European Conference on Optical Communication (ECOC) 2022 © Optica Publishing Group 2022

A Parallel Structure for Polar Codes with Adaptive Frozen Set

Hamid Ebrahimzad[(1)], Ali Farsiabi[(1)], Chuandong Li[(1)], Zhuhong Zhang[(1)]

[(1)] Huawei Technologies Canada Co., Ltd, Ottawa, ON, Canada
hamid.ebrahimzad@huawei.com

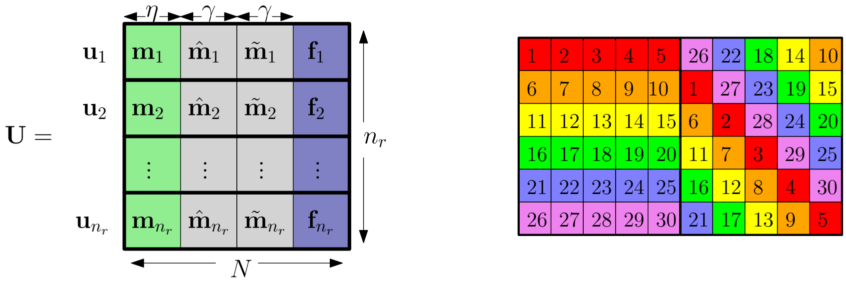
Abstract *We propose a parallel structure for polar codes which is suitable for parallel/pipelined decoding. Our proposed structure outperforms the regular polar code with same length by 0.2 to 0.4dB and can achieve the performance of a polar code with a length twice the length of component codes.*

Introduction

Polar codes are a class of capacity-achieving linear block codes with explicit code constructions and low-complexity encoding and decoding algorithms[1]. The generator matrix of polar codes, also known as polar transform, leads to the channel polarization phenomena based on which, at infinite length, the synthesized bit channels turn into either completely noisy or noise-free. In practice, the bit positions (bit channels) can be sorted according to their reliability. In a polar code with length N and rate $R = K/N$, the K most reliable bit positions are used to carry information bits while the remaining $N-K$ bits referred to as *frozen bits* are set to some values (typically zero) that are known to decoder. The recent innovations in algorithm development of low-complexity, power efficient and high-throughput successive cancellation-based decoders,[2]–[5], have made the polar codes competitive to other state-of-the-art codes. The parallel concatenated codes such as product codes, thanks to their highly parallelizable decoding algorithms, good error correction capability and high throughput, are of great interest in optical communication systems. In this work, we propose a parallel polar structure with multiple polar codes as its horizontal components. In this scheme, following a mapping rule, a small number of bits have an extra level of protection and are vertically coupled using rate-0.5 repetition codes. In optical communication, the concatenated FEC solutions that are composed of high-performance soft inner-codes and low-power hard outer-codes are of practical interests[6],[7]. Our proposed code can be particularly useful as the soft inner-code of a concatenated scheme where the hard outer-code is capable of reducing the error from 10^{-4} to less than 10^{-15}.

Proposed Parallel Structure

Let us define the (N, K) polar codes P_i, $i = 1, \ldots, n_r$, as the component codes that are obtained by applying the polar transform on different rows of the $n_r \times N$ matrix $\mathbf{U}\Pi$ where Π is a permutation matrix used to reorder the columns of the matrix $\mathbf{U}$ illustrated in Fig. 1.a. All the component polar codes follow the same reliability order. It is assumed the columns of $\mathbf{U}$ are sorted according to the reliability order of bit-channels such that the columns on the left and right contain the most and the least reliable bit positions, respectively. In fact, the permutation matrix Π will return the columns of $\mathbf{U}$ to their original order. The columns of $\mathbf{U}$ can be partitioned into four column-groups. The first group on the left, denoted by $\mathbf{M}$, is an $n_r \times \eta$ matrix which is used to carry ηn_r independent message bits. The entries of $\mathbf{M}$ correspond to the most reliable bit-channels of the component codes and is called the *private* part of $\mathbf{U}$. In Fig. 1.a, the second and third column-groups shown with gray color are two $n_r \times \gamma$ matrices, denoted by $\hat{\mathbf{M}}$ and $\tilde{\mathbf{M}}$, that carry γn_r message bits shared among different component codes. The partition related to the matrices $\hat{\mathbf{M}}$ and $\tilde{\mathbf{M}}$, whose bit-channels have medium quality, is referred to as *public* part of $\mathbf{U}$. Finally, the fourth column-group denoted by $\mathbf{F}$ is an all-zero matrix with size $n_r \times (N - \eta - 2\gamma)$ that corresponds to the frozen bits of component codes, thus, $\mathbf{F}$ is called the *frozen* part of $\mathbf{U}$. Following the structure explained above, the row vectors in $\mathbf{U}$ can be written as $\mathbf{u}_i = [\mathbf{m}_i|\hat{\mathbf{m}}_i|\tilde{\mathbf{m}}_i|\mathbf{f}_i]$ where $\mathbf{m}_i$, $\hat{\mathbf{m}}_i$, $\tilde{\mathbf{m}}_i$ and $\mathbf{f}_i$ are the corresponding ith row in $\mathbf{M}$, $\hat{\mathbf{M}}$, $\tilde{\mathbf{M}}$ and $\mathbf{F}$, respectively. In the encoding step, the $2\gamma n_r$ bit locations available in the public part of the code are used to carry, only, γn_r information bits. To achieve this, γn_r message bits are inserted at different entries of $\hat{\mathbf{M}}$ in the public part of $\mathbf{U}$. The rows of $\hat{\mathbf{M}}$ are then partitioned into γ_b subvectors as $\hat{\mathbf{m}}_i = [\hat{\mathbf{m}}_i[1], \hat{\mathbf{m}}_i[2], \ldots, \hat{\mathbf{m}}_i[\gamma_b]]$ for



(a) (b)

Fig. 1: (a) Structure of matrix $\mathbf{U}$ (b) An example of bit-mapping within the public part of $\mathbf{U}$ (gray area) where $n_r = 6$ and $\gamma_b = 5$. The number of bits at each block (subvector) is $\gamma/5$.

$i = 1, \ldots, n_r$. Each subvector $\mathbf{m}_i[k]$ consists of γ/γ_b bits. Subsequently, the corresponding subvectors at each rows of $\tilde{\mathbf{M}}$, which is the second partition in the public part, are determined according to the following mapping rule,

$$\hat{\mathbf{m}}_i[k] = \tilde{\mathbf{m}}_{\bmod(k+i-1,n_r)+1}[k], \quad i = 1, \ldots, n_r \;\;, \;\; k = 1, \ldots, \gamma_b \;\; \text{and} \;\; \gamma_b < n_r, \tag{1}$$

where $\tilde{\mathbf{m}}_i[k]$ denotes the kth subvector of $\tilde{\mathbf{m}}_i$ and $\mathrm{mod}(k + i - 1, n_r)$ represents the remainder of $(k + i - 1)$ divided by n_r. An example of the mapping rule in (1) is illustrated in Fig. 1.b where $\gamma_b = 5$ and $n_r = 6$. As can be observed, each of the subvectors $\hat{\mathbf{m}}_i[k]$ or $\tilde{\mathbf{m}}_i[k]$ in the public part of $\mathbf{U}$ are repeated within two component polar codes and, moreover, a component code does not share more than one subvector with another code. Due to the decoding algorithm that will be explained in the following, cyclic redundancy check (CRC) with length l_c is performed on each rows of matrix $\mathbf{U}$. In this regard, $(\eta - l_c)n_r$ bits of information are equally split between the row vectors $\mathbf{m}_i$ in $\mathbf{M}$. Then the remaining l_c positions of each vector $\mathbf{m}_i$ are allocated to the CRC bits of the corresponding row in $\mathbf{U}$. It is noted that the CRC bits for each $\mathbf{u}_i$ are obtained based on the $(\eta - l_c)$ message bits in $\mathbf{m}_i$ and the $2\gamma n_r$ bits jointly stored in $\hat{\mathbf{m}}_i$ and $\tilde{\mathbf{m}}_i$. After the construction of $\mathbf{U}$, the polar transform is applied on each of the rows in $\mathbf{U}\Pi$ to yield n_r codewords, with length N, related to component polar codes . By considering the overheads (OH) due to public part of $\mathbf{U}$ as well as the CRC, the effective rate of our proposed code can be computed as $R_{eff} = \frac{\eta + \gamma - l_c}{N}$.

In our proposed code structure, each row is a component polar code of length N with $N - \eta - 2\gamma$ frozen bits initially known to decoder. Decoding a received noisy block can be performed in an iterative fashion where each iteration consists of a horizontal and a vertical step. In the horizontal step, the CRC-aided successive cancellation list (SCL) decoding algorithm,[8],[9], with the list size of L, is utilized and all the component polar codes are decoded in parallel. Subsequently, in the vertical step, the decoding of the rows with valid CRCs are declared as successful. The public bits of the successful rows can be appended to the list of frozen bits in other rows whose decoding have been failed. This can enhance the chance of successful decoding of the failed rows by reducing their effective rates at the next iteration. It should be noticed that, at each iteration, only the failed rows of the previous vertical step are redecoded.

Simulation Results

In our simulations, we use fixed-point SC and SCL decoders with 5 bits for quantization. To do so, the channel inputs are, first, saturated with the clipping threshold of 4 and, then, mapped to the integer numbers within the interval $[-15, 15]$. In the following, we use the notations PSC and PSCL to refer to our proposed code where the component codes are decoded by SC and SCL decoders, respectively. In Fig. 2, bit error rate (BER) results of our proposed structure decoded by SC and SCL algorithms with $I_{max} = 2, 4$ are presented. The code parameters are considered as $N = 1024$, $n_r = 40$, $\gamma = 12$, $\gamma_b = 2$, $\eta = 814$ and $l_c = 11$. Therefore, the effective rate of the code is $R_{eff} = 0.8$. In the same figure, along with the BER results of our proposed code, the simulation results of two regular polar codes with rates 0.8 and lengths $N = 1024$ and $N = 2048$ are also presented. As can be seen, in all the cases, our proposed scheme with 2 and 4 iterations outperforms the regular polar code with $N = 1024$. The improvement, at BER equal to 10^{-4}, is between 0.2 to 0.4dB. It is noted that the BER of 10^{-4} is typically considered as the threshold of the outer-code where a hard FEC can be used to reduce the error to less than 10^{-15}. As we expected, increasing the list size from 0 (i.e. SC decoder) to 8 in SCL algorithm improves the error

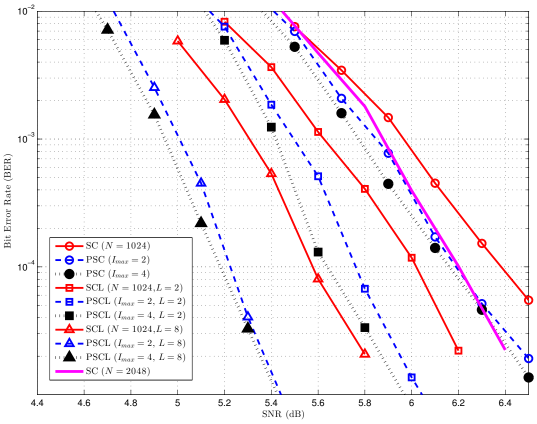


Fig. 2: Bit error rate results showing the performance of the proposed parallel polar code in comparison with the regular polar codes decoded by SC and SCL algorithms.

rate performance of all the codes. However, our code benefits more from the list decoding as the improvement in the error rate is much more than the regular polar code. It can also be observed that, regarding the SC decoding, the error curve of our proposed code with $N = 1024$ and $I_{max} = 2$ is matched with the curve of BER related to the regular polar code with $N = 2048$. We have observed that when $I_{max} = 2$, for the majority of channel realizations, the received blocks can be decoded within the first iteration. In addition, we observed that if the number of component codes going to the 2nd iteration be limited to $0.2n_r$, there will not be any performance loss. This implies that if the dedicated hardware of the 2nd iteration be limited to 20% of the component codes, the proposed code with $N = 1024$ have similar latency but lower complexity than the regular polar code with $N = 2048$. Our analysis, based on the approximate formula provided in[3] for unrolled decoders (which are suitable for high-throughput applications), shows that the memory area of our proposed code is about 40% less than the regular polar code. Regarding the average complexity, we have observed that when the operating SNR is 1dB above the hard-FEC threshold, which is a typical scenario in optical systems, the average number of iterations approaches to 1 leading to a low-power consumption.

Conclusions

We introduced a parallel structure for polar codes that enables parallel/pipelined decoding. We showed that, in terms of bit error rate, our code can outperform the regular polar code by 0.2 to 0.4dB. Also, our proposed code structure with $N = 1024$ can achieve the performance of a regular polar code with length 2048. This is while the decoder of our proposed code has lower average-complexity and requires about 40% less memory area than the regular polar code with $N = 2048$.

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, 2009.

[2] P. Giard, G. Sarkis, C. Thibeault, and W. J. Gross, “237 gbit/s unrolled hardware polar decoder”, *Electron. Lett.*, vol. 51, no. 10, pp. 762–76, 2015.

[3] P. Giard, G. Sarkis, C. Thibeault, and W. J. Gross, “Multi-mode unrolled architectures for polar decoders”, *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 9, pp. 1443–1453, 2016.

[4] A. Sural, E. G. Sezer, Y. Ertugrul, O. Arikan, and E. Arikan, “Terabits-per-second throughput for polar codes”, in *2019 IEEE 30th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC Workshops)*, 2019, pp. 1–7.

[5] A. Sural, E. G. Sezer, K. Ertugrul, V. Derudder, and K. Bertrand, “Tb/s polar successive cancellation decoder 16nm asic implementatio”, *arXiv:2009.09388*, 2020.

[6] M. Barakatain and F. R. Kschischang, “Low-complexity concatenated ldpc-staircase codes”, *Journal of Lightwave Technology*, vol. 36, no. 12, pp. 2443–2449, 2018.

[7] T. Mehmood, M. P. Yankov, A. Fisker, K. Gormsen, and S. Forchhammer, “Rate-adaptive concatenated polar-staircase codes for data center interconnects”, in *2020 Optical Fiber Communications Conference and Exhibition (OFC)*, 2020, pp. 1–3.

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

[9] A. Balatsoukas-Stimming, M. B. Parizi, and A. Burg, “Llr-based successive cancellation list decoding of polar codes”, *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5165–5179, 2015.