Area-Efficient Hardware Parallelization of Neural Network CD Equalizers for 4×200 Gb/s PAM4 CWDM4 Systems

Bo Liu[(1,*)], Christian Bluemm[(2)], Stefano Calabrò[(2)], Bing Li[(1)], and Ulf Schlichtmann[(1)]

[(1)] Technical University of Munich, Chair for Electronic Design Automation, Munich 80333, Germany
[(2)] Huawei Technologies Duesseldorf GmbH, Munich 80992, Germany
* Author e-mail address: bo.liu@tum.de

Abstract *We compare hardware parallelization of CD equalizers on 10 km 4x200 Gb/s IM/DD PAM4 O-band measurements. A single neural network equalizer with multiple output symbols saves 20% of reference chip area versus multiple single-symbol output variants and 77% versus Volterra non-linear equalizers. Multi-task learning enables cost-efficient scenario flexibility. ©2023 The Author(s)*

Introduction

In fibers, transmission of optical signals suffers from chromatic dispersion (CD) and non-linearity effects in intensity modulation / direct detection (IM/DD) systems[1]. Such effects lead to intersymbol interference and bit errors. Volterra non-linear equalizers (VNLEs), as shown in Fig. 1(a), are effective countermeasures. However, VNLEs suffer from huge computation complexity and numerical instability[2].

To realize equalization and demapping, neural network non-linear equalizers (NN-NLEs) are proposed[2]–[4]. In high speed transceiver implementations, digital equalizers have to operate in parallel to reach data rates with system clocks, which run at a much lower frequency. Most NN-NLEs are built by direct parallelization of single-symbol output neural network non-linear equalizers (SSO-NLEs). An SSO-NLE is illustrated in Fig. 1(b). Whereas the accuracy of SSO-NLEs is promising, their complexity remains critical for high-speed communication systems.

Multi-symbol output neural network non-linear equalizers (MSO-NLEs) are explored to replace SSO-NLEs over the last two years[5]–[7], as shown in Fig. 1(c). MSO-NLEs are effective to achieve both high performance and low complexity, because shared signal features are extracted a single time and leveraged efficiently to predict multiple consecutive symbols. Researchers also apply compression techniques such as pruning to reduce the complexity of MSO-NLEs further[6].

However, the above MSO-NLEs are trained, pruned, and fine-tuned on different problem instances separately, leading to different neural network structures and multiple sets of weights and biases. Thus, we need to design a configurable hardware implementation for MSO-NLEs with full multipliers[4] to support different parameters for different situations. Moreover, large on-chip memory is required to store the parameters for a single problem instance. To reduce the hardware cost and following our previous work[4], we use multi-task learning (MTL) to train a single MSO-NLE on datasets of multiple wavelengths jointly to share the weights and keep the biases still flexible. Flexible biases equip the NN-NLE with the adaptability for different wavelengths to achieve a low BER (bit error rate). On the other hand, the shared weights are fixed to simplify the multipliers. We also apply pruning to this approach and achieve area savings of 20% and 77% compared with an MTL-trained SSO-NLE and VNLEs, respectively.

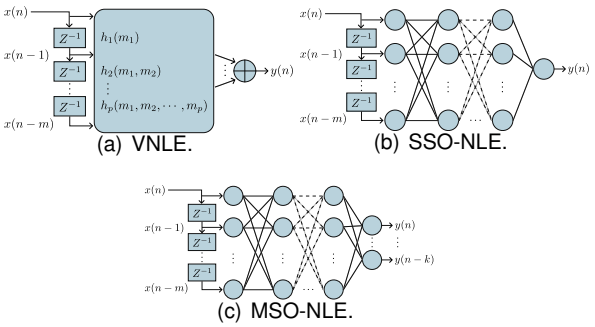


Fig. 1: Volterra non-linear equalizer and single/multiple symbol output neural network non-linear equalizers.

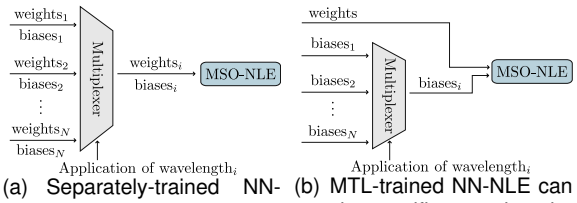
Hardware Parallelization

NN-NLEs outperform classical equalizers, but the necessity of hardware parallelization is often overlooked. In fact, hardware parallelization[8] is inevitable for high-speed transceivers because of the mismatch between the high-speed baud rate of the transmission and the low-speed clock frequency of the transceiver ASIC. For example, a typical operating clock frequency could be around $1\ \mathrm{GHz}$ or less for today's transceivers, while the

baud rate can be 112 GBd or even higher. In order to support the much higher baud rate, a straightforward approach is to instantiate multiple parallel copies of a trained equalizer to obtain multiple consecutive symbols simultaneously. The number of parallelized copies of an equalizer with 1 sample per symbol processing, denoted as N_{copies}, is expressed as follows

$$N_{copies} \geq \frac{R_{baud}}{f_{clk}} \times \frac{1}{N_{out}},\ N_{copies} \in \mathbb{Z}^{+} \qquad (1)$$

where R_{baud} is the baud rate of the transmission, f_{clk} is the operating clock frequency of the transceiver, and N_{out} is the number of predicted symbols at the output of each copy. For example, assuming $R_{baud} = 112$ GBd and $f_{clk} = 1$ GHz, we need at least 112 parallelized copies of a trained VNLE or SSO-NLE ($N_{out} = 1$), and at least 38 copies of a trained MSO-NLE (if $N_{out} = 3$).



(a) Separately-trained NN-NLE can match specific wavelengths by switching both weights and biases.

(b) MTL-trained NN-NLE can match specific wavelengths by switching merely between biases.

Fig. 2: Separate learning and multi-task learning.

Multi-Symbol Output Neural Network Non-Linear Equalizer

MSO-NLEs[5]–[7], which are designed to output multiple consecutive symbols per copy, proved to be effective to achieve both high performance and low complexity compared with SSO-NLEs and VNLEs. The information required to predict multiple symbols is already contained in the input, so predicting multiple symbols will be more efficient than predicting only a single symbol. Moreover, compression techniques are applied to reduce the complexity of MSO-NLEs further[6].

However, according to the conventional approaches, such MSO-NLEs are trained, pruned, and fine-tuned on datasets of different wavelengths separately as shown in Fig. 2(a), leading to multiple sets of weights and biases. When the equalization target for an MSO-NLE changes, a specific set of weights and biases should be determined and loaded into the hardware. Thus, full multipliers[4] are required to enable different parameters for different wavelengths during VLSI implementation. In addition, large on-chip memory is required to store the parameters for a specific wavelength application. Moreover, although some weights of MSO-NLEs are pruned during separate learning, the corresponding multipliers are difficult to be eliminated unless those weights are pruned in the same position of the neural network structure for all involved problem instances.

Multi-task Learning for Multi-symbol Output Neural Network Non-linear Equalizer

To solve the above problems, we use MTL to train each MSO-NLE copy on datasets of multiple wavelengths jointly as shown in Fig. 2(b), which allows freezing the weights while keeping the biases still flexible. Flexible biases provide the MSO-NLE with the adaptability for different wavelengths. On the other hand, frozen weights can be used to simplify the multipliers to reduce their area, leading to a smaller hardware cost. Moreover, frozen weights are fixed, so on-chip memory to store the weights can be reduced. Unlike separate learning, when some weights are pruned during multi-task learning, the corresponding multipliers can be really removed because there is only one set of weights for all involved wavelengths.

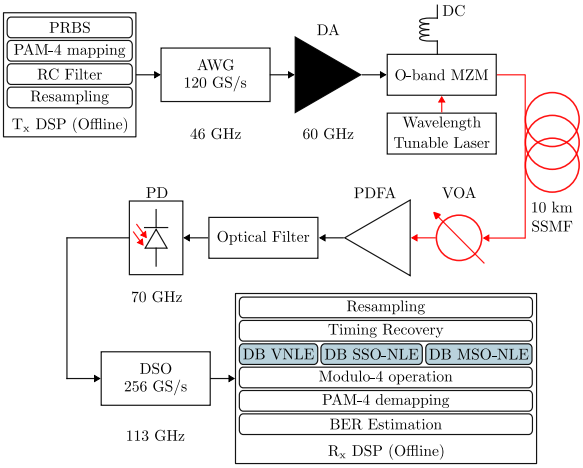


Fig. 3: Experimental 10 km setup with offline DSP.

Experimental Setup

The IM/DD measurement setup with offline DSP for 10 km PAM4 transmission with 112 GBd per lane is shown in Fig. 3. Below each electrical/optical component, the 3 dB bandwidths are written. At the transmitter (Tx), after duobinary precoding[3], pseudorandom binary sequences (PRBS) are Gray-mapped to PAM-4 symbols. A raised cosine (RC) filter with roll-off factor of 0.14 implements pulse shaping. After resampling, a 120 GS/s arbitrary waveform generator (AWG) is used to convert the digital samples to analog signals which are amplified by a 60 GHz driver amplifier (DA) towards an O-band Mach Zehnder modulator (MZM). While standard O-band CWDM4 wavelengths 1270 nm, 1290 nm, 1310 nm, and

Tab. 1: Designs and experimental results of DB VNLEs, DB SSO-NLEs, and DB MSO-NLEs.

Notation	Design*	Trained	Pruned	Copy†	Area (μm^2)‡
DB VNLE	[21,9,7]	Separately	0.0%	112	1.66E+7
DB SSO-NLE	21|11|7|1	Separately	11.43% $\sim$ 26.35%	112	1.29E+7
DB MSO-NLE	23|25|13|3	Separately	15.75% $\sim$ 35.24%	38	1.29E+7
DB SSO-NLE	21|11|7|1	MTL	0.32%	112	4.80E+6
DB MSO-NLE	23|25|13|3	MTL	24.63%	38	3.82E+6

VNLEs, SSO-NLEs, and MSO-NLEs are all trained on a duobinary (DB) target[3].
* The design for DB VNLEs indicates the number of memory taps of [1^{st}, 2^{nd}, 3^{rd}] order.
The designs for DB SSO-NLEs and DB MSO-NLEs indicate the number of neurons in each layer.
The activation function of 1^{st}, 2^{nd}, 3^{rd}, and 4^{th} layer is none, $\tanh$, $\tanh$, and linear, respectively (1^{st} layer is the input). $\tanh$ is implemented using $\mathrm{H\text{-}\tanh}$, a low-cost variant of $\tanh$[3].
† The number of parallelized copies of an equalizer required in the hardware parallelization architecture.
‡ Area evaluation is conducted using Design Compiler[9] for logic synthesis using 45nm process technology.

1330 nm are focused, further captures at in-between wavelengths allow for exploring the performance/wavelength relationship. After 10 km standard single mode fiber (SSMF) transmission, the received optical power (ROP) at the input of a praseodymium-doped fiber amplifier (PDFA) is controlled by a variable optical attenuator (VOA) at 7 dBm. An optical filter suppresses the broadband noise of the PDFA. The filtered optical signal is input to a photodiode (PD). The electrical output of the PD is digitized by a 256 GS/s digital oscilloscope. At the receiver (Rx), timing recovery operates at 2 samples per symbol (sps) first, and then the output signals are downsampled to 1 sps for equalization. Modulo-4 operation[3] and PAM4 slicing are applied before BER estimation.

We train equalizers on a duobinary (DB) target[3]. The designs of DB VNLEs and DB SSO-NLEs follow our previous work[4]. DB SSO-NLEs and DB MSO-NLEs are trained with 500 epochs to reach a low mean squared error. We use 182k PAM4 symbols for training and other 45k symbols for inference. DB SSO-NLEs and DB MSO-NLEs are first trained separately and jointly on datasets of multiple wavelengths respectively, and then pruned to reduce complexity and fine-tuned to mitigate BER degradation. Magnitude-based pruning is used to zero the weights whose absolute values are smaller than a threshold. DB VNLEs are only separately-trained without pruning. Uniform quantization is adopted. Multipliers have 8-bit input and 16-bit output. Inputs with larger bit-width are truncated to the 8 most significant bits (MSB). The bit-width of adders increases level by level to keep the carry bit.

Experimental Results

Fig. 4 and Tab. 1 show the BER, hardware area, and percentage of pruned weights comparison between separately-trained DB VNLEs, separately/MTL-trained DB SSO-NLEs, and separately/MTL-trained DB MSO-NLEs.

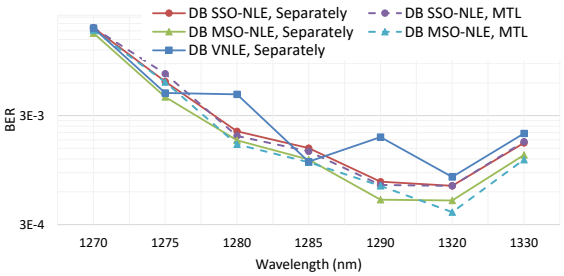


Fig. 4: BER comparison.

The percentage of pruned weights depends on the specific wavelength when we train and prune DB SSO-NLEs and DB MSO-NLEs on datasets of different wavelength separately. Multipliers in separately-trained DB SSO-NLEs and DB MSO-NLEs are hard to be eliminated to reduce the area, because very few pruned weights are in the same position of the neural network structure among all wavelengths. Although the areas of separately-trained DB SSO-NLEs and DB MSO-NLEs in Tab. 1 are similar, separately-trained DB MSO-NLEs achieve lower BER.

Pruned weights of the MTL-trained DB SSO-NLE and DB MSO-NLE can be used to eliminate multipliers and reduce the hardware area. To maintain the BER, only 0.32% weights are pruned in the MTL-trained DB SSO-NLE. But 24.63% weights of the MTL-trained DB MSO-NLE can be pruned and its BER is still better than the MTL-trained DB SSO-NLE. The MTL-trained DB MSO-NLE has area savings of 20% and 77% compared with the MTL-trained DB SSO-NLE and DB VNLEs, respectively.

Conclusions

We consider parallelization of CD equalizers. We use MTL to train an MSO-NLE on datasets of multiple wavelengths jointly. By sharing a single set of weights among all involved wavelengths, keeping biases reconfigurable, and applying compression techniques, we achieve area savings of 20% and 77% compared with an MTL-trained SSO-NLE and VNLEs, respectively.

Acknowledgment

This work is funded in part by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - Project-ID 497488621.

References

[1] J. C. Cartledge, "Volterra Equalization for Nonlinearities in Optical Fiber Communications", in *Signal Processing in Photonic Communications (SPPCom)*, 2017.

[2] C. Bluemm, M. Schaedler, S. Calabrò, *et al.*, "Equalizing Nonlinearities with Memory Effects: Volterra Series vs. Deep Neural Networks", in *European Conference on Optical Communication (ECOC)*, 2019.

[3] C. Bluemm, B. Liu, B. Li, *et al.*, "800Gb/s PAM4 Transmission over 10km SSMF Enabled by Low-Complex Duobinary Neural Network Equalization", in *European Conference on Optical Communication (ECOC)*, 2022.

[4] B. Liu, C. Bluemm, S. Calabrò, B. Li, and U. Schlichtmann, "Area-Efficient Neural Network CD Equalizer for 4$\times$200Gb/s PAM4 CWDM4 Systems", in *Optical Fiber Communication Conference (OFC)*, 2023.

[5] Z. Xu, S. Dong, J. H. Manton, and W. Shieh, "Low-Complexity Multi-Task Learning Aided Neural Networks for Equalization in Short-Reach Optical Interconnects", *Journal of Lightwave Technology (JLT)*, vol. 40, no. 1, pp. 45–54, 2021.

[6] B. Sang, W. Zhou, Y. Tan, *et al.*, "Low Complexity Neural Network Equalization Based on Multi-Symbol Output Technique for 200+ Gbps IM/DD Short Reach Optical System", *Journal of Lightwave Technology (JLT)*, vol. 40, no. 9, pp. 2890–2900, 2022.

[7] Q. Bian, J. Jia, Z. Li, J. Shi, N. Chi, and J. Zhang, "Low-Complexity Multi-Symbol Output Complex-Valued Neural Network for Nonlinear Equalization in 100G Coherent Photonic-Assisted W-Band Fiber-Wireless Integrated Communication", in *European Conference on Optical Communication (ECOC)*, 2022.

[8] S. Srivallapanondh, P. J. Freire, B. Spinnler, *et al.*, "Knowledge Distillation Applied to Optical Channel Equalization: Solving the Parallelization Problem of Recurrent Connection", in *Optical Fiber Communication Conference (OFC)*, 2023.

[9] P. Kurup and T. Abbasi, *Logic Synthesis Using Synopsys®*. Springer Science & Business Media, 2012.