Large-Scale Optical Reservoir Computing for Spatiotemporal Chaotic Systems Prediction

Mushegh Rafayelyan, Jonathan Dong, Yongqi Tan, Florent Krzakala, and Sylvain Gigan

Laboratoire Kastler Brossel, Sorbonne Université, École Normale Supérieure-Paris Sciences et Lettres (PSL) Research University, Centre National de la Recherche Scientifique (CNRS) UMR 8552, Collège de France, 24 rue Lhomond, 75005 Paris, France
Laboratoire de Physique de l’École Normale Supérieure, Université Paris Sciences et Lettres (PSL), Centre National de la Recherche Scientifique (CNRS), Sorbonne Université, Université Paris-Diderot, Sorbonne Paris Cité, 24 rue Lhomond, 75005 Paris, France.

Reservoir computing is a relatively recent computational paradigm that originates from a recurrent neural network, and is known for its wide-range of implementations using different physical technologies. Large reservoirs are very hard to obtain in conventional computers as both the computation complexity and memory usage grows quadratically. We propose an optical scheme performing reservoir computing over very large networks of up to $10^6$ fully connected photonic nodes thanks to its intrinsic properties of parallelism. Our experimental studies confirm that in contrast to conventional computers, the computation time of our optical scheme is only linearly dependent on the number of photonic nodes of the network, which is due to electronic overheads, while the optical part of computation remains fully parallel and independent of the reservoir size. To demonstrate the scalability of our optical scheme, we perform for the first time predictions on large multidimensional chaotic datasets using the Kuramoto-Sivashinsky equation as an example of a spatiotemporal chaotic system. Our results are extremely challenging for conventional Turing-von Neumann machines, and they significantly advance the state-of-the-art of unconventional reservoir computing approaches in general.

I. INTRODUCTION

Recent studies in machine learning have shown that large neural networks can dramatically improve the network performance, however, their realization with conventional computing technologies is to date a significant challenge. Towards this end, a number of alternative computing approaches have emerged recently. Among them, one of the most studied approaches is reservoir computing (RC). RC is a relatively recent computational framework derived from independently proposed Recurrent Neural Network (RNN) models, such as echo state networks (ESNs) and liquid state machines (LSMs). The main objective of ESN and LSM was the significant simplification of the RNN training algorithm by using fixed random injection and fixed internal connectivity matrices. However, it was rapidly understood that the temporally fixed connections allows for the straightforward implementation of RC in optics, electronics, spintronics, mechanics, biology, and in other fields. Optics is one of the most promising fields to realize large and efficient neural networks due to its intrinsic properties of parallelism and its ability to process the data at the speed of light and low energy consumption.

There are many interesting approaches to realize photonic reservoir networks based on both time- and spatial-multiplexing of photonic nodes. The first approach is based on a single nonlinear node with a time-delayed optoelectronic or all-optical feedback in order to get time-multiplexed virtual nodes in the temporal domain. Such architectures can reach supercomputer performances, e.g., gigabyte per second data rates for chaotic time-series prediction tasks or million words per second classification for speech recognition tasks. However, their information processing rate is inherently limited as it is inversely proportional to the number of virtual nodes of the reservoir. Furthermore, a preprocessing of the input information is required, according to the initially defined virtual nodes, which can bring additional complexity to the problem, especially for the large multidimensional inputs. To this end, multi-channel delay-based RC architectures consisting of several nonlinear nodes are of special interest.

Another popular approach of photonic RC is based on spatially distributed nonlinear nodes. The latter is endowed by its intrinsic property to process large-scale input information without sacrificing the computation speed. Several theoretical and experimental studies have been performed using on-chip silicon photonics reservoirs consisting of optical waveguides, optical splitters, and optical combiners. As reported in, a 16-node reservoir network of modest sizes can reach high information processing bitrates, up to speeds $> 100$ Gbit s$^{-1}$. Another approach towards the spatially extended photonics reservoir is based on a network of vertical-cavity surface-emitting lasers (VCSEL) and a standard diffractive optical element (DOE) providing the complex interconnections between the reservoir nodes. The bias current of each laser can be controlled individually, which allows the encoding of the input data.

Recently, a new approach to spatially scalable photonics reservoir has been introduced based on both liquid crystal spatial light modulators (SLM) and digital micromirror device (DMD). In particular, Bueno et al. in demonstrated a reservoir network of up to 2500
diffractively coupled photonic nodes using a liquid crystal SLM coupled with a DOE and a camera. The input and output information in their network is provided via single nodes. This last limitation has been waived by Dong et al. in [38] using a DMD to encode both the reservoir and the input information through the binary intensity modulation of the light. Later, Dong et al. in [39] implemented the same approach to get large-scale optical reservoir networks using a phase-only SLM that could provide an 8-bit encoding of the reservoir and the input information through the spatial phase profile of the light instead of the former binary encoding option. We stress out that the key element in both aforementioned optical networks was the strongly scattering medium that guaranteed a random coupling weights of very large number of photonic nodes and their parallel processing. Such networks practically can host as many nodes as the number of pixels provided by the DMD and the camera [41, 42].

In this work we exploit the potential of the platform provided by [38, 39] to extend our recent achievements towards multidimensional large chaotic systems predictions. Accordingly, we report on the first experimental realization of recently introduced state-of-the-art benchmark test [13], performing recursive predictions on the Kuramoto-Sivashinsky (KS) chaotic systems. To highlight the scalability of our approach, we measure the computation time of similar reservoir networks provided either by an high-end conventional computer or by our optical scheme. In contrast to conventional computers, where the time of the computation scales quadratically against the size of the network, the computation time of our optical scheme is almost independent of the number of photonic nodes. More precisely, we observe a relatively mild linear dependence due to electronic overheads, while the optical computation remains fully parallel and independent of the reservoir size. Our results are hardly reachable by the conventional Turingvon Neumann machines, and they significantly advance the state of the art of the unconventional reservoir computing approaches in general.

II. CONVENTIONAL RESERVOIR COMPUTING

We now briefly introduce the concept of conventional RC. An input vector $i(t)$ of dimension $D_{in}$ is injected to a high-dimensional dynamical system called the “reservoir” (see Fig. 1(a)). The reservoir is described by a vector $r(t)$ of dimension $D_{res}$ that is the number of reservoir nodes. The initial state of the reservoir is defined randomly. Let $W_{res}$ matrix defines the internal connections of the reservoir nodes and $W_{in}$ matrix defines the connections between the input and the reservoir nodes. Both matrices are initialized randomly and fixed during the whole RC process. The state of each reservoir node is a scalar $r_j(t)$ which evolves according to the following recursive relation

$$r(t + \Delta t) = f \left[ W_{in} i(t) + W_{res} r(t) \right],$$

where $\Delta t$ is the discrete time-step of the input, $f$ is an element-wise nonlinear function. According to the Eq. (1), the reservoir is defined as a high-dimensional dynamical system endowed with a unique memory property, namely, each consequent state of the reservoir contains some exponentially decaying information about its previous states and about the inputs injected until that moment. Noteworthy, the memory size of the reservoir is mainly defined by the number of reservoir nodes and the nonlinear activation function $f$.

During the training phase, the input $i(t)$, defined in the time-interval $-T \leq t \leq 0$, is fed to the reservoir, and the corresponding reservoir states are recursively calculated. The final step of the information processing is to perform a simple linear regression that adjusts the $W_{out}$ weights so that their linear combination with the calculated reservoir states makes the actual output $o(t)$ to be as close as possible to the desired output $\hat{o}(t)$

$$\text{RMSE} = \sqrt{\frac{1}{D_{out}} \sum_{t=-T}^{0} \| \hat{o}(t) - o(t) \|^2},$$

where

$$o(t) = W_{out} \cdot r(t),$$

$$W_{out} = \text{argmin}(\text{RMSE}).$$

RMSE is the root mean square error, and $D_{out}$ is the number of the output nodes, i.e., the dimension of the vector $o(t)$. An additional regularization term $\lambda \| W_{out} \|^2$ ($\lambda$ is a scalar) can be used to find the solution of Eq. (4) to avoid overfitting, especially when the number of reservoir nodes is larger than the number of training examples. Note, the output weights are the only parameters that are modified during the training. The random input and reservoir weights are fixed throughout the whole computational process and they serve to randomly project the input into a high-dimensional space, which increases the linear separability of inputs.

![FIG. 1. The sketch of the conventional reservoir computing paradigm in (a) training and (b) predicting phases. The vectors $i(t)$, $r(t)$ and $o(t)$ describe the injected input, the corresponding reservoir states and the trained output, respectively. All three layers of the network are described by $W_{in}$, $W_{res}$ and $W_{out}$ interconnection matrices. The first two ones are initialized randomly and are held fixed throughout the whole computation process, while the last one is trained by linear regression. In the prediction phase, the feedback loop from the predicted output defines the next injected input.](Image)
In order to perform predictions about $t > 0$ future evolution of $i(t)$ using the calculated reservoir states $r(t)$ in $-T \leq t \leq 0$, one needs to train the output weights $W_{\text{out}}$ to predict the next time-step of the input, namely $\tilde{o}(t) = i(t + \Delta t)$. Afterwards, the future evolution of $i(t)$ for $t > 0$ can be predicted replacing the input by the subsequent prediction $o(t)$, as shown in Fig. 1(b). Consequently, during the prediction the reservoir evolves step by step, by replacing the subsequent input with the last prediction every time.

III. OPTICAL RESERVOIR COMPUTING

The experimental setup to perform the optical RC is shown in Fig. 2 and detailed in the Appendix. The key optical components in the setup are the phase-only SLM, the scattering medium and the camera. The SLM provides both the encoding of the input vector $i(t)$ of dimension $D_{\text{in}}$ and the encoding of the subsequent reservoir state $r(t)$ of dimension $D_{\text{res}}$ (total dimension $D_{\text{in}} + D_{\text{res}}$) into the phase spatial profile of the light. The scattering medium ensures their random linear mixing which is equivalent to their linear multiplications with large dense random matrices consisting of independent and identically distributed (i.i.d.) random complex variables [44, 45] (see more details about light scattering in the Appendix). Finally, the camera performs a nonlinear readout of the complex field intensity for the next reservoir state $r(t + \Delta t)$, that is sent back by the computer to the SLM in order to be displayed with new input, and the process repeats. The upper and the lower insets in Fig. 2 are respective examples of images displayed on the SLM and detected by the camera.

There are a number of tunable parameters regarding the encoding of the input and the reservoir states onto the SLM that we will describe here. Without loss of generality, we assume that the the number of grey levels of the camera and the SLM are equal to 256. The SLM is calibrated such that the grey levels from 0 to 255 linearly map to phase delays of 0 to $2\pi$. Furthermore, we assume without loss of generality that the whole input dataset is initially scaled from 0 to 255 and the acquisition time of the camera is initially adjusted to provide unsaturated reservoir states again ranging from 0 to 255. Accordingly, the encoding of the input and the reservoir states onto the SLM can be described by $i(t) \rightarrow s_{\text{in}}i(t)$ and $r(t) \rightarrow s_{\text{res}}r(t)$ with two scaling factors $0 \leq s_{\text{in}}/s_{\text{res}} \leq 1$. These modifications are performed in the computer, every time before sending the input and reservoir states to the SLM. Additionally, each scalar value from the input and reservoir states can be encoded into multiple number of SLM pixels forming a macropixel. The number of pixels in one macropixel is denoted by $p_{\text{in}}$ for the input encoding, and $p_{\text{res}}$ for the reservoir states encoding. Accordingly, the reservoir computing in our optical scheme can be described by the following recursive relation

$$r(t + \Delta t) = F \left[ s_{\text{res}}r(t) \oplus J_{\text{pout}}s_{\text{in}}i(t) \oplus J_{\text{pin}} \right],$$

where the function $F$ stands for the whole optical setup, i.e., it takes the encoded matrices corresponding to the input and the reservoir state as two arguments, sends them to the SLM, and returns the next reservoir state detected by the camera. The symbol $\oplus$ refers to the Kronecker product and $J_{\text{pin/}}_{\text{res}}$ refers to the all-ones matrix with $p_{\text{in/}}_{\text{res}}$ number of rows and columns in order to ensure the macropixel encoding of the SLM.

In order to get a more detailed description of our optical scheme, we also provide a mathematical relation that models the light propagation and the consequent RC with well-known mathematical functions:

$$r(t + \Delta t) = f \left[ W_{\text{res}}g \left( r(t) \right) + W_{\text{in}}g \left( i(t) \right) \right],$$

where $W_{\text{res}}$ and $W_{\text{in}}$ are random dense matrices describing the scattering of the light in the setup. $f$ and $g$, are nonlinear functions associated with the intensity readout by the camera and the phase encoding by the SLM, respectively. Namely, for a vector $q = [q_1, q_2, \ldots]^T$, $f(q) = [\exp(\pi sq_1), \exp(\pi sq_2), \ldots]^T$ and $g(q) = [\exp(\pi sq_1), \exp(\pi sq_2), \ldots]^T$ with $0 \leq s \leq 2$. Note that all above mentioned operations are implicitly included in the function $F$ in Eq. (5).

The mathematical framework describing our optical network is very similar to the conventional RC network

![FIG. 2. Experimental setup to perform an optical reservoir computing. The SLM receives from the computer the consequent input $i(t)$ concatenated with the reservoir state $r(t)$ and imprints it into the spatial phase profile of the reflected beam (see the upper inset as a typical example). The scattering medium (SM) provides a complex linear mixing of the whole encoded information. Finally, the camera performs a nonlinear readout for the next reservoir state $r(t + \Delta t)$ (see the lower inset as a typical example), which is sent by the computer back to the SLM to be displayed with new input, and the process repeats. LP1, LP2: linear polarizers; HWP: half-wave plate; BE: beam expander; BS: beam splitter; O1, O2: objectives.](image)
provided by Eq. (1). The main difference is that an additional nonlinear function, a complex exponent, is applied in Eq. (3) to account for the phase encoding of the SLM. One can also note, that $W_{\text{res}}$ and $W_{\text{in}}$ are complex-valued matrices here in contrast to the conventional RC where the connection matrices are real valued. Accordingly, Eq. (3) and Eq. (6) together give the whole picture of information processing in our optical scheme.

During the training phase, as soon as the reservoir states for the given time interval $-T \leq t \leq 0$ are optically calculated, a simple linear regression is executed in the conventional computer to adjust the $W_{\text{out}}$ weights such that their linear combination with the calculated reservoir states makes the actual output to be as close as possible to the next time step of the input $i(t + \Delta t)$ (see Eqs. (2)–(4)). Finally, to predict the future evolution of $i(t)$ for $t > 0$, we make a feedback loop from the output to the input by replacing the next input $i(t + \Delta t)$ on the SLM with the one-step prediction $W_{\text{out}}r(t)$, as it was done in conventional RC in Fig. 1(b).

In general, the RC and its different optical implementations have proven to be very successful for various tasks, such as spoken digits recognition, Temporal XOR task, Santa Fe, MG or NARMA time series prediction [5, 9, 11, 13, 17, 27, 46]. Recently, Pathak et al. [43, 47] proposed a new state-of-the-art benchmark test performing predictions on KS spatiotemporal chaotic datasets with the conventional RC (see more details about the KS equation in the Appendix). In the next section, we will use the optical RC setup of Fig. 2 to predict the dynamical evolution of KS spatiotemporal chaotic systems.

**IV. EXPERIMENTAL RESULTS**

Initially, we apply the optical RC to the spatiotemporal KS datasets with a similar set of parameters as reported in [43]. Namely, the spatial domain size $L$ of the scalar field $u(x, t)$ is $L = 22$ in the KS equation (see Eq. (7) in the appendix), which is integrated on the grid of $N_x = 64$ equally spaced spatial points and $N_t = 90500$ equally spaced time steps with $\Delta t = 0.25$ using an open-source code from [48]. The first $9 \cdot 10^5$ time steps of the dataset are used to train the optical reservoir, while the remaining 500 time steps are kept in order to be compared with predicted data. The input and reservoir sizes are $D_{\text{in}} = 64$ and $D_{\text{res}} = 10^4$, respectively.

In general, it is believed that the optimum prediction performance of RC schemes is reached when the reservoir computer parameters are tuned to the edge of chaos [49]. Accordingly, before starting the actual experiment, we perform a grid search to optimize a set of tunable parameters in our optical scheme. It turns out that the optimal prediction performance is observed when $s_{\text{res}} = s_{\text{in}} = 0.5$, i.e., the input and reservoir states are encoded between 0 and 128 thus providing a phase modulation of the light from 0 to $\pi$. Furthermore, the macropixel sizes are taken $p_{\text{res}} = 64$ and $p_{\text{in}} = 10000$ to ensure equal importance ratios between the input and reservoir states encoded on the SLM. Consequently, during the RC process, the total number of pixels occupied on the SLM by the input and the reservoir states together is equal to $p_{\text{res}}D_{\text{res}} + p_{\text{in}}D_{\text{in}} = 128 \cdot 10^4$. We also apply a slight regularization with $\lambda = 0.07$ during the linear regression process (see Eqs. (2)–(4)). Noteworthy, the nonlinear activation function provided by the camera intensity readout may easily be further tuned throughout the grid search process as well. There are two relatively simple options to tune the nonlinear readout that we could explore in the future: to change the camera gain parameter as an analog solution or to apply an additional nonlinear function in the computer on the detected camera image as a numerical solution. Both approaches may improve the performance of our optical scheme, but for the sake of simplicity, we remained with the basic nonlinearity provided by the system that already provides good results.

Fig. 3 shows an example of the true KS dataset (see panel (a)), the corresponding prediction (see panel (b)), and their difference (see panel (c)). As it is seen, the optical reservoir network can predict with excellent accuracy the dynamical change of the KS dataset up to two Lyapunov time. Lyapunov time is a characteristic quantity of the dynamical chaotic systems defining the minimum amount of the time for two infinitesimally close states.
FIG. 4. (a) Normalized root mean square errors (NRMSE) calculated for 100 set of training and testing KS datasets having the same parameters of the problem as in Fig. 3. (b) the mean NRMSE as a result of averaging the panel (a) along its vertical axis.

of the system to diverge by a factor of $e$. The latter is defined by the largest Lyapunov exponent $\Lambda_{\text{max}}$, and in this particular case $\Lambda_{\text{max}} = 0.043$ (see the Appendix and Table I). Furthermore, for quantitative analyses, we repeat the same experiment of Fig. 3 for 100 different sets of training and testing datasets. The RMSE values for each testing sample is calculated and normalized according to the RMSE of a random prediction, namely $\sigma(t)$ is a random matrix having the same dimensions as $\tilde{o}$. Accordingly, the normalized RMSE (NRMSE) value close to one means that the network does not perform better than a random prediction. Fig. 4 shows the NRMSE dependencies for each testing sample (see panel (a)) and the mean NRMSE curve averaged over all the 100 samples (see panel(b)). We note that the prediction performance varies significantly depending on the test sample, as seen from Fig. 4(a). This effect is related to the RC algorithm in general which is addressed in [50].

Although the prediction results of Fig. 3 and Fig. 4 indicate the potential of the optical RC to predict large spatiotemporal chaos, we emphasize that, for larger sizes of the problem, i.e., for larger values of $L$, in order to get qualitatively similar prediction performances, one needs to increase the size of the reservoir. To this end, we performed experiments applying the same reservoir network hosting $D_{\text{res}} = 10^4$ photonic nodes on KS datasets with the spatial sizes of $L = 12, 22, 36, 60, \text{ and } 100$. As seen in Fig. 4(a), the prediction performance of optical RC decreases rapidly as the system size $L$ increases. On the other hand, for the given KS dataset of spatial size $L = 60$, Fig. 4(b) shows that the prediction performance of our optical scheme is recovered back by increasing the size of the network. In both plots, the temporal axis is normalized according to the $\Lambda_{\text{max}} = 0.043$ corresponding to $L = 22$, however, we note that the value of the largest Lyapunov exponent is dependent on the spatial domain size $L$ of the system (see Table I in Appendix). Finally, the different reservoir dimensions in Fig. 4(a) imply different macropixel sizes of encoding in order to maintain the same overall encoding number of pixels on the SLM corresponding to the reservoir states.

Note that, the realization of large reservoir networks in conventional computing is not an easy task since the computation time grows quadratically with respect to the number of network nodes. Therefore, Pathak et al. proposed in [43] a new scheme consisting of a large set of parallel reservoirs of moderate sizes, each of which predicts a local region of the spatiotemporal chaos. However, in optical RC we are able to realize large networks due to its intrinsic properties of parallelism. As a proof of principle, we performed a number of experiments on our optical scheme for different reservoir sizes and recorded the average time of the reservoir updating process. We use the same parameters of the problem as in Fig. 3, but without applying a Kronecker product in Eq. (5), i.e., taking $p_{\text{res}} = p_{\text{in}} = 1$. Consequently, each pixel of the SLM is one node in the optical network. We also performed numerical computations with conventional RC for the same reservoir sizes. Fig. 5 shows that the op-
tical RC is relatively slower than the conventional RC only for small reservoir sizes, $D_{\text{res}} < 25000$. The situation changes rapidly for large network sizes, since the computation time of optical RC scales with a mild linear dependence with respect to the number of nodes of the reservoir, in contrast to the conventional RC, which exhibits a quadratic growth in time. Hence, for large reservoir sizes, our optical network is much faster than conventional reservoir computers. Noteworthy, the optical computation in our setup is inherently parallel, and the linear slope is only due to the limited communication bandwidth from the camera to the SLM. Furthermore, the large reservoirs require tremendous sizes of operating memory from the conventional computers to store the large random connection matrices $W_{\text{res}}$ and $W_{\text{in}}$, while our optical scheme can leverage large networks of $10^6$ photonic nodes without using large sizes of operating memory. We note that the conventional RC tests have been performed on a high-end computer with one of the latest generation processors of Intel having 14 cores and supported by 64 GB operative memory. We emphasize that presently faster SLMs and cameras are available that can considerably lower the absolute time of computation in our optical scheme, maintaining its linear dependence on the size of the reservoir (see more information about the SLM and the camera used in our setup in the Appendix).

Finally, we stress that the advantage of our optical scheme over other optical realizations is not only due to the possibility of using a large number of pixels from the camera and SLM as nodes in the optical network. An important advantage lies in using the complex of the multiply scattering medium, that corresponds a random mixing of millions of SLM modes to millions of CCD pixels, which allows to reach such large network sizes. Wavefront shaping techniques have already reached the million mode milestone, e.g. in [42], where authors achieved a light focusing through the scattering medium with an unprecedented enhancement factor. Relatively large network sizes are also reachable using diffractive optics, for instance, the possibility to reach up to 30000 nodes has been claimed in [52], however, without all-to-all random connectivity allowed by the complex mixing process.

V. DISCUSSION AND CONCLUSION

To estimate the computing performances our simple setup can reach, we can estimate the average number of operations per second performed during the process of the RC. As a rough estimate, the optical scheme we propose can host $10^6$ photonic nodes in the network (limited by the pixel numbers on SLM and CCD respectively). One iteration of the network approximately corresponds to $10^{12}$ trivial mathematical operations in Eq. (1), such as multiplication, sums, etc. Assuming that the SLM and the camera have typical speeds of 100 Hz, our optical setup will perform on the order of $10^{14}$ OPS (operations per second). This is not far from the current state-of-the-art of supercomputers, which ranges from $10^{15}$ OPS to $10^{17}$ OPS. Consequently, without significant energy consumption and nor a large number of processing units, the optical setup we propose can perform an RC close to the performances of the supercomputers of current state-of-the-art technology. Note that similar calculation have been performed using the hardware of LightOn, with different modulation scheme (binary amplitude modulation) in [55, 56].

Although light propagation in our optical setup provides fully parallel information processing independently of the size of the network, Fig. 6 shows that the electronic feedback from the camera to SLM is a bottleneck resulting in a slight linear growth of the overall computation time as the amount of the data increases. One way to overcome this might be the use of the field-programmable gate arrays (FPGAs) instead of the computer in the setup to provide the information transfer in much larger bandwidths. Furthermore, FPGAs contain an array of programmable logic blocks that can be configured to apply a given complex operation on the data transferred from the camera to the SLM. Another approach that can impact the overall computation speed is based on nonlinear light-matter interactions, where the naturally generated response from the matter can be used as feedback of the RC network [57, 58].

In conclusion, we proposed an optical reservoir computing network that can perform, for the first time to our knowledge, predictions on large multidimensional chaotic datasets. We used the Kuramoto-Sivashinsky equation as an example of a spatiotemporal chaotic system. Our predictions on the chaotic systems of large spatial sizes confirm that in order to have comparable prediction performances one has to increase the optical network sizes too. Finally, we experimentally demonstrated that our optical network can be scaled to a million of nodes. Its computation time only grows linearly with the the number of nodes increases, due to electronic overheads, while the speed of the optical part (the matrix multiplication) is independent of the reservoir size and does not require any memory storage. Our results, that are very hard to achieve by conventional Turingvon Neumann machines, open the prospect to achieve predictions on very large datasets of practical interest, such as turbulence, at high speed and low energy consumption.

ACKNOWLEDGEMENTS

We acknowledge funding from the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00111890042. Sylvain Gigan and Jonathan Dong also acknowledge partial support from H2020 European Research Council (ERC) (Grant 724473).
VI. APPENDIX: METHODS

A. Experimental setup

The laser beam with 532 nm wavelength is expanded using a beam expander (BE) with 10× optical magnification. The linear polarizer LP1 and the half-wave plate (HWP) are used to polarize the light parallel to the extraordinary axis of the liquid crystal SLM to ensure a pure phase shaping of the light. The SLM receives from the computer the consequent input information $i(t)$ concatenated with the reservoir state $r(t)$ at the given moment and imprints it into the phase spatial profile of the reflected beam. The light propagates further through the first objective O1 with 10× optical magnification and numerical aperture $NA = 0.1$. Furthermore, the light gets focused on the strongly scattering medium (SM) with approximately 0.5 mm thickness and the scattered light is collected by the second objective O2 with 20× optical magnification and numerical aperture $NA = 0.4$. The resulted intensity speckle pattern is detected by the CMOS camera. We use a second linear polarizer LP2 in front of the camera, with the polarization axis oriented orthogonal to the initial polarization of the beam in order to enhance the contrast of detected speckle pattern. In the final stage, the camera sends back to the computer the detected speckle pattern as a new state of the reservoir, that is going to be displayed on the SLM with new input, and the process repeats. We have used in our experimental setup a liquid crystal SLM from Meadowlark Optics (model: HSP192-532) and CMOS camera from Basler (model: acA2040-55um) respectively having 1920 x 1152 and 2048 x 1536 spatially distributed pixels and respectively providing $\sim 50$ Hz and 64 Hz speeds at fully functioning regimes.

B. Light scattering

When light encounters refractive index inhomogeneities, it gets scattered and its direction of propagation is modified. Light scattering through the thick scattering medium is a complex process accompanied by a tremendously high number of scattering events and at the exit of the scattering medium, one typically observes a speckle pattern. The speckle pattern is the total interference between all complex scattering paths. Thanks to a large number of scattering events, the speckle image is seemingly random and its statistical properties are well characterized [59]. It represents a signature of the particular disordered medium and for a given incident field will be different from one scattering sample to another.

Light propagation in the multiple scattering regime still is a linear process. Therefore, the output over a set of detectors for the given set of input sources can be described as the product between the incident electric field and the Transmission Matrix (TM). So, the TM is a characteristic for the particular setup including the input sources, output detectors, and all the optical elements with the scattering medium used inside the setup. As shown in [41, 45], the TM is a dense random matrix when a thick disordered medium is placed between the Spatial Light Modulator (SLM) and the camera, and it can be measured experimentally. Nowadays, SLMs and cameras based on silicon photonics can afford a few millions of pixels, thus the TM in conventional computers can reach gigantic sizes. We cannot possibly hope to measure such a large matrix, as it will require very large computation time, and it would be impossible to store it in the memory of a computer. However, we can leverage the very large dimensionality of TM without measuring it by using in well-developed algorithms where the explicit form of the TM is not required [44]. One of those algorithms is reservoir computing (RC), which requires large random matrices held fixed throughout the whole computation process.

C. Kuramoto-Sivashinsky time series

KuramotoSivashinsky (KS) equation is a model of nonlinear partial differential equation frequently encountered in the study of nonlinear chaotic systems with intrinsic instabilities, such as wave propagation in chemical reaction-diffusion systems, the velocity of laminar flame front instabilities, thin fluid film flow down inclined planes and hydrodynamic turbulence [60]. Very interestingly, a chimera state which is an unexpected solution arising in the electro-optic delayed dynamical systems can also be described by the KS equation [15]. The one-dimensional KuramotoSivashinsky partial differential equation is

\[ u_t = -u_{xx} - u_{xxxx}, \]

where we assume that the scalar field $u = u(x,t)$ is periodic with period $L$, $u(x+L,t) = u(x,t)$, thus the solution is defined in the interval $[0, L]$. Note that the dimension of the attractor is defined by the value of $L$ and the dependence is linear for large values of $L$. We integrate the Eq. [7] on a grid of $Q = 64$ equally spaced spatial points with $\Delta t = 0.25$ time-step as in [43]. The obtained solution will contain $Q$ time series, which we denote by the vector $u(t)$ and use as the reservoir input.

The dynamics of chaotic systems can be described by a quantity called Lyapunov exponent that measures the exponential divergence of initially close trajectories in the phase space of the system. In dynamical system theory, a phase space is a space in which all possible states of a system are represented as unique points. As is known, the

\begin{table}[h]
\centering
\caption{The largest Lyapunov exponent for different spatial domain sizes.}
\begin{tabular}{|c|c|c|c|c|}
\hline
$L$ & 12 & 22 & 36 & 60 & 100 \\
\hline
$\Lambda_{\text{max}}$ & 0.003 & 0.043 & 0.080 & 0.089 & 0.088 \\
\hline
\end{tabular}
\end{table}
spatial domain size $L$ of the KS system strongly affects its dynamics thus changing the corresponding largest Lyapunov exponent. We provide in Table I the $\Lambda_{\text{max}}$ values for typical domain sizes as measured in [61].

[1] D. Verstraeten, B. Schrauwen, M. dHaene, and D. Stroobandt, An experimental unification of reservoir computing methods, Neural networks 20, 391 (2007).
[2] M. Lukosevičius and H. Jaeger, Reservoir computing approaches to recurrent neural network training, Computer Science Review 3, 127 (2009).
[3] H. Jaeger, The echo state approach to analysing and training recurrent neural networks with an erratum note, Bonn, Germany: German National Research Center for Information Technology GMD Technical Report 148, 13 (2001).
[4] W. Maass, T. Natschläger, and H. Markram, Real-time computing without stable states: A new framework for neural computation based on perturbations, Neural computation 14, 2531 (2002).
[5] G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, and A. Hirose, Recent advances in physical reservoir computing: A review, Neural Networks (2019).
[6] P. Antonik, A. Smerieri, F. Duport, M. Haelterman, and S. Massar, Fpga implementation of reservoir computing with online learning, in 24th Belgian-Dutch Conference on Machine Learning (2015).
[7] C. Donahue, C. Merkel, Q. Saleh, L. Dolgovs, Y. K. Ooi, D. Kudithipudi, and B. Wysocki, Design and analysis of neuronomorphic echo state networks with limited precision synapses, in 2015 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA) (IEEE, 2015) pp. 1–6.
[8] M. Dale, J. F. Miller, S. Stepney, and M. A. Trefzer, Evolving carbon nanotube reservoir computers, in International Conference on Unconventional Computation and Natural Computation (Springer, 2016) pp. 49–61.
[9] C. Fernando and S. Sojatka, Pattern recognition in a bucket, in European conference on artificial life (Springer, 2003) pp. 588–597.
[10] S. Ghosh, A. Opala, M. Matuszewski, T. Paterek, and T. C. Liew, Quantum reservoir processing, npj Quantum Information 5, 35 (2019).
[11] J. Moon, W. Ma, J. H. Shin, F. Cai, C. Du, S. H. Lee, and W. D. Lu, Temporal data classification and forecasting using a memristor-based reservoir computing system, Nature Electronics 2, 480 (2019).
[12] G. Van der Sande, D. Brunner, and M. C. Soriano, Advances in photonic reservoir computing, Nanophotonics 6, 561 (2017).
[13] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckartment, S. Massar, J. Dambere, B. Schrauwen, C. R. Mirasso, and I. Fischer, Information processing using a single dynamical node as complex system, Nature communications 2, 468 (2011).
[14] Y. Paquot, J. Dambere, B. Schrauwen, M. Haelterman, and S. Massar, Reservoir computing: a photonic neural network for information processing, in Nonlinear Optics and Applications IV, Vol. 7728 (International Society for Optics and Photonics, 2010) p. 77280B.
[15] Y. Paquot, F. Duport, A. Smerieri, J. Dambere, B. Schrauwen, M. Haelterman, and S. Massar, Optoelectronic reservoir computing, Scientific reports 2, 287 (2012).
[16] L. Larger, M. C. Soriano, D. Brunner, L. Appeltant, J. M. Gutiérrez, L. Pesquera, C. R. Mirasso, and I. Fischer, Photonic information processing beyond turing: an optoelectronic implementation of reservoir computing, Optics express 20, 3241 (2012).
[17] D. Brunner, B. Penkovsky, B. A. Marquez, M. Jacquot, I. Fischer, and L. Larger, Tutorial: Photonic neural networks in delay systems, Journal of Applied Physics 124, 152004 (2018).
[18] J. D. Hart, L. Larger, T. E. Murphy, and R. Roy, Delayed dynamical systems: networks, chimeras and reservoir computing, Philosophical Transactions of the Royal Society A 377, 20180123 (2019).
[19] F. Duport, A. Smerieri, A. Akrout, M. Haelterman, and S. Massar, Fully analogue photonic reservoir computer, Scientific reports 6, 22381 (2016).
[20] F. Duport, A. Smerieri, A. Akrout, M. Haelterman, and S. Massar, Virtualization of a photonic reservoir computer, Journal of lightweight technology 34, 2085 (2016).
[21] R. Martinenghi, S. Rybalko, M. Jacquot, Y. K. Chembo, and L. Larger, Photonic nonlinear transient computing with multiple-delay wavelength dynamics, Physical review letters 108, 244101 (2012).
[22] S. Ortín, M. C. Soriano, L. Pesquera, D. Brunner, D. San-Martín, I. Fischer, C. Mirasso, and J. Gutiérrez, A unified framework for reservoir computing and extreme learning machines based on a single time-delayed neuron, Scientific reports 5, 14945 (2015).
[23] Q. Vinckier, F. Duport, A. Smerieri, K. Vandoorne, P. Bienstman, M. Haelterman, and S. Massar, High-performance photonic reservoir computer based on a coherently driven passive cavity, Optica 2, 438 (2015).
[24] B. Schneider, J. Dambere, and P. Bienstman, Using digital masks to enhance the bandwidth tolerance and improve the performance of on-chip reservoir computing systems, IEEE transactions on neural networks and learning systems 27, 2748 (2015).
[25] D. Brunner, M. C. Soriano, C. R. Mirasso, and I. Fischer, Parallel photonic information processing at gigabyte per second data rates using transient states, Nature communications 4, 1364 (2013).
[26] L. Larger, A. Baylón-Fuentes, R. Martinenghi, V. S. Udaltsov, Y. K. Chembo, and M. Jacquot, High-speed photonic reservoir computing using a time-delay-based architecture: Million words per second classification, Physical Review X 7, 011015 (2017).
[27] X. X. Guo, S. Y. Xiang, Y. H. Zhang, L. Lin, A. J. Wen, and Y. Hao, Four-channels reservoir computing based on polarization dynamics in mutually coupled vesels system, Optics express 27, 23293 (2019).
[28] Y.-S. Hou, G.-Q. Xia, E. Jayaprasath, D.-Z. Yue, W.-Y. Yang, and Z.-M. Wu, Prediction and classification performance of reservoir computing system using mutually delay-coupled semiconductor lasers, Optics Communications.
S. Popoff, G. Lerosey, R. Carminati, M. Fink, A. Boc-

cara, and S. Gigan, Measuring the transmission matrix in optics: an approach to the study and control of light propagation in disordered media, Physical review letters **104**, 100601 (2010).

N. Bertschinger and T. Natschläger, Real-time computation at the edge of chaos in recurrent neural networks, Neural computation **16**, 1413 (2004).

J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott, Using machine learning to replicate chaotic attractors and calculate lyapunov exponents from data, Chaos: An Interdisciplinary Journal of Nonlinear Science **27**, 121102 (2017).

R. Davidchack, Kuramoto-sivashinsky simulations, in *Chaos: Classical and Quantum* edited by P. Cvitanović, R. Artuso, R. Mainieri, G. Tanner, and G. Vattay (Niels Bohr Institute, 2012) 14th ed., Chap. 29.

F. Schürmann, K. Meier, and J. Schennec, Edge of chaos computation in mixed-mode vlsi-a hard liquid, in *Advances in neural information processing systems* (2005) pp. 1201–1208.

J. Isensee, G. Datseris, and U. Parlitz, Predicting spatio-temporal time series using dimension reduced local states, arXiv preprint arXiv:1904.06089 (2019).

Dell Precision 7920 Workstation Desktop Tower with 2x Intel Xeon Gold 5120 (14 cores, 2.2-3.7 GHz Turbo, 19 Mb cache), 64 Gb 2666 MHz DDR4, 2 SSD M.2 2 Tb Pcie NVMe.

M. Jacquot, L. Larger, and D. Brunner, Diffractive coupling for photonic networks: how big can we go?, IEEE Journal of Selected Topics in Quantum Electronics **26**, 1 (2019).

R. Ohana, J. Wacker, J. Dong, S. Marmin, F. Krzakala, M. Filippone, and L. Daudet, Kernel computations from large-scale random features obtained by optical processing units, arXiv preprint arXiv:1910.09880 (2019).

T. W. Hughes, I. A. Williamson, M. Minkov, and S. Fan, Wave physics as an analog recurrent neural network, arXiv preprint arXiv:1904.12831 (2019).

G. Marcucci, D. Pierangeli, and C. Conti, Theory of neuromorphic computing by waves: machine learning by rogue waves, dispersive shocks, and solitons, arXiv preprint arXiv:1912.07044 (2019).

Y. Zuo, B. Li, Y. Zhao, Y. Jiang, Y.-C. Chen, P. Chen, G.-B. Jo, J. Liu, and S. Du, All optical neural network with nonlinear activation functions, arXiv preprint arXiv:1904.10819 (2019).

T. Yan, J. Wu, T. Zhou, H. Xie, F. Xu, J. Fan, L. Fang, X. Lin, and Q. Dai, Fourier-space diffractive deep neural network, Physical review letters **123**, 023001 (2019).

X. Guo, T. D. Barrett, Z. M. Wang, and A. Lvovsky, End-to-end optical backpropagation for training neural networks, arXiv preprint arXiv:1912.12256 (2019).

J. W. Goodman,* Speckle phenomena in optics: theory and applications* (Roberts and Company Publishers, 2007).

P. Hohenberg and B. I. Shraiman, Chaotic behavior of an extended system, Physica D: Nonlinear Phenomena **37**, 109 (1989).

R. A. Edson, J. E. Bunder, T. W. Mattner, and A. J. Roberts, Lyapunov exponents of the kuramoto-sivashinsky pde, The ANZIAM Journal **61**, 270 (2019).