*Thirteenth International Workshop on Algebraic and Combinatorial Coding Theory*
*June 15-21, 2012, Pomorie, Bulgaria* *pp. 317–322*

# Soft-decision decoding of polar codes with Reed-Solomon kernels [1]

Peter Trifonov petert@dcn.ftk.spbstu.ru
Saint-Petersburg State Polytechnic University

**Abstract.** The problem of efficient soft-decision decoding of polar codes with Reed-Solomon kernel is considered. A decomposition of the kernel based on the cyclotomic FFT algorithm is proposed, which enables one to implement near-optimal evaluation of log-likelihood ratios in the successive cancellation decoding algorithm.

## 1 Introduction

Polar codes represent the first class of error correcting codes approaching the capacity of a wide range of communication channels [1]. However, the rate of polarization provided by the original Arikan kernel is quite low. As a result, the performance of polar codes construct up to now is inferior compared to the existing LDPC and turbo codes. It was shown in [3] that high-dimensional kernels (in particular, those based on BCH codes) provide higher polarization rate. This approach was further extended in [4], where non-binary polarization kernels based on Reed-Solomon codes were proposed, which achieve the highest possible polarization rate. This enables one to obtain better performance under the successive cancellation decoding algorithm. However, employing this algorithm essentially requires one to be able to perform SISO decoding of nested Reed-Solomon codes.

This paper introduces a generalization of the Vardy-Be'ery decomposition of Reed-Solomon codes, which enables one to efficiently implement this step. The decomposition is based on the cyclotomic FFT algorithm. The paper is organized as follows. Section 2 presents the necessary background. The proposed SISO decoding algorithm is introduced in Section 3. Numeric results are provided in Section 4. Finally, some conclusions are drawn.

## 2 Background

### 2.1 Polar codes

Let $G$ be a $l \times l$ matrix over $\mathbb{F}_q$. Polar code is a linear block code with generator matrix given by some rows of $G^{\otimes L}$, where $\otimes L$ denotes the $L$-times Kronecker product of a matrix with itself. Polar codes were shown to be instances of generalized concatenated and multilevel codes [5]. The encoder of a polar code can be decomposed into a number of layers, as shown in Figure 1. Layer $L$

[1] This research is partially supported by the Russian Foundation of Basic Research under the grant 12-01-00365-a.

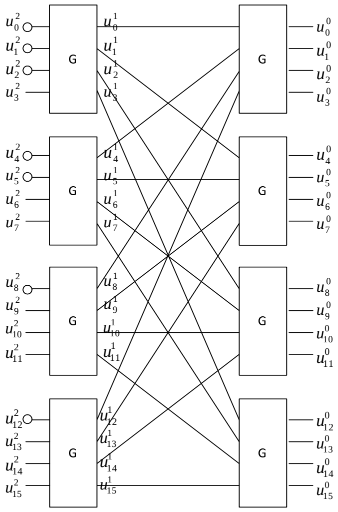

Figure 1: Two-layer polar code based on a $4 \times 4$ kernel

corresponds to the information symbols, while layer 0 corresponds to codeword symbols. $n-k$ symbols at layer $L$ are always fixed to zero (frozen symbols).

The successive cancelation decoding algorithm computes a-posteriori distributions for the input symbols given the distributions for the symbols at layer $L-1$, and makes decisions on the values of the information symbols. As soon as estimates for all symbols connected to a single kernel block are obtained, they are re-encoded. The re-encoded symbols are used at layer $L-1$ while computing the distributions for the remaining information symbols. The distributions at layer $L-1$ are recursively obtained from those at previous layers. It can be seen that this algorithm requires one to be able to compute $P\left\{v_i = s|v_0, \ldots, v_{i-1}\right\}, s \in \mathbb{F}_q$, given the distributions of each symbol of $(w_0, \ldots, w_{l-1}) = (v_0, \ldots, v_{l-1})G$, where $v_0, \ldots, v_{i-1}$ are the symbols processed at the previous steps, i.e. their values are assumed to be known precisely. Without loss of generality, one can assume that $v_0 = \cdots = v_{i-1} = 0$. Hence, implementing the successive cancelation decoding algorithm requires employing SISO decoders for $(l, l-i)$ linear codes generated by last $l-i$ rows of $G$, $0 \leq i < l$.

## 2.2 Cyclotomic FFT algorithm

The discrete Fourier transform of polynomial $f(x) = \sum_{i=0}^{n-1} f_i x^i$ over $\mathbb{F}_{2^m}$ is given by

$$F_j = f(\alpha^j) = \sum_{i=0}^{n-1} f_i \alpha^{ij}, 0 \leq j < n,$$

where $\alpha$ is a primitive $n$-th root of unity. By constructing a linearized decomposition

$$f(x) \equiv \sum_{s=0}^{C} L_s(x^{c_s}) \bmod (x^n - 1),$$

where $L_s(y) = \sum_{t=0}^{m_s-1} f_{c_s 2^t \bmod n} y^{2^t}$, $c_s$ are the leaders of cyclotomic cosets modulo $n$, and $c_s 2^{m_s} \equiv c_s \bmod n$, one can express the DFT components via the values of $L_s(y)$ at some basis points. It is convenient to select normal bases $\left\{\gamma_s, \gamma_s^2, \ldots, \gamma_s^{2^{m_s-1}}\right\}$ of $\mathbb{F}_{2^{m_s}} \subset \mathbb{F}_{2^m}$. In this case one obtains

$$F_j = f(\alpha^j) = \sum_{s=0}^{C} L_s(\alpha^{jc_s}) = \sum_{s=0}^{C} \sum_{i=0}^{m_s-1} a_{jsi} \sum_{t=0}^{m_s-1} \gamma_s^{2^{t+i}} f_{c_s 2^t \bmod n}.$$

This can be expressed in matrix form as [6]

$$F = fLA, \tag{1}$$

where $f$ is a vector of coefficients $f_j$, re-ordered according to cyclotomic cosets, $L$ is a block-diagonal matrix, and $A$ is a binary matrix. This algorithm can be easily augmented to compute also $F_{-\infty} = f(0)$.

# 3 SISO decoding based on cyclotomic decomposition of the Reed-Solomon kernel

The straightforward implementation of a SISO decoder for a Reed-Solomon code would require enumerating all its codewords. More efficient implementation was proposed in [2, 7], where a cycle-free factor graph was constructed based on a decomposition of the Reed-Solomon code into BCH codes and a "glue" code. Due to high complexity this algorithm is not able to decode efficiently Reed-Solomon code of practical length. However, the dimension $l$ of polar code kernel $G$ is usually small. Furthermore, one has to successively decode a sequence of $l$ nested codes with the same input data. In this paper a generalization of this method is proposed, which exploits the structure of the cyclotomic FFT to obtain a decomposition of the nested Reed-Solomon code induced by the corresponding kernel. In order to reduce the overall complexity, only decoding of a binary image of the code is considered. This may result in a suboptimal performance of the successive cancelation decoding algorithm for the corresponding polar code.

Let $(\Lambda_{0,0}, \ldots, \Lambda_{m-1,0}, \Lambda_{0,1}, \ldots, \Lambda_{m-1,l-1})$ be the log-likelihood ratios for each bit $w_{ij}$ of each symbol of vector $w = (w_0, \ldots, w_{l-1}) = vG, v, w \in \mathbb{F}_{2^m}^l$, where $w_i = \sum_{j=0}^{m-1} w_{js}\alpha^s, w_{js} \in \{0, 1\}$, $\alpha$ is a primitive element of $\mathbb{F}_{2^m}$, and $G$ is a Reed-Solomon kernel, which is in fact a DFT matrix. According to (1), one can also write $w = v\Pi LA$, where $\Pi$ is a permutation matrix corresponding to the re-ordering the elements of $v$ according to cyclotomic cosets. Then one obtains

$$p_{j,z} = P\left\{(v\Pi L)^{(j)} = z|\Lambda_{j,0},\ldots,\Lambda_{j,l-1}\right\} = B\exp\left(-\frac{1}{2}\sum_{i=0}^{l-1}\mu(zA,i)\Lambda_{s,i}\right), \quad (2)$$

where $(y)^{(j)}$ denotes the vector $(y_{j,0},\ldots,y_{j,l-1}) \in \mathbb{F}_2^l$, such that $y = \sum_{j=0}^{m-1}(y)^{(j)}\alpha^j$, $B$ is a normalization constant, and $\mu(y,i)$ returns 1 if $y_i = 1$ and $-1$ otherwise. Assume now that one needs to compute the a-posteriori probabilities for some symbol $v_i$, where $i \equiv c_s 2^t \bmod n$ for some $t$. By marginalizing the distribution given by (2), one can obtain the probabilities

$$\pi_{j,s,y} = P\left\{(v\Pi L)^{(j,s)} = y|\Lambda_{j,0},\ldots,\Lambda_{j,l-1}\right\} = \sum_{z:z_{i_t}=y_t, i_t\in I_s} p_{j,z}, y \in \mathbb{F}_2^{m_s}, 0 \le j < m, \quad (3)$$

where $I_s$ denotes the set of columns occupied by the non-zero elements of the $s$-th block of matrix $L$, and $(y)^{(j,s)}$ denotes the corresponding subvector of $(y)^{(j)}$. Now one can construct a trellis corresponding to the code (possibly, trivial) generated by the $s$-th block of matrix L, and use the BCJR algorithm with input given by $\pi_{j,s,y}$ to compute the a-posteriori probabilities (or LLRs) for each of $m$ bits of $v_i$. These LLR values should be used as input to this algorithm at the subsequent layers of the successive cancelation decoder.

After the successive cancellation decoding algorithm makes decisions on symbols $v_0,\ldots,v_i$, it may happen due to block-diagonal structure of matrix $L$ that some parts of $\widehat{z} = v\Pi L$ vector are fixed. In this case one should set the probabilities $\pi_{j,z} = 0$ for all $z$ not matching the detected elements of $\widehat{z}$, and re-normalize the remaining non-zero probabilities. Some blocks $(v)^{(j,s)}$ may be detected incompletely. While the values of the known part of this vector can be taken into account by the BCJR algorithm while processing block $s$, there does not seem to be an easy way to do this while processing other blocks. This represents another source of suboptimality of this algorithm.

It can be seen that the complexity of computing (2) is given by $O(m2^l)$ (this step is performed only once), while the complexity of running the BCJR algorithm for each $i$ is given by $O(2^{\min(i_s,m_s-i_s)m)}m_s)$, where $i_s$ is the number of elements of the cyclotomic coset generated by $c_s$ less than $i$ (i.e. the number of already detected symbols in the corresponding block).

# 4 Numeric results

Figure 2 presents simulation results illustrating the performance of a binary image of the polar code with Reed-Solomon and the polar code with Arikan

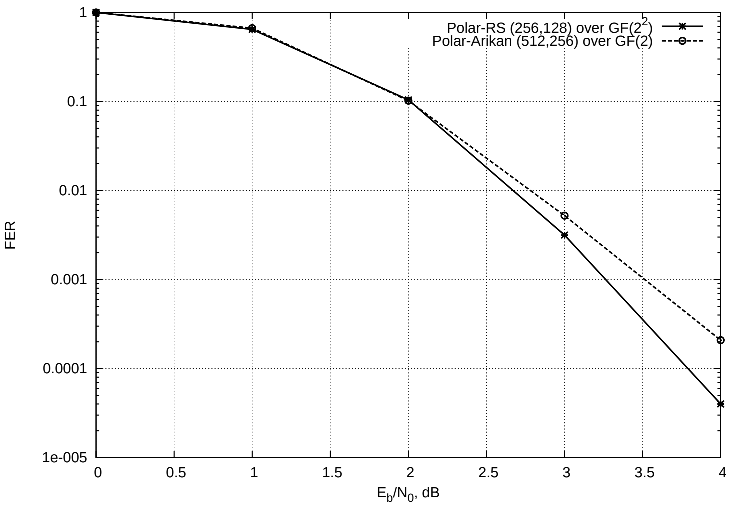

Figure 2: Performance of polar codes

kernel in AWGN channel with BPSK modulation. The Reed-Solomon kernel is given by

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \alpha & \alpha^2 & 0 \\ 0 & \alpha^2 & \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix},$$

where $\alpha$ is a primitive element of $\mathbb{F}_{2^2}$. This factorization was obtained by augmenting the cyclotomic decomposition (1) with the operations needed to compute $F_{-\infty}$. The set of frozen subchannels was obtained by computer simulations. In the case of polar code over $\mathbb{F}_{2^2}$ transmission of its binary image was considered. It can be seen that polar codes with Reed-Solomon kernels outperform binary ones with Arikan kernel.

## 5 Conclusions

In this paper a reduced-complexity decoding algorithm for polar codes with Reed-Solomon kernel was proposed. The algorithm essentially implements SISO decoding of Reed-Solomon codes by exploiting the structure of the cyclotomic FFT over finite fields. This enables one to obtain a factor graph of the code with reduced number of nodes compared to straightforward implementation. The proposed approach generalizes the Vardy-Be'ery decomposition, but the obtained factor graph is not cycle free, resulting thus in suboptimal decoding.

## References

[1] E. Arikan. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Transactions on Information Theory*, 55(7):3051–3073, July 2009.

[2] T. R. Halford, V. Ponnampalam, A. J. Grant, K. M. Chugg. Soft-in soft-out decoding of reed-solomon codes based on vardy and be'ery's decomposition. *IEEE Transactions On Information Theory*, 51(12), Dec. 2005.

[3] S. B. Korada, E. Sasoglu, R.L. Urbanke. Polar codes: Characterization of exponent, bounds, and constructions. *IEEE Transactions On Information Theory*, 56(12), Dec. 2010.

[4] R. Mori, T. Tanaka. Non-binary polar codes using Reed-Solomon codes and algebraic geometry codes. In *Proceedings of IEEE Information Theory Workshop*, 2010.

[5] P. Trifonov. Efficient design and decoding of polar codes. *IEEE Transactions on Communications*, 2011. submitted for publication.

[6] P. V. Trifonov, S. V. Fedorenko. A method for fast computation of the Fourier transform over a finite field. *Problems of Information Transmission*, 39(3):231–238, July-September 2003. Translation of Problemy Peredachi Informatsii.

[7] A. Vardy, Y. Beery. Bit-level soft-decision decoding of Reed-Solomon codes. *IEEE Transactions on Communications*, 39(3):440–444, March 1991.