven in Lemma 6.3.

6.2.3 Step 3

Here we prove the final polarization step, which implies the results alluded at the beginning of this chapter:

Lemma 6.4. $\mathbb{P}\left[\mathsf{Z}_t \leq \exp\left(-st \cdot \ell^{\alpha \cdot t}\right)\right] \geq I(W) - \ell^{-(1/2-16\alpha)t+2\sqrt{t}}$ *for* $t \geq C \cdot \log^6 s$*, where* C *is a constant. Moreover, this polarization is poly-time constructible.*

Proof. We will again closely follow the approach from [WD19], though we are going to change the indexing notations to avoid any confusion with the previous step. We return to having the total depth of the tree to be t, and we will have $\sqrt{t}$ stages in the tree, each of length $\sqrt{t}$, similarly to the previous step. As before, we will define several events, starting with $C_0^{(0)} = \emptyset$ and $Q_0^{(0)} = \emptyset$. Then, for n being $\sqrt{t}, 2\sqrt{t}, \ldots, t-\sqrt{t}$, we define:

$$\begin{aligned}
C_n &= R_n \setminus C_0^{(n-\sqrt{t})} \\
C_0^{(n)} &= C_0^{(n-\sqrt{t})} \cup C_n \\
D_n &= C_n \bigcap \left\{ \sum_{i=1}^{s(t-n)} g_i \leq \alpha \cdot s \cdot t \right\} \\
Q_n &= C_n \setminus D_n \\
Q_0^{(n)} &= Q_0^{(n-\sqrt{t})} \cup Q_n,
\end{aligned} \tag{6.2}$$

where R_n is defined at the end of previous step, and g_i's can again be thought of as independent Bern(1/2) random variables. The intuition behind what these events correspond to is almost the same as in Step 2, but the bit-channels in D_n have conditions on branching from level n down to the bottom level t (instead of levels between n and $n+\sqrt{t}$). Here, the channels in $Q_0^{(n)}$ are the channels that we mark as "good" up to level n in the tree, and we will be interested in the final set $Q_0^{(t-\sqrt{t})}$ of "good" channels in the end. We again denote by corresponding lowercase letters the probabilities of these events. Define also

$$f_n = I(W) - c_0^{(n)} \quad \text{and} \quad p_n = I(W) - q_0^{(n)} .$$

First, consider event C_n happening. It means that R_n happens, so $\mathsf{Z}_n < \exp\left(-2^{n^{1/3}}\right)$. Then at least for some time, we are going to pick Arıkan's kernel in the construction phase, since the Bhattacharyya parameter is small enough. But assuming that we take Arıkan's kernels all the way down to the bottom of the tree, one can see

$$\mathsf{Z}_t < \ell^{t-n} \cdot \mathsf{Z}_n < \ell^t \cdot \exp\left(-2^{n^{1/3}}\right) \leq 2^{st} \cdot \exp\left(-2^{t^{1/6}}\right) < 2^{-4s} = \ell^{-4}$$

for $t \geq C \log^6 s$, where C is large enough. Again, by using Proposition 5.15 and (5.18) it is easy to show that the entropy of the binned version of the bit-channel will also always be below the threshold ℓ^{-4}. It means that we cannot in $(t-n)$ levels go over the threshold of choosing Arıkan's kernel, thus we indeed take Arıkan's kernel all the way down in the tree for the path for which R_n happens. Thus, similarly to the proof of Lemma 6.3 in Step 2, we can think of it as taking the basic 2×2 Arıkan's kernels $s \cdot (t-n)$ times, starting at level n. Therefore if R_n happens, the branching down from level n can be viewed as taking "good" or "bad" branches in the A_2 kernels, so we again define indicator random variables g_i, for $i \in [s(t-n)]$, to denote these branches. It is clear that these random variables are going to be independent Bern(1/2). These are exactly the random variables g_i, for $i \in [s(t-n)]$, appearing in the definition of D_n.

We have

$$\frac{d_n}{c_n} = \mathbb{P}\left[\sum_{i=1}^{s(t-n)} g_i \leq \alpha s t\right] \leq 2^{-s(t-n)(1-h_2(\delta))}\ , \tag{6.3}$$

where we denote $\delta \coloneqq \min\left\{\frac{\alpha t}{t-n}, 1\right\}$. The inequality again follows from the entropic inequality on the sum of binomial coefficients.

Recall that we denoted $f_n = I(W) - c_0^{(n)}$. The event $C_0^{(n)}$ contains the event R_n, thus $f_n \leq \ell^{-\frac{n}{2}+11\alpha n+\sqrt{n}}$, which follows from the proof of Lemma 6.3. Same inequality holds for f_n^+.

We will obtain a recurrence on $p_n - f_n^+$ as follows:

$$\begin{aligned}
p_n - f_n^+ &= I(W) - q_0^{(n)} - (I(W) - c_0^{(n)})^+ \\
&= p_{n-\sqrt{t}} - q_n - (f_{n-\sqrt{t}} - c_n)^+ \\
&\leq p_{n-\sqrt{t}} - q_n - \frac{q_n}{c_n}(f_{n-\sqrt{t}} - c_n)^+ \\
&\leq p_{n-\sqrt{t}} - q_n - \frac{q_n}{c_n}(f^+_{n-\sqrt{t}} - c_n) \\
&\leq p_{n-\sqrt{t}} - f^+_{n-\sqrt{t}} + \left(1 - \frac{q_n}{c_n}\right) f^+_{n-\sqrt{t}} \\
&= p_{n-\sqrt{t}} - f^+_{n-\sqrt{t}} + \frac{d_n}{c_n} f^+_{n-\sqrt{t}} \\
&\leq p_{n-\sqrt{t}} - f^+_{n-\sqrt{t}} + \ell^{-(1/2-11\alpha)(n-\sqrt{t})+\sqrt{n}} \cdot 2^{-s(t-n)(1-h_2(\delta))},
\end{aligned} \tag{6.4}$$

where recall that $\delta = \min\left\{\frac{\alpha t}{t-n}, 1\right\}$. We want to obtain an upper bound on the additive term in the inequality above. Consider the following two cases:

i) $\delta > \frac{1}{10}$, i.e. $10\alpha t > t - n$, thus $n > (1-10\alpha)t$. Then we give up on the term $2^{-s(t-n)(1-h_2(\delta))}$ completely, and we can write

$$\ell^{-(1/2-11\alpha)(n-\sqrt{t})+\sqrt{n}} \cdot 2^{-s(t-n)(1-h_2(\delta))} \leq \ell^{-(1/2-11\alpha)(1-10\alpha)t+\frac{3}{2}\sqrt{t}} \leq \ell^{-(1/2-16\alpha)t+\frac{3}{2}\sqrt{t}};$$

ii) $\delta \le \frac{1}{10}$, and then $h_2(\delta) < 1/2$. In this case we derive

$$\begin{aligned}\ell^{-(1/2-11\alpha)(n-\sqrt{t})+\sqrt{n}} \cdot 2^{-s(t-n)(1-h_2(\delta))} &\le \ell^{-(1/2-11\alpha)n+\frac{3}{2}\sqrt{t}} \cdot \ell^{-1/2\cdot(t-n)} \\ &= \ell^{-1/2\cdot t+11\alpha n+\frac{3}{2}\sqrt{t}} < \ell^{-1/2\cdot t+11\alpha t+\frac{3}{2}\sqrt{t}}.\end{aligned}$$

Putting the above together, we obtain

$$\begin{aligned}&p_0 - f_0^+ = 0 \\ &p_n - f_n^+ \le p_{n-\sqrt{t}} - f_{n-\sqrt{t}}^+ + \ell^{-(1/2-16\alpha)t+\frac{3}{2}\sqrt{t}}.\end{aligned}$$

Therefore $p_{t-\sqrt{t}} - f_{t-\sqrt{t}}^+ \le \sqrt{t}\cdot\ell^{-(1/2-16\alpha)t+\frac{3}{2}\sqrt{t}}$. Combining this with $f_{t-\sqrt{t}}^+ \le \ell^{-(1/2-11\alpha)(t-\sqrt{t})+\sqrt{t}}$, we obtain $p_{t-\sqrt{t}} \le \ell^{-(1/2-16\alpha)t+2\sqrt{t}}$, and thus

$$\mathbb{P}\left[Q_0^{(t-\sqrt{t})}\right] = q_0^{(t-\sqrt{t})} \ge I(W) - \ell^{-(1/2-16\alpha)t+2\sqrt{t}}. \tag{6.5}$$

Let us now check that the event $Q_0^{(t-\sqrt{t})}$ is actually "good" and allows us achieve the needed polarization. If $Q_0^{(t-\sqrt{t})}$ happens, then Q_n happened for some $n = k\cdot\sqrt{t}$. It means that C_n, and therefore R_n takes place, thus $\mathsf{Z}_n < \exp\left(-2^{n^{1/3}}\right)$. It also means that D_n does not happen, and thus there is at least αst "good" branches taken in the way down the tree, which corresponds to αst squarings of the Bhattacharyya parameter. Therefore

$$\begin{aligned}\mathsf{Z}_t \le \left(\ell^{t-n}\mathsf{Z}_n\right)^{2^{\alpha st}} < \left(2^{st}\exp\left(-2^{n^{1/3}}\right)\right)^{2^{\alpha st}} < \exp\left(-st\cdot 2^{\alpha st}\right) = \exp\left(-st\cdot\ell^{\alpha t}\right) \\ = \frac{1}{N}\exp\left(-N^{\alpha}\right),\end{aligned} \tag{6.6}$$

where the third inequality trivially follows from $n \ge \sqrt{t}$ and $t \ge C\log^6 s$ for large enough C. Combining this with (6.5), we obtain the desired polarization:

$$\mathbb{P}\left[\mathsf{Z}_t < \exp\left(-st\cdot 2^{\alpha st}\right)\right] \ge q_0^{(t-\sqrt{t})} \ge I(W) - \ell^{-(1/2-16\alpha)t+2\sqrt{t}}.$$

It only remains to argue that this polarization is poly-time constructible. But this easily follows from the fact that the event R_n is poly-time checkable, which we proved in Step 2. Indeed, now for any bit-channel W_i, $i \in [\ell^t]$, we need to check if it is in $Q_0^{(t-\sqrt{t})}$. This means that one need to see if Q_n happened for some $n = k\sqrt{t}$. To do this, one checks in poly-time if C_n happened, which reduces to checking R_n (which can be done in poly-time). If R_n happened, then the only thing to check is how many "good" branches the remaining path to W_i has, which is easily (in poly-time) retrievable information from the index i. Therefore, the event $Q_0^{(t-\sqrt{t})}$ is indeed poly-time checkable, which finishes the proof of the lemma. □

6.3 Moderate deviations regime

The proof in Section 6.2 is what we use in [GRY22] to achieve scaling exponent $\mu = 2 + \alpha$ and $P_e \leq \exp(-N^{O(\alpha)})$. Our primary goal there was to get the scaling exponent as close to 2, however, the framework used in [WD18a, WD19] applies more generally to the moderate deviation regime, where one is interested in the tradeoff between μ and φ, where we denote by φ such a value that $P_e \leq \exp(-N^{\varphi})$.

For this section, denote the inverse of the scaling exponent as $\rho = \frac{1}{\mu}$, and we will use these interchangeably. Recall that any achievable pair (φ, ρ) must satisfy $\varphi + 2\rho < 1$ (e.g. [WD19, Proposition 2]) for non-trivial BMS channels (for the more general case of a discrete memoryless channel, this tradeoff holds for any channel with non-zero dispersion). In [MHU16], where the authors proved new upper bounds on the scaling exponents $\mu \leq \mu^* := 4.714$ and $\mu_{BEC} \leq 3.639$, they also proved that all for all pairs of scaling regimes (φ, ρ) which lie under some curve connecting $\left(0, \frac{1}{1+\mu^*}\right)$ and $(1/2, 0)$ (where $\beta = 1/2$ is the best constant for which $P_e \leq \exp(-N^{\beta})$ for *standard* polar codes) are achievable. This was improved in [WD18a] to the curve connecting $\left(0, \frac{1}{\mu^*}\right)$ and $(1/2, 0)$ for any BMS, and furthermore to any pair of points satisfying $\varphi + 2\rho < 1$ for the BEC, when large kernels are used. The similar approach of "interpolating" the tradeoff between φ and μ given the best achievable values for μ^* and φ^* was used in [WD19] for their dynamic random kernels code construction to prove that actually any pair which satisfies $\varphi + \frac{2}{\mu} < 1$ is achievable for arbitrary discrete memoryless channel W. The reader can also refer to [Wan21] for a unified view on this approach and these results.

In this section we show that the approach from [WD18a] for "interpolating" best-known φ and μ parameters is also applicable to our codes, while retaining polynomial-time construction. This shows that the curve that was described there also applies to our codes, except we can take the best achievable scaling exponent of $\mu^* = 1/2$.

Moderate deviations for our construction

We will in fact only need to modify Step 3 of the proof in Section 6.2, following the proof in [WD18a] (or [Wan21, Theorem 2.18]). We have $\alpha > 0$ fixed, denote then $\xi = \frac{1}{2} - 11\alpha$, and take any pair (φ, μ) of positive numbers such that

$$1 - h\left(\frac{\varphi x}{x - y}\right) > \frac{\frac{x}{\mu} - \xi \cdot y}{x - y} \tag{6.7}$$

for all $0 < x < y$, where $h(\cdot)$ is a binary entropy function. This is equivalent to saying that the point $(\varphi, 1/\mu)$ lies strictly to the left of the convex hull of a union of a point $(0, \xi)$ and an epigraph of the function $1 - h(x)$ (and in the first quadrant), see Figure 6.1.

The idea is to give up some of the scaling exponent (notice that we have scaling exponent close to 2 after Step 2) to improve the decoding error probability. In Step 3, change the definition for the event D_n in (6.2) to

$$D_n = C_n \bigcap \left\{ \sum_{i=1}^{s(t-n)} g_i \leq \varphi \cdot s \cdot t \right\},$$

so we only change the threshold of how many "good" branches we want to take. Recall that before that, the threshold was αst, which was exactly what gave us $P_e \leq \exp(-N^{\alpha})$. Then the bound (6.3) turns into similar

$$\frac{d_n}{c_n} = \mathbb{P}\left[\sum_{i=1}^{s(t-n)} g_i \leq \varphi st \right] \leq 2^{-s(t-n)(1-h_2(\delta))},$$

but now for $\delta = \frac{\varphi t}{t-n}$. Our condition (6.7) on where the point (φ, μ) lies implies (for $x = t, y = n$) that $s(t-n)\left(1 - h_2(\delta)\right) > \frac{st}{\mu} - sn \cdot \xi$, and so derive

$$\frac{d_n}{c_n} \leq 2^{\xi sn - \frac{st}{\mu}} = \ell^{\xi n - \frac{t}{\mu}}.$$

Notice that ξ is exactly the scaling exponent we got after Step 2. Therefore in (6.4) we can write

$$\begin{aligned} p_n - f_n^+ &\leq p_{n-\sqrt{t}} - f_{n-\sqrt{t}}^+ + \ell^{-\xi(n-\sqrt{t})+\sqrt{n}} \cdot \ell^{\xi n - \frac{t}{\mu}} \\ &= p_{n-\sqrt{t}} - f_{n-\sqrt{t}}^+ + \ell^{-t/\mu + \xi\sqrt{t} + \sqrt{n}} \leq p_{n-\sqrt{t}} - f_{n-\sqrt{t}}^+ + \ell^{-t/\mu + \frac{3}{2}\sqrt{t}}. \end{aligned}$$

But then we obtain similarly to (6.5) that $p_{n-\sqrt{t}} \leq \ell^{-t/\mu + 2\sqrt{t}}$, and further

$$\mathbb{P}\left[Q_0^{(t-\sqrt{t})} \right] = q_0^{(t-\sqrt{t})} \geq I(W) - \ell^{-t/\mu + 2\sqrt{t}}.$$

Finally, in (6.6) we conclude (just by substituting α with μ) that if the event $Q_0^{(t-\sqrt{t})}$ happens, then $\mathsf{Z}_t \leq \left(\ell^{t-n}\mathsf{Z}_n\right)^{2^{\varphi st}} \leq 1/N \exp(-N^{\varphi})$, again with the only condition being $t \geq C \log^6 s$. This then implies the codes with scaling exponent $1/\mu$ and $P_e \leq 1/N \exp(-N^{\varphi})$ for any such pair of μ and φ for which the condition (6.7) satisfied. It only remains to point out that the polarization we obtained is also poly-time constructible, as we are still only tracking the number of good branches on Step 3, to check when the event D_n holds.

Denote then by $\mathcal{D}$ an open region of pairs $(\varphi, 1/\mu)$ for which (6.7) is satisfied with $\xi = \frac{1}{2}$ (as we can set α arbitrarily small). So $\mathcal{D}$ is a region in the first quadrant which lies to the left of a convex hull of $\left\{ \left(0, \frac{1}{2}\right) \bigcup \text{epi}(1 - h(x)) \right\}$, see Figure 6.1. Then [WD18a] for our code construction implies the following

Corollary 6.5. *For any BMS channel and any $(\varphi, 1/\mu) \in \mathcal{D}$, there is a large enough ℓ for which there exist codes with rate $R \geq I(W) - N^{-1/\mu}$, decoding error probability $P_e \leq$ $\exp(-N^{\alpha})$, encoding and decoding complexities $O_{\alpha}(N \log N)$, which we can construct in polynomial time.*

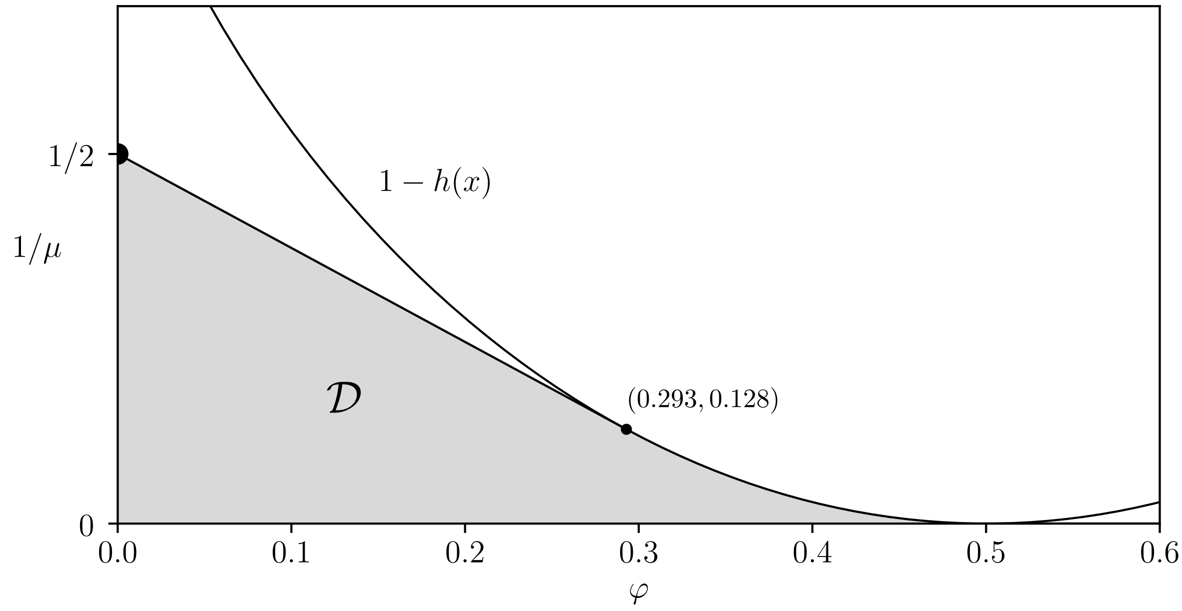


Figure 6.1: Region $\mathcal{D}$ of attainable pairs $(\varphi, 1/\mu)$ with polynomial-time construction. The line coming from $(0, 1/2)$ is a tangent to the plot of $1-h(x)$, $h(x)$ being the binary entropy function.

This corollary essentially shows that the arguments from [WD18a] work in exactly the same way if the optimal scaling exponent $\mu^* = 2$ is obtained using mixed-kernel construction but when Arıkan's kernels are used for suction at the ends regime, which allows polynomial-time construction. Notice that the fact that our construction only reaches the point $(1/2, 0)$ also stems from the fact that we are using Arıkan's kernels. This is because the scaling of decoding error probability is actually dictated by our speed of convergence at suction at the end regimes, where we capped ourselves with the performance of the original polar codes, by our choice of kernels there.

6.4 Concluding remarks

We conclude with a discussion about how the code constructions in [GRY22] and [WD21] might be slightly modified to improve provable performance. However, no rigorous claims are provided here.

Recall the two reasons why Arıkan's kernels were chosen for the suction at the ends regime in our construction: a) to simplify the proof of (5.10) when $g_\alpha(H(W))$ is tiny (see Remark 5.6; and b) it allows for polynomial-time construction and very small P_e at the same time. However, neither of these are actually specific to Arıkan's kernel.

The evolution of the potential function is, in general, much faster for the suction at the ends regime, for any polarizing matrix. This is inherent to the fact that the Bhattacharyya parameter Z is raised to some power for the bit-channels that become better, and on the other hand it is at worst multiplied by constant for the worse channel (recall $Z(W^+) \leq Z(W)^2$, $Z(W^-) \leq 2Z(W)$ from (5.19) and (5.20)). Although we did not discuss it for our construction and just proved all the necessary claims for the A_2 kernel, such evolution happens for any $\ell \times \ell$ polarizing (mixing) matrix. The powering that is happening to Z is exactly what defines the kernel error exponent $E(K)$, which shows in $P_e \leq \exp(-N^{E(K)})$. Without formalizing it here, it is possible to show that the suction at the end regime would work if some other fixed polarizing matrix was taken instead of A_2, and used recursively a sufficient amount of times. In particular, a fixed sub-kernel with exponent $E(K)$ close to 1 would be of interest ($E(A_2) = 1/2$).

As for b), if we use another fixed kernel K for suction at the ends regime, then instead of only counting "good" and "bad" branches to track the evolution of the true underlying parameters, we would just need to look at the exact indices of bit-channels that were taken during the recursive construction, but these are also easily available to us. In other words, the construction procedure can also be made polynomial-time if another fixed kernel is used. But then, by using a kernel with a larger error exponent $E(K)$, this would directly correspond to a larger region of attainable parameters $(\varphi, 1/\mu)$.

Instead of making the last statement more precise, we claim that similar arguments will work in the general case of [WD19, Wan21], where arbitrary discrete memoryless channels are considered. By fixing the kernel K with a good exponent $E(K)$ and some additional properties for the suction at the ends regime in their construction, this eliminates the need to track the true bit-channels in order to find a good local kernel. The existence of such a good fixed kernel K for which $E(K)$ is close to 1 can be derived from their arguments. The only remaining piece would be to apply degraded binning to the construction procedure, to make it polynomial-time. While this does introduce additional technical difficulties which need to be dealt with, we believe these are solvable by carefully bounding the parameters of the true and approximated channels, as we have done in our analysis. If done rigorously, this would imply that any pair of parameters (φ, μ) for which $\varphi + 2/\mu < 1$ is achievable with codes that can be constructed in poly(N) time for an arbitrary discrete memoryless channel.

Bibliography

[Arı09] Erdal Arıkan. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Transactions on Information Theory*, pages 3051–3073, July 2009. 1, 3, 9, 49, 50, 52, 53, 54, 55, 70, 74, 75, 83, 88

[Arı10] Erdal Arıkan. Source polarization. *2010 IEEE International Symposium on Information Theory*, pages 899–903, 2010. 75

[AT09] Erdal Arıkan and Emre Telatar. On the rate of channel polarization. In *Proceedings of 2009 IEEE International Symposium on Information Theory*, pages 1493–1495, 2009. 3, 5, 54, 55, 76

[AW10] Yücel Altuğ and Aaron B. Wagner. Moderate deviation analysis of channel coding: Discrete memoryless case. In *2010 IEEE International Symposium on Information Theory*, pages 265–269, 2010. 4

[AW14] Yücel Altuğ and Aaron B. Wagner. Moderate deviations in channel coding. *IEEE Transactions on Information Theory*, 60(8):4417–4426, 2014. 4

[AY20] Emmanuel Abbe and Min Ye. Reed-muller codes polarize. *IEEE Transactions on Information Theory*, 66(12):7311–7332, 2020. 56

[BBGL17] M. Benammar, V. Bioglio, F. Gabry, and I. Land. Multi-kernel polar codes: Proof of polarization and error exponents. In *2017 IEEE Information Theory Workshop (ITW)*, pages 101–105. IEEE, 2017. 5, 63, 65

[BF02] Alexander Barg and G. David Forney. Random codes: minimum distances and error exponents. *IEEE Transactions on Information Theory*, 48(9):2568–2573, Sep. 2002. 65, 72

[BGN$^+$18] Jaroslaw Blasiok, Venkatesan Guruswami, Preetum Nakkiran, Atri Rudra, and Madhu Sudan. General strong polarization. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 485–492. ACM, 2018. 57, 61, 64, 91

[BGS18] Jaroslaw Blasiok, Venkatesan Guruswami, and Madhu Sudan. Polar codes with exponentially small error at finite block length. In *APPROX-RANDOM*, 2018. 57

[BGT93] 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*, volume 2, pages 1064–1070 vol.2, 1993. 56

[DR96] Devdatt Dubhashi and Desh Ranjan. Balls and bins: A study in negative dependence. *BRICS Report Series*, 3(25), Jan. 1996. 32

[Dra11] S. Dragomir. A refinement and a divided difference reverse of jensen's inequality with applications. *Revista Colombiana de Matemáticas*, 50, 2011. 39

[DZF16] Y. Domb, R. Zamir, and M. Feder. The random coding bound is tight for the average linear code or lattice. *IEEE Transactions on Information Theory*, 62(1):121–130, Jan 2016. 72

[FHMV17] Arman Fazeli, S. Hamed Hassani, Marco Mondelli, and Alexander Vardy. Binary Linear Codes with Optimal Scaling and Quasi-Linear Complexity. *ArXiv e-prints*, November 2017. 4, 57, 59, 61, 62, 64, 69

[FHMV18] Arman Fazeli, Hamed Hassani, Marco Mondelli, and Alexander Vardy. Binary linear codes with optimal scaling: Polar codes with large kernels. In *2018 IEEE Information Theory Workshop (ITW)*, pages 1–5, 2018. 57, 58

[For65] G. David Forney. *Concatenated codes*. Cambridge, MA: MIT Press, 1965. 56

[For05] G. David Forney. On exponential error bounds for random codes on the BSC. *Lecture notes*, 2005. Available at http://web.mit.edu/6.441/spring05/reading/Forney_ExpEBBSC.pdf. 72, 73

[FV14] Arman Fazeli and Alexander Vardy. On the scaling exponent of binary polarization kernels. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 797–804, Sep. 2014. 57

[Gal65] R. Gallager. A simple derivation of the coding theorem and some applications. *IEEE Transactions on Information Theory*, 11(1):3–18, January 1965. 56, 65, 72

[Gal68] Robert G Gallager. *Information theory and reliable communication*, volume 2. Springer, 1968. 70

[GB14] Dina Goldin and David Burshtein. Improved bounds on the finite length scaling of polar codes. *IEEE Trans. Information Theory*, 60(11):6966–6978, 2014. 3, 57

[GBLB17] F. Gabry, V. Bioglio, I. Land, and J. Belfiore. Multi-kernel construction of polar codes. In *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 761–765. IEEE, 2017. 5, 63, 65

[GRY20] Venkatesan Guruswami, Andrii Riazanov, and Min Ye. Arikan meets shannon: Polar codes with near-optimal convergence to channel capacity. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2020, page 552–564, New York, NY, USA, 2020. Association for Computing Machinery. 5, 6, 58, 63

[GRY22] Venkatesan Guruswami, Andrii Riazanov, and Min Ye. Arıkan meets shannon: Polar codes with near-optimal convergence to channel capacity. *IEEE Transactions on Information Theory*, pages 1–1, 2022. 5, 6, 58, 107, 109

[GX13] Venkatesan Guruswami and Patrick Xia. Polar codes: Speed of polarization and polynomial gap to capacity. In *Proceedings of the 54th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 310–319, October 2013. 57, 98

[GX15] Venkatesan Guruswami and Patrick Xia. Polar codes: Speed of polarization and polynomial gap to capacity. *IEEE Trans. Information Theory*, 61(1):3–16, 2015. Preliminary version in Proc. of FOCS 2013. 3, 5, 57, 67, 80

[HAU14] S. H. Hassani, K. Alishahi, and R. L. Urbanke. Finite-length scaling for polar codes. *IEEE Transactions on Information Theory*, 60(10):5875–5898, Oct 2014. 3, 4, 57, 75

[HMTU13] S. Hamed Hassani, Ryuhei Mori, Toshiyuki Tanaka, and Rüdiger L. Urbanke. Rate-dependent analysis of the asymptotic behavior of channel polarization. *IEEE Transactions on Information Theory*, 59(4):2267–2276, 2013. 5

[Hoe63] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. 38

[iFLM11] A. G. i. Fàbregas, I. Land, and A. Martinez. Extremes of random coding error exponents. In *2011 IEEE International Symposium on Information Theory Proceedings*, pages 2896–2898, July 2011. 72

[JDP83] Kumar Joag-Dev and Frank Proschan. Negative association of random variables with applications. *The Annals of Statistics*, 11(1):286–295, 1983. 32

[KKM$^+$17] Shrinivas Kudekar, Santhosh Kumar, Marco Mondelli, Henry D. Pfister, Eren

Şaşoğlu, and Rüdiger L. Urbanke. Reed–muller codes achieve capacity on erasure channels. *IEEE Transactions on Information Theory*, 63(7):4298–4316, 2017. 56

[KMTU10] Satish Babu Korada, Andrea Montanari, Emre Telatar, and Rüdiger L. Urbanke. An empirical scaling law for polar codes. In *Proceedings of 2010 IEEE International Symposium on Information Theory*, pages 884–888, 2010. 57, 61

[Kor09] Satish Babu Korada. *Polar Codes for Channel and Source Coding*. PhD thesis, École Polytechnique Fédérale De Lausanne, 2009. 9, 23, 74, 75

[KRU13] Shrinivas Kudekar, Tom Richardson, and Rüdiger L. Urbanke. Spatially coupled ensembles universally achieve capacity under belief propagation. *IEEE Transactions on Information Theory*, 59(12):7761–7813, 2013. 56

[KSU10] Satish Babu Korada, Eren Sasoglu, and Rüdiger L. Urbanke. Polar codes: Characterization of exponent, bounds, and constructions. *IEEE Transactions on Information Theory*, 56(12):6253–6264, 2010. 3, 5, 54, 55, 56

[KU10] Satish Babu Korada and Rüdiger L. Urbanke. Polar codes are optimal for lossy source coding. *IEEE Transactions on Information Theory*, 56(4):1751–1768, 2010. 68, 70

[LH06] Ingmar Land and Johannes Huber. Information combining. *Foundations and Trends in Communications and Information Theory*, 3(3):227–330, 2006. 23

[LMSS98] Michael Luby, Michael Mitzenmacher, Amin Shokrollahi, and Daniel A. Spielman. Analysis of low density codes and improved designs using irregular graphs. In *STOC '98*, 1998. 56

[LMSS01] M.G. Luby, M. Mitzenmacher, M.A. Shokrollahi, and D.A. Spielman. Efficient erasure correcting codes. *IEEE Transactions on Information Theory*, 47(2):569–584, 2001. 56

[Mac99] D.J.C. MacKay. Good error-correcting codes based on very sparse matrices. *IEEE Transactions on Information Theory*, 45(2):399–431, 1999. 56

[MHU16] Marco Mondelli, S. Hamed Hassani, and Rüdiger L. Urbanke. Unified scaling of polar codes: Error exponent, scaling exponent, moderate deviations, and error floors. *IEEE Trans. Information Theory*, 62(12):6698–6712, 2016. 3, 5, 57, 58, 61, 107

[MS77] Florence Jessie MacWilliams and Neil James Alexander Sloane. *The theory of error-correcting codes*, volume 16. Elsevier, 1977. 20

[MT12] Vera Miloslavskaya and Peter Trifonov. Design of binary polar codes with arbitrary kernel. *2012 IEEE Information Theory Workshop*, pages 119–123, 2012. 57

[MT14] Ryuhei Mori and Toshiyuki Tanaka. Source and channel polarization over finite fields and reed-solomon matrices. *IEEE Trans. Information Theory*, 60(5):2720–2736, 2014. 5, 57, 77

[PPV10] Yury Polyanskiy, H Vincent Poor, and Sergio Verdú. Channel coding rate in the finite blocklength regime. *IEEE Transactions on Information Theory*, 56(5):2307, 2010. 2, 4

[PSL15] Noam Presman, Ofer Shapira, and Simon Litsyn. Mixed-kernels constructions of polar codes. *IEEE Journal on Selected Areas in Communications*, 34(2):239–253, 2015. 5, 63, 65

[PU16] Henry D. Pfister and Rüdiger L. Urbanke. Near-optimal finite-length scaling for polar codes over large alphabets. In *IEEE International Symposium on Information Theory, ISIT*, pages 215–219, 2016. 57

[RU08] Thomas Richardson and Rüdiger Urbanke. *Modern Coding Theory*. Cambridge University Press, 2008. 75

[Sha48] Claude Elwood Shannon. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948. 1, 2, 13, 56

[Spi96] D.A. Spielman. Linear-time encodable and decodable error-correcting codes. *IEEE Transactions on Information Theory*, 42(6):1723–1731, 1996. 56

[STA09] E. Sasoglu, E. Telatar, and E. Arikan. Polarization for arbitrary discrete memoryless channels. In *2009 IEEE Information Theory Workshop*, pages 144–148, Oct 2009. 55

[Str62] Volker Strassen. Asymptotische Abschatzungen in Shannon's Informationstheories. In *Trans. 3rd Prague Conf. Info. Theory*, pages 689–723, 1962. 2

[Str09] Volker Strassen. Asymptotic estimates in Shannon's information theory. In *Proc. Trans. 3rd Prague Conf. Inf. Theory*, pages 689–723, 2009. 2

[Top01] Flemming Topsøe. Bounds for entropy and divergence for distributions over a two-element set. *JIPAM. Journal of Inequalities in Pure & Applied Mathematics [electronic only]*, 2(2):Paper No. 25, 13 p.–Paper No. 25, 13 p., 2001. 22, 31

[TV13] Ido Tal and Alexander Vardy. How to construct polar codes. *IEEE Transactions on Information Theory*, 59(10):6562–6582, Oct 2013. 11, 24, 42, 65, 67, 98

[Wan21] Hsin-Po Wang. Complexity and second moment of the mathematical theory of communication. *arXiv preprint arXiv:2107.06420*, 2021. 58, 107, 110

[WD18a] Hsin-Po Wang and Iwan Duursma. Polar-like codes and asymptotic tradeoff among block length, code rate, and error probability. *arXiv:1812.08112*, 2018. 4, 5, 6, 57, 58, 65, 97, 99, 107, 108, 109

[WD18b] Hsin-Po Wang and Iwan M. Duursma. Polar code moderate deviation: Recovering the scaling exponent. *ArXiv*, abs/1806.02405, 2018. 5, 57, 58

[WD19] Hsin-Po Wang and Iwan M. Duursma. Polar codes' simplicity, random codes' durability. *ArXiv*, abs/1912.08995, 2019. 5, 6, 58, 63, 77, 97, 98, 99, 100, 104, 107, 110

[WD21] Hsin-Po Wang and Iwan M. Duursma. Polar codes' simplicity, random codes' durability. *IEEE Transactions on Information Theory*, 67(3):1478–1508, 2021. 5, 58, 109

[Wol57] Jacob Wolfowitz. The coding of messages subject to chance errors. *Illinois J. Math.*, 1:591–606, 1957. 2, 5, 13

[YB15] Min Ye and Alexander Barg. Polar codes using dynamic kernels. In *2015 IEEE International Symposium on Information Theory (ISIT)*, pages 231–235. IEEE, 2015. 5, 63, 65, 68

[YFV19] Hanwen Yao, Arman Fazeli, and Alexander Vardy. Explicit polar codes with small scaling exponent. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pages 1757–1761, July 2019. 57