New Construction and Performance Analysis of Polar Codes over AWGN Channels

Bashar Tahir, and Markus Rupp
Institute of Telecommunications
Technische Universität (TU) Wien
Vienna, Austria
Email: {bashar.tahir, markus.rupp}@tuwien.ac.at

***Abstract*—We propose a new construction algorithm for Polar codes operating over Additive White Gaussian Noise channels under Successive Cancellation decoding. Our approach is based on tracking the bit error probabilities of the bit channels as they evolve through the decoder, allowing us on the one hand to characterize the performance of these channels, and on the other hand, providing a solid construction algorithm. We then use our approach to derive a modification for the density evolution (with Gaussian approximation) based construction, providing better accuracy and implementation.**

I. Introduction

Polar codes attracted a lot of attention since they were introduced by Arıkan in 2008 [1]. They are the first practical codes that are proven to achieve the channel capacity at infinite length. The structure of polar codes is based on the channel polarization transform, where a specific dependency between the input bits is introduced. Such dependency can be exploited using a Successive Cancellation (SC) based decoder, which causes subsets of the bits to have improved reliability when the successively fedback bits are correct. In this case, the task of constructing polar codes involves finding the most reliable bits' positions (bit channels) that gain the most from the successive decoding, and use them for the transmission of information bits. While foreknown bits (usually zeros) are transmitted over the other unreliable channels (*Frozen* set), ensuring correct feedback of the those bits since they are known by the receiver, thus producing the coding gain.

Polar codes are non-universal, in the sense that their construction is dependent on the Signal-to-Noise Ratio (SNR) of the receiver. Such property is actually a drawback, since the optimum transmission scheme would then require an adaptive construction based on the SNR. Therefore, it's no surprise that the SNR is a design parameter when it comes to the construction algorithms of polar codes.

The first construction algorithm was introduced by Arıkan, where the *Bhattacharyya* parameters of the bit channels are evolved through the polarization structure. A construction based on Density Evolution with Gaussian Approximation (DEGA) is given in [2], which is based on tracking (evolving) the density functions of the Log-Likelihood Ratios (LLRs) through the decoder. Some other constructions are given in [3]–[6], with varying performance and complexity.

In this paper, we propose a new construction algorithm for polar codes operating over Additive White Gaussian Noise (AWGN) channels under SC decoding, based on tracking the bit error probabilities of the bit channels as they evolve through the decoder, providing also a performance characterization for those channels. Our results show that the new algorithm delivers similar performance to that of DEGA, and because the two algorithms are connected through the bit error probabilities, we derive a modification for DEGA, improving its accuracy and implementation. We finally benchmark their performance.

II. Probability Analysis

In this section, we perform the probability analysis for the successive cancellation decoder. Fig. 1 shows the SC decoder for a polar code of length 4.

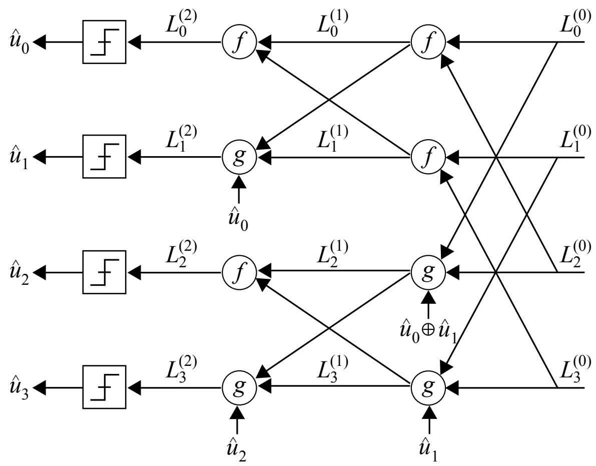

Fig. 1. Polar decoder of length 4.

In the LLR domain, the nodes f and g perform the following calculations for given input LLRs L_a and L_b

$$f(L_a, L_b) = \log\left(\frac{e^{L_a+L_b}+1}{e^{L_a}+e^{L_b}}\right), \quad (1)$$
$$g(L_a, L_b, u_s) = (-1)^{u_s} L_a + L_b,$$

where u_s is the *Partial Sum*, which is the sum of the previously decoded bits that are fedback to the current node g.

The financial support by the Austrian Federal Ministry of Science, Research and Economy and the National Foundation for Research, Technology and Development is gratefully acknowledged.

978-1-5386-0643-8/17/$31.00 ©2017 IEEE

In order to simplify the derivation, we assume that the input bits to the encoder are all zeros, and the signaling used is Binary Phase Shift Keying (BPSK).

Let's look at the first bit u_0. It will be decoded correctly if its LLR $L_0^{(2)}$ is positive. This is satisfied if and only if the two input LLRs $L_0^{(1)}$ and $L_1^{(1)}$ to its node f, are both positive or both negative. In either case, the output will be positive and u_0 is decoded correctly. We also notice that the two inputs to any node in Fig. 1 are processed by the same sequence of nodes in the previous stages, and therefore their statistical properties are the same. Denoting $P_C\{b_i^{(s)}\}$ as the probability of correct decision of the bit channel i at stage s, and similarly $P_E\{b_i^{(s)}\}$ as the probability of error, the probability of decoding u_0 correctly can be calculated as

$$P_C\{b_0^{(2)}\} = P_C\{b_0^{(1)}\}P_C\{b_1^{(1)}\} + P_E\{b_0^{(1)}\}P_E\{b_1^{(1)}\}, \quad (2)$$

i.e. u_0 will be decoded correctly if the inputs to its node f are both correct or both incorrect. Since the two inputs have the same statistical properties as discussed above, then

$$P_C\{b_0^{(2)}\} = P_C\{b_0^{(1)}\}^2 + P_E\{b_0^{(1)}\}^2. \quad (3)$$

Under our assumption of all-zero transmission, we now notice that the output of the channel $b_0^{(1)}$ will be positive if and only if the two inputs to its corresponding node f, are both positive or both negative, which is the same case considered above. Based on this, we generalize (3) to

$$\begin{aligned} P_C\{b_i^{(s)}\} &= P_C\{b_i^{(s-1)}\}^2 + P_E\{b_i^{(s-1)}\}^2, \\ &= (1 - P_E\{b_i^{(s-1)}\})^2 + P_E\{b_i^{(s-1)}\}^2, \\ &= 1 - 2P_E\{b_i^{(s-1)}\} + 2P_E\{b_i^{(s-1)}\}^2, \end{aligned} \quad (4)$$

and the probability of error is then

$$\begin{aligned} P_E\{b_i^{(s)}\} &= 1 - P_C\{b_i^{(s)}\}, \\ &= 2P_E\{b_i^{(s-1)}\} - 2P_E\{b_i^{(s-1)}\}^2, \\ &= 2P_E\{b_i^{(s-1)}\}(1 - P_E\{b_i^{(s-1)}\}). \end{aligned} \quad (5)$$

We consider now the decoding of bit u_1. Under the all-zero transmission, the output of its node g is equal to

$$L_1^{(2)} = L_0^{(1)} + L_1^{(1)}. \quad (6)$$

Unfortunately, this requires invoking the underlying distribution of the LLRs, which is no longer Gaussian due to the non-linear transformations by the previous f nodes. At this point, we drop optimality, and similarly to DEGA, we approximate them with Gaussian densities. Consider the Bhattacharyya parameter of the Binary AWGN channel with variance σ^2

$$Z = \exp(-1/2\sigma^2) = \exp(-\text{SNR}/2). \quad (7)$$

If the partial sum u_s is correct, the parameter Z evolves to Z^2 when the bit channel is transformed by a node g [1], i.e.

$$Z_i^{(s)} = (Z_i^{(s-1)})^2 = \exp(-\text{SNR}_i^{(s-1)}). \quad (8)$$

In other words, under correct feedback, node g (in the Gaussian sense) improves the SNR of the transformed bit channel by a factor of two. We now proceed and calculate the probability of error based on the SNRs. For AWGN channels, the tail probability is given by the Q-function

$$P_E\{b_i^{(s)}\} = Q\left(\sqrt{\text{SNR}_i^{(s)}}\right), \quad (9)$$

in which the calculation of SNR follows directly as

$$\text{SNR}_i^{(s)} = Q^{-1}\left(P_E\{b_i^{(s)}\}\right)^2, \quad (10)$$

where the $Q^{-1}(.)$ is the inverse Q-function. As shown in (8), the SNR will double, and therefore the probability of error at the next stage after passing through node g is given by

$$P_E\{b_i^{(s)}\} = Q\left(\sqrt{2\text{SNR}_i^{(s-1)}}\right), \quad (11)$$

which can be expanded using (10) into

$$P_E\{b_i^{(s)}\} = Q\left(\sqrt{2}\ Q^{-1}\left(P_E\{b_i^{(s-1)}\}\right)\right). \quad (12)$$

Summarizing, the bit error probability of the ith bit channel evolves from stage $(s-1)$ to (s) according to

$$P_E\{b_i^{(s)}\} = \begin{cases} 2P_E\{b_i^{(s-1)}\}(1 - P_E\{b_i^{(s-1)}\}), & \text{if node } f, \\ Q\left(\sqrt{2}\ Q^{-1}\left(P_E\{b_i^{(s-1)}\}\right)\right), & \text{if node } g. \end{cases}$$

III. The New Construction Algorithm

Armed with the results from the previous section, we evolve the bit error probabilities until the last stage is reached, and then we choose the channels with lowest probability of error to transport the information bits.

Due the special structure of the polarization transform, we have seen that at each stage, subsets of the bit channels have the same statistical properties, meaning that for a polar code of length N, we do not need to track N probabilities of error at each stage, but as can be seen, the number increases as a power of two. This allow us to fold the calculation of the bit error probability into the simplified form given in Algorithm 1 below.

Algorithm 1 The new construction algorithm

Input: Target SNR (TarSNR) in dB, Code length (N)
Output: Vector of unsorted bit error probabilities (P)
Initialization: $P[0] = Q(\sqrt{10^{\text{TarSNR}/10}})$

```
for i = 1 to log_2(N) do
    j = 2^(i-1)
    for k = 0 to j - 1 do
        p = P[k]
        P[k] = 2p(1 - p)
        P[k + j] = Q(√2 Q^{-1}(p))
    end for
end for
return P
```

The output vector P can be sorted as desired, and depending on the encoder/decoder implementation, bit reversal sorting might also be needed. The SNR can be represented in terms of the ratio E_b/N_0, and the Q-function can be reformulated in terms of the complementary error function.

Now, we would like to check how good is our approximation of the bit error probabilities. For that, we run a Monte-Carlo simulation with 10^9 repetitions for a polar code of length 16, and estimate the average Bit Error Ratio (BER) of the individual bit channels (positions). During that, we make sure that all the feedback bits are correct. The results are then compared against our algorithm. This is shown in Fig. 2.

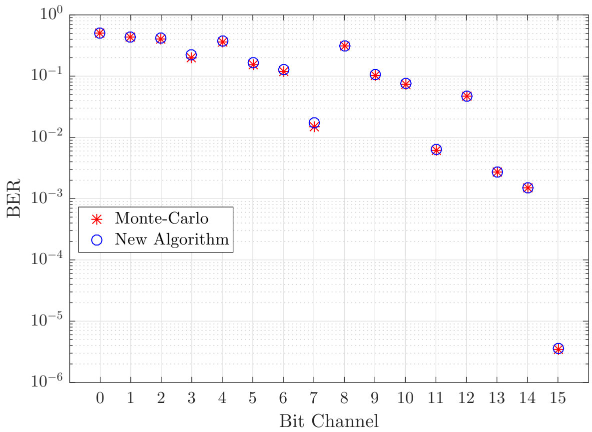

Fig. 2. Estimated BER using Monte-Carlo simulation vs. bit error probabilities approximated by the new algorithm.

The SNR is set to 1 dB here. We can see that our algorithm did very well in approximating the bit error probabilities. Of course, the results might vary depending on the SNR and the code length.

IV. The Modified DEGA

In this section, we use the probability analysis to derive a modification for DEGA. We call the resultant algorithm Modified DEGA (M-DEGA). In DEGA, the mean value of the LLRs is evolved through the decoder, and similarly, we have two transforms which we summarize in the following [2]

$$m_i^{(s)} = \begin{cases} \phi^{-1}\left(1-\left(1-\phi\left(m_i^{(s-1)}\right)\right)^2\right), & \text{if node } f, \\ 2m_i^{(s-1)}, & \text{if node } g, \end{cases}$$

where $m_i^{(s)}$ is the mean value of the LLR $L_i^{(s)}$. The function $\phi(.)$ is given by the approximation [7]

$$\phi(x) = \begin{cases} \exp(-0.4527x^{0.86}+0.0218), & 0 < x < 10, \\ \sqrt{\frac{\pi}{x}}\exp\left(-\frac{x}{4}\right)\left(1-\frac{10}{7x}\right), & x \geq 10, \end{cases}$$

and the inverse function $\phi^{-1}(.)$ might be approximated numerically with a piecewise function.

The LLR of the Binary AWGN channel is equal to

$$L = 2y/\sigma^2, \tag{13}$$

where y is the channel output. Under the assumption of all-zero transmission, the mean value of the LLR above

$$m = E\{L\} = 2/\sigma^2, \tag{14}$$

and the variance

$$var\{L\} = 4/\sigma^2 = 2m. \tag{15}$$

This relationship between the mean and the variance is assumed to hold at all stages [8], i.e. at any stage we have a Gaussian distributed LLR with mean $m_i^{(s)}$ and variance $2m_i^{(s)}$. The SNR can then be calculated as

$$\text{SNR}_i^{(s)} = \frac{(m_i^{(s)})^2}{2m_i^{(s)}} = \frac{m_i^{(s)}}{2}. \tag{16}$$

Using (10), we get

$$m_i^{(s)} = 2Q^{-1}\left(P_E\{b_i^{(s)}\}\right)^2. \tag{17}$$

We've shown in the probability analysis that passing through node f evolves the error probability as in (5), i.e.

$$m_i^{(s)} = 2Q^{-1}\left(2P_E\{b_i^{(s-1)}\}(1-P_E\{b_i^{(s-1)}\})\right)^2, \tag{18}$$

with

$$P_E\{b_i^{(s-1)}\} = Q\left(\sqrt{\frac{m_i^{(s-1)}}{2}}\right). \tag{19}$$

Summarizing, the modified DEGA is given by

$$m_i^{(s)} = \begin{cases} 2Q^{-1}\left(2P_E\{b_i^{(s-1)}\}(1-P_E\{b_i^{(s-1)}\})\right)^2, & \text{if } f, \\ 2m_i^{(s-1)}, & \text{if } g, \end{cases}$$

with $P_E\{b_i^{(s-1)}\}$ given in (19). Therefore, node f transformation is now performed by the well known Q-function and its inverse. It is also possible to complete the square inside the Q^{-1} function and obtain a form similar to the original DEGA.

V. Performance Benchmark

We benchmark the performance of the two new algorithms against DEGA. We transmit with BPSK, for code lengths of $N = 512$ and 4096 at a code rate of $R = 1/2$. On each SNR point, the polar code is reconstructed, and the current SNR point is used as the TargetSNR in the construction algorithm. The results are shown in Fig. 3. It can be seen that the three algorithms are performing similar to each other. However, there is a slight improvement in the low SNR region compared to DEGA.

VI. Conclusion

In this paper, we propose a new construction algorithm for polar codes operating over AWGN channels, acting also as a performance analysis tool. We use our approach to derive a modification for DEGA, which is more accurate and better implementable. The three algorithms perform close to each other, with the new ones having a slight gain in the low SNR regime.

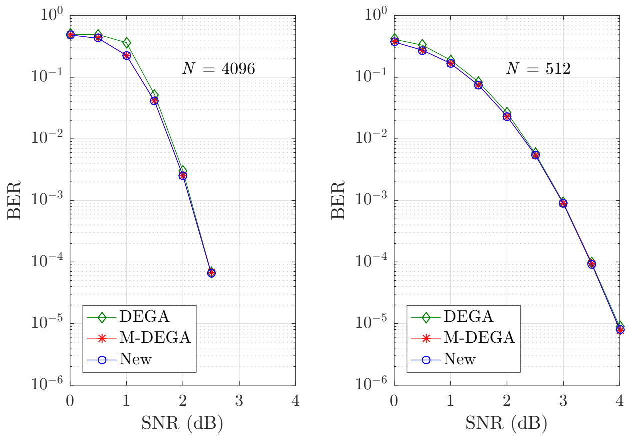

Fig. 3. BER comparison between DEGA, M-DEGA, and the new algorithm for code lengths of 512, and 4096, at a code rate of 1/2.

References

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

[2] P. Trifonov, "Efficient design and decoding of polar codes," *IEEE Transactions on Communications*, vol. 60, no. 11, pp. 3221–3227, November 2012.

[3] I. Tal and A. Vardy, "How to construct polar codes," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6562–6582, Oct 2013.

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

[5] D. Kern, S. Vorkper, and V. Khn, "A new code construction for polar codes using min-sum density," in *2014 8th International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*, Aug 2014, pp. 228–232.

[6] Y. Zhang, A. Liu, K. Pan, C. Gong, and S. Yang, "A practical construction method for polar codes," *IEEE Communications Letters*, vol. 18, no. 11, pp. 1871–1874, Nov 2014.

[7] S.-Y. Chung, T. J. Richardson, and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a gaussian approximation," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 657–670, Feb 2001.

[8] D. Wu, Y. Li, and Y. Sun, "Construction and block error rate analysis of polar codes over awgn channel based on gaussian approximation," *IEEE Communications Letters*, vol. 18, no. 7, pp. 1099–1102, July 2014.