GENERATOR MATRIX SELECTION FOR FINITE-LENGTH POLAR CODES

by

Berksan Şerbetci

B.S., Electrical and Electronics Engineering, Middle East Technical University, 2009

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Electrical and Electronics Engineering
Boğaziçi University
2012

GENERATOR MATRIX SELECTION FOR FINITE-LENGTH POLAR CODES

APPROVED BY:

Assist. Prof. Ali Emre Pusane ....
(Thesis Supervisor)

Prof. Hakan Deliç ....

Assoc. Prof. Murat Uysal ....

DATE OF APPROVAL:

ACKNOWLEDGEMENTS

Annem *Seval* ve babam *K. Cüneyt Şerbetci*'ye; ilgisini ve desteğini hiçbir zaman eksik etmeyen ve kısıtlı sürede elimizden geldiğince güzel işler çıkarmamızda en büyük pay sahibi olan danışmanım *Ali Emre Pusane*'ye; varlıklarıyla süreci katlanılabilir hale getiren ve benden desteğini esirgemeyen tüm arkadaşlarıma ve *Büyük Fenerbahçe Camiası*'na en içten teşekkürlerimi ve saygılarımı sunuyorum.

ABSTRACT

GENERATOR MATRIX SELECTION FOR FINITE-LENGTH POLAR CODES

Polar coding is a recently proposed coding technique, which has been proven to achieve the channel capacity. The original 2×2 generator matrix polarizes the channels and a portion of channels' capacity approach 1, while the remaining channel capacities approach 0. In the literature, it was shown that, as the codelength goes to infinity, polarization performance of Arıkan's 2×2 matrix is better than any matrix of size less than 16×16. In this thesis, we show that this observation does not necessarily hold for the finite-length case and the channel polarization is attainable by using different generator matrices. The main contribution of this thesis consists of filling the gap on the analysis of the finite-length polar code generation. A normalized polarization distance measure was defined and polar codes from different generator matrices showing different amount of polarization effects were obtained using this measure. Also, the coding structure for these generalized polar codes were obtained. Polarization performances in both asymptotical and finite-length cases were investigated especially for generator matrices of size 3×3 and 4×4 using Bhattacharyya parameter histograms, polarization rate exponents and normalized polarization distance measures; also upper bound on block error probabilities for these matrices were analyzed. Moreover, the recursive likelihood ratio equations for a specific 4×4 matrix showing the best polarization performance among all 4×4 generator matrices were defined. A decoding algorithm was implemented for a generator matrix from the best group of 4×4 generator matrices and its erasure rate was compared with the Arıkan's original generator matrix' decoding performance.

ÖZET

SONLU UZUNLUKLU KUTUPSAL KODLAR İÇİN ÜRETİM MATRİSİ SEÇİMİ

Kutupsal kodlar yakın zamanda ortaya atılmış, kanal kapasitesine eriştiği ispatlanmış bir kod ailesidir. Bu kodların üretiminde kullanılan 2×2'lik temel matris ele alınan kanalları kutuplaştırarak bir kısım kanalın kapasitesinin 1'e, kalanların kapasitelerinin ise 0'a yakınsamalarına neden olmaktadır. Literatürde rastlanılan çalışmalarda sonsuz kod uzunlukları için 16×16'dan küçük matrislerin Arıkan'ın 2×2'lik temel matrisinden daha iyi kutuplaşma başarımına sahip olmadığı gösterilmiştir. Bu çalışmada sonlu kod uzunluğu durumunda bu gözlemin doğru olmak zorunda olmadığı ve değişik üretim matrisleri kullanılarak kanal kutuplaşmasının sağlanabileceği gösterilmiştir. Bu tezin ana amacı sonlu uzunluklu kod tasarımı konusunda literatürdeki boşluğu doldurmaktır. Bunu sağlamak için normalleştirilmiş bir uzaklık ölçevi tanımlanmış ve bu ölçev kullanarak farklı miktarda kutuplaşma başarımı sağlayan farklı temel üretim matrisleri kullanarak oluşturulmuş kutupsal kodlar ve bu genellştirilmiş kodlar için kodlama yapıları elde edilmiştir. 3×3 ve 4×4 boyutuna sahip temel matrislerin hem sonsuz hem de sonlu uzunluklu kutuplaşma başarımları Bhattacharyya değiştirgenleri histogramları, kutuplaşma oranı üstelleri ve normalleştirilmiş kutuplaşma uzaklık ölçevleri kullanılarak incelenmiş ve bu üretim matrisleri için üst sınır blok hata olasılıkları analiz edilmiştir. Bununla birlikte 4×4 boyutlu üretim matrisleri arasında en fazla kutuplaşma başarımını gösteren gruptan bir matris için özyineli olabilirlik oranı denklemleri tanımlanmıştır. 4×4 boyutlu üretim matrisleri arasında en iyi başarımı gösteren gruptan bir matris için kodçözüm algoritması geliştirilmiş ve hata oranı Arıkan'ın orijinal üretim matrisinin kodçözüm başarımıyla karşılaştırılmıştır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$\oplus$	Modulo-2 sum
$\mathcal{A}$	Information set
$\mathcal{A}^c$	Frozen set
B_N	Bit-reversal operator
C	Capacity
d_p	Normalized polarization distance measure
D_i	Partial distance
$E(G)$	Polarization rate
F	Original generator matrix
G	Generator matrix
$H(X)$	Entropy of a RV X
$H(X\|Y)$	Conditional entropy of the RV X conditional on the RV Y
i	Bit index
I_N	N-dimensional identity matrix
$I(X;Y)$	Mutual information between the RV pair X and Y
$I(W)$	Symmetric capacity
K	Dimension of the code
n	Number of recursions
N	Block length
$\mathcal{N}$	Normal distribution
p	Crossover probability
$p(x)$	Probability mass function of RV X
P_e	Erasure (crossover) probability
R	Code rate
R_N	Reverse shuffle operator
W	Channel capacity
W_N	Combined channel
$W_N^{(i)}$	Split channel

X	Random variable
$\mathcal{X}$	Input alphabet
$\mathcal{Y}$	Output alphabet
$Z(W)$	Bhattacharyya parameter

α	Erasure probability
ϵ	Erasure probability for BEC
$\Phi(x)$	Cumulative normal function

LIST OF ACRONYMS/ABBREVIATIONS

AWG	Additive White Gaussian
B-DMC	Binary Discrete Memoryless Channel
BEC	Binary Erasure Channel
BER	Bit Error Rate
BP	Belief Propagation
BPSK	Binary Phase Shift Keying
BSC	Binary Symmetric Channel
DE	Decision Element
DMC	Discrete Memoryless Channel
FER	Frame Error Rate
FFT	Fast Fourier Transform
LDPC	Low Density Parity Check
LP	Linear Programming
LR	Likelihood Ratio
ML	Maximum Likelihood
RM	Reed-Muller
RV	Random Variable
SC	Successive Cancellation

1. INTRODUCTION

In 1940s, after the proposition of Hamming codes, channel coding techniques received a considerable amount of attention. With channel coding, the error performance of the communication system can be improved without decreasing the communication throughput. The main idea is to correct errors that occur as a result of the channel noise at the decoder. Channel coding technique is based on the term redundancy. In channel coding systems, redundant information is added to the transmission sequence (encoding) and this longer code sequence (called a codeword) is sent to the channel. Considering the effects of the channel noise, the redundant part of the received vector is used to estimate the original information sequence from the noisy received vector (decoding). In the early years of this research field, researchers attempted to design efficient encoding and decoding schemes that could employ various levels of redundancy. Without proper design guidelines or a theoretical framework, though, most of these designs performed poorly.

Shannon, in his 1948 paper [1], calculated the minimum amount of redundancy that must be added to ensure error-free communications. In the same paper, he also proved that error-free communication is impossible for smaller values of the redundancy. Equivalently, it is impossible to achieve error-free communication as long as the code rate R, defined as the ratio between the message and code lengths, is larger than some value called the channel capacity W. On the other hand, it is possible to obtain error-free communication if the communication rate is less than the channel capacity. Under the circumstances, error-free communication is viable if and only if the redundancy is larger than an exact value.

Prior to 1990s, researchers systematically designed coding schemes that operate with minimal redundancy requirements (closer to the channel capacity). However, the infeasible code lengths required for good error performance made it impossible to obtain practical coding schemes that approach the capacity. The beginning of the

1990s witnessed the emergence of iteratively decodable codes, a class of codes with sparse representations that allow them to be decoded efficiently with (suboptimal) iterative decoding algorithms. Even though the employed decoding algorithms were suboptimal, the extremely low implementation complexity (usually linear in the code length) allowed codes with much larger code lengths to be decoder. Among these codes, the most famous one is the family of low-density parity-check (LDPC) codes, which were first introduced in 1962 by Gallager, [2, 3]. These codes, together with their sparse graph representations were the first family of codes that could approach the channel capacity via simulations, e.g., a code family with a decoding threshold $0.0045dB$ away from the channel capacity was successfully designed in [4].

Even though LDPC codes were shown to approach the channel capacity, the required code lengths to actually do so is still in the order of several millions and as a code's error performance approaches channel capacity, the associated error floor rises and the code becomes unusable for high signal-to-noise ratios. Regardless, LDPC code family was still regarded as the best and even the only solution of approaching the channel capacity, until Arıkan proposed channel polarization and the associated polar coding technique to provably achieve the channel capacity, [5, 6]. According to channel polarization, a channel cluster undergoes a transformation and the capacity of the channels in the newly obtained channel set *polarizes* (for sufficiently large number of channels), i.e., the channel capacities converge to either zero or one. If channel transformation process is applied as coding, information sequence becomes ready to be sent from the channels with high capacity. Channel polarization, in this regard, is the first channel capacity achieving code family that can be proven analytically in the literature and allows us to obtain the polar code family, which has low complexity in both encoding and decoding steps. Error performance of polar codes, which are constructed using a technique similar to digital signal processing techniques such as Fast Fourier Transform (FFT), was analyzed in [7, 8]. A polar code is actually an extension of a Reed-Muller (RM) code and the relation between these two code families is extensively studied in [9].

The speed of channel polarization was scrutinized in [10–12]. Practical and finite channel polarization and scaling of polar codes were studied in [13,14] and [15]. In order to shed some light on the realization on designing practical polar codes, error bounds, and exponents were studied in [16, 17] and more studies were made on the construction of polar codes with low complexity in [18] and [19]. At first, Arıkan had showed that polarization only occurs for binary channels; his results were later generalized to discrete memoryless channels (DMC) in [20] and it was also shown that in binary-input memoryless channels, channel polarization is possible with a class of transformations [21].

Another study on structural properties of polar codes was systematic polar coding [22]. In the original polar code construction, the information sequence and the redundant part are blended while obtaining the coding sequence; whereas in systematic polar codes, the information sequence directly appears as part of the codeword, expediting the encoding and decoding processes. On the decoding front, linear programming (LP) and list decoding of polar codes were studied in [23, 24].

Notwithstanding that the studies are mainly focused on structural properties, practical encoding and decoding techniques of polar codes and channel polarization; there are also many studies on the feasibility of polar codes in different applications. It is shown in [25, 26] that polar codes are optimal for lossy source coding. Lossless source coding with polar codes is studied in [27,28] and polar coding for q-ary source coding is shown in [29].

The last group of studies in the literature for polar coding is about the suitability of the technique within different communication systems. Concatenated polar codes were analyzed in [30]. A practical approach that uses LDPC codes and polar codes together is proposed in [31]. Finally, the use of polar codes in wiretap and relay channels, multiple-access-channels, non-binary channels and channels with memory are also studied in [32–36].

In this thesis, polarization performance of different polar code generator matrices is analyzed. Main focus is on the finite-length polarization behavior of polarized channels generated by different size of generator matrices and these matrices were evaluated via both asymptotical polarization rate exponents and finite-length polarization measures. Since comparing different generator matrices with the same size is not easy via histogram plots, specifically, a finite-length normalized polarization distance measure is proposed in order to compare different polarization structures and provide distance plots. This measure is universal, i.e., can be used for any proposed specific code rate. In other words, the generator matrix showing the best polarization performance at a specific size for any given rate can be determined with this measure. Upper bound on block error probability for the generator matrices with different size and different polarization level are given and compared with Arıkan's original generator matrix' bound. Also, from the decoding point of view, the recursive likelihood ratio equations for a specific 4×4 matrix showing the best polarization performance among all 4×4 generator matrices are computed and a decoding algorithm is implemented for a generator matrix again from the group of showing the best polarization performance among 4×4 generator matrices; its erasure rate is compared with the Arıkan's original generator matrix' decoding performance in the decoding section.

2. CHANNEL POLARIZATION

2.1. Introduction

For any given binary-input symmetric channel, the channel capacity ,W, is an easy-to-compute parameter and satisfies the inequality $0 \leq W \leq 1$. Based on Shannon's channel capacity theorem, the coding rate of a communication channel $0 \leq R \leq 1$ also fits into this inequality as $0 \leq R \leq W \leq 1$ for error-free communication. If the two extreme values the capacity takes on are considered, i.e., $W = 0$ and $W = 1$, it is seen that the code design problem is rather simplified. For the case of $W = 0$, the code rate satisfies $R = 0$ and error-free communication is impossible regardless of the coding scheme employed, namely no information can be transmitted across the channel. For the case of $W = 1$, the code rate can be simply chosen as $R = 1$, which corresponds to the uncoded case and still the error-free communication is achieved, namely uncoded bits can be transmitted across the channel. For any value of W that falls strictly between the two extreme cases, things get complicated. Returning to the two already mentioned extreme cases, they are two weaknesses in Shannon's riddle and one can avoid the task of constructing codes for those cases. Arıkan's *channel polarization* and the associated *polar coding* techniques exploit these weaknesses.

The polar coding scheme proposed in [6] takes a set of N binary-input symmetric channels of capacity $W = \delta$, transforms them (via a recursive structure) to another set of channels with capacity values approaching either $W = 0$ or $W = 1$. In the asymptotical case ($N \to \infty$), $N\delta$ channel capacity values approach $W = 1$, whereas the remaining $N(1 - \delta)$ approach $W = 0$. These channels are said to polarize to the two extreme cases: bad channels that do not allow reliable transmission of any data, and good channels that allow reliable transmission of data even without any use of further coding. Also the total channel capacity of the two sets before and after polarization is preserved at $W_{\text{total}} = N\delta$. Among the newly created channels, even uncoded information symbols can be sent via channels approaching $W = 1$, whereas channels approaching $W = 0$ become redundant and remain unused. This channel cluster transformation and distribution of information sequence to channels with high capacity is together called polar coding.

N channels *Channel polarization* $NI(W)$ channels with capacity 1.
with capacity $I(W)$ $\longrightarrow$ $N(1 - I(W))$ channels with capacity 0.

Channel polarization is a recursive method that is used to define provably capacity-achieving polar code sequences with low encoding and decoding complexities.
A generic B-DMC is defined as $W : \mathcal{X} \rightarrow \mathcal{Y}$ with input alphabet $\mathcal{X}$, output alphabet $\mathcal{Y}$, and transition probabilities $W(x|y)$, $x \in \mathcal{X}$, $y \in \mathcal{Y}$. W^N is to denote the channel corresponding to N uses of W; thus, $W : \mathcal{X}^N \rightarrow \mathcal{Y}^N$ with $W^N(y_1^N|x_1^N) = \prod_{i=1}^{N} W(y_i|x_i)$.
Given a B-DMC W, the symmetric capacity is given as

$$I(W) \triangleq \sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} \frac{1}{2} W(y|x) \log \frac{W(y|x)}{\frac{1}{2}W(y|0) + \frac{1}{2}W(y|1)} \tag{2.1}$$

and the Bhattacharyya parameter is given as

$$Z(W) \triangleq \sum_{y \in \mathcal{Y}} \sqrt{W(y|0)W(y|1)}. \tag{2.2}$$

The symmetric capacity is used as the measure of *rate*, and the Bhattacharyya parameter is used as the measure of *reliability*. As already mentioned in the previous Section, the effective rate of transmission in average is equal to the capacity. Therefore, it is possible to say that $I(W)$ is the highest rate at which reliable communication is possible across W. The Bhattacharyya parameter $Z(W)$, is an upper bound on the probability of maximum-likelihood (ML) decision error on one-shot transmission of W. $Z(W)$ is an easy-to-compute value compared to actual symmetric capacity, intuitively it is easy to see that $I(W) \approx 1$ iff $Z(W) \approx 0$ and $I(W) \approx 0$ iff $Z(W) \approx 1$. The relationship between two values is inversely proportional; moreover, $I(W)$ is exactly equal to $1 - Z(W)$ for BEC and this fact allows us to procure exact channel capacity values easily.

2.2. How It Works?

N independent copies of a given B-DMC W is transferred into another set of channels $\{W_N^{(i)} : 1 \leq i \leq N\}$ that show a polarization effect. As $N \to \infty$, the symmetric capacity terms $\{I(W_N^{(i)})\}$ approach either 0 or 1.

2.2.1. Channel combining

This phase renders service to the aim of expanding the number of the channels in an interconnected manner. The copies of the given B-DMC W are combined in a recursive manner to produce the vector $W_N : X^N \to Y^N$, where N can be a power of two, i.e., $N = 2^n$. At the 0-th level ($n = 0$), there is only one copy of W and W_1 is set to the value W. The first level ($n = 1$) of the recursion considers a generator matrix of size 2×2 (actually this is the reason why $N = 2^n$ must be a power of 2). This level corresponds to combining two independent copies of a single channel in order to create $W_2 : \mathcal{X}^2 \to \mathcal{Y}^2$ using this generator matrix,

$$G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}.$$

The corresponding input connection is shown in Figure 2.1. The transition probabilities for W_2 is calculated as follows

$$W_2(y_1, y_2 | u_1, u_2) = W(y_1 | u_1 \oplus u_2) W(y_2 | u_2). \tag{2.3}$$

where ⊕ symbol denotes the modulo-2 sum.

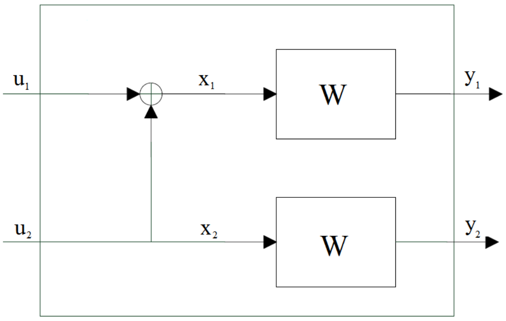

Figure 2.1. Combination of two W channels using G_2: the W_2 channel.

The next level of recursion consists of two independent copies of W_2 combined together to create the channel $W_4 : \mathcal{X}^4 \to \mathcal{Y}^4$ with transition probabilities $W_4(y_1^4|u_1^4) = W_2(y_1^2|u_1 \oplus u_2, u_3 \oplus u_4)W_2(y_3^4|u_2, u_4)$. The corresponding channel is shown in Figure 2.2.

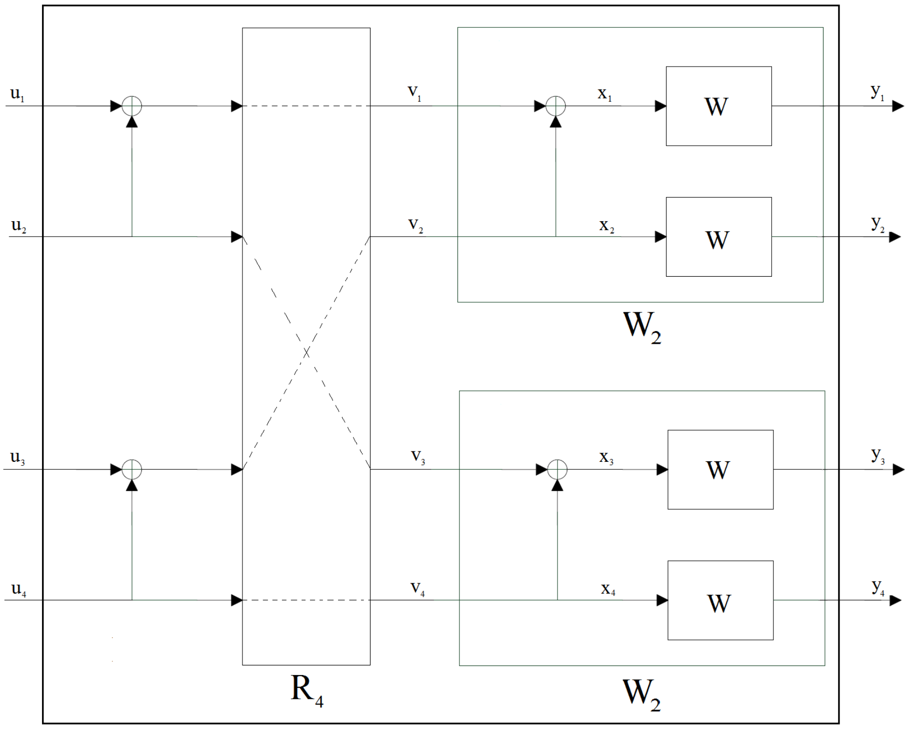

Figure 2.2. The channel W_4.

In Figure 2.2, R_4 is the permutation operation that maps an input (s_1, s_2, s_3, s_4) to $v_1^4 = (s_1, s_3, s_2, s_4)$. The mapping from the input of W_4 to the input of W^4

$(u_1^4 \to x_1^4)$, can be written as $x_1^4 = u_1^4 G_4$ where $G_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$. Therefore, the relation between the transition probabilities between W_4 and W^4 becomes $W_4(y_1^4|u_1^4) = W^4(y_1^4|u_1^4 G_4)$.

The general form of the recursion is shown in Figure 2.3. Two independent copies of $W_{N/2}$ are combined using the *generator matrix* and the *reverse shuffle* operation to produce the channel W_N.

The input vector u_1^N is first transformed into s_1^N so that $s_{2i-1} = u_{2i-1} \oplus u_{2i}$ and $s_{2i} = u_{2i}$ for $1 \leq i \leq N/2$. The reverse shuffle operator R_N acts on its input to produce v_1^N such that $v_1^N = (s_1, s_3, ..., s_{N-1}, s_2, s_4, ..., s_N)$ and the output of the reverse shuffle operator, v_1^N becomes the input to the two copies of $W_{N/2}$. It is easy to observe that the whole mapping process is linear over $GF(2)$. Therefore, the overall process can be represented by a matrix G_N so that $x_1^N = u_1^N G_N$. G_N is called the *generator matrix* of size N. The relation between transition probabilities becomes

$$W_N(y_1^N|u_1^N) = W^N(y_1^N|u_1^N G_N) \tag{2.4}$$

for all $y_1^N \in \mathcal{Y}^N$, $u_1^N \in \mathcal{X}^N$. $G_N = B_N F^{\otimes n}$ for any $N = 2^n$, $n \geq 0$, where ${}^{\otimes n}$ denotes the n^{th} degree Kronecker product, B_N is the bit-reversal operator and

$$F \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}.$$

B_N permutation matrix can be computed recursively as

$$B_N = R_N(I_2 \otimes B_{N/2}) \tag{2.5}$$

where R_N is the reverse shuffle operator and I_2 is the 2-dimensional identity matrix. The recursion is initialized with $B_2 = I_2$.

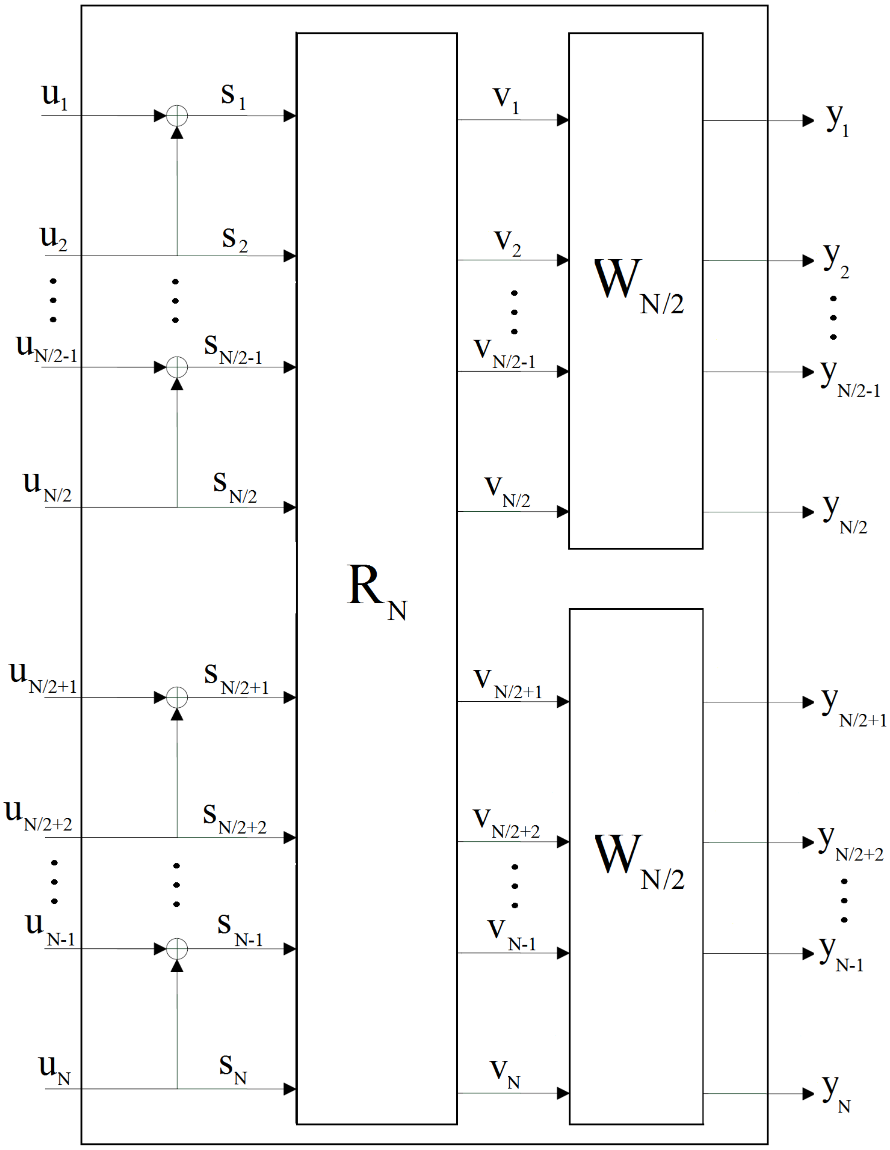

Figure 2.3. The channel W_N.

Channel combining operation is fully dependent on the original generator matrix F. G_N and $F^{\otimes n}$ have the same set of rows with different (bit-reversed) orders.

2.2.2. Channel splitting

In order to be able to see the polarization effect on the channels individually at the output section, W_N should be split back into a set of N binary-input coordinate channels $W_N^{(i)} : \mathcal{X} \to \mathcal{Y}^N \times \mathcal{X}^{i-1}$, $1 \leq i \leq N$, with transition probabilities

$$W_N^{(i)}(y_1^N, u_1^{i-1}|u_i) \triangleq \sum_{u_{i+1}^N \in \mathcal{X}^{N-i}} \frac{1}{2^{N-1}} W_N(y_1^N|u_1^N), \tag{2.6}$$

where (y_1^N, u_1^{i-1}) denotes the output of $W_N^{(i)}$ with u_i as its input.
To see how the operation goes on at the first sight, a successive cancellation (SC) decoder estimates the ith decision element u_i after observing y_1^N and the past correctly supplied channel inputs u_1^{i-1}. $W_N^{(i)}$ is the effective channel seen by the ith decision element (DE) if u_1^N is a-priori uniform on $\mathcal{X}^N$.

2.2.3. Channel polarization

Theorem 2.1 (Arıkan [6]). *For any B-DMC W, the channels $\{W_N^{(i)}\}$ polarize in the sense that, for any fixed $\delta \in (0,1)$, as $N \to \infty$, a fraction of indices $i \in \{1, ..., N\}$ for which $I(W_N^{(i)}) \in (1-\delta, 1]$ goes to $I(W)$ and the fraction for which $I(W_N^{(i)}) \in [0, \delta)$ goes to $1 - I(W)$.*

Therefore, the polarization performance gets better as the case becomes asymptotical; namely, the channels become either completely noisy or noise free and they are known at the transmitter side so that bad channels become frozen and uncoded bits can be sent over the good channels.

The numbers $\{I(W_N^{(i)})\}$ can be computed using the recursive relations

$$\begin{aligned} I(W_N^{(2i-1)}) &= I(W_{N/2}^{(i)})^2, \\ I(W_N^{(2i)}) &= 2I(W_{N/2}^{(i)}) - I(W_{N/2}^{(i)})^2, \end{aligned} \tag{2.7}$$

with $I(W_1^{(1)}) = 1 - \epsilon$. The recursion only stands for BECs. The reason is shown in the following propositions:

Proposition 2.1. [Arıkan [6]]: Suppose $(W, W) \to (W', W'')$ for some set of binary-input channels. Then,

$$Z(W'') = Z(W)^2, \tag{2.8}$$

$$Z(W') \leq 2Z(W) - Z(W)^2, \tag{2.9}$$

$$Z(W') \geq Z(W) \geq Z(W''). \tag{2.10}$$

Equality holds in Equation 2.9 iff W is a BEC. Therefore, $Z(W') = Z(W'')$ iff $Z(W)$ equals to 0 or 1, or in other words iff $I(W)$ is equal to 1 or 0. Then the reliability can only improve in the sense that

$$Z(W') + Z(W'') \leq 2Z(W) \tag{2.11}$$

with equality iff W is BEC. Hence, BEC is the local extreme case in the sense of behavior of reliability.

Proposition 2.2. [Arıkan [6]]: Considering the channel transformation $(W, W) \to (W', W'')$; if W is a BEC with some erasure probability ϵ, then the channels W' and W'' are BECs with erasure probabilities $2\epsilon - \epsilon^2$ and ϵ^2, respectively.

For $N = 2^{10} = 1024$ channels with channel capacity $W = 0.5$, using the recursive relations, the sorted polarization effect is plotted in Figure 2.4; where among $N = 1024$ constructed channels, nearly 400 of them have $W \approx 0$, 400 of them have $W \approx 1$, and the rest have capacity in some mid-values. In the asymptotical case, the ratio of these intermediate channels is expected to diminish.

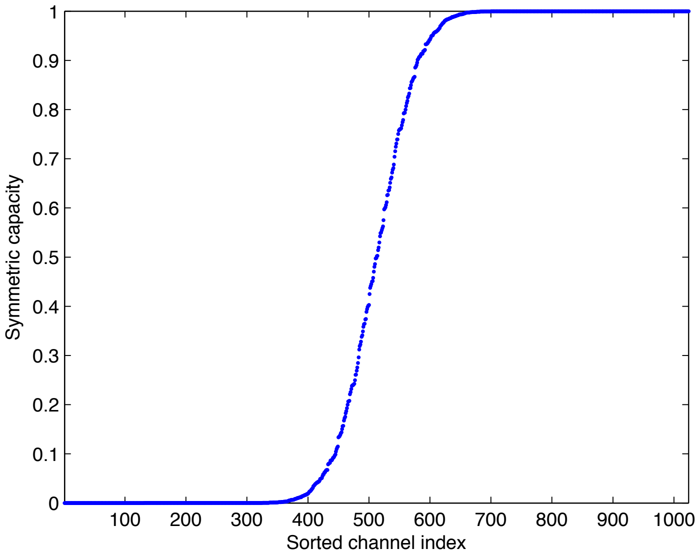

Figure 2.4. Polarization effect for $N = 2^{10}$ for a BEC with $\epsilon = 0.5$.

2.2.4. Rate of polarization

The speed with which the polarization effect is captured as a function of N is an important parameter and it is given as follows

$$Z(W_N^{(i)}) = \sum_{y_1^N \in \mathcal{Y}^N} \sum_{u_1^{i-1} \in \mathcal{X}^{i-1}} \sqrt{W_N^{(i)}(y_1^N, u_1^{i-1}|0) W_N^{(i)}(y_1^N, u_1^{i-1}|1)}. \tag{2.12}$$

2.3. Polar Coding

The polarization effect phenomenon is used to construct codes achieving the symmetric channel capacity $I(W)$ by a method called polar coding. The main idea is to be able to access any coordinate channel $W_N^{(i)}$ individually and send data only through the good ones, namely the ones for which $Z(W_n^{(i)})$ is near 0 or equivalently $I(W_n^{(i)})$ is near 1.

2.3.1. G_N coset codes

Polar code family is a subset of G_N coset codes. The block lengths for this class are in the form of $N = 2^n$, $n \geq 0$ and each code is encoded in the same manner, namely

$$x_1^N = u_1^N G_N = u_{\mathcal{A}} G_N(\mathcal{A}) \oplus u_{\mathcal{A}^c} G_N(\mathcal{A}^c), \tag{2.13}$$

where G_N is the generator matrix, $\mathcal{A}$ is an arbitrary subset of $\{1, ..., N\}$, $G_N(\mathcal{A})$ is the submatrix of G_N formed by the rows with indices in $\mathcal{A}$.

The mapping from source blocks $u_{\mathcal{A}}$ to codeword blocks x_1^N is obtained if $\mathcal{A}$ and $u_{\mathcal{A}^c}$ is fixed and $u_{\mathcal{A}}$ is left as a free variable. The obtained mapping is a coset code of the linear block code with generator matrix $G_N(\mathcal{A})$, with the coset set from the fixed vector $u_{\mathcal{A}^c} G_N(\mathcal{A}^c)$. The name of these codes are called G_N-coset codes each with parameter vector $(N, K, \mathcal{A}, u_{\mathcal{A}^c})$, where K is the dimension of the code and determines the size of $\mathcal{A}$. The ratio K/N is the called code rate, $\mathcal{A}$ is the information set, and $u_{\mathcal{A}^c} \in \mathcal{X}^{N-K}$ are the frozen bits. Let

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

Then the $(4, 2, \{2, 4\}, (1, 0))$ code has the encoder mapping given by

$$\begin{aligned} x_1^4 &= u_1^4 G_4 \\ &= (u_2, u_4) \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} + (1, 0) \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}. \end{aligned} \tag{2.14}$$

For $(u_2, u_4) = (1, 1)$, the coded block is $x_1^4 = (1, 1, 0, 1)$. By choosing a particular information set $\mathcal{A}$, polar codes can be specified.

2.3.2. A successive cancellation decoder

For a G_N-coset code with parameter $(N, K, \mathcal{A}, u_{\mathcal{A}^c})$, after the input u_1^N is encoded into codeword x_1^N and this code is sent over the channel W^N, output y_1^N is received. Then the decoder generates an estimate $\hat{u}_1^N$ of u_1^N. Since the estimated bits in the frozen part are simply set to the actual input values, the real task is to estimate $\hat{u}_{\mathcal{A}}$ of $u_{\mathcal{A}}$. The decision for a bit is computed as

$$\hat{u}_i \triangleq \begin{cases} u_i, & \text{if } i \in \mathcal{A}^c, \\ h_i(y_1^N, \hat{u}_1^{i-1}), & \text{if } i \in \mathcal{A}, \end{cases}$$

in the order i from 1 to N, $h_i : \mathcal{Y}^N \times \mathcal{X}^{i-1} \to \mathcal{X}$, $i \in \mathcal{A}$ and decision functions defined as

$$h_i(y_1^N, \hat{u}_1^{i-1}) \triangleq \begin{cases} 0, & \text{if } \frac{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1}|0)}{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1}|1)} \geq 1, \\ 1, & \text{otherwise.} \end{cases}$$

A block error occurs if $\hat{u}_1^N \neq u_1^N$. The decision functions may resemble ML decision functions but they are not much similar because they treat the *future* frozen bits as RV's but in our case these bits are fixed. Therefore, $\{h_i\}$ values are computed using recursive formulas instead. At this point, all to the good to see that the loss in performance due to not using the total ML decision functions is negligible and the symmetric capacity is still achievable.

2.3.3. Code performance

The probability of block error for a $(N, K, \mathcal{A}, u_{\mathcal{A}^c})$ code can be bounded as follows

$$P_e(N, K, \mathcal{A}, u_{\mathcal{A}^c}) \leq \sum_{i \in \mathcal{A}} Z(W_N^{(i)}). \tag{2.15}$$

According to this result, choosing $\mathcal{A}$ among all K-subsets of $\{1, ..., N\}$ should result in minimizing the right hand side of Equation 2.15.

2.3.4. Polar codes

From the previous sections, a conclusion can be made that a G_N-coset code is said to be a polar code if the set $\mathcal{A}$ is chosen such that $Z(W_N^{(i)}) \leq Z(W_N^{(j)})$ for all $i \in \mathcal{A}$ and $j \in \mathcal{A}^c$. By this rule, it is possible to achieve $I(W)$ regardless of the choice of $u_{\mathcal{A}^c}$.

2.3.5. Coding theorems

Theorem 2.2 (Arıkan [6]). *For any B-DMC W and any fixed $R < I(W)$, there exists a sequence of sets $\mathcal{A}_N \subset \{1, ..., N\}$, $N \in \{1, 2, ..., 2^n, ...\}$ such that $|\mathcal{A}_N| \geq NR$ and $Z(W_N^{(i)}) \leq O(N^{-\frac{5}{4}})$ for all $i \in \mathcal{A}_N$.*

Theorem 2.3 (Arıkan [6]). *For any B-DMC W and any fixed $R < I(W)$, block error probability for polar codes under SC decoding satisfies*

$$P_e(N, R) = O(N^{-\frac{1}{4}}). \tag{2.16}$$

Theorem 2.3 together with the bound Equation 2.15 yields to a stronger version of this theorem:

Theorem 2.4 (Arıkan [6]). *For any B-DMC W and any fixed $R < I(W)$, considering any G_N-coset codes $(N, K, \mathcal{A}, u_{\mathcal{A}^c})$ with $N \to \infty$, $K = \lfloor NR \rfloor$, $\mathcal{A}$ is chosen according to the polar coding rule for W, and $u_{\mathcal{A}^c}$ is fixed arbitrarily, the block error probability under SC decoding satisfies*

$$P_e(N, K, \mathcal{A}, u_{\mathcal{A}^c}) = O(N^{-\frac{1}{4}}). \tag{2.17}$$

2.3.6. Complexity

For the class of G_N-coset codes, the complexity of encoding and the complexity of successive cancellation decoding are both $O(N \log N)$ as functions of code blocklength N, independent of the code rate.
Compared to LDPC codes, the polar code has a less encoding complexity - remember the encoding complexity of LDPC codes was $O(N^2)$. The best decoder performance has the complexity of $O(N \log N)$ for LDPC codes again which is equal to the SC decoding complexity of polar codes. To conclude, the general complexity of polar codes can be said to be less than LDPC codes.

2.4. Recursive Channel Transformations

Equation 2.4 and Equation 2.6 together show the transition probabilities for channel combining and channel splitting sections respectively. After a quick look, it can be seen that there are N independent copies of W channels are transformed into $W_N^{(1)}, ..., W_N^{(N)}$ channels. As the number of channels N increases, calculating these transition probabilities become impossible in those forms. Therefore, it is necessary to make a proper modification so that the block-wise channel transformation can be represented as single-step channel transformations in a recursive structure.
The main idea is first to provide a single-step transformation as $(W, W) \to (W_2^{(1)}, W_2^{(2)})$ and put this in a recursive form in order to extend the block size to N. For any given B-DMC W,

$$\begin{aligned} W_2^{(1)}(y_1^2|u_1) &\triangleq \sum_{u_2} \frac{1}{2} W_2(y_1^2|u_1^2) \\ &= \sum_{u_2} \frac{1}{2} W(y_1|u_1 \oplus u_2) W(y_2|u_2), \end{aligned} \tag{2.18}$$

$$\begin{aligned} W_2^{(2)}(y_1^2, u_1|u_2) &\triangleq \frac{1}{2}W_2(y_1^2|u_1^2) \\ &= \frac{1}{2}W(y_1|u_1 \oplus u_2)W(y_2|u_2). \end{aligned} \tag{2.19}$$

Then it turns out that $W_N^{(i)}, W_N^{(i)} \rightarrow (W_{2N}^{(2i-1)}, W_{2N}^{(2i)})$ can be written. Therefore, the following corollary is valid:

Proposition 2.3. [Arıkan [6]]: For any $n \geq 0$, $N = 2^n$, $1 \leq i \leq N$,

$$W_{2N}^{(2i-1)}(y_1^{2N}, u_1^{2i-2}|u_{2i-1}) = \sum_{u_{2i}} \frac{1}{2}W_N^{(i)}(y_1^N, u_{1,o}^{2i-2} \oplus u_{1,e}^{2i-2}|u_{2i-1} \oplus u_{2i})W_N^{(i)}(y_{N+1}^{2N}, u_{1,e}^{2i-2}|u_{2i}) \tag{2.20}$$

and

$$W_{2N}^{(2i)}(y_1^{2N}, u_1^{2i-1}|u_{2i}) = \frac{1}{2}W_N^{(i)}(y_1^N, u_{1,o}^{2i-2} \oplus u_{1,e}^{2i-2}|u_{2i-1} \oplus u_{2i})W_N^{(i)}(y_{N+1}^{2N}, u_{1,e}^{2i-2}|u_{2i}). \tag{2.21}$$

Thus, the block-wise channel transformation from W_N to $W_N^{(1)}, ..., W_N^{(N)}$ can be represented as single-step channel transformations in a recursive fashion.

2.5. Polar Code Encoding

Generator matrix G_N was already defined in the previous sections as $G_N = B_N F^{\otimes n}$ where B_N is equal to Equation 2.5, and F is the actual original generator matrix. In this section, an illustration for a polar code would be more sufficient. For $N = 8$ and rate 1/2, the encoder circuit for implementing a polar code is shown in Figure 2.5. Signal flow is from left to right. Each edge carries a binary signal and each node acts as a $\mod -2$ sum operator for the incoming edges from the left and sends the result out on all edges to the right.

2.6. Polar Code Decoding

As already mentioned, the SC decoding algorithm is used. The complexity is $O(N \log N)$. For an arbitrary G_N-coset code with parameter $(N, K, \mathcal{A}, u_{\mathcal{A}^c})$, the

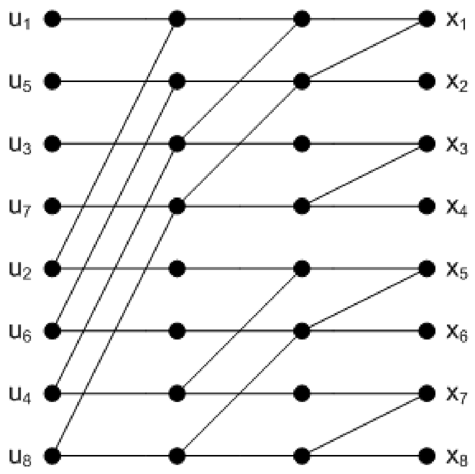

Figure 2.5. Polar code encoder circuit.

source vector u_1^N has two parts; a random part $u_{\mathcal{A}}$ and a frozen part $u_{\mathcal{A}^c}$. After this vector passes the channel, a channel output y_1^N is obtained with probability $W_N(y_1^N|u_1^N)$. The decoder observes all of the output values and also the frozen part and generates an estimate $\hat{u}_1^N$ of u_1^N. In order to understand the nature, there are N decision elements, one for each source element u_i and they are activated in the order from 1 to N. If i is a frozen index, namely if $i \in \mathcal{A}^c$, the element u_i is known, so the ith DE, therefore $\hat{u}_i = u_i$ and this result is ready to be sent to all succeeding DEs. If i is a random index, namely if $i \in \mathcal{A}$, then the ith DE has to wait for the previous decisions to be received; once $\hat{u}_1^{i-1}$ is known, it will compute the likelihood ratio (LR)

$$L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) \triangleq \frac{W_N^{(i)}(y_1^N, u_1^{i-1}|0)}{W_N^{(i)}(y_1^N, u_1^{i-1}|1)} \tag{2.22}$$

and generate its decision as

$$\hat{u}_i = \begin{cases} 0, & \text{if } L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) \geq 1 \\ 1, & \text{otherwise} \end{cases}$$

to be sent to all succeeding DEs.

Using the recursive formulas Equation 2.20 and Equation 2.21, we obtain

$$L_N^{(2i-1)}(y_1^N, \hat{u}_1^{2i-2}) = \frac{L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2}) L_{N/2}^{(i)}(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2}) + 1}{L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2}) + L_{N/2}^{(i)}(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2})} \tag{2.23}$$

and

$$L_N^{(2i)}(y_1^N|\hat{u}_1^{2i-1}) = [L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2})]^{1-2\hat{u}_{2i-1}} L_{N/2}^{(i)}(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2}). \tag{2.24}$$

Now the calculation of an LR at length N is reduced to two LR calculations at length $N/2$. The recursion continues down to block length 1 and first step LR values $L_1^{(1)}(y_i) = W(y_i|0)/W(y_i|1)$ can be computed directly.

It should be noted that the complexity for this decoder is not $O(N \log N)$ but $O(N^2)$. A more efficient implementation which has a complexity $O(N \log N)$ is possible. The previous decoder could skip the LR calculation for some frozen indexes, therefore the improvement will be as such that if $i \in \mathcal{A}^c$, the decision $\hat{u}_i$ is set to the known frozen value u_i, regardless of the LR value. The computational savings come from LR values as follows: each LR value in the pair $(L_N^{(2i-1)}(y_1^N, \hat{u}_1^{2i-2}), L_N^{(2i)}(y_1^N|\hat{u}_1^{2i-1}))$ is assembled from the same pair of LRs: $(L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2}), L_{N/2}^{(i)}(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2}))$. Therefore, to calculate all N LRs at length N, we need N LR calculations at length $N/2$. If this process continues, each class with length $N/2$ generates a set of $N/2$ LR calculations at length $N/4$ sums up to total N requests. With this reasoning, considering all the check-lengths $\{N, N/2, ..., 1\}$, the total number of LRs that will be calculated is $N(1 + \log N)$. Hence, the complexity is $O(N \log N)$.

Considering a G_N-coset code with parameter $(N, K, \mathcal{A}, u_{\mathcal{A}^c}) = (8, 5, \{3, 5, 6, 7, 8\}, (0, 0, 0))$, SC decoder is shown in Figure 2.6. There are $N(1 + \log N) = 32$ nodes in the graph, where each node computes an LR request. The rightmost column represents the channel level, whereas the leftmost column represents the decision level. There are two labels at each node, one represents the LR value to be calculated and the other indicates the number in which order that node to be activated. For instance, the second node of the second column is the $22th$ node to be activated and the LR value to

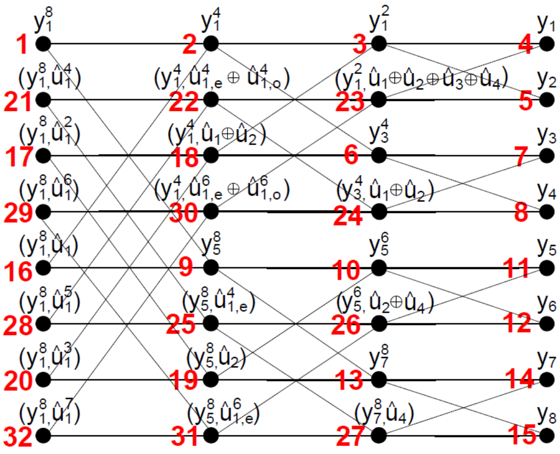

Figure 2.6. SC decoder for polar coding at $N = 8$.

be calculated at this node is $L_8^{(3)}(y_1^4, \hat{u}_{1,e}^4 \oplus \hat{u}_{1,o}^4)$. The leftmost column is the decision level and its positioning of the DEs is in the bit-reversed index order as in Figure 2.5. Decoding starts from the left-side, begins with DE 1 activating node 1 for the calculation of $L_8^{(1)}(y_1^8)$. Node 1 activates node 2 and node 2 activates node 3 which activated node 4. Since this is the channel level, $L_1^{(1)}(y_1)$ is computed at this node. Then this value is passed to nodes 3 and 23, namely to node 4's left-side neighbors. Node 3 also receives the calculated LR value from node 5 and is now ready to be computed, after it being computed, it sends its calculated LR value to nodes 2 and 18. Also LR value at node 9 is computed in a similar way and node 1 now has the required LR values from its right-side neighbors, namely from node 2 and node 9. Now $L_8^{(1)}(y_1^8)$ is calculated but then DE 1 ignores the received LR and declares $\hat{u}_1 = 0$ since u_1 is a frozen node. Then the control is passed to DE 2 and it activates node 16, the LR values to compute $L_8^{(2)}(y_1^8, \hat{u}_1)$ is ready, hence this value is computed, however since u_2 is frozen, DE 2 ignores the computed LR value and declares $\hat{u}_2 = 0$. Continuing like this, DE 3 activates node 17 and this node requests LR values from the nodes 18 and 19, after receiving these two values, the decision $\hat{u}_3$ is made in accordance with $L_8^{(3)}(y_1^8, \hat{u}_1^2)$ and DE 4 takes the control afterwards. The algorithm goes on in this manner and all 8 decision elements are formed in sequence.

3. GENERALIZED GENERATOR MATRICES

In [5], it is stated that polarization is a general phenomenon and does not require a special $F^{\otimes n}$ transformation. In [16], this claim is partially verified and transformations of $l \times l$ for $(l \geq 3)$ generator matrices are studied, and necessary and sufficient conditions for polarization are obtained. In [10], it is showed that when using polar coding and SC decoding for the original F matrix, block error probability is $O(2^{-2^{n\beta}})$, where $\beta < \frac{1}{2}$ and 2^n is the code length. Therefore it is said that the error exponent of F is $\frac{1}{2}$. In [16], it is shown that this exponent can be improved as the generator matrix dimension increases. Ideally, this exponent can approach 1 when larger generator matrices are used. This shows that, it is viable to obtain reliable communication using the polar coding technique with low block probability of error when larger generator matrices are used. In general, especially for large values of n, using larger generator matrices allows us to consider polarizations that are not spanned by $F^{\otimes n}$. Hence, the main idea is to see the polarization performances of generalized generator matrices, namely the matrices with the larger size than Arıkan's original generator matrix F.

In order to have an understanding on polarization behavior of these matrices, Bhattacharyya parameters can be used since they are directly inversely proportional to the channel capacity values. Using the Bhattacharyya parameter values, polarization performance of any generator matrix can be seen at any recursion step. Because of this reason, the first section consists of the computation of these parameters and it contains an important role on comparing different generator matrices with the same size on their finite-length polarization behaviors.

In the second section, polarization rate exponents are computed. Using the partial distances between the rows, polarization rate exponent for each generator matrix can be computed. In the asymptotical case, generator matrices having larger polarization rate exponents show a better polarization effect.

Even though seeing the Bhattacharyya parameter distributions on histograms is a useful tool for visually observing the individual polarization behavior, it is not quite helpful to compare the polarization effects of different generator matrices with the same size. Therefore, a normalized polarization distance measure is defined in the third section. Using this defined measure, it is possible to plot the polarization curves for all generator matrices of size 3 × 3 and 4 × 4.

In the next section, the upper bounds on block error probabilities are computed for the generator matrices of size 3 × 3 and 4 × 4 using Arıkan's Theorem 2.4.

In the last section, a SC decoder is implemented. The transition probability and the likelihood ratio computations for each generator matrix must be computed separately; therefore, a 4 × 4 generator matrix has been chosen which is a member of the group showing the best polarization effect according to normalized polarization distance measures and the transition probability and the likelihood ratio equations are computed for this specific matrix. The step-by-step SC decoding algorithm is explained briefly and decoding results are given at the end.

3.1. Bhattacharyya Parameters

For BEC, Bhattacharyya parameters can be directly used to obtain channel capacity values. There are two reasons on using these parameters to show the capacity performance; firstly, they are easily computable and secondly, these parameters are inversely proportional to the actual capacity value. For the F matrix, Bhattacharyya parameters were computed in Equation 2.12 and recursive channel capacity equations were obtained in Equation 2.7 in order to demonstrate the polarization effect. However, for larger generator matrices, computing these values is more complicated and obtaining similar recursive equations is not always viable.

To illustrate, for the generator matrix

$$G_3 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix},$$

the recursive combination of three independent copies of a single channel has the transition probability given by

$$W_3(y_1^3|u_1^3) = W(y_1|u_1 \oplus u_2)W(y_2|u_2 \oplus u_3)W(y_3|u_3). \tag{3.1}$$

Then, the channels are split using Equation 2.6 as follows

$$W_3^{(1)}(y_1^3|u_1) = \sum_{u_2^3} \frac{1}{4} W_3(y_1^3|u_1^3), \tag{3.2}$$

$$W_3^{(2)}(y_1^3, u_1|u_2) = \sum_{u_3} \frac{1}{4} W_3(y_1^3|u_1^3), \tag{3.3}$$

$$W_3^{(3)}(y_1^3, u_1^2|u_3) = \frac{1}{4} W_3(y_1^3|u_1^3). \tag{3.4}$$

Putting Equation 3.1) in Equation 3.2, Equation 3.3 and Equation 3.4, the Bhattacharyya parameters can be computed using the following equations

$$Z(W_3^{(1)}) = \sum_{y_1^3} \sqrt{W_3^{(1)}(y_1^3|0)W_3^{(1)}(y_1^3|1)} \tag{3.5}$$

$$Z(W_3^{(2)}) = \sum_{y_1^3} \sum_{u_1} \sqrt{W_3^{(2)}(y_1^3, u_1|0)W_3^{(2)}(y_1^3, u_1|1)}, \tag{3.6}$$

$$Z(W_3^{(3)}) = \sum_{y_1^3} \sum_{u_1^2} \sqrt{W_3^{(3)}(y_1^3, u_1^2|0)W_3^{(3)}(y_1^3, u_1^2|1)}. \tag{3.7}$$

To illustrate the computational complexity, computing Equation 3.7 is as follows

$$
\begin{aligned}
Z(W_3^{(3)}) &= \sum_{y_1^3}\sum_{u_1^2}\sqrt{W_3^{(3)}(y_1^3,u_1^2|0)W_3^{(3)}(y_1^3,u_1^2|1)} \\
&= \sum_{y_1^3,u_1^2}\frac{1}{2}\sqrt{W(y_1|u_1\oplus u_2)W(y_2|u_2)W(y_3|0)}\sqrt{W(y_1|u_1\oplus u_2)W(y_2|u_2\oplus 1)W(y_3|1)} \\
&= \sum_{y_1^3}\frac{1}{4}(\sqrt{W(y_1|0)W(y_2|0)W(y_3|0)}\sqrt{W(y_1|0)W(y_2|1)W(y_3|1)}+ \\
&\sqrt{W(y_1|1)W(y_2|1)W(y_3|0)}\sqrt{W(y_1|1)W(y_2|0)W(y_3|1)}+ \\
&\sqrt{W(y_1|1)W(y_2|0)W(y_3|0)}\sqrt{W(y_1|1)W(y_2|1)W(y_3|1)}+ \\
&\sqrt{W(y_1|0)W(y_2|1)W(y_3|0)}\sqrt{W(y_1|0)W(y_2|0)W(y_3|1)}). \\
&= \sum_{y_1}\frac{1}{2}(\epsilon^2 W(y_1|0)+\epsilon^2 W(y_1|1)) = \frac{\epsilon^2(1-\epsilon)}{2}+\frac{\epsilon^2(1-\epsilon)}{2}+\epsilon^3=\epsilon^2. \qquad (3.8)
\end{aligned}
$$

Note that transition probabilities $W(0|0)=W(1|1)=1-\epsilon$, $W(0|1)=W(1|0)=0$ and $W(e|0)=W(e|1)=\epsilon$, where e denotes an erasure. Similarly, it can be found that

$$Z(W_3^{(2)}) = -\epsilon^3+2\epsilon^2, \qquad (3.9)$$

and

$$Z(W_3^{(1)}) = \epsilon^3-3\epsilon^2+3\epsilon. \qquad (3.10)$$

For $i=1,...,N$, the equations can be written in a recursive form similar to Equation 2.7 again using *Proposition 2.1.* and *Proposition 2.2.* such that

$$I(W_N^{(3i-2)}) = I(W_{N/3}^{(i)})^2, \qquad (3.11)$$

$$I(W_N^{(3i-1)}) = -I(W_{N/3}^{(i)})^3+I(W_{N/3}^{(i)})^2, \qquad (3.12)$$

$$I(W_N^{(3i)}) = I(W_{N/3}^{(i)})^3 - 3I(W_{N/3}^{(i)})^2 + 3I(W_{N/3}^{(i)}). \tag{3.13}$$

Although calculating these values analytically for larger matrices is quite difficult, these parameters can be numerically calculated via a computer. A structure has been implemented to compute these parameters recursively.

For all possible lower-triangular 3 × 3 matrices defined as

$$G_{abc} = \begin{bmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ b & c & 1 \end{bmatrix},$$

the polarization histograms corresponding to all possible lower-triangular 3 × 3 generator matrices for 7 recursions, namely $N = 3^7 = 2187$ channels over binary erasure channels with $\epsilon = 0.5$ is shown in Figure 3.1.

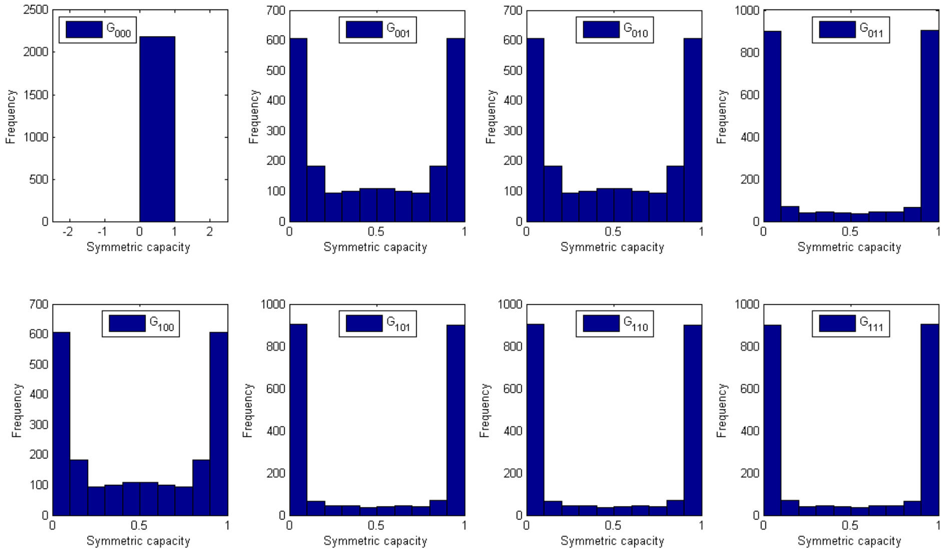

Figure 3.1. Polarization histograms for all lower triangular 3 × 3 generator matrices with $\epsilon = 0.5$.

The generator matrices showing the best polarization performance are G_{011}, G_{101}, G_{110}, and G_{111}. Among the remaining matrices, G_{001}, G_{010}, and G_{100} demonstrate a weaker polarization effect, and G_{000} is not polarized at all. Note that this last observation for G_{000} is expected, since this generator matrix corresponds to using three channels independently without any processing of the channel inputs (identity matrix of size 3×3).

At this point, it should be noted that computing the Bhattacharyya parameters only for lower-triangular matrices is enough because there is a symmetric relation between the matrices and parameter values for upper-triangular matrices and lower-triangular matrices and there occurs a polarization performance level-based cluster of groups as can also be seen only in lower-triangular case. This kind of grouping will also be seen in the 4×4 case as well.

The evolution of the polarization tree for G_{101} is given in Figure 3.2 for up to 7 iterations. Although this plot successfully demonstrates the creation of new channels with different capacities at different recursions, the overlapping lines corresponding to the $N = 3^7 = 2187$ polarized channels at recursion 7 make it harder to observe the polarization effect. In order to solve this problem, the polarization histogram evolution is replotted in Figure 3.3. In this plot, it can be seen that while the entire histogram consists of channels with erasure values equal to $\epsilon = 0.5$ for the first recursion (the gray color), the subsequent iterations result in channels with colors polarizing towards white ($\epsilon = 1$) and black ($\epsilon = 0$).

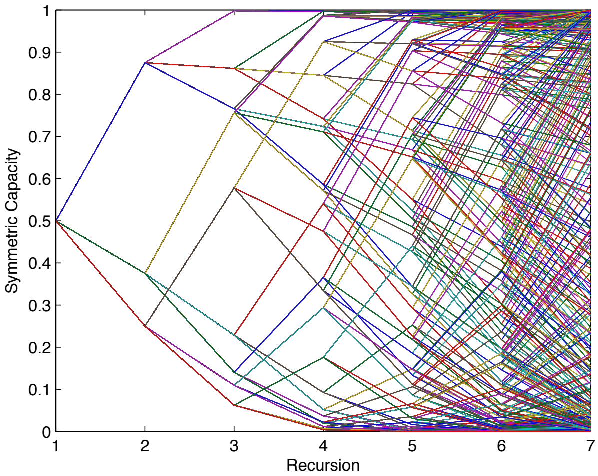

Figure 3.2. Evolution of the polarization tree for G_{101} with $\epsilon = 0.5$.

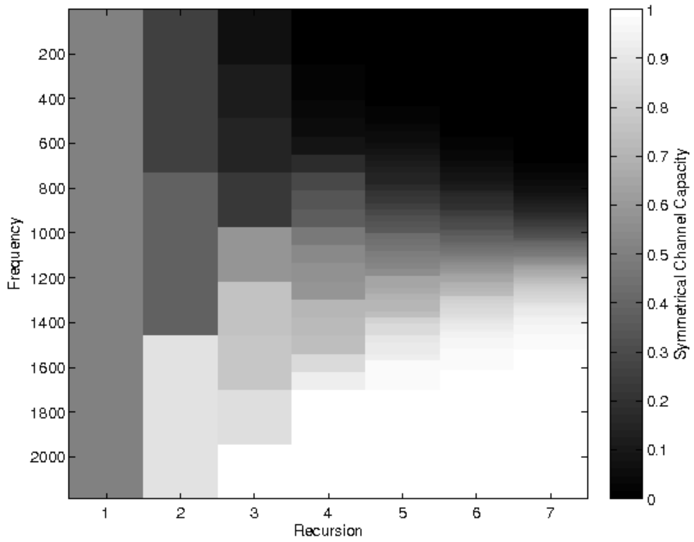

Figure 3.3. Evolution of the polarization histogram for G_{101} with $\epsilon = 0.5$.

3.2. Polarization Rate Exponents

In [17], it has been shown that the Hamming distances between generator matrix rows can be used as a polarization performance measure for block error probability. Given an $l \times l$ matrix $G = [g_1^T, ..., g_l^T]^T$, the partial distance D_i is defined as

$$D_i \triangleq d_H(g_i, < g_{i+1}, ..., g_l >), \qquad i = 1, ..., l-1, \tag{3.14}$$

$$D_l \triangleq d_H(g_l, 0). \tag{3.15}$$

To obtain the polarization rate exponent from partial distances, the relation is as follows. For any symmetric B-DMC W and any $l \times l$ matrix G with partial distances $\{D_i\}_{i=1}^{l}$, the rate of polarization $E(G)$ is given by

$$E(G) = \frac{1}{l} \sum_{i=1}^{l} log_l D_i. \tag{3.16}$$

Now, let us recall the generator matrix

$$G_{101} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}.$$

According to Equation 3.14, the partial distances for this matrix are $D_1 = 1$, $D_2 = 2$ and $D_3 = 2$. Using Equation 3.16, the polarization rate exponent for the generator matrix G_{101} can then be calculated as $E(G_{101}) = \frac{1}{l}\sum_{i=1}^{l} log_l D_i = \frac{2}{3} log_3(2) = 0.4206$. The polarization rate exponents for all possible lower-triangular 3 × 3 matrices are shown in Table 3.1. This result shows that the generator matrices showing the same amount of polarization (according to histogram plots) may have different polarization rate exponents. Therefore, more than one parameter should be taken into account for the performance evaluation and comparison of polar coded systems for finite-length systems. In order to create the most suitable, fastest, and most efficient finite-length polar code, only checking the Hamming distances between generator matrix rows is not enough. However, the generator matrix with larger polarization rate exponent gives a better polarized system in the asymptotical case.

Table 3.1. Polarization Rate Exponents.

Matrix	E_G	Matrix	E_G
$E(G_{000})$	0.000	$E(G_{100})$	0.210
$E(G_{001})$	0.210	$E(G_{101})$	0.421
$E(G_{010})$	0.210	$E(G_{110})$	0.421
$E(G_{011})$	0.333	$E(G_{111})$	0.333

3.3. Normalized Polarization Distance Measure

Plotting the polarization histograms and seeing the Bhattacharyya parameter distributions for any generator matrix is a useful tool for visually observing the polarization behavior, however, it is not quite helpful in comparing the polarization effects of different generator matrices with the same size. In order to come up with a solution to this, we define a normalized polarization distance measure $d_p^{\epsilon_0}(\bar{\epsilon}, N)$ for an initial channel erasure probability ϵ_0 and a vector $\bar{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_N)$ consisting of N polarized channel erasure probabilities using the relation

$$d_p^{\epsilon_0}(\bar{\epsilon}, N) = \frac{1}{N\epsilon_0^2} \sum_{i=1}^{N} \min(|\epsilon_i|, |1-\epsilon_i|)^2. \tag{3.17}$$

The normalization guarantees that the measure values satisfy

$$0 \leq d_p^{\epsilon_0}(\bar{\epsilon}, N) \leq 1$$

for all values of N, ϵ_0, and $\bar{\epsilon}$. Therefore, the generator matrices showing no polarization effect will have $d_p^{\epsilon_0}(\bar{\epsilon}, N) = 1$. As the normalized polarization distance measure goes to 0, the generator matrix will show a better polarization performance.

Using Equation 3.17, the normalized polarization distance measures of all lower-triangular 3×3 generator matrices are plotted in Figure 3.4. The measure curves corresponding to generator matrices yielding better polarization effects fall more rapidly than others. At the extreme case of no polarization (G_{000}), the measure remains con-

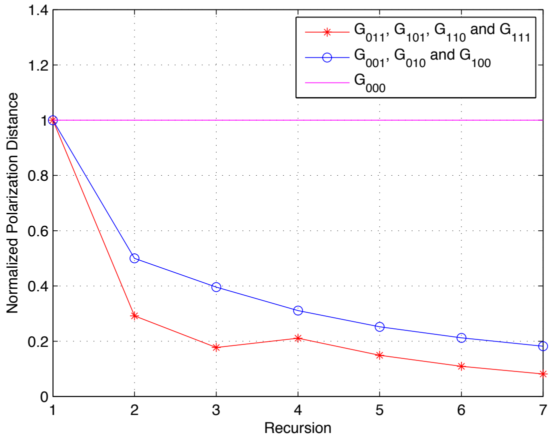

Figure 3.4. Normalized polarization distance measures for all lower-triangular 3×3 generator matrices with $\epsilon = 0.5$.

stant at the normalized 1 value, which is the polarization-based worst case scenario. Again using Equation 3.17, the normalized polarization distance measures for all 4×4 generator matrices are plotted in Figure 3.5. As already mentioned in the earlier chapter, there is also a clustering behavior among $2^{16} = 65536$ generator matrices of size 4×4. There are only 11 groups of matrices showing different amount of polarization. Hence, there is only 11 different normalized polarization distance curves. At this point, we should note that only 18624 matrices show polarization effect among 65536 possible 4×4 generator matrices and only 192 matrices are in Group 1.

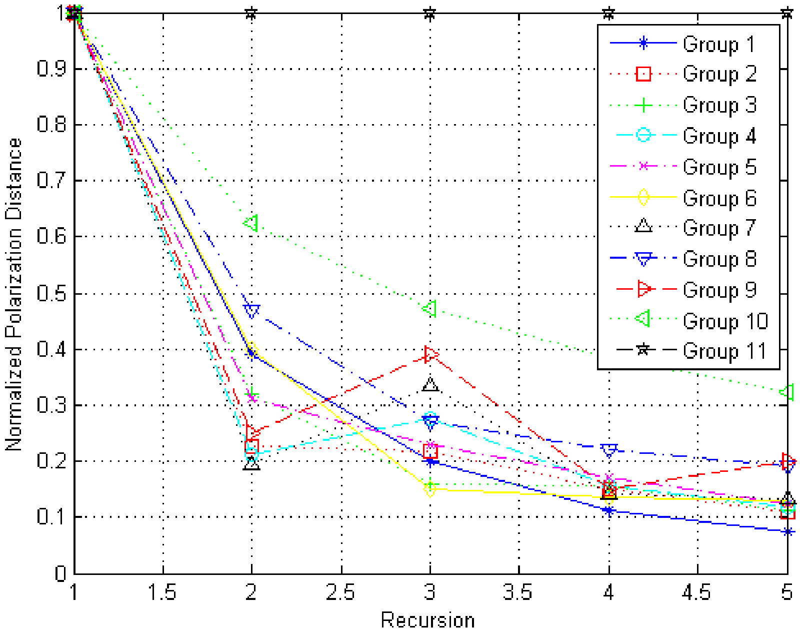

Figure 3.5. Normalized polarization distance measures for all 4×4 generator matrices with $\epsilon = 0.5$.

3.4. Upper Bound on Block Error Probability

Using Theorem 2.4, upper bounds on block error probability for all 3×3 lower-triangular matrices with $\epsilon = 0.5$ are shown in Figure 3.6. The original upper bound for Arıkan's 2×2 code is computed at block length $N = 2^{11} = 2048$, however the bound for 3×3 generator matrices are computed at block length $N = 3^7 = 2187$.

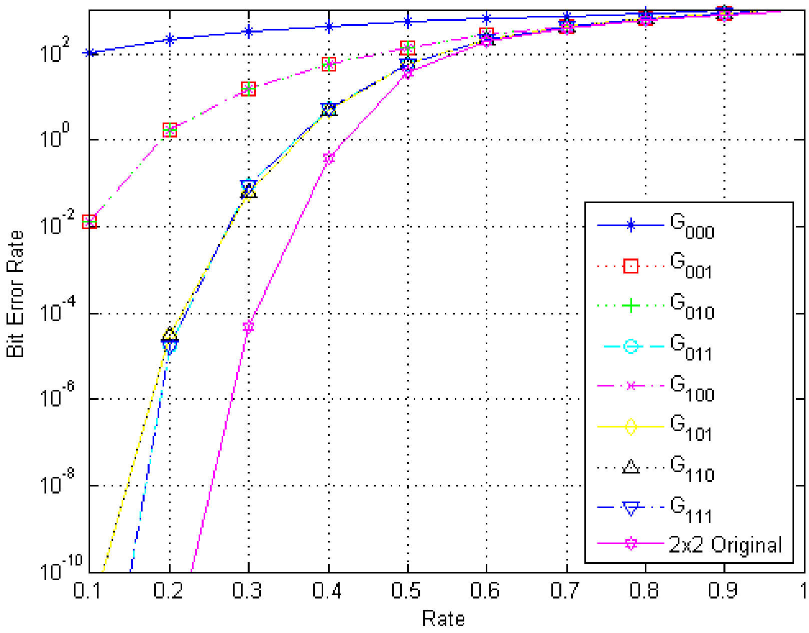

Figure 3.6. Upper bound on block error probability for all 3×3 lower-triangular matrices with $\epsilon = 0.5$.

Therefore, the comparison is not quite possible between the 2×2 and 3×3 generator matrices, but it is viable to say that the matrices showing better polarization effect also has a more strict bound on block error probability. Also note that generator matrices showing same characteristics on polarization histograms also have the same upper bounds on block error.

Now, again using Theorem 2.4, upper bounds on block error probability for all 4×4 generator matrices with $\epsilon = 0.5$ are shown in Figure 3.7. The original upper bound for Arıkan's 2×2 code is computed at block length $N = 2^{10} = 1024$, and the bound for 4×4 generator matrices are computed at block length $N = 4^{5} = 1024$. Recalling that Group 1 was the group showing the best polarization performance, its upper bound on block error probability curve intersects with Arıkan's curve. Therefore, using a matrix from Group 1 and using it on channel combining and splitting sections is expected to show a good block error performance. This is the main reason why a generator matrix from Group 1 is chosen for the investigation of decoding performance and on the implementation of a decoder at the next section.

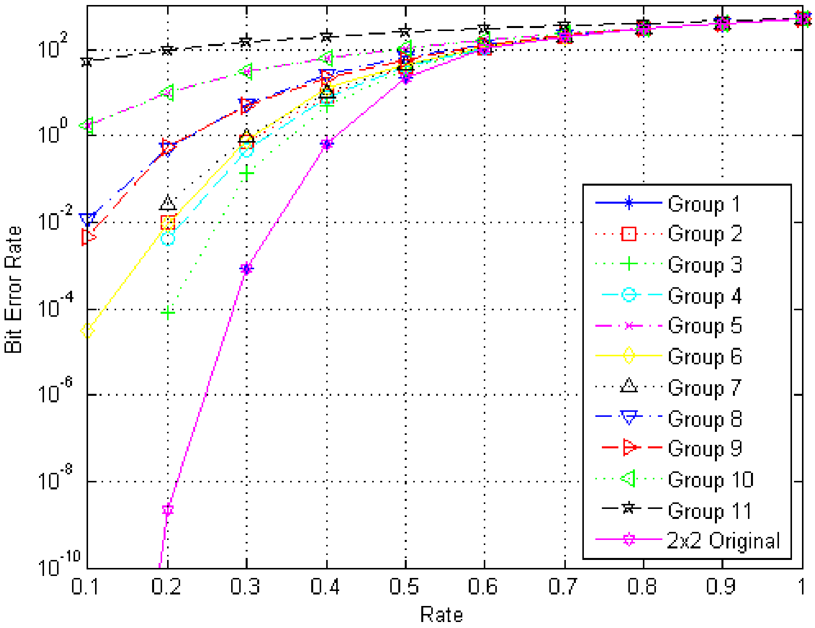

Figure 3.7. Upper bound on block error probability for all 4 × 4 matrices with $\epsilon = 0.5$.

3.5. Decoder

For any generator matrix, the likelihood ratio equations can be computed via Equation 2.22. However, direct computation of $W_N^{(i)}$ values at large block-lengths is impossible. For instance, for $i = 1$, while computing $W_N^{(1)}$ with using Equation 2.6, the summation starts from u_2 and ends at u_N. That is to say, all possible binary combinations should be computed for 2^{N-i} values (namely for the first step, from u_2 to u_N, 2^{N-1} values should be computed and so on). Therefore, it is necessary to compute almost infinite number of parameters and a recursive structure is necessary for the decoding algorithm as in Equation 2.23 and Equation 2.24 for each generator matrix.

To illustrate, a generator matrix from the best group according to normalized polarization distance measure is chosen (a matrix from the Group 1).

$$G_e = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Then the channel after channel combining phase has the transition probability as follows

$$W_4(y_1^4|u_1^4) = W(y_1|u_1 \oplus u_2 \oplus u_4)W(y_2|u_3 \oplus u_4)W(y_3|u_4)W(y_4|u_2 \oplus u_3 \oplus u_4). \quad (3.18)$$

Having obtained the combined channel transition probability for the specified matrix, channels must be split back into a set of N binary-input coordinate channels $W_N^{(i)}$ using Equation 2.6. Then the transition probabilities for the split channels can be computed as follows

$$W_4^{(1)}(f(y_1^4)|u_1) = \sum_{u_2',u_3',u_4'} \frac{1}{8} W(y_1|u_1 \oplus u_2' \oplus u_4')W(y_2|u_3' \oplus u_4')W(y_3|u_4')W(y_4|u_2' \oplus u_3' \oplus u_4'), \quad (3.19)$$

$$W_4^{(2)}(f(y_1^4), u_1|u_2) = \sum_{u_3',u_4'} \frac{1}{8} W(y_1|u_1 \oplus u_2 \oplus u_4')W(y_2|u_3' \oplus u_4')W(y_3|u_4')W(y_4|u_2 \oplus u_3' \oplus u_4'), \quad (3.20)$$

$$W_4^{(3)}(f(y_1^4), u_1, u_2|u_3) = \sum_{u_4'} \frac{1}{8} W(y_1|u_1 \oplus u_2 \oplus u_4')W(y_2|u_3 \oplus u_4')W(y_3|u_4')W(y_4|u_2 \oplus u_3 \oplus u_4'), \quad (3.21)$$

$$W_4^{(4)}(f(y_1^4), u_1, u_2, u_3|u_4) = \frac{1}{8} W(y_1|u_1 \oplus u_2 \oplus u_4)W(y_2|u_3 \oplus u_4)W(y_3|u_4)W(y_4|u_2 \oplus u_3 \oplus u_4). \quad (3.22)$$

Now, it is time to generalize these individual channel transition probabilities into a more general form. In other words, we need to obtain the recursive channel transformations for our generator matrix G_e similar to the forms Equation 2.20 and Equation 2.21 which were the valid computations for Arıkan's original 2 × 2 matrix. Therefore, following the similar procedures, we now have 4 groups rather than having only *even* and *odd* indices and we will name them by $G1$, $G2$, $G3$ and $G4$ subscripts, each representing a group of $N/4$ elements: $G1 = \{1, 5, 9, 13, ...\}$, $G2 = \{2, 6, 10, 14, ...\}$, $G3 = \{3, 7, 11, 15, ...\}$ and $G4 = \{4, 8, 12, 16, ...\}$. Then the recursive transition probabilities are in the following form

$$\begin{aligned} W_{4N}^{(4i-3)}(y_1^{4N}, u_1^{4i-4}|u_{4i-3}) = \sum_{u_{4i-2}^{4i}} \frac{1}{8} & W_N^{(i)}(y_1^N, u_{G1}^{4i-4} \oplus u_{G2}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-3} \oplus u_{4i-2} \oplus u_{4i}) \\ & W_N^{(i)}(y_{N+1}^{2N}, u_{G3}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-1} \oplus u_{4i}) W_N^{(i)}(y_{2N+1}^{3N}, u_{G4}^{4i-4}|u_{4i}) \\ & W_N^{(i)}(y_{3N+1}^{4N}, u_{G2}^{4i-4} \oplus u_{G3}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-2} \oplus u_{4i-1} \oplus u_{4i}) \end{aligned} \tag{3.23}$$

$$\begin{aligned} W_{4N}^{(4i-2)}(y_1^{4N}, u_1^{4i-4}|u_{4i-3}) = \sum_{u_{4i-1}^{4i}} \frac{1}{8} & W_N^{(i)}(y_1^N, u_{G1}^{4i-4} \oplus u_{G2}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-3} \oplus u_{4i-2} \oplus u_{4i}) \\ & W_N^{(i)}(y_{N+1}^{2N}, u_{G3}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-1} \oplus u_{4i}) W_N^{(i)}(y_{2N+1}^{3N}, u_{G4}^{4i-4}|u_{4i}) \\ & W_N^{(i)}(y_{3N+1}^{4N}, u_{G2}^{4i-4} \oplus u_{G3}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-2} \oplus u_{4i-1} \oplus u_{4i}) \end{aligned} \tag{3.24}$$

$$\begin{aligned} W_{4N}^{(4i-1)}(y_1^{4N}, u_1^{4i-4}|u_{4i-3}) = \sum_{u_{4i}} \frac{1}{8} & W_N^{(i)}(y_1^N, u_{G1}^{4i-4} \oplus u_{G2}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-3} \oplus u_{4i-2} \oplus u_{4i}) \\ & W_N^{(i)}(y_{N+1}^{2N}, u_{G3}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-1} \oplus u_{4i}) W_N^{(i)}(y_{2N+1}^{3N}, u_{G4}^{4i-4}|u_{4i}) \\ & W_N^{(i)}(y_{3N+1}^{4N}, u_{G2}^{4i-4} \oplus u_{G3}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-2} \oplus u_{4i-1} \oplus u_{4i}) \end{aligned} \tag{3.25}$$

$$\begin{aligned} W_{4N}^{(4i)}(y_1^{4N}, u_1^{4i-4}|u_{4i-3}) = \frac{1}{8} & W_N^{(i)}(y_1^N, u_{G1}^{4i-4} \oplus u_{G2}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-3} \oplus u_{4i-2} \oplus u_{4i}) \\ & W_N^{(i)}(y_{N+1}^{2N}, u_{G3}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-1} \oplus u_{4i}) W_N^{(i)}(y_{2N+1}^{3N}, u_{G4}^{4i-4}|u_{4i}) \\ & W_N^{(i)}(y_{3N+1}^{4N}, u_{G2}^{4i-4} \oplus u_{G3}^{4i-4} \oplus u_{G4}^{4i-4}|u_{4i-2} \oplus u_{4i-1} \oplus u_{4i}) \end{aligned} \tag{3.26}$$

Then, the likelihood ratio equations for the generator matrix G_e can be computed via Equation 2.22. Since abbreviations will be used on defining the likelihood ratios, first the tree structure of the decoder must be shown. As already explained in the Section 2.6, there is a tree structure similar to the one in Figure 2.6; however, this time since the generator matrix size is 4×4, the simplified structure will look as in Figure 3.8. Even though there are only 2 recursion steps and only the connections of the first group at the third level were made, it is still not easy to investigate the decoder graph; this is the actual reason of the fact that the other connections at the third level is not shown. Now let us take the channel level and the middle level nodes of the first group. LRs at the channel level are easily computable and computations can be made identical to the process as already defined in Section 2.6. Using Equation 2.22 and the recursive channel transition probabilities Equation 3.23, Equation 3.24, Equation 3.25 and Equation 3.26; the LRs are in the following forms

$$LL_1 = \frac{L_1L_2L_3L_4 + L_1L_3 + L_2L_4 + L_1L_4 + L_2L_3 + L_1L_2 + L_3L_4 + 1}{L_1L_2L_3 + L_1L_2L_4 + L_1L_3L_4 + L_2L_3L_4 + L_1 + L_2 + L_3 + L_4},$$

$$LL_2 = \begin{cases} \frac{L_1L_2L_3L_4+L_1L_3+L_2L_4+1}{L1_L2+L_3L_4+L_1L_4+L_2L_3}, & \text{if } \hat{u}_{4i-3} = 0, \\ \frac{L_2L_3L_4+L_1L_2L_4+L_1+L_3}{L_1L_2L_3+L_1L_3L_4+L_2+L_4}, & \text{if } \hat{u}_{4i-3} = 1, \end{cases}$$

$$LL_3 = \begin{cases} \frac{L_1L_2L_3L_4+1}{L_1L_3+L_2L_4}, & \text{if } \hat{u}_{4i-3} = 0 \quad \text{and} \quad \hat{u}_{4i-2} = 0, \\ \frac{L_1L_4+L_2L_3}{L_1L_2+L_3L_4}, & \text{if } \hat{u}_{4i-3} = 0 \quad \text{and} \quad \hat{u}_{4i-2} = 1, \\ \frac{L_1+L_2L_3L_4}{L_1L_2L_4+L_3}, & \text{if } \hat{u}_{4i-3} = 1 \quad \text{and} \quad \hat{u}_{4i-2} = 0, \\ \frac{L_1L_2L_3+L_4}{L_1L_3L_4+L_2}, & \text{if } \hat{u}_{4i-3} = 1 \quad \text{and} \quad \hat{u}_{4i-2} = 1, \end{cases}$$

$$LL_4 = [L_1]^{1-2(\hat{u}_{4i-3} \oplus \hat{u}_{4i-2})}[L_2]^{1-2(\hat{u}_{4i-1})}L_3[L_4]^{1-2(\hat{u}_{4i-2} \oplus \hat{u}_{4i-1})}$$

For the group containing nodes $1, 2, 3, 4, 17, 18, 19$ and 20: L_1 is computed at Node 1, L_2 is computed at Node 2, L_3 is computed at Node 3 and L_4 is computed at Node

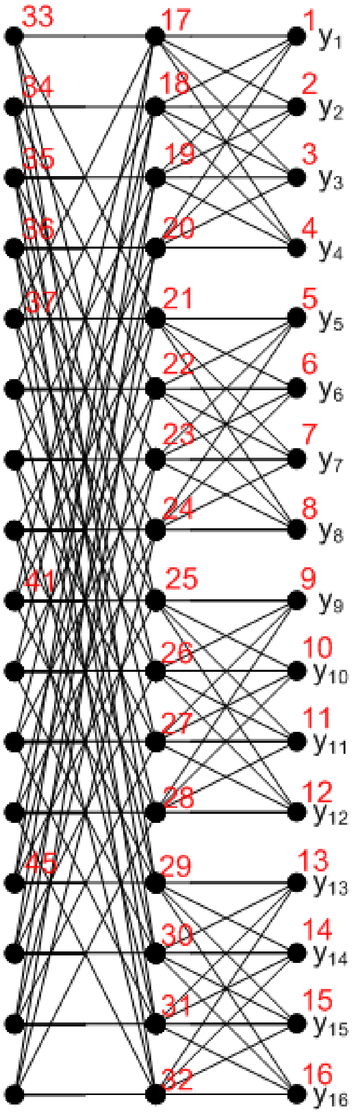

Figure 3.8. SC decoder for polar coding with a 4 × 4 generator matrix at $N = 16$.

4 at the channel level. Then LL_1 computation refers to the LR value of Node 17, LL_2 to the LR value of Node 18, LL_3 to LR value of Node 19 and LL_4 to the LR value of Node 20. At the second level, grouping is important and while computing the LR value at Node 33, LL_1 equation will be used and L_1, L_2, L_3 and L_4 values refer to the LR values of Nodes 17, 21, 25 and 29 respectively. At this level, using the same nodes at the second level, LL_2 equation will give the LR value at Node 37, LL_3 will give the LR value at Node 41 and LL_4 gives the LR value at Node 45 all dependent on the previous decisions. The whole computation process will continue like this recursively as the decoder reaches to the decision level.

The bit error rate (BER) and frame error rate (FER) performance curves for the SC decoder for G_e are shown in Figure 3.9. Block length for G_e is $4^5 = 1024$ and block length for Arıkan's original generator matrix is $2^{10} = 1024$. The error performance of the codes from Group 1 of size 4×4 generator matrices is the same as Arıkan's original generator matrix' performance. This kind of decoding performance was expectable as it was already shown that upper bound on block error probability for Group 1 generator matrices of size 4×4 intersects with Arıkan's upper bound at the same block length.

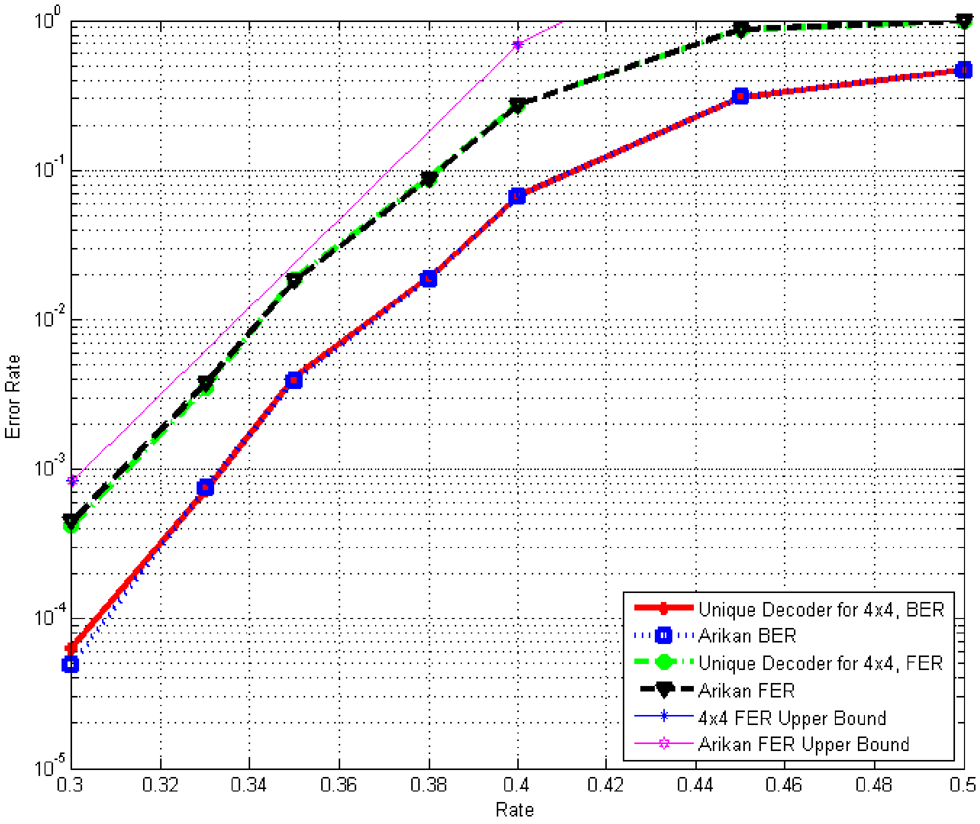

Figure 3.9. BER and FER performance curves for SC decoding on a BEC with $\epsilon = 0.5$.

4. CONCLUSION

In this thesis, polarization performance of polar codes using different generator matrices was analyzed via both asymptotical and finite-length measures to design and compare efficient polar coded systems. The Bhattacharyya parameters were computed to obtain finite-length measures and the polarization rate exponents to obtain asymptotical results. It was also shown that these measures do not always point at the same practical designs and a finite-length polarization distance measure is also proposed to analytically demonstrate this. This measure can be used for any proposed code rate. For a given specific rate, the generator matrix showing the best polarization performance can be chosen among all generator matrices using the polarization distance measure. The main idea on defining this normalized polarization distance measure was to fill the gap on the analysis of the finite-length polar code generation and obtain efficient polar codes and hence the coding structures for these generalized polar codes. Upper bound on error block probabilities for 3×3 and 4×4 generator matrices were given and it was shown that this bound matches with Arıkan's original bound for our best group of size 4×4 computed at the same block-length. Also for the SC decoder, the recursive equations for LR calculations for a specific 4×4 matrix were given. Results on decoding performance show that our transition probability and likelihood ratio equations for G_e are valid. The matrices in Group 1 are not all the Kronecker products of Arıkan's matrix and using different connections on channels can result in the same amount of polarization and the same amount of block error as well. Another aspect of our SC decoder designed for G_e is that it has only 5 recursions (steps); whereas the SC decoder has to go for 10 iterations with Arıkan's generator matrix. However, at each recursion we compute 4 LR values and Arıkan's SC decoder computes 2 and trade-off on computational complexity is obvious. On the other hand, as the number of computed edges is the same, the number of visited nodes is less in our simulations. Looking from the RAM access point of view, computation of LR values are quite straightforward and easy and the overall computation progress will be faster if one uses a fast processor. To sum up, using larger generator matrices on finite-length cases may yield to better polarization and block error performances, and is worth trying.

APPENDIX A: PERFORMANCE TOOL

```
Set the generator matrix $G$, number of iterations $n$, blocklength $[size(G)]^n = N$ and erasure probability $\epsilon$
Compute the $W_N^{(i)}$ values using Equation 2.6
**for** $iter = 1$ to $n$ **do**
    Compute the Bhattacharyya parameters $\mathbf{Z}$ using Equation 2.12
    Form the $N \times (n + 1)$ matrix $\mathbf{Z}$ as each iteration, so the each column of the matrix $\mathbf{Z}$ results in $[size(G)]^{iter}$ different Bhattacharyya parameter values as shown in Figure 3.2
**end for**
**for** $iter = 1$ to $n$ **do**
    Compute the normalized polarization distance measure at each iteration using Equation 3.17
**end for**
```

Figure A.1. Channel Polarization Computation Tool.

REFERENCES

1. Shannon, C. E., “A Mathematical Theory of Communication”, *Bell Systems Technical Journal*, Vol. 27, pp. 379–423, 1948.

2. Gallager, R. G., “Low-Density Parity-Check Codes”, *IRE Transactions on Information Theory*, Vol. IT-8, pp. 21–28, 1962.

3. Gallager, R. G., *Low-Density Parity-Check Codes*, M.I.T. Press, Cambridge, MA, 1963.

4. Chung, S. Y., G. D. Forney, Jr., T. J. Richardson and R. L. Urbanke, “On the Design of Low-Density Parity-Check Codes within 0.0045 dB of the Shannon Limit”, *IEEE Communications Letters*, Vol. 5, pp. 58–60, Feb. 2001.

5. Arıkan, E., “Channel Polarization: A Method for Constructing Capacity-Achieving Codes”, *Proc. IEEE Intl. Symposium on Inform. Theory*, pp. 1173–1177, 2008.

6. Arıkan, E., “Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels”, *IEEE Transactions on Information Theory*, Vol. IT-55, No. 7, pp. 3051–3073, 2009.

7. Arıkan, E., “A Performance Comparison of Polar Codes and Reed-Muller Codes”, *IEEE Communications Letters*, Vol. 12, No. 6, pp. 447–449, 2008.

8. Mori, R. and T. Tanaka, “Performance of Polar Codes with the Construction Using Density Evolution”, *IEEE Communications Letters*, Vol. 13, No. 7, pp. 519 –521, 2009.

9. Arikan, E., “A Survey of Reed-Muller Codes from Polar Coding Perspective”, *Proceedings of IEEE Information Theory Workshop*, pp. 1 –5, 2010.

10. Arikan, E. and E. Telatar, “On the Rate of Channel Polarization”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 1493 –1495, 2009.

11. Tanaka, T., “On Speed of Channel Polarization”, *Proceedings of IEEE Information Theory Workshop*, pp. 1 –5, 2010.

12. Tanaka, T. and R. Mori, “Refined Rate of Channel Polarization”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 889 –893, 2010.

13. Hassani, S. and R. Urbanke, “On the Scaling of Polar Codes: I. The Behavior of Polarized Channels”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 874 –878, 2010.

14. Hassani, S., K. Alishahi and R. Urbanke, “On the Scaling of Polar Codes: II. The Behavior of Un-Polarized Channels”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 879 –883, 2010.

15. Korada, S., A. Montanari, E. Telatar and R. Urbanke, “An Empirical Scaling Law for Polar Codes”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 884 –888, 2010.

16. Korada, S., E. Sasoglu and R. Urbanke, “Polar Codes: Characterization of Exponent, Bounds, and Constructions”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 1483 –1487, 2009.

17. Korada, S., E. Şaşoğlu and R. Urbanke, “Polar Codes: Characterization of Exponent, Bounds, and Constructions”, *IEEE Transactions on Information Theory*, Vol. 56, No. 12, pp. 6253 –6264, 2010.

18. Mori, R. and T. Tanaka, “Performance and Construction of Polar Codes on Symmetric Binary-Input Memoryless Channels”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 1496 –1500, 2009.

19. Pedarsani, R., S. H. Hassani, I. Tal and E. Telatar, “On the Construction of Polar Codes”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 11–15, 2011.

20. Sasoglu, E., E. Telatar and E. Arikan, “Polarization for Arbitrary Discrete Memoryless Channels”, *Proceedings of IEEE Information Theory Workshop*, pp. 144 –148, 2009.

21. Korada, S. and E. Sasoglu, “A Class of Transformations That Polarize Binary-Input Memoryless Channels”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 1478 –1482, 2009.

22. Arikan, E., “Systematic Polar Coding”, *IEEE Communications Letters*, Vol. 15, No. 8, pp. 860 –862, 2011.

23. Goela, N., S. Korada and M. Gastpar, “On LP Decoding of Polar Codes”, *Proceedings of IEEE Information Theory Workshop*, pp. 1 –5, 2010.

24. Tal, I. and A. Vardy, “List Decoding of Polar Codes”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 1–5, 2011.

25. Korada, S. and R. Urbanke, “Polar Codes Are Optimal for Lossy Source Coding”, *Proceedings of IEEE Information Theory Workshop*, pp. 149 –153, 2009.

26. Korada, S. and R. Urbanke, “Polar Codes are Optimal for Lossy Source Coding”, *IEEE Transactions on Information Theory*, Vol. 56, No. 4, pp. 1751 –1768, 2010.

27. Abbe, E., “Universal Source Polarization and Sparse Recovery”, *Proceedings of IEEE Information Theory Workshop*, pp. 1 –5, 2010.

28. Cronie, H. and S. Korada, “Lossless Source Coding with Polar Codes”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 904 –908, 2010.

29. Karzand, M. and E. Telatar, “Polar Codes for Q-ary Source Coding”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 909 –912, 2010.

30. Bakshi, M., S. Jaggi and M. Effros, “Concatenated Polar Codes”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 918 –922, 2010.

31. Eslami, A. and H. Pishro-Nik, “A Practical Approach to Polar Codes”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 16–20, 2011.

32. Andersson, M., V. Rathi, R. Thobaben, J. Kliewer and M. Skoglund, “Nested Polar Codes for Wiretap and Relay Channels”, *IEEE Communications Letters*, Vol. 14, No. 8, pp. 752 –754, 2010.

33. Mahdavifar, H. and A. Vardy, “Achieving the Secrecy Capacity of Wiretap Channels Using Polar Codes”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 913 –917, 2010.

34. Şaşoğlu, E., E. Telatar and E. Yeh, “Polar Codes for the Two-User Binary-Input Multiple-Access Channel”, *Proceedings of IEEE Information Theory Workshop*, pp. 1 –5, 2010.

35. Mori, R. and T. Tanaka, “Non-Binary Polar Codes Using Reed-Solomon Codes and Algebraic Geometry Codes”, *Proceedings of IEEE Information Theory Workshop*, pp. 1 –5, 2010.

36. Sasoglu, E., “Polarization in the Presence of Memory”, *Proceedings of IEEE International Symposium on Information Theory*, pp. 189–193, 2011.