Efficient Polar and LDPC Coding for Asymmetric Channels and Sources

（非対称な通信路と情報源の効率的な polar 符号化および LDPC 符号化）

Junya Honda　本多 淳也

Abstract

Channel coding and source coding are the most fundamental problems in information theory. Shannon formulated these problems and derived their theoretical limits on the efficiency of the codes. Since the original codes proposed by him require exponential complexity to achieve the limits, the main topic of information theory has been to construct a practical coding scheme approaching the limits.

Among the above coding problems, we consider channel coding and lossy source coding. Previous works on these problems mainly focused their attention on symmetric settings, that is, channel coding for symmetric channels or lossy source coding for uniform sources and symmetric distortion measures, which make the code construction and analysis much easier. However, real-world settings are not always symmetric and therefore practical coding schemes for the asymmetric settings are also important as well.

In coding problems for asymmetric settings, the optimal symbol distribution of codewords is not uniform in general. In most literatures for these settings, coding schemes are constructed based on Gallager's nonlinear mapping, which reduces generation of nonuniform codewords into that of uniform codewords with an extended alphabet. However, coding schemes constructed by Gallager's method are often inefficient in practice whereas they can achieve the the Shannon bound theoretically. It is mainly because the complexity of a code increases rapidly with the alphabet size in most coding schemes whereas infinitely increasing alphabet size is required to achieve the Shannon bound by Gallager's method in general.

To achieve the Shannon bound with realistic complexity, we investigate coding schemes which do not require any extended alphabet to generate nonuniform codewords in this thesis. The key to the nonuniform codewords is in lossless compression, which transforms a source sequence into a uniform sequence. This nature of the lossless compression implies that we can obtain a nonuniform codeword from a uniform sequence by the inverse operation of lossless compression.

First we apply the idea of using lossless compression to *polar codes*. Polar codes are recently proposed for symmetric channels by Arikan and attracting much attention because of their achievability of the channel capacity with polynomial complexity. In the polar codes, a sequence of i.i.d. channel inputs is transformed by an invertible linear operation into another sequence. The

input sequence polarizes by this transformation, that is, each variable of the transformed sequence becomes almost deterministic or almost random given the preceding variables. Then, reliable transmission can be realized by assigning a message to the almost deterministic variables of the transformed sequence.

The idea of using lossless compression for asymmetric channels can be easily applied to polar codes. It is because the probability distributions of the channel inputs also polarize into almost deterministic or almost random ones as well as the reliabilities of the variables given channel outputs after the transformation for polar codes. The inverse operation of the lossless compression can be easily realized by simply assigning a uniform message to the almost random variables and determining the other variables to be the most likely ones. Also for lossy source coding with nonuniform sources and/or asymmetric distortion measures, optimal polar codes can be constructed easily by exchanging the encoding and the decoding of those for channel coding.

Next we consider *low density parity check (LDPC) codes* for asymmetric settings. An LDPC code is a linear code using a sparse matrix, and it is one of the most promising codes for its good performance with practical decoding algorithms although the theoretical analysis is difficult compared to polar codes. In the framework of LDPC codes, Miyake and Muramatsu proposed coding schemes based on fixed length lossless compression for the asymmetric settings. They proved that their scheme can achieve the Shannon bound under the maximum-likelihood (ML) decoding. However, the ML decoding is computationally intractable and it is not shown that their scheme can attain good performance under polynomial-time decoding algorithms. It is because most known practical algorithms such as belief propagation often fail in the decoding of fixed length lossless compression.

In this thesis, we propose a new variable length coding scheme using LDPC codes. We first investigate a fixed-to-variable length lossy coding scheme for nonuniform sources and/or asymmetric distortion measures. We use arithmetic coding for the lossless compressor and show that our scheme can achieve the rate-distortion function. Note that the exact computation of the probability assigned for the arithmetic code is also intractable practically. However, in our scheme we can achieve near optimal performance by approximating the assigned probability with moderate accuracy. This fact contrasts with the fixed length coding where any codeword other than the ML codeword leads to a large distortion.

We next investigate variable length channel coding scheme using LDPC

codes. Generally it is known that a good channel code can be constructed by exchanging the encoder with the decoder of a good lossy code. Then it is natural to consider a variable-to-fixed length channel code obtained from the lossy code described above. However, this technique cannot be used in this case because the arithmetic code does not work as a code when the encoder and the decoder are exchanged. Then we propose a fixed-to-variable length channel coding scheme using a homophonic code. Homophonic codes are the codes used for transforming a random sequence reversibly to another random sequence with a different distribution. We prove that the channel capacity can be achieved by generating nonuniform LDPC codewords using a good homophonic code.

Finally we consider practical algorithms for LDPC codes. In the proposed schemes for the asymmetric settings, two types of computation are required for the encoding and the decoding. The first type of computation is the search of the ML codeword, and we propose two algorithms for this problem based on a linear programming relaxation technique and a reinforced belief propagation combined with a matroid optimization. We show by simulation that they attain good performance for channel coding and lossy source coding, respectively. The second type of computation is sequential computation of marginal probabilities of symbols over nonuniform LDPC codewords for an arithmetic code or a homophonic code. We propose an algorithm using a belief propagation which makes the complexity of the arithmetic coding or the homophonic coding nearly linear time.

Keywords LDPC codes, polar codes, channel coding, lossy source coding, arithmetic coding, homophonic coding, belief propagation, linear programming decoding, nonbinary codes

Acknowledgment

It is my great pleasure to thank my supervisor, Prof. Hirosuke Yamamoto, for his guidance during many years. From when I was assigned to his laboratory for the bachelor's thesis, he always showed me his insight of information theory. His knowledge and experience were precious resources for a new idea and his advice always led me to my research success. This thesis would not have been possible without his helpful suggestions and continuous effort to improve my presentation and writing.

I am grateful to Prof. Akimichi Takemura for his suggestions on statistics and probability theory. He continued to guide me after I completed my master's course in his laboratory and I was able to get to know many researchers in his research area.

I wish to express my gratitude to Prof. Toshiyuki Tanaka and Mr. Ryuhei Mori for valuable discussions on polar codes. Especially, Mr. Ryuhei Mori introduced me polar codes and Chapter II of this thesis owes much to his comments.

I am deeply indebted to Dr. Taiji Suzuki and Dr. Yusuke Kobayashi for their support through three years of my doctoral studies. Dr. Taiji Suzuki spent much time giving me advice on everything from daily life to technical topics. Dr. Yusuke Koboyashi patiently answered my questions on mathematical programming both in theory and implementation.

I also thank Prof. Noboru Kunihiro, Prof. Masato Okada, Prof. Shigeo Takahashi and Prof. Hiroki Koga for reading my thesis carefully.

I thank all the students and alumni of Yamamoto-Kunihiro laboratory, who have contributed to the wonderful time at the laboratory. I enjoyed with them not only seminars and lunches in Kashiwa campus but also special activities outside the campus such as cycling, traveling and big eating challenge. Special thanks go to Dr. Yu Nureki who demonstrated the fun of research life when I did not decide to start working or go on to the doctoral course.

I acknowledge that this work is supported by JSPS Research Fellowships

for Young Scientists.

Finally, I would like to thank my friends and family for encouraging and supporting me throughout.

December 2012

Junya Honda

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

Channel coding and source coding are the ones of the most fundamental tools for digital data processing. Shannon formulated these problems and derived theoretical bounds on efficiency for these problems in his seminal work [1]. He also constructed coding schemes which actually achieve these bounds. However, the time and space complexity of his schemes was exponential in the blocklength of the code and it has been a central topic of information theory to find a coding scheme which achieves the bounds with polynomial (or possibly linear) complexity.

1.1.1 Channel Coding

Channel coding is a framework of information transmission via a noisy channel. The noise of the channel can be caused by signal decay through the air or interference with other signals when you communicate with someone by a cell phone. It may also be a scratch on a CD or a DVD, or an error of its reader when you enjoy music or a movie recorded in the disc. The fundamental idea for reliable communication is to add redundancy to the data. The data strengthened by redundancy is called a *codeword*. A *channel code* is a manner of transforming an original data, or a message, to a codeword and recovering the message from the received codeword disturbed by noise. The *coding rate* is the ratio of a message length to the codeword length. Shannon derived the maximum coding rate, called *channel capacity* (or *capacity* for short), such that arbitrarily small decoding error probability is achievable.

The simplest example of channel codes may be a *repetition code*. For one-bit message 0 or 1 and parameter k, the codeword of the message is the k

repetitions of the message. Through the channel, each bit sometimes flips to the other bit, that is, 0 flips to 1 and 1 to 0. The receiver estimates the sent message from the received sequence by the following way. The receiver estimates that 1 will be the original message if the sequence contains more 1s than 0s and 0 otherwise. Under this decoding rule, the repetition code can correct less than $k/2$ bit flips through the channel. However, unfortunately, this code is constructed too naively and the capability of the error correction is far from practical.

Early researches on practical channel codes were based on an algebraic approach so that the *minimum distance*, or the smallest distance between two codewords, is maximized. This approach was initiated with Hamming [2] and much progress has been made such as invention of Reed-Muller codes [3][4], BCH codes [5][6] and Reed-Solomon codes [7]. These codes are still used widely in CDs, DVDs, digital television transmission, and so on. However, in those algebraic codes, the minimum distance or the coding rate approaches zero as the block length increases. Although many practical coding techniques had been proposed such as code concatenation [8] and convolutional codes [9] with Viterbi algorithm [10] or BCJR algorithm [11], it had been unsolved long time whether there exists an efficiently implementable code with a rate near the capacity and a decoding error probability close to zero.

Invention of Turbo codes [12] and rediscovery of low density parity check (LDPC) codes [13][14][15] were the significant progress in this line of research. Those codes are experimentally shown to achieve rates close to the capacity by using iterative decoding algorithms which can be implemented efficiently. Especially, LDPC codes with an iterative decoding algorithm called belief propagation (BP) have excellent performance and applied to many communication systems such as satellite communication. However, the analysis of the BP is somewhat difficult although density evolution [16] is known as a powerful tool for the analysis. Linear programming (LP) decoding [17] was proposed later as an alternative decoding technique to the BP decoding. In this technique the maximum likelihood (ML) decoding is relaxed to a linear programming problem which can be solved efficiently by e.g. the simplex method. The LP decoding performs comparably to the BP and has an advantage in simplicity of the theoretical analysis. However, it still remains as an unsolved problem to analyze the achievability of the capacity of LDPC codes under polynomial-time decoding algorithms.

Recently Arikan proposed polar codes [18]. Polar codes are the first family of coding schemes which achieves the channel capacity with polynomial

complexity. Currently the empirical performance is not better than LDPC codes because the convergence speed to the capacity of polar codes is not fast. Nevertheless, this coding scheme is researched intensively because of its potential.

1.1.2 Source Coding

Source coding is also called data compression, which transforms a data sequence into another sequence with a shorter length. Recently there have been many advances in communication systems and storage devices, but, source coding is still an important tool because the volumes of data we deal with are also increasing according to the advances.

Source coding is classified into two types, lossless coding and lossy coding. Lossless coding literally refers to data compression such that the original data can be recovered without any loss of information. For example, zip format and rar format are widely used for lossless compression of e.g. document data. On the other hand, lossy coding allows some loss of information to compress the data. MP3 format for sound data and JPEG format for image data can be regarded as this type of compression.

There had been significant progress in lossless source coding. Huffman coding [19] was proved to be optimal for memoryless sources for the case that the symbol distribution is known and arithmetic coding [20][21][22] also achieves the optimal rate asymptotically with reasonable complexity. Even for general stationary ergodic sources, Lempel-Ziv algorithms [23][24] achieve the asymptotically optimal rate without knowledge on the source statistics. These optimal lossless codes can be implemented easily and are widely used.

However, in turn, lossy source coding is a little difficult and there is some gap between the Shannon bound and the achievable rate by practical algorithms, although this problem has been researched for a long time (see, e.g., [25]). Since Shannon implied in [26] that a good channel code can be used as a good lossy code, lossy coding algorithms have mainly been considered based on known efficient channel codes. Matsunaga-Yamamoto [27] showed that LDPC codes with the optimal encoding algorithm can achieve the Shannon bound, although the algorithm is NP-complete [28]. Korada-Urbanke [29] showed that the bound can be achieved with polynomial complexity by polar codes.

Although LDPC codes with the BP decoding perform empirically better than polar codes in channel coding, the BP for LDPC codes does not work

well in lossy source coding. To attain good performance in the framework of lossy coding based on sparse matrices, low density generator matrix (LDGM) codes have been researched extensively [30][31] as a dual of LDPC codes. The LDGM codes are defined as images of sparse linear operators whereas LDPC codes are defined as kernels of sparse linear operators. It is shown by simulation in [32] that LDGM codes achieve performance very close to the Shannon bound by using *reinforced belief propagation* (RBP), which is a generalization of the BP inspired by statistical physics. The RBP was also applied to lossy coding by LDPC codes and confirmed that it works successfully combined with nonbinary LDPC codes [33][34].

1.2 Coding Problems for Asymmetric Models

In channel coding and lossy source coding, the main targets of research have been symmetric settings, that is, channel coding for symmetric channels and lossy source coding for uniform sources and symmetric distortion measures. In the symmetric settings the optimal symbol distribution of codewords is uniform distribution, and it can be realized by linear codes such as LDPC codes and polar codes. On the other hand, in asymmetric settings, the optimal symbol distribution is not always uniform although each symbol is distributed uniformly when we use linear codes.

The most common technique to realize nonuniform distributions of codewords is Gallager's nonlinear mapping [35, p. 208], or Gallager's method for short. In most literatures, the optimalities of their proposed codes were extended to asymmetric settings based on this method. For example, the asymptotic optimality of LDPC codes for asymmetric channels is proved in [36] based on this method. This method is also used in a family of LDGM codes [37][38] and polar codes [29] for lossy source coding with nonuniform sources and/or asymmetric distortion measures.

One exception that the optimality for asymmetric settings is proved without Gallager's method can be found in the coding scheme proposed by Miyake-Muramatsu for channel coding [39] and lossy source coding [40]. In their scheme, a nonuniform symbol distribution of codewords is realized by a lossless code. Their idea can be described as follows: since lossless compression can be regarded as an invertible transformation of a nonuniform sequence into a uniform sequence, a nonuniform codeword can be generated by the inverse transformation of lossless compression, i.e., the decoding of lossless compression. Based on this idea they proposed a coding

scheme using a lossless code. In their scheme, a fixed length lossless code using LDPC code is adopted. Generally, a fixed length code sometimes fails in the decoding in spite of a "lossless" code. Although they showed that the decoding error probability can be arbitrarily small under the ML decoding, the ML decoding is computationally intractable and known polynomial-time algorithms such as the BP often fail in decoding of the lossless code. Currently it has not been shown that their scheme can attain good performance under practical algorithms.

1.3 Contribution of this Thesis

In this thesis we consider channel coding with asymmetric channels and lossy source coding with nonuniform sources and/or asymmetric distortion measures. We propose efficient coding schemes by using lossless source coding. Whereas the complexity of Gallager's method heavily depends on the optimal symbol distribution, our scheme can treat any symbol distribution in the same manner and has an advantage in the complexity.

1.3.1 Asymptotically Optimal Coding Schemes for Asymmetric Settings

We first apply the idea of using a lossless code to polar codes and propose a new polar coding scheme for channel coding and lossy source coding. In the asymmetric setting, Sutter et al. [41] proposed another channel code based on the same idea independently of our work. However, the construction of their scheme is a little complicated because two polar codes are concatenated, one of which is for lossless coding and the other is for channel coding. Furthermore, due to this concatenation, the convergence speed of the decoding error probability is slow; their scheme requires block length n^2 to achieve the error probability which can be achieved by block length n for the symmetric cases. We show that channel coding (or lossy source coding) and lossless coding can be carried out simultaneously by one polar code using a characteristic of polar codes, and we also show that our scheme can achieve the same asymptotic performance as the symmetric cases.

Next we propose a new coding scheme using LDPC codes. Differently from polar codes, it is difficult to carry out lossless compression together with channel coding or lossy coding by one LDPC code. Then we propose a kind of concatenated code which is constructed by combining an LDPC code

with another code for lossless coding. For the lossless coding we use a variable length code in our scheme whereas a fixed length lossless code using a LDPC code is adopted in Miyake-Muramatsu scheme. By using the variable length code, lossless compression can be executed without decoding error.

Generally, channel coding and lossy source coding are dual to each other and a good lossy code can be constructed by exchanging the roles of encoding and decoding of a good channel code. In fact, in the polar coding scheme proposed in this thesis, the encoding of the lossy code and the decoding of the channel code are almost the same and vice versa. However, in the proposed scheme for LDPC codes, the code construction is a little different between channel coding and lossy coding.

First we consider lossy coding. In lossy coding, a source sequence is transformed into a distorted sequence, which we call a vector-quantized sequence to follow [40]. The vector-quantized sequence is next compressed by a variable length lossless code, or more precisely, a fixed-to-variable (FV) lossless code. In FV codes, a source sequence on a set $\mathcal{X}$ with a fixed length is compressed to another sequence on set $\tilde{\mathcal{X}}$ with a length depend on the source sequence. Here, if every sequence on $\tilde{\mathcal{X}}$ can appear as a prefix of a sequence of codewords then the decoder of the FV code can be used as an encoder for sequences on $\tilde{\mathcal{X}}$. Well-designed FV codes such as Huffman codes satisfy this property if $\tilde{\mathcal{X}}$ is binary. However, the construction of Huffman code is computationally infeasible when the size of a source alphabet $\mathcal{X}$ is large. To avoid this problem, we use an arithmetic code in the proposed scheme. Generally an arithmetic code does not satisfy the above property and therefore the decoder of the proposed lossy coding scheme cannot be used as an encoder for channel coding.

In the proposed coding scheme for asymmetric channels, a message is encoded into an LDPC codeword with a desired symbol distribution. In decoding, first the LDPC codeword is recovered from the received sequence and next the message is extracted from the codeword. Here the conversion of messages to LDPC codewords corresponds to decoding of lossy source coding in principle, but, the decoder of an arithmetic code cannot be used for this purpose as described before. Such an invertible conversion of a sequence with a symbol distribution to another sequence with a different distribution is called *homophonic coding*. This problem is first considered in the context of cryptography [42][43], where a biased sequence causes an attack of the ciphertext and completely random (i.e. uniform) sequences are required. Hoshi-Han [44] generalized this problem to generation of sequences with ar-

bitrary distributions. They showed that asymptotically optimal rate can be achieved by an interval algorithm similar to that for random number generation [45]. We apply the homophonic code with the interval algorithm to generation of an LDPC codeword with the desired symbol distribution and show that channel capacity can be achieved by the proposed scheme.

1.3.2 Practical Algorithms for LDPC Codes

Whereas the proposed scheme using polar codes can achieve the Shannon bound asymptotically by a polynomial-time algorithm, the scheme using LDPC codes requires computation of NP-complete problem and investigation of suboptimal algorithms is essential.

First we propose a practical algorithm for an arithmetic code in lossy coding and a homophonic code in channel coding. In these codes, the probability distribution of each element in LDPC codewords is necessary to compute. The coding rate is improved as the distribution is computed more accurately. It is well known that such a distribution can be approximated accurately by the BP. However, a naive application of the BP causes a quadratic complexity in the blocklength, which is a little unrealistic although it is a polynomial complexity. Then we propose an algorithm for the approximation of the probability such that the number of executions of the BP is reduced and nearly linear complexity can be achieved.

Next we investigate some algorithms for vector-quantization in encoding of the lossy code and for recovering of the LDPC codeword in decoding of the channel code. These problems are the same kind of optimization as those for symmetric settings and there have been some known efficient algorithms. In the symmetric settings, these problems simply correspond to the encoding of the lossy code and the decoding of the channel code, respectively, and we refer to the problems by these counterparts in the rest of this section. For these problems we improve the known algorithms to obtain better performance.

It is known that the RBP algorithm performs efficiently in encoding of LDPC codes or LDGM codes for lossy coding. However, when it is applied to LDPC codes, it does not necessarily converge to an LDPC codeword and some additional procedure is required to obtain a valid codeword. In [34], the encoder executes the RBP repeatedly until the RBP converges. Although this algorithm finally obtains a good codeword, it is not time-efficient since it sometimes takes many times of RBP execution. Then we propose an algorithm which can choose a good LDPC codeword from the result of one RBP

execution by a technique of matroid optimization.

For decoding of LDPC codes in channel coding, we investigate a new LP decoding technique. When using BP decoding, it is known that nonbinary LDPC codes achieve fairly good performance in a short blocklength [13][46]. Although LP decoding is also extended to nonbinary LDPC codes by Flanagan et al. [47], it is unrealistic to apply their scheme to nonbinary LDPC codes on large finite fields because of its large complexity. Therefore we propose another LP decoding technique for nonbinary LDPC codes in which the complexity increases slowly in the size of the finite field. Although our scheme relaxes a maximum likelihood decoding problem more loosely to an LP problem than their scheme, the deterioration of the decoding error probability is small.

1.4 Organization of the Thesis

- In **Chapter 2** we formulate the framework of channel coding and source coding and give basic coding theorems proved by Shannon. We also introduce an arithmetic code and an interval algorithm as asymptotically optimal coding schemes for lossless coding and homophonic coding, respectively.
- In **Chapter 3** we construct an asymptotically optimal coding scheme using polar codes for channel coding and lossy source coding in the asymmetric settings. We first introduce polar codes proposed by Arikan. After the review of polar codes, we discuss polar codes for lossless coding with side information. Using this result, we construct polar codes for our coding problems and show that the Shannon bound can be achieved asymptotically.
- In **Chapter 4** we investigate coding schemes using LDPC codes. We review LDPC codes and introduce the notion of hash property [48], which is useful for theoretical analysis of LDPC codes. We construct new coding schemes by combining a LDPC code with a lossless code for lossy coding and with homophonic code for channel coding. We show that the proposed schemes can achieve the Shannon bound.
- In the proof of the optimality of the coding schemes in Chapter 4, we assume that the optimal encoding and decoding can be performed. We investigate practical algorithms in **Chapter 5** for the LDPC codes with some simulation results. First we propose an algorithm using the BP for computation of marginal probabilities of LDPC codes required for

arithmetic coding and homophonic coding. Next we consider vector-quantization for lossy source coding. We introduce the RBP algorithm and propose an auxiliary algorithm to obtain a good LDPC codeword from the result of the RBP algorithm. Finally we give an LP decoding technique for nonbinary LDPC codes in channel coding. We also discuss the difference from another LP decoding scheme proposed by Flanagan et al.

- Finally the obtained results of this thesis are summarized in **Chapter 6** with some future works.

1.5 Notation

We use the following notation in this thesis.

First, calligraphic letters such as $\mathcal{X}, \mathcal{Y}, \mathcal{S}, \mathcal{T}, \cdots$ denote sets of some symbols or events. Upper case letters $X, Y, \cdots$ usually denote random variables on corresponding sets $\mathcal{X}, \mathcal{Y}, \cdots$. Lower case letters $x, y, \cdots$ are used to denote realizations of them.

Probability of an event A is denoted by $\Pr[A]$. Especially we write e.g., $P_X[A]$ to declare that the event A depends on a random variable X. Similarly, we write $P_X[A|B]$ for the conditional probability of A given B when events A and B depend on X. The conditional probability distribution of X given Y is denoted by $P_{X|Y}$. We also write $P_X(x) = P_X[X = x]$ and $P_{X|Y}(x|y) = P_{X|Y}[X = x|Y = y]$. The expectation of $f(X) \in \mathbb{R}$ over a random variable X is denoted by $\mathrm{E}_X[f(X)]$. We sometimes write $\mathrm{E}_{P_X}[f(X)]$ to express that X follows distribution P_X. In the case that there can be two distributions of X, we denote the second distribution by Q_X.

A sequence of symbols is denoted by, e.g., $x_1^n = (x_1, x_2, \cdots, x_n)$. A sequence of such sequences is denoted by $(x^n_{(1),1}, x^n_{(2),1} \cdots, x^n_{(l),1})$. For a set of indices $\mathcal{A}$, a subvector $\{x_i\}_{i\in\mathcal{A}}$ is denoted by $x_{\mathcal{A}}$. We sometimes write e.g. $\boldsymbol{x}$ instead of x_1^n when the length is obvious from the context.

An $n \times k$ matrix is denoted by $H = \{h_{ij}\}_{(i,j)\in\mathcal{I}\times\mathcal{J}}$ for $\mathcal{I} \equiv \{1, 2, \cdots, n\}$ and $\mathcal{J} = \{1, 2, \cdots, k\}$. The submatrix of H which consists of rows with indices contained in $\mathcal{A}$ is denoted by $H_{\mathcal{A}}$. Similarly, $H_{\mathcal{A},\mathcal{B}}$ denotes the submatrix consisting of rows with indices in $\mathcal{A} \subset \mathcal{I}$ and columns with indices in $\mathcal{B} \subset \mathcal{J}$.

Other notation is listed as follows.

$i : j$	Set of indices $i, i+1, \cdots, j$ if $i \leq j$. Otherwise it denotes the empty set.

$\mathrm{GF}(q)$	Finite field with size q.
$\lvert\mathcal{A}\rvert$	Cardinality of a set $\mathcal{A}$.
$\mathcal{A}^*$	Family of finite sequences of $\mathcal{A}$ defined by $\bigcup_{l=1}^{\infty} \mathcal{A}^l$.
$w(H)$	Number of nonzero entries of a matrix H.
log	Logarithm with base 2.
ln	Natural logarithm.
$\lceil x \rceil$	The smallest integer larger than or equal to x.
$\lfloor x \rfloor$	The largest integer less than or equal to x.
$\lfloor x \rfloor_l$	The largest number $y \leq x$ such that $2^l y$ is an integer.
$\mathbb{1}[A]$	Indicator function which takes value 1 if A is true and 0 otherwise.
$H(X)$	Entropy defined by $H(X) \equiv -\sum_{x \in \mathcal{X}} P_X(x) \log P_X(x)$. Sometimes we write $H(P_X)$ for the same value.
$H(X\|Y)$	Conditional entropy $H(X\|Y) \equiv -\sum_{x \in \mathcal{X},\, y \in \mathcal{Y}} P_{XY}(x,y) \log P_{X\|Y}(x\|y)$.
$I(X,Y)$	Mutual information defined by $I(X;Y) \equiv H(X) - H(X\|Y)$.
$I(W)$	Symmetric capacity of a channel W (see (3.1)).
$D(P_X \\| Q_X)$	Relative entropy or KL divergence defined by $D(P_X \\| Q_X) \equiv \sum_{x \in \mathcal{X}} P_X(x) \log(P_X(x)/Q_X(x))$.
$\\|P_X - Q_X\\|$	Variational distance defined by $\\|P_X - Q_X\\| \equiv (1/2) \cdot \sum_{x \in \mathcal{X}} \lvert P_X(x) - Q_X(x)\rvert$.

Chapter 2

Preliminaries: Coding Theorems

In this chapter we formulate channel coding and source coding with their fundamental bound on the efficiency called Shannon bound. We also introduce homophonic coding, which is quite similar to lossless source coding. Although this thesis mainly treats channel coding and lossy source coding, we review efficient coding schemes for lossless source coding and homophonic coding, which are useful for our coding problems when they are combined with LDPC codes.

2.1 Channel Coding

In channel coding we consider transmitting a message via a channel W. The channel W takes an input x in a finite set $\mathcal{X}$ and outputs y in a finite set $\mathcal{Y}$ with probability $W(y|x)$. We always assume that W is memoryless, that is, for a sent sequence $x_1^n = (x_1, x_2, \cdots, x_n)$, the channel outputs $y_1^n = (y_1, y_2, \cdots, y_n)$ occurs with probability

$$W^n(y_1^n|x_1^n) \equiv \prod_{i=1}^{n} W(y_i|x_i)\,. \tag{2.1}$$

Now we construct a channel code with block length n. Let $\mathcal{C}$ be a subset of $\mathcal{X}^n$. An encoder is defined as a bijection $\varphi : \{1, 2, \cdots, |\mathcal{C}|\} \to \mathcal{C}$ and the encoder send $\varphi(M)$ for message $M = 1, 2, \cdots, |\mathcal{C}|$. Here $\mathcal{C}$ is called a *codebook* or simply a *code* and each member of $\mathcal{C}$ is called a *codeword*. A decoder is defined as a map $\psi : \mathcal{Y}^n \to \{1, 2, \cdots, |\mathcal{C}|\}$. This estimates the sent message M by $\psi(Y_1^n)$ for received sequence Y_1^n. The decoding is successful if $M = \psi(Y_1^n)$ and the decoding error probability is defined by $\Pr[M \neq \psi(Y_1^n)]$, where we assume that the message is distributed uniformly on $\{1, 2, \cdots, |\mathcal{C}|\}$. Here the message can be represented by a binary sequence with length $\lceil \log \mathcal{C} \rceil$. We

define the *coding rate* by $R = \frac{\log|\mathcal{C}|}{n}$ as the ratio of the message length to the block length of codewords.

For this framework of channel coding, a quantity called *channel capacity* $C(W)$ plays a central role. For a random variable $X \in \mathcal{X}$, let Y be a random variable such that $P_{XY}(x,y) = P_X(x)W(y|x)$ for $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. Then the channel capacity $C(W)$ is defined as

$$\begin{aligned} C(W) &\equiv \max_X I(X;Y) \\ &= \max_X \sum_{x,y} P_{XY}(x,y) \log \frac{P_{XY}(x,y)}{P_X(x)P_Y(y)} \,. \end{aligned} \tag{2.2}$$

The channel capacity is usually computed numerically by e.g. Arimoto-Blahut algorithm [49][50]. A *symmetric channel* is a special case that the capacity can be computed analytically.

Definition 2.1 (Symmetric Channel)**.** *A memoryless channel is said to be symmetric if* $\sum_{x\in\mathcal{X}} W(y_j|x) = |\mathcal{X}|/|\mathcal{Y}|$ *for all* $j = 1, 2, \cdots, |\mathcal{Y}|$ *and the members of the set* $\{(W(y_1|x), W(y_2|x), \cdots, W(y_{|\mathcal{Y}|}|x))\}_{x\in\mathcal{X}}$ *are permutations of each other for* $\mathcal{Y} = \{y_1, y_2, \cdots, y_{|\mathcal{Y}|}\}$.

For any symmetric channel, the maximum in (2.2) is achieved by the uniform distribution on $\mathcal{X}$, and the capacity is given by

$$C(W) = \log|\mathcal{Y}| + \sum_y W(y|x) \log W(y|x) \,, \tag{2.3}$$

where $x \in \mathcal{X}$ is arbitrary (see, e.g., [51, Chapter 7] for detail).

The following theorem shows that the channel capacity is the tight bound on the achievable rate with arbitrarily small decoding error probability.

Theorem 2.1 (Channel Coding Theorem)**.** *Let* $P_{\mathrm{e}}^{(n)}$ *be the decoding error probability of a channel code with block length* n *and coding rate* R*.*

(direct part [1]) If $R < C(W)$ *then there exists a sequence of* $(\varphi^{(n)}, \psi^{(n)})$ *such that*

$$\lim_{n\to\infty} P_{\mathrm{e}}^{(n)} = 0 \,. \tag{2.4}$$

(converse part [52]) If $R > C(W)$ *then for any* $(\varphi^{(n)}, \psi^{(n)})$

$$\lim_{n\to\infty} P_{\mathrm{e}}^{(n)} \to 1 \,. \tag{2.5}$$

The original proof of the direct part by Shannon is based on *random code*, where each codeword is drawn randomly from $P_{X_1^n}$. Here $X_1^n =$

$(X_1, X_2, \cdots, X_n)$ denotes i.i.d. copies of X which attains the maximum in (2.2). Since such a code has exponential space and time complexity, construction of codes with some "structure" has been considered extensively.

The most fundamental family of codes is *linear codes*. The codebook of a linear code can be expressed by

$$\mathcal{C} = \{x_1^n \in (\mathrm{GF}(q))^n : x_1^n H = v_1^k\} \tag{2.6}$$

for an $n \times k$ matrix H and a vector $v_1^k \in (\mathrm{GF}(q))^k$, where $\mathrm{GF}(q) = \mathcal{X}$ is a finite field with size $q = |\mathcal{X}|$. This matrix H is called a parity check matrix. The size of the codebook is given by $|\mathcal{C}| = |q|^{n-\mathrm{rank}H}$. Then, the coding rate of this code is expressed as

$$R = \left(1 - \frac{\mathrm{rank}H}{n}\right) \log q\,. \tag{2.7}$$

The linear code in (2.6) can also be expressed by

$$\mathcal{C} = \{u_1^m G + \tilde{v}_1^n : u_1^m \in (\mathrm{GF}(q))^m\}, \quad m = n - \mathrm{rank}\, H, \tag{2.8}$$

for an $m \times n$ matrix G and a vector $\tilde{v}_1^n$ such that $GH = 0$ and $\tilde{v}_1^n H = v_1^k$. Such G and $\tilde{v}_1^n$ always exist if $n - \mathrm{rank}H > 0$ and G is called a generator matrix. Csiszár [53] showed that there exists a linear code which achieves the channel capacity asymptotically for any symmetric channel.

2.2 Lossy Source Coding

Lossy source coding treats a data compression such that some loss of information, or distortion, is allowed. A series of studies on lossy source coding is called *rate-distortion theory*. This theory literally seeks the bound on the tradeoff between the coding rate and the distortion. If the distortion is allowed to be arbitrarily large then the sender has not to transmit any information and the rate can be zero. On the other hand, if any distortion is not admissible then this problem becomes a lossless coding discussed in the next section. The rate-distortion theory reveals for cases between the above two extreme cases how the rate can be small under a given admissible distortion, or equivalently, how the distortion can be reduced for a given coding rate.

Let Y denote the random variable of information source[*1]. After encoding and decoding, each source symbol $y \in \mathcal{Y}$ is reproduced to a symbol $x \in \mathcal{X}$.

[*1] We use this notation rather than X for consistency with channel coding.

Here we assume that $\mathcal{X}$ and $\mathcal{Y}$ are finite sets. A *distortion measure* is a function $d : \mathcal{Y} \times \mathcal{X} \to [0, \infty)$ and the distortion of x from y is measured by $d(y, x)$.

As in the case of channel coding, we consider a block code $\mathcal{C} \subset \mathcal{X}^n$ for block length n. An encoder is defined by a map $\varphi : \mathcal{Y}^n \to \{1, 2, \cdots, |\mathcal{C}|\}$ and the decoder is a bijection $\psi : \{1, 2, \cdots, |\mathcal{C}|\} \to \mathcal{X}^n$. The coding rate is given by $\frac{\log |\mathcal{C}|}{n}$. For sequence Y_1^n of random variables independent and identically distributed by P_Y, performance of a coding scheme (φ, ψ) is evaluated by the average distortion

$$\frac{1}{n} \mathrm{E}_{Y_1^n} [d^n(Y_1^n, \psi(\varphi(Y_1^n)))] , \tag{2.9}$$

where

$$d^n(y_1^n, x_1^n) \equiv \sum_{i=1}^{n} d(y_i, x_i) . \tag{2.10}$$

Now we give the coding theorem for lossy source coding. Let define the *rate-distortion function* $R(D)$ by

$$\begin{aligned} R(D) &\equiv \min_{X : \mathrm{E}_{XY}[d(Y,X)] \le D} I(X; Y) \\ &= \min_{X : \mathrm{E}_{XY}[d(Y,X)] \le D} \sum_{x,y} P_{XY}(x, y) \frac{P_{XY}(x, y)}{P_X(x) P_Y(y)} . \end{aligned} \tag{2.11}$$

We define a class of distortion measure that the rate-distortion function can be easily computed on some condition similarly to symmetric channels.

Definition 2.2 (Symmetric Distortion Measure)**.** *A distortion measure $d(\cdot, \cdot)$ for $\mathcal{X} = \{x_1, x_2, \cdots, x_{|\mathcal{X}|}\}$ and $\mathcal{Y} = \{y_1, y_2, \cdots, y_{|\mathcal{Y}|}\}$ is said to be symmetric if the members of the set $\{(d(y_1, x_i), d(y_2, x_i), \cdots, d(y_{|\mathcal{Y}|}, x_i))\}_{i=1,2,\cdots,|\mathcal{X}|}$ are permutations of each other and the members of the set $\{(d(y_i, x_1), d(y_i, x_2), \cdots, d(y_i, x_{|\mathcal{X}|}))\}_{i=1,2,\cdots,|\mathcal{Y}|}$ are permutations of each other.*

We can show that if the source Y is uniformly distributed and the distortion measure is symmetric then the distribution P_X which achieves the minimum in (2.11) is a uniform distribution (see, e.g., [51, Chapter 10] for detail).

The following theorem shows that $R(D)$ is the tight bound of the achievable coding rate within average distortion D.

Theorem 2.2 (Rate-Distortion Theorem [26])**.** *There exists a sequence of*

lossy code $(\varphi^{(n)}, \psi^{(n)})$ *with coding rate* R *and block length* n *such that*

$$\lim_{n\to\infty} \frac{1}{n} \mathrm{E}_{Y_1^n}[d^n(Y_1^n, \psi^{(n)}(\varphi^{(n)}(Y_1^n)))] \leq D \tag{2.12}$$

if and only if $R \leq R(D)$.

2.3 Coding Scheme for Asymmetric Settings

In channel coding and lossy source coding, the symbol distribution of the codebook $|\mathcal{C}|$ is necessary to be close to P_X which achieves the channel capacity in (2.2) or the rate-distortion function in (2.11). In the case that P_X is a uniform distribution, such a codebook can be constructed by a linear code in (2.6) or (2.8). This is the case for channel coding with symmetric channels or lossy source coding with uniform sources and symmetric distortion measures. However, for source coding with asymmetric channels or lossy source coding with nonuniform sources and/or asymmetric distortion measures, P_X is not uniform in general and it is difficult to construct linear codes which achieve the Shannon bound.

The most popular technique to achieve the Shannon bound in the framework of linear codes is Gallager's method[*2] [35, p. 208]. This technique can be illustrated by the following example. Consider channel coding of an asymmetric channel such that the optimal input distribution is $(P_X(0), P_X(1)) = (2/3, 1/3)$ with alphabet $\mathcal{X} = \{0, 1\}$. This input distribution can be realized by considering a ternary linear code with an extended alphabet $\mathcal{X}' = \{a, b, c\}$. Mapping symbols $a, b \in \mathcal{X}'$ to $0 \in \mathcal{X}$ and $c \in \mathcal{X}'$ to $1 \in \mathcal{X}$ in codewords, we obtain codewords on $\mathcal{X}$ with the desired distribution.

Generally, when the optimal input distribution is expressed by $P_X(x_i) = p_i/\tilde{q}$ for integers $\{p_i\}_{i=1,2,\cdots,|\mathcal{X}|}$ and $\tilde{q} = \sum_{i=1}^{|\mathcal{X}|} p_i$, the Gallager's method first constructs a linear code over $\tilde{\mathcal{X}} = \mathrm{GF}(\tilde{q})$ and next transforms each symbol $\tilde{x} = 1, 2, \cdots, \tilde{q}$ in codewords by the map

$$Q : \tilde{x} \mapsto \begin{cases} x_1, & \tilde{x} = 1, 2 \cdots, p_1, \\ x_2, & \tilde{x} = p_1 + 1,\ p_1 + 2, \cdots, p_1 + p_2, \\ \vdots & \qquad \vdots \\ x_{|\mathcal{X}|}, & \tilde{x} = \sum_{i=1}^{|\mathcal{X}|-1} p_i + 1,\ \sum_{i=1}^{|\mathcal{X}|-1} p_i + 2, \cdots, \tilde{q}. \end{cases} \tag{2.13}$$

[*2] This technique is generally applicable to any codebook with uniform symbol distributions but we only refer to linear codes for brevity.

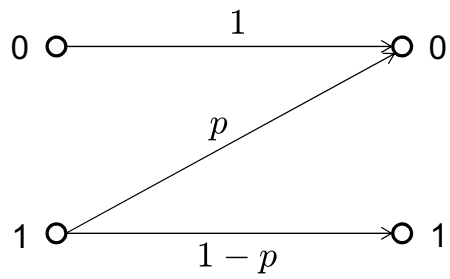

Fig. 2.1. Z-channel.

We can see that this map does not worsen the achievable rate for channel coding by the following discussion; the map Q is equivalent to the case that the auxiliary channel $\tilde{W} : \tilde{\mathcal{X}} \to \mathcal{X}$ such that

$$\tilde{W}(x|\tilde{x}) = \begin{cases} 1, & \text{if } Q(\tilde{x}) = x, \\ 0, & \text{otherwise.} \end{cases} \tag{2.14}$$

is pre-cascaded to the original channel W. This cascade does not decrease the channel capacity since we have $I(\tilde{X};Y) = I(X;Y) - I(X;Y|\tilde{X}) = I(X;Y)$ for the cascade of channels $\tilde{W}(x|\tilde{x})$ and $W(y|x)$. It is easy to see that this map does not increase the rate-distortion function in the similar way.

This Gallager's method is simple and applicable widely. However, a large size of extended alphabet $\tilde{\mathcal{X}}$ is required if the optimal distribution $P_X(\cdot)$ cannot be approximated by simple rational numbers. In such cases, the complexity of decoding increases considerably. Practically, we seldom have the case that the optimal input distribution is biased as in the above example where the optimal distribution is $(P_X(0), P_X(1)) = (2/3, 1/3)$. For example, consider a *Z-channel* in Fig. 2.1, which is one of the most fundamental asymmetric channels. It is the binary-input and binary-output channel such that for $p \in (0,1)$

$$\begin{aligned} W(0|0) &= 1, \\ W(1|0) &= 0, \\ W(0|1) &= p, \\ W(1|1) &= 1 - p. \end{aligned} \tag{2.15}$$

In this channel, the optimal input distribution satisfies $1/2 < P_X(0) < 1 - 1/\mathrm{e} = 0.6321\cdots < 2/3$ for all $p \in (0,1)$, where e is the base of the natural logarithm. As represented by a Z-channel, the optimal input distribution does not deviate from uniform so much except for very peculiar channels,

especially for binary input channels. In such a case, it requires a quite large alphabet size to approximate the optimal input distribution accurately.

2.4 Lossless Source Coding and Homophonic Coding

Lossless source coding and homophonic coding are frameworks of transforming a source sequence into an output sequence without loss of information. In both problems, output sequences are desirable to be short as much as possible. Whereas there is no constraint on the code construction in lossless coding, it is required in homophonic coding that the probability distribution of the output sequence is assured to be close to (or strictly equal to) a target distribution.

Generally, when a sequence is compressed by a "good" lossless encoder, the probability distribution of the output sequence is close to the uniform distribution since the uniform distribution can express the maximum amount of information. Therefore, lossless coding is closely related to homophonic coding such that the target distribution is uniform. However, in lossless coding, except for rare cases, the output distribution is not strictly equal to a uniform distribution. Furthermore, there sometimes exist subsequences which cannot appear in output sequences even for some known good lossless coders. In homophonic coding, a randomized algorithm is used to assure the output distribution. Therefore, the output sequence is sometimes different even if the same source sequence is encoded in homophonic coding.

2.4.1 Entropy and Lossless Coding

Let X be a random variable on a finite set $\mathcal{X}$. In this section, we consider compression of a (sequence of) symbol(s) in $\mathcal{X}$ into a sequence on $\mathcal{U}$. For simplicity we assume that $\mathcal{U} = \{0, 1\}$.

Define $\mathcal{A}^*$ for any set $\mathcal{A}$ by

$$\mathcal{A}^* \equiv \bigcup_{l=1}^{\infty} \mathcal{A}^l \,. \tag{2.16}$$

A lossless encoder is defined as an injection

$$\varphi : \mathcal{X} \to \{0, 1\}^* \tag{2.17}$$

and a decoder is given by $\psi = \varphi^{-1}$. Each $\varphi(x)$ is called a codeword and the length of the codeword $\varphi(x)$ is denoted by $l(x)$. We sometimes write l_i

instead of $l(a_i)$ when $\mathcal{X}$ is expressed as $\mathcal{X} = \{a_1, a_2, \cdots, a_{|\mathcal{X}|}\}$. Note that not all maps in (2.17) are appropriate as a lossless encoder. For example, when we use an encoder $(\varphi(1), \varphi(2), \varphi(3), \varphi(4)) = (0, 010, 01, 10)$ for $\mathcal{X} = \{1, 2, 3, 4\}$, we cannot figure out which input sequence "144", "24" or "32" corresponds to an output sequence "01010". A *uniquely decodable* code is a class of codes in which such a problem does not occur. It is defined as a code such that any pair of distinct input sequences $(x_1, x_2, \cdots, x_n) \in \mathcal{X}^n$ and $(\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_{\tilde{n}}) \in \mathcal{X}^{\tilde{n}}$ satisfies

$$(\varphi(x_1)\varphi(x_2)\cdots\varphi(x_n)) \neq (\varphi(\tilde{x}_1)\varphi(\tilde{x}_2)\cdots\varphi(\tilde{x}_{\tilde{n}}))\,. \tag{2.18}$$

An important class of uniquely decodable codes is a *prefix code*. A code is called a prefix code (or an instantaneous code) if no codeword is a prefix of any other codeword. Whereas decoding of a uniquely decodable code cannot be done until reading the entire sequence of codewords in general, each input symbol of a prefix code can be decoded instantaneously from the codeword corresponding to the input symbol. Based on Kraft inequality, it is known that it suffices to consider only prefix codes to examine shortest code lengths in uniquely decodable codes.

Theorem 2.3 (Kraft inequality [54]). *For any uniquely decodable code, the set of code lengths* $(l_1, l_2, \cdots, l_{|\mathcal{X}|})$ *must satisfy*

$$\sum_{i=1}^{|\mathcal{X}|} 2^{-l_i} \leq 1\,. \tag{2.19}$$

Conversely, a set of code lengths $(l_1, l_2, \cdots, l_{|\mathcal{X}|})$ *is possible as a prefix code if* (2.19) *is satisfied.*

A prefix code can be related to a *code tree*. The code tree for the set of codewords $\{\varphi(x)\}_{x\in\mathcal{X}}$ is a binary tree such that (1) each edge corresponds to an output symbol, (2) each leaf corresponds to a source symbol $x \in \mathcal{X}$ and (3) the path to each leaf from the root node is the codeword of the source symbol. Fig. 2.2 shows the code tree for code $(\varphi(\mathrm{a}), \varphi(\mathrm{b}), \varphi(\mathrm{c}), \varphi(\mathrm{d}), \varphi(\mathrm{e}), \varphi(\mathrm{f})) = (00, 01, 1000, 1001, 101, 11)$. From the viewpoint of a code tree, a prefix code can be regarded as a code such that the code tree has no codeword at inner nodes.

Now we return to lossless coding of a random variable X. For a lossless code φ, the expected code length is given by

$$\mathrm{E}_X[l(X)] = \sum_{x\in\mathcal{X}} P_X(x)l(x)\,. \tag{2.20}$$

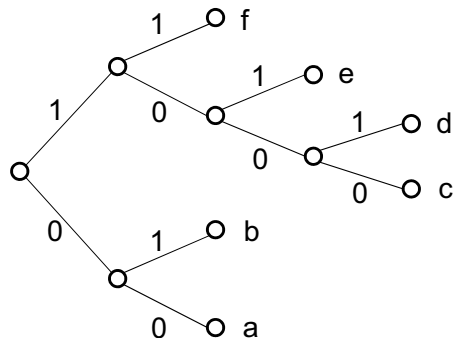

Fig. 2.2. A code tree of a prefix code.

The expected code length is closely related to the entropy of the source X, which is defined by

$$H(X) \equiv -\sum_{x \in \mathcal{X}} P_X(x) \log P_X(x) \,. \tag{2.21}$$

Shannon showed that the expected code length of any uniquely decodable code cannot be smaller than the entropy, that is,

$$\mathrm{E}_X[l(X)] \geq H(X) \,. \tag{2.22}$$

It is known that Huffman code [19] achieves the shortest expected code length and it satisfies

$$\mathrm{E}_X[l(X)] < H(X) + 1 \,. \tag{2.23}$$

Remark 2.1. As mentioned before, lossless coding can be regarded as a special case of lossy coding such that the allowed distortion is $D = 0$ and the distortion measure is

$$d(x, \tilde{x}) = \begin{cases} 0, & \text{if } x = \tilde{x}, \\ 1, & \text{otherwise.} \end{cases} \tag{2.24}$$

In fact, the rate-distortion function in (2.11) becomes $R(D) = H(X)$ in this case. However, when lossless coding is considered in the framework of lossy coding, the code usually becomes an FF (fixed-to-fixed) code. An FF lossless code sometimes fails in decoding although the decoding error probability can be arbitrarily close to zero. Furthermore, in the framework of lossy coding, it is a little difficult to consider the case that the sources $X_1, X_2 \cdots$ are not independent to each other. Therefore we consider lossless coding separately from lossy coding to derive useful results.

2.4.2 Arithmetic Coding

When using Huffman code, there is a loss in the code length up to 1 bit per symbol compared to the optimal code length $H(X)$ as in (2.22) and (2.23). This inefficiency can be avoided by regarding multiple symbols as one symbol; if $X_1, X_2, \cdots, X_n$ are i.i.d., the entropy can be expressed as

$$H(X_1, X_2, \cdots, X_n) = nH(X) \tag{2.25}$$

and therefore $X_1, \cdots, X_n$ can be encoded in at most $nH(X) + 1$ bits by Huffman code such that n symbols are regarded as one symbol in $\mathcal{X}^n$. In this case, the coding rate per symbol is bounded by $H(X) + 1/n$. However, this technique is not practical because the complexity of the construction of Huffman code grows exponentially in the block length n.

Arithmetic coding is one solution for this problem. This coding scheme can encode and decode a sequence with linear complexity in the block length n and the coding rate per symbol approaches the entropy, although the coding rate is a little worse than Huffman code in a finite block length. Arithmetic coding has many different versions [21][22][55] and we introduce one based on Shannon-Fano-Elias code in this thesis.

Let us consider a sequence of random variables $X_1^n = (X_1, X_2, \cdots, X_n) \in \mathcal{X}^n$, components of which may not be i.i.d. We assume that $P_{X_1^n}(x_1^n) > 0$ for all $x_1^n \in \mathcal{X}^n$. Let $\leq$ be a (total) order of $\mathcal{X}$ and define an order on $\mathcal{X}^n$ by the lexicographic order induced by the ordered set $(\mathcal{X}, \leq)$, which we also refer to by $\leq$. For convenience, we write the largest element which is smaller than x_1^n in the order $\leq$ by $x_1^n - 1$, that is, $x_1^n - 1 = \max\{a_1^n : a_1^n < x_1^n\}$.

Define the cumulative distribution of $P_{X_1^n}(x_1^n)$ by

$$F(x_1^n) \equiv \sum_{a_1^n \leq x_1^n} P_{X_1^n}(a_1^n)\,. \tag{2.26}$$

Also, we define the mid-point between $F(x_1^n - 1)$ and $F(x_1^n)$ by

$$\begin{aligned} \bar{F}(x_1^n) &\equiv \frac{1}{2}(F(x_1^n - 1) + F(x_1^n)) \\ &= \sum_{a_1^n < x_1^n} P_{X_1^n}(a_1^n) + \frac{1}{2} P_{X_1^n}(x_1^n)\,. \end{aligned} \tag{2.27}$$

Let $\lfloor p \rfloor_l \in \{0,1\}^l$ denote the first l bits of the binary expansion of p for $p \in [0,1)$ and $l \in \mathbb{N}$. If $\lfloor p \rfloor_l = a_1 a_2, \cdots a_l$ then p can be expressed as

$p = 0.a_1a_2, \cdots a_{l-1}a_l \cdots$. For notational convenience, we also refer to real number

$$0.a_1a_2, \cdots a_{l-1}a_l \in \mathbb{R} \tag{2.28}$$

by $\lfloor p \rfloor_l$. We define the codeword of Shannon-Fano-Elias code by $\varphi(x_1^n) \equiv \lfloor \bar{F}(x_1^n) \rfloor_{l(x_1^n)}$ where $l(x_1^n) = \lceil -\log P_{X_1^n}(x_1^n) \rceil + 1$. Let $F^{-1}(p)$, $p \in [0,1]$, denote $x_1^n \in \mathcal{X}^n$ such that $F(x_1^n - 1) \le p < F(x_1^n)$. The decoder is defined as $\psi = F^{-1}$.

Theorem 2.4. *Shannon-Fano-Elias code is a prefix code and the expected code length satisfies*

$$\mathrm{E}_{X_1^n}[l(X_1^n)] < H(X_1^n) + 2\,. \tag{2.29}$$

See e.g. [51, Section 5.9] for the proof. It is easy to see $\psi = \varphi^{-1}$ since it holds from (2.28) that

$$\begin{aligned} F(x_1^n) > \lfloor \bar{F}(x_1^n) \rfloor_{l(x)} &\ge \bar{F}(x_1^n) - 2^{-\lceil -\log P_{X_1^n}(x_1^n) \rceil - 1} \\ &\ge \bar{F}(x_1^n) - \frac{1}{2} P_{X_1^n}(x_1^n) \\ &\ge F(x_1^n - 1)\,. \end{aligned} \tag{2.30}$$

In arithmetic coding, it is necessary to compute $F(x_1^n)$ and $F^{-1}(p)$ efficiently. First, $F(x_1^n)$ can be expressed as

$$\begin{aligned} F(x_1^n) &= \sum_{a_1^n \le x_1^n} P_{X_1^n}(x_1^n) \\ &= \sum_{i=1}^{n} \sum_{a_1^n : a_1^{i-1} = x_1^{i-1},\, a_i < x_i} P_{X_1^n}(a_1^n) + P_{X_1^n}(x_1^n) \\ &= \sum_{i=1}^{n} P_{X_1^i}(x_1^i) \sum_{a : a < x_i} P_{X_i | X_1^{i-1}}(a_i | x_1^{i-1}) + P_{X_1^n}(x_1^n)\,. \end{aligned} \tag{2.31}$$

Let c be the complexity for computation of a probability $P_{X_i|X_1^{i-1}}(x_i|x_1^{i-1})$. Then $\{P_{X_1^i}(x_1^i)\}_{i=1}^n$ can be computed with complexity $\mathrm{O}(cn)$ from $P_{X_1^i}(x_1^i) = P_{X_1^{i-1}}(x_1^{i-1}) \cdot P_{X_i|X_1^{i-1}}(x_i|x_1^{i-1})$. Combining this fact with (2.31), we can compute $F(x_1^n)$ with complexity $\mathrm{O}(cn|\mathcal{X}|)$.

Next we consider the computation of F^{-1}. Define $F_{(i)}(x_1^i)$ by

$$F_{(i)}(x_1^i) \equiv \sum_{a_1^i : a_1^i \le x_1^i} F_{X_1^i}(a_1^i)\,. \tag{2.32}$$

Then $F_{(n)}(x_1^n) = F(x_1^n)$ and

$$F_{(i)}(x_1^i) = F((x_1, x_2, \cdots, x_i, 0, \cdots, 0)) . \tag{2.33}$$

Let $F^{-1}(p) = a_1^n$ and assume that a_1^{i-1} is known. Note that a_1^n satisfies $F(a_1^n - 1) \leq p < F(a_1^n)$. Then, from (2.33) and the monotonicity of $F(x_1^n)$ in x_1^n, we have

$$F_{(i)}(a_1^i - 1) \leq p < F_{(i)}(a_1^i) . \tag{2.34}$$

Then we can compute a_i from a_1^{i-1} by

$$\begin{aligned} a_i &= \max\{a : F_{(i)}((a_1, a_2, \cdots, a_{i-1}, a) - 1) \leq p\} \\ &= \max\left\{ a : \sum_{x:x \leq a} P_{X|X_1^{i-1}}(x|a_1^{i-1}) \leq \frac{p - F^{(i-1)}(a_1^{i-1} - 1)}{P_{X_1^{i-1}}(a_1^{i-1})} \right\} \end{aligned} \tag{2.35}$$

and we can obtain a_1^n with complexity $\mathrm{O}(cn|\mathcal{X}|)$.

We see from the above discussion that the arithmetic coding can achieve near optimal coding rate with linear complexity in the block length although the rate in (2.29) is a little worse than that of Huffman code in (2.23).

2.4.3 Homophonic Coding and Interval Algorithm

In lossless source coding, we considered a transformation of a sequence $X_1, \cdots, X_n$ to a binary sequence almost uniformly distributed. When we apply the duality between channel coding and lossy source coding, we sometimes want a converse of lossless coding, that is, transformation of uniform binary sequence to a sequence $X_1, \cdots, X_n$. One may expect that a decoder of a lossless code works for this purpose. However, this is not always true; for example, Fig. 2.3 shows the code tree of Shannon-Fano-Elias code for source sequence X_1, X_2, X_3 such that each $X_i \in \mathcal{X} = \{\mathrm{a}, \mathrm{b}\}$ is independently distributed by $(P_X(\mathrm{a}), P_X(\mathrm{b})) = (1/3, 2/3)$. This coding tree is not a complete tree and has many incomplete inner nodes. A lossless decoder cannot be used as an encoder for binary sequences in such a case that the coding tree is not complete. For example, the lossless decoder in the figure is "confused" if sequence "111" is received because the lossless encoder never outputs this sequence.

Homophonic coding is a technique which transforms a sequence with some probability distribution into an invertible sequence with a different probability distribution. This problem was first considered in [42][43] to generate

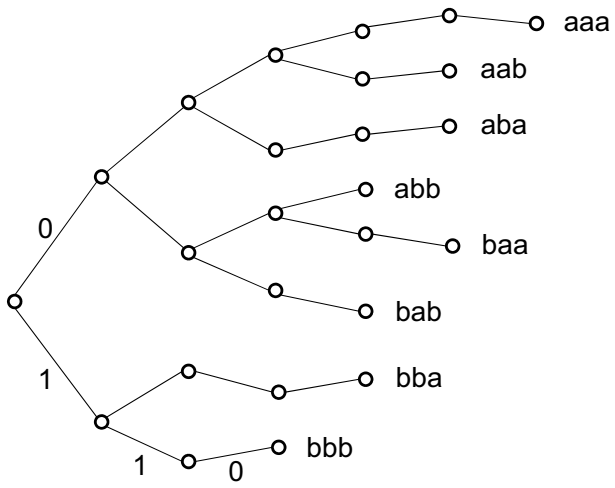

Fig. 2.3. A code tree of a Shannon-Fano-Elias code.

a (uniformly distributed) random sequence so that any sequence appears with the same frequency to ensure the security of a ciphertext [56]. Later Hoshi-Han [44] generalized this problem to generation of sequences with arbitrary distributions. A homophonic coding scheme is called P_X*-perfect* if each symbol of the codeword sequence for an input sequence of random variables follows the distribution P_X independently of each other.

Homophonic coding is illustrated by the following example. Consider a problem that we have a sequence of random variables U_i, $i = 1, 2, \cdots$, uniformly distributed on $\{0, 1\}$ independent of each other and we want to generate an i.i.d. sequence with distribution $(P_X(\mathrm{a}), P_X(\mathrm{b}), P_X(\mathrm{c})) = (18/30, 7/30, 5/30)$. This sequence can be generated by encoding

$$
\begin{aligned}
0 &\to \begin{cases} \mathrm{aa} & \text{with probability } 18/25, \\ \mathrm{ab} & \text{with probability } 7/25, \end{cases} \\
1 &\to \begin{cases} \mathrm{ac} & \text{with probability } 3/15, \\ \mathrm{b} & \text{with probability } 7/15, \\ \mathrm{c} & \text{with probability } 5/15. \end{cases}
\end{aligned}
\tag{2.36}
$$

In this encoding, $\{\mathrm{aa}, \mathrm{ab}, \mathrm{ac}, \mathrm{b}, \mathrm{c}\}$ can appear as a codeword and each codeword is also called a *homophone*. Applying this encoding to an infinite sequence $U_1, U_2, \cdots$, we can obtain an infinite sequence on $\mathcal{X}^\infty$ with the desired distribution, that is, this coding scheme is P_X-perfect.

The code structure for homophonic coding can be expressed by the *homophonic coding tree*, which illustrates how decoding can be executed for each

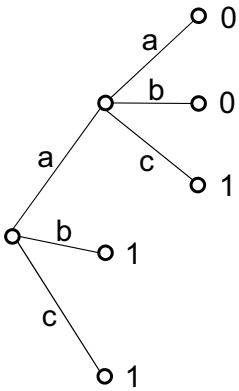

Fig. 2.4. Homophonic coding tree for encoding (2.36).

sequence on $\mathcal{X}^n$, corresponding to a code tree in lossless compression. Fig. 2.4 shows the homophonic code tree for homophonic coding in (2.36).

Generally the number of homophones to realize an output distribution is not finite and the homophonic coding tree is not a finite tree. Consider a generation of an i.i.d. sequence with distribution $(P_X(\mathrm{a}), P_X(\mathrm{b}), P_X(\mathrm{c})) = (4/9, 3/9, 2/9)$ from a sequence of i.i.d. random variables with distribution $(P_U(0), P_U(1)) = (1/2, 1/2)$. This sequence can be generated by encoding

$$
\begin{aligned}
0 \to &\begin{cases} \mathrm{a} & \text{with probability } 8/9, \\ \mathrm{bba} & \text{with probability } 8/9^2, \\ \mathrm{bbbba} & \text{with probability } 8/9^3, \\ \vdots & \qquad\qquad \vdots \end{cases} \\
1 \to &\begin{cases} \mathrm{c} & \text{with probability } 12/(3\cdot 9), \\ \mathrm{ba} & \text{with probability } 8/(3\cdot 9), \\ \mathrm{bc} & \text{with probability } 4/(3\cdot 9), \\ \mathrm{bbc} & \text{with probability } 12/(3\cdot 9^2), \\ \mathrm{bbba} & \text{with probability } 8/(3\cdot 9^2), \\ \mathrm{bbbc} & \text{with probability } 4/(3\cdot 9^2), \\ \vdots & \qquad\qquad \vdots \end{cases}
\end{aligned}
\tag{2.37}
$$

with homophonic coding tree in Fig. 2.5. However, we also seldom have a case that the homophonic coding tree can be expressed by a simple recursive structure.

A perfect homophonic coding algorithm for general input distributions P_U and output distributions P_X was first investigated by Hoshi-Han [44] and

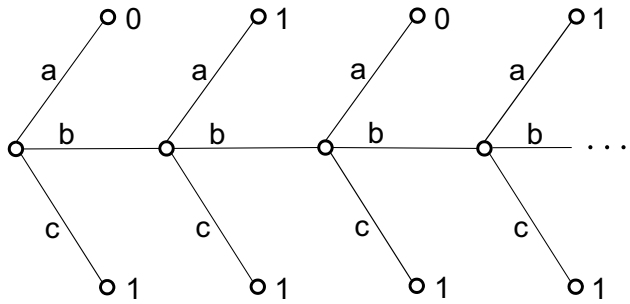

Fig. 2.5. Homophonic coding tree for encoding (2.37).

Algorithm 2.1 Encoding of Interval Algorithm for Homophonic Coding

Input: $u \in \mathcal{U}$.

1. Set $\boldsymbol{s}$ to be an empty sequence and $I, J := [0, 1)$.
2. $I := I(u, P_U)$ and draw $r \in [0, 1)$ from uniform distribution on I.
3. Search $x \in \mathcal{X}$ such that $r \in J(x, P_X)$ and set $\boldsymbol{s} := (\boldsymbol{s}, x)$.
4. If $J \subset I$ then output $\boldsymbol{s}$ and terminate the algorithm.
5. $J := J(x, P_X)$ and go to step 3.

Algorithm 2.2 Decoding of Interval Algorithm for Homophonic Coding

Input: $\boldsymbol{x} = (x_1, x_2, \cdots, x_n) \in \mathcal{X}^*$.

1. Set $\boldsymbol{s}$ to be an empty sequence.
2. $I, J := [0, 1)$ and $i := 1$.
3. $J := J(x_i, P_X)$.
4. $i := i + 1$ and if $i > n$ then output $\boldsymbol{u}$.
5. If there exists u such that $J \subset I(u, P_U)$ then $\boldsymbol{s} := (\boldsymbol{s}, u)$.
6. Go to step 2.

named *interval algorithm for homophonic coding (IAHC)*. For an interval $I = [a, b) \subset [0, 1)$ and an input symbol $u \in \mathcal{U}$, let define a subinterval of I by

$$I(u, P_U) \equiv \left[a + (b - a) \sum_{\tilde{u}:\tilde{u}<u} P_U(\tilde{u}),\ a + (b - a) \sum_{\tilde{u}:\tilde{u}\leq u} P_U(\tilde{u}) \right). \tag{2.38}$$

The subinterval $J(x, P_X) \subset [0, 1)$ is defined similarly for $J \subset [0, 1)$ and $x \in \mathcal{X}$.

Encoding and decoding of the algorithm IAHC is described in Algorithms 2.1 and 2.2, which are very similar to interval algorithm for random number generation [45]. Although Hoshi and Han [44] just call their algorithm "algorithm", we call their coding scheme with this algorithm *IAHC coding* or *IAHC code* for convenience.

Theorem 2.5 ([44, Theorem 3]). *The IAHC code is P_X-perfect and the average code length $\mathrm{E}_U[L]$ satisfies*

$$\frac{H(P_U)}{H(P_X)} \le \mathrm{E}_U[L] \le \frac{H(P_U)}{H(P_X)} + \frac{\log(2(|\mathcal{X}|-1))}{H(P_X)} + \frac{h(p_{\max})}{(1-p_{\max})H(P_X)}\,, \quad (2.39)$$

where $p_{\max} = \max_{x\in\mathcal{X}} P_X(x)$ and $h(p) = -p\log p - (1-p)\log(1-p)$ is the binary entropy function.

Note that if U in (2.39) is replaced with an i.i.d. sequence U_1^n then $H(P_U)$ becomes $nH(P_U)$. As a result, we have $\mathrm{E}_{U_1^n}[L]/n \to H(P_U)/H(P_X)$ as $n \to \infty$. Note that it is easily seen from Algorithms 2.1 and 2.2 that the complexity increases only linearly in n by this replacement. We see that the IAHC code is asymptotically optimal in view of the following (trivial) theoretical bound on the code length of homophonic coding.

Lemma 2.6. *Under any P_X-perfect fixed-to-variable length homophonic coding scheme, the average code length $\mathrm{E}_U[L]$ satisfies*

$$E_U[L] \ge \frac{H(P_U)}{H(P_X)}\,. \quad (2.40)$$

Proof. Consider a source sequence $(U_1, U_2, \cdots, U_{n^2}) \in \mathcal{U}^{n^2}$ for any $n \in \mathbb{N}$. When we encode this sequence by any lossless code, the expectation of the code length $\tilde{L}$ satisfies

$$\mathrm{E}_{U_1^{n^2}}[\tilde{L}] \ge n^2 H(P_U)\,. \quad (2.41)$$

On the other hand, consider a lossless code described as follows.

1. Encode each $U_i, i = 1, \cdots, n^2$, by a P_X-perfect homophonic coding scheme to $X_1, \cdots, X_{L_{n^2}}$ with length L_{n^2}.
2. If L_{n^2} is not a multiple of n then append an i.i.d. sequence $X_{L_{n^2}+1}$, $X_{L_{n^2}+2}, \cdots, X_{n\lceil L_{n^2}/n\rceil}$ with distribution P_X to the sequence $X_1, \cdots, X_{L_{n^2}}$.
3. Encode $(X_1, \cdots, X_n), (X_{n+1}, \cdots, X_{2n}), \cdots, (X_{(n-1)\lceil L_{n^2}/n\rceil+1}, \cdots, X_{n\lceil L_{n^2}/n\rceil})$ by Huffman code for inputs in $\mathcal{X}^n$.

From the definition of P_X-perfect and the property of Huffman code in (2.23), the average code length of this lossless code satisfies

$$\begin{aligned}\mathrm{E}_{U_1^{n^2}}[\tilde{L}] &< \mathrm{E}_{U_1^{n^2}}[\lceil L_{n^2}/n\rceil](H(P_{X_1^n})+1)\\ &< \mathrm{E}_{U_1^{n^2}}[L_{n^2}/n+1](nH(P_X)+1)\\ &= (n\mathrm{E}_U[L]+1)(nH(P_X)+1) \quad (2.42)\end{aligned}$$

From (2.41) and (2.42) we have

$$\mathrm{E}_U[L] > \frac{H(P_U)}{H(P_X) + \frac{1}{n}} - \frac{1}{n}$$

and we obtain the lemma since n is arbitrary. $\square$

Chapter 3

Polar Codes for Asymmetric Channels and Sources

Recently polar coding is attracting much attention for its achievability of the Shannon bound with polynomial complexity. Polar codes were originally proposed by Arikan [18] for binary memoryless symmetric channels and generalized for Galois fields [57] and arbitrary q-ary alphabets [58]. The idea of polar codes was also extended to lossless and lossy source coding and some multiterminal problems [59].

In this chapter, we consider channel coding with polar codes for asymmetric memoryless channels and lossy source coding for nonuniform sources and/or asymmetric distortion measures. In these asymmetric settings, the optimal symbol distribution of codewords to achieve the Shannon bound is not always uniform.

In known polar coding schemes for asymmetric settings (see [59, Section 4.5.1] [58, Section 3]), codewords with a nonuniform symbol distribution are generated based on Gallager's method discussed in Section 2.3. Then, the complexity increase considerably in the case that the optimal symbol distribution cannot be approximated by simple rational numbers.

To overcome this defect, we need to generate the given symbol distribution $P_X(\cdot)$ of codewords without any extended alphabet. A key idea to generate a desired distribution can be found in the lossless compression by polar codes [59]. In this setting, an original message $X_1^n = (X_1, X_2, \ldots, X_n)$ with nonuniform distribution is transformed to $U_1^n = X_1^n G_n$ by the generator matrix G_n of polar codes. It is shown that the elements of U_1^n polarize into two groups, $\mathcal{F}$ and $\mathcal{F}^c$. For each $i \in \mathcal{F}$, U_i is almost uniformly distributed and independent of the leading sequence $U_1^{i-1} = (U_1, U_2, \ldots, U_{i-1})$ and, for each

$i \in \mathcal{F}^c$, U_i is determined from U_1^{i-1} almost surely.

Now we apply this technique to channel coding. The result on the lossless coding implies that when we have a uniform source, we can obtain a nonuniform input for a given channel in the following way: (a) choose a value of U_i uniformly for each $i \in \mathcal{F}$, (b) determine U_i for each $i \in \mathcal{F}^c$ appropriately from U_1^{i-1} and (c) transform U_1^n to X_1^n by $X_1^n = U_1^n G_n^{-1} = U_1^n G_n$. In the case of channel coding with channel input X and channel output Y, U_i for each $i \in \mathcal{F}$ polarizes further into $\mathcal{I} \subset \mathcal{F}$ and $\mathcal{F} \setminus \mathcal{I}$, where this polarization corresponds to lossless source coding with side information [60]. Here $\mathcal{I}$ is the set of indices i such that U_i is almost independent of U_1^{i-1} but can be determined uniquely from channel output Y_1^n almost surely. $\mathcal{F} \setminus \mathcal{I}$ is the set of indices i such that U_i is almost independent of both U_1^{i-1} and Y_1^n. By assigning a message to random variables U_i for $i \in \mathcal{I}$, we can send it with small decoding error probability, that is, $\mathcal{I} \subset \mathcal{F}$ can be used as an information set.

This idea can also be applied to lossy source coding. In this case Y_1^n and X_1^n correspond to a source sequence and a reproduction sequence, respectively. By sending U_i only for $i \in \mathcal{I}$, we can recover X_1^n within a given distortion from Y_1^n.

Note that recently Sutter et al. [41] have considered a channel coding scheme for asymmetric settings based on lossless coding independently of our work. However, their scheme uses a concatenated code of two polar codes and the code construction is not simple. Furthermore, the decoding error probability of their scheme is approximately $\mathrm{O}(2^{-n^{1/4}})$ since their scheme requires polarization of both the inner code and the outer code, whereas we show that the decoding error probability of our scheme is approximately $\mathrm{O}(2^{-n^{1/2}})$ by using a single polar code.

We start with an introduction of polar codes proposed by Arikan in Section 3.1. In Section 3.2, we generalize the polarization phenomenon to general random variables as in [60] and discuss its relation with the polarization of symmetric channels considered in Section 3.1. The main results are given in Sections 3.3 and 3.4. We propose a new polar coding scheme for asymmetric channels and show that it can achieve the channel capacity asymptotically in Section 3.3. In Section 3.4, we propose an asymptotically optimal polar coding scheme for lossy source coding with nonuniform sources and/or asymmetric distortion measures. We give proofs of the optimality of the proposed schemes in Section 3.5.

3.1 Polarization of Symmetric Channels

We review results of Arikan [18] on channel polarization in this section. Let $W : \mathcal{X} \to \mathcal{Y}$ be a memoryless channel. We assume that $\mathcal{X} = \{0, 1\}$, that is, W is a binary-input discrete memoryless channel, or a B-DMC in short. See [57][58] for extension to a nonbinary input alphabet $\mathcal{X}$. When we consider polarization of a B-DMC, symmetric capacity $I(W)$ and Bhattacharyya parameter $Z_{\mathrm{B}}(W)$ are the main targets of the analysis. Symmetric capacity is defined by

$$\begin{aligned} I(W) &\equiv \sum_{x\in\mathcal{X},\, y\in\mathcal{Y}} \frac{1}{2} W(y|x) \log \frac{W(y|x)}{\frac{1}{2}(W(y|0) + W(y|1))} \\ &= I(X;Y)\,, \end{aligned} \tag{3.1}$$

where X is uniformly distributed on $\mathcal{X} = \{0, 1\}$ and Y is the random variable obtained by transmitting X via the channel W. $I(W)$ coincides with the channel capacity $C(W)$ if the channel W is symmetric. Bhattacharyya parameter $Z_{\mathrm{B}}(W)$ is defined by

$$Z_{\mathrm{B}}(W) \equiv \sum_{y} \sqrt{W(y|0)W(y|1)}\,. \tag{3.2}$$

$I(W)$ and $Z_{\mathrm{B}}(W)$ are related to each other by the following lemma.

Lemma 3.1 ([18, Proposition 1])**.** *For any D-BMC, we have*

$$\begin{aligned} I(W) &\geq \log \frac{2}{1 + Z_{\mathrm{B}}(W)}\,, \\ I(W) &\leq \sqrt{1 - Z_{\mathrm{B}}(W)}\,. \end{aligned} \tag{3.3}$$

From this lemma we see that $Z_{\mathrm{B}}(W) \to 1$ implies $I(W) \to 0$ and $Z_{\mathrm{B}}(W) \to 0$ implies $I(W) \to 1$.

We start with channel $W_2 : \{0, 1\}^2 \to \mathcal{Y}^2$ in Fig. 3.1 for understanding of channel polarization. In channel W_2, a pair of independent and uniformly distributed random variables (U_1, U_2) is first transformed to $(X_1, X_2) = (U_1 \oplus U_2, U_2)$ and (X_1, X_2) is transmitted by W^2, where $\oplus$ denotes the addition on GF(2). Then the transition probability of W_2 is given by

$$W_2(y_1, y_2|u_1, u_2) = W^2(y_1, y_2|u_1 \oplus u_2, u_2)\,. \tag{3.4}$$

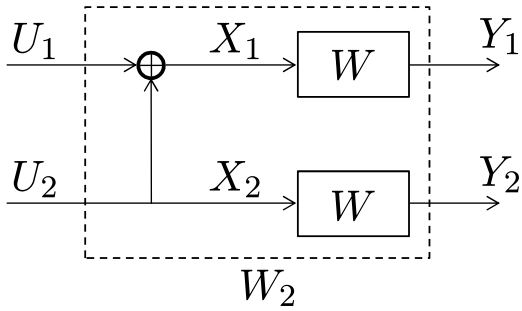

Fig. 3.1. A representation of W_2.

In this channel, we can decompose the mutual information $I(U_1^2; Y_1^2) = I(X_1^2; Y_1^2) = 2I(W)$ by

$$\begin{aligned} I(U_1^2; Y_1^2) &= I(U_1; Y_1^2) + I(U_2; Y_1^2 | U_1) \\ &= I(U_1; Y_1^2) + I(U_2; U_1, Y_1^2)\,. \end{aligned} \tag{3.5}$$

Since

$$\begin{aligned} I(U_2; U_1, Y_1^2) &= H(X_2) - H(X_2 | U_1, Y_1^2) \\ &\geq H(X_2) - H(X_2 | Y_2) \\ &= I(W)\,, \end{aligned} \tag{3.6}$$

we have

$$I(U_1; Y_1^2) \leq I(W) \leq I(U_2; U_1, Y_1^2)\,, \tag{3.7}$$

that is, $2I(W)$ can be decomposed to a larger one and a smaller one. We can also consider a decomposition of W_2 into $W_2^{(1)} : \mathcal{X} \to \mathcal{Y}^2$ and $W_2^{(2)} : \mathcal{X} \to \mathcal{X} \times \mathcal{Y}^2$ given by

$$W_2^{(1)}(y_1^2 | u_1) = P_{Y_1^2 | U_1}(y_1^2 | u_1) \tag{3.8}$$

and

$$W_2^{(2)}(u_1, y_1^2 | u_2) = P_{U_1, Y_1^2 | U_2}(u_1, y_1^2 | u_2)\,. \tag{3.9}$$

Then we have $I(U_1; Y_1^2) = I(W_2^{(1)})$ and $I(U_2; U_1, Y_1^2) = I(W_2^{(2)})$ and (3.7) is equivalent to

$$I(W_2^{(1)}) \leq I(W) \leq I(W_2^{(2)})\,. \tag{3.10}$$

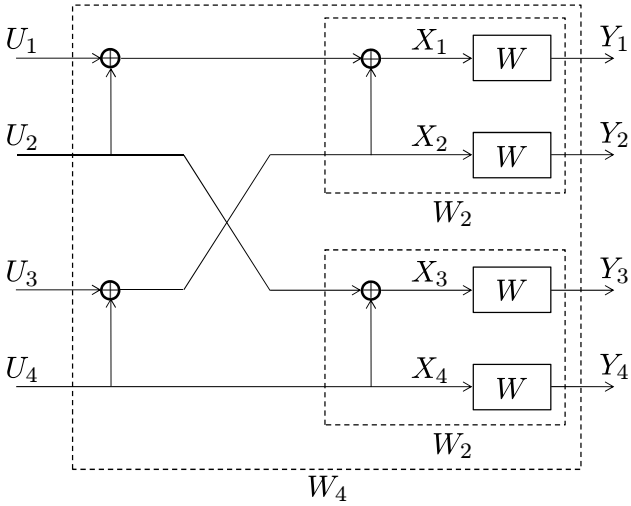

Fig. 3.2. A representation of W_4.

Next we consider channel $W_4 : \{0,1\}^4 \to \mathcal{Y}^4$ in Fig. 3.2 which can be obtained from two copies of W_2 by combining each pair of equivalent bits with addition $\oplus$. For decomposition of channel W_4 given by

$$W_4^{(i)}(u_1^{i-1}, y_1^4|u_i) = P_{U_1^{i-1},Y_1^4|U_i}(u_1^{i-1}, y_1^4|u_i)\,, \qquad i = 1,2,3,4\,, \quad (3.11)$$

symmetric capacities $I(W_2^{(1)})$ and $I(W_2^{(2)})$ are further decomposed to

$$\begin{aligned} &I(W_4^{(1)}) \le I(W_2^{(1)}) \le I(W_4^{(2)}), \quad I(W_4^{(1)}) + I(W_4^{(2)}) = 2I(W_2^{(1)})\,, \\ &I(W_4^{(3)}) \le I(W_2^{(2)}) \le I(W_4^{(4)}), \quad I(W_4^{(3)}) + I(W_4^{(4)}) = 2I(W_2^{(2)})\,. \end{aligned} \quad (3.12)$$

Now we formulate the procedure to construct channel $W_n : \mathcal{X}^n \to \mathcal{Y}^n$ for $n = 2^k$. Let $W_1 \equiv W$ and consider channel W_n, $n = 2^k \ge 2$, constructed from $W_{n/2}$ by

$$W_n(y_1^n|u_1^n) = W_{n/2}(y_1^{n/2}|u_{1,\mathrm{o}}^n \oplus u_{1,\mathrm{e}}^n)W(y_{n/2+1}^n|u_{1,\mathrm{e}}^n)\,, \quad (3.13)$$

where $u_{1,\mathrm{o}}^n$ and $u_{1,\mathrm{e}}^n$ denote the subsequences of u_1^n with even and odd indices, respectively. Applying (3.13) recursively, we can see that W_n can be expressed as

$$W_n(y_1^n|u_1^n) = W^n(y_1^n|u_1^n G_n) \quad (3.14)$$

for a matrix G_n. Closer inspection reveals that G_n can be expressed as $G_n = F^{\otimes k} B_n$ where $F = \left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$, $\otimes$ denotes the Kronecker power and B_n is a permutation matrix called *bit reversal matrix* (see [18, Proposition 16] for detail).

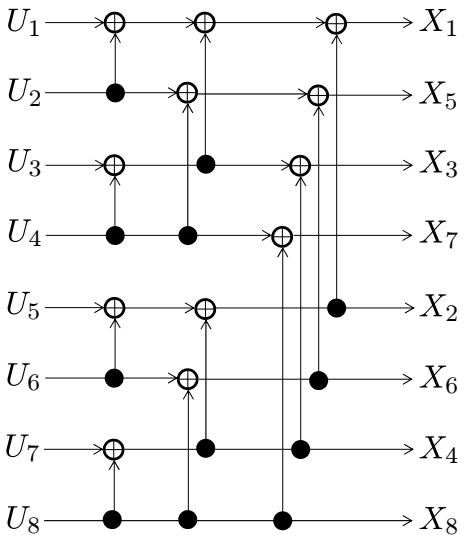

Fig. 3.3. X_1^n defined by $U_1^n F^{\otimes k} B_n$.

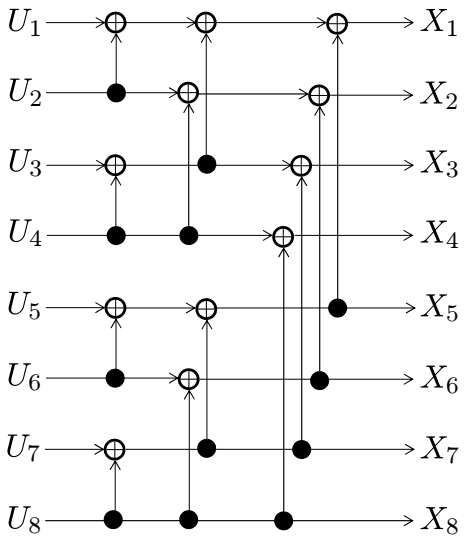

Fig. 3.4. X_1^n defined by $U_1^n F^{\otimes k}$.

Let $U_1, \cdots, U_n$ be uniform random variables on $\mathcal{X}$ independent of each other and define $X_1^n \equiv U_1^n G_n$. We call G_n a generator matrix of polar codes[*1]. Y_1^n is defined as the output of channel W^n for input X_1^n. Then we can express $W_n(y_1^n|u_1^n) = P_{Y_1^n|U_1^n}(y_1^n|u_1^n)$ and $W^n(y_1^n|x_1^n) = P_{Y_1^n|X_1^n}(y_1^n|x_1^n)$. Note that X_1^n and Y_1^n are always treated as sets of n random variables and permutation of them does not affect the subsequent results. Then we redefine $G_n \equiv F^{\otimes k}$ and $W_n(y_1^n|u_1^n) \equiv W^n(y_1^n|F^{\otimes k}u_1^n)$ excluding the permutation B_n as in most literatures for simple representation and implementation of polar codes. Under the definition $W_n^{(i)}(y_1^n|u_1^n) = W^n(y_1^n|F^{\otimes k}u_1^n)$, the recursive formula for W_n becomes

$$W_n(y_1^n|u_1^n) = W_{n/2}(y_{1,\mathrm{o}}^n|u_{1,\mathrm{o}}^n \oplus u_{1,\mathrm{e}}^n)W(y_{1,\mathrm{e}}^n|u_{1,\mathrm{e}}^n)\,, \tag{3.15}$$

instead of (3.13). Figs. 3.3 and 3.4 illustrate the difference between these two definitions of G_n for $n = 2^3 = 8$. X_1^n defined by $U_1^n F^{\otimes k} B_n$ and $U_1^n F^{\otimes k}$ are given in the left and the right figures, respectively.

The subchannel $W_n^{(i)} : \mathcal{X} \to \mathcal{X}^{i-1} \times \mathcal{Y}^n$ is defined by

$$W_n^{(i)}(u_1^{i-1}, y_1^n|u_i) \equiv P_{U_1^{i-1},Y_1^n|U_i}(u_1^{i-1}, y_1^n|u_i). \tag{3.16}$$

Then the recursive formula for the subchannels is expressed for $i =$

[*1] The generator matrix of a polar code is a submatrix of G_n. But for simplicity we call G_n the generator matrix of polar codes in this thesis.

$1, 2, \cdots, n/2$ as

$$\begin{aligned}
&W_n^{(2i-1)}(u_1^{2i-2}, y_1^n|u_{2i-1}) \\
&\quad = \sum_{u_{2i}} \frac{1}{2} W_{n/2}^{(i)}(y_{1,\mathrm{o}}^n, u_{1,\mathrm{o}}^{2i-2} \oplus u_{1,\mathrm{e}}^{2i-2}|u_{2i-1} \oplus u_{2i}) W_{n/2}^{(i)}(y_{1,\mathrm{e}}^n, u_{1,\mathrm{e}}^{2i-2}|u_{2i}) \,, \\
&W_n^{(2i)}(u_1^{2i-1}, y_1^n|u_{2i}) \\
&\quad = \frac{1}{2} W_{n/2}^{(i)}(y_{1,\mathrm{o}}^n, u_{1,\mathrm{o}}^{2i-2} \oplus u_{1,\mathrm{e}}^{2i-2}|u_{2i-1} \oplus u_{2i}) W_{n/2}^{(i)}(y_{1,\mathrm{e}}^n, u_{1,\mathrm{e}}^{2i-2}|u_{2i}) \,. \qquad (3.17)
\end{aligned}$$

As implied from (3.10) and (3.12), the symmetric capacity $I(W_n^{(i)})$ divides into larger one and smaller one as n increases. We can show that almost all subchannels converges to almost noiseless one or almost noisy one, that is, the subchannels $\{W_n^{(i)}\}_i$ *polarize* into two groups.

Theorem 3.2 ([59, Theorems 2.11 and 3.15])**.** *For any B-DMC W and $\beta < 1/2$, we have*

$$\lim_{n\to\infty} \frac{1}{n} \left| \left\{ i : Z_{\mathrm{B}}(W_i^{(n)}) \le 2^{-n^\beta} \right\} \right| = I(W) \,, \qquad (3.18)$$

$$\lim_{n\to\infty} \frac{1}{n} \left| \left\{ i : Z_{\mathrm{B}}(W_i^{(n)}) \ge 1 - 2^{-n^\beta} \right\} \right| = 1 - I(W) \,. \qquad (3.19)$$

Recall that $Z_{\mathrm{B}}(W) \to 1$ implies $I(W) \to 0$ and $Z_{\mathrm{B}}(W) \to 0$ implies $I(W) \to 1$ from Lemma 3.1. Then we see that $I(W_n^{(i)})$ approaches one for $nI(W)$ of i's and approaches 1 for $n(1 - I(W))$ of i's.

3.2 Polarization of Nonuniform Random Variables

In the previous section, we considered polarization of channel W with a uniformly distributed input. In this section we consider polarization for general random variables $X \in \mathcal{X} = \{0, 1\}$ and $Y \in \mathcal{Y}$ where X is not necessarily uniformly distributed. This type of polarization is first considered for lossless compression of source X with side information Y in [60], but we give further analysis for this problem so that the result can be applied to lossy source coding and channel coding.

Let $X_1^n = (X_1, X_2, \ldots, X_n)$ and $Y_1^n = (Y_1, Y_2, \ldots, Y_n)$ denote i.i.d. copies of X and Y, respectively. U_1^n is defined as $U_1^n \equiv X_1^n G_n^{-1} = X_1^n G_n$ for $G_n = F^{\otimes k}$. For a set $\mathcal{A} \subset \{1, 2, \cdots, n\}$, we define subvector of U_1^n with indices in $\mathcal{A}$ by $U_{\mathcal{A}} \equiv \{U_i\}_{i \in \mathcal{A}}$.

When we consider polarization of random variables, it is convenient to

consider a parameter $Z(X|Y)$ defined as

$$\begin{aligned} Z(X|Y) &\equiv 2\sum_y P_Y(y)\sqrt{P_{X|Y}(0|y)P_{X|Y}(1|y)} \\ &= 2\sum_y \sqrt{P_{X,Y}(0,y)P_{X,Y}(1,y)}\,. \end{aligned} \tag{3.20}$$

rather than Bhattacharyya parameter $Z_{\mathrm{B}}(W)$ in (3.2). Note that $Z(X|Y)$ coincides with the Bhattacharyya parameter $Z_{\mathrm{B}}(P_{Y|X})$ when X is uniformly distributed. The parameter $Z(X|Y)$ is related to conditional entropy $H(X|Y)$ by the following lemma.

Lemma 3.3 ([60, Proposition 2])**.**

$$(Z(X|Y))^2 \le H(X|Y)\,, \tag{3.21}$$

$$H(X|Y) \le \log(1+Z(X|Y)) \le Z(X|Y)\,. \tag{3.22}$$

Now we give the main result of this section on the polarization for nonuniform random variables.

Theorem 3.4. *For any $\beta < 1/2$, i.i.d. random variables (X_1^n, Y_1^n) and $U_1^n = X_1^n G_n$,*

$$\lim_{n\to\infty} \frac{|\{i : Z(U_i|U_1^{i-1}, Y_1^n) \le 2^{-n^\beta},\ Z(U_i|U_1^{i-1}) \ge 1-2^{-n^\beta}\}|}{n} = I(X;Y)\,, \tag{3.23}$$

and

$$\lim_{n\to\infty} \frac{|\{i : Z(U_i|U_1^{i-1}, Y_1^n) \ge 1-2^{-n^\beta} \text{ or } Z(U_i|U_1^{i-1}) \le 2^{-n^\beta}\}|}{n} = 1 - I(X;Y)\,. \tag{3.24}$$

Before proving this theorem, we discuss how to prove the asymptotic optimality of polar codes for lossless coding. In [60], a recursive formula is derived for $Z(U_i|U_1^{i-1}, Y_1^n)$ and the polarization of $Z(U_i|U_1^{i-1}, Y_1^n)$ is shown from the fact that the formula has the same form as the symmetric case in (3.17). On the other hand, in [59], the asymptotic optimality is derived by reducing the source coding problem to a channel coding one. In this thesis we apply this reduction technique to parameter $Z(U_i|U_1^{i-1}, Y_1^n)$ and show that $Z(U_i|U_1^{i-1}, Y_1^n)$ is equal to a Bhattacharyya parameter $Z_{\mathrm{B}}(\tilde{W}_n^{(i)})$ for some symmetric channel $\tilde{W}_n^{(i)}$. By this representation we can apply known results on the symmetric settings directly to our asymmetric settings.

Now we construct the symmetric channel $\tilde{W}_n^{(i)}$ satisfying the above property. Let $\tilde{X}$ be a uniform random variable on $\{0,1\}$ independent of (X,Y) and $\tilde{Y}$ be defined as $\tilde{Y} = (\tilde{X} \oplus X, Y) \in \tilde{\mathcal{Y}} = \{0,1\} \times \mathcal{Y}$. We define a symmetric channel $\tilde{W} : \{0,1\} \to \tilde{\mathcal{Y}}$ by

$$\tilde{W}(\tilde{y}|\tilde{x}) = P_{\tilde{Y}|\tilde{X}}(\tilde{y}|\tilde{x}) \,. \tag{3.25}$$

Let $(\tilde{X}_1^n, \tilde{Y}_1^n)$ be vectors of i.i.d. copies (X,Y) and define $\tilde{U}_1^n = \tilde{X}_1^n G_n^{-1} = \tilde{X}_1^n G_n$. Then, when we substitute the symmetric channel $\tilde{W}$ into W in the previous section, the i-th subchannel is expressed as

$$\tilde{W}_i^{(n)}(\tilde{u}_1^{i-1}, \tilde{y}_1^n|\tilde{u}_i) = P_{\tilde{U}_1^{i-1}, \tilde{Y}_1^n|\tilde{U}_i}(\tilde{u}_1^{i-1}, \tilde{y}_1^n|\tilde{u}_i) \,. \tag{3.26}$$

The following theorem enables us to apply known results on symmetric channels including Theorem 3.2 to our asymmetric setting.

Theorem 3.5. *For $\tilde{W}_i^{(n)}$ defined by* (3.26),

$$P_{U_1^i, Y_1^n}(u_1^i, y_1^n) = 2^{n-1} \tilde{W}_i^{(n)}(u_1^{i-1}, (0^n, y_1^n)|u) \tag{3.27}$$

and

$$Z(U_i|U_1^{i-1}, Y_1^n) = Z_{\mathrm{B}}(\tilde{W}_i^{(n)}) \,. \tag{3.28}$$

Proof. Denote a member of $\tilde{\mathcal{Y}}^n = \{0,1\}^n \times \mathcal{Y}$ by $\tilde{y}_1^n = (\tilde{z}_1^n, y_1^n)$. Then we have

$$\begin{aligned}
&\tilde{W}_i^{(n)}(\tilde{u}_1^{i-1}, \tilde{y}_1^n|\tilde{u}_i) \\
&\quad = P_{\tilde{U}_1^{i-1}, \tilde{Y}_1^n|\tilde{U}_i}(\tilde{u}_1^{i-1}, \tilde{y}_1^n|\tilde{u}_i) \\
&\quad = P_{\tilde{X}_1^n \oplus X_1^n, Y_1^n, \tilde{U}_1^{i-1}|\tilde{U}_i}(\tilde{z}_1^n, y_1^n, \tilde{u}_1^{i-1}|\tilde{u}_i) \\
&\quad = \sum_{x_1^n} P_{X_1^n, Y_1^n, \tilde{X}_1^n, \tilde{U}_1^{i-1}|\tilde{U}_i}(x_1^n, y_1^n, \tilde{z}_1^n \oplus x_1^n, \tilde{u}_1^{i-1}|\tilde{u}_i) \\
&\quad \overset{\text{(a)}}{=} \sum_{x_1^n} P_{X_1^n, Y_1^n}(x_1^n, y_1^n) P_{\tilde{X}_1^n, \tilde{U}_1^{i-1}|\tilde{U}_i}(\tilde{z}_1^n \oplus x_1^n, \tilde{u}_1^{i-1}|\tilde{u}_i) \\
&\quad \overset{\text{(b)}}{=} 2 \sum_{x_1^n} P_{X_1^n, Y_1^n}(x_1^n, y_1^n) P_{\tilde{X}_1^n}(\tilde{z}_1^n \oplus x_1^n) \mathbb{1}\big[((\tilde{z}_1^n \oplus x_1^n) G_n)_1^i = \tilde{u}_1^i\big] \\
&\quad \overset{\text{(c)}}{=} 2^{-n+1} \sum_{x_1^n} P_{X_1^n, Y_1^n}(x_1^n, y_1^n) \mathbb{1}\big[(x_1^n G_n)_1^i = \tilde{u}_1^i \oplus (\tilde{z}_1^n G_n)_1^i\big] \\
&\quad \overset{\text{(d)}}{=} 2^{-n+1} P_{U_1^i, Y_1^n}(\tilde{u}_1^i \oplus (\tilde{z}_1^n G_n)_1^i, y_1^n) \,,
\end{aligned} \tag{3.29}$$

where $\mathbb{1}[\cdot]$ denotes the indicator function and the equalities follow from

(a): $(\tilde{X}_1^n, \tilde{U}_1^n)$ is independent of (X_1^n, Y_1^n),
(b): $P_{\tilde{X}_1^n, \tilde{U}_1^{i-1}|\tilde{U}_i} = P_{\tilde{X}_1^n} P_{\tilde{U}_1^i|\tilde{X}_1^n} / P_{\tilde{U}_i}$, $\tilde{U}_1^n = \tilde{X}_1^n G_n$ and $P_{\tilde{U}_i}(\tilde{u}_i) = 1/2$.
(c): $\tilde{X}_1^n$ is uniformly distributed over $\{0,1\}^n$,
(d): $U_1^n = X_1^n G_n$.

We obtain (3.27) by letting $\tilde{z}_1^n = 0_1^n$.

Now we prove (3.28). From the definition of Z_{B} and (3.29) we have

$$\begin{aligned}
Z_{\mathrm{B}}(\tilde{W}_i^{(n)}) &= \sum_{\tilde{y}_1^n, \tilde{u}_1^{i-1}} \sqrt{\tilde{W}_i^{(n)}(\tilde{u}_1^{i-1}, \tilde{y}_1^n|0) \cdot \tilde{W}_i^{(n)}(\tilde{u}_1^{i-1}, \tilde{y}_1^n|1)} \\
&= \sum_{\tilde{z}_1^n, y_1^n, \tilde{u}_1^{i-1}} \sqrt{2^{-n+1} P_{U_1^i, Y_1^n}((\tilde{u}_1^{i-1}, 0) \oplus (\tilde{z}_1^n G_n)_1^i, y_1^n)} \\
&\qquad \cdot \sqrt{2^{-n+1} P_{U_1^i, Y_1^n}((\tilde{u}_1^{i-1}, 1) \oplus (\tilde{z}_1^n G_n)_1^i, y_1^n)} \\
&= 2^{-n+1} \sum_{\tilde{z}_1^n, y_1^n, \tilde{u}_1^{i-1}} \sqrt{P_{U_1^i, Y_1^n}((\tilde{u}_1^{i-1} \oplus (\tilde{z}_1^n G_n)_1^{i-1}, 0), y_1^n)} \\
&\qquad \cdot \sqrt{P_{U_1^i, Y_1^n}((\tilde{u}_1^{i-1} \oplus (\tilde{z}_1^n G_n)_1^{i-1}, 1), y_1^n)}\,.
\end{aligned} \tag{3.30}$$

Let $z_1^n \equiv \tilde{z}_1^n G_n$ and $u_1^{i-1} \equiv \tilde{u}_1^{i-1} \oplus (\tilde{z}_1^n G_n)_1^{i-1} = \tilde{u}_1^{i-1} \oplus z_1^{i-1}$. Since $(\tilde{z}_1^n, \tilde{u}_1^{i-1}) \mapsto (z_1^n, u_1^{i-1})$ is a bijection on $\{0,1\}^n \times \{0,1\}^{i-1}$, it holds that

$$\begin{aligned}
Z_{\mathrm{B}}(\tilde{W}_i^{(n)}) &= 2^{-n+1} \sum_{z_1^n, y_1^n, u_1^{i-1}} \sqrt{P_{U_1^i, Y_1^n}((u_1^{i-1}, 0), y_1^n) \cdot P_{U_1^i, Y_1^n}((u_1^{i-1}, 1), y_1^n)} \\
&= 2 \sum_{y_1^n, u_1^{i-1}} \sqrt{P_{U_1^i, Y_1^n}((u_1^{i-1}, 0), y_1^n) \cdot P_{U_1^i, Y_1^n}((u_1^{i-1}, 1), y_1^n)} \\
&= Z(U_i | U_1^{i-1}, Y_1^n)
\end{aligned} \tag{3.31}$$

and the proof is completed. □

Now we prove Theorem 3.4 from Theorems 3.2 and 3.5.

Proof of Theorem 3.4. First we have

$$\begin{aligned}
I(\tilde{W}) &= I(\tilde{X}; \tilde{X} \oplus X, Y) \\
&= H(\tilde{X} \oplus X, Y) - H(\tilde{X} \oplus X, Y | \tilde{X}) \\
&= H(\tilde{X} \oplus X | Y) + H(Y) - H(X, Y | \tilde{X}) \\
&\overset{\text{(a)}}{=} H(\tilde{X}) + H(Y) - H(X, Y) \\
&\overset{\text{(b)}}{=} 1 - H(X|Y)\,,
\end{aligned} \tag{3.32}$$

where (a) and (b) hold since $\tilde{X}$ is uniformly distributed and independent of (X, Y). Combining (3.32) with Theorem 3.5 and Theorem 3.2 we have

$$\lim_{n\to\infty} \frac{1}{n}\left|\left\{i : Z(U_i|U_1^{i-1}, Y_1^n) \le 2^{-n^\beta}\right\}\right| = 1 - H(X|Y)\,, \tag{3.33}$$

$$\lim_{n\to\infty} \frac{1}{n}\left|\left\{i : Z(U_i|U_1^{i-1}, Y_1^n) \ge 1 - 2^{-n^\beta}\right\}\right| = H(X|Y)\,. \tag{3.34}$$

Next consider the case that Y is a random variable which takes a fixed value with probability 1. For this case $Z(U_i|U_1^{i-1}, Y_1^n) = Z(U_i|U_1^{i-1})$ and $H(X|Y) = H(X)$. Thus, (3.33) and (3.34) become

$$\lim_{n\to\infty} \frac{1}{n}\left|\left\{i : Z(U_i|U_1^{i-1}) \le 2^{-n^\beta}\right\}\right| = 1 - H(X)\,, \tag{3.35}$$

$$\lim_{n\to\infty} \frac{1}{n}\left|\left\{i : Z(U_i|U_1^{i-1}) \ge 1 - 2^{-n^\beta}\right\}\right| = H(X)\,. \tag{3.36}$$

Define sets

$$\begin{aligned} A &\equiv \{i : Z(U_i|U_1^{i-1}, Y_1^n) \le 2^{-n^\beta}\}\,, \\ B &\equiv \{i : Z(U_i|U_1^{i-1}, Y_1^n) \ge 1 - 2^{-n^\beta}\}\,, \\ C &\equiv \{i : Z(U_i|U_1^{i-1}) \le 2^{-n^\beta}\}\,, \\ D &\equiv \{i : Z(U_i|U_1^{i-1}) \ge 1 - 2^{-n^\beta}\}\,. \end{aligned} \tag{3.37}$$

It is easy to see that $B \cap C$ is empty for sufficiently large n from Lemma 3.3 and $H(U_i|U_1^{i-1}, Y_1^n) \le H(U_i|U_1^{i-1})$. Furthermore, we also note from (3.33)–(3.36) that

$$\lim_{n\to\infty} \frac{|A| + |B|}{n} = \lim_{n\to\infty} \frac{|C| + |D|}{n} = 1\,. \tag{3.38}$$

Hence (3.23) and (3.24) hold because

$$\lim_{n\to\infty} \frac{|B \cup C|}{n} = \lim_{n\to\infty} \frac{|B| + |C|}{n} = 1 - I(X;Y) \tag{3.39}$$

and

$$\lim_{n\to\infty} \frac{|A \cap D|}{n} = 1 - \lim_{n\to\infty} \frac{|B \cup C|}{n} = I(X;Y)\,. \tag{3.40}$$

□

3.3 Polar Codes for Asymmetric Channels

Let X and Y denote the random variables that achieve the channel capacity in (2.2). We assume without loss of generality that $0 < P_X(0) < 1$. In this section, we propose a new polar coding scheme which can achieve the capacity $C(W) = I(X;Y)$ for B-DMC W which is not necessarily symmetric.

3.3.1 Code Construction

Assume that an information set $\mathcal{I} \subset \{1, 2, \ldots, n\}$ and a frozen set $\mathcal{I}^c = \{1, 2, \ldots, n\} \setminus \mathcal{I}$ are fixed for a given cannel W. We use bits $u_{\mathcal{I}} = \{u_i\}_{i \in \mathcal{I}}$ to send a message.

In the case of symmetric channels, the values of frozen bits $u_{\mathcal{I}^c}$ are chosen randomly with the uniform distribution on $\{0, 1\}$ in the code construction but they are fixed when the code is used. In our scheme, the frozen bits $u_{\mathcal{I}^c}$ are deterministic but dependent on the value of previous bits u_1^{i-1}. Furthermore, unlike the symmetric case, we choose the value of u_i given u_1^{i-1} *not uniformly* in the code construction.

Let $\mathcal{L}_i$ be the family of functions $\lambda_i : \{0, 1\}^{i-1} \to \{0, 1\}$. Now we consider a polar code with frozen set $\mathcal{I}^c$ and maps $\lambda_{\mathcal{I}^c} \equiv \{\lambda_i\}_{i \in \mathcal{I}^c}$. The maps $\lambda_{\mathcal{I}^c}$ are used to determine the frozen bits and are shared between the encoder and the decoder.

Let $M_1^{|\mathcal{I}|}$ denote a message uniformly distributed on $\{0, 1\}^{|\mathcal{I}|}$. The encoder determines a codeword from a realization $m_1^{|\mathcal{I}|}$ of $M_1^{|\mathcal{I}|}$ in the following way. First, the encoder determines the information bits by $u_{\mathcal{I}} = m_1^{|\mathcal{I}|}$. Next, for the frozen bits $\mathcal{I}^c$, the encoder determines the value u_i, $i \in \mathcal{I}^c$, in the ascending order by $u_i = \lambda_i(u_1^{i-1})$. We represent the resulting sequence of u_i by $u_1^n(m_1^{|\mathcal{I}|}, \lambda_{\mathcal{I}^c})$. Third, the encoder sends the codeword $x_1^n = u_1^n G_n = u_1^n(m_1^{|\mathcal{I}|}, \lambda_{\mathcal{I}^c})\, G_n$ with code length n. Thus the coding rate is given by $R = |\mathcal{I}|/n$.

The decoder receives a sequence y_1^n according to the channel transition probability $W^n(y_1^n|x_1^n)$. The decoder estimates u_1^n by $\hat{u}_1^n = \hat{u}_1^n(y_1^n, \lambda_{\mathcal{I}^c})$ as follow:

$$\hat{u}_i = \begin{cases} \operatorname*{argmax}_u P_{U_i|U_1^{i-1},Y_1^n}(u|\hat{u}_1^{i-1}, y_1^n) & i \in \mathcal{I}, \\ \lambda_i(\hat{u}_1^{i-1}) & i \in \mathcal{I}^c. \end{cases} \tag{3.41}$$

The decoding is successful if $\hat{u}_{\mathcal{I}} = u_{\mathcal{I}}$ which means $\hat{u}_1^n = u_1^n$. The average decoding error probability over the uniform message $M_1^{|\mathcal{I}|}$ is denoted by $P_{\mathrm{e}}(\lambda_{\mathcal{I}^c})$.

Now consider the choice of the map $\lambda_{\mathcal{I}^c}$. Let $\Lambda_{\mathcal{I}^c} \equiv \{\Lambda_i \in \mathcal{L}_i\}_{i \in \mathcal{I}^c}$ be random variables which are independent of each other and of (X_1^n, Y_1^n), and satisfy

$$P_{\Lambda_i}[\Lambda_i(u_1^{i-1}) = 1] = P_{U_i|U_1^{i-1}}(1|u_1^{i-1}) \tag{3.42}$$

for all $u_1^{i-1} \in \{0,1\}^{i-1}$. Practically, we can realize this randomized map by using pseudo random numbers shared between the encoder and the decoder as follows.

$$u_i = \begin{cases} 0 & \text{with probability } P_{U_i|U_1^{i-1}}(0|u_1^{i-1}), \\ 1 & \text{with probability } P_{U_i|U_1^{i-1}}(1|u_1^{i-1}). \end{cases} \tag{3.43}$$

The idea of this randomized algorithm comes from the polar coding for lossy compression for symmetric sources [59]. As in the case of the lossy coding for symmetric sources, the randomization makes the theoretical analysis much easier in our setting.

From Theorem 3.4 there exists a subset $\mathcal{I}$ of $\{1, \cdots, n\}$ such that $|\mathcal{I}| = nR$,

$$Z(U_i|U_1^{i-1}, Y_1^n) \le 2^{-n^\beta} \quad \text{and} \quad Z(U_i|U_1^{i-1}) \ge 1 - 2^{-n^\beta} \tag{3.44}$$

for all $i \in \mathcal{I}$ if $R < I(X;Y)$, $\beta < 1/2$, and n is sufficiently large. For this $\mathcal{I}$ the following theorem holds.

Theorem 3.6. *Let $M_1^{|\mathcal{I}|}$ be a message chosen uniformly from $\{0,1\}^{|\mathcal{I}|}$ and $\mathcal{I} \subset \{1, \cdots, n\}$ be a set satisfying* (3.44)*. Then the expectation of the decoding error probability over the maps $\Lambda_{\mathcal{I}^c}$ satisfies $\mathrm{E}_{\Lambda_{\mathcal{I}^c}}[P_{\mathrm{e}}(\Lambda_{\mathcal{I}^c})] = \mathrm{O}(2^{-n^{\beta'}})$ for any $\beta' < \beta < 1/2$. Consequently, there exists a deterministic map $\lambda_{\mathcal{I}^c} = \{\lambda_i \in \mathcal{L}_i\}_{i \in \mathcal{I}^c}$ such that $P_{\mathrm{e}}(\lambda_{\mathcal{I}^c}) = \mathrm{O}(2^{-n^{\beta'}})$.*

The proof of this theorem is given in Section 3.5.

3.3.2 Implementation

In the construction of the proposed coding scheme, information set $\mathcal{I}$ has to be chosen from $\{1, \ldots, n\}$ so that $Z(U_i|U_1^{i-1}, Y_1^n)$ is small and $Z(U_i|U_1^{i-1})$ is large for every $i \in \mathcal{I}$. From Theorem 3.5 these parameters can be represented as Bhattacharyya parameters for symmetric channels and the approximation technique in [61] for symmetric cases can be applied.

In the encoding of the proposed polar coding scheme, probability $P_{U_i|U_1^{i-1}}(u|u_1^{i-1})$ in (3.43) has to be computed. Similarly, in the decoding, we need to compute $P_{U_i|U_1^{i-1},Y_1^n}(u|\hat{u}_1^{i-1}, y_1^n)$ in (3.41) and $P_{U_i|U_1^{i-1}}(u|u_1^{i-1})$ in (3.43). From (3.27) in Theorem 3.5, we can represent the ratio of the posterior probability by

$$\frac{P_{U_i|U_1^{i-1},Y_1^n}(1|\hat{u}_1^{i-1}, y_1^n)}{P_{U_i|U_1^{i-1},Y_1^n}(0|\hat{u}_1^{i-1}, y_1^n)} = \frac{P_{U_1^i,Y_1^n}((\hat{u}_1^{i-1}, 1), y_1^n)}{P_{U_1^i,Y_1^n}((\hat{u}_1^{i-1}, 0), y_1^n)}$$

$$= \frac{\tilde{W}_i^{(n)}((0^n, y_1^n), \hat{u}_1^{i-1}|1)}{\tilde{W}_i^{(n)}((0^n, y_1^n), \hat{u}_1^{i-1}|0)} . \tag{3.45}$$

Note that the RHS of (3.45) can be expressed by the recursive formula given in (3.17) and can be computed with complexity $\mathrm{O}(n \log n)$ using a dynamic programming. Hence, we can compute $P_{U_i|U_1^{i-1},Y_1^n}(u|\hat{u}_1^{i-1}, y_1^n)$ with complexity $\mathrm{O}(n \log n)$. We can also compute $P_{U_i|U_1^{i-1}}(u|u_1^{i-1})$ similarly by letting Y be a constant random variable.

3.3.3 Simulation

In this section we compare the proposed scheme with a polar code using Gallager's method discussed in Section 1.2. We used a binary asymmetric erasure channel illustrated by Fig. 3.5 such that erasure probabilities for inputs 0 and 1 are $\epsilon_0 = 0.4$ and $\epsilon_1 = 0.8159$, respectively. The ideal input distribution of this channel is given by $(P_X(0), P_X(1)) = (7/16, 9/16)$ and the capacity is $C(W) = 0.4498$. For Gallager's method, we used 16-ary polar codes with generator matrix $G_n = F^{\otimes k}$ where $F = \left(\begin{smallmatrix} 1 & 0 \\ \gamma & 1 \end{smallmatrix}\right)$ for a primitive element $\gamma \in \mathrm{GF}(16)$. In the proposed scheme we used binary polar codes with generator matrix $F^{\otimes k}$ for $F = \left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$. Then, the complexity of the proposed scheme is at least $16/2 = 8$ times less than that by Gallager's method.

In Fig. 3.6, "GM" denotes the result for Gallager's method, and "random" stands for the scheme proposed in Section 3.3.1. Furthermore, "MAP" denotes the scheme obtained by replacing the randomization (3.43) with MAP assignment given by

$$u_i = \operatorname*{argmax}_{u \in \{0,1\}} P_{U_i|U_1^{i-1}}(u|u_1^{i-1}) . \tag{3.46}$$

As reported in the case of lossy coding [59], the MAP scheme works better than the randomized scheme although the theoretical analysis seems to be difficult. Further, in spite of the small complexity, both proposed schemes using (3.43) or (3.46) can achieve a better or compatible performance in decoding error probability compared with the coding scheme by Gallager's method especially in large coding rate.

Note that the channel used in this simulation is somewhat artificially designed so that the ideal input distribution can be represented by simple rational numbers. However, in general cases, the ideal input distribution may require a large alphabet size for Gallager's method. In such cases, the proposed scheme has much advantage since it can deal with any input distribution in

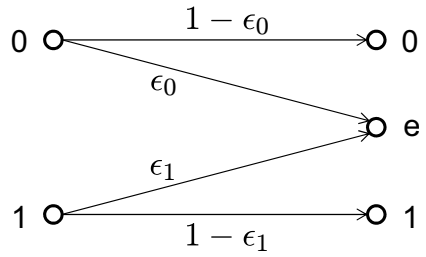

Fig. 3.5. Binary asymmetric erasure channel.

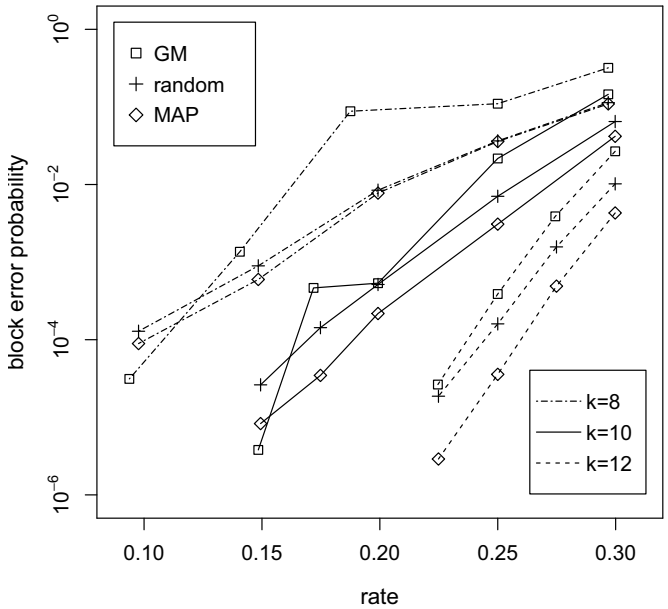

Fig. 3.6. Block decoding error probabilities of polar codes for an asymmetric erasure channel with $k = 8, 10, 12$.

the same manner.

3.4 Polar Codes for Lossy Source Coding

In this section we consider polar coding for nonuniform sources and/or asymmetric distortion measures.

Recall that for information source $Y \in \mathcal{Y}$ and distortion measure $d : \mathcal{Y} \times \{0, 1\} \to [0, +\infty)$, the rate-distortion function is given by

$$R(D) = \min_{X : \mathrm{E}_{XY}[d(Y,X)] \leq D} I(X; Y) \,. \tag{3.47}$$

In the following we assume that X is the random variable achieving this minimum.

Now we construct a polar code for the lossy coding problem. Assume that an information set $\mathcal{I} \subset \{1, \ldots, n\}$ and a frozen set $\mathcal{I}^c = \{1, \ldots, n\} \setminus \mathcal{I}$ are given and satisfy $|\mathcal{I}| = nR$ and

$$Z(U_i|U_1^{i-1}, Y_1^n) \geq 1 - 2^{-n^\beta} \quad \text{or} \quad Z(U_i|U_1^{i-1}) \leq 2^{-n^\beta} \tag{3.48}$$

for all $i \in \mathcal{I}^c$. From Theorem 3.4, such $\mathcal{I}$ exists if $R > I(X;Y) = R(D)$, $\beta < 1/2$, and n is sufficiently large.

As in the case of channel coding, let $\mathcal{L}_i$ be the family of functions $\lambda_i : \{0,1\}^{i-1} \to \{0,1\}$ and assume that $\lambda_{\mathcal{I}^c} \in \prod_{i \in \mathcal{I}^c} \mathcal{L}_i$ is shared between the encoder and the decoder. In the proposed scheme, the encoder determines $u_1^n = u_1^n(\lambda_{\mathcal{I}^c}, y_1^n)$ from a given source sequence y_1^n by

$$u_i = \begin{cases} 0 & \text{with probability } P_{U_i|U_1^{i-1},Y_1^n}(0|u_1^{i-1}, y_1^n) \\ 1 & \text{with probability } P_{U_i|U_1^{i-1},Y_1^n}(1|u_1^{i-1}, y_1^n) \end{cases} \tag{3.49}$$

for $i \in \mathcal{I}$ and $u_i = \lambda_i(u_1^{i-1})$ for $i \in \mathcal{I}^c$. The encoder sends $u_{\mathcal{I}}$ to the decoder. The decoder determines $u_{\mathcal{I}^c}$ by $u_i = \lambda_i(u_1^{i-1})$ and output reproduction sequence $x_1^n = u_1^n G_n$. Then, the coding rate is given by $R = |\mathcal{I}|/n$. We define the average distortion by

$$D_n(\lambda_{\mathcal{I}^c}) \equiv \frac{1}{n} \mathrm{E}_{Y_1^n}\left[\mathrm{E}[d^n(Y_1^n, u_1^n(\lambda_{\mathcal{I}^c}, Y_1^n)\, G_n)]\right] \tag{3.50}$$

where $d^n(y_1^n, x_1^n) \equiv \sum_{i=1}^n d(y_i, x_i)$ and the inner expectation is taken over randomization in (3.49).

As in the case of channel coding, we consider the expectation of $D_n(\Lambda_{\mathcal{I}^c})$ for random variable $\Lambda_{\mathcal{I}^c}$ such that $P_{\Lambda_i}[\Lambda_i(u_1^{i-1}) = 1] = P_{U_i|U_1^{i-1}}(1|u_1^{i-1})$ for all $u_1^{i-1} \in \{0,1\}^{i-1}$.

Theorem 3.7. *Let $\mathcal{I} \subset \{1, \ldots, n\}$ be a set satisfying* (3.48)*. Then the expectation of the average distortion $D_n(\Lambda_{\mathcal{I}^c})$ over the maps $\Lambda_{\mathcal{I}^c}$ satisfies $\mathrm{E}_{\Lambda_{\mathcal{I}^c}}[D_n(\Lambda_{\mathcal{I}^c})] = D + \mathrm{O}(2^{-n^{\beta'}})$ for any $R > R(D)$ and $\beta' < \beta < 1/2$. Consequently, there exists a deterministic map $\lambda_{\mathcal{I}^c} = \{\lambda_i \in \mathcal{L}_i\}_{i \in \mathcal{I}^c}$ such that $D_n(\lambda_{\mathcal{I}^c}) = D + \mathrm{O}(2^{-n^{\beta'}})$.*

The proof follows the same line as that of Theorem 3.6 and is given in Section 3.5.

Remark 3.1. In the lossy coding for symmetric setting [59], $u_{\mathcal{I}^c}$ is determined beforehand uniformly from $\{0,1\}^{|\mathcal{I}^c|}$ and the randomized map $\Lambda_{\mathcal{I}^c}$ for the frozen set is not required. In our setting, the achievability of the rate-distortion function can be proved in the similar way as [59] for a simplified

rule such that for $i \in \mathcal{I}^c$

$$u_i = \begin{cases} \bar{u}_i & \text{if } Z(U_i|U_1^{i-1}, Y_1^n) \geq 1 - 2^{-n^\beta}, \\ \operatorname*{argmax}_u P_{U_i|U_1^{i-1}}(u|u_1^{i-1}) & \text{if } Z(U_i|U_1^{i-1}) \leq 2^{-n^\beta}, \end{cases} \tag{3.51}$$

where $\bar{u}_i$ is determined beforehand uniformly from $\{0, 1\}$. However, since this rule makes the proof of Theorem 3.7 a little longer, the rule in (3.49) is used for simplicity although the map $\Lambda_{\mathcal{I}^c}$ has to be shared between the encoder and the decoder under this rule.

3.5 Proofs of Coding Theorems

In this section, we give proofs of Theorems 3.6 and 3.7 on the optimality of polar codes for channel coding and lossy source coding.

3.5.1 Channel Coding

Let $\mathcal{E}_i$ be the set of pairs of every codeword $u_1^n = u_1^n(M_1^{|\mathcal{I}|}, \lambda_{\mathcal{I}^c})$ and every received word y_1^n such that decoding error occurs at the i-th bit. The block decoding error event is given by $\mathcal{E} \equiv \bigcup_{i \in \mathcal{I}} \mathcal{E}_i$. Under decoding given in (3.41) with an arbitrary tie-breaking rule, every $(u_1^n, y_1^n) \in \mathcal{E}_i$ satisfies

$$P_{U_i|U_1^{i-1},Y_1^n}(u_i|u_1^{i-1}, y_1^n) \leq P_{U_i|U_1^{i-1},Y_1^n}(u_i \oplus 1|u_1^{i-1}, y_1^n)\,. \tag{3.52}$$

Consider the block decoding error probability $P_{\mathrm{e}}(\lambda_{\mathcal{I}^c})$ for map $\lambda_{\mathcal{I}^c}$. Since each codeword u_1^n appears with probability

$$2^{-|\mathcal{I}|}\mathbb{1}\left[\bigcap_{i \in \mathcal{I}^c}\{\lambda_i(u_1^{i-1}) = u_i\}\right], \tag{3.53}$$

$P_{\mathrm{e}}(\lambda_{\mathcal{I}^c})$ is given by

$$P_{\mathrm{e}}(\lambda_{\mathcal{I}^c}) = \sum_{u_1^n, y_1^n} 2^{-|\mathcal{I}|}\mathbb{1}\left[\bigcap_{i \in \mathcal{I}^c}\{\lambda_i(u_1^{i-1}) = u_i\}\right] P_{Y_1^n|U_1^n}(y_1^n|u_1^n)\,\mathbb{1}[(u_1^n, y_1^n) \in \mathcal{E}]\,. \tag{3.54}$$

From (3.42), the expectation of the decoding error probability is obtained as

$$\begin{aligned} &\mathrm{E}_{\Lambda_{\mathcal{I}^c}}[P_{\mathrm{e}}(\Lambda_{\mathcal{I}^c})] \\ &= \sum_{u_1^n, y_1^n} 2^{-|\mathcal{I}|}\left(\prod_{i \in \mathcal{I}^c} P_{U_i|U_1^{i-1}}(u_i|u_1^{i-1})\right) P_{Y_1^n|U_1^n}(y_1^n|u_1^n)\,\mathbb{1}[(u_1^n, y_1^n) \in \mathcal{E}]\,. \end{aligned} \tag{3.55}$$

Then, using probability distribution $Q_{U_1^n,Y_1^n}$ defined as

$$Q_{U_1^n,Y_1^n}(u_1^n,y_1^n) \equiv P_{Y_1^n|U_1^n}(y_1^n|u_1^n)2^{-|\mathcal{I}|}\prod_{i\in\mathcal{I}^c}P_{U_i|U_1^{i-1}}(u_i|u_1^{i-1})\,, \quad (3.56)$$

we can represent (3.55) as $\mathrm{E}_{\Lambda_{\mathcal{I}^c}}[P_{\mathrm{e}}(\Lambda_{\mathcal{I}^c})] = Q_{U_1^n,Y_1^n}(\mathcal{E})$. Let $\|F-G\|$ be the variational distance defined by

$$\|F-G\| \equiv \frac{1}{2}\sum_x |F(x)-G(x)| = \sum_{x:F(x)>G(x)}(F(x)-G(x)) \quad (3.57)$$

for probability distributions F and G. The variational distance between $Q_{U_1^n,Y_1^n}$ and $P_{U_1^n,Y_1^n}$ satisfies the following lemma.

Lemma 3.8. *For any $\beta < 1/2$ satisfying* (3.44) *and $\beta' < \beta$,*

$$\|P_{U_1^n,Y_1^n} - Q_{U_1^n,Y_1^n}\| = \mathrm{O}(2^{-n^{\beta'}})\,. \quad (3.58)$$

Proof. We use an argument similar to [59, Lemma 3.5] based on the expression

$$B_1^n - A_1^n = \sum_{i=1}^n A_1^{i-1}B_i^n - \sum_{i=1}^n A_1^i B_{i+1}^n = \sum_{i=1}^n (B_i - A_i)A_1^{i-1}B_{i+1}^n\,, \quad (3.59)$$

where A_j^k and B_j^k denote products $\prod_{i=j}^k A_i$ and $\prod_{i=j}^k B_i$, respectively.

For simplicity, we omit the symbols of random variables, e.g. $P(u_1^n,y_1^n)$ and $Q(u_i|u_1^{i-1},y_1^n)$ for $P_{U_1^n,Y_1^n}(u_1^n,y_1^n)$ and $Q_{U_i|U_1^{i-1},Y_1^n}(u_i|u_1^{i-1},y_1^n)$ in the following. Now $\|P_{U_1^n,Y_1^n} - Q_{U_1^n,Y_1^n}\|$ is bounded as follows.

$$\begin{aligned}
&2\|P_{U_1^n,Y_1^n} - Q_{U_1^n,Y_1^n}\| \\
&= \sum_{u_1^n,y_1^n}|Q(u_1^n,y_1^n) - P(u_1^n,y_1^n)| \\
&\overset{(a)}{=} \sum_{u_1^n,y_1^n}|(Q(u_1^n) - P(u_1^n))P(y_1^n|u_1^n)| \\
&\overset{(b)}{=} \sum_{u_1^n,y_1^n}\Bigg|\sum_i (Q(u_i|u_1^{i-1}) - P(u_i|u_1^{i-1})) \\
&\qquad\qquad \cdot\left(\prod_{j=1}^{i-1}P(u_j|u_1^{j-1})\right)\left(\prod_{j=i+1}^{n}Q(u_j|u_1^{j-1})\right)P(y_1^n|u_1^n)\Bigg| \\
&\overset{(c)}{\le} \sum_{i\in\mathcal{I}}\sum_{u_1^n,y_1^n}\left|Q(u_i|u_1^{i-1}) - P(u_i|u_1^{i-1})\right|
\end{aligned}$$

$$\cdot \left(\prod_{j=1}^{i-1} P(u_j|u_1^{j-1})\right)\left(\prod_{j=i+1}^{n} Q(u_j|u_1^{j-1})\right) P(y_1^n|u_1^n)$$

$$= \sum_{i\in\mathcal{I}} \sum_{u_1^{i-1}} 2P(u_1^{i-1}) \| Q_{U_i|U_1^{i-1}=u_1^{i-1}} - P_{U_i|U_1^{i-1}=u_1^{i-1}} \|$$

$$\overset{\text{(d)}}{\leq} \sum_{i\in\mathcal{I}} \sum_{u_1^{i-1}} P(u_1^{i-1}) \sqrt{(2\ln 2) D(P_{U_i|U_1^{i-1}=u_1^{i-1}} \| Q_{U_i|U_1^{i-1}=u_1^{i-1}})}$$

$$\overset{\text{(e)}}{\leq} \sum_{i\in\mathcal{I}} \sqrt{(2\ln 2) \sum_{u_1^{i-1}} P(u_1^{i-1}) D(P_{U_i|U_1^{i-1}=u_1^{i-1}} \| Q_{U_i|U_1^{i-1}=u_1^{i-1}})}\,, \tag{3.60}$$

where $D(\cdot\|\cdot)$ is the relative entropy, and equalities (a),(b) and inequalities (b)–(d) follow from

(a): $Q(y_1^n|u_1^n) = P(y_1^n|u_1^n)$,
(b): (3.59),
(c): $Q(u_i|u_1^{i-1}) = P(u_i|u_1^{i-1})$ for $i \in \mathcal{I}^c$,
(d): Pinsker's inequality (see, e.g., [51, Lemma 11.6.1]) given by

$$\|F - G\| \leq \sqrt{(\ln 2) D(F\|G)/2}\,, \tag{3.61}$$

(e): Jensen's inequality (see, e.g., [51, P. 43]).

Hence, it holds that

$$\begin{aligned}
2\|P_{U_1^n,Y_1^n} - Q_{U_1^n,Y_1^n}\| &\leq \sum_{i\in\mathcal{I}} \sqrt{(2\ln 2) D(P_{U_i} \| Q_{U_i}|U_1^{i-1})} \\
&\overset{\text{(f)}}{=} \sum_{i\in\mathcal{I}} \sqrt{(2\ln 2)(1 - H(U_i|U_1^{i-1}))} \\
&\overset{\text{(g)}}{\leq} \sum_{i\in\mathcal{I}} \sqrt{(2\log 2)(1 - (Z(U_i|U_1^{i-1}))^2)} \\
&\overset{\text{(h)}}{\leq} n\sqrt{(4\log 2)\cdot 2^{-n^\beta}} \\
&\overset{\text{(i)}}{=} \mathrm{O}(2^{-n^{\beta'}})\,,
\end{aligned} \tag{3.62}$$

where the equality and the inequalities follow from

(f): $Q_{U_i|U_1^{i-1}} = \frac{1}{2}$ for $i \in \mathcal{I}$,
(g): Lemma 3.3,
(h): (3.44) and
(i): $\beta' < \beta$.

□

Proof of Theorem 3.6. First we have

$$\begin{aligned}\mathrm{E}_{\Lambda_{\mathcal{I}^c}}[P_{\mathrm{e}}(\Lambda_{\mathcal{I}^c})] &= Q_{U_1^n,Y_1^n}(\mathcal{E}) \\ &\le \|Q_{U_1^n,Y_1^n} - P_{U_1^n,Y_1^n}\| + P_{U_1^n,Y_1^n}(\mathcal{E}) \\ &\le \|Q_{U_1^n,Y_1^n} - P_{U_1^n,Y_1^n}\| + \sum_{i\in\mathcal{I}} P_{U_1^n,Y_1^n}(\mathcal{E}_i)\,. \end{aligned} \tag{3.63}$$

Each term in the summation can be bounded as

$$\begin{aligned} &P_{U_1^n,Y_1^n}(\mathcal{E}_i) \\ &\le \sum_{u_1^i,y_1^n} P(u_1^{i-1},y_1^n)P(u_i|u_1^{i-1},y_1^n)\mathbb{1}\left[P(u_i|u_1^{i-1},y_1^n) \le P(u_i\oplus 1|u_1^{i-1},y_1^n)\right] \\ &\le \sum_{u_1^i,y_1^n} P(u_1^{i-1},y_1^n)P(u_i|u_1^{i-1},y_1^n)\sqrt{\frac{P(u_i\oplus 1|u_1^{i-1},y_1^n)}{P(u_i|u_1^{i-1},y_1^n)}} \\ &= Z(U_i|U_1^{i-1},Y_1^n) \\ &\le 2^{-n^\beta}\,, \end{aligned} \tag{3.64}$$

where the last inequality follows from (3.44). From (3.58), (3.63), (3.64) and $|\mathcal{I}| \le n$, we have $\mathrm{E}_{\Lambda_{\mathcal{I}^c}}[P_{\mathrm{e}}(\Lambda_{\mathcal{I}^c})] = \mathrm{O}(2^{-n^{\beta'}})$. □

3.5.2 Lossy Source Coding

Proof of Theorem 3.7. For a source sequence y_1^n and the encoding rule (3.49), $u_1^n = u_1^n(y_1^n, \lambda_{\mathcal{I}^c})$ appears with probability

$$\left(\prod_{i\in\mathcal{I}} P_{U_i|U_1^{i-1},Y_1^n}(u_i|u_1^{i-1},y_1^n)\right)\mathbb{1}\left[\bigcap_{i\in\mathcal{I}^c}\{\lambda_i(u_1^{i-1}) = u_i\}\right]. \tag{3.65}$$

The average distortion for map $\Lambda_{\mathcal{I}^c} = \lambda_{\mathcal{I}^c}$ is expressed as

$$\begin{aligned} D_n(\lambda_{\mathcal{I}^c}) = \frac{1}{n}\sum_{u_1^n,y_1^n} P_{Y_1^n}(y_1^n)&\left(\prod_{i\in\mathcal{I}} P_{U_i|U_1^{i-1},Y_1^n}(u_i|u_1^{i-1},y_1^n)\right) \\ &\cdot\mathbb{1}\left[\bigcap_{i\in\mathcal{I}^c}\{\lambda_i(u_1^{i-1}) = u_i\}\right] d^n(y_1^n, u_1^n G_n) \end{aligned} \tag{3.66}$$

and its expectation over $\Lambda_{\mathcal{I}^c}$ is

$$\begin{aligned} \mathrm{E}_{\Lambda_{\mathcal{I}^c}}[D_n(\Lambda_{\mathcal{I}^c})] = \frac{1}{n}\sum_{u_1^n,y_1^n} P_{Y_1^n}(y_1^n)&\left(\prod_{i\in\mathcal{I}} P_{U_i|U_1^{i-1},Y_1^n}(u_i|u_1^{i-1},y_1^n)\right) \\ &\cdot\left(\prod_{i\in\mathcal{I}^c} P_{U_i|U_1^{i-1}}(u_i|u_1^{i-1})\right) d^n(y_1^n, u_1^n G_n)\,. \end{aligned} \tag{3.67}$$

Then, for probability distribution $Q_{U_1^n,Y_1^n}$ defined as

$$\begin{aligned}&Q_{U_1^n,Y_1^n}(u_1^n,y_1^n)\\&\quad\equiv P_{Y_1^n}(y_1^n)\left(\prod_{i\in\mathcal{I}}P_{U_i|U_1^{i-1},Y_1^n}(u_i|u_1^{i-1},y_1^n)\right)\left(\prod_{i\in\mathcal{I}^c}P_{U_i|U_1^{i-1}}(u_i|u_1^{i-1})\right),\end{aligned}\tag{3.68}$$

(3.67) is represented as

$$\mathrm{E}_{\Lambda_{\mathcal{I}^c}}[D_n(\Lambda_{\mathcal{I}^c})]=\frac{1}{n}\mathrm{E}_{Q_{U_1^n,Y_1^n}}[d(Y_1^n,U_1^nG_n)]\,.\tag{3.69}$$

Therefore we obtain

$$\begin{aligned}&\mathrm{E}_{\Lambda_{\mathcal{I}^c}}[D_n(\Lambda_{\mathcal{I}^c})]\\&\quad\le\frac{1}{n}\mathrm{E}_{P_{U_1^n,Y_1^n}}[d^n(Y_1^n,G_nU_1^n)]+\frac{\max_{y,x}d(y,x)}{n}\|P_{U_1^n,Y_1^n}-Q_{U_1^n,Y_1^n}\|\end{aligned}\tag{3.70}$$

and the following lemma shows that the second term of the RHS of (3.70) is $\mathrm{O}(2^{-n^{\beta'}})$. □

Lemma 3.9. *For any $\beta<1/2$ satisfying* (3.48) *and $\beta'<\beta$,*

$$\|P_{U_1^n,Y_1^n}-Q_{U_1^n,Y_1^n}\|=\mathrm{O}(2^{-n^{\beta'}})\,.\tag{3.71}$$

Proof. By the same argument and notation as the proof of Lemma 3.8, $\|P_{U_1^n,Y_1^n}-Q_{U_1^n,Y_1^n}\|$ is bounded as follows.

$$\begin{aligned}&2\|P_{U_1^n,Y_1^n}-Q_{U_1^n,Y_1^n}\|\\&\overset{\text{(a)}}{=}\sum_{u_1^n,y_1^n}\Bigg|\sum_i(Q(u_i|u_1^{i-1},y_1^n)-P(u_i|u_1^{i-1},y_1^n))P(y_1^n)\\&\qquad\qquad\cdot\left(\prod_{j=1}^{i-1}P(u_j|u_1^{j-1},y_1^n)\right)\left(\prod_{j=i+1}^{N}Q(u_j|u_1^{j-1},y_1^n)\right)\Bigg|\\&\overset{\text{(b)}}{\le}\sum_{i\in\mathcal{I}^c}\sum_{u_1^i,y_1^n}\left|Q(u_i|u_1^{i-1},y_1^n)-P(u_i|u_1^{i-1},y_1^n)\right|P(y_1^n)\left(\prod_{j=1}^{i-1}P(u_j|u_1^{j-1},y_1^n)\right)\\&=\sum_{i\in\mathcal{I}^c}\sum_{u_1^{i-1},y_1^n}2P(u_1^{i-1},y_1^n)\|Q_{U_i|Y_1^n=y_1^n,U_1^{i-1}=u_1^{i-1}}-P_{U_i|Y_1^n=y_1^n,U_1^{i-1}=u_1^{i-1}}\|\\&\le\sum_{i\in\mathcal{I}^c}\sum_{u_1^{i-1},y_1^n}P(u_1^{i-1},y_1^n)\end{aligned}$$

$$
\begin{aligned}
&\qquad \cdot \sqrt{(2\ln 2) D(P_{U_i|Y_1^n=y_1^n,U_1^{i-1}=u_1^{i-1}} \| Q_{U_i|Y_1^n=y_1^n,U_1^{i-1}=u_1^{i-1}})} \\
&\le \sum_{i\in\mathcal{I}^c} \sqrt{(2\ln 2)} \\
&\qquad \cdot \sqrt{\sum_{u_1^{i-1},y_1^n} P(u_1^{i-1},y_1^n) D(P_{U_i|Y_1^n=y_1^n,U_1^{i-1}=u_1^{i-1}} \| Q_{U_i|Y_1^n=y_1^n,U_1^{i-1}=u_1^{i-1}})} \\
&= \sum_{i\in\mathcal{I}^c} \sqrt{(2\ln 2) D(P_{U_i} \| Q_{U_i} | U_1^{i-1}, Y_1^n)} \\
&\overset{\text{(c)}}{=} \sum_{i\in\mathcal{I}^c} \sqrt{(2\ln 2)(H(U_i|U_1^{i-1}) - H(U_i|U_1^{i-1},Y_1^n))}\,, \qquad (3.72)
\end{aligned}
$$

where the equalities and the inequalities follow from

(a): (3.59) and $Q(y_1^n) = P(y_1^n)$,
(b): $Q(u_i|u_1^{i-1}, y_1^n) = P(u_i|u_1^{i-1}, y_1^n)$ for $i \in \mathcal{I}$,
(c): $Q_{U_i|U_1^{i-1},Y_1^n} = P_{U_i|U_1^{i-1}}$ for $i \in \mathcal{I}^c$.

Furthermore it holds for all $i \in \mathcal{I}^c$ that

$$
\begin{aligned}
&H(U_i|U_1^{i-1}) - H(U_i|U_1^{i-1},Y_1^n) \\
&\qquad \overset{\text{(d)}}{\le} Z(U_i|U_1^{i-1}) - (Z(U_i|U_1^{i-1},Y_1^n))^2 \\
&\qquad \overset{\text{(e)}}{\le} \min\{Z(U_i|U_1^{i-1}),\, 1 - (Z(U_i|U_1^{i-1},Y_1^n))^2\} \\
&\qquad \overset{\text{(f)}}{\le} 2\cdot 2^{-n^\beta}\,. \qquad (3.73)
\end{aligned}
$$

from

(d): Lemma 3.3,
(e): $Z(\cdot|\cdot) \in [0,1]$,
(f): (3.48).

We obtain the lemma by combining (3.72) and (3.73). □

Chapter 4

LDPC Codes for Asymmetric Channels and Sources

4.1 Introduction

Low density parity check (LDPC) codes are a family of linear codes constructed by sparse parity check matrices called LDPC matrices, i.e., linear codes such that the codebook can be represented by

$$\mathcal{C} = \{x_1^n \in \mathcal{X}^n : x_1^n H = v_1^k\} \tag{4.1}$$

for some $\mathcal{X} = \mathrm{GF}(q)$ and an $n \times k$ sparse matrix H over $\mathrm{GF}(q)$. These codes were originally invented for channel coding by Gallager [13] in 60's and rediscovered by MacKay [14] in 90's. Since after the rediscovery, these codes have been researched extensively and known to achieve good performance very near the channel capacity [15]. Miller-Burshtein [62] showed that the asymptotic optimality of LDPC codes for symmetric channels and later the optimality was extended to general channels [36].

Because of the duality of channel coding and lossy source coding, LDPC codes can also be applied to lossy source coding. In fact, an ensemble of LDPC codes achieves the rate-distortion function asymptotically for binary symmetric sources [27]. Extending this result, an asymptotically optimal coding scheme was proposed for general sources by Miyake-Muramatsu [40].

Since their results on the asymptotic optimality of LDPC codes are based on a maximum likelihood decoder, it is still important to find an efficient algorithm to attain near the Shannon bound practically. In channel coding, this goal can be accomplished by a decoding algorithm called belief propagation (BP). Especially, it is known that nonbinary LDPC codes over

GF(q), $q \geq 64$, achieve performance very close to the channel capacity with small block length [63][64] with complexity $\mathrm{O}(q \log q)$ [65].

On the other hand in lossy source coding, the BP does not work successfully because lossy source coding corresponds to channel coding of very noisy channels. Low density generator matrix (LDGM) codes were investigated to overcome this difficulty. LDGM codes are a family of linear codes constructed by sparse generator matrices and can be regarded as a dual of LDPC codes. These codes can achieve the rate-distortion function asymptotically for symmetric sources [30][31] and the near optimal performance can be attained practically by reinforced belief propagation (RBP) [32]. It was reported in [33][34] that the RBP algorithm also works well for LDPC codes by using nonbinary LDPC matrices.

4.1.1 LDPC Codes for Asymmetric Settings

Let consider channel coding for asymmetric channels and lossy source coding for nonuniform sources and/or asymmetric distortion measures. In [36], an asymptotically optimal channel code was constructed based on Gallager's method discussed in Section 1.2. Also, in lossy source coding, nonlinear coding schemes based on LDGM matrices were proposed in [37][38] and their asymptotic optimality was proved based on the Gallager's method.

From a practical viewpoint, Gallager's method causes large complexity when the optimal symbol distribution of codewords is not approximated by simple rational numbers. Furthermore, when the method is applied to LDPC codes for channel coding as in [36], the decoding error probability worsens considerably as explained in the following. As in Section 1.2, consider the case that the optimal symbol distribution of codewords for a channel W is expressed as

$$P_X(x_i) = \frac{p_i}{\tilde{q}} \tag{4.2}$$

for integers $\{p_i\}_{i=1,2,\cdots,|\mathcal{X}|}$ and $\tilde{q} = \sum_{i=1}^{|\mathcal{X}|} p_i$. The Gallager's method realizes this distribution by an LDPC code over GF($\tilde{q}$) with a codebook $\mathcal{C} \ni \tilde{x}_1^n$ and a map

$$Q : \tilde{x} \mapsto \begin{cases} x_1, & \tilde{x} = 1, 2 \cdots, p_1, \\ x_2, & \tilde{x} = p_1 + 1,\ p_1 + 2, \cdots, p_1 + p_2, \\ \vdots & \qquad \vdots \\ x_{|\mathcal{X}|}, & \tilde{x} = \sum_{i=1}^{|\mathcal{X}|-1} p_i + 1,\ \sum_{i=1}^{|\mathcal{X}|-1} p_i + 2, \cdots, \tilde{q}. \end{cases} \tag{4.3}$$

Here recall (2.7), that is, the coding rate of this code is given by $R = (\log|\mathcal{C}|)/n = (1 - (\mathrm{rank}H)/n)\log\tilde{q}$. When the LDPC matrix H has full rank, it can be simply expressed as $R = (1 - k/n)\log\tilde{q}$. Then it holds for a code achieving coding rate near the channel capacity that

$$C(W) \approx \left(1 - \frac{k}{n}\right)\log\tilde{q} \tag{4.4}$$

and therefore

$$\frac{k}{n} \approx 1 - \frac{C(W)}{\log\tilde{q}}\,. \tag{4.5}$$

Then, the $n \times k$ LDPC matrix has to be close to a square matrix to achieve the capacity when $\tilde{q}$ in (4.2) is large. However, it has been reported that LDPC codes perform very poorly for such LDPC matrices [66].

Miyake-Muramatsu took another approach for the asymmetric settings in [39][40]. In their coding scheme for channel coding, they picked an LDPC codeword up with a desired symbol distribution by using the inverse operation of lossless compression. Similarly, for lossy source coding, they losslessly compressed an LDPC codeword with small distortion from the source sequence. By using such a lossless compression procedure, we can equally treat any symbol distribution P_X whereas the design of $\tilde{q}$ in (4.2) significantly affects the complexity and the performance in Gallager's method.

In Miyake-Muramatsu scheme, the lossless compression is performed by a fixed length code using another LDPC matrix. Theoretically, the fixed length lossless code can be decoded without errors with probability close to one for sufficiently large block length. However, it is necessary to compute the maximum likelihood (ML) codeword strictly although it is NP-complete in general [28]. Practically, known methods frequently fail in the ML decoding unless the redundancy of a code is set considerably large.

4.1.2 Our Contribution

We propose a new variable length coding scheme for lossy source coding and channel coding as an improvement of the Miyake-Muramatsu scheme in this chapter. We use arithmetic coding in Section 2.4.2 for the lossless compression required in lossy source coding, and use IAHC coding in Section 2.4.3 for the inverse operation of lossless compression required in channel coding. We prove that the proposed scheme can achieve the Shannon bound asymptotically under optimal encoding and decoding algorithms. In the proposed

scheme, the marginal probability of each symbol of LDPC codewords has to be computed for arithmetic coding or IAHC coding. We can achieve a near optimal performance by estimating this probability accurately. This fact contrasts with the fixed length coding where any codeword other than the ML codeword leads to a large distortion or a large decoding error probability. We propose a practical algorithm using belief propagation (BP) for this probability estimation in the next chapter.

4.2 Hash Property

The asymptotic optimality of LDPC codes was proved individually for channel coding and lossy source coding in [39] and [40], respectively. Later, these proofs were unified based on the notion of *hash property* of LDPC matrices in [48]. We use this property to prove the asymptotic optimality of the proposed scheme.

Let $\boldsymbol{\mathcal{A}} \equiv \{\mathcal{A}_n\}$ be a sequence of sets $\mathcal{A}_n$ of functions $A : \mathcal{X}^n \to \mathcal{X}^k$. The image of $\mathcal{A}_n$ is defined as $\mathrm{Im}\mathcal{A}_n \equiv \{A_n(x_1^n) : A_n \in \mathcal{A}_n,\, x_1^n \in \mathcal{X}^n\}$. Hash property is defined for the sequence $\boldsymbol{\mathcal{A}}$ and the sequence $\boldsymbol{P}_A \equiv \{P_{A_n}\}$ of probability distributions over $\mathcal{A}_n$ as follows.

Definition 4.1. *Assume that* $\boldsymbol{\mathcal{A}} \equiv \{\mathcal{A}_n\}$ *satisfies*

$$\lim_{n\to\infty} \frac{1}{n} \log \frac{|\mathcal{X}^k|}{|\mathrm{Im}\mathcal{A}_n|} = 0\,. \tag{4.6}$$

An ensemble $(\boldsymbol{\mathcal{A}}, \boldsymbol{P}_A)$ *has an* $(\boldsymbol{\alpha}_A, \boldsymbol{\beta}_A)$*-hash property if there exist two sequences* $\boldsymbol{\alpha}_A \equiv \{\alpha_A(n)\}_{n=1}^{\infty}$ *and* $\boldsymbol{\beta}_A \equiv \{\beta_A(n)\}_{n=1}^{\infty}$ *such that*

$$\lim_{n\to\infty} \alpha_A(n) = 1 \tag{4.7}$$

$$\lim_{n\to\infty} \beta_A(n) = 0 \tag{4.8}$$

and

$$\sum_{\substack{\boldsymbol{u}\in\mathcal{T}\\ \boldsymbol{u}'\in\mathcal{T}'}} P_{A_n}(\{A_n : A_n(\boldsymbol{u}) = A_n(\boldsymbol{u}')\}) \le |\mathcal{T}\cap\mathcal{T}'| + \frac{|\mathcal{T}|\,|\mathcal{T}'|\alpha_A(n)}{|\mathrm{Im}\mathcal{A}_n|} + \min\{|\mathcal{T}|,\, |\mathcal{T}'|\}\beta_A(n) \tag{4.9}$$

for any $\mathcal{T}, \mathcal{T}' \subset \mathcal{X}^n$.

We often write e.g. $\mathcal{A}$ or A instead of $\mathcal{A}_n$ or A_n when n is obvious from the context.

Algorithm 4.1 Construction of an LDPC Matrix H

1. Initialize every element of H by 0.
2. Repeat the following two steps c times in each row.
   2.1. Choose an element of the row and $a \in \mathrm{GF}(q) \setminus \{0\}$ uniformly at random.
   2.2. Add a to the chosen element.

Now we regard an LDPC matrix H as a function $\mathcal{X}^n \to \mathcal{X}^k : x_1^n \mapsto x_1^n H$. Let $(\mathcal{H}, P_H)$ be the ensemble of LDPC matrices which are randomly generated by Algorithm 5.1, where $\mathcal{H}$ is the family of such LDPC matrices and P_H denotes the probability distribution of LDPC matrix H. We consider this ensemble of LDPC matrices for the theoretical analysis in this chapter whereas we use another ensemble in the next chapter for practical implementation. It is shown in [48, Lemma 12] for even c that

$$|\mathrm{Im}\mathcal{H}| = \begin{cases} 2^{k-1}, & \text{if } q = 2, \\ 2^k, & \text{otherwise.} \end{cases} \tag{4.10}$$

We always assume that c is even to use the following result in [48].

Lemma 4.1 ([48, Theorem 2])**.** *Consider the ensemble* $(\boldsymbol{\mathcal{H}}, \boldsymbol{P}_H)$ *of LDPC matrices generated by Algorithm 4.1 with maximum row weight* $c = 2\lceil \ln(k^2/n) \rceil$*. Then,* $(\boldsymbol{\mathcal{H}}, \boldsymbol{P}_H)$ *satisfies* $(\boldsymbol{\alpha}_H, \boldsymbol{\beta}_H)$*-hash property for some* $(\boldsymbol{\alpha}_H, \boldsymbol{\beta}_H)$ *satisfying* (4.7)–(4.9)*.*

4.3 Variable Length Coding Schemes

We propose variable length coding schemes using LDPC matrices for lossy source coding and channel coding in this section. We prove that the proposed schemes achieve the Shannon bound when the LDPC matrices have the hash property.

First we introduce Miyake-Muramatsu schemes for lossy source coding [40] in Fig. 4.1 and channel coding [39] in Fig. 4.2. In their scheme for lossy source coding, a source sequence y_1^n is first vector-quantized to an LDPC codeword $\hat{x}_1^n$ which satisfies $\hat{x}_1^n H = v_1^k$. The sequence $\hat{x}_1^n$ is next compressed into u_1^n by another matrix $\tilde{H}$. In the decoder, the sequence $\hat{x}_1^n$ is estimated as $\tilde{x}_1^n$, which coincides with $\hat{x}_1^n$ with high probability. It is easy to see that, in their coding scheme for channel coding, the encoder and the decoder are just the same

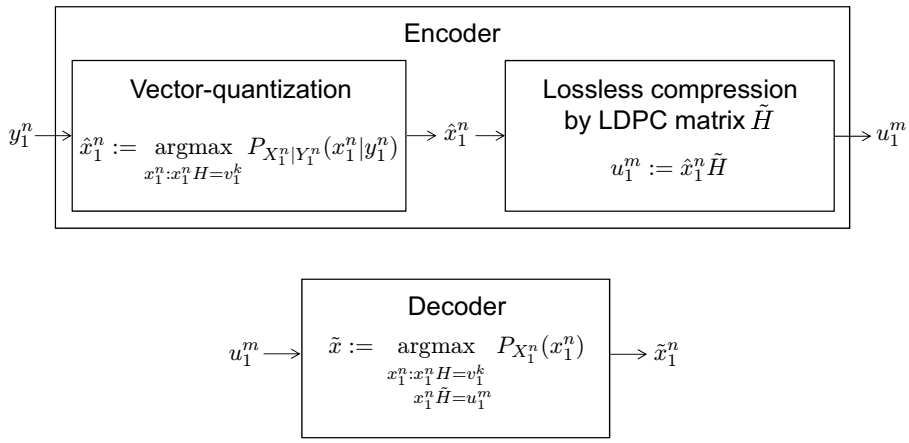

Fig. 4.1. Miyake-Muramatsu scheme [40] for lossy source coding.

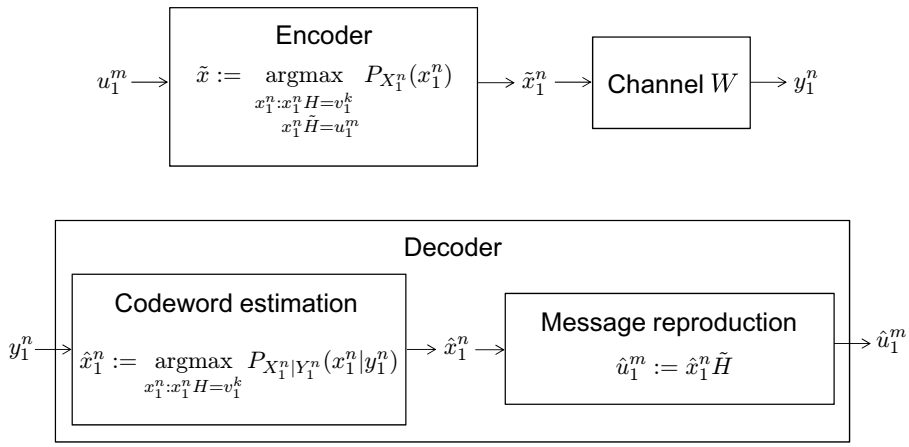

Fig. 4.2. Miyake-Muramatsu scheme [39] for channel coding.

as the decoder and the encoder of their scheme for lossy source coding. The following sections give our coding schemes which are obtained by replacing the lossless compression using an LDPC matrix $\tilde{H}$ with variable length codes introduced in Section 2.4. We propose variable length lossy source code and channel code in Sections 4.3.1 and 4.3.2, respectively.

4.3.1 Lossy Source Coding

First we treat a variable length lossy source code. The encoder φ of our scheme also consists of a vector-quantization part and a lossless compression part as Miyake-Muramatsu scheme. The flow of our coding scheme is given in Fig. 4.3.

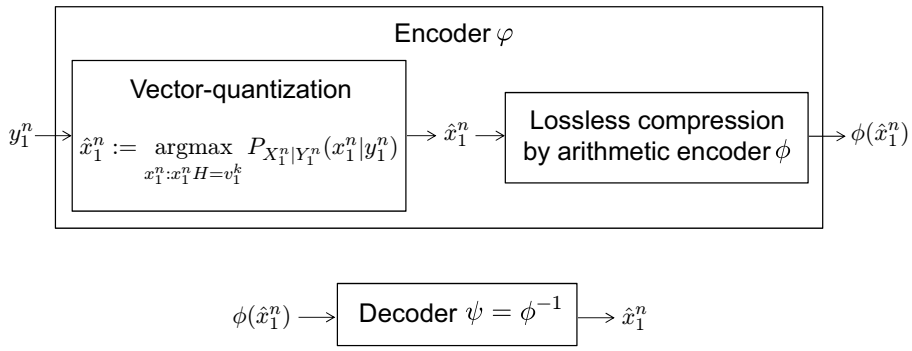

Fig. 4.3. Proposed lossy source coding scheme.

In the vector-quantization part, we use the same procedure as Miyake-Muramatsu scheme, which is described as follows. For an adequately selected $v_1^k \in \mathcal{X}^k$, the vector-quantized sequence $\hat{x}_1^n$ for a source sequence y_1^n is given by

$$\hat{x}_1^n = \operatorname*{argmax}_{x_1^n : x_1^n H = v_1^k} P_{X_1^n|Y_1^n}(x_1^n|y_1^n)\,. \tag{4.11}$$

We sometimes write $\hat{x}_1^n(y_1^n)$ or $\hat{x}_1^n(y_1^n, H, v_1^k)$ to clarify the arguments of $\hat{x}_1^n$.

In the lossless compression part, $\hat{x}_1^n = (\hat{x}_1, \cdots, \hat{x}_n)$ is compressed in this order by an arithmetic code consisting of encoder ϕ and decoder $\phi^{-1} = \psi$. Without loss of generality, we can assume that the rows of H are arranged so that

$$\operatorname{rank} H = \operatorname{rank} H_{m+1:n}\,, \tag{4.12}$$

where $m \equiv n - \operatorname{rank} H$ and H_i^j, $i \le j$, denotes a submatrix of A which consists of the $i, i+1, \cdots, j$-th rows of H.

The probability distribution for the arithmetic encoding of $\hat{x}_i$ is given by

$$\hat{P}_i(x_i|\hat{x}_1^{i-1}) \equiv P_{X_1^n}[X_i = x_i | X_1^{i-1} = \hat{x}_1^{i-1}, X_1^n H = v_1^k]\,, \quad x_i \in \mathcal{X}\,. \tag{4.13}$$

Let $\phi(\hat{x}_1^n)$ and $L(\hat{x}_1^n)$ denote the codeword and the code length for $\hat{x}_1^n$, respectively. Our encoder $\varphi(y_1^n)$ for source sequence y_1^n is defined by $\varphi(y_1^n) \equiv \phi(\hat{x}_1^n(y_1^n))$. Therefore the decoded sequence satisfies $\psi(\varphi(y_1^n)) = \hat{x}_1^n(y_1^n)$ and the code length $l(y_1^n)$ is given by $l(y_1^n) = L(\hat{x}_1^n)$.

Note that $\hat{P}_i(\hat{x}_i|\hat{x}_1^{i-1})$ given by (4.13) equals 1 for $i = m+1, \cdots, n$ and $\hat{x}_{m+1}, \cdots, \hat{x}_n$ are encoded into the null string with 0-bit length in this scheme. Also, note that the arithmetic coding is not necessary when P_X is uniform,

which occurs for symmetric settings, e.g., uniform input distribution with Hamming distortion measure. For this case, it suffices to send $(\hat{x}_1, \cdots, \hat{x}_m)$ without compression as shown in [27].

Assume that v_1^k is chosen randomly with uniform distribution $P_{V_1^k}$ on $\mathrm{Im}\mathcal{H}$. Hence, (H, v_1^k) has the distribution $P_{HV_1^k} = P_H \cdot P_{V_1^k}$. Then, the following theorem holds.

Theorem 4.2. *Assume that the ensemble of LDPC matrices H has the hash property. For any fixed $\epsilon > 0$, take $\delta > 0$ satisfying*

$$\epsilon > \max\left\{ d_{\max}\sqrt{\frac{5(\ln 2)\delta}{2}},\ 8\delta - \sqrt{10\,\delta}\log\frac{\sqrt{10\,\delta}}{|\mathcal{X}|} \right\}. \tag{4.14}$$

Then, if $H(X|Y) - 2\delta \le k/n \le H(X|Y) - \delta$, it holds for any $\xi > 0$ that

$$\lim_{n\to\infty} P_{HV_1^k}\left[P_{Y_1^n}\left[\frac{d^n(Y_1^n, \psi(\varphi(Y_1^n)))}{n} > D + \epsilon \right] > \xi \right] = 0 \tag{4.15}$$

and

$$\lim_{n\to\infty} P_{HV_1^k}\left[P_{Y_1^n}\left[\frac{l(Y_1^n)}{n} > R(D) + \epsilon \right] > \xi \right] = 0\,. \tag{4.16}$$

This theorem means that the proposed scheme with randomly generated (H, v_1^k) achieves the rate-distortion function (in the probability sense) with arbitrarily high probability.

We can also derive the achievability in the average sense as follows.

Corollary 4.3. *On the same conditions as Theorem 4.2, it holds for any $\epsilon' > 0$ that*

$$\lim_{n\to\infty} P_{HV_1^k}\left[\mathrm{E}_{Y_1^n}\left[\frac{d^n(Y_1^n, \psi(\varphi(Y_1^n)))}{n} \right] > D + \epsilon' \right] = 0 \tag{4.17}$$

and

$$\lim_{n\to\infty} P_{HV_1^k}\left[\mathrm{E}_{Y_1^n}\left[\frac{l(Y_1^n)}{n} \right] > R(D) + \epsilon' \right] = 0\,. \tag{4.18}$$

We prove these theorem and corollary in Section 4.4.

4.3.2 Channel Coding

Next we propose a variable length channel code. As described before, the channel code and the lossy source code of Miyake-Muramatsu scheme are

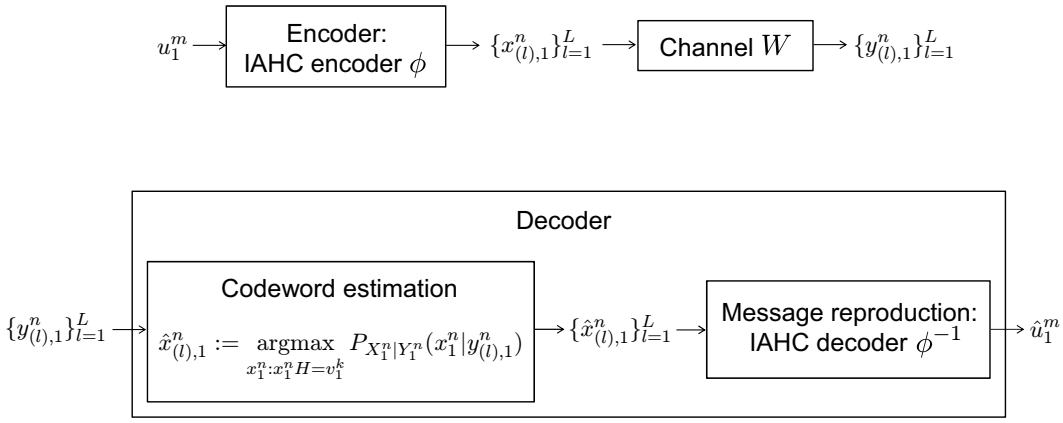

Fig. 4.4. Proposed channel coding scheme.

exactly dual to each other, that is, the encoder of one code coincides with the decoder of the other code. On the other hand, the arithmetic decoder of our lossy coding scheme cannot be used as an encoder for uniform sequences. Therefore, for the encoder of the channel code, we use IAHC coding introduced in Section 2.4.3 instead of arithmetic coding.

Now we describe our coding scheme, the flow of which is given in Fig. 4.4. Define a codebook of LDPC codewords by $\mathcal{C} \equiv \{x_1^n \in \mathcal{X}^n : x_1^n H = v_1^k\}$ for $\mathcal{X} = \mathrm{GF}(q)$. We encode an m-bit message U_1^m uniformly distributed on $\{0,1\}^m$ into a sequence in $\mathcal{C}^*$ by the IAHC coding. Then the LDPC code and the IAHC code can be regarded as an inner and an outer code, respectively. For the probability distribution on $\mathcal{C}$, we use

$$\begin{aligned}\hat{P}(x_1^n) &= P_{X_1^n}[X_1^n = x_1^n | X_1^n \in \mathcal{C}] \\ &= \prod_{i=1}^n P_{X_1^n}[X_i = x_i | X_1^{i-1} = x_1^{i-1},\, X_1^n \in \mathcal{C}] \\ &= \prod_{i=1}^n \hat{P}_i(x_i | x_1^{i-1})\,. \end{aligned} \tag{4.19}$$

More precisely, the IAHC encoder ϕ of our scheme is defined by Algorithm 2.1 where the input distribution P_U and the output distribution P_X are replaced with $P_U := P_{U_1^n}$ and $P_X := \hat{P}$, respectively. The IAHC decoder ϕ^{-1} is defined similarly for Algorithm 2.2. We define the encoder φ of the proposed scheme by $\varphi = \phi$. We denote the l-th inner codeword by $x^n_{(l),1} \in \mathcal{C}$. Then,

for the length $L = L(u_1^m)$[*1] of the inner codeword sequence, we can write $\varphi(u_1^m) = \{x_{(l),1}^n\}_{l=1}^L$. The average coding rate over uniform message is given by

$$R = \frac{m}{n\mathrm{E}_{U_1^m}[L(U_1^m)]}\,. \tag{4.20}$$

The decoder receives a sequence $\{y_{(l),1}^n\}_{l=1}^L$ according to the probability

$$\prod_{l=1}^{L} W^n(y_{(l),1}^n|x_{(l),1}^n)\,. \tag{4.21}$$

The decoder first estimates the sent sequence of LDPC codewords by

$$\hat{x}_{(l),1}^n = \operatorname*{argmax}_{x_1^n : x_1^n H = v_1^k} P_{X_1^n|Y_1^n}(x_{(l),1}^n|y_{(l),1}^n), \quad i = 1, 2, \cdots, L, \tag{4.22}$$

and next reproduce the message by the IAHC decoder ϕ^{-1}. Then the decoder ψ is given by $\psi(\{y_{(l),1}^n\}_{l=1}^L) = \phi^{-1}(\{\hat{x}_{(l),1}^n\}_{l=1}^L)$ and the decoding is successful if $\hat{x}_{(l),1}^n = x_{(l),1}^n$ for all $l = 1, 2, \cdots, L$. The average decoding error probability of the outer code over uniform message U_1^m is denoted by $P_{\mathrm{e,out}}$. The following theorem shows that the proposed scheme can achieve the channel capacity.

Theorem 4.4. *Assume that the ensemble of LDPC matrices H satisfies the hash property. Then there exists a sequence $\{(n_t, k_t, m_t)\}_t$ such that it holds for the message length m_t and the ensemble of $n_t \times k_t$ LDPC matrices H that*

$$\begin{aligned} &\lim_{t\to\infty} \mathrm{E}_{HV_1^k}[R] = C(W)\,, \\ &\lim_{t\to\infty} \mathrm{E}_{HV_1^k}[P_{\mathrm{e,out}}] = 0\,. \end{aligned} \tag{4.23}$$

We prove this theorem in Section 4.4.

Remark 4.1. Let us consider the case that the sequence $(u_{(1),1}^m, u_{(2),1}^m, \cdots)$ of m-bit messages is sent successively by the proposed variable length coding scheme. In this scheme, if a decoding error occurs for one inner code then the error propagates to the subsequent messages since the decoder fails to recover the code length of the outer code.

To avoid such an error propagation, it is desirable to encode a variable length message into a fixed length sequence of LDPC codewords, that is, use

[*1] Note that the dependence of the code length on the randomization of the IAHC coding is omitted in this notation.

a VF (variable-to-fixed) code instead of the proposed FV (fixed-to-variable) code. Unfortunately, an asymptotically optimal VF coding scheme has not been known for homophonic coding. However, in channel coding, each LDPC codeword does not have to follow the distribution $\hat{P}$ exactly and it suffices to follow a distribution close to $\hat{P}$. Thus, the homophonic coding scheme is not necessary to be $\hat{P}$-perfect and we may be able to devise such a coding scheme.

4.4 Proofs of Coding Theorems

In this section we prove Theorem 4.2 and Corollary 4.3 for lossy source coding and prove Theorem 4.4 for channel coding. The proofs are based on the method of types and hash property.

Let $P_{x_1^n}$ and $P_{x_1^n y_1^n}$ be the type and the joint type of sequences x_1^n and (x_1^n, y_1^n), respectively. The conditional type of x_1^n given y_1^n is denoted by $P_{x_1^n|y_1^n}$. For given $P_Y, P_{X|Y}, \gamma > 0$ and $\delta > 0$, we define a set of typical sequences $\mathcal{T}_{Y,\gamma}$ and a set of conditionally typical sequences $\mathcal{T}_{X|Y,\gamma}(y_1^n)$ as

$$\begin{aligned} \mathcal{T}_{Y,\gamma} &\equiv \{y_1^n : D(P_{y_1^n} \| P_Y) < \gamma\} \\ \mathcal{T}_{X|Y,\gamma}(y_1^n) &\equiv \{x_1^n : D(P_{x_1^n|y_1^n} \| P_{X|Y} | P_{y_1^n}) < \gamma\} . \end{aligned} \tag{4.24}$$

We denote the codebook $\mathcal{C}$ of LDPC codes by $\mathcal{C}_H(v_1^k) \equiv \{x_1^n : x_1^n H = v_1^k\}$. We use the following Lemmas 4.5–4.14 to prove the theorems. In the following, λ, η and ζ are defined for $\gamma, \gamma' > 0$ as

$$\begin{aligned} \lambda_{\mathcal{X}} &\equiv \frac{|\mathcal{X}| \log(n+1)}{n} , \\ \lambda_{\mathcal{X}\mathcal{Y}} &\equiv \frac{|\mathcal{X}||\mathcal{Y}| \log(n+1)}{n} , \\ \zeta_{\mathcal{X}}(\gamma) &\equiv \gamma - \sqrt{2\gamma} \log \frac{\sqrt{2\gamma}}{|\mathcal{X}|} , \end{aligned} \tag{4.25}$$

$$\begin{aligned} \zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma) &\equiv \gamma' - \sqrt{2\gamma'} \log \frac{\sqrt{2\gamma'}}{|\mathcal{X}||\mathcal{Y}|} + \sqrt{2\gamma} \log |\mathcal{X}| , \\ \eta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma) &\equiv -\sqrt{2\gamma'} \log \frac{\sqrt{2\gamma'}}{|\mathcal{X}||\mathcal{Y}|} + \sqrt{2\gamma} \log |\mathcal{X}| + \frac{|\mathcal{X}||\mathcal{Y}| \log(n+1)}{n} . \end{aligned} \tag{4.26}$$

Note that every above quantity approaches zero as $n \to \infty$ and $\gamma, \gamma' \to 0$.

Lemma 4.5 ([48, Lemma 21]).

$$\frac{1}{n}\log P_{X_1^n}(x_1^n) = -H(P_{x_1^n}) - D(P_{x_1^n}\|P_X)\,,$$
$$\frac{1}{n}\log P_{X_1^n|Y_1^n}(x_1^n|y_1^n) = -H(P_{x_1^n|y_1^n}|P_{y_1^n}) - D(P_{x_1^n|y_1^n}\|P_{X|Y}|P_{y_1^n}). \quad (4.27)$$

Lemma 4.6 ([48, Lemma 22]). *If $y_1^n \in \mathcal{T}_{Y,\gamma}$ and $x_1^n \in \mathcal{T}_{X|Y,\gamma'}(y_1^n)$ then $(x_1^n, y_1^n) \in \mathcal{T}_{XY,\gamma+\gamma'}$. If $(x_1^n, y_1^n) \in \mathcal{T}_{XY,\gamma}$ then $x_1^n \in \mathcal{T}_{X,\gamma}$.*

Lemma 4.7 ([48, Lemma 24]). *If $0 < \gamma < 1/8$ then*

$$\left|\frac{1}{n}\log\frac{1}{P_{X_1^n}(x_1^n)} - H(X)\right| \le \zeta_{\mathcal{X}}(\gamma) \quad (4.28)$$

for any $x_1^n \in \mathcal{T}_{X,\gamma}$ and

$$\left|\frac{1}{n}\log\frac{1}{P_{X_1^n|Y_1^n}(x_1^n|y_1^n)} - H(X|Y)\right| \le \zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma) \quad (4.29)$$

for any $y_1^n \in \mathcal{T}_{Y,\gamma}$ and $x_1^n \in \mathcal{T}_{X|Y,\gamma'}(y_1^n)$.

Lemma 4.8 ([48, Lemma 25]). *For any $\gamma > 0$ and $y_1^n \in \mathcal{Y}^n$,*

$$P_{Y_1^n}[Y_1^n \notin \mathcal{T}_{Y,\gamma}] \le 2^{-n(\gamma-\lambda_{\mathcal{Y}})},$$
$$P_{X_1^n|Y_1^n}[X_1^n \notin \mathcal{T}_{X|Y,\gamma}(y_1^n)|Y_1^n = y_1^n] \le 2^{-n(\gamma-\lambda_{\mathcal{X}\mathcal{Y}})}\,. \quad (4.30)$$

Lemma 4.9 ([48, Lemma 26]). *If $y_1^n \in \mathcal{T}_{Y,\gamma}$ then*

$$\left|\frac{1}{n}\log|\mathcal{T}_{Y,\gamma}| - H(Y)\right| \le \eta_{\mathcal{Y}}(\gamma)\,,$$
$$\left|\frac{1}{n}\log|\mathcal{T}_{X|Y,\gamma'}(y_1^n)| - H(X|Y)\right| \le \eta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma)\,. \quad (4.31)$$

Lemma 4.10 ([67, Lemma 2.5]). *For any conditional type Q of x_1^n given y_1^n,*

$$|\mathcal{T}_Q(y_1^n)| \le 2^{nH(Q|P_{y_1^n})}\,. \quad (4.32)$$

Lemma 4.11 ([48, Lemma 2]). *For (α_H, β_H) given in Lemma 4.1 and any set $\mathcal{T} \neq \emptyset$,*

$$P_{HV_1^k}[\mathcal{T} \cap \mathcal{C}_H(V_1^k) = \emptyset] \le \alpha_H - 1 + \frac{|\mathrm{Im}\mathcal{H}|(\beta_H + 1)}{|\mathcal{T}|}\,. \quad (4.33)$$

Lemma 4.12 ([48, Lemma 3]). *Let $\mathcal{T} \subset \mathcal{X}^n$ and $x_1^n \notin \mathcal{T}$ be arbitrary. For (α_H, β_H) given in Lemma 4.1,*

$$P_{HV_1^k}[x_1^n \in \mathcal{C}_H(V_1^k) \text{ and } \mathcal{C}_H(V_1^k) \cap \mathcal{T} \neq \emptyset] \le \frac{|\mathcal{T}|\alpha_H}{|\mathrm{Im}\mathcal{H}|^2} + \frac{\beta_H}{|\mathrm{Im}\mathcal{H}|}\,. \quad (4.34)$$

Lemma 4.13 ([48, Lemma 8]). *For any $\mathcal{T} \subset \mathcal{X}^k$,*

$$\mathrm{E}_{HV_1^k}[|\mathcal{T} \cap \mathcal{C}_H(V_1^k)|] \leq \frac{|\mathcal{T}|}{|\mathrm{Im}\mathcal{H}|}\,. \tag{4.35}$$

Lemma 4.14. *Let $\mathcal{T} \neq \emptyset$ be arbitrary. For (α_H, β_H) given in Lemma 4.1 and any $r < 1/|\mathrm{Im}\mathcal{H}|$,*

$$P_{HV_1^k}[|\mathcal{T} \cap \mathcal{C}_H(V_1^k)| \leq r|\mathcal{T}|] \leq \frac{1}{\left(\frac{1}{|\mathrm{Im}\mathcal{H}|} - r\right)^2}\left(\frac{1+\beta_H}{|\mathcal{T}||\mathrm{Im}\mathcal{H}|} + \frac{\alpha_H - 1}{|\mathrm{Im}\mathcal{H}|^2}\right). \tag{4.36}$$

The proof is very similar to that of [48, Lemma 2], in which $r|\mathcal{T}|$ is fixed to one.

Proof. First we have

$$\begin{aligned}
\mathrm{E}_{HV_1^k}[|\mathcal{T} \cap \mathcal{C}_H(V_1^k)|^2] &= \mathrm{E}_{HV_1^k}\left[\left(\sum_{x_1^n \in \mathcal{T}} \mathbb{1}\left[x_1^n H = V_1^k\right]\right)^2\right] \\
&= \sum_{x_1^n \in \mathcal{T}} \sum_{\tilde{x}_1^n \in \mathcal{T}} \mathrm{E}_{HV_1^k}\left[\mathbb{1}\left[x_1^n H = V_1^k,\, \tilde{x}_1^n H = V_1^k\right]\right] \\
&= \sum_{x_1^n \in \mathcal{T}} \sum_{\tilde{x}_1^n \in \mathcal{T}} \mathrm{E}_H\left[\mathbb{1}[x_1^n H = \tilde{x}_1^n H]\, P_{V_1^k}[\tilde{x}_1^n H = V_1^k]\right] \\
&= \frac{1}{|\mathrm{Im}\mathcal{H}|} \sum_{x_1^n \in \mathcal{T}} \sum_{\tilde{x}_1^n \in \mathcal{T}} P_H\left[x_1^n H = \tilde{x}_1^n H\right] \\
&\leq \frac{1}{|\mathrm{Im}\mathcal{H}|}\left(|\mathcal{T}| + \frac{|\mathcal{T}|^2 \alpha_H}{|\mathrm{Im}\mathcal{H}|} + |\mathcal{T}|\beta_H\right),
\end{aligned} \tag{4.37}$$

where the inequality follows from (4.9). From Lemma 4.13 and (4.37), the variance of $|\mathcal{T} \cap \mathcal{C}_H(V_1^k)|$ over (H, V_1^k) is bounded as

$$\begin{aligned}
\mathrm{Var}[|\mathcal{T} \cap \mathcal{C}_H(V_1^k)|] &= \mathrm{E}[|\mathcal{T} \cap \mathcal{C}_H(V_1^k)|^2] - \left(\mathrm{E}[|\mathcal{T} \cap \mathcal{C}_H(V_1^k)|]\right)^2 \\
&\leq \frac{1}{|\mathrm{Im}\mathcal{H}|}\left(|\mathcal{T}| + \frac{|\mathcal{T}|^2 \alpha_H}{|\mathrm{Im}\mathcal{H}|} + |\mathcal{T}|\beta_H\right) - \frac{|\mathcal{T}|^2}{|\mathrm{Im}\mathcal{H}|^2} \\
&= \frac{|\mathcal{T}|(1+\beta_H)}{|\mathrm{Im}\mathcal{H}|} + \frac{|\mathcal{T}|^2(\alpha_H - 1)}{|\mathrm{Im}\mathcal{H}|^2}\,.
\end{aligned} \tag{4.38}$$

Finally we obtain from Chebyshev's inequality (see, e.g., [68, p. 33]) that

$$\begin{aligned}
&P_{HV_1^k}[|\mathcal{T} \cap \mathcal{C}_H(V_1^k)| \le r|\mathcal{T}|] \\
&\le \frac{1}{\left(\frac{|\mathcal{T}|}{|\mathrm{Im}\mathcal{H}|} - r|\mathcal{T}|\right)^2} \left(\frac{|\mathcal{T}|(1+\beta_H)}{|\mathrm{Im}\mathcal{H}|} + \frac{|\mathcal{T}|^2(\alpha_H - 1)}{|\mathrm{Im}\mathcal{H}|^2} \right) \\
&= \frac{1}{\left(\frac{1}{|\mathrm{Im}\mathcal{H}|} - r\right)^2} \left(\frac{1+\beta_H}{|\mathcal{T}||\mathrm{Im}\mathcal{H}|} + \frac{\alpha_H - 1}{|\mathrm{Im}\mathcal{H}|^2} \right) . \qquad (4.39)
\end{aligned}$$

□

4.4.1 Lossy Source Coding

Define sets $\mathcal{S}_1$ and $\mathcal{S}_2$ as

$$\begin{aligned}
\mathcal{S}_1 &\equiv \mathcal{T}_{Y,\gamma} , \\
\mathcal{S}_2 &\equiv \{y_1^n : \hat{x}_1^n(y_1^n) \in \mathcal{T}_{X|Y,4\delta}(y_1^n)\} . \qquad (4.40)
\end{aligned}$$

First we prove the following lemma.

Lemma 4.15. *If $H(X|Y) - 2\delta < k/n < H(X|Y) - \delta$ then it holds for $\gamma > 0$ sufficiently small with respect to $\delta > 0$ that*

$$\lim_{n\to\infty} \mathrm{E}_{HV_1^k}[P_{Y_1^n}(\mathcal{S}_1^c \cup \mathcal{S}_2^c)] = 0 , \qquad (4.41)$$

where $\mathcal{S}^c$ represents the complement set of $\mathcal{S}$.

Proof. First we have

$$\mathrm{E}_{HV_1^k}[P_{Y_1^n}(\mathcal{S}_1^c \cup \mathcal{S}_2^c)] = \mathrm{E}_{HV_1^k}[P_{Y_1^n}(\mathcal{S}_1^c)] + \mathrm{E}_{HV_1^k}[P_{Y_1^n}(\mathcal{S}_1 \cap \mathcal{S}_2^c)] . \qquad (4.42)$$

For the first term of the right-hand side of (4.42) we obtain from Lemma 4.8 that

$$\begin{aligned}
\mathrm{E}_{HV_1^k}[P_{Y_1^n}(\mathcal{S}_1^c)] &= P_{Y_1^n}(\mathcal{S}_1^c) \\
&\le 2^{-n(\gamma - \lambda_{\mathcal{Y}})} \\
&\to 0 \quad (\text{as } n \to \infty) . \qquad (4.43)
\end{aligned}$$

For the second term we have

$$\mathrm{E}_{HV_1^k}[P_{Y_1^n}(\mathcal{S}_1 \cap \mathcal{S}_2^c)] = \sum_{y_1^n \in \mathcal{T}_{Y,\gamma}} P_{Y_1^n}(y_1^n) P_{HV_1^k}[\hat{x}_1^n(y_1^n, H, V_1^k) \notin \mathcal{T}_{X|Y,4\delta}(y_1^n)] . \qquad (4.44)$$

Now we show for any $y_1^n \in \mathcal{T}_{Y,\gamma}$ that

$$\begin{aligned}&\hat{x}_1^n(y_1^n, H, V_1^k) \notin \mathcal{T}_{X|Y,4\delta}(y_1^n)\\ &\qquad \Rightarrow \{\mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X|Y,\gamma}(y_1^n) = \emptyset\} \cup \{\mathcal{C}_H(V_1^k) \cap \tilde{\mathcal{T}}_\delta(y_1^n) \neq \emptyset\}\,, \end{aligned} \tag{4.45}$$

or equivalently

$$\{\hat{x}_1^n(y_1^n, H, V_1^k) \notin \mathcal{T}_{X|Y,4\delta}(y_1^n)\} \cap \{\mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X|Y,\gamma}(y_1^n) \neq \emptyset\} \tag{4.46}$$

$$\Rightarrow \{\mathcal{C}_H(V_1^k) \cap \tilde{\mathcal{T}}_\delta(y_1^n) \neq \emptyset\}\,, \tag{4.47}$$

where

$$\tilde{\mathcal{T}}_\delta(y_1^n) \equiv \left\{x_1^n : H(P_{x_1^n|y_1^n}|P_{y_1^n}) \leq \frac{k}{n} - \delta\right\}\,. \tag{4.48}$$

Assume that the condition (4.46) holds. Recall (4.11), i.e., $\hat{x}_1^n = \hat{x}_1^n(y_1^n, H, V_1^k) = \mathrm{argmax}_{x_1^n \in \mathcal{C}_H(v_1^k)} P_{X_1^n|Y_1^n}(x_1^n|y_1^n)$. Since $\gamma < 4\delta$, we have $\mathcal{T}_{X|Y,\gamma} \subset \mathcal{T}_{X|Y,4\delta}$ and therefore (4.46) means that

$$\exists \tilde{x}_1^n \in \mathcal{T}_{X|Y,\gamma} : P_{X_1^n|Y_1^n}(\hat{x}_1^n|y_1^n) \geq P_{X_1^n|Y_1^n}(\tilde{x}_1^n|y_1^n)\,. \tag{4.49}$$

For this $\tilde{x}_1^n$, we obtain from Lemma 4.7 and $H(X|Y) - 2\delta < k/n$ that

$$\begin{aligned}\frac{1}{n}\log\frac{1}{P_{X_1^n|Y_1^n}(\tilde{x}_1^n|y_1^n)} &\leq H(X|Y) + \zeta_{\mathcal{X}|\mathcal{Y}}(\gamma|\gamma)\\ &< \frac{k}{n} + 2\delta + \zeta_{\mathcal{X}|\mathcal{Y}}(\gamma|\gamma)\,. \end{aligned} \tag{4.50}$$

On the other hand, we obtain from Lemma 4.5 and $\hat{x}_1^n \notin \mathcal{T}_{X|Y,4\delta}(y_1^n)$ that

$$\frac{1}{n}\log\frac{1}{P_{X_1^n|Y_1^n}(\hat{x}_1^n|y_1^n)} \geq H(P_{\hat{x}_1^n|y_1^n}|P_{y_1^n}) + 4\delta\,. \tag{4.51}$$

Combining (4.49), (4.50) and (4.51), we obtain

$$\begin{aligned}H(P_{\hat{x}_1^n|y_1^n}|P_{y_1^n}) &\leq \frac{k}{n} - (2\delta - \zeta_{\mathcal{X}|\mathcal{Y}}(\gamma|\gamma))\\ &\leq \frac{k}{n} - \delta\,, \end{aligned} \tag{4.52}$$

which means that (4.47), and hence (4.45), holds.

Now we evaluate (4.44) as

$$\begin{aligned}\mathrm{E}_{HV_1^k}[P_X(\mathcal{S}_1 \cap \mathcal{S}_2^c)] \leq &\sum_{y_1^n \in \mathcal{T}_{Y,\gamma}} P_{Y_1^n}(y_1^n) P_{HV_1^k}[\mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X|Y,\gamma}(y_1^n) = \emptyset]\\ &+ \sum_{y_1^n \in \mathcal{T}_{Y,\gamma}} P_{Y_1^n}(y_1^n) P_{HV_1^k}[\mathcal{C}_H(V_1^k) \cap \tilde{\mathcal{T}}_\delta(y_1^n) \neq \emptyset]\,. \end{aligned} \tag{4.53}$$

From Lemmas 4.9 and 4.11 and $k/n < H(X|Y) - \delta$, the first term of (4.53) is bounded as

$$\begin{aligned}
&\sum_{y_1^n \in \mathcal{T}_{Y,\gamma}} P_{Y_1^n}(y_1^n) P_{HV_1^k}[\mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X|Y,\gamma}(y_1^n) = \emptyset] \\
&\quad \le \sum_{y_1^n \in \mathcal{T}_{Y,\gamma}} P_{Y_1^n}(y_1^n) \left(\alpha_H - 1 + \frac{2^k(\beta_H + 1)}{2^{n(H(X|Y) - \eta_{\mathcal{X}|\mathcal{Y}}(\gamma|\gamma))}} \right) \\
&\quad \le \alpha_H - 1 + \frac{2^k(\beta_H + 1)}{2^{n(H(X|Y) - \eta_{\mathcal{X}|\mathcal{Y}}(\gamma|\gamma))}} \\
&\quad \le \alpha_H - 1 + (\beta_H + 1) 2^{-n(\delta - \eta_{\mathcal{X}|\mathcal{Y}}(\gamma|\gamma))} \\
&\quad \to 0 \quad (\text{as } n \to \infty) . \qquad (4.54)
\end{aligned}$$

For the second term of (4.53), we bound $|\tilde{\mathcal{T}}_\delta(y_1^n)|$ from Lemma 4.10 as

$$\begin{aligned}
|\tilde{\mathcal{T}}_\delta(y_1^n)| &= \sum_{\tilde{P}: H(\tilde{P}|P_{y_1^n}) \le \frac{k}{n} - \delta} |\mathcal{T}_{\tilde{P}}(y_1^n)| \\
&\le \sum_{\tilde{P}: H(\tilde{P}|P_{y_1^n}) \le \frac{k}{n} - \delta} 2^{k - n\delta} , \qquad (4.55)
\end{aligned}$$

where $\tilde{P}$ is a conditional type and $\mathcal{T}_{\tilde{P}}(y_1^n) \subset \mathcal{X}^n$ is the set of x_1^n such that the conditional type of x_1^n given y_1^n is $\tilde{P}$. Since there are at most $(n+1)^{|\mathcal{X}||\mathcal{Y}|}$ conditional types, we have

$$|\tilde{\mathcal{T}}_\delta(y_1^n)| \le (n+1)^{|\mathcal{X}||\mathcal{Y}|} 2^{k - n\delta} . \qquad (4.56)$$

Then it holds from Lemma 4.13 that

$$\begin{aligned}
&\sum_{y_1^n \in \mathcal{T}_{Y,\gamma}} P_{Y_1^n}(y_1^n) P_{HV_1^k}[\mathcal{C}_H(V_1^k) \cap \tilde{\mathcal{T}}_\delta(y_1^n) \ne \emptyset] \\
&\quad \le \sum_{y_1^n \in \mathcal{T}_{Y,\gamma}} P_{Y_1^n}(y_1^n) \mathrm{E}_{HV_1^k}[|\mathcal{C}_H(V_1^k) \cap \tilde{\mathcal{T}}_\delta(y_1^n)|] \\
&\quad \le \sum_{y_1^n \in \mathcal{T}_{Y,\gamma}} P_{Y_1^n}(y_1^n) \frac{(n+1)^{|\mathcal{X}||\mathcal{Y}|} 2^{k - n\delta}}{2^{k-1}} \\
&\quad \le \frac{(n+1)^{|\mathcal{X}||\mathcal{Y}|} 2^{k - n\delta}}{2^{k-1}} \\
&\quad \to 0 \quad (\text{as } n \to \infty) . \qquad (4.57)
\end{aligned}$$

From (4.53), (4.54) and (4.57) we obtain

$$\lim_{n \to \infty} \mathrm{E}_{HV_1^k}[P_{Y_1^n}(\mathcal{S}_1 \cap \mathcal{S}_2^c)] = 0 \qquad (4.58)$$

and the lemma is proved. □

Now we prove Theorem 4.2 based on Lemma 4.15.

Proof of Theorem 4.2. Since an average convergence implies a probability convergence, it suffices to show that

$$\lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[P_{Y_1^n}\left[\frac{d^n(Y_1^n,\psi(\varphi(Y_1^n)))}{n} > D+\epsilon\right]\right] = 0 \tag{4.59}$$

and

$$\lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[P_{Y_1^n}\left[\frac{l(Y_1^n)}{n} > R(D)+\epsilon\right]\right] = 0\,. \tag{4.60}$$

From Lemma 4.6, if $y_1^n \in \mathcal{S}_1 \cap \mathcal{S}_2$ then $(\hat{x}_1^n, y_1^n) \in \mathcal{T}_{XY,\gamma+4\delta} \subset \mathcal{T}_{XY,5\delta}$. Since the variational distance is expressed as

$$\begin{aligned}\|P_1 - P_2\| &= \sum_{x:P_1(x)>P_2(x)} (P_1(x) - P_2(x)) \\ &= \frac{1}{2}\sum_x |P_1(x) - P_2(x)|\,,\end{aligned} \tag{4.61}$$

if $(\hat{x}_1^n, y_1^n) \in \mathcal{T}_{XY,5\delta}$ then

$$\begin{aligned}\frac{d^n(\hat{x}_1^n, y_1^n)}{n} &= \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} P_{\hat{x}_1^n y_1^n}(x,y)d(y,x) \\ &\le \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} P_{XY}(x,y)d(y,x) + d_{\max}\|P_{XY} - P_{\hat{x}_1^n y_1^n}\| \\ &\le \mathrm{E}_{XY}[d(Y,X)] + d_{\max}\sqrt{\frac{(\ln 2)D(P_{\hat{x}_1^n y_1^n}\|P_{XY})}{2}} \\ &< D + d_{\max}\sqrt{\frac{5(\ln 2)\delta}{2}} \\ &\le D+\epsilon\,.\end{aligned} \tag{4.62}$$

where the second inequality follows from Pinsker's inequality (3.61) and the last inequality follows from (4.14). We obtain (4.59) from

$$\begin{aligned}&\lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[P_{Y_1^n}\left[\frac{d^n(Y_1^n,\psi(\varphi(Y_1^n)))}{n} > D+\epsilon\right]\right] \\ &\quad\le \lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[P_{Y_1^n}\left[Y_1^n \notin \mathcal{S}_1^c \cup \mathcal{S}_2^c\right]\right] \\ &\quad= 0\,.\end{aligned} \tag{4.63}$$

Next we prove (4.60). From the property of the arithmetic code in (2.29), the code length of $\hat{x}_1^n$ satisfying $\hat{x}_1^n H = v_1^k$ is bounded as

$$\begin{aligned}
L(\hat{x}_1^n) &\le 2 - \sum_{i=1}^{n} \log \hat{P}_i(\hat{x}_i|\hat{x}_1^{i-1}) \\
&= 2 - \sum_{i=1}^{n} \log P_{X_1^n}[X_i = \hat{x}_i|X_1^{i-1} = \hat{x}_1^{i-1}, X_1^n H = v_1^k] \\
&= 2 - \log P_{X_1^n}[X_1^n = \hat{x}_1^n|X_1^n H = v_1^k] \\
&= 2 - \log P_{X_1^n}[X_1^n = \hat{x}_1^n, X_1^n H = v_1^k] + \log P_{X_1^n}[X_1^n H = v_1^k] \\
&= 2 - \log P_{X_1^n}[X_1^n = \hat{x}_1^n] + \log P_{X_1^n}[X_1^n H = v_1^k]\,. \qquad (4.64)
\end{aligned}$$

Now we show that

$$\{y_1^n \in \mathcal{S}_1 \cap \mathcal{S}_2\} \cap \{P_{X_1^n}[X_1^n H = v_1^k] \le 2^{-2-k+n\delta}\} \qquad (4.65)$$

$$\Rightarrow \left\{\frac{l(y_1^n)}{n} \le R(D) + \epsilon\right\}. \qquad (4.66)$$

Assume that the condition (4.65) holds. Recall that $(\hat{x}_1^n, y_1^n) \in \mathcal{T}_{XY,5\delta}$ on this condition and $\hat{x}_1^n(y_1^n) \in \mathcal{T}_{X,5\delta}$ from Lemma 4.6. We obtain from Lemma 4.7, $H(X|Y) - 2\delta \le k/n$ and $R(D) = I(X;Y)$ that

$$\begin{aligned}
-\log P_{X_1^n}[X_1^n = \hat{x}_1^n] &\le n(H(X) + \zeta_{\mathcal{X}}(5\delta)) \\
&= n(H(X|Y) + I(X;Y) + \zeta_{\mathcal{X}}(5\delta)) \\
&\le k + n(R(D) + 2\delta + \zeta_{\mathcal{X}}(5\delta))\,. \qquad (4.67)
\end{aligned}$$

Combining (4.64) and (4.65), we have (4.66) since

$$\begin{aligned}
\frac{L(\hat{x}_1^n)}{n} &\le R(D) + 3\delta + \zeta_{\mathcal{X}}(5\delta) \\
&\le R(D) + \epsilon\,, \qquad (4.68)
\end{aligned}$$

where the last inequality holds from (4.14) and (4.25).

On the other hand, we obtain from Markov's inequality (see, e.g., [51, p. 64]) that

$$\begin{aligned}
&P_{HV_1^k}\left[P_{X_1^n}[X_1^n H = V_1^k] \ge 2^{-2-k+n\delta}\right] \\
&\quad \le 2^{2+k-n\delta} \mathrm{E}_{HV_1^k}[P_{X_1^n}[X_1^n H = V_1^k]] \\
&\quad = 2^{2+k-n\delta} \sum_H \sum_{v_1^k} P_H(H) P_{V_1^k}(v_1^k) \sum_{x_1^n} P_{X_1^n}(x_1^n) \mathbb{1}[x_1^n H = v_1^k] \\
&\quad = 2^{2+k-n\delta} \sum_H P_H(H) \sum_{x_1^n} P_{X_1^n}(x_1^n) \sum_{v_1^k} P_{V_1^k}(v_1^k) \mathbb{1}[x_1^n H = v_1^k]
\end{aligned}$$

$$\leq 2^{2+k-n\delta} \sum_{H} P_H(H) \sum_{x_1^n} P_{X_1^n}(x_1^n) \frac{1}{|\mathrm{Im}\mathcal{H}|}$$
$$\leq \frac{2^{2+k-n\delta}}{2^{k-1}} \to 0 \quad (\text{as } n \to \infty)\,, \tag{4.69}$$

where $\mathbb{1}[\cdot]$ denotes the indicator function. Finally we obtain from Lemma 4.15, (4.66) and (4.69) that

$$\begin{aligned}
&\lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[P_{Y_1^n}\left[\frac{l(Y_1^n)}{n} > R(D)+\epsilon\right]\right] \\
&\quad \leq \lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[P_{Y_1^n}\left[Y_1^n \in \mathcal{S}_1^c \cup \mathcal{S}_2^c\right]\right] \\
&\qquad + \lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[P_{Y_1^n}\left[P_{X_1^n}[X_1^n H = V_1^k] \geq 2^{-2-k+n\delta}\right]\right] \\
&\quad = 0 + \lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[P_{X_1^n}[X_1^n H = V_1^k] \geq 2^{-2-k+n\delta}\right] \\
&\quad = 0
\end{aligned} \tag{4.70}$$

and (4.60) is proved. □

Proof of Corollary 4.3. Let $\epsilon' > \epsilon$. Since

$$\begin{aligned}
&\mathrm{E}_{Y_1^n}\left[\frac{d^n(Y_1^n, \psi(\varphi(Y_1^n)))}{n}\right] \\
&\qquad \leq D + \epsilon + d_{\max} P_{Y_1^n}\left[\frac{d^n(Y_1^n, \psi(\varphi(Y_1^n)))}{n} > D+\epsilon\right],
\end{aligned} \tag{4.71}$$

we have (4.17) from (4.15) and

$$\begin{aligned}
&P_{HV_1^k}\left[\mathrm{E}_{Y_1^n}\left[\frac{d^n(Y_1^n, \psi(\varphi(Y_1^n)))}{n}\right] > D+\epsilon'\right] \\
&\quad \leq P_{HV_1^k}\left[P_{Y_1^n}\left[\frac{d^n(Y_1^n, \psi(\varphi(Y_1^n)))}{n} > D+\epsilon\right] > \frac{\epsilon'-\epsilon}{d_{\max}}\right] \\
&\quad \to 0 \quad (\text{as } n \to \infty)\,.
\end{aligned} \tag{4.72}$$

Next we show (4.18). From (4.64), if $P_{X_1^n}[X_1^n H = v_1^k] \leq 2^{-2-k+n\delta}$ then

$$\begin{aligned}
\frac{L(\hat{x}_1^n)}{n} &\leq \log\frac{1}{P_{\min}} - \frac{k}{n} + \delta \\
&\leq \log\frac{1}{P_{\min}} + \delta\,,
\end{aligned} \tag{4.73}$$

where $P_{\min} = \min_{x\in\mathcal{X}} P_X(x) > 0$, and therefore

$$\mathrm{E}_{Y_1^n}\left[\frac{l(Y_1^n)}{n}\right] \leq R(D) + \epsilon + \left(\log\frac{1}{P_{\min}} + \delta\right) P_{Y_1^n}\left[\frac{l(Y_1^n)}{n} > R(D)+\epsilon\right]. \tag{4.74}$$

Then we obtain from (4.16) that

$$\begin{aligned}
&P_{HV_1^k}\left[\mathrm{E}_{Y_1^n}\left[\frac{l(Y_1^n)}{n}\right] > R(D)+\epsilon'\right] \\
&\quad \le P_{HV_1^k}\left[P_{X_1^n}[X_1^n H = V_1^k] > 2^{-2-k+n\delta}\right] \\
&\qquad + P_{HV_1^k}\left[P_{Y_1^n}\left[\frac{l(Y_1^n)}{n} > R(D)+\epsilon\right] > \frac{\epsilon'-\epsilon}{\left(\log\frac{1}{P_{\min}}+\delta\right)}\right] \\
&\quad \to 0 \quad (\text{as } n\to\infty)
\end{aligned} \tag{4.75}$$

and the proof is completed. □

4.4.2 Channel Coding

Define a set $\mathcal{T}_{HV,\gamma}$ by

$$\mathcal{T}_{HV,\gamma} \equiv \{(H, v_1^k) : P_{X_1^n}[X_1^n \in \mathcal{C}_H(v_1^k)] > 2^{-k-n\gamma}\}\,. \tag{4.76}$$

First we prove the following lemma.

Lemma 4.16. *Let $\delta, \gamma > 0$ be arbitrary. If $k/n < H(X) - \delta$ then*

$$\lim_{n\to\infty} P_{HV_1^k}[(H, V_1^k) \notin \mathcal{T}_{HV,\gamma}] = 0\,. \tag{4.77}$$

Proof. From Lemma 4.7, it holds for all $x_1^n \in \mathcal{T}_{X,\gamma'}$ that

$$P_{X_1^n}(x_1^n) \ge 2^{-n(H(X)+\zeta_{\mathcal{X}}(\gamma'))}\,. \tag{4.78}$$

Therefore $(H, V_1^k) \notin \mathcal{T}_{HV,\gamma}$ implies that

$$\begin{aligned}
|\mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X,\gamma'}| &\le 2^{n(H(X)-\gamma+\zeta_{\mathcal{X}}(\gamma'))-k} \\
&\le |\mathcal{T}_{X,\gamma'}| 2^{-n(\gamma-\eta_{\mathcal{X}}(\gamma')-\zeta_{\mathcal{X}}(\gamma'))-k}
\end{aligned} \tag{4.79}$$

where the last inequality follows from Lemma 4.9. Then we obtain from Lemma 4.14 that

$$\begin{aligned}
&P_{HV_1^k}[(H, V_1^k) \notin \mathcal{T}_{HV,\gamma}] \\
&\le P_{HV_1^k}[|\mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X,\gamma'}| \le |\mathcal{T}_{X,\gamma'}| 2^{-n(\gamma-\eta_{\mathcal{X}}(\gamma')-\zeta_{\mathcal{X}}(\gamma'))-k}] \\
&\le \frac{1}{\left(1-2^{-n(\gamma-\eta_{\mathcal{X}}(\gamma')-\zeta_{\mathcal{X}}(\gamma'))}\right)^2}\left(\frac{2^k(1+\beta_H)}{|\mathcal{T}_{X,\gamma'}|} + (\alpha_H - 1)\right) \\
&\le \frac{1}{\left(1-2^{-n(\gamma-\eta_{\mathcal{X}}(\gamma')-\zeta_{\mathcal{X}}(\gamma'))}\right)^2}\left(\frac{2^{n(H(X)-\delta)}(1+\beta_H)}{2^{n(H(X)-\eta_{\mathcal{X}}(\gamma'))}} + (\alpha_H - 1)\right),
\end{aligned} \tag{4.80}$$

where the last inequality follows from Lemma 4.9. We complete the proof by letting $n \to \infty$ for γ' sufficiently small with respect to δ and γ. $\square$

Remark 4.2. Closer inspection of (4.80) reveals that Lemma 4.16 is valid even if $\gamma > 0$ depends on n as long as $\lim_n n\gamma = +\infty$. We set γ depending on n in the proof of Lemma 4.17 below.

Now we formulate properties of the inner code generated from distribution

$$\hat{P}(x_1^n) = \frac{P_{X_1^n}(x_1^n)\mathbb{1}\left[x_1^n \in \mathcal{C}_H(v_1^k)\right]}{P_{X_1^n}[X_1^n \in \mathcal{C}_H(v_1^k)]} . \tag{4.81}$$

Let define the decoding error probability for an inner codeword $x_1^n \in \mathcal{C}$ by

$$P_{\mathrm{e,in}}(x_1^n) \equiv \sum_{y_1^n} P_{Y_1^n|X_1^n}(y_1^n|x_1^n)\,\mathbb{1}\big[x_1^n \neq \hat{x}_1^n(y_1^n, H, v_1^k)\big] . \tag{4.82}$$

The average decoding error probability of the inner code over codeword distribution $\hat{P}$ is denoted by

$$P_{\mathrm{e,in}} \equiv \mathrm{E}_{\hat{P}}[P_{\mathrm{e,in}}(X_1^n)] = \sum_{x_1^n \in \mathcal{C}} \hat{P}(x_1^n) P_{\mathrm{e,in}}(x_1^n) . \tag{4.83}$$

We prove Theorem 4.4 by combining Lemmas 4.18 and 4.17 given in the following, which concern the inner and the outer code, respectively.

Lemma 4.17. *Assume that the ensemble of H satisfies the hash property. For any fixed $\epsilon > 0$, take δ such that*

$$0 < \delta < \min\left\{\frac{\epsilon}{2}, \frac{H(X) - H(X|Y)}{3}\right\} . \tag{4.84}$$

Then, for all sufficiently large (n, k) satisfying $H(X|Y) + \delta \leq k/n \leq H(X|Y) + 2\delta$, it holds that

$$\lim_{n\to\infty} P_{HV_1^k}\left[P_{\mathrm{e,in}} \geq \epsilon\right] = 0 , \tag{4.85}$$

$$\lim_{n\to\infty} P_{HV_1^k}\left[\frac{H(\hat{P})}{n} \leq I(X;Y) - \epsilon\right] = 0 , \tag{4.86}$$

$$\lim_{n\to\infty} P_{HV_1^k}\left[\max_{x_1^n \in \mathcal{C}} \hat{P}(x_1^n) \geq \frac{1}{2}\right] = 0 . \tag{4.87}$$

Lemma 4.18. *The average coding rate and the decoding error probability of the proposed scheme is given by*

$$R \geq \frac{H(\hat{P})}{n} \frac{1}{1 + \frac{n-k+1+\frac{1}{1-p_{\max}}}{m}}\,, \tag{4.88}$$

$$P_{\mathrm{e,out}} \leq \frac{m}{H(\hat{P})} \left(1 + \frac{n-k+1+\frac{1}{1-p_{\max}}}{m}\right) P_{\mathrm{e,in}}\,, \tag{4.89}$$

where $p_{\max} = \max_{x_1^n \in \mathcal{C}} \hat{P}(x_1^n)$.

Proof of Lemma 4.17. From Lemma 4.16 and $k/n < H(X|Y) + 2\delta < H(X) - (H(X) - H(X|Y))/3$, it suffices to prove that

$$\lim_{n\to\infty} P_{HV_1^k}\left[(H, V_1^k) \in \mathcal{T}_{HV,\gamma} \text{ and } P_{\mathrm{e,in}} \geq \epsilon\right] = 0\,, \tag{4.90}$$

$$\lim_{n\to\infty} P_{HV_1^k}\left[(H, V_1^k) \in \mathcal{T}_{HV,\gamma} \text{ and } \frac{H(\hat{P})}{n} \leq I(X;Y) - \epsilon\right] = 0\,, \tag{4.91}$$

$$\lim_{n\to\infty} P_{HV_1^k}\left[(H, V_1^k) \in \mathcal{T}_{HV,\gamma} \text{ and } \max_{x_1^n \in \mathcal{C}} \hat{P}(x_1^n) \geq \frac{1}{2}\right] = 0 \tag{4.92}$$

instead of (4.85)–(4.87).

First we prove

$$\lim_{n\to\infty} \mathrm{E}_{HV_1^k}\left[\mathbb{1}\left[(H, V_1^k) \in \mathcal{T}_{HV,\gamma}\right] \cdot P_{\mathrm{e,in}}\right] = 0\,, \tag{4.93}$$

which means (4.90) since an average convergence implies the probability convergence. This expectation is evaluated as

$$\begin{aligned}
&\mathrm{E}_{HV_1^k}\left[\mathbb{1}[(H, V_1^k) \in \mathcal{T}_{HV,\gamma}] \cdot P_{\mathrm{e,in}}\right] \\
&= \mathrm{E}_{HV_1^k}\left[\mathbb{1}[(H, V_1^k) \in \mathcal{T}_{HV,\gamma}] \cdot \sum_{x_1^n} \frac{P_{X_1^n}(x_1^n)\mathbb{1}\left[x_1^n H = V_1^k\right]}{P_{X_1^n}[X_1^n H = V_1^k]} P_{\mathrm{e,in}}(x_1^n)\right] \\
&\leq 2^{k+n\gamma} \sum_{x_1^n} P_{X_1^n}(x_1^n) \mathrm{E}_{HV_1^k}\left[\mathbb{1}\left[x_1^n H = V_1^k\right] \cdot P_{\mathrm{e,in}}(x_1^n)\right] \\
&= 2^{k+n\gamma} \sum_{x_1^n, y_1^n} P_{X_1^n Y_1^n}(x_1^n, y_1^n) P_{HV_1^k}\left[x_1^n H = V_1^k,\, x_1^n \neq \hat{x}_1^n(y_1^n, H, V_1^k)\right] \\
&\leq 2^{k+n\gamma} \sum_{\substack{y_1^n \in \mathcal{T}_{Y,\gamma'} \\ x_1^n \in \mathcal{T}_{X|Y,\gamma'}(y_1^n)}} P_{X_1^n Y_1^n}(x_1^n, y_1^n) P_{HV_1^k}\left[x_1^n H = V_1^k,\, x_1^n \neq \hat{x}_1^n(y_1^n, H, V_1^k)\right] \\
&\quad + 2^{k+n\gamma} \sum_{y_1^n \notin \mathcal{T}_{Y,\gamma'}} P_{Y_1^n}(y_1^n) P_{HV_1^k}\left[x_1^n H = V_1^k\right]
\end{aligned}$$

$$
\begin{aligned}
&+2^{k+n\gamma} \sum_{\substack{y_1^n \in \mathcal{T}_{Y,\gamma'} \\ x_1^n \notin \mathcal{T}_{X|Y,\gamma'}(y_1^n)}} P_{X_1^n Y_1^n}(x_1^n, y_1^n) P_{HV_1^k}\left[x_1^n H = V_1^k\right] \\
&\leq 2^{k+n\gamma} \sum_{\substack{y_1^n \in \mathcal{T}_{Y,\gamma'} \\ x_1^n \in \mathcal{T}_{X|Y,\gamma'}(y_1^n)}} P_{X_1^n Y_1^n}(x_1^n, y_1^n) P_{HV_1^k}\left[x_1^n H = V_1^k,\, x_1^n \neq \hat{x}_1^n(y_1^n, H, V_1^k)\right] \\
&+2^{1-n(\gamma'-\gamma-\lambda_{\mathcal{Y}})} + 2^{1-n(\gamma'-\gamma-\lambda_{\mathcal{X}\mathcal{Y}})}\,, \qquad (4.94)
\end{aligned}
$$

where the last inequality follows from Lemma 4.8 and $P_{HV_1^k}[x_1^n H = V_1^k] = 1/|\mathrm{Im}\mathcal{H}|$.

Now assume that

$$
y_1^n \in \mathcal{T}_{Y,\gamma'} \text{ and } x_1^n \in \mathcal{T}_{X|Y,\gamma'}(y_1^n)\,. \qquad (4.95)
$$

Then it holds from Lemma 4.7 that

$$
P_{X_1^n|Y_1^n}(x_1^n|y_1^n) \geq 2^{-n(H(X|Y)+\zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma'))}\,. \qquad (4.96)
$$

As a result, we obtain

$$
\begin{aligned}
\left\{x_1^n \neq \hat{x}_1^n(y_1^n, H, V_1^k)\right\} &= \left\{x_1^n \neq \operatorname*{argmax}_{\tilde{x}_1^n \in \mathcal{C}_H(V_1^k)} P_{X_1^n|Y_1^n}(\tilde{x}_1^n|y_1^n)\right\} \\
&\subset \left\{\mathcal{C}_H(V_1^k) \cap \mathcal{G}(y_1^n) \setminus \{x_1^n\} \neq \emptyset\right\}, \qquad (4.97)
\end{aligned}
$$

where

$$
\mathcal{G}(y_1^n) \equiv \left\{x_1^n : P_{X_1^n|Y_1^n}(x_1^n|y_1^n) \geq 2^{-n(H(X|Y)+\zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma'))}\right\}. \qquad (4.98)
$$

Note that it is easy to see that

$$
|\mathcal{G}(y_1^n)| \leq 2^{n(H(X|Y)+\zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma'))}\,. \qquad (4.99)
$$

Then it holds under condition (4.95) that

$$
\begin{aligned}
&P_{HV_1^k}\left[\{x_1^n \in \mathcal{C}_H(V_1^k)\} \cap \left\{x_1^n \neq \hat{x}_1^n(y_1^n, H, V_1^k)\right\}\right] \\
&\quad \leq P_{HV_1^k}\left[\{x_1^n \in \mathcal{C}_H(V_1^k)\} \cap \{\mathcal{C}_H(V_1^k) \cap \mathcal{G}(y_1^n) \setminus \{x_1^n\} \neq \emptyset\}\right] \\
&\quad \leq \frac{2^{n(H(X|Y)+\zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma'))}\alpha_H}{|\mathrm{Im}\,\mathcal{H}|^2} + \frac{\beta_H}{|\mathrm{Im}\,\mathcal{H}|} \\
&\quad \leq \frac{2^{k-n(\delta-\zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma'))}\alpha_H}{2^{2k-2}} + \frac{\beta_H}{2^{k-1}}\,, \qquad (4.100)
\end{aligned}
$$

where the second inequality holds from Lemma 4.12.

From (4.94) and (4.100), we have

$$\begin{aligned}
&\mathrm{E}_{HV_1^k}\left[\mathbb{1}[(H,V_1^k)\in\mathcal{T}_{HV,\gamma}]\cdot P_{\mathrm{e,in}}\right]\\
&\le 2^{k+n\gamma}\sum_{\substack{y_1^n\in\mathcal{T}_{Y,\gamma'}\\ x_1^n\in\mathcal{T}_{X|Y,\gamma'}(y_1^n)}} P_{X_1^nY_1^n}(x_1^n,y_1^n)\left(\frac{2^{-n(\delta-\zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma'))}\alpha_H}{2^{k-2}}+\frac{\beta_H}{2^{k-1}}\right)\\
&\quad+2^{1-n(\gamma'-\gamma-\lambda_{\mathcal{X}})}+2^{1-n(\gamma'-\gamma-\lambda_{\mathcal{XY}})}\\
&\le 4\alpha_H 2^{-n(\delta-\gamma-\zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma'))}+2\beta_H 2^{n\gamma}+2^{1-n(\gamma'-\gamma-\lambda_{\mathcal{X}})}+2^{1-n(\gamma'-\gamma-\lambda_{\mathcal{XY}})}\,.
\end{aligned}\tag{4.101}$$

Now recall that Lemma 4.16 is valid if $\gamma=\gamma_n$ satisfies

$$\lim_{n\to\infty} n\gamma_n=+\infty\,.\tag{4.102}$$

When we set

$$\gamma_n=\min\left\{\frac{1}{2n}\log\frac{1}{\beta_H},\ \frac{\delta}{2},\ \frac{\gamma'}{2}\right\}\,,\tag{4.103}$$

(4.102) is satisfied and (4.101) is bounded as

$$\begin{aligned}
&\mathrm{E}_{HV_1^k}\left[\mathbb{1}[(H,V_1^k)\in\mathcal{T}_{HV,\gamma}]\cdot P_{\mathrm{e,in}}\right]\\
&\le 4\alpha_H 2^{-n(\delta/2-\zeta_{\mathcal{X}|\mathcal{Y}}(\gamma'|\gamma'))}+2\sqrt{\beta_H}+2^{1-n(\gamma'/2-\lambda_{\mathcal{X}})}+2^{1-n(\gamma'/2-\lambda_{\mathcal{XY}})}\,.
\end{aligned}\tag{4.104}$$

Then we obtain (4.93) by letting γ' sufficiently small with respect to δ.

Next we prove (4.91). Assume that $(H,V_1^k)\in\mathcal{T}_{HV,\gamma}$ and

$$\sum_{x_1^n\in\mathcal{C}_H(V_1^k)\cap\mathcal{T}^c_{X,2\gamma}}\frac{P_{X_1^n}(x_1^n)}{P_{X_1^n}[X_1^n\in\mathcal{C}_H(V_1^k)]}\le\gamma\,.\tag{4.105}$$

Then

$$\begin{aligned}
\frac{H(\hat{P})}{n}&=\frac{1}{n}\sum_{x_1^n\in\mathcal{C}_H(V_1^k)}\hat{P}(x_1^n)\log\frac{P_{X_1^n}[X_1^n\in\mathcal{C}_H(V_1^k)]}{P_{X_1^n}(x_1^n)}\\
&\overset{\text{(a)}}{\ge}-\frac{k}{n}-\gamma-\frac{1}{n}\sum_{x_1^n\in\mathcal{C}_H(V_1^k)}\hat{P}(x_1^n)\log P_{X_1^n}(x_1^n)\\
&=-\frac{k}{n}-\gamma+\frac{1}{n}\sum_{x_1^n\in\mathcal{C}_H(V_1^k)}\hat{P}(x_1^n)\cdot n(H(X)-\zeta_{\mathcal{X}}(2\gamma))\\
&\quad-\frac{1}{n}\sum_{x_1^n\in\mathcal{C}_H(V_1^k)}\hat{P}(x_1^n)(\log P_{X_1^n}(x_1^n)+n(H(X)-\zeta_{\mathcal{X}}(2\gamma))
\end{aligned}$$

$$
\begin{aligned}
&\overset{\text{(b)}}{\geq} -\frac{k}{n} - \gamma + H(X) - \zeta_{\mathcal{X}}(2\gamma) \\
&\quad -\frac{1}{n} \sum_{x_1^n \in \mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X,2\gamma}^c} \hat{P}(x_1^n)(\log P_{X_1^n}(x_1^n) + n(H(X) - \zeta_{\mathcal{X}}(2\gamma)) \\
&\overset{\text{(c)}}{\geq} -\frac{k}{n} - \gamma + H(X) - \zeta_{\mathcal{X}}(2\gamma) \\
&\quad -\frac{1}{n} \sum_{x_1^n \in \mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X,2\gamma}^c} \hat{P}(x_1^n) n(H(X) - \zeta_{\mathcal{X}}(2\gamma)) \\
&\overset{\text{(d)}}{\geq} -\frac{k}{n} - \gamma + H(X) - \zeta_{\mathcal{X}}(2\gamma) - \gamma(H(X) - \zeta_{\mathcal{X}}(2\gamma)) \\
&\overset{\text{(e)}}{\geq} I(X;Y) - 2\delta - \gamma - \zeta_{\mathcal{X}}(2\gamma) - \gamma(H(X) - \zeta_{\mathcal{X}}(2\gamma)) \,, \qquad (4.106)
\end{aligned}
$$

where the inequalities follows from

(a): $(H, V_1^k) \in \mathcal{T}_{HV,\gamma}$,
(b): $\log P_{X_1^n}(x_1^n) \leq -n(H(X) - \zeta_{\mathcal{X}}(2\gamma))$ for $x_1^n \in \mathcal{T}_{X,2\gamma}$ by Lemma 4.7,
(c): $P_{X_1^n}(x_1^n) \leq 1$,
(d): (4.105),
(e): $k/n < H(X|Y) + 2\delta$.

Since $2\delta < \epsilon$, we obtain

$$
\frac{H(\hat{P})}{n} > I(X;Y) - \epsilon \qquad (4.107)
$$

under assumptions $(H, V_1^k) \in \mathcal{T}_{HV,\gamma}$ and (4.105) by letting γ sufficiently small with respect to $\epsilon - 2\delta$.

Now the LHS of (4.91) is evaluated as

$$
\begin{aligned}
&P_{HV_1^k}\left[(H, V_1^k) \in \mathcal{T}_{HV,\gamma} \text{ and } \frac{H(\hat{P})}{n} \leq H(X;Y) - \epsilon\right] \\
&\leq P_{HV_1^k}\left[(H, V_1^k) \in \mathcal{T}_{HV,\gamma} \text{ and } \sum_{x_1^n \in \mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X,2\gamma}^c} \frac{P_{X_1^n}(x_1^n)}{P_{X_1^n}[X_1^n \in \mathcal{C}_H(V_1^k)]} \geq \gamma\right] \\
&\leq P_{HV_1^k}\left[2^{k+n\gamma} \sum_{x_1^n \in \mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X,2\gamma}^c} P_{X_1^n}(x_1^n) \geq \gamma\right]. \qquad (4.108)
\end{aligned}
$$

On the other hand, we have

$$
\mathrm{E}_{HV_1^k}\left[2^{k+n\gamma} \sum_{x_1^n \in \mathcal{C}_H(V_1^k) \cap \mathcal{T}_{X,2\gamma}^c} P_{X_1^n}(x_1^n)\right]
$$

$$
\begin{aligned}
&= 2^{k+n\gamma} \sum_{x_1^n \in \mathcal{T}_{X,2\gamma}^c} P_{X_1^n}(x_1^n) P_{HV_1^k}\left[x_1^n \in \mathcal{C}_H(V_1^k)\right] \\
&\le 2^{1+n\gamma} \sum_{x_1^n \in \mathcal{T}_{X,2\gamma}^c} P_{X_1^n}(x_1^n) \\
&\le 2^{1-n(\gamma-\lambda_{\mathcal{X}})} \\
&\to 0 \qquad (\text{as } n \to \infty)\,, \qquad (4.109)
\end{aligned}
$$

where the second inequality holds from Lemma 4.8. Then (4.108) also approaches 0 as $n \to \infty$ and (4.91) is proved.

Finally we prove (4.92). Assume that $(H, V_1^k) \in \mathcal{T}_{HV,\gamma}$ and $\max_{x_1^n \in \mathcal{C}_H(V_1^k)} \hat{P}(x_1^n) \ge 1/2$. Then there exists $\tilde{x}_1^n \in \mathcal{C}_H(V_1^k)$ such that

$$
P_{X_1^n}(\tilde{x}_1^n) \ge 2^{-k-n\gamma-1}\,, \qquad (4.110)
$$

which implies from Lemma 4.5 that

$$
H(P_{\tilde{x}_1^n}) + D(P_{\tilde{x}_1^n} \| P_X) \le k/n + \gamma + 1/n\,. \qquad (4.111)
$$

Now assume further that $D(P_{\tilde{x}_1^n} \| P_X) < 2\gamma$. Then, from Pinsker's inequality (3.61), we have $\|P_{\tilde{x}_1^n} - P_X\| < \sqrt{(\ln 2)\gamma}$. Since the entropy $H(P)$ is continuous with respect to the variational distance, we have $H(P_{\tilde{x}_1^n}) \ge H(P_X) - \gamma'$ for any fixed γ' by letting γ sufficiently small.

As a result, we have

$$
\begin{aligned}
H(P_{\tilde{x}_1^n}) + D(P_{\tilde{x}_1^n} \| P_X) &\ge H(P_{\tilde{x}_1^n}) \\
&\ge H(X) - \gamma' \\
&\ge H(X|Y) + 3\delta - \gamma' \\
&\ge k/n + \delta - \gamma'\,, \qquad (4.112)
\end{aligned}
$$

where the third and the last equalities follow from the assumption of the lemma. Eq. (4.112) contradicts with (4.111) if γ and γ' are sufficiently small with respect to δ. Then we have $D(P_{\tilde{x}_1^n} \| P_X) \ge 2\gamma$ and, from (4.111),

$$
H(P_{\tilde{x}_1^n}) \le k/n - \gamma + 1/n\,. \qquad (4.113)
$$

Thus, we see that

$$
\begin{aligned}
&\left\{(H, V) \in \mathcal{T}_{HV,\gamma} \text{ and } \max_{x_1^n \in \mathcal{C}} \hat{P}(x_1^n) \ge \frac{1}{2}\right\} \\
&\quad \subset \left\{\exists \tilde{x}_1^n \in \mathcal{C}_H(V_1^k),\ H(P_{\tilde{x}_1^n}) \le k/n - \gamma + 1/n\right\} \\
&\quad = \left\{\mathcal{C}_H(V_1^k) \cap \tilde{\mathcal{T}}_{X,\delta} \ne \emptyset\right\}\,, \qquad (4.114)
\end{aligned}
$$

where

$$\tilde{\mathcal{T}}_{X,\delta} \equiv \left\{ x_1^n : H(P_{x_1^n}) \leq \frac{k}{n} - \gamma - \frac{1}{n} \right\} . \tag{4.115}$$

Finally, the LHS of (4.92) is bounded as

$$P_{HV_1^k} \left[(H, V_1^k) \in \mathcal{T}_{HV,\gamma} \text{ and } \max_{x_1^n \in \mathcal{C}} \hat{P}(x_1^n) \geq \frac{1}{2} \right] \leq P_{HV_1^k}[\mathcal{C}_H(V_1^k) \cap \tilde{\mathcal{T}}_{X,\delta} \neq \emptyset] . \tag{4.116}$$

We can prove that RHS of (4.116) converges to 0 as $n \to \infty$ in the same way as (4.57). □

Proof of Lemma 4.18. First we evaluate the average coding rate. From Theorem 2.5, it holds that

$$\begin{aligned} \mathrm{E}_{U_1^m}[L(U_1^m)] &\leq \frac{m}{H(\hat{P})} + \frac{\log(2(2^{n-k}-1))}{H(\hat{P})} + \frac{h(p_{\max})}{(1-p_{\max})H(\hat{P})} \\ &\leq \frac{m}{H(\hat{P})} + \frac{n-k+1}{H(\hat{P})} + \frac{1}{(1-p_{\max})H(\hat{P})} . \end{aligned} \tag{4.117}$$

We obtain (4.88) from (4.20).

Let $X_{(l),1}^n$ and $\hat{X}_{(l),1}^n$ denote the random variables representing the l-th inner codeword and its estimated value, respectively. Then the average decoding error probability can be expressed as

$$\begin{aligned} P_{\mathrm{e,out}} &= \sum_{t=1}^{\infty} \Pr \left[\{L(U_1^m) = t\} \cap \bigcup_{l=1}^{t} \{\hat{X}_{(l),1}^n \neq X_{(l),1}^n\} \right] \\ &\leq \sum_{t=1}^{\infty} \sum_{\{x_{(l),1}^n\}_{l=1}^t \in \mathcal{C}^t} \sum_{l=1}^{t} \Pr[L(U_1^m) = t,\, X_{(l),1}^n = x_{(l),1}^n,\, \hat{X}_{(l),1}^n \neq x_{(l),1}^n] \\ &= \sum_{l=1}^{\infty} \sum_{x_1^n \in \mathcal{C}} \sum_{t=l}^{\infty} \Pr[L(U_1^m) = t,\, X_{(l),1}^n = x_1^n,\, \hat{X}_{(l),1}^n \neq x_1^n] \\ &= \sum_{l=1}^{\infty} \sum_{x_1^n \in \mathcal{C}} \Pr[L(U_1^m) \geq l,\, X_{(l),1}^n = x_1^n,\, \hat{X}_{(l),1}^n \neq x_1^n] \\ &= \sum_{l=1}^{\infty} \sum_{x_1^n \in \mathcal{C}} \Pr[L(U_1^m) \geq l] \hat{P}(x_1^n) P_{\mathrm{e,in}}(x_1^n) \qquad (4.118) \\ &= \mathrm{E}_{U_1^m}[L(U_1^m)] P_{\mathrm{e,in}} , \qquad (4.119) \end{aligned}$$

where (4.118) follows since the IAHC code is $\hat{P}$-perfect. We obtain (4.89) by combining (4.119) with (4.117). □

Proof of Theorem 4.4. From Lemma 4.17, there exists a sequence of an $n_t \times k_t$ LDPC matrix H and a vector $v_1^k \in \mathcal{X}^k$ such that

$$\begin{aligned} \lim_{t\to\infty} P_{\mathrm{e,in}} &= 0 \\ \lim_{t\to\infty} \frac{H(\hat{P})}{n_t} &\geq I(X;Y) \end{aligned} \tag{4.120}$$

and $p_{\max} \leq 1/2$. Then, for

$$m_t = \left\lceil \frac{n_t}{\sqrt{P_{\mathrm{e,in}}}} \right\rceil , \tag{4.121}$$

we obtain from Lemma 4.18 that

$$\begin{aligned} \lim_{t\to\infty} R &\geq I(X;Y) , \\ \lim_{t\to\infty} P_{\mathrm{e,out}} &\leq \lim_{t\to\infty} \frac{n_t}{H(\hat{P})} \frac{m_t}{n_t} \left(1 + \frac{n_t - k_t + 1 + \frac{1}{1-p_{\max}}}{m_t} \right) P_{\mathrm{e,in}} \\ &\leq \frac{1}{I(X;Y)} \lim_{t\to\infty} \left(\frac{n_t}{\sqrt{P_{\mathrm{e,in}}}} + 1 \right) \frac{P_{\mathrm{e,in}}}{n_t} \\ &= 0 \end{aligned} \tag{4.122}$$

and the proof is completed. □

Chapter 5

Practical Algorithms for LDPC Codes

We proposed asymptotically optimal coding schemes using LDPC codes for lossy source coding and channel coding in Chapter 4. In the proof of the asymptotic optimality, we assumed the exact computation of marginal probabilities in (4.13) and (4.19) expressed by

$$\hat{P}_i(x_i|\hat{x}_1^{i-1}) = P_{X_1^n}[X_i = x_i|X_1^{i-1} = \hat{x}_1^{i-1}, X_1^n H = v_1^k], \quad i = 1, 2, \cdots, n, \tag{5.1}$$

and the optimization in (4.11) and (4.22) expressed by

$$\text{maximize } \; P_{X_1^n|Y_1^n}(x_1^n|y_1^n), \; \text{ subject to } \; x_1^n H = v_1^k \,. \tag{5.2}$$

We consider efficient suboptimal algorithms for these computations in this chapter.

To the author's knowledge, there has not been known a coding scheme which explicitly requires (an approximation of) the probability with form (5.1). However, this probability is required implicitly for bit-wise MAP estimation of LDPC codes and is known to be approximated accurately by belief propagation (BP). We propose an algorithm to compute the probability by improving a naive application of the BP in Section 5.2.

In Sections 5.3 and 5.4, we consider suboptimal algorithms for optimization in (5.2). This optimization is also required in symmetric cases and much research has been conducted. Although the optimization with form (5.2) is common to lossy source coding in (4.11) and channel coding in (4.22), it is known that the nature of the optimization is quite different between these two coding problems. In channel coding, any codeword other than the sent

Algorithm 5.1 Construction of a (c, d) regular LDPC Matrix H

input: $c, d \in \mathbb{N}$ s.t. $cn = kd$.

1. Initialize every elements of H to 0.
2. Prepare a multiset of integers in which the number of integer i is d.
3. Repeat the following c times for each row of H:

   3.1 Draw a number t randomly from the multiset prepared in Step 2 and remove the number from the multiset.

   3.2 Choose $a \in \mathrm{GF}(q) \setminus \{0\}$ uniformly at random and add a to the t-th bit of the row.

codeword (which is usually equal to the ML codeword) leads to decoding error and the optimization has to be computed exactly. However, the *noise* of the channel is usually not very large and the optimal solution comes close to the channel output y_1^n. On the other hand in lossy source coding, every codeword with a small distortion is acceptable and we do not have to compute the optimal solution exactly. However, the source sequence y_1^n usually does not come close to the optimal solution and the optimization itself is more difficult than in channel coding. In this point of view, we consider problem (5.2) separately in channel coding and lossy source coding. First we consider vector-quantization using reinforced belief propagation (RBP) for lossy source coding in Section 5.3 and next consider codeword estimation using linear programming (LP) for channel coding in Section 5.4.

5.1 Practical LDPC Codes

In the theoretical analysis of LDPC codes in Chapter 4, we used LDPC codes constructed by Algorithm 4.1 with row weight $\mathrm{O}(\log n)$ to assure the hash property. However, this code does not perform well for practical algorithm such as the BP and the LP. Then, in this chapter, we use different LDPC codes such that each row weight is fixed, and therefore, the number of nonempty entries of LDPC matrices is $\mathrm{O}(n)$. In most simulations, we set the row weight to 2. Whereas LDPC codes with row weight 2 behave poorly in the case of GF(2), it has been reported that LDPC matrices over $\mathrm{GF}(2^m)$ performs excellently when each row weight is 2 [64][34].

The most basic family of practical LDPC matrices is a *regular* LDPC matrix constructed by Algorithm 5.1. In (c, d) regular LDPC matrices, each row contains at most c nonzero entries and each column contains at most

d nonzero entries. Also, we use LDPC matrices constructed by progressive edge-growth (PEG) algorithm [69] to improve the performance of the codes. This algorithm constructs an irregular LDPC matrix so that the minimum girth of loops in the Tanner graph (see e.g. [13]) is small as much as possible for given row weight c. Although the hash property is not proved for these two constructions of LDPC codes, these codes perform successfully with the BP or the LP decoding.

It was reported in [34] that, in the case of vector-quantization for lossy source coding, performance is improved significantly when some rows are removed from LDPC matrices. Then we use LDPC matrices such that one row is removed in simulations on lossy source coding.

5.2 BP Coding for Variable Length Codes

In this section, we consider computation of (5.1) used in arithmetic coding for lossy source coding or homophonic coding for channel coding. We use belief propagation given in Algorithm 5.2 for this computation. In Algorithm 5.2, each λ_i denotes a prior distribution on $\mathrm{GF}(q)$. For $H = (h_{ij})$, $\mathcal{M}(i) \equiv \{j : h_{ij} \neq 0\}$ and $\mathcal{N}(j) \equiv \{i : h_{ij} \neq 0\}$ denote the sets of nodes connected to variable node i and check node j, respectively. Variables μ_{ji} and ν_{ij} are defined for (i, j) such that $i \in \mathcal{N}(j)$, or equivalently, $j \in \mathcal{M}(i)$. Then, the complexity of this algorithm is $\mathrm{O}(w(H)) = \mathrm{O}(n)$ for a fixed repetition number r, where $w(H)$ denotes the weight of H, that is, the number of nonzero entries of H. More precisely, the complexity including the dependency on the field size q can be expressed as $\mathrm{O}(nq \log q)$ when we use a technique similar to Fast-Fourier-Transformation [65].

Recall that $x_{\mathcal{A}}$ is the subvector $\{x_i\}_{i\in\mathcal{A}}$ of x_1^n. We also define $H_{\mathcal{A}}$ as the submatrix of H which consists of i-th rows of H for $i \in \mathcal{A}$. Similarly, $H_{\mathcal{A},\mathcal{B}}$ is defined as the submatrix H which consists of i-th rows and j-th columns of H for $i \in \mathcal{A}$ and $j \in \mathcal{B}$. We define "$i : j$" as the set $\{i, i+1, \cdots, j\}$. If $i < j$ then it denotes the empty set. Then, x_i^j can also be written as $x_{i:j}$.

Eq. (5.1) can be expressed as

$$\hat{P}_i(x_i|\hat{x}_1^{i-1}) = P_{X_i^n}[X_i = x_i | X_i^n H_{i:n} = v_1^k - \hat{x}_1^{i-1} H_{1:i-1}]\,. \tag{5.3}$$

Note that $\hat{P}_i$ does not need to be computed for $i = n - \operatorname{rank} H + 1, n - \operatorname{rank} H + 2, \cdots, n$ since $\hat{x}_{n-\operatorname{rank} H+1}^n$ such that $\hat{x}_1^n H = v_1^k$ determines uniquely from $\hat{x}_1^{n-\operatorname{rank} H}$ under constraint $\hat{x}_1^n H = v_1^k$. In the following, we always assume that H has full rank, and our problem is to compute $\{\hat{P}_i\}_{i=1}^{n-k}$ for

Algorithm 5.2 Belief propagation

input: $H, v_1^k, \{\lambda_i\}_i$ and $r > 0$.

1. Initialization: $\nu_{ij}^1 := \lambda_i$, $\nu_i^0 := 1$ and $l := 1$.
2. Row operation: for each $x \in \mathrm{GF}(q)$,

$$\mu_{ji}^{(l)}(x) \propto \sum_{\{x'_{i'}\} \in \mathrm{conf}_{ij}(x)} \prod_{i' \in \mathcal{N}(j) \setminus \{i\}} \nu_{i'j}^{(l)}(x'_{i'})$$

   where

$$\mathrm{conf}_{ij}(x) \equiv \left\{ \{x'_{i'}\}_{i' \in \mathcal{N}(j)} : x'_i = x, \sum_{i' \in \mathcal{N}(j)} H_{ji'} x'_{i'} = v_j \right\}.$$

3. Column operation: for each $x \in \mathrm{GF}(q)$,

$$\nu_{ij}^{(l+1)}(x) \propto \lambda_i(x) \left(\nu_i^{(l-1)}(x) \right)^{\gamma} \prod_{j' \in \mathcal{M}(i) \setminus \{j\}} \mu_{j'i}^{(l)}(x)$$

   and

$$\nu_i^{(l+1)}(x) \propto \lambda_i(x) \prod_{j' \in \mathcal{M}(i)} \mu_{j'i}^{(l)}(x).$$

4. Repetition: repeat $l := l + 1$ and Steps 2 and 3 while $l \leq r$.
5. Return $\nu_i^{(r+1)}$.

$i = 1, 2, \cdots, n-k$.

It is well known that the marginal probability with form

$$P_{X_1^n}[X_i = x_i | X_1^n H = v_1^k] \tag{5.4}$$

can be approximated accurately by BP with input $\lambda_i = P_{X_i} = P_X$. Then, we can approximate $\{\hat{P}_i\}_{i=1}^n$ by executing BP with inputs $H_{i:n}$ and $v_1^k - \hat{x}_1^{i-1} H_{1:i-1}$ for H and v_1^k in Algorithm 5.2, respectively, for each $i = 1, 2, \cdots, n-k$. We call this procedure to obtain $\{\hat{P}_i\}_i$ *naive BP approximation*. This algorithm requires $n-k$ executions of BP and the total complexity amounts to $\mathrm{O}(n^2)$.

Let us consider a submatrix $H_{\mathcal{A},\mathcal{B}}$ for $\mathcal{A} = \{a_1, a_2, \cdots, a_{|\mathcal{A}|}\}$. We define $\tilde{P}_{a_i}(\,\cdot\,; H_{\mathcal{A},\mathcal{B}}, v_1^{|\mathcal{B}|})$ as the output $\nu_i(\cdot)$ of the BP in Algorithm 5.2 with input $H_{\mathcal{A},\mathcal{B}}$, $v_1^{|\mathcal{B}|}$ and prior probability $\lambda_i = P_X$. For example, if $\mathcal{A} = \{3, 5\}$ then we have $\tilde{P}_3(\,\cdot\,; H_{\mathcal{A},\mathcal{B}}, v_1^{|\mathcal{B}|}) = \nu_1(\cdot)$ and $\tilde{P}_5(\,\cdot\,; H_{\mathcal{A},\mathcal{B}}, v_1^{|\mathcal{B}|}) = \nu_2(\cdot)$, but $\tilde{P}_i(\,\cdot\,; H_{\mathcal{A},\mathcal{B}}, v_1^{|\mathcal{B}|})$ for $i \notin \{3, 5\}$ are undefined. The naive BP approximation

can be regarded as the algorithm such that $\hat{P}_i$ is approximated by

$$\tilde{P}_i(x_i; H_{i:n}, v_1^k - H_{1:i-1}\hat{x}_1^{i-1}) \,. \tag{5.5}$$

In Sections 5.2.1 and 5.2.2, we improve the complexity of naive BP approximation from the following points of view. 1) Some check and variable nodes may be eliminated without affecting the set of possible codewords. 2) A previous result of the BP may be a good approximation of the BP for the current i.

5.2.1 Simplification of Tanner Graph

First, for example, consider a linear code $\mathcal{C} = \{x_1^n : x_1^n H = v_1^k\}$ on GF(3) for

$$H = \left(\begin{array}{ccc|ccccc} 0 & 1 & 2 & 1 & 2 & 0 & 1 & 2 \\ 1 & 2 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 2 & 1 \\ 2 & 2 & 0 & 0 & 0 & 2 & 2 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{array}\right)^t \tag{5.6}$$

and consider a codeword $\hat{x}_1^n = (\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_8) \in \mathcal{C}$, where H^t denotes the transpose of matrix H. We have to compute marginal probabilities on $(\hat{x}_1, \cdots, \hat{x}_3)$. When we use (5.5) for the approximation of $\hat{P}_i$,

$$H_{2:8} = \left(\begin{array}{cc|ccccc} 1 & 2 & 1 & 2 & 0 & 1 & 2 \\ 2 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 2 & 1 \\ 2 & 0 & 0 & 0 & 2 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{array}\right)^t \tag{5.7}$$

and

$$H_{3:8} = \left(\begin{array}{c|ccccc} 2 & 1 & 2 & 0 & 1 & 2 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & 1 \\ 0 & 0 & 0 & 2 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 2 \end{array}\right)^t \tag{5.8}$$

are required as arguments in the BP to derive the marginal probability of $\hat{x}_2$ and $\hat{x}_3$, respectively.

On the other hand, $\hat{x}_8$ can be determined from $\hat{x}_1$ by $\hat{x}_8 = (v_5 - \hat{x}_1)/2$, which is obtained from the fifth row of (5.6). Similarly, $\hat{x}_6$ and $\hat{x}_7$ can be

determined from $\hat{x}_1$ and $\hat{x}_2$ as

$$\begin{pmatrix} \hat{x}_6 \\ \hat{x}_7 \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 2 & 2 \end{pmatrix}^{-1} \left(\begin{pmatrix} v_3 \\ v_4 \end{pmatrix} - \begin{pmatrix} 0 & 1 & 1 \\ 2 & 2 & 1 \end{pmatrix} \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_8 \end{pmatrix} \right). \quad (5.9)$$

Therefore, it is sufficient to consider

$$H^{(2)} \equiv H_{2:7,1:4} = \left(\begin{array}{cc|cccc} 1 & 2 & 1 & 2 & 0 & 1 \\ 2 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 2 \\ 2 & 0 & 0 & 0 & 2 & 2 \end{array} \right)^t \quad (5.10)$$

for $\hat{x}_2$ and

$$H^{(3)} \equiv H_{3:5,1:2} = \left(\begin{array}{c|cc} 2 & 1 & 2 \\ 1 & 1 & 1 \end{array} \right)^t \quad (5.11)$$

for $\hat{x}_3$. We can reduce the complexity of BP by using these matrices $H^{(2)}$ and $H^{(3)}$ instead of $H_{2:n}$ and $H_{3:n}$, respectively. Furthermore, the accuracy is also improved for the marginal probability of $\hat{x}_3$ in the BP since a loop in the Tanner graph is removed.

Now we formulate the procedure of the above example. The i'-th element of the solution of

$$\hat{x}_i^n H_{i:n} = v_1^k - H_{1:i-1} \hat{x}_1^{i-1} \quad (5.12)$$

cannot be determined uniquely if and only if $\mathrm{rank} H_{i:n} = \mathrm{rank} H_{(i:n)\setminus\{i'\}}$ holds. Then the set of elements depending on $\hat{x}_i, \cdots, \hat{x}_m$ is given by

$$\mathcal{I}(i) \equiv \left\{ i' \in \{m+1, \cdots, n\} : \mathrm{rank} H_{i:n} = \mathrm{rank} H_{(i:n)\setminus\{i'\}} \right\}. \quad (5.13)$$

The columns in $\tilde{\mathcal{I}}(i) \equiv \{m+1, \cdots, n\} \setminus \mathcal{I}(i)$ can be ignored in the BP for encoding of $\hat{x}_{i+1}, \cdots, \hat{x}_m$ since $\hat{x}_{\tilde{\mathcal{I}}(i)}$ can be determined from $\hat{x}_1^{i-1}$. Note that, as in the case of the previous example, it is easy to see that we can eliminate $|\tilde{\mathcal{I}}(i)|$ constraints (i.e. columns) as well as $|\tilde{\mathcal{I}}(i)|$ variables (i.e. rows) of the constraint $x_1^n H = v_1^k$. Let $\mathcal{J}(i) \subset \mathcal{J}$ be a set of undeletable constraints, that is, a subset of $\mathcal{J}$ such that $k - |\mathcal{J}(i)| = |\tilde{\mathcal{I}}(i)|$ and $\mathrm{rank} H_{\mathcal{I}(i),\mathcal{J}(i)} = \mathrm{rank} H_{\mathcal{I}(i)}$.

Then using the above $\mathcal{I}(i)$ and $\mathcal{J}(i)$, we obtain

$$\begin{aligned} \hat{P}_i(x|\hat{x}_1^{i-1}) &= P_{X_i^n}[X_i = x_i | X_i^n H_{i:n} = v_1^k - \hat{x}_1^{i-1} H_{1:i-1}] \\ &= P_{X_{\mathcal{I}(i)}}[X_i = x_i | X_{\mathcal{I}(i)} H_{\mathcal{I}(i),\mathcal{J}(i)} = v_{\mathcal{J}(i)} - \hat{x}_{\mathcal{I}^c(i)} H_{\mathcal{I}^c(i),\mathcal{J}(i)}] \end{aligned} \quad (5.14)$$

Algorithm 5.3 Batch BP approximation

input: $H, v_1^k, \hat{x}_1^n$ and $\{s_t\}_{t=1,\cdots,T}$.

1. Set $t := 1,\ i_t := 1$.
2. Repeat the following while $t \leq T$.

   2.1 Execute BP with input $H_{\mathcal{I}(i_t),\mathcal{J}(i_t)}$, $v_{\mathcal{J}(i_t)} - \hat{x}_{\mathcal{I}^c(i_t)} H_{\mathcal{I}^c(i_t),\mathcal{J}(i_t)}$ and $\lambda_i = P_X$.

   2.2 Approximate $\hat{P}_i$ by $\tilde{P}_i^{(i_t)}(x_i|\hat{x}_1^{i-1})$ for $i = i_t,\ i_t+1, \cdots, i_{t+1}-1$.

   2.3 Update $i_{t+1} := i_t + s_t$ and $t := t+1$.

and $\hat{P}_i(x_i|\hat{x}_1^{i-1})$ can be approximated by

$$\tilde{P}_i(x_i; H_{\mathcal{I}(i),\mathcal{J}(i)}, v_{\mathcal{J}(i)} - \hat{x}_{\mathcal{I}^c(i)} H_{\mathcal{I}^c(i),\mathcal{J}(i)}) \tag{5.15}$$

instead of (5.5).

5.2.2 Exploitation of Past BP Outputs

Next we consider the case that for some $s > 0$, $\hat{P}_i, \hat{P}_{i+1}, \cdots, \hat{P}_{i+s-1}$ can be approximated from an output of one BP execution. If variable nodes $i, \cdots, i+s-1$ are sufficiently distant from each other on the Tanner graph, the value of each $X_{i'}$, $i' \in \{i, \cdots, i+s-1\}$, does not affect the probabilities on the other variables so much. For this $i' \in \{i, \cdots, i+s-1\}$,

$$\hat{P}_{i'}(x_{i'}|\hat{x}_1^{i'-1}) = P_{X_1^n}[X_{i'} = x_{i'}|X_1^{i'-1} = \hat{x}_1^{i'-1}, X_1^n H = v_1^k] \tag{5.16}$$

$$\approx P_{X_1^n}[X_{i'} = x_{i'}|X_1^{i-1} = \hat{x}_1^{i-1}, X_1^n H = v_1^k] \tag{5.17}$$

and (5.17) can be approximated by the output of BP

$$\tilde{P}_{i'}^{(i)}(x_{i'}|\hat{x}_1^{i-1}) \equiv \tilde{P}_{i'}(x_{i'}; H_{\mathcal{I}(i),\mathcal{J}(i)}, v_{\mathcal{J}(i)} - \hat{x}_{\mathcal{I}^c(i)} H_{\mathcal{I}^c(i),\mathcal{J}(i)}). \tag{5.18}$$

These probabilities for $i' \in \{i, \cdots, i+s-1\}$ are obtained at once by the BP with input $(H_{\mathcal{I}(i),\mathcal{J}(i)}, v_{\mathcal{J}(i)} - \hat{x}_{\mathcal{I}^c(i)} H_{\mathcal{I}^c(i),\mathcal{J}(i)})$. This idea can be described as Algorithm 5.3, which encodes x_1^n with $\{s_t\}_{t=1,\cdots,T}$, where $\{s_t\}$ satisfies $\sum_{t=1}^T s_t = m$ and is determined in advance by Algorithm 5.4 described later.

In Algorithm 5.3, $i_t, \cdots, (i_t+s_t-1)$-st elements of $\hat{x}_1^n$ are encoded by one BP. Therefore, the computational complexity decreases when each s_t is large or T is small.

Now we consider the loss in the code length arising from this approximation. If we do not use the past result of BP, the assigned probability for the

Algorithm 5.4 Preprocessing of Batch BP Compression

input: H, v_1^k, integer $N > 0$ and real $\rho > 0$.

1. Initialize $t := 1, i_t := 1$.
2. Generate $\hat{x}^n_{(1),1}, \cdots, \hat{x}^n_{(N),1}$ from distribution $P_{\hat{X}_1^n}$.
3. while $i_t < m$ do

   3.1. Set $s_t := 1$.

   3.2. Compute $\tilde{P}^{(i_t)}(\hat{x}^n_{(l),1})$ for each l.

   3.3. Compute $\tilde{P}^{(i_t+s_t)}(\hat{x}^n_{(l),1})$ for each l.

   3.4. If there exists $i \in \{i_t + s_t, \cdots, n\} \setminus \tilde{\mathcal{I}}(i_t + s_t)$ such that

$$\frac{1}{N}\sum_{l=1}^{N}\left(\log \tilde{P}_i^{(i_t+s_t)}(\hat{x}_i|\hat{x}_1^{i_t+s_t-1}) - \log \tilde{P}_i^{(i_t)}(\hat{x}_i|\hat{x}_1^{i_t-1})\right) \le \rho \tag{5.20}$$

   then

   3.4.1. Take such i arbitrarily and exchange the i-th element with the $i_t + s_t$-th element for $\hat{x}^n_{(l),1}$, $\tilde{P}^{(i)}(\hat{x}^n_{(l),1})$ and each row of H.

   3.4.2. Update $s_t := s_t + 1$.

   3.4.3. Go to step 3.3.

   3.5. Update $i_{t+1} := i_t + s_t$ and $t := t + 1$.

4. Return H and $(s_1, s_2, \cdots, s_t)$.

arithmetic coding of $\hat{x}_{i'}$ is given by $\tilde{P}_{i'}^{(i')}(\hat{x}_{i'}|\hat{x}_1^{i'-1})$. Hence, the expectation of the redundant average code length for the i'-th bit of $\hat{x}_1^n$ is expressed as

$$I_{i,i'} \equiv \mathrm{E}_{\hat{X}_1^{i'}}\left[\log \tilde{P}_{i'}^{(i')}(\hat{X}_{i'}|\hat{X}_1^{i'-1}) - \log \tilde{P}_{i'}^{(i)}(\hat{X}_{i'}|\hat{X}_1^{i-1})\right], \tag{5.19}$$

where $\hat{X}_1^n$ denotes the random variables corresponds to $\hat{x}_1^n$ which is generated by vector-quantization from source sequence Y_1^n in lossy source coding or is generated randomly by IAHC algorithm in source coding. Then, we can estimate (5.19) by Monte Carlo simulations generating $\hat{x}_1^n$ many times. We can approximate $\{\hat{P}_i\}$ efficiently by designing $\{s_t\}$ so that each s_t of Algorithm 5.3 becomes large and $I_{i_t,i_t+1}, \cdots, I_{i_t,i_t+s_t-1}$ become sufficiently small. For this goal, we arrange rows of H and determine $\{s_t\}$ by Algorithm 5.4, where $\tilde{P}^{(i)}(\hat{x}_1^n) \equiv \{\tilde{P}_{i'}^{(i)}(x_{i'}|\hat{x}_1^{i-1})\}_{i'}$. This algorithm greedily searches variables with a parameter ρ which is used at (5.20) in the algorithm to check whether the variables can be encoded efficiently by the past result of BP. Letting N suffi-

ciently large, the left-hand side of (5.20) approaches (5.19) and we can assure that the average redundant code length is at most ρm compared to the algorithm where the probability for the arithmetic coding is computed by the BP for each element of $\hat{x}_1^n$.

Remark 5.1. If H and $\hat{x}_1^n$ are taken ideally and n is sufficiently large then $\hat{x}_1^n$ follows the distribution $P_{X_1^n}[X_1^n = \hat{x}_1^n | X_1^n H = v_1^k]$. Since $\tilde{P}_{i'}^{(i')}(x_{i'}|\hat{x}_1^{i'-1})$ and $\tilde{P}_{i'}^{(i)}(x_{i'}|\hat{x}_1^{i-1})$ are approximations of (5.16) and (5.17), respectively, $I_{i,i'}$ corresponds to the mutual information $I(X_{i'}; X_i^{i'-1}|X_1^i,\, X_1^n H = v_1^k) \geq 0$. Thus $I_{i,i'}$ is usually positive unless the approximation of the marginal probability by the BP is very inaccurate.

Remark 5.2. In Algorithm 5.3, $m - i_t$ elements remain unencoded at the end of the t-th iteration. If the same fraction, say ξ, of the remaining elements are encoded in every iteration, the number of loops is $\log_\xi m = \mathrm{O}(\log n)$. Since the complexity of one BP execution is $\mathrm{O}(n)$, the total complexity of the batch BP compression is $\mathrm{O}(n \log n)$ on the above assumption. We confirm by simulation that a curve $a \cdot n \log m$ fits well to the actual complexity.

5.2.3 Simulation

In this section, we evaluate performance of the algorithms proposed in Section 5.2. In all results, each shown value is the average over 200 trials. These algorithms represent a sequence $\hat{x}_1^n$, which is supposed to follow probability $\hat{P}(\hat{x}_1^n) = \prod_{i=1}^n \hat{P}_i(\hat{x}_i|\hat{x}_1^{i-1})$, by arithmetic coding or IAHC coding with assigned probability

$$\prod_{i=1}^{m} \tilde{P}_i(\hat{x}_i; H_{i:n}, v_1^k - H_{1:i-1}\hat{x}_1^{i-1}) \tag{5.21}$$

in naive BP approximation and

$$\prod_{t=1}^{T} \prod_{i=i_t}^{i_{t+1}-1} \tilde{P}_i^{(i_t)}(\hat{x}_{i'}|\hat{x}_1^{i-1}) \tag{5.22}$$

in batch BP approximation.

Then, we measure the performance of these approximation algorithms by taking $\hat{x}_1^n$ many times and comparing the empirical means of

$$-\frac{1}{n} \log \tilde{P}_i(\hat{x}_i; H_{i:n}, v_1^k - H_{1:i-1}\hat{x}_1^{i-1}) \tag{5.23}$$

and

$$-\frac{1}{n}\log\prod_{t=1}^{T}\prod_{i=i_t}^{i_{t+1}-1}\tilde{P}_i^{(i_t)}(\hat{x}_{i'}|\hat{x}_1^{i-1}) \tag{5.24}$$

with

$$-\frac{1}{n}\log P_{X_1^n}[X_1^n=\hat{x}_1^n|X_1^nH=v_1^k]=-\log P_{X_1^n}(\hat{x}_1^n)+\log P_{X_1^n}[X_1^nH=v_1^k]\,. \tag{5.25}$$

Note that it is difficult to compute the exact value of (5.25). However, for the last term of (5.25), it is natural to estimate $P_{X_1^n}[X_1^nH=v_1^k]\approx 1/|\mathrm{Im}\,\mathcal{H}|$ since H is used as a "hash" function. In fact, it is easy to see from Jensen's inequality that $\mathrm{E}_{V_1^k}\left[\log P_{X_1^n}[X_1^nH=V_1^k]\right]\le-\log|\mathrm{Im}\,\mathcal{H}|$ holds for V_1^k uniformly distributed on $\mathrm{Im}\,\mathcal{H}$. Then we use

$$-\frac{1}{n}\left(\log P_{X_1^n}(\hat{x}_1^n)-\log|\mathrm{Im}\,\mathcal{H}|\right) \tag{5.26}$$

instead of (5.25) for the ideal rate. If $\hat{x}_1^n$ is ideally generated then $P_{X_1^n}(\hat{x}_1^n)\approx 2^{-nH(X)}$ and (5.26) approaches the ideal rate $R=I(X;Y)$ from (4.10).

The complexity of these algorithms is measured by the following criterion. For fixed field size q and repetition number r, the complexity of the BP mainly depends on the number of edges of the Tanner graph, which equals the weight $w(H)$ of the matrix H. Therefore, we measure the complexity of the lossless compression part by

$$\sum_{i=1}^{m}w(H_{i:n}) \tag{5.27}$$

for the naive BP compression and by

$$\sum_{t=1}^{T}w(H_{\mathcal{I}(i_t),\mathcal{J}(i_t)}) \tag{5.28}$$

for the batch BP compression.

Table 5.1 shows the empirical means of coding rates (5.21), (5.24) and (5.25) and complexities (5.27) and (5.28) for distribution $(P_X(0),\,P_X(1))=(0.6,\,0.4)$, where the ideal rate is $R=I(X;Y)=0.4502$. We used 64-ary 2000×1000 LDPC matrix with row weight 2 constructed by PEG algorithm introduced in Section 5.1. The sequences $\hat{x}_1^n$ is generated based on the vector-quantization algorithm given in the next section. For the batch BP compression, we arranged H and determined $\{s_t\}$ by Algorithm 5.4 with $N=50$ and

varying parameter ρ. The repetition number r_0 of BP in the preprocessing, given in Algorithm 5.4, was also used as a parameter, whereas the repetition number for naive BP approximation and batch BP approximation was fixed to $r_1 = 30$. We see from Table 5.1 that we can reduce the complexity of the lossless compression significantly by the batch BP compression with slight loss in the coding rate compared to the naive BP compression. Further, the effect of the repetition number r_0 on the coding rate is relatively small compared to the parameter ρ. Based on this result we set $r_0 = 10$ and $\rho = 0.001$ in the following simulations.

Next, Fig. 5.1 shows the relation between the total complexity of the lossless compression and the block length n for the setting that $P_X(1) = 0.4$, $k/n = 1/2$. The dotted line denotes the least-square regression for the model

$$(\text{total complexity}) = a \cdot n \log m = a \cdot n \log(n-k)\,. \tag{5.29}$$

As discussed in Remark 5.2, the complexity fits well to this model and we can assume that the complexity of the batch BP compression is roughly $\mathrm{O}(n \log n)$.

Finally, we give a simulation result of the channel coding scheme proposed in Section 4.4 with BP for the codeword estimation and batch BP approximation for IAHC coding. We used quaternary-input and quaternary-output asymmetric channel

$$W(y|x) = \begin{cases} 1-3p, & x = 0,1,\ y = x, \\ p, & x = 0,1,\ y \neq x, \\ 1-3q, & x = 2,3,\ y = x, \\ q, & x = 2,3,\ y \neq x. \end{cases} \tag{5.30}$$

Table. 5.1. Comparison between the naive BP approximation and the batch BP approximation.

coding scheme	coding rate	total complexity
ideal compression	0.4502	—
naive	0.4525	2.9×10^6
batch, $\rho = 0.01$, $r_0 = 10$	0.4546	1.7×10^4
batch, $\rho = 0.01$, $r_0 = 30$	0.4545	1.8×10^4
batch, $\rho = 0.001$, $r_0 = 10$	0.4538	2.7×10^4
batch, $\rho = 0.001$, $r_0 = 30$	0.4537	3.0×10^4

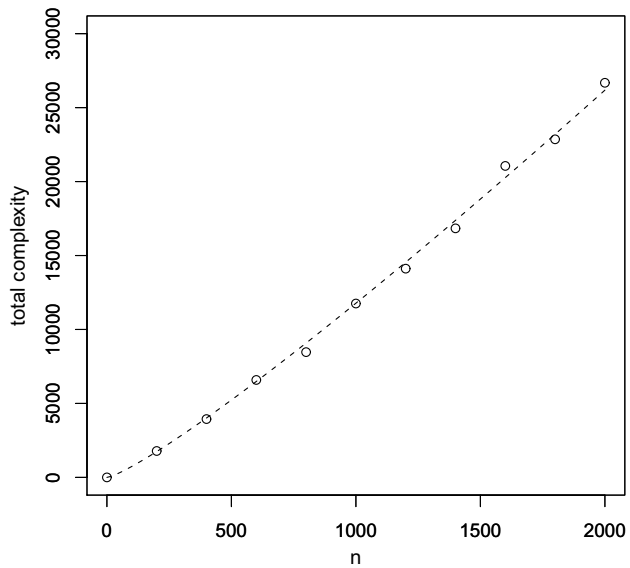

Fig. 5.1. Relationship between the block length and the complexity of the batch BP approximation.

We set the parameters to $(p, q) = (0.00319, 0.0597)$ so that the ideal input distribution is

$$(P_X(0), P_X(1), P_X(2), P_X(3)) = (3/16, 3/16, 5/16, 5/16)\,. \tag{5.31}$$

We used $1000 \times k$ LDPC codes with row weight 3 by PEG construction. Here the LDPC codes are over GF(4) in the proposed scheme and over GF(16) in the scheme by Gallager's method. Since the complexity of BP for LDPC codes over GF(q) is $\mathrm{O}(q \log q)$, the complexity of one iteration of BP is about 8 times faster in the proposed scheme than that by Gallager's method. The number of iteration of BP is set to 200 in the codeword estimation and set to 30 in the batch BP approximation. The number of BP executions in the batch BP approximation was $T \leq 6$. Note that it is also necessary to determine the message length m as a parameter in the proposed scheme. In this simulation, we designed m so that the number of inner codewords becomes $L(U_1^n) = 1$ with probability more than 99%.

Fig. 5.2 shows the block decoding error probabilities of the proposed scheme and the scheme by Gallager's method for varying the number of columns k. The capacity of this channel is $C(W) = 1.5$ and we see from the figure that our scheme can attain small decoding error probability with coding rate close to the capacity.

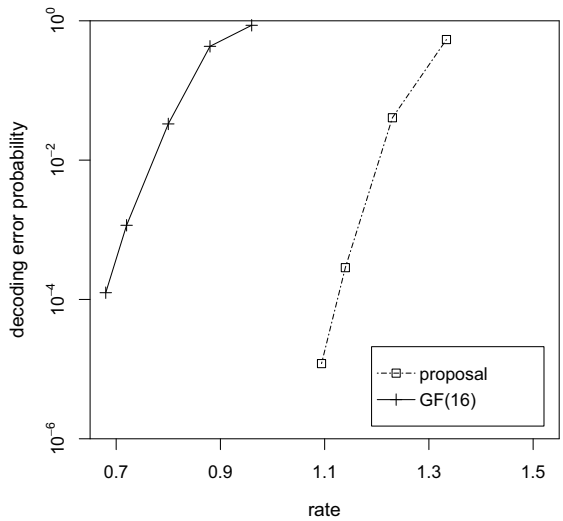

Fig. 5.2. Relation between block decoding error probability and coding rate.

5.3 Vector-quantization for Lossy Source Coding

In the proposed scheme for lossy source coding, it is necessary to compute (5.2) for the vector-quantization part as well as $\{\hat{P}_i\}$ for the lossless compression part. It has been known that the vector-quantization can be performed efficiently by reinforced belief propagation (RBP) [33][34], which is a generalization of the algorithm proposed for LDGM code [32]. The RBP algorithm is described in Algorithm 5.5. It is easy to see that BP can be regarded as RBP with parameters $\gamma = 0$ and $L = 1$.

Let $\tilde{x}_1^n = \tilde{x}_1^n(y_1^n) = (\tilde{x}_1, \tilde{x}_1, \cdots, \tilde{x}_n)$ be the symbolwise MAP sequence estimated by Algorithm 5.5 with prior distribution $\lambda_i = P_{X|Y}(\cdot|y_i)$. Then $\tilde{x}_1^n$ is given by

$$\tilde{x}_i \equiv \underset{x \in \{0, \cdots, q-1\}}{\operatorname{argmax}} \ \nu_i(x).$$

Note that $\tilde{x}_1^n$ does not always satisfy $\tilde{x}_1^n H = v_1^k$ especially in the case that the RBP did not converge. In this case a proper codeword $\hat{x}_1^n = (\hat{x}_{\mathcal{F}}, \hat{x}_{\mathcal{F}^c})$ can be obtained from $\tilde{x}_{\mathcal{F}}$ by solving

$$\begin{cases} \hat{x}_{\mathcal{F}} &= \tilde{x}_{\mathcal{F}}\,, \\ \hat{x}_{\mathcal{F}^c} H_{\mathcal{F}^c} &= v_1^k + \hat{x}_{\mathcal{F}} H_{\mathcal{F}}\,, \end{cases} \tag{5.32}$$

where a free-variable subset $\mathcal{F} \subset \mathcal{I}$ and a bounded-variable subset $\mathcal{F}^c = \mathcal{I} \backslash \mathcal{F}$ are taken so that $\mathrm{rank} H_{\mathcal{F}^c} = |\mathcal{F}^c| = k$.

Algorithm 5.5 Reinforced Belief Propagation

parameter: $L, \gamma \geq 0$ and $r > 0$.

input: H, v_1^k, λ_i.

1. Initialization: $\lambda_i := \lambda_i^L$, $\nu_{ij}^1 := \lambda_i$, $\nu_i^0 := 1$ and $l := 1$.
2. Row operation: for each $y \in \mathrm{GF}(q)$,

$$\mu_{ji}^{(l)}(x) \propto \sum_{\{y'_{i'}\} \in \mathrm{conf}_{ij}(x)} \prod_{i' \in \mathcal{N}(j) \setminus \{i\}} \nu_{i'j}^{(l)}(x'_{i'}) \,.$$

3. Column operation: for each $x \in \mathrm{GF}(q)$,

$$\nu_{ij}^{(l+1)}(x) \propto \lambda_i(x) \left(\nu_i^{(l-1)}(x)\right)^{\gamma} \prod_{j' \in \mathcal{M}(i) \setminus \{j\}} \mu_{j'i}^{(l)}(x)$$

   and

$$\nu_i^{(l+1)}(x) \propto \lambda_i(x) \prod_{j' \in \mathcal{M}(i)} \mu_{j'i}^{(l)}(x).$$

4. Repetition: repeat $l := l + 1$ and Steps 2 and 3 while $l \leq r$.
5. Return $\nu_i := \nu_i^{(r+1)}$.

In [34], the encoder executes the RBP repeatedly as changing the matrix H until the RBP converges. After the RBP converged for some H, codeword $\hat{x}_1^n$ is computed from $\tilde{x}_1^n$ for a set $\mathcal{F}$ fixed in advance. Then, the number of RBP executions sometimes becomes large and it worsens the complexity of encoding.

5.3.1 Matroid-based Rounding

In our coding scheme, we consider how to derive a good codeword $\hat{x}_1^n$ from $\tilde{x}_1^n$ obtained by only one RBP execution to decrease the complexity. For this goal, we regard the value $\bar{\nu}_i \equiv \max_{x \in \{0, \cdots, q-1\}} \nu_i(x)$ as a confidence measure, which represents how reliable "$\hat{x}_i = \tilde{x}_i$" is, and select the free-variable subset $\mathcal{F}$ in the decreasing order of $\bar{\nu}_i$ from the largest one, or equivalently, the bounded-variable subset $\mathcal{F}^c$ in the increasing order of $\bar{\nu}_i$ from the smallest one.

Now our problem is to solve

$$\text{minimize} \quad \sum_{i \in \mathcal{F}^c} \bar{\nu}_i \,, \qquad \text{subject to} \quad \mathrm{rank} H_{\mathcal{F}^c} = |\mathcal{F}^c| = k \,. \tag{5.33}$$

Algorithm 5.6 Greedy Algorithm for a Minimum Independent Set

input: H, $\{\nu_i\}$.

1. Sort $\bar{\nu}_1, \cdots, \bar{\nu}_n$ in the increasing order of $\bar{\nu}_i$ so that $\bar{\nu}_{i_1} \leq \cdots \leq \bar{\nu}_{i_n}$.
2. Set $\mathcal{F}^c := \emptyset$.
3. While $|\mathcal{F}^c| < m$
   - 3.1. If $\mathrm{rank} H_{\mathcal{F}^c \cup \{i_l\}} = |\mathcal{F}^c| + 1$ then $\mathcal{F}^c := \mathcal{F}^c \cup \{i_l\}$.
   - 3.2. $l := l + 1$.
4. Return $\mathcal{F}^c$.

The solution of (5.33) is known as a minimum weight basis of a matroid with ground set $\mathcal{I} = \{1, \cdots, n\}$ and independent sets $\{\mathcal{F}^c \subset \mathcal{I} : \mathrm{rank} H_{\mathcal{F}^c} = |\mathcal{F}^c|\}$ (see, e.g., [70] for matroid theory). It can be computed efficiently by a greedy algorithm given in Algorithm 5.6.

In Algorithm 5.6 we need to check whether $\mathrm{rank} H_{\mathcal{F}^c \cup \{i_l\}}$ is equal to $|\mathcal{F}^c|+1$ or not. In the simulations for lossy source coding, we use nonbinary LDPC codes such that each row weight of H is 1 or 2. In this case, the rank structure of H is strongly related to a subgraph $G(H, \mathcal{F}^c)$ of Tanner graph $G(H)$ of H. $G(H, \mathcal{F}^c)$ consists of variable nodes $\{v_i : i \in \mathcal{F}^c\}$ and check nodes $\{c_j : j \in \mathcal{J}\}$ and edges connecting these variable and check nodes in $G(H)$. We can check the event $\{\mathrm{rank} H_{\mathcal{F}^c} = |\mathcal{F}^c|\}$ in Algorithm 5.6 by using Lemma 5.1 with $G(H, \mathcal{F}^c)$. Note that for any feasible solution $\mathcal{F}^c$ of (5.33), we can solve (5.32) in linear time using a similar graph used in [71], where all variable nodes have degree two.

Lemma 5.1. *Let G_{c} be a connected component of $G(H, \mathcal{F}^c)$, $\mathcal{F}^c \neq \emptyset$, and let $\mathcal{F}_{\mathrm{c}}$ (resp. $\mathcal{J}_{\mathrm{c}}$) be the set of indices of variable (resp. check) nodes of G_{c}. Then the following (i) and (ii) hold.*

(i) $\mathrm{rank} H_{\mathcal{F}^c} = |\mathcal{F}^c|$ *if and only if* $\mathrm{rank} H_{\mathcal{F}_{\mathrm{c}}, \mathcal{J}_{\mathrm{c}}} = |\mathcal{F}_{\mathrm{c}}|$ *and* $\mathrm{rank} H_{\mathcal{F}^c \setminus \mathcal{F}_{\mathrm{c}}, \mathcal{J} \setminus \mathcal{J}_{\mathrm{c}}} = |\mathcal{F}^c \setminus \mathcal{F}_{\mathrm{c}}|$.

(ii) $\mathrm{rank} H_{\mathcal{F}_{\mathrm{c}}, \mathcal{J}_{\mathrm{c}}} = |\mathcal{F}_{\mathrm{c}}|$ *if and only if the following (a) or (b) holds.*

(a) G_{c} contains no cycle and at most one degree-one variable node,

(b) $\mathcal{F}_{\mathrm{c}}$ contains no degree-one variable node, G_{c} has exactly one cycle $c_{j_1}, v_{i_1}, \cdots, c_{j_r}, v_{j_r}, c_{j_{r+1}} = c_{j_1}$ and it satisfies

$$\sum_{l=1}^{r} \frac{h_{i_l, j_l}}{h_{i_l, j_{l+1}}} \neq 1. \tag{5.34}$$

This lemma means that we can check the full-rankness by decomposing

$G(H, \mathcal{F}^c)$ into connected components. The following proof is similar to the proof of linear-time encodability of cycle codes [71] and this reference may also help readers to understand the relation between the rank of a matrix and the corresponding graph.

Proof. (i) The claim holds obviously since the fact that G_{c} is a connected component means that $H_{\mathcal{F}^c}$ is expressed as

$$H_{\mathcal{F}^c} = \begin{pmatrix} H_{\mathcal{F}_{\mathrm{c}}^c, \mathcal{J}_{\mathrm{c}}} & O \\ O & H_{\mathcal{F}^c \setminus \mathcal{F}_{\mathrm{c}}, \mathcal{J} \setminus \mathcal{J}_{\mathrm{c}}} \end{pmatrix}.$$

(ii) First assume that G_{c} has no cycle and no degree-one variable node. Take an arbitrary check node c_{j_0} of G_{c}. Let $\mathcal{J}_l$ be the set of check nodes with distance $2l$ from c_{j_0} and $\mathcal{F}_l$ be the set of variable nodes with distance $2l-1$ from c_{j_0}. If the largest distance of nodes in G_{c} from c_{j_0} is $2r$, then $H_{\mathcal{F}_{\mathrm{c}}, \mathcal{J}_{\mathrm{c}}}$ can be expressed as the form

$$H_{\mathcal{F}_{\mathrm{c}}, \mathcal{J}_{\mathrm{c}}} = \begin{array}{c} \\ \mathcal{F}_1 \\ \mathcal{F}_2 \\ \vdots \\ \mathcal{F}_r \end{array} \begin{array}{c} \begin{array}{ccccc} \mathcal{J}_0 & \mathcal{J}_1 & \mathcal{J}_2 & \cdots & \mathcal{J}_r \end{array} \\ \begin{pmatrix} A_1 & D_1 & O & \cdots & O \\ O & A_2 & D_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & O \\ O & \cdots & O & A_r & D_r \end{pmatrix} \end{array}, \tag{5.35}$$

where $\mathcal{J}_0 = \{j_0\}$ and $D_1, \cdots, D_r$ are diagonal matrices. It is straightforward from (5.35) that $\mathrm{rank} H_{\mathcal{F}_{\mathrm{c}}, \mathcal{J}_{\mathrm{c}}} = |\mathcal{F}_{\mathrm{c}}|$. Similarly, assume that G_{c} contains no cycle and one degree-one variable node v_{i_1}. Letting $\mathcal{J}_l$ and $\mathcal{F}_l$ be the set of check and variable nodes with distance $2l-1$ and $2l-2$ from v_{i_1}, respectively, $H_{\mathcal{F}_{\mathrm{c}}, \mathcal{J}_{\mathrm{c}}}$ can be expressed as the form

$$\begin{array}{c} \\ \mathcal{F}_1 \\ \mathcal{F}_2 \\ \vdots \\ \mathcal{F}_r \end{array} \begin{array}{c} \begin{array}{cccc} \mathcal{J}_1 & \mathcal{J}_2 & \cdots & \mathcal{J}_r \end{array} \\ \begin{pmatrix} D_1 & O & \cdots & O \\ A_2 & D_2 & \ddots & \vdots \\ \ddots & \ddots & \ddots & O \\ \cdots & O & A_r & D_r \end{pmatrix} \end{array}. \tag{5.36}$$

Thus $\mathrm{rank} H_{\mathcal{F}_{\mathrm{c}}, \mathcal{J}_{\mathrm{c}}} = |\mathcal{F}_{\mathrm{c}}|$ since the above matrix is triangular.

Next assume that G_{c} contains one cycle and no degree-one variable node.

Then $H_{\mathcal{F}_c,\mathcal{J}_c}$ can be expressed as the form

$$H_{\mathcal{F}_c,\mathcal{J}_c} = \begin{pmatrix} H^- & O & \cdots & \cdots & O \\ A_1 & D_1 & O & \cdots & O \\ O & A_2 & D_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & O \\ O & \cdots & O & A_r & D_r \end{pmatrix}, \tag{5.37}$$

where $D_1, \cdots, D_r$ are diagonal matrices and the submatrix H^- corresponding to the cycle in G_c becomes

$$H^- = \begin{pmatrix} h_{i_1,j_1} & h_{i_1,j_2} & 0 & \cdots & 0 \\ 0 & h_{i_2,j_2} & h_{i_2,j_3} & \ddots & \vdots \\ \vdots & 0 & h_{i_3,j_3} & \ddots & 0 \\ 0 & \vdots & \ddots & \ddots & h_{i_{r-1},j_r} \\ h_{i_r,j_1} & 0 & \cdots & 0 & h_{j_r,i_r} \end{pmatrix}. \tag{5.38}$$

Since $D_1, \cdots, D_r$ are diagonal matrices, $\mathrm{rank} H_{\mathcal{F}_c,\mathcal{J}_c} = |\mathcal{F}_c|$ if and only if H^- has full rank. Furthermore, by checking the determinant of H^-, we note that H^- has full rank if and only if (5.34) holds.

Finally consider the case that the total number of cycles and degree-one variables nodes in G_c is more than two. In this case, the number of variable nodes is larger than that of check nodes, that is, $|\mathcal{F}_c| > |\mathcal{J}_c|$. Then $H_{\mathcal{F}_c,\mathcal{J}_c}$ cannot have full row-rank.

We obtain the lemma by putting these case analyses together. □

Now we consider the time complexity of Algorithm 5.6. The complexity of the sort in Step 1 is $\mathrm{O}(n \log n)$. Another factor for the complexity is to check whether $\mathrm{rank} H_{\mathcal{F}^c \cup \{i_l\}} = |\mathcal{F}^c| + 1$ holds in Step 3.1. Note that we can decompose a graph into connected components and check a cycle in each component with complexity at most $\mathrm{O}(\log |\mathcal{F}^c|) = \mathrm{O}(\log n)$ by using a disjoint-set data structure [72]. Thus, except for the following case, we can check the full-rankness in $\mathrm{O}(\log n)$ time, and the total time complexity amounts to $\mathrm{O}(n \log n)$ through $\mathrm{O}(n)$ loops. The exceptional case is that v_{i_l} forms a first cycle in a connected component when it is added to $G(H, \mathcal{F}^c)$. In this case the condition (5.34) has to be checked. Since the evaluation of (5.34) requires $\mathrm{O}(|\mathcal{F}^c|) = \mathrm{O}(n)$ time, the worst case complexity is $\mathrm{O}(n^2)$.

Next we consider the average time complexity. Let f_k be the total time to check (5.34) for the case that Algorithm 5.6 is applied to $G(H)$ with k check nodes, and G_c be a connected component of $G(H, \mathcal{F}^c)$ with no cycle. Consider the case that (5.34) is satisfied for the cycle formed by G_c and v_{i_l}.

In this case v_{i_l} is added to G_c, then G_c has a cycle by this addition. Once G_c has a cycle, we do not need to check (5.34) for the connected component G_c any more, and hence, the residual complexity becomes f_{n-r} if G_c has r check nodes. Since the left-hand side of (5.34) takes a value on $\{1, \cdots, q-1\}$, it is natural to assume that (5.34) holds with probability $(q-2)/(q-1)$. Then the expected complexity is given by

$$f_k = \mathrm{O}(r) + \frac{q-2}{q-1} f_{k-r} + \frac{1}{q-1} f_k \,. \tag{5.39}$$

By solving (5.39), we obtain that

$$f_k = \mathrm{O}\left(\frac{q-2}{q-1} k\right) = \mathrm{O}(k) = \mathrm{O}(n)\,, \tag{5.40}$$

which is smaller than $\mathrm{O}(n \log n)$. Therefore the computation for checking the rank of a graph with a cycle does not affect the complexity in average, and the whole average complexity of Algorithm 5.6 is $\mathrm{O}(n \log n)$.

5.3.2 Simulation

Now we show some simulation results for the proposed lossy source coding scheme with the vector-quantization algorithm in Section 5.3.1 and batch BP approximation for arithmetic coding in the lossless compression part. In all results, we used Hamming distortion measure for alphabet $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ and 64-ary LDPC codes with block length $n = 2000$, i.e., 12000 bits by PEG construction with row weight 2. Each symbol of GF(64) corresponds to 6 binary symbols by using a binary expression of GF(64).

Figs. 5.3 and 5.4 show the simulation results on the relation between coding rate and distortion for $(P_X(0), P_X(1)) = (0.6, 0.4)$ and $(P_X(0), P_X(1)) = (0.7, 0.3)$, respectively. The label "BP compression" ($\times$) denotes the batch BP approximation in Algorithm 5.3 and "ideal compression" ($+$) denotes the ideal coding rate given in (5.25). We see from Fig. 5.3 that the performance of the BP compression is almost indistinguishable from the ideal compression. On the other hand, there are small gaps in Fig. 5.4 in low designed rate. The reason can be conjectured as follows.

When we use the proposed scheme, the designed rate R is decomposed to $R = H(X) - k/n$. On the other hand, the rate $1 - k/n$ is assured from the constraint for a codeword that n variables of a codeword satisfy k constraints. Therefore, to fill the gap $1 - H(X)$ of the rate is the task of the lossless compression part. As $R \to 0$ and P_X biases, this gap increases and therefore the

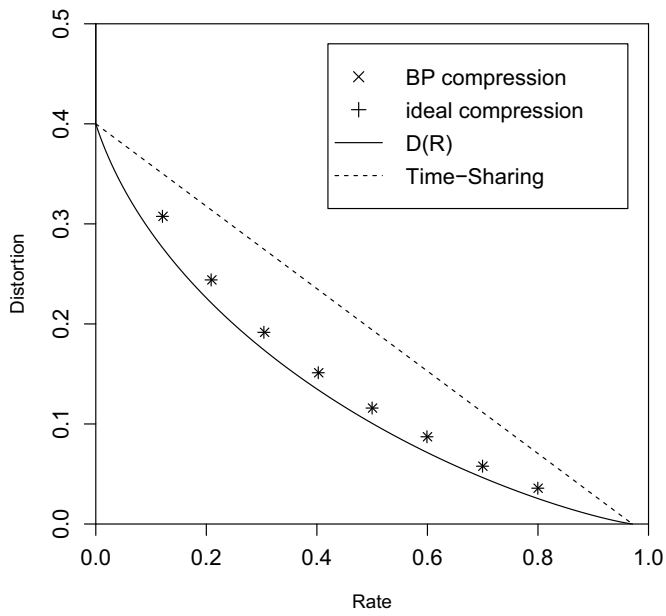

Fig. 5.3. Average distortion for $P_X(1) = 0.4$.

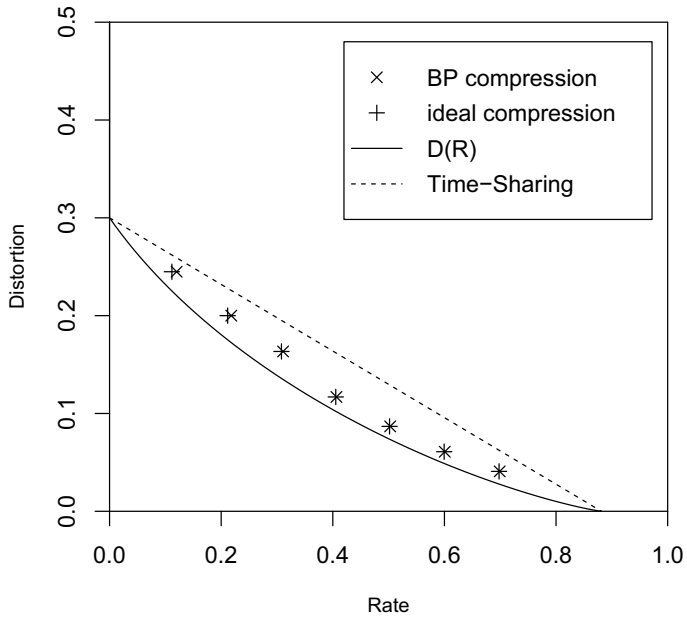

Fig. 5.4. Average distortion for $P_X(1) = 0.3$.

coding rate depends strongly on the performance of the lossless compression and the non-optimality of the BP compression appears outstandingly. Furthermore, the prior P_X is more biased in this case and it also worsens the performance of the BP compression, since the influence of a variable node over distant variable nodes in the Tanner graph becomes large.

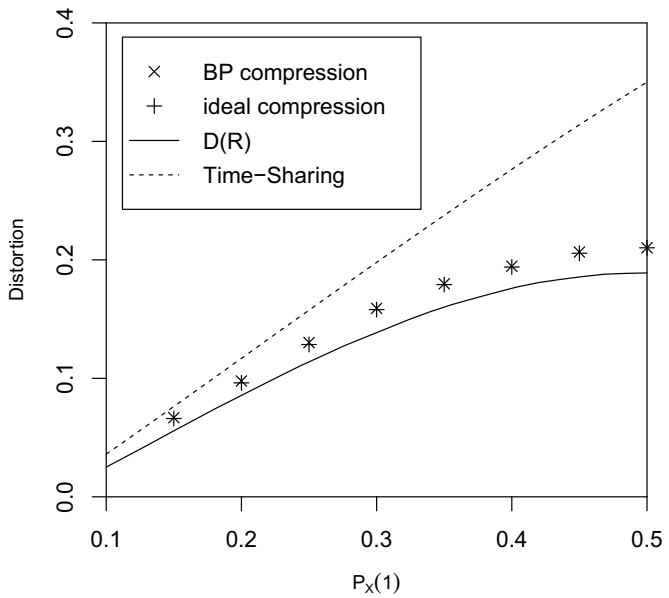

Fig. 5.5. Distortions for various input distributions with coding rate $R = 0.3$.

Finally Fig. 5.5 shows simulation results for varying input distribution P_X with fixed coding rate $R = 0.3$. Note that we cannot specify the coding rate strictly since our scheme is a variable length coding scheme. Therefore we took two simulation results with average rates slightly larger and smaller than 0.3, and plotted the convex combination of these two results. We see from the figure that our scheme can work adaptively for various source distributions.

5.4 LP Relaxation Technique for Channel Coding

In the previous sections, we have mainly considered message passing algorithms, say BP and RBP. In this section, in turn, we consider a decoding algorithm based on linear programming for channel coding. LP decoding is originally proposed for decoding of binary LDPC code by Feldman et al. [17], which relaxes the ML decoding problem to an LP problem. Then, the relaxed problem can be computed efficiently by LP solvers using e.g. the simplex method or the interior method. Whereas the decoding error probability of the LP decoding is usually worse than the BP decoding, the LP decoding has an advantage in terms of the improvability of the error probability by techniques in the field of mathematical optimization, such as a cutting-plane method [73] and a mixed integer programming [74].

Flanagan et al. [47] extended the result in [17] and proposed an LP decoding scheme for LDPC codes over finite rings $\mathbb{Z}/q\mathbb{Z}$. However, although their scheme is applicable to LDPC codes over finite fields, the number of variables in the relaxed problem grows with $\mathrm{O}(q^2)$. Then, it is difficult to use their LP decoding scheme over large finite fields.

In this section, we propose a new LP decoding scheme for LDPC codes over $\mathrm{GF}(2^m)$, in which the number of variables increases linearly in the field size. Although our scheme relaxes a maximum likelihood decoding problem more loosely to an LP problem than their scheme, we confirm by simulation that the deterioration of the decoding error probability is small. We also show that our scheme has two desirable properties, ML certificate property and all-zero codeword assumption, as in the binary case.

5.4.1 Nonbinary LDPC Codes for Binary-input Channels

In this section we give coding framework and notation treated through Section 5.4. We consider a binary-input memoryless channel such that the input and output alphabets are given by $\mathcal{X} = \{0, 1\}$ and $\mathcal{Y}$, respectively, where $\mathcal{Y}$ is not necessarily binary. The transition probability of the channel is denoted by $W(b|a)$ for $a \in \{0, 1\}$ and $b \in \mathcal{Y}$.

Let x_i and y_i be members of $\mathrm{GF}(2^m)$ and $\mathcal{Y}^m$, respectively. For notational convenience, we write sequences of these symbols by $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ and $\boldsymbol{y} = (y_1, y_2, \cdots, y_n)$ instead of x_1^n and y_1^n. Substrings of them such as $(x_i, x_{i+1}, \cdots, x_j)$ do not appear in this section and the length of the sequence can be easily seen from the context.

An $n \times k$ LDPC matrix is defined on $\mathrm{GF}(2^m)$ and an LDPC code $\mathcal{C}$ is the set of all codewords $\boldsymbol{x} = (x_1, \cdots, x_n) \in (\mathrm{GF}(2^m))^n$ such that $\boldsymbol{x}H = 0$. The encoder assigns $m \cdot \operatorname{rank} H$ bit message to a codeword $\boldsymbol{x}$ and send the binary expression of $\boldsymbol{x}$. Let $(x)_l$ denote the l-th bit of the binary expression of $x \in \mathrm{GF}(2^m)$. For example, if the binary expression of $z \in \mathrm{GF}(8)$ is 110, then we have $(z)_1 = 1$, $(z)_2 = 1$ and $(z)_3 = 0$.

The decoder receives $\boldsymbol{y} = (y_1, \cdots, y_n) \in \mathcal{Y}^{mn}$, where $y_i = ((y_i)_1, \cdots, (y_i)_m) \in \mathcal{Y}^m$, according to the probability

$$\prod_{i=1}^{n} W^m(y_i|x_i)\,, \tag{5.41}$$

where $W^m(y_i|x_i) \equiv \prod_{l=1}^{m} W((y_i)_l|(x_i)_l)$. The decoder estimates the code-

word by

$$\hat{\boldsymbol{x}} = \operatorname*{argmax}_{\boldsymbol{z}\in\mathcal{C}} \prod_{i=1}^{n} W^m(y_i|z_i) \tag{5.42}$$

and reproduces the message from $\hat{\boldsymbol{x}}$. The coding rate of this LDPC coding is given by $R = (n - \operatorname{rank} H)/n$.

5.4.2 Relaxation of Integer Variables

Now we explain the LP decoding technique for the NP complete problem required in (5.42).

In the relaxed problem we consider real variables $f_i = (f_i^0, f_i^1, \cdots, f_i^{q-1}) \in \mathcal{F}$ instead of $z_i \in \mathrm{GF}(q)$, $i = 1, \cdots, n$, where $q = 2^m$ and

$$\mathcal{F} \equiv \left\{ (f^0, \cdots, f^{q-1}) \in [0,1]^q : \quad \sum_{z=0}^{q-1} f^z = 1 \right\} . \tag{5.43}$$

Each $z \in \mathrm{GF}(q)$ is related to a point in $\mathcal{F}$ by map $G : \mathrm{GF}(q) \to \mathcal{F}$ given by[*1]

$$G(z) \equiv \begin{cases} (1, 0, \cdots, 0), & z = 0, \\ (0, 1, \cdots, 0), & z = 1, \\ \quad\vdots & \quad\vdots \\ (0, 0, \cdots, 1), & z = q-1. \end{cases} \tag{5.44}$$

We define $G^n(\boldsymbol{z}) \equiv (G(z_1), \cdots, G(z_n))$ for $\boldsymbol{z} \in (\mathrm{GF}(q))^n$.

The logarithm of the objective function in the ML decoding problem (5.42) is given by

$$\begin{aligned} L(\boldsymbol{z}; \boldsymbol{y}) &\equiv \log \prod_{i=1}^{n} W^m(y_i|z_i) \\ &= \sum_{i=1}^{n} \sum_{z=0}^{q-1} \mathbb{1}[z_i = z] \log W^m(y_i|z) , \end{aligned} \tag{5.45}$$

where $\mathbb{1}[\cdot]$ denotes the indicator function. For the relaxed problem we use a new objective function given by

$$\tilde{L}(\{f_i\}; \boldsymbol{y}) = \sum_{i=1}^{n} \sum_{a=0}^{q-1} f_i^a \log W^m(y_i|a) . \tag{5.46}$$

$\tilde{L}$ can be regarded as an extension of L to the real space in the sense that $L(\boldsymbol{z}; \boldsymbol{y}) = \tilde{L}(G^n(\boldsymbol{z}); \boldsymbol{y})$.

[*1] For simplicity of notation the elements of $\mathrm{GF}(q)$ are represented as $0, 1, \cdots, q-1$.

5.4.3 Feasible Region for GF(2)

First we introduce the construction of the feasible region of the relaxed problem for linear codes over GF(2) proposed in [17].

Recall that the sets of row and column indices of the matrix H are denoted by $i \in \mathcal{I} \equiv \{1, \cdots, n\}$ and $j \in \mathcal{J} \equiv \{1, \cdots, k\}$, respectively. $\mathcal{N}(j) \equiv \{i : h_{ij} \neq 0\}$ denotes the set of row indices of nonzero entries in the j-th column of H. The l-th element of $\mathcal{N}(j)$ is denoted by $i(j,l)$ for $l = 1, \cdots, |\mathcal{N}(j)|$. We sometimes use $+_q$, $\times_q$ to clarify the operation on the finite field. Similarly, we use $\sum_q$ for summation on the finite field.

Define $\mathcal{S}_r$ for $r \in \mathbb{N}$ by

$$\mathcal{S}_r \equiv \left\{ \boldsymbol{s} = (s_1, \cdots, s_r) \in (\mathrm{GF}(2))^r : \sum_{l=1}^{r}{}_2 \, s_l = 1 \right\} . \tag{5.47}$$

For each j and $\boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}$ we consider a constraint on $\{f_i\}$

$$\sum_{l=1}^{|\mathcal{N}(j)|} f_{i(j,l)}^{s_l} \leq |\mathcal{N}(j)| - 1 . \tag{5.48}$$

The feasible region of the relaxed problem in [17] is the set of points in $\mathcal{F}^n$ such that (5.48) is satisfied for all $j \in \mathcal{J}$ and $\boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}$. The number of constraints (5.48) is given by $|\mathcal{J}| \cdot |\mathcal{S}_{|\mathcal{N}(j)|}| = k \cdot 2^{d-1}$ for the case that each column weight is $|\mathcal{N}(j)| = d$.

Example 5.1. Consider a matrix H on GF(2) given by

$$H = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}^t . \tag{5.49}$$

Here $\mathcal{N}(1) = \{1, 2, 4\}$ and $\mathcal{N}(2) = \{1, 3, 4\}$. Since $\mathcal{S}_3 = \{(0,0,1), (0,1,0), (1,0,0), (1,1,1)\}$, the constraints corresponding to the first column of H is given by

$$\begin{aligned} f_1^0 + f_2^0 + f_4^1 &\leq 2 , \\ f_1^0 + f_2^1 + f_4^0 &\leq 2 , \\ f_1^1 + f_2^0 + f_4^0 &\leq 2 , \\ f_1^1 + f_2^1 + f_4^1 &\leq 2 . \end{aligned} \tag{5.50}$$

Table. 5.2. Operations on GF(4).

$+$	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

$\times$	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

Similarly, from the second column of H, we have

$$\begin{aligned} f_1^0 + f_3^0 + f_4^1 &\leq 2\,, \\ f_1^0 + f_3^1 + f_4^0 &\leq 2\,, \\ f_1^1 + f_3^0 + f_4^0 &\leq 2\,, \\ f_1^1 + f_3^1 + f_4^1 &\leq 2\,. \end{aligned} \tag{5.51}$$

5.4.4 Feasible Region for GF(2^m)

Next we construct the feasible region of the relaxed problem for linear codes over GF(2^m).

We start with the following example. Consider a parity check matrix H over GF(4) given by

$$H = \begin{pmatrix} 2 & 1 & 0 & 3 \\ 3 & 0 & 3 & 1 \end{pmatrix}^t . \tag{5.52}$$

The first column of H corresponds to the constraint

$$(2 \times_4 z_1) +_4 z_2 +_4 (3 \times_4 z_4) = 0\,. \tag{5.53}$$

Using the binary expression of GF(2^m) we can express the addition on GF(2^m) by the following bitwise addition on GF(2),

$$(z +_q w)_l = (z)_l +_2 (w)_l, \quad l = 1, \cdots, m. \tag{5.54}$$

Therefore the constraint (5.53) can be decomposed to two constraints

$$(2 \times_4 z_1)_1 +_2 (z_2)_1 +_2 (3 \times_4 z_4)_1 = 0\,, \tag{5.55}$$

$$(2 \times_4 z_1)_2 +_2 (z_2)_2 +_2 (3 \times_4 z_4)_2 = 0\,. \tag{5.56}$$

On the operation given in Table 5.2, the event $(2 \times_4 z_1)_1 = 1$ occurs if $z_1 \in \{1, 2\}$. Similarly, $(z_2)_1 = 1$ and $(3 \times_4 z_4)_1 = 1$ occur if $z_2 \in \{2, 3\}$ and

$z_4 \in \{1,3\}$, respectively. Therefore, we can take the following four constraints for (5.55):

$$\begin{array}{l}(f_1^0+f_1^3)+(f_2^0+f_2^1)+(f_4^1+f_4^3)\le 2\,,\\(f_1^0+f_1^3)+(f_2^2+f_2^3)+(f_4^0+f_4^2)\le 2\,,\\(f_1^1+f_1^2)+(f_2^0+f_2^1)+(f_4^0+f_4^2)\le 2\,,\\(f_1^1+f_1^2)+(f_2^2+f_2^3)+(f_4^1+f_4^3)\le 2\,,\end{array} \tag{5.57}$$

which correspond to (5.48) or (5.51). Similarly we can derive four constraints for the second bit from (5.56).

Note that there exists another constraint on GF(2) which can be derived from (5.53). Let B be a nonempty subset of $\{1,\cdots,m\}$ and define the summation on GF(2) of l-th bits of z for all $l \in B$ by

$$(z)_B \equiv \sum_{l\in B}{}_2 (z)_l\,.$$

It is obvious that $(z)_{\{l\}} = (z)_l$ and, hence, $(z)_B$ can be regarded as a generalization of $(z)_l$. For example, it holds on GF(8) that $(z)_{\{1,3\}} = 1$ and $(z)_{\{1,2,3\}} = 0$ if $z \in$ GF(8) has binary expression 110.

Since it holds similarly to (5.54) that

$$(z +_q w)_B = (z)_B +_2 (w)_B\,, \tag{5.58}$$

we obtain another constraint on GF(2) from (5.53) as follows.

$$(2 \times_4 z_1)_{\{1,2\}} +_2 (z_2)_{\{1,2\}} +_2 (3 \times_4 z_3)_{\{1,2\}} = 0\,. \tag{5.59}$$

Since $(2 \times_4 z_1)_{\{1,2\}} = 1, (z_2)_{\{1,2\}} = 1$ and $(3 \times_4 z_4)_{\{1,2\}} = 1$ hold if $z_1 \in \{1,3\}$, $z_2 \in \{1,2\}$ and $z_4 \in \{2,3\}$, respectively, the corresponding constraints on $\{f_i\}$ are

$$\begin{array}{l}(f_1^0+f_1^2)+(f_2^0+f_2^3)+(f_4^2+f_4^3)\le 2\,,\\(f_1^0+f_1^2)+(f_2^1+f_2^2)+(f_4^0+f_4^1)\le 2\,,\\(f_1^1+f_1^3)+(f_2^0+f_2^3)+(f_4^0+f_4^1)\le 2\,,\\(f_1^1+f_1^3)+(f_2^1+f_2^2)+(f_4^2+f_4^3)\le 2\,.\end{array} \tag{5.60}$$

For the above example we can derive twelve relaxed inequalities corresponding to the first column of H, eight of which are given in (5.57) and (5.60). The constraints corresponding to the second column of H can be obtained similarly.

Now we formulate the constraints in the relaxed problem. For $j \in \mathcal{J}$, $\boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}$ and $B \in \mathcal{B} \equiv 2^{\{1,\cdots,m\}} \setminus \{\emptyset\}$ define $\mathcal{P}_j^{B,\boldsymbol{s}}$ as the set of $\{f_i\} \in \mathcal{F}^n$ satisfying

$$\sum_{l=1}^{|\mathcal{N}(j)|} \sum_{z:(h_{i(j,l),j} \times_q z)_B = s_l} f_{i(j,l)}^z \leq |\mathcal{N}(j)| - 1\,. \tag{5.61}$$

We use

$$\mathcal{P} \equiv \bigcap_{j \in \mathcal{J},\, \boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|},\, B \in \mathcal{B}} \mathcal{P}_j^{B,\boldsymbol{s}} \tag{5.62}$$

as the feasible region for the relaxed problem. When each column weight is $|\mathcal{N}(j)| = d$, the number of inequality constraints is given by $|\mathcal{J}| \cdot |\mathcal{B}| \cdot |\mathcal{S}_{|\mathcal{N}(j)|}| = k(2^m - 1)2^{d-1} = k(q-1)2^{d-1}$, which is linear in the size $q = 2^m$ of the field.

Remark 5.3. A vector $\boldsymbol{z} \in (\mathrm{GF}(q))^n$ satisfying the constraint corresponding to the j-th column of H

$$\sum_{i \in \mathcal{N}(j)}{}_q\, h_{ij} \times_q z_i = 0$$

is called a local codeword. Let $\mathcal{C}_j$ be the set of local codewords for j-th column of H and $G(\mathcal{C}_j) \equiv \{G^n(\boldsymbol{z}) : \boldsymbol{z} \in \mathcal{C}_j\}$ be the image of $\mathcal{C}_j$ to the real space by map $G(\cdot)$.

Now consider a region $\mathcal{P}_j \equiv \bigcap_{B \in \mathcal{B},\, \boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}} \mathcal{P}_j^{B,\boldsymbol{s}}$. In the proposed construction of the relaxed problem,

$$\mathcal{P}_j = \mathrm{conv}(G(\mathcal{C}_j)) \tag{5.63}$$

holds only for $q = 2, 4$, where $\mathrm{conv}(A)$ denotes the convex hull of A. For $q \geq 8$, $\mathrm{conv}(G(\mathcal{C}_j)) \subsetneq \mathcal{P}_j$ holds and $\mathcal{P}_j$ has vertices other than $\mathcal{C}_j$. On the other hands, (5.63) holds for any q when $\mathcal{P}_j$ is constructed by the scheme in [47]. In this sense, our problem is relaxed more loosely than that in [47].

Remark 5.4. In the scheme of Feldman et al. for GF(2) described in Section 5.4.3 and our scheme for $\mathrm{GF}(2^m)$ given in Section 5.4.4, the number of constraints is $\mathrm{O}(2^d)$, i.e., grows exponentially, in the column weight d. But, it is shown in [75] that for the scheme of Feldman et al. we can span an equivalent polytope with $\mathrm{O}(d)$ constraints by using auxiliary variables. This technique can also be applied to our scheme and a polytope equivalent to (5.62) can be spanned with $\mathrm{O}(d)$ constraints.

5.4.5 Properties of Relaxed Problem

The relaxed problem for GF(2^m) discussed in Sections 5.4.2 and 5.4.4 is expressed as

$$\begin{aligned} &\text{maximize} \quad \tilde{L}(\{f_i\}_{i\in\mathcal{I}}; \boldsymbol{y}) \\ &\text{subject to} \quad \{f_i\} \in \mathcal{P}_j^{B,\boldsymbol{s}}, \ \ j \in \mathcal{J}, \ \boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}, \ B \in \mathcal{B}\,. \end{aligned} \tag{5.64}$$

When we use this relaxed problem in the decoding, the following properties holds as in the case of GF(2).

Theorem 5.2 (Maximum Likelihood Certificate Property)**.** *If the optimal solution* $\{f_i^*\}_{i\in\mathcal{I}}$ *of* (5.64) *is an integer vector, then* $(G^n)^{-1}(\{f_i^*\}_{i\in\mathcal{I}})$ *is the optimal solution of* (5.42)*.*

Theorem 5.3 (All-zero Codeword Assumption)**.** *Assume that the channel is symmetric in the sense that there exists a permutation* $\sigma : \mathcal{Y} \to \mathcal{Y}$ *such that* $\sigma^{-1} = \sigma$ *and*

$$W(y|0) = W(\sigma(y)|1)\,. \tag{5.65}$$

Then the decoding error probability does not depend on the codeword $\boldsymbol{x}$*, that is, it holds for all* $\boldsymbol{x} \in \mathcal{C}$ *that*

$$\Pr[\hat{\boldsymbol{x}} \neq \boldsymbol{x}|\boldsymbol{x}] = \Pr[\hat{\boldsymbol{x}} \neq \mathbf{0}|\mathbf{0}]\,. \tag{5.66}$$

The proof of Theorem 5.2 is omitted since it is straightforward. Theorem 5.3 is proved in Section 5.4.7.

5.4.6 Simulation

In this section we give some simulation results of LP decoding of nonbinary LDPC codes for binary symmetric channel (BSC).

First we compare the proposed scheme with that of Flanagan et al. [47] for regular LDPC codes given in Section 5.1. Fig. 5.6 shows decoding error probabilities for $(2,3)$ regular LDPC codes with block length 252 bits. Fig. 5.7 shows the execution times of LP for 36×24 $(2,3)$ regular LDPC codes with BSC with crossover probability $10^{-1.5}$. The dual simplex method of CPLEX with CPU Intel Core i7 is used to solve LP problems. Here note that the relaxed problems are equivalent for $q = 2, 4$.

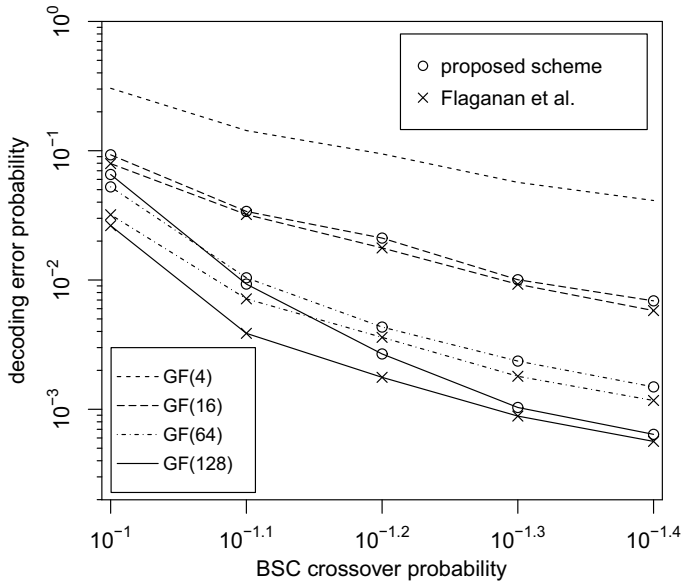

Fig. 5.6. Decoding error probabilities for $(2, 3)$ regular LDPC codes with block length 252 bits.

As shown in the figures the difference of decoding error probabilities between these two schemes is small compared with improvement by enlarging field size, whereas the execution time is improved significantly by the proposed scheme.

It is reported that regular LDPC codes with row weight 2 are the best and the error probability is improved significantly for $q \gtrsim 64$ when BP decoding is used [64]. But, it seems from Fig. 5.6 that the effect of using a large field size is limited in the case of LP decoding.

Next Fig. 5.8 shows decoding error probabilities of the proposed scheme for irregular LDPC codes with row weight 2 obtained by PEG construction so that girths of loops in the Tanner graph are large. The coding rate and the block length is the same as in Fig. 5.6. We see from the figure that irregular LDPC codes can improve the decoding error probability even for large field size when LP decoding is used.

5.4.7 Proof of Theorem 5.3

For $\{f_i\} \in \mathcal{F}^n$, $\boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}$, $B \in \mathcal{B}$, $\boldsymbol{x} \in \mathcal{C}$, define

$$\begin{aligned} \{f_i\}^{\boldsymbol{x}} &\equiv \{(f_i^{0+_q x_i}, \cdots, f_i^{(q-1)+_q x_i})\}_{i \in \mathcal{I}} \\ (\boldsymbol{s})_j^{B,\boldsymbol{x}} &\equiv \{s_l +_2 (h_{i(j,l),j} \times_q x_{i(j,l)})_B\}_{l=1,\cdots,|\mathcal{N}(j)|} \,. \end{aligned} \tag{5.67}$$

In the case of GF(2), Theorem 5.3 is proved based on the property called

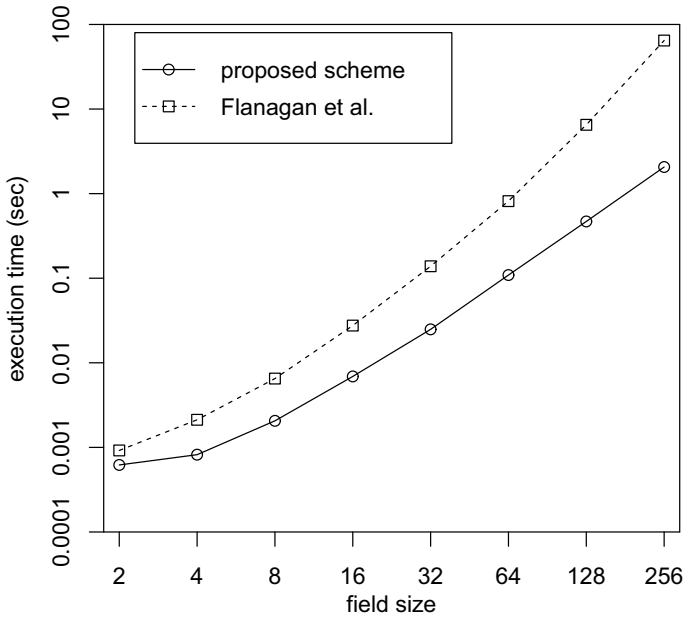

Fig. 5.7. Comparison of execution time for 24×36 (2,3) regular LDPC codes.

$\mathcal{C}$-symmetry [17]. In our case $\mathcal{C}$-symmetry can be extended as Lemma 5.4 below.

Lemma 5.4 ($\mathcal{C}$-symmetry)**.** *Let $\mathcal{P}$ be the feasible region of the relaxed problem* (5.64)*. Then it holds for every $\boldsymbol{x} \in \mathcal{C}$ that $\mathcal{P} = \{\{f_i\}^{\boldsymbol{x}} : \{f_i\} \in \mathcal{P}\}$.*

We prove Lemma 5.4 based on the following lemma.

Lemma 5.5. (i) $(\boldsymbol{s})_j^{B,\boldsymbol{x}} \in \mathcal{S}_{|\mathcal{N}(j)|}$.
(ii) $\{f_i\} \in \mathcal{P}_j^{B,\boldsymbol{s}} \Leftrightarrow \{f_i\}^{\boldsymbol{x}} \in \mathcal{P}_j^{B,(\boldsymbol{s})_j^{B,\boldsymbol{x}}}$.

Proof. We first prove part (i). From the definition (5.47) of $\mathcal{S}_r$ we have

$$(\boldsymbol{s})_j^{B,\boldsymbol{x}} \in \mathcal{S}_{|\mathcal{N}(j)|} \Leftrightarrow \sum_{l=1}^{|\mathcal{N}(j)|}{}_2 \left(s_l +_2 (h_{i(j,l),j} \times_q x_{i(j,l)})_B \right) = 1 \,. \tag{5.68}$$

Hence (i) holds since summation part of (5.68) can be calculated as follows.

$$\begin{aligned} &\sum_{l=1}^{|\mathcal{N}(j)|}{}_2 \left(s_l +_2 (h_{i(j,l),j} \times_q x_{i(j,l)})_B \right) \\ &\quad = \left(\sum_{l=1}^{|\mathcal{N}(j)|}{}_2 s_l \right) +_2 \left(\sum_{l=1}^{|\mathcal{N}(j)|}{}_2 (h_{i(j,l),j} \times_q x_{i(j,l)})_B \right) \end{aligned}$$

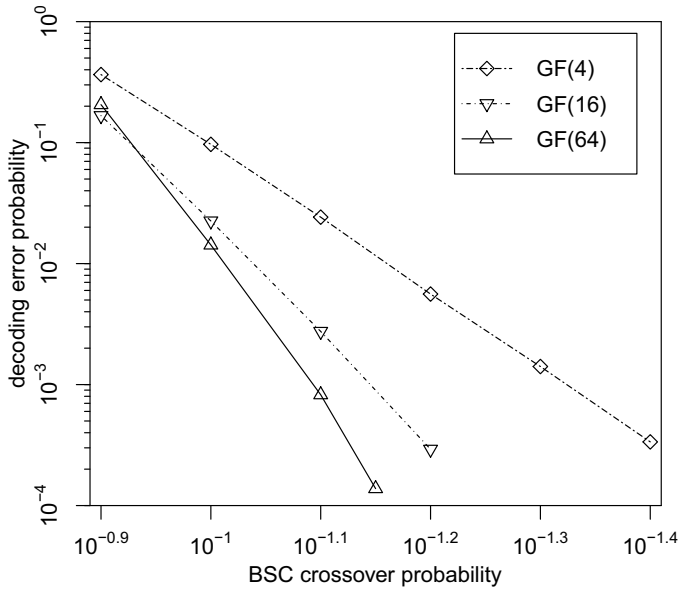

Fig. 5.8. Decoding error probabilities of the proposed scheme for irregular LDPC codes with block length 252 bits and coding rate 1/3.

$$
\begin{aligned}
&= 1 +_2 \left(\sum_{l=1}^{|\mathcal{N}(j)|}{}_q \, h_{i(j,l),j} \times_q x_{i(j,l)} \right)_B \\
&= 1 +_2 (0)_B = 1 \,, \qquad (5.69)
\end{aligned}
$$

where the second equality follows from (5.58).

Next we prove part (ii). From the definition of $\mathcal{P}_j^{B,\boldsymbol{s}}$, we have

$$
\{f_i\} \in \mathcal{P}_j^{B,\boldsymbol{s}} \Leftrightarrow \left\{ \sum_{l=1}^{|\mathcal{N}(j)|} \sum_{z:(h_{i(j,l),j} \times_q z)_B = s_l} f_{i(j,l)}^z \leq |\mathcal{N}(j)| - 1 \right\} . \qquad (5.70)
$$

By letting $\{g_i\}_{i \in \mathcal{I}} \equiv \{f_i\}^{\boldsymbol{x}}$ the summation part of (5.70) can be calculated as follows.

$$
\begin{aligned}
&\sum_{l=1}^{|\mathcal{N}(j)|} \sum_{z:(h_{i(j,l),j} \times_q z)_B = s_l} f_{i(j,l)}^z \\
&\overset{\text{(a)}}{=} \sum_{l=1}^{|\mathcal{N}(j)|} \sum_{w:(h_{i(j,l),j} \times_q (w +_q x_{i(j,l)}))_B = s_l} g_{i(j,l)}^w \\
&\overset{\text{(b)}}{=} \sum_{l=1}^{|\mathcal{N}(j)|} \sum_{w:(h_{i(j,l),j} \times_q w)_B = s_l +_2 (h_{i(j,l),j} \times_q x_{i(j,l)})_B} g_{i(j,l)}^w \,, \qquad (5.71)
\end{aligned}
$$

where (a) follows by letting $z \equiv w +_q x_{i(j,l)}$ and (b) follows from (5.58). We conclude from (5.71) that $\{f_i\} \in \mathcal{P}_j^{B,\boldsymbol{s}} \Leftrightarrow \{g_i\} \in \mathcal{P}_j^{B,(\boldsymbol{s})_j^{B,\boldsymbol{x}}}$,

□

Proof of Lemma 5.4. It is easy to see that the map $(\cdot)_j^{B,\boldsymbol{x}} : \mathcal{S}_{|\mathcal{N}(j)|} \to \mathrm{GF}(2)^{|\mathcal{N}(j)|} : \boldsymbol{s} \mapsto (\boldsymbol{s})_j^{B,\boldsymbol{x}}$ is an injection. Further, from Lemma 5.5 (a) we have $(\boldsymbol{s})_j^{B,\boldsymbol{x}} \in \mathcal{S}_{|\mathcal{N}(j)|}$. Therefore $(\cdot)_j^{B,\boldsymbol{x}}$ is a bijective map $\mathcal{S}_{|\mathcal{N}(j)|} \to \mathcal{S}_{|\mathcal{N}(j)|}$, and it holds for all $j \in \mathcal{J}$, $B \in \mathcal{B}$ and $\{f_i\} \in \mathcal{P}$ that

$$\begin{aligned} \{f_i\} \in \bigcap_{\boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}} \mathcal{P}_j^{B,\boldsymbol{s}} \quad &\overset{\text{(a)}}{\Leftrightarrow} \quad \{f_i\}^{\boldsymbol{x}} \in \bigcap_{\boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}} \mathcal{P}_j^{B,(\boldsymbol{s})_j^{B,\boldsymbol{x}}} \\ &\overset{\text{(b)}}{\Leftrightarrow} \quad \{f_i\}^{\boldsymbol{x}} \in \bigcap_{\boldsymbol{s} \in \mathcal{S}_{|\mathcal{N}(j)|}} \mathcal{P}_j^{B,\boldsymbol{s}} , \end{aligned} \tag{5.72}$$

where (a) follows from Lemma 5.5 and (b) follows since $(\cdot)_j^{B,\boldsymbol{x}}$ is a bijection.

□

Proof of Theorem 5.3. For $a \in \mathcal{X} = \{0,1\}$ and $b \in \mathcal{Y}$, let

$$\sigma_a(b) \equiv \begin{cases} b\,, & a = 0\,, \\ \sigma(b)\,, & a = 1\,. \end{cases} \tag{5.73}$$

We define $\sigma_{\boldsymbol{x}}(\boldsymbol{y}) \equiv \{\sigma_{x_i}(y_i)\}_{i=1,\cdots,n} \in \mathcal{Y}^{mn}$ for

$$\sigma_{x_i}(y_i) \equiv \big(\sigma_{(x_i)_1}((y_i)_1), \cdots, \sigma_{(x_i)_m}((y_i)_m)\big) \in \mathcal{Y}^m\,.$$

From (5.54) and (5.65) it is easy to show that $W(u|t) = W(\sigma_t(u)|0)$, $W^m(\sigma_{x_i}(y_i)|z) = W^m(y_i|z +_q x_i)$ and

$$L(\boldsymbol{x};\boldsymbol{y}) = L(\boldsymbol{0};\sigma_{\boldsymbol{x}}(\boldsymbol{y}))\,. \tag{5.74}$$

Therefore we obtain

$$\begin{aligned} \tilde{L}(\{f_i\}^{\boldsymbol{x}};\sigma_{\boldsymbol{x}}(\boldsymbol{y})) &= \sum_{i=1}^{n}\sum_{z=0}^{q-1} f_i^{z+_q x_i} \log W^m(\sigma_{x_i}(y_i)|z) \\ &= \sum_{i=1}^{n}\sum_{z=0}^{q-1} f_i^{z+_q x_i} \log W^m(y_i|z +_q x_i) \\ &= \sum_{i=1}^{n}\sum_{w=0}^{q-1} f_i^{w} \log W^m(y_i|w) \\ &= \tilde{L}(\{f_i\};\boldsymbol{y})\,, \end{aligned} \tag{5.75}$$

where the third equality follows by letting $w \equiv z +_q x_i$.

Now define the set of received sequences $\boldsymbol{y} \in \mathcal{Y}^{mn}$ such that the decoding error occurs for sent codeword $\boldsymbol{x} \in \mathcal{C}$ by

$$\mathcal{E}(\boldsymbol{x}) \equiv \{\boldsymbol{y} : \exists \{f_i\} \in \mathcal{P},\, \tilde{L}(\{f_i\}; \boldsymbol{y}) > L(\boldsymbol{x}; \boldsymbol{y})\}\,. \tag{5.76}$$

For this $\mathcal{E}(\boldsymbol{x})$ we have

$$\begin{aligned} \boldsymbol{y} \in \mathcal{E}(\boldsymbol{x}) &\Leftrightarrow \exists \{f_i\} \in \mathcal{P} : \tilde{L}(\{f_i\}; \boldsymbol{y}) > L(\boldsymbol{x}; \boldsymbol{y}) \\ &\overset{\text{(a)}}{\Leftrightarrow} \exists \{f_i\} \in \mathcal{P} : \tilde{L}(\{f_i\}^{\boldsymbol{x}}; \sigma_{\boldsymbol{x}}(\boldsymbol{y})) > L(\boldsymbol{0}; \sigma_{\boldsymbol{x}}(\boldsymbol{y})) \\ &\overset{\text{(b)}}{\Leftrightarrow} \sigma_{\boldsymbol{x}}(\boldsymbol{y}) \in \mathcal{E}(\boldsymbol{0})\,, \end{aligned} \tag{5.77}$$

where (a) follows from (5.74) and (5.75) and (b) follows from Lemma 5.4.

Finally we obtain

$$\begin{aligned} \Pr[\hat{\boldsymbol{x}} \neq \boldsymbol{x} | \boldsymbol{x}] &= \sum_{\boldsymbol{y} \in \mathcal{E}(\boldsymbol{x})} 2^{L(\boldsymbol{x}; \boldsymbol{y})} \\ &\overset{\text{(c)}}{=} \sum_{\boldsymbol{y} \in \mathcal{E}(\boldsymbol{x})} 2^{L(0; \sigma_{\boldsymbol{x}}(\boldsymbol{y}))} \\ &\overset{\text{(d)}}{=} \sum_{\boldsymbol{y} \in \mathcal{E}(\boldsymbol{0})} 2^{L(0; \boldsymbol{y})} \\ &= \Pr[\hat{\boldsymbol{x}} \neq \boldsymbol{0} | \boldsymbol{0}]\,, \end{aligned} \tag{5.78}$$

where (c) and (d) follow from (5.74) and (5.77), respectively. □

Chapter 6

Conclusion

In this thesis, we proposed new coding schemes using polar codes and LDPC codes for channel coding with asymmetric channels and for lossy source coding with asymmetric sources and/or asymmetric distortion measures. The proposed schemes are constructed without Gallager's nonlinear mapping and have an advantage in the complexity. We proved that the proposed schemes can achieve the Shannon bound, i.e. the channel capacity or the rate-distortion function, asymptotically. Whereas the optimality for the LDPC codes assumed an infeasible computation, we proposed practical suboptimal algorithms for such a computation and we confirmed by simulation that near optimal performance can be achieved by these suboptimal algorithms.

6.1 Summary of Results

In Chapter 3, we proposed coding schemes using polar codes for channel and lossy source coding based on lossless source coding by polar codes. As in the scheme by Gallager's method, our coding schemes can achieve the Shannon bound with a polynomial time algorithm called successive cancellation. Our scheme has an advantage not only in the complexity, but also in the decoding error probability for channel coding, which is better empirically than that by Gallager's method.

In Chapter 4, we considered channel coding and lossy source coding with LDPC codes. First we discussed a problem of Gallager's method which appears in the case of LDPC codes. Next we proposed new coding schemes using a family of asymptotically optimal variable length lossless codes, arithmetic coding and IAHC coding instead of Gallager's method. We proved that the proposed scheme can achieve the Shannon bound if the LDPC matrix has

a hash property.

In Chapter 5, we considered practical implementation of the scheme using LDPC codes proposed in Chapter 4. First we proposed an approximation algorithm using belief propagation (BP) for marginal probabilities of symbols of LDPC codewords. Next we improved an algorithm for vector-quantization in the lossy source coding. In the proposed scheme, a codeword with small distortion is generated from the output of reinforced belief propagation (RBP). Finally we considered the codeword estimation in channel coding and proposed a new linear programming (LP) decoding scheme for nonbinary LDPC codes. The LP constructed by our scheme has variables and constraints increasing only linearly in the field size and can be computed efficiently by LP solvers.

6.2 Future Works

To construct coding schemes for asymmetric settings, we generated nonuniform codeword distribution by applying a technique of lossless compression. The lossless compression is performed based on arithmetic coding and IAHC coding in LDPC codes. On the other hand, no additional coding is not required to realize lossless compression in the proposed schemes using polar codes.

It is generally known that variable length lossless codes such as arithmetic codes and IAHC codes can achieve better coding rate than fixed length codes such as polar codes. Therefore, the performance may be improved by using variable length codes in lossless compression of the proposed polar codes. When this idea is applied to lossy compression, each information bit can be compressed to length slightly smaller than 1 bit. Similarly when it is applied to channel coding, some bits which were frozen in the original scheme can convey information with length slightly larger than 0 bit. However, it is not obvious whether this idea actually improves the performance or not, since it requires to divide the coding procedure into two parts, whereas the original scheme can perform encoding and decoding in one-shot procedure.

Another important future work can be investigation of VF homophonic coding scheme. As mentioned in Remark 4.1, an error propagation can occur when an FV homophonic coding scheme is applied to channel coding. Furthermore, in our scheme, one symbol (i.e., one inner codeword) in the homophonic coding corresponds to an LDPC code (or it can be a polar code). Then, although the scheme is a variable length code, the length cannot change

smoothly and the advantage of a variable length code is weakened. For these reasons, the message length rather than the code length is desirable to be variable and it is important to devise a VF homophonic coding scheme which is near P-perfect.

The final remark is on a theoretical analysis of performance of coding schemes using lossless compression. When using Gallager's method, the error exponent inevitably becomes the random coding exponent, since nonuniform codewords are generated in a manner somewhat "random" and the same codeword may be generated from different auxiliary codewords with a larger alphabet. On the other hand in usual lossless compressors, different (biased) sequences are related to different messages. Then, the codebook generated by the technique of lossless compression can be regarded as an "expurgated" one of the random codes and hence we may be able to prove the advantage of our scheme in the error probability.

Bibliography

[1] C. E. Shannon, "A mathematical theory of communication," *Bell system technical journal*, vol. 27, 1948.

[2] R. W. Hamming, "Error detecting and error correcting codes," *Bell System Technical Journal*, vol. 26, no. 2, pp. 147–160, 1950.

[3] I. Reed, "A class of multiple-error-correcting codes and the decoding scheme," *IRE Transactions on Information Theory*, vol. 4, no. 4, pp. 38–49, Sep. 1954.

[4] D. E. Muller, "Application of boolean algebra to switching circuit design and to error detection," *IRE Transactions on Electronic Computer*, vol. 3, pp. 6–12, 1954.

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

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

[7] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *Journal of the Society for Industrial and Applied Mathematics*, vol. 8, no. 2, pp. 300–304, 1960.

[8] G. D. Forney, Jr., *Concatenated Codes*. MIT Press, 1966.

[9] P. Elias, "Coding for noisy channels," in *IRE Convention Record, Pt. 4*, 1955, pp. 37–46.

[10] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[11] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *Information Theory, IEEE Transactions on*, vol. 20, no. 2, pp. 284–287, Jan. 1974.

[12] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo-codes," in *Proceedings of IEEE International Conference on Communications*, vol. 2, Geneve,

Switzerland, 1993, pp. 1064–1070.

[13] R. G. Gallager, *Low-density parity-check codes*, ser. M.I.T. Press research monographs. Cambridge: MIT Press, 1963, no. 21.

[14] D. J. C. MacKay, "Good error correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 399–431, 1999.

[15] M. Sipser and D. A. Spielman, "Expander codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1710–1722, 1996.

[16] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 599–618, Sep. 2006.

[17] J. Feldman, M. J. Wainwright, and D. R. Karger, "Using linear programming to decode binary linear codes," *IEEE Trans. Inform. Theory*, vol. 51, pp. 954–972, 2005.

[18] E. Arikan, "Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inform. Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.

[19] D. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the Institute of Radio Engineers*, vol. 40, no. 9, pp. 1098–1101, Sep. 1952.

[20] E. N. Gilbert and E. F. Moore, "Variable length binary encodings," *Bell system technical journal*, vol. 38, pp. 933–967, 1959.

[21] R. C. Pasco, "Source coding algorithms for fast data compression." Ph.D. dissertation, Stanford, CA, USA, 1976.

[22] J. Rissanen, "Generalized kraft inequality and arithmetic coding," *IBM Journal of Research and Development*, vol. 20, pp. 198–203, 1976.

[23] J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 337–343, May 1977.

[24] ——, "Compression of individual sequences via variable-rate coding," *IEEE Transactions on Information Theory*, vol. 24, no. 5, pp. 530–536, 1978.

[25] T. Berger, *Rate distortion theory; a mathematical basis for data compression.* Prentice-Hall, 1971.

[26] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," in *IRE National Convention Record, Pt. 4*, 1959, pp. 142–163.

[27] Y. Matsunaga and H. Yamamoto, "A coding theorem for lossy data compression by ldpc codes," *IEEE Trans. Inform. Theory*, vol. 49, p.

2003, 2003.

[28] D. J. C. MacKay, *Information Theory, Inference & Learning Algorithms.* New York, NY, USA: Cambridge University Press, 2002.

[29] S. Korada and R. Urbanke, "Polar codes are optimal for lossy source coding," *IEEE Trans. Inform. Theory*, vol. 56, no. 4, pp. 1751–1768, 2010.

[30] M. J. Wainwright, "Lossy source encoding via message-passing and decimation over generalized codewords of ldgm codes," in *Proceedings of the International Symposium on Information Theory (ISIT05)*, 2005, pp. 1493–1497.

[31] E. Martinian and M. J. Wainwright, "Analysis of LDGM and compound codes for lossy compression and binning," in *Workshop on Information Theory and Applications (ITA)*, 2006, pp. 229–233.

[32] T. Murayama, "Thouless-anderson-palmer approach for lossy compression." *Physical Review E*, vol. 69, pp. 035 105(1)–035 105(4), 2004. [Online]. Available: http://www.biomedsearch.com/nih/Thouless-Anderson-Palmer-approach-lossy/15089348.html

[33] A. Braunstein, F. Kayhan, G. Montorsi, and R. Zecchina, "Encoding for the blackwell channel with reinforced belief propagation," in *Proceedings of IEEE International Symposium on Information Theory (ISIT07)*, 2007, pp. 1891–1895.

[34] A. Braunstein, F. Kayhan, and R. Zecchina, "Efficient LDPC codes over GF(q) for lossy data compression," in *Proceedings of IEEE International Symposium on Information Theory (ISIT09)*, 2009, pp. 1978–1982. [Online]. Available: http://portal.acm.org/citation.cfm?id=1701116.1701225

[35] R. G. Gallager, *Information Theory and Reliable Communication.* New York: Wiley, 1968.

[36] A. Bennatan and D. Burshtein, "On the application of ldpc codes to arbitrary discrete-memoryless channels," *IEEE Trans. Inform. Theory*, vol. 50, pp. 417–438, 2004.

[37] A. Gupta and S. Verdú, "Nonlinear sparse-graph codes for lossy compression," *IEEE Trans. Inform. Theory*, vol. 55, pp. 1961–1975, May 2009. [Online]. Available: http://portal.acm.org/citation.cfm?id=1669487.1669490

[38] A. Gupta, S. Verdu, and T. Weissman, "Rate-distortion in near-linear time," in *Proceedings of IEEE International Symposium on Information Theory (ISIT08)*, July 2008, pp. 847 –851.

[39] S. Miyake and J. Muramatsu, "A construction of channel code, joint source-channel code, and universal code for arbitrary stationary memoryless channels using sparse matrices," *IEICE Trans. Fundam.*, vol. 92-A, no. 9, pp. 2333–2344, 2009.

[40] ——, "A construction of lossy source code using LDPC matrices," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E91-A, pp. 1488–1501, June 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1521952.1521977

[41] D. Sutter, J. M. Renes, F. Dupuis, and R. Renner, "Achieving the capacity of any DMC using only polar codes," in *Proceedings of IEEE Information Theory Workshop (ITW2012)*, Lausanne, Switzerland, Sep. 2012, pp. 114–118. [Online]. Available: http://arxiv.org/abs/1205.3756v2

[42] C. G. Günther, "A universal algorithm for homophonic coding," in *EUROCRYPT '89*. Springer, 1988, pp. 405–414.

[43] H. N. Jendal, Y. J. B. Kuhn, and J. L. Massey, "An information-theoretic treatment of homophonic substitution," in *EUROCRYPT '89*, 1989, pp. 382–394.

[44] M. Hoshi and T. S. Han, "Interval algorithm for homophonic coding," *IEEE Trans. Inform. Theory*, vol. 47, no. 3, pp. 1021–1031, 2001.

[45] T. S. Han and M. Hoshi, "Interval algorithm for random number generation," *IEEE Trans. Inform. Theory*, vol. 43, no. 2, pp. 599–611, 2006.

[46] M. C. Davey and D. J. C. MacKay, "Low density parity check codes over GF(q)," *IEEE Communications Letters*, vol. 2, no. 6, pp. 165–167, 1998.

[47] M. F. Flanagan, V. Skachek, E. Byrne, and M. Greferath, "Linear-programming decoding of nonbinary linear codes," *IEEE Trans. Inform. Theory*, vol. 55, no. 9, pp. 4134–4154, 2009.

[48] J. Muramatsu and S. Miyake, "Hash property and coding theorems for sparse matrices and maximum-likelihood coding," *IEEE Trans. Inform. Theory*, vol. 56, pp. 2143–2167, May 2010.

[49] S. Arimoto, "An algorithm for calculating the capacity of an arbitrary discrete memoryless channel," *IEEE Transactions on Information Theory*, vol. 18, pp. 14–20, 1972.

[50] R. E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Transactions on Information Theory*, vol. 18, pp. 460–473, 1972.

[51] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed.

Wiley-Interscience, July 2006.

[52] J. Wolfowitz, "The coding of messages subject to chance errors," *Illinois Journal of Mathematics*, vol. 1, no. 4, pp. 591–606, 1957.

[53] I. Csiszar, "Linear codes for sources and source networks: Error exponents, universal coding," *IEEE Transactions on Information Theory*, vol. 28, no. 4, pp. 585–592, Sep. 2006.

[54] B. Mcmillan, "Two inequalities implied by unique decipherability," *IEEE Transactions on Information Theory*, vol. IT-2, pp. 115–116, 1956.

[55] G. Martin, "Range encoding: an algorithm for removing redundancy from a digitised message," in *Video & Data Recording Conference, Southampton*, 1979.

[56] C. E. Shannon, "Communication theory of secrecy systems," *Bell Systems Technical Journal*, vol. 28, pp. 656–715, 1949.

[57] R. Mori and T. Tanaka, "Channel polarization on q-ary discrete memoryless channels by arbitrary kernels," in *Proceedings of IEEE International Symposium on Information Theory (ISIT10)*, 2010, pp. 894–898.

[58] E. Şaşoğlu, E. Telatar, and E. Arikan, "Polarization for arbitrary discrete memoryless channels," in *Proceedings of IEEE Information Theory Workshop (ITW2009)*, 2009, pp. 144–148.

[59] S. B. Korada, "Polar codes for channel and source coding," Ph.D. dissertation, Lausanne, 2009. [Online]. Available: http://library.epfl.ch/theses/?nr=4461

[60] E. Arikan, "Source polarization," in *Proceedings of IEEE International Symposium on Information Theory (ISIT10)*, 2010, pp. 899–903.

[61] I. Tal and A. Vardy, "How to construct polar codes," *submitted to IEEE Trans. Inform. Theory*, 2011. [Online]. Available: http://arxiv.org/abs/arXiv:1105.6164v2

[62] G. Miller and D. Burshtein, "Bounds on the maximum-likelihood decoding error probability of low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, pp. 2696–2710, 2001.

[63] D. J. MacKay and I. Background, "Low density parity check codes over GF(q)," *IEEE Communications Letters*, vol. 2, pp. 70–71, 1996.

[64] M. C. Davey, "Error-correction using low-density parity-check codes," Ph.D. dissertation, Cambridge, 1999.

[65] D. Declercq and M. Fossorier, "Decoding algorithms for nonbinary LDPC codes over GF(q)," *IEEE Trans. Communications*, vol. 55, no. 4, pp. 633–643, April 2007. [Online]. Available: http://publi-etis.ensea.fr/2007/DF07

[66] T. Richardson and R. Urbanke, "Multi-edge type LDPC codes," 2004.

[67] I. Csiszár and J. Körner, *Information theory : coding theorems for discrete memoryless systems*, 3rd ed. Budapest: Akadémiai Kiadó, 1981.

[68] J. Lamperti, *Probability; a survey of the mathematical theory*, 2nd ed., ser. Wiley Series in Probability Statistics. New York: John Wiley & Sons Ltd., 1996.

[69] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Progressive edge-growth tanner graphs," in *Global Telecommunications Conference, 2001*, vol. 2, 2001, pp. 995–1001.

[70] B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms*, 3rd ed. Germany: Springer, 2006. [Online]. Available: http://www.springer.com/math/numbers/book/978-3-540-71843-7

[71] J. Huang and J. Zhu, "Linear time encoding of cycle GF(2^p) codes through graph analysis," *Communications Letters, IEEE*, vol. 10, no. 5, pp. 369–371, may 2006.

[72] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. MIT Press, Sep. 2001. [Online]. Available: http://www.worldcat.org/isbn/0262531968

[73] M. Taghavi and P. Siegel, "Adaptive linear programming decoding," in *Proceedings of IEEE International Symposium on Information Theory (ISIT06)*, 2006, pp. 1374–1378.

[74] S. C. Draper, J. S. Yedidia, and Y. Wang, "Ml decoding via mixed-integer adaptive linear programming," in *Proceedings of IEEE International Symposium on Information Theory (ISIT07)*, 2007, pp. 1656–1660.

[75] K. Yang, X. Wang, and J. Feldman, "A new linear programming approach to decoding linear block codes," *IEEE Trans. Inform. Theory*, vol. 54, no. 3, pp. 1061–1072, 2008.

List of Publications

1. J. Honda and H. Yamamoto, "Variable length lossy coding using an LDPC code," in *Proceedings of IEEE International Symposium on Information Theory (ISIT09)*, Seoul, Korea, 2009, pp. 1973–1977.
2. J. Honda and H. Yamamoto, "Polar coding without alphabet extension for asymmetric channels," in *Proceedings of IEEE International Symposium on Information Theory (ISIT12)*, Cambridge, USA, 2012, pp. 2157–2161.
3. J. Honda and H. Yamamoto, "Fast linear-programming decoding of LDPC codes over GF(2^m)," in *Proceedings of The International Symposium on Information Theory and its Applications (ISITA2012)*, Honolulu, Hawaii, USA, 2012, pp. 754–758.