Chaos

Predicting phase and sensing phase coherence in chaotic systems with machine learning

Cite as: Chaos **30**, 083114 (2020); https://doi.org/10.1063/5.0006304
Submitted: 07 March 2020 . Accepted: 29 June 2020 . Published Online: 28 July 2020

Chun Zhang, Junjie Jiang, Shi-Xian Qu, and Ying-Cheng Lai

Predicting phase and sensing phase coherence in chaotic systems with machine learning

Cite as: Chaos **30**, 083114 (2020); doi: 10.1063/5.0006304
Submitted: 7 March 2020 · Accepted: 29 June 2020 ·
Published Online: 27 July 2020 · Publisher error corrected: 3 August 2020

View Online Export Citation CrossMark

Chun Zhang,[1,2] Junjie Jiang,[2] Shi-Xian Qu,[1,a)] and Ying-Cheng Lai[2,3,b)]

AFFILIATIONS

[1] School of Physics and Information Technology, Shaanxi Normal University, Xi'an 710062, China
[2] School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, Arizona 85287, USA
[3] Department of Physics, Arizona State University, Tempe, Arizona 85287, USA

[a)] **Email:** sxqu@snnu.edu.cn
[b)] **Author to whom correspondence should be addressed:** Ying-Cheng.Lai@asu.edu

ABSTRACT

Recent interest in exploiting machine learning for model-free prediction of chaotic systems focused on the time evolution of the dynamical variables of the system as a whole, which include both amplitude and phase. In particular, in the framework based on reservoir computing, the prediction horizon as determined by the largest Lyapunov exponent is often short, typically about five or six Lyapunov times that contain approximately equal number of oscillation cycles of the system. There are situations in the real world where the phase information is important, such as the ups and downs of species populations in ecology, the polarity of a voltage variable in an electronic circuit, and the concentration of certain chemical above or below the average. Using classic chaotic oscillators and a chaotic food-web system from ecology as examples, we demonstrate that reservoir computing can be exploited for long-term prediction of the phase of chaotic oscillators. The typical prediction horizon can be orders of magnitude longer than that with predicting the entire variable, for which we provide a physical understanding. We also demonstrate that a properly designed reservoir computing machine can reliably sense phase synchronization between a pair of coupled chaotic oscillators with implications to the design of the parallel reservoir scheme for predicting large chaotic systems.

The idea of using reservoir computing, a type of recurrent neural networks, for predicting chaotic systems was articulated previously but recently it has gained momentum and focused attention. The core of a reservoir computing machine is a nonlinear dynamical network capable of self-evolution. After proper training with time series data from the target chaotic system to be predicted, starting from the same initial condition, the machine can generate a dynamical trajectory that stays close to the true trajectory but only for a limited amount of time, as stipulated by the hallmark of chaos: sensitive dependence on initial conditions. In existing studies, the goal has been to predict the time evolution of the full dynamical variables of the target system for which the prediction horizon is typically on the order of several Lyapunov time. In physical or biological applications, it is often desired to know the tendency of state evolution, i.e., the phase or the ups and downs of some key dynamical variables of interest. The main point of this article is that if the goal is to predict the phase evolution of the system, then the prediction horizon can be orders of magnitude longer than that with predicting the full dynamical variables. We demonstrate this result using two representative chaotic oscillators and provide a heuristic explanation for the long phase prediction horizon with support from the estimated Lyapunov spectrum of the reservoir computing system. We also test the ability for a reservoir computing machine to correctly sense and distinguish phase coherence of coupled chaotic oscillators with the implication to the utilization of the configuration of parallel reservoirs.

I. INTRODUCTION

The idea and principle of exploiting reservoir computing, a type of recurrent neural networks[1–4] in machine learning, for predicting chaotic systems were first laid out about two decades ago.[1,3] With the rapid development of the field of machine learning, interest in reservoir computing as a powerful paradigm for model-free prediction of nonlinear dynamical systems spiked in recent years.[5–21] In general, there are three main components in a reservoir computing machine:

an input layer, a high-dimensional neural networked system (the reservoir), and an output layer. The input layer maps the typically low-dimensional, time series data into the high-dimensional state space of the reservoir neural network, and the output layer projects the high-dimensional dynamical evolution of the neural network state back into the low-dimensional time series (readout). Training is administered to adjust the parameters associated with the projection matrix of the output layer to minimize the difference between the output and the true input time series. Because of the nature of the recurrent neural network, the reservoir network structure and parameters are chosen *a priori* and fixed during the training and prediction phases so as to achieve highly efficient learning. In terms of hardware realization, reservoir computing can be implemented using electronic, time-delay autonomous Boolean systems[5] or high-speed photonic devices.[6]

In existing work on reservoir computing-based prediction of chaotic systems, the goal has been to predict the time evolution of the full dynamical variables of interest[7,8,11–14,16,18,20] that encompass both amplitude and phase information. The prediction horizon is determined by the inverse of the largest Lyapunov exponent of the target system. For classic chaotic systems such as the Rössler or Lorenz oscillators, the prediction time with some reasonable accuracy is typically short: about five or six Lyapunov times that correspond to an approximately equal number of oscillation cycles of the underlying system.[7,8,11] Indeed, because of the sensitive dependence on initial conditions (the hallmark of chaos), without state updates to reduce and reset the error in the dynamical variables between the reservoir and true systems from time to time,[21] long-term prediction of the detailed evolution of the dynamical variables as a whole is fundamentally unachievable, regardless of the specific methodology.

The main point of this paper is that there are situations in physics, biology, and chemistry where the phase information is important. In general, any oscillatory dynamical variable can be regarded as containing two components: an amplitude and a phase component, where both are time-dependent and the latter characterizes the ups and downs of the variable. Often, one is interested in the long-term behavior of the ups and downs of some key dynamical variables of physical interest. For example, in ecology, the tendency for the population of a species to go up or down in the future is of critical importance, while the actual number of the species may be less relevant. In a chaotic electronic circuit, one may be interested in the time evolution of a voltage variable but only in terms of its polarity, e.g., in logic circuit design. In such cases, complete information about the time evolution of the dynamical variables is not necessary—only the phase information is needed. Herein, we demonstrate that reservoir computing can be powerful for long-term prediction of the evolution of the phase of chaotic oscillators. The heuristic reason lies in that, for a continuous-time chaotic oscillator, the phase evolution corresponds to the Lyapunov exponent with respect to perturbations along the flow. In an ideal situation, along the flow the length of an infinitesimal vector is unchanged, leading to a zero Lyapunov exponent. When a reservoir computing system has been properly trained, it tends to reproduce the dynamical behaviors and time series of the original dynamical system, i.e., it strives to achieve "synchronization" with the original dynamical system but phase synchronization[22] can be achieved, often much more readily, than complete synchronization. Because of the discrete time nature of the reservoir system, the Lyapunov exponent determining the phase of the system will never be zero but can have a value close to zero when training has been accomplished. The inverse of this near-zero exponent value can be large, giving rise to a large prediction time. Indeed, we find that the typical phase prediction horizon of chaotic oscillators can be on the order of hundreds of natural cycles of the system oscillation. Not only can the phase of a single chaotic oscillator be predicted for a long time, we also demonstrate that distinct levels of phase coherence between a pair of coupled chaotic oscillators can be correctly "sensed" when the variables from both oscillators are input into a single, integrated reservoir. In contrast, an independent input scheme that employs two separated reservoirs, one receiving input from an individual but different oscillators fails to sense phase coherence. This has implications to the design of parallel reservoir systems for predicting large chaotic systems.

II. RESERVOIR COMPUTING SCHEME TO PREDICT PHASE AND SENSE PHASE COHERENCE

The working of reservoir computing to predict the phase of a chaotic oscillator can be briefly described as follows. Let $\mathbf{u}(t)$ be the D_{in}-dimensional input vector constituting sequential measurements of the dynamical variables of oscillator. At time t, $\mathbf{u}(t)$ is mapped into a high-dimensional vector of dimension D_r through a pre-defined input matrix $\mathbf{W}_{in}$ of dimension $D_r \times D_{in}$, where $D_r \gg D_{in}$. The input data vector to the reservoir is $\mathbf{W}_{in} \cdot \mathbf{u}(t)$. The state of the reservoir dynamical network at time t is $\mathbf{r}(t)$, whose time evolution follows the nonlinear updating rule: $\mathbf{r}(t + \Delta t) = \tanh\left[\mathbf{A} \cdot \mathbf{r}(t) + \mathbf{W}_{in} \cdot \mathbf{u}(t)\right]$, where $\mathbf{A}$ is the connection matrix defining the structure and topology of the reservoir network. The updated reservoir state is mapped to an output vector $\mathbf{v}$ of dimension D_{out} as $\mathbf{v}(t + \Delta t) = \mathbf{W}_{out} \cdot \mathbf{f}[\mathbf{r}(t + \Delta t)]$, where the reservoir-to-output matrix $\mathbf{W}_{out}$ has dimension $D_{out} \times D_r$, $D_{out} \ll D_r$ and $\mathbf{f}(\mathbf{r})$ is a nonlinear output function whose components are chosen as $f_i(\mathbf{r}) = r_i$ and $f_i(\mathbf{r}) = r_i^2$ for odd and even indexes i $(i = 1, \ldots, D_r)$, respectively.[7,12]

The elements of the input matrix $\mathbf{W}_{in}$ and those of the reservoir network matrix $\mathbf{A}$ are pre-determined so, together with their sizes, they constitute the set of hyperparameters of the recurrent neural network. Especially, a large number of elements of $\mathbf{W}_{in}$ are set to zero so that every node in the network receives exactly one scalar component of the input vector $\mathbf{u}(t)$ but each such component is connected to $[D_r/D_{in}]$ nodes in the network. The non-zero elements of $\mathbf{W}_{in}$ are chosen randomly from a uniform distribution in the range $[-\sigma, \sigma]$. A common choice of the reservoir network is a large, sparse, directed, or undirected random network whose elements or link weights are appropriately scaled so that the value of its spectral radius (the largest eigenvalue) ρ can be freely adjusted. For a variety of nonlinear, chaotic, and spatiotemporally chaotic systems, the choice of the value of ρ cannot be arbitrary:[18] for a specific system, a unique interval of ρ values exists from which successful training and the subsequent short-term prediction can be guaranteed. To choose the values of all the hyperparameters is a difficult task, and no general rules are available to guide the choices, although optimization methods such as those based on Bayesian estimation can be used for small networks.[17] The elements of the output matrix $\mathbf{W}_{out}$ are set via training based on some standard optimization methods,[7,12] which can be accomplished with a proper amount of the input data **u**.

After the reservoir machine has been properly trained, it can be used to predict the time evolution of the target system by simply closing the system, i.e., to replace the input vector $\mathbf{u}(t)$ by the output vector $\mathbf{v}(t)$ so as to make the whole reservoir machine a self-evolving dynamical system. With the same initial condition as that of the target system, the machine can generate an oscillatory trajectory that stays close to that of the target system for a finite amount of time. The underlying phase variable can be calculated with the standard approach of analytic signals based on the Hilbert transform.[22]

Under what circumstance is a reservoir computing machine able to correctly sense phase coherence between a pair of coupled chaotic oscillators? For near zero coupling, the oscillators are phase incoherent. As the coupling strength is increased, there can be a transition to phase synchronization: there is partial phase coherence prior to the transition and complete phase coherence afterward. There are two possible schemes to sense phase coherence, as shown in Fig. 1. In the first scheme, the inputs from the two oscillators are fed into a single reservoir network, where there is a probability for each node in the network to receive inputs from both oscillators. At the output, the dynamical variables of the two oscillators are separated through training. Their phase variables can then be calculated, enabling correct sensing of phase coherence. We call this the *integrated input scheme*. In the second scheme, as shown in Fig. 1(b), the reservoir consists of two non-connected networks, each receiving input from an individual (but different) oscillator. That is, each node in each network receives inputs from one oscillator only, and the outputs of the networks can be analyzed to determine if there is phase coherence. This is essentially an *independent input*

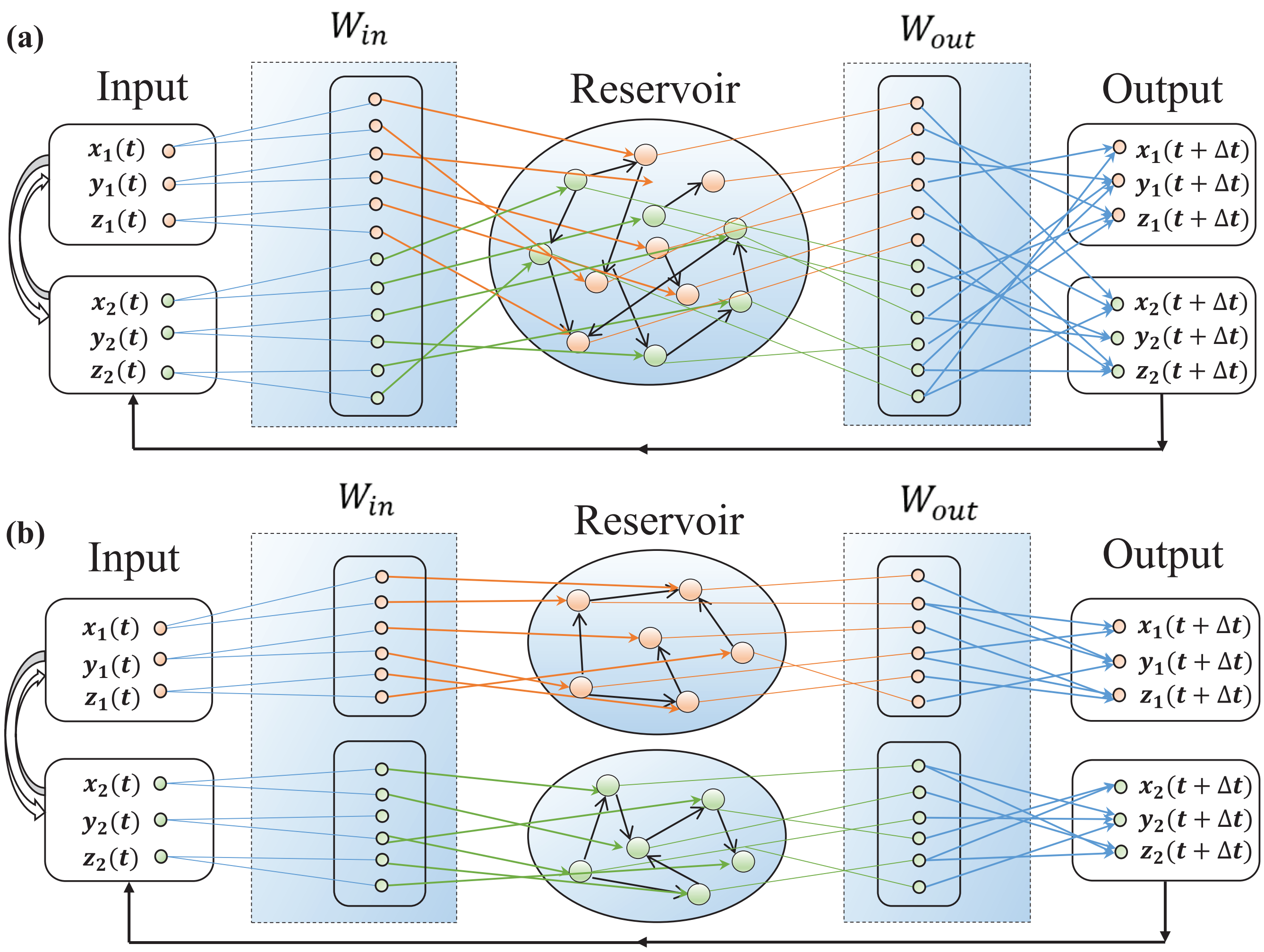

FIG. 1. Two schemes of reservoir computing for sensing phase coherence between a pair of coupled chaotic oscillators. (a) Integrated input scheme: time series from both oscillators are fed into a single reservoir network for training and prediction. (b) Independent input scheme: the reservoir consists of two non-connected networks, each receiving input from an individual oscillator.

scheme. We demonstrate that the integrated input scheme can correctly sense phase coherence while the independent input scheme fails.

III. PREDICTING CHAOTIC PHASE EVOLUTION

We use a classic chaotic oscillator and a chaotic food-web system from ecology to demonstrate that reservoir computing is capable of long-term prediction of phase evolution.

A. Chaotic Rössler oscillator

The system equations[23] are $\dot{x} = -y - z$, $\dot{y} = x + 0.15y$, and $\dot{z} = 0.2 + z(x - 10)$. We generate time series $x(t)$, $y(t)$, and $z(t)$ with integration step size $\Delta t = 0.02$. The values of the parameters for the reservoir computing system are $D_{in} = D_{out} = 3$, $D_r = 500$, $\sigma = 0.1$, network sparsity $s_p = 0.98$, network spectral radius $\rho = 0.01$, and the ridge parameter for optimization is $\eta = 10^{-4}$. After training, the neural machine is closed and becomes a self-evolving dynamical system: it generates $x(t + \Delta t)$, $y(t + \Delta t)$, and $z(t + \Delta t)$ from $x(t)$, $y(t)$, and $z(t)$, respectively. Figure 2(a) presents an example of prediction, where the predicted time series $y(t)$ together with its true evolution are shown, and time is in units of T ($T \approx 6.060$), the average period of oscillation of the Rössler chaotic attractor. It can be seen that the predicted waveform matches with the true waveform but only for about five periods. That is, the reservoir computing machine is able to accurately predict the evolution of the dynamical variable as a whole for about five oscillation cycles. The remarkable observation is, for $t/T > 5$, even though the predicted and true waveforms do not agree with each other, the mismatch occurs only in amplitude. In fact, the phases, i.e., the ups and downs, of the predicted and true waveforms match exactly, indicating temporal phase synchronization between the two waveforms. The phase match occurs for a longer time duration as exemplified in Fig. 2(b). Since the reservoir computing system represents only an approximate replica of the true system, phase match or synchronization cannot last indefinitely: at certain time, a phase mismatch will be inevitable.

To find out how long it takes for a mismatch to occur, we calculate the phase variables of the predicted and true waveforms using the standard Hilbert transform approach.[22] Figure 2(c) shows the absolute phase difference $|\Delta\phi|$ as a function of time for a large number of realizations of the reservoir computing prediction. A phase mismatch occurs when $|\Delta\phi|$ exceeds 2π [indicated by the horizontal line at 2π in Fig. 2(c)]. It can be seen that the phase synchronization time is between about 400 and 800 average cycles of oscillation. A histogram of the phase synchronization time is shown in Fig. 2(d) with the mean time about 600. That is, on average, the reservoir computing machine is capable of accurately predicting the phase of the chaotic oscillator for about 600 cycles. As a reference, we also include a distribution of the prediction time of the actual dynamical variable (in both amplitude and phase)—the narrow peak near zero in Fig. 2(d) with a magnified view in the inset, where the prediction horizon is mostly less than five cycles of oscillation. Thus, if one is content to predict the phase of the dynamical variable, the prediction horizon can be two orders of magnitude longer than that with predicting the actual variable itself!

What is the dynamical mechanism behind the remarkable power of a reservoir computing machine for long-term phase prediction? The answer lies in the Lyapunov exponents. In particular, a well trained reservoir machine can be viewed as a high-dimensional replica of the target chaotic oscillator. For a continuous dynamical system, the evolution of the phase is determined by the null Lyapunov exponent.[22] Strictly, the reservoir computing machine is a discrete time dynamical system that evolves the state over time step Δt. However, for sufficiently small Δt, the reservoir system is approximately continuous in time. Although none of the Lyapunov exponents is exactly zero, the one that is closest to zero determines the phase evolution. The phase prediction time is determined by the inverse of this near zero exponent, which can be large.

To test this mechanism, we calculate the Lyapunov spectrum of the reservoir computing machine. Note that the network in the reservoir has D_r nodes, where $D_r \gg D_{in}$, which is thus a high-dimensional dynamical system with D_r Lyapunov exponents [in Fig. 2, $D_r = 500$]. Figure 2(e) shows, from one predicted trajectory of the reservoir network, all the Lyapunov exponents. Figure 2(f) displays a histogram of the Lyapunov exponent that is closest to zero, whose center cannot be distinguished from zero. Since the average phase prediction time is proportional to the inverse of this exponent, the implication is that the phase of the chaotic oscillator can be accurately predicted for significantly longer time than that with predicting both amplitude and phase. Qualitatively, this is consistent with the result in Fig. 2(d).

From Fig. 2(e), we see that the remaining, vast number of Lyapunov exponents have large negative values. The emergence of the large set of very negative exponents is necessary, which can be understood as follows. For $D_r = 500$, in order for the reservoir system, a 500-dimensional dynamical system to be trained as a replica of the original three-dimensional chaotic Rössler system, the high-dimensional system must be "squeezed" along 497 directions in the 500-dimensional phase space, thereby requiring 497 very negative Lyapunov exponents.

B. A chaotic food-web system

We study the chaotic food-web system given by[24] $\dot{x} = x - 0.2xy/(1 + 0.05x)$, $\dot{y} = -y + 0.2xy/(1 + 0.05x) - yz$, and $\dot{z} = -10(z - 0.006) + yz$, where x, y, and z represent vegetation, the normalized populations of herbivores and predators, respectively. The system exhibits approximately uniform phase evolution but with chaotic amplitude modulations. To obtain the training data, we integrate the system using time step $\Delta t = 0.02$, and the length of the training dataset is 1600. The parameter setting of the reservoir computing system is $D_{in} = D_{out} = 3$, $D_r = 600$, $\sigma = 1$, $s_p = 0.98$, $\rho = 1.37$, and $\eta = 1 \times 10^{-4}$. Figure 3(a) shows the result of predicting the evolution of the herbivores population $y(t)$ in the time interval $0 < t/T < 15$, where $T \approx 6.626$ is the average oscillating period. It can be seen that the reservoir computing machine can accurately predict about 10 cycles of $y(t)$ (in both amplitude and phase) but the predicted phase evolution is accurate in the entire time interval and much beyond, as shown in Fig. 3(b) for $0 < t/T < 30$. Figure 3(c) shows the evolution of the absolute phase difference between the predicted and true variable for 1000 random

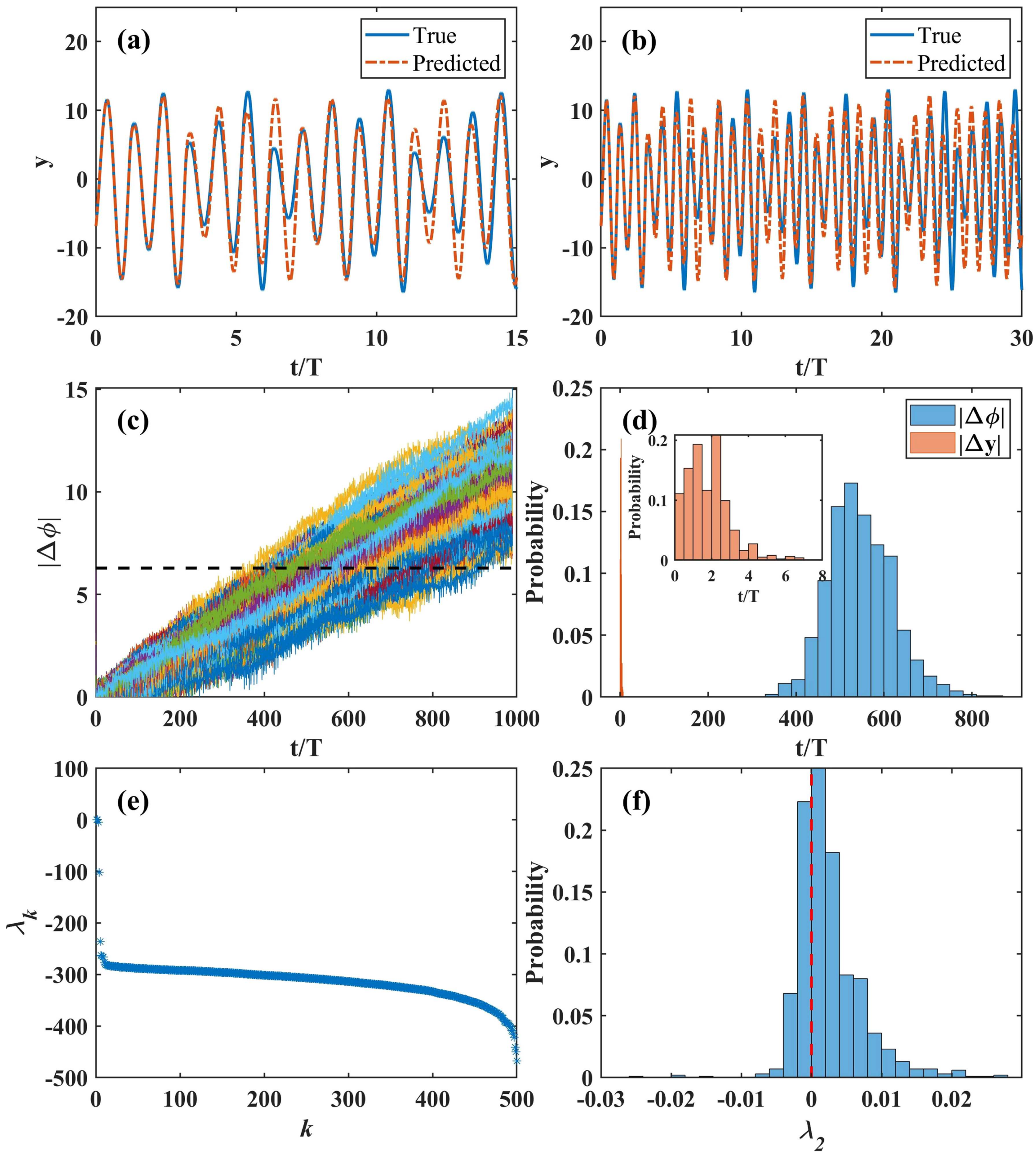

FIG. 2. Using reservoir computing to predict phase evolution of the chaotic Rössler oscillator and the Lyapunov exponents of the reservoir computing system. (a) An example of the predicted and true evolution of the dynamical variable $y(t)$. Time is measured in units of T, the average oscillation period. The predicted and true waveforms match but only for a few cycles of oscillation. In contrast, there is a match in the phases of the two waveforms for the time interval plotted. (b) The phase match occurs in a longer time interval. (c) The absolute phase difference between the predicted and true waveforms vs time for a large number of statistical realizations of prediction run. A mismatch occurs if the absolute phase difference exceeds 2π (the horizontal line). The time horizon for accurate prediction of phase falls in the range between 400 and 800 cycles of oscillation. (d) Statistical distribution of the phase prediction time, where the average prediction horizon is about 600 cycles. In contrast, the time horizon for predicting the dynamical variable itself is less than six cycles, as indicated by the peak near zero and the inset. (e) For a reservoir network of dimension $D_r = 500$, the corresponding 500 Lyapunov exponents. The values of the first three exponents are $(\lambda_1, \lambda_2, \lambda_3) \approx (0.090, 0.002, -9.868)$, which agree with the true values of the three Lyapunov exponents [approximately (0.064, 0.0025, −4.552)] of the chaotic Rössler system in the bulk part, while the values of the remaining 497 exponents are largely negative. (f) Distribution of the exponent value that is closest to zero from 1000 statistical realizations of prediction run. The mean value of this exponent cannot be distinguished from zero, giving a long phase prediction horizon.

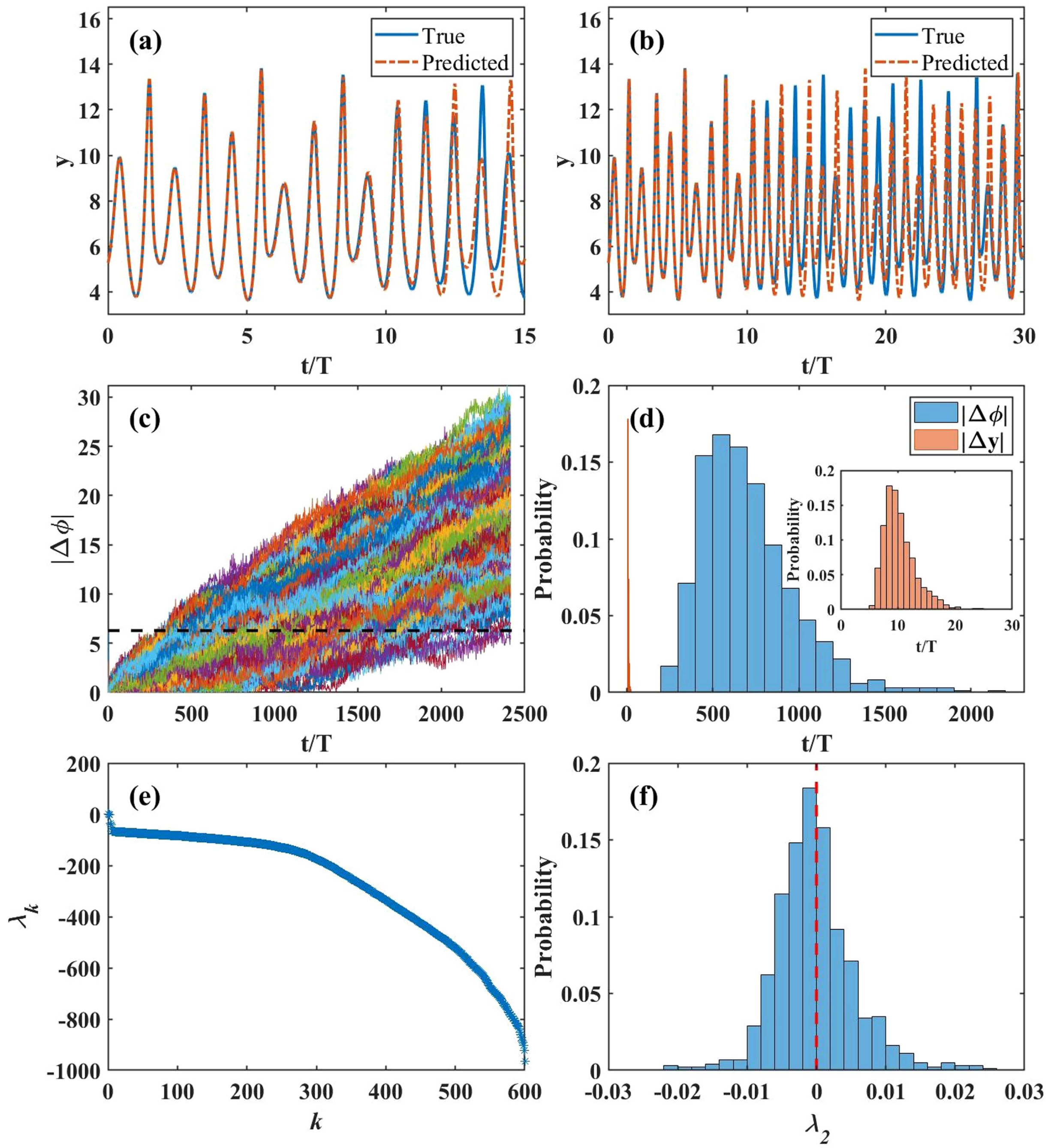

FIG. 3. Prediction of the state evolution of a chaotic food-web system and Lyapunov exponents of the reservoir computing system. (a) and (b) Predicted and true time series of the herbivores population $y(t)$ in the time interval $0 < t/T \leq 15$ and $0 < t/T \leq 30$, respectively. For the dynamical variable itself, its evolution can be accurately predicted for about 10 cycles of oscillation but the phase can be predicted for a much longer time. (c) The absolute phase difference between the predicted and true variable vs time for 1000 random realizations, where the horizontal line denotes $|\Delta\phi| = 2\pi$. (d) Statistical distribution of the phase prediction time with the mean value of approximately 700 cycles. The narrow peak near ten cycles is the distribution of the prediction time of the dynamical variable in both amplitude and phase, which is magnified in the inset. (e) All 600 Lyapunov exponents of the reservoir network system. The values of the first three exponents are $(\lambda_1, \lambda_2, \lambda_3) \approx (0.061, 0.003, -2.870)$, which agree well with those of the food-web system: approximately $(0.060, -2.178 \times 10^{-4}, -2.667)$. The values of the remaining 597 exponents are strongly negative. (f) Distribution of the second Lyapunov exponent, whose mean value cannot be distinguished from zero, giving rise to a long average phase prediction time.

realizations, where the phase prediction time (or the phase synchronization time) is the time when $|\Delta\phi|$ reaches 2π. The histogram of the phase prediction time is shown in Fig. 3(d), whose mean is approximately 700 cycles. In comparison, the distribution of the prediction time of the dynamical variable is centered about 10 cycles, as shown by the narrow peak near zero in Fig. 3(d) and the inset. The phase evolution of the chaotic food-web system can be predicted about two orders of magnitude longer than that with predicting the evolution of the entire variable.

Since the size of the reservoir network is $D_r = 600$, the system has 600 Lyapunov exponents, as shown in Fig. 3(e). As indicated in the caption of Fig. 3, the first three exponents agree with the respective true values of the actual system. The remaining 597 exponents are strongly negative, as the reservoir system has been well trained so its dynamics in the 600-dimensional phase space is effectively reduced to three-dimensional. The distribution of the second Lyapunov exponent is shown in Fig. 3(f), whose mean cannot be statistically distinguished from zero,

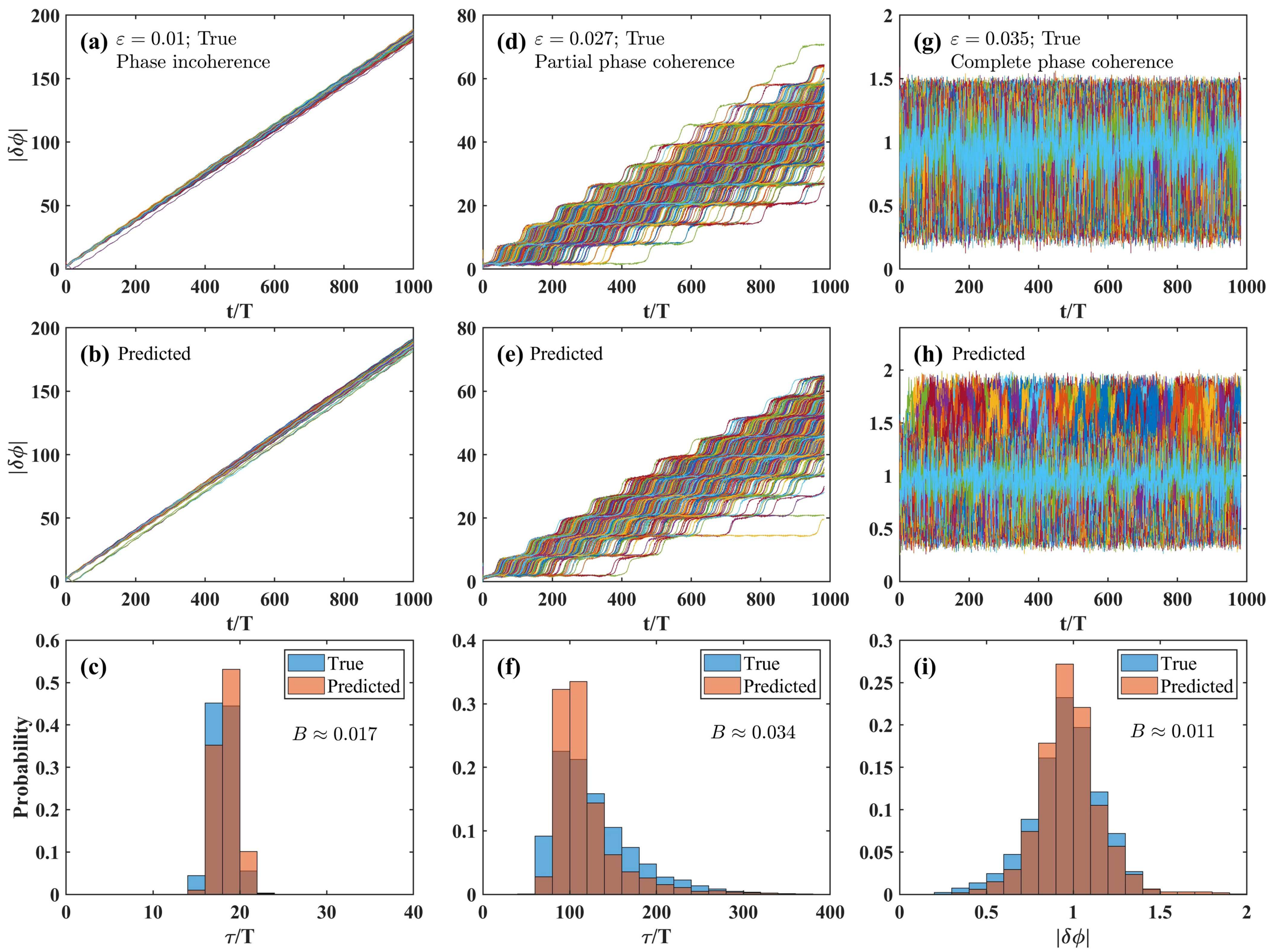

FIG. 4. Reservoir computing based sensing of phase coherence between a pair of coupled chaotic Rössler oscillators with the integrated input scheme. The left, middle, and right columns correspond, respectively, to phase incoherence ($\varepsilon = 0.01$, average oscillation period $T \approx 6.00$), partial phase coherence ($\varepsilon = 0.027$, $T \approx 6.08$), and complete phase coherence ($\varepsilon = 0.035$, $T \approx 6.10$). (a) and (b) True and predicted evolution of the absolute phase difference $|\delta\phi(t)|$ between the two oscillators, respectively, for $\varepsilon = 0.01$. (c) The probability distribution of the phase coherence time, i.e., the time during which $|\delta\phi(t)|$ stays within 2π, for $\varepsilon = 0.01$. (d)–(f) The true, predicted evolution of $|\delta\phi(t)|$, and the probability distribution of the phase coherence time for $\varepsilon = 0.027$, respectively. (g) and (h) The true and predicted evolution of $|\delta\phi(t)|$ for $\varepsilon = 0.035$, respectively. (i) The probability distribution of $|\delta\phi(t)|$ for $\varepsilon = 0.035$, where it is in the phase synchronization regime so that the phase coherence time is infinite. Parameter values of the reservoir computing machine are $D_r = 1000$, $\sigma = 0.1$, $s_p = 0.97$, $\rho = 0.01$ for the left column, $\rho = 0.04$ for the middle and right columns, $\eta = 10^{-4}$, and training time is 1600 with time step $\Delta t = 0.02$. The main message is that reservoir computing with the integrated input scheme is able to accurately sense the degree of phase coherence between a pair of coupled chaotic Rössler oscillators.

giving rise to the observed long time horizon for predicting the phase.

IV. SENSING PHASE COHERENCE WITH RESERVOIR COMPUTING

The results in Sec. III demonstrate that a successfully trained reservoir computing system is able to predict the phase evolution of a chaotic oscillator for a significantly longer time than that with predicting the evolution of the state variable through temporal phase synchronization between the reservoir and the target systems. Phase synchronization of this sort is between the target chaotic system and machine, which is a form of artificial or synthetic phase synchronization. What about genuine phase synchronization among coupled chaotic oscillators? In particular, can a reservoir computing machine *sense* the phase coherence between a pair of coupled chaotic oscillators?

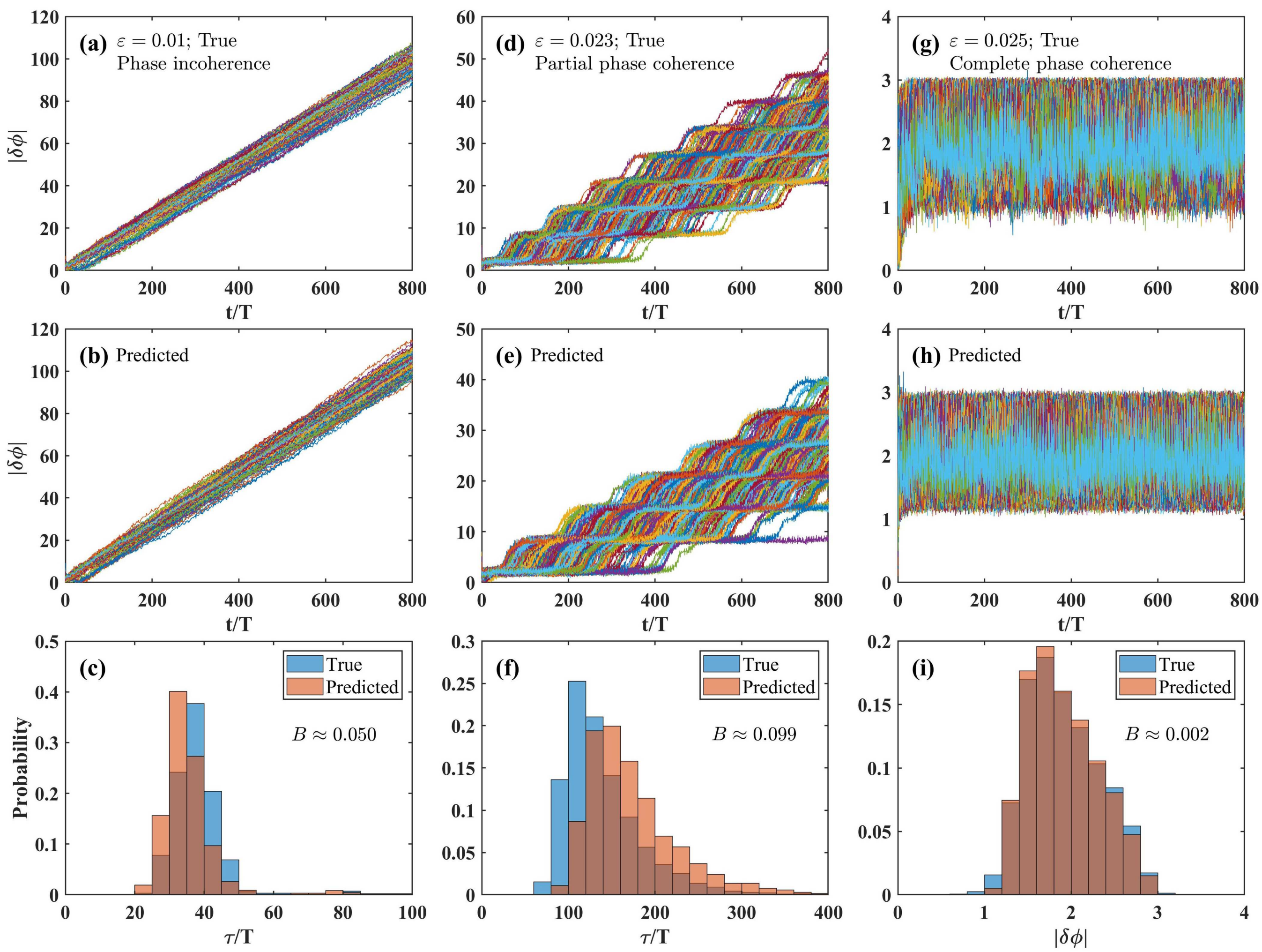

FIG. 5. Reservoir computing based sensing of phase coherence between a pair of spatially coupled chaotic food-web systems with the integrated input scheme. (a) and (b) True and predicted evolution of the absolute phase difference $|\delta\phi(t)|$ between the two oscillators, respectively, for $\varepsilon = 0.01$. (c) The probability distribution of the phase coherence time for $\varepsilon = 0.01$. (d)–(f) The true, predicted evolution of $|\delta\phi(t)|$ and the probability distribution of the phase coherence time for $\varepsilon = 0.023$, respectively. (g) and (h) The true and predicted evolution of $|\delta\phi(t)|$ for $\varepsilon = 0.025$, respectively. (i) The probability distribution of $|\delta\phi(t)|$ for $\varepsilon = 0.025$ in the phase synchronization regime. Parameter values of the reservoir computing system are $D_r = 800$, $\sigma = 1.0$, $s_p = 0.98$, $\rho = 1.29, 0.97, 1.13$ for the left ($\varepsilon = 0.01$, $T \approx 6.42$), middle ($\varepsilon = 0.023$, $T \approx 6.60$), and right ($\varepsilon = 0.025$, $T \approx 6.65$) columns, $\eta = 10^{-4}$, and training time is 1600 with time step $\Delta t = 0.02$. For the probability distributions in the bottom row, the respective Bhattacharyya distances for the left, middle, and right panels are $B \approx 0.077$, $B \approx 0.100$, and $B \approx 0.003$. The message is that reservoir computing with the integrated input scheme is capable of accurately sensing the degree of phase coherence between a pair of coupled chaotic food-web systems.

For a pair of coupled, slightly non-identical chaotic oscillators, phase coherence can gradually emerge as the coupling parameter, denoted as ε, is increased from zero.[22] Depending on the degree of phase coherence, there are three regimes of ε values of interest: phase incoherence, partial phase coherence, and complete phase coherence. In particular, let ε_c be the critical coupling value at which a transition to phase synchronization or complete phase coherence occurs. For $\varepsilon \gtrsim 0$, the phase evolution of the two oscillators is incoherent in that the absolute phase difference $|\delta\phi(t)|$ increases linearly with time. For $\varepsilon \lesssim \varepsilon_c$, partial phase coherence sets in, which is manifested as the occurrences of intermittent plateaus in which $|\delta\phi(t)|$ fluctuates between zero and 2π, where the distribution of the time duration of the plateaus follows a scaling law[25,26] characteristic of superpersistent chaotic transients.[27–34] Complete phase coherence arises for $\varepsilon > \varepsilon_c$, where $|\delta\phi(t)|$ remains bounded within 2π indefinitely. Our question is whether reservoir computing is capable of distinguishing the three regimes of different levels of phase coherence. In the following, we address this question by testing two possible configurations of reservoir computing: integrated and independent input schemes, as schematically illustrated in Fig. 1.

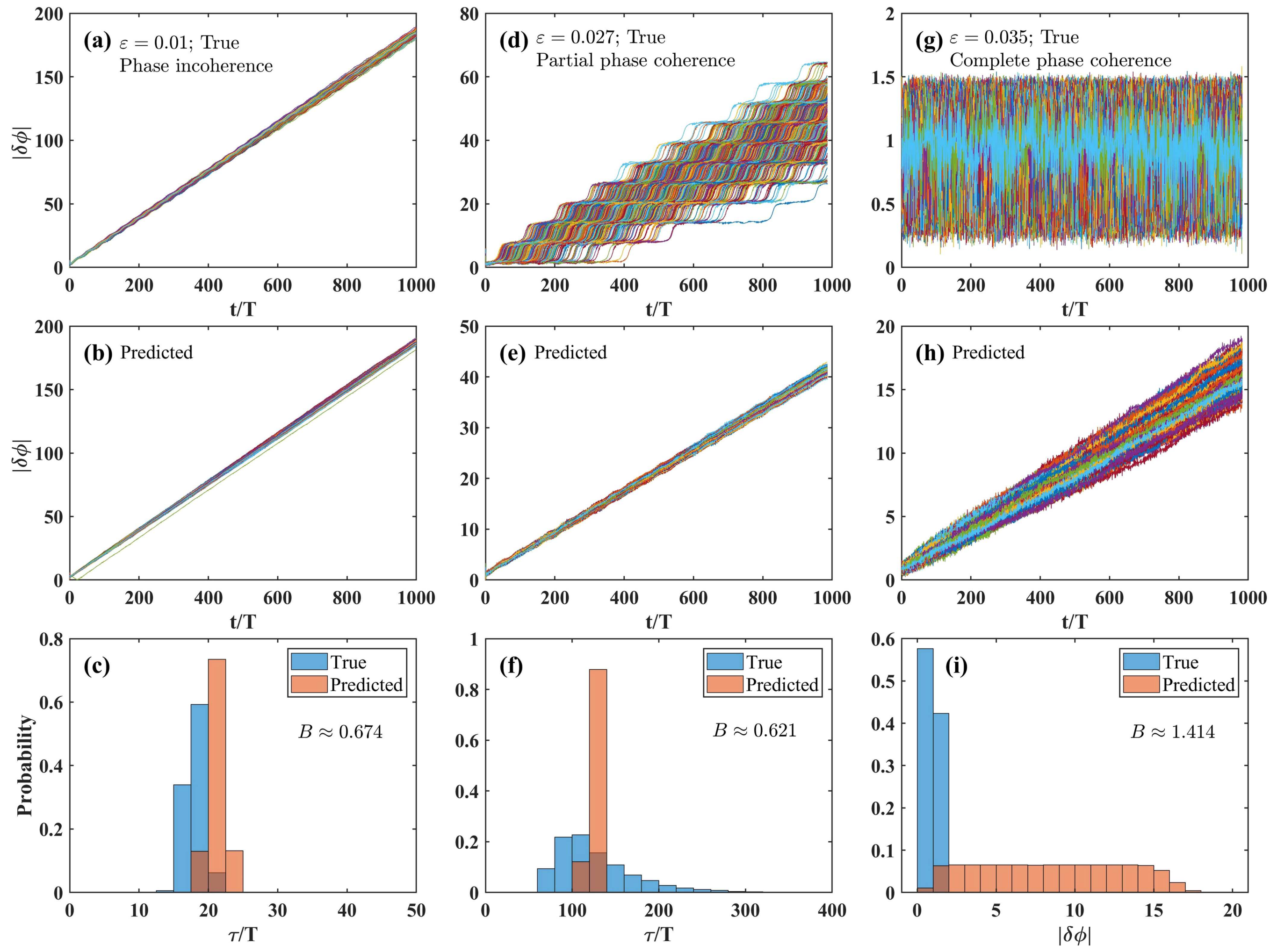

FIG. 6. Failure of the independent input scheme to sense phase coherence for the coupled chaotic Rössler system. (a) and (b) True and predicted evolution of the absolute phase difference $|\delta\phi(t)|$ between the two oscillators, respectively, for $\varepsilon = 0.01$. (c) The probability distribution of the phase coherence time for $\varepsilon = 0.01$. (d)–(f) The true, predicted evolution of $|\delta\phi(t)|$ and the probability distribution of the phase coherence time for $\varepsilon = 0.027$, respectively. (g) and (h) The true and predicted evolution of $|\delta\phi(t)|$ for $\varepsilon = 0.035$, respectively. (i) The probability distribution of $|\delta\phi(t)|$ for $\varepsilon = 0.035$ in the phase synchronization regime. The two non-connected reservoirs have the same set of parameter values: $D_r = 500$, $\sigma = 0.1$, $s_p = 0.98$, $\rho = 0.01$, and $\eta = 10^{-4}$, and training time is 1600 with time step $\Delta t = 0.02$. Both direct visualization (the top and middle rows) and quantitative assessment in terms of the Bhattacharyya distance (the bottom row) indicate the failure.

A. Integrated input scheme

A pair of coupled chaotic Rössler oscillators. The system equations are

$$\begin{aligned}\dot{x_{1,2}} &= -\omega_{1,2}y_{1,2} - z_{1,2} + \varepsilon(x_{2,1} - x_{1,2}),\\ \dot{y_{1,2}} &= \omega_{1,2}x_{1,2} + 0.15y_{1,2},\\ \dot{z_{1,2}} &= 0.2 + z_{1,2}(x_{1,2} - 10),\end{aligned}$$

where $\omega_{1,2} = 1.0 \pm \Delta\omega$ with the frequency mismatch $\Delta\omega = 0.015$. The three columns of Fig. 4 show, respectively, three distinct cases of phase coherence. Especially, Figs. 4(a)–4(c) show, for $\varepsilon = 0.01$ in the phase incoherence regime, the evolution of the actual absolute phase difference $|\delta\phi(t)|$ between the two chaotic oscillators, the predicted evolution, and the probability distributions of the true and predicted time duration in which $|\delta\phi(t)|$ is below 2π, respectively. Comparing the results in Figs. 4(a) and 4(b), we see that the reservoir computing machine with the integrated input scheme can accurately sense that the two coupled oscillators are not phase coherent at a quantitative level. As shown in Fig. 4(c), the time duration for maintaining temporal phase coherence is short, and the two distributions are close to each other. The similarity between the two probability distributions

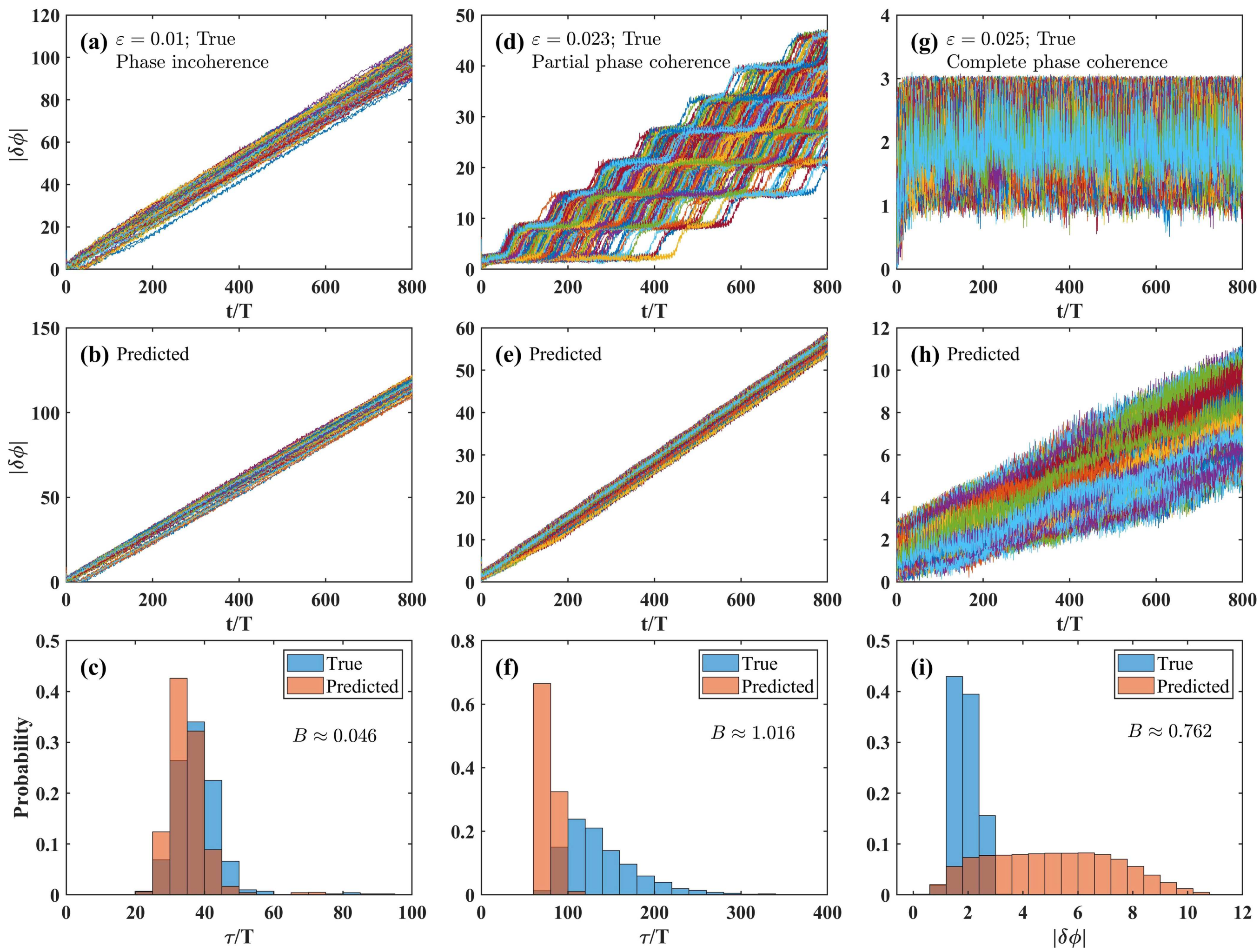

FIG. 7. Failure of the independent input scheme to sense phase coherence for the coupled chaotic food-web system. (a) and (b) True and predicted evolution of the absolute phase difference $|\delta\phi(t)|$ between the two oscillators, respectively, for $\varepsilon = 0.01$. (c) The probability distribution of the phase coherence time for $\varepsilon = 0.01$. (d)–(f) The true, predicted evolution of $|\delta\phi(t)|$ and the probability distribution of the phase coherence time for $\varepsilon = 0.023$, respectively. (g) and (h) The true and predicted evolution of $|\delta\phi(t)|$ for $\varepsilon = 0.025$, respectively. (i) The probability distribution of $|\delta\phi(t)|$ for $\varepsilon = 0.025$ in the phase synchronization regime. The two independent reservoir networks are slightly different in their values of the spectral radius: $(\rho_1, \rho_2) = (0.4, 0.5)$ for the left column, $(\rho_1, \rho_2) = (0.41, 0.69)$ for the middle column, and $(\rho_1, \rho_2) = (0.41, 0.41)$ for the right column. Other parameter values of the reservoirs are: $D_r = 600$, $\sigma = 1$, $s_p = 0.98$, and $\eta = 10^{-4}$.

can be measured by the Bhattacharyya distance[35] B, where $B = 0$ indicates that the two distributions are exactly identical. For the distributions in Fig. 4(c), we have $B \approx 0.017$, indicating a high degree of similarity. Essentially identical results are obtained for the partial and complete coherence regimes, as shown in Figs. 4(d)–4(f) and 4(g)–4(i), respectively, where the B values are $B \approx 0.034$ and $B \approx 0.011$. (Note that, in the regime of complete phase coherence, the phase synchronization time is infinite, so the distribution in Fig. 4(i) is one on the absolute phase difference.) The results in Figs. 4(a)–4(i) are thus strong evidence that the reservoir computing machine with the integrated input scheme is capable of sensing the degree of phase coherence between a pair of coupled chaotic oscillators.

A pair of coupled chaotic food-web systems. The system equations are[24]

$$
\begin{aligned}
\dot{x}_{1,2} &= x_{1,2} - 0.2x_{1,2}y_{1,2}/(1 + 0.05x_{1,2}),\\
\dot{y}_{1,2} &= -b_{1,2}y_{1,2} + 0.2x_{1,2}y_{1,2}/(1 + 0.05x_{1,2}) - y_{1,2}z_{1,2} + \varepsilon(y_{2,1} - y_{1,2}),\\
z_{1,2} &= -10(z_{1,2} - 0.006) + y_{1,2}z_{1,2} + \varepsilon(z_{2,1} - z_{1,2}),
\end{aligned}
$$

where $b_1 = 1.1$ and $b_2 = 1.055$ so that the two food-web systems are not identical. This set of equations describes the population dynamics of two spatially adjacent patches with mutual dispersal of herbivores and predators between the two patches. With the integrated input scheme, we obtain essentially the same result as for the case of a pair of coupled Rössler oscillators that reservoir computing is capable of accurately sensing phase coherence and distinguishing regimes of characteristically distinct phase coherence, as shown in Figs. 5(a)–5(i) [with the same legends as in Fig. 4(a)–4(i)].

B. Independent input scheme

The independent input scheme is illustrated in Fig. 1(b), where the dynamical variables of each oscillator are input into its own reservoir network, and the two networks are not connected. Each network is well trained with sufficient amount of input data from the respective oscillator. Figure 6 shows the results with the same coupled chaotic Rössler system as in Fig. 4. With the same legends as in Figs. 4, Fig. 6 demonstrates complete failure of the reservoir computing system to sense phase coherence. In particular, for $\varepsilon = 0.01$, where there is no phase coherence, while the true and machine predicted $|\delta\phi(t)|$ both exhibit a monotonically increasing behavior [Figs. 6(a) and 6(b), respectively], the Bhattacharyya distance between the true and machine generated probability distributions of the temporal phase coherence time is large: $B \approx 0.674$. For $\varepsilon = 0.027$, the true system exhibits partial phase coherence with $|\delta\phi(t)|$ exhibiting intermittent plateaus [Fig. 6(d)]. However, the reservoir machine does not give any such apparent plateau [Fig. 6(e)]. In this case, the Bhattacharyya distance is still quite large: $B \approx 0.622$. For $\varepsilon = 0.035$, where there is complete phase coherence with $|\delta\phi(t)| < 2\pi$ for all time in the true system [Fig. 6(g)], the reservoir system generates a monotonically increasing behavior [Fig. 6(h)]. In this case, the Bhattacharyya distance between the true and machine generated probability distributions of the phase change over a certain time interval is even larger: $B \approx 1.414$. Results with the coupled chaotic food-web system are characteristically the same, as shown in Fig. 7.

V. DISCUSSION

Reservoir computing machines have been demonstrated to be able to predict the state evolution of chaotic systems for several Lyapunov time.[7,8,11–14,16,18,20] While the actual prediction time varies among different chaotic systems, for a typical chaotic system such as the Rössler oscillator, this time corresponds to approximately several cycles of oscillation. The fundamental reason for this relatively short time horizon of prediction lies in the very nature of chaos: sensitive dependence on initial conditions. In particular, a well trained reservoir system is effectively a high-dimensional replica of the target system with inevitable differences between the two systems. Due to chaos, the difference grows exponentially at the rate determined by the largest positive Lyapunov exponent. Without state updating with real data to correct the errors from time to time, long-term prediction of the state evolution is unlikely.[21]

For any chaotic oscillator, its state naturally can be represented by amplitude and phase variables. The main point of our work is that if we relax the prediction criterion by being content to predict the evolution of the phase variable only, then the prediction horizon with reservoir computing can be much longer, typically at least two orders of magnitude longer than that with predicting the evolution of the entire state. The basic reason is that the phase evolution is determined by the null Lyapunov exponent of the system. For its replica realized by reservoir computing, a strictly zero exponent does not exist but its value can be quite close to zero, so the phase prediction time, which is proportional to the inverse of this exponent, can be long. This result should be encouraging for situations in physical, biological, and applied sciences where the ups and downs of the system are more important than the amplitude variations of the state variables.

We have also studied the ability for reservoir computing to sense phase coherence between a pair of coupled chaotic oscillators. The main finding is that the integrated input scheme, in which data from both oscillators are sent to a single, large reservoir and are thus completely mixed, has the power to discern different degrees of phase coherence, e.g., phase incoherence, partial and complete phase coherence. However, the independent input scheme, in which the single, large reservoir is replaced by two smaller and unconnected reservoirs, each receiving inputs from one oscillator, fails to correctly sense phase coherence. The finding may have implications in using parallel reservoirs to predict large chaotic systems, e.g., a system of a large number of coupled chaotic oscillators or a spatially extended chaotic system of a large size. In particular, to predict the state evolution of a very large spatiotemporal chaotic system, if a single reservoir is used, its size needs to be so large that computations are infeasible. It was demonstrated that a number of small, parallel reservoirs, each receiving data from a spatially localized region with the whole spatial domain covered by all the reservoirs altogether, can be used for the prediction task.[12,15] While short-term prediction is possible with the parallel reservoir scheme, our result implies a potential difficulty for this scheme to sense the collective dynamical behavior (e.g., phase coherence) of the whole system.

We discuss three general issues concerning reservoir computing based prediction of chaotic systems.

A. Prediction performance of reservoir computing for different parameter values of the target dynamical system

For a fixed set of parameter values of the target nonlinear dynamical system, insofar as the reservoir computing machine is trained based on time series obtained from the same set of parameter values and the machine's hyperparameter values are determined via some standard optimization procedure as described in Sec. II, the prediction accuracy, time interval and horizon change little with respect to different initial conditions. For the configuration of the reservoir computing machine studied in this paper, when there is a change in the parameter values of the target system, it is only necessary to re-train the machine using time series from the new set of parameter values and to re-optimize the hyperparameter values. Empirically, it is always possible to find a set of optimal hyperparameter values for different dynamical systems for a fixed structure of the reservoir neural network.

B. Effects of network structure on training efficiency and prediction accuracy

In our work, the reservoir network structure was chosen to be random. A question is whether it is possible to optimize the reservoir network structure to improve the training efficiency and prediction accuracy. At the present, there is no general, theoretically justified answer to this question, but some recent studies have provided insights. For example, in Ref. 18, two types of network topology, random and small-world, were studied for predicting spatiotemporal dynamical systems with the finding that the network structure has no significant effects on the training efficiency and prediction accuracy. In fact, both directed and undirected networks led to similar performances, provided that the hyperparameter values are chosen properly. In Refs. 17 and 36, reservoir computing with a simple cyclic network structure as the standard echo state machine was studied for prediction. In another recent work,[20] similar performances were obtained of reservoir computing with both symmetric and completely random networks. These results are encouraging from the point of view of experimental implementation of reservoir computing, as they suggest the possibility of using some simple network structure to achieve the desired prediction performance.

C. Intuitive explanation for the failure of the independent input scheme to sense phase coherence

While there has been no solid theoretical explanation for the results of machine learning in general, the success of the integrated input scheme and the failure of the independent input scheme to sense phase coherence may be intuitively understood, as follows. Note that phase coherence in the weakly coupling regime in a system of coupled non-identical chaotic oscillators is an aspect of the collective dynamics of the whole system. With the integrated input scheme through training, the reservoir computing system is able to learn well the collective dynamics of the target system as a whole, even when the coupling is weak. However, with the independent input scheme, each reservoir system learns the behavior of the specific oscillator that provides the input time series for training. As a result, while each reservoir is able to independently predict the evolution of its target oscillator for a finite amount of time, the time is typically too short for the reservoir to be "aware" of the behaviors of the other oscillators due to the weak coupling, leading to the failure to detect phase coherence.

ACKNOWLEDGMENTS

Y.-C.L. and J.J. would like to acknowledge support from the Vannevar Bush Faculty Fellowship program sponsored by the Basic Research Office of the Assistant Secretary of Defense for Research and Engineering and funded by the Office of Naval Research through Grant No. N00014-16-1-2828. C.Z. is supported by the China Scholarship Council under Grant No. 201806870036. S.-X.Q. is supported by the National Natural Science Foundation of China (NNSFC) through Grant No. 11975144.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] H. Jaeger, "The 'echo state' approach to analysing and training recurrent neural networks-with an erratum note," German National Research Center for Information Technology GMD Technical Report No. 148, 2001, p. 13.
[2] W. Mass, T. Nachtschlaeger, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," Neural Comput. **14**, 2531–2560 (2002).
[3] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," Science **304**, 78–80 (2004).
[4] G. Manjunath and H. Jaeger, "Echo state property linked to an input: Exploring a fundamental characteristic of recurrent neural networks," Neural Comput. **25**, 671–696 (2013).
[5] N. D. Haynes, M. C. Soriano, D. P. Rosin, I. Fischer, and D. J. Gauthier, "Reservoir computing with a single time-delay autonomous Boolean node," Phys. Rev. E **91**, 020801 (2015).
[6] 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," Phys. Rev. X **7**, 011015 (2017).
[7] 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).
[8] 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).
[9] T. Duriez, S. L. Brunton, and B. R. Noack, *Machine Learning Control-Taming Nonlinear Dynamics and Turbulence* (Springer, 2017).
[10] Z. Lu, B. R. Hunt, and E. Ott, "Attractor reconstruction by machine learning," Chaos **28**, 061104 (2018).
[11] J. Pathak, A. Wilner, R. Fussell, S. Chandra, B. Hunt, M. Girvan, Z. Lu, and E. Ott, "Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model," Chaos **28**, 041101 (2018).
[12] 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).
[13] T. L. Carroll, "Using reservoir computers to distinguish chaotic signals," Phys. Rev. E **98**, 052209 (2018).

[14] K. Nakai and Y. Saiki, "Machine-learning inference of fluid variables from data using reservoir computing," Phys. Rev. E **98**, 023111 (2018).
[15] Z. S. Roland and U. Parlitz, "Observing spatio-temporal dynamics of excitable media using reservoir computing," Chaos **28**, 043118 (2018).
[16] 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).
[17] A. Griffith, A. Pomerance, and D. J. Gauthier, "Forecasting chaotic systems with very low connectivity reservoir computers," Chaos **29**, 123108 (2019).
[18] J. Jiang and Y.-C. Lai, "Model-free prediction of spatiotemporal dynamical systems with recurrent neural networks: Role of network spectral radius," Phys. Rev. Res. **1**, 033056 (2019).
[19] P. R. Vlachas, J. Pathak, B. R. Hunt, T. P. Sapsis, M. Girvan, E. Ott, and P. Koumoutsakos, "Forecasting of spatio-temporal chaotic dynamics with recurrent neural networks: A comparative study of reservoir computing and backpropagation algorithms," arXiv:1910.05266 (2019).
[20] T. L. Carroll and L. M. Pecora, "Network structure effects in reservoir computers," Chaos **29**, 083130 (2019).
[21] H. Fan, J. Jiang, C. Zhang, X. Wang, and Y.-C. Lai, "Long-term prediction of chaotic systems with machine learning," Phys. Rev. Res. **2**, 012080 (2020).
[22] M. G. Rosenblum, A. S. Pikovsky, and J. Kurths, "Phase synchronization of chaotic oscillators," Phys. Rev. Lett. **76**, 1804–1807 (1996).
[23] O. E. Rössler, "An equation for continuous chaos," Phys. Lett. A **57**, 397–398 (1976).
[24] B. Blasius, A. Huppert, and L. Stone, "Complex dynamics and phase synchronization in spatially extended ecological systems," Nature **399**, 354 (1999).
[25] V. Andrade, R. L. Davidchack, and Y.-C. Lai, "Noise scaling of phase synchronization of chaos," Phys. Rev. E **61**, 3230–3233 (2000).
[26] V. Andrade and Y.-C. Lai, "Super persistent chaotic transients in physical systems: Effect of noise on phase synchronization of coupled chaotic oscillators," Int. J. Bifurcation Chaos **11**, 2607–2619 (2001).
[27] C. Grebogi, E. Ott, and J. A. Yorke, "Fractal basin boundaries, long-lived chaotic transients, and unstable-unstable pair bifurcation," Phys. Rev. Lett. **50**, 935–938 (1983).
[28] C. Grebogi, E. Ott, and J. Yorke, "Super persistent chaotic transients," Ergod. Theory Dyn. Syst. **5**, 341–372 (1985).
[29] Y.-C. Lai, C. Grebogi, J. A. Yorke, and S. Venkataramani, "Riddling bifurcation in chaotic dynamical systems," Phys. Rev. Lett. **77**, 55–58 (1996).
[30] Y. Do and Y.-C. Lai, "Superpersistent chaotic transients in physical space: Advective dynamics of inertial particles in open chaotic flows under noise," Phys. Rev. Lett. **91**, 224101 (2003).
[31] Y. Do and Y.-C. Lai, "Extraordinarily superpersistent chaotic transients," Europhys. Lett. **67**, 914–920 (2004).
[32] Y. Do and Y.-C. Lai, "Scaling laws for noise-induced superpersistent chaotic transients," Phys. Rev. E **71**, 046208 (2005).
[33] O. D'Huys, J. Lohmann, N. D. Haynes, and D. J. Gauthier, "Super-transient scaling in time-delay autonomous boolean network motifs," Chaos **26**, 094810 (2016).
[34] J. Lohmann, O. D'Huys, N. D. Haynes, E. Schöll, and D. J. Gauthier, "Transient dynamics and their control in time-delay autonomous boolean ring networks," Phys. Rev. E **95**, 022211 (2017).
[35] T. Kailath, "The divergence and Bhattacharyya distance measures in signal selection," IEEE Trans. Commun. Technol. **15**, 52–60 (1967).
[36] A. Rodan and P. Tino, "Minimum complexity echo state network," IEEE Trans. Neural Netw. **22**, 131–144 (2010).