International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 15
May 2018

Simulation of the Memory-Efficient Scan Decoder of Polar Codes

K. Kiran Babu & D.Pitchaiah

M.Tech Student Scholar Department of Digital Electronics & Communication Systems Dr. SAMUEL GEORGE INSTITUTE OF ENGINEERING AND TECHNOLOGY, MARKAPUR,A.P, India.

Assoc.professor Department of Electronics & Communication Engineering, Dr. SAMUEL GEORGE INSTITUTE OF ENGINEERING AND TECHNOLOGY, MARKAPUR A.P,

Abstract: *The low-complexity soft-output version of the Successive Cancellation (SC) decoder is the Soft Cancellation (SCAN) decoder that produces reliable information for both the coded and message bits. Methods/Analysis: In polar codes the Belief Propagation (BP) soft-output decoding is a message-passing algorithm, which performs well, at the cost of high processing and storage requirements. In this paper, The SCAN decoder offers better performance than BP decoder which significantly reduces processing and storage requirements. Findings: In this paper the SCAN decoder outperforms the SC decoder both in FER and BER performance with only two and one iteration. Additionally, the SCAN decoder exhibits larger gain in BER performance as compared to FER performance. The SCAN decoder requires only two iterations compared to 60 iterations of the BP decoder to achieve the same FER performance over a dicode channel and outperforms then the BP decoder with further increase in the number of iterations. In the SCAN decoder the normalized memory requirement decreases with the increasing the block length n. Novelty /Improvement: Polar codes outperform Low-Density Parity-Check (LDPC) codes if concatenated with very simple codes in AWGN channel. In this respect, The SCAN decoder used in polar codes has potential applications because of their ability to provide soft outputs.*

Keywords - Polar codes, Soft-output decoding, Turbo equalization, BP decoder

1. Introduction

THE ERROR correcting code in a magnetic recording application must meet stringent error-floor and throughput requirements at a relatively large block length; the sector size for hard disk drives is typically 32768 bits, and the throughput can be 2 Gb/s or more. Regularity in the structure of the encoder/decoder facilitates hardware implementation by reducing interconnect congestion and processing requirements [1]. There has been significant research into finding regularly structured codes. For example, to avoid the high complexity of the early low-density parity check (LDPC) codes, which were random, different structured codes such as quasi-cyclic LDPC codes [2] emerged after their rediscovery in [3] and secured their place in different standards such as IEEE802.11n [4] and IEEE 802.16e [5].

An attractive alternative to LDPC codes are polar codes, discovered by Arikan [6], which feature a highly structured encoder and decoder that asymptotically achieve capacity on discrete memory less channels. Additionally, they possess the desirable properties of universal rate adaptability, explicit construction and reconfigurability. As a result, they naturally demand further exploration for the long-block length throughput-limited magnetic recording channel. The first questions that arise are whether polar codes are a good fit for the magnetic recording application, and if they are, how well they perform. A typical detector architecture in magnetic recording channel relies on the turbo equalization principle [7] that iteratively exchanges soft information between a channeldetector and an error-control decoder. To make polar codes feasible for such an iterative receiver, we need a decoder that can produce soft information about the coded bits.

In this work, we propose a low-complexity soft-output decoder for polar codes that not only outperforms the existing soft-output decoder for polar codes, but also performs about 0.3 dB away from the belief propagation (BP) decoder with LDPC codes on a dicode channel for an FER = 10^{-3}. In the seminal paper [6], Arikan proposed a hard-output successive cancellation (SC) decoder of complexity O(N log N), where N is the block length, that achieved capacity in the limit of large block lengths; its performance with finite-length codes was less promising. Since [6], improving the performance of the SC decoder has been at the forefront of the related research while the generation of soft information with reasonable complexity remained in background. In [8], the authors proposed a successive cancellation list decoder that performed better than the SC decoder at the cost of increased processing and storage complexity. They also showed that polar codes are themselves weak at small block lengths (e.g., N =

2048 or 4096), but if concatenated with a high-rate cyclic redundancy check (CRC) code (e.g., CRC-16), they perform comparably to state-of-the-art LDPC codes. Later in [9], the authors demonstrated that polar codes concatenated with CRC-24 codes can come within 0.2 dB of the information theoretic limit at as low a block length as N = 2048 using an adaptive successive cancellation list decoder with a very large list size. In [10], the authors proposed a successive cancellation stack decoder that also improved the performance over the SC decoder, but incurred huge storage requirements.

Although all of these decoders offer better performance than the SC decoder, none provides the soft outputs essential for turbo-based receivers. To the best of our knowledge, the only soft-output decoder for long polar codes that has appeared in the literature is a belief propagation (BP) decoder [11], [12]. The BP decoder has the advantages of having better performance than the SC decoder and providing soft outputs, but has very high storage and processing complexity. Consequently, the SC decoder remained an attractive choice for low-cost decoding of polar codes [11] for the applications that do not require soft outputs, and polar codes remained infeasible for turbo-based receivers.

This work aims at making polar codes feasible for applications that require soft-output decoders. In particular, we develop a low-complexity soft-output version of the SC decoder called the soft cancellation (SCAN) decoder that produces reliability information for both the coded and message bits. SCAN significantly reduces complexity. For example, the SCAN decoder requires only two iterations compared to 60 iterations of the BP decoder to achieve the same FER performance over a dicode channel and outperforms the BP decoder with further increase in the number of iterations.

An attractive alternative to LDPC1 codes are polar codes, discovered by Arikan2, which feature a highly structured encoder and decoder that asymptotically achieve capacity on discrete memory less channels. Additionally, they possess the desirable properties of universal rate adaptability, explicit construction and reconfigurability. As a result, they naturally demand further exploration for the long-block length throughput-limited magnetic1recording channel. Typical detector architecture in magnetic recording channel relies on the turbo equalization3 principle that iteratively exchanges soft information between a channel detector and an error-control decoder. To make polar codes feasible for such an iterative receiver, we need a decoder that can produce soft information about the coded bits. In this work, we propose SCAN decoder for polar codes. Arikan proposed a hard-output successive cancellation (SC) decoder of complexity O(N logN), improving the performance of the SC decoder has been at the forefront of the related research while the generation of soft information with reasonable complexity remained in back ground. The successive4 cancellation list decoder performed better than the SC decoder at the cost of increased processing and storage complexity.

The successive cancellation stack decoder5 also improves the performance over the SC decoder, but increased huge storage requirements. Although all of these decoders offer better performance than the SC decoder, none provides the soft outputs essential for turbo-based receivers. To the best of our knowledge, the only soft-output decoder for long polar codes is the BP decoder. The BP decoder has the advantages of having better performance than the SC decoder and providing soft outputs, but has very high storage and processing complexity. For that by using different algorithms we develop memory-efficient SCAN decoder.

Polar codes were proposed in [1] as a coding technique that provably achieves the capacity of symmetric binary-input discrete memoryless channels (B-DMCs) with low encoding and decoding complexity. The analysis and construction of polar codes are summarized as follows: (1) Given a B-DMC, virtual channels between the bits at the input of a linear encoder and the channel output sequence are created, such that the mutual information in each of these channels converges to either zero or one as the block length tends to infinity; the proportion of virtual channels with mutual information close to one converges to the original channel's capacity; (2) By transmitting bits through the noiseless virtual channels, under successive cancellation decoding, polar codes achieve the capacity.

The performance of polar codes with successive cancellation decoders is inferior to that of the maximum-likelihood (ML) decoder. Decoders with better finite-length performance have been proposed in the literature. Soft-output decoders such as belief propagation decoders of polar codes were proposed in [2]–[4]. In [5], a list successive cancellation decoder was proposed and the performance was comparable to that of low-density parity-check (LDPC) codes. ML decoding of polar codes implemented by means of sphere decoding (SD) [6], [7] was studied in [8], [9]. These decoders

International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 15
May 2018

have high computational complexity at block length as short as N = 64. Later, [10] proposed SD of binary polar codes via a non-binary tree search, which can decode binary polar codes with length up to 256.

SD decoding was originally proposed to decode multidimensional modulations for fading channels based on lattices in the Euclidean space [11]. In essence, SD decoding performs a depth-first tree search, pruning the search tree according to certain branching conditions, e.g. [12]. SD has been extensively studied in the context of multiple-input multiple-output channels (e.g. [13]–[16]).

In this paper, we improve standard SD branching conditions by computing lower bounds on the optimal decoding metric. We first derive fixed lower bounds that only depend on the received signals. Instead of obtaining fixed lower bounds by enlarging the search space to the real field (as in [12]), we propose fixed lower bounds that keep the finite field constraints. We then propose dynamic lower bounds that are updated during the tree search. Dynamic lower bounds depend both on the received signals and the current tree path.

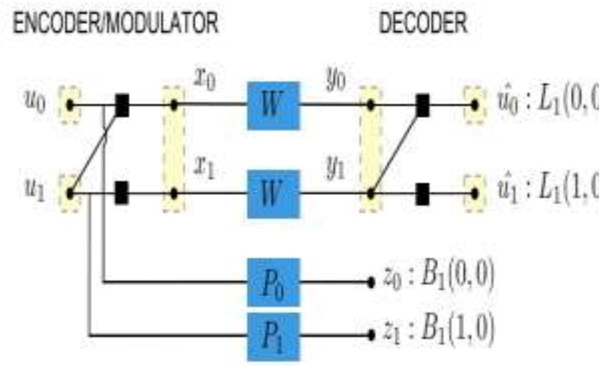

Fig. 1. System model for Lemma: 1

2. The SCAN decoder

Fig 1: shows the system model6 for the SC decoder, The only problem here is to show how to provide additional LLRs B1 (0,0), B1(1,0) in all decision elements in a factor graph for any N. By using SCAN decoder algorithm7 the transition from the hard-output SC decoder to the soft-output SCAN decoder is performed. The SCAN decoder in this form requires as many memory elements as the BP decoder does. To reduce this huge storage requirement we are using another algorithm that is memory-efficient8 SCAN decoder. From Fig 2: In this example9, memory elements required to store B with corresponding φ displayed next to a particular Bλ(φ). In any iteration, the SCAN decoder does not need {Bλ(φ) : φ is even} for the next iteration, and we can overwrite particular Bλ(0) (shown with green rectangles) at any depth λ with { Bλ(φ): φ is even, φ≠0} (shown with yellow rectangles). On the other hand, the SCAN decoder requires {Bλ(φ) : φ is odd} (shown with white rectangles) for processing in the next iteration. The SCAN decoder reuses B2(0) and B3(0) (shown with green rectangles) by overwriting them with the values of B2(2),B3(2), B3(4),B3(6) and does not need extra memory for them. From this we save five memory elements.

2.1. Comparison with the BP decoder

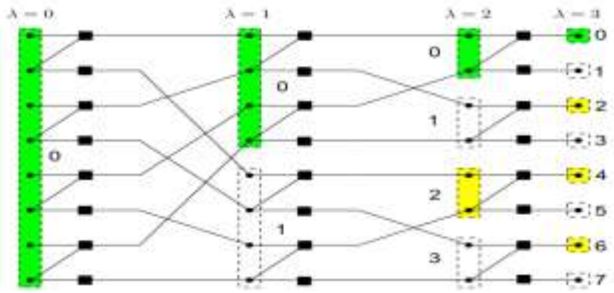

Fig. 2. Memory elements required to store B with corresponding φ displayed next to a particular Bλ(φ).

The main difference between the two decoders lies in the schedule of LLR updates. We explain this difference of the schedule between the two decoders using Fig 2. In the BP decoder updates L and B on a photography-by-photography basis, where as in the SCAN decoder updates L and B on node-by-node basis. From this, the BP decoder needs at least n iterations to disperse the information contained in L0 (0) in the entire factor graph, whereas the SCAN decoder achieves this with only one iteration. In this way, the memory efficient SCAN decoder achieves faster convergence than the BP decoder.

3. Results and Discussion

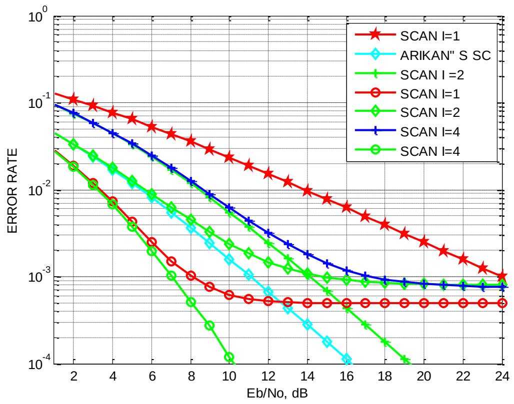

Fig 3: FER, BER performance of the SCAN decoder in AWGN channel

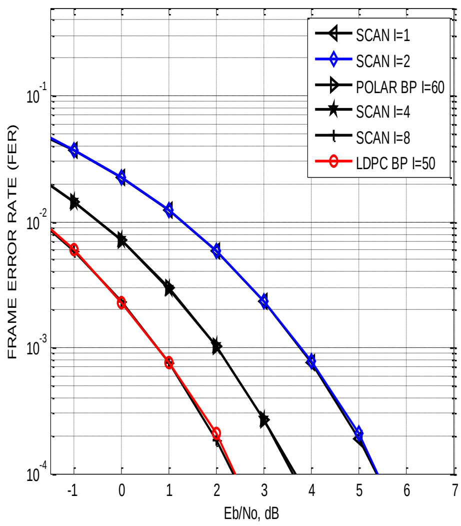

Fig 4: The performance of LDPC code with the BP and the polar code with the SCAN decoder in a dicode channel.

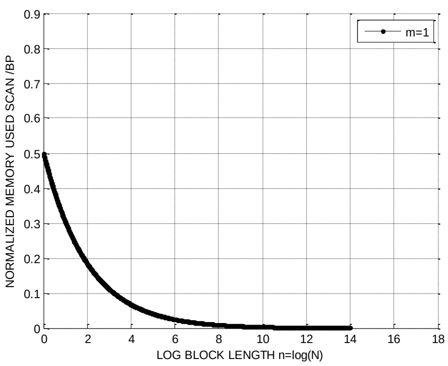

Fig 5: Memory efficiency improves with increasing block length

To demonstrate the improved performance of our algorithm, we simulated the SCAN decoder for a block length of N = 32768 and dimension K = 16, 384 on the AWGN channel. We simulated a maximum of 10^6 frames, if 100 or more frames are found in error. Fig 3: shows the SCAN decoder outperforms the SC decoder both in FER and BER

Performance with only two and one iteration. Fig 4: shows the performance of the SCAN decoder on the dicode channel with N = 8192 and dimension K = 4096 under turbo equalization architecture. The SCAN decoder with only two iterations outperforms than the BP decoder on the dicode channel. Fig 5: shows how the normalized memory requirement decreases with the increasing block length in n.

4. CONCLUSION

The SCAN decoder of polar codes offers good performance, low computational complexity and low memory requirements. The SCAN decoder's performance improves with the increase in the number of iterations. Monte Carlo simulations are performed on the AWGN channel and dicode channel to demonstrate the functionality of the SCAN decoder algorithms. By using the SCAN decoder in polar codes complexity is reduced and there is an improvement in memory.

(5) References

[1] H. Zhong, W. Xu, N. Xie, and T. Zhang, "Area-efficient min-sum decoder design for high-rate quasi-cyclic low-density parity- check codes in magnetic recording," IEEE Trans.Magn.,vol. 43, no. 12, pp. 4117-4122,

[2] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memory less channels," IEEE Trans. Inf. Theory, vol. 55, no.7, pp. 3051–3073, Jul. 2009.

[3] C. Douillard, M. Jzquel, C. Berrou, D. Electronique, A. Picart, P. Didier,and A. Glavieux, "Iterative correction of inter symbol interference: Turbo-equalization," European Trans. Telecommun., vol. 6, no. 5, pp. 507–511, 1995.

[4] I. Tal and A. Vardy, "List decoding of polar codes," in Proc. IEEE Int. Symp. Inform. Theory, Aug. 2011, pp. 1–5.

[5] N. Hussami, S. Korada, and R. Urbanke, "Performance of polar codes for channel

e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 15
May 2018

and source coding," in Proc. IEEE Int. Symp. Inform. Theory, 2009, pp. 1488–1492.

[6] I. Tal and A. Vardy, "How to construct polar codes," IEEE Trans. Inf. Theory, vol. 59, no. 10, pp. 6562–6582, 2013.

[7] Ubaid U. F ayyaz and John R. Barry"Low-Complexity Soft-Output Decoding of Polar Codes "IEEE Commun. Lett., vol. 32, May 2014.

[8] C. Leroux, I. Tal, A. Vardy, and W. Gross, "Hardware architectures for successive cancellation decoding of polar codes," in Proc. IEEE Int. Conf. Acoust., Speech and Signal Process., 2011, pp. 1665–1668.

[9] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Progressive edge-growth tanner graphs," in Proc. IEEE Global Telecommun. Conf., vol. 2, 2001, pp. 995–1001.