Improved Low-Power LDPC FEC for Coherent Optical Systems

Kevin Cushon[(1)], Per Larsson-Edefors[(2)], Peter Andrekson[(3)]

[(1)] Computer Science and Engineering, Chalmers University of Technology, cushon@chalmers.se
[(2)] Computer Science and Engineering, Chalmers University of Technology, perla@chalmers.se
[(3)] Photonics Laboratory, MC2, Chalmers University of Technology, peter.andrekson@chalmers.se

Abstract *We propose and demonstrate a low-complexity LDPC FEC system for coherent optical applications. Implementation results show an estimated NCG of 11.0 dB with 20% overhead, 160 Gbps throughput, and energy consumption of 3.4 pJ per bit.*

Introduction

Forward error correction (FEC) is a critical component of modern optical communication systems, which demand FEC with very high performance, featuring throughput of at least 100 Gbps, low power consumption, and coding gain approaching the theoretical limit[3]. There is a continual demand for higher performing FEC, as higher net coding gain (NCG) permits longer maximum ranges and increased capacity.

Soft-decision (SD) FEC operates on reliability measures of received symbols, and thus can achieve higher performance than hard-decision (HD) FEC, which operate only on the most likely received symbols. In coherent optical systems, soft information is readily available, so SD-FEC is possible. Turbo product codes (TPCs)[2] and low-density parity check (LDPC) codes[5,7] are the most commonly proposed FEC codes for such systems. Spatially-coupled (SC) LDPC codes, which provide greater NCG than block LDPC codes, have been the focus of much recent work[3,9,10]. Staircase codes, a type of SC HD code that achieve coding gains approaching SD codes, have also been proposed for optical transport applications[11].

TPCs and LDPC codes are decoded using iterative message passing algorithms, which are quite costly in terms of computation and power consumption. It has been estimated that SD-FEC using LDPC codes consumes approximately 15-20% of the total energy in a long-haul 100 Gbps coherent optical link[8]. The application-specific integrated circuit (ASIC) implementations of these high-throughput, computationally complex decoders also require high silicon area, which translates directly to high capital cost. Careful consideration of the trade-offs between conflicting performance and complexity requirements is therefore needed when designing such FEC systems[6].

We previously proposed the adaptive degeneration (AD) LDPC decoding algorithm as a compromise between performance and complexity for SD-FEC in coherent optical communication systems[1]. In this paper, we present an improved version of the AD algorithm, called the "prior-assisted AD" (PAD) algorithm, so named because it preserves prior information and uses it throughout decoding. In the following sections, we describe the algorithm, then present field-programmable gate array (FPGA) simulation and ASIC synthesis results of a PAD decoder using a (36000, 30000) quasi-cyclic (QC) LDPC code. It achieves an estimated NCG of 11.0 dB at a bit error rate (BER) of 10^{-15} with 20% coding overhead (OH), which represents an improvement of about 0.4 dB over standard AD, and reduces the gap to normalized min-sum algorithm (NMSA) based FEC to 0.4 dB.

The PAD Algorithm

An LDPC code is characterized by an $m \times n$ parity check matrix $\mathbf{H}$, with n bits and m parity checks. A bit i participates in parity check j iff $\mathbf{H}_{j,i} = 1$. If all parity checks are met, the n bits form a valid codeword. An (n, k) LDPC code has k information bits and $n - k$ parity bits. An LDPC code can also be described with a Tanner graph, in which the columns of $\mathbf{H}$ are represented by variable nodes (VNs) v_i, and the rows by check nodes (CNs) c_j. An edge connects v_i and c_j iff $\mathbf{H}_{j,i} = 1$. The degree d_c of a CN or d_v of a VN is equal to the number of edges connecting to it. The set of VNs connecting to CN c is represented by V_c, and the set of CNs connecting to VN v is C_v.

The PAD algorithm is described in Algorithm 1, while free parameters are listed and described in Table 1, along with their values used in this implementation. Like the AD algorithm, the VN memories M_v are initialized with the LLRs of symbols received from the channel. The VNs and CNs exchange binary messages $b_{v \to c}$ and $b_{c \to v}$. The val-

Algorithm 1 The PAD Decoding Algorithm

for all $v \in [0 \,..\, n-1]$ **do**
  $M_v \leftarrow LLR_v$
  $\kappa_v \leftarrow \text{sgn}(M_v)$
  $\lambda_v \leftarrow \max(\gamma_1, |M_v|)$
end for
$i, \epsilon_2, \epsilon_1, \epsilon_0 \leftarrow 0$
$u_\ell, u_{\ell-1}, \ldots, u_0 \leftarrow m+1$
repeat
  for all $v \in [0 \,..\, n-1]$ **do**
    $b_{v\to c} \leftarrow \begin{cases} 0, & \text{if } M_v \geq 0 \\ 1, & \text{if otherwise} \end{cases}$
    $h_v \leftarrow b_{v\to c}$
  end for
  for all $c \in [0 \,..\, m-1]$ **do**
    $p_c = \bigoplus_{v \in V_c} (b_{v\to c})$
    for $v \in V_c$ **do**
      $b_{c\to v} = p_c \oplus b_{v\to c}$
    end for
  end for
  $u \leftarrow u << 1$
  $u_0 \leftarrow \sum p$
  if $u_0 = 0$ **then**
    Declare decoding successful, output h
  end if
  if $\epsilon_0 = 1$ **then**
    $\delta \leftarrow \gamma_1$
    $u_\ell, u_{\ell-1}, \ldots, u_0 \leftarrow m+1$
  else
    $\delta \leftarrow \gamma_0$
  end if
  for all $v \in [0 \,..\, n-1]$ **do**
    $\rho \leftarrow 0$
    if $\lambda_v \geq T_0$ **then** $\rho \leftarrow \kappa_v \cdot \epsilon_1 \cdot \epsilon_2$
    if $\lambda_v \geq T_1$ **then** $\rho \leftarrow \kappa_v \cdot \epsilon_1$
    $\sigma \leftarrow d_v - 2 \cdot \sum_{c \in C_v} b_{c\to v} + \rho$
    $M_v \leftarrow \begin{cases} M_v + s\cdot\sigma - \delta, & \text{if } M_v \geq 0 \\ M_v + s\cdot\sigma + \delta, & \text{if otherwise} \end{cases}$
    if $M_v < -\lambda_v$ **then** $M_v \leftarrow -\lambda_v$
    if $M_v > \lambda_v$ **then** $M_v \leftarrow \lambda_v$
  end for
  $\epsilon_2, \epsilon_1, \epsilon_0 \leftarrow 0$
  if $u_0 \geq u_\ell$ **and** $u_0 < \tau$ **and** $i < i_m - \ell$ **then** $\epsilon_0 \leftarrow 1$
  if $u_0 \geq \tau$ **then** $\epsilon_1 \leftarrow 1$
  if $i \mod 4 = 3$ **then** $\epsilon_2 \leftarrow 1$
  $i \leftarrow i + 1$
until $i = i_m$
Declare decoding failed, output h

ues of M_v are then updated by summing the $b_{c\to v}$ messages, and adding this sum to the previous value of M_v, along with a degeneration factor δ. Normally, the value of δ is the small constant γ_1, but if no decoding progress has been made in the previous ℓ iterations (as defined by a reduction in the number of unsatisfied parity checks), a larger value γ_1 is used.

Tab. 1: Description of Free Parameters in Algorithm 1

	Description	Value
s	CN-to-VN message scaling factor, $0 < s \leq 1$	0.5
γ_0, γ_1	Possible values of δ, $0 \leq \gamma_0 < \gamma_1$	0.5, 3.0
ℓ	How many iterations to look back when calculating δ	6
τ	Threshold of unsatisfied parity checks, affects δ and ρ	64
T_0, T_1	Thresholds for prior LLRs to contribute to σ, $0 < T_0 < T_1$	4.0, 7.5

The PAD algorithm adds measures that improve trapping set correction capability in LDPC codes with $d_v = 4$, as well as performance in the waterfall region. The major addition is the storage and use of prior LLR information in κ_v and λ_v. If the prior LLR magnitude λ_v is sufficiently large, then its sign κ_v acts as an additional input to the VN message sum, either on every 4th iteration (if $T_0 < \lambda_v < T_1$), or on every iteration if λ_v is very large ($\lambda_v \geq T_1$). This is disabled via control register ϵ_1 when the number of unsatisfied parity checks falls below a threshold τ, since it can interfere with the correction of trapping sets containing VNs with high-magnitude but incorrect λ_v.

Additionally, M_v is prevented from having a higher magnitude than λ_v. This measure increases the ability of the algorithm to correct certain classes of trapping sets, in which incorrect VNs would otherwise allow M_v to increase to the maximum magnitude. To prevent correct bits from being flipped, λ_v cannot be set smaller than γ_1. This threshold was found to achieve a good balance between trapping set correction capability and avoiding mass flips of correct bits.

Decoder Implementation Results

To characterize the performance of the PAD algorithm, we implemented a PAD decoder for a (36000, 30000) regular QC-LDPC code with $d_v = 4$ and $d_c = 24$, constructed using the finite field subset method[4]. LLRs and M_v all use 5 bit quantization with a standard fixed-point format of 1 sign bit, 3 integer bits, and 1 fractional bit. The maximum number of iterations i_m is set to 74. The decoder architecture is fully parallel (the FPGA implementation is partially parallel, but emulates a fully parallel design).

Fig. 1 shows frame error rate (FER) and BER performance results obtained from software

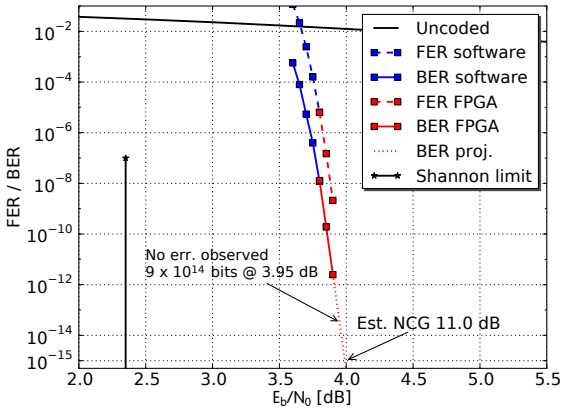

Fig. 1: FER/BER plot of the implemented PAD decoder with (36000, 30000) QC-LDPC code.

FPGA implementations using BPSK modulation over an AWGN channel. No failures due to small trapping sets were observed with E_b/N_0 set to 3.9 dB, and no errors were observed in $9 \cdot 10^{14}$ bits at 3.95 dB. Based on these observations, we estimate NCG of 11.0 dB at a BER of 10^{-15} by extrapolating the BER curve[1,9]. This is roughly 1.65 dB away from the Shannon limit. Compared to other FEC systems, this decoder's estimated NCG is about 0.4 dB less than NMSA with the same OH[7], 0.4 dB more than AD[1], and 0.6 dB more than hard-decision staircase codes[11].

ASIC synthesis results for this decoder are summarized in Table 2. The fabrication process is STMicroelectronics 28 nm FD-SOI, using a nominal supply voltage of 0.9 V. The power estimate was obtained via netlist simulation with back-annotated parasitics, and random codewords with E_b/N_0 set to 4.0 dB. Due to the PAD algorithm's increased complexity, silicon area and energy consumption are both about 50% higher than AD. Throughput is lower as well, which is primarily due to slower convergence of the $d_v = 4$ LDPC code compared to the $d_v = 6$ codes used with AD, which necessitates a higher maximum number of iterations.

However, the estimated energy consumption of 3.37 pJ per bit remains several times lower than reported figures for more complex FEC systems. Estimates for the energy consumption of SD-LDPC decoders for coherent optical systems range from 20 pJ per bit[5] to 60 pJ per bit[8] in 28 nm fabrication technologies, and 70 pJ per bit for a TPC decoder in 40 nm CMOS[2].

Conclusions

The PAD algorithm is more complex than AD, but achieves significantly higher NCG. With an estimated NCG of 11.0 dB with 20% OH, and energy consumption of 3.37 pJ per bit, it represents an excellent trade-off for FEC in coherent optical systems where cost and power consumption are high priorities.

Tab. 2: Synthesis Results Using 28 nm FD-SOI

Cell area (mm^2)	7.72 (6.74)[a]
Clock freq. (MHz)	400
Max. iterations	74
Info. throughput (Gbps)	160
Latency (ns)	187.5
Power (mW)	539 (491)[a]
Energy (pJ / info. bit)	3.37 (3.07)[a]

[a] Decoder core only (i.e., excluding the I/O register buffers).

Acknowledgements

This work was supported by a grant from the Knut and Alice Wallenberg Foundation.

References

[1] K. Cushon, et al. Low-power 400-Gbps soft-decision LDPC FEC for optical transport networks. J. Lightw. Technol., 34(18):4304–4311, Sept 2016.

[2] S. Dave, et al. Soft-decision forward error correction in a 40-nm ASIC for 100-Gbps OTN applications. In OFC 2011, Mar. 2011.

[3] A. Leven, et al. Status and recent advances on forward error correction technologies for lightwave systems. J. Lightw. Technol., 32(16):2735–2750, Aug 2014.

[4] J. Li, et al. Algebraic quasi-cyclic LDPC codes: Construction, low error-floor, large girth and a reduced-complexity decoding scheme. IEEE Trans. Commun., 62(8):2626–2637, Aug 2014.

[5] M. Li, et al. Low-overhead low-power-consumption LDPC-based FEC solution for next-generation high-speed optical systems. In OFC 2015, Mar. 2015.

[6] D. A. Morero, et al. Design tradeoffs and challenges in practical coherent optical transceiver implementations. J. Lightw. Technol., 34(1):121–136, Jan 2016.

[7] D. A. Morero, et al. Experimental demonstration of a variable-rate LDPC code with adaptive low-power decoding for next-generation optical networks. In IPC 2016, pages 307–308, Oct 2016.

[8] B. Pillai, et al. End-to-end energy modeling and analysis of long-haul coherent transmission systems. J. Lightw. Technol., 32(18):3093–3111, Sept 2014.

[9] L. Schmalen, et al. Spatially coupled soft-decision error correction for future lightwave systems. J. Lightw. Technol., 33(5):1109–1116, March 2015.

[10] L. Schmalen, et al. On the design of capacity-approaching unit-memory spatially coupled LDPC codes for optical communications. In ECOC 2016, pages 1–3, Sept 2016.

[11] L. Zhang, et al. Staircase codes with 6% to 33% overhead. J. Lightw. Technol., 32(10):1999–2002, May 2014.