LDPC Design for Block Differential Modulation in Optical Communications

Sandro Bellini, Marco Ferrari, *Member, IEEE*, Alessandro Tomasoni, *Member, IEEE*, Carlo Costantini, Luca Razzetti, and Giancarlo Gavioli

I. Introduction

THE increasing demand of capacity in optical transport networks (OTN) has pushed transmission systems to 100 Gbps per wavelength in dense WDM and beyond. In these conditions, the receiver is required to perform ultra-narrow filtering as well as polarization mode and chromatic dispersion compensation. Besides, inter-channel non linearities limit the power at the transmitter. Today, high speed analog to digital converters (ADC) enable coherent detection of the modulated optical signal. Coherent detection acts like an ultra-narrow WDM filter and enables DSP for electronic dispersion compensation. Current 100 G transceivers employ polarization multiplexed QPSK and QAM formats and intradyne digital coherent receivers.

In coherent optical receivers, third generation forward error correction (FEC) codes with net coding gain (NCG) higher than 10 dB are based on soft-decision (SD) and iterative decoding. This has been the key enabling factor to improve performance and extend transmission distances even further. In the last two decades, a lot of activity in FEC code design has been triggered by soft-iterative decoding of concatenated codes, such as turbo codes [1] or low-density parity-check (LDPC) codes [2]. Many authors have worked extensively on the application of soft-FEC codes to OTN. This application field specifically requires high reliability and parallelization, due to a very high bit rate processing. These two needs have driven the choice on turbo product codes (TPC) [3] or LDPC codes. At the beginning, the need for high code rates and very low error rates made researchers focus on TPCs [4], which still show some advantages in terms of iterations and low power consumption [5]. LDPC codes have also been investigated in binary [6], non-binary [7], and convolutional form [8], up to the more recent spatially coupled LDPC codes [9].

In LDPC codes for OTN, special attention must be paid to the *error floor*. To overcome this problem, the simplest solution is to rely on an outer (generally hard decoded) FEC code concatenated with the inner LDPC code [10], [11], [12], [9]. Recently, *floorless* LDPC codes have been designed [13], [14]. These codes are able to reach the desired error rate without an outer concatenated code, due to their extremely low error floor.

All of the above SD-FEC solutions assume ideal coherent detection, that is often not the case in practice. In an intradyne transmission system, the incoming optical signal is mixed by means of a polarization and phase diversity optical front end with the signal emitted by a local receiver laser. The local receiver laser frequency is not locked to the incoming signal. Instead, the two generally differ by a much smaller value than the modulation symbol-rate. Estimation and compensation of the residual phase and frequency error is performed after the ADC by DSP algorithms for carrier recovery and demodulation. Due mostly to implementation constraints, DSP operations currently implemented in coherent receivers are based on non-data aided, or blind, equalization and synchronization algorithms. Blind carrier recovery algorithms have the key advantage of using the contribution of all transmitted symbols for channel estimation, and thus exhibit fast carrier phase and frequency tracking, as required by optical transmission. However, they generally suffer from phase ambiguity due to phase slips, which prevents absolute phase estimation of the received carrier. As a consequence of a phase slip, the demodulated signal is phase rotated compared to the receiver phase reference from the instant the slip event has occurred.

This problem is often overcome by using differential modulation (DM), which encodes information on phase changes rather than on the absolute phase values. However, differential

Manuscript received July 15, 2014; revised October 3, 2014; accepted November 7, 2014. Date of publication November 19, 2014; date of current version December 16, 2014.

S. Bellini is with the Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria, 20133 Milano, Italy (e-mail: sandro.bellini@polimi.it).

M. Ferrari and A. Tomasoni are with the Consiglio Nazionale delle Ricerche, Istituto di Elettronica e di Ingegneria dell'Informazione e delle Telecomunicazioni, 20133 Milano, Italy (e-mail: marco.ferrari@ieiit.cnr.it; alessandro.tomasoni@ieiit.cnr.it).

C. Costantini, L. Razzetti, and G. Gavioli are with the Alcatel-Lucent, Vimercate 20059, Italy (e-mail: carlo.costantini@alcatel-lucent.com; luca_gabriele.razzetti@alcatel-lucent.com; giancarlo.gavioli@alcatel-lucent.com).

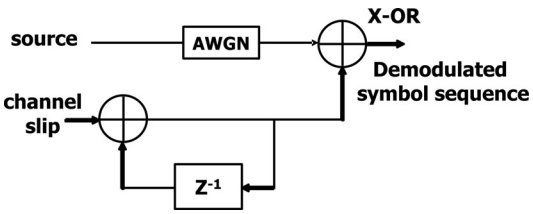

Fig. 1. Equivalent model for the channel affected by phase slips (BPSK).

decoding leads to a significant decline in receiver sensitivity, that exceeds 1 dB at the SD-FEC target BER for QPSK. An alternative approach, widely used in radio communications, is to avoid differential encoding by periodic insertion of time-division multiplexed pilot symbols. Pilot symbols provide the carrier phase recovery circuit with the transmitter phase reference, which can be used for absolute phase error estimation. Since pilot symbols do not carry information, their rate should be kept as low as possible, to avoid reducing the information data rate or introducing SNR penalties. In fiber communications, however, the carrier phase recovery circuit requires phase estimation on each symbol to efficiently track the carrier phase noise, which is a fast random process introduced by the laser source and fiber nonlinearities on the optical carrier [15]. Operating with a low pilot rate leads to a reduction of the demodulator phase noise tracking abilities, and thus, to frequent phase errors, even in presence of moderate laser phase noise.

Phase estimation can be performed by any blind search algorithm, such as the Viterbi & Viterbi synchronizer or the maximum likelihood phase search (MLPS) algorithm followed by phase unwrapping. The probability that a blind phase estimator commits phase slips depends on the statistical properties of the phase noise affecting the received signal. After demodulation with the estimated phase, the phase noise will be compensated and the signal will be left with additive white Gaussian noise (AWGN) from the channel and phase slips (PS) introduced by the process of phase estimation. Fig. 1 shows the effect of this PS channel on the transmitted bits, in case of BPSK. A similar channel model, with independent slips and (polarization dependent) AWGN is considered in [16].

A careful capacity analysis of the PS channel reveals that the loss from coherent demodulation can be considerably reduced by block differential modulation (BDM) [17]. In this modulation scheme, described in Section II, each codeword is divided into blocks of n adjacent symbols. Within each block, any rotated version of an admissible transmitted sequence corresponds to the same bit sequence so that the absolute phase does not matter. In this way, $n-1$ information symbols can be transmitted. In addition, another symbol is differentially encoded through the phase difference between the blocks. BDM is robust against PS, even when the PS occurrence rate is unknown, without paying the cost in SNR required by DM.

This paper assumes a concatenated FEC scheme with an outer block code, and proposes an inner FEC design based on LDPC codes that embeds block differential demodulation, making the code robust to occasional phase slips. Latency and complexity are almost unchanged compared to a standard LDPC decoder. It starts with the design of a binary LDPC code for BPSK in Section III. Then, the principle is extended to QPSK in Section IV that proposes an LDPC code design in $\mathbb{Z}_4$ (QLDPC), suited to BDM. Using this code on QPSK, the transmission rate is doubled with the same code and symbol rate adopted for BPSK. Finally, in Section V, to further increase the spectral efficiency, mapping methods that allow modulation, demodulation and decoding of QLDPC on 16-QAM symbols are designed. This coding and modulation scheme, thus, covers all rates from 40 to 200 Gbps in WDM optical transmission, enduring occasional cycle slips of the carrier phase.

II. System Model

At time i, a BPSK, QPSK, or QAM symbol X_i is transmitted, and in absence of phase slips the received symbol is

$$Y_i = X_i + N_i \tag{1}$$

where $\{N_i\}$ is an *iid* sequence of Gaussian random variables with variance $N_0/2$ per axis. In presence of phase slips, the received symbol is

$$Y_i = S_i X_i + N_i \tag{2}$$

where $S_i \in \{+1, -1\}$ for BPSK and $S_i \in \{\exp(jk\pi/2)\}$ $(k = 0, 1, 2, 3)$ for QPSK and QAM. The channel phase state S_i changes at random time instants.

This paper assumes that the recovered carrier is perfect, but for phase slips. The detailed evolution of the instantaneous phase reference when a phase slip occurs is nonlinear, and depends on the closed loop bandwidth of the synchronizer. For simplicity, these details are not taken into account.

The channel model is a *memoryless phase slip channel*, where phase slips are instantaneous, independent events occurring with probability P_s. A slip changes the phase S_i of all subsequent symbols, until a new phase slip occurs.

Typically, carrier synchronizers for QPSK and QAM introduce phase slips only toward adjacent symbols (with symmetrical probabilities, $P_s/2$ each). This behavior is included in the channel model, and in the simulation results presented in this paper.

Block differential modulation (BDM) [17] is designed to withstand phase slips. The codeword to be transmitted is mapped onto small blocks of n modulated symbols. Let $V_i^{(j)}$ be the ith symbol of the jth block. For the sake of simplicity, in the following the superscript (j) is omitted whenever possible, in particular when symbols belonging to the same n-tuple are considered.

Let $m = n - 1$. In M-PSK BDM a transmitted sequence $\{X_0, \ldots, X_{n-1}\} = \{1, V_1, \ldots, V_m\}$ and its rotated versions $\{U, UV_1, \ldots, UV_m\}$, where

$$U = \exp\left(j\frac{2\pi k}{M}\right) \quad k \in \{0, 1\} \text{ or } k \in \{0, 1, 2, 3\} \tag{3}$$

carry the same information. Due to the phase ambiguity, information is mapped onto $m = n - 1$ symbols instead of n.

However, an additional symbol $Z^{(j)}$ can be transmitted by differential encoding between adjacent blocks

$$U^{(j)} = Z^{(j)} U^{(j-1)}. \tag{4}$$

Thus, BDM is a generalized form of differential modulation, applied to blocks of symbols instead of one symbol at a time. This allows to recover the large performance gap between differential modulation and coherent transmission almost completely [17].

In case of BPSK and QPSK, the symbols U, V, Z, and X belong to the same (binary or quaternary) constellation. In case of QAM, the symbols U and Z are QPSK, while the symbols V and their rotated versions $X = UV$ are QAM. Eventually *only symbols X are transmitted.*

Roughly speaking, the information carried by symbols V is not affected by phase slips, which have an effect only on the symbols U. In the following, it is shown that BDM allows to design efficient LDPC codes for the phase slip channel, with BPSK, QPSK, and also QAM signaling.

III. LDPC Codes for BDM-BPSK

BDM for BPSK divides each codeword into blocks of n bits $x_i \in \{0,1\}$[1]. Within each block, every admissible sequence $0, v_1 ... v_m$ and its complement $1, \bar{v}_1 ... \bar{v}_m$ transmit the same information $v_1 ... v_m$. An additional information bit u is differentially encoded through the phase of the blocks. The hypothesis is that phase slips are rare enough to keep the carrier phase unchanged within two consecutive n-tuples of symbols, with high probability. If a cycle slip occurs between adjacent n-tuples, the decision on $v_1 ... v_m$ is untouched. The effect is an error in u, which is encoded through the phases of the adjacent blocks. If the cycle slip occurs inside an n-tuple also the information bits v inside the n-tuple are corrupted. This additional noise affects few symbols, yet. BDM over very short blocks, even with $n = 4$, has almost no loss compared to coherent transmission. Above 0.7 bits per dimension, no loss is expected [17, Figs. 3, 4, and 6(a)]. Besides, in presence of phase slips, the BDM capacity degrades gracefully [17, Fig. 5].

In general, achieving capacity with multilevel constellations and binary codes requires a multi-level coding (MLC) and multistage decoding (MSD) strategy [18]. For instance, if $n = 4$, there should be four different codes, one for each of the four bits mapped onto the 4-D constellation. MSD requires that the codes be decoded in cascade. Unfortunately, this increases the required memory as well as the latency.

Parallel decoding can be used to avoid MSD. In general, parallel decoding of a multilevel construction is not optimal, as some exchange of messages between different levels is required to achieve the constellation capacity. The goal is to achieve this by means of LDPC codes, thanks to the exchange of information during the iterations, which is intrinsic to LDPC decoding. In fact, the modulator for BDM can be implemented simply by adding to each m-tuple of bits $v_1, \ldots, v_m$ one reference bit u,

[1]To ease the presentation of the LDPC code, in this Section, we let $u, v, x \in \{0,1\}$ instead of considering $U, V, X \in \{+1, -1\}$.

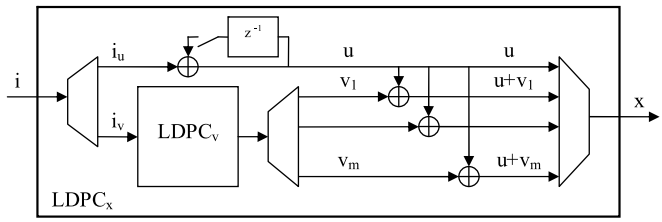

Fig. 2. The encoder for embedded LDPC coding and BDM for BPSK symbols.

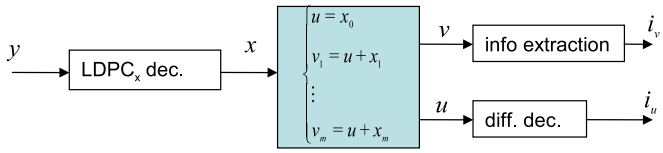

Fig. 3. Parallel $u, \ldots, u + v_m$ decoder based on iterative LDPC decoding on parity check matrix H_x.

producing the transmitted n-tuple

$$\begin{cases} x_0 = u \\ x_1 = u + v_1 \\ x_2 = u + v_2 \\ \quad \vdots \\ x_m = u + v_m . \end{cases} \tag{5}$$

The sequence of bits u is differentially encoded, and the sequence of bits v is an LDPC codeword, as shown in Fig. 2. The problem with the BDM demodulator is that if individual metrics are evaluated just once for all bits v, and sent to the decoder without iterative updating (exchanging information with the LDPC decoder), the capacity loss of differential BPSK cannot be avoided. It is essential that the metrics of bits v be updated during decoding, and the metrics of u must be refined as well. With the above choice (5), this can be done by decoding the LDPC code $\mathcal{X}$ whose parity check matrix H_x is obtained by combining the parity check matrix H_v with the equation (5) of the BDM modulator. Once the bits x are decoded, bits v and u can be derived inverting (5) and the corresponding information bits can be recovered by simple extraction (the code $\mathcal{V}$ is systematic) and differential decoding, respectively, as shown in Fig. 3.

A. LDPC Code Design

To design a capacity achieving LDPC code, its variable node (VN) and check node (CN) degree distributions must be fine tuned. The performance of LDPC code ensembles can be predicted by density evolution (DE), which is able to track the *probability density functions (pdf)* of extrinsic messages along the iterations on a tree [19].

The LDPC code to be optimized for iterative decoding is $\mathcal{X}$, but the unconstrained LDPC code is $\mathcal{V}$. Hence, the need to seek an LDPC code $\mathcal{V}$ that produces a good LDPC code $\mathcal{X}$ as, of course, there is a relationship between the degree distributions of codes $\mathcal{V}$ and $\mathcal{X}$.

Consider codeword $v_i, i = 1 ... N_v$ encoded with an LDPC code of rate $R_v = 1 - \frac{N_c}{N_v}$ of given parity check matrix

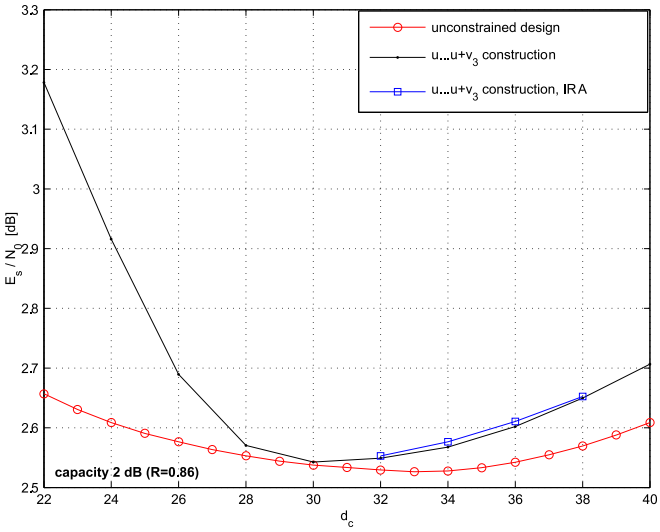

Fig. 4. Thresholds for right-regular LDPC codes under various design constraints, as a function of the CN degree (d_c) (Gaussian DE approximation).

$H_v[N_c \times N_v]$. Apply BDM equation (5) to the stream v_i. An equivalent LDPC parity matrix for the variables x_i can be derived, with $N_v\ (1+1/m)$ VNs, N_c CNs, and rate

$$R_x = 1 - \frac{N_c}{N_v + \frac{N_v}{m}} = 1 - \frac{N_c}{N_v}\frac{m}{m+1}. \tag{6}$$

Let $\rho_v(z) = \sum_j \rho_j^{(v)} z^{j-1}$ and $\lambda_v(z) = \sum_i \lambda_i^{(v)} z^{i-1}$ be the check and variable degree distributions of the LDPC code $\mathcal{V}$ (edge perspective). In the LDPC code $\mathcal{V}$ a CN of degree j is connected to j variables v, taken from different n-tuples. In the LDPC code $\mathcal{X}$ the same CN is also connected to j variables u (one for each n-tuple). Then, $\rho_{2j}^{(x)} = \rho_j^{(v)}$, i.e.,

$$\rho_x(z) = \sum_j \rho_j^{(v)} z^{2j-1} = z\rho_v(z^2). \tag{7}$$

For simplicity, let all the m variables v of an m-tuple have the same degree i. An edge connecting a variable v to a CN produces another edge connecting u to the same CN. Then, beside the mi edges of degree i stemming from variables v inside the m-tuple, there are another mi edges of degree mi stemming from the variable u. Therefore

$$\begin{aligned}\lambda_x(z) &= \frac{1}{2}\sum_i \lambda_i^{(v)} \left(z^{i-1} + z^{mi-1}\right) \\ &= \frac{1}{2}\lambda_v(z) + \frac{1}{2}z^{m-1}\lambda_v\left(z^m\right).\end{aligned} \tag{8}$$

Equations (7) and (8) are additional constraints on $\lambda_x(z)$ and $\rho_x(x)$ that may penalize the LDPC convergence compared to an unconstrained design. The DE analysis has been modified to include the above constraints. The search is limited to right-regular codes (i.e., codes with just one CN degree) to simplify the decoder implementation.

Fig. 4 plots the best thresholds, for a target BER of 10^{-4} after 16 iterations with a right-regular LDPC with CN degree d_c.

Another useful constraint is the absence of deg-2 information variables. This can be obtained by designing an irregular repeat-and-accumulate (IRA) code, with the additional benefit that the encoder is simplified [20]. The minimum CN degree compatible with these requirements is 32. Fig. 4 shows that the loss compared to an unconstrained code design is negligible. A gap of 0.55 dB from capacity is expected, notwithstanding the BDM and IRA additional constraints.

The complexity of the LDPC decoder increases with the CN degree d_c. An unconstrained design could work with, say, $d_c = 28$ (or even $d_c = 26$, if some further loss can be accepted). The increased CN degree is the (small) complexity cost to be paid for the BDM feature.

B. LDPC Decoding and Simulation

One of the advantages of the FEC scheme proposed here is that BDM demodulation is embedded into LDPC decoding of code $\mathcal{X}$, which is well suited to hardware implementation.

LDPC decoding is done by message passing (MP), iteratively activating the CNs and the VNs of the Tanner graph of the code. The message sent by an VN over an edge is the sum of the messages received from the other edges, plus the channel message. CN decoding is more complex. In a practical implementation, the $\tanh$ operations [2] of the CNs are replaced by a simple $\min$ search among the input messages, with a conditional or unconditional correction factor, leading to the so called *min-sum-offset* implementation. The best performance is obtained when the offset is *conditional*. The reader can refer to the vast literature about LDPC decoding implementation for more details (see, e.g., [21] and [22]).

Different activation strategies in the MP schedule show different advantages and drawbacks [23]. In parallel MP, at each iteration all the VNs are simultaneously activated, and extrinsic messages going toward CNs are computed. Then, all CNs simultaneously update their extrinsic information, going backward to the VNs and so on. In sequential (or *serial* or *layered*) message passing, CNs are activated sequentially. Usually, sequential decoding guarantees faster convergence than parallel message passing [23]. A heuristic assumption says that the number of iterations can nearly be halved.

In addition to the faster convergence of sequential decoding, another property makes its adoption mandatory in the case of BDM-LDPC. The construction (5) introduces unescapable short cycles (with girth four) among each variable v and its phase reference u. These cycles penalize the code performance when all parity check equations are updated in parallel. When the equations are processed sequentially, the reference variable u is updated through many other variables of each m-block, between two consecutive exchanges of messages with the same variable v.

We have designed an LDPC code of blocksize $N = 14220$ and rate $R = 0.86$ embedding BDM with $m = 3$. Performing Monte Carlo simulations, the carrier recovered at the demodulator is affected by phase slips, according to the PS channel model in Fig. 1. Phase slips are instantaneous, random, and happen with probability P_s. The BDM-LDPC code has the performance shown in Fig. 5(a) as a function of the number of iterations with $P_s = 0$, and in Fig. 5(b) as a function of

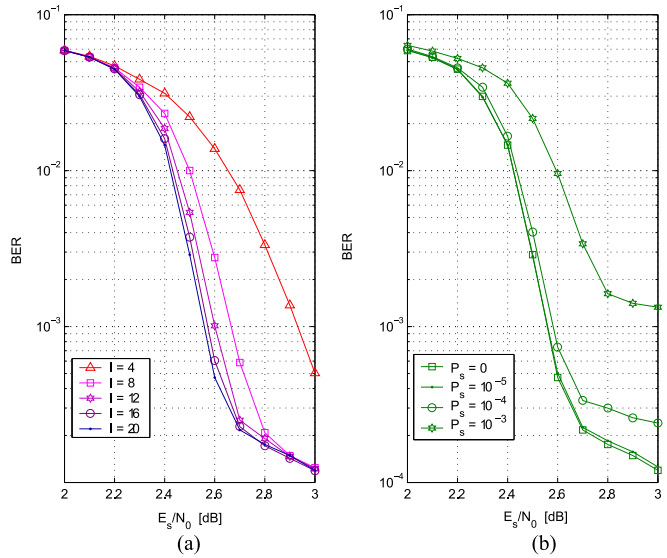

Fig. 5. Performance of BPSK BDM-LDPC, with various decoding iterations and $P_s = 0$ (a), and with various phase slip probabilities after 20 decoding iterations (b).

P_s after 20 iterations. For rare phase slips $\left(P_s \leq 10^{-4}\right)$, the BDM-LDPC achieves BER $= 10^{-3}$ with $E_s/N_0 < 2.6$ dB, with no practical loss compared to $P_s = 0$. Note that with $P_s = 10^{-4}$, about half the codewords of length $N_v = 14220$ are affected by random phase changes inside the block.

In Fig. 5(b), the error floors are due to the errors occurring in the (uncoded) information stream differentially modulated in symbols u. Each phase slip at the demodulator that occurs during the reception of an n-tuple produces an error in the information bit i_u of the reference phase symbol u. Since this occurs with probability $\approx nP_s$, the expected error floor is

$$\mathrm{BER}_f\left(P_s\right) \approx nP_s \frac{N_v/m}{N_v/m + R_v N_v} = \frac{P_s}{R_x}. \qquad (9)$$

There is also an error floor due to the small minimum distance of the code. The BDM construction with $n = 4$ produces codewords at Hamming distance 4. The corresponding error floor, visible in Fig. 5(b) when $P_s = 0$, has no effect on the concatenated code.

With an outer hard-decoded code such as the one described in ITU G975-I9 [24], it was verified by FPGA that an input BER approximately equal to $2 \cdot 10^{-3}$ achieves an overall BER below 10^{-13}. Fig. 5(b) shows that any P_s below 10^{-4} is completely transparent to the system. BDM works also at $P_s = 10^{-3}$, at the cost of some SNR increase (0.2 dB, roughly)[2].

IV. LDPC Codes for BDM-QPSK

In this Section, the approach presented so far is generalized to QPSK modulation, to double the spectral efficiency to 2 bits per symbol. With 25 Gbaud and dual polarization mode, the target of 100 Gbit/s transmission is reached. The goal of this code design is a system still robust to occasional phase slips, if possible without loss compared to coherent transmission and detection, and with little additional complexity.

Now we let $u, v, x \in \{0, 1, 2, 3\}$ instead of considering $U, V, X \in \{\exp(jk\pi/2)\}$. Again the BDM scheme corresponds to (5), mod 4 of course. The stream of variables u is differentially encoded. Once again, BDM demodulation is embedded into the LDPC decoding of symbols v to avoid increasing the receiver complexity. The difference here is that false locks may induce phase slips (i.e., carrier rotations) at multiples of $\pi/2$. To build a symbol sequence "robust" to these phase rotations, an LDPC code with symbols in the ring $\mathbb{Z}_4$, namely a quaternary LDPC (QLDPC) code, must be designed. As for BPSK, each parity equation of an LDPC code $\mathcal{V}$ involving the ith VN of the jth m-tuple $v_i^{(j)}$ is translated into the corresponding parity equation of the LDPC code $\mathcal{X}$ that includes also its phase reference $u^{(j)} = x_0^{(j)}$

$$\sum_{i,j} v_i^{(j)} = 0 \quad \Rightarrow \quad \sum_{i,j} \left(x_i^{(j)} - x_0^{(j)}\right) = 0 \mod 4. \qquad (10)$$

A. QLDPC Code Design

The design of non-binary LDPC codes has been investigated in particular for finite fields $GF(q)$ [25], [26], [27]. Many aspects also hold for LDPC codes in rings. Still, they have some peculiarities compared to $GF(q)$ LDPC codes, discussed in [28] and [29]. In [28], irregular LDPC codes are designed for M-PSK, on the basis of EXIT curves empirically evaluated. In [30], an LDPC decoder proposed for a sampled and wrapped Gaussian channel aims at achieving almost the same performance as a (vectorial) BP decoder aware of the sampling and wrapping effects, yet with smaller complexity.

The ring $\mathbb{Z}_4$ is not widely investigated, mainly because Gray-mapped QPSK is equivalent to BPSK, and thus, binary LDPC codes can be used with QPSK modulation without any constellation penalty. In [31], a $\mathbb{Z}_4$ LDPC code is designed with degree distributions borrowed from binary LDPC codes, without an explicit justification. The use of these polynomials in conjunction with Gray mapping and simplified decoding is justified in [29]. Therein, it is shown that irregular QLDPC codes can be designed on the ring $\mathbb{Z}_4$ as if they were binary LDPC codes, with uniform distribution of the parity check coefficients in the reduced ring $\{1, 3\}$. It is apparent from (10) that in the parity check matrix of the LDPC code $\mathcal{X}$ only coefficients $\{+1, -1\} \equiv \{1, 3\}$ are needed, with an exact uniform distribution. This perfectly matches the aforementioned problem.

The transmitted symbols are Gray mapped (separable) QPSK symbols whose phases are labeled according to $k \in \{0, 1, 2, 3\}$. The QLDPC code is defined by a sparse parity check matrix H whose entry $h_{j,i} \in \mathbb{Z}_4$ is the coefficient of the ith QPSK symbol x_i in the jth parity equation, i.e.:

$$\sum_i h_{j,i} x_i = 0, \quad j = 1 \ldots N_c. \qquad (11)$$

[2]Beyond $P_s = 10^{-3}$, concatenation with the outer code is not enough to tackle the error floor. In this case, only pure differential modulation can produce a BER below $2 \cdot 10^{-3}$ at the input of the outer decoder. This implies the use of a differential soft-out demodulator, but does not require changes in the LDPC decoder. Of course, an increased SNR is required, due to the capacity loss of differential modulation.

B. QLDPC Decoding Issues

Non-binary LDPC decoding has been widely investigated, e.g., in [25], [32], [33], and [34], paying attention to performance and complexity issues.

Ideal BP decoding is much heavier than that for binary codes, both in terms of computational burden and memory requirements. In fact, messages are q-ary *pdf*s instead of scalar LLRs, for both $\mathrm{GF}(q)$ or $\mathbb{Z}_q$. Besides q-ary *pdf*s need be multiplied at VNs and convolved at CNs to produce extrinsic messages.

Soft CN decoders (like soft VN decoders), receive and output extrinsic messages. These are the likelihoods of the four values of a symbol x, given the received symbol sequence $\mathbf{y}$. Messages can take the form of a vector of likelihoods or probabilities (after normalization) $\mathbf{p}$ with $p_i = p(\mathbf{y}|x=i)$, $i = 0, 1, 2, 3$; of a vector of log likelihoods (LL) $\mathbf{L}$ with $L_i = \log p_i$; or of a vector of DFT transformed likelihoods $\mathbf{f} = \mathbf{F}\mathbf{p}$, with $F_{i,k} = j^{ik}$, where $i, k = 0, 1, 2, 3$. These three forms are equivalent. Hereinafter, "message" refers to any of these forms.

In the log-probability domain, products (that are badly suited for hardware implementation) are replaced with sums, and sums are replaced with $\max$ (or LUT-$\max$) operations. Let $\mathbf{L}_0$ be the channel likelihood of the variable v_i, and let $\mathbf{L}_{v_i \leftarrow c_j}$ be the incoming message from CN c_j. Then, $\mathbf{L}_{v_i \rightarrow c_m}$, the output extrinsic message to CN c_m is

$$\mathbf{L}_{v_i \rightarrow c_m} = \mathbf{L}_0 + \sum_{j=1, j\neq m}^{d_v} \mathbf{L}_{v_i \leftarrow c_j}. \tag{12}$$

This solves the computational complexity at VNs. At CNs convolutions are quite heavy. For large fields $GF(q)$ or rings $\mathbb{Z}_q$, the replacement of convolutions with DFTs and products can be convenient, lowering the complexity of the single CN processing from $O(d_{cX}\, q^2)$ to $O(d_{cX}\, q \log q)$. Some way to overcome the implementation of products is still needed [34]. Let $\mathbf{p}_{c_j \leftarrow v_i}$ be the incoming message from VN v_i to CN c_j, and let $\mathbf{p}_{c_j \rightarrow v_m}$ be the output extrinsic message to VN v_m. Then, $\mathbf{p}_{c_j \rightarrow v_m}$ is the convolution of all input messages $\mathbf{p}_{v_i \rightarrow c_j}$ $\forall i \neq m$, and its DFT $\mathbf{f}_{c_j \rightarrow v_m}$ reads

$$\mathbf{f}_{c_j \rightarrow v_m} = \prod_{i=1, i\neq m}^{d_c} \odot \left(\mathbf{W}_{j,i} \mathbf{f}_{c_j \leftarrow v_i}\right) \tag{13}$$

where $\odot$ is the Hadamard product. In general, $\mathbf{W}_{j,i}$ is a $q \times q$ permutation matrix that takes into account the coefficient $h_{c_j \leftarrow v_i}$ in (11) associated to the edge $c_j \leftarrow v_i$. With $q = 4$ and $h_{c_j \leftarrow v_i} \in \{1, 3\}$, there are only two cases. When $h_{c_j \leftarrow v_i} = 3$, $\mathbf{W}_{j,i}$ is a 4×4 identity matrix, whereas if $h_{c_j \leftarrow v_i} = 1$, $\mathbf{W}_{j,i}$ swaps f_1 and f_3 of the vector $\mathbf{f}_{c_j \leftarrow v_i}$. Note that this also corresponds to exchanging p_1 and p_3 in the *pdf*.

In this case ($q = 4$), there is no practical advantage in performing DFTs because the ring is small, not to mention that treating q-ary instead of scalar messages only increases memory requirements. To simplify message storage and evaluation, the *pdf*s of the $\mathbb{Z}_4$ symbols have been approximated with the separable *pdf* that can be described by the two LLRs m_M and m_L of the two bits transmitted in phase (MSB) and quadrature (LSB), according to the Gray mapping: $00 \rightarrow 0$, $01 \rightarrow 1$, $11 \rightarrow 2$, $10 \rightarrow 3$. During the decoding process, each extrinsic vector $\mathbf{p}$ generated by the CN decoders is marginalized so that at each VN the combination of separable *pdf*s produces a separable *pdf*. The following relations hold between a separable *pdf* $\mathbf{p}$, or its LL $\mathbf{L}$, and the LLRs of its bits $\mathbf{m}$:

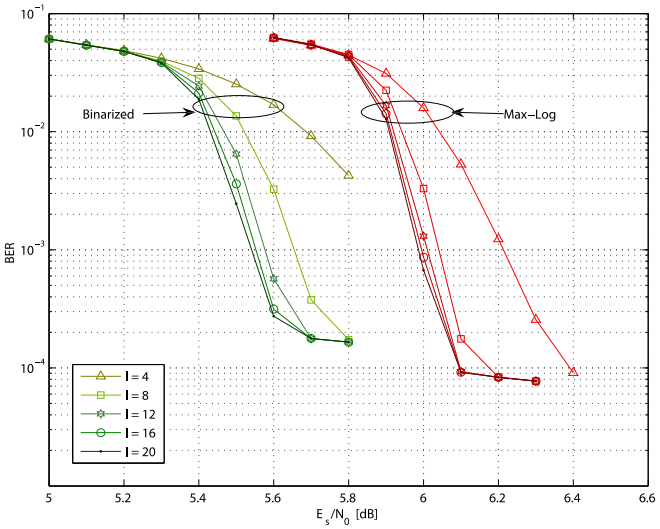

Fig. 6. Performance of the QLDPC under binarized decoding and MAX-Log decoding, with phase slip probability $P_s = 0$ at the demodulator, after various decoding iterations.

$$\mathbf{p} = \begin{bmatrix} e^{m_M/2} e^{m_L/2} \\ e^{m_M/2} e^{-m_L/2} \\ e^{-m_M/2} e^{-m_L/2} \\ e^{-m_M/2} e^{m_L/2} \end{bmatrix}, \quad L_i = \log p_i \tag{14}$$

$$m_M = \log \frac{p_0 + p_1}{p_2 + p_3}, \quad m_L = \log \frac{p_0 + p_3}{p_2 + p_1}. \tag{15}$$

Normalization of $\mathbf{p}$ is not necessary.

A comparison of the performance of the two decoders in Fig. 6 with (binarized) and without marginalization (quaternary), shows that the loss induced by marginalization is less than 0.05 dB while the simplification is precious. The memory requirements are the same as those for binary LDPC codes (1 scalar message per bit). In the binarized form, each CN receives a vector of d_c LLR pairs $\mathbf{m}^{(i)}$ and for the VN v_n has to evaluate the extrinsic *pdf* $\mathbf{p}_{c_j \rightarrow v_n}$, namely π', i.e., the *pdf* of the symbol:

$$x'_n = -\sum_{i\neq n} h_{j,i} x_i. \tag{16}$$

After exchanging the two LLRs of symbols x_i when $h_{j,i} = 1$, the problem reduces to the convolution of $d_c - 1$ separable *pdf*s. The separable *pdf* of x'_n is determined by the pair

$$\mu'_M = \log \frac{\pi'_0 + \pi'_1}{\pi'_2 + \pi'_3}$$

$$\mu'_L = \log \frac{\pi'_0 + \pi'_3}{\pi'_2 + \pi'_1}. \tag{17}$$

The final result does not change when marginalized at each step or just once at the end of the convolutions [29]. Thus,

the problem can be solved recursively, knowing how to derive the pair $\boldsymbol{\mu}'$ arising from convolution-and-marginalization of the separable *pdf*s $\mathbf{p}$ and $\boldsymbol{\pi}$ of two symbols, defined by the pairs $\mathbf{m}$ and $\boldsymbol{\mu}$, respectively.

In the log-domain, the resulting LL $\boldsymbol{\Lambda}'$ can be evaluated exactly by convolution as

$$\Lambda'_i = \log\left(\sum_{j=0}^{3} \exp\left(\Lambda_j + L_{i-j}\right)\right), \quad i \in \mathbb{Z}_4 \qquad (18)$$

where $\mathbf{L}$ and $\boldsymbol{\Lambda}$ are the LL linked to $\mathbf{m}$ and $\boldsymbol{\mu}$ through (14). A well-known approximation of (18), namely MAX-Log, considers only the dominant term in the sum, and (18) reduces to

$$\Lambda'_i \approx \Lambda^*_i = \max_{j=0\ldots3}\{\Lambda_j + L_{i-j}\}, \quad i \in \mathbb{Z}_4. \qquad (19)$$

A full CN processing requires $d_c - 2$ convolutions of $d_c - 1$ *pdf*s. This can be accomplished by a forward and backward, BCJR-style algorithm, with only $3d_c$ convolutions.

C. Code Design and Simulation

An irregular QLDPC has optimal degrees that obey the same equations as a binary irregular LDPC [29]. Therefore, degree polynomials can be taken from the binary LDPC design of Section III. With the same LDPC code rate as in the BPSK case, the transmission rate is doubled thanks to the doubled spectral efficiency of QPSK.

The number of CNs (now operating in $\mathbb{Z}_4$) is the same as in the BPSK case. Like the LDPC project for BPSK modulation, there is one phase reference variable u every third variable v.

After 20 sequential iterations the code with block length $N_v = 14220$ reaches the target BER at $E_s/N_0 = 5.5$ dB under ideal quaternary decoding, at 0.5 dB from capacity for QPSK (Fig. 6).

Fig. 7 compares the performance of the ideal quaternary and binarized decoders, in presence of phase slips. The performance of the decoder with MAX-Log (19) convolutions has been plotted as a reference. The phase slip probability P_s ranges from 0 to $2 \cdot 10^{-3}$. The QLDPC code could withstand also sudden phase rotations by π, should they ever occur.

Each slip produces an error in the differentially modulated symbol u. Thanks to Gray mapping, each slip affects only one bit. Therefore, phase slips produce an error floor given by

$$\mathrm{BER}_f \approx \frac{P_s}{2R_x}. \qquad (20)$$

This error floor prevents the use of a coherent receiver above $P_s = 2 \cdot 10^{-3}$. A worsening of the SNR is visible only with $P_s > 10^{-4}$, and is within one or two tenths of dB[3].

V. Application of QLDPC to 16-QAM

The goal of this design is to double the bit rate using the same FEC code designed for QPSK, moving from 1 bit per dimension (QPSK) to 2 bits per dimension (16-QAM).

[3]When the probability of phase slip exceeds $2 \cdot 10^{-3}$ pure differential modulation must be used, as in the BPSK case.

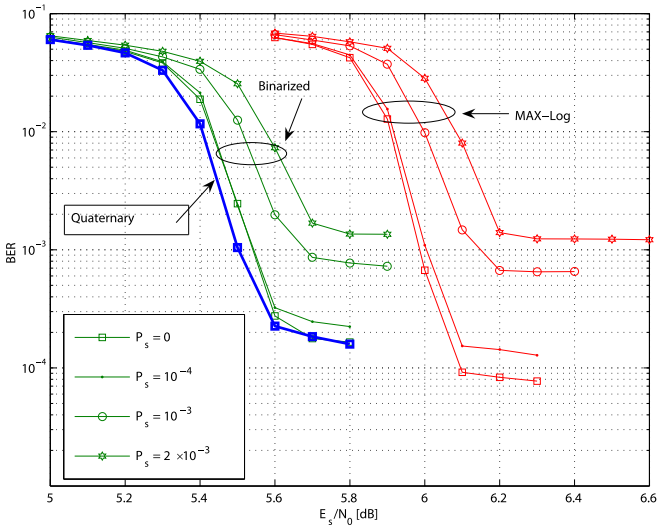

Fig. 7. Performance of the QLDPC under Quaternary, binarized decoding and MAX-Log decoding, with phase slip probability P_s at the demodulator, after 20 sequential iterations.

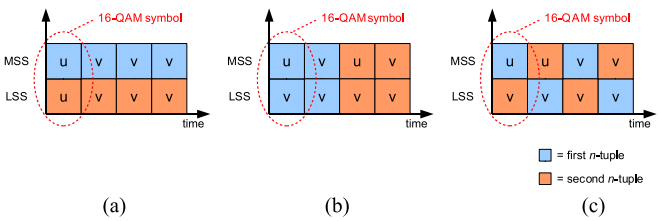

Fig. 8. Three possible mappings on 16-QAM symbols: QPSK n-tuples can be transmitted on MSS/LSS streams as in (a), on 16-QAM symbols as in (b), or interleaved as in (c).

The QLDPC code is designed to be robust to phase rotations at multiples of $\pi/2$. The 16-QAM constellation exhibits the same symmetry, thus, the recovered carrier has the same phase ambiguity as QPSK. The QLDPC encoder/decoder is used also for the new modulation. However, the constellation mapping must be chosen carefully, to ensure that robustness to phase slips is maintained. The points of the 16-QAM constellation are labeled with a pair of $\mathbb{Z}_4$ symbols, namely the most significant symbol (MSS) and the least significant symbol (LSS). The labeling must insure that a rotation of the axes is mapped into a rotation of the $\mathbb{Z}_4$ symbols. In principle, the rotations in the $\mathbb{Z}_4$ labels MSS and LSS can be independent, provided that a whole n-tuple is mapped onto the same $\mathbb{Z}_4$ stream (MSS or LSS), as in Fig. 8(a). If the same rotation rule is used for MSS and LSS, the n-tuple can be split on the two streams. In this case, it is preferable not to map a pair from the same n-tuple onto the same 16-QAM symbol, as shown in Fig. 8(b). A very short interleaver, involving two n-tuples and n 16-QAM symbols, is enough to avoid this impairment [see Fig.8 (c)]. Simulations confirm that such a simple interleaver is beneficial. However, with high P_s, the mapping of Fig. 8(b) is preferable, because each slip affects a single n-tuple.

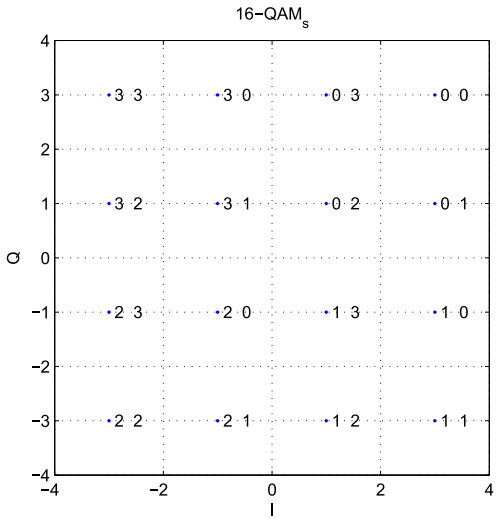

Fig. 9. Constellation with *separable* mapping 16-QAM$_S$.

Exhaustive inspection of all possible mappings found that the largest capacity with coherent detection can be obtained either by using the same rotation rule or by leaving one of the labels (LSS) invariant to rotations. Eventually two mappings were chosen. One, namely 16-QAM$_S$, has bit labeling separable on the two axes, thus has a simpler demodulator. The second, namely 16-QAM$_I$ offers a better capacity and a lower BER. Other mappings have been discarded because they offered less capacity, or because the demodulation was too complex and the capacity gain was negligible.

A. The Separable 16-QAM Mapping

Among many different choices with the same capacity, the bit labeling 16-QAM$_S$ shown in Fig. 9 with Gray mapping of $\mathbb{Z}_4$ symbols $\{0 = 00, 1 = 01, 2 = 11, 3 = 10\}$ is separable on the two axes. As the QLDPC decoder works with bit LLR inputs, this allows to design a very simple demodulator. Basically each axis is divided into M intervals: the in-phase component of the received signal determines the LLRs of the MSBs of the two $\mathbb{Z}_4$ symbols (MSS and LSS), whereas the quadrature component determines the LLRs of the LSBs of the two $\mathbb{Z}_4$ symbols. The 16-QAM$_S$ constellation with bit-by-bit LLR output under coherent modulation has a 0.6 dB penalty compared to Gray mapped 16-QAM. For instance, the capacity is 3.45 bits per symbol with $E_s/N_0 = 11.9$ dB, instead of $E_s/N_0 = 11.3$ dB with Gray mapping. There is a (small) advantage in mapping all symbols u over MSSs as in Fig.8 (c). The reason is that MSS bits carry the largest capacity in this constellation. The block diagram of the transmitter for the 16-QAM$_S$ constellation is plotted in Fig. 10.

Fig. 11 plots the BER at the output of the QLDPC decoder, as a function of the number of decoding iterations (a) and of P_s (b). With low P_s the target BER $= 2 \cdot 10^{-3}$ is obtained with $E_s/N_0 = 12.6$ dB, less than 0.7 dB far from capacity.

In Fig. 11(b), the probability of slip ranges from 0 to $5 \cdot 10^{-3}$. With $P_s > 10^{-3}$ mapping (b) is preferable. As for QPSK, each

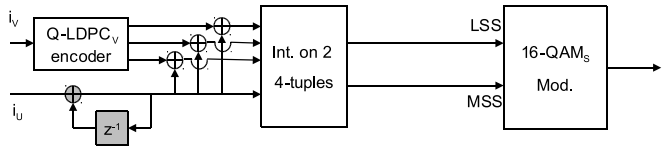

Fig. 10. Block diagram of the transmitter for 16-QAM$_S$ constellation.

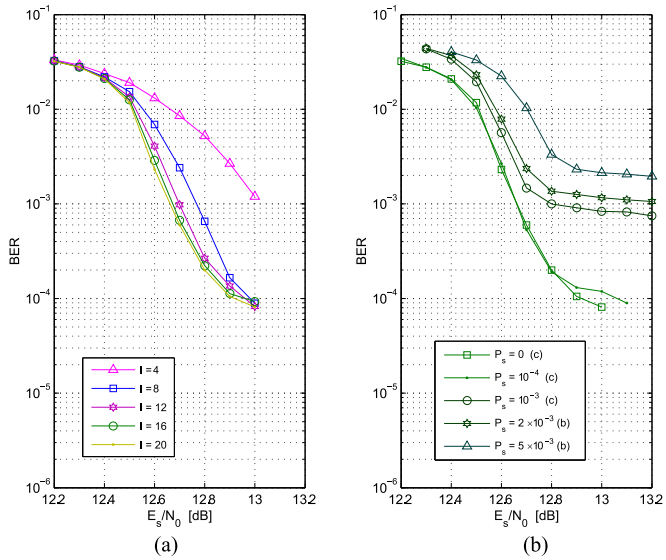

Fig. 11. Performance of the QLDPC decoder with 16QAM$_S$ after various sequential iterations with $P_s = 0$ (a) and after 20 sequential iterations with various P_s (b).

PS produces an error in one phase reference u. Each 16-QAM symbol carries a pair of $\mathbb{Z}_4$ symbols, thus the error floor at high SNR is half the level given by (20) for QLDPC-BDM and the BDM scheme works up to $P_s = 5 \cdot 10^{-3}$.

B. The Invariant 16-QAM Mapping

Another interesting choice is the "invariant" mapping, shown in Fig. 12, hereinafter referred to as 16-QAM$_I$. The $\mathbb{Z}_4$ LSS are invariant to carrier phase slips, while the MSS $\mathbb{Z}_4$ symbols rotate with the axes as in 16-QAM$_S$. Thus, for the n-tuples on the LSS stream, there is no need to accumulate the symbols u. Only n-tuples mapped on the MSS stream need an accumulator. Fig. 13 shows the block diagram of the transmitter for 16-QAM$_I$. Note the stream separation before accumulation and mapping.

The 16-QAM$_I$ constellation with bit-by-bit LLR output exhibits a small advantage compared to 16-QAM$_S$. Note that the LSSs in this constellation are invariant, and can be modulated and demodulated coherently even under frequent phase slips. Thus, this constellation is more robust to the inherent "noise" due to frequent phase slips, as it only affects the MSS stream that is more protected.

Fig. 14(a) plots the BER at the output of the QLDPC decoder after various decoding iterations with $P_s = 0$. Fig. 14(b) shows the effect of slip probabilities ranging from 0 to $5 \cdot 10^{-3}$. With

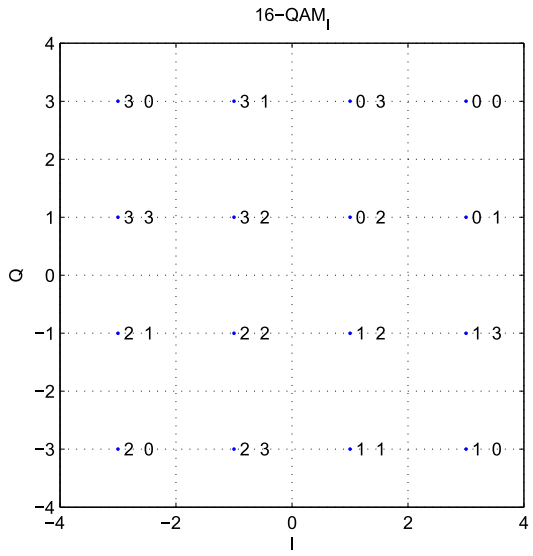

Fig. 12. Constellation with *invariant LSS* mapping: 16-QAM$_I$.

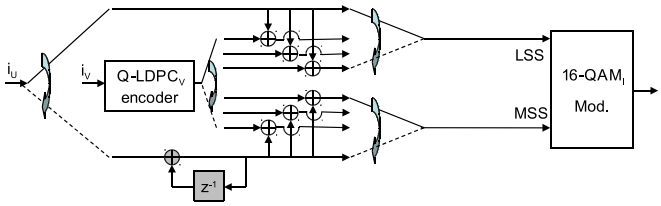

Fig. 13. Block diagram of the transmitter for 16-QAM$_I$.

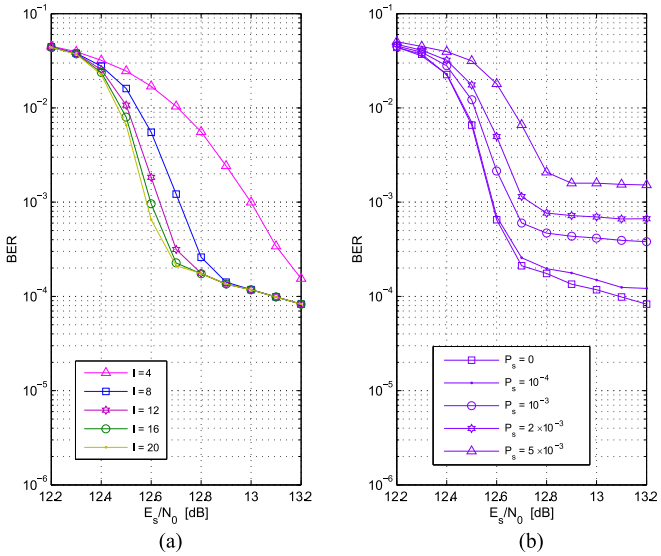

Fig. 14. Performance of the QLDPC decoder with 16QAM$_I$ after various sequential iterations with $P_s = 0$ (a) and after 20 sequential iterations with various P_s (b).

low P_s, there is a small gain compared to 16-QAM$_S$ (0.1 dB at $P_s = 10^{-3}$). When $P_s = 5 \cdot 10^{-3}$, the 16-QAM$_I$ takes advantage of the fact that the equivalent noise due to slips affects only the MSS stream, and converges more rapidly to its error floor. This gives a 0.4 dB advantage in SNR over 16-QAM$_S$.

VI. Conclusion

Current 100 G transceivers employ polarization multiplexed QPSK and QAM modulation formats and intradyne digital coherent receivers. In an intradyne transmission system, blind carrier phase recovery may incur errors, causing phase slips. The overall channel including blind phase estimation can be modeled as an AWGN channel, affected by phase slips.

Inspired by the capacity analysis of block differential modulation on the PS channel [17], this paper proposed an inner FEC design based on LDPC codes that embeds block differential demodulation, to make it robust to occasional phase slips. This additional feature comes at almost the same cost as a standard LDPC decoder, and performs very close to SD-FEC codes for an ideal coherent channel. A binary code for BPSK-BDM has been designed. Then, the concept has been extended to QPSK, with an LDPC code design in $\mathbb{Z}_4$ (QLDPC). Finally, there exist (at least) two mappings for the QLDPC code on 16-ary QAM symbols that maintain the robustness to occasional cycle slips of the carrier phase. This allows to further double the spectral efficiency of the system.

References

[1] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.

[2] R. G. Gallager, "Low Density Parity Check Codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[3] R. Pyndiah, "Near optimum decoding of product codes: block turbo codes," *IEEE Trans. Commun.*, vol. 46, no. 8, pp. 1003–1010, Aug. 1998.

[4] T. Mizuochi, "Recent progress in forward error correction and its interplay with transmission impairments," *IEEE J. Sel. Topics Quantum Electron.*, vol. 12, no. 4, pp. 544–554, Jul./Aug. 2006.

[5] S. Dave, L. Esker, F. Mo, W. Thesling, J. Keszenheimer, and R. Fuerst, "Soft-decision forward error correction in a 40-nm ASIC for 100-Gbps OTN applications," presented at the *Optical Fiber Communication Conf. and Expo.*, Los Angeles, CA, USA, Mar. 6–10 2011, pp. 1–3, Paper JWA14.

[6] B. Vasic and I. Djordjevic, "Low-density parity check codes for long-haul optical communication systems," *IEEE Photon. Technol. Lett.*, vol. 14, no. 8, pp. 1208–1210, Aug. 2002.

[7] I. Djordjevic, M. Arabaci, and L. Minkov, "Next Generation FEC for high-capacity communication in optical transport networks," *J. Lightw. Technol.*, vol. 27, no. 16, pp. 3518–3530, May 2009.

[8] D. Chang, Y. Fan, X. Zhiyu, N. Stojanovic, F. Hauske, C. Yi, X. Changsong, L. Liangchuan, X. Xiaogeng, and X. QianJin, "LDPC convolutional codes using layered decoding algorithm for high speed coherent optical transmission," presented at the Optical Fiber Communication Conf. and Expo., Anaheim, CA, USA, Mar. 17–21 2012, pp. 1–3, Paper OW1H.4.

[9] K. Sugihara, Y. Miyata, T. Sugihara, K. Kubo, H. Yoshida, W. Matsumoto, and T. Mizuochi, "A spatially-coupled type LDPC code with an NCG of 12 dB for optical transmission beyond 100 Gb/s," presented at the *Optical Fiber Communication Conf. and Expo.*, Los Angeles, CA, USA, Mar. 6–8 2013, pp. 1–3, Paper OM2B.4.

[10] T. Mizuochi, Y. Konishi, Y. Miyata, T. Inoue, K. Onohara, S. Kametani, T. Sugihara, K. Kubo, H. Yoshida, T. Kobayashi, and T. Ichikawa, "Experimental demonstration of concatenated LDPC and RS codes by FPGAs emulation," *IEEE Photon. Technol. Lett.*, vol. 21, no. 18, pp. 1302–1304, Sep. 2009.

[11] N. Kamiya and S. Shioiri, "Concatenated QC-LDPC and SPC codes for 100 Gbps ultra long-haul optical transmission systems," presented at the *Fiber Communication Conf. Optical Society of America, San Diego, CA, USA*, Mar. 21–25 2010, pp. 1–3, Paper OThL2.

[12] C.-S. Choi, H. Lee, N. Kaneda, and Y.-K. Chen, "Concatenated nonbinary LDPC and HD-FEC codes for 100 Gb/s optical transport systems," in

Proc. IEEE Int. Symp. Circuits Syst., Seoul, Korea, May. 20–23 2012, pp. 1783–1786.

[13] D. Chang, F. Yu, Z. Xiao, Y. Li, N. Stojanovic, C. Xie, X. Shi, X. Xu, and Q. Xiong, "FPGA verification of a single QC-LDPC code for 100 Gb/s optical systems without error floor down to BER of 10^{-15}," presented at the *Optical Fiber Communication Conf. and Expo.*, Los Angeles, CA, USA, Mar. 6–10 2011, pp. 1–3, Paper OTuN2.

[14] D. A. Morero, M. A. Castrillon, F. A. Ramos, T. A. Goette, O. E. Agazzi, and M. R. Hueda, "Non-concatenated FEC codes for ultra-high speed optical transport networks," in *Proc. IEEE Globecom*, Houston, TX, USA, Dec. 2011, pp. 1–6.

[15] C. R. S. Fludger, D. Nuss, and T. Kupfer, "Cycle-slips in 100G DP-QPSK transmission systems optical fiber communication conference and exposition," presented at the *Optical Fiber Communication Conf. and Expo.*, Los Angeles, CA, USA, Mar. 4–8 2012, pp. 1–3, Paper OTu2G.1.

[16] A. Bisplinghoff, S. Langenbach, T. Kupfer, and B. Schmauss, "Turbo differential decoding failure for a coherent phase slip channel," presented at the *38th Eur. Conf. and Exhib. on Optical Communication*, Amsterdam, Netherlands, Sep. 16–20 2012, pp. 1–3, Paper Mo.1.A.5.

[17] S. Bellini, M. Ferrari, and A. Tomasoni, "On the constrained capacity of block differential modulation and detection," *IEEE Trans. Commun.*, vol. 61, no. 7, pp. 2840–2852, Jul. 2013.

[18] U. Wachsmann, R. Fischer, and J. Huber, "Multilevel codes: Theoretical concepts and practical design rules," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1361–1391, May 2001.

[19] S.-Y. Chung, T. Richardson, and R. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 657–670, Feb. 2001.

[20] Y. Zhang and W. Ryan, "Structured IRA codes: Performance analysis and construction," *IEEE Trans. Commun.*, vol. 55, no. 5, pp. 837–844, May 2007.

[21] J. Chen, A. Dholakia, E. Eleftheriou, M. P. C. Fossorier, and X.-Y. Hu, "Reduced-complexity decoding of LDPC codes," *IEEE Trans. Commun.*, vol. 53, no. 8, pp. 1288–1299, Aug. 2005.

[22] J. Zhao, F. Zarkeshvari, and A. H. Banihashemi, "On implementation of min-sum algorithm and its modifications for decoding low-density parity-check (LDPC) codes," *IEEE Trans. Commun.*, vol. 53, no. 4, pp. 549–554, Apr. 2005.

[23] E. Sharon and J. G. S. Litsyn, "Efficient serial message-passing schedules for LDPC decoding," *IEEE Trans. Inf. Theory*, vol. 53, no. 11, pp. 4076–4091, Nov. 2007.

[24] *Forward Error Correction for High Bit-Rate DWDM Submarine Systems*, ITU-T Recommendation G.975.1, 2003.

[25] M. C. Davey and D. J. C. Mackay, "Low density parity check codes over GF(q)," *IEEE Commun. Lett.*, vol. 2, no. 6, pp. 165–167, Jun. 1998.

[26] A. Bennatan and D. Burshtein, "Design and analysis of nonbinary LDPC codes for arbitrary discrete-memoryless channels," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 549–583, Feb. 2006.

[27] G. Li, I. Fair, and W. Krzymien, "Density evolution for nonbinary LDPC codes under Gaussian approximation," *IEEE Trans. Inf. Theory*, vol. 55, no. 3, pp. 997–1015, Mar. 2009.

[28] D. Sridhara and T. E. Fuja, "LDPC codes over rings for PSK modulation," *IEEE Trans. Inf. Theory*, vol. 51, no. 9, pp. 3209–3220, Sep. 2005.

[29] M. Ferrari, S. Bellini, and A. Tomasoni, "Low-complexity $\mathbb{Z}_4$ LDPC code design under a Gaussian approximation," *IEEE Wireless Commun. Lett.*, vol. 1, no. 6, pp. 589–592, Dec. 2012.

[30] G. Montorsi, "Analog digital belief propagation: From theory to practice," in *Proc. IEEE Int. Conf. Commun.*, Ottawa, ON, Canada, Jun. 2012, pp. 2591–2595.

[31] S. Karuppasami and W. Cowley, "Construction and iterative decoding of LDPC codes over rings for phase-noisy channels," *EURASIP J. Wireless Commun.*, vol. 2008, pp. 385421-1–385421-9, 2008.

[32] D. Declerq and M. Fossorier, "Decoding algorithms for nonbinary LDPC codes over GF(q)," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 633–634, Apr. 2007.

[33] V. Savin, "Min-Max decoding for nonbinary LDPC codes," in *Proc. Int. Symp. Inf. Theory*, Toronto, Canada, Jul. 2008, pp. 960–964.

[34] C. Spagnol, E. Popovici, and W. Marnane, "Hardware implementation of GF(2^m) LDPC decoders," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 12, no. 12, pp. 2609–2620, Dec. 2009.