Neuromorphic computing based on silicon photonics

Andrew Katumba *Student Member, IEEE*, Matthias Freiberger, *Student Member, IEEE*, Floris Laporte, Alessio Lugnan, Stijn Sackesyn, Chonghuai Ma, Joni Dambre, *Member, IEEE* and Peter Bienstman, *Member, IEEE*

(Invited Paper)

***Abstract*—We present our latest progress using new neuromorphic paradigms for optical information processing in silicon photonics. We show how passive reservoir computing chips can be used to perform a variety of tasks (bit level tasks, non-linear dispersion compensation, ...) at high speeds and low power consumption. In addition, we present a spatial analog of reservoir computing based on pillar scatterers and a cavity, that can be used to speed up classification of biological cells.**

***Index Terms*—silicon photonics, neuromorphic computing, reservoir computing.**

I. Introduction

THE surge in the volumes of data generated and consumed by devices has pushed the bounds of the scalability and speed of signal processing and computational systems. The consequence is a re-ignition of research into non-conventional information processing techniques that deviate from the well-established von-Neumann approach.

Analog computing has been proposed as a promising implementation in this regard on a number of platforms. The analog computing paradigm, unlike conventional computing, relies on the information processing capabilities of certain physical systems. The processing usually hinges on the evolution of the internal state of the dynamical behavior of the physical system in response to an appropriately encoded input.

Analog information processing platforms are typically coupled to an adaptation and readout system to guide the evolution of the system state (or a subset of it) towards a solution whose accuracy metric is usually known beforehand. This is then followed by a readout that transforms the observed physical signal of interest into a form that is suitable for interpretation by digital computers. A prevalent way of implementing the adaptation of the computational system's internal states is through machine learning.

Machine learning encompasses a set of techniques to teach computer systems how to perform complex tasks on previously unseen data, without explicitly programming them. Examples of tasks that are suitable for some form of machine learning are classification, regression or pattern recognition. The collection of machine learning techniques is extensive and for every application the most appropriate technique has to be selected, depending on the applications specific demands.

A. Katumba, F. Laporte, A. Lugnan, S. Sackesyn, C. Ma, and P. Bienstman are with Photonics Research Group, Department of Information Technology, Ghent University - imec, Belgium (Peter.Bienstman@UGent.be)

M. Freiberger and J. Dambre are with IDLab, Department of Electronics and Information Systems, Ghent University - imec, Belgium

Manuscript received February 1, 2018; revised XX, 2018.

One important class of techniques are the so-called artificial neural networks (ANNs), that consist of a network of interconnected computational units, dubbed neurons. The layout and operation of the ANN is inspired by the structure and information processing mechanism of the human brain. Recurrent neural networks (RNNs), a subtype of neural networks, introduce memory into the network by creating directed interconnection cycles between neurons in order to tackle tasks with temporal behavior.

Reservoir computing (RC) [1], [2], [3] was proposed as a methodology to ease the training of these recurrent networks, which was typically rather challenging.. More recently, however, it has gained popularity as a neuromorphic computational paradigm to solve a variety of complex problems. It has been shown that RC performs very well on e.g. speech recognition and time series prediction. Reservoir computing initially emerged as a software-only technique and merely presented another algorithmic way of processing temporal data on digital computers. However, it has evolved into much more over the past decade. The RC system consists of three basic parts: the input layer which couples the input signal into a non-linear dynamical system, the "reservoir" (i.e. the recurrent neural network, which is kept untrained) and finally the output layer that typically linearly combines the states of the reservoir to provide the time-dependent output signal. An illustration of this reservoir computing architecture is given in Fig. 1.

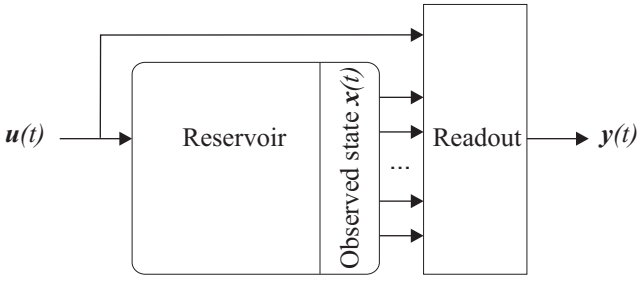


Fig. 1: Schematic representation of a reservoir computing system. The input signal $u(t)$ is fed into the reservoir and the resulting reservoir states $x(t)$ are used to learn a linear readout that is then used to generate the output signal $y(t)$.

To use the reservoir to solve a particular task, a machine learning algorithm is used to train a set of weights (the readout) using a set of known labeled example data, such that a linear combination of the optical signals recorded at each node approximates a desired output as closely as possible. These weights are then used to generate the output signal for any unseen subsequently injected input signal sequences. RC

systems are fast to train and quickly converge to a global optimum. They have shown state-of-the-art performance on a range of complex tasks on time-dependent data (such as speech recognition, nonlinear channel equalization, robot control, time series prediction, financial forecasting, handwriting recognition, etc.).

A key discovery was that the reservoir computing platform provides a natural framework for implementing hardware-based learning systems for which there may be only a limited ability to granularly influence the internal state of the dynamical system (i.e. the reservoir). Examples of RC implementations in mechanical systems, memristive systems, atomic switch networks, boolean logic elements and photonic systems can be found in [4], [5], [6], [7], [8].

To date, experimental demonstrations of photonic reservoirs routinely achieve state-of-the-art performance on various information processing tasks. Implementations based on a single nonlinear node with a delayed feedback architecture have proven that photonic RC is competitive for analog information processing [9], [10], [11], [12], [13], [14], [15], [16], [17]. Moreover, integrated photonic reservoirs can push computation speeds even higher for digital information processing. The performance of integrated photonic reservoirs has been studied numerically for networks of ring resonators [18], [19], [20], [21], [22], networks of SOAs [7], and experimentally with networks of delay lines and splitters [23]. Integrated photonic reservoirs are particularly compelling, especially when implemented in a CMOS-compatible platform as they can take advantage of its associated benefits for technology reuse and mass production.

II. Passive reservoir computing

A recent development in the design of RC systems is the realization that for certain tasks that are not strongly nonlinear, it is possible to achieve state-of-the-art performance using a completely passive linear network, i.e., one without amplification or nonlinear elements. The required nonlinearity is introduced at the readout point, typically with a photodetector [23]. The work discussed in this paper is also based on this architecture. Aside from the integrated implementation introduced in [23], the passive architecture has been adapted to the single node with delayed feedback architecture in form of a coherently driven passive cavity [9].

Apart from simplicity from a fabrication point-of-view, a further advantage of such a passive architecture is the reduced power consumption, since the computation itself does not require external energy.

The integrated photonic reservoirs typically studied in the past are limited to planar architectures in a bid to minimize crossings that manifest as a source of signal cross-talk and extra losses. This constrains the design space from which reservoir configurations can be chosen. The swirl reservoir architecture, which is also used in this work, was introduced in [18] as a way to satisfy planarity constraints while allowing for a reasonable mixing of the input signals. A 16-node photonic swirl reservoir is shown in Fig. 2. One way of looking at this reservoir is that it acts as a giant multi-path interferometer, which mixes the input in such a way that it gets converted into a higher dimensional space where a linear classifier will be able to separate the different classes more easily. The input to the integrated photonics reservoir chip could be to a single input node as is in [23] or to multiple inputs, which has some advantages over the former strategy, as is discussed extensively in [24].

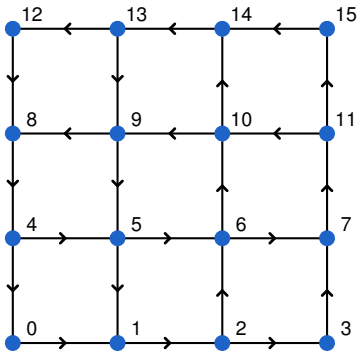


Fig. 2: Signal flow in a 16-node swirl reservoir architecture. The time-dependent output at each numbered node is linearly combined to result in the answer of the computation. As for inputs, depending on the application the input can be inserted in one or more of the numbered nodes.

In discretized time, the reservoir state update equation is given in a general form by:

$$\vec{x}[k+1] = \boldsymbol{W}_{res}\vec{x}[k] + \vec{w}_{in}(\vec{u}[k+1] + u_{bias}) \qquad (1)$$

where $\vec{u}$ is the input to the reservoir and u_{bias} is a fixed scalar bias applied to the inputs of the reservoir. For an N-node reservoir, $\boldsymbol{W_{res}}$ is an $N \times N$ matrix representing the interconnections between reservoir components taking into account splitting ratios and losses, with phases drawn from a random uniform distribution on $[-\pi, \pi]$, $U(-\pi, \pi)$. $\vec{w}_{in}$ is an N-dimensional column vector whose elements are nonzero for each active input node. These input weights are similarly chosen from $U(-\pi, \pi)$.

Our work in [23] experimentally verifies that a passive integrated photonic reservoir can yield error free performance on the header recognition task for headers up to 3 bit in length with simulations indicating that it should be possible to go up to 8 bit headers (see Fig. 3). We additionally demonstrated that the passive integrated photonics reservoir can be used for bit level manipulations on digital optical bit streams that could be useful for various telecommunications tasks.

III. Integrated optical readout

A. Rationale

A drawback of our current experiments is that the taking of a linear combination of the reservoir states is still happening in

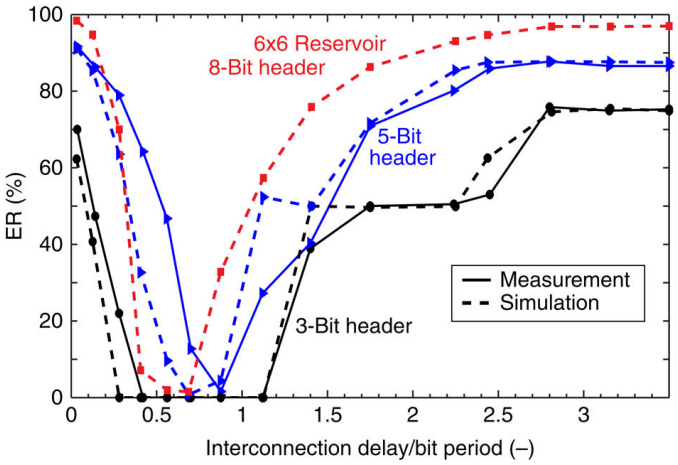


Fig. 3: Performance of a 6×6 swirl passive integrated photonics reservoir on the 3, 5 and 8 bit header recognition task [23].

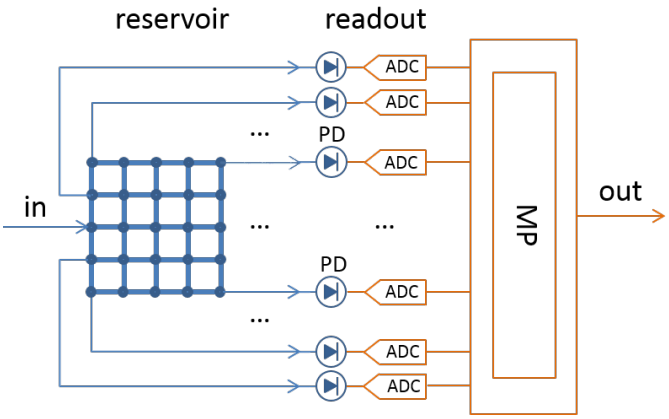


Fig. 4: Reservoir and readout system of existing RC photonic chip prototypes, with the nodes of the reservoir being collected in the readout section, where optical output signals are converted to electrical signals and then processed to a final output. PD: photodiode, ADC: AD converter, MP: Microprocessor. The blue and orange parts represent respectively the optical and electronic signals and components.

the electrical domain. Indeed, Vandoorne et al. [23] transferred the signal at each node from the optical to the electrical domain using a photodetector, and then sent it through an AD converter. Finally, the required linear combination of signals and readout weights was performed using a microprocessor. See Fig. 4 for a detailed illustration of the process.

In order to truly reap the benefits of optical computing though, signals need to be processed at very high data rates in an energy-efficient way. Considering ecological as well as economical factors, minimizing power consumption is of paramount importance for future computing technologies. From that perspective, the approach pursued in [23] for reading out integrated photonic reservoirs is inefficient, since there is a significant energy and latency cost associated to it. Hence, it is desirable to perform the summing of signals in the optical domain instead of in the electrical domain. Using such an

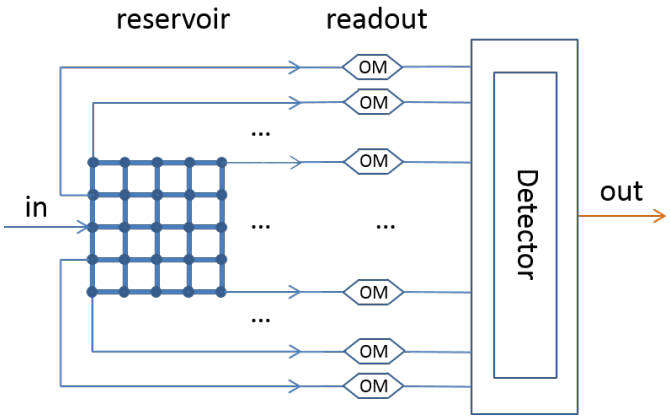


Fig. 5: Schematic of a fully optical readout. Each optical output signal is modulated by an Optical Modulator (OM) implementing the weights. The optical outputs are then sent to a photodiode where all signals are summed and then converted to a final electric output signal.

integrated optical readout, only a single photodetector, which receives the weighted sum of all optical signals, is required. A straightforward low-power optical weighting element can take the form of a reverse-biased pn-junction. An even better solution would be to use non-volatile optical weighting elements, such as the ones that are currently being developed by several groups [25], [26], [27]. Fig. 5 illustrates the concept of a fully optical integrated readout.

B. Training integrated optical readouts

While a fully optical readout is definitely desireable, employing it means we lose direct observability of the states of the photonic reservoir. Observing all states is mandatory though, in order to use classical linear readout training algorithms such as ridge regression [28] and other least-squares approaches. At first glance, a possible solution could be to add a separate high-speed photodetector to each reservoir node, which is only used during training to observe the states. The weights would then be calculated in the electrical domain, while the trained reservoir could still be operated entirely in the optical domain. This is unfortunately challenging due to a number of reasons. First, high-speed photodetectors tend to be costly in terms of chip footprint. Therefore, such an architecture would not scale well when increasing the number of nodes in the photonic reservoirs to numbers common in classic echo state networks [2]. Second, since passive photonic reservoirs make use of coherent light for added richness, the tunable readout weights in the optical domain have to be complex-valued as well. However, unless we go to even more complicated coherent detectors, these photodiodes can only measure the intensities of the states and not their phases, so we cannot calculate the correct complex-valued weights. Note that this is different from the approach in [23], which used real-valued weights on real-valued signals, whereas here we would need to use complex-valued weights on complex-valued signals.

A second solution could be to train the weights based on simulations of the behavior of a virtual reservoir, using photonic circuit simulation software, which will obviously have full observability of all the nodes. However, the fabrication tolerances of these devices are such that the propagation phase of two nominally identical waveguides could be completely different. This prohibits the successful transfer of weights trained using the idealized simulated reservoir to actual hardware.

A possible resolution to this issue is the application of a pretraining-retraining approach, where a passive photonic reservoir is pretrained in simulation and the trained weights are transferred to an actual reservoir on chip. Thereafter, the actual training error is minimized on the reservoir by fine-tuning the weights of a given integrated optical readout using a black-box optimization approach. Unfortunately, previous simulation studies have shown that such an approach is not feasible due to high manufacturing variations in effective index of planar waveguides [29].

A final possibility, which is the one we actually prefer, is to exploit the weighting mechanism of the optical readout of the reservoir to read out all reservoir states through the single photodetector available at the end of the summing structure. Reading out the state variable s_i in response to the training input sequence can be simply realized by setting the weight of that state variable to 1 and all other weights to 0. By presenting the whole training input sequence to the reservoir n times, where n is the number of nodes of the reservoir, the training responses of all nodes can be collected through the single photodetector. By taking the square root of each measured power value, we can approximately invert the nonlinearity of the photodetector and obtain an estimate for the evolution of the light intensity over time at the corresponding reservoir node. However, since passive photonic reservoirs work with coherent light, it is not sufficient to know only the light intensities at the points predefined as reservoir states: we also need to know the corresponding phase of the light. While the absolute phase of the optical signal inside the reservoir is lost within the photodetection process, the relative phases between the optical state signals influence the power at the detector output. We therefore estimate the phase between two given optical signals within the integrated reservoir by obtaining the evolution of the sum of their states through time as we apply the training signal at the reservoir's input. We now use the phase of one state signal (node) as a reference. Using the evolution of the power of the complex sum between the reference node's signal and each other node signal, as well as the previously determined powers of all individual states, we are able to estimate the relative phase of each node signal with respect to the reference node using basic trigonometric relationships. The last stage of this calculation consists of an inverse cosine, which is is injective, in the sense that there are always two solutions within the range $[-\pi, \pi]$. To discriminate between them, we perform a third measurement between the reference node's signal and each other node's signal, now shifting the phase of the reference node's readout weight by $\frac{\pi}{2}$ and comparing with the phase estimate obtained before. As a result, the whole process requires that we feed the training sequence through the reservoir $3n - 2$ times. Under ideal conditions, this nonlinearity inversion procedure is exact. In order to confirm that, we conduct the following experiment:

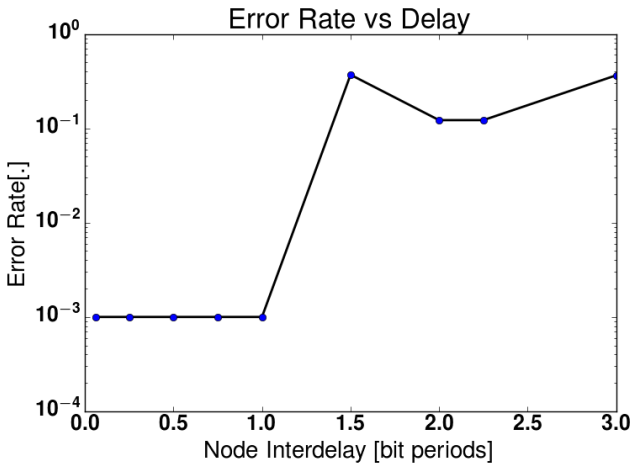


Fig. 6: Bit error rate versus the time delay between nodes (interdelay) in bit periods for a 4x4 passive swirl architecture with integrated optical readout performing 3-bit header recognition (pattern 101). The minimal detectable error rate is 10^{-3}. Results are averaged over 10 reservoirs with different phase configurations for every data point.

We train a passive 4x4 photonic swirl reservoir architecture with integrated readout to perform 3 bit header recognition on a power-modulated digital signal fed into to the reservoir. The integrated optical readout is simulated by inner product multiplication of the complex reservoir states with a complex weight vector and subsequently taking the square of the absolute value of the resulting complex output signal vector. The weight vector is trained on the complex reservoir states using complex-valued ridge regression. Input is fed to the reservoir via node 2.

To determine suitable values of the delay time between reservoir nodes (node interdelay) for the architecture, we train our simulated reservoirs with increasing delay time using complex-valued ridge regression. Fig. 6 shows the achieved bit error rate as a function of increasing interdelay between nodes at a input signal bitrate of 10 Gbps. After having set up this baseline, we exchange the true reservoir states in our setup with the states estimated through the method described above, and find the resulting bit error plot to be identical to the plot in Fig. 6. A detailed mathematical description of the approach as well as more extensive experiments using a realistic detector model will be found in [30] .

C. Influence of weight resolution

When the weights of the linear combination are implemented through reverse-biased pn-junctions, the resolution of these weights is determined by the DAC which drives the junction and is therefore typically fine enough. However, we are also investigating alternative non-volatile weighting elements which, due to the physics of their operation, have

an intrinsically lower weighing resolution. An example of this are weights based on the integration of transition-metal oxide materials like barium titanate (BaTiO3) [25]. These elements typically only have about 20 discrete weight levels. Under such circumstances it is important to investigate how the reservoir computing performance is influenced by such a limited resolution.

To simulate the impact of the limited resolution, we study the performance of the reservoir for 4 different weight resolutions (3, 4, 5 and 6 bit), both for the amplitude and the phase. After the reservoir, all the states are used to perform a complex-valued ridge regression together with desired output. In the regression, different regularization parameters are being used, and the one with the lowest error rate is be chosen. Meanwhile, for each regularization parameter, the limited weight resolution is simulated by simply rounding each weight to the nearest discretized weighting value. For the amplitude, the weight range is chosen to correspond to that of the situation with infinite resolution. As for the phase, the discretized weighting levels are in the interval $[-\pi, \pi]$.

Simulation results for the 2-bit XOR task in figure 7 show how the performance approaches that of a system with infinite weight resolution (blue line) as the number of bits increases. For 6 bits, there is a narrow interval of interdelays where the error rates almost coincide. Additionally, the error bars related to different random instantiations of the reservoir also decrease in that regime, which is indicative of robust behavior.

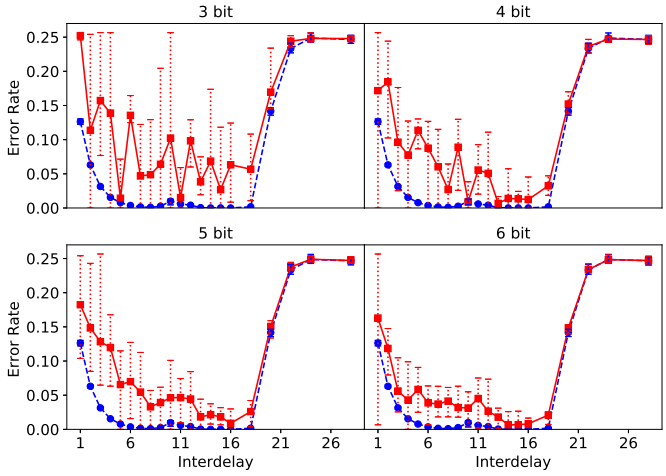


Fig. 7: Error rate as a function of interdelay for different weight resolutions (red curves). The blue curves are for the case of infinite weight resolution.

IV. Applications related to telecom and bitstreams

In this section, we will show some simulation results on how the passive swirl architecture as described above can be used to perform a number of applications in the domain of communications and bitstream processing. We will study non-linear dispersion compensation, both for amplitude and phase-encoded signals. Additionally, we will show boolean operations on multilevel logic signals.

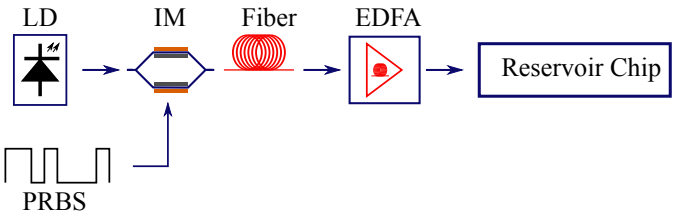


Fig. 8: Schematic representation of the simulation setup to generate data for the signal equalization task. The input pseudo-random bit sequence (PRBS) signal is modulated onto a laser signal. This is transmitted over a fiber link, after which the data is saved to file and used as input for the nanophotonic reservoir simulation.

A. Non-linear dispersion compensation

Optical technology is at the core of all modern telecommunications for high speed, long and short reach applications. Fiber-based technologies are pervasive in all forms of networks from core and metro to data center and local area networks. Moreover, current industrial efforts are towards getting lightwave technology closer and closer to the last-mile end-user and in so doing take advantage of the massive bandwidth, energy efficiency and other benefits it provides. These advantages combined with advances in photonics leading to cheaper lasers, modulation technology, optical domain broadband amplification with Erbium Doped Amplifiers, place this technology squarely at the backbone of todays internet information superhighway. However, the various elements constituting fiber-based lightwave networks also contribute to the degradation of optical signals during generation, transmission and reception phases [31].

Signal impairments are an inevitability for any kind of communications system manifesting at the receiver as erroneous detections that need to be addressed. In optical fibre systems, these imperfections can mainly be traced back to dispersion, amplified spontaneous emission at amplification points, attenuation and reflections in fiber links; optical nonlinearities in fibers; timing jitter introduced at O/E and E/O points.

These issues are exacerbated in modern high-speed systems based on coherent modulation formats, where the nonlinearity in the fiber poses serious problems related to error-free propagation. These are typically solved in the electronic domain using advanced DSP post-processing, but such an approach consumes a lot of power and chip real estate. Photonic reservoir computing could provide an alternative here, to undo (part of) these signal impairments already in the optical domain.

To illustrate this, we will first show simulation results related to NRZ signal propagating over a realistic optical link, followed by BPSK signals propagating over an artificial link with extreme nonlinearities and intersymbol interference.

1) NRZ over an optical link: Simulated telecom data is generated with *VPI Transmission Maker v9.2* software, using the setup as in figure 8. VPI software incorporates realistic models of the signal degradation mechanisms encountered in telecom links, caused by the various optical components and physical phenomena outlined above, and also takes into account how they evolve over the transmission distances

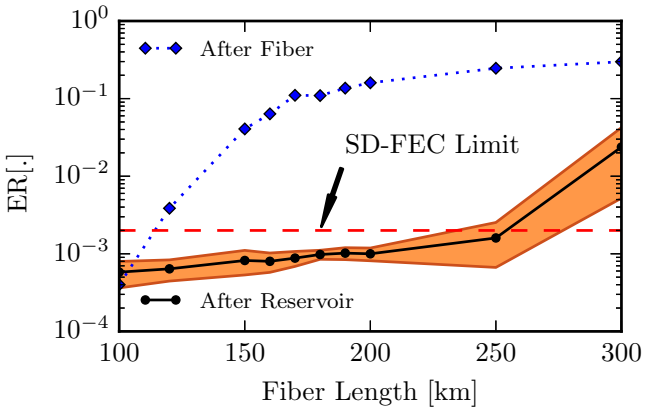


Fig. 9: Error rates after the fiber link and after the reservoir lengths up to 300 km. Error margins are also indicated for 5 random initialization of the phases in the reservoir. A soft decision Forward Error Correction limit (SD FEC limit) of 0.2×10^{-2} is also shown. Error free operation is possible for all error rate values below this limit.

considered (which can be long for example in metro and long-haul networks). The generated data is then used for training and testing reservoir designs using in-house circuit simulations and machine learning libraries.

Results are shown in Fig. 9 for a 10 Gbps link. The results indicate a BER improvement to well below the Soft-Decicion Forward Error Correction (FEC) limit of 0.2×10^{-2} for connections containing up to 250 km of fiber. This means that the chip can be used in conjunction with appropriately chosen error correcting codes to achieve error-free communication on the link. Such a design would be suitable for signal equalization in, e.g., metro networks.

2) BPSK over an artificial channel: In order to show that the techniques described above also work for phase-encoded symbols, we perform a simulation where we send a BPSK signal (i.e., with symbols $+1$ and -1) over an artificial channel with extreme non-linearity and intersymbol interference. Modulation speed is 10 GHz. Idealised square pulses are sent through a first-order Butterworth filter with 3dB cutoff at 25 GHz. Intersymbol interference is modeled by the following expression:

$$x_n = 0.6x_{n-1} + x_n - 0.7x_{n+1} \tag{2}$$

This means that almost half of the energy of the bit pulse is spread over adjacent time slots.

For our artificial channel, we consider a strong 9-th order non-linearity. Given an input x, the output x_{NL} of the non-linear channel is given by:

$$\tilde{x} = x + 0.036x^2 - 0.011x^3 \tag{3}$$

$$x_{NL} = \tilde{x} + 0.5\tilde{x}^2 - 0.3\tilde{x}^3 \tag{4}$$

Finally, AWGN with signal-to-noise ratio of 20 dB is added.

The constellation diagram at the output of the link is shown in Fig. 10. Rather than having a blue blob at -1 and a red blob

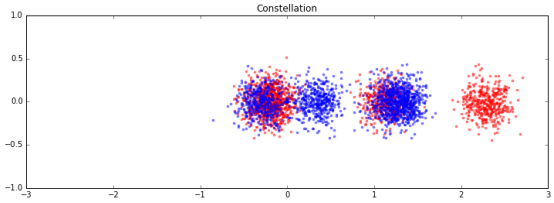


Fig. 10: Constellation diagram of a BPSK signal at the output of an artificial channel with strong intersymbol interference and non-linearity

at +1, as would be the case for a much more benign channel, the multiple echoes due to the intersymbol interference and the distortion due to the nonlinearity are clearly visible.

In order to try and undo the transmission impairments, we send that signal through a 7×7 silicon photonics swirl network, and train the reservoir to recover the original bitstream, but with a delay of 1 period. The signal at the output of the reservoir is sampled in the middle of the period. After a simple threshold detection, a bit error rate of 4% is obtained. While not zero, this number has to be compared with a 33% error rate that is obtained using a tapped filter having access the last 3 samples of the signal (given the size and the losses in the reservoir, this corresponds to a similar memory for both cases). The reservoir clearly has better performance, moreover in a way that is amenable to a direct implementation in the optical domain, without the need for analog-to-digital conversion.

B. PAM-4 logic

In machine learning, and in particular in photonic reservoir computing, the XOR-task with memory is a commonly used benchmark task. It consists of training the reservoir to correctly compute the XOR of two symbols appearing at a certain time differences in the datastream, typically the current bit and the previous bit. Usually this task is performed on data encoded with a binary modulation format, but here we will extend the problem to the much more complex case of computing the XOR on a PAM-4 signal, a two bit per symbol amplitude modulation format.

We consider a PAM-4 signal modulated at 10 GHz, which translates to a bitrate of 20 GHz. A first-order Butterworth filter is used, with the same parameters as in section IV-A2, to generate realistic input pulses from a square pulse bitstream. Subsequently, AWGN with signal-to-noise ratio of 30 dB is added. The signal is then low-pass filtered at 25 GHz before downsampling to have a smoother signal for classification.

Results of a sweep in the size of the reservoir (Fig. 11) give a good idea of the increasing computational power of the photonic reservoir when the reservoir consists of more nodes. As illustrated in Fig. 12, the 8×8 reservoir does not have enough computational power to correctly compute the XOR task, and has a SER (symbol error rate) of 30%. Similar behavior is observed for smaller reservoirs. From 10×10 nodes on, all four levels start to be distinguishable with some overlap remaining. Gradually proceeding to 20×20 nodes, the artificial levels in the center that were present for the 8×8

reservoir get pushed away to the correct levels for the symbols 0 and 3. This 400-node reservoir is capable of computing the XOR with delay with a SER of less than 5%.

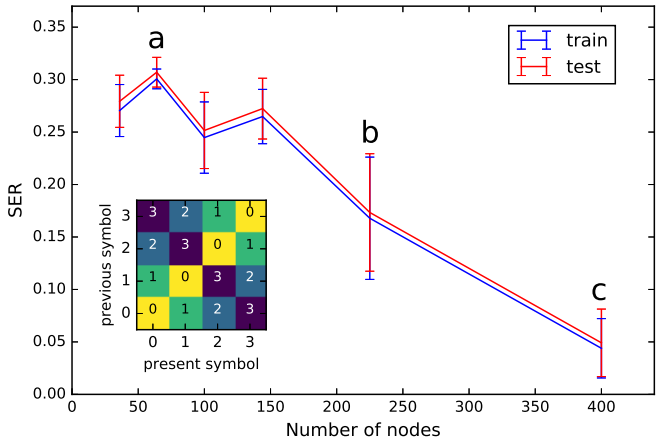
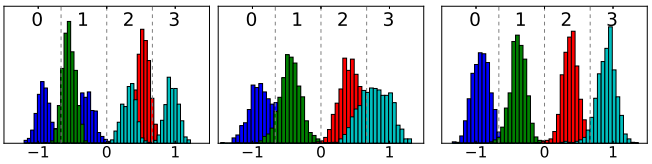


Fig. 11: Evolution of the SER as a function of the photonic swirl reservoir dimension. The inset shows the truth table for two-bit XOR with one symbol delay. A, B and C refer to the histograms in Fig. 12.



(a) 8 × 8 (b) 15 × 15 (c) 20 × 20

Fig. 12: Histograms illustrating the classification of the symbols for a photonic swirl reservoir for a given number of nodes, color-labeled with the desired XOR outcome.

V. Pillar Scatterers for Cell Identification

The basic conceptual structure of reservoir computing, i.e. an untrained recurrent nonlinear network that improves the performance of a trainable linear readout, can be also applied to time-independent signals such as images. In this case, memory in the nonlinear part is not required and the whole system is usually called extreme learning machine (ELM) [32]. We previously provided a proof of concept, based on finite-difference time-domain (FDTD) simulations, of an integrated photonics application of ELM for fast and label-free classification of biological cells [33]. In this application, a passive optical stage comprising a collection of pillar silica scatterers embedded in a silicon nitride cladding is used to process the light forward-scattered by a cell when illuminated via a green monochromatic source (Fig. 13). We will consider two different cases: our previous work without a cavity, and a new extension where an extra cavity increases the classification performance.

Pillar scatterers without a cavity

The diffraction pattern projected by the cell and by the dielectric scatterers is acquired by an image sensor whose pixel outputs are used as input for a linear classifier (logistic regression) implemented in the electric domain. Such a configuration represents an ELM system where the light scattered by the cell is the input signal while the dielectric scatterers and the image sensor play the role of a random nonlinear network whose output nodes are represented by the sensor pixels. Indeed, the image sensor performs a nonlinear function of the impinging phase-encoded[1] signal by giving its intensity as output. The scatterer configuration determines how the light coming from different regions of the cell is split and overlapped on the sensor display, changing the acquired interference pattern. Since the scatterer stage includes a relatively high number of pillars and therefore is difficult to completely optimize, only its overall complexity was tuned to maximize the classification performance, similarly to how it is usually done in reservoir computing when tuning the reservoir to an almost chaotic regime. In particular, the transfer function complexity of the scatterer stage was tuned by changing only one parameter, that is the amplitude of a slight random displacement applied to the scatterers with respect to their ordered position along layers.

In order to properly train the readout classifier and to test its performance once trained, a sufficient number (thousands) of diffraction pattern samples had to be computed and provided to the training algorithm. Randomized cell models were employed to create reasonable variability in the diffraction pattern acquisition. The employed models are the result of a trade-off between computational cost and closeness to reality. Such a trade-off is legitimated by the fact that the goal of the work is not to provide absolute references for real applications but, instead, to investigate a relative difference between the classification performance with and without using scatterers.

In [33], two different classification tasks were considered. The first is based on average nucleus size and aims to distinguish between “normal” cells (small nucleus) and “cancer” cells (bigger nucleus). The names in quotation marks were chosen because of the common tendency of cancer cells to show evident irregularities in nucleus size [34]. The second task is based on nucleus shape and aims to distinguish between “lymphocytes” (big quasi-spherical nucleus) and “neutrophils” (nucleus divided in 3 lobes). The names in quotation marks refer to two among the most common white blood cells that are present in human blood.

Even if significant improvements ($\sim 50\%$, Fig. 14 a) due to the introduction of pillar scatterers were achieved in the classification based on nucleus size, a simple quantitative reasoning on how nucleus size changes affect the input of the readout classifier suggested that better results could be in principle obtained. In particular, the difference in phase shift accumulated by the light through the 2 types of nucleus corresponding to the 2 cell classes was too small and therefore was expressed in an almost linear way by the acquired intensity pattern, while nonlinearity is required by the ELM approach (see [33] under *Nonlinear phase sensitivity* in *Results and discussion* section). The employment of an UV laser source ($\lambda = 337.1nm$), that implies an increase in the optical path

[1]The cell structure information is mainly encoded in the phase of the scattered light, since cell absorption is usually negligible.

through the given nuclei, was considered as a solution and much better classification performance was obtained (Fig. 14 b). However, when real applications are concerned, this option is highly impractical as UV lasers are usually quite expensive and would probably damage or even kill the illuminated cells.

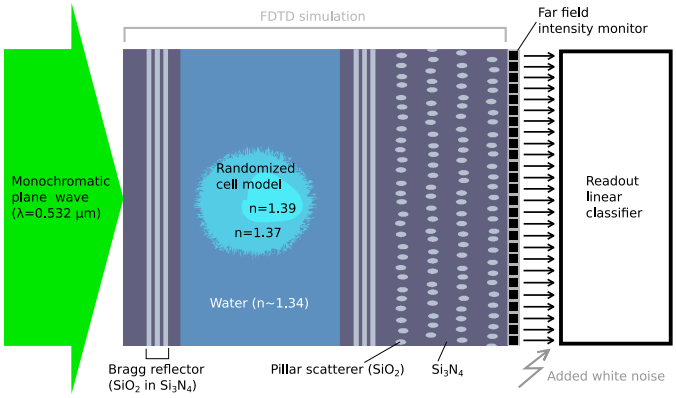


Fig. 13: Schematic of the classification process. A monochromatic plane wave impinges on a Fabry-Pérot optical cavity composed by Bragg reflectors and containing a microfluidic channel with a cell in water ($n_{H_2O} \sim 1.34$), which has a low refractive index contrast ($n_{cytoplasm} = 1.37$, $n_{nucleus} = 1.39$); the forward scattered light passes through a collection of silica scatterers ($n_{SiO_2} \sim 1.461$) embedded in silicon nitride ($n_{Si_3N_4} \sim 2.027$) and organized in layers; the radiation intensity is then collected by a far-field monitor, which is divided into bins (pixels); each pixel value is fed into a trained linear classifier (logistic regression) that consists of weighted sums (one per class) of the pixel values. The weights are trained so that the sum exceeds a certain threshold value only if the corresponding input class is recognized.

Pillar scatterers with a cavity

In this work a more feasible solution is presented. It consists of increasing the effective optical path length through the cell by inserting it in an optical cavity. Intuitively, this makes the impinging light pass, on average, more than once through the cell. In practice, in the FDTD simulation design 2 Bragg reflectors are placed at the 2 external sides of the microfluidic channel, orthogonally to the light beam direction, creating a Fabry-Pérot cavity (Fig. 13). The employed Bragg reflectors are each composed of 3 layers of SiO_2 with a width of $(455 \pm 10)nm$ in a Si_3N_4 cladding. The error in the layer width was implemented by adding a random value sampled from an uniform probability distribution between 10 and 10 nm. It approximately accounts for fabrication errors. The distance $D = 21.02\mu m$ between the reflectors was chosen so that the portion of light passing through and near the nucleus of the cell was resonant. This was done by monitoring the light intensity inside the cavity for different values of D. Note that such a tuning was relatively easy to perform because the cell acts as a weak converging lens, providing an additional light confinement along the microfluidic channel direction.

The reflectivity R of the reflectors is also a crucial parameter, since it controls the cavity Q-factor and therefore controls both the sensitivity of the resonance to intracavity optical path lengths and how long the light stays, on average, inside the cavity. This means that by tuning R a trade-off has to be achieved between how much the phase shift due to the selected resonant cell part is increased and how much the corresponding resonance is stable. In this case, for instance, the sensitivity of the acquired intensity pattern to the nucleus size is to be improved by increasing the average time that the resonant light passing through the nucleus stays in the cavity. On the other hand, if the cavity Q-factor is too high, the resonance strength might be strongly influenced by uninteresting small details of the cell structure or by fabrication errors. Generally, the cavity should be designed so that the average light phase shift differences due to the optical feature of interest corresponding to the considered classes are roughly between $\pi/2$ and 2π. Indeed, a phase shift difference significantly lower than $\pi/2$ would induce a nearly linear difference on the acquired intensity pattern, undermining the nonlinearity requirement of the ELM approach. On the other hand, a phase shift difference significantly bigger than 2π would make the acquired intensity pattern too sensitive to uninteresting small changes in the cell optical features, undermining the stability requirement of the ELM approach.

In particular, the reflectors employed in the simulations (composed of 3 layers) have a satisfying reflectivity of $\sim 56\%$, while it turned out that similar reflectors with 4 and 5 layers have a too high reflectivity, respectively of $\sim 73\%$ and $\sim 85\%$. All the simulation and machine learning aspects concerning the investigation presented in this work are as in [33], apart from the introduction of the described optical cavity, that also required longer FDTD simulation times.

Results

Fig. 14 compares the results of green ($\lambda = 532nm$) and UV ($\lambda = 337.1nm$) light presented in [33] to the new structure described here with green light and an optical cavity. A substantial improvement is observed with respect to the green source case without cavity. In particular, the classification improvement due to the use of scatterers is increased by factor 5 by the cavity, for sufficiently low but still plausible noise levels ($< 10\%$). At these noise levels, the results are similar to what was obtained with an UV light source, without the drawback of possible cell damage. For higher noise levels an increased sensitivity to noise pushes the classification error rate to significantly higher values. Finally, it should be stressed that an additional advantage arising from the use of an optical cavity is that it can be designed to increase the intensity pattern sensitivity towards specific optical path lengths, making the optical features of interest more evident to the readout classifier with respect to other competing ones.

VI. Conclusion

In this paper, we reviewed the concept of photonic reservoir computing, with a special emphasis on passive reservoir structures. We presented new simulation results that show that this paradigm can be used successfully to perform a variety of tasks (bit level tasks, non-linear dispersion compensation,

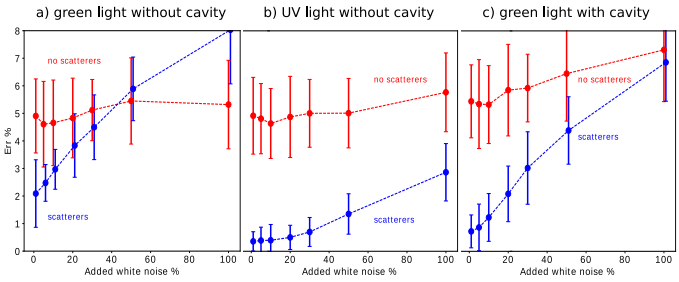


Fig. 14: Comparisons between the error rates of "normal" and "cancer" cell classification as a function of the white noise percentage added to the acquired intensity patterns, corresponding to the absence (in red) and the presence (in blue) of scatterers. The displayed error rate values are averages on the values obtained using $250, 260, 270, 280, 290, 300$ number of pixels in the simulation far field monitor. The graphed error bars represent a confidence interval of 2 standard deviations and corresponds to the error obtained through validation, by randomly shuffling the simulated intensity patterns 20 times between training and test sets. The other simulation and machine learning details are as described in [33]. **a)** A green laser source ($\lambda = 532nm$) was employed, without optical cavity. **b)** An UV laser source ($\lambda = 337.1nm$) was employed, without optical cavity. **c)** A green laser source ($\lambda = 532nm$) and a Fabry-Pérot cavity were employed, as in Fig. 13.

...) at high speeds and low power consumption. In addition, we presented a spatial analog of reservoir computing based on pillar scatterers and a cavity, that can be used to speed up classification of biological cells.

ACKNOWLEDGMENT

This research was funded by the EU Horizon 2020 PHRESCO Grant (Grant No 688579), the BELSPO IAP P7-35 program Photonics@be and the Research Foundation Flanders (FWO)(Grant No G024715N).

REFERENCES

[1] W. Maass, T. Natschläger, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," *Neural computation*, vol. 2560, pp. 2531–2560, 2002.

[2] H. Jaeger and H. Haas, "Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication." *Science (New York, N.Y.)*, vol. 304, pp. 78–80, 2004.

[3] D. Verstraeten, B. Schrauwen, M. D'Haene, and D. Stroobandt, "An experimental unification of reservoir computing methods," *Neural Networks*, vol. 20, no. 3, pp. 391–403, 4 2007.

[4] H. Hauser, A. Ijspeert, R. Füchslin, R. Pfeifer, and W. Maass, "Towards a theoretical foundation for morphological computation with compliant bodies," *Biological Cybernetics*, vol. 105, no. 2011, pp. 355–370, 12 2011.

[5] H. O. Sillin, R. Aguilera, H.-H. Shieh, A. V. Avizienis, M. Aono, A. Z. Stieg, and J. K. Gimzewski, "A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing." *Nanotechnology*, vol. 24, p. 384004, 2013.

[6] M. S. Kulkarni and C. Teuscher, "Memristor-based reservoir computing," in *Proceedings of the 2012 IEEE/ACM International Symposium on Nanoscale Architectures - NANOARCH '12*. New York, New York, USA: ACM Press, 2012, pp. 226–232.

[7] K. Vandoorne, "Photonic reservoir computing with a network of coupled semiconductor optical amplifiers," Ph.D. dissertation, 2011.

[8] Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, and S. Massar, "Optoelectronic Reservoir Computing," *Scientific Reports*, vol. 2, p. 287, 2 2012.

[9] Q. Vinckier, F. Duport, A. Smerieri, K. Vandoorne, P. Bienstman, M. Haelterman, and S. Massar, "High-performance photonic reservoir computer based on a coherently driven passive cavity," *Optica*, vol. 2, no. 5, pp. 438–446, 2015.

[10] D. Brunner, M. C. Soriano, C. R. Mirasso, and I. Fischer, "Parallel photonic information processing at gigabyte per second data rates using transient states." *Nature communications*, vol. 4, p. 1364, 1 2013.

[11] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer, "Information processing using a single dynamical node as complex system." *Nature communications*, vol. 2, p. 468, 9 2011.

[12] L. Larger, M. C. Soriano, D. Brunner, L. Appeltant, J. M. Gutierrez, L. Pesquera, C. R. Mirasso, and I. Fischer, "Photonic information processing beyond Turing: an optoelectronic implementation of reservoir computing," *Optics Express*, vol. 20, no. 3, p. 3241, 1 2012.

[13] F. Duport, B. Schneider, A. Smerieri, M. Haelterman, and S. Massar, "All-optical reservoir computing," *Optics Express*, vol. 20, no. 20, p. 22783, 9 2012.

[14] A. Dejonckheere, F. Duport, A. Smerieri, L. Fang, J.-L. Oudar, M. Haelterman, and S. Massar, "All-optical reservoir computer based on saturation of absorption," *Optics Express*, vol. 22, no. 9, p. 10868, 5 2014.

[15] M. C. Soriano, S. Ortín, D. Brunner, L. Larger, C. R. Mirasso, I. Fischer, and L. Pesquera, "Optoelectronic reservoir computing: tackling noise-induced performance degradation," *Optics Express*, vol. 21, no. 1, p. 12, 1 2013.

[16] R. M. Nguimdo, G. Verschaffelt, J. Danckaert, and G. Van der Sande, "Fast photonic information processing using semiconductor lasers with delayed optical feedback: Role of phase dynamics," *Optics Express*, vol. 22, no. 7, p. 8672, 4 2014.

[17] K. Hicke, M. Escalona-Morán, D. Brunner, M. C. Soriano, I. Fischer, and C. R. Mirasso, "Information Processing Using Transient Dynamics of Semiconductor Lasers Subject to Delayed Feedback," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 19, no. 4, pp. 1 501 610–1 501 610, 7 2013.

[18] K. Vandoorne, J. Dambre, D. Verstraeten, B. Schrauwen, and P. Bienstman, "Parallel reservoir computing using optical amplifiers." *IEEE transactions on neural networks*, vol. 22, no. 9, pp. 1469–81, 9 2011.

[19] C. Mesaritakis, V. Papataxiarhis, and D. Syvridis, "Micro ring resonators as building blocks for an all-optical high-speed reservoir-computing bit-pattern-recognition system," *JOSA B*, no. October, 2013.

[20] M. A. A. Fiers, T. Van Vaerenbergh, F. Wyffels, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman, "Nanophotonic reservoir computing with photonic crystal cavities to generate periodic patterns," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 344–355, 2014.

[21] H. Zhang, X. Feng, B. Li, Y. Wang, K. Cui, F. Liu, and W. Dou, "Integrated photonic reservoir computing based on hierarchical time-multiplexing structure," *Opt. Express*, vol. 22, no. 25, pp. 31 356–31 370, 12 2014.

[22] C. Mesaritakis, A. Kapsalis, and D. Syvridis, "All-optical reservoir computing system based on InGaAsP ring resonators for high-speed identification and optical routing in optical networks," M. Razeghi, E. Tournié, and G. J. Brown, Eds., vol. 9370, 2 2015, p. 937033.

[23] K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman, "Experimental demonstration of reservoir computing on a silicon photonics chip." *Nature communications*, vol. 5, p. 3541, 1 2014.

[24] A. Katumba, M. Freiberger, P. Bienstman, and J. Dambre, "A Multiple-Input Strategy to Efficient Integrated Photonic Reservoir Computing," *Cognitive Computation*, pp. 1–8, 4 2017.

[25] S. Abel, T. Stferle, C. Marchiori, C. Rossel, M. Rossell, R. Erni, D. Caimi, M. Sousa, A. Chelnokov, B. Offrein, and J. Fompeyrine, "A strong electro-optically active lead-free ferroelectric integrated on silicon," *Nature Communications*, vol. 4, p. 1671, 2013.

[26] C. Ríos, M. Stegmaier, P. Hosseini, D. Wang, T. Scherer, C. Wright, H. Bhaskaran, and W. Pernice, "Integrated all-photonic non-volatile multi-level memory," *Nature Photonics*, vol. 9, no. 11, pp. 725–732, 2015.

[27] B. V. Bilzen, P. Homm, L. Dillemans, C. Su, M. Menghini, M. Sousa, C. Marchiori, L. Zhang, J. Seo, and J. Locquet, "Production of vo 2 thin films through post-deposition annealing of v 2 o 3 and vo x films," *Thin Solid Films*, vol. 591, pp. 143–148, 2015.

[28] A. Hoerl and R. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.
[29] M. Freiberger, A. Katumba, P. Bienstman, and J. Dambre, "On-chip passive photonic reservoir computing with integrated optical readout," in *Rebooting Computing (ICRC), 2017 IEEE International Conference on*, 2017.
[30] ——, "Training passive photonic reservoirs with integrated optical readout," *Unpublished, Submitted for review to Neural Networks and Learning Systems, IEEE Transactions on*.
[31] I. Djordjevic, W. Ryan, and B. Vasic, *Coding for Optical Channels*. Boston, MA: Springer US, 2010.
[32] G.-b. Huang, G.-b. Huang, Q.-y. Zhu, and C.-k. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, 2006.
[33] A. Lugnan, J. Dambre, and P. Bienstman, "Integrated pillar scatterers for speeding up classification of cell holograms," *Optics Express*, vol. 25, no. 24, p. 30526, 2017.
[34] D. Zink, A. H. Fische, and J. A. Nickerson, "Nuclear structure in cancer cells," *Nature Reviews Cancer*, vol. 4, no. 9, pp. 677–687, 2004.

Andrew Katumba Andrew Katumba was born in Masaka, Uganda in 1985. He received an M.Sc. degree in Optics and Photonics from Karlsrule Institute of Technology, Germany in 2013. He is pursuing a Ph.D. degree in Photonics Engineering at the Photonics Research Group, Gent University-imec, Belgium. His current research focuses on photonic neuromorphic architectures for high speed optical telecommunications systems. He is a student member of IEEE Photonics Society and the International Society for Optics and Photonics.

Matthias Freiberger was born in Graz, Austria, in 1983. He received a M.Sc. degree in information and computer engineering from Graz University of Technology, Graz, Austria, in 2016. He is currently pursuing a Ph.D degree at the UGent-imec IDLab, Department of Electronics and Information Systems at Ghent University. His current research focuses on training algorithms for neuromorphic devices. His research interests include scaling up neuromorphic systems, deep learning, and recurrent neural networks on chip.

Floris Laporte was born in Ghent, Belgium, in 1990. He received a joined M.Sc. degree in Photonics from Ghent University, the Free University of Brussels, Belgium and the University of St. Andrews, UK in 2015. He is currently pursuing a Ph.D. degree in Photonics Engineering at the Photonics Research Group, Ghent University-imec, Belgium. His research focuses on photonic neuromorphic computing with photonic crystal cavities.

Alessio Lugnan was born in Rovereto, Italy, in 1990. He received a M.Sc. degree in Physics with specializtion in Photonics from University of Trento, Italy. He is currently pursuing a Ph.D. degree in Photonics Engineering at the Photonics Research Group, Ghent University-imec, Belgium. His research focuses on photonic implementation of machine learning techniques for fast classification of biological cells.

Stijn Sackesyn was born in Ghent, Belgium, in 1994. He received a M.Sc. degree in Photonics from Ghent University, Belgium and the Free University of Brussels, Belgium in 2017. He is currently pursuing a Ph.D. degree in Photonics Engineering at the Photonics Research Group, Ghent University-imec, Belgium. His research focuses on photonic neuromorphic signal processing for optical communications networks.

Chonghuai Ma was born in Shenyang, China, in 1991. He received a M.Sc. degree in Photonics from Friedrich Schiller University Jena, Germany in 2016. He is currently pursuing a Ph.D. degree in Photonics Engineering at the Photonics Research Group, Ghent University-imec, Belgium. His research focuses on integrated photonic neuromorphic computing systems.

Joni Dambre Joni Dambre was born in Ghent, Belgium, in 1973. She received a degree in electrical engineering from Ghent University, Belgium, in 1996 and a Ph.D. in computer science engineering from the same university in 2003, at the Department of Electronics and Information Systems (ELIS). As a machine learning professor, she currently leads an UGent-imec research team (part of IDLab) addressing various topics related to machine learning, artificial and biological neural networks, and neuromorphic hardware.

Peter Bienstman was born in Ghent, Belgium, in 1974. He received a degree in electrical engineering from Ghent University, Belgium, in 1997 and a Ph.D. from the same university in 2001, at the Department of Information Technology (INTEC), where he is currently a full professor. His research interests include several applications of nanophotonics (biosensors, photonic information processing, ...) as well as nanophotonics modelling. He has published over 110 papers and holds several patents. He has been awarded a ERC starting grant for the Naresco-project: Novel paradigms for massively parallel nanophotonic information processing.