Decoder-in-the-Loop: Genetic Optimization-Based LDPC Code Design

AHMED ELKELESH [1], **(Student Member, IEEE), MOUSTAFA EBADA** [1],
SEBASTIAN CAMMERER [1], **(Student Member, IEEE),**
LAURENT SCHMALEN [2], **(Senior Member, IEEE), AND**
STEPHAN TEN BRINK [1], **(Senior Member, IEEE)**
[1]Institute of Telecommunications, University of Stuttgart, 70569 Stuttgart, Germany
[2]Communications Engineering Lab, Karlsruhe Institute of Technology, 76133 Karlsruhe, Germany

Corresponding author: Ahmed Elkelesh (elkelesh@inue.uni-stuttgart.de)

ABSTRACT LDPC code design tools typically rely on asymptotic code behavior and are affected by an unavoidable performance degradation due to model imperfections in the short length regime. We propose an LDPC code design scheme based on an evolutionary algorithm, the Genetic Algorithm (GenAlg), implementing a "decoder-in-the-loop" concept. It inherently takes into consideration the channel, code length and the number of iterations while optimizing the error-rate of the actual decoder hardware architecture. We construct short length LDPC codes (i.e., the parity-check matrix) with error-rate performance comparable to, or even outperforming that of well-designed standardized short length LDPC codes over both AWGN and Rayleigh fading channels. Our proposed algorithm can be used to design LDPC codes with special graph structures (e.g., accumulator-based codes) to facilitate the encoding step, or to satisfy any other practical requirement. Moreover, GenAlg can be used to design LDPC codes with the aim of reducing decoding latency and complexity, leading to coding gains of up to 0.325 dB and 0.8 dB at BLER of 10^{-5} for both AWGN and Rayleigh fading channels, respectively, when compared to state-of-the-art short LDPC codes. Also, we analyze what can be learned from the resulting codes and, as such, the GenAlg particularly highlights design paradigms of short length LDPC codes (e.g., codes with degree-1 variable nodes obtain very good results).

INDEX TERMS LDPC codes, belief propagation decoding, short LDPC code design, EXIT charts, genetic algorithm, evolutionary algorithms, artificial intelligence, decoding complexity.

I. INTRODUCTION

The design of Low-Density Parity-Check (LDPC) codes is well-established at the limits of the infinite length regime. The classical design tools, e.g., density evolution [1] and EXIT charts [2], provide the required analysis to design long LDPC codes of superior performance at negligible gaps from the Shannon limit [3]. However, for finite length LDPC codes, a sufficiently *good* LDPC code turns out to deviate from the guidelines (e.g., degree distributions) provided by the classical design tools which are based on the asymptotic length code analysis. It turns out that LDPC codes lack performance in the ultra-short length regime when compared to more structured, and thus *explicit*, coding schemes such as

The associate editor coordinating the review of this manuscript and approving it for publication was Zilong Liu [ID].

Polar, Reed–Muller (RM) or Bose-Chaudhuri-Hocquenghem (BCH) codes (see [4] for an exhaustive comparison).

Nonetheless, LDPC codes can be seen as the workhorse of many of today's (and upcoming) communication standards motivated by a simple and well-understood decoder, namely the belief propagation (BP) decoder. However, emerging applications based on short block transmission have urged the need for well-designed "ultra-short" codes, cf. ultra-reliable and low-latency communications (URLLC); e.g., for machine-to-machine type communications and Internet of Things networks. In these applications, it is also preferable to work with a unified decoding hardware, i.e., one (de-)coding scheme *fits all* – from block lengths of several hundred up to ten-thousands of bits. This trend is also reflected by the fact that the 3GPP group agreed to replace the Turbo codes by LDPC codes in the upcoming New Radio (NR) access

technology standard [5], [6]. Also, short LDPC codes are used in near-earth and deep space applications [7]. Thus, rather than finding new coding schemes for URLLC implying new decoding algorithms and hardware structures, we aim to leverage *short-length* LDPC codes by explicitly optimizing them at short length and under actual decoder constraints. To some extent, our approach also follows the current trend of *data-driven* computation/optimization in the field of machine learning, i.e., rather than tailoring the code to a specific canonical model (e.g., additive white Gaussian noise (AWGN) channel with non-quantized messages), our method inherently takes any practical decoder hardware constraints ''in-the-loop'' into account and directly optimizes from the data (i.e., the *actual* decoder behavior).

Typically, the LDPC code design is divided into two sub-problems: 1.) finding a good general code structure (degree profile or protograph) and 2.) optimizing the explicit realization of the code (edges of the actual graph). Although there exists some work on short-length code design, e.g., [8], most practical approaches rely on heuristics, e.g., greedy-based optimization techniques such as Progressive Edge Growth (PEG) [9] which, however, typically require degree profiles found by asymptotic assumptions. In [10], a differential evolution algorithm-based approach has been used to optimize the protograph of an LDPC code. However, to the best of our knowledge, no differential evolution-based optimization of the *full* **H**-matrix has been reported so far, probably, due to the demanding computational complexity of the algorithm. In [11], it has been shown that a simple concatenation of an LDPC code with a cyclic redundancy check (CRC) code significantly enhances its performance under high-complexity ordered statistic decoding (OSD). Unfortunately, the gain vanishes for *classical* iterative decoding. Thus, it seems as if finding sparse graphs with good short length performance remains to be a cumbersome task. Yet, there is simply no suitable design strategy to find such a sparse **H**-matrix due to the exponentially increasing design-space of the problem.

A practical coding scheme also implies some further constraints on the parity-check matrix to enable low-complexity encoding such as, e.g., accumulator-based structures. We show that our approach can also be applied to given code structures such as irregular repeat-accumulate (IRA) codes. We refer to [12] for details on these structural graph constraints.

The main contribution of this work is an efficient LDPC code design tool resulting in short codes comparable to, or even outperforming, state-of-the-art short LDPC codes over both AWGN and Rayleigh fading channels. The proposed scheme is used to design the complete parity-check matrix (i.e., **H**-matrix) directly, unlike the classical way where the degree distribution is optimized first. This optimization involves no PEG [9] or similar algorithms, but is only based on the GenAlg, similar to what has been proposed for polar codes in [13]. One strong asset of the proposed design tool is that it can be tailored to any specific required constraint on the **H**-matrix, resulting in codes which are of

both low encoding and, thus, low hardware complexity. It is worth mentioning that extensions to longer LDPC codes are straightforward with the current framework. Furthermore, designing LDPC codes tailored to other types of decoders (e.g., quantized BP decoder, OSD) is possible. The source code and the **H**-matrices from this work can be found online.[1]

II. LDPC CODES

An LDPC code, originally introduced by Gallager [14], is conventionally represented by its corresponding ($m \times n$) parity-check matrix $\mathbf{H} = \left[h_{ji} \right]_{m \times n}$ (referred to as **H**-matrix throughout this work), where n represents the number of variable nodes (VNs) (i.e., also the code block length) and m represents the number of check nodes (CNs) the code has. The number of information bits per codeword is $k = n - \text{rank}\,(\mathbf{H})$. Therefore, the actual code rate is designated by $R_c = k/n$ which could be potentially higher than the so-called design rate $r_d = (n-m)/n$. A corresponding graphical representation [15] is the Tanner graph, in which a VN v_i is connected to a CN c_j if $h_{ji} = 1$, with $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, m\}$.

The decoding of LDPC codes is iteratively performed over the Tanner graph where soft messages (i.e., Log Likelihood Ratio (LLR) messages) are propagated over the graph between variable nodes and check nodes according to

$$L_{c_j \to v_i} = 2 \cdot \tanh^{-1} \left(\prod_{i' \neq i} \tanh \left(\frac{L_{v_{i'} \to c_j}}{2} \right) \right)$$

$$L_{v_i \to c_j} = L_{ch,i} + \sum_{j' \neq j} L_{c_{j'} \to v_i}$$

where $L_{ch,i}$ is the LLR channel output, $L_{c_j \to v_i}$ is the message from CN c_j to VN v_i and $L_{v_i \to c_j}$ is the message from VN v_i to CN c_j. For more details, we refer to [12], [16] and [17].

LDPC code design is, thus, the process of determining (i.e., optimizing) the connections $h_{ji} \in \{0, 1\}$ of the bipartite Tanner graph under certain requirements (e.g., a target error floor or some hardware constraints). Optimizing the degree distributions of the Tanner graph is conventionally pursued via EXIT charts [2], by matching the EXIT curves of the check node decoder (CND) and variable node decoder (VND). This means that the open decoding tunnel between the two EXIT curves should be minimal to operate close to the channel capacity [2]. Another method, density evolution [1], iteratively tracks the (average) probability density functions of the messages propagated between the VND and CND.

The classical design methods assume infinitely long lengths $n \to \infty$, a graph that contains no cycles and infinite number of decoding iterations [18]. These assumptions are not valid when considering the problem of short LDPC code design, which raises the need for a design tool tailored to short length codes [8]. An optimal EXIT chart-based LDPC code design (e.g., matched EXIT curves) is inefficient for a short LDPC code, as shown in Fig. 1. One can see that a
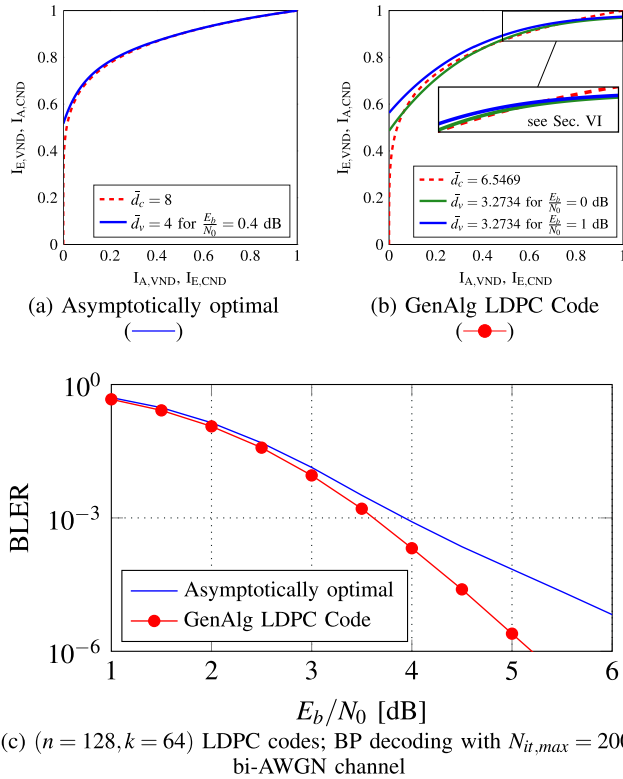
---

[1]https://github.com/AhmedElkelesh/Genetic-Algorithm-based-LDPC-Code-Design

(a) Asymptotically optimal
(——)

(b) GenAlg LDPC Code
(—●—)

(c) ($n = 128, k = 64$) LDPC codes; BP decoding with $N_{it,max} = 200$; bi-AWGN channel

FIGURE 1. The LDPC code design paradigm for asymptotic length fails in the short-length regime.

PEG-optimized short LDPC code realization following the asymptotically optimal degree profiles (see Fig. 1a) has a worse error-rate performance, shown in Fig. 1c, when compared to another short LDPC code designed by our proposed algorithm, and whose EXIT curves are not well-matched and even intersecting in the high mutual information region (see Fig. 1b). This reaffirms our aforementioned statements about the inefficiency of the classical code design tools in the short-length regime.

III. GENETIC ALGORITHM-BASED LDPC CODE DESIGN

We consider the design of an LDPC code as an optimization problem, in which the target is to minimize the block error-rate (BLER) at a certain design SNR. This optimization problem has some constraints defined by the problem at hand. The code rate $R_c = 1 - \frac{\text{rank}(\mathbf{H})}{n}$, the number of VNs and the number of CNs should remain constant. In other words, we keep the number of VNs fixed to n (i.e., no puncturing involved) and the number of CNs fixed to m (i.e., we assume no redundant checks[2]). Furthermore, every variable (or check) node must be connected to at least one check (or variable) node, respectively. To solve this problem, similar to [13], we apply the genetic algorithm (GenAlg) [19].

[2]In case the resulting $\mathbf{H}$-matrix is not of full rank, it holds that $R_c > r_d$ which, if needed, could be (in a naive implementation) solved by freezing some VNs. Thus, we do not impose any further constraint other than m CNs.

The design process starts with an initial population of some randomly constructed LDPC codes (i.e., *population 1*). An error-rate computation framework is used to assess the error-rate performance of the found LDPC codes at a certain design SNR and a fixed maximum number of BP iterations $N_{it,max}$. The best LDPC codes from this population are picked and then undergo evolutionary transformations (mutations and crossovers). This process is repeated until a certain target error-rate or a maximum number of populations (i.e., epochs) is reached, see Fig. 2.
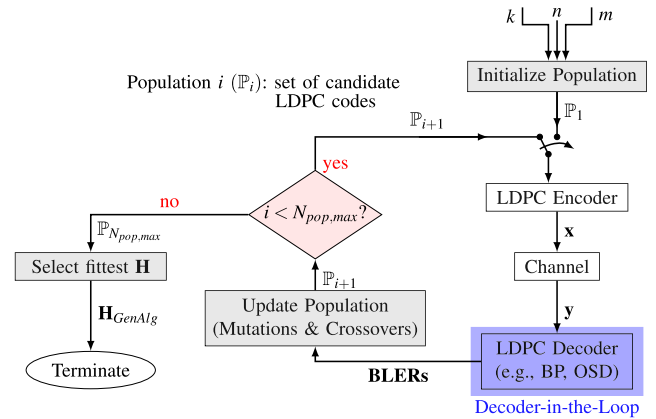

FIGURE 2. Abstract view of genetic algorithm (GenAlg)-based LDPC code design.

The "mutations" are done by adding (or removing) an edge to (or from) the parent $\mathbf{H}$-matrix at a random position, or a combination of both, Fig. 3. The "crossover" is a symmetric 2D-crossover between two parent $\mathbf{H}$-matrices (i.e., $\mathbf{H}_1$ and $\mathbf{H}_2$). The left (or upper) half matrix of $\mathbf{H}_1$ is concatenated with the right (or lower) half matrix of $\mathbf{H}_2$ forming two offspring $\mathbf{H}$-matrices in the next population, Fig. 4.

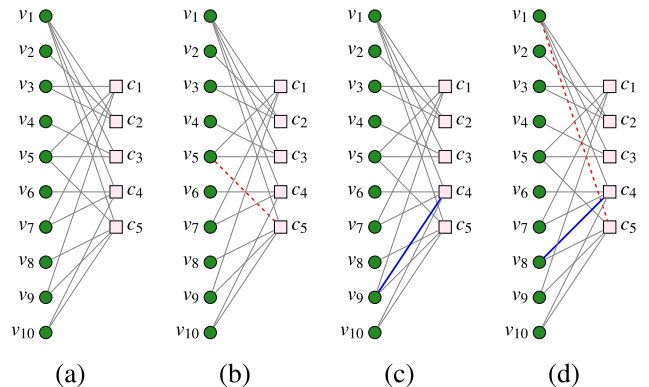**FIGURE 3.** Mutation examples; All derived from parent (a) by: Removing an edge (b), adding an edge (c), or a combination of both (d).

Population ($i + 1$) contains the best (in terms of error-rate) T $\mathbf{H}$-matrices from population i, together with mutated offsprings from those T $\mathbf{H}$-matrices and offsprings due to crossover between each pair of T parent $\mathbf{H}$-matrices. For all
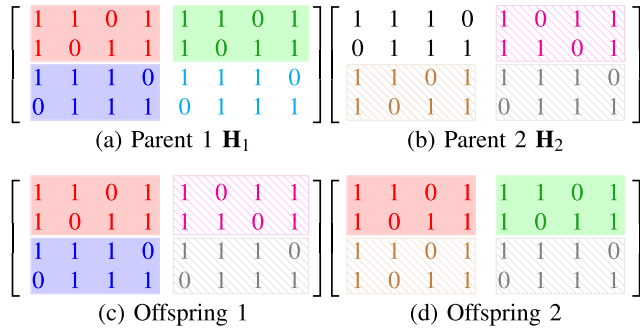
(a) Parent 1 $\mathbf{H_1}$ (b) Parent 2 $\mathbf{H_2}$

(c) Offspring 1   (d) Offspring 2

FIGURE 4. Crossover examples between the two parents $\mathbf{H_1}$ and $\mathbf{H_2}$.

simulation results using the GenAlg as discussed next, we set $T = 20$. We refer to [13] for further details on the GenAlg for code design. For the sake of reproducibility, the source code is available online.

## IV. INSIGHTS FROM OPTIMIZED LDPC CODES OVER AWGN CHANNEL

To be consistent with the results in [4], we design LDPC codes with code length $n = 128$, code dimension $k = 64$ and, thus, code rate $R_c = 0.5$. All considered LDPC codes in this section are simulated over the binary-input AWGN (bi-AWGN) channel. All reference LDPC codes taken from [4] were designed through a girth optimization technique based on the PEG algorithm and are considered state-of-the-art: the standardized LDPC code by the Consultative Committee for Space Data Systems (CCSDS) for satellite telecommand links (──✕──), an accumulate-repeat-3-accumulate (AR3A) LDPC code (──◇──), an accumulate-repeat-jagged-accumulate (ARJA) LDPC code (──□──), and the proposed protograph-based LDPC code for the upcoming 5G NR standard with a base graph (base graph 2 in [20]) optimized for small blocklengths (──⊗──). As a calibration step of our decoding framework, we were able to reproduce exactly the same BLER curves using our own simulation setup. Therefore, the presented gains are not an artefact of different decoder implementations.

To get started, our initial population $\mathbb{P}_1$ contains a set of randomly constructed regular $(3, 6)$ LDPC codes (no PEG used).

A. ERROR-RATE PERFORMANCE

Using GenAlg, we inherently design the whole edge interleaver (i.e., $\mathbf{H}$-matrix) tailored to BP decoding with a maximum number of BP iterations $N_{it,max} = 200$ at a design SNR of 5 dB. The resulting LDPC code (──●──) performs equally good as the 5G LDPC code over the whole simulated SNR range, as shown in Fig. 5a.

To facilitate the encoding of the GenAlg-based LDPC codes, we design accumulator-based codes (i.e., a structured interleaver). We refer the interested reader to [12], [17] for more details about different structured types and design
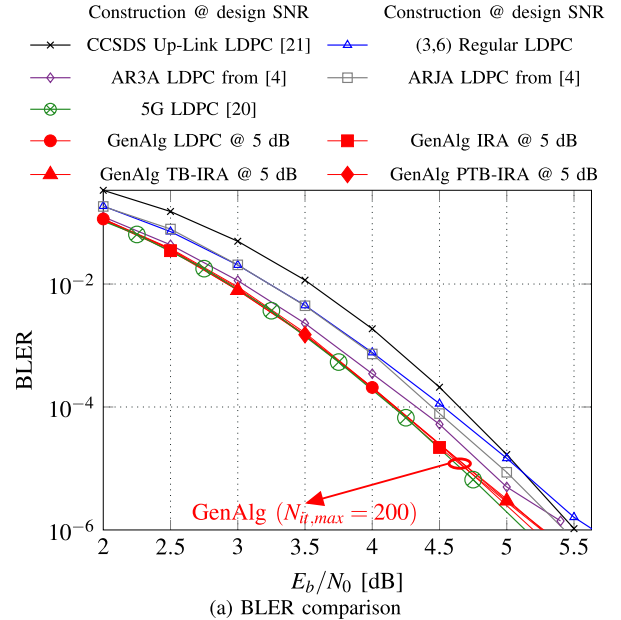
(a) BLER comparison
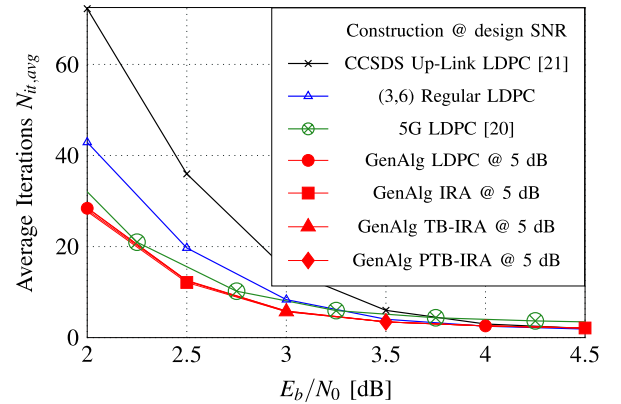
(b) Average number of required iterations

(c) Average decoding complexity; E is the number of edges in the graph of the code

FIGURE 5. Several $(n = 128, k = 64)$ LDPC codes decoded with BP decoding using $\underline{N_{it,max} = 200}$ iterations over the $\underline{\text{bi-AWGN channel}}$.

methods of LDPC codes. In this work, we design IRA codes[3] such that the $\mathbf{H}$-matrix has the form

$$\mathbf{H} = [\mathbf{H}_L \quad \mathbf{H}_R]$$

[3]Similar to the LDPC codes of the DVB-S.2 standard.

where $\mathbf{H}_L$ is the sub-matrix to be optimized and $\mathbf{H}_R$ is a fixed "dual-diagonal" square sub-matrix with dimensions $m \times m$

$$\mathbf{H}_R^{\text{IRA}} = \begin{bmatrix} 1 & & & & & & \\ 1 & 1 & & & & & \\ & 1 & \ddots & & & & \\ & & \ddots & \ddots & & & \\ & & & \ddots & \ddots & & \\ & & & & \ddots & 1 & \\ & & & & & 1 & 1 \\ & & & & & & 1 & 1 \end{bmatrix}.$$

The resulting IRA code without tailbiting (—■—) designed at 5 dB under 200 BP iterations approaches, again, the performance of the 5G LDPC code, as shown in Fig. 5a.

To avoid degree-1 VNs, an upper-right 1 is included in the **H**-matrix for tailbiting irregular repeat-accumulate (TB-IRA) codes, such that $\mathbf{H}_R$ has the form

$$\mathbf{H}_R^{\text{TB-IRA}} = \begin{bmatrix} 1 & & & & & & 1 \\ 1 & 1 & & & & & \\ & 1 & \ddots & & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & 1 & & \\ & & & & 1 & 1 & \\ & & & & & 1 & 1 \end{bmatrix}.$$

The TB-IRA code has only degree-2 VNs in the $\mathbf{H}_R$-matrix. The GenAlg-based TB-IRA code (—▲—) designed at 5 dB again approaches the performance of the 5G LDPC code, as shown in Fig. 5a.

To facilitate the encoding step, while still avoiding degree-1 VNs, a weight-three column replaces the weight-one column in the IRA code and is then moved to the first column in $\mathbf{H}_R$, and thus the name pseudo-tailbiting irregular repeat-accumulate (PTB-IRA) codes, as

$$\mathbf{H}_R^{\text{PTB-IRA}} = \begin{bmatrix} 1 & 1 & & & & & \\ & 1 & 1 & & & & \\ & & 1 & \ddots & & & \\ 1 & & & \ddots & \ddots & & \\ & & & & \ddots & 1 & \\ & & & & & 1 & 1 \\ 1 & & & & & & 1 \end{bmatrix}.$$

This code is similar to the WiMAX LDPC codes (IEEE 802.16e) and the WiFi LDPC codes (IEEE 802.11n). The resulting GenAlg-based PTB-IRA code (—◆—) designed at 5 dB again approaches the performance of the 5G LDPC code, as shown in Fig. 5a.

In some applications, it is crucial to ensure that the worst case decoding latency is relatively low while having an acceptable error-rate performance. In BP decoding, the worst case latency is proportional to $N_{it,max}$. So we design LDPC codes tailored to iterative BP decoding with reduced

(a) BLER comparison

(b) Average number of required iterations

(c) Average decoding complexity; E is the number of edges in the graph of the code

FIGURE 6. Several ($n = 128$, $k = 64$) LDPC codes decoded with BP decoding using $N_{it,max} = 20$ iterations over the bi-AWGN channel.

maximum number of BP iterations $N_{it,max}$. Using GenAlg, an LDPC code tailored to $N_{it,max} = 20$ BP iterations (—○—) designed at 5 dB outperforms the 5G LDPC code, leading to an E_b/N_0 gain of 0.325 dB at BLER of 10^{-5}, as shown in Fig. 6a. The resulting code under 20 BP iterations (—○—)

approaches the error-rate performance of the LDPC codes from [4], with lower $N_{it,max}$. For example, when compared to an AR3A LDPC code decoded with $N_{it,max} = 200$ iterations (- ◆ -). Thus, the proposed code can be decoded with reduced worst case decoding latency with competitive error-rate performance.

It is fair to mention that the 5G LDPC code was designed to support a wide range of blocklengths and code rates. Thus, the error-rate performance was not the only design target. The 5G LDPC codes enables high degree of implementation parallelism and an organized message passing process, besides being described in a compact manner [6]. Fortunately, other such structural constraints can be imposed in our genetic optimization problem to further simplify encoding and decoding implementations of the resulting GenAlg-based LDPC codes.

B. DECODING LATENCY AND COMPLEXITY

We show that for a fixed $N_{it,max}$, significant decoding latency and decoding complexity reduction can be achieved only by optimizing the LDPC code edge interleaver. In BP decoding of LDPC codes, the average decoding latency can be measured using the average number of BP iterations needed by the decoder $N_{it,avg}$. Due to the early stopping condition used, the average number of performed iterations $N_{it,avg}$ is much lower than $N_{it,max}$, especially in the high SNR region. Fig. 5b and Fig. 6b show that our proposed GenAlg-based LDPC codes required on average a lower number of BP iterations when compared to conventionally designed (reference) LDPC codes for the same $N_{it,max}$. This potentially leads to a reduction in the decoding latency. Thus, higher throughput decoder implementations are possible.

In a parallel iterative (message passing) decoder, the decoding complexity depends on the number of performed iterations and the number of arithmetic operations per iteration. The number of arithmetic operations per iteration is proportional to the number of edges in the Tanner graph of the code E. In other words, the decoding complexity heavily depends on the total number of messages passed between the VNs and the CNs. This means that an average decoding complexity measure can be calculated as the product of the average number of iterations $N_{it,avg}$ (due to the early stopping condition) and the number of edges in the Tanner graph E.

Similar to [18], we use the average decoding complexity per information bit η as the decoding complexity measure throughout this work

$$\eta = \frac{N_{it,avg} \cdot E}{k}.$$

Fig. 5c and Fig. 6c show a decoding complexity comparison between our proposed LDPC codes and the 5G LDPC codes under BP decoding with $N_{it,max} = 200$ and $N_{it,max} = 20$, respectively. This means that decoding complexity reduction was possible by designing the LDPC code using GenAlg.

Although our design algorithm, which depends on error-rate simulations (Fig. 7), is more complex than most of the
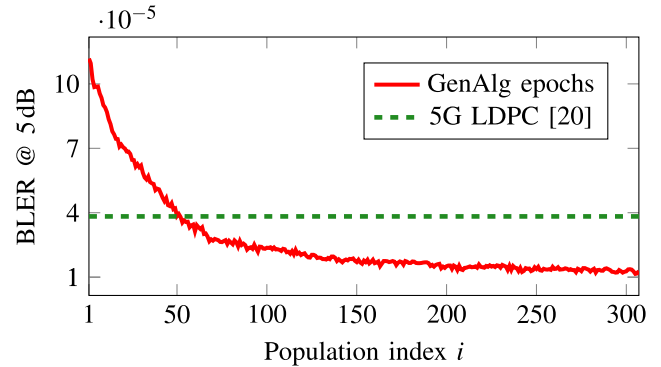

FIGURE 7. Evolution of the BLER at design SNR $E_b/N_0 = 5$ dB; ($n = 128$, $k = 64$) LDPC codes; BP decoding with $N_{it,max} = 20$; bi-AWGN channel.

conventional design tools, our proposed codes can be decoded with (much) lower complexity (Fig. 5c and Fig. 6c). Thus, there is a trade-off between *offline* (i.e., design) complexity and *online* (i.e., decoding) complexity. However, one should keep in mind that the design is only done once while the decoding complexity applies to every later usage of the designed code. Furthermore, our proposed framework can be potentially used to design LDPC codes with the aim of reducing the decoding latency and/or complexity with a slightly relaxed error-rate performance constraint.

C. MINIMUM DISTANCE d_{min}

Computing the minimum distance d_{min} of LDPC codes can be formulated as an integer program (see equation (1)), which can be solved by numerical optimization methods [22]:

$$\min \sum_{i=1}^{n} x_i$$
$$\text{subject to } \mathbf{Hx} - 2\mathbf{z} = \mathbf{0},$$
$$\sum_{i=1}^{n} x_i \geq 1 \qquad (1)$$

where $\mathbf{x} \in \{0, 1\}^n$, $\mathbf{z} \in \mathbb{Z}^m$ and all operations are performed over integer numbers. d_{min} is the value of the objective function $\sum_{i=1}^{n} x_i$ at the minimum.

Table 1 shows the d_{min} of different LDPC codes. As a reference, we included the d_{min} of the 5G polar code specified by the 3GPP group [20] and the RM code with the same

TABLE 1. d_{min} of different ($n = 128$, $k = 64$)-codes.

Code	in Figures	d_{min}
CCSDS Up-Link LDPC	—✕—	14
(3, 6) Regular LDPC	—△—	8
GenAlg LDPC @ 5 dB, 200 iter	—●—	9
GenAlg LDPC @ 5 dB, 20 iter	—○—	8
GenAlg IRA @ 5 dB	—■—	9
GenAlg TB-IRA @ 5 dB	—▲—	9
GenAlg PTB-IRA @ 5 dB	—◆—	9
5G Polar (no CRC)		8
RM		16

code length n and code dimension k. The results in Table 1, Fig. 5a and Fig. 6a reassure that the minimum distance d_{min} is not the only parameter to consider in order to enhance the performance of short linear codes under BP decoding. This can be attributed to the fact that the performance of a linear code under iterative decoding is dominated by the Tanner graph structure of the code and not its d_{min} [23]. However, it is worth mentioning that maximizing d_{min} is important to enhance the error-floor behavior of a code [12]. The CCSDS LDPC code has the largest d_{min} when compared to other LDPC codes considered in this work, because it was designed to operate in the very low error-rate region.

V. RESULTS FOR THE RAYLEIGH FADING CHANNEL

To demonstrate the flexibility of our proposed design algorithm, we also design LDPC codes for the Rayleigh fading channel. To be more precise, we assume an ergodic Rayleigh fading model with full Channel State Information (CSI) which can be motivated as the result of an Orthogonal Frequency-Division Multiplexing (OFDM)-based transmission in a multi-path propagation environment and, thus, is omnipresent in today's wireless communication systems. Our proposed design algorithm benefits from having the decoder-in-the-loop and the channel-in-the-loop. Thus, designing LDPC codes tailored to both the decoder and the channel is possible.

We design LDPC codes tailored to BP decoding with $N_{it,max} = 200$ iterations at a design SNR $E_b/N_0 = 8$ dB. As shown in Fig. 8a, the resulting LDPC code ($-\bullet-$) outperforms the 5G LDPC code optimized for short block lengths over the whole simulated SNR range (i.e., an E_b/N_0 gain of 0.2 dB at a BLER of 10^{-4}). Moreover, our proposed LDPC code uses a lower number of iterations on average when compared to the 5G LDPC code as depicted in Fig. 8b. Also, the average decoding complexity needed to decode our proposed LDPC code is lower than that required for decoding the 5G LDPC code, see Fig. 8c. Thus, our proposed LDPC code has a better error-rate performance and can be decoded with reduced decoding latency and decoding complexity when compared to the 5G LDPC code.

Similarly, we use the same procedure to design LDPC codes tailored to a lower number of BP iterations $N_{it,max} = 20$. In Fig. 9a, we show that our GenAlg-designed LDPC code ($-\circ-$) outperforms the 5G LDPC code over the whole simulated SNR range and an E_b/N_0 gain of 0.8 dB is achieved at a BLER of 10^{-4}. Again, a significant reduction in the average number of required iterations (i.e., decoding latency measure) and the average decoding complexity is achieved when compared to the reference (standardized) LDPC codes, see Fig. 9b and Fig. 9c.

VI. LESSONS LEARNED FROM THE GENETIC LEARNING ALGORITHM

Besides superior decoding performance, we aim to *understand* what makes the resulting codes so powerful and further
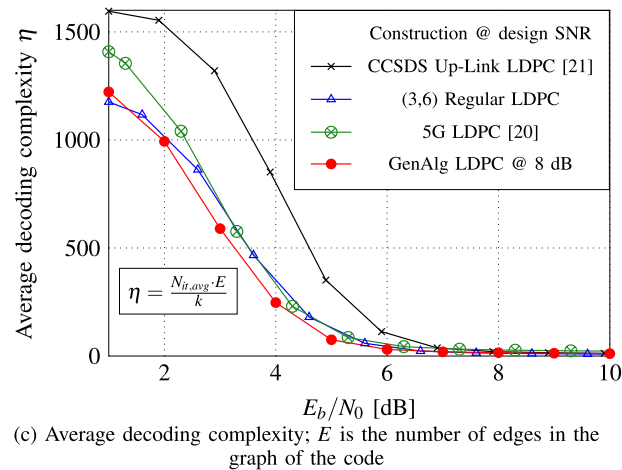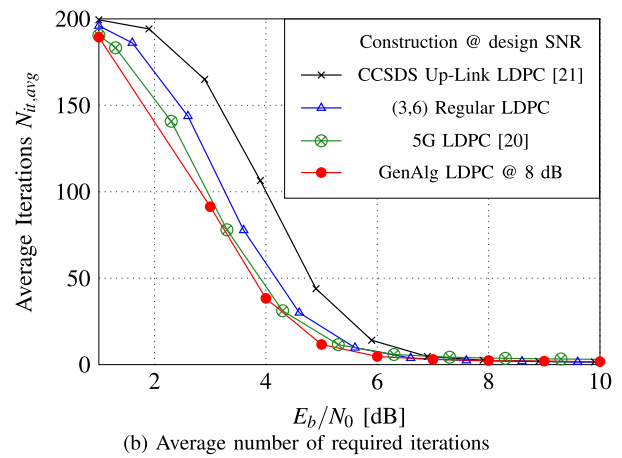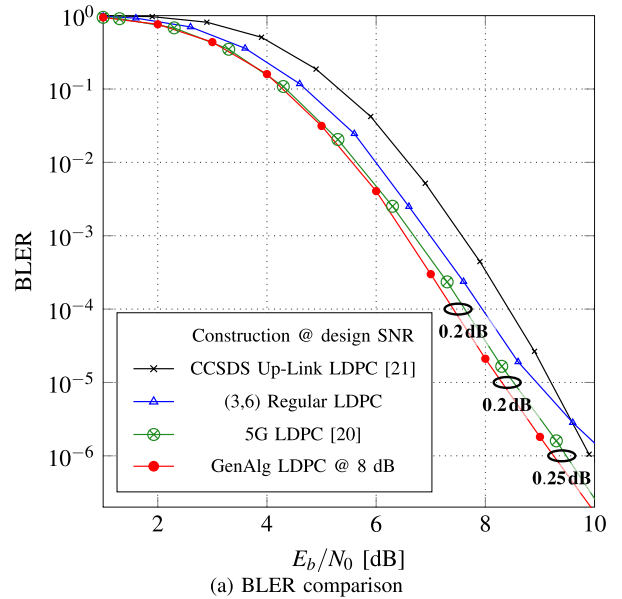

(a) BLER comparison

(b) Average number of required iterations

(c) Average decoding complexity; E is the number of edges in the graph of the code

FIGURE 8. Several ($n = 128, k = 64$) LDPC codes decoded with BP decoding using $N_{it,max} = 200$ iterations over the Rayleigh fading channel.

analyze the final code structure. As such, the GenAlg may even provide new design paradigms for short length codes.

First, we observed that our optimized GenAlg-based LDPC codes (i.e., **H**-matrices) contain some degree-1 VNs. It is
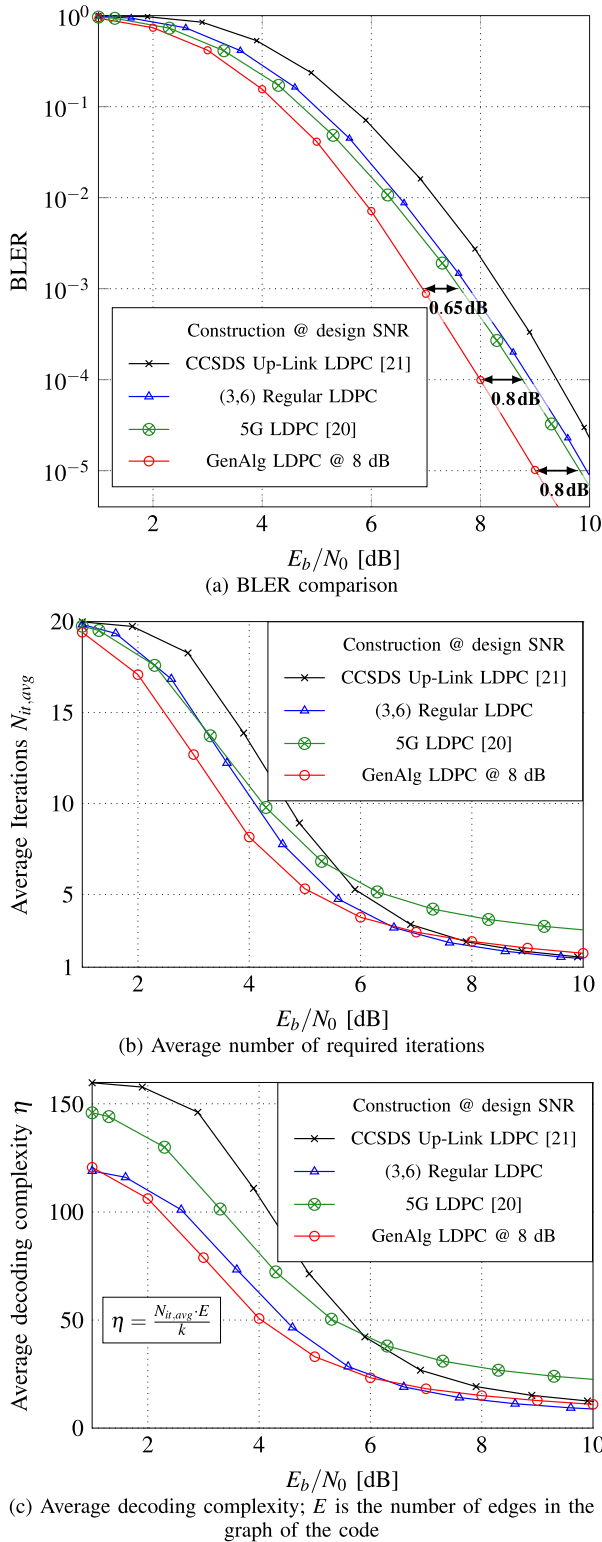
(a) BLER comparison

(b) Average number of required iterations

(c) Average decoding complexity; E is the number of edges in the graph of the code

FIGURE 9. Several ($n = 128$, $k = 64$) LDPC codes decoded with BP decoding using $N_{it,max} = 20$ iterations over the Rayleigh fading channel.

well-known that degree-1 VNs are unfavorable in unstructured LDPC codes as they increase the probability of having more than one degree-1 VN being connected to the same CN

(recall TB-IRA and PTB-IRA). In that case, these degree-1 VNs suffer from unrecoverable poor decoding performance. Obviously, this, prevents having an open tunnel between the VND and CND EXIT curves as shown in Fig. 1b, and converging to the $(1, 1)$ point is not possible. However, as shown in Fig. 1, if wisely placed, those degree-1 VNs are not degrading the actual error-rate performance of the short-length LDPC code. Similarly in [24], degree-1 VNs were used to improve the iterative decoding thresholds of protograph-based LDPC codes. One interesting observation is that in our optimized GenAlg-based LDPC codes, there is no CN which is connected to more than one degree-1 VN, despite the fact that this was not a constraint in the optimization problem. Thus, GenAlg was able to "learn" that the design in which a CN is connected to more than one degree-1 VN should be avoided. A similar application of degree-1 VNs was reported by Richardson in multi-edge type LDPC codes [25].

In order to further investigate the found degree profile, we show a BER comparison between two long length LDPC codes ($n = 128000$) in Fig. 10, as a sanity check:

1) *Conventional design:* the first code is a single realization following the (asymptotically) optimal degree profile found by EXIT chart-based curve matching as shown in Fig. 1a.
2) *GenAlg-based design:* the second code is a *scaled* version (realization) following the degree profile of our optimized GenAlg-based short length LDPC code as shown in Fig. 1b. It has a non-negligible amount of degree-1 VNs, but we ensure per-design that only one degree-1 VN is connected per CN.
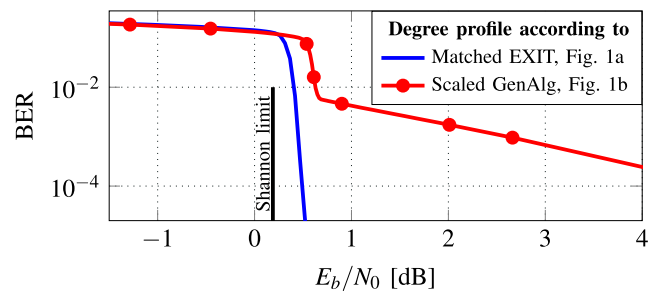

FIGURE 10. Sanity check for long codes; ($n = 128000$, $k = 64000$) LDPC codes; BP decoding with $N_{it,max} = 200$; bi-AWGN channel.

As shown in Fig. 10, it can be seen that the long LDPC code designed based on EXIT chart curve matching benefits from a lower threshold than the code based on GenAlg. Further, the GenAlg-based long LDPC code indeed suffers from an inevitable error-floor due to the significant portion of degree-1 VNs it has. Therefore, Fig. 1 and Fig. 10 clearly indicate that short length codes follow *different* design paradigms and, thus, the GenAlg-based design process leads to better (in terms of error-rate) short length LDPC codes. The intuition behind this effect is the fact that, typically, the price-to-pay for non-matched EXIT curves is a degraded waterfall performance. However, in the short length regime, the slope

in the waterfall region is more important than its exact starting position (i.e., threshold).

Further, we also observe that in our GenAlg-based short LDPC codes, no degree-2 VNs are involved in a short cycle. Despite being a well-known structural constraint in LDPC design, this was not explicitly outlined as a constraint in our GenAlg optimization process. Again, GenAlg was able to "*learn*" this constraint independently.

Next we summarize a few remarkable observations from our GenAlg results:

- Degree-1 VNs do not necessarily cause a performance degradation in short length LDPC codes.
- In the short length LDPC code design process, a non-matched EXIT curves scenario can lead to a good short length LDPC code, while obviously leading to a poor long length LDPC code.
- Unfavorable graph structures (e.g., degree-2 VNs involved in short girth) were inherently avoided by GenAlg without an explicit constraint imposed on it.

VII. CONCLUSION

The classical LDPC code design tools are based on asymptotic length assumptions which are not valid in the short-length regime. Therefore, we focus on constructing short-length LDPC codes (i.e., the parity-check matrix) using the genetic algorithm. We propose a flexible framework accommodating practical decoding requirements and channel constraints. We construct LDPC codes without any special graph structure (i.e., we use a random edge interleaver) and demonstrate the flexibility of the proposed framework. We also construct accumulator-based LDPC codes which can be encoded easily. Our proposed LDPC codes outperform some well-designed state-of-the-art (standardized) LDPC codes over both AWGN and Rayleigh fading channels. Moreover, we design LDPC codes tailored to a reduced number of BP iterations in order to reduce the decoding complexity and latency with good error-rate performance (e.g., a coding gain of up to 0.8 dB when compared to the 5G LDPC code is reported over the Rayleigh fading channel). Finally, we observed that allowing the presence of carefully placed degree-1 VNs opens up more degrees of freedom for code design, and does not degrade the error-rate performance of our proposed short length LDPC codes.

REFERENCES

[1] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge, U.K.: Cambridge Univ. Press, 2007.
[2] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, no. 4, pp. 670–678, Apr. 2004.
[3] S.-Y. Chung, G. D. Forney, Jr., T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, no. 2, pp. 58–60, Feb. 2001.
[4] M. C. Coşkun, G. Durisi, T. Jerkovits, G. Liva, W. E. Ryan, B. Stein, and F. Steiner, "Efficient error-correcting codes in the short blocklength regime," *Phys. Commun.*, vol. 34, pp. 66–79, Jun. 2019.
[5] *New SID Proposal: Study on New Radio Access Technology*, document RP-160671, 3GPP TSG RAN Meeting, NTT DOCOMO, Gothenburg, Sweden, Mar. 2016.
[6] T. Richardson and S. Kudekar, "Design of low-density parity check codes for 5G new radio," *IEEE Commun. Mag.*, vol. 56, no. 3, pp. 28–34, Mar. 2018.
[7] T. de Cola, E. Paolini, G. Liva, and G. P. Calzolari, "Reliability options for data communications in the future deep-space missions," *Proc. IEEE*, vol. 99, no. 11, pp. 2056–2074, Nov. 2011.
[8] M. Ebada, A. Elkelesh, S. Cammerer, and S. ten Brink, "Scattered EXIT charts for finite length LDPC code design," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–7.
[9] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 386–398, Jan. 2005.
[10] H. Uchikawa, "Design of non-precoded protograph-based LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun./Jul. 2014, pp. 2779–2783.
[11] J. Van Wonterghem, A. Alloum, J. J. Boutros, and M. Moeneclaey, "Performance comparison of short-length error-correcting codes," in *Proc. Symp. Commun. Veh. Technol. (SCVT)*, Nov. 2016, pp. 1–6.
[12] G. Liva, S. Song, L. Lan, Y. Zhang, S. Lin, and W. E. Ryan, "Design of LDPC codes: A survey and new results," *J. Commun. Softw. Syst.*, vol. 2, no. 3, pp. 191–211, Sep. 2006.
[13] A. Elkelesh, M. Ebada, S. Cammerer, and S. ten Brink, "Decoder-tailored polar code design using the genetic algorithm," *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 4521–4534, Jul. 2019.
[14] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.
[15] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. IT-27, no. 5, pp. 533–547, Sep. 1981.
[16] W. E. Ryan, "An introduction to LDPC codes," in *Coding and Signal Processing for Magnetic Recording Systems*. Boca Raton, FL, USA: CRC Press, 2004.
[17] W. E. Ryan and S. Lin, *Channel Codes: Classical and Modern*. Cambridge, U.K.: Cambridge Univ. Press, 2009.
[18] B. Smith, M. Ardakani, W. Yu, and F. R. Kschischang, "Design of irregular LDPC codes with optimized performance-complexity tradeoff," *IEEE Trans. Commun.*, vol. 58, no. 2, pp. 489–499, Feb. 2010.
[19] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. Cambridge, MA, USA: MIT Press, 1992.
[20] *Technical Specification Group Radio Access Network*, document 3GPP, TS 38.212 V.15.1.1, 2018. [Online]. Available: http://www.3gpp.org/ftp/Specs/archive/38_series/38.212/
[21] M. Helmling, S. Scholl, F. Gensheimer, T. Dietz, K. Kraft, S. Ruzika, and N. Wehn. (2019). *Database of Channel Codes and ML Simulation Results*. [Online]. Available: https://www.uni-kl.de/channel-codes
[22] M. Punekar, F. Kienle, N. Wehn, A. Tanatmis, S. Ruzika, and H. W. Hamacher, "Calculating the minimum distance of linear block codes via integer programming," in *Proc. 6th Int. Symp. Turbo Codes Iterative Inf. Process.*, Sep. 2010, pp. 329–333.
[23] M. Schwartz and A. Vardy, "On the stopping distance and the stopping redundancy of codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 922–932, Mar. 2006.
[24] A. Abbasfar, D. Divsalar, and K. Yao, "Accumulate-repeat-accumulate codes," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 692–702, Apr. 2007.
[25] T. Richardson, "Multi-edge type LDPC codes," presented at the Workshop Honoring Prof. Bob McEliece 60th Birthday, Pasadena, CA, USA: California Institute of Technology, May 2002.

AHMED ELKELESH received the B.Sc. degree (Hons.) in information engineering and technology from the German University in Cairo, in 2013, and the M.Sc. degree (Hons.) in communications engineering and media technology from the University of Stuttgart, in 2016, where he is currently pursuing the Ph.D. degree. During his years of study in Germany, he was a Research Assistant with Fraunhofer IPA Stuttgart, and also an Intern with the Sony Stuttgart Technology Center. Since 2016, he has been a member of the Research Staff with the Institute of Telecommunications, University of Stuttgart. His main research topic is channel coding, with the particular emphasis on polar codes and LDPC codes. Furthermore, his research interests include the areas of information theory, modulation, machine learning, and SDR. He was a recipient of the Anton-und-Klara-Röser prize 2017 for his master thesis.

MOUSTAFA EBADA received the B.Sc. degree (Hons.) from the Communications Department, German University in Cairo, Egypt, in 2013, and the M.Sc. degree in electrical engineering and information technology from the University of Stuttgart, Germany, in 2016, where he is currently pursuing the Ph.D. degree. During his master studies, he was a Research Assistant with multiple institutes of the University of Stuttgart, including fields of radio frequency technology, signal processing, and telecommunications. Since 2016, he has been a member of Research Staff with the Institute of Telecommunications, University of Stuttgart. His main research topics are channel coding, particularly polar code construction and decoding, and designing short LDPC codes for high speed applications. Furthermore, his research interest includes machine learning, particularly designing error correction codes and utilization of the state-of-the-art coding schemes in the field of multiple access channel.

SEBASTIAN CAMMERER (S'16) received the B.Sc. and M.Sc. degrees (Hons.) in electrical engineering and information technology from the University of Stuttgart, Germany, in 2013 and 2015, respectively, where he is currently pursuing the Ph.D. degree. During his years of study, he worked as a Research Assistant at multiple institutes of the University of Stuttgart. Since 2015, he has been a member of the research staff at the Institute of Telecommunications, University of Stuttgart. His main research topics are channel coding and machine learning for communications. Furthermore, his research interests are in the areas of modulation, parallelized computing for signal processing, and information theory. He is a recipient of the Best Publication Award of the University of Stuttgart 2019, the Anton- und Klara Röser Preis 2016, the Rohde&Schwarz Best Bachelor Award 2015, and the VDE-Preis 2016 for his master thesis.

LAURENT SCHMALEN received the Dipl.Ing. degree in electrical engineering and information technology, and the Dr. Ing. degree from the RWTH Aachen University of Technology, Aachen, Germany. From 2011 to 2019, he was with Bell Labs as a member of technical staff, and also as the Department Head of the Coding in Optical Communications Department, from 2016 to 2019. From 2014 to 2019, he was a Guest Lecturer with the University of Stuttgart, Germany. He is currently a Professor with the Karlsruhe Institute of Technology (KIT), where he leads the Communications Engineering Lab (CEL). His research interests include forward error correction, modulation formats, and information theory for future optical networks. He is a recipient and co-recipient of several awards, including the E-Plus Award for his Ph.D. thesis, the Best Paper Award of the 2010 ITG Speech Communication Conference, the 2013 Best Student Paper Award at the IEEE Signal Processing Systems Workshop, and the 2016 Journal of Lightwave Technology Best Paper Award. In addition, he received the 2014 IEEE Transactions on Communications Exemplary Reviewer Award and the 2019 Journal of Lightwave Technology Outstanding Reviewer Recognition.

STEPHAN TEN BRINK (M'97–SM'11) has been a Faculty Member of the University of Stuttgart, Germany, since July 2013, where he is currently the Head of the Institute of Telecommunications. From 1995 to 1997 and from 2000 to 2003, he was with Bell Laboratories, Holmdel, NJ, USA, conducting research on multiple antenna systems. From July 2003 to March 2010, he was with Realtek Semiconductor Corp., Irvine, CA, USA, as the Director of the Wireless ASIC Department, developing WLAN and UWB single chip MAC/PHY CMOS solutions. In April 2010, he returned to Bell Laboratories as the Department Head of the Wireless Physical Layer Research Department, Stuttgart, Germany. He is a recipient and co-recipient of several awards, including the Vodafone Innovation Award, the IEEE Stephen O. Rice Paper Prize, and the IEEE Communications Society Leonard G. Abraham Prize for contributions to channel coding and signal detection for multiple-antenna systems. He is best known for his work on iterative decoding (EXIT charts) and MIMO communications (soft sphere detection and massive MIMO).

• • •