

Reducing the Power Consumption of the CMA Equalizer Update for a Digital Coherent Receiver

Daniel Cardenas, Domaniç Lavery, Philip Watts and Seb J. Savory

University College London, Department of Electronic and Electrical Engineering, Torrington Place, London, WC1E 7JE, UK
d.cardenas@ucl.ac.uk

Abstract: A reduced complexity multiplier-free CMA equalizer update is proposed and synthesized for a 10 Gb/s receiver using a 45-nm CMOS process. The proposed algorithm allows up to 52% power consumption reduction without penalty in performance.

OCIS codes: (060.1660) Coherent communications; (060.2330) Fiber optics communications

1. Introduction

Digital coherent receivers translate the full optical field of a signal into the digital domain enabling the compensation of transmission impairments. However, digital signal processing (DSP) complexity and, correspondingly, power consumption are among the main issues that still need to be addressed in order to prove a cost-effective solution and enable power efficient implementation in an application specific integrated circuit (ASIC). This is particularly true in access networks, where such characteristics would allow the deployment of ultra-dense wavelength division multiplexing passive optical networks. In this and similar scenarios, low complexity DSP subsystems are particularly desirable for economic reasons. One significant contribution to the power consumption of the DSP comes from adaptive equalization therefore, in this paper, we report on the performance and power consumption of different hardware architectures of a constant modulus algorithm (CMA) adaptive equalizer for a 10 Gb/s dual polarization quadrature phase shift keying (DP-QPSK) coherent receiver.

2. Equalizer algorithm and complexity analysis

The CMA adaptive equalizer is formed by a filter and a tap weight update algorithm that computes an error term which is used to update the coefficients of the filter [1]. The filter stage is implemented with four finite impulse response (FIR) filters of N taps in a butterfly structure and its output samples are given by

$$x_{out} = \mathbf{h}_{xx}^H \mathbf{x}_{in} + \mathbf{h}_{xy}^H \mathbf{y}_{in} ; \quad y_{out} = \mathbf{h}_{yx}^H \mathbf{x}_{in} + \mathbf{h}_{yy}^H \mathbf{y}_{in} \quad (1)$$

where \mathbf{h}_{xx} , \mathbf{h}_{xy} , \mathbf{h}_{yx} and \mathbf{h}_{yy} are vectors of N taps complex coefficients, \mathbf{x}_{in} and \mathbf{y}_{in} are vectors of N input complex samples of the respective x and y polarizations. The taps are computed recursively by

$$\begin{aligned} \mathbf{h}_{xx} &\leftarrow \mathbf{h}_{xx} + \mu e_x \mathbf{x}_{in} x_{out}^* & \mathbf{h}_{yx} &\leftarrow \mathbf{h}_{yx} + \mu e_y \mathbf{x}_{in} y_{out}^* \\ \mathbf{h}_{xy} &\leftarrow \mathbf{h}_{xy} + \mu e_x \mathbf{y}_{in} x_{out}^* & \mathbf{h}_{yy} &\leftarrow \mathbf{h}_{yy} + \mu e_y \mathbf{y}_{in} y_{out}^* \end{aligned} \quad (2)$$

where $e_x = 1 - |x_{out}|^2$ and $e_y = 1 - |y_{out}|^2$ denote the error terms for x and y polarizations, respectively, and μ is the learning parameter which is taken as a negative power of 2 for convenience of implementation in hardware (as a bit shift operation). The gradient term is the multiplication between the error and the output sample e.g. $e_x x_{out}$.

Each of the four FIR filters requires N complex multiplications per symbol, therefore the whole filtering stage requires $4N$ complex multipliers. There are a few options to implement the error term computation in hardware; the direct form comprises 2 real multipliers for each polarization (4 real multipliers in total).

It is clear from Eq. (2) that improving the algorithm for updating the coefficients will have a significant impact on power consumption. Herein, we present a complexity analysis of the tap weight update algorithm in terms of hardware resources allocated for arithmetic operations (adders and multipliers) for two implementations: a) direct implementation and b) sign-sign algorithm.

In the direct implementation of the tap weight update, as from Eq. (2), the error is a real-valued number and the filter output is complex, thus a total of 4 real multipliers are needed for computing the gradient; this is then multiplied by the input vector, requiring $4N$ complex multipliers as in the filtering process. We note that in this implementation not only the complexity of the filtering and tap updating increases with N , but also that tap updating contributes to more than half of the equalizer complexity.

In the sign-sign tap weight update algorithm (see, for example [2]), the updates can be simplified by keeping only the sign of the error term and the output sample, which is essentially using only the sign of the gradient to update the taps. The following modified *sgn* function determines the sign of a real number x by

$$\text{sgn}(x) = \begin{cases} -1 & x < 0 \\ 1 & x \geq 0 \end{cases} \quad (3)$$

We define a *complex signum* function by treating the real and imaginary components of a complex number z , independently

$$csgn(z) = sgn(\text{Re}(z)) + j sgn(\text{Im}(z)) \quad (4)$$

The tap update can then be written as follows:

$$\begin{aligned} \mathbf{h}_{xx} &\leftarrow \mathbf{h}_{xx} + \mu sgn(e_x) \mathbf{x}_{in} csgn(x_{out}^*) & \mathbf{h}_{yx} &\leftarrow \mathbf{h}_{yx} + \mu sgn(e_y) \mathbf{x}_{in} csgn(y_{out}^*) \\ \mathbf{h}_{xy} &\leftarrow \mathbf{h}_{xy} + \mu sgn(e_x) \mathbf{y}_{in} csgn(x_{out}^*) & \mathbf{h}_{yy} &\leftarrow \mathbf{h}_{yy} + \mu sgn(e_y) \mathbf{y}_{in} csgn(y_{out}^*) \end{aligned} \quad (5)$$

From Eqs. (3) and (5) the advantage in terms of reduced complexity when using the *signum* function is evident as it requires only 1 bit of information (the sign of the gradient), which is computed by retaining or inverting the sign of, for example, $csgn(x_{out})$ according to $sgn(e_x)$. As a consequence, an additional important advantage of this implementation arises in that the product of the sign of the gradient by the input vector \mathbf{x}_{in} does not require multipliers but only adders. Table 1 shows a summary of the ‘‘arithmetic’’ resources for both implementations, direct and sign-sign, where a complex multiplier has been implemented with 4 real multipliers and 2 adders.

Table 1. Complexity of Tap Weight Update Implementations

Resource/Implementation	Direct	Sign-sign
Real Multipliers	16N+4	0
Real Adders	16N	16N
Bit Shifts	4	4
Inverters	6	12N+6
Comparators	0	6

3. Hardware implementation

Both configurations of the CMA equalizer are considered: the direct and sign-sign implementations. Note that, clock frequency is limited when realizing algorithms in hardware, therefore the equalizer must be implemented in parallel. Thus, the required resources listed in Table 1 scale linearly according to the number of parallel branches. However, the tap update algorithm for the parallel implementation need only run once per cycle, and therefore does not need to be fully parallel (provided the channel does not change significantly between cycles). Therefore, we also evaluated the performance of a low power version of the tap update stage (which is not fully parallel) for both configurations; direct and sign-sign algorithms. We remark that our interest is not only to see the impact of the sign-sign in the reduction of power consumption compared to the direct implementation but also to achieve the lowest complexity architecture.

A previous analysis [3] concluded that the performance of the equalizer did not improve significantly if employing more than 3 taps; additionally, an input and coefficients resolution as low as 4 bits showed a 1.8 dB penalty with respect to a resolution of 8 bits, but with the obvious advantage of being less complex and requiring an ADC of only 4 bits which results in lower complexity. Therefore, as the aim in this work is also to maintain low complexity, we implemented a 3 tap CMA equalizer, 2 samples per symbol and 4 bits of resolution for input and coefficients.

The equalizer was tested with experimental data obtained from the setup described in [4] and shown in Fig. 1, it comprises a DP-QPSK transmitter followed by 100km of single mode fiber (SMF), an optical attenuator and a coherent receiver. We considered a 20% redundancy for a forward error correction (FEC) with a bit error rate (BER) correction threshold of 1.1×10^{-2} , giving a total bit rate of 12 Gb/s which translates into a symbol rate of 3 Gbaud. External cavity lasers (ECL) were used at the transmitter and as a local oscillator (LO) at the receiver. The received signal was captured by a digital storage oscilloscope and resampled at 2 samples per symbol for off-line testing of the equalizer.

The four different architectures were synthesized for hardware on a 45-nm technology process, 1.1V standard cell library, following a methodology for ASIC implementation similar to the one described in [5].

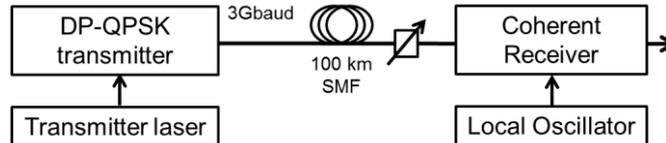


Fig. 1. Experimental configuration

4. Results and discussion

Firstly, we evaluated the tracking performance of the two algorithms by introducing polarization rotations in simulation, the results are shown in Fig. 2. Note that taking the sign of the gradient effectively scales the learning parameter μ , consequently limiting its maximum value; in any case, when selecting a suitable value of μ , both algorithms can track polarization rotations $> 1\text{Mrad/s}$. Therefore, the tracking performance is not significantly

compromised when using the sign-sign algorithm. Fig. 3 shows the BER performance versus received optical power of the direct and sign-sign algorithms for both fully parallel and low power hardware implementations. We observe similar performance for all implementations tested.

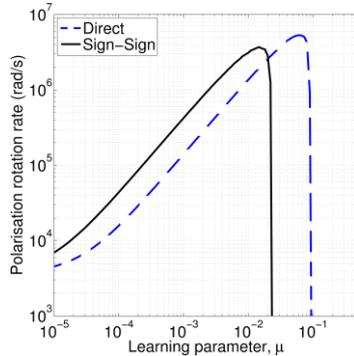


Fig. 2. Ability to track polarization rotations

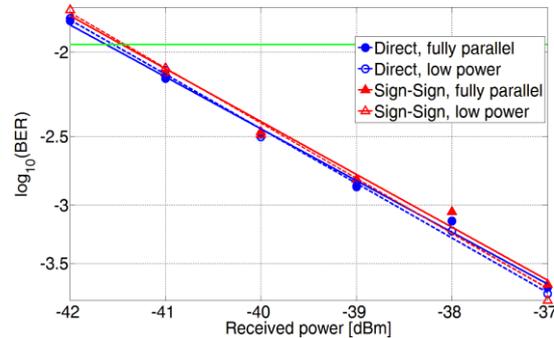


Fig. 3. Performance of the CMA equalizer architectures

Fig. 4 shows the power consumption of the CMA equalizer for the four architectures described. As expected the filter implementation, Eq. (1), maintains a constant power consumption of 32mW in all cases. The low power version of the sign-sign algorithm, being the one with the lowest complexity, is the most power efficient implementation requiring 51mW; if compared with the fully parallel direct implementation, with consumes 247mW, we observe a global reduction in power consumption of ~80%.

Regarding exclusively the tap weight update stage, as we are interested specifically in the power efficiency of the sign-sign versus the direct algorithm, we see that the power reduces from 215.2mW to 103.9mW, or 52%, for the fully parallel case, and from 25.2mW to 19mW, or 25%, for the low power version. This difference is explained since there are other tasks associated with the update, which are not optimized by the sign-sign algorithm, which must be implemented regardless of the degree of parallelism (the error term computation for example). In any case, the high efficiency gains (25% and 52%) obtained show the huge benefit of implementing the sign-sign algorithm instead of a direct implementation.

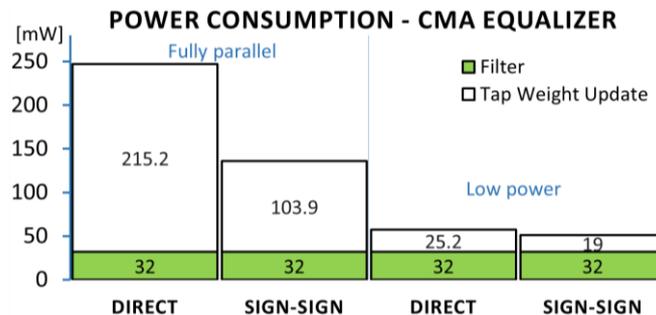


Fig. 4. Power consumption for the direct and sign-sign algorithms, fully parallel and low power versions

5. Conclusions

We synthesized a low complexity 3 tap CMA equalizer for a 10 Gb/s digital coherent receiver with only 4 bits resolution for the input signal and equalizer coefficients on a 45-nm CMOS process. A significantly reduced complexity is obtained with a multiplier-free weight update due to a sign-sign algorithm which reduces the power consumption by up to 52% without compromising performance; both the low complexity and conventional algorithms maintain equivalent receiver sensitivity and track polarization rotations $>1\text{Mrad/s}$. The sign-sign implementation constitutes a suitable architecture where low-complexity digital coherent receivers are required, such as access networks.

This work was supported by the European Commission PIANO+ project CRITICAL.

References

- [1] S. J. Savory, "Digital Coherent Optical Receivers: Algorithms and Subsystems," *IEEE J. Sel. Top. Quant. Electron.*, **16**, 1164 (2010)
- [2] A. Poularikas and Z. Ramadan, *Adaptive filtering primer with Matlab* (Taylor & Francis, 2006), Chap.8
- [3] D. Cardenas, D. Madan, S. Win, D. Lavery, and Seb Savory, "Fixed Point and Power Consumption Analysis of a Coherent Receiver for Optical Access Networks Implemented in FPGA," in *ECOC 2013 Proceedings*, London, 2013, paper Mo.3.C.4
- [4] D. Lavery, R. Maher, D. Millar, B. Thomsen, P. Bayvel and S. Savory, "Digital Coherent Receivers for Long-Reach Optical Access Networks," *J. Lightwave Technol.* **31**, 609 (2013)
- [5] T. Ahmad, Y. Ai, P. Muralidharan, "Methodology for Power-Aware Coherent Receiver Design," *Advanced Photonics Congress (SPPCom) 2013 paper SPT4D.4*