

Optical LDPC decoders for beyond 100 Gbits/s optical transmission

Ivan B. Djordjevic,^{1,*} Lei Xu,² and Ting Wang²

¹Electrical and Computer Engineering, University of Arizona, 1230 E. Speedway Boulevard, Tucson, Arizona 85721, USA

²NEC Laboratories America, 4 Independence Way, Princeton, New Jersey 08540, USA

*Corresponding author: ivan@ece.arizona.edu

Received February 9, 2009; revised March 25, 2009; accepted March 27, 2009;
posted April 2, 2009 (Doc. ID 107332); published April 27, 2009

We present an *optical* low-density parity-check (LDPC) decoder suitable for implementation above 100 Gbits/s, which provides large coding gains when based on large-girth LDPC codes. We show that a basic building block, the probabilities multiplier circuit, can be implemented using a Mach-Zehnder interferometer, and we propose corresponding probabilistic-domain sum-product algorithm (SPA). We perform simulations of a fully parallel implementation employing girth-10 LDPC codes and proposed SPA. The girth-10 LDPC(24015,19212) code of the rate of 0.8 outperforms the BCH(128,113) × BCH(256,239) turbo-product code of the rate of 0.82 by 0.91 dB (for binary phase-shift keying at 100 Gbits/s and a bit error rate of 10⁻⁹), and provides a net effective coding gain of 10.09 dB. © 2009 Optical Society of America
OCIS codes: 060.4510, 130.3120, 060.2330, 200.4560, 230.2090.

Network operators consider 100 Gbits/s per wavelength channel transmission as the transmission technology for the next generation of Ethernet. However, the performance of fiber-optic communication systems operating at those data rates is degraded significantly owing to several transmission impairments including intrachannel and interchannel nonlinearities, the nonlinear phase noise, and polarization-mode dispersion (PMD). To deal with these channel impairments, development of a novel powerful forward-error correction (FEC) scheme suitable for beyond 100 Gbits/s transmission, future 100 Gbits/s, and 1 Tbit/s Ethernet is of the utmost importance [1].

Given the lack of A/D converters operating at data rates above 100 Gbits/s, we propose an alternative FEC scheme based on the optical low-density parity-check (LDPC) decoding. One of the first researchers to propose the use of analog decoders in electrical domain to avoid the problem of nonexistence analog-to-digital (A/D) converters at high speeds was Hagenauer *et al.* [2]. Hagenauer *et al.* showed in [2] that analog decoders, implemented in electrical domain, perform comparably to the corresponding digital implementation but can operate at higher speeds. Here we propose to implement a LDPC decoder in the optical domain by using the integrated optics. Another major difference of our approach is the use of a probabilistic-domain implementation instead of the log domain used in [2]. We propose a particular version of probabilistic decoding algorithm that is applicable to arbitrary degree nodes. To avoid the error-floor phenomena we employ large-girth ($g \geq 10$) quasicyclic LDPC codes, which do not exhibit an error floor down to the bit error rate (BER) of 10⁻¹⁵ [1]. (The girth represents the shortest cycle in corresponding bipartite graph representation of a parity-check matrix.)

The sum-product algorithm (SPA) [3] is an iterative LDPC decoding algorithm in which extrinsic

probabilities are iterated forward and backward between variable (bit) and check nodes of bipartite (Tanner) graph representation of a parity-check matrix (see [3] for more details). Let $q_{i \rightarrow j}(b)$ denote the extrinsic information to be passed from variable node v_i to function node c_j regarding the probability that $v_i = b$, where $b \in \{0, 1\}$; and let $r_{j \rightarrow i}(b)$ denote the extrinsic information to be passed from check node c_j to variable node v_i , which represents the probability that the j th parity-check equation is satisfied given that $v_i = b$, and other bits connected to the same function node have separable distribution given by $\{q_{i \rightarrow j'}\}_{j' \neq j}$. Before the probabilistic SPA begins we have to calculate the bit log-likelihood ratios (LLRs) $L(v_i) = \log[\Pr(v_i = 0 | y_i) / \Pr(v_i = 1 | y_i)]$ ($i = 1, 2, \dots, N$), where $L(v_i)$ denotes a LLR of the i th bit v_i in codeword v of length n , and y_i is the corresponding receiver sample.

The probabilistic-domain SPA can be described as follows:

(0) Initialization: $q_{i \rightarrow j} = P_i = [1 + \exp(L(v_i))]^{-1}$.

(1) The first half iteration:

$$r_{j \rightarrow i} = q_{l \rightarrow j}(1 - q_{m \rightarrow j}) + (1 - q_{l \rightarrow j})q_{m \rightarrow j}.$$

(2) The second half iteration:

$$q_{i \rightarrow j} = r_{s \rightarrow i} r_{t \rightarrow i} / [r_{s \rightarrow i} r_{t \rightarrow i} + (1 - r_{s \rightarrow i})(1 - r_{t \rightarrow i})].$$

(3) Variable-node update:

$$P_i \leftarrow \frac{r_{s \rightarrow i} r_{t \rightarrow i}}{r_{s \rightarrow i} r_{t \rightarrow i} + (1 - r_{s \rightarrow i})(1 - r_{t \rightarrow i})},$$

$$P_i \leftarrow \frac{P_i r_{j \rightarrow i}}{P_i r_{j \rightarrow i} + (1 - P_i)(1 - r_{j \rightarrow i})}.$$

(4) Decision step:

$$\hat{v}_i = 1 \text{ for } \log[(1 - P_i)/P_i] < 0 \quad (0 \text{ otherwise}).$$

The initialization and decision steps require LLR-to-probability and probability-to-LLR conversions,

which can be done by using differential bipolar transistor pairs as shown in Fig. 1 of [2]. In the case when the variable and check-node degrees are larger than 3 we have to implement steps (1) and (2) in a similar fashion as we did in step (3). In such a way, we do not need to modify the bipartite graph by introducing the hidden states so that all check and variable nodes are of degree 3. Notice also that because of the way we performed the initialization in step (0), only probabilities $r_{j \rightarrow i}$ and $q_{i \rightarrow j}$ are to be memorized in steps (1) and (2) while in standard probability-domain SPA [3] the four probabilities are required: $r_{j \rightarrow i}(0)$, $r_{j \rightarrow i}(1)$, $q_{i \rightarrow j}(0)$, and $q_{i \rightarrow j}(1)$, plus the normalization in steps (1)–(3).

Steps (1)–(3) are performed by circuits given in Figs. 1(b) and 1(c), respectively. Both the parity-check node probability update circuit [see Fig. 1(b)] and the variable-node probability update circuit [see Fig. 1(c)] are implemented based on Mach-Zehnder interferometer (MZI) multiplier circuit shown in Fig. 1(a). Using the directional coupler theory it can be

shown that electrical fields of output ports $E_{out,1}$ and $E_{out,2}$ are related to the input electrical field E_{in} by $E_{out,1} = jE_{in} \sin(\Delta\varphi/2)$ and $E_{out,2} = jE_{in} \cos(\Delta\varphi/2)$, where $\Delta\varphi$ is the phase shift introduced by voltage applied across electrodes in the upper branch of the phase shifter. The corresponding output powers are $P_{out,1} = P_{in} \sin^2(\Delta\varphi/2)$ and $P_{out,2} = P_{in} \cos^2(\Delta\varphi/2)$. The input power is distributed between output ports as follows: $P_{out,1} + P_{out,2} = P_{in}$. By normalizing the last two equations we obtain

$$P_{out,1}/P_{in} = \sin^2(\Delta\varphi/2) = p, \quad P_{out,2}/P_{in} = 1 - p. \quad (1)$$

Obviously, the parameter $0 \leq p \leq 1$ and satisfies the axioms of probability. Let the input power be proportional to the probability of a bit at input a being 1, P_a , and the control voltage of the phase shifter be proportional to the probability of a bit at input b being 1, P_b . According to Eq. (1), the output power of the upper branch will be proportional to $P_a P_b$, while the output power of lower branch to $P_a(1 - P_b)$. Therefore, the MZI circuit can be used as a probability multiplier circuit, which is a basic building block to implement steps (1)–(3). The normalization required in steps (2) and (3) for the implementation in electrical domain requires the use of analog dividers, while in the optical domain the normalization can be performed by using an optical amplifier because $1/[P_a P_b + (1 - P_a) \times (1 - P_b)] \geq 1$ [see Fig. 1(d)].

The implementation of an MZI probability multiplier [Fig. 1(a)] requires one optical input (input a) and one electrical input (input b), and for conversion from optical to electrical domain a p-i-n photodetector is to be employed. Given the recent advances in photonic integrated circuits (PICs), the optical implementation of the LDPC decoder by using PIC technology is possible. Because the proposed optical decoder is an analog device, it does not require the use of high-speed A/D converters and logical gates, and as such it is an interesting candidate for beyond 100 Gbits/s optical transmission.

The parity-check matrix H of quasicyclic LDPC codes [4] considered in this Letter can be represented by

$$H = \begin{bmatrix} I & I & I & \dots & I \\ I & P^{S[1]} & P^{S[2]} & \dots & P^{S[c-1]} \\ I & P^{2S[1]} & P^{2S[2]} & \dots & P^{2S[c-1]} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ I & P^{(r-1)S[1]} & P^{(r-1)S[2]} & \dots & P^{(r-1)S[c-1]} \end{bmatrix}, \quad (2)$$

where I is a $p \times p$ (p is a prime number) identity matrix, P is a $p \times p$ permutation matrix (with elements $p_{i,i+1} = p_{p,1} = 1$, where $i = 1, 2, \dots, p-1$; other elements of P are 0), while r and c represent the number of rows and columns in Eq. (2), respectively. The set of integers S are to be carefully chosen from the set $\{0, 1, \dots, p-1\}$ so that the cycles of short length, in corresponding Tanner (bipartite) graph representation of Eq. (2), are avoided. The codeword length is $N = |S|p$, where $|S|$ denotes the cardinality of set S , and the code rate is lower bounded by $(1 - r/|S|)$.

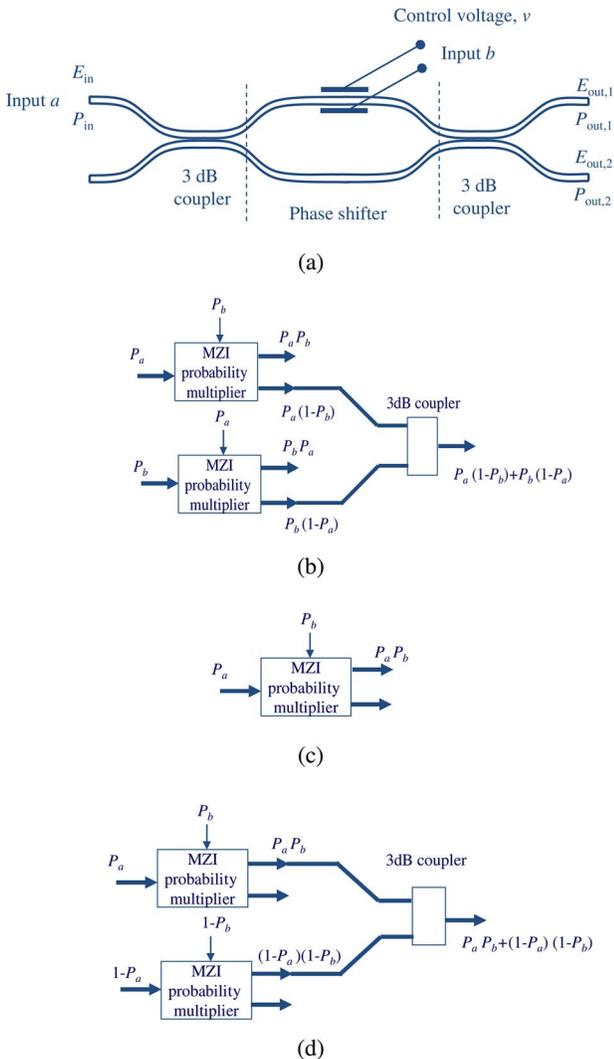


Fig. 1. (Color online) SPA implementation building blocks: (a) MZI probability multiplier, (b) parity-check node probability update circuit, (c) variable-node probability update circuit, and (d) circuit to calculate the normalization factor required in steps (2) and (3).

The different options to perform decoding in an optical domain can be classified as (i) serial, (ii) fully parallel, and (iii) partially parallel implementations. In the serial version, one parity-check node update circuit [Fig. 1(b)] is needed to perform step (1), one variable-node probability update circuit [Fig. 1(c)] and one normalization circuit [Fig. 1(d)] are needed to perform step (2), and one variable-node probability update circuit and one normalization circuit are needed to perform step (3). Therefore, the hardware requirements for serial implementation are rather small; only eight MZI probability multiplier circuits are needed. However, the latency of this implementation is high. Moreover, since MZIs are lossy devices we need to perform the amplification periodically, which is inherent to amplified spontaneous emission (ASE) noise addition and will lead to performance degradation.

In fully parallel implementation, for an (N, K) LDPC code, N bit-node update circuits, and $(N-K)$ check-node update circuits operate in parallel. The fully parallel decoder requires $2(4N-K)$ MZI multiplier circuits and has therefore the highest complexity among three different decoder architectures, but the lowest latency. Moreover, since all bit/check-node processing elements operate in parallel, this scheme is the least sensitive to attenuation introduced by MZIs.

In partially parallel implementation, we have to group the bit nodes and the check nodes into a set of larger nodes along the submatrix borders in Eq. (2), which are called here the bit processing elements (BPEs) and check processing elements (CPEs), respectively. The bit nodes (check nodes) in BPEs (CPEs) are processed in a serial fashion, while all BPEs (CPEs) perform their operations in parallel. The total number of MZI multiplier circuits is $8cr$.

The results of simulations obtained by employing probabilistic-domain SPA introduced above (that performs the same as log-domain SPA described in [3]), and fully parallel implementation of SPA, are shown in Fig. 2 for a data rate of 100 Gbits/s. To avoid the need for storage elements to keep intermediate results [as in step (3)], we use the previous result as the input to a new variable-node/check-node update circuit in addition to the other input. With this approach, we have to increase the number of required MZI multiplier circuits to perform the optical LDPC decoding, but we solve the problem of nonexistence of appropriate storage elements. Here we compare (i) the girth-10 quasicyclic LDPC codes (of column weight $c=3$ and code rate $R=0.8$) with parity-check matrix (2) and (ii) the girth-8 LDPC code (of column weight $c=4$ and code rate $R=0.8$) (both for 30 iterations in SPA algorithm described above) against RS, concatenated RS, and turbo-product codes. The girth-10 LDPC(24015,19212) code of the rate of 0.8

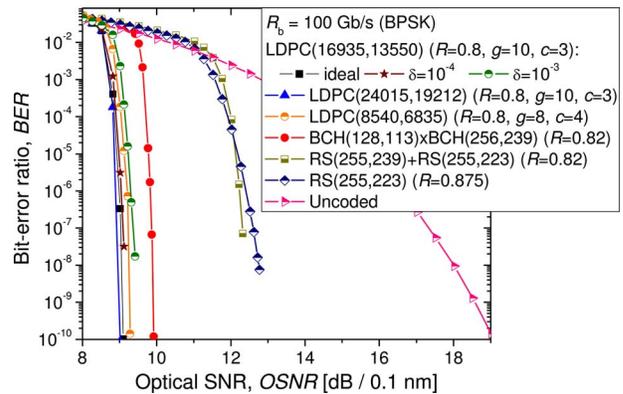


Fig. 2. (Color online) BER performance of large-girth LDPC codes employing the optical LDPC decoder operating at 100 Gbits/s for binary phase-shift keying (BPSK) modulation format.

outperforms the RS(255,223) (of the rate of 0.875) by 3.79 dB at a BER of 10^{-8} . At a BER of 10^{-10} the same LDPC code outperforms the BCH(128,113) \times BCH(256,239) TPC of the rate of 0.82 by 0.91 dB and provides the net effective coding gain of 10.09 dB. The girth-8 LDPC(8540,6835) code outperforms the same TPC by 0.61 dB at a BER of 10^{-10} . In all simulations, except for the LDPC(16935,13550) code, we assume that parity-check node and variable-node probabilities update circuits (shown in Fig. 2) are ideal. For the LDPC(16935,13550) code we provide the simulations with different accuracy δ of parity-check node (variable-node) probability update circuits. For accuracy δ of 10^{-4} degradation with respect to the ideal case is 0.08 dB at a BER of 3×10^{-8} , while for δ of 10^{-3} corresponding degradation is 0.36 dB.

In summary, we present an optical probability-domain LDPC decoder suitable for implementation at data rates above 100 Gbits/s. Because it is essentially an analog device, it does not require the use of A/D converters and logical gates and as such represents an interesting candidate for very-high-speed implementation (well above 100 Gbits/s) and for future 100 Gbits/s and 1 Tbit/s Ethernet. We show that the basic building block, the probabilities multiplier circuit, required to implement both check node and probability-node update circuits, can be implemented using a Mach-Zehnder delay interferometer.

References

1. Y. Miyata, W. Matsumoto, and T. Mizuochi, in *Proceedings of the OFC/NFOEC* (2008), paper OTuE4.
2. J. Hagenauer, M. Moerz, and A. Schaefer, in *Proceedings of the 2002 International Zurich Seminar on Broadband Communications* (2002), pp. 3-1-3-6.
3. W. E. Ryan, in *CRC Handbook for Coding and Signal Processing for Recording Systems* (CRC Press, 2004).
4. M. P. C. Fossorier, *IEEE Trans. Inf. Theory* **50**, 1788 (2004).