Design of binary polar codes with arbitrary kernel

Vera Miloslavskaya, Peter Trifonov
Saint-Petersburg State Polytechnic University, Russia,
{veram,petert}@dcn.ftk.spbstu.ru

***Abstract*—The problem of construction of binary polar codes with high-dimensional kernels is considered. A novel method for computing the erasure probability in the bit subchannels induced by the polarization kernel is proposed. The codes obtained using the proposed method outperform those based on the Arikan kernel.**

I. Introduction

Polar codes is the first class of error correcting codes achieving the symmetric capacity of arbitrary binary-input discrete memoryless channel (B-DMC). Very simple algorithm for construction of polar codes with 2-dimensional Arikan kernel exists for the case of the binary erasure channel [1]. For other output-symmetric binary-input memoryless channels designing a polar code requires employing much more computationally involved density evolution algorithm [2]. An efficient way to implement this approach was presented in [3]. A low complexity alternative to density evolution for the case of AWGN channel based on Gaussian approximation of LLR densities was presented in [4]. These methods recursively estimate the quality of the intermediate bit subchannels of the polarizing transformation, and eventually obtain a list of good bit subchannels suitable for transmission of payload data. However, the performance of polar codes constructed up to now turns out to be inferior compared to the existing turbo and LDPC ones.

It was shown in [5] that high-dimensional kernels (e.g. based on BCH codes) provide greater polarization rate than the Arikan kernel. That is, the decoding error probability of such polar codes decreases much faster with code length compared to similar Arikan codes. However, there are still no efficient methods for designing these codes.

In this paper, a novel method for construction of binary polar codes with high-dimensional kernels is introduced. It is quite difficult to optimize the code for an arbitrary memoryless output symmetric channel, so we consider only the case of binary erasure channel. However, the obtained codes provide quite good performance in the case of AWGN channel too.

The paper is organized as follows. Section II introduces the necessary definitions, notations and algorithms. Section III presents two algorithms for computing the number of uncorrectable erasure configurations for a binary linear block code. The first one employs binary decision diagrams (BDD) to implement explicit counting of erasure configurations uncorrectable by the codes generated by submatrices of the kernel. The second one approximately estimates this quantity using only weight spectrum of these codes. The output of these algorithms is used to recursively estimate the erasure probability in the intermediate subchannels of the polarizing transformation. Numeric results are given in Section V. Finally some conclusions are drawn.

II. Polar codes

A. Construction of codes

A generator matrix of $(n = l^m, k)$ polar code consists of k rows of matrix $F^{\otimes m}$, where F is a $l \times l$ matrix (polarization kernel), and $\otimes m$ denotes the m-times Kronecker product of a matrix with itself. Encoding operation can be represented as $u^m F^{\otimes m} = u^0$, where u^m is a vector of length n that consist of k information symbols and $n - k$ zero (frozen) symbols. The polarizing transformation $F^{\otimes m}$ can be decomposed into $m + 1$ layers, where layer 0 corresponds to codeword $u^0 \in GF(2)^n$, layers $1, \ldots, m-1$ correspond to intermediate vectors $u^1, \ldots, u^{m-1} \in GF(2)^n$, while layer m corresponds to the vector of information and frozen symbols (see Fig. 1).

B. Decoding

The successive cancellation (SC) decoding algorithm is an asymptotically optimal low complexity method for decoding polar codes. We present it in the general case of binary-input output-symmetric memoryless channels.

At layer j the combination of transformations given by F, transformations at layers $(j-1), \ldots, 1$ and the physical data transmission channel induces a number of equivalent subchannels. The quality of these subchannels can be obtained from the probability density function of the LLR for symbols u_i^j [6]. Information symbol estimates $\widehat{u}_i^m$ are obtained by traversing the graph corresponding to the encoding scheme of the code (see Fig. 1). Let the received symbols be represented by the corresponding log-likelihood ratios (LLR) $\nu_0^0, \ldots, \nu_{n-1}^0$. Let ν_i^j be the LLR of symbol u_i^j conditioned on the LLRs $\nu_t^{j-1}, \nu_{t+w}^{j-1}, \ldots, \nu_{t+(l-1)w}^{j-1}$ of symbols at the previous layer and estimates $\widehat{u}_t^j, \widehat{u}_{t+w}^j, \ldots, \widehat{u}_{i-w}^j$, where $w = l^{j-1}$, $t = \lfloor i/l^j \rfloor l^j + (i \mod w)$.

Having computed LLRs $\nu_t^m, \nu_{t+w}^m, \ldots, \nu_{t+(l-1)w}^m$, one can make decisions on the values of the corresponding symbols, i.e. obtain estimates $\widehat{u}_t^m, \widehat{u}_{t+w}^m, \ldots, \widehat{u}_{t+(l-1)w}$. The estimates of symbols at the previous layer are given by

$$(\widehat{u}_t^j, \widehat{u}_{t+w}^j, \ldots, \widehat{u}_{t+(l-1)w}^j) = (\widehat{u}_t^{j+1}, \widehat{u}_{t+w}^{j+1}, \ldots, \widehat{u}_{t+(l-1)w}^{j+1})F.$$

Computation of LLR ν_i^j reduces to decoding in a coset of the code generated by last $l-(i-t)/w$ rows of matrix F. The coset is given by the product of $(\widehat{u}_t^j, \widehat{u}_{t+w}^j, \ldots, \widehat{u}_{i-w}^j)$ and the matrix consisting of first $(i-t)/w$ rows of matrix F.

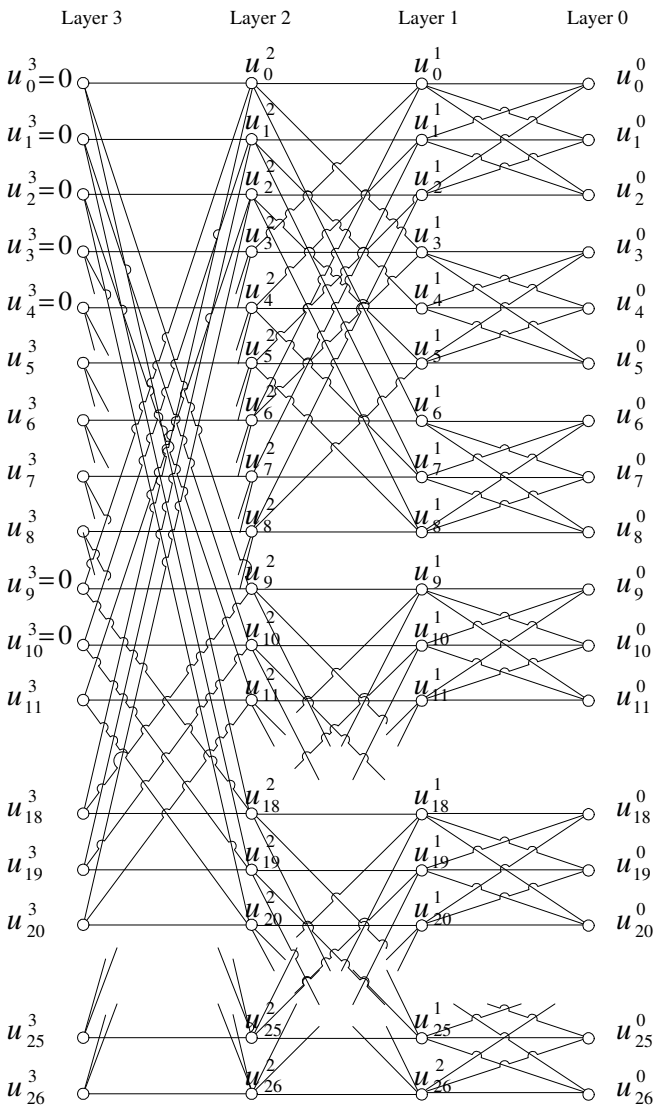

Fig. 1. Encoding/decoding scheme for a polar code of length 27

III. Design of polar codes

The problem of construction of (n, k) polar code reduces to selection of k rows of matrix $F^{\otimes m}$ corresponding to k best bit subchannels. The quality of these subchannels depends on the probability distributions of LLRs ν_i^m, $i = 0, \ldots, n-1$. Finding these distributions is, in general, a very difficult problem. However, in the case of the binary erasure channel a subchannel at layer j can be characterized just by its erasure probability, i.e. the probability that the corresponding decoder at layer j is not able to recover the u_i^j. Hence, bit layer erasure correction performance of linear block codes has to be analyzed.

Consider transmission of polar code codewords over a binary erasure channel with erasure probability $\overline{p}$. Let p_i^j be the erasure probability for symbol u_i^j. It can be seen that the decoders operating at layer 0 observe the same channel erasure probability $\overline{p}$, so it is sufficient to compute only $p_0^1, \ldots, p_{l-1}^1$. In general, all l symbols at the input of a decoder experience identical channel erasure probability. Hence,

$$p_i^j = \begin{cases} Q(p_{i \bmod w}^{j-1}, (i-t)/w), & 0 < j \le m \\ \overline{p}, & j = 0, \end{cases} \tag{1}$$

where $Q(p, \eta)$ is the bit subchannel quality function, which is defined as the decoding failure probability for the 0-th information symbol of the code generated by $\mu = l - \eta$ last rows of matrix F.

If one is able to compute $Q(p, \eta)$, then one can recursively evaluate p_i^j, and construct a generator matrix of $(n = l^m, k)$ polar code by taking k rows of $F^{\otimes m}$ corresponding to the smallest values of p_i^m.

A. Decoding failure probability for an information symbol

Consider transmission of codewords $u\Gamma$ of (l, μ) code with generator matrix Γ over the binary erasure channel with erasure probability p. Define erasure configuration as the vector $E \in \{0, 1\}^l$, where 1 corresponds to erased positions of received vector y. Let us compute probability that the maximum likelihood decoder is not able to estimate u_0, i.e. the erasure configuration is uncorrectable. Erasure configuration is uncorrectable if there exist at least two vectors u', u'' such that $u'_0 \neq u''_0$ and $(u'\Gamma)_i = (u''\Gamma)_i = y_i$ for all non-erased positions $i \in \{0, \ldots, l-1\} \setminus \mathrm{supp}(E)$. Observe that it may be possible to recover u_0 even if the whole codeword can not be uniquely identified for a particular erasure configuration.

The information symbol decoding failure probability can be represented as

$$Q(p, \eta) = \sum_{e=d}^{l} B_e p^e (1-p)^{l-e}, \tag{2}$$

where p is symbol erasure probability, and B_e is the number of uncorrectable erasure configurations of weight e.

B. BDD based method

This subsection presents an efficient method for enumeration of all uncorrectable erasure configurations.

Decoding of vector y with e erasures is equivalent to solving the system of linear equations

$$(u_0, \ldots, u_{\mu-1})\tilde{\Gamma} = \tilde{y}, \tag{3}$$

where $\tilde{y}$ and $\tilde{\Gamma}$ are the subvector of y and the $\mu \times (l-e)$ submatrix of Γ, respectively, corresponding to non-erased symbols. To find the value of u_0 choose a vector $\tilde{z} \in \{0, 1\}^{l-e}$ such that

$$\tilde{\Gamma}\tilde{z}^T = \begin{pmatrix} 1 & 0 & \ldots & 0 \end{pmatrix}^T. \tag{4}$$

Therefore, $u_0 = \tilde{y}\tilde{z}^T$. If vector $\begin{pmatrix} 1 & 0 & \ldots & 0 \end{pmatrix}^T$ does not belong to the column space of matrix $\tilde{\Gamma}$, then the value of u_0 can not be uniquely recovered from equation (3).

The solutions of (4) are given by the solutions of

$$\Gamma z^T = \begin{pmatrix} 1 & 0 & \ldots & 0 \end{pmatrix}^T \tag{5}$$

such that $\mathrm{supp}(z) \cap E = \emptyset$, while (5) specifies all possible ways for recovering u_0 from the codeword symbols. That is, each such vector z defines an expression $u_0 = \sum_{i:z_i=1} y_i$, which can be used by the decoder if all needed elements of vector y are not erased. Consider the set

$$\overline{Y} = \left\{ E | E_i = 1 - z_i, \Gamma z^T = \begin{pmatrix} 1 & 0 & \dots & 0 \end{pmatrix}^T \right\}. \quad (6)$$

It can be seen that all erasure configurations in $\overline{Y}$ are correctable. However, if some erasure configuration E is correctable, then all configurations $E' : \mathrm{supp}(E') \subset \mathrm{supp}(E)$ are correctable too. Notice that each erasure configuration E'': $\mathrm{supp}(E'') \not\subset \mathrm{supp}(E)$ for any $E \in \overline{Y}$, is uncorrectable, since in this case it is not possible to construct a linear expression for u_0 involving only non-erased symbols. Therefore,

$$f(E) = \bigvee_{S \in \overline{Y}} \Big(\bigwedge_{i:S_i=0} \neg E_i \Big)$$

returns true iff erasure configuration E is correctable, while the function $\overline{f}(E) = 1 \Leftrightarrow E \in \overline{Y}$ defines the basis of the set of correctable erasure configurations, i.e. if $\overline{f}(E) = 1$, then $f(E) = 1$.

In order to compute the number of uncorrectable erasure configurations B_e, we construct an ordered binary decision diagram (BDD) corresponding to function $f(E)$ [7], [8], [9]. BDD is an acyclic directed graph with two terminal vertices, which correspond to possible values of a binary function. This graph is partitioned into l layers. The edges of this graph at layer s correspond to possible values of E_s. All paths from the root of the diagram to terminal vertex '1' and to terminal vertex '0' correspond to correctable and uncorrectable erasure configurations, respectively. B_e is equal to the number of paths from the root of the diagram to terminal vertex '0', containing e edges labeled by 1.

The set $\overline{Y}$ is a coset of the code with check matrix Γ (6). This code can be represented by its trellis, which can be transformed to a BDD, defining the membership function $\overline{f}(E)$ for this coset [10], which can be further transformed into a BDD corresponding to $f(E)$.

The transformation algorithm presented below is based on the observation that if E is correctable, then $E' : \mathrm{supp}(E') \subset \mathrm{supp}(E)$ is correctable too. Let $BDD_{s,j}[c], c \in \{0, 1\}$, be the subgraph of the BDD corresponding to $\overline{f}(E)$, which defines some set of erasure configuration suffixes $(E_s, \dots, E_{l-1})$ with $E_s = c$. If there are any paths in $BDD_{s,j}[1]$ from its root to terminal vertex '1', then one should include them into the set of paths from the root of $BDD_{s,j}[0]$ to terminal vertex '1', i.e. one should replace all subgraphs of BDD defining $\overline{f}(E)$ at layer s by subgraphs, corresponding to both the $E_s = 1$ and the $E_s = 0$. This is equivalent to operation $\vee$. Fig. 2 presents this algorithm.

The complexity of this algorithm depends on the state complexity profile of the code with check matrix Γ.

To compute the number of uncorrectable erasure configurations B_e, one should perform the following steps:

1) Construct a minimal trellis for the coset of a code with check matrix Γ given by (5).

PROCESSBDD($\overline{BDD}$)
1 $BDD \leftarrow \overline{BDD}$
2 **for** $s \leftarrow 0$ **to** $l - 1$
3 **do for** $j \leftarrow 1$ **to** $|BDD_{s,-}|$
4 **do** $BDD_{s,j}[0] \leftarrow BDD_{s,j}[0] \vee BDD_{s,j}[1]$
5 **return** BDD

Fig. 2. Transformation of BDD for $\overline{f}(E)$ into BDD for $f(E)$

2) Transform trellis to a BDD corresponding to $\overline{f}(E)$. Let $\overline{BDD}$ denote the result of this operation.
3) Use the algorithm presented in Fig. 2 to transform $\overline{BDD}$ into a BDD corresponding to $f(E)$.
4) Compute coefficients B_e as the number of paths from the root of BDD obtained at step 3 to its terminal vertex '0'.

C. Approximate method

The complexity of the above described algorithm becomes prohibitively high for large kernels. This subsection presents a low complexity alternative to this aproach, which is based on a generalization of the method given in [11].

Theorem 1. *The number of uncorrectable erasure configurations is given by*

$$B_e = \sum_{i=d}^{e} \binom{l-i}{e-i} A_i, \quad e < \lfloor (3d+1)/2 \rfloor, \quad (7)$$

where d is the minimum distance of a coset of a code generated by $\mu - 1$ last rows of F, A_i is the number of codewords of weight i such that $u_0 = 1$, i.e. the weight spectrum of the coset. For $e \geq \lfloor (3d+1)/2 \rfloor$

$$B_e \leq \min \left(\sum_{i=1}^{e} \binom{l-i}{e-i} A_i, \binom{l}{e} \right). \quad (8)$$

Proof: Recall that an erasure configuration is uncorrectable iff it covers at least one codeword, corresponding to $u_0 = 1$. For each codeword x of weight i one can construct $\binom{l-i}{e-i}$ erasure configurations of weight e which cover all nonzero bits of x. If $e < \lfloor (3d+1)/2 \rfloor$, then the sets of such erasure configurations constructed for codewords x and x' are disjoint. Indeed, codewords differ at least in $t_1 \geq d$ positions. The number of positions containing units simultaneously in both codewords is given by $t_2 \geq d - \lfloor t_1/2 \rfloor$. Hence, if the number of erasures satisfies $e = t_1 + t_2 \geq t_1 + d - \lfloor t_1/2 \rfloor = d + \lfloor (t_1+1)/2 \rfloor \geq \lfloor (3d+1)/2 \rfloor$, then such erasure configuration can cover more than one codeword.

In the case of $e \geq \lfloor (3d+1)/2 \rfloor$ the sets of uncorrectable erasure configurations for different codewords can intersect each other. Hence, one obtains an upper bound $B_e \leq \sum_{i=1}^{e} \binom{l-i}{e-i} A_i$. On the other hand, this number can not exceed $\binom{l}{e}$. ■

Observe that the decoding failure probability (see (2)) is dominated by the terms with small e. Hence, the untightness

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

Fig. 3. Extended BCH kernel F_{32}

of bound (8) does not result in significant inaccuracy while computing $Q(p, \eta)$.

IV. Discussion

An expression for bitwise decoding failure probability was derived in [12], which requires computing the number of uncorrectable erasure configurations. However, no techniques for solving this problem were provided.

Similar problem was considered also in [13], where an algorithm for computing the local weight profile of a linear code was presented. Employing local weight profile of a code enables one to improve the tightness of the union bound for the case of AWGN channel. The codewords included into calculation of the local weight profile (neighbors of 0) can be used instead of the set $\overline{Y}$. However the complexity of computing local weight profile is given by $O(n^2 k 2^k)$ in the general case, and by $O(nk2^k)$ in the case of cyclic codes. Construction of $\overline{Y}$ reduces to construction of a minimal trellis for a code, which costs $O(n2^{\min(k,n-k)})$.

V. Numeric results

This section presents the performance of polar codes with the kernels based on extended BCH codes. Fig. 3 presents an example of an extended BCH kernel. The leftmost column of this matrix represents the parity check bit (except the top row). The v-th row (excluding the leftmost symbol) represents a vector of coefficients of polynomial

$$\pi_v(x) = x^j \underbrace{\prod_{i=0}^{t} m_i(x)}_{g_t(x)},$$

where $\deg \pi_v(x) = v - 1, 1 \le v < l$, $m_i(x) | (x^{l-1} - 1)$, and $g_t(x)$ are generator polynomials of nested BCH codes.

Fig. 4 presents the performance of bit subchannels (sorted by BER) of a single 32×32 polarization kernel in the case

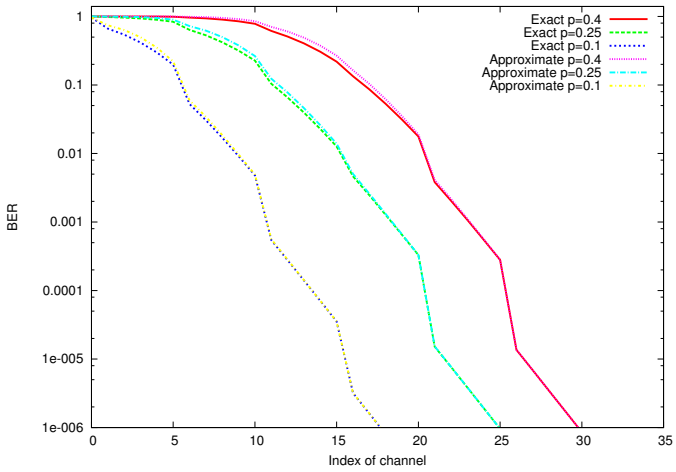

Fig. 4. Performance of bit subchannels of F_{32} kernel

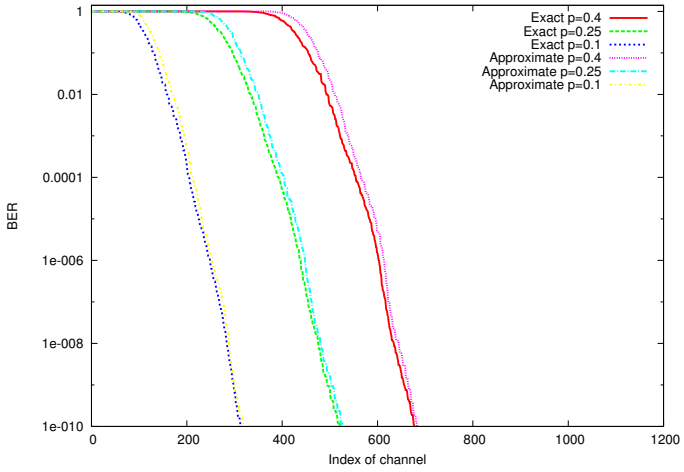

Fig. 5. Performance of bit subchannels of $F_{32}^{\otimes 2}$ polarizing transformation

of binary erasure channel with erasure probability p. The curves labeled "Exact" were obtained using exact values of B_e computed using the algorithm in Section III-B. Those designated "Approximate" correspond to the bound given by Theorem 1. It can be seen that the proposed approximate method enables one to accurately estimate the performance of all except a few very bad subchannels. The non-smooth behavior of curves in this figure is due to stepwise increase of minimum distances of subcodes generated by rows of F_{32} (it takes values in the set $\{1, 2, 4, 6, 8, \ldots\}$). In the case of $F_{32}^{\otimes 2}$ (see Fig. 5) the inaccuracy propagates to medium quality subchannels too. However the discrepancy remains quite low and the estimates obtained with the proposed approximate method can still be used for construction of polar codes.

The proposed methods were used for construction of polar codes optimized for the case of BEC. The erasure probability p was selected empirically, so that the decoding error probability of the obtained codes in AWGN channel with BPSK modulation for $E_b/N_0 = 2$ dB ($(1024, 512)$ code)

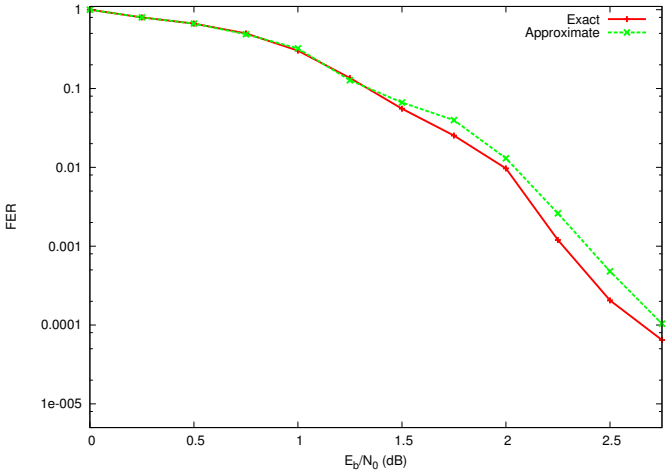

Fig. 6. $(1024, 512)$ polar codes with 32×32 BCH kernel (exponent=0.537)

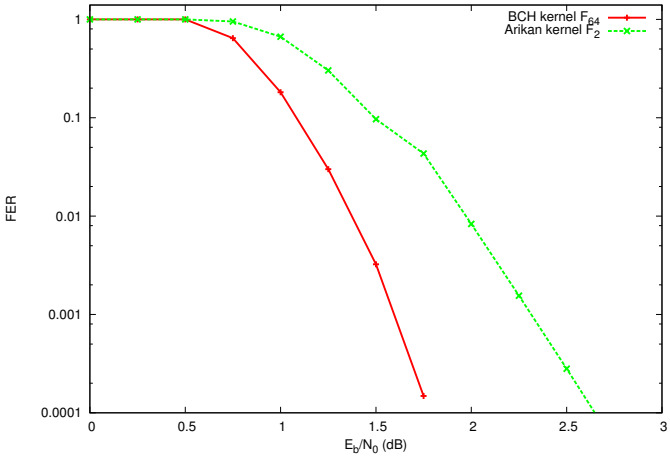

Fig. 7. $(4096, 2048)$ polar codes with 64×64 BCH kernel (exponent=0.564) and 2×2 Arikan kernel (exponent=0.5)

and $E_b/N_0 = 1.5$ dB ($(4096, 2048)$ code) is minimized. In both cases the optimal value turned out to be equal to 0.35. However, it may be possible to obtain better codes by explicit optimization for AWGN channel.

Fig. 6 presents the performance of polar codes constructed using exact and approximate methods for the case of AWGN channel. It can be seen that the code designed using the exact method outperforms the one obtained using the approximate method, but the difference is negligible. The comparison of $(4096, 2048)$ polar code based on BCH F_{64} and Arikan kernels is provided in Fig. 7. The curve for the Arikan polar code was obtained using the decoding algorithm presented in [14], which was shown to outperform the SC decoder. However, the higher polarization exponent of the BCH kernel results in substantially better performance.

The results presented in Fig. 6 and 7 were obtained using the soft-decision SC decoding algorithm based on BCJR and ordered statistics algorithms [15], [16]. These algorithms were used for computing a-posteriori LLR for information symbols of codes generated by μ last rows of the polarization kernel. The BCJR algorithm was used for $l - \mu \leq 5$ for F_{32}, and $l - \mu \leq 7$ for F_{64}, while ordered statistics algorithms were used in all other cases. In both cases simulations were run until 30 errors occur.

VI. Conclusion

In this paper a novel method for construction of polar codes with arbitrary binary polarization kernels was proposed. The proposed approach is based on the information symbol decoding failure probability analysis of codes generated by submatrices of the polarization kernel. Due to high complexity of the bit subchannel performance analysis problem, only the special case of the binary erasure channel was considered. Nevertheless, the obtained polar codes provide good performance in the case of AWGN channel.

Acknowledgement

This work was partially supported by the grant MK-1976.2011.9 of the President of Russia and by the Saint-Petersburg government research grant for students.

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] R. Mori and T. Tanaka, "Performance and construction of polar codes on symmetric binary-input memoryless channels," in *Proceedings of IEEE International Symposium on Information Theory*, 2009.

[3] I. Tal and A. Vardy, "How to construct polar codes," *IEEE Transactions On Information Theory*, 2011, submitted for publication.

[4] P. Trifonov and P. Semenov, "Generalized concatenated codes based on polar codes," in *Proceedings of IEEE International Symposium on Wireless Communication Systems*, 2011.

[5] 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.

[6] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[7] C. Y. Lee, "Representation of switching circuits by binary-decision programs," *Bell Systems Technical Journal*, vol. 38, pp. 985–999, 1959.

[8] S. B. Akers, "Binary decision diagrams," *IEEE Transactions on Computers*, vol. C-27(6), pp. 509–516, 1978.

[9] R. T. Boute, "The binary decision machine as a programmable controller," *EUROMICRO Newsletter*, vol. 1(2), pp. 16–22, 1976.

[10] J. Lafferty and A. Vardy, "Ordered binary decision diagrams and minimal trellises," *IEEE Transactions on Computers*, vol. 48, no. 9, pp. 971 – 986, September 1999.

[11] R. Heller, "Forced-erasure decoding and the erasure reconstruction spectra of group codes," *IEEE Transactions on Communication Technology*, vol. 15, no. 3, pp. 390–397, 1967.

[12] J. H. Weber and K. A. Abdel-Ghaffar, "On decoding failure probabilities for linear block codes on the binary erasure channel," in *IEEE Information Theory Workshop*, 2006.

[13] E. Agrell, "Voronoi regions for binary linear block codes," *IEEE Transactions on Information Theory*, vol. 42, no. 1, pp. 310–316, January 1996.

[14] P. Trifonov, "Efficient design and decoding of polar codes," *IEEE Transactions on Communications*, 2012, accepted for publication.

[15] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, pp. 284–287, 1974.

[16] M. P. Fossorier and S. Lin, "Soft-decision decoding of linear block codes based on ordered statistics," *IEEE Transactions on Information Theory*, vol. 41, no. 5, pp. 1379–1396, September 1995.