Design of Randomized Polar Subcodes with Non-Arikan Kernels

Peter Trifonov
Saint Petersburg Polytechnic University
Email: petert@dcn.icc.spbstu.ru

***Abstract*—A method for construction of polar subcodes with generic kernels is proposed. The proposed approach relies on a set of randomized dynamic freezing constraints, which eliminate almost all non-zero low-weight codewords. Simulation results for codes with some 16×16 kernels are provided, which show that the obtained codes under SCL decoding with sufficiently large list size require lower decoding complexity compared to polar subcodes with Arikan kernel with the same performance.**

I. Introduction

Polar codes is a novel class of capacity-achieving codes, which admit low-complexity construction, encoding and decoding [1]. However, their finite-length performance is quite bad compared to known LDPC and turbo codes. The reasons for this is both the suboptimality of the successive cancellation (SC) decoding algorithm, poor minimum distance and very low rate of polarization provided by Arikan kernel. Tal-Vardy successive cancellation list decoding (SCL) algorithm with sufficiently large list size was shown to be able to provide near-ML decoding of polar codes [2]. Improved constructions of polar codes, like polar codes with CRC and polar subcodes, were shown to have substantially better minimum distance or, at least, lower error coefficient, compared to classical polar codes [2], [3], [4].

Essentially, the SCL algorithm is used to correct errors in imperfectly polarized subchannels used to transmit unfrozen symbols of a polar code. Although the fraction of such subchannels decreases with code length, their absolute number does increase. Hence, list size needed to implement near-ML decoding of improved polar codes, has to grow exponentially with code length. Replacing the Arikan 2×2 kernel with larger matrices enables one to obtain higher rate of polarization, reducing therefore the number of unfrozen symbols transmitted over imperfectly polarized subchannels, and required decoder list size. However, polar codes with such kernels still suffer from poor minimum distance.

In this paper a method for construction of polar subcodes with non-Arikan kernels is presented. The proposed approach is a generalization of the randomized construction introduced in [4] for the case of Arikan kernel. The obtained codes outperform polar subcodes of extended BCH codes [3] with the same kernel. Together with the efficient kernel processing algorithm introduced in [5], the proposed codes provide (under SCL decoding with sufficiently large list size) better performance at lower decoding complexity compared to polar subcodes with Arikan kernel.

The paper is organized as follows. Section II introduces the background on polar codes and polar subcodes. The proposed code construction is persented in section III. Simulation results are provided in Section IV.

II. Background

A. Channel polarization

Let F_l be an $l \times l$ matrix not permutation-equivalent to an upper triangular matrix. $(n = l^m, k)$ polar code with kernel F_l is a set of vectors $c_0^{n-1} = u_0^{n-1} A$, where $A = B_m F_l^{\otimes m}$, $u_i = 0, i \in \mathcal{F}$, B_m is the digit-reversal permutation matrix, which corresponds to mapping $\sum_{j=0}^{m-1} i_j l^j \to \sum_{j=0}^{m-1} i_j l^{m-1-j}, 0 \leq i_j < l, \mathcal{F} \subset \{0, \ldots, n-1\}$ is the set of frozen symbol indices. Let U_i, C_i, Y_i be random variables corresponding to input values of the polarizing transformation, channel input and output symbols, respectively. Matrix A together with a binary-input memoryless channel $W(Y|C)$ gives rise to synthetic bit subchannels

$$W_m^{(i)}(Y_0^{n-1}, U_0^{i-1}|U_i) = \sum_{U_{i+1}^{n-1} \in \mathbb{F}_2^{n-i-1}} \frac{W(Y_0^{n-1}|U_0^{n-1}A)}{2^{n-1}},$$

where $W(Y_0^{n-1}|C_0^{n-1}) = \prod_{j=0}^{n-1} W(Y_j|C_j)$. It is convenient to define probabilities

$$\begin{aligned} W_m^{(i)}\left\{u_0^i|y_0^{l^m-1}\right\} =& W_m^{(i)}\left\{U_0^i = u_0^i|Y_0^{l^m-1} = y_0^{l^m-1}\right\} \\ =& \frac{W_m^{(i)}(y_0^{n-1}, u_0^{i-1}|u_i) P\left\{u_i\right\}}{W(y_0^{n-1})}, \end{aligned} \quad (1)$$

where $P\left\{u_i\right\} = 1/2, u_i \in \{0, 1\}$ is the probability distribution of input symbols, and $W(y_0^{n-1})$ is the probability density function of the channel output. These probabilities can be recursively computed as

$$W_\lambda^{(lj+i)}\left\{u_0^{lj+i}|y_0^{N-1}\right\} = \sum_{u_{lj+i+1}^{lj+l-1}} \prod_{s=0}^{l-1} W_{\lambda-1}^{(j)}\left\{u_{lt}^{lt+l-1} F_l)_s, 0 \leq t \leq j|y_{\frac{N}{l}s}^{\frac{N}{l}s+\frac{N}{l}-1}\right\}, \quad (2)$$

where $N = l^\lambda$ and $W_0^{(0)}\left\{c|y\right\} = W\left\{c|y\right\}$.

The successive cancellation decoding algorithm makes decisions

$$\widehat{u}_i = \begin{cases} \arg\max_{u_i \in \mathbb{F}_q} W_m^{(i)}\left\{\widehat{u}_0^{i-1}, u_i|y_0^{n-1}\right\}, & i \notin \mathcal{F} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

This algorithm requires $O(n \log n)$ evaluations of $W_\lambda^{(lj+i)}$. Its error probability can be estimated as

$$P_{SC}(\mathcal{F}) \leq \sum_{i \notin \mathcal{F}} Z_{m,i},$$

where $Z_{m,i}$ is the Bhattacharyya parameter of $W_m^{(i)}$.

The performance of polar codes depends on the following parameters:

- Rate of polarization $E(F_l)$, so that for any $\beta < E(F_l)$
$$\liminf_{m\to\infty} P\left\{Z_{m,i} \leq 2^{-l^{m\beta}}\right\} = I(W),$$
and for any $\beta > E(F_l)$,
$$\liminf_{m\to\infty} P\left\{Z_{m,i} \geq 2^{-l^{m\beta}}\right\} = 1,$$
where i is uniformly distributed over $\{0,\dots,l^m-1\}$, and $I(W)$ is the capacity of the original channel $W(y|c)$. Rate of polarization can be computed from the set of partial distances of F_l as [6]
$$E(F_l) = \frac{1}{l}\sum_{i=0}^{l-1} \log_l D_{l,i},$$
where $D_{l,i} = d_H(F_{l,i}, \langle F_{l,i+1},\dots,F_{l,l-1}\rangle)$, $F_{l,j}$ is the j-th row of F_l, and $\langle a_0,\dots,a_j\rangle$ is the linear space spanned by vectors $a_0,\dots,a_j$.
- Scaling exponent $\mu(F_l)$, which gives the minimal length n of a code of rate R, which can provide some fixed decoding error probability on channel W, where
$$n = O\left(\frac{1}{(I(W)-R)^{\mu(F_l)}}\right).$$
A heuristic algorithm for computing the scaling exponent of polar codes for the binary erasure channel is provided in [7].
- Weight distribution w_i, i.e. the number of codewords of weight i. The maximum likelihood decoding error probability is bounded by
$$P_{ML} \leq \sum_{i=d}^{n} w_d Q\left(\sqrt{2iR\frac{E_b}{N_0}}\right),$$
where $d = \min\{i|0\leq i\leq n, w_i > 0\}$ is the minimum distance of the code. For sufficiently large list size, the SCL algorithm can provide maximum likelihood decoding.

B. Mixed-kernel polar codes

The construction of polar codes can be generalized by introducing mixed-kernel polarizing transformation [8]

$$A = BF_{l_0} \otimes F_{l_1} \otimes \cdots F_{l_{m-1}}, \quad (4)$$

where B is the digit-reversal permutation matrix for the mixed-radix representation, and F_{l_i} are some kernels. This results in polar codes of length $n = \prod_{i=0}^{m-1} l_i$. Rate of polarization for such transformation can be computed from $E(F_{l_i})$ as described in [9].

C. Polar subcodes

The performance of polar codes under SCL decoding can be substantially improved by replacing static freezing constraints $u_i = 0, i \in \mathcal{F}$, with dynamic freezing constraints

$$u_i = \sum_{j<i} V_{s_i,j} u_j, i \in \mathcal{F}, \quad (5)$$

where V is a $(n-k)\times n$ constraint matrix, such that distinct rows end[1] in distinct columns $i \in \mathcal{F}$, and s_i is the index of the row ending in column i. Symbols u_i with at least one term in the r.h.s. of (5) are referred to as dynamic frozen (DFS), and those with $V_{s_i,j} = 0, j < s_i$, are denoted static frozen. Matrix V can be constructed so that codewords c_0^{n-1} belong to some $(n, k' > k, d)$ *parent code* (e.g. extended BCH) with sufficiently high minimum distance d, and the SC decoding error probability $P_{SC}(\mathcal{F})$ is minimized. The obtained codes are referred to as polar subcodes [3].

III. Randomized polar subcodes

A. Low-weight codewords of a polar code

The following theorem provides a very simple characterization of the set of unfrozen symbol indices, which are responsible for introducing low-weight non-zero codewords into a polar code.

Theorem 1. *Consider a (n,k) polar code given by a polarizing transformation $A = BF_{l_0}\otimes\cdots F_{l_{m-1}}$ and the set of frozen symbol indices $\mathcal{F}$, where $n = \prod_{i=0}^{m-1} l_i$,*

$$D_{l,i} = \mathrm{wt}(F_{l_i,j}), 0 \leq j < l_i, 0 \leq i < m. \quad (6)$$

Then:

1) *The minimum distance of the polar code is $d = \min_{i\notin\mathcal{F}} \mathrm{wt}(A_i)$, where A_i is the i-th row of matrix A.*
2) *Any codeword $c_0^{n-1} = u_0^{n-1}A$ of weight d, where d is the minimum distance, has $u_i = 1$ for at least one $i : \mathrm{wt}(A_i) = d$.*

Proof. It is sufficient to consider the case of $D_{l_i,j} \leq D_{l_i,j+1}, 0 \leq j < l_i - 1$, since otherwise one can swap the corresponding rows of F_{l_i} and modify appropriately the set $\mathcal{F}$ [6]. In this case $D_{l_i,j}$ is the minimum distance of the code generated by rows $j,\dots,l_i-1$ of F_{l_i}.

Both statements for $m=1$ follow from (6). Assume that the theorem holds for some $A = BF_{l_0}\otimes\cdots F_{l_{m-1}}$ with $m \geq 1$, and consider the polarizing transformation $A' = A \otimes F_{l_m}$. Then $\mathrm{wt}(A_{il_m+j}) = \mathrm{wt}(A_i)\,\mathrm{wt}((F_{l_m})_j)$, where $n = \prod_{i=0}^{m-1} l_i$. Consider a polar code with polarizing transformation A' and the set of frozen symbol indices $\mathcal{F}$. It can be considered as a generalized concatenated code with outer (l_m, k_i, d_i) codes $\mathcal{C}^{(i)}$ and inner (n, K_i, D_i) codes $\mathbb{C}^{(i)}$, where $\mathcal{C}^{(i)}$ is generated by $F_{l_m,j} : l_m i + j \notin \mathcal{F}$, and $\mathbb{C}^{(i)}$ is a polar code with polarizing transformation A and the set of frozen symbol indices $\mathcal{F}^{(i)} = \{j | l_m j + s \in \mathcal{F}, i \leq j < n, 0 \leq s < l_m\}, 0 \leq i < n$. Then the minimum distance of the GCC is $d \geq d_i D_i$, where $D_i =$

[1]Given some binary vector a_0^{n-1}, we say that it ends in position j iff $a_j = 1$ and $a_t = 0, j < t < n$.

$\min_{j \notin \mathcal{F}^{(i)}} \mathrm{wt}(A_i)$, and $d_i \geq \min_{j: l_m i + j \in \mathcal{F}} D_{l_m, j}$. This bound is achieved with equality, since there is a codeword in the considered code given by $A_{il_m+j}, il_m + j \in \mathcal{F}$.

The second statement also holds for any m, since if a codeword has $u_i = 0$ for all $i : \mathrm{wt}(A_i) = d$, then it belongs to a code with the set of frozen symbol indices $\mathcal{F}' = \mathcal{F} \setminus \{i | \mathrm{wt}(A_i) = d\}$, which has minimum distance $d' > d$. □

B. Polar subcodes

We propose to extend the randomized construction introduced in [4].

In order to obtain an (n, k, d) code $\mathcal{C}$ with good performance at high SNR under near-ML decoding, we propose to construct an $(n, k+t, d'), t > 0$, polar code $\mathcal{C}'$, and select randomly its k-dimensional linear subcode. If subcodes are selected according to the uniform distribution, then the components of their weight distribution satisfy

$$\mathbf{E}\left[w_s\right] = w'_s \frac{2^k - 1}{2^{k+t} - 1} \approx w'_s 2^{-t}, s > 0,$$

where w'_s are the components of weight distribution of $\mathcal{C}'$. The parameter t should be selected so that the expected number $\mathbf{E}[w_{d'}]$ of codewords of weight d' in code $\mathcal{C}$ becomes sufficiently small. If $\mathbf{E}[w_{d'}] < 1$, then with high probability the minimum distance d of $\mathcal{C}$ is higher than d'.

Selection of a k-dimensional linear subcode of $\mathcal{C}'$ is equivalent to introducing t dynamic freezing constraints (5), where the submatrix of the overall $(n-k) \times n$ constraint matrix V corresponding to these constraints is uniformly selected from the set of $t \times n$ non-singular matrices.

However, for a small list size most of the errors of the SCL algorithm are caused by the events corresponding to the correct path being killed at some early phase. Hence, the dynamic freezing constraints should be added in such way, so that they can be exploited by the decoder as early as possible, reducing thus the probability of correct path loss.

1) Type-A dynamic frozen symbols: In order to obtain codes better decodable by the Tal-Vardy algorithm, we propose to impose dynamic freezing constraints (5) onto symbols u_i, where $i \in \{0, \dots, n-1\} \backslash \mathcal{F}'$ are t maximal integers, such that $\mathrm{wt}(A_i) = d'$. If there are not enough such integers, the indices $i : \mathrm{wt}(A_i) > d$ should be also considered. The coefficients $V_{s_i,j}$ can be selected as independent equiprobable random binary values. The parameter t needs to be optimized by simulations in order to take into account both the probability of a correct path being prematurely killed by a list decoder, as well as the ML decoding error probability of the obtained code. Let $\mathcal{F}$ be the combined set of $n-k$ indices of frozen symbols.

2) Type-B dynamic frozen symbols: To reduce the probability of correct path being killed at an early phase of the SCL algorithm, we propose to introduce q dynamic constraints (5) on symbols u_i, where $i \in \mathcal{F}'$ correspond to most reliable bit subchannels. Again, their coefficients can be selected as independent equiprobable random binary values. The parameter q needs to be optimized taking into account both the performance and implementation complexity. Simulations suggest that good performance is obtained by setting $q = 64 - t$.

$$K_1 = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{pmatrix}$$

Fig. 1: A kernel with $\mu(K_1) = 3.346$, $E(K_1) = 0.51828$ [5]

3) Construction complexity: To obtain a practical code construction, the coefficients $V_{s_i,j}$ in the dynamic freezing constraints can be obtained as an output of a pseudo-random number generator (e.g. linear feedback shift register) with some fixed seed. The proposed approach requires at most $(q+t)k$ calls to a pseudo-random number generator.

Observe that the complexity of construction of a constraint matrix for a (n, k, d) polar subcode of an extended (n, k', d) BCH code is $O((n-k')^2 n)$. The reduced construction complexity of the proposed randomized polar subcodes enables one to construct them online for any combination of parameters n and k.

To obtain codes of arbitrary length, one can combine kernels of various dimensions l_i, as well as use puncturing and shortening techniques.

IV. Numeric results

The performance of the proposed codes was investigated for the case of AWGN channel with BPSK modulation. The codes were constructed using kernels K_1, shown in Figure 1, and K_2, which was obtained by rearranging rows of K_1 according to permutation $\sigma = [0, 1, 2, 7, 3, 4, 5, 6, 9, 10, 11, 12, 8, 13, 14, 15]$ [5]. The latter kernel has scaling exponent $\mu(K_2) = 3.45$ and $E(K_2) = E(K_1)$. The codes were constructed for AWGN channel with $E_b/N_0 = 1.5$ dB. Bit error probability, obtained via Monte-Carlo simulations, was used to as a reliability measure of bit subchannels while constructing the codes.

Figure 2 presents simulation results for $(4096, 2048)$ polar subcodes based on kernel K_2 with different values of t and q at $E_b/N_0 = 1.5$ dB. Decoding was performed by a straightforward generalization of the SCL algorithm [2], using efficient LLR evaluation method presented in [5]. It can be seen that employing type-B dynamic freezing constraints

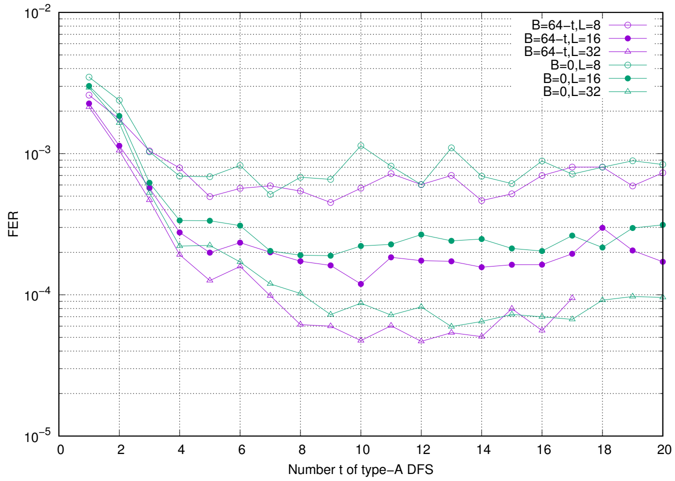


Fig. 2: The impact of type-A dynamic frozen symbols on code performance

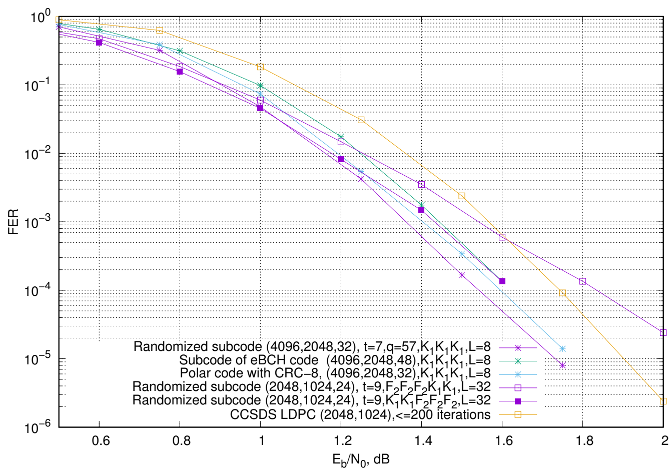


Fig. 3: Performance of polar subcodes with kernel K_2

results in some performance improvement. Furthermore, for each value of list size L there exists an optimal value of t, which minimizes (up to simulation inaccuracy) the frame error rate. This means that the construction of polar subcodes can be optimized for target decoding complexity.

Figure 3 illustrates the performance of the proposed polar subcodes. For comparison, we report also the performance of a polar subcode of an extended BCH code [3], polar code with CRC and CCSDS LDPC code. The minimum distance of the considered codes was obtained using the algorithm presented in [10]. It can be seen that the proposed randomized polar subcodes provide the best performance, although in some cases they have lower minimum distance, compared to other considered codes. In particular, the randomized polar subcode outperforms the polar subcode of an extended BCH code. The reason for this is that the SCL decoder with small list size is not able to fully exploit higher minimum distance of the latter code.

Observe also, that for the case of $(2048, 1024)$ codes, where a mixed kernel construction based on three instances of Arikan kernel F_2 and two instances of kernel K_1 is needed, the order of the kernels in the Kronecker product (4) has quite significant impact on the performance of the obtained codes under SCL decoding.

Figure 4 illustrates the performance of $(4096, 2048)$ polar subcodes at $E_b/N_0 = 1.25$ dB vs. decoding complexity [5]. The complexity is reported in terms of both list size of the SCL algorithm, and actual number of arithmetic operations. It can be seen that for a fixed list size the kernel K_1 with smaller scaling exponent provides the best performance. However, kernel K_2 has a simpler algorithm for computing (approximate) LLRs based on (2), so decoding of the codes based on the latter kernel is more efficient in practice. Furthermore, it can be seen that for sufficiently large L the codes based on both kernels K_1 and K_2 with improved rate of polarization admit simpler decoding compared to polar codes with Arikan kernel F_2 with the same performance.

Figure 5 presents the SNR required for achieving some target FER vs the decoding complexity for $(4096, 2048)$ polar subcodes with Arikan kernel and kernel K_2, as well as $(4000, 2000)$ 5G LDPC code[2] It can be seen that for a fixed decoding complexity polar subcode with kernel K_2 provides up to 1.5 dB performance gain with respect to the LDPC code. Furthermore, for $E_b/N_0 < 1.4$ dB polar subcode with kernel K_2 requires less operations to achieve the same performance under SCL decoding as the code with Arikan kernel. Furthermore, reducing the target FER results in higher gain of polar subcodes with kernel K_2 compared to those based on Arikan kernel.

V. Conclusions

In this paper a method for construction of randomized polar subcodes with non-Arikan kernels was presented. The proposed codes were shown to outperform polar subcodes obtained with an algebraic construction based on extended BCH codes, polar codes with CRC and LDPC codes. Furthermore, it was shown that employing non-Arikan kernels enables one to obtain codes, which admit simpler decoding compared to polar subcodes with Arikan kernel with the same performance under the SCL algorithm, as well as obtain substantially better performance-complexity tradeoff compared to a 5G LDPC code.

Acknowledgements

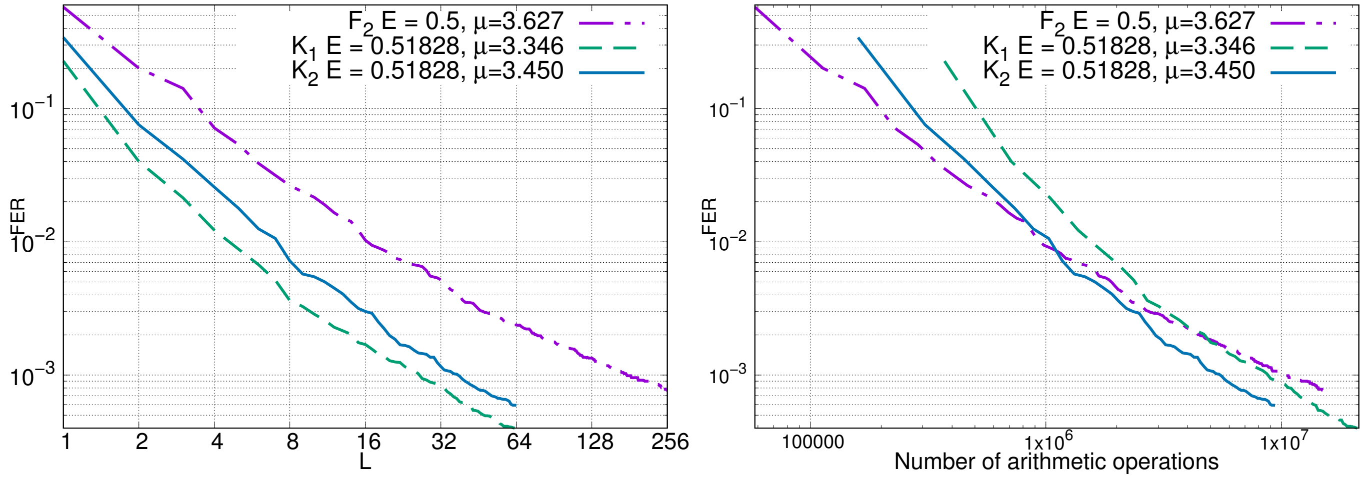
The author thanks Grigorii Trofimiuk for providing kernels K_1 and K_2, and the associated decoding algorithms.

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] I. Tal and A. Vardy, “List decoding of polar codes,” *IEEE Transactions On Information Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.

[2]The results for the LDPC code are reproduced from [11].



(a) Fixed list size

(b) Fixed complexity

Fig. 4: SCL decoding of polar subcodes with different kernels

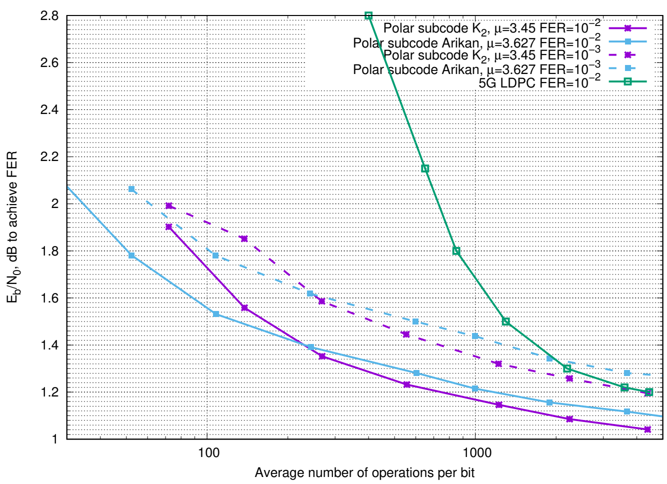


Fig. 5: Performance and decoding complexity of LDPC and polar codes

[3] P. Trifonov and V. Miloslavskaya, "Polar subcodes," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 254–266, February 2016.

[4] P. Trifonov and G. Trofimiuk, "A randomized construction of polar subcodes," in *Proceedings of IEEE International Symposium on Information Theory*. Aachen, Germany: IEEE, 2017, pp. 1863–1867.

[5] G. Trofimiuk and P. Trifonov, "Efficient decoding of polar codes with some 16×16 kernels," in *Proceedings of IEEE Information Theory Workshop*, 2018, submitted.

[6] S. B. Korada, E. Sasoglu, and R. Urbanke, "Polar codes: Characterization of exponent, bounds, and constructions," *IEEE Transactions on Information Theory*, vol. 56, no. 12, pp. 6253–6264, December 2010.

[7] A. Fazeli and A. Vardy, "On the scaling exponent of binary polarization kernels," in *Proceedings of 52nd Annual Allerton Conference on Communication, Control and Computing*, 2014, pp. 797 – 804.

[8] N. Presman, O. Shapira, and S. Litsyn, "Mixed-kernels constructions of polar codes," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 239–253, Feb 2016.

[9] M.-K. Lee and K. Yang, "The exponent of a polarizing matrix constructed from the kronecker product," *Des. Codes Cryptography*, vol. 70, no. 3, pp. 313–322, Mar. 2014. [Online]. Available: http://dx.doi.org/10.1007/s10623-012-9689-z

[10] A. Canteaut and F. Chabaud, "A new algorithm for finding minimum-weight words in a linear code: Application to McEliece's cryptosystem and to narrow-sense BCH codes of length 511," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 367–378, January 1998.

[11] T. Richardson and S. Kudekar, "Design of low-density parity check codes for 5g new radio," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 28–34, March 2018.