



## A photonics perspective on computing with physical substrates

S. Abreu<sup>a</sup>, I. Boikov<sup>b</sup>, M. Goldmann<sup>c</sup>, T. Jonuzi<sup>d</sup>, A. Lupo<sup>e</sup>, S. Masaad<sup>f</sup>, L. Nguyen<sup>g</sup>,  
 E. Picco<sup>e</sup>, G. Pourcel<sup>a</sup>, A. Skalli<sup>h</sup>, L. Talandier<sup>c</sup>, B. Vettelschoss<sup>i</sup>, E.A. Vlieg<sup>j</sup>,  
 A. Argyris<sup>c</sup>, P. Bienstman<sup>f</sup>, D. Brunner<sup>h</sup>, J. Dambre<sup>i</sup>, L. Daudet<sup>k</sup>, J.D. Domenech<sup>d</sup>,  
 I. Fischer<sup>c</sup>, F. Horst<sup>j</sup>, S. Massar<sup>e</sup>, C.R. Mirasso<sup>c</sup>, B.J. Offrein<sup>j</sup>, A. Rossi<sup>b</sup>,  
 M.C. Soriano<sup>c,\*</sup>, S. Sygletos<sup>g</sup>, S.K. Turitsyn<sup>g</sup>

<sup>a</sup> University of Groningen, Netherlands

<sup>b</sup> Thales Research & Technology, Palaiseau Cedex 91767, France

<sup>c</sup> Instituto de Física Interdisciplinar y Sistemas Complejos (IFISC, UIB-CSIC), Campus Universitat de les Illes Balears E-07122, Palma de Mallorca, Spain

<sup>d</sup> VLC Photonics S.L., c/ Camino de Vera s/n, 46022 Valencia, Spain

<sup>e</sup> Laboratoire d'Information Quantique, Université libre de Bruxelles (ULB), Av. Roosevelt 50, 1050 Brussels, Belgium

<sup>f</sup> Photonics Research Group, Department of Information Technology, Ghent University - IMEC, Ghent, Belgium

<sup>g</sup> Aston Institute of Photonic Technologies, Aston University, Birmingham B4 7ET, UK

<sup>h</sup> FEMTO-ST Institute/Optics Department, CNRS & University Franche-Comté, Besançon Cedex 25030, France

<sup>i</sup> IDLab, Department of Electronics and Information Systems, Ghent University - IMEC, Ghent, Belgium

<sup>j</sup> IBM Research Europe—Zurich Research Laboratory, Rüschlikon, Switzerland

<sup>k</sup> LightOn, 3-5 Impasse Reille, 75014 Paris, France

### ARTICLE INFO

#### Keywords:

Unconventional computing  
 Physical substrates  
 Machine learning  
 Photonic neural networks

### ABSTRACT

We provide a perspective on the fundamental relationship between physics and computation, exploring the conditions under which a physical system can be harnessed for computation and the practical means to achieve this. Unlike traditional digital computers that impose discreteness on continuous substrates, unconventional computing embraces the inherent properties of physical systems. Exploring simultaneously the intricacies of physical implementations and applied computational paradigms, we discuss the interdisciplinary developments of unconventional computing. Here, we focus on the potential of photonic substrates for unconventional computing, implementing artificial neural networks to solve data-driven machine learning tasks. Several photonic neural network implementations are discussed, highlighting their potential advantages over electronic counterparts in terms of speed and energy efficiency. Finally, we address the challenges of achieving learning and programmability within physical substrates, outlining key strategies for future research.

### Contents

1. Introduction .....	2
2. Unconventional computing with physical substrates.....	3
2.1. Introduction.....	3
2.2. Adapting physical systems for computing tasks .....	4
2.3. Random mappings for unconventional computing with physical substrates.....	5

\* Corresponding author.

E-mail address: [miguel@ifisc.uib-csic.es](mailto:miguel@ifisc.uib-csic.es) (M.C. Soriano).

<https://doi.org/10.1016/j.revip.2024.100093>

Received 28 February 2024; Received in revised form 30 May 2024; Accepted 4 June 2024

Available online 14 June 2024

2405-4283/© 2024 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

3.	Applications .....	5
3.1.	Image and video processing .....	6
3.2.	Time-series prediction .....	6
3.3.	Telecommunication tasks .....	6
3.4.	Hardware accelerators .....	8
4.	Physics-driven implementations of unconventional computing substrates .....	8
4.1.	Introduction .....	8
4.2.	Space multiplexing .....	8
4.3.	Time multiplexing .....	9
4.3.1.	Optoelectronic feedback system .....	10
4.3.2.	All-optical feedback system .....	10
4.4.	Frequency multiplexing .....	12
4.5.	Towards integration .....	13
4.5.1.	Passive silicon photonics chip .....	14
4.5.2.	Coupled optical resonators .....	15
5.	Theory-driven implementations of unconventional computing substrates: Towards industrial applications .....	16
5.1.	Introduction .....	16
5.2.	Matrix–vector multiplications .....	17
5.3.	Temporal convolutional processing .....	18
6.	Benchmarking .....	18
7.	Programmability and learning in physical substrates for unconventional computing .....	19
7.1.	Introduction .....	19
7.2.	Model-based physical programming .....	20
7.3.	Partially model-free physical programming .....	20
7.4.	Model-free physical programming .....	20
7.5.	Computational abstractions beyond end-to-end learning .....	21
8.	Conclusion .....	21
	Declaration of competing interest .....	22
	Acknowledgments .....	22
	References .....	22

## 1. Introduction

When does a physical system compute? What does it mean to compute *with* physics? How can we *make* a physical system compute? Only the first of these three fundamental questions has been answered to a certain degree, first at a high level of abstraction [1], and recently worked out in greater detail [2–4]. As an example of a physical substrate performing unconventional computing, the slime mold *Physarum polycephalum* is surprisingly good at finding the shortest paths through a network [5]. When placed on a surface with food sources distributed in specific patterns, the slime mold extends tendrils to explore and then retracts, leaving a network of connections marking the most efficient routes [5,6]. This behavior can then be used as a model for naturally computing the solutions of network optimization problems in fields like traffic management and urban planning.

We may broadly contrast *computing with physics* with digital computing, which can be seen as computing *against* physics, by imposing a digital nature – necessitated by a symbolic theory of computation – onto an inherently continuous physical substrate. Indeed, digital computing started with a symbolic theory of computation, with the Turing machine model as its still valid central theoretical basis [7]. Digital computing systems are then implemented, over many decades, into increasingly well-engineered computer chips that are used in a bi-stable switching regime to fit the constraints of our theory of computing. Thus, the physics of our computers is made to fit our theory of computation. In contrast, unconventional computing with physical substrates can be seen as starting from the physics, and developing a theory of computation that effectively uses the physical substrate at hand. On first glance, this is what quantum computing seems to have done. However, quantum computing in its traditional formulation *expanded* our symbolic theory of computation — the foundation of quantum computing still rests firmly on digital computing [8]. Moreover, it was only after we had a sufficiently good theory of quantum physics that it was possible to merge this into our theory of computation. This suggests a lack of examples for theories of computation developed by looking at the dynamical diversity of physical systems. Remarkable efforts in this direction were triggered by the physics of computation courses in Caltech, which were a series of interdisciplinary courses that explored the fundamental limits and possibilities of computation from a physical perspective [9]. They were initiated in 1981 by Richard Feynman, Carver Mead, and John Hopfield, three eminent professors at Caltech with backgrounds in physics, engineering, and biology, respectively. These courses covered topics such as quantum computation, quantum information, neural networks, neural computation, and neuromorphic engineering, sparking new ideas and directions for research in the physics of computation, as well as its applications and implications.

We are currently witnessing a renewed interest in exploring the capability of various physical substrates to compute. The quest is rooted in fundamental scientific curiosity and economic and ecological necessities. Therefore, efforts to understand, design, and implement computing systems that are superior in terms of energy consumption, hardware requirements, hardware cost, speed, latency, bandwidth exploitation, robustness, precision, inference abilities, and more have grown dramatically in recent years. Essential for achieving breakthroughs in this field is gaining a better understanding of how the intricate connections

between hardware and software, or substrate, its dynamical properties, configuration and operating conditions, contribute to solving demanding computing tasks in the most effective way [10,11]. The current scientific efforts build upon a long history of analog computing with physical substrates [12–14]. How physical systems inherently perform computation in various analog computing paradigms has been studied, including in-materio, analog optical, and neuroscience-inspired computing [15–17], among others. The ultimate underlying aim is getting closer to answering the longstanding question of how intelligence can emerge in or on a physical substrate [18]. Photonic systems have proven particularly attractive and versatile for effective analog physical computing. In the following, we provide our perspective on this approach.

In this context of unconventional computing from a physical perspective, we highlight two noteworthy directions for developing novel computing paradigms. Firstly, neuromorphic computing aims to abstract the computations implemented in one physical system, the brain, into a model of computing that can be effectively implemented in a different physical system, e.g., mixed-signal CMOS chips [19] or other physical media [20]. Secondly, reservoir computing effectively established a bridge between a class of dynamical systems (exemplified by echo state networks and liquid state machines) and a class of physical systems that can act as reservoir computers [21].

In the following, we provide a perspective on the challenges faced by unconventional computing with physical substrates and an overview of implementations in photonic substrates. We start by outlining a general conceptual framework to clarify what we mean by computing with a physical system and present the physical reservoir computing paradigm as an example. We then proceed to describe several implementations of unconventional computing in photonic substrates and discuss them in the context of possible applications, going into enough technical detail to illustrate the nuances of this emerging field. We focus on photonic substrates as a platform with high potential for computing [16,22,23]. In particular, photonic implementations of neural network architectures potentially offer fundamental advantages over their electronic counterparts in terms of speed, processing parallelism, scalability and energy efficiency. We end by presenting strategies towards learning and full programmability of physical substrates, whose practical implementation is often elusive. Overall, our overview of unconventional computing in photonic substrates aims to provide a comprehensive perspective by combining abstract concepts with practical implementations, gaining a deeper understanding of the opportunities and challenges that lie ahead.

## 2. Unconventional computing with physical substrates

### 2.1. Introduction

Unconventional computing with physical substrates involves using a specifically chosen substrate to exploit its inherent information processing for solving complex problems. Consequently, physical phenomena such as nonlinear effects, amplification/absorption, distribution and various dynamical behavior transform the injected information while passing through the physical substrate. Thereby, the information processing appears in analog, noisy and often parallel fashion — clearly differentiating it from clocked, sequential digital computation.

To convert a physical substrate into an effective information processor requires an interface that encodes input information into a form that the substrate can interact with. The information may be carried by various media such as electrons, photons, or ions. The substrate's response to this information is typically complex and nonlinear, transforming the input into a high-dimensional state. A set of control parameters define the particular way how the input information flows through the physical substrate. Tuning these parameters will later allow for improved computational capabilities and, in turn, efficient use of the substrate. Finally, sensors need to extract the state of the physical substrate to measure the result of the processing and generate the desired solution. Thereby, designing a physical computer needs to be addressed from two sides. First, the design of a *hardware* that includes the physical substrate, its modulators, interfaces and sensors. Second, the definition and synthesis of computing paradigms that interpret the intrinsic physical information processing and guide towards solving the computational problem at hand.

Fig. 1 contrasts the process of conventional computer programming with the process of programming a physical substrate for unconventional computing. We identify three separate domains: the platonic domain of computational tasks or intentions, denoted by  $\Omega$ , the abstract domain of computations as represented by a program in some mathematical formalism, denoted by  $\lambda$ , and finally the tangible domain of physical systems that compute, denoted by  $\Psi$ . We refer to the boundary between the *platonic* domain of intentions or computational tasks and the formalized domain of computer programs as the *formalization barrier* because crossing this barrier requires the programmer to formalize their intention into a computer program in some formalism. We further refer to the boundary between the formalized program and the physical system as the *implementation barrier* because crossing this barrier requires the programmer to implement their abstract computation into a concrete physical system.

Based on this hierarchy, we identify four fundamentally different approaches to computing that differ in how they relate physics to theory of computations:

- **Physics-first** ( $\Psi \rightarrow \lambda$ ): *Bottom-up* approach that starts from the domain of the physical substrate, e.g., photonics, and builds computational models and theories from primitives and architectures that harness the substrate's inherent properties.
- **Theory-first** ( $\lambda \rightarrow \Psi$ ): *Top-down* approach that forges a physical substrate to match an existing theory of computation, like in digital computing where transistors – fundamentally analog devices – are used as binary switches for digital computation.
- **Substrate transfer** ( $\Psi_1 \rightarrow \lambda \rightarrow \Psi_2$ ): An existing physical computing system  $\Psi_1$  is viewed through the lens of some computational model or theory  $\lambda$ , which is then transferred to another physical substrate  $\Psi_2$ . For example, neuromorphic engineering models the brain with differential equations as a spiking neural network and then implements this model in CMOS chips.

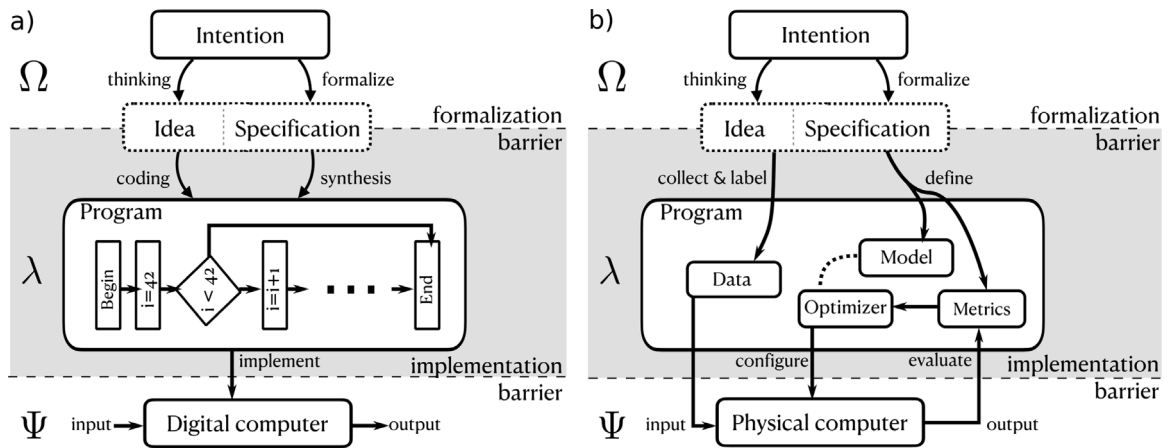


Fig. 1. Conceptual framework of the programming process for: (a) a digital computer using classical logic-based computational primitives, and (b) a physical substrate for unconventional computing using data-driven physical programming. See text for explanation.

- **Bridge ( $\Psi \leftrightarrow \lambda$ ):** Taking a bidirectional approach and finding an equivalence between a computational model and a class of physical substrates, e.g., physical reservoir computing matches echo state networks to the class of nonlinear physical systems with fading memory.

In the following section, we will discuss various hardware and software approaches for unconventional computing together with their potential field of application. We will start with how intrinsic information processing in physical substrates can be interpreted and adapted for computation.

## 2.2. Adapting physical systems for computing tasks

In conventional symbolic computing, we have a worked-out theory and an established instruction set architecture for computers that allows us to adjust the computer towards solving the computational task. This adjustment, referred to as *programming* can proceed in a predominantly linear way, as illustrated in Fig. 1(a). One starts with an abstract intention for what kind of computation is desired and proceeds to work out an idea or specification for how this computation can be implemented (which is strongly guided by both the theory of symbolic computation and the given digital hardware). This idea or specification is then formalized into a program written in a programming language, for example an imperative programming language that consists of a sequence of logical conditions like if-then, mathematical operations, and loops. It is worth noting that iterations do not modify the program itself. The last step, the implementation, is facilitated by a compilation hierarchy that bridges many levels of programming language abstractions into the physical machine language. Finally, the program is executed on the physical computer that deterministically follows the programmed instructions. Naturally, mismatches between the resulting computation and the original intention may occur, for example due to bugs in the program, mistakes in the task specification, or hardware limitations. The programmer may require multiple rounds of refinements to arrive at a satisfying computation.

In contrast, programming a physical substrate for desired computational tasks requires considering several new aspects, such as its analog nature, stochasticity, multiple physical timescales, and others. Accordingly, aside from a new formal theory for unconventional computing, a practical way of *making a physical system compute* is needed, which we refer to as “physical programming” in the following. The capacity to program a given physical system depends on various factors such as: how much is known about the physical substrate, *i.e.*, how detailed is the best model or simulation of this system? How much can we control about the system? How stable is the behavior of the physical substrate over time?

On first sight, the implementation step is where the difficulty of programming physical computers lies. There are mismatches between theoretical models and physical devices, as well as between devices themselves. This is due to the limited observability of the physical computer, device-to-device variation, noise, and device degradation. Such mismatches can be amended by a feedback loop between the desired program and the physical computer. Alternatively, they can be amended in the abstract computational domain alone, for example by designing a program that is robust to noise and device mismatch.

Another factor that shapes the programmability is the granularity of the physical system model, meaning the degree of control over the system: How many parameters or degrees of freedom can be tuned in the granularity of the system? How fast can these parameters be varied? Can the system be combined with other systems into a richer computing architecture? This ranges from no control *i.e.* fixed input–output systems, to limited parameter tuning *e.g.* a global bifurcation parameter in a dynamical system, to the possibility of local parameter tuning *e.g.* setting all synaptic weights in a neuromorphic chip and hence full access to the placement of primitives and dynamics.

Based on these factors, we argue that in unconventional computing approaches from a physical perspective, the distinction between program and substrate is blurred. In a way, the substrate is the program. Implementing physical computers requires us to

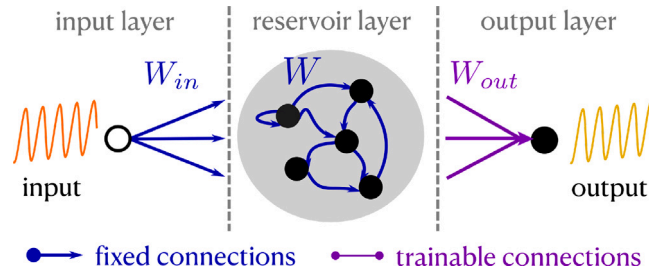


Fig. 2. A recurrent neural network used for time series prediction using the reservoir computing paradigm. The output weights  $W_{out}$  are trained to map the reservoir response onto the targeted output. The input weights  $W_{in}$  and internal weights  $W$  are initialized at random and kept fixed during training.

change the very language in which programs are expressed, thus further disrupting the largely linear process known from digital computer programming. Given the intention or specification of the computation that should be implemented into the physical system, alternative procedures for implementing, programming, and configuring this behavior into the system are needed.

To address this, advanced design and programming tools are being developed, e.g. by defining a set of computational primitives in integrated photonics [24,25] and electronic neuromorphic computing [26]. Nevertheless, much of current work focuses on end-to-end learning or optimization, where the physical system is configured through an automatic data-driven optimization procedure. Here, learning algorithms are the most prevalent way of “programming” systems with large numbers of tunable parameters in an automated way. Thereby, data is processed through the physical substrate and based on the measured response the configurations of the substrate are adapted towards a potentially better performing configuration. The performance is evaluated using predefined metrics that can be purely performance-based but also might consider aspects such as stability against device changes or generalization abilities. Given some error metric, an optimization algorithm is used to update the system’s configuration. This optimizer may or may not rely on a model of the physical computer. Model-free optimization is often less efficient but can increase the physical computer’s autonomy.

As depicted in Fig. 1(b), data-driven physical programming implements a loop of setting configurations, executing computation, evaluating error metrics and, finally, updating the configuration. These primitives place data-driven physical programming close to the recently derived field of deep learning and, more generally, differentiable programming [27–30]. One of the simplest and generic learning algorithms is the reservoir computing paradigm described in the next Section 2.3. This paradigm is suitable for most of the photonic implementations later discussed in Section 4, and provides a starting point for most researchers interested in unconventional computing with physical dynamical systems.

### 2.3. Random mappings for unconventional computing with physical substrates

Recent advances in computing with physical substrates have been heavily influenced by Reservoir Computing (RC) [31,32] and Extreme Learning Machines (ELM) [33–35]. These computing paradigms provide an excellent starting point for validating the computing capacity of unconventional computing hardware. Both RC and ELM exploit the computing capabilities of physical substrates based on unaltered system configurations and random input mapping. This significantly simplifies the hardware design process while still allowing the implementation of learning systems in hardware.

RC and ELM both leverage the concept of using random systems for computational purposes. ELMs are tailored for static systems, like feed-forward neural networks, and are predominantly used in the classification and regression of image and multivariate data. In contrast, RC (as depicted in Fig. 2) is designed for time series data, employing dynamical systems such as recurrent neural networks. In both methods the input data drives the physical system and the way the input information affects different parts of the system is chosen randomly. The state of the system is recorded and only a set of linear output weights is used to map the random nonlinear projection generated by the physical system towards the desired target data. Since this output layer is purely linear, the weights can be computed efficiently using, e.g., least squares or the FORCE algorithm [36]. During the training process the physical system’s configuration remains fixed. Therefore, both RC and ELM allow to use various systems that generate a nonlinear and high-dimensional mapping. Often reservoir systems offer a set of global parameters like the input gain or the feedback gain that allow to tune the response of the reservoir towards a more optimal configuration. The configuration of these parameters is often carried out using search methods such as grid search, random search or Bayesian optimization [37]. Different substrates have been exploited for RC and ELM (see e.g. [21,38] and references therein), with numerous examples in photonics [22,39–43].

## 3. Applications

Given that most implementations of unconventional computing with physical substrates are still being perfected, their applications can be seen as proofs of concept rather than fully-fledged industrially applicable solutions. We showcase four different

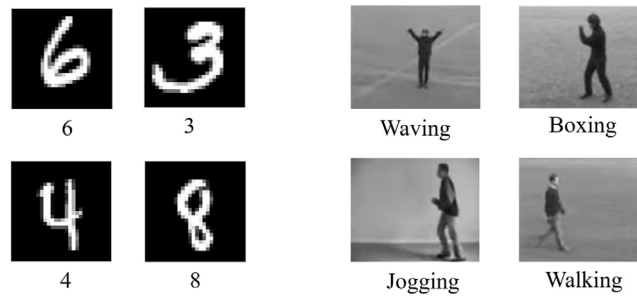


Fig. 3. Samples of input data taken from benchmark databases on image (left) and video (right) recognition, respectively the MNIST dataset [44] and the KTH dataset [45].

categories of applications that are relevant to photonic implementations of artificial neural networks, namely image processing, time-series prediction, telecommunication tasks, and hardware acceleration. In the following subsections, each of these applications are discussed.

### 3.1. Image and video processing

Artificial Neural Networks (ANNs) excel at computationally-hard problems such as pattern recognition and classification. In particular, ANNs have become mainstream for image and video processing. Several of the implementations discussed later in Section 4 have been tested in this context. For the sake of comparability, these unconventional computing systems are often evaluated on benchmark databases. Fig. 3 displays some samples of such data used in image or video classification, taken from benchmark datasets.

In the case of image processing, the MNIST database is widely used. This database contains 60,000 training images and 10,000 testing images of handwritten digits from 0 to 9, each with a dimension of  $28 \times 28$  pixels. It has been shown that photonic systems are capable of recognizing the MNIST digits with high speed and accuracy [46–48].

Differently from static image recognition, video recognition adds an additional challenge since the temporal relation between the frames becomes the key information that has to be somehow extracted by the ANN. In particular, human action recognition stands out as a benchmark task in the context of video processing [49]. Photonic systems have demonstrated real-time processing capabilities of human actions with state of the art performance [50,51].

### 3.2. Time-series prediction

Multiple information sources generate outputs in a sequential manner. These sequential outputs can be represented in the form of timeseries, whose prediction is a relevant field of application with examples such as the stock market or electrical load forecasting [52,53], among many others.

In the context of unconventional computing using photonic systems, several implementations have been demonstrated for timeseries prediction. Often, the demonstrations have focused on solving benchmark timeseries that are commonly used in the context of machine learning. These examples include the prediction of chaotic signals, which originate from high-dimensional and nonlinear systems. Examples of benchmarks for time-series prediction used in photonics include the so-called Mackey-Glass and Santa Fe timeseries, with the tasks being either one-step ahead prediction or autonomous continuation [54–56].

Fig. 4 shows an example of a time series derived from the Mackey-Glass model, which is often used as a benchmark in one-step-ahead forecasting tasks. If the quality of the one step ahead prediction is sufficient, the prediction of the system can be used as the next input to the system and eventually become a generative digital twin of the original timeseries [57,58].

While photonic systems have shown promising results in timeseries prediction, it is important to consider these achievements in the context of traditional statistical methods. These include techniques such as Nonlinear Vector Autoregression (NVAR) and Autoregressive Integrated Moving Average (ARIMA) models. For instance a recent optical hardware implementation that exploits the advantages of the NVAR model by directly generating nonlinear combinations of the input data with varying delays demonstrates the potential for synergies between these approaches and reservoir computing [59] (see [60] for the original proposal of this method). By acknowledging and building upon these ideas, one can hope to significantly improve the performance of photonic systems for time series prediction.

### 3.3. Telecommunication tasks

Optical communications are the backbone of today's highly connected world, enabling high-speed data transport over distances ranging from short distances in data centers to thousands of kilometers in submarine fiber-optic links. Indeed, optics boasts several characteristics that support this high throughput, including large bandwidths, low-loss channels, and multiplexing in amplitude,

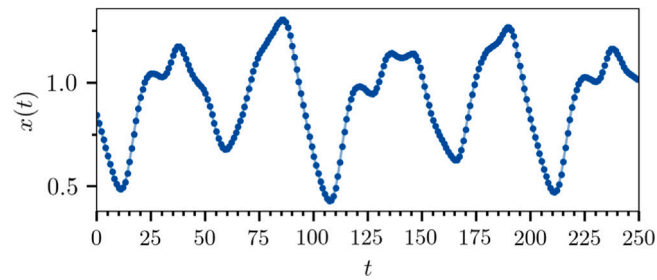


Fig. 4. Time series of the chaotic Mackey-Glass system, which is often used as a benchmark in forecasting tasks.

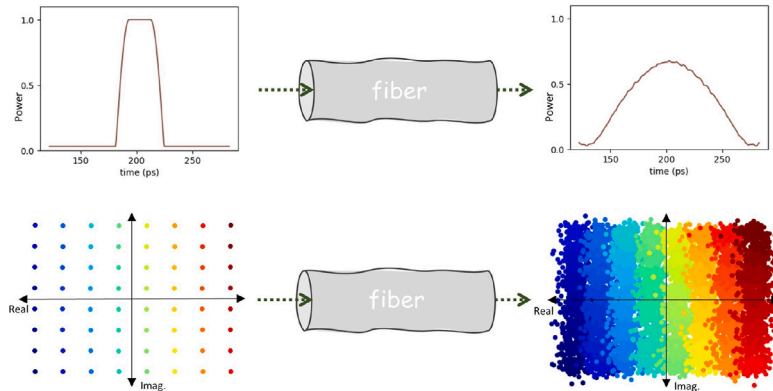


Fig. 5. Distortion of optical signals after propagating through an optical fiber. (top) Temporal broadening of a pulse due to dispersion. (bottom) Effect of mild distortions on a coherently modulated signal.

phase, space, polarization, and wavelength [61]. While commercial use of optical data transport started around 50 years ago, the throughput and reach of these systems was not what we know today. Both advances in individual components and in the migration of electrically implemented stages to the optical domain enabled the scaling of aggregate data rates from modest megabytes/s to today's terabytes/s and higher (see, e.g. [62,63] and references therein). Most notably, for the first 20 years, signals were regenerated on a span-by-span basis because optical amplification was not yet possible. The advent of erbium-doped amplifiers allowed signals to be amplified optically, not only saving costs and allowing longer-reach transmission, but also enabling the use of wavelength division multiplexing, which would not have been feasible if regeneration were necessary [62]. Unconventional computing could open new frontiers in optical telecommunications, enabling faster speeds, lower energy consumption, and more complex network management.

The idea to extend the advantages of optics from transmitting data to processing has been studied since the 1960s, and a variety of designs and architectures have been proposed [64]. The current resurgence of interest in optical computing is due to the shift of the research focus from replicating digital computers to development of special-purpose, analog systems. Evidently, higher speed and bandwidth of optical signals compared to electronics are attractive for signal processing. Moreover, while multiplexing is naturally implemented in optics, processing multiplexed signals in the electric domain poses a scaling problem. For example, a dual-polarization signal modulated in both phase and amplitude would require 4 analog-to-digital converters (ADC) and consecutive digital processing pipelines implemented for each of these signals. Compact, power-conservative optical solutions can, therefore, be potentially advantageous compared to their electronic counterparts, though, there are numerous challenges to be addressed in practical implementation.

Fiber-optic channel equalization – mitigation of the transmission signal distortions – is one of the important challenges in telecommunications. The key physical effects degrading quality of signal transmission in fiber channels are: chromatic and polarization mode dispersion, optical noise and fiber Kerr nonlinearity. Fig. 5 illustrates two simple examples of signal distortion during fiber propagation, with the upper panel showing the temporal effect of chromatic dispersion on an intensity modulated pulse after propagation through a short section of optical fiber. The lower panel shows mild distortions on a coherently modulated signal, where both the phase and amplitude of the optical carrier are used to transmit data. Depending on the number of distinct amplitude/phase combinations (referred to as symbols), multiple bits can be encoded on each symbol. The distorted post-fiber signal shows symbol overlapping causing wrong decoding and again leading to errors.

The linear dispersive effects (at low signal power) can be compensated by various all-optical methods, such as dispersion-compensation fibers [65], with the dispersion opposite to the transmission fibers, of fiber grating-based dispersion compensators [66]. However, optical dispersion compensation typically adds extra losses and introduces additional optical noise, thus

degrading the system performance. In the coherent detection systems chromatic and polarization mode dispersion can be compensated using digital electronics. After the equalization of linear effects, noise and nonlinearity become the principal factors deteriorating the performance of optical networks. Nonlinear channel equalization can also partially be done in the digital domain using digital signal processing and machine learning (see, e.g. [67–69] and references therein). To this end, several photonic solutions that support optical signal processing and targeted towards channel equalization have been proposed, commonly, but not necessarily, relying on machine learning concepts to enable this equalizing behavior [70–73].

Aside from the physical layer, photonic processing is also fundamental in enabling switching at the transport layer [74]. For example, Reconfigurable Optical Add-Drop Multiplexers (ROADMs) in optical networks facilitate efficient wavelength routing, boasting high throughput, low latency, and low power usage, surpassing electric counterparts [75]. Furthermore, current research efforts investigate the feasibility of performing optical packet switching. Similar to electric packet switching where routers need to recognize headers and route signals, optical headers may be recognized and routed. Optical header recognition is a challenging task, as it requires high speed, accuracy, and robustness to cope with the increasing data traffic and complexity of modern telecommunication systems. Photonic implementations of artificial neural networks have been successfully used in header recognition tasks [76], whose purpose is to identify and extract the header information from a data packet in telecommunications.

### 3.4. Hardware accelerators

A photonic hardware accelerator is a device that uses light to perform computations faster, with greater parallelism and energy efficiency than conventional electronic methods [77]. Matrix vector multiplication (MVM) is an example of an operation common to many computing tasks that can be very time-consuming and resource-intensive, especially for large and high-dimensional problems. Optics can offer a solution to MVM by using various techniques to encode, manipulate and decode the matrix and vector elements as optical signals, such as spatial light modulators, Mach–Zehnder interferometers or wavelength division multiplexing [78,79]. This allows optics to perform MVM in a single step, without the need for multiple arithmetic operations or memory access, with potential applications in machine learning, signal processing and data analysis. Some examples of photonic hardware accelerators will be presented later in Section 5.

## 4. Physics-driven implementations of unconventional computing substrates

### 4.1. Introduction

In this section, we consider photonic and optoelectronic implementations designed for general purpose information processing and computing that can in principle solve any of the tasks introduced in Section 3. The physical systems must satisfy some general conditions, such as having multiple independent degrees of freedom that can respond to the external information and that can be monitored. Given the richness of the processes that can be exploited for unconventional computing with physical substrates, it is common to classify different implementations in terms of the type of multiplexing that is employed for information encoding and decoding [80]. Space multiplexing refers to having different spatial locations perform different parts of the computation, which can then be combined. Time multiplexing is used when information is distributed over time, with different time slots encoding different parts of the computation. Frequency multiplexing refers to cases where the computation is performed by elements whose dynamics are spectrally separated. Finally, implementations based on the integration of photonic components on a chip are discussed separately. For the sake of readability, we abuse the notation typical of artificial neural networks to refer to the various parts of the physical systems described below.

### 4.2. Space multiplexing

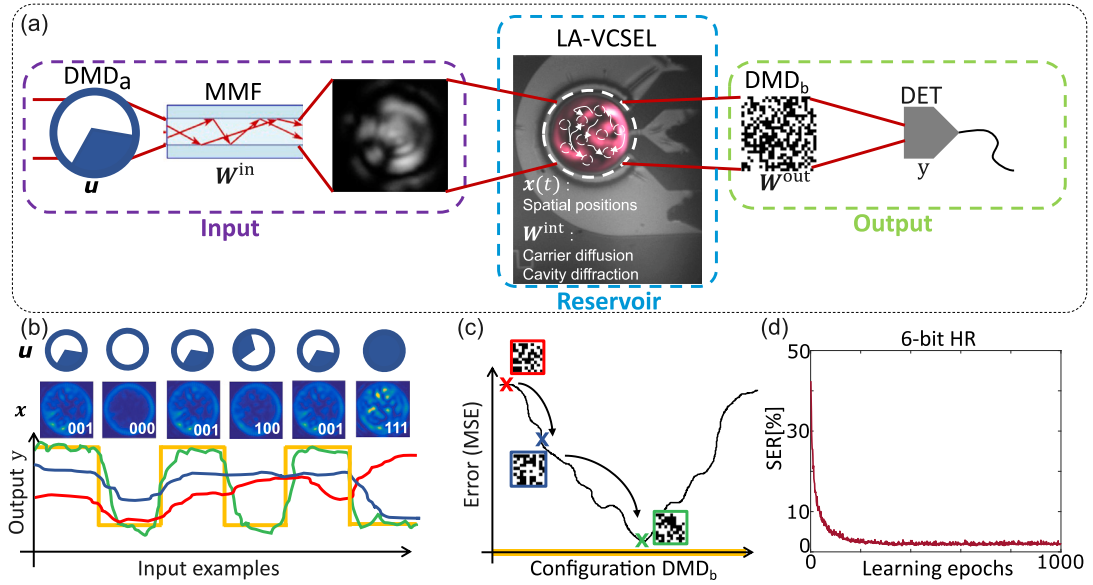
Space multiplexing is the most direct approach to translate an abstraction of computing to a physical implementation, where different parts of the system are in charge of different operations. We present here an example of an implementation where different parts of a continuous physical substrate correspond to the different nodes in a network.

Using large-area vertical-cavity surface-emitting lasers (LA-VCSELs), a truly parallel, spatially multiplexed, LA-VCSEL-based photonic neural network (PNN) based on the reservoir computing architecture was introduced in [81]. Due to parallelism, computing speed does not depend on the PNN's size. In addition, an online learning strategy can be implemented, making the system's computation fully autonomous and relegating the external computer to a simple supervision and instrument-control role.

The physical system implementing a PNN can be broken up into three functional sections, c.f. Fig. 6(a). First, the input layer is realized via a digital micromirror device (DMD<sub>a</sub>) and a multimode fibre (MMF). Spatial patterns (Boolean images) displayed on DMD<sub>a</sub> constitute the input information,  $\mathbf{u}$  in Fig. 6(a), and the MMF passively implements the PNN's input weights via its complex transmission matrix.

The second part is the nonlinear information processing device itself. The output field of the MMF is optically injected via imaging onto the LA-VCSEL top facet, which has an aperture diameter of 25  $\mu\text{m}$ . By exploiting the highly multimode nature of the LA-VCSEL, its spatio-temporal nonlinear dynamics form the core of the processing. Nodes are spatially multiplexed positions on the LA-VCSEL's surface, and coupling is taken care of by carrier diffusion and optical diffraction inside the LA-VCSEL's cavity. The state of the system is then the perturbed mode-profile of the LA-VCSEL under optical injection, denoted in Fig. 6(a) as  $\mathbf{x}$ . In Fig. 6(b), the VCSEL's response to such injection is shown for several 3-bit headers. Responses to each input pattern are complex and different.





**Fig. 6.** (a) Working Principle of the LA-VCSEL spatially multiplexed processing system. (b) Input information  $\mathbf{u}$  and the LA-VCSEL response for 3-bit binary headers. The graph shows the target output  $y^{\text{target}}$  and different outputs  $y^{\text{out}}$  of decreasing mean square error (MSE) (red, blue and green). (c) Schematic illustration of the error landscape, showing the MSE as a function of the output weights configuration. The outlined (red, blue and green) Boolean matrices correspond to the output weights giving the output from (b). (d) Representative performance of the PNN on a 6 bit header recognition (HR) task. Figure adapted with permission from [84] ©Optica Publishing Group. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

This explains, in an intuitive sense, how finding a configuration of output weights that solve a certain computational task like header recognition, XOR, and digital-analog conversion [81] is possible. Ultimately, the LA-VCSEL is used at speeds orders of magnitudes below its inherent timescales, being operated in its steady state, hence no recurrent properties such as fading memory of the device were exploited. In order to use the concept presented here for memory-dependent tasks such as time series prediction, one would have to encode the input information on the timescale of the VCSEL's intrinsic dynamics by using for instance, a gigahertz-rate modulator.

The last part of this PNN is its output layer with weights  $\mathbf{W}^{\text{out}}$ , which are realized by imaging the LA-VCSEL's near field onto a second DMD ( $\text{DMD}_b$ ). The reflection off  $\text{DMD}_b$  in one direction is imaged onto a large area detector, and the mirrors of  $\text{DMD}_b$  sample the different positions, i.e. neurons on the LA-VCSEL's surface. This applies a Boolean weight matrix to the state of the system, and  $\sim 90$  trainable Boolean readout weights are implemented. The output of the network  $y^{\text{out}}$  was recorded at the detector for a set of input patterns called the training batch. For each image  $I$  in the training batch, the desired target output  $y^{\text{target}}(I)$  is known. Thus, after each training epoch (a run of one batch),  $y^{\text{out}}$  is recorded and a normalized mean square error is calculated  $\epsilon_k = \|y_k^{\text{out}}(I) - y^{\text{target}}(I)\|$ . Training is realized via a simple, yet effective evolutionary algorithm presented in [82,83]. Boolean weights (mirrors) at random positions are flipped at the transition from  $k$  to  $k+1$ . If the change is beneficial, i.e.  $\epsilon_{k+1} < \epsilon_k$ , it is kept, otherwise the output weights are reset to the configuration at epoch  $k$  as shown in Fig. 6(b,c). This operation is repeated until the desired performance threshold is met. Fig. 6(d) shows a representative learning curve for a 6 bit header recognition task, for which the system reaches around a 1.5% symbol error rate (SER). The PNN was also tested on the MNIST handwritten digit data-set with promising initial results.

The main strength of this implementation is the potentially high inference/injection bandwidth, although it is currently limited to 15 kHz by the frame rate of the input DMD. In practice, the VCSEL's transformation occurs on the nanosecond timescale. Finally one crucial advantage of this approach is that increasing the size of the LA-VCSEL and therefore the size of the implemented PNN has virtually no impact on inference bandwidth.

Future improvements to this experimental setup include increasing the resolution of the output weights, implementing trainable input weights that map the input data onto the LA-VCSEL, as well as working on different more complex, hardware compatible online learning strategies.

### 4.3. Time multiplexing

The concept of using a single nonlinear node with delayed feedback for reservoir computing (RC) was pioneered in [85] and subsequently gained significant attention within the RC research field. This technique involves dividing the duration of a delay loop ( $\tau$ ) into  $N$  intervals, creating  $N$  virtual reservoir nodes that emulate a larger reservoir network. Through time multiplexing, input values are sampled, held for a duration of  $\tau$ , and then multiplied by a (usually) random mask. This mask expands the input into a

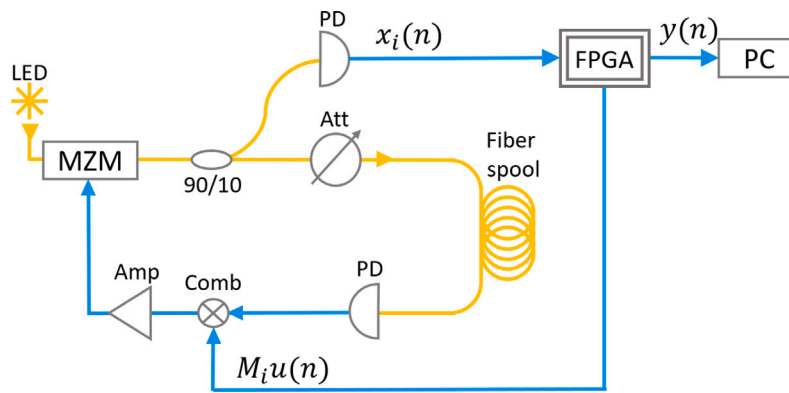


Fig. 7. Optoelectronic time-multiplexed reservoir computer. The fiber-based photonic part is in yellow, and the electronic part in blue. MZM: Mach-Zehnder intensity modulator. Att: optical attenuator. PD: photodetector. Comb: resistive combiner. Amp: amplifier. FPGA: Field Programmable Gate Array. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)  
Source: The figure is readapted from [50].

higher-dimensional space within the reservoir, facilitating easier linear data separation. The output layer in this delay-based RC can also be time-distributed, measuring the response of the reservoir at the time instants corresponding to the virtual nodes within each period of the input, allowing for time-demultiplexing the output. The following sections discuss two hardware implementations of delay-based RC.

#### 4.3.1. Optoelectronic feedback system

The experimental system described here is based on a time-delay reservoir computer [85] implemented with an optoelectronic system, as developed and studied in [86–89]. The analog reservoir states are represented by the light intensity in an optic fiber, while digital electronic hardware, i.e. a FPGA board, is used to interface the optical signal. As shown in Fig. 7, the optical part consists in an incoherent light source, a delay loop (fiber spool) and a nonlinear node (Mach-Zehnder Intensity Modulator). The FPGA acts as the input and output layer, driving the modulator with the input sequence, collecting the reservoir states and calculating the optimal readout weights. The system is similar to the one used in [57], where the electronics was the speed bottleneck of the optoelectronic system: this affects the reservoir computer's processing speed especially when processing complex data structures, such as images or videos. For this reason, the electronic interface was redesigned and an increase of speed of the overall system of two orders of magnitude was achieved: in this way, real-time data processing was achieved, even when investigating complex datasets [50] or more complex RC architectures [90]. In terms of flexibility, the FPGA is re-configurable and the setup can be modified according to the architecture or application under test. The size of the network (number of nodes, or neurons) can be increased with a longer fiber spool (i.e. longer delay) or a faster sampling frequency.

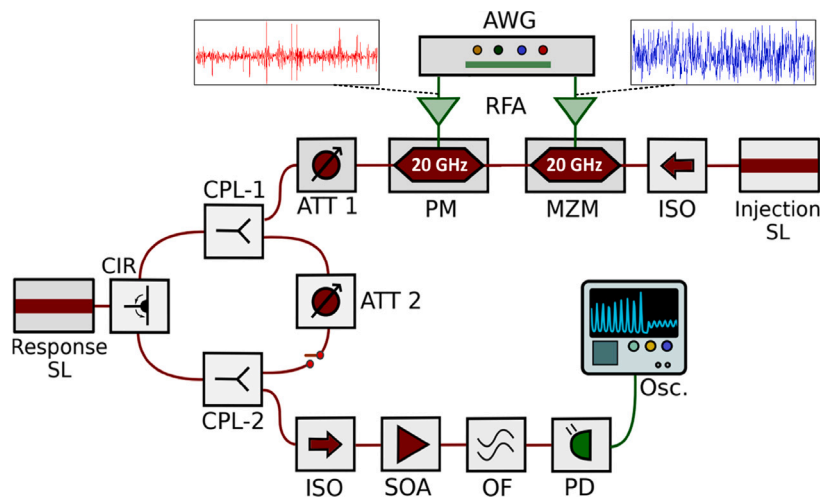
This system was successfully benchmarked on human action recognition in videos, with results comparable to up-to-date digital approaches [50]. The flexibility of this system makes it an ideal playground to test and benchmark different architectures and algorithms. The setup was modified to stack several reservoirs in series, thus creating a deep reservoir computer: this deep configuration was tested on tasks involving speech processing [90,91]. Moreover, the system was used to experimentally validate a novel optimization procedure for reservoir computers based on the use of a delayed input [92].

Future works on this system can focus on replacing the slower electronics with faster photonics to realize a high-speed fully photonic reservoir computer, or exploiting the setup's flexibility to investigate more ideas and procedures.

#### 4.3.2. All-optical feedback system

Semiconductor lasers (SLs) with delayed optical feedback and external optical injection have been validated as nonlinear temporal information transformers within the framework of physical reservoir computing [93,94]. This all-optical approach led to noteworthy advances in information processing with hardware systems across a variety of benchmark and real-world application tasks [95–98]. In almost all prior investigations of time delay reservoir computing (TDRC) systems, the encoded information to be processed was in the amplitude domain of the electrical field. Different variants have been introduced recently, which consider phase, or both amplitude and phase encoding of the input information using the appropriate modulation elements and by modifying TDRC configuration [99–101].

An example is shown in the experimental setup of Fig. 8, where the information is introduced via an amplitude and a phase modulator, both supporting high bandwidth encoding (here 20 GHz) [102]. When considering this topology for solving a data recovery task from fiber-optic transmission in an optical communications task, there is different information in the phase and the amplitude of the input signal. After photodetecting these two signals with a coherent receiver, the corresponding time-series (blue for amplitude and red for phase in Fig. 8) are independently introduced in the photonic reservoir. The length of the delay loop of the reservoir is mostly imposed by the fiber lengths of the fiber-based components and is equal to  $\tau = 24.5$  ns. This length allows for the definition of 418 virtual nodes separated by  $\theta = 58.6$  ps, in a time-multiplexed encoding approach. In synchronous encoding,



**Fig. 8.** Photonic TDRC with amplitude and phase encoding. ISO: optical isolator, RFA: RF amplifier, MZM: Mach-Zehnder modulator, PM: Phase modulator, AWG: arbitrary waveform generator, ATT: optical attenuator, CIR: optical circulator, SOA: semiconductor optical amplifier, OF: optical filter, PD: photoreceiver, Osc: Real-time oscilloscope. This figure refers to the system used in [102]. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

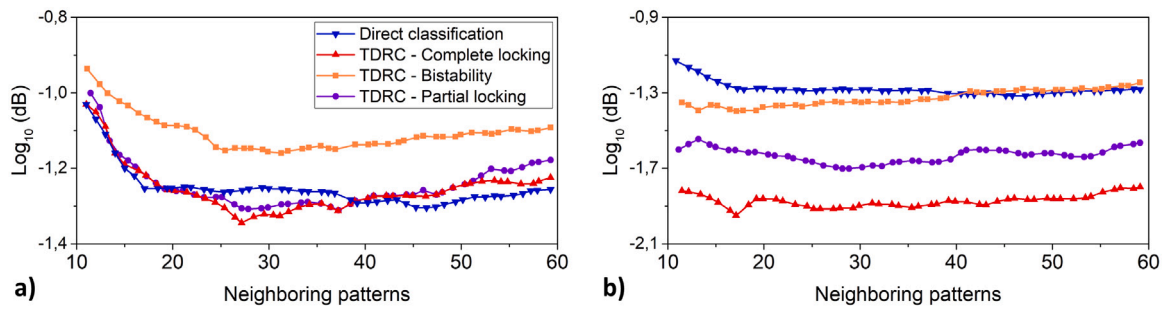
the feedback delay is matched with the information rate. This is implemented by encoding the information to be processed at a sampling rate of 17 GSa/s. Once both input time-series are pre-processed with a random masking sequence [85] and temporally stretched to match the time delay of the reservoir [95], they are loaded to the arbitrary waveform generator (AWG). The electrically generated signals are amplified (RFA) and inserted into the TDRC system, via the modulation elements: a Mach-Zehnder modulator for the amplitude (MZM), and a phase modulator (PM) for the phase information. This information is loaded to an injection optical carrier, created by an external injection SL. Via a 50/50 coupler (CPL-1), the optically injected carrier enters the feedback loop and undergoes a nonlinear transformation when interacting with the response SL. Part of the signal remains in the feedback loop, while the rest is photodetected to obtain the virtual nodes' responses. The dynamical properties of the TDRC system can be controlled via specific key parameters [103]. These are:

(a) The strength of optical feedback, which is regulated via optical attenuation (ATT2) and is influenced by the splitting ratios employed within the optical loop (CPL-1 and CPL-2). Large optical feedback levels enhance the system's capacity to retain previously processed information, but ultimately lead to coherence collapse of the optical carrier and a reduced consistency in the input-output transformation [103].

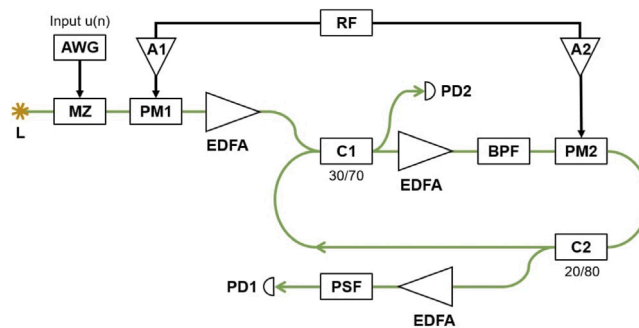
(b) The injection optical power, which determines the influence of the input information on the dynamical behavior of the system. Typically, the injection strength upon entry into CPL-1 is set to a range between 0.1–1 mW, with higher values being more efficient towards a bandwidth-enhanced system operation [104,105]. In this operating regime, the inherent chaotic instabilities resulting from strong feedback ratios can eventually be suppressed by the optical injection.

(c) The optical frequency detuning between the injection source and the response laser, which constitutes a critical parameter that defines the nonlinear interaction between the input information and the laser and cavity dynamics of the reservoir. Here the encoding plays a significant role. Amplitude encoding exhibits higher robustness when the frequency detuning is set to provide a dynamical setting that is characterized by partial locking of the two carriers. However, it is important to note that the TDRC configuration can be viewed as a hybrid coherent setup, where phase encoding also exerts an influence on the reservoir's dynamics and the resulting amplitude output.

The system presented in Fig. 8 (with the switch closed in the feedback loop) evaluated in a fiber-optic communication data recovery task, as a post-processing tool combined with a linear classifier. Specifically, a 50 km, 28 GBaud PAM-4 fiber transmission link, with high-launched optical power (10 dBm) was considered. This link suffers from chromatic dispersion and self-phase modulation, severely distorting the initially encoded signal, which was a pseudorandom bit sequence encoded at the emitter of the transmission system. The amplitude and phase information of the propagated carrier at the end of the link were coherently detected and fed into the reservoir through phase and intensity modulation simultaneously. However, only the amplitude of the TDRC output carrier is photo-detected and used to train a linear classifier for the data recovery task. In Fig. 9 we show an example of the computational performance of the TDRC, when considering only the amplitude information encoding from the transmission (Fig. 9, a), and when considering both the amplitude and phase information (Fig. 9, b). There is a clear advantage when also using the phase information, as the lowest logarithmic bit error rate (BER) improves from  $-1.35$  to  $-2$ . This also happens for a significantly smaller number of neighbors, which determines the number of bits considered for the output layer, allowing the linear classifier to operate faster. In both cases, for this experimental setup, the optimal operational point of the TDRC was for frequency detuning values that corresponded to a complete injection locking between the injection and the reservoir SL (Fig. 9, red line). These results



**Fig. 9.** Data recovery performance of the TDRC system shown in Fig. 8, when applying (a) only amplitude information encoding, and (b) both amplitude and information encoding, versus the number of neighboring responses (taps) considered in the linear classifier. The blue line corresponds to the performance of the classifier, excluding the TDRC. The remaining colored lines represent different dynamical operating regimes of the reservoir. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 10.** Schematic of the frequency multiplexing reservoir computer experiment reported in [109]. Black lines: electrical connections. Green lines: polarization maintaining optical fiber. L: laser. AWG: Arbitrary Waveform Generator. MZ: Mach Zehnder modulator. PM1 and PM2: Phase Modulators. EDFA: Erbium Doped Fiber Amplifier. A1 and A2: RF Amplifiers. C1 and C2: Couplers. BPF: Band Pass Filter. PSF: Programmable Spectral Filter. PD1 and PD2: Photodiodes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

also validate the capability of the TDRC to offer efficient post-processing in comparison with a solitary linear classifier (Fig. 9, blue line).

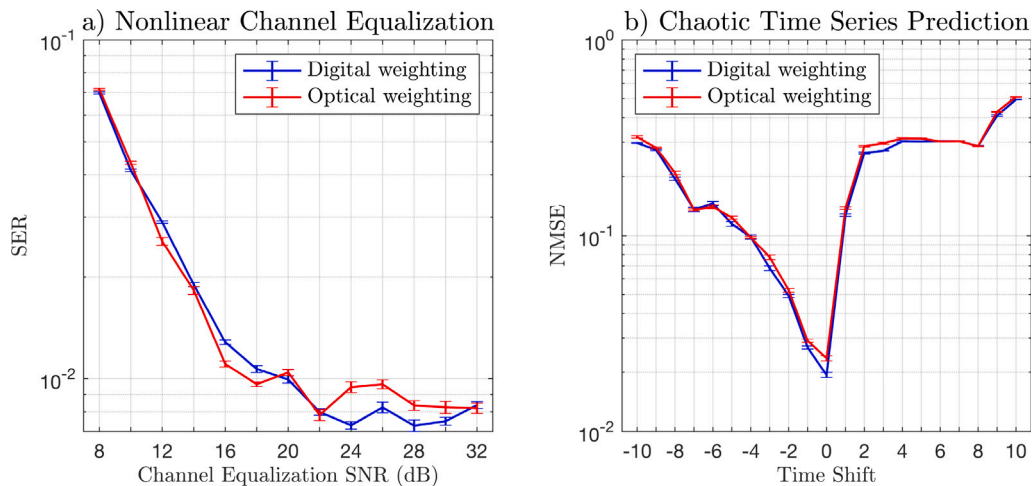
The configuration depicted in Fig. 8 can be translated into a photonic integrated circuit with much shorter lengths, as the number of virtual nodes that are needed for data recovery tasks is not so high (typically around 20) [101]. In addition, the use of ultrafast encoding, facilitated by amplitude and phase, allows the definition of numerous virtual nodes along a very short delay line, in a physical path of a few centimetres. Notably, such an approach will substantially mitigate all the phase instabilities, inherent to phase modulation techniques, that have been observed in the system of Fig. 8.

#### 4.4. Frequency multiplexing

Encoding signals in a frequency-multiplexed fashion is an interesting alternative to both space and time multiplexing. The main advantage is that frequency-multiplexed signals, being encoded in different wavelengths, can coexist in the same waveguide without the need for time multiplexing. This enhances scalability both in terms of space occupation and processing speed. Moreover, frequency multiplexed signals can be conveniently weighted in the optical domain by spectral filtering. Different schemes based on frequency multiplexing have been explored for optical neuromorphic computing, especially to achieve high processing speed by exploiting parallelism [73,106–108].

We present here a frequency multiplexed reservoir computer in which the reservoir states are encoded in an optical frequency comb, meaning that each comb line encodes a reservoir node (or neuron) signal in its complex amplitude [109]. Signals encoded in different comb lines are mixed by frequency-domain interference, employing a scheme based on radio-frequency phase modulation already explored in quantum optics [110,111]. The recurrence is achieved by circulating the frequency comb in a fiber loop. In this way, at each round-trip, the reservoir state evolves keeping a (fading) memory of the past states. A feed-forward processing scheme can be obtained too, simply by removing the fiber loop. The input signal is supplied by modulating the input comb. The main nonlinearity is introduced in the read-out phase, by a photodiode measuring optical intensities (signals are encoded in amplitude, thus giving a quadratic nonlinearity). By spectral filtering, the neuron signals in each comb line can either be measured individually or can be weighted and summed together in the optical domain.

The experimental scheme is reported in Fig. 10 and refers to the RC experiment described in Ref. [109]. The input light source is a C-band CW narrow-band laser. The radiation is modulated by a Mach-Zehnder modulator (MZ) according to the input signal



**Fig. 11.** Summary of the results obtained by the frequency multiplexing reservoir computer experiment described in Fig. 10 [109]. Panel (a): nonlinear channel equalization task. Results are plotted as Symbol Error Rate (SER) vs. Signal to Noise Ratio (SNR) in the channel. Panel (b): chaotic time series prediction. Results are plotted as Normalized Mean Square Error (NMSE) vs. shift in time series. Positive shifts correspond to predicting the future, while negative shifts correspond to remembering the past.

$u(n)$  generated by an Arbitrary Waveform Generator (AWG). The input  $u(n)$  is held constant for a time equal to the round trip time of the fiber loop, which is approximately 50 ns (corresponding to a frequency of 20 MHz). The monochromatic radiation encoding  $u(n)$  passes through a Phase Modulator (PM1), thereby generating a frequency comb. PM1 is driven by a periodic RF signal at frequency  $\Omega \approx 20$  GHz, amplified by Amplifier A1. The frequency  $\Omega$  defines the comb spacing. An EDFA increases the optical power to approximately 17 dBm before injecting the signal into the fiber loop. The loop contains a second EDFA and a second Phase Modulator (PM2). PM2 is driven by the same RF signal driving PM1, amplified by Amplifier A2. Since PM2 acts on radiation already featuring a comb-like spectrum, the phase modulation results in line interference. Part of the radiation leaves the fiber loop through Coupler C2 and reaches the output layer. In the output layer, the radiation is amplified by an EDFA, reaching an optical power of 10 dBm, filtered by a Programmable Spectral Filter (PSF – Finisar Waveshaper) and read by the photodiode PD1. The PSF either selects comb lines individually, by setting bandpass filters or applies the output weights in the optical domain, by applying the correct attenuation on each line. The entire system is realized in polarization-maintaining single-mode fiber, and the optical path is stabilized against acoustic noise and thermal drift.

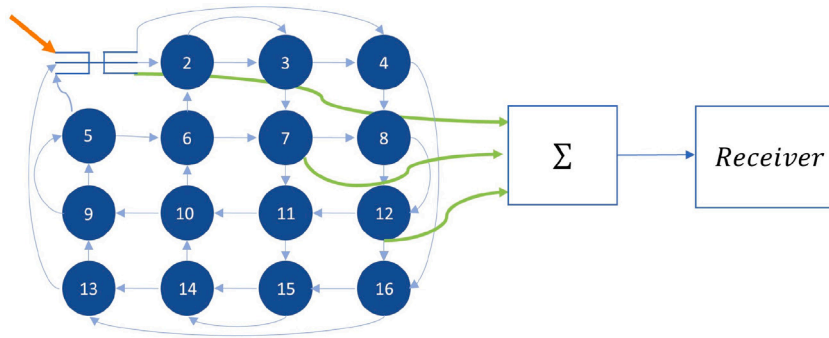
If the fiber loop is opened, the system is no longer recursive, i.e. it does not have memory about past inputs. This configuration has been employed to implement a randomized feed-forward neural network model, also known as Extreme Learning Machine (ELM) [112].

The frequency degree of freedom can be exploited to multiplex entire reservoirs (or neuron layers), by encoding them in different frequency combs, provided that the spectral separation is large enough to guarantee no interference among different combs. In this case, multiple reservoir computers [113] or multiple extreme learning machines [114] can be executed on the same experimental setup. In the case of reservoir computing, multiple parallel reservoirs can be combined in series (using the output of one reservoir to generate the input of the following one) achieving a more complex architecture known as deep reservoir computer [113,115].

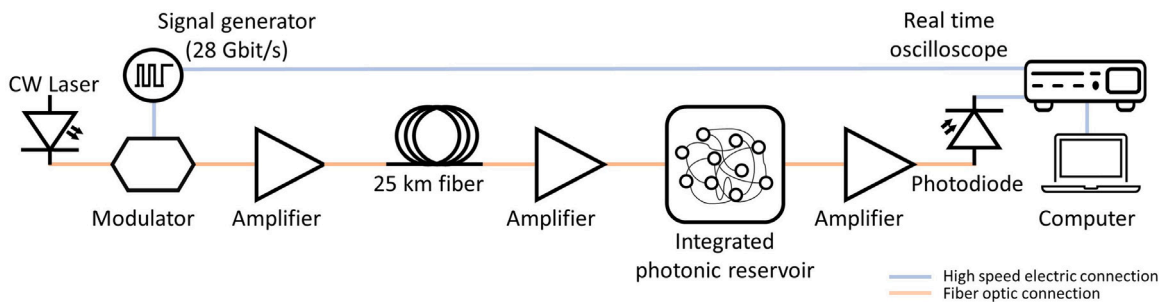
Frequency multiplexed reservoir computing has been tested on multiple time series processing tasks, including nonlinear channel equalization and time series prediction. The first task consists of retrieving the original signal after the propagation in a nonlinear and noisy communication channel, while the second task consists of predicting the future evolution or remembering the past history of a chaotic dynamical system. Fig. 11 contains the results obtained on the two mentioned tasks. Full results are reported in Ref. [109].

#### 4.5. Towards integration

Integrated photonics enables combining optical components like lasers, modulators, waveguides, and detectors on a single compact chip [116]. While efforts in this direction are as old as electronic integrated circuits, the scale at which this field grew is much slower. As such, photonic integrated circuits are not as ubiquitous, and are primarily deployed for optical communications as well as for other smaller markets, such as for various sensing applications. Ultimately, the goal is to fabricate chips on which light can be generated, modulated, transported, processed, and detected using the same integrated circuit. Nonetheless, current maturity in this field has facilitated computing with light in small compact circuits by leveraging aspects like reservoir computing, neural networks, and programmable photonics [47,117,118]. Below we present two designs of photonic integrated circuits for unconventional computing, the first of which has been experimentally tested.



**Fig. 12.** 16 node reservoir based on  $3 \times 3$  MMI nodes. Besides node 1, which shows the abstract structure of an MMI, nodes are shown as numbered vertices. The orange arrow shows a node's external input port used to inject an external input signal into the reservoir. The green arrows show three nodes' external output ports connected to the readout for weighting and summing, followed by a receiver. The nodes are interconnected in a topology that allows each node to have two links from and two links to other nodes in the reservoir. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 13.** Experimental setup for signal equalization using the silicon photonic chip of Fig. 12. Intensity modulated signals are propagated through a 25 km fiber and then equalized by the photonic reservoir.  
Source: Adapted from [120].

#### 4.5.1. Passive silicon photonics chip

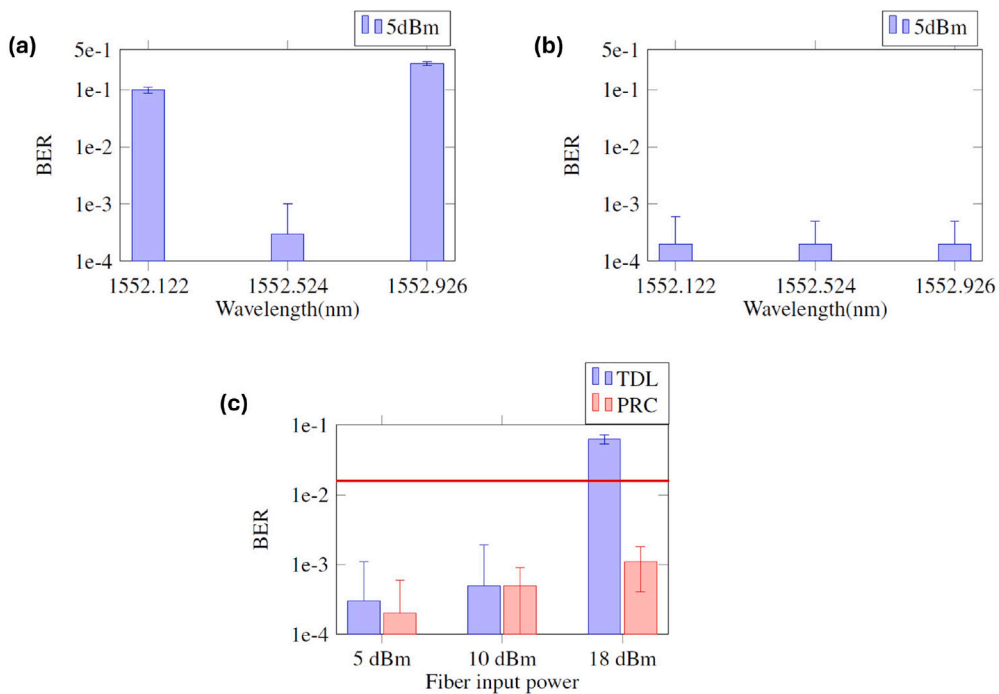
Fig. 12 shows a schematic for a passive Silicon-Nitride photonic chip that can be used to perform channel equalization tasks, which belong to the family of optical telecommunication applications described in Section 3.3. The reservoir nodes are realized through multi-mode interferometers (MMIs) with three input ports and three output ports [119]. Two of the input ports feed into the node from within the reservoir, while the third port allows injecting an external signal into the reservoir. Similarly, two of the output ports lead into the reservoir while the third port is connected to trainable weighting elements that allow manipulating both the amplitude and phase of this readout signal. The inter-node connections are realized through spiralled waveguides that introduce appropriate delays to the signal. Additionally, the waveguides alter the signal's amplitude through propagation losses and introduce random phases due to the waveguide's sidewall roughness.

The passive nature of this photonic chip is a power advantage, since it does not require driving with external power. Furthermore, the linearity of the reservoir in the complex domain simplifies its training when solving linear tasks like the equalization of chromatic dispersion in fiber optic systems. This task requires optical "memory", which is supported through the spiralled waveguides that keep the signals in the reservoir for some time and allow meaningful mixing between new input signals and these in-memory signals.

To solve nonlinear tasks, a nonlinear receiver such as a photodiode – typically used in such photonic systems – is leveraged. The most basic form of nonlinearity in the receiver is the inherent conversion it performs from complex-valued optical fields to real-valued intensities. The reservoir, as a trainable network, pre-distorts the signal before a nonlinear kernel (i.e., the receiver) with the final target of solving a nonlinear task. By relying on such external nonlinearity, the reservoir's fabrication and operation are simplified at no additional system costs [76].

This architecture has been shown, in both simulation and experiments, to solve linear and nonlinear tasks, both for fiber communications and for other benchmark applications. For example, the experimental setup in Fig. 13 shows the photonic reservoir chip in a fiber optic system deploying intensity modulated signals at 28 Gbaud. The signals traverse a 25 km fiber that causes linear distortions, and, if enough power is injected, nonlinear distortions as well. Experimental results showed successful mitigation of both forms of distortion, outperforming an electronic feed-forward equalizer [120].

A similar setup was also used to showcase the reservoir's ability to process several signals with the same trained readout weights, which makes it an attractive solution for wavelength division multiplexed systems [121]. In this experiment, instead of transmitting at 1550 nm, the carrier frequency is swept over a range of wavelengths. The reservoir is then trained to equalize all wavelengths with



**Fig. 14.** Results for the setup of Fig. 13 for wavelength multiplexed communication. (a) A poor performance is obtained when the readout is trained for one wavelength and used on unseen wavelengths. (b) Training implemented for all wavelengths results in a single readout that performs well on all. (c) Comparing reservoir solution to tapped delay lines on multiple input powers. The red line indicates the required bit error rate threshold for forward error correction in the optical transport system under study. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Source: Adapted from [121].

the same set of reservoir weights. The results of this experiment are shown in Fig. 14. The effect of training the readout for a single wavelength and then using this readout for other wavelengths is shown in panel a. For the trained central frequency, the system performs well while for the other two frequencies its performance is low. Panel b shows that by changing the training technique to incorporate data from all wavelengths, a readout that performs well for the three wavelengths can be achieved. These results are compared to a tapped delay line (TDL) using different fiber input powers. The plot in panel c shows the significant advantage of the reservoir in high power and nonlinear transmission scenarios, even when multiple wavelength training is used, as opposed to the TDL, despite the TDL solution being trained on the specific wavelength shown.

The experimental results of signal equalization were carried out using offline training. This is done through measuring the optical states of individual nodes and then performing the weighted summation digitally. This was necessary in early generations of the reservoir chip due to some lossy on-chip components. As such, the weighting elements and summation tree had to be bypassed to avoid losing more power and thus rendering the signal undetectable. However, recently, online and on-chip training were experimentally demonstrated for header recognition and delayed XOR on new generation chips [122].

Finally, numerical results for coherently modulated signals were obtained using the same reservoir architecture [123]. The receiver of this setup was based on the recently proposed Kramers–Kronig receiver, where a single photodiode is used to detect a complex signal. This is an example of a self-coherent receiver, where an unmodulated wave propagates with the modulated carrier, saving receiver costs by eliminating the need for a local oscillator at the receiver. However, this approach makes the signal sensitive to nonlinear effects in the fiber due to the high propagation power. The results of this simulation show that the reservoir system can be leveraged to simultaneously perform bandpass filtering and nonlinearity mitigation for a 64-QAM signal at a 64 Gbaud modulation rate [123]. These results underpin the reservoir’s ability to be trained for performing simultaneous tasks on higher-order modulation formats in both amplitude and phase.

#### 4.5.2. Coupled optical resonators

Evanescence coupled integrated nonlinear optical resonators have been proposed in [124] to implement a space-multiplexed reservoir computer. The reservoir layer is composed of a grid of resonators with spacing small enough to overlap their modes, inducing evanescent coupling. As shown in Fig. 15, a waveguide is coupled to resonators at the edge of the grid to inject an input signal. Resonators act as band-pass filters that pick up spectral parts of an input that are mixed by nonlinearities, ensuring a complex response of the reservoir to an excitation. The readout layer is implemented by a set of waveguides that extract the activation of the optical resonators (“neurons”) by coupling. Then, each signal is passed through a dedicated attenuator and a phase shifter to

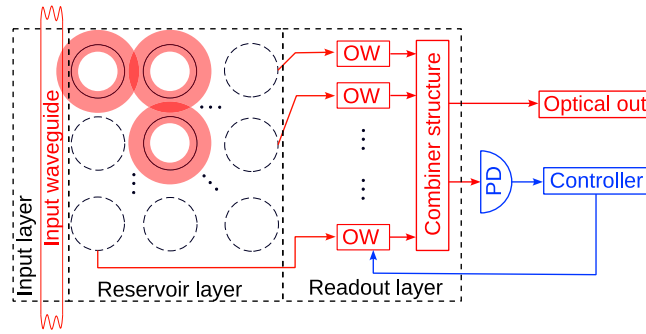


Fig. 15. Reservoir computer based on evanescently coupled optical resonators. Here, OW is an optical weighting element. Due to resonators' dense packing, waveguides can only extract signals from outer resonators (as drawn). This issue can be solved by multiple photonic layers [125] or three-dimensional structures [126].

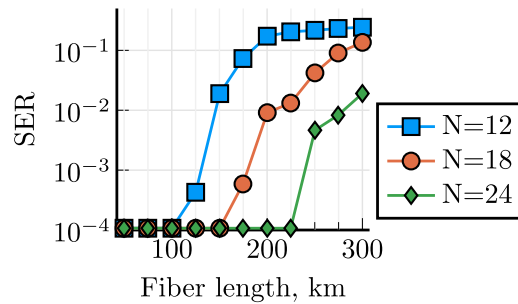


Fig. 16. Numerical results for the scaling of channel equalization performance with the number of resonators in the reservoir ( $N$ ) using the system described in Fig. 15. Here, SER is the symbol error rate. The model of the system can be found in [124].

perform all-optical weighting, and a combiner tree performs summation. As reservoir responses depend solely on the input signal, training can be efficiently carried out with ridge regression. This way, parallel computing is carried out fully autonomously on-chip.

Linear and nonlinear timescales of this photonic reservoir determine the processing speed. Linear timescales are defined by the photon lifetime, which sets the memory length, and the evanescent coupling strength that allows the reservoir to accept input signals with higher bandwidth. Nonlinear timescales vary between different physical processes. For example, free carrier dispersion depends on electron lifetime, which, in turn, depends on the resonator geometry and the material used.

There are three main potential advantages of this setup, namely (i) high-speed real-time inference that can reach tens of gigahertz, (ii) compact reservoir layout promises high scalability (thousands of resonators per square millimeter), and (iii) excessive optoelectronic conversions can be avoided by eliminating the use of electronics in the computing process.

Using numerical simulations, the system illustrated in Fig. 15 was applied to an optical telecommunications task, namely post-processing signals from up to 300 km, 25 GBaud OOK transmission link with the same launch power (10 dBm). The distortion was numerically simulated, and noise was not included in the model. The distorted signal was fed directly into the reservoir via the input waveguide. Fig. 16 shows that the reservoir successfully recovers the signal. Here, the reservoir operated in a linear regime, i.e. nonlinear distortion is not as severe, and its mitigation could be offloaded to a linear classifier.

The system described in Fig. 15 was also applied to a 3-step prediction of the Mackey-Glass equation using numerical simulations. For the generation of Mackey-Glass data, the chosen parameters were  $\tau = 17$ ,  $\alpha = 0.2$ ,  $n = 10$  and  $\gamma = 0.1$ , which result in mildly chaotic behavior of the solution. The signal timescale was scaled such that its bandwidth was roughly 12 GHz, and the sampling time corresponded to 17 ps (approximately 60 GHz sampling). The results obtained when considering nonlinearities that are common in integrated photonic platforms are shown in Fig. 17. For this task, two-photon absorption seems to lead to better prediction accuracy. However, free carrier dispersion and the Kerr effect could allow for a higher output signal power due to the higher dimensionality of the reservoir layer, see [124] for details.

## 5. Theory-driven implementations of unconventional computing substrates: Towards industrial applications

### 5.1. Introduction

We now present two examples of photonic implementations of unconventional computing with physical substrates that are driven by the knowledge that machine learning models of interest rely on certain mathematical operations such as vector-matrix



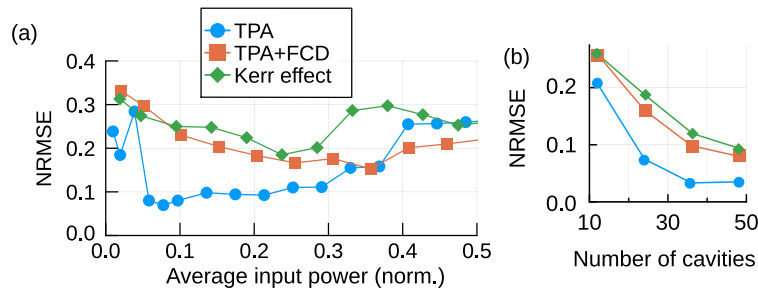


Fig. 17. Numerical results for 3 step ahead prediction of the MG time series. (a) Positive impact of nonlinearities on prediction accuracy with 24 resonators. (b) Scalability of prediction accuracy with the number of resonators at the optimal input power. Here, TPA is two-photon absorption, and FCD is free carrier dispersion.

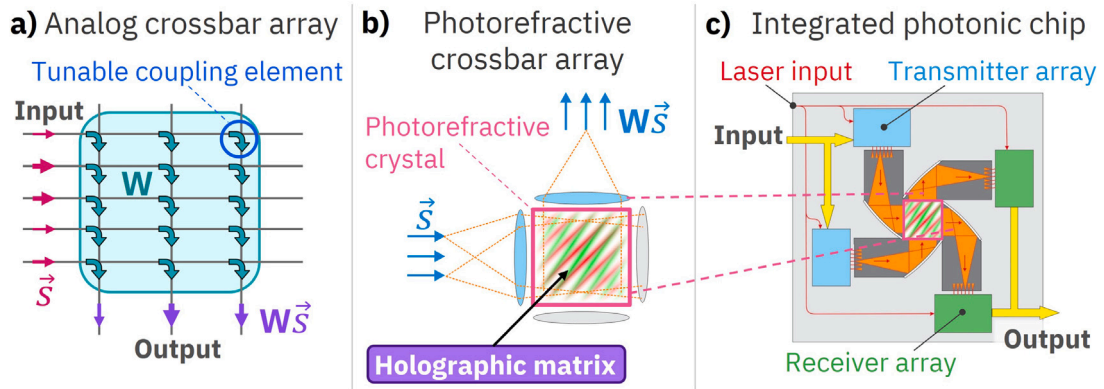


Fig. 18. (a) The analog crossbar array concept. The analog amplitudes of vector  $\vec{s}$  are transferred to output signal lines by tunable coupling elements into which the matrix  $W$  is programmed, thereby, yielding the matrix–vector multiplication  $W\vec{s}$ . (b) A crossbar array hardware concept. The photorefractive effect is leveraged to adaptively diffract laser beams by a hologram to output vector channels. (c) The photorefractive crossbar array can be implemented in integrated photonics including all core optical components and electrical I/O. Source: Adapted from [129].

multiplications or convolutions. This is in contrast to general-purpose implementations such as reservoir computers, to which applications are matched subsequently. In theory-driven implementations of unconventional computing substrates, the hardware is specifically engineered to accelerate the desired operation while targeting a precise application.

Photonics has been deployed with particular success to accelerate linear algebraic operations like vector matrix multiplication, temporal convolution, and Fourier transformation [79,107]. Photonics excels at linear signal processing (but not only) since optical propagation can occur linearly at high modulation rates and with low latency. Since the wavelength of the optical signal carrier is typically in the micron range, beaming effects can be exploited on relatively small length scales, for instance to perform the Fourier transform by a collimation lens. In addition, wavelength multiplexing techniques may be used to enhance parallelism.

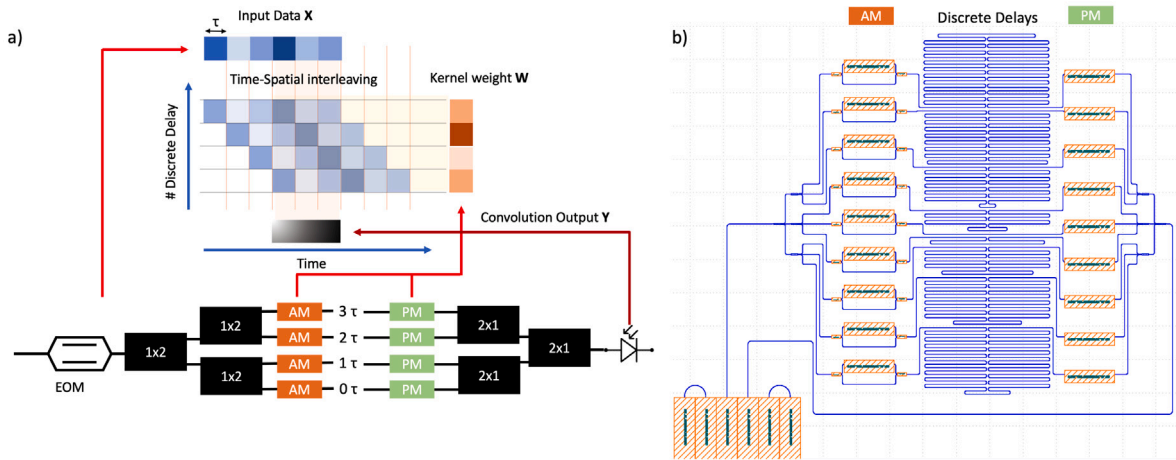
Linear operations are a cornerstone in several computationally heavy tasks including neural network inferences, backpropagation training, convolutional filtering, and optical signal processing [127,128]. The two examples below describe photonic implementations for vector matrix multiplication and convolution processors, respectively, based on photonic integrated technologies.

## 5.2. Matrix–vector multiplications

Matrix–vector multiplications (MVM) in machine learning can be efficiently accelerated in the analog domain by a crossbar array architecture, as illustrated in Fig. 18(a) [129]. Importantly, MVMs represent the synaptic signal transfer between neuron layers in artificial neural networks. The crossbar array allows for MVM with linear power scaling on vector length, whereas digital computing implies quadratic scaling.

The ideal hardware design of an analog crossbar array is subject to scientific investigation [130]. Here, we discuss in more detail the photonic design illustrated in Fig. 18(b) and (c) [129,131]. Vector coefficients are represented by the complex optical amplitudes of laser beams. The laser beams are made to interfere inside a photorefractive crystal, thereby, a hologram forms. The hologram can be programmed to diffract light from the input vector to the output vector in accordance with the intended matrix. Below we elaborate on the key benefits of this design.

Optical holography is attractive since it allows for high density storage of the matrix coefficients [132]. Spatial scaling is crucial to enable large matrices to fully benefit from computational parallelism, and to keep signal paths short to limit propagation loss.



**Fig. 19.** (a) Architecture of the 1D discrete convolutional network consisting of an electro-optical Mach–Zehnder Modulator (EOM), the time-spatial convolutional system, and a photodetector. (b) Layout of the convolutional integrated photonic accelerator that can be used for a 9-element 1D kernel or a  $3 \times 3$  2D kernel.

Next, the photorefractive crossbar array supports all matrix operations for backpropagation training of neural networks, which require the transposed matrix operation and the matrix update by the vector outer product [129]. Neural network training is accelerated efficiently by strictly storing the matrix coefficients locally and by avoiding weight update errors with respect to the loss gradient.

Lastly, the photorefractive hologram formation is governed by the redistribution of trapped electrons by photoexcitation [133]. This implies that the process is nondestructive and completely reversible. Additionally, due to the large number of electrons involved, an almost continuous formation of photorefractive synapses is possible, which enables incremental weight updates for the convergence of neural network training.

The photorefractive crossbar array described here is still under development, and currently the basic operating mechanisms have been demonstrated, including the integrated photorefractive crossbar circuitry and individual analog coupling elements with the desired tuning behavior [129,134].

### 5.3. Temporal convolutional processing

Convolution filters have become a standard part of the pipeline for image processing using NNs, often known in this context as convolutional neural networks. The characteristics of the convolution operation make it suitable for hardware acceleration using optics.

Integrated photonic circuits can accelerate convolutions through e.g. space–time multiplexing. In this case, streams of information can be encoded in the temporal domain and a photonic integrated circuit can convolve input bits with high speed and low latency [107]. Fig. 19(a) shows a simplified schematic that depicts a time-spatial convolutional accelerator implementing either a 4-element convolutional 1D kernel vector or a  $2 \times 2$  2D kernel matrix. For applications like image classification, the architecture requires pre- and post-processing of the input image. The input data is electronically projected into a vector ( $X$ ) and then optically encoded using an electro-optic Mach–Zehnder Modulator, which modulates a coherent monochromatic beam to represent each symbol in a time slot ( $\tau$  representing the slot duration). The convolutional photonic integrated circuit discussed here comprises cascaded splitter stages arranged in a tree configuration, spatially dividing the input signal into  $K$  optical paths ( $K$  being the dimension of the convolutional kernel). Each path includes amplitude and phase modulators to configure a complex-valued element of the vector kernel weight ( $W$ ). Amplitude modulators set the absolute value of the kernel element, while phase modulators determine the sign and establish phase-matching conditions for constructive or destructive interference. A cascaded network is then used to recombine spatial contributions into a single output, performing spatial–temporal interleaving through a sliding kernel operation. The modulators in the paths execute dot product operations, and the combination of splitters, delays, and combiners performs sliding summation. Finally, a photodetector measures the output convolution intensities vector ( $Y$ ). A schematic of the designed chip is shown in Fig. 19(b), which can be used as a 9-element 1D kernel or a  $3 \times 3$  2D kernel. The fabricated chip has been preliminarily tested with promising results [135].

## 6. Benchmarking

Benchmarking is a challenging aspect of unconventional computing approaches, especially when compared to conventional digital computing. For example, effective computational performance is determined by both the time required to complete a given task and the desired accuracy of the result, which creates ambiguity for physically based analog computers. Another example of

**Table 1**

Figures of merit of selected unconventional computing implementations. Abbreviations: Field Programmable Gate Array (FPGA), Reservoir Computing (RC), Extreme Learning Machine (ELM), Matrix Vector Multiplication (MVM), Feed-forward neural network (FNN), Backpropagation through time (BPTT).

System	Algorithm	# neurons	Processing time	Energy consumption	Footprint
High speed Opto-Electronic Delay System [105]	RC	30–2000	0.3–20 ns	1 W + electronics	–
Opto-Electronic Delay System + FPGA [50]	RC	50–600	2–200 ms	0,14 W (optical)+ 12 W (electronics)	30 * 100 cm <sup>2</sup>
Integrated Opto-Electronic with feedback laser [54]	RC	0–250	254 ps	–	~1 cm <sup>2</sup> (for chip)
Evanescence coupling between resonators (numerical simulation) [124]	RC	12–48	40 ps	–	<1 cm <sup>2</sup> (for chip)
Optoelectronic using Wavelength Multiplexing (RC) [109]	RC	20–40	50 ns	1200 W (including electronics)	60 * 80 cm <sup>2</sup>
Optoelectronic using Wavelength Multiplexing (ELM) [112]	ELM	20–60	0.5 s	1600 W (including electronics)	100 * 50 cm <sup>2</sup>
Vector Matrix Multiplication using integrated photonics (predicted) [129]	MVM	1000 * 1000	1 ns	20 W	1.6 * 1.6 cm <sup>2</sup>
Optoelectronic event-based neuromorphic system + offline spiking neurons [136]	FNN	500–1000	1–100 ms	–	50 * 10 cm <sup>2</sup>
MMI-based Passive Integrated Photonic Reservoir	RC	4–40	<0.5 ns	120 mW–140 mW (including electronics)	0.3 cm <sup>2</sup> –5 cm <sup>2</sup>

potentially ill-defined comparisons is getting a fair estimate of the computational efficiency of an experimental proof of concept. For example, experimental peripheral hardware is typically highly simplified and optimized for commercial implementations, but may impose a large overhead on the energy efficiency achieved by technologies under development, resulting in potentially misleading comparisons.

For the reasons stated above, we restrict our benchmarking to some of those systems presented in Sections 4 and 5, providing when possible further comparisons in the general context of the field of unconventional computing with photonic substrates. We compare the different unconventional computation solutions with respect to the type of computational paradigm implemented, the number of nodes (or “neurons”), processing speed, estimated energy consumption, and footprint. Other figures of merit that one could consider include latency, reconfigurability, scalability, hardware complexity, robustness (stability), life time (endurance), performance on some standard tasks, cost, or maturity of the underlying technology. A schematic comparison is given in Table 1.

Most of the implementations presented on Table 1 are based in the Reservoir Computing paradigm due to the simplicity of its training mechanism. The number of nodes or neurons in these systems ranges from order 10 to 1000. Concerning processing rates, the fastest implemented systems are the reservoir computers that can process one input every ~300 ps, and the vector matrix multiplication accelerator that is estimated to take approximately 1 ns to process an input. The slowest system in this comparison processes one input every 0.5 s (due to the very slow refresh rate of the spectral filter used in the experiment). Concerning energy consumption, a wide variety of results are reported. Many of the systems described are optical, and one needs to separate the energy consumption of the optical systems from the energy consumption of the drivers and current sources, and of the supporting electronics. Because these are proof of principle demonstrators, no real effort has been put in minimizing the overall energy consumption. In this context, the passive integrated reservoir is the most energy-efficient system of the ones presented in Table 1. Concerning footprint, a big difference can be made between systems using integrated optics in which case the footprint of the optical system is on the order of 1 cm<sup>2</sup>, and tabletop experiments in which case the footprint is a fraction of a m<sup>2</sup>. We note that these figures of merit are interconnected, with integrated systems typically having a smaller number of nodes, a lower energy consumption, and a reduced footprint.

Finally, the performance of these physical computing systems has been benchmarked on several tasks. In particular, for one-step ahead time-series prediction in the Santa Fe dataset (see Section 3.2), accuracies in the range of NMSE ~0.01–0.1 have been achieved in several experimental implementations [54,105,109]. For human action recognition (see Section 3.1), accuracies above 90% have been reported [50,51]. Because the computation of these systems is analog at its core, accuracy and computation speed can be traded off [137].

## 7. Programmability and learning in physical substrates for unconventional computing

### 7.1. Introduction

In the previous sections, we discussed various hardware implementations that perform computations based on the RC and ELM paradigm. As such, only the output weights are trained whereas global parameters such as the input or feedback gain are optimized using grid search. However, many of the systems presented have more fine-grained parameters that directly influence the information processing. In the RC and ELM paradigms, they are set randomly. Optimizing these fine-grained parameters promises improved

computational capabilities, but yet there is a lack of efficient ways to do that. Accordingly, we now discuss physical programming methods beyond the random mappings introduced in Section 2.3. Here, physical programming refers to the training methods that allow to learn how the parameters of the physical substrate need to be changed for a given computing task, assuming that these parameters can be modified. Thereby, we distinguish different approaches depending on the level of knowledge about the physical substrate, ranging from fully modeled to partially model-free to completely model-free optimization.

Within the order from model-based to model-free methods, we indicate several trends. Whereas model-based methods allow for efficient optimization once an accurate model is developed, these methods need a digital computer to run that exact model. Accordingly, the analog unconventional physical substrate may not become autonomous because it relies on a digital computing companion. On the other hand, model-free methods might potentially run almost without the assistance of a digital computer and hence being more autonomous. However, the lack of a model implies so far to process huge amounts of data for their optimization, rendering them inefficient. Each physical substrate might have its own optimal optimization algorithm depending on its complexity, inference speed and energy cost. Furthermore, besides better optimization strategies to generate more complex unconventional physical computers, new computing paradigms and abstractions are needed as we describe in the next subsections.

### 7.2. Model-based physical programming

In case there exists a model that describes the input-driven behavior of the physical substrate accurately, an optimizer might rely on this model to perform an accurate identification of which changes need to be made in the physical substrate for a specific outcome. Accordingly, such model-based programming can offer an efficient way of optimization that is related to the credit assignment problem in machine learning. Furthermore, specialized software tools can be used in a digital computer to first model the behavior of the physical substrate before optimizing it. Thereby, both learning the model and optimizing the physical substrate can rely on the well-established backpropagation algorithm. As shown in [138,139], running the forward pass on the hardware while computing the backward-pass digitally yields promising results on various platforms and domains. Besides the ability to program a huge set of parameters efficiently, these methods need a digital computer accompanying the physical substrate thereby reducing its autonomy. Furthermore, the backpropagation algorithm especially in combination with recurrent systems is computationally expensive and can therefore increase the energy costs of the potentially low-energy physical substrate.

### 7.3. Partially model-free physical programming

Often, especially for more complex substrates, a full model is not available and only some features or dependencies of the substrate are known. This could include the (approximate) shape of the nonlinearity, or the (approximate) energy-function that underlies the relaxation of the system. Recent methods like augmented direct feedback alignment [140] and equilibrium propagation [141] exploit this partial knowledge to optimize the computing system.

Strongly inspired by the gradient-based backpropagation algorithm, augmented direct feedback alignment, introduced in [140], replaces parts of the backward propagation by random projections. Nevertheless, partial model knowledge about the nonlinearity is needed to compute approximated gradients. Using this method, deep architectures in photonic and optoelectronic hardware can be trained.

Recent methods, like equilibrium propagation (EP) proposed in [141], integrate inference and learning in a single system. Inspired by contrastive hebbian learning, in EP two phases are performed. The first phase, called inference phase, is carried out without external teacher signal and the second is the training phase in which the system is slightly adapted using a teacher signal. Based on the differences in the solutions (fixed points) obtained during the two phases, the weights can be updated requiring only local information. Thereby, both phases rely on the same system reducing the complexity of the optimization process. Further advancements reduce the need for the exact energy-function and optimize a system while being agnostic of its underlying dynamics [142].

Interestingly, the simplifications made on the model required for learning in partial-model free methods suggest that they could be fully implemented physically. Instead of relying on a digital twin model within a digital computer, one can directly use the model's information to design a physical learning circuit. There are promising advancements in this area, particularly for EP, as highlighted by [143,144].

### 7.4. Model-free physical programming

Due to the complexity of physical processes and substrate structure, often there is no, or only a very inaccurate, model available that describes the dynamics of the hardware substrate. To optimize such setups beyond RC/ELM, the so called black box optimization algorithms need to be applied. These algorithms need to update parameters mainly based on an error value related to the current parameter configuration. Here, simple sampling based methods can be exploited such as extensive tree search. However, a usually large amount of measurements around the systems parameter setting are required to update towards a potentially better configuration, which render these rather brute force search algorithms inefficient. Recently, evolutionary strategies that implement reinforcement learning are being explored. These strategies exhibit higher sampling efficiency than brute-force methods such as random search and tree search. Methods like Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES) [145] and Parameter Exploring Policy Gradient (PEPG) [146] extract more information about the dependencies of each single parameter related to the loss and adapt the search space based on that information.

### 7.5. Computational abstractions beyond end-to-end learning

Section 2.2 already hinted at the existence of programming methods beyond learning algorithms. Although machine learning, including prompt programming, is contributing much to the modern landscape of computer systems, the majority of software systems are programmed manually using high-level programming languages. Clearly, such systems can reach levels of complexity and expressiveness that are unprecedented in unconventional computing with physical substrates.

A key challenge of programming physical systems is the non-existence of abstraction mechanisms for designing and reasoning about non-symbolic physical systems. Defining powerful abstractions for unconventional computing with physical substrates in generality likely requires a new fundamental theory of generalized computing [3,147]. For specialized cases, such as spiking neural networks, steps have been made towards defining a common representation language that serves as a starting point for future work in this direction [26]. We believe that a promising step for photonic computing is the careful definition of computational primitives that can be composed into functional units.

## 8. Conclusion

Exploring the dynamical richness within physical systems represents a promising avenue for advancing the field of computation. The strengths discussed in the previous sections underscore the potentially transformative impact of exploiting physical systems for computational purposes. First and foremost, achieving a high degree of control and programmability of physical systems will provide a robust foundation for the design of novel computing architectures. The ability to manipulate and control these systems opens up new possibilities for tailoring computing processes to specific needs, paving the way for increased flexibility and adaptability in unconventional computing paradigms.

The potential of photonic systems stands out as a remarkable opportunity, combining the possibilities of high-speed information processing and communication. The unique characteristics of photons make them a compelling candidate to push the boundaries of conventional computing and enable innovative solutions in diverse application domains, offering low latency and low power consumption solutions. In particular, the combination of multiplexing techniques further enhances the efficiency and capacity of photonics for unconventional computing. By exploiting time, frequency, and space multiplexing strategies, it becomes possible to optimize information processing and transmission, enabling a more streamlined and resource-efficient computational experience.

Cascadability emerges as a key challenge that would allow the seamless integration of multiple physical systems. This property would not only facilitate the scalability of unconventional computing architectures, but also promote the development of complex and interconnected systems, contributing to the creation of more sophisticated computational frameworks. In particular, the interconnectivity between physical substrates of different nature is key, fostering interdisciplinary collaborations and encouraging the integration of diverse scientific principles into computing. This interconnectivity broadens the scope of unconventional computing and opens avenues for cross-disciplinary research and innovation.

Finally, let us step back to the general framework of the power of different computational approaches. This is the topic of computational complexity [148], which compares different computational architectures and different computational tasks at a high level. This field focuses on the number of operations required to solve a task, and in particular the asymptotic number of operations required when the task complexity grows. To simplify the considerations, in this field polynomial overheads in the number of operations are neglected. Thus  $P$  is the class of problems that can be solved using a polynomial number (in the problem description size) of operations on a classical computer,  $NP$  the class of problems that can be verified using a polynomial number of operations on a classical computer,  $BQP$  the class of problems that can be solved with high probability using a polynomial number of operations on a quantum computer (which may be larger than  $P$ , i.e. quantum computing may be able to solve some problems with exponentially fewer operations than classical computers). In the present review, we have focused on classical computing with physical substrates. It is anticipated that all problems that can be efficiently solved in this approach will belong to the class  $P$ . Indeed, classical physical systems could in principle be efficiently modeled on a classical digital computer using polynomially small space and time discretization and polynomial precision. Hence they could be simulated on a digital computer with polynomial overhead. For specific physical systems, more efficient digital models than this naïve discretization can generally be built. However, in complex physical systems, the accumulated errors from approximations can significantly impede accurate simulation. For instance, in analog CMOS chip design, even sophisticated simulation tools cannot capture every nuance of real-world chip performance, thus necessitating physical prototyping to validate chip functionality. Thus from the point of view of computational complexity, computing with classical physical substrates may not change the class of tractable problems in general (although this may be different when using quantum systems to compute), but *some*  $NP$ -hard problems may become tractable through physical implementation. In addition, the gains provided by computing with physical systems may be substantial: polynomials with large coefficients may relate the resources and efficiency of the physics-based computation and its best digital emulation. Finally, we note that the physical energy of computation is not explicitly modeled in computational complexity theory, and time and memory are often poor substitutes. For all these reasons, a general theory of what gains are possible is still lacking.

Importantly, the new approaches to computing and to algorithmic design that are necessary in the physics-based approach may lead to new algorithms that are of interest in themselves for classical digital computing. Two examples will illustrate this last point. First, artificial neural networks, as they are currently used, are a digital abstraction of biological neural networks and provide a compelling example of this conceptual transfer from physics (or, in this case, biology) based computing to the digital world. Second, coherent Ising machines, which are photonic systems designed to solve  $NP$ -hard problems (although they almost certainly

cannot solve all instances of these combinatorial problems) [149,150] (and see [151] for a general review of Ising machines), have inspired interesting digital algorithms to solve the same problems [152,153].

In conclusion, the implementations and concepts highlighted in this work collectively underscore the potential of unconventional computing with physical substrates. As the capabilities of these systems are explored and harnessed, we anticipate transformative advances that may redefine the landscape of computing and push the boundaries of what is currently achievable. The integration of control, programmability, multiplexing techniques, and connectivity will position physics-based computing at the frontier of innovation.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Acknowledgments

The present review is largely based on the workshops and brainstorming that took place within the POSTDIGITAL project. POSTDIGITAL aimed, through a multidisciplinary approach, to address the challenges of computing with physical substrates. We are very grateful to Herbert Jaeger for his help and advice during the POSTDIGITAL project.

We acknowledge financial support by the European Union's Horizon 2020 program under the Marie Skłodowska-Curie grant agreement No. 860360 (POST DIGITAL), by the Fonds de la Recherche Scientifique - FNRS, Belgium under funding EOS n° 40007536 and CDR n° 40021243, and by the Spanish State Research Agency, through the INFOLANET and María de Maeztu projects (PID2022-139409NB-I00 and CEX2021-001164-M) funded by the MCIN/AEI/10.13039/501100011033.

### References

- [1] C. Horsman, S. Stepney, R.C. Wagner, V. Kendon, When does a physical system compute? *Proc. R. Soc. A* 470 (2014) 20140182.
- [2] H. Jaeger, Toward a generalized theory comprising digital, neuromorphic, and unconventional computing, *Neuromorphic Comput. Eng.* 1 (2021) 012002.
- [3] H. Jaeger, B. Noheda, W. van der Wiel, Toward a formal theory for computing machines made out of whatever physics offers, *Nature Commun.* 14 (2023) 4911.
- [4] H. Jaeger, B. Noheda, W.G. van der Wiel, Toward a formal theory for computing machines made out of whatever physics offers: extended version, 2023, arXiv:2307.15408.
- [5] T. Nakagaki, H. Yamada, Á. Tóth, Maze-solving by an amoeboid organism, *Nature* 407 (2000) 470.
- [6] V. Bonifaci, K. Mehlhorn, G. Varma, Physarum can compute shortest paths, *J. Theoret. Biol.* 309 (2012) 121.
- [7] A.M. Turing, On computable numbers, with an application to the Entscheidungsproblem, *Proc. Lond. Math. Soc.* 42 (1936) 230.
- [8] D. Deutsch, Quantum theory, the church-turing principle and the universal quantum computer, *Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* 400 (1985) 97.
- [9] C. Mead, How we created neuromorphic engineering, *Nat. Electron.* 3 (2020) 434.
- [10] S. Hooker, The hardware lottery, *Commun. ACM* 64 (2021) 58.
- [11] J. Laydevant, L.G. Wright, T. Wang, P.L. McMahon, The hardware is the software, *Neuron* 112 (2024) 180.
- [12] K.H. Lundberg, The history of analog computing: introduction to the special section, *IEEE Control Syst. Mag.* 25 (2005) 22.
- [13] B.J. MacLennan, Analog computation, in: R.A. Meyers (Ed.), *Encyclopedia of Complexity and Systems Science*, Springer New York, New York, NY, 2009, pp. 271–294.
- [14] J. Hasler, Opportunities in physical computing driven by analog realization, in: 2016 IEEE International Conference on Rebooting Computing, ICRC, IEEE, 2016, pp. 1–8.
- [15] M. Dale, J.F. Miller, S. Stepney, Reservoir computing as a model for in-materio computing, in: A. Adamatzky (Ed.), *Advances in Unconventional Computing: Volume 1: Theory*, Springer International Publishing, Cham, 2017, pp. 533–571.
- [16] P.L. McMahon, The physics of optical computing, *Nat. Rev. Phys.* 5 (2023) 717.
- [17] D. Hassabis, D. Kumaran, C. Summerfield, M. Botvinick, Neuroscience-inspired artificial intelligence, *Neuron* 95 (2017) 245.
- [18] R.A. Brooks, Intelligence without reason, in: *Proceedings of the 12th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'91*, Morgan Kaufmann Publishers Inc., 1991, pp. 569–595.
- [19] D. Zndrikov, S. Solinas, G. Indiveri, Brain-inspired methods for achieving robust computation in heterogeneous mixed-signal neuromorphic processing systems, *Neuromorphic Comput. Eng.* 3 (2023) 034002.
- [20] D. Marković, A. Mizrahi, D. Querlioz, J. Grollier, Physics for neuromorphic computing, *Nat. Rev. Phys.* 2 (2020) 499.
- [21] G. Tanaka, T. Yamane, J.B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, A. Hirose, Recent advances in physical reservoir computing: A review, *Neural Netw.* 115 (2019) 100.
- [22] A. Lugnan, A. Katumba, F. Laporte, M. Freiberger, S. Sackesyn, C. Ma, E. Gooskens, J. Dambre, P. Bienstman, Photonic neuromorphic information processing and reservoir computing, *APL Photonics* 5 (2020).
- [23] B.J. Shastri, A.N. Tait, T. Ferreira de Lima, W.H. Pernice, H. Bhaskaran, C.D. Wright, P.R. Prucnal, Photonics for artificial intelligence and neuromorphic computing, *Nat. Photonics* 15 (2021) 102.
- [24] Y. Jiang, P.T. DeVore, B. Jalali, Analog optical computing primitives in silicon photonics, *Opt. Lett.* 41 (2016) 1273.
- [25] D. Pérez, I. Gasulla, P.D. Mahapatra, J. Capmany, Principles, fundamentals, and applications of programmable integrated photonics, *Adv. Opt. Photonics* 12 (2020) 709.
- [26] J.E. Pedersen, S. Abreu, M. Jobst, G. Lenz, V. Fra, F.C. Bauer, D.R. Muir, P. Zhou, B. Vogginger, K. Heckel, et al., Neuromorphic intermediate representation: A unified instruction set for interoperable brain-inspired computing, 2023, arXiv preprint arXiv:2311.14641.
- [27] Y. LeCun, Deep learning est mort. vive différentiable programming, 2018.
- [28] F. Wang, J. Decker, X. Wu, G. Essertel, T. Rompf, Backpropagation with callbacks: Foundations for efficient and expressive differentiable programming, *Adv. Neural Inf. Process. Syst.* 31 (2018).
- [29] C. Olah, Neural networks, types, and functional programming, 2015.

- [30] M. Innes, A. Edelman, K. Fischer, C. Rackauckas, E. Saba, V.B. Shah, W. Tebbutt, A differentiable programming system to bridge machine learning and scientific computing, 2019, arXiv preprint [arXiv:1907.07587](https://arxiv.org/abs/1907.07587).
- [31] H. Jaeger, The Echo State Approach To Analysing and Training Recurrent Neural Networks-with an Erratum Note, Vol. 148, German National Research Center for Information Technology GMD Technical Report, Bonn, Germany, 2001, p. 13.
- [32] K. Nakajima, I. Fischer, Reservoir Computing, Springer, 2021.
- [33] G.-B. Huang, L. Chen, C.K. Siew, et al., Universal approximation using incremental constructive feedforward networks with random hidden nodes, *IEEE Trans. Neural Netw.* 17 (2006) 879.
- [34] G.-B. Huang, L. Chen, Convex incremental extreme learning machine, *Neurocomputing* 70 (2007) 3056, neural Network Applications in Electrical Engineering Selected papers from the 3rd International Work-Conference on Artificial Neural Networks (IWANN 2005).
- [35] G.-B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, *IEEE Trans. Syst. Man Cybern. B* 42 (2012) 513.
- [36] D. Sussillo, L.F. Abbott, Generating coherent patterns of activity from chaotic neural networks, *Neuron* 63 (2009) 544.
- [37] J. Yperman, T. Becker, Bayesian optimization of hyper-parameters in reservoir computing, 2016, arXiv preprint [arXiv:1611.05193](https://arxiv.org/abs/1611.05193).
- [38] C. López, Artificial intelligence and advanced materials, *Adv. Mater.* 35 (2023) 2208683.
- [39] G. Van der Sande, D. Brunner, M.C. Soriano, Advances in photonic reservoir computing, *Nanophotonics* 6 (2017) 561.
- [40] M. Sorokina, S. Sergejev, S. Turitsyn, Fiber echo state network analogue for high-bandwidth dual-quadrature signal processing, *Opt. Express* 27 (2019) 2387.
- [41] M. Sorokina, Dispersion-managed fiber echo state network analogue with high (including thz) bandwidth, *J. Lightwave Technol.* 38 (2020) 3209.
- [42] T. Hülser, F. Köster, L. Jaurigue, K. Lüdge, Role of delay-times in delay-based photonic reservoir computing, *Opt. Mater. Express* 12 (2022) 1214.
- [43] D. Pierangeli, G. Marucci, C. Conti, Photonic extreme learning machine by free-space optical propagation, *Photon. Res.* 9 (2021) 1446.
- [44] L. Deng, The mnist database of handwritten digit images for machine learning research, *IEEE Signal Process. Mag.* 29 (2012) 141.
- [45] C. Schuldt, I. Laptev, B. Caputo, Recognizing human actions: a local svm approach, in: Proceedings of the 17th International Conference on Pattern Recognition, 2004, ICPR 2004, Vol. 3, 2004, pp. 32–36, Vol. 3.
- [46] P. Antonik, N. Marsal, D. Rontani, Large-scale spatiotemporal photonic reservoir computer for image classification, *IEEE J. Sel. Top. Quantum Electron.* 26 (2019) 7700812.
- [47] M. Nakajima, K. Tanaka, T. Hashimoto, Scalable reservoir computing on coherent linear photonic processor, *Commun. Phys.* 4 (2021) 20.
- [48] H. Zhu, J. Zou, H. Zhang, Y. Shi, S. Luo, N. Wang, H. Cai, L. Wan, B. Wang, X. Jiang, et al., Space-efficient optical computing with an integrated chip diffractive neural network, *Nat. Commun.* 13 (2022) 1044.
- [49] D. Wu, N. Sharma, M. Blumenstein, Recent advances in video-based human action recognition using deep learning: A review, in: 2017 International Joint Conference on Neural Networks, IJCNN, IEEE, 2017, pp. 2865–2872.
- [50] E. Picco, P. Antonik, S. Massar, High speed human action recognition using a photonic reservoir computer, *Neural Netw.* 165 (2023) 662.
- [51] P. Antonik, N. Marsal, D. Brunner, D. Rontani, Human action recognition with a large-scale brain-inspired photonic computer, *Nat. Mach. Intell.* 1 (2019) 530.
- [52] O.B. Sezer, M.U. Gudelek, A.M. Ozbayoglu, Financial time series forecasting with deep learning: A systematic literature review: 2005–2019, *Appl. Soft Comput.* 90 (2020) 106181.
- [53] M.Q. Raza, A. Khosravi, A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings, *Renew. Sustain. Energy Rev.* 50 (2015) 1352.
- [54] K. Takano, C. Sugano, M. Inubushi, K. Yoshimura, S. Sunada, K. Kanno, A. Uchida, Compact reservoir computing with a photonic integrated circuit, *Opt. Express* 26 (2018) 29424.
- [55] J. Dong, M. Rafayelyan, F. Krzakala, S. Gigan, Optical reservoir computing using multiple light scattering for chaotic systems prediction, *IEEE J. Sel. Top. Quantum Electron.* 26 (2019) 7701012.
- [56] D. Brunner, M.C. Soriano, G. Van der Sande, Photonic Reservoir Computing: Optical Recurrent Neural Networks, Walter de Gruyter GmbH & Co KG, 2019.
- [57] P. Antonik, M. Haelterman, S. Massar, Brain-inspired photonic signal processor for generating periodic patterns and emulating chaotic systems, *Phys. Rev. A* 7 (2017) 054014.
- [58] I. Estébanez, I. Fischer, M.C. Soriano, Constructive role of noise for high-quality replication of chaotic attractor dynamics using a hardware-based reservoir computer, *Phys. Rev. A* 12 (2019) 034058.
- [59] N. Cox, J. Murray, J. Hart, B. Redding, Photonic next-generation reservoir computer based on distributed feedback in optical fiber, 2024.
- [60] D.J. Gauthier, E. Bolt, A. Griffith, W.A. Barbosa, Next generation reservoir computing, *Nat. Commun.* 12 (2021) 5564.
- [61] G.P. Agrawal, Fiber-Optic Communication Systems, John Wiley & Sons, 2012.
- [62] P.J. Winzer, D.T. Neilson, A.R. Chraplyvy, Fiber-optic transmission and networking: the previous 20 and the next 20 years, *Opt. Express* 26 (2018) 24190.
- [63] T. Li, Advances in optical fiber communications: An historical perspective, *IEEE J. Sel. Areas Commun.* 1 (1983) 356.
- [64] J. Goodman, A Short History of the Field of Optical Computing, CRC Press, 2020, pp. 7–21.
- [65] L. Grüner-Nielsen, M. Wandel, P. Kristensen, C. Jorgensen, L.V. Jorgensen, B. Edvold, B. Pálsdóttir, D. Jakobsen, Dispersion-compensating fibers, *J. Lightwave Technol.* 23 (2005) 3566.
- [66] N.M. Litchinitser, M. Sumetsky, P.S. Westbrook, Fiber-based tunable dispersion compensation, in: Fiber Based Dispersion Compensation, Springer New York, New York, NY, 2007, pp. 379–423.
- [67] P.J. Freire, A. Napoli, B. Spinnler, N. Costa, S.K. Turitsyn, J.E. Prilepsky, Neural networks-based equalizers for coherent optical transmission: Caveats and pitfalls, *IEEE J. Sel. Top. Quantum Electron.* 28 (2022) 7600223.
- [68] D. Zibar, M. Piels, R. Jones, C.G. Schäffer, Machine learning techniques in optical communication, *J. Lightwave Technol.* 34 (2016) 1442.
- [69] P. Freire, E. Manuylovich, J.E. Prilepsky, S.K. Turitsyn, Artificial neural networks for photonic applications—from algorithms to implementation: tutorial, *Adv. Opt. Photon.* 15 (2023) 739.
- [70] A. Argyris, J. Cantero, M. Galletero, E. Pereda, C.R. Mirasso, I. Fischer, M.C. Soriano, Comparison of photonic reservoir computing systems for fiber transmission equalization, *IEEE J. Sel. Top. Quantum Electron.* 26 (2019) 5100309.
- [71] A. Katumba, X. Yin, J. Dambre, P. Bienstman, A neuromorphic silicon photonics nonlinear equalizer for optical communications with intensity modulation and direct detection, *J. Lightwave Technol.* 37 (2019) 2232.
- [72] F. Da Ros, S.M. Ranzini, H. Bülow, D. Zibar, Reservoir-computing based equalization with optical pre-processing for short-reach optical transmission, *IEEE J. Sel. Top. Quantum Electron.* 26 (2020) 7701912.
- [73] B. Wang, T.F. de Lima, B.J. Shastri, P.R. Prucnal, C. Huang, Multi-wavelength photonic neuromorphic computing for intra and inter-channel distortion compensations in wdm optical communication systems, *IEEE J. Sel. Top. Quantum Electron.* 29 (2023) 1.
- [74] D.M. Marom, Y. Miyamoto, D.T. Neilson, I. Tomkos, Optical switching in future fiber-optic networks utilizing spectral and spatial degrees of freedom, *Proc. IEEE* 110 (2022) 1835.
- [75] M. Nance Hall, K.-T. Foerster, S. Schmid, R. Durairajan, A survey of reconfigurable optical networks, *Opt. Switch. Netw.* 41 (2021) 100621.
- [76] K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, P. Bienstman, Experimental demonstration of reservoir computing on a silicon photonics chip, *Nat. Commun.* 5 (2014) 3541.

- [77] K.-i. Kitayama, M. Notomi, M. Naruse, K. Inoue, S. Kawakami, A. Uchida, Novel frontier of photonics for data processing—photonic accelerator, *Appl Photonics* 4 (2019).
- [78] J. Spall, X. Guo, T.D. Barrett, A. Lvovsky, Fully reconfigurable coherent optical vector–matrix multiplication, *Opt. Lett.* 45 (2020) 5752.
- [79] H. Zhou, J. Dong, J. Cheng, W. Dong, C. Huang, Y. Shen, Q. Zhang, M. Gu, C. Qian, H. Chen, et al., Photonic matrix multiplication lights up photonic accelerator and beyond, *Light: Science Appl.* 11 (2022) 30.
- [80] Y. Bai, X. Xu, M. Tan, Y. Sun, Y. Li, J. Wu, R. Morandotti, A. Mitchell, K. Xu, D.J. Moss, Photonic multiplexing techniques for neuromorphic computing, *Nanophotonics* 12 (2023) 795.
- [81] X. Porte, A. Skalli, N. Haghghi, S. Reitzenstein, J.A. Lott, D. Brunner, A complete, parallel and autonomous photonic neural network in a semiconductor multimode laser, *J. Phys.: Photonics* 3 (2021) 024017.
- [82] J. Bueno, S. Maktoobi, L. Froehly, I. Fischer, M. Jacquot, L. Larger, D. Brunner, Reinforcement learning in a large-scale photonic recurrent neural network, *Optica* 5 (2018) 756.
- [83] L. Andreoli, X. Porte, S. Chrétien, M. Jacquot, L. Larger, D. Brunner, Boolean learning under noise-perturbations in hardware neural networks, *Nanophotonics* 9 (2020) 4139.
- [84] A. Skalli, J. Robertson, D. Owen-News, M. Hejda, X. Porte, S. Reitzenstein, A. Hurtado, D. Brunner, Photonic neuromorphic computing using vertical cavity semiconductor lasers, *Opt. Mater. Express* 12 (2022) 2395.
- [85] L. Appeltant, M.C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C.R. Mirasso, I. Fischer, Information processing using a single dynamical node as complex system, *Nat. Commun.* 2 (2011) 468.
- [86] Y. Paquot, F. Dupont, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, S. Massar, Optoelectronic reservoir computing, *Sci. Rep.* 2 (2012) 287.
- [87] L. Larger, M.C. Soriano, D. Brunner, L. Appeltant, J.M. Gutiérrez, L. Pesquera, C.R. Mirasso, I. Fischer, Photonic information processing beyond Turing: an optoelectronic implementation of reservoir computing, *Opt. Express* 20 (2012) 3241.
- [88] M. Hermans, P. Antonik, M. Haelterman, S. Massar, Embodiment of learning in electro-optical signal processors, *Phys. Rev. Lett.* 117 (2016) 128301.
- [89] L. Larger, A. Baylón-Fuentes, R. Martinenghi, V.S. Udaltsov, Y.K. Chembo, M. Jacquot, High-speed photonic reservoir computing using a time-delay-based architecture: Million words per second classification, *Phys. Rev. X* 7 (2017) 011015.
- [90] E. Picco, A. Lupo, S. Massar, Deep photonic reservoir computer for speech recognition, 2023, arXiv:2312.06558 [cs.NE].
- [91] E. Picco, S. Massar, Real-time photonic deep reservoir computing for speech recognition, in: 2023 International Joint Conference on Neural Networks, IJCNN, IEEE, 2023, pp. 1–7.
- [92] E. Picco, L. Jaurigue, K. Lidge, S. Massar, Efficient optimisation of physical reservoir computers using only a delayed input, 2024, arXiv:2401.14371.
- [93] D. Brunner, M.C. Soriano, C.R. Mirasso, I. Fischer, Parallel photonic information processing at gigabyte per second data rates using transient states, *Nature Commun.* 4 (2013) 1364.
- [94] M.C. Soriano, J. García-Ojalvo, C.R. Mirasso, I. Fischer, Complex photonics: Dynamics and applications of delay-coupled semiconductor lasers, *Rev. Modern Phys.* 85 (2013) 421.
- [95] A. Argyris, J. Bueno, I. Fischer, Photonic machine learning implementation for signal recovery in optical communications, *Sci. Rep.* 8 (2018) 8487.
- [96] A. Argyris, J. Bueno, I. Fischer, PAM-4 transmission at 1550 nm using photonic reservoir computing post-processing, *IEEE Access* 7 (2019) 37017.
- [97] I. Estébanez, S. Li, J. Schwind, I. Fischer, S. Pachnicke, A. Argyris, 56 Gbaud pam-4 100 km transmission system with photonic processing schemes, *J. Lightwave Technol.* 40 (2022) 55.
- [98] A. Argyris, Photonic neuromorphic technologies in optical communications, *Nanophotonics* 11 (2022) 897.
- [99] I. Bauwens, K. Harkhoe, P. Bienstman, G. Verschaffelt, G.V. der Sande, Influence of the input signal phase modulation on the performance of optical delay-based reservoir computing using semiconductor lasers, *Opt. Express* 30 (2022) 13434.
- [100] K. Kanno, A.A. Haya, A. Uchida, Reservoir computing based on an external-cavity semiconductor laser with optical feedback modulation, *Opt. Express* 30 (2022) 34218.
- [101] L. Talandier, I. Fischer, A. Argyris, Hybrid-coherent reservoir computing for 200 km pam-4 transmission links, in: Optical Interconnects XXIII, Vol. 12427, SPIE, 2023, pp. 9–15.
- [102] I. Estébanez, L. Talandier, I. Fischer, A. Argyris, Phase vs. intensity encoding in an experimental time delay reservoir computing scheme, in: European Quantum Electronics Conference, Optica Publishing Group, 2023, p. jsiii\_1.4.
- [103] J. Bueno, D. Brunner, M.C. Soriano, I. Fischer, Conditions for reservoir computing performance using semiconductor lasers with delayed optical feedback, *Opt. Express* 25 (2017) 2401.
- [104] I. Estébanez, J. Schwind, I. Fischer, A. Argyris, Accelerating photonic computing by bandwidth enhancement of a time-delay reservoir, *Nanophotonics* 9 (2020) 4163.
- [105] I. Estébanez, A. Argyris, I. Fischer, Experimental demonstration of bandwidth enhancement in photonic time delay reservoir computing, *Opt. Lett.* 48 (2023) 2449.
- [106] M. Sorokina, Multi-channel optical neuromorphic processor for frequency-multiplexed signals, *J. Phys.: Photonics* 3 (2020) 014002.
- [107] X. Xu, M. Tan, B. Corcoran, J. Wu, A. Boes, T.G. Nguyen, S.T. Chu, B.E. Little, D.G. Hicks, R. Morandotti, A. Mitchell, D.J. Moss, 11 Tops photonic convolutional accelerator for optical neural networks, *Nature* 589 (2021) 44–51.
- [108] B. Dong, S. Aggarwal, W. Zhou, U.E. Ali, N. Farmakidis, J.S. Lee, Y. He, X. Li, D.-L. Kwong, C. Wright, et al., Higher-dimensional processing using a photonic tensor core with continuous-time data, *Nat. Photonics* 17 (2023) 1080–1088.
- [109] L. Butschek, A. Akrot, E. Dimitriadou, A. Lupo, M. Haelterman, S. Massar, Photonic reservoir computer based on frequency multiplexing, *Opt. Lett.* 47 (2022) 782.
- [110] M. Bloch, S.W. McLaughlin, J.-M. Merolla, F. Patois, Frequency-coded quantum key distribution, *Opt. Lett.* 32 (2007) 301.
- [111] L. Ollislager, E. Woodhead, K. Phan Huy, J.-M. Merolla, P. Emplit, S. Massar, Creating and manipulating entangled optical qubits in the frequency domain, *Phys. Rev. A* 89 (2014) 052323.
- [112] A. Lupo, L. Butschek, S. Massar, Photonic extreme learning machine based on frequency multiplexing, *Opt. Express* 29 (2021) 28257.
- [113] A. Lupo, E. Picco, M. Zajnulina, S. Massar, Deep photonic reservoir computer based on frequency multiplexing with fully analog connection between layers, *Optica* 10 (2023) 1478.
- [114] A. Lupo, S. Massar, Parallel extreme learning machines based on frequency multiplexing, *Appl. Sci.* 12 (2021) 214.
- [115] C. Gallicchio, A. Micheli, L. Pedrelli, Deep reservoir computing: A critical experimental analysis, *Neurocomputing* 268 (2017) 87–99.
- [116] L. Thylén, L. Wosinski, Integrated photonics in the 21st century, *Photon. Res.* 2 (2014) 75.
- [117] A. Tsakyridis, M. Moralis-Pegios, G. Giamougiannis, M. Kirtas, N. Passalis, A. Tefas, N. Pleros, Photonic neural networks and optics-informed deep learning fundamentals, *APL Photonics* 9 (2024) 011102.
- [118] D. Pérez-López, A. Gutierrez, D. Sánchez, A. López-Hernández, M. Gutierrez, E. Sánchez-Gomáriz, J. Fernández, A. Cruz, A. Quirós, Z. Xie, J. Benitez, N. Bekesi, A. Santomé, D. Pérez-Galacho, P. DasMahapatra, A. Macho, J. Capmany, General-purpose programmable photonic processor for advanced radiofrequency applications, *Nature Commun.* 15 (2024) 1563.
- [119] S. Sackesyn, C. Ma, J. Dambre, P. Bienstman, An enhanced architecture for silicon photonic reservoir computing, in: Cognitive Computing 2018 -Merging Concepts with Hardware, 2018.
- [120] S. Sackesyn, C. Ma, J. Dambre, P. Bienstman, Experimental realization of integrated photonic reservoir computing for nonlinear fiber distortion compensation, *Opt. Express* 29 (2021) 30991.



- [121] E. Gooskens, S. Sackesyn, J. Dambre, P. Bienstman, Experimental results on nonlinear distortion compensation using photonic reservoir computing with a single set of weights for different wavelengths, *Sci. Rep.* 13 (2023) 21399.
- [122] C. Ma, J.V. Kerrebrouck, H. Deng, S. Sackesyn, E. Gooskens, B. Bai, J. Dambre, P. Bienstman, Integrated photonic reservoir computing with an all-optical readout, *Opt. Express* 31 (2023) 34843.
- [123] S. Masaad, E. Gooskens, S. Sackesyn, J. Dambre, P. Bienstman, Photonic reservoir computing for nonlinear equalization of 64-qam signals with a kramers-kronig receiver, *Nanophotonics* 12 (2023) 925.
- [124] I. Boikov, D. Brunner, A. De Rossi, Evanescent coupling of nonlinear integrated cavities for all-optical reservoir computing, *New J. Phys.* 25 (2023) 093056.
- [125] W.D. Sacher, J.C. Mikkelsen, Y. Huang, J.C. Mak, Z. Yong, X. Luo, Y. Li, P. Dumais, J. Jiang, D. Goodwill, et al., Monolithically integrated multilayer silicon nitride-on-silicon waveguide platforms for 3-d photonic circuits and devices, *Proc. IEEE* 106 (2018) 2232.
- [126] J. Moughames, X. Porte, M. Thiel, G. Ulliac, L. Larger, M. Jacquot, M. Kadic, D. Brunner, Three-dimensional waveguide interconnects for scalable integration of photonic neural networks, *Optica* 7 (2020) 640.
- [127] J.W. Goodman, *Introduction To Fourier Optics*, Roberts and Company publishers, 2005.
- [128] C.C. Aggarwal, et al., *Neural Networks and Deep Learning*, Vol. 10, Springer, 2018, p. 3.
- [129] E.A. Vlieg, L. Talandier, R. Dangel, F. Horst, B.J. Offrein, An integrated photorefractive analog matrix-vector multiplier for machine learning, *Appl. Sci.* (2022).
- [130] B. Offrein, J. Geler-Kremer, J. Weiss, R. Dangel, P. Stark, A. Sharma, S. Abel, F. Horst, Prospects for photonic implementations of neuromorphic devices and systems, in: *IEDM*, 2020.
- [131] D. Psaltis, D. Brady, K. Wagner, Adaptive optical networks using photorefractive crystals, *Appl. Opt.* 27 (1988) 1752.
- [132] D. Psaltis, G. Burr, Holographic data storage, *Computer* 31 (1998) 52.
- [133] N.V. Kukhtarev, V.B. Markov, S.G. Odulov, M.S. Soskin, V.L. Vinetskii, Holographic storage in electrooptic crystals. i. steady state, *Ferroelectrics* 22 (1978) 949.
- [134] E.A. Vlieg, F. Horst, R. Dangel, B.J. Offrein, The demonstration of photorefractive synaptic connections for an integrated photonic crossbar array, in: *2023 Conference on Lasers and Electro-Optics Europe & European Quantum Electronics Conference, CLEO/Europe-EQEC*, 2023.
- [135] T. Jonuzi, A. Lupo, L. Talandier, M. Goldmann, I. Fischer, A. Argyris, S. Massar, M.C. Soriano, D. Doménech, Reconfigurable time-space photonic integrated convolutional accelerator, in: *AI and Optical Data Sciences V*, SPIE, 2023.
- [136] S. Abreu, M. Gouda, A. Lugnan, P. Bienstman, Flow cytometry with event-based vision and spiking neuromorphic hardware, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4138–4146.
- [137] M. Goldmann, I. Fischer, C.R. Mirasso, M.C. Soriano, Exploiting oscillatory dynamics of delay systems for reservoir computing, *Chaos* 33 (2023).
- [138] M. Hermans, M.C. Soriano, J. Dambre, P. Bienstman, I. Fischer, Photonic delay systems as machine learning implementations, *J. Mach. Learn. Res.* 16 (2015) 2081.
- [139] L.G. Wright, T. Onodera, M.M. Stein, T. Wang, D.T. Schachter, Z. Hu, P.L. McMahon, Deep physical neural networks trained with backpropagation, *Nature* 601 (2022) 549.
- [140] M. Nakajima, K. Inoue, K. Tanaka, Y. Kuniyoshi, T. Hashimoto, K. Nakajima, Physical deep learning with biologically inspired training method: gradient-free approach for physical hardware, *Nature Commun.* 13 (2022) 7847.
- [141] B. Scellier, Y. Bengio, Equilibrium propagation: Bridging the gap between energy-based models and backpropagation, *Front. Comput. Neurosci.* 11 (2017) 24.
- [142] B. Scellier, S. Mishra, Y. Bengio, Y. Ollivier, Agnostic physics-driven deep learning, 2022, arXiv preprint [arXiv:2205.15021](https://arxiv.org/abs/2205.15021).
- [143] J. Laydevant, D. Markovic, J. Grollier, Training an Ising machine with equilibrium propagation, 2023, [arXiv:2305.18321](https://arxiv.org/abs/2305.18321), [cs.NE].
- [144] J. Kendall, R. Pantone, K. Manickavasagam, Y. Bengio, B. Scellier, Training end-to-end analog neural networks with equilibrium propagation, 2020, [arXiv:2006.01981](https://arxiv.org/abs/2006.01981), [cs.NE].
- [145] N. Hansen, A. Auger, Cma-es: evolution strategies and covariance matrix adaptation, in: *Proceedings of the 13th Annual Conference Companion on Genetic and Evolutionary Computation*, 2011, pp. 991–1010.
- [146] F. Sehnke, C. Osendorfer, T. Rückstieß, A. Graves, J. Peters, J. Schmidhuber, Parameter-exploring policy gradients, *Neural Netw.* 23 (2010) 551.
- [147] H. Jaeger, D. Doorkackers, C. Lawrence, G. Indiveri, Dimensions of timescales in neuromorphic computing systems, 2021, arXiv preprint [arXiv:2102.10648](https://arxiv.org/abs/2102.10648).
- [148] C.H. Papadimitriou, Computational complexity, in: *Encyclopedia of Computer Science*, 2003, pp. 260–265.
- [149] S. Utsunomiya, K. Takata, Y. Yamamoto, Mapping of Ising models onto injection-locked laser systems, *Opt. Express* 19 (2011) 18091.
- [150] T. Honjo, T. Sonobe, K. Inaba, T. Inagaki, T. Ikuta, Y. Yamada, T. Kazama, K. Enbutsu, T. Umeki, R. Kasahara, et al., 100,000-Spin coherent Ising machine, *Sci. Adv.* 7 (2021) eabh0952.
- [151] N. Mohseni, P.L. McMahon, T. Byrnes, Ising machines as hardware solvers of combinatorial optimization problems, *Nat. Rev. Phys.* 4 (2022) 363.
- [152] H. Goto, K. Tatsumura, A.R. Dixon, Combinatorial optimization by simulating adiabatic bifurcations in nonlinear hamiltonian systems, *Sci. Adv.* 5 (2019) eaav2372.
- [153] H. Goto, K. Endo, M. Suzuki, Y. Sakai, T. Kanao, Y. Hamakawa, R. Hidaka, M. Yamasaki, K. Tatsumura, High-performance combinatorial optimization based on classical mechanics, *Sci. Adv.* 7 (2021) eabe7953.