Ultrafast Computing with Nonlinear Photonics

Thesis by

Gordon Han Ying Li

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2025
Defended January 30, 2025

Gordon Han Ying Li
ORCID: 0000-0001-8184-4915

ACKNOWLEDGEMENTS

The biggest acknowledgment goes to my PhD adviser Prof. Alireza Marandi. This thesis would surely not exist without his thoughtful support and guidance. I thank him for affording me the complete academic freedom to pursue my desired research direction in optical computing, and for providing me with access to the tools needed to succeed in this endeavor. I have found his lab to be a highly stimulating and pleasant environment to conduct research.

I acknowledge the contributions made by many members (both past and present) of the Marandi group towards my research goals. Special thanks go to my roommate, office-mate, and most prolific collaborator James Williams for his help with everything and anything to do with electronics. I thank Christian Leefmans for being my Applied Physics mentor, teaching me all the experimental techniques to do with optical fibers, and never complaining when I "*borrowed*" his equipment. I thank Robert Gray, whom possesses the greatest experimental optics skills that I have ever encountered, for his patience in assisting me with many (probably easy) measurements. I am grateful towards Ryoto Sekine for fabricating the photonic chips used in my experiments, which laid the groundwork for my entire thesis. I thank Midya Parto for being my main co-conspirator in the pursuit of optical computing and for entertaining even my craziest research ideas. Other fellow group members including (but not limited to) Nicolas Englebert, Rajveer Nehra, Arkadev Roy, Mingchen Liu, Luis Ledezma, Saman Jahani, Luis Costa, and Qiushi Guo have each also positively impacted my time in the Marandi lab in their own unique ways. Although unrelated to my research efforts, I very much enjoyed surveying the local dim sum landscape with Selina Zhou and Maximilian Shen.

I thank Prof. Andrei Faraon, Prof. Jehoshua (Shuki) Bruck, and Prof. Kerry Vahala for serving on my PhD thesis committee. Their tutelage and guidance has been inspiring. I am forever indebted to Prof. Vahala and his group members, whom have played an outsized role in my research. I thank his students Qing-Xin Ji, Maodong Gao, Jinhao Ge, and Yan Yu for their selfless willingness to help me with my experiments, despite no obvious upside to them. It would have been severely difficult to progress further in my research without their invaluable collaboration. I also thank past Vahala group members Lue Wu and Seung Hoon Lee for their pivotal roles in my career development.

Caltech has proven to be a truly magical place to undertake my graduate studies and I feel proud to be considered part of such a storied and high-caliber institution. The uniquely small size, intense research fervor, and unyielding pursuit of excellence has indelibly shaped my own perspectives on science and engineering. I have had the opportunity of interacting with many brilliant peers and teachers, which has propelled me to greater heights. My only gripe is that *the swimming pool is too small*. I thank Jennifer Blankenship for going above and beyond in her role as the Option Manager for Applied Physics. I had the privilege of reviving the Optica (formerly OSA) Student Chapter at Caltech and enjoyed growing it with the help of other Marandi group members. I thank Optica for their support and leave the student chapter as my legacy to future optics students at Caltech.

I acknowledge support from the Quad Fellowship. The generous financial support and network of Quad Fellows has greatly enriched my graduate school experience. I am deeply honored to be part of the inaugural cohort for this visionary initiative by the governments of Australia, Japan, India, and the United States.

Throughout my physics studies, I have been blessed with many good mentors and teachers. The one who stands out as having the largest impact on me as a physicist is my undergraduate research adviser Prof. C. Martijn de Sterke. I thank him for personally imbuing me with my entire knowledge of nonlinear optics over the course of more than four years under his tutelage. This knowledge has served me well and his style of doing physics has heavily influenced my own approach to physics. I am grateful for his support and encouragement in attending graduate school at Caltech.

Finally, my most heartfelt thanks goes to my family for their unwavering emotional support. I am especially grateful to my parents for believing in me every step of the way along my educational journey. I find solace in that the one silver lining from the COVID pandemic was that I was fortunate to spend my first year as a Caltech graduate student doing remote learning from the comfort of my parents' home in Sydney, Australia. This thesis is dedicated to my parents.

ABSTRACT

Computers have revolutionized almost every facet of modern society, and as we approach the physical limits of digital electronics, it becomes imperative to investigate alternative computing hardware paradigms to enable the next generation of faster and more energy-efficient computers. This thesis embarks on building the foundation for a new kind of computer, based on ultrafast nonlinear photonics, aiming to overcome some of the limitations plaguing current computers. In particular, we primarily focus on the clock rate, which has stagnated at $\sim$ 5 GHz for conventional microprocessors over the past two decades.

We begin by identifying single nonlinear devices in lithium niobate nanophotonics that can act as essential building blocks for computers, showing a variety of nonlinear functions with operational speeds $>$ 13 THz for artificial intelligence computing workloads. Then, we progress to small-scale photonic computing circuits combining both strong nonlinearity and memory feedback in a physical reservoir computer for temporal information processing with $\sim$ 10 GHz clock rates. Additionally, we explore unconventional computer architectures such as Cellular Automata, which reveals key system-level considerations that maximize the benefits of ultrafast nonlinear photonics in large-scale computers. This culminates in the demonstration of truly end-to-end and all-optical computing with $>$ 100 GHz clock rates, which represents over an order-of-magnitude advancement compared to existing electronic computers. Finally, we prove mathematically how coupled nonlinear optical resonators are Turing-complete computers.

Overall, this work builds on the recent advances in nonlinear photonics and highlights a path for a new class of ultrafast photonic computers that can surpass the clock rate and latency limits of electronic computers, hence enabling nascent applications requiring real-time control or information processing at picosecond timescales.

PUBLISHED CONTENT AND CONTRIBUTIONS

[1] **Gordon H.Y. Li***, Ryoto Sekine*, Rajveer Nehra*, Robert M. Gray*, Luis Ledezma, Qiushi Guo, and Alireza Marandi. All-optical ultrafast ReLU function for energy-efficient nanophotonic deep learning. *Nanophotonics*, 12(5): 847–855, 2023. doi:10.1515/nanoph-2022-0137.
G.H.Y.L. conceived the project, developed the theory, executed the numerical simulations, designed the experiments, participated in the experimental measurements, analyzed the data, and wrote the manuscript.
* denotes equal contributions.

[2] **Gordon H.Y. Li**, Christian R. Leefmans, James Williams, and Alireza Marandi. Photonic elementary cellular automata for simulation of complex phenomena. *Light: Science & Applications*, 12(1):132, 2023. doi:10.1038/s41377-023-01180-9.
G.H.Y.L. conceived the project, developed the theory, executed numerical simulations, designed the experiments, participated in building the experimental setup, performed the experimental measurements, analyzed the data, and wrote the manuscript.

[3] **Gordon H.Y. Li**, Christian R. Leefmans, James Williams, Robert M. Gray, Midya Parto, and Alireza Marandi. Deep learning with photonic neural cellular automata. *Light: Science & Applications*, 13(1):283, 2024. doi:10.1038/s41377-024-01651-7.
G.H.Y.L. conceived the project, developed the theory, executed numerical simulations, designed the experiments, built the experimental setup, performed the experimental measurements, analyzed the data, and wrote the manuscript.

[4] **Gordon H.Y. Li***, Midya Parto*, Jinhao Ge*, Qing-Xin Ji, Maodong Gao, Yan Yu, James Williams, Robert M. Gray, Christian R. Leefmans, Nicolas Englebert, Kerry J. Vahala, and Alireza Marandi. All-optical computing with beyond 100-GHz clock rates. *arXiv preprint arXiv:2501.05756*, 2025. doi:10.48550/arXiv.2501.05756.
G.H.Y.L. conceived the project, developed the theory, executed numerical simulations, designed the experiments, built the experimental setup, performed the experimental measurements, analyzed the data, and wrote the manuscript.
* denotes equal contributions.

[5] **Gordon H.Y. Li** and Alireza Marandi. Turing-completeness and undecidability in coupled nonlinear optical resonators. *arXiv preprint arXiv:2501.06966*, 2025. doi:10.48550/arXiv.2501.06966.
G.H.Y.L. conceived the project, developed the theory, and wrote the manuscript.

[6] Midya Parto*, **Gordon H.Y. Li***, Ryoto Sekine, Robert M. Gray, Luis L. Ledezma, James Williams, Arkadev Roy, and Alireza Marandi. Ultrafast neu-

romorphic computing with nanophotonic optical parametric oscillators. *arXiv preprint arXiv:2501.16604*, 2025. doi:10.48550/arXiv.2501.16604.

G.H.Y.L. conceived the project, participated in developing the theory, executed initial numerical simulations, participated in the design of the experiments, and participated in the writing of the manuscript.

* denotes equal contributions.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

> *"It would appear that we have reached the limits of what it is possible to achieve with computer technology, although one should be careful with such statements, as they tend to sound pretty silly in five years."*
>
> — John von Neumann (ca. 1949)

1.1 Overview

The invention of general-purpose programmable computers during the 20th century propelled humanity into the current Information Age. Indeed, computers have revolutionized almost every facet of modern society, including the ways in which we work, communicate, travel, educate, and entertain, among countless other impacts on our everyday lives. This is evidenced by the world's insatiable demand for information technologies, which has driven a never-ending quest for faster computer hardware — beginning with early mechanical computers such as the Z1 [1], then progressing to electronic computers constructed from vacuum tubes such as ENIAC [2], and finally maturing into today's central processing units (CPUs) [3] consisting of billions of integrated silicon transistors. Each dramatic increase in computing speed throughout history has yielded innumerable new applications and innovations which were previously computationally infeasible. Now, in the 21st century, we continue to witness the widespread adoption of personal computers and smart devices such as smartphones, proliferation of the Internet and cloud computing, and advent of artificial intelligence (AI) [4]. As we approach the fundamental physical limits of digital electronics, it becomes imperative to investigate alternative paradigms to enable the next generation of faster and more energy-efficient computers. This will require a radical departure from the status quo to ensure that computers continue to impact positive change for society long into the future.

This thesis embarks on an ambitious journey to develop an entirely new kind of computer from the ground up based on ultrafast nonlinear optics and photonics, which can overcome some of the limitations plaguing current electronic computers. The results presented in this thesis span the entire gamut of computer science and engineering; starting from applying the physics of nonlinear optics to enable single devices that act as building blocks for photonic computers, then assembling

these photonic devices into small-scale computing circuits, followed by building unconventional system-level computer architectures that maximize the benefits of nonlinear photonics, enabling nascent ultrafast computing applications, and finally proving fundamental limits associated with large-scale nonlinear photonic computers. The remainder of this thesis is organized as outlined below.

In **Chapter 1**, some key computational trends over recent decades are explained, which provides the impetus for the remainder of this thesis. In particular, this thesis is primarily concerned with the computational clock rate, which is an important hardware metric that ultimately determines the minimum time between sequential operations or instructions in a computer. The key takeaway is that despite exponential advances in electronic computer performance owing to Moore's Law and increasingly parallel system architectures, computer clock rates have remained stagnant at $\sim$ 5 GHz for almost two decades, which highlights a unique opportunity for a new class of ultrafast photonic computers. Additionally, the applications of time-multiplexing and parametric nonlinear photonics to computing are reviewed since they are essential concepts that will feature heavily in the following chapters.

Chapter 2 describes how an optical parametric amplifier can act as the elusive and previously absent nonlinear optical element for photonic computers, akin to how the transistor serves as the fundamental nonlinear element in digital electronic computers. Although optics and photonics excels at linear operations, most special-purpose photonic processors lack nonlinearity, which is an essential ingredient for general computation. Recent advances in thin-film lithium niobate nanophotonics, ultrafast $\chi^{(2)}$ nonlinear optics, and ultrashort laser pulses are exploited to demonstrate the example application of nonlinear activation functions for deep learning. A record-breaking energy-time product of $\sim 1.2 \times 10^{-27}$ J $\cdot$ s per activation with > 13.3 THz operational clock rate is achieved, which is a staggering three orders-of-magnitude improvement over electronic counterparts. This enables the exciting possibility of truly end-to-end and all-optical neural networks for emerging AI applications.

Chapter 3 continues the theme of ultrafast $\chi^{(2)}$ nonlinear optics in thin-film lithium niobate nanophotonics by moving from single photonic devices to small-scale computing circuits. An optical parametric oscillator combines the strong nonlinearity of optical parametric amplifiers introduced in the previous chapter with optical cavity feedback for memory, which is another essential ingredient for general computation. It is shown how the nanophotonic optical parametric oscillator acts as an on-chip and resource-efficient reservoir for time-multiplexed reservoir computers, which are

a special type of random recurrent neural network. Important applications to time-series forecasting and optical signal processing are demonstrated with $\sim$ 10 GHz clock rates and sub-nanosecond latencies, which exceeds the speed of existing digital electronic computers. This represents a significant milestone towards large-scale integrated photonic neural networks with both ultrafast nonlinearity and memory capabilities.

Chapter 4 initiates a foray into unconventional computer architectures that can maximize the benefits of photonic computers and minimize potential drawbacks by taking into account fundamental system-level differences between digital electronics and analog photonics. A time-multiplexed optical fiber network, as opposed to small-scale photonic integrated circuits in previous chapters, is used for rapid prototyping of large-scale computing systems. Cellular automata, which is a class of computational models based on simple rules and local interactions, are implemented directly in the special-purpose photonic computer for simulating complex phenomena such as fractals, chaos, and solitons. Howevever, this approach is only a stepping stone towards the ultimate goal of an ultrafast all-optical computer since it still utilizes a hybrid optoelectronic nonlinearity and memory. Nonetheless, it demonstrates the crucial first step of how photonic computers utilizing ultrashort laser pulses can achieve universal computation.

Chapter 5 improves upon the results of the previous chapter by replacing the optoelectronic nonlinearity with an ultrafast $\chi^{(2)}$ optical nonlinearity. The cellular automata concept is combined with modern deep learning techniques to create a photonic implementation of neural cellular automata. This novel approach alleviates some difficulties associated with previous photonic neural networks by delivering robust, reliable, and parameter-efficient information processing. It performs image classification tasks, achieving a total image inference time as low as $\sim 1.3\ \mu$s, which is over an order-of-magnitude faster than current state-of-the-art graphics processing units (GPUs). These results further highlight the need for unconventional architectures and system-level considerations in next-generation photonic computers.

Chapter 6 showcases the first truly end-to-end and all-optical computer with ultrafast clock rates $>$ 100 GHz. This represents the culmination of all the experimental work presented in this thesis and combines lessons learned from all the previous chapters. It utilizes a time-multiplexed network of ultrashort laser pulses with linear optical operations, $\chi^{(2)}$ nonlinear optical activations functions, and memory feedback implemented directly in the optical domain. The end result is an all-

optical recurrent neural network with no electronic bottleneck, operating at clock rates over an order-of-magnitude faster than any existing digital electronic computer. This new ultrafast computer based on nonlinear photonics can therefore serve as a key enabling technology for a myriad of nascent applications requiring information processing and control at picosecond timescales, which was previously impossible for electronic computers. It is the first in a new class of large-scale, all-optical and ultrafast computers that will irrevocably change the ways in which we compute.

Finally, **Chapter 7** explores the theoretical limits of nonlinear photonic computing. Techniques from computability theory are applied to mathematically prove that networks of coupled nonlinear optical resonators are Turing-complete, which endows them with much greater computational power than previously realized. Moreover, the minimum threshold of photonic hardware complexity needed for Turing-completeness is surprisingly low, requiring as few as $N = 12$ optical pulses. This poses interesting problems of undecidability in the study of coupled nonlinear optical resonators and ultrafast optical science more broadly. The valuable insights developed can serve as a theoretical foundation for future ultrafast computers based on nonlinear photonics.

1.2 Recent Computational Trends

The most important and reliable trend governing the semiconductor industry over the past half-century is Moore's Law. In 1965, Gordon Moore made the empirical observation that the number of transistors integrated on a computer microprocessor chip doubled every year [5]. This doubling rate was later revised to once every roughly two years, and has held remarkably steadfast since 1975. Therefore, simply by waiting a few years, one can expect exponential increases in computer performance owing to increasingly higher transistor density counts. Moore's Law has become a self-fulfilling prophecy that heavily informs the design of new CPUs.

For example, Table. 1.1 shows the forecasts made by the 2003 International Technology Roadmap for Semiconductors (ITRS) [6] around the turn of the century. It was expected that the physical gate length of transistors would continue to shrink and the number of transistors would continue to increase exponentially with each new generation as per Moore's Law. Simultaneously, a commensurate increase in the CPU clock rate and power dissipation was also expected. For the most part, the predictions made by ITRS have been accurate, with the notable exception of the CPU clock rate.

Year	Gate Length (nm)	Clock Rate (GHz)	Maximum Power (W)	Transistors per chip ($\times 10^6$)
2003	45	2.976	149	153
2004	37	4.171	158	193
2005	32	5.204	167	243
2006	28	6.783	180	307
2007	25	9.285	189	386
2008	22	10.972	200	487
2009	20	12.369	210	614
2010	18	15.079	218	773
2012	14	20.065	240	1227
2013	13	22.980	251	1546
2015	10	33.403	270	2454
2016	9	39.683	288	3092
2018	7	53.207	300	4908

Table 1.1: **2003 ITRS forecasts for chip power dissipation and speed in high-performance logic.** Data adapted from Ref. [6].

This is highlighted in Fig. 1.1, which shows the actual CPU clock rate trends over the past few decades [7]. It can be seen that CPU clock rates increased exponentially and commensurately with Moore's Law up until 2005. However, since 2005, CPU clock rates have only increased incrementally. Indeed, modern CPUs have stagnated at $\sim$ 5 GHz clock rates despite continued increases in transistor counts. There are two main reasons contributing to the stagnating CPU clock rates. First, the breakdown of Dennard scaling at the single-transistor level, which roughly states that the power density scales with transistor size. Dennard scaling implies that the number of transistors and clock rate can be increased whilst keeping the chip power density constant, and hence did not require complete overhauls of transistor design from generation to generation. Unfortunately, the breakdown of Dennard scaling around 2005 forced CPU designers to abandon further increases in CPU clock rate in order to keep heat power dissipation manageable.

Second, the increasing prevalence of the von Neumann bottleneck at the system-level, which arises due to the separation of the CPU and memory that are connected by a shared data bus in the widely used von Neumann computer architecture [3]. It was found that many modern computing applications are bandwidth-limited due to the input/output data rates imposed by the von Neumann bottleneck, which means that they cannot fully utilize higher CPU clock rates even if available. This problem

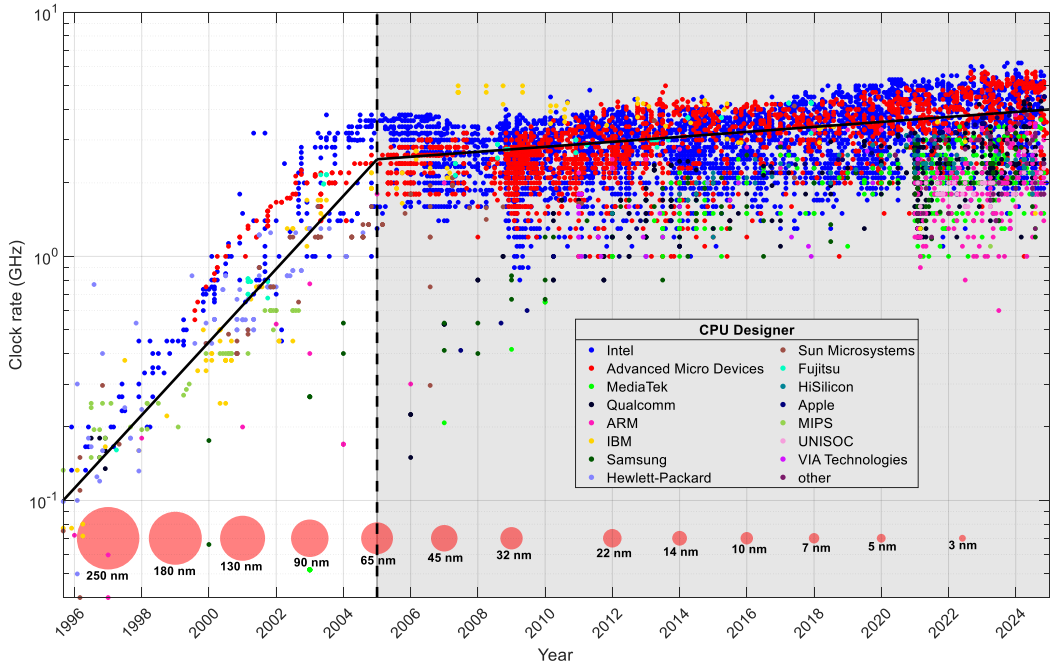

Figure 1.1: **CPU clock rate trends.** Each point indicates the clock rate and testing date for a different type of commercially-available CPU. The color corresponds to the entity that designed the CPU, with some prominent CPU designers listed in the legend. Red circle and text bottom insets represent increasingly advanced semiconductor process nodes and when they were introduced. Adapted from [7]

is especially apparent in recent AI computing workloads, which has prompted the development of application-specific integrated circuits (ASICs) such as graphics processing units (GPUs) [8]. These ASICs can achieve much higher throughput and energy-efficiency for specific operations compared to general-purpose CPUs by harnessing many parallel computing cores and high-bandwidth memory. In fact, GPUs typically operate at far lower clock rates compared to CPUs in favor of increased parallelism. The trend of increasing parallelism instead of clock rate has spilled over into recent CPUs, with the introduction of multi-core and multi-threaded general-purpose processors. This is shown in Fig. 1.2, which shows the proportion of new and different CPU designs each year over the past few decades, separated according to the number of CPU cores. We can see that share of single-core CPUs has steadily fallen since around 2000. Now, in 2025, we see that new CPUs that feature single-core architectures are virtually nonexistent. Most CPUs in recent years have between 2-8 cores. These recent trends suggest that ultrafast nonlinear photonics has a unique opportunity to enable single-threaded computers with exceptionally high clock rates exceeding the limitations of digital electronics.

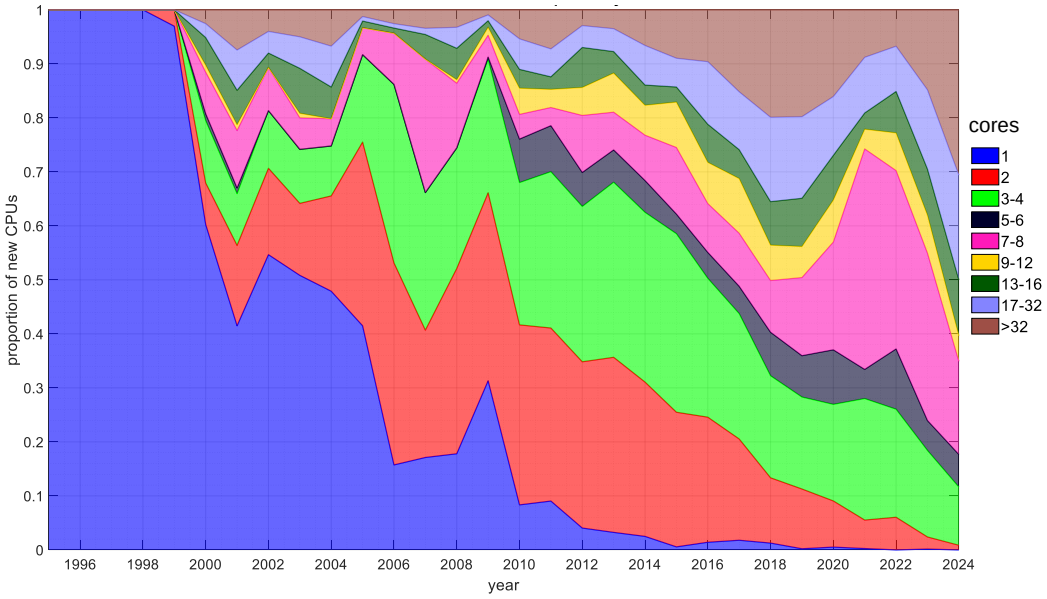

Figure 1.2: **CPU core count trends.** Proportion of new and different CPUs designed over the past 29 years, separated according to the number of cores.

1.3 Review of Time-Multiplexed Photonic Computing

Photonic computers, which process information using light, are receiving renewed attention as a means to overcome the limitations of digital electronic computers [9]. It has long been appreciated how communications can benefit from the enormous bandwidth of optics compared to electronics for transmitting/receiving information. However, the same principles and methods originally developed for communications have recently been extended to the realm of computing to unlock superior information processing capabilities. In particular, time-multiplexing has featured as an indispensable technique for many state-of-the-art optical computing demonstrations [10]. The ability to process information directly in the optical domain by exploiting the time degree-of-freedom is an exciting frontier poised to redefine the landscape of computing. We review some notable recent developments in time-multiplexed optical computers for applications that have traditionally been difficult for digital electronic computers.

Coherent Ising machines

Ising machines are special-purpose physical hardware solvers for combinatorial optimization problems [11]. The Ising Hamiltonian is a well-known model originally

developed in statistical physics for studying spin glasses. It is stated in Eq. 1.1:

$$H = -\sum_{i,j=1}^{N} J_{ij}\sigma_i\sigma_j - \sum_{i=1}^{N} h_i\sigma_i \,, \tag{1.1}$$

where $\sigma_i \in \{-1, 1\}$ for $i = 1, 2, \ldots, N$ are binary spin states, N is the number of spins, $J_{ij} \in \mathbb{R}$ is the coupling strength between spins i and j, and $h_i \in \mathbb{R}$ is an optional on-site interaction field strength for spin i. In general, the Ising optimization problem of finding the ground-state energy, which minimizes the Ising Hamiltonian H, is NP-hard. This falls in the most difficult class of combinatorial optimization problems, and possesses a time complexity that grows exponentially with the problem size N. Alternatively, the Ising decision problem, which is a question with a yes or no answer about the ground state energy of H, is known to be NP-complete. This implies that any other problem in the NP-complete class can be efficiently mapped to the Ising decision problem using only polynomial resources [12]. Thus, solving the Ising problem unlocks the solution to a large number of other important combinatorial optimization problems. Examples include the Travelling Salesman Problem, Boolean Satisfiability Problem, and MaxCut problems. These types of NP-complete problems are often encountered in real-world applications including planning and scheduling, portfolio management, and drug discovery [11]. Therefore, there is significant interest in solving the Ising problem.

It is currently unknown whether there exists an algorithm that can efficiently solve NP-complete problems, which is the subject of the famous open problem $P\ vs.\ NP$. However, Ising machines have been developed as a method for obtaining approximate or heuristic solutions, which have been demonstrated to be at or close to the optimal solution in many practical scenarios [11]. The main idea is to map the solution to the Ising problem onto the fixed-point or steady-state solution of a dynamical system. These dynamical systems often possess natural or more efficient implementations in physical systems compared to digital computers. In this context, analog optical implementations of Ising machines based on networks of degenerate optical parametric oscillators (DOPOs), also known as Coherent Ising Machines (CIMs), have emerged as a promising option [13]. DOPOs are a kind of nonlinear optical resonator utilizing either spontaneous parametric down-conversion or four-wave mixing processes in which the degenerate signal fields have two possible phases, 0 or π, when operated above threshold. The phase 0 and π of the signal fields are used to represent the binary spin states $\sigma_i = \pm 1$. A network of N coupled

DOPOs can be modeled using a simplified classical dynamical system according to Eq. 1.2:

$$\frac{dx_i}{dt} = \alpha x_i - x_i^3 + \sum_{j=1}^{N} J_{ij} x_j \ , \tag{1.2}$$

where $x_i \in \mathbb{R}$ is an analog amplitude representing the i^{th} spin state, t is the time, α is the gain parameter, and J_{ij} represent the coupling weights for the Ising Hamiltonian. The element-wise nonlinearity $\alpha x_i - x_i^3$ represents a parametric gain from a nonlinear optical process, and has the normal form of a supercritical pitchfork bifurcation. An uncoupled DOPO has only one stable trivial solution $x = 0$ when $\alpha < 0$, and bistable solutions $x = \pm\sqrt{\alpha}$ when $\alpha > 0$. The solution to the Ising problem is taken to be the sign of the corresponding analog amplitude, $\sigma_i = \text{sign}(x_i)$ once the dynamical system reaches steady-state.

One way to interpret the network of DOPOs finding the solution to the Ising problem is that each DOPO undergoes spontaneous symmetry breaking when the gain transitions from below threshold ($\alpha < 0$) to above threshold ($\alpha > 0$) such that the lowest-loss eigenmode of the system will be the first to oscillate. Another way to interpret the solution-finding process is by interpreting Eq. 1.2 as a gradient-descent system $dx_i/dt = -\partial V/\partial x_i$ where $V = -1/2\alpha x_i^2 + 1/4 x_i^4 - \sum_{i,j} J_{ij} x_i x_j$ is a Lyapunov function that guarantees local asymptotic convergence. The solution is only approximate since the exact mapping to the Ising problem requires amplitude homogeneity $|x_i| = x$ for all $i = 1, 2, \ldots, N$. However, amplitude homogeneity is almost always violated due to the analog nature of the DOPO amplitudes, and results in sub-optimal solutions by getting stuck in local minima [13].

The first demonstration of time-multiplexed DOPOs was by Marandi *et al.* [14] using a free-space optics platform. A network of $N = 4$ DOPOs was represented using pulses of light produced by a femtosecond mode-locked laser with repetition rate of 250 MHz. Each pulse occupies a time bin specified by the repetition rate of the laser and acts an independent DOPO assuming that the time bin length is much shorter than the roundtrip time of the bow-tie cavity. The coupling interactions between pulses are implemented using a set of optical delay lines. The nonlinear gain mechanism here is spontaneous parametric down conversion from a periodically-poled lithium niobate (PPLN) crystal. The relative phases of signal pulses can be determined using an interference measurement of successive pulses. Therefore, there is only one physical cavity, but a network of $N = 4$ time-multiplexed DOPOs exists within the cavity. This is possible because the $\chi^{(2)}$ optical nonlinearity of the PPLN crystal

is ultrafast and effectively instantaneous compared to the time length of each pulse, which eliminates cross-talk between pulses due to memory effects in the nonlinear crystal. The only coupling between pulses are the linear couplings from the optical delay lines, which specifies the desired Ising problem. A single cavity roundtrip corresponds to a single iteration of the linear couplings and nonlinear gain for all pulses. Takata *et al.* [15] extended the network size to $N = 16$ and 1 GHz repetition rate, but only had a limited connectivity for the Ising couplings.

The use of bulky free-space cavities and optical delay lines quickly becomes impractical for larger N since both the cavity length and number of delay lines increase linearly with N. Therefore, subsequent demonstrations of time-multiplexed DOPO networks were performed in fiber-optical platforms since they are easier to scale up to larger N. Inagaki *et al.* [16] demonstrated a fiber-based system with $N = 10000$ and 2 GHz repetition rate. It utilized four-wave mixing in a highly nonlinear fiber as the nonlinear gain mechanism. These systems consisted of a main cavity fiber loop and multiple optical delay lines similar to in free-space optical architectures. However, the number of pulses N circulating in the main cavity is much easier to increase simply by increasing the fiber length. For example, the fiber length in [16] was 1 km. A drawback is that all-optical approaches with optical delay lines can only implement limited connectivity such as nearest-neighbour coupling. This is because the signal fields are tapped off from the main cavity, amplified using similar nonlinear optical gain as in the main cavity, split into individually modulated delay lines, then recombined to interfere coherently with main cavity pulses. The number of delay lines is constrained by the total available gain before splitting and the dynamic range of the desired coupling weights J_{ij}.

One method to overcome the limited connectivity and optical delay lines is to use a measurement-feedback scheme. In this approach, the optical delay lines are replaced with homodyne detection of the signal pulse amplitudes, a field-programmable gate array (FPGA) that computes the cumulative coupling terms based on the measured signal amplitudes in real-time, and electro-optic feedback of the desired coupling pulse to interfere coherently with the appropriate main cavity pulse. This allows for arbitrary connectivity and large N so long as the FPGA can compute the coupling terms sufficiently fast to keep up with the main cavity pulses. McMahon *et al.* [17] demonstrated a time-multiplexed network of $N = 100$ DOPOs with repetition rate of 100 MHz and all-to-all connectivity using a measurement-feedback scheme. Inagaki *et al.* [18] later extended this design to $N = 2000$ with 1 GHz repetition

rate, and Honjo *et al.* [19] further extended the system to $N = 100000$ with 5 GHz repetition rate. These measurement-feedback schemes all used periodically poled lithium niobate crystals as the nonlinear gain mechanism.

Finally, there have also been efforts to utilize optoelectronic nonlinearities in time-multiplexed networks of oscillators to implement Ising machines. This is because nonlinear optical interactions in $\chi^{(2)}$ or $\chi^{(3)}$ crystals have stringent phase-matching requirements that are not readily accessible under general conditions, and optoelectronics may provide more practical flexibility in its implementation. The operating principles of optoelectronic Ising machines are analogous to networks of DOPOs, however, the nonlinear optical gain mechanism is replaced with an optoelectronic mechanism. For example, Cen *et al.* [20] demonstrated a time-multiplexed network of $N = 25600$ pulses with 250 MHz repetition rate that used an optoelectronic microwave oscillator as the nonlinearity. The pulses travelling in the main fiber cavity were still in the optical domain. Bohm *et al.* [21] and Mwamsojo *et al.* [22] replaced the parametric gain function with a sinusoidal nonlinearity based on the transfer function of a Mach-Zehnder configuration of an electro-optic intensity modulator.

Overall, time-multiplexed networks of optical or optoelectronic oscillators continue to be among the most scalable platforms for implementing Ising machines. More work needs to be done on understanding algorithmic pitfalls in networks of DOPOs and similar networks in order to escape local minima or add error-correction methods for more optimal solutions to the Ising problem [23]. Similar computational principles and optical hardware for solving the Ising problem can also be applied to find the ground state of XY spin-glass models [24, 25], which may provide more efficient mappings for some combinatorial optimization problems. Current trends suggest the move towards larger network sizes N, all-to-all connectivity, lower latency, and higher repetition rates/clock rates. A promising path towards this direction is further miniaturization of time-multiplexed oscillator networks by utilizing photonic integrated circuits for performing all-to-all couplings [26] or extremely broadband nonlinear optical gain elements [27].

Photonic neural networks

Machine learning and artificial intelligence models have experienced exponential growth over the past decade. They have revolutionized countless applications including computer vision, natural language processing, drug discovery, and generative art. The largest drivers behind this growth have been deep learning models [28]. For

example, consider a fully-connected layer in a feed-forward deep neural network. The layer's effect can be described as in Eqs. 1.3:

$$\mathbf{y}^{(k)} = W^{(k)}\mathbf{z}^{(k-1)} \,, \tag{1.3a}$$

$$\mathbf{z}^{(k)} = f(\mathbf{y}^{(k)}) \,, \tag{1.3b}$$

where $\mathbf{z}^{(k)} \in \mathbb{R}^{m_k}$ is a vector of neuron activations in the k^{th} layer containing m_k neurons, $\mathbf{y}^{(k)}$ is the result of synaptic connections given by matrix-vector multiplication between the synaptic weights of the k^{th} layer $W \in \mathbb{R}^{m_k \times m_{k-1}}$ and the previous layer's activations $\mathbf{z}^{(k-1)} \in \mathbb{R}^{m_{k-1}}$, and $f : \mathbb{R} \to \mathbb{R}$ is an element-wise nonlinear activation function such as the rectified linear unit (ReLU) function. The bulk of the computational workload comes from the matrix multiplication, which can be decomposed into elementary operations called multiply-accumulate (MAC) operations. A matrix-vector multiplication between a matrix of size $N \times N$ and a vector of size $N \times 1$ requires N^2 MAC operations. Whereas, the number of operations performed for nonlinear activations in a layer of size N is $O(N)$. Therefore, the number of MAC operations can grow quickly as the size of the neural network increases. Convolutional Neural Networks (CNNs) are a popular neural network architecture for computer vision applications, which use a series of convolutional layers as a way to efficiently extract salient image features. For simplicity, consider the discrete 1D convolution operation or sliding filter $*$ between two vectors a and b in Eq. 1.4:

$$(a * b)(n) = \sum_{m=-\infty}^{\infty} a(m) \cdot b(n-m) \,. \tag{1.4}$$

The convolution operation can also be decomposed into MAC operations, and can always be recast as a matrix multiplication.

The enormous computational demand for deep learning has rapidly outgrown general-purpose digital computers such as central processing units (CPUs), which has necessitated the use of more efficient hardware accelerators that can enable further advances [8]. For example, Graphics Processing Units (GPUs) and later application-specific integrated circuits called Tensor Processing Units (TPUs) were designed to efficiently perform matrix multiplication and MAC operations in parallel. Recently, there has been interest in harnessing analog optics for MAC operations since many linear operations such as matrix multiplication and convolution can be naturally mapped onto the physics of light propagation in various optical systems. The enormous bandwidth and parallelism inherent to optical systems presents an

exciting opportunity to accelerate deep learning beyond the limitations of digital electronics. There have been various optical neural networks exploiting time-multiplexing (often in combination with other multiplexing techniques) to perform MAC operations faster and more energy-efficiently.

Hamerly *et al.* [29] proposed the use of a photoelectric multiplication scheme in a free-space optics platform that combines time-multiplexing and spatial-multiplexing with coherent homodyne detection to perform matrix-vector multiplication. The input vector $\mathbf{x}$ is encoded in time onto an optical field amplitude such that each time bin represents a different input neuron activation x_i, and then fanned out into different spatial channels. Each spatial channel computes the result of a different neuron's synaptic connections. The weights in each row of the weight matrix A are also encoded in time onto an optical field amplitude such that each time bin represents a different weight, and coherently interfere with the input activations in the corresponding spatial channel. Each spatial channel is terminated by an optical homodyne detector. During each time step, the detector in each channel measures an optical amplitude proportional to the desired product $A_{ij}x_j$. The photoelectric charge accumulates over the duration of the optical signal to produce an electrical current that computes the desired MAC operation or dot product $\sum_j A_{ij}x_j$ giving the output neuron values after synaptic connections. It is claimed that a network layer with size $N = 1000$ can achieve an energy consumption as low as 100 zJ/MAC, which approaches the standard quantum limit, and can be operated at clock rates exceeding 1 GHz. However, there is a trade-off between the energy/MAC and the simulated accuracy for image classification tasks.

Xu *et al.* [30] demonstrated a photonic convolutional accelerator for optical neural networks in a fiber optical platform. The convolutional kernel weights are encoded using wavelength multiplexing onto different spectral lines produced by a micro-resonator frequency comb. The input signal is encoded in time onto an optical field amplitude using an electro-optic modulator such that each value of the input vector occupies a different time bin. The time bins are made to interact using dispersion in a 13 km long single-mode optical fiber, which causes different wavelength channels to propagate at different speeds. The length of fiber is chosen such that the dispersive delay between successive wavelength channels equals the time bin length of the input signal. Therefore, this system performs time-wavelength interleaving that effectively computes a convolution or sliding filter operation upon photodetection at the fiber output. Input signals up to $N = 250000$ were convolved with maximum kernel

size of $m = 90$, achieving a maximum throughput speed of 5.6×10^{12} MAC/s with an input signal clock rate of 62.9 GHz. A potential downside is the demonstrated accuracy of 88% for classification of MNIST handwritten digit images, which is far below the $> 99\%$ accuracy achieved by digital computers. This suggests that, despite impressive device-level performance, further improvements are needed at the system-level to improve the overall accuracy of time-multiplexed photonic processors for practical deep learning tasks.

The optical neural networks of Hamerly *et al.* [29] and Xu *et al.* [30] show the great potential of linear optics to accelerate linear operations such as matrix multiplication and convolution for deep learning. However, they still rely on electronics for performing the neuron nonlinear activations, which imposes a significant bottleneck on the maximum achievable computational clock rates and speeds due to the need for optoelectronic and electro-optic conversions between neural network layers. Overcoming this optoelectronic bottleneck requires moving towards all-optical neural networks. Time-multiplexing can play a central role in this regard since it allows information to be encoded onto the amplitude of ultrashort laser pulses. This can enable computational clock rates > 1 THz, which greatly exceeds electronic processors that are typically restricted to < 10 GHz clock rates. Furthermore, ultrashort laser pulses have a high peak power that enhances nonlinear optical processes. For example, Li *et al.* [31] demonstrated an all-optical implementation of the ReLU nonlinear activation function $\mathrm{ReLU}(x) = \max(0, x)$. It exploits the strong $\chi^{(2)}$ optical nonlinearity in a thin-film lithium niobate integrated nanophotonic waveguide platform. The all-optical approach achieved a maximum clock rate of 13.3 THz with energy consumption of 16 fJ per activation, which combines for a record energy-time product of 1.2×10^{-27} J $\cdot$ s per activation. The simulated accuracy for MNIST handwritten digit image classification is 98.8%.

An exciting future direction is the possibility to construct time-multiplexed optical systems containing both linear operations and nonlinear activations for an end-to-end all-optical neural network with orders of magnitude greater speed and energy efficiency compared to digital electronics [7, 32]. A limitation of current optical neural networks is that they can only efficiently perform forward inferences on trained deep neural networks. Optical neural networks cannot directly perform training of deep neural networks because there is no general method of calculating gradients in the optical domain, which are needed for the backwards pass of the backpropagation training algorithm. Therefore, current optical neural networks still

rely heavily on electronics for performing all or part of the neural network training. Finding methods for optical gradient calculation [33] or modified training algorithms for optical neural networks [34] can further advance the practical applicability of time-multiplexed optical neural networks.

Finally, there has been growing interest recently to develop biologically-plausible neural networks that more closely resemble the spiking neuron dynamics observed in biological brains. These spiking neural networks (SNNs) encode and process information entirely differently compared to other artificial neural networks. The main motivation is to produce more robust and efficient neural network models [35]. Spiking neuron dynamics is extremely costly to accurately simulate using digital computers, which has spurred the search for physical hardware systems with comparable spiking dynamics. Makinwa *et al.* [36] implemented a SNN as a time-multiplexed network of 512 DOPOs, similar to the measurement-feedback Ising machine schemes reviewed above. Each pair of coupled DOPOs acts as a single spiking neuron, which gives an effective network size of $N = 256$ spiking neurons. Characteristic collective dynamical behaviour of SNNs such as spiking neuron synchronization and chimera states can be readily observed with spiking frequencies up to 10 kHz. Note that the repetition rate of the physical DOPO pulses is 1 GHz, but this is not directly correlated with the information processing capabilities of the spiking dynamics, which occurs at a much slower rate [37]. Therefore, it is unclear whether or not the high computational clock rates of time-multiplexed optical computers has any advantage for SNNs. More work is needed to better understand the spiking dynamics of optical neurons.

Time-delay reservoir computers

Reservoir computers are a type of random recurrent neural network that are well-suited for temporal information processing tasks such as time-series prediction and sequence classification [38]. A popular architecture is called the Echo State Network (ESN) [39]. It contains an input layer with random weights, a reservoir layer containing random recurrent connections, and a linear output layer. The ESN follows the driven discrete-time dynamical system in Eqs. 1.5:

$$\mathbf{x}(t+1) = f\left(W^r\mathbf{x}(t) + W^i\mathbf{s}(t)\right) , \tag{1.5a}$$

$$\mathbf{y}(t) = W^o\mathbf{x}(t) , \tag{1.5b}$$

where t is the discrete time step, $f : \mathbb{R} \to \mathbb{R}$ is an element-wise nonlinear activation function such as the sigmoid function, $\mathbf{s} \in \mathbb{R}^m$ is the m-dimensional input signal,

$\mathbf{x} \in \mathbb{R}^N$ are the activations for the N reservoir neurons, $\mathbf{y} \in \mathbb{R}^n$ is the n-dimensional output, $W^i \in \mathbb{R}^{N\times m}$ are the input layer weights, $W^r \in \mathbb{R}^{N\times N}$ are the reservoir layer weights, and $W^o \in \mathbb{R}^{n\times N}$ are the output layer weights. Typically, the input weights W^i and reservoir weights W^r are sampled from i.i.d Gaussian or Uniform distributions, and chosen such that the echo-state property is satisfied. The echo-state property guarantees that the ESN has a kind of in-built memory of past events, which is needed for temporal information processing. The appealing feature of the reservoir computing approach is that only the output layer weights W^o are trained, while W^i and W^r are fixed after initialization. The use of random weights relaxes the need for full programmability of couplings, which is often a limiting factor for other types of optical computers. Furthermore, the optimal output weights W^o can be found using a simple linear regression or ridge regression algorithm that is easy to perform and only requires the reservoir computer to see each independent sample once during training. It avoids the full complexity and difficulties associated with the backpropagation-through-time training algorithm for recurrent neural networks that also seek to train recurrent weights.

It has been recognized that the complex nonlinear dynamics often encountered in physical systems can be used as a useful resource for physical reservoir computing [38]. In particular, time-multiplexed photonic and optoelectronic networks have been studied extensively as promising candidates for efficient physical reservoir computers. This is most commonly achieved by using a reduced reservoir scheme based on a single nonlinear node with delayed feedback [40]. Instead of N physically distinct nonlinear nodes comprising the reservoir layer, a single nonlinear node is time-multiplexed to produce N virtual nodes. In this case, the equations describing the recurrent evolution become delay differential or dynamical equations. Coupling between nodes can be achieved using either optical delay lines as described above, or by an input masking procedure in which the same input is reinjected multiple times but with different scaling factors. Therefore, time-delay photonic reservoir computers are rapidly scalable and easy-to-implement systems for fast signal processing.

An early example of a time-delay photonic reservoir computer is by Larger *et al.* [41]. The electrical input signal modulates a laser light amplitude. It is also subject to optoelectronic time-delay feedback through a 4.2 km long optical fiber loop. The single nonlinear node that is time-multiplexed is a Mach-Zehnder electro-optical modulator, which possesses a sinusoidal transfer function. A network with

$N = 400$ and computational clock rate of 19.2 MHz is demonstrated for spoken digit classification and chaotic time-series prediction. A similar approach based on a time-delayed electro-optic modulator nonlinear node was used by Paquot *et al.* [42] with network size of $N = 50$ and clock rate of 5.9 MHz for the NARMA10 time-series prediction task, nonlinear channel equalization, and spoken digit recognition. Larger *et al.* [43] later improved the performance of the system by using high-speed electro-optic phase modulators with a system size $N = 1113$ and clock rate of 17.6 GHz.

Alternative optoelectronic components were also considered for use as nonlinear nodes. For example, Duport *et al.* [44] proposed the use of a time-delay photonic reservoir computer with saturable semiconductor optical amplifier as the nonlinear node. It achieves $N = 50$ reservoir nodes with clock rate of 6.4 MHz. Similarly, Brunner *et al.* [45] used an injection current modulated semiconductor laser as the nonlinear node. It contains $N = 388$ reservoir nodes with an input clock rate of 5 GHz.

Most physical reservoir computer demonstrations only implemented the reservoir layer in hardware, and instead resorted to using digital electronics for the linear output layer readout step. This is because the reservoir layer comprises the bulk of the computation and poses the most difficulty for conventional digital computers due to complex nonlinear dynamical behaviors. Nonetheless, the need for analog-to-digital conversions limits the overall latency and clock-rate of the system since the electronic bandwidth is much less than the bandwidth of optical or optoelectronic components. Therefore, it is desirable to have an end-to-end photonic reservoir computer that directly implements all layers including the output layer. Duport *et al.* [46] showed a first step towards this goal by having fully analog input, reservoir, and output layers in a time-delay photonic reservoir computer. It operates with $N = 47$ reservoir nodes at a clock rate of 200 MHz. The analog output layer requires the addition of another electro-optic modulator for encoding output layer weights, balanced photodiode, and low-pass RLC filter. The system performs well on real-time nonlinear channel equalization and radar signal forecasting.

Recent efforts have concentrated on miniaturizing time-delay photonic reservoir computers onto chip-scale integrated platforms. This reduces the overall latency and increases the potential efficiency of the system. For example, Nakajima *et al.* [47] demonstrates the use of micro-scale silica waveguides in a platform called a planar lightwave circuit. The linear input layer consists of a series on delay lines with

programmable couplings by integrated thermo-optic phase shifters. The reservoir layer consists of a series of optical cavities with programmable attenuation and phases. This architecture uses coherent light processes that take advantage of the full complex amplitude of light, however, there is no nonlinearity in the reservoir layer. The output layer is performed digitally. The reservoir layer contains a total of $N = 512$ virtual neuron nodes at a clock rate of 3.75 GHz. It was trained to perform image classification for MNIST handwritten digits and fashionMNIST image datasets. A related approach by Wang *et al.* [48] uses silicon photonic integrated circuits with $N = 8$ and a clock rate of 60 GHz. Another chip-scale time-delay photonic reservoir computer by Borghi *et al.* [49] uses a silicon microring. Unlike previous approaches, it contains a nonlinear input layer that generates free carriers in the silicon via a strong pump beam, and then imprints this nonlinear behaviour via free carrier dispersion onto a weak continuous-wave probe beam. It achieves network size $N = 50$ with a clock rate of 20 MHz and performs iris species recognition. The slow speed despite the small size is due to the slow free carrier dynamics. This can be significantly improved by using faster absorption or nonlinear optical mechanisms for the reservoir dynamics.

Finally, the time-delay photonic reservoir computer architecture has also recently been extended to include multiple reservoir layers to produce deep reservoir computers. Although a single reservoir layer is sufficient in theory, it has been found that using multiple deep layers can improve performance and accuracy in practice. Shen *et al.* [50] demonstrate an optical fiber based system with four deep reservoir layers, each containing 80 neurons, giving a total of $N = 320$ hidden reservoir neurons. The nonlinear dynamics are produced by a series of cascaded Fabry-Perot semiconductor lasers that are optical injection-locked to a master laser. The output is calculated digitally by sampling reservoir neurons from all four layers, and is trained to perform real-time nonlinear channel equalization with a clock rate of 20 GHz. More work is needed to understand the optimal deep reservoir computer architecture for practical signal processing tasks.

Current trends show that the field is moving towards smaller, faster, and more efficient physical reservoir layers. Promising unexplored options include integrated optical parametric oscillators [51] exploiting ultrafast nonlinear optical processes. Another challenge is the limited connectivity schemes of current time-delay lines, typically only providing few-nearest neighbour connections through complicated input-masking schemes. More flexible and single-shot input operation can be

achieved by the use of intra-cavity optical delay lines. An all-optical output read-out layer, which is fully programmable and has all-to-all optical connections with reservoir neurons, is thus far absent and ultimately constrains the overall latency of current photonic reservoir computers relying on digital electronics for the output layer. Solving all of these aforementioned problems will enable the development of an all-optical and highly-scalable time-multiplexed photonic reservoir computer.

Cellular automata

Cellular automata are a class of simple computational models based on local interaction rules and algorithms. They consist of a lattice of cells with states that follow an update rule, which defines how the state of a cell evolves over time based on the states of its neighboring cells [52]. The microscopic local interactions between cells governed by the update rule produces complex phenomena and emergent patterns at the macroscopic global scale that are commonly encountered in biological and natural systems. It has been shown that even very simple rules are capable of producing Turing-universal computation [53]. Popular examples of cellular automata include von Neumann's Universal Constructor and Conway's Game of Life [54].

Consider the simplest cellular automata called elementary cellular automata [52], which are defined over a 1D lattice of cells with binary states that update according to nearest-neighbour interactions as in Eq. 1.6:

$$x_i(t+1) = f(x_{i-1}(t), x_i(t), x_{i+1}(t)) \,, \tag{1.6}$$

where t is the discrete time step, $x_i \in \{0, 1\}$ is the binary state for the i^{th} cell, and $f : \{0, 1\}^3 \rightarrow \{0, 1\}$ is the update rule. We take the lattice to extend infinitely in both left and right directions.

The rule can be depicted pictorially as a truth table with 8 entries, giving 256 possible elementary rules. The top row of the truth table entry shows the local 3-cell neighbourhood, and the bottom row shows the updated cell state during the next iteration. Iterating the rule repeatedly for all cells in the lattice produces a space-time diagram, which exhibits complex and emergent phenomena.

Li *et al.* [55] implement elementary cellular automata using a time-multiplexed photonic network. The cells are represented using pulses of light produced by a mode-locked laser. Interactions between cells are achieved by coherent interference using a series of optical delay lines with variable delays and weights to encode the desired local update rule. In this case, the nonlinearity is electronic thresholding to

enforce the binary cell states. Optoelectronic feedback is used to reinject cell states for subsequent iterations. A network with size up to $N = 1000$ cells with a clock rate of 250 MHz is used to demonstrate complex phenomena including fractals, chaos, and solitons. The system can implement any elementary cellular automata rule, including those capable of Turing-universal computation. This is, to the best of our knowledge, the first demonstration of a universal time-multiplexed optical computer.

Time-multiplexing is well-suited to implementing cellular automata rules since it only requires local interactions, not all-to-all connections. This reduces the hardware complexity for having many optical delay lines and permits an all-optical implementation for many rules. Furthermore, the same local update rule is applied to all cells. This means real-time weight updates are not required, which further simplifies the implementation of programmable couplings. Extending the cell lattice size is straightforward since the time-multiplexing concept allows the use of a single nonlinear element.

Li *et al.* [56] extended the photonic elementary cellular automata architecture by combining modern deep learning concepts to implement a neural cellular automata. The binary cell states are replaced with continuous-valued states so that the time-evolution rule becomes fully differentiable. Therefore, deep learning techniques including backpropagation training can be applied to learn update rules for specific tasks. The system additionally uses a nonlinear optical function based on a periodically-poled lithium niobate waveguide to perform nonlinear activations. It performs self-organized image classification of fashionMNIST images and shows superior parameter-efficiency and noise-robustness compared to the conventional deep learning architectures described above.

Photonic implementations of cellular automata is a nascent field with great potential as an unconventional optical computing methodology. Future work can explore different rules, cell neighbourhoods, and complex phenomena. The possibility of universal computation is an interesting feature absent in other optical computers. Time-multiplexing is a key technique to enable large-scale photonic cellular automata. Combining time-multiplexing with the reduced complexity of encoding local update rules compared to global couplings means that further integration to produce chip-scale photonic cellular automata is a practical near-term possibility.

Ref.	Application	Platform	Network Size N	Clock Rate (GHz)	Linear Operations	Nonlinearity	Connectivity
[14]	Ising machine	free space	4	0.25	optical delay lines	PPLN	all-to-all
[15]	Ising machine	free space	16	1	optical delay lines	PPLN	3-nearest neighbour
[16]	Ising machine	optical fiber	10000	2	optical delay line	four-wave mixing	nearest neighbour
[17]	Ising machine	optical fiber	100	0.1	measurement feedback	PPLN	all-to-all
[18]	Ising machine	optical fiber	2000	1	measurement feedback	PPLN	all-to-all
[19]	Ising machine	optical fiber	100000	5	measurement feedback	PPLN	all-to-all
[20]	Ising machine	optical fiber	25600	0.25	optical delay lines	optoelectronic microwave oscillator	nearest neighbour

Continued on next page. . .

Ref.	Application	Platform	Network Size N	Clock Rate (GHz)	Linear Operations	Nonlinearity	Connectivity
[21]	Ising machine	optical fiber	100	13	measurement feedback	electro-optic modulator	all-to-all
[22]	Ising machine	optical fiber	64	0.1	measurement feedback	electro-optic modulator	nearest neighbour
[29]	deep learning	free space	1000	1	photoelectric multiplication	electronic ReLU	all-to-all
[30]	deep learning	optical fiber	250000	62.9	time and wavelength interleaving	electronic ReLU	all-to-all
[32]	deep learning	free space + optical fiber	541	5	optical delay line	SOA	all-to-all
[7]	deep learning	optical fiber	~ 3000	120	optical delay lines	PPLN	nearest neighbour
[36]	spiking neural network	optical fiber	256	1	measurement feedback	PPLN	all-to-all

Continued on next page. . .

Ref.	Application	Platform	Network Size N	Clock Rate (GHz)	Linear Operations	Nonlinearity	Connectivity
[37]	spiking neural network	optical fiber	240	1	measurement feedback	PPLN	all-to-all
[41]	reservoir computer	optical fiber	400	0.0192	optical delay line	electro-optic modulator	few-nearest neighbour
[42]	reservoir computer	optical fiber	50	0.0059	optical delay line	electro-optic modulator	nearest neighbour
[43]	reservoir computer	optical fiber	1113	17.6	optical delay line	electro-optic modulator	few-nearest neighbour
[44]	reservoir computer	optical fiber	50	0.0064	optical delay line	semiconductor optical amplifier	nearest neighbour
[45]	reservoir computer	optical fiber	388	5	optical delay line	semiconductor laser	few-nearest neighbour
[46]	reservoir computer	optical fiber	47	0.2	optical delay line	electro-optic modulator	few-nearest neighbour

Continued on next page. . .

Ref.	Application	Platform	Network Size N	Clock Rate (GHz)	Linear Operations	Nonlinearity	Connectivity
[47]	reservoir computer	planar lightwave circuit	512	3.75	optical delay lines	photodetection	few-nearest neighbour
[49]	reservoir computer	silicon microring	50	0.02	optical delay line	free carrier dispersion	few-nearest neighbour
[50]	reservoir computer	optical fiber	320	20	optical delay lines	semiconductor laser	few-nearest neighbour
[48]	reservoir computer	silicon photonics	8	60	optical delay lines	photodetector	all-to-all
[57]	reservoir computer	thin-film lithium niobate	10	9.47	optical delay line	PPLN	few-nearest neighbour
[55]	cellular automata	optical fiber	1000	0.25	optical delay lines	electronic threshold	nearest neighbour
[56]	cellular automata	optical fiber	784	0.25	optical delay lines	PPLN	3-cell neighbours

Table 1.2: **Comparison of time-multiplexed photonic computers.**

1.4 Review of Parametric Nonlinear Photonic Computing

Light wave propagation described by Maxwell's equations in a vacuum is a purely linear phenomenon. This means that light waves can interfere constructively and destructively according to the principle of superposition, but otherwise do not interact. However, in the presence of intense electric fields enabled by lasers, it becomes possible to observe nonlinear optical interactions in some materials [58]. The most well-studied nonlinear optical interactions are ultrafast parametric processes that result from higher-order terms in the expansion of the electric polarization shown in Eq. 1.7,

$$\mathbf{P} = \varepsilon_0 \left(\chi^{(1)} \cdot \mathbf{E} + \chi^{(2)} : \mathbf{EE} + \chi^{(3)} \vdots \mathbf{EEE} + \cdots \right) \tag{1.7}$$

where $\chi^{(j)}$ is the j^{th} order optical susceptibility tensor of rank $j+1$. The higher order terms in the polarization are treated as a perturbation to the first-order linear term. The microscopic origins of the nonlinear optical susceptibilities are the anharmonic motion of bound electronics in the presence of intense electric fields **E**. For materials with a centrosymmetric structure, even-order terms in the polarization vanish, and the lowest-order nonlinear polarization term is of third-order. These parametric nonlinear optical processes are ordinarily very weak, however, can be greatly enhanced under strict phase-matching conditions. In this context, nanophotonics has provided a key avenue for dispersion engineering and enhanced electric field modal confinement that can allow efficient and intense nonlinear optical interactions [27]. Here we review some notable integrated platforms and implementations of parametric nonlinear photonics for computing applications.

Ultrafast nonlinear photonics has been used to address several intractable problems facing modern digital computers. Firstly, the von Neumann bottleneck, which arises due to the limited bandwidth of the shared data bus between the memory and processor in digital computers, greatly limits emerging data-heavy applications such as artificial intelligence (AI). To overcome this, Xu *et al.* [30, 59] exploited the enormous optical bandwidth and dense wavelength-division multiplexing of a $\chi^{(3)}$ Kerr soliton crystal microcomb in silica to accelerate convolution and matrix operations, which form the bulk of computations for deep learning [28]. The comb lines can be individually modulated to yield many parallel channels ($\sim$ 100). This is similar to the concept of microcomb-based communication systems, however, the computation is also performed directly in the optical domain. An impressive total throughput of up to 1.1×10^{13} operations/s was demonstrated, which is comparable to modern graphics processing units (GPUs) that are the current workhorse hardware

for AI [60]. The integrated ultrafast microcomb was essential due to its stability, ease of generation, overall conversion efficiency, and high pulse repetition rate compared to other conventional laser sources. Similar approaches based on a $\chi^{(3)}$ dissipative Kerr soliton microcomb in silicon nitride [61] and a $\chi^{(3)}$ Kerr dark soliton microcomb in AlGaAs [62] were also demonstrated. The enormous bandwidth and large number of parallel channels afforded by ultrafast microcombs has also been applied to random number generation through chaotic microcombs [63] and video signal processing [64].

Secondly, digital computers are ultimately limited by the maximum computational clock rate ($\sim$ 5 GHz), which is constrained by how fast transistors can reasonably switch due to heat dissipation limits. In fact, the photonic convolutional accelerators [30, 59, 61, 62] described above were still limited in clock rate due to optoelectronic conversions needed to perform nonlinear activation functions digitally. Beyond this, the speed of electronic computers is ultimately determined by the fundamental response time of semiconductor physics. This limitation can also be addressed using ultrafast nonlinear photonics by directly performing nonlinear functions in the optical domain. Parametric nonlinear optical interations of the ultrafast variety occur at near-instantaneous (attosecond) timescales, which is orders of magnitude faster than electronic nonlinearities. For example, Li *et al.* [31] implemented the rectified linear unit (ReLU) function, which is the most widely used nonlinear activation function for deep learning [28], by engineering the interplay between $\chi^{(2)}$ second-harmonic generation (SHG) and degenerate optical parametric amplification (DOPA) processes in a thin-film lithium niobate (TFLN) nanophotonic waveguide platform. A record-breaking maximum clock rate of 13.3 THz was demonstrated, which is an absolute speed-up of more than 3 orders of magnitude over digital computers. A similar $\chi^{(2)}$ engineering approach was also used to demonstrate ultrafast switching [65] at terahertz clock rates. These unprecedented ultrafast computational clock rates are enabled by the ultrabroadband phase-matching and remarkable conversion efficiency of nanophotonic PPLNs, which is otherwise absent in weakly-guiding or free-space PPLNs.

So far, research in optical computing benefiting from integrated ultrafast nonlinear photonics has been focused on the single component or device level, e.g., a microcomb or PPLN. A significant portion of the computation in the demonstrations described above were performed electronically or off-chip. The main future challenge will be to achieve truly end-to-end and all-optical computers capable of

ultra-high throughput and ultrafast clock rates [7]. This will require the integration (monolithic, heterogeneous, or otherwise) of many ultrafast nonlinear photonic components to build a scalable and complete system-on-chip. System-level requirements for important computing applications, such as AI, present new conceptual and algorithmic challenges that are absent when considering only single components at the device-level. Further work into developing large-scale integrated ultrafast computers based on nonlinear photonics may also enable novel unconventional computing architectures such as physical Ising solvers for NP-hard combinatorial optimization [66] and reversible computing for ultra-low energy consumption [67], which are not possible in digital electronic computers.

Ref.	**Application**	**Process**	**Platform**	**Clock Rate (GHz)**
[31]	ReLU nonlinear function	$\chi^{(2)}$ SHG and DOPA	TFLN	13300
[65]	switching and modulation	$\chi^{(2)}$ SHG and DOPA	TFLN	20800
[57]	recurrent neural network	$\chi^{(2)}$ DOPA	TFLN	9.46
[30]	convolutional neural network	$\chi^{(3)}$ Kerr soliton crystal	silica	62.9
[59]	single perceptron	$\chi^{(3)}$ Kerr soliton crystal	silica	11.9
[61]	convolution	$\chi^{(3)}$ Kerr dissipative soliton	Si_3N_4	13.5
[66]	Ising machine	$\chi^{(3)}$ FWM DOPO	Si_3N_4	4×10^{-4}
[62]	convolutional neural network	$\chi^{(3)}$ dark soliton	AlGaAs + silicon	17
[63]	random number generation	$\chi^{(3)}$ chaotic comb	AlGaAs	20
[64]	video and image signal processing	$\chi^{(3)}$ Kerr soliton crystal	silica	64

Table 1.3: **Comparison of integrated parametric nonlinear photonic computing.**

References

[1] Friedrich L. Bauer. *Origins and foundations of computing: In cooperation with Heinz Nixdorf MuseumsForum*. Springer Science & Business Media, 2009.

[2] Arthur Walter Burks. Electronic computing circuits of the ENIAC. *Proceedings of the IRE*, 35(8):756–767, 1947.

[3] John L. Hennessy and David A. Patterson. *Computer architecture: A quantitative approach*. Elsevier, 2011.

[4] IEEE. International roadmap for devices and systems (IRDS™) 2023 update. https://irds.ieee.org/editions/2023, 2023. Accessed: 2025-01-14.

[5] Robert R. Schaller. Moore's law: Past, present and future. *IEEE Spectrum*, 34 (6):52–59, 1997.

[6] Peter M. Zeitzoff and James E. Chung. A perspective from the 2003 itrs: Mosfet scaling trends, challenges, and potential solutions. *IEEE Circuits and Devices Magazine*, 21(1):4–15, 2005.

[7] Gordon H.Y. Li, Midya Parto, Jinhao Ge, Qing-Xin Ji, Maodong Gao, Yan Yu, James Williams, Robert M. Gray, Christian R. Leefmans, Nicolas Englebert, et al. All-optical computing with beyond 100-GHz clock rates. *arXiv preprint arXiv:2501.05756*, 2025.

[8] Vivienne Sze, Yu-Hsin Chen, Joel Emer, Amr Suleiman, and Zhengdong Zhang. Hardware for machine learning: Challenges and opportunities. In *2017 IEEE Custom Integrated Circuits Conference (CICC)*, pages 1–8. IEEE, 2017.

[9] Peter L. McMahon. The physics of optical computing. *Nature Reviews Physics*, 5(12):717–734, 2023.

[10] Yunping Bai, Xingyuan Xu, Mengxi Tan, Yang Sun, Yang Li, Jiayang Wu, Roberto Morandotti, Arnan Mitchell, Kun Xu, and David J. Moss. Photonic multiplexing techniques for neuromorphic computing. *Nanophotonics*, 12(5): 795–817, 2023.

[11] Naeimeh Mohseni, Peter L. McMahon, and Tim Byrnes. Ising machines as hardware solvers of combinatorial optimization problems. *Nature Reviews Physics*, 4(6):363–379, 2022.

[12] Andrew Lucas. Ising formulations of many np problems. *Frontiers in physics*, 2:5, 2014.

[13] Yoshihisa Yamamoto, Kazuyuki Aihara, Timothee Leleu, Ken-ichi Kawarabayashi, Satoshi Kako, Martin Fejer, Kyo Inoue, and Hiroki Takesue. Coherent Ising machines—Optical neural networks operating at the quantum limit. *npj Quantum Information*, 3(1):49, 2017.

[14] Alireza Marandi, Zhe Wang, Kenta Takata, Robert L Byer, and Yoshihisa Yamamoto. Network of time-multiplexed optical parametric oscillators as a coherent ising machine. *Nature Photonics*, 8(12):937–942, 2014.

[15] Kenta Takata, Alireza Marandi, Ryan Hamerly, Yoshitaka Haribara, Daiki Maruo, Shuhei Tamate, Hiromasa Sakaguchi, Shoko Utsunomiya, and Yoshihisa Yamamoto. A 16-bit coherent ising machine for one-dimensional ring and cubic graph problems. *Scientific Reports*, 6(1):34089, 2016.

[16] Takahiro Inagaki, Kensuke Inaba, Ryan Hamerly, Kyo Inoue, Yoshihisa Yamamoto, and Hiroki Takesue. Large-scale ising spin network based on degenerate optical parametric oscillators. *Nature Photonics*, 10(6):415–419, 2016.

[17] Peter L. McMahon, Alireza Marandi, Yoshitaka Haribara, Ryan Hamerly, Carsten Langrock, Shuhei Tamate, Takahiro Inagaki, Hiroki Takesue, Shoko Utsunomiya, Kazuyuki Aihara, et al. A fully programmable 100-spin coherent ising machine with all-to-all connections. *Science*, 354(6312):614–617, 2016.

[18] Takahiro Inagaki, Yoshitaka Haribara, Koji Igarashi, Tomohiro Sonobe, Shuhei Tamate, Toshimori Honjo, Alireza Marandi, Peter L. McMahon, Takeshi Umeki, Koji Enbutsu, et al. A coherent ising machine for 2000-node optimization problems. *Science*, 354(6312):603–606, 2016.

[19] Toshimori Honjo, Tomohiro Sonobe, Kensuke Inaba, Takahiro Inagaki, Takuya Ikuta, Yasuhiro Yamada, Takushi Kazama, Koji Enbutsu, Takeshi Umeki, Ryoichi Kasahara, et al. 100,000-spin coherent ising machine. *Science Advances*, 7(40):eabh0952, 2021.

[20] Qizhuang Cen, Hao Ding, Tengfei Hao, Shanhong Guan, Zhiqiang Qin, Jiaming Lyu, Wei Li, Ninghua Zhu, Kun Xu, Yitang Dai, et al. Large-scale coherent ising machine based on optoelectronic parametric oscillator. *Light: Science & Applications*, 11(1):333, 2022.

[21] Fabian Böhm, Guy Verschaffelt, and Guy Van der Sande. A poor man's coherent ising machine based on opto-electronic feedback systems for solving optimization problems. *Nature Communications*, 10(1):3538, 2019.

[22] Nickson Mwamsojo, Frederic Lehmann, Kamel Merghem, Badr-Eddine Benkelfat, and Yann Frignac. Optoelectronic coherent ising machine for combinatorial optimization problems. *Optics Letters*, 48(8):2150–2153, 2023.

[23] Sam Reifenstein, Satoshi Kako, Farad Khoyratee, Timothée Leleu, and Yoshihisa Yamamoto. Coherent ising machines with optical error correction circuits. *Advanced Quantum Technologies*, 4(11):2100077, 2021.

[24] Ryan Hamerly, Kensuke Inaba, Takahiro Inagaki, Hiroki Takesue, Yoshihisa Yamamoto, and Hideo Mabuchi. Topological defect formation in 1d and 2d spin chains realized by network of optical parametric oscillators. *International Journal of Modern Physics B*, 30(25):1630014, 2016.

[25] Mostafa Honari-Latifpour and Mohammad-Ali Miri. Mapping the x y hamiltonian onto a network of coupled lasers. *Physical Review Research*, 2(4): 043335, 2020.

[26] Mihika Prabhu, Charles Roques-Carmes, Yichen Shen, Nicholas Harris, Li Jing, Jacques Carolan, Ryan Hamerly, Tom Baehr-Jones, Michael Hochberg, Vladimir Čeperić, et al. Accelerating recurrent ising machines in photonic integrated circuits. *Optica*, 7(5):551–558, 2020.

[27] Luis Ledezma, Ryoto Sekine, Qiushi Guo, Rajveer Nehra, Saman Jahani, and Alireza Marandi. Intense optical parametric amplification in dispersion-engineered nanophotonic lithium niobate waveguides. *Optica*, 9(3):303–308, 2022.

[28] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[29] Ryan Hamerly, Liane Bernstein, Alexander Sludds, Marin Soljačić, and Dirk Englund. Large-scale optical neural networks based on photoelectric multiplication. *Physical Review X*, 9(2):021032, 2019.

[30] Xingyuan Xu, Mengxi Tan, Bill Corcoran, Jiayang Wu, Andreas Boes, Thach G. Nguyen, Sai T. Chu, Brent E. Little, Damien G. Hicks, Roberto Morandotti, et al. 11 tops photonic convolutional accelerator for optical neural networks. *Nature*, 589(7840):44–51, 2021.

[31] Gordon H.Y. Li, Ryoto Sekine, Rajveer Nehra, Robert M. Gray, Luis Ledezma, Qiushi Guo, and Alireza Marandi. All-optical ultrafast relu function for energy-efficient nanophotonic deep learning. *Nanophotonics*, 12(5):847–855, 2023.

[32] Tiankuang Zhou, Wei Wu, Jinzhi Zhang, Shaoliang Yu, and Lu Fang. Ultrafast dynamic machine vision with spatiotemporal photonic computing. *Science Advances*, 9(23):eadg4391, 2023.

[33] Xianxin Guo, Thomas D. Barrett, Zhiming M. Wang, and AI Lvovsky. Backpropagation through nonlinear units for the all-optical training of neural networks. *Photonics Research*, 9(3):B71–B80, 2021.

[34] Logan G. Wright, Tatsuhiro Onodera, Martin M. Stein, Tianyu Wang, Darren T. Schachter, Zoey Hu, and Peter L. McMahon. Deep physical neural networks trained with backpropagation. *Nature*, 601(7894):549–555, 2022.

[35] Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kheradpisheh, Timothée Masquelier, and Anthony Maida. Deep learning in spiking neural networks. *Neural Networks*, 111:47–63, 2019.

[36] Tumi Makinwa, Kensuke Inaba, Takahiro Inagaki, Yasuhiro Yamada, Timothée Leleu, Toshimori Honjo, Takuya Ikuta, Koji Enbutsu, Takeshi Umeki, Ryoichi Kasahara, et al. Experimental observation of chimera states in spiking neural networks based on degenerate optical parametric oscillators. *Communications Physics*, 6(1):121, 2023.

[37] Takahiro Inagaki, Kensuke Inaba, Timothée Leleu, Toshimori Honjo, Takuya Ikuta, Koji Enbutsu, Takeshi Umeki, Ryoichi Kasahara, Kazuyuki Aihara, and Hiroki Takesue. Collective and synchronous dynamics of photonic spiking neurons. *Nature Communications*, 12(1):1–8, 2021.

[38] Gouhei Tanaka, Toshiyuki Yamane, Jean Benoit Héroux, Ryosho Nakane, Naoki Kanazawa, Seiji Takeda, Hidetoshi Numata, Daiju Nakano, and Akira Hirose. Recent advances in physical reservoir computing: A review. *Neural Networks*, 115:100–123, 2019.

[39] Ali Rodan and Peter Tino. Minimum complexity echo state network. *IEEE Transactions on Neural Networks*, 22(1):131–144, 2010.

[40] Lennert Appeltant, Miguel Cornelles Soriano, Guy Van der Sande, Jan Danckaert, Serge Massar, Joni Dambre, Benjamin Schrauwen, Claudio R Mirasso, and Ingo Fischer. Information processing using a single dynamical node as complex system. *Nature Communications*, 2(1):468, 2011.

[41] Laurent Larger, Miguel C Soriano, Daniel Brunner, Lennert Appeltant, Jose M Gutiérrez, Luis Pesquera, Claudio R Mirasso, and Ingo Fischer. Photonic information processing beyond turing: An optoelectronic implementation of reservoir computing. *Optics Express*, 20(3):3241–3249, 2012.

[42] Yvan Paquot, Francois Duport, Antoneo Smerieri, Joni Dambre, Benjamin Schrauwen, Marc Haelterman, and Serge Massar. Optoelectronic reservoir computing. *Scientific Reports*, 2(1):287, 2012.

[43] Laurent Larger, Antonio Baylón-Fuentes, Romain Martinenghi, Vladimir S. Udaltsov, Yanne K. Chembo, and Maxime Jacquot. High-speed photonic reservoir computing using a time-delay-based architecture: Million words per second classification. *Physical Review X*, 7(1):011015, 2017.

[44] François Duport, Bendix Schneider, Anteo Smerieri, Marc Haelterman, and Serge Massar. All-optical reservoir computing. *Optics Express*, 20(20):22783–22795, 2012.

[45] Daniel Brunner, Miguel C. Soriano, Claudio R. Mirasso, and Ingo Fischer. Parallel photonic information processing at gigabyte per second data rates using transient states. *Nature Communications*, 4(1):1364, 2013.

[46] François Duport, Anteo Smerieri, Akram Akrout, Marc Haelterman, and Serge Massar. Fully analogue photonic reservoir computer. *Scientific Reports*, 6(1): 22381, 2016.

[47] Mitsumasa Nakajima, Kenji Tanaka, and Toshikazu Hashimoto. Scalable reservoir computing on coherent linear photonic processor. *Communications Physics*, 4(1):20, 2021.

[48] Dongliang Wang, Yikun Nie, Gaolei Hu, Hon Ki Tsang, and Chaoran Huang. Ultrafast silicon photonic reservoir computing engine delivering over 200 tops. *Nature Communications*, 15(1):10841, 2024.

[49] Massimo Borghi, Stefano Biasi, and Lorenzo Pavesi. Reservoir computing based on a silicon microring and time multiplexing for binary and analog operations. *Scientific Reports*, 11(1):15642, 2021.

[50] Yi-Wei Shen, Rui-Qian Li, Guan-Ting Liu, Jingyi Yu, Xuming He, Lilin Yi, and Cheng Wang. Deep photonic reservoir computing recurrent network. *Optica*, 10(12):1745–1751, 2023.

[51] Luis Ledezma, Arkadev Roy, Luis Costa, Ryoto Sekine, Robert Gray, Qiushi Guo, Rajveer Nehra, Ryan M Briggs, and Alireza Marandi. Octave-spanning tunable infrared parametric oscillators in nanophotonics. *Science Advances*, 9 (30):eadf9711, 2023.

[52] Stephen Wolfram. Statistical mechanics of cellular automata. *Reviews of Modern Physics*, 55(3):601, 1983.

[53] Matthew Cook et al. Universality in elementary cellular automata. *Complex Systems*, 15(1):1–40, 2004.

[54] Martin Gardner. The fantastic combinations of john conway's new solitaire game "Life". *Scientific American*, 223:120–123, 1970.

[55] Gordon H.Y. Li, Christian R. Leefmans, James Williams, and Alireza Marandi. Photonic elementary cellular automata for simulation of complex phenomena. *Light: Science & Applications*, 12(1):132, 2023.

[56] Gordon H.Y. Li, Christian R. Leefmans, James Williams, Robert M. Gray, Midya Parto, and Alireza Marandi. Deep learning with photonic neural cellular automata. *Light: Science & Applications*, 13(1):283, 2024.

[57] Midya Parto, Gordon H.Y. Li, Ryoto Sekine, Robert M. Gray, Luis L. Ledezma, James Williams, Arkadev Roy, and Alireza Marandi. Ultrafast neuromorphic computing with nanophotonic optical parametric oscillators. *arXiv preprint arXiv:2501.16604*, 2025.

[58] Toshiaki Suhara and Masatoshi Fujimura. *Waveguide nonlinear-optic devices*, volume 11. Springer Science & Business Media, 2013.

[59] Xingyuan Xu, Mengxi Tan, Bill Corcoran, Jiayang Wu, Thach G. Nguyen, Andreas Boes, Sai T. Chu, Brent E. Little, Roberto Morandotti, Arnan Mitchell, et al. Photonic perceptron based on a kerr microcomb for high-speed, scalable, optical neural networks. *Laser & Photonics Reviews*, 14(10):2000070, 2020.

[60] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE*, 105(12):2295–2329, 2017.

[61] Johannes Feldmann, Nathan Youngblood, Maxim Karpov, Helge Gehring, Xuan Li, Maik Stappers, Manuel Le Gallo, Xin Fu, Anton Lukashchuk, Arslan S. Raja, et al. Parallel convolutional processing using an integrated photonic tensor core. *Nature*, 589(7840):52–58, 2021.

[62] Bowen Bai, Qipeng Yang, Haowen Shu, Lin Chang, Fenghe Yang, Bitao Shen, Zihan Tao, Jing Wang, Shaofu Xu, Weiqiang Xie, et al. Microcomb-based integrated photonic processing unit. *Nature Communications*, 14(1):66, 2023.

[63] Bitao Shen, Haowen Shu, Weiqiang Xie, Ruixuan Chen, Zhi Liu, Zhangfeng Ge, Xuguang Zhang, Yimeng Wang, Yunhao Zhang, Buwen Cheng, et al. Harnessing microcomb-based parallel chaos for random number generation and optical decision making. *Nature Communications*, 14(1):4590, 2023.

[64] Mengxi Tan, Xingyuan Xu, Andreas Boes, Bill Corcoran, Thach G. Nguyen, Sai T. Chu, Brent E. Little, Roberto Morandotti, Jiayang Wu, Arnan Mitchell, et al. Photonic signal processor based on a kerr microcomb for real-time video image processing. *Communications Engineering*, 2(1):94, 2023.

[65] Qiushi Guo, Ryoto Sekine, Luis Ledezma, Rajveer Nehra, Devin J. Dean, Arkadev Roy, Robert M. Gray, Saman Jahani, and Alireza Marandi. Femtojoule femtosecond all-optical switching in lithium niobate nanophotonics. *Nature Photonics*, 16(9):625–631, 2022.

[66] Yoshitomo Okawachi, Mengjie Yu, Jae K. Jang, Xingchen Ji, Yun Zhao, Bok Young Kim, Michal Lipson, and Alexander L. Gaeta. Demonstration of chip-based coupled degenerate optical parametric oscillators for realizing a nanophotonic spin-glass. *Nature Communications*, 11(1):4119, 2020.

[67] Eyal Cohen, Shlomi Dolev, and Michael Rosenblit. All-optical design for inherently energy-conserving reversible gates and circuits. *Nature Communications*, 7(1):11424, 2016.

Chapter 2

ALL-OPTICAL ULTRAFAST RELU FUNCTION FOR ENERGY-EFFICIENT NANOPHOTONIC DEEP LEARNING

Gordon H.Y. Li*, Ryoto Sekine*, Rajveer Nehra*, Robert M. Gray*, Luis Ledezma, Qiushi Guo, and Alireza Marandi. All-optical ultrafast ReLU function for energy-efficient nanophotonic deep learning. *Nanophotonics*, 12(5):847–855, 2023. doi:10.1515/nanoph-2022-0137.
G.H.Y.L. conceived the project, developed the theory, executed the numerical simulations, designed the experiments, participated in the experimental measurements, analyzed the data, and wrote the manuscript.
* denotes equal contributions.

2.1 Abstract

In recent years, the computational demands of deep learning applications have necessitated the introduction of energy-efficient hardware accelerators. Optical neural networks are a promising option; however, thus far they have been largely limited by the lack of energy-efficient nonlinear optical functions. Here, we experimentally demonstrate an all-optical Rectified Linear Unit (ReLU), which is the most widely used nonlinear activation function for deep learning, using a periodically-poled thin-film lithium niobate nanophotonic waveguide and achieve ultra-low energies in the regime of femtojoules per activation with near-instantaneous operation. Our results provide a clear and practical path towards truly all-optical, energy-efficient nanophotonic deep learning.

2.2 Introduction

Over the past decade, deep learning has revolutionized many important applications including computer vision, speech recognition, and natural language processing [1]. However, the explosive growth of modern deep learning models has quickly outpaced improvements in conventional von Neumann computing architectures and ushered in the use of dedicated hardware accelerators. The quest for ever-faster and more energy-efficient hardware for deep learning began with exploiting the graphics processing unit (GPU), then application-specific integrated circuits such as Google's tensor processing unit (TPU), and more recently the development of non-von Neumann analog architectures [2, 3]. Naturally, photonics has attracted

attention as a promising candidate due to its potential for massive parallelism and ultrafast operation [4]. Indeed, optical neural networks (ONNs) have been experimentally demonstrated in a variety of platforms including free-space optics [5–11], optical fiber [12–17], and photonic integrated circuits [18–22].

In general, deep neural networks require two major types of computations: (1) linear operations in the form of matrix multiplications and convolutions, which represent the synaptic connections of the network, and (2) nonlinear activation functions, which represent the neuron activations. ONNs excel at performing energy-efficient linear operations in the optical domain, which forms the bulk of computations for deep learning. However, a major remaining roadblock is achieving scalable energy-efficient nonlinear activation functions, which comprises a smaller but essential part of the deep learning workload. Thus, the majority of ONN implementations still opt to utilize digital electronics to perform the nonlinear activation functions. In doing so, the optoelectronic and analog-to-digital conversion typically imposes significant speed and energy limitations. On the other hand, the demonstrated all-optical approaches based on various processes [7, 13, 17, 19, 23–25] are still too energy-intensive and/or slow compared to electronics. This is because photon-photon interactions are typically weak and require either high light intensities or high-Q resonant cavities, both of which are undesirable for scalable computing purposes. An all-optical, ultrafast, and energy-efficient nonlinear activation function is yet to be demonstrated to unlock the full capabilities of ONNs. Such a function should also be compact, highly scalable, and compatible with existing deep learning models.

In this work, we propose and experimentally demonstrate the first photonic device, to the best of our knowledge, that satisfies all the aforementioned criteria for an all-optical nonlinear activation function. It implements the Rectified Linear Unit (ReLU) function, defined as $\mathrm{ReLU}(x) = \max(0, x)$, which is one of the most widely used nonlinear activation functions for deep learning. The widespread adoption of the ReLU function was essential in sparking the deep learning revolution due to its favorable properties for backpropagation training and simple implementation in digital electronics [1]. However, its optical implementation has remained challenging and posed a major hurdle for the real-world applicability of ONNs.

2.3 Method

Principle of operation

The operating principle of our device is illustrated in Fig. 2.1. We encode the signal information into the coherent optical field of pulses centered at frequency 2ω, with positive values represented by $\phi_{2\omega} = +\pi/2$ phase states, and negative values represented by $\phi_{2\omega} = -\pi/2$ phase states. By co-propagating the signal pulses with bias pulses centered at frequency ω, with fixed input power and phase at $\phi_\omega = +\pi/2$, we can induce different nonlinear optical effects for the two possible $\phi_{2\omega}$ signal phases depending on the value of the phase relationship $2\phi_\omega - \phi_{2\omega}$. For the positive signal values with phase $\phi_{2\omega} = +\pi/2$, the phase relationship yields $2\phi_\omega - \phi_{2\omega} = +\pi/2$. This induces second harmonic generation (SHG), which is a $\chi^{(2)}$ nonlinear optical process that converts two photons of frequency ω into a photon of frequency 2ω, hence depleting ω and amplifying 2ω. Conversely, for the negative signal values with phase $\phi_{2\omega} = -\pi/2$, the phase relationship yields $2\phi_\omega - \phi_{2\omega} = 3\pi/2 \rightarrow -\pi/2$. This induces degenerate optical parametric amplification (DOPA),

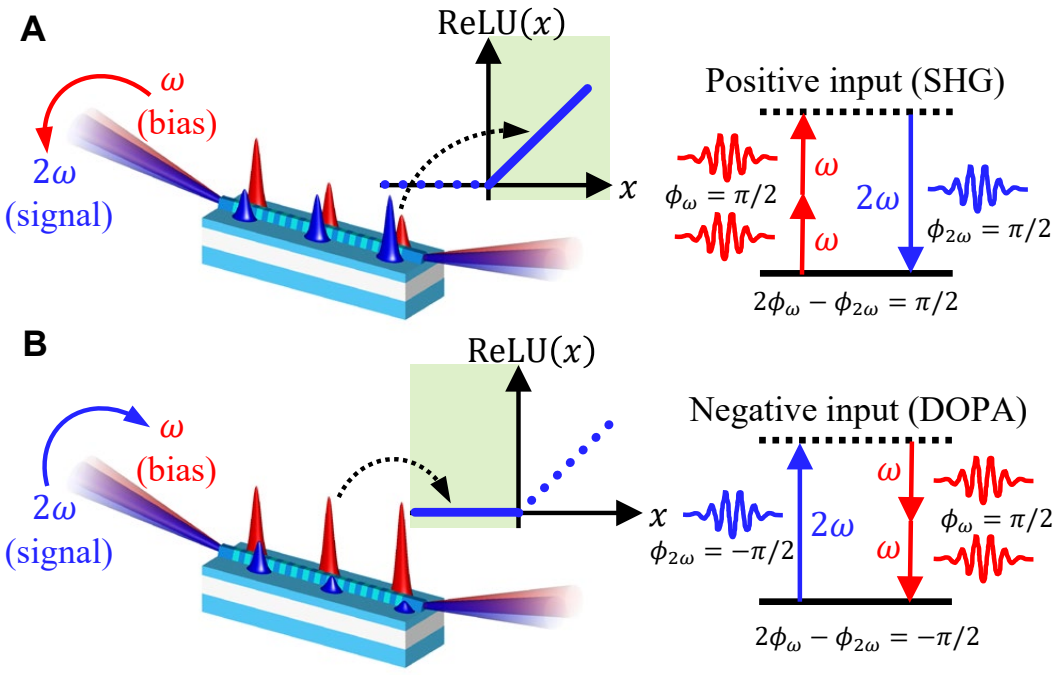

Figure 2.1: **Operating principle of the all-optical ReLU function using a nonlinear photonic waveguide.** (A) For positive inputs with phase of $\phi_{2\omega} = +\pi/2$, the phase relationship between the signal and bias is $2\phi_\omega - \phi_{2\omega} = \pi/2$, which causes SHG that depletes ω and amplifies 2ω. (B) For negative inputs, $\phi_{2\omega} = -\pi/2$, the phase relationship $2\phi_\omega - \phi_{2\omega} = 3\pi/2 \rightarrow -\pi/2$ causes DOPA that amplifies ω and depletes 2ω.

which is the inverse process of SHG that converts a photon of frequency 2ω into two photons of frequency ω, hence depleting 2ω and amplifying ω. By judiciously choosing the length and bias power, we can achieve the desired shape of the ReLU function. We emphasize that our approach utilizes coherent parametric processes which allows us to implement both positive and negative values (i.e., the information is encoded in the field amplitude), unlike previous optical [7, 13, 17, 19, 23–25] and optoelectronic methods [11, 14–16, 21, 26–29] based on incoherent absorption processes that can only implement positive values (i.e., the information is encoded in the optical power).

Device design

To implement the $\chi^{(2)}$-based ReLU function, we use a periodically poled thin-film lithium niobate (PPLN) nanophotonic waveguide that exploits the strong and instantaneous $\chi^{(2)}$ optical nonlinearity of lithium niobate and tight spatial confinement of the waveguide modes to enhance the nonlinearity [30]. Additionally, careful qausi-phase matching and dispersion engineering enables ultra-broadband and low-energy interactions over mm-long propagation lengths, further enhancing the nonlinear optical processes using femtosecond laser pulses [31–33]. Images of the device are shown in Fig. 2.2. The PPLN nanophotonic waveguide is $L = 2.5$ mm long and was fabricated on a 700-nm thick X-cut MgO-doped lithium niobate thin-film on 2-μm

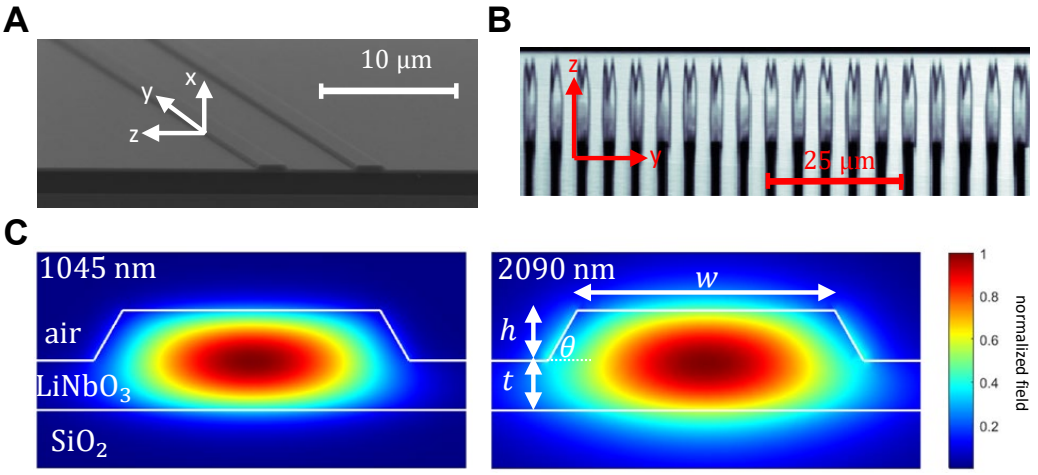

Figure 2.2: **Images of the PPLN nanophotonic waveguide.** (A) Scanning electron microscope image of the ridge waveguide. (B) Two-photon absorption microscope image of the PPLN ferroelectric domains with poling period of 5 μm. (C) Simulated electric field distributions of the fundamental TE modes at 1045 nm (2ω) and 2090 nm (ω).

thick SiO_2 with lithium niobate substrate by dry etching with Ar^+ plasma, achieving smooth ridge side-walls with slant angle of $\theta \approx 60^\circ$ as shown in Fig. 2.2A. The waveguide was electrically poled with a period of 5.17 μm, as shown in Fig. 2.2B, to ensure efficient SHG and DOPA. Dispersion engineering of the fundamental TE mode of the ridge waveguide, shown in Fig. 2.2C, allows for negligible group velocity mismatch and group velocity dispersion of ω and 2ω pulses centered at 1045 nm and 2090 nm, respectively. This enforces good temporal overlap of the pulses over the entire PPLN propagation length. The ideal parameters found from simulation were a ridge top width of $w = 1700$ nm and etch-depth of $h = 350$ nm. See [33] for further details about fabrication and dispersion engineering of PPLN nanophotonic waveguides.

2.4 Results

Femtojoule ReLU function

The measured response of the all-optical ReLU is shown in Fig. 2.3. The nonlinear function given by the PPLN was measured using a free-space chip characterization setup. The source at 1045 nm (signal) was a Yb:fiber mode-locked laser producing 75-fs long pulses at a 250-MHz repetition rate (Menlo Systems Orange). The same laser pumped a homemade degenerate optical parametric oscillator to generate the pulses at 2090 nm (bias). The 2ω and ω pulses were coupled into and out of the PPLN using reflective objectives focused on the waveguide facets. Finally, the relative phase of the 2ω signal and ω bias was set using a delay arm, and the power varied using a tunable attenuator. See Supplementary Section 1 for further details about the experimental setup.

Our experimental results show good agreement with the ideal ReLU function ($R^2 = 0.9895$), and demonstrates energy-efficient signal pulse energies in the regime of femtojoules per activation. Note that the important feature of the function is its nonlinear shape, since scaling/shifting the horizontal/vertical directions can be accomplished with linear optical transformations. In theory, the ideal ReLU function requires an arbitrarily long PPLN and low bias pulse energy. However, in practice we must choose the bias pulse energy so as to best approximate the ReLU function given our fixed device length. Thus, there are small discrepancies around $E_{2\omega}(0) = 0$, since neither the SHG nor DOPA processes sufficiently saturate at the ultra-low energies. The maximum cutoff pulse energy is determined by the onset of supercontinuum generation from strong back-conversion processes, which undesirably degrades the pulse shape. To verify that the expected device response matches

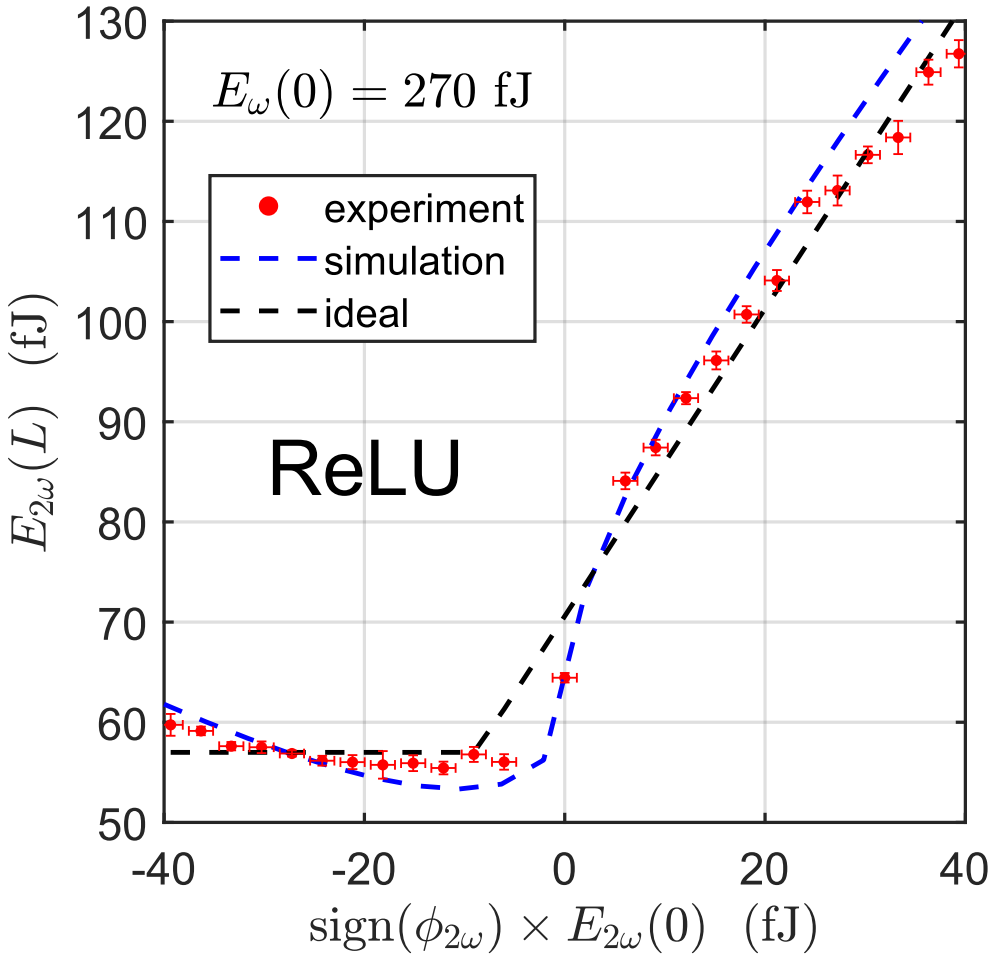

Figure 2.3: **Output signal pulse energy versus input signal pulse energy for both negative and positive inputs.** There is good agreement between the ideal ReLU function (dashed black line), simulation (dashed blue line) and experimental results (red circles) for a bias pulse energy of $E_\omega(0) = 270$ fJ, and signal pulse energies of femtojoules per activation.

our physical picture of the operating principle, we also performed nonlinear pulse propagation simulations of the PPLN nanophotonic waveguide. See Supplementary Section 3 for more details about the simulation methods.

Remarkably, we show that the PPLN nanophotonic waveguide can also approximate other commonly used variants of the ReLU function, simply by tuning the bias pulse energy. For example, the Exponential Linear Unit (ELU) defined as $\mathrm{ELU}(x) = x$ if $x > 0$ and $\mathrm{ELU}(x) = \exp(x) - 1$ if $x < 0$, which has been shown to outperform the ReLU function in certain cases [34], is achieved using a bias pulse energy of $E_\omega(0) = 450$ fJ as shown in Fig. 2.4A. In addition, we also implement the

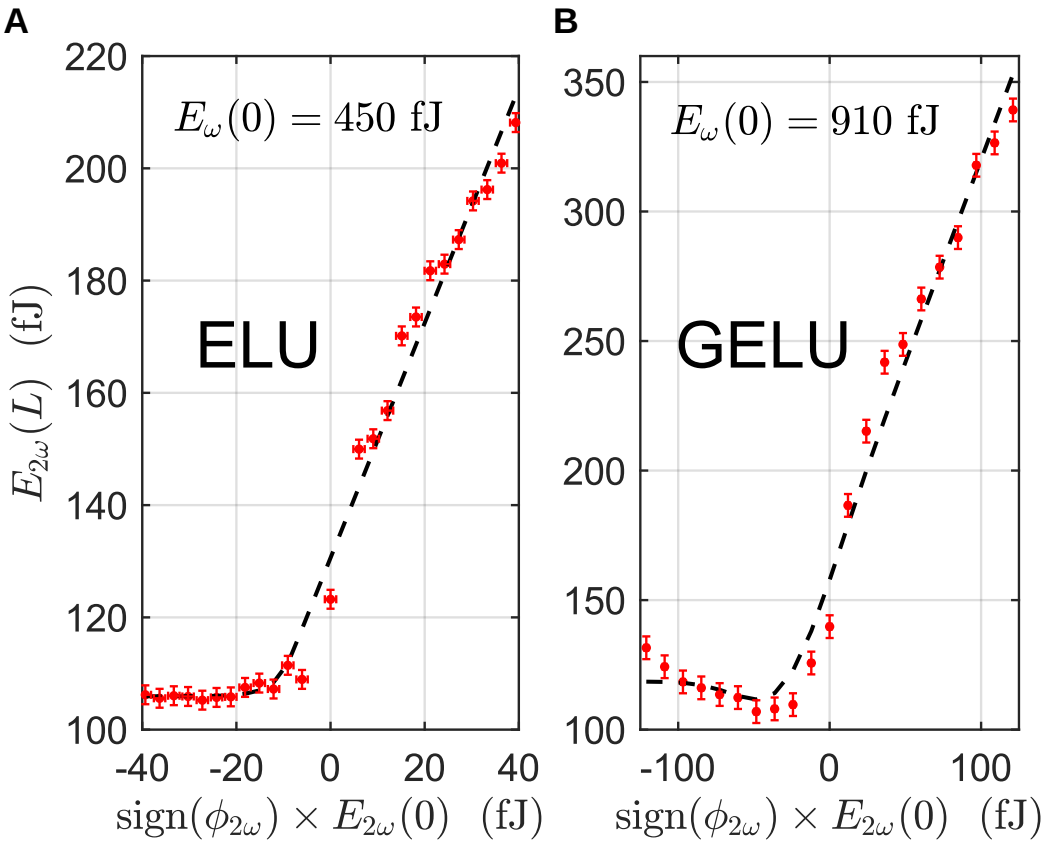

Figure 2.4: **Other variants of the ReLU function can be approximated by tuning the bias pulse energy.** For example, the (A) ELU function using bias pulse energy of $E_\omega(0) = 450$ fJ and (B) GELU function using bias pulse energy of $E_\omega(0) = 910$ fJ. Ideal function curves are shown by the dashed black lines, and experimental results with red circles.

Gaussian Error Linear Unit (GELU) defined as $\mathrm{GELU}(x) = x\Phi(x)$ where $\Phi(x)$ is the Gaussian cumulative distribution using a bias pulse energy of $E_\omega(0) = 910$ fJ as shown in Fig. 2.4B. The GELU function is used extensively in Transformer networks for natural language processing, which are regularly amongst the largest deep learning models [35]. Thus, our all-optical PPLN nanophotonic waveguide implementation gains greater real-world applicability by being compatible with a wide range of existing deep learning models, especially the largest models where energy efficiency is paramount. Indeed, compatibility has been problematic in previous implementations of optical [7, 17, 23–25] and optoelectronic [11, 14, 15, 26, 29] nonlinear activation functions, which do not reflect the most commonly used functions in digital electronic neural networks. By alleviating this problem, we expand the potential functionality of ONNs by avoiding the need to train new

specialized models.

Ultrafast time response

Ideally, the time per activation should be near-instantaneous due to the ultrafast $\chi^{(2)}$ nonlinearity in lithium niobate. However, in practice, the response time is limited by the finite phase-matching bandwidth as well as non-zero group velocity mismatch, group velocity dispersion, and higher-order dispersion terms. To determine the response time of the device, we used the pump-probe technique commonly used to characterize all-optical switches [32, 36, 37] (see Supplementary for more details). In this case, the pump pulse is the ω pulse and the probe pulse is the 2ω pulse. We measured the ultrafast ReLU dynamics by varying the time delay between the ω

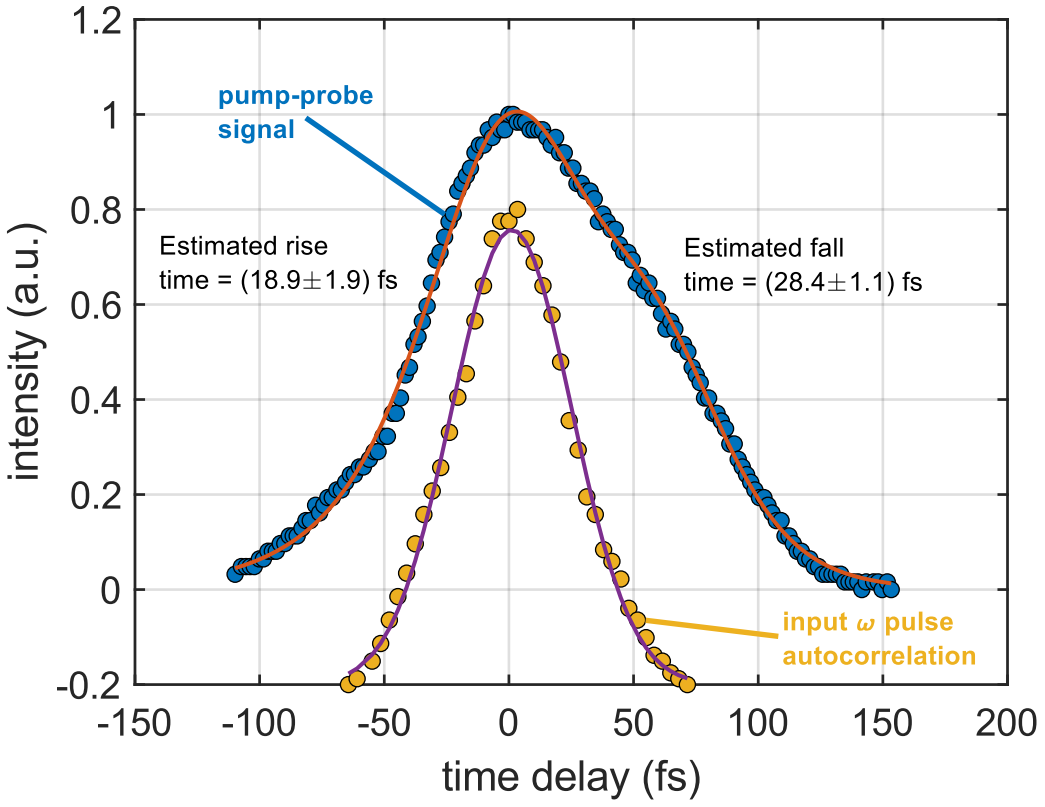

Figure 2.5: **Pump-probe ultrafast timing measurements of the ReLU dynamics.** The autocorrelation (yellow circles shifted vertically for clarity) of the input ω pulse is well-explained by a Gaussian profile (purple line) with FWHM of (56.4 ± 1.5) fs. The pump-probe signal obtained at a fixed pulse energy (blue circles) is fit (orange line) by convolving the input autocorrelation with exponential growth and decay for positive and negative time delays, respectively. The best fit yields a rise time of (18.9 ± 1.9) fs and a fall time of (28.4 ± 1.1) fs.

and 2ω pulses at a fixed pulse energy. Fig. 2.5 shows the intensity envelope of the pump-probe signal as the time delay is varied as well as the autocorrelation of the input ω pulse.

The input autocorrelation is well-explained by a Gaussian profile with FWHM of (56.4 ± 1.5) fs. We extract the characteristic rise and fall times by fitting the pump-probe signal with exponential growth and decay functions for positive and negative time delays, respectively, convolved with the input autocorrelation. The best fit yields a rise time of (18.9 ± 1.9) fs and a fall time of (28.4 ± 1.1) fs. This implies that the characteristic response time of the ReLU dynamics is (47.3 ± 3.0) fs, thus verifying that the ultrafast optical nonlinearity is responsible for the ReLU response, and ruling out the possibility of any slower optical nonlinearities such as photorefractive or thermo-optic effects. Therefore, we can reasonably regard the 2ω signal pulse length of ~ 75 fs as the time per activation for the all-optical ReLU. We note that better dispersion engineering can lead to even faster activation times.

Simulated deep learning performance

One distinct advantage of our approach is that, unlike previous all-optical [19] and optoelectronic [21] nonlinear activation functions, it can faithfully reproduce the ideal ReLU function, which uses both positive and negative values. Therefore, we can leverage the large number of existing pretrained deep learning models that use the ReLU function (or its variants) for nonlinear activations. Although ONNs have been demonstrated that accurately reproduce linear operations such as matrix multiplication and convolution, the use of atypical nonlinear activation functions in the optical domain has required the training of new custom deep learning models [38, 39]. To improve upon this, we simulated the performance of the all-optical ReLU function when used as part of a pretrained convolutional neural network (CNN) for the prototypical task of MNIST handwritten digits image classification [40]. The MNIST dataset contains 28×28 pixels gray-scale images of handwritten digits with 50000 training samples and 10000 test samples. We used a standard CNN architecture (see Supplementary Section 5 for full details) containing convolutional layers and ideal ReLU layers followed by a fully-connected layer and softmax classification output. The pretrained CNN achieved an ideal test accuracy of 99.13%. Next, the ideal ReLU layers were replaced with custom layers representing the experimentally measured ReLU response (after proper shifting/scaling) without changing any of the other layers. This caused a slight drop in test accuracy to 98.8% due to the slight deviations between the experimentally measured and

ideal ReLU functions. To remedy this, the CNN was then fine-tuned by training for only 2 epochs (the CNN sees each sample once per epoch) to regain the ideal pretrained model accuracy of 99.13% as shown in Fig. 2.6. Fine-tuning is necessary for any analog hardware implementation due to unavoidable fabrication errors, noise

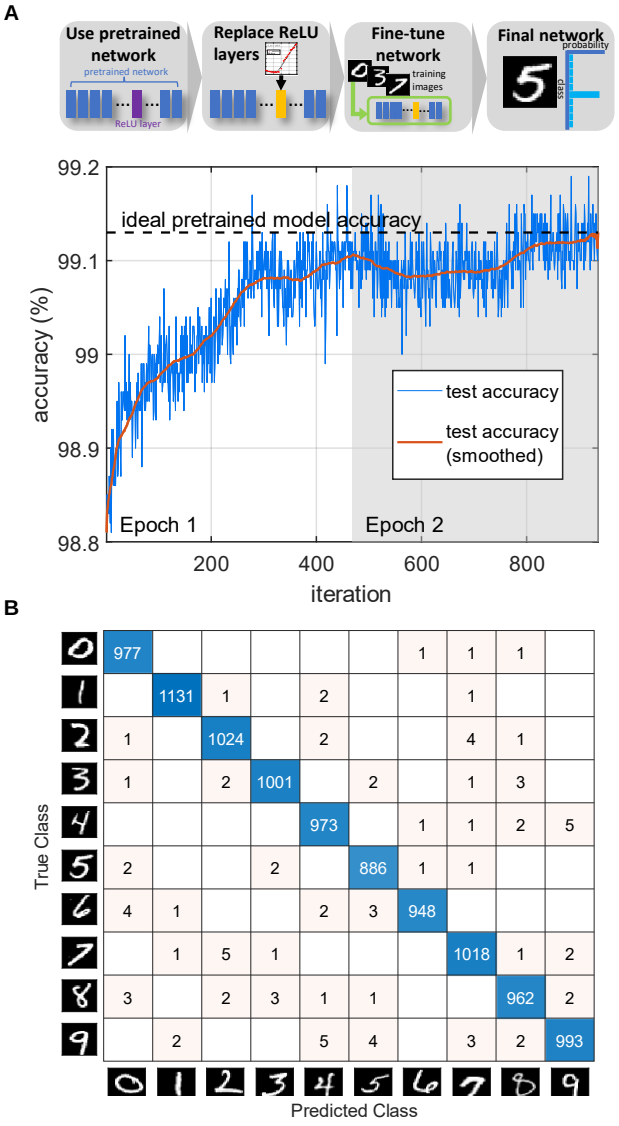

Figure 2.6: **Simulated deep learning performance of the experimentally measured all-optical ReLU function for MNIST handwritten digits image classification.** (A) A pretrained CNN was used where the ideal ReLU layers are replaced with custom layers representing the experimentally measured ReLU response (after shifting/scaling) then fine-tuned by training for 2 epochs (batch size of 128) to improve the test accuracy (blue line) back to the ideal pretrained model accuracy (dashed black line). (B) Confusion matrix on the MNIST task for the final network, which achieved 99.13% test accuracy.

and other non-idealities encountered [41]. Note that this method requires far less time compared to previous proposals for training new custom ONN models, which required > 25 training epochs [38, 39]. Therefore, our all-optical ReLU provides the missing link to allow ONNs to take advantage of existing pretrained models. We note that the softmax classification layer is yet to be faithfully implemented in an ONN which accounts for a small portion of the computation compared to the convolutions, matrix multiplications and ReLU nonlinear activations.

2.5 Discussion

Comparison of energy and time per activation

In this section, we compare the PPLN nanophotonic waveguide to other optical [13, 17, 19, 23], optoelectronic [11, 14–16, 21, 26–29], analog electronic [42–44], and digital electronic [45] nonlinear activation functions to demonstrate the state-of-the-art performance of our device. In this case, the appropriate figure of merit is the energy-time product, which properly accounts for both the energy consumed and time taken per activation. To quantify the energy per activation, we follow the convention in [39], as being the energy needed to generate a 50% change in the power transmission with respect to the transmission with null input. In this case, our device has an energy per activation of $\sim$ 16 fJ. The bias pulse energy is not included since it is not destroyed and can, at least theoretically, be reused for many signal pulses. This is because the bias pulse is not dissipated as heat, unlike the case often encountered for absorption-based processes. Assuming perfect phase-matching and that positive/negative values occur equally likely, then the bias pulse should be amplified/deamplified equally likely by the processes of DOPA/SHG, respectively. The time per activation is given by the signal pulse width of $\sim$ 75 fs, owing to the near-instantaneous $\chi^{(2)}$ nonlinearity of lithium niobate as explained in Section 3.2. Therefore, we achieve an energy-time product of 1.2×10^{-27} J $\cdot$ s. The energy and time per activation of our device is compared to other experimental demonstrations in Fig. 2.7. We attempted to consider device-level metrics wherever possible to provide a fair comparison, however, we acknowledge that this was not always possible for nonlinear activations as part of complete networks since fan-out and cascadability constraints impose additional energy and time costs. Despite this, the outstanding metrics of our device represents a significant breakthrough for optical nonlinear activation functions. For state-of-the-art digital electronics, such as the NVIDIA A100 Tensor Core GPU [46] based on 7-nm process node [47], we generously assume that the ReLU function consumes $\sim$ 1 fJ per activation, and occurs in a

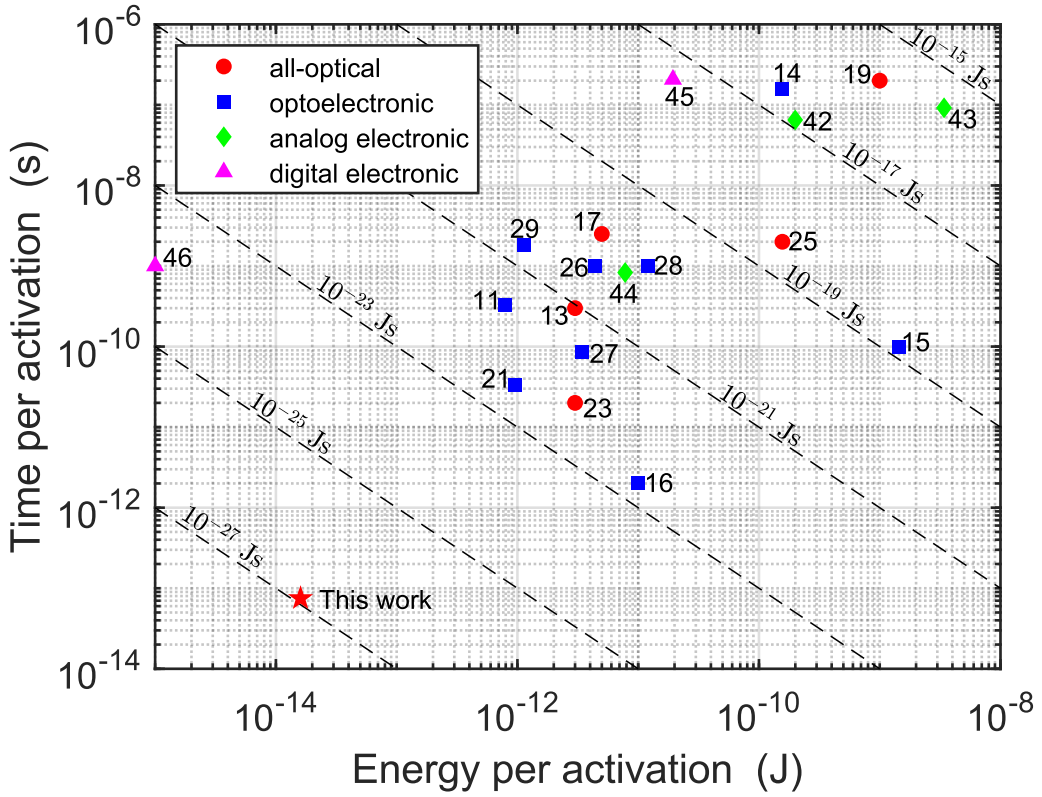

Figure 2.7: **Comparison of energy and time per activation** of this work (red star) to previous all-optical (red circle), optoelectronic (blue square), analog electronic (green diamond), and digital electronic (magenta triangle) nonlinear activation functions. The numeric labels show reference numbers and dashed black lines show the energy-time product contours.

single 1 GHz clock cycle. We see that, although our device still has an order of magnitude greater energy per activation, the time per activation is four orders of magnitude faster. Hence, we achieve an energy-time product that is three orders of magnitude better than state-of-the-art digital electronics. Our numerical simulations (Supplementary Section 3) predict that the PPLN nanophotonic waveguide can realistically achieve a ReLU-like response with sub-femtojoule energy per activation. This would even surpass the energy efficiency of state-of-the-art digital electronics. We attribute the discrepancy between our experimental results and the theoretically predicted limits for the energy scale to the imperfect phase-matching and fabrication error of our device. It is worth mentioning how these device-level metrics potentially translate to those of complete neural networks. In this case, additional system-level energy costs such as laser wall-plug efficiency and transport losses can significantly increase the effective activation energy. However, we note that the same is also true

in digital electronics such as GPUs where electrical data movement energy costs can exceed the actual switching energy by several orders of magnitude [48].

Potential network architectures

So far, we have demonstrated how PPLN nanophotonic waveguides can implement all-optical, ultrafast, energy-efficient nonlinear activation functions, which forms only one building block of a full neuron. In this section, we briefly discuss how our device can be integrated into a complete ONN architecture. Interestingly, DOPA and SHG are theoretically noiseless amplification/deamplification processes. Therefore, the all-optical ReLU function should not contribute additional noise to a photonic neural network. In principle, the all-optical ReLU is compatible with most existing ONN architectures that can accurately implement linear operations such as matrix multiplication and convolutions. However, in practice, the speed bottleneck will likely be the encoding of information into the required coherent optical amplitudes. In this case, PPLN nanophotonic waveguides can be monolithically integrated with high-speed electro-optic modulators in thin-film lithium niobate, demonstrated to achieve bandwidths beyond 100 GHz [49]. Furthermore, the light sources can also be integrated on-chip using thin-film lithium niobate optical parametric oscillators [50]. Therefore, all the fundamental building blocks needed for a complete ONN in thin-film lithium niobate already exist. Given the rapid increases in scalability of thin-film lithium niobate photonics, we are confident that a complete ONN can be demonstrated in the near-future. One potential approach is to use Mach-Zehnder interferometer meshes [18] or photonic tensor cores with waveguide cross-bar arrays [20] to implement the linear matrix multiplications, then cascaded into PPLN nanophotonic waveguides to perform nonlinear activations. Another promising method is to use a time-multiplexed architecture similar to ones demonstrated for coherent Ising machines [51] or photonic reservoir computers [14, 15]. See Supplementary Section 6 for more detailed descriptions and schematics of potential integrated lithium niobate nanophotonic neural networks for deep learning.

A valid concern is harnessing the full capabilities of the all-optical ReLU function. It is challenging to fully exploit the ultrafast time response of the nonlinear optical processes since current interfacing electronics is currently limited to GHz bandwidths [48]. However, this should not automatically preclude the use of ultrafast nonlinear optics for optical computing. For example, coherent Ising machines [51] and optical signal processing [52], which require optical input and optical output, are prime candidates for near-term applications. In the future, all-optical computing hardware

using such parametric ultrafast nonlinear activation functions may operate with THz clock rates. Crucially, the all-optical ReLU is cascadable since DOPA/SHG are inherently energy-conserving, i.e., the output is sufficiently energetic to serve as the input trigger for at least one other neuron. If multiple outputs are desired, i.e., fan-out, then intermediate amplification is needed, which can be provided by the same type of PPLNs demonstrated. Therefore, in principle, the bottleneck of optoelectronic conversion and analog-to-digital conversion can be bypassed.

2.6 Conclusion

In conclusion, we have demonstrated an all-optical ultrafast ReLU function using a PPLN nanophotonic waveguide. It has an energy per activation of $\sim$ 16 fJ and time per activation of $\sim$ 75 fs, thus achieving a state-of-the-art energy-time product of 1.2×10^{-27} J $\cdot$ s. Furthermore, we demonstrated how the same device can be used to implement other common variants of the ReLU function, and showed how it can exploit existing pretrained deep learning models to greatly reduce training time. Given the simplicity of our device, and the rapid improvements in scalability of thin-film lithium niobate photonics, we envisage that it will be able to replace periphery digital electronic circuits for calculating nonlinear activations in ONNs. Therefore, we have presented a clear and practical path towards truly all-optical, energy-efficient photonic deep learning.

2.7 Supplementary Information

Experimental setup

The experimental setup for the optical ReLU measurements is depicted in Fig. 2.8. The pump laser is a mode-locked Yb-fiber laser which provides 70 fs pulses at 1045 nm with up to 1 W average power at a 250 MHz repetition rate (Menlo Systems Orange A). The laser output is then split into two paths. The first path is sent to a synchronously pumped degenerate optical parametric (SPDOPO) oscillator based on periodically-poled lithium niobate (PPLN) which is used to efficiently generate pulses at 2090 nm [53]. The OPO is locked using a "dither and lock" scheme, facilitated by the Lock-In + PID application for Red Pitaya [54, 55]. A variable ND filter is added to the output of the OPO to control the 2090 nm power sent to the device. The second 1045 nm path is sent to a delay stage. Course adjustment of the delay is done through manual tuning of the stage position and micrometer arm while fine adjustment is performed using a piezoelectric actuator. This delay enables temporal overlap of the two paths, and fine adjustment is used to change

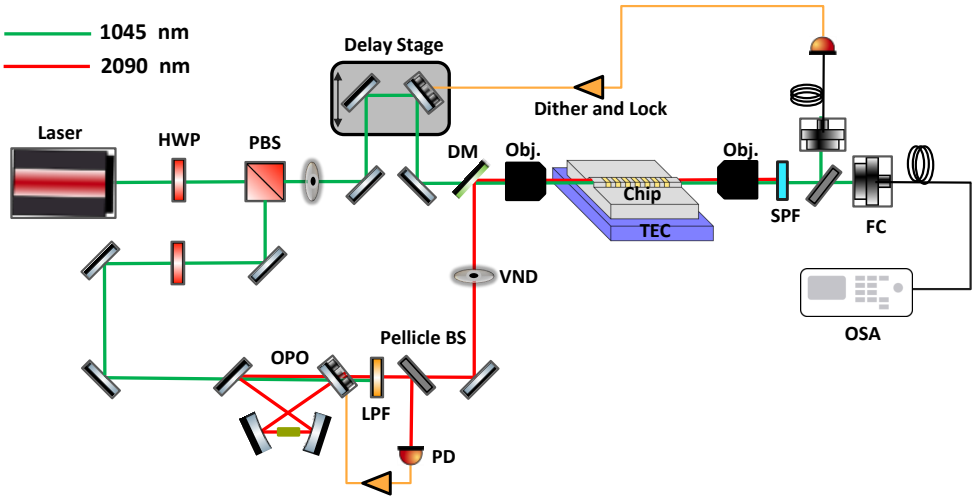

Figure 2.8: **Experimental Schematic for all-optical ReLU measurements.** The pump laser at 1045 nm is first split into two paths. One beam is used to pump our SPDOPO above threshold generating signal at centered at 2090 nm. The other beam is guided to a delay stage and further overlaps with the 2090 nm OPO signal at a dichroic mirror. Both beams are then coupled in and out from the chip using high NA reflective objectives. Next, the waveguide output is filtered with a short pass filter for filtering out the 2090 nm followed by splitting 1045 nm into two paths. Both of the 1045 nm beams are coupled into multimode fibers; one beam is measured by the OSA while the other beam is used to lock the delay stage. PBS: Polarizing beamsplitter, HWP: Half-wave plate, DM: Dichroic mirror, Obj.: Reflective objective, VND: Variable neutral-density filter, LPF: Long-pass filter, SPF: Short-pass filter, FC: Fiber Coupler, OSA: Optical spectrum analyzer, PD: Photodetector, OPO: Optical parametric oscillator.

the relative phase of the fundamental and second harmonic for the OPA process. Like the other path, a variable ND filter is also placed along this path for adjusting the 1045 nm power. The two paths are recombined at a dichroic mirror with high transmission at 1045 nm and high reflectivity at 2090 nm before going to the device.

Focusing to and coupling from the device is done using a reflective objective (Newport 50102-02). Temperature tuning of the device for fine adjustment of the quasi-phase matching condition is done using a thermoelectric cooling stage (TEC). The output of the chip is short-pass filtered around 1700 nm to remove all remaining

signal at 2090 nm and then split into two paths. The signal on one path is measured with a detector and used for feedback to the delay stage. A "dither and lock" scheme, similar to that used for the OPO, is employed here to lock the relative phases of the two inputs to switch between amplification and deamplification in the OPA process [54, 55]. The second path is coupled to fiber and sent to an optical spectrum analyzer (OSA) for measuring the output power and spectrum (Yokogawa AQ6370D).

Device fabrication and characterization

For our devices, a wafer with 700 nm of X-cut MgO-doped LN on top of 2 μm of SiO_2 was used. 15 nm of Cr underneath 55 nm of Au were then e-beam evaporated and patterned via e-beam lithography to form poling electrodes. 300V pulses were used to pole the chip, and the quality was confirmed using second harmonic microscopy. Waveguides were subsequently patterned on the chip using hydrogen silsesquioxane (HSQ) as the e-beam resist and 15 nm of Ti as an adhesion layer. They were dry etched with Ar^+ plasma in an inductively-coupled plasma reactive-ion etcher (ICP-RIE), and the remaining resists and side-wall re-deposition were removed using Buffered oxide etchant (BOE) and RCA-1. Finally, the waveguide facets were mechanically polished.

In Fig. 2.9, we display the measured spectrums of 2ω, ω and their non-linear interaction in the waveguide. (a) and (b) show the input spectrums of 2ω and ω, respectively, and (c) shows the evolution of 2ω pulse as the phase difference is modulated. We can see that for the positive signal values corresponding to the phase relationship $2\phi_\omega - \phi_{2\omega} = \pi/2$, the 2ω signal grows due to SHG process while depleting the ω pulse. On the other hand, for the phase relationship $2\phi_\omega - \phi_{2\omega} = -\pi/2$, the ω pulse grows due to optical parametric amplification, thereby depleting the 2ω as evident from the dip in the spectrum.

We now estimate the input and output coupling efficiencies of our device. A detailed discussion is provided in previous work on optical parametric generation (OPG) and amplification (OPA) [33, 56]; here we outline the main steps. For a degenerate OPG process in the high parametric gain regime, the generated average photon-number in an ideal case is given by

$$\langle N \rangle \approx \frac{1}{4} e^{2L\sqrt{\eta P}}, \tag{2.1}$$

where P, L and η are pump power, interaction length, and non-linear interaction efficiency. In the presence of experimental imperfections such as off-chip coupling, coupling to optical fibers, and detection inefficiencies, the average photon-number

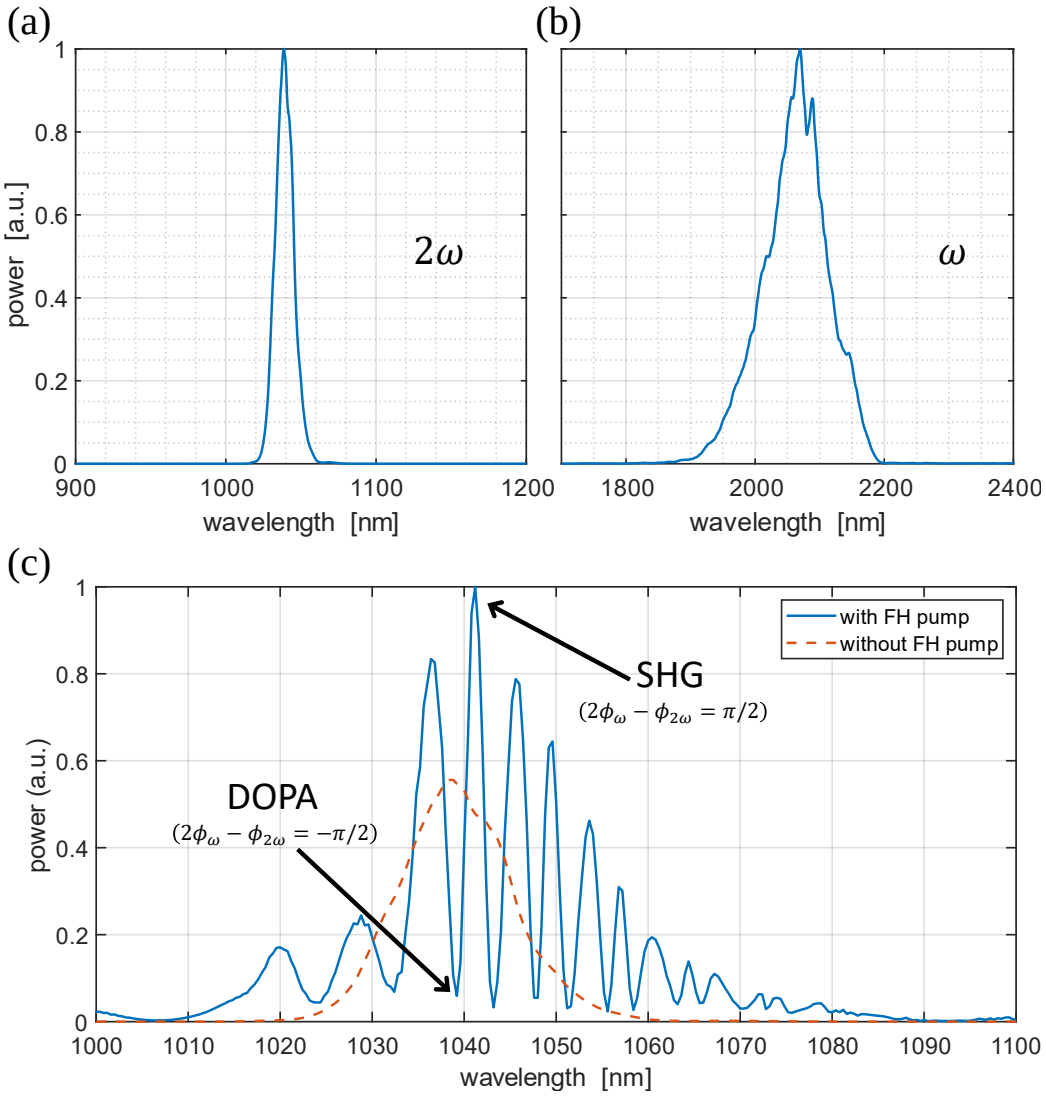

Figure 2.9: **Measured spectrums of ω and 2ω.** In (a) and (b) correspond to the waveguide input 2ω and ω, respectively. (c) shows the evolution of the waveguide output 2ω as the phase difference between 2ω and ω is modulated.

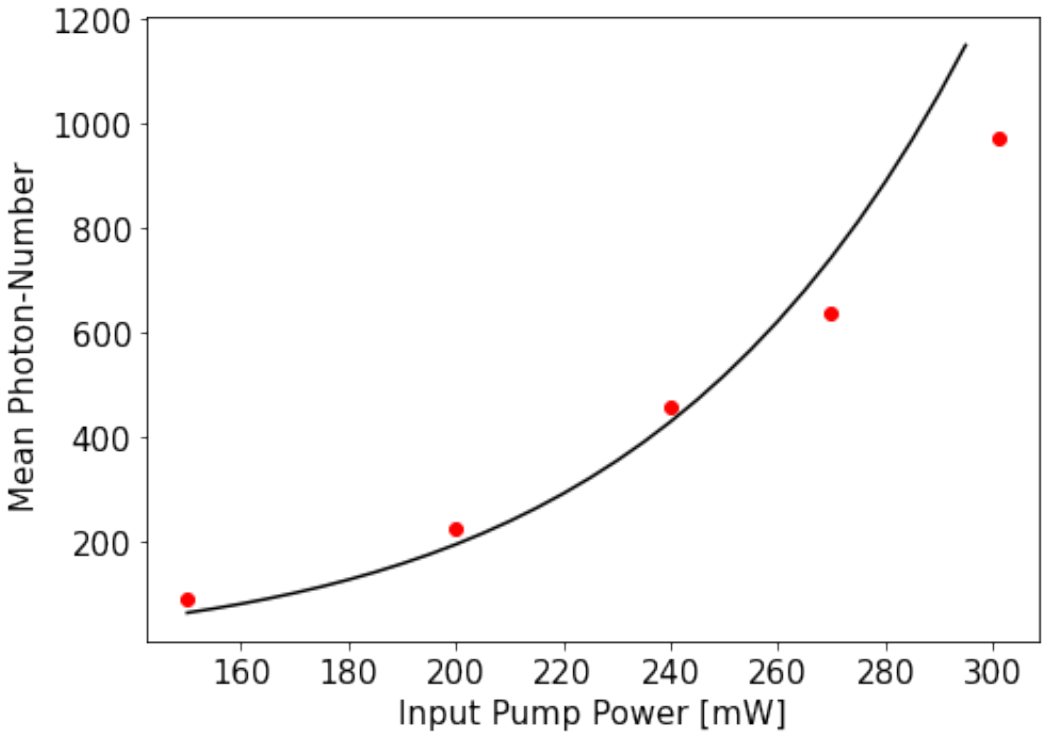

Figure 2.10: **Number of signal photons as the input pump power is varied.** The red points are experimentally measured data for several values of pump power and the black curve shows the exponential fit used for estimating the output coupling efficiency, i.e., the η_1 parameter in Eq. 2.2.

is given as

$$\langle N \rangle \approx \frac{\eta_1}{4} e^{2L\sqrt{\eta_2 P}}, \tag{2.2}$$

where all optical losses on the OPG signal are combined in η_1 parameter and η_2 quantifies the non-linear interaction strength and the input coupling efficiency of our second harmonic signal. From our measured data for OPG power, we determine the average photon number for various values of the second harmonic pump. By fitting the data, we can extract the η_1 and η_2 parameters. In Fig. 2.10, the measured average number of photons are displayed with respect to the input pump power. From the fit, we extract $\eta_1 \approx 0.20$, i.e., the estimated output coupling loss is about 7 dB, which shows a good agreement with our previous paper [33, 56]. Given the total coupling loss measured at low power, we then determine the input coupling loss as the difference between total and output coupling losses. We note that coupling losses < 1 dB per facet have been reported for thin-film lithium niobate photonics [57], which is promising for large-scale circuits.

Simulation method

We numerically solved an analytical nonlinear envelope equation (NEE) in the frequency domain using a split-step Fourier technique to simulate the pulse propagation and nonlinear dynamics in the waveguide. The nonlinear step was implemented using fourth-order Runge-Kutta method. We obtained the NEE by ignoring counter-propagating modes, which are usually phase mismatched, and assuming a constant nonlinear coefficient across the entire simulation bandwidth. The fundamental and second harmonic pulses were assumed to have a transform-limited, hyperbolic-secant profile. The NEE is given by:

$$\frac{\partial A}{\partial z} = -i\left[\beta(\omega) - \beta_0 - \frac{\Omega}{v_{ref}} - i\frac{\alpha}{2}\right]A \tag{2.3}$$
$$-\frac{i\omega\varepsilon_0 X_0}{8} d(z)\mathcal{F}_\Omega\left\{a^2(z,t)e^{j\phi(z,t)} + 2a(z,t)a^*(z,t)e^{-j\phi(z,t)}\right\}$$

where $A(z,\omega)$ is the complex amplitude of the field during propagation, $a(z,t)$ is the time domain representation of $A(z,\Omega)$, $\phi(z,t) = \omega_0 t - (\beta_0 - \omega_0/v_{ref})z$, β_0 is the waveguide propagation constant at frequency ω_0, $\Omega = \omega - \omega_0$ is the envelope frequency, ω is the optical frequency, α is the attenuation constant, $d(z) = \pm 1$ is the instantaneous sign of the nonlinear coefficient due to quasi-phase matching, $\mathcal{F}_\Omega$ is the Fourier transform in Ω, and X_0 is the effective nonlinear coefficient.

To simulate the ReLU response of our experimental device, shown in Fig. 3 in the main text, we assumed $\alpha \approx 0.1$ dB/cm and used the following waveguide geometry obtained from atomic force microscope measurements: waveguide top width of 1768 nm, ridge height (etch depth) of 377 nm, and a total lithium niobate thin-film thickness (before etching) of 713 nm. We use the effective nonlinear coefficient as a fitting parameter to match the experimental data and inferred a value of $X_0 \approx 0.36 \times 10^{-12}$ V^2, which is about $\sim 1/3$ of its ideal value.

Given the fabrication error and imperfect phase-matching of our device, we could only experimentally achieve an energy of $\sim$ 16 fJ per activation. However, Fig. 2.11 shows the simulated ideal performance of a PPLN with length $L = 2.5$ mm, ridge top width of $w = 1700$ nm, etch-depth of $h = 350$ nm, and bias pulse energy of $E_\omega(0) = 10$ fJ. We see that it can achieve a ReLU-like function with sub-femtojoule energy per activation.

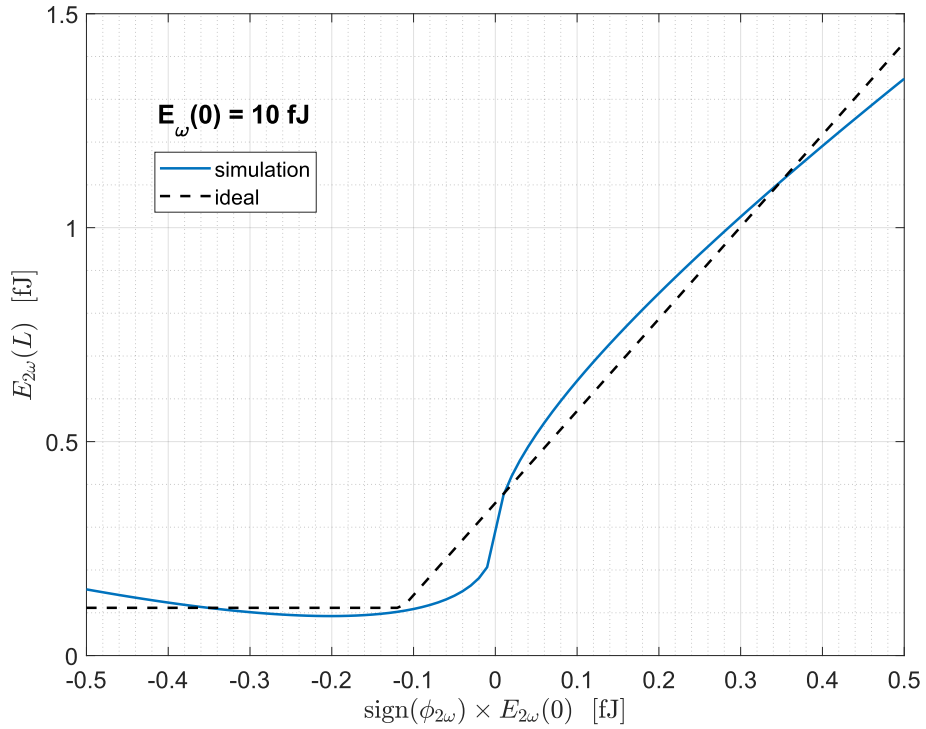

Figure 2.11: **Simulated ReLU-like nonlinear activation function with sub-femtojoule energies** achieved using bias pulse energy of $E_\omega(0) = 10$ fJ and ideal PPLN parameters.

Fitting of pump-probe signal

The input autocorrelation was fit using a Gaussian profile:

$$G(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-t^2}{2\sigma^2}}, \tag{2.4}$$

where σ is related to the FWHM by $FWHM = 2\sigma\sqrt{2\ln 2}$. The exponential function with characteristic decay time of $\tau = 1/\lambda$ is defined as:

$$F(t) = e^{-\lambda t}. \tag{2.5}$$

The convolution between $G(t)$ and $F(t)$ is defined as:

$$I(t') = F(t) * G(t) = \frac{1}{\sigma\sqrt{2\pi}} \int_0^\infty e^{-\lambda t} e^{-\frac{(t'-t)^2}{2\sigma^2}} dt. \tag{2.6}$$

We fit the pump-probe signal with exponential growth and decay functions for positive and negative time delays, respectively, convolved with the input autocorrelation

by using the analytical formula for Eq. 2.6:

$$I(t') = \frac{1}{2} e^{-\lambda\left(t'-\sigma^2\lambda/2\right)} \left[1 + \mathrm{erf}\left(\frac{t' - \sigma^2\lambda}{\sqrt{2}\sigma}\right)\right] , \tag{2.7}$$

where $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-z^2}\, dz$ is the error function.

Convolutional neural network architecture

The pretrained convolutional neural network (CNN) architecture is shown in Fig. 2.12. The CNN was trained on the MNIST handwritten digits image classification [40] using stochastic gradient descent with momentum (SGDM) with initial learn rate of 0.01 and batch size of 128. For fine-tuning after the ideal ReLU layers were replaced with custom layers representing the experimentally measured ReLU response, the initial learn rate was decreased to 0.001.

	Name	Type	Activations	Learnables
1	imageinput 28×28×1 images with 'zerocenter' normalization	Image Input	28×28×1	-
2	conv_1 8 3×3×1 convolutions with stride [1 1] and padding 'same'	Convolution	28×28×8	Weights 3×3×1×8 Bias 1×1×8
3	batchnorm_1 Batch normalization with 8 channels	Batch Normalization	28×28×8	Offset 1×1×8 Scale 1×1×8
4	relu_1 ReLU	ReLU	28×28×8	-
5	maxpool_1 2×2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	14×14×8	-
6	conv_2 16 3×3×8 convolutions with stride [1 1] and padding 'same'	Convolution	14×14×16	Weights 3×3×8×16 Bias 1×1×16
7	batchnorm_2 Batch normalization with 16 channels	Batch Normalization	14×14×16	Offset 1×1×16 Scale 1×1×16
8	relu_2 ReLU	ReLU	14×14×16	-
9	maxpool_2 2×2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	7×7×16	-
10	conv_3 32 3×3×16 convolutions with stride [1 1] and padding 'same'	Convolution	7×7×32	Weights 3×3×16×32 Bias 1×1×32
11	batchnorm_3 Batch normalization with 32 channels	Batch Normalization	7×7×32	Offset 1×1×32 Scale 1×1×32
12	relu_3 ReLU	ReLU	7×7×32	-
13	fc 10 fully connected layer	Fully Connected	1×1×10	Weights 10×1568 Bias 10×1
14	softmax softmax	Softmax	1×1×10	-
15	classoutput crossentropyex with '0' and 9 other classes	Classification Output	-	-

Figure 2.12: **Pretrained convolutional neural network architecture.**

Potential integrated photonic neural networks

A promising approach to integrating the all-optical ultrafast ReLU into a complete ONN is to monolithically integrate it with high-speed electro-optic modulators in thin-film lithium niobate nanophotonic circuits. Fig. 2.13 shows two examples of

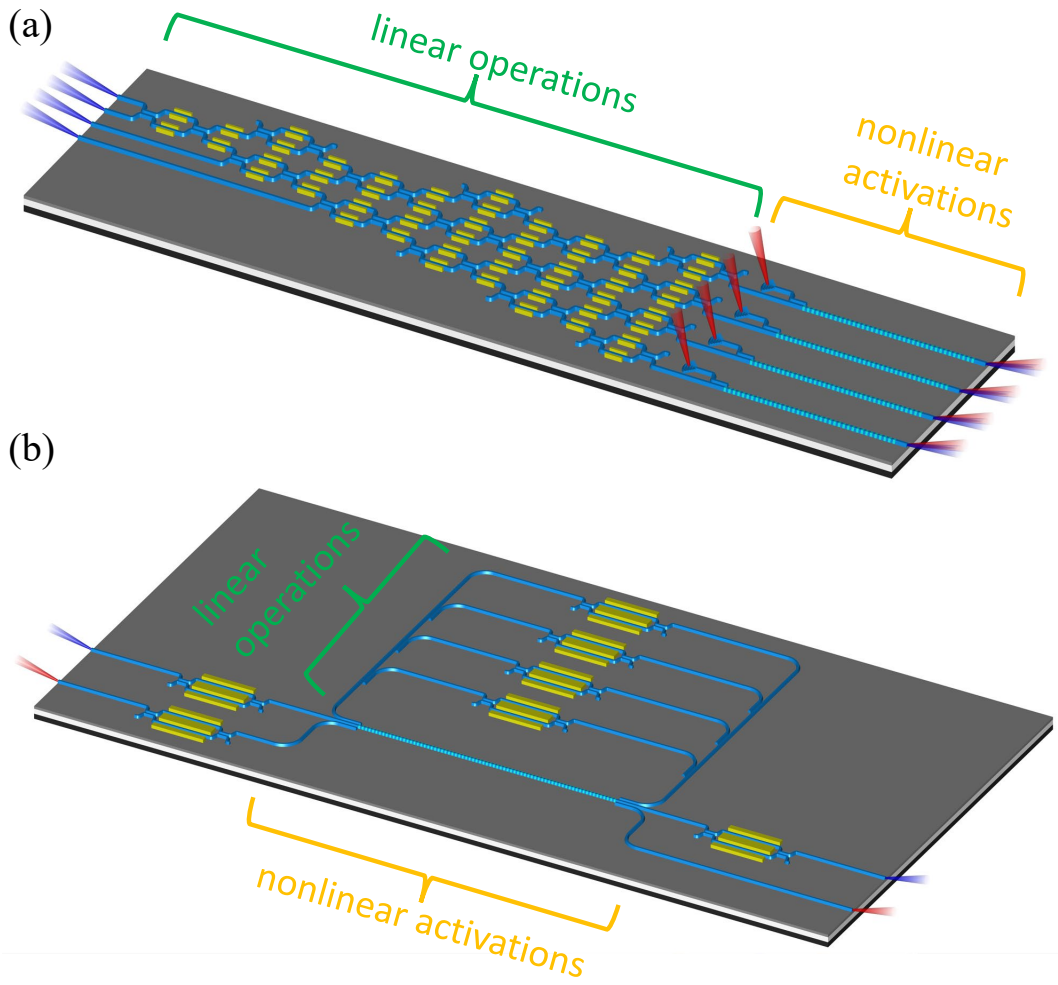

Figure 2.13: **Potential integrated photonic neural networks using the all-optical ultrafast ReLU function.** (a) Spatially-multiplexed design based on a mesh of Mach-Zehnder interferometers performing linear operations, directly cascaded into an array of PPLNs performing the ReLU activations. (b) Time-multiplexed design based on feedback-modulated delay loops performing linear operations and the PPLN performing ReLU activations, acting as the single photonic neuron folded in time.

how this can be accomplished. One method, shown in Fig. 2.13(a), uses a spatially-multiplexed approach. It consists of a mesh of Mach-Zehnder inteferometers, akin to those demonstrated in silicon photonics [18], to perform linear operations, directly followed by an array of PPLNs to perform the ReLU activations. Therefore, in this approach, each neuron represents a separate PPLN and the entire neural network layer is computed in a constant time step. Subsequent layers are identical in structure and can be directly cascaded following the array of PPLNs. The bias pulse can be

directly fed to each PPLN using out-of-plane couplers as shown in Fig. 2.13(a), or by using in-plane photonic crossbar switches. The bias and signal pulses can be decoupled using wavelength-division multiplexing (WDM) filters either on-chip or off-chip.

The second method, shown in Fig. 2.13(b), uses a time-multiplexed approach based on a single photonic neuron folded in time with feedback-modulated delay loops [58]. In this architecture, each delay loop at each time step represents a different synaptic connection in the neural network layer. By properly updating the feedback modulators at each time step, the required linear operations can be achieved. Therefore, only one PPLN performing ReLU activations is needed to represent all neurons, but the number of delay loops and time steps to compute each neural network layer equals the number of synapses for each neuron. This architecture may be advantageous in that it relaxes the experimental constraints for fabricating and controlling a large number of PPLNs like in the spatially-multiplexed method.

Given the relatively long (~mm) length of the PPLN, but ultrafast response time, it is desirable to employ a time-multiplexed approach for scalability. Furthermore, although we show the use of free-space coupling here, this can be eliminated through the monolithic integration of thin-film lithium niobate lasers [59], and integrated detectors [60]. This is substantially more complicated than any previously demonstrated thin-film lithium niobate photonic circuit [32, 33, 61, 62], but rapid improvements in fabrication quality/tolerance in thin-film lithium niobate photonics promise a path toward a monolithically integrated photonic neural network in the near future.

References

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[2] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE*, 105(12):2295–2329, 2017.

[3] Yann LeCun. Deep learning hardware: Past, present, and future. In *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*, pages 12–19. IEEE, 2019.

[4] Gordon Wetzstein, Aydogan Ozcan, Sylvain Gigan, Shanhui Fan, Dirk Englund, Marin Soljačić, Cornelia Denz, David A.B. Miller, and Demetri Psaltis.

Inference in artificial intelligence with deep optics and photonics. *Nature*, 588 (7836):39–47, 2020.

[5] Xing Lin, Yair Rivenson, Nezih T. Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi, and Aydogan Ozcan. All-optical machine learning using diffractive deep neural networks. *Science*, 361(6406):1004–1008, 2018.

[6] Tiankuang Zhou, Xing Lin, Jiamin Wu, Yitong Chen, Hao Xie, Yipeng Li, Jingtao Fan, Huaqiang Wu, Lu Fang, and Qionghai Dai. Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit. *Nature Photonics*, 15(5):367–373, 2021.

[7] Ying Zuo, Bohan Li, Yujun Zhao, Yue Jiang, You-Chiuan Chen, Peng Chen, Gyu-Boong Jo, Junwei Liu, and Shengwang Du. All-optical neural network with nonlinear activation functions. *Optica*, 6(9):1132–1137, 2019.

[8] Tianyu Wang, Shi-Yuan Ma, Logan G. Wright, Tatsuhiro Onodera, Brian Richard, and Peter L McMahon. An optical neural network using less than 1 photon per multiplication. *arXiv preprint arXiv:2104.13467*, 2021.

[9] Ziyu Gu, Yesheng Gao, and Xingzhao Liu. Optronic convolutional neural networks of multi-layers with different functions executed in optics for image classification. *Optics Express*, 29(4):5877–5889, 2021.

[10] Mario Miscuglio, Zibo Hu, Shurui Li, Jonathan K. George, Roberto Capanna, Hamed Dalir, Philippe M. Bardet, Puneet Gupta, and Volker J. Sorger. Massively parallel amplitude-only fourier neural network. *Optica*, 7(12):1812–1819, 2020.

[11] Xavier Porte, Anas Skalli, Nasibeh Haghighi, Stephan Reitzenstein, James A. Lott, and Daniel Brunner. A complete, parallel and autonomous photonic neural network in a semiconductor multimode laser. *Journal of Physics: Photonics*, 3(2):024017, 2021.

[12] Xingyuan Xu, Mengxi Tan, Bill Corcoran, Jiayang Wu, Andreas Boes, Thach G. Nguyen, Sai T. Chu, Brent E. Little, Damien G. Hicks, Roberto Morandotti, et al. 11 tops photonic convolutional accelerator for optical neural networks. *Nature*, 589(7840):44–51, 2021.

[13] George Mourgias-Alexandris, Apostolos Tsakyridis, Nikolaos Passalis, Anastasios Tefas, Konstantinos Vyrsokinos, and Nikolaos Pleros. An all-optical neuron with sigmoid activation function. *Optics Express*, 27(7):9620–9630, 2019.

[14] François Duport, Bendix Schneider, Anteo Smerieri, Marc Haelterman, and Serge Massar. All-optical reservoir computing. *Optics Express*, 20(20):22783–22795, 2012.

[15] François Duport, Anteo Smerieri, Akram Akrout, Marc Haelterman, and Serge Massar. Fully analogue photonic reservoir computer. *Scientific Reports*, 6(1): 1–12, 2016.

[16] Bhavin J. Shastri, Mitchell A. Nahmias, Alexander N. Tait, Alejandro W. Rodriguez, Ben Wu, and Paul R. Prucnal. Spike processing with a graphene excitable laser. *Scientific Reports*, 6(1):1–12, 2016.

[17] Antoine Dejonckheere, François Duport, Anteo Smerieri, Li Fang, Jean-Louis Oudar, Marc Haelterman, and Serge Massar. All-optical reservoir computer based on saturation of absorption. *Optics Express*, 22(9):10868–10881, 2014.

[18] Yichen Shen, Nicholas C. Harris, Scott Skirlo, Mihika Prabhu, Tom Baehr-Jones, Michael Hochberg, Xin Sun, Shijie Zhao, Hugo Larochelle, Dirk Englund, et al. Deep learning with coherent nanophotonic circuits. *Nature Photonics*, 11(7):441–446, 2017.

[19] Johannes Feldmann, Nathan Youngblood, C. David Wright, Harish Bhaskaran, and Wolfram H.P. Pernice. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature*, 569(7755):208–214, 2019.

[20] Johannes Feldmann, Nathan Youngblood, Maxim Karpov, Helge Gehring, Xuan Li, Maik Stappers, Manuel Le Gallo, Xin Fu, Anton Lukashchuk, Arslan Sajid Raja, et al. Parallel convolutional processing using an integrated photonic tensor core. *Nature*, 589(7840):52–58, 2021.

[21] Farshid Ashtiani, Alexander J. Geers, and Firooz Aflatouni. Single-chip photonic deep neural network for instantaneous image classification. *arXiv preprint arXiv:2106.11747*, 2021.

[22] Shaofu Xu, Jing Wang, Haowen Shu, Zhike Zhang, Sicheng Yi, Bowen Bai, Xingjun Wang, Jianguo Liu, and Weiwen Zou. Optical coherent dot-product chip for sophisticated deep learning regression. *arXiv preprint arXiv:2105.12122*, 2021.

[23] Bin Shi, Nicola Calabretta, and Ripalta Stabile. Inp photonic integrated multi-layer neural networks: Architecture and performance analysis. *APL Photonics*, 2021.

[24] Mario Miscuglio, Armin Mehrabian, Zibo Hu, Shaimaa I. Azzam, Jonathan George, Alexander V. Kildishev, Matthew Pelton, and Volker J. Sorger. All-optical nonlinear activation function for photonic neural networks. *Optical Materials Express*, 8(12):3851–3863, 2018.

[25] Aashu Jha, Chaoran Huang, and Paul R. Prucnal. Reconfigurable all-optical nonlinear activation functions for neuromorphic photonics. *Optics Letters*, 45 (17):4819–4822, 2020.

[26] Alexander N. Tait, Thomas Ferreira De Lima, Ellen Zhou, Allie X. Wu, Mitchell A. Nahmias, Bhavin J. Shastri, and Paul R. Prucnal. Neuromorphic photonic networks using silicon photonic weight banks. *Scientific Reports*, 7 (1):1–10, 2017.

[27] Jasna Crnjanski, Marko Krstić, Angelina Totović, Nikos Pleros, and Dejan Gvozdić. Adaptive sigmoid-like and prelu activation functions for all-optical perceptron. *Optics Letters*, 46(9):2003–2006, 2021.

[28] Rubab Amin, Jonathan K. George, Shuai Sun, Thomas Ferreira de Lima, Alexander N. Tait, Jacob B. Khurgin, Mario Miscuglio, Bhavin J. Shastri, Paul R. Prucnal, Tarek El-Ghazawi, et al. ITO-based electro-absorption modulator for photonic neural activation function. *APL Materials*, 7(8):081112, 2019.

[29] Charis Mesaritakis, Alexandros Kapsalis, Adonis Bogris, and Dimitris Syvridis. Artificial neuron based on integrated semiconductor quantum dot mode-locked lasers. *Scientific Reports*, 6(1):1–10, 2016.

[30] Cheng Wang, Carsten Langrock, Alireza Marandi, Marc Jankowski, Mian Zhang, Boris Desiatov, Martin M. Fejer, and Marko Lončar. Ultrahigh-efficiency wavelength conversion in nanophotonic periodically poled lithium niobate waveguides. *Optica*, 5(11):1438–1441, 2018.

[31] Marc Jankowski, Carsten Langrock, Boris Desiatov, Alireza Marandi, Cheng Wang, Mian Zhang, Christopher R Phillips, Marko Lončar, and Martin M. Fejer. Ultrabroadband nonlinear optics in nanophotonic periodically poled lithium niobate waveguides. *Optica*, 7(1):40–46, 2020.

[32] Qiushi Guo, Ryoto Sekine, Luis Ledezma, Rajveer Nehra, Devin J. Dean, Arkadev Roy, Robert M Gray, Saman Jahani, and Alireza Marandi. Femtojoule, femtosecond all-optical switching in lithium niobate nanophotonics. *arXiv preprint arXiv:2107.09906*, 2021.

[33] Luis Ledezma, Ryoto Sekine, Qiushi Guo, Rajveer Nehra, Saman Jahani, and Alireza Marandi. Intense optical parametric amplification in dispersion engineered nanophotonic lithium niobate waveguides. *arXiv preprint arXiv:2104.08262*, 2021.

[34] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[35] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[36] Masaaki Ono, Masanori Hata, Masato Tsunekawa, Kengo Nozaki, Hisashi Sumikura, Hisashi Chiba, and Masaya Notomi. Ultrafast and energy-efficient all-optical switching with graphene-loaded deep-subwavelength plasmonic waveguides. *Nature Photonics*, 14(1):37–43, 2020.

[37] Gustavo Grinblat, Michael P. Nielsen, Paul Dichtl, Yi Li, Rupert F. Oulton, and Stefan A. Maier. Ultrafast sub–30-fs all-optical switching based on gallium phosphide. *Science Advances*, 5(6):eaaw3262, 2019.

[38] Xianxin Guo, Thomas D. Barrett, Zhiming M Wang, and Alexander I. Lvovsky. Backpropagation through nonlinear units for the all-optical training of neural networks. *Photonics Research*, 9(3):B71–B80, 2021.

[39] Ian A.D. Williamson, Tyler W. Hughes, Momchil Minkov, Ben Bartlett, Sunil Pai, and Shanhui Fan. Reprogrammable electro-optic nonlinear activation functions for optical neural networks. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(1):1–12, 2019.

[40] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[41] Saumil Bandyopadhyay, Ryan Hamerly, and Dirk Englund. Hardware error correction for programmable photonics. *Optica*, 8(10):1247–1255, Oct 2021.

[42] Sangheon Oh, Yuhan Shi, Javier Del Valle, Pavel Salev, Yichen Lu, Zhisheng Huang, Yoav Kalcheim, Ivan K. Schuller, and Duygu Kuzum. Energy-efficient mott activation neuron for full-hardware implementation of neural networks. *Nature Nanotechnology*, 16(6):680–687, 2021.

[43] Olga Krestinskaya, Khaled Nabil Salama, and Alex Pappachen James. Learning in memristive neural network architectures using analog backpropagation circuits. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 66(2): 719–732, 2018.

[44] Yucong Huang, Zhitao Yang, Jianghan Zhu, and Terry Tao Ye. Analog circuit implementation of neurons with multiply-accumulate and relu functions. In *Proceedings of the 2020 on Great Lakes Symposium on VLSI*, pages 493–498, 2020.

[45] Massimo Giordano, Giorgio Cristiano, Koji Ishibashi, Stefano Ambrogio, Hsinyu Tsai, Geoffrey W Burr, and Pritish Narayanan. Analog-to-digital conversion with reconfigurable function mapping for neural networks activation function acceleration. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 9(2):367–376, 2019.

[46] Jack Choquette, Wishwesh Gandhi, Olivier Giroux, Nick Stam, and Ronny Krashinsky. Nvidia a100 tensor core gpu: Performance and innovation. *IEEE Micro*, 41(2):29–35, 2021.

[47] Qing Xie, Xue Lin, Yanzhi Wang, Shuang Chen, Mohammad Javad Dousti, and Massoud Pedram. Performance comparisons between 7-nm finfet and conventional bulk cmos standard cell libraries. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 62(8):761–765, 2015.

[48] Chris Cole. Optical and electrical programmable computing energy use comparison. *Optics Express*, 29(9):13153–13170, 2021.

[49] Mian Zhang, Cheng Wang, Prashanta Kharel, Di Zhu, and Marko Lončar. Integrated lithium niobate electro-optic modulators: When performance meets scalability. *Optica*, 8(5):652–667, 2021.

[50] Juanjuan Lu, Ayed Al Sayem, Zheng Gong, Joshua B Surya, Chang-Ling Zou, and Hong X Tang. Ultralow-threshold thin-film lithium niobate optical parametric oscillator. *Optica*, 8(4):539–544, 2021.

[51] Yoshihisa Yamamoto, Kazuyuki Aihara, Timothee Leleu, Ken-ichi Kawarabayashi, Satoshi Kako, Martin M. Fejer, Kyo Inoue, and Hiroki Takesue. Coherent ising machines—optical neural networks operating at the quantum limit. *npj Quantum Information*, 3(1):1–15, 2017.

[52] Stefan Wabnitz and Benjamin J. Eggleton. All-optical signal processing. *Springer Series in Optical Sciences*, 194, 2015.

[53] Marc Jankowski, Alireza Marandi, Christopher R. Phillips, Ryan Hamerly, Kirk A. Ingold, Robert L. Byer, and Martin M. Fejer. Temporal simultons in optical parametric oscillators. *Physical Review Letters*, 120:053904, Feb 2018.

[54] Alireza Marandi, Nick C. Leindecker, Vladimir Pervak, Robert L. Byer, and Konstantin L. Vodopyanov. Coherence properties of a broadband femtosecond mid-ir optical parametric oscillator operating at degeneracy. *Optics Express*, 20:7255–7262, 2012.

[55] Marcelo Alejandro Luda, Martin Drechsler, Christian Tom'as Schmiegelow, and Jorge Codnia. Compact embedded device for lock-in measurements and experiment active control. *Review of Scientific Instruments*, 90:023106, 2019.

[56] Luis Ledezma, Ryoto Sekine, Qiushi Guo, Rajveer Nehra, Saman Jahani, and Alireza Marandi. 100 db/cm broadband optical parametric amplification in dispersion engineered nanophotonic lithium niobate waveguides. In *CLEO: Science and Innovations*, pages SF1C–7. Optical Society of America, 2021.

[57] Changran Hu, An Pan, Tingan Li, Xuanhao Wang, Yuheng Liu, Shiqi Tao, Cheng Zeng, and Jinsong Xia. High-efficient coupler for thin-film lithium niobate waveguide devices. *Optics Express*, 29(4):5397–5406, 2021.

[58] Florian Stelzer, André Röhm, Raul Vicente, Ingo Fischer, and Serhiy Yanchuk. Deep neural networks using a single neuron: Folded-in-time architecture using feedback-modulated delay loops. *Nature Communications*, 12(1):1–10, 2021.

[59] Xiangmin Liu, Xiongshuo Yan, Hao Li, Yuping Chen, Xianfeng Chen, et al. Tunable single-mode laser on thin film lithium niobate. *Optics Letters*, 46(21): 5505–5508, 2021.

[60] Ayed Al Sayem, Risheng Cheng, Sihao Wang, and Hong X Tang. Lithium-niobate-on-insulator waveguide-integrated superconducting nanowire single-photon detectors. *Applied Physics Letters*, 116(15):151102, 2020.

[61] Luis Ledezma, Arkadev Roy, Luis Costa, Ryoto Sekine, Robert Gray, Qiushi Guo, Ryan M. Briggs, and Alireza Marandi. Widely-tunable optical parametric oscillator in lithium niobate nanophotonics. *arXiv preprint arXiv:2203.11482*, 2022.

[62] Rajveer Nehra, Ryoto Sekine, Luis Ledezma, Qiushi Guo, Robert M Gray, Arkadev Roy, and Alireza Marandi. Few-cycle vacuum squeezing in nanophotonics. *arXiv preprint arXiv:2201.06768*, 2022.

Chapter 3

ULTRAFAST NEUROMORPHIC COMPUTING WITH NANOPHOTONIC OPTICAL PARAMETRIC OSCILLATORS

Midya Parto*, **Gordon H.Y. Li***, Ryoto Sekine, Robert M. Gray, Luis L. Ledezma, James Williams, Arkadev Roy, and Alireza Marandi. Ultrafast neuromorphic computing with nanophotonic optical parametric oscillators. *arXiv preprint arXiv:2501.16604*, 2025. doi:10.48550/arXiv.2501.16604.
G.H.Y.L. conceived the project, participated in developing the theory, executed initial numerical simulations, participated in the design of the experiments, and participated in the writing of the manuscript.
* denotes equal contributions.

3.1 Abstract

Over the past decade, artificial intelligence (AI) has led to disruptive advancements in fundamental sciences and everyday technologies. Among various machine learning algorithms, deep neural networks [1] have become instrumental in revealing complex patterns in large datasets with key applications in computer vision, natural language processing, and predictive analytics. With the increasing prevalence and adoption of deep learning, the quest for hardware solutions that can efficiently process data in real time with high speeds and low latencies has come to the forefront of research in many fields. On-chip photonic neural networks offer a promising platform that leverage high bandwidths and low propagation losses associated with optical signals to perform analog computations for deep learning [2–6]. However, nanophotonic circuits are yet to achieve the required linear and nonlinear operations simultaneously in an all-optical and ultrafast fashion. Here, we report an ultrafast nanophotonic neuromorphic processor using an optical parametric oscillator (OPO) fabricated on thin-film lithium niobate (TFLN). The input data is used to modulate the optical pulses synchronously pumping the OPO. The consequent signal pulses generated by the OPO are coupled to one another via the nonlinear delayed dynamics of the OPO, thus forming the internal nodes of a deep recurrent neural network. We use such a nonlinearly coupled OPO network for chaotic time series prediction, nonlinear error correction in a noisy communication channel, as well as noisy waveform classification and achieve accuracies exceeding 93% at an operating clock rate of $\sim$ 10 GHz. Our OPO network is capable of achieving sub-nanosecond

latencies, a timescale comparable to a single clock cycle in state-of-the-art digital electronic processors. By circumventing the need for optical-electronic-optical (OEO) conversions, our ultrafast nanophotonic neural network paves the way for the next generation of compact all-optical neuromorphic processors with ultralow latencies and high energy efficiencies.

3.2 Introduction

Deep neural networks (DNNs) have revolutionized modern data processing with transformative results across numerous fields ranging from fundamental sciences to automotive and healthcare industries to generative art [7–9]. As deep learning algorithms become more prominent, traditional computing hardware proves to be increasingly less suitable for their implementation. This incompatibility poses a formidable challenge for the future of AI and originates from several characteristics of current computing architectures. The substantial growth of the compute required by recent deep learning models far surpasses the current trend in the advancement of digital electronic processors [10]. The resulting gap becomes even more acute in many practical scenarios that require real-time processing where ultralow latencies are critical. In addition, as algorithms become more complex and datasets grow larger, the energy required to perform computations surges, resulting in serious financial and environmental consequences. Providing viable solutions to these challenges has brought the quest for efficient and high-speed computing hardware to the forefront of research across various disciplines to drive disruptive AI technologies.

Against this backdrop, photonic neural networks (PNNs) have recently emerged as promising candidates that can harness high bandwidths offered by optical systems to provide ultrafast operation while maintaining high energy efficiencies facilitated by the low-loss propagation of light [5, 6, 11]. In particular, recent advances in nanofabrication and photonic integrated circuits (PICs) have provided a fertile ground for implementing light-based AI architectures on compact chip-scale devices [12]. DNNs consist of many layers of linear operations such as matrix multiplications or convolutions interleaved by nonlinear activation functions. Existing integrated PNNs utilize various architectures such as Mach-Zehnder inteferometers [13] as well as microring resonator arrays to accelerate the linear operations necessary for deep learning [2, 4, 14]. Typically, opto-electronic components provide the nonlinear activation functions in PNNs. Alternatively, one can realize the nonlinear operations by using all-optical nonlinearities [15] such as phase change materials [16–18] or optical parametric processes [19, 20]. This latter approach can therefore eliminate

the need for successive optical-electronic-optical (OEO) conversions and provide a route for true all-optical integrated neural networks with ultrashort latencies beyond those attainable by hybrid optoelectronic architectures. Yet, despite significant progress in PNNs [21], achieving a unifying platform that simultaneously provides both linear multiply-accumulate (MAC) operations and ultrafast nonlinear activation functions in nanophotonic circuits has remained elusive.

In this work, we address this challenge and demonstrate an integrated photonic neural network that harnesses coherent optical pulse propagation in conjunction with ultrafast parametric nonlinear processes [22, 23] to achieve ultralow-latency operations. To this end, we use a nanophotonic optical parametric oscillator (OPO) fabricated on a TFLN chip to implement a deep recurrent neural network which we dub OPONN. In our scheme, as shown in Fig. 3.1, masked sequential data is used to modulate the optical pulses of an electro-optic (EO) frequency comb used to synchronously pump the OPO. In response, the nanophotonic OPO, which operates at degeneracy [24], generates signal pulses at the half-harmonic of the input pump. The optical feedback provided by the cavity on one hand and the parametric amplification on the other hand act upon the generated signal fields creating dynamics akin to those associated with neurons in a recurrent neural network. We utilize our OPONN for machine learning on three different tasks: (i) ultrafast chaotic time series prediction involving the archetypal Lorenz and Mackey-Glass systems of equations, (ii) compensating nonlinear distortions in a noisy communication channel, and (iii) waveform classification on noisy signals that are randomly chosen from three different classes of sinusoidal, rectangular, and sawtooth functions. In all cases, the OPONN achieved accuracies exceeding 93% while operating at $\sim$ 10 GHz clock rates. Remarkably, this clock rate is only limited by the electronic sources responsible for generating the input pump pulses, and is not an inherent limit of our OPO-based neural network, which is operable with femtosecond pulses. By realizing the linear MAC operations as well as the nonlinear activation functions in the optical domain, our implementation also eliminates the need for OEO conversions that are typically required in state-of-the-art PNN architectures.

Figure 3.1 illustrates the OPO-based neurmorphic processor. At the heart of the processor is an on-chip OPO which is fabricated on an x-cut TFLN wafer with a silica buffer layer. It consists of a dispersion-engineered periodically-poled lithium niobate (PPLN) section that provides broadband quasi phase matching (QPM) between the pump at 1045 nm and the generated signal and idler waves. The cavity is formed by

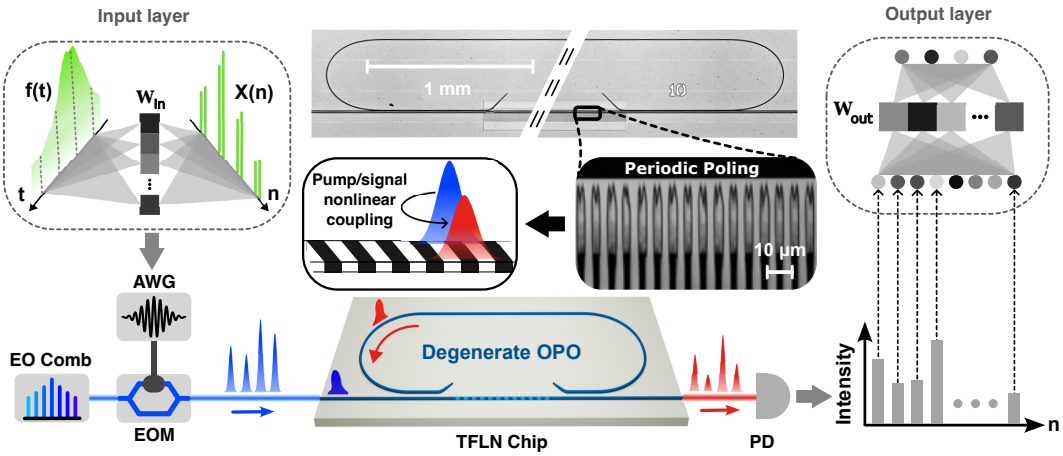

Figure 3.1: **Nanophotonic OPO-based neuromorphic processor.** Schematic view of the nanophotonic OPO utilized in our experiements for neuromorphic computing. The top panel depicts the microscope image of the device (more information about the chip design can be found in [25]). The central part of the racetrack resonator is removed for clarity. The structure is designed in such a way that only the degenerate signal/idler resonate inside the cavity. At the input, the optical pump generated from an EO comb at $f_{\text{rep}} \approx 10$ GHz and a center wavelength of ~ 1045 nm is modulated by an EOM. The modulating signal is generated by an AWG that applies discrete data $X(n)$ at ~ 10 GSa/s. This data represents a time series that is obtained by sampling an arbitrary function $f(t)$ in time, followed by masking with a randomized set of input weights $\mathbf{W}_{\text{in}}$, thus forming the input layer of our OPO neural network. The inset depicts a two-photon microscope image of the periodically poled region in the OPO cavity necessary for quasi phase matching (QPM). Within this poled region, the generated signal pulses from previous roundtrips nonlinearly couple to the incoming pump pulses, collectively acting as a nonlinear recurrent network that connects the input layer to the output (the inset schematically displays this nonlinear coupling). At the output, the signal pulses at ~ 2090 nm are separated from the pump using a fiber-based WDM filter and sent to a fast photodetector. The output layer of our OPONN is then set up by forming a weighted average of these detected pulses.

two input/output adiabatic couplers which are designed to only allow the signal/idler modes to resonate within the cavity [24]. Throughout our experiments, the OPO is operated in the degenerate regime. We synchronously pump the OPO using transform-limited pulses with a duration of ~ 2 ps generated by an EO comb with a repetition rate of ~ 10 GHz matching the cavity free spectral range (FSR). The input layer of the neural network is formed by using an electro-optic modulator (EOM) driven by an arbitrary waveform generator (AWG) operating at $\sim 10\,\text{GSa}/\text{s}$ to prepare the masked sequential data fed into the OPO as the pump pulses. During successive cycles defined by the roundtrip time of the OPO cavity, the input pump pulse and

the signal pulse traveling in the cavity overlap in the PPLN region. The signal pulse carries information from the previous signals due to the feedback provided by the cavity. In this respect, the resonator serves as an optical memory. Followed by this coherent interference between the signal and pump pulses, the PPLN element acts as a nonlinear neuron that generates a signal pulse which is a nonlinear function of its inputs, i.e., the current pump pulse and the previous signal pulse. In this respect, the nanophotonic OPO provides an all-optical implementation of a deep recurrent neural network. Finally, the weighted average of the signal pulse intensities resolved by a fast detector forms the output layer of the OPONN (Fig. 3.1).

Low latency time domain signal processing using OPONN

We employed the OPONN to perform machine learning on a variety of benchmark tasks that involve time-domain signals. The first task is to predict the evolution of a chaotic time series generated by the Lorenz63 model proposed by Lorenz in 1963 [26] to describe atmospheric convection. This is governed by a three-dimensional system of coupled nonlinear differential equations $\dot{x} = 10(y - x), \dot{y} = x(28 - z) - y, \dot{z} = xy - 8z/3$, where x, y and z denote physical observables associated with the convective current and the temperature variations in different spatial directions. For the set of parameters and the initial conditions chosen here, the Lorenz system exhibits deterministic chaos that is manifested by an aperiodic trajectory in the phase space known as a strange attractor. Under such conditions, the evolution of the observables show a dramatic sensitivity to the initial conditions, hence defying conventional methods of prediction. Our goal here is to train the OPONN to forecast the next time step of the signal $x(t)$ based on its past history. To achieve this, we first sample the input data at a sampling rate of $\sim$ 10 GHz to obtain the discrete signal $u(n)$. We then configure the network with $N_{\text{in}} = 3$ and $N_{\text{out}} = 10$ nodes in the input and the output layers, respectively. The input weights defined by the vector $\mathbf{W}_{\text{in}}$ are chosen randomly. We split the entire span of the available data into two equal periods of training and testing. Training is performed in silico by singular value decomposition to obtain the optimal output weight matrix $\mathbf{W}_{\text{out}}$ that results in the least estimation error during the training phase. In contrast to other types of deep neural networks that typically require computationally intensive training techniques such as backpropagation, our neural network can be trained considerably simpler with minimal computation requirements. Figure 3.2**a** shows measurement results obtained from experiments, where both the target and the predicted signals are displayed. To quantify the performance of the OPONN for this forecasting task,

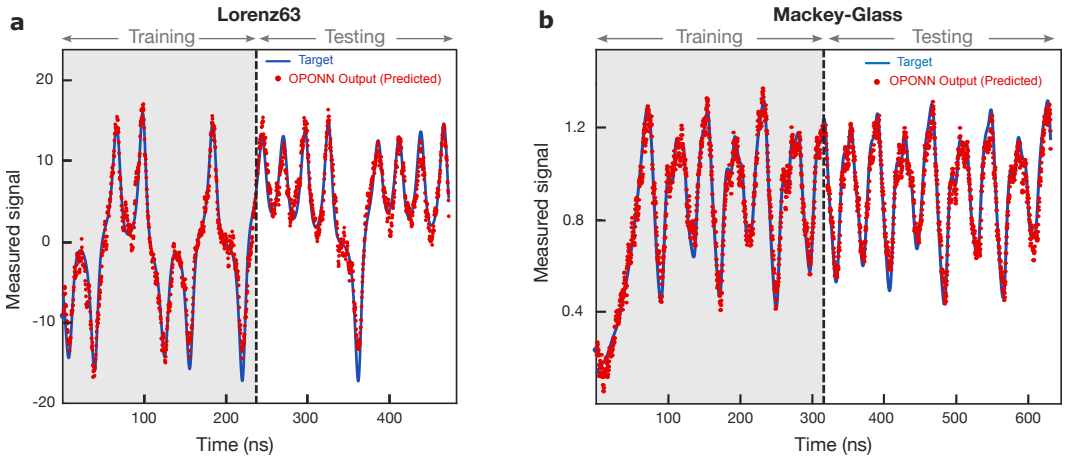

Figure 3.2: **Chaotic time series prediction using OPONN. a,** Numerically calculated values of the $x(t)$ signal associated with the Lorenz63 model (target) and the experimentally measured signal at the output of our OPONN (predicted). As shown in the figure, the entire span of the signal with a $\sim$ 500 ns duration is split into two equal sections for training (the shaded area) and testing. As evident, the predictions of the OPONN closely matches the target values. **b,** Similar results for the Mackey-Glass system of equations in the chaotic regime.

we calculate the normalized mean square error (NMSE) between the target and the predicted time series given as 0.07 ± 0.017 for this task.

As a second example, we consider another benchmark time series prediction task associated with the Mackey-Glass system (MGS), which was originally developed in the study of physiological control mechanisms that are known to exhibit a host of complex behaviors [27]. The model is described by a class of nonlinear delay differential equations exhibiting dynamics that critically depend on the system parameters. Of particular interest is the scenario when the solutions become chaotic, signaling an abnormal physiological behavior associated with pathological cases. Here, we use our OPO network to forecast the future instances of the resulting chaotic time series. Figure 3.2**b** depicts the target MGS solution together with predicted values obtained by measuring the OPONN output, indicating a prediction error of NMSE = 0.06 ± 0.017.

To demonstrate the versatility of our ultrafast nanophotonic processor, we next exploit it to compensate for nonlinear distortions that typically occur in a wireless communication channel. We assume that the input message to be transmitted contains a random stream of symbols $M(n)$ of the four-level-pulse-amplitude-modulation (PAM4) format. This modulation scheme is of practical interest in data centers

and is implemented for instance in state-of-the-art 800G transceivers. Upon entering the communication channel, the message is subjected to multiple adverse effects that degrade the fidelity of the received signal $S(n)$ at the end of the channel. First, the presence of scatterers located around the path between the source and the receiver result in multipath fading that causes intersymbol interference among adjacent symbols. In addition, to overcome the losses that are incurred through the signal propagation, it is desirable to maximize the power of the transmitted signal by increasing the gain from the power amplifier in the output stage of the transmitter. This increase in the power is accompanied by nonlinear effects that distort the message. Additionally, the signal to noise ratio (SNR) is diminished by various noise sources, including thermal fluctuations and the noise introduced by amplifiers. Previous studies have shown the potential of recurrent neural networks in correcting for these cumulative distortions [28–30]. Our objective is to perform this error correction task by exploiting the OPONN. Here, $N_{\text{in}} = 5$ and $N_{\text{out}} = 21$ nodes are used in the input and output layers, respectively. To assess the success of our approach, we calculate the symbol error rate (SER) defined as the percentage of the symbols detected incorrectly at the receiver end. Simulation results predict that the OPONN can effectively correct these errors and significantly improve the SER (Fig. 3.3**a**). Our experiments corroborate this and show that a direct detection of the corrupted signal at the channel output results in a high SER of 19% ± 1.6% (Fig. 3.3**b**). In contrast, once trained, the OPONN is able to improve this result to SER = 7% ± 2.3%. In order to confirm the role of the nonlinear processing performed by the OPO in error compensation, we also considered a purely linear

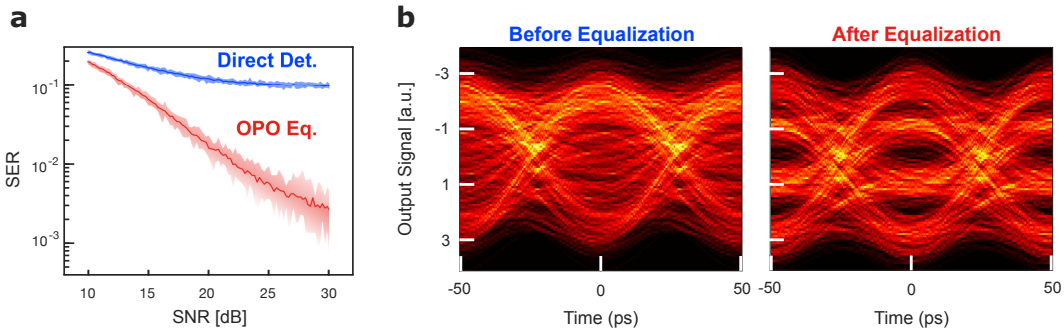

Figure 3.3: **Nonlinear channel equalization of PAM4 signals using ultrafast nonlinear response of the OPO network. a,** Simulated SERs obtained using direct detection versus those achieved after OPONN equalization for various SNR levels. **b,** Experimentally obtained eye diagrams without (left) and with (right) OPONN equalization.

equalization applied directly to the input pump, for which a SER = 11% ± 1.8% was measured. Given the fact that the measured SNR of the OPO output was lower than that associated with the input pump (please see Supplementary section 1), this latter comparison clearly indicates the efficacy of the OPO network to harness ultrafast photonic nonlinear processes to perform this equalization task.

One of the key applications in time domain signal processing involves the classification of various waveforms, with numerous applications ranging from bioinformatics [31, 32] to seismology [33]. In order to showcase the generality of our approach, we next utilized the OPO neural network to distinguish among three classes of different waveforms randomly chosen from sinusoidal, square, and sawtooth signals that are contaminated with noise (Fig. 3.4**a**). Each waveform contains $N = 50$ samples constituting two consecutive cycles. For this experiment, the OPONN is set up with $N_{\text{in}} = 5$ nodes at the input. At the output of the chip, the sub-harmonic signal generated by the OPO is sampled at three equidistant points, forming $K = 3$ output neurons (Fig. 3.4**a**). The output layer of the neural network consists of a fully connected layer with $N_{\text{out}} = 3$ nodes. During the training period, the linear weights associated with this output layer $\mathbf{W}_{\text{out}}$ are optimized according to a winner-takes-all approach. We use a training set of 300 waveforms (100 from every class) and a

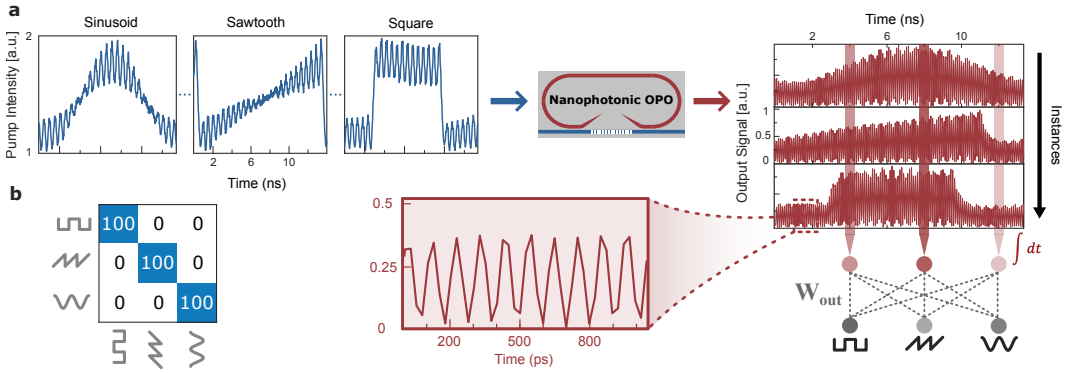

Figure 3.4: **Noisy waveform classification task. a,** We use the AWG to encode samples representing two cycles of periodic waveforms randomly chosen from three different classes of sawtooth, sinusoid and square signals with additive noise on the input pump (left). The measured signals at the output of the OPO associated with these waveforms are shown on the right. The output signals are sampled at three equally distant instances corresponding to three different nodes. The output layer of the OPO network is then set up by forming a fully-connected layer between these nodes and the output nodes that are activated based on a winner-takes-all approach. **b,** Using this simple architecture, we achieved 100% accuracy for this classification task.

different set of equal size for testing. In all cases, the OPONN was able to successfully classify all waveforms with 100% accuracy, despite the presence of significant noise levels at the input (Fig. 3.4**b**).

Discussion

In conclusion, we have demonstrated an ultrafast photonic neuromorphic processor based on a nanophotonic OPO chip operating at 10 GHz clock rates. Our approach leverages coherent optical pulse propagation in conjunction with ultrafast parametric nonlinear processes available in TFLN to perform both the linear MAC operations and nonlinear activation functions in the optical domain. Operating at $\sim$ 10 GHz clock rates, our OPO neural network is capable of achieving sub-nanosecond latencies, a timescale comparable to a single clock cycle in state-of-the-art digital electronic processors. We showcased the power of OPONN in performing a variety of machine learning tasks that involve time series signals including chaotic time series prediction, nonlinear channel equalization, as well as noisy waveform classification. In all cases, the OPONN achieved success rates exceeding 93%. By eliminating successive slow and energy-intensive OEO conversions, our OPONN represents a significant leap towards realizing integrated all-optical neuromorphic processors operating at unprecedented high speeds while maintaining low energy consumptions.

3.3 Supplementary Information

Experimental setup

Figure 3.5 shows a detailed schematic of our experimental setup. We use an electro-optic (EO) comb to generate optical pulses at 10 GHz repetition rate and center wavelength of 1045 nm with a pulsewidth of $\sim 2p$s to synchronously pump the nanophotonic OPO. These pulses are passed through a booster optical amplifier (BOA), wave shaper, and Ytterbium-doped fiber amplifier (YDFA) for amplification and dispersion compensation. The sequential data at the input layer is prepared by modulating the pump pulses using an intensity modulator (IM) driven by an arbitrary waveform generator (AWG) operating at 10 GSa/s. The modulated pump pulses are further amplified using a second YDFA before being coupled into the chip. Lensed fibers are used to couple the pump into the chip and to collect the emitted signal at the output. Our on-chip OPO is fabricated on an x-cut, 700-nm-thick thin-film lithium niobate (TFLN) wafer with a silica buffer layer. It comprises two input/output adiabatic couplers before and after the periodically poled section which

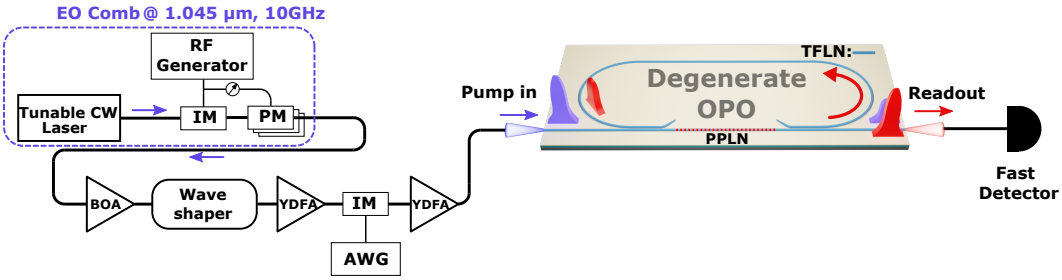

Figure 3.5: **Schematic diagram of the OPONN experimental setup.**

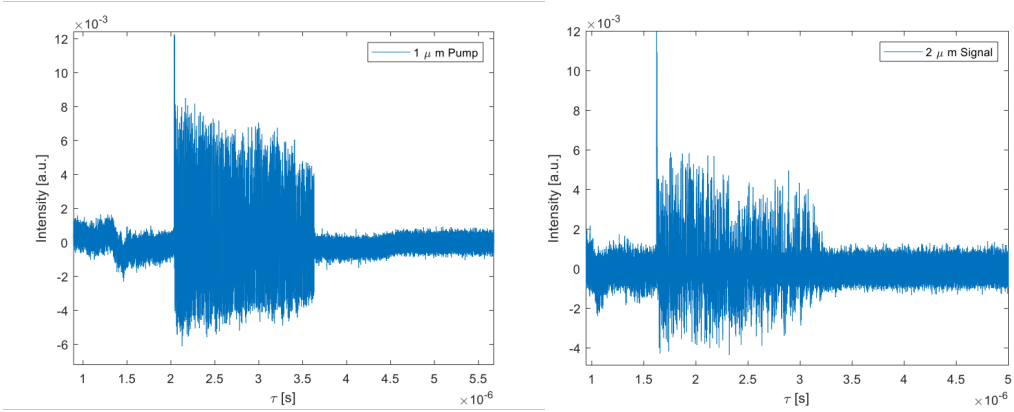

Figure 3.6: **Different SNR levels associated with the measured pump and signal pulses.** Example measurement traces involving the pump pulses @ 1045 nm (left) and the emitted signal pulses @ 2090 nm by the OPO (right).

are designed to only allow the degenerate signal/idler modes to resonate within the cavity. The emitted signal pulses at 2090 nm are resolved using a fast biased detector (with 12 GHz bandwidth) and form the output layer of the OPONN. As evident from experimental measurements presented in Fig. 3.6, the signal-to-noise ratio (SNR) of the detected pump pulses were significantly higher than that associated with the signal pulses.

Comparison to other physical reservoir computers

The concept of the OPONN is closely related to physical reservoir computing [34], which exploits complex nonlinear dynamics in analog physical systems. In this section, we compare our OPONN with other experimentally demonstrated physical reservoir computers in different platforms based on various physical mechanisms including optoelectronic, magnetic, electrical, mechanical, chemical, atomic, and

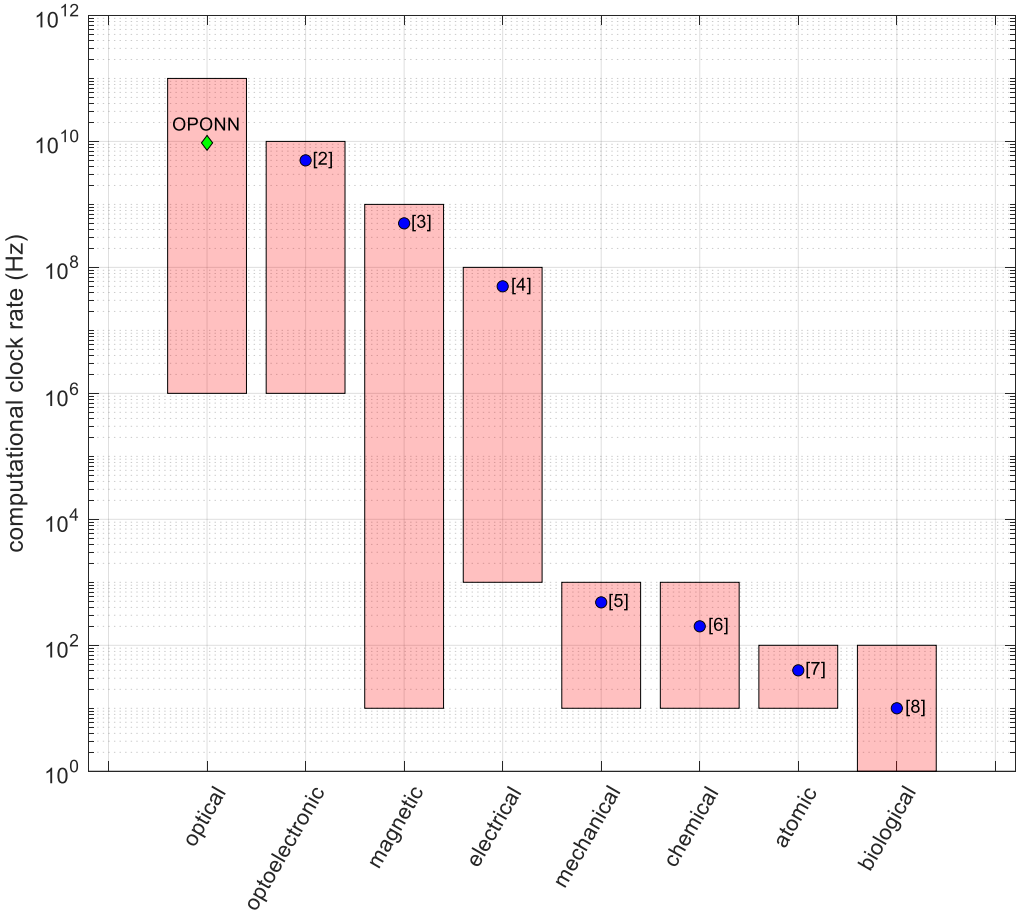

Figure 3.7: **Comparison of computational clock rates.** Bars show the rough order-of-magnitude ranges for clock rates in experimentally demonstrated physical reservoir computers based on different types of physical mechanisms. The marker point in each bar shows a representative state-of-the-art example for that category, numbers in brackets correspond to the reference numbers in Table 1.

biological. The unique advantage of the OPONN is its exceptionally high computational clock rate, whereas other physical reservoir computers usually prioritize low power consumption. Here, we restrict our attention to experimentally demonstrated results. Although the types of computational operations and network architectures can vary greatly between physical reservoir computers, we consider the maximum effective computational clock rate (even if the implementation does not have a literal hardware clock as part of the design) to be given by the fastest reported input update rate or minimum time between successive operations in the physical recurrent/reservoir layer. The maximum clock rate therefore relates to the theoretical

Ref.	Type	Platform	Tasks	Clock Rate
This Work	Optical	Nanophotonic optical parametric oscillator	Chaotic time series forecasting, nonlinear error correction, waveform classification	9.47 GHz
[35]	Optoelectronic	Semiconductor fiber laser	Spoken digit recognition, chaotic time-series forecasting	5 GHz
[36]	Magnetic	Spin-torque nano-oscillator	Spoken digit recognition, waveform classification	500 MHz
[37]	Electrical	Ferroelectric field-effect transistor	Boolean logic, nonlinear time series forecasting	50 MHz
[38]	Mechanical	Miura-ori origami	Nonlinear filter emulation, Volterra series, nonlinear limit cycles	480 Hz
[39]	Chemical	Organic electro-chemical transistor	Heartbeat classification	200 Hz
[40]	Atomic	Boron dopants in silicon	Boolean logic, MNIST image classification	40 Hz
[41]	Biological	Dissociated rat E18 cortical neurons	Spike timing, random music classification	10 Hz

Table 3.1: **Comparison of state-of-the-art experimentally reported physical reservoir computers based on different types of physical mechanisms.**

minimum latency or fastest input signal time scales that the physical reservoir computer was demonstrated to respond to in real-time. The rough order-of-magnitude ranges for different categories of experimentally demonstrated physical reservoir computers and a corresponding state-of-the-art example for each category are shown in Fig. 3.7. A comparison of the representative state-of-the-art examples for each category of physical mechanisms of the physical reservoir computer is shown in Table 3.1.

All-optical output layer

In the main text experiments, we focused on demonstrating the ultrafast nonlinear recurrent layer of the OPONN, which comprises the bulk of the computation and primary bottleneck of neuromorphic processors. We chose to perform the linear output layer during post-processing on a conventional digital electronic computer for simplicity since many previous works have shown how to accelerate linear operations using optics. In this section, we propose a simple implementation for an end-to-end all-optical OPONN that does not rely on electronics for executing the linear output layer.

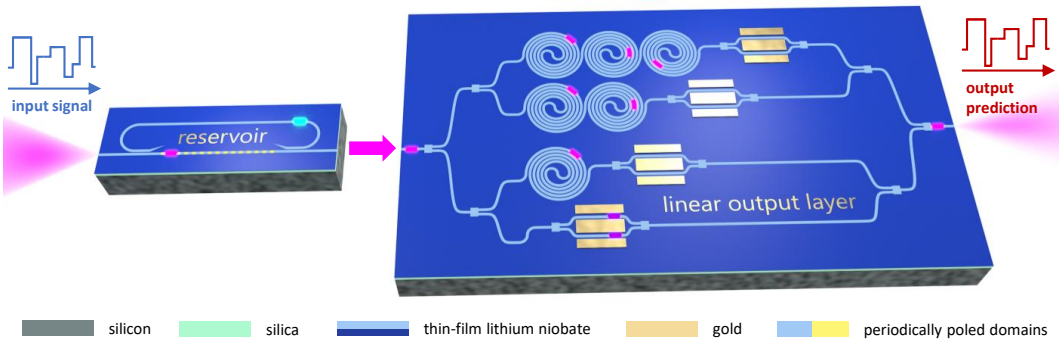

Figure 3.8: **An end-to-end all-optical OPONN.** Both the reservoir and linear output layers can be implemented directly in the same thin-film lithium niobate nanophotonic platform.

In the context of physical reservoir computers, the reservoir layer acts as a recurrent layer with highly nonlinear and complex dynamics, and the linear output layer contains fixed weights trained by linear regression. In the OPONN, the linear output layer can be implemented in the same thin-film lithium niobate nanophotonic platform as the reservoir layer. For example, Fig. 3.8 shows a multi-arm Mach-Zehnder interferometer that can perform a linear dot product for a single-channel output (e.g., for time series forecasting tasks) directly in the optical domain. The relative delays in each arm encode for the corresponding time-delayed reservoir variables, which are represented by the coherent amplitudes of ultrashort laser pulses. The electro-optical amplitude modulators in each arm encode the fixed weights of the linear output layer. Therefore, the reservoir variables can be measured during the training phase, then the optimal output layer weights can be determined using linear regression (usually on a digital computer), and finally used to set the bias voltages for the electro-optic amplitude modulators in the output layer. Once the output layer weights are determined, the computation can be performed entirely in the optical domain during testing or inference stages.

References

[1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[2] Farshid Ashtiani, Alexander J. Geers, and Firooz Aflatouni. An on-chip photonic deep neural network for image classification. *Nature*, 606(7914):501–506, 2022.

[3] Yitong Chen, Maimaiti Nazhamaiti, Han Xu, Yao Meng, Tiankuang Zhou, Guangpu Li, Jingtao Fan, Qi Wei, Jiamin Wu, Fei Qiao, Lu Fang, and Qionghai Dai. All-analog photoelectronic chip for high-speed vision tasks. *Nature*, 623 (7985):48–57, 2023.

[4] Sunil Pai, Zhanghao Sun, Tyler W. Hughes, Taewon Park, Ben Bartlett, Ian A. D. Williamson, Momchil Minkov, Maziyar Milanizadeh, Nathnael Abebe, Francesco Morichetti, Andrea Melloni, Shanhui Fan, Olav Solgaard, and David A. B. Miller. Experimentally realized in situ backpropagation for deep learning in photonic neural networks. *Science*, 380(6643):398–404, 2023.

[5] Gordon Wetzstein, Aydogan Ozcan, Sylvain Gigan, Shanhui Fan, Dirk Englund, Marin Soljačić, Cornelia Denz, David A. B. Miller, and Demetri Psaltis. Inference in artificial intelligence with deep optics and photonics. *Nature*, 588 (7836):39–47, 2020.

[6] Bhavin J. Shastri, Alexander N. Tait, T. Ferreira de Lima, Wolfram H. P. Pernice, Harish Bhaskaran, C. D. Wright, and Paul R. Prucnal. Photonics for artificial intelligence and neuromorphic computing. *Nature Photonics*, 15(2): 102–114, 2021.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[8] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596 (7873):583–589, 2021.

[9] Trieu H. Trinh, Yuhuai Wu, Quoc V. Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.

[10] Jaime Sevilla, Lennart Heim, Anson Ho, Tamay Besiroglu, Marius Hobbhahn, and Pablo Villalobos. Compute Trends Across Three Eras of Machine Learning. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2022.

[11] Peter L. McMahon. The physics of optical computing. *Nature Reviews Physics*, 5(12):717–734, 2023.

[12] Hanke Feng, Tong Ge, Xiaoqing Guo, Benshan Wang, Yiwen Zhang, Zhaoxi Chen, Sha Zhu, Ke Zhang, Wenzhao Sun, Chaoran Huang, Yixuan Yuan, and Cheng Wang. Integrated lithium niobate microwave photonic processing engine. *Nature*, 627(8002):80–87, 2024.

[13] Yichen Shen, Nicholas C. Harris, Scott Skirlo, Mihika Prabhu, Tom Baehr-Jones, Michael Hochberg, Xin Sun, Shijie Zhao, Hugo Larochelle, Dirk Englund, and Marin Soljačić. Deep learning with coherent nanophotonic circuits. *Nature Photonics*, 11(7):441–446, 2017. Publisher: Nature Publishing Group.

[14] SeyedMohammad SeyedinNavadeh, Maziyar Milanizadeh, Francesco Zanetto, Giorgio Ferrari, Marco Sampietro, Marc Sorel, David A. B. Miller, Andrea Melloni, and Francesco Morichetti. Determining the optimal communication channels of arbitrary optical systems using integrated photonic processors. *Nature Photonics*, 18(2):149–155, 2024.

[15] Zaijun Chen, Alexander Sludds, Ronald Davis, Ian Christen, Liane Bernstein, Lamia Ateshian, Tobias Heuser, Niels Heermeier, James A. Lott, Stephan Reitzenstein, Ryan Hamerly, and Dirk Englund. Deep learning with coherent VCSEL neural networks. *Nature Photonics*, 17(8):723–730, 2023.

[16] Johannes Feldmann, Nathan Youngblood, Christopher D. Wright, Harish Bhaskaran, and Wolfram H.P. Pernice. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature*, 569(7755):208–214, 2019.

[17] Johannes Feldmann, Nathan Youngblood, Maxim Karpov, Helge Gehring, Xuan Li, Maik Stappers, Manuel Le Gallo, Xin Fu, Anton Lukashchuk, Arslan S. Raja, et al. Parallel convolutional processing using an integrated photonic tensor core. *Nature*, 589(7840):52–58, 2021.

[18] Frank Brückerhoff-Plückelmann, Ivonne Bente, Marlon Becker, Niklas Vollmar, Nikolaos Farmakidis, Emma Lomonte, Francesco Lenzini, C. David Wright, Harish Bhaskaran, Martin Salinga, Benjamin Risse, and Wolfram H. P. Pernice. Event-driven adaptive optical neural network. *Science Advances*, 9 (42):eadi9127, 2023.

[19] Gordon H. Y. Li, Ryoto Sekine, Rajveer Nehra, Robert M. Gray, Luis Ledezma, Qiushi Guo, and Alireza Marandi. All-optical ultrafast ReLU function for energy-efficient nanophotonic deep learning. *Nanophotonics*, 12(5):847–855, 2023.

[20] Logan G. Wright, Tatsuhiro Onodera, Martin M. Stein, Tianyu Wang, Darren T. Schachter, Zoey Hu, and Peter L. McMahon. Deep physical neural networks trained with backpropagation. *Nature*, 601(7894):549–555, 2022.

[21] James Spall, Xianxin Guo, and A. I. Lvovsky. Hybrid training of optical neural networks. *Optica*, 9(7):803–811, 2022.

[22] Charles Roques-Carmes, Yannick Salamin, Jamison Sloan, Seou Choi, Gustavo Velez, Ethan Koskas, Nicholas Rivera, Steven Kooi, John D. Joannopoulos, and Marin Soljačić. Biasing the quantum vacuum to control macroscopic probability distributions. *Science*, 381(6654):205–209, 2023.

[23] Seou Choi, Yannick Salamin, Charles Roques-Carmes, Rumen Dangovski, Di Luo, Zhuo Chen, Michael Horodynski, Jamison Sloan, Shiekh Zia Uddin, and Marin Soljačić. Photonic probabilistic machine learning using quantum vacuum noise. *Nature Communications*, 15(1):7760, 2024.

[24] Arkadev Roy, Luis Ledezma, Luis Costa, Robert Gray, Ryoto Sekine, Qiushi Guo, Mingchen Liu, Ryan M Briggs, and Alireza Marandi. Visible-to-mid-ir tunable frequency comb in nanophotonics. *Nature Communications*, 14(1): 6549, 2023.

[25] Luis Ledezma, Arkadev Roy, Luis Costa, Ryoto Sekine, Robert Gray, Qiushi Guo, Rajveer Nehra, Ryan M. Briggs, and Alireza Marandi. Octave-spanning tunable infrared parametric oscillators in nanophotonics. *Science Advances*, 9 (30):eadf9711, 2023.

[26] Edward N. Lorenz. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141, 1963.

[27] Michael C. Mackey and Leon Glass. Oscillation and chaos in physiological control systems. *Science*, 197(4300):287–289, 1977. Publisher: American Association for the Advancement of Science.

[28] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667): 78–80, April 2004.

[29] Quentin Vinckier, François Duport, Anteo Smerieri, Kristof Vandoorne, Peter Bienstman, Marc Haelterman, and Serge Massar. High-performance photonic reservoir computer based on a coherently driven passive cavity. *Optica*, 2(5): 438–446, 2015.

[30] Alessandro Lupo, Enrico Picco, Marina Zajnulina, and Serge Massar. Deep photonic reservoir computer based on frequency multiplexing with fully analog connection between layers. *Optica*, 10(11):1478–1485, 2023.

[31] Seonwoo Min, Byunghan Lee, and Sungroh Yoon. Deep learning in bioinformatics. *Briefings in Bioinformatics*, 18(5):851–869, 2017.

[32] Bowei Dong, Samarth Aggarwal, Wen Zhou, Utku Emre Ali, Nikolaos Farmakidis, June Sang Lee, Yuhan He, Xuan Li, Dim-Lee Kwong, C. D. Wright, Wolfram H. P. Pernice, and H. Bhaskaran. Higher-dimensional processing using a photonic tensor core with continuous-time data. *Nature Photonics*, 17 (12):1080–1088, 2023.

[33] S. Mostafa Mousavi and Gregory C. Beroza. Deep-learning seismology. *Science*, 377(6607):eabm4470, 2022.

[34] Kohei Nakajima. Physical reservoir computing—An introductory perspective. *Japanese Journal of Applied Physics*, 59(6):060501, 2020.

[35] Daniel Brunner, Miguel C. Soriano, Claudio R. Mirasso, and Ingo Fischer. Parallel photonic information processing at gigabyte per second data rates using transient states. *Nature Communications*, 4(1):1364, 2013.

[36] Jacob Torrejon, Mathieu Riou, Flavio Abreu Araujo, Sumito Tsunegi, Guru Khalsa, Damien Querlioz, Paolo Bortolotti, Vincent Cros, Kay Yakushiji, Akio Fukushima, et al. Neuromorphic computing with nanoscale spintronic oscillators. *Nature*, 547(7664):428–431, 2017.

[37] Kasidit Toprasertpong, Eishin Nako, Zeyu Wang, Ryosho Nakane, Mitsuru Takenaka, and Shinichi Takagi. Reservoir computing on a silicon platform with a ferroelectric field-effect transistor. *Communications Engineering*, 1(1): 21, 2022.

[38] Priyanka Bhovad and Suyi Li. Physical reservoir computing with origami and its application to robotic crawling. *Scientific Reports*, 11(1):13002, 2021.

[39] Matteo Cucchi, Christopher Gruener, Lautaro Petrauskas, Peter Steiner, Hsin Tseng, Axel Fischer, Bogdan Penkovsky, Christian Matthus, Peter Birkholz, Hans Kleemann, et al. Reservoir computing with biocompatible organic electrochemical networks for brain-inspired biosignal classification. *Science Advances*, 7(34):eabh0693, 2021.

[40] Tao Chen, Jeroen van Gelder, Bram van de Ven, Sergey V. Amitonov, Bram De Wilde, Hans-Christian Ruiz Euler, Hajo Broersma, Peter A. Bobbert, Floris A. Zwanenburg, and Wilfred G. van der Wiel. Classification with a disordered dopant-atom network in silicon. *Nature*, 577(7790):341–345, 2020.

[41] Han Ju, Mark R. Dranias, Gokulakrishna Banumurthy, and Antonius M.J. VanDongen. Spatiotemporal memory is an intrinsic property of networks of dissociated cortical neurons. *Journal of Neuroscience*, 35(9):4040–4051, 2015.

Chapter 4

PHOTONIC ELEMENTARY CELLULAR AUTOMATA FOR SIMULATION OF COMPLEX PHENOMENA

Gordon H.Y. Li, Christian R. Leefmans, James Williams, and Alireza Marandi. Photonic elementary cellular automata for simulation of complex phenomena. *Light: Science & Applications*, 12(1):132, 2023. doi:10.1038/s41377-023-01180-9.
G.H.Y.L. conceived the project, developed the theory, executed numerical simulations, designed the experiments, participated in building the experimental setup, performed the experimental measurements, analyzed the data, and wrote the manuscript.

4.1 Abstract

Cellular automata are a class of computational models based on simple rules and algorithms that can simulate a wide range of complex phenomena. However, when using conventional computers, these 'simple' rules are only encapsulated at the level of software. This can be taken one step further by simplifying the underlying physical hardware. Here, we propose and implement a simple photonic hardware platform for simulating complex phenomena based on cellular automata. Using this special-purpose computer, we experimentally demonstrate complex phenomena including fractals, chaos, and solitons, which are typically associated with much more complex physical systems. The flexibility and programmability of our photonic computer presents new opportunities to simulate and harness complexity for efficient, robust, and decentralized information processing using light.

4.2 Introduction

Modern digital electronic computers, which are based on the von Neumann architecture, exhibit extreme hardware complexity in their construction and are composed of billions of transistors engineered in a hierarchical and highly structured manner. Unlike the von Neumann architecture, nature is abound with emergent phenomena and complex systems containing many interacting components following simple rules with no hierarchical control. For example, social insects like ants with only limited local information can collectively self-organize to form global structures [1]. This suggests that an alternative, and potentially more efficient, way to simulate such

phenomena is to harness simple and decentralized physical hardware that directly emulates the underlying rules of a complex system.

One class of computational models that can benefit from simple and decentralized physical hardware are Cellular Automata (CA), which exhibit complex behaviour emerging from the local interactions of cells arranged on a regular lattice [2]. CA were introduced in the 1940s to study how self-replication and evolution can emerge in artificial life [3], and was later popularized in Conway's Game of Life [4], which exhibits self-organizing patterns reminiscent of biological systems. Subsequent landmark studies revealed that CA are also capable of replicating other complex behaviour such as fractals [5], chaos [6], self-organized criticality [7], synchronization [8], and universal computation [9]. Consequently, CA have found utility in modelling a wide range of natural phenomena in physics [10, 11], chemistry [12–14], and biology [15]. Furthermore, CA have important applications in real-world computational problems such as cryptography [16], data compression [17], error-correction [18], simulating traffic flow [19], and developing more robust artificial intelligence [20]. Owing to their simple formulations, certain CA of interest are computationally irreducible [21], i.e., there are no analytical shortcuts to evaluate their state after an arbitrary time without resorting to executing the sequential simulation in its entirety. On the other hand, most CA are only implemented as high-level software on conventional computers, resulting in unnecessary overhead. Therefore, it is desirable to seek out physical hardware that better encapsulates the computational principles of CA to enable more efficient simulation. Notable previous attempts to implement physical systems tailored to perform CA include self-assembling DNA molecules [22], arrays of nanomagnets [23], memristor networks [24], and living slime moulds [25].

In this work, we propose and experimentally implement a photonic computational platform capable of simulating complex phenomena using CA. Compared to other approaches, our photonic platform offers several distinct advantages: (1) the inherently high-bandwidth endowed by computing using light offers potentially orders-of-magnitude speed-up in clock rate over the simulation of CA on conventional von Neumann computers, (2) rapid reconfigurability for easy programming of a variety of CA rules enables many different complex phenomena to be observed in the *same* physical system, and (3) the kind of sparse, local, and shift-invariant connections required for CA are well-suited for this platform. We will demonstrate how even simple photonic hardware can host a wide range of complex emergent phenomena

and is capable of sophisticated (or even universal) computation. By exploiting this complexity, we reveal a path towards the next generation of more efficient or robust photonic hardware accelerators for reservoir computing [26, 27] and deep learning [20, 28].

4.3 Results

We focus on the simplest types of CA called Elementary Cellular Automata (ECA) [5]. These are discrete-time dynamical systems defined on a 1D lattice of cells with binary states that evolve according to Eq. 4.1:

$$x_i(t+1) = f\left(x_{i-1}(t), x_i(t), x_{i+1}(t)\right) \ , \tag{4.1}$$

where $x_i(t) \in \mathbb{Z}_2$ is the state (i.e., dead or live) of the i^{th} cell at time step t, and $f : (\mathbb{Z}_2)^3 \to \mathbb{Z}_2$ is the update rule. Crucially, the rules specifying interactions amongst cells are computed using only local nearest-neighbour information without reference to the global pattern. Remarkably, not only can the underlying rules be simple, but the initial conditions can also be simple — consisting, for example, of just a single live cell — and yet the collective behaviour produced can still be highly complex [5]. The 256 possible ECA rules encapsulate a wide range of complex phenomena and are representative of the four universality classes of increasing complexity introduced by Wolfram [29].

Here, we experimentally implement ECA in a time-multiplexed photonic system as shown in Fig. 4.1(a). Cell states are represented using pulses of light produced by a mode-locked laser (MLL) with a fixed repetition rate T_R. The presence of a pulse indicates a live cell; the absence, a dead cell. In this way, the complete pulse train from the MLL represents a synthetic 1D lattice [30], where each site in the lattice is encoded as a time bin of width T_R in the pulse train. The pulse amplitude/phase representing the initial cell state is encoded using an electro-optic modulator (EOM), and the pulse is then split between three optical delay lines. These delay lines, which are labeled $0T_R$ and $\pm 1T_R$ in Fig. 4.1(a), are chosen such that the $+(-)1T_R$ delay line is one pulse repetition period longer (shorter) than the $0T_R$ delay line. With this choice of lengths, light in the $0T_R$ delay line coherently interferes with its nearest-neighbor time slots in the temporal synthetic lattice. The coherent interference is followed by optoelectronic thresholding to enforce the binary constraint for the updated cell state in the ECA. Finally, the optoelectronic signal is stored on a field-programmable gate array (FPGA), which feeds back the measured cell states by driving the input EOM for the next iteration. By repeating this process for many

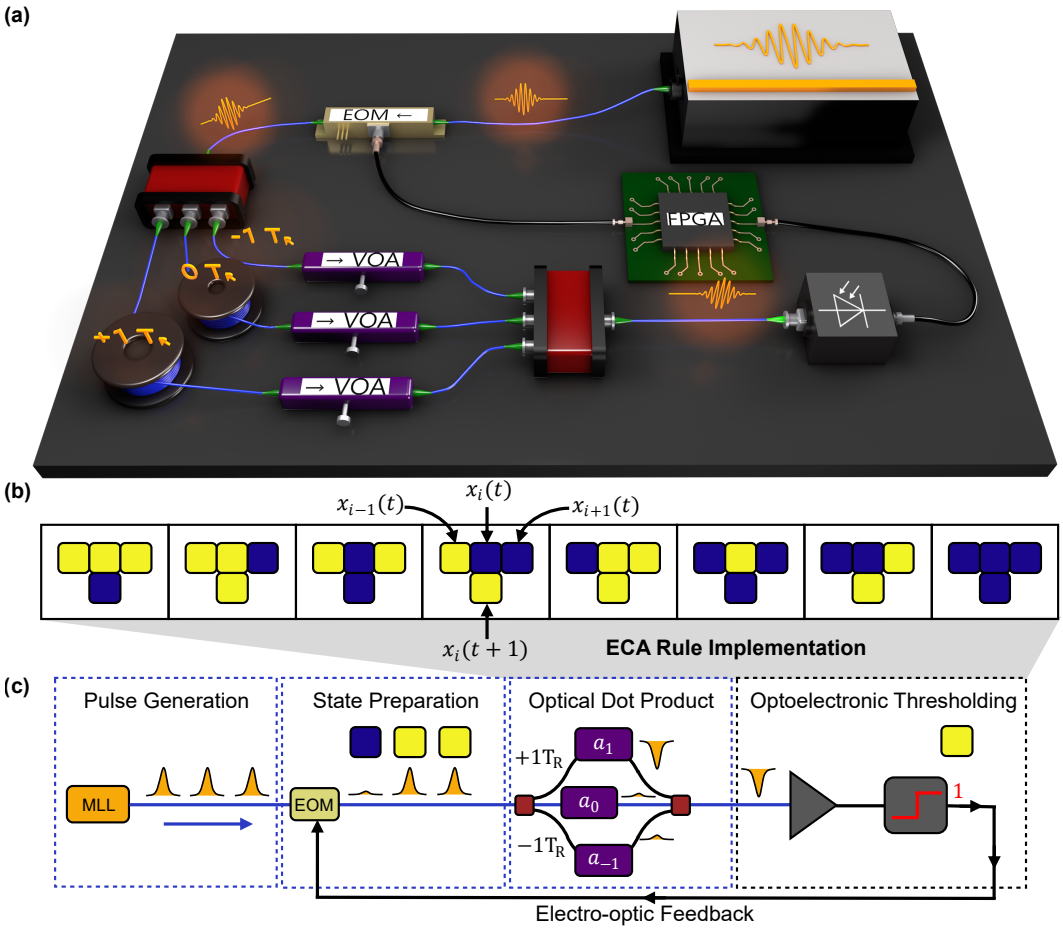

Figure 4.1: **Photonic platform for simulating complex phenomena using Elementary Cellular Automata.** (a) Schematic of the experimental setup. Cells are represented by pulses of light produced by a mode-locked laser (MLL) with a repetition rate of T_R. The cell states are encoded by an electro-optic modulator (EOM) and are split into optical fiber delay lines (blue lines) to induce local interactions of neighbouring light pulses. Specific ECA rules are programmed by tuning the variable optical attenuator (VOA) in each delay line. Optoelectronic thresholding is performed following the coherent interference of light pulses, with the resultant cell states stored on a field-programmable gate array (FPGA) and reinjected (black lines) to drive the input EOM for the next iteration. (b) Truth table showing the uniform and synchronous update for ECA Rule 90 with the top row in each case representing the current states of the 3-cell neighbourhood and the bottom row showing the cell state during the next iteration. (c) Block diagram showing the different stages of computation and flow of information in the photonic ECA implementation.

cycles, we observe the emergence of complex phenomena in the cell states of the ECA. The desired ECA update rule, such as ECA Rule 90 (following the Wolfram naming convention) shown in Fig. 4.1(b), is programmed by tuning the thresholding value and variable optical attentuator (VOA) in each delay line, which represents constant amplitude/phase weights. As shown in Fig. 4.1(c), this rule encoding can be interpreted as an optical dot product followed by a nonlinear thresholding function, which is akin to a single perceptron in the context of artificial neural networks [31]. Therefore, the dynamics of the abstract ECA rule is exactly mapped to the physical time-evolution of the photonic simulator. We note that similar time-multiplexing techniques with delay dynamics have been used for photonic reservoir computing [32], coherent Ising machines [33], simulating topological lattices [34], and optoelectronic oscillators with time-delayed feedback [35]. Analogous results can also be obtained using other multiplexing methods such as frequency or spatial multiplexing [30].

Firstly, one of the most striking patterns that emerge in CA are fractals, which are often self-similar geometric shapes that appear the same at any scale. Fractals are ubiquitous in nature and occur in a diverse range of physical phenomena including the rings of Saturn [36], snowflakes [37], and fault geology [38]. ECA Rule 90, defined in Fig. 4.1(b), provides a simple model for fractal formation and self-replication. The local update rule can be expressed succinctly in terms of Boolean algebra as $x_i(t+1) = x_{i-1}(t) \bigoplus x_{i+1}(t)$, where $\bigoplus$ denotes the exclusive-or (XOR) logical operation. Thus, for this specific rule, the iterated cell state depends only on the states of its two neighbours. The fractal pattern is an emergent property of the nonlinear dynamics in the photonic computer, rather than being imposed on the system by an external ordering influence such as explicit geometric constructions in previous studies of photonic fractals [39–42]. We show the experimentally measured space-time equivalent diagram of ECA Rule 90, starting from a single live cell, in Fig. 4.2(a). The position of a cell in space (left-to-right) is represented by the pulse number in the synthetic temporal dimension, and the discrete-time step (top-to-bottom) is defined according to Eq. 4.1. The color of each cell is determined by the normalized peak pulse intensity *before* thresholding. We see that the space-time diagram is in the shape of the well-known Sierpinski Triangle. This fractal can be constructed by recursively subdividing an equilateral triangle into four smaller equilateral triangles and removing the central triangle. It is characterized by a non-integer Hausdorff or fractal dimension of $\log 3/\log 2 \approx 1.585$. The self-similarity of the fractal shape persists down to the cellular scale as shown in Fig. 4.2(b) and can be

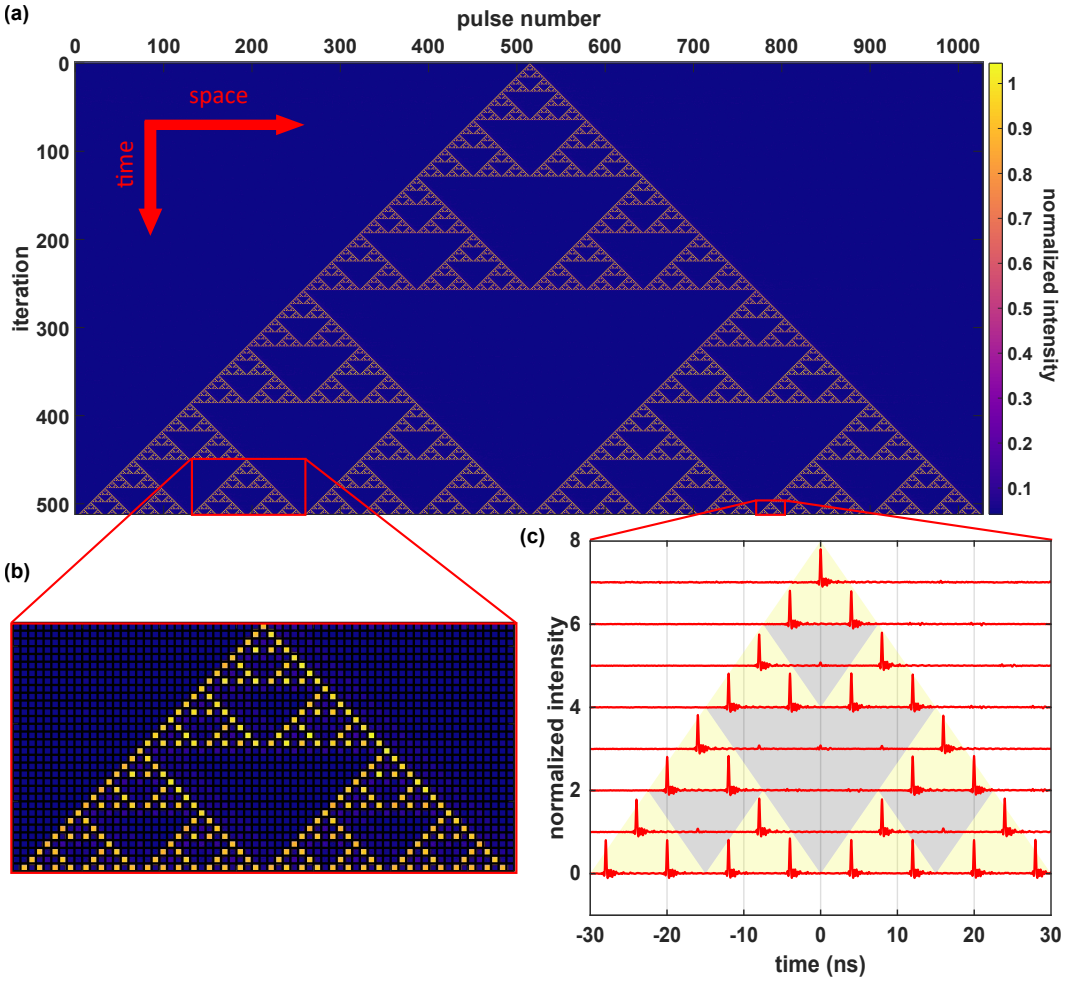

Figure 4.2: **Experimental result of ECA Rule 90 on the photonic hardware starting from a single live cell.** (a) Zoomed-out equivalent space-time diagram showing emergence of the Sierpinski Triangle fractal. (b) Zoomed-in view showing the fractal self-similarity down to the cellular scale. (c) Time traces (vertically separated for easier viewing) of the individual light pulses representing each cell separated by 4 ns.

seen in the time traces shown in Fig. 4.2(c), where individual light pulses represent each live cell. In this case, the middle optical delay line (0 T_R in Fig. 4.1(a)) can be ignored since the iterated cell state does not depend on its current state. Experimentally, this allows us to achieve an excellent extinction ratio between pulse peaks for live and dead cells and indicates that the ECA Rule 90 is implemented as intended.

Next we investigate ECA Rule 30, defined in Fig. 4.3(a), which is categorized as a member of class 3 CA according to Wolfram's universal complexity classes. These are CA that produce chaotic and seemingly random patterns, although some small-

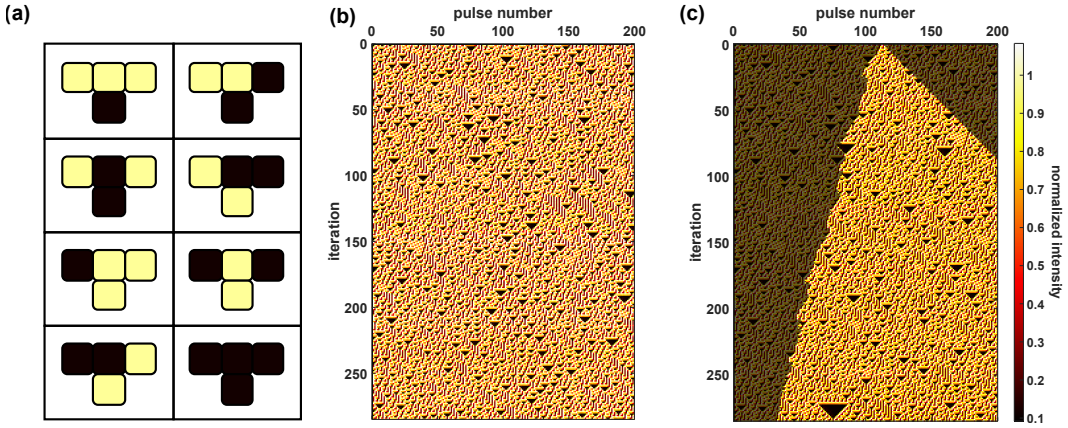

Figure 4.3: **Chaotic patterns produced by ECA Rule 30 on the time-multiplexed photonic hardware.** (a) Truth table showing the update for ECA Rule 30. (b) Space-time diagram of ECA Rule 30 starting from a random initial condition showing chaotic dynamics. (c) Inverting a single cell state in the initial condition produces a pattern with differences that grow linearly to the right and asymptotically linearly to the left (regions that are different to (b) are highlighted, and regions that are identical are displayed as partially transparent), hence demonstrating sensitivity to initial conditions.

scale structures are present [29]. Remarkably, ECA Rule 30 is one of the simplest known systems to exhibit chaos [6]. We experimentally demonstrate such a chaotic behaviour of ECA Rule 30 on the same simple photonic hardware in Fig. 4.3(b) starting from a random initial condition. In this case, there is greater variability in the peak pulse intensities compared to Rule 90 due to the lower interference visibility between three optical delay lines. However, the optoelectronic thresholding is still adequate to ensure the intended operation of ECA Rule 30. A necessary (but not sufficient) condition for chaos is sensitivity to initial conditions. Fig. 4.3(c) shows the space-time diagram starting from the same initial condition as Fig. 4.3(b), but with one cell inverted. The region of differences between the two patterns grows linearly to the right with Lyapunov exponent $\lambda_R = 1$ and asymptotically linearly to the left with Lyapunov exponent $\lambda_L \approx 0.24$, hence implying an exponential divergence in the cell configurations over time and sensitivity to initial conditions. Other necessary conditions for chaos such as non-periodicity and topological mixing have also been verified empirically [6].

Due to the simplicity of ECA Rule 30, it can be used as an efficient pseudo-random number generator. This can be accomplished, for example, by taking the sequence defined by the states of the central cell as it evolves in time, i.e., the middle column of the space-time diagram. Therefore, the initial condition acts as the seed. Impor-

tantly, ECA Rule 30 is highly nonlinear and computationally irreducible, unlike ECA Rule 90, which is linear (modulo 2) and amenable to algebraic analysis [43]. Indeed, detailed statistical analysis of the sequences produced by ECA Rule 30 shows that it is both a fast and safe random number generator [6]. Unlike previous photonic random number generators [44–46] relying on quantum processes or other continuous fluctuations, our system is pseudo-random, which means it is deterministic and repeatable given the initial seed. This is often useful in practice to reliably reproduce results in applications such as Monte Carlo simulations [47], stream ciphers [48], and generative adversarial networks [49]. We note that ECA Rule 30 was previously demonstrated using free-space optics [50], however, this implementation encoded cells on pixels of 2D liquid-crystal screens, which introduced some redundancy. In contrast, our approach more faithfully implements the 1D lattice for ECA, can be easily extended to an arbitrary number of cells, and is easily programmable to implement more than just a single rule.

Finally, we study class 4 CA, which involve a mixture of order and randomness, with localized structures that move and interact in complicated ways [51]. A well-studied example of this is ECA Rule 54, defined in Fig. 4.4(a), which can be interpreted as a discrete analogue of excitations in an active nonlinear medium with mutual inhibition [52]. In this case, the mobile self-localizations called *gliders* appear on a stable periodic background called the *ether*. Gliders behave like solitons in many regards [53]. However, while optical solitons usually arise due to a balance between nonlinear and linear dispersive effects [54], we have demonstrated optical soliton-like behaviour in a synthetic temporal lattice with only simple binary rules. Despite its simplicity, our system captures physically relevant features since a reversible extension of ECA Rule 54 has produced insightful results in non-equilibrium statistical mechanics and generalized hydrodynamics [55]. By properly programming ECA Rule 54 in our photonic simulator, we experimentally demonstrated a glider collision, shown in Fig. 4.4(b), whereby gliders emerge after the collision with the same shape and velocity but with a phase shift, which is characteristic of soliton collisions [56]. Such glider collisions can be used to construct logic gates [57] and Universal Turing Machines [9] for unconventional computing. Furthermore, we also observed a glider gun, shown in Fig. 4.4(c), in which a higher-order localization produces lower-order gliders akin to the process of soliton fission [58]. Conversely, a glider black hole, shown in Fig. 4.4(d), looks like the process of soliton fusion. Therefore, we have demonstrated a diverse range of glider and soliton interactions in our simple photonic computational platform, which can help unlock new methods

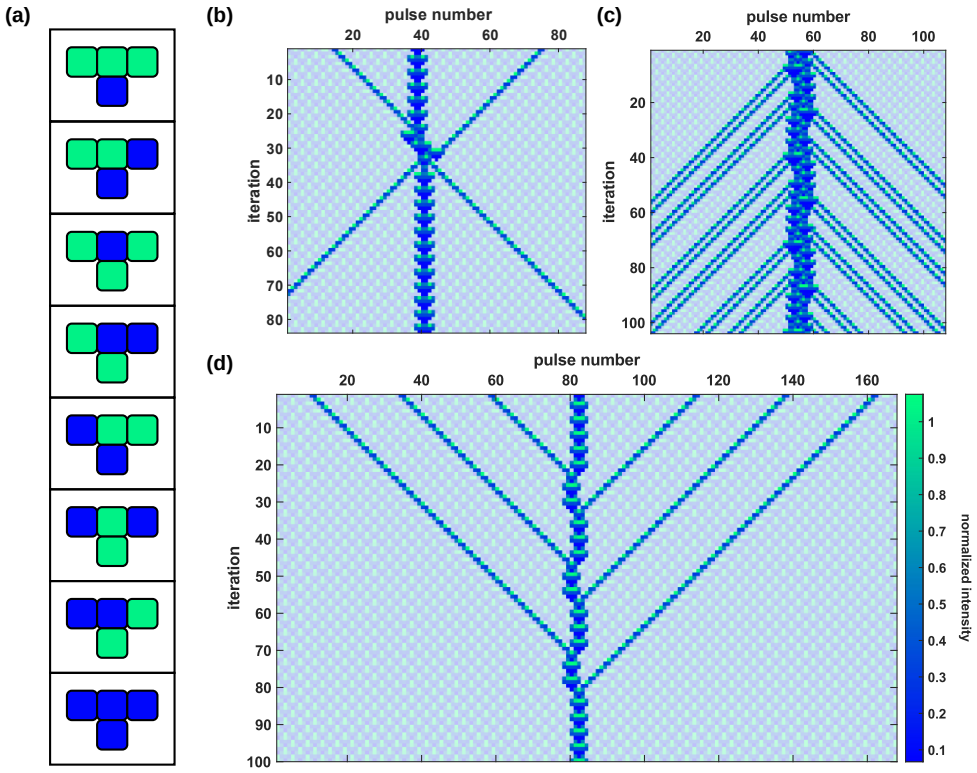

Figure 4.4: **Soliton-like and glider interactions produced by ECA Rule 54 in the photonic hardware.** (a) Truth table showing the update for ECA Rule 54. Space-time diagrams of ECA Rule 54 with periodic background filtered out, showing (b) glider collision, (c) glider gun, and (d) black hole.

of optical information processing.

4.4 Discussion

In our current experiments, we performed the weighted linear summations in the optical domain, and applied nonlinearity digitally for convenience in demonstrating the photonic ECA concept. However, this incurs additional overhead due to the optoelectronic and digital-to-analog conversions performed. This overhead can be reduced by using fully analog electrical circuits, but the photodetector bandwidth still ultimately bottlenecks the clock speed of CA. To overcome these electronic limitations, we propose an integrated all-optical implementation of photonic CA so that the self-evolution of the CA occurs entirely in the optical domain. An example on-chip implementation of photonic CA is shown in Fig. 4.5. This implementation is based on the same time-multiplexed architecture as our current experiment, but it utilizes a monolithic thin-film lithium niobate platform [59] to increase computation speed and energy efficiency by potentially orders of magnitude. As a measure of

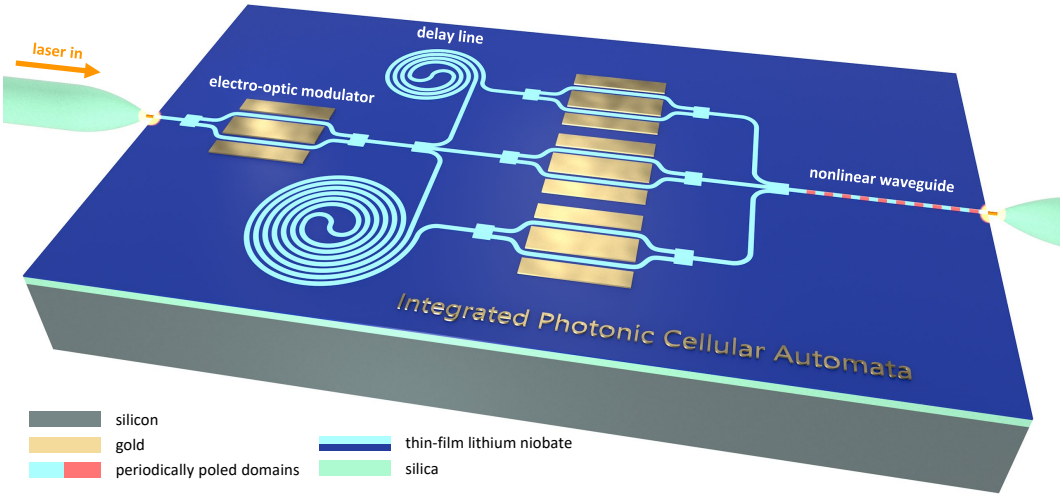

Figure 4.5: **Lithium niobate nanophotonic cellular automata.** The simplicity of the photonic hardware components for simulating complexity can be maximized by on-chip integration with lithium niobate nanophotonic circuits. For example, integrated EOMs offer greater performance and a periodically-poled nonlinear waveguide can enable efficient all-optical thresholding and feedback. This simple nanophotonic circuit can yield orders of magnitude improvement in the speed and energy-efficiency for simulating complexity in CA.

these speed and energy enhancements, we consider the energy-time product, which captures both the energy and speed of the device operation. All-optical rectification and switching operations [60, 61] can be performed using nonlinear waveguides in thin-film lithium niobate with energy-time products on the order of 10^{-27} J $\cdot$ s, which is more than three orders of magnitude better than the kind of optoelectronic thresholding performed in the current experiments. This can be followed by an optical feedback loop to enable terahertz clock rates (Supplementary Information Section 4) that are unattainable by digital electronics. Moreover, the VOAs of our current implementation can be replaced by static integrated EOMs to provide greater control and reproducibility in setting amplitude weights for specific rules. Similarly, other photonic components in our system can be replaced by their simpler and higher performance integrated counterparts.

In summary, we have demonstrated a special-purpose photonic computational platform utilizing a synthetic temporal dimension and simple hardware components capable of simulating a wide range of complex phenomena. Simple rules based on local shift-invariant interactions are used to effectively implement different ECA. Our decentralized and non-von Neumann photonic computer can be programmed

to represent different rules and initial conditions for the light pulses due to the flexibility and rapid reconfigurability afforded by our hardware system. A range of important complex phenomena including fractals, chaos, and solitons are shown on the same hardware. We focused only on 3 ECA rules (90, 30, and 54) that exhibit some representative complex phenomena. However, a slight modification to the present photonic hardware mapping can allow for implementation of all 256 possible ECA rules (see Supplementary Information Table IV). Future work can involve generalizing the time-multiplexed photonic network to exploit spatial or frequency multiplexing techniques [30] for greater synchronous parallel cell updates or implementing other types of CA including filter CA [62], reversible CA [55], and stochastic CA [63]. This can enable the study of experimentally-challenging complex dynamics in kinetic critical phenomena [64], Ising models [65], and lattice Boltzmann models [10]. Furthermore, achieving complexity from simple photonic hardware is an important first step towards harnessing this complexity for efficient and robust artificial intelligence, for example in reservoir computing [26, 27] and deep learning [20, 28]. Our results can inspire a path for special-purpose photonic computers enabling ultrafast low-power operation for critical real-time and edge-computing applications, and new information processing strategies using light.

4.5 Materials and Methods

Experimental Setup

For a more detailed picture of our experimental setup, please see Supplementary Information Fig. 1. A mode-locked laser (MLL) is used that outputs femtosecond optical pulses with a center wavelength of 1550 nm and a repetition period of $T_R = 4$ ns. Then, the pulses are stretched to ~ 5 ps with a 200 GHz Channel 34 filter to reduce the effects of dispersion. After the pulses are stretched, 10% of the power is tapped with a 90:10 optical fiber splitter and sent directly to a 600 MHz-bandwidth photodetector. The RF output of the detector passes through a 300 MHz low pass filter, which isolates the 250 MHz component of the signal. This signal acts as a clock for the FPGA (Zynq UltraScale+ RFSoC), which generates the modulator driving signals for the EOM in the experiment. Deriving the FPGA's clock directly from the optical pulse train eliminates any timing drift between the optical path and electronic signals. The 90% of the optical power that is not used to clock the FPGA is instead sent through two consecutive intensity modulators (IMs). The first IM, converts the uniform input pulse train to a binary string that contains either an initial condition or the previous state of the ECA under study. The second IM,

helps to achieve a better extinction ratio for the zeros in these binary strings. After exiting the modulators, the binary pulse train passes through an erbium-doped fiber amplifer (EDFA) and another 200 GHz Channel 34 filter. Then, pulses are first split between two paths at a 50:50 spliter. One of these paths leads to a second 50:50 splitter, where the pulses are again divided between another two paths. The paths after the second 50:50 splitter are labeled the $\pm 1T_{\mathrm{R}}$ delay lines. The lengths of these lines are chosen to delay advance the pulse train by one repetition period relative to the $0T_{\mathrm{R}}$ delay line, which is the other line after the first 50:50 splitter. The result of delaying and advancing the pulse train in this manner is coherent interference of nearest-neighbour pulses once the delay lines are recombined. To detect the state, the output pulse train passes through another EDFA and 200 GHz Channel 34 filter. The pulses are split at a final 50:50 splitter and the signal is then measured on both a fast 5 GHz photodetector and a slow kHz photodetector. The RF output of the slow detector is sent to the stabilization electronics for the delay lines, whilst the RF output of the fast detector is recorded on an oscilloscope. The optoelectronic signal is thresholded electronically to produce binary states, which are then sent to and stored on the FPGA, which uses a digital-to-analog converter (DAC) to convert the array into an RF pulse pattern for the next ECA iteration.

Experimental Procedure

Setting the desired ECA rule involves adjusting both the relative intensities and phases between the three delay lines. VOAs are used to adjust the intensities in the lines by detuning the coupling in the free space delays shown in Supplementary Information Fig. 1, and the relative phases are set to either 0 or π by changing the feedback signals from the PIDs used to stabilize the $\pm 1T_{\mathrm{R}}$ delay lines. A relative phase of 0 represents constructive interference between two delay lines, and conversely a relative phase of π represents destructive interference. Therefore, the result of the $\pm 1T_R$ delay lines, tuning the VOAs, and setting relative phases can be summarized as:

$$y_i(t) = a_{-1}x_{i-1}(t) + a_0 x_i(t) + a_1 x_{i+1}(t)\ , \tag{4.2}$$

where $x_i(t) \in \mathbb{Z}_2$ is the amplitude of the i^{th} light pulse in the t^{th} iteration before being split into the delay lines, $y_i(t)$ is the amplitude of the light pulse after recombining delay lines, and $\{a_{-1}, a_0, a_1\} \in [-1, 1]$ are the losses set by the VOAs and phases representing constant linear weights. The light pulse amplitude $y_i(t)$ is converted to an intensity $|y_i(t)|^2$ after passing through the photodetector and then optoelectronic

thresholding performs the function:

$$x_i(t+1) = H\left(|y_i(t)|^2 - b\right) , \tag{4.3}$$

where $H(x)$ is the Heaviside step function, $b \in \mathbb{R}$ is the thresholding value, and $x_i(t+1)$ is the output result to be reinjected as the light pulse amplitude for the next iteration. Therefore, any light intensity $|y_i(t)|^2 < b$ represents a dead cell, and conversely any light intensity $|y_i(t)|^2 > b$ represents a live cell. The particular mappings for each ECA rule studied in the Results section is given in Supplementary Information Section 3.

In the present experiments, we considered effectively infinite lattices by padding both sides of the initial conditions with large numbers of zeros that exceeded the number of measured iterations for each experiment. It is also possible to implement other types of boundary conditions such as closed boundaries by imposing constant boundary cell states that do not update, or periodic boundary conditions by turning the 1D lattice into a ring geometry.

4.6 Supplementary Information

Experimental Setup

In Fig. 4.6, we present a detailed schematic of the experiment setup used to realize photonic elementary cellular automata (ECA). The setup may be divided into two segments: the optical segment and the electronic segment. The optical segment is essentially a multimode interferometer that takes a pulse train as its input and that outputs the interference between each pulse and its nearest neighbors. The electronic segment sets and stabilizes the relative phases interferometer arms, which helps us to implement different rules with the same hardware. It also converts the output of the interferometer to a binary signal and uses this binary signal to update the pulse train input into the interferometer.

The optical segment of our experimental setup begins at the mode-locked laser (MLL) shown in Fig. 4.6. The MLL outputs femtosecond optical pulses with a center wavelength of 1550 nm and a repetition period of $T_R = 4$ ns. We stretch the pulses to $\sim$5 ps with a 200 GHz Channel 34 filter to reduce the effects of dispersion.

After stretching the pulses, we tap 10% of the power with a 90:10 splitter and send it directly to a 600 MHz detector. We pass the RF output of the detector through a 300 MHz low pass filter, which isolates the 250 MHz component of the signal. This signal acts as a clock for our system's FPGA, which, as we shall explain shortly,

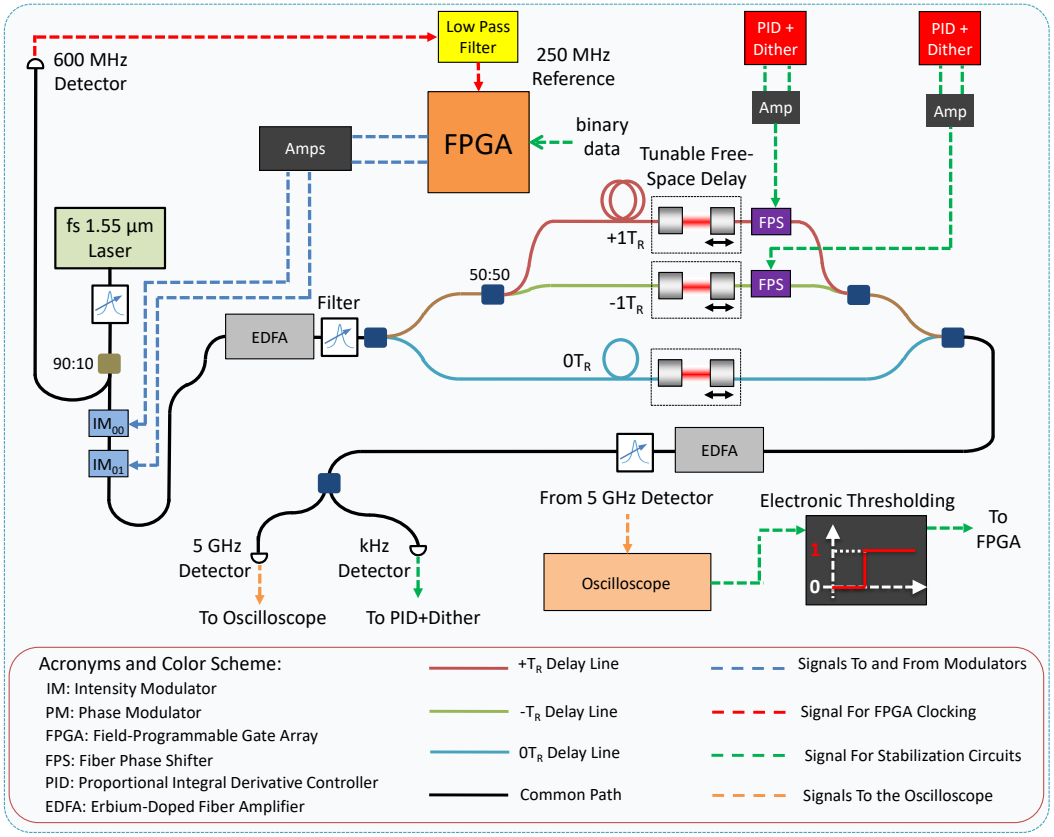

Figure 4.6: **Detailed Schematic of Experimental Setup.**

generates the modulator driving signals for our experiment. Deriving the FPGA's clock directly from the optical pulse train eliminates any timing drift between our optical path and our electronic signals.

The 90% of the optical power that is not used to clock the FPGA is instead sent through two consecutive intensity modulators (IMs). The first IM, IM_{00}, converts the uniform input pulse train to a binary string that contains either an initial condition or the previous state of the ECA under study. The second IM, IM_{01}, helps us to achieve a better extinction ratio for the zeros in these binary strings.

After exiting the modulators, the binary pulse train passes through an erbium-doped fiber amplifer (EDFA) and another 200 GHz Channel 34 filter before finally reaching the interferometer. The pulses are first split between two paths at a 50:50 spliter. One of these paths leads to a second 50:50 splitter, where the pulses are again divided between another two paths. The paths after the second 50:50 splitter are labeled the $\pm 1T_\mathrm{R}$ delay lines. The lengths of these lines are chosen to delay advance the pulse train by one repetition period relative to the $0T_\mathrm{R}$ delay line, which is the

other line after the first 50:50 splitter. The result of delaying and advancing the pulse train in this manner is that we interfere each pulse in the $0T_R$ delay line with its nearest-neighbor pulses once we recombine all of the paths.

The interferometer implements the rule for a particular ECA, so the output of the interferometer is the updated state of the ECA under study. To detect this state, we pass the output pulse train through another EDFA and 200 GHz Channel 34 filter. We then split the pulse train at a final 50:50 splitter and detect the signal on both a fast 5 GHz detector and a slow kHz detector. We send the RF output of the slow detector to the stabilization electronics for the interferometer arms, and we record the RF output of the 5 GHz detector on an oscilloscope. We then transfer the raw data trace from the oscilloscope to our lab computer.

We perform the thresholding operation electronically based on the peak voltage for each pulse time bin, which is necessary to prepare the next input to our ECA. After thresholding, we pass the binary values to the FPGA, which uses a digital-to-analog converter (DAC) to convert the array into an RF pulse pattern. This pulse pattern is amplified and sent to IM_{00} and IM_{01}, where it modulates the pulses from the MLL to produce the next input to the interferometer.

It is important to note that, while it was convenient for our proof-of-concept experiments, a digital computer is not necessary to perform the threshold function for our ECA. Indeed, one could imagine using a carefully calibrated comparator or another analog logic circuit to convert the output of our 5 GHz detector into a binary pulse train that can be fed back to the IMs before the interferometer.

The final components of our setup are the stabilization electronics. We utilize a Pound-Drever-Hall locking scheme [66] to lock the phases of the $\pm 1T_R$ delay lines to the phase of the $0T_R$ delay line. The RF output of our kHz detector is sent to two Red Pitaya STEMLabs, which contain built-in proportional-integral-derivative (PID) controllers. These devices output dither and control signals, which we combine and amplify with a custom printed circuit boards (PCBs). We send the outputs of these PCBs to fiber phase shifters in the $\pm 1T_R$ delay lines to stabilize the phases of the lines.

A key benefit of using PDH locking for our photonic ECA experiments is that it enables us to independently lock each delay line in-phase or out-of-phase with the $0T_R$ delay line. This ability allows us to either add or subtract the fields in the $\pm 1T_R$ delay lines to or from the field in the $0T_R$ delay line, and it is essential to enabling

us to implement multiple ECA rules with the same photonic hardware.

Experimental Procedure

We begin our experiments by configuring the lines of our interferometer to produce the correct rule for the ECA that we would like to study. Generating these rules involves adjusting both the relative intensities and phases between the three lines. We adjust the intensities in the lines by detuning the coupling in the free space delays shown in Fig. 4.6, and we set the relative phases to either 0 or π by changing the feedback signals from the PIDs used to stabilize the $\pm 1 T_R$ delay lines. In Supplementary Sec. 3, we discuss the particular mapping between each ECA studied in the main text and our photonic hardware.

After programming a particular rule into the interferometer, we program our FPGA to launch an initial condition into the system. We detect the response of the interferometer to this initial condition and average over five repetitions of this measurement. We then perform the threshold operation on the averaged data trace. As discussed above, the array of binary values that results from the threshold operation is the input to the next iteration of the ECA. We pass this array to our FPGA, which produces RF pulses to drive IM_{00} and IM_{01} in Fig. 4.6. In turn, these IMs to write the binary string onto the pulse train from our mode-locked laser, and this string corresponds to the updated state of our ECA. We repeat this process of performing the threshold operation on the averaged interferometer output and reinjecting the result of the threshold operation into the interferometer until we have iterated the ECA for the desired number of iterations.

To ensure that the phases of our delay lines are stable during our experiment, we track the output of our interferometer with a real-time simulation of the ECA rule. If the binary array produced by our threshold operation differs from what we would expect in our simulation - even by just a single cell - we rerun that iteration of the ECA until the iteration succeeds. Without this form of error-correction, the system would behave as stochastic ECA. In practice, the number of failed iterations depends on the particular ECA rule being implemented, but the iteration success rate is high for each of the experiments presented in the main text. The success rates are $\sim 100\%$ for Rule 90, $> 81\%$ for Rule 30..., and $> 90\%$ for Rule 54. The iteration success rates for Rule 30 and Rule 54 appear lower than ideal because of a lower interference visibility compared to Rule 90. Note that the individual cell-wise bit error rate is $< 10^{-3}$ for each of the rules, indicating good cell-wise fidelity. The failures that do

occur are due to instability in the delay line phases; power drift in the input pulse train, which can causes pulses that are close to one side of the threshold to cross to the other side; or insufficient averaging over the fluctuations in the recorded traces, which can cause some pulses to fluctuate across the threshold. While this last cause can be alleviated by averaging over additional traces, there is a trade-off between the likelihood that an iteration will fail and the time it takes to run our experiments. The longer we run our experiment, the more susceptible it is to fail because of drift in the input power.

Mapping ECA Rules to Photonic Hardware

Here, we provide an explicit mapping between each of the ECA rules studied in the main text and the paramters of our photonic platform. Setting the desired ECA rule involves adjusting both the relative intensities and phases between the three delay lines. We adjust the intensities in the lines by detuning the coupling in the free space delays shown in Supplementary Information Fig. 1, and we set the relative phases to either 0 or π by changing the feedback signals from the PIDs used to stabilize the $\pm 1T_{\mathrm{R}}$ delay lines. A relative phase of 0 produces constructive interference between two delay lines, and conversely a relative phase of π produces destructive interference. Therefore, the result of the $\pm 1T_R$ delay lines, tuning the VOAs, and setting relative phases can be summarized as:

$$y_i(t) = a_{-1}x_{i-1}(t) + a_0 x_i(t) + a_1 x_{i+1}(t) \,, \tag{4.4}$$

where $x_i(t) \in \mathbb{Z}_2$ is the amplitude of the i^{th} light pulse in the t^{th} iteration before being split into the delay lines, $y_i(t)$ is the amplitude of the light pulse after recombining delay lines, and $\{a_{-1}, a_0, a_1\} \in [-1, 1]$ represent the relative strengths and phases of the different delay lines. The light pulse amplitude $y_i(t)$ is converted to an intensity $|y_i(t)|^2$ after passing through the photodetector and then optoelectronic thresholding performs the function:

$$x_i(t+1) = H\left(|y_i(t)|^2 - b\right) \,, \tag{4.5}$$

where $H(x)$ is the Heaviside step function, $b \in \mathbb{R}$ is the thresholding value, and $x_i(t+1)$ is the output result to be reinjected as the light pulse amplitude in the next iteration. As a result of thresholding, any light intensity $|y_i(t)|^2 < b$ represents a dead cell, and conversely any light intensity $|y_i(t)|^2 > b$ represents a live cell. Examples of possible mappings for each ECA rule are listed below. Note that the set of parameters $\{a_{-1}, a_0, a_1, b\}$ implementing a particular rule in our photonic system are not unique, many different sets of parameters can yield the same rule.

Rule 90

The truth table for ECA Rule 90 is shown in the main text Fig. 1(b). Its equivalent Boolean form is:

$$x_i(t+1) = x_{i-1}(t) \oplus x_{i+1}(t) \ . \tag{4.6}$$

The table below shows how this rule can be reconstructed using the linear interference, photodetection and thresholding in our photonic system.

x_{i-1}	x_i	x_{i+1}	$y_i = -1x_{i-1} + 0x_i + 1x_{i+1}$	$\lvert y_i\rvert^2$	$H(\lvert y_i\rvert^2 - 0.5)$	$\mathbf{x_{i-1} \oplus x_{i+1}}$
0	0	0	0	0	0	**0**
0	0	1	1	1	1	**1**
0	1	0	0	0	0	**0**
0	1	1	1	1	1	**1**
1	0	0	-1	1	1	**1**
1	0	1	0	0	0	**0**
1	1	0	-1	1	1	**1**
1	1	1	0	0	0	**0**

Table 4.1: **Truth table and explicit mapping of ECA Rule 90 to photonic hardware.**

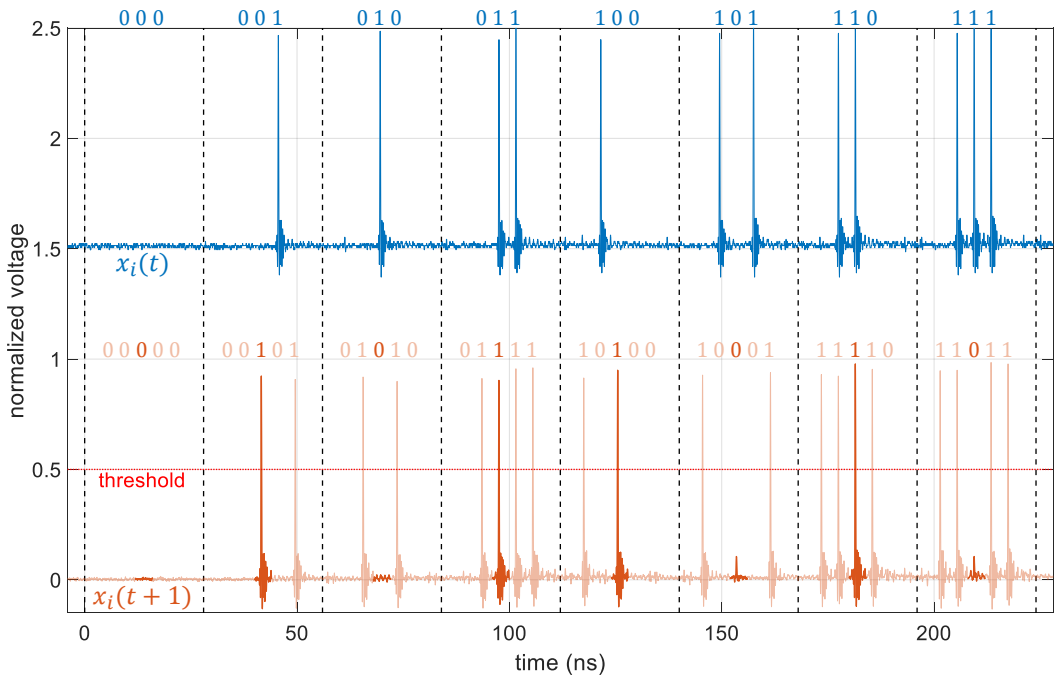

Figure 4.7: **Experimentally computing the truth table for ECA Rule 90.**

Fig. 4.7 shows an experimentally obtained time trace for a single iteration that explicitly computes the truth table for ECA Rule 90 given the 8 possible initial conditions for a 3-cell neighborhood. The initial condition $x_i(t)$ is shown is blue (vertically separated for easier viewing), and the resultant cell state $x_i(t+1)$ is shown below in red. The voltage represents the detected optoelectronic signal $|y_i|^2$. The dotted red line indicates the electronic thresholding value that will binarize the peak voltages in each time bin, which shows that the output value $x_i(t+1) = x_{i-1}(t) \oplus x_{i+1}(t)$ is implemented as intended.

Rule 30

The truth table for ECA Rule 30 is shown in the main text Fig. 3(a). Its equivalent Boolean form is:

$$x_i(t+1) = x_{i-1}(t) \oplus (x_i(t) \vee x_{i+1}(t)) \ . \tag{4.7}$$

The table below shows how this rule can be reconstructed using the linear interference, photodetection and thresholding in our photonic system.

x_{i-1}	x_i	x_{i+1}	$y_i = 1x_{i-1}$ $-0.66x_i$ $-0.66x_{i+1}$	$\lvert y_i\rvert^2$	$H(\lvert y_i\rvert^2 - 0.4)$	$\mathbf{x_{i-1}}$ $\oplus$ $(\mathbf{x_i} \vee \mathbf{x_{i+1}})$
0	0	0	0	0	0	**0**
0	0	1	-0.66	~ 0.44	1	**1**
0	1	0	-0.66	~ 0.44	1	**1**
0	1	1	-1.32	~ 1.74	1	**1**
1	0	0	1	1	1	**1**
1	0	1	0.34	~ 0.12	0	**0**
1	1	0	0.34	~ 0.12	0	**0**
1	1	1	-0.32	~ 0.10	0	**0**

Table 4.2: **Truth table and explicit mapping of ECA Rule 30 to photonic hardware.**

Rule 54

The truth table for ECA Rule 54 is shown in the main text Fig. 4(a). Its equivalent Boolean form is:

$$x_i(t+1) = (x_{i-1}(t) \vee x_{i+1}(t)) \oplus x_i(t) \ . \tag{4.8}$$

The table below shows how this rule can be reconstructed using the linear interference, photodetection and thresholding in our photonic system.

x_{i-1}	x_i	x_{i+1}	$y_i = -0.66x_{i-1}$ $+1x_i$ $-0.66x_{i+1}$	$\lvert y_i\rvert^2$	$H(\lvert y_i\rvert^2 - 0.4)$	$(\mathbf{x_{i-1}} \vee \mathbf{x_{i+1}})$ $\oplus$ $\mathbf{x_i}$
0	0	0	0	0	0	**0**
0	0	1	-0.66	~ 0.44	1	**1**
0	1	0	1	1	1	**1**
0	1	1	0.34	~ 0.12	0	**0**
1	0	0	-0.66	~ 0.44	1	**1**
1	0	1	-1.32	~ 1.74	1	**1**
1	1	0	0.34	~ 0.12	0	**0**
1	1	1	-0.32	~ 0.10	0	**0**

Table 4.3: **Truth table and explicit mapping of ECA Rule 54 to photonic hardware.**

We focused only on 3 ECA rules that exhibited some representative complex phenomena, however, this kind of photonic computer can be used to implement all 256 possible ECA rules. To show this more clearly, we give examples of explicit photonic hardware mappings for all 256 ECA rules in Table. 4.4. Note that the photonic hardware mappings are not unique, there exist many possible mappings for each rule. Only the 88 inequivalent ECA rules are listed since some rules are equivalent up to reflection or complementing (equivalent rules in parenthesis). We slightly modify Eq. 4.4 to allow for the possible addition of a constant bias term $d \in [-1, 1]$, which corresponds to an extra interference path with a constant input:

$$y_i(t) = a_{-1}x_{i-1}(t) + a_0 x_i(t) + a_1 x_{i+1}(t) + d \ , \tag{4.9}$$

and introduce an upper thresholding value $c \in \mathbb{R}$ such that:

$$x_i(t+1) = \begin{cases} 1, & \text{if } b \leq |y_i(t)|^2 \leq c \\ 0, & \text{otherwise} \end{cases} \ . \tag{4.10}$$

The Heaviside step function $H(x)$ used in Eq. 4.5 is just a special case of Eq. 4.10 with $c = \infty$. An upper threshold is needed to implement odd-numbered ECA rules, which will have update rules satisfying $f(0, 0, 0) = 1$. Alternatively, a single threshold value can be used like in Eq. 4.5 if the linear weights are allowed to be complex, $a_i \in \mathbb{C}$, by controlling the full complex amplitude of light.

Rule(s)	a_{-1}	a_0	a_1	b	c	d
0 (255)	0	0	0	0	0	0
1 (127)	1	1	1	0	0.9	0
2 (16, 191, 247)	1	1	0.9	0.1	0.9	0
3 (17, 63, 119)	1	1	0.9	0	0.9	0
4 (223)	1	0.9	1	0.1	0.9	0
5 (95)	1	0.9	1	0	0.9	0
6 (20, 159, 215)	1	0.9	0.9	0.1	0.9	0
7 (21, 31, 87)	1	0.9	0.9	0	0.9	0
8 (64, 239, 253)	1	0.9	0.9	1.1	3.6	0
9 (65, 111, 125)	1	0.9	-0.5	0	0.2	0
10 (80, 175, 245)	1	0.9	-0.4	0.1	0.3	0
11 (47, 81, 117)	1	0.9	-0.1	0	0.8	0
12 (68, 207, 221)	1	0.9	0	0.1	0.9	0
13 (69, 79, 93)	1	0.4	-0.5	0	0.2	0
14 (84, 143, 213)	1	0.5	0.4	0.1	0.9	0
15 (85)	1	0.9	0	0	0.9	0
18 (183)	1	0.9	1	0.9	3.6	0
19 (55)	0.9	1	0.9	0	0.9	0
22 (151)	1	1	1	0.1	3.9	0
23	1	1	1	0	3.9	0
24 (66, 189, 231)	1	0.9	0.9	0.9	3.6	0
25 (61, 67, 103)	0.4	1	-0.9	0	0.2	0
26 (82, 167, 181)	1	0.8	0.9	0.7	3.2	0
27 (39, 53, 83)	1	0.8	0.9	0.1	1.6	-0.5
28 (70, 157, 199)	1	0.9	0.8	0.7	3.2	0
29 (71)	1	0.9	0.8	0.1	1.6	-0.5
30 (86, 135, 149)	1	0.9	0.9	0.1	3.6	0
32 (251)	1	0.9	1	3.7	5	0
33 (123)	1	0.6	-1	0	0.1	0
34 (48, 187, 243)	1	0.9	-0.5	0.2	0.8	0
35 (49, 59, 115)	0.9	1	-0.1	0	0.8	0
36 (219)	1	0.9	-0.1	0.7	0.9	0
37 (91)	1	0.4	-0.9	0	0.2	0
38 (52, 155, 211)	1	0.9	-0.5	0.2	0.9	0
40 (96, 235, 249)	1	1	0.9	1.1	3.9	0
41 (97, 107, 121)	1	1	-0.6	0	0.3	0
42 (112, 171, 241)	1	1	-0.4	0.1	0.9	0
43 (113)	1	1	-0.1	0	0.9	0

Continued on next page. . .

Rule(s)	a_{-1}	a_0	a_1	b	c	d
44 (100, 203, 217)	1	0.9	-0.1	0.1	0.9	0
45 (75, 89, 101)	1	0.8	-0.9	0	0.8	0
46 (116, 139, 209)	1	0.9	-0.4	0.1	0.9	0
50 (179)	0.5	1	0.4	0.1	0.9	0
51	0.9	1	0	0	0.9	0
54 (147)	1	0.9	-0.5	0.2	1.9	0
56 (98, 185, 227)	1	0.9	0.8	0.9	3.6	0
57 (99)	0.8	1	-0.9	0	0.8	0
58 (114, 163, 177)	0.9	1	-0.4	0.1	0.9	0
60 (102, 153, 195)	1	1	0.9	0.9	3.9	0
62 (118, 131, 145)	1	1	0.9	0.1	3.9	0
72 (237)	1	0.9	1	1.1	3.9	0
73 (109)	1	-0.6	1	0	0.3	0
74 (88, 173, 229)	1	-0.1	0.9	0.1	0.9	0
76 (205)	1	-0.4	1	0.1	0.9	0
77	1	-0.1	1	0	0.9	0
78 (92, 141, 197)	1	-0.4	0.9	0.1	0.9	0
90 (165)	1	0.9	1	0.9	3.9	0
94 (133)	1	0.9	1	0.1	3.9	0
104 (233)	1	1	1	1.1	5	0
105	1	1	1	0.1	5	-0.7
106 (120, 169, 225)	1	-0.1	-0.5	0.2	0.9	0
108 (201)	1	-0.4	-0.3	0.1	0.9	0
110 (124, 137, 193)	1	-0.4	-0.4	0.1	0.9	0
122 (161)	1	0.9	1	0.9	5	0
126 (129)	1	1	1	0.1	5	0
128 (254)	1	1	0.2	4.1	5	0
130 (144, 190, 246)	1	0.8	-0.9	0.7	0.9	0
132 (222)	1	0.9	-1	0.1	0.9	0
134 (148, 158, 214)	1	0.8	-0.9	0.1	0.9	0
136 (192, 238, 252)	1	0.4	-0.9	0.2	0.8	0
138 (174, 208, 244)	1	0.4	-0.9	0.2	0.9	0
140 (196, 206, 220)	1	0.6	-1	0.1	0.9	0
142 (212)	1	0.5	-0.9	0.1	0.9	0
146 (182)	1	0.8	-0.9	0.7	3.2	0
150	1	1	-0.6	0.2	3.9	0
152 (188, 194, 230)	1	0.8	-0.7	0.1	0.3	-0.6
154 (166, 180, 210)	1	0.4	-0.9	0.2	1.9	0

Continued on next page. . .

Rule(s)	a_{-1}	a_0	a_1	b	c	d
156 (198)	1	-0.6	0.1	0.2	1.2	0
160 (250)	1	0.5	0.7	2.3	5	0
162 (176, 186, 242)	1	0.4	-0.5	0.2	0.9	0
164 (218)	1	0.8	-0.9	0	0.2	-0.5
168 (224, 234, 248)	1	0.1	-0.6	0.1	0.3	0
170 (240)	1	0.2	-0.6	0.1	0.9	0
172 (202, 216, 228)	1	0.1	-0.9	0.1	0.3	0.3
178	1	0.4	-0.5	0.2	1.9	0
184 (226)	1	0.9	-0.8	0.1	0.3	-0.6
200 (236)	1	-0.3	-0.1	0.1	0.8	0
204	1	-0.4	0	0.1	0.9	0
232	1	1	0.2	1.1	5	0

Table 4.4: **Example mappings of all 256 possible ECA rules** (equivalent rules are listed in parenthesis if they exist).

Scaling to Terahertz Clock Rates

The pulse repetition rate can be increased to achieve potentially terahertz clock rates by using a similar time-multiplexing delay-line technique as in the current experiments. However, instead of interacting neighbouring pulses in the ECA, we can interleave pulses to effectively increase the repetition rate. For example, suppose we want to achieve a 1 THz pulse repetition rate. Then, we can utilize a setup as shown in Fig. 4.8. In the left part of the figure, we illustrate time-multiplexed input

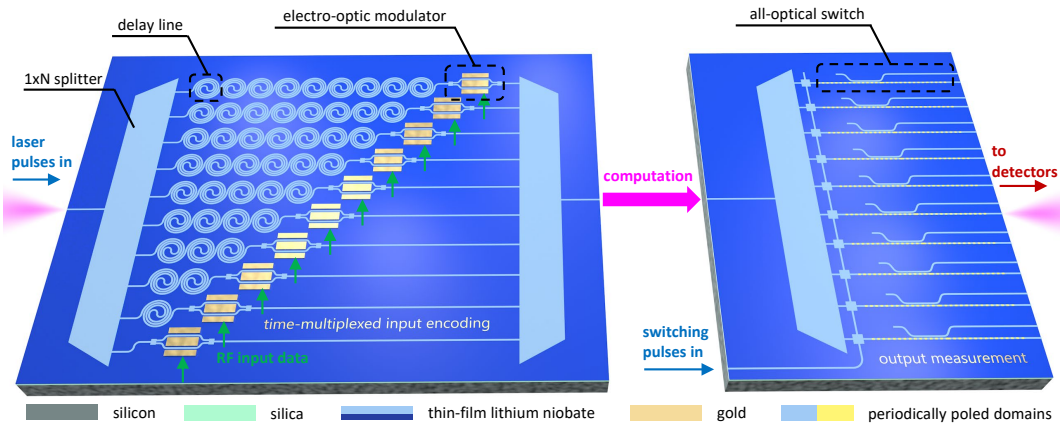

Figure 4.8: **Time-multiplexing method for achieving terahertz pulse repetition rates.**

encoding for the initial condition. Consider a 100 GHz repetition rate femtosecond laser source, but integrated electro-optic modulators (EOMs) that only have 100 GHz bandwidth [67]. We can split the input pulse train into $N = 10$ delay lines, each offset by a relative delay of 10 ps and then individually modulate each of these delay lines with a 100 GHz EOM. Then, upon recombination, we have effectively produced a pulse train with a 1 THz repetition rate. This initial condition can then be used as the input to our proposed all-optical ECA in which the nonlinear waveguides for all-optical thresholding have optical bandwidth > 10 THz. On the other hand, electronic thresholding and feedback is limited to the electronic bandwidth of < 10 GHz, which cannot support the ultrafast repetition rates attainable by all-optical methods. In the right part of the figure, we present a scheme to measure this 1 THz repetition rate pulse train for outputs by selectively switching the pulse train into 10 sets of 100 GHz repetition rate pulse trains using a set of 10 all-optical switches [61]. These 100 GHz repetition rate channels are then amenable to direct photodetection. Therefore, we see that the repetition rate of the laser source is not the limiting factor since we can always employ time-multiplexing methods to achieve any desired pulse repetition rate.

Rule 54 space-time diagrams without filtering

We applied tiling theory to filter out the periodic background, i.e., the *ether*, from the space-time diagrams of ECA Rule 54 shown in the main text Fig. 4. For completeness, we also show the same space-time diagrams below but without filtering.

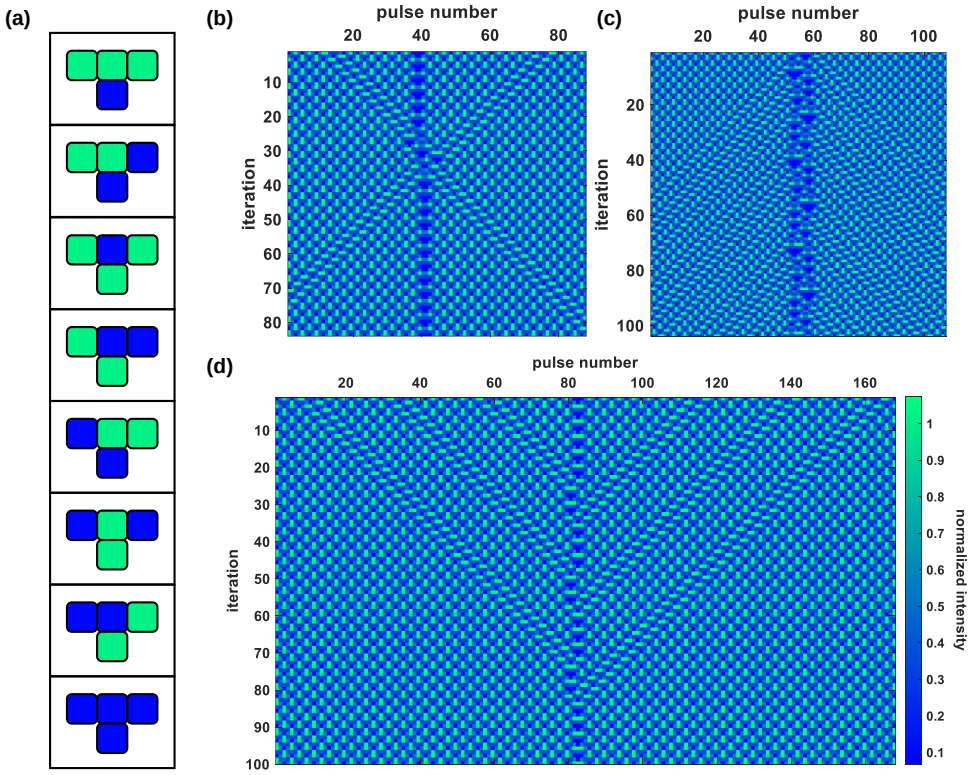

Figure 4.9: **Soliton-like and glider interactions produced by ECA Rule 54 in the photonic hardware.** (a) Truth table showing the update for ECA Rule 54. Space-time diagrams of ECA Rule 54 *without* periodic background filtered out, showing (b) glider collision, (c) glider gun, and (d) black hole.

References

[1] Eric Bonabeau, Guy Theraulaz, Jean-Louls Deneubourg, Serge Aron, and Scott Camazine. Self-organization in social insects. *Trends in Ecology & Evolution*, 12(5):188–193, 1997.

[2] Stephen Wolfram. Cellular automata as models of complexity. *Nature*, 311 (5985):419–424, 1984.

[3] John von Neumann and Arthur W. Burks. *Theory of Self-Reproducing Automata*. University of Illinois Press, Urbana, IL, 1966.

[4] Martin Gardner. The fantastic combinations of john conway's new solitaire game "life". *Scientific American*, 223:20–123, 1970.

[5] Stephen Wolfram. Statistical mechanics of cellular automata. *Reviews of Modern Physics*, 55(3):601, 1983.

[6] Stephen Wolfram. Random sequence generation by cellular automata. *Advances in Applied Mathematics*, 7(2):123–169, 1986.

[7] Per Bak, Chao Tang, and Kurt Wiesenfeld. Self-organized criticality: An explanation of the 1/f noise. *Physical Review Letters*, 59(4):381, 1987.

[8] Robert Balzer. An 8-state minimal time solution to the firing squad synchronization problem. *Information and Control*, 10(1):22–42, 1967.

[9] Matthew Cook et al. Universality in elementary cellular automata. *Complex Systems*, 15(1):1–40, 2004.

[10] Dieter A Wolf-Gladrow. *Lattice-gas cellular automata and lattice Boltzmann models: An introduction*. Springer, 2004.

[11] B Chopard and M Droz. Cellular automata model for the diffusion equation. *Journal of Statistical Physics*, 64(3):859–892, 1991.

[12] Martin Gerhardt and Heike Schuster. A cellular automaton describing the formation of spatially ordered structures in chemical systems. *Physica D: Nonlinear Phenomena*, 36(3):209–221, 1989.

[13] Peter Ashwin, Prasad Patnaik, and C. David Wright. Fast simulation of phase-change processes in chalcogenide alloys using a gillespie-type cellular automata approach. *Journal of Applied Physics*, 104(8):084901, 2008.

[14] Dierk Raabe. Cellular automata in materials science with particular reference to recrystallization simulation. *Annual Review of Materials Research*, 32(1): 53–76, 2002.

[15] G. Bard Ermentrout and Leah Edelstein-Keshet. Cellular automata approaches to biological modeling. *Journal of Theoretical Biology*, 160(1):97–133, 1993.

[16] Stephen Wolfram. Cryptography with cellular automata. In *Conference on the Theory and Application of Cryptographic Techniques*, pages 429–432. Springer, 1985.

[17] Olu Lafe. Data compression and encryption using cellular automata transforms. *Engineering Applications of Artificial Intelligence*, 10(6):581–591, 1997.

[18] Dipanwita Roy Chowdhury, Saugata Basu, Idranil Sen Gupta, and Parimal Pal Chaudhuri. Design of caecc-cellular automata based error correcting code. *IEEE Transactions on Computers*, 43(6):759–764, 1994.

[19] Bing-Hong Wang, Yvonne-Roamy Kwong, and Pak-Ming Hui. Statistical mechanical approach to fukui-ishibashi traffic flow models. *Physical Review E*, 57(3):2568, 1998.

[20] Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, and Michael Levin. Growing neural cellular automata. *Distill*, 5(2):e23, 2020.

[21] Navot Israeli and Nigel Goldenfeld. Computational irreducibility and the predictability of complex physical systems. *Physical Review Letters*, 92(7): 074105, 2004.

[22] Paul W K Rothemund, Nick Papadakis, Erik Winfree, and Anne Condon. Algorithmic self-assembly of dna sierpinski triangles. *PLoS biology*, 2(12): e424, 2004.

[23] Alexandra Imre, Gyorgy Csaba, L. Ji, Alexei Orlov, Gary H. Bernstein, and Wolfgang Porod. Majority logic gate for magnetic quantum-dot cellular automata. *Science*, 311(5758):205–208, 2006.

[24] Makoto Itoh and Leon Chua. Memristor cellular automata and memristor discrete-time cellular neural networks. In *Handbook of Memristor Networks*, pages 1289–1361. Springer, 2019.

[25] Tomohiro Shirakawa, Hiroshi Sato, and Shinji Ishiguro. Construction of living cellular automata using the physarum plasmodium. *International Journal of General Systems*, 44(3):292–304, 2015.

[26] Ozgur Yilmaz. Reservoir computing using cellular automata. *arXiv preprint arXiv:1410.0162*, 2014.

[27] Stefano Nichele and Andreas Molund. Deep reservoir computing using cellular automata. *arXiv preprint arXiv:1703.02806*, 2017.

[28] Ettore Randazzo, Alexander Mordvintsev, Eyvind Niklasson, Michael Levin, and Sam Greydanus. Self-classifying mnist digits. *Distill*, 5(8):e00027–002, 2020.

[29] Stephen Wolfram. Universality and complexity in cellular automata. *Physica D: Nonlinear Phenomena*, 10(1-2):1–35, 1984.

[30] Luqi Yuan, Qian Lin, Meng Xiao, and Shanhui Fan. Synthetic dimension in photonics. *Optica*, 5(11):1396–1405, 2018.

[31] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[32] Laurent Larger, Miguel C. Soriano, Daniel Brunner, Lennert Appeltant, Jose M. Gutiérrez, Luis Pesquera, Claudio R. Mirasso, and Ingo Fischer. Photonic information processing beyond turing: an optoelectronic implementation of reservoir computing. *Optics Express*, 20(3):3241–3249, 2012.

[33] Alireza Marandi, Zhe Wang, Kenta Takata, Robert L Byer, and Yoshihisa Yamamoto. Network of time-multiplexed optical parametric oscillators as a coherent ising machine. *Nature Photonics*, 8(12):937–942, 2014.

[34] Christian Leefmans, Avik Dutt, James Williams, Luqi Yuan, Midya Parto, Franco Nori, Shanhui Fan, and Alireza Marandi. Topological dissipation in a time-multiplexed photonic resonator network. *Nature Physics*, 18(4):442–449, 2022.

[35] Yanne K. Chembo, Daniel Brunner, Maxime Jacquot, and Laurent Larger. Optoelectronic oscillators with time-delayed feedback. *Reviews of Modern Physics*, 91(3):035006, 2019.

[36] Jun Li and Martin Ostoja-Starzewski. Edges of saturn's rings are fractal. *SpringerPlus*, 4(1):1–8, 2015.

[37] Kenneth G. Libbrecht. The physics of snow crystals. *Reports on Progress in Physics*, 68(4):855, 2005.

[38] Paul G. Okubo and Keiiti Aki. Fractal geometry in the san andreas fault system. *Journal of Geophysical Research: Solid Earth*, 92(B1):345–355, 1987.

[39] Xiao-Yun Xu, Xiao-Wei Wang, Dan-Yang Chen, C. Morais Smith, and Xian-Min Jin. Quantum transport in fractal networks. *Nature Photonics*, 15(9): 703–710, 2021.

[40] Zhaoju Yang, Eran Lustig, Yaakov Lumer, and Mordechai Segev. Photonic floquet topological insulators in a fractal lattice. *Light: Science & Applications*, 9(1):1–7, 2020.

[41] Tobias Biesenthal, Lukas J. Maczewsky, Zhaoju Yang, Mark Kremer, Mordechai Segev, Alexander Szameit, and Matthias Heinrich. Fractal photonic topological insulators. *Science*, 376(6597):1114–1119, 2022.

[42] Mordechai Segev, Marin Soljačić, and John M. Dudley. Fractal optics and beyond. *Nature Photonics*, 6(4):209–210, 2012.

[43] Olivier Martin, Andrew M. Odlyzko, and Stephen Wolfram. Algebraic properties of cellular automata. *Communications in Mathematical Physics*, 93(2): 219–258, 1984.

[44] Alireza Marandi, Nick C. Leindecker, Konstantin L. Vodopyanov, and Robert L. Byer. All-optical quantum random bit generation from intrinsically binary phase of parametric oscillators. *Optics Express*, 20(17):19322–19330, 2012.

[45] Igor Reidler, Yaara Aviad, Michael Rosenbluh, and Ido Kanter. Ultrahigh-speed random number generation based on a chaotic semiconductor laser. *Physical Review Letters*, 103(2):024102, 2009.

[46] André Stefanov, Nicolas Gisin, Olivier Guinnard, Laurent Guinnard, and Hugo Zbinden. Optical quantum random number generator. *Journal of Modern Optics*, 47(4):595–598, 2000.

[47] Harald Niederreiter. Quasi-monte carlo methods and pseudo-random numbers. *Bulletin of the American Mathematical Society*, 84(6):957–1041, 1978.

[48] Rainer A. Rueppel. *Analysis and design of stream ciphers*. Springer Science & Business Media, 2012.

[49] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2014.

[50] Maria Madjarova, Mitsugu Kakuta, Masahiro Yamaguchi, and Nagaaki Ohyama. Optical implementation of the stream cipher based on the irreversible cellular automata algorithm. *Optics Letters*, 22(21):1624–1626, 1997.

[51] Stephen Wolfram. *A New Kind of Science*. Wolfram Media, Champaign, IL, 2002.

[52] Escuela Superior de Cómputo. Complete characterization of structure of rule 54. *Complex Systems*, 23(3):259–293, 2014.

[53] Genaro J Martínez, Andrew Adamatzky, Fangyue Chen, and Leon Chua. On soliton collisions between localizations in complex elementary cellular automata: Rules 54 and 110 and beyond. *Complex Systems*, 21(2), 2012.

[54] Govind P. Agrawal. Nonlinear fiber optics. In *Nonlinear Science at the Dawn of the 21st Century*, pages 195–211. Springer, 2000.

[55] Berislav Buča, Katja Klobas, and Tomaž Prosen. Rule 54: Exactly solvable model of nonequilibrium statistical mechanics. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(7):074001, 2021.

[56] Norman J. Zabusky and Martin D. Kruskal. Interaction of" solitons" in a collisionless plasma and the recurrence of initial states. *Physical Review Letters*, 15(6):240, 1965.

[57] Genaro Juárez Martínez, Andrew Adamatzky, and Harold V. McIntosh. Phenomenology of glider collisions in cellular automaton rule 54 and associated logical gates. *Chaos, Solitons & Fractals*, 28(1):100–111, 2006.

[58] Song Wang, Xiao-yan Tang, and Sen-Yue Lou. Soliton fission and fusion: Burgers equation and Sharma–Tasso–Olver equation. *Chaos, Solitons & Fractals*, 21(1):231–239, 2004.

[59] Di Zhu, Linbo Shao, Mengjie Yu, Rebecca Cheng, Boris Desiatov, C.J. Xin, Yaowen Hu, Jeffrey Holzgrafe, Soumya Ghosh, Amirhassan Shams-Ansari, et al. Integrated photonics on thin-film lithium niobate. *Advances in Optics and Photonics*, 13(2):242–352, 2021.

[60] Gordon H.Y. Li, Ryoto Sekine, Rajveer Nehra, Robert M. Gray, Luis Ledezma, Qiushi Guo, and Alireza Marandi. All-optical ultrafast relu function for energy-efficient nanophotonic deep learning. *Nanophotonics*, 2023.

[61] Qiushi Guo, Ryoto Sekine, Luis Ledezma, Rajveer Nehra, Devin J. Dean, Arkadev Roy, Robert M. Gray, Saman Jahani, and Alireza Marandi. Femtojoule femtosecond all-optical switching in lithium niobate nanophotonics. *Nature Photonics*, 16(9):625–631, 2022.

[62] James K. Park, Kenneth Steiglitz, and William P. Thurston. Soliton-like behavior in automata. *Physica D: Nonlinear Phenomena*, 19(3):423–432, 1986.

[63] Alexander V. Goltsev, Fernao V. De Abreu, Sergey N. Dorogovtsev, and Jose F.F. Mendes. Stochastic cellular automata model of neural networks. *Physical Review E*, 81(6):061921, 2010.

[64] Peter Grassberger, Friedrich Krause, and Tassilo von der Twer. A new type of kinetic critical phenomenon. *Journal of Physics A: Mathematical and General*, 17(3):L105, 1984.

[65] Eytan Domany and Wolfgang Kinzel. Equivalence of cellular automata to ising models and directed percolation. *Physical Review Letters*, 53(4):311, 1984.

[66] Eric D. Black. An introduction to Pound–Drever–Hall laser frequency stabilization. *American Journal of Physics*, 69(1):79–87, 2001.

[67] Cheng Wang, Mian Zhang, Xi Chen, Maxime Bertrand, Amirhassan Shams-Ansari, Sethumadhavan Chandrasekhar, Peter Winzer, and Marko Lončar. Integrated lithium niobate electro-optic modulators operating at cmos-compatible voltages. *Nature*, 562(7725):101–104, 2018.

Chapter 5

DEEP LEARNING WITH PHOTONIC NEURAL CELLULAR AUTOMATA

Gordon H.Y. Li, Christian R. Leefmans, James Williams, Robert M. Gray, Midya Parto, and Alireza Marandi. Deep learning with photonic neural cellular automata. *Light: Science & Applications*, 13(1):283, 2024. doi:10.1038/s41377-024-01651-7.

G.H.Y.L. conceived the project, developed the theory, executed numerical simulations, designed the experiments, built the experimental setup, performed the experimental measurements, analyzed the data, and wrote the manuscript.

5.1 Abstract

Rapid advancements in deep learning over the past decade have fueled an insatiable demand for efficient and scalable hardware. Photonics offers a promising solution by leveraging the unique properties of light. However, conventional neural network architectures, which typically require dense programmable connections, pose several practical challenges for photonic realizations. To overcome these limitations, we propose and experimentally demonstrate Photonic Neural Cellular Automata (PNCA) for photonic deep learning with sparse connectivity. PNCA harnesses the speed and interconnectivity of photonics, as well as the self-organizing nature of cellular automata through local interactions to achieve robust, reliable, and efficient processing. We utilize linear light interference and parametric nonlinear optics for all-optical computations in a time-multiplexed photonic network to experimentally perform self-organized image classification. We demonstrate binary (two-class) classification of images using as few as 3 programmable photonic parameters, achieving high experimental accuracy with the ability to also recognize out-of-distribution data. The proposed PNCA approach can be adapted to a wide range of existing photonic hardware and provides a compelling alternative to conventional photonic neural networks by maximizing the advantages of light-based computing whilst mitigating their practical challenges. Our results showcase the potential of PNCA in advancing photonic deep learning and highlights a path for next-generation photonic computers.

5.2 Introduction

Deep learning models have demonstrated remarkable capabilities in numerous domains, ranging from computer vision to natural language processing, scientific discovery, and generative art [1–4]. However, as the complexity and scale of these models continue to surge, a critical challenge emerges: the need for efficient and scalable hardware solutions to handle the ever-increasing computational demands. For example, recent trends show that the compute requirements for deep learning models are doubling approximately every 5-6 months [5]. This is far outpacing improvements in conventional digital electronic computers, which has spurred the use of application-specific hardware accelerators such as Graphics Processing Units and Tensor Processing Units [6]. In this context, the convergence of deep learning with photonics has emerged as a promising frontier, poised to redefine the landscape of neural network computation. By leveraging the distinct characteristics of light, photonic hardware can unlock unprecedented processing speeds, parallelism, and energy efficiencies that surpass the capabilities of traditional electronic architectures [7, 8]. To enable this new paradigm of photonic deep learning, much of the focus so far has been on developing the fundamental devices needed for crucial neural network operations. Indeed, there have been impressive demonstrations of photonics for linear operations such as matrix multiplication and convolutions [9–11], as well as nonlinear activation functions such as rectified linear unit [12–14]. These photonic building blocks are now comparable to or surpass their electronic counterparts in certain important computing metrics.

However, studying system-level architectures for photonic neural networks (PNNs) beyond single devices is also of vital importance. This is crucial since photonics and electronics operate in entirely different regimes [15]. The computational advantages of photonic building blocks can quickly diminish when used to implement conventional neural network architectures that were optimized for digital electronics [16]. Advancing photonic deep learning towards end-to-end and scalable photonic systems requires properly considering neural network architectures that can benefit from implementation with specific photonic hardware. One important hurdle is that conventional deep learning architectures such as Multi-layer Perceptrons (MLPs) and Convolutional Neural Networks (CNNs), which have so far been mainstays for PNNs, require densely-connected layers with large numbers of parameters, which are challenging to realize in typical photonic platforms and current demonstrations of PNNs. For example, integrated PNNs can possess fast input update rates (> 1 GHz) but feature a small number of programmable parameters ($< 10^3$) [9, 10, 14],

whereas free-space PNNs can contain a large number of parameters ($> 10^6$) but have slow input update rates (< 10 kHz) [17–19]. Finally, PNNs are usually operated with fixed weights that cannot be rapidly updated in real-time. This constraint makes it difficult for PNNs to efficiently implement the complex structures of modern deep learning models, and also poses reliability concerns when generalizing to out-of-distribution data.

To overcome these apparent disparities between photonics capabilities and conventional neural network architectures, we propose and experimentally demonstrate a novel type of PNN based on Neural Cellular Automata (NCA) [20]. Cellular automata (CA) are computational models composed of a lattice of cells with states that follow an update rule, which defines how the state of a cell evolves over time based on the states of its neighboring cells (Fig. 5.1a) [21, 22]. Inspired by biological systems, the local interactions between cells governed by the update rule gives rise to complex phenomena and emergent patterns at the global-scale [23] (Supplementary Information Section I). Unlike conventional human-designed update rules, NCA (Fig. 5.1b) harness the complex dynamics of cellular automata by using modern deep learning techniques to learn the local update rules needed to perform specific tasks such as regenerating patterns [20], self-classifying images [24], and texture generation [25]. Our Photonic Neural Cellular Automata (PNCA) combines the advantages of photonic hardware with NCA to achieve self-organized image classification (Fig. 5.1c). The PNCA leverages a completely different methodology for computer vision tasks compared to previous PNNs based on MLPs or CNNs. This enables noise-robust processing, as well as convenient measures of uncertainty for identifying anomalies and out-of-distribution data. Furthermore, PNCA achieves parameter-efficient solutions since the photonic hardware can operate with fixed weights and only needs to encode the parameters for local update rules instead of global network weights. The proposed PNCA approach can be generalized to suit a wide variety of existing photonic hardware, which can potentially greatly increase the functionality of PNNs and addresses several important challenges facing photonic deep learning.

5.3 Results

PNCA architecture

The key concepts of the general PNCA architecture are shown in Fig. 5.1, which can be adapted to suit a wide range of different photonic hardware platforms (e.g., see Supplementary Information Section II). For computer vision tasks, each pixel in the input image corresponds to a cell in the PNCA. Cells are designated as either *alive*

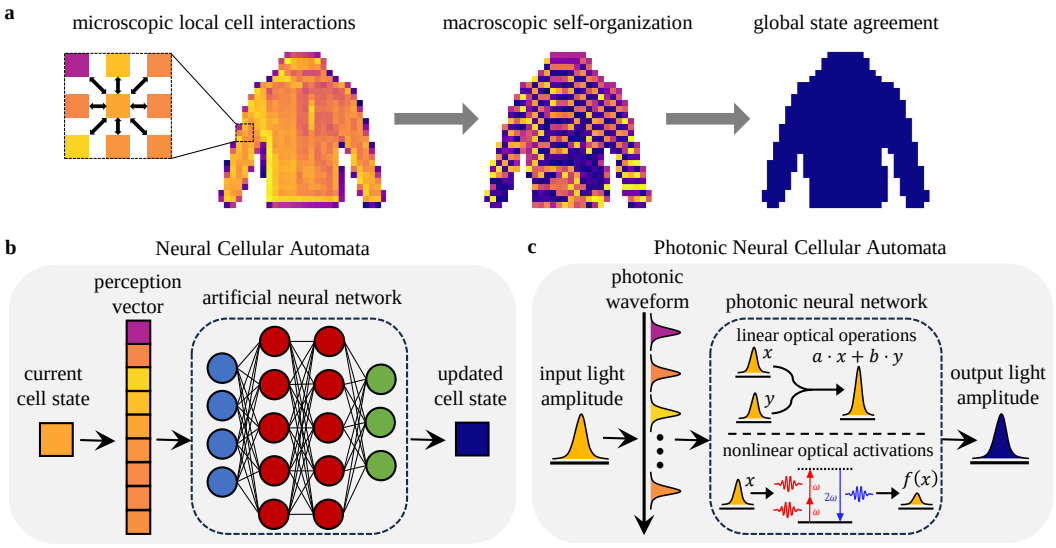

Figure 5.1: **Introduction to PNCA.** (a) Cellular Automata (CA) consist of computational units called cells, which update states according to interactions with neighboring cells. These microscopic local cell interactions can lead to emergent phenomena such as self-organization at the macroscopic scale, and even a global state agreement. (b) Neural Cellular Automata (NCA) encode the local update rules for CA using artificial neural networks and can be trained using modern deep learning techniques to perform tasks, such as image classification through collective agreement of cells. (c) Photonic Neural Cellular Automata (PNCA) directly implement NCA in physical systems by harnessing the speed and interconnectivity of analog photonic hardware, which includes linear operations via light interference and nonlinear activations via nonlinear optics. This endows photonic neural networks with the robust, reliable, and efficient information processing capabilities of NCA, hence overcoming several practical challenges facing light-based computing.

or *dead* through an alive masking procedure. This can be done by setting a threshold for the initial pixel value, below which the cell is considered dead. Only alive cells are actively updated by the PNCA, whereas dead cells can influence the updates of alive cells but are otherwise quiescent. The cell state updates according to a rule that depends on the cells in a local m-cell neighborhood. For example, Fig. 5.1a shows the prototypical Moore neighborhood composed of the cell and the 8 cells that surround it. Other types of local cell neighborhoods are also possible. In the PNCA, the optical field corresponding to each cell is split into m optical paths to define the desired m-cell neighborhood for the local update rule. The local update rule for the PNCA is encoded by the photonic hardware (Fig. 5.1c), which accepts the m inputs given by the m-cell neighborhood and outputs the next cell state. Although Fig. 5.1a only shows each cell state having a single channel, this can also be extended to multiple

channels (e.g., RGB color image channels) by increasing the inputs and outputs accordingly. In general, the programmable photonic hardware contains feed-forward layers with linear operations which can be implemented through meshes of Mach-Zehnder interferometers [9], photonic cross-bar arrays [10], micro-ring resonator weight banks [26], or other linear photonic devices [11, 14]. In addition, there must also be layers performing nonlinear activations such as photonic devices based on optoelectronic measurement-feedback [14, 27] or nonlinear-optical crystals [12, 13]. This kind of feed-forward programmable photonic hardware specifying a single input-output function has been used in previous PNNs. However, for PNCA, the key difference is that the photonic hardware only needs sparse connections and enough parameters to encode for the local update rule, which is usually orders-of-magnitude fewer than the number of parameters needed to encode global network weights in fully-connected layers for MLPs or CNNs. In other words, the parameter-efficient PNCA architecture can enable existing PNN hardware with relatively few parameters to perform larger and more complicated tasks than otherwise possible in conventional neural network architectures. Furthermore, this local update rule can more easily tolerate the use of fixed-weights after training since every cell follows the same update rule. Note that the weights/parameters encoding the local update rule for cells do not vary across cell index or time step iteration, which avoids the need for costly parameter updates in photonic hardware. Finally, the output is recurrently fed back to update the cell state for the next iteration. This can be accomplished by photodetection and electro-optic feedback or by using all-optical feedback lines (e.g., see Supplementary Information Section IV).

Unlike conventional CA with discrete cell states [21], NCA use cell states that are continuous-valued [20], which allows the model to be end-to-end differentiable and compatible with gradient-descent based learning algorithms. In this work, we consider the task of self-organized image classification. The target output after the final iteration is to have every alive cell in the state that corresponds to the class label for the input image. The alive cells must form this collective agreement through only the local interactions defined by repeated iteration of the update rule. This can be interpreted as a kind of recurrent neural network, which can be trained (Fig. S3) using the standard backpropagation-though-time algorithm [28]. Using a cell-wise L_2 loss was found to give better performance compared to cross-entropy loss of labels, which is more commonly used for image classification tasks [20]. The training can either be done *in situ* by performing the forward pass in PNCA to more accurately capture the physics, or completely digitally by simulating the photonic

hardware with noise [29, 30].

Experimental realization of PNCA

We used a time-multiplexed scheme and commercially-available optical-fiber components to experimentally demonstrate proof-of-concept for a simple version of PNCA as shown in Fig. 5.2. Each cell state is given by the amplitude of a laser light pulse generated by a mode-locked laser with a fixed repetition rate such that the cells are inputted one at a time in a flattened 1D lattice by raster scanning across the 2D image. In this way, each cell occupies a time-bin site in a synthetic temporal dimension [31]. Therefore, distances in a real-space lattice correspond to time-differences in the temporal dimension and cells at different lattice sites can be made to interact by using temporal delay lines. The pulse amplitude/phase representing the cell state is set using an electro-optic modulator (EOM), and the pulse is then split between 3 temporal optical delay lines with relative delays T_1 and T_2 chosen to enforce the desired 3-cell local neighborhood shown in Fig. 5.2b. In this simple example, the local update rule is encoded by a single perceptron neuron shown in Fig. 5.2c, which consists of a linear dot product followed by a nonlinear activation function. The dot product is achieved by coherent interference of the optical delay lines, each equipped with a variable optical attenuator (VOA) to program the desired weights, which can be either positive (in-phase/constructive interference) or negative (out-of-phase/destructive interference). The nonlinear activation is performed using pump-depleted second harmonic generation (see Supplementary Information Section VI) in a reverse-proton exchange periodically-poled lithium niobate waveguide [32]. This produces a sigmoid-like function as shown in Fig. 5.2d. Thus, the computations in the local update rule are achieved all-optically. Overall, the local update rule contains only 3 programmable parameters, but can still perform complex tasks. Finally, the cell state is measured using a photodetector, stored on a field-programmable gate array (FPGA), and electro-optically re-injected for the next iteration after alive-cell masking.

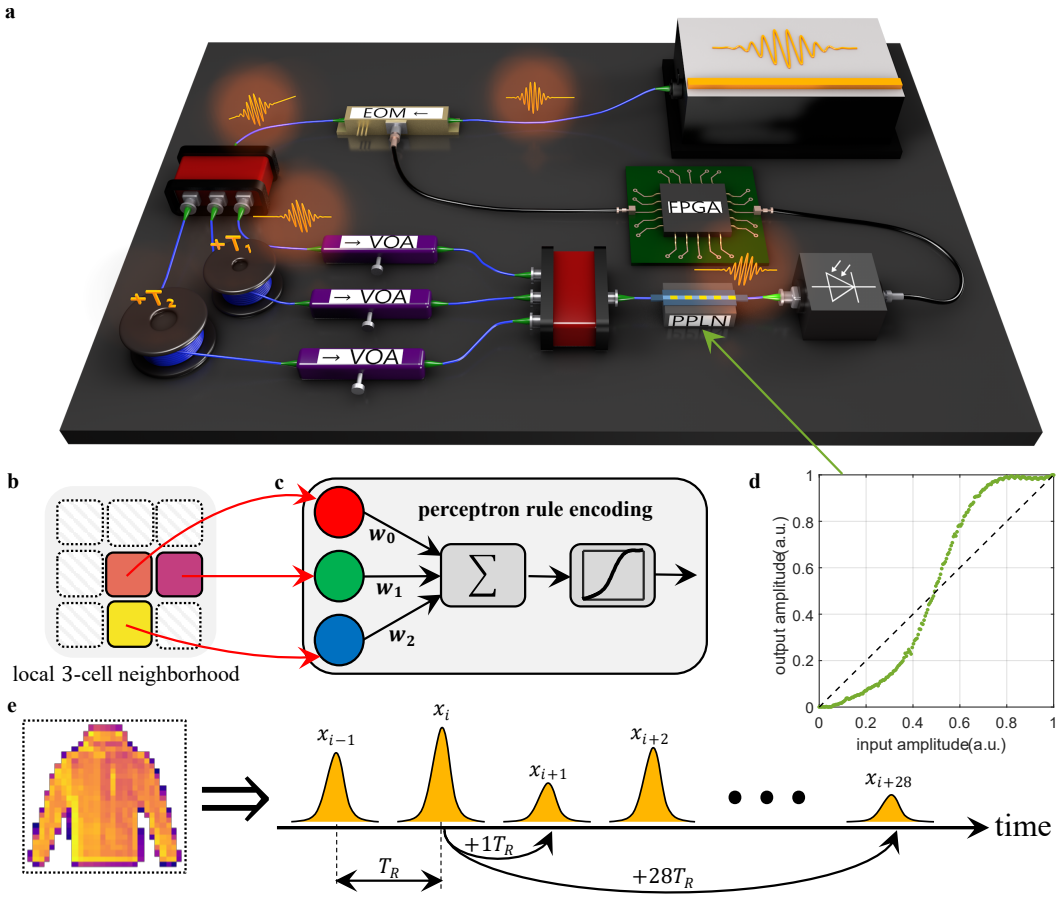

Figure 5.2: **Experimental setup for PNCA.** (a) Schematic of the experimental setup. Pulses of light produced by a mode-locked laser pass through an electro-optic modulator (EOM) and are split into optical fiber delay lines (blue lines) with relative delays T_1 and T_2. Linear dot product weights are programmed by tuning the variable optical attenuator (VOA) in each delay line. Nonlinear activation using a periodically-poled lithium niobate (PPLN) waveguide is performed following the coherent interference of light pulses, with the resultant amplitudes stored on a field-programmable gate array (FPGA) and reinjected (black lines) to drive the input EOM for the next iteration. (b) Local 3-cell neighborhood enforced by relative delays T_1 and T_2. (c) The local update rule is encoded by a single perceptron with 3 programmable parameters. (d) PPLN nonlinear activation function. (e) Cells representing pixels of an image are encoded by the amplitude of light pulses with repetition period T_R in a synthetic temporal dimension. For example, pulses can be coupled using optical delay lines with $T_1 = +1T_R$ and $T_2 = +28T_R$ to implement the local 3-cell neighborhood shown in (b) for fashion-MNIST images.

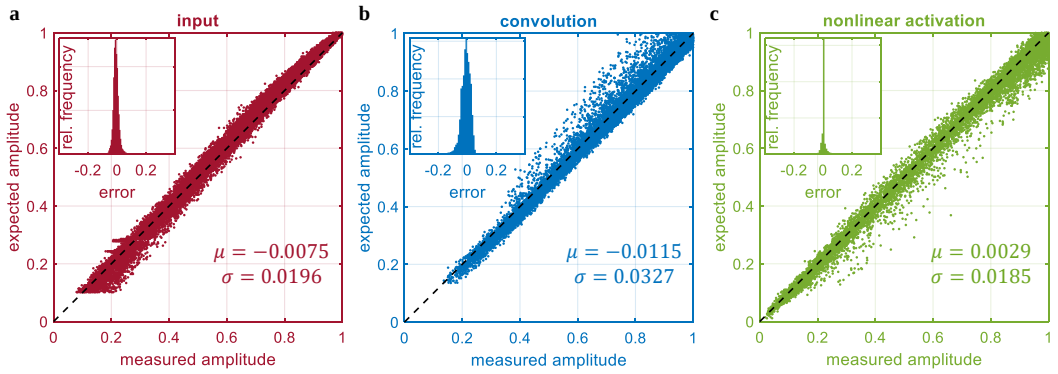

Figure 5.3: **Measurements of noise and errors in PNCA operations.** Expected vs. measured light amplitude for (a) input cell state by EOM, (b) linear dot product by coherent interference and (c) nonlinear activation by PPLN. Each scatter point represents an alive cell from the 200 images tested. The top right insets show the histograms for the error (expected amplitude — measured amplitude) in each case and the bottom right shows the mean and standard deviation, respectively.

A crucial aspect of photonic hardware is that it is analog and noisy. A key advantage of the PNCA architecture is that it is robust to noise due to the self-organizing nature of the cell states. We rigorously characterized the noise and errors in our PNCA implementation, which arises from three main operations: (1) the input cell state due to thermal and electronic noise in the EOM, (2) the linear dot product due to phase noise and imperfect pulse temporal overlap in the coherent interference, and (3) the nonlinear activation due to thermal noise and photorefractive effects in the PPLN. We characterized these errors using 200 test images. The expected vs. measured amplitudes of alive cells in these images are shown in Fig. 5.3. The mean and standard deviation of the errors (expected amplitude — measured amplitude) achieved in our system are typical of photonic hardware, and we show that this is tolerable for the PNCA architecture due to its noise-robustness.

Self-organized image classification

We trained the experimental PNCA to perform binary image classification using the fashion-MNIST dataset consisting of 28×28 pixel gray-scale images of clothing items [33]. For example, Fig. 5.4a shows how the PNCA can classify images of sneakers and trousers. The alive cell masking is performed by designating any pixel with initial value $\alpha > 0.1$ as an alive cell, and all other pixels as dead cells with constant value of zero. Each input image was iterated for $t = 21$ time steps in the PNCA, which was sufficient for the cells to reach an approximate global agreement.

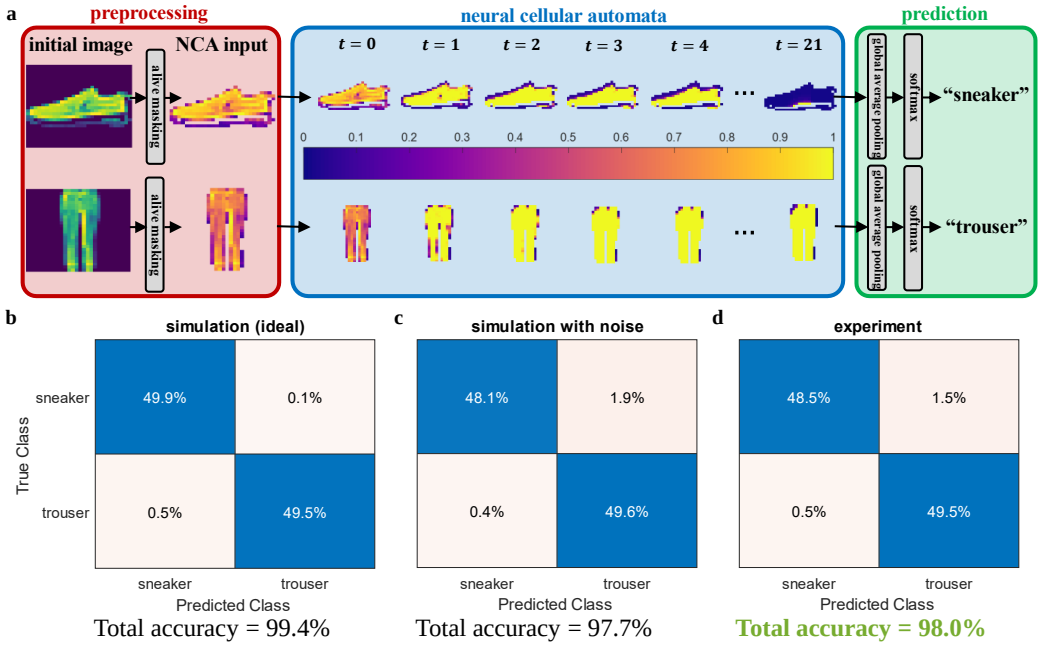

Figure 5.4: **Experimental results for fashion-MNIST binary image classification.** (a) Information flow for the PNCA trained to classify images of sneakers and trousers, beginning with alive cell masking, followed by $t = 21$ iterations of the trained PNCA. The predicted image label is obtained by global average pooling and softmax classification of the final self-organized alive cells. Confusion matrices for (b) idealized simulation model, (c) noisy simulation model, and (d) experiment.

The alive cells self-organize to have state values close to zero (unity) for images of sneakers (trousers). Finally, the predicted image label is obtained in postprocessing (see Supplementary Information Section III) by performing global average pooling of the final alive cell states followed by softmax classification. In this case, a global average closer to zero (unity) indicates that the predicted image label is sneaker (trouser).

The training procedure was performed digitally using an idealized simulation model of the PNCA that had no noise. The confusion matrix for the idealized model is shown in Fig. 5.4b, which yielded a final test accuracy of 99.4%. Next, the trained model parameters were frozen, and the model was tested again but with additional simulated Gaussian noise for each operation, matching the noise characteristics shown in Fig. 5.3. The confusion matrix for the noisy model is shown in Fig. 5.4c, which has a slightly lower final test accuracy of 97.7%. The trained model parameters were implemented in the experimental PNCA by appropriately tuning the VOAs. The confusion matrix for the experimental result tested on the same 200

images (100 for each class) used to characterize the noise in Fig. 5.3 is shown in Fig. 5.4d and has a final test accuracy of 98.0%. This experimental test accuracy is in close agreement with the simulated noisy model, which shows that the PNCA operates as desired and can successfully tolerate the use of noisy photonic hardware. No special training or noise regularization techniques were used for the PNCA. We emphasize that the robustness emerges through the local interactions between cells forming a global agreement. Therefore, even if one cell fails, the collective state can still persist (Supplementary Information Section VII).

Out-of-distribution data

Furthermore, conventional neural networks are prone to making overconfident predictions and failing to generalize to out-of-distribution data [34]. This lack of reliability is especially problematic for photonic deep learning in which the weights are fixed and online learning is not practical. The NCA approach addresses this shortcoming by using the average state value of all alive cells as a built-in measure of uncertainty. We experimentally demonstrated this for PNCA by using the same network as before that was trained on images of sneakers and trousers. Now, we test the PNCA on images of bags, which is an out-of-distribution class that the PNCA was not exposed to during training. The distributions for the alive cell averages of the sneaker, trouser, and bag classes are shown for the initial test images in Fig. 5.5a. It clearly shows that the initial distributions for alive cell averages closely overlap

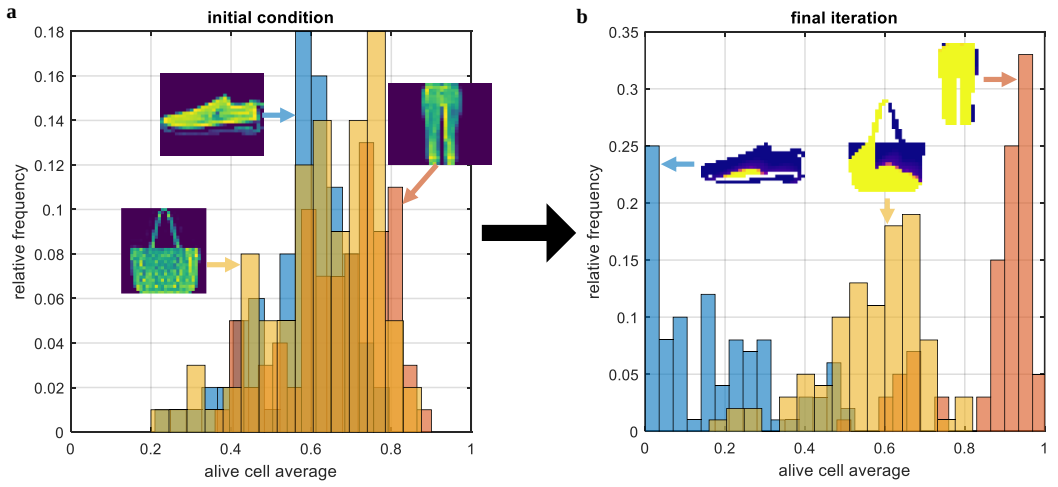

Figure 5.5: **Recognizing out-of-distribution data.** Histograms of alive cell averages for (a) initial condition and (b) final iteration of test images of sneakers (blue), trousers (red), and out-of-distribution bags (yellow).

between all classes. Upon iteration of the local update rule that was learned during training, the PNCA is able to successfully separate the distributions for sneaker and trouser, with final alive cell averages of 0.1743 and 0.8742, respectively, as shown in Fig. 5.5b. In this case, the difference between the final alive cell average and zero/one indicates the uncertainty in the prediction. However, the final alive cell average for out-of-distribution test images of bags is 0.5682, which is close to 0.5 and means that the cells did not reach a global agreement. This shows that the PNCA can use the alive cell average as a proxy for uncertainty and to detect out-of-distribution data. Unlike for conventional neural network architectures, neither special training/inference techniques nor additional training data is required.

Simulated benchmarks

In the current experimental implementation of PNCA, we represented the local update rule using only a single neuron with 3 parameters. However, the PNCA architecture can also be used with more advanced PNN hardware that can represent the local update rule using a greater number of neurons/parameters. In general, a greater number of neurons/parameters can allow for more complicated tasks and higher classification accuracy while the hardware complexity remains far less demanding than other neural network architectures. Tab. 5.1 shows the simulated binary classification accuracy of the proposed PNCA with different numbers of neurons/parameters encoding the local update rule (see Methods). The simulated PNCA was tested on different classes within the fashionMNIST dataset, as well as other similar benchmark datasets including EMNIST (28×28 grayscale images of handwritten letters) [35], MNIST (28×28 grayscale images of handwritten digits) [36], and CIFAR10 (32×32 RGB images of animals and vehicles) [37]. The corresponding classification accuracies for conventional multi-layer perceptrons (MLPs) with different numbers of hidden neurons/parameters are also shown for reference. It can be seen that the PNCA requires far fewer parameters and achieves comparable (or sometimes even better) classification accuracy to MLPs across a wide variety of examples. Therefore, the PNCA architecture may provide an efficient way for PNNs with only few parameters [14] to increase their task performance with minimal changes to existing hardware.

Dataset	Classes	PNCA			Multi-Layer Perceptron		
		1 neuron, 10 params.	10 neurons, 111 params.	100 neurons, 1101 params.	$784 \rightarrow 1 \rightarrow 1$, 787 params.	$784 \rightarrow 10 \rightarrow 1$, 7861 params.	$784 \rightarrow 100 \rightarrow 1$, 78601 params.
fashionMNIST (28×28, grayscale)	trouser/sneaker	99.4	99.9	99.9	99.8	99.9	99.9
	t-shirt/pullover	87.0	93.5	94.1	94.0	95.8	96.8
	coat/sandal	98.8	99.3	99.3	99.7	99.8	99.8
	dress/boot	94.7	98.0	98.2	99.8	99.9	99.9
	shirt/bag	89.5	95.6	95.8	95.0	95.2	97.3
EMNIST (28×28, grayscale)	A/B	83.7	99.5	99.5	95.3	97.5	98.1
	C/D	97.4	98.5	98.6	98.5	98.5	99.6
	P/Q	89.3	97.8	98.0	97.7	98.9	99.1
	R/S	87.9	96.1	96.5	96.6	98.3	99.6
	Y/Z	99.0	99.3	99.3	96.4	98.3	98.6
MNIST (28×28, grayscale)	one/two	99.4	99.5	99.5	98.6	99.5	99.7
	three/four	99.3	99.6	99.7	98.5	99.6	99.7
	five/six	95.0	97.4	97.4	95.1	95.3	97.6
	seven/eight	91.2	99.1	99.3	96.5	98.3	99.1
	nine/zero	90.1	96.5	96.6	98.2	99.0	99.4
CIFAR10 (32×32, RGB)	automobile/bird	83.0	88.8	89.7	60.1	80.7	84.7
	dog/truck	78.6	90.4	90.5	78.9	80.2	86.1
	horse/ship	77.3	84.7	85.6	72.9	78.9	85.2

Table 5.1: **Simulated benchmarks.** Image binary classification accuracy (%) for PNCA with 1, 10, and 100 neurons (10, 111, and 1101 parameters), respectively, for examples of classes in datasets: fashionMNIST (28×28 grayscale images of fashion items), EMNIST (28×28 grayscale images of handwritten letters), MNIST (28×28 grayscale images of handwritten digits), and CIFAR10 (32×32 RGB images of animals and vehicles). The corresponding accuracy for a conventional multi-layer perceptron (MLP) with 1, 10, and 100 hidden layer neurons (787, 7861, and 78601 parameters), respectively, is also shown for reference.

5.4 Discussion

We note that CA with simple rules and only nearest-neighbour connections are known to be Turing-universal models of computation [38]. This means that CA can, at least in principle, compute any function that a fully-connected network (neural or otherwise) can compute. There is no fundamental loss of computational power or information processing ability imposed by the sparsity. Therefore, given enough time, the PNCA approach (albeit with more advanced input encoding schemes) must be able to achieve at least the same accuracy as conventional neural networks such as MLPs. However, in practice, the time steps are truncated to be finite, which means the classification accuracy may not always be the same as MLPs. It is difficult to determine *a priori* on which examples the PNCA will perform better/worse compared to MLPs.

In the proposed PNCA architecture, the maximum throughput is ultimately limited by the speed of the nonlinear activation function. We chose to utilize ultrafast nonlinear optics since it can be orders of magnitude faster than digital electronics for performing nonlinear activations. The reverse-proton exchange PPLN waveguide [32] used in the experiment utilizes strong $\chi^{(2)}$ optical nonlinearity and has a phase-matching bandwidth of $\sim$ 100 GHz, which determines the maximum possible computational clock rate. This is an important step towards achieving end-to-end PNNs since it is much faster than other nonlinear activation methods utilizing optoelectronics [14], slower optical nonlinearities [39, 40], or spectral shaping [41]. Note that in our experiment, we used optoelectronic conversions after the PPLN nonlinear activation to perform feedback between iterations, however, this was not a fundamental limitation and can in principle be replaced by an all-optical feedback loop in the form of a sufficiently long optical fiber (see Supplementary Information Section IV). The scalability and performance can be further improved by using nanophotonic PPLN waveguides, which were recently demonstrated to achieve a maximum speed $>$ 10 THz and energy of $\sim$ 10 fJ per nonlinear activation [13].

In summary, we have proposed and experimentally demonstrated a novel approach to photonic deep learning based on PNCA. It addresses several system-level challenges in previous photonic neural networks, and can serve as a general architecture for a wide variety of photonic hardware platforms. In particular, we showed that PNCA enables noise-robust image classification through local interactions between cells with an inherent measure of uncertainty based on alive cell averages. Moreover, the efficient PNCA model encoding requires orders of magnitude fewer parameters

compared to MLPs or CNNs. Our single perceptron neuron rule encoding can be straightforwardly extended to a shallow neural network with a greater number of programmable parameters to perform more complicated and larger-scale computer vision tasks. For example, we focused on binary image classification for simplicity, but it is possible to perform image classification with more classes (e.g., the full 10-class MNIST image classification) if the number of output neuron channels is increased (e.g., see Supplementary Information Section II). Furthermore, we only used standard backpropagation training and did not employ any special training or regularization techniques. More advanced noise-aware or physics-aware training schemes [29] are also compatible with the PNCA architecture and may further increase performance. We used a time-multiplexed photonic network based on a synthetic temporal dimension, however, it is also possible to use an analogous PNCA approach based on other synthetic dimensions such as frequency dimensions [40, 42]. In addition to robustness to noise, it has also been previously shown that NCA are generally robust against sudden changes or failures in the underlying cell states [20, 24]. This fault-tolerance property has not yet been explored for optical implementations and can be an interesting avenue for future work on PNCA. Our work therefore highlights a clear path to advancing photonic deep learning based on PNCA and paves the way for next-generation photonic computers.

5.5 Materials and Methods

Experimental Setup

A more detailed schematic of the experimental setup is shown in Fig. S1. A femtosecond laser source (MenloSystems FC1500-250-WG) produces pulses of light at a fixed repetition rate of $\sim$ 250 MHz. The light pulses are filtered using a 200 GHz band-pass filter with center wavelength $\sim$ 1550 nm to stretch the pulse length to $\sim$ 5 ps and reduce the effects of dispersion. The light pulses are photodetected (MenloSystems FPD610-FC-NIR) as a reference clock signal for the FPGA (Xilinx Zynq UltraScale+ RFSoC) to eliminate timing drift between the optical and electronic signals. The FPGA drives an electro-optic modulator (IXblue MXAN-LN-10) that is used to modulate the amplitude of the light pulses. The light pulses are split into a 3-path interferometer by cascading 50:50 optical fiber splitters. Two paths of the interferometer have delays $+1T_R$ and $+28T_R$, respectively, relative to the shortest path, where T_R is the repetition period of the light pulses. The relative delays in each arm are set using a combination of optical fiber patch cords and free-space delay stages. Tuning the free-space coupling efficiency also acts a variable opti-

cal attenuator to set the relative amplitude weight in each arm. The output of the 3-arm interferometer is tapped using a 90:10 optical fiber splitter. The 10% tap is photodetected (Newport New Focus Model 2053) and used as an electronic locking signal input to a proportional-integral derivative controller (Red Pitaya). The electronic locking signal output is amplified (Thorlabs Piezo Controller MDT693B) and drives fiber phase-shifters (General Photonics FPS-002-L) that stabilize the relative phases of each delay arm. The 90% output of the 3-arm interferometer is amplified using an erbium-doped fiber amplifier (Thorlabs Fiber Amplifier 1550nm PM) and filtered using a 200 GHz band-pass filter to reduce the amplified spontaneous emission noise. The amplified light pulses pass through a 40 mm long reverse-proton exchange PPLN waveguide [32] that is heated to $\sim 52^{\circ}$C with a thermocouple controller. The PPLN waveguide contains a wavelength division multiplexer on the output to separate the fundamental harmonic centered at $\sim$ 1550 nm and the second harmonic centered at $\sim$ 775 nm. The second harmonic output is dumped and the fundamental harmonic is photodetected (Thorlabs DET08CFC). The final photodetected signal is read as a time trace using an oscilloscope (Tektronix MSO6B) and light pulse amplitude values are stored on the FPGA to be electro-optically reinjected.nA single photodetector can be used for tasks only requiring positive-valued inputs/outputs, such as the image classification tasks considered in this work. However, the electro-optic feedback scheme can also handle negative-valued outputs by instead using a local-oscillator with balanced photodetector. All optical fiber paths are single-mode polarization-maintaining (PM).

Photonic Neural Cellular Automata Model

The Neural Cellular Automata (NCA) comprises a lattice of cells indexed by lattice site numbers $i \in \mathbb{N}$ with states $\mathbf{x_i} \in \mathbb{C}^d$, where d is the number of channels for each cell. Each cell interacts locally in an m-cell neighborhood $\mathbb{M}_i$ according to a fixed update rule. We consider discrete-time synchronous updates $t \in \mathbb{N}$ for cells:

$$\mathbf{x_i}(t+1) = f_\theta(\mathbf{x_{m_{i1}}}(t), \mathbf{x_{m_{i2}}}(t), \mathbf{x_{m_{i3}}}(t), \cdots) \,, \tag{5.1}$$

where $m_{i1}, m_{i2}, m_{i3}, \cdots \in \mathbb{M}_i$ are the lattice sites in the local neighborhood of the i^{th} cell and $f_\theta : \left(\mathbb{C}^d\right)^m \rightarrow \mathbb{C}^d$ is the local update rule. The local update rule is parameterized by $\{\theta\}$ and is differentiable so that it can be trained using modern deep learning techniques. For example, f_θ can represent a neural network. The key aspect is that the update rule f_θ is the same for all cells and all time steps.

We experimentally demonstrated a simple version of NCA implemented directly

on analog photonic hardware, which we call Photonic Neural Cellular Automata (PNCA). In PNCA, lattice sites are represented by laser light pulses in time bins of a synthetic temporal dimension with a fixed repetition period T_R and cell states are represented by the complex amplitude of the light pulses. For simplicity, we consider a single image channel $d = 1$ and the local update rule f_θ encoded by a single perceptron neuron with an $m = 3$ neighborhood as shown in Fig. 5.2b,c. The temporal delay lines $T_1 = +1T_R$ and $T_2 = +28T_R$ set the desired local cell neighborhood and the VOAs in each arm of the 3-arm interferometer set the desired weights $\{w_0, w_1, w_2\} \in [-1, +1]$. The PIDs are used to enforce a relative phase of 0 for constructive interference, or conversely a relative phase of π for destructive interference. Therefore, at the output of the 3-arm interferometer, the combined result of the delay lines, VOAs, and phases can be summarized as a linear dot product or sliding convolutional filter:

$$y_i(t) = w_0 x_i(t) + w_1 x_{i+T_1}(t) + w_2 x_{i+T_2}(t) \ , \tag{5.2}$$

where the result of the linear operation $y_i(t)$ is fed into a PPLN to perform a nonlinear activation function:

$$x_i(t+1) = g(y_i(t)) \ , \tag{5.3}$$

where g is the sigmoid-like function shown in Fig. 5.2d. The PNCA approach is very general and Eq. 5.1 can be implemented using more complicated photonic hardware platforms with different cells neighborhoods, more neurons, deeper layers, and more programmable parameters (see Supplementary Information Section II).

Experimental Procedure

The input modulator was calibrated by using a sequence of 200 consecutive light pulses and performing a linear voltage sweep of the input EOM, which was DC biased open. The peak pulse amplitude or maximum value in each time bin (i.e., pulse repetition period) of the measured time trace was used to construct a look-up table for the voltage-to-light amplitude conversion. To input a specific 28×28 fashion-MNIST image, the 2D pixel map was unrolled column-wise to form a 784×1 vector of input cell values. Alive masking was applied such that any initial pixel value < 0.1 was designated as a dead cell. The accuracy of the input operation was checked by measuring the difference between the measured input cell states and the expected value, such as shown in Fig. S2. The aggregate results are shown in Fig. 5.2a. Each desired weight in the linear dot product was set by tuning the coupling efficiency of a free-space section contained within each VOA in the 3-arm

interferometer. Note that the VOAs were completely passive and did not consume any power. The optical power was directly measured in each arm to roughly tune the attenuation factor, and then fine-tuning of the weight was performed by checking the result of the linear interference matched the expected value like in Fig. S2. A standard Pound-Drever-Hall locking scheme was used to stabilize the relative phases in each delay arm to either 0 or π to ensure coherent interference. It is also possible to make use of the full complex amplitude of light, although we restricted our attention to only real values. The relative delays in each temporal delay line was set roughly using optical fiber patch cords, then fine-tuned using free-space delay stages to ensure maximal temporal overlap between interfering light pulses. The aggregate results of the linear dot product or convolution operation are shown in Fig. 5.2b. The temperature of the PPLN was fine-tuned around 52°C until maximal average power was measured on the output second-harmonic given a small input fundamental harmonic average power $\sim$ 1 mW. The PPLN nonlinear activation function shown in Fig. 5.2d was measured using a sequence of consecutive light pulses with linearly increasing input amplitude. To ensure stable operation over long-periods of time ($>$ 12 hours) throughout the experiment, we regularly check that the calibrated PPLN nonlinear activation function remains the same and does not change significantly due to photo-refractive or thermal effects. The measured values for PPLN nonlinear activations were also compared against the expected simulated values as shown in Fig. S2. The aggregate results of the PPLN nonlinear activation operation are shown in Fig. 5.2c. To perform self-organized image classification using the experimental PNCA, the input modulator was first calibrated. Then, the PPLN nonlinear activation function was measured, and a simulated digital model of the PNCA was trained (see Model Training) to determine the optimal weights to be set in the temporal delay lines. The light pulse amplitudes were stored digitally on the FPGA in between iterations, however, the iteration feedback can also be performed all-optically using an optical fiber cavity (Supplementary Information Section IV).

Model Training

The PNCA can be trained using the standard backpropagation-through-time algorithm (Fig. S3) for recurrent neural networks if a differentiable model of the update rule f_θ is known. The goal is to learn the parameters $\{\theta\}$ for a particular task such as self-organized image classification. We consider a cell-wise L_2 loss at each time

step:

$$L = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{N}\sum_{i=1}^{N}||\mathbf{x_i}(t) - \mathbf{z_i}||^2 \,, \tag{5.4}$$

where $\mathbf{z_i}$ is the target state for the i^{th} cell. The parameter values are updated using stochastic gradient descent:

$$\theta^{[l+1]} = \theta^{[l]} - \alpha\nabla L(\theta^{[l]}) \,, \tag{5.5}$$

where l is the epoch number and $\alpha > 0$ is the learning rate. The gradient ∇L is calculated by unrolling the network in time for T time steps and applying the chain rule or automatic differentiation. More complicated gradient-based optimization such as stochastic gradient descent with momentum or adaptive moment estimation can also be used to perform parameter updates. We trained a PNCA to perform binary image classification of sneakers and trousers classes from the fashion-MNIST dataset using 5000 training and 420 validation images for each class, learning rate of $\alpha = 0.002$, and 200 training epochs. An example of a training curve is shown in Fig. S4.

Simulation Procedure

For the simulated benchmarks, we considered PNCA using the classic Moore neighborhood (composed of the current cell plus its 8 neighboring cells in a square lattice). The local update rule f_θ was encoded by a 2-layer fully-connected network $9 \to N \to 1$, where N is the number of hidden neurons. The simulation parameters are shown below in Tab. 5.2. For the CIFAR10 dataset examples, we applied the *same* local update rule channel-wise to each RGB input channel for the images, then averaged over the channels for the final classification.

For the simulated Multi-Layer Perceptrons (MLPs) used for comparison, we used a 2-layer fully-connected network $784 \to N \to 1$, where N is the number of hidden neurons. We used clipped ReLU nonlinear activation function $f(x) = \min(1, \max(0, x))$ to ensure that the final output probability is in range $[0, 1]$. Each MLP was trained using binary cross-entropy loss, the same number of training/validation/test images as for the corresponding PNCA, learning rate of 0.0001, and 500 epochs. The images were flattened column-wise to form the input to the MLPs and we resampled the images to be 28×28 grayscale for CIFAR10 since MLPs can only accept inputs with a fixed dimension, whereas PNCA can handle arbitrary image input sizes.

dataset	iterations	alive cell threshold	training images per class	validation images per class	test images per class	learning rate	epochs
fashionMNIST	25	0.1	5000	420	1000	0.01	500
EMIST	15	0.1	500	420	500	0.01	1000
MNIST	10	0.1	5000	420	800	0.01	1000
CIFAR10	10	0.15	500	420	500	0.005	2000

Table 5.2: **Simulated benchmark parameters.**

5.6 Supplementary Information

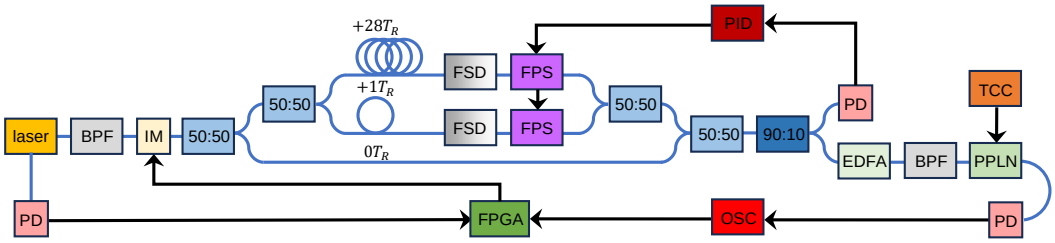

Figure 5.6: **Detailed schematic of experimental setup.** BPF: band-pass filter, IM: intensity modulator, FSD: free-space delay stage, FPS: fiber phase-shifter, EDFA: erbium-doped fiber amplifier, PD: photodetector, PPLN: periodically-poled lithium niobate, TCC: thermocouple controller, PID: proportional integral derivative controller, OSC: oscilloscope, FPGA: field programmable gate array. Blue lines represent single-mode polarization-maintaining optical fiber paths and black lines represent electronic connections.

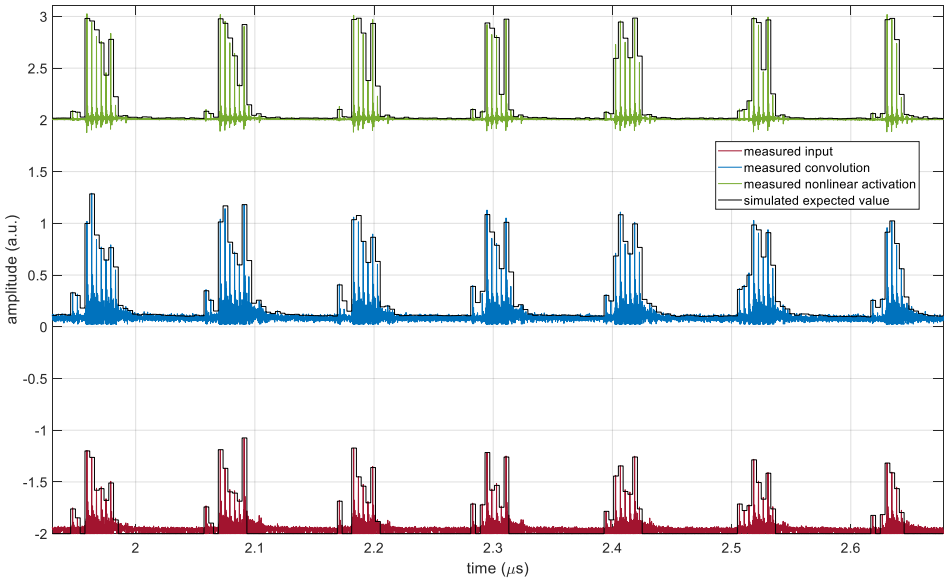

Figure 5.7: **Example of experimentally-obtained time traces in photonic neural cellular automata.** Oscilloscope time traces for a portion of one iteration of one image in the photonic neural cellular automata showing the results of the input cell states (red line), linear convolution (blue line), and nonlinear activation (green line) compared against the expected values based on digital simulations (black lines). The peak or maximum amplitudes of each light pulse, which occupy a specific time bin in the synthetic temporal lattice, agrees well with the expected cell state values and shows that the calibration was accurate. The time traces for each operation are shifted vertically and corresponding cell lattice sites are aligned horizontally in time for clarity.

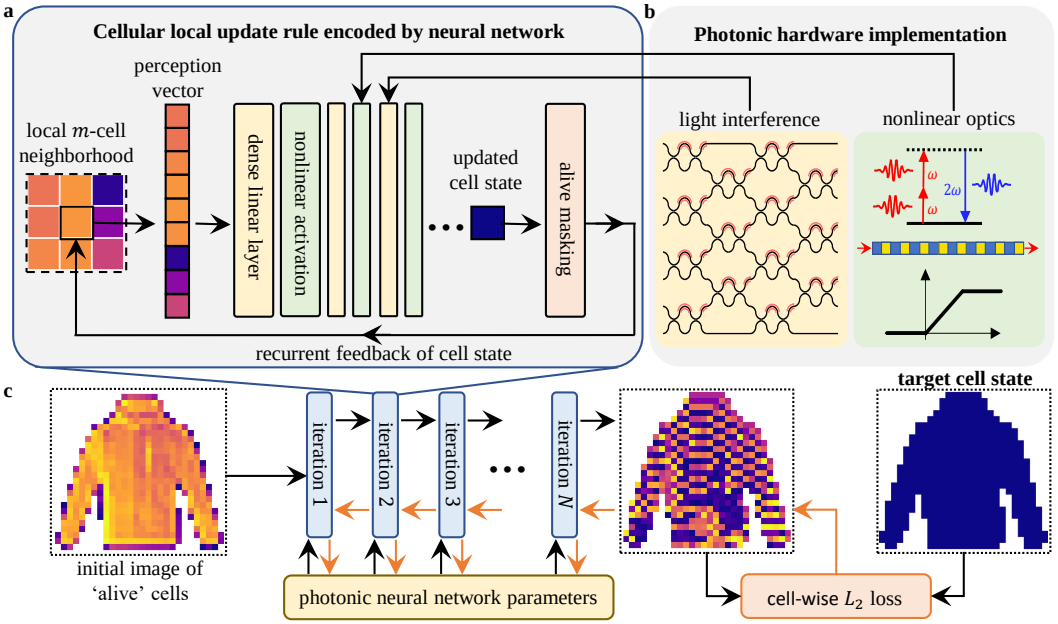

Figure 5.8: **Training method for photonic neural cellular automata.** (a) A single iteration of PNCA consists of alive cells that are encoded into an optical signal, m optical paths encoding a local m-cell neighborhood and perception vector for each cell, updating the state of each cell according to a local update rule represented by a neural network, and alive cell masking. (b) Photonic hardware encodes the local update rule, which includes linear operations implemented physically via light interference, and nonlinear operations implemented physically via nonlinear optics. (c) Backpropagation-through-time algorithm for training PNCA to learn a local update rule, which upon repeated iteration causes self-organization of cells for an image classification task. A cell-wise L_2 loss is used for optimizing the photonic neural network parameters.

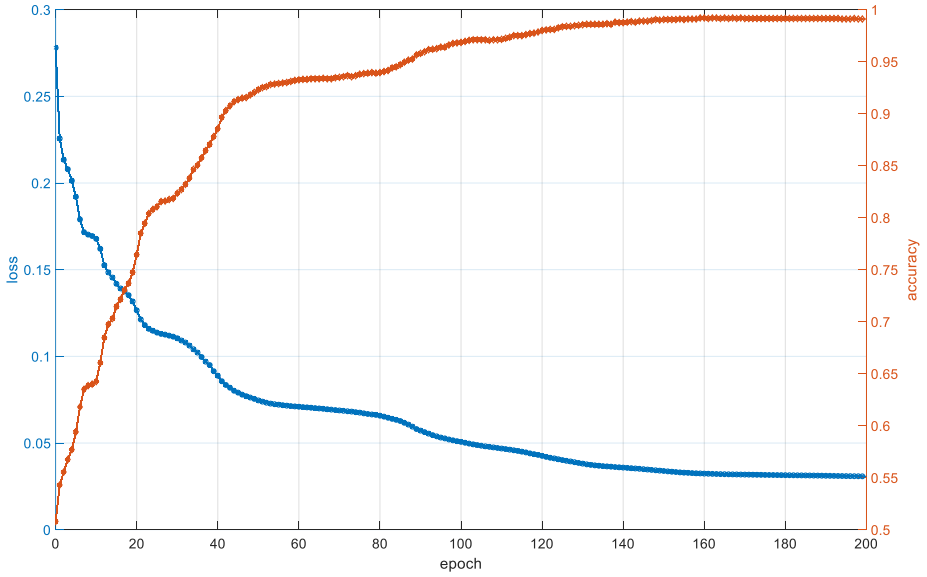

Figure 5.9: **Training progress for photonic neural cellular automata.** Cell-wise L_2 loss (blue line) and classification accuracy (orange line) for training PNCA to perform binary image classification of sneakers and trousers classes from fashion-MNIST dataset.

Introduction to Neural Cellular Automata

In this section, we introduce the principles of Neural Cellular Automata (NCA) and how it differs from conventional neural network architectures. Cellular Automata (CA) are computational models composed of a lattice of cells with states that update over time according to simple rules. Crucially, these rules only depend on the local information of neighboring cell states and every cell follows the same local update rule. For example, consider the simple example of a CA known as Elementary Cellular Automata Rule 90. It is a discrete-time CA defined over a 1D lattice of cells with binary states, which we denote by $x_i(t) \in \{0, 1\}$ where $i \in \mathbb{Z}$ is the cell index or lattice site and $t \in \mathbb{N}$ is the discrete time step. The lattice extends infinitely in both directions. Rule 90 follows the simple local update rule: $x_i(t+1) = x_{i-1}(t) \oplus x_{i+1}(t)$, i.e., a cell's next state is given by the logical XOR of its neighboring cell states. This local update rule can also be depicted visually in a truth table as shown in Fig. 5.10a. The top row shows the local 3-cell neighborhood cell states for all 8 possible cases, and the bottom row shows the updated cell state. This local update is applied uniformly and simultaneously to all cells in the 1D lattice. The global effect of this local update rule can be visualized using a space-time diagram with the horizontal axis representing cell lattice sites and the vertical axis representing successive time

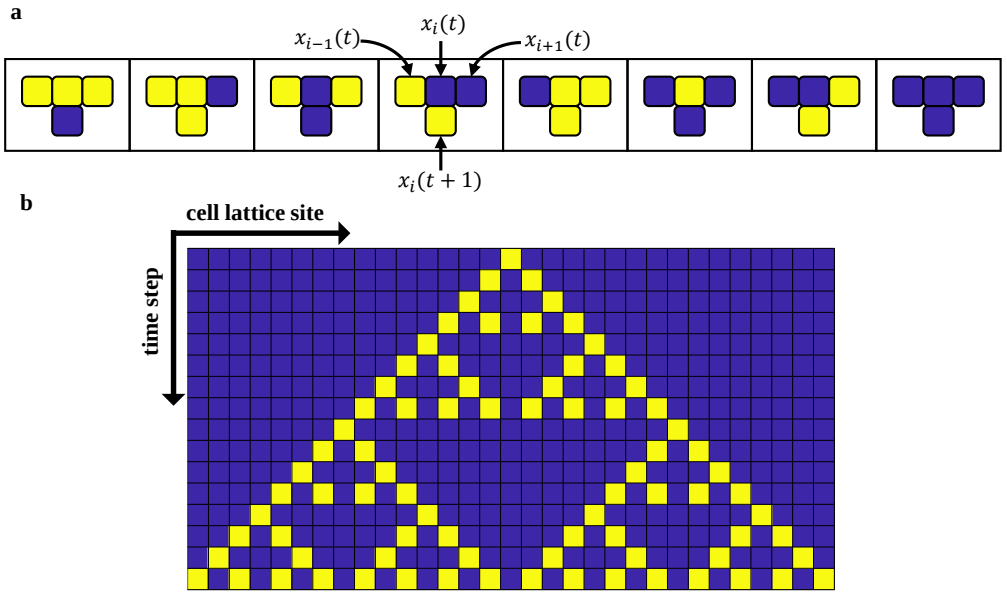

Figure 5.10: **Elementary Cellular Automata Rule 90.** (a) Truth table for local update rule. (b) Space-time diagram for Rule 90 starting from a single live cell.

steps. An example is shown in Fig. 5.10b for an initial condition in which only a single cell is live (has state 1) and all other cells are dead (has state 0). In this case, the space-time diagram is in the shape of a Sierpinski Triangle fractal. A common theme for CA is that even very simple local update rules can lead to global complex or emergent phenomena such as fractals, chaos, synchronization, self-organization, and solitons [21]. It has also been proven that even simple elementary rules are capable of Turing-universal computation [38].

However, although CA are capable of universal computation, it is not always obvious how to harness the complex dynamics in CA to perform *useful* computational tasks. Indeed, much of the early work in CA focused on studying human-designed or bespoke local update rules. In general, we can consider a lattice of cells indexed by lattice site numbers $i \in \mathbb{N}$ with states $\mathbf{x_i} \in \mathbb{C}^d$, where d is the number of channels for each cell (i.e., each cell state is given by a d-dimensional vector). Each cell interacts locally in an m-cell neighborhood $\mathbb{M}_i$ according to a fixed update rule. We consider discrete-time synchronous updates $t \in \mathbb{N}$ for cells:

$$\mathbf{x_i}(t+1) = f(\mathbf{x_{m_{i1}}}(t), \mathbf{x_{m_{i2}}}(t), \mathbf{x_{m_{i3}}}(t), \cdots) \,, \tag{5.6}$$

where $m_{i1}, m_{i2}, m_{i3}, \cdots \in \mathbb{M}_i$ are the lattice sites in the local neighborhood of the i^{th} cell and $f : \left(\mathbb{C}^d\right)^m \to \mathbb{C}^d$ is the local update rule.

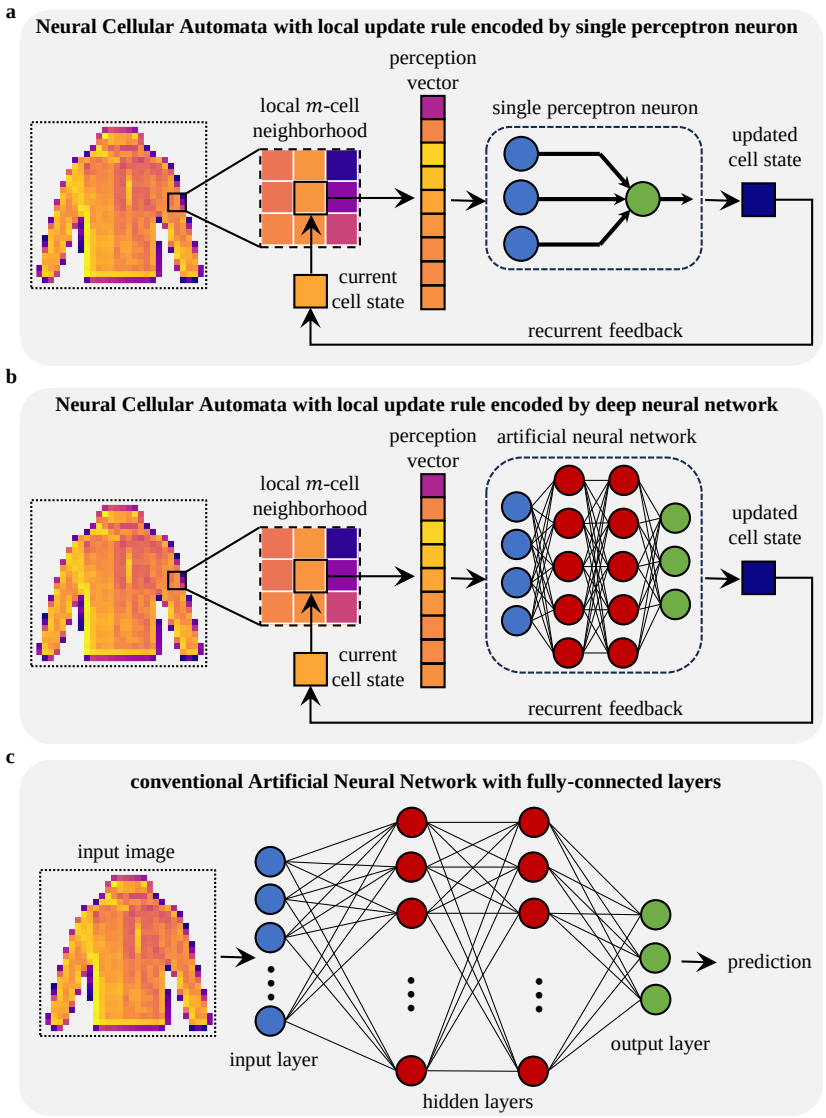

Figure 5.11: **Network Topology.** Examples of Neural Cellular Automata (NCA) encoding the local update rule for Cellular Automata (CA) using (a) single perceptron neuron or (b) a deep neural network. (c) Conventional artificial neural networks are feed-forward networks requiring fully-connected layers with global information.

The key innovation of the Neural Cellular Automata (NCA) approach is to use modern deep learning techniques to discover local update rules f that can perform specific tasks [24]. To do this, the cell states are continuous and the local update rule is encoded by a neural network f_θ with parameters θ to be trained. The input to the neural network is a vector composed of the local m-cell neighborhood. For example, Fig. 5.11a shows the simplest possible example in which the local update

rule is encoded by a single perceptron neuron, which is parameterized by its linear input weights. The output of the neuron represents the updated cell state, which undergoes recurrent feedback for the next time step iteration. Therefore, NCA is a special kind of recurrent neural network. We showed in our experimental implementation of NCA using photonics that a local update rule encoded by just a single neuron with 3 parameters can still perform surprisingly complex tasks. Fig. 5.11b shows the more general case in which the local update rule is encoded by a deeper neural network with more parameters. In order to perform tasks such as image classification, ancillary steps such as alive cell masking and global averaging for label prediction are required (e.g., see Supplementary Section III for more details). The NCA can be trained using the standard backpropagation-through-time algorithm (Fig. 5.8). It is also common to consider NCA with stochastic updates in which cell states only update with a given probability during each time step. For simplicity, we consider only deterministic NCA in our work. In contrast, the typical network topology of a conventional artificial neural network such as Multi-Layer Perceptron (MLP) is shown in Fig. 5.11c. It contains an input layer, hidden layer/s, and an output layer arranged in a purely feed-forward network. The layers are fully-connected and require global information. Therefore, NCA can be very parameter-efficient compared to MLPs since the number of parameters needed to encode local update rules is generally far fewer than the number of parameters needed to encode network layers with global information. The process of self-organization through local interactions in NCA for image classification is fundamentally different to the process by which MLPs classify images.

Examples of how to implement PNCA

In this section, we give more examples of how PNCA may be implemented using existing photonic neural network hardware. We focus on time-multiplexed photonic networks, however, we note that analogous results can be obtained using spatial or wavelength multiplexing [42]. The general architecture for PNCA is shown in Fig. 5.12a. Each iteration consists of an input optical encoding (e.g., setting initial conditions or alive cell masking using electro-optic modulators) followed by photonic hardware encoding the update rule for cells in a local neighborhood. The key principle of CA is that cells update according to rules that only depend on local information. Therefore, in PNCA the photonic hardware only encodes for the local update rule, not the global network weights. The local neighbourhood may be encoded, for example, using temporal delay lines to specify which cells will interact

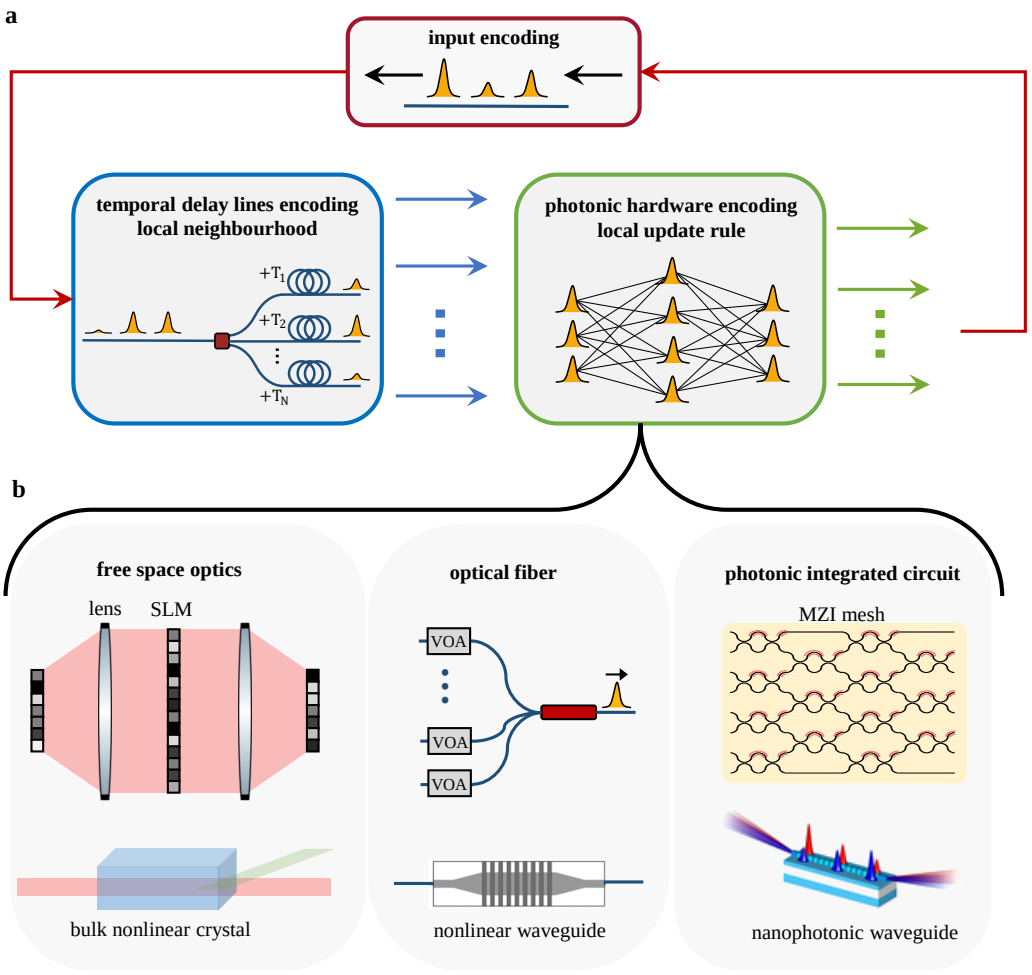

Figure 5.12: **PNCA architecture.** (a) The general architecture for PNCA using time-multiplexing includes input encoding of optical cells, temporal delay lines to encode the local cell neighborhood, and photonic hardware to encode the local update rule. (b) The photonic hardware for the local update rule can be implemented using many different methods including free space optics, optical fiber, and photonic integrated circuits.

in the photonic local update rule. The photonic hardware for the local update rule can be implemented using a myriad of different methods. Notably, it requires minimal or no changes to existing photonic neural network hardware. For example, PNCA can accommodate any combination of: free space optical neural networks such as coherent [18] or incoherent [43] diffractive neural networks and bulk nonlinear crystals [29]; fiber-based optical networks such as based on time-delay dynamics [22] or wavelength-time interleaving [11], and integrated photonic neural networks such as Mach-Zehnder interferometer meshes [9], cross-bar arrays [10] and nanophotonic

nonlinear waveguides [13]. These examples are not an exhaustive list, but rather intended to show how switching to the PNCA architecture can potentially expand the functionality of existing photonic neural network hardware.

Step-by-step explanation of data processing procedure

In this section, we give a detailed overview of the steps used to perform self-organized image classification using the experimentally implemented PNCA. For each input image, we performed the following steps:

1. Pre-processing: prepare the data to be acceptable as input to the PNCA. This step is performed on an external control computer.

    a) Concatenate the columns of the 28×28 input image to flatten it into a 784×1 vector.
    This is necessary because our experimental implementation of PNCA only accepts a 1D input sequence. However, this step can be skipped if the PNCA can accept 2D imputs.

    b) Calculate alive cell mask: any pixel with initial value ≥ 0.1 is labelled as *alive* and any pixel with initial value < 0.1 is labelled as *dead*.

2. Neural Cellular Automata: iterate the local update rule for cells. This step is performed in the photonic hardware. For each iteration $t = 1, 2, \cdots, 21$:

    a) Alive cell masking: cell values represented by the 784×1 vector are modulated onto the amplitude of light pulses. Additionally, the amplitude of any light pulse corresponding to a dead cell is set to 0.

    b) Update cell values: propagate light pulses through the photonic neural network encoding the local update rule. For example, in this case, it is a single photonic neuron consisting of a multi-arm interferometer and nonlinear PPLN waveguide.

    c) Feed back cell values: return light pulses back to the start of the network for the next iteration.
    In our experiment, this was done optoelectronically by photodetecting the light pulse after the network and then reinjecting it electro-optically. However, the feedback can also be performed all-optically using a sufficiently long optical delay line.

3. Prediction: assign the output image classification label. This step is performed on an external control computer.

    a) Alive cell average: calculate the average cell value of alive cells (i.e., do not include dead cells) after the final iteration.
    Although this step can also be done in photonic hardware, it is not necessary to perform it fast in real-time since it is only computed once after all cell update iterations are completed.

    b) Output label: the averaged alive cell value gives the probability of the input image belonging to class 1. If the value is closer to zero, then the label is class 0 (e.g., sneaker), otherwise the label is class 1 (e.g., trouser).
    In this experiment, we only performed binary (2-class) classification. However, a more general PNCA classifying more than 2 classes can use the standard softmax activation to assign probabilities and class labels.

Optoelectronic vs. All-Optical Feedback

In this section, we describe possible implementations of the memory in PNCA. Recurrent feedback is needed to input cell states between iterations of the local update rule. The first case is an optoelectronic memory making use of a measurement-feedback scheme. Fig. 5.13a shows a possible implementation using a time-multiplexed network. Input cell states and alive masking are performed using an amplitude modulator, e.g., an electro-optic modulator, to set the amplitude of light pulses representing cells. Time-delayed copies of the pulses are used to implement the local m-cell neighborhood, which comprises the perception vector that is input to the local update rule. For example, in our experimental setup (see Experimental realization of PNCA), the local update rule is encoded by a single neuron with linear weights set by variable optical attenuators and nonlinear activation performed by a periodically-poled lithium niobate (PPLN) waveguide. Other photonic neural network hardware with more neurons and parameters can also be used to encode the local update rule (e.g., see Supplementary Section II). The updated cell states are photodetected and stored on an electronic memory (e.g., FPGA), which are then electro-optically reinjected for the next iteration of the PNCA. The measurement-feedback scheme is a simple and convenient method for cell state feedback, however, its speed is ultimately limited by the bandwidth of the optoelectronic conversions. The second case overcomes this issue by using an all-optical memory. Fig. 5.13b shows a possible implementation using a time-multiplexed network. Input cell states

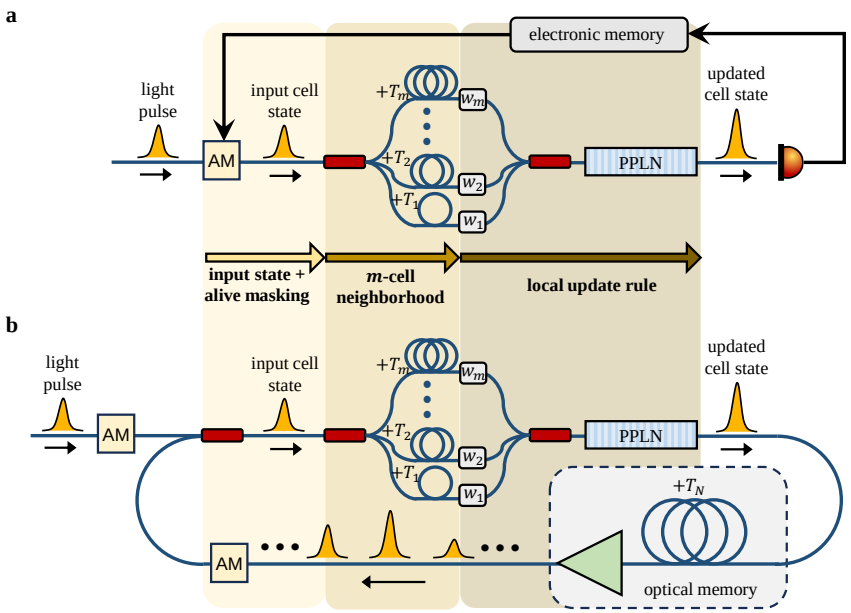

Figure 5.13: **Photonic Neural Cellular Automata Memory.** Example of an implementation using: (a) optoelectronic conversion, electronic memory and amplitude modulator (AM); and (b) all-optical memory and AM.

and alive masking are performed using an amplitude modulator (AM) to set the initial amplitude of light pulses representing cells. This initial AM is used to set the initial conditions (i.e., image pixel values) and is not active beyond the initial $t = 0$ iteration. The PNCA computation based on local m-cell neighborhood and local update rule proceeds similarly as before. The updated cell state pulse passes through an optical memory comprised of an optical delay line with time delay T_N, where N is the number of cells in the network, and an optical amplifier (e.g., erbium-doped fiber amplifier or optical parametric amplifier) to compensate the optical loss. A second AM is needed in this case to perform the alive masking between each iteration in the optical feedback line, which inputs the pulses for the next iteration. Note that the AMs can also be implemented using previously demonstrated all-optical switches based on ultrafast nonlinear optics [44] so that it is not limited by electronic bandwidths. Therefore, the entire PNCA can be implemented all-optically without any electronic bottleneck. This all-optical method requires a long delay line with delay $+T_N$ to ensure that cells in each iteration do not undesirably interfere with cells in subsequent iterations. The network size N is only limited by the length of available optical fiber.

Architecture	Image Dataset (Image Size)	Input Update Method	Input Updates per Image	Input Update Rate	Weight Update Method	Weight Updates per Image	Weight Update Rate	Linear Operation Method	Nonlinear Activation Method	Image Inference Time	Claimed Energy Efficiency	Claimed Operation Speed
PNCA (this work)	fashion-MNIST (28×28)	Electro-optic modulator	15680 [a]	12 GHz	Variable optical attenuators	0	0	Optical	Optical	**1.307 μs**	5.6 TOPs/J	84 GOPs/s
CNN [11]	MNIST (30×30)	Electro-optic modulator	900	11.9 GHz	Spectral waveshaper	1 [b]	2 Hz	Optical	Digital Electronic	>500 ms [*]	N/A	11.321 TOPs/s
D^2NN [18]	MNIST (28×28)	Digital micro-mirror display	3	14.989 kHz	Spatial light modulator	3	422.4 Hz	Optical	Opto-electronic	20.7 ms [c]	0.717 TOPs/J	**114.1** TOPs/s
VCSEL-ONN [45]	MNIST (28×28)	VCSEL	15900 [d]	1 GHz	VCSEL array	15900 [d]	1 GHz	Opto-electronic	Opto-electronic	15.9 μs	**140 TOPs/J**	$\underset{e}{\sim}$1 TOPs/s
PNN [29]	MNIST (14×14)	Digital micro-mirror display	8 [f]	200 Hz	Digital micro-mirror display	8 [f]	200 Hz	Digital Electronic	Optical	>40 ms [*]	N/A	N/A
CNN [46]	fashion-MNIST (14×14)	Variable optical attenuators	169	1 kHz	Electrical PCM cell	4	1.193 MHz [g]	Optical + Digital Electronic	Digital Electronic	0.17 s	5 TMACs/J	25 Gb/s
CNN [10]	MNIST (28×28)	Electro-optic modulator	784	2 GHz	Optical PCM cell	0	0	Optical + Digital Electronic	Digital Electronic	>8.1 μs [*]	58.8 TMACs/J	2 TMACs/s
OONA [43]	QuickDraw (28×28)	Image intensifier	1	10 Hz	Spatial light modulator	0	0	Optical + Digital Electronic	Opto-electronic + Digital Electronic	>5 ms [*]	N/A	N/A

Table 5.3: **Comparison of state-of-the-art optical neural networks performing image classification.**

[a] Assuming $t = 20$ iterations ($784 \times 20 = 15680$)
[b] Assuming the waveshaper was only updated once in between the convolutional and fully-connected layers
[c] Assuming a 3-layer D^2NN with DPU working cycle of 6.9 ms including sensor exposure time
[d] Assuming a $784 \rightarrow 100 \rightarrow 10 \rightarrow 10$ 3-layer model with arbitrary waveform generator using 5 VCSELs per acquisition ($784 \times 20 + 100 \times 2 + 10 \times 2 = 15900$)
[e] Assuming a compute density of 6 TOPs/mm^2/s and a VCSEL chip area of $25 \times 80\mu$m$\times 80\mu$m
[f] Assuming 4 independent channels with 2 SHG transformation steps per channel
[g] Assuming 282 ns write time and 556 erase time per GST cell switching cycle
[*] digital electronic processing time not reported

Comparison to other optical neural networks

In this section, we compare the experimentally demonstrated PNCA to other state-of-the-art optical neural networks performing image classification. We consider system-level demonstrations of images with size at least 14×14 since anything smaller is not representative. We compare only the best value for experimentally reported hardware and not theoretically proposed or future devices. We consider the image inference time, which is the minimum time needed to classify one image, to be a key figure of merit for photonic hardware accelerators. This can be calculated from considering the number of input updates and weight updates needed for each image. For example, in the case of our PNCA, the number of input updates required per image (assuming $t = 20$ iterations of the local update rule) is $784 \times 20 = 15680$. The input update rate is limited by the bandwidth of our electro-optic modulator, which is 12 GHz. One of the crucial points about PNCA is that there is no weight updates required, so this does not contribute to the inference time. We ignore the light propagation time since it is almost always negligible compared to the time taken for input/weight updates. Therefore, the image inference time is $15680/12$ GHz = 1.307μs. Another key consideration is how the linear/nonlinear operations are implemented since using electronics for parts of the computation will add significant time and overhead, which is usually not reported. The experimental PNCA implements both linear and nonlinear operations in the optical domain so there is minimal overhead. Besides the equipment control electronics, optoelectronics was used in storing and reinjecting the cell states between iterations, but this does not affect the speed of the computation since the time needed to reinject a single cell state is much less than the time taken for a complete iteration. Similarly, the time taken to output classification labels electronically in the last step is much less than the time taken to complete all iterations. During each iteration, each cell experiences 1 alive masking operation, 3 MAC operations, and 1 nonlinear activation, hence giving a total of 7 operations per cell (assuming 1 MAC = 2 OPs). Therefore, the current operation speed is 7 OPs $\times$12 GHz = 84 GOPs/s. The average optical power was 15 mW, so we can estimate the optical energy consumption for the computation as (84 GOPs/s)/15 mW = 5.6 TOPs/J. However, we note that the power consumption of the control electronics is much higher than the optical power, which means the overall wall-plug energy efficiency is likely much worse than implied by just the optical energy consumed during the computation. Interestingly, we see that a higher claimed operation speed does not directly translate into a faster inference time. The claimed operation speeds often assume full utilization of the photonic hardware, which ig-

nores important system-level constraints such as input and/or weight updates that can drastically slow the system when performing a useful image classification task. Our PNCA experimental implementation is the only end-to-end photonic neural network combining linear optical operations, nonlinear optical operations, fast input update speeds, static weights, and no electronic bottleneck.

$\chi^{(2)}$ Nonlinear Optical Activation Functions

In this section, we describe the equations governing the pump-depleted second harmonic generation (SHG) in the periodically-poled lithium niobate (PPLN) waveguide used for nonlinear activations in the main text experiments. SHG arises due to the second-order ($\chi^{(2)}$) parametric processes in non-centrosymmetric crystals such as lithium niobate. This $\chi^{(2)}$ nonlinearity is of the ultrafast variety with an effectively instantaneous response time, which is only limited in practice by the phase-matching bandwidth of the device. Unlike other slower optical nonlinearities [12, 14] that are incoherent, the PPLN waveguide can correctly handle both positive and negative valued input light amplitudes and is a completely travelling-wave process (no resonator required). In its simplest form, the SHG process is governed by the nonlinear coupled-mode equations [47] in Eq. 5.7:

$$\frac{d}{dz}A(z) = -i\kappa^* A^*(z)B(z)\exp(-i2\Delta z)\ , \tag{5.7a}$$

$$\frac{d}{dz}B(z) = -i\kappa\left[A(z)\right]^2\exp(i2\Delta z)\ , \tag{5.7b}$$

where $A(z)$ is the amplitude of the fundamental harmonic light of frequency ω, $B(z)$ is the amplitude of the second harmonic light of frequency 2ω, κ is the nonlinear coupling coefficient, Δ is the phase-mismatch parameter between the ω and 2ω light waves, and z is the propagation distance along the waveguide's longitudinal direction. In the case of our experiments, the input value is encoded onto the coherent light amplitude of the fundamental harmonic wave and we restrict our attention to only real-valued inputs. Solving the initial value problem Eq. 5.7 with $A(0) = A_0$ and $B(0) = 0$ for a waveguide of length L gives the nonlinear activation function, $g : \mathbb{R} \to \mathbb{R}$, as the output fundamental harmonic light amplitude $A(L) = g(A_0)$. Suppose that $[A^+(z),\ B^+(z)]$ with $A^+(0) = A_0$ and $B^+(0) = 0$ is a solution to Eq. 5.7 with initial values $A^+(0) = A_0$ and $B^+(0) = 0$. Then, by direct substitution, we see that $[A^-(z) = -A^+(z),\ B^-(z) = B^+(z)]$ is also a solution with initial values $A^-(0) = -A_0$ and $B^-(0) = 0$. Therefore, the nonlinear activation function $g(\cdot)$ is an anti-symmetric (odd) function, i.e., $g(-x) = -g(x)$ for $x \in \mathbb{R}$. For example, in

the simplified case of continuous-wave light and perfect phase-matching ($\Delta = 0$), the exact analytical solution to the initial value problem Eq. 5.7 with $A(0) = A_0$ and $B(0) = 0$ is:

$$A(z) = A_0 \text{sech}(\kappa A_0 z),\ B(z) = -iA_0 \tanh(\kappa A_0 z)\ . \tag{5.8}$$

Clearly, the nonlinear activation function $A(L) = g(A_0)$ is anti-symmetric (odd) for this simple case. Alternatively, we could also use the second harmonic output $B(L) = f(A_0)$ if we desire a symmetric (even) nonlinear activation function, however, the output would be at a different frequency to the input. The case for short laser pulses is more complicated and can only be solved numerically, e.g., see Ref. [13].

Signal-to-Noise Ratio

In this section, we elaborate on the signal-to-noise (SNR) of the PNCA and the maximum noise tolerance. A key point for PNCA is that the information (e.g., image class label) is encoded in the global or collective cell states, and not necessarily reliant on any individual cell. Therefore, we can assign the alive cell average as the relevant "signal" and track its evolution across iterations for a specific image class. The noise can then be considered as the standard deviation of the alive cell average distribution. For example, Fig. 5.14(a) shows the SNR at different iterations of the experimentally demonstrated PNCA (see main text Fig. 4) for fashionMNIST trouser images. Initially, the SNR is low since the alive cell states are the same as the input image pixel values and have a large variance. Interestingly, the SNR increases during the first few iterations before decreasing and eventually plateauing. This is because the most rapid changes due to the self-organizing behaviour occur in the first few iterations.

We can also study how the noise from specific optical operations impacts the classification accuracy. In our experimental implementation of PNCA, we carefully characterized the noise (see main text Fig. 3), which can be considered typical of photonic hardware. We showed that PNCA was robust to this level of noise. To determine the maximum degree of noise robustness, we now simulate the PNCA classification accuracy for the same task of binary image classification of fashionMNIST sneaker and trouser images as the noise is increased. For simplicity, we vary only the standard deviation σ of the noise of the linear operations (convolutions) since it is the dominant noise source, and keep the input and nonlinear activation noise fixed (as shown in main text Fig. 3). The accuracy for different levels of noise

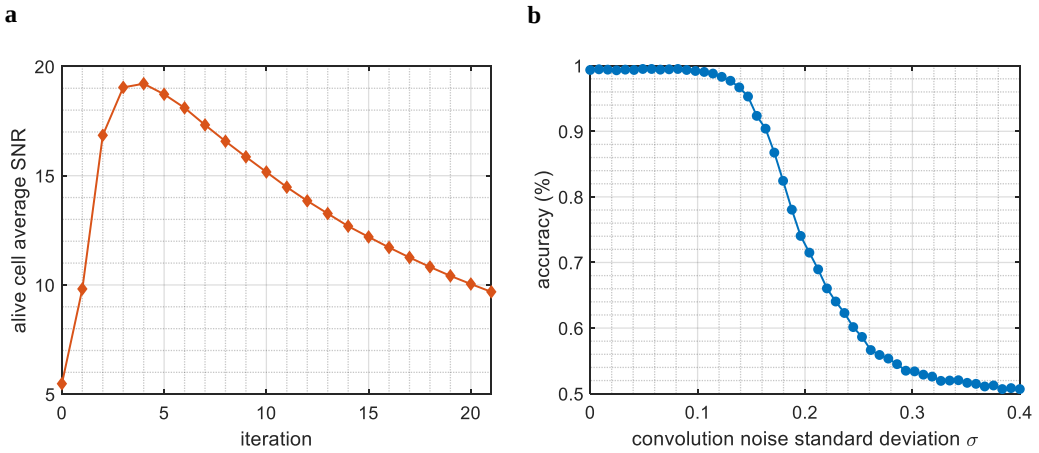

Figure 5.14: **Noise Properties of PNCA.** (a) Signal-to-noise ratio at each iteration of the experimentally demonstrated PNCA tested on fashionMNIST trouser images. (b) Simulated PNCA accuracy for binary image classification of fashionMNIST sneaker and trouser images when the convolution noise standard deviation is varied.

σ is shown in Fig. 9(b). The PNCA maintains near-perfect binary classification accuracy up to $\sigma \approx 0.1$, which is > 3 times the actual experimental noise level of $\sigma = 0.0327$. The PNCA maintains $> 90\%$ accuracy up to > 5 times the experimental noise level. The PNCA doesn't completely fail until $\sigma \approx 0.35$, which is > 10 times the experimental noise level. This emphasizes the fact that the noise properties of specific operations acting on individual cells are not straightforwardly correlated with the emergent global properties of the alive cell averages.

References

[1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[3] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative ad-

versarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[5] Jaime Sevilla, Lennart Heim, Anson Ho, Tamay Besiroglu, Marius Hobbhahn, and Pablo Villalobos. Compute trends across three eras of machine learning. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022.

[6] Norman P. Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th annual international symposium on computer architecture*, pages 1–12, 2017.

[7] Bhavin J. Shastri, Alexander N. Tait, Thomas Ferreira de Lima, Wolfram H.P. Pernice, Harish Bhaskaran, C. David Wright, and Paul R. Prucnal. Photonics for artificial intelligence and neuromorphic computing. *Nature Photonics*, 15 (2):102–114, 2021.

[8] Gordon Wetzstein, Aydogan Ozcan, Sylvain Gigan, Shanhui Fan, Dirk Englund, Marin Soljačić, Cornelia Denz, David A.B. Miller, and Demetri Psaltis. Inference in artificial intelligence with deep optics and photonics. *Nature*, 588 (7836):39–47, 2020.

[9] Yichen Shen, Nicholas C. Harris, Scott Skirlo, Mihika Prabhu, Tom Baehr-Jones, Michael Hochberg, Xin Sun, Shijie Zhao, Hugo Larochelle, Dirk Englund, et al. Deep learning with coherent nanophotonic circuits. *Nature Photonics*, 11(7):441–446, 2017.

[10] Johannes Feldmann, Nathan Youngblood, Maxim Karpov, Helge Gehring, Xuan Li, Maik Stappers, Manuel Le Gallo, Xin Fu, Anton Lukashchuk, Arslan Sajid Raja, et al. Parallel convolutional processing using an integrated photonic tensor core. *Nature*, 589(7840):52–58, 2021.

[11] Xingyuan Xu, Mengxi Tan, Bill Corcoran, Jiayang Wu, Andreas Boes, Thach G. Nguyen, Sai T. Chu, Brent E. Little, Damien G. Hicks, Roberto Morandotti, et al. 11 tops photonic convolutional accelerator for optical neural networks. *Nature*, 589(7840):44–51, 2021.

[12] Johannes Feldmann, Nathan Youngblood, C. David Wright, Harish Bhaskaran, and Wolfram H.P. Pernice. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature*, 569(7755):208–214, 2019.

[13] Gordon H.Y. Li, Ryoto Sekine, Rajveer Nehra, Robert M. Gray, Luis Ledezma, Qiushi Guo, and Alireza Marandi. All-optical ultrafast relu function for energy-efficient nanophotonic deep learning. *Nanophotonics*, 12(5):847–855, 2023.

[14] Farshid Ashtiani, Alexander J. Geers, and Firooz Aflatouni. An on-chip photonic deep neural network for image classification. *Nature*, 606(7914):501–506, 2022.

[15] David A.B. Miller. Are optical transistors the logical next step? *Nature Photonics*, 4(1):3–5, 2010.

[16] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE*, 105(12):2295–2329, 2017.

[17] Nabil H. Farhat, Demetri Psaltis, Aluizio Prata, and Eung Paek. Optical implementation of the hopfield model. *Applied Optics*, 24(10):1469–1475, 1985.

[18] Tiankuang Zhou, Xing Lin, Jiamin Wu, Yitong Chen, Hao Xie, Yipeng Li, Jingtao Fan, Huaqiang Wu, Lu Fang, and Qionghai Dai. Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit. *Nature Photonics*, 15(5):367–373, 2021.

[19] Xing Lin, Yair Rivenson, Nezih T. Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi, and Aydogan Ozcan. All-optical machine learning using diffractive deep neural networks. *Science*, 361(6406):1004–1008, 2018.

[20] Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, and Michael Levin. Growing neural cellular automata. *Distill*, 5(2):e23, 2020.

[21] Stephen Wolfram. Statistical mechanics of cellular automata. *Reviews of Modern Physics*, 55(3):601, 1983.

[22] Gordon H.Y. Li, Christian R. Leefmans, James Williams, and Alireza Marandi. Photonic elementary cellular automata for simulation of complex phenomena. *Light: Science & Applications*, 12(1):132, 2023.

[23] Martin Gardner. The fantastic combinations of john conway's new solitaire game "life". *Scientific American*, 223:20–123, 1970.

[24] Ettore Randazzo, Alexander Mordvintsev, Eyvind Niklasson, Michael Levin, and Sam Greydanus. Self-classifying mnist digits. *Distill*, 5(8):e00027–002, 2020.

[25] Eyvind Niklasson, Alexander Mordvintsev, Ettore Randazzo, and Michael Levin. Self-organising textures. *Distill*, 6(2):e00027–003, 2021.

[26] Alexander N. Tait, Thomas Ferreira De Lima, Ellen Zhou, Allie X. Wu, Mitchell A. Nahmias, Bhavin J. Shastri, and Paul R. Prucnal. Neuromorphic photonic networks using silicon photonic weight banks. *Scientific Reports*, 7 (1):1–10, 2017.

[27] Ian A.D. Williamson, Tyler W. Hughes, Momchil Minkov, Ben Bartlett, Sunil Pai, and Shanhui Fan. Reprogrammable electro-optic nonlinear activation functions for optical neural networks. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(1):1–12, 2019.

[28] Paul J. Werbos. Backpropagation through time: What it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[29] Logan G. Wright, Tatsuhiro Onodera, Martin M. Stein, Tianyu Wang, Darren T. Schachter, Zoey Hu, and Peter L. McMahon. Deep physical neural networks trained with backpropagation. *Nature*, 601(7894):549–555, 2022.

[30] Sunil Pai, Zhanghao Sun, Tyler W. Hughes, Taewon Park, Ben Bartlett, Ian A.D. Williamson, Momchil Minkov, Maziyar Milanizadeh, Nathnael Abebe, Francesco Morichetti, et al. Experimentally realized in situ backpropagation for deep learning in photonic neural networks. *Science*, 380(6643): 398–404, 2023.

[31] Christian Leefmans, Avik Dutt, James Williams, Luqi Yuan, Midya Parto, Franco Nori, Shanhui Fan, and Alireza Marandi. Topological dissipation in a time-multiplexed photonic resonator network. *Nature Physics*, 18(4):442–449, 2022.

[32] Carsten Langrock and Martin M. Fejer. Fiber-feedback continuous-wave and synchronously-pumped singly-resonant ring optical parametric oscillators using reverse-proton-exchanged periodically-poled lithium niobate waveguides. *Optics Letters*, 32(15):2263–2265, 2007.

[33] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[34] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

[35] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2921–2926. IEEE, 2017.

[36] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[37] Alex Krizhevsky, Geoffrey Hinton, et al. *Learning multiple layers of features from tiny images*. Toronto, ON, Canada, 2009.

[38] Matthew Cook et al. Universality in elementary cellular automata. *Complex Systems*, 15(1):1–40, 2004.

[39] Xavier Porte, Anas Skalli, Nasibeh Haghighi, Stephan Reitzenstein, James A Lott, and Daniel Brunner. A complete, parallel and autonomous photonic neural network in a semiconductor multimode laser. *Journal of Physics: Photonics*, 3(2):024017, 2021.

[40] Tiankuang Zhou, Wei Wu, Jinzhi Zhang, Shaoliang Yu, and Lu Fang. Ultrafast dynamic machine vision with spatiotemporal photonic computing. *Science Advances*, 9(23):eadg4391, 2023.

[41] Mustafa Yildirim, Ilker Oguz, Fabian Kaufmann, Marc Reig Escalé, Rachel Grange, Demetri Psaltis, and Christophe Moser. Nonlinear optical feature generator for machine learning. *Apl Photonics*, 8(10), 2023.

[42] Luqi Yuan, Qian Lin, Meng Xiao, and Shanhui Fan. Synthetic dimension in photonics. *Optica*, 5(11):1396–1405, 2018.

[43] Tianyu Wang, Mandar M. Sohoni, Logan G. Wright, Martin M. Stein, Shi-Yuan Ma, Tatsuhiro Onodera, Maxwell G. Anderson, and Peter L. McMahon. Image sensing with multilayer nonlinear optical neural networks. *Nature Photonics*, 17(5):408–415, 2023.

[44] Qiushi Guo, Ryoto Sekine, Luis Ledezma, Rajveer Nehra, Devin J. Dean, Arkadev Roy, Robert M. Gray, Saman Jahani, and Alireza Marandi. Femtojoule femtosecond all-optical switching in lithium niobate nanophotonics. *Nature Photonics*, 16(9):625–631, 2022.

[45] Zaijun Chen, Alexander Sludds, Ronald Davis III, Ian Christen, Liane Bernstein, Lamia Ateshian, Tobias Heuser, Niels Heermeier, James A. Lott, Stephan Reitzenstein, et al. Deep learning with coherent vcsel neural networks. *Nature Photonics*, 17(8):723–730, 2023.

[46] Wen Zhou, Bowei Dong, Nikolaos Farmakidis, Xuan Li, Nathan Youngblood, Kairan Huang, Yuhan He, C. David Wright, Wolfram H.P. Pernice, and Harish Bhaskaran. In-memory photonic dot-product engine with electrically programmable weight banks. *Nature Communications*, 14(1):2887, 2023.

[47] Toshiaki Suhara and Masatoshi Fujimura. *Waveguide nonlinear-optic devices*, volume 11. Springer Science & Business Media, 2013.

Chapter 6

ALL-OPTICAL COMPUTING WITH BEYOND 100-GHZ CLOCK RATES

Gordon H.Y. Li*, Midya Parto*, Jinhao Ge*, Qing-Xin Ji, Maodong Gao, Yan Yu, James Williams, Robert M. Gray, Christian R. Leefmans, Nicolas Englebert, Kerry J. Vahala, and Alireza Marandi. All-optical computing with beyond 100-GHz clock rates. *arXiv preprint arXiv:2501.05756*, 2025. doi:10.48550/arXiv.2501.05756.
G.H.Y.L. conceived the project, developed the theory, executed numerical simulations, designed the experiments, built the experimental setup, performed the experimental measurements, analyzed the data, and wrote the manuscript.
* denotes equal contributions.

6.1 Abstract

A computer's clock rate ultimately determines the minimum time between sequential operations or instructions. Despite exponential advances in electronic computer performance owing to Moore's Law and increasingly parallel system architectures, computer clock rates have remained stagnant at $\sim$ 5 GHz for almost two decades. This poses an intractable problem for applications requiring real-time processing or control of ultrafast information systems. Here we break this barrier by proposing and experimentally demonstrating computing based on an end-to-end and all-optical recurrent neural network harnessing the ultrafast nature of linear and nonlinear optical operations while avoiding electronic operations. The all-optical computer realizes linear operations, nonlinear functions, and memory entirely in the optical domain with $>$ 100 GHz clock rates. We experimentally demonstrate a prototypical task of noisy waveform classification as well as perform ultrafast in-situ analysis of the soliton states from integrated optical microresonators. We further illustrate the application of the architecture for generative artificial intelligence based on quantum fluctuations to generate images even in the absence of input optical signals. Our results highlight the potential of all-optical computing beyond what can be achieved with digital electronics by utilizing ultrafast linear, nonlinear, and memory functions and quantum fluctuations.

6.2 Introduction

The clock rate ultimately determines the minimum time between sequential operations or instructions in a computer [1], and a computer cannot effectively process information or respond to input signals occurring on timescales faster than a single clock cycle. The evolution of computer hardware has been characterized by many major technological shifts: starting from early mechanical computers such as the Z1 [2] with a clock rate of 1 Hz, then progressing to general purpose electronic computers constructed from vacuum tubes such as ENIAC [3] with a clock rate of 100 kHz, and finally maturing into today's central processing units (CPUs) [1] consisting of billions of integrated silicon transistors with GHz clock rates. Each dramatic increase in clock rate throughout history has yielded countless new applications and innovations which were previously computationally infeasible.

Modern CPU clock rates have stagnated at $\sim$ 5 GHz since circa 2005 as shown in Fig. 6.1. Prior to 2005, CPU clock rates increased commensurately with Moore's Law [4]. This abrupt change is mainly due to the breakdown of Dennard scaling [5] for transistors at the device level and the increasing prevalence of the von-Neumann bottleneck [6] at the system level, which prompted CPU designers to abandon further significant increases in clock rate. Indeed, recent gains in computer performance can be largely attributed to the introduction of multi-core and other highly parallel computer architectures. Although the clock rate is not a directly comparable measure of computing speed between different families of computer processors since the instruction sets and operations during each clock cycle may differ, it remains clear that the limited clock rates of electronic computers preclude real-time processing or control of emerging ultrafast information systems at picosecond or faster timescales. This highlights a unique opportunity for optical computing, particularly where all the computational operations, i.e., linear and nonlinear functions, as well as the memory, are realized in the optical domain. Such an all-optical computing platform has been challenging to implement in a scalable and programmable fashion especially because of the limitations with the nonlinear functions and all-optical memory [7].

Optical computers have experienced a resurgence in recent years as application-specific hardware for both linear operations [8–10] and nonlinear functions [11–13] in deep learning [14–16], neuromorphic computing [17–19], and combinatorial optimization [20–22] workloads. However, previous approaches mainly prioritized energy-efficient, high-throughput, or parallel processing and still relied on digital processors or optoelectronics to perform intermediate steps of the computation,

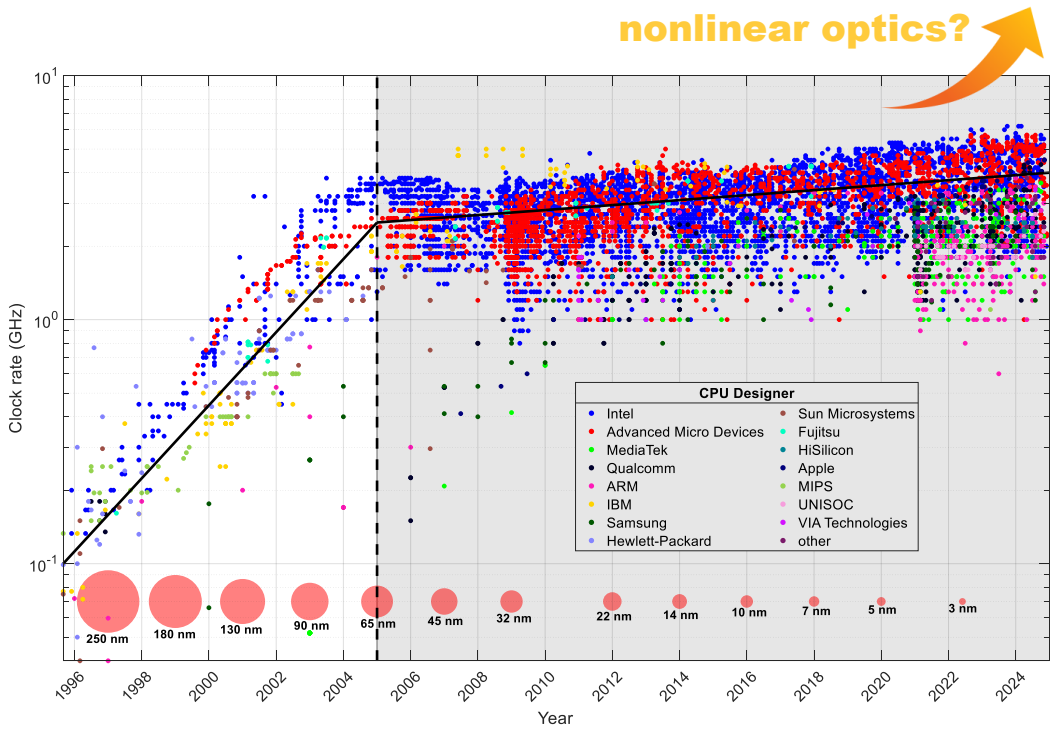

Figure 6.1: **CPU clock rates over the past 29 years.** Each point indicates the clock rate and testing date for a different type of commercially-available CPU. The colour corresponds to the entity that designed the CPU, with some prominent CPU designers listed in the legend. Red circle and text bottom insets represent increasingly advanced semiconductor process nodes and approximately when they were introduced. Prior to 2005, clock rates increased exponentially and commensurately with Moore's Law. However, clock rates have stagnated and only increased incrementally since 2005.

hence were ultimately bottlenecked by electronic response times. Additionally, several of the previous all-optical approaches [20, 23–27] suffered from one or a combination of (i) lacking crucial operations such as nonlinearity and/or memory, (ii) lack of programmability, and (iii) utilization of slow nonlinearities.

An all-optical and programmable computer that can exceed the clock rates of current electronic computers is lacking. In this work, we propose and experimentally demonstrate all-optical computing at high clock rates beyond the limitations of electronic computers by combining ultrafast nonlinear optics for nonlinear operations, interference for linear operations, and active cavities for optical memories. Ultrafast nonlinear optics provides two unique advantages over previous approaches: (i) the near-instantaneous response time of the parametric $\chi^{(2)}$ optical nonlinear-

ity is orders-of-magnitude faster than electronic nonlinearities, and (ii) the ability to generate ultrashort laser pulses, which allows for time-multiplexing and higher single-channel data-encoding rates compared to electronics or continuous-wave light. We provide several experimental examples of computing tasks including noisy waveform classification with clock rates > 100 GHz, in-situ processing of native ultrafast optical input signals, time-series forecasting, and all-optical image generation seeded from quantum noise.

6.3 Results

All-optical computer architecture

The optical computer architecture is based on a recurrent neural network (RNN) as shown in Fig. 6.2a. An RNN contains an input layer, hidden recurrent layer, and output layer [28]. Unlike purely feed-forward architectures, this kind of driven dynamical system is well-suited for temporal or sequential information processing due to the inherent memory endowed by recurrent neuron states that are propagated between successive time steps. We also note that RNNs are Turing-complete [29]. Compared to digital or von-Neumann computer architectures, the RNN architecture lends itself more naturally to ultrafast optics since it is inherently analog and utilizes a dynamical memory instead of non-volatile memory elements. We construct an experimental proof-of-concept for an all-optical recurrent neural network (AO-RNN) using off-the-shelf optical fibre components, with operating wavelength of $\lambda \approx 1.55$ μm, as shown in Fig. 6.2b. The AO-RNN is based on a time-multiplexed photonic network [20, 30, 31] in which information and input data sequences $\{s_t\}$ are encoded onto the coherent amplitude of ultrashort laser pulses. Therefore, the effective clock rate of the AO-RNN is equivalent to the laser pulse repetition rate f_c. We utilized different kinds of optical frequency combs [32] including mode-locked lasers, electro-optic frequency combs, and optical microcombs to generate input signals for different tasks. Recurrent connections between time steps are performed using an active optical cavity, which acts as an optical feedback loop. The optical cavity also contains intra-cavity linear couplings implemented using a multi-arm Mach-Zehnder interferometer in which the coupling weights are encoded using electro-optic amplitude modulators in each arm. The specific network topology of the recurrent layer is determined by the lengths of the optical delay lines in each arm, so the temporal delays T_m should be an integer multiple of the clock period $1/f_c$. In-line nonlinear activation functions are performed using a reverse-proton exchange periodically-poled lithium niobate (PPLN) waveguide [33].

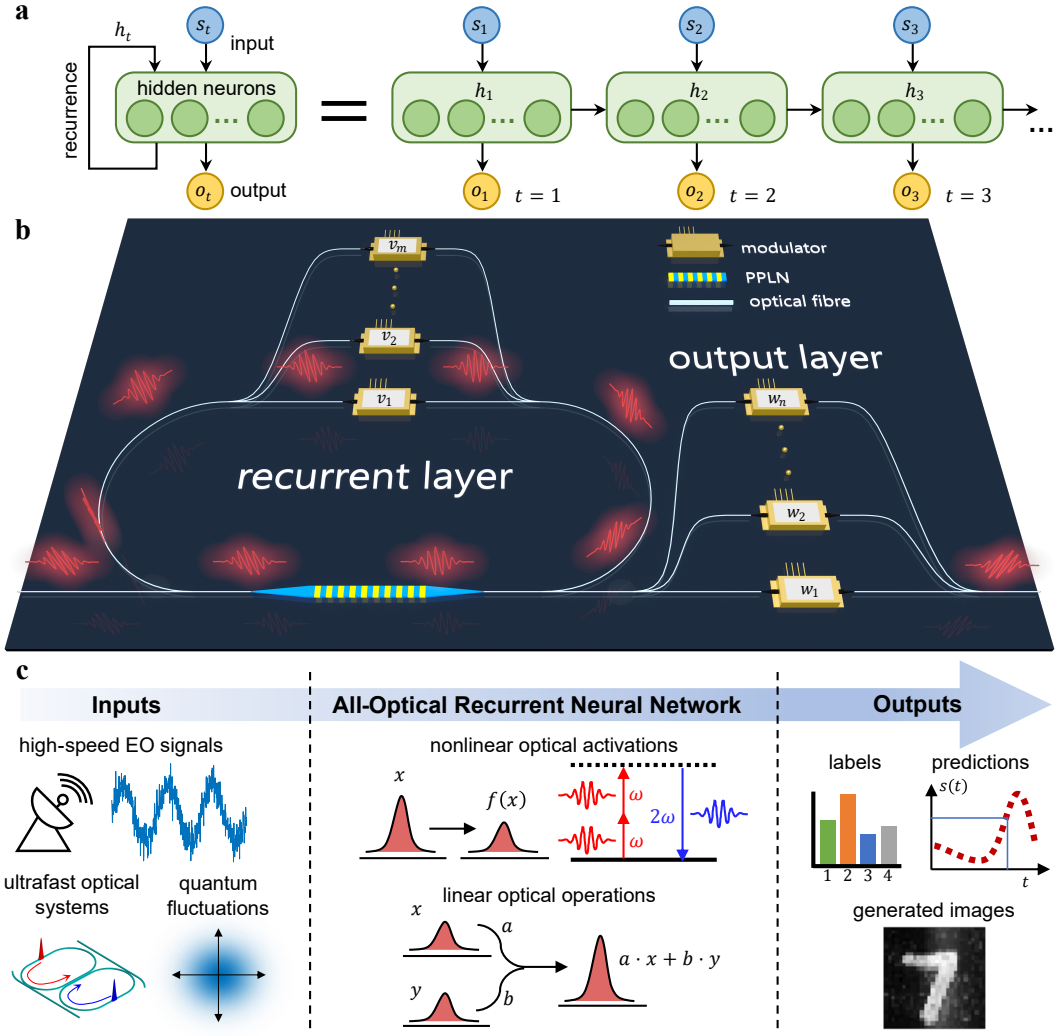

Figure 6.2: **All-optical computer architecture.** (a) A general recurrent neural network (RNN) consists of an input layer, hidden layer with recurrent connections, and output layer. (b) Schematic of the end-to-end all-optical recurrent neural network (AO-RNN) architecture based on a time-multiplexed photonic network with (c) ultrafast optical inputs undergoing optical feedback and recurrent connections, nonlinear optical activations, and linear optical operations to produce optical outputs.

The PPLN enables strong $\chi^{(2)}$ nonlinear optical processes such as pump-depleted second-harmonic generation at low pulse energies, which results in a sigmoid-like input-output function for the pulse amplitude at the fundamental harmonic [34]. Finally, the linear output layer is performed using another multi-arm Mach-Zehnder interferometer with weights encoded by electro-optic modulators and connections determined by temporal delay lines. Therefore, the AO-RNN is an end-to-end and fully-analog optical computer that accepts ultrafast optical inputs and produces

optical outputs through a combination of linear operations, nonlinear activations, and memory feedback entirely in the optical domain.

Noisy waveform classification

To test the maximum possible clock rate f_c of the experimental AO-RNN, we trained it (see Materials and Methods) to perform the prototypical task of noisy waveform classification. We consider four classes of temporal waveforms: sawtooth, triangle, square, and sine. The waveforms have the same period and duration, but are also corrupted by some random noise as shown in Fig. 6.3a. The waveform samples are fed sequentially as a single-channel input into the AO-RNN, and the task is to classify each waveform into the correct class. Therefore, this task is inherently single-threaded and cannot necessarily be accelerated through parallel processing. The AO-RNN produces a single-channel optical output pulse for each optical input pulse. To assign a single class label for verification purposes, the output pulses are photodetected, which effectively acts as a low-pass filter performing temporal global average pooling. The average optical output is compared against threshold decision boundaries (Materials and Methods) in digital postprocessing to assign a final class label. Alternatively, if multiple output channels are available, then a conventional softmax classification can also be applied. Note that the digital postprocessing required to assign class labels in this case only occurs after the entire duration of the waveform sequence, hence it only needs to be performed at a much lower rate compared to the clock rate f_c of the AO-RNN and does not bottleneck the computation. In principle, the output optical pulses containing information about the classification result could be used as the input to a second optical computer stage for further optical processing if desired.

The classification accuracy of the AO-RNN generally decreases as the clock rate f_c increases as shown in Fig. 6.3b. It achieves a peak classification accuracy of 97.5% at a clock rate of $f_c = 10$ GHz (Fig. 6.3c), which exceeds the clock rate of commercially-available CPUs. The classification accuracy decreases to 92% at a clock rate of $f_c = 50$ GHz (Fig. 6.3d) as the AO-RNN begins to confuse some sine/triangle waveforms. The classification accuracy further decreases to 58% at a clock rate of $f_c = 100$ GHz (Fig. 6.3e) as the AO-RNN also confuses square/sine and triangle/sawtooth waveforms. Even at the maximum tested clock rate of $f_c = 120$ GHz, the AO-RNN still achieves a classification accuracy significantly higher than random guessing. Repeating the same task using a purely linear model (i.e., without nonlinear activation function) with $f_c = 10$ GHz achieves a

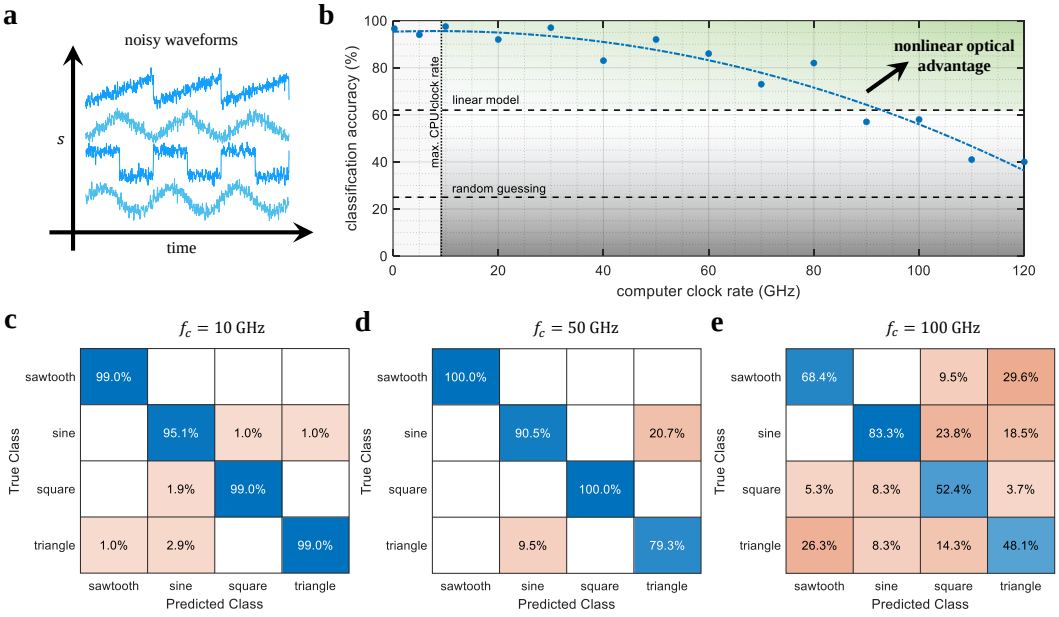

Figure 6.3: **All-optical noisy waveform classification.** (a) Temporal seqeunces of sawtooth, triangle, square, and sine waves corrupted with noise are classified by the AO-RNN. (b) The measured classification accuracy (blue dots) generally decreases as the clock rate f_c increases. The bottom horizontal dashed line corresponds to random guessing accuracy of 25%, the upper horizontal dashed line corresponds to the accuracy of a purely linear model, and the vertical dashed line indicates the maximum achievable clock rates by digital electronic computers. Therefore, a nonlinear optical computational advantage is achieved in the green-shaded region. The (column-normalized) confusion matrices are shown for clock rates of (c) f_c = 10 GHz, (d) f_c = 50 GHz, and (e) f_c = 100 GHz.

classification accuracy of only 62%, which confirms that the PPLN optical nonlinearity is important for achieving a high classification accuracy. We designate an AO-RNN with both a clock rate higher than any electronic CPU and classification accuracy higher than a linear model as exhibiting a "*nonlinear optical advantage*".

We have identified several reasons explaining why the classification accuracy decreases as the clock rate f_c increases. First, for testing purposes, we generated the input signals electro-optically using optical time-interleaving techniques (Supplementary Information Section V), which becomes increasingly more difficult as f_c increases. The AO-RNN relies on coherent interference for performing linear operations and memory feedback, hence the relative phase and temporal separation of coherent laser pulses is critically important. Therefore, the computation fidelity decreases as the phase noise and timing jitter in the input signal increases. Second, although the $\chi^{(2)}$ optical nonlinearity is of the ultrafast variety and near-

instantaneous, the PPLN possesses a finite phase-matching bandwidth ~ 100 GHz. Therefore, the effectiveness of the nonlinear activation function , and hence classification accuracy, begins to degrade when f_c exceeds the phase-matching bandwidth. Finally, neighbouring pulses begin to overlap temporally and experience undesirable cross-talk when the clock period $1/f_c$ becomes comparable to the pulse duration.

Native ultrafast optical signals

Many of the above-mentioned issues can be avoided entirely if the input data occurs natively in the optical domain, thus not requiring electro-optic or optoelectronic conversions for generating input signals. Fortunately, the study of ultrafast optical science is rife with examples of optical signals that possess a high degree of coherence and occur on timescales that are too fast for real-time processing or control by electronic computers. Here we introduce the AO-RNN as an in-situ tool for ultrafast optical science, which can potentially enable new functionalities that are infeasible using existing experimental techniques.

For example, Fig. 6.4a shows ultra-low-loss Si_3N_4 integrated coupled microres-

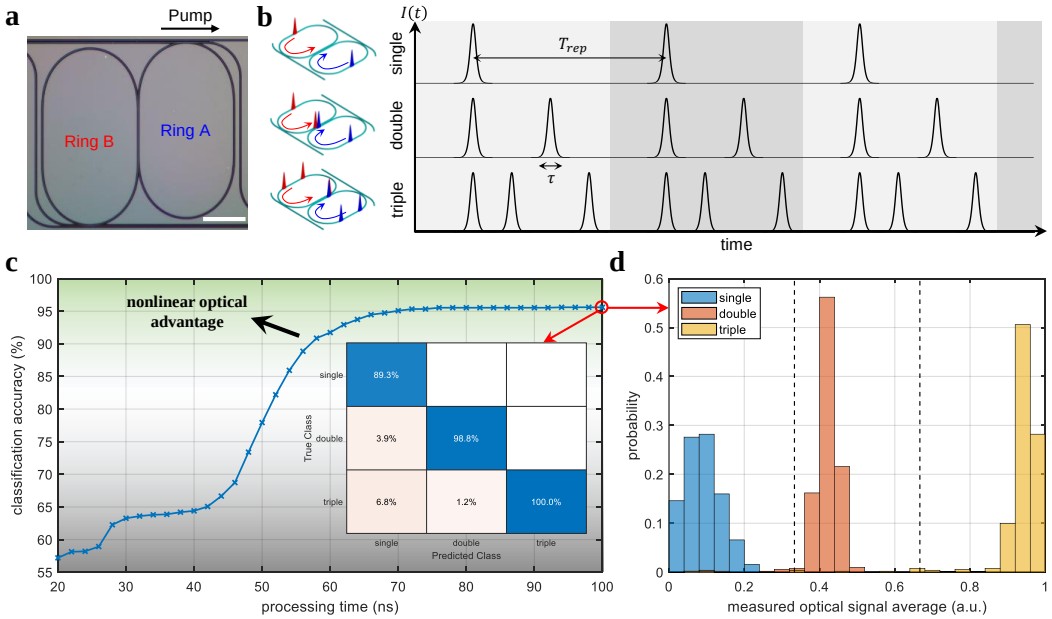

Figure 6.4: **Microcomb bipartite soliton state classification.** (a) Micrograph of a Si_3N_4 integrated coupled microresonator (white scale bar: 1 mm). (b) Bipartite bright soliton states with single, double, or triple pulse pairs. (c) Classification accuracy of soliton states with different AO-RNN processing times. Inset: confusion matrix for a processing time of 100 ns. (d) Histogram and decision boundaries for measured optical output averages with a processing time of 100 ns.

onators [35]. This configuration can be operated as an optical frequency comb or "microcomb" through the balance of $\chi^{(3)}$ optical nonlinearity and dispersion, giving rise to bipartite bright-soliton states. Microcombs have attracted immense research interest due to their technological importance in a wide range of applications including optical computing, LIDAR, dual-comb spectroscopy, low-noise microwave synthesis, optical metrology, and astrocombs [9, 10, 36]. The Si_3N_4 microcomb can produce different soliton states including a single soliton pulse pair, double soliton pulse pair, or triple soliton pulse pair (Supplementary Information Fig. S6). The output optical signal in the time-domain is a periodic waveform composed of sub-picosecond pulses with repetition rate of $f_{rep} \approx 19.97$ GHz as shown in Fig. 6.4b. We consider the task of classifying bipartite soliton states (single, double, or triple) given that the optical waveforms have the same average power, repetition rate, centre wavelength, polarization, and spectral bandwidth. This is challenging to do in real-time since the optical waveforms are too fast for direct photodetection. Moreover, conventional optical measurement techniques using spectrometers or autocorrelation often require scanning elements at millisecond or slower timescales. Other ultrafast characterization methods such as time-stretching [37–39] struggle with the combination of both high-repetition rate and high-duty cycle, which is needed to distinguish the bipartite soliton states. Existing single-shot [40] or few-shot [41] methods suffer from low frame update rates and/or require extensive postprocessing algorithms to extract the useful information. Additionally, our all-optical computing architecture can be realized as a feedback mechanism for controlling the soliton state generation and/or ultrafast sensing schemes based on similar soliton dynamics [42].

The AO-RNN can classify bipartite soliton states with varying amounts of processing time given by the length of the input optical waveform as shown in Fig. 6.4c. The final class label is assigned in the same way as for noisy waveform classification. The AO-RNN achieves a high classification accuracy of 95.6% for processing times shorter than 100 ns. The minimum processing time or latency is limited by the main cavity roundtrip light propagation time in the AO-RNN, which is $\sim$ 24 ns. We observe that multiple cavity roundtrips are necessary to ensure high classification accuracy. This latency is a consequence of using relatively long optical-fibre components/connectors, and can be drastically reduced by direct splicing or instead using a corresponding integrated photonic platform with sub-nanosecond roundtrip times. Therefore, it is possible to use the AO-RNN as an in-situ tool for near-real-time and few-shot classification of optical soliton states, which can enable faster measurement and feedback control loops.

Time-series forecasting

The previous two example tasks were classification tasks. We further show that the AO-RNN can also be trained to perform regression tasks such as time-series forecasting with a high clock rate. It is desirable to have faster regression methods to make real-time decisions in many applications including quantitative finance [43], experimental particle physics [44], and optical signal processing [45]. In this task, samples of a time-series are encoded onto the coherent amplitude of optical pulses and inputted one at a time into the AO-RNN. The task is to predict the next value in the time-series given the current input value and past inputted values. The corresponding output pulse should have an amplitude representing the predicted value of the next time step sample.

We show two examples of time-series forecasting for triangle and sine waveforms in Fig. 6.5. The AO-RNN predictions show close agreement with the target values for these simple waveforms, and achieves normalized mean square error (NMSE) as low as 0.0144 and 0.0094, respectively, up to a clock rate of $f_c = 10$ GHz. Unlike for classification tasks in which the speed of the output measurement can be amortized over the entire duration of the time-series, regression tasks require single-shot and rapid output measurements. In this case, our ability to test and verify the output predictions is limited by the maximum bandwidth of our photodetector that is ~ 25 GHz. Although the AO-RNN can operate with far higher clock rates, as evidenced by the noisy waveform classification task, we were experimentally limited in accurately measuring the real-time pulse-to-pulse output values beyond a clock rate of $f_c = 10$ GHz. This may be improved by using a faster photodetector up to ~ 100 GHz and better data acquisition tools. Nevertheless, this is not a fundamental limitation of the AO-RNN, but rather a constraint of our current output measurement

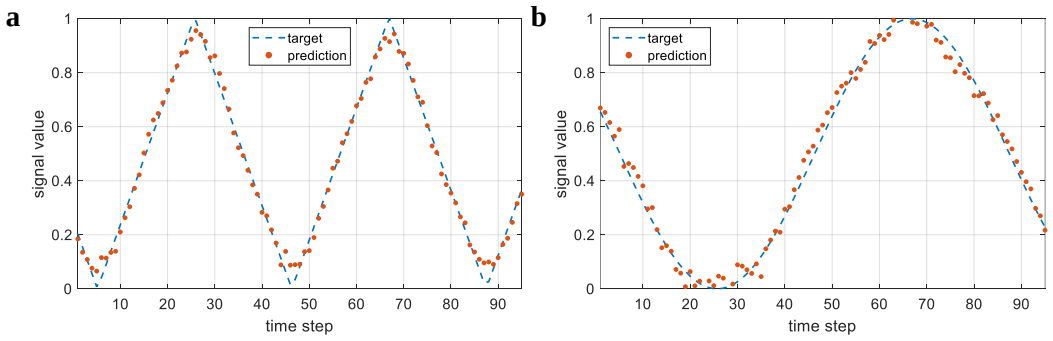

Figure 6.5: **Temporal waveform prediction.** Predicted (dots) one-step-ahead values for target (dashed lines) (a) triangle and (b) sine input time-series.

techniques. If an output remaining in the optical domain is sufficient, then the clock rate may be much higher since the optical outputs can in principle directly serve as optical inputs for another optical computer or actuator.

Image generation seeded from quantum fluctuations

Finally, we demonstrate an example of a generative task in which the AO-RNN can use quantum fluctuations as the seed to generate images in the absence of any input optical signals [46, 47]. The central problem for generative models is to learn a complicated unknown target distribution from which samples (e.g., images) are available, and then to use the model to efficiently generate new samples from the target distribution. We take inspiration from recent advances in generative artificial intelligence based on diffusion [48] and flow-based [49] models. In this case, a simple known distribution (typically a standard Gaussian distribution) is continuously perturbed to match the unknown target distribution. Then, the learned mapping can be applied to an initial random sample from the simple distribution to generate a new sample from the target distribution.

We trained the AO-RNN using the MNIST handwritten digits dataset [50] to generate 28×28 greyscale images of the class "seven" as shown in Fig. 6.6a. For this task, we ignore the input and output layers of the AO-RNN and only directly utilize the recurrent layer. The main cavity contains an optical amplifier that acts as a programmable gain/loss mechanism. The optical amplifier also supplies quantum noise in the form of spontaneous emission, which approximately follows a Gaussian distribution (Supplementary Information Fig. S9). Therefore, the AO-RNN can be interpreted as a highly nonlinear laser cavity. In conventional lasers, spontaneous emission in the gain medium serves as the spark to initiate stimulated emission, whereupon population inversion and gain-clamping after the laser threshold (i.e., gain equals loss) is exceeded will lead to a steady-state laser emission [51]. Here we modulate the gain and intra-cavity connection weights at a clock rate f_c that is much faster compared to the cavity roundtrip time. Therefore, the lasing dynamics are also highly non-equilibrium. We define cavity modes by virtually splitting the cavity into equally-spaced time bins. Each time bin corresponds to a different pixel location in the image and the average power contained in each time bin encodes for the pixel greyscale intensity. Upon initially turning on, the AO-RNN starts from spontaneous emission in the optical amplifier and then gradually generates a macroscopic image after $T = 100$ cavity roundtrips by effectively controlling which cavity modes temporarily go above/below lasing threshold.

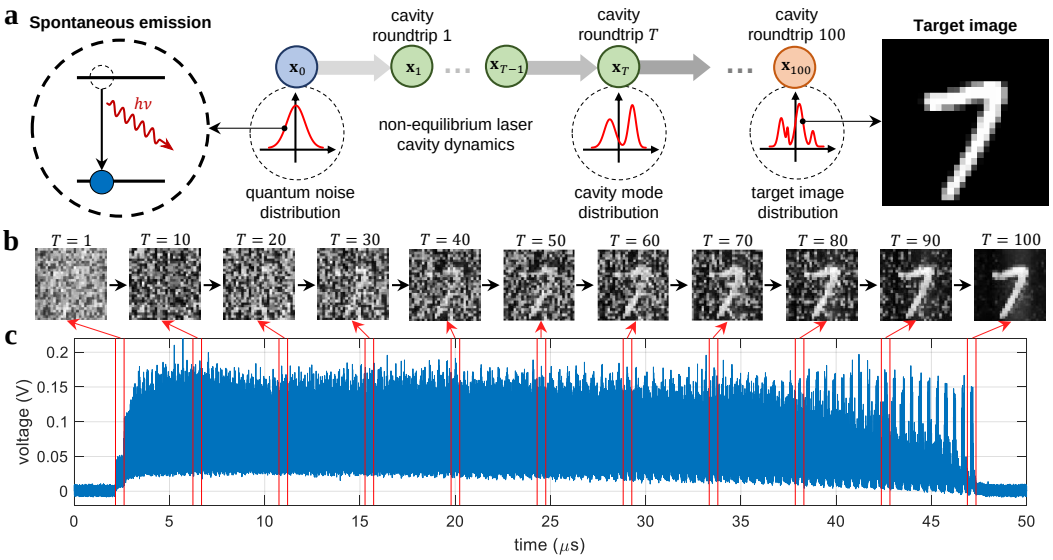

Figure 6.6: **Quantum all-optical image generation.** (a) An all-optical image generator maps a quantum noise distribution from spontaneous emission into an unknown target image distribution of MNIST handwritten digits using highly non-equilibrium laser cavity dynamics. (b) Example images after every 10 cavity roundtrips and (c) measured cavity output time trace up to $T = 100$ roundtrips representing a generated training sample of a seven.

An example time trace of the cavity dynamics and resultant image generated of a sample during the training process is shown in Fig. 6.6b, c. Examples of new images (i.e., not part of the original MNIST dataset) of sevens generated after the training process and seeded by quantum noise are shown in Supplementary Information Fig. S10. We note that our simple generative AO-RNN lacks model expressiveness and struggled to generate high-quality images of multiple different digit classes or learn more complicated image distributions beyond the MNIST dataset. We believe that the quality and diversity of generated sevens from this simple proof-of-concept model is promising, and better model expressiveness can be achieved through exploring more complicated AO-RNN architectures.

6.4 Discussion

There are a few key timescales limiting the maximum allowable clock rate of the AO-RNN. We opted to use off-the-shelf optical fibre components for simplicity, however, this was not ideal for optimal computer performance. For example, the PPLN used for nonlinear activations based on a weakly-guiding reverse-proton exchange waveguide [33] has a relatively small phase-matching bandwidth of $\sim$ 100 GHz. However, we previously demonstrated higher-performance nonlinear activation func-

tions using thin-film lithium niobate (TFLN) with maximum allowable clock rates > 13 THz [11]. Therefore, the AO-RNN could greatly benefit from on-chip integration using TFLN. Furthermore, other devices demonstrated in TFLN, such as high-speed electro-optic modulators with > 100 GHz bandwidth [52] or all-optical switches with speeds > 10 THz [12] can enable faster real-time updating of weights. The longest duration laser pulse width used in our experiments was ~ 5 ps, which limits the maximum allowable clock rates since pulses will begin overlapping in time and suffering from undesirable cross-talk beyond ~ 200 GHz clock rates. This issue can be overcome by using even shorter laser pulses, for example, few-cycle pulses generated using nonlinear optical pulse compression [53] in TFLN. Finally, implementing the main cavity in the AO-RNN using an integrated optical parametric oscillator [54–56] in TFLN could also drastically reduce the overall latency given by the light propagation time through the network.

The conventional definition of clock rate may not be directly applicable to some kinds of asynchronous or clock-less analog computers [15, 57]. Here we use the concept of clock period more generally to mean the minimum time between successive computer operations. This concept can thus still apply to clock-less processors since there is a characteristic physical timescale associated with each nonlinear device (electronic, optoelectronic, optical, or otherwise). The key advantage of the AO-RNN is that it implements all linear operations, nonlinear activations, and memory feedback directly in the optical domain so that all operations can simultaneously benefit from exceptionally high clock rates without an electronic bottleneck for some parts of the computation. During each clock period, the experimental AO-RNN performs a maximum of 8 operations (1 nonlinear activation and 7 multiply-accumulate operations). This number can be further increased by including more optical delay lines and modulators, as well as utilizing wavelength and/or spatial multiplexing techniques [31]. We believe that the most useful near-term applications for this kind of ultrafast optical computer will be those in which the input signal occurs natively in the optical domain, hence bypassing the need for electro-optic input signal generation. Some prime examples include in-situ analysis of ultrafast imaging and spectroscopy data [58, 59], optical signal processing for high-speed coherent telecommunications systems [45, 60], and precision ranging or LIDAR using femtosecond lasers [61, 62].

In conclusion, we have harnessed ultrafast nonlinear optics to build a new kind of end-to-end optical computer that can surpass the limited clock rates of existing digital

electronic computers. The proof-of-concept experimental results for a combination of classification, regression, and generative computational tasks demonstrate that the AO-RNN can achieve clock rates > 100 GHz, which is more than an order-of-magnitude improvement over current digital computers. This work highlights a new regime for ultrafast optical computing, enabling nascent applications requiring real-time information processing and feedback control at picosecond timescales.

6.5 Materials and Methods

Experimental setup

A detailed schematic of the experimental setup for the AO-RNN is shown in Supplementary Information Fig. S1. Wherever possible, we used commercially-available, single-mode polarization-maintaining (PM) optical fibre components with centre operating wavelength of $\lambda \approx 1.55$ μm. The recurrent layer consists of a main cavity and contains an intra-cavity Mach-Zehnder interferometer with two arms. The relative temporal delay T_0 between the arms of the Mach-Zehnder interferometer determines the connection topology, and is a hyper-parameter of the AO-RNN to be chosen depending on the task. For proper operation, T_0 is chosen to be an integer multiple of the clock period $1/f_c$. The relative delay is fine-tuned using a homemade free-space delay stage since it is difficult to cut/splice optical fibers to precisely match the desired length. The approximate temporal delay is set by propagating a single reference laser pulse and manually moving the stage. Then, fine-tuning is done by a high-precision linear micrometer stage to maximize the pulse temporal overlap by observing the coherent interference fringe visibility. The weights are set using electro-optic amplitude modulators (IXBlue MXAN-LN-10) in each arm of the Mach-Zehnder interferometer. Each modulator accepts a constant bias voltage input to set the operating point, and also a high-speed RF voltage input to rapidly modulate the amplitude of optical signals. Bias voltages are controlled using a ADC/DAC (National Instruments 782258) and RF waveforms up to 12 GSa/s are generated using an arbitrary waveform generator (Keysight M8190A). The voltages are the learnable weight parameters of the AO-RNN. We only trained the bias voltages for the noisy waveform classification, optical soliton classification, and time-series forecasting tasks. We set the bias point to "closed" for the image generation task and trained the RF voltages at a 10 GSa/s update rate. The main cavity additionally contains a fibre-coupled PPLN waveguide [33] with a wavelength-division multiplexer on both the input/output to separate the fundamental harmonic ($\lambda \approx 1.55$ μm) and second harmonic ($\lambda \approx 0.775$ μm). The second

harmonic signal is only used to monitor the phase-matching of the PPLN, and is not used as part of the AO-RNN computation. The phase-matching of the PPLN is adjusted using a heater stage and thermocouple controller (Omega CSC32), with an optimal operating temperature around $\sim 51.5°$C. The maximum average optical power in the main cavity must be < 100 mW to avoid photo-induced damage to the PPLN. The PPLN was bypassed for the noisy waveform classification task using a purely linear model. A booster optical amplifier (Thorlabs S9FC1004P) was used to set the overall gain/loss in the main cavity for the image generation task. The amplifier was bypassed for the noisy waveform classification, optical soliton classification, and time-series forecasting tasks since we desired to have a fading memory property for these tasks. The main cavity roundtrip light propagation time was ~ 24 ns for the noisy waveform classification, optical soliton classification, and time-series forecasting tasks. The main cavity roundtrip light propagation time was ~ 453 ns for the image generation task since we required more time bins for a sufficient number of pixels to generate MNIST images. For the image generation task, the output of the recurrent layer was directly detected since the output layer is not used. For the other tasks requiring an output layer, the optical output of the recurrent layer is amplified using an erbium-doped fibre amplifier (Thorlabs EDFA100S) and filtered through a 200 GHz band-pass filter to minimize the amplified spontaneous emission noise before entering the output layer. The output layer consists of a four-arm Mach-Zehnder interferometer of similar construction as the one in the recurrent layer. Three arms encode the output layer weights, and the fourth arm is used as an optional constant optical bias. The relative delays, $T_1 \approx 0.3$ ns and $T_2 \approx 1.6$ ns, are hyper-parameters of the AO-RNN to be chosen depending on the task. Only the constant bias voltages were trained for the output layer modulators due to the limited number of available high-speed arbitrary waveform generator channels. Optical outputs are detected using a high-speed photodetector (Newport 1414) and stored on a real-time oscilloscope (Keysight MSOV334A) for post-processing if necessary. The high-speed RF and optical inputs are synchronized using a low-noise 10 MHz reference clock. The relative delay between generated RF and measured optical signal is calibrated using the sample marker output from the arbitrary waveform generator. We use a backwards-propagating locking reference, which is tapped from the unmodulated input laser source, to perform active phase-stabilization of the temporal delay lines. The recurrent layer and output layer each contain independent backwards-propagating optical locking loops based on a Pound-Drever-Hall scheme [63]. Slow photodetctors (Newport 2053) are used to

measure the backwards-propagating locking signals. The electronic locking signals are input to proportional-integral derivative controllers (Red Pitaya STEMlab 125-14) and outputs are amplified (Thorlabs MDT693B) to drive fibre phase-shifters (General Photonics FPS-002-L) that actively stabilize the relative phase in each optical delay line.

Input signals

We used a variety of different optical frequency combs to generate optical input signals for the experimental tasks. External optical signals input to the AO-RNN are gated using an electro-optic amplitude modulator, which is biased closed so that signals only enter the AO-RNN when desired to mark the start of a computation. For noisy waveform classification, the input waveforms were generated electro-optically using a high-speed arbitrary waveform generator and electro-optic amplitude modulators. The laser source for clock rates $f_c \in [250\,\text{MHz}, 5\,\text{GHz}]$ was a fibre mode-locked laser (MenloSystems FC1500-250-WG) and the laser source for clock rates $f_c \in [10\,\text{GHz}, 120\,\text{GHz}]$ was a homemade electro-optic frequency comb (Supplementary Information Section III). We can perform real-time input generation up to clock rates of $f_c \approx 48$ GHz using optical time-interleaving (Supplementary Information Section V), and offline (i.e., prepared ahead-of-time) input generation up to clock rates of $f_c \approx 200$ GHz using an asynchronously-pumped cavity. Each noisy waveform (sawtooth, triangle, square, and sine) was 120 periods in duration with a total of 5120 equally-spaced samples. The period for all noisy waveform classes was the same, and sawtooth/square waves had a duty cycle of $1/2$. The ideal noiseless waveforms had normalized amplitudes in the range [-1,1] and the measured optical noise for the waveforms is approximately given by an additive Gaussian distribution with zero mean and standard deviation of ~ 0.158 as shown in Supplementary Information Fig. S2. The Si_3N_4 coupled optical microresonators used to generate bipartite soliton states is of the same design as in Ref. [35]. We maintained the average input optical power to the AO-RNN at ~ 5.3 mW for all bipartite-soliton states by monitoring on a slow thermal power metre (Thorlabs PM20). We carefully characterized the single, double, and triple pulse pairs by separately measuring the optical spectrum (Yokogawa AQ6370C), autocorrelation (Femtochrome FR-103XL), and RF beat-note (Rhode & Schwarz FSW85) of each state as shown in Supplementary Information Fig. S4. The input signals used for the time-series forecasting task were also generated electro-optically in the same way as for the noisy waveform classification. The triangle wave had 44 samples per period

with a duty cycle of $1/2$. The sine wave had the same sample time-spacing as the triangle wave but double the period.

RNN model

The general RNN architecture is given by Eq. 6.1:

$$h_i(t+1) = f_i\left(\sum_{j=1}^{N} W_{ij}^{r} \cdot h_j(t) + \sum_{j=1}^{m} W_{ij}^{in} \cdot s_j(t)\right) \tag{6.1a}$$

$$o_k(t) = \sum_{l=1}^{N} W_{kl}^{out} \cdot h_l(t) \tag{6.1b}$$

where $t \in \mathbb{N}$ is the discrete-time step (one time-step represents one clock period), $\mathbf{h} \in \mathbb{R}^N$ is the N-dimensional hidden recurrent layer activation, $\mathbf{s} \in \mathbb{R}^m$ is the m-dimensional input sequence, $\mathbf{o} \in \mathbb{R}^n$ is the n-dimensional output value, $\mathrm{W}^{\mathrm{r}} \in \mathbb{R}^{N\times N}$ is the matrix of recurrent layer weights, $\mathrm{W}^{\mathrm{in}} \in \mathbb{R}^{N\times m}$ is the matrix of input layer weights, $\mathrm{W}^{\mathrm{out}} \in \mathbb{R}^{n\times N}$ is the matrix of output layer weights, and $f_i : \mathbb{R} \to \mathbb{R}$ is an element-wise activation function for $i = 1, 2, \ldots, N$ and $k = 1, 2, \ldots, n$. We give a simplified model of the AO-RNN, which is similar to the rotating neuron architecture proposed in Ref. [64], by dividing the main cavity into equally-spaced time bins containing pulses. The hidden recurrent layer has a cyclic structure with weights:

$$W_{ij}^{r} = \begin{cases} \alpha_i(t), & \text{if} \quad i - j \equiv 1 \pmod{N} \\ \beta_j(t), & \text{if} \quad i = i_r \text{ and } j \in T^r \\ 0, & \text{otherwise} \end{cases} \tag{6.2}$$

where $\{\alpha_i(t)\}$ are weights representing the loss/gain factor for the pulse propagating from time bin i to time bin $(i+1) \bmod N$. The weights $\{\beta_j(t)\}$ represent the intra-cavity couplings between time bin i_r and other time bins in T^r. The set T^r of time bin indices represents the choice of optical delay lines in the intra-cavity Mach-Zehnder interferometer. In our experimental AO-RNN, $T^r = \{j_r\}$ represents a single connection since we used a two-arm Mach-Zehnder interferometer. This cyclic structure is a special case of the fully-connected RNN model and is still Turing-complete, however, in practice may lead to reduced model expressiveness for some tasks. We used constant weights for the noisy waveform classification, optical soliton classification, and time-series forecasting tasks, and time-varying weights for the image generation task. Our AO-RNN has single-channel input/outputs, $m = n = 1$, due to experimental constraints. In this case, the input weights are

given by $\mathrm{W}^{\mathrm{in}} = \varepsilon \mathbf{e}_{i_0}$ where $\mathbf{e}_k$ is the k^{th} unit vector, i_0 is the index of the time bin coupled to the input line, and ε is the input coupling factor. We treat the input scaling ε as a hyper-parameter that is not trained. However, it is also possible to employ more complicated input masking techniques, such as in previous time-multiplexed photonic reservoir computers [25, 26]. The output weights are given by $W_{1l}^{out} = \gamma_l(t)$ if $l \in T^{out}$, and 0 otherwise, where $\{\gamma_l(t)\}$ are determined by the output layer modulators and T^{out} is the set of time bin indices representing the choice of optical delay lines in the output layer Mach-Zehnder interferometer. For the noisy waveform and optical soliton classification tasks, we additionally perform temporal global average pooling of the optical power to yield a single output value $y = \langle |o(t)|^2 \rangle_{0<t\leq L}$, over the entire input sequence of length L, and normalize the output averages $\{y\}$ to the range $[0, 1]$. The predicted class label is assigned by comparing y against threshold decision boundaries $\mathbb{Z}_q/q$ where q is the number of classes, such that y belongs to class $p \in \mathbb{Z}_q$ if $(y \geq p/q) \wedge (y < (p+1)/q)$. Alternatively, the more conventional softmax classification can be used if the number of output channels in the AO-RNN equals the number of classes. For the image generation task, only the recurrent layer is used. The input sequence can be replaced by an additive noise term (Supplementary Information Fig. S9), representing the amplified spontaneous emission from the optical amplifier, and the output samples are then equivalent to sampling from a single point in the cavity. Strictly speaking, the nonlinear laser cavity becomes a continuous-field distribution so the concept of discrete time bins is not well-defined. However, we discretize the cavity field based on the fast output measurement sampling time τ. Then, during each cavity roundtrip, we assign the value p_v for pixel $v \in \mathbb{Z}_V$ in an image sequence with V pixels by coarse-graining the fast output samples into slower time bins with period t' such that $p_v = \langle |o(\tau)|^2 \rangle_{vt'\leq\tau<(v+1)t'}$. Pixel values are rescaled to be in the range $[-1, 1]$ for each cavity roundtrip. We used a time bin period of $t' = 0.4$ ns and applied symmetric zero-padding for unused pixels in each roundtrip ($\sim$ 453 ns) since MNIST images only contain 784 pixels.

Training procedure

We use a model-agnostic forward-only training algorithm based on the method proposed in Ref. [16]. For each training iteration, we perform the following steps:

1. Choose a random direction vector $\Delta \in \{+\delta, -\delta\}^d$ where d is the number of trainable model parameters, the elements of Δ are sampled from a Bernoulli distribution $\Delta_i \sim B(1/2)$ for $i = 1, 2, \ldots, d$, and δ is the step size.

2. Perturb the model parameters $\Theta \in \mathbb{R}^d$ by Δ and perform a forward-pass through the model to evaluate the loss function $\mathcal{L}(\Theta + \Delta)$.

3. Perturb the model parameters Θ in the opposite direction $-\Delta$ and perform a forward-pass through the model to evaluate the loss function $\mathcal{L}(\Theta - \Delta)$.

4. Estimate the directional derivative of the loss as:

$$\nabla_\Delta \mathcal{L}(\Theta) \approx \frac{\mathcal{L}(\Theta + \Delta) - \mathcal{L}(\Theta - \Delta)}{2\|\Delta\|} \,. \tag{6.3}$$

5. Update the model parameters: $\Theta \rightarrow \Theta - \eta \nabla_\Delta \mathcal{L}(\Theta)\Delta$ where η is the learning rate.

For the experimental AO-RNN, the forward-pass steps are performed directly in the optical hardware, but the other training steps are performed on a digital computer. During testing, the trained parameters are frozen and so the forward-pass inference is all-optical. The model paramters in our experimental AO-RNN correspond to electro-optic modulator voltages (both DC and RF). The half-wave voltages are $V_\pi \approx 6$ V and we found that a perturbation step-size of $\delta = 0.02$ V with learning rate of $\eta = 2 \times 10^{-3}$ was adequate for all our experimental tasks. For the noisy waveform and optical soliton classification tasks, we used a mean squared error loss $\mathcal{L} = \langle (y - z)^2 \rangle$ where $z = (2p + 1)/(2q)$ is the midpoint of the decision boundaries for the true class label. For the time-series forecasting task, we used a mean squared error loss $\mathcal{L} = \langle [o(t) - s(t + 1)]^2 \rangle_t$ for the one-step-ahead prediction. For each training sample in the image generation task, we used a diffusion process to generate intermediate target values during each cavity roundtrip T:

$$\mathbf{z}_{T-1} = \sqrt{1 - \sigma_T} \cdot \mathbf{z}_T + \sqrt{\sigma_T} \cdot \epsilon \tag{6.4}$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is standard Gaussian noise, $\mathbf{z}_{100}$ is the ideal target from the image dataset, and σ_T is a noise-variance schedule that increases linearly each from $\sigma_{100} = 0.001$ to $\sigma_1 = 0.02$. We used a mean-squared error loss $\mathcal{L} = \langle [\mathbf{x}_T - \mathbf{z}_T]^2 \rangle_{t',T}$ taken over all roundtrips and time bins for each sample where $\mathbf{x}_T$ are the measured pixel values for each roundtrip T. The training and testing sample sizes for each task are shown in Table. 6.1.

CPU clock rates

Each scatter point in Fig. 6.1 represents the clock rate and testing date for a different commericially-available CPU. The data was collected from a variety of online CPU benchmarking sources [65, 66] and press/product release information from

task	batch size	iterations	testing size
noisy waveform classification	4	200	800
optical soliton classification	10	150	1500
time-series forecasting	460	200	9200
all-optical image generation	12	1200	-

Table 6.1: **Sample sizes for experimental AO-RNN tasks.**

prominent CPU designers. We consider CPUs with the same architecture design but different generations, performance tiers, or optimizations as different processors. For each processor, we show the maximum reported clock rate, including possible turbo clock rates. For processors containing multiple cores operating at different clock rates, we show the clock rate of the fastest core. We restrict our attention to general purpose CPUs designed for desktops, laptops, servers, tablets, smartphones, wearable devices, etc; however, we exclude application-specific hardware accelerators such as graphics or tensor processing units, which typically have far lower clock rates compared to CPUs. The maximum CPU clock rate used for the dashed vertical line in Fig. 6.2b is based on the current (as of 14 March 2024) CPU-Z overclocking world-record of 9117.75 MHz [67].

6.6 Supplementary Information

Experimental setup

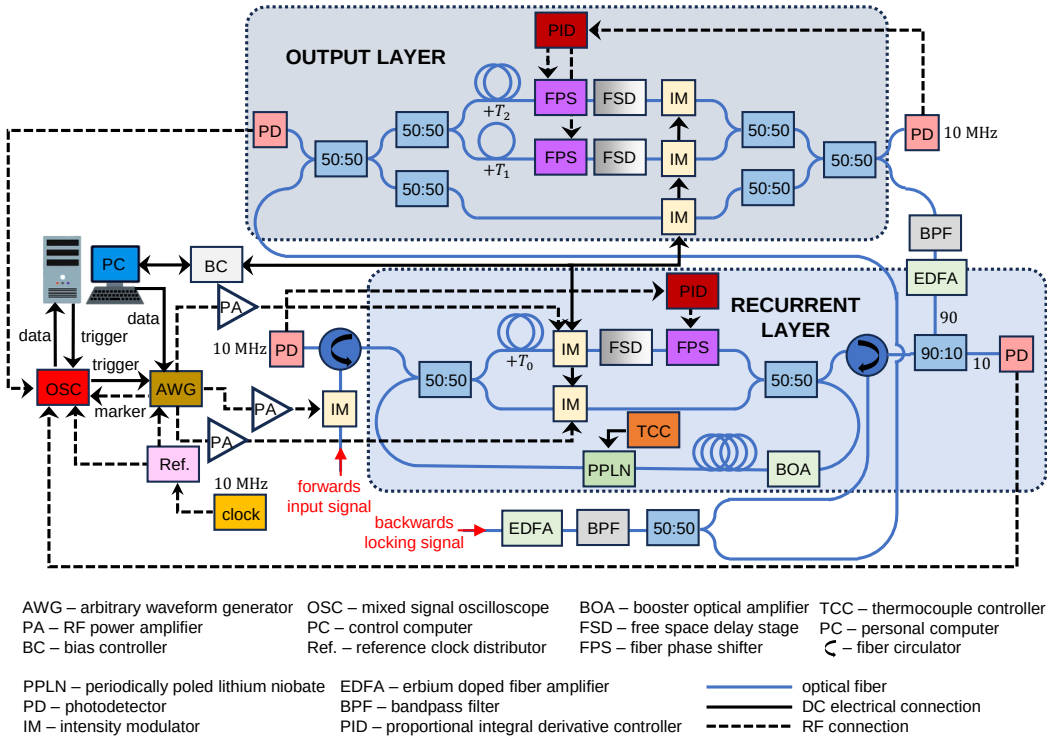

Figure 6.7: **Detailed schematic of AO-RNN experimental setup.**

Noisy waveforms

Electro-optic frequency comb

The physical configuration of the electro-optic frequency comb system is illustrated in Fig. 6.9. A continuous-wave, single-frequency laser is phase-modulated by three cascaded modulators, generating a series of sidebands that form a frequency comb, as shown in Fig. 6.10(a). The spacing between adjacent comb lines is determined by the radio-frequency (RF) signal driving the phase modulators, which is set to 10 GHz in our experiment. To maximize the number of comb lines, the RF signals driving the individual modulators are amplified and phase-shifted to ensure effective in-phase modulation. Following phase modulation, an additional intensity modulation stage is introduced to flatten the comb spectrum. The intensity modulator (IM) is biased at the half-power point of its transmission curve, and the modulation signal frequency matches that of the phase modulators. The modulation strength is controlled using a variable RF attenuator. Subsequently, the comb is amplified using an erbium-doped fiber amplifier (EDFA) to facilitate further operations with enough power. To generate optical pulses in the time domain, the phase of each comb line is adjusted using a programmable waveshaper, which applies second-order dispersion

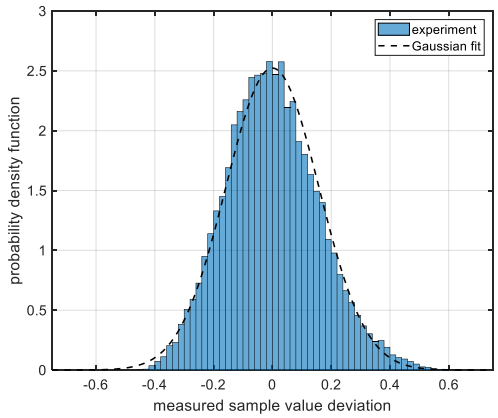

Figure 6.8: **Noise distribution for noisy waveform input signals.** The histogram shows the statistics for the deviation between the ideal noiseless waveform samples and the measured optical input signal values (amplitudes normalized to [-1,1]) for 8000 input waveform sample points. It is well-approximated by a Gaussian distribution (dashed black line) with zero mean and standard deviation of ~ 0.158.

compensation to the frequency comb. After phase compensation, the time-domain field becomes a series of optical pulses, characterized by an autocorrelator, as shown in Fig. 6.10(b). The dispersion applied by the waveshaper is optimized by minimizing the pulse width measured through autocorrelation.

Soliton microcomb

The coupled-ring device is fabricated using an ultra-low-loss Si_3N_4 platform [68]. It consists of two partially coupled racetrack resonators with slightly different free-spectral ranges (FSRs), which provide anomalous dispersion required for bright soliton generation. The laser output is modulated by a fast single-sideband modulator (QPSK) and subsequently amplified by an erbium-doped fiber amplifier (EDFA).

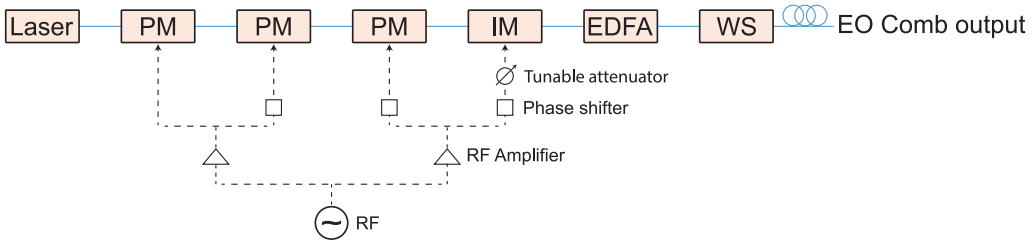

Figure 6.9: **Experimental setup for Electro-optic frequency comb.** Abbreviations: PM, phase modulator; IM, intensity modulator; EDFA, erbium-doped fiber amplifier; WS, waveshaper.

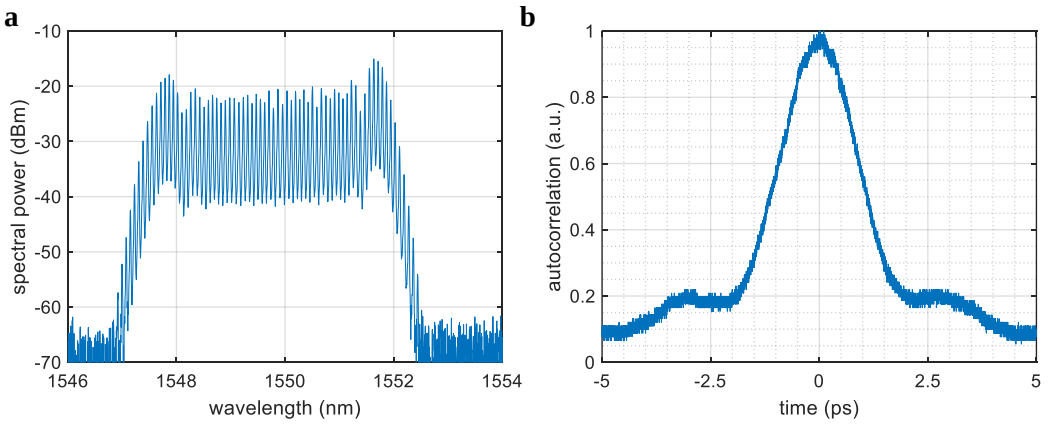

Figure 6.10: **Electro-optic frequency comb characterization.** (a) Optical spectrum and (b) autocorrelation for the electro-optic frequency comb operating with a repetition rate of 10 GHz.

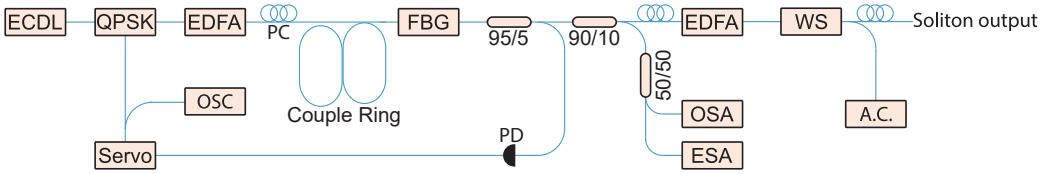

Figure 6.11: **Experimental setup for generating the multi-soliton state in coupled-ring resonators.** Abbreviations: ECDL, external cavity diode laser; QPSK, quadrature phase shift keying; EDFA, erbium-doped fiber amplifier; FBG, fiber Bragg grating; PC, polarization controller; PD, photodetector; OSA, optical spectrum analyzer; ESA, electrical spectrum analyzer; WS, waveshaper; OSC, oscilloscope; A.C., autocorrelator.

The QPSK modulator facilitates the rapid frequency sweeping of the pump laser in order to avoid thermal effect during the soliton locking process [69]. Lens fibers (not shown in Fig. 6.11) are used to couple light in and out of the coupled-ring system. The through-port output is filtered using a fiber Bragg grating (FBG) to separate the comb and pump signals. The filtered signal is then split into multiple beams for different purposes. One beam is directed to a photodetector, which measures the comb power that are used for stabilizing the pump-resonance detuning. Another beam is sent to an optical spectrum analyzer (OSA) and an electrical spectrum analyzer (ESA) for characterizing soliton spectrum and RF beat-note signal. The remaining beam, carrying the main comb power, is amplified by an EDFA and then passed through a waveshaper to compensate the fiber dispersion. The amplified soliton is then directed to the AO-RNN (all-optical recurrent neural network) setup.

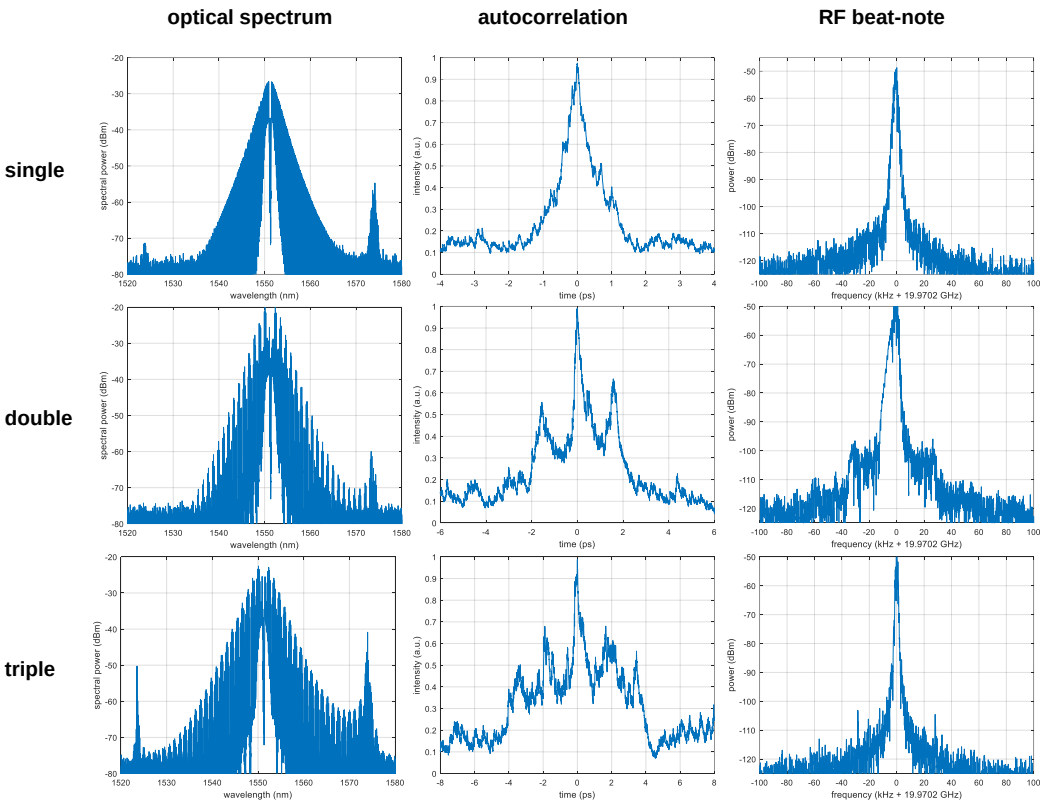

Figure 6.12: **Bipartite-soliton pulse pair state characterization.** The optical spectrum (left column), autocorrelation (middle column), and RF beat-note (right column) for the single (top row), double (middle row), and triple (bottom row) bipartite-soliton pulse states.

When single, double, or triple soliton pulse pairs are formed in the resonator, the measured total comb power varies accordingly. By selectively stabilizing specific comb power levels, different stable soliton states can be reliably and individually locked for classification experiments, as shown in Fig. 6.12. For comparative analysis, the multi-soliton states are further characterized and validated using optical spectrum measurements and autocorrelation signals. These characterization results provide a detailed basis for the classification experiments conducted with the AO-RNN setup. The soliton spectrum and autocorrelation traces for different pulse numbers are shown in Fig. 6.12. The soliton spectrum exhibits an approximately $sech^2$ envelope shape, with dispersive waves [70] appearing at frequencies where the mode and comb frequencies coincide, leading to resonant power enhancement. The soliton repetition rate, measured using the ESA, is approximately 19.97 GHz with a resolution bandwidth of 1 kHz. The autocorrelation of the generated periodic soliton pulse train is measured using an autocorrelator, and the result is shown in

Fig. 6.12. For single, double, and triple soliton states, the autocorrelation traces display 1, 3, and 5 peaks, respectively. As an additional note, the formation location of solitons within the resonator is a random process and is not actively controlled during the experiment (only the number of solitons is controlled). The data presented in Fig. 6.12 are selected as representative results.

Ultrafast optical input time-multiplexing

The signal information in the AO-RNN is encoded onto the coherent amplitude of ultrashort laser pulses. One unique advantage of this approach over using continuous-wave light is that it allows for optical time-interleaving techniques to multiply the effective repetition rate and input sampling rate. For the task of noisy waveform classification, we can use two different time-multiplexing techniques to generate equivalent ultrafast optical input signals beyond the limited sampling rate of our arbitrary waveform generator. The first technique allows for real-time input generation as shown conceptually in Fig. 6.13. Consider an input laser pulse train with repetition period T. Suppose we wish to increase the effective input sampling rate by a factor of 4. Then, to do this, we use a Mach-Zehnder interferometer with 4 arms. Each arm is delayed by $T/4$ relative to the previous arm and contains an intensity modulator with sampling period of T. Upon recombining at the output, the optical input signal effectively has an input sampling period of $T/4$. This method exploits the fact that the pulse length of the laser pulse τ is much less than the RF sampling period T. The maximum allowable number of arms in the Mach-Zehnder interferometer to upconvert the sampling rate is $\sim T/\tau$ before the output optical pulses begin to undesirably overlap temporally.

In practice, we are limited by the number of available modulators and channels on our arbitrary waveform generator. The scope of this work is to demonstrate ultrafast optical computing, and not to design new ultrafast optical transceivers. Therefore, to reach even higher effective input sampling rates for very high clock rate computing, we use another time-multiplexing technique based on an asynchronously-pumped optical cavity. As opposed to real-time input generation, we call this technique "offline" input generation since the input signals must be prepared ahead-of-time before beginning the AO-RNN computation. The concept for offline input generation is shown in Fig. 6.14. Suppose that we have an input laser pulse train with repetition period T and we can generate RF samples with period T, but desire to have an input optical signal with sample period of T/m where $m > 1$. A synchronously-pumped optical cavity has a roundtrip time NT where N is an integer so that each input laser

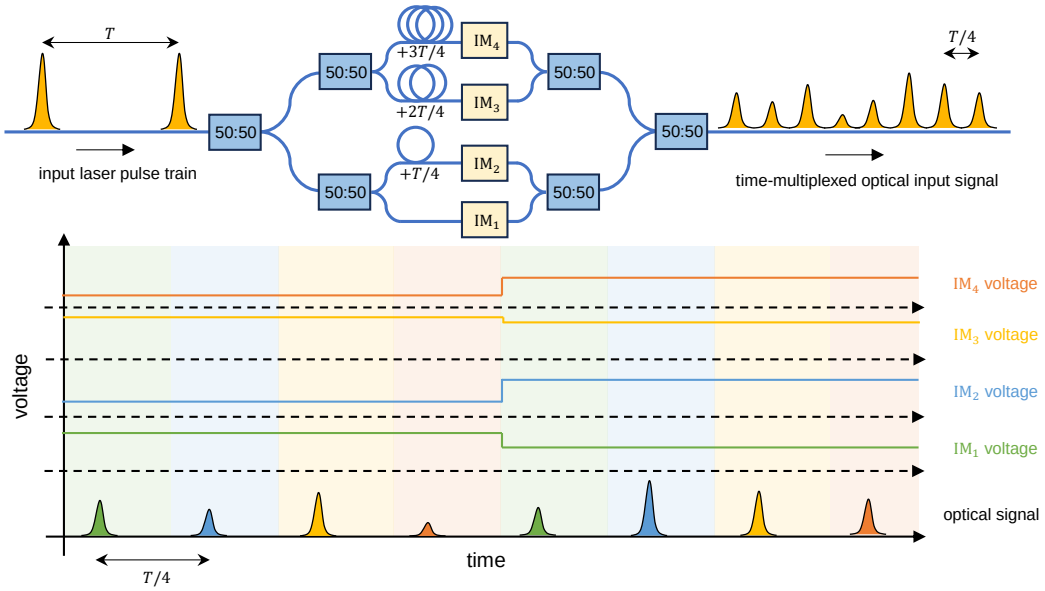

Figure 6.13: **Real-time time-multiplexed optical input generation.** A multi-arm Mach-Zehnder interferometer in which each arm is delayed by an integer fraction of T relative to the other arms and contains an intensity modulator (IM) with sampling period of T can generate an optical input signal with effective sampling period that is an integer fraction of T.

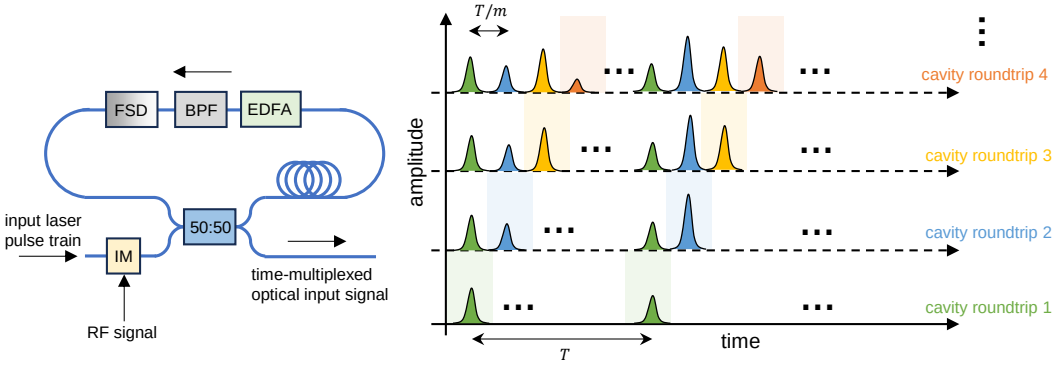

Figure 6.14: **Offline time-multiplexed optical input generation.** The desired optical input signal is gradually built-up over many roundtrips of an asynchronously-pumped optical cavity. IM: intensity modulator, EDFA: erbium doped fibre amplifier, BPF: band-pass filter, FSD: free-space delay.

pulse will overlap temporally with a laser pulse in the cavity. We detune the cavity by adjusting a free-space delay stage such that the roundtrip time is reduced by T/m, so that the rountrip time of the asynchronously-pumped optical cavity is $NT - T/m$. During the first cavity roundtrip we modulate the input laser pulses with every m^{th} waveform sample (i.e., sample 1, $m+1$, $2m+1$, ...). Then, during the next cavity

roundtrip we modulate the input laser pulses with every m^{th} waveform sample offset by 1 sample (i.e., sample 2, $m + 2$, $2m + 2$, ...). The cavity pulses from the first roundtrip will be displaced forwards by T/m relative to the input laser pulses for the second roundtrip. We repeat this procedure until we have completed modulation of all waveform samples. In this way, we gradually build-up the desired optical waveform with sampling period T/m over multiple cavity roundtrips using only a single input modulator and RF channel with sampling period T. The cavity contains an EDFA that is tuned to compensate the roundtrip loss so that many (typically > 20) cavity roundtrips are possible without exponential signal degradation from the cavity out-coupling and propagation loss. Amplitude scaling factors to account for loss variations between roundtrips can be calibrated by sending a single laser pulse into the cavity and measuring its amplitude decay after each roundtrip. Using this offline input generation method, the cavity roundtrip time must be at least $> nT/m$ where n is the total number of waveform samples. The maximum upconversion rate factor is limited by $\tau \approx T/m$ where τ is the input laser pulse length. If the cavity detuning T/m is comparable to the pulse length τ, then pulses will overlap from roundtrip to roundtrip, which will result in undesirable sample cross-talk. The gating modulator controlling inputs into the AO-RNN is synchronized to the offline input generation such that it only allows the final desired cavity roundtrip to be transmitted, which prevents input signal artifacts from the previous roundtrips.

All-optical image generation

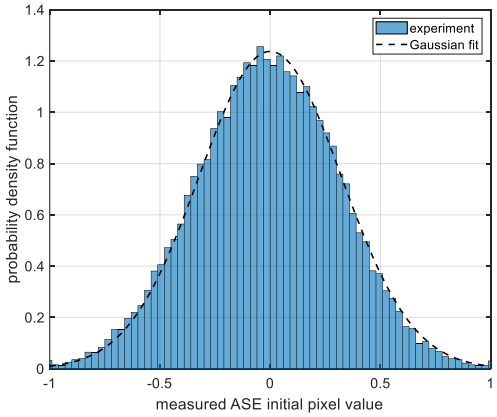

Figure 6.15: **Quantum noise distribution.** The histogram shows the statistics for the initial pixel values sampled from amplified spontaneous emission (ASE) for $\sim$ 35000 sample points. It is well-approximated by a Gaussian distribution (dashed black line) with zero mean and standard deviation of $\sim$ 0.322.

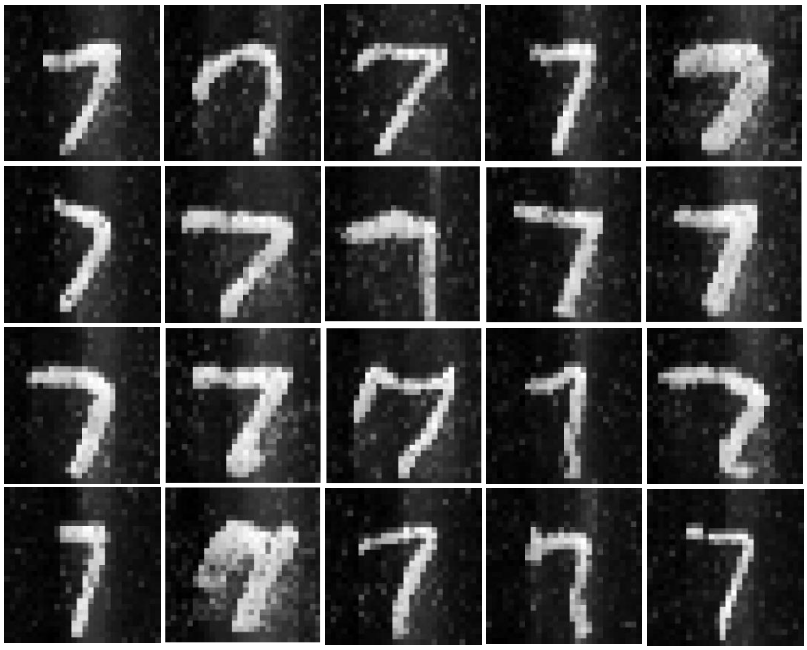

Figure 6.16: **25 generated images of "seven" using the AO-RNN.**

References

[1] John L. Hennessy and David A. Patterson. *Computer architecture: A quantitative approach*. Elsevier, 2011.

[2] Friedrich L. Bauer. *Origins and foundations of computing: In cooperation with Heinz Nixdorf MuseumsForum*. Springer Science & Business Media, 2009.

[3] Arthur Walter Burks. Electronic computing circuits of the eniac. *Proceedings of the IRE*, 35(8):756–767, 1947.

[4] Robert R. Schaller. Moore's law: Past, present and future. *IEEE Spectrum*, 34 (6):52–59, 1997.

[5] Robert H. Dennard, Fritz H. Gaensslen, Hwa-Nien Yu, V. Leo Rideout, Ernest Bassous, and Andre R. LeBlanc. Design of ion-implanted mosfet's with very small physical dimensions. *IEEE Journal of Solid-State Circuits*, 9(5):256–268, 1974.

[6] John Backus. Can programming be liberated from the von neumann style? A functional style and its algebra of programs. *Communications of the ACM*, 21 (8):613–641, 1978.

[7] Peter L. McMahon. The physics of optical computing. *Nature Reviews Physics*, 5(12):717–734, 2023.

[8] R.A. Heinz, J.O. Artman, and S.H. Lee. Matrix multiplication by optical methods. *Applied Optics*, 9(9):2161–2168, 1970.

[9] Xingyuan Xu, Mengxi Tan, Bill Corcoran, Jiayang Wu, Andreas Boes, Thach G. Nguyen, Sai T. Chu, Brent E. Little, Damien G. Hicks, Roberto Morandotti, et al. 11 tops photonic convolutional accelerator for optical neural networks. *Nature*, 589(7840):44–51, 2021.

[10] Johannes Feldmann, Nathan Youngblood, Maxim Karpov, Helge Gehring, Xuan Li, Maik Stappers, Manuel Le Gallo, Xin Fu, Anton Lukashchuk, Arslan S. Raja, et al. Parallel convolutional processing using an integrated photonic tensor core. *Nature*, 589(7840):52–58, 2021.

[11] Gordon H.Y. Li, Ryoto Sekine, Rajveer Nehra, Robert M. Gray, Luis Ledezma, Qiushi Guo, and Alireza Marandi. All-optical ultrafast relu function for energy-efficient nanophotonic deep learning. *Nanophotonics*, 12(5):847–855, 2023.

[12] Qiushi Guo, Ryoto Sekine, Luis Ledezma, Rajveer Nehra, Devin J. Dean, Arkadev Roy, Robert M. Gray, Saman Jahani, and Alireza Marandi. Femtojoule femtosecond all-optical switching in lithium niobate nanophotonics. *Nature Photonics*, 16(9):625–631, 2022.

[13] George Mourgias-Alexandris, Apostolos Tsakyridis, Nikolaos Passalis, Anastasios Tefas, Konstantinos Vyrsokinos, and Nikolaos Pleros. An all-optical neuron with sigmoid activation function. *Optics Express*, 27(7):9620–9630, 2019.

[14] Yichen Shen, Nicholas C. Harris, Scott Skirlo, Mihika Prabhu, Tom Baehr-Jones, Michael Hochberg, Xin Sun, Shijie Zhao, Hugo Larochelle, Dirk Englund, et al. Deep learning with coherent nanophotonic circuits. *Nature Photonics*, 11(7):441–446, 2017.

[15] Farshid Ashtiani, Alexander J. Geers, and Firooz Aflatouni. An on-chip photonic deep neural network for image classification. *Nature*, 606(7914):501–506, 2022.

[16] Saumil Bandyopadhyay, Alexander Sludds, Stefan Krastanov, Ryan Hamerly, Nicholas Harris, Darius Bunandar, Matthew Streshinsky, Michael Hochberg, and Dirk Englund. Single-chip photonic deep neural network with forward-only training. *Nature Photonics*, pages 1–9, 2024.

[17] Johannes Feldmann, Nathan Youngblood, C. David Wright, Harish Bhaskaran, and Wolfram H.P. Pernice. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature*, 569(7755):208–214, 2019.

[18] Alexander N. Tait, Thomas Ferreira De Lima, Ellen Zhou, Allie X. Wu, Mitchell A. Nahmias, Bhavin J. Shastri, and Paul R. Prucnal. Neuromorphic photonic networks using silicon photonic weight banks. *Scientific Reports*, 7 (1):7430, 2017.

[19] Paul R. Prucnal and Bhavin J. Shastri. *Neuromorphic photonics*. CRC Press, 2017.

[20] Alireza Marandi, Zhe Wang, Kenta Takata, Robert L Byer, and Yoshihisa Yamamoto. Network of time-multiplexed optical parametric oscillators as a coherent ising machine. *Nature Photonics*, 8(12):937–942, 2014.

[21] Takahiro Inagaki, Yoshitaka Haribara, Koji Igarashi, Tomohiro Sonobe, Shuhei Tamate, Toshimori Honjo, Alireza Marandi, Peter L. McMahon, Takeshi Umeki, Koji Enbutsu, et al. A coherent ising machine for 2000-node optimization problems. *Science*, 354(6312):603–606, 2016.

[22] Toshimori Honjo, Tomohiro Sonobe, Kensuke Inaba, Takahiro Inagaki, Takuya Ikuta, Yasuhiro Yamada, Takushi Kazama, Koji Enbutsu, Takeshi Umeki, Ryoichi Kasahara, et al. 100,000-spin coherent ising machine. *Science Advances*, 7(40):eabh0952, 2021.

[23] Midya Parto, William Hayenga, Alireza Marandi, Demetrios N. Christodoulides, and Mercedeh Khajavikhan. Realizing spin hamiltonians in nanoscale active photonic lattices. *Nature Materials*, 19(7):725–731, 2020.

[24] Xing Lin, Yair Rivenson, Nezih T. Yardimci, Muhammed Veli, Yi Luo, Mona Jarrahi, and Aydogan Ozcan. All-optical machine learning using diffractive deep neural networks. *Science*, 361(6406):1004–1008, 2018.

[25] François Duport, Bendix Schneider, Anteo Smerieri, Marc Haelterman, and Serge Massar. All-optical reservoir computing. *Optics Express*, 20(20):22783–22795, 2012.

[26] Antoine Dejonckheere, François Duport, Anteo Smerieri, Li Fang, Jean-Louis Oudar, Marc Haelterman, and Serge Massar. All-optical reservoir computer based on saturation of absorption. *Optics Express*, 22(9):10868–10881, 2014.

[27] Ying Zuo, Bohan Li, Yujun Zhao, Yue Jiang, You-Chiuan Chen, Peng Chen, Gyu-Boong Jo, Junwei Liu, and Shengwang Du. All-optical neural network with nonlinear activation functions. *Optica*, 6(9):1132–1137, 2019.

[28] Larry Medsker and Lakhmi C. Jain. *Recurrent neural networks: Design and applications*. CRC Press, 1999.

[29] Hava T. Siegelmann and Eduardo D. Sontag. On the computational power of neural nets. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 440–449, 1992.

[30] Christian Leefmans, Avik Dutt, James Williams, Luqi Yuan, Midya Parto, Franco Nori, Shanhui Fan, and Alireza Marandi. Topological dissipation in a time-multiplexed photonic resonator network. *Nature Physics*, 18(4):442–449, 2022.

[31] Yunping Bai, Xingyuan Xu, Mengxi Tan, Yang Sun, Yang Li, Jiayang Wu, Roberto Morandotti, Arnan Mitchell, Kun Xu, and David J. Moss. Photonic multiplexing techniques for neuromorphic computing. *Nanophotonics*, 12(5): 795–817, 2023.

[32] Tara Fortier and Esther Baumann. 20 years of developments in optical frequency comb technology and applications. *Communications Physics*, 2(1): 153, 2019.

[33] Carsten Langrock and Martin M. Fejer. Fiber-feedback continuous-wave and synchronously-pumped singly-resonant ring optical parametric oscillators using reverse-proton-exchanged periodically-poled lithium niobate waveguides. *Optics Letters*, 32(15):2263–2265, 2007.

[34] Gordon H.Y. Li, Christian R. Leefmans, James Williams, Robert M. Gray, Midya Parto, and Alireza Marandi. Deep learning with photonic neural cellular automata. *Light: Science & Applications*, 13(1):283, 2024.

[35] Zhiquan Yuan, Maodong Gao, Yan Yu, Heming Wang, Warren Jin, Qing-Xin Ji, Avi Feshali, Mario Paniccia, John Bowers, and Kerry Vahala. Soliton pulse pairs at multiple colours in normal dispersion microresonators. *Nature Photonics*, 17(11):977–983, 2023.

[36] Tobias J. Kippenberg, Alexander L. Gaeta, Michal Lipson, and Michael L. Gorodetsky. Dissipative kerr solitons in optical microresonators. *Science*, 361 (6402):eaan8083, 2018.

[37] Georg Herink, Felix Kurtz, Bahram Jalali, Daniel R. Solli, and Claus Ropers. Real-time spectral interferometry probes the internal dynamics of femtosecond soliton molecules. *Science*, 356(6333):50–54, 2017.

[38] Mark A. Foster, Reza Salem, David F. Geraghty, Amy C. Turner-Foster, Michal Lipson, and Alexander L. Gaeta. Silicon-chip-based ultrafast optical oscilloscope. *Nature*, 456(7218):81–84, 2008.

[39] Keisuke Goda and Bahram Jalali. Dispersive fourier transformation for fast continuous single-shot measurements. *Nature Photonics*, 7(2):102–112, 2013.

[40] Daniel J. Kane and Rick Trebino. Single-shot measurement of the intensity and phase of an arbitrary ultrashort pulse by using frequency-resolved optical gating. *Optics Letters*, 18(10):823–825, 1993.

[41] Xu Yi, Qi-Fan Yang, Ki Youl Yang, and Kerry J. Vahala. Imaging soliton dynamics in optical microcavities. *Nature Communications*, 9(1):3565, 2018.

[42] Robert M. Gray, Mingchen Liu, Selina Zhou, Arkadev Roy, Luis Ledezma, and Alireza Marandi. Quadratic-soliton-enhanced mid-ir molecular sensing. *Nature Communications*, 15(1):9086, 2024.

[43] Peter Gomber and Martin Haferkorn. High-frequency-trading: High-frequency-trading technologies and their implications for electronic securities trading. *Business & Information Systems Engineering*, 5:97–99, 2013.

[44] Vladimir V. Gligorov and Mike Williams. Efficient, reliable and fast high-level triggering using a bonsai boosted decision tree. *Journal of Instrumentation*, 8 (02):P02013, 2013.

[45] Alan E. Willner, Salman Khaleghi, Mohammad Reza Chitgarha, and Omer Faruk Yilmaz. All-optical signal processing. *Journal of Lightwave Technology*, 32(4):660–680, 2013.

[46] Seou Choi, Yannick Salamin, Charles Roques-Carmes, Rumen Dangovski, Di Luo, Zhuo Chen, Michael Horodynski, Jamison Sloan, Shiekh Zia Uddin, and Marin Soljačić. Photonic probabilistic machine learning using quantum vacuum noise. *Nature Communications*, 15(1):7760, 2024.

[47] Charles Roques-Carmes, Yannick Salamin, Jamison Sloan, Seou Choi, Gustavo Velez, Ethan Koskas, Nicholas Rivera, Steven E Kooi, John D Joannopoulos, and Marin Soljačić. Biasing the quantum vacuum to control macroscopic probability distributions. *Science*, 381(6654):205–209, 2023.

[48] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[49] Yaron Lipman, Ricky T.Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[50] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[51] Peter W. Milonni and Joseph H. Eberly. *Laser physics*. John Wiley & Sons, 2010.

[52] Cheng Wang, Mian Zhang, Xi Chen, Maxime Bertrand, Amirhassan Shams-Ansari, Sethumadhavan Chandrasekhar, Peter Winzer, and Marko Lončar. Integrated lithium niobate electro-optic modulators operating at cmos-compatible voltages. *Nature*, 562(7725):101–104, 2018.

[53] Robert M. Gray, Thomas Zacharias, Rahul Chawlani, Luis Ledezma, Ryoto Sekine, James A. Williams, and Alireza Marandi. Soliton pulse compression in lithium niobate nanophotonics. In *2024 Conference on Lasers and Electro-Optics (CLEO)*, pages 1–2. IEEE, 2024.

[54] Arkadev Roy, Luis Ledezma, Luis Costa, Robert Gray, Ryoto Sekine, Qiushi Guo, Mingchen Liu, Ryan M Briggs, and Alireza Marandi. Visible-to-mid-ir tunable frequency comb in nanophotonics. *Nature Communications*, 14(1): 6549, 2023.

[55] Luis Ledezma, Arkadev Roy, Luis Costa, Ryoto Sekine, Robert Gray, Qiushi Guo, Rajveer Nehra, Ryan M. Briggs, and Alireza Marandi. Octave-spanning tunable infrared parametric oscillators in nanophotonics. *Science Advances*, 9 (30):eadf9711, 2023.

[56] Robert M. Gray, Ryoto Sekine, Luis Ledezma, Gordon H.Y. Li, Selina Zhou, Arkadev Roy, Midya Parto, and Alireza Marandi. Large-scale time-multiplexed nanophotonic parametric oscillators. *arXiv preprint arXiv:2405.17355*, 2024.

[57] Tiankuang Zhou, Wei Wu, Jinzhi Zhang, Shaoliang Yu, and Lu Fang. Ultrafast dynamic machine vision with spatiotemporal photonic computing. *Science Advances*, 9(23):eadg4391, 2023.

[58] Peng Wang, Jinyang Liang, and Lihong V. Wang. Single-shot ultrafast imaging attaining 70 trillion frames per second. *Nature Communications*, 11(1):2091, 2020.

[59] Margherita Maiuri, Marco Garavelli, and Giulio Cerullo. Ultrafast spectroscopy: State of the art and open challenges. *Journal of the American Chemical Society*, 142(1):3–15, 2019.

[60] Kazuro Kikuchi. Fundamentals of coherent optical fiber communications. *Journal of Lightwave Technology*, 34(1):157–179, 2015.

[61] Myoung-Gyun Suh and Kerry J. Vahala. Soliton microcomb range measurement. *Science*, 359(6378):884–887, 2018.

[62] William C. Swann and Nathan R. Newbury. Frequency-resolved coherent lidar using a femtosecond fiber laser. *Optics Letters*, 31(6):826–828, 2006.

[63] Eric D. Black. An introduction to pound–drever–hall laser frequency stabilization. *American Journal of Physics*, 69(1):79–87, 2001.

[64] Xiangpeng Liang, Yanan Zhong, Jianshi Tang, Zhengwu Liu, Peng Yao, Keyang Sun, Qingtian Zhang, Bin Gao, Hadi Heidari, He Qian, et al. Rotating neurons for all-analog implementation of cyclic reservoir computing. *Nature Communications*, 13(1):1549, 2022.

[65] Standard Performance Evaluation Corporation. Spec benchmarks and tools. `https://www.spec.org/benchmarks.html`, 2024. Accessed: 2024-12-31.

[66] PassMark Software. Passmark - cpu benchmarks. `https://www.cpubenchmark.net/CPU_mega_page.html`, 2024. Accessed: 2024-12-31.

[67] HWBOT. elmor‘s cpu frequency score 9117.75 mhz with core i9 14900ks (8p). https://hwbot.org/submission/5508265_elmor_cpu_frequency_core_i9_14900ks_(8p)_9117.75_mhz, 2024. Accessed: 2024-12-31.

[68] Warren Jin, Qi-Fan Yang, Lin Chang, Boqiang Shen, Heming Wang, Mark A Leal, Lue Wu, Maodong Gao, Avi Feshali, Mario Paniccia, et al. Hertz-linewidth semiconductor lasers using cmos-ready ultra-high-q microresonators. *Nature Photonics*, 15(5):346–353, 2021.

[69] Xu Yi, Qi-Fan Yang, Ki Youl Yang, Myoung-Gyun Suh, and Kerry Vahala. Soliton frequency comb at microwave rates in a high-q silica microresonator. *Optica*, 2(12):1078–1085, 2015.

[70] Qi-Fan Yang, Xu Yi, Ki Youl Yang, and Kerry Vahala. Spatial-mode-interaction-induced dispersive waves and their active tuning in microresonators. *Optica*, 3(10):1132–1135, 2016.

Chapter 7

TURING-COMPLETENESS AND UNDECIDABILITY IN COUPLED NONLINEAR OPTICAL RESONATORS

Gordon H.Y. Li and Alireza Marandi. Turing-completeness and undecidability in coupled nonlinear optical resonators. *arXiv preprint arXiv:2501.06966*, 2025. doi:10.48550/arXiv.2501.06966.
G.H.Y.L. conceived the project, developed the theory, and wrote the manuscript.

7.1 Abstract

Networks of coupled nonlinear optical resonators have emerged as an important class of systems in ultrafast optical science, enabling richer and more complex nonlinear dynamics compared to their single-resonator or travelling-wave counterparts. In recent years, these coupled nonlinear optical resonators have been applied as application-specific hardware accelerators for computing applications including combinatorial optimization and artificial intelligence. In this work, we rigorously prove a fundamental result showing that coupled nonlinear optical resonators are Turing-complete computers, which endows them with much greater computational power than previously thought. Furthermore, we show that the minimum threshold of hardware complexity needed for Turing-completeness is surprisingly low, which has profound physical consequences. In particular, we show that several problems of interest in the study of coupled nonlinear optical resonators are formally undecidable. These theoretical findings can serve as the foundation for better understanding the promise of next-generation, ultrafast all-optical computers.

7.2 Introduction

Networks of coupled nonlinear optical resonators (CNORs) are a general class of optical systems combining nonlinear optical interactions, resonator or cavity feedback mechanisms, and linear couplings between resonators. CNORs have been well-studied in a variety of experimental platforms including pairs or dimers of coupled cavities [1–3], time-multiplexed cavities with time-delayed couplings [4], spatially-multiplexed arrays of cavities [5–7], and wavelength-multiplexed cavities with a synthetic frequency dimension [8]. They offer richer dynamics compared to their linear, traveling-wave, or single-cavity counterparts and have served as a fertile

ground for exploring complex nonlinear dynamics such as solitons [9–11], phase transitions [12], topological phenomena [13–15], and non-Hermitian physics [16]. Furthermore, CNORs have technologically-important applications such as low-noise microwave generation [17, 18], optical switching [19, 20], optical isolators [21, 22], and quantum state generation [23–25].

Here we are interested in better understanding the computational properties of CNORs. At an abstract level, CNORs can possess both strong nonlinearity and memory, which are the essential ingredients for computation. Indeed, CNORs have been demonstrated as physical Ising solvers for combinatorial optimization [4, 26, 27], as well as efficient hardware accelerators for deep learning [28] and neuromorphic computing [29, 30]. However, previous experimental works treated CNORs as special-purpose or application-specific optical processors performing a single kind of computational task. In this work, we rigorously prove that CNORs are Turing-complete (see Theorem 1), which means that they can be used to compute any computable function, and thus possess much greater computational power than previously considered. This result is not obvious since CNORs are described by continuous-valued or analog field amplitudes instead of discrete-valued states, and have a completely dynamical memory instead of the static memory elements typically associated with Turing Machines.

Turing-completeness also immediately implies that several properties of CNORs such as the existence of a steady-state or periodic oscillation are formally undecidable, i.e., there is no algorithm that can always correctly answer the associated decision problem (either in the affirmative or negative). This stems directly from the undecidability of the *Entscheidungsproblem* (or Halting Problem) [31], which we now interpret in the context of CNORs. Therefore, we may add CNORs to the rapidly growing list of physical theories that exhibit undecidable properties, e.g., classical mechanical systems [32–34], fluid dynamics [35, 36], quantum many-body systems [37–39], quantum field theory [40, 41], and quantum gravity [42].

The general strategy for our mathematical proof is to show by explicit construction how any Turing Machine can be exactly simulated by an associated CNOR. This explicit construction also allows us to conveniently place an upper bound on the minimum hardware complexity needed for Turing-completeness in CNORs. If the minimum complexity needed for Turing-complete CNORs is high, then we could just simply agree to avoid such niche situations and hence also avoid vexing issues of undecidability. On the other hand, if the minimum complexity is low, then we

should expect the study of CNORs (and ultrafast optical science more broadly) to be rife with examples of undecidable problems. Interestingly, we show that the latter is true and that the minimum complexity needed for Turing-complete CNORs is well-within current experimental capabilities. Therefore, we cannot easily dismiss issues of undecidability and must seriously confront what can and cannot be logically reasoned regarding CNORs.

7.3 Results

Computational models

We begin by defining our computational models, then give a detailed proof of the main Turing-completeness result, and point out some interesting physical consequences for the study of CNORs. Finally, we discuss the limitations and practicalities associated with our results.

We follow the definition of a (one-tape) Turing machine given by Minsky [43] as shown conceptually in Fig. 7.1a. Other equivalent definitions of Turing machines and their generalizations are also possible.

Definition 1 (Turing Machine) *A Turing Machine (TM) is a finite-state machine with access to a linear tape that extends infinitely in both left and right directions. The tape consists of a sequence of cells each printed with a single symbol from a finite tape alphabet* $S = \{s_1, \ldots, s_n\}$. *A movable head is situated on some cell and can read/write symbols on that cell. The current symbol read by the head is the input symbol to the finite-state machine with internal states* $Q = \{q_1, \ldots, q_m\}$. *The TM is described by three (partial) functions* $G : Q \times S \nrightarrow Q$, $F : Q \times S \nrightarrow S$, *and* $D : Q \times S \nrightarrow \{0, 1\}$, *denoting the updated state, new symbol to write, and direction to move the head, respectively. We take '0' to mean 'move left' and '1' to mean 'move right'. At each time step, the machine starts in some state* q_i, *reads the symbol* s_j *written on the cell under the head, prints there the new symbol* $F(q_i, s_j)$, *moves one cell to the left or right according to* $D(q_i, s_j)$, *and then enters the new state* $G(q_i, s_j)$.

□

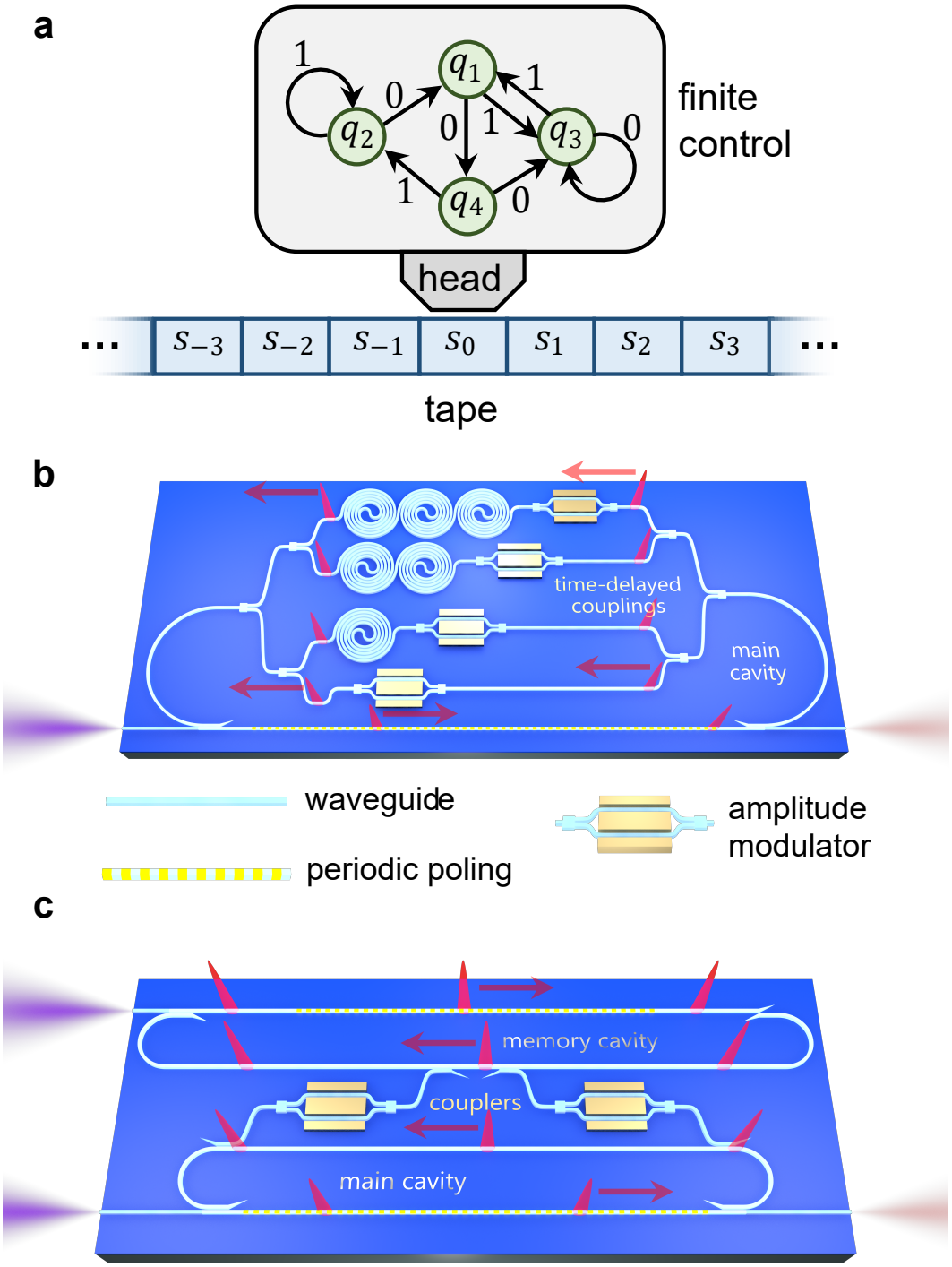

Figure 7.1: **Computational models.** (a) A Turing Machine (TM) consists of a bi-infinite tape of symbols, a head that can read/write symbols and move left/right, and a finite state control. A degenerate optical parametric oscillator network (DOPON) can be implemented using time-multiplexing in which short laser pulses act as independent nonlinear resonators that interact via either (b) intra-cavity time-delayed couplings or (c) coupled cavities.

Without loss of generality, henceforth we consider just a binary tape alphabet $S = \{0, 1\}$ where '0' is the special *blank* tape symbol. Importantly, the starting configuration of the tape can only contain a finite number of non-blank symbols. In the following, we consider a type of CNOR based on networks of time-multiplexed degenerate optical parametric oscillators with dissipitive couplings. These are the most well-studied and relevant class of CNORs for computing applications and have a wide range of possible experimental platforms including free-space optical systems [4], optical fiber networks [27], and photonic integrated circuits [44].

Definition 2 (Degenerate Optical Parametric Oscillator Network) *A Degenerate Optical Parametric Oscillator Network (DOPON) is described by N optical pulses with amplitudes $x_i \in \mathbb{R}$ for $i = 1, 2, \ldots, N$ that evolve in time according to the dynamical system in Eq. 7.1*

$$x_i(t+1) = \rho\left[x_i(t)\right] + \sum_{j=1}^{N} J_{ij}(t) \cdot x_j(t) \tag{7.1}$$

where t is the discrete time step, $J_{ij}(t)$ is the coupling weight between the i^{th} and j^{th} pulses at time step t, and $\rho : \mathbb{R} \to \mathbb{R}$ is an anti-symmetric (odd) function $\rho(-x) = -\rho(x)$ representing saturable parametric gain (i.e., linear for small signal values, and saturated for large signal values) in each optical resonator as shown in Eq. 7.2

$$\rho(x) = \begin{cases} 1\,, & \text{if } x \geq 1 \\ g(x)\,, & \text{if } 3/4 < x < 1 \\ x\,, & \text{if } 0 \leq x \leq 3/4 \end{cases} \tag{7.2}$$

where $g(x)$ is any continuous monotonically-increasing function defined over the domain $x \in [3/4, 1]$.

□

We consider two possible designs for DOPONs, which can both be operated in a way that obeys Eq. 7.1. The first implementation, shown schematically in Fig. 7.1b is based on a single main cavity containing multiple short laser pulses. Each pulse occupies one of equally-spaced time bins that act as independent optical resonators experiencing the nonlinear optical parametric gain mechanism (e.g., provided by periodically-poled lithium niobate waveguides [45]). The pulses can be coupled

using intra-cavity optical delay lines with delays matching the temporal separation of pulses, shown as a multi-arm Mach-Zehnder interferometer in which coupling weights are set using amplitude modulators (e.g., electro-optic modulators [46]). In this configuration, all-to-all coupling can be achieved for a main cavity containing N pulses if there are N optical delay lines in the intra-cavity interferometer. Therefore, the discrete time step t represents the main cavity roundtrip number, as all pulses can be coupled after a single roundtrip.

The second implementation, shown schematically in Fig. 7.1c is based on a main cavity containing N equally-spaced pulses that is coupled to a second cavity containing $N + 1$ pulses (with the same pulse spacing as the main cavity), which we call the memory cavity. The programmable couplers between the main and memory cavities allow the coupling between pulses to occur according to Eq. 7.1. To do this, the memory cavity is operated with a purely linear gain that compensates the roundtrip loss and the main cavity has a saturable gain as before. Coupling terms are accumulated one at a time with intermediate values stored in the memory cavity. Memory cavity pulses are only allowed to interfere with main cavity pulses after all coupling terms are accumulated. Therefore, in this configuration with all-to-all coupling, the discrete-time step t represents $N + 1$ roundtrips of the main cavity. Compared to the first implementation, the second implementation achieves a constant $O(1)$ number of modulators and delay lines at the expense of more cavity roundtrips $O(N)$ to achieve all-to-all coupling.

Some remarks about Definition 2: (1) Previous studies [26] of DOPONs considered continuous-time dynamical systems. However, our use of discrete time in terms of the resonator roundtrip number is more realistic since the gain and couplings are experimentally implemented as lumped elements at fixed locations, and are not continuously distributed throughout the resonator. (2) A single degenerate optical parametric oscillator without couplings is described by $x(t + 1) = \rho[x(t)]$ and has the normal form of a supercritical pitchfork bifurcation $x(t + 1) = \alpha \cdot x(t) - x^3$. For $\alpha < 1$, we say that the oscillator is *below threshold* and the only stable fixed-point is the trivial solution $x = 0$. For $\alpha > 1$, the oscillator is *above threshold* and has two stable fixed-point solutions $x = \pm\sqrt{\alpha - 1}$. Previous studies of DOPONs considered only the normal form of the nonlinearity, however the inclusion of a saturable gain is more physical and ensures that the pulse amplitudes remain bounded. (3) Strictly speaking, there should be additional Langevin noise operators in Eq. 7.1 when operating DOPONs below threshold, however for simplicity, we ignore small noise

perturbations since we consider only above threshold DOPONs. Therefore, the trajectory of the DOPON is deterministic and completely specified by the coupling weights $\{J_{ij}(t)\}$ and initial conditions $x_i(0)$ for $i = 1, 2, \ldots, N$. We require that the couplings $\{J_{ij}(t)\}$ be known *a priori*, i.e., they cannot depend on the result of any intermediate step of the time-evolution.

Turing-completeness proof

Theorem 1 *For every Turing Machine $\mathcal{T}$, there exists a Degenerate Optical Parametric Oscillator Network $\mathcal{N}_\mathcal{T}$ with $N = 12$ optical pulses that simulates $\mathcal{T}$.*

Proof: We will show by explicit construction how a DOPON $\mathcal{N}_\mathcal{T}$ with just $N = 12$ optical pulses can be used to simulate any TM. We denote these pulse amplitudes in order as $\{q, r, l, \mathbb{1}, a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8\}$ and begin by describing how the TM is encoded in $\mathcal{N}_\mathcal{T}$. Suppose that $\mathcal{T}$ has states $Q = \{q_1, \ldots, q_m\}$, where we index the states from 1 to m. If $\mathcal{T}$ is in state $q_i \in Q$ after step T of the TM operation, then we define $\mathcal{U} : Q \to \mathbb{R}$ and require

$$q(8mT) = \mathcal{U}(q_i) \coloneqq \sum_{k=1}^{i} \frac{(-1)^{k+1}}{2^k} \,. \tag{7.3}$$

In other words, the TM state is represented using a unary-like encoding scheme $\mathcal{U}$ for the optical pulse amplitude q in $\mathcal{N}_\mathcal{T}$. The idea is to choose the dynamics of $\mathcal{N}_\mathcal{T}$ such that the states are decoded one at a time, checking one state every 8 time steps and performing the appropriate state update.

Next, we describe how the TM tape is encoded in $\mathcal{N}_\mathcal{T}$. Consider the tape after step T as a sequence of binary symbols $\ldots s_{-2}s_{-1}\boxed{s_0}s_1s_2\ldots$ where the boxed symbol $\boxed{s_0}$ denotes the current symbol being read by the head. We represent the tape in $\mathcal{N}_\mathcal{T}$ using two optical pulse amplitudes r and l, which encode the right/left sequences $\boxed{s_0}s_1s_2\ldots$ and $s_{-1}s_{-2}\ldots$, respectively, and define $C^\pm : \mathbb{N} \to \mathbb{R}$ such that

$$r(8mT) = C^+(s_0s_1s_2\ldots) \coloneqq \sum_{k=0}^{\infty} (-1)^k \frac{2s_k + 1}{4^{k+1}} \tag{7.4a}$$

$$l(8mT) = C^-(s_{-1}s_{-2}\ldots) \coloneqq \sum_{k=1}^{\infty} (-1)^{k+1} \frac{2s_{-k} + 1}{4^k} \,. \tag{7.4b}$$

This particular Cantor-like encoding scheme $C^\pm$ in base 4 (similar to in Ref. [47]) is chosen instead of a simpler unary or binary encoding for two main reasons.

Turing Machine $\mathcal{T}$	Degenerate Optical Parametric Oscillator Network $\mathcal{N}_{\mathcal{T}}$
State $q_i \in Q$	Optical pulse amplitude with unary-like encoding $q = \mathcal{U}(q_i) = \sum_{k=1}^{i} \frac{(-1)^{k+1}}{2^k}$
Tape $\ldots s_{-2}s_{-1}\boxed{s_0}s_1s_2\ldots$	Optical pulse amplitudes with Cantor-like encoding $r = \mathcal{C}^{+}(s_0s_1s_2\ldots) = \sum_{k=0}^{\infty}(-1)^k \frac{2s_k+1}{4^{k+1}}$ $l = \mathcal{C}^{-}(s_{-1}s_{-2}\ldots) = \sum_{k=1}^{\infty}(-1)^{k+1} \frac{2s_{-k}+1}{4^k}$
Time Steps $T : \mathbb{N} \to \mathbb{N}$	Optical resonator roundtrips $t = 8mT$

Table 7.1: Encoding of Turing Machine $\mathcal{T}$ in Degenerate Optical Parametric Oscillator Network $\mathcal{N}_{\mathcal{T}}$.

Firstly, it ensures that the necessary amplitudes are bounded despite a potentially unbounded TM tape length, which avoids the unphysical scenario of an optical pulse with unbounded amplitude and energy. Secondly, consider the two numbers $0.1000\ldots$ and $0.0111\ldots$ using a binary encoding. This runs into the difficulty that distinguishing the two numbers requires traversing the entire sequence of bits, which may be potentially unbounded if representing a TM tape. To overcome this difficulty, the Cantor-like encoding $\mathcal{C}^{\pm}$ introduces 'gaps' into the unit interval, as shown in Fig. 7.2, such that the most-significant bit can be determined in $O(1)$ time. For example, using the nonlinearity available in $\mathcal{N}_{\mathcal{T}}$ as a decision threshold, we have that $\rho(8r-3) = 1$ if $s_0 = 1$, and $\rho(8r-3) = -1$ if $s_0 = 0$.

The pulse amplitude $\mathbb{1}$ is used as a constant bias or reference amplitude, and does not change from resonator roundtrip to roundtrip. Finally, the remaining pulses $a_1, a_2, \ldots, a_8$ are used as ancillary variables to help store intermediate results of the computation. The encoding of $\mathcal{T}$ in $\mathcal{N}_{\mathcal{T}}$ is summarized in Table 7.1.

Now, we give the full construction for the simulation of $\mathcal{T}$ by $\mathcal{N}_{\mathcal{T}}$ according to the encoding scheme defined in Table 7.1. The objective is to find the set of coupling weights $\{J_{ij}(t)\}$ for $i, j = 1, 2, \ldots, 12$ and $t \in \mathbb{N}$, and the initial optical pulse amplitudes $\{q(0), r(0), l(0), \mathbb{1}(0), a_1(0), a_2(0), \ldots, a_8(0)\}$ that ensure the time-evolution of $\mathcal{N}_{\mathcal{T}}$ exactly reproduces the operation of $\mathcal{T}$. The initial conditions are easy, we simply let $q(0)$, $r(0)$, and $l(0)$ be as defined in Table 7.1 for the initial configuration of $\mathcal{T}$. The bias pulse satisfies $\mathbb{1}(t) = 1$ for all $t \in \mathbb{N}$, and let $a_j(0) = 0$

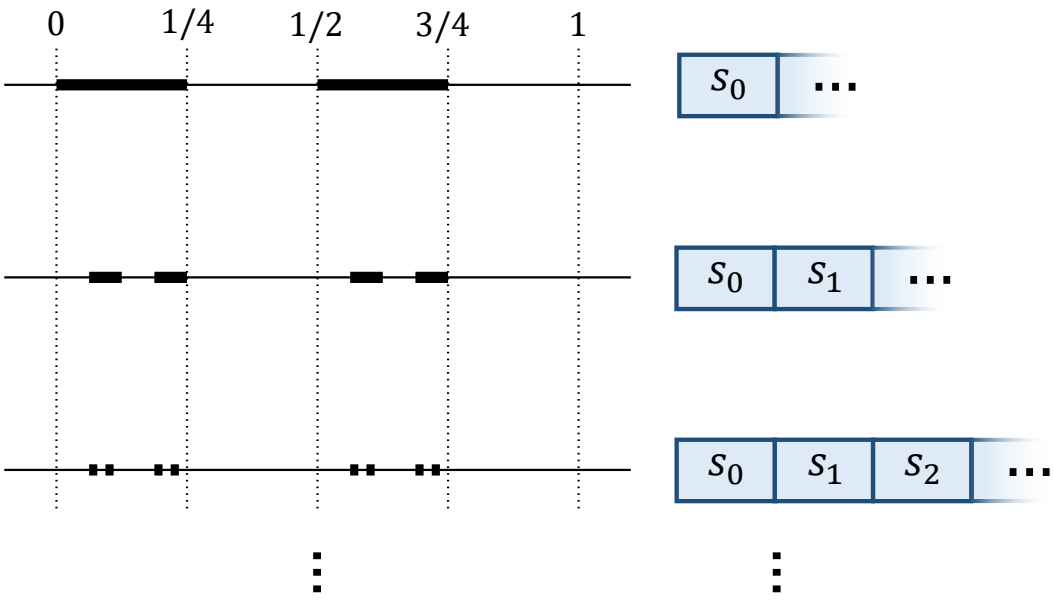

Figure 7.2: **Cantor-like tape encoding.** The Cantor-like encoding in base 4 introduces gaps into the unit interval such that the most-significant bit of the tape can be read in constant $\mathcal{O}(1)$ time without traversing the entire tape sequence.

for $j = 1, 2, \ldots, 8$. The ancillary variables begin and end with zero amplitude during each step of $\mathcal{T}$. For the coupling weights $J_{ij}(t)$, the general strategy is to find a set of time-periodic weights that encode the behaviour of $\mathcal{T}$. During each step of $\mathcal{T}$, the current state is decoded in $\mathcal{N}_{\mathcal{T}}$ by iterating through all states, one state at a time. We can perform this process using 8 time steps per state, hence requiring $8m$ time steps per step of $\mathcal{T}$. Therefore, the coupling weights $J_{ij}(t)$ repeat with a period of $8m$. During each 8-cycle in $\mathcal{N}_{\mathcal{T}}$, the current symbol s_0 is also read, and the appropriate updates (if any) to q, r, and l are performed. We show the construction for one 8-cycle, and the rest follows analogously.

The first step is to check the current state. During the first 8-cycle, we check if $q = \mathcal{U}(q_1)$, otherwise move on to the next state. This is done by reading the most-significant bit of q, then storing the result in a_3 through the steps:

$$q(1) = \rho(q) - 3q + \mathbb{1} \tag{7.5a}$$

$$a_3(2) = \rho(a_3) - 4q \tag{7.5b}$$

$$a_3(3) = \rho(a_3) + \mathbb{1} \tag{7.5c}$$

where we henceforth suppress the time step t for optical pulse amplitudes on the right-hand side of equations since it is always one less than on the left-hand side.

We use the fact that $\rho(q) = q$ since $1/4 \leq q \leq 1/2$ so that Eq. 7.5a performs the action $q \to -2q + 1$, which has the effect of subtracting the most-significant bit of q and shifting all remaining bits to the left. After this action, a_3 performs a decision threshold such that $a_3(3) = 1$ if $q = 0$ and $a_3(3) = 0$ otherwise. Therefore, we will only have $a_3(3) = 1$ if $q(0) = \mathcal{U}(q_1)$, i.e., q encoded for the state q_1 being checked during the first 8-cycle. Iterating through all states by repeating this procedure m times guarantees that $a_3(3) = 1$ during exactly one 8-cycle, which allows the current state to be effectively decoded. In parallel, we also decode the most-significant bits of r and l, which represent the current symbol s_0 read by the head and the left-adjacent symbol s_{-1}, respectively:

$$a_1(1) = \rho(a_1) + 8r - 3\mathbb{1} \tag{7.6a}$$

$$a_1(2) = \rho(a_1) \tag{7.6b}$$

$$a_1(3) = \rho(a_1) - \frac{1}{2}a_1 + \frac{1}{2}\mathbb{1} \tag{7.6c}$$

$$a_2(1) = \rho(a_2) + 8l - 3\mathbb{1} \tag{7.6d}$$

$$a_2(2) = \rho(a_2) \tag{7.6e}$$

$$a_2(3) = \rho(a_2) - \frac{1}{2}a_2 + \frac{1}{2}\mathbb{1}\ . \tag{7.6f}$$

Therefore, we have that $a_1(3) = s_0$ and similarly $a_2(3) = s_{-1}$. Next, we consider the product $a_3(3)a_1(3)$, which is needed to represent the domain $Q \times S$ over which the TM (partial) functions G, F, and D are defined. We have that $a_3(3)a_1(3) = 1$ if $q(0) = \mathcal{U}(q_1)$ *and* $s_0 = 1$. Similarly, $a_3(3)(1 - a_1(3)) = 1$ if $q(0) = \mathcal{U}(q_1)$ *and* $s_0 = 0$. To construct the product pairs using only linear couplings, we make use of the identity $a \cdot b = \rho(a + b - 2) + 1$ for $a, b \in \{0, 1\}$. We store these product pairs in a_5 and a_6, respectively. We also store a copy of $a_3(3)$ in a_7 for later use:

$$a_5(4) = \rho(a_5) + a_1 + a_3 - 2\mathbb{1} \tag{7.7a}$$

$$a_5(5) = \rho(a_5) + \mathbb{1} \tag{7.7b}$$

$$a_6(4) = \rho(a_6) - a_1 + a_3 - \mathbb{1} \tag{7.7c}$$

$$a_6(5) = \rho(a_6) + \mathbb{1} \tag{7.7d}$$

$$a_7(5) = \rho(a_7) + a_3\ . \tag{7.7e}$$

Note that if $a_5(5) = 1$, then $a_6(5) = 0$, and vice versa. After reading the current state and symbol, we perform state, tape, and head updates (if any) defined by $\mathcal{T}$.

First, consider the tape and head updates. Suppose that $\mathcal{T}$ is defined such that $F(q_1, s_0) = s'$. If $D(q_1, s_0) = 0$ (move head left), then the tape updates from $\ldots s_{-2}s_{-1}\boxed{s_0}s_1s_2\ldots$ to $\ldots s_{-2}\boxed{s_{-1}}s's_1s_2\ldots$ where the head is now reading symbol $\boxed{s_{-1}}$. To represent this tape change in $\mathcal{N}_\mathcal{T}$, the desired actions are:

$$l \to C^-(s_{-2}s_{-3}\ldots) = -4l + (2s_{-1} + 1) \tag{7.8a}$$

$$\begin{aligned} r \to C^+(s_{-1}s's_2\ldots) \quad &= -\tfrac{r}{4} + \tfrac{2s_{-1}+1}{4} \\ &+\tfrac{2s_0+1}{4^2} - \tfrac{2s'+1}{4^2}\ . \end{aligned} \tag{7.8b}$$

On the other hand, if $D(q_1, s_0) = 1$ (move head right), then the tape updates from $\ldots s_{-2}s_{-1}\boxed{s_0}s_1s_2\ldots$ to $\ldots s_{-2}s_{-1}s'\boxed{s_1}s_2\ldots$ where the head is now reading symbol $\boxed{s_1}$. To represent this tape change in $\mathcal{N}_\mathcal{T}$, the desired actions are:

$$l \to C^-(s's_{-1}s_{-2}\ldots) = -\frac{l}{4} + \frac{2s'+1}{4} \tag{7.9a}$$

$$r \to C^+(s_1s_2\ldots) = -4r + (2s_0 + 1)\ . \tag{7.9b}$$

Consider the required update for l. We use the fact that $\rho(l) = l$ since $l \in [0, 3/4]$, so the eventual desired update can be expressed as:

$$\begin{aligned} l(8) = \rho(l) + 4a_5(5)\Big[(1 - D(q_1, 1))\ \tfrac{-5l+2a_2(3)+1}{4} \\ +\tfrac{D(q_1,1)}{4}\ \tfrac{-5l+2F(q_1,1)+1}{4})\Big] + 4a_6(5)\Big[\ (1 - D(q_1, 0)) \\ \tfrac{-5l+2a_2(3)+1}{4} + \tfrac{D(q_1,0)}{4}\ \tfrac{-5l+2F(q_1,0)+1}{4}\Big]\ . \end{aligned} \tag{7.10}$$

Recall that if $a_5(5) = 1$, then $a_6(5) = 0$, and vice versa. Thus only one of the coupling terms involving box brackets in Eq. 7.10 can be non-zero. To re-write Eq. 7.10 using only linear coupling terms allowed in $\mathcal{N}_\mathcal{T}$, we use the identity $a \cdot x = \rho(x + 2a - 2) + 1 - a$ for $a \in \{0, 1\}$ and $x \in [-3/4, 3/4]$. We compute the two terms involving box brackets in Eq. 7.10 separately, and store them in a_3 (recycling an ancillary variable) and a_4:

$$\begin{aligned} a_3(6) \quad &= \rho(a_3) - a_3 + \Big[\tfrac{-5(1-D(q_1,1))}{4} - \tfrac{5D(q_1,1)}{16}\Big]\, l \\ &+\tfrac{1-D(q_1,1)}{2}a_2 + 2a_5 + \Big[\tfrac{1-D(q_1,1)}{4} \\ &+\tfrac{D(q_1,1)F(q_1,1)}{8} + \tfrac{D(q_1,1)}{16} - 2\Big]\,\mathbb{1} \end{aligned} \tag{7.11a}$$

$$a_3(7) = \rho(a_3) - a_5 + \mathbb{1} \tag{7.11b}$$

$$\begin{aligned} a_4(6) \;&= \rho(a_4) + \left[\tfrac{-5(1-D(q_1,0))}{4} - \tfrac{5D(q_1,0)}{16}\right] l \\ &\quad + \tfrac{1-D(q_1,0)}{2} a_2 + 2a_6 + \left[\tfrac{1-D(q_1,0)}{4}\right. \\ &\quad \left. + \tfrac{D(q_1,0)F(q_1,0)}{8} + \tfrac{D(q_1,0)}{16} - 2\right] \mathbb{1} \end{aligned} \tag{7.11c}$$

$$a_4(7) = \rho(a_4) - a_6 + \mathbb{1} \,. \tag{7.11d}$$

Therefore, the desired update for l using only linear couplings is:

$$l(8) = \rho(l) + 4a_3 + 4a_4 \,. \tag{7.12}$$

Now, consider the required update for r. We use the fact that $\rho(r) = r$ since $r \in [0, 3/4]$, so the eventual desired update can be expressed as:

$$\begin{aligned} r(8) = \rho(r) &+ 4a_5(5) \left[\tfrac{1-D(q_1,1)}{4} \left(\tfrac{-5r}{4} + \tfrac{2a_1(3)+1}{16}\right.\right. \\ &\left.\left. - \tfrac{2F(q_1,1)+1}{16} + \tfrac{2a_2(3)+1}{4}\right) + D(q_1,1)\tfrac{-5r+2a_1(3)+1}{4}\right] \\ &+ 4a_6(5) \left[\tfrac{1-D(q_1,0)}{4} \left(\tfrac{-5r}{4} + \tfrac{2a_1(3)+1}{16} - \tfrac{2F(q_1,0)+1}{16}\right.\right. \\ &\left.\left. + \tfrac{2a_2(3)+1}{4}\right) + D(q_1,0)\tfrac{-5r+2a_1(3)+1}{4}\right] \,. \end{aligned} \tag{7.13}$$

Similar to above for l, we compute the two terms involving box brackets in Eq. 7.13 separately, and store them in a_1 and a_2 (recycling ancillary variables):

$$\begin{aligned} a_1(6) = \rho(a_1) &+ \left(\tfrac{-5(1-D(q_1,1))}{16} - \tfrac{5D(q_1,1)}{4}\right) r \\ &+ \left(\tfrac{1-D(q_1,1)}{32} + \tfrac{D(q_1,1)}{2} - 1\right) a_1 + \tfrac{1-D(q_1,1)}{8} a_2 + 2a_5 \\ &+ \left(\tfrac{1-D(q_1,1)}{16} - \tfrac{(1-D(q_1,1))F(q_1,1)}{32} + \tfrac{D(q_1,1)}{4} - 2\right) \mathbb{1} \end{aligned} \tag{7.14a}$$

$$a_1(7) = \rho(a_1) - a_5 + \mathbb{1} \,, \tag{7.14b}$$

$$\begin{aligned} a_2(6) = \rho(a_2) &+ \left(\tfrac{-5(1-D(q_1,0))}{16} - \tfrac{5D(q_1,0)}{4}\right) r \\ &+ \left(\tfrac{1-D(q_1,0)}{32} + \tfrac{D(q_1,0)}{2}\right) a_1 + \left(\tfrac{1-D(q_1,0)}{8} - 1\right) a_2 + 2a_6 \\ &+ \left(\tfrac{1-D(q_1,0)}{16} - \tfrac{(1-D(q_1,0))F(q_1,0)}{32} + \tfrac{D(q_1,0)}{4} - 2\right) \mathbb{1} \end{aligned} \tag{7.14c}$$

$$a_2(7) = \rho(a_2) - a_6 + \mathbb{1} \,. \tag{7.14d}$$

Therefore, the desired update for r using only linear couplings is:

$$r(8) = \rho(r) + 4a_1 + 4a_2 \,. \tag{7.15}$$

Next, consider the state update for q. During the first 8-cycle, if $q(0) = \mathcal{U}(q_1)$, then the desired action $q \to \mathcal{U}(G(q_1, s_0))$. Otherwise, we simply move on to decoding for the next state q_2. This can be implemented in $\mathcal{N}_{\mathcal{T}}$ as:

$$\begin{aligned} q(8) \quad &= \rho(q) - a_3(3)q \\ &\quad + \left[\textstyle\sum_{k=1}^{m-1} \frac{(-1)^{k+1}}{2^k} + \frac{(-1)^m}{2^{m-1}}\mathcal{U}(G(q_1, 1))\right] a_5 \\ &\quad + \left[\textstyle\sum_{k=1}^{m-1} \frac{(-1)^{k+1}}{2^k} + \frac{(-1)^m}{2^{m-1}}\mathcal{U}(G(q_1, 0))\right] a_6 \, . \end{aligned} \tag{7.16}$$

The optical pulse amplitude q will correctly encode the next state $G(q_1, s_0)$ after repeating the analogous 8-cycle another $m - 1$ times. An additional $m - 1$ dummy ones are added to the front of q since the most-significant bit is deleted during each 8-cycle. We use the same identity as above to express the product $a_3(3)q$ in terms of linear couplings stored in ancillary variable a_8:

$$a_8(6) = \rho(a_8) + q + 2a_7 - 2\mathbb{1} \tag{7.17a}$$

$$a_8(7) = \rho(a_8) - a_7 + \mathbb{1} \, . \tag{7.17b}$$

Therefore, the desired update for q using only linear couplings is:

$$\begin{aligned} q(8) \quad &= \rho(q) - a_8 \\ &\quad + \left[\textstyle\sum_{k=1}^{m-1} \frac{(-1)^{k+1}}{2^k} + \frac{(-1)^m}{2^{m-1}}\mathcal{U}(G(q_1, 1))\right] a_5 \\ &\quad + \left[\textstyle\sum_{k=1}^{m-1} \frac{(-1)^{k+1}}{2^k} + \frac{(-1)^m}{2^{m-1}}\mathcal{U}(G(q_1, 0))\right] a_6 \, . \end{aligned} \tag{7.18}$$

More generally, during the i^{th} 8-cycle corresponding to the first step of $\mathcal{T}$, if $q(0) = \mathcal{U}(q_i)$, then we have:

$$\begin{aligned} q(8i) \quad &= \rho(q) - a_8 \\ &\quad + \left[\textstyle\sum_{k=1}^{m-i} \frac{(-1)^{k+1}}{2^k} + \frac{(-1)^{m-i+1}}{2^{m-i}}\mathcal{U}(G(q_i, 1))\right] a_5 \\ &\quad + \left[\textstyle\sum_{k=1}^{m-i} \frac{(-1)^{k+1}}{2^k} + \frac{(-1)^{m-i+1}}{2^{m-i}}\mathcal{U}(G(q_i, 0))\right] a_6 \, . \end{aligned} \tag{7.19}$$

Finally, the last step is to reset the ancillary variables during each 8-cycle:

$$a_j(8) = \rho(a_j) - a_j \quad \text{for } j = 1, 2, \ldots, 8 \, . \tag{7.20}$$

The construction is complete. The coupling weights for the proceeding 8-cycles follows analogously to the first 8-cycle with just minor modifications to the coupling weights according to the definition of $\mathcal{T}$ via the (partial) functions G, F, and D. A visual summary of the construction showing the pulses actively involved during the time steps of each 8-cycle is shown in Fig. 7.3.

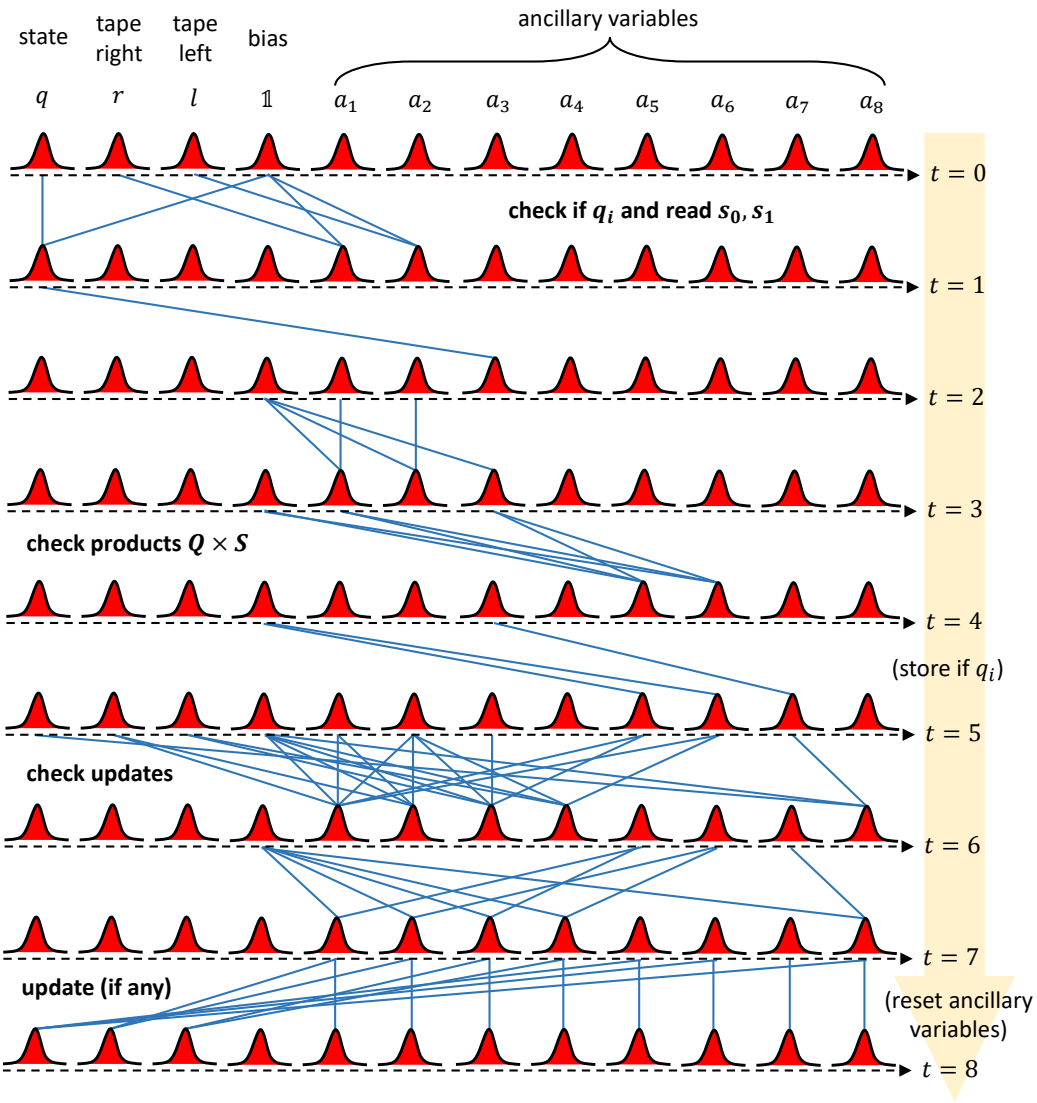

Figure 7.3: **Turing Machine simulation steps.** Network connectivity for a DOPON simulating a TM where red pulses represent the time-multiplexed laser pulses in the DOPON during each time step $t \equiv 0, 1, 2, \ldots, 7 \pmod 8$ and blue lines show the active pulse couplings $J_{ij}(t) \neq 0$ during each 8-cycle. The 8-cycle repeats m times per step of the TM where m is the number of TM finite control states, so the coupling weights $J_{ij}(t)$ repeat with a period of $8m$ to exactly simulate the TM.

□

Physical consequences

The Turing-completeness of DOPONs has many physical implications that result from the undecidability of the *Entscheidungsproblem* (the problem of determining whether an arbitrary program and input will either finish running or continue to

run forever) [31]. Together with Theorem 1, this implies that *the problem of determining if a DOPON (or CNOR more generally), with arbitrary couplings and initial conditions, will eventually reach a steady-state or periodic oscillation is undecidable*. This is because we can construct a DOPON that simulates a Universal Turing Machine according to the scheme used to prove Theorem. 1. Therefore, halting in the TM corresponds to a steady-state/periodic oscillation in the corresponding DOPON.

In fact, any decision problem that can ultimately be reduced to the existence of a steady-state/periodic oscillation in DOPONs is also undecidable. For example, DOPONs have been used extensively as physical Ising solvers for combinatorial optimization [26]. Previous numerical studies [48] have used a time-to-solution to quantify the problem run-time. Notably, this usually involves *ad hoc* disregarding any instance in which the DOPON exceeds some finite cut-off time to reach a steady-state representing the solution to the combinatorial optimization problem. Imposing such a finite cut-off is of course necessary for practical reasons. However, the existence of long-running DOPONs is not just a numerical artifact or bug, but rather it is a feature of their Turing-completeness and moreover *the time-to-solution in DOPONs is uncomputable*. Suppose that there exists a finite time-to-solution T for any DOPON $\mathcal{N}$. Then we can say that $\mathcal{N}$ will be in a steady-state/periodic oscillation after a time of T. But, this implies that the Halting Problem for $\mathcal{N}$ is decidable. This is a contradiction, as discussed above. Therefore, there does not exist any procedure to always find such a finite T.

A related problem is determining if a DOPON will find a local minimum solution or if it will find the desired global minimum solution to an optimization problem. There have been efforts to develop heuristics and scaling laws based on analyses of DOPONs with small N that are tractable, then extrapolating the results to better understand problems with large N [49]. However, this strategy is not prudent because our results imply that there exists a finite N below which the DOPON dynamics are decidable, but above which the DOPON dynamics are undecidable. Therefore, *there is no way to decide if heuristics developed for small N will also be valid for large N*. In fact, in our DOPON model, this threshold is as low as $N = 12$. A caveat is that the heuristics are often not seeking to guarantee optimal solutions, and it may still be possible to develop heuristics that yield near-optimal solutions within some error range of the optimal solution.

7.4 Discussion

Since undecidability is not often discussed in the context of optics, we emphasize that undecidability is a property regarding infinite classes of functions. Specific instances of a problem class are of course solvable. In addition, the simpler the system under consideration, the more powerful the result since adding complexity generally only increases the computational power of a system. Therefore, we consider a simple class of CNORs based on DOPONs so that our results are as widely-applicable as possible. We showed that a DOPON with as few as $N = 12$ pulses is already sufficient for Turing-completeness. This is well-within current experimental capabilities, which means our results have practical consequences. However, we do not claim that our construction is optimal. This naturally begs the question: what is the minimum N needed for Turing-completeness?

There are several obvious methods to further reduce the necessary N. Our construction uses a DOPON that efficiently simulates a TM since the number of time steps is linear $O(T)$. But, using a less efficient TM simulation with an exponential slowdown, e.g., based on Minsky Register Machines [43], can allow one to decrease N at the expense of a worse time-complexity. Another way to reduce N is to use more complicated dynamics. For example, the coupled cavities implementation of DOPONs shown in Fig. 7.1c actually allows more general dynamics than stated in Eq. 7.1. It can be shown that a coupled cavity design with a main cavity containing $N = 6$ pulses and memory cavity containing $N + 1 = 7$ pulses is Turing-complete. The present construction can also be modified for DOPONs with nonlinear function ρ different to the simple saturable gain [50]. It may also be interesting to investigate the effects of different constraints on the computational power of CNORs. For example, many experimental CNORs utilize conservative, static, and nearest-neighbour couplings. In contrast, our construction uses dissipative, time-varying, and all-to-all couplings. We expect that the minimum N needed for Turing-completeness will increase greatly if conservative, static, or nearest-neighbour coupling constraints are imposed.

In any Turing-complete model of computation, there is always the issue of infinity due to the potentially unbounded length of the TM tape, which may be considered unphysical. In our DOPON model, the number of physical components, e.g., waveguides and modulators, is constant. Moreover, the number of pulses N is constant and the pulse amplitudes are bounded. The hidden infinity in our DOPON model is therefore the potentially unbounded precision needed for specifying the

pulse amplitudes l and r that represent the TM tape. Interestingly, there is nothing obvious in classical optics that fundamentally forbids this potentially unbounded precision. Furthermore, in our construction, the coupling weights $J_{ij}(t)$ are purely rational numbers and have a *finite* maximum precision. The need for potentially unbounded precision in Turing-complete DOPONs should not be confused with chaos, which results from a sensitivity to initial conditions. Our results show that even with perfect knowledge of the initial conditions, the dynamics of DOPONs is still unpredictable.

On the other hand, in quantum optics, it is well-known that DOPONs have quantum-noise dynamics, which is often cited as a computational resource for escaping local minima [51]. In this case, there are quantum limits to the maximum precision and signal-to-noise ratio when measuring optical pulse amplitudes. Our simple model for DOPONs operating deterministically above threshold breaks down when considering DOPONs operating stochastically with quantum noise below threshold, or in the case of transitioning stochastically from below to above threshold. It may be interesting to investigate the computational power of stochastic DOPONs in the context of non-deterministic or probabilistic TMs. We leave it as an open problem to study the computational power of DOPONs with finite precision in the context of finite automata, with the ultimate goal of establishing a hierarchy for the computational power of DOPONs given a certain size N and/or finite amount of precision.

In summary, we have shown that coupled nonlinear optical resonators can be used as universal computers. Our proof is based on an explicit construction using a degenerate optical parametric oscillator network, which shows that as few as only $N = 12$ pulses is needed for Turing-completeness. This has profound consequences for both experiments and applications because any physical property reducible to the existence of a steady-state or periodic oscillation is logically undecidable. We hope that our findings will stimulate further inquiry into the computational properties of coupled nonlinear optical resonators and inform their use as part of next-generation optical computing systems.

References

[1] Mian Zhang, Cheng Wang, Yaowen Hu, Amirhassan Shams-Ansari, Tianhao Ren, Shanhui Fan, and Marko Lončar. Electronically programmable photonic molecule. *Nature Photonics*, 13(1):36–40, 2019.

[2] Alexey Tikan, Johann Riemensberger, Kenichi Komagata, Simon Hönl, Mikhail Churaev, Connor Skehan, Hairun Guo, Rui Ning Wang, Junqiu Liu, Paul Seidler, et al. Emergent nonlinear phenomena in a driven dissipative photonic dimer. *Nature Physics*, 17(5):604–610, 2021.

[3] Jesús Yelo-Sarrión, Pedro Parra-Rivas, Nicolas Englebert, Carlos Mas Arabí, François Leo, and Simon-Pierre Gorza. Self-pulsing in driven-dissipative photonic Bose-Hubbard dimers. *Physical Review Research*, 3(4):L042031, 2021.

[4] Alireza Marandi, Zhe Wang, Kenta Takata, Robert L. Byer, and Yoshihisa Yamamoto. Network of time-multiplexed optical parametric oscillators as a coherent ising machine. *Nature Photonics*, 8(12):937–942, 2014.

[5] Midya Parto, William Hayenga, Alireza Marandi, Demetrios N. Christodoulides, and Mercedeh Khajavikhan. Realizing spin hamiltonians in nanoscale active photonic lattices. *Nature Materials*, 19(7):725–731, 2020.

[6] Aleksandr Tusnin, Alexey Tikan, Kenichi Komagata, and Tobias J. Kippenberg. Nonlinear dynamics and kerr frequency comb formation in lattices of coupled microresonators. *Communications Physics*, 6(1):317, 2023.

[7] Yoshitomo Okawachi, Mengjie Yu, Jae K. Jang, Xingchen Ji, Yun Zhao, Bok Young Kim, Michal Lipson, and Alexander L. Gaeta. Demonstration of chip-based coupled degenerate optical parametric oscillators for realizing a nanophotonic spin-glass. *Nature Communications*, 11(1):4119, 2020.

[8] Luqi Yuan, Qian Lin, Meng Xiao, and Shanhui Fan. Synthetic dimension in photonics. *Optica*, 5(11):1396–1405, 2018.

[9] Zhiquan Yuan, Maodong Gao, Yan Yu, Heming Wang, Warren Jin, Qing-Xin Ji, Avi Feshali, Mario Paniccia, John Bowers, and Kerry Vahala. Soliton pulse pairs at multiple colours in normal dispersion microresonators. *Nature Photonics*, 17(11):977–983, 2023.

[10] Qing-Xin Ji, Peng Liu, Warren Jin, Joel Guo, Lue Wu, Zhiquan Yuan, Jonathan Peters, Avi Feshali, Mario Paniccia, John E. Bowers, et al. Multimodality integrated microresonators using the moiré speedup effect. *Science*, 383(6687): 1080–1083, 2024.

[11] Maodong Gao, Zhiquan Yuan, Yan Yu, Warren Jin, Qing-Xin Ji, Jinhao Ge, Avi Feshali, Mario Paniccia, John E. Bowers, and Kerry J. Vahala. Observation

of interband kelly sidebands in coupled-ring soliton microcombs. *Optica*, 11 (7):940–944, 2024.

[12] Arkadev Roy, Rajveer Nehra, Carsten Langrock, Martin Fejer, and Alireza Marandi. Non-equilibrium spectral phase transitions in coupled nonlinear optical resonators. *Nature Physics*, 19(3):427–434, 2023.

[13] Arkadev Roy, Midya Parto, Rajveer Nehra, Christian Leefmans, and Alireza Marandi. Topological optical parametric oscillation. *Nanophotonics*, 11(8): 1611–1618, 2022.

[14] Christian R. Leefmans, Midya Parto, James Williams, Gordon H.Y. Li, Avik Dutt, Franco Nori, and Alireza Marandi. Topological temporally mode-locked laser. *Nature Physics*, pages 1–7, 2024.

[15] Christian R. Leefmans, Nicolas Englebert, James Williams, Robert M. Gray, Nathan Goldman, Simon-Pierre Gorza, François Leo, and Alireza Marandi. Cavity soliton-induced topological edge states. *arXiv preprint arXiv:2311.04873*, 2023.

[16] Arkadev Roy, Saman Jahani, Qiushi Guo, Avik Dutt, Shanhui Fan, Mohammad-Ali Miri, and Alireza Marandi. Nondissipative non-hermitian dynamics and exceptional points in coupled optical parametric oscillators. *Optica*, 8(3):415–421, 2021.

[17] Andrea C. Triscari, Aleksandr Tusnin, Alexey Tikan, and Tobias J. Kippenberg. Quiet point engineering for low-noise microwave generation with soliton microcombs. *Communications Physics*, 6(1):318, 2023.

[18] Igor Kudelin, William Groman, Qing-Xin Ji, Joel Guo, Megan L. Kelleher, Dahyeon Lee, Takuma Nakamura, Charles A. McLemore, Pedram Shirmohammadi, Samin Hanifi, et al. Photonic chip-based low-noise microwave oscillator. *Nature*, 627(8004):534–539, 2024.

[19] Jacob Scheuer, Andrey A. Sukhorukov, and Yuri S. Kivshar. All-optical switching of dark states in nonlinear coupled microring resonators. *Optics Letters*, 35(21):3712–3714, 2010.

[20] Fang Chen and Duanzheng Yao. Tunable multiple all-optical switch based on multi-nanoresonator-coupled waveguide systems containing Kerr material. *Optics Communications*, 312:143–147, 2014.

[21] Xin Zhou and Yidong Chong. PT symmetry breaking and nonlinear optical isolation in coupled microcavities. *Optics Express*, 24(7):6916–6930, 2016.

[22] Xin Zhou, You Wang, Daniel Leykam, and Yi Dong Chong. Optical isolation with nonlinear topological photonics. *New Journal of Physics*, 19(9):095002, 2017.

[23] Shota Yokoyama, Ryuji Ukai, Seiji C. Armstrong, Chanond Sornphiphatphong, Toshiyuki Kaji, Shigenari Suzuki, Jun-ichi Yoshikawa, Hidehiro Yonezawa, Nicolas C. Menicucci, and Akira Furusawa. Ultra-large-scale continuous-variable cluster states multiplexed in the time domain. *Nature Photonics*, 7 (12):982–986, 2013.

[24] Warit Asavanant, Yu Shiozawa, Shota Yokoyama, Baramee Charoensombutamon, Hiroki Emura, Rafael N. Alexander, Shuntaro Takeda, Jun-ichi Yoshikawa, Nicolas C. Menicucci, Hidehiro Yonezawa, et al. Generation of time-domain-multiplexed two-dimensional cluster state. *Science*, 366(6463): 373–376, 2019.

[25] Mikkel V. Larsen, Xueshi Guo, Casper R. Breum, Jonas S. Neergaard-Nielsen, and Ulrik L. Andersen. Deterministic generation of a two-dimensional cluster state. *Science*, 366(6463):369–372, 2019.

[26] Zhe Wang, Alireza Marandi, Kai Wen, Robert L. Byer, and Yoshihisa Yamamoto. Coherent ising machine based on degenerate optical parametric oscillators. *Physical Review A*, 88(6):063853, 2013.

[27] Takahiro Inagaki, Kensuke Inaba, Ryan Hamerly, Kyo Inoue, Yoshihisa Yamamoto, and Hiroki Takesue. Large-scale ising spin network based on degenerate optical parametric oscillators. *Nature Photonics*, 10(6):415–419, 2016.

[28] Jasvith Raj Basani, Mikkel Heuck, Dirk R. Englund, and Stefan Krastanov. All-photonic artificial-neural-network processor via nonlinear optics. *Physical Review Applied*, 22(1):014009, 2024.

[29] Johannes Feldmann, Nathan Youngblood, C. David Wright, Harish Bhaskaran, and Wolfram H.P. Pernice. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature*, 569(7755):208–214, 2019.

[30] Tumi Makinwa, Kensuke Inaba, Takahiro Inagaki, Yasuhiro Yamada, Timothée Leleu, Toshimori Honjo, Takuya Ikuta, Koji Enbutsu, Takeshi Umeki, Ryoichi Kasahara, et al. Experimental observation of chimera states in spiking neural networks based on degenerate optical parametric oscillators. *Communications Physics*, 6(1):121, 2023.

[31] Alan M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1): 230–265, 1937.

[32] Edward Fredkin and Tommaso Toffoli. Conservative logic. *International Journal of Theoretical Physics*, 21(3):219–253, 1982.

[33] Cristopher Moore. Unpredictability and undecidability in dynamical systems. *Physical Review Letters*, 64(20):2354, 1990.

[34] Newton C.A. da Costa and Francisco A. Doria. Undecidability and incompleteness in classical mechanics. *International Journal of Theoretical Physics*, 30:1041–1073, 1991.

[35] Terence Tao. Finite time blowup for an averaged three-dimensional navier-stokes equation. *Journal of the American Mathematical Society*, 29(3):601–674, 2016.

[36] Robert Cardona, Eva Miranda, Daniel Peralta-Salas, and Francisco Presas. Constructing turing complete euler flows in dimension 3. *Proceedings of the National Academy of Sciences*, 118(19):e2026818118, 2021.

[37] Toby S. Cubitt, David Perez-Garcia, and Michael M. Wolf. Undecidability of the spectral gap. *Nature*, 528(7581):207–211, 2015.

[38] Johannes Bausch, Toby S. Cubitt, and James D. Watson. Uncomputability of phase diagrams. *Nature Communications*, 12(1):452, 2021.

[39] Naoto Shiraishi and Keiji Matsumoto. Undecidability in quantum thermalization. *Nature Communications*, 12(1):5084, 2021.

[40] Arthur Komar. Undecidability of macroscopically distinguishable states in quantum field theory. *Physical Review*, 133(2B):B542, 1964.

[41] Yuji Tachikawa. Undecidable problems in quantum field theory. *International Journal of Theoretical Physics*, 62(9):199, 2023.

[42] Robert Geroch and James B. Hartle. Computability and physical theories. *Foundations of Physics*, 16:533–550, 1986.

[43] Marvin Lee Minsky. *Computation: Finite and Infinite Machines*. Prentice-Hall Englewood Cliffs, 1967.

[44] Robert M. Gray, Ryoto Sekine, Luis Ledezma, Gordon H.Y. Li, Selina Zhou, Arkadev Roy, Midya Parto, and Alireza Marandi. Large-scale time-multiplexed nanophotonic parametric oscillators. *arXiv preprint arXiv:2405.17355*, 2024.

[45] Luis Ledezma, Ryoto Sekine, Qiushi Guo, Rajveer Nehra, Saman Jahani, and Alireza Marandi. Intense optical parametric amplification in dispersion-engineered nanophotonic lithium niobate waveguides. *Optica*, 9(3):303–308, 2022.

[46] Cheng Wang, Mian Zhang, Xi Chen, Maxime Bertrand, Amirhassan Shams-Ansari, Sethumadhavan Chandrasekhar, Peter Winzer, and Marko Lončar. Integrated lithium niobate electro-optic modulators operating at cmos-compatible voltages. *Nature*, 562(7725):101–104, 2018.

[47] Hava T. Siegelmann and Eduardo D. Sontag. On the computational power of neural nets. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pages 440–449, 1992.

[48] Sam Reifenstein, Satoshi Kako, Farad Khoyratee, Timothée Leleu, and Yoshihisa Yamamoto. Coherent ising machines with optical error correction circuits. *Advanced Quantum Technologies*, 4(11):2100077, 2021.

[49] Edwin Ng, Tatsuhiro Onodera, Satoshi Kako, Peter L. McMahon, Hideo Mabuchi, and Yoshihisa Yamamoto. Efficient sampling of ground and low-energy ising spin configurations with a coherent ising machine. *Physical Review Research*, 4(1):013009, 2022.

[50] Gordon H.Y. Li, Ryoto Sekine, Rajveer Nehra, Robert M. Gray, Luis Ledezma, Qiushi Guo, and Alireza Marandi. All-optical ultrafast relu function for energy-efficient nanophotonic deep learning. *Nanophotonics*, 12(5):847–855, 2023.

[51] Yoshihisa Yamamoto, Kazuyuki Aihara, Timothee Leleu, Ken-ichi Kawarabayashi, Satoshi Kako, Martin Fejer, Kyo Inoue, and Hiroki Takesue. Coherent ising machines—optical neural networks operating at the quantum limit. *npj Quantum Information*, 3(1):49, 2017.