Photonic Reservoir Computing with Coupled Semiconductor Optical Amplifiers

Kristof Vandoorne[1], Wouter Dierckx[1], Benjamin Schrauwen[2], David Verstraeten[2], Peter Bienstman[1], Roel Baets[1], and Jan Van Campenhout[2]

[1] Photonics Research Group, Dept. of Information Technology, Ghent University – IMEC, Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium
kristof.vandoorne@UGent.be
[2] PARIS, Dept. of Electronics and Information Systems, Ghent University, Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium

Abstract. We propose photonic reservoir computing as a new approach to optical signal processing and it can be used to handle for example large scale pattern recognition. Reservoir computing is a new learning method from the field of machine learning. This has already led to impressive results in software but integrated photonics with its large bandwidth and fast nonlinear effects would be a high-performance hardware platform. Therefore we developed a simulation model which employs a network of coupled Semiconductor Optical Amplifiers (SOA) as a reservoir. We show that this kind of photonic reservoir performs even better than classical reservoirs on a benchmark classification task.

1 Introduction

Reservoir Computing was recently proposed [1,2] as a general framework to handle classification and recognition problems. The reservoir itself consists of a network of coupled nonlinear elements and their interactions facilitate the processing and classification of the incoming signals by the readout function. These reservoirs were until now mainly software-based and they have been employed successfully in a large variety of applications like speech recognition [3,4,5], event detection [6], robot control [7], chaotic time series generation and prediction [1,8].

Photonic reservoir computing could be used for a large variety of problems, going from large scale pattern recognition in (real-time) video data to signal processing (header recognition, error correction, etc.) in optical fiber networks. We studied a photonic reservoir, made of coupled SOAs, and this paper presents the first results. The structure of the paper is as following. In section 2 we will go deeper into this new concept of reservoir computing. The next section deals with the (photonic) implementation aspects. Section 4 describes the classification task we used, to show the potential of photonic reservoir computing. This

S. Dolev, T. Haist, and M. Oltean (Eds.): OSC 2008, LNCS 5172, pp. 46–55, 2008.

task requires the reservoir to distinguish between a triangular and a rectangular waveform. It turns out that a photonic reservoir — with only a little bit of tuning and a limited number of 25 SOAs — can already distinguish between the two signals over 97 % of the time.

2 Reservoir Computing

2.1 Classical Approach

In this digital age, signals are often transferred to the digital domain for signal processing. Nature shows us however that there are alternatives, which can be superior for complex classification and recognition problems, like the human brain combined with eyesight. Machine learning looks to the biological world for inspiration, where organisms often learn from their failures and successes or in other words from examples. Systems in machine learning are accordingly trained to perform certain tasks. Artificial Neural Networks (ANN) are an example of such a system and take the analogy with the biological world one step further [9]. The inspiration for the system comes from its biological counterpart, the human brain, which consists of neurons. The human brain lacks speed compared to a computer, but it compensates this by having a rich interconnection topology. Each connection has a certain weight attached to it and these weights can be adapted during the training process.

Feed-forward neural networks have been extensively used for non-temporal problems and they are well understood due to their non-dynamic nature. At the same time, this limits their applicability in dealing with time varying signals. Indeed, neural networks with feedback loops (so-called recurrent neural networks) provide some kind of internal memory which allows them to extract time correlations. However, this turned out to be a hurdle in finding a general learning rule, which is a method used to train the neural network to perform a desired task. This is why different rules exist for different tasks and topologies, thus limiting their broad applicability.

Around 2002 two solutions (Liquid State Machines and Echo State Networks) were independently proposed [1,2]. They have in common that the network is split up in two parts. One part — the reservoir — is a random recurrent neural network that is left untrained and kept fixed while using. The input is fed into this reservoir. The second part is a readout function, which takes as input the reservoir state — a collection of the states of the individual elements. In order to be able to achieve useful functionality, this part of the system needs to be trained, typically on a set of inputs with known classifications. This process is visualized in figure 1. Any kind of classifier or regression function could be used, but it turns out that for most applications a simple linear discriminant suffices. In this way the interesting properties of recurrent neural networks are kept in the reservoir part, while the training is now restricted to the memoryless readout function.

One might wonder why such an approach would be useful to solve complex classification tasks. However, it is well known in the machine learning community that projecting a low-dimensional input into a high-dimensional space can actually be beneficial for the performance of a classification algorithm. As classes, which are only separable by a nonlinear function in the low-dimensional space, can become separable by an easier, linear function in the high-dimensional space [10]. This concept is applied e.g. in support vector machines.

The reservoir could be seen as an integration of the temporal correlations in the signal into a spatial correlation in the reservoir state. This is not to say that any recurrent neural network will do. Rather, it appears that the dynamics of the network should be in the dynamic region which corresponds to the edge of stability [11]. The dynamics depend on the amount of gain and losses in the network and they should be balanced. If the network is over-damped there is no memory inside the reservoir, if it is under-damped the network will react chaotically.

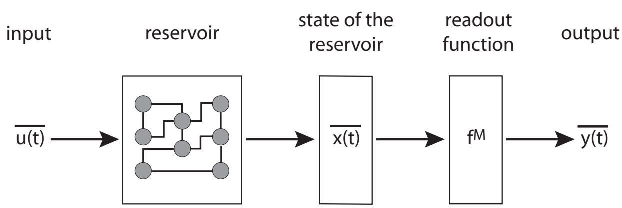

Fig. 1. Reservoir Computing

Recently a toolbox, able to simulate and test reservoirs, was created [12]. In this toolbox the reservoirs are neural networks. One of those is the classical variant where the signals are analog and every node is a hyperbolic tangent function operating on a weighed sum of its inputs. This function is S-shaped as in figure 2 (left). In this kind of network the nodes themselves are very simple, while the dynamics come from the complex interconnection topology.

2.2 Photonic Approach

The theory behind reservoir computing is not restricted to neural networks. One requirement for the reservoir is fading memory, which means that the influence of an input should fade away slowly. The present software implementations are rather slow and therefore we investigated the potential of a hardware implementation based on light. This could be faster and more power efficient due to the large bandwidth and fast nonlinear effects inherent to light.

Due to the nature of reservoir computing, its implementation can be split up in two distinctive parts: the reservoir on the one hand and the readout function on the other. Since the computational power of reservoir computing seems to reside mainly in the reservoir due to its feedback and nonlinearities, the focus of the research was on a photonic reservoir. As mentioned before the readout is a simple linear discriminant, but its training depends on mathematical calculations like matrix inversion. This could initially be done off-line with a computer or by an electronic chip.

Because the specifications for the reservoir do not seem to be very rigid, the choice of possible nanophotonic components was vast. We opted for coupled Semiconductor Optical Amplifiers (SOA), based on two observations. First of all, the steady-state curve of an SOA resembles the S-curve used in analog neural networks (figure 2) — at least for the upper branch, but since optical power is non-negative this is a restriction we have to cope with. This resemblance made SOAs more likely to be able to bridge the reservoir and the photonic world. The dynamic behavior of an SOA is however more complex in comparison to the classical analog implementations. The carrier dynamics come into play at high data rates and because of this a reservoir of SOAs is a middle ground between simple nodes with a complex network (the classical tanh reservoirs) and one very complex node. Second, SOAs are broadband which makes the communication between different nodes less critical as would be the case with resonating structures.

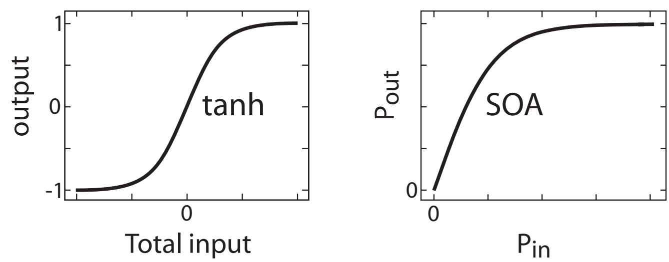

Fig. 2. (Left) tanh for analog ANN — (right) SOA: steady state

3 Simulation Model

We developed our simulation program for photonic reservoirs within the framework of the toolbox, mentioned previously. This allows us to use the existing training schemes for the memoryless readout function. For further details about this open source toolbox we refer to the manual online [12] and the article by D. Verstraeten et al. [13].

3.1 SOA Model

In our simulations we work with a traveling wave SOA. This kind of SOA has anti-reflection coatings on its facets, which allows us to neglect the influence of reflections. We use the standard traveling wave SOA equations [14]. We neglect the influence of Amplified Spontaneous Emission (ASE) and spectral hole burning in this model. This means that we assume that the input signal itself will be strong enough to dominate the ASE and that we only use light at one wavelength. To incorporate the longitudinal dependence of the gain, the equations can be solved for a concatenation of small sections of the SOA. Since the latter can be time consuming when working with large networks of SOAs, we work mainly with one section. Moreover since reflections are neglected at this stage, we use unidirectional signal injection. This reduces the number of rate equations to be solved to one.

3.2 Topology and Reservoir Simulation

The classical reservoir implementations with neural networks have random interconnection topologies. Since the standard optical chip is still 2D we investigated structures that can be realized without intersections. Two of those structures are depicted in figure 3. The left structure is a waterfall system which acts as a nonlinear delay-line. Although this feed-forward topology is relatively simple, it has already been successfully used to model nonlinear systems [15]. The other network has feedback connections on the sides in order to avoid crossings. Since the SOAs are modeled as unidirectional, the connections are too.

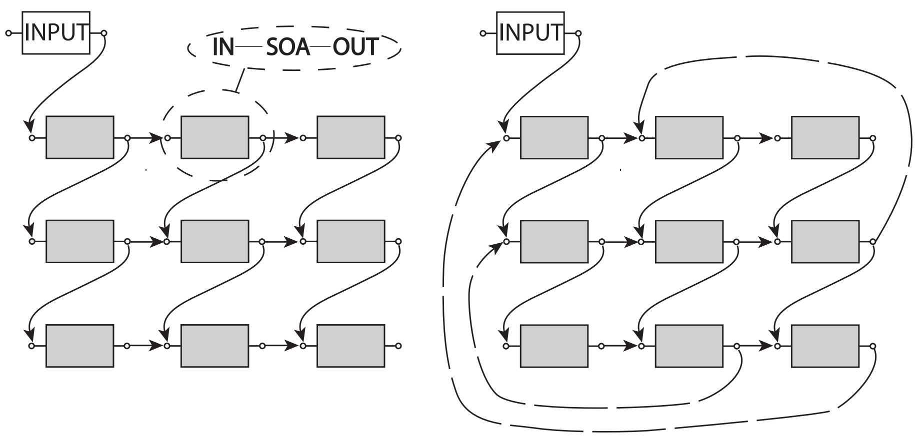

Fig. 3. Two topologies: (left) a feed-forward network – (right) a waterfall network with feedback connections (long dash) at the edges

At every time instant two computational steps are taken. During the first step the internal state of every node is updated, while during the second step the outputs are transferred to the inputs they are connected to. The splitters and combiners are modeled as adiabatic and every connection can have a different delay and attenuation.

The readout function takes as input the power of every SOA in the network at every time instant. This is the basic structure of the simulation model. Next, we will look at tasks that can be solved with these kind of networks.

4 Pattern Recognition

4.1 Task Description

We will use a simple classification task to demonstrate the potential of photonic reservoir computing. The task is depicted in figure 4. The system has to be able, by means of training by examples, to instantly differentiate between a rectangular and a triangular waveform. Moreover, if the input signal changes the system has to change its output as fast as possible. In the top part (a) of figure 4, an example of such an input is depicted. Figure 4b shows the output that the system should generate accordingly. If the input is triangular than the system should constantly return 1, if the input is rectangular it should return -1. Figure 4c shows the state of a few SOAs as they are excited by the input, while figure 4d shows the result of the readout function. The readout uses a linear combination of the states of the reservoir nodes, to approximate the desired output (black curve) as closely as possible (blue curve). In the last stage a sign function is used on this approximation to define the final output of the system. As a result the output is either 1 or -1 as can be seen in figure 4e. In the example the system manages most of the time to follow the desired output.

Since the output function is memoryless, it should be able to handle transitions of the waveform at different instants. Hence several samples are made with different transitions. One part of these samples is used to train the readout function, while the other part is used to test it. These test results are used to define the quality of the reservoir.

4.2 Results

In figure 5 some results are displayed from our simulations. The vertical lines are error bars, which show the standard deviation on the reservoir performance over ten runs. The variation comes, for the photonic reservoirs, from different samples with different transitions at different instants. The tanh networks have an extra variation source because they are randomly created.

In the left figure the two photonic topologies from figure 3, with 25 SOAs, are compared against the attenuation in the connections. This attenuation influences the dynamic regime and the higher the attenuation, the more damped the system is. It shows that feedback is beneficial for the performance of photonic reservoirs, when used in the appropriate dynamic regime. The best result is an error rate of 2.5 % for the network with feedback loops. This means that the reservoir is only in 2.5 % of the time incorrect in its distinction between the rectangular and the triangular waveform. If the attenuation in the connections gets too small then the performance decreases dramatically for the feedback network.

On the right of figure 5 a feedback reservoir with SOAs and the classical tanh reservoirs are compared, each with 25 nodes. Since there is feedback in both networks we use the spectral radius ($\rho(\cdot)$) as a measure for the dynamics in the system. The spectral radius is the absolute value of the largest eigenvalue of the connection matrix C, containing all the gain and loss in the network and is an

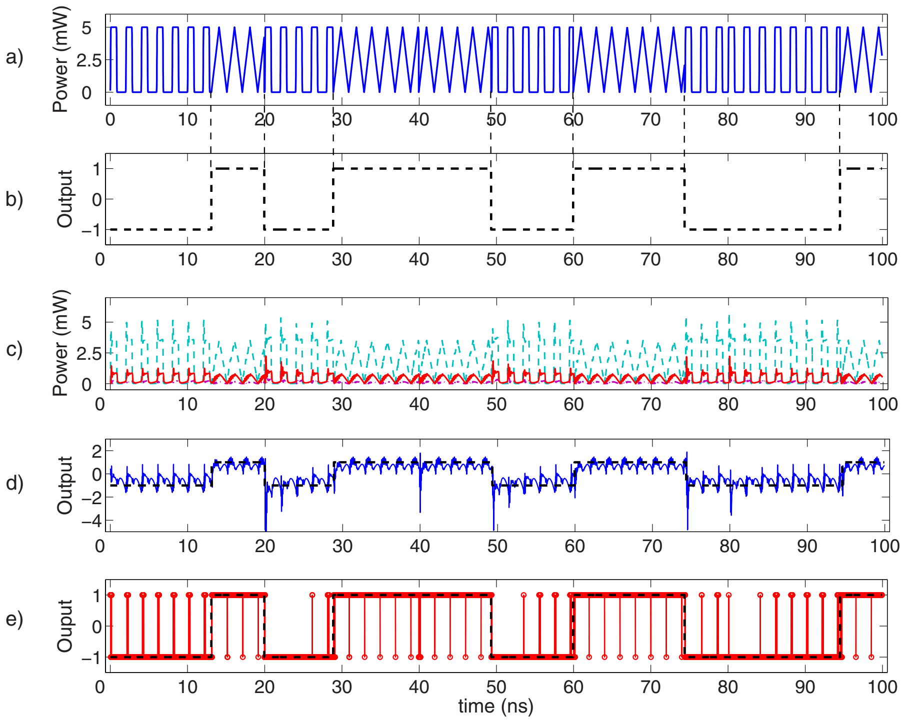

Fig. 4. Pattern recognition task: a) Input signal with different transitions between the rectangular and triangular waveform b) desired output c) state (i.e. optical power level) of some of the reservoir nodes d) The approximation (blue) of the desired output (black) by the readout function, e) final output of the system (red)

often used parameter in the field of reservoir computing. In a linear network a spectral radius smaller than one means the network is stable, a value larger than one means chaotic. The interesting dynamical region, the edge of stability, holds for spectral radii just below one. Although our network is nonlinear, we can still use this as an approximation, where the spectral radius acts as an upper estimate. The gain for every node is linearized around zero input power and for a connection matrix C with eigenvalues $\lambda_i, \ldots, \lambda_n$ this leads to the following spectral radius calculation:

$$\rho(C_{lin(0)}) = \max_{1 \leq i \leq n} |\lambda_i| \tag{1}$$

The classical reservoir appears to behave better for small spectral radii (when the system is damped) with an optimum value around 3.5 %. As soon as the spectral radius gets too high, the system becomes chaotic which explains the large error bars and bad results. The same holds for the photonic reservoirs at higher spectral radii. The curve for the SOA network with feedback is the same in the two figures but plotted against a different parameter. An optimal value of

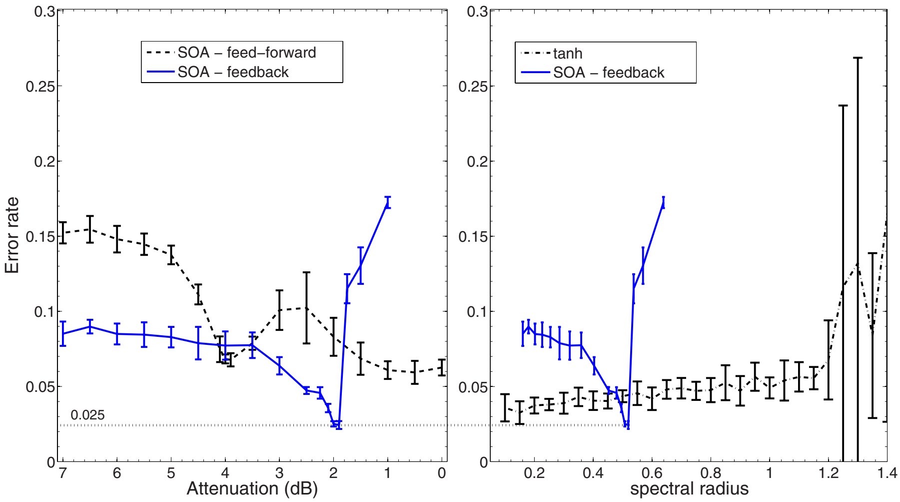

Fig. 5. Results with frequency of 0.5 GHz, simulation time: 100 ns, amplitude power: 5 mW, delay: 6.25 ps, fixed input–output weights, SOA input current: 80 mA, 25 nodes: (left) – photonic reservoirs with and without feedback, (right) – classical reservoirs versus photonic reservoirs with feedback

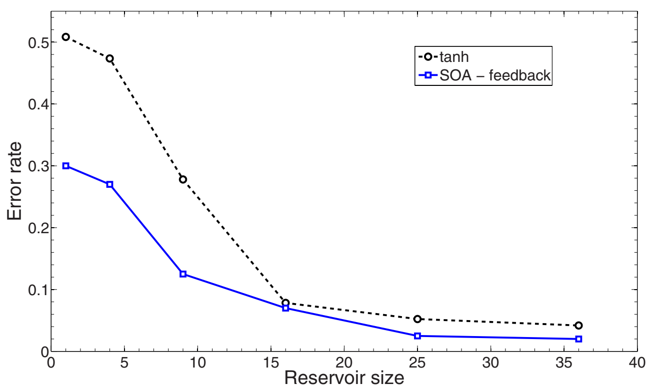

Fig. 6. This figure shows the influence of the reservoir size on the performance of the classical and photonic reservoirs (with feedback)

2.5 % is obtained for a spectral radius around 0.5. This means that the system detects the correct waveform almost 97.5 % of the time.

It is remarkable that the photonic reservoir with feedback is slightly better than the classical reservoir for this task, considering the simple photonic topology. One explanation is the different response of an SOA to different rise times. The rectangular waveform, although smoothed with a tanh, rises faster than the triangular waveform, causing a depletion of the carriers in the SOA. This can be seen in figure 4c, when peaks appear whenever a rising edge of the rectangular waveform passes through an SOA. This result indicates that the planar-topology limitation for photonic reservoirs is at least compensated by the richer dynamics of the SOAs.

In figure 6 we see that the reservoir performance enhances with larger reservoirs, although it seems to saturate. The choice for a certain reservoir size is then a trade-off between the specifications of the task and cost of the chip.

5 Conclusions

We have demonstrated in this paper the potential of photonic reservoir computing, since our photonic reservoirs manage to discriminate over 97 % of the time between the two waveforms in our classification task. Even though, they work with a limited number of SOAs and limited amount of feedback. This is a promising step toward the use of photonic reservoirs for large scale image recognition and signal processing. In future work we want to obtain an experimental verification of the described simulation results.

Acknowledgments

KV acknowledges the Special Research Fund of Ghent University (BOF – UGent) for a doctoral mandate.

References

1. Jaeger, H., Haas, H.: Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. Science 304, 78–80 (2004)
2. Maass, W., Natschläger, T., Markram, H.: Real-time computing without stable states: A new framework for neural computation based on perturbations. Neural Computing 14, 2531–2560 (2002)
3. Maass, W., Natschläger, T., Markram, H.: A model for real-time computation in generic neural microcircuits. In: Proceedings of NIPS, pp. 229–236. MIT Press, Vancouver (2003)
4. Skowronski, M.D., Harris, J.G.: Minimum mean squared error time series classification using an echo state network prediction model. In: Proceedings of IEEE International symposium on circuits and systems. Institute of Electrical and Electronics Engineers, Island of Kos (2006)
5. Verstraeten, D., Schrauwen, B., Stroobandt, D., Van Campenhout, J.: Isolated word recognition with the Liquid State Machine: a case study. Information Processing Lett. 95, 521–528 (2005)
6. Jaeger, H.: Reservoir riddles: Suggestions for echo state network research (extended abstract). In: Proceedings of IEEE International Joint Conference on Neural Networks, pp. 1460–1462. Institute of Electrical and Electronics Engineers, Montreal (2005)
7. Joshi, P., Maass, W.: Movement generation and control with generic neural micrrocircuits. In: Proceedings of BIO-ADIT, pp. 16–31. Springer, Berlin (2004)

8. Steil, J.J.: Online stability of backpropagation-decorrelation recurrent learning. Neurocomputing 69, 642–650 (2006)
9. Bishop, C.M.: Neural Networks for Pattern Recognition. Clarendon Press, Oxford (1995)
10. Vapnik, V.N.: An overview of statistical learning theory. IEEE Trans. Neural Networks 10, 988–999 (1999)
11. Legenstein, R., Maass, W.: What makes a dynamical system computationally powerful? In: New directions in statistical signal processing: from systems to brain, pp. 127–154. MIT Press, Cambridge (2007)
12. Reservoir lab: Reservoir Computing Toolbox, http://snn.elis.ugent.be/node/59
13. Verstraeten, D., Schrauwen, B., D'Haene, M., Stroobandt, D.: An experimental unification of reservoir computing methods. Neural Networks 20, 391–403 (2007)
14. Agrawal, G.P., Olsson, N.A.: Self-Phase Modulation and Spectral Broadening of Optical Pulses in Semiconductor-Laser Amplifiers. IEEE J. Quantum Electron. 25, 2297–2306 (1989)
15. Cernansky, M., Makula, M.: Feed-forward echo state networks. In: Proceedings of IEEE International Joint Conference on Neural Networks, pp. 1479–1482. Institute of Electrical and Electronics Engineers, Montreal (2005)