PERFORMANCE ANALYSIS OF POLAR CODES

ALI MAJID HASAN AL-IBRAHEEMI

JANUARY 2015

PERFORMANCE ANALYSIS OF POLAR CODES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF
ÇANKAYA UNIVERSITY

BY
ALI MAJID HASAN AL-IBRAHEEMI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF
ELECTRONIC AND COMMUNICATION ENGINEERING

JANUARY 2015

Title of the Thesis: **Performance Analysis of Polar Codes**

Submitted by **Ali Majid Hasan AL-IBRAHEEMI**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University.

Prof. Dr. Taner ALTUNOK
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Halil Tanyer EYYUBOĞLU
Head of Department

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

Assoc. Prof. Dr. Orhan GAZİ
Supervisor

Examination Date: 30.01.2015

Examining Committee Members:

Assoc. Prof. Dr. Orhan GAZİ	(Çankaya Univ.)	
Assoc. Prof. Dr. Fahd JARAD	(UTAA)	
Assist. Prof. Dr. Göker ŞENER	(Çankaya Univ.)	

STATEMENT OF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name	: Ali, AL-IBRAHEEMI
Signature	:
Date	: 30.01.2015

ABSTRACT

PERFORMANCE ANALYSIS OF POLAR CODES

AL-IBRAHEEMI, Ali
M.Sc., Department of Electronic and Communication Engineering
Supervisor: Assoc. Prof. Dr. Orhan GAZİ

January 2015, 37 pages

In this thesis work we propose an algorithm for the successive cancellation decoding of the polar codes. The proposed algorithm is suitable for parallel processing operations and easy to implement in hardware. The proposed algorithm is simulated using Matlab programming and performance graphs are obtained for binary erasure channels. The proposed decoding algorithm can be applied to any other discrete memoryless channel.

Keywords: Polar Codes, Channel Polarization, Successive Cancellation Decoding.

ÖZ

KUTUP KODLARININ PERFORMANS ANALİZİ

AL-IBRAHEEMI, Ali
Yüksek Lisans, Elektronik ve Haberleşme Mühendisliği Anabilim Dalı
Tez Yöneticisi: Doç. Dr. Orhan GAZİ

Ocak 2015, 37 sayfa

Bu tez çalışmasında kutup kodların ardışık olarak çözümlenmesi için bir yöntem önerdik. Önerilen yöntem paralel işlemcilerle gerçekleşebilir ve yüksek çözüm hızına sahip iletişim sistemlerinin yapılmasında kullanılabilir. Önerilen metodun donanımsal olarak gerçekleştirimi oldukça basittir. Önerilen algoritma Matlab programlama dili kullanılarak bilgisayar ortamında benzetilmiştir. Benzetim sonuçları ikili silinti kanallar için yapılmış ve performans grafikleri çizilmiştir. Önerilen algoritma diğer ayrık hafızasız kanallar için de kullanılabilir.

Anahtar Kelimeleri: Kutup Kodları, Kanal Kutuplanması, Sıralı İptal Çözümü.

ACKNOWLEDGEMENTS

I would like to take this opportunity to pass my honest appreciation to Assoc. Prof. Dr. Orhan Gazi for the continuous backup and the valuable information and guidance, which helped me in completing this thesis through, varies stages.

I want also to make use of this this chance to thank my father, the man who spends his life just to effort a better life for us no matter what I am going to say about him, it will not be enough, never.

Mother, you are the brightness of my life, to be or not to be an important person in this world I am certain that you will always love me, support me and you will be proud of me. You are the kindest person in this world.

I also wish to thank my friends and all those who supported me with my appreciation and respect for their encouragement and assistance. My heartfelt thanks.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

DMC	Discrete Memoryless Channel
BCJR	Bahl Cocke Jelinek Raviv
SNR	Signal to Noise Ratio
PCM	Pulse Coded Modulation
BEC	Binary Erasure Channel
BSC	Binary Symmetric Channel
B-DMC	Binary-Discrete Memoryless Channel
RM	Reed-Muller
SC	Successive Cancellation
DE	Decision Element
AWGN	Additive White Gaussian Noise
BCH	Bose Chaudhuri Hocquenghem
LDPC	Low Density Parity-Check

CHAPTER 1

INTRODUCTION

1.1. Channel Coding

The primary research in the coding was found on an algebraic approach. The main attention was given to the development of linear binary codes which were huge minimum distance i.e., the least gap between two discrete code words and also good arithmetical properties. The main objective of these code words is to construct the data which is distorted by noise during the transmission. To make the decoding procedure simple at the receiver end, a two-step approach was adopted.

i. The value of single bits are fixed to either 0 or 1 by making use of channel outputs, provided whatever is more probable. It is known as a hard decision.
ii. The second step uses the code constraints. The decoder chooses such a cord word which seems most nearby to the word received. Therefore, large least distances were needed as it makes possible to correct enormous number of errors. This procedure stands true only as long as the error count is less than 50% of the minimum distance.

Hamming is known as first developer of the algebraic codes and that's why these are named after him. He developed Hamming codes are used to rectify single error codes. Many other algebraic codes were also developed which are worth importance like, BCH codes, Golay codes, Reed-Solomon codes, Reed-Muller codes, etc. Proficient algebraic decoding algorithms have been developed for the above mentioned codes and these codes are vastly used in CD's, DVD's, modems, etc.
Product codes were introduced by Elias and were the very first assemblies which achieved both non-vanishing relative distance and rate in the block length. The hint is to make large codes by joining two or more codes of lesser length.

One more construction which is based on combining codes is Code Concatenation as introduced by Forney. Hereby the data is encoded using the C1 the resulting outcome is encoded with C2. Probabilistic decoding also played a vital role in improving the steps of decoding performance. Elias introduced Convolutional codes which were very well suited for the probabilistic decoding. The structure of codes enabled the building of effective decoding algorithms [1].

The Viterbi algorithm is well known to cut off the lock error probability and BCJR algorithm reduces the bit error probability and both operate with the complexity which is linear in the block length. Fano's sequential decoding algorithm is used for practical purposes. Another course of codes presented in the 60's were Low-density parity-check (LDPC) codes whose parity check matrix of codes has very less non-zero entries. In fact, these matrices have a continuous figure of non-zero entries per row and column [2]. Gallager revealed that such codes have a non-zero comparative distance. He also presented a less-complex decoding algorithm but because of insufficient calculating and computational resources at that time, the real power of the codes and decoding algorithms was not truly realized. In the field of coding the development of turbo codes by Berrou, Glavieux and Thitimajshima was a great revolution. Using the linear complexity decoding algorithm, Turbo codes achieved values close to the size and that too at too reasonable than the cutoff rate. The code is built by joining character strings end-to-end i.e., concatenation of two convolutional codes but with a haphazard bit interleaver in between. The decoding algorithm functions in repetitions. In each reiteration the BCJR algorithm is executed on each of the module codes and the consistencies are swapped. Since the complexity of the BCJR algorithm is linear in the block length, the consequential decoding algorithm is also of linear complexity [3].

Codes founded on scarce matrices were constructed by MacKay and Neal and detected that they achieved very well and low complexity belief propagation algorithm. It was lately observed that codes were exceptional case of LDPC codes and the decoding algorithm was similar to the probabilistic decoding as have been recommended by Gallager. Meanwhile, Sipser and Spielman created the expander codes which offered simple decoding algorithm that enabled rectification of linear fraction of combative errors.

Wiberg, Loeliger and Kotter integrated turbo codes and LDPC codes beneath the outline of codes on graphs. The triumph of turbo codes followed by the rekindling of LDPC codes revitalized the curiosity in LDPC codes and message passing algorithms.

1.2. Shannon's Theorem

How much data rate can be carried or supported by the medium in one second is a diversified question. The Shannon's Theorem acknowledges communication systems design as an in vain, that have a huge debate about. Shannon's theory reveals information about the quantity of data any medium can carry per second. This theorem determining the capacity of the channel can be stated simply as below [4]:

- Each transmission system has its Channel capacity (C) which specifies the maximum rate of information it can carry.

- Being transmition rate R, lower than C, the communication in the noise can happen with randomly trivial error possibilities by spending smart coding methods.

- To reduce the error possibilities, the encoder needs to execute on longer blocks of the signal information but this will be requiring relatively longer postponements and crucial calculations.

With randomly small errors, as the *Shannon-Hartley theorem* specifies, and with satisfactorily progressive coding techniques, transmission of data close to maximum capacity of the medium is possible to attain but as the data rate will increase, the chances of probable errors per second will also increases.

1.2.1. Shannon–Hartley equation

The equation depicts a relationship between the maximum capacity C of a given medium with certain noise presence and bandwidth. According to Shannon- Hartley equation, for an AWGN the maximum capacity is specified as:

$$C = B log_2 (1 + SN) \qquad (1.1)$$

B here denoted as a bandwidth and C considered as a highest capacity, capacity value counted in bits and bandwidth in hertz. These both attributes for a given channel calls as *Shannon's capacity limit*. Although, single power here is S and noise power is N and both of them major value is *watt* respectively and together their ratio denotes as *SNR (signal to noise ratio)*. A connection by which data can be communicated over with minor errors, limited signal, noise and bandwidth level as well, should be established. It actually defines the volume of data can be transferred within specified time frame, without errors and highest speed over the channel, especially when the signal power is in *Watts.* This reveals this transmission Gaussian White (uncorrelated) Noise (*Watts*) of additive nature [5].

The limit of Shannon's capacity is specifically designed for the given channel in this study. This channel is ideal to broadcast highest capacity that can be travelled smoothly with several coding combination and decoding structures. These above characteristics of channel capacity define the following communication rules:

- The transmission time and bandwidth requirement for the data symbols to be transmitting over the channel.

- It defines SNR (signal noise ratio) as a communication function that measure signal quality, capacity and channel properties. It actually identifies the amount of information to be wrapped in each symbol to be transmitted. SNR ensures the reliability of transmitted data against errors and noise at receiver end.

- In order to speed up the data transfer rate, the allocated bandwidth and the signal-to-noise ratio have to be traded against each other

- The infinite data transfer rate needs less bandwidth to avoid noise.

Thus we may trade off bandwidth for SNR. However, the increment of the bandwidth value will cause an increment in the power noise. Whenever the bandwidth *B* converge to infinity, that will make the channel capacity to be finite.

The communication channel defined by Shannon is based on following two points:

- SNR and network speed can possibly have a trade-off.

- SNR and network speed are the major sources on which information capacity relies.

Edward Amstrong in 1936 presented his study in support of these above analysis, he stated that frequency modulation (FM) is an appropriate and feasible choice for trade-off between SNR and bandwidth. He also added that by utilizing the FM, it is possible to increase the SNR in transmission system by assigning the more bandwidth [6].

W.M Miner also introduced a similar concept in 1903. In his study, he used time division and sampling methods. In 1973, A.H Reeves has also added in this concept by integrating a quantizer and introduced a technique called PCM (Pulse Coded Modulation). In his research, he includes some extra repeaters to combine the noise over channel in each transmission interval. In his case he required more network speed than usual. He utilized a quantizer with a large number of quantization levels, with the goal of minimizing the quantization noise,. Reeves patent relies on two important facts:

- An analog signal can be presented with an random accuracy, if an appropriate sampling frequency and quantized sample is applied properly in a pre-decided levels of amplitude

- The quantized samples can be possibly collected with less random errors, if SNR is large enough for it.

According to Reeve's patent, an unlimited volume of data can be sent over a network which is noise free needs less bandwidth then it is implicit. This links the information rate with SNR and bandwidth.

1.2.2. Unconstrained Shannon Limit for AWGN channel

Gaussian channel can be demonstrated by some general characteristics. The transmission rate denoted as R is similar to the capacity C: R=C. Let's observe if you send some random binary digits over the channel known as AWGN. The average signal power here is S, thus the energy per bit can be derived as E_b=S/C, while bit duration here is 1/C per second. The total noise power would be $N0BWatts$, if the other hand noise power spectral density is $N0/2Watts/Hertz$ (power regularized to 1Ω resistance). Thus the Shannon-Hartley equation becomes [7].

$$CB=log_2(1+E_bN_0CB) \tag{1.2}$$

Rearranging the equation,

$$E_bN_0=BC(2CB–1) \tag{1.3}$$

Assume $C/B=\eta$ (in *bits/seconds/Hz*), the spectral efficiency

$$E_bN_0=2\eta–1\eta \tag{1.4}$$

In the Figure below, the red dashed line in the plot refers to the converge of $Eb/N0$ as the bandwidth B near to infinity. The asymptote is at $Eb/N0=ln(2)=-1.59dB$. The previous amount is named Shannon's Limit (Shannon's power efficiency limit).

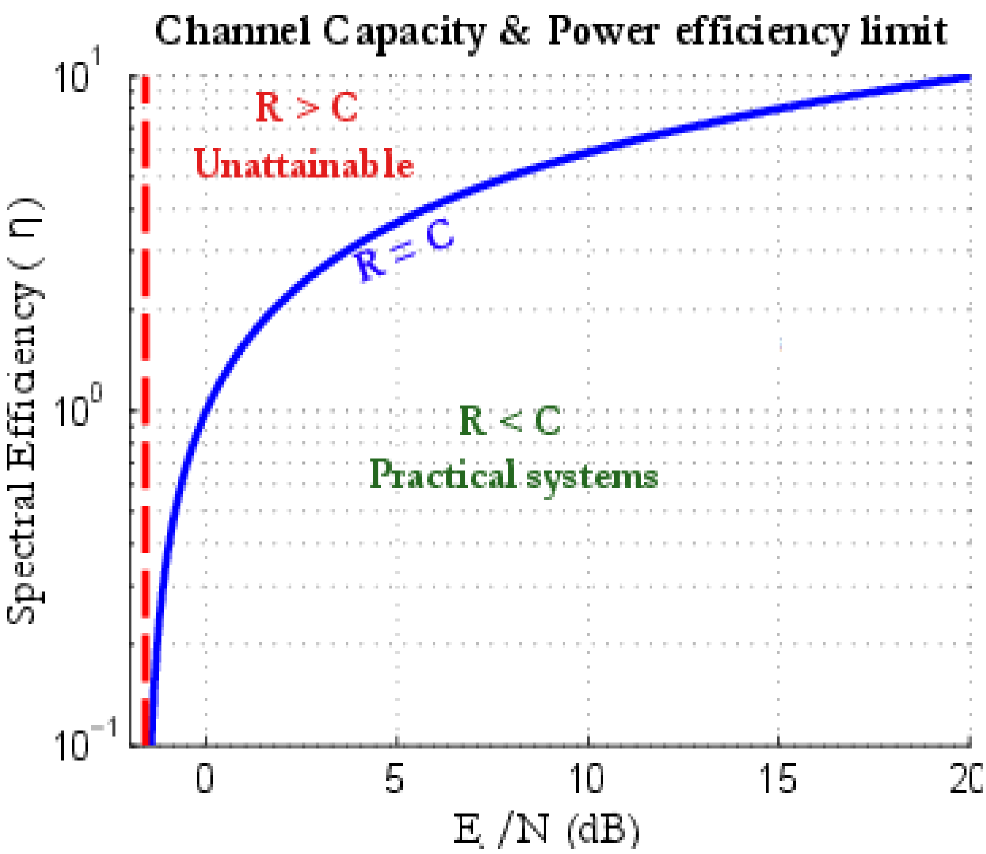

Figure 1: Channel Capacity

1.3. Discrete Memoryless Channel (DMC)

DMC is a discrete channel with input variable {x € X} and output variable {y € Y} and the transition probability function p(y/x) for {x € X} and {y € Y} ,both of input and output are random variable, the output of channel is theoretical to depend only on the present input ,because of that its known as Memoryless channel [3].

$$P_r\,[Y = y \mid X = x] \triangleq P(y \mid x) \tag{1.5}$$

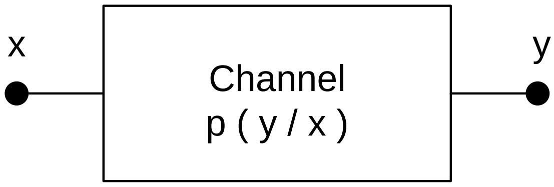

Figure 2: Discrete Memoryless Channel

In an ideal channel, the output and the input are equal, but in a non-ideal channel the output and the input can be different from each other with a presented probability [8].

Popular models of DMC are:-

1- Binary erasure channel (BEC).
2- Binary symmetric channel (BSC).

1.3.1. Binary symmetric channel (BSC)

BSC is the type of channel used in coding and information theory BSC is a popular communication channel model, it is one of the simplest channels to analyze. These channels work with binary input and binary output (0 and 1) and the transitional error probability (p). The possibility of getting a (1) if a (0) is sent and the possibility of getting a (0) if a (1) is sent are the same, that is why it called symmetric channel [9]. The conditional distribution of BSC is represented as follow:

$$\begin{aligned} P(y=0/x=1) &= P(y=1/x=0) = P_e \\ P(y=0/x=0) &= P(y=1/x=1) = 1-P_e \end{aligned} \tag{1.6}$$

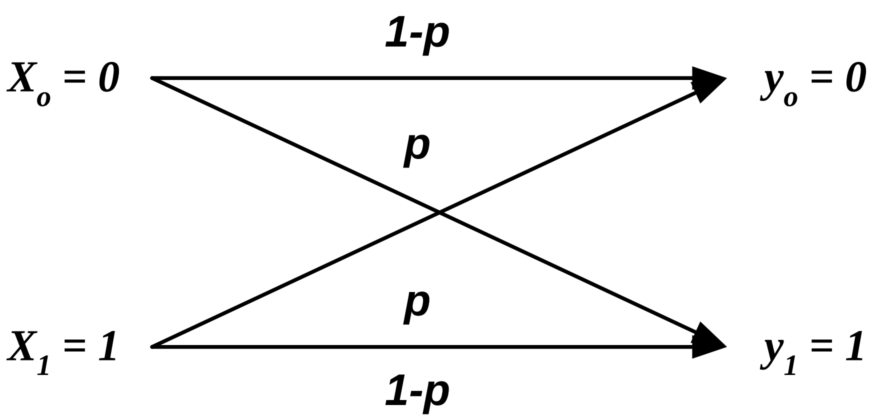

Figure 3: Binary Symmetric Channel.

1.3.2. Capacity of BSC

$$
\begin{aligned}
&c = \max_{p(x)} I(\tilde{x};\tilde{y}) \\
&I(\tilde{x};\tilde{y}) = H(Y) - H(Y/X) \\
&I(\tilde{x};\tilde{y}) = H(Y) - \sum_{x} p(x)\ \mathrm{H(Y/X)} \\
&I(\tilde{x};\tilde{y}) = H(Y) - \left[p(0)\ \mathrm{H(y/0)} + p(1)\ \mathrm{H(y/1)}\right] \\
&I(\tilde{x};\tilde{y}) = H(Y) - \left[p(0)\ \mathrm{H}_b(\mathrm{p}) + p(1)\ \mathrm{H}_b(\mathrm{p})\right] \\
&I(\tilde{x};\tilde{y}) = H(Y) - \left[\underbrace{(p(0)+p(1))}_{1}\ \mathrm{H}_b(\mathrm{p})\right] \\
&I(\tilde{x};\tilde{y}) = H(Y) - \mathrm{H}_b(\mathrm{p}) \\
&\max_{p(x)} I(\tilde{x};\tilde{y}) = \max H(Y) - \mathrm{H}_b(\mathrm{p})
\end{aligned}
$$

$\tilde{y}$ Contain only (0, 1), so the maximum entropy of $\tilde{y}$

$$
\left.
\begin{aligned}
&\log|R\tilde{y}| = \log|2| = 1 \\
&\max H(\tilde{y}) = 1 \\
&C = \max I(\tilde{x};\tilde{y}) \\
&C = 1 - H_b(p)
\end{aligned}
\right. \quad \text{bits} \tag{1.7}
$$

When the input distribution is uniform, the equality is achieved. The capacity of the binary symmetric channel (BSC) with the probability of error P is:

$$C = 1 - H(p)\ \text{bits} \tag{1.8}$$

1.3.3. Binary erasure channel (BEC)

In BEC channel the transmitter transmits (0) or (1) and the receiving station receives bit or erasure bit (was not received), it is a popular communication channel model. BEC is used frequently in information theory and is also one of the simplest noisy channels to analyze. Sometimes the bit gets erased (with erasure probability) and this type of channels is not perfect and receivers have no idea what the transmitted bit. BEC is a discrete memoryless channel, it deals with two input and three output and the erasure probability (α), let ***X*** be the transmitted random variable (0, 1) and Y the received variable (0, 1, e), then the conditional distribution will be [9]:

$$
\begin{aligned}
&p(y=0/x=0)=1-\alpha\\
&p(y=0/x=1)=0\\
&p(y=1/x=0)=0\\
&p(y=1/x=1)=1-\alpha\\
&p(y=e/x=0)=\alpha\\
&p(y=e/x=1)=\alpha
\end{aligned}
$$

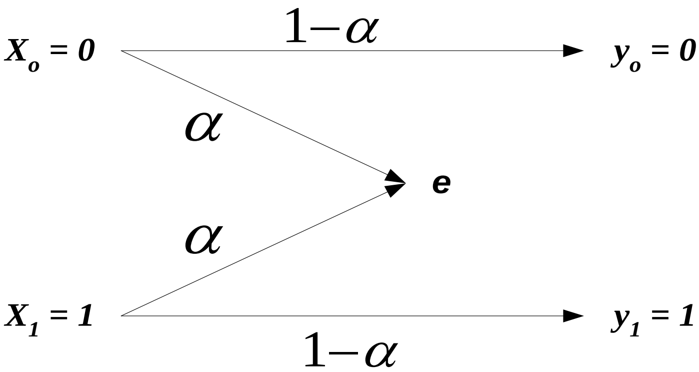

Figure 4: Binary Erasure Channel

1.3.4. Capacity of BEC

$$
\begin{aligned}
&C=\max_{p(x)} I(\tilde{x};\tilde{y})\\
&I(\tilde{x};\tilde{y})=H(Y)-H(Y/X)\\
&I(\tilde{x};\tilde{y})=H(Y)-\left[-\sum_{y} p(y/x)\log p(y/x)\right]\\
&I(\tilde{x};\tilde{y})=H(Y)-\left[-\left((1-\propto)\log(1-\alpha)+(\propto)\log(\alpha)\right)\right]
\end{aligned}
$$

$$
\begin{aligned}
&I(\tilde{x};\tilde{y})=H(Y)-H(\propto)\\
&I(\tilde{x};\tilde{y})=\left[-\sum p(y)\log p(y)\right]-H(\alpha)\\
&I(\tilde{x};\tilde{y})=H(\alpha)+(1-\alpha)H(a)-H(\alpha)\\
&I(\tilde{x};\tilde{y})=(1-\alpha)H(a)\\
&\max I(\tilde{x};\tilde{y})=\max\left[(1-\alpha)H(a)\right]
\end{aligned}
\tag{1.9}
$$

Maximum value of ***H(a)*** is log2=1

$$C=1-\propto \text{ bits} \tag{1.10}$$

1.4. Polar codes

Polar codes invented by Arikan in [10], Polar codes deals with any symmetric binary-input discrete memoryless channel (B-DMC) these type of codes have low complexity in the encoding part and in the decoding part. Polar code is the first class of coding to achieve the provably capacity for any binary discrete memoryless channel (B-DMC). Where N refers to the block length, with the complexity $O(N \log N)$.

CHAPTER 2

CONSTRUCTION OF POLAR CODES

2.1. Polar Codes

The Arıkan's earlier works on improving the computational cutoff rate of channels [11] are the source of the idea of polar codes which can be traced back in this paper. Before the transmission process starts, a meek linear inputs the channel and late a cancellation decoding node at the output by Arikan. By assuming 'W' as a real mode whereas $W_1 W_2$ assumed as the modes to be decoded bit at first and second correspondingly. A larger average (over W_1 and W_2) cutoff rate compared to that of can be calculated through such a transformation.

Process described above "recursively" can be repeated for the idea of polar codes. A linear transform on a larger number of bits at the encoder can be applied through the recursive process.

The bits get decoded correspondingly in a defined sequence. Hence, consequently, some of the bits see some effective channels which are way better than W and can be worse sometimes. The effective channels as identified by bits interestingly incline in a direction of either a completely noisy channel or a clean channel with the fraction of clean channels approaching the capacity of as the block-length increases. Arıkan define it to be as channel polarization. Without any coding this explains a simple plan where we fix the inputs to the channels that are bad and transmit reliably over the clean channels. The capacity of the channel approaches the rate of such a scheme[12]. However, with low-complexity it relics to display recursive conversion while encoding and the successive cancellation decoder can be applied for the formula to work successfully. He also discovered that the decoding and encoding, both procedures, can be realized by applying a Fast-Fourier-like conversion method. Thus, by using a low-complexity procedure, the capacity is finally attained by the sequence [13].

Arıkan's contribution conversion matrix is agreed by $G_2^{\otimes n}$ whereas

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

And "$\otimes n$" stands for the *n-th* Kronecker product. An equivalent process to select some of the input bits to convert information and by fixing the remainder bits, is achieved by selecting the rows of $G_2^{\otimes n}$ which make the author matrix of the code. These codes are mentioned as polar codes and are carefully connected to RM codes. From the rows of $G_2^{\otimes n}$ the producer matrix of RM ciphers is also built. The instruction used for opting the rows of their generator matrices lays the vital variance between the two codes. The tiniest distance of the code is exploiting the RM codes which resemble to the choice. On the other hand for polar codes, the channel which is used the choice is dependent on it. The choice enhances the presentation under successive cancellation decoding [10] [12].

When $N = 2^n, n \geq 1,\ 1 \leq K \leq N$, a (N, K) polar code is a block code and the generator matrix of polar codes is a $(K \times N)$ sub matrix of $F^{\otimes n}$ structured in correspondence to the steps below. In the beginning, the vector $z_N = (z_{N,1}, ..., z_{N,N})$ must be estimated by the recursion below:

$$z_{2k,j} = \begin{cases} 2z_{k,j} - z_{k,j}^2 & \text{for } 1 \leq \text{j} \leq \text{k} \\ z_{k,j-k}^2 & \text{for k+1} \leq \text{j} \leq \text{2k} \end{cases} \tag{2.1}$$

The value of $k = (1, 2, 2^2, ..., 2^{n-1})$, the initial $z_{1,1} = 0.5$. After that, the permutation $\pi_N = (i_1, ..., i_N)$ must be formed of the set $(1, ..., N)$ so that, for any $1 \leq j \leq k \leq N$, the allowance $z_{N,ij} \leq z_{N,ik}$ is correct. The generator matrix of polar code $\text{G}(N, K)$ is introduced as the sub-matrix of $F^{\otimes n}$ depending on the rows with indexes $\left(i_1,....,i_k\right)$. It is simple to notice that the complexity of the construction of the polar codes methodology is $O(N\ \log N)$.

2.1.1. G$_N$-coset codes encoder

G_N-coset codes encoders are more general and contain polar codes as well. The format of the block length is $N = 2^n$, $n \geq 0$ and the decoding formula is:

$$x_1^N = u_1^N G_N(A) \oplus u_{A^c} G_N(A^c) \tag{2.2}$$

G_N represents the creator array of N order, A represents an random subgroup of $\{1,....,N\}$, $G_N\left(A\right)$ represents a sub-matrix of G_N constructed with indexes in *A*. G_N-coset codes must be decided via a $\left(K, N, A, u_{A^c}\right)$variable vector, K represents the volume of *A* and the code. if the group *A* is selected as a *K*-subgroup of $\{1,....,N\}$,then a G_N-coset code is named the polar code.

Given B-DMC W , there are two channel parameters, for the measurement of reliability and rate these parameters can be used. The I(W) here is the maximum rate that can make the communication possible over *W*. it is possible by giving an input of *W*, which is equal to frequency. However, when *W* used only once to convey a 0 or 1, then $W(Z)$ is a bound input that have maximum-likelihood (ML) decision error, the symmetric capacity [12]:

$$I(W) \triangleq \sum_{y \in Y} \sum_{x \in X} \frac{1}{2} W(y \mid x) \log \frac{W(y \mid x)}{\frac{1}{2} W(y \mid 0) + \frac{1}{2} W(y \mid 1)} \tag{2.3}$$

The reliability factor (Bhattacharyya):

$$Z(W) \triangleq \sum_{y \in Y} \sqrt{W(y \mid 0)W(y \mid 1)} \tag{2.4}$$

From the above equations, it is clear that $W(Z)$ accept values in [0, 1] form. Hence, base-2 logarithm will be ideal to use here, because $I(W)$ also worked with [0, 1] values. The code rates and channel volumes unit is used in bits. Spontaneously, it would be as $I(W)$ approaches to 1 if $Z(W)$ approaches to 0, and $I(W)$ approaches to 0 if $Z(W)$ approaches to 1.

2.1.2. Successive cancellation (SC) decoder

A G_N-coset codes perform encoding of u_1^N input to x_1^N, output. That later on, transfer to channel W^N and converts into y_1^N output. After that y_1^N decoding estimates the $\tilde{u}_1^N$ as the real input. While, In the frozen part errors can be avoided in decoding part, however, the actual decoding measures $\tilde{u}_A$ of u_A . The given an $\left(N, K, A, u_{A^c}\right)$ G_N-coset code, here a SC decoding is performed that produces its decision $\tilde{u}_1^N$ by calculating [14]:

$$P_e\left(N, K, A, u_{A^c}\right) \leq \sum_{i \in A} Z\left(W_N^{(i)}\right) \tag{2.5}$$

$$\tilde{u}_i \triangleq \begin{cases} u_i, & \text{if } i \in A^c \\ h_i(y_1^N, \tilde{u}_1^{i-1}), & \text{if } i \in A \end{cases} \tag{2.6}$$

In the arrangement *i* from 1 to *N*, where $h_i : Y^N \times X^{i-1} \rightarrow X, i \in A$, are the functions of decision known as:

$$h_i(y_1^N, \tilde{u}_1^{i-1}) = \begin{cases} 0, & \text{if } \dfrac{W_N^{(i)}(y_1^N, \tilde{u}_1^{i-1} \mid 0)}{W_N^{(i)}(y_1^N, \tilde{u}_1^{i-1} \mid 1)} \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{2.7}$$

For all $y_1^N \in Y^N, \tilde{u}_1^{i-1} \in X^{Ni-1}$. a decoder block error occurred if $\tilde{u}_1^N \neq u_1^N$.

2.1.3. Decoding

It considers successive cancellation (SC) decoding process for a parameter vector $\left(N, K, A, u_{A^C}\right)$ G_N-coset code. It may be visualized the decoder as depending on number (N) of decision elements, each u_i (source element) has one. Whether $i \in A^c$, the u_i element is recognized. If $i \in A$, *i-th* decision elements doesn't process further unless received any direction that had done previously $\tilde{u}_1^{i-1}$, and as soon as it receives the instructions then the ratio is calculated as follows [14]:

$$L_N^{(i)}(y_1^N, \tilde{u}_1^{i-1}) \triangleq \frac{W_N^{(i)}(y_1^N, \tilde{u}_1^{i-1} \mid 0)}{W_N^{(i)}(y_1^N, \tilde{u}_1^{i-1} \mid 1)} \tag{2.8}$$

And generates it's decision as:

$$\tilde{u}_i = \begin{cases} 0, & \text{if } \mathrm{L}_N^{(i)}(y_1^N, \tilde{u}_1^{i-1}) \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{2.9}$$

And the recursive formulas give:

$$L_N^{(2i-1)}(y_1^N, \tilde{u}_1^{2i-2}) = \frac{L_{N/2}^{(i)}(y_1^{N/2}, \tilde{u}_{1,o}^{2i-2} \oplus \tilde{u}_{1,e}^{2i-2}) L_{N/2}^{(i)}(y_{N/2+1}^N, \tilde{u}_{1,e}^{2i-2}) + 1}{L_{N/2}^{(i)}(y_1^{N/2}, \tilde{u}_{1,o}^{2i-2} \oplus \tilde{u}_{1,e}^{2i-2}) + L_{N/2}^{(i)}(y_{N/2+1}^N, \tilde{u}_{1,e}^{2i-2})} \tag{2.10}$$

And

$$L_N^{(2i)}(y_1^N, \tilde{u}_1^{2i-1}) = \left[L_{N/2}^{(i)}(y_1^{N/2}, \tilde{u}_{1,o}^{2i-2} \oplus \tilde{u}_{1,e}^{2i-2})\right]^{1-2\tilde{u}_{2i-1}} . L_{N/2}^{(i)}(y_{N/2+1}^N, \tilde{u}_{1,e}^{2i-2}) \tag{2.11}$$

The mathematical operation remains continuously downward to block length 1, where LRs has the following form:

$$L_1^{(1)}(y_i) = \frac{W(y_i \mid 0)}{W(y_i \mid 1)} \tag{2.12}$$

Hence, the values are computed directly.

The next example shows the decoding of a recursive successive cancellation decoding of polar codes. Let the input bits assumed to be

$u = [1 \quad 1]$

And the generator matrix is $G = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$

So the encoded bits will be

$x = u * G$

$x = [0 \quad 1]$

These encoded bits will send over a binary erasure channel with erasure probability $\alpha = 0.01$

So the received bits will be

$y = [0 \quad 1]$

Solution:

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

$$h_i(y_1^N, \tilde{u}_1^{i-1}) = \begin{cases} 0, & \text{if } \dfrac{W_N^{(i)}(y_1^N, \tilde{u}_1^{i-1} \mid 0)}{W_N^{(i)}(y_1^N, \tilde{u}_1^{i-1} \mid 1)} \geq 1 \\ 1, & \text{otherwise} \end{cases}$$

$$W_N^{(i)}(y_1^N, \tilde{u}_1^{i-1} \mid u_i) \triangleq \sum_{U_{i+1}^N \in X^{N-i}} \frac{1}{2^{N-1}} W_N(y_1^N \mid u_1^N)$$

$$W_N(y_1^N \mid u_1^N) = \prod_{i=1}^{N} W(y_i \mid x_i)$$

$$W_N(y_1^N \mid u_1^N) = W^N(y_1^N \mid x_1^N)$$

For the first channel i=1

$$W_2^{(1)}(y_1^2 \mid u_1) = \sum_{u_2} \frac{1}{2} W_N(y_1^2 \mid x_1^2)$$

$$W_2^{(1)}(y_1^2 \mid u_1) = \frac{1}{2}[W(y_1 \mid x_1) W(y_2 \mid x_2)]$$

$x_1 = u_1 \oplus u_2$

$x_2 = u_2$

$$W_2^{(1)}(y_1^2 \,|\, u_1) = \frac{1}{2}[W(y_1 \,|\, u_1 \oplus u_2)W(y_2 \,|\, u_1)]$$

$u_1 = 0$
$u_2 = 0,1$

$$W_2^{(1)}(y_1^2 \,|\, u_1) = \frac{1}{2}[W(y_1 \,|\, u_1 \oplus u_2)W(y_2 \,|\, u_1) + W(y_1 \,|\, u_1 \oplus u_2)W(y_2 \,|\, u_1)]$$

$$W_2^{(1)}(y_1^2 \,|\, 0) = \frac{1}{2}[W(0 \,|\, 0 \oplus 0)W(1 \,|\, 0) + W(0 \,|\, 0 \oplus 1)W(1 \,|\, 1)]$$

$$W_2^{(1)}(y_1^2 \,|\, 0) = \frac{1}{2}[(1-\alpha)(0) + (0)(1-\alpha)]$$

$$W_2^{(1)}(y_1^2 \,|\, 0) = 0$$

$u_1 = 1$
$u_2 = 0,1$

$$W_2^{(1)}(y_1^2 \,|\, u_1) = \frac{1}{2}[W(y_1 \,|\, u_1 \oplus u_2)W(y_2 \,|\, u_1) + W(y_1 \,|\, u_1 \oplus u_2)W(y_2 \,|\, u_1)]$$

$$W_2^{(1)}(y_1^2 \,|\, 1) = \frac{1}{2}[W(0 \,|\, 1 \oplus 0)W(1 \,|\, 0) + W(0 \,|\, 1 \oplus 1)W(1 \,|\, 1)]$$

$$W_2^{(1)}(y_1^2 \,|\, 1) = \frac{1}{2}[(0)(0) + (1-\alpha)(1-\alpha)]$$

$$W_2^{(1)}(y_1^2 \,|\, 1) = 0.49$$

$$\frac{W_2^{(1)} \text{ for } \mathrm{u}_1 = 0}{W_2^{(1)} \text{ for } \mathrm{u}_1 = 1} \geq 1$$

$\dfrac{0}{0.49} = 0 \qquad \Rightarrow u_1 = 1$ is the first decoded bit

To compute the second decoded bit

For the second channel i=2

$$W_2^{(2)}(y_1^2, u_1 \,|\, u_2) = \frac{1}{2} W_2(y_1^2 \,|\, u_1^2)$$

$$W_2^{(2)}(y_1^2, u_1 \,|\, u_2) = \frac{1}{2} W(y_1 \,|\, u_1 \oplus u_2)W(y_2 \,|\, u_2)$$

$u_1 = 1$ known
$\mathrm{u}_2 = 0$

$$W_2^{(2)}(y_1^2,1|0)=\frac{1}{2}W(0|1\oplus 0)W(1|0)$$
$$W_2^{(2)}(y_1^2,1|0)=0$$
$u_1=1$ known
$u_2=1$
$$W_2^{(2)}(y_1^2,1|1)=\frac{1}{2}W(0|1\oplus 1)W(1|1)$$
$$W_2^{(2)}(y_1^2,1|1)=0.49$$
$$\frac{W_2^{(2)}\ \text{for } u_2=0}{W_2^{(2)}\ \text{for } u_2=1}\geq 1$$
$\frac{0}{0.49}=0 \qquad \Rightarrow u_2=1$ is the second decoded bit

Finally the decoded bits are the same input bits

2.2. Channel Polarization

A step by step guideline that construct out of N autonomous copies of binary discrete memoryless W, another group of N number of channels $\{W_N^{(i)}:1\leq i\leq N\}$ that disclose a division outcome in a manner, where N turns to maximum, the symmetric volume terms $\{I(W_N^{(i)})\}$ incline to 0 or 1for all but a disappearing portion of indexes i, called channel polarization. In short, this process is a combination of channel combining and splitting [15].

2.2.1. Channel combining

Here, the B-DMC W combines in a recursive mathematical process to generate $W_N:X^N\rightarrow Y^N$ as vector channel v, N can be computed by, $N=2^n, n\geq 0$. The recursion mathematical procedure starts at level 0 $(when\ n=0)$ with W's single copy that creates set $W_1\triangleq W$. $(n=1)$ represents the first level of the recursion combine two autonomous copies of W_1 as mention below in fig. 5 and finds the channel $W_2 : X^2 \rightarrow Y^2$ by the alteration likelihoods:

$$W_2(y_1, y_2 | u_1, u_2) = W(y_1 | u_1 \oplus u_2) W(y_2 | u_2) \quad (2.13)$$

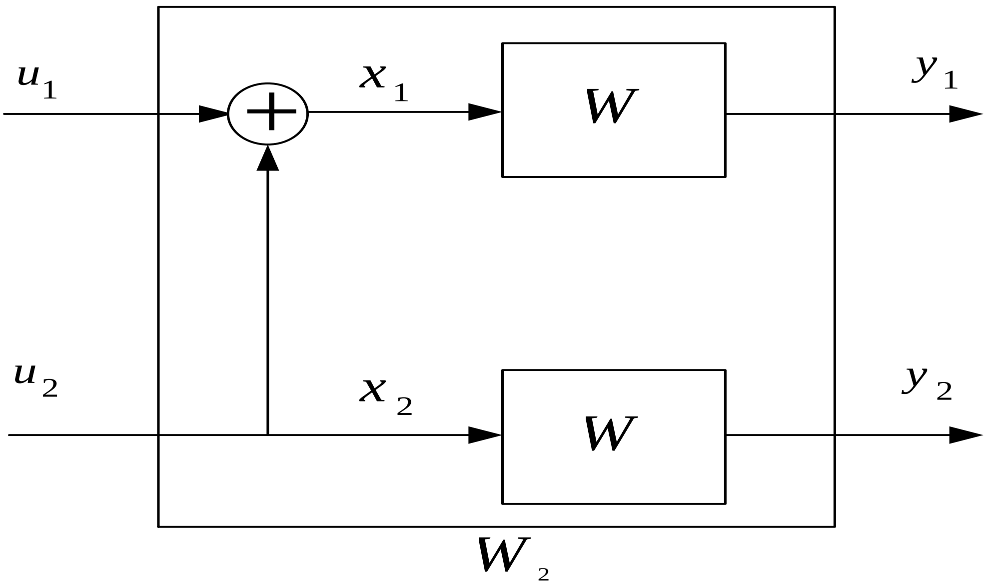

Figure 5: Channel W_2 Combining

Furthermore, the next phase presents the recursion, as shown in fig 2. Here, W_2 have two autonomous copies that together design the channel $W_4 : X^4 \rightarrow Y^4$ with transmission probability:

$$W_4(y_1^4 | u_1^4) = W_2(y_1^2 | u_1 \oplus u_2, u_3 \oplus u_4) W_2(y_3^4 | u_2, u_4) \quad (2.14)$$

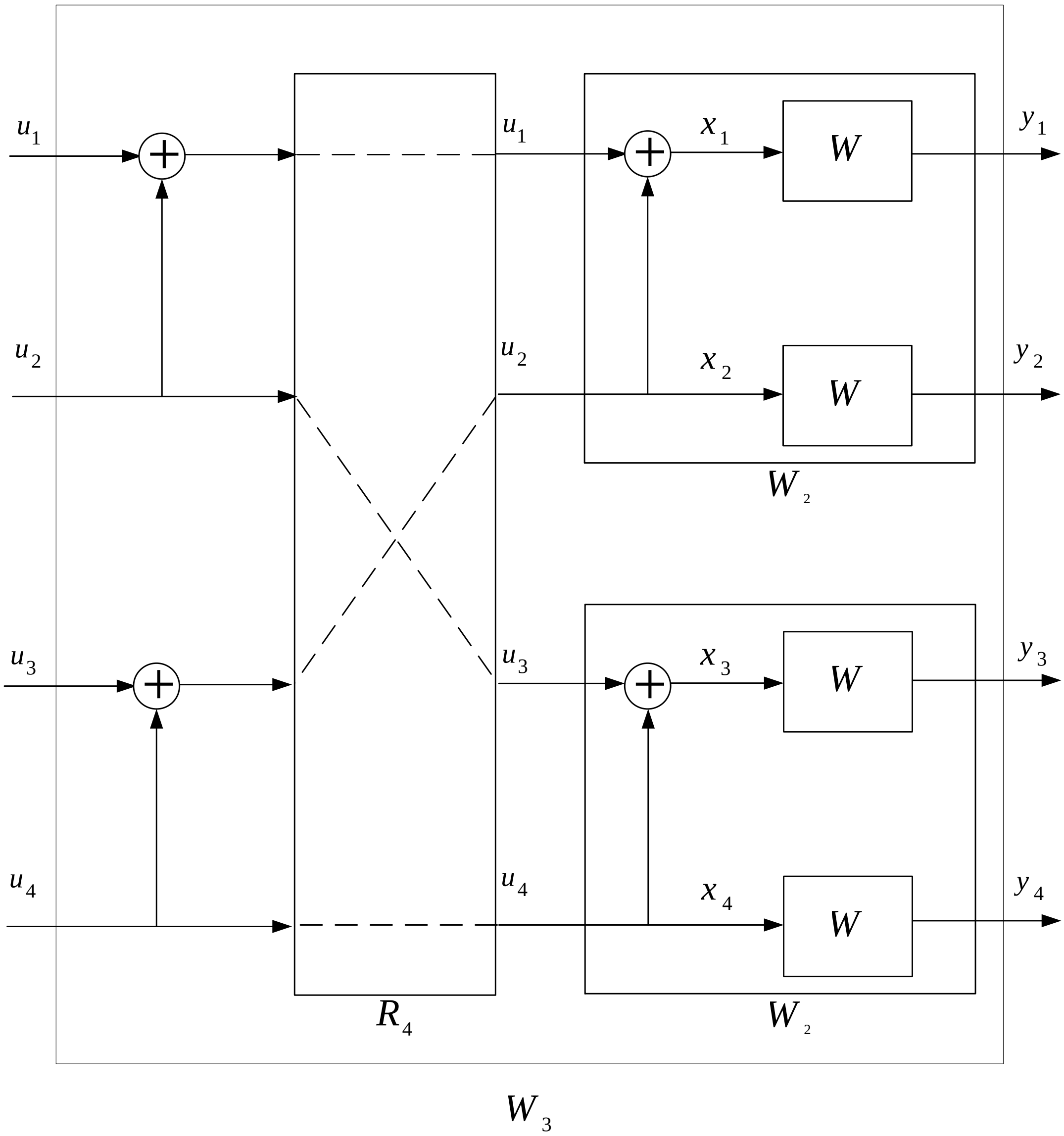

Figure 6: Channel W_4 Combining

The fig. 6 presents R_4 that perform permutation process to plot (s_1, s_2, s_3, s_4) input into $v_1^4 = (s_1, s_2, s_3, s_4)$. this plotting $u_1^4 \to x_1^4$ out of the W_4 to the input W^4 can be mentioned as $x_1^4 = u_1^4 G_4$ with:

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

Thus, the relation $W_4\left(y_1^4 \mid u_1^4\right)=W^4\left(y_1^4 \mid u_1^4 G_4\right)$ between the transition probabilities of W_4 and those of W^4.

The fig. 7 shows the common form of recursion. where $W_{N/2}$ have two autonomous copies that together generate the channel W_N. u_1^N the input vector to W_N is first altered into s_1^N then $s_{2i-1}=u_{2i-1} \oplus u_{2i}$ and $s_{2i} = u_{2i}$ for $1 \le i \le N/2$. R_N factor in the figure is a variation that represent the reverse mix process, and s_1^N perform on driven input from it to generate $v_1^N = \left(s_1, s_3, ..., s_{N-1}, s_2, s_4, ..., s_N\right)$, that converts that driven input into the following two channels copies of $W_{N/2}$ as the figure illustrate .

With above calculations, it is founded by plotting $u_1^N \rightarrow v_1^N$ as a linear over GF(2), It trails the entire plotted $u_1^N \rightarrow x_1^N$, by induction. the W_N from manufactured channel's input to the underlined input of the group channels W^N, is furthermore linear and may be signified by a G_N matrix so $x_1^N = u_1^N G_N$. G_N called the generator matrix of volume N. the transmission probability of the channels W_N and W^N are relevant with:

$$W_N\left(y_1^N \mid u_1^N\right)=W^N\left(y_1^N \mid u_1^N G_N\right) \tag{2.15}$$

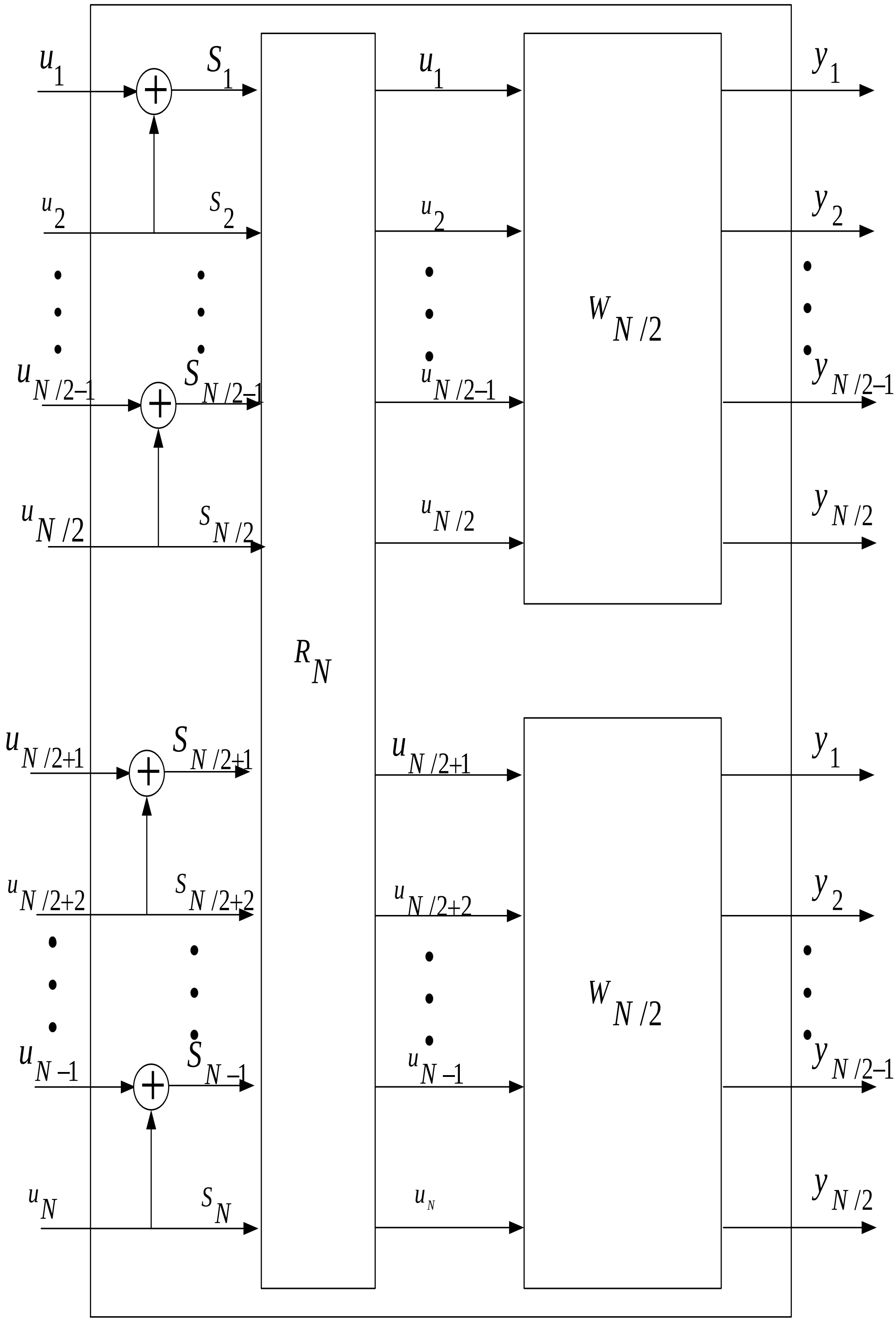

Figure 7: A Recursive Construction of W_N

2.2.2. Channel splitting

As W_N vector channel is derived from W^N, which is shown above, so now the next step in channel polarization is to break the W_N again in a group of number of N synchronized input (binary) channels $W_N^{(i)}: X \to Y^N \times X^{i-1}, 1 \le i \le N$, that are distinct via the transition probability:

$$W_N^{(i)}\left(y_1^N, u_1^{i-1} | u_i\right) \triangleq \sum_{u_{i+1}^N \in x^{N-1}} \frac{1}{2^{N-1}} W_N\left(y_1^N | u_1^N\right) \tag{2.16}$$

Here $\left(y_1^N, u_1^{i-1}\right)$ symbolizes the output of $W_N^{(i)}$ and u_i represents the input of $W_N^{(i)}$. To get an distinctive and meaningful channels $\left\{W_N^{(i)}\right\}$, that measured as a genie-aided SC decoder (successive cancellation decoder) that will the *i*th decision component computes u_i and after detecting y_1^N and the data inputs of the previous channel such as u_1^{i-1} (it delivers properly by the genie irrespective at earlier phases of any decision errors). If u_1^N is a-priori constant on X^N, thus $W_N^{(i)}$ is the actual channel seen by the *i*th decision component in this situation [11] [10].

CHAPTER 3

ANALYSIS OF DECODIND ALGORTHIM

The chapter 3 comprises of the methodology of the study of decoding the polar codes. This decoding method is in such a way that the perceived arrangement is distributed to the nodules of the arrangement and then it sums up these bits with the help of formulae used for decoding the Polar codes. Thus, in short, the decoding procedure depends on following two steps:

- Distribution.
- Combination.

These steps of decoding process are described shortly as below:

3.1. Distribution

Distribution is the first step of decoding procedure. The received sequence bits to such an arrangement is shown as below in figure. The usual sequence is fixed at the most top of the arrangement. The classification is divided as odd and even bits whereby the left line bits subsequent to odd and even bits and on the other hand, the right side aligned bits are called the even bits and it continues till the last layer of sequence. Now, the last bit must be secured to the L1-end if the bits sequence numeral is odd, and for even number the same steps as explained above will be repeated.

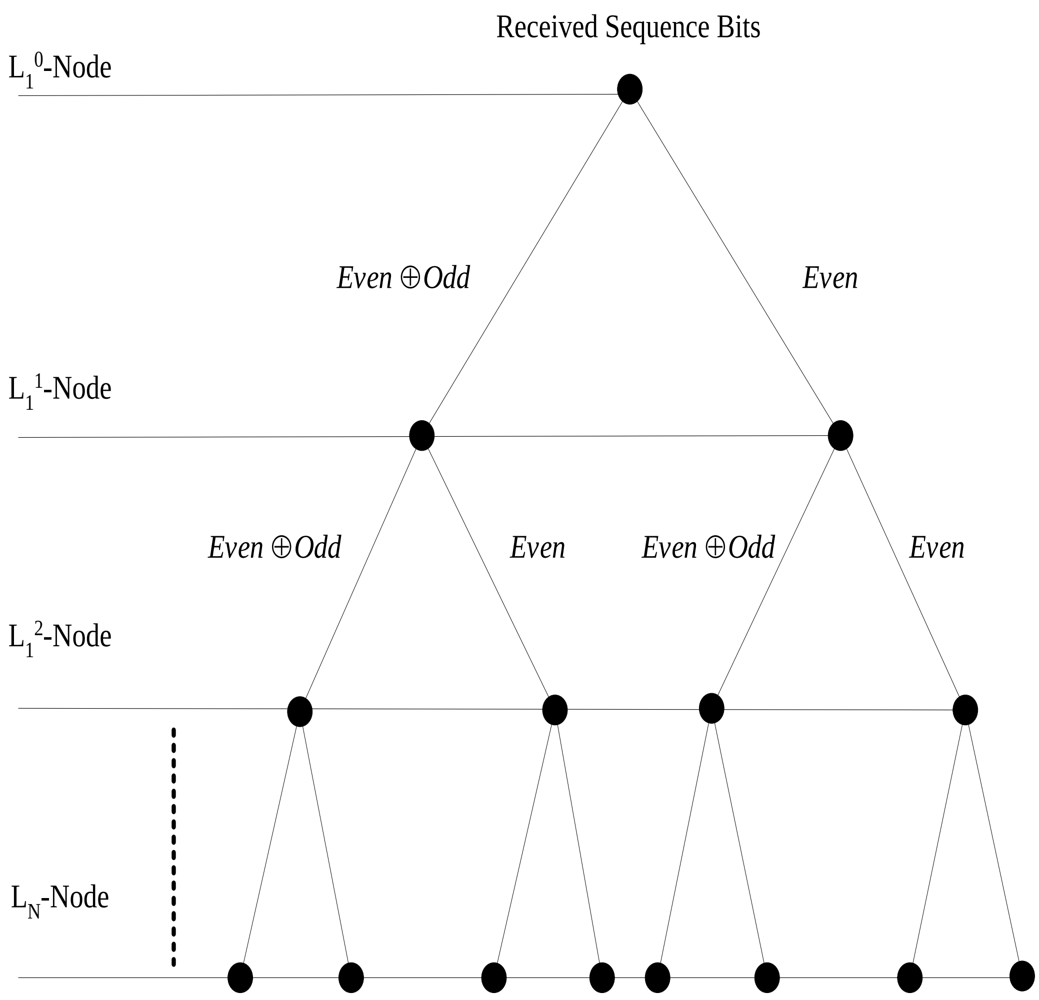

Figure 8: Distribution in General

3.2. Combination

Combining these distributed bits is the second step of the decoding process. Here the distributed bits are combined by inserting them to polar-codes decoding formulae. This step is basically reversing the process of distribution as the combination is started from the bottommost layer of the arrangement unlikely of that in distribution process. The combination is constructed as bellow inserting the bits arrangement to the formulae:

$$L_1^{(1)}(y_i) = \frac{W(y_i \mid 0)}{W(y_i \mid 1)} \tag{3.1}$$

The combination of the next layer is constructed by combining two values from the lower layer as shown below in fig. (9) by the formulas below:

$$F1 = \frac{(1st * 2nd) + 1}{(1st + 2nd)} \tag{3.2}$$

$$F2 = (1st)^{1-2u} * (2nd) \tag{3.3}$$

In this case, L-node should check, if there is a bit in L-node, so $F1$formula should use, and if there is no bit in L-node, so $F2$ formula should use.

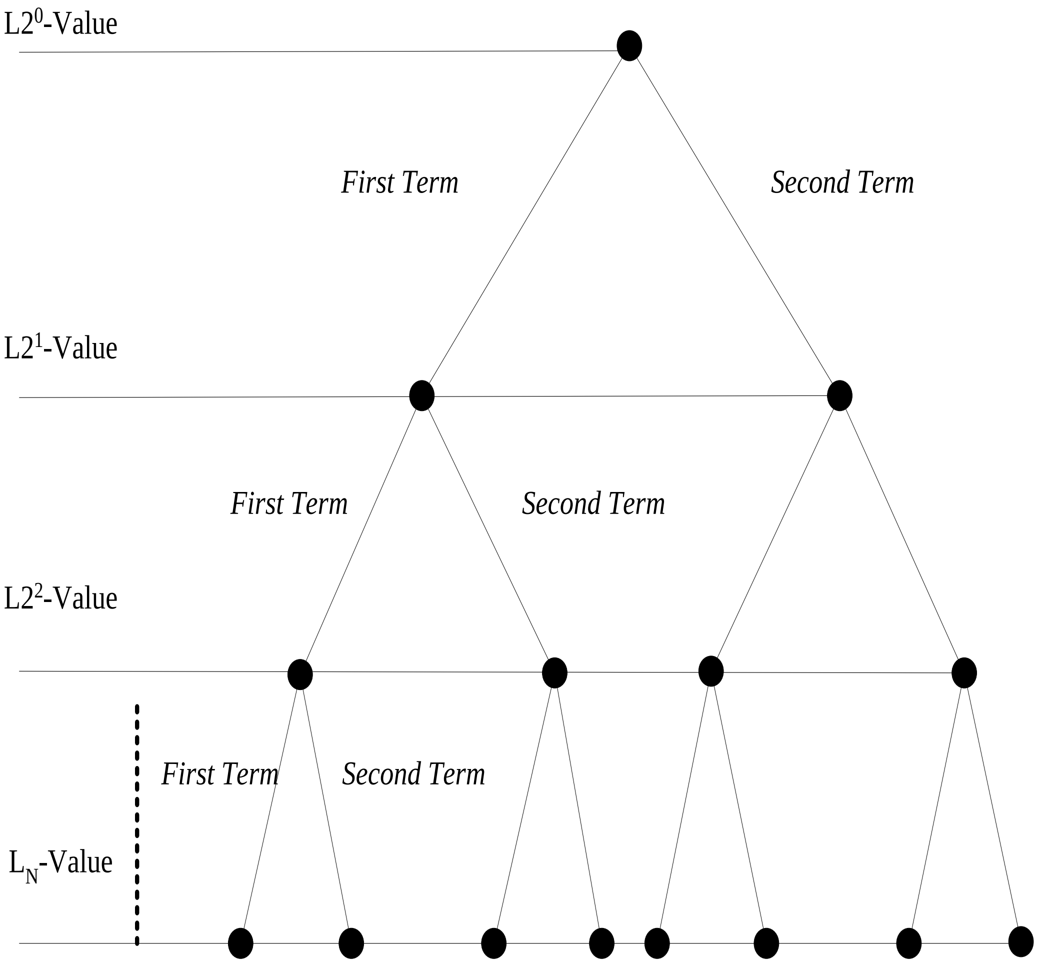

Figure 9: Combination in General

3.3. Numerical Example

The following example is placed to show the way of decoding the polar codes by distributing the conventional encoded classification to manner to sort this process as complete. It also elaborates how to fusion these bits via equations to receive the actual message by decoding those polar ciphers.

1st Step: Assume input data bits:

Data=[1 0 0 1]

And the generator matrix of polar codes for 4 channels is :

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

The input data bits will multiply by the generator matrix to get the encoded bit sequence.

$X = data * G$

$$X = [1 \quad 0 \quad 0 \quad 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$X = \begin{bmatrix} 0 & 1 & 1 & 1 \end{bmatrix}$ The encoded sequence

The sequence will send over discrete memoryless channel (in this example binary erasure channel will use), the erasure probability of binary erasure channel assume to be low value, and then the received sequence will be:

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

First step of decoding process is distribution. The received sequence will distribute to the nodes of the scheme

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

$$odd - bits = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

$$even - bits = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

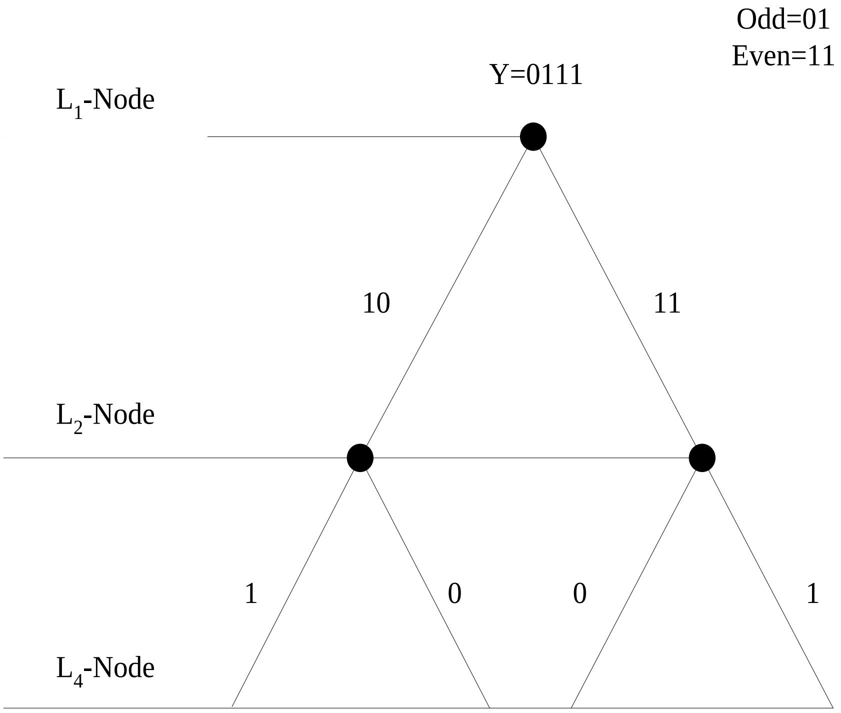

Figure 10: Distribution Scheme

Second step of decoding part is combination. For the first layer L4, The value of L will get by using equation (3.1). By putting the received sequence bit to the formula above, the value of L4-value will be:

$$L4 - value = [1000 \quad 0 \quad 0 \quad 0]$$

Note: large Number (1000) will put instead of infinity to make the solution easier.

After that L2-node should check, there is no bit in L2-node, so the formula 1 of decoding will use to get L2-value:

$$F1 = \frac{(1st * 2nd) + 1}{(1st + 2nd)}$$

Where 1^{st} (first term) and 2^{nd} (second term) represented the values of the previous layer L4-value respectively.

$$L2 - value = [1.0000e - 003 \quad 1.0000e + 009]$$

And the same steps for the last layer L1-value

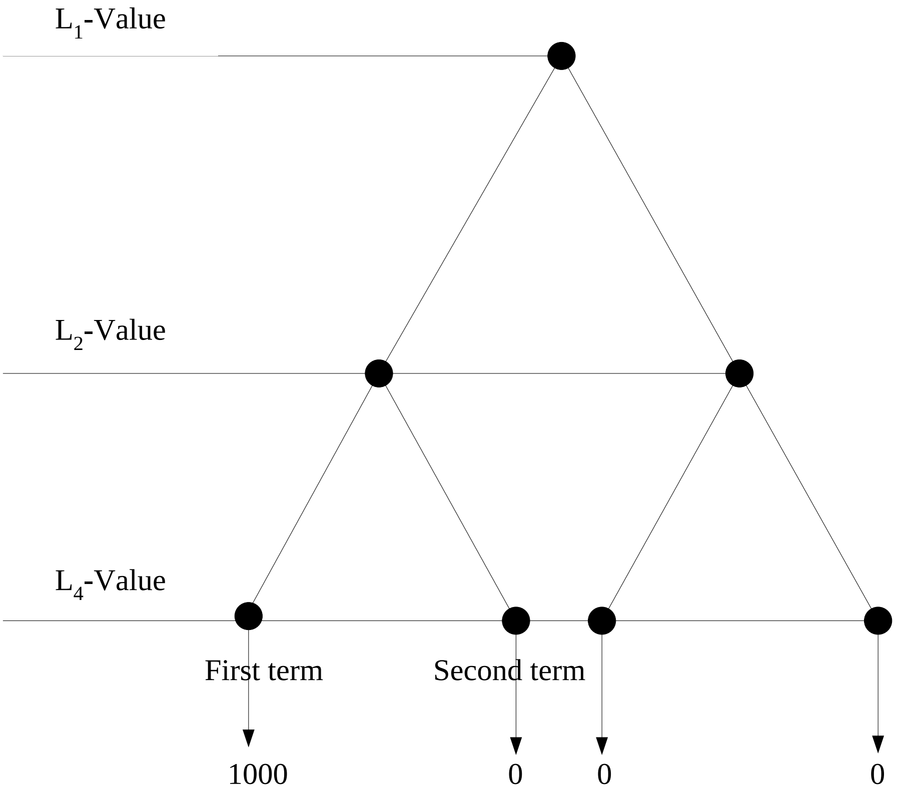

Figure 11: Combination Scheme

The first decoded bit will solve by take the value of the last layer L1-value and put it to the formula of decision:

$$L1\!-\!value = [0.0010]$$

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

So the 1^{st} decoded bit is [1]

To get the second decoded bit, the first decoded bit should distribute

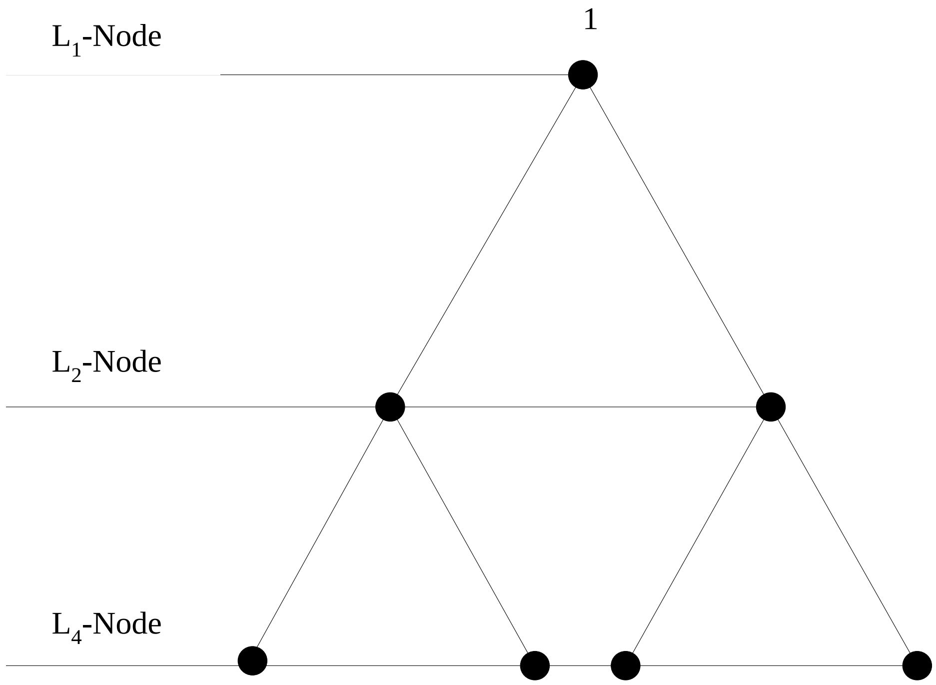

Figure 12: Distribution of 1-bit

L4-value and L2-value will be the same as in the previous steps

$L4\!-\!value = [1000 \quad 0 \quad 0 \quad 0]$

$L2\!-\!value = [1.0000e-003 \quad 1.0000e+009]$

In L1-node there is a bit, so formula (2) should use in this case to get new L1-value, formula (2) is:

$F2 = (1st)^{1-2u} * (2nd)$

Where u=1, is the node-bit (L1-node)

$F2 = (1.0000e-003)^{1-2(1)} * (1.0000e+009)$

$F2 = 1.0000e+012$

L1-value= $F2$

To get 2nd decoded bit ,new L1-value should put to the decision formula :

$$\tilde{u}_i = \begin{cases} 0, & \text{if } L_N^{(i)}(y_1^N, \tilde{u}_1^{i-1}) \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{3.4}$$

So the 2nd decoded bit is [0]

To get the 3rd decoded bit, the 1st and the 2nd decoded bit is distributed.

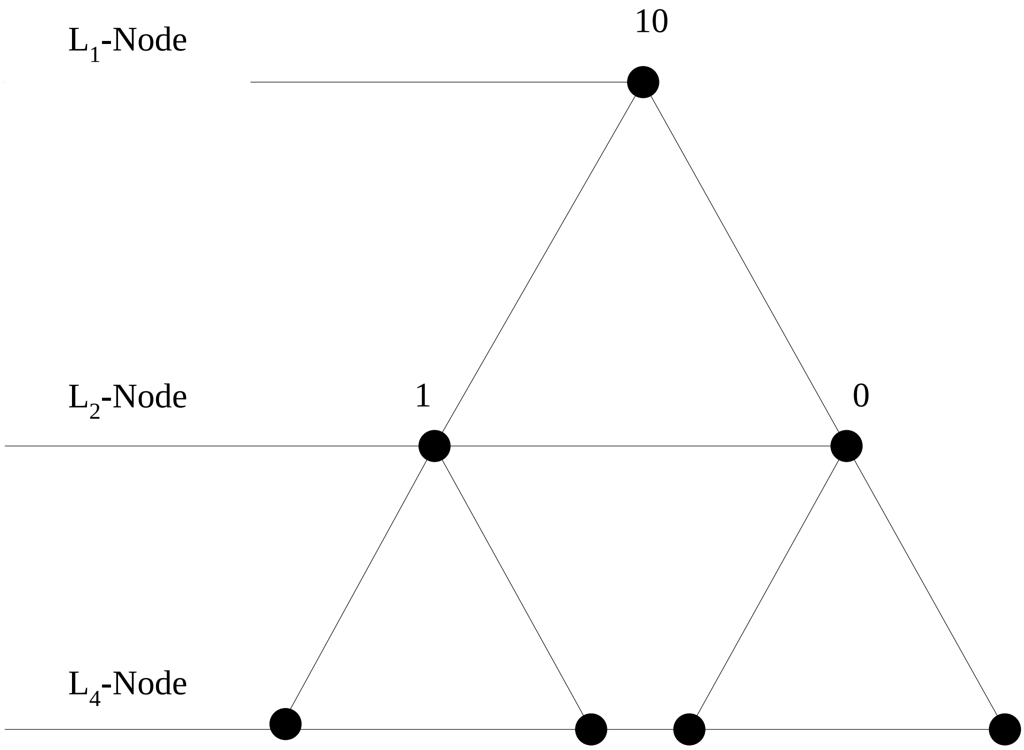

Figure 13: Distribution of 2-bit

L4-value will be the same as in the previous steps

$L4 - value = [1000 \quad 0 \quad 0 \quad 0]$

In L2-node there is a bit, so formula (2) should use in this case to get new L2-value, formula (2) is:

$F2 = (1st)^{1-2u} * (2nd)$

$L2 - value = [0 \quad 0]$

By using formula (1) to get new L1-value:

$$F1 = \frac{(1st * 2nd) + 1}{(1st + 2nd)}$$

$L1 - value = [1.0000e + 009]$

To get 3rd decoded bit , new L1-value should put to the decision formula :

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

So the 3rd decoded bit is [0]

To get the 4th decoded bit, the 1st , 2nd and 3rd decoded bit should distribute

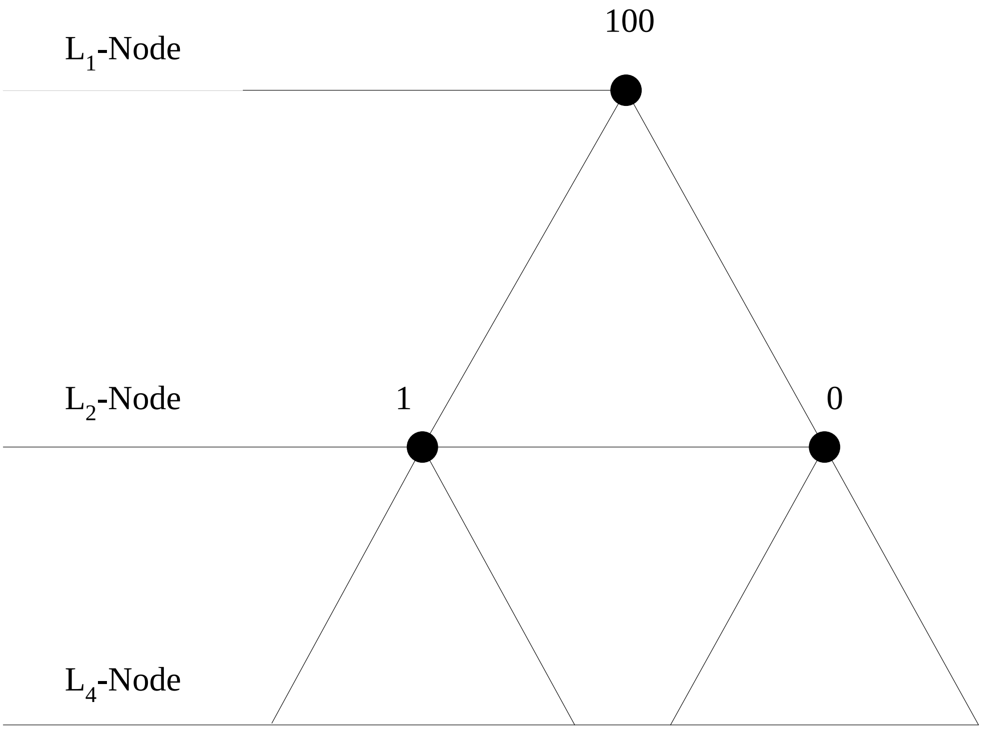

Figure 14: Distribution of 3-bit

L4-value will be the same as in the previous steps

$L4 - value = [1000 \quad 0 \quad 0 \quad 0]$

In L2-node there is a bit, so formula (2) should use in this case to get new L2-value, formula (2) is:

$$F2 = (1st)^{1-2u} * (2nd)$$

$L2-value = \begin{bmatrix} 0 & 0 \end{bmatrix}$

There is a bit in L1-node bit, so by using formula (2) to get new L1-value:

$$F1 = \frac{(1st * 2nd) + 1}{(1st + 2nd)}$$

$L1-value = \begin{bmatrix} 0 \end{bmatrix}$

To get 4^{th} decoded bit ,new L1-value should put to the decision formula

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

So the 3^{rd} decoded bit is [1]

Finally the decoded sequence is [1 0 0 1], same as the data.

3.3. The Simulation Results

In this part, the simulation results shows the performance of polar codes under successive cancellation decoder at block length (2^{10} and 2^{11}) over a BEC (binary erasure channel) and the erasure probability (0.5) as shown in figure (15) below. As seen below that polar codes performance become better when the block length increases from 2^{10} to 2^{11} .

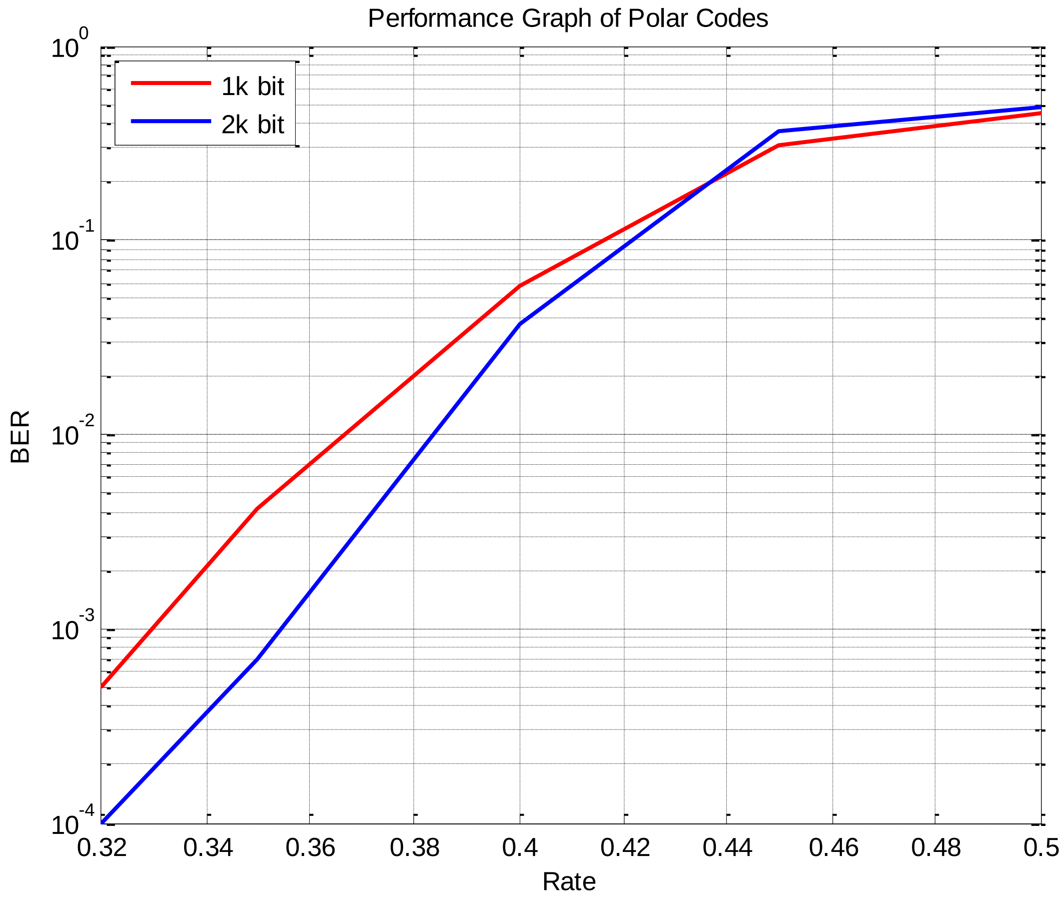

Figure 15: Performance of Polar Codes

CHAPTER 4

CONCLUSION AND FUTURE WORK

4.1 Conclusion

Polar codes invented by Arıkan are capacity achieving codes for binary memoryless channels constructed using the idea of channel polarization. And these type of codes show good performance for long frame lengths which increases the decoding latency of communication systems. This drawback needs to be combated especially for delay sensitive systems, such as real time communication systems.

Polar codes can be decoded using the successive cancellation algorithm. However, the computational amount of this algorithm is significant and smart decoding approaches are needed for the efficient implementation of the successive decoding algorithm for practical systems.

In this thesis work an algorithm is proposed for the successive cancellation decoding of polar codes. The proposed algorithm is friendly for practical implementations and suitable for parallel processing applications. In addition the proposed algorithm can be efficiently implemented in FPGA devices. The proposed algorithm is simulated for binary erasure channels using Matlab programming environment and performance graphs are obtained.

4.2 Future work

The proposed successive cancellation decoding technique can be implemented in hardware, such as on FPGA or DSP chips. And some early termination algorithms considering the node probabilities of successive layers can be studies as future work. In addition, the polarization concept can be extended to other class of channels such as wireless MIMO channels.

REFERENCES

1- **Gallager R., Glavieux A., (2013),** "*Channel Coding in Communication Networks: from Theory to Turbo Codes"*, John Wiley & Sons, vol. 97, pp. 98-112.

2- **Dilpazir H., Mahmood H., Shah T., Malik H., (2011),** "*Source Coding and Channel Coding for Mobile Multimedia Communication*", Mobile Multimedia–User and Technology Perspectives, vol. 97, pp. 98-112.

3- **Gallager R. G., (1968),** *"Information Theory and Reliable Communication",* New York, Wiley, vol. 2, pp.50-83.

4- **Shannon C. E., (1958),** "*A Note on Partial Ordering for Communication Channels*", Information and Control, vol. 1, pp. 390–397.

5- **Kaufman C., Perlman R., Speciner M., (2002),** *"Network Security: Private Communication in a Public World",* Prentice Hall Press, pp.524-532.

6- **Shannon C. E., (2001),** *"A Mathematical Theory of Communication",* ACM SIGMOBILE Mobile Computing and Communications Review, vol. 5, no. 1, pp. 3-55.

7- **Reed I., (1954),** "*A Class of Multiple-Error-Correcting Codes and The Decoding Scheme,"* Information Theory, Transactions of the IRE Professional Group, vol. 4, no. 4, pp. 38-49.

8- **Crooks G. E., Hon G., Chandonia J. M., Brenner S. E., (2004**), *"WebLogo: a Sequence Logo Generator*", Genome Research, vol. 14, no. 6, pp. 1188-1190.

9- **MacKay D. J., (2003),** "*Information Theory, Inference, and Learning Algorithms*" Cambridge University Press, vol. 7, pp. 78-90.

10- **Arıkan E., (2006),** "*Channel Combining and Splitting for Cutoff Rate Improvement",* IEEE Transactions Information Theory, vol. 52, no. 2, pp. 628–639.

11- **Niu K., Chen K., Lin J., Zhang Q. T., (2014).** "*Polar Codes: Primary Concepts and Practical Decoding Algorithms"*. Communications Magazine, IEEE, vol. 52, no. 7, pp. 192-203.

12- **Arıkan E., (2009),** "*Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels*", Information Theory, IEEE Transactions, vol. 55, no. 7, pp. 3051-3073.

13- **Arıkan E., (2008),** "*A Performance Comparison of Polar Codes and Reed-Muller Codes*", IEEE Common Letter, vol. 12, no. 6, pp. 447–449.

14- **Chen K., Niu K., Lin J., (2013),** "*Improved Successive Cancellation Decoding of Polar Codes Communications"* IEEE Transactions, vol. 61, no. 8, pp. 3100-3107.

15- **Arıkan E**., **Telatar E., (2009*),*** *"On the Rate of Channel Polarization"*, In Information Theory. ISIT 2009. IEEE International Symposium, pp. 1493-1495.

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Alibraheemi, Ali

Date and Place of Birth: 11.08.1988

Marital Status: Single

Phone: 05364409141

Email: ali88_fcbarca@yahoo.com

EDUCATION

Degree	Institution	Year of Graduation
M.Sc.	Cankaya University	2015
B.Sc.	Technical College of Engineering Najaf	2010
High School	Najaf High School	2006

WORK EXPERIENCE

Year	Place	Enrollment
2011- Present	Karbala TV	SNG Engineer

FOREIN LANGUAGES
Arabic, English, Turkish.

HOBBIES
Football, Bodybuilding, Traveling.