

Multilevel LDPC-Coded High-Speed Optical Systems: Efficient Hard Decoding and Code Optimization

Chen Gong and Xiaodong Wang, *Fellow, IEEE*

Abstract—We consider a multilevel coding scheme employing low-density parity-check (LDPC) codes and high-order modulations for high-speed optical transmissions, where the coherent receiver performs either parallel independent decoding (PID) or multistage decoding (MSD). To meet the severe complexity constraint imposed by the ultrahigh data rate of the emerging optical transmission systems, we focus on hard-decision decoding of LDPC codes. A new LDPC hard decoding method is developed, which is equivalent to the Gallager decoding algorithm B, but is more efficient in terms of circuit implementation, since no variable node degree information is needed. Two variants of this decoder is also proposed, which offers significant performance gain for finite-length codes. We optimize the system by allocating rates and designing profiles for component codes. Both numerical evaluations and simulation results show that the optimized multilevel coding systems with either PID or MSD substantially outperform the optimized single-level LDPC-coded system.

Index Terms—Code optimization, hard-decision decoding, low-density parity-check (LDPC) codes, multilevel coding, optical communications.

I. INTRODUCTION

RECENTLY, high-order modulation formats, such as M-ary phase-shift keying (M-PSK) and M-ary quadrature amplitude modulation (M-QAM), have been proposed for optical transmission systems to obtain higher spectral efficiency [1]–[4]. Moreover, coherent systems are gaining interest due to the availability of high-speed signal processing and low-priced components, as well as the partly relaxed receiver requirements at high data rates [5]–[7]. Coherent receivers can exploit all optical field parameters (amplitude, phase, frequency, and polarization) in the electrical domain and permit to reach the ultimate limits of spectral efficiency. On the other hand, recent works have also considered applications of turbo [8], [9] and low-density parity-check (LDPC) codes [10]–[12] to optical communications. These codes offer capacity approaching performance when the code word length is very large.

The very high information rate that needs to be sustained by the emerging optical transmission systems, e.g., 40–100 Gb/s, poses a severe complexity constraint on the decoder. The soft decoding algorithms that are traditionally associated with turbo

or LDPC codes may be too complex for such systems. In this paper, therefore, we focus on hard decoding methods. The original LDPC hard decoders are given in [13] as the Gallager decoding algorithm A and B (GA and GB). Other LDPC hard decoding methods include the majority-based (MB) time-invariant decoding algorithms [14], the probabilistic flipping algorithms [15], and several switch-type hybrid algorithms [16]. On the other hand, although GB is optimal for decoding infinite-length codes, it suffers performance degradation while decoding finite-length codes [16]. The expanded optimal switch algorithms for regular codes are proposed in [16], which provide significant performance improvement over GB. However, for irregular codes, the existing hard decoding algorithms require the degree information of variable nodes. Such requirement significantly increases the circuit-level implementation complexity due to the operations of storing and retrieving the degree information.

In this paper, we consider an LDPC-based multilevel coding [17] scheme for high-speed optical systems employing high-order modulation and coherent detection. To keep the decoding complexity low, the LDPC decoder performs hard-decision decoding based on the hard-demodulated bits from the coherent detector output. Different from the multilevel coding scheme in [11], where the receiver performs parallel independent decoding (PID) based on high-complexity soft decoding, in this paper, we focus on efficient hard decoding, which is more practical for ultrahigh throughput systems, and consider both PID and multistage decoding (MSD). We propose a new efficient switch type hard-decision decoding algorithm that is equivalent to GB, but does not require the degree information of variable nodes, leading to a significantly simplified circuit implementation compared with the existing hard decoders. We also give two variants of the proposed decoder that significantly outperforms the GB decoder for finite-length codes. Moreover, for a given modulation scheme, we optimize the multilevel coding for both PID and MSD by allocating rates and designing profiles for component codes. The single-level coding, where only one optimized code is employed for all the mapping bits, is used as the benchmark for performance comparisons. Both numerical evaluations and simulation results show that for various high-order modulation schemes of interest, the optimized multilevel coding with PID/MSD outperforms the optimized single-level coding.

The remainder of this paper is organized as follows. In Section II, we present the system descriptions. In Section III, we analyze the potential performance gains offered by PID and MSD over the single-level coding. In Section IV, we develop a

Manuscript received August 13, 2009; revised September 1, 2009; accepted September 29, 2009. Date of publication December 1, 2009; date of current version October 6, 2010. This work was supported in part by the U.S. National Science Foundation under Grant CCF-0726480.

The authors are with the Department of Electrical Engineering, Columbia University, New York, NY 10027 USA (e-mail: chengong@ee.columbia.edu; wangx@ee.columbia.edu).

Digital Object Identifier 10.1109/JSTQE.2009.2034025

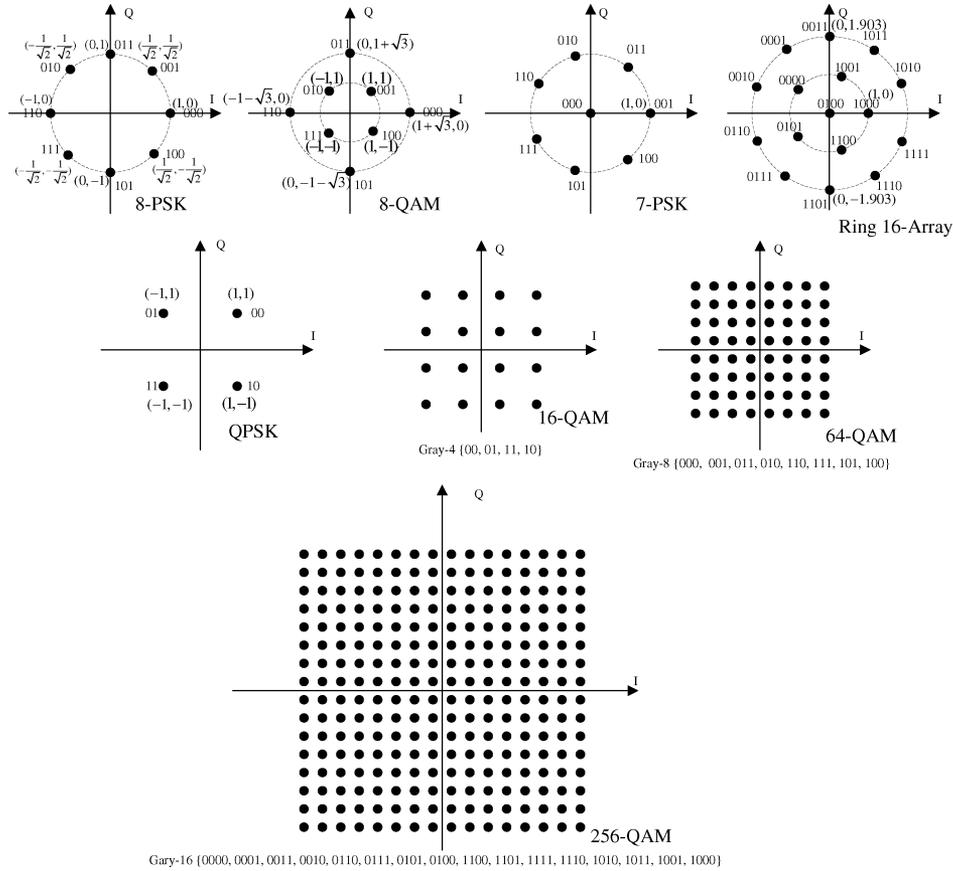


Fig. 1. Mappings for various modulation schemes.

new efficient LDPC hard decoding algorithm and its variants for finite-length codes. In Section V, we develop the optimization of the proposed multilevel coding system with either PID or MSD. Simulation results are given in Section VI. Section VII contains the conclusions.

II. SYSTEM DESCRIPTIONS

A. Multilevel Coding

The various modulation signal constellations considered in this paper are illustrated in Fig. 1, where each signal point is labeled by its mapping bits. The 64-QAM and 256-QAM are employed in [1]–[4] to increase the spectrum efficiency in high SNR regions.

Five of these constellations can be Gray mapped, i.e., 8-PSK, QPSK, 16-QAM, 64-QAM, and 256-QAM. In particular, for the M^2 -QAM ($M = 4, 8, 16$), the signal points lie at $\{(2i - M + 1, 2j - M + 1), 0 \leq i, j \leq M - 1\}$. For the signal point at $(2i - M + 1, 2j - M + 1)$, the first-half of the mapping bits are the Gray- M mapping for i and second-half are the Gray- M mapping for j . The Gray- M mappings of the indexes from 0 to $M - 1$ are shown in Fig. 1 along with the M^2 -QAM constellations. However, for 7-PSK, 8-QAM, and Ring 16-Array, Gray mapping is not possible. In particular, for 7-PSK, besides the signal point (000), which lies at the origin,

the other seven signal points lie uniformly on the unit circle. For the Ring 16-Array, one signal point lies at the origin, five and ten signal points lie uniformly on the circle of radii 1 and 1.902, respectively.

The proposed multilevel coding scheme is as follows. Suppose for a particular constellation, n bits (b_1, b_2, \dots, b_n) are mapped to a signal point. The set of the n mapping bits are partitioned into m nonoverlapping subsets $S^{(i)}$ for $1 \leq i \leq m$, where $|S^{(i)}| = n^{(i)}$. We employ a code $C^{(i)}$ of rate $r^{(i)}$ for the mapping bits in subset $S^{(i)}$. The overall code rate is given by

$$R = \frac{\sum_{i=1}^m r^{(i)} n^{(i)}}{n}. \quad (1)$$

Note that the special case of $m = 1$ corresponds to the traditional single-level coding. The specific bit partitions for the constellations in Fig. 1 are given in Section III-C.

B. Decoding Strategies

Let the transmitted symbol at time k be s_k . We consider an effective additive white Gaussian noise (AWGN) channel, e.g., the output of a linear or nonlinear equalizer in a single-carrier dispersive optical channel or the single-tap equalizer output of each subcarrier in an optical orthogonal frequency-division multiplex system. The received signal at time k is $y_k = s_k + n_k$, where $n_k \sim \mathcal{N}(0, \sigma^2)$ is the independent identically

distributed (i.i.d.) noise sample. Due to the very high throughput requirement of the underlying optical system, in this paper, we focus on hard-decision decoding techniques. In particular, we consider the following two decoding approaches.

PID: We find the constellation signal points \hat{s}_k closest to y_k , demultiplex the mapping bits of \hat{s}_k into m levels according to the encoding scheme $(S^{(i)}, C^{(i)})$ for $i = 1, \dots, m$, and perform the hard-decoding for each component code $C^{(i)}$ simultaneously.

MSD: The decoding is performed in m stages, where in the i th stage, we decode the mapping bits in $S^{(i)}$. In the first stage, we find the signal points $\hat{s}_k^{(1)}$ closest to y_k , retrieve the mapping bits of $\hat{s}_k^{(1)}$ corresponding to $S^{(1)}$, and perform the decoding of $C^{(1)}$. In the i th stage ($2 \leq i \leq m$), we find the signal points $\hat{s}_k^{(i)}$ closest to y_k , subject to the constraint that the mapping bits of $\hat{s}_k^{(i)}$ corresponding to $S^{(1)}$ to $S^{(i-1)}$ are equal to the corresponding decoded bits in the preceding $(i-1)$ stages, and perform the decoding of $C^{(i)}$.

In the next section, we will show that compared with the PID, smaller error rates for the bits in $S^{(i)}$ can be achieved by the MSD.

III. ANALYSIS OF MULTILEVEL CODING

In this section, we show that for the proposed multilevel coding, the MSD in general offers better performance than the PID in terms of bit error rate. Furthermore, the multilevel coding with PID outperforms the traditional single-level coding. The component LDPC codes at different levels are optimized based on the equivalent crossover channel error probability in corresponding levels. The code optimization procedures will be discussed in Section V.

A. PID versus MSD

Here, we show that the MSD outperforms the PID. Let s be the transmitted signal and $s(k)$ be the k -th mapping bit for s .

The key point here is to demonstrate that, given the already decoded bits $s(k_j)$ for $1 \leq j \leq p$ and $1 \leq k_j \leq n$, if we demodulate $s(k)$ ($k \neq k_j$) by finding the signal point \hat{s} closest to the received signal y subject to the constraint that $\hat{s}(k_j) = s(k_j)$ for $1 \leq j \leq p$, the bit error rate will be smaller than if we find the signal point \hat{s} closest to y without any constraint as in PID.

We start from the following union bound, which gives an approximation to the error probability p_k for the bit b_k [18],

$$p_k \leq \sum_{d \leq d_{\text{th}}} \frac{N_k(d)}{2^n} Q\left(\frac{d}{2\sigma}\right), \quad (2)$$

where d_{th} is a threshold value that dominates the error probability, $N_k(d)$ is the number of signal point pairs that differ in bit b_k with Euclidian distances less than or equal to d , and σ^2 is the noise variance. For each signal point, there is one shortest Euclidian distance between it and other signal points. Usually, d_{th} is chosen to be the maximum of the above shortest distances corresponding to all signal points. We define

$$R(k, d_{\text{th}}) \triangleq \{(s_i, s_j), d(s_i, s_j) \leq d_{\text{th}}, s_i(k) \neq s_j(k)\}, \quad (3)$$

which is a metric related to the error probability for the bit b_k for PID. We have $|R(k, d_{\text{th}})| = \sum_{d \leq d_{\text{th}}} N_k(d)$.

Next, we consider the MSD. We partition the 2^n signal points in the constellation into 2^p nonoverlapping subsets, denoted as A_1, A_2, \dots, A_{2^p} , according to the different combined values of the mapping bits b_{k_1}, \dots, b_{k_p} . If we decode b_k by finding \hat{s} closest to y subject to the constraint that $s(k_j) = \hat{s}(k_j)$ for $1 \leq j \leq p$, then the decoding errors are dominated by the set $R(k, d_{\text{th}}) \cap (\bigcup_{k=1}^{2^p} A_k \times A_k)$, where $A_k \times A_k = \{(s_i, s_j) | s_i, s_j \in A_k\}$ for $1 \leq k \leq 2^p$. This is because once the mapping bits b_{k_l} for $1 \leq l \leq p$ are correctly decoded, the uncertainty for b_k only exists among the pairs of signal points in $R(k, d_{\text{th}})$ with the same values of b_{k_l} for $1 \leq l \leq p$.

Since $|R(k, d_{\text{th}}) \cap (\bigcup_{k=1}^{2^p} A_k \times A_k)| \leq |R(k, d_{\text{th}})|$, the MSD is no worse than the PID. If $|R(k, d_{\text{th}}) \cap (\bigcup_{k=1}^{2^p} A_k \times A_k)| < |R(k, d_{\text{th}})|$, then the error rate for decoding b_k with the known bits b_{k_l} for $1 \leq l \leq p$ will be smaller than that if we decode b_k without any constraint in PID. Therefore, for the same channel E_s/N_0 , the MSD can achieve a larger code rate than PID. On the other hand, given the same code rate, the MSD can achieve a smaller E_s/N_0 threshold than PID.

For Gray-mapped constellations, since the pairs of the closest signal points that differ at the mapping bit b_k actually differ only at b_k , we have $|R(k, d_{\text{th}}) \cap (\bigcup_{k=1}^{2^p} A_k \times A_k)| = |R(k, d_{\text{th}})|$. The uncertainty for b_k cannot be reduced by decoding other bits, so the MSD provides little performance improvement compared with PID. Therefore, the MSD significantly outperforms the PID for the constellations for which the Gray mapping is not possible, e.g., 7-PSK, 8-QAM, and Ring 16-Array in Fig. 1.

Consider the 8-QAM in Fig. 1. Since the shortest distance from one signal point to any other point is always 2, we let $d_{\text{th}} = 2$. It can be computed that $|R(1, d_{\text{th}})| = |R(2, d_{\text{th}})| = 8$ and $|R(3, d_{\text{th}})| = 16$. If we partition the signal points into four subsets A_1, \dots, A_4 , where the signal points with the same $(b_1 b_2)$ are grouped into one subset, then, we have $|R(k, d_{\text{th}}) \cap (\bigcup_{k=1}^4 A_k \times A_k)| = 8$. In simulations for PID, the bit error rates for b_1, b_2 , and b_3 are of the ratio 1:1:2, which validates the use of metric $|R(k, d_{\text{th}})|$ given by (3). Moreover, with the known values of bits b_1 and b_2 , the bit error rate for b_3 is about half of that for b_3 in the PID. Therefore, if we first decode b_1 and b_2 , and then decode b_3 , the component code for b_3 can sustain a larger rate than in the PID.

B. PID versus Single-Level Coding

Next, we show that the PID for multilevel coding outperforms the single-level coding. Let $r(e)$ be the rate of the optimized profile for the channel crossover error probability e . We first argue that the function $r(e)$ is convex.

Consider a two-state Gilbert channel with the crossover error probability e_1 with probability p and the crossover error probability e_2 with probability $1-p$. With the perfect channel state information (CSI), we can design the channel codes for both channel states. The average transmission rate with perfect CSI is then $pr(e_1) + (1-p)r(e_2)$. On the other hand, if the CSI is not available, we can design one channel code for

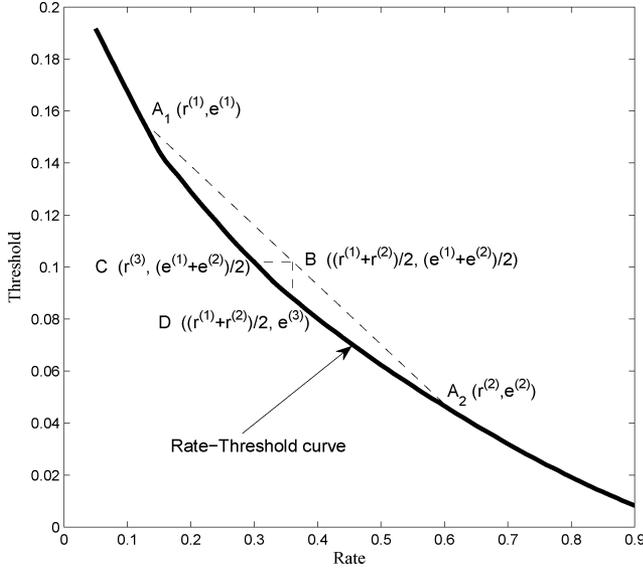


Fig. 2. Rate-threshold curve for the optimized profiles under the proposed decoder.

the average error probability $pe_1 + (1-p)e_2$, resulting in a transmission rate of $r(pe_1 + (1-p)e_2)$. For the optimal coding scheme, $r(e) = 1 - H_2(e)$, where $H_2(e)$ is the binary entropy function given by $H_2(e) = -e \log_2(e) - (1-e) \log_2(1-e)$. Since $H_2(e)$ is convex, it follows that $r(pe_1 + (1-p)e_2) \leq pr(e_1) + (1-p)r(e_2)$. Thus, with perfect CSI the transmission rate is higher than that without CSI. By employing an optimized code for an efficient decoder, we expect this conclusion also holds true, i.e., $r(pe_1 + (1-p)e_2) \leq pr(e_1) + (1-p)r(e_2)$, and hence, the function $r(e)$ is convex. This is confirmed by the rate-threshold curve in Fig. 2 for the LDPC hard decoding algorithm discussed in Section IV, where the threshold optimization is described in Section V. Next, we justify that the multilevel coding with PID is superior to the single-level coding.

Let $\alpha^{(i)} = |S^{(i)}|/|S| = n^{(i)}/n$. At a given E_s/N_0 , suppose that the bits in $S^{(i)}$ suffer the error probability $e^{(i)}$ for $1 \leq i \leq m$. For PID, since the rate for the optimized component code $C^{(i)}$ is given by $r^{(i)} = r(e^{(i)})$, the average code rate, denoted as R_{PID} , is given by

$$R_{\text{PID}} = \sum_{i=1}^m \alpha^{(i)} r(e^{(i)}). \quad (4)$$

For the single-level coding, since the average error probability for all the mapping bits is $\sum_{i=1}^m \alpha^{(i)} e^{(i)}$, the rate of the optimized code is given by

$$R_{\text{SLC}} = r(e_{\text{ave}}) = r\left(\sum_{i=1}^m \alpha^{(i)} e^{(i)}\right). \quad (5)$$

Since $r(e)$ is convex, we have

$$R_{\text{SLC}} = r\left(\sum_{i=1}^m \alpha^{(i)} e^{(i)}\right) \leq \sum_{k=1}^m \alpha^{(k)} r(e^{(k)}) = R_{\text{PID}}. \quad (6)$$

Therefore, for a given E_s/N_0 a larger code rate can be achieved by the PID than the single-level coding. On the other hand, for

a given average code rate, the PID can achieve a smaller E_s/N_0 threshold.

As an example, Fig. 2 illustrates the performance improvement offered by the PID. The rate-threshold curve is plotted for the optimized LDPC code profile for each rate under the hard decoding algorithm discussed in Section IV. Suppose two subsets of bits $S^{(1)}$ and $S^{(2)}$, where $\alpha^{(1)} = \alpha^{(2)} = 0.5$, have the bit error rates $e^{(1)}$ and $e^{(2)}$, respectively. The corresponding optimized codes are of the rates $r^{(1)} = r(e^{(1)})$ and $r^{(2)} = r(e^{(2)})$, shown as $A_1(r^{(1)}, e^{(1)})$ and $A_2(r^{(2)}, e^{(2)})$, respectively. Let $B((r^{(1)} + r^{(2)})/2, (e^{(1)} + e^{(2)})/2)$ denote the rate of the PID and the average bit error rate for $S^{(1)}$ and $S^{(2)}$. Since it lies above the rate-threshold curve, it is not achievable by the single-level coding, for which smaller code rate $r^{(3)}$ is needed to achieve the threshold $(e^{(1)} + e^{(2)})/2$, and the rate $(r^{(1)} + r^{(2)})/2$ can only achieve the smaller threshold $e^{(3)}$, shown as C and D , respectively.

C. Partition of Mapping Bits

For the PID, we employ the Monte Carlo simulations to evaluate the error rate for each mapping bits, and group the mapping bits suffering the same error rate into one subset $S^{(i)}$. Alternatively, the partition can also be performed based on the cardinality of the sets $R(k, d_{\text{th}})$ given in (3) directly. Specifically, we group the bits b_k with the same $|R(k, d_{\text{th}})|$ into one subset $S^{(i)}$. For the MSD, we first decode the mapping bits b_k with smaller $|R(k, d_{\text{th}})|$, and then, decode these with larger $|R(k, d_{\text{th}})|$.

Table I lists the partitions of mapping bits for the modulation schemes in Fig. 1 under both PID and MSD. For MSD, the superscripts of the subsets indicate the decoding order of the mapping bits. For the 8-PSK, 16-QAM, 64-QAM, and 256-QAM, the MSD offers little performance improvement over the PID, since the Gray mapping is employed.

IV. EFFICIENT LDPC HARD DECODING

We consider an irregular LDPC code with N variable nodes and M check nodes. Assume that we first perform a hard decision from the channel output and obtain $u_n \in \{+1, -1\}$, the binary message of the n th variable node from the channel. Denote $u_{c \rightarrow v, j}$ and $v_{v \rightarrow c, j}$ as respectively, the binary extrinsic message from a check node passing along the j th edge to a variable node and the binary extrinsic message from a variable node passing along the j th edge to a check node. Define \mathcal{V}_n as the set of the edges connected to the n th variable node and \mathcal{U}_m as the set of the edges connected to the m th check node. Obviously, $|\mathcal{V}_n|$ and $|\mathcal{U}_m|$ are the degrees of the n th variable node and the m th check node, respectively.

The ensemble of an irregular LDPC code can be specified by two polynomials

$$\lambda(x) = \sum_{j=2}^{D_L} \lambda_j x^{j-1} \quad \text{and} \quad \rho(x) = \sum_{j=2}^{D_R} \rho_j x^{j-1}$$

where λ_j and ρ_j are the fractions of edges in the bipartite code graph that are connected to variable nodes of degree j and check nodes of degree j , respectively; D_L and D_R denote the

TABLE I
PARTITION OF MAPPING BITS FOR THE MODULATION SCHEMES IN FIG. 1

Modulation	d_{th}	Mapping bits partition
8-PSK	$2 \sin(\pi/8)$	$S^{(1)} = \{b_1, b_2\}, S^{(2)} = \{b_3\}$
8-QAM	2	$S^{(1)} = \{b_1, b_2\}, S^{(2)} = \{b_3\}$
16-QAM	2	$S^{(1)} = \{b_1, b_3\}, S^{(2)} = \{b_2, b_4\}$
64-QAM	2	$S^{(1)} = \{b_1, b_4\}, S^{(2)} = \{b_2, b_5\}, S^{(3)} = \{b_3, b_6\}$
256-QAM	2	$S^{(1)} = \{b_1, b_5\}, S^{(2)} = \{b_2, b_6\}, S^{(3)} = \{b_3, b_7\}, S^{(4)} = \{b_4, b_8\}$
7-PSK	1	$S^{(1)} = \{b_1, b_2\}, S^{(2)} = \{b_3\}$
Ring 16-Array	$2 \sin(\pi/5)$	$S^{(1)} = \{b_1, b_2\}, S^{(2)} = \{b_3, b_4\}$

maximum degree of variable nodes and check nodes, respectively. Equivalently, the degree profiles can also be specified from the node perspective by two polynomials

$$\tilde{\lambda}(x) = \sum_{j=2}^{D_L} \tilde{\lambda}_j x^{j-1} \quad \text{and} \quad \tilde{\rho}(x) = \sum_{j=2}^{D_R} \tilde{\rho}_j x^{j-1}$$

where $\tilde{\lambda}_j$ and $\tilde{\rho}_j$ are the fractions of variable and check nodes of degree j , respectively. Note that the profile for variable nodes starts from the degree 2 variable nodes. This is because the extrinsic messages for degree variable nodes is exactly the channel inputs, and thus, their error probability always equals to the channel error probability, which prevents the convergence of error probability to zero. The check nodes profile should also start from degree 2; otherwise coded bits corresponding to the variable nodes connected to the degree-1 check nodes are forced to be zero.

In what follows, we summarize the GB hard decoding algorithm. Then, we present a new hard decoding method that is equivalent to GB, but is amenable to more efficient implementations for irregular codes. Two variants of the proposed decoder with improved performance for finite-length codes is also proposed.

A. GB Decoder

The GB decoder is a generalized switch-type binary message-passing algorithm based on the majority vote, which can be summarized as follows.

- 1) Initialization: $v_{v \rightarrow c, k}^{(0)} = u_n, k \in \mathcal{V}_n, n = 1, \dots, N$.
- 2) Check node decoding:

$$u_{c \rightarrow v, k}^{(\ell)} = \prod_{l \in \mathcal{U}_m, l \neq k} v_{v \rightarrow c, l}^{(\ell)}, k \in \mathcal{U}_m, m = 1, \dots, M. \quad (7)$$

- 3) Variable node decoding:

$$v_{v \rightarrow c, k}^{(\ell+1)} = \begin{cases} -u_n & \text{if } |\Omega_k^{(\ell)}| \geq \nu_n^{(\ell)}, \\ u_n, & \text{otherwise,} \end{cases} \quad (8)$$

where $\Omega_k^{(\ell)} = \{l : u_{c \rightarrow v, l}^{(\ell)} = -u_n, l, k \in \mathcal{V}_n, l \neq k\}$ is the set of disagreements against the channel input and $\nu_n^{(\ell)}$ denotes the flipping threshold of the n th variable node at the ℓ th iteration, $\nu_n^{(\ell)} \in \{\lceil |\mathcal{V}_n|/2 \rceil, \lceil |\mathcal{V}_n|/2 \rceil + 1, \dots, |\mathcal{V}_n| \}$.

It is seen that in the above decoding algorithm, when the number of input extrinsic messages that disagree with the channel input is above the threshold value $\nu_n^{(\ell)}$, the variable node decoder flips the channel input as the extrinsic output. Otherwise,

the channel input is set as the extrinsic output. Moreover, the threshold value $\nu_n^{(\ell)}$ for flipping is constrained by the degree $|\mathcal{V}_n|$. Denote $p^{(\ell)}$ as the error probability of the extrinsic messages along the edges at the ℓ th decoding iteration. Given the initial error probability p_0 from the channel and the extrinsic error probability $p^{(\ell)}$, the optimal $\nu_n^{(\ell+1)}$ for the next iteration is the smallest $\nu_n^{(\ell+1)} \in \{\lceil |\mathcal{V}_n|/2 \rceil, \lceil |\mathcal{V}_n|/2 \rceil + 1, \dots, |\mathcal{V}_n| \}$ that satisfies the following inequality [13], [19]:

$$\frac{1 - p_0}{p_0} \leq \left(\frac{1 + \rho(1 - 2p^{(\ell)})}{1 - \rho(1 - 2p^{(\ell)})} \right)^{2\nu_n^{(\ell+1)} - (|\mathcal{V}_n| - 1)}, \quad (9)$$

where $\rho(\cdot)$ is the degree polynomial function for check node as previously defined for the LDPC code ensemble. For a variable node with degree j , we denote the corresponding solution to (9) as $b_j^{(\ell+1)}$.

B. Proposed LDPC Hard Decoder

The conventional implementation of the GB decoder requires the degree information, since as shown in (8), the flipping threshold at the variable node of degree $|\mathcal{V}_n|$ takes values from $\lceil |\mathcal{V}_n|/2 \rceil$ to $|\mathcal{V}_n|$. Such a degree information requirement can significantly complicate the circuit implementation of the decoder. We now present an implementation-friendly LDPC hard decoding method that does not require the degree information of variable nodes for irregular codes. The proposed method shares the same decoding rule at the check nodes as previously described. At the variable nodes, we first find the discrepancy of the extrinsic inputs, given by

$$m_k^{(\ell)} = -u_n \sum_{j \in \mathcal{V}_n, j \neq k} u_{c \rightarrow v, j}^{(\ell)}, k \in \mathcal{V}_n. \quad (10)$$

The decoding rule at the variable nodes is then given by

$$v_{v \rightarrow c, k}^{(\ell+1)} = \begin{cases} -u_n & \text{if } m_k^{(\ell)} \geq d^{(\ell)}, \\ u_n, & \text{otherwise,} \end{cases} \quad (11)$$

where $d^{(\ell)}$ is the flipping threshold that is the same for all variable nodes but varies during the decoding iterations.

Note that there are alternative ways to obtain $m_k^{(\ell)}$ in (10) for the proposed decoding rule, such as for $j \in \mathcal{V}_n, j \neq k$,

$$\begin{cases} m_k^{(\ell)} \leftarrow m_k^{(\ell)} + 1, & \text{if } u_{c \rightarrow v, j}^{(\ell)} \neq u_n, \\ m_k^{(\ell)} \leftarrow m_k^{(\ell)} - 1, & \text{if } u_{c \rightarrow v, j}^{(\ell)} = u_n. \end{cases} \quad (12)$$

We can choose the form with the lowest complexity for circuit level implementations.

Based on (8) and (11), we now summarize the relationship between the flipping threshold $d^{(\ell)}$ in the proposed method and $b_j^{(\ell)}$ in the GB decoder that produce the same decoding output for a particular degree j . Denote the number of extrinsic inputs that disagree with the channel input as $m_{j,D}$ and the number of extrinsic agreements as $m_{j,A}$. Since $m_{j,D} + m_{j,A} = j - 1$, we have that the discrepancy in (10) $m_j = m_{j,D} - m_{j,A} = 2m_{j,D} - (j - 1)$. Let us look at the exponent $2\nu_n^{(\ell+1)} - (|\mathcal{V}_n| - 1)$ in (9), where $\nu_n^{(\ell+1)} = m_{j,D}$ and $|\mathcal{V}_n| = j$. It follows that the flipping threshold in terms of discrepancy is given by

$$d^{(\ell)} = 2b_j^{(\ell)} - (j - 1), \quad (13)$$

where $b_j^{(\ell)}$ is the flipping threshold in terms of the number of agreeing extrinsic binary messages.

Using (9) and (13), we can easily obtain the best $d^{(\ell+1)}$ as the smallest $d^{(\ell+1)} \in \{0, 1, \dots\}$ that satisfies the following inequality [19]:

$$\frac{1 - p_0}{p_0} \leq \left(\frac{1 + \rho(1 - 2p^{(\ell)})}{1 - \rho(1 - 2p^{(\ell)})} \right)^{d^{(\ell+1)}}. \quad (14)$$

It is seen that $d^{(\ell)}$ can be obtained without the degree information.

C. Computing the Flipping Thresholds

1) *Extrinsic Error Probability Analysis:* In the proposed hard decoder, the optimal flipping threshold $d^{(\ell)}$ for variable node decoding has to be obtained in every decoding iteration. One approach is to use the predetermined sequence of $d^{(\ell)}$ analytically obtained based on the initial error probability p_0 from the channel. With p_0 , given the degree profiles of the LDPC codes, we can track the extrinsic error probability (EEP) along the edges, which is the same as the elementary extrinsic information transfer (EXIT) functions given in [20]. We call it EEP function here in this paper because it describes the error probability of the extrinsic 0–1 binary messages in iterative hard-decision decoding. The EEP function $f_{j,k}(p_0, x)$ describes the output extrinsic error probability for degree- j variable nodes after one decoding iteration with input extrinsic error probability x (e.g., $x = p^{(\ell)}$), the channel error p_0 , and the flipping threshold k (e.g., $k = b_j^{(\ell)}$) [13]. Define $\xi(x) \triangleq (1 + \rho(1 - 2x))/2$. We then have [13]

$$\begin{aligned} f_{j,k}(p_0, x) &= p_0 - p_0 \sum_{l=k}^{j-1} \binom{j-1}{l} \xi(x)^l (1 - \xi(x))^{j-1-l} \\ &\quad + (1 - p_0) \sum_{l=k}^{j-1} \binom{j-1}{l} (1 - \xi(x))^l \xi(x)^{j-1-l}. \end{aligned} \quad (15)$$

Denote $s_{k_1, k_2}(p_0, j)$ as the x -value of the intersection point of two functions, $f_{j, k_1}(p_0, x)$ and $f_{j, k_2}(p_0, x)$, i.e., $f_{j, k_1}(p_0, s_{k_1, k_2}(p_0, j)) = f_{j, k_2}(p_0, s_{k_1, k_2}(p_0, j))$. The optimal switching points for degree- j nodes in GB decoding are at $x = s_{k, k+1}(p_0, j)$, $k = \lceil j/2 \rceil, \dots, j - 2$, i.e., the threshold switches

from $b_j^{(\ell)} = k + 1$ to $b_j^{(\ell+1)} = k$, if $p^{(\ell)} \leq s_{k, k+1}(p_0, j)$ and $p^{(\ell-1)} > s_{k, k+1}(p_0, j)$ [16]. It has been shown that there are only two sets of the optimal switching points represented by $\{u_{\omega, o}(p_0)\}$ for odd- j and $\{u_{\omega, e}(p_0)\}$ for even- j and that $u_{\omega, e}(p_0), u_{\omega, o}(p_0)$ are the roots of [14]

$$\frac{p_0}{1 - p_0} \left(\frac{1 + \rho(1 - 2x)}{1 - \rho(1 - 2x)} \right)^d = 1 \quad (16)$$

with $d = 2\omega + 1$ and $d = 2\omega + 2$, respectively, and $u_{\omega, e}(p_0) < u_{\omega, o}(p_0) < u_{\omega+1, e}(p_0) < u_{\omega+1, o}(p_0)$ for $\omega \geq 0$. Denote $\{v_d(p_0)\}$ as the roots of (16) for various d 's. We have $v_{2\omega+1}(p_0) = u_{\omega, e}(p_0, e)$, $v_{2\omega+2}(p_0) = u_{\omega, o}(p_0)$, and thus, $v_{2\omega+1}(p_0) < v_{2\omega+2}(p_0) < v_{2\omega+3}(p_0) < v_{2\omega+4}(p_0)$. We can see that changing the flipping threshold from $d = 2\omega + 1$ to 2ω does not change the decoding output of all variable nodes with even degrees, since the optimal switching points are the roots of (16) with $d = 2\omega + 1$. Consequently, it does not change the component EEP function that represents the decoding of variable nodes with even degrees. Similarly, changing d from $2\omega + 2$ to $2\omega + 1$ does not change the decoding output of all variable nodes with odd degrees. The optimal switching points for irregular LDPC codes with the proposed efficient decoder are $\{v_d(p_0), d = 1, 2, \dots\}$. We can see that the set of optimal switching points for the proposed efficient decoder, $\{v_d^{p_0}\}$, is exactly the same as the one for the GB decoder $\{u_{\omega, e}(p_0), u_{\omega, o}(p_0)\}$. Hence, the proposed efficient decoder performs exactly the same as the GB decoder. The advantage of the proposed method is that the optimal flipping threshold in (11) does not depend on the degrees of variable nodes, that is, $\{v_d^{p_0}\}$ does not depend on j , which facilitates efficient implementation of the decoder.

The EEP function of degree- j nodes for the proposed decoder is then given by

$$h_j(p_0, x) = \begin{cases} f_{j, \lfloor (j+1)/2 \rfloor}(p_0, x), & 0 < x \leq v_1(p_0), \\ f_{j, \lfloor (j+2)/2 \rfloor}(p_0, x), & v_1(p_0) < x \leq v_2(p_0), \\ \vdots, & \vdots \\ f_{j, \lfloor (2j-1)/2 \rfloor}(p_0, x), & v_{j-2}(p_0) < x. \end{cases} \quad (17)$$

Define $v_0(p_0) \triangleq 0$ and

$$\tilde{h}_{j, \lfloor (j+d)/2 \rfloor}(p_0, x) \triangleq \begin{cases} f_{j, \lfloor (j+d)/2 \rfloor}(p_0, x), & 0 \leq d \leq j - 1, \\ p_0, & d > j - 1. \end{cases} \quad (18)$$

The EEP function for an irregular code under the proposed decoder is then given by

$$h(p_0, x) = \sum_{j=2}^{D_L} \lambda_j \tilde{h}_{j, \lfloor (j+d)/2 \rfloor}(p_0, x) \quad (19)$$

if $v_{d-1}(p_0) < x \leq v_d(p_0)$, $d = 1, 2, \dots$. The value of $v_d(p_0)$ can be obtained by solving (16).

The EEP chart for an irregular code is shown in Fig. 3 under the proposed decoding method. The code ensemble is given by

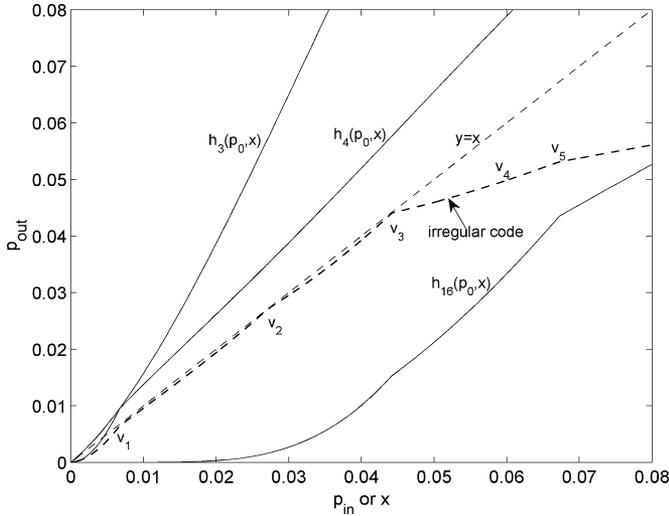


Fig. 3. EEP charts of an irregular code under the proposed decoder.

$\{\lambda_3 = 0.1234, \lambda_4 = 0.5551, \lambda_{16} = 0.3215, \text{ and } \rho_{10} = 1\}$ with the decoding threshold $p_0^* = 0.0577$. The channel input is set at $p_0 = p_0^*$. The EEP functions $h_j(p_0, x)$ from (IV-C.1) for $j = 3, 4, \text{ and } 16$ are illustrated in the chart. The weighted sum EEP function, i.e., the EEP function for irregular codes $h(p_0, x)$ from (19), is also illustrated. For $x \leq p_0$, the switching points of the proposed decoder are $v_1, v_2, \text{ and } v_3$. It is seen that the EEP function $h(p_0, x)$ is under the line $y = x$, demonstrating that the value p_0 is less than the decoding threshold of this code ensemble.

2) *Flipping Thresholds for Finite-Length Codes:* By tracking the EEP function $h(p_0, x)$ under the iterative decoding, we can obtain $d^{(\ell)}$ analytically. However, for finite-length codes, the actual EEP during the decoding usually deviates from the analytical result. Therefore, the analytically predetermined $d^{(\ell)}$ may not be the optimal flipping threshold for the ℓ th decoding iteration.

One solution is to obtain $d^{(\ell)}$ based on the estimation of EEP during decoding. We first count the number of unsatisfied check nodes, denoted as $M_e^{(\ell)}$. The EEP can be estimated by solving

$$\frac{1 - \sum_i \tilde{\rho}_i (1 - 2\hat{p}^{(\ell)})^i}{2} = \frac{M_e^{(\ell)}}{M}. \quad (20)$$

With the estimated EEP $\hat{p}^{(\ell)}$, we can compute $d^{(\ell)}$. However, this method incurs additional implementation complexity at the decoder.

We now present some simple but effective methods that does not require the EEP estimation. It has been shown that the expanded optimal switch scheme (exp-OSS) is effective for hard decoding of finite-length regular codes [16]. We now extend it to irregular codes as follows.

Let $d^{*(\ell)}$ be the optimal flipping thresholds obtained analytically. For the K -exp-OSS, the flipping thresholds in the decoder are given by $d^{(\ell)} = d^{*(\lfloor \ell/K \rfloor)}$. That is, in the actual decoding, we simply use the same analytical flipping threshold $d^{*(\ell)}$ for K iterations before using the next flipping threshold $d^{*(\ell+1)}$. For example, if the optimal thresholds $(d^{*(1)}, d^{*(2)}, d^{*(3)}, \dots) =$

$(5, 4, 3, \dots)$, then, the flipping thresholds for the 3-exp-OSS are given by $(d^{(1)}, d^{(2)}, d^{(3)}, \dots) = (5, 5, 5, 4, 4, 4, 3, 3, 3, \dots)$, where each optimal threshold is repeated three times. As discussed in [16], we usually choose $K = 3$ or 4.

Another improved decoder is to simply add several more iterations on one value of the threshold before moving to the next value, called the L -add-OSS. Suppose the first l_1 optimal flipping thresholds are d_1^* , the following l_2 are d_2^* , and the next l_3 are d_3^*, \dots . Then for the L -add-OSS, the first $l_1 + L$ optimal flipping thresholds are d_1^* , the following $l_2 + L$ are d_2^* , and the next $l_3 + L$ are d_3^*, \dots . For example, if the optimal thresholds $(d^{*(1)}, d^{*(2)}, d^{*(3)}, d^{*(4)}, \dots) = (5, 4, 4, 3, \dots)$, then the flipping thresholds for the 2-add-OSS are given by $(d^{(1)}, d^{(2)}, d^{(3)}, \dots) = (5, 5, 5, 4, 4, 4, 4, 3, 3, 3, \dots)$. We can choose $L = 5$.

The rationale of the proposed K -exp-OSS and L -add-OSS is as follows. The actual EEP may deviate from the analytical result. If the actual EEP is smaller than the analytical one, the flipping thresholds based on the analytical result may be larger than needed, which slows down the convergence speed of actual EEP but does not cause decoding failure. If the actual EEP is larger than the analytical one, the flipping thresholds based on the analytical result may be smaller than needed, which will cause the decoding failure. To avoid the decoding failure in the latter case, we propose the K -exp-OSS and L -add-OSS, where the decoder runs more iterations on larger flipping thresholds to ensure that the actual EEP is small enough to allow for smaller thresholds. This makes the decoding more stable, and thus, provides performance improvement for finite-length codes.

D. Analysis of Decoding Complexity

The proposed decoder can decrease the circuit complexity in implementation because the degree information of variable nodes does not have to be retrieved. Let N_{para} be the number of parallel decoding units. For the proposed decoder, since the optimal flipping threshold is the same for all variable nodes, the central unit only needs to distribute this threshold to all the N_{para} decoding units for variable nodes. In the conventional decoder, the central unit needs to retrieve the degree information for each of the N_{para} variable decoding units, get the corresponding N_{para} flipping thresholds, and send the flipping thresholds to the N_{para} variable decoding units. The reduction of complexity in terms of the number of operations is evident.

V. CODE OPTIMIZATION FOR MULTILEVEL CODING

A. LDPC Code Optimization Under Proposed Decoder

The error-free decoding constraint simply states that the output extrinsic error probability should be smaller than the input, i.e., $h(p_0, x) < x$ [21]. The idea of code optimization here is similar to that given in [22], where we seek the maximal channel error probability or the maximal code rate that makes the EEP function lie under the line $y = x$. Given a code rate R , the problem of irregular code ensemble optimization under the

proposed decoder can be formulated as

$$\begin{aligned} \max_{\{\lambda_j, \rho_j\}} \quad & p_0, \\ \text{s.t.} \quad & h(p_0, x) < x, \quad \forall x \leq p_0; \\ & R = 1 - \frac{\sum \rho_j/j}{\sum \lambda_j/j}, \end{aligned} \quad (21)$$

where the EEP function $h(p_0, x)$ is given in (19). As discussed earlier, given a code ensemble, the proposed efficient decoder with optimal switching points offers the same performance as the GB decoder. Therefore, the optimized decoding thresholds for the proposed decoder are exactly the same as that of the GB decoder. As seen from (19), the EEP function of an irregular code under the proposed decoder is continuous but its first-order derivative is not continuous. The EEP function consists of several segments. The number of segments is one plus the number of $v_d(p_0)$'s ($v_d(p_0) < p_0$). We have the following observation.

Remark: It is empirically observed that for the EEP functions given the set of $v_d(p_0) < p_0, d = 1, \dots, L$, if $h(p_0, v_d(p_0)) < v_d(p_0)$, we have $h(p_0, x) < x$ for all $x \in (0, v_L]$.

Based on the above observation, we can simplify the code optimization under the proposed decoder as follows:

$$\begin{aligned} \max_{\{\lambda_j, \rho_j\}} \quad & p_0, \\ \text{s.t.} \quad & h(p_0, v_d(p_0)) < v_d(p_0), \\ & \forall v_d(p_0) < p_0. \end{aligned} \quad (22)$$

In (22) we need to consider only the points $\{v_d(p_0)\}$ for the error-free decoding constraint, which significantly simplifies the optimization procedure. We can solve above optimizations in (21) and (22), using the differential evolution (DE) technique [21], [23]. We have observed that the code optimizations (21) and (22) yield identical results.

On the other hand, instead of maximizing the decoding threshold p_0 , we can design the code profiles to maximize the code rate for a given p_0 . The code optimization then becomes

$$\begin{aligned} \max_{\{\lambda_j, \rho_j\}} \quad & R = 1 - \frac{\sum \rho_j/j}{\sum \lambda_j/j}, \\ \text{s.t.} \quad & h(p_0, x) < x, \quad \forall x \leq p_0. \end{aligned} \quad (23)$$

Moreover, we can fix the check node profile and design the variable node profile. With the simplification given in (22), the code optimization in (23) becomes

$$\begin{aligned} \max_{\{\lambda_j\}} \quad & \sum \lambda_j/j, \\ \text{s.t.} \quad & \sum_j \lambda_j f_{j, \lfloor (j+d)/2 \rfloor}(p_0, v_d(p_0)) < v_d(p_0), \\ & \forall v_d(p_0) < p_0, \end{aligned} \quad (24)$$

which can be solved by linear programming (LP). We further consider a concentrated check node profile, i.e., only one degree component d_c . Therefore, for a given p_0 , we can fix the check node degree d_c , and solve the optimization in (24) by LP. The resulting code rate is then given by $R = 1 - 1/(d_c \sum_j \lambda_j/j)$.

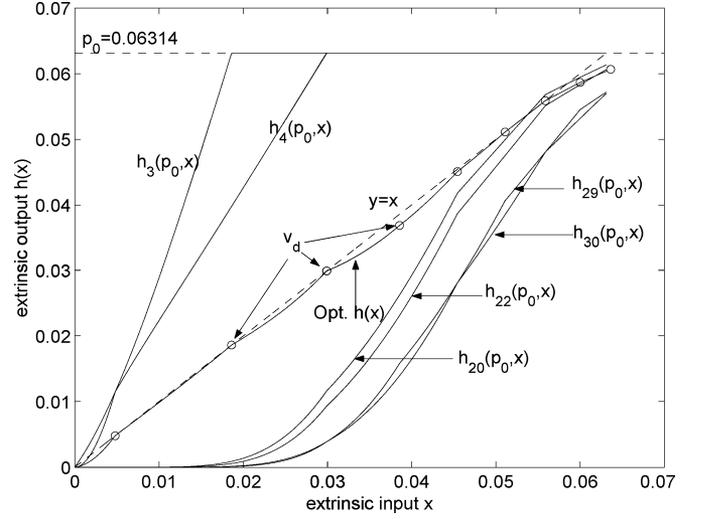


Fig. 4. EEP charts of the optimized irregular LDPC codes under the proposed decoder.

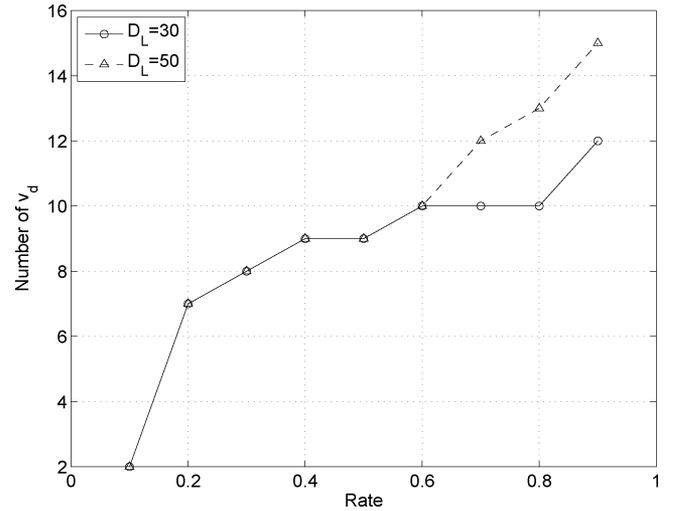


Fig. 5. Number of v_d 's for code optimization in the proposed decoder.

By changing d_c , we obtain a set of optimization results. Among them, the maximum code rate and the corresponding code profile $\{\{\lambda_j\}, d_c\}$ give the optimized results for the given p_0 . To design the code for a target rate R^* , we start with a small p_0 and solve the optimization in (24) by LP to find the optimized profile $\{\{\lambda_j\}, d_c\}$, and gradually increase p_0 until the optimized code rate reaches the desired R^* . The final p_0 is then the optimized threshold for the code rate R^* .

We illustrate the EEP chart of some optimized irregular codes for the proposed decoder in Fig. 4. The code rate is $R = 0.5$. The optimization considers only the constraints for several $v_d(p_0)$'s as in (22). We can see that here only nine switching points are considered for code optimization. The resulting EEP curve for the optimized code is below $y = x$ for all $x < p_0$.

Fig. 5 shows the number of switching points $\{v_d(p_0)\}$ used in code optimization for code rates $R = 0.1, 0.2, \dots, 0.9$ with $D_L = 30$ and $D_L = 50$, indicating the number of component EEP functions in (IV-C.1) (used in the optimization. For the

simplified optimization, it represents the number of switching points considered in the optimization. It is seen that for all code rates, only a small number of switching points (less than 15) are considered for the code ensemble optimization, e.g., only two points for $R = 0.1$. We find that the simplified formulation (22) provides the same optimized threshold results as the original formulation in (21). This demonstrates that the irregular codes for the proposed decoder can be designed in a very efficient way. The optimized threshold results are shown in Fig. 2 for the code rates $R = 0.05$ to 0.9, where the maximum left degree $D_L = 30$.

B. Component Code Design for PID/MSD

The component code design includes two aspects. One is to optimize the profiles of component codes given the channel E_s/N_0 , and the other is to allocate rates and optimize profiles for component codes given the overall code rate.

Given the channel E_s/N_0 for $1 \leq i \leq m$, we evaluate the error rates e_i for $S^{(i)}$, and perform the rate optimization for $C^{(i)}$ by the code rate optimization (23) or (24). Let $R_{\text{PID}}(E_s/N_0)$ be the average rate of the component codes according to (1).

Given the overall rate $R_{\text{PID}} = r$, we design the component codes $C^{(i)}$ for $1 \leq i \leq m$, using a bisection search. The search starts from a low value $(E_s/N_0)_l$ and a high value $(E_s/N_0)_h$. We let $(E_s/N_0)_m = ((E_s/N_0)_h + (E_s/N_0)_l)/2$, and compute the $R_{\text{PID}}((E_s/N_0)_m)$. If $R_{\text{PID}}((E_s/N_0)_m) < r$, then we update $(E_s/N_0)_l = (E_s/N_0)_m$; otherwise, we update $(E_s/N_0)_h = (E_s/N_0)_m$, until $R_{\text{PID}}((E_s/N_0)_m)$ is sufficiently close to r . Assuming $(E_s/N_0)_m = (E_s/N_0)^*$ when the bisection search terminates, we output $(E_s/N_0)^*$ as the threshold and the optimized profiles of component codes for $(E_s/N_0)^*$ as the designed code profiles.

The component code design for MSD is similar to that for PID. Given E_s/N_0 , we optimize the component codes based on the code rate optimization given in (23) or (24). Given the average code rate, we allocate the rates and optimize the profiles for component codes by a bisection search. The difference is that the bit error rates e_i for $S^{(i)}$ ($i = 1, 2, \dots, m$) are evaluated as follows. Suppose we transmit the signal s and receive the noisy signal y . The evaluation of the bit error rate $e^{(1)}$ for $S^{(1)}$ is the same as that for PID. To evaluate the error rates $e^{(i)}$ for $S^{(i)}$ for $i \geq 2$, we find the signal point closest to y in the constellation denoted as $\hat{s}^{(i)}$, subject to the constraint that the mapping bits for $\hat{s}^{(i)}$ in the subsets $S^{(j)}$ for $1 \leq j \leq i-1$ are equal to those for s .

VI. NUMERICAL AND SIMULATION RESULTS

A. Threshold Evaluations

We compare the thresholds in terms of E_s/N_0 for PID and MSD with those for single-level coding at different code rates.

Fig. 6 shows the threshold improvement of PID over single-level coding for 16-QAM, 64-QAM, and 256-QAM. The improvement is less significant for lower-order modulations, and more significant for higher-order modulations. As the code rate increases, the improvement becomes less significant. This can be explained as follows. For lower-order modulation schemes

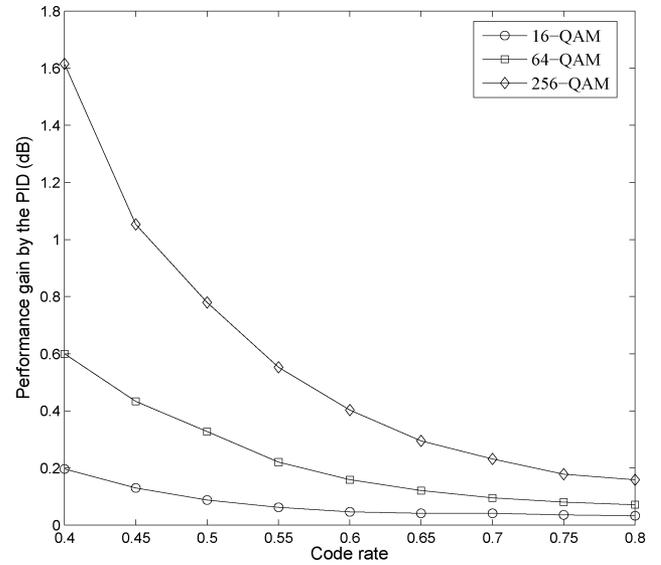


Fig. 6. Threshold improvement of the PID over the single-level coding.

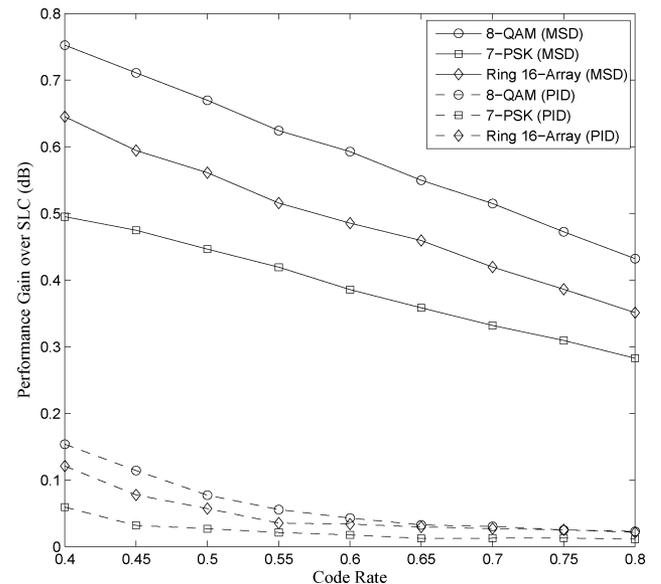


Fig. 7. Threshold improvement of the MSD over the single-level coding.

or higher SNR (where we use high rate codes), the difference between the error rates for different subsets $S^{(k)}$ is smaller, so the rate gap in (6) is less significant. As noted earlier, for these Gray-mapped modulations, the MSD has almost the same performance as the PID.

Fig. 7 shows the threshold improvement of the MSD over single-level coding for 8-QAM, 7-PSK, and Ring 16-Array, as well as the threshold improvement of the PID. It is seen that the MSD significantly outperforms the PID for these non-Gray-mapped modulations.

On the other hand, the MSD incurs a larger decoding delay due to the serial decoding. For the PID, all the component codes are simultaneously decoded, therefore, the delay is the maximum decoding time among all component codes. For the MSD, it is the sum of the decoding time for all component codes.

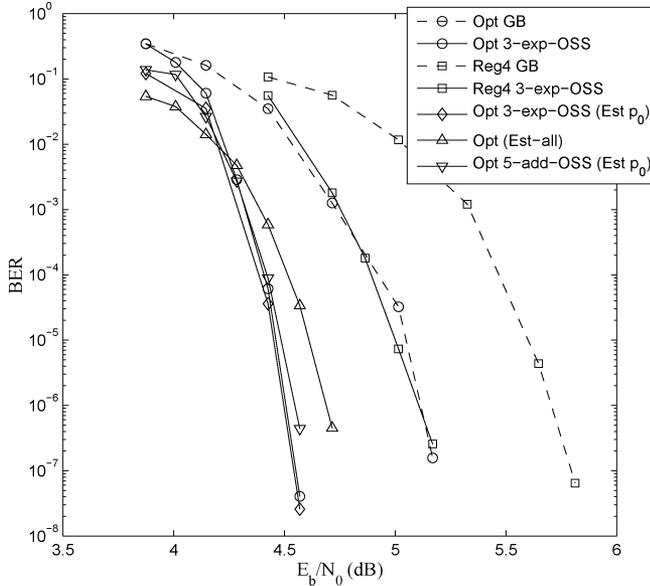


Fig. 8. BER performance of the rate-0.5 optimized codes of length 15 000 with QPSK modulation decoded by the proposed decoder and its variants 3-exp-OSS and 5-add-OSS.

B. Simulation Results

The performance is plotted against the normalized electrical SNR E_b/N_0 , where E_b denotes the energy for each bit, and N_0 denotes the power spectrum density of the AWGN. To reduce the error floor, all the optimized LDPC codes are concatenated with an outer shortened (1000, 970) BCH code. In all simulations, the concatenated codes show no error floor above 10^{-8} . We compare the performance of the optimized multilevel coding systems with PID and MSD with the optimized single-level coding systems.

Fig. 8 shows the performance of rate-0.5 optimized LDPC codes with QPSK modulation. We employ single-level coding, since the Gray mapping is employed and both mapping bits suffer the same error rate. The performances of the known crossover error probability (without Est) and of the estimated crossover probability, using (20) (with Est), are shown. For the decoding with error probability estimation given by (20), we consider two cases. In the first case, we estimate only the crossover channel error probability p_0 in the first iteration, compute the flipping thresholds by carrying out EEP analysis, using the estimated p_0 , and then, perform the 3-exp-OSS (denoted as Est p_0). In the second case, we estimate the error probabilities $p^{(\ell)}$ in all iterations, and then, determine the flipping thresholds (denoted as Est-all). Since the overall code rate is $0.97 \times 0.5 = 0.485$, we show the performance of the optimized codes as well as that of the rate-0.485 code with all degree-4 variable nodes (denoted as Opt and Reg4, respectively). The simulated codes are of length 15 000, and decoded by the proposed decoder as well as its finite-length variant (denoted as GB and 3-exp-OSS, respectively). With the known error probability, the 3-exp-OSS significantly outperforms the original decoder for both the optimized codes and the “Reg4” codes. The optimized codes exhibit about 0.6 dB improvement over the “Reg4” codes. The decod-

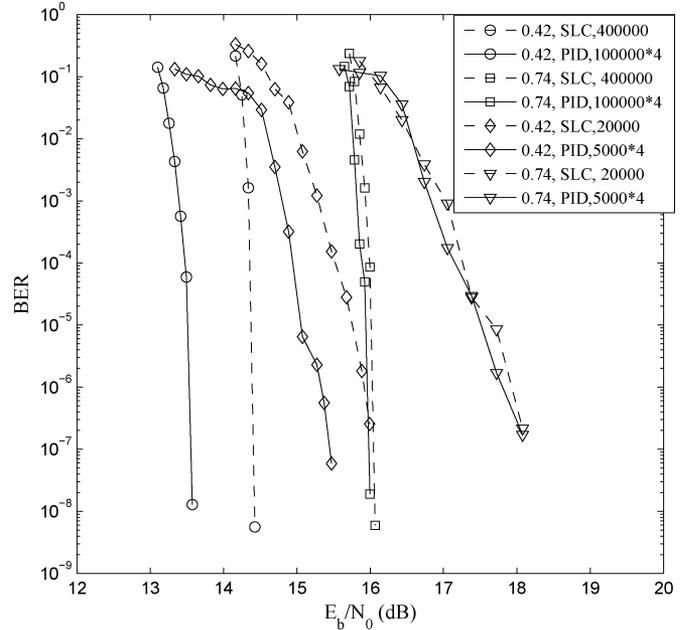


Fig. 9. BER performance of the optimized multilevel coding system with 256-QAM and PID, and the optimized single-level coding system, with average code rates 0.42 and 0.74.

ing performance based on estimating only p_0 is better than that based on estimating the error probability in all iterations given by (20). In the case of estimating only p_0 and computing the flipping thresholds, the 5-add-OSS (denoted as 5-add-OSS) shows little performance degradations compared with the 3-exp-OSS. However, the average number of iterations needed by the 5-add-OSS for successful decoding is only about 70 percent of that for the 3-exp-OSS, which shows that the 5-add-OSS is more efficient.

Fig. 9 shows the performance of the PID for 256-QAM with the average code rates 0.42 and 0.74. All codes are decoded by 3-exp-OSS. For long codes, four component codes of length 100 000 and average rate 0.42 (0.74) are employed for b_1b_5, b_2b_6, b_3b_7 , and b_4b_8 for PID, and one code of length 400 000 and rate 0.42 (0.74) is employed for single-level coding. For moderate-length codes, four component codes of length 5000 and average rate 0.42 (0.74) are employed for b_1b_5, b_2b_6, b_3b_7 , and b_4b_8 for PID, and one length-20 000 code consisting of four interleaved length-5000 and rate-0.42 (0.74) codes is employed for single-level coding. We use the four interleaved length-5000 codes for single-level coding because the performance improvement in terms of the threshold can be offset by quadrupling the code length if we employ a length-20 000 code for single-level coding. The PID shows significant performance improvement over the single-level coding at the rate 0.42 for both long and moderate-length codes, but little improvement at the rate 0.74. Since in Fig. 6, the predicted threshold improvement for 256-QAM at the rate 0.74 is less than 0.2 dB, the little performance improvement in simulations is expected.

Fig. 10 shows the performance of MSD for the 8-QAM for the average code rate 0.7. The mapping bits in each level are given in Table I. All codes are decoded by 3-exp-OSS. For long codes,

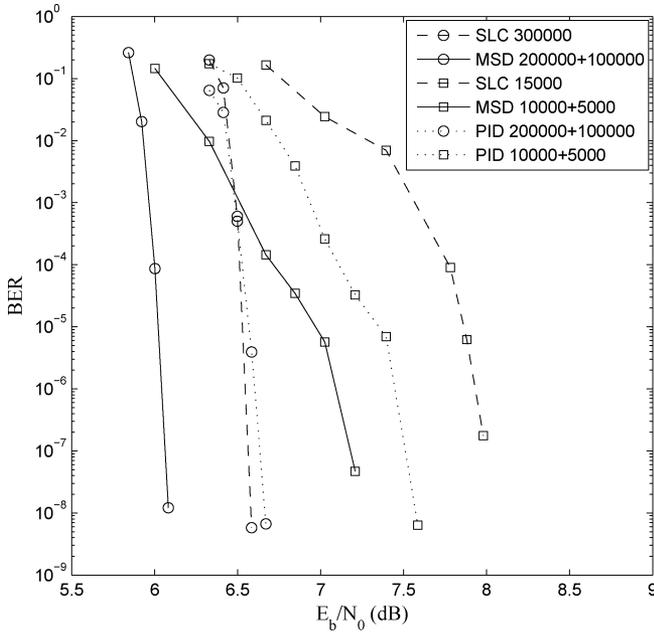


Fig. 10. BER performance of the optimized multilevel coding system with 8-QAM and MSD, and that of the optimized single-level coding system, with the average code rate 0.7.

we employ one length-200 000 code for b_1b_2 and one length-100 000 code for b_3 for MSD and PID, and one length-300 000 code for single-level coding. For moderate-length codes, we employ one length-10 000 code consisting of two interleaved length-5000 codes for b_1b_2 , and one length-5000 code for b_3 for MSD and PID, and one length-15000 code consisting of three interleaved length-5000 codes for single-level coding. The reason of using interleaved codes here is the same as before. Compared with single-level coding, the MSD shows significant performance improvement for both long and moderate-length codes. The performance of PID is also shown. For long codes, the PID and single-level coding have almost the same performance, which is not surprising, since from Fig. 7 the PID shows little threshold improvement over the single-level coding. For moderate-length codes, the performance improvement of the PID is more significant. This is because for moderate-length codes the performance prediction from the threshold is less accurate. Simulation results confirm that for 8-QAM, the MSD provides more significant performance improvement than the PID, which is predicted in Fig. 7.

C. Discussions

We next analyze the gain of the proposed decoding method in optical communications in the context of the optical SNR (OSNR) and the total transmission distance.

First, the OSNR is defined as

$$\text{OSNR} = \frac{E_b}{N_0} \frac{R_b}{B_{\text{ref}}}, \quad (25)$$

where R_b is the information bit rate and B_{ref} is the reference optical bandwidth, which is often chosen to be 0.1 nm (the

size of wavelength range). Hence, an improvement in E_b/N_0 translates into the same improvement on OSNR.

Now use the transmission model specified in the optical transmission standard International Telecommunication Union Standard G.692 [24] to give a clear explanation on the performance gains introduced by the proposed schemes. Suppose there is a chain of optical amplifiers and between two amplifiers there is an optical fiber with some path loss. From the formula (I.3) in [24], the OSNR can be approximated by

$$\text{OSNR} = P_{\text{out}} - L - NF - 10 \log_{10} N - 10 \log_{10}(h\nu\Delta\nu_0) \quad (26)$$

where P_{out} is the output power of the amplifier in dBm, L is the span loss between amplifiers in dB, NF is the external noise figure in dB, $\Delta\nu_0$ is the optical bandwidth, N is the number of spans in the optical transmission chain, and equal span loss is assumed. For the same P_{out} , L , NF , and $h\nu\Delta\nu_0$, the coding gain in terms of E_b/N_0 can increase the number of spans N allowed in optical transmissions assuming the same receiving OSNR. For example, 0.7 dB performance gain of the length 15000 MSD over single-layer coding in Fig. 10 will result in an increase of $10^{0.07} - 1 = 17.5\%$ in terms of the number of spans in optical communications, i.e., the 17.5% increase of the transmission distance. The 0.6 dB performance gain of the length 15000 3-exp-OSS over GB for optimized code in Fig. 8 will result in an increase of $10^{0.06} - 1 = 14.8\%$ of the transmission distance.

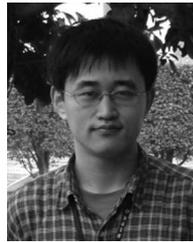
VII. CONCLUSION

We have considered the multilevel LDPC-coded systems with high-order modulations and hard decoding for high-speed optical communications. We have shown that for Gray-mapped modulations, the PID is the suitable receiver decoding strategy; whereas for non-Gray-mapped modulations, the MSD is more powerful. We have developed a new switch-type LDPC hard decoding method that does not require the degree information of the variable nodes, and therefore, is more efficient for circuit-level implementations, as well as its variant for finite-length codes. We have optimized both PID and MSD systems by allocating rates and designing profiles for component codes. In both threshold evaluations and finite-length code simulations, the optimized PID and MSD schemes show significant performance improvement over the single-level coding, which also employs the optimized codes. The low-complexity feature makes the proposed schemes promising for practical use in high-speed optical transmissions.

REFERENCES

- [1] E. Vergnol, J. Cadiou, A. Carencio, and C. Kazmierski, "New modulation scheme for integrated single side band lightwave source allowing fiber transport up to 256 QAM over 38 GHz carrier," in *Proc. OFC 2000*, pp. 134–136.
- [2] E. Funk, V. Urick, S. Strutz, J. Dexter, and K. Williams, "110 km 256-QAM digital microwave over fiber link," in *Proc. MTT-S 2008*, vol. 1, pp. 269–272.
- [3] M. Yoshida, H. Goto, T. Omiya, K. Kasai, and M. Nakazawa, "Frequency division multiplexed 1 Gsymbol/s, 64 QAM coherent optical transmission with a spectral efficiency of 8.6 bit/s/Hz," in *Proc. ECOC 2008*, pp. 1–4.

- [4] H. Jumpei, K. Kasai, M. Yoshida, and M. Nakazawa, "1 Gsymbol/s, 64 QAM coherent optical transmission over 150 km with a spectral efficiency of 3 bit/s/Hz," in *Proc. OFC/NFOEC 2007*, 2010, pp. 1–3.
- [5] S. Bigo, "Coherent detection: A key enabler for next-generation optical transmission systems," in *Proc. 2007 Transparent Opt. Netw., ICTON 2007. 9th Int. Conf.*, Jul., pp. 332–335.
- [6] J. Renaudier, G. Charlet, M. Salsi, O. Pardo, H. Mardoyan, P. Tran, and S. Bigo, "Linear fiber impairments mitigation of 40-Gbit/s polarization-multiplexed QPSK by digital processing in a coherent receiver," *J. Lightwave Technol.*, vol. 26, no. 1, pp. 36–42, Jan. 2008.
- [7] C. Zhang, Y. Mori, K. Igarashi, K. Katoh, and K. Kikuchi, "Ultrafast operation of digital coherent receivers using their time-division demultiplexing function," *J. Lightwave Technol.*, vol. 27, no. 3, pp. 224–232, Feb. 1, 2009.
- [8] O. A. Sab and V. Lemaire, "Block turbo code performance for long-haul DWDM optical transmission systems," in *Proc. 2000 Opt. Fiber Commun. Conf.*, Mar., pp. ThS5-1–TuS5-4.
- [9] M. Akita, H. Fujita, T. Mizuochi, K. Kubo, H. Yoshida, K. Kuno, and S. Kurahashi, "Third generation FEC employing turbo product code for long-haul DWDM transmission systems," in *Proc. 2002 Opt. Fiber Commun. Conf.*, Mar., pp. 289–290.
- [10] C.-J. Ahn, S. Takahashi, H. Fujisaka, T. Kamio, and K. Haeiwa, "Power consumption for coherent optical orthogonal frequency division multiplexing with punctured LDPC codes and variable amplitude block codes," *J. Lightwave Technol.*, vol. 26, no. 14, pp. 2227–2234, Jul. 2008.
- [11] I. Djordjevic and B. Vasic, "Multilevel coding in M-ary DPSK/differential QAM high-speed optical transmission with direct detection," *J. Lightwave Technol.*, vol. 24, no. 1, pp. 420–428, Jan. 2006.
- [12] B. Vasic, I. Djordjevic, and R. Kostuk, "Low-density parity check codes and iterative decoding for long-haul optical communication systems," *J. Lightwave Technol.*, vol. 21, no. 2, pp. 438–446, Feb. 2003.
- [13] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.
- [14] P. Zarrinkhat and A. H. Banihashemi, "Hybrid hard-decision iterative decoding of irregular low-density parity-check codes," *IEEE Trans. Commun.*, vol. 55, no. 2, pp. 292–302, Feb. 2007.
- [15] N. Miladinovic and M. Fossorier, "Improved bit-flipping decoding of low-density parity-check codes," *IEEE Trans. Info. Theory*, vol. 51, no. 4, pp. 1594–1606, Apr. 2005.
- [16] C. Gong, Q. Liu, H. Cui, and K. Tang, "Switch-type hybrid hard-decision decoding algorithms for regular low-density parity-check codes," *IEEE Trans. Info. Theory*, vol. 54, no. 7, pp. 3181–3188, Jul. 2008.
- [17] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel codes: Theoretical concepts and practical design rules," *IEEE Trans. Info. Theory*, vol. 45, no. 5, pp. 1361–1391, Jul. 1999.
- [18] G. D. Forney and G. Ungerboeck, "Modulation and coding for linear Gaussian channels," *IEEE Trans. Info. Theory*, vol. 44, no. 6, pp. 2384–2415, Oct. 1998.
- [19] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Info. Theory*, vol. 47, no. 2, pp. 585–598, Feb. 2001.
- [20] M. Ardakani and F. R. Kschischang, "A more accurate one-dimensional analysis and design of irregular LDPC codes," *IEEE Trans. Commun.*, vol. 52, no. 12, pp. 2106–2114, Dec. 2004.
- [21] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Info. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.
- [22] M. Ardakani and F. R. Kschischang, "Properties of optimum binary message-passing decoders," *IEEE Trans. Info. Theory*, vol. 51, no. 10, pp. 3658–3665, Oct. 2005.
- [23] K. Price and R. Storn, "Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Opt.*, vol. 11, pp. 341–359, 1997.
- [24] *International Telecommunication Union, Optical Interfaces for Multichannel Systems with Optical Amplifiers ITU-T Recommendation*, ITU-T G.692 Standard, 1998.



Chen Gong received the B.S. degree in electrical engineering and mathematics (minor) from Shanghai Jiaotong University, Shanghai, China, in 2005 and the M.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 2008. He is currently working toward the Ph.D. degree with the Department of Electrical Engineering, Columbia University, New York.

His current research interests include the area of channel coding and modulation techniques for wireless communications.



Xiaodong Wang (S'98–M'98–SM'04–F'08) received the Ph.D. degree in electrical engineering from Princeton University, NJ.

He is currently a Professor with the Department of Electrical Engineering, Columbia University, New York. He is author or coauthor of a recent book entitled *Wireless Communication Systems: Advanced Techniques for Signal Reception* (Prentice Hall, NJ: 2003). His current research interests include wireless communications, statistical signal processing, and genomic signal processing.

Dr. Wang received the 1999 National Science Foundation CAREER Award, and the 2001 IEEE Communications Society and Information Theory Society Joint Paper Award. He was an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and the IEEE TRANSACTIONS ON INFORMATION THEORY.