Reconstructing bifurcation diagrams of chaotic circuits with reservoir computing

Haibo Luo,[1] Yao Du,[1] Huawei Fan,[2] Xuan Wang,[1] Jianzhong Guo,[1,*] and Xingang Wang[1,†]
[1]School of Physics and Information Technology, Shaanxi Normal University, Xi'an 710062, China
[2]School of Science, Xi'an University of Posts and Telecommunications, Xi'an 710121, China

(Received 12 September 2023; accepted 26 January 2024; published 13 February 2024)

Model-free reconstruction of bifurcation diagrams of Chua's circuits using the technique of parameter-aware reservoir computing is investigated. We demonstrate that (1) reservoir computer can be utilized as a noise filter to restore the system dynamics from noisy signals; (2) for a single Chua circuit, a machine trained by the noisy time series measured at several sampling states is capable of reconstructing the whole bifurcation diagram of the circuit with a high precision; and (3) for two coupled chaotic Chua circuits with mismatched parameters, the machine trained by the noisy time series measured at several coupling strengths is able to anticipate the variation of the synchronization degree of the coupled circuits with respect to the coupling strength over a wide range. Our studies verify the capability of the technique of parameter-aware reservoir computing in learning the dynamics of chaotic circuits from noisy signals, signifying the potential application of this technique in reconstructing the bifurcation diagram of real-world chaotic systems.

DOI: 10.1103/PhysRevE.109.024210

I. INTRODUCTION

In exploring chaotic systems, one of the central tasks is to characterize how the system dynamics varies with the system parameters, namely, finding the bifurcation diagram of the system dynamics [1,2]. Studying the bifurcation diagram is not only of theoretical interest because it reveals the route from regular behaviors to chaos but also of practical significance because it pinpoints the tipping points where a small change in the system parameters might result in a drastic change in the system dynamics [3,4]. The latter is of particular concern to modern society because accumulating evidence indicates that many real-world complex systems are already in the vicinity of their tipping points, e.g., the global climate [5,6], complex ecological systems [7,8], and financial markets [9,10]. When the exact equations governing the system dynamics are known, the bifurcation diagram can be constructed using the approach of model simulations. Yet in realistic situations the exact equations of the system dynamics are generally unknown, and what is available are only measured data. Different from model-based studies in which the signals are noise-free and the system parameters can be tuned arbitrarily according to the research request, signals measured from realistic systems are inevitably contaminated by noise. In addition, due to the cost of data acquisition and practical restrictions, it is infeasible to construct the bifurcation diagram of a realistic system with a fine scan of the system parameters over a wide range. These practical concerns make model-free reconstruction of the bifurcation diagram of realistic chaotic systems a challenging question of active research in the field of nonlinear science and complex systems [11–25].

To reconstruct the bifurcation diagram of chaotic systems based on measured data, one approach is to rebuild the model first, including inferring the terms contained in the dynamical equations and estimating the system parameters, and then reconstruct the bifurcation diagram through the approach of model simulations [26–28]. The advantage of this model-rebuilding approach is that the equations governing the system dynamics can be obtained explicitly, while the drawbacks are that the data should be of high quality (with weak noise) and some prior knowledge of the system dynamics should be available, e.g., the form of the nonlinear terms in the equations. An alternative approach to reconstructing the bifurcation diagram is to exploit machine learning techniques [15–25]. Owing to the superpower of regression analysis, machine learning techniques are able to infer from data not only the dynamics of chaotic systems but also the system parameters and therefore are capable of reconstructing bifurcation diagrams. Compared to the model-rebuilding approach, the advantages of the machine learning approach are that no prior knowledge of the system dynamics is required and the techniques can be applied to noisy signals in general, yet the disadvantages are that the system dynamics are unknown (i.e., the machines are working as "black boxes") and a large amount of data is normally required to train the machines.

Reservoir computing (RC) [29,30], a special technique based on recurrent neural networks in machine learning, was exploited recently for predicting chaos and reconstructing the bifurcation diagram of chaotic systems [20–25,31–36]. From the point of view of dynamical systems, a reservoir computer can be regarded as a complex network of coupled nonlinear units which, driven by the input signals, generates the outputs through a readout function [37]. Compared to other types of deep learning techniques such as convolutional neural networks, RC contains only a single hidden layer, namely, the reservoir. Except for the output matrix which is to be estimated from the data through a training process, the machine is fixed at construction, including the input matrix, the reservoir network, and the updating rules. Although structurally simple,

*guojz@snnu.edu.cn
†wangxg@snnu.edu.cn

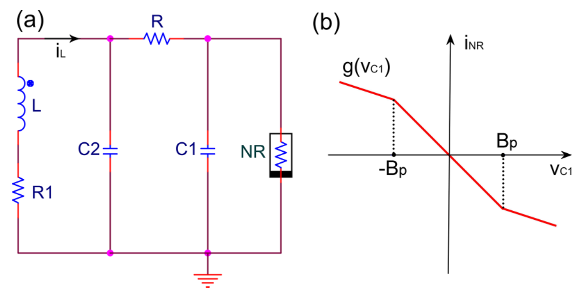

FIG. 1. (a) Schematic of Chua's circuit. NR denotes the nonlinear resistor. The linear resistor R plays the role of the bifurcation parameter, which is adjusted to generate different dynamics. (b) The piecewise-linear characteristic curve of the NR.

RC has shown great potential in many data-oriented applications [37], e.g., speech recognition, channel equalization, robot control, and chaos prediction. In particular, it has been shown that a properly trained RC is able to predict accurately the state evolution of typical chaotic systems for about half a dozen Lyapunov times [30,34], which is much longer than the prediction horizon of the traditional methods developed in nonlinear science. Besides predicting the short-term state evolution, RC is also able to replicate faithfully the long-term statistical properties of chaotic systems, e.g., the dimension of strange attractors and the Lyapunov exponents [33]. This ability, known as climate replication, was exploited very recently to predict the critical transitions and bifurcation points in complex dynamical systems [20–22,24,25]. In particular, by incorporating a parameter-control channel into the standard RC, it has been demonstrated that a machine trained by the time series of several sampling states of a chaotic system is able to infer the dynamical properties of the other states not included in the training set. This technique, which was named parameter-aware RC (PARC) in Ref. [20], has been successfully applied to predict the critical transition of system collapses, infer the bifurcation diagram of chaotic systems [21,24,25], and anticipate the critical coupling for synchronization in coupled oscillators [22]. Whereas the efficacy of the PARC technique has been well demonstrated in these applications, the studies were restricted to modeling systems of noise-free signals and exact parameters. (An exception is Ref. [25], in which the PARC technique was utilized to infer the bifurcation diagram of modeling systems based on noisy data.) As noise perturbations and parameter uncertainty are inevitable in realistic systems, a question of general interest is therefore whether the PARC technique can be applied to realistic chaotic systems.

It is worth noting that the impact of noise on the performance of RC in predicting chaotic systems is twofold. On the one hand, noise-corrupted signals blur the system trajectories, making it difficult to infer accurately the equations of the system dynamics [38–41]. A typical case of this kind is measurement noise, which is commonly regarded as destructive to machine learning. To cope with measurement noise, techniques such as low-pass filters are usually adapted to process the data before feeding them into the machine [38,40]. On the other hand, noise might play a constructive role in machine learning in some circumstances [42–46]. For measurement noise, studies have shown that in the training phase the role of noise is similar to that of Tikhonov regularization [42], and the performance of the machine reaches its maximum at moderate noise [25,43,46]. For dynamical (intrinsic) noise, studies have shown that the introduction of a certain amount of noise is helpful for exploring the global information of the system dynamics and therefore is beneficial for machine learning, e.g., extending the transient dynamics and inferring the "unseen" attractors [44,45]. The nontrivial relationship between noise and machine learning makes the inference of chaotic dynamics from noisy signals not only a practical concern in applications but also an effective approach for exploring the working mechanism of the machines. For that, growing attention has been paid in recent years to the prediction and inference of chaos based on noisy signals [38–41,43–46]. Studies, however, are mostly conducted for modeling systems with artificial noise, and the validity of the results in realistic systems is yet to be checked.

In our present work, employing classic Chua circuits as examples, we attempt to reconstruct from measured data the bifurcation diagrams of the circuits using the PARC technique proposed recently for machine learning. Two specific scenarios are considered and investigated. In the first scenario, we collect the time series from a single circuit under several sampling parameters, and the mission is to reconstruct the whole bifurcation diagram in the parameter space. In the second scenario, we collect the time series of two coupled chaotic circuits under several coupling parameters, and the mission is to anticipate the variation of the synchronization degree of the coupled circuits with respect to the coupling parameter over a large range. We will demonstrate that, despite the presence of noise (measurement and dynamical noise) and parameter mismatch (between two coupled circuits), the PARC technique is capable of reconstructing the bifurcation diagrams with high precision in both scenarios. The rest of the paper is organized as follows. In the following section, we will describe the experimental setups and the how the data are acquired. The PARC technique will be introduced briefly in Sec. III. Our main results on the application of the PARC technique will be presented in Sec. IV, including the filtering effect of RC on noisy signals, the reconstruction of the bifurcation diagram for a single circuit, and the inference of the synchronization relationship between two coupled chaotic circuits. Finally, concluding remarks will be given in Sec. V.

II. EXPERIMENTAL SETUPS

Chua's circuit adopted in our studies is schematically shown in Fig. 1(a), which consists of two capacitors (C_1 and C_2), two linear resistors (R and R_1), one inductor (L), and a nonlinear resistor (NR) [47–50]. The equations of the system dynamics read

$$
\begin{aligned}
C_1\frac{dv_{C_1}}{dt} &= \frac{1}{R}(v_{C_2} - v_{C_1}) - g(v_{C_1}),\\
C_2\frac{dv_{C_2}}{dt} &= \frac{1}{R}(v_{C_1} - v_{C_2}) + i_L,\\
L\frac{di_L}{dt} &= -v_{C_2} - R_1 i_L,
\end{aligned}
\tag{1}
$$

with $g(v_{C_1}) = m_0 v_{C_1} + 0.5(m_1 - m_0)(|v_{C_1} + B_p| - |v_{C_1} - B_p|)$ being the characteristic curve of the nonlinear resistor. The characteristic curve of the nonlinear resistor is schematically plotted in Fig. 1(b), in which the parameters are $m_0 = -0.41$ mS (mA/V) $\pm$ 10%, $m_1 = -0.76$ mS $\pm$ 10%, and $B_p = 1.7$ V $\pm$ 5%. In our experiments, we fix the components $R_1 = 10\,\Omega \pm 1\%$, $C_1 = 10\,\text{nF} \pm 5\%$, $C_2 = 100\,\text{nF} \pm 5\%$, and $L = 20\,\text{mH} \pm 10\%$, while changing R over the range $(1.73\,\text{k}\Omega, 1.77\,\text{k}\Omega)$ to generate different dynamics. The variables measured in the experiments are v_{C_1} (the voltage of capacitor C_1), v_{C_2} (the voltage of capacitor C_2), and $v_{R_1} = i_L R_1$ (the voltage of resistor R_1), which are acquired by the sampling frequency $f_0 = 50$ kHz. For each value of R, we first let the circuit operate for a transient period of 1000 ms and then record the system state $(v_{C_1}, v_{C_2}, v_{R_1})$ for a period of 100 ms. As such, each time series contains $n = 5000$ data points.

Setting $R = 1.738\,\text{k}\Omega$ in the circuit, we plot in Figs. 2(a) and 2(b) the system trajectories projected onto the two-dimensional (2D) phase spaces (v_{C_1}, v_{C_2}) and (v_{C_2}, v_{R_1}), respectively. We see that the trajectories are blurred by noise severely, rendering it difficult to figure out accurately the periodicity of the trajectories. (The trajectories seem to be period 3 but might be period 6 or weakly chaotic.) We also see from Figs. 2(a) and 2(b) that compared to the variables v_{C_1} and v_{C_2}, the variable v_{R_1} is more corrupted by noise. For this reason, we choose the variable v_{C_1} to investigate experimentally the bifurcation diagram. Decreasing R from 1.77 to 1.73 kΩ by the decrement $\Delta R = 0.5\,\Omega$, we measure the time series of v_{C_1} for each value of R and, by recording the local minimums of v_{C_1}, plot in Fig. 2(c) the bifurcation diagram of the circuit. We see that, while Fig. 2(c) shows roughly the route from limit cycle to chaos through the period-doubling bifurcations, the bifurcation details are not clearly seen. For instance, we cannot infer from Fig. 2(c) when the system dynamics will present the period-8 orbit and what happens in the window $R \in [1735\,\Omega, 1741\,\Omega]$. The first objective of the present work is to reconstruct the bifurcation diagram of Chua's circuit with high quality (precision) based on the noisy series acquired for several values of R in experiments.

The second experiment we conduct is the synchronization of two coupled chaotic Chua circuits. The diagram of the coupled circuits is schematically shown in Fig. 3(a), and a photo of the experimental setup is given in Fig. 3(b). The dynamics of the coupled circuits are governed by the equations

$$
\begin{aligned}
C_3\frac{dv_{C_3}}{dt} &= \frac{1}{R_2}(v_{C_3} - v_{C_4}) - g(v_{C_3}) + \frac{1}{R_6}(v_{C_5} - v_{C_3}),\\
C_4\frac{dv_{C_4}}{dt} &= \frac{1}{R_2}(v_{C_4} - v_{C_3} + i_{L_1}),\\
L_1\frac{di_{L_1}}{dt} &= -v_{C_4} - R_4 i_{L_1},\\
C_5\frac{dv_{C_5}}{dt} &= \frac{1}{R_3}(v_{C_5} - v_{C_6}) - g(v_{C_5}) + \frac{1}{R_6}(v_{C_3} - v_{C_5}),\\
C_6\frac{dv_{C_6}}{dt} &= \frac{1}{R_3}(v_{C_6} - v_{C_5} + i_{L_2}),\\
L_2\frac{di_{L_2}}{dt} &= -v_{C_6} - R_5 i_{L_2},
\end{aligned}
\tag{2}
$$

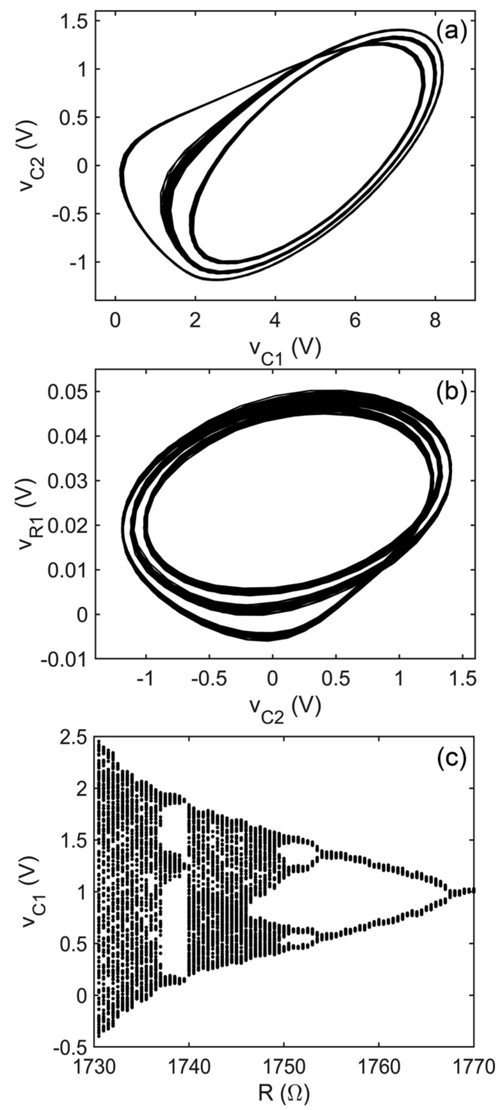

FIG. 2. Setting $R = 1.738\,\text{k}\Omega$ in Chua's circuit, system trajectories plotted on the planes (a) (v_{C_1}, v_{C_2}) and (b) (v_{C_2}, v_{R_1}). (c) With the data measured from experiments, the bifurcation diagram of Chua's circuit plotted according to the local minimums of v_{C_1}.

with $g(v_C)$ being the piecewise-linear function characterizing the nonlinear resistors. [The parameters of the nonlinear resistors are identical to the ones used in Fig. 1(b)]. Here, to better demonstrate the synchronization phenomenon, we choose the circuit components $R_{2,3} = 1.6\,\text{k}\Omega$, $C_{3,5} = 10\,\text{nF} \pm 5\%$, $C_{4,6} = 100\,\text{nF} \pm 5\%$, $L_{1,2} = 26\,\text{mH} \pm 10\%$, and $R_{4,5} = 10\,\Omega \pm 10\%$. Note that due to the mismatched parameters (components), the two circuits are not identical. Despite the mismatched parameters, both circuits present chaotic motion when isolated, as depicted in Fig. 3(c). The two circuits are coupled through the resistor R_6, which can be adjusted between 9 kΩ (strong coupling) and 13 kΩ (weak coupling) with high precision ($\sim$0.1Ω). Still, the currents of the inductors i_{L_1} and i_{L_2} are monitored by the voltages v_{R_4} and v_{R_5}, respectively, and data are acquired with the sampling frequency $f_0 = 50$ kHz for a period of 100 ms in each experiment.

Setting $R_6 = 10.2\,\text{k}\Omega$, we plot in Fig. 3(d) the relationship between the voltages v_{C_3} (from circuit 1) and v_{C_5} (from

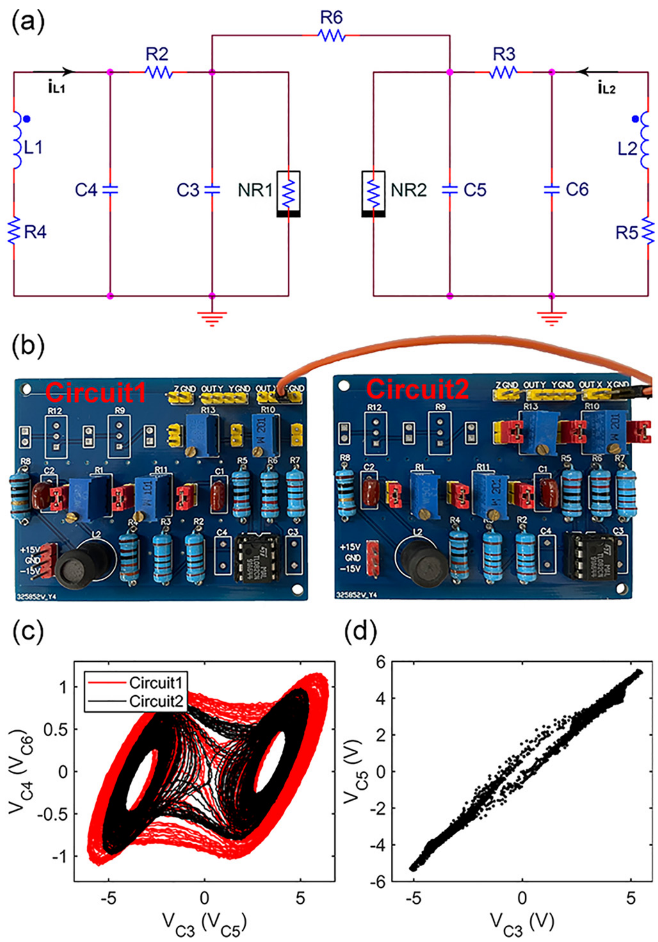

FIG. 3. (a) Schematic of two coupled Chua circuits. (b) The experimental setup. (c) The trajectories of isolated chaotic circuits on the 2D phase spaces (v_{C_3}, v_{C_4}) and (v_{C_5}, v_{C_6}). (d) For $R_6 = 10.2\,\mathrm{k}\Omega$ in the experiment, v_{C_3} versus v_{C_5} plotted according to the measured data.

circuit 2). We see that the data are distributed roughly along the diagonal line, indicating that the two circuits are oscillating in a weakly coherent fashion. The synchronization degree of the coupled circuits is evaluated by the time-averaged synchronization error $\delta r = \langle \delta e(t) \rangle_T$, with $\delta e = \sqrt{(v_{C_3} - v_{C_5})^2 + (v_{C_4} - v_{C_6})^2 + (v_{R_4} - v_{R_5})^2}$ being the instant synchronization error between the circuits and $\langle \cdot \rangle$ being the time-average function. For the results shown in Fig. 3(d), we have $\delta r \approx 0.303V$. Here, the question we are interested in is: Given experiments are conducted for only several values of R_6 and the time series of the sampling states are available, can we anticipate the synchronization degree of the coupled circuits for a random R_6 and, furthermore, the variation of the synchronization degree with respect to R_6 over a wide range? The second objective of the present work is to demonstrate that this question can be addressed using the technique of PARC in machine learning.

III. PARAMETER-AWARE RESERVOIR COMPUTING

The PARC technique exploited for reconstructing bifurcation diagrams is generalized from the one proposed in

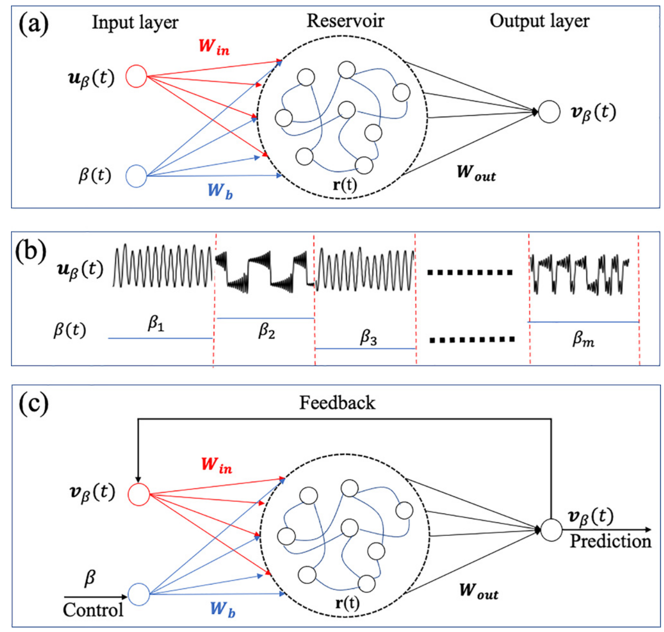

FIG. 4. Schematic of the PARC technique. (a) The open-loop configuration of the machine in the training phase. (b) Schematic of the training data. (c) The closed-loop configuration of the machine in the predicting phase.

Refs. [20–23]. Like conventional RCs, the machine employed here is also constructed using four modules: the I/R layer (input to reservoir), the parameter-control channel, the reservoir network, and the R/O layer (reservoir to output). The structure of the machine is schematically shown in Fig. 4(a). The I/R layer is characterized by the matrix $\mathbf{W}_{\text{in}} \in \mathbb{R}^{D_r \times D_{\text{in}}}$, which couples the input vector $\mathbf{u}_\beta(t) \in \mathbb{R}^{D_{\text{in}}}$ to the reservoir network. Here, $\mathbf{u}_\beta(t)$ denotes the input vector acquired from the target system at time t under the specific bifurcation parameter β. (For the first objective, in which the task is to reconstruct the bifurcation diagram of a single circuit, we have $\beta = R$; for the second objective, in which the task is to anticipate the variation of the synchronization degree of coupled chaotic circuits, we have $\beta = R_6$.) The elements of $\mathbf{W}_{\text{in}}$ are randomly drawn from a uniform distribution within the range $[-\sigma, \sigma]$. The parameter-control channel is characterized by the vector $\mathbf{s} = \beta \mathbf{W}_b$, with β being the control parameter and $\mathbf{W}_b \in \mathbb{R}^{D_r}$ being the bias vector. The control parameter β can be treated as an additional input channel marking the input vector $\mathbf{u}(t)$. The elements of $\mathbf{W}_b$ are also drawn randomly within the range $[-\sigma, \sigma]$. The reservoir network contains D_r nodes, with the initial states of the nodes being randomly chosen from the interval $[-1, 1]$. The states of the nodes in the reservoir network, $\mathbf{r}(t) \in \mathbb{R}^{D_r}$, are updated as

$$\mathbf{r}(t + \Delta t) = (1 - \alpha)\mathbf{r}(t) + \alpha \tanh[\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{\text{in}}\mathbf{u}_\beta(t) + \beta \mathbf{W}_b]. \quad (3)$$

Here, Δt is the time step for updating the reservoir network, $\alpha \in (0, 1]$ is the leaking rate, and $\mathbf{A} \in \mathbb{R}^{D_r \times D_r}$ is a weighted adjacency matrix representing the coupling relationship between nodes in the reservoir. The adjacency matrix $\mathbf{A}$ is constructed as a sparse random Erdös-Rényi matrix: with probability p, each element of the matrix is set as a nonzero value drawn randomly from the interval $[-1, 1]$. The matrix $\mathbf{A}$ is rescaled to make its spectral radius equal λ. The output layer is characterized by the matrix $\mathbf{W}_{\text{out}} \in \mathbb{R}^{D_{\text{out}} \times D_r}$, which generates the output vector, $\mathbf{v}(t) \in \mathbb{R}^{D_{\text{out}}}$, according to the equation

$$\mathbf{v}(t + \Delta t) = \mathbf{W}_{\text{out}} \tilde{\mathbf{r}}(t + \Delta t), \tag{4}$$

with $\tilde{\mathbf{r}} \in \mathbb{R}^{D_r}$ being the new state vector transformed from the reservoir state (i.e., $\tilde{r}_i = r_i$ for the odd nodes and $\tilde{r}_i = r_i^2$ for the even nodes) [34] and $\mathbf{W}_{\text{out}}$ being the output matrix to be estimated by a training process. Except for $\mathbf{W}_{\text{out}}$, all other parameters of the RC, e.g., $\mathbf{W}_{\text{in}}$, $\mathbf{A}$, and $\mathbf{W}_b$, are fixed at construction. For the sake of simplicity, we set $D_{\text{out}} = D_{\text{in}}$ in our studies [32–34].

The implementation of PARC consists of three phases: training, validating, and predicting. The mission of the training phase is to find a suitable output matrix $\mathbf{W}_{\text{out}}$ so that the output vector $\mathbf{v}(t + \Delta t)$ as calculated by Eq. (4) is as close as possible to the input vector $\mathbf{u}(t + \Delta t)$ for $t = (\tau + 1)\Delta t, \ldots, (\tau + \hat{L})\Delta t$, with $T_0 = \tau \Delta t$ being the transient period (used for removing the impact of the initial conditions of the reservoir) and $\hat{L}$ being the length of the training series. This is done by minimizing the cost function with respect to $\boldsymbol{W}_{\text{out}}$ [32–34], which gives

$$\boldsymbol{W}_{\text{out}} = \boldsymbol{U}\boldsymbol{V}^T(\boldsymbol{V}\boldsymbol{V}^T + \eta \mathbb{I})^{-1}. \tag{5}$$

Here, $\boldsymbol{V} \in \mathbb{R}^{D_r \times \hat{L}}$ is the state matrix whose kth column is $\tilde{\boldsymbol{r}}[(\tau + k)\Delta t]$, $\boldsymbol{U} \in \mathbb{R}^{D_{\text{out}} \times \hat{L}}$ is a matrix whose kth column is $\boldsymbol{u}[(\tau + k)\Delta t]$, $\mathbb{I}$ is the identity matrix, and η is the ridge regression parameter for avoiding the overfitting. We note that in the training phase the input data consist of two different time series: (1) the input vector $\mathbf{u}_\beta(t)$ representing the state of the target system and (2) the control parameter $\beta(t)$ labeling the condition under which the input vector $\mathbf{u}_\beta(t)$ is acquired. Specifically, the input vector $\mathbf{u}_\beta(t)$ is composed of m segments of length $\hat{n}$, while each segment is a time series obtained from the target system under the specific control parameter β. As such, the training data set is a concatenation of the sampling series, and $\beta(t)$ is a step function of time. The structure of the training data is schematically shown in Fig. 4(b).

A machine that performs well on the training data might not perform equally well on the testing data. Finding the optimal machine that performs well on both the training and testing data is the mission for the validating phase. The set of hyperparameters to be optimized in the machine include D_r (the size of the reservoir network), p (the density of the adjacency matrix $\mathbf{A}$), σ (the range defining the input matrix and the bias vector), λ (the spectral radius of the adjacency matrix $\mathbf{A}$), η (the regression coefficient), and α (the leaking rate). In our studies, the optimal hyperparameters are obtained by scanning each hyperparameter over a certain range in the parameter space using conventional optimization algorithms such as the Bayesian and surrogate optimization algorithms [20]. After finding the optimal machine, we then utilize it to reconstruct the bifurcation diagrams, namely, the predicting phase. Figure 4(c) shows the flowchart of the machine in the predicting phase. In making the predictions, we replace $\mathbf{u}_\beta(t)$ with $\mathbf{v}(t)$ (so that the machine is working in the closed-loop configuration) while setting the control parameter β to a specific value of interest. As such, in the predicting phase the machine is still driven by the externally added parameter β. The output vector $\mathbf{v}(t)$ then gives the predictions, on the basis of which the climate of the system dynamics associated with β can be replicated. (Still, before making the predictions, a short transient is discarded to avoid the impact of the initial conditions of the reservoir.) Finally, by tuning β in the parameter space, we can reconstruct the whole bifurcation diagram according to the machine predictions.

IV. RESULTS

We first utilize the PARC technique to reconstruct the bifurcation diagram of a single circuit. We begin by choosing the set of sampling states from which the data are acquired from experiments. Previous studies showed that the performance of PARC is influenced by both the number and locations of the sampling states [20,22,23]. In general, the more sampling states there are, the better the machine predictions are. Additionally, to replicate the dynamics of a new state that is not included in the sampling set, it is better to choose the sampling states evenly over the parameter space. For demonstration purposes, here we choose $m = 3$ sampling states over the bifurcation range plotted in Fig. 5(c), $R = 1.735$, 1.745, and 1.755 kΩ. For each of the sampling states, we record the system evolution for $T = 100$ ms, from which we obtain a time series of $n = 5000$ data points. Following the standard strategies in machine learning, we separate the time series into two segments of equal length, with the first half being used as training data and the second half as validating data. The size (length) of the whole training dataset therefore is $\hat{N} = m \times n/2 = 7500$, so it is the validating dataset. (To make the predictions more relevant to the experimental results, here we use the raw data as the input; i.e., the data are not processed.)

We next train the machine and find the optimal set of hyperparameters. In training the machine, the transient series used to remove the impact of the initial conditions of the reservoir contains $\tau = 200$ data points (which applies to each of the sampling series in the training data). As such, the total number of data points used for estimating the output matrix $\mathbf{W}_{\text{out}}$ is $\hat{L} = m \times \hat{n} = m \times (n/2 - \tau) = 6900$. To find the optimal set of hyperparameters, we search the hyperparameters over the ranges $D_r \in (200, 1000)$, $p \in (0, 0.2)$, $\sigma \in (0, 1)$, $\lambda \in (0.5, 1)$, $\eta \in (1 \times 10^{-8}, 1 \times 10^{-2})$, and $\alpha \in (0, 1]$ using the Bayesian optimization algorithm. Each set of hyperparameters defines a machine whose performance is evaluated on the validating data according to the prediction error $\langle |\mathbf{u}(t) - \mathbf{v}(t)| \rangle_T$. Still, in evaluating the machine performance using the validating data, a transient series of $\tau = 200$ points is used to remove the impact of the initial conditions of the reservoir. For this application, the optimal hyperparameters are $(D_r, p, \sigma, \lambda, \eta, \alpha) = (502, 0.15, 0.32, 0.85, 1.2 \times 10^{-5}, 0.54)$, which define the optimal machine to be used for prediction purposes.

Before employing the trained machine to reconstruct the bifurcation diagram, we first check the capability of the machine for predicting the dynamics of a new state not included in the sampling set. The example state we choose is $R = 1.738$ kΩ. [The trajectories of this state plotted according to

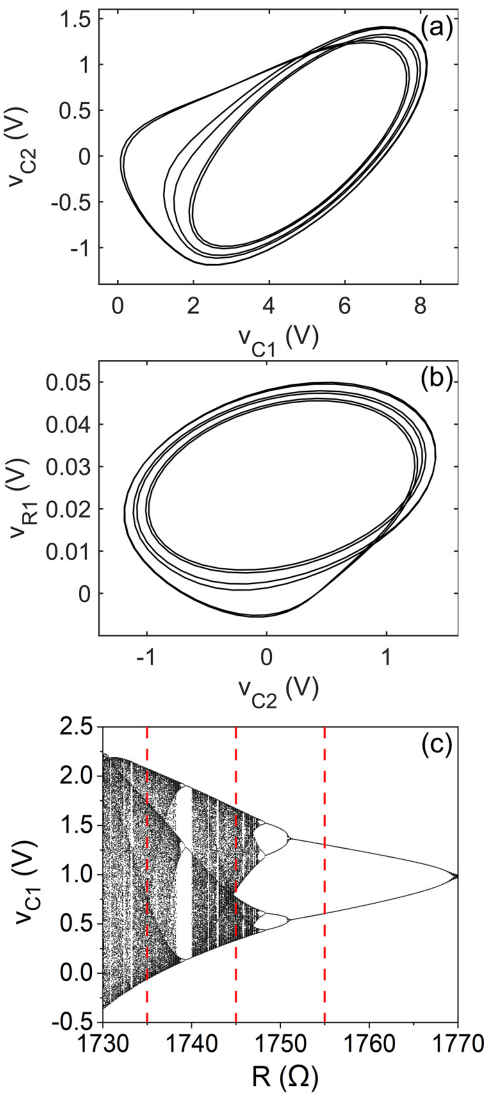

FIG. 5. Reconstructing the bifurcation diagram of Chua's circuit using the PARC technique. (a) and (b) The trajectories predicted by the machine for the parameter $R = 1.738\,\mathrm{k\Omega}$, which is not included in the sampling set. (c) The bifurcation diagram predicted by the PARC technique. Red dashed lines denote the sampling states from which data are measured from experiments.

experimental data are shown in Figs. 2(a) and 2(b).] Setting the control parameter as $\beta = 1.738\,\mathrm{k\Omega}$, we now operate the machine in the closed-loop configuration [see Fig. 4(c)]. After a transient period of $\tau = 1000$ iterations, the machine begins to output the predictions. The trajectories predicted by the machine are plotted in Figs. 5(a) and 5(b). We see that compared to the smeared trajectories plotted in Figs. 2(a) and 2(b), in Figs. 5(a) and 5(b) the trajectories clearly show the period-6 orbits. We therefore see that the machine is able not only to infer the dynamics of a new state but also to restore from noise-contaminated signals the main features of the true trajectories (i.e., the "climate" of the system dynamics) [51]. We proceed to reconstruct the bifurcation diagram of the circuit using the PARC technique. This is done by increasing the control parameter from $\beta = 1.73$ to $1.77\,\mathrm{k\Omega}$ gradually, while for each value of R we collect from the machine output a sequence of 10 000 data points. Figure 5(c) shows the bifurcation diagram plotted according to the machine predictions. Compared with the experimentally obtained results [see Fig. 2(c)], we see that the bifurcation diagram predicted by the machine is of high quality and precision. Specifically, we can infer from the reconstructed bifurcation diagram not only the transition points of the high-order periodic orbits but also the periodic windows embedded in the chaotic regions.

We continue to anticipate the synchronization degree of two coupled chaotic Chua circuits using the PARC technique. Still, to generate the training and validating datasets, we acquire from experiments the time series of $m = 3$ sampling states, $R_6 = 9.4\,\mathrm{k\Omega}$, $10.2\,\mathrm{k\Omega}$ [the state shown in Fig. 3(d)], and $11\,\mathrm{k\Omega}$. Each series contains $n = 10\,000$ data points, with the first half being used as training data and the second half being used as validating data. The transient period of the training phase contains $\tau = 500$ data points, and the same transient period is applied in the validating phase. Still, the machine hyperparameters are optimized using the Bayesian optimization algorithm. In this application, the optimal hyperparameters are $(D_r, p, \sigma, \lambda, \eta, \alpha) = (983, 4.8 \times 10^{-3}, 0.88, 0.39, 2.9 \times 10^{-3}, 0.73)$.

We first check the capability of the trained machine to replicate the synchronization dynamics of the sampling states. Setting the control parameter as $\beta = 10.2\,\mathrm{k\Omega}$, we operate the machine in the closed-loop configuration [see Fig. 4(c)] and estimate from the machine outputs the synchronization error δr between the circuits. The results show that $\delta r \approx 0.34V$, which is in good agreement with the experimental results ($\delta r \approx 0.30V$). Figure 6(a) shows the relationship between v_{C_3} and v_{C_5} for the machine-predicted data (red dots), which is also consistent with the one plotted according to the experimental data (black dots).

We next check the capability of the machine to anticipate the synchronization climate of a new state not included in the sampling set. To demonstrate, we set $\beta = 12\,\mathrm{k\Omega}$ and, based on the machine predictions, plot in Fig. 6(b) the relationship between v_{C_3} and v_{C_5}. Compared to the results for $\beta = 10.2\,\mathrm{k\Omega}$, we see that the synchronization degree between the circuits clearly decreases for $\beta = 12\,\mathrm{k\Omega}$. Specifically, for $\beta = 12\,\mathrm{k\Omega}$, the synchronization error estimated from the machine predictions is $\delta r \approx 0.65V$. This estimation is also in good agreement with the experimental result ($\delta r \approx 0.64V$), as depicted in Fig. 6(b).

We finally utilize the machine to anticipate the variation of the synchronization error δr with respect to the coupling coefficient R_6 over a wide range in the parameter space. In doing this, we increase β from 9 to $13\,\mathrm{k\Omega}$ with the increment $\Delta\beta = 0.2\,\mathrm{k\Omega}$, and for each β we calculate from the machine outputs the value of δr. The results are plotted in Fig. 6(c) (red circles), which shows that with the increase of β, the value of δr is monotonically increased. To validate the predictions, we tune R_6 in the experiment over the same range, and for each R_6 we calculate from the measured data the synchronization error. The experimental results are also plotted in Fig. 6(c) (black squares). We see that the predicted and experimental results are consistent within the range $R_6 \in (9\,\mathrm{k\Omega}, 12\,\mathrm{k\Omega})$ but slightly diverge when $R_6 > 12\,\mathrm{k\Omega}$. The difference between the predicted and experimental results at large R_6 is attributed to the large distance between the sampling and testing states,

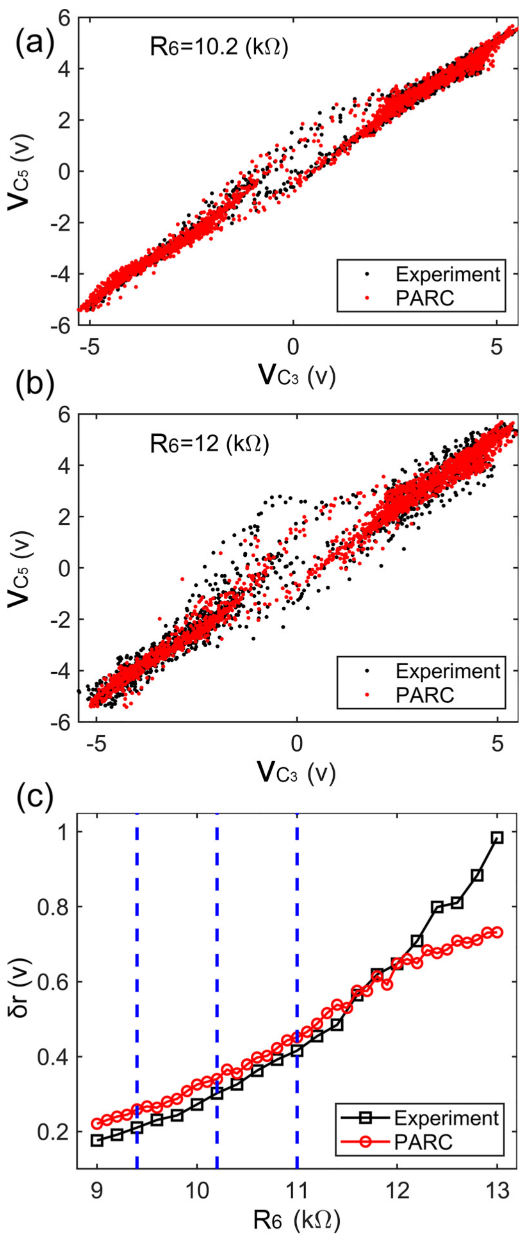

FIG. 6. Reconstructing the synchronization transition of two coupled chaotic Chua circuits using the PARC technique. The relationship between v_{C_3} and v_{C_5} for (a) $R_6 = 10.2\,\text{k}\Omega$ and (b) $R_6 = 12\,\text{k}\Omega$. Black dots are results acquired from experiments. Red dots are results predicted by the machine. (c) The variation of the synchronization error between the coupled circuits δr with respect to the coupling coefficient R_6. Black squares are results obtained from experiments. Red circles are results predicted by the machine. Blue dashed lines denote the sampling states from which data are measured from experiments.

which was also observed in previous studies [20,22,23] and was confirmed by additional simulations [51].

V. CONCLUDING REMARKS

In reconstructing the bifurcation diagram of chaotic systems based on measured data, two of the major difficulties encountered in practice are that (1) the signals are contaminated by noise and (2) the signals are acquired at only a few sampling states. The former makes the reconstructed bifurcation diagram coarse and unclear; the latter renders the reconstructed bifurcation diagram fragmented and incomplete. In the present work, using the experimental data of chaotic Chua circuits, we showed that both difficulties can be well addressed with the PARC technique proposed recently for machine learning. Two scenarios were considered and investigated: reconstructing the bifurcation diagram of a single circuit and anticipating the synchronization transition of two coupled chaotic circuits. In the first scenario, we demonstrated that with the noisy signals acquired at several sampling states, the trained machine is able to reconstruct the whole bifurcation diagram with high precision. The success of the machine in reconstructing the bifurcation diagram is attributed to the noise-filtering effect of the reservoir and the property of transfer learning. Specifically, fed with noisy signals from which the system dynamics cannot be inferred directly, the reservoir is able to output a smooth and clear trajectory capturing the main features of the noise-free dynamics. Guided by the parameter-control channel, the knowledge that the machine learned from the time series of the sampling states can be transferred to infer the dynamics of a new state not included in the sampling set. In the second scenario, we demonstrated that, trained by the noisy signals collected at a handful of coupling parameters, the machine is able to anticipate the variation of the synchronization degree of the coupled circuits with respect to the coupling parameter over a wide range. Whereas the capability of PARC for inferring the dynamics climate of chaotic systems has been well demonstrated in the literature, previous studies were mainly based on modeling systems of noise-free signals [20–24]. (Model-free inference of the bifurcation diagram of chaotic systems based on noisy signals was studied very recently in Ref. [25].) Our studies show that this technique can also be applied to noisy signals generated by realistic systems.

Although our studies preliminarily demonstrated the capability of the PARC technique for reconstructing the bifurcation diagram of realistic chaotic systems, many questions remain to be addressed. First, for convenience and simplicity, we adopted Chua's circuits as examples to demonstrate the performance of the PARC technique. The applicability of this technique to other real-world chaotic systems is yet to be checked. Second, recent studies showed that noise might play a constructive role in the machine learning of chaotic systems [43–46]. In particular, a resonance phenomenon was observed in both chaos prediction and climate replication, where it was shown that machine performance can be improved by introducing a certain amount of noise [43,46]. It will be interesting to check whether a similar phenomenon can be observed in experimental data generated by Chua's circuits. Third, our studies focused on only the low-dimensional chaotic systems (a single Chua's circuit and two coupled chaotic Chua circuits). It remains unclear whether the same PARC technique can be applied to high-dimensional chaotic systems, e.g., spatially extended chaotic systems and large-size complex networks of coupled oscillators. In applying the technique to high-dimensional chaotic systems, one difficulty concerns the large size of the reservoir network. One possible approach for addressing this difficulty could be adopting the scheme of parallel RC [34], which, however, might require a significant modification of the machine structure. Fourth, an important feature of many real-world chaotic systems is that their asymptotic dynamics are dependent on the initial conditions, namely, the property of multistability [52]. The application of the PARC technique to reconstruct the bifurcation diagrams of multistable chaotic systems, probably by incorporating some additional modules in the current machine, is an interesting topic warranting further study. Finally, recent studies showed that the technique of RC can also be applied to predict the dynamics of quantum systems [53,54]. In particular, it was demonstrated that under some circumstances, the introduction of a certain amount of noise can improve the performance of a quantum RC [55]. The application of the PARC technique to quantum systems is another intriguing topic warranting further study.

The program codes and experimental data used in this study can be obtained from Ref. [56].

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (NNSFC) under Grants No. 12275165 and No. 12105165. Xingang Wang was also supported by the Fundamental Research Funds for the Central Universities under Grant No. GK202202003.

[1] E. Ott, *Chaos in Dynamical Systems* (Cambridge University Press, Cambridge, 2002).

[2] M. Scheffer, *Critical Transitions in Nature and Society* (Princeton University Press, Princeton, NJ, 2009).

[3] M. Gladwell, *The Tipping Point: How Little Things Can Make a Big Difference* (Little, Brown & Company, 2007).

[4] M. Scheffer, J. Bascompte, W. A. Brock, V. Brovkin, S. R. Carpenter, V. Dakos, H. Held, E. H. van Nes, M. Rietkerk, and G. Sugihara, Early-warning signals for critical transitions, Nature (London) **461**, 53 (2009).

[5] T. M. Lenton, H. Held, E. Kriegler, J. W. Hall, W. Lucht, S. Rahmstorf, and H. J. Schellnhuber, Tipping elements in the Earth's climate system, Proc. Natl. Acad. Sci. USA **105**, 1786 (2008).

[6] T. M. Lenton, J. Rockström, O. Gaffney, S. Rahmstorf, K. Richardson, W. Steffen, and H. J. Schellnhuber, Climate tipping points - too risky to bet against, Nature (London) **575**, 592 (2019).

[7] M. Scheffer, S. Carpenter, J. A. Foley, C. Folke, and B. Walker, Catastrophic shifts in ecosystems, Nature (London) **413**, 591 (2001).

[8] M. Hirota, M. Holmgren, E. H. V. Nes, and M. Scheffer, Global resilience of tropical forest and savanna to critical transitions, Science **334**, 232 (2011).

[9] A. G. Haldane and R. M. May, Systemic risk in banking ecosystems, Nature (London) **469**, 351 (2011).

[10] A. Majdandzic, L. A. Braunstein, C. Curme, I. Vodenska, S. Levy-Carciente, H. E. Stanley, and S. Havlin, Multiple tipping points and optimal repairing in interacting networks, Nat. Commun. **7**, 10850 (2016).

[11] F. Takens, Detecting strange attractors in turbulence, in *Dynamical Systems and Turbulence, Warwick*, edited by D. Rand and L.-S. Young, Lecture Notes in Mathematics Vol. 898 (Springer, Berlin, Heidelberg, 1980), pp 366–381.

[12] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw, Geometry from a time series, Phys. Rev. Lett. **45**, 712 (1980).

[13] R. Tokunaga, S. Kajiwara, and T. Matsumoto, Reconstructing bifurcation diagrams only from time-waveforms, Phys. D (Amsterdam, Neth.) **79**, 348 (1994).

[14] E. Bagarinao, K. Pakdaman, T. Nomura, and S. Sato, Reconstructing bifurcation diagrams from noisy time series using nonlinear autoregressive models, Phys. Rev. E **60**, 1073 (1999).

[15] G. Langer and U. Parlitz, Modeling parameter dependence from time series, Phys. Rev. E **70**, 056217 (2004).

[16] R. Cestnik and M. Abel, Inferring the dynamics of oscillatory systems using recurrent neural networks, Chaos **29**, 063128 (2019).

[17] R. Follmann and R. Epaminondas, Predicting slow and fast neuronal dynamics with machine learning, Chaos **29**, 113119 (2019).

[18] Y. Itoh, S. Uenohara, M. Adachi, T. Morie, and K. Aihara, Reconstructing bifurcation diagrams only from time-series data generated by electronic circuits in discrete-time dynamical systems, Chaos **30**, 013128 (2020).

[19] H. Zhao, Inferring the dynamics of "black-box" systems using a learning machine, Sci. China: Phys., Mech. Astron. **64**, 270511 (2021).

[20] L.-W. Kong, H. Fan, C. Grebogi, and Y.-C. Lai, Machine learning prediction of critical transition and system collapse, Phys. Rev. Res. **3**, 013090 (2021).

[21] J. Z. Kim, Z. Lu, E. Nozari, G. J. Pappas, and D. S. Bassett, Teaching recurrent neural networks to infer global temporal structure from local examples, Nat. Mach. Intell. **3**, 316 (2021).

[22] H. Fan, L.-W. Kong, Y.-C. Lai, and X. G. Wang, Anticipating synchronization with machine learning, Phys. Rev. Res. **3**, 023237 (2021).

[23] H. Zhang, H. Fan, L. Wang, and X. G. Wang, Learning Hamiltonian dynamics with reservoir computing, Phys. Rev. E **104**, 024205 (2021).

[24] M. Roy, S. Mandal, C. Hens, A. Prasad, N. V. Kuznetsov, and M. D. Shrimali, Model-free prediction of multistability using echo state network, Chaos **32**, 101104 (2022).

[25] L.-W. Kong, Y. Weng, B. Glaz, M. Haile, and Y.-C. Lai, Reservoir computing as digital twins for nonlinear dynamical systems, Chaos **33**, 033111 (2023).

[26] S. L. Brunton, J. L. Proctor, and J. N. Kutz, Discovering governing equations from data: Sparse identification of nonlinear dynamical systems, Proc. Natl. Acad. Sci. USA **113**, 3932 (2016).

[27] W. Wang, Y.-C. Lai, and C. Grebogi, Data based identification and prediction of nonlinear and complex dynamical systems, Phys. Rep. **644**, 1 (2016).

[28] T. Y. Chen, Y. Chen, H. J. Yang, J. H. Xiao, and G. Hu, Reconstruction of dynamic structures of experimental setups based on

measurable experimental data only, Chin. Phys. B **27**, 030503 (2018).

[29] W. Maass, T. Natschlager, and H. Markram, Real-time computing without stable states: A new framework for neural computation based on perturbations, Neural Comput. **14**, 2531 (2002).

[30] H. Jaeger and H. Haas, Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication, Science **304**, 78 (2004).

[31] L. Domingo, M. Grande, F. Borondo, and J. Borondo, Anticipating food price crises by reservoir computing, Chaos Solit. Fractals **174**, 113854 (2023).

[32] Z. Lu, J. Pathak, B. Hunt, M. Girvan, R. Brockett, and E. Ott, Reservoir observers: Model-free inference of unmeasured variables in chaotic systems, Chaos **27**, 041102 (2017).

[33] J. Pathak, Z. Lu, B. Hunt, M. Girvan, and E. Ott, Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data, Chaos **27**, 121102 (2017).

[34] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach, Phys. Rev. Lett. **120**, 024102 (2018).

[35] T. Weng, H. Yang, C. Gu, J. Zhang, and M. Small, Synchronization of chaotic systems and their machine-learning models, Phys. Rev. E **99**, 042203 (2019).

[36] H. Fan, J. Jiang, C. Zhang, X. G. Wang, and Y.-C. Lai, Long-term prediction of chaotic systems with machine learning, Phys. Rev. Res. **2**, 012080(R) (2020).

[37] K. Nakajima and I. Fischer, *Reservoir Computing: Theory, Physical Implementations, and Applications* (Springer, Singapore, 2021).

[38] N. A. K. Doan, W. Polifke, and L. Magri, Physics-informed echo state networks, J. Comput. Sci. **47**, 101237 (2020).

[39] D. Patel, D. Canaday, M. Girvan, A. Pomerance, and E. Ott, Using machine learning to predict statistical properties of non-stationary dynamical processes: System climate, regime transitions, and the effect of stochasticity, Chaos **31**, 033149 (2021).

[40] C. Nathe, C. Pappu, N. A. Mecholsky, J. D. Hart, T. Carroll, and F. Sorrentino, Reservoir computing with noise, Chaos **33**, 041101 (2023).

[41] Z. Lin, Y. Liang, J. Zhao, J. Li, and T. Kapitaniak, Prediction of dynamic systems driven by Lévy noise based on deep learning, Nonlinear Dyn. **111**, 1511 (2023).

[42] C. Bishop, Training with noise is equivalent to Tikhonov regularization, Neural Comput. **7**, 108 (1995).

[43] I. Estébanez, I. Fischer, and M. C. Soriano, Constructive role of noise for high-quality replication of chaotic attractor dynamics using a hardware-based reservoir computer, Phys. Rev. Appl. **12**, 034058 (2019).

[44] A. Röhm, D. J. Gauthier, and I. Fischer, Model-free inference of unseen attractors: Reconstructing phase space features from a single noisy trajectory using reservoir computing, Chaos **31**, 103127 (2021).

[45] H. Fan, L. Wang, Y. Du, Y. F. Wang, J. H. Xiao, and X. G. Wang, Learning the dynamics of coupled oscillators from transients, Phys. Rev. Res. **4**, 013137 (2022).

[46] Z.-M. Zhai, L.-W. Kong, and Y.-C. Lai, Emergence of a resonance in machine learning, Phys. Rev. Res. **5**, 033127 (2023).

[47] L. O. Chua, Chua's circuit: An overview ten years later, J. Circuits Syst. Comput. **4**, 117 (1994).

[48] M. P. Kennedy, Robust OP amp realization of Chua's circuit, Frequenz **46**, 66 (1992).

[49] L. O. Chua, L. Kocarev, K. Eckart, and M. Itoh, Experimental chaos synchronization in Chua's circuit, Int. J. Bifurcation Chaos **2**, 705 (1992).

[50] M. S. Baptista, T. P. Silva, J. C. Sartorelli, and I. L. Caldas, Phase synchronization in the perturbed Chua circuit, Phys. Rev. E **67**, 056212 (2003).

[51] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevE.109.024210 for more details on the filtering effect of RC on noisy trajectories and the impacts of sampling states on machine performance.

[52] A. N. Pisarchik and U. Feudel, Control of multistability, Phys. Rep. **540**, 167 (2014).

[53] A. Kutvonen, K. Fujii, and T. Sagawa, Optimizing a quantum reservoir computer for time series prediction, Sci. Rep. **10**, 14687 (2020).

[54] P. Mujal, R. Martínez-Peña, J. Nokkala, J. García-Beni, G. L. Giorgi, M. C. Soriano, and R. Zambrini, Opportunities in quantum reservoir computing and extreme learning machines, Adv. Quantum Technol. **4**, 2100027 (2021).

[55] L. Domingo, G. Carlo, and F. Borondo, Taking advantage of noise in quantum reservoir computing, Sci. Rep. **13**, 8790 (2023).

[56] https://github.com/yike7758258/Reconstructing-Bifurcation-Diagrams-with-RC.git.