Dynamic effects on reservoir computing with a Hopf oscillator

Md Raf E Ul Shougat, XiaoFu Li, and Edmon Perkins
Department of Mechanical & Aerospace Engineering, LAB2701: Nonlinear Dynamics Laboratory, North Carolina State University, Raleigh, North Carolina 27695, USA

(Received 12 December 2021; accepted 6 April 2022; published 25 April 2022)

Limit cycle oscillators have the potential to be resourced as reservoir computers due to their rich dynamics. Here, a Hopf oscillator is used as a physical reservoir computer by discarding the delay line and time-multiplexing procedure. A parametric study is used to uncover computational limits imposed by the dynamics of the oscillator using parity and chaotic time-series prediction benchmark tasks. Resonance, frequency ratios from the Farey sequence, and Arnold tongues were found to strongly affect the computation ability of the reservoir. These results provide insights into fabricating physical reservoir computers from limit cycle systems.

DOI: 10.1103/PhysRevE.105.044212

I. INTRODUCTION

Reservoir computing (RC) is an unconventional computation technique, which utilizes the physics of a nonlinear dynamical system for computation [1–8]. Emerging from *echo state networks* [9] and *liquid state machines* [10], the RC scheme differs from the Turing machine principle since the computation performed by RCs do not rely on static memory. The computation is obtained by mapping the transient dynamics of the nonlinear physical system to a higher dimensional space. Some of the popular applications of RC include logical operations [11–15], spoken and handwritten digit recognition [8,16,17], wireless communications [2], complex and chaotic time-series predictions [2,8,18–21], long-term chaotic time-series prediction [22,23], image recognition [24], and morphological computation [25,26]. Due to the echo state network structure, many physical systems have been used as reservoirs, which are commonly known as *physical reservoir computers* (PRCs). Some of the classical PRCs include an array of Duffing oscillators [11,27,28], a limit cycle-based Hopf oscillator [29], soft robotic bodies [25,30–33], tensegrity structures [26,34], and origami structures [35].

Besides systems from classical physics, quantum physical systems have also been used as RCs to perform tasks from both classical and quantum domains. The naturally disordered quantum dynamics of an ensemble system was utilized to emulate nonlinear time series, including a chaotic system [36]. A Kerr nonlinear oscillator was used in sine wave phase estimation using its complex amplitudes as computational nodes [37]. A nuclear-magnetic-resonance spin-ensemble system was used for a nonlinear dynamics emulation task by implementing a spatial multiplexing approach to increase computational power [38]. Dissipative quantum dynamics was used to build a quantum reservoir computer for nonlinear temporal tasks [39]. Statistical physics has played an important role in the theoretical development of neural networks, which formed a connection between information processing and physics [40,41].

Popularly, delayed dynamical systems have been used as reservoirs from a single nonlinear node [42,43]. Coupled delay systems have also been used in computing by making deep neural networks [6] and signal processors [44]. A simpler implementation can also be achieved by excluding the delay or feedback line [45]. Here, a reservoir computer is built by implementing a two-state Hopf oscillator. The Hopf oscillator reservoir computer was previously studied, and it was shown that it could successfully complete several benchmark tasks [29]. The Hopf oscillator has an inherent capability of storing and learning information due to the presence of stable limit cycles, which also makes it suitable for building adaptive oscillators [46,47]. Conventionally, a binary mask is used in a time-multiplexing procedure to create virtual nodes for computation [8,13,48]. Besides this, noise can also be used as a mask [49]. Previously, an eigenvalue analysis was linked with the nonresonant condition to design a reservoir computer operating near the stable equilibrium [50]. However, the focus of this article is different since the Hopf reservoir is a limit cycle-based reservoir. So the analysis from Ref. [50] would not be applicable here. It is also noted that the popular notion of the "edge of chaos" was not used in this article to optimize the reservoir performance. It is supported in many literature findings that the edge of chaos is not a necessary condition to achieve good computational ability for a reservoir computer [31,51]. Hence, being distant from a chaotic region and tuning a set of network parameters can also be a route to construct a reservoir computer with good performance.

Microwave-based magnetic forced synchronization was implemented in a spintronic oscillator to increase the reservoir computing performance in Ref. [52]. The spin dynamics of a magnetic tunnel junction was also used to build a reservoir system [53]. Additionally, a nanoscale spintronic oscillator was optimized as a reservoir, based on the magnetization dynamics [54]. These spintronic oscillators may further be optimized by utilizing the relationship between the period of input (pseudofrequency) and the forcing as described in the current paper.

Here, a driven Hopf oscillator is studied as a reservoir computer with both masking function used in Ref. [29] and the commonly used delay line excluded; the masking function and delay lines were discarded to focus on the dynamics of the oscillator on computation. Resonance phenomena, Arnold tongues, and the Farey sequence all contribute to the performance of the Hopf oscillator as a reservoir computer. *Arnold tongues* refer to a phase-locked or synchronized region in the parameter space, which has a strong effect on this Hopf oscillator reservoir. Parity and chaotic laser time-series benchmarks are used to perform the parametric study of this Hopf oscillator computer. This oscillator was experimentally realized as an analog electrical circuit to study the information processing capability of the reservoir. A modified version of Shannon's information rate is used as the performance metric for the parity tasks [55].

II. DESCRIPTION OF THE HOPF RESERVOIR COMPUTER

The nonlinear system is perturbed by an input signal, which carries the information to be processed. The input $u(t)$ is embedded into the reservoir dynamics using the single nonlinear node as follows:

$$\begin{aligned} u(t) &= r(z) \quad \text{for } (n-1)T_p \leqslant t < (n)T_p, \\ f(t) &= 1 + u(t). \end{aligned} \tag{1}$$

Here, $r(z)$ is a discrete signal, which encodes logical values sequentially. This discrete signal is then mapped to a continuous function as described by Eq. (1), where $n, z \in \mathbf{Z}^+$. Since $r(z)$ is a random sequence of logical statements, $u(t)$ is a random square wave with a pseudoperiod T_p and a pseudofrequency $\omega_p = \frac{2\pi}{T_p}$.

The Hopf reservoir computer is described by the following equations of motion:

$$\begin{aligned} \dot{x} &= [\mu f(t) - (x^2 + y^2)]x - \omega_0 y + A f(t)\sin(\Omega t + \phi), \\ \dot{y} &= [\mu f(t) - (x^2 + y^2)]y + \omega_0 x. \end{aligned} \tag{2}$$

This system is a two-state forced Hopf oscillator, where x and y are the states, Ω is the harmonic forcing frequency, ω_0 is the resonance constant, and μ is a parameter controlling the limit cycle radius [46,47,56].

The governing equation of the Hopf RC in Eq. (2) is numerically integrated, and the x state is then scaled by subtracting the mean and dividing by the standard deviation. Next, dividing each pseudoperiod equally, N virtual nodes are collected from each pseudoperiod T_p. The nodal states are then nonlinearly scaled using a nonlinear activation function $\tanh^{-1}x$. Some 80% of the scaled nodal states are used for the training process, and the remaining 20% are used for testing the RC's performance. These virtual nodes, which are extracted by down-sampling the time histories, are similar to the nodes found in a delay-based reservoir. Since there is only one real node, which is the oscillator itself, the other nodes are called *virtual* in keeping with the terminology of delay-based reservoirs. However, the Hopf RC studied in this paper does not include any delay lines nor masking functions (time multiplexing), which simplifies the system.

The reservoir computer is trained using ridge regression with Tikhonov regularization as shown in Eq. (3),

$$\begin{aligned} w &= ML^T(LL^T + \lambda I)^{-1}, \\ o(k) &= \sum_{i=1}^{N} w_i X_i(k). \end{aligned} \tag{3}$$

Here, M is the target vector, which the reservoir should match. X is the scaled nodal states. L is the matrix containing the nodal states of the reservoir. λ, which is set to 10^{-1}, is the regularization parameter to avoid overfitting. I is the identity matrix. N is the number of nodes. w is the weight vector, which is found from the training procedure. $o(k)$ is the reservoir's prediction, where $k \in \mathbf{Z}^+$.

The δ delayed nth order parity function P_n is defined by the following equation [13]:

$$P_{n,\delta}(t) = \prod_{i=0}^{n-1} u[t - (i+\delta)T_p], \tag{4}$$

where $\delta \in \mathbf{Z}^+$ is the delay. For the tested parity tasks here, $\delta = 0$ is used. For $n = 2$, Eq. (4) is the second order parity, which is the exclusive or (XOR) task $\veebar$. Since this paper only deals with parity benchmarks that are logical tasks, the input $u(t) = \{-1, +1\}$ is chosen randomly for each pseudoperiod. Hence, the final prediction of the reservoir is also binarized, making a high $(+1)$ or low (-1) bit. Shannon's information rate is used to evaluate the reservoir's performance [11]. The logical bits are used to calculate the information metric R as follows:

$$R = H(x) - H_y(x). \tag{5}$$

Here, $H(x)$ is the *Shannon entropy*, which is a measure of the encoded information in a signal; it can be defined as

$$H(x) = -\sum_i p_i \log_2(p_i). \tag{6}$$

p_i is the probability of getting a particular bit i. $H_y(x)$ is the *conditional entropy*, which measures the probability of getting an incorrect bit in the target signal,

$$H_y(x) = -\sum_{i,j} p(i,j)\log_2[p_i(j)]. \tag{7}$$

Here, $p_i(j) = p(j|i) = \frac{p(i,j)}{\sum_j p(i,j)}$. $p(i,j)$ is the *joint probability distribution* of the two variables, i and j. i and j can be valued as 1 or -1 for a logical task. i is associated with the target signal, whereas j is the associated bit value from the prediction signal of the RC. It should be noted that the maximum value of R is 1.0 for these parity tasks. Using Eq. (2), the Hopf RC was fabricated as an analog circuit [29,46,47]. The circuit was built using TL082 operational amplifiers and AD633 multipliers in standard integrator network configurations. National Instrument cDAQ-9174 was used as the data acquisition device.

III. PARAMETRIC STUDIES

In this section, the dynamic limits of the Hopf reservoir computer will be explored by studying the effects of different

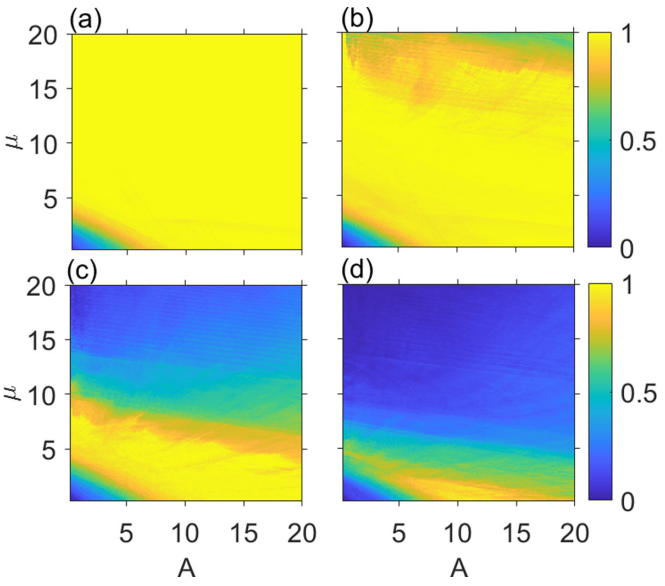

FIG. 1. Parametric study of the Hopf RC's limit cycle radius constant μ and harmonic forcing amplitude A, based on parity tasks. $\omega_p = 20\pi$ ($T_p = 0.1$ s), $\Omega = \omega_0 = 40\pi$ rad/s, $N = 1000$ nodes, $\phi = \pi/3$ rad, and the simulation time is $4000T_p$ s. (a) Second order parity, (b) third order parity, (c) fourth order parity, and (d) fifth order parity. The color bar denotes the information metric R.

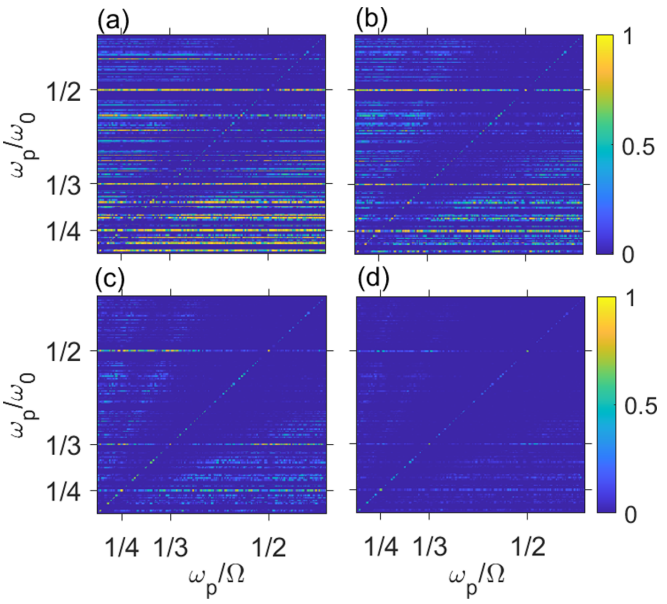

FIG. 2. Fine resolution parametric study of the Hopf RC's resonance constant ω_0 and harmonic forcing frequency Ω, based on parity tasks. $\omega_p = 20\pi$ ($T_p = 0.1$ s), $\mu = 5$, $A = 0.5$, $N = 1000$ nodes, $\phi = \pi/3$ rad, and the simulation time is $4000T_p$ s. (a) Second order parity, (b) third order parity, (c) fourth order parity, and (d) fifth order parity. The color bar denotes the information metric R.

parameters on its information processing capability. The performance is quantified using the information rate for various parameter combinations and orders of the parity tasks. The results render a deeper understanding of the interplay of the oscillator's dynamics and its computational ability as a reservoir computer. Some of these results can be used as guiding principles for tuning other virtual node-based reservoir computers, such as those found in Refs. [8,13,48].

The input signal to the Hopf RC is embedded into the oscillator through the limit cycle radius $\mu f(t)$ and the harmonic forcing amplitude $Af(t)$. Hence, tuning the parameters μ and A potentially controls the amount of information being sent into the oscillator. A parametric sweep of this two-parameter space is presented in Fig. 1. It is observed that a small limit cycle with a low forcing amplitude is ineffective for computation as expected. However, a band is observed for higher order tasks, which is unintuitive. This relationship could be used to maximize the computational ability of the oscillator. Previously, it was reported that the information processing capacity can be changed by simply altering the input magnitude of a spintronic reservoir [57]. The input magnitude was varied to get a different limit cycle response in the spintronic oscillator [57]. Similarly, the current Hopf RC varies the input magnitude by varying A to optimize the RC performance.

The external forcing frequency has been found to be important in determining a reservoir computer's performance [11,13,27]. Tuning the forcing frequency of an RC created from an array of Duffing oscillators, Arnold tongue-like structures and topological mixing were observed [11]. In a similar manner, a parametric study is performed using the resonance constant ω_0 and harmonic forcing frequency Ω.

A resonance condition of the Hopf oscillator is achieved when the resonance constant and harmonic forcing frequency are equal ($\omega_0 = \Omega$). In Figs. 2(a)–2(d), a band along a 45° angle in each of the ω_0-Ω parametric plots corresponds to this resonance condition. However, if the resonance condition ($\omega_0 = \Omega$) is achieved, the reservoir's performance jumps suddenly from poor performance to successful computation. Hence, the resonance phenomenon is a necessary condition for the Hopf oscillator to function as an effective RC.

Another matching condition is met when ω_0 is an integer multiple of the pseudoperiod, ω_p ($\omega_0 = z\omega_p$, where $z \in \mathbb{Z}$). This can be observed for all the parity tasks presented in Figs. 2 and 3. The ω_0-Ω parametric space is also explored experimentally, which is presented in Fig. 3 by building an analog circuit that can be modeled with Eq. (2). Due to experimental limitations, a comparatively coarse parametric space is shown in Fig. 3. The experiments show a similar trend near the resonance and at the matching conditions. From Fig. 3, the best performance for a numerical reservoir is found for the resonance condition when $\frac{\omega_p}{\Omega} = \frac{\omega_p}{\omega_0} = 1/3$; the best performance for the experimental reservoir is found when $\frac{\omega_p}{\omega_0} = 0.333$ and $\frac{\omega_p}{\Omega} = 0.344$. Deviation between the experiment and the simulation are likely explained by the precision of the circuit elements and nonlinear effects of the circuit (e.g., parasitic effects).

Next, the effects of the frequency ratio $\frac{\omega_p}{\omega_0}$ and the harmonic forcing amplitude A on the computational ability of the Hopf RC are shown in Fig. 4. The $\frac{\omega_p}{\omega_0}$-$A$ parametric space is studied using the second order parity and chaotic laser intensity prediction tasks, whereas keeping the system parameters the same. For the chaotic time-series benchmark [29], the task for the RC was to predict one step ahead based on the previous steps. The RMSE was used as the performance metric for this task. To verify that frequency ratios from the Farey sequence

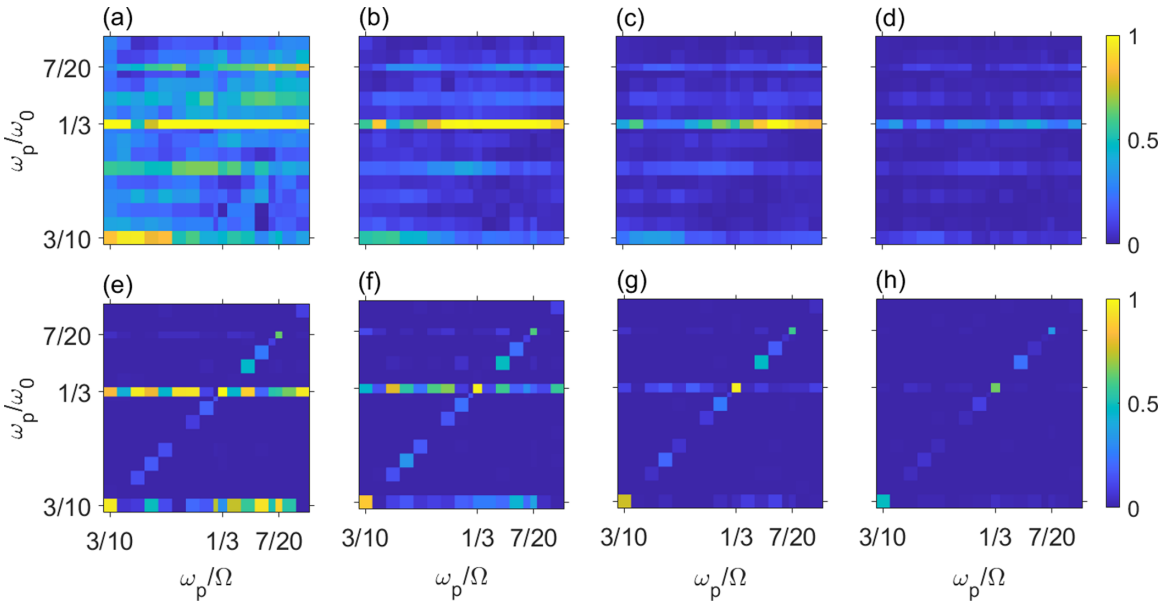

FIG. 3. Coarse resolution experimental and numerical parametric qualitative study of the Hopf RC's resonance constantω_0 and harmonic forcing frequency Ω, based on parity tasks. $\omega_p = 20\pi$ ($T_p = 0.1$ s), $\mu = 5$, $A = 0.5$, $N = 1000$ nodes, $\phi = \pi/3$ rad, and the length of time is $4000T_p$ s. Top: analog circuit experiment; (a) second order parity; (b) third order parity; (c) fourth order parity; and (d) fifth order parity. The color bar denotes information metric R. Bottom: numerical simulation; (e) second order parity; (f) third order parity; (g) fourth order parity; and (h) fifth order parity. The color bar denotes information metric$R$.

are important to other tasks, a chaotic time-series benchmark is also used to compare the performance with a nonbinary task. The harmonic frequency Ω is set to 40π such that the Hopf oscillator experiences resonance when $\omega_p/\omega_0 = \frac{1}{2}$. In Fig. 4, it is observed that the Hopf reservoir can have high computational ability even when the resonance condition is not met.

From Fig. 4, it is observed that the Hopf RC has a high computational ability when $\frac{\omega_p}{\omega_0}$ is a number from the Farey sequence from number theory [58]. Using a 30th order Farey sequence, it is observed that the reservoir's computational ability is strongly influenced by the frequency ratio $\frac{\omega_p}{\omega_0}$, matching a number from the Farey sequence. To observe the correlation between the Farey ratio and the RC's performance

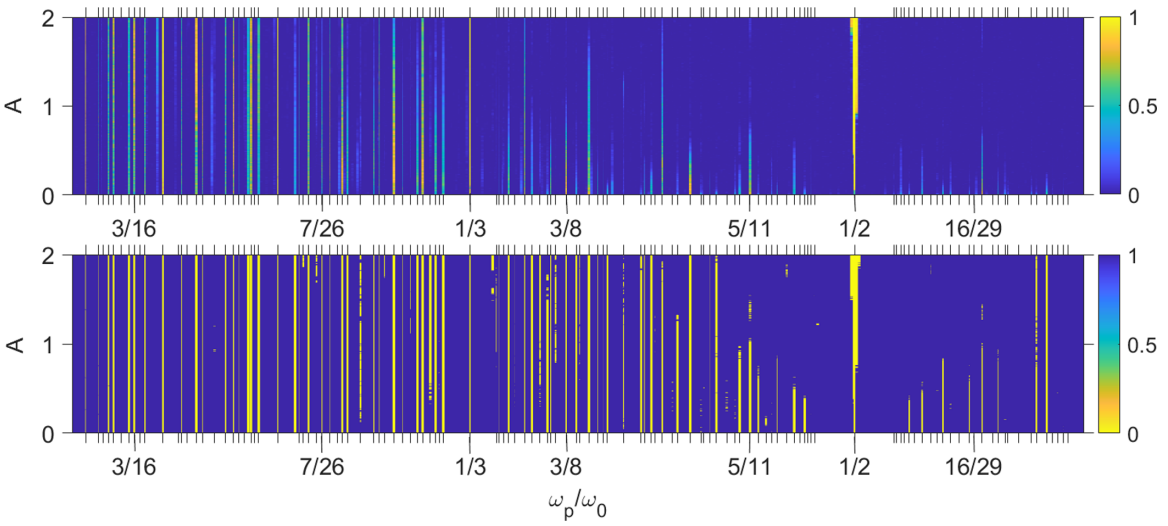

FIG. 4. The ratio of $\frac{\omega_p}{\omega_0}$ is varied along the horizontal axis with values from the Farey sequence (30th order) marked with tick marks, and the forcing amplitude A is varied along the vertical axis. Top: second order parity task where the color bar denotes the information metric. Bottom: Chaotic laser time-series prediction where the color bar denotes performance based on the root mean square error (RMSE), which has been binarized to be high (logical 1) if RMSE > 0.3 and low (logical 0) for RMSE $\leqslant 0.3$. For these tasks, $\mu = 5$, $\Omega = 40\pi$ rad/s, $N = 1000$ nodes, $\phi = 0$ rad, $\omega_p = 20\pi$ rad/s ($T_p = 0.1$ s), and the simulation time is $3000T_p$ s. In both cases, the computational ability of the Hopf RC is strongly predicted by $\frac{\omega_p}{\omega_0}$ aligning with a Farey sequence number.

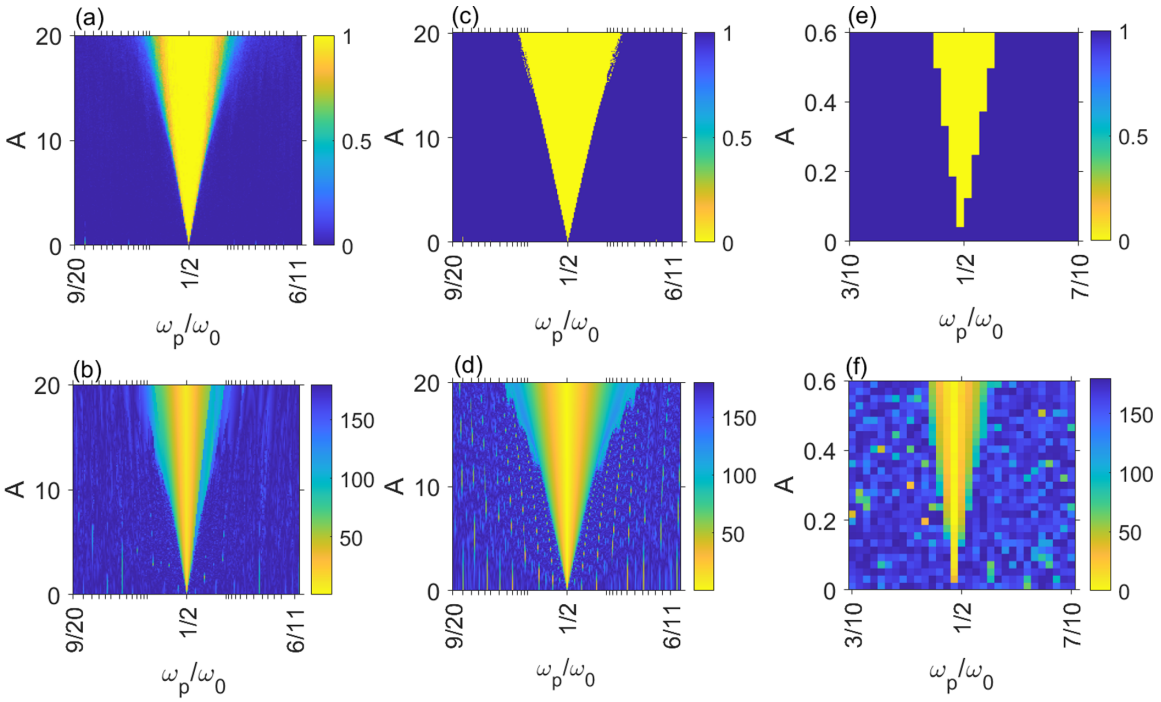

FIG. 5. Synchronization plays an important role in the RC's computational ability, which is observed as an Arnold tongue around the resonance region where $\frac{\omega_p}{\omega_0} = \frac{1}{2}$. (a) Simulation showing an Arnold tongue region for the second order parity task for the RC. The color bar denotes the information metric R. (b) Simulation showing an Arnold tongue region in the second order parity task for the Hopf oscillator. The color bar denotes the phase lag (degrees) between the Hopf oscillator response$x$ and the external forcing ($\sin\ \Omega t + \phi$). (c) Simulation showing an Arnold tongue region in the chaotic time-series task for the RC. The color bar denotes the RMSE. (d) Simulation showing an Arnold tongue region in the chaotic time-series task for the Hopf oscillator. The color bar denotes the phase lag (○). (e) Experiment showing an Arnold tongue region in the chaotic time-series task for the RC. The color bar denotes the RMSE. (f) Experiment showing an Arnold tongue region in the chaotic time-series task for the Hopf oscillator. The color bar denotes the phase lag (○). In (c) and (e), RMSE has been binarized to be high (logical 1) if RMSE $>$ 0.1 and low (logical 0) for RMSE $\leqslant$ 0.1. For these tasks, $\omega_p = 20\pi$ rad/s ($T_p = 0.1$ s), $\mu = 5$, $A = 0.5$, $N = 1000$ nodes, $\phi = 0$ rad.

for the chaotic time-series task, the RMSE was binarized to be high (logical 1) if RMSE $>$ 0.3 and low (logical 0) for RMSE $\leqslant$ 0.3. The Farey sequence is found in many natural phenomena, such as in the auditory system [59], the resonance diagrams of accelerators [60], mode locking in quantum accelerators [61], and cardiac dysrhythmias [62]. However, this is when the Farey sequence has been reported in the context of reservoir computing.

Figures 5(b), 5(d) and 5(f) show the phase difference between the Hopf oscillator's x state and the harmonic forcing $\sin(\Omega t + \phi)$ for parity and chaotic time-series prediction tasks. When $\frac{\omega_p}{\omega_0}$ is near $\frac{1}{2}$, a resonance relationship forms an Arnold tongue in the $\frac{\omega_p}{\omega_0}$-$A$ space. These tongues are formed both in the reservoir's performance metric space and in the oscillator's phase deviation space. It is noted that the phase difference is not constant throughout the time history. Hence, the maximum phase difference is taken into consideration by binning the response, taking fast Fourier transforms, and then plotting the maximum phase difference. Arnold tonguelike regions were also found in a reservoir computer composed of a Duffing oscillator array [11]. The Hopf RC performs best when this resonance condition holds. In Figs. 5(a), 5(c) and 5(e), an Arnold tongue is observed in the performance space of the reservoir when the Hopf RC operates near the resonance frequency. Here, Figs. 5(e) and 5(f) depict Arnold tongues for the experimental Hopf RC, and Figs. 5(a)–5(d) are found from numerical simulations. Figure 6 shows the time history of the oscillator's response when it is locked with the forcing and when it is not phase locked. The tongue region could be particularly important in experimental design. For this tongue region, there is a range of frequency ratios centering on the resonance frequency, which can result in better computation. Hence, this is the only region found where the reservoir has some tolerance to mistuning. This implies that the synchronization in the Arnold tongue region causes robust computing. There is also a comparatively broader range of amplitudes and frequency ratios available for which the resonance constant can be tuned such that the Hopf RC has high computational ability whereas staying inside the Arnold tongue. It is noted that the studies presented in this paper are performed considering no masking in the system. The presence of robust computational ability despite the absence of a masking function suggests that a single nonlinear node based reservoir can also be reliably constructed discarding the conventional periodic or nonperiodic mask.

To understand the effect of resonance on computation, the memory capacity of the Hopf oscillator reservoir is calculated. For a δ delayed nth order parity function given in Eq. (4),

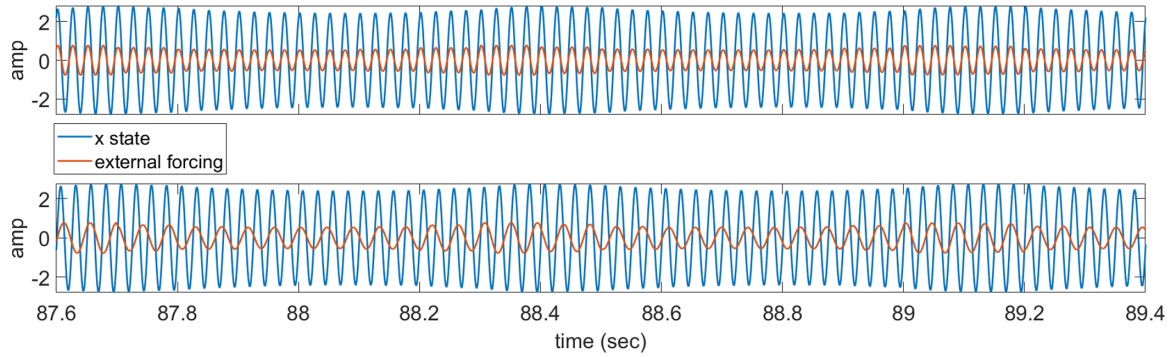

FIG. 6. The time series of the x state of the Hopf RC is shown for a portion of the chaotic time-series task. Top: The Hopf RC is at resonance ($\omega_0 = \Omega$), and the oscillations are locked with the external forcing. Bottom: The Hopf RC is not at resonance ($\omega_0 = 1.7391 * \Omega$), and the oscillations are not locked with external forcing. Here, $\omega_0 = 80\pi$, $\omega_p = 20\pi$ rad/s ($T_p = 0.1$ s), $\mu = 5$, $A = 0.5$, $N = 1000$ nodes, $\phi = \pi/3$ rad.

the memory capacity of a system can be calculated as follows [13]:

$$MI_{n,\delta} = p_{n,\delta}\log_2(2p_{n,\delta}) + (1 - p_{n,\delta})\log_2[2(1 - p_{n,\delta})],$$
$$MC_n = \sum_{\delta=0}^{\infty} MI_{n,\delta}. \tag{8}$$

The memory capacity (bits) is plotted against ω_p/ω_0 in Fig. 7, which shows that the reservoir possesses highest memory at the resonance condition when $\frac{\omega_p}{\omega_0} = \frac{1}{2}$. Hence, with sufficient nonlinearity being present at this condition, the Hopf oscillator should conduct better computation than at other parametric combinations. This explains the superior computing performance of the resonance condition as seen in Figs. 2–5.

The effect of the nonlinear activation function ($\tanh^{-1} X$) on the performance of the Hopf oscillator RC is also studied, and the results are presented in Fig. 8. It is found that in the absence of nonlinear activation, the Hopf oscillator RC shows similar performance for lower order tasks (e.g., second and third order parity tasks). However, the nonlinear activation function becomes important in performing higher order tasks. Hence, the base Hopf oscillator dynamics has some level of computing ability. Furthermore, a linear oscillator is also tested as a reservoir computer in the presence of a nonlinear activation function. In this case, the nonlinear activation function cannot make the linear oscillator act as a reservoir computer. This is similar to the effect of nonlinearity in the acoustic transformation for a digit recognition task [64].

The ESP is one of the basic properties found in a successful reservoir computing framework [9]. Limit cycle-based systems were not found to satisfy ESP requirements in Refs. [65,66]. However, the Hopf RC formulation described in this paper is different since the limit cycle radius keeps changing depending on the forcing used to encode the information. Additionally, the resonance phenomenon was taken into consideration to build the reservoir system. In the literature, generalized synchronization [67] or common signal induced synchronization [65,66,68] were used to verify the existence of ESP in a reservoir. ESP was empirically studied in Ref. [63] to measure the stability of input-driven reservoir

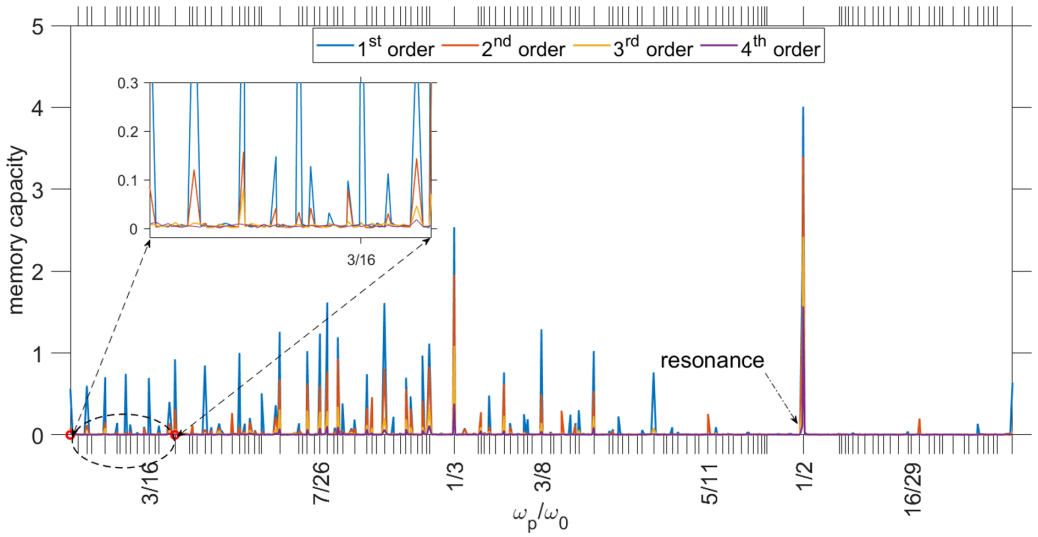

FIG. 7. Memory capacity calculations of the Hopf RC with δ delayed first, second, third, and fourth order parity tasks ($\delta = 0$ to 10). $\omega_p = 20\pi$ ($T_p = 0.1$ s), $\Omega = 40\pi$, $A = 0.5$, $\mu = 5$, $N = 1000$ nodes, $\phi = \pi/3$ rad, and the simulation time is $4000T_p$ s.

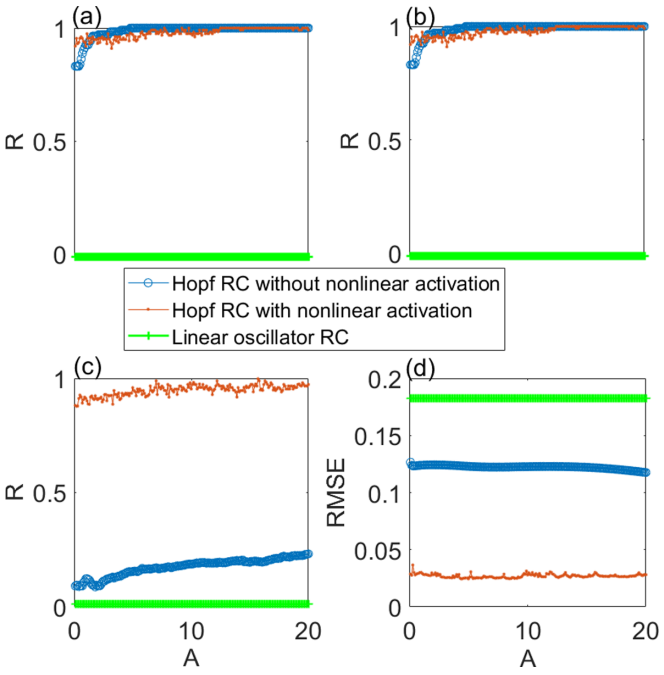

FIG. 8. Effects of nonlinear activation function on the Hopf RC's performance. $\omega_p = 20\pi$ ($T_p = 0.1$ s), $\omega_0 = 40\pi$, $\Omega = 40\pi$, $\mu = 5$, $N = 1000$ nodes, $\phi = \pi/3$ rad, and the simulation time is $4000T_p$ s. (a) Second order parity, (b) third order parity, (c) fourth order parity, and (d) chaotic time series

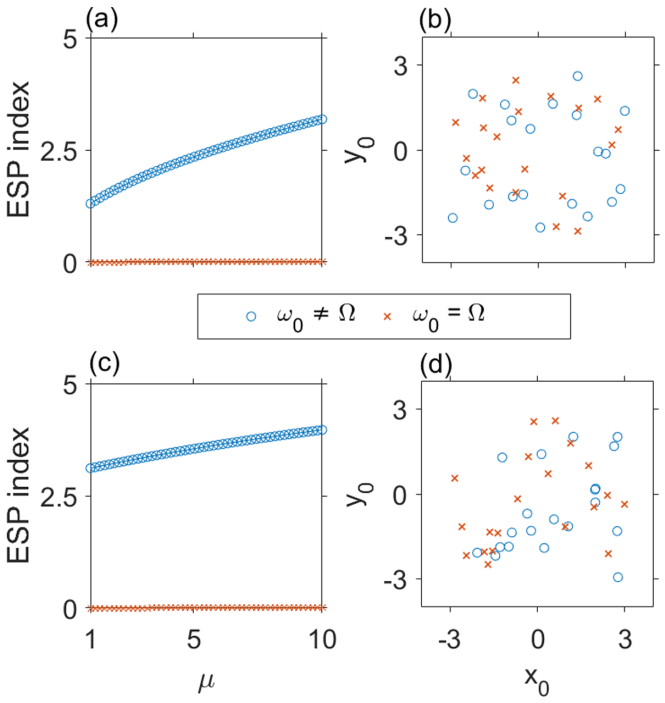

FIG. 9. The echo state property (ESP) index calculation [63] for the Hopf RC for the chaotic laser time-series task and the parity task. (a) ESP index for resonance and nonresonance conditions for the chaotic time-series task. (b) One set of random initial conditions (x_0 and y_0 chosen from $\{-3, 3\}$) for the chaotic time-series task. (c) ESP index for resonance and nonresonance conditions for the parity task. (d) One set of random initial conditions (x_0 and y_0 chosen from $\{-3, 3\}$) for the parity task. $\omega_p = 20\pi$ ($T_p = 0.1$ s), $\omega_0 = 80\pi$, $\phi = \pi/3$ rad, $A = 0.5$ and the simulation time is $3000T_p$ s. For the resonance condition $\Omega = \omega_0$ and for the nonresonance condition $\Omega = 1.7391\omega_0$.

dynamics. It is implied that a reservoir possesses the echo state property if the asymptotic trajectory of the reservoir state relies distinctively on the inputs and is independent of the initial conditions. Hence, to obtain the echo state property, the effect of the initial conditions on the reservoir dynamics should fade as the time progresses.

The echo state property of the Hopf reservoir is studied when the reservoir is encoded with the inputs of the chaotic laser time-series task and a parity task. Following the algorithm of estimating ESP of a reservoir from Ref. [63], an ESP index was calculated for the two benchmarks, which is averaged over 20 randomly generated initial conditions in the range of $\{-3, 3\}$. The deviation of the reservoir state trajectories for different initial conditions was calculated using the same input sequence for all of the different initial conditions. To calculate the deviation, initial conditions of $(x_0, y_0) = (0, 0)$ were used as the common trajectory whereas the other trajectory came from each of the different initial conditions, whereas discarding the initial transients. Finally, an average of the deviation is calculated to find the ESP index. If the ESP index goes to zero for a reservoir's dynamics, the reservoir is said to possess the echo state property. These results are given in Fig. 9 where it is observed that the reservoir has the echo state property for the resonance condition ($\omega_0 = \Omega$), whereas it does not possess this property when the resonance is not met ($\omega_0 \neq \Omega$). This can also explain why the resonance helps in computation. It is also important to note that a reservoir may still have information processing capability when the ESP is not met [65].

IV. CONCLUDING REMARKS

In this paper, the Hopf oscillator is constructed as a reservoir computer to gain insights into the relationship between the oscillator's dynamics and the RC's computational ability. This implementation of a Hopf reservoir computer offers a simpler design by discarding the popularly used delayed feedback line and masking function. An analog electrical circuit is used as a physical realization of the reservoir. The reservoir demonstrates high computational ability when the ratio of the pseudofrequency of the input ω_p and the natural frequency of the oscillator ω_0 are taken from the Farey sequence. Additionally, a resonance phenomenon happens when the harmonic forcing frequency and natural frequency of the oscillator are equal, which provides a favorable condition to construct the reservoir computer. Enhanced computational ability is achieved when the limit cycle radius is relatively small whereas the forcing amplitude is relatively large. An Arnold tongue structure is observed in the reservoir's information metric space near the resonance location, which is correlated with a Arnold tongue exhibited in the phase deviation space. The reservoir is also found to possess both maximum memory capacity and the echo state property when the resonance condition is met, which is indicative of better computing performance in principal. Finally, the results also

suggest that a reservoir computer can be constructed with only a single nonlinear node and neither a time-multiplexing process nor a delayed feedback. By harnessing some of the underlying dynamics of the system, limit cycle reservoir computers can be constructed that are both simple and robust.

ACKNOWLEDGMENTS

Partial support for this project from DARPA's Young Faculty Award was greatly appreciated. Research was sponsored by the Army Research Office and was accomplished under Grant No. W911NF-20-1-0336. The views and conclusions contained in this paper are those of the authors and do not necessarily reflect the official policies either expressed or implied of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. We acknowledge the computing resources provided on *Henry2*, a high-performance computing cluster operated by North Carolina State University; we thank Dr. A. Petersen for his assistance with parallel computing, which was provided through the Office of Information Technology HPC services at NC State University.

[1] K. Nakajima and I. Fischer, *Reservoir Computing* (Springer, Singapore, 2021).
[2] H. Jaeger and H. Haas, Science **304**, 78 (2004).
[3] M. Lukoševičius and H. Jaeger, Comput. Sci. Rev. **3**, 127 (2009).
[4] K. Nakajima, Jpn. J. Appl. Phys. **59**, 060501 (2020).
[5] K. Nakajima, H. Hauser, T. Li, and R. Pfeifer, Sci. Rep. **5**, 10487 (2015).
[6] B. Penkovsky, X. Porte, M. Jacquot, L. Larger, and D. Brunner, Phys. Rev. Lett. **123**, 054101 (2019).
[7] N. D. Haynes, M. C. Soriano, D. P. Rosin, I. Fischer, and D. J. Gauthier, Phys. Rev. E **91**, 020801(R) (2015).
[8] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer, Nat. Commun. **2**, 1 (2011).
[9] H. Jaeger, Bonn, Germany: German National Research Center for Information Technology GMD Technical Report No. 148, 2001, p. 13.
[10] T. Natschläger, W. Maass, and H. Markram, Special issue on Foundations of Information Processing of TELEMATIK **8**, 39 (2002).
[11] M. R. E. U. Shougat, X. Li, T. Mollik, and E. Perkins, J. Comput. Nonlinear Dyn. **16**, 081004 (2021).
[12] G. Marcucci, D. Pierangeli, and C. Conti, Phys. Rev. Lett. **125**, 093901 (2020).
[13] G. Dion, S. Mejaouri, and J. Sylvestre, J. Appl. Phys. **124**, 152132 (2018).
[14] F. Laporte, J. Dambre, and P. Bienstman, Sci. Rep. **11**, 1 (2021).
[15] M. R. E. U. Shougat, X. Li, T. Mollik, and E. Perkins, *International Design Engineering Technical Conferences & Computers and Information in Engineering Conference* (ASME, Virtual, 2021).
[16] R. Martinenghi, S. Rybalko, M. Jacquot, Y. K. Chembo, and L. Larger, Phys. Rev. Lett. **108**, 244101 (2012).
[17] C. Du, F. Cai, M. A. Zidan, W. Ma, S. H. Lee, and W. D. Lu, Nat. Commun. **8**, 1 (2017).
[18] M. Inubushi and K. Yoshimura, Sci. Rep. **7**, 1 (2017).
[19] R. Wang, E. Kalnay, and B. Balachandran, Nonlinear Dyn. **98**, 2903 (2019).
[20] M. Rafayelyan, J. Dong, Y. Tan, F. Krzakala, and S. Gigan, Phys. Rev. X **10**, 041037 (2020).
[21] A. Röhm, D. J. Gauthier, and I. Fischer, Chaos **31**, 103127 (2021).
[22] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, Phys. Rev. Lett. **120**, 024102 (2018).
[23] K. Srinivasan, N. Coble, J. Hamlin, T. Antonsen, E. Ott, and M. Girvan, arXiv:2108.12129.
[24] S. Borlenghi, M. Boman, and A. Delin, Phys. Rev. E **98**, 052101 (2018).
[25] G. Urbain, J. Degrave, B. Carette, J. Dambre, and F. Wyffels, Frontiers in neurorobotics **11**, 16 (2017).
[26] K. Caluwaerts, M. D'Haene, D. Verstraeten, and B. Schrauwen, Artificial life **19**, 35 (2013).
[27] J. C. Coulombe, M. C. York, and J. Sylvestre, PLoS One **12**, e0178663 (2017).
[28] T. Zheng, W. Yang, J. Sun, X. Xiong, Z. Li, and X. Zou, Sci. Rep. **11**, 1 (2021).
[29] M. R. E. U. Shougat, X. Li, T. Mollik, and E. Perkins, Sci. Rep. **11**, 1 (2021).
[30] H. Hauser, A. J. Ijspeert, R. M. Füchslin, R. Pfeifer, and W. Maass, Biol. Cybern. **105**, 355 (2011).
[31] K. Nakajima, H. Hauser, R. Kang, E. Guglielmino, D. G. Caldwell, and R. Pfeifer, in *2013 IEEE International Conference on Robotics and Automation* (IEEE, Piscataway, NJ, 2013) pp. 1504–1511.
[32] K. Nakajima, H. Hauser, R. Kang, E. Guglielmino, D. G. Caldwell, and R. Pfeifer, Front. Comput. Neurosci. **7**, 91 (2013).
[33] K. Nakajima, T. Li, H. Hauser, and R. Pfeifer, J. R. Soc., Interface **11**, 20140437 (2014).
[34] K. Caluwaerts, J. Despraz, A. Işçen, A. P. Sabelhaus, J. Bruce, B. Schrauwen, and V. SunSpiral, J. R. Soc., Interface **11**, 20140520 (2014).
[35] P. Bhovad and S. Li, Sci. Rep. **11**, 1 (2021).
[36] K. Fujii and K. Nakajima, Phys. Rev. Appl. **8**, 024030 (2017).
[37] L. C. G. Govia, G. J. Ribeill, G. E. Rowlands, H. K. Krovi, and T. A. Ohki, Phys. Rev. Res. **3**, 013077 (2021).
[38] K. Nakajima, K. Fujii, M. Negoro, K. Mitarai, and M. Kitagawa, Phys. Rev. Appl. **11**, 034021 (2019).
[39] J. Chen, H. I. Nurdin, and N. Yamamoto, Phys. Rev. Appl. **14**, 024065 (2020).
[40] E. Agliari, F. Alemanno, A. Barra, and G. De Marzo, Neural Networks **148**, 232 (2022).
[41] E. Agliari, A. Barra, P. Sollich, and L. Zdeborová, J. Phys. Math. Theor **53**, 500401 (2020).

[42] D. Pinna, G. Bourianoff, and K. Everschor-Sitte, Phys. Rev. Appl. **14**, 054020 (2020).
[43] F. Köster, S. Yanchuk, and K. Lüdge, arXiv:2108.12643.
[44] M. Hermans, P. Antonik, M. Haelterman, and S. Massar, Phys. Rev. Lett. **117**, 128301 (2016).
[45] D. Marković, N. Leroux, M. Riou, F. Abreu Araujo, J. Torrejon, D. Querlioz, A. Fukushima, S. Yuasa, J. Trastoy, P. Bortolotti *et al.*, Appl. Phys. Lett. **114**, 012409 (2019).
[46] X. Li, M. R. E. U. Shougat, S. Kennedy, C. Fendley, R. N. Dean, A. N. Beal, and E. Perkins, PLoS One **16**, e0249131 (2021).
[47] X. Li, M. R. E. U. Shougat, T. Mollik, A. N. Beal, R. N. Dean, and E. Perkins, J. Appl. Phys. **129**, 224901 (2021).
[48] L. Appeltant *et al.*, These de Doctorat, Vrije Universiteit Brussel/Universitat de les Illes Balears, 2012.
[49] J. Nakayama, K. Kanno, and A. Uchida, Opt. Express **24**, 8679 (2016).
[50] F. Köster, S. Yanchuk, and K. Lüdge, J. Phys. Photonics **3**, 024011 (2021).
[51] M. Mitchell, J. P. Crutchfield, and P. T. Hraber, *Santa Fe Institute Studiesin the Sciences of Complexity-Proceedings*, Vol. 19 (Addison-Wesley, Boston, 1994), pp. 497–497.
[52] S. Tsunegi, T. Taniguchi, K. Nakajima, S. Miwa, K. Yakushiji, A. Fukushima, S. Yuasa, and H. Kubota, Appl. Phys. Lett. **114**, 164101 (2019).
[53] T. Furuta, K. Fujii, K. Nakajima, S. Tsunegi, H. Kubota, Y. Suzuki, and S. Miwa, Phys. Rev. Appl. **10**, 034063 (2018).
[54] J. Torrejon, M. Riou, F. A. Araujo, S. Tsunegi, G. Khalsa, D. Querlioz, P. Bortolotti, V. Cros, K. Yakushiji, A. Fukushima *et al.*, Nature (London) **547**, 428 (2017).
[55] C. E. Shannon, Bell Syst. Tech. J. **27**, 379 (1948).
[56] A. H. Nayfeh and B. Balachandran, *Applied Nonlinear Dynamics: Analytical, Computational, and Experimental Methods* (Wiley, Hoboken, NJ, 2008).
[57] N. Akashi, T. Yamaguchi, S. Tsunegi, T. Taniguchi, M. Nishida, R. Sakurai, Y. Wakao, and K. Nakajima, Phys. Rev. Res. **2**, 043303 (2020).
[58] V. Sukhoy and A. Stoytchev, Sci. Rep. **11**, 1 (2021).
[59] Y. V. Ushakov, A. A. Dubkov, and B. Spagnolo, Phys. Rev. Lett. **107**, 108103 (2011).
[60] R. Tomás, Phys. Rev. Spec.l Top.-Accel. Beams **17**, 014001 (2014).
[61] A. Buchleitner, M. B. d'Arcy, S. Fishman, S. A. Gardiner, I. Guarneri, Z.-Y. Ma, L. Rebuzzini, and G. S. Summy, Phys. Rev. Lett. **96**, 164101 (2006).
[62] H.-X. Wang, R. de Paola, and W. I. Norwood, Phys. Rev. Lett. **70**, 3671 (1993).
[63] C. Gallicchio, arXiv:1811.10892.
[64] F. Abreu Araujo, M. Riou, J. Torrejon, S. Tsunegi, D. Querlioz, K. Yakushiji, A. Fukushima, H. Kubota, S. Yuasa, M. D. Stiles *et al.*, Sci. Rep. **10**, 1 (2020).
[65] T. Kubota, H. Takahashi, and K. Nakajima, Phys. Rev. Res. **3**, 043135 (2021).
[66] K. Goto, K. Nakajima, and H. Notsu, New J. Phys. **23**, 063051 (2021).
[67] Z. Lu, B. R. Hunt, and E. Ott, Chaos **28**, 061104 (2018).
[68] M. Inubushi, K. Yoshimura, Y. Ikeda, and Y. Nagasawa, in *Reservoir Computing* (Springer, Singapore, 2021), pp. 97–116.