LOSSLESS DATA COMPRESSION WITH POLAR CODES

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND

ELECTRONICS ENGINEERING

AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Semih Çaycı

August, 2013

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Orhan Arıkan and Prof. Dr. Erdal Arıkan(Advisors)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Sinan Gezici

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Emre Aktaş

Approved for the Graduate School of Engineering and Science:

Prof. Dr. Levent Onural
Director of the Graduate School

ABSTRACT

LOSSLESS DATA COMPRESSION WITH POLAR CODES

Semih Çaycı
M.S. in Electrical and Electronics Engineering
Supervisors: Prof. Dr. Orhan Arıkan and Prof. Dr. Erdal Arıkan
August, 2013

In this study, lossless polar compression schemes are proposed for finite source alphabets in the noiseless setting. In the first part, lossless polar source coding scheme for binary memoryless sources introduced by Arıkan is extended to general prime-size alphabets. In addition to the conventional successive cancellation decoding (SC-D), successive cancellation list decoding (SCL-D) is utilized for improved performance at practical block-lengths. For code construction, greedy approximation method for density evolution, proposed by Tal and Vardy, is adapted to non-binary alphabets. In the second part, a variable-length, zero-error polar compression scheme for prime-size alphabets based on the work of Cronie and Korada is developed. It is shown numerically that this scheme provides rates close to minimum source coding rate at practical block-lengths under SC-D, while achieving the minimum source coding rate asymptotically in the block-length. For improved performance at practical block-lengths, a scheme based on SCL-D is developed. The proposed schemes are generalized to arbitrary finite source alphabets by using a multi-level approach. For practical applications, robustness of the zero-error source coding scheme with respect to uncertainty in source distribution is investigated. Based on this robustness investigation, it is shown that a class of prebuilt information sets can be used at practical block-lengths instead of constructing a specific information set for every source distribution. Since the compression schemes proposed in this thesis are not universal, probability distribution of a source must be known at the receiver for reconstruction. In the presence of source uncertainty, this requires the transmitter to inform the receiver about the source distribution. As a solution to this problem, a sequential quantization with scaling algorithm is proposed to transmit the probability distribution of the source together with the compressed word in an efficient way.

Keywords: Polar codes, source polarization, source coding, lossless data compression.

ÖZET

KUTUPSAL KODLARLA YİTİMSİZ VERİ SIKIŞTIRMA

Semih Çaycı
Elektrik ve Elektronik Mühendisliği, Yüksek Lisans
Tez Yöneticileri: Prof. Dr. Orhan Arıkan ve Prof. Dr. Erdal Arıkan
Ağustos, 2013

Bu çalışmada, gürültüsüz ortamda sonlu kaynak alfabeleri için yitimsiz kutupsal veri sıkıştırma yöntemleri önerilmektedir. İlk kısımda, Arıkan tarafından tanıtılan, ikilik kaynaklar için yitimsiz kutupsal kodlama yöntemi genel asal boyutlu kaynak alfabelerine genişletilmiştir. Konvansiyonel ardışık iptal kod çözücüsüne ek olarak, pratik blok uzunluklarında iyileştirilmiş performans için ardışık iptal liste kod çözücüsü kullanılmıştır. Kod yapımı için, Tal ve Vardy tarafından önerilen yoğunluk evrimi için açgözlü yaklaşıklama algoritması ikilik olmayan kaynak alfabelerine uyarlanmıştır. İkinci bölümde Cronie ve Korada'nın çalışmaları esas alınarak, asal boyutlu alfabeler için değişken uzunluklu, sıfır hata kutupsal sıkıştırma şeması geliştirilmiştir. Önerilen kodlama şemasının ardışık iptal kod çözücüsü ile blok uzunluğuyla asimptotik olarak minimum kaynak kodlama oranına erişmenin yanı sıra pratik blok uzunluklarında minimum kaynak kodlama oranına yakın oranlar sağladığı nümerik olarak gösterilmektedir. Pratik blok uzunluklarında iyileştirilmiş performans için ardışık iptal liste kod çözücüsü tabanlı bir şema geliştirilmiştir. Önerilen yöntemler, çoklu seviye yaklaşımı kullanılarak rastgele sonlu kaynak alfabelerine genelleştirilmiştir. Pratik uygulamalar için, önerilen sıfır hata sıkıştırma yönteminin kaynak dağılımındaki belirsizliğe karşı gürbüzlüğü araştırılmıştır. Bu araştırma esas alınarak, pratik blok uzunluklarında her kaynak dağılımı için özel bir enformasyon kümesi oluşturmak yerine önceden inşa edilmiş enformasyon kümeleri öbeği kullanılabileceği gösterilmiştir. Bu tezde önerilen sıkıştırma yöntemleri evrensel olmadığı için bir kaynağın olasılık dağılımı alıcıda bilinmelidir. Bu durum, kaynak belirsizliği varlığında vericinin alıcıyı kaynak dağılımı hakkında bilgilendirmesini zorunlu kılar. Bu soruna bir çözüm olarak, kaynak olasılık dağılımını etkin bir şekilde sıkıştırılmış kelime ile gönderebilmek için bir ölçeklemeli sırasal basamaklama algoritması önerilmiştir.

Anahtar sözcükler: Kutupsal kodlar, kaynak kutuplaştırma, kaynak kodlama,

yitimsiz veri sıkıştırma.

Acknowledgement

I would like to thank my supervisor Prof. Orhan Arıkan for his persistent help and guidance in all stages of this thesis. This thesis could not have been completed without his support. I would like to thank Prof. Erdal Arıkan for insightful comments and suggestions, which have been key in this thesis. I consider myself very fortunate to work on polar codes under their supervision.

This work was supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) under contract no. 110E243. I am very grateful to TÜBİTAK for funding my thesis.

I would like to dedicate this thesis to the memory of my grandmother.

Contents

List of Figures

Chapter 1

Introduction

The subject of this thesis is the compression of discrete memoryless sources by using polar codes in the noiseless setting. In most practical compression problems, such as discrete cosine transform-based image compression, zero-error compression of memoryless sources over a non-binary alphabet is required. The objective in this problem is to develop compression schemes that provide rates close to minimum source coding rate, the entropy of the source, by using low complexity encoding and decoding algorithms. In this thesis, data compression schemes based on polarization, introduced in [1], that have low complexity encoding and decoding algorithms and an efficient deterministic code construction method are proposed as a solution to the problem.

1.1 Lossless Data Compression: Definitions and Theoretical Limits

In this thesis, lossless compression of discrete memoryless sources with side information in the noiseless setting is considered. Let (X, Y) be a pair of random variables over $\mathcal{X} \times \mathcal{Y}$ with a joint probability mass function $p_{X,Y}$. Throughout the thesis, unless stated otherwise, X represents the source to be compressed, and

Y represents side information. The cardinality of the source alphabet, denoted as $|\mathcal{X}|$, is a positive integer $q < \infty$, and $\mathcal{Y}$ is a finite set. For a positive integer block-length N, N independent and identically distributed (iid) realizations $\{(X_i, Y_i),\ i = 0, 1, \ldots, N-1\}$ from $p_{X,Y}$ are taken and vectors X_0^{N-1} and Y_0^{N-1} are obtained. Side information vector Y_0^{N-1} is assumed to be known at encoder and decoder. This scenario is called (0101)-scheme in [2], and conditional source coding in [3] and [4]. The coding scheme is illustrated in Figure 1.1.

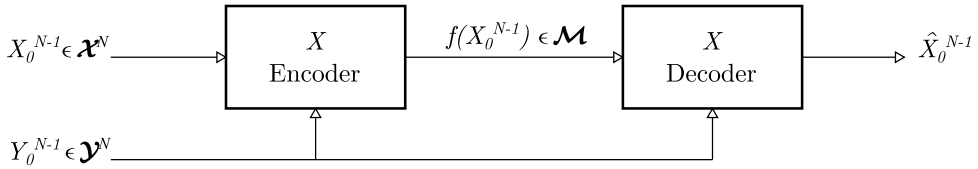

Figure 1.1: Lossless source coding with side information.

An N-length block code with side information is a pair of mappings (f, φ), called encoder and decoder, respectively, such that $f : \mathcal{X}^N \times \mathcal{Y}^N \mapsto \mathcal{M}$ and $\varphi : \mathcal{M} \times \mathcal{Y}^N \mapsto \mathcal{X}^N$ where $\mathcal{M} = \{1, 2, \ldots, M\}$. The error probability associated with the code (f, φ) is $e(f, \varphi) = Pr\{\varphi(f(X^N, Y^N), Y^N) \neq X^N\}$, and the code rate is $R = \frac{1}{N} \log M$.

Theorem 1 (The noiseless coding theorem for discrete memoryless sources)**.** *For any $\epsilon > 0$, there exists an N_0 such that for all $N > N_0$, there exists a code (f, φ) with error probability $e(f, \varphi) \leq \epsilon$ if $R > H(X|Y)$ [2, 5].*

The noiseless coding theorem for discrete memoryless sources states fundamental limit of lossless source coding. The objective is to develop compression schemes that achieve this limit with low complexity encoding and decoding algorithms.

1.2 Review of the Related Work

Channel polarization achieved a significant breakthrough in coding theory for providing the first provably capacity achieving coding scheme with low complexity

encoding and decoding algorithms [1]. The application of polarization in source coding is first considered in [6] and [7], which exploit the duality between channel coding and source coding in the solution of the problem. As a complementary to channel polarization, source polarization was introduced in [8], and a lossless source coding scheme based on source polarization, which asymptotically achieves minimum source coding rate was described. In [9], a zero-error, fixed-to-variable length source coding scheme is developed for binary memoryless sources without considering side information, using a similar approach as in [10] for data compression in the noiseless setting. It was shown in [9] that the proposed scheme provides rates close to minimum source coding rate at practical block-lengths besides achieving it asymptotically in the block-length under successive cancellation decoder (SC-D). In practice, cardinality of the source alphabet can be large; thus a generalization to non-binary alphabets is necessary. In addition, side information, if available, must be exploited for reduced code rates. In this paper, compression schemes for arbitrary finite source alphabets that exploit side information are proposed based on the ideas derived from [9].

Polarization concept was extended to arbitrary discrete memoryless channels in [11], and it is shown that a polarization transform similar to the binary case leads to polarization for prime-size alphabets. For simplicity and efficiency, polarization scheme used in this paper is based on this work.

Successive cancellation decoder (SC-D), proposed in [1], is the first known decoding algorithm for polar codes that achieves channel capacity asymptotically in the block-length with a complexity of $O(N \log N)$. In order to improve performance of polar codes at practical block-lengths, Tal and Vardy proposed successive cancellation list decoder (SCL-D), an adaptation of the list decoding algorithm for Reed-Muller codes proposed in [12], and they numerically showed that SCL-D approaches maximum-likelihood (ML) decoding performance. For improved finite-length performance, SCL-D-based data compression schemes are introduced in this thesis.

Monte Carlo method was used for polar code construction in [1] to estimate Bhattacharyya parameters, which are used in the selection of good channels.

In [13], Mori and Tanaka showed that density evolution can be utilized as a method of deterministic code construction. However, due to its high computational complexity, direct application of their method proved to be impractical for large block-lengths. Tal and Vardy proposed quantization methods to overcome this problem, and they described an efficient polar code construction method for binary discrete memoryless channels in [14]. For efficient code construction, a greedy density evolution method for non-binary alphabets, based on [14], is presented in this thesis.

Robustness of polar source codes with respect to source uncertainty is analyzed in [15]. It was shown that the information set constructed for a q-ary probability distribution p_0 is included in another information set that is constructed for a q-ary distribution p_1 circularly dominated by p_0. In this respect, it was concluded that source coding at rate $R = H(p_1) \geq H(p_0)$ can be performed asymptotically in the block-length. In this thesis, robustness of the proposed scheme is analyzed from two perspectives including this, and efficient schemes for practical applications in the presence of source uncertainty is proposed.

1.3 Outline

The outline of the thesis is as follows.

In Chapter 2, source polarization is briefly reviewed, and a fixed-to-fixed length lossless source coding scheme for non-binary discrete memoryless sources based on source polarization is described as a generalization of [8]. An efficient greedy algorithm based on density evolution is proposed for polar code construction.

In Chapter 3, fixed-to-variable length, zero-error lossless polar compression scheme introduced by Cronie and Korada is generalized to prime-size alphabets. In order to reduce code rate at practical block-lengths, a compression scheme based on SCL-D is proposed. These schemes for prime-size alphabets are generalized to arbitrary finite source alphabets by using a specific scenario for the

compression of correlation sources, and it is shown that minimum source coding rate can be achieved by this scheme. Robustness of the proposed compression scheme with respect to source uncertainty is investigated. Based on this investigation, in order to transmit the source distribution at the expense of extra overhead in the presence of source uncertainty, a sequential quantization with scaling algorithm is proposed. In order to reduce computational complexity in practical applications, a method for constructing and using a pre-constructed information sets is proposed.

Finally, conclusions and future work are presented in Chapter 4.

Chapter 2

Lossless Data Compression with Polar Codes

2.1 Preliminaries

Let (X, Y) be a pair of random variables over $\mathcal{X} \times \mathcal{Y}$ with a joint distribution $p_{X,Y}(x, y)$, where $\mathcal{X} = \{0, 1, \ldots, q-1\}$ for a prime number q, and $\mathcal{Y}$ is a countable set. Following the notation of [8], (X, Y) is considered as a memoryless source with X to be compressed, and Y to be utilized as side information in the compression of X. For a positive integer n and $N = 2^n$, let $\{(X_i, Y_i)\}_{i=0}^{N-1}$ be independent drawings from the source (X, Y). By using the following polarization transformation:

$$\mathbf{G}_N = \left[\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right]^{\otimes n} \mathbf{B}_N, \tag{2.1}$$

where all operations are performed in $GF(q)$, $^{\otimes n}$ is the n^{th} Kronecker power, and $\mathbf{B}_N$ is the bit-reversal operation. the random vector X_0^{N-1} is transformed into U_0^{N-1} as:

$$U_0^{N-1} = X_0^{N-1} \mathbf{G}_N. \tag{2.2}$$

This transformation can be expressed recursively as in Figure 2.1.

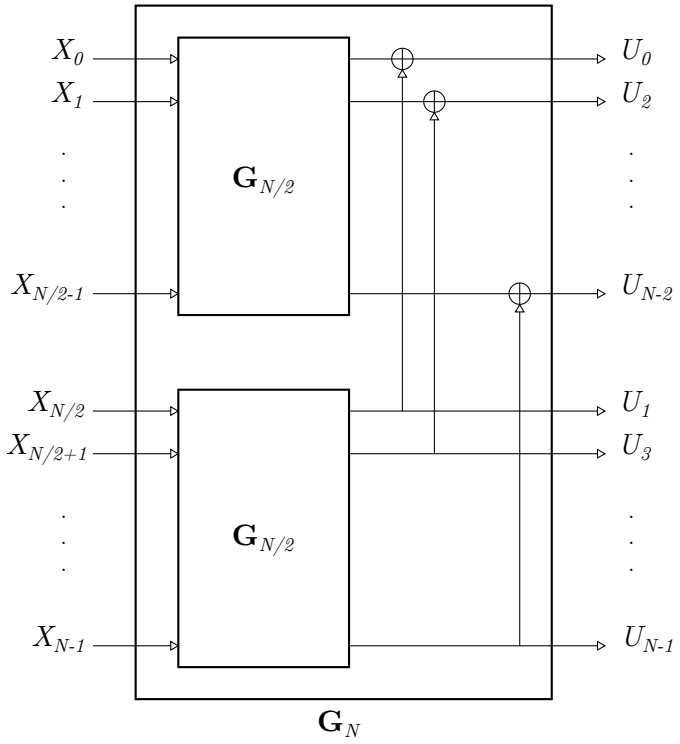

Figure 2.1: Recursive polar transformation of X_0^{N-1}.

The input vector X_0^{N-1} is polarized by this transformation in the following sense:

$$\frac{|\{i : H(U_i|U_0^{i-1}, Y_0^{N-1}) \in [0, \delta)\}|}{2^n} = 1 - H(X|Y), \tag{2.3}$$

and

$$\frac{|\{i : H(U_i|U_0^{i-1}, Y_0^{N-1}) \in (1-\delta, 1]\}|}{2^n} = H(X|Y), \tag{2.4}$$

for any given $\delta > 0$, and $n \to \infty$ [16]. Here the default base of the entropy function is chosen as q.

For a ternary source with distribution $p_X = (0.84,\ 0.09,\ 0.07)$ and block-length $N = 2^{14}$, the set of conditional entropies, $\{H(U_i|U_0^{i-1})\}_{i=0}^{N-1}$, is given in Figure 2.2. For the same source distribution, the sorted conditional entropies with respect to normalized indices together with the entropy of the source is given in Figure 2.3. As $N \to \infty$, the edge of the sorted conditional entropies coincides with the line indicating the entropy of the source.

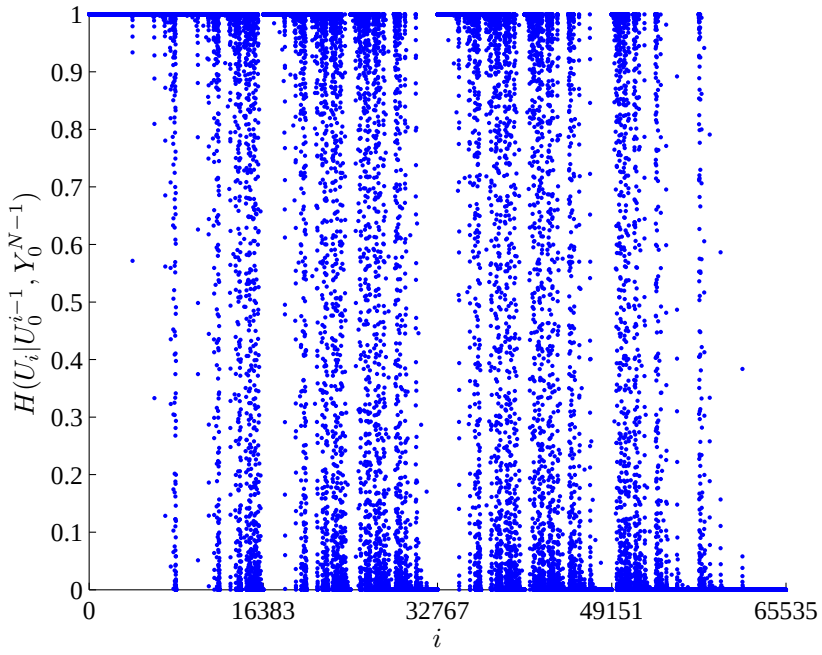

Figure 2.2: The set of conditional entropies for a ternary source X with entropy $H(X) = 0.5$ at block-length $N = 2^{16}$.

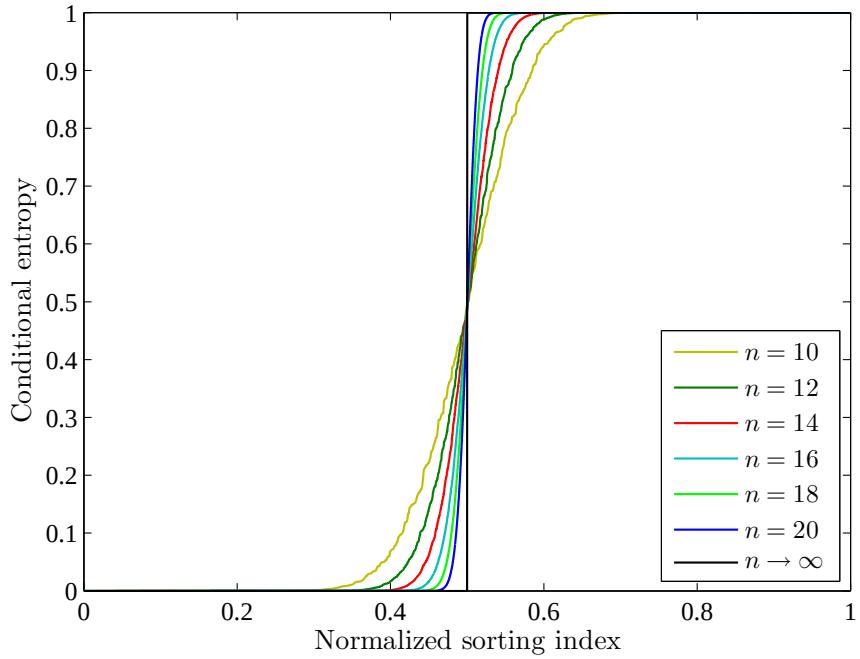

Figure 2.3: Sorted conditional entropies for a ternary source X with entropy $H(X) = 0.5$ at various block-lengths.

2.2 Encoding

Based on source polarization, a lossless source coding scheme for binary memoryless sources is introduced in [8]. The basic idea is to transmit symbols which can be reconstructed using previous symbols, i.e., those with high $H(U_i|U_0^{i-1}, Y_0^{N-1})$.

Let $R \in (0,1)$ be a given rate, and $r = \lceil NR \rceil$. The set of indices corresponding to the r highest $H(U_i|U_0^{i-1}, Y_0^{N-1})$ terms is defined as the information set:

$$\mathcal{I}_{X|Y}(N,R) = \{i \in \{0,1,\ldots,N-1\} : H(U_i|U_0^{i-1}, Y_0^{N-1}) \geq \delta(r)\} \tag{2.5}$$

where $\delta(r)$ corresponds to the rth highest $H(U_i|U_0^{i-1}, Y_0^{N-1})$. The information set is assumed to be known at both encoder and decoder, and shown as $\mathcal{I}_{X|Y}$ in the short form.

In the encoding process, a source realization x_0^{N-1} is transformed into u_0^{N-1} by the polar transformation given in (2.2). Then, the vector consisting of elements of u_0^{N-1} corresponding to the information set $\mathcal{I}_{X|Y}$, denoted as $u_{\mathcal{I}_{X|Y}}$, is transmitted as the codeword. The complexity of encoding is $O(N \log N)$ [1].

2.3 Decoding

In this section, two decoding schemes, namely successive cancellation decoder (SC-D) and successive cancellation list decoder (SCL-D), will be described. In both schemes, recursive computation of path probabilities, i.e., $Pr\{U_0^{i-1} = u_0^{i-1}|Y_0^{N-1} = y_0^{N-1}\}$ for all $i \in \mathcal{I}_{X|Y}^c = \{0,1,\ldots,N-1\}\backslash\mathcal{I}_{X|Y}$, plays a fundamental role. Thus, the recursive computation scheme will be derived first.

The probability of observing a sequence u_0^{N-1} at the output of the polarization transform is defined as follows:

$$P_N(u_0^{N-1}|y_0^{N-1}) \triangleq Pr\{U_0^{N-1} = u_0^{N-1}|Y_0^{N-1} = y_0^{N-1}\}. \tag{2.6}$$

Lemma 1. *The probability of a sequence u_0^{N-1} at the output of the polarization*

transform can be calculated in a recursive way as:

$$P_N(u_0^{N-1}|y_0^{N-1}) = P_{N/2}(u_{0,e}^{N-1} \ominus u_{0,o}^{N-1}|y_0^{N/2-1})P_{N/2}(u_{0,o}^{N-1}|y_{N/2}^{N-1}), \tag{2.7}$$

where $P_1(u|y) = Pr\{X = u|Y = y\}$, $u_{0,e}^{N-1}$ *and* $u_{0,o}^{N-1}$ *denote the elements of* u_0^{N-1} *with even and odd indices, respectively, and* $\ominus$ *denotes subtraction in* $GF(q)$.

The proof directly follows from the recursive structure of the polarization transform in Figure 2.1.

The probability of a subsequence u_0^i is denoted and computed as:

$$P_N^{(i)}(u_0^{i-1}, u_i|y_0^{N-1}) = \sum_{u_{i+1}^{N-1} \in \mathcal{X}^{N-i-1}} P_N(u_0^{N-1}|y_0^{N-1}). \tag{2.8}$$

In order to evaluate probability of a decoding path u_0^i in a recursive way, the recursive computation of joint probability in (2.7) is utilized together with (2.8) as in the following proposition, which is an extension of recursive channel transformations in [1] to non-binary case.

Proposition 1. *For* $i \in \{0, 1, \ldots, N/2 - 1\}$ *and* $N \geq 2$, *marginal probability in (2.8) can be computed recursively as:*

$$\begin{aligned} P_N^{(2i)}(u_0^{2i-1}, u_{2i}|y_0^{N-1}) = & \sum_{u_{2i+1} \in \mathcal{X}} P_{N/2}^{(i)}(u_{0,e}^{2i-1} \ominus u_{0,e}^{2i-1}, u_{2i} \ominus u_{2i+1}|y_0^{N/2-1}) \\ & \cdot P_{N/2}^{(i)}(u_{0,e}^{2i-1}, u_{2i+1}|y_{N/2}^{N-1}), \\ P_N^{(2i+1)}(u_0^{2i}, u_{2i+1}|y_0^{N-1}) = & P_{N/2}^{(i)}(u_{0,e}^{2i-1} \ominus u_{0,e}^{2i-1}, u_{2i} \ominus u_{2i+1}|y_0^{N/2-1}) \\ & \cdot P_{N/2}^{(i)}(u_{0,e}^{2i-1}, u_{2i+1}|y_{N/2}^{N-1}). \end{aligned}$$

Proof. The first equality is proved in the following steps:

$$\begin{aligned} P_N^{(2i)}(u_0^{2i-1}, u_{2i}|y_0^{N-1}) & \overset{a}{=} \sum_{u_{2i+1}^{N-1}} P_{N/2}(u_{0,e}^{N-1} \ominus u_{0,o}^{N-1}|y_0^{N/2-1})P_{N/2}(u_{0,o}^{N-1}|y_{N/2}^{N-1}) \\ & \overset{b}{=} \sum_{u_{2i+1}} \sum_{u_{2i+2,o}^{N-1}} P_{N/2}(u_{0,o}^{N-1}|y_{N/2}^{N-1}) \\ & \qquad \cdot \sum_{u_{2i+2,e}^{N-1}} P_{N/2}(u_{0,e}^{N-1} \ominus u_{0,o}^{N-1}|y_0^{N/2-1}) \end{aligned}$$

$$
\begin{aligned}
&\overset{c}{=} \sum_{u_{2i+1}} \sum_{u_{2i+2,o}^{N-1}} P_{N/2}^{(i)}(u_{0,e}^{2i-1} \ominus u_{0,o}^{2i-1}, u_{2i} \ominus u_{2i+1} | y_0^{N/2-1}) \\
&\qquad\qquad \cdot P_{N/2}(u_{0,o}^{N-1} | y_{N/2}^{N-1}) \\
&\overset{d}{=} \sum_{u_{2i+1}} P_{N/2}^{(i)}(u_{0,e}^{2i-1} \ominus u_{0,o}^{2i-1}, u_{2i} \ominus u_{2i+1} | y_0^{N/2-1}) \\
&\qquad\qquad \cdot P_{N/2}^{(i)}(u_{0,e}^{2i-1}, u_{2i+1} | y_{N/2}^{N-1}).
\end{aligned}
$$

(a) directly follows from Lemma 2.7. From (b) to (c), we devise the fact that marginalizing inner probability with first input argument $u_{0,o}^{N-1} \ominus u_{0,e}^{N-1}$ over $u_{2i+2,e}^{N-1}$ for a fixed $u_{2i+2,o}^{N-1}$ corresponds to marginalization over $u_{2i+2,e}^{N-1} \ominus u_{2i+2,o}^{N-1}$. In (d), it is considered that the marginalization over $u_{2i+2,o}^{N-1}$ does not affect the term with first input argument $u_{0,e}^{2i-1} \ominus u_{0,o}^{2i-1}$.

The second part of the proof is identical with the first part and omitted. □

2.3.1 Successive Cancellation Decoder

In SC-D, each element $\widehat{u}_i$ is reconstructed by using the most probable symbol given side information and previous decisions $\widehat{u}_0^{i-1}$ successively. Note that the decoder has the information set $\mathcal{I}_{X|Y}$ and correct symbols corresponding to the information set, $u_{\mathcal{I}_{X|Y}}$. Hence, if an index i is contained in $\mathcal{I}_{X|Y}$, the decoder directly assigns the corresponding value $\widehat{u}_i = u_i$. In the other case, the probability of each symbol is computed using recursive formulas given in Proposition 1, and the symbol with maximum probability is assigned to $\widehat{u}_i$. A high-level description of the SC-D algorithm is given in Algorithm 1.

SC-D can be implemented with space-complexity $O(N)$ and time-complexity $O(N \log N)$ using the algorithms in [17]. Note that in SC-D, if an incorrect decision is made at phase i, it cannot be corrected at further phases. Moreover, the respective incorrect symbol $\widehat{u}_i$ is used in the decoding of the succeeding symbols, which constitutes error propagation. In order to avoid such situations and improve performance at practical block-lengths at the expense of increased computational complexity, SCL-D is proposed in [17].

Algorithm 1: SC_D($u_{\mathcal{I}_{X|Y}}, y_0^{N-1}$)

input : $u_{\mathcal{I}_{X|Y}}$: Codeword, y_0^{N-1}: Side information
output: $\widehat{x}_0^{N-1}$: Reconstructed sequence

```
for i = 0, 1, ..., N − 1 do
    if i ∈ I_{X|Y} then
        û_i = u_i,
    else
        û_i = argmax_{u_i ∈ X} P_N^{(i)}(û_0^{i−1}, u_i | y_0^{N−1}).
Return x̂_0^{N−1} = û_0^{N−1} G_N^{−1}.
```

2.3.2 Successive Cancellation List Decoder

For a q-ary source code of length N and rate R, the number of decoding paths in the decision tree is q^{NR}, which makes this ML decoder infeasible. However, this problem can be solved by selecting L paths with highest probabilities and terminating the rest at each phase i [17]. This scheme is called successive cancellation list decoding (SCL-D). The parameter L is called the list size and $L = 1$ corresponds to the conventional SC-D.

The basic idea in list decoding is to make decisions based on the probability of a whole sequence $\widehat{u}_0^{N-1}$ instead of individual decisions at each phase i. This procedure can be represented by a q-ary decision tree of depth N. An example of a decision tree for a code with parameters $q = 3,\ N = 2^3,\ L = 4,\ \mathcal{I}_{X|Y} = \{0, 3, 5, 7\},\ u_{\mathcal{I}_{X|Y}} = [2, 1, 0, 1]$ is illustrated in Figure 2.4.

In a decoding tree, each path from a leaf node to the root node constitutes a reconstructed sequence $\widehat{u}_0^{N-1}$. In the example, the correct decision path is shown by red.

At phase i, the path from the parent node to the root node constitutes a decoded sequence $\widehat{u}_0^{i-1}$. If i is contained in $\mathcal{I}_{X|Y}$, one node is appended to each leaf node at phase $i-1$ with value $\widehat{u}_i$. If i is not contained in $\mathcal{I}_{X|Y}$, q nodes are appended to each node at phase $i-1$, representing each possible symbol for $\widehat{u}_i$, together with the path probability as in (2.8). Continuing this way, the decoded

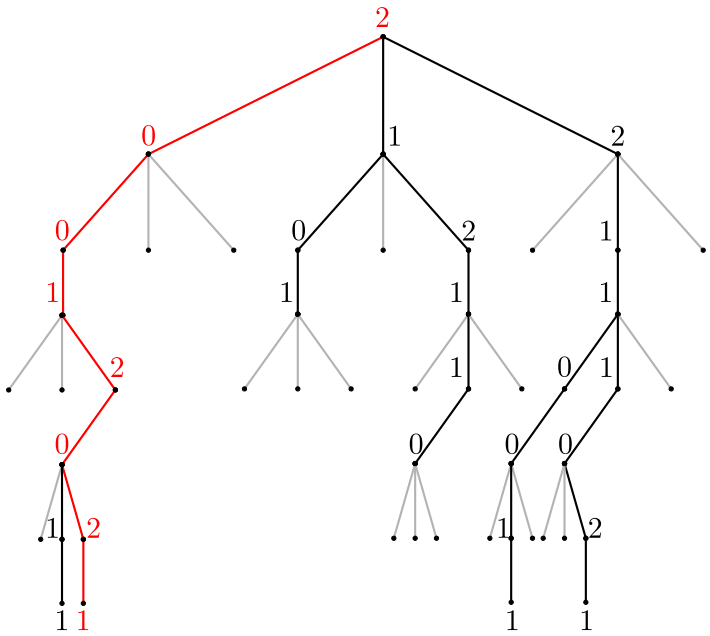

Figure 2.4: An example SCL-D tree for $q = 3$, $L = 4$ and $N = 2^3$.

sequence $\widehat{u}_0^{N-1}$ corresponds to the path with highest probability:

$$\widehat{u}_0^{N-1} = \underset{\tilde{u}_0^{N-1} \in \mathcal{X}^N}{\operatorname{argmax}} \; P_N^{(N-1)}(\tilde{u}_0^{N-2}, \tilde{u}_{N-1}) \tag{2.9}$$

A high-level description of SCL-D is given in Algorithm 2. For the implementation of low-level functions, refer to [17]. SCL-D can be implemented with $O(LN \log N)$ complexity using the lazy-copy algorithm described in [17].

As a remark, it must be mentioned that Algorithm 14 in [17] is prone to numerical unstability at large block-lengths in the case of q-ary source coding. As a solution to this problem, it is recommended that all probability values are converted to log domain, and appropriate scaling is performed in Line 14 of Algorithm 14.

Algorithm 2: SCL_D($u_{\mathcal{I}_{X|Y}}, y_0^{N-1}, L$)

input : $u_{\mathcal{I}_{X|Y}}$: Codeword, y_0^{N-1}: Side information, L: List size
output: $\widehat{x}_0^{N-1}$: Reconstructed sequence

// $\mathcal{L}_i$: The set of active paths at phase i.
for $i = 0, 1, \ldots, N-1$ **do**
  if $i \in \mathcal{I}_{X|Y}$ **then**
    Append u_i to each $\widehat{u}_0^{i-1}[l] \in \mathcal{L}_{i-1}$, and obtain $(\widehat{u}_0^{i-1}[l], u_i)$
  else
    Append all $\widehat{u}_i \in \mathcal{X}$ to each $\widehat{u}_0^{i-1}[l] \in \mathcal{L}_{i-1}$;
    Calculate $P_N^{(i)}(\widehat{u}_0^{i-1}[l], \widehat{u}_i | y_0^{N-1})$ for all $(\widehat{u}_0^{i-1}[l], \widehat{u}_i) \in \mathcal{L}_i$;
    Prune all but L paths with highest probabilities.
Return $\widehat{x}_0^{N-1} = (\underset{\widehat{u}_0^{N-1} \in \mathcal{L}_{N-1}}{\operatorname{argmax}} P_N^{(N-1)}(\widehat{u}_0^{N-1} | y_0^{N-1}))\mathbf{G}_N^{-1}$.

2.4 Code Construction

In polar coding, it is assumed that the information set, $\mathcal{I}_{X|Y}$, is known at both encoder and decoder. In order to construct $\mathcal{I}_{X|Y}$ as in (2.5), the set of conditional entropies $\{H(U_i|U_0^{i-1}, Y_0^{N-1})\}_{i=0}^{N-1}$ must be available. Except a small class of examples, e.g., binary erasure channels in channel coding, analytic solutions do not exist for computing conditional entropy values. As a solution, density evolution is proposed for computing $\{H(U_i|U_0^{i-1}, Y_0^{N-1})\}_{i=0}^{N-1}$, and constructing $\mathcal{I}_{X|Y}$ for any generic distribution $p_{X,Y}(x, y)$ [13, 14]. In this section, a greedy approximation algorithm for computing $\{H(U_i|U_0^{i-1}, Y_0^{N-1})\}_{i=0}^{N-1}$ using density evolution is proposed for prime-size alphabets.

2.4.1 Density Evolution

Let $\mathbf{p} = (p_0, p_1, \ldots, p_{q-1}) \in \mathbb{R}^q$ be a q-dimensional probability vector, i.e., $p_i > 0$, $\forall i$ and $\sum_{i=0}^{q-1} p_i = 1$. q-ary entropy function is defined as follows:

$$\mathcal{H}(\mathbf{p}) = \sum_{i=0}^{q-1} p_i \log \frac{1}{p_i} \tag{2.10}$$

For $f : \mathcal{X} \mapsto \mathbb{R}$, a function defined on $\mathcal{X} = \{0, 1, \dots\ q-1\}$, $[f(x)]_{x=0}^{q-1}$ represents q-dimensional vector formed by the values of f:

$$[f(x)]_{x=0}^{q-1} = [f(0), f(1), \dots, f(q-1)]$$

Let (X, Y) be a pair of random variables over $\mathcal{X} \times \mathcal{Y}$, $\mathcal{X} = \{0, 1, \dots, q-1\}$, with joint distribution $p_{X,Y}(x, y)$. The conditional entropy $H(X|Y)$ is computed as follows:

$$\begin{aligned} H(X|Y) &= \sum_{y \in \mathcal{Y}} p_Y(y) \sum_{x \in \mathcal{X}} p_{X|Y}(x|y) \log \frac{1}{p_{X|Y}(x|y)} \\ &= \sum_{y \in \mathcal{Y}} p_Y(y) \mathcal{H}([p_{X|Y}(x|y)]_{x=0}^{q-1}) \end{aligned} \tag{2.11}$$

Hence, for the computation of $H(X|Y)$, the set of $(q+1)$-dimensional vectors $\{(p_Y(y), [p_{X|Y}(x|y)]_{x=0}^{q-1})\}_{y \in \mathcal{Y}}$ is needed. In density evolution, the objective is to obtain $\{(p_{U_0^{i-1}, Y_0^{N-1}}(u_0^{i-1}, y_0^{N-1}), [p_{U_i|U_0^{i-1}, Y_0^{N-1}}(u|u_0^{i-1}, y_0^{N-1})]_{u=0}^{q-1})\}_{u_0^{i-1} \in \mathcal{X}^i, y_0^{N-1} \in \mathcal{Y}^N}$ by evolving the distributions of the source through the polar transform so that $H(U_i|U_0^{i-1}, Y_0^{N-1})$ can be computed.

Probability distributions of the pair of random variables (X, Y) are expressed in a list as follows:

$$X|Y \sim \sum_{y \in \mathcal{Y}} p_Y(y) Q([p_{X|Y}(x|y)]_{x=0}^{q-1}) \tag{2.12}$$

Remark: Consider a random variable Z that is independent from X and Y. Then, by notation, $X|Y = X|Y, Z$ since for a fixed y, $[p_{X|Y,Z}(x|y, z)]_{x=0}^{q-1} = [p_{X|Y}(x|y)]_{x=0}^{q-1}$ for all $z \in \mathcal{Z}$, and $p_{Y,Z}(y, z) = p_Y(y) p_Z(z)$ for all y, z, which together imply that Z has no effect in the computation of conditional entropy.

In order to investigate how input distributions are transformed in polar transformation, let us consider the basic transform given in Figure 2.5 first. Assume that $|\mathcal{X}| = q$ is a prime number and (X_0, Y_0), (X_1, Y_1) are independently drawn from $p_{X,Y}(x, y)$, i.e., they can be expressed identically in the form of 2.12.

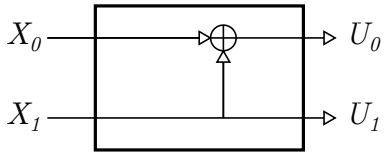

Figure 2.5: Basic polar transform.

The first operation is defined as follows:

$$\begin{aligned} U_0|Y_0,Y_1 &\sim \sum_{y_0,y_1\in\mathcal{Y}} p_{Y_0,Y_1}(y_0,y_1)Q([p_{X_0\oplus X_1|Y_0,Y_1}(x|y_0,y_1)]_{x=0}^{q-1}) \\ &\triangleq \sum_{y_0\in\mathcal{Y}} p_Y(y_0)Q([p_{X|Y}(x|y_0)]_{x=0}^{q-1}) \circledast \sum_{y_1\in\mathcal{Y}} p_Y(y_1)Q([p_{X|Y}(x|y_1)]_{x=0}^{q-1}) \end{aligned} \tag{2.13}$$

Proposition 2. *The expression in (2.13) can be written in the following form:*

$$U_0|Y_0,Y_1 \sim \sum_{y_0,y_1} p_Y(y_0)p_Y(y_1)Q([\sum_{z=0}^{q-1} p_{X|Y}(z|y_0)p_{X|Y}(x\ominus z|y_1)]_{x=0}^{q-1})$$

Proof. Since (X_0,Y_0) and (X_1,Y_1) are independent and identically distributed, $p_{Y_0,Y_1}(y_0,y_1) = p_Y(y_0)p_Y(y_1)$ for all $y_0,y_1 \in \mathcal{Y}$, and the probability distribution $p_{X_0\oplus X_1|Y_0,Y_1}(x|y_0,y_1)$ can be written as the convolution of $p_{X|Y}(x|y_0)$ and $p_{X|Y}(x|y_1)$. □

By using the remark, $U_0|Y_0,Y_1$ can be expressed as $(X_0|Y_0,Y_1)\oplus(X_1|Y_0,Y_1) = X_0\oplus X_1|Y_0,Y_1$. Note that $\circledast$ is a commutative operation.

The second operation is defined as follows:

$$\begin{aligned} U_1|U_0,Y_0,Y_1 &\sim \sum_{z\in\mathcal{X}}\sum_{y_0,y_1\in\mathcal{Y}} p_{X_0\oplus X_1,Y_0,Y_1}(z,y_0,y_1)Q([p_{X_1|X_0\oplus X_1,Y_0,Y_1}(x|z,y_0,y_1)]_{x=0}^{q-1}) \\ &\triangleq \sum_{y_0\in\mathcal{Y}} p_Y(y_0)Q([p_{X|Y}(x|y_0)]_{x=0}^{q-1}) \circledcirc \sum_{y_1\in\mathcal{Y}} p_Y(y_1)Q([p_{X|Y}(x|y_1)]_{x=0}^{q-1}) \end{aligned}$$

Proposition 3. *The operation $\circledcirc$ can be written in the following form:*

$$U_1|U_0,Y_0,Y_1 \sim \sum_{z}\sum_{y_0,y_1}\left([\sum_{u=0}^{q-1} p_{X|Y}(u|y_0)p_{X|Y}(z\ominus u|y_1)]p_Y(y_0)p_Y(y_1)\right).$$
$$Q\left(\left[\frac{p_{X|Y}(x|y_1)p_{X|Y}(z\ominus x|y_0)}{[\sum_{u=0}^{q-1} p_{X|Y}(u|y_0)p_{X|Y}(z\ominus u|y_1)]}\right]_{x=0}^{q-1}\right)$$

The proof is similar to Proposition 2.

Note that polar transform is a successive application of the basic transform given in Figure 2.5. The following theorem states that $\{H(U_i|U_0^{i-1}, Y_0^{N-1})\}_{i=0}^{N-1}$ can be computed by using the density evolution operations successively through the polar transform.

Theorem 2. *Let $\{(X_i, Y_i)\}_{i=0}^{N-1}$ be iid pairs of random variables with probability distribution $p_{X,Y}(x, y)$. Operations $\circledast$ and $\boxast$ applied successively through the polar transform suffice to obtain $\{U_i|U_0^{i-1}, Y_0^{N-1}\}_{i=0}^{N-1}$.*

Proof. The proof will be based on induction. Proposition 2 and Proposition 3 together prove the initial step, i.e., for n = 1, $H(U_0|Y_0^1)$ and $H(U_1|U_0, Y_0^1)$ can be computed by using density evolution. For the inductive step, consider 2.6. Assume that the hypothesis is true for $n-1$. In this case, the input variables $\{X_i|Y_i\}_{i=0}^{N/2-1}$ are transformed into $\{R_i|R_0^{i-1}, Y_0^{N-1}\}_{i=0}^{N/2-1}$, and, similarly, $\{X_i|Y_i\}_{i=N/2}^{N-1}$ are transformed into $\{S_i|S_0^{i-1}, Y_0^{N-1}\}_{i=0}^{N/2-1}$. At the last step of polar transform, these vectors are combined as in Figure 2.6.

For $i = 0, 1, \ldots, N/2-1$,

$$\begin{aligned}(R_i\big|R_0^{i-1}Y_0^{N/2-1}) \oplus (S_i\big|S_0^{i-1}Y_{N/2}^{N-1}) &= (R_i|R_0^{i-1}S_0^{i-1}Y_0^{N-1}) \oplus (S_i|R_0^{i-1}S_0^{i-1}Y_0^{N-1}) \\ &= R_i \oplus S_i|R_0^{i-1}S_0^{i-1}Y_0^{N-1} \\ &= U_{2i}|R_0^{i-1} \oplus S_0^{i-1}, S_0^{i-1}Y_0^{N-1} \\ &= U_{2i}|U_{0,e}^{2i-1}U_{0,o}^{2i-1}Y_0^{N-1} \\ &= U_{2i}|U_0^{2i-1}Y_0^{N-1}\end{aligned}$$

In the first line above, the fact that R_0^{i-1} is independent from S_0^{i-1} and $Y_{N/2}^{N-1}$ is used. The operations on the variables are matched to the density evolution operations similar to the $n = 1$ case. For odd indices,

$$\begin{aligned}S_i|S_0^{i-1}\Big|R_i \oplus S_i|R_0^{i-1}, S_0^{i-1}, Y_0^{N-1} &= S_i|R_i \oplus S_i, R_0^{i-1}, S_0^{i-1}, Y_0^{N-1} \\ &= S_i|U_{2i}, U_0^{2i-1}, Y_0^{N-1} \\ &= U_{2i+1}|U_0^{2i}, Y_0^{N-1}\end{aligned}$$

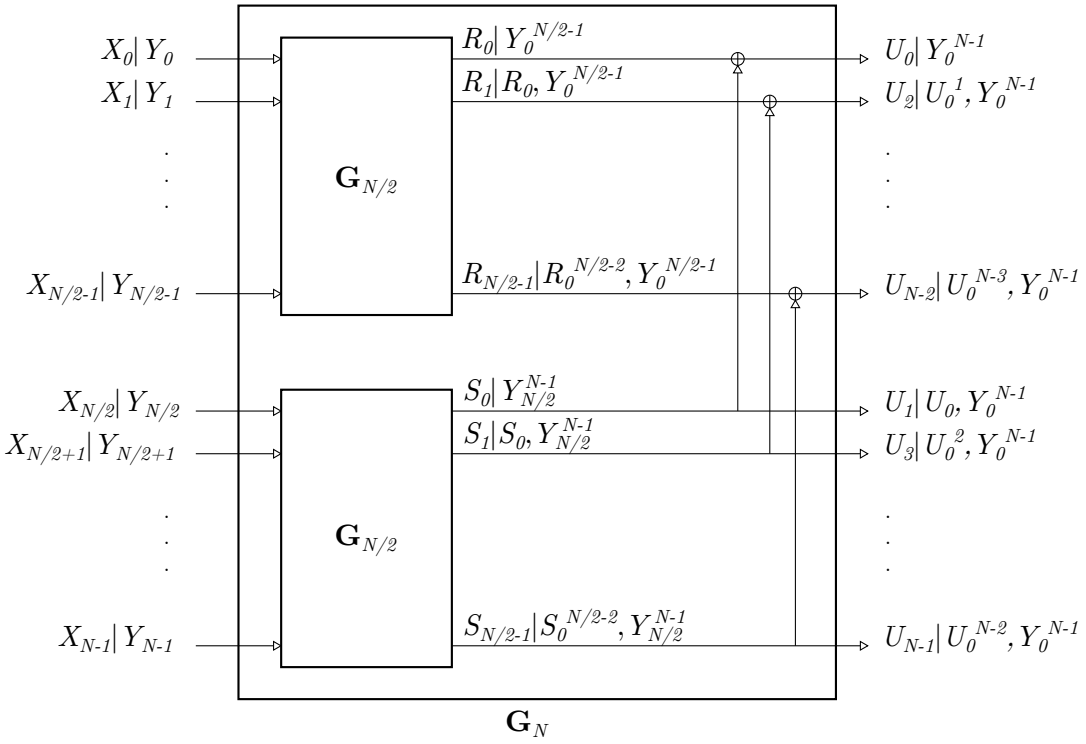

Figure 2.6: Density evolution at block-length N.

Hence, all conditional entropies can be computed using $\circledast$ and $\boxtimes$ operations. $\square$

Theorem 2 indicates that $\{U_i|U_0^{i-1}, Y_0^{N-1}\}_{i=0}^{N-1}$ can be computed by successive application of the density evolution operations. Next, the procedure to obtain $U_i|U_0^{i-1}, Y_0^{N-1}$ will be described. Let the right-hand side of (2.12) be denoted as $\chi_i^{(0)}$. In order to compute $U_i|U_0^{i-1}, Y_0^{N-1}$ where the binary expansion of i is $(b_0 b_1 \dots b_{n-1})_2$, the following procedure is applied [7]:

$$\chi_i^{(k+1)} = \begin{cases} \chi_i^{(k)} \circledast \chi_i^{(k)} & \text{, if } \quad b_k = 0 \\ \chi_i^{(k)} \boxtimes \chi_i^{(k)} & \text{, if } \quad b_k = 1 \end{cases} \tag{2.14}$$

for $k = 0, 1, \dots, n-1$. The output of this procedure is $\chi_i^{(n)}$ such that $U_i|U_0^{i-1}, Y_0^{N-1} \sim \chi_i^{(n)}$.

In order to estimate the complexity of density evolution, the growth in the alphabet sizes of the conditioned terms is used. In Figure 2.5, the alphabet sizes of the conditioned terms of $U_0|Y_0, Y_1$ and $U_1|U_0, Y_0, Y_1$ are $|\mathcal{Y}|^2$

and $|\mathcal{Y}|^2 q$, respectively. Note that if two different y, say y_1 and y_2, in the RHS of (2.12) have the same $[p_{X|Y}(x|y)]_{x=0}^{q-1}$, then they can be unified as $[p_Y(y_1) + p_Y(y_2)]Q([p_{X|Y}(x|y_1)]_{x=0}^{q-1})$, which implies that the alphabet sizes are upper bounds. At block-length N, the alphabet size becomes $O(q^{N-1})$. Thus, the direct application of density evolution in polar code construction infeasible. Approximation methods are proposed to solve this problem [14]. In the following subsection, approximation methods for q-ary code construction based on [14] will be proposed.

2.4.2 Greedy Approximation Algorithm for Code Construction

The basic idea in the approximation algorithm is to unify symbols $y, \bar{y} \in \mathcal{Y}$, whose unification changes the entropy of $X|Y$ minimally, successively until the alphabet size becomes lower than a given parameter μ. By such an approximation, the growth in the number of $(p_Y(y), [p_{X|Y}(x|y)]_{x=0}^{q-1})$ through the polar transform can be controlled, hence efficient code construction can be performed.

Assume that we have $X|Y \sim \sum_{y \in \mathcal{Y}} p_Y(y) Q([p_{X|Y}(x|y)]_{x=0}^{q-1})$ where $|\mathcal{Y}| > \mu$ for a given positive integer μ. Denote the conditional probability of X given a realization y by $\mathbf{p}_{X|y}$, i.e., $\mathbf{p}_{X|y} = [p_{X|Y}(x|y)]_{x=0}^{q-1}$. For $y, \bar{y} \in \mathcal{Y}$, the following operation is performed to reduce the alphabet size of the side information:

$$p_Y(y) Q(\mathbf{p}_{X|y}) + p_Y(\bar{y}) Q(\mathbf{p}_{X|\bar{y}}) \mapsto (p_Y(y) + p_Y(\bar{y})) Q(\mathbf{p}) \tag{2.15}$$

for a valid probability vector $\mathbf{p}$. The new alphabet, $\widehat{\mathcal{Y}} = \mathcal{Y} \backslash \{y, \bar{y}\} \cup \{\widehat{y}\}$, has cardinality $|\mathcal{Y}| - 1$. In greedy approximation algorithm, symbol pairs $y, \bar{y}$ and unified distribution $\mathbf{p}$ are chosen in an intelligent way at each step, and the alphabet size is reduced by one. By successively applying the operation in (2.15), the alphabet size is reduced below the given parameter, i.e., $|\widehat{\mathcal{Y}}| < \mu$.

A different approximation algorithm is proposed for non-binary polar code construction in [18]. The main difference is that [18] involves parametrized quantization levels, whereas the method presented here is based on a greedy algorithm

as in [14] and [19]. The algorithm that will be proposed here can be considered as a modification of the mass merging algorithm in [19]. For detailed description and analysis of mass merging and mass transportation algorithms, we refer to [19].

Let $X|Y \sim \chi$ be a source as defined in (2.12) with $\chi = \sum_{y \in \mathcal{Y}} p_Y(y) Q(\mathbf{p}_{X|Y}(x|y))$, $y, \bar{y} \in \mathcal{Y}$ be given symbols for merging, and $\gamma \in [0,1]$. Merging of these masses with weight γ is the following transformation:

$$p_Y(y)Q(\mathbf{p}_{X|y}) + p_Y(\bar{y})Q(\mathbf{p}_{X|\bar{y}}) \mapsto (p_Y(y) + p_Y(\bar{y}))Q(\gamma \mathbf{p}_{X|y} + (1-\gamma)\mathbf{p}_{X|\bar{y}}) \quad (2.16)$$

This transformation corresponds to mass transportation if $\gamma = 0$ and $\gamma = 1$, and degrading approximation (and mass merging as defined in [19]) if $\gamma = \frac{p_Y(y)}{p_Y(y)+p_Y(\bar{y})}$. In order to define the approximation error due to (2.16), consider the following function:

$$\begin{aligned} f_{y,\bar{y}}(\gamma) = (p_Y(y) + p_Y(\bar{y}))\mathcal{H}(\gamma \mathbf{p}_{X|y} + (1-\gamma)\mathbf{p}_{X|\bar{y}}) \quad (2.17) \\ - \; p_Y(y)\mathcal{H}(\mathbf{p}_{X|y}) - p_Y(\bar{y})\mathcal{H}(\mathbf{p}_{X|\bar{y}}). \end{aligned}$$

The approximation error is defined as the change in the conditional entropy due to the mass merging transformation:

$$\epsilon_{y,\bar{y}}(\gamma) = |f_{y,\bar{y}}(\gamma)|. \quad (2.18)$$

For any $y, \bar{y} \in \mathcal{Y}$, the following proposition holds:

Proposition 4. *$f_{y,\bar{y}}(\gamma) = 0$ has exactly one root in the interval $(0,1]$.*

Proof. The case $\mathcal{H}(\mathbf{p}_{X|y}) = \mathcal{H}(\mathbf{p}_{X|\bar{y}})$ is trivial: $\gamma = 1$ satisfy the claim. In order to prove that the proposition holds for the non-trivial case, first, it is proved that $f_{y,\bar{y}}$ is a concave function of $\gamma \in [0,1]$. For $0 \leq \lambda, \gamma_1, \gamma_2 \leq 1$ and $\overline{\lambda} = 1 - \lambda$,

$$\begin{aligned} f_{y,\bar{y}}(\lambda\gamma_1 + \overline{\lambda}\gamma_2) &= [p_Y(y) + p_Y(\bar{y})]\mathcal{H}(\tilde{\mathbf{p}}) - \sum_{y' \in \{y,\bar{y}\}} p_Y(y')\mathcal{H}(\mathbf{p}_{X|y'}) \\ &\geq \lambda f_{y,\bar{y}}(\gamma_1) + \overline{\lambda} f_{y,\bar{y}}(\gamma_2) \end{aligned}$$

where $\tilde{\mathbf{p}} = \lambda(\gamma_1 \mathbf{p}_{X|y} + \overline{\gamma}_1 \mathbf{p}_{X|\bar{y}}) + \overline{\lambda}(\gamma_2 \mathbf{p}_{X|y} + \overline{\gamma}_2 \mathbf{p}_{X|\bar{y}})$. The inequality follows from the concavity of $\mathcal{H}$.

In the non-trivial case, $f_{y,\bar{y}}(\gamma)$ has different signs at the boundary points, $\gamma = 0$ and $\gamma = 1$. Since $f_{y,\bar{y}}$ is a concave and continuous function of $\gamma \in [0,1]$, this property implies that $f_{y,\bar{y}}$ has only one zero crossing on $[0,1]$. □

Proposition 4 implies that for any $y, \bar{y} \in \mathcal{Y}$, the approximation error $\epsilon_{y,\bar{y}}(\gamma)$ can attain its minimum value 0 by solving the logarithmic equation $f_{y,\bar{y}}(\gamma) = 0$ that has a unique solution on $[0,1]$. Fast-converging bisection method can be used to solve this problem [20]. Since this method is based on a greedy algorithm, the error in $H(U_i|U_0^{i-1}, Y_0^{N-1})$ due to the approximation cannot be analyzed easily. However, the approximation error $\epsilon_{y,\bar{y}}(\gamma)$ has a close relation to the error in the computation of $H(U_i|U_0^{i-1}, Y_0^{N-1})$ as numerical examples will indicate.

For a given source $X|Y \sim \chi$ and pair of symbols $y, \bar{y}$, mass merging operation is summarized in Algorithm 3.

Algorithm 3: merge$(y, \bar{y})$

input : $y, \bar{y}$: Symbols to be merged

1 Find γ^* such that $f_{y,\bar{y}}(\gamma^*) = 0$;
2 Set $\mathbf{p}_{X|y'} = \gamma^* \mathbf{p}_{X|y} + (1-\gamma^*)\mathbf{p}_{X|\bar{y}}$;
3 Set $\chi = \sum_{y'' \in \mathcal{Y} \setminus \{y,\bar{y}\}} p_Y(y'') Q(\mathbf{p}_{X|y''}) + [p_Y(y) + p_Y(\bar{y})] Q(\mathbf{p}_{X|y'})$.

Algorithm 3 provides the basic tool to reduce the alphabet size of the side information Y by one without changing the entropy of the source for any $y, \bar{y}$. Thereafter, the question of how to choose the symbols y_1, y_2 to be merged comes up. At this point, our consideration is to deviate a source as little as possible so that the effect of mass merging on $H(U_i|U_0^{i-1}, Y_0^{N-1})$ is kept small. For any $y, \bar{y} \in \mathcal{Y}$, the following error function is defined:

$$\widehat{\epsilon}_{y,\bar{y}} = \max(|p_Y(\bar{y})[\mathcal{H}(\mathbf{p}_{X|y}) - \mathcal{H}(\mathbf{p}_{X|\bar{y}})]|, |p_Y(y)[\mathcal{H}(\mathbf{p}_{X|\bar{y}}) - \mathcal{H}(\mathbf{p}_{X|y})]|) \quad (2.19)$$

$\widehat{\epsilon}_{y,\bar{y}}$ corresponds to the maximum of the approximation errors if y and $\bar{y}$ are unified using mass transportation algorithm. The pair $y, \bar{y}$ that has the minimum error (2.19) is chosen as the input to the mass merging algorithm since their unification

is likely to affect $H(U_i|U_0^{i-1}, Y_0^{N-1})$ minimally. Therefore, for each $y \in \mathcal{Y}$, the pair is defined as follows:

$$\pi(y) = \underset{\widehat{y} \in \mathcal{Y} \setminus \{y\}}{\operatorname{argmin}} \widehat{\epsilon}_{y,\widehat{y}} \tag{2.20}$$

Given $X|Y \sim \chi$, the following symbol y is chosen for merging together with its pair $\pi(y)$:

$$y = \underset{y \in \mathcal{Y}}{\operatorname{argmin}} \widehat{\epsilon}_{y,\pi(y)} \tag{2.21}$$

Since the greedy approximation method calls the mass merging function successively, the above procedure for finding the most appropriate pair of symbols for merging is inefficient. In order to improve efficiency at the expense of a slight performance degradation, the following procedure, an extension of the approach in [14], can be applied:

- Sort $(p_Y(y_i), \mathbf{p}_{X|y_i})$ such that $\mathcal{H}(\mathbf{p}_{X|y_i}) \leq \mathcal{H}(\mathbf{p}_{X|y_{i+1}})$ for all $i \in \{0, 1, \ldots, |\mathcal{Y}| - 1\}$,
- Compute $\widehat{\epsilon}_{y_i,y_{i+1}}$ for all i,
- Choose y_i, y_{i+1} that has minimum $\widehat{\epsilon}_{y_i,y_{i+1}}$ for mass merging.

In this approach, $\widehat{\epsilon}_{y_i,y_{i+1}}$ values can be computed once, and after each application of the merge(y, y_{i+1}) function, $\widehat{\epsilon}_{y_{i-1},y_i}$ and $\widehat{\epsilon}_{y_i,y_{i+1}}$ are updated. In Algorithm 4, the greedy mass merging function that follows the second approach is summarized.

Combining the mass merging algorithm with the code construction scheme (2.14), the efficient polar code construction algorithm is implemented as in Algorithm 5.

Using mass merging algorithm with parameter μ on $\chi_i^{(k)}$, the size of $\chi_i^{(k+1)}$ is bounded above by $\mu^2 q$, which follows from (2.14). Therefore, complexity is controlled through the polar transform. Since the size of $\chi_i^{(k)}$, $k = 1, \ldots, n$ changes through polar transform and mass merging, a modified doubly linked list data structure is proposed in [14], and computational complexity of the algorithm is found as $O(\mu^2 \log \mu)$. Similar data structures can be utilized to implement the greedy code construction algorithm proposed in this subsection.

Algorithm 4: mass_merging(χ, μ)

input : χ: Source distribution, μ: Maximum allowed cardinality for $\mathcal{Y}$

if $|\mathcal{Y}| < \mu$ **then**
  Exit.
else
  Sort $(p_Y(y_i), \mathbf{p}_{X|y_i})$ such that $\mathcal{H}(\mathbf{p}_{X|y_i}) \leq \mathcal{H}(\mathbf{p}_{X|y_{i+1}})$;
  Compute $\widehat{\epsilon}_{y_i,y_{i+1}}$ for all $i = 0, 1, \ldots, |\mathcal{Y}| - 2$;
  while $|\mathcal{Y}| > \mu$ **do**
    Find $y_i = \underset{y_j : j=0,1,\ldots,|\mathcal{Y}|-2}{\operatorname{argmin}} \widehat{\epsilon}_{y_j,y_{j+1}}$;
    merge(y_i, y_{i+1});
    Update $\widehat{\epsilon}_{y_{i-1},y_i}$, $\widehat{\epsilon}_{y_i,y_{i+1}}$;

Algorithm 5: code_construction(χ, N, μ)

input : $X|Y \sim \chi$: Input source, N: Block-length, μ: Maximum allowed alphabet size for side information

output: $\{H(U_i|U_0^{i-1}, Y_0^{N-1})\}_{i=0}^{N-1}$

for $i = 0, 1, \ldots, N-1$ **do**
  $(b_0 b_1 \ldots b_{n-1})_2 = i$;
  $\chi_i^{(0)} = \chi$;
  for *k=0,1,. . . ,n-1* **do**
    if $b_k = 0$ **then**
      $\chi_i^{(k+1)} = \chi_i^{(k)} \circledast \chi_i^{(k)}$;
    else
      $\chi_i^{(k+1)} = \chi_i^{(k)} \boxtimes \chi_i^{(k)}$;
    mass_merging$(\chi_i^{(k+1)}, \mu)$;
  $H(U_i|U_0^{i-1}, Y_0^{N-1}) = H(\chi_i^{(n)})$;
Return $\{H(U_i|U_0^{i-1}, Y_0^{N-1})\}_{i=0}^{N-1}$.

2.5 Numerical Results

In this section, the performance of the proposed data compression scheme is investigated. The figures of merit in this investigation are the block error rate P_b and symbol error rate P_s for a fixed code rate R. Code constructions in all examples are performed with mass merging algorithm with parameter $\mu = 16$.

In Figure 2.7, the performance of polar codes for compressing a ternary source with probability distribution $p_X = (0.84,\ 0.09,\ 0.07)$ at block-length $N = 2^{10}$ under SC-D and SCL-D with list sizes $L = 2, 4, 8, 32$ is illustrated. It must be noted that SCL-D outperforms SC-D, and increasing L beyond $L = 8$ at large code rates does not make significant improvement in P_b.

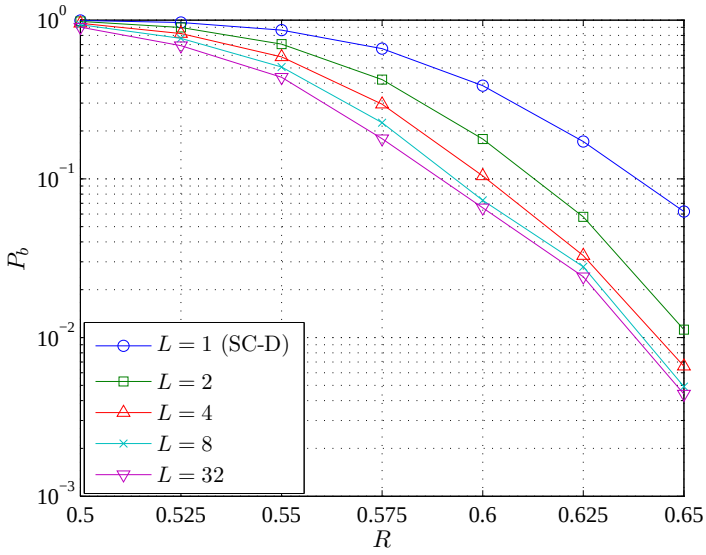

Figure 2.7: Block error rates in the compression of a source with distribution $p_X = (0.84,\ 0.09,\ 0.07)$ at block-length $N = 2^{10}$ under SCL-D with $L = 1, 2, 4, 8, 32$.

For the same source, symbol error rates, P_s, are given in Figure 2.8.

In order to investigate the effect of increasing N on P_b and P_s for a fixed source distribution, performance of lossless polar compression scheme for source distribution $p_X = (0.84,\ 0.09,\ 0.07)$ at block-length $N = 2^{12}$ is given in Figure 2.9. This example indicates that block error rates decrease at rates above the minimum source coding rate as N increases as expected. At a fixed rate $R > H(X) = 0.5$,

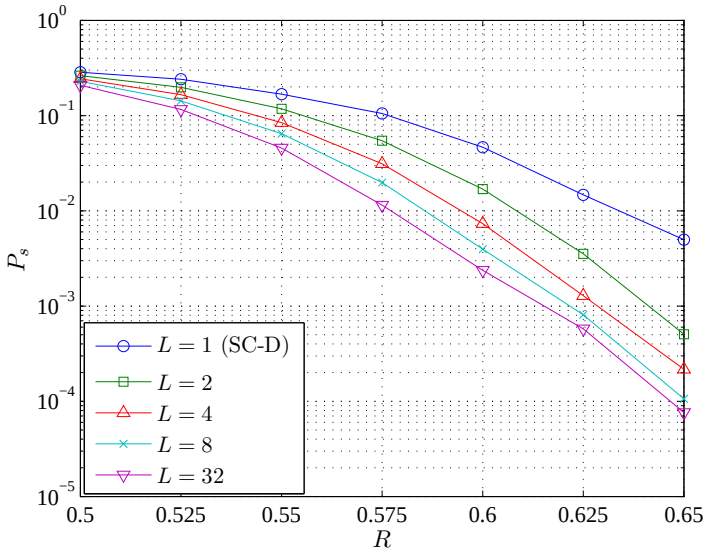

Figure 2.8: Symbol error rates in the compression of a source with distribution $p_X = (0.84,\ 0.09,\ 0.07)$ at block-length $N = 2^{10}$ under SCL-D with $L = 1, 2, 4, 8, 32$.

an arbitrary P_b can be obtained at a sufficiently large N.

For the same source, symbol error rates, P_s, are given in Figure 2.10.

For analyzing the performance of the coding scheme at a larger alphabet size, block error rate performance of a quinary source with distribution $p_X = (0.05,\ 0.05,\ 0.055,\ 0.055,\ 0.79)$ at block-length $N = 2^{10}$ is given in Figure 2.12. For the same quinary source, symbol error rate performance is shown in Figure 2.12. The last example indicates that at a fixed block-length and base-q source entropy, similar performance can be obtained at an increased alphabet size.

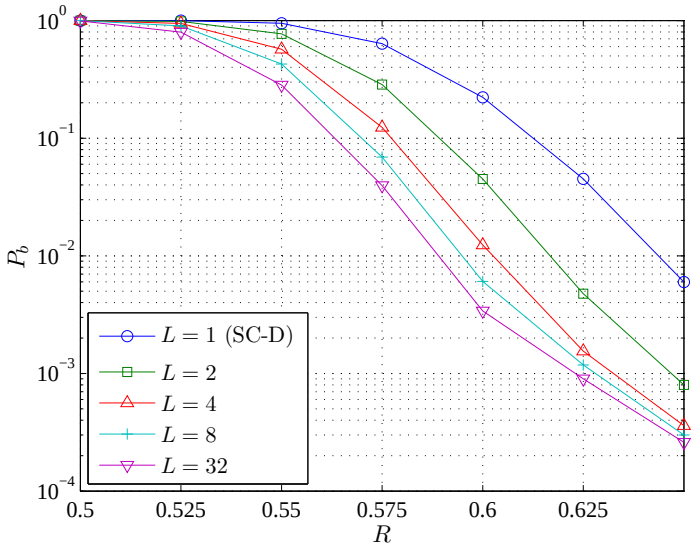

Figure 2.9: Block error rates in the compression of a source with distribution $p_X = (0.84,\ 0.09,\ 0.07)$ at block-length $N = 2^{12}$ under SCL-D with $L = 1, 2, 4, 8, 32$.

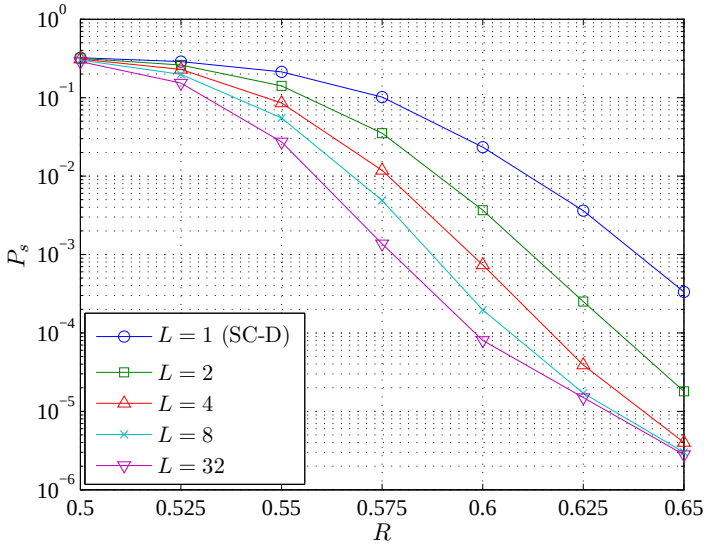

Figure 2.10: Symbol error rates in the compression of a source with distribution $p_X = (0.84,\ 0.09,\ 0.07)$ at block-length $N = 2^{12}$ under SCL-D with $L = 1, 2, 4, 8, 32$.

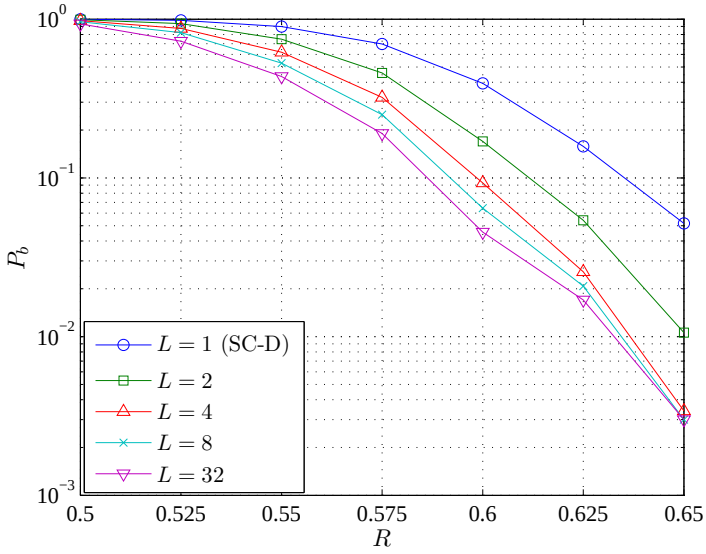

Figure 2.11: Block error rates in the compression of a source with distribution $p_X = (0.05,\ 0.05,\ 0.055,\ 0.055,\ 0.79)$ at block-length $N = 2^{10}$ under SCL-D with $L = 1, 2, 4, 8, 32$.

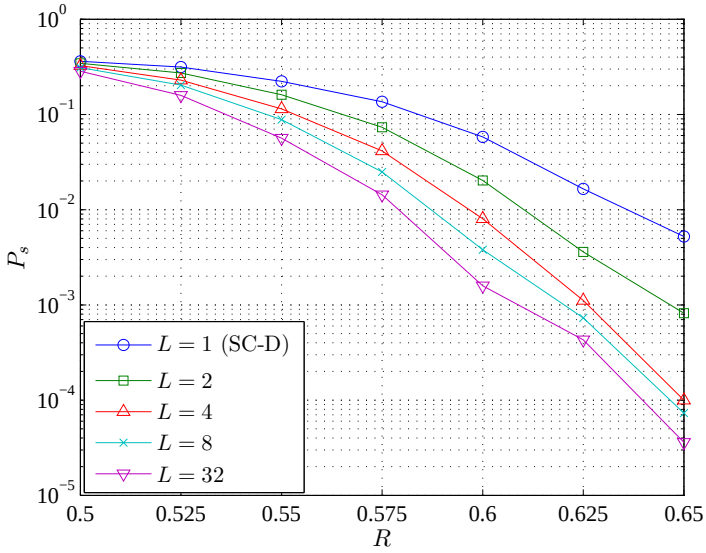

Figure 2.12: Symbol error rates in the compression of a source with distribution $p_X = (0.05,\ 0.05,\ 0.055,\ 0.055,\ 0.79)$ at block-length $N = 2^{12}$ under SCL-D with $L = 1, 2, 4, 8, 32$.

Chapter 3

Oracle-Based Lossless Polar Compression

3.1 Introduction

In Chapter 2, in order to compress a sequence $\{(X_i, Y_i)\}_{i=0}^{N-1}$, an information set $\mathcal{I}_{X|Y}(N, R)$ consisting of indices i that correspond to NR highest $H(U_i|U_0^{i-1}, Y_0^{N-1})$ terms is constructed. Then, a given realization x_0^{N-1} is transformed into u_0^{N-1} by (2.2) and the compressed word $u_{\mathcal{I}_{X|Y}}$ is formed. For sufficiently large N, this scheme is proved to achieve arbitrarily small probability of error under conventional SC-D with codeword length $NH(X|Y)$ [8]. In [9], for binary sources, an oracle-based polar compression method that has an improved performance at finite block-lengths is introduced. Here, a similar approach is taken in the design of fixed-to-variable length, zero-error compression methods for q-ary discrete memoryless sources. The methods are based on appending a block, namely oracle set $\mathcal{T}$, to the compressed word $u_{\mathcal{I}_{X|Y}}$ indicating the locations of the errors that will be encountered in decoding, and correcting them. This block enables zero-error coding at any block-length. Moreover, it is shown that this extra block has a diminishing fraction in the transmitted word, which means that the minimum source coding rate is still achievable asymptotically in

the block-length. The discussion in this chapter is partly presented in [21].

3.2 Preliminaries

For finite-length analysis and code construction, the minimal error probability, analyzed in [22], provides a more convenient measure than conditional entropy [9]. The minimal error probability, denoted by $\pi(X|Y=y)$, is the probability of error in the maximum a posteriori estimation of X given an observation $Y=y$:

$$\begin{aligned}\pi(X|Y=y) &= Pr[X \neq \underset{x\in\mathcal{X}}{\operatorname{argmax}}\ p_{X|Y}(x|y)|Y=y], \\ &= 1 - \max_{x\in\mathcal{X}}\ p_{X|Y}(x|y).\end{aligned}$$

Therefore, the average minimal probability of error is as follows:

$$\pi(X|Y) = \sum_{y\in\mathcal{Y}} p_Y(y)\pi(X|Y=y). \tag{3.1}$$

$\pi(X|Y)$ has a range $[0, \frac{q-1}{q}]$ and is a concave function of $p_{X|Y}(x|y)$.

3.3 Encoding

In noiseless source coding, the encoder has a copy of the codeword received by the decoder. This specific property enables the encoder to run the decoder at the transmitter side and check if a decoding error occurs. In polar compression, this capability can be utilized to prevent any errors by appending a variable length block of error positions and their correct symbols to the codeword; thus fixed-to-variable length, zero-error coding schemes can be designed. The oracle-based lossless polar compression scheme with successive cancellation type decoders is illustrated in Figure 3.1.

The encoding is specific to the type of decoder. Therefore, we will consider schemes with SC-D and SCL-D separately. First, let us consider the encoding in the case of SC-D, which is a straightforward extension of [9].

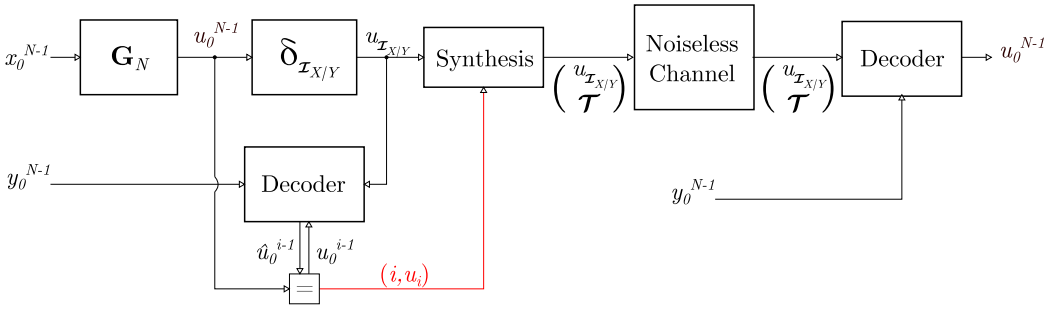

Figure 3.1: Oracle-based lossless polar compression scheme.

3.3.1 Encoding with Successive Cancellation Decoder

For a given source realization x_0^{N-1}, the encoder forms the codeword $u_{\mathcal{I}_{X|Y}}$ and conveys it to the mirror SC-D at the transmitter side. If an error occurs at phase i, the encoder interferes, records the error location (i, u_i), corrects the error and resumes the decoding process. Following this routine, the encoder records the set of all error locations together with their respective correct symbols:

$$\mathcal{T}_{SC} = \{(i, u_i) : u_i \neq \widehat{u}_i | u_0^{i-1}, y_0^{N-1}\}. \tag{3.2}$$

Then, the encoder appends $\mathcal{T}_{SC}$ to the codeword $u_{\mathcal{I}_{X|Y}}$ and transmits $(u_{\mathcal{I}_{X|Y}}, \mathcal{T}_{SC})$ to the receiver side. Having the error locations and their respective correct symbols, the decoder at the receiver side performs decompression with no error. Note that if $q = 2$, there is no need to record the symbol u_i since knowing the location of an error is sufficient to correct it through inversion. In the rest of the discussion, a general q will be considered, and the correct symbol value will be included in $\mathcal{T}_{SC}$. The encoder with SC-D is summarized in Algorithm 6.

Given a correctly decoded subsequence $\widehat{u}_0^{i-1}$ and observation y_0^{N-1}, the probability of error at phase i of SC-D is $\pi(U_i|\widehat{u}_0^{i-1}, y_0^{N-1})$. Thus, the average probability of error at phase i is $\pi(U_i|U_0^{i-1}, Y_0^{N-1})$. If an error occurs at phase i, it costs an additional overhead of $(\log N + 1)$ symbols. Therefore, the average cost of not including i in the information set in terms of extra overhead is $\pi(U_i|U_0^{i-1}, Y_0^{N-1})[\log N + 1]$ symbols. The cost of including i in the information

Algorithm 6: SC_Encoder(u_0^{N-1}, y_0^{N-1})

input : u_0^{N-1}: Output of the polar transform, y_0^{N-1}: Side information
output: $u_{\mathcal{I}_{X|Y}}$: Codeword, $\mathcal{T}_{SC}$: Oracle set

for $i = 0, 1, \ldots, N-1$ **do**
  if $i \in \mathcal{I}_{X|Y}$ **then**
    $\widehat{u}_i = u_i$;
    Record $u_i \rightarrow u_{\mathcal{I}_{X|Y}}$;
  else
    $\widehat{u}_i = \underset{u_i \in \mathcal{X}}{\operatorname{argmax}} \; P_N^{(i)}(u_i | \widehat{u}_0^{i-1}, y_0^{N-1})$;
    if $\widehat{u}_i \neq u_i$ **then**
      Record $(i, u_i) \rightarrow \mathcal{T}_{SC}$;
      Correct the symbol: $\widehat{u}_i = u_i$;
Return $(u_{\mathcal{I}_{X|Y}}, \mathcal{T}_{SC})$.

set is 1 symbol. Combining these results, the expected code rate R is as follows:

$$\mathbb{E}[R] = \frac{1}{N}\Big\{|\mathcal{I}_{X|Y}| + \sum_{i \in \mathcal{I}_{X|Y}^c} \pi(U_i | U_0^{i-1}, Y_0^{N-1}).[\log N + 1]\Big\}. \tag{3.3}$$

This analysis can be used in the construction of $\mathcal{I}_{X|Y}$ as well [9]. The objective is to minimize the expected code rate over all information sets. If the average cost of including an index i in $\mathcal{I}_{X|Y}^c$ is higher than including it in $\mathcal{I}_{X|Y}$, then the symbol is transmitted in $u_{\mathcal{I}_{X|Y}}$. Thus, in this approach, the information set is formed as follows:

$$\mathcal{I}_{X|Y} = \{i : \pi(U_i | U_0^{i-1}, Y_0^{N-1})[\log N + 1] > 1\}. \tag{3.4}$$

For sufficiently large N, $\mathcal{I}_{X|Y}$ consists of indices such that $\pi(U_i | U_0^{i-1}, Y_0^{N-1}) \in (\frac{q-1}{q} - \epsilon, \frac{q-1}{q}]$. By source polarization theorem, the cardinality of $\mathcal{I}_{X|Y}$ approaches $NH(X|Y)$. Therefore, the expected rate goes to the minimum source coding rate as $n \rightarrow \infty$:

$$\mathbb{E}[R] \rightarrow H(X|Y).$$

Hence, this zero-error compression scheme designed for finite block-lengths achieves the theoretical bound asymptotically as well.

The following question arises: How to compute $\{\pi(U_i|U_0^{i-1}, Y_0^{N-1})\}_{i=0}^{N-1}$ to construct $\mathcal{I}_{X|Y}$ using (3.4)? By (3.1), the set of $(q+1)$-dimensional vectors $\{(p_Y(y), [p_{X|Y}(x|y)]_{x=0}^{q-1})\}_{y\in\mathcal{Y}}$ is required for this computation, which implies that $U_i|U_0^{i-1}, Y_0^{N-1}$ is required to compute $\pi(U_i|U_0^{i-1}, Y_0^{N-1})$ similar to the case of conditional entropy. Therefore, a slight modification in the code construction method proposed in Section 2.4 suffices for code construction in this case. One alternative for this modification is to replace $\mathcal{H}$ by π in (2.17) and perform mass merging using average minimal error probability. Since π is a concave function of $\mathbf{p}$, this alternative works. The other alternative is to perform code construction in the same way as Chapter 2 until Line 10 of Algorithm 5 and computing $\pi(U_i|U_0^{i-1}, Y_0^{N-1})$ from $\chi_i^{(n)}$.

3.3.2 Encoding with Successive Cancellation List Decoder

The SC-D flags a block error once an incorrect decision is made and causes additional overhead because of oracle employment. Successive cancellation list decoder is likely to correct an incorrect decision at succeeding phases in the expense of increased complexity. In noiseless source coding, this property of SCL-D can be utilized to reduce the expected codeword length. Consider an SCL-D of list size L at phase $i \notin \mathcal{I}_{X|Y}$. Assume that the correct decoding path $\widehat{u}_0^{i-1} = u_0^{i-1}$ is contained among the active paths. At phase i, all symbols in $\mathcal{X}$ is appended to each active path, and all paths are pruned keeping L of the highest probability values. Denoting the set of all active paths at phase i by $\mathcal{L}_i$, an error is flagged if the correct subsequence u_0^i is not in $\mathcal{L}_i$. If such an event occurs, the encoder interferes, takes a record of (i, u_i) and appends u_i to each active path as if i is contained in the information set. Eventually, the oracle set is formed as follows:

$$\mathcal{T}_{SCL} = \{(i, u_i) : u_0^i \notin \mathcal{L}_i | u_0^{i-1} \in \mathcal{L}_{i-1}, y_0^{N-1}\}. \tag{3.5}$$

The employment of this oracle set guarantees the correct decoding path u_0^{N-1}

to survive until the end. In the last phase, SCL-D returns the sequence among $\mathcal{L}_{N-1}$ with highest probability. An incorrect sequence is returned if there is a path $\tilde{u}_0^{N-1} \in \mathcal{L}_{N-1}$ with higher probability than u_0^{N-1}. In order to prevent this error, the list index l of the correct sequence can be annexed to the codeword. This increases the codeword length by $\log L$ symbols. On the other hand, since the probability of error event, $Pr\{u_0^i \notin \mathcal{L}_i | y_0^{N-1}\}$, is smaller than the probability of error event in SC-D, $Pr\{\widehat{u}_i \neq u_i | \widehat{u}_0^{i-1} = u_0^{i-1}, y_0^{N-1}\}$, and hence the use of the oracle becomes less frequent, the overall overhead decreases compared to the oracle-based compression scheme with SC-D.

3.4 Decoding

In this section, oracle decoders that reconstruct $\widehat{x}_0^{N-1}$ from $u_{\mathcal{I}_{X|Y}}$ and oracle set $\mathcal{T}$ will be proposed. The decoders are basically similar to the ones discussed in Chapter 2 with the difference that oracle sets are also exploited for zero-error reconstruction.

3.4.1 Successive Cancellation Decoder for Oracle-Based Compression

For a given source (X, Y) and observation y_0^{N-1}, the probability of observing u_0^{N-1} at the output of the polarization transform is denoted as $P_N(u_0^{N-1}|y_0^{N-1})$, where $P_1(x|y) = p_X(x|y)$. Similarly, the probability of a subsequence u_0^i is denoted as $P_N^{(i)}(u_i|u_0^{i-1}, y_0^{N-1})$. SC-D algorithm is summarized in Algorithm 7.

SC-D can be implemented with $O(N)$ memory and $O(N \log N)$ run-time complexity [17].

Algorithm 7: SC_Decoder($u_{\mathcal{I}_{X|Y}}, y_0^{N-1}, \mathcal{T}_{SC}$)

input : $u_{\mathcal{I}_{X|Y}}$: Codeword, y_0^{N-1}: Side information, $\mathcal{T}_{SC}$: Oracle set
output: $\widehat{x}_0^{N-1}$: Reconstructed sequence

for $i = 0, 1, \ldots, N-1$ **do**
  if $i \in \mathcal{I}_{X|Y}$ or $(i, u_i) \in \mathcal{T}_{SC}$ **then**
    $\widehat{u}_i = u_i$
  else
    $\widehat{u}_i = \underset{u_i \in \mathcal{X}}{\operatorname{argmax}} \; P_N^{(i)}(u_i | \widehat{u}_0^{i-1}, y_0^{N-1})$;
Return $\widehat{x}_0^{N-1} = \widehat{u}_0^{N-1} \mathbf{G}_N^{-1}$.

3.4.2 Successive Cancellation List Decoder for Oracle-Based Compression

The high-level description of the SCL-D is given in Algorithm 8. SCL-D algorithm has $O(LN \log N)$ run-time complexity [17]. Note that SCL-D with list size $L = 1$ corresponds to the SC-D.

Algorithm 8: SCL_Decoder($u_{\mathcal{I}_{X|Y}}, y_0^{N-1}, \mathcal{T}_{SCL}, l_0, L$)

input : $u_{\mathcal{I}_{X|Y}}$: Codeword, y_0^{N-1}: Side information, $\mathcal{T}_{SCL}$: Oracle set, l_0: Index of the correct decision path, L: List size
output: $\widehat{x}_0^{N-1}$: Reconstructed sequence

for $i = 0, 1, \ldots, N-1$ **do**
  if $i \in \mathcal{I}_{X|Y}$ or $(i, u_i) \in \mathcal{T}_{SCL}$ **then**
    Append u_i to each $l \in \mathcal{L}_{i-1}$, i.e., $\widehat{u}_0^{i-1}[l]$, and obtain $(\widehat{u}_0^{i-1}[l], u_i)$
  else
    Append all $\widehat{u}_i \in \mathcal{X}$ to each $l \in \mathcal{L}_{i-1}$;
    Calculate $P_N(\widehat{u}_i | \widehat{u}_0^{i-1}[l], y_0^{N-1})$ for all $l \in \mathcal{L}_i$;
    Prune all but L paths with highest probabilities.
$\widehat{u}_0^{N-1} = P_N^{(N-1)}(\widehat{u}_{N-1}[l_0] | \widehat{u}_0^{N-2}[l_0], y_0^{N-1})$
Return $\widehat{x}_0^{N-1} = \widehat{u}_0^{N-1} \mathbf{G}_N^{-1}$.

3.5 Compression of Sources over Arbitrary Finite Alphabets

In this section, we generalize the oracle-based compression schemes to sources over any arbitrary finite alphabets. In order to realize this, we first consider a specific configuration for the noiseless compression of two correlated sources (X, Y). In this scenario, the source output Y_0^{N-1} is available to the X-encoder, the decompressed word $\widehat{Y}_0^{N-1}$ is available to the X-decoder, and neither X_0^{N-1} nor $\widehat{X}_0^{N-1}$ is used in the compression of Y. The scheme is illustrated in Figure 3.2.

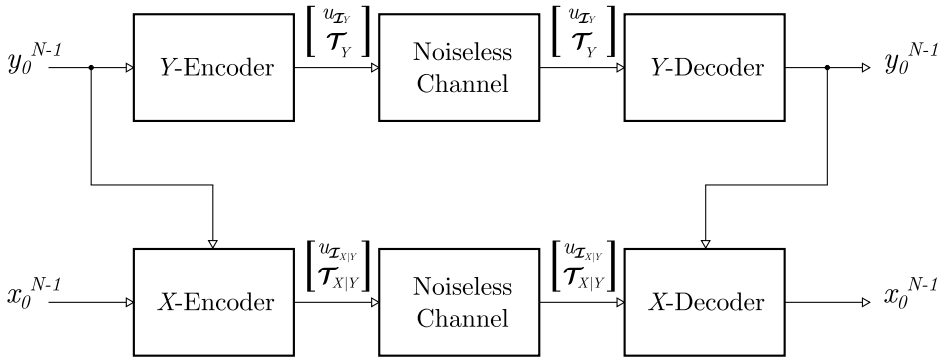

Figure 3.2: (0101)-configuration for the compression of (X, Y).

This configuration is analyzed in [2], where it is called (0101)-configuration. It is possible to achieve a compression rate $H(X, Y)$ for (X, Y) at rates $R_Y = H(Y) + \epsilon$ and $R_X = H(X|Y) + \epsilon$ for Y and X, respectively, which is referred to as the corner point of the admissible region.

Lemma 2. *The oracle-based polar compression scheme achieves the corner point of the admissible region for (0101)-configuration.*

Proof. In order to compress Y, the oracle-based scheme is used with no side information. The compression rate R_Y asymptotically approaches to $H(Y)$. Since

this is a zero-error coding scheme, the Y-source output is reconstructed faithfully at the receiver side. In order to compress X, the oracle-based compression scheme is used with the side information Y. Note that Y-source output is available at both transmitter and receiver sides with no error. Thus, X can be compressed at rate $R_X = H(X|Y)$ with the oracle described in (3.2), and the corner point of (0101)-configuration is achieved asymptotically. □

An extension of this configuration is the noiseless source coding over arbitrary finite alphabets, using a similar approach as in [16]. Let Z be a random variable over a finite alphabet $\mathcal{Z}$. Z can be decomposed into K symbols using the Chinese remainder theorem as:

$$Z = (Z^{K-1}, Z^{K-2}, \ldots, Z^0),$$

where Z^k is over $\mathcal{Z}_k$, provided that $|\mathcal{Z}_k| = q_k$ and all q_k are pairwise coprime. Note that q_k can be an integer power of a prime, in which a further expansion can be carried out to obtain prime alphabet sizes for compression, and the result can be used to uniquely reconstruct Z. Hence, without loss of generality, it can be assumed in further discussions that all q_k are prime.

At the first step of compressing Z, Z^0 is compressed with no side information, analogous to Y in the previous case, at rate approximately equal to $H(Z^0)$. Then, Z^1 is compressed with side information Z^0 at rate approximately equal to $H(Z^1|Z^0)$. Now that the source outputs of (Z^1, Z^0) are transmitted, they are utilized as side information and the compression of Z^2 is performed at rate approximately equal to $H(Z^2|Z^1, Z^0)$. Following this routine, Z^k can be compressed at rate $H(Z^k|Z^{k-1}, \ldots, Z^0)$ for any $k = 0, 1, \ldots, K-1$. After the decompression of Z^{K-1}, Z can be reconstructed faithfully. The total compression in this scheme has the following asymptotical rate:

$$\begin{aligned} R_Z &= \sum_{k=0}^{K-1} R_{Z_k} \\ &\to \sum_{k=0}^{K-1} H(Z^k|Z^{k-1}, \ldots, Z^0) \\ &= H(Z^{K-1}, Z^{K-2}, \ldots, Z^0) = H(Z), \end{aligned}$$

which shows that the entropy bound can be achieved asymptotically by the proposed q-ary polar compression scheme.

In the general case, assume that the alphabet size is $q = \prod_{k=0}^{K-1} q_k^{t_k}$ for pairwise coprimes q_k and positive integers t_k for all $k = 0, 1, \ldots, K-1$. In this case, for a list size L and block-length N, the complexity of the multi-level compression scheme is $O(\sum_{k=0}^{K-1} t_k q_k LN \log N)$. Therefore, it is possible to perform data compression for large source alphabets at low complexity by the multi-level scheme.

3.6 Source Distribution Uncertainty at the Receiver

In the previous sections, it was assumed that the exact probability distribution of the source, denoted as $\mathbf{p}_X$ in the q-dimensional vector form, is available at the receiver, and the information set, denoted as $\mathcal{I}_X(\mathbf{p}_X, N)$, is constructed specifically for $\mathbf{p}_X$. In practice, however, these assumptions are unrealistic since the receiver does not have $\mathbf{p}_X$ unless it is informed by the transmitter side. Moreover, even if the exact knowledge of $\mathbf{p}_X$ is available at the transmitter, it is infeasible to construct $\mathcal{I}_X$ specifically for every given $\mathbf{p}_X$. In this section, we propose methods to address these issues by exploiting robustness of the oracle-based polar compression scheme with respect to the inaccuracies in the source distribution and information set. We consider only data compression in the absence of side information in this section, and the alphabet size is q for a prime integer q. We note that it is straightforward to extend the presented results to the case with side information and non-prime alphabet sizes.

Throughout the section, following the notation in [15], $M(q)$ denotes the set of all probability distributions on a q-ary alphabet:

$$M(q) = \{\mathbf{p} \in \mathbb{R}^q : p_i > 0 \text{ for all } i \in \{0, 1, \ldots, q-1\}, \sum_{i=0}^{q-1} p_i = 1\}.$$

For given $\mathbf{p}_X, \mathbf{p}_Y \in M(q)$, if there exists $\mathbf{p}_Z \in M(q)$ such that $\mathbf{p}_Y = \mathbf{p}_X * \mathbf{p}_Z$ where $*$ denotes circular convolution [23], $\mathbf{p}_Y$ is said to be circularly dominated by $\mathbf{p}_X$, and this relation is represented as $\mathbf{p}_Y \prec_c \mathbf{p}_X$.

First, we propose a method to inform the receiver about the probability distribution of the source $\mathbf{p}_X$ efficiently in the oracle-based framework. In order to inform the receiver side about $\mathbf{p}_X$, our approach is to quantize $q-1$ elements of $\mathbf{p}_X$ by using β symbols, and append the quantized probability distribution $\widehat{\mathbf{p}}_X$ to the compressed word.

3.6.1 Sequential Quantization with Scaling Algorithm for Probability Mass Functions

For the quantization of a probility distribution $\mathbf{p}_X$, sequential quantization with scaling provides an efficient solution. In this quantization approach, the original probability distribution is sorted in the descending order, i.e., a permutation transformation $\phi : M(q) \mapsto M(q)$ must be applied such that $\mathbf{p} = \phi(\mathbf{p}_X)$ where $p_0 \geq p_1 \geq \ldots \geq p_{q-1}$ holds between the elements of $\mathbf{p}$. Note that the permutation ϕ can be transmitted to the receiver by $q-1$ symbols, which implies that the original ordering of the elements of $\mathbf{p}_X$ can be restored at the receiver with no loss of fidelity.

For a probability distribution function $\mathbf{p}$, the sorted order implies the following bounds for the value of p_i:

$$p_i \in [\frac{1-\sum_{k=0}^{i-1} p_k}{q-i}, 1-\sum_{k=0}^{i-1} p_k]. \tag{3.6}$$

The sequential quantization with scaling algorithm starts with the quantization of p_0, and then the bounds in (3.6) are found for each p_i, $i = 0, 1, \ldots, q-1$ by using the previously quantized elements $\widehat{p}_0^{i-1}$, and the interval in (3.6) is divided into q^β uniform levels. The level $d_i \in \{0, 1, \ldots, q^\beta - 1\}$ that provides the best approximation to p_i is chosen for the compressed word in its q-ary expansion.

The sequential quantization with scaling algorithm is summarized in Algorithm 9.

Algorithm 9: pmf_quantizer($\mathbf{p}_X, \beta$)

input : $\mathbf{p}_X$: Source distribution, β: Symbols per pmf element
output: $(d_0, d_1, \ldots, d_{q-2})$: Quantization levels, ϕ: Sorting permutation

1 Sort $\mathbf{p}_X$ and obtain $\mathbf{p}$: $\mathbf{p} = \phi(\mathbf{p}_X)$;
2 Compute $d_0 = \underset{d=0,1,\ldots,q^\beta-1}{\operatorname{argmin}} |\frac{1}{q}[1 + d.\frac{q-1}{q^\beta-1}] - p_0|$;
3 Compute $\widehat{p}_0 = \frac{1}{q}[1 + d_0.\frac{q-1}{q^\beta-1}]$;
4 **for** $i = 1, \ldots, q-2$ **do**
5 Compute $\widehat{p}_{i,max} = 1 - \sum_{k=0}^{i-1} \widehat{p}_k$;
6 Compute $d_i = \underset{d=0,1,\ldots,q^\beta-1}{\operatorname{argmin}} |\frac{\widehat{p}_{i,max}}{q-i}[1 + d.\frac{q-i-1}{q^\beta-1}] - p_i|$;
7 Compute $\widehat{p}_i = \frac{\widehat{p}_{i,max}}{q-i}[1 + d_i.\frac{q-i-1}{q^\beta-1}]$;
8 Return (d_0^{q-2}, ϕ).

The receiver can reconstruct the first $q-1$ elements of $\widehat{\mathbf{p}}$ successively from d_0^{q-2} using Line 7 of Algorithm 9. The complete pmf in the descending order is obtained by using the relation $\widehat{p}_{q-1} = 1 - \sum_{i=0}^{q-2} \widehat{p}_i$. Using the permutation ϕ, the quantized version $\widehat{\mathbf{p}}_X(d_0^{q-2}, \phi)$ of $\mathbf{p}_X$ is obtained.

The overall increase in the codeword length by transmitting the quantized probability distribution in the form of (d_0^{q-2}, ϕ) is $(q-1)(\beta+1)$ symbols if β symbols are used to represent each d_i.

To sustain the zero-error compression property of the oracle-based scheme, the mirror decoder in the oracle encoder must use the quantized probability distribution $\widehat{\mathbf{p}}_X(d_0^{q-2}, \phi)$ to foresee errors and include them in the oracle set. Also, the information set $\mathcal{I}_X$ should be constructed for the quantized pmf both at the transmitter and receiver.

3.6.2 Information Sets under Source Uncertainty and the Concept of Class of Information Sets

The computation of $\{\pi(U_i|U_0^{i-1})\}_{i=0}^{N-1}$ using density evolution algorithm with the q-ary extension of the approximation methods proposed in [14] has run-time complexity of $O(N)$. However, for accurate approximations, the approximation parameter of the greedy algorithm μ must be chosen sufficiently large as the alphabet size q increases. Considering that the algorithmic complexity of the code construction algorithm $O(\mu^2 \log \mu)$, constructing the information set $\mathcal{I}_X$ by using density evolution technique for every given source is infeasible in practice. To overcome this problem, we propose using a class of pre-built information sets. In the following, the concept of class of information sets (CIS) will be defined first, and then a CIS construction technique for the source polarization will be proposed by investigating robustness of the compression scheme with respect to the perturbations in the information set.

Definition 1. *For a given block-length $N = 2^n$, a positive integer C, and a set of probability distributions $\boldsymbol{p}_i \in M(q)$ for $i = 0, 1, \ldots, C-1$, the class of information set (CIS)*

$$\mathcal{C} = \{\mathcal{I}_X(\boldsymbol{p}_i, N),\ i = 0, 1, \ldots, C-1\}$$

is a pre-built class of index sets which is known at both the transmitter and the receiver, together with the corresponding probability distributions $\boldsymbol{p}_i$.

For a given realization x_0^{N-1} drawn from a probability distribution $\mathbf{p}_X$ whose β-symbol quantized version is $\widehat{\mathbf{p}}_X$, the CIS $\mathcal{C}$ is used as follows:

1. The transmitter finds $\mathbf{p}_i$ such that using $\mathcal{I}_X(\mathbf{p}_i, N) \in \mathcal{C}$ minimizes the code rate, and uses this information set to form the codeword $(u_{\mathcal{I}_X}, \mathcal{T}, d_0^{q-2}, \phi)$.

2. After the receiver reconstructs $\widehat{\mathbf{p}}_X$, it finds $\mathbf{p}_i$ in the same way as the transmitter, and uses the corresponding information set in the recontruction of x_0^{N-1}.

In order to measure the robustness of the lossless polar compression scheme with respect to inaccuracies in the information set, we consider the average increase in the length of the compressed word if an information set constructed for a different probability distribution than the source distribution is used. Let $\mathbf{p}_X \in M(q)$ be the probability distribution of the source X_0^{N-1} and $\mathbf{p}$ be an arbitrary distribution over $M(q)$. The average cost of using $\mathcal{I}_X(\mathbf{p}, N)$ in the compression of X_0^{N-1} is defined as follows:

$$J_N(\mathbf{p}_X, \mathbf{p}) = \mathbb{E}[R(\mathcal{I}_X(\mathbf{p}, N))|\mathbf{p}_X] - \mathbb{E}[R(\mathcal{I}_X(\mathbf{p}_X, N))|\mathbf{p}_X], \tag{3.7}$$

where $\mathbb{E}[R(\mathcal{I}_X)|\mathbf{p}_X]$ is the average code rate if the source distribution is $\mathbf{p}_X$ and information set $\mathcal{I}_X$ is used in the compression.

Proposition 5. *For a given source $X_0^{N-1} \sim \boldsymbol{p}_X \in M(q)$ and $\boldsymbol{p} \in M(q)$, the cost function in (3.7) can be computed as follows:*

$$J_N(\boldsymbol{p}_X, \boldsymbol{p}) = \sum_{i=0}^{N-1} \frac{\pi(U_i|U_0^{i-1})[\log N + 1] - 1}{N} . \omega_i, \tag{3.8}$$

where $\{\pi(U_i|U_0^{i-1})\}_{i=0}^{N-1}\}$ is the set of average minimal error probabilities for X_0^{N-1}, and

$$\omega_i = \begin{cases} 1 & i \in \mathcal{I}_X(\boldsymbol{p}_X, N) \cap \mathcal{I}_X(\boldsymbol{p}, N)^c, \\ -1 & i \in \mathcal{I}_X(\boldsymbol{p}_X, N)^c \cap \mathcal{I}_X(\boldsymbol{p}, N), \\ 0 & \textit{otherwise.} \end{cases}$$

The proof simply follows from (3.3). Since only $\mathcal{I}_X(\widehat{\mathbf{p}}_X, N)$ can be used at the receiver side, replacement of $J_N(\mathbf{p}_X, \mathbf{p})$ by $J_N(\widehat{\mathbf{p}}_X, \mathbf{p})$ provides a more appropriate analysis given that β is large enough for accurate β-symbol quantization.

3.6.2.1 A CIS Construction Scheme

Proposition 5 provides a useful tool for investigating robustness of polar compression scheme and constructing CIS. Spike probability distributions defined in [15] are of particular interest for constructing CIS because of the design simplicity they provide and numerical stability considerations. For constructing information

sets to build a CIS, we propose using spike probability distributions. Note that in code construction, the order of the elements of the probability distribution is not important. Therefore, we consider the spike probability distributions in the following form:

$$\boldsymbol{\varphi}_q(p) = (1-p, \underbrace{\frac{p}{q-1}, \ldots, \frac{p}{q-1}}_{q-1}), \tag{3.9}$$

for $p \in [0, 1-1/q]$ and prime q. The set of all spike distributions is denoted as $\widehat{M}(q)$:

$$\widehat{M}(q) = \{\boldsymbol{\varphi}_q(p) : p \in [0, 1-1/q]\}. \tag{3.10}$$

To investigate the robustness of the polar compression scheme with respect to information sets constructed for spike measures, the cost function $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_i))$ can be evaluated for given distributions $\mathbf{p}_X$ and various $p_i \in [0, 1-1/q]$, $i = 0, 1, \ldots, C-1$. For a given $\mathbf{p}_X \in M(q)$, we are interested in finding the minimum value of $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_i))$ together with the index i such that $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_i))$ is minimized. As an example, consider $\mathbf{p}_X = (0.75,\ 0.21,\ 0.04)$, and $\mathcal{C}$ constructed such that $\mathcal{H}(\boldsymbol{\varphi}_q(p_{i+1})) - \mathcal{H}(\boldsymbol{\varphi}_q(p_i)) = 0.02$ for all i. In Figure 3.3, $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_i))$ is plotted. Note that $\mathcal{H}(\mathbf{p}_X) = 0.61191$, and the cost function attains its minimum value if p_i is in the vicinity of points such that $|\mathcal{H}(\boldsymbol{\varphi}_q(p_i)) - \mathcal{H}(\widehat{\mathbf{p}}_X)|$ is minimum, and the cost function attains a considerably small value for that distribution.

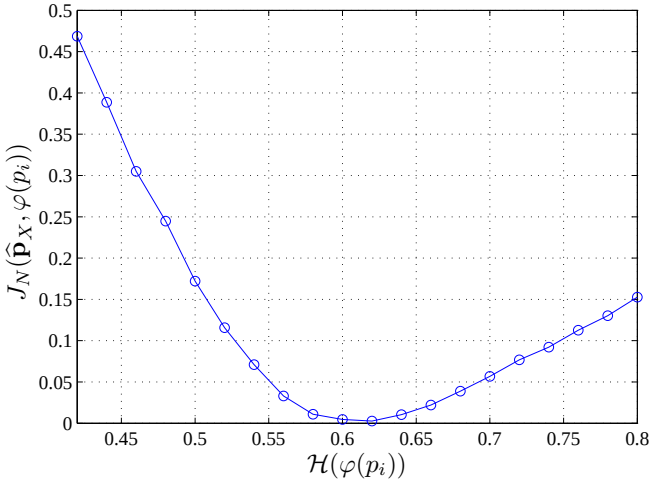

Figure 3.3: The cost function $J_N(\widehat{\mathbf{p}}_X, \boldsymbol{\varphi}_q(p_i))$ with respect to $\mathcal{H}(\boldsymbol{\varphi}_q(p_i))$ for $\mathbf{p}_X = (0.75,\ 0.21,\ 0.04)$, $\beta = 6$, and $N = 2^{10}$.

As a framework for investigating the proposed coding scheme under source uncertainty, the following approach is adopted: For a given q, source distribution $\mathbf{p}_X$ is modelled as a random probability mass function drawn from the Dirichlet distribution with parameter $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_{q-1}),\ \alpha_i > 0$ for all $i = 0, 1, \ldots, q-1$: $\mathbf{p}_X \sim Dir(\boldsymbol{\alpha})$ [24]. Using the methodology in [24], the probability distribution of $\mathbf{p}_X \sim Dir((3,4,5))$ is shown in Figure 3.4. If $\mathbf{p}_X \sim Dir((1,1,\ldots,1))$, the random pmf is distributed uniformly over $M(q)$, which is considered primarily for the analysis in this thesis. There exist efficient methods to generate uniformly

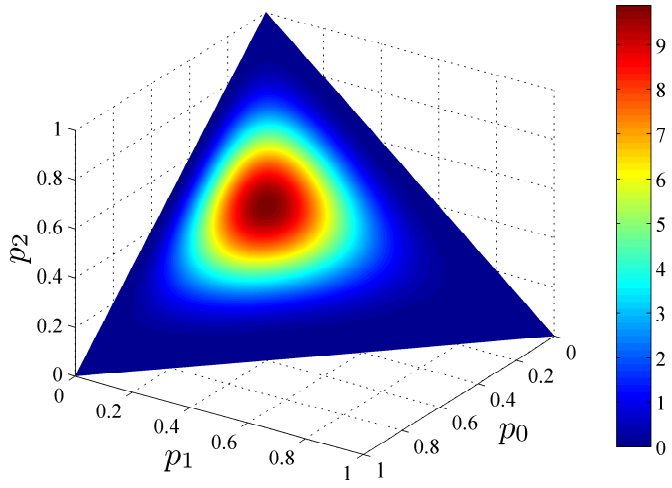

Figure 3.4: Probability distribution of $\mathbf{p}_X \sim Dir((3,4,5))$.

distributed vectors over a unit simplex, i.e., $\mathbf{p}_X \sim Dir((1,1,\ldots,1))$ [25], [26], [27].

In the investigation, independent and identically distributed probability mass functions are drawn: $\mathbf{p}_X \overset{iid}{\sim} Dir(\boldsymbol{\alpha})$. Given a realization $\mathbf{p}_X$, $J_N(\mathbf{p}_X, \boldsymbol{\varphi}(p_i))$ is computed for each $i = 0, 1, \ldots, C-1$. Through this, the robustness of the coding scheme can be investigated in terms of cost function J, and a method for choosing the most appropriate $\boldsymbol{\varphi}_q(p_i)$ for a given pmf realization $\mathbf{p}_X$ can be obtained. The index i such that $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_i))$ is minimized is denoted as i^* throughout the thesis:

$$i^* = \underset{i=0,1,\ldots,C-1}{\operatorname{argmin}} \ J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_i)). \tag{3.11}$$

If there is no prior information about the probability distribution of the source for constructing CIS, then it can be assumed that $\mathbf{p}_X$ is distributed uniformly over $M(q)$. 1000 independent drawings from $\mathbf{p}_X \overset{iid}{\sim} Dir((1,1,1))$ are generated. A CIS of size $C = 32$ is constructed such that $\mathcal{H}(\boldsymbol{\varphi}_q(p_0)) = 1/C$, $p_{C-1} = 1 - 1/q$ and $\mathcal{H}(\boldsymbol{\varphi}_q(p_{i+1})) - \mathcal{H}(\boldsymbol{\varphi}_q(p_i))$ constant for all i. The set $\{p_i\}_{i=0}^{C-1}$ is shown in Figure 3.5 together with the entropy of probability distributions over $M(3)$.

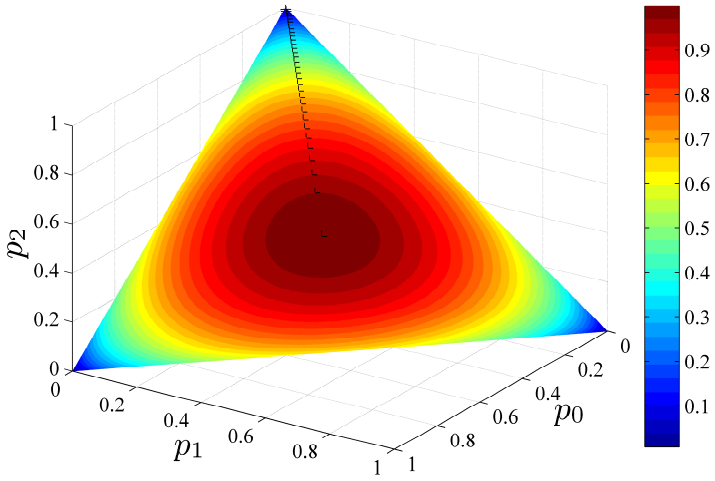

Figure 3.5: Construction of $\mathcal{C}$ with $C = 32$ over $M(3)$. p_i is spotted for all $i = 0, 1, \ldots, C-1$, and entropy is shown as a density plot.

Using this $\mathcal{C}$, for each realization of $\mathbf{p}_X$, $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_{i^*}))$ is computed. In order to find a rule to determine i^* for a given $\mathbf{p}_X$, $\mathcal{H}(\mathbf{p}_X) - \mathcal{H}(\boldsymbol{\varphi}_q(p_{i^*}))$ is also computed. For $N = 2^{10}$, the probability distributions are given in Figure 3.6, and corresponding minimum costs together with $\mathcal{H}(\mathbf{p}_X) - \mathcal{H}(\boldsymbol{\varphi}_q(p_{i^*}))$ are given in Figure 3.7. The probability distributions such that $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_{i^*})) \geq 0.025$ are marked with green circles in both Figure 3.6 and Figure 3.7.

In this investigation, it is observed that the cost can be reduced to acceptable values if $|\mathcal{H}(\widehat{\mathbf{p}}_X) - \mathcal{H}(\boldsymbol{\varphi}_q(p_i))|$ is in the vicinity of its minimum value. Therefore, as a straightforward solution to the problem of determining i^*, minimizer of $|\mathcal{H}(\widehat{\mathbf{p}}_X) - \mathcal{H}(\boldsymbol{\varphi}_q(p_i))|$ among all i can be proposed. For improved performance, machine learning tools can be employed in this problem. The probability distributions marked with green circles correspond to regions that circularly dominate

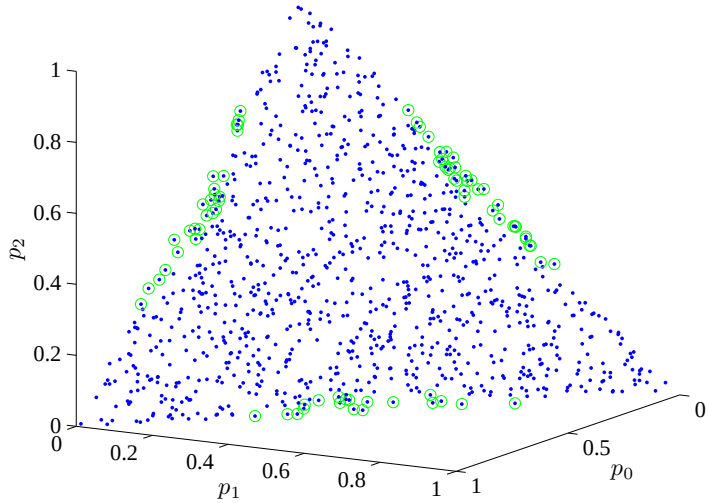

Figure 3.6: Independent and identically distributed realizations from $\mathbf{p}_X \sim Dir((1,1,1))$ for cost analysis at block-length $N = 2^{10}$.

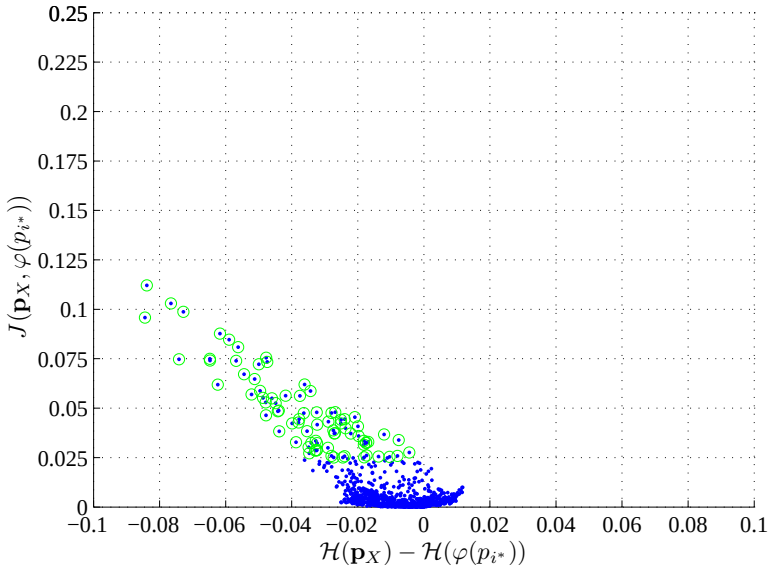

Figure 3.7: $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_{i^*}))$ and $\mathcal{H}(\mathbf{p}_X) - \mathcal{H}(\boldsymbol{\varphi}_q(p_{i^*}))$ values at block-length $N = 2^{10}$ for uniformly distributed $\mathbf{p}_X$.

spike measures with only high entropy values. Non-spike probability measures can be employed in the construction of CIS to reduce the cost J in those regions.

The procedure in the first experiment with 1000 randomly chosen probability distributions is performed for $N = 2^{12}$ and $\mathcal{C}$ of size $C = 64$. In Figure 3.8 and Figure 3.9, the probability mass functions and corresponding minimum cost functions are given. The results indicate that the proposed CIS construction scheme provides similar performance at $N = 2^{12}$.

3.6.2.2 An Alternative CIS Construction Scheme

In [15], robustness of polar source codes (without oracles) with respect to the uncertainty in the probability distribution of the source is investigated. An alternative CIS construction scheme based on this work can be proposed. For $D \subset M(q)$, let

$$\widehat{\mathbf{p}}(D) = \underset{\widehat{\mathbf{p}} \in \widehat{M}(q): \widehat{\mathbf{p}} \prec_c D}{\operatorname{argmin}} H(\widehat{\mathbf{p}}),$$

where $\widehat{\mathbf{p}} \prec_c D$ implies that $\widehat{\mathbf{p}} \prec_c \mathbf{p}_X$ for all $\mathbf{p}_X \in D$. For any $\mathbf{p}_X \in D$, it is shown in [15] that $\mathcal{I}_X(\mathbf{p}_X, N) \subset \mathcal{I}_X(\widehat{\mathbf{p}}(D), N)$. Therefore, it is possible to achieve code rate $H(\widehat{\mathbf{p}}(D))$ for any $\mathbf{p}_X \in D$ asymptotically in the block-length by using the information set $\mathcal{I}_X(\widehat{\mathbf{p}}(D), N)$. Therefore, a method for constructing CIS is to partition probability simplex into C disjoint and exhaustive regions, denoted by $\{D_i\}_{i=0}^{C-1}$, depending on source uncertainty, and to choose the spike measure $\widehat{\mathbf{p}}(D_i)$ for code construction that will be used in the compression of probability measures in D_i for all $i = 0, 1, \ldots, C-1$. In this scheme, if $\mathbf{p}_X \in D_i$, the minimum achievable rate is $H(\widehat{\mathbf{p}}(D_i))$, which implies that $J_N(\mathbf{p}_X, \widehat{\mathbf{p}}(D_i)) \geq H(\widehat{\mathbf{p}}(D_i)) - \mathbb{E}[R(\mathcal{I}_X(\mathbf{p}_X, N))]$. For $\mathbf{p}_X \in M(q)$, let

$$\widehat{\mathbf{p}}(\mathbf{p}_X) = \underset{\widehat{\mathbf{p}} \in \widehat{M}(q): \widehat{\mathbf{p}} \prec_c \mathbf{p}_X}{\operatorname{argmin}} H(\widehat{\mathbf{p}}). \tag{3.12}$$

For any partitioning $\{D_i\}_{i=0}^{C-1}$ and for any $\mathbf{p}_X \in D_i$, the following inequality holds:

$$\mathcal{H}(\widehat{\mathbf{p}}(\mathbf{p}_X)) \leq \mathcal{H}(\widehat{\mathbf{p}}(D_i)). \tag{3.13}$$

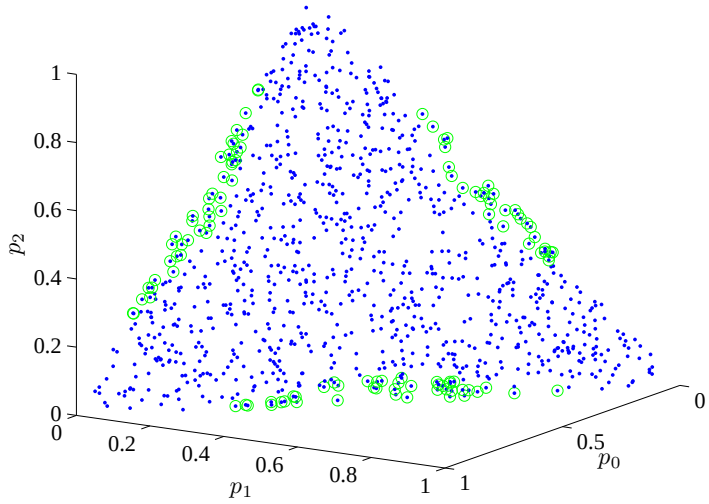

Figure 3.8: Independent and identically distributed realizations from $\mathbf{p}_X \sim Dir((1,1,1))$ for cost analysis at block-length $N = 2^{12}$.

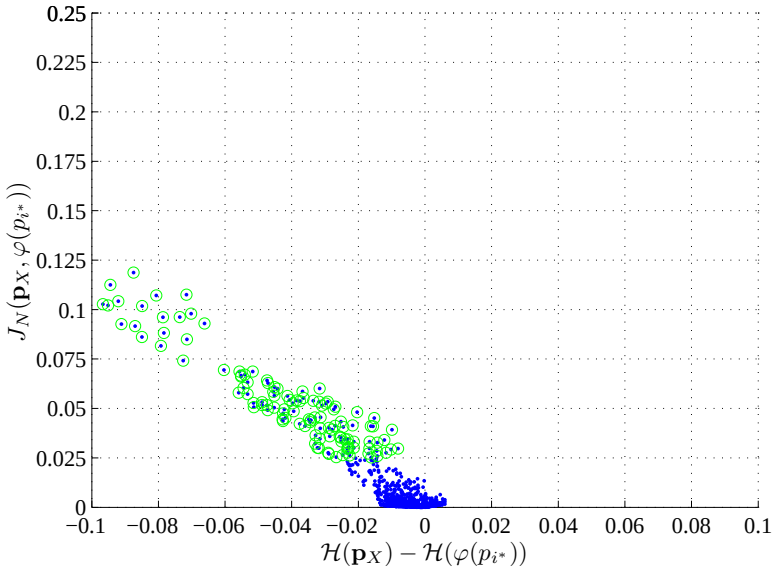

Figure 3.9: $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_{i^*}))$ and $\mathcal{H}(\mathbf{p}_X) - \mathcal{H}(\boldsymbol{\varphi}_q(p_{i^*}))$ values at block-length $N = 2^{12}$ for uniformly distributed $\mathbf{p}_X$.

Let $\tilde{J}_N(\mathbf{p}_X, \widehat{\mathbf{p}}(\mathbf{p}_X)) \triangleq \mathcal{H}(\widehat{\mathbf{p}}(\mathbf{p}_X)) - \mathbb{E}[R(\mathcal{I}_X(\mathbf{p}_X, N))|\mathbf{p}_X]$. Then, the following inequalities are obtained:

$$\begin{aligned} \tilde{J}_N(\mathbf{p}_X, \widehat{\mathbf{p}}(\mathbf{p}_X)) &\leq J_N(\mathbf{p}_X, \widehat{\mathbf{p}}(\mathbf{p}_X)) \\ &\leq J_N(\mathbf{p}_X, \widehat{\mathbf{p}}(D_i)). \end{aligned} \tag{3.14}$$

In order to compare CIS construction schemes, we consider the difference between $\tilde{J}_N(\mathbf{p}_X, \widehat{\mathbf{p}}(\mathbf{p}_X))$ and $J_N(\mathbf{p}_X, \boldsymbol{\varphi}_q(p_{i^*}))$. For the randomly generated probability distributions in Figure 3.6, the performance comparison of the two proposed schemes is given in Figure 3.10. Note that the lower bound on the cost $J_N(\mathbf{p}_X, \widehat{\mathbf{p}}(D_i))$ is

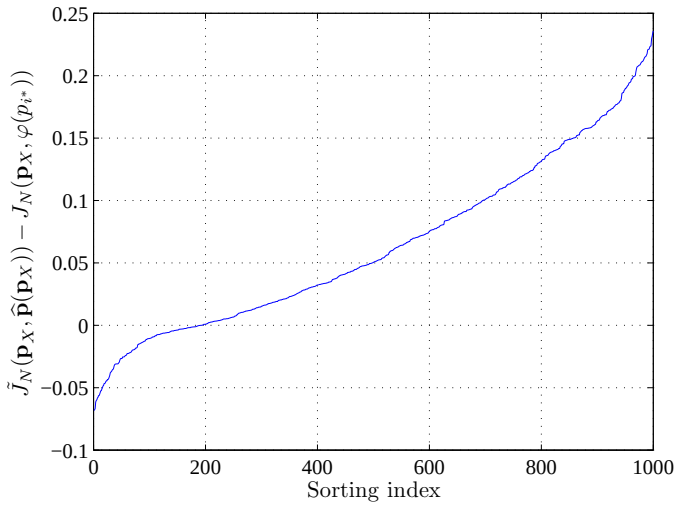

Figure 3.10: Performance comparison of CIS construction techniques for ternary sources of length $N = 2^{10}$.

used in the analysis. From this investigation, it can be concluded that at practical block-lengths, the performance of the alternative CIS construction scheme is degraded with respect to the first scheme even if its performance is maximized.

This investigation indicates that it is possible to build a class of information sets $\mathcal{C}$ by using spike probability distributions which results acceptable increase in the code rate. Since the cost increases as the distance $|H(\mathbf{p}_X) - H(\boldsymbol{\varphi}_q(p))|$ becomes larger, it is appropriate to use an information set constructed for $\boldsymbol{\varphi}_q(p_i)$ whose entropy is close to that of the source distribution. Prior information on the source probability distribution can be modelled as a probability distribution over

$M(q)$. Given this probabilistic model for the source distribution, the optimum $\mathcal{C}$ that minimizes $\mathbb{E}_{\mathbf{p}_X}[\mathbb{E}_{X_0^{N-1}}[R(\mathcal{I}_X(\boldsymbol{\varphi}_q(p_{i^*}), N))|\mathbf{p}_X]]$ can be constructed where the first expectation is over the source distributions, and the latter is over the source vectors drawn from a sample of these distributions.

3.7 Numerical Results

In the first part of performance evaluations, the expected code rates are computed via Monte Carlo simulations with 10000 trials. The code construction is performed with the method described in Algorithm 5 with $\mu = 20$.

In Figure 3.11, the average compression rates for ternary sources with probability mass functions $p_1 = (0.1,\ 0.275,\ 0.625)$, $p_2 = (0.07,\ 0.09,\ 0.84)$ and $p_3 = (0.9214,\ 0.0393,\ 0.0393)$ under SC-D are presented at block-lengths $N = 2^8, 2^9, \ldots, 2^{15}$. Base-3 entropy values are marked by lines. The coding scheme provides rates close to the minimum source coding rate at practical block-lengths, and the backoff from the minimum source coding rate diminishes as N increases.

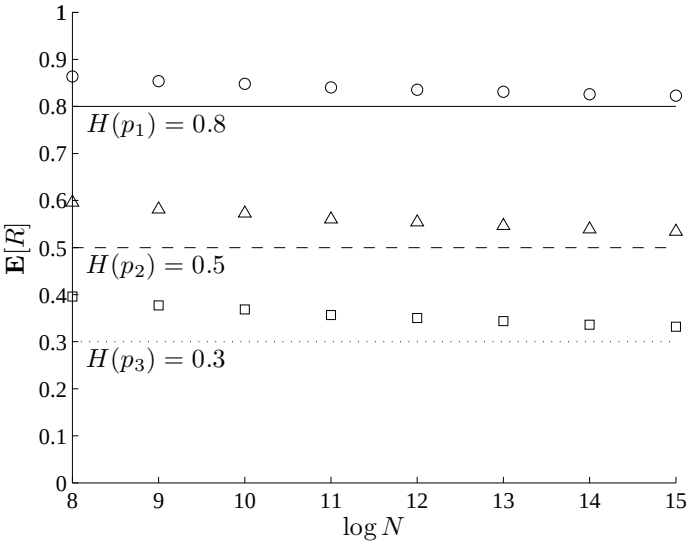

Figure 3.11: Average compression rates for ternary sources under SC-D at block-lengths $N = 2^8, 2^9, \ldots, 2^{15}$.

In Figure 3.12, the improvement in the code rate by using SCL-D is investigated for a source with distribution $p_2 = (0.07,\ 0.09,\ 0.84)$ at various block-lengths. It is observed that SCL-D provides significant improvement at practical block-lengths, and also the improvement due to increasing L becomes less significant as N increases.

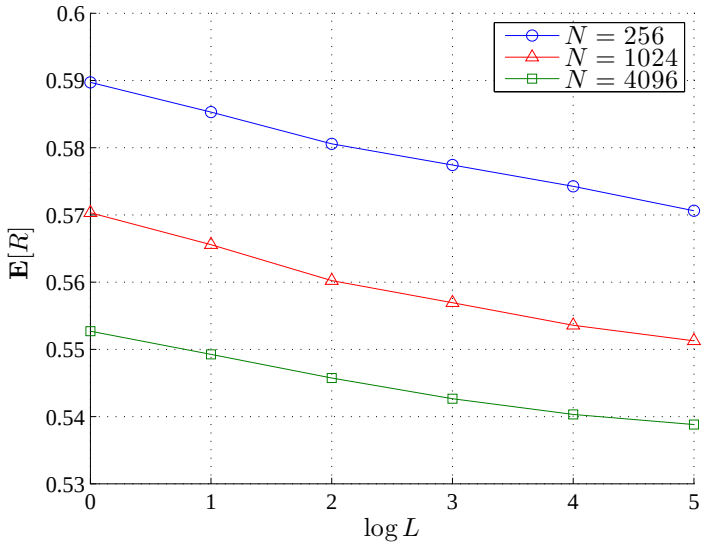

Figure 3.12: The improvement in the expected code rate by SCL-D for a ternary source with distribution $p_2 = (0.07,\ 0.09,\ 0.84)$ at various block-lengths.

Next, the performance of multi-level oracle-based lossless polar compression scheme for composite alphabet sizes is analyzed. For a 6-ary source with probability distribution $p_Z = (0.0077,\ 0.7476,\ 0.0675,\ 0.0623,\ 0.0924,\ 0.0225)$, the expected code rates under SC-D is given in Figure 3.13. The source Z is compressed in two layers: $Z = (X, Y)$, where Y is a ternary and X is a binary random variable. This example indicates that the extension of the proposed oracle-based polar compression scheme developed for prime-size alphabets to arbitrary finite alphabets provides similar performance.

In Figure 3.14 and Figure 3.15, the probability distribution and expected code rates in the compression of a source Z with alphabet size $q = 256$ and block-length $N = 2^{14}$ are given, respectively. The source is expanded into binary form for multi-level compression: $Z = (Z^7, Z^6, \ldots, Z^0)_2$. The blue bars indicate the

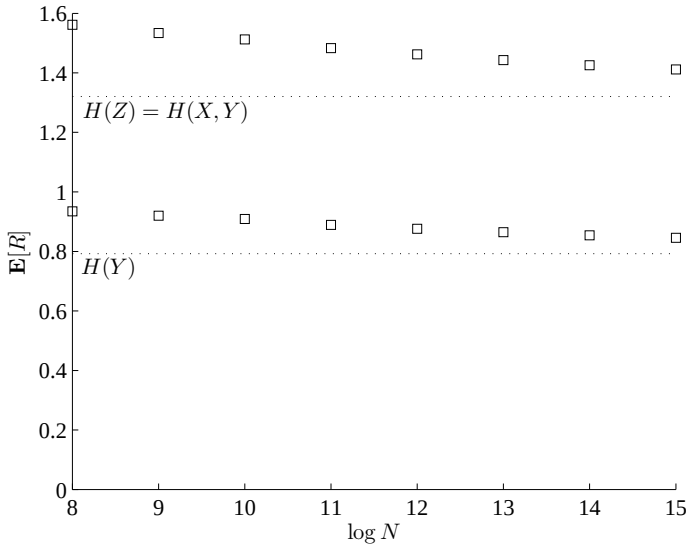

Figure 3.13: Expected code rates for a 6-ary source $Z = (X, Y)$ with probability distribution $p_Z = (0.0077,\ 0.7476,\ 0.0675,\ 0.0623,\ 0.0924,\ 0.0225)$. Base-2 entropy values are marked by dotted lines.

expected code rates for each level, and the red bars indicate the minimum source coding rate for the corresponding level.

For each level, the code rate provided by the oracle-based lossless polar compression is close to the theoretical limit.

In the second part of performance evaluations, the proposed sequential quantization with scaling algorithm for probability distributions and the CIS construction method are analyzed together under source uncertainty. The approach is based on Monte Carlo simulations by random vector sampling over the unit simplex, which is introduced in Section 3.6. In this approach, $\mathbf{p}_X$ is modelled as a random vector over the unit simplex $M(q)$. Each probability distribution is compressed by using β symbols, and a CIS $\mathcal{C}$ of size C is constructed using the scheme introduced in Subsection 3.6.2.1. In the construction of $\mathcal{C}$, the set $\{p_i\}_{i=0}^{C-1}$ is chosen such that $\mathcal{H}(\boldsymbol{\varphi}(p_{i+1})) - \mathcal{H}(\boldsymbol{\varphi}(p_i))$ is constant for all i. For a given realization $\mathbf{p}_X$, $m_c = 10000$ source realizations are drawn from $\mathbf{p}_X$, i.e., $X_0^{N-1} \overset{iid}{\sim} \mathbf{p}_X$, and the expected rate $\mathbb{E}_{X_0^{N-1}}[R|\mathbf{p}_X]$ is computed.

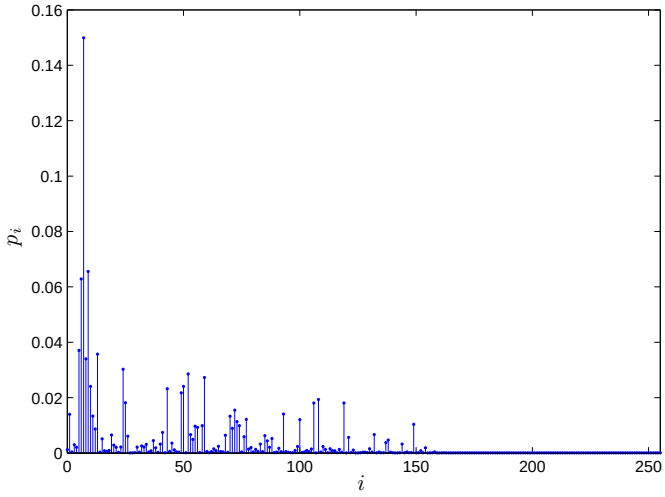

Figure 3.14: The probability distribution of a source $Z = (Z^7, Z^6, \ldots, Z^0)_2$ with alphabet size $q = 256$.

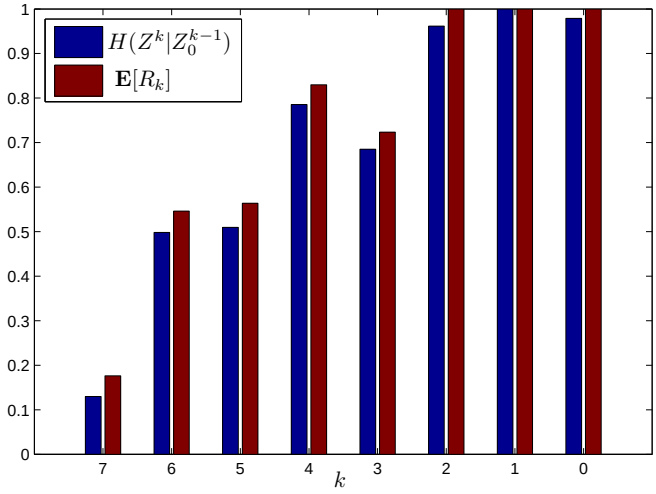

Figure 3.15: $\mathbb{E}[R_k]$ and $H(Z^k|Z_0^{k-1})$ values for each k at block-length $N = 2^{14}$.

In the first set of experiments, $\mathbf{p}_X[i]$ is sampled such that $\mathbf{p}_X[i] \in M(3)$ and $\mathcal{H}(\mathbf{p}_X)$ is uniformly distributed on the interval $[H_{min}, H_{max}]$ for two constants $0 \leq H_{min} < H_{max} \leq 1$ for $i = 0, 1, \ldots, m_s - 1$, where m_s is the number of randomly sampled probability distributions. $\mathcal{C}$ is constructed such that $\mathcal{H}(\boldsymbol{\varphi}_q(p_0)) = H_{min} + 1/C$, $\mathcal{H}(\boldsymbol{\varphi}_q(p_{C-1})) = H_{max}$ and with equal entropy difference between successive p_i.

For $N = 2^{12}$, $C = 8$, $H_{min} = 0.2$ and $H_{max} = 0.7$, expected code rates $\mathbb{E}[R_i|\mathbf{p}_X[i]]$ sorted such that $\mathcal{H}(\mathbf{p}_X[i]) \leq \mathcal{H}(\mathbf{p}_X[i+1])$ for $i = 0, 1, \ldots, m_s - 1$ are given in Figure 3.16 for $m_s = 1000$. The source distribution is quantized by using Algorithm 9 with $\beta = 3$ symbols per element.

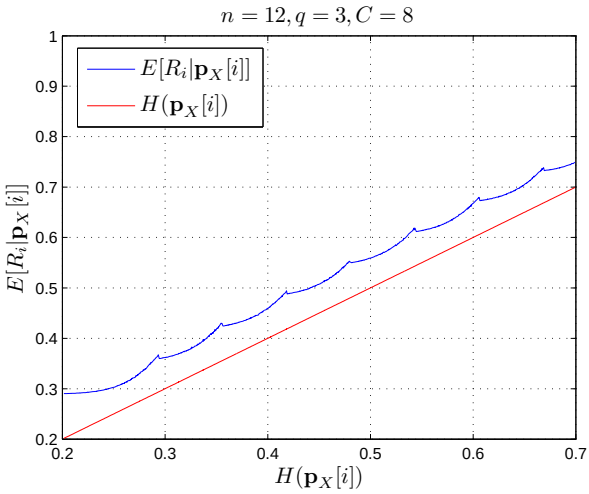

Figure 3.16: The average code rate $\mathbb{E}[R|\mathbf{p}_X]$ values if $\mathbf{p}_X$ is chosen randomly such that $\mathcal{H}(\mathbf{p}_X) \sim \mathcal{U}[0.2,\ 0.7]$, $C = 8$, $N = 2^{12}$, and $\beta = 3$

Two parameters are considered for performance measurement: Mean and maximum backoff from the minimum source coding rate over all $\mathbf{p}_X[i]$, $i = 0, 1, \ldots, m_s - 1$. In Figure 3.16, the following values are obtained:

$$\frac{1}{m_s} \sum_{i=0}^{m_s-1} \{\mathbb{E}[R_i|\mathbf{p}_X[i]] - \mathcal{H}(\mathbf{p}_X[i])\} = 0.0610,$$

and

$$\max_{i=0,\ldots,m_s-1} \{\mathbb{E}[R_i|\mathbf{p}_X[i]] - \mathcal{H}(\mathbf{p}_X[i])\} = 0.0888.$$

If the difference $|\mathcal{H}(\boldsymbol{\varphi}_q(p_{i^*})) - \mathcal{H}(\mathbf{p}_X)|$ is large for a source distribution $\mathbf{p}_X$, then the backoff from the minimum source coding rate becomes large. The ripples in

Figure 3.16 correspond to such probability distributions. In order to decrease the rate backoff at such distributions, increasing C is a solution. In Figure 3.17, the same procedure is repeated with $C = 16$.

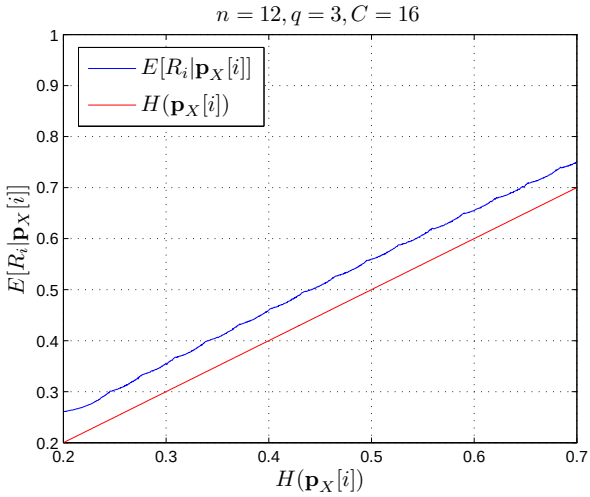

Figure 3.17: The average code rate $\mathbb{E}[R|\mathbf{p}_X]$ values if $\mathbf{p}_X$ is chosen randomly such that $\mathcal{H}(\mathbf{p}_X) \sim \mathcal{U}[0.2,\ 0.7]$, $C = 16$, $N = 2^{12}$, and $\beta = 3$

In this case,

$$\frac{1}{m_s} \sum_{i=0}^{m_s-1} \{\mathbb{E}[R_i|\mathbf{p}_X[i]] - \mathcal{H}(\mathbf{p}_X[i])\} = 0.0559,$$

and

$$\max_{i=0,\ldots,m_s-1} \{\mathbb{E}[R_i|\mathbf{p}_X[i]] - \mathcal{H}(\mathbf{p}_X[i])\} = 0.0624.$$

values are obtained, i.e., ripple magnitude is decreased.

In order to analyze the effect of increasing block-length on the code rate performance, the procedure is followed with parameters $N = 2^{14}$ and $C = 8$. In Figure 3.18, the performance of the coding scheme is analyzed.

From the ripple magnitude, it is observed that robustness to source uncertainty decreases as N increases due to the polarization effect. Mean and maximum backoff from the minimum source coding rate in Figure 3.18 are as follows:

$$\frac{1}{m_s} \sum_{i=0}^{m_s-1} \{\mathbb{E}[R_i|\mathbf{p}_X[i]] - \mathcal{H}(\mathbf{p}_X[i])\} = 0.0474,$$

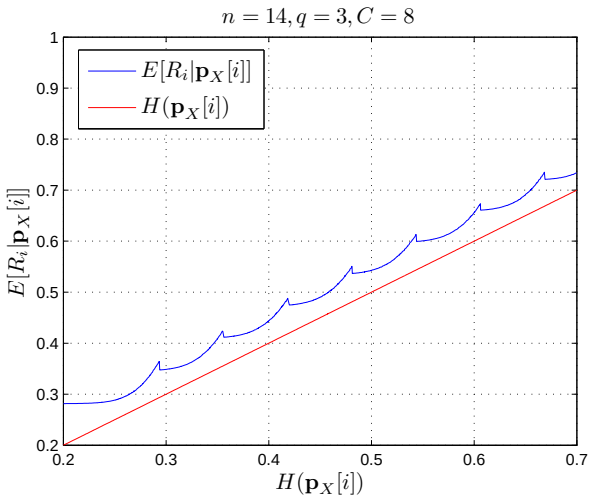

Figure 3.18: The average code rate $\mathbb{E}[R|\mathbf{p}_X]$ values if $\mathbf{p}_X$ is chosen randomly such that $\mathcal{H}(\mathbf{p}_X) \sim \mathcal{U}[0.2,\ 0.7]$, $C = 8$, $N = 2^{14}$, and $\beta = 3$

and

$$\max_{i=0,\dots,m_s-1} \{\mathbb{E}[R_i|\mathbf{p}_X[i]] - \mathcal{H}(\mathbf{p}_X[i])\} = 0.0815.$$

Hence, at large block-lengths, the size of the CIS must be chosen sufficiently large.

Finally, it is assumed that the probability distributions $\mathbf{p}_X$ is distributed uniformly over the following subset of $M(3)$:

$$D = \{\mathbf{p} \in M(3) : p_i \geq 0.05,\ i = 0, 1, 2\}.$$

$D \subset M(3)$ is illustrated in Figure 3.19. The procedure in the first set of experiments is performed for $N = 2^{12}$, $\beta = 6$ and $C = 16$ with $m_s = 1000$ probability vectors sampled from D. The resulting code rates are given in Figure 3.20. Mean and maximum backoff from the minimum source coding rate are as follows:

$$\frac{1}{m_s} \sum_{i=0}^{m_s-1} \{\mathbb{E}[R_i|\mathbf{p}_X[i]] - \mathcal{H}(\mathbf{p}_X[i])\} = 0.0529,$$

and

$$\max_{i=0,\dots,m_s-1} \{\mathbb{E}[R_i|\mathbf{p}_X[i]] - \mathcal{H}(\mathbf{p}_X[i])\} = 0.0793.$$

The increase in the backoff at probability vectors with high entropy stems from the CIS construction scheme, which causes high J at regions indicated in Subsection 3.6.2.1.

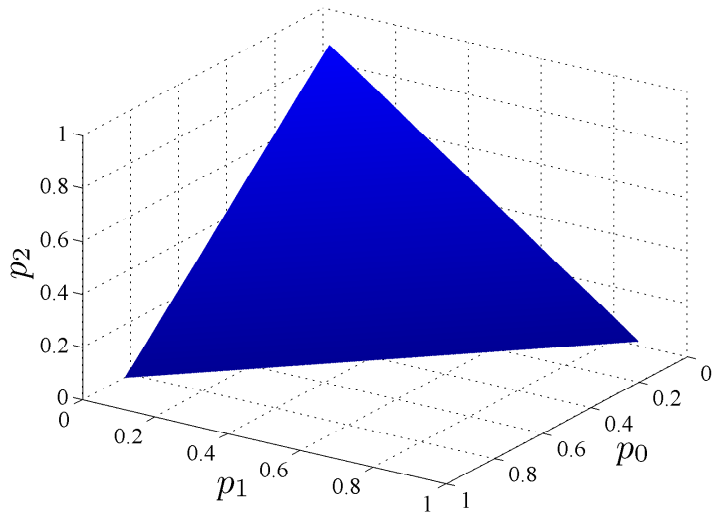

Figure 3.19: The region D over which $\mathbf{p}_X$ is distributed uniformly.

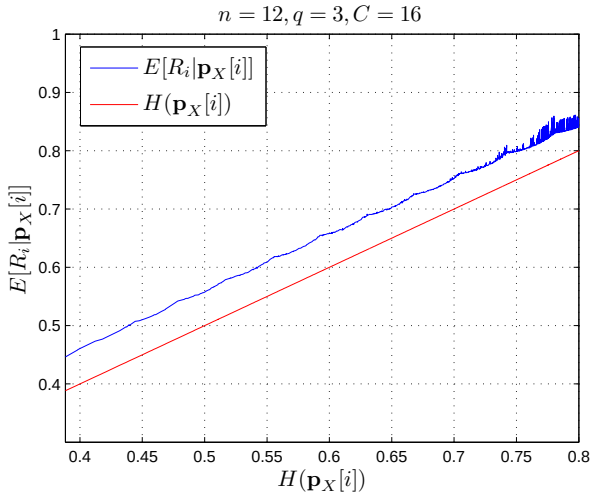

Figure 3.20: The average code rate $\mathbb{E}[R|\mathbf{p}_X]$ values for $C = 8$, $N = 2^{12}$, and $\beta = 6$

Chapter 4

Conclusions

In this thesis, a fixed-to-variable length, zero-error lossless data compression scheme with polar codes is proposed. The coding scheme achieves the fundamental limit, minimum source coding rate, asymptotically in the block-length, and provides acceptable backoff from this fundamental limit at practical block-lengths by low-complexity encoding and decoding algorithms. In order to improve performance at small to moderate block-lengths at the expense of increased complexity, a scheme based on successive cancellation list decoding is proposed. By using conditional source coding, a multi-level approach is adopted for extending the compression scheme from prime-size alphabets to arbitrary finite alphabets. The main advantage of the multi-level scheme is that zero error probability and small backoff from minimum source coding rate are inherited at any block-length with low-complexity encoding and decoding algorithms. Therefore, the scheme can be used in the compression of discrete memoryless sources over large alphabets, which is the case in most data compression applications such as image and speech compression. Since polar codes are structured codes, the receiver needs the exact information of source distribution and information set for non-binary alphabets [15]. The consequences of this property in the presence of source distribution uncertainty are two-fold: The source distribution must be known at the receiver, and the information set must be constructed for every given source distribution, which implies infeasibility in practical applications. In order to

solve these problems, a probability mass function quantization algorithm and a scheme for constructing and using pre-built information sets are proposed. For analyzing the robustness of the proposed schemes to source uncertainty, a Monte Carlo-based method is developed.

In [28] and [29], bounds are introduced for lossless data compression rate at finite block-lengths. For performance evaluation, the performance of the proposed compression scheme at non-asymptotic regime can be compared to these bounds.

As a first application of the proposed zero-error compression framework, DCT-based JPEG image compression using polar compression is proposed. For improved performance, polar JPEG compression schemes with Burrows-Wheeler Transform [30, 31] and second generation wavelets [32] can be implemented. In these schemes, a mixed multi-level approach with lossless and lossy polar compression methods can be utilized for performance improvement. Together with a method for exploiting the memory information, the proposed compression method can be used in the compression of any signal efficiently.

Bibliography

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

[2] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 471–480, 1973.

[3] R. M. Gray, "Conditional rate-distortion theory," *Technical report, No. 6502-2, Information Systems Laboratory, Stanford Electronics Laboratories*, 1972.

[4] S. C. Draper and E. Martinian, "Compound conditional source coding, slepian-wolf list decoding, and applications to media coding," in *IEEE International Symposium on Information Theory, ISIT 2007*, pp. 1511–1515, 2007.

[5] I. Csiszar and J. Korner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*. Orlando, FL, USA: Academic Press, Inc., 1982.

[6] N. Hussami, S. B. Korada, and R. Urbanke, "Performance of polar codes for channel and source coding," in *IEEE International Symposium on Information Theory, ISIT 2009*, pp. 1488–1492, 2009.

[7] S. B. Korada, "Polar codes for channel and source coding," *Ph.D. dissertation, Computer, Communication and Information Sciences, EPFL, Lausanne, Switzerland*, 2009.

[8] E. Arıkan, "Source polarization," in *IEEE International Symposium on Information Theory Proceedings, ISIT 2010*, pp. 899–903, 2010.

[9] H. S. Cronie and S. B. Korada, "Lossless source coding with polar codes," in *IEEE International Symposium on Information Theory Proceedings, ISIT 2010*, pp. 904–908, 2010.

[10] G. Caire, S. Shamai, and S. Verdu, *Noiseless data compression with low-density parity-check codes*. Advances in network information theory, Piyush Gupta, Gerhard Kramer, and Adriaan J. van Wijnggaardenl (Eds), Volume 66, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, 09 2004.

[11] E. Şaşoğlu, E. Telatar, and E. Arıkan, "Polarization for arbitrary discrete memoryless channels," *CoRR*, vol. abs/0908.0302, 2009.

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

[13] R. Mori and T. Tanaka, "Performance and construction of polar codes on symmetric binary-input memoryless channels," in *IEEE International Symposium on Information Theory, ISIT 2009*, pp. 1496–1500, 2009.

[14] I. Tal and A. Vardy, "How to construct polar codes," *CoRR*, vol. abs/1105.6164, 2011.

[15] E. Abbe, "Universal polar coding and sparse recovery," *CoRR*, vol. abs/1012.0367, 2010.

[16] E. Şaşoğlu, "Polar coding theorems for discrete systems," *Ph.D. dissertation, Computer, Communication and Information Sciences, EPFL, Lausanne, Switzerland*, 2011.

[17] I. Tal and A. Vardy, "List decoding of polar codes," *CoRR*, vol. abs/1206.0050, 2012.

[18] I. Tal, A. Sharov, and A. Vardy, "Constructing polar codes for non-binary alphabets and macs," in *IEEE International Symposium on Information Theory Proceedings, ISIT 2012*, pp. 2132–2136, 2012.

[19] R. Pedarsani, S. H. Hassani, I. Tal, and I. E. Telatar, "On the construction of polar codes," in *IEEE International Symposium on Information Theory Proceedings, ISIT 2011*, pp. 11–15, 2011.

[20] M. T. Heath, *Scientific Computing: An Introductory Survey*. McGraw-Hill Higher Education, 2nd ed., 2002.

[21] S. Çaycı and O. Arıkan, "Lossless polar compression of q-ary sources," in *IEEE International Symposium on Information Theory Proceedings, ISIT 2013*, 2013.

[22] M. Feder and N. Merhav, "Relations between entropy and error probability," *IEEE Transactions on Information Theory*, vol. 40, no. 1, pp. 259–266, 1994.

[23] R. M. Gray, "Toeplitz and circulant matrices: A review," *Foundations and Trends in Communications and Information Theory*, vol. 2, no. 3, 2005.

[24] B. A. Frigyik, A. Kapila, and M. R. Gupta, "Introduction to the dirichlet distribution and related processes," *Department of Electrical Engineering, University of Washington, UWEETR-2010-0006*, 2010.

[25] S. Onn and I. Weissman, "Generating uniform random vectors over a simplex with implications to the volume of a certain polytope and to multivariate extremes," *Annals of Operations Research*, vol. 189, no. 1, pp. 331–342, 2011.

[26] D. P. Kroese, T. Taimre, and Z. I. Botev, *Handbook of Monte Carlo Methods*, vol. 706. John Wiley & Sons, 2011.

[27] R. Tempo, G. Calafiore, and F. Dabbene, *Randomized Algorithms for Analysis and Control of Uncertain Systems with Applications*. Springer, 2nd ed., 2013.

[28] W. Szpankowski and S. Verdú, "Minimum expected length of fixed-to-variable lossless compression without prefix constraints," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4017–4025, 2011.

[29] S. Verdú and I. Kontoyiannis, "Lossless data compression rate: Asymptotics and non-asymptotics," in *46th Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–6, IEEE, 2012.

[30] M. Burrows and D. J. Wheeler, "A block-sorting lossless data compression algorithm," tech. rep., Digital Equipment Corporation, 1994.

[31] M. Effros, K. Visweswariah, S. R. Kulkarni, and S. Verdú, "Universal lossless source coding with the burrows wheeler transform," *IEEE Transactions on Information Theory*, vol. 48, no. 5, pp. 1061–1081, 2002.

[32] W. Sweldens, "The lifting scheme: A construction of second generation wavelets," *SIAM Journal on Mathematical Analysis*, vol. 29, no. 2, pp. 511–546, 1998.