

# Parallel Reservoir Computing Using Optical Amplifiers

Kristof Vandoorne, *Student Member, IEEE*, Joni Dambre, David Verstraeten, *Member, IEEE*, Benjamin Schrauwen, *Member, IEEE*, and Peter Bienstman, *Member, IEEE*

**Abstract**—Reservoir computing (RC), a computational paradigm inspired on neural systems, has become increasingly popular in recent years for solving a variety of complex recognition and classification problems. Thus far, most implementations have been software-based, limiting their speed and power efficiency. Integrated photonics offers the potential for a fast, power efficient and massively parallel hardware implementation. We have previously proposed a network of coupled semiconductor optical amplifiers as an interesting test case for such a hardware implementation. In this paper, we investigate the important design parameters and the consequences of process variations through simulations. We use an isolated word recognition task with babble noise to evaluate the performance of the photonic reservoirs with respect to traditional software reservoir implementations, which are based on leaky hyperbolic tangent functions. Our results show that the use of coherent light in a well-tuned reservoir architecture offers significant performance benefits. The most important design parameters are the delay and the phase shift in the system's physical connections. With optimized values for these parameters, coherent semiconductor optical amplifier (SOA) reservoirs can achieve better results than traditional simulated reservoirs. We also show that process variations hardly degrade the performance, but amplifier noise can be detrimental. This effect must therefore be taken into account when designing SOA-based RC implementations.

**Index Terms**—Integrated optics, optical neural networks, photonic reservoir computing, semiconductor optical amplifiers, speech recognition.

## I. INTRODUCTION

RESERVOIR computing (RC) is a training concept for recurrent neural networks (RNNs), introduced a few years ago, that combines the advantages of both recurrent and feed forward networks [1], [2]. In the RC framework, a randomly initialized RNN, the *reservoir*, is excited by an input signal. The network itself is left untrained but the responses

of all neurons to the input stimulus are combined through a simple, often linear readout function. It can be optimized (*trained*) with simple and well-established methods such that its output matches the required output signal as closely as possible according to some error metric. In this way, as only the readout is changed, the difficulties of training a RNN are avoided. Hence, the RC approach combines the memory and spatio-temporal processing of RNNs with the ease of training of linear regression. One way of explaining the success of RC is to view the reservoir as performing a high-dimensional spatio-temporal pre-processing or filtering of the input. In this view, the reservoir essentially performs a nonlinear mixing of the input signals into a high-dimensional feature space, so that the interesting features are more easily extracted by the readout. RC has been demonstrated to give similar results as state-of-the-art techniques for several complex machine learning tasks like speech recognition [3] and event detection in robotics [4] or to outperform them as with the prediction of the Mackey-Glass chaotic time series, where RC reaches a prediction accuracy several orders of magnitude better than classical methods [1].

Although the reservoir itself remains untrained, its performance depends critically on its dynamical regime. Optimal performance is usually obtained near the edge of stability, i.e., the region in between stable and unstable or chaotic behavior, because this regime optimizes the system's memory [5]. This region is determined by the total amount of gain and loss in the network. Hence, to obtain good performance, we need to be able to tune a reservoir's dynamic regime to this edge of stability, using a small number of global parameters. However, this optimum is generally task-dependent.

Reservoirs are intrinsically parallel structures. However, most implementations thus far have been software-based, leading to extensive simulation times for large reservoirs and long input time series. A fully parallel implementation, where all neuron states are computed simultaneously, would drastically speed up computation times and might also lead to power savings. Like a neural network, a reservoir often consists of a large number of interconnected nonlinear nodes. Therefore, existing hardware implementations of neural networks can (and have been) used as reservoirs [6]. However, in contrast to traditional neural networks, the interconnection *weights* (typically translated into gains or losses in an analogue hardware implementation) do not need to be tunable or even exactly controllable. Only a global gain scaling is required. This makes the requirements for reservoir implementations more relaxed

Manuscript received December 6, 2010; revised April 4, 2011; accepted June 24, 2011. Date of publication July 29, 2011; date of current version August 31, 2011. This work was carried out in the framework of the International Academic Projects Project Photonics of the Belgian Science Policy and the European Research Council Project NaResCo.

K. Vandoorne and P. Bienstman are with the Photonics Research Group, Department of Information Technology, Ghent University-Interuniversity Microelectronics Center, Ghent 9000, Belgium (e-mail: kristof.vandoorne@intec.ugent.be; Peter.Bienstman@UGent.be).

J. Dambre, D. Verstraeten, and B. Schrauwen are with the Computer Systems Laboratory, Electronics and Information Systems Department, Ghent University, Ghent 9000, Belgium (e-mail: joni.dambre@UGent.be; david.verstraeten@ugent.be; benjamin.schrauwen@ugent.be).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TNN.2011.2161771

and allows for the exploration of technologies that are may be less suitable for implementing traditional, fully trainable, neural networks. Recent work has indicated that, besides interconnected networks of sigmoid or spiking neurons, a wide range of sufficiently high-dimensional nonlinear dynamical systems can be used as a reservoir [6]–[10].

Photonics is an interesting candidate technology for building reservoirs, because it offers a range of different nonlinear interactions working on different timescales. It also offers the promise of being more power efficient than electronic implementations, especially for processing high-bandwidth signals. In a previous paper [11], we have proposed a specific integrated photonic reservoir implementation, consisting of a network of coupled semiconductor optical amplifiers (SOAs). Although this technology does not yet fulfill the promised power efficiency or high speed, it presents a suitable platform for evaluating the differences between a photonic implementation and traditional software implementations. In particular, the topology and the behavior of both neurons and interconnections differs on several points from what is usually modeled in simulation. Still, for a simple classification task—distinguishing between a rectangular and triangular waveform—we found comparable performance to a traditional software reservoir implementation.

The goal of this paper is threefold. First, the performance of SOA reservoirs is evaluated on the much harder isolated digit recognition task with babble noise, second, the impact of most important design parameters is investigated and finally the impact of fabrication accuracies is addressed. This paper is structured as follows. In Section II, the concept of reservoir computation is reviewed. In Section III, our photonic implementation is presented, the most important differences with traditional reservoirs are discussed and the relevant design parameters are indicated. Section IV describes the speech task used throughout this paper and the results for this task with a traditional RC setup. Section V discusses the performance of our implementation for the speech task and finally, Section VI summarizes the most important results of this paper.

## II. RC

Reservoirs exist in many flavors, the most common of which use either analogue sigmoid neurons (*echo state networks* or ESNs [1]) or spiking neurons (*liquid state machines* [2]). Our proposed photonic reservoir architecture most closely resembles an ESN, so that's what we will use as a baseline to evaluate our implementation's performance. The ESN paradigm was originally formulated in discrete time through the reservoir state update function

$$\mathbf{x}[t + \Delta t] = (1 - \lambda)\mathbf{x}[t] + \lambda f(\mathbf{W}_{in}\mathbf{u}[t] + \mathbf{W}_{res}\mathbf{x}[t]). \quad (1)$$

In this equation  $\mathbf{x}[t]$  is a vector containing the states of all neurons at time  $t$  and  $\mathbf{u}[t]$  collates the values of the input stimuli<sup>1</sup> at time  $t$ . The neuron function  $f$  is usually a sigmoid function (e.g., the commonly used hyperbolic tangent or tanh function). The matrices  $\mathbf{W}$  are the weight matrices for all the

connections from the input to the reservoir ( $\mathbf{W}_{in}$ ) and inside the reservoir itself ( $\mathbf{W}_{res}$ ). The parameter  $\lambda$  is called the *leak rate* and reflects the strength with which the neuron outputs are fed back to their inputs. This neuron model, usually referred to as *leaky integrator neurons*, was introduced in [12]. It effectively adds a first-order recursive low-pass filter to every neuron. If  $\lambda$  equals one, the next state of the reservoir depends only on the external input and the current state of the reservoir. The closer  $\lambda$  is to zero, the more the history of the previous states becomes important. For many temporal tasks (such as speech recognition), it is crucial that a reservoir has sufficient memory of past inputs [4]. Using leaky integrator neurons is a way to increase this memory.

The RC-system is trained by collecting the state vectors of the reservoir in response to the training data, and using this to train the readout layer using some form of linear regression. For this paper we use linear regression with Tikhonov regularization, also known as ridge regression. If the reservoir states are concatenated into a matrix  $\mathbf{A}$ , and the target outputs are given by a matrix  $\mathbf{B}$ , then the weights for the linear readout layer are given by the following closed form:

$$\mathbf{W}_{out} = (\mathbf{A}^T \mathbf{A} + \gamma \mathbf{I})^{-1} \mathbf{A}^T \mathbf{B} \quad (2)$$

where  $\gamma$  is a regularization constant, which is optimized using cross-validation.

For a reservoir to be useful for general computation on time series, it must have *fading memory*. This means that, although the reservoir state should depend on past inputs, this impact should fade away through time. A common way to obtain this is by rescaling the reservoir weight matrix  $\mathbf{W}_{res}$  such that the reservoir is stable. The closer the reservoir dynamics are tuned to the edge of stability, the longer the system retains some information about past inputs. Tuning the reservoir dynamics close to the edge of stability has been found to yield optimal reservoir behavior for many tasks [13], [14].

A common parameter for tuning the dynamic regime of a reservoir is its *spectral radius*  $\rho$ , defined as the largest eigenvalue of the system's Jacobian at its maximal gain state. When all neurons have a maximum gain of one, as is the case with tanh neurons, this corresponds to the largest eigenvalue of the network's interconnection weight matrix. Otherwise, an upper bound can be obtained by multiplying this largest eigenvalue by the maximal gain of the neuron function  $f$ . The spectral radius is an indication of the stability of the network. If its value is larger than one, the network can become unstable. However, at each point in time, the actual gain of a nonlinear reservoir depends on its state and its input signal(s). Instability can occur if this gain on average exceeds one. As, by definition, the spectral radius gives an upper bound to the system's gain at each point in time, the edge of stability is usually found for a spectral radius slightly above one. However, tuning the spectral radius close to one often yields reservoirs with close to optimal performance. For randomly generated connection matrices, the spectral radius can be tuned to a value  $\rho_{desired}$  by scaling the entire connection matrix  $\mathbf{W}_{res}$  by a factor  $(\rho_{desired}/\rho_{W_{res}})$ .

<sup>1</sup>In many complex tasks, such as speech recognition or robot control, the reservoir is driven by a large number of signals.

### III. PHOTONIC RESERVOIR COMPUTING

This section describes our photonic reservoir architecture and highlights its differences with traditional ESNs. The three parts of this section deal with the general properties of the SOAs we use as neurons, the interconnection topology and the interconnection weights, respectively.

#### A. SOAs

SOAs were chosen as a first testbed for an integrated version of photonic RC for the following reasons.

- 1) SOAs provide gain, so no separate component is needed to compensate for losses.
- 2) They are broadband, which relaxes fabrication tolerances in comparison to resonator-type devices where the resonances of different nodes should overlap.
- 3) Their steady state characteristic somewhat resembles the upper part of a hyperbolic tangent, making them a perfect bridge between the existing knowledge of RC with tanh neural networks and the new field of photonic RC.

SOAs are the default integrated component for boosting optical signals. Their operation is similar to that of semiconductor lasers. Mostly through electrical pumping (although it can also be done optically), carriers are brought into an excited state within the gain medium. A photon can interact with these excited carriers, forcing the carrier to release its energy and returning to its ground state, while emitting a photon at the same time. This photon will be identical to the original photon having the same frequency, phase and direction. This process is called stimulated emission. The other two processes are absorption, where a photon excites a carrier from its ground state, and spontaneous emission, where an excited carrier without external stimulation falls back to its ground state while emitting a photon. All three processes occur simultaneously but stimulated emission is the dominant one only if the carriers reach a state of population inversion. This means that more carriers are in their excited state than in their ground state. Pumping is needed to obtain population inversion [15].

In a laser, feedback, with some kind of mirror, is added to reflect the light back and forth and let it pass multiple times through the gain medium. This will result in some form of wavelength selectivity, making the laser what it usually is, a coherent source. For an amplifier a broadband operation is desired, so the challenge is to remove any form of wavelength selectivity, other than that of the gain medium itself. A traveling wave SOA is an amplifier through which the light only passes once, and therefore any reflection needs to be eliminated as much as possible. This is usually done by adding a high-quality anti-reflection coating onto the facets.

The two most relevant nonlinear aspects of SOAs for RC are *gain saturation* and *carrier lifetime*. Gain saturation means that the gain will decrease for higher input powers as can be seen in Fig. 1, where the relation between input power and corresponding output power is portrayed for SOAs and the upper part of a hyperbolic tangent with the same gain for zero input. As mentioned before, the gain in an SOA originates from excited carriers being consumed by photons

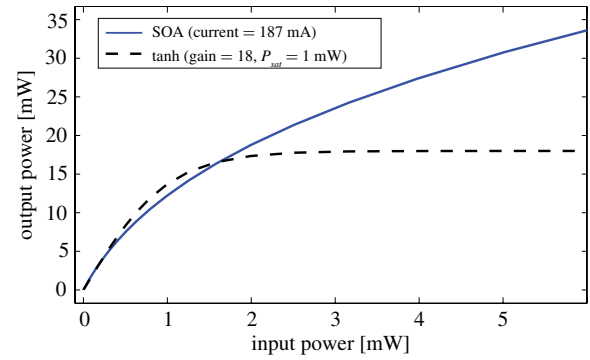


Fig. 1. Modeled power in - power out curve for an SOA and a tanh with gain (18) and power saturation (1 mW).

creating copies of themselves through stimulated emission. The more photons are inserted into the SOA, the more carriers are consumed. This decreases the density of excited carriers, leaving less opportunity for other photons to be amplified and thus diminishing the gain. This effect is the basis for the gain saturation in SOAs. The steady state characteristic of an SOA resembles a hyperbolic tangent but it does not completely saturate as can be seen in Fig. 1. In our simulations, the input of the SOAs was often in the more linear regime with input powers not exceeding 1 mW.

When carriers are stimulated into an excited state, they only ‘live’ for a certain amount of time before they relax to their ground state. The average time associated with this process is called the *carrier lifetime*, with typical values of 100–300 ps. One could say that the SOA encodes information about its past inputs into its internal state variable, the carrier density. The dynamic response of this variable to the input power resembles that of a first-order system with the carrier lifetime as a dominant time constant. This is especially important for large, abrupt changes in the input power. When such a change occurs, the gain cannot change abruptly to the corresponding value on the steady state curve and output power cannot follow instantaneously. Rather, the light will respond according to the gain at the moment of the transition, determined by the number of carriers in the gain medium. After the initial response, the gain will relax to its steady state value with the carrier lifetime as time constant, as can be seen in Fig. 2. This means that, for a sudden increase in the input, which should decrease the SOAs gain, the output will initially ‘overreact’ to the input change, followed by a relaxation as the gain approaches its lower steady state value. A similar effect occurs for a sudden decrease of the input power (Fig. 2). This dynamic carrier effect is one thing that sets the behavior of SOAs apart from a hyperbolic tangent, which is a purely static component.

The gain of SOAs is often polarization-dependent, which is an issue when light is guided by optical fibers supporting two orthogonal polarizations. Integrated waveguides in contrast, are inherently polarization specific due to their asymmetry and therefore circuits and components only work well for a single polarization [16]. This is actually an advantage when we control the polarization of the input light with polarization

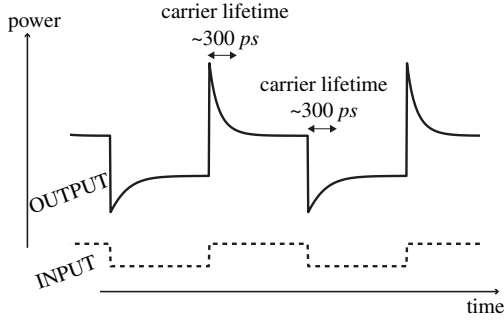


Fig. 2. Modeled response to sudden changes in the input power for an SOA.

controllers, only exciting the desired polarization, which will be maintained throughout the circuit [17]. We therefore only considered light of a single polarization.

### B. SOA Simulation Model

The unidirectional model we use in our simulations was proposed by Agrawal [18]. It allows for fast computation and captures all the basic features, such as gain saturation, carrier dynamics, and a phase shift depending on the gain.

Our implementation uses light to carry information. The input signals are translated into optical power values  $P$ , which cannot be negative, by adding the minimal input signal amplitude as a bias. Our architecture also allows the assumption of coherent light, meaning that internally all signals are characterized not by real-valued powers but complex amplitudes  $A$ .  $P(z, \tau)$  and  $\phi(z, \tau)$  are the power and phase in this complex representation  $A = [\sqrt{P} \exp(i\phi)]$  of the slowly-varying envelope of the light. When the internal losses are neglected, the output power and phase can be calculated through (3) and (4) [18]

$$P_{out}(\tau) = P_{in} \exp[h(\tau)] \quad (3)$$

$$\phi_{out}(\tau) = \phi_{in} - \frac{1}{2} \alpha h(\tau) \quad (4)$$

where  $\alpha$  is the linewidth enhancement factor responsible for the phase shift inside the SOA. The reduced time  $\tau = t - zn_g/c$  is measured in a reference frame moving with the light, where  $n_g$  is the group index denoting how much slower a light pulse travels inside the SOA. The function  $h(\tau)$  is the gain  $g(z, \tau)$  integrated over the length  $l$  of the SOA (5)

$$h(\tau) = \int_0^l g(z, \tau) dz. \quad (5)$$

Its value can be calculated by the following ordinary differential equation:

$$\frac{dh}{d\tau} = \frac{g_0 l - h}{\tau_c} - \frac{P_{in}(\tau)}{P_{sat} \tau_c} [\exp(h) - 1] \quad (6)$$

where  $\tau_c$  is the spontaneous carrier lifetime and  $P_{sat}$  the saturation power of the amplifier. So the two dominant effects of gain saturation and carrier dynamics are captured by, respectively,  $P_{sat}$  and  $\tau_c$ .  $g_0$  is the small signal gain and  $g_0 l$  is the gain for zero inputs. The values typically used are  $\alpha = 5$ ,

$n_g = 3.75$ ,  $l = 500 \mu\text{m}$ ,  $\tau_c = 300 \text{ ps}$ ,  $g_0 = 6075 \text{ m}^{-1}$ , and  $P_{sat} = 0.0211 \text{ W}$ .

The last rate equation is solved by a fixed time step-based fourth-order Runge-Kutta method.

### C. Connection Weights

In our architecture, light can be represented by either power values or complex amplitudes. For the latter case, connections are modeled as complex weights instead of pure attenuators. This can be summarized in the complex representation of optical signals at the output of a connection

$$A_{out}(t + \Delta t) = A_{in}(t) (L_c + L_d d) \exp \left[ j \frac{2\pi}{\lambda_0} n_{eff} d \right] \quad (7)$$

$$= A_{in}(t) L \exp[j \Delta \Phi] \quad (8)$$

where  $\lambda_0$  is the vacuum wavelength of the light carrying the signal,  $n_{eff}$  is the effective index—a number quantifying the phase delay per unit length in a waveguide, relative to the phase delay in vacuum.  $d$  is the physical length of the connection and  $L_c$  represents the fixed loss mechanisms (e.g., fan-out splitters, fan-in combiners, and coupling losses) and  $dL_d$  the length-dependent propagation loss in the connection.  $\Delta t$  is the propagation delay of the connection, which is again proportional to its length. We see that the input-output relationship for a connection should now be modeled by a complex-valued connection weight

$$w = L \exp[j \Delta \Phi]$$

and a delay  $\Delta t$ , whereas traditional ESN-connections are instantaneous and have real-valued connection weights. The physical length of the interconnections, affecting the loss  $L$ , phase change  $\Delta \Phi$  and time delay  $\Delta t$  in the interconnections will turn out to be an important new optimization parameter for our architecture.

The fact that we are working with complex-valued signals and weights requires an extension of the definition of spectral radius to incorporate the effects of interference. We use the magnitude of the largest eigenvalue of the complex weight matrix, which consists of the complex interconnection weights multiplied by the square root of the maximal gain in the SOAs state curve (the tangent to this curve at zero input).

Note that our readout function only uses real weights to combine the optical powers at the neuron outputs. Probably, better results could be obtained by using both power and phase. However, we are currently targeting an electronically implemented readout function. Detecting not only the power of optical signals but also their phase requires more than just a simple photodetector. Different schemes exist but any of them entail either complex optics, electronics or both.<sup>2</sup>

### D. Connection Topology

For simulated reservoirs, there are no restrictions on the interconnection topology. In classical reservoirs, the connection topology is randomly initialized. The network

<sup>2</sup>In general easier optics need more complex electronics.

can be sparsely or fully interconnected. For analog-but not spiking-networks the connectivity has little influence on the performance [14]. Photonic chips, and most other hardware implementations, are implemented on a planar surface. Furthermore, in contrast to e.g., multilayer electronic VLSI implementations, interconnections on single-layer optical chips cannot cross without some losses [typically  $-0.16$  dB in Silicon-On-Insulator (SOI) [19]]. Therefore, crossings should be avoided where possible.

Fig. 3 shows the planar type of topology used in this paper, which slightly differs from the one used in our previous work [11]. We refer to it as the *swirl* topology, because the connections are oriented as if they were in a whirling motion. As with the SOAs, we use all connections in a unidirectional way with an input and an output and neglect the influence of backscattering and backreflection. For any number of nodes that can be arranged in a rectangular grid, the topology can be constructed as follows:

- 1) order the nodes in a rectangular grid and generate all nearest neighbor connections;
- 2) all vertical connections on or to the left of the center get an upward orientation, those to the right of the center get a downward orientation;
- 3) all horizontal connections on or above the center are oriented to the right, those below the center are oriented to the left.

In all simulations, a  $9 \times 9$  network was used.

This topology uses only nearest neighbor interconnections, avoiding crossings altogether, while still providing a large number of feedback cycles. All interconnections can be implemented with comparable lengths. It can be easily scaled to larger network sizes, without affecting interconnection lengths. This allows us to treat the interconnection length as a global optimization parameter. The area and design of concrete components needed for an integrated implementation of this topology heavily depend on the material system used.<sup>3</sup>

In the resulting topology, either all nodes or only a fraction of them can be driven by the external input signal. In the experiments presented in this paper, all the nodes were driven by the external input. However, selected simulations where only 10 % of the inputs were used gave similar results (not shown). This could be relevant for an optical implementation on chip where the inner nodes of the swirl topology are harder to reach. Inputs can, however, be negative and since we are dealing with non-negative optical powers, all the inputs are made non-negative by adding the absolute minimum over all

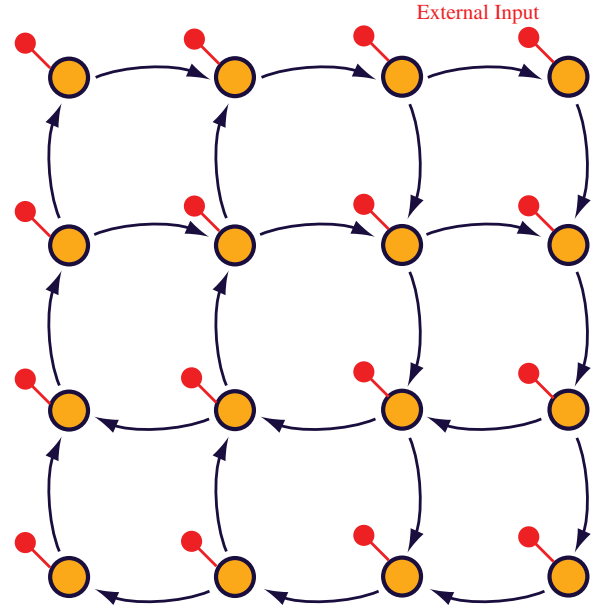


Fig. 3.  $4 \times 4$  swirl topology.

the inputs to all the inputs. Since, the inputs are only translated their shape and interrelation remains.

#### IV. SPEECH RECOGNITION

Speech recognition remains a nontrivial task to solve and neural networks have long been investigated as an alternative for standard speech recognition methods, which seem to have reached their limits. The research interest in RNNs was reinvigorated in recent years especially due to the increasing availability of computational power [22], [23]. RC, too, has proven its worth for this task [3]. The task used in this paper is the classification of isolated spoken digits ‘zero’ to ‘nine’. In the data set, these words are each spoken 10 times by 5 female speakers, giving 500 samples, taken from the TI46 speech corpus [24]. This data set, as well as a MATLAB simulation framework for classical RC<sup>4</sup> are freely available online. For speech recognition, some pre-processing of the raw speech signal is commonly performed. These methods often involve a transformation to the frequency domain and a selective filtering based on known psycho-acoustic properties of the human ear and/or spectral properties of speech. For the experiments in this paper, we used the ear model introduced by Lyon *et al.* [25]. To shorten the simulation time, a decimation of the input signals with a factor of 128 was also applied.

The output is obtained by training ten distinct linear classifiers, one for each digit. Each trained output should return the value  $+1$  whenever the corresponding digit is spoken and  $-1$  otherwise. During testing, a winner-take-all approach is used to determine which word was detected. The performance for this task will be expressed with the word error rate (WER), which is  $(N_{nc}/N_{tot})$ , with  $N_{nc}$  the number of incorrectly classified samples, and  $N_{tot}$  the total number of samples.

<sup>3</sup>On chips, light is guided in waveguides which need a core that has a higher refractive index than the surrounding cladding layers. The index contrast of a material system determines the size of the waveguides and other components. The higher the index contrast, the smaller the components and the sharper the possible bends. We have ordered a chip with 12 SOAs (3 by 4 swirl configuration) on an InP platform (<http://www.jeppix.eu/>) with low index contrast. The size of this chip is  $16 \text{ mm}^2$  with bending radii of  $100 \mu\text{m}$ . However, in a material system with high index contrast, the dimensions shrink and bending radii down to  $5 \mu\text{m}$  are possible with negligible loss ( $0.009 \text{ dB}/90^\circ$ ) [20]. Amplifiers in this material system are still in an experimental phase [21], but when it has matured, this technology will offer tenfold chip size reductions or more. If further scaling is required, a transition to smaller and more low-power components such as coupled cavities is an option.

<sup>4</sup>Available at <http://snn.elis.ugent.be/rctoolbox>.



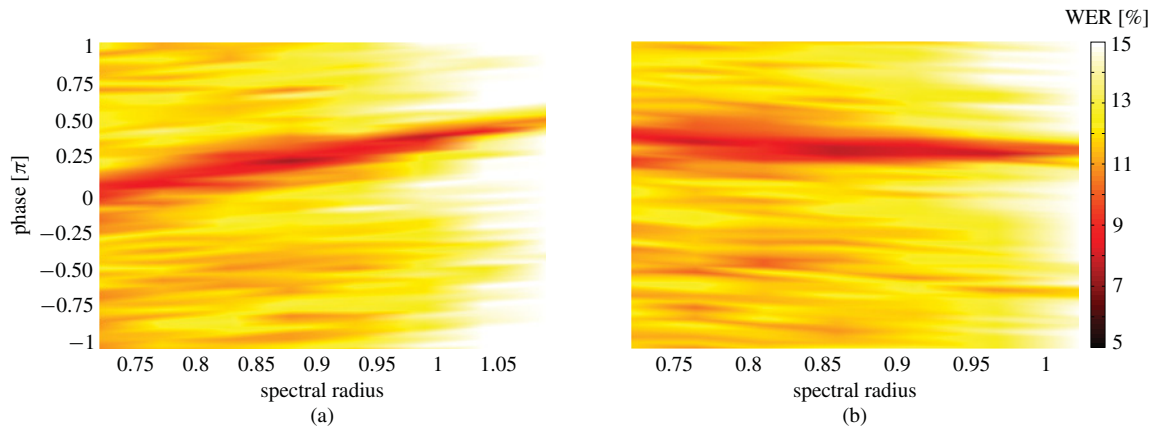


Fig. 4. WER for SOA reservoirs with coherent light and a delay of 6.25 ps. (a) Spectral radius is changed through the input current of all the SOAs. (b) Spectral radius is changed through the attenuation in all the connections.

Since it is possible to achieve a WER very close to 0%, babble noise from the NOISEX database<sup>5</sup> was added with a SNR of 3 dB. The results are always averaged over 10 runs. We also used ridge regression to avoid over-fitting and five-fold cross-validation to make our results more robust [26].

Audio signals are much slower in comparison to the typical timescales of delays in an integrated optical network. Therefore, we have accelerated the speech signal to correspond to those timescales. The delays on a chip vary with the physical length of the interconnections from a few ps to a few 100 ps, depending on the material systems and structures used. The audio samples used have a duration of about 0.5 ms. So, for the photonic reservoirs we feed them to the network 9 orders of magnitude faster, which makes for samples only lasting around 500 ps. Although we use this task to demonstrate the potential of photonic RC, we do not propose to use photonic reservoirs as a platform for standard real-time, slow audio signals.

Throughout the rest of this paper, we will compare our results to those of classical ESNs with tanh neurons. In all experiments, the output of the Lyon ear model, consisting of 77 channels, was fed into a reservoir of 81 neurons. This relatively small number of neurons was a compromise between the photonic network simulation time that increases with larger networks, and using a number of nodes surpassing the number of channels. For each neuron, the channels were mixed differently, with weights randomly alternating between  $-0.1$  and  $0.1$ . We envision this being done electronically, and then converting the resulting input signals to optical node inputs with a modulator. Our architecture allows the information to be carried by either light intensities only (incoherent light) or by light at a single wavelength (coherent light). Inter-wavelength interaction such as cross-phase modulation and four-wave mixing is not relevant in either scenario, so nonlinear effects between signals at a distinct set of wavelengths were not considered in this paper. The use of multiple wavelengths to further improve the computational power of our architecture is a topic for future work.

For classical ESN reservoirs with random interconnections, the optimal WER for this isolated digit speech recognition task

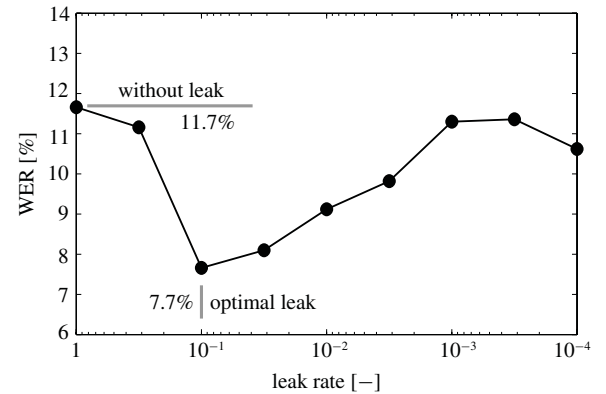


Fig. 5. WER for classical ESNs with a sweep over the leak rate parameter  $\lambda$  (1). This was done for networks with random topology and a spectral radius equal to one. Using some leak rate is clearly advantageous for this speech task. To obtain a broad range of values, a logarithmic spacing was used yielding an optimal value of 0.1 for the leak rate parameter.

with babble noise is around 11.7% if we do not use leaky integrator neurons and around 7.7% if we do and optimize the leak rate parameter (Fig. 5). The optimal value of 0.1 for the leak rate parameter was used for all leaky integrator reservoirs.

## V. PERFORMANCE ANALYSIS

### A. Tuning Parameters

For traditional RC, a number of global parameters are usually tuned to optimize an architecture for a given task. In our design, too, there are a number of parameters we can explore to optimize performance. Some of those need to be fixed at design time, while others can be used at operation time to compensate for differences between the simulation model and the actual hardware. These can be caused by inaccuracies in the model, but also by process variations or temperature effects.

At operation time, we can still tune the system's gain (spectral radius  $\rho$ ) and the phase shift  $\Delta\Phi$  in each connection. The gain can be altered by adapting the input current of the SOAs. There are two ways to optimize the phase shift and both can be understood from (7) and the definition of  $\Delta\Phi$ .

<sup>5</sup>Available at [http://spib.rice.edu/spib/select\\_noise.html](http://spib.rice.edu/spib/select_noise.html).

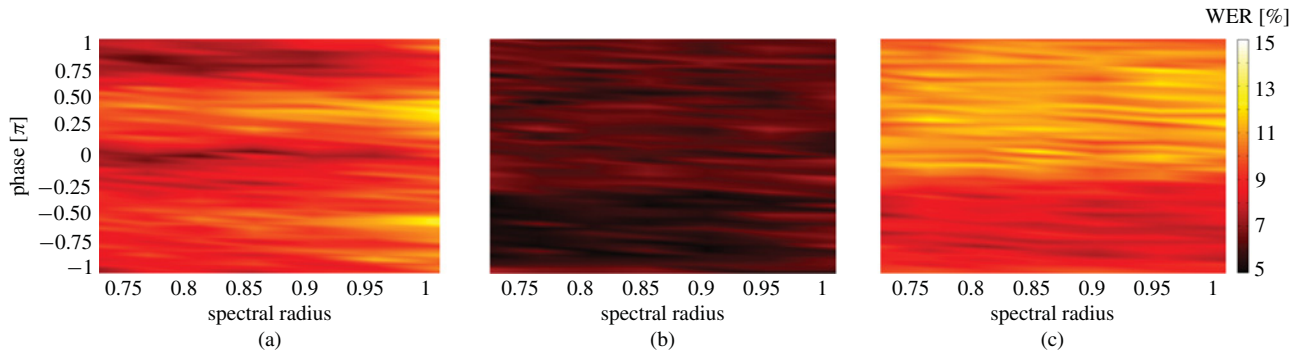


Fig. 6. WER for SOA reservoirs with coherent light and different delays. (a) delay = 75 ps. (b) delay = 187.5 ps. (c) delay = 312.5 ps.

Changing the wavelength is the easiest approach, although the requirements for the wavelength tunability become stricter for longer interconnections as the wavelength steps have to be smaller to scan the entire  $[-\pi, \pi]$  range with the same accuracy. Another approach is to actively tune the effective index  $n_{eff}$  in a section of the waveguide to achieve the required phase shift. This depends only on the length of the tuning section and not on total connection length as for the wavelength. This can be done for example with carrier injection or temperature. The exact mechanism for sweeping the phase does not matter for our experiments.

At design time, we fix the physical properties of the SOAs, the length of the interconnections and the overall loss in the system. For most of our experiments, we have evaluated each set of physical design parameters by sweeping the tunable parameters  $\rho$  and  $\Delta\Phi$ . Note that the input current not only affects the gain, but also the phase change at the output of the SOAs (Equation 4). This phase interacts with the phase change of the interconnections, so input current and interconnect phase are not orthogonal parameters. This is illustrated in Fig. 4(a), showing a  $(\rho, \Delta\Phi)$  plot, averaged over 10 runs, of the performance (color-coded WER) of a SOA reservoir with the swirl topology and minimal-length interconnections. We can see that the optimal (dark) region is slanted. We have therefore swept the spectral radius by adjusting the attenuation in the connections [shown in Fig. 4(b)]. This simplification makes our results easier to interpret without affecting the conclusions to be drawn from them.

Several things can be concluded from the plot in Fig. 4. First, our extended definition of the spectral radius for complex-valued systems has a similar impact on system performance as the spectral radius in traditional RC. Indeed, for this application, we see that the results start to deteriorate strongly for a spectral radius above one. Comparing Fig. 4 with the baseline WER of 7.3%, we can conclude that, for most of the parameter space, this particular network of SOAs performs considerably worse than traditional reservoirs with leaky tanh neurons. However, for a relatively narrow region of the parameter space, the performance improves and approaches that of the leaky tanh reservoirs. Unfortunately, whether or not this optimum can be reached depends on the accuracy with which we can control the phase change in the interconnections. As this control is in practice not straightforward (sensitive to temperature and on-chip noise),

a good architecture is one that shows a relatively broad  $(\rho, \Delta\Phi)$ -region with high performance.

### B. Optimized Interconnection Delays

In simulated reservoirs, time is usually discretized. At each simulated time step, all neuron states are updated using the previous neuron states and a new input sample. One could view this as the neurons having a single time step delay and the interconnections being instantaneous. In contrast, in our photonic implementation, time is continuous and no synchronization mechanism is used. The input signals are continuous signals, only discretized for simulation purposes. The SOAs have a certain propagation delay and so have the interconnections between them.

Fig. 6 shows the results of  $(\rho, \Delta\phi)$  parameter sweeps for the SOA reservoir using three different interconnection delay values. It shows that performance can be optimized by sweeping the interconnection length across a wide range of values. Also, for longer delays, the performance becomes less sensitive to the phase change in the interconnections, making the design more robust.

Thus far, very little is known about the impact of interconnection delay in traditional reservoirs. In this section, we will investigate its impact in tanh reservoirs, operating in continuous time. For this analysis, we will assume that all interconnections have equal delay. We have analyzed tanh reservoirs with random and swirl topology, as well as with and without leak rate. For random networks the inputs and weights can be negative as is the case for typical reservoirs, while the swirl tanh networks only have positive inputs and weights just like SOA networks. The results are summarized in Fig. 7. For each delay value, we report the best score found in a spectral radius sweep.

First, we see that the differences related to positive input and topology are negligible, both for tanh networks with and without leak-except for very small delays. When leak rate is used, the swirl topology performs even slightly better than random networks. This confirms the results of our previous paper [11], namely that the performance impact of using a planar topology is small in ESN reservoirs. It is far less significant than the impact of memory-optimizations like the introduction of leak rate to the neurons. In the remainder of this paper, we will therefore only compare the performance of our SOA

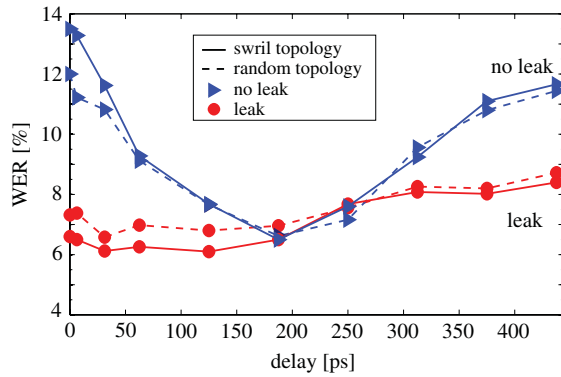


Fig. 7. Performance of classical tanh networks with random and swirl topology differs only slightly, while the presence of leak matters a lot. For reservoirs without leak rate there exists a delay for which the performance is optimal.

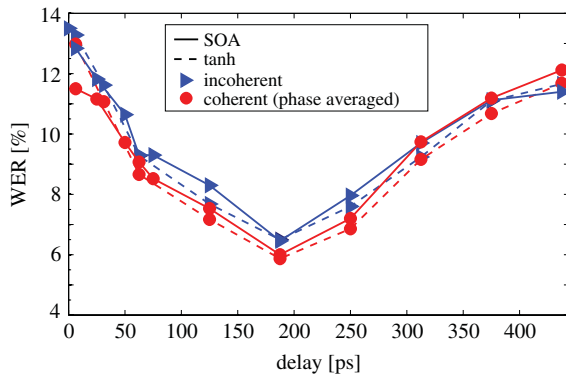


Fig. 8. Incoherent versus coherent tanh and SOA networks where the coherent results are first averaged over the phase. All reservoirs had a swirl topology with positive weights and leak was not used. In this case, there is hardly any improvement using coherent reservoirs.

reservoirs to that of tanh networks with positive weights and the swirl topology. This implies that later comparisons between swirl SOA and tanh networks will only differ in their node functionality.

Second, Fig. 7 shows that the optimal delay is not just a feature of SOA networks but also of tanh networks without leak. Reservoirs with leak rate are initially hardly affected by longer delays, but their performance decreases as well for very long delays, albeit slower than without leak. The result of a tanh network without leak rate at its optimal delay, is actually comparable to a tanh with leak rate and no (or minimal) delay. This reinforces the view that delay is an alternative approach to introducing memory next to leak rate. Unpublished experiments with a simple signal classification task where a square and a triangular wave need to be distinguished also identified delay as a very important parameter for improving performance. For this task too, an SOA network with optimal delay achieved similar results to a classical tanh network with leak rate.

For this application, the optimal performance can always be found around a delay of 190 ps, which roughly corresponds to half of the duration of the audio signals. Selected experiments with other tasks (results not shown) indicate that, there too, an optimal delay exists. Clearly, the interconnection length needs

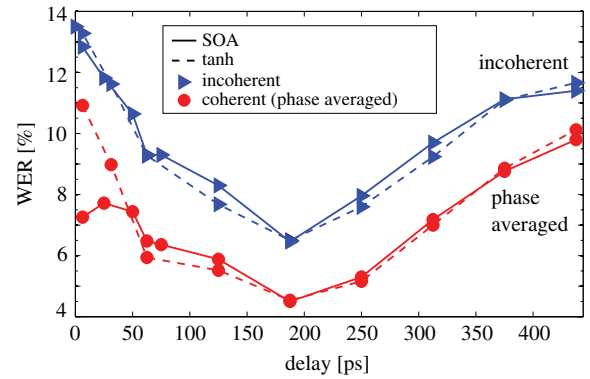


Fig. 9. Incoherent versus coherent tanh and SOA networks. For the coherent networks, the optimal value of the phase was used (phase controlled). All reservoirs had a swirl topology with positive weights and leak was not used. In this case, there is a clear improvement using coherent reservoirs.

to be optimized for the task at hand as part of the design effort and cannot be changed after fabrication. In SOA implementations, long delays can be realized compactly by using spiraling waveguides.<sup>6</sup> Alternatively the input can be resampled to match the delays, but further research is required to investigate to what extent these two approaches are equivalent. Furthermore, since there are limits to how fast you can resample the input and how long you can make the connections, a combination of both approaches will probably be needed.

### C. Coherence

Our architecture can handle light at one wavelength coming from a laser, but it can also handle light from an incoherent source such as a superluminescent light-emitting diode (SLED).<sup>7</sup> In the first case, the system must be modeled as coherent, i.e., using complex values for all state variables and connection weights. This implies that the number of internal state variables in coherent reservoirs is twice the number of nodes, but only half of them (the power values) are being used for the readout. In the second case, the light is represented by power values instead of complex amplitudes. In this section, we quantify the impact of using coherent light on the performance of our SOA reservoir implementation. We will systematically refer to coherent and incoherent simulations, respectively, to make the distinction between both.

Plots like the ones in Fig. 6 are very informative to qualitatively compare simulations for coherent reservoirs, but for a quantitative study we extract two values from these

<sup>6</sup>The longer the spiral, the higher the losses and the higher the SOA gain should be to compensate for these losses. In SOI spirals of 5 cm with a loss of 3 dB/cm have been used, corresponding to delays around 625 ps. This is longer than what we have used in this paper (~500 ps) [20]. However, recent improvements have reduced the waveguide loss to around 0.3 dB/cm, a tenfold reduction, allowing for smaller SOA gains for the same spiral length [27].

<sup>7</sup>We did not use an actual laser or SLED source in our simulations. We explored the two extremes: perfect incoherent light which is broadband and where only intensities count, and perfect coherent light at one wavelength where light can be represented by complex amplitudes. The first can be approximated by using broadband sources such as SLEDs, the second by single mode lasers. Our experiments could be done for light around 1550 nm or 1300 nm where SOAs, lasers and SLEDs with bandwidths around 50 nm, are readily available.



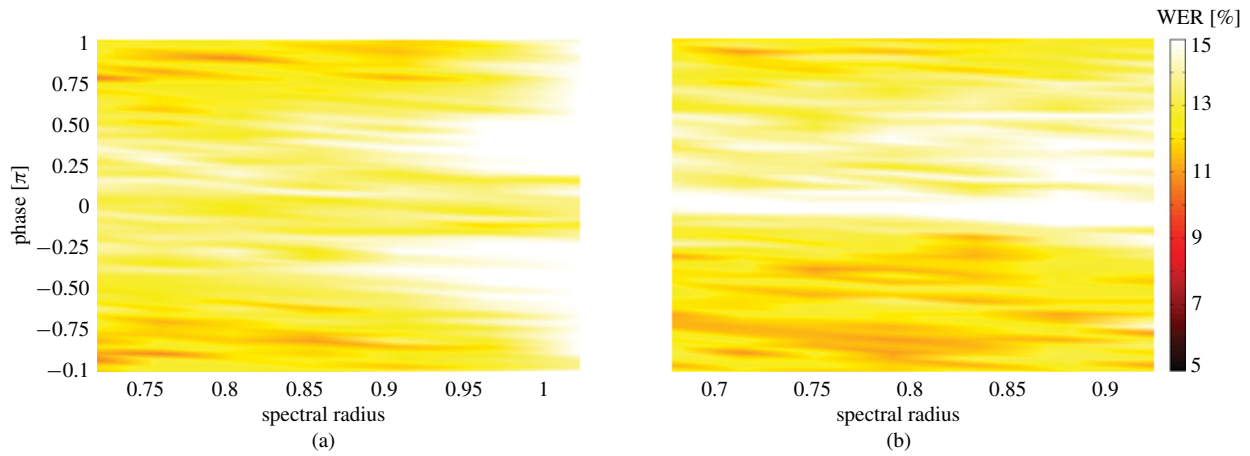


Fig. 10.  $(\rho, \Delta\Phi)$  plots for coherent networks with short delays. (a) SOA reservoir without gain-dependent phase shift. (b) tanh reservoir.

sweeps. The first value is the minimal result in a  $(\rho, \Delta\Phi)$  plot, reflecting a situation of perfect phase control of the interconnections. For the second value, the  $(\rho, \Delta\phi)$  results are first averaged over all the phase values for each value of the spectral radius. Then, the minimal value over all the spectral radii is picked, corresponding to the average performance when no phase control is available, but the spectral radius is optimized.

The influence of coherence is addressed in Figs. 8 and 9 where coherent and incoherent swirl reservoirs are compared using tanh and SOA nodes. Fig. 8 shows the average performance without phase control, whereas Fig. 9 shows results for reservoirs with perfect phase control. Both figures show that the use of SOA nodes or the addition of coherence does not alter the conclusions of the previous section. Indeed, in both figures, all four curves roughly follow the same trend and show an optimum for the same interconnection delay.

In Fig. 8, the four curves are very close. Without phase control, we observe only a minor performance benefit for the coherent simulations for all delay values except for the very smallest. As Fig. 9 shows, a perfect control of the phase drastically increases the performance of coherent reservoirs, both with tanh and with SOA neurons: the coherent results are shifted downward by about 2%. In the remainder of this paper when we refer to coherent results, perfect phase control will be assumed unless stated otherwise.

We can also conclude from Figs. 8 and 9 that the replacement of tanh nodes by SOA nodes barely affects the performance, except in the coherent simulations for small interconnection delays, where SOA nodes are clearly beneficial. The rather good optimal performance of SOA networks with short delays, already shown in Fig. 4(b), turned out to be for a very narrow region in  $(\rho, \Delta\Phi)$  space. This optimal region is completely absent for a coherent tanh swirl network with a short delay as can be seen in Fig. 10(b). The effect responsible for this difference is the gain-dependent phase shift at the output of the SOA nodes, scaled by the linewidth enhancement factor  $\alpha$  in (4). If we perform a  $(\rho, \Delta\phi)$  parameter sweep for coherent SOA networks, but artificially switch off the gain-dependent phase shift by setting  $\alpha$  to zero, we find no well-performing parameter regions [Fig. 10(a)]. The minimum

results for these 3 situations tell the same story (Table I). Due to the influence of the carrier lifetime  $\tau_c$  in (6), the gain also represents a memory mechanism in the reservoir nodes, similar to leak rate. For networks without leak rate or sufficiently large delay, this third mechanism slightly improves the performance for SOA reservoirs. By adding phase control, the interaction between the phase shift and the coherent interaction of incoming signals can be optimized, so the impact increases. However, in contrast to leak rate, this effect is not strong enough to dominate other memory optimizations. For longer delays (over 50 ps) the advantage disappears (Fig. 9) as the impact of the phase shift becomes negligible.

In general, we find that besides interconnection delay, coherence, and phase shift for small delays, all other behavioral differences between our implementation and traditional tanh reservoirs, e.g., the carrier dynamics and gain saturation, have a negligible influence on performance for the input signal properties considered in these experiments. Indeed, in our experiments, all nodes were operating mostly in their low-power regime, for which the steady state curves for tanh and SOA are very well matched and almost linear (Fig. 1). The bandwidth of the input signals was much lower than the frequency range of the node dynamics. The dynamic range of the node states can be changed by scaling up the power of the input signals. Selective experiments (not shown) showed that results did not vary a lot for small input powers. When increasing the power, a weak optimum was reached, before results deteriorated heavily. Because the improvement in the optimum was small and adding an extra sweep parameter consumes a lot of simulation time and resources, this parameter was not varied.

Fig. 11 compares the optimal results achieved with classical networks (with leak) and SOA reservoirs (with and without coherence). Since the two most important parameters, delay and phase control, appear to be orthogonal (i.e., they can be optimized independently), the best design strategy consists of optimizing the delay first and providing a phase tuning mechanism. Under these conditions, SOAs with coherence at an optimal delay perform better than classical networks with leak rate, not only at the optimal delay but also for a relatively

TABLE I

MINIMUM RESULTS FOR A COHERENT tanh SWIRL NETWORK, A COHERENT SOA NETWORK AND A COHERENT SOA NETWORK WITHOUT GAIN-DEPENDENT PHASE CHANGE. ALL HAVE AN INTERCONNECTION DELAY OF 6.25 ps

	minimum
SOA swirl coherent	7.2 %
SOA swirl coherent without phase change	10.4 %
tanh swirl coherent	10.9 %

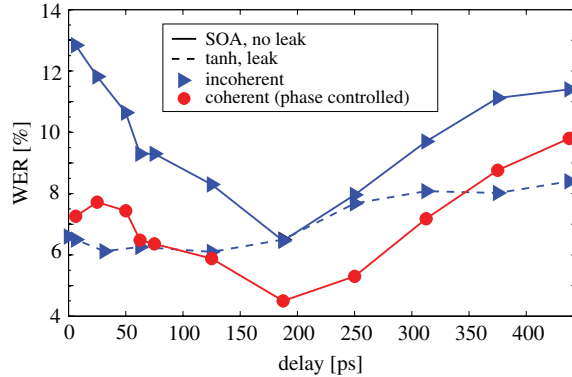


Fig. 11. Incoherent and coherent phase controlled SOA networks versus classical networks with leak rate. All reservoirs used a swirl topology with positive weights.

wide range of delay values around the optimal delay. This is also summarized in Table II showing that optimal coherent SOA reservoirs achieve error rates that are approximately 1.5% lower. Although incoherent SOA networks are outperformed by coherent SOA networks, making their performance comparable to that of traditional leaky tanh networks, the measurement, and fabrication tolerances are considerably more relaxed when incoherent light is used, making this option an interesting candidate for an initial small-scale prototype.

#### D. Process Variations

The previous simulations were done under the assumption that all connections had the same length and hence they respond in the same way to, e.g., a wavelength shift. On a real chip this is never the case, since small process variations always occur. In Fig. 12(a), we show results of a  $(\rho, \lambda_0)$  parameter sweep of a coherent SOA network with an interconnection delay of about 50 ps (i.e., an interconnection length of about 5 mm, depending on the material system). This corresponds roughly to the delays on a chip prototype that is currently being fabricated. In such a design, process variations of about 100 nm are realistic. This means that, for this design, the variations are 4–5 orders of magnitude smaller than the total length of the connections and the resulting time differences because of this are negligible. The phase variations, however, are not. Looking back at (7), the interconnection phase shift is determined by the vacuum wavelength  $\lambda_0$ , the interconnection length  $d$ , and the effective index  $n_{eff}$ , which is wavelength-dependent. If  $d$  is the same for all the connections as in Fig. 12(a), the phase change for every connection will

TABLE II

MINIMUM RESULTS AT THE OPTIMAL DELAY FOR COHERENT AND INCOHERENT SOA SWIRL NETWORKS AND tanh SWIRL NETWORKS WITH LEAK RATE

	minimum
incoherent tanh with leak	6.1 %
incoherent SOA	6.5 %
coherent SOA	4.5 %

TABLE III

MINIMUM RESULTS FOR NETWORKS WITH 0 nm AND 100 nm STANDARD DEVIATION ON THE CONNECTION LENGTHS

$\sigma$	phase controlled	averaged over the phase
0 nm	7.3 %	9.7 %
100 nm	7.9 %	9.6 %

also be the same, but with length variations the impact of wavelength is felt differently for each interconnection.

Fig. 12(b) shows the impact of interconnection length variations on the performance results obtained by a  $(\rho, \lambda_0)$  parameter sweep. Except for variations on the nominal interconnection length, the setup for generating this figure was the same as for Fig. 12(a). The length variations were randomly selected from a zero-mean Gaussian distribution with a standard deviation of 100 nm. For all individual simulation runs, the same length values were used as would be the case for a practical implementation: once a chip is made, the fabrication imperfections remain the same. The y-axis for both figures shows the same wavelength range. For (a) this corresponds with the same phase scan as in previous simulations  $[-\pi, \pi]$ , but for (b) the phase scan will be different for every connection since the length variations will result in random phase additions.

From visual comparison of both figures, we can conclude that the performance has become less phase-dependent. The optimal regions have become smaller and the poorly performing phase regions have almost entirely disappeared. One would therefore expect the minimum result with length variations to be worse (decrease in optimal performance), but the minimum results after averaging over the phase (wavelength) results to be better (increase in robustness). Although Table III confirms this intuition, the differences are fairly small, so we can conclude that small length variations do not severely affect reservoir performance in SOA networks.

#### E. Noise Robustness

In the previous sections, all the SOA simulations were done without any *amplified spontaneous emission* (ASE), to focus purely on the effects of the major design parameters. In this section we will address the influence of ASE. It is the main source of noise in SOAs and comes from the spontaneous relaxation of excited carriers and subsequent emission of photons whose frequency, phase, and direction are for the most part not matched to those of the incoming photons. In practice, ASE can be very strong, severely degrading the signal quality. Although complicated models exist where the ASE is modeled through separate rate equations, we used a simpler

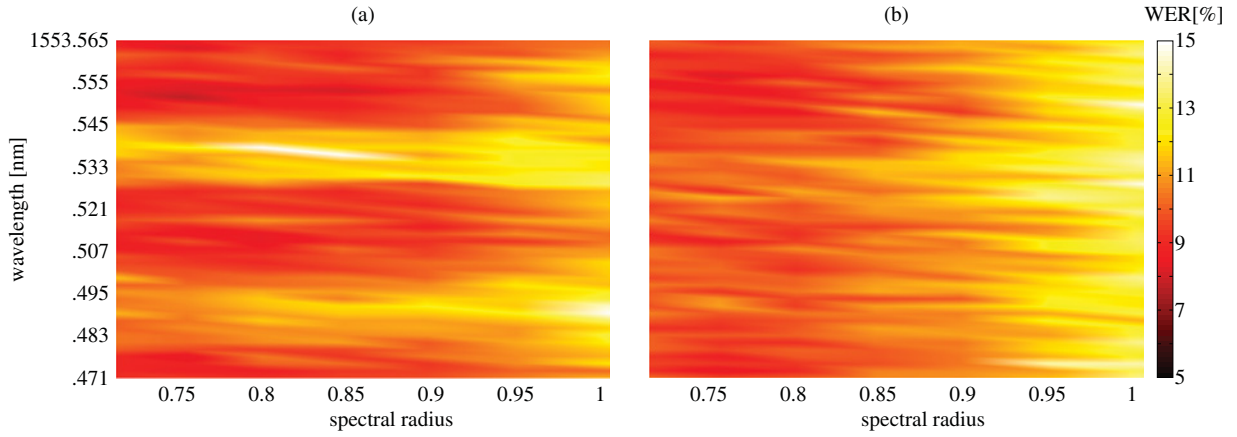


Fig. 12. Two reservoirs with delay = 50 ps. (a) All interconnections have the same length. (b) All interconnections have small length variations with a standard deviation of 100 nm (the y-axis shows the wavelength, which affects the phase change in all the connections).

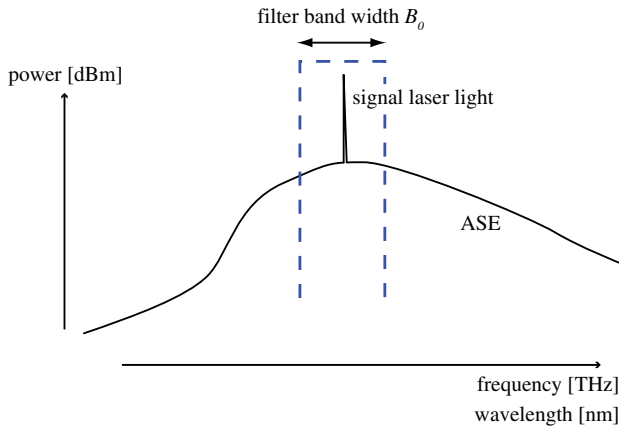


Fig. 13. Drawing of the measured output spectrum of an SOA with a single laser input signal. The sharp peak corresponds with the amplified input, superimposed on a background of ASE. The blue dashed curve shows an ideal bandpass filter used to filter out the amplified input signal. The y-axis has a logarithmic (dB) scale, the x-axis is in frequency or wavelength units.

model where random complex amplitudes are added to the SOA outputs (appendix).

Fig. 13 illustrates the output spectrum of an SOA receiving light at wavelength  $\lambda_0$  at its input. The input is amplified, but there remains a background of amplified light at other wavelengths, i.e., ASE. The power level of this ASE increases with the input current, but if we keep that fixed, Fig. 13 shows two ways to decrease the influence of ASE noise and improving the signal to noise ratio (SNR). The first is to increase the output signal power by increasing the input signal power as this will not affect the ASE power. This approach is limited by the gain saturation. A second approach is to limit the bandwidth of the ASE, since the total amount of ASE power is determined by the width of its spectrum. The broader the spectrum, the higher the power. If we cut away a part of this spectrum with a bandpass filter with bandwidth  $B_0$  around the signal wavelength  $\lambda_0$ , the ASE power is reduced, increasing the SNR.

For the experiments, we took a coherent SOA network with swirl topology and optimal delay, and selected the optimal parameter combination from a  $(\rho, \Delta\Phi)$  sweep without ASE.

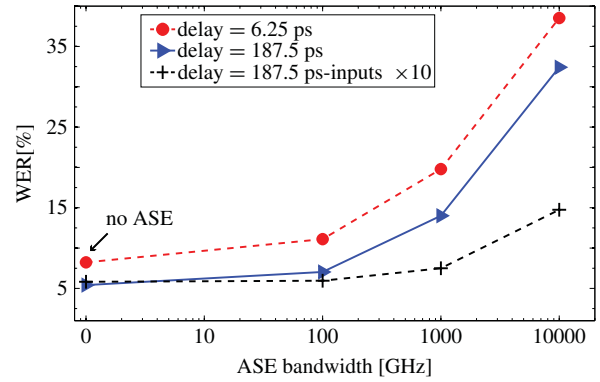


Fig. 14. Different intensities for ASE noise are compared for two different coherent SOA swirl reservoirs—one with a short delay, the other with the optimal delay. It shows the mean of all the results over 1000 runs. The left value actually corresponds to no ASE and was added afterward to the log plot, the right value to unfiltered ASE. For the network with optimal delay, a situation where the input is scaled 10 times higher is also shown.

The input scaling was the same as in the previous experiments with input powers always below 1 mW. This network was used in Monte Carlo simulations with 1000 runs in which the mixing of the inputs and the cross-validation were varied from run to run. The results are shown in Fig. 14 for different ASE powers (i.e., different filter bandwidths  $B_0$ ). The left part of the graph corresponds to the unrealistic case of no ASE (put at 1 on the logarithmic x-axis for easier comparison), the right part of the graph corresponds to unfiltered ASE.

It is obvious that the effect of ASE can be detrimental to the performance of the reservoir and that careful considerations are needed to mitigate its influence. The two approaches discussed are both shown in Fig. 14. In case the inputs are scaled ten times higher, the degradation for unfiltered ASE is much lower. Using increasingly narrower filters up to 100 GHz<sup>8</sup> also limits the degradation. In practice some kind of filtering will be necessary to achieve the optimal performance.

<sup>8</sup>100 GHz is a filter bandwidth that is for example used in the G.694.1 standard of the International Telecommunication Union for dense wavelength division multiplexing channel spacings (<http://www.itu.int/rec/T-REC-G.694.1-200206-I/en>). This standard is used to determine which wavelengths and their spacings are to be used to send information over optical fibers. The same standard specifies channel spacings up to 12.5 GHz.

Filters, however, complicate fabrication since all of them will often have to be tuned or trimmed afterward to make them operate at the same filter wavelength. Alternative components for photonic reservoirs such as nonlinear coupled cavities, however, will not be susceptible to ASE, but they are a topic of future research.

## VI. CONCLUSION

This paper has presented a detailed analysis of the performance of a photonic reservoir implementation, based on integrated SOAs. We have systematically linked design parameters and behavioral properties of our implementation to the observed performance trends. One of the most important design parameters turned out to be the delay of the inter-node connections, a parameter that has thus far hardly been considered in traditional reservoir research. We have shown that the impact of this parameter is also found in traditional reservoirs without leak rate. A second important parameter is the phase shift in the interconnections. If this can be controlled, a significant performance improvement is achieved. Other design parameters were found to have less impact on performance. When both interconnection delay and phase shift can be optimized, our SOA implementation outperforms optimized traditional tanh reservoirs with leak rate. If the phase shift cannot be tuned, we still find an optimal performance similar to that of traditional simulated reservoirs. The influence of noise in the amplifiers and fabrication variations was also investigated. While process variations do not turn out to be very important, careful considerations to limit ASE will be necessary.

Although our experimental section offered an in-depth analysis for one specific task (recognizing isolated spoken digits with babble noise), selective experiments on other tasks (not shown in this paper) have indicated that similar conclusions would follow for other tasks. This claim will be validated extensively in future research, both through simulation and on a small-scale implementation. The importance of our work is not only in evaluating this specific SOA architecture. Indeed, our conclusions can be generalized to other coherent photonic implementations with similar node properties. In particular, we target integrated nanoscale implementations that are optimized for speed and/or power consumption.

## APPENDIX

### ASE SIMULATION MODEL

The ASE model is based on dividing the ASE spectrum with bandwidth  $B_0$  from Fig. 13 into bins with width  $\delta\nu$ , around a central frequency  $\nu_0$  or central wavelength  $\lambda_0$ . The spectral components that correspond with these bins have a certain power and random phase and are added to the output field of the SOA [28]. The number of bins is  $2M + 1 = B_0/\delta\nu$ . In our simulations we varied  $B_0$ . This can be done in practice by adding a filter after the SOA (Fig. 13), and kept the number of bins constant at 75. The total spontaneous emission power is given by

$$P_{sp} = n_{sp}(G - 1)h\nu B_0 \quad (9)$$

where  $n_{sp}$  is the noise figure of the amplifier,  $h$  Planck's constant, and  $G = \exp(h)$  is the gain of the SOA. Since, this expression is not dependent on the power of the input signal, one can diminish the relative importance of ASE by using higher input powers.

The field for the spontaneous emission added to the SOA output field is then described by the following equation:

$$E_{sp}(t) = \sum_{m=-M}^M \sqrt{n_{sp}(G - 1)h\nu_0\delta\nu} \exp(j(m2\pi\delta\nu t + \Phi_m)) \quad (10)$$

where  $\Phi_m$  is a random phase for each component of the spontaneous emission and its value is taken from a uniform random distribution with interval  $[-\pi, \pi]$ , the rest of the expression inside the exp is determined by the distance between the spectral component and the central frequency. The values typically used are  $\nu_0 = 193.1$  THz and  $n_{sp} = 2$ .  $B_0$  was varied as in Fig. 14.

## ACKNOWLEDGMENT

K. Vandoorne acknowledges the Special Research Fund of Ghent University, Ghent, Belgium, for a specialization grant.

## REFERENCES

- [1] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *Science*, vol. 304, no. 5667, pp. 78–80, Apr. 2004.
- [2] W. Maass, T. Natschl ger, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," *Neural Comput.*, vol. 14, no. 11, pp. 2531–2560, Nov. 2002.
- [3] M. D. Skowronski and J. G. Harris, "Automatic speech recognition using a predictive echo state network classifier," *Neural Netw.*, vol. 20, no. 3, pp. 414–423, Apr. 2007.
- [4] E. Antonelo, B. Schrauwen, X. Dutoit, D. Stroobandt, and M. Nuttin, "Event detection and localization in mobile robot navigation using reservoir computing," in *Proc. Int. Conf. Artif. Neural Netw.*, Porto, Portugal, Sep. 2007, pp. 660–669.
- [5] R. Legenstein and W. Maass, "What makes a dynamical system computationally powerful?" in *New Directions in Statistical Signal Processing: From Systems to Brains*, S. Haykin, J. C. Principe, T. Sejnowski, and J. McWhirter, Eds. Cambridge, MA: MIT Press, 2007, pp. 127–154.
- [6] D. Verstraeten, S. Xavier-de-Souza, B. Schrauwen, J. A. K. Suykens, D. Stroobandt, and J. Vandewalle, "Pattern classification with CNNs as reservoirs," in *Proc. Int. Symp. Nonlin. Theory Appl.*, Budapest, Hungary, Sep. 2008, pp. 101–104.
- [7] K. P. Dockendorf, I. Park, P. He, J. C. Principe, and T. B. DeMarse, "Liquid state machines and cultured cortical networks: The separation property," *Biosystems*, vol. 95, no. 2, pp. 90–97, Feb. 2009.
- [8] C. Fernando and S. Sojakka, "Pattern recognition in a bucket," in *Proc. 7th Eur. Conf. Artif. Life*, LNAI 2801. Dortmund, Germany, Sep. 2003, pp. 588–597.
- [9] W. J. Goh and N. Crook, "Pattern recognition using chaotic transients," in *Proc. 15th Eur. Symp. Artif. Neural Netw.*, Bruges, Belgium, Apr. 2007, pp. 7–12.
- [10] B. Jones, D. Stekel, J. Rowe, and C. Fernando, "Is there a liquid state machine in the bacterium *escherichia coli*?" in *Proc. IEEE Symp. Artif. Life*, Honolulu, HI, Apr. 2007, pp. 187–191.
- [11] K. Vandoorne, W. Dierckx, B. Schrauwen, D. Verstraeten, R. Baets, P. Bienstman, and J. Van Campenhout, "Toward optical signal processing using photonic reservoir computing," *Opt. Exp.*, vol. 16, no. 15, pp. 11182–11192, Jul. 2008.
- [12] H. Jaeger, "Tutorial on training recurrent neural networks, covering BPTT, RTRL, EKF and the 'echo state network' approach," German Nat. Research Center Information Technology, St. Augustin, Germany, Tech. Rep. GMD 159, 2002.



- [13] D. Verstraeten, B. Schrauwen, M. D'Haene, and D. Stroobandt, "An experimental unification of reservoir computing methods," *Neural Netw.*, vol. 20, no. 3, pp. 391–403, Apr. 2007.
- [14] L. Busing, B. Schrauwen, and R. Legenstein, "Connectivity, dynamics, and memory in reservoir computing with binary and analog neurons," *Neural Comput.*, vol. 22, no. 5, pp. 1272–1311, May 2010.
- [15] B. E. A. Saleh and M. V. Teich, *Fundamentals of Photonics*. New York: Wiley, 1991.
- [16] P. De Heyn, B. Kuyken, D. Vermeulen, W. Bogaerts, and D. Van Thourhout, "High-performance low-loss silicon-on-insulator microring resonators using TM-polarized light," in *Proc. Opt. Fiber Commun. Conf. Exposit. Nat. Fiber Opt. Eng. Conf.*, Los Angeles, CA, Mar. 2011, pp. 1–3.
- [17] W. Bogaerts, D. Taillaert, P. Dumon, D. Van Thourhout, and R. Baets, "A polarization-diversity wavelength duplexer circuit in silicon-on-insulator photonic wires," *Opt. Exp.*, vol. 15, no. 4, pp. 1567–1578, Feb. 2007.
- [18] G. P. Agrawal and N. A. Olsson, "Self-phase modulation and spectral broadening of optical pulses in semiconductor laser amplifiers," *IEEE J. Quantum Electron.*, vol. 25, no. 11, pp. 2297–2306, Nov. 1989.
- [19] W. Bogaerts, P. Dumon, D. Van Thourhout, and R. Baets, "Low-loss, low-cross-talk crossings for silicon-on-insulator nanophotonic waveguides," *Opt. Lett.*, vol. 32, no. 19, pp. 2801–2803, Oct. 2007.
- [20] W. Bogaerts, S. K. Selvaraja, P. Dumon, J. Brouckaert, K. De Vos, D. Van Thourhout, and R. Baets, "Silicon-on-insulator spectral filters fabricated with CMOS technology," *IEEE J. Sel. Topics Quantum Electron.*, vol. 16, no. 1, pp. 33–44, Jan.–Feb. 2010.
- [21] M. Tassaert, S. Keyvaninia, D. Van Thourhout, W. M. J. Green, Y. Vlasov, and G. Roelkens, "A nanophotonic InP/InGaAlAs optical amplifier integrated on a silicon-on-insulator waveguide circuit," in *Proc. Inf. Photon.*, Ottawa, ON, Canada, May 2011, pp. 1–2.
- [22] A. J. Robinson, G. D. Cook, D. P. W. Ellis, E. Fosler-Lussier, S. J. Renals, and D. A. G. Williams, "Connectionist speech recognition of broadcast news," *Speech Commun.*, vol. 37, nos. 1–2, pp. 27–45, May 2002.
- [23] A. J. Robinson, "An application of recurrent nets to phone probability estimation," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 298–305, Mar. 1994.
- [24] G. R. Doddington and T. B. Schalk, "Speech recognition: Turning theory to practice," *IEEE Spectr.*, vol. 18, no. 9, pp. 26–32, Sep. 1981.
- [25] R. Lyon, "A computational model of filtering, detection, and compression in the cochlea," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 7. Paris, France, May 1982, pp. 1282–1285.
- [26] A. N. Tikhonov and V. I. Arsenin, *Solutions of Ill-Posed Problems* (Scripta in Mathematics). New York: Winston, 1977.
- [27] W. Bogaerts and S. K. Selvaraja, "Compact single-mode silicon hybrid rib/strip waveguide with adiabatic bends," *IEEE Photon. J.*, vol. 3, no. 3, pp. 422–432, Jun. 2011.
- [28] N. A. Olsson, "Lightwave systems with optical amplifiers," *J. Lightw. Technol.*, vol. 7, no. 7, pp. 1071–1082, Jul. 1989.



**Kristof Vandoorne** (S'07) was born in Tielt, Belgium, in 1983. He received the M.Sc. degree in electrical engineering from Ghent University, Ghent, Belgium, in 2006.

He has been with the Photonics Research Group, Department of Information Technology, Ghent University-Interuniversity Microelectronics Center, since 2006. From 2008 to 2009, he spent a year in the Arai-Nishiyama Lab, Tokyo Institute of Technology, Tokyo, Japan. He is currently working in the field of photonic reservoir computing.



**Joni Dambre** was born in Ghent, Belgium, in 1973. She received the M.Sc. degree in electronics engineering and the Ph.D. degree in computer engineering from Ghent University, Ghent, in 1996 and 2003, respectively.

She is currently working as a Post-Doctoral Researcher and part-time Professor in the Department of Electronics and Information Systems with the same university. She has performed research on very large scale integration architecture and technology evaluation and novel (including photonic) interconnection techniques. Her current research interests include hardware implementations of reservoir computing and the computational power of general dynamical systems.



**David Verstraeten** (M'06) was born in Brussels, Belgium, on January 17, 1979. He received the M.Sc. degree in computer engineering and the Ph.D. degree from Ghent University, Ghent, Belgium, in 2004 and 2009, respectively.

He is currently a Post-Doctoral Researcher with the Department of Electronics and Information Systems, Ghent University. His current research interests include the study of reservoir computing dynamics and machine learning for biomedical applications and speech recognition.



**Benjamin Schrauwen** (M'03) received the M.Sc. degree in computer science from the University of Antwerp, Edegem, Belgium, in 2000, and the M.Sc. and Ph.D. degrees in computer engineering from Ghent University, Ghent, Belgium, in 2002 and 2008, respectively.

He has been a Professor with Ghent University, affiliated with the Department of Electronics and Information Systems in the Faculty of Engineering, since 2010. He currently leads a research group of about 15 people with interests in reservoir computing, spiking neural networks, and deep learning architectures, with applications in the domains of speech recognition, robotics, and biomedical signal processing. His current research interests include machine learning and biologically inspired computing systems.

Prof. Schrauwen is a member of the American Association for the Advancement of Science.



**Peter Bienstman** (M'96) was born in Ghent, Belgium, in 1974. He received the M.Sc. degree in electrical engineering and the Ph.D. degree from Ghent University, Ghent, in 1997 and 2001, respectively.

He is currently an Associate Professor with the Department of Information Technology, Ghent University. From 2001 to 2002, he was with the Joannopoulos Research Group, Massachusetts Institute of Technology, Boston. He has published over 50 papers and holds several patents. His current

research interests include several applications of nanophotonics (biosensors and photonic information processing), as well as nanophotonics modeling.

Dr. Bienstman is a member of the IEEE PHOTONICS SOCIETY. He has been awarded the European Research Council starting grant for the Naresco-project: Novel paradigms for massively parallel nanophotonic information processing.