POST-DIGITAL - European Training Network on Post-Digital Computing [GA860360]

Document Details

Title	D2.3 Report on the evaluation of the photonic- and electronic- based analogue computing implementations
Deliverable number	D2.3
Deliverable Type	Report (public)
Deliverable title	Report on the evaluation of the photonic- and electronic- based analogue computing implementations
Work Package	WP2 – Implementation and characterization
Description	Document describing the evaluation of the different approaches to implement analogue computing in the POST-DIGITAL European Training Network and the level of progress of the ESRs
Deliverable due date	30/09/2023
Actual date of submission	29/09/2023
Lead beneficiary	CSIC
Version number	V1.0
Status	Final

Dissemination level

PU	Public	X
CO	Confidential, only for members of the consortium (including Commission Services	

Project Details

Grant Agreement	860360
Project Acronym	POST-DIGITAL
Project Title	POST-DIGITAL - European Training Network on Post-Digital Computing
Call Identifier	H2020-MSCA-ITN-2019
Project Website	https://postdigital.astonphotonics.uk/
Start of the Project	1 April 2020
Project Duration	48 months

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 860360.

EXECUTIVE SUMMARY

This deliverable of the POST-DIGITAL project describes the evaluation of the analogue computing devices developed in the consortium. The degree of evolution of the different hardware approaches implemented by the POST-DIGITAL partners is remarkable. Using photonic substrates, designs based on spatial, temporal or frequency multiplexing have been developed, characterised and evaluated. In addition, this report includes a detailed account of the progress made in establishing general rules for benchmarking neuromorphic hardware. The progress of the ESRs is also documented in this deliverable.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ACRONYMS

AWG	Arbitrary Wave Generator
CMA-ES	Covariance Matrix Adaptation Evolution Strategy
DMD	Digital Micromirror Device
DRC	Deep Reservoir Computing
ELM	Extreme Learning Machine
ES	Electric Switch
ESR	Early Stage Researcher
ETN	European Training Network
FM	Frequency Multiplexing
ISI	Inter-Spike Interval

LIF	Leaky-Integrate-and-Fire
MMF	Multi-Mode Fiber
MZM	Mach-Zehnder Modulator
NIR	Neuromorphic Intermediate Representation
NMSE	Normalized Mean Square Error
NN	Neural Network
ODE	Ordinary Differential Equation
PD	PhotoDiode
PEPG	Parameter Exploring Policy Gradient
PNN	Photonic Neural Network
POST-DIGITAL	Project 'POST-DIGITAL - European Training Network on Post-Digital Computing' EC GA 860830
PSF	Programmable Spectral Filter
RC	Reservoir Computing
SLM	Spatial Light Modulator
SNN	Spiking Neural Network
SOA	Semiconductor Optical Amplifier
VCSEL	Vertical-Cavity Surface-Emitting Laser

1 INTRODUCTION

The ESRs of the POST-DIGITAL ETN are investigating different hardware platforms and information processing strategies in the context of analogue computing. In the photonic domain, several approaches based on spatial, temporal or frequency multiplexing techniques have been successfully implemented and evaluated. The ESRs have also contributed to the field of neuromorphic computing by developing software tools to benchmark and evaluate different hardware platforms using a common language.

The initial designs reported in Deliverable 2.1 were characterised in Deliverable 2.2. The current deliverable focuses on the evaluation of the hardware approaches considered promising by the POST-DIGITAL partners. The ESRs have added the ability to benchmark analogue computing systems to their skill set. This report shows that the ESRs are at the forefront of post-digital computing.

List of contributors (alphabetical order): Steven Abreu (ESR1, Groningen), Mirko Goldmann (ESR5, CSIC), Alessandro Lupo (ESR12, ULB), Enrico Picco (ESR4, ULB), Anas Skalli (ESR3, Femto-ST), Lucas Talandier (ESR14, CSIC). Document compiled by Miguel C. Soriano, IFISC (CSIC-UIB).

2 Evaluation of Photonic- and Electronic- based Analogue Computing Devices

Photonic computing is increasingly reaching a high level of maturity, with competitive accuracies in machine learning benchmark tasks. In Section 2.1, we report on the evaluation of different implementations developed in the POST-DIGITAL consortium. In Section 2.2, we present a software approach to evaluate different neuromorphic computing platforms on a common basis. While the validation of this software has been performed on digital electronic systems, the transfer to analogue electronic systems is underway.

2.1 Photonic Systems

As described in Deliverables 2.1 and 2.2, the different photonic designs follow complementary strategies that exploit multiplexing either in space (Section 2.1.1), in time (Section 2.1.2) or in frequency (Section 2.1.3).

This deliverable covers the evaluation of a spatial multiplexing approach based on a large aperture laser, carried out by ESR3 and ESR5. In turn, ESR4 and ESR14 have made progress in the evaluation of two complementary temporal multiplexing information processing systems. Finally, ESR12 has evaluated the performance of a photonic system using frequency multiplexing for reservoir computing and extreme learning machines.

2.1.1 Space multiplexing

This section focuses on the evaluation of an artificial neural network-like system implemented in optics. The optical system is based on a large area vertical cavity surface emitting laser (LA-VCSEL), as it can be seen later in Figure 1. The design and characterization of this spatially multiplexed approach has been presented in previous deliverables. A performance evaluation is reported here. In particular, ESR3 worked in close collaboration with ESR5 to improve the performance of this photonic neural network (PNN) through a detailed numerical and experimental analysis.

The ESRs first performed a ceiling analysis on a digitally simulated neural network (NN) of a representative size (100 neurons) to identify which parts of the NN most improve the system's performance when optimized in an image classification task. The well-known MNIST dataset was used to perform this analysis, as it can later be used in the PNN for comparison purposes. The main results of the ceiling analysis are summarized in Table 1. Restricting the output weights $\mathbf{W}^{out}$ of the neural network to be positive only, while training it and keeping the input weights random, yields the worst performance with a classification accuracy of 60%. Allowing $\mathbf{W}^{out}$ to take negative values dramatically improves the performance to 87%. Furthermore, training the input weights $\mathbf{W}^{in}$ in addition to the output weights yields another significant jump in performance to 97% accuracy.

Table 1. Numerical evaluation of the influence of the weight training on the system performance.

System	Performance
NN 100 training $\mathbf{W}^{out}$ positive only	60%
NN 100 training $\mathbf{W}^{out}$ positive and negative	87%
NN 100 training $\mathbf{W}^{out}$ and $\mathbf{W}^{in}$	97%

These conclusions may seem trivial in the context of conventional machine learning, but they have profound consequences when it comes to building a setup that uses a physical system, i.e., the LA-VCSEL, as the central piece of an optical neural network. First, while negative weights are critical for accuracy, they are not trivial to implement optically. Second, implementing trainable input weights is challenging because one cannot rely on error backpropagation to train them, as is done in digitally simulated neural networks. Therefore, a model-free or "black-box" optimization algorithm that is sufficiently efficient is required. Here, we use an evolutionary strategy that updates the weights based on multiple samples of the network's classification error within a certain (potentially small) variation of the weights. In particular, we use the Parameter Exploring Policy Gradient (PEPG) algorithm for the following reasons:

1. The algorithm is model-free and therefore does not require any knowledge of the PNN to train the input weights.
2. The PNN with its high throughput (15 kHz) allows fast sampling of the error for different weights.
3. Updating the weights requires minimal computation and scales linearly with the number of parameters.
4. The algorithm works even with limited resolution of the weights to be trained.

With these considerations in mind, the ESRs designed an improved version of the setup in which all weights in the PNN can be trained, resulting in a highly tunable network that can begin to handle real data sets and move beyond the basic proof of concept.

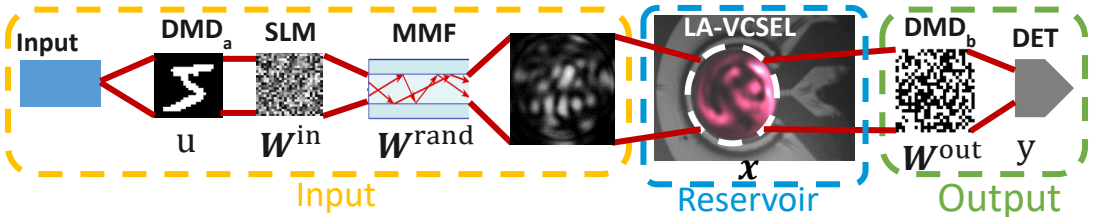


Figure 1 Working principle of the VCSEL-based experimental photonic neural network.

The updated experimental setup looks as follows: Input images u displayed on a digital micromirror device (DMDa) are passed through a phase mask displayed on a spatial light modulator (SLM), which encodes the trainable input weights $\mathbf{W}^{in}$. The phase modulated input is injected onto the LA-VCSEL through a multimode fiber (MMF) which passively implements a random linear mixing $\mathbf{W}^{rand}$. The VCSEL then transforms the injected information non-linearly yielding the perturbed mode profile x yielding up to 350 fully parallel neurons. The final part of the PNN is its output layer. The VCSEL's surface is imaged onto DMDb, whose pixels can flip between two positions, for one of which it reflects light onto the photodetector (DET), giving us Boolean output weights $\mathbf{W}^{out}$. Here, up to 5 times optical magnification of the LA-VCSEL is implemented to increase the imaged size of the LA-VCSEL on the DMDb and accordingly the amount of available Boolean weights is increased as well. Finally, the output of the PNN y is the optical power detected at DET. Negative output weights are achieved via recording the output of the PNN twice and implementing an electronic subtraction. As described above, $\mathbf{W}^{out}$ and $\mathbf{W}^{in}$ are trained via the PEPG evolutionary optimization based on multiple samplings of the error using slightly perturbed weights. Ideally, a second SLM should be used at the output to provide higher resolution and to increase classification accuracy further. The described experimental scheme is shown in Fig. 1.

Figure 2 shows preliminary classification performance for the MNIST task using Boolean weights (BOOL), and trinary weights (3 values; -1, 0, +1). Trinary weights have a significant positive impact on performance, reaching an average performance of 90%. In addition, a long-term stability analysis was conducted, over a period of 10 hours, showing little to no degradation in performance or drift. Moreover, the average cross correlation between different outputs over the 10 hours was 98%.

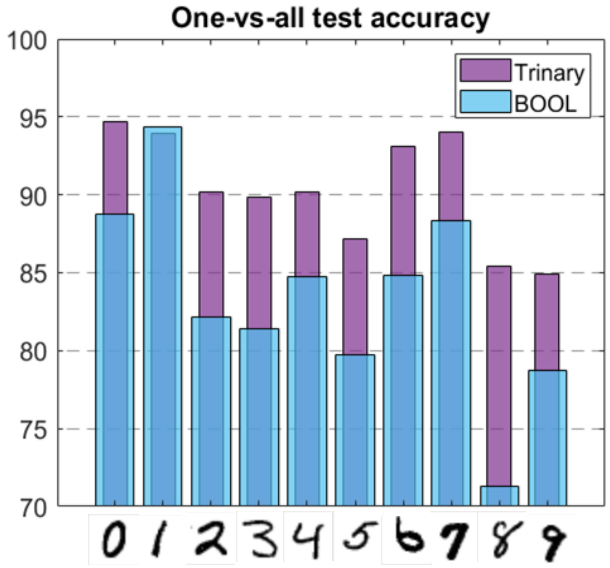


Figure 2 Experimental evaluation of the photonic neural network for the MNIST dataset.

In summary, the VCSEL-based PNN shows promising initial results on the MNIST dataset, while achieving classification at a bandwidth of 15 kHz. This approach is highly relevant, fully parallel and scalable both in terms of network size and depth, as it provides a clear way to use VCSELs in a deep PNN configuration. Finally, the inference bandwidth is also highly scalable due to the fast VCSEL response time without a significant increase in power consumption. ESR3 and ESR5 will continue their collaboration with the goal of fully exploiting the computational capabilities of the setup by training the input weights of the hardware.

2.1.2 Time multiplexing

In this section, the evaluation of delay-based photonic and opto-electronic approaches for reservoir computing and extreme learning machines is reported. This entails the work of ESR4 and ESR14, respectively.

2.1.2.1 Delay-based reservoir computing based on a semiconductor laser subject to optical feedback

ESR14, in collaboration with other members of the Nonlinear Photonics team at IFISC, has implemented and evaluated an experimental phase-encoded photonic time-delay reservoir computer. The physical implementation was guided by the results of numerical simulations, which consistently showed a significant performance improvement for various tasks when phase encoding is incorporated. This improvement was subsequently confirmed by the experimental evaluation.

The experimental apparatus was an advanced version of a previously used setup, now augmented with a phase modulator (Figure 3). The specific timing of the input modulation in the experiment is as follows:

- The signal underwent a modulation at a frequency of 17.0667GHz, serving as the masking rate.

- The node separation time of the virtual nodes was recorded at 58.6ps.

The feedback loop had a duration of 24.5ns, representing the system's delay

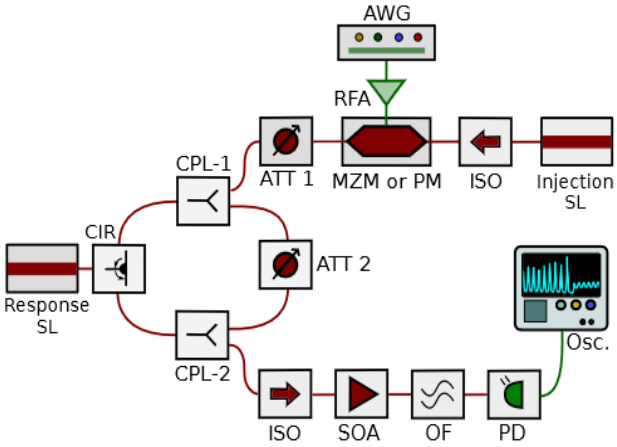


Figure 3 Experimental photonic setup. ISO: optical isolator, RFA: RF amplifier, MZM: Mach-Zehnder intensity modulator, PM: Phase modulator, AWG: arbitrary waveform generator, ATT: optical attenuator, CIR: optical circulator, SOA: semiconductor optical amplifier, OF: optical filter, PD: photoreceiver, Osc: Real-time oscilloscope

Two primary objectives were pursued for the evaluation of this setup:

1. **One-Step Ahead Chaotic Prediction**: The main goal was to demonstrate that the use of phase modulation alone can improve performance on a one-step-ahead chaotic prediction task compared to intensity modulation alone. The conclusive results of the experiment corroborated the numerical simulations, showing a halving of the NMSE when using phase as opposed to intensity.
2. **Signal Recovery Post Fiber Impairments**: The second objective was to test another hypothesis derived from the numerical simulations. This hypothesis was that using the full information of the electric field (both amplitude and phase) would aid signal recovery. The experiment involved transmitting a PAM4-modulated signal over 50 km of single-mode fibre at a high launch power of 10 dBm. To test this, both phase and intensity modulation were modulated simultaneously. The intensity modulation was performed with the received amplitude, delayed and masked, and the phase modulation was performed with the delayed received phase. The result of this experiment showed an improvement of almost an order of magnitude by parameter optimisation.

2.1.2.2 Opto-electronic reservoir computing based on a delay loop

The experimental system reported by ESR4 in this section is a time-delayed reservoir computer [1], implemented with an optoelectronic system similar to [2]. The design and characterization of this approach has been presented in previous deliverables, see Fig. 4 for a schematic of the experimental setup. The analog reservoir states are represented by the light intensity in an optical fiber, while digital electronic hardware is used to interface the optical signal. The speed bottleneck in these optoelectronic systems is often the slower electronics, which affects the reservoir computer especially when processing complex data structures such as images or video. For this reason, ESR4 has improved the electronic interface and increased the speed of the entire system by two orders of magnitude. This makes it possible to achieve real-time data processing even when studying large data sets or more complex RC architectures. The evaluation of the system on two practical data sets is reported here.

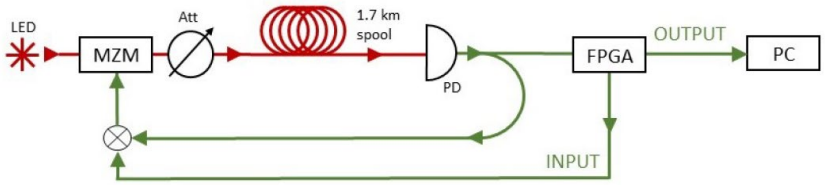
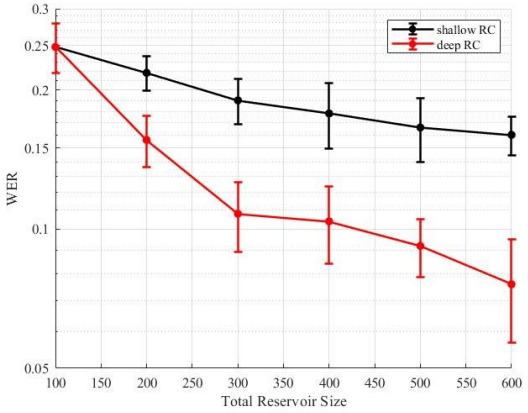


Figure 4 Scheme of the experimental setup. The optical and electronic components are shown in red and green, respectively. The reservoir layer consists of an incoherent light source, a Mach-Zehnder intensity modulator (MZM), an optical attenuator (Att), an approximately 1.7-km fibre spool, a feedback photodiode (PD) and a resistive combiner.

First, ESR4 and collaborators tested the optoelectronic setup on human action recognition in videos [3]. This dataset consists of recordings of 25 actors performing 6 different actions (walking, jogging, running, boxing, hand waving, and hand clapping). The results are shown in Fig. 5(a). With 600 nodes, a classification accuracy comparable to state-of-the-art algorithms with network sizes ranging from thousands to hundreds of millions of neurons is reported. At the same time, the processing speed of the system exceeds the typical video frame rate of 25 fps by a factor of 6, providing a platform for real-time multi-channel video processing. A more comprehensive comparison of this work with other machine learning platforms, including a full list of references, can be found in [3].

Second, ESR4 evaluated the performance of the setup on human speech processing, the spoken digit recognition task. In addition to standard RC, ESR4 tested another architecture, called Deep Reservoir Computing (DRC), in which different reservoirs are stacked in series to increase the richness of the reservoir dynamics. The results [4], shown in Fig. 5(b), experimentally confirm the superiority of DRC over standard RC. Moreover, the system can classify speech in real time even when multiple reservoir layers are concatenated in series. Similar results on DRC have been obtained by the ESR4 group using a photonic reservoir computer based on frequency multiplexing [5].

Publication	Classifier	Network Size	Accuracy
Khan (2021) [5]	Deep CNN	~ 195 millions	98.3%
Rathor (2022) [6]	DNN	~ 5000	93.9%
Ramya (2020) [7]	NN	4310	91.4%
This work	**Photonic RC**	**600**	**90.83%**
Liu (2022) [8]	RC + SNN + EA	2000	86.3%



(a) (b)

Figure 5 Experimental results for the optoelectronic reservoir computer. Figure (a) shows classification accuracy on the human action recognition task, comparing ESR4 results to some other previous work on the KTH dataset. Figure (b) shows the Word Error Rate (WER) in the classification of spoken digits when noise is present in the input audio signals (SNR = 3 dB), using both standard, "shallow", RC (black) and deep RC (red). The deep RC consists of 2 to 6 reservoirs of size 100 used sequentially.

Overall, this time-multiplexed optoelectronic system is an important step towards the development of real-time signal processing with photonic neuromorphic hardware.

2.1.3 Frequency multiplexing

ESR12's work focuses on frequency-multiplexing schemes for optical neuromorphic computing, including both frequency-multiplexing reservoir computers (FM-RC) and frequency-multiplexing extreme learning machines (FM-ELM). Both schemes have been successfully demonstrated in fiber-based experimental setups [6, 7]. The working principle of these computers consists of encoding different neuron signals in the lines of a frequency comb. Neuron signals are mixed by applying a periodic phase modulation to the radiation, which induces frequency-domain interference between comb lines.

One of the strengths of frequency-multiplexing is the high parallelization capability, which allows the propagation of multiple signals on the same channel by employing different wavelengths. For example, parallel computations are achievable by employing multiple non-interfering frequency combs. This concept has been demonstrated both for FM-ELMs and FM-RCs in fiber-based experiments [8, 5]. Parallel RCs running on the same substrate can execute independent tasks or be combined to form deep reservoir computer schemes (deep-RC).

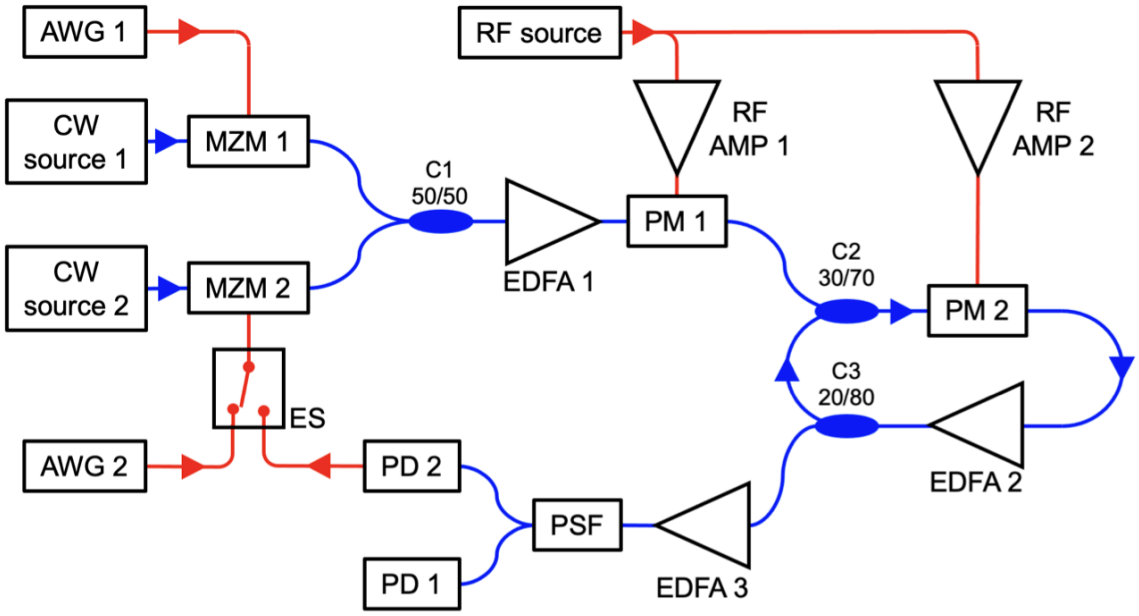


Figure 6 Experimental setup for the frequency-multiplexing deep-RC. Optical connections are in blue, electrical connections in red. MZM: Lithium Niobate Mach-Zehnder modulator; AWG: arbitrary waveform generator; C: fiber couplers; EDFA: Erbium-doped-fiber amplifier; PM: phase modulator; RF source: radio frequency source at frequency Ω; RF AMP: radio frequency amplifiers; PSF: programmable spectral filter; PD: photodiode; ES: electric switch.

In [5] ESR12 and collaborators reported the benchmark of a frequency-multiplexing (deep-)RC based on the experimental scheme represented in Figure 6. Two frequency combs propagate in the setup, generated by applying a radiofrequency (20 GHz) periodic phase modulation (PM 1) to two C-band laser sources (CW source 1 and 2). Both light sources are modulated by a Mach-Zehnder modulator (MZM 1 and 2). MZM 1 is driven by an arbitrary waveform generator (AWG 1), while MZM 2 can be driven by AWG 2 or by a signal proportional to the power in the first comb, depending on the experimental configuration (position of the electric switch ES). In the first configuration (AWG 2 driving MZM 2), two independent signals propagate in the setup without interaction, realizing two parallel computations; in the second configuration (PD 2 driving MZM 2), the second frequency comb is modulated by a signal proportional to the power in the first frequency comb, realizing a deep-RC. Independently of the configuration, both frequency combs propagate without interacting in a fiber loop, experiencing phase modulation (PM 2) for each roundtrip. Frequency comb lines

encode neuron signals, which are mixed by the phase modulation. Only the neuron signals from the same comb are mixed, without crosstalk between frequency combs. For each roundtrip, a part of the radiation is extracted from the loop, entering a readout circuit. The readout circuit is based on a two-channel programmable spectral filter (PSF) with two roles. First, independently of the position of the switch ES, the first channel of PSF is employed with the photodiode PD 1 to measure reservoir states. Second, when the switch ES is positioned in the deep-RC configuration, the second channel of PSF applies an attenuation mask to the comb encoding the state of the first reservoir, which is then measured by PD 2 and employed as driving signal for MZM 2, thus driving the second reservoir. In our configuration, each reservoir has 20 neurons.

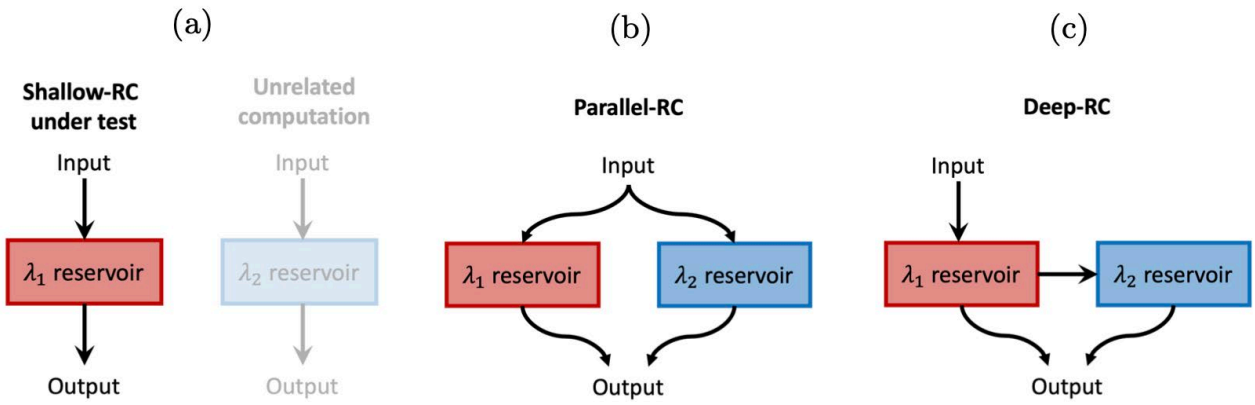


Figure 7 The three tested configurations for the deep-RC system.

We tested the system in the three configurations, as reported in Figure 7:

(a) shallow-RC, consisting of executing a traditional RC on a single frequency comb, while unrelated computation is performed on the other comb.
(b) parallel-RC, consisting of executing the same computation on two independent RCs (receiving the same input signal) and connecting the output layers.
(c) deep-RC, consisting of driving the first RC with the input signal and the second RC with a signal proportional to the power in the first comb.

When testing the deep-RC configuration, we experimented with two methods to tune the attenuation mask applied by PSF to the first frequency comb before it is transmitted to PD 2: as first method we applied a uniform attenuation mask, sweeping the attenuation α in search for the best performance, as second method we employed an evolutive optimization algorithm (CMA-ES).

Two benchmark tasks have been executed: a nonlinear channel equalization and a chaotic time-series prediction. Results are reported in Figure 8 and more detailed in [5]. First, we note that, as expected, deep-RC performs better than parallel-RC, which, in turn, performs better than shallow-RC. This proves that increasing the number of neurons by creating a deep configuration performs better than increasing the number of neurons by merging two reservoirs in a parallel fashion. Second, concerning the optimization of the connection mask between the two deep-RC layers, we note that the two tested techniques perform comparably, although one (sweeping α) is sensibly simpler than the other (CMA-ES). We identified two reasons for this behavior. First, the CMA-ES algorithm could get stuck in local minima. Second, the search of the optimal set of weights could be affected negatively by slow drifts in the operating conditions of the deep-RC.

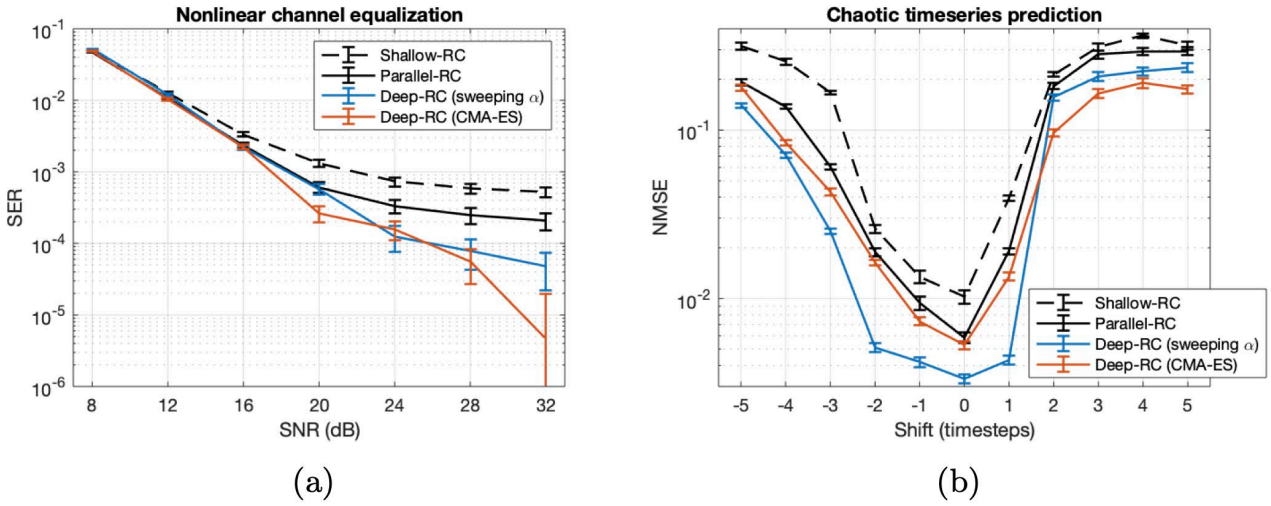


Figure 8 Results of the two benchmarks on the four tested operating conditions. SER: symbol error rate; SNR signal-to-noise ratio; NMSE: normalized mean square error. In the chaotic timeseries prediction task positive shifts consist of predicting the future, negative shifts consist of remembering the past, and zero shift consists of reproducing the current input.

In summary, developing deep architectures for neuromorphic photonic computing is a highly promising avenue for increasing the system performance. Here we have demonstrated a scheme where the connection between layers is performed in the analog domain and does not require analog-to-digital or digital-to-analog converters.

2.2 Software for the Evaluation of Neuromorphic Computing Platforms

This year, ESR 1 participated in the yearly 3-week Telluride workshop on neuromorphic computing, where he has started a new collaborative project that contributes both to formal ideas and the evaluation of neuromorphic computing implementations. The goal of ESR1 and collaborators is to develop an intermediate representation for neuromorphic computing platforms, namely the neuromorphic intermediate representation (NIR). Such an intermediate representation will allow easier conversion between different neuromorphic platforms, and thus also easier and more comprehensive evaluation and benchmarking.

Currently, NIR supports digital electronic neuromorphic platforms, as the collaborators of ESR 1 on this project are mainly working on these platforms. NIR already supports two of the largest digital electronic neuromorphic platforms: the SpiNNaker2 platform and Intel's Loihi 2 chip. Furthermore, NIR also provides support for the digital hardware platforms from SynSense, namely the Xylo chip for auditory processing and the Speck chip for vision processing. Additionally, NIR supports many of the most popular spiking neural network (SNN) simulators and training frameworks, including snnTorch, Norse, lava-dl, Nengo, Rockpool, and Sinabs.

NIR is an open-source project that is freely available on GitHub via: https://github.com/neuromorphs/NIR.

Technical description

Technically, NIR uses a dataflow graph to represent computations in an abstract and hardware-agnostic way. The dataflow graph consists of compute nodes, which are described by ordinary differential equations (ODEs) or discrete input-output functions. These compute nodes are connected through edges which represent data transfer without any transformation. All nodes can be vectorized, such that their input is not scalar but tensors of arbitrary shape. This enables NIR to compactly represent large neural networks with commonly used architectures. Table 2 shows a list of currently supported NIR primitives.

Table 2: NIR primitives with corresponding parameters and computational models. Note that the input is denoted by x and the output is denoted by y.

Node	Parameters	Computational model
Input	Input shape	-
Output	Output shape	-
Affine	W, b	$y = Wx + b$
Delay	τ	$y = x(t - \tau)$
Integrator	R	$\dot{x} = Rx$
Leaky integrator	τ, R	$\tau\dot{y} = -y + Rx$
Threshold	θ	$y = 1\ if\ x > \theta$ $y = 0\ otherwise$
Scale	s	$y = sx$

Some primitives, especially those representing neurons, are represented in differential form. The reason for this is twofold: firstly, to make NIR compatible with analog computing implementations where hardware is modelled using continuous-time ODEs, and secondly, to model computations in digital hardware on a coarse-grained level where details of the discretization procedure are abstracted away. It can be argued that digital neuromorphic hardware tries to approximate the real, continuous-time equations of spiking neuron models and thus it is justified to use these equations in our intermediate representation.

It is further noted that computational primitives are composable. As such, it is possible to use the "Leaky Integrator" and "Threshold" nodes to create a "Leaky integrate-and-fire" neuron model, see Figure 9.

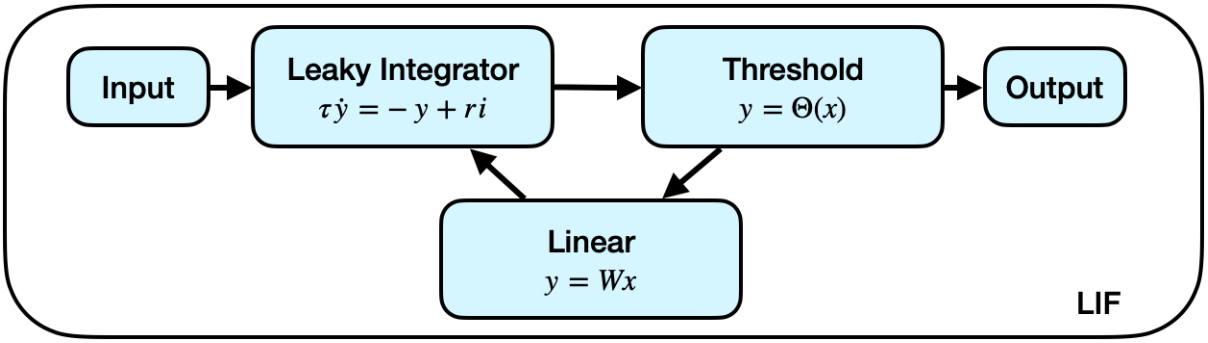


Figure 9: Schematic of how to compose new NIR primitives from existing ones. Here, a LIF neuron model is created from the primitives shown in Table 2.

Academic output

Together with another researcher from KTH Stockholm, ESR 1 is leading the effort of a journal paper that is to be submitted later this year. In this paper, the goal is to present NIR to the digital neuromorphic hardware community and demonstrate convincingly that NIR provides an unprecedented level of interoperability between neuromorphic simulators, design tools, and digital hardware platforms - while also arguing for the generality of NIR and paving the way for future work on expanding NIR.

The interoperability of NIR with respect to neuromorphic hardware and simulators is demonstrated with three exemplary NIR graphs that are run across different hardware and software platforms using NIR as a conversion tool between platforms. These NIR graphs include:

1. A single leaky-integrate-and-fire (LIF) neuron, whose dynamics are shown to be equivalent (subject to some small error tolerance) across platforms with respect to the neurons' firing rates and inter-spike interval (ISI).
2. A reservoir-like recurrent neural network of LIF neurons that is used to solve an auditory or tactile perception classification task (either the Spiking Heidelberg Digit task, or the Braille letter reading task). It is shown that the performance is equivalent, in terms of task classification accuracy as well as the network's prediction confusion matrix.
3. A convolutional neural network of Integrate-and-Fire (I&F) neurons that is used to solve a video gesture recognition task (the IBM gesture recognition dataset using an event-based camera). Similar to point 2 above), it is shown that the performance is equivalent, in terms of task classification accuracy as well as the network's prediction confusion matrix.

Once this initial paper on NIR is published, ESR 1 will expand the effort of this intermediate representation to analog and photonic platforms. A first priority is to make NIR useful to collaborators within the Post-Digital consortium. Thereafter, collaborations in the wider neuromorphic and physical computing community will be sought. Some such collaborations have already been initiated when ESR 1 presented NIR at the "frontiers of neuromorphic computing" workshop at the Max Planck Institute for the Science of Light in Erlangen, Germany.

In collaboration with Prof. Sadas Shankar at Stanford University, ESR 1 and a collaborator from KTH Stockholm will use NIR to demonstrate, theoretically and experimentally, the increased energy efficiency of computing systems whose underlying physics is more closely aligned to their computational workload, e.g., using analogue hardware to implement continuous-time neural networks.

2.3 References

[1] L. Appeltant, M. C. Soriano, G. Van der Sande et al., "Information processing using a single dynamical node as complex system," Nature communications, vol. 2, no. 1, p. 468, 2011.

[2] P. Antonik, M. Haelterman, and S. Massar, "Brain-inspired photonic signal processor for generating periodic patterns and emulating chaotic systems," Physical Review Applied, vol. 7, no. 5, p. 054014, 2017.

[3] E. Picco, P. Antonik, S. Massar, "High speed human action recognition using a photonic reservoir computer", Neural Networks, Volume 165, p. 662-675, 2023, https://doi.org/10.1016/j.neunet.2023.06.014.

[4] E. Picco, S. Massar, "Real-Time Photonic Deep Reservoir Computing for Speech Recognition", in 2023 International Joint Conference on Neural Networks (IJCNN), IEEE, 2023.

[5] A. Lupo, E. Picco, M. Zajnulina, S. Massar, "Fully analog photonic deep Reservoir Computer based on frequency multiplexing." arXiv preprint arXiv:2305.08892, 2023.

[6] L. Butschek, A. Akrout, E. Dimitriadou, A. Lupo, M. Haelterman, and S.Massar. Photonic reservoir computer based on frequency multiplexing. Optics Letters, 47(4), 782-785, 2022.

[7] A. Lupo, L. Butschek, and S. Massar. Photonic extreme learning machine based on frequency multiplexing. Optics express, 29(18), 28257-28276, 2021.

[8] A. Lupo, and S. Massar. Parallel extreme learning machines based on frequency multiplexing. Applied Sciences, 12(1), 214, 2021.