



**HAL**  
open science

# Large scale autonomous all analogue photonic neural networks

Ivan Boikov

► **To cite this version:**

Ivan Boikov. Large scale autonomous all analogue photonic neural networks. Optics [physics.optics]. Université Paris-Saclay, 2023. English. NNT : 2023UPASP205 . tel-04850266

**HAL Id: tel-04850266**

**<https://theses.hal.science/tel-04850266v1>**

Submitted on 20 Dec 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Large scale autonomous all analogue photonic neural networks

*Réseaux neuronaux évolutifs et autonomes en  
photonique*

## Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 572 Ondes et Matière (EDOM)

Spécialité de doctorat : Physique

Graduate School : Physique. Référent : ENS Paris-Saclay

Thèse préparée dans la unité de recherche **LuMin (Université Paris-Saclay, ENS Paris-Saclay, CNRS)**, sous la direction de **Alfredo de Rossi**, ingénieur de recherche, HDR, Thales Research and Technology (Palaiseau), la co-direction de **Daniel Brunner**, chargé de recherche CNRS, HDR, l'Institut FEMTO-ST

Thèse soutenue à Palaiseau, le 19 décembre 2023, par

**Ivan BOIKOV**

### Composition du jury

Membres du jury avec voix délibérative

<b>Xavier Checoury</b> Professeur, Université Paris-Sud	Président
<b>Peter Bienstman</b> Professeur, Université de Gand	Rapporteur
<b>Miguel C. Soriano</b> Maître de conférences, HDR équiv., IFISC	Rapporteur
<b>Christelle Monat</b> Maîtresse de conférences, HDR, EC Lyon	Examinatrice

**Titre :** Réseaux neuronaux évolutifs et autonomes en photonique

**Mots clés :** calcul neuromorphique, photonique intégrée, optique non linéaire, systèmes dynamiques, laser

**Résumé :** Un réseau neuronal est une approche de l'informatique inspirée du cerveau permettant de résoudre des problèmes difficilement abordables avec les méthodes algorithmiques traditionnelles. En conséquence, les réseaux neuronaux ont trouvé leur place dans nombreuses applications. Certaines de ces tâches nécessitent non seulement d'une grande puissance de calcul, mais aussi exigent une latence faible, ce qui a inspiré la recherche sur les réseaux neuronaux dans le domaine optique.

Ce travail porte sur un *reservoir computer* constitué de micro résonateurs optiques couplés de manière évanescente. On montre que cette architecture est prometteuse dans certaines applications, par exemple le traitement en temps réel de signaux optiques cohérents à grande vitesse. Dans cette architecture, contrairement à la plupart des études de la lit-

térature, les lignes à retard optiques sont supprimées, ce qui réduit considérablement l'encombrement de la puce. Le *reservoir* est étudié d'une manière générale, qui s'applique à une large gamme de semi-conducteurs et de résonateurs. Un cadre mathématique est établi pour sa simulation et l'évaluation de ses propriétés intrinsèques. Sa capacité de calcul est démontrée par la prédiction d'un système physique et la récupération de données à partir d'une fibre optique à une vitesse de plusieurs dizaines de gigahertz. Une piste pour le développement de réseaux photoniques intégrés à grande échelle est ainsi proposée.

Ce travail étudie également la possibilité d'utiliser les nanolasers excitables, récemment démontrés, en « neurones » ultrarapides dans des réseaux probabilistes, capables de traiter des millions d'échantillons par seconde.

**Title:** Large scale autonomous all-analogue photonic neural networks

**Keywords:** neuromorphic computing, integrated photonics, nonlinear optics, dynamical systems, laser

**Abstract:** A neural network is a brain-inspired approach to computing that allows us to solve problems that are difficult for traditional algorithmic methods. As a result, neural networks have found their place in many applications. Proposals for new tasks call for neural networks with high computational power, high speed and low latency, which has inspired research into neural networks in the optical domain.

This work considers a reservoir computer made by evanescently coupled optical microresonators. This architecture allows complex online processing of high-speed coherent optical signals and, in contrast to most studies in the literature, optical delay lines are suppressed, significantly reducing the chip foot-

print. The reservoir is assessed in a general sense by establishing a mathematical framework for its simulation and the evaluation of intrinsic properties, which applies to a wide range of semiconductors and types of resonators. Its computing capability is demonstrated by predicting a physical system and data recovery from an optical fiber at a speed of tens of gigahertz. In this way, a direction for the development of large-scale integrated photonic networks is proposed.

This work also investigates the possibility of harnessing the recently demonstrated spiking nanolasers for the development of high-speed stochastic neural networks capable of sampling at hundreds of millions of samples per second.

This project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the Maria Skłodowska-Curie project<sup>1</sup> POST-DIGITAL (grant agreement 860360). We acknowledge *imec* for chip fabrication and the Horizon 2020 Fun-Comp project<sup>2</sup> (grant agreement 780848) for its funding.

---

<sup>1</sup><https://postdigital.astonphotonics.uk>

<sup>2</sup><https://fun-comp.org>

## Acknowledgments

The last three years have certainly been a tough ride with high highs and low lows and many people on the way that have contributed to its conclusion.

First and foremost, I would like to express my gratitude to my supervisors Alfredo and Daniel, without whose active participation nothing would have come out. Many things in this field are... nontrivial, and their insight, patience and support have been central to getting this work to where it is today. The final chapter of this work was only there because of Mihai A. Petrovici at the University of Bern, who gave a lot of valuable pieces of advice.

Even though I did not spend much time in the lab, my thanks to those who showed me the ropes and especially Sylvain Combri , who helped with almost everything I have done there.

I am grateful to those who helped with the chip: J r me Bourderionnet and a fellow POST-DIGITAL student Tigers Jonuzi at VLC Photonics, for taking time during a hectic period to help and figure out minor (and not very minor) details of the design. Your input was invaluable, and I learned a lot during that time. The same goes for Alessio Lugnan at Ghent University, who not only allowed to run design rule checks many times (and even during that one deadline evening!) and suggested corrections but also was an essential link between us and the foundry. And thank you to those who helped with the simulation of resonators: my labmate Luca Aimone Giggio for microrings and, once again, Alfredo for photonic crystals.

I am also indebted to the POST-DIGITAL consortium, which has provided an intellectually stimulating environment. Workshops they organized alongside other events have been a source of inspiration and broke an occasionally onsetting monotony.

Moving to another country is never easy, and the 'paperasse' has been a significant obstacle to overcome. I thank Arnaud Brignon, Veronique Hauterre and the project manager Adriana Svetozarova for keeping a good chunk of it off my back and to everyone who advised me on navigating the sea of French bureaucracy.

And last but not least, I thank my family. It is not easy to part ways for years with few opportunities to reunite, considering these three years. Yet, they still supported my decision to pursue a degree far away. Knowing there is a home to return to made such a journey that much easier.

One final time: thank you, everyone.



## Nomenclature

$\alpha_{\text{loss}}$	Fiber attenuation	$\mathbf{x}$	Reservoir state vector
$\beta$	Propagation constant	$\mathbf{K}^{\text{in}}$	Input waveguide-cavity coupling coefficients
$\beta^{\text{se}}$	Spontaneous emission factor	$\mathbf{K}^{\text{out}}$	Output waveguide coupling coefficients
$\beta^{\text{TPA}}$	Two-photon absorption coefficient	$\mathbf{U}(\mathbf{r})$	Normal mode
$\beta_2$	Dispersion coefficient	$a(t)$	Complex envelope of a mode
$\gamma$	Fiber nonlinear parameter	$c$	Speed of light
$\gamma^{\kappa}$	Optical loss to waveguide	$f(x)$	Activation function
$\gamma^{\text{c}}$	Free-electron recombination rate	$F_{\Gamma}$	Normalized average supermode spacing, Eq. (3.29)
$\gamma^{\text{o}}$	Intrinsic optical loss rate	$L_{\text{p}}$	Power penalty
$\gamma^{\text{S}}$	Average supermode bandwidth, Eq. (3.25)	$n$	Refractive index
$\gamma^{\text{TPA}}$	Two-photon absorption rate, Eq. (3.9)	$n_2$	Kerr coefficient
$\hat{J}$	Linearized Jacobian, Eq. (3.32)	$N_{\parallel/\perp}$	Reservoir grid size parallel/perpendicular to input waveguide, see Figure 3.6
$\hat{M}^{\kappa}$	Waveguide-mediated coupling matrix	$q$	Elementary charge
$\hat{M}^{\mu}$	Evanescent coupling matrix	$r$	Regularization parameter for ridge regression
$\hat{W}^{\text{out}}$	Output weight matrix	$u$	Input signal
$\kappa$	Waveguide-cavity coupling strength	$V$	Resonator volume
$\kappa_{\text{in/out}}$	Input/output waveguide-cavity coupling strength	$V^{\text{c}}$	Free carrier diffusion volume
$\mu$	Evanescent coupling strength	$V^{\text{Kerr}}$	Effective Kerr volume
$\mu_0$	Permeability of free space	$V^{\text{TPA}}$	Effective two-photon absorption volume, Eq. (3.10)
$\omega$	Resonant frequency of a cavity	$y^{\text{tgt}}$	Target output signal
$\sigma(x)$	Sigmoid function, Eq. (2.2)	$d\omega/dN$	Free-carrier dispersion coefficient
$\sigma_{\text{t}/n}$	Standard deviation over time/index		
$\varepsilon_0$	Permittivity of free space		



## List of acronyms and abbreviations

### A

**ACSN** Abstract continuous spiking neuron  
**ADC** Analog-to-digital converter  
**AI** Artificial intelligence  
**ANN** Artificial neural network  
**AWG** Arbitrary waveform generator

### B

**BM** Boltzmann machine  
**BP** Backpropagation  
**BPTT** Backpropagation through time  
**BW** Bandwidth

### C

**CMOS** Complementary metal-oxide-semiconductor  
**CMT** Coupled mode theory  
**CNN** Convolutional neural network  
**CPU** Central processing unit

### D

**DAC** Digital-to-analog Converter  
**DBP** Digital backpropagation  
**DFA** Direct feedback alignment  
**DFB** Distributed-feedback (laser)  
**DMD** Digital micromirror device  
**DNN** Deep neural network

### E

**EM** Electromagnetic  
**ESN** Echo state network  
**ESP** Echo state property

### F

**FA** Feedback alignment  
**FCD** Free-carrier dispersion

**FDTD** Finite-difference time-domain

**FLOPS** Floating point operation per second

**FNN** Feed-forward neural network

**FORCE** First-order reduced and corrected error

**FPGA** Field-programmable gate array

**FSR** Free spectral range

**FWHM** Full width at half maximum

### G

**GPU** Graphics processing unit

### I

**ISI** Interspike interval

### L

**LIF** Leaky integrate-and-fire (neuron)

**LSM** Liquid state machine

**LSTM** Long short-term memory

### M

**MAC** Multiply-accumulate (operation)

**MC** Memory capacity

**MEMS** Micro-electro-mechanical systems

**MLP** Multilayer perceptron

**MLSE** Maximum likelihood sequence estimation

**MMI** Multimode interferometer

**MRR** Microring resonator

**MVM** Matrix-vector multiplication

**MVP** Matrix-vector product

**MZI** Mach-Zehnder interferometer

### N

**NARMA** Nonlinear autoregressive-moving-average

**NN** Neural network

**NRMSE** Normalized root mean square error

### O

**OEO** Optical-electrical-optical (conversion)

**OOK** On-off keying

## **P**

**PCA** Principal component analysis

**PCM** Phase-change material

**PhC** Photonic crystal

**PSN** Photonic spiking neuron

**PSP** Postsynaptic potential

## **Q**

**QAM** Quadrature amplitude modulation

## **R**

**RBM** Restricted Boltzmann machine

**RBW** Reservoir bandwidth Eq. (3.27)

**RC** Reservoir computing(-er)

**RF** Radio frequency

**RMS** Root mean square

**RNN** Recurrent neural network

**RRCF** Root-raised cosine filter

## **S**

**SER** Symbol error rate

**SNN** Spiking neural network

**SNR** Signal-to-noise ratio

**SOA** Semiconductor optical amplifier

**SOI** Silicon on insulator

**SSMF** Standard single-mode fiber

## **T**

**TIA** Transimpedance amplifier

**TPA** Two-photon absorption

**TPU** Tensor processing unit

## **V**

**VCSEL** Vertical-cavity surface-emitting laser

## **W**

**WDM** Wavelength division multiplexing

## **X**

**XOR** Exclusive OR operation

## List of Figures

1.1	Training compute requirements of milestone machine learning systems over time. Adapted from [6]. . . . .	3
1.2	Improvements of CPUs over time. Here SpecINT is a CPU integer processing power benchmark. Adapted from [7]. . . . .	3
1.3	Von Neumann architecture, the core of an overwhelming majority of modern digital computers. Separation of compute and memory modules is the source of the ‘von Neumann bottleneck’ limiting computing performance. Adapted from [11]. . . . .	4
1.4	Latency in terms of technology node for two different interconnect lengths [15]. Here CNT – carbon nanotube, $l_0$ – mean free path, i.e. distance travelled by electrons between scattering. . . . .	5
1.5	Nonlinear distortion compensation after 2800 km of fiber using a neural network. Here optical signal is modulated into 16 combinations of phase and amplitude that lay on a square grid. Figures shows samples of optical signal after recovery without (left) and with (right) a NN with 12 neurons. In the perfect case samples end up perfectly in corresponding dots. Alongside the signal samples, additional information about the link was passed as inputs. Adapted from [17]. . . . .	6
1.6	Evolution of optical communications. Adapted from [25]. . . . .	8
1.7	Radio-frequency fingerprinting facilitated by a photonic RNN. Here, residual data is a difference between the received raw data and the ground truth data, the signal regenerated after decoding the transmission. Redrawn based on [22]. . . . .	9
2.1	Biological neuron. Adapted from [35]. . . . .	13
2.2	Perceptron. Redrawn based on Figure 2.3. . . . .	14
2.3	Multilayer perceptron. Adapted from [39]. . . . .	14
2.4	Deep neural network. Redrawn based on Figure 2.3. . . . .	14
2.5	Echo State Network [46]. . . . .	15
2.6	Long short-term memory unit. Adapted from [53]. . . . .	17
2.7	Training perceptron to classify animals based on size and domestication. As more data points are added, the decision boundary encoded by perceptron weights and bias is updated. Adapted from [59]. . . . .	19
2.8	Comparison of Feedback Alignment (FA) and Direct Feedback Alignment (DFA) to back-propagation (BP). Adapted from [72]. . . . .	20
2.9	Unfolding a recurrent neural network (top) into a series of feed-forward neural networks (bottom). Here $x_k, u_k$ and $y_k$ are the network’s state, input and output at time $k$ , $f$ and $g$ are activation functions in the recurrent and feed-forward layers. . . . .	21

2.10	Overfitting example. A model is trained to separate two classes of points. An overfitted model (green line) also separates irregularities that arise, e.g., due to a noisy measurement. A properly generalizing model is shown as the black line. Inset shows a typical error landscape of an overfitting model: at some point, more training reduces error on the training dataset but increases on the testing one due to a loss of generalization capability. Adapted from [75]. . . . .	22
2.11	Some tasks cannot be solved with a linear regression. Left figure shows two classes of points on a plane. They can be separated with a circle, which is a nonlinear classifier. Instead, inputs can be mapped to a higher-dimensional space where a linear separation might be possible. Adapted from [82]. . . . .	25
2.12	Example of the Principal Component Analysis. A dataset is shown as dots, and arrows show two principal components with the arrow length proportional to the variance of data along its direction. . . . .	25
2.13	Graphical explanation of the Lyapunov exponent. Exponential divergence of evolution trajectories $x(t)$ and $x(t) + \delta(t)$ starting from close initial conditions $x(0)$ and $x(0) + \delta(0)$ is a marker of chaoticity. . . . .	26
2.14	An increase of memory capacity near the synchronization index border, but not of the Lyapunov coefficient. Adapted from [86]. Figures share vertical axes. . . . .	27
2.15	Timeline of publications concerning neural networks on the integrated photonic platform. Here MPLC is multi-plane light conversion, MZI is a Mach-Zehnder interferometer. Adapted from [90]. . . . .	28
2.16	Coherent matrix-vector product accelerator on the silicon-on-insulator platform based on 56 Mach-Zehnder interferometers and 213 phase shifters [91]. . . . .	30
2.17	Incoherent matrix-vector product accelerator, details in text. Here, ADC is an analog-to-digital converter. Adapted from [92]. . . . .	31
2.18	Incoherent matrix-vector product accelerator based on a photonic crossbar array. Here, CW is a continuous wave, VOA is a variable optical attenuator, DE-/MUX is de-/multiplexing, PD is a photodiode, and DSP is digital signal processing. Adapted from [93]. . . . .	31
2.19	Matrix multiplication accelerator based on the photorefractive effect. Adapted from [96].	32
2.20	Deep neural networks on chip. Left: one hidden and output layers [98]. Right: input layer (red), convolutional layer with four neurons (orange), one hidden layer with three neurons (purple) and output layer with two neurons (yellow). Adapted from [97]. . . . .	33
2.21	Proposal of a recurrent neural network on the integrated photonic platform. The output of a photonic weight bank is looped back as an input with a delay. Here, the delay is large enough (1 ns) to avoid time-delayed dynamics. Adapted from [99]. . . . .	33
2.22	Time-multiplexed reservoir computing scheme. Adapted from the arXiv source <sup>3</sup> of [103] due to higher quality. . . . .	35
2.23	Time-multiplexed photonic reservoir computer implementation based on optoelectronic feedback. Adapted from [104]. . . . .	35
2.24	Left: integrated photonic time-multiplexed reservoir computer based on a distributed feedback laser [106]. Right: improvement of the design by using multiple delay lines [107].	36

2.25	Left: scheme of a frequency-multiplexed reservoir computer. Right: spectrum of a signal in optical fiber. Adapted from [110]. . . . .	37
2.26	Left: swirl topology, only 4×4 is shown, but the actual size is 5×5 [112] or, later, 9×9 [114]. Right: semiconductor optical amplifier (SOA) nonlinearity steady-state characteristic compared to a hyperbolic tangent. Adapted from [114]. . . . .	38
2.27	Integrated reservoir computer with whirl (left) and four-port (right) topology on the silicon-on-insulator platform. Adapted from [116, 30]. . . . .	40
2.28	Microring resonator-based reservoir computer in swirl [118] (left) and rainfall [24] (right) topologies. . . . .	41
2.29	Integrated photonic reservoir based on a photonic crystal cavity. Adapted from [31]. . . . .	41
2.30	Reservoir computer based on a large area vertical cavity surface emitting laser. Here, DMD is a digital micromirror device, MMF is a multimode fiber, and DET is a photodetector. Adapted from [119]. . . . .	42
2.31	Spatio-temporal reservoir computer based on an array of coherent cavities. Here, VOA is a variable optical attenuator, PS is a phase shifter and PD is a photodetector. Adapted from [76]. . . . .	42
2.32	Left: recurring neural network in free-space optics based on a spatial light modulator (SLM) and digital micromirror device (DMD) that implement internal and readout weights. Here, DET is a detector, SLM is a spatial light modulator, (P)BS is a (polarizing) beam splitter, MO is a microscope objective, DOE is a diffractive optical element, Right: deep neural network based on layers of diffractive planes. Each point on a layer is a neuron with complex-valued transmission, and interference of the diffracted light creates coupling. Adapted from [120, 122]. . . . .	43
2.33	Overview of integrated neuromorphic setups. Details are given in Table 2.1. The work of Khoram E. et al. [126] is a photonic scatterer used for inference. . . . .	45
2.34	Swirl reservoir connected to a fully optical readout. Here, optical modulators (OM) perform weighting and the combiner structure – summation. Adapted from [131]. . . . .	47
3.1	Fabry-Perot (left), microdisk (center) and photonic crystal (right) resonators seen from the top. Red arrows represent the direction of electrooptical wave propagation. . . . .	50
3.2	Linear absorption and two-photon absorption on a semiconductor band diagram. Left: photon energy $\hbar\omega$ is higher than bandgap $E_g$ . A photon (red) is absorbed by an electron (blue) in the valence band, moving it to the conduction band. Excess energy is dissipated due to collisions with the crystal lattice and other electrons. An electron can return to the valence band in various ways, but in nanostructures, surface recombination is dominant and shown in figure. The surface is a lattice defect, which creates states inside the band gap. An electron can fall into one of these, emitting a phonon. After a few surface states, the electron returns to the valence band. Right: $\hbar\omega < E_g \leq 2\hbar\omega$ , a single photon is not enough to excite an electron, but two can be absorbed simultaneously. . . . .	52
3.3	Coupling of microring (left) and photonic crystal (right) resonator modes to a waveguide. Mode directions are shown as colors. The photonic crystal mode is the standing-wave type and is a superposition of back and forward-propagating modes. . . . .	54
3.4	Cavity mode overlap induces an interaction between modes. . . . .	55

3.5	Complex mode envelopes of two photonic crystal cavities (blue, orange) coupled evanescently (left) and by a waveguide (center, right). Solid and dashed show real and imaginary parts. Center and right demonstrate a ‘good’ and a ‘bad’ phase conditions. Time is normalized by $2\pi/\mu$ (left) and $2\pi/\kappa^2$ (center, right). . . . .	56
3.6	Geometry of the proposed integrated photonic reservoir computer. . . . .	57
3.7	Integration convergence validation of a reservoir with default parameters (see Table 3.1). Left: integration step normalized by the reservoir timescale. A set of identical reservoirs with identical random input is integrated with various time steps. The result with the smallest step is assumed to be accurate and serves as the baseline. Integration error is computed for a trajectory of each cavity w.r.t. the corresponding cavity of the baseline, and the error is normalized by its mean amplitude over the integration period. Each box represents the largest error across all cavities at each integration step. Right: each step shows an average error across all cavities during this step. The integration step is the largest present in the left figure. Time is normalized by the inverse reservoir bandwidth. The error does not increase even after a long computation. . . . .	60
3.8	Visual representation of signal bandwidth BW. . . . .	60
3.9	Amplitudes of principal components reservoir designed according to the following sections. Here, the drop-off is rather smooth and a change of the amplitude cut-off would not change the resulting dimensionality drastically. . . . .	61
3.10	Supermodes of evanescently coupled microcavities for various grid geometries. Bottom figure shows a histogram of supermode frequencies. . . . .	63
3.11	Dimensionality of linear photonic reservoir w.r.t. input signal bandwidth. Reservoir parameters are given in Table 3.1 but $\mu$ is increased to 80 GHz to increase the upper limit of dimensionality. . . . .	64
3.12	Dimensionality of linear photonic reservoir w.r.t. average supermode spacing. Reservoir parameters are given in Table 3.1 and $BW > RBW$ . Each point corresponds to specific $\mu$ and $\gamma^o$ . . . . .	64
3.13	Left: impact of $\kappa_{in}^2/\gamma^o$ to dimensionality. Reservoir parameters are given in Table 3.1 and $BW > RBW$ , legend shows $N_{  } \times N_{\perp}$ , reservoir geometry. Right: $N_{  } \times N_{\perp}$ reservoir geometry. . . . .	65
3.14	(a) Scalability of linear reservoir dimensionality w.r.t. optical losses. Dimensionalities of all possible combinations of $N_{  }$ and $N_{\perp}$ for given $N$ are shown as histograms. Reservoir parameters are shown in Table 3.1, with $\mu = 80$ GHz, reduced $\kappa_{out}$ and $BW > RBW$ . (b) Impact of $F_T$ and grid geometry on dimensionality. Both figures use the same dataset. . . . .	66
3.15	Comparison of material platforms w.r.t. $P_{NL}$ and an operating frequency of a single MRR or PhC cavity. Solid and dashed lines show free carrier dispersion- (FCD) and two-photon absorption-dominated regimes, respectively. The dotted line continues the FCD line beyond the achieved electron recombination rate, details in text. Here, ALD is atomic layer deposition. For references, see Table 3.2. . . . .	68
3.16	A case when nonlinearity is considerable, but its variation over time is not. Top: input signal, a sum of a constant signal with a weaker random component. Bottom: two-photon absorption rate of two coupled cavities normalized by average supermode bandwidth. . . . .	70

3.17	Scaling of power necessary to induce a nonlinear response in a reservoir w.r.t. the number of cavities. Reservoir parameters are given in Table 3.1. Signal bandwidth matches reservoir bandwidth. . . . .	71
3.18	Effect of (a) two-photon absorption (b) same with free-carrier dispersion (c) the Kerr effect on reservoir dimensionality and consistency. Reservoir parameters are given in Table 3.1. . . . .	71
3.19	Dimensionality reduction caused by two-photon absorption. Spans show the immediate bandwidth of all supermodes, i.e. real part of the linearized Jacobian matrix, of a reservoir computer with three cavities around corresponding eigenfrequency. The reservoir is excited with a random input with a high enough bandwidth and increasingly higher power. Two-photon absorption causes supermodes to broaden, erasing their differences. For the first nanosecond, the Principal Component Analysis returns 5, but during the last nanosecond – 4. . . . .	72
3.20	Effect of nonlinearities (shown in legend) on reservoir dimensionality at various input bandwidths. Each point corresponds to an input power at which consistency is 0.98. . . . .	73
3.21	Dimensionality improvement caused by a resonance-shifting nonlinearity. Left: time traces of linearized Jacobian matrix eigenfrequencies of linear (dashed) and nonlinear (solid) otherwise identical reservoirs with three cavities. Inputs are identical for both reservoirs – a random signal with a spectrum shown in the right figure. In the linear case, only blue and orange supermodes are excited, and the Principal Component Analysis returns a dimensionality of 4. With frequency shift, the green supermode can also accept input, and the dimensionality becomes 6, the maximum. . . . .	73
3.22	Neuron connectivity graph of (a) a typical Echo-State Network and (b) a photonic reservoir computer with a grid of cavities. Connection strength is represented by arrow thickness. . . . .	74
3.23	Left: linearized Jacobian eigenvalues of an Echo State Network during a typical computation, adapted from [160]. Right: same for the proposed reservoir computer with various nonlinearities (see Section 4.1 for details). . . . .	75
3.24	Impact of cavity resonance frequency disorder on dimensionality. Resonance disorder (in regular frequency units) is normalized to evanescent coupling strength Here, std. is the standard deviation. Reservoir parameters given in Table 3.1. Input signal bandwidth is higher than reservoir bandwidth. Each dot corresponds to a $N_{\parallel} \times N_{\perp}$ (shown in legend) linear microring resonator-based reservoir with resonances $\omega_k \in \mathcal{N}(\omega_0, \sigma_{\omega})$ with various $\sigma_{\omega}$ . Insets represent supermode intensity distribution over cavities $\hat{\Lambda}$ . . . . .	78
3.25	Histogram of reservoir supermode frequencies with evanescent coupling strength normally distributed around $\mu$ with the standard deviation of $0.2\mu$ . . . . .	79

3.26	Finite-difference time-domain (FDTD) simulation results of a microring resonator (MRR). Top-left: evanescent coupling strength as a function of their distance. Exponential fit is used due to an exponential reduction of mode amplitude outside a resonator. Top-right: waveguide (WG) coupling strength as a function of its width; here distance to the waveguide is 200 nm, and a resonant behaviour can be seen. Bottom: same, w.r.t. its distance to a resonator. Here, the waveguide width is 550 nm and 'loss to WG' corresponds to $\kappa^2/2$ . . . . .	80
3.27	Design of used one-dimensional photonic crystal coupled to a waveguide. Gray areas show silicon region, white is silicon oxide and red shows the resonator mode's electromagnetic field distribution. Here, only nine sections of silicon are drawn for simplicity; the actual structure includes a few tens. . . . .	81
3.28	Simulation results of a photonic crystal resonator. Left: Q-factor as a function of the number of waveguide segments in a cavity. Right: loaded Q-factor as a function of the distance between the longitudinal resonator axis and the axis of the coupled waveguide, assuming its width is 450 $\mu\text{m}$ . . . . .	81
3.29	Layout of a microring (left) and a photonic crystal (right) resonators. Etch region is shown in black, blue and violet are two metallization layers; their interconnects are shown in green. Orange shows tungsten heaters (connected to the violet layer). . . . .	82
3.30	Layout of the reservoir computer based on microrings (left) and photonic crystal (right) resonators. Etch regions are shown in black. The microring-based reservoir has a staggered design (see Figure 3.31). . . . .	82
3.31	Staggered reservoir design with every second cavity in the first row coupled to the waveguide (blue, the arrow shows input signal direction) to excite only one of the counter-propagating modes, their direction shown as arrows. While using both modes can double the dimensionality, two waveguides per readout microring would be required to harness it, which was challenging under given constraints. Therefore, we opted for such a design so as not to waste input power to excite unused modes. . . . .	82
3.32	Complete reservoir computer layout; the spiral at the bottom is part of an unrelated design. Black shows etch regions, blue shows metallization layers and pink shows foundry-proposed phase shifters, MMIs and couplers. Most of the chip edge was taken by contact (square) pads. . . . .	83
3.33	Heater wiring scheme of a microring resonator-based reservoir. Etch region is shown in black, blue and violet are two metallization layers with green showing their interconnects, orange shows tungsten heaters (connected to the violet layer). . . . .	84
3.34	Top-left: entire chip photo. Top-right: ring resonator as an amplitude controller. Photos of microring (center) and photonic crystal (bottom) resonator-based reservoir computers on a chip. . . . .	85
4.1	Solution of the Mackey-Glass equation Eq. (4.1) with parameters in text. . . . .	88

4.2	The normalized Fourier transform of the input signal (blue) with reservoir resonances overlaid (orange). Frequency is normalized by the reservoir bandwidth. Here, the reservoir bandwidth is larger than the input bandwidth, which goes against the conclusions of Section 3.6. High peaks in the input spectra correspond to a dominating harmonic ( $\xi(t)$ largely oscillates – see Figure 4.1); however, for a close prediction, it is important to capture high-frequency irregularities despite their weak amplitude. . . . .	90
4.3	Impact of nonlinearities on (a) prediction error and (b) power penalty on a 3-step Mackey-Glass prediction. For all cases, GaAs microring parameters are used (see Table 3.1), but for the TPA-only regime, $d\omega/dN$ is set to zero, while for the Kerr effect $\beta^{\text{TPA}} = 0$ and $d\omega/dI < 0$ is chosen arbitrarily. The regularization constant was increased until the power penalty reached -25 dB. . . . .	91
4.4	3-step Mackey-Glass prediction by an $8 \times 3$ GaAs microring resonator at an optimal input power. Here, the NRMSE is 0.15 and $L_p \approx -22$ dB. . . . .	91
4.5	Scalability of 3-step prediction performance with various geometries and (a) TPA, (b) TPA with FCD and (c) the Kerr effect. The regularization constant was increased for each point until the power penalty reached -25 dB. Here, minimum NRMSE (a) 0.037 (b) 0.070 (c) 0.087. . . . .	92
4.6	Non- and return-to-zero on-off keying modulation. Adapted from [183]. . . . .	94
4.7	Constellations of various quadrature amplitude modulation formats on a complex plane. Adapted from [185]. . . . .	95
4.8	Impulse response function of a raised- (top) and a root raised-cosine filters (bottom). The legend shows the roll-off factor; a lower value corresponds to a narrower bandwidth. Raised cosine passes through zero every bit length, i.e. samples are unchanged. . . . .	97
4.9	Scheme of an optical fiber link. Electrical and optical components are shown in black and red. Here, RC is the reservoir computer, and RRCF is the root-raised cosine filter. . . . .	97
4.10	Top: timetraces of a real part of the original signal (dashed line) and the distorted signal after 150 km of optical fiber (solid line). Black dots show a bit middle points and act as the training target; blue and orange are only for visualization and show distorted signal samples with color corresponding to each symbol. Bottom: constellations of the distorted optical signal samples after propagating through a length of optical fiber given above each plot. Parameters are given in Table 4.2. . . . .	99
4.11	Normalized Fourier spectrum of the distorted signal after 150 km of fiber (blue) with $8 \times 3$ reservoir resonances overlaid (orange). Frequency is normalized by the reservoir bandwidth. . . . .	99
4.12	(a) Effect of output delay (shown in legend) on output symbol separability measured with NRMSE at a given power penalty, controlled with regularization parameter. Here, fiber length is 150 km, and the signal is processed with an $8 \times 3$ linear reservoir. (b)(c) Constellations of output symbols at power penalty of approximately -20 dB with 0- and 4-bit delay, respectively. Figure 4.10 (bottom) shows the constellation of input signal samples. . . . .	100

4.13	Performance scaling of optical signal recovery w.r.t. reservoir size. The reservoir is (a) $N$ independent, (b) $N_{\parallel} \times 1$ grid and (c) $N_{\parallel} \times 3$ grid of linear microring resonators. The regularization parameter is increased for each computation until the power penalty is no less than -25 dB. Here, SER is the symbol error rate. . . . .	100
4.14	Constellations of optical signal samples after propagating through a length of optical fiber given above each plot. Parameters are given in the left column of Table 4.3. In the left figure constellation are simply dots enhanced for visibility. . . . .	102
4.15	Constellation (left) and histogram (right) of signal samples after 80 km of optical fiber recovered with a linear reservoir. The power penalty is -12 dB. Figure 4.14 shows the distorted signal. . . . .	102
4.16	Constellations of optical signal samples in highly nonlinear case after propagating through a length of optical fiber given above each plot. Parameters are given in the right column of Table 4.3. In the left figure constellation are simply dots enhanced for visibility. . . . .	103
4.17	Recovery of strongly nonlinearly distorted signal after 30 km of optical fiber with linear and nonlinear otherwise identical reservoirs. Each pair shows a constellation and a histogram of samples. The power penalty is around -20 dB in both cases. Figure 4.16 shows the distorted signal. . . . .	103
5.1	Excitability of a giant squid neuron, vertical axis shows neuron voltage [192]. Weak inputs induce a linear response, but stronger responses elicit a spike due to ionic currents inside the cell membrane. Here, the excitation strength only impacts the spike delay. .	105
5.2	Examples of spiking photonic systems. Top left: injection-locking setup [208]. Top right: distributed-feedback laser, adapted from [212]. Bottom left: photonic crystal with saturable absorber [214]. Bottom right: vertical-cavity surface-emitting laser with saturable absorber [209]. . . . .	108
5.3	Photo (left) and schematic (center, right) of a hybrid one-dimensional photonic crystal InP nanolaser heterogeneously integrated on top of a silicon-on-insulator waveguide. Here BCB is benzocyclobutene. Adapted from [213]. . . . .	108
5.4	Statistical properties of an abstract continuous spiking neuron [218]. . . . .	111
5.5	Sampling of four continuous spiking neurons. Black bars show spiking events and gray rectangles – neuron activation interval (i.e. when $z_i = 1$ ). Reproduced based on [219]. .	112
5.6	Sampling from a Boltzmann Machine with a network of leaky integrate-and-fire neurons. (a) Spike pattern of 5 such neurons during sampling. (b) Comparison of sampled distribution (blue) and target distribution (red). (c) Increase of sampling accuracy with more samples collected, ten trials. (d) Sampling accuracy for a set of random Boltzmann Machines with $10^6$ ms of sampling time. Figure is taken from [219]. . . . .	112
5.7	Stochastic inference of three digits from an image with leaky integrate-and-fire neurons. Left: when no input is imposed, the Boltzmann Machine infers three digits with equal probability. Right: when the it is exposed to an ambiguous input. Adapted from [219]. .	113
5.8	Resonator mode over gain and saturable absorption sections. The Gaussian mode shape in the middle of a cavity with a given mode size is assumed. Here, FWHM is full width at half maximum. Adapted from [214]. . . . .	115

5.9	Quantum well gain curve, the differential gain slowly reduces as more charge carriers are added to a cavity [214]. . . . .	115
5.10	Bifurcation diagram of a photonic spiking neuron w.r.t. to injection power. . . . .	118
5.11	Spiking at $\mathcal{I} = 7.5$ , dashed line shows $G_{\text{tgt}}$ . (a) Interplay of gain and normalized number of photons during a spike emission. The number of photons corresponds to $P$ . (b) Gain recovery after a large drop caused by a spike emission. (c) Random walk of gain between spikes. Note y-axis scaling. . . . .	119
5.12	Stochastic properties of a photonic spiking neuron. (a) Autocorrelation of gain time traces below and above the self-spiking threshold ( $\mathcal{I} = 7.2$ and $7.5$ ) and of the Ornstein-Uhlenbeck (OU) process. (b) Histograms of gain values, dashed line shows $G_{\text{tgt}}$ . (c) Histogram of interspike interval at $\mathcal{I} = 7.5$ . . . . .	120
5.13	Left: sigmoid fit of a provisional PSN activation curve. A dotted line shows $G_{\text{tgt},0}$ . Poor fit quality suggests that a more careful approach is required. Right: gain time trace with $G_{\text{tgt}} \approx -0.07$ shown as a dashed line. . . . .	121
5.14	Sigmoid fit of an improved photonic spiking neuron activation curve inside the shaded region. A dotted line shows $G_{\text{tgt},0}$ . Here BM is Boltzmann machine. . . . .	121
5.15	Probability of second spike emission w.r.t. delay between the first and the second excitations. The legend shows the power of the second excitation relative to the strength of the first. Right: examples of refractory functions of absolute and relative refractory mechanisms. Generally speaking, the curve of a relative mechanism can have an arbitrary shape. . . . .	122
5.16	Left: various postsynaptic potential models used in literature normalized by the area under the curve. Right: photonic spiking neuron gain upon receiving a spike from another such neuron. Here, for clarity, the second neuron has $\mathcal{I} = 7.1$ , which is relatively far below the self-spiking threshold so that it can endure stronger excitations without emitting a spike. The spike's impact on gain is weaker in actual computations. . . . .	124
5.17	Impact of the relative refractory mechanism and non-rectangular post-synaptic potential on sampling accuracy. Top: impact in terms of the Kullback-Leibler divergence. Bottom: visual comparison of sampling accuracy with the alpha-shaped postsynaptic potential, rows of dots represent all possible states the networks can be in, and each dot is an active neuron. Adapted from [218]. . . . .	125
5.18	Behaviour of photonic spiking neurons (dots) and Boltzmann Machine neurons (lines) w.r.t. global scaling of their connection weights for various $\tau_U/\tau_{\text{on}}$ , from left to right: 0.2, 0.4, 0.8 and 1.6. Corresponding neurons from each system are shown with a unique color. . . . .	126
5.19	Comparing a Boltzmann Machine (BM) neuron and photonic spiking neuron (PSN) behaviour w.r.t. weight scaling. Here, PSN weights are scaled to BM weights according to Eq. (5.32). . . . .	127
5.20	Probability density function of the Beta-distribution $\mathcal{B}(0.5, 0.5)$ . . . . .	128

5.21	Sampling of a Boltzmann Machine with 5 neurons with a network of 5 photonic spiking neurons. Here $b_i \sim 1.2[(0.5, 0.5) - 0.5]$ . Left column: $W_{ij} \sim 1.2[(0.5, 0.5) - 0.5]$ . Right column: $W_{ij} \sim 2.4[(0.5, 0.5) - 0.5]$ . Top: visual comparison between sampled and true probability distributions. Here circle columns represent $z$ : black dots correspond to $z_k = 1$ and white otherwise. Bottom: sampling accuracy improvement with more samples collected. . . . .	128
A.1	Waveguide-resonator side-coupling from two perspectives. Vertical dotted lines in left figure show reference planes containing the waveguide-cavity interaction region. . . . .	138
B.1	Scheme of waveguide-cavity interaction. A rectangle with $a_n$ represents a row of $n$ cavities coupled to the waveguide. . . . .	145
C.1	Group velocity dispersion parameters for two photonic crystal fibers and SMF-28 [232].	150
C.2	Spectral and temporal evolution of a $\text{sech}^2$ pulse at 10 kW peak power and 50 fs FWHM during propagation through a photonic crystal fiber. The top row shows results in [232] and the bottom row shows our reproduction. . . . .	151

## List of Tables

2.1	Details for Figure 2.33. For linear setups for which the processing speed was not discussed, a detection limit of 100 GHz is assumed according to [91]. . . . .	46
3.1	Default parameters of a linear reservoir with Eq. (3.12). Others are omitted as they only impact the input power. . . . .	59
3.2	References used in Figure 3.15. . . . .	69
4.1	Parameters of the SMF-28 optical fiber [188]. . . . .	96
4.2	Parameters for simulation of propagation through a metro optical link. . . . .	98
4.3	Parameters for simulation of propagation through a short-reach optical link in default and highly nonlinear cases. Fiber parameters are given in Table 4.1. . . . .	101
5.1	Simulation parameters for a single cavity [214]. See Figure 5.8 for cavity and mode size definitions. . . . .	118



# Contents

<b>1</b>	<b>General introduction</b>	<b>1</b>
1.1	Artificial neural networks . . . . .	1
1.2	Deep Learning Era and beyond . . . . .	2
1.3	Challenges of electronic computing . . . . .	2
1.4	High-speed neural networks . . . . .	5
1.4.1	Optical channel equalization . . . . .	6
1.4.2	Radio-frequency fingerprinting . . . . .	7
1.5	All-optical analog neuromorphic computing . . . . .	8
1.6	Objectives . . . . .	11
1.7	Thesis outline . . . . .	11
<b>2</b>	<b>Integrated neuromorphic computing</b>	<b>13</b>
2.1	Artificial neural networks . . . . .	13
2.2	Reservoir Computing . . . . .	15
2.3	Training neural networks . . . . .	17
2.3.1	Overview . . . . .	17
2.3.2	Offline reservoir training: ridge regression . . . . .	21
2.3.3	Regularization . . . . .	22
2.4	Reservoir performance tips and tricks . . . . .	23
2.4.1	Dimensionality . . . . .	24
2.4.2	Consistency . . . . .	26
2.5	Integrated photonics for neuromorphic computing . . . . .	27
2.5.1	Photonic accelerators . . . . .	29
2.5.2	Photonic neural networks . . . . .	32
2.5.3	Reservoir computers . . . . .	34
2.6	Scalability of integrated photonic neural networks . . . . .	43
2.6.1	A question of readout . . . . .	46
2.7	Conclusion . . . . .	48
<b>3</b>	<b>Integrated reservoir computer based on evanescent coupling</b>	<b>49</b>
3.1	Introduction . . . . .	49
3.2	Microresonator as a neuron . . . . .	49
3.2.1	Microresonator model . . . . .	50
3.2.2	Nonlinear effects . . . . .	51
3.3	Dynamical model of coupled nonlinear resonators . . . . .	54
3.3.1	Cavity-waveguide coupling . . . . .	54
3.3.2	Evanescent intercavity coupling . . . . .	55
3.3.3	Waveguide-mediated intercavity coupling . . . . .	55

3.3.4	Comparing waveguide-mediated coupling with evanescent	55
3.4	Reservoir computer master equation	56
3.5	Numerical simulation	58
3.5.1	Rotating wave approximation	58
3.5.2	Implementation	59
3.6	Dimensionality in the linear regime	59
3.6.1	Methodology	61
3.6.2	Spectral line spread of coupled resonators	62
3.6.3	Bandwidth matching	63
3.6.4	Spectral line separation	63
3.6.5	Effect of input waveguide coupling	65
3.6.6	Scalability of dimensionality	65
3.7	Nonlinear regime	66
3.7.1	Methodology	66
3.7.2	Nonlinear response strength	67
3.7.3	Results	70
3.8	Discussion	74
3.8.1	Differences with Echo State Machines	74
3.8.2	Integrated platforms and neuron density	75
3.8.3	Timescales	76
3.8.4	Readout	77
3.8.5	Resonance disorder	77
3.9	Evanescent coupling strength disorder	79
3.10	Prototype	79
3.11	Conclusions and outlook	84
<b>4</b>	<b>Solving tasks with an integrated all-optical reservoir</b>	<b>87</b>
4.1	Mackey-Glass prediction	87
4.1.1	Task description	87
4.1.2	Performance metrics	89
4.1.3	Results and discussion	89
4.1.4	Conclusions	92
4.2	Optical fiber data recovery	93
4.2.1	Signal transmission	94
4.2.2	Task description	95
4.2.3	Results and discussion	98
4.2.4	Conclusions	103
<b>5</b>	<b>Spiking neural network in integrated photonics</b>	<b>105</b>
5.1	Boltzmann Machine	109
5.1.1	Stochastic inference	113
5.1.2	Sampling from arbitrary distributions	113
5.2	Single laser model	114

5.3	Spiking nanolaser as abstract spiking neuron . . . . .	117
5.3.1	Spiking demonstration . . . . .	117
5.3.2	Activation curve of photonic spiking neurons . . . . .	120
5.3.3	Refractory mechanism . . . . .	122
5.4	Network of spiking nanolasers . . . . .	123
5.4.1	Postsynaptic potential . . . . .	123
5.4.2	Weight translation rule . . . . .	124
5.5	Photonic sampling . . . . .	127
5.6	Conclusion . . . . .	129
	<b>Conclusions and perspectives</b>	<b>131</b>
	<b>Résumé en français</b>	<b>135</b>
	<b>Appendices</b>	<b>137</b>
<b>A</b>	<b>Side-coupling cavity and waveguide</b>	<b>137</b>
A.1	Change of amplitude in space . . . . .	138
A.2	Forward-propagating mode . . . . .	140
A.3	Backward-propagating mode . . . . .	141
A.4	Energy conservation . . . . .	142
<b>B</b>	<b>Waveguide coupling matrices</b>	<b>145</b>
<b>C</b>	<b>Simulation of optical signal propagation in an optical fiber</b>	<b>149</b>
C.1	Split-step method . . . . .	149
C.2	Solver validation . . . . .	149
C.2.1	Recomputing dispersion coefficients . . . . .	149
C.2.2	Results . . . . .	151
	<b>Bibliography</b>	<b>151</b>

# 1 - General introduction

## 1.1 . Artificial neural networks

Since the birth of computer science as a field, many tasks have been solved using algorithms implemented on electronic computers. However, there was a large portion of problems that were notoriously difficult to tackle. A good example is image recognition: to recognise a dog, we might need to handle different breeds and pictures taken from different angles. Is a dog sitting, lying down, or running? What about animals that look similar to a dog? There are many sides to this problem; solving them by a deterministic feature extraction is a daunting challenge. Other examples of such tasks include keeping balance while walking on rough terrain, driving an automobile and performing medical analysis. Yet, we can solve such problems, given enough training (and, possibly, aptitude for a task).

Just as it happened with a sonar [1] and the simulated annealing optimization algorithm [2], by drawing inspiration from functional solutions given by nature, we can try to imitate them and harness millenia of evolution to our benefit. Thus, Artificial Intelligence and its subset Machine Learning appeared. They have been wildly successful in numerous applications, ranging from highly specific, such as distinguishing music and speech in the *Opus* music coding format [3], to everyday tasks, such as internet search suggestions.

Machine Learning is a general term that includes a wide range of computation approaches inspired by several sciences. For example, Boltzmann Machines (BM) take roots in statistical physics and are used to reconstruct information based on incomplete inputs, the Principal Component Analysis (PCA) is used to extract the most important components from the input. Another approach is probably the most widely known – artificial neural networks (ANN) inspired by neuroscience, which tries to like brains do – neuromorphic computing. Artificial neural networks began with the proposal of a perceptron and a subsequent multilayer perceptron (MLP) that was capable of solving nontrivial classification tasks and, eventually, Deep Neural Network (DNN), that laid the foundation for Deep Learning.

## 1.2 . Deep Learning Era and beyond

In the early 2000s, Deep Learning started overtaking other ANN types in performance. The further growth of Deep Learning accelerated for several reasons:

- a widespread use of the internet and social media has led to the availability of massive amounts of labeled data necessary for training neural networks,
- improvements of graphics- and tensor processing units (GPUs and TPUs) that can emulate large neural networks more efficiently than central processing units (CPUs),
- the development of widely accessible open-source libraries for training and using neural networks such as TensorFlow [4],
- increased use in a commercial setting.

All that led to a circle where more demand for Deep Learning attracted more people to the field, which led to more workforce for the development of better and more accessible tools, which simplified its adoption by more companies, creating more demand.

As the complexity of the tasks grew, they required more complex ANNs that necessitated more computing power to run and, especially, train. Until the 2000s, the computational complexity of neural networks doubled approximately every 22 months (see Figure 1.1). Interestingly, it coincides with Moore's law, which initially is a prediction and, subsequently, an empirically confirmed observation that the number of transistors on a chip doubles every two years [5] (see Figure 1.2), meaning that electronics and neural networks grew at the same rate. However, with the advent of Deep Learning, the complexity started doubling every seven months, quickly outpacing electronics. By splitting the computing load between several GPUs and TPUs, large networks could still be trained in a reasonable time. Eventually, the complexity grew to the point where the training process moved to centralized compute clusters (in the 'cloud'), marking the beginning of the Large Scale Era.

## 1.3 . Challenges of electronic computing

A human brain has approximately  $10^{11}$  neurons, and a neuron is, on average, connected to  $10^4$  neurons [8]. The Human Brain Project has estimated that to simulate so many components, one would need to perform on the order of  $10^{18}$  floating point operations per second (FLOPS) [9], which seems like a huge number, but this much compute has been achieved by the supercomputer *Frontier* [10]. The difference is that a brain consumes just 25 W of power

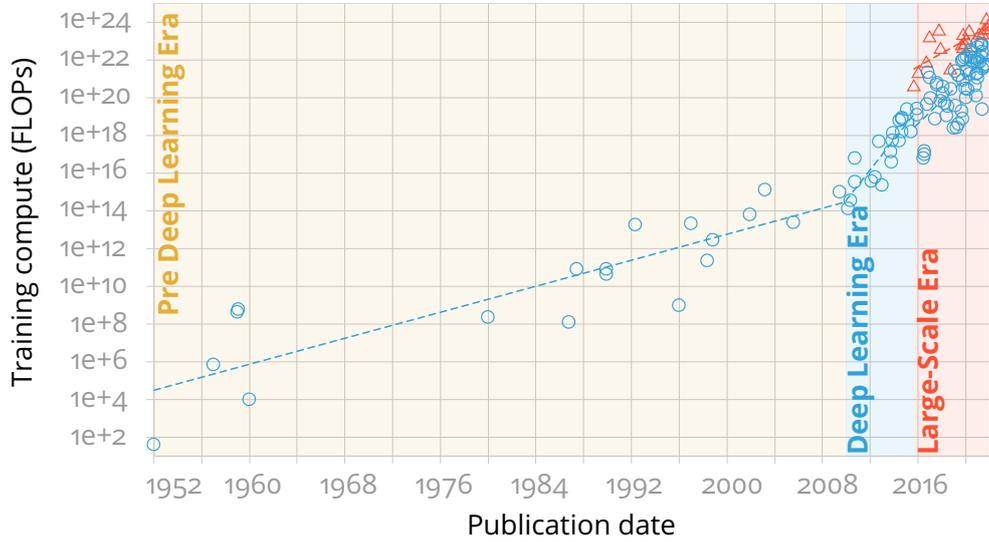


Figure 1.1: Training compute requirements of milestone machine learning systems over time. Adapted from [6].

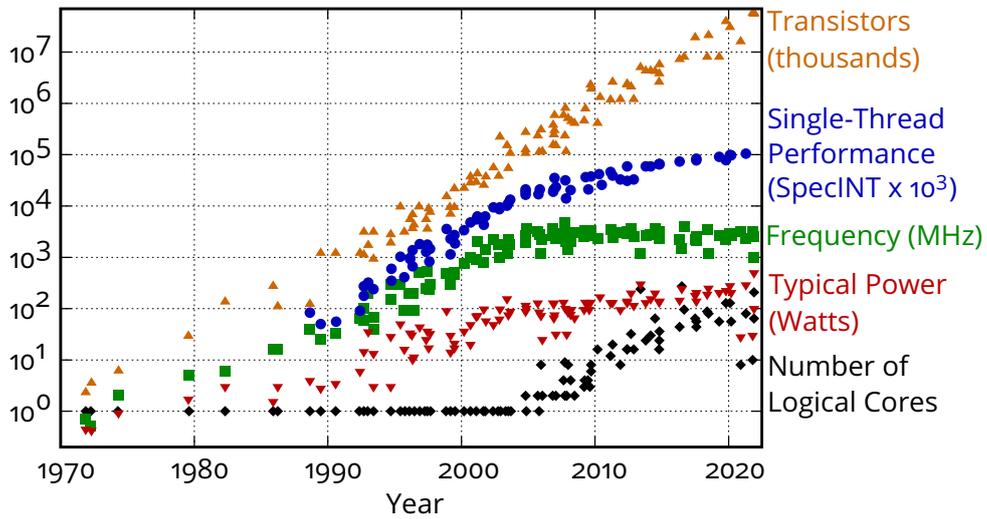


Figure 1.2: Improvements of CPUs over time. Here SpecINT is a CPU integer processing power benchmark. Adapted from [7].

and takes  $10^{-3} \text{ m}^3$  of space, whereas *Frontier* – 21 MW and  $10^3 \text{ m}^3$ , six orders of magnitude on both fronts.

The problem is that electronic computers are created to handle different tasks than our brains do and are, therefore, intrinsically ill-suited. Conventional computers are very good at handling sequential data processing given a specific algorithm. A CPU picks up pieces of data from a memory unit through a bus, performs necessary operations and pushes the result back to the memory unit (see Figure 1.3). Not only does the frequent exchange of data over a bus take time – the ‘von Neumann bottleneck’ – this is not how brains operate. If we were to perform brain-like computing on a conventional computer, a severe overhead takes place. In order to avoid such an outcome, we should discard preconceived notions of what computing is, consider more biologically plausible approaches, and design new hardware platforms with that in mind.

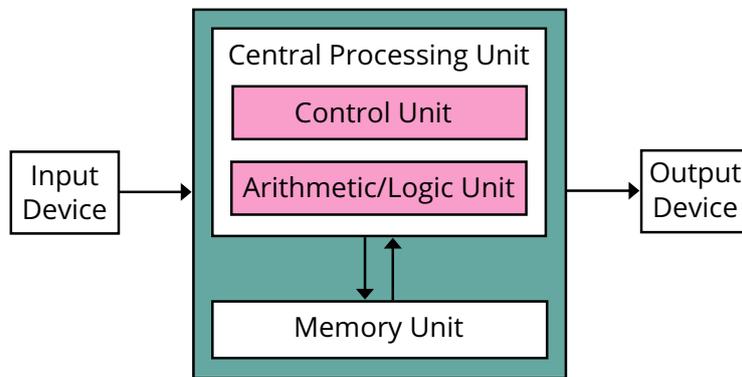


Figure 1.3: Von Neumann architecture, the core of an overwhelming majority of modern digital computers. Separation of compute and memory modules is the source of the ‘von Neumann bottleneck’ limiting computing performance. Adapted from [11].

Electronic hardware has other, more technological difficulties. First, it is hard to increase the clock speed any further (see Figure 1.2). The CPU components have capacitance; they need to be dis-/recharged to change their state. This process can be carried out faster with a smaller feature size (and therefore lower capacitance), which is what CPU manufacturers strive to achieve. However, it is becoming more and more challenging with every iteration. An alternative is to supply more electrical current to the chip, generating more heat. To remove that heat from a chip in a cost-effective manner, it should not exceed 200 W [12], and indeed, notice how the power consumption has plateaued around this value in Figure 1.2.

Interconnects are also a significant issue. The number of bits per second an electronic connection can transfer is limited by  $B \sim B_0 A / l^2$ , where  $B_0 = 10^{16}$  for small on-chip lines due to bulk resistive losses and losses induced

by the skin effect,  $A$  is the wire cross-section area, and  $l$  is the connection length. An interesting point here is that  $A/l^2$  is dimensionless, which means that even when wiring takes up the entire chip space, the overall bit rate will not increase with a simple change of the device size – the entire approach to signal transmission has to be changed [13]. Interconnects are also costly in terms of power, taking up to 80% of the total power consumption of a CPU [14]. Latency also becomes an issue at a smaller feature size, as a smaller cross-section increases the resistance of wires, increasing the time required to re-charge/discharge them (see Figure 1.4).

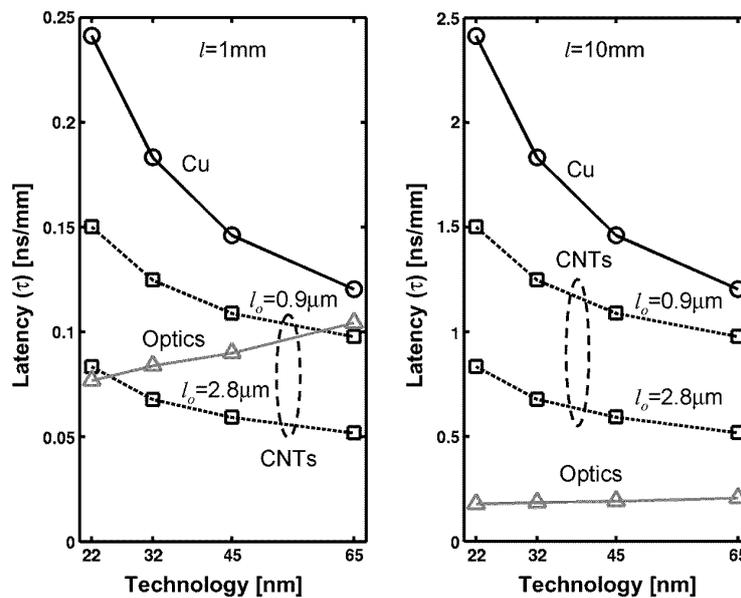


Figure 1.4: Latency in terms of technology node for two different interconnect lengths [15]. Here CNT – carbon nanotube,  $l_0$  – mean free path, i.e. distance travelled by electrons between scattering.

## 1.4 . High-speed neural networks

Many applications of neural networks do not require high speed and efficiency. ChatGPT generating a response in a couple of seconds is tolerable, and a couple of minutes is fine for Stable Diffusion to generate an image. Requiring a few hundred watts in the process is also acceptable. However, new applications for NNs are not as lenient.

### 1.4.1 . Optical channel equalization

In order to optically transmit information, it is encoded on an optical carrier by modulating its amplitude and, possibly, phase, and such modulated signal is subsequently passed through an optical fiber to reach the receiver (Section 4.2.1 explains in more detail). During propagation along the optical fiber, the signal is subject to chromatic dispersion and the Kerr effect, which distort the signal, so its postprocessing is required on the receiving end to remove these deteriorations – a common problem in the telecommunications field.

The chromatic dispersion by itself is not very complicated to compensate since it is a linear effect and so can be performed with a linear filter. However, when the optical signal in a fiber becomes strong enough, e.g., to mitigate optical losses during long-haul communication, the Kerr effect induces a nonlinear response, and the compensation becomes challenging. Current solutions include

- maximum likelihood sequence estimation (MLSE),
- Volterra-series-based equalizers, which is a nonlinear version of a finite response filter,
- digital back-propagation (DBP) that relies on solving the optical signal propagation equation in the backward direction.

These methods are computationally intensive, and processing signals at tens of GBaud with low delay and power use is challenging. While DBP is somewhat easier computing-wise than others, it relies on coherent detection, which is costly on a large scale. Neural networks are often proposed as a promising alternative [16]. Figure 1.5 gives an example of a neural network undoing long-haul link distortions.

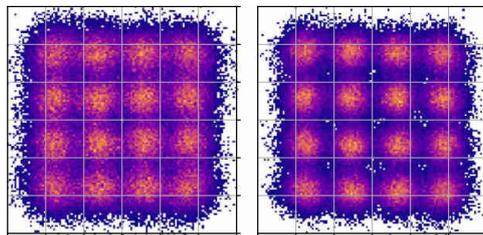


Figure 1.5: Nonlinear distortion compensation after 2800 km of fiber using a neural network. Here optical signal is modulated into 16 combinations of phase and amplitude that lay on a square grid. Figures shows samples of optical signal after recovery without (left) and with (right) a NN with 12 neurons. In the perfect case samples end up perfectly in corresponding dots. Alongside the signal samples, additional information about the link was passed as inputs. Adapted from [17].

A field-programmable gate array (FPGA) is the best option, when high-speed electronic computing is required. There, arrays of programmable logic blocks are wired together by configurable interconnects. This device offers a high parallelization capability with low latency and a typical power use of 10-20 W. Nevertheless, it is still not fast enough for online treatment of the signal with an ANN.

A research team in Bell Labs used a DNN to recover distortion of an on-off keying-encoded optical signal at 50 Gbps after 30 km of standard single-mode fiber (SSMF) – a scenario typical for Passive Optical Networks. Such DNN had 11 inputs and 47 neurons, enough to outperform a 6-tap MLSE. The capacity of an FPGA allowed four copies of such DNN in parallel. The study concluded that during offline computing, an FPGA could only handle a data stream at a 2.6 Gbps bitrate, and an FPGA needs to be 20 times faster for online computing [18].

Another point is that the task is solved in another domain. The optical signal has to be detected, discretized and then finally can be processed. Suppose the output also has to be optical (for example, for signal regeneration). In that case a discrete electronic signal must be converted back to analog and then to the optical domain. These extra steps induce latency [19] and energy costs.

#### **1.4.2 . Radio-frequency fingerprinting**

Due to fabrication tolerances, every radio transmitter is not exactly the same. Minor variations of its parameters induce a slight variation in the transmitted signal. Since parameters are random, such distortion is unique for every transmitter, which can be used for identification. Solving this task is particularly interesting for edge computing and the Internet of Things, where the device identification can be performed on a physical layer, avoiding the overhead of sending extra data to a server for it to determine the sender [20]. This task can also be applied to enhance wireless transmission security since such parameter variation is hard to spoof [21]. Like optical channel equalization, this task must be solved quickly and efficiently as information can arrive from numerous devices at a gigahertz frequency. However, the computational complexity of an appropriate convolutional neural network (CNN) renders its use impractical [22].

There are other tasks, such as the recognition of modulation formats (and other parameters) for elastic optical networks that adapt to suit demand and optical channel conditions [23] and header recognition for all-optical routing [24].

## 1.5 . All-optical analog neuromorphic computing

Computing tasks outlined in the previous section share the same trait: either the information is carried by light or will soon be carried by light, which is unsurprising, as light is being used to transfer data more and more commonly.

Optical signals have a considerably higher carrier frequency of around 200 THz, which not only supports modulation with faster signals but also usage of wavelength division multiplexing (WDM), where several optical carriers (i.e. monochromatic optical signals) with different wavelengths can be modulated independently, increasing the amount of information that can be transferred. Such multiplexing is possible due to a lack of interference between different wavelengths. Another very advantageous feature is the immunity to electromagnetic interference: it is common advice to avoid routing Ethernet cables next to power cables, but for optical fiber, it is not an issue. As a result, the information transfer across more than tens of meters is exclusively handled with light. Current research aims towards optical interconnects between chips across a few millimeters with a bandwidth of several terabits per second (see Figure 1.6).

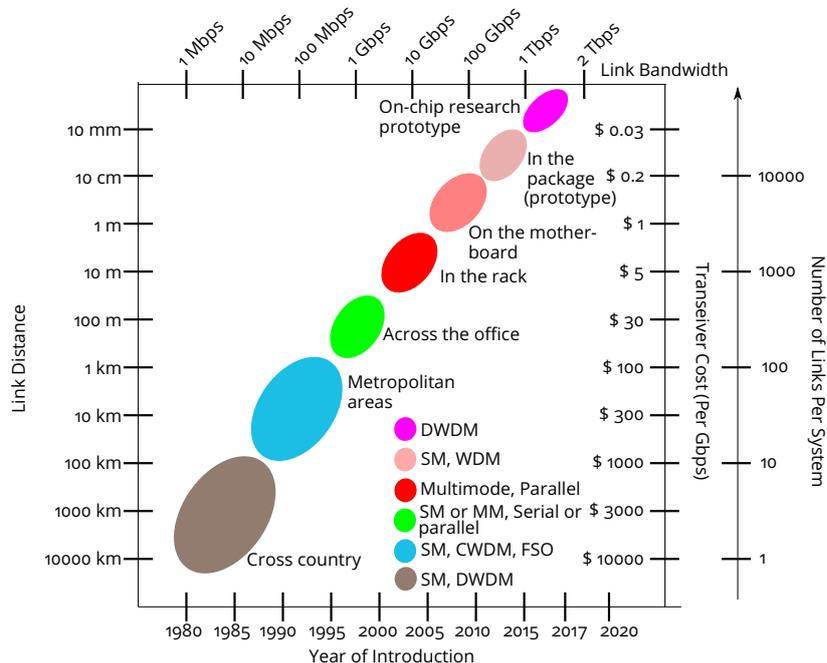


Figure 1.6: Evolution of optical communications. Adapted from [25].

Then why don't we compute directly in the optical domain? This idea is not novel: decades ago, a free-space optical network was used for face recognition [26], and a significant research effort is currently aimed towards all-

optical switching on integrated photonic chips to allow changing trajectories of signals without the involvement of electronic devices [27] which can be considered as a kind of computing, or at least a building block. Finally, the recent growth of AI has led to the rise of all-optical neuromorphic computing [28].

Several works have demonstrated that optical NNs have the potential for optical channel equalization [29, 30] and header recognition [24, 31] entirely in the optical domain. They can be used in conjunction with electronic devices as well. A good example is RF fingerprinting assisted by a photonic recurrent neural network (RNN) shown in Figure 1.7. By preprocessing optical inputs in the optical domain, helpful features can be accentuated and unnecessary data discarded, making further processing easier. In this example, a photonic RNN allowed a 50 times smaller electronic CNN [22].

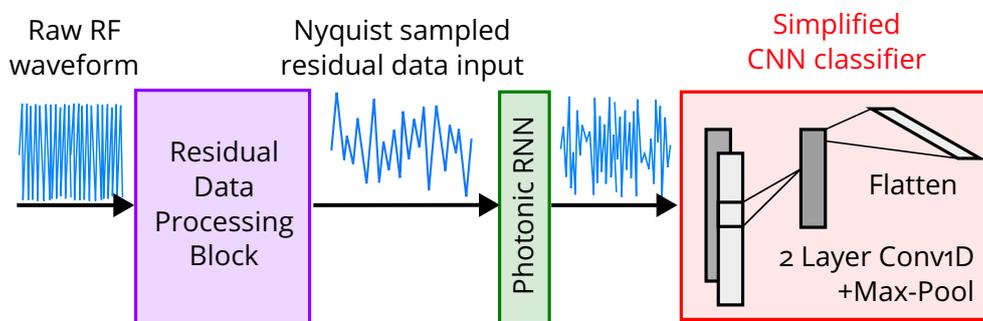


Figure 1.7: Radio-frequency fingerprinting facilitated by a photonic RNN. Here, residual data is a difference between the received raw data and the ground truth data, the signal regenerated after decoding the transmission. Redrawn based on [22].

How is that possible? From a simplified perspective, a neuron accumulates signals from other neurons and applies a nonlinear function, the result of which is then picked up by other neurons. Summation is a linear operation and is relatively easy to implement. Besides, there is no need for high accuracy [32], and a highly accurate digital summation can be replaced with an analog counterpart, which is important since physical systems are smooth and continuous as long as the quantum limit can be ignored. We should not impose 'unnatural' restrictions when possible to get the most out of a new computing platform.

Nonlinearity is more challenging as the response of most materials used for optics is linear under practical conditions. Nevertheless, a few approaches to harnessing the nonlinear response have been proposed, either optoelectronic or all-optical; see [28] for an overview. One way is by exploiting photonic integrated circuits.

Integration reduces the size of components, allowing for a strong confinement of photons in a small volume, leading to a high intensity of light. In that

case, nonlinearity becomes considerable even when the optical power is in the milliwatt range, as has been demonstrated numerous times [27]. Integrated photonic devices also consume less power than their free-space counterparts and can be mass-produced, reducing costs, as was hinted early by S. Miller [33]. Their mechanical stability is also convenient, allowing for easier handling of coherent signals where phase is important. Finally, compatibility with complementary metal-oxide-semiconductor CMOS technology allows for co-integration with electronic components [34], allowing for easier control, as neuromorphic devices require many parameters, and dedicating an electrical port to each quickly becomes problematic with larger networks [28].

Another significant complication of small device size is its scalability potential, which directly impacts the capabilities of a device. Modern CPUs have reached current performance levels while drawing a reasonable amount of power mainly because of a smaller and smaller feature size, which allows packing trillions of transistors in one CPU. These are very big shoes to fill! Such density is impossible to achieve in photonics as optical wavelengths are much larger than the length of de Broglie waves of free electrons in a semiconductor. However, we pursue different goals. The goal of CPUs is general computing (with some application-specific accelerator submodules), whereas our ends are much more specific, which allows us to keep system complexity at manageable levels. Nevertheless, some tasks can be difficult enough to warrant more compute to tackle successfully: recovery of signals from longer fibers, recognizing more and more complicated modulation patterns and longer headers.

Thus, the motivation for this thesis formed. We aim for

**Large Scale** to be able to increase the number of ‘photonic neurons’ in the network, boosting its computing capabilities and solving more challenging problems,

**Autonomous** to avoid direct involvement of electronic computers and reap the benefits of photonics in full,

**All Analogue** to employ physical systems for computing as they are – analogue,

**Photonic** to take advantage of the broad bandwidth of photonics and seamlessly integrate into existing communication systems,

**Neural Networks** to solve tasks that are difficult to undertake with algorithms.

## **1.6 . Objectives**

This work pursues two goals.

First, we propose and evaluate numerically and analytically a novel approach towards scalable all-optical neuromorphic computing by implementing a reservoir computer using the evanescent coupling of nonlinear resonators. In contrast to implementations found in literature, this approach combines a much higher neuron density and a nonlinear operation, allowing for a complex treatment of optical signals without the direct involvement of electronics. We also provide an insight into the details of such a system, showing its limitations and potential improvements. We then fabricated the proposed system, a characterization of which is planned for future work.

The second goal is to show that a recently demonstrated spiking neuron based on a photonic crystal cavity can be used to create a spiking neural network that can sample from a BM at nanosecond timescale, performing stochastic inference.

## **1.7 . Thesis outline**

Chapter 2 discusses ANNs in more detail and gives an overview of their hardware implementations in the literature. We introduce the Reservoir Computing (RC) paradigm and benefits for integrated photonic neuromorphic computing.

Chapter 3 proposes a new scalable approach to integrated optical RC and discusses its behaviour.

Chapter 4 demonstrates how the proposed RC handles two tasks: prediction of physical models with an example of the Mackey-Glass equation and nonlinear channel equalization.

Chapter 5 is dedicated to spiking neural networks, where the recent result on a spiking neuron based on a photonic crystal cavity is used to perform stochastic computation.



## 2 - Integrated neuromorphic computing

The previous chapter focused on the 'why', and explained the general interest in ANNs and their all-optical implementation. This chapter, however, focuses on the 'how'. It briefly explains how ANNs are structured, how they solve tasks, and how to make them do this. We also discuss different kinds of ANNs and how they could be implemented in the optical domain.

### 2.1 . Artificial neural networks

Humans and animals compute using a brain – a massive network of interconnected neurons, basic building blocks, that communicate using electrical spikes. Each neuron is composed of dendrites, through which a neuron receives inputs, a soma accumulates them, and, if conditions are right, sends a spike that propagates to other neurons via an axon. While the structure

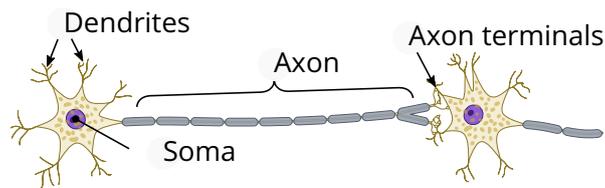


Figure 2.1: Biological neuron. Adapted from [35].

of such a network is incredibly complex (there are many other components, such as astrocytes) to faithfully replicate, perhaps there is no need for a precise replica.

In 1943, W. McCulloch and W. Pitts introduced a perceptron that was a much-simplified biological neuron model [36]. Consider a set of weights  $w$  and a bias  $b$  as perceptron parameters and a vector of inputs  $x$ . A perceptron performs a multiply-and-accumulate (MAC) operation  $w \cdot x + b$ , and if the result is non-negative, it outputs one and, otherwise, zero; in other words, it applies the Heaviside step function

$$H(y) = \begin{cases} 0, & y < 0 \\ 1, & y \geq 0 \end{cases} . \quad (2.1)$$

Such perceptron was used for classification tasks – weights and bias are optimized or 'trained' such that the output is 1 for particular inputs. However, a perceptron can only classify linearly separable inputs, so a large portion of tasks would not be solvable, for example, an XOR operation [37].

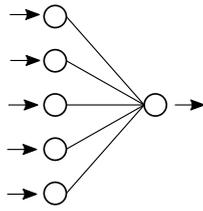


Figure 2.2: Perceptron. Redrawn based on Figure 2.3.

This problem has been solved with the introduction of a multilayer perceptron (MLP) by F. Rosenblatt in 1958 [38]. There, the output of a perceptron can be an input to other perceptrons. Specifically, an MLP had three layers in that work: an input, hidden and output layer. Only connections to the output layer were trained, while others were random (see Figure 2.3). Later, such a network would be called an extreme learning machine. Analog output was another addition. A ‘classic’ perceptron applied a Heaviside function on  $y = \mathbf{w}\mathbf{x} + b$ , but now smooth functions such as the sigmoid were used

$$\sigma(y) = \frac{1}{1 + \exp(-y)}. \quad (2.2)$$

Because of such properties, an MLP could perform complex nonlinear transformations of inputs, allowing for better classification performance. An MLP is one example of many feed-forward neural networks (FNNs).

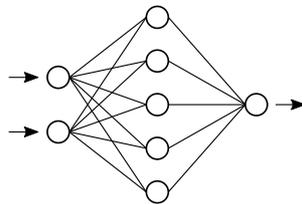


Figure 2.3: Multilayer perceptron. Adapted from [39].

More layers can be added to improve performance further, and all connections can be trained, which results in a deep neural network (DNN), a specimen of the deep learning architecture.

The tasks mentioned in Section 1.4 require the processing of time-dependent inputs. While this is still possible to carry out with a stateless ANN such as DNN by supplying delayed copies of a signal as inputs, a recurrent neural network (RNN) is better suited for such purposes. In contrast to FNNs with neuron connections oriented only to a subsequent layer, an RNN can have looped-back neuron connections. They make an RNN a dynamical system, allowing a network to preserve some input history, leading to a much more powerful computing capability. Theoretically speaking, RNNs with an infinite precision are proven to be Turing-complete [40], i.e. are capable of solving any

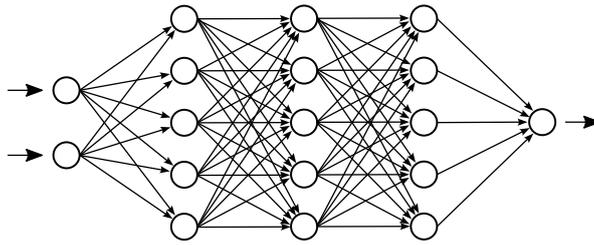


Figure 2.4: Deep neural network. Redrawn based on Figure 2.3.

computational task. The downside is a very challenging training procedure (on that later).

## 2.2 . Reservoir Computing

In order to harness the capabilities of RNNs without excessive training complications, H. Jaeger introduced the Echo-State Network (ESN) [41], shown in Figure 2.5. An ESN is comprised of three parts:

- input units that are equivalent to those of a typical ANN,
- internal units, typical neurons interconnected in an arbitrary fashion that receive signals from input units,
- output units (also called a ‘readout’) are linear neurons unidirectionally connected to internal units.

Only these unidirectional connections to output units are optimized, and this immensely simplifies the training of ESNs compared to typical RNNs. In some cases, the output may be fed back into ESN as an input, but this is rarely needed and is not recommended. It has been shown that despite such severe simplifications, ESNs found success in a wide array of tasks such as recognition of spoken Japanese vowels [42], wireless communication channel equalization [43], financial forecasting [44] and seizure detection [45].

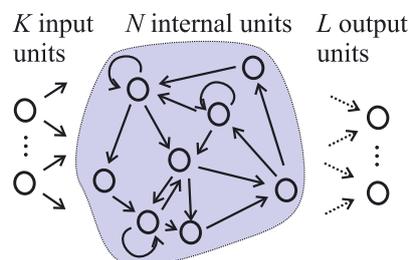


Figure 2.5: Echo State Network [46].

The main ESN equation is often written as [47]

$$\mathbf{x}(n+1) = (1 - \alpha)\mathbf{x}(n) + \alpha \mathbf{f}(\hat{W}\mathbf{x}_n + \hat{W}_{\text{in}}\mathbf{u}_{n+1}), \quad (2.3)$$

where  $\mathbf{x}(n)$  is a vector of internal units state at time  $n$ ,  $\alpha$  is leak rate,  $f(\mathbf{x})$  is an element-wise nonlinear function, commonly called the activation function,  $\hat{W}$  and  $\hat{W}_{\text{in}}$  are connectivity matrices between internal units' and between input and internal units, respectively, and  $u(n)$  is a vector-valued input signal. The output is given by

$$\mathbf{y}(n) = \hat{W}^{\text{out}}[1; \mathbf{x}(n)]^{\top}. \quad (2.4)$$

Later, ESN and other similar approaches were put under a single term – reservoir computing (RC)<sup>1</sup> [48]. Besides ESNs, RC includes the following architectures.

**Liquid state machine** (LSM) was proposed by W. Maass at the same time as ESNs and was aimed towards networks of spiking neurons with more biologically plausible parameter assignment [49]. An LSM consists of a dynamical network of spiking neurons (or 'liquid') and a linear readout. It was proven that in order to compute with an LSM, it has to have three properties:

- fading memory, i.e. the most significant bits of the current ( $t = 0$ ) output value depends just on the most significant bits of the values of input function in some finite time interval  $[-T; 0]$  [50],
- separation property, i.e. different inputs result in different trajectories of the network's dynamical variables,
- approximation property, i.e. readout can map the liquid state into a target function with a negligible or acceptably small error.

While successfully used in speech recognition [51] and other tasks, LSMs got little traction due to the complicated treatment of spiking signals. As a result, most of the RCs are, in fact, ESNs.

**Evolino** is an ESN with long short-term memory (LSTM) units composed of an input, an output and a 'forget' gates (see Figure 2.6). This unit remembers its state for an arbitrary amount of time until the 'forget' gate erases it. This way, a longer memory is attained. Training is performed with evolutionary methods [52].

---

<sup>1</sup>RC' may also mean 'reservoir computer' depending on the context

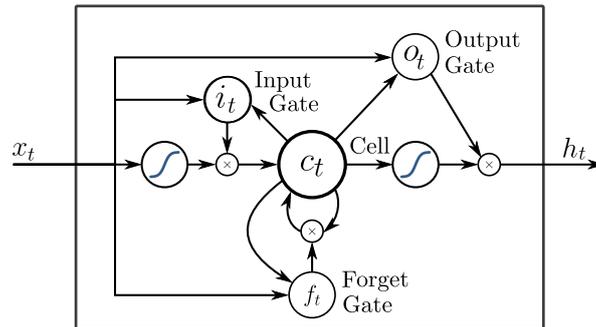


Figure 2.6: Long short-term memory unit. Adapted from [53].

**Backpropagation-decorrelation** is an online training method for an ESN with output feedback. This algorithm is claimed to be insensitive to the choice of internal weights. It also converges quickly but is heavily biased to more recent data [54].

The introduced approaches share the same idea: transform an input signal into a high-dimensional space in which we perform a linear separation. Such generality implies that an RC does not have to be a neural network at all – it is enough to be able to read the state of an arbitrary dynamical system, which could range from quantum oscillators [55] and magnetic skyrmions [56] to a water tank, quite literally, *reservoir* computing [57].

## 2.3 . Training neural networks

Until this point, the training of neural networks was mentioned only briefly, described with vague terms like ‘easy’ and ‘hard’. This section discusses the training methods of various neural networks and the importance of the RC concept.

### 2.3.1 . Overview

Training algorithms can be classified into three groups:

- Unsupervised learning, where an ANN is supplied with input data, but each input data point is not assigned to a particular output. The network then tries to find patterns in the given data.
- Reinforcement learning, where the training feedback informs an ANN how good the *sequence* of outputs is. Then, an ANN learns by itself, only knowing how good it is doing compared to before, which is useful when the ‘goodness’ of the result is easy to estimate but not the way towards it. Games such as chess and Go are good examples – it is easy to tell who won, but if we knew the best moves for all possible situations, there would be no need for ANNs in the first place.

- Supervised learning, where the target output is given alongside input. It typically is the fastest and results in the most accurate solution, but knowing the target is not always possible. A semi-supervised approach can be used if only a part of the inputs has a corresponding target.

This work targets supervised methods since targets are known for tasks we aim to solve with photonic ANNs. To demonstrate how such methods work, we come back to perceptrons.

Assume a dataset  $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$ , where  $\mathbf{x}_i \in \mathbb{R}^K$  is an input and  $y_i \in \{0, 1\}$  is its corresponding class. The goal of a perceptron is to separate classes, meaning that

$$\forall i = 1 \dots N \quad H(\mathbf{w}\mathbf{x}_i + b) = y_i, \quad (2.5)$$

where  $\mathbf{w}$  and  $b$  are the perceptron's weights and bias. For that, one needs to find, i.e. optimize  $\mathbf{w}$  and  $b$ . For simplicity, we append unity to the state  $\tilde{\mathbf{x}}_i = [1; \mathbf{x}_i]$  and the bias to the weights  $\tilde{\mathbf{w}} = [b; \mathbf{w}]$ , then  $\mathbf{w}\mathbf{x}_i + b = \tilde{\mathbf{w}}\tilde{\mathbf{x}}_i$ . Then, starting from an arbitrary  $\tilde{\mathbf{w}}_0$ , the following update rule is used

$$\tilde{\mathbf{w}}_{k+1} = \tilde{\mathbf{w}}_k + \delta\tilde{\mathbf{w}}_k, \quad (2.6)$$

where

$$\delta\tilde{\mathbf{w}}_k = r \sum_{i=1}^N [y_i - \sigma(\tilde{\mathbf{w}}_k\tilde{\mathbf{x}}_i)]\tilde{\mathbf{x}}_i \quad (2.7)$$

and  $r$  is the learning rate. This rule can be derived in a few ways. For example, a quick and not very rigorous would be to use gradient descent with a loss function

$$L = \sum_{i=1}^N L_i = \sum_{i=1}^N -[y_i - \sigma(\tilde{\mathbf{w}}\tilde{\mathbf{x}}_i)]\tilde{\mathbf{w}}\tilde{\mathbf{x}}_i. \quad (2.8)$$

Then, whenever a perceptron correctly classifies class  $i$ ,  $L_i = 0$ , we do nothing. But if  $\tilde{\mathbf{w}}\tilde{\mathbf{x}}_i < 0$  and  $y_i = 1$  then  $L_i > 0$ . Same if  $\tilde{\mathbf{w}}\tilde{\mathbf{x}}_i > 0$  and  $y_i = 0$ . Therefore, the loss increases whenever the perceptron makes a mistake, and the loss is higher when  $\tilde{\mathbf{w}}\tilde{\mathbf{x}}_i$  is further from a correct class. Then, according to the gradient descent algorithm

$$\Delta\tilde{\mathbf{w}} = -r\nabla_{\tilde{\mathbf{w}}}L = r \sum_{i=1}^N [y_i - \sigma(\tilde{\mathbf{w}}\tilde{\mathbf{x}}_i)]\tilde{\mathbf{x}}_i, \quad (2.9)$$

where a term with a derivative of  $\sigma$  was omitted since it is nil everywhere except zero, but the probability of getting a data point there would be infinitely small. Graphically, this algorithm is shown in Figure 2.7. This algorithm is guaranteed to converge if the solution exists, i.e. data points are linearly separable [58].

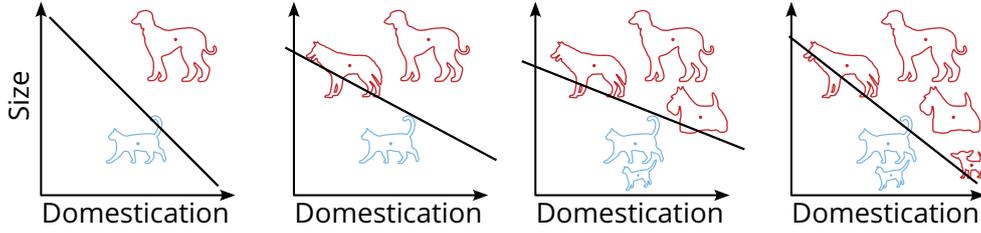


Figure 2.7: Training perceptron to classify animals based on size and domestication. As more data points are added, the decision boundary encoded by perceptron weights and bias is updated. Adapted from [59].

This algorithm would also work with minor corrections for a ‘classic’ MLP where only output weights are trained. Here, at the first hidden layer, an input vector  $\tilde{x}_i$  is multiplied by a random matrix, and a sigmoid is applied element-wise. The result is processed similarly at the second hidden layer but with a different matrix until the output layer is reached. This chain of operations could be described by a vector-valued nonlinear function  $F$ . Then, we could consider this MLP as a perceptron with inputs preprocessed by  $F$ . Then, in the equations above, the input  $\tilde{x}_i$  is replaced by  $F(\tilde{x}_i)$ , and the gradient of the loss function  $\nabla_{\tilde{w}} L$  would also depend on the sigmoid derivative.

Training complicates if all layers are to be optimized. Then, the derivative of the loss function can be computed using the chain rule, and gradient descent can be used. The backpropagation (BP) algorithm does so efficiently for all network parameters. Due to the popularity of this method, numerous articles with its derivation can be found (for example, see [60, 61]); therefore, it is omitted here. The main idea is to start with some set of parameters, do a forward step to compute neuron activations and loss, and then compute updates of parameters by backpropagating the error vector starting from the last layer to the first – the backward step [62]:

$$\delta \hat{w}_i = -[(\hat{w}_{i+1}^\top \delta \mathbf{a}_{i+1}) \odot \sigma'(\mathbf{a}_i)] \sigma(\mathbf{a}_{i-1}^\top), \quad (2.10)$$

where  $\sigma(\mathbf{x}) = [\sigma(x_1), \sigma(x_2), \dots, \sigma(x_n)]^\top$ ,  $\mathbf{a}_i = \hat{w}_i \sigma(\mathbf{a}_{i-1})$ ,  $\delta \mathbf{a}_i = \partial L / \partial \mathbf{a}_i$ ,  $\sigma' = d\sigma/dx$  and  $\odot$  is the Hadamard product. For biases, the equation is similar.

While very successful for digital ANNs, this algorithm suffers from several issues

1. biological implausibility due to the weight transport problem, as during the backward step, synaptic weights, which are supposed to be ‘local’, have to be transmitted to other parts of the network; also, synaptic symmetry is enforced [63],
2. difficult on-chip realization as BP requires a lot of information on the ANN state, including a derivative of the activation function.

An algorithm for on-chip BP of an integrated photonic accelerator using a network of MZIs using the adjoint variable method has been proposed [64] and implemented [65]. The nonlinearity, however, was implemented on a digital computer. Another recent work implements backpropagation of weights in an optical NN with a saturable absorber as nonlinearity, and the role of electronics was reduced to the preparation of inputs, detection and assignment of weights [66]. However, to the best of my knowledge, there have been no autonomous implementations of BP on an integrated chip.

Alternative training methods have been proposed to resolve the weight transport problem. One of them is feedback alignment (FA), where a per-layer random matrix is used instead of a transposed ANN weight matrix [67] (see Figure 2.8). Here, ‘alignment’ means that ANN weights align with these random feedback matrices at the beginning of training and, later, they change the direction to minimize the error. Direct feedback alignment (DFA) is an even simpler FA where, instead of backpropagating the error vector layer by layer, it is propagated to all layers simultaneously, allowing for better parallelization [68]. They have been shown to work with some limitations, e.g., they do not apply to CNNs [69]. It has also been recently shown that DFA struggles to reach the same performance as BP [70]. Besides FA and DFA, other algorithms are proposed. One is equilibrium propagation, which is a physics-inspired method that optimizes the network by minimizing its free energy as defined through the loss function [71].

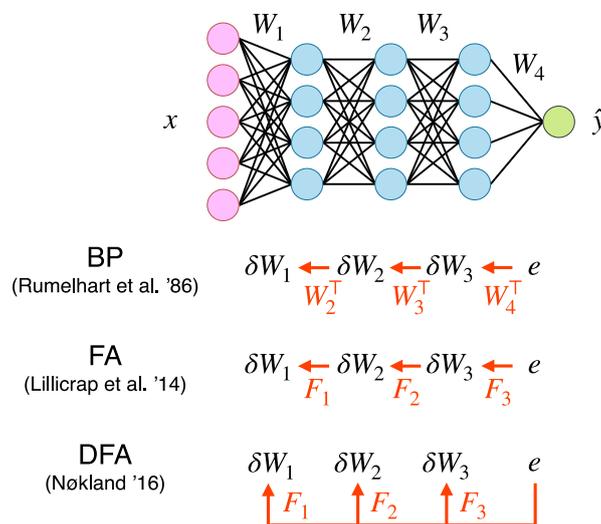


Figure 2.8: Comparison of Feedback Alignment (FA) and Direct Feedback Alignment (DFA) to backpropagation (BP). Adapted from [72].

Training RNNs is considerably more difficult than FNNs. Backpropagation through time (BPTT) uses the fact that an RNN can be unfolded in a series of FNNs (see Figure 2.9). Currently, BPTT appears to be the best method of train-

ing RNNs. Other options are generic nonlinear optimization algorithms such as BFGS (Broyden, Fletcher, Goldfarb and Shanno) and Levenberg-Marquardt or evolutionary algorithms. Just like FNNs can be trapped in a local minima, so can RNNs, but more easily [73].

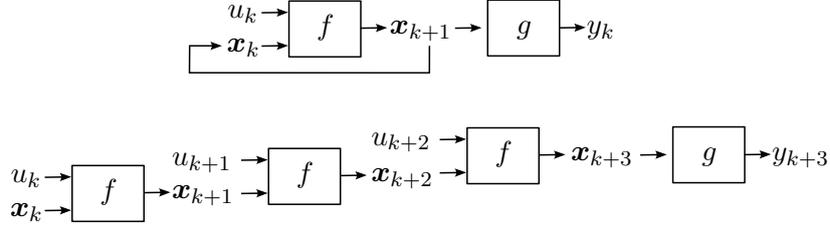


Figure 2.9: Unfolding a recurrent neural network (top) into a series of feed-forward neural networks (bottom). Here  $x_k$ ,  $u_k$  and  $y_k$  are the network's state, input and output at time  $k$ ,  $f$  and  $g$  are activation functions in the recurrent and feed-forward layers.

Training methods of RCs can be classified into two groups:

- Offline methods, where information on RC dynamics is recorded, and the training is carried out in one go. This might not be an option in some cases, for example, if there is output feedback. In our cases, such training will be applicable and therefore used. Ridge regression is an example of such training and is explained in the next paragraph.
- Online methods, where the reservoir improves incrementally as more data arrives. One example is the First-Order Reduced and Corrected Error (FORCE) learning rule [74].

### 2.3.2 . Offline reservoir training: ridge regression

Offline training of RCs is trivial compared to RNNs. In a typical RC, only the output layer is trained, and there is no output feedback. Therefore, a change in training parameters does not change the behaviour of the internal units. The training is then carried out by ridge regression as follows:

- choose a training input  $\mathbf{u}(n)$  and target output  $\mathbf{y}^{\text{tgt}}(n)$
- run the RC using  $\mathbf{u}(n)$  as input, recording RC states  $\mathbf{x}(n)$
- stack RC states vertically to make a design matrix  $\hat{X}$ , similarly make  $\hat{Y}^{\text{tgt}}$  out of  $\mathbf{y}^{\text{tgt}}(n)$
- since the RC output is then  $\hat{Y} = \hat{W}^{\text{out}} \hat{X}$ , the goal is  $\hat{Y}^{\text{tgt}} \approx \hat{W}^{\text{out}} \hat{X}$ , which is a linear problem that can be solved with ridge regression [47]

$$\hat{W}^{\text{out}} = \hat{Y}^{\text{tgt}} \hat{X}^{\text{T}} (\hat{X} \hat{X}^{\text{T}} + r \hat{I})^{-1}, \quad (2.11)$$

where  $r$  is a regularization parameter and  $\hat{I}$  is an identity matrix.

This way an RC can reach the best possible performance in one step and without the need to know the derivatives of activation functions.

### 2.3.3 . Regularization

When an ANN is trained on a given dataset, it might learn to replicate minute details of a dataset instead of looking for general patterns – overfitting shown in Figure 2.10, which can be an outcome of an excessive number of parameters in a model or an incorrectly prepared training dataset.

For example, assume an ANN is trained to compute a product of two numbers. However, we were careless in preparing the training dataset, and there is only a number ‘2’ on the second input neuron. A training procedure is carried out, and we might see an accuracy improvement. However, it is likely that the ANN only doubles the number on the first input neuron. It would be quickly discovered if the value on the second input neuron is changed. This is how we check if an ANN is overfitting: using a separate test dataset alongside the training dataset. This dataset is not involved in optimizing of network parameters, and the performance of an ANN on both of them is checked during training. One can find that the error reduces on both the training and test datasets, but testing error starts increasing at some point – a sign of overfitting (see inset in Figure 2.10).

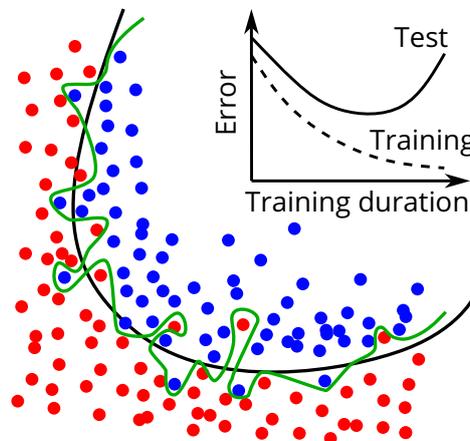


Figure 2.10: Overfitting example. A model is trained to separate two classes of points. An overfitted model (green line) also separates irregularities that arise, e.g., due to a noisy measurement. A properly generalizing model is shown as the black line. Inset shows a typical error landscape of an overfitting model: at some point, more training reduces error on the training dataset but increases on the testing one due to a loss of generalization capability. Adapted from [75].

Overfitting also happens for RCs, so the regularization coefficient is used. It punishes large output weights, which are indicate unstable training result,

i.e. the output will drastically change when the input changes slightly, which is another symptom of overfitting. Another option is to add noise to the training dataset, though it might complicate reproducibility [47].

## 2.4 . Reservoir performance tips and tricks

The simplicity of RC training comes from the assumption that the internal states of an RC are ‘rich’ enough by themselves without the need to tune them, such that a simple linear readout would suffice. A good question is how to make an RC a good one. The answer is discouraging, best expressed by a quote from the author of ESNs, H. Jaeger [76, p.vi]:

*... the closer one becomes involved with RC, the more difficult it gets, and if the one to embrace it in full contact, it gets almost impossibly difficult. ... A full theory of reservoir dynamics would be a full theory of everything that evolves in time. ... After two decades of RC research, we only have the faintest inklings of how to match reservoir dynamics with task dynamics.*

All in all, there are a few good ideas that usually work but still do not guarantee a good RC.

Since the internal state of an RC is a complex nonlinear dynamical system, it is recommended to look at RC’s global parameters, metaparameters and properties instead:

- Echo state property (ESP), i.e. if the current state of an RC is defined only by the fading history of inputs [41]. That also means an RC forgets about inputs further in time than a certain threshold.
- The spectral radius of the internal connection matrix  $\hat{W}$ , i.e. the maximum absolute value of its eigenvalues. It is noted that an RC typically has the ESP if the spectral radius of  $\hat{W}$  is less than unity. While this is true in many cases [47], it is neither a necessary nor a sufficient condition for good performance, and some tasks are better solved with a spectral radius of much more than unity [77].
- Sparsity of the internal connection matrix  $\hat{W}$ . Sparse connections sometimes lead to better computing performance, but not much [47]. Rather, it allows linear scaling of non-zero elements in  $\hat{W}$  and, therefore, faster matrix-vector multiplication (MVM) for large RCs.
- Leaking rate, i.e. how quickly neurons attenuate, controlled by  $\alpha$  in Eq. (2.3).
- Memory capacity, i.e. RC’s ability to remember previous inputs, a commonly used computational capacity metric. It is measured by providing an RC with random input  $u_n$  and then training the RC to output

$y_n \approx u_{n-\tau}$  for each delay  $\tau > 0$ . The MC is then found by [46]

$$\text{MC} = \sum_{\tau>0} \text{cor}(u_{n-\tau}, y_n) \quad (2.12)$$

where  $\text{cor}(x, y)$  is the Pearson correlation coefficient [78], or [79]

$$\text{MC} = \sum_{\tau>0} [1 - \text{NRMSE}(u_{n-\tau}, y_n)], \quad (2.13)$$

where [47]

$$\text{NRMSE}(x, y) = \frac{1}{\sigma_n[y_n]} \sqrt{\frac{1}{T} \sum_{n=1}^T |x_n - y_n|^2}. \quad (2.14)$$

is the Normalized Root Mean Square Error (NRMSE) and  $\sigma_n(x_n)$  is a standard deviation of  $x_n$ . Nonlinearity degrades the memory capacity [80]. In that case, an RC trades this linear MC for a nonlinear one, and the total capacity stays almost identical [79].

- Input scaling, i.e. how strongly inputs impact RC dynamics.
- Dimensionality, i.e. the number of neurons that operate differently [81].
- Consistency, a measure of reproducibility of RC's behaviour.

Dimensionality and consistency will be extensively used in this work and are explained in more detail in the following sections.

### 2.4.1 . Dimensionality

Since an RC expands the input to a high-dimensional space, it would be beneficial to expand to a more high-dimensional space and increase the chances of a successful linear separation (see Figure 2.11). It has been noted that a higher dimensionality often leads to better computing performance [81].

To measure dimensionality, an input signal is passed to an RC, and activations of its neurons are recorded in a matrix  $\hat{X}$ . This matrix carries information about trajectories each neuron is making in the RC's phase space, i.e. a space in which all possible states of a dynamical system are represented (see Figure 2.12 for an example). Then, the PCA of  $\hat{X}$  is performed, producing orthogonal principal components that show the main directions along which data varies the most and give amplitudes of the corresponding variations. If some columns of  $\hat{X}$  are linearly dependent, i.e. some neurons behave similarly, they will be mapped to principal components with a smaller amplitude. After that, one needs to decide how much variance is significant. In our case, we discard principal components whose length is less than 1% of the maximum length of all components – the default behaviour of the library used, *MultivariateStats.jl*<sup>2</sup>.

<sup>2</sup><https://github.com/JuliaStats/MultivariateStats.jl>

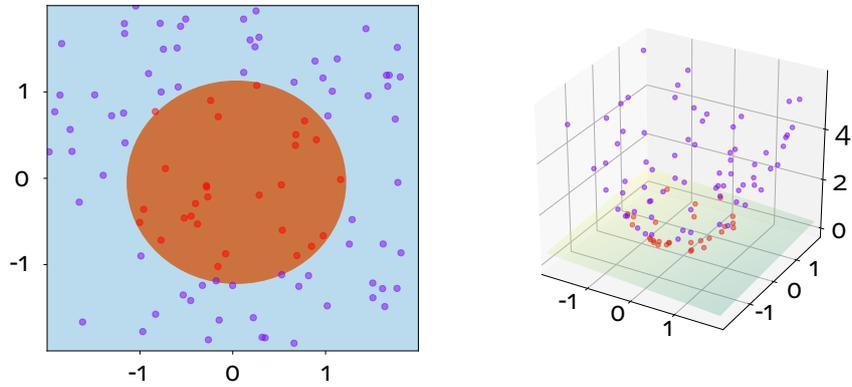


Figure 2.11: Some tasks cannot be solved with a linear regression. Left figure shows two classes of points on a plane. They can be separated with a circle, which is a nonlinear classifier. Instead, inputs can be mapped to a higher-dimensional space where a linear separation might be possible. Adapted from [82].

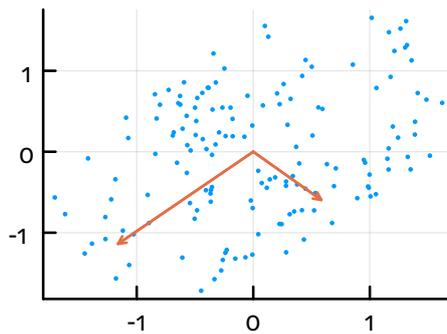


Figure 2.12: Example of the Principal Component Analysis. A dataset is shown as dots, and arrows show two principal components with the arrow length proportional to the variance of data along its direction.

Suppose an RC is noisy, or the dimensionality computation is performed on an experimental setup. In that case, noise can significantly increase the amplitude of otherwise weak principal components, even though variance along them does not contribute to computing capacity. In that case, assuming the noise is Gaussian, such components can be systematically removed [81].

The kind of input provided for the dimensionality analysis depends on the objectives. A reasonable choice is to use white noise, which would test all possible excitations an RC could encounter, and the total expansion capability of an RC would be tested. Another option is to provide an input of a task to be solved, and then only the expansion for this particular task is tested. In this work, a random excitation is used when an RC is analysed from the general point of view, whereas for task benchmarking the task's input signal is used.

### 2.4.2 . Consistency

A claim that an RC operates better at the edge of chaos, i.e. close to being chaotic, is commonly found in literature. To detect the chaoticity of an RC, the Lyapunov exponent can be used [83, 84].

Assume two identical dynamical systems starting from very close initial conditions with a distance of  $|\delta(0)|$  (see Figure 2.13). Tracking the evolution of these systems, one may find that the distance between current states may follow

$$|\delta(t)| \approx \exp(\lambda t) |\delta(0)|. \quad (2.15)$$

In that case,  $\lambda$  is the Lyapunov exponent. Strictly speaking, a system with  $n$  variables has  $n$  Lyapunov exponents [85], but we are mainly interested only in the largest exponent. Positive  $\lambda$  makes infinitesimally close trajectories diverge exponentially over time, which is a signature of chaos [85]. The boundary between  $\lambda = 0$  and  $\lambda > 0$  is called the edge of chaos [86].

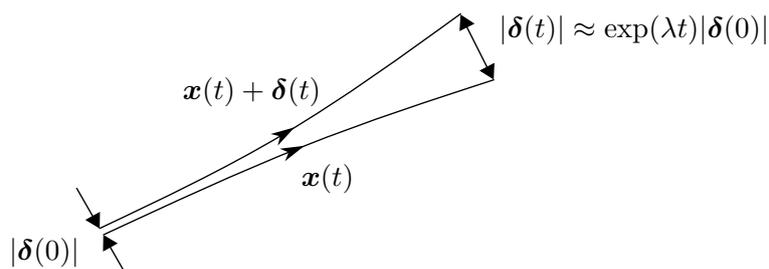


Figure 2.13: Graphical explanation of the Lyapunov exponent. Exponential divergence of evolution trajectories  $x(t)$  and  $x(t) + \delta(t)$  starting from close initial conditions  $x(0)$  and  $x(0) + \delta(0)$  is a marker of chaoticity.

However, a claim that operating at the edge of chaos is optimal is not exactly correct. First, chaos is not strictly defined for input-driven systems [76]. Second, it is valid only for a subset of tasks [87]. Rather, an RC should be at the edge of ESP instead, which does not necessarily mean proximity to chaos [76].

This has been recently addressed by Yamaguchi T. et al. in an RC made of spin-torque oscillators [86]. They introduce the ‘synchronization index’, a long-time average of the distance between two trajectories of a neuron that only differ in their initial state. Suppose an autonomous system eventually converges to a fixed point, the index is zero then. However, if the system is self-oscillating, the distance would not change, and the index would be positive. Authors argue that even though the synchronization index and the Lyapunov exponent appear functionally similar, there are differences, demonstrated in Figure 2.14.

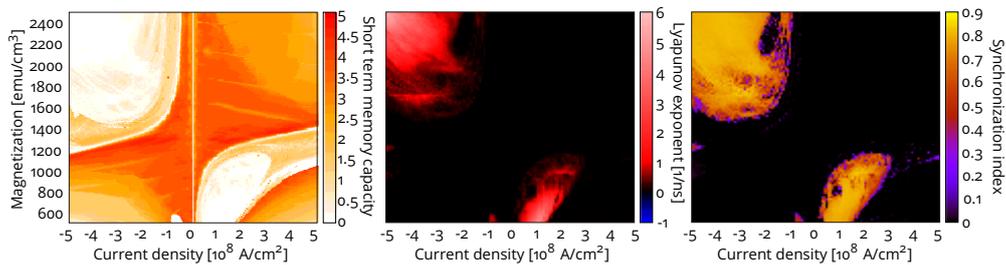


Figure 2.14: An increase of memory capacity near the synchronization index border, but not of the Lyapunov coefficient. Adapted from [86]. Figures share vertical axes.

A similar metric, consistency, had also been introduced. Consistency is a property describing the reproducibility of a dynamical system’s (here, the reservoir’s) responses under an injection of slightly different input signals [88]. It is conceptually close to chaoticity: if a second copy of a dynamical system received a slightly different input, its state would be slightly displaced from the state of the first copy. If trajectories start to diverge after that, it indicates chaotic behaviour. As far as RC’s training is concerned, it should not be chaotic. Indeed, if responses change every time, training would be impossible, and it is typically desirable that the consistency is high [89].

In this work, consistency is used to determine the chaoticity of an RC. The analysis could be done by providing an RC with slightly different input signals, recording neuron trajectories for all inputs, and computing their cross-correlation matrix [81]. If a neuron is chaotic, its responses will differ, and a corresponding correlation will reduce. A root mean square (RMS) of diagonal elements is the consistency of the entire RC.

## 2.5 . Integrated photonics for neuromorphic computing

It is not surprising that photonics is often considered for fast neural networks. The advantages outlined in Section 1.5 motivated research into neural networks in integrated photonic alongside the boom of deep learning in the early 2010s (see Figure 2.15).

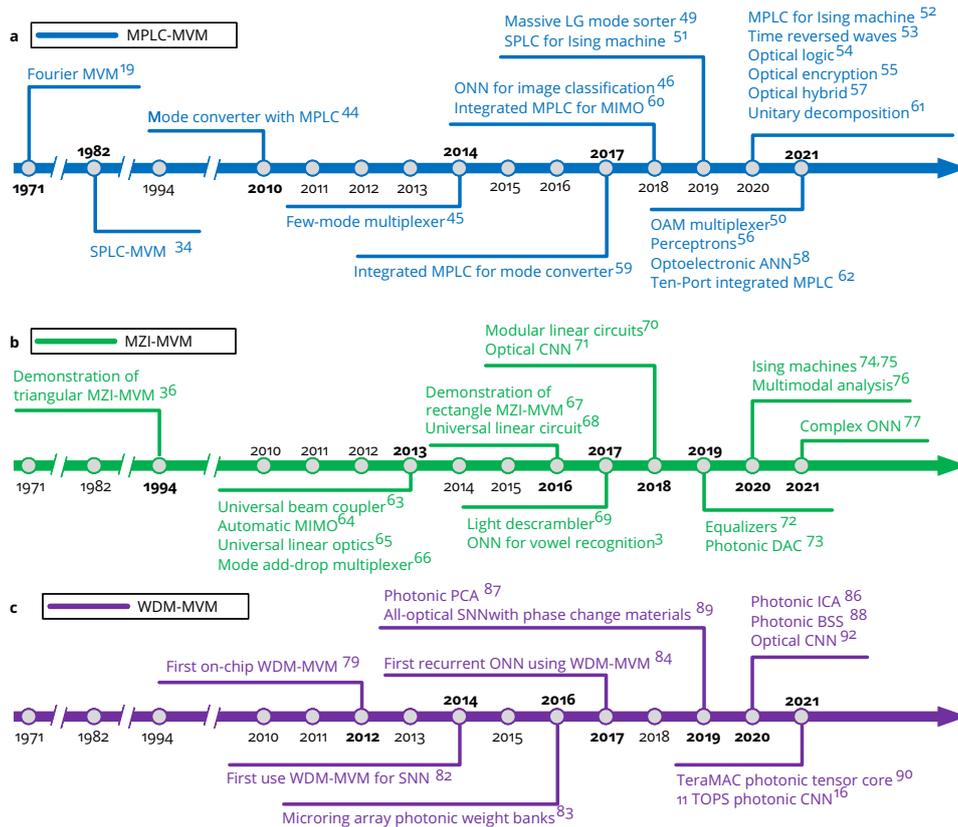


Figure 2.15: Timeline of publications concerning neural networks on the integrated photonic platform. Here MPLC is multi-plane light conversion, MZI is a Mach-Zehnder interferometer. Adapted from [90].

Efforts towards integrated photonic neuromorphic computing can be separated in two directions: accelerators and actual neural networks. Accelerators (or co-processors) aim to offload the most intensive MAC or matrix-vector product (MVP) operation to optics. The advantages of accelerators are:

- optics is typically linear, and so is the MAC operation,
- more straightforward implementation since an accelerator only serves a particular role,
- better marketability due to the similarity to how GPUs are used.

There are disadvantages as well:

- reliance on an electronic computer since an accelerator only implements neuron connectivity (possibly with an activation function), and providing the next batch of inputs is up to a computer,
- the use of many high-speed (and therefore costly) electric components for OEO conversion to handle inputs and outputs,
- they require quite high accuracy, as usually training is not done on the substrate, and hence their configuration needs to match the model system closely.

The neural network approach is the complete opposite: its implementation is more challenging, but everything is handled in hardware; it promises a faster and more efficient operation with fewer OEO conversions.

### 2.5.1 . Photonic accelerators

A commonly cited integrated optical accelerator is shown in Figure 2.16. There, a network of MZIs is used to implement the multiplication of a vector of coherent optical inputs by a unitary matrix [91]. Then, an array of MZIs is used to multiply each output by a separate complex number. Such a system is only limited by the detection speed (100 GHz) but can suffer from

- finite precision of phase-shifter control,
- photodetection noise,
- thermal crosstalk between phase shifters; however, it can be eliminated by using phase-change materials instead of the thermo-optic effect.

Such an accelerator was trained by forward propagation and the finite difference method:

$$\Delta W_{ij} = \frac{E(W_{ij} + \delta_{ij}) - E(W_{ij})}{\delta_{ij}}, \quad (2.16)$$

which would be inefficient on an electronic computer since computing  $E(\hat{W})$  is expensive. However, it takes almost constant time on a chip w.r.t. chip size. Ultimately, this chip was successfully used on the vowel recognition task in simulation and experiment [91].

This was an example of an MVP accelerator with coherent inputs and outputs. Therefore, only a small portion of the available bandwidth is used. Instead, one could consider an incoherent accelerator; one example is shown in Figure 2.17. There, each input corresponds to a particular wavelength, and input signals are encoded by modulating the amplitude of each channel. Then, multiplexed inputs are split into several rows of microring resonators (MRR), comprising a photonic weight bank. Each MRR picks up a particular channel, and then the transmission through the MRR can be controlled by controlling its resonance frequency. After that, the through and drop channels are passed

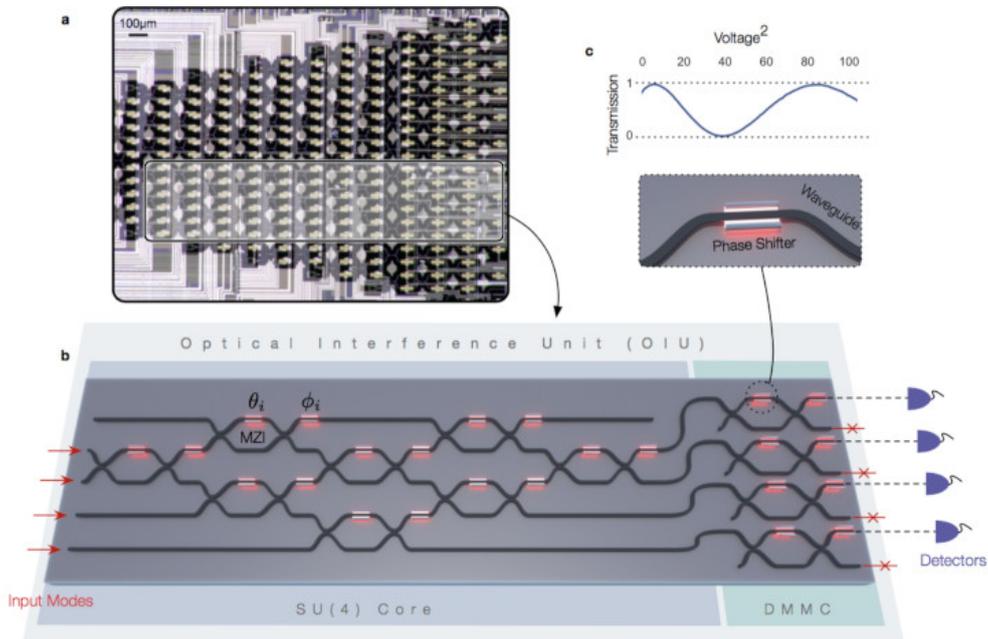


Figure 2.16: Coherent matrix-vector product accelerator on the silicon-on-insulator platform based on 56 Mach-Zehnder interferometers and 213 phase shifters [91].

to a balanced photodetector to incoherently sum channels. Here, using a balanced photodetector allowed for positive and negative channel weights. Afterward, a nonlinear transformation is applied by a transimpedance amplifier (TIA), converted into a digital signal, and received by a computer. Such an approach better utilizes available optical bandwidth but does not handle information encoded in phase, followed by an analog-to-digital conversion and communication with a computer.

Another approach is to use a crossbar array demonstrated experimentally in [93]. Each tooth of a frequency comb is modulated and passed through a dedicated waveguide on one side of a crossbar array (see Figure 2.18). Then, signals from each waveguide are coupled into perpendicular waveguides via couplers at every crossing, acting as matrix weights. The coupling strength is tuned by depositing on top of a coupling waveguide some phase-change material (PCM). Some examples of PCMs are  $\text{Ge}_2\text{Sb}_2\text{Te}_5$  (commonly referred to as GST) and  $\text{Sb}_2\text{Se}_3$ . The refractive index of a PCM in crystalline and amorphous states is different, and depending on a material, its crystallization can be partial, which allows the refractive index of a coupling waveguide to accept values within some interval [94]. The signals from perpendicular waveguides are the output read by an array of photodiodes. Such a device was applied for handwritten digit recognition.

Another interesting example is based on the photorefractive effect [95]

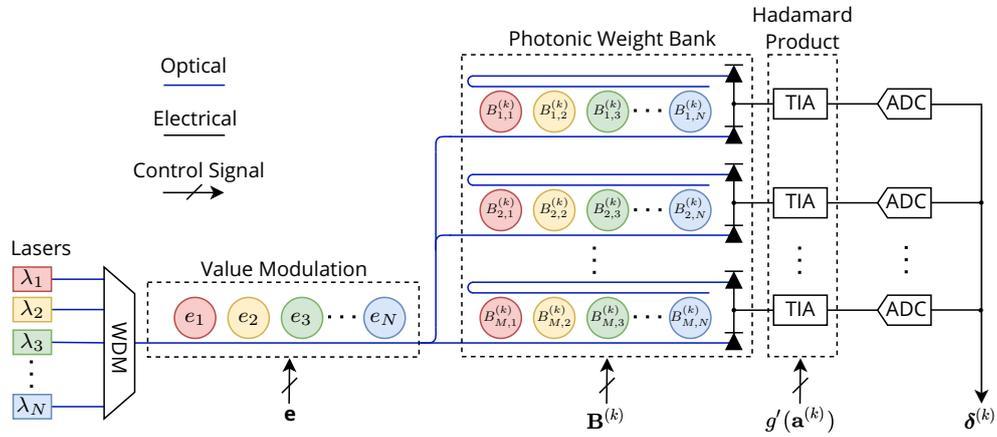


Figure 2.17: Incoherent matrix-vector product accelerator, details in text. Here, ADC is an analog-to-digital converter. Adapted from [92].

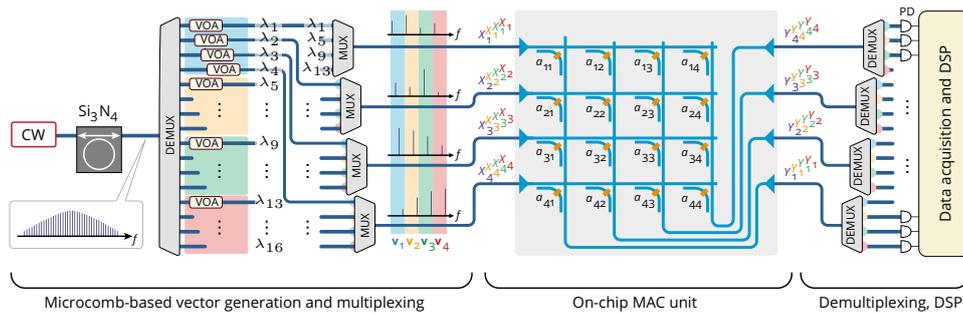


Figure 2.18: Incoherent matrix-vector product accelerator based on a photonic crossbar array. Here, CW is a continuous wave, VOA is a variable optical attenuator, DE-/MUX is de-/multiplexing, PD is a photodiode, and DSP is digital signal processing. Adapted from [93].

shown in Figure 2.19. In a thin sheet of GaAs, two beams are sent, and their interference causes electron excitation in particular parts of the material, which leads to local changes of refractive index, and this way, a matrix can be encoded. Afterwards, a single beam encoding a vector can be sent to perform an MVM. By providing input through a different side, the chip will perform the MVM with a transposed matrix, which is essential for backpropagation discussed in Section 2.3.

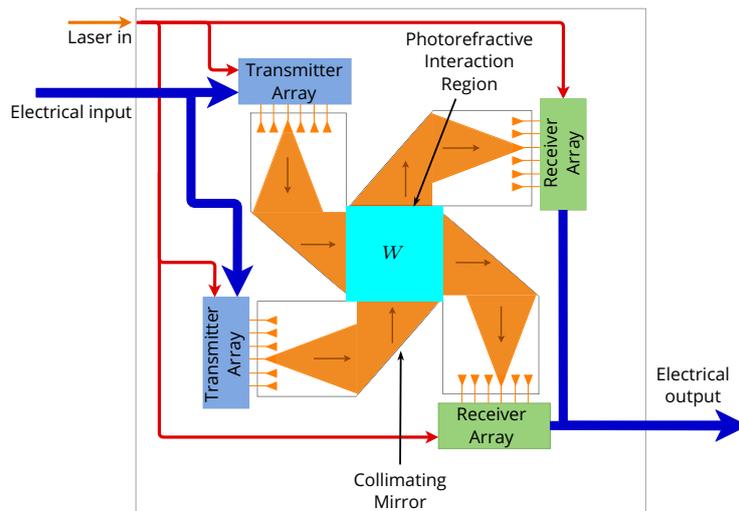


Figure 2.19: Matrix multiplication accelerator based on the photorefractive effect. Adapted from [96].

### 2.5.2 . Photonic neural networks

Implementing a complete neural network on a chip is a more challenging problem, with few implementations in the literature. Two recent examples of a coherent DNN on a chip are shown in Figure 2.20. The first implementation had six inputs encoded by on-chip modulators, one hidden layer with six neurons and an output layer with six output neurons. Networks of MZIs implement the MVP, while an optoelectronic nonlinearity – the activation function. Implementation-wise, the second example is similar but accepts a  $5 \times 6$  image formed by 30 incoherent signals that are then shrunk to 4 signals by a convolutional layer. Then, these signals are processed by hidden and output layers [97]. Weighting was performed with attenuators.

Implementing RNNs on-chip is even more challenging, as already difficult training is even harder on-chip. There has been a proposal with a partial implementation (see Figure 2.21). To the best of my knowledge, a complete ‘classic’<sup>3</sup> RNN on a photonic chip has not been implemented. Simplified RNNs – RCs – however, have been.

<sup>3</sup>i.e. all connections are trainable

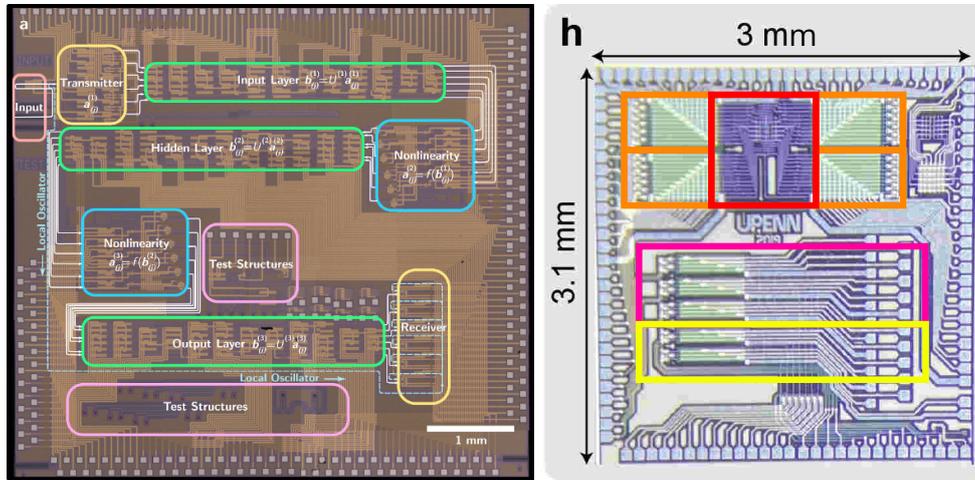


Figure 2.20: Deep neural networks on chip. Left: one hidden and output layers [98]. Right: input layer (red), convolutional layer with four neurons (orange), one hidden layer with three neurons (purple) and output layer with two neurons (yellow). Adapted from [97].

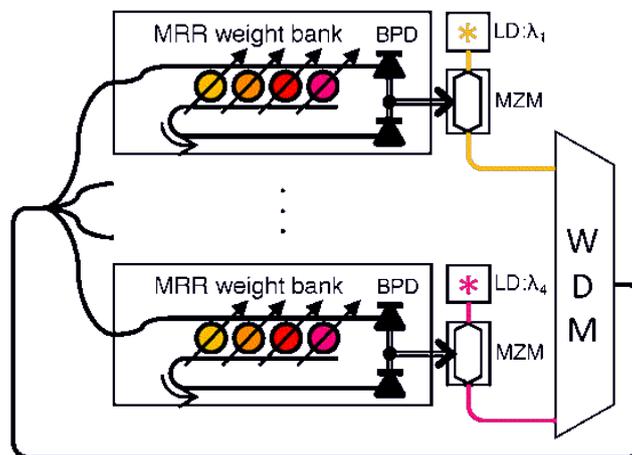


Figure 2.21: Proposal of a recurrent neural network on the integrated photonic platform. The output of a photonic weight bank is looped back as an input with a delay. Here, the delay is large enough (1 ns) to avoid time-delayed dynamics. Adapted from [99].

### 2.5.3 . Reservoir computers

The generality of RC allows for much freedom. Current implementations of RC can be classified into three groups:

- time-multiplexed,
- frequency-multiplexed,
- space-multiplexed,

but they can overlap. In this section we explain these groups and give their notable examples.

#### Time-multiplexed reservoir computing

This approach utilizes a single nonlinear node and a feedback loop [100] (see Figure 2.22). While a rigorous system description involves a spatio-temporal analogy and is nontrivial [101], an intuitive explanation is provided here

When a signal is circulating in the feedback loop, it eventually enters the node, where it is nonlinearly transformed ( $f$  in Eq. (2.3)). Sometimes, the node's bandwidth is designed to be lower than the signal bandwidth. Then, the node's impulse response function will effectively compute a linear transform of a recently passed signal, coupling the portions of the signal ( $\hat{W}$  in Eq. (2.3)) in a ring architecture. This is how the RC state is evolving.

In order to perform computing, inputs need to be provided. For that, a multiplicative temporal mask is applied on an input, and the result is then passed to the node, which will add it to a circulating signal. Using a mask is critical as it breaks the otherwise perfect symmetry of the ring network implemented by the delay system. Therefore, it ensures high-dimensional responses in the non-chaotic regime. The length of a mask can differ for more complex coupling architecture. For example, a small difference can overcome the lack of mixing in the node if its bandwidth is instantaneous [102]. Each mask step corresponds to a single RC neuron, and amplitudes of mask elements encode the  $\hat{W}_{in}$  matrix. In order to implement the readout, signals corresponding to each neuron are recorded from the feedback loop and weighted.

The simplicity of this approach secured its popularity in experimental realizations. The first time-multiplexed RC with an optoelectronic feedback were proposed almost at the same time [102, 104] shown in Figure 2.23. A mask is applied electronically to produce an RF input, which is then combined with an existing RF signal in the loop, and the result is used to modulate an optical carrier using an MZI, which acts as nonlinearity since the output optical signal is proportional to the cosine of an electric signal. The optical signal then passes through a spool of fiber and gets converted back into an RF signal by a

---

<sup>4</sup><https://arxiv.org/abs/2111.03332>

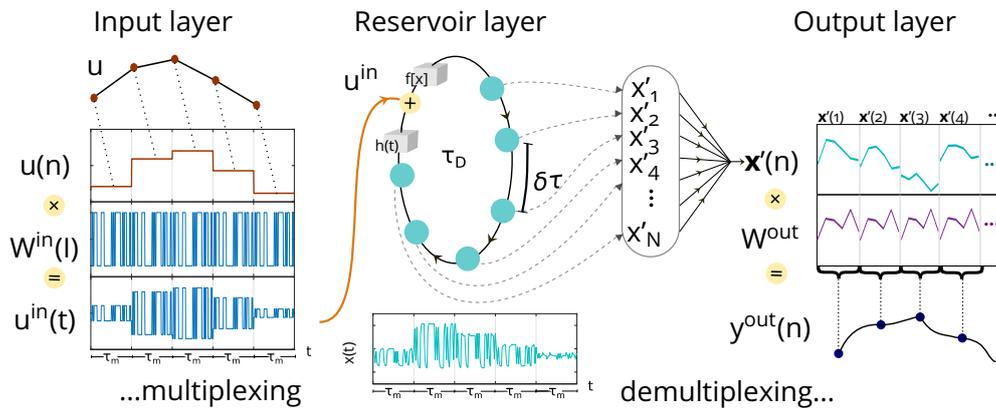


Figure 2.22: Time-multiplexed reservoir computing scheme. Adapted from the arXiv source<sup>4</sup> of [103] due to higher quality.

photodiode, which is then looped back to the RF combiner. A readout was implemented electronically. Such reservoirs were used for speech recogni-

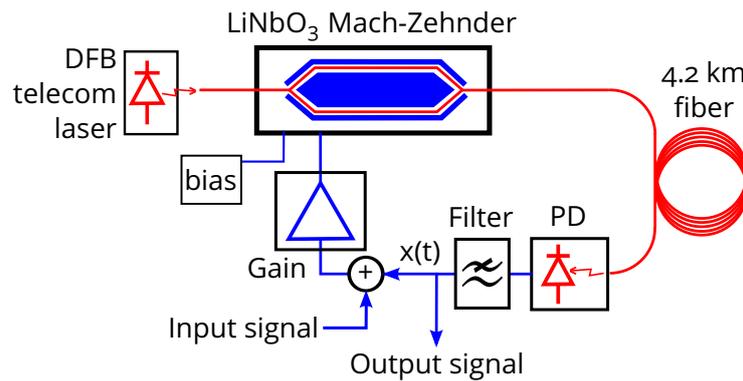


Figure 2.23: Time-multiplexed photonic reservoir computer implementation based on optoelectronic feedback. Adapted from [104].

tion [104, 102], waveform recognition, the prediction of a 10th-order nonlinear autoregressive-moving average (NARMA10) of a random signal and nonlinear channel equalization [102]. Later, a semiconductor laser with optical feedback was used for spoken digit recognition and chaotic series prediction [105].

There is also an effort towards the integration of time-multiplexed RCs. However, to obtain a sufficient number of virtual neurons in an RC, a feedback delay has to be long, which is challenging in integrated waveguides and why so many opt for an optical fiber instead. Nevertheless, there have been efforts towards the integration of delays as well. For example, a delay was implemented as an external cavity of a distributed feedback (DFB) laser [106] (see Figure 2.24, left). Such a reservoir was used for nonlinear channel equalization and chaotic time series prediction. A later work of this group tried another ap-

proach: reducing the feedback length, sacrificing the number of virtual neurons, and compensating with multiple independent feedback loops [107] (see Figure 2.24, right). This is one example of both time- and space-multiplexed RC, and it was used for the same tasks as a previous work. Another promising option is the silicon nitride platform that allows for less than 1 dB/m loss and would be able to accommodate long delay lines, although at the cost of a considerable bend radius on the order of 1 mm [108].

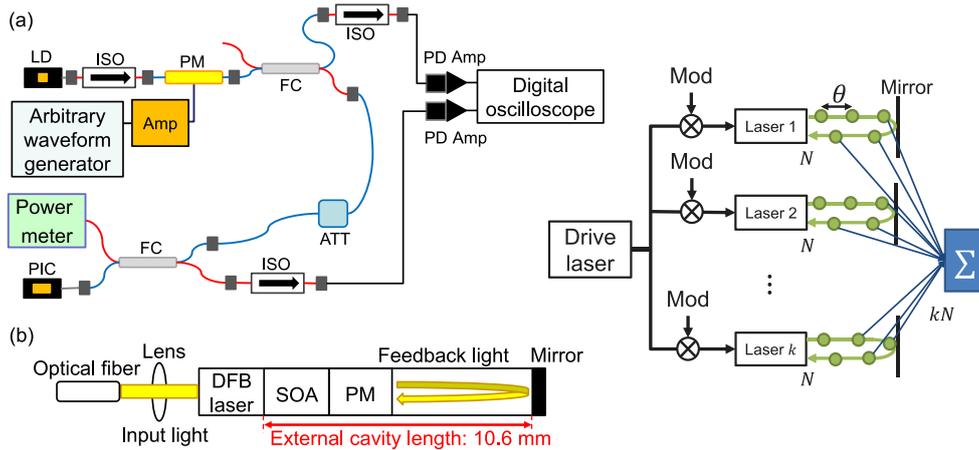


Figure 2.24: Left: integrated photonic time-multiplexed reservoir computer based on a distributed feedback laser [106]. Right: improvement of the design by using multiple delay lines [107].

Such an architecture has a significant drawback regarding high-speed computing: masking. In order to process a signal with a bandwidth of  $f$  GHz signal by an RC with  $N$  neurons, one would require a modulator with a bandwidth of  $Nf$  GHz to apply a mask, which requires high-speed electronics. It means that with this architecture, one has to trade computing power for speed, at least at this time. Since the tasks to be solved are nontrivial and require high speed, we chose not to use time-multiplexing in this work.

## Frequency-multiplexed reservoir computing

This newly-emerged approach separates neurons by frequency [109]. In this theoretical work, an optical fiber was looped to create a cavity, which holds a signal composed of several on-resonance spectral lines. Then, a phase modulator is used to mix them. For readout, this signal is demultiplexed and detected by photodiodes. This RC was used on a nonlinear channel equalization task, with the integration on the InP platform envisioned, where 2.5 GHz input signals can be processed.

A later work of this group demonstrates an implementation of such a setup [110]. The fiber is 15 m long, corresponding to a 49.5 ns round-trip

time and free spectral range (FSR) of about 20.2 MHz (see Figure 2.25). The timescale of the input signal matched the round-trip time, i.e. the processing speed is also 20.2 MHz. The readout is implemented by picking up each line by a pass-band filter, detecting by a photodiode, and then applying weights electronically. Alternatively, a programmable spectral filter can be used to attenuate all lines simultaneously and a balanced photodetector would allow for negative weights. This reservoir had 25 neurons and was used for nonlinear channel equalization and chaotic time series prediction. A recent follow-up work improved on this setup and used a deep RC architecture, where an output of the first RC is an input to the second RC [111].

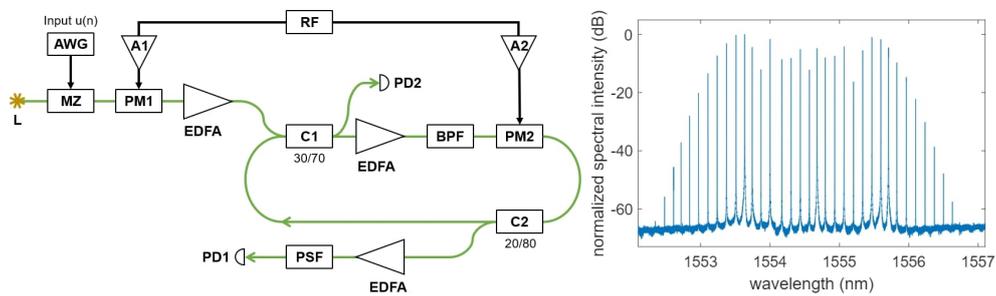


Figure 2.25: Left: scheme of a frequency-multiplexed reservoir computer. Right: spectrum of a signal in optical fiber. Adapted from [110].

This approach utilizes the main advantage of optics – a large bandwidth containing many spectral lines, and with the development of on-chip frequency comb sources, this method has integration potential. However, as of now, it relies on high-speed electronics for phase modulation and remains limited in the number of neurons.

### Space-multiplexed reservoir computing

This method separates neurons in space, which makes for a more intuitive translation of ANN to a physical substrate. Therefore, contrary to time- and frequency multiplexing, physical implementations with such an approach started appearing much earlier [57].

**Semiconductor Optical Amplifier-based reservoir** One of the first proposals for an integrated photonic RC used a  $5 \times 5$  network of semiconductor optical amplifiers connected with waveguides [112]. In this theoretical work, SOAs are used as neurons: they perform coherent summation of optical signals arriving through the waveguides and apply a nonlinear transformation due to gain saturation. The steady-state characteristic of SOAs is similar to hyperbolic tangent commonly used as an activation function (see Figure 2.26). They also provide gain, so no separate devices were needed to counteract losses, and their broadband operation is favorable for a prototype. Waveguides are arranged in a swirl topology such that optical signals loop back on themselves, allowing for complex self-mixing, assisted by the SOA nonlinearity (see Figure 2.26). The optical field was coherent, but the readout weights were real-valued since an electronic readout was planned at that point. This reservoir was used for

- pattern recognition, where the amplitude of an optical carrier is modulated with various waveforms, and the reservoir has to classify them,
- prediction of the NARMA [113], but its results were planned to be published later.

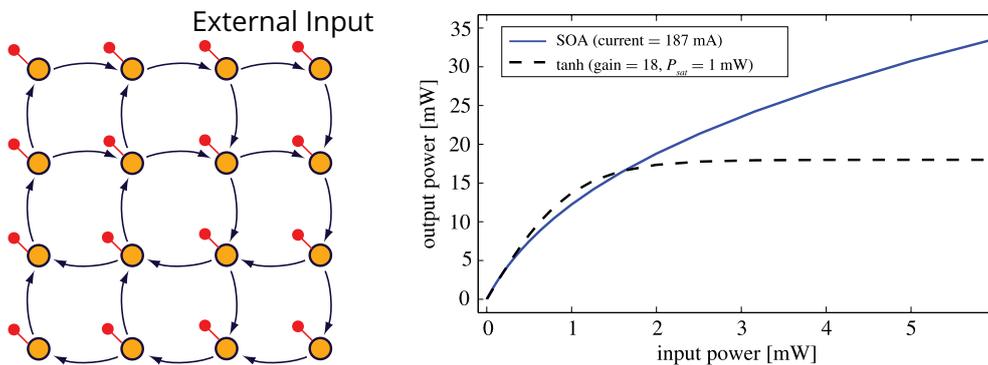


Figure 2.26: Left: swirl topology, only  $4 \times 4$  is shown, but the actual size is  $5 \times 5$  [112] or, later,  $9 \times 9$  [114]. Right: semiconductor optical amplifier (SOA) nonlinearity steady-state characteristic compared to a hyperbolic tangent. Adapted from [114].

In a later version of this work [114], a larger version of this RC was applied to a speech recognition task where recordings of digit pronunciations are passed as input, and the reservoir's task is to classify digits. Recordings were accelerated to match optical timescales. Additionally, recordings were preprocessed according to the ear model [115], which outputs 77 channels that were fed into the reservoir as inputs with power below 1 mW. The length of the connecting waveguides was optimized for the best performance. It was found that the reservoir performs best if the propagation of light between two neurons takes

190 ps, which roughly corresponds to half of the recording length. According to the authors, it was easy to achieve almost perfect classification, so a babble noise at 3 dB signal-to-noise ratio (SNR) was added, then the classification error was 7.7%. The downside of such a setup is that SOAs require considerable power [116].

**Passive reservoir** This group's later work revolved around a similar network of splitters [116]. This time, it was theoretical and experimental work with a fabricated chip on the SOI platform (see Figure 2.27, left). Similar to SOAs, splitters receive signals from multiple waveguides and output their sum, however, without a nonlinear transformation. Nonlinearity came from the readout instead: since a photodiode was used to extract signals from each splitter, a  $|x|^2$  nonlinear function was applied to every signal. Because of that, a reservoir was capable of solving nontrivial tasks:

- binary operations, including XOR, where an amplitude of an optical carrier was modulated with a random signal with two levels, corresponding to 'true' and 'false' and the RC has to compute a binary operation of last two bits,
- header recognition, where a signal is the same as in the previous task, but the reservoir has to classify patterns of a certain number of bits, which is applicable for all-optical routing,
- speech recognition described in the previous section.

It has been demonstrated that a linear reservoir cannot compute XOR, which is a nonlinear function, but both  $|x|$  and  $|x|^2$  nonlinearities before the readout make it possible.

A recent experimental work by this group aimed at improving connectivity [30]. In a swirl configuration, neurons at the edges and corners have only one input, which as suggested to be a performance limiting factor, so a new four-port architecture was proposed, where all neurons receive two inputs (see Figure 2.27, right). This reservoir was used for an optical data recovery task where a 32 Gbps signal encoded with on-off-keying (OOK, see Section 4.2.1) was recovered after distortion by a 25 km single mode fiber.

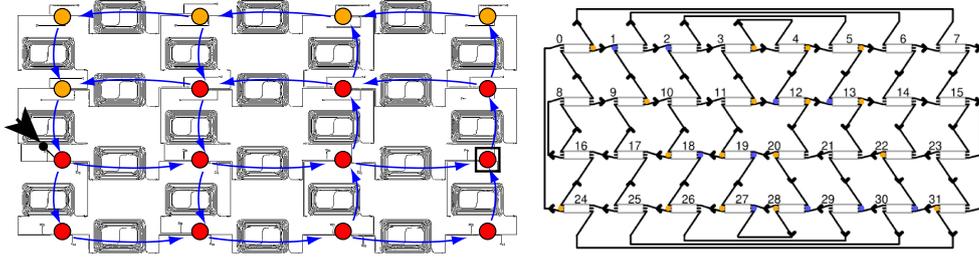


Figure 2.27: Integrated reservoir computer with whirl (left) and four-port (right) topology on the silicon-on-insulator platform. Adapted from [116, 30].

**Nonlinear photonic crystal-based reservoir** Another work considered a rectangular grid of 25 nonlinear (the Kerr effect) PhCs as neurons [117] in simulation. The task was to autonomously generate a specific output signal with arbitrary waveform generation in mind. For that, the output was fed back as an input, which made ridge regression inapplicable. Instead, the FORCE algorithm was used. The target signal was a sum of two sines with different frequencies:

$$s(t) = \sin(\omega_1 t) + \sin(\omega_2 t) \quad (2.17)$$

where  $\omega_1 \approx 2\pi \times 216$  GHz and  $\omega_2 = 0.311\omega_1/0.2$ , the choice was based on the cavity photon lifetime - 1.39 ps.

Contrary to previous works, the delay between the cavities was removed, though splitters and waveguides were still used for coupling. That, however, required randomness of the waveguide coupling phase and stronger coupling between the cavities. It was explained that otherwise, cavities enter a harmful collective self-pulsing.

**Nonlinear microring resonator-based reservoir** This group's later work returns to the whirl architecture and improves it by incorporating nonlinear Si MRR cavities in a 4x4 grid [118]. This reservoir could solve the delayed XOR task at 20 GHz while requiring a few milliwatts of optical power. Photodiodes were used in the readout, and even though an improvement with higher input power was reported, it is not clear if intrinsic nonlinearity has contributed significantly since the same task in similar conditions has been solved by a linear reservoir with photodiodes in the readout [116]. A similar theoretical article has considered random delays between MRRs [24] in a rainfall topology and demonstrated 8-bit header recognition at 160 Gbps.

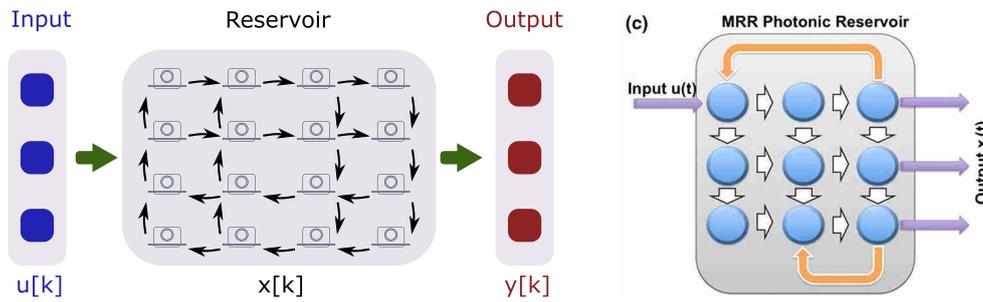


Figure 2.28: Microring resonator-based reservoir computer in swirl [118] (left) and rainfall [24] (right) topologies.

**Passive photonic crystal-based reservoir** A different approach using a large quarter-stadium-shaped PhC was also demonstrated in a simulation [31]. Interacting with a cavity was possible via wall defects: 1 for input and 5 for outputs. Such a linear reservoir equipped with photodiode nonlinearity could solve tasks at a wide range of bitrates: binary operations including XOR up to 67 Gbps and up to 6-bit header recognition up to 100 Gbps.

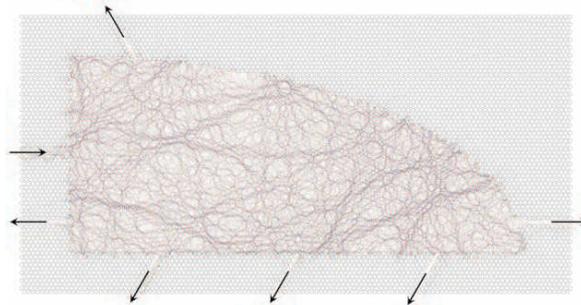


Figure 2.29: Integrated photonic reservoir based on a photonic crystal cavity. Adapted from [31].

**Free-space setup with large-area laser** In a recent article, a large-area VCSEL was combined the advantage of free-space optics – a high number of neurons (here 350) – with a nonlinearity of integrated photonics [119] and is shown in Figure 2.30. A digital micromirror device (DMD) encodes an input image sent through a multimode fiber, performing input weighting. Then, the weighted image is directed into a large VCSEL, where spots of high-intensity light increase the local density of charge carriers that cause the nonlinear response. The drift of charge carriers spreads the nonlinear response to neighbouring regions, creating a Gaussian-like local coupling between neurons. Finally, the result is sent to a second DMD that performs output weighting. This RC was used for the header recognition task, and it was noted that this task

could not be solved without a VCSEL, which also increased the dimensionality of the RC.

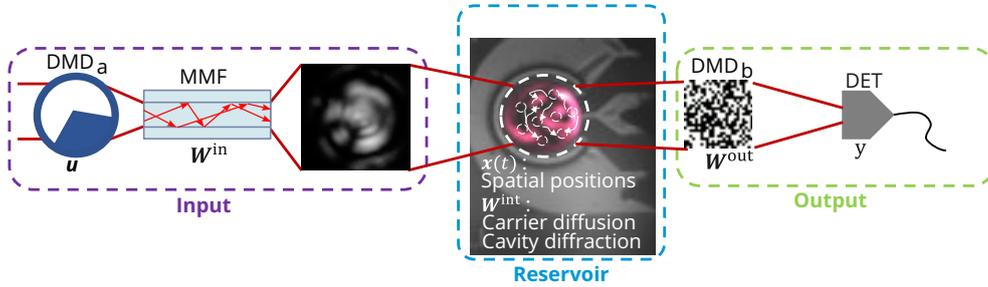


Figure 2.30: Reservoir computer based on a large area vertical cavity surface emitting laser. Here, DMD is a digital micromirror device, MMF is a multimode fiber, and DET is a photodetector. Adapted from [119].

**Spatio-temporal reservoir** The experimental work of Nakajima M. et al. merged space- and time-multiplexed approaches to improve RC scaling [76] (see Figure 2.31). There, modulated pulses are split into  $N$  parts subject to various delays. Then, they are linearly mixed into  $L$  signals that are then injected into an array of cavities that act as delays for time-multiplexing. The array of photodiodes reads neurons from these cavities, and the readout is performed on a computer. However, cavities are linear, and nonlinearity comes from the optical modulator at the input stage and from photodiodes in the readout. This RC was applied for chaotic time series prediction and handwritten digit classification.

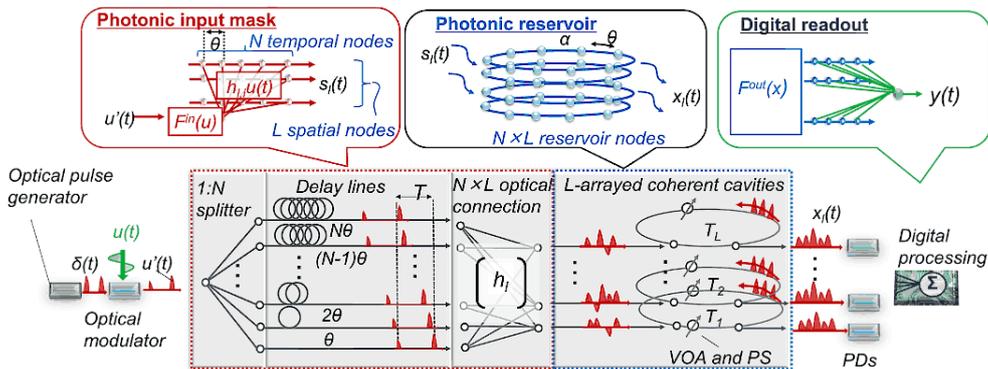


Figure 2.31: Spatio-temporal reservoir computer based on an array of coherent cavities. Here, VOA is a variable optical attenuator, PS is a phase shifter and PD is a photodetector. Adapted from [76].

## 2.6 . Scalability of integrated photonic neural networks

To solve more challenging tasks, an ANN with more neurons is required. However, chip space is always limited and so the goal is to get as much computing power as possible out of available space.

This, however, is a recurring problem with integrated photonics – one could have noticed that most designs above implement modest ANNs with just a few tens of neurons. Of course, those are research prototypes, but a strategy for scaling up conventional ANNs on-chip is yet to be determined.

For comparison, ANNs in free-space setups often use spatial light modulators to implement internal weights and are then limited by their resolution. For example, the setup shown in Figure 2.32 (left) used 2025 neurons for chaotic series prediction with, 90,000 being possible [120] and another – 16,384 neurons for human action recognition. A theoretical limit was 262,144, though not all of them could be in the focus of a camera [121]. Diffractive units promise even higher numbers. One work proposed to train a large DNN on a computer and, based on its parameters, fabricate diffractive layers using a 3D printer (see Figure 2.32, right). In the experiment, 450,000 neurons were demonstrated, and tens of millions of neurons were claimed to be possible, though the parameters of layers cannot be changed after fabrication [122].

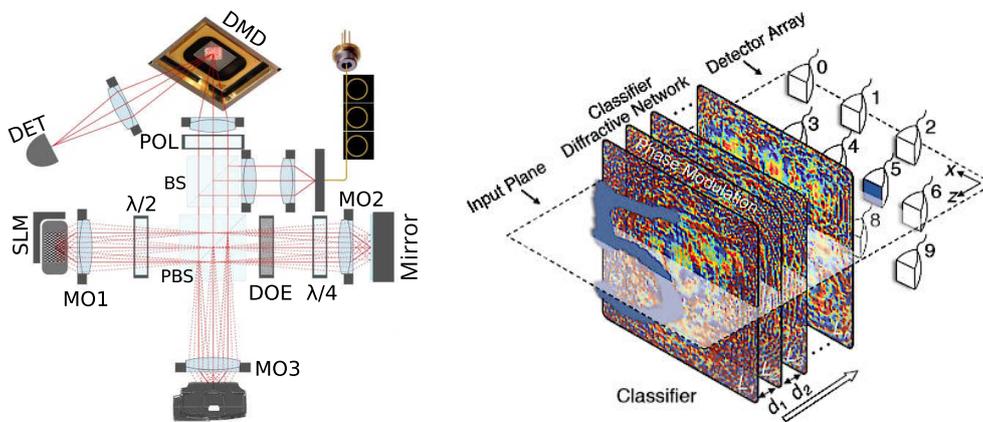


Figure 2.32: Left: recurring neural network in free-space optics based on a spatial light modulator (SLM) and digital micromirror device (DMD) that implement internal and readout weights. Here, DET is a detector, SLM is a spatial light modulator, (P)BS is a (polarizing) beam splitter, MO is a microscope objective, DOE is a diffractive optical element, Right: deep neural network based on layers of diffractive planes. Each point on a layer is a neuron with complex-valued transmission, and interference of the diffracted light creates coupling. Adapted from [120, 122].

Here, we estimate the efficiency of designs found in literature in two ways: with a neuron density and a synapse density. The former is the same as MAC

per area, which is directly linked to the computing power. However, synapses can also occupy space; their number can scale quadratically with the number of neurons. The synapse density, however, will not and is, therefore, a more general measure.

We start with the MZI-based MVP accelerator [91] in Figure 2.16, which can perform a  $4 \times 4$  matrix multiplication while taking, judging by the photo,  $2.6 \text{ mm} \times 2.0 \text{ mm}$  or  $5.2 \text{ mm}^2$  of chip area. Considering that an MVP imitates MAC operations of 4 neurons, a neuron density is around  $0.7 \text{ neurons/mm}^2$  and  $3 \text{ synapses/mm}^2$ .

The incoherent MVP accelerator [92] in Figure 2.17 was not fabricated as a whole, but based on the photos of prototype MRRs to be used, a rough estimate could still be made. Each MRR takes roughly  $0.06 \times 0.03 \text{ mm}^2$  of space, considerably less than a MZI. For the same four neurons, we would need four MRRs for modulation and 16 MRRs for a weight bank. Assuming the maximally efficient design, we would need  $0.036 \text{ mm}^2$  of space. Of course, since weights are set thermally, the spacing between devices would need to be increased to reduce the thermal crosstalk, but we would still get a result on the order of  $100 \text{ neurons/mm}^2$  and  $450 \text{ synapses/mm}^2$ . Much better!

The work on a crossbar array [93] shown in Figure 2.18 implemented a  $16 \times 16$  MVP accelerator on a roughly  $8 \times 9 \text{ mm}^2$  chip, corresponding to around  $0.2 \text{ neurons/mm}^2$  and  $3.3 \text{ synapses/mm}^2$ .

The integrated DNN [98] shown in Figure 2.20 has six inputs, three  $6 \times 6$  layers, two layers of nonlinearity and six outputs. Excluding test structures, the chip area is roughly  $30 \text{ mm}^2$ , corresponding to the neuron density of  $0.8 \text{ neurons/mm}^2$  or  $3.6 \text{ synapses/mm}^2$ . It is not surprising that it matches the MZI-based accelerator since matrices here are also implemented with MZIs, which take up most of the chip space.

As for RCs, the passive version [116] shown in Figure 2.27 (left) occupies  $16 \text{ mm}^2$  of chip space while having 16 neurons and 24 synapses, which gives  $1 \text{ neuron/mm}^2$  or  $1.5 \text{ synapses/mm}^2$ . There, most of the space was taken by spirals to slow down the system. Moreover, to reduce losses, waveguides were shallow-etched and, therefore, had a larger bend radius of  $40 \text{ }\mu\text{m}$  (SOI platform).

The footprint of the MRR-enhanced version [118] in Figure 2.28 had not been estimated, but since delay loops were still there, the results would be similar to the RC above.

For the RC with the four-port architecture [30] in Figure 2.27 (right), the chip footprint was also not specified, but the chip was fabricated on the bipolar CMOS SiGe platform. Assuming that shallow-etched waveguides were used again, the chip footprint is estimated by the waveguide bend radius – about  $16 \text{ mm}^2$ , which with 32 nodes (including non-observable ones) and 64 synapses gives  $2 \text{ neurons/mm}^2$  and  $4 \text{ synapses/mm}^2$ .

It is unclear how to estimate the neuron density of the RC based on a large PhC [31] in Figure 2.29. The size of the structure is  $30 \times 60 \mu\text{m}^2$  with six outputs. However, looking at the complexity of field propagation inside the crystal, intuitively, more signals could have been extracted. However, even with six outputs, there are 300 neurons/ $\text{mm}^2$ . This demonstrates an important point of RC – *if no particular structure is imposed and only internal complexity is required, there is a potential for a large neuron density.*

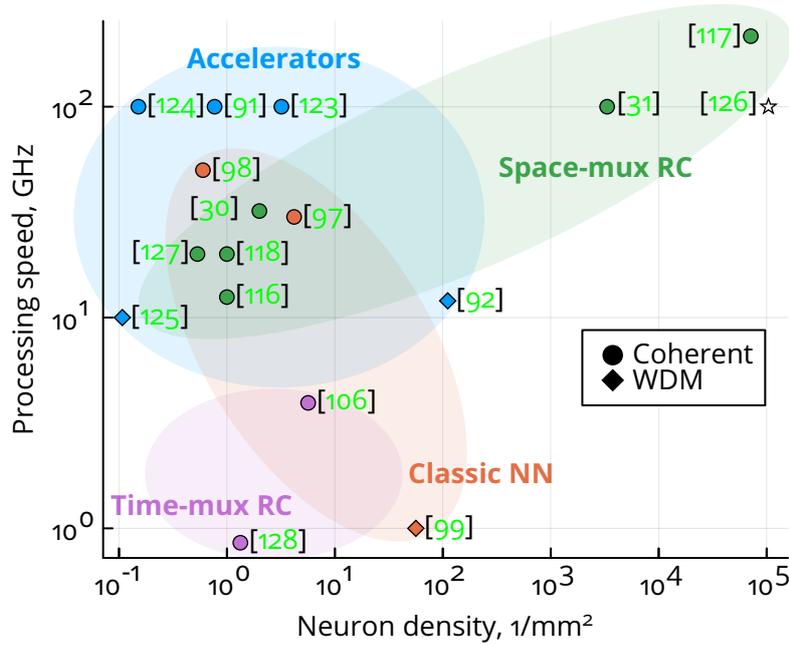


Figure 2.33: Overview of integrated neuromorphic setups. Details are given in Table 2.1. The work of Khoram E. et al. [126] is a photonic scatterer used for inference.

The complete scalability comparison is given in Figure 2.33. It is true that such a comparison is not particularly fair due to differences between systems. For one, accelerators and complete neural networks pursue different goals. Accelerators can perform almost arbitrary MVMs but rely heavily on electronics. Complete ANNs perform more operations internally and do rely as much. Moreover, in a DNN, there are fully tunable MVM and nonlinearity layers, and the readout was on-chip. In RCs shown here, MVM is not controlled, and often, the nonlinearity often came from an external photodetector, and readout was implemented electronically. In any case, the goal was to have a rough estimate of compute density, which dramatically increased, when WDM or the RC paradigm was used.

Nevertheless, integrated structures are two-dimensional, which limits the scalability of the number of I/O channels of interconnects [129], and there is an effort towards their three-dimensional implementation. For example,

Table 2.1: Details for Figure 2.33. For linear setups for which the processing speed was not discussed, a detection limit of 100 GHz is assumed according to [91].

Ref.	Notes
[123]	linear waveguides
[124]	chip size is estimated by stud holes, neuron size is interaction length (200 $\mu\text{m}$ ), linear operation after weight writing
[92]	speed is DAC-limited
[125]	speed limit not discussed, 10 GS/s AWG was used
[97]	30 GHz is a lower limit of available technology
[126]	scatterers can be nonlinear still assuming detection-limited speed

using two-photon polymerization, J. Moughames et al. demonstrated three-dimensional splitters up to  $1 \times 16$  [130] and a coupler structure with 225 inputs and 529 outputs in the  $0.46 \times 0.46 \text{ mm}^2$  area [129]. This way, the number of I/O channels of interconnects scales linearly with the chip area.

### 2.6.1 . A question of readout

In many implementations of space-multiplexed RCs, photodiodes in the readout add nonlinearity. In the experiments, a single photodiode is used to extract all signals from the RC: an input signal is passed to an RC, and a signal is extracted from a single neuron. Then, the fiber is moved to a second neuron, the same input is injected into the network again, and the process is repeated until all neurons have been read. High-speed real-time computing, however, will warrant the addition of a photodiode for each neuron. For example, the *imec* foundry proposes Ge photodetectors, which, judging by given microscope images in the datasheet<sup>5</sup>, take about  $0.15 \times 0.02 \text{ mm}^2$  of space, not considering RF pads. Then, in the best-case scenario, such an incoherent readout would limit the neuron density to 300 neurons/ $\text{mm}^2$ , and a coherent version would take even more space, not to mention more complicated handling of electric RF signals.

For that reason, an integrated readout has been proposed [131], which allows the weighting and summation of RC signals on-chip (see Figure 2.34). Not only does this allow the use of a single photodiode, but also an optical output. However, the nonlinearity of a single photodiode might be insufficient. For that reason, the use of intrinsic nonlinearity would be advantageous. Out of the aforementioned space-multiplexed integrated RCs [114, 117, 24, 118] have utilized it, whereas [116, 31, 30, 127] opted for the photodiode-based nonlinearity.

One might argue that weighting with MZIs is not as compact as one could

<sup>5</sup><https://www.imeciclink.com/en/asic-fabrication/si>

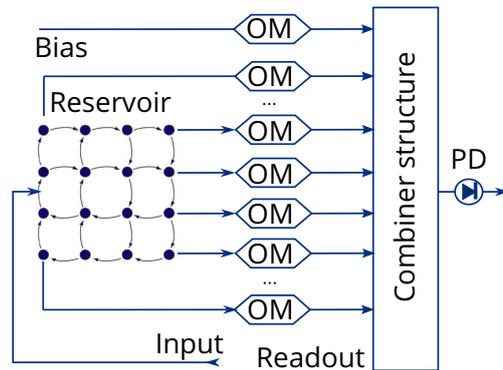


Figure 2.34: Swirl reservoir connected to a fully optical readout. Here, optical modulators (OM) perform weighting and the combiner structure – summation. Adapted from [131].

hope – the state-of-the-art footprint is  $0.10 \times 0.02 \text{ mm}^2$  [132], not taking thermal crosstalk into account. Luckily, phase change materials (PCMs) are in active development.

Crystal and amorphous phases of a PCM can have a different refractive index, which can be used for phase shifting and might become a viable alternative to conventional thermo-optic effect-based phase shifters [133]. Crystallization of a PCM can be partial, and then its effective refractive index attains some intermediate value. For example,  $\text{Ge}_2\text{Sb}_2\text{Te}_5$  or GST has been demonstrated to be capable of switching up to 34 levels [134], while  $\text{Sb}_2\text{Se}_3$  – only a few [135]. A good review of PCMs is given in [133]. It has been demonstrated that even with a limited-resolution readout, a proper training procedure correction, having eight levels of precision is manageable, and 16 levels is not much different from the full precision [136]. Apart from compactness, non-volatility is another interesting property. A thermo-optic phase shifter requires a constant supply of current to operate, but a PCM can stay for prolonged periods of time in the same state [133]. Although not as compact, microelectromechanical systems (MEMS) also have their benefits, such as higher energy efficiency and absence of thermal crosstalk – a good review is given in [132].

## **2.7 . Conclusion**

There is a push towards the implementation of ANNs on a photonic chip. While complete RNNs have not been implemented yet, their simplified version, RCs, have been due to easier implementation and training procedures. Current implementations have been successful but were limited by large chip footprint to slow down the propagation of light or underutilized intrinsic non-linearity, which would require high-speed photodetectors on each neuron to solve nonlinear tasks. In the next chapter, a new approach to integrated photonic RC is proposed that aims to overcome both these issues.

## 3 - Integrated reservoir computer based on evanescent coupling

### 3.1 . Introduction

Pioneering research by K. Vandoorne et al. demonstrated how optical signal can be mixed with its delayed copies in a complex fashion to perform computing, how mixing can be improved by using intrinsic physical properties of SOAs [114], and that a nonlinear transform at the readout can compensate for linear RC internals [116]. F. Laporte et al. took the RC paradigm very close to heart, used unstructured but complicated media as a large PhC resonator, and drastically reduced the chip footprint [31]. M. A. A. Fiers et al. demonstrated that an intrinsic nonlinearity can generate a nontrivial waveform that is impossible with a linear RC [117].

However, a few questions remained. What is the exact role of chosen parameters besides a benchmark performance? Is there a profound meaning in delayed connections between neurons? Can an all-optical RC be slowed down such that interfacing with other (electronic) devices is possible without high footprint cost? Do different nonlinearities provide the same benefit? The final and the biggest question: is it possible to simultaneously achieve scalability and all-optical computation with intrinsic nonlinearity?

Guided by these questions, this chapter, considers an integrated RC based on evanescently coupled nonlinear microresonators and discusses how its parameters impact general performance metrics: dimensionality and consistency.

### 3.2 . Microresonator as a neuron

The linearity of light has its up- and downsides. On the one hand, light is predominantly used in data transfer, and here, nonlinearity leads to distortion and crosstalk between channels. On the other hand, our goal is all-optical neuromorphic computing, where neurons are supposed to perform local nonlinear transformations, i.e. without the involvement of photodetectors and the conversion of the signal into the electric domain. For that end, light has to reach a high enough intensity, which is made practical inside optical microresonators. Here we will consider dielectric and semiconductor resonators, where internal loss may be very low.

### 3.2.1 . Microresonator model

An dielectric or semiconductor microresonator (or cavity) is an object capable of localizing light by exploiting the laws of refraction. Depending on the type of cavity, refraction will differ. For example, in Fabry-Perot resonators, a light wave impinges perpendicularly on the mirror, which could be, for example, a cleaved crystal facet (see Figure 3.1). In microdisk or microring resonators, light is guided along a curved path so that the total internal reflection is satisfied.

After propagating through a resonator multiple times, the light waves interfere, and if the interference is constructive, the wave intensity increases. Interference only happens for a particular set of frequencies, corresponding to the eigenmodes of the homogenous (i.e. no-source) Maxwell equations for the electromagnetic (EM) field. An analytical approach to finding them is often untractable, and numerical approaches are used instead, such as the Finite Difference Time Domain (FDTD) method. Since the computer implementations of these methods are readily available, mode frequencies are assumed to be known. Integration of Maxwell equations is a computationally intensive

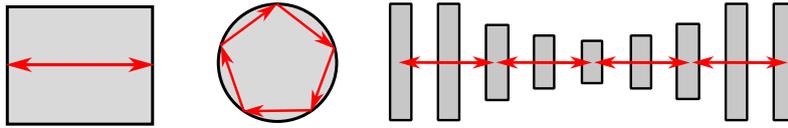


Figure 3.1: Fabry-Perot (left), microdisk (center) and photonic crystal (right) resonators seen from the top. Red arrows represent the direction of electrooptical wave propagation.

process, and a simulation of even one cavity might take considerable time; therefore, to simulate a network of cavities, simplifications must be put in place.

Assume that our resonators we use are capable of conserving resonant modes for a long enough time, or, in other words, their Q-factor is large i.e.

$$Q = \tau_0 \omega \gg 1, \quad (3.1)$$

where  $\tau_0$  is the lifetime of the EM energy stored in that mode of the resonator, i.e. a duration of time during which it reduces  $e$  times, and  $\omega$  is the resonance frequency of a resonator. Similarly, we also assume that the timescale of other effects, such as interactions of this cavity with waveguides and other cavities, are much longer than  $1/\omega$ . In this case, fast spatiotemporal changes of the EM field are averaged over time, and the Coupled Mode Theory (CMT) becomes applicable [137]. Then, the electric field in the cavity can be factorized into the complex envelope  $a(t)$  and the normal mode  $U(\mathbf{r})$  [138]:

$$\mathbf{E}(\mathbf{r}, t) = a(t)U(\mathbf{r}), \quad (3.2)$$

units of which are chosen such that

$$\frac{1}{2} \int \varepsilon_0 \varepsilon_r |\mathbf{U}(\mathbf{r})|^2 d\mathbf{r} = 1, \quad (3.3)$$

and, consequently, EM energy in the cavity is  $|a(t)|^2$ . For a single lossless resonator, its complex envelope is described as

$$\frac{da}{dt} = i\omega a. \quad (3.4)$$

In this case,  $a \sim \exp(i\omega t)$  and  $|a(t)|^2$  is therefore constant, meaning that the resonator does not lose the stored energy, which never happens in reality due to, for example, imperfections of cavity surfaces, which scatter the light. These losses are linear and are described by adding an intrinsic loss term:

$$\frac{da}{dt} = \left( i\omega - \frac{\gamma^\circ}{2} \right) a, \quad (3.5)$$

where  $\gamma^\circ = 1/\tau_0$  describes how quickly a resonator loses the stored energy. Here,  $\gamma^\circ$  is divided by two to correspond to the energy loss rate:

$$\frac{d}{dt}|a|^2 = \frac{da}{dt}a^* + a\frac{da^*}{dt} = -\gamma^\circ|a|^2 \quad (3.6)$$

where  $a^*$  stands for a complex conjugate of  $a$ .

### 3.2.2 . Nonlinear effects

We account for nonlinear effects by adding another term to the cavity equation:

$$\frac{da}{dt} = \left( i\omega - \frac{\gamma^\circ}{2} \right) a + f_{\text{NL}}(a). \quad (3.7)$$

We consider three nonlinear effects present in integrated photonic microresonators: two-photon absorption (TPA), free carrier dispersion (FCD) and the Kerr effect.

Assume a photon travelling through a semiconductor. If its energy is larger than the band gap, it can be absorbed to excite an electron in the valence band and move it to the conductance band (see Figure 3.2). Otherwise, it does not happen due to the energy conservation law. However, if there are two photons, each with energy below the band gap, but together, their energy exceeds it, they can be absorbed at effectively the same time and still excite that electron. This process is called TPA, included in Eq. (3.7) as [139]

$$f_{\text{NL}}(a) = -\frac{1}{2}\gamma^{\text{TPA}}(a)a, \quad (3.8)$$

where

$$\gamma^{\text{TPA}}(a) = \frac{\beta^{\text{TPA}}c^2}{n^2V^{\text{TPA}}}|a|^2, \quad (3.9)$$

$\beta^{\text{TPA}}$  is the TPA coefficient defined as  $\partial|a|^2/\partial z = -\beta^{\text{TPA}}|a|^2$  [138],  $c$  is the speed of light,  $n$  is the cavity refractive index and  $V^{\text{TPA}}$  is the effective TPA volume [139]

$$V^{\text{TPA}} = \frac{\beta^{\text{TPA}}}{n^2} \frac{(\int n(\mathbf{r})^2 |\mathbf{U}(\mathbf{r})|^2 d\mathbf{r})^2}{\int n(\mathbf{r})^2 \beta^{\text{TPA}}(\mathbf{r}) |\mathbf{U}(\mathbf{r})|^4 d\mathbf{r}}, \quad (3.10)$$

where  $\mathbf{U}(\mathbf{r})$  is found by solving the Maxwell equations. Typical values for PhC cavities are given in [138], and for an MRR, it is approximately equal to its physical volume  $V$ . The effect of carrier diffusion is not negligible in PhCs [140], and an accurate model is given in [138]. However, since here we do not consider a specific resonator geometry, and an accurate estimation of TPA is not the goal of this work, a simple model outlined above is used.

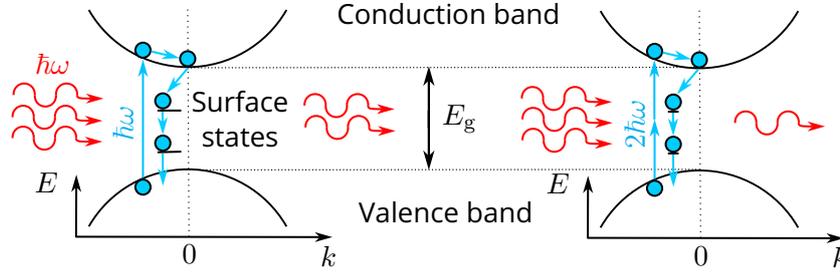


Figure 3.2: Linear absorption and two-photon absorption on a semiconductor band diagram. Left: photon energy  $\hbar\omega$  is higher than bandgap  $E_g$ . A photon (red) is absorbed by an electron (blue) in the valence band, moving it to the conduction band. Excess energy is dissipated due to collisions with the crystal lattice and other electrons. An electron can return to the valence band in various ways, but in nanostructures, surface recombination is dominant and shown in figure. The surface is a lattice defect, which creates states inside the band gap. An electron can fall into one of these, emitting a phonon. After a few surface states, the electron returns to the valence band. Right:  $\hbar\omega < E_g \leq 2\hbar\omega$ , a single photon is not enough to excite an electron, but two can be absorbed simultaneously.

The presence of free charge carriers induces a change in refractive index according to the Drude model [139]. As a result, a resonance shifts by  $\Delta\omega$  proportionally to the free electron density  $N$  [141, 142] with  $d\omega/dN$  as a proportionality coefficient. This process is FCD, included in Eq. (3.7) as:

$$f_{\text{NL}}(a) = i \frac{d\omega}{dN} N a. \quad (3.11)$$

Such free electrons could be generated via a linear photon absorption or the TPA. The former is possible in direct-bandgap semiconductors but causes high optical loss independent of the input power, unlike with TPA, where it linearly increases with the optical energy stored in a cavity. Considerable losses in systems without gain make large networks unrealistic. We therefore exclude the

linear photon absorption as a sustainable effect producing nonlinearity and only consider TPA for free electron generation. That said, hybrid photonic integration allows to combine various materials, such as III-V and silicon. Thus, the former can be used for a localized strong nonlinearity related to direct absorption and the latter for confinement and low-loss propagation. An example is based on III-V and Si PhC for all-optical signal processing [27].

The density of TPA-generated electrons is computed by [139]

$$\frac{dN}{dt} = -\gamma^c N + \frac{\gamma^{\text{TPA}}(a)}{2\hbar\omega V^c} |a|^2, \quad (3.12)$$

where  $\gamma^c$  is a free electron recombination rate and  $V^c$  is a volume across which electrons spread through a diffusion, which, for simplicity, is assumed to be equal to a cavity volume [138].

A strong electric field can deform electron orbits of atoms or molecules, which leads to a change in refractive index; this is the Kerr effect, which, similarly to FCD, causes a resonance frequency shift in response to a stronger electric field inside a cavity [143]

$$f_{\text{NL}}(a) = i\Delta\omega_{\text{NL}}a = -i\frac{\omega}{n}\Delta na = -i\frac{\omega}{n^2}\frac{n_2c}{V^{\text{Kerr}}} |a|^2a, \quad (3.13)$$

where  $n_2$  is the Kerr coefficient and  $V^{\text{Kerr}}$  is the effective Kerr volume equal to  $V^{\text{TPA}}$  [139]. Usually, FCD dominates in semiconductors over the Kerr effect unless the charge carrier lifetime is very short.

Nonradiative recombination of charge carriers, among other effects, leads to an increase in the temperature of a resonator, which changes its resonance frequency due to the thermo-optic effect, which can be significant. However, we omit them for the following reasons:

- spatial proximity of resonators due to the evanescent coupling will lead to strong thermal crosstalk; therefore, all resonances are expected to shift almost equally,
- the timescale of thermal effects is on the order of a microsecond, much larger than that of the input; the input itself is not expected to change on average significantly; therefore, the frequency shift due to the thermal effects would appear almost constant,
- the thermal effects analysis is complicated and depends on the resonator geometry and the material, whereas we aim for a general analysis.

### 3.3 . Dynamical model of coupled nonlinear resonators

As demonstrated in perceptron's case, a single neuron cannot perform complex computing alone. This section discusses how cavities interact with waveguides and other cavities to make a network.

#### 3.3.1 . Cavity-waveguide coupling

In order to inject or extract the light into the cavity, a waveguide is positioned next to it. Due to an overlap of waveguide and cavity modes, mode coupling occurs. With the derivation given in Section A, the equation for  $a(t)$  becomes

$$\frac{da}{dt} = \left( i\omega - \frac{\gamma^o}{2} - \frac{\gamma^\kappa}{2} \right) a + \kappa s(t), \quad (3.14)$$

where  $\kappa$  is the coupling strength,  $s(t)$  is an electric field travelling along the waveguide towards the cavity and  $\gamma^\kappa$  is the rate at which the energy stored in a cavity escapes into the waveguide. It can be helpful to assume that  $s(t)$  is a complex amplitude with a carrier with a frequency  $\omega_{\text{in}}$  and write  $s(t) \exp(i\omega_{\text{in}}t)$  instead. This becomes relevant later in Section 3.5.1.

Here,  $\gamma^\kappa$  is proportional to  $|\kappa|^2$  with the proportionality factor depending on the cavity type. For example, an MRR localizes travelling modes that circulate in one direction, and each direction is an independent mode. Then,  $\gamma^\kappa = |\kappa|^2/2$  since light can escape only in one waveguide direction. On the other hand, a PhC localizes standing wave modes, which can be described as a superposition of waves travelling in opposite directions. Then light can escape in both directions, giving  $\gamma^\kappa = |\kappa|^2$ .

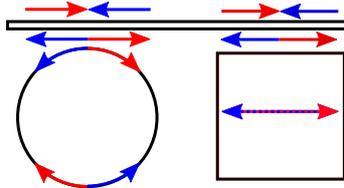


Figure 3.3: Coupling of microring (left) and photonic crystal (right) resonator modes to a waveguide. Mode directions are shown as colors. The photonic crystal mode is the standing-wave type and is a superposition of back and forward-propagating modes.

### 3.3.2 . Evanescent intercavity coupling

Two cavities can be placed close to each other to induce an interaction. Since a typical resonator does not localize modes inside it perfectly, the modes of two cavities can overlap, which induces evanescent coupling. If two lossless cavities with aligned resonances are evanescently coupled, their dynamics are described by [137, Section 7.5]:

$$\begin{cases} \dot{a}_1 = \mu a_2 \\ \dot{a}_2 = -\mu^* a_1 \end{cases} \quad (3.15)$$

where  $\mu$  is the evanescent coupling strength. The extrapolation to the case of multiple cavities is straightforward.

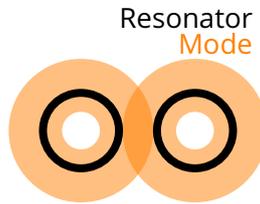


Figure 3.4: Cavity mode overlap induces an interaction between modes.

The coupling strength  $|\mu|$  can be obtained via a numerical solution of Maxwell equations or an experimental measurement. One could derive from Eq. (3.15) that if two uncoupled cavities had resonances at  $\omega$ , after being evanescently coupled, they become  $\omega \pm |\mu|$ . Then,  $|\mu|$  is found via splitting of resonant frequencies.

### 3.3.3 . Waveguide-mediated intercavity coupling

Light can change after passing through a waveguide next to a cavity (see Section A). If another cavity is on the way, it can also couple into it. As a result, the first cavity influences the dynamics of the second, which is coupling. In Section B, it is shown that, in a simplified manner and assuming proper mode direction matching:

$$\begin{cases} \dot{a}_1 = -\gamma^\kappa a_1/2 + |\kappa|^2 \exp(i\varphi_1) a_2 \\ \dot{a}_2 = -\gamma^\kappa a_2/2 + |\kappa|^2 \exp(i\varphi_2) a_1 \end{cases} \quad (3.16)$$

### 3.3.4 . Comparing waveguide-mediated coupling with evanescent

Both coupling methods are described with similar equations, but important differences exist.

Let us look at Eq. (3.15) and Eq. (3.16) from the eigenvalue perspective. For the evanescent coupling, the equation matrix is:

$$\begin{bmatrix} 0 & \mu \\ -\mu^* & 0 \end{bmatrix} \quad (3.17)$$

whose eigenvalues are  $\lambda = \pm i\mu$ . Therefore, such coupling always splits resonances and induces no losses. For the waveguide coupling:

$$\begin{bmatrix} -\gamma^\kappa/2 & |\kappa|^2 \exp(i\varphi_1) \\ |\kappa|^2 \exp(i\varphi_2) & -\gamma^\kappa/2 \end{bmatrix}. \quad (3.18)$$

Then its eigenvalues are  $\lambda = -\gamma^\kappa/2 \pm |\kappa|^2 \exp(i\varphi)$ , where  $\varphi$  includes all phases. Therefore, depending on effectively random  $\varphi$ , the eigenvalues' imaginary parts might become smaller than real parts, and coupling is closer to the evanescent coupling or more real, with almost no interaction (see Figure 3.5).

In other words, the scheme with waveguide-mediated coupling introduces a critical dependence on the phase, which is difficult to control. While it could be possible to optimize phases and perhaps improve RC's performance, it would require active control and related stabilization electronics, not to mention an increased chip footprint. Evanescent coupling, however, is resistant to the disorder of coupling strength (see Section 3.9). Therefore, the evanescent coupling is chosen as the primary source of neuron interaction.

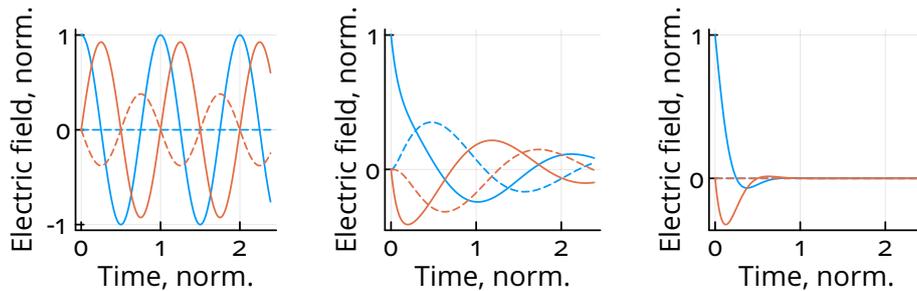


Figure 3.5: Complex mode envelopes of two photonic crystal cavities (blue, orange) coupled evanescently (left) and by a waveguide (center, right). Solid and dashed show real and imaginary parts. Center and right demonstrate a 'good' and a 'bad' phase conditions. Time is normalized by  $2\pi/\mu$  (left) and  $2\pi/\kappa^2$  (center, right).

### 3.4 . Reservoir computer master equation

The backbone of our photonic RC is an  $N_{\parallel} \times N_{\perp}$  square grid of evanescently coupled nonlinear cavities (see Figure 3.6(a)). One side of the grid is coupled to the input waveguide according to cavity-waveguide coupling coefficient  $\kappa$ . Through this waveguide, an optical input  $u(t)$  is injected. Assuming that intra-cavity and cavity-waveguide coupling rates are much smaller than the angular frequency, the coupled-mode theory can be applied [144]. For a purely linear case, the electric field of microcavity modes  $\mathbf{a}(t)$  can be described with

$$\frac{d\mathbf{a}}{dt} = \left( i\boldsymbol{\omega} - \frac{\boldsymbol{\Gamma}^o + \boldsymbol{\Gamma}^{\kappa}}{2} \right) \odot \mathbf{a} + (\hat{M}^{\mu} + \hat{M}^{\kappa})\mathbf{a} + \mathbf{K}^{\text{in}}s(t), \quad (3.19)$$

where  $\boldsymbol{\omega}$  are the frequencies of the normal modes (for simplicity, we assume one mode per cavity),  $\boldsymbol{\Gamma}^o$  is intrinsic optical loss,  $\boldsymbol{\Gamma}^{\kappa}$  is optical loss to waveguide,  $\odot$  is the Hadamard product,  $\hat{M}^{\mu}$ ,  $\hat{M}^{\kappa}$  are evanescently and waveguide-mediated intercavity coupling matrices,  $\mathbf{K}^{\text{in}}$  is an input waveguide-cavity coupling matrix (see Section B).

Here  $\hat{M}_{km}^{\mu} = -(\hat{M}_{mk}^{\mu})^*$  [137, Section 7.5] if the  $k$ -th and  $m$ -th cavities are evanescently coupled.

For simplicity, assume

- $|\hat{M}_{km}^{\mu}| = \mu$  if  $k$ -th and  $m$ -th cavities are coupled,
- $|K_k^{\text{in}}| = \kappa_{\text{in}}$  if a  $k$ -th cavity is coupled to the waveguide,
- $\Gamma_k^o = \gamma^o$ .

While this is a good approximation, they can vary to some degree, e.g. due to fabrication tolerances. The phase components of these vectors are defined by a chip geometry and are considered independent and identically distributed (i.i.d.).

In order to extract signals from cavities, they are coupled to independent waveguides, outputs of which are

$$\mathbf{z}(t) = \mathbf{K}^{\text{out}} \odot \mathbf{a}(t), \quad (3.20)$$

where  $\mathbf{K}^{\text{out}}$  is a vector of output waveguide coupling coefficients, for which uniformity is assumed according to  $K_k^{\text{out}} = \kappa^{\text{out}}$ . These waveguides introduce an additional optical loss, which has to be included in  $\boldsymbol{\Gamma}^{\kappa}$ .

In the end, a system receives an optical signal  $s(t)$  as an input and provides a set of optical signals  $\mathbf{z}(t)$ . Therefore, it can be treated as an RC. In the integrated photonic hardware, readout weights could be realized based on tunable attenuators with transmission  $w_k$ , phase-shifters according to  $\varphi_k$

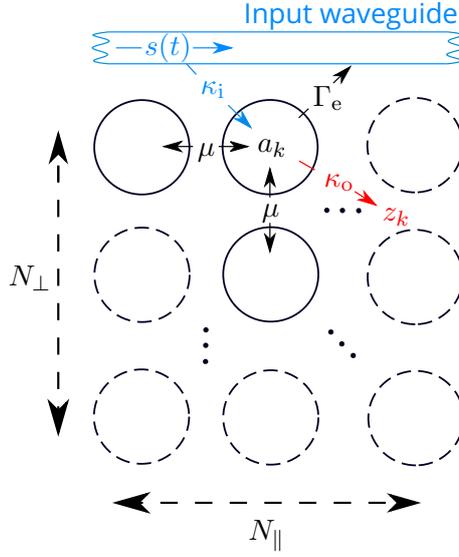


Figure 3.6: Geometry of the proposed integrated photonic reservoir computer.

and a combiner tree to implement a linear combination of  $z_k(t)$  that creates the output optical signal  $y(t)$ :

$$y(t) = \sum_k w_k \exp(i\varphi_k) z_k(t) = \hat{W}^{\text{out}} \mathbf{z}(t). \quad (3.21)$$

This way, the conversion of optical input  $s(t)$  to optical output  $y(t)$  is done fully optically, as electronics only control attenuators and phase shifters; both are constant during a computation.

Such a reservoir has many parameters, but how do we choose them? The next sections analyse RC's generic performance metrics to answer this question.

### 3.5 . Numerical simulation

Since RCs are nonlinear dynamical systems with very complex behaviour, their numerical integration and subsequent analysis of the results is the only tractable approach.

### 3.5.1 . Rotating wave approximation

If  $|da/dt| \ll \omega|a|$ , then the amplitude  $a$  obeys the rotating wave approximation. Then, one can formally introduce  $\tilde{a} = a \exp(i\omega_{\text{REF}}t)$  and Eq. (3.19) becomes

$$\frac{d\tilde{a}}{dt} = \left( i\Delta\omega - \frac{\Gamma^o + \Gamma^\kappa}{2} \right) \odot \tilde{a} + (\hat{M}^\mu + \hat{M}^\kappa)\tilde{a} + \mathbf{K}^{\text{in}}s(t), \quad (3.22)$$

where  $\Delta\omega_i = \omega_i - \omega_{\text{REF}}$  and  $\omega_{\text{REF}}$  can be chosen such that  $\max_i |\Delta\omega_i|$  is minimized. For example, the carrier frequency  $\omega_{\text{in}}$  of the input signal  $s(t)$  is a good option.

Considering that a typical  $\omega$  is on the order of  $2\pi \cdot 200$  THz, whereas  $\omega - \omega_{\text{REF}}$  is comparable to a hundred gigahertz, the numerical integration timestep can be significantly reduced.

Such substitution does not change the nonlinear equations since they are proportional to  $|a|$ , which is identical to  $|\tilde{a}|$ . For that reason, for simplicity, the rotating wave approximation is isolated to numerical integration; the formalism in Eq. (3.19) is used otherwise.

### 3.5.2 . Implementation

The differential equations are integrated using the *DifferentialEquations.jl* library [145] of the *Julia* programming language [146]. Default model parameters are given in Table 3.1.

Table 3.1: Default parameters of a linear reservoir with Eq. (3.12). Others are omitted as they only impact the input power.

Parameter	Value	Units	Comments
Geometry	8×3 grid		GaAs MRRs
$\mu$	25	GHz	defines RBW
$\gamma^c$	28.5	GHz	[147]
Q-factor	$3 \times 10^5$		near-optimal $\gamma^s/2\pi \approx \text{RBW}/N \approx 0.9$ GHz
$\kappa_{\text{in}}^2$	$4\pi$	GHz	every other cavity next to input waveguide is coupled
$\kappa_{\text{out}}^2$	$\pi$	GHz	provides sufficiently strong $z_k(t)$

For integration, the 3-step Adams-Bashforth explicit method was used (*AB3* in *DifferentialEquations.jl*) with a typical timestep  $0.15\tau$  where  $\tau$  is the RC timescale, i.e. the minimum of inverse eigenfrequencies, inverse eigenlosses and electron lifetime. Integration convergence verification is provided in Figure 3.7.

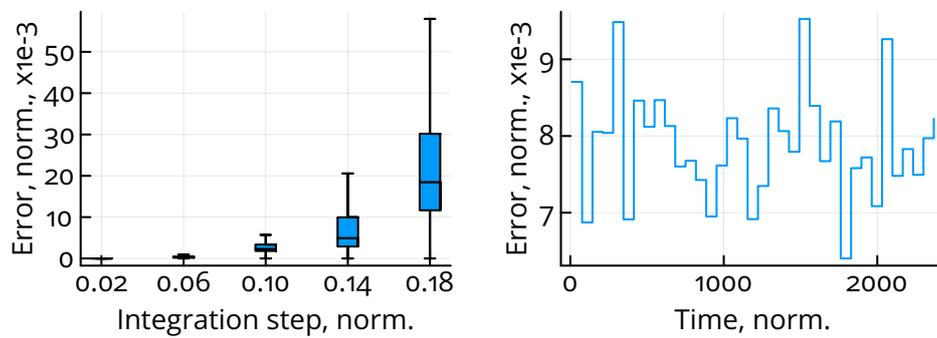


Figure 3.7: Integration convergence validation of a reservoir with default parameters (see Table 3.1). Left: integration step normalized by the reservoir timescale. A set of identical reservoirs with identical random input is integrated with various time steps. The result with the smallest step is assumed to be accurate and serves as the baseline. Integration error is computed for a trajectory of each cavity w.r.t. the corresponding cavity of the baseline, and the error is normalized by its mean amplitude over the integration period. Each box represents the largest error across all cavities at each integration step. Right: each step shows an average error across all cavities during this step. The integration step is the largest present in the left figure. Time is normalized by the inverse reservoir bandwidth. The error does not increase even after a long computation.

### 3.6 . Dimensionality in the linear regime

First, consider the case when the input signal is not strong enough to induce a nonlinear response by the microcavities. An important value for the following discussion is the input signal's bandwidth  $BW$ , which is defined as the span between its spectral half-power positions (see Figure 3.8).

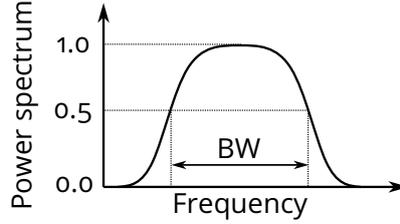


Figure 3.8: Visual representation of signal bandwidth  $BW$ .

As will be shown, the computational properties of such photonic reservoirs depend primarily on the microcavities' spectral characteristics. In that regard, an important consequence when evanescently coupling microcavities is the formation of supermodes with split resonances (see Figure 3.10).

#### 3.6.1 . Methodology

To determine the dimensionality, a random sequence is generated

$$\tilde{s}_0(t_n) = \sqrt{I_0}(t_n) \exp(i\varphi_0(t_n)), \quad (3.23)$$

where  $\sqrt{I_0}(t_n)$  and  $\varphi_0(t_n)$  are randomly sampled according to a white noise distribution. Then, an 8th-order Butterworth low-pass filter with a given bandwidth  $BW$  is applied on  $\tilde{s}_0(t_n)$  to obtain  $s_0(t_n)$ , which is then linearly interpolated to  $s_0(t)$ . This signal is used to modulate an optical carrier that is injected into the reservoir as an input. The photonic reservoir's dimensionality is then determined by using the PCA (see Section 2.4.1) on signals from cavities  $z_n$  defined in Eq. (3.20). Importantly, for the PCA electric field is split into a real and an imaginary part, which implies that the maximum dimensionality our reservoir can have is  $2N$ . Therefore, it is convenient to introduce a normalized dimensionality  $\tilde{D} = D/2N$ .

As mentioned in the PCA description, we use the principal component cut-off of 1%. While a higher/lower value would decrease/increase the apparent dimensionality, to the best of my knowledge, there is no universal rule to its choice. Still, the goal is that the dimensionality represents the dynamics of the reservoir. In our case, they could be altered either by noise or numerical integration errors. However, the former is not present in the model, and the latter is smaller than 1% (see Section 3.5.2). Here, any reasonable cut-off satisfying these constraints would be valid (see Figure 3.9), but it is important to be consistent with the choice.

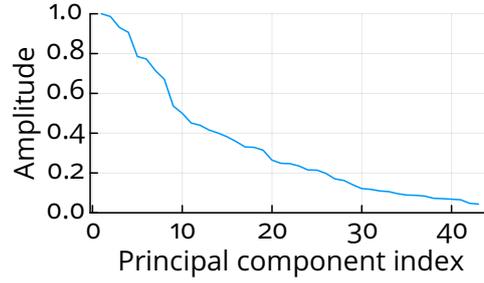


Figure 3.9: Amplitudes of principal components reservoir designed according to the following sections. Here, the drop-off is rather smooth and a change of the amplitude cut-off would not change the resulting dimensionality drastically.

### 3.6.2 . Spectral line spread of coupled resonators

Consider the linear part of Eq. (3.19):

$$\text{diag} \left( i\omega - \frac{\Gamma^o + \Gamma^\kappa}{2} \right) + \hat{M}^\mu + \hat{M}^\kappa, \quad (3.24)$$

eigenvalues of which are  $\lambda_k^E$ . Then, the average supermode bandwidth is

$$\gamma^S = \langle -2\text{Re}(\lambda_k^E) \rangle_k. \quad (3.25)$$

Importantly, the coupling-induced supermode splitting increases the reservoir's response bandwidth beyond the bandwidth of a single resonator. The reservoir bandwidth RBW is defined as the maximum distance between the network's supermodes measured at half intensity. For example, without the input waveguide, two identical independent cavities with resonance frequencies at  $\omega_0$  and coupled with rate  $\mu$  become split  $\omega_0 \pm |\mu|$ , leading to  $\text{RBW} \approx (2|\mu| + \gamma^S)/2\pi$ . For multiple identical cavities in a chain, Eq. (3.24) is a tridiagonal Toeplitz matrix, the eigenvalues of which are given by [148]

$$\lambda_k^E = \left( i\omega_0 + \frac{\gamma^o}{2} \right) + 2\sqrt{-|\mu|^2} \cos \left( \frac{k\pi}{N+1} \right), \quad k = 1 \dots N, \quad (3.26)$$

meaning that  $\text{RBW} \approx (4|\mu| + \gamma^S)/2\pi$ . For nontrivial connectivity architectures, an analytical derivation of RBW is challenging. However, in numerical simulations, it was found that supermode resonances stay within a  $\omega_0 \pm \pi|\mu|$  interval (see Figure 3.10). Therefore, it is assumed that

$$\text{RBW} \approx (2\pi|\mu| + \gamma^S)/2\pi \quad (3.27)$$

for such an evanescently coupled microcavity system. Furthermore, as will be shown later, in the cases we are interested in  $\gamma^S \ll |\mu|$ , and hence  $\text{RBW} \approx |\mu|$ .

At this stage, it is beneficial to consider the supermode concept in more detail. A supermode is an independent, i.e. an orthogonal oscillation of the

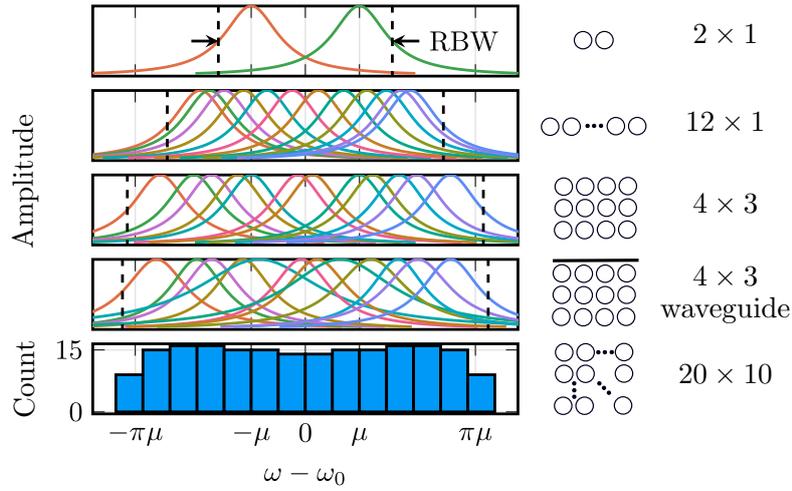


Figure 3.10: Supermodes of evanescently coupled microcavities for various grid geometries. Bottom figure shows a histogram of supermode frequencies.

coupled system's electric fields. Each supermode can be considered a virtual microcavity that performs a pass-band filtering at its resonant frequency and bandwidth. Crucially, the coupling-induced splitting of supermode resonances makes their frequencies vary and, as we will see later, promising high dimensionality of an RC.

Considering the linear part of the system, 'supermodes' stand for diagonalization of the operator, with no particular implication. Yet supermodes, which extend over many resonators and are mutually overlapped, are coupled by the local nonlinear interaction considered here. In other words, the reservoir can be considered a system of nonlinearly coupled oscillators.

### 3.6.3 . Bandwidth matching

A supermode needs to receive input. For that, consider the impact of the injection BW on the reservoir's dimensionality. In Figure 3.11 one can see that, when  $BW \ll RBW$ , the dimensionality is low. In this case, many supermodes are off-resonance relative to the injected field and cannot be excited. With an increasing BW, more and more supermodes can interact with the injected information. Consequently, the reservoir's dimensionality increases until  $BW = RBW$ , when all supermodes become excited. Accordingly, the physics of supermodes in coupled arrays result in a necessary bandwidth matching condition for maximizing the system's dimensionality:

$$BW \geq RBW. \quad (3.28)$$

Importantly, here, the network was driven with white noise, i.e. by a signal with a flat spectrum. However, in realistic and application-relevant situations,

such a signal is unlikely. Certain supermodes will be excited less than others and will contribute less to the network's dimensionality.

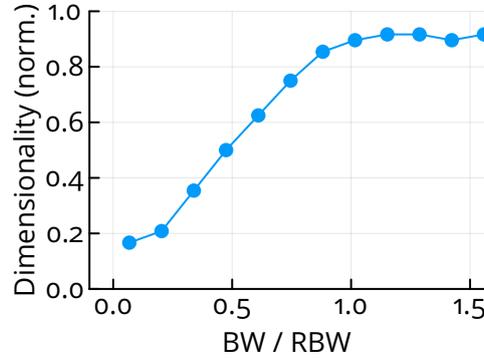


Figure 3.11: Dimensionality of linear photonic reservoir w.r.t. input signal bandwidth. Reservoir parameters are given in Table 3.1 but  $\mu$  is increased to 80 GHz to increase the upper limit of dimensionality.

### 3.6.4 . Spectral line separation

The microcavity network's dimensionality relies upon individual supermodes being driven by different aspects of an input signal, which requires supermodes to be sufficiently separated. Therefore, to quantify this metric, consider an average supermode spacing  $\text{RBW}/N$ , normalized by the average supermode bandwidth

$$F_{\Gamma} = \frac{\text{RBW}/N}{\gamma^S/2\pi}, \quad (3.29)$$

which can be considered a measure of spectral span available for individual supermodes. In Figure 3.12, one can see that the dimensionality is low for  $F_{\Gamma} \ll 1$ , and for  $F_{\Gamma} > 1$ , dimensionality is maximized. This threshold may change depending on system parameters; however, it was found that it is typically comparable to unity. Therefore, a second requirement for a high dimensionality is  $F_{\Gamma} \gtrsim 1$ . Again, this is not a sufficient condition since it does not ensure that all supermodes are separated.

### 3.6.5 . Effect of input waveguide coupling

Spacing between supermodes is typically inhomogeneous, and supermodes can potentially overlap, even if  $F_{\Gamma}$  is large. This effect can be mitigated through a stronger coupling to the input waveguide  $\kappa_{\text{in}}$ , which can shift resonances or increase the bandwidth of some supermodes, as can be seen in Figure 3.10 for a  $4 \times 3$  reservoir with and without the input waveguide. Figure 3.13 shows that this approach can be effective when  $N_{\perp}$  is low, i.e. when overall coupling to the input waveguide is relatively strong for the supermode split-

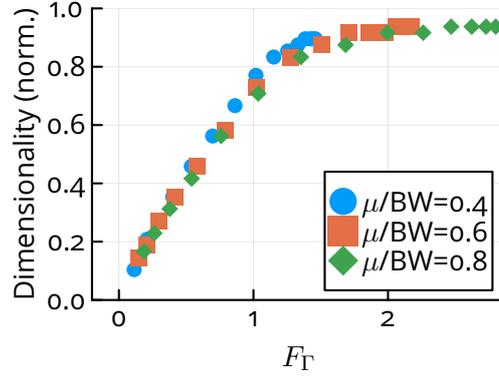


Figure 3.12: Dimensionality of linear photonic reservoir w.r.t. average supermode spacing. Reservoir parameters are given in Table 3.1 and  $BW > RBW$ . Each point corresponds to specific  $\mu$  and  $\gamma^o$ .

ting. That said, the waveguide-induced coupling may interfere when phase conditions are unfavourable (see Section 3.3.4).

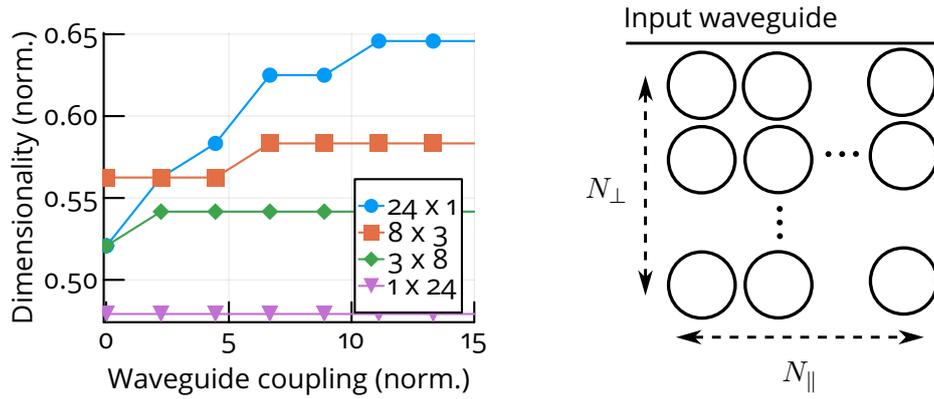


Figure 3.13: Left: impact of  $\kappa_{in}^2/\gamma^o$  to dimensionality. Reservoir parameters are given in Table 3.1 and  $BW > RBW$ , legend shows  $N_{\parallel} \times N_{\perp}$ , reservoir geometry. Right:  $N_{\parallel} \times N_{\perp}$  reservoir geometry.

### 3.6.6 . Scalability of dimensionality

From  $F_T$  a scaling behaviour of dimensionality can be determined. Since  $F_T \propto N^{-1}$ , a larger number of cavities leads to a lower  $F_T$ , which leads to a lower  $\tilde{D}$ . This, however, can be compensated by lowering optical losses, as  $F_T \propto (\gamma^S)^{-1}$ . Indeed, in Figure 3.14(a), one can see that with a high  $\mu/\gamma^o$  (which is analogous to  $F_T$ ), the dimensionality can reach the theoretical limit. Furthermore, depending on their geometric arrangement, dimensionality varies for a given number of cavities. A minor fluctuation of dimensionality is due to the intricate nature of supermode formation – supermode frequencies of a

24×1 grid are different from a 12×2 grid. However, there are also outliers with a considerably lower dimensionality. In these cases,  $N_{\perp}$  is high, which allows supermodes to localize spatially further from the input waveguide. As a result, these supermodes become too weakly coupled to the input and, as a consequence, contribute less to the system's dimensionality (see Figure 3.14(b)). In addition, a lower  $N_{\perp}$  allows for a higher dimensionality even at a low  $F_{\Gamma}$ , likely due to the impact of the input waveguide discussed above.

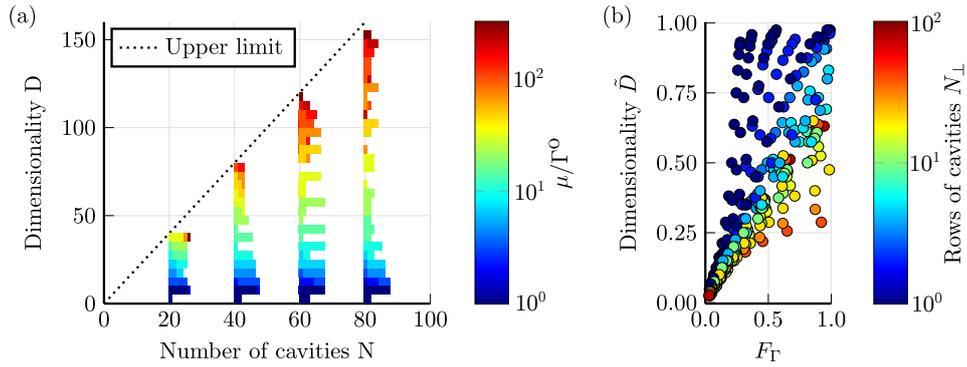


Figure 3.14: (a) Scalability of linear reservoir dimensionality w.r.t. optical losses. Dimensionalities of all possible combinations of  $N_{\parallel}$  and  $N_{\perp}$  for given  $N$  are shown as histograms. Reservoir parameters are shown in Table 3.1, with  $\mu = 80$  GHz, reduced  $\kappa_{\text{out}}$  and  $\text{BW} > \text{RBW}$ . (b) Impact of  $F_{\Gamma}$  and grid geometry on dimensionality. Both figures use the same dataset.

### 3.7 . Nonlinear regime

Now the RC is analysed at higher input power when a nonlinear response appears. Nonlinearity can result in a reservoir operating in a chaotic regime, which, for most purposes, is harmful to computation. Chaos corresponds to the situation where a network's state is sensitive to infinitesimal changes in the input. Hence, the unavoidable presence of noise in hardware would render computations not reproducible for even perfectly identical inputs. In order to characterise this effect, consistency analysis is carried out alongside the dimensionality test [8].

### 3.7.1 . Methodology

A set of inputs in the form  $s_n(t) = s_0(t) + \Delta s_n(t)$  is generated, where  $s_0(t)$  and  $\Delta s_n(t)$  are values sampled from a complex-valued white noise distribution, which is filtered with a Butterworth eighth-order pass-band filter. Here,  $s_0(t)$  is the ‘true input’ and is the same for all  $n$ , while  $\Delta s_n(t)$  emulates an impact of noise in an experiment and is different for each  $n$ . A signal-to-noise ratio

$$\text{SNR} = \frac{\int |s_0(t)|^2 dt}{\int |\Delta s_n(t)|^2 dt} \quad (3.30)$$

of 20 dB is attained by scaling  $\Delta s_n(t)$  amplitude. Here, the choice of 20 dB is semi-arbitrary: it is important that this noise is much weaker than  $s_0(t)$  but strong enough to push a RC away from the response to  $s_0(t)$ . To the best of my knowledge, there were no claims for the optimal deviation strength in the literature. Each  $s_n(t)$  is injected into a separate identical reservoir. The absolute value of correlation between trajectories of a  $k$ -th cavity of  $i$ -th and  $j$ -th reservoirs  $z_k^i(t)$  and  $z_k^j(t)$  is then computed [149]

$$\gamma_k^{ij} = \left| \frac{\text{E} [z_k^i (z_k^j)^*] - \text{E} [z_k^i] \text{E} [(z_k^j)^*]}{\sqrt{\text{E} [|z_k^i|^2]} \sqrt{\text{E} [|z_k^j|^2]}} \right|, \quad (3.31)$$

where  $\text{E}[A]$  is the expected value of  $A$ . The consistency for this pair of reservoirs is an RMS across all their cavities  $\gamma^{ij} = \sqrt{\langle (\gamma_k^{ij})^2 \rangle_k}$  and, equally, the consistency of system  $\gamma$  is the RMS of  $\gamma^{ij}$  of all combinations of  $i$  and  $j$ .

Even though this section started with the mention of chaotic reservoirs, it is important to note that a low consistency does not necessarily imply chaos. That said, it can lead to a reservoir being in unexpected states outside of a training dataset, invalidating currently trained weights and harming computing.

### 3.7.2 . Nonlinear response strength

How do we distinguish when a nonlinearity is ‘weak’ or ‘strong’? In order to draw conclusions, the nonlinear response strength needs to be related to other parameters.

Nonlinearity strength increases with stronger optical power inside a cavity, which, in turn, increases with stronger input. We define  $P_{\text{NL}}$  as an input power at which a nonlinear effect equals a related linear effect. Two-photon absorption increases losses inside the cavities according to  $\gamma^{\text{TPA}}$ , which would be natural to compare to linear optical loss. FCD and the Kerr effect cause resonance shift  $\Delta\omega$ , which, in turn, is natural to be compared to a cavity bandwidth, which is also defined by the linear optical loss. Therefore, an optical loss rate is the normalization factor determining the cavity array’s nonlinearity power threshold for all considered nonlinear effects.

## Single cavity

For a single cavity injected with a monochromatic wave, one can determine  $P_{\text{NL}}$  analytically (see supplementary material). The result is shown in Figure 3.15 for various integrated platforms for MRR and PhC cavities based on data obtained from the literature (see Table 3.2), and the horizontal axis shows an operating frequency that is  $\text{BW}/2$ .

How is there a bandwidth when a monochromatic signal is considered for analytical computation? It is assumed that when a non-monochromatic signal is passed to the cavity, it is coupled into the cavity almost entirely. Assume a cavity with a bandwidth  $\gamma^{\text{S}}$  and an electron relaxation rate  $\gamma^{\text{c}}$ . This cavity can receive most of the optical power of an injected signal if  $\text{BW} \leq \gamma^{\text{S}}/2\pi$ . However, a low  $\gamma^{\text{S}}$  corresponds to a higher loaded Q-factor and allowing for a stronger nonlinearity and, consequently, a lower  $P_{\text{NL}}$ . It is therefore assumed that  $\text{BW} \approx \gamma^{\text{S}}/2\pi$ .

Additionally, the scale of  $\text{BW}$  relative to  $\gamma^{\text{c}}$  is also important. When  $\text{BW} \ll \gamma^{\text{c}}$ , the generated free electrons recombine too fast, and FCD becomes negligible, whereas TPA becomes the dominant nonlinearity. Then,  $P_{\text{NL}}$  is computed w.r.t.  $\gamma^{\text{TPA}}$ , see the dashed line in Figure 3.15.

When  $\text{BW}$  is comparable to  $\gamma^{\text{c}}$ , FCD becomes stronger than TPA. Then,  $P_{\text{NL}}$  is computed w.r.t.  $\Delta\omega$ , and this case is shown as a solid line in Figure 3.15. Finally, when  $\text{BW} \gg \gamma^{\text{c}}$ , the electron density cannot react to the changes to the input signal in time, and FCD becomes irrelevant, see the dotted line in Figure 3.15.

Compared to MRRs, PhC cavities require less power and allow a faster operation on the same material, as a smaller mode volume corresponds to a stronger light confinement and a stronger nonlinearity, and a higher surface area enhances  $\gamma^{\text{c}}$  [150].

## Network of coupled cavities

For a network comprised of a large number of cavities, such analytical treatment is not immediately tractable. A numerical approach is therefore used instead.

Consider a reservoir with TPA as a nonlinearity. Since its supermodes are split due to the evanescent coupling and, as was found in the previous section,  $\Gamma^{\text{S}} \ll \text{RBW}$ , a monochromatic excitation of this nonlinearity is not suitable, as for a proper test all supermodes need to be excited, which is done with a signal with a finite bandwidth. Therefore, the input changes in time and modulates  $\gamma_k^{\text{TPA}}$  on a potentially individual resonator level. In order to then compare it to the scalar average supermode bandwidth  $\gamma^{\text{S}}$  (see Eq. (3.25)), its standard deviation is considered across many input samples  $\sigma_t[\gamma_k^{\text{TPA}}]$  so that we consider its fluctuations in time.

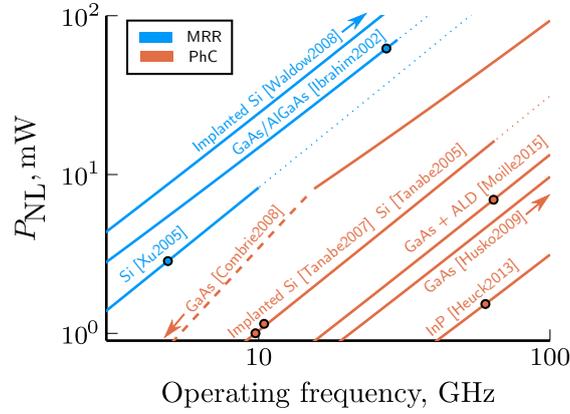


Figure 3.15: Comparison of material platforms w.r.t.  $P_{NL}$  and an operating frequency of a single MRR or PhC cavity. Solid and dashed lines show free carrier dispersion- (FCD) and two-photon absorption-dominated regimes, respectively. The dotted line continues the FCD line beyond the achieved electron recombination rate, details in text. Here, ALD is atomic layer deposition. For references, see Table 3.2.

Table 3.2: References used in Figure 3.15.

Key	Cavity	Reference
<b>GaAs</b>		
Ibrahim2002	MRR	[151]
Combric2008	$L_5$ PhC	[152]
Husko2009	$H_0$ PhC	[153]
Moille2015	$2 \times H_0$ PhC + ALD	[154]
<b>Silicon</b>		
Xu2005	MRR	[155]
Tanabe2005	$L_3$ and $L_4$ PhC	[150]
<b>Implanted silicon</b>		
Waldow2008	MRR	[156]
Tanabe2007	$H_1$ PhC	[157]
<b>InP</b>		
Ibrahim2003	MRR	[147]
Heuck2013	$H_0$ PhC	[158]

Why standard deviation when an average is a natural choice? Assume that a special signal is passed such that, depending on nonlinearity,  $\Gamma^{\text{TPA}}$  or  $\Delta\omega$  of each cavity changes in time, but that change is much smaller than the average value. One such example is given in Figure 3.16. In this case, the average of a nonlinear term could be included in the linear terms since it does not change in time, while the remainder is negligible. If an average is used to quantify the nonlinearity level, this case would be considered nonlinear when, in terms of the impact on the dynamics, it is not. A standard deviation has no such flaw.

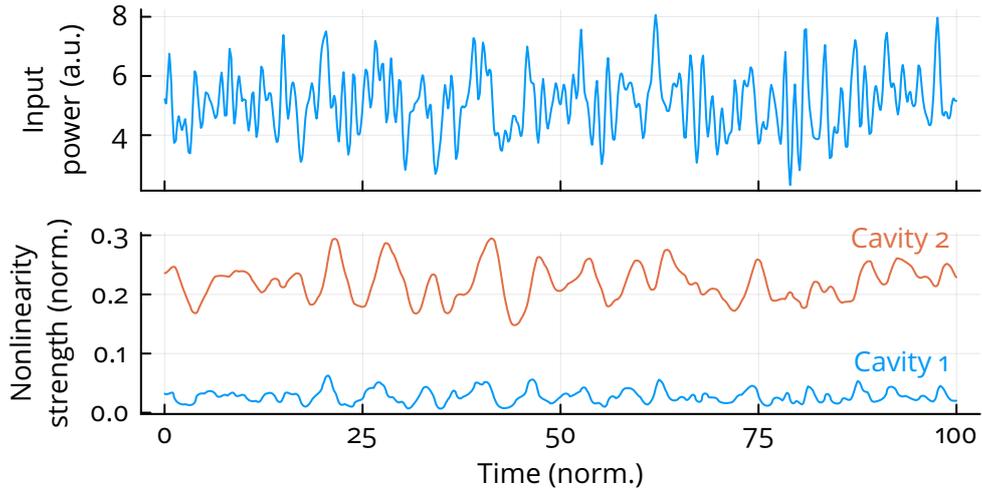


Figure 3.16: A case when nonlinearity is considerable, but its variation over time is not. Top: input signal, a sum of a constant signal with a weaker random component. Bottom: two-photon absorption rate of two coupled cavities normalized by average supermode bandwidth.

Then  $\langle \sigma_t[\gamma_k^{\text{TPA}}] \rangle_k$  is computed to obtain a scalar TPA rate for a given input power  $P$ . Repeating this process for a range of  $P$ , a largely monotonic curve  $P(\langle \sigma_t[\gamma_k^{\text{TPA}}] \rangle_k)$  is obtained, through which the input power necessary to induce a particular level of TPA-induced nonlinearity can be determined. In that case,  $P_{\text{NL}} = P(\gamma^{\text{S}})$ . For FCD and the Kerr effect, obtaining a scalar characteristic capturing the network's relevant response is identical, except that  $\Delta\omega_k$  is used instead of  $\gamma_k^{\text{TPA}}$ . Figure 3.17 shows that  $P_{\text{NL}}$  scales linearly with the number of cavities, provided  $N_{\perp}$  is limited. Indeed, for large  $N_{\perp}$ , some supermodes can localize far from the input waveguide and couple to input much weaker than those near it. This creates a nonlinearity imbalance in the reservoir, and more optical power is needed to reach  $\langle \sigma_t[\gamma_k^{\text{TPA}}] \rangle_k = \gamma^{\text{S}}$ .

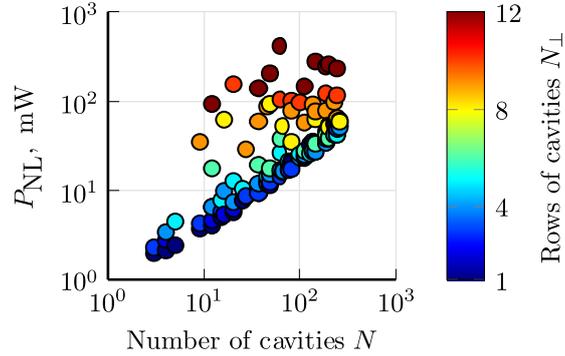


Figure 3.17: Scaling of power necessary to induce a nonlinear response in a reservoir w.r.t. the number of cavities. Reservoir parameters are given in Table 3.1. Signal bandwidth matches reservoir bandwidth.

### 3.7.3 . Results

Figure 3.18 shows the consistency and dimensionality of reservoirs with different nonlinearities as a function of input power. Noteworthy, it was found that a stronger TPA reduces the dimensionality while the consistency remains constant and close to unity. The drop in dimensionality results from reduced normalized supermode spacing  $F_T$ , as increasing TPA-induced losses increase the average supermode bandwidth (see Figure 3.19). A high consistency could be explained by the negative feedback nature of TPA, where the system's response to a stronger input is an increased damping that often stabilizes the system.

According to the data in Figure 3.18, FCD and the Kerr effect increase the dimensionality, but eventually, the system loses consistency for too high input power. Generally, consistency decreased when a nonlinearity reaches a related linear effect, in other words, when input power surpasses  $P_{NL}$ . The dimensionality increase is due to the nonlinear shift of individual supermodes, which otherwise might overlap in the linear regime. It was found that, generally, other system parameters have a relatively small impact compared to  $P_{NL}$ .

Interestingly, the nonlinear shift of supermode frequencies also compensates to some degree the negative impact on dimensionality caused by a mismatch between BW and RBW (see Figure 3.20). This could be explained by the fact that a cavity might belong to multiple supermodes. When a cavity resonance shifts, it causes multiple supermodes to shift as well, even those that are not excited in a linear regime. If the shift is strong enough, these supermodes can end up inside the input bandwidth and receive input (see Figure 3.21). Moreover, as BW reduces, a stronger nonlinearity is needed to

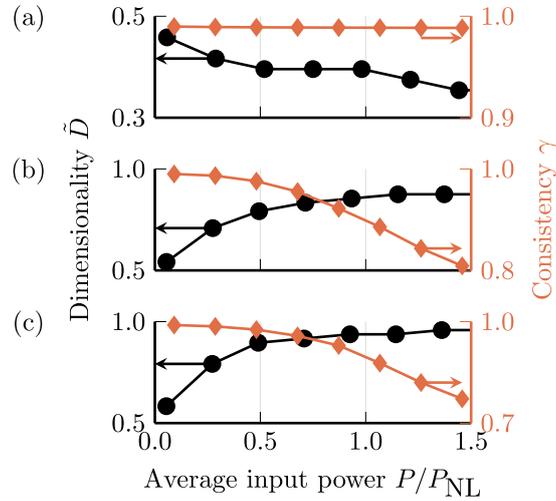


Figure 3.18: Effect of (a) two-photon absorption (b) same with free-carrier dispersion (c) the Kerr effect on reservoir dimensionality and consistency. Reservoir parameters are given in Table 3.1.

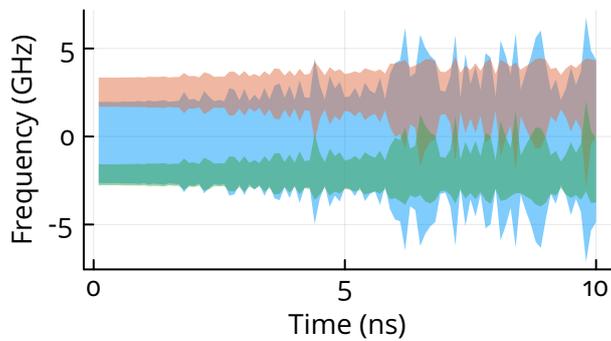


Figure 3.19: Dimensionality reduction caused by two-photon absorption. Spans show the immediate bandwidth of all supermodes, i.e. real part of the linearized Jacobian matrix, of a reservoir computer with three cavities around corresponding eigenfrequency. The reservoir is excited with a random input with a high enough bandwidth and increasingly higher power. Two-photon absorption causes supermodes to broaden, erasing their differences. For the first nanosecond, the Principal Component Analysis returns 5, but during the last nanosecond – 4.

move resonances into the input bandwidth, and at some point, the concept of supermodes stops being applicable.

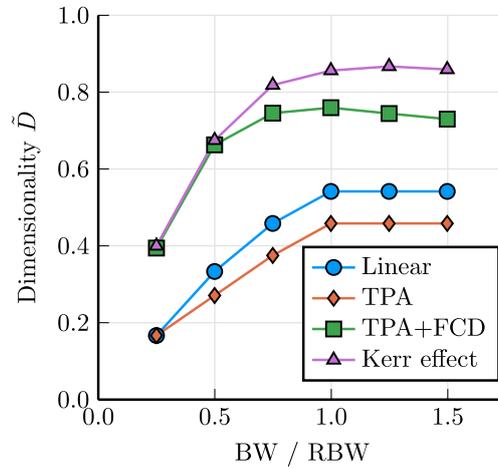


Figure 3.20: Effect of nonlinearities (shown in legend) on reservoir dimensionality at various input bandwidths. Each point corresponds to an input power at which consistency is 0.98.

To conclude, nonlinearity has a nontrivial effect on a reservoir. Importantly, this sensitively depends on the particular nonlinear effect that is leveraged.

- TPA: reduces dimensionality, keeps consistency,
- FCD: increases dimensionality , eventually reduces consistency,
- the Kerr effect: same as FCD.

However, in the case of FCD and Kerr nonlinearities, the dimensionality increase comes before the consistency loss. Hence, an interval of optimal input power allows for a better dimensionality with a high consistency.

### 3.8 . Discussion

So far, we have considered a system of coupled nonlinear cavities as an RC and discussed how its parameters determine its behaviour. However, our design differs from traditional ESNs in a few aspects and also makes a few bold assumptions. Can this RC and a readout be implemented in practice?

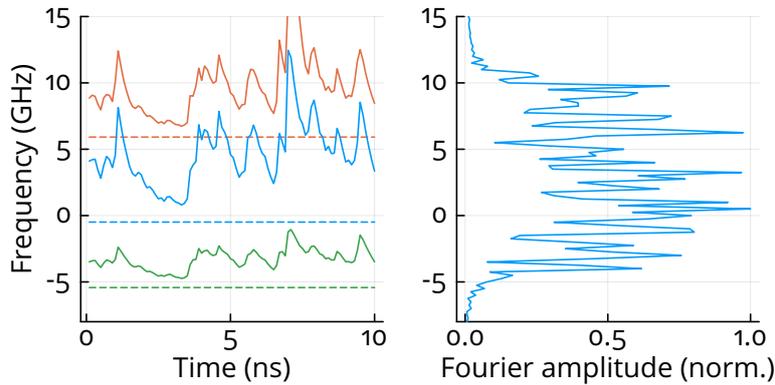


Figure 3.21: Dimensionality improvement caused by a resonance-shifting non-linearity. Left: time traces of linearized Jacobian matrix eigenfrequencies of linear (dashed) and nonlinear (solid) otherwise identical reservoirs with three cavities. Inputs are identical for both reservoirs – a random signal with a spectrum shown in the right figure. In the linear case, only blue and orange supermodes are excited, and the Principal Component Analysis returns a dimensionality of 4. With frequency shift, the green supermode can also accept input, and the dimensionality becomes 6, the maximum.

### 3.8.1 . Differences with Echo State Machines

One striking difference is that ESNs are discrete, whereas our RC is continuous. The discrete update equation can be mapped onto a continuous one by assuming an infinitely small timestep [117]. Then, an ESN is simply a result of the integration of a continuous RC with the Euler integration scheme.

Another difference is that an ESN connection matrix is typically random and sparse [47], and two neurons are unlikely to have a connection equally strong in both directions. In the photonic case, if two cavities are evanescently coupled, the connection is reciprocal, and coupling coefficients are negative complex conjugate of each other (see Figure 3.22). However, it appears that

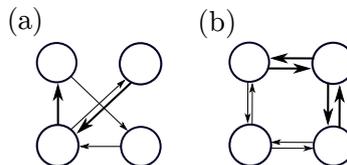


Figure 3.22: Neuron connectivity graph of (a) a typical Echo-State Network and (b) a photonic reservoir computer with a grid of cavities. Connection strength is represented by arrow thickness.

there is no optimal connection structure at all. Simple topologies with sparse connectivity behave similarly to more complex topologies, and only very limited topologies, such as ring topology, show limited performance in terms of

separation property, generalization property and memory capacity [159]. In the proposed RC, each cavity is coupled to 2-4 other cavities, which is sparse and corresponds to common guidelines [47].

In ESNs, a nonlinear function encompasses the input and the neuron connection terms (see Eq. (2.3)). In our reservoir, however, a nonlinearity depends only on the current reservoir state (see Eq. (3.19)). Nonlinearities themselves are different as well. Typically, in ESNs, a sigmoid or hyperbolic tangent is used, which adds a saturation of neuron activation, effectively, increasing loss. During a typical computation, ESN's linearized Jacobian eigenvalues largely move towards the z-plane origin (corresponding to an increase of loss) and go back with, perhaps, slight frequency deviations (see Figure 3.23, left). In the photonic case, the linearized Jacobian  $\hat{J}$  is found by

$$J_{km} = \text{diag} \left( i\omega - \frac{\Gamma^o + \Gamma^e}{2} \right)_{km} + \hat{M}_{km}^\mu + \hat{M}_{km}^\kappa + \left[ -\frac{1}{2} \frac{\beta^{\text{TPA}} c^2}{n^2 V^{\text{TPA}}} |a_k|^2 + i \frac{d\omega}{dN} N_k - i \frac{\omega n_2 c}{n^2 V^{\text{Kerr}}} |a_k|^2 \right] \delta_{km}, \quad (3.32)$$

where the last three terms correspond to TPA, FCD and the Kerr effect, with  $\delta_{km}$  as the Dirac delta. With TPA as the only nonlinearity, the behaviour is similar to an ESN, but with FCD and the Kerr effect, the trajectories of eigenvalues change considerably (see Figure 3.23, right). There, the two figures can be reconciled by considering the integration of a continuous RC with the Euler method. Then, the evolution of a supermode  $\tilde{a}$  with an eigenvalue  $\lambda$  is mapped to a discrete case like so

$$\tilde{a}(t) = \tilde{a}(0) \exp(\lambda t) \Rightarrow \tilde{a}(t + \Delta t) \approx \exp(\lambda \Delta t) \tilde{a}(t). \quad (3.33)$$

On the complex plane the exponent operator converts  $i\text{Im}(\lambda)z$  to a unit circle  $\exp(i\text{Im}(\lambda)\Delta t z)$  and a higher loss (here, negative  $\text{Re}(\lambda\Delta t)$ ) reduces its radius, which is seen on the left side of Figure 3.23.

### 3.8.2 . Integrated platforms and neuron density

The proposed system can be implemented in an integrated photonic platform, for instance silicon on insulator. The advantage would be a very compact geometry and relatively large nonlinearity, where three nonlinearities considered (TPA, Kerr and free carrier-related) will contribute. On this platform, MRR with a radius of 5  $\mu\text{m}$  could be implemented [161], allowing for a neuron density of  $10^4$  neurons/ $\text{mm}^2$ . Alternatively, silicon nitride could be used, offering much lower losses but requiring a larger footprint [108]. The nonlinear response will be pure Kerr and strong enough if the supermode lifetimes are adjusted to large values through coupling to input and outputs. Both platforms offer thermo-electric control for the readout, while integrated detectors are available with a silicon platform. Other technologies could be

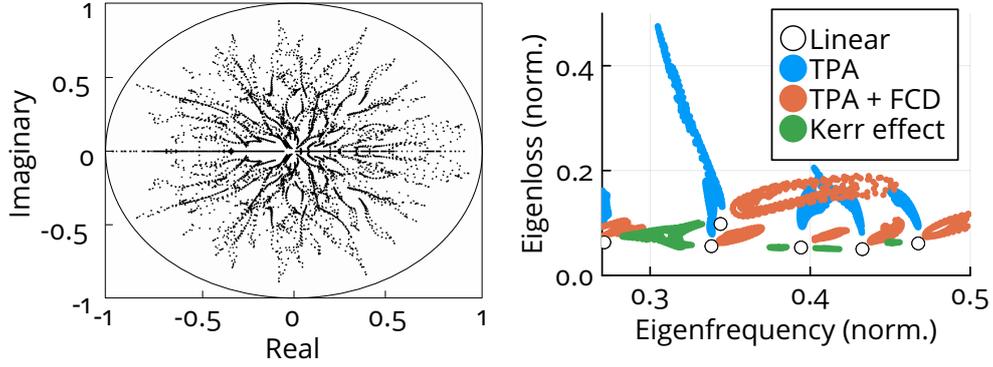


Figure 3.23: Left: linearized Jacobian eigenvalues of an Echo State Network during a typical computation, adapted from [160]. Right: same for the proposed reservoir computer with various nonlinearities (see Section 4.1 for details).

considered, particularly those based on the hybrid integration of materials such as III-V on silicon [27].

### 3.8.3 . Timescales

Section 3.6 proposed a relation between cavity parameters and their effect on reservoir dimensionality. Whether these requirements could be met with the available technology needs to be discussed. First, consider that  $\gamma^o < \gamma^s$ ,  $F_\Gamma \approx 1$  and  $\text{RBW} \leq \text{BW}$

$$N \frac{\gamma^o}{2\pi} \leq N \frac{\gamma^s}{2\pi} \approx N F_\Gamma \frac{\gamma^s}{2\pi} = \text{RBW} \leq \text{BW} \Rightarrow \gamma^o \lesssim \frac{2\pi}{N} \text{BW}. \quad (3.34)$$

Another requirement demands the reservoir to remember previous inputs: an optical timescale should be larger than an input signal timescale, i.e.

$$\frac{1}{\text{BW}} < \frac{2}{\gamma^s} < \frac{2}{\gamma^o} \Rightarrow \gamma^o < 2\text{BW}, \quad (3.35)$$

which is already satisfied by Eq. (3.34). For example, consider a reservoir with 20 cavities. To process a signal with  $\text{BW} = 20$  GHz, an approximate lower bound for  $\gamma^o$  would be  $2\pi$  GHz, which would correspond to an intrinsic Q-factor of  $\omega_0/\gamma^o \approx 0.2 \cdot 10^6$ , which is rather high but not impossible. An intrinsic Q-factor of  $0.7 \cdot 10^6$  was demonstrated in passive GaAs PhCs, with  $10^6$  considered achievable [152]. In an AlGaAs MRR, a Q-factor of  $1.5 \cdot 10^6$  was demonstrated [162], and in a silicon PhC a Q-factor of more than  $11 \cdot 10^6$  [163]. The low-loss silicon nitride allowed for an even higher Q-factor of  $37 \cdot 10^6$  in an MRR [164], but one should consider the difference in nonlinearity strength of different materials. Following the same logic, one could estimate the maximum number of cavities one could use in an RC, knowing the possible Q-factor of cavities.

Next, consider a nonlinearity timescale. TPA and the Kerr effect respond almost instantaneously to a change of electric field, similar to an ESN. However, FCD is also affected by the free electron recombination rate (see Figure 3.15), which should be comparable to the rate at which the optical signal in a cavity changes, i.e. half of its bandwidth. In our case, each cavity belongs to multiple supermodes, and thus, its bandwidth approaches the RBW, which is close to BW. The electron recombination rate of silicon MRRs was shown to be approximately 2 GHz [165], i.e.  $BW \lesssim 4$  GHz would be supported. The recombination can be accelerated to almost 20 GHz with a reverse-biased p-i-n junction embedded in a cavity [165]. Due to a higher surface area, in PhCs, the recombination can reach 10 GHz for silicon [166] and 100 GHz for GaAs [152].

### 3.8.4 . Readout

In a fully integrated system, cavities would be coupled to readout waveguides. This, however, can be challenging to realize for cavities inside the grid since the compact cavity arrangement leaves no space for such waveguides. One could consider using  $N_{\parallel} \times 1$  or  $2 \times N_{\perp}$  grids, where all cavities are accessible. An alternative could be using multiple photonic layers [167] or three-dimensional waveguides [130].

Apart from the internal layer of an RC, the readout's scalability also requires consideration. A common strategy for coherent signals is the use of Mach-Zehnder interferometers with two phase shifters, typically based on the thermo-optic effect. Such phase shifter requires a constant supply of electrical current, and its state-of-the-art chip footprint is  $109 \times 21 \mu\text{m}^2$  [132]. If we were to estimate the neuron density of an RC with a readout, it would be limited to 200 neurons/ $\text{mm}^2$ . There are a few alternatives; for example, MEMS phase shifters require less power by three orders of magnitude, although their footprint is larger –  $100 \times 45 \mu\text{m}^2$  [132]. Phase shifters based on plasmonic nonlinear polymers are even more compact, with the device length being  $29 \mu\text{m}$  [168]. The footprint of phase change materials is just a few square microns [169], which, combined with non-volatility, makes them an attractive option for a readout implementation, even with the limited switching resolution [136].

### 3.8.5 . Resonance disorder

Until this point, it was assumed that cavities' resonances are aligned  $\forall k = 1..N \omega_k = \omega_0$ . However, this might not be the case due to fabrication tolerances. For evanescently coupled cavities to interact, resonances should be sufficiently close. A weak intercavity interaction leads to two issues.

The first issue is the dimensionality reduction when some cavities are not coupled to the input waveguide directly (e.g. for  $N_{\perp} > 1$ ). Since such cavities can only receive signals through other cavities, they stop receiving signals without interaction and hence cannot contribute to dimensionality, as demonstrated in Figure 3.24 for  $8 \times 3$  and  $24 \times 1$  linear MRR reservoirs. As a resonance

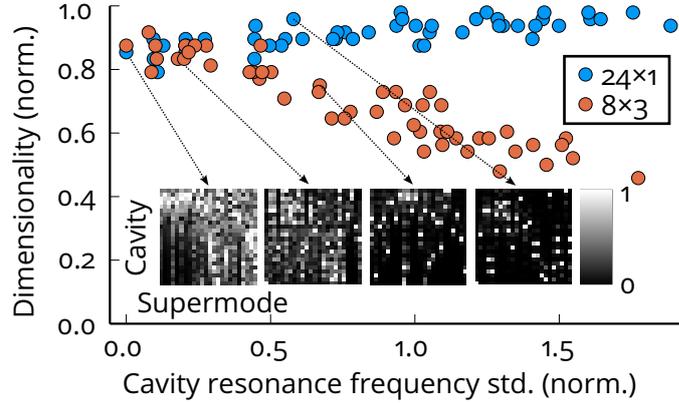


Figure 3.24: Impact of cavity resonance frequency disorder on dimensionality. Resonance disorder (in regular frequency units) is normalized to evanescent coupling strength. Here, std. is the standard deviation. Reservoir parameters given in Table 3.1. Input signal bandwidth is higher than reservoir bandwidth. Each dot corresponds to a  $N_{\parallel} \times N_{\perp}$  (shown in legend) linear microring resonator-based reservoir with resonances  $\omega_k \in \mathcal{N}(\omega_0, \sigma_{\omega})$  with various  $\sigma_{\omega}$ . Insets represent supermode intensity distribution over cavities  $\hat{\Lambda}$ .

disorder is increased, at first  $\tilde{D}$  increases as the supermode overlap is alleviated but decreases afterwards. The  $24 \times 1$  reservoir is affected less than the  $8 \times 3$  one since in the former, 12 cavities are coupled to the waveguide (due to MRR mode direction matching, see Figure 3.6(b)), but only 4 in the latter. If MRR mode direction was not considered and both MRR modes were excited, all cavities of a  $24 \times 1$  MRR reservoir are coupled to a waveguide. In that case, a further increase of resonance disorder increases  $F_T$ , and  $\tilde{D}$  will eventually reach unity. A similar outcome is expected for a PhC reservoir.

The second issue is a weakening of supermode mixing, i.e. delocalization. When cavities interact well, supermodes span multiple cavities; otherwise, they isolate in individual cavities. Because of that, a nonlinearity of an individual cavity can only affect a single supermode. As will be shown later in Figure 4.5, nonlinearity cannot contribute to computing performance as much in this case. Here,  $N_{\perp} > 1$  cases could be affected to a lesser degree than  $N_{\perp} = 1$  since for  $N_{\perp} > 1$ , each cavity has 2-4 neighbouring cavities, while only 1-2 for  $N_{\perp} = 1$ , and the probability of resonance mismatch with all neighbours is lower with more neighbours. The weakening of supermode mixing is shown in the inset of Figure 3.24. Each heatmap represents a matrix  $\hat{\Lambda}$  where  $\Lambda_{km}$  is the intensity of an  $m$ -th supermode in the  $k$ -th cavity. For clarity, each  $m$ -th column of  $\hat{\Lambda}$  is normalized to unity. One could see that when  $\sigma_k[\omega_k/2\pi] < 0.5\mu$ , supermodes span multiple cavities but only a few otherwise. The resonance disorder can be mitigated with the thermo-optic effect, but for more cavities,

it becomes challenging. In addition to that, a close proximity of cavities results in a strong thermal crosstalk. However, the resonance disorder can be less severe when cavities are spatially close [170].

### 3.9 . Evanescent coupling strength disorder

Throughout this chapter, we assumed that evanescent coupling strength is the same for each pair of coupled resonators. As in the previous section, this might not be the case due to fabrication tolerances. In Figure 3.25, we consider the case of normally distributed coupling strength. As a result, supermode frequencies can slightly shift around their mean value, but that is not very harmful as we have never assumed their exact position, and there was no qualitative change in the reservoir spectrum.

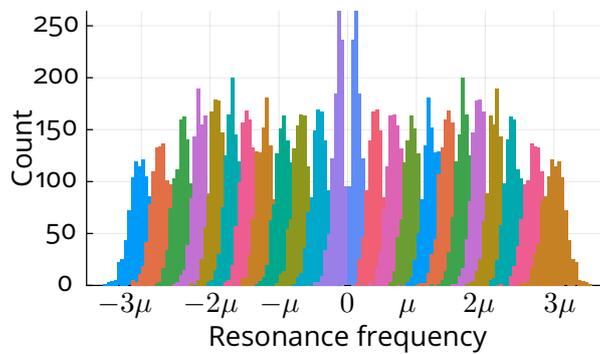


Figure 3.25: Histogram of reservoir supermode frequencies with evanescent coupling strength normally distributed around  $\mu$  with the standard deviation of  $0.2\mu$ .

### 3.10 . Prototype

A prototype template has been designed and fabricated at *imec* on the SOI platform. The available chip is  $5 \times 2.5$  mm<sup>2</sup> in size.

Optical I/O is performed with edge couplers that allows for a lower insertion loss compared to grating couplers. After an optical signal is injected, it is routed to a reservoir composed of an  $8 \times 3$  grid of cavities. A chip was comprised of two RCs, one with MRRs in the reservoir and the other with PhCs. Complex-valued weighting is performed with an amplitude controller made of an MRR and a phase shifter based on the thermo-optic effect. Each MRR in the amplitude controller was strongly coupled to its waveguides; its supermode bandwidth would be much larger than the expected RBW, and the filtering effect is expected to be negligible. A more robust solution would be an MZI, but we opted for a more compact, although risky, approach. Afterwards,

weighted signals were passed through a combiner tree composed of multi-mode interferometers (MMIs) to perform summation. The summation result is finally passed to a router, which allows to send signals to one of two optical outputs or two RF photodiodes.

The geometry of cavities was chosen to have their resonances in the C-band (1550 nm) based on the solution of the Maxwell equations with the FDTD method.

The simulation of MRRs was performed with an in-house FDTD solver by a fellow Ph.D. student, Luca Aimone Giggio. The finally selected microring geometry had a width of 200 nm and an outer radius of 5  $\mu\text{m}$ . The fabrication process determined the height; its expected value is 225 nm. The coupling between cavities was targeted to be about 40 GHz, corresponding to 330 nm of separation between MRRs. The coupling to an input waveguide was optimized by tuning its width (400 nm) and distance to an MRR (230 nm) (see Figure 3.26).

The simulation of the PhC was performed by the supervisor, Alfredo de Rossi. The PhC is based on a distributed Bragg reflector formed by alternating silicon and silicon oxide sections as proposed in [171] and is shown in Figure 3.27. Here, the width of sections is tapered while the period is kept constant. Due to geometry, evanescent coupling was only possible along the longitudinal PhC axis. Coupling between rows of PhC was implemented with a waveguide with a mirror on one side and a taper on the other to avoid resonant mode formation (see Figure 3.29). Cavity parameters were chosen based on FDTD simulation results shown in Figure 3.28.

The chip layout was prepared in *Synopsys OptoDesigner*.

Due to the compactness of the RC design, the main limitation was electrical I/O rather than chip space (see Figure 3.32). The accessible technology only allows the placement of contact pads at the edge of the chip. Then, electrical probes can supply current to contact pads. Alternatively, the chip can be packaged and wired to a printed circuit board (PCB) for more convenient I/O. Electrical I/O was used for:

- controlling readout, each signal requires an amplitude controller and a phase shifter, i.e. two contact pads excluding ground,
- controlling splitter at the end – three contact pads,
- tuning resonance of RC cavities by using tungsten heaters on the top of each cavity, i.e. one contact pad per cavity by default,
- (optional) extracting output signal with a photodiode that requires a pair of RF pads.

We therefore had to be careful about the I/O allocation.

First, we have limited the number of cavities in the readout for optical and electrical motivations. The optical motivation is that since we use  $2\times 1$  MMIs

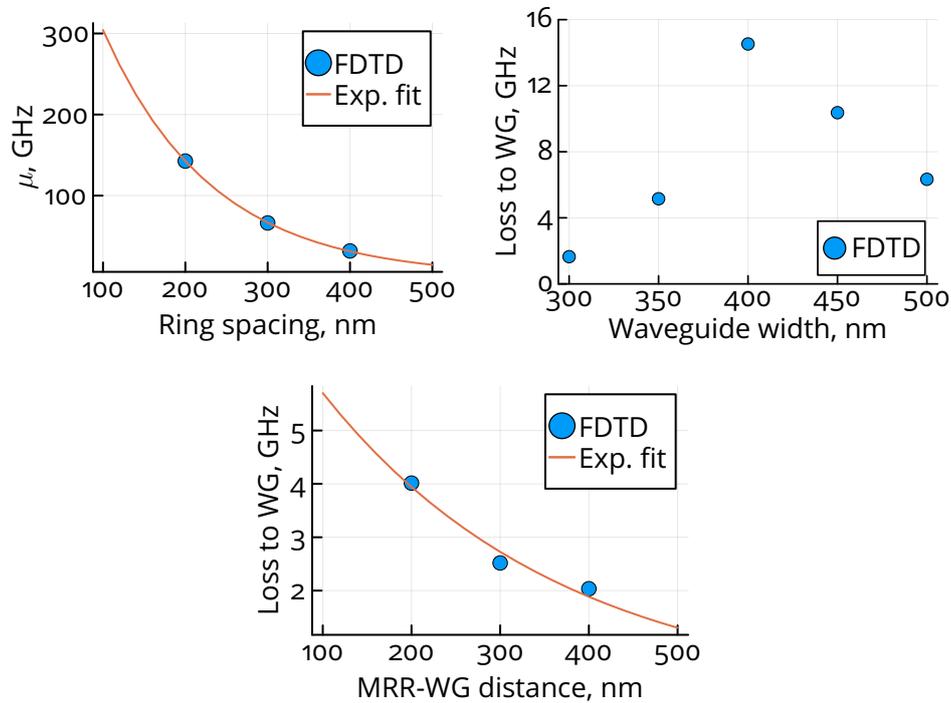


Figure 3.26: Finite-difference time-domain (FDTD) simulation results of a microring resonator (MRR). Top-left: evanescent coupling strength as a function of their distance. Exponential fit is used due to an exponential reduction of mode amplitude outside a resonator. Top-right: waveguide (WG) coupling strength as a function of its width; here distance to the waveguide is 200 nm, and a resonant behaviour can be seen. Bottom: same, w.r.t. its distance to a resonator. Here, the waveguide width is 550 nm and 'loss to WG' corresponds to  $\kappa^2/2$ .

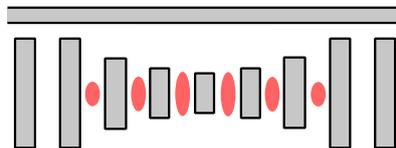


Figure 3.27: Design of used one-dimensional photonic crystal coupled to a waveguide. Gray areas show silicon region, white is silicon oxide and red shows the resonator mode's electromagnetic field distribution. Here, only nine sections of silicon are drawn for simplicity; the actual structure includes a few tens.

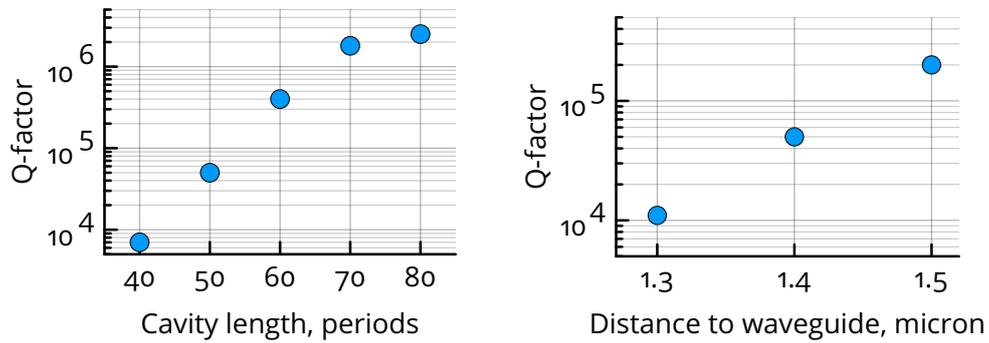


Figure 3.28: Simulation results of a photonic crystal resonator. Left: Q-factor as a function of the number of waveguide segments in a cavity. Right: loaded Q-factor as a function of the distance between the longitudinal resonator axis and the axis of the coupled waveguide, assuming its width is 450  $\mu\text{m}$ .

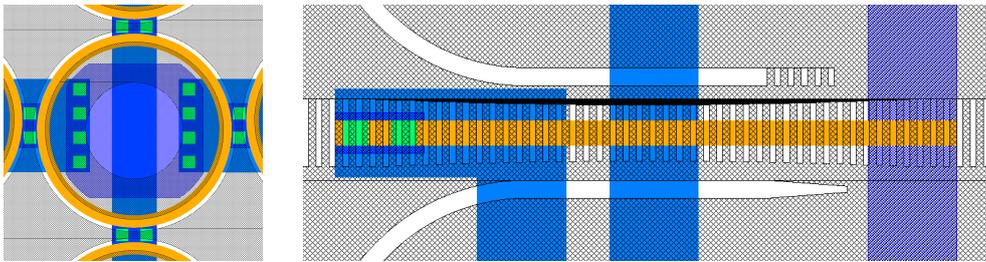


Figure 3.29: Layout of a microring (left) and a photonic crystal (right) resonators. Etch region is shown in black, blue and violet are two metallization layers; their interconnects are shown in green. Orange shows tungsten heaters (connected to the violet layer).

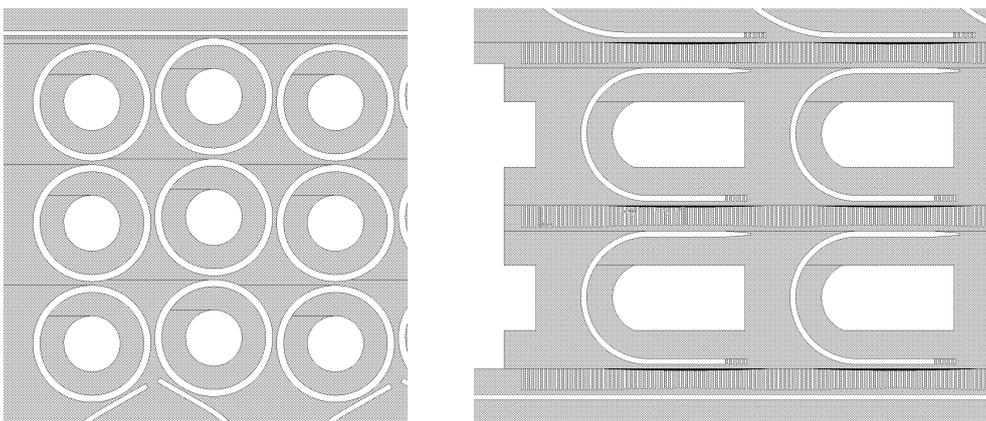


Figure 3.30: Layout of the reservoir computer based on microrings (left) and photonic crystal (right) resonators. Etch regions are shown in black. The microring-based reservoir has a staggered design (see Figure 3.31).

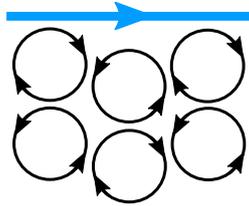


Figure 3.31: Staggered reservoir design with every second cavity in the first row coupled to the waveguide (blue, the arrow shows input signal direction) to excite only one of the counter-propagating modes, their direction shown as arrows. While using both modes can double the dimensionality, two waveguides per readout microring would be required to harness it, which was challenging under given constraints. Therefore, we opted for such a design so as not to waste input power to excite unused modes.

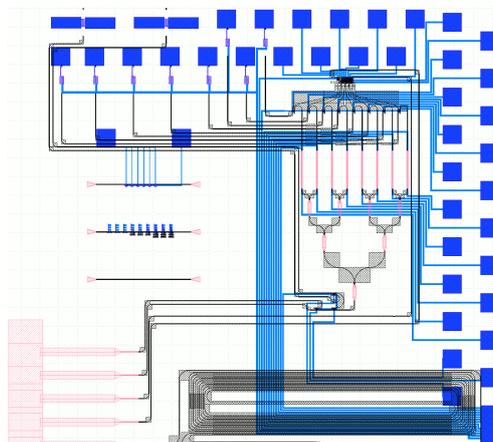


Figure 3.32: Complete reservoir computer layout; the spiral at the bottom is part of an unrelated design. Black shows etch regions, blue shows metallization layers and pink shows foundry-proposed phase shifters, MMIs and couplers. Most of the chip edge was taken by contact (square) pads.

for a combiner tree, a power of 2 readout signals would be convenient for a readout design – we, therefore, chose eight cavities. That leaves us with 16 electrical contact pads for readout.

Second, controlling each heater would require  $N_{\perp} \times N_{\parallel} = 24$  contact pads. Moreover, thermal crosstalk is expected due to the proximity of cavities. Therefore, such a fine-grained control would be wasteful as the longer-range temperature gradients would most likely wash it out. Instead, we opted for a wiring scheme where only columns and rows were independently controlled, which requires  $N_{\perp} + N_{\parallel} + 1$  contact pads instead (see Figure 3.33).

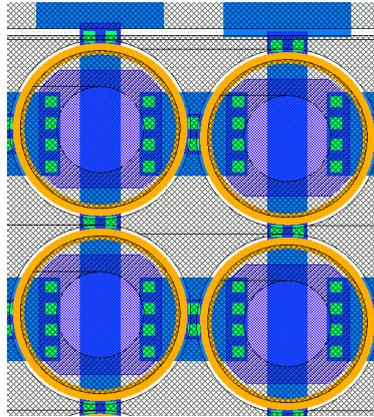


Figure 3.33: Heater wiring scheme of a microring resonator-based reservoir. Etch region is shown in black, blue and violet are two metallization layers with green showing their interconnects, orange shows tungsten heaters (connected to the violet layer).

Due to how optical I/O is implemented, we can only observe a single optical signal at a time. However, to utilize the ridge regression training algorithm, we must record the time traces of all resonators. Consider amplitude weights as  $|w|$ . First, we set  $|w|_k = \delta_{k1}$ , where  $\delta_{ij}$  is the Kronecker delta. Then, the first neuron signal is the only one that enters the rest of the readout circuit and passes through it unchanged. We then pass a training input signal and record the response of that neuron at the optical port. Then we repeat the process, but  $|w|_k = \delta_{k2}$  and the same training input signal. Repeating the procedure for all neurons connected to the readout circuit, we can record the entire observable reservoir state and train the reservoir with ridge regression.

The first batch of chips has been received, and their characterization is ongoing.

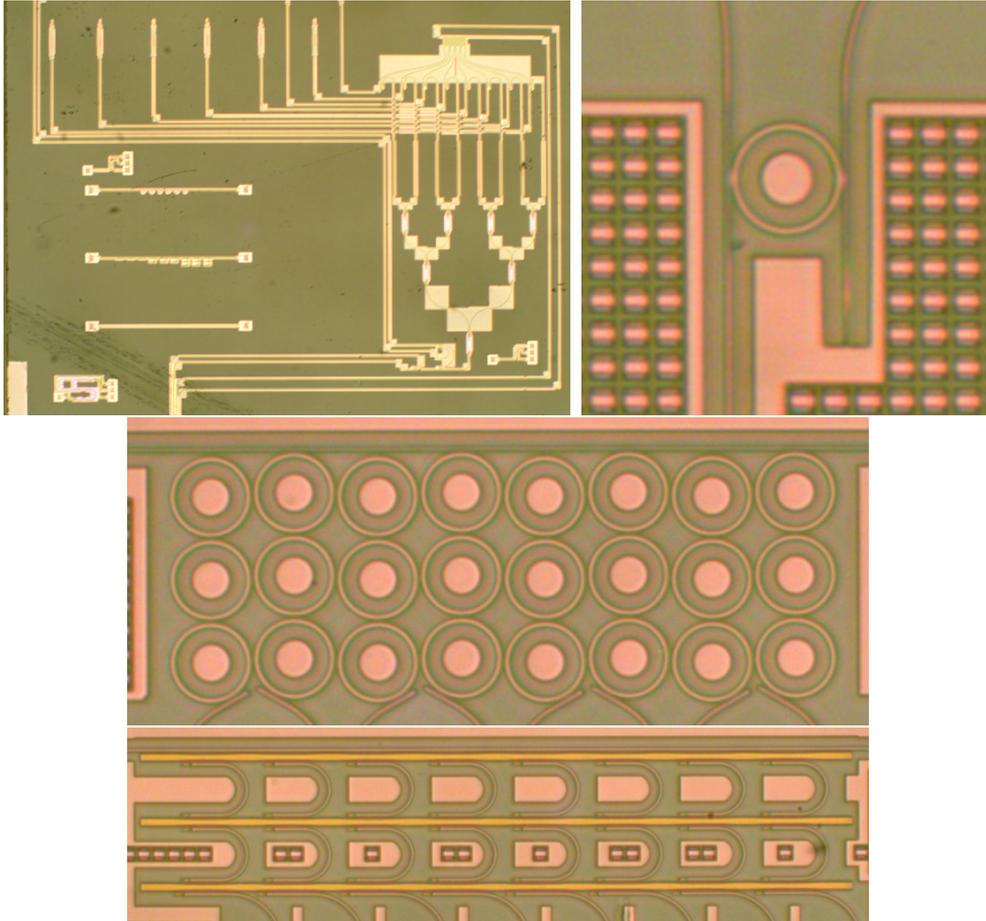


Figure 3.34: Top-left: entire chip photo. Top-right: ring resonator as an amplitude controller. Photos of microring (center) and photonic crystal (bottom) resonator-based reservoir computers on a chip.

### 3.11 . Conclusions and outlook

A highly compact integrated photonic RC was proposed, and its operation was explained. Dimensionality in the case of an exclusively linear photonic system is strongly tied to supermodes – if two supermodes overlap, only one will contribute to dimensionality. Since passive cavities have an upper Q-factor limit, the practical size of an RC is limited. Dimensionality can be increased by frequency-shifting nonlinearities: FCD and the Kerr effect, while TPA does the opposite. However, one should take care of the loss of consistency, which is detrimental to computing performance and which starts appearing when FCD and the Kerr effect are strong enough. These conclusions are general since no specific resonator type or material has been assumed and will prove helpful for future implementations.

Limited Q-factors and the associated dimensionality limitations can be mitigated with a delay line. It is possible that having two copies of RC that receive an input and its delayed version would effectively double the dimensionality and increase the memory depth.

A layout of an RC was developed and subsequently fabricated in *imec* on SOI technology. Future work will be aimed at developing circuit controls, performing extensive chip characterization, and computing itself.

## 4 - Solving tasks with an integrated all-optical reservoir

### 4.1 . Mackey-Glass prediction

In the most general sense, an RC is a dynamical system, like almost everything surrounding us. Since RCs are practically and realistically trainable today, the question arises if we can make them behave like a particular system we want. If so, that would unlock a variety of applications.

Simulation of systems is one example. Some models require considerable computational effort to integrate, and an ANN can be trained to behave like a model in question while requiring just a fraction of compute. A recent work used an ANN to accelerate a simulation of spintronic devices. A conventional approach is splitting a device into nanometer-sized cells and solving numerous Landau-Lifshitz-Gilbert equations, including local and nonlocal interactions. Some simulations may take up to a few weeks, but it was shown that using machine learning for simulation can reduce the wait more than 200 times<sup>1</sup> [172]. Also, systems could be too complex to be rigorously described by equations, for example, in meteorological predictions, but ANNs can also be helpful there [173]. Some systems may operate at gigahertz frequencies, and we may need to predict those in real-time, for example, for laser stabilization. This is where our all-optical RC can prove useful.

#### 4.1.1 . Task description

In this section, we attempt the prediction of the Mackey-Glass equation solution. This equation describes the dynamics of mature white blood cell concentration and comprises the generation and loss terms [174]. While the loss term is straightforward, the generation term is not. There is a delay between the initiation of cellular production in the bone marrow and the release of mature cells into the blood; therefore the generation term depends on a concentration from some time ago, and so this equation is the delay-differential kind:

$$\frac{d\xi}{dt} = \frac{\alpha\xi(t-\tau)}{1 + \xi(t-\tau)^g} - \gamma\xi(t). \quad (4.1)$$

Systems with delays are notorious for being prone to chaoticity [175]. In this case, with  $\tau = 17$ ,  $\alpha = 0.2$ ,  $g = 10$  and  $\gamma = 0.1$   $\xi(t)$  exhibits a moderately chaotic behaviour [174], and because of that, an inference of a future value is a task far from trivial.

---

<sup>1</sup>The cited paper does not mention the training time. However, 150 neurons were used there, and their training is not a challenge for modern computers.

The goal of an RC is to receive  $\xi(t)$  as input and generate  $\xi(t + \Delta t)$  as output, where  $\Delta t > 0$ . This problem is commonly used for benchmarking RCs as

- generating the datasets requires only solving Eq. (4.1),
- the difficulty of prediction requires both memory capacity and nonlinear transformation; this way, many facets of an RC can be tested.

Since ESNs are much more commonly used in literature, a discrete version of this task is used, where  $\xi(t)$  and  $\xi(t + \Delta t)$  are sampled. In order to compare our RC with the others, a discrete dataset is also considered and adopted for our system.

The dataset is generated in the same manner as [176]. Integrating Eq. (4.1) with the Euler method:

$$\xi(t_{n+1}) = \xi(t_n) + h \frac{d\xi}{dt}(\xi(t_n), \xi(t_n - \tau), t) \quad (4.2)$$

where  $t_{n+1} = t_n + h$  and  $h = 0.17$ . The result is then downsampled with a ratio of 3/0.17 to obtain a discrete series  $\xi(t_n)$  shown in Figure 4.1. The task then is

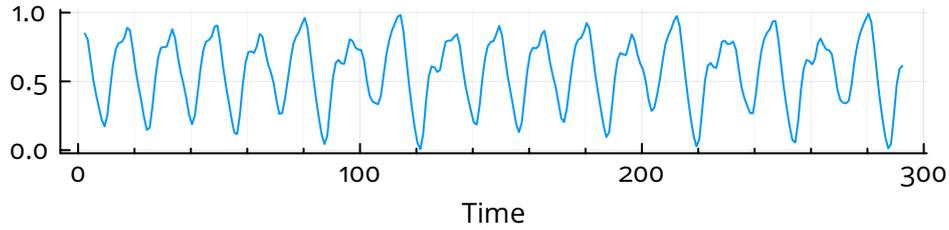


Figure 4.1: Solution of the Mackey-Glass equation Eq. (4.1) with parameters in text.

to predict  $y_n^{\text{tgt}} = \xi(t_{n+\delta})$  with  $\xi(t_n)$  as an input and a given  $\delta > 0$ . For use in our RC,  $\xi(t_n)$  is converted into a continuous signal by the zero-hold interpolation:

$$s(t) = \sum_n \xi(t_n) \text{rect} \left( \frac{t - t_n}{t_{n+1} - t_n} \right) \quad (4.3)$$

where  $\text{rect}(A)$  is a rectangular function

$$\text{rect}(x) = \begin{cases} 0, & |x| > 0.5 \\ 0.5, & |x| = 0.5 \\ 1, & |x| < 0.5 \end{cases} \quad (4.4)$$

This signal is then used to modulate the amplitude of the optical carrier that is injected into the reservoir as an input. In the same way, the target output signal is prepared. Consequently, the phase of the target signal is locked to zero, which implies we treat the task as coherent even though only the amplitude is modulated.

### 4.1.2 . Performance metrics

The prediction accuracy is measured with the NRMSE given in Eq. (2.14). For that, the RC output is sampled at  $t_n$ .

The output signal power also becomes a concern in noisy systems. Even though noise will not be present in simulations, a power penalty is also considered an additional performance metric:

$$L_p = 10 \log \left( \frac{\langle |y(t)|^2 \rangle_t}{\langle |s(t)|^2 \rangle_t} \right), \quad (4.5)$$

where  $\langle \dots \rangle_t$  is an average over time.

There is a tradeoff between NRMSE and power penalty, the balance of which is controlled by the training regularization parameter. The conjecture is that sometimes two neurons can behave similarly, with a slight difference, but that difference would be important for solution accuracy. In order to extract such a difference, ridge regression may try to increase the absolute value of readout weights, which can lead to an overly input-sensitive output [47]. In our case, a passive readout is considered; therefore, all readout weights are normalized such that the largest absolute value is unity. That means that previously obtained large weights normalize to unity, while others effectively go to zero, which harms the output power. The regularization parameter punishes large weights, so NRMSE can be sacrificed to improve the output power.

In this task, the regularization parameter is tuned so that the power penalty never goes below -25 dB. The threshold is chosen somewhat arbitrarily – the input power is expected to be on the order of 10..50 mW, whereas detecting output weaker than 100  $\mu$ W becomes tricky. The difference between the two is -25..-20 dB.

### 4.1.3 . Results and discussion

#### Spectral characteristics

Due to the nature of  $\xi(t_n)$ , the Fourier spectrum of optical input is uneven (see Figure 4.2). As a result, the excitation of supermodes is not uniform, with a nontrivial impact on the reservoir.

A commonly suggested ESN optimization procedure is the addition of bias alongside the input signal and tuning its strength [47]. It was found that in our RC, such advice is not only ineffective but also harmful. An addition of bias would increase spectra amplitude at the carrier's frequency, further damaging the supermode excitation balance.

In order to visualize the impact of nonlinearities, a linearized Jacobian of the reservoir (see Eq. (3.32)) is considered. Figure 3.23 (right) shows the trajectories of its eigenvalues while the input is injected. The inequality of non-linearity strength is evident. A supermode near a Fourier peak with a strong

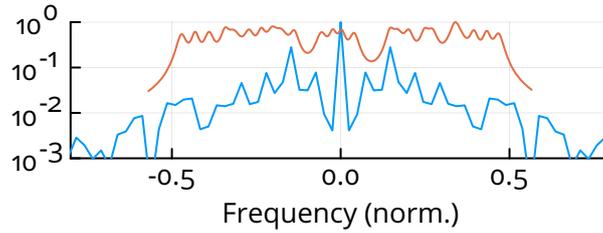


Figure 4.2: The normalized Fourier transform of the input signal (blue) with reservoir resonances overlaid (orange). Frequency is normalized by the reservoir bandwidth. Here, the reservoir bandwidth is larger than the input bandwidth, which goes against the conclusions of Section 3.6. High peaks in the input spectra correspond to a dominating harmonic ( $\xi(t)$  largely oscillates – see Figure 4.1); however, for a close prediction, it is important to capture high-frequency irregularities despite their weak amplitude.

connection to an input waveguide is subject to several times stronger nonlinearity than others. This poses a challenge when FCD and the Kerr effect are used as nonlinearity, as few supermodes would reach chaos before nonlinearity in other supermodes becomes strong enough.

### Impact of nonlinearities

The Mackey-Glass equation is a nonlinear differential equation; therefore, predicting it with a linear system is unlikely to be satisfactory.

In Figure 4.3, a considerable improvement with all three nonlinearities is seen. However, with frequency-shifting nonlinearities, the performance is lost after the normalized average input power approaches unity. This is consistent with Section 3.7, as the training becomes ineffective with a loss of consistency. However, before the consistency is lost, the output power is higher compared to TPA, which is likely due to an increase of dimensionality (see Figure 3.18).

### GaAs microring reservoir performance

For an  $8 \times 3$  reservoir with TPA and FCD, the NRMSE of 0.15 was achieved (see Figure 4.4) and 0.08 for TPA, considering the power penalty restriction. For comparison, the performance baseline is  $\text{NRMSE}(u_n, u_{n+3}) \approx 1.1$ , an all-optical time-multiplexed reservoir with 330 virtual neurons based on a semiconductor laser experimentally achieved NRMSE of 0.23 [176], and a simulated time-multiplexed reservoir based on a nonlinear MRR with 25 virtual nodes had NRMSE of  $7.2 \cdot 10^{-2}$  [177]. However, an ESN with 400 neurons outperforms them by a large margin, predicting approximately 20 steps ahead with an NRMSE of  $1.2 \cdot 10^{-4}$  [41]. Nevertheless, the proposed reservoir performs such computation fully-optically in real-time.

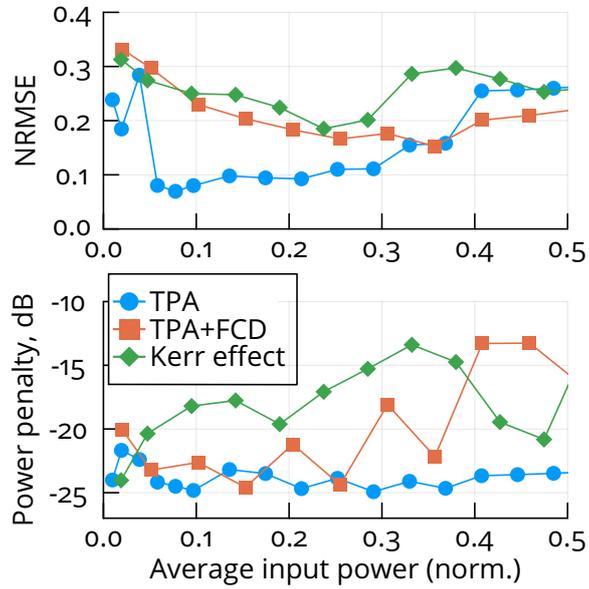


Figure 4.3: Impact of nonlinearities on (a) prediction error and (b) power penalty on a 3-step Mackey-Glass prediction. For all cases, GaAs microring parameters are used (see Table 3.1), but for the TPA-only regime,  $d\omega/dN$  is set to zero, while for the Kerr effect  $\beta^{\text{TPA}} = 0$  and  $d\omega/dI < 0$  is chosen arbitrarily. The regularization constant was increased until the power penalty reached -25 dB.

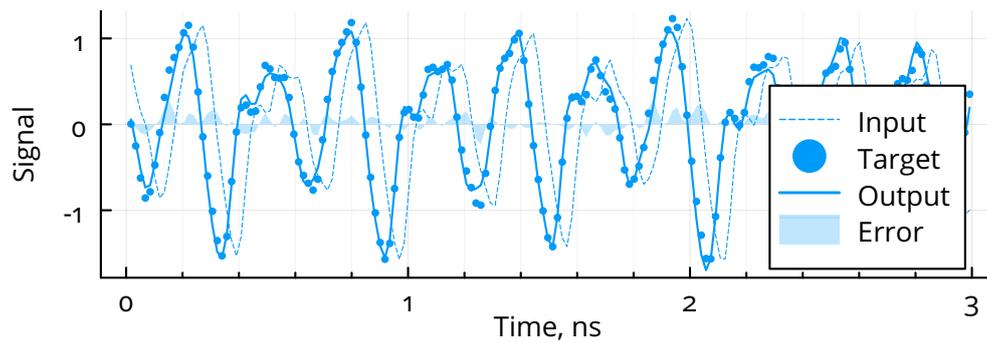


Figure 4.4: 3-step Mackey-Glass prediction by an  $8 \times 3$  GaAs microring resonator at an optimal input power. Here, the NRMSE is 0.15 and  $L_p \approx -22$  dB.

If we require only the amplitude of the output signal to be correct, for the same reservoir with TPA and FCD, the NRMSE reduces to 0.06. However, the original training problem  $\hat{W}^{\text{out}}\mathbf{x}(t) = y^{\text{tgt}}(t)$  becomes  $|\hat{W}^{\text{out}}\mathbf{x}(t)| = |y^{\text{tgt}}(t)|$ , and ridge regression stops being applicable.

### Performance scalability

Simulations have shown that prediction performance improves with the number of cavities (see Figure 4.5) for  $N_{\perp} = 1, 3$ , and all three nonlinearities. With FCD and the Kerr effect,  $N_{\perp} = 1$  and  $N_{\perp} = 3$  reservoirs have shown similar performance, while for TPA  $N_{\perp} = 3$  was better than  $N_{\perp} = 1$ . The lowest error with an NRMSE of  $3.7 \cdot 10^{-2}$  and  $L_p$  of -25 dB is achieved for 48 cavities with  $N_{\perp} = 3$ . The importance of the evanescent coupling is also noted by comparing our system with a set of independent cavities, a linear regime of which was considered by [178]. For a fair comparison, resonances of cavities were set to resonances of supermodes of a corresponding  $N \times 1$  reservoir. Coupled cavities provided considerably better performance with all nonlinearities, likely due to a nonlinearity-induced supermode mixing.

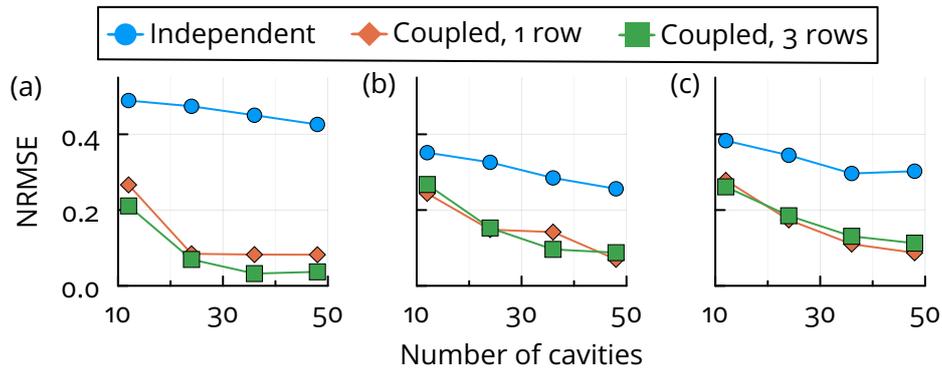


Figure 4.5: Scalability of 3-step prediction performance with various geometries and (a) TPA, (b) TPA with FCD and (c) the Kerr effect. The regularization constant was increased for each point until the power penalty reached -25 dB. Here, minimum NRMSE (a) 0.037 (b) 0.070 (c) 0.087.

#### 4.1.4 . Conclusions

The proposed integrated all-optical RC can imitate the nontrivial dynamics of the Mackey-Glass equation. The importance of evanescent coupling and scalability of prediction performance with the number of cavities are noted. It was found that all TPA, FCD and the Kerr effect contributed to the prediction performance, with TPA giving a somewhat better result at the cost of a lower output power. No difference in the performance of an RC with 1 or 3 rows of cavities was found. In the previous section it was shown that having many

rows would reduce dimensionality due to supermodes that are not excited, which is expected to harm performance. However, in this case, 3 rows is not enough to significantly uncouple supermodes.

Prediction accuracy is comparable to physical RC found in the literature; however, an ESN showed better performance [41], which, apart from a larger neuron count (400 compared to a few tens in our case), can likely be attributed to:

- an unrestricted neuron leak rate, which could improve dimensionality,
- a more complex neuron connectivity structure, as the connection matrix was generated randomly,
- a more homogenous input injection, which is shown to be important in [179],
- an unrestricted power penalty.

In addition to that, our RC has not been optimized for this task, and its parameters have been chosen according to general conclusions made in the previous section; therefore, an improvement of the prediction accuracy is possible.

## 4.2 . Optical fiber data recovery

Regarding information transmission, an optical fiber is usually the best option. However, it still introduces distortions into the transmitted signal, a notorious problem limiting transmission performance [180]. This section aims to apply the proposed RC to recovery from such distortions.

The distortion of the signal propagating in optical fiber communication channels can be separated into linear and nonlinear contributions. Linear distortions include filter effects, chromatic dispersion (CD) and polarization mode dispersion, among others. Nonlinear distortions include self-phase modulation (SPM), relevant for single-channel communication, cross-phase modulation (XPM) and four-wave mixing (FWM), relevant at multi-channel communication (e.g. DWDM), and other effects [181].

Dispersion could be countered optically with dispersion-compensated and dispersion-shifted fibers or, after detection, feed-forward and decision feedback equalizers. Countering nonlinearities is considerably more challenging. In Section 1.4.1, a few approaches are given; however, all of them are computationally expensive, and the algorithm complexity has to be reduced for the online processing of fast signals, which limits how much nonlinearity can be undone. It was also shown that ANNs could be an efficient alternative, but running complex enough ANNs even on the fastest electronic devices available off-the-shelf is a challenge.

For low-latency networks, keeping the signal in the same domain as long as possible is important. Each OEO conversion takes up to 100  $\mu$ s, depending

on how much electronic processing is required [182]. Reamplification, reshaping and retiming (full '3R') regeneration should also be avoided when possible [19].

Therefore, there is an interest in replacing the electronic treatment of optical distortion or at least its partial offload to the optical domain. This section discusses how the proposed RC can handle this problem.

#### 4.2.1 . Signal transmission

In order to transmit information over an optical fiber, it has to be mapped to an optical signal – an optical carrier's amplitude and, possibly, phase is modulated, and the result is designated as  $s_0(t)$ . Each unique pair of amplitude and phase would form an 'optical alphabet', to which the original datastream is translated in the binary form (i.e. a stream of '0's and '1's). The way binary data is mapped onto an optical field is called the modulation format.

The rate at which we switch between the 'letters' is called baud. Baud and bandwidth are related but different, as the former describes the rate at which information is transmitted, whereas the latter corresponds to how broad the optical signal's spectrum is. If the modulation speed doubles, baud and bandwidth double, all else being equal. If the 'alphabet' capacity increases two times, baud doubles, while bandwidth does not. Therefore, increasing the 'alphabet' improves the data transmission rate.

There are various modulation formats in use. The simplest one is on-off keying (OOK), where the amplitude can have two values: zero and non-zero. Here, we have only two 'letters' in the 'alphabet'; therefore, the translation is simple: each 0 is mapped to the zero amplitude and each 1 to the non-zero amplitude (see Figure 4.6). Simply switching between the letters would be non-return-to-zero on-off-keying (NRZ-OOK). There is also a variation called return-to-zero on-off-keying (RZ-OOK), where the amplitude is set to zero between each symbol. In this case, periodic zero-out acts as a clock, which helps synchronize a receiver and a transmitter. The price is that the bandwidth of the signal almost doubles with the same baud.

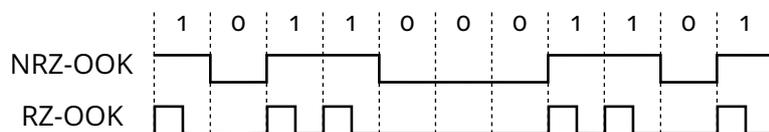


Figure 4.6: Non- and return-to-zero on-off keying modulation. Adapted from [183].

The amplitude can be modulated to more than just two levels. For example, modulating to four levels would result in the Pulse Amplitude Modulation format with four levels (PAM-4). The 'alphabet' capacity is then increased  $\log_2(4) = 2$  times since  $N$  binary digits correspond to one of  $2^N$  different

states optical field. What if the number of levels is not a power of 2? Then, a bit sequence can be transmitted over a sequence of symbols. Theoretically, each PAM-3 symbol carries  $\log_2(3) \approx 1.58$  bits. For example, PAM-3 is used in USB4 Gen4 to transmit 11 bits over seven symbols [184, p. 128] – this way, each symbol carries  $11/7 \approx 1.57$  bits, which utilizes approximately 99% of the total capacity.

However, using only the amplitude is wasteful – a second dimension, phase, can also carry information. Then, such a modulation format would be coherent. One format utilizing phase is quadrature amplitude modulation (QAM), shown in Figure 4.7. For example, each symbol of 64-QAM would carry  $\log_2(64) = 6$  bits.

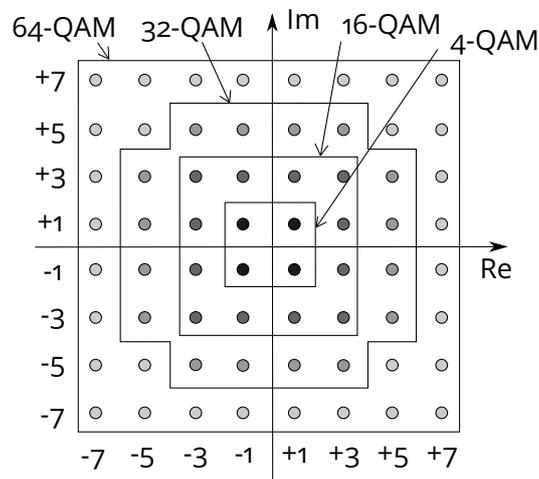


Figure 4.7: Constellations of various quadrature amplitude modulation formats on a complex plane. Adapted from [185].

An increase in ‘alphabet’ size does come at a cost:

- The distance in the complex plane between symbols reduces, and the link becomes more sensitive to noise.
- Demapping becomes more computationally intensive.
- The use of a coherent modulation format requires the use of coherent receivers, which are costly compared to incoherent ones [186].

### 4.2.2 . Task description

Omitting negligibly weaker effects like the Raman scattering, an electric field  $E(z, t)$  in a fiber can be described with a nonlinear Schrödinger equation [187]:

$$i \frac{\partial E}{\partial z} + i \frac{\alpha_{\text{loss}}}{2} E - \frac{\beta_2}{2} \frac{\partial^2 E}{\partial t^2} + \gamma |E|^2 E = 0, \quad (4.6)$$

where  $\alpha_{\text{loss}}$  is attenuation,  $\beta_2$  is dispersion coefficient, and  $\gamma$  is a fiber nonlinear parameter. At the boundaries,  $E(0, t) = s_0(t)$  and  $E(L, t) = s(t)$  is the received signal. Here,  $\gamma$  is related to  $n_2$  via [187, eq. 2.3.29]

$$\gamma = \frac{n_2 \omega}{c A_{\text{eff}}}, \quad (4.7)$$

where  $A_{\text{eff}}$  is an effective fiber mode area

$$A_{\text{eff}} = \frac{\left[ \iint_{xy} |\mathbf{E}(x, y)|^2 dx dy \right]^2}{\iint_{xy} |\mathbf{E}(x, y)|^4 dx dy}, \quad (4.8)$$

though experimentally determined values of  $\gamma$  are usually available for commonly used fibers.

Table 4.1: Parameters of the SMF-28 optical fiber [188].

Parameter	Designation	Value
Attenuation	$\alpha_{\text{loss}}$	0.2 dB/km
Dispersion coefficient	$\beta_2$	21.7 ps <sup>2</sup> /km
Kerr coefficient	$\gamma$	1.3 W <sup>-1</sup> /km

The training target depends on the use case. If after an RC, no further processing is done, i.e. just detection, then it can be trained to recover samples:  $y(t_n) \approx s_0(t_n)$ . Otherwise, extra info may be required, for example, when a raised cosine filter (see Figure 4.8) is used for pulse shaping – a common strategy for reducing intersymbol interference. In this case, more than one sample per bit is needed, and recovering those is almost equivalent to recovering the waveform:  $y(t) \approx s_0(t)$  due to the Nyquist-Shannon sampling theorem. In practice, a raised cosine filter is split into two matched root-raised cosine filters (RRCF) at the transceiver and the receiver sides. Since filtering is performed electronically, it is natural to use an RC before the final filtering step (see Figure 4.9). In this case,  $s_0(t)$  is a signal filtered by an RRCF before propagation and  $s(t)$  is its propagation result.

For RC training,  $y^{\text{tgt}}(t) = s_0(t - \tau)$  is chosen where  $\tau$  is an output delay and sample  $s_0(t)$ ,  $s(t)$  and  $z_k(t)$  once per bit in their middle. The presence of an output delay is motivated by group velocity dispersion that causes different input signal frequencies to arrive at a receiver at different times. By delaying

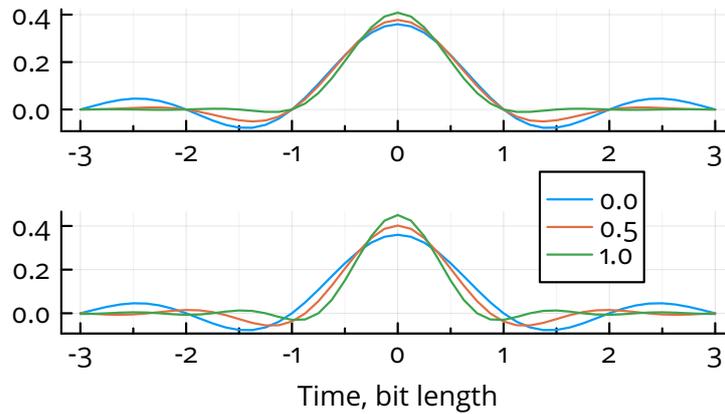


Figure 4.8: Impulse response function of a raised- (top) and a root raised-cosine filters (bottom). The legend shows the roll-off factor; a lower value corresponds to a narrower bandwidth. Raised cosine passes through zero every bit length, i.e. samples are unchanged.

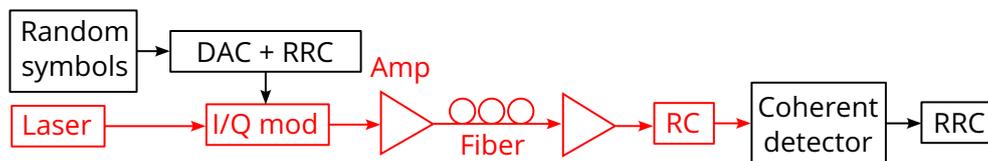


Figure 4.9: Scheme of an optical fiber link. Electrical and optical components are shown in black and red. Here, RC is the reservoir computer, and RRCF is the root-raised cosine filter.

$y^{\text{tgt}}(t)$ , the reservoir can accumulate more information about the current symbol before producing an output. It will be shown later that the output delay leads to a better separation of output samples at a given power penalty. For RC testing,  $y(t)$  and  $y^{\text{tgt}}(t)$  are similarly sampled to produce  $y_n$  and  $y_n^{\text{tgt}}$ ; an appropriate classifier  $C$  is used for demapping.

Performance is measured with the Symbol Error Rate (SER):

$$\text{SER}(y, y^{\text{tgt}}) = \sum_n \mathbb{1}[C(y_n) \neq C(y_n^{\text{tgt}})] / N_t, \quad (4.9)$$

where  $\mathbb{1}[x]$  is 1 when  $x$  is true and 0 otherwise and  $N_t$  is the number of transmitted symbols in the testing dataset.

Since a simulation or an experiment counting the number of errors in the stream cannot be infinitely long, the minimum measurable SER is limited. In [189],  $1/N_t$  was used as a lower limit, but here, we choose such that it cannot be lower than an inverse of a count of the least common symbol in the stream

$$1 / \min_S \left\{ \sum_n \mathbb{1}[C(y_n^{\text{tgt}}) = S] \right\}, \quad (4.10)$$

which is a similar but more pessimistic limit. Previously introduced power penalty Eq. (4.5) is also used.

### 4.2.3 . Results and discussion

#### Metro link

An optical signal carrying a random sequence of symbols  $s_0(t_n)$  is generated according to the parameters given in Table 4.2. Before propagating through a fiber, a signal was passed through a low-pass filter (see Figure 4.10).

Table 4.2: Parameters for simulation of propagation through a metro optical link.

Parameter	Value
Random number generator	Mersenne Twister
Modulation format	OOK
Baud	25 GHz
Launch power	10 dBm
Optical fiber	50 . . . 300 km SMF-28
Pulse shaping	Low-pass filter, 0.6×25 GHz cutoff
Samples per bit	8
Integration algorithm	<i>ode45</i> in <i>Octave</i>
Relative error tolerance	$10^{-3}$

Then, Eq. (4.6) is solved numerically in *Octave* using an in-house solver to obtain distorted signal samples  $s(t_n)$ , which are linearly interpolated to  $s(t)$  and injected into the reservoir.

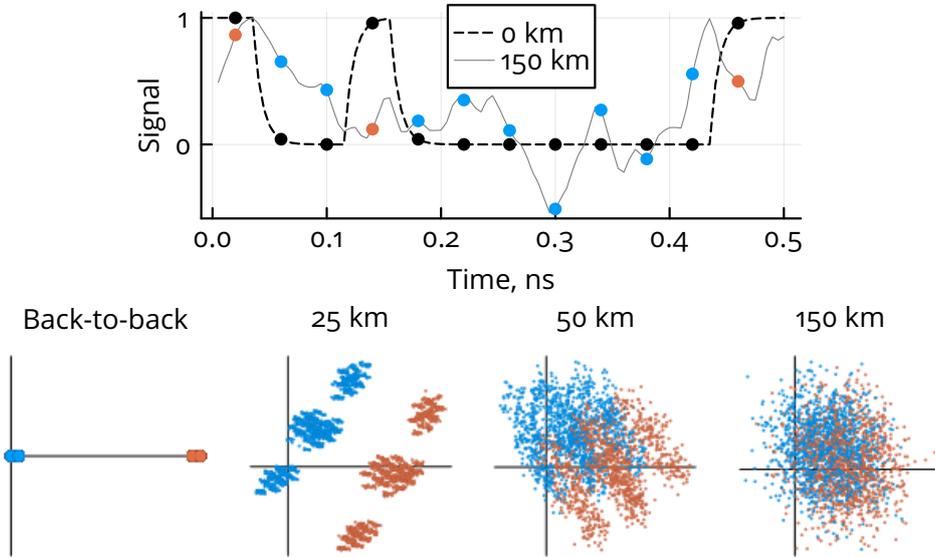


Figure 4.10: Top: timetraces of a real part of the original signal (dashed line) and the distorted signal after 150 km of optical fiber (solid line). Black dots show a bit middle points and act as the training target; blue and orange are only for visualization and show distorted signal samples with color corresponding to each symbol. Bottom: constellations of the distorted optical signal samples after propagating through a length of optical fiber given above each plot. Parameters are given in Table 4.2.

The reservoir parameters are the same as in the Mackey-Glass task (see Table 3.1) except for  $\mu = 40$  GHz, chosen according to the  $s(t)$  spectrum (see Figure 4.11) and nonlinear effects disabled, as at these fiber parameters and launch power, the impact of nonlinearity is relatively weak and can be handled by a classifier. As in the previous section,  $\mu$  is increased for  $N_{\parallel} \times 1$  reservoir versions to equalize reservoir bandwidth.

As was mentioned before, a pulse in the input signal gets spread out in time due to dispersion. By delaying the output signal we allow the RC to accumulate more data, allowing for better equalization. For used fiber parameters, a delay of 4 bits is adequate and is used in this section (see Figure 4.12).

A larger reservoir can handle a distortion from a longer fiber (see Figure 4.13). The performance of an  $N_{\parallel} \times 1$  grid of cavities was comparable to an  $N_{\parallel} \times 3$  grid and outperformed independent cavities.

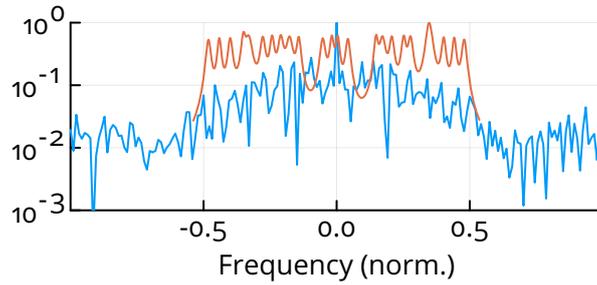


Figure 4.11: Normalized Fourier spectrum of the distorted signal after 150 km of fiber (blue) with  $8 \times 3$  reservoir resonances overlaid (orange). Frequency is normalized by the reservoir bandwidth.

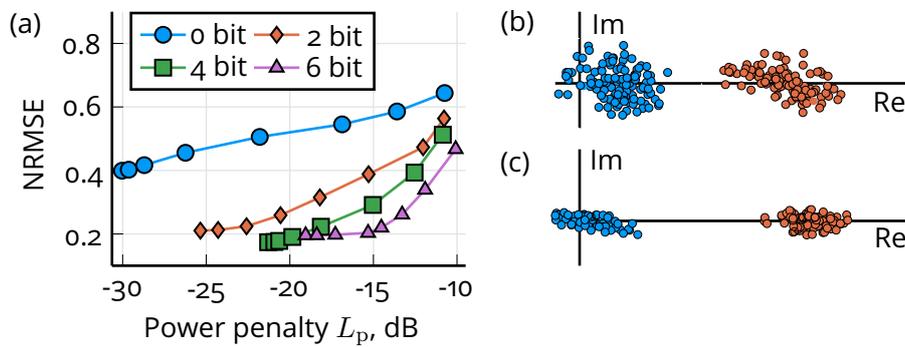


Figure 4.12: (a) Effect of output delay (shown in legend) on output symbol separability measured with NRMSE at a given power penalty, controlled with regularization parameter. Here, fiber length is 150 km, and the signal is processed with an  $8 \times 3$  linear reservoir. (b)(c) Constellations of output symbols at power penalty of approximately -20 dB with 0- and 4-bit delay, respectively. Figure 4.10 (bottom) shows the constellation of input signal samples.

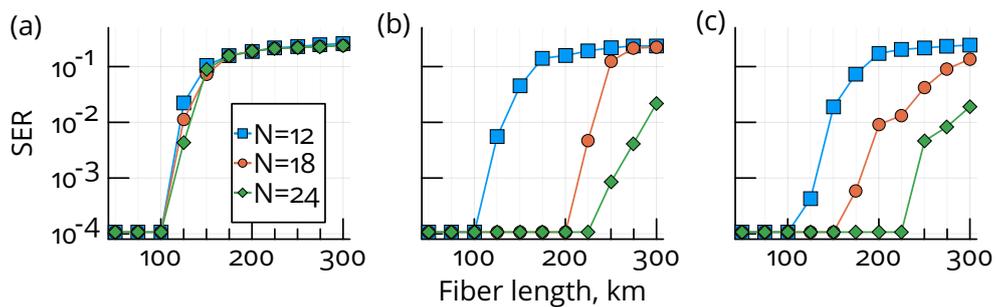


Figure 4.13: Performance scaling of optical signal recovery w.r.t. reservoir size. The reservoir is (a)  $N$  independent, (b)  $N_{\parallel} \times 1$  grid and (c)  $N_{\parallel} \times 3$  grid of linear microring resonators. The regularization parameter is increased for each computation until the power penalty is no less than -25 dB. Here, SER is the symbol error rate.

## High-speed short-reach link

In all previous tasks, the output was incoherent<sup>2</sup>. In this section, the recovery of coherent signal samples is considered, where both amplitude and phase recovery are required, which is one of the strong points of an all-optical coherent RC.

The preparations are similar to the previous section, with the parameters given in the left column of Table 4.3, where we consider a 16-QAM modulation format at a commonly used launch power. The same linear reservoir is used as in the previous section, and  $\mu$  is scaled to match the signal bandwidth.

In the previous section, *Octave* was used to solve Eq. (4.6). Since then, we have reimplemented a solver in the *Julia* programming language [146] in order to utilize a wider toolset and modernize the codebase. However, since results for a metro link have already been submitted to a journal, they have not been recreated. In this section, we use a new solver (see Section C for details) with parameters in Table 4.3.

Table 4.3: Parameters for simulation of propagation through a short-reach optical link in default and highly nonlinear cases. Fiber parameters are given in Table 4.1.

Parameter	Value (default)	Value (nonlin.)
Random number generator	Xoshiro256++ [190]	
Modulation format	16-QAM	
Baud	25 GHz	
Launch power	5 dBm	14 dBm
Optical fiber	80 km SMF-28	30 km SMF-28
Pulse shaping	RRCF, 0.1 roll-off factor	
Samples per bit	8	
Integration algorithm	<i>Verng</i> in <i>DifferentialEquations.jl</i>	
Relative error tolerance	$\cdot 10^{-3}$	

Signal distortion and recovery results are shown in Figure 4.14 and Figure 4.15. A linear reservoir performed well here since such launch power does not cause a strong nonlinear response.

<sup>2</sup>although the RC was trained as if it was, the phase was simply locked to zero

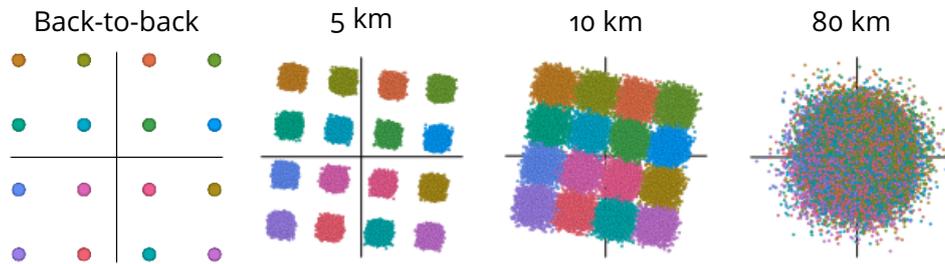


Figure 4.14: Constellations of optical signal samples after propagating through a length of optical fiber given above each plot. Parameters are given in the left column of Table 4.3. In the left figure constellation are simply dots enhanced for visibility.

### Highly nonlinear high-speed short-reach link

Consider a more challenging case, where RC's intrinsic nonlinearity will have to be used. With the parameters given in the right column of Table 4.3, a dramatic increase in launch power will elicit a strong nonlinear response in a fiber. In Figure 4.16, a 'smearing' of samples can be seen, which reduces the SER even after 5 km. Combined with a coherent modulation format, this poses a much harder challenge for an RC.

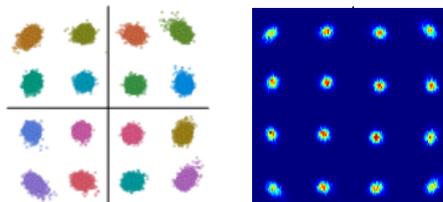


Figure 4.15: Constellation (left) and histogram (right) of signal samples after 80 km of optical fiber recovered with a linear reservoir. The power penalty is -12 dB. Figure 4.14 shows the distorted signal.

In order to handle such a task, the reservoir size is increased to 36 GaAs MRRs arranged in three rows. In this task, the photon lifetime was significantly reduced since a long linear memory is unnecessary (and even harmful) after just 30 km of fiber. To that end, the Q-factor was reduced  $3 \cdot 10^5 \rightarrow 1 \cdot 10^5$ , and  $\kappa_{\text{out}}^2$  increased  $\pi \rightarrow 8\pi$  GHz, which reduced photon lifetime to about two bit periods.

One can see that a linear reservoir fails to separate the symbols (see Figure 4.17). Symbols at the edge of the constellation correspond to a higher amplitude and are the most affected. However, an intrinsic nonlinearity provides a noticeable improvement. Even though a perfect recovery was not achieved

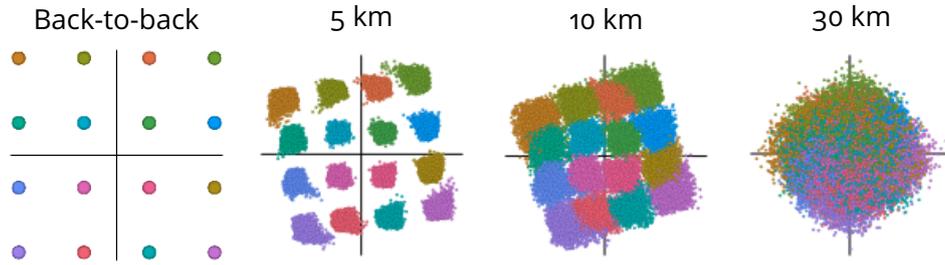


Figure 4.16: Constellations of optical signal samples in highly nonlinear case after propagating through a length of optical fiber given above each plot. Parameters are given in the right column of Table 4.3. In the left figure constellation are simply dots enhanced for visibility.

(for that reason, there is no SER estimate), the difficulty of further processing is expected to reduce considerably. Moreover, this is an exaggerated case, and such strong launch power is not used.

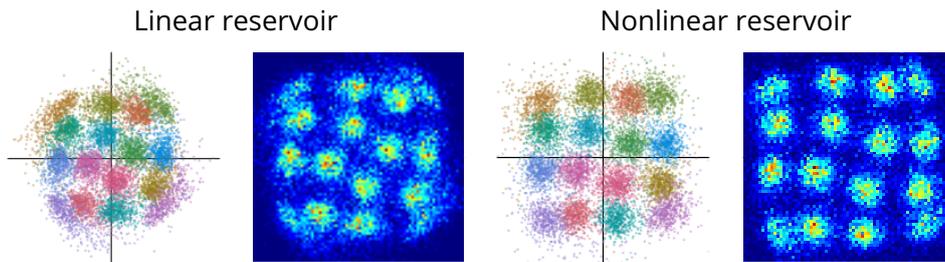


Figure 4.17: Recovery of strongly nonlinearly distorted signal after 30 km of optical fiber with linear and nonlinear otherwise identical reservoirs. Each pair shows a constellation and a histogram of samples. The power penalty is around -20 dB in both cases. Figure 4.16 shows the distorted signal.

#### 4.2.4 . Conclusions

This section demonstrates that the proposed RC is capable of a highly relevant task – recovering data from an optical fiber. This RC has been applied to recovering data from metro and short-range links. Even though our simulation of distortions is rather simplistic as several technical details were omitted, like optical carrier noise, detection noise and filtering effect, the main objective was an inversion of distortion even in a highly nonlinear case, which has been demonstrated.

Further work may include recovery of WDM channels and recovery of distortion of long haul optical links, hence dealing with interchannel interference.



## 5 - Spiking neural network in integrated photonics

Previous chapters discuss a few implementations of neural networks. Even though neuromorphic computing is biologically inspired, such networks employ drastically simplified models of neurons and synapses, and the entire approach to computing differs from a real brain.

- Digital implementations are almost universally discrete, and neurons in such emulations exchange signals in a 'synchronized', i.e. clocked fashion. This approach is used when digital electronics are heavily involved in the computing process
- In continuous implementations, neurons exchange non-discretized and smooth signals in real time. Since most physical systems are smooth and, therefore, suit well, several implementations outlined in Section 2 have been demonstrated.

Real neurons exchange spikes asynchronously, with some works speculating that their operation has inherent stochasticity [191], which previously discussed models mostly discard. Nonlinearity is also different. Biological dendrites, for example, are nonlinear. In typical ANNs, all neurons have time-invariant nonlinearities, whereas the real neurons are excitable, see Figure 5.1. According to A. L. Hodgkin, excitable neurons can be classified into three

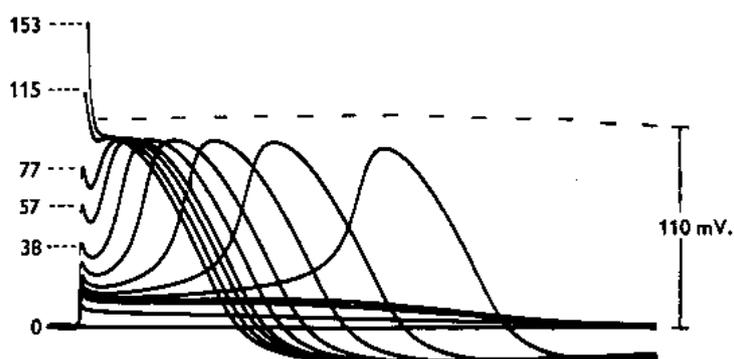


Figure 5.1: Excitability of a giant squid neuron, vertical axis shows neuron voltage [192]. Weak inputs induce a linear response, but stronger responses elicit a spike due to ionic currents inside the cell membrane. Here, the excitation strength only impacts the spike delay.

classes [193]:

class 1: neurons that emit spike trains in a wide range of frequencies, depending on the excitation strength,

class 2: neurons that emit spike trains in a narrow range of frequencies, largely insensitive to the excitation strength,

class 3: neurons that emit only one spike in a certain period of time, irrespective of the excitation strength,

and each class would be suitable for a particular way of transmitting information.

Suppose a neuron is a class 1 or 2. In that case, information can, for example, be encoded in how frequently spikes are emitted, which is so-called rate-based coding, somewhat similar to continuous neural networks. For some time, rate-coding was assumed to be the way information is coded in a brain [194]. However, in the 1990s, an experiment on macaque monkeys demonstrated a fault in such an assumption. The macaque's cortical area can spike at 100 Hz, whereas visual pattern recognition was carried out as fast as 30 ms [195], which is hardly enough for a faithful rate coding. Another way is temporal coding, where the time a spike has arrived is important. This approach is particularly interesting for event-based computing, where a computation process needs to run only at a particular moment. For example, run an object-recognition algorithm only when something moves in a frame, which reduces power consumption [196].

There is an interest in spiking neural networks (SNNs) as they are closer to the source of ANN inspiration – a brain that demonstrates what *is* possible – an exceptionally complex computing device using just 20 W of power. There are also claims about the practical advantages of SNNs, such as

- more computing power per neuron compared to regular ones [197],
- in principle, more energy efficient operation, as a neuron is in the emission state only a fraction of time [198],
- better resistance to noisy environments, as sharp spikes are easier to spot in the sea of noise [194],

that, however, are still under debate.

Several neuron models approximate biological neurons [199], leaky integrate-and-fire (LIF) being one of the simplest and widely used. This model is very old [200] and only received its name half a century later [201]. In this model, a neuron is represented as a capacitor with potential  $u$  and capacitance  $C$ , which can be charged by external currents  $I$  that can be a weighted sum of signals from other neurons, and this charge dissipates over time with a time constant  $C/g$ :

$$C \frac{du}{dt} = -gu + I. \quad (5.1)$$

If the charge exceeds a certain threshold at a time  $t_{\text{spike}}$ , a spike is emitted, and  $u$  is forced to a ‘reset’ potential  $\rho$  for a duration called ‘refractory period’  $\tau_{\text{ref}}$ , i.e.

$$u(t_{\text{spike}} \leq t < t_{\text{spike}} + \tau_{\text{ref}}) = \rho. \quad (5.2)$$

Spiking networks have been implemented in digital and analog electronics. In 2014, IBM created the 4.3 cm<sup>2</sup> electronic chip ‘TrueNorth’ with 4096 cores capable of simulating 1 million neurons with 256 million synapses with processing speed limited to a few hundred hertz [202]. In 2018, Intel presented the 0.6 cm<sup>2</sup> ‘Loihi’ chip with 128 cores, each implementing 1024 neurons, i.e. 131,072 total, capable of faster computing of a few megahertz [203]. To compare with Figure 2.33, Loihi’s neuron density is about 2200 neurons/mm<sup>2</sup>. In 2021, its successor, ‘Loihi 2’, was announced, with almost the same number of neurons and synapses but improved interfaces and programmability [204]. Other implementations have been compared in a review [205]. These chips have put computing modules and memory next to each other, avoiding the ‘von Neumann bottleneck’. Similar to the case of conventional high-performance computing, photonic interconnects may be considered to overcome the bottleneck. In a long-term perspective, all-optical SNNs may be attractive.

Spiking in passive MRRs [206] and PhCs [207] has been demonstrated, though they relied on slow thermal effects, which limited the operation frequency up to 1 MHz. Injection-locking can produce spikes at a faster timescale – a perturbation of an injection beam can induce the excitable behaviour of a VCSEL with a response of a few hundred picoseconds [208] (see Figure 5.2). However, injection locking requires a laser per neuron, which limits scalability. A more scalable alternative is a two-section VCSEL with a saturable absorber [209]. However, VCSELs are hard to couple on a single chip due to the vertical mode orientation. In-plane resonators resolve this issue, and they have been considered in very recent works in the form of Fabry-Perot [210] and photonic crystal cavities [211], each with the threshold energy of a few femtojoules and gigahertz timescale. An optoelectronic DFB laser-based setup was also demonstrated [212]. This chapter targets integrated PhC lasers shown in Figure 5.3 as they are very compact and require low power to operate [213]; recent work has achieved spiking behavior by incorporating a two-section design with a saturable absorber [214].

A reasonable question one might ask is how to make anything useful out of spiking neurons. One answer lies in the Boltzmann Machines, which we discuss in the next section.

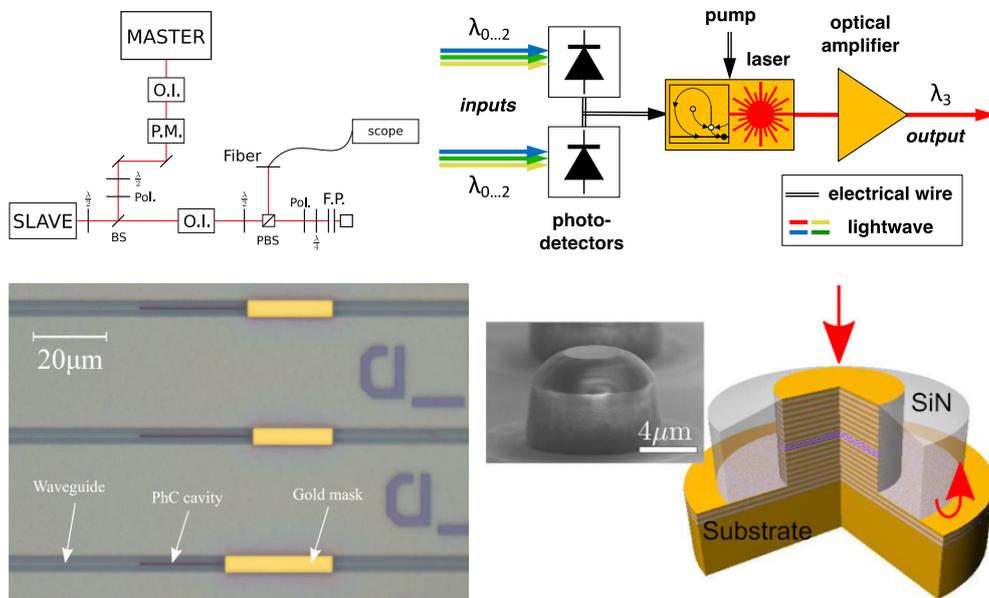


Figure 5.2: Examples of spiking photonic systems. Top left: injection-locking setup [208]. Top right: distributed-feedback laser, adapted from [212]. Bottom left: photonic crystal with saturable absorber [214]. Bottom right: vertical-cavity surface-emitting laser with saturable absorber [209].

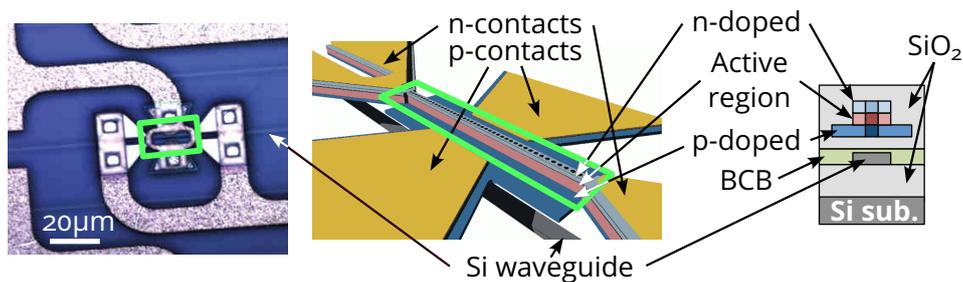


Figure 5.3: Photo (left) and schematic (center, right) of a hybrid one-dimensional photonic crystal InP nanolaser heterogeneously integrated on top of a silicon-on-insulator waveguide. Here BCB is benzocyclobutene. Adapted from [213].

## 5.1 . Boltzmann Machine

A Boltzmann Machine (BM) is a stochastic Ising model, where binary neurons (i.e. that can be in two states) interact with each other according to a given connection weight matrix. This matrix also dictates the system's energy, which impacts a BM's evolution. This section introduces BMs in a very simple manner; for a rigorous approach, I refer to [215].

First, assume a network of perceptrons with a threshold function

$$f(x) = \begin{cases} +1, & x \geq 0 \\ -1, & x < 0 \end{cases} . \quad (5.3)$$

Each pair of neurons is connected symmetrically, i.e. connection matrix  $\hat{W}$  is symmetric, i.e.  $W_{ij} = W_{ji}$ , but  $W_{ii} = 0$ . Then, for each neuron, we can compute the sum of 'forces' acting upon it:

$$u_i(n+1) = \sum_j W_{ij} z_j(n) + b_i, \quad (5.4)$$

where  $z_j(n) = \pm 1$  shows if a  $j$ -th neuron is active (+1) or not (-1) at time  $n$  and  $b_i$  is a constant bias. This sum will define the new state of a neuron according to

$$z_i(n+1) = f[u_i(n+1)]. \quad (5.5)$$

For a state of the network  $\mathbf{z}$ , we can define an energy function

$$E(\mathbf{z}) = -\frac{1}{2} \mathbf{z}^\top \hat{W} \mathbf{z} - \mathbf{z}^\top \mathbf{b}. \quad (5.6)$$

If a connection between two neurons is positive,  $z_i = z_j$  would correspond to a lower energy than  $z_i = -z_j$ , and the opposite is true for negative weights. If we were to compute  $-\partial E / \partial z_k$  formally, we would obtain Eq. (5.4), i.e. an update of each neuron would reduce the total network energy<sup>1</sup>. This way, a Hopfield network is defined [216].

How is it useful? Each  $\mathbf{z}$  can correspond to some dataset, e.g. the brightness of pixels of an image. We can set  $\hat{W}$  and  $\mathbf{b}$  such that  $E(\mathbf{z})$  has local minima at specific points, e.g. when corresponding pixels form an image of a handwritten number. Each time we write a number on paper, it turns out differently. So, a new image to be processed will correspond to  $\mathbf{z}'$ , which is not at the minimum. Now we force the initial state of the network to the data we have, i.e.  $\mathbf{z}(0) = \mathbf{z}'$ . During the update procedure, the network will slide down the energy slope towards a minimum, hopefully, corresponding to a 'true'  $\mathbf{z}$ . This way, we infer from partial, distorted or noisy data.

---

<sup>1</sup>the order at which neurons are updated is typically not important [215]

What if the energy surface is complex and there are unwanted local minima? Hopfield networks are deterministic, so if the energy gradient drives towards such a minimum, the network will end up and stay there, and inference will fail. This problem can be solved by allowing the network to ‘make mistakes’ by making it stochastic. For that, we change the update procedure.

Instead of a deterministic thresholding Eq. (5.3) we will use the sigmoid that would give a probability that a neuron would follow where the ‘forces’ lead:

$$p(z_i = 1) = \sigma \left( \sum_j W_{ij} z_j + b_i \right), \quad (5.7)$$

where  $p(z_i = 1)$  means a probability of a neuron turning on in this network state. In the end, the network randomly walks through its phase space, preferring to go towards lower energy states. Here, the network follows some probability distribution, and the states it steps in are referred to as ‘samples’, and the process of randomly walking is called ‘sampling’. One can show that after walking for a long enough time, the probability that the network would be in state  $\mathbf{z}$  would be described by the Boltzmann distribution

$$p(\mathbf{z}) = Z^{-1} \exp(-E(\mathbf{z})), \quad (5.8)$$

where the temperature term is unity  $kT = 1$  and

$$Z = \sum_{\mathbf{z}} \exp(-E(\mathbf{z})) \quad (5.9)$$

is a partition function that ensures the probability normalization

$$\sum_{\mathbf{z}} p(\mathbf{z}) = 1. \quad (5.10)$$

Hence the name: the Boltzmann Machine. Then, to use a BM as we did a Hopfield Network, we need to find the most visited state during the walk.

We have seemingly eliminated the need for sampling – why bother when we can compute  $p(\mathbf{z})$  analytically? The answer is in the computational complexity: if a BM has  $N$  neurons, computing  $Z$  requires a summation of  $2^N$  terms, quickly becoming untractable for higher  $N$ . For the same reason, a computation of marginal and conditional distributions is easier since a part of the sampled state can be simply dropped.

The tradeoff of sampling, however, is that it does not provide a perfectly accurate result, but the accuracy will increase with more samples collected<sup>2</sup>. Here, photonic systems would be advantageous – due to the fast timescale,

---

<sup>2</sup>Sometimes, it can be beneficial to stop sampling prematurely and make a decision now, potentially correcting later – anytime computing [217].

many samples can be accumulated quickly. The problem is how to make such systems sample from a Boltzmann distribution of our choice.

The sampling process described above is discrete, but many physical systems, ours included, are continuous. Luckily, it has been proven that an intuitive transfer of this procedure to the continuous domain would still provide correct samples [218]. The system is the following:

- each neuron has a membrane potential  $u_i(t)$  such that in the resting state, the probability density of a spike emission is

$$p_{\text{spike}}(u_i) = \tau_{\text{ref}}^{-1} \exp(u_i), \quad (5.11)$$

- if a neuron has spiked at time  $t$ , it cannot spike until  $t + \tau_{\text{ref}}$ , i.e. it is in the refractory regime,
- neuron connectivity is similar to the discrete case

$$u_i(t) = b_i + \sum W_{ij} z_j(t), \quad (5.12)$$

where  $z_j(t) = 1$  if a  $j$ -th neuron is in the refractory regime and 0 otherwise.

Later, we will refer to these neurons as abstract continuous spiking neurons (ACSNs). The conversion process from continuous  $z_i(t)$  to discrete is shown in Figure 5.5.

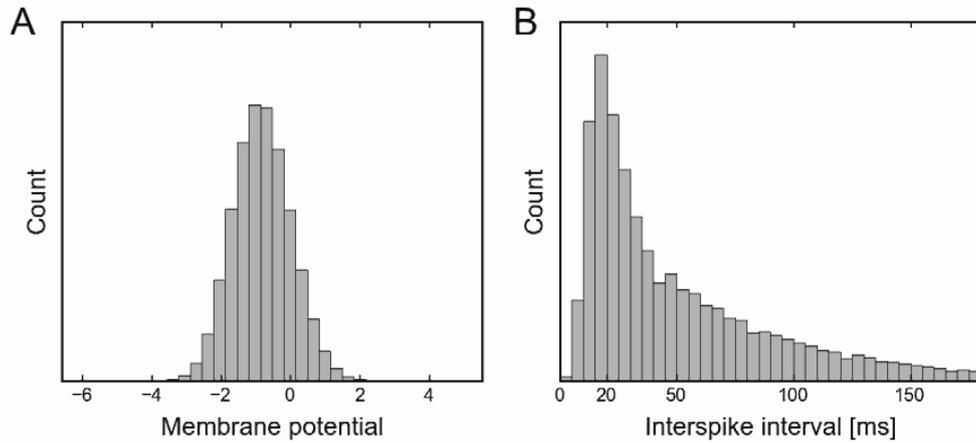


Figure 5.4: Statistical properties of an abstract continuous spiking neuron [218].

To determine the accuracy of sampling, the Kullback-Leibler divergence is used [220]

$$D_{\text{KL}}(P \parallel Q) = \sum_{x \in X} P(x) \log \left( \frac{Q(x)}{P(x)} \right), \quad (5.13)$$

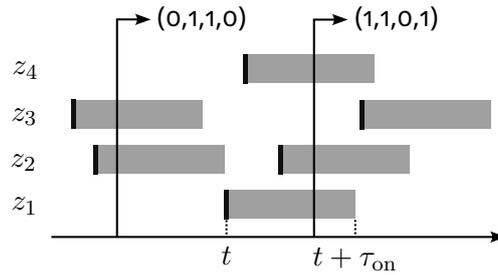


Figure 5.5: Sampling of four continuous spiking neurons. Black bars show spiking events and gray rectangles - neuron activation interval (i.e. when  $z_i = 1$ ). Reproduced based on [219].

which is a measure of how a probability distribution  $P(x)$  is different from a reference probability distribution  $Q(x)$  defined on a sample space  $X$ . To give an example, Figure 5.6 provides a visual comparison between sampled and target distributions and corresponding  $D_{KL}$  values.

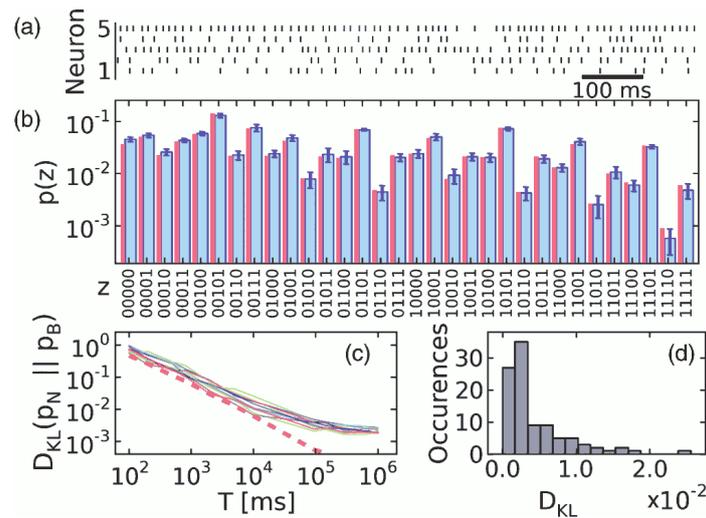


Figure 5.6: Sampling from a Boltzmann Machine with a network of leaky integrate-and-fire neurons. (a) Spike pattern of 5 such neurons during sampling. (b) Comparison of sampled distribution (blue) and target distribution (red). (c) Increase of sampling accuracy with more samples collected, ten trials. (d) Sampling accuracy for a set of random Boltzmann Machines with  $10^6$  ms of sampling time. Figure is taken from [219].

### 5.1.1 . Stochastic inference

Consider a two-dimensional pixelated image on which a digit is drawn. Each pixel is associated with a neuron of a BM (black pixel – neuron is active and vice-versa). The connections are trained such that the BM has minima in three states corresponding to three digits: 0, 3 and 4. If no input is imposed and the BM is updated, it randomly jumps between three states corresponding to these digits (see Figure 5.7, left). During exposure to a partial input – some pixels are forced to be in a particular state, which will impact the trajectory of the BM. The more pixels corresponding to a digit are forced, the more likely the more likely the BM stays in a corresponding state. However, if the input is ambiguous, i.e. two digits have share some pixels, a BM will not know for sure what digit we have meant, and it will jump between the corresponding states. For example, 3 and 4 have black pixels in the middle, while 0 does not (see Figure 5.7, right). The BM then jumps between 3 and 4, but not 0.

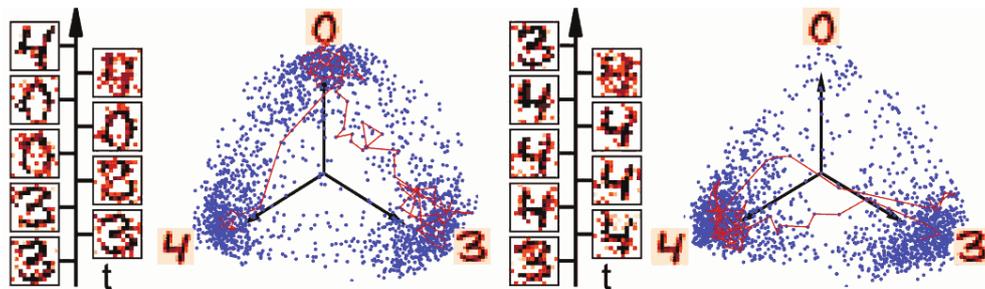


Figure 5.7: Stochastic inference of three digits from an image with leaky integrate-and-fire neurons. Left: when no input is imposed, the Boltzmann Machine infers three digits with equal probability. Right: when the it is exposed to an ambiguous input. Adapted from [219].

### 5.1.2 . Sampling from arbitrary distributions

Consider a BM that corresponds to a probability distribution  $p(z)$ . However, its marginal distribution, w.r.t. to some  $z$  components, can be arbitrary. Therefore, a system that samples from a BM can sample from all distributions.

In practice, it is carried out by separating BM neurons into two groups, visible  $v$  and hidden  $h$ ; intra-group connections are removed. This is a Restricted BM (RBM). Then,  $p(v, h)$  is in accord with the Boltzmann distribution, but

$$p(v) = \sum_h p(v, h) \quad (5.14)$$

might not. Then  $p(v, h)$  can be trained such that  $p(v)$  is close to the target distribution (which might not even have a closed form and be in the form of a dataset), for example, using the ‘wake-sleep’ algorithm [221]:

1. wake: force visible neurons to a state from the target distribution, run the BM and compute  $\langle v_i h_j \rangle_{\text{wake}}$
2. sleep: all neurons are free, run the BM and compute  $\langle v_i h_j \rangle_{\text{sleep}}$
3. update weights according to  $w_{ij} \sim \langle v_i h_j \rangle_{\text{wake}} - \langle v_i h_j \rangle_{\text{sleep}}$ .

To conclude, sampling from a BM allows for solving several tasks. A network of ACSNs can sample accurately from a BM [218]. The sampling precision depends on the number of collected samples, which is dictated by the network's timescale. Therefore, we aim to approximate the ACSN with our spiking photonic network.

## 5.2 . Single laser model

The spiking behaviour of nanolasers with saturable absorbers has been demonstrated several times. The Yamada model [222] is often used to describe their behaviour. However, in the system we are interested in, noise plays an integral role, and, therefore, the stochastic model from [223] is used as a basis with recent corrections [224, 225]. The model including noise is discussed in a Ph.D. thesis of M. Delmulle [214]; here, only the general ideas are given.

Consider a laser composed of a gain and a saturable absorber sections, parameters of which are marked with 'g' and 'sa' respectively (later, this laser is referred to as a photonic spiking nanolaser or PSN). Together, they compose a single cavity, where a resonator mode is born. The mode is split between the two sections (see Figure 5.8), characterized by the field overlap factor:

$$\mathcal{P}_i = \int_i |E|^2 dx / \int |E|^2 dx \quad (5.15)$$

where  $i \in \{g, sa\}$  represents the integration region.

The gain  $G$  is added by quantum wells [226]:

$$G_{\text{QW}}(N) = G' N_{\text{tr}} \log \frac{N}{N_{\text{tr}}}, \quad (5.16)$$

where  $G' = dG_{\text{QW}}/dN$  is the differential gain at transparency,  $N$  is the electron density and  $N_{\text{tr}}$  is the electron density at transparency (see Figure 5.9). The differential gain of sections will differ due to the different overlap factors and the fact that only the gain region is pumped:

$$\begin{aligned} G'_g &= \mathcal{P}_g G', \\ G'_{sa} &= \mathcal{P}_{sa} G' \chi_g. \end{aligned} \quad (5.17)$$

Since we assume an incoherent PSN model, it can be described with three variables: a number of photons in the cavity  $S$  and a density of charge carriers

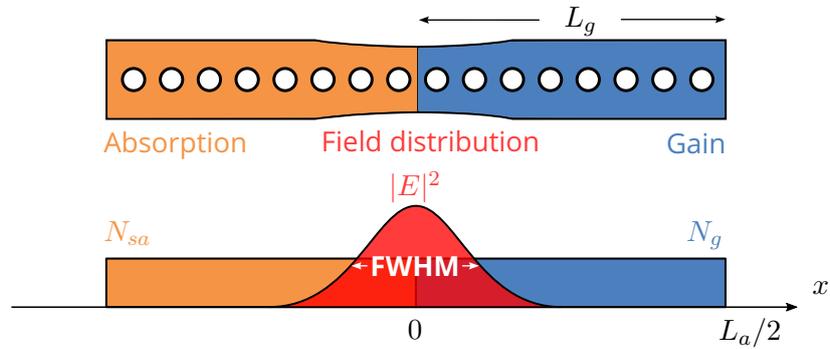


Figure 5.8: Resonator mode over gain and saturable absorption sections. The Gaussian mode shape in the middle of a cavity with a given mode size is assumed. Here, FWHM is full width at half maximum. Adapted from [214].

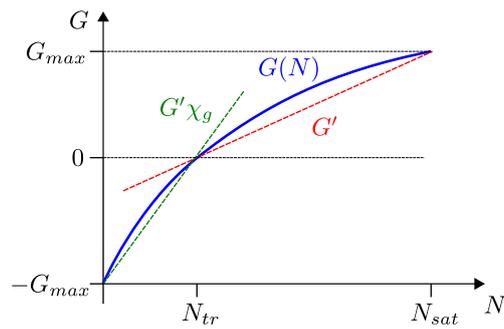


Figure 5.9: Quantum well gain curve, the differential gain slowly reduces as more charge carriers are added to a cavity [214].

in gain and absorber sections  $N_g$  and  $N_{sa}$ , respectively. They are normalized as follows:

- time to photon lifetime  $t \rightarrow t\gamma$ ,
- number of photons  $S$  w.r.t. the number of photons saturating the gain  $S_{sat} = N_{tr}V_g$ , where  $V_g$  is the gain section volume,
- carrier densities w.r.t. the carrier densities at transparency:  
 $N_g \rightarrow D = N_g/N_{tr} - 1$  and  $N_{sa} \rightarrow Q = N_{sa}/N_{tr} - 1$ ,

then:

$$\begin{aligned}\frac{dP}{dt} &= (\eta_g D + \eta_{sa} Q - 1)P + \frac{\eta_g}{2S_{sat}}(D + 1) + \frac{\eta_{sa}}{2S_{sat}}(Q + 1) + \mathcal{F}_{S,g} + \mathcal{F}_{S,sa}, \\ \frac{dD}{dt} &= \mathcal{I} - \gamma_g(D + 1) - \eta_g DP + \mathcal{F}_{N,g}, \\ \frac{dQ}{dt} &= -\gamma_{sa}(Q + 1) - \eta_{sa}\chi QP + \mathcal{F}_{N,sa},\end{aligned}\tag{5.18}$$

where  $\eta_g = \Gamma_g G' N_{tr}/\gamma$  and  $\eta_{sa} = (1 - \Gamma_g)G' N_{tr}/\gamma$  are the normalized gain in gain and absorber regions,

$$\mathcal{I} = \frac{I}{qN_{tr}V_g\gamma}\tag{5.19}$$

and  $I$  are the normalized and the regular injection currents,  $q$  is the elementary charge,  $\gamma_i = (\gamma_{nr,i} + 0.5G'/V_a)/\gamma$  are the total electron recombination rate where  $i \in \{g, sa\}$ ,

$$\gamma_{nr,i} = \frac{1}{\tau_{nr,i}} + \frac{1}{2V_a}G'_i \left( \frac{1}{\beta^{se}} - 1 \right)\tag{5.20}$$

is the electron recombination rate not associated with the cavity mode and  $\beta^{se}$  is the spontaneous emission factor. The total gain is designated as

$$G = \eta_g D + \eta_{sa} Q - 1.\tag{5.21}$$

Pump saturation adds a multiplier to its term:

$$\mathcal{I} \rightarrow \mathcal{I} \left( 1 - \frac{D + 1}{\chi_g + 1} \right).\tag{5.22}$$

where  $\chi_g$  is a ratio of differential gain between gain and absorption regions (see Figure 5.9).

The Langevin forces are formulated with a finite time step  $\Delta t$  and are integrated at a bandwidth  $1/\Delta t$ , then

$$\begin{aligned}\mathcal{F}_{S,i} &= \sqrt{\frac{2\mathcal{D}_{SS,i}}{\Delta t}}\zeta_{S,i} \\ \mathcal{F}_{N,i} &= -\sqrt{\frac{2\mathcal{D}_{SS,i}}{\Delta t}}\zeta_{S,i} + \sqrt{\frac{2\gamma_{nr,i}N_i}{\Delta t}}\zeta_{N,i}\end{aligned}\tag{5.23}$$

where  $i \in \{g, sa\}$ ,  $\zeta_{j,i}(t)$  are independent Gaussian processes with zero mean and unitary variance and

$$\begin{aligned} \mathcal{D}_{SS,g} &= \frac{\eta_g}{2S_{sat}} P(D + 1), \\ \mathcal{D}_{SS,sa} &= \frac{\eta_{sa}}{2S_{sat}} P(Q + 1). \end{aligned} \tag{5.24}$$

It was pointed out that the Langevin model here has issues [227], such as the problem of granularity of light highlighted by [228]. The discrete nature of the process explains the jitter in the pulse timing in a nanolaser and may not be captured by the continuous variable equations [229]. Future work will utilize an advanced discrete model, which accounts for the discrete nature of light.

### 5.3 . Spiking nanolaser as abstract spiking neuron

First, consider a single PSN. The goal is to make it behave like a ACSN, i.e. we should

- figure out if PSNs can be refractory like ACSNs and, if so, determine  $\tau_{on}$ ,
- define a membrane potential that behaves identically to one of ACSN.

#### 5.3.1 . Spiking demonstration

Eq. (5.18) is integrated with the *SROCK2* algorithm available in *DifferentialEquations.jl* library [145]. This is a weak second-order and strong first-order fixed step stabilized method for stiff Ito problems that can handle one-dimensional, diagonal and non-diagonal noise. Cavity parameters are chosen close to those of a recently fabricated structure presented in [214]; see Table 5.1.

First, pumping  $\mathcal{I}$  is increased until the PSN starts self-pulsing. Figure 5.10 shows the maximum number of photons w.r.t.  $\mathcal{I}$ . There is a sharp increase after a certain threshold – a sign of spiking. Figure 5.11 shows the dynamics of gain and photon number above the threshold visible in Figure 5.10. One could see that the emission of a short spike causes a drop in gain that takes a considerably long time to recover; after recovery, gain performs a random walk due to noise.

Figure 5.12 shows the stochastic properties of a PSN in more detail. When the PSN is not spiking, it was found that the gain autocorrelation is similar to the one of the Ornstein-Uhlenbeck process, i.e. gain is primarily a result of first-order low-pass filtering of a white gaussian noise, which means that PSN dynamics are largely dominated by noise. Above the self-spiking threshold, this autocorrelation is deformed. Looking at the gain histogram, one can see that the gain distribution below the threshold is symmetric. This matches the behaviour of ACSN's membrane potential (see Figure 5.4, A). Above the

Table 5.1: Simulation parameters for a single cavity [214]. See Figure 5.8 for cavity and mode size definitions.

Param.	Value	Description
FWHM	5 $\mu\text{m}$	FWHM of spatial mode profile
$L_g$	8.9 $\mu\text{m}$	Length of gain region
$L_a$	16 $\mu\text{m}$	Total cavity length (incl. carrier diffusion)
$h_{\text{wg}}$	0.6 $\mu\text{m}$	Cavity thickness
$w_{\text{wg}}$	0.65 $\mu\text{m}$	Cavity width
$G'$	$4.6 \cdot 10^{-13} \text{ m}^3/\text{s}$	Differential gain
$N_{\text{tr}}$	$1.85 \cdot 10^{18} \text{ cm}^{-3}$	Electron density at transparency
$\chi_g$	3	Differential gain ratio
$\tau_{\text{nr,g}}$	4 ns	Nonrad. electron lifetime in gain region
$\tau_{\text{nr,sa}}$	0.5 ns	Nonrad. electron lifetime in absorber
$\beta^{\text{se}}$	0.1	Spontaneous emission factor
$\gamma$	200 GHz	Optical loss rate

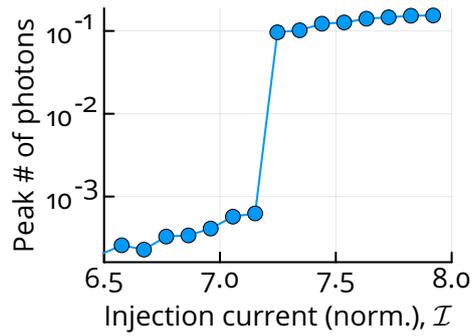


Figure 5.10: Bifurcation diagram of a photonic spiking neuron w.r.t. to injection power.

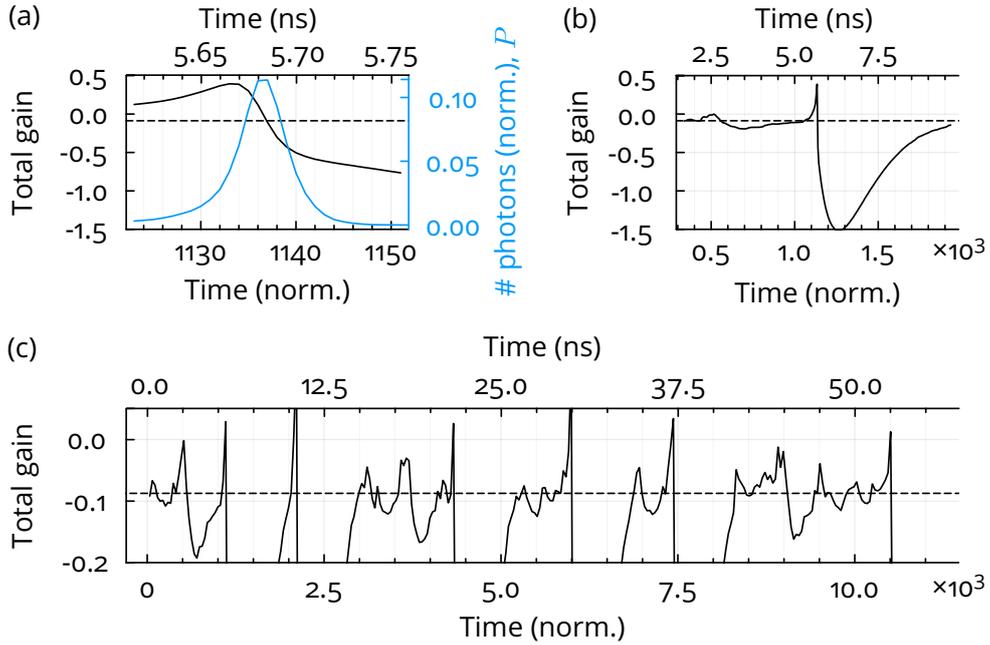


Figure 5.11: Spiking at  $\mathcal{I} = 7.5$ , dashed line shows  $G_{tgt}$ . (a) Interplay of gain and normalized number of photons during a spike emission. The number of photons corresponds to  $P$ . (b) Gain recovery after a large drop caused by a spike emission. (c) Random walk of gain between spikes. Note y-axis scaling.

threshold, the target gain is increased, and the gain spends some time in ‘recovery regions’. The interspike interval (ISI) distribution is roughly similar to the Gamma distribution, which matches the ACSN case (see Figure 5.4, B). This also proves that self-pulsing is not a periodic process, and noise plays a major role in the emission of spikes. Similarities between the ACSN’s membrane potential and PSN’s gain hint that they might be closely related.

In the case of ACSNs, spiking is defined as an intrinsically stochastic process, whereas for PSNs, the stochasticity is introduced through the gain. The gain cannot be used as a ‘membrane potential’ directly, as a real membrane potential is affected purely by bias, inputs and other neurons, whereas the gain is also affected by noise. Therefore, a proper substitute for the membrane potential must be found to make the analogy between ACSNs and PSNs.

The gain makes a random walk around an average value, estimated with the ‘target gain’

$$G_{tgt} = \eta_g D_0 + \eta_{sa} Q_0 - 1, \quad (5.25)$$

where  $D_0$  and  $Q_0$  are steady-state solutions of Eq. (5.18) without noise terms, which, unfortunately, can only be found numerically. The target gain is shown in Figure 5.11 and Figure 5.12, and one can find that it estimates the peak of the distribution, at least, as long as the gain still does the random walk.

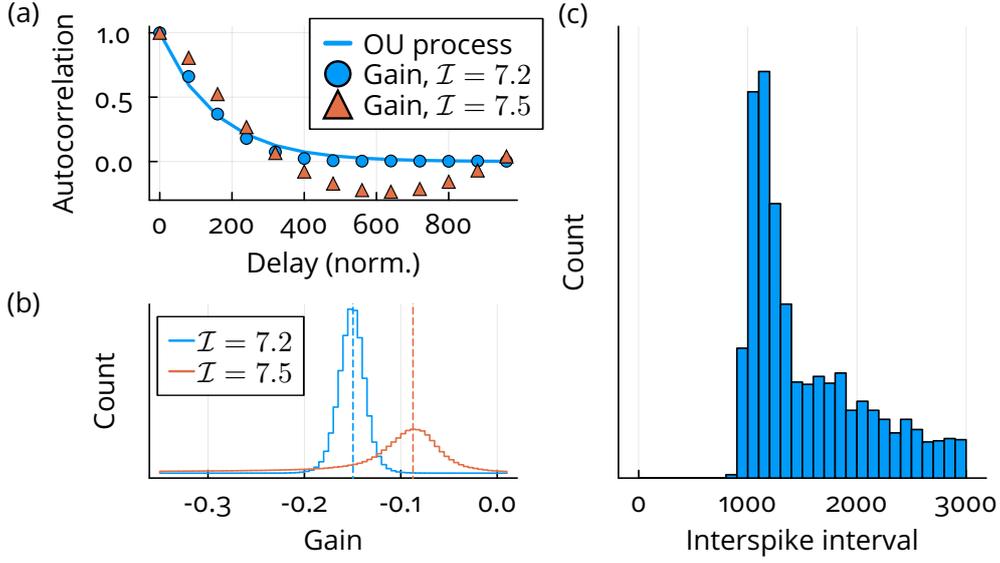


Figure 5.12: Stochastic properties of a photonic spiking neuron. (a) Autocorrelation of gain time traces below and above the self-spiking threshold ( $\mathcal{I} = 7.2$  and  $7.5$ ) and of the Ornstein-Uhlenbeck (OU) process. (b) Histograms of gain values, dashed line shows  $G_{\text{tgt}}$ . (c) Histogram of interspike interval at  $\mathcal{I} = 7.5$ .

The target gain is proposed to be used as PSN's membrane potential:

$$u_G = \alpha(G_{\text{tgt}} - G_{\text{tgt},0}), \quad (5.26)$$

where  $G_{\text{tgt},0}$  is target gain at which  $u_G = 0$ , where a PSN spends half the time in the refractory state, and  $\alpha$  is a proportionality coefficient.

In order to check the validity of our choice of  $u_G$  we need to verify that the activation curve resembles the sigmoid

$$p(z = 1) \approx \sigma(u_G) = \sigma(\alpha(G_{\text{tgt}} - G_{\text{tgt},0})), \quad (5.27)$$

which is true for ACSNs  $p(z = 1) = \sigma(u)$  [218].

### 5.3.2 . Activation curve of photonic spiking neurons

We need to verify if Eq. (5.27) holds. First, the pumping rate is swept, and  $G_{\text{tgt}}$  is computed. Then, the minimum ISI encountered is chosen as the provisional  $\tau_{\text{on}}$ , and an estimate of a  $p(z = 1)(u_G)$  curve is computed, which is then fit with a sigmoid according to Eq. (5.27). Figure 5.13 shows that although data points resemble a sigmoid, the fit quality is poor. To better understand why, let us consider gain time traces at high  $G_{\text{tgt}}$  shown in Figure 5.13 (right). One can find that most of the time, the PSN emits spike trains, and its gain rarely does a random walk. Therefore, noise has little impact on PSN spiking, which reduces its stochasticity. Therefore,  $G_{\text{tgt}} \geq -0.07$  should be avoided.

A new set of data points is generated, and the following steps are performed:

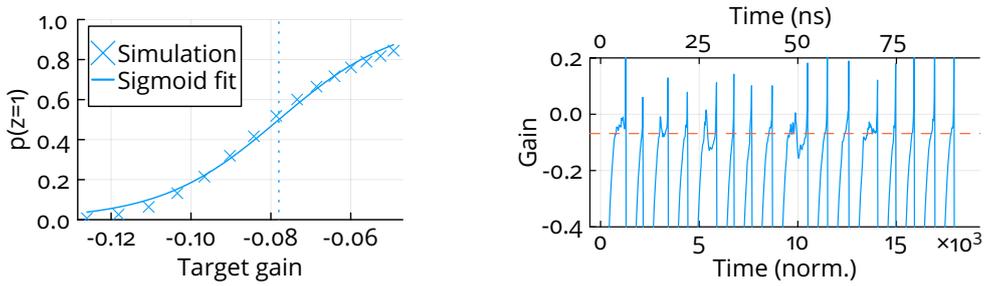


Figure 5.13: Left: sigmoid fit of a provisional PSN activation curve. A dotted line shows  $G_{\text{tgt},0}$ . Poor fit quality suggests that a more careful approach is required. Right: gain time trace with  $G_{\text{tgt}} \approx -0.07$  shown as a dashed line.

- Again, limit data points to fit. Points with too small  $G_{\text{tgt}}$  lead to rare spiking, and a long simulation is required to obtain a good sample size. For points with too high  $G_{\text{tgt}}$ , we may still have a stochasticity problem. Luckily, they can be omitted since there is no need for a wide range of  $G_{\text{tgt}}$ , as we will sample from BMs with  $b_k \in [-0.6, 0.6]$  [219]. However, due to coupling to other neurons,  $u_G$  may leave the  $[-0.6, 0.6]$  interval, so we anticipate  $b_k \in [-1.2, 1.2]$  shown in Figure 5.14.
- Optimize the duration a PSN is considered active ( $\tau_{\text{on}}$ ) instead of choosing the minimum ISI (see Figure 5.12). A higher value ( $820 \rightarrow 1000$ , i.e. 5 ns) was found to improve fitting.

These steps lead to a much better fit shown in Figure 5.14.

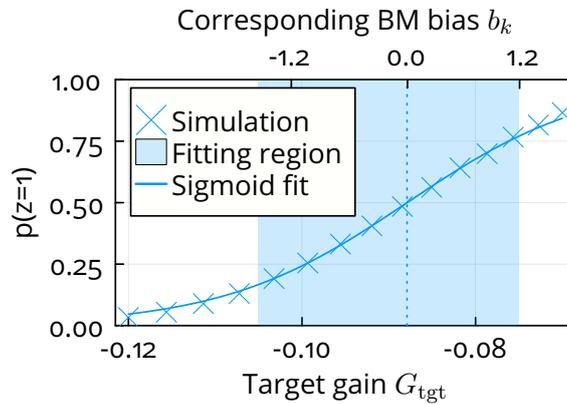


Figure 5.14: Sigmoid fit of an improved photonic spiking neuron activation curve inside the shaded region. A dotted line shows  $G_{\text{tgt},0}$ . Here BM is Boltzmann machine.

To conclude, a target gain is well-suited to be a ‘membrane potential’ of a PSN. Therefore, a PSN can be used for sampling of BMs.

### 5.3.3 . Refractory mechanism

As shown in Figure 5.11, after a spike is emitted, gain significantly drops, and a PSN is therefore unlikely to emit again if exposed to a new excitation soon after. In Figure 5.15 (left), we reduced pumping below the self-spiking threshold and then injected a single strong enough excitation pulse into the PSN such that it is guaranteed to induce a spike in a steady state. Then, the same excitation is sent again after some delay. Indeed, no spike was emitted again within a certain window of delay between the first and the second excitation pulses. There are two problems with this result:

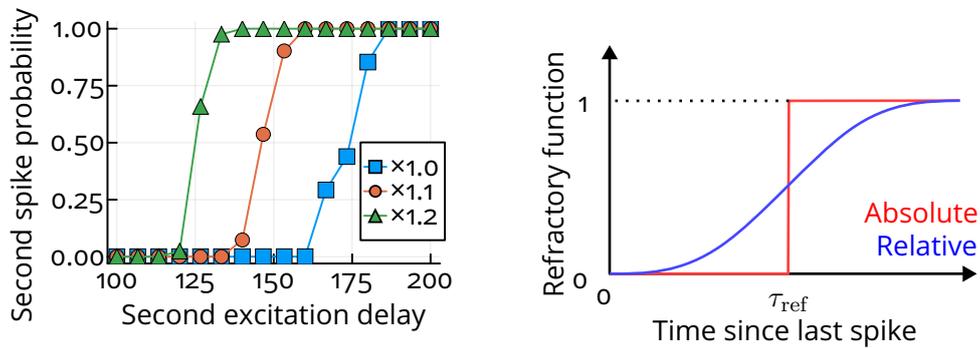


Figure 5.15: Probability of second spike emission w.r.t. delay between the first and the second excitations. The legend shows the power of the second excitation relative to the strength of the first. Right: examples of refractory functions of absolute and relative refractory mechanisms. Generally speaking, the curve of a relative mechanism can have an arbitrary shape.

- there is an interval of time when a second spike can be emitted with some probability,
- if the second excitation is stronger, a second spike can be emitted earlier.

The first problem is a case of the relative refractory mechanism as opposed to the absolute refractory mechanism used by ACSNs. This mechanism can be described by a refractory function  $g(t)$  such that  $p_{\text{spike}}(t) \rightarrow g(t)p_{\text{spike}}(t)$  and  $0 \leq g(t) \leq 1$  [218], an example of which is shown in Figure 5.15 (right).

Theoretically, correct BM sampling with ACSNs was rigorously proven for the absolute refractory mechanism [218] but, to the best of my knowledge, not the relative. Luckily, simulations show that, when handled properly, the relative refractory mechanism only slightly reduces the sampling accuracy [218] (see Figure 5.17).

The second problem is solved together with the first one the following way. As handling of the relative refractory mechanism is rather complicated [218], we will try to treat a PSN as if it had the absolute refractory mechanism. We

assume that the level of excitation is small enough not to modify the dynamics of gain recovery and, therefore, the probability of emitting a spike. This way, the refractory period will be primarily defined by static PSN parameters and bias. In the next section we find it together with  $G_{\text{tgt},0}$  and  $\alpha$ .

## 5.4 . Network of spiking nanolasers

In order to couple PSNs, we assume incoherent coupling via a balanced photodetector as was proposed in [212]. A spike is split in two by a controlled splitter and then sent to a pair of photodiodes, as was shown in Figure 5.2 (top-right). This way, by tuning the splitting ratio, the optical spike is converted into signed electrical current  $\sim wP$ , where  $w$  is a weight due to a controlled splitter. Then, this current is passed to the second PSN, which increases or decreases its charge carrier density.

### 5.4.1 . Postsynaptic potential

The shape of the  $D(t)$  curve after receiving a spike is the postsynaptic potential (PSP). In ACSNs, the PSP is  $z_k(t)$ ; by definition, it has a rectangular shape, whereas in PSNs, it does not.

Photodiodes have a finite bandwidth, and the sharp spike can be filtered to become a smooth electrical current<sup>3</sup>. The induced electrical current changes the charge carrier density inside a PSN, which then attenuates according to  $\gamma_g$  and related nonlinearities, resulting in further filtering.

Biological neurons also have a non-rectangular PSP. A release of neurotransmitters and their diffusion towards postsynaptic terminals happens quickly, but their removal from postsynaptic terminals may happen on a wide range of timescales [215]. A good kernel for such processes is a difference-of-exponentials

$$\text{PSP} \sim H(t) \left[ \exp\left(-\frac{t}{\tau_{\text{rise}}}\right) - \exp\left(-\frac{t}{\tau_{\text{fall}}}\right) \right], \quad (5.28)$$

its simplified version that assumes equality of both timescales – the alpha-kernel

$$\text{PSP} \sim H(t)t \exp\left(-\frac{t}{\tau_{\text{syn}}}\right). \quad (5.29)$$

If neurotransmitters diffuse much faster, one can consider a simple exponential kernel

$$\text{PSP} \sim H(t) \exp\left(-\frac{t}{\tau_{\text{syn}}}\right). \quad (5.30)$$

All three are shown in Figure 5.16, left.

---

<sup>3</sup>Although the photodiode bandwidth is assumed to be high enough

A non-rectangular PSP negatively impacts the accuracy of BM sampling (see Figure 5.17). Moreover, there is no definite rule for choosing their timescale. The general recommendation is that it is comparable to  $\tau_{\text{on}}$ , though other times could still lead to a good result [218]. In order to control a PSN's PSP timescale, we add a variable  $\mathcal{U}$  with a timescale  $\tau_{\mathcal{U}}$ . In that case, Eq. (5.18) needs to be updated:

$$\begin{aligned}\frac{dD}{dt} &= \frac{\mathcal{U}}{\tau_{\mathcal{U}}} - \gamma_{\text{g}}(D + 1) - \eta_{\text{g}}DP + \mathcal{F}_{N,\text{g}}, \\ \frac{d\mathcal{U}}{dt} &= -\frac{\mathcal{U}}{\tau_{\mathcal{U}}} + \mathcal{I} + \frac{d\mathcal{I}}{dP} \sum_{i \in \text{other PSNs}} w_i P_i,\end{aligned}\tag{5.31}$$

where  $d\mathcal{I}/dP$  is similar to a photodiode responsivity that incorporates photodiode parameters, the photon energy, among others and  $w_i$  is a tunable connection strength to an  $i$ -th PSN. As a result, upon a reception of a spike, a linear increase (or decrease) of gain is seen, followed by an exponential return to normal (Figure 5.16, right) reminiscent of the alpha-kernel.

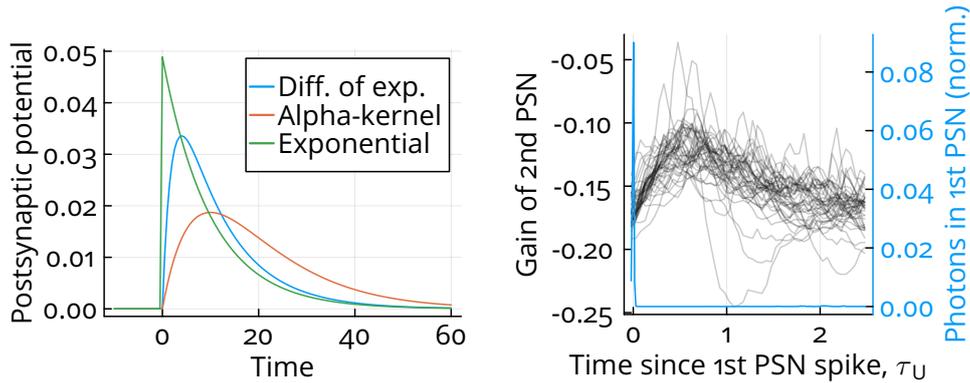


Figure 5.16: Left: various postsynaptic potential models used in literature normalized by the area under the curve. Right: photonic spiking neuron gain upon receiving a spike from another such neuron. Here, for clarity, the second neuron has  $\mathcal{I} = 7.1$ , which is relatively far below the self-spiking threshold so that it can endure stronger excitations without emitting a spike. The spike's impact on gain is weaker in actual computations.

#### 5.4.2 . Weight translation rule

The PSP timescale  $\tau_{\mathcal{U}}$  and translation rule between BM and PSN weights must be defined to determine PSN coupling.

Following [219], a linear mapping between BM and PSN weights is assumed. However, we need to clarify whether that leads to proper statistics. Consider a BM with five neurons with random biases  $\mathbf{b}$  and weights  $\hat{W}$ . We will scale its weights by a multiplier  $\xi_{\text{BM}}$ , which changes  $\langle z_i^{\text{BM}} \rangle$ , i.e. the proba-

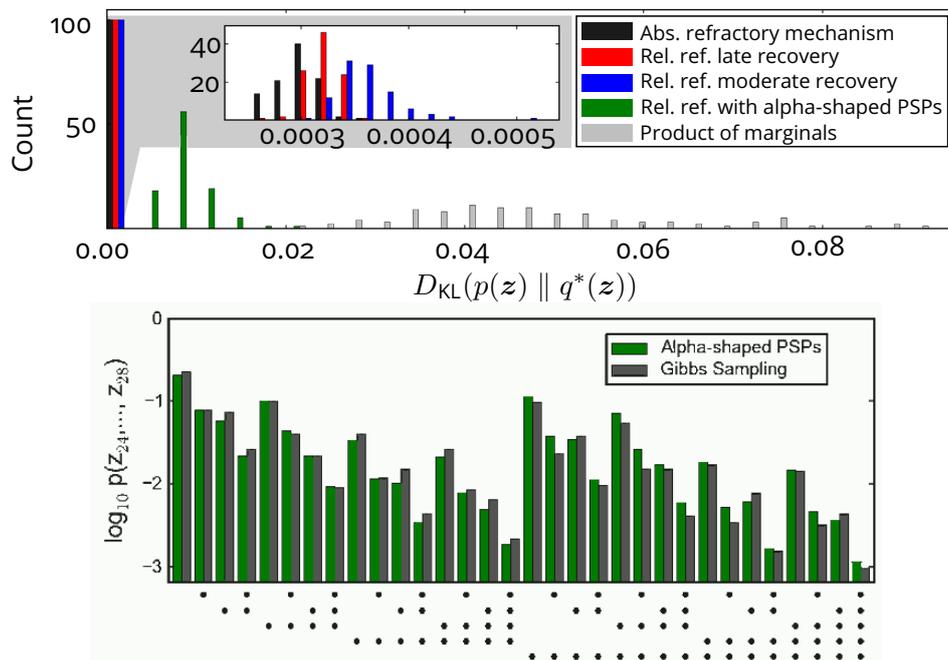


Figure 5.17: Impact of the relative refractory mechanism and non-rectangular post-synaptic potential on sampling accuracy. Top: impact in terms of the Kullback-Leibler divergence. Bottom: visual comparison of sampling accuracy with the alpha-shaped postsynaptic potential, rows of dots represent all possible states the networks can be in, and each dot is an active neuron. Adapted from [218].

bility of each BM neuron being active. By sweeping  $\xi_{\text{BM}}$ , each neuron gets a  $\langle z_i^{\text{BM}} \rangle(\xi_{\text{BM}})$  curve as a result.

The same is done for a network of 5 PSNs. Biases  $\mathbf{b}$  are translated into target gain according to the previous section; weights are chosen as  $\xi_{\text{PSN}} \hat{W}$ , and  $\xi_{\text{PSN}}$  is swept. This results in  $\langle z_i^{\text{PSN}} \rangle(\xi_{\text{PSN}})$  curves for each  $i$ -th PSN.

If BM neurons and PSNs are stochastically similar, for every  $i$ -th neuron  $\langle z_i^{\text{BM}} \rangle(\xi_{\text{BM}})$  and  $\langle z_i^{\text{PSN}} \rangle(\xi_{\text{PSN}})$  should have a similar shape.

As for the PSP timescale  $\tau_{\mathcal{U}}$ , as mentioned before, there is no general rule to determine its precise value. We then proceed to sweep  $\tau_{\mathcal{U}}$  and check the statistics of PSNs according to the previous paragraph. The result is shown in Figure 5.18. It was found that there is little difference past  $\tau_{\mathcal{U}} > 0.2\tau_{\text{on}}$  except at strong weights. We chose  $\tau_{\mathcal{U}} = 0.4\tau_{\text{on}}$  as it behaves well even in those regions.

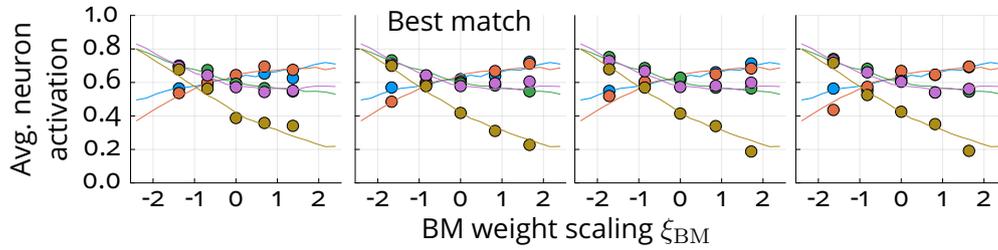


Figure 5.18: Behaviour of photonic spiking neurons (dots) and Boltzmann Machine neurons (lines) w.r.t. global scaling of their connection weights for various  $\tau_{\mathcal{U}}/\tau_{\text{on}}$ , from left to right: 0.2, 0.4, 0.8 and 1.6. Corresponding neurons from each system are shown with a unique color.

Now, we can find a weight proportionality coefficient. Consider only two BM neurons connected unidirectionally. The first neuron has a bias of 1 and acts as an excitation for the second neuron. The connection strength is swept in the  $[-1.2, 1.2]$  interval, and the average activation of the second neuron is tracked. Such sweep interval is chosen following [219] we sampling from BMs with  $W_{ij} \in [-1.2, 1.2]$  was performed. As a result, two curves,  $\langle z_i^{\text{BM}} \rangle(\xi_{\text{BM}})$  and  $\langle z_i^{\text{PSN}} \rangle(\xi_{\text{PSN}})$ , are obtained. Finally, we introduce a proportionality coefficient  $\zeta$  such that

$$\forall i \langle z_i^{\text{BM}} \rangle(\xi_{\text{BM}}) \approx \langle z_i^{\text{PSN}} \rangle(\zeta \xi_{\text{PSN}}), \quad (5.32)$$

which is then used as a weight transfer coefficient. The fitting result is shown in Figure 5.19.

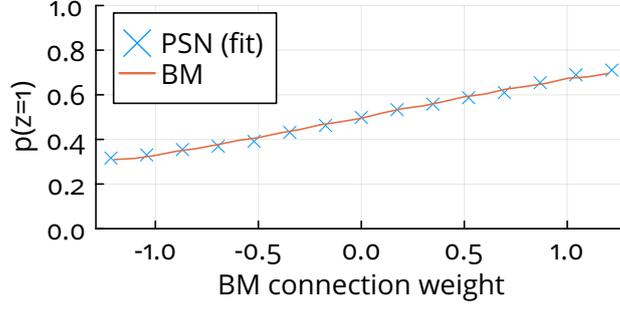


Figure 5.19: Comparing a Boltzmann Machine (BM) neuron and photonic spiking neuron (PSN) behaviour w.r.t. weight scaling. Here, PSN weights are scaled to BM weights according to Eq. (5.32).

### 5.5 . Photonic sampling

In this section, we perform sampling from a BM using our network of PSNs. We generate two random BMs with five neurons with moderate weights

$$W_{ij} \sim 1.2(\mathcal{B}(0.5, 0.5) - 0.5) \quad (5.33)$$

and large weights

$$W_{ij} \sim 2.4(\mathcal{B}(0.5, 0.5) - 0.5). \quad (5.34)$$

where  $\mathcal{B}(\alpha, \beta)$  is the Beta distribution shown in Figure 5.20. In both cases, biases are chosen according to

$$b_i \sim 1.2(\mathcal{B}(0.5, 0.5) - 0.5). \quad (5.35)$$

Then, these weights and biases are transferred into PSN parameters according to the previously established rules. After integrating the nonlinear equations of PSNs – Eq. (5.18) with corrections by Eq. (5.31) – their spikes are sampled according to Figure 5.5, and the sampled distribution is computed by

$$p(\mathbf{z}_k) = N(\mathbf{z}_k) / \sum_{\mathbf{z}} N(\mathbf{z}), \quad (5.36)$$

where  $N(\mathbf{z})$  is the number of collected samples corresponding to  $\mathbf{z}$ . The comparison between true and sampled distributions is given in Figure 5.21. A Kullback-Leibler divergence of  $2 \cdot 10^{-3}$  for moderate and  $5 \cdot 10^{-3}$  for large weights is achieved, which is in line with Figure 5.17 and Figure 5.6. Here,  $15 \cdot 10^3$  samples were collected, which took  $15 \cdot 10^6$  in normalized time or  $75 \mu\text{s}$  in physical device units, which corresponds to 200 million samples per second or 0.2 GHz, although after  $3 \cdot 10^3$  samples, the accuracy did not improve much.

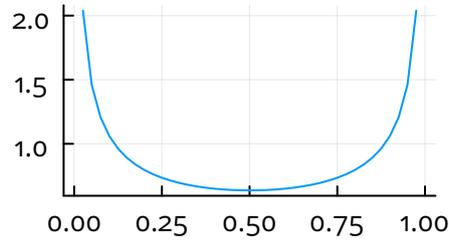


Figure 5.20: Probability density function of the Beta-distribution  $\mathcal{B}(0.5, 0.5)$ .

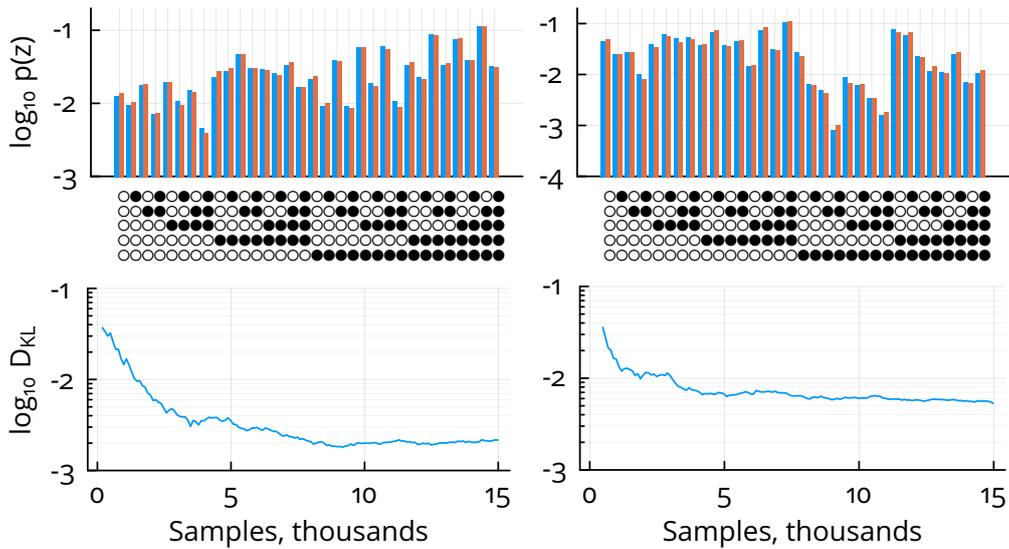


Figure 5.21: Sampling of a Boltzmann Machine with 5 neurons with a network of 5 photonic spiking neurons. Here  $b_i \sim 1.2[(0.5, 0.5) - 0.5]$ . Left column:  $W_{ij} \sim 1.2[(0.5, 0.5) - 0.5]$ . Right column:  $W_{ij} \sim 2.4[(0.5, 0.5) - 0.5]$ . Top: visual comparison between sampled and true probability distributions. Here circle columns represent  $z$ : black dots correspond to  $z_k = 1$  and white otherwise. Bottom: sampling accuracy improvement with more samples collected.

## 5.6 . Conclusion

In this chapter, we have implemented a network of newly developed coupled spiking nanolasers and demonstrated that it is capable of approximating a Boltzmann machine. In order to achieve such a result, several conditions must be met. First, the activation function of a PSN needs to closely approximate a sigmoid, which requires a correct choice of an allowed injection current interval and an optimization of the time a PSN is considered active after having emitted a spike ( $\tau_{on}$ ). Second, the relative refractory mechanism must be considered by limiting injection current and PSN coupling strength. Then, weights of a BM can be linearly mapped to connection strength between PSNs and their bias to target gain, which, in turn, is almost linearly mapped to injection current. Finally, a network of PSNs can sample from a BM at high speed, provided the BM's parameters are limited within a specific interval similar to [219].

Such a device would be suited for power- and time-critical applications, for example, in 'edge computing' [230]. We also conjecture that such a system could be used to recover data in optical communication channels due to the properties outlined in Section 5.1 and header recognition.



## Conclusions and perspectives

Neural networks have found their place in numerous fields. The newly proposed applications, such as those proposed in Section 1, call for larger and faster neural networks, and digital electronics struggle to meet the demand due to intrinsic limitations of their architecture and the digital platform. As a result, all-optical neural networks are being pursued to utilize wide optical bandwidth and improve interconnectivity. In particular, the integrated photonic platform is attractive for such neural networks due to its compact device size and accessible nonlinearities. A large number of such implementations have been proposed; their overview is given in Section 2. However, there is yet to be a clear way to scale up neural networks on the integrated photonic platform.

This thesis proposes two ways towards large-scale integrated photonic neural networks by demonstrating that the evanescent coupling is a viable method of coupling nonlinear cavities to perform RC and that spiking nanolasers based on a two-section design with a saturable absorber could be used to make stochastic neural networks.

In Section 3, we propose a new approach to all-optical integrated RC using nonlinear resonators through evanescent coupling, which opens an opportunity for very dense neuron positioning. This concept raises their surface density, depending on technology, up to thousands of neurons per square millimeter. We developed a model based on the coupled mode theory and implemented it in the *Julia* programming language. We analyse the RC in a general manner, employing dimensionality and consistency as performance metrics. We found that the spectral properties of an RC are a major deciding factor in dimensionality, and we linked it to basic RC parameters.

We considered three nonlinearities common in the integrated photonic platform: TPA, FCD and the Kerr effect. To analyse their impact, we use consistency as a metric alongside dimensionality. We found that TPA reduces dimensionality while retaining consistency, whereas FCD and the Kerr effect have the opposite effect. Therefore, for those two the common RC advice applies – that the input power should be optimized, whereas for TPA, such fine-tuning is not required.

We investigated our RC's scalability and found that as long as the RC does not extend too far from the input waveguide, the power necessary to induce a nonlinear response in the system scales almost linearly with the number of cavities. Overall, it has been demonstrated this system could be configured to optimize its general computational properties: dimensionality and consistency.

In Section 4.1, we use the proposed RC to predict the solution of the

Mackey-Glass equation, which we use as a model of a generic physical system. We demonstrated that the evanescent coupling and nonlinearities by themselves are not sufficient to solve this task. However, together, they boost prediction accuracy comparable to hardware implementations found in the literature, even though the performance of an abstract RC, echo state network, shows better results.

In Section 4.2, we tackle data recovery from a nonlinear fiber. This is a very interesting application for such a system since the advantages of the integrated photonic platform, our goals and requirements of this task align perfectly. We demonstrated that such RC could recover an OOK-encoded signal travelling through a 150 km metro link and a 16-QAM signal through a high-speed short link. Then, we artificially increased nonlinear distortion in a fiber and noted a visible constellation improvement after being processed by our RC in a nonlinear regime, again demonstrating that intrinsic nonlinearities of a semiconductor platform are sufficient to solve a task and nonlinearity of detectors is not required. In this sense, the studied system is all-optical and autonomous.

Section 5 takes a direction towards stochastic computing in photonics. We considered a photonic spiking neuron (PSN) composed of a two-section PhC nanolaser capable of emitting sharp optical spikes under strong enough injection. Such a system is interesting for a few reasons

- spiking is achieved autonomously,
- lasers can be easily coupled to a waveguide, which allows for networking, and integrated in a fairly large number on a photonic chip,
- in a suitable range of injection strength, ISI is random due to the presence of noise, and PSNs are therefore stochastic,
- a typical interspike interval is a few nanoseconds, which allows for considerably faster sampling compared to digital hardware.

All these properties have not yet been demonstrated at the same time [214]. We found that such a laser can be applied for sampling from BMs. We proposed a framework that maps BM parameters to PSN ones for high-speed sampling.

Future work on a proposed RC could overcome some of its limitations. For example, dimensionality limited by resonator losses could be overcome by using multiple coupled RCs that receive delayed versions of an input. Alternatively, several RCs could be cascaded to create a deep RC [231] or a similar architecture. Another approach could be to consider embedding active structures inside the RC to reduce losses. In this work, we assumed only a single mode per resonator, but using multiple resonator modes could improve the compute density. We recently received fabricated chips with the RC de-

sign outlined in Section 3.10. We plan their characterization and application for computing.

Future work on PSNs may include a more accurate discrete model of noise. Moreover, a successful sampling from a BM could be expanded upon by implementing stochastic inference and sampling from arbitrary probability distributions.



## Résumé en français

Un réseau neuronal est une approche de l'informatique inspirée par le cerveau permettant de résoudre des problèmes difficilement abordables avec les méthodes algorithmiques traditionnelles. En conséquence, les réseaux neuronaux ont trouvé leur place dans de nombreuses applications. Certaines de ces tâches nécessitent non seulement une grande puissance de calcul, mais exigent aussi une latence faible, ce qui a inspiré la recherche sur les réseaux neuronaux dans le domaine optique.

Ce travail porte sur un « réservoir computer » constitué de micro résonateurs optiques couplés de manière évanescente. Un « réservoir computer » ne requiert pas le contrôle de la force des connexions entre les « neurones » ; cela rend possible l'utilisation d'un arrangement extrêmement compact d'éléments optiques intégrés, sans que les tolérances de fabrication posent de problème. Nous montrons que cette architecture est prometteuse dans certaines applications, par exemple le traitement en temps réel de signaux optiques cohérents à grande vitesse.

Dans cette architecture, contrairement à la plupart des études de la littérature, les lignes à retard optiques sont supprimées, ce qui réduit considérablement l'encombrement de la puce. Un cadre mathématique est établi pour sa simulation et l'évaluation de ses propriétés intrinsèques.

Le réservoir est étudié d'une manière générale autant que possible. Pour ce faire, on utilise la « dimensionnalité » et la « cohérence » en tant que mesure de la performance indépendante du type de tâche de calcul. Nous avons montré que la dimensionnalité est fortement liée aux états propres d'un « réservoir », c'est à dire des resonances collectives de l'ensemble des cavités aussi appelés « supermodes ». Cela permet de mettre en relation directe les paramètres les plus importants, à savoir la force de couplage évanescent et le facteur de qualité (Q) des résonateurs. La cohérence devient importante quand les effets non linéaires cessent d'être négligeables. Nous avons étudié plusieurs effets courants sur la photonique intégrée : absorption à deux photons, dispersion d'électrons libre et l'effet Kerr. Nous avons constaté que même si les trois peuvent contribuer positivement aux performances informatiques, le premier réduit la dimensionnalité mais maintient la cohérence du réservoir tandis que deux autres ont un effet opposé. Cette analyse n'assume aucune géométrie de résonateur, ni matériel, par suite elle est applicable dans de nombreux cas. Enfin, nous avons introduit une procédure pour définir de manière quantitative le niveau de puissance du signal optique à partir duquel les effets non linéaires entrent en jeu. Afin de donner davantage de généralité à nos conclusions, la réponse du réservoir est systématiquement référencée à ce niveau de puissance.

Sa capacité de calcul est démontrée par la prédiction d'un système physique en forme de tâche de prédiction Mackey-Glass, une référence de réservoir commun, et la récupération de données à partir d'une fibre optique à une vitesse de plusieurs dizaines de gigahertz, une tâche très pertinente pour les télécommunications. Dans les deux cas, nous avons explicitement démontré l'impact positif de la non-linéarité intrinsèque sur la performance de calcul.

En outre, nous démontrons un prototype fabriqué sur une plate-forme de silicium sur isolant avec des micro-anneaux et des cristaux photoniques comme résonateurs. Des travaux expérimentaux sur ce prototype sont prévus dans l'avenir proche. Une piste pour le développement de réseaux photoniques intégrés à grande échelle est ainsi proposée.

Ce travail étudie également la possibilité d'utiliser les nanolasers excitables en tant que « neurones ». Par rapport au modèle très abstrait de « perceptron », à la base des réseaux de neurones, ce type de réseaux est un peu plus proche du cerveau, car l'« information » échangée est représentée par des signaux dépendant du temps, plus précisément par des impulsions. De ce fait, ces « neurones » sont aussi plus difficiles à utiliser que les réseaux de neurones « conventionnels ». D'autre part, ce type de neurone semble plus idoine à des tâches du type « event-driven computing », c'est-à-dire le calcul initié par un stimulus externe; dans le scénario du traitement local des signaux générés par les capteurs, cette approche serait en principe beaucoup plus efficace du point de vue de l'utilisation de l'énergie.

Nous avons pris en considération un réseau de « neurones » constitué de nanolasers excitables. Ces dispositifs ont été démontrés récemment [214] en exploitant les propriétés des cristaux photoniques. Ces lasers sont compacts et permettent un couplage plus facile à d'autres lasers grâce à son mode résonateur latéral. En conséquence, ces lasers ont un potentiel d'évolutivité. Nous étudions numériquement ses propriétés stochastiques et l'effet du couplage entre les neurones. Nous avons trouvé qu'un tel réseau peut imiter avec une haute précision une machine de Boltzmann. Parmi les applications possibles de ce type de « machine », on peut citer l'inférence stochastique et l'échantillonnage de distributions de probabilités arbitraires.

L'échelle de temps caractéristique d'un tel réseau est fixée par la durée de vie des électrons, de l'ordre d'une nanoseconde dans le cas de la technologie utilisée ici. De ce fait, un réseau de ces « neurones » serait capable de traiter des millions d'échantillons par seconde en temps réel, d'après nos simulations. En conclusion, un réseau neuronal à base de nanolasers excitables serait intéressant pour les applications où la latence et la consommation d'énergie seraient critiques.

## A - Side-coupling cavity and waveguide

If a cavity is weakly coupled to a waveguide (see Figure A.1), the CMT assumes a linear coupling:

$$\frac{da}{dt} = \left( i\omega - \frac{\gamma^e}{2} \right) a + \kappa_1 s_{1+} + \kappa_2 s_{2+}. \quad (\text{A.1})$$

This section aims to find  $\gamma^e$  and  $\kappa_{1,2}$  from the first principles. We assume that

1. the system is in a quasi-steady state, i.e. changes are slow compared to  $1/\omega$ ,
2. the EM wave in the waveguide is close to a plane wave,
3. the EM wave is monochromatic and is at resonance with the resonator, i.e.  $\Omega = \omega$ .

Consider an infinite waveguide in an infinite homogenous media. Inside it a monochromatic wave with an amplitude of unity propagates from  $z = -\infty$  to  $z = +\infty$ :

$$\begin{aligned} \mathbf{E}_0 &= \mathbf{e}_+(x, y) \exp(-i\beta z + i\Omega t) \\ \mathbf{H}_0 &= \mathbf{h}_+(x, y) \exp(-i\beta z + i\Omega t) \end{aligned} \quad (\text{A.2})$$

where  $\mathbf{e}_+$  and  $\mathbf{e}_+$  are wave profiles and  $\beta$  is a propagation constant.

Now we add a resonator with an unperturbed mode  $\mathbf{e}_r(x, y, z)$  (see Figure A.1). The permittivity changes  $\varepsilon(\mathbf{r}) \rightarrow \varepsilon(\mathbf{r}) + \Delta\varepsilon(\mathbf{r})$ , where  $\Delta\varepsilon(\mathbf{r}) = \varepsilon_r(\mathbf{r}) - \varepsilon(\mathbf{r})$  in a resonator. If this perturbation is weak, the EM field inside the waveguide changes linearly:

$$\begin{aligned} \mathbf{E} &= \left[ b_+(z)\mathbf{e}_+(x, y) + b_-(z)\mathbf{e}_-(x, y) + a\mathbf{e}_r(x, y, z) \right] \exp(i\Omega t) \\ \mathbf{H} &= \left[ b_+(z)\mathbf{h}_+(x, y) + b_-(z)\mathbf{h}_-(x, y) + a\mathbf{h}_r(x, y, z) \right] \exp(i\Omega t) \end{aligned} \quad (\text{A.3})$$

where  $\mathbf{e}_-$  and  $\mathbf{h}_-$  represent wave profile of a wave propagating in the opposite direction from  $z = +\infty$  to  $z = -\infty$  due to reflection induced by the resonator and  $b_+$ ,  $b_-$  and  $a$  are EM field amplitudes in the waveguide and the resonator. For brevity, coordinates of modes and amplitudes are omitted later. A mode amplitude reduces exponentially outside a resonator; therefore, we can assume that each cavity-waveguide interaction happens in a restricted region between two reference planes separated by  $d$  (dashed lines in Figure A.1, left) [144].

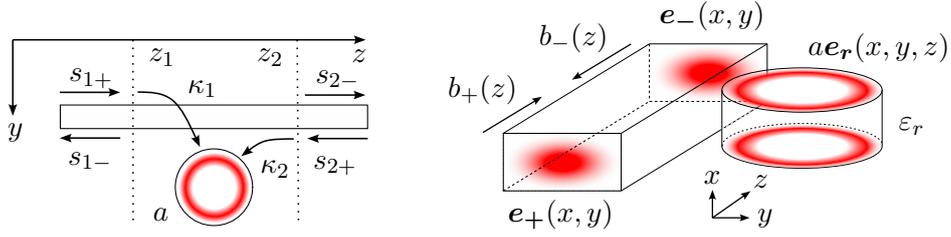


Figure A.1: Waveguide-resonator side-coupling from two perspectives. Vertical dotted lines in left figure show reference planes containing the waveguide-cavity interaction region.

### A.1 . Change of amplitude in space

Consider the Maxwell equations for the EM wave guided by the waveguide:

$$\begin{aligned}
\nabla \times \mathbf{E}_0 &= -\mu_0 \frac{\partial \mathbf{H}_0}{\partial t} = -i\omega\mu_0 \mathbf{H}_0, \\
\nabla \times \mathbf{H}_0 &= \varepsilon_0 \varepsilon \frac{\partial \mathbf{E}_0}{\partial t} = i\omega\varepsilon_0 \varepsilon \mathbf{E}_0, \\
\nabla \times \mathbf{E} &= -i\omega\mu_0 \mathbf{H}, \\
\nabla \times \mathbf{H} &= i\omega\varepsilon_0 (\varepsilon + \Delta\varepsilon) \mathbf{E},
\end{aligned} \tag{A.4}$$

where  $\varepsilon_0$  and  $\mu_0$  are the permittivity and the permeability of free space. Since the system is in a steady state, the derivative of  $a$  is negligibly small. Now consider an expression:

$$\begin{aligned}
&\nabla(\mathbf{H} \times \mathbf{E}_0^* + \mathbf{H}_0^* \times \mathbf{E}) = \\
&= \nabla \left[ (b_+ \mathbf{h}_+ + b_- \mathbf{h}_- + a \mathbf{h}_r) \times \mathbf{e}_+^* \exp(i\beta z) \right] + \\
&+ \nabla \left[ \mathbf{h}_+^* \exp(i\beta z) \times (b_+ \mathbf{e}_+ + b_- \mathbf{e}_- + a \mathbf{e}_r) \right],
\end{aligned} \tag{A.5}$$

which can also be rewritten differently:

$$\begin{aligned}
&\nabla(\mathbf{H} \times \mathbf{E}_0^* + \mathbf{H}_0^* \times \mathbf{E}) = \\
&= (\nabla \times \mathbf{H}) \mathbf{E}_0^* + \mathbf{H} (\nabla \times \mathbf{E}_0^*) + (\nabla \times \mathbf{H}_0^*) \mathbf{E} + \mathbf{H}_0^* (\nabla \times \mathbf{E}) = \\
&= i\omega [\varepsilon_0 (\varepsilon + \Delta\varepsilon) \mathbf{E} \mathbf{E}_0^* + \mu_0 \mathbf{H}_0^* \mathbf{H} - \varepsilon_0 \varepsilon \mathbf{E}_0^* \mathbf{E} - \mu_0 \mathbf{H} \mathbf{H}_0^*] = \\
&= i\omega\varepsilon_0 \exp(i\beta z) \Delta\varepsilon \mathbf{E} \mathbf{E}_0^* = \\
&= i\omega\varepsilon_0 \exp(i\beta z) \Delta\varepsilon [b_+ \mathbf{e}_+ + b_- \mathbf{e}_- + a \mathbf{e}_r] \mathbf{e}_+^* = \\
&= i\omega\varepsilon_0 \exp(i\beta z) \Delta\varepsilon (b_+ |\mathbf{e}_+|^2 + b_- \mathbf{e}_- \mathbf{e}_+^* + a \mathbf{e}_r \mathbf{e}_+^*)
\end{aligned} \tag{A.6}$$

We integrate both sides over the XY plane. For the left-hand side:

$$\begin{aligned} \iint_{XY} \nabla(\mathbf{H} \times \mathbf{E}_0^* + \mathbf{H}_0^* \times \mathbf{E}) dx dy &= \\ &= \iint_{XY} \nabla_{xy}(\mathbf{H} \times \mathbf{E}_0^* + \mathbf{H}_0^* \times \mathbf{E}) dx dy + \\ &\quad + \iint_{XY} \frac{\partial}{\partial z}(\mathbf{H} \times \mathbf{E}_0^* + \mathbf{H}_0^* \times \mathbf{E}) \hat{z} dx dy \quad (\text{A.7}) \end{aligned}$$

Here  $\nabla_{xy} = \partial_x \hat{x} + \partial_y \hat{y}$  and  $\hat{x}$ ,  $\hat{y}$ ,  $\hat{z}$  are unit vectors along x-, y- and z-axes. Because of the Gauss-Ostrogradsky theorem:

$$\iint_{XY} \nabla_{xy}(\mathbf{H} \times \mathbf{E}_0^* + \mathbf{H}_0^* \times \mathbf{E}) dx dy = \oint (\mathbf{H} \times \mathbf{E}_0^* + \mathbf{H}_0^* \times \mathbf{E}) dl, \quad (\text{A.8})$$

as the integration contour length scales  $(x^2 + y^2)^{1/2}$ , whereas the EM field exponentially decays outside the waveguide and the resonator. Therefore, at infinity, the contour integral is zero. We expand the second term of Eq. (A.7) to get

$$\begin{aligned} &= \frac{\partial}{\partial z} \exp(i\beta z) \iint_{XY} \left[ b_+(\mathbf{h}_+ \times \mathbf{e}_+^* + \mathbf{h}_+^* \times \mathbf{e}_+) + \right. \\ &\quad \left. + b_-(\mathbf{h}_- \times \mathbf{e}_+^* + \mathbf{h}_+^* \times \mathbf{e}_-) + a(\mathbf{h}_r \times \mathbf{e}_+^* + \mathbf{h}_+^* \times \mathbf{e}_r) \right] \hat{z} dx dy. \quad (\text{A.9}) \end{aligned}$$

Here, the second and third terms are zero because of waveguide and resonator mode orthogonality [137, p.190]

$$(\beta_i - \beta_j) \iint_{XY} (\mathbf{e}_i \times \mathbf{h}_j^* + \mathbf{e}_j^* \times \mathbf{h}_i) \hat{z} dx dy = 0. \quad (\text{A.10})$$

Then Eq. (A.9) is simply

$$= \frac{\partial}{\partial z} b_+ \exp(i\beta z) \iint_{XY} (\mathbf{h}_+ \times \mathbf{e}_+^* + \mathbf{h}_+^* \times \mathbf{e}_+) \hat{z} dx dy. \quad (\text{A.11})$$

If waveguide mode profiles are normalized to unit power

$$\frac{1}{4} \iint_{XY} (\mathbf{e}_+ \times \mathbf{h}_+^* + \mathbf{e}_+^* \times \mathbf{h}_+) dx dy = 1, \quad (\text{A.12})$$

then Eq. (A.11) gives us

$$= 4 \frac{\partial b_+}{\partial z} \exp(i\beta z) + 4i\beta b_+ \exp(i\beta z), \quad (\text{A.13})$$

which, combined with the right-hand side of Eq. (A.6), becomes

$$\begin{aligned} &= 4 \frac{\partial b_+}{\partial z} \exp(i\beta z) + 4i\beta b_+ \exp(i\beta z) = \\ &= i\omega \varepsilon_0 \exp(i\beta z) \iint_{XY} \Delta \varepsilon (b_+ |\mathbf{e}_+|^2 + b_- \mathbf{e}_- \mathbf{e}_+^* + a \mathbf{e}_r \mathbf{e}_+^*) dx dy. \quad (\text{A.14}) \end{aligned}$$

Splitting the integral, we get

$$\begin{aligned} \frac{\partial}{\partial z} b_+ = & -i\beta b_+ + i\frac{\omega\varepsilon_0}{4} \left[ b_+ \iint_{XY} \Delta\varepsilon |e_+|^2 dx dy + \right. \\ & \left. + b_- \iint_{XY} \Delta\varepsilon e_- e_+^* dx dy + a \iint_{XY} \Delta\varepsilon e_r e_+^* dx dy \right]. \end{aligned} \quad (\text{A.15})$$

Here, three integrals correspond to a change in the propagation constant, a coupling between waveguide modes and the waveguide-cavity coupling. We are interested in the last one.

## A.2 . Forward-propagating mode

According to the CMT assumption, wave amplitude changes linearly according to

$$\frac{\partial}{\partial z} b_+ = -i\beta b_+ + \kappa_+ a \quad (\text{A.16})$$

we get

$$\kappa_+ = i\frac{\omega\varepsilon_0}{4} \iint_{XY} \Delta\varepsilon e_r e_+^* dx dy. \quad (\text{A.17})$$

If we change the direction of the unperturbed wave and follow the same procedure, we get

$$\kappa_- = -i\frac{\omega\varepsilon_0}{4} \iint_{XY} \Delta\varepsilon e_r e_-^* dx dy. \quad (\text{A.18})$$

Following Manolatu's approach, we integrate Eq. (A.16) between the reference planes at  $z_1$  and  $z_2$ . First, we make  $b_+$  into a plane wave:

$$\frac{\partial}{\partial z} b_+ = -i\beta b_+ + \kappa_+ a \Leftrightarrow \frac{\partial}{\partial z} b_+ \exp(i\beta z) = \kappa_+ a \exp(i\beta z). \quad (\text{A.19})$$

Then, we start the integration:

$$\int_{z_1}^{z_2} \frac{\partial}{\partial z} b_+ \exp(i\beta z) dz = i\frac{\omega\varepsilon_0}{4} a \int_{z_1}^{z_2} \exp(i\beta z) dz \iint_{XY} \Delta\varepsilon e_r e_+^* dx dy. \quad (\text{A.20})$$

The left-hand side is easily integrated

$$\begin{aligned} b_+(z_2) \exp(i\beta z_2) - b_+(z_1) \exp(i\beta z_1) = \\ = i\frac{\omega\varepsilon_0}{4} a \int_{z_1}^{z_2} \exp(i\beta z) dz \iint_{XY} \Delta\varepsilon e_r e_+^* dx dy. \end{aligned} \quad (\text{A.21})$$

We factor out  $b_+(z_2)$  to derive the transfer function:

$$b_+(z_2) = b_+(z_1) \exp[-i\beta(z_2 - z_1)] + i \frac{\omega \varepsilon_0}{4} a \int_{z_1}^{z_2} \exp[i\beta(z - z_2)] dz \iint_{XY} \Delta \varepsilon \mathbf{e}_r \mathbf{e}_+^* dx dy \quad (\text{A.22})$$

Since waveguide mode profiles are normalized to unit power,  $b_+(z_1) = s_{1+}$  and  $b_+(z_2) = s_{2-}$ , then:

$$s_{2-} = \exp[-i\beta(z_2 - z_1)] \times \left[ s_{1+} + i \frac{\omega \varepsilon_0}{4} a \int_{z_1}^{z_2} \exp[i\beta(z - z_1)] dz \iint_{XY} \Delta \varepsilon \mathbf{e}_r \mathbf{e}_+^* dx dy \right]. \quad (\text{A.23})$$

Here, the terms in red differ from the ones in [144], but we continue nonetheless. We define

$$I_+ = i \frac{\omega \varepsilon_0}{4} \int_{z_1}^{z_2} \exp[i\beta(z - z_1)] dz \iint_{XY} \Delta \varepsilon \mathbf{e}_r \mathbf{e}_+^* dx dy, \quad (\text{A.24})$$

which describes the coupling strength of the waveguide to the resonator. Then

$$s_{2-} = \exp[-i\beta(z_2 - z_1)] [s_{1+} + a I_+]. \quad (\text{A.25})$$

### A.3 . Backward-propagating mode

We do the same for the backpropagating wave:

$$\frac{\partial}{\partial z} b_- \exp(-i\beta z) = \kappa_- a \exp(-i\beta z), \quad (\text{A.26})$$

$$\begin{aligned} \int_{z_1}^{z_2} \frac{\partial}{\partial z} b_- \exp(-i\beta z) dz &= \\ &= -i \frac{\omega \varepsilon_0}{4} a \int_{z_1}^{z_2} \exp(-i\beta z) dz \iint_{XY} \Delta \varepsilon \mathbf{e}_r \mathbf{e}_-^* dx dy, \quad (\text{A.27}) \end{aligned}$$

$$\begin{aligned} b_-(z_2) \exp(-i\beta z_2) - b_-(z_1) \exp(-i\beta z_1) &= \\ &= -i \frac{\omega \varepsilon_0}{4} a \int_{z_1}^{z_2} \exp(-i\beta z) dz \iint_{XY} \Delta \varepsilon \mathbf{e}_r \mathbf{e}_-^* dx dy, \quad (\text{A.28}) \end{aligned}$$

$$b_-(z_1) = b_-(z_2) \exp[-i\beta(z_2 - z_1)] + i\frac{\omega\varepsilon_0}{4}a \int_{z_1}^{z_2} \exp[-i\beta(z - z_1)]dz \iint_{XY} \Delta\varepsilon \mathbf{e}_r \mathbf{e}_-^* dx dy. \quad (\text{A.29})$$

Considering that  $b_-(z_1) = s_{1-}$  and  $b_-(z_2) = s_{2+}$ :

$$s_{1-} = \exp[-i\beta(z_2 - z_1)] \times \left[ s_{2+} + i\frac{\omega\varepsilon_0}{4}a \int_{z_1}^{z_2} \exp[-i\beta(z - z_2)]dz \iint_{XY} \Delta\varepsilon \mathbf{e}_r \mathbf{e}_-^* dx dy \right]. \quad (\text{A.30})$$

Again, the terms in red differ from [144]. Similarly, we define

$$I_- = i\frac{\omega\varepsilon_0}{4}a \int_{z_1}^{z_2} \exp[-i\beta(z - z_2)]dz \iint_{XY} \Delta\varepsilon \mathbf{e}_r \mathbf{e}_-^* dx dy. \quad (\text{A.31})$$

Finally, we arrive at a similar result

$$s_{1-} = \exp[-i\beta(z_2 - z_1)] \left[ s_{2+} + aI_- \right]. \quad (\text{A.32})$$

#### A.4 . Energy conservation

We found how the EM field in the waveguide changes due to a resonator. We can find the resonator mode behaviour by using the energy conservation law.

The flow of energy for a lossless cavity coupled to a waveguide:

$$\frac{d|a|^2}{dt} = |s_{1+}|^2 + |s_{2+}|^2 - |s_{1-}|^2 - |s_{2-}|^2. \quad (\text{A.33})$$

From Eq. (A.25) and Eq. (A.32) we can write

$$\begin{aligned} s_{2-}^* &= \exp[i\beta(z_2 - z_1)] \left[ s_{1+}^* + a^* I_+^* \right], \\ s_{1-}^* &= \exp[i\beta(z_2 - z_1)] \left[ s_{2+}^* + a^* I_-^* \right], \end{aligned} \quad (\text{A.34})$$

which leads to

$$\begin{aligned} |s_{2-}|^2 &= |s_{1+}|^2 + |aI_+|^2 + s_{1+}a^* I_+^* + s_{1+}^* a I_+, \\ |s_{1-}|^2 &= |s_{2+}|^2 + |aI_-|^2 + s_{2+}a^* I_-^* + s_{2+}^* a I_-. \end{aligned} \quad (\text{A.35})$$

Then Eq. (A.33) transforms into

$$\frac{d|a|^2}{dt} = -(|I_+|^2 + |I_-|^2)|a|^2 - s_{1+}a^* I_+^* - s_{1+}^* a I_+ - s_{2+}a^* I_-^* - s_{2+}^* a I_-. \quad (\text{A.36})$$

On the other hand, the CMT assumes a linear coupling to incoming EM fields:

$$\frac{da}{dt} = \left( i\omega - \frac{\gamma^e}{2} \right) a + \kappa_1 s_{1+} + \kappa_2 s_{2+}, \quad (\text{A.37})$$

where  $\gamma^e$  is loss into *both* directions of the waveguide. Then, the energy in the cavity follows the following equation

$$\frac{d|a|^2}{dt} = -\gamma^e |a|^2 + \kappa_1 s_{1+} a^* + \kappa_1^* s_{1+}^* a + \kappa_2 s_{2+} a^* + \kappa_2^* s_{2+}^* a. \quad (\text{A.38})$$

Both Eq. (A.36) and Eq. (A.38) must hold; therefore

$$\gamma^e = |I_+|^2 + |I_-|^2 = |\kappa_1|^2 + |\kappa_2|^2, \quad (\text{A.39})$$

$$\kappa_1 = -I_+^* = i \frac{\omega \varepsilon_0}{4} \int_{z_1}^{z_2} \exp[-i\beta(z - z_1)] dz \iint_{XY} \Delta \varepsilon \mathbf{e}_r^* \mathbf{e}_+ dx dy, \quad (\text{A.40})$$

$$\kappa_2 = -I_-^* = i \frac{\omega \varepsilon_0}{4} \int_{z_1}^{z_2} \exp[i\beta(z - z_2)] dz \iint_{XY} \Delta \varepsilon \mathbf{e}_r^* \mathbf{e}_- dx dy, \quad (\text{A.41})$$

With that, we can write:

$$\begin{aligned} s_{2-} &= \exp[-i\beta(z_2 - z_1)] [s_{1+} - \kappa_1^* a] = \exp(-i\beta d) [s_{1+} - \kappa_1^* a], \\ s_{1-} &= \exp[-i\beta(z_2 - z_1)] [s_{2+} - \kappa_2^* a] = \exp(-i\beta d) [s_{2+} - \kappa_2^* a], \end{aligned} \quad (\text{A.42})$$

Despite the differences in previous sections, these results match [144] – the paper probably has a typo.

Losses in the  $n$ -th direction of the waveguide can be defined as  $\gamma^{en}$  such that  $\gamma^{en} = |\kappa_n|^2$  or formally

$$\kappa_n = \sqrt{\gamma^{en}} \exp(i\theta_n). \quad (\text{A.43})$$

Then, for both directions

$$\gamma^e = \gamma^{e1} + \gamma^{e2} \quad (\text{A.44})$$

Assuming the symmetry of the waveguide-resonator system, coupling in both directions is equal.



## B - Waveguide coupling matrices

In Section A, we found how to describe waveguide-resonator interaction. What happens when we add more resonators alongside the first one, as in Figure B.1)? Assume that cavities are separated enough so that there is a distance  $L$  between areas defined by reference planes.

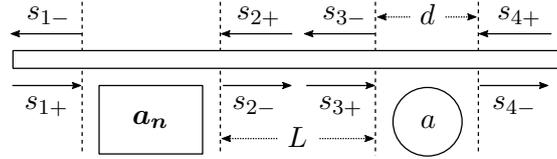


Figure B.1: Scheme of waveguide-cavity interaction. A rectangle with  $a_n$  represents a row of  $n$  cavities coupled to the waveguide.

For a single cavity coupled to a waveguide [144]:

$$\frac{da}{dt} = - \left( \frac{|\kappa_1|^2}{2} + \frac{|\kappa_2|^2}{2} \right) a + \kappa_1 s_{1+} + \kappa_2 s_{2+}, \quad (\text{B.1})$$

$$\begin{cases} s_{1-} = \exp(-i\beta d)(s_{2+} - \kappa_2^* a) \\ s_{2-} = \exp(-i\beta d)(s_{1+} - \kappa_1^* a) \end{cases} . \quad (\text{B.2})$$

Omitting waveguide-induced loss and generalizing for a row of  $n$  cavities coupled to one waveguide

$$\frac{d\mathbf{a}_n}{dt} = \hat{M}^n \mathbf{a}_n + \mathbf{R}_1^n s_{1+} + \mathbf{R}_2^n s_{2+}, \quad (\text{B.3})$$

$$\begin{pmatrix} s_{1-} \\ s_{2-} \end{pmatrix} = \begin{pmatrix} 0 & T_1^n \\ T_2^n & 0 \end{pmatrix} \begin{pmatrix} s_{1+} \\ s_{2+} \end{pmatrix} + \begin{pmatrix} Q_1^n \\ Q_2^n \end{pmatrix} \mathbf{a}_n, \quad (\text{B.4})$$

where  $\hat{M}^n$ ,  $\mathbf{R}_1^n$  and  $\mathbf{R}_2^n$  are assumed to be known. Adding a new cavity to the row

$$\frac{da}{dt} = \kappa_1 s_{3+} + \kappa_2 s_{4+}, \quad (\text{B.5})$$

$$\begin{cases} s_{3-} = \exp(-i\beta d)(s_{4+} - \kappa_2^* a) \\ s_{4-} = \exp(-i\beta d)(s_{3+} - \kappa_1^* a) \\ s_{3+} = \exp(-i\beta L)s_{2-} \\ s_{2+} = \exp(-i\beta L)s_{3-} \end{cases} . \quad (\text{B.6})$$

The goal is to find the matrix representation of a system with  $n + 1$  cavities based on a system with  $n$  cavities. We first eliminate the flux between the cavities

$$\begin{cases} s_{2+} = \exp(-i\beta L) \exp(-i\beta d)(s_{4+} - \kappa_2^* a) \\ s_{3+} = \exp(-i\beta L) (T_2^n s_{1+} + \mathbf{Q}_2^n \mathbf{a}_n) \end{cases}. \quad (\text{B.7})$$

Then, cavity dynamics can be rewritten using only new inputs  $s_{1+}$  and  $s_{4+}$ :

$$\begin{aligned} \frac{d\mathbf{a}_n}{dt} &= \hat{M}^n \mathbf{a}_n + \mathbf{R}_1^n s_{1+} + \exp(-i\beta L) \exp(-i\beta d) \mathbf{R}_2^n s_{4+} - \\ &\quad - \exp(-i\beta L) \exp(-i\beta d) \mathbf{R}_2^n \kappa_2^* a, \\ \frac{da}{dt} &= \exp(-i\beta L) T_2^n \kappa_1 s_{1+} + \exp(-i\beta L) \mathbf{Q}_2^n \kappa_1 \mathbf{a}_n + \kappa_2 s_{4+}, \end{aligned} \quad (\text{B.8})$$

Consequently, the new output on the first port is

$$\begin{aligned} s_{1-} &= T_1^n s_{2+} + \mathbf{Q}_1^n \mathbf{a}_n = \\ &= T_1^n \exp(-i\beta L) [\exp(-i\beta d)(s_{4+} - \kappa_2^* a)] + \mathbf{Q}_1^n \mathbf{a}_n \end{aligned} \quad (\text{B.9})$$

or, expanded

$$\begin{aligned} s_{1-} &= T_1^n \exp(-i\beta L) \exp(-i\beta d) s_{4+} + \mathbf{Q}_1^n \mathbf{a}_n - \\ &\quad - T_1^n \exp(-i\beta L) \exp(-i\beta d) \kappa_2^* a, \end{aligned} \quad (\text{B.10})$$

while on the fourth

$$\begin{aligned} s_{4-} &= \exp(-i\beta d) (\exp(-i\beta L) s_{2-} - \kappa_1^* a) = \\ &= \exp(-i\beta d) [\exp(-i\beta L) (T_2^n s_{1+} + \mathbf{Q}_2^n \mathbf{a}_n) - \kappa_1^* a], \end{aligned} \quad (\text{B.11})$$

which, again, expands to

$$\begin{aligned} s_{4-} &= \exp(-i\beta d) \exp(-i\beta L) T_2^n s_{1+} + \\ &\quad + \exp(-i\beta d) \exp(-i\beta L) \mathbf{Q}_2^n \mathbf{a}_n - \exp(-i\beta d) \kappa_1^* a. \end{aligned} \quad (\text{B.12})$$

Therefore, the differential equation and the output matrices can be built incrementally:

$$\begin{aligned} \frac{d}{dt} \begin{pmatrix} \mathbf{a}_n \\ a \end{pmatrix} &= \begin{pmatrix} \hat{M}^n & -\exp[-i\beta(L+d)] \mathbf{R}_2^n \kappa_2^* \\ \exp(-i\beta L) \mathbf{Q}_2^n \kappa_1 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{a}_n \\ a \end{pmatrix} + \\ &+ \begin{pmatrix} \mathbf{R}_1^n & \exp[-i\beta(L+d)] \mathbf{R}_2^n \\ \exp(-i\beta L) T_2^n \kappa_1 & \kappa_2 \end{pmatrix} \begin{pmatrix} s_{1+} \\ s_{4+} \end{pmatrix}, \end{aligned} \quad (\text{B.13})$$

$$\begin{aligned}
\begin{pmatrix} s_{1-} \\ s_{4-} \end{pmatrix} &= \begin{pmatrix} 0 & \exp[-i\beta(L+d)]T_1^n \\ \exp[-i\beta(L+d)]T_2^n & 0 \end{pmatrix} \begin{pmatrix} s_{1+} \\ s_{4+} \end{pmatrix} + \\
&+ \begin{pmatrix} \mathbf{Q}_1^n & -\exp[-i\beta(L+d)]T_1^n \kappa_2^* \\ \exp[-i\beta(L+d)]\mathbf{Q}_2^n & -\exp(-i\beta d)\kappa_1^* \end{pmatrix} \begin{pmatrix} \mathbf{a}_n \\ a \end{pmatrix}, \tag{B.14}
\end{aligned}$$

while matrices for a single cavity can be built from Section A.



## C - Simulation of optical signal propagation in an optical fiber

### C.1 . Split-step method

The split-step method is an approach to the numerical integration of partial differential equations. The main idea is to split an equation into a linear and nonlinear parts and treat the former in the Fourier domain and the latter in the time domain. We have a single first-order partial derivative in space Eq. (4.6), so we can discretize the time domain and perform integration steps in the space domain.

The motivation is that the linear part consists of derivatives that are computationally unstable operations. However, in the Fourier domain, they become an element-wise product of two vectors. Moreover, moving between time- and Fourier domains is fast with the help of the Fast Fourier Transform (FFT) algorithm, which has  $O(n \log(n))$  complexity in the worst-case scenario, where  $n$  is the number of points on the time grid.

### C.2 . Solver validation

In order to test solver correctness, we use it to replicate pulse propagation from a paper by J.M Dudley et al. [232] shown in Figure C.2. High-power short-length pulse dynamics are sensitive to a variety of effects, like high-order dispersion and other nonlinearities besides the Kerr effect; therefore, replication of such result is a suitable validation method.

#### C.2.1 . Recomputing dispersion coefficients

There are several ways to describe dispersion. One way is through Taylor coefficients of the group velocity dispersion (GVD) as was done in [232] (see Figure C.1). Another way is to provide a dispersion parameter curve  $D(\lambda)$  [188], which can be expanded into Taylor series. In this section, we connect these approaches.

The equation for electric field time-domain envelope, omitting linear absorption and nonlinearity, is [232]

$$\frac{\partial A}{\partial z} = \sum_{k \geq 2} \frac{i^{k+1}}{k!} \beta_k \frac{\partial^k A}{\partial t^k} \quad (\text{C.1})$$

We then apply an inverse Fourier transform over time

$$\tilde{A}(\omega, z) = \text{ifft}[A(t, z)]. \quad (\text{C.2})$$

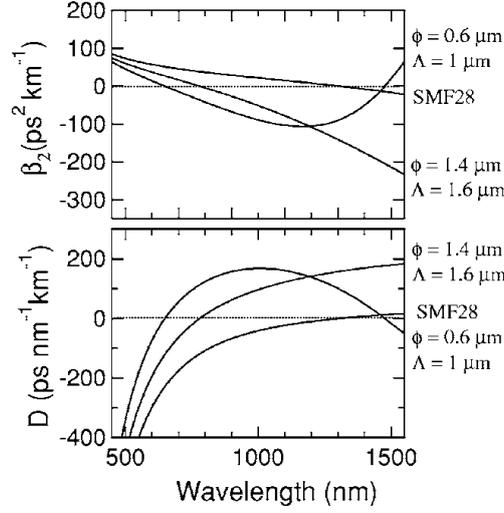


Figure C.1: Group velocity dispersion parameters for two photonic crystal fibers and SMF-28 [232].

where  $\text{ifft}(x)$  is the inverse FFT. Due to how  $A(t, z)$  is defined,  $\tilde{A}(\omega, z)$  is centered around the carrier frequency  $\omega_0$ . Therefore,  $\omega$  here is, effectively,  $\Delta\omega$ . Then

$$\partial^k A / \partial t^k \rightarrow (-i\Delta\omega)^k \tilde{A}. \quad (\text{C.3})$$

and

$$\frac{\partial \tilde{A}}{\partial z} = \sum_{k \geq 2} \frac{i^{k+1}}{k!} \beta_k (-i\Delta\omega)^k \tilde{A} = i \sum_{k \geq 2} \frac{\beta_k}{k!} (\Delta\omega)^k \tilde{A}. \quad (\text{C.4})$$

So, the summation here is a part of a Taylor series for  $\beta(\omega)$ :

$$\beta(\omega) - \beta_0 - \beta_1 \Delta\omega = \sum_{k \geq 2} \frac{\beta_k}{k!} (\Delta\omega)^k. \quad (\text{C.5})$$

How do we find  $\beta_k$ ?

Fiber manufacturers can provide the dispersion parameter

$$D(\lambda) = -2\pi c \beta_2(\lambda) / \lambda^2 \quad (\text{C.6})$$

for a range of wavelengths. Therefore, the dispersion of the group velocity dispersion is

$$\beta_2(\lambda) = -D(\lambda) \lambda^2 / 2\pi c, \quad (\text{C.7})$$

where  $\beta_2(\lambda)$  is a *function*, whereas  $\beta_2$  is a Fourier expansion term of  $\beta(\omega)$  around  $\omega_0$  i.e. a *constant* - to prevent confusion for  $\beta_2(\dots)$  we always write parentheses. We could use it in Eq. (C.1); alternatively, we can also consider a Taylor series of  $\beta_2$ :

$$\beta(\omega) - \beta_0 - \beta_1 \Delta\omega = \frac{\beta_2(\omega)}{2!} (\Delta\omega)^2 = \frac{1}{2!} \sum_{k=0} \frac{\beta_2^{(k)}}{k!} (\Delta\omega)^{k+2}, \quad (\text{C.8})$$

where  $\beta_2^{(k)}/k!$  coefficients can be found by using a numerical library for a function  $(-D(\lambda)\lambda^2/2\pi c) \circ \lambda(\Delta\omega)$ .

Here  $\beta_2^{(k)}$  are terms for  $\beta_2(\omega)$ . Sometimes, in literature,  $\beta_k$  coefficients are given instead. They are related via

$$\sum_{k=2} \frac{\beta_k}{k!} (\Delta\omega)^k = \frac{1}{2!} \sum_{k=2} \frac{\beta_2^{(k-2)}}{(k-2)!} (\Delta\omega)^k \Rightarrow \beta_{2+k} = \frac{(2+k)!}{2(k)!} \beta_2^k. \quad (\text{C.9})$$

### C.2.2 . Results

In Figure C.2, we reproduce [232, Figure 3], where a high-power pulse propagates through a photonic crystal fiber. Our solver closely resembles these results, which affirms the validity of our solver implementation.

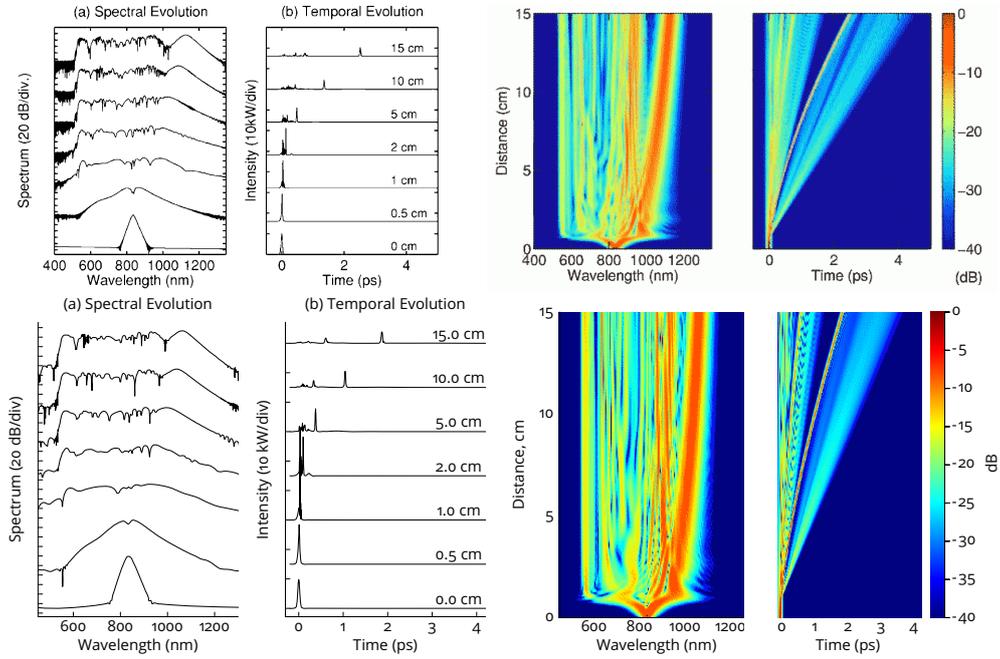


Figure C.2: Spectral and temporal evolution of a  $\text{sech}^2$  pulse at 10 kW peak power and 50 fs FWHM during propagation through a photonic crystal fiber. The top row shows results in [232] and the bottom row shows our reproduction.



## Bibliography

- [1] Lewis Fry Richardson. "Apparatus for warning a ship of its approach to large objects in a fog". In: *British Patent* 9423 (1912).
- [2] Scott Kirkpatrick, C Daniel Gelatt Jr, and Mario P Vecchi. "Optimization by simulated annealing". In: *science* 220.4598 (1983), pp. 671–680.
- [3] *Opus 1.3 Released*. <https://jmvalin.ca/opus/opus-1.3/>. Accessed: 2023-07-11.
- [4] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. url: <https://www.tensorflow.org/>.
- [5] Gordon E Moore. "Cramming more components onto integrated circuits". In: *Proceedings of the IEEE* 86.1 (1998), pp. 82–85.
- [6] Jaime Sevilla et al. "Compute trends across three eras of machine learning". In: *2022 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2022, pp. 1–8.
- [7] *Microprocessor Trend Data*. <https://github.com/karlrupp/microprocessor-trend-data>. Accessed: 2023-07-11.
- [8] Wulfram Gerstner and Werner M Kistler. *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press, 2002.
- [9] Henry Markram. "The human brain project". In: *Scientific American* 306.6 (2012), pp. 50–55.
- [10] Michel Bakni. *Frontier system specs*. Accessed: 2023-08-09. url: <https://www.olcf.ornl.gov/frontier/#4>.
- [11] Kapooht. *Von Neumann Architecture*. Accessed: 2023-04-20. Apr. 2013. url: [https://commons.wikimedia.org/wiki/File:Von\\_Neumann\\_Architecture.svg](https://commons.wikimedia.org/wiki/File:Von_Neumann_Architecture.svg).
- [12] *ITRS 2007 Report*. <http://www.itrs2.net/itrs-reports.html/>. Accessed: 2023-07-11.
- [13] David A. B. Miller and Haldun M. Ozaktas. "Limit to the bit-rate capacity of electrical interconnects from the aspect ratio of the system architecture". In: *Journal of parallel and distributed computing* 41.1 (1997), pp. 42–52.
- [14] David AB Miller. "Device requirements for optical interconnects to silicon chips". In: *Proceedings of the IEEE* 97.7 (2009), pp. 1166–1185.
- [15] Kyung-Hoae Koo et al. "Performance comparisons between carbon nanotubes, optical, and Cu for future high-performance on-chip interconnect applications". In: *IEEE Transactions on Electron Devices* 54.12 (2007), pp. 3206–3215.

- [16] Pedro J Freire et al. "Reducing computational complexity of neural networks in optical channel equalization: From concepts to implementation". In: *Journal of Lightwave Technology* (2023).
- [17] Shaoliang Zhang et al. "Field and lab experimental demonstration of nonlinear impairment compensation using neural networks". In: *Nature communications* 10.1 (2019), p. 3033.
- [18] Noriaki Kaneda et al. "Fixed-point analysis and FPGA implementation of deep neural network based equalizers for high-speed PON". In: *Journal of Lightwave Technology* 40.7 (2021), pp. 1972–1980.
- [19] Vjaceslavs Bobrovs, Sandis Spolitis, and Girts Ivanovs. "Latency causes and reduction in optical metro networks". In: *Optical Metro Networks and Short-Haul Systems VI*. Vol. 9008. SPIE. 2014, pp. 91–101.
- [20] Tong Jian et al. "Radio frequency fingerprinting on the edge". In: *IEEE Transactions on Mobile Computing* 21.11 (2021), pp. 4078–4093.
- [21] Oktay Ureten and Nur Serinken. "Wireless security through RF fingerprinting". In: *Canadian Journal of Electrical and Computer Engineering* 32.1 (2007), pp. 27–33.
- [22] Hsuan-Tung Peng et al. "A photonic-circuits-inspired compact network: Toward real-time wireless signal classification at the edge". In: *arXiv preprint arXiv:2106.13865* (2021).
- [23] Bijoy Chand Chatterjee, Nityananda Sarma, and Eiji Oki. "Routing and spectrum allocation in elastic optical networks: A tutorial". In: *IEEE Communications Surveys & Tutorials* 17.3 (2015), pp. 1776–1800.
- [24] Charis Mesaritakis, Vassilis Papataxiarhis, and Dimitris Syvridis. "Micro ring resonators as building blocks for an all-optical high-speed reservoir-computing bit-pattern-recognition system". In: *JOSA B* 30.11 (2013), pp. 3048–3055.
- [25] Janibul Bashir, Eldhose Peter, and Smruti R Sarangi. "A survey of on-chip optical interconnects". In: *ACM Computing Surveys (CSUR)* 51.6 (2019), pp. 1–34.
- [26] Hsin-Yu Sidney Li, Yong Qiao, and Demetri Psaltis. "Optical network for real-time face recognition". In: *Applied Optics* 32.26 (1993), pp. 5026–5035.
- [27] Léa Constans et al. "III-V/Silicon Hybrid Non-linear Nanophotonics in the Context of On-Chip Optical Signal Processing and Analog Computing". In: *Frontiers in Physics* 7 (2019), p. 133.
- [28] Bhavin J Shastri et al. "Photonics for artificial intelligence and neuromorphic computing". In: *Nature Photonics* 15.2 (2021), pp. 102–114.
- [29] Andrew Katumba et al. "Neuromorphic computing based on silicon photonics and reservoir computing". In: *IEEE Journal of Selected Topics in Quantum Electronics* 24.6 (2018), pp. 1–10.

- [30] Stijn Sackesyn et al. "Experimental realization of integrated photonic reservoir computing for nonlinear fiber distortion compensation". In: *Optics Express* 29.20 (2021), pp. 30991–30997.
- [31] Floris Laporte et al. "Numerical demonstration of neuromorphic computing with photonic crystal cavities". In: *Optics express* 26.7 (2018), pp. 7955–7964.
- [32] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. "Training deep neural networks with low precision multiplications". In: *arXiv preprint arXiv:1412.7024* (2014).
- [33] Stewart E Miller. "Integrated optics: An introduction". In: *The Bell system technical journal* 48.7 (1969), pp. 2059–2069.
- [34] Amir H Atabaki et al. "Integrating photonics with silicon nanoelectronics for the next generation of systems on a chip". In: *Nature* 556.7701 (2018), pp. 349–354.
- [35] Tomáš Kebert and umimeto.org. *Simple neuron scheme*. Accessed: 2023-07-12. Sept. 2020. url: [https://commons.wikimedia.org/wiki/File:Simple\\_neuron\\_scheme.svg](https://commons.wikimedia.org/wiki/File:Simple_neuron_scheme.svg).
- [36] Warren S McCulloch and Walter Pitts. "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5 (1943), pp. 115–133.
- [37] Marvin L Minsky and Seymour A Papert. *Perceptrons: expanded edition*. 1988.
- [38] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6 (1958), p. 386.
- [39] Mysid Dake. *Neural network*. Accessed: 2023-07-12. Nov. 2006. url: [https://commons.wikimedia.org/wiki/File:Neural\\_network.svg](https://commons.wikimedia.org/wiki/File:Neural_network.svg).
- [40] Hava T Siegelmann and Eduardo D Sontag. "On the computational power of neural nets". In: *Proceedings of the fifth annual workshop on Computational learning theory*. 1992, pp. 440–449.
- [41] Herbert Jaeger. "The "echo state" approach to analysing and training recurrent neural networks-with an erratum note". In: *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report* 148.34 (2001), p. 13.
- [42] Herbert Jaeger. "Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach". In: (2002).
- [43] Herbert Jaeger and Harald Haas. "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication". In: *science* 304.5667 (2004), pp. 78–80.
- [44] Iulian Ilies et al. "Stepping forward through echoes of the past: forecasting with echo state networks". In: (2007).

- [45] Pieter Buteneers et al. "Automatic detection of epileptic seizures on the intra-cranial electroencephalogram of rats using reservoir computing". In: *Artificial intelligence in medicine* 53.3 (2011), pp. 215–223.
- [46] Herbert Jaeger. "Short term memory in echo state networks". In: (2001).
- [47] Mantas Lukoševičius. "A practical guide to applying echo state networks". In: *Neural Networks: Tricks of the Trade: Second Edition*. Springer, 2012, pp. 659–686.
- [48] David Verstraeten et al. "An experimental unification of reservoir computing methods". In: *Neural networks* 20.3 (2007), pp. 391–403.
- [49] Wolfgang Maass, Thomas Natschläger, and Henry Markram. "Real-time computing without stable states: A new framework for neural computation based on perturbations". In: *Neural computation* 14.11 (2002), pp. 2531–2560.
- [50] Wolfgang Maass. "Liquid state machines: motivation, theory, and applications". In: *Computability in context: computation and logic in the real world* (2011), pp. 275–296.
- [51] David Verstraeten et al. "Isolated word recognition with the liquid state machine: a case study". In: *Information Processing Letters* 95.6 (2005), pp. 521–528.
- [52] Jürgen Schmidhuber et al. "Training recurrent networks by evolino". In: *Neural computation* 19.3 (2007), pp. 757–779.
- [53] Alex Graves, Abdel-rahman Mohamed, and Hinton Geoffrey. *Neural network*. Accessed: 2023-07-12. June 2017. url: [https://commons.wikimedia.org/wiki/File:Peephole\\_Long\\_Short-Term\\_Memory.svg](https://commons.wikimedia.org/wiki/File:Peephole_Long_Short-Term_Memory.svg).
- [54] Jochen J Steil. "Backpropagation-decorrelation: online recurrent learning with  $O(N)$  complexity". In: *2004 IEEE international joint conference on neural networks (IEEE Cat. No. 04CH37541)*. Vol. 2. IEEE, 2004, pp. 843–848.
- [55] Johannes Nokkala et al. "Gaussian states of continuous-variable quantum systems provide universal and versatile reservoir computing". In: *Communications Physics* 4.1 (2021), p. 53.
- [56] George Bourianoff et al. "Potential implementation of reservoir computing models based on magnetic skyrmions". In: *Aip Advances* 8.5 (2018).
- [57] Chrisantha Fernando and Sampsa Sojakka. "Pattern recognition in a bucket". In: *European conference on artificial life*. Springer, 2003, pp. 588–597.
- [58] BJ Albert. *Noviko. On convergence proofs for perceptrons*. Tech. rep. Technical report, DTIC Document, 1963.

- [59] Elizabeth Goodspeed. *A diagram showing a perceptron updating its linear boundary as more training examples are added*. Accessed: 2023-08-03. May 2015. url: [https://commons.wikimedia.org/wiki/File:Perceptron\\_example.svg](https://commons.wikimedia.org/wiki/File:Perceptron_example.svg).
- [60] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. "Learning representations by back-propagating errors". In: *nature* 323.6088 (1986), pp. 533–536.
- [61] Yann LeCun et al. "A theoretical framework for back-propagation". In: *Proceedings of the 1988 connectionist models summer school*. Vol. 1. San Mateo, CA, USA. 1988, pp. 21–28.
- [62] Julien Launay et al. "Direct feedback alignment scales to modern deep learning tasks and architectures". In: *Advances in neural information processing systems* 33 (2020), pp. 9346–9360.
- [63] Stephen Grossberg. "Competitive learning: From interactive activation to adaptive resonance". In: *Cognitive science* 11.1 (1987), pp. 23–63.
- [64] Tyler W Hughes et al. "Training of photonic neural networks through in situ backpropagation and gradient measurement". In: *Optica* 5.7 (2018), pp. 864–871.
- [65] Sunil Pai et al. "Experimentally realized in situ backpropagation for deep learning in nanophotonic neural networks". In: *arXiv preprint arXiv:2205.08501* (2022).
- [66] Xianxin Guo et al. "Backpropagation through nonlinear units for the all-optical training of neural networks". In: *Photonics Research* 9.3 (2021), B71–B80.
- [67] Timothy P Lillicrap et al. "Random feedback weights support learning in deep neural networks". In: *arXiv preprint arXiv:1411.0247* (2014).
- [68] Arild Nøkland. "Direct feedback alignment provides learning in deep neural networks". In: *Advances in neural information processing systems* 29 (2016).
- [69] Julien Launay, Iacopo Poli, and Florent Krzakala. "Principled training of neural networks with direct feedback alignment". In: *arXiv preprint arXiv:1906.04554* (2019).
- [70] Matthew J Filipovich et al. "Scaling Laws Beyond Backpropagation". In: *arXiv preprint arXiv:2210.14593* (2022).
- [71] Benjamin Scellier and Yoshua Bengio. "Equilibrium propagation: Bridging the gap between energy-based models and backpropagation". In: *Frontiers in computational neuroscience* 11 (2017), p. 24.
- [72] Maria Refinetti et al. "Align, then memorise: the dynamics of learning with feedback alignment". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 8925–8935.

- [73] Manuel P Cuéllar, Miguel Delgado, and MC Pegalajar. "An application of non-linear programming to train recurrent neural networks in time series prediction problems". In: *Enterprise information systems VII*. Springer. 2006, pp. 95–102.
- [74] David Sussillo and Larry F Abbott. "Generating coherent patterns of activity from chaotic neural networks". In: *Neuron* 63.4 (2009), pp. 544–557.
- [75] Chabacano. *Overfitting*. Accessed: 2023-07-20. Feb. 2008. url: <https://commons.wikimedia.org/wiki/File:Overfitting.svg>.
- [76] Kohei Nakajima and Ingo Fischer. *Reservoir Computing*. Springer, 2021.
- [77] Izzet B Yildiz, Herbert Jaeger, and Stefan J Kiebel. "Re-visiting the echo state property". In: *Neural networks* 35 (2012), pp. 1–9.
- [78] Karl Pearson. "LIII. On lines and planes of closest fit to systems of points in space". In: *The London, Edinburgh, and Dublin philosophical magazine and journal of science* 2.11 (1901), pp. 559–572.
- [79] Joni Dambre et al. "Information processing capacity of dynamical systems". In: *Scientific reports* 2.1 (2012), p. 514.
- [80] Masanobu Inubushi and Kazuyuki Yoshimura. "Reservoir computing beyond memory-nonlinearity trade-off". In: *Scientific reports* 7.1 (2017), p. 10199.
- [81] Anas Skalli et al. "Computational metrics and parameters of an injection-locked large area semiconductor laser for neural network computing". In: *Optical Materials Express* 12.7 (2022), pp. 2793–2804.
- [82] Shiyu Ji. *An illustration of kernel trick in SVM*. Accessed: 2023-07-11. June 2017. url: [https://commons.wikimedia.org/wiki/File:Kernel\\_trick\\_idea.svg](https://commons.wikimedia.org/wiki/File:Kernel_trick_idea.svg).
- [83] Robert Legenstein and Wolfgang Maass. "Edge of chaos and prediction of computational performance for neural circuit models". In: *Neural networks* 20.3 (2007), pp. 323–334.
- [84] Lars Büsing, Benjamin Schrauwen, and Robert Legenstein. "Connectivity, dynamics, and memory in reservoir computing with binary and analog neurons". In: *Neural computation* 22.5 (2010), pp. 1272–1311.
- [85] Steven H Strogatz. *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. 2nd ed. CRC Press, 2015.
- [86] Terufumi Yamaguchi et al. "Computational capability for physical reservoir computing using a spin-torque oscillator with two free layers". In: *Physical Review B* 107.5 (2023), p. 054406.
- [87] Benjamin Cramer et al. "Control of criticality and computation in spiking neuromorphic networks with plasticity". In: *Nature communications* 11.1 (2020), p. 2853.

- [88] Atsushi Uchida, Ryan McAllister, and Rajarshi Roy. "Consistency of non-linear system response to complex drive signals". In: *Physical review letters* 93.24 (2004), p. 244102.
- [89] Thomas Lymburn et al. "Consistency in echo-state networks". In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 29.2 (2019), p. 023118.
- [90] Hailong Zhou et al. "Photonic matrix multiplication lights up photonic accelerator and beyond". In: *Light: Science & Applications* 11.1 (2022), p. 30.
- [91] Yichen Shen et al. "Deep learning with coherent nanophotonic circuits". In: *Nature photonics* 11.7 (2017), pp. 441–446.
- [92] Matthew J Filipovich et al. "Silicon photonic architecture for training deep neural networks with direct feedback alignment". In: *Optica* 9.12 (2022), pp. 1323–1332.
- [93] Johannes Feldmann et al. "Parallel convolutional processing using an integrated photonic tensor core". In: *Nature* 589.7840 (2021), pp. 52–58.
- [94] C David Wright, Harish Bhaskaran, and Wolfram HP Pernice. "Integrated phase-change photonic devices and systems". In: *MRS Bulletin* 44.9 (2019), pp. 721–727.
- [95] Stefan Abel et al. "Silicon photonics integration technologies for future computing systems". In: *2019 24th OptoElectronics and Communications Conference (OECC) and 2019 International Conference on Photonics in Switching and Computing (PSC)*. IEEE, 2019, pp. 1–3.
- [96] Lorenzo De Marinis et al. "Photonic neural networks: A survey". In: *IEEE Access* 7 (2019), pp. 175827–175841.
- [97] Farshid Ashtiani, Alexander J Geers, and Firooz Aflatouni. "An on-chip photonic deep neural network for image classification". In: *Nature* 606.7914 (2022), pp. 501–506.
- [98] Saumil Bandyopadhyay et al. "Single chip photonic deep neural network with accelerated training". In: *arXiv preprint arXiv:2208.01623* (2022).
- [99] Alexander N Tait et al. "Neuromorphic photonic networks using silicon photonic weight banks". In: *Scientific reports* 7.1 (2017), p. 7430.
- [100] Lennert Appeltant et al. "Information processing using a single dynamical node as complex system". In: *Nature communications* 2.1 (2011), p. 468.
- [101] Laurent Larger et al. "High-speed photonic reservoir computing using a time-delay-based architecture: Million words per second classification". In: *Physical Review X* 7.1 (2017), p. 011015.
- [102] Yvan Paquot et al. "Optoelectronic reservoir computing". In: *Scientific reports* 2.1 (2012), p. 287.

- [103] Daniel Brunner et al. "Tutorial: Photonic neural networks in delay systems". In: *Journal of Applied Physics* 124.15 (2018).
- [104] Laurent Larger et al. "Photonic information processing beyond Turing: an optoelectronic implementation of reservoir computing". In: *Optics express* 20.3 (2012), pp. 3241–3249.
- [105] Daniel Brunner et al. "Parallel photonic information processing at gigabyte per second data rates using transient states". In: *Nature communications* 4.1 (2013), p. 1364.
- [106] Kosuke Takano et al. "Compact reservoir computing with a photonic integrated circuit". In: *Optics express* 26.22 (2018), pp. 29424–29439.
- [107] Chihiro Sugano, Kazutaka Kanno, and Atsushi Uchida. "Reservoir computing using multiple lasers with feedback on a photonic integrated circuit". In: *IEEE Journal of Selected Topics in Quantum Electronics* 26.1 (2019), pp. 1–9.
- [108] Daniel J Blumenthal et al. "Silicon nitride in silicon photonics". In: *Proceedings of the IEEE* 106.12 (2018), pp. 2209–2231.
- [109] Wosen Kassa et al. "Towards integrated parallel photonic reservoir computing based on frequency multiplexing". In: *Neuro-inspired Photonic Computing*. Vol. 10689. SPIE. 2018, p. 1068903.
- [110] Lorenz Butschek et al. "Photonic reservoir computer based on frequency multiplexing". In: *Optics Letters* 47.4 (2022), pp. 782–785.
- [111] Alessandro Lupo et al. "Fully analog photonic deep Reservoir Computer based on frequency multiplexing". In: *arXiv preprint arXiv:2305.08892* (2023).
- [112] Kristof Vandoorne et al. "Toward optical signal processing using photonic reservoir computing". In: *Optics express* 16.15 (2008), pp. 11182–11192.
- [113] Herbert Jaeger. "Adaptive nonlinear system identification with echo state networks". In: *Advances in neural information processing systems* 15 (2002).
- [114] Kristof Vandoorne et al. "Parallel reservoir computing using optical amplifiers". In: *IEEE transactions on neural networks* 22.9 (2011), pp. 1469–1481.
- [115] Richard Lyon. "A computational model of filtering, detection, and compression in the cochlea". In: *ICASSP'82. IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 7. IEEE. 1982, pp. 1282–1285.
- [116] Kristof Vandoorne et al. "Experimental demonstration of reservoir computing on a silicon photonics chip". In: *Nature communications* 5.1 (2014), p. 3541.

- [117] Martin Andre Agnes Fiers et al. "Nanophotonic reservoir computing with photonic crystal cavities to generate periodic patterns". In: *IEEE transactions on neural networks and learning systems* 25.2 (2013), pp. 344–355.
- [118] Florian Denis-Le Coarer et al. "All-optical reservoir computing on a photonic chip using silicon-based ring resonators". In: *IEEE Journal of Selected Topics in Quantum Electronics* 24.6 (2018), pp. 1–8.
- [119] Anas Skalli et al. "Photonic neuromorphic computing using vertical cavity semiconductor lasers". In: *Optical Materials Express* 12.6 (2022), pp. 2395–2414.
- [120] Julian Bueno et al. "Reinforcement learning in a large-scale photonic recurrent neural network". In: *Optica* 5.6 (2018), pp. 756–760.
- [121] Piotr Antonik et al. "Human action recognition with a large-scale brain-inspired photonic computer". In: *Nature Machine Intelligence* 1.11 (2019), pp. 530–537.
- [122] Xing Lin et al. "All-optical machine learning using diffractive deep neural networks". In: *Science* 361.6406 (2018), pp. 1004–1008.
- [123] Jeff Chiles et al. "Design, fabrication, and metrology of  $10 \times 100$  multiplanar integrated photonic routing manifolds for neural networks". In: *APL Photonics* 3.10 (2018).
- [124] Elger A Vlieg et al. "An Integrated Photorefractive Analog Matrix-Vector Multiplier for Machine Learning". In: *Applied Sciences* 12.9 (2022), p. 4226.
- [125] Bin Shi, Nicola Calabretta, and Ripalta Stabile. "Deep neural network through an InP SOA-based photonic integrated cross-connect". In: *IEEE Journal of Selected Topics in Quantum Electronics* 26.1 (2019), pp. 1–11.
- [126] Erfan Khoram et al. "Nanophotonic media for artificial neural inference". In: *Photonics Research* 7.8 (2019), pp. 823–827.
- [127] Chonghuai Ma et al. "Integrated Photonic Reservoir Computing with All-Optical Readout". In: *arXiv preprint arXiv:2306.15845* (2023).
- [128] Guy Van der Sande et al. "Time-multiplexed optical systems for reservoir computing and coherent Ising machines". In: *Emerging Topics in Artificial Intelligence (ETAI) 2021*. Vol. 11804. SPIE. 2021, pp. 46–54.
- [129] Johnny Moughames et al. "Three-dimensional waveguide interconnects for scalable integration of photonic neural networks". In: *Optica* 7.6 (2020), pp. 640–646.
- [130] Johnny Moughames et al. "3D printed multimode-splitters for photonic interconnects". In: *Optical Materials Express* 10.11 (2020), pp. 2952–2961.

- [131] Matthias Freiberger et al. "Training passive photonic reservoirs with integrated optical readout". In: *IEEE transactions on neural networks and learning systems* 30.7 (2018), pp. 1943–1953.
- [132] Niels Quack et al. "Integrated silicon photonic MEMS". In: *Microsystems & Nanoengineering* 9.1 (2023), p. 27.
- [133] Tiantian Li et al. "Neuromorphic Photonics Based on Phase Change Materials". In: *Nanomaterials* 13.11 (2023), p. 1756.
- [134] Xuan Li et al. "Fast and reliable storage using a 5 bit, nonvolatile photonic memory cell". In: *Optica* 6.1 (2019), pp. 1–6.
- [135] Chunmin Liu et al. "A study on optical properties of Sb<sub>2</sub>Se<sub>3</sub> thin films and resistive switching behavior in Ag/Sb<sub>2</sub>Se<sub>3</sub>/W heterojunctions". In: *Results in Physics* 13 (2019), p. 102228.
- [136] Chonghuai Ma et al. "Addressing limited weight resolution in a fully optical neuromorphic reservoir computing readout". In: *Scientific Reports* 11.1 (2021), p. 3102.
- [137] Hermann Haus. "Waves and fields in optoelectronics." In: *PRENTICE-HALL, INC., ENGLEWOOD CLIFFS, NJ 07632, USA, 1984, 402* (1984).
- [138] Gregory Moille, Sylvain Combrié, and Alfredo De Rossi. "Modeling of the carrier dynamics in nonlinear semiconductor nanoscale resonators". In: *Physical Review A* 94.2 (2016), p. 023814.
- [139] Toshitsugu Uesugi et al. "Investigation of optical nonlinearities in an ultra-high-Q Si nanocavity in a two-dimensional photonic crystal slab". In: *Optics express* 14.1 (2006), pp. 377–386.
- [140] Takasumi Tanabe, Hideaki Taniyama, and Masaya Notomi. "Carrier diffusion and recombination in photonic crystal nanocavity optical switches". In: *Journal of Lightwave Technology* 26.11 (2008), pp. 1396–1403.
- [141] Brian R Bennett, Richard A Soref, and Jesus A Del Alamo. "Carrier-induced change in refractive index of InP, GaAs and InGaAsP". In: *IEEE Journal of Quantum Electronics* 26.1 (1990), pp. 113–122.
- [142] Qianfan Xu and Michal Lipson. "Carrier-induced optical bistability in silicon ring resonators". In: *Optics letters* 31.3 (2006), pp. 341–343.
- [143] Alfredo de Rossi et al. "Interplay of plasma-induced and fast thermal nonlinearities in a GaAs-based photonic crystal nanocavity". In: *Physical Review A* 79.4 (2009), p. 043818.
- [144] C Manolatou et al. "Coupling of modes analysis of resonant channel add-drop filters". In: *IEEE journal of quantum electronics* 35.9 (1999), pp. 1322–1331.
- [145] Christopher Rackauckas and Qing Nie. "Differenialequations.jl—a performant and feature-rich ecosystem for solving differential equations in julia". In: *Journal of Open Research Software* 5.1 (2017).

- [146] Jeff Bezanson et al. "Julia: A fresh approach to numerical computing". In: *SIAM review* 59.1 (2017), pp. 65–98.
- [147] Tarek A Ibrahim et al. "All-optical AND/NAND logic gates using semiconductor microresonators". In: *IEEE Photonics Technology Letters* 15.10 (2003), pp. 1422–1424.
- [148] Devadatta Kulkarni, Darrell Schmidt, and Sze-Kai Tsui. "Eigenvalues of tridiagonal pseudo-Toeplitz matrices". In: *Linear Algebra and its Applications* 297 (1999), pp. 63–80.
- [149] Kun Il Park and M Park. *Fundamentals of probability and stochastic processes with applications to communications*. Springer, 2018.
- [150] Takasumi Tanabe et al. "All-optical switches on a silicon chip realized using photonic crystal nanocavities". In: *Applied Physics Letters* 87.15 (2005), p. 151112.
- [151] TA Ibrahim, V Van, and P-T Ho. "All-optical time-division demultiplexing and spatial pulse routing with a GaAs/AlGaAs microring resonator". In: *Optics Letters* 27.10 (2002), pp. 803–805.
- [152] Sylvain Combrié et al. "GaAs photonic crystal cavity with ultrahigh Q: microwatt nonlinearity at 1.55  $\mu\text{m}$ ". In: *Optics letters* 33.16 (2008), pp. 1908–1910.
- [153] Chad Husko et al. "Ultrafast all-optical modulation in GaAs photonic crystal cavities". In: *Applied Physics Letters* 94.2 (2009), p. 021111.
- [154] Gregory Moille et al. "100ghz integrated all-optical switch enabled by ald". In: *arXiv preprint arXiv:1506.01609* (2015).
- [155] Qianfan Xu et al. "Micrometre-scale silicon electro-optic modulator". In: *nature* 435.7040 (2005), pp. 325–327.
- [156] Michael Waldow et al. "25ps all-optical switching in oxygen implanted silicon-on-insulator microring resonator". In: *Optics Express* 16.11 (2008), pp. 7693–7702.
- [157] Takasumi Tanabe et al. "Fast all-optical switching using ion-implanted silicon photonic crystal nanocavities". In: *Applied Physics Letters* 90.3 (2007), p. 031115.
- [158] Mikkel Heuck et al. "Heterodyne pump probe measurements of nonlinear dynamics in an indium phosphide photonic crystal cavity". In: *Applied Physics Letters* 103.18 (2013), p. 181120.
- [159] Matthew Dale et al. "Reservoir computing quality: connectivity and topology". In: *Natural Computing* 20 (2021), pp. 205–216.
- [160] Mustafa C Ozturk, Dongming Xu, and Jose C Principe. "Analysis and design of echo state networks". In: *Neural computation* 19.1 (2007), pp. 111–138.

- [161] Shijun Xiao et al. "Compact silicon microring resonators with ultra-low propagation loss in the C band". In: *Optics express* 15.22 (2007), pp. 14467–14475.
- [162] Lin Chang et al. "Ultra-efficient frequency comb generation in AlGaAs-on-insulator microresonators". In: *Nature communications* 11.1 (2020), pp. 1–8.
- [163] Takashi Asano et al. "Photonic crystal nanocavity with a Q factor exceeding eleven million". In: *Optics express* 25.3 (2017), pp. 1769–1777.
- [164] Xingchen Ji et al. "Ultra-low-loss on-chip resonators with sub-milliwatt parametric oscillation threshold". In: *Optica* 4.6 (2017), pp. 619–624.
- [165] Stefan F Preble et al. "Ultrafast all-optical modulation on a silicon chip". In: *Optics letters* 30.21 (2005), pp. 2891–2893.
- [166] SW Leonard et al. "Ultrafast band-edge tuning of a two-dimensional silicon photonic crystal via free-carrier injection". In: *Physical Review B* 66.16 (2002), p. 161102.
- [167] Wesley D Sacher et al. "Monolithically integrated multilayer silicon nitride-on-silicon waveguide platforms for 3-D photonic circuits and devices". In: *Proceedings of the IEEE* 106.12 (2018), pp. 2232–2245.
- [168] Argishti Melikyan et al. "High-speed plasmonic phase modulators". In: *Nature Photonics* 8.3 (2014), pp. 229–233.
- [169] Takumi Moriyama et al. "Ultra-compact, self-holding asymmetric Mach-Zehnder interferometer switch using Ge<sub>2</sub>Sb<sub>2</sub>Te<sub>5</sub> phase-change material". In: *IEICE Electronics Express* 11.15 (2014), pp. 20140538–20140538.
- [170] Delphin Dodane et al. "Fully embedded photonic crystal cavity with Q=0.6 million fabricated within a full-process CMOS multiproject wafer". In: *Optics express* 26.16 (2018), pp. 20868–20877.
- [171] Masaya Notomi, E Kuramochi, and H Taniyama. "Ultrahigh-Q nanocavity with 1D photonic gap". In: *Optics Express* 16.15 (2008), pp. 11095–11102.
- [172] Xing Chen et al. "Forecasting the outcome of spintronic experiments with neural ordinary differential equations". In: *Nature communications* 13.1 (2022), p. 1016.
- [173] Imran Maqsood, Muhammad Riaz Khan, and Ajith Abraham. "An ensemble of neural networks for weather forecasting". In: *Neural Computing & Applications* 13 (2004), pp. 112–122.
- [174] Michael C Mackey and Leon Glass. "Oscillation and chaos in physiological control systems". In: *Science* 197.4300 (1977), pp. 287–289.
- [175] Hendrik Wernecke, Bulcsú Sándor, and Claudius Gros. "Chaos in time delay systems, an educational review". In: *Physics Reports* 824 (2019), pp. 1–40.

- [176] Julián Bueno et al. "Conditions for reservoir computing performance using semiconductor lasers with delayed optical feedback". In: *Optics express* 25.3 (2017), pp. 2401–2412.
- [177] Giovanni Donati et al. "Microring resonators with external optical feedback for time delay reservoir computing". In: *Optics Express* 30.1 (2022), pp. 522–537.
- [178] Michiel Hermans and Benjamin Schrauwen. "Memory in linear recurrent neural networks in continuous time". In: *Neural Networks* 23.3 (2010), pp. 341–355.
- [179] Andrew Katumba et al. "A multiple-input strategy to efficient integrated photonic reservoir computing". In: *Cognitive Computation* 9 (2017), pp. 307–314.
- [180] Erik Agrell et al. "Roadmap of optical communications". In: *Journal of Optics* 18.6 (2016), p. 063002.
- [181] Govind P Agrawal. *Fiber-optic communication systems*. John Wiley & Sons, 2012.
- [182] DAVID Mazzaresse. "Minimizing latency in long-haul networks". In: *Light-wave Magazine* 28.5 (2011), pp. 11–16.
- [183] Michel Bakni. *Digital signal encoding formats*. Accessed: 2023-07-28. Oct. 2021. url: [https://commons.wikimedia.org/wiki/File:Digital\\_signal\\_encoding\\_formats-en.svg](https://commons.wikimedia.org/wiki/File:Digital_signal_encoding_formats-en.svg).
- [184] USB-IF. *USB4® Specification v2.0*. Accessed: 2023-07-18. June 2023. url: <https://www.usb.org/document-library/usb4r-specification-v20>.
- [185] Michel Bakni. *Rectangular constellation for QAM*. Accessed: 2023-07-18. Sept. 2021. url: [https://commons.wikimedia.org/wiki/File:Rectangular\\_constellation\\_for\\_QAM.svg](https://commons.wikimedia.org/wiki/File:Rectangular_constellation_for_QAM.svg).
- [186] Yapeng Xie et al. "Machine learning applications for short reach optical communication". In: *Photonics*. Vol. 9. 1. MDPI. 2022, p. 30.
- [187] GP Agrawal. *Nonlinear fiber optics*. Academic Press, 2007.
- [188] Corning. *Corning® SMF-28® Ultra Optical Fiber Product Information*. Accessed: 2023-07-19. Nov. 2021. url: <https://www.corning.com/optical-communications/worldwide/en/home/products/fiber/optical-fiber-products/smf-28-ultra.html>.
- [189] Apostolos Argyris, Julián Bueno, and Ingo Fischer. "Photonic machine learning implementation for signal recovery in optical communications". In: *Scientific reports* 8.1 (2018), p. 8487.
- [190] David Blackman and Sebastiano Vigna. "Scrambled linear pseudorandom number generators". In: *ACM Transactions on Mathematical Software (TOMS)* 47.4 (2021), pp. 1–32.

- [191] Edmund T Rolls and Gustavo Deco. "The noisy brain: stochastic dynamics as a principle of brain function". In: (2010).
- [192] Alan L Hodgkin, Andrew F Huxley, and Bernard Katz. "Measurement of current-voltage relations in the membrane of the giant axon of Loligo". In: *The Journal of physiology* 116.4 (1952), p. 424.
- [193] Alan L Hodgkin. "The local electric changes associated with repetitive action in a non-medullated axon". In: *The Journal of physiology* 107.2 (1948), p. 165.
- [194] Bhavin J Shastri et al. "Principles of neuromorphic photonics". In: *arXiv preprint arXiv:1801.00016* (2017).
- [195] David I Perrett, Edmond T Rolls, and Woody Caan. "Visual neurones responsive to faces in the monkey temporal cortex". In: *Experimental brain research* 47 (1982), pp. 329–342.
- [196] Federico Paredes-Vallés, Kirk YW Scheper, and Guido CHE De Croon. "Unsupervised learning of a hierarchical spiking neural network for optical flow estimation: From events to global motion perception". In: *IEEE transactions on pattern analysis and machine intelligence* 42.8 (2019), pp. 2051–2064.
- [197] Wolfgang Maass. "Networks of spiking neurons: the third generation of neural network models". In: *Neural networks* 10.9 (1997), pp. 1659–1671.
- [198] Bruno Romeira et al. "Brain-inspired nanophotonic spike computing: challenges and prospects". In: *Neuromorphic Computing and Engineering* (2023).
- [199] Wulfram Gerstner et al. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press, 2014.
- [200] LM Lapicque. "Recherches quantitatives sur l'excitation électrique des nerfs". In: *J Physiol Paris* 9 (1907), pp. 620–635.
- [201] Nicolas Brunel and Mark CW Van Rossum. "Lapicque's 1907 paper: from frogs to integrate-and-fire". In: *Biological cybernetics* 97.5-6 (2007), pp. 337–339.
- [202] Paul A Merolla et al. "A million spiking-neuron integrated circuit with a scalable communication network and interface". In: *Science* 345.6197 (2014), pp. 668–673.
- [203] Mike Davies et al. "Loihi: A neuromorphic manycore processor with on-chip learning". In: *Ieee Micro* 38.1 (2018), pp. 82–99.
- [204] Garrick Orchard et al. "Efficient neuromorphic signal processing with loihi 2". In: *2021 IEEE Workshop on Signal Processing Systems (SiPS)*. IEEE, 2021, pp. 254–259.
- [205] Jiadi Zhu et al. "A comprehensive review on emerging artificial neuromorphic devices". In: *Applied Physics Reviews* 7.1 (2020).

- [206] Thomas Van Vaerenbergh et al. "Cascadable excitability in microrings". In: *Optics express* 20.18 (2012), pp. 20292–20308.
- [207] Maia Brunstein et al. "Excitability and self-pulsing in a photonic crystal nanocavity". In: *Physical Review A* 85.3 (2012), p. 031803.
- [208] M Turconi et al. "Control of excitable pulses in an injection-locked semiconductor laser". In: *Physical Review E* 88.2 (2013), p. 022923.
- [209] F Selmi et al. "Relative refractory period in an excitable semiconductor laser". In: *Physical review letters* 112.18 (2014), p. 183902.
- [210] Shuiying Xiang et al. "Hardware-algorithm collaborative computing with photonic spiking neuron chip based on an integrated Fabry–Perot laser with a saturable absorber". In: *Optica* 10.2 (2023), pp. 162–171.
- [211] Maxime Delmulle et al. "Self-Pulsing Nanobeam Photonic Crystal Laser". In: *CLEO: QELS\_Fundamental Science*. Optica Publishing Group. 2022, JW3B–23.
- [212] Mitchell A Nahmias et al. "A laser spiking neuron in a photonic integrated circuit". In: *arXiv preprint arXiv:2012.08516* (2020).
- [213] Guillaume Crosnier et al. "Hybrid indium phosphide-on-silicon nanolaser diode". In: *Nature Photonics* 11.5 (2017), pp. 297–300.
- [214] Maxime Delmulle. "Le nanolaser neurone: excitabilité dans les lasers à cristaux photoniques". PhD thesis. 2023.
- [215] Mihai Alexandru Petrovici. *Form versus function: theory and models for neuronal substrates*. Vol. 1. Springer, 2016.
- [216] John J Hopfield. "Neural networks and physical systems with emergent collective computational abilities." In: *Proceedings of the national academy of sciences* 79.8 (1982), pp. 2554–2558.
- [217] Shlomo Zilberstein. "Using anytime algorithms in intelligent systems". In: *AI magazine* 17.3 (1996), pp. 73–73.
- [218] Lars Buesing et al. "Neural dynamics as sampling: a model for stochastic computation in recurrent networks of spiking neurons". In: *PLoS computational biology* 7.11 (2011), e1002211.
- [219] Mihai A Petrovici et al. "Stochastic inference with spiking neurons in the high-conductance state". In: *Physical Review E* 94.4 (2016), p. 042312.
- [220] Solomon Kullback and Richard A Leibler. "On information and sufficiency". In: *The annals of mathematical statistics* 22.1 (1951), pp. 79–86.
- [221] Geoffrey E Hinton et al. "The "wake-sleep" algorithm for unsupervised neural networks". In: *Science* 268.5214 (1995), pp. 1158–1161.
- [222] Minoru Yamada. "A theoretical analysis of self-sustained pulsation phenomena in narrow-stripe semiconductor lasers". In: *IEEE Journal of Quantum Electronics* 29.5 (1993), pp. 1330–1336.

- [223] Gunnar Björk, Anders Karlsson, and Yoshihisa Yamamoto. "On the linewidth of microcavity lasers". In: *Applied physics letters* 60.3 (1992), pp. 304–306.
- [224] Gian Luca Lippi, Jesper Mørk, and Gian Piero Puccioni. "Numerical solutions to the Laser Rate Equations with noise: technical issues, implementation and pitfalls". In: *Nanophotonics VII*. Vol. 10672. SPIE. 2018, pp. 82–95.
- [225] Emil C André, Jesper Mørk, and Martijn Wubs. "Efficient stochastic simulation of rate equations and photon statistics of nanolasers". In: *Optics Express* 28.22 (2020), pp. 32632–32646.
- [226] Larry A Coldren, Scott W Corzine, and Milan L Mashanovitch. *Diode lasers and photonic integrated circuits*. John Wiley & Sons, 2012.
- [227] J Mork and GL Lippi. "Rate equation description of quantum noise in nanolasers with few emitters". In: *Applied Physics Letters* 112.14 (2018).
- [228] A Lebreton et al. "Stochastically sustained population oscillations in high- $\beta$  nanolasers". In: *New Journal of Physics* 15.3 (2013), p. 033039.
- [229] David Elvira et al. "Higher-order photon correlations in pulsed photonic crystal nanolasers". In: *Physical Review A* 84.6 (2011), p. 061802.
- [230] Huaqiang Wu et al. "Multiplication on the edge". In: *Nature Electronics* 1.1 (2018), pp. 8–9.
- [231] Claudio Gallicchio, Alessio Micheli, and Luca Pedrelli. "Deep reservoir computing: A critical experimental analysis". In: *Neurocomputing* 268 (2017), pp. 87–99.
- [232] John M Dudley, Goëry Genty, and Stéphane Coen. "Supercontinuum generation in photonic crystal fiber". In: *Reviews of modern physics* 78.4 (2006), p. 1135.