Photonic Reservoir Computing: a New Approach to Optical Information Processing

Kristof Vandoorne
PRG, Intec
Ghent University – IMEC
Sint-Pietersnieuwstraat 41
9000 Gent, Belgium
Kristof.Vandoorne@intec.UGent.be

Martin Fiers
PRG, Intec
Ghent University – IMEC
Sint-Pietersnieuwstraat 41
9000 Gent, Belgium
Martin.Fiers@intec.UGent.be

David Verstraeten
PARIS, Elis
Ghent University
Sint-Pietersnieuwstraat 41
9000 Gent, Belgium
David.Verstraeten@elis.UGent.be

Benjamin Schrauwen
PARIS, Elis
Ghent University
Sint-Pietersnieuwstraat 41
9000 Gent, Belgium
Benjamin.Schrauwen@elis.UGent.be

Joni Dambre
PARIS, Elis
Ghent University
Sint-Pietersnieuwstraat 41
9000 Gent, Belgium
Joni.Dambre@UGent.be

Peter Bienstman
PRG, Intec
Ghent University – IMEC
Sint-Pietersnieuwstraat 41
9000 Gent, Belgium
Peter.Bienstman@UGent.be

Abstract

Despite ever increasing computational power, recognition and classification problems remain challenging to solve. Recently, advances have been made by the introduction of the concept of reservoir computing. This is a methodology coming from the field of machine learning and neural networks that has been successfully used in several pattern classification problems, like speech and image recognition. Thus far, most implementations have been in software, limiting their speed and power efficiency. Photonics could be an excellent platform for a hardware implementation of this concept because of its inherent parallelism and unique nonlinear behaviour. Moreover, a photonic implementation offers the promise of massively parallel information processing with low power and high speed. We propose using a network of coupled Semiconductor Optical Amplifiers (SOA) and show in simulation that it could be used as a reservoir by comparing it to conventional software implementations using a benchmark speech recognition task. In spite of the differences with classical reservoir models, the performance of our photonic reservoir is comparable to that of conventional implementations with the same number of nodes. As our implementation uses coherent light for information processing, we find that phase tuning is crucial to obtain high performance.

1 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 neural networks [1,2]. In this framework a randomly initialized RNN, called the 'reservoir', is used and left untrained. The states of all the nodes of the RNN are then fed into a linear readout, which can then be trained with simple and well established methods. Usually, a mere linear regression is used. Hence, the difficulties of training a recurrent network are avoided as only the readout is changed. In this way, the RC approach combines the memory and spatio-temporal processing of RNNs with the ease of training of feedforward neural networks. Reservoir computing has been demonstrated to equal or outperform other state-of-the-art techniques for several complex machine learning tasks. An example is the prediction of the Mackey-Glass chaotic time series several of orders of magnitude better than classic methods [1].

Although the reservoir itself remains untrained, its performance depends drastically 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 behaviour, because this regime optimizes the system's memory. 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.

A common measure for the dynamic regime is the spectral radius, the largest eigenvalue of the system's Jacobian, calculated at its maximal gain state (for classical hyperbolic tangent reservoirs, this corresponds to the largest eigenvalue of the network's interconnection weight matrix). The spectral radius is an indication of the stability of the network. If its value is larger than one, the network might become unstable. At each point in time, the actual gain of a nonlinear reservoir depends on its state and its input signal(s). Instability occurs if this gain on average exceeds one. As 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.

2 Photonic reservoir computing

Most reported results on reservoir computing use a (randomized) network of hyperbolic tangent or spiking neurons. However, recent work has indicated that a wide range of sufficiently high-dimensional nonlinear dynamic systems can be used as a reservoir. One way of explaining the success of reservoir computing 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 mixes the inputs together, so that the interesting features are more easily extracted by the readout. A nonlinear mixing often seems to offer advantages over linear mixing when dealing with more complicated problems such as speech recognition.

Most implementations thus far have been software based, hence the pursuit of finding a suitable hardware platform for performing the reservoir calculation. This transition offers the potential for huge power consumption savings and speed enhancement. What makes a hardware implementation even more attractive is the fact that the computation in RC happens through the transient states and changing dynamical behaviour. This is in sharp contrast with digital computation where the state is only important after the transients have died out.

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. There remain, however, many challenges. If you encode the information in changing power levels, then it becomes difficult to have negative weights and to subtract signals. A topology made on a 2D chip, which is the case for most photonic chips nowadays, limits the freedom in connectivity that exists in software implementations, since one would like to minimise the number of crossings. When using a coherent light source, the amplitude and accompanying phase start to play a role as well, whereas traditional RC is only amplitude based. In this paper we will show through simulation results that despite these limitations, photonic reservoirs can perform quite well on benchmark problems.

3 Simulation results

In this section the speech recognition task that we used as a benchmark problem will be described, as well as the model we used to simulate a photonic reservoir. In our case the reservoir consists of a network of coupled Semiconductor Optical Amplifiers (SOAs).

3.1 Speech Recognition

Speech recognition is a very difficult problem to solve and methods based on ANNs have been among the state of the art for a long time. Reservoir Computing with classical neural networks has been employed with success for speech recognition. The task used in this paper is the discrimination between spoken digits, the words 'zero' to 'nine', uttered by 5 female speakers. The dataset and the simulation framework for classical reservoirs is publicly available (http://snn.elis.ugent.be/rctoolbox). As is standard for speech recognition, some pre-processing of the raw speech signal is performed before it is fed into the reservoir. Often these methods involve a transformation to the frequency domain and highlighting certain frequencies typical for our ear by using some kind of ear model. The model used for the results in this paper was the Lyon ear model [3].

Reservoir memory is related to the typical time scales of the reservoir itself. Therefore, to achieve optimal memory in a reservoir, the relevant time scales of the input signals must be adapted to those of the physical reservoir implementation. Audio signals are rather slow, so we accelerated the speech signal to accord with timescales typical for the delays in a network of SOAs (duration of one digit in the order of a few hundred ps). Hence, although we use this task to demonstrate the potential of photonic reservoir computing, we don’t propose to use photonic reservoirs as a platform for standard real-time, slow audio signals.

3.2 Simulation model

SOAs, with a saturation of gain and output power, are the optical device closest to the hyperbolic tangent functions used in many ANN implementations. That is the reason we chose them as a first medium to verify the usefulness of photonic reservoirs. The SOA model we used is one proposed by Agrawal [5]. It captures the most important features such as gain saturation, carrier lifetime and phase shift depending on the gain. Spectral hole burning, cross gain and phase modulation were not considered, since the operation is set to be at one wavelength.

A time step based rate equation model, solved by a fourth order Runge-Kutta, of a network of coupled SOAs was then plugged into the freely available RC toolbox mentioned earlier. This toolbox offers a number of reservoir simulations and benchmark problems that can be simulated and tested. Furthermore almost every aspect of it can be changed to make it suitable to specific experiments.

3.3 Results

In our experiments the input consists of 77 channels, resulting from the pre-processing of the speech data. With such high-dimensional input, the number of nodes needs to be sufficiently large. All the experiments were done with a network of 81 nodes. An example of the topology used can be seen in Fig. 1, showing a network of 3 by 3 SOAs. The connections are made in such a way that they don’t cross. The information flows from the top left to the bottom right SOAs with nearest neighbour connections. The feedback is assured by having as many feedback connections on the edges of the network as possible without having to use crossings. The topology we used was a 9 by 9 network but the construction method was the same.

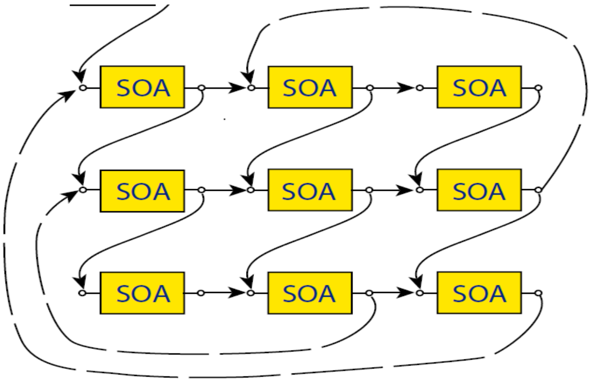

Figure 1: the topology used for the SOA simulations

In the experiment we varied two variables: the phase change and attenuation in every connection. Although in practice these wouldn't be the variables that are swept, they are orthogonal and therefore provide an interesting insight in the behaviour of the network. In reality the input current of the SOAs and the wavelength of the light can be used, but since the input current of the SOAs influences the gain, which in turn affects the phase change inside an SOA, these variables are not orthogonal.

The total amount of gain/loss in the network is also calculated by means of the spectral radius, mentioned earlier. Since coherent light is used, the signals can be represented by complex amplitudes. Since the propagation through an interconnection also corresponds to a phase change of the signal, connection weights, too, need to be modeled as complex values. Hence, in this work, we use the maximal eigenvalue amplitude of the complex interconnection matrix as an extension of the traditional spectral radius measure. In this way interference effect is taken into account.

An example result for one of our experiments is shown in Fig. 2. Here a clear transition at a spectral radius around 1 can be seen. Above 1 the results suddenly become a lot worse due to the fact that instability kicks in because there is gain in the network. This regime actually corresponds with one where some SOAs would become lasers. Our simulation model is probably not suited for addressing a network of coupled lasers, so it is not possible to state with certainty that this regime is very bad for RC. Maybe some kind of emergent behaviour, useful for RC, could arise from coupled lasers, but this is the topic of ongoing investigation.

Another consequence of using coherent light is the fact that the delay in the connections changes the phase of the light according to the wavelength, length and the effective index of the connection. For computational reasons all the connections were considered equally long in this experiment, which means that the phase change in every connection can be changed at the same rate, for example by changing the wavelength. Some variations on the length will inevitably exist in real hardware. Preliminary results indicate that this might cause a slight deterioration of the optimal result, but this is still under investigation. Phase is important, since it determines the interference of the light when it combines in front of every SOA. From Fig. 2, it is clear that the network performance depends on the phase change (the interference). In fact, below the instability boundary, our system is much more sensitive to phase than it is to gain. To make a photonic reservoir robust to process variations and noise, affecting the phase behaviour of the system, it is therefore important to design architectures with an optimal region that is as wide as possible along the phase axis. The design of networks with large optimal areas (making them phase 'independent') is the topic of ongoing research. An easy way of doing this is working with incoherent light, but the results are typical a lot worse for incoherent networks.

The performance for speech recognition tasks is typically expressed as a Word Error Rate (WER). For the experiment in Fig. 2, we can identify a relatively wide optimal region in which, with our network of only 81 SOAs, we obtained a WER of about 1%. This is comparable to the results achieved by classical hyperbolic tangent networks with the same number of nodes, but other parameters such as the spectral radius and connection fractions optimized. The performance, however, of both software and photonic reservoirs could be improved when a low-pass filter is added to all of the nodes. In the case of optical components this could be achieved by adding a narrowband filter to every SOA. This integration of SOAs and filters, however, complicates the fabrication because of the strict tolerances on the wavelength selectivity and uniformity of the filters.

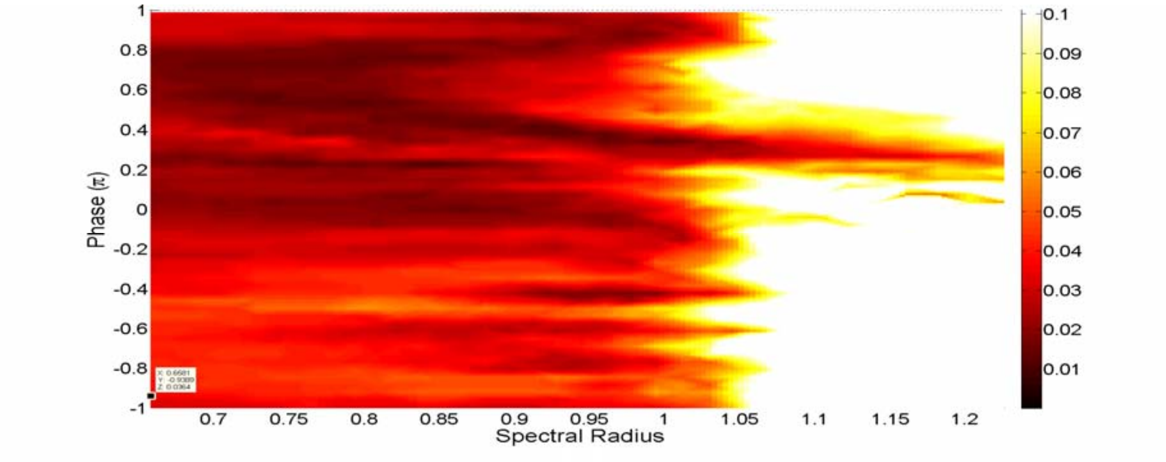

Figure 2: Simulation result for a network of coupled SOAs for speech recognition. The x-axis shows the spectral radius, the y-axis the phase change in every connection. The darker the colour, the better the performance.

4 Conclusions

In this paper we have investigated a network of coupled SOAs as a reservoir for RC by means of evaluating this kind of reservoir on a benchmark speech recognition task. It turns out that SOA reservoirs can be used to solve such kind of complex problems, despite the limitations imposed by a practical implementation. In the future, a practical demonstration of a chip of SOAs used as a photonic reservoir will be further pursued.

Acknowledgments

K. Vandoorne acknowledges the Special Research Fund (BOF) of Ghent University for a specialization grant. This work has been carried out in the framework of the IAP project Photonics@be of the Belgian Science Policy.

References

[1] H. Jaeger, H. Haas: Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication, *Science,* vol. 304, pp. 78-80, Apr 2 2004.

[2] W. Maass, T. Natschlager, H. Markram: Real-time computing without stable states: A new framework for neural computation based on perturbations, *Neural Computation*, vol. 14, pp. 2531-2560, Nov 2002.

[3] R. Lyon: A computational model of filtering, detection and compression in the cochlea, *in Proc. IEEE ICASSP,* Paris, May 1982, pp. 1282-1285.

[4] D. Verstraeten, PhD thesis: Reservoir Computing: Computation with Dynamical Systems, in PARIS, Dept. of Electronics and Information Systems Ghent: Ghent University, 2009.

[5] G. P. Agrawal, N. A. Olsson,: Self-Phase Modulation and Spectral Broadening of Optical Pulses in Semiconductor-Laser Amplifiers, *IEEE Journal of Quantum Electronics,* vol. 25, pp. 2297-2306, Nov 1989.