TECHNICAL UNIVERSITY OF CRETE

School of Electrical and Computer Engineering

Pascal-matrix Polar Coding for Prime-input Channels

By:

Ioannis-Themistoklis Papoutsidakis

Advisor: Associate Professor G. N. Karystinos
Commitee Member: Professor A. P. Liavas
Commitee Member: Dr. D. Toumpakaris

Submitted in partial fulfilment of the requirements for the
Electrical and Computer Engineering diploma degree

July 2016

Abstract

In this thesis, first we present the original polar codes. We describe the basic polarization effect and present an efficient recursive formula to compute the best choices for frozen bits in the case of the binary erasure channel. Both the encoder and the decoder have log-linear complexity. We highlight that, by using the same construction as in binary polar codes, we can create polarized extreme (perfect or useless) channels for any prime-input channel.

Then, we indicate the characteristics of the matrices that achieve channel polarization. We present a strict method that allows us to recursively construct generator matrices based on the Pascal matrix for prime alphabets. We observe their characteristics and properties.

Finally, using the above-mentioned technique, we develop a new ternary kernel and an encoder and successive cancellation decoder with log-linear complexity. We also construct formulas that efficiently calculate the optimal choice of frozen symbols for a ternary erasure channel. It is shown that our construction polarizes the capacities of the channels relatively faster in comparison to the conventional polar construction. The latter is illustrated by considering the error-correction capability of both the conventional Polar code and our proposed code and simulating the symbol error rate for the TEC.

Contents

List of Figures

1 Introduction

The field of channel coding started with Claude Shannon's 1948 landmark paper [1]. The following 50 years, the main task was to find codes that approach or even achieve channel capacity. It was more than once that the research in this field considered dead by many. A typical example is narrated by Robert W. Lucky [2]:

"A small group of us in the communications field will always remember a workshop held in Florida about 20 years ago [1971]... One of my friends [Ned Weldon] gave a talk that has lived in infamy as the coding is dead talk. His thesis was that he and the other coding theorists formed a small, inbred group that had been isolated from reality for too long. He illustrated this talk with a single slide showing a pen of rats that psychologists had penned in a confined space for an extensive period of time. I cannot tell you what those rats were doing, but suffice it to say that the slide has since been borrowed many times to depict the depths of depravity into which such a disconnected group can fall... Give up this fantasy and take up a useful occupation, exhorted my friend. Coding is dead."

Life itself came to deny this statement. The first revolutionary change became in the 90's with Turbo codes. Their performance was incredibly better in relation to the state of the art at the time, that many did not believe the results and the paper got rejected two times! A few years later, a new family of codes came under the spotlight. Low-density parity-check codes were rediscovered in 1996, since they were impractical to implement when first developed by Robert G. Gallagher in 1963.

The advent of these two code families, which approach channel capacity, resumed the perception that the field of coding is saturated. Once again, this opinion was wrong. In 2009, the work by E. Arikan in [3] introduced for the first time a class of codes, namely the Polar codes, that are provably capacity achieving for any binary symmetric memoryless channel. Later, this coding scheme was generalized to arbitrary-input channels [4], [5], [6], [7], [8]. This innovation reignited the coding theory field with many publication and variations of the channel polarization technique. Recently, Reed-Muller (RM) codes were proved to be capacity achieving for the binary erasure channel [9].

Unfortunately, capacity achievability is an asymptotic property and the performance of the original Polar code is disappointing compared to state-of-the-art codes for short to moderate block lengths. Much research work has been done to tackle this problem with relatively good results [10]. However, it seems that we need to change the very nature of the code to improve its performance.

In this work, we are motivated by the fact that two capacity-achieving codes, RM and Polar codes, share the same generator matrix, also known as the Sierpinski triangle in the mathematical society. This indicates that other coding schemes in this generator matrix that perform better may also exist. In addition, it suggests that, by using similar techniques for constructing generator matrices, it may be possible to create new codes for arbitrary-input channels that outperform Polar codes. The latter is the one considered in this thesis.

2 Polarization for Binary-input Channels

Channel polarization was originally proposed in [3] for binary-input discrete memoryless channels as a coding technique that was used for the construction of a new family of codes for data transmission. These codes, called "Polar codes," can achieve the "symmetric capacity" of any binary-input channel by employing low-complexity encoding and decoding algorithms. Later, it was proved that this code construction can achieve the capacity of any prime-input discrete memoryless channel [4]. This section rehearses the construction of binary Polar codes.

2.1 Symmetric Capacity for Binary-input Channels

Given a binary-input channel $W : \mathcal{X} \to \mathcal{Y}$ with $\mathcal{X} = \{0, 1\}$, we define its symmetric capacity as

$$I(W) \doteq \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} \frac{1}{2} W(y|x) \log_2 \frac{W(y|x)}{\frac{1}{2}W(y|0) + \frac{1}{2}W(y|1)}. \tag{1}$$

Symmetric capacity is nothing but the mutual information between the input and the output of the channel when the input is uniformly distributed [3]. Therefore, if the channel is symmetric, then its Shannon capacity is equal to its symmetric capacity. It is known that linear codes produce uniformly distributed codewords and, in the case of symmetric channels, uniform input distribution is needed to maximize mutual information between the transmitter and the receiver.

2.1.1 Binary symmetric channel

The binary symmetric channel (BSC) is shown in Figure 1. Once a bit is transmitted, the receiver obtains the bit either correctly with probability (w.p.) $1 - p$ or inverted w.p. p. It can be proved that the symmetric capacity of the BSC is

$$I(W) = 1 + p \log_2(p) + (1 - p) \log_2(1 - p). \tag{2}$$

The proof is provided in [11].

2.1.2 Binary erasure channel

In our analysis, we mainly use the binary erasure channel, as in Figure 2. In this channel, the transmitted symbols are bits. Once a bit is transmitted, the receiver either obtains the bit correctly w.p. $1 - \varepsilon$ or receives a message that the bit was not received w.p. ε. We choose this type of channel because it is relatively easy to construct algorithms for evaluating the polarized channels, due to the closed-form expression for the BEC capacity. The symmetric capacity of the BEC is

$$I(W) = 1 - \varepsilon. \tag{3}$$

The proof is provided in [11].

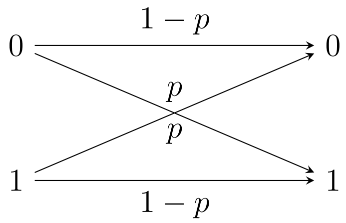


Figure 1: The binary symmetric channel with error probability p.

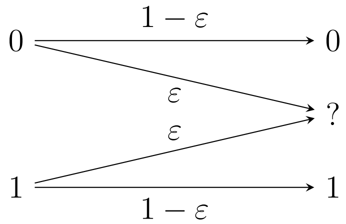


Figure 2: The binary erasure channel with erasure probability ε.

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

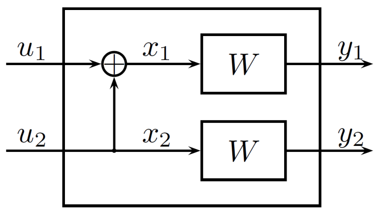
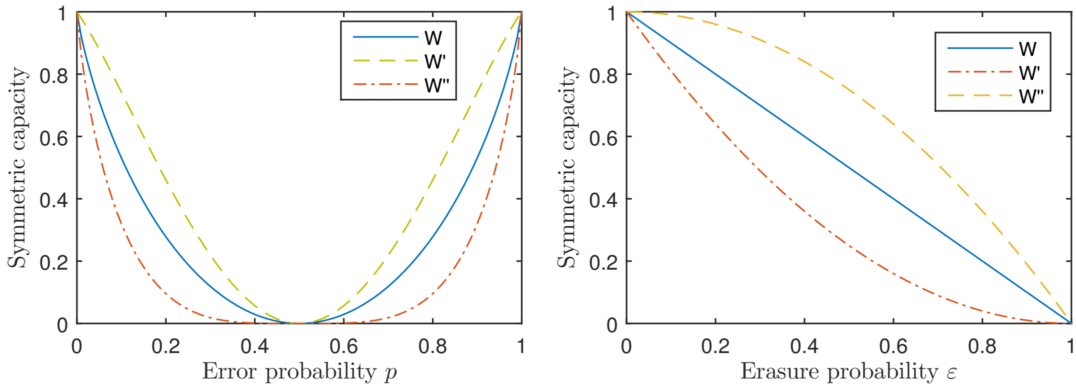


Figure 3: Basic polarization step.

2.2 Basic Construction

Channel polarization is an operation by which, out of N independent copies of a given discrete memory channel W, one manufactures a second set of N channels $\{W_N^{(i)} : 1 \leq i \leq N\}$ that show a polarization effect in the sense that, as N becomes large, the symmetric capacity terms $\{I(W_N^{(i)})\}$ tend towards 0 or 1 for all but a vanishing fraction of indices i [3].

To achieve this effect, we use a linear transformation over GF(2) for combining two identical channels to a new synthetic channel W_2: $\{W, W\} \mapsto \{(y_1^2; u_1), (y_1^2, u_1; u_2)\} \iff \{W, W\} \mapsto \{W', W''\}$. This transformation is shown in Figure 3. As this transformation occurs, the first channel degrades and the second upgrades in terms of symmetric capacity. In Figure 4, the latter is observed for the BEC and BSC.



(a) BSC.

(b) BEC.

Figure 4: Symmetric capacity for the BSC and BEC before and after the basic polarization step.

Having shown how the basic step of original polarization works, we can define the recursion that constructs the generator matrix of Polar codes. As in [3], we define the Kronecker power $G^{\otimes n}$ as $G \otimes G^{\otimes n-1}$ for all n $\geq$ 1,where $\otimes$ denotes the Kronecker product and $G^{\otimes 0} = [1]$.

$$G_N = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{\otimes n}. \tag{4}$$

With this method we can construct $N \times N$ generator matrices, where $N = 2^n$. The next step is to define which rows of G correspond to the perfect channels and which to the useless. In the perfect rows, there will be put information bits, while in the useless rows, there will be put frozen (known to the decoder) bits. This task is easily managed for the BEC. We can recursively calculate the symmetric capacities of the manufactured channels using the following formulas [3].

$$I(W_N^{(2i-1)}) = I(W_{N/2}^{(i)})^2, \tag{5}$$

$$I(W_N^{(2i)}) = 2I(W_{N/2}^{(i)}) - I(W_{N/2}^{(i)})^2. \tag{6}$$

In the case of BEC(0.5), the symmetric capacity limit is at 0.5 bits per channel use. In Figure 5, using (5) and (6), we observe the effect of channel polarization. Indeed, almost half of the channels are perfect and the other half are useless.

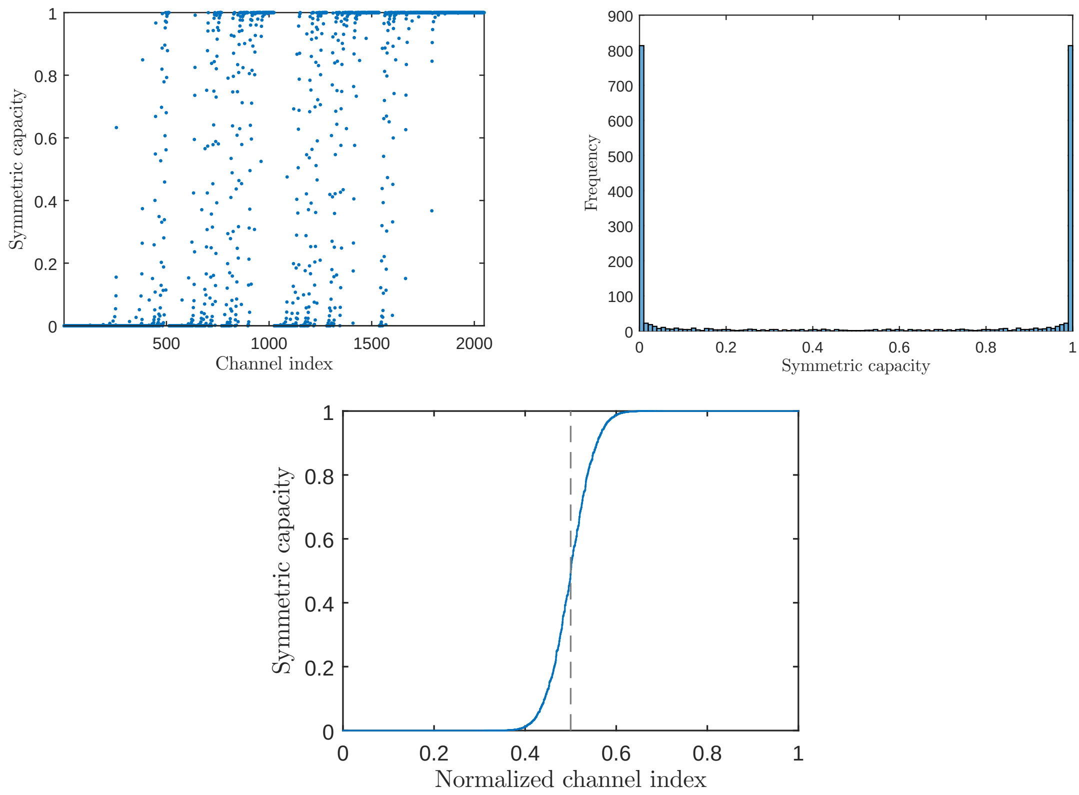


Figure 5: Channel polarization for a BEC with $\varepsilon = 0.5$ and $N = 2^{11}$.

2.3 Encoding

In this section, we consider the implementation of the encoder of Polar codes. Matrix multiplication of G with an information vector is easy for small block lengths but not convenient for bigger block lengths. We design a recursion based on the main channel combination W_2 that has been indicated in the Section 2.2.

The general form of the recursion is shown in Figure 6 where two independent copies of $W_{N/2}$ are combined to produce channel W_N. The operator R_N is a permutation, known as the reverse shuffle operation, and simply separates the odd-indexed from the even-indexed signals. Odd-indexed signals become input to the first copy of $W_{N/2}$ and even-indexed to the second.

In terms of complexity, if we take the complexity of a scalar mod-2 addition as 1 unit and the complexity of the reverse shuffle operation R_N as N units of time we have

$$\begin{aligned} & T(N) = \frac{N}{2} + O(N) + 2T(\frac{N}{2}) \\ \Rightarrow\ & T(N) = O(N \log_2 N). \end{aligned}$$

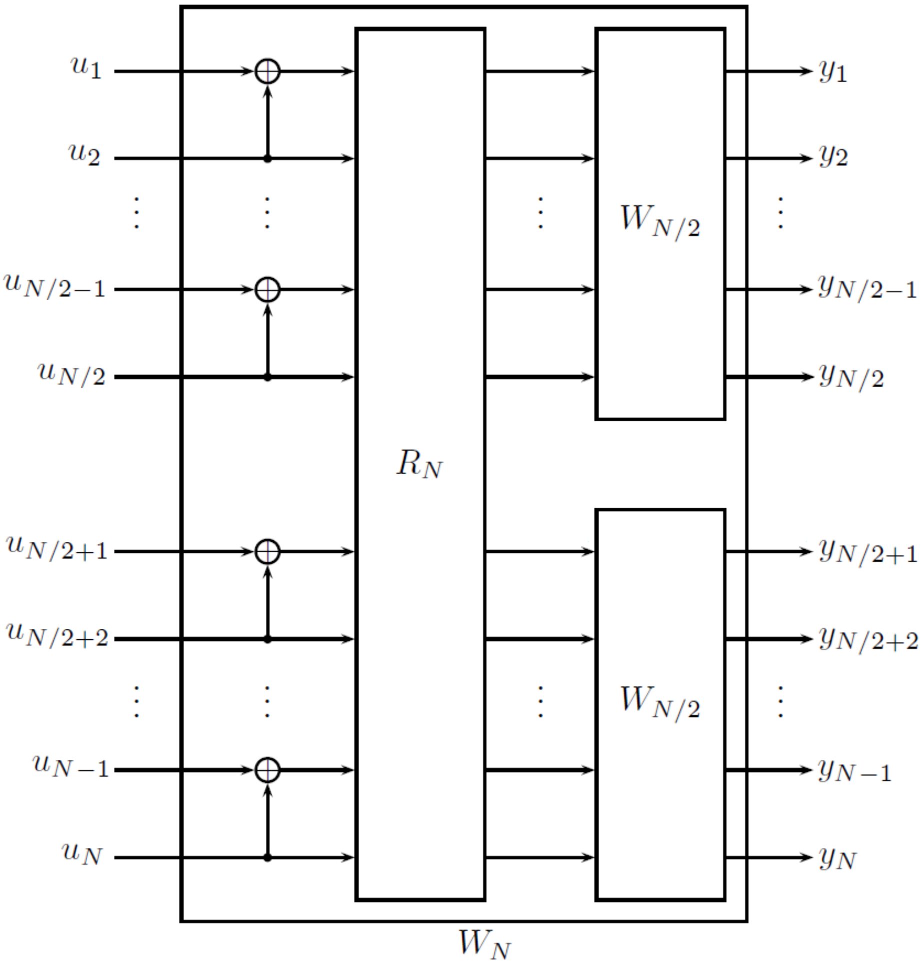


Figure 6: Recursive construction of W_N from two copies of $W_{N/2}$.

2.4 Decoding

The decoder introduced in [3] is called successive cancellation decoder. Its role is to decide with the rule of closest neighbour on the i^{th} symbol ($1 \leq i \leq N$) that is transmitted over $W_N^{(i)}$ by computing

$$\hat{u}_i = \begin{cases} u_i, & \text{when } u_i \text{ is a frozen bit} \\ \arg\max_{x \in \{0,1\}} W_N^{(i)}(y_1^N, u_1^{i-1} | x), & \text{otherwise.} \end{cases} \tag{7}$$

This decoding scheme estimates sequentially every information symbol. Each estimation is carried out by using the knowledge of frozen and previously estimated symbols. We calculate the probabilities of (7) using the recursive formulas (8) and (9).

$$W_{2N}^{(2i-1)}(y_1^{2N}, u_1^{2i-2}|u_{2i-1}) = \sum_{u_{2i}} \frac{1}{2} W_N^{(i)}(y_1^N, u_{1,o}^{2i-2} \oplus u_{1,e}^{2i-2}|u_{2i-1} \oplus u_{2i}) W_N^{(i)}(y_{N+1}^{2N}, u_{1,e}^{2i-2}|u_{2i}), \tag{8}$$

$$W_{2N}^{(2i)}(y_1^{2N}, u_1^{2i-1}|u_{2i}) = \frac{1}{2} W_N^{(i)}(y_1^N, u_{1,o}^{2i-2} \oplus u_{1,e}^{2i-2}|u_{2i-1} \oplus u_{2i}) W_N^{(i)}(y_{N+1}^{2N}, u_{1,e}^{2i-2}|u_{2i}). \tag{9}$$

Every transition probability in this recursion is used over one time. For this reason, we implement a data structure to store these values in order not to calculate them again. We use 2 matrices of size $N \times (\log_2 N + 1)$. Each cell is filled after $\Theta(1)$ calculations, which implies that the complexity of the decoder is $O(N \log_2 N)$.

Since we use the binary alphabet, it is convenient to define the likelihood ratio as

$$L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) = \frac{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1}|0)}{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1}|1)}. \tag{10}$$

This way, the SC decoder is defined as

$$\hat{u}_i = \begin{cases} u_i, & \text{if } u_i \text{ is a frozen bit} \\ 0, & \text{if } L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) \geq 1 \\ 1, & \text{otherwise.} \end{cases} \tag{11}$$

For computing $L_N^{(i)}(y_1^N, \hat{u}_1^{i-1})$, a straightforward calculation using the recursive formulas (8) and (9) gives

$$L_N^{(2i-1)}(y_1^N, \hat{u}_1^{2i-2}) = \frac{L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2}) L_{N/2}^{(i)}(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2}) + 1}{L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2}) + L_{N/2}^{(i)}(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2})}, \tag{12}$$

$$L_N^{(2i)}(y_1^N, \hat{u}_1^{2i-1}) = \left[L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2})\right]^{1-2\hat{u}_{2i-1}} L_{N/2}^{(i)}(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2}). \tag{13}$$

2.5 Performance on the BEC

We consider transmissions of rate-$\frac{1}{2}$ polar-coded bits over the BEC. The block length N is set to 32, 64, 128, 256, 512, and 1024. In Figure 7, we plot the bit error rate as a function of the channel erasure probability ϵ and observe the improvement as the block length grows larger. This happens because perfect channel polarization is accomplished when the block length increases to infinity.

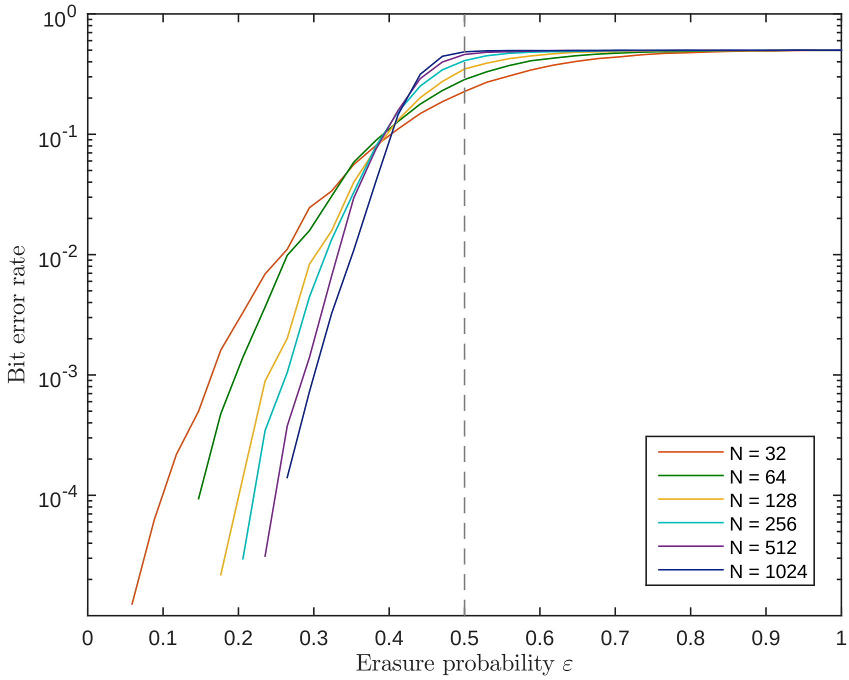


Figure 7: Bit error rate of the rate-$\frac{1}{2}$ polar code with block length $N = 32$, 64, 128, 256, 512, and 1024 on the BEC, as a function of the channel erasure probability ε.

3 Polarization for Prime-input Channels

In [4], it was proved that one can use the aforementioned method to polarize q-ary input memoryless symmetric channels, when q is a prime. In this section, we present the symmetric capacity of q-ary input channels, a ternary channel example, and the minor alterations of the encoder and the decoder.

3.1 Symmetric Capacity for q-ary Channels

Given a q-ary input channel $W : \mathcal{X} \to \mathcal{Y}$ with $\mathcal{X} = \{0, 1, \ldots, q-1\}$, its symmetric capacity is defined as

$$I(W) \doteq \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} \frac{1}{q} W(y|x) \log_q \frac{W(y|x)}{\sum_{x' \in \mathcal{X}} \frac{1}{q} W(y|x')}. \tag{14}$$

To set the scene for our analysis, we present in (14) a formula to compute the symmetric capacity of q-ary input channels [4]. Since we use base-q logarithm, we calculate capacity in q-ary symbols per channel use. Consequently,

$$0 \leq I(W) \leq 1.$$

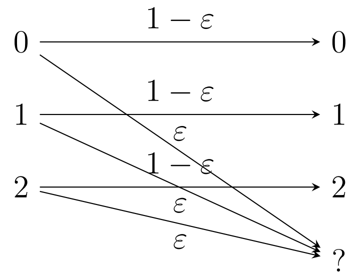


Figure 8: Ternary erasure channel with erasure probability ε.

3.1.1 Ternary symmetric channel

We define the ternary symmetric channel (TSC) with the following conditional probabilities.

$\mathcal{Y}$ \ $\mathcal{X}$	**0**	**1**	**2**
0	$1-p$	$p/2$	$p/2$
1	$p/2$	$1-p$	$p/2$
2	$p/2$	$p/2$	$1-p$

Using (14), in the case that W is a TSC(p), we have

$$I(W) = 1 + p \log_3(\frac{p}{2}) + (1-p)\log_3(1-p). \tag{15}$$

The proof is given in the Appendix.

3.1.2 Ternary erasure channel

The ternary erasure channel (TEC) is shown in Figure 8. This channel transmits ternary information symbols. Once a symbol is transmitted, the receiver either obtains the symbol correctly w.p. $1-\varepsilon$ or receives a message that the symbol was not received w.p. ε. We choose this type of channel to test this coding scheme, because the recursive formulas (5) and (6) apply to the TEC as well. (The proof is given in the Appendix.)

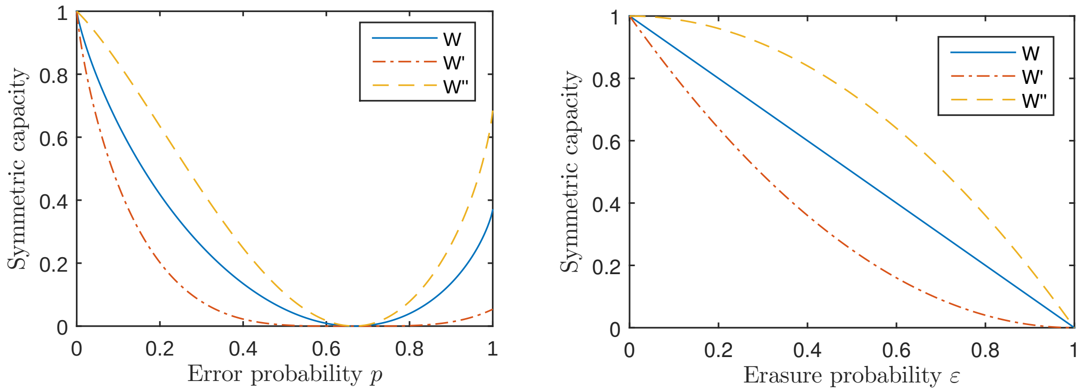
Using (14), in the case that W is a TEC(ε), we have

$$I(W) = 1 - \varepsilon. \tag{16}$$

The proof is given in the Appendix.

3.2 Basic Polarization and Modifications

The effect of the basic polarization step on the TSC and the TEC is shown in Figure 9. We observe that, as mentioned above, polarization of the TEC occurs in exactly the same way as polarization of the BEC. The required modifications are that every



(a) TSC.

(b) TEC.

Figure 9: Symmetric capacity for the TSC and the TEC, before and after the basic polarization step.

operation in the encoder and the decoder is over GF(q) and the decoder generalizes as in (17).

$$\hat{u}_i = \begin{cases} u_i, & \text{when } u_i \text{ is a frozen symbol} \\ \arg\max_{x\in\{0,1,\ldots,q-1\}} W_N^{(i)}(y_1^N, u_1^{i-1}|x), & \text{otherwise,} \end{cases} \tag{17}$$

$$W_{2N}^{(2i-1)}(y_1^{2N}, u_1^{2i-2}|u_{2i-1}) = \sum_{u_{2i}} \frac{1}{q} W_N^{(i)}(y_1^N, u_{1,o}^{2i-2} \oplus u_{1,e}^{2i-2}|u_{2i-1} \oplus u_{2i}) W_N^{(i)}(y_{N+1}^{2N}, u_{1,e}^{2i-2}|u_{2i}), \tag{18}$$

$$W_{2N}^{(2i)}(y_1^{2N}, u_1^{2i-1}|u_{2i}) = \frac{1}{q} W_N^{(i)}(y_1^N, u_{1,o}^{2i-2} \oplus u_{1,e}^{2i-2}|u_{2i-1} \oplus u_{2i}) W_N^{(i)}(y_{N+1}^{2N}, u_{1,e}^{2i-2}|u_{2i}). \tag{19}$$

As far as complexity is concerned, given that the mod-q addition has complexity of 1 unit, the encoding and decoding complexities are both $O(N \log_2 N)$. The likelihood ratio is based on the binary alphabet and does not generalize for larger alphabets.

3.3 Performance on the TEC

We consider transmissions of rate-$\frac{1}{2}$ polar-coded ternary symbols over the TEC. The block length N is set to 32, 64, 128, 256, 512, and 1024. In Figure 10, we plot the symbol error rate as a function of the channel erasure probability ε and, as with the BEC(ε), observe the improvement as the block length grows larger.

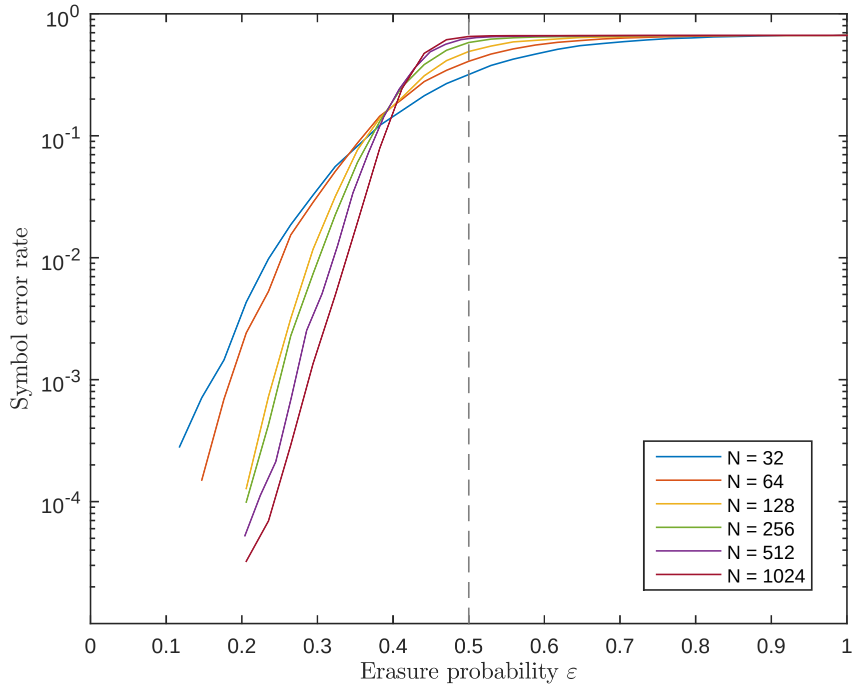


Figure 10: Symbol error rate of the rate-$\frac{1}{2}$ polar code with block length $N = 32$, 64, 128, 256, 512, and 1024 on the TEC, as a function of the channel erasure probability ε.

4 Generalized Constructions

In Section 3, polarization was achieved using the original fixed method of Polar codes. It is proved in [12] that the polarization effect is more common than one would believe. Finding transformations that polarize memoryless channels of prime alphabets becomes an easy task if one completely disregards complexity issues. In fact, almost all invertible matrices polarize such processes. In this section, we present a theorem about polarizing matrices over prime finite fields and then a strict method based on the Pascal matrix to construct them.

4.1 Polarizing Matrices

Theorem 1 *For all prime q, an invertible $\mathbb{F}_q$-matrix is polarizing if and only if it is not upper-triangular.*

Theorem 1 was proved in [12]. It implies that the family of polarizing matrices over prime finite fields is vast. One may therefore hope to find matrices that yield better codes than the original polar codes in terms of their error probabilities. Any good code can be thought of as one which polarizes the given channel. Polarization, if defined as the creation of extremal symmetric capacities from mediocre ones, is then not peculiar to polar codes, but is common to all good codes. The main virtue of polar

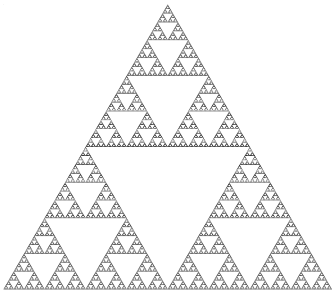


Figure 11: The Sierpinski Triangle.

codes is not that they polarize channels, but that they do so in a recursive way. It is this recursive structure that enables their good performance under low-complexity successive cancellation decoding and made it possible to prove that they are capacity achieving. It is reasonable, then, to restrict the search for methods of polarization to recursive ones.

4.2 Fractals of the Pascal Matrix

Polar codes were the first family of codes that were proven to be capacity-achieving but not the last one. Recently, Reed-Muller codes were declared capacity-achieving for the binary erasure channel [9]. In fact, Polar and RM codes share more than their "capacity achieving" property. They are constructed by the same generator matrix that was presented in Section 2.2. We suspect that this fact is not a coincidence.

As we observe in Figures 11 and 12, this generator matrix is a fractal, also known as the Sierpinski Triangle. This triangle is a fractal in the sense that it reproduces itself on rescaling. This property of fractals is used to generate them by simple reproduction rules. In our case, this reproduction rule is the Kronecker power as described in Section 2.2. Remarkably, one can produce this fractal and many others by using the modulus operation on the Pascal matrix. The Pascal matrix is an infinite matrix containing the binomial coefficients as its elements. It is an alternative formulation of Pascal's triangle, which has been studied since medieval times for the patterns it forms and its properties.

The generator matrix of Polar and RM codes can be produced simply by using the modulo-2 operation on the Pascal matrix. As a matter of fact, one can produce infinitely many fractals by using the following expression.

$$G_N = [\text{Pascal matrix}]_N \mod q \text{ (for prime } q). \tag{20}$$

The above method is not unique [13]. We can construct any fractal that originates from the Pascal matrix using the Kronecker power. We only need to define the kernel matrix F. This is an easy task, since it is possible to obtain the kernel matrix using (21).

$$F = [\text{Pascal matrix}]_q \mod q \text{ (for prime } q). \tag{21}$$

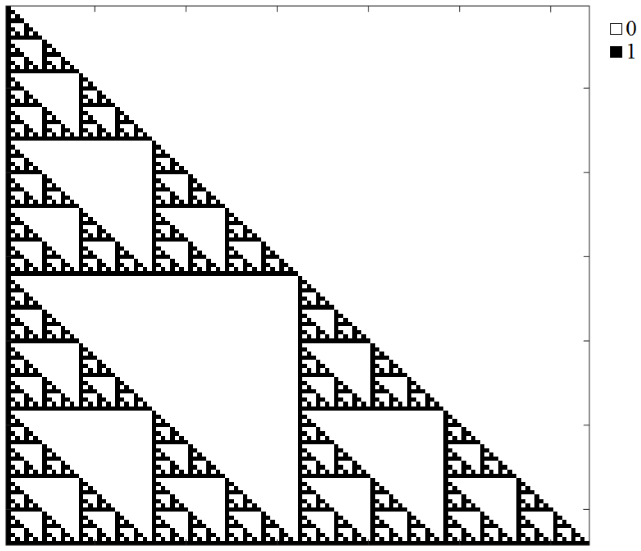


Figure 12: $G_{128}/[\text{Pascal matrix}]_{128} \mod 2$.

$$\begin{bmatrix} 1 & 6 & 21 & 56 & 126 & 252 \\ 1 & 5 & 15 & 35 & 70 & 126 \\ 1 & 4 & 10 & 20 & 35 & 56 \\ 1 & 3 & 6 & 10 & 15 & 21 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Figure 13: The 6×6 Pascal matrix.

This is based on the first of the following observations that we make about the kernel matrix F.

- Its dimensions are always $q \times q$.
- It is always a triangular matrix.
- It is an involutory matrix over GF(q).

The second and third observations hold for any derivative of the kernel matrix F. Using the following method, we can construct any $q^n \times q^n$ fractal, for $n \geq 1$.

$$G_N = F^{\otimes n} \text{ over GF}(q). \tag{22}$$

In [14], the recursive construction of $[\text{Pascal matrix}]_N \mod q^2$ for prime q is studied. We mention this result for future work.

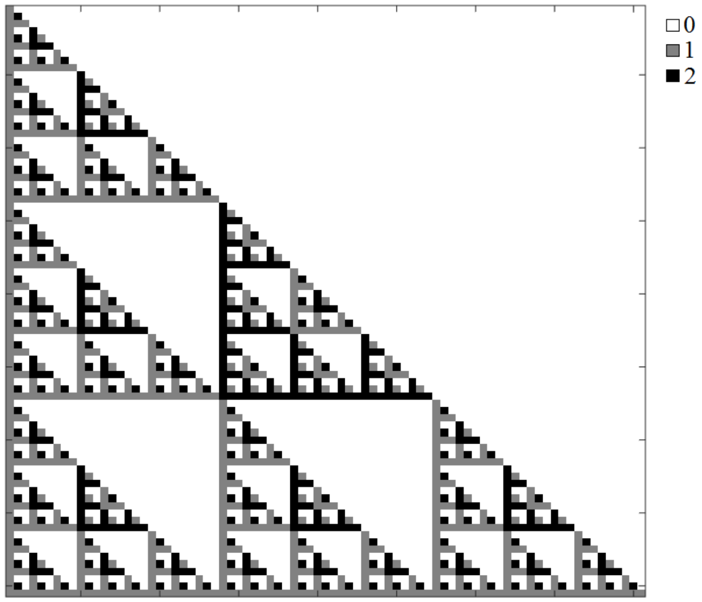


Figure 14: [Pascal matrix]$_{81}$ mod 3.

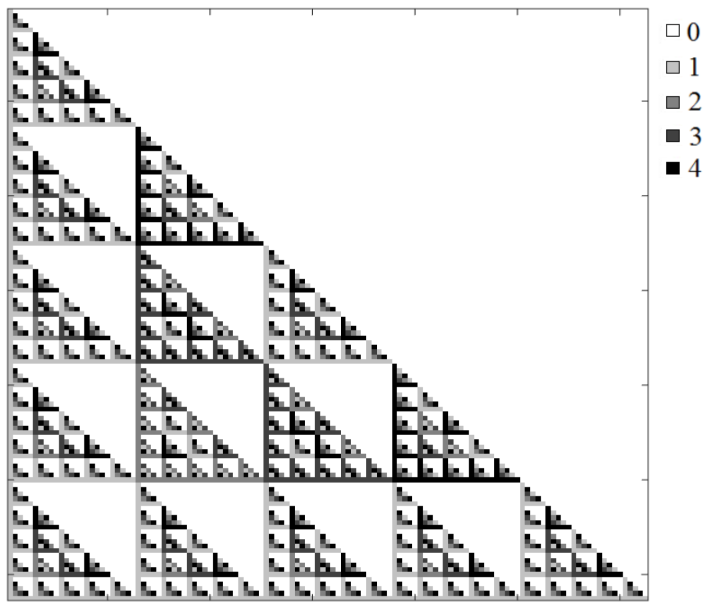
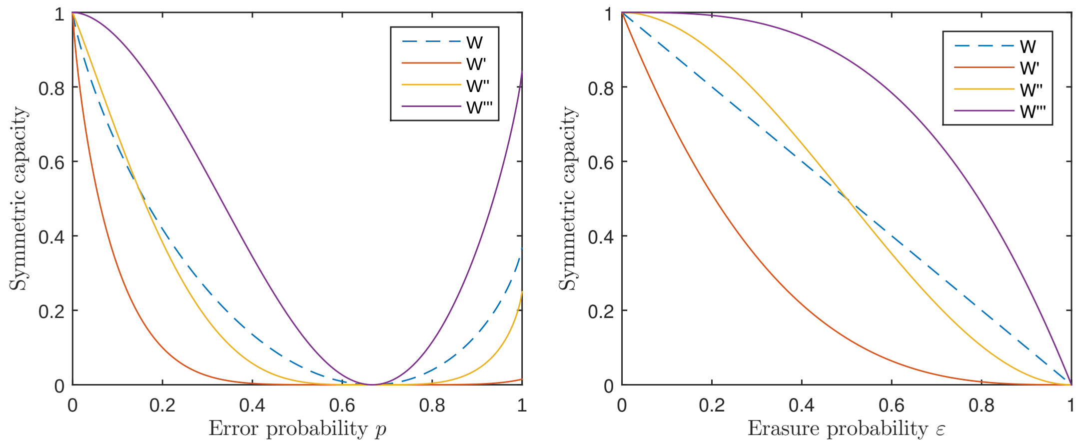


Figure 15: [Pascal matrix]$_{125}$ mod 5.



(a) TSC. (b) TEC.

Figure 16: Symmetric capacity for the TSC and the TEC, before and after the basic polarization step (23).

5 Pascal-matrix Ternary Codes

In this section, we put to test the findings of Section 4. We consider a coding scheme based on $[\text{Pascal matrix}]_N \mod 3$ for the ternary erasure channel. We first study the polarization effect of our transformation. Then, we present a recursive encoder, a SC decoder, and simulation results comparing the performance to that of the original polar codes.

5.1 The Proposed Construction

We construct the kernel matrix of our coding scheme by applying (21) for $q = 3$.

$$F = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 1 & 1 & 1 \end{bmatrix}. \tag{23}$$

The above linear transformation combines three identical channels to a new synthetic channel W_3: $\{W, W, W\} \mapsto \{(y_1^3; u_1), (y_1^3, u_1; u_2^3), (y_1^3, u_1^2; u_3)\} \iff \{W, W, W\} \mapsto \{W', W'', W'''\}$. Then,

$$I(W) = \frac{I(W') + I(W'') + I(W''')}{3}. \tag{24}$$

The proof is given in the Appendix.

In Figure 17, we observe the basic polarization step of our recursive code. What is required next is an efficient method to evaluate the symmetric capacities of the fabricated channels. Again, this task is not complicated for the TEC. We use the following recursive formulas.

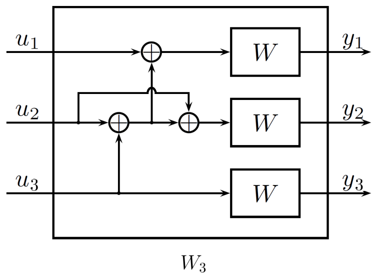


Figure 17: Synthesized channel W_3.

$$I(W_N^{(3i-2)}) = I(W_{N/3}^{(i)})^3, \tag{25}$$
$$I(W_N^{(3i-1)}) = 3I(W_{N/3}^{(i)})^2 - 2I(W_{N/3}^{(i)})^3, \tag{26}$$
$$I(W_N^{(3i)}) = 3I(W_{N/3}^{(i)}) - 3I(W_{N/3}^{(i)})^2 + I(W_{N/3}^{(i)})^3. \tag{27}$$

The proof is given in the Appendix.

In general, absolute polarization comes with infinitely large block length. This phenomenon is witnessed in Figure 18. As the block length grows larger through powers of 3, the fraction of mediocre channels in terms of symmetric capacity shrinks.

At this point, we can make the first comparison of this code with the original one. We cannot carry out the comparison at the same block lengths, so we choose block lengths that are very close, giving an advantage to the original scheme. Despite the fact that the Pascal-matrix ternary polar code has a handicap of almost 6500 symbols, in Figure 19 we observe its dominance over the conventional polar code.

5.2 Encoding

For the encoder, we construct an iteration of the original algorithm for Polar codes, as in Figure 20. This time, we combine three $W_{N/3}$ synthetic channels to construct W_N. R_N shuffles the signals in the following way. It computes the signal index mod 3 and, depending on the result, assigns the signals to $W_{N/3}$'s. The first $W_{N/3}$ corresponds to result 1, the second $W_{N/3}$ corresponds to result 2, and the third $W_{N/3}$ corresponds to result 0.

In terms of complexity, if we take the complexity of a scalar mod-3 addition as 1 unit and the complexity of the reverse shuffle operation R_N as N units of time, we have

$$T(N) = N + O(N) + 3T(\frac{N}{3})$$
$$\Rightarrow T(N) = O(N \log_3 N).$$

We observe that, by using similar iterations, we can construct encoders for any code described in the Section 4 with log-linear complexity.

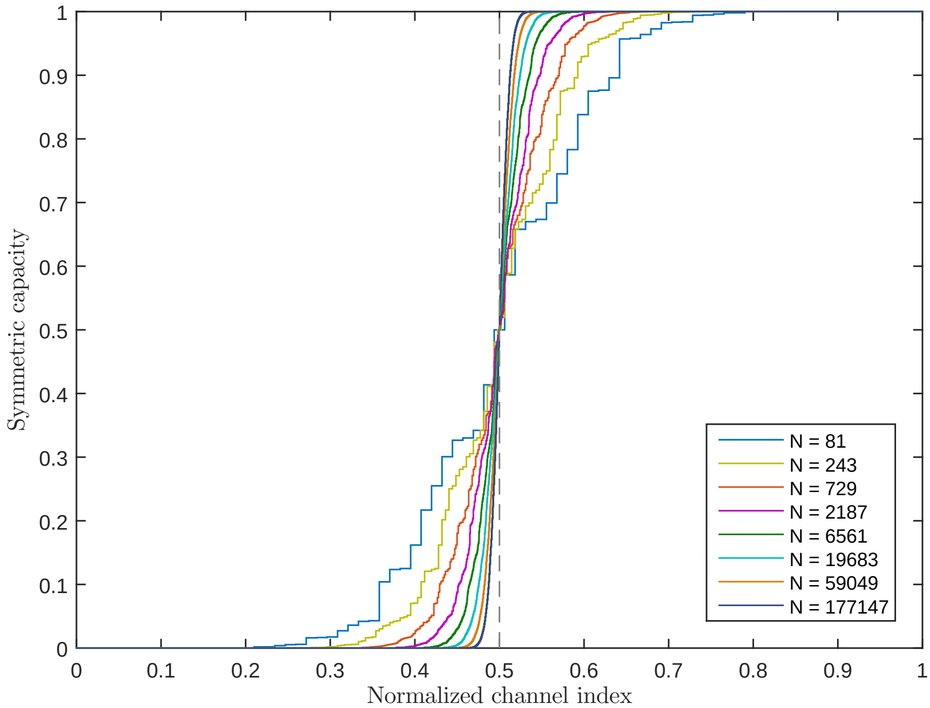


Figure 18: Symmetric capacity of the proposed Pascal-matrix ternary code with block length $N = 3^4 = 81$, $3^5 = 243$, $\ldots$, and $3^{11} = 177147$ as a function of the normalized channel index, for transmissions over a TEC with channel erasure probability $\varepsilon = 0.5$.

5.3 Decoding

We consider a successive cancellation decoder as in the original scheme.

$$\hat{u}_i = \begin{cases} u_i, & \text{when } u_i \text{ is a frozen symbol,} \\ \arg\max_{x\in\{0,1,2\}} W_N^{(i)}(y_1^N, u_1^{i-1}|x), & \text{otherwise.} \end{cases} \tag{28}$$

This time, we calculate the probabilities of (28) by using the recursive formulas (29)-(31). Symbols m_0, m_1, and m_2 stand for index mod 3 equal to 1, 2, and 0, respectively.

$$\begin{aligned} &W_{3N}^{(3i-2)}(y_1^{3N}, u_1^{3i-3}|u_{3i-2}) \\ &= \sum_{u_{3i},u_{3i-1}} \frac{1}{9} W_N^{(i)}(y_1^N, u_{1,m_0}^{3i-3} \oplus u_{1,m_1}^{3i-3} \oplus u_{1,m_2}^{3i-3}|u_{3i-2} \oplus u_{3i-1} \oplus u_{3i}) \\ &\cdot W_N^{(i)}(y_{N+1}^{2N}, 2u_{1,m_1}^{3i-3} \oplus u_{1,m_2}^{3i-3}|2u_{3i-1} \oplus u_{3i}) W_N^{(i)}(y_{2N+1}^{3N}, u_{1,m_2}^{3i-3}|u_{3i}), \end{aligned} \tag{29}$$

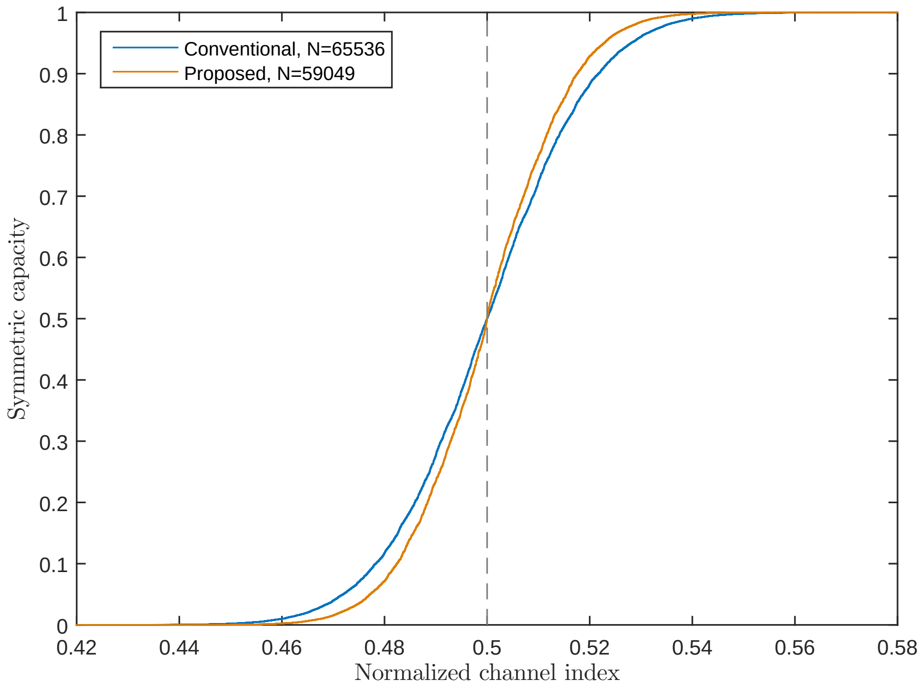


Figure 19: Symmetric capacity of the conventional ternary polar code with block length $N = 2^{16} = 65536$ and the proposed Pascal-matrix ternary polar code with block length $N = 3^{10} = 59049$ as a function of the normalized channel index, for transmissions over a TEC with channel erasure probability $\varepsilon = 0.5$.

$$
\begin{aligned}
&W_{3N}^{(3i-1)}(y_1^{3N}, u_1^{3i-3}|u_{3i-1}) \\
&= \sum_{u_{3i}} \frac{1}{9} W_N^{(i)}(y_1^N, u_{1,m_0}^{3i-3} \oplus u_{1,m_1}^{3i-3} \oplus u_{1,m_2}^{3i-3}|u_{3i-2} \oplus u_{3i-1} \oplus u_{3i}) \\
&\cdot W_N^{(i)}(y_{N+1}^{2N}, 2u_{1,m_1}^{3i-3} \oplus u_{1,m_2}^{3i-3}|2u_{3i-1} \oplus u_{3i}) W_N^{(i)}(y_{2N+1}^{3N}, u_{1,m_2}^{3i-3}|u_{3i}),
\end{aligned}
\tag{30}
$$

$$
\begin{aligned}
&W_{3N}^{(3i)}(y_1^{3N}, u_1^{3i-3}|u_{3i}) \\
&= \frac{1}{9} W_N^{(i)}(y_1^N, u_{1,m_0}^{3i-3} \oplus u_{1,m_1}^{3i-3} \oplus u_{1,m_2}^{3i-3}|u_{3i-2} \oplus u_{3i-1} \oplus u_{3i}) \\
&\cdot W_N^{(i)}(y_{N+1}^{2N}, 2u_{1,m_1}^{3i-3} \oplus u_{1,m_2}^{3i-3}|2u_{3i-1} \oplus u_{3i}) W_N^{(i)}(y_{2N+1}^{3N}, u_{1,m_2}^{3i-3}|u_{3i}).
\end{aligned}
\tag{31}
$$

The proof is given in the Appendix.

In general, the SC decoder is suboptimal. The reason is that, due to the sequential decoding, some information symbols are estimated without utilizing the knowledge of all frozen symbols. In our case, we expect the Pascal-matrix ternary decoder to be closer to the optimal decoder compared to the original scheme, due to the shorter decoding tree. However, this conjecture is not examined in this work.

For this decoder, we use a data structure with three matrices of size $N \times (\log_3 N + 1)$. Each cell is filled after $\Theta(1)$ calculations, which implies that the complexity of

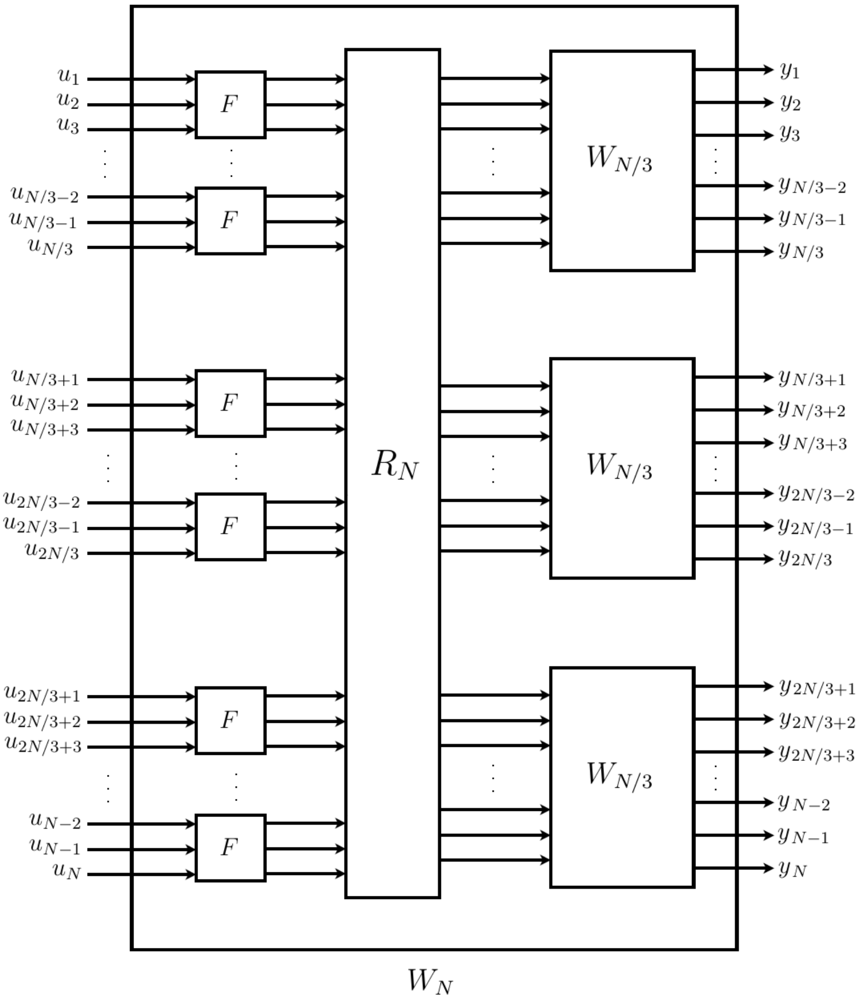


Figure 20: Recursive construction of W_N from three copies of $W_{N/3}$.

the decoder is $O(N \log_3 N)$.

5.4 Performance on the TEC

We consider transmissions of polar-coded ternary symbols over the TEC. To compare the proposed Pascal-matrix ternary polar code with the conventional ternary polar code, the block lengths cannot equal each other due to the construction of each code. Hence, we choose block lengths that are relatively close to each other and give the advantage of longer block to the conventional polar code. We handicap even more the proposed code by transmitting with slightly higher rate. Specifically, for the conventional ternary polar code, we set the block length to $N = 2^8 = 256$ and the rate to $R = 128/256 = 0.5$. For the proposed Pascal-matrix ternary code, we set the block length to $N = 3^5 = 243$ and the rate to $R = 122/243 = 0.502$. That is, the proposed code has both a shorter block length and a higher rate than the conventional one.

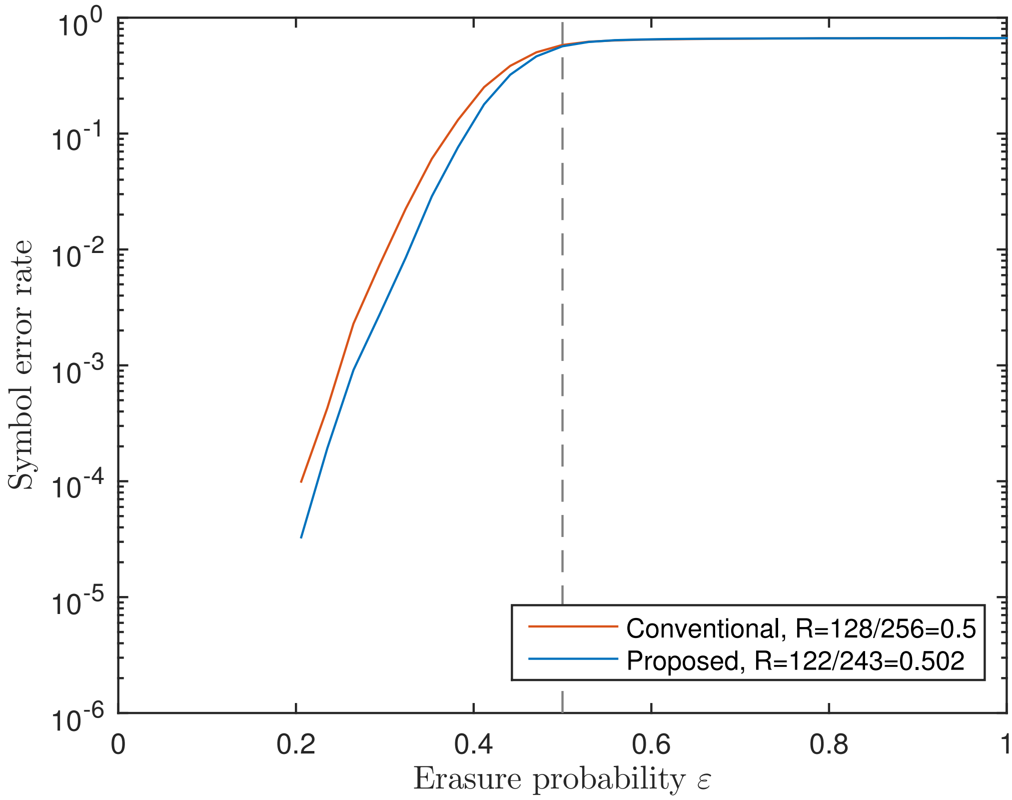


Figure 21: Symbol error rate of the conventional ternary polar code with block length $N = 2^8 = 256$ and rate $R = 128/256 = 0.5$ and the proposed Pascal-matrix ternary polar code with block length $N = 3^5 = 243$ and rate $R = 122/243 = 0.502$ on the TEC, as a function of the channel erasure probability ε.

In Figure 21, we plot the symbol error rate of both codes for transmissions over the TEC, as a function of the channel erasure probability ε. We observe that, although the proposed coding scheme has a slightly shorter block and a slightly higher rate, it is superior to the conventional one. In Figure 22, the proposed Pascal-matrix ternary code remains better than the conventional one, although the former has a considerably shorter block than the latter.

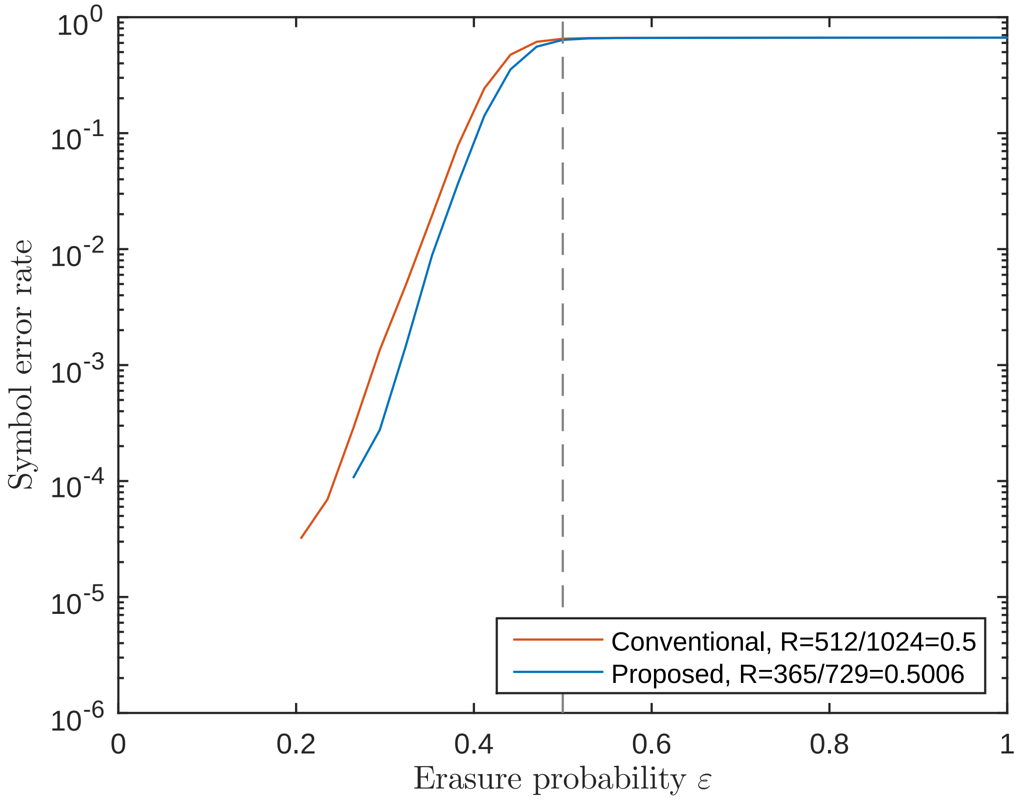


Figure 22: Symbol error rate of the conventional ternary polar code with block length $N = 2^{10} = 1024$ and rate $R = 512/1024 = 0.5$ and the proposed Pascal-matrix ternary polar code with block length $N = 3^6 = 729$ and rate $R = 365/729 = 0.5006$ on the TEC, as a function of the channel erasure probability ε.

A Appendix

Proof of (15), (16), and the recursive formulas of the symmetric capacity of the TEC: Symmetric capacity (14) for $q = 3$:

$$I(W) \doteq \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} \frac{1}{3}W(y|x)\log_3 \frac{W(y|x)}{\frac{1}{3}W(y|0)+\frac{1}{3}W(y|1)+\frac{1}{3}W(y|2)}.$$

If W is the TSC, then

$$\begin{aligned} I(W) =& \sum_{x\in\mathcal{X}} \Big[\frac{1}{3}W(0|x)\log_3(3W(0|x)) + \frac{1}{3}W(1|x)\log_3(3W(1|x)) \\ &+ \frac{1}{3}W(2|x)\log_3(3W(2|x))\Big] \\ =&(1-p)\log_3(3(1-p)) + p\log_3(\frac{3p}{2}) \\ =&(1-p)(\log_3 3 + \log_3(1-p)) + p(\log_3 3 + \log_3\frac{p}{2}) \\ =&1 + (1-p)\log_3(1-p) + p\log_3\frac{p}{2}. \end{aligned}$$

If W is the TEC, then

$$\begin{aligned} I(W) =& \sum_{x\in\mathcal{X}} \Big[\frac{1}{3}W(0|x)\log_3\frac{3W(0|x)}{1-\varepsilon} + \frac{1}{3}W(1|x)\log_3\frac{3W(1|x)}{1-\varepsilon} \\ &+ \frac{1}{3}W(2|x)\log_3\frac{3W(2|x)}{1-\varepsilon} + \frac{1}{3}W(?|x)\log_3\frac{3W(?|x)}{1-\varepsilon}\Big] \\ =&\frac{1}{3}(1-\varepsilon)\log_3\frac{3(1-\varepsilon)}{1-\varepsilon} + \frac{1}{3}(1-\varepsilon)\log_3\frac{3(1-\varepsilon)}{1-\varepsilon} \\ &+ \frac{1}{3}(1-\varepsilon)\log_3\frac{3(1-\varepsilon)}{1-\varepsilon} \\ =&1-\varepsilon. \end{aligned}$$

We have proved that the TEC and the BEC have the same symmetric capacity forms. We observe that, since we deal with erasures, the erasure probability is independent of the cardinality of the channel for the conventional transformation in Figure 3. This means that conventional polar codes polarize exactly the same way the capacities of the synthesized channels. Therefore, (5) and (6) apply to the TEC.

Proof of (24):

$$F = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 1 & 1 & 1 \end{bmatrix}.$$

This transformation is invertible. Therefore,

$$I(U_0^2; Y_0^2) = I(X_0^2; Y_0^2) = 3I(W).$$

Through the chain rule for mutual information,

$$\begin{aligned}
I(U_0^2;Y_0^2) &= I(U_0;Y_0^2) + I(U_1^2;Y_0^2|U_0) \stackrel{U_0,U_1^2 i.i.d.}{=} \\
&= I(U_0;Y_0^2) + I(U_1^2;Y_0^2,U_0) = \\
&= I(U_0;Y_0^2) + I(U_1;Y_0^2,U_0) + I(U_2;Y_0^2,U_0|U_1) \stackrel{U_1,U_2 i.i.d.}{=} \\
&= I(U_0;Y_0^2) + I(U_1;Y_0^2,U_0) + I(U_2;Y_0^2,U_0^1) = \\
&= I(W') + I(W'') + I(W''').
\end{aligned}$$

Then,

$$I(W) = \frac{I(W') + I(W'') + I(W''')}{3}.$$

Proof of (25), (26), and (27):

$$(W,W,W) \mapsto (W',W'',W''')$$

Given $P_e(W) = \varepsilon$, we calculate the erasure probabilities of the synthesized channels.

$$\begin{aligned}
P_e(W') &= \varepsilon^3 - 3\varepsilon^2 + 3\varepsilon, \\
P_e(W'') &= 3\varepsilon^2 - 2\varepsilon^3, \\
P_e(W''') &= \varepsilon^3.
\end{aligned}$$

By using (16) and the above forms we construct formulas (25), (26), and (27).

Proof of (29), (30), and (31): We define the transition probabilities as

$$W_N^{(i)}(y_1^N, u_1^{i-1}|u_i) = \sum_{u_{i-1}^N} \frac{1}{3^{N-1}} W_N(y_1^N|u_1^N).$$

Therefore,

$$\begin{aligned}
W_{3N}^{(3i)}(y_1^{3N}, u_1^{3i-1}|u_{3i}) &= \sum_{u_{3i+1}^{3N}} \frac{1}{3^{N-1}} W_{3N}(y_1^{3N}|u_1^{3N}) \\
&= \sum_{u_{3i-1}^{3N}} \frac{1}{3^{N-1}} W_N(Y_1^N|u_{1,m_0}^{3N} \oplus u_{1,m_1}^{3N} \oplus u_{1,m_2}^{3N}) W_N(Y_{N+1}^{2N}|2u_{1,m_1}^{3N} \oplus u_{1,m_2}^{3N}) \\
&\quad \cdot W_N(Y_{2N+1}^{3N}|u_{1,m_2}^{3N}) \\
&= \frac{1}{9} \sum_{u_{3i+1,m_0}^{3N}} \frac{1}{3^{N-1}} W_N(Y_1^N|u_{1,m_0}^{3N} \oplus u_{1,m_1}^{3N} \oplus u_{1,m_2}^{3N}) \\
&\quad \cdot \sum_{u_{3i+1,m_1}^{3N}} \frac{1}{3^{N-1}} W_N(Y_{N+1}^{2N}|2u_{1,m_1}^{3N} \oplus u_{1,m_2}^{3N}) \\
&\quad \cdot \sum_{u_{3i+1,m_2}^{3N}} \frac{1}{3^{N-1}} W_N(Y_{2N+1}^{3N}|u_{1,m_2}^{3N})
\end{aligned}$$

$$
\begin{aligned}
=&\frac{1}{9}W_N^{(i)}(y_1^N, u_{1,m_0}^{3i-3} \oplus u_{1,m_1}^{3i-3} \oplus u_{1,m_2}^{3i-3} | u_{3i-2} \oplus u_{3i-1} \oplus u_{3i}) \\
&\cdot W_N^{(i)}(y_{N+1}^{2N}, 2u_{1,m_1}^{3i-3} \oplus u_{1,m_2}^{3i-3} | 2u_{3i-1} \oplus u_{3i}) W_N^{(i)}(y_{2N+1}^{3N}, u_{1,m_2}^{3i-3} | u_{3i}).
\end{aligned}
$$

In the cases of $W_{3N}^{(3i-2)}(y_1^{3N}, u_1^{3i-3} | u_{3i-2})$ and $W_{3N}^{(3i-1)}(y_1^{3N}, u_1^{3i-2} | u_{3i-1})$, we use the above formula but handle u_{3i-1}^{3i} and u_{3i}, respectively, as noise.

References

[1] C. E. Shannon, "A mathematical theory of communication," *Bell System Tech. J.*, vol. 27, pp. 379-423, 623-656, July-Oct. 1948.

[2] R. W. Lucky, "Coding is dead," *IEEE Spectrum*, c. 1991. Reprinted in R. W. Lucky, *Lucky Strikes...Again*, pp. 243-245. Piscataway, NJ: IEEE Press, 1993.

[3] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels, *IEEE Trans. Inform. Theory*, vol. IT-55, pp. 3051-3073, July 2009.

[4] E. Şaşoğlu, E. Telatar, and E. Arıkan, "Polarization for arbitrary discrete memoryless channels," in *Proc. 2009 IEEE Inform. Theory Workshop*, Taormina, Italy, Oct. 2009, pp. 144-148.

[5] W. Park and A. Barg, "Polar codes for q-ary channels, $q = 2^r$," *IEEE Transactions on Information Theory*, vol. 59, pp. 955-969, Sept. 2012.

[6] R. Mori and T. Tanaka, "Channel polarization on q-ary discrete memoryless channels by arbitrary kernels," in *Proc. 2010 IEEE Intern. Symp. Inform. Theory*, Austin, TX, June 2010, pp. 894-898.

[7] R. Mori and T. Tanaka, "Source and channel polarization over finite fields and Reed-Solomon matrices," *IEEE Trans. Inform. Theory*, vol. 60, pp. 2720-2736, May 2014.

[8] I. Tal, "On the construction of polar codes for channels with moderate input alphabet sizes," *arXiv:1506.08370*, 2015.

[9] S. Kudekar, S. Kumar, M. Mondelli, H. D. Pfister, E. Şaşoğlu, and R. Urbanke, "Reed-Muller codes achieve capacity on erasure channels," *arXiv:1601.04689*, Jan. 2016.

[10] I. Tal and A. Vardy, "List decoding of polar codes," in *Proc. 2011 IEEE Intern. Symp. Inform. Theory*, Saint Petersburg, Russia, Aug. 2011, pp. 1-5.

[11] T. M. Cover and J. A. Thomas, Elements of Information Theory, 2nd Edition, ISBN: 978-0-471-24195-9.

[12] E. Şaşoğlu, Ph.D. Thesis, pp. 47-50, Switzerland, 2011.

[13] D. Doan, T. Sykes, and J. Smith, The Modular Pascal's Triangle, *http://orion.math.iastate.edu/reu/oldREU/modupasc.htm.*

[14] D. Doan, B. Kivunge, J. J. Poole, J. D. H. Smith, T. Sykes, and M. Teplitskiy, "Partial semigroups and primality indicators in the fractal generation of binomial coefficients to a prime square modulus," *DEMONSTRATIO MATHEMATICA*, Vol. XLII, no. 1, 2009.