Digital twins of nonlinear dynamical systems

Ling-Wei Kong[a], Yang Weng[a], Bryan Glaz[b], Mulugeta Haile[b], and Ying-Cheng Lai[a,c,1]

[a]School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, Arizona 85287, USA; [b]Vehicle Technology Directorate, CCDC Army Research Laboratory, 2800 Powder Mill Road, Adelphi, MD 20783-1138, USA; [c]Department of Physics, Arizona State University, Tempe, Arizona 85287, USA

This manuscript was compiled on October 26, 2022

We articulate the design imperatives for creating machine-learning based digital twins for nonlinear dynamical systems subject to external driving, which can be used to monitor the "health" of the target system and to anticipate its possible future collapse in different scenarios. The digital twins are tested on prototypical systems from optics, ecology, and climate, where the respective specific examples are a driven chaotic CO_2 laser system, a model of phytoplankton subject to seasonality, and the driven Lorenz-96 climate network. We demonstrate that, with a single or parallel reservoir computers as the platform, the digital twins are capable of a variety of challenging forecasting and monitoring tasks. In particular, a digital twin created according to our design imperatives has the following capabilities: (1) extrapolating the dynamics of the target system to certain parameter regimes that it has never experienced before, (2) making continual forecasting and monitoring with sparse real-time updates under non-stationary external driving, (3) inferring the existence of hidden variables in the target system and accurately reproducing/predicting their dynamical evolution into the future, (4) adapting to external driving of different waveform, and (5) extrapolating the global bifurcation behaviors to network systems of some different sizes. These features make our digital twins appealing in significant applications such as monitoring the health of critical systems of current interest and forecasting their potential collapse induced by environmental changes or perturbations. Such systems can be an infrastructure, an ecosystem, or a regional climate system.

nonlinear dynamics | digital twins | machine learning

The concept of digital twins originated from aerospace engineering for aircraft structural life prediction (1). In general, a digital twin can be used for predicting dynamical systems and generating solutions of emergent behaviors that can potentially be catastrophic (2). Digital twins have attracted a great deal of attention from a wide range of fields (3) including medicine and health care (4, 5). For example, the idea of developing medical digital twins in viral infection through a combination of mechanistic knowledge, observational data, medical histories, and artificial intelligence has been proposed recently (6), which can potentially lead to a powerful addition to the existing tools to combat future pandemics. In a more dramatic development, the European Union plans to fund the development of digital twins of Earth for its green transition (7, 8).

The physical world is nonlinear. Many engineering systems, such as complex infrastructural systems, are governed by nonlinear dynamical rules, too. In nonlinear dynamics, various bifurcations leading to chaos and system collapse can take place (9). For example, in ecology, environmental deterioration caused by global warming can lead to slow parameter drift towards chaos and species extinction (10, 11). In an electrical power system, voltage collapse can occur after a parameter shift that lands the system in transient chaos (12). The various climate systems in different geographic regions of the world are also nonlinear and the emergent catastrophic behaviors as the result of increasing human activities are of grave concern. In all these cases, it is of interest to develop a digital twin of the system of interest to monitor its "health" in real time as well as for predictive problem solving in the sense that, if the digital twin indicates a possible system collapse in the future, proper control strategies should and can be devised and executed in time to prevent the collapse.

What does it take to create a digital twin for a nonlinear dynamical system? For natural and engineering systems, there are two general approaches: one is based on mechanistic knowledge and another is based on observational data. In principle, if the detailed physics of the system is well understood, it should be possible to construct a digital twin through mathematical modeling. However, there are two difficulties associated with this modeling approach. First, a real-world system can be high-dimensional and complex, preventing the rules governing its dynamical evolution from being known at a sufficiently detailed level. Second, the hallmark of chaos is sensitive dependence on initial conditions. Because no mathematical model of the underlying physical system can be perfect, the small deviations and high dimensionality of the system coupled with environmental disturbances can cause the model predictions of the future state of the system to be inaccurate and completely irrelevant (13, 14). These difficulties motivate the proposition that data-based approach can have advantages in many realistic scenarios and a viable method to develop a digital twin is through data. While in certain cases, approximate system equations can be found from data through sparse optimization (15–17), the same difficulties with the modeling approach arise. These considerations have led us to

Significance Statement

Digital twins have attracted a great deal of recent attention from a wide range of fields. A basic requirement for digital twins of nonlinear dynamical systems is the ability to predict the system evolution and generate potentially catastrophic emergent behaviors, providing early warnings. As an essential design imperative, one wishes to develop digital twins of critical systems of interest for system "health" monitoring in real time and for predictive problem solving in the sense that, if the digital twin forecasts a possible system collapse in the future, an optimal control strategy can be devised and executed as early intervention to prevent the collapse. This article develops a class of reservoir computing based digital twins for low- and high-dimensional chaotic systems.

Please provide details of author contributions here.

Please declare any competing interests here.

[1]To whom correspondence should be addressed. E-mail: Ying-Cheng.Lai@asu.edu

exploit machine learning to create digital twins for nonlinear dynamical systems.

Given a nonlinear dynamical system, its digital twin is also a dynamical system, rendering appropriate exploitation of recurrent neural networks that can be designed to generate self-dynamical evolution with memory. In this regard, reservoir computers (RC) (18–20) that have been extensively studied in recent years (21–42) provide a starting point, which can be trained from observational data to generate closed-loop dynamical evolution that follows the evolution of the target system for a finite amount of time. Another advantage of RC is that no back-propagation is needed for optimizing the parameters - only a linear regression is required in the training so it is computationally efficient. A common situation is that the target system is subject to external driving, such as a driven laser, a regional climate system, or an ecosystem under external environmental disturbances. Accordingly, the digital twin must accommodate a mechanism to control or steer the dynamics of the RC neural network to account for the external driving. Introducing a control mechanism distinguishes our work from existing ones in the literature of RC as applied to nonlinear dynamical systems. Of particular interest is whether the collapse of the target chaotic system can be anticipated from the digital twin. The purpose of this paper is to demonstrate that the digital twin so created can accurately produce the bifurcation diagram of the target system and faithfully mimic its dynamical evolution from a statistical point of view. The digital twin can then be used to monitor the present and future "health" of the system. More importantly, with proper training from observational data the twin can reliably anticipate system collapses, providing early warnings of potentially catastrophic failures of the system.

More specifically, using three prototypical systems from optics, ecology, and climate, respectively, we demonstrate that the RC based digital twins developed in this paper solve the following challenging problems: (1) extrapolation of the dynamical evolution of the target system into certain "uncharted territories" in the parameter space, (2) long-term continual forecasting of nonlinear dynamical systems subject to non-stationary external driving with sparse state updates, (3) inference of hidden variables in the system and accurate prediction of their dynamical evolution into the future, (4) adaptation to external driving of different waveform, and (5) extrapolation of the global bifurcation behaviors of network systems to some different sizes. These features make our digital twins appealing in applications.

Results

For clarity, we present results on the digital twin for a prototypical nonlinear dynamical systems with adjustable phase-space dimension: the Lorenz-96 climate network model (43). In the Supporting Material (SM), we present two additional examples: a chaotic laser (SM 1) and a driven ecological system (SM 2), together with a number of pertinent issues.

A low-dimensional Lorenz-96 climate network and its digital twin. The Lorenz-96 system (43) is an idealized atmospheric climate model. Mathematically, the toy climate system is described by m coupled first-order nonlinear differential equations subject to external periodic driving $f(t)$:

$$\frac{dx_i}{dt} = x_{i-1}(x_{i+1} - x_{i-2}) - x_i + f(t), \qquad [1]$$

where $i = 1, \ldots, m$, is the spatial index. Under the periodic boundary condition, the m nodes constitute a ring network, where each node is coupled to three neighboring nodes. To be concrete, we set $m = 6$ (more complex high-dimensional cases are treated below). The driving force is sinusoidal with a bias F: $f(t) = A\sin(\omega t) + F$. We fix $\omega = 2$ and $F = 2$, and use the forcing amplitude A as the bifurcation parameter. For relatively large values of A, the system exhibits chaotic behaviors, as exemplified in Fig. 1(A1) for $A = 2.2$. Quasi-periodic dynamics arise for smaller values of A, as exemplified in Fig. 1(A2). As A decreases from a large value, a critical transition from chaos to quasi-periodicity occurs at $A_c \approx 1.9$. We train the digital twin with time series from four values of A, all in the chaotic regime: $A = 2.2, 2.6, 3.0$, and 3.4. The size of the random reservoir network is $D_r = 1,200$. For each value of A in the training set, the training and validation lengths are $t = 2,500$ and $t = 12$, respectively, where the latter corresponds to approximately five Lyapunov times. The warming-up length is $t = 20$ and the time step of the reservoir dynamical evolution is $\Delta t = 0.025$. The hyperparameter values (Please refer to the Methods section for their meanings) are optimized to be $d = 843$, $\lambda = 0.48$, $k_{\text{in}} = 0.29$, $k_c = 0.113$, $\alpha = 0.41$, and $\beta = 1 \times 10^{-10}$. Our computations reveal that, for the deterministic version of the Lorenz-96 model, it is difficult to reduce the validation error below a small threshold. However, adding an appropriate amount of noise into the training time series (18) can lead to smaller validation errors. We add an additive Gaussian noise with standard deviation σ_{noise} to each input data channel to the reservoir network [including the driving channel $f(t)$]. The noise amplitude σ_{noise} is treated as an additional hyperparameter to be optimized. For the toy climate system, we test several noise levels and find the optimal noise level giving the best validating performance: $\sigma_{\text{noise}} \approx 10^{-3}$.

Figures 1(B1) and 1(B2) show the dynamical behaviors generated by the digital twin for the same values of A as in Figs. 1(A1) and 1(A2), respectively. It can be seen that not only does the digital twin produce the correct dynamical behavior in the same chaotic regime where the training is carried out, it can also extrapolate beyond the training parameter regime to correctly predict the unseen system dynamics there (quasiperiodicity in this case). To provide support in a broader parameter range, we calculate true bifurcation diagram, as shown in Fig. 1(C), where the four vertical dashed lines indicate the four values of the training parameter. The bifurcation diagram generated by the digital twin is shown in Fig. 1(D), which agrees reasonably well with the true diagram. Note that the digital twin fails to predict the the periodic window about $A = 3.2$, due to its high period (period-21 - see Supporting Material for a discussion). To quantify the prediction performance, we examine the smallest simple connected region that encloses the entire attractor - the spanned region, and calculate the overlapping ratio of the true to the predicted spanned regions. Figure 1(E) shows the relative error of the spanned regions (RESR) versus A, where the spanned regions are calculated from a two-dimensional projection of the attractor. Except for the locations of two periodic windows, RESR

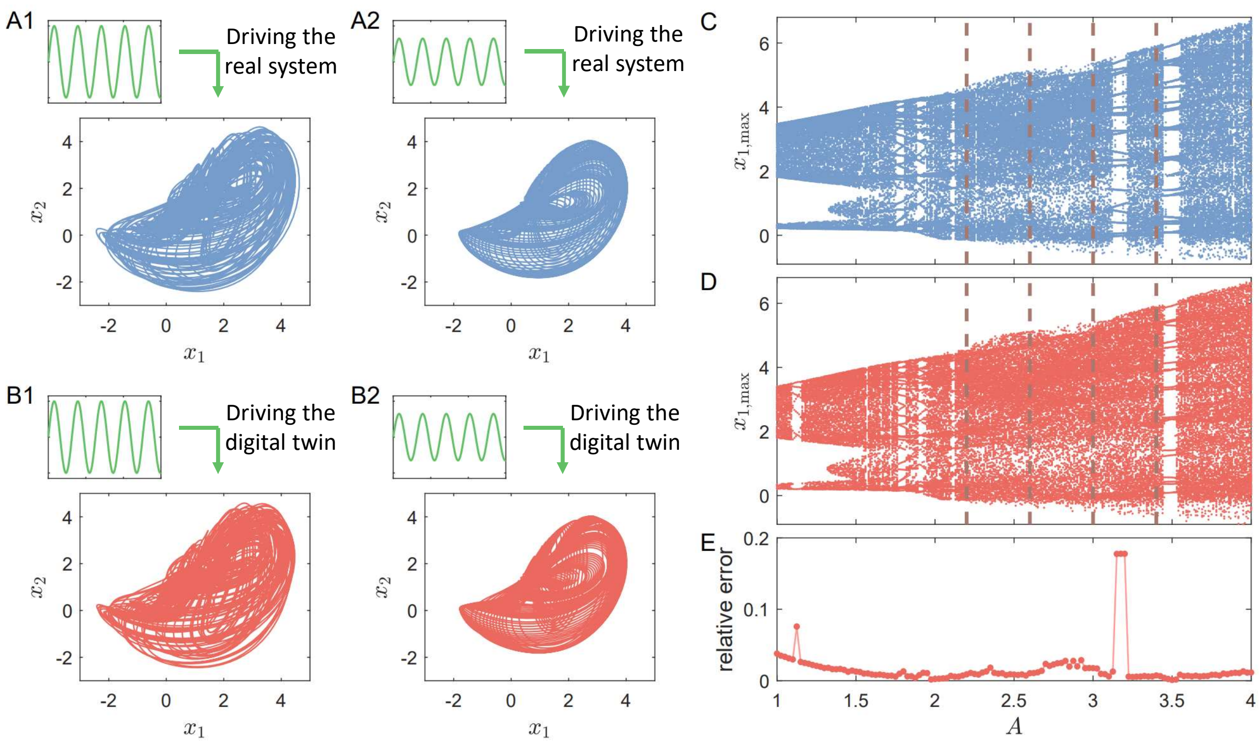

Fig. 1. Digital twin of the Lorenz-96 climate system. The toy climate system is described by six coupled first-order nonlinear differential equations (phase-space dimension $m = 6$), which is driven by a sinusoidal signal $f(t) = A\sin(\omega t) + F$. (A1,A2) Ground truth: chaotic and quasi-periodic dynamics in the system for $A = 2.2$ and $A = 1.6$, respectively, for $\omega = 2$ and $F = 2$. The sinusoidal driving signals $f(t)$ are schematically illustrated. (B1, B2) The corresponding dynamics of the digital twin under the same driving signal $f(t)$. Training of the digital twin is conducted using time series from the chaotic regime. The result in (B2) indicates that the digital twin is able to extrapolate outside the chaotic regime to generate the unseen quasi-periodic behavior. (C, D) True and digital-twin generated bifurcation diagrams of the toy climate system, where the four vertical red dashed lines indicate the values of driving amplitudes A, from which the training time series data are obtained. The reasonable agreement between the two bifurcation diagrams attests to the ability of the digital twin to reproduce the distinct dynamical behaviors of the target climate system in different parameter regimes, even with training data only in the chaotic regime. (E) Relative error of the spanned regions (RESR) versus A. The error is within 4%, except for the locations of two periodic windows at which the large errors are due to long transients (see Sec. 8 in Supporting Information).

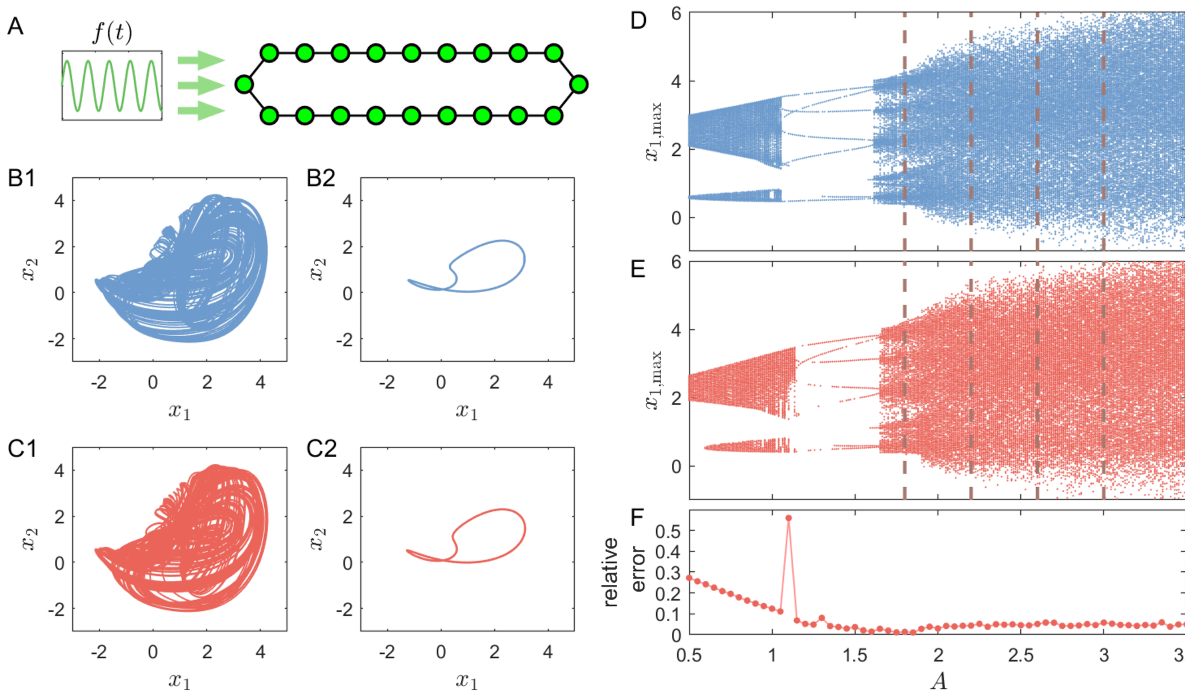

Fig. 2. Digital twin consisting of a number of parallel RC neural networks for high-dimensional chaotic systems. The target system is the Lorenz-96 climate network of $m = 20$ nodes, subject to a global periodic driving $f(t) = A\sin(\omega t) + F$. (A) The structure of the digital twin, where each filled green circle represents a small RC network with the input dimension $D_{in} = 5$ and output dimension $D_{\rm out} = 2$. (B1, B2) A chaotic and periodic attractor in a two-dimensional subspace of the target system for $A = 1.8$ and $A = 1.6$, respectively, for $\omega = 2$ and $F = 2$. (C1, C2) The attractors generated by the digital twin corresponding to those in (B1, B2), respectively, where the training is done using four time series from four different values of forcing amplitude A, all in the chaotic regime. The digital twin with a parallel structure is able to successfully extrapolate the unseen periodic behavior with completely chaotic training data. (D, E) The true and digital-twin generated bifurcation diagrams, respectively, where the four vertical dashed lines in (c) specify the four values of A from which the training time series are obtained. (F) RESR versus A, where the peak at $A \approx 1.1$ is due to the mismatched ending point of the wide periodic window for $A \in (1.2, 1.7)$.

is within 4%. When the testing values of A are further away from the training values, RESR tends to increase.

Previously, it was suggested that RC can have a certain degree of extrapolability (34–39). Figure 1 represents an example where the target system's response is extrapolated to external sinusoidal driving with unseen amplitudes. In general, extrapolation is a difficult problem. Some limitations of the extrapolability with respect to the external driving signal is discussed in SM 1, where the digital twin can predict the crisis point but cannot extrapolate the asymptotic behavior after the crisis.

In the following, we systematically study the applicability of the digital twin in solving forecasting problems in more complicated situations than the basic settings demonstrated in Fig. 1. The issues to be addressed are high dimensionality, the effect of the waveform of the driving on forecasting, and the generalizability across Lorenz-96 networks of different sizes. Results of continual forecasting and inferring hidden dynamical variables using only rare updates of the observable are presented in SM 3 and 4, respectively.

Digital twins of parallel RC neural networks for high-dimensional Lorenz-96 climate networks. We extend the methodology of digital twin to high-dimensional Lorenz-96 climate networks, e.g., $m = 20$. To deal with such a high-dimensional target system, if a single reservoir system is used, the required size of the neural network in the hidden layer will be too large to be computationally efficient. We thus turn to the parallel configuration (25) that consists of many small-size RC networks, each "responsible" for a small part of the target system. For the Lorenz-96 network with $m = 20$ coupled nodes, our digital twin consists of ten parallel RC networks, each monitoring and forecasting the dynamical evolution of two nodes ($D_{\rm out} = 2$). Because each node in the Lorenz-96 network is coupled to three nearby nodes, we set $D_{\rm in} = D_{\rm out} + D_{\rm couple} = 2 + 3 = 5$ to ensure that sufficient information is supplied to each RC network.

The specific parameters of the digital twin are as follows. The size of the recurrent layer is $D_r = 1,200$. For each training value of the forcing amplitude A, the training and validation lengths are $t = 3,500$ and $t = 100$, respectively. The "warming up" length is $t = 20$ and the time step of the dynamical evolution of the digital twin is $\Delta t = 0.025$. The optimized hyperparameter values are $d = 31$, $\lambda = 0.75$, $k_{\rm in} = 0.16$, $k_c = 0.16$, $\alpha = 0.33$, $\beta = 1 \times 10^{-12}$, and $\sigma_{\rm noise} = 10^{-2}$.

The periodic signal used to drive the Lorenz-96 climate network of 20 nodes is $f(t) = A\sin(\omega t) + F$ with $\omega = 2$, and $F = 2$. The structure of the digital twin consists of 20 small RC networks as illustrated in Fig. 2(A). Figures 2(B1) and 2(B2) show a chaotic and a periodic attractor for $A = 1.8$ and $A = 1.6$, respectively, in the (x_1, x_2) plane. Training of the digital twin is conducted by using four time series from four different values of A, all in the chaotic regime. The attractors generated by the digital twin for $A = 1.8$ and $A = 1.6$ are shown in Figs. 2(C1) and 2(C2), respectively, which agree well with the ground truth. Figure 2(D) shows the bifurcation diagram of the target system (the ground truth), where the four values of A: $A = 1.8$, 2.2, 2.6, and 3.0, from which the training chaotic time series are obtained, are indicated by the four respective vertical dashed lines. The bifurcation diagram generated by the digital twin is shown in Fig. 2(E), which agrees well with the ground truth in Fig. 2(D). Figure 2(F) shows the relative error RESR versus A, where a peak occurs at $A \approx 1.1$ due to the mismatched ending point of the large periodic window. The error values are between 2% to 6%.

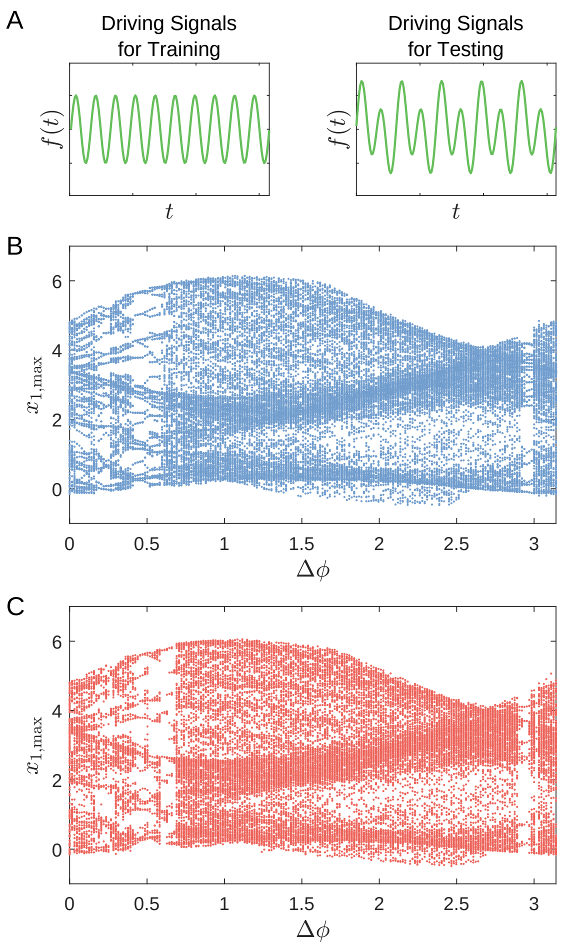

Fig. 3. Effects of waveform change in the external driving on the performance of the digital twin. The time series used to train the digital twin are from the target system subject to external driving of a particular waveform. A change in the waveform occurs subsequently, leading to a different driving signal during the testing phase. (A) During the training phase, the driving signal is of the form $f(t) = A\sin(\omega t) + F$ and time series from four different values of A are used for training the digital twin. The right panel illustrates an example of the changed driving signal during the testing phase. (B) The true bifurcation diagram of the target system under a testing driving signal. (C) The bifurcation diagram generated by the digital twin, facilitated by an optimal level of training noise determined through hyperparameter optimization.

Digital twins under external driving with varied waveform. The external driving signal is an essential ingredient in our articulation of the digital twin, which is particularly relevant to critical systems of interest such as the climate systems. In applications, the mathematical form of the driving signal may change with time. Can a digital twin produce the correct system behavior under a driving signal that is different than the one it has "seen" during the training phase? Note that, in the examples treated so far, it has been demonstrated that our digital twin can extrapolate the dynamical behavior of a target system under a driving signal of the same mathematical form but with a different amplitude. Here, the task is more challenging as the form of the driving signal has changed.

As a concrete example, we consider the Lorenz-96 climate network of $m = 6$ nodes, where a digital twin is trained with a purely sinusoidal signal $f(t) = A\sin(\omega t) + F$, as illustrated in the left column of Fig. 3(A). During the testing phase, the driving signal has the form of the sum of two sinusoidal signals with different frequencies: $f(t) = A_1 \sin(\omega_1 t) + A_2 \sin(\omega_2 t + \Delta\phi) + F$, as illustrated in the right panel of Fig. 3(A). We set $A_1 = 2$, $A_2 = 1$, $\omega_1 = 2$, $\omega_2 = 1$, $F = 2$, and use $\Delta\phi$ as the bifurcation parameter. The RC parameter setting is the same as that in Fig. 1. The training and validating lengths for each driving amplitude A value are $t = 3,000$ and $t = 12$, respectively. We fine that this setting prevents the digital twin from generating an accurate bifurcation diagram, but a small amount of dynamical noise to the target system can improve the performance of the digital twin. To demonstrate this, we apply an additive noise term to the driving signal $f(t)$ in the training phase: $df(t)/dt = \omega A\cos(\omega t) + \delta_{\rm DN}\xi(t)$, where $\xi(t)$ is a Gaussian white noise of zero mean and unit variance, and $\delta_{\rm DN}$ is the noise amplitude (e.g., $\delta_{\rm DN} = 3 \times 10^{-3}$). We use the 2nd-order Heun method (44) to solve the stochastic differential equations describing the target Lorenz-96 system. Intuitively, the noise serves to excite different modes of the target system to instill richer information into the training time series, making the process of learning the target dynamics more effective. Figures 3(B) and 3(C) show the actual and digital-twin generated bifurcation diagrams. Although the digital twin encountered driving signals in a completely "uncharted territory," it is still able to generate the bifurcation diagram with a reasonable accuracy. The added dynamical noise is creating small fluctuations in the driving signal $f(t)$. This may yield richer excited dynamical features of the target system in the training data set, which can be learned by the RC network. This should be beneficial for the RC network to adapt to different waveform in the testing. Additional results with varying testing waves $f(t)$ are presented in SM 5.

Extrapolability of digital twin with respect to system size. In the examples studied so far, it has been demonstrated that our RC based digital twin has a strong extrapolability in certain dimensions of the parameter space. Specifically, the digital twin trained with time series data from one parameter region can follow the dynamical evolution of the target system in a different parameter regime. One question is whether the digital twin possesses certain extrapolability in the system size. For example, consider the Lorenz-96 climate network of size m. In Fig. 2, we use an array of parallel RC networks to construct a digital twin for the climate network of a fixed size m, where the number of parallel RCs is $m/2$ (assuming that m is even), and training and testing/monitoring are carried out for the same system size. We ask, if a digital twin is trained for climate networks of certain sizes, will it have the ability to generate the correct dynamical behaviors for climate networks of different sizes? If yes, we say that the digital twin has the extrapolability with respect to system size.

As an example, we create a digital twin with time series data from Lorenz-96 climate networks of sizes $m = 6$ and $m = 10$, as shown in Fig. 4(A). For each system size, four values of the forcing amplitude A are used to generate the training time

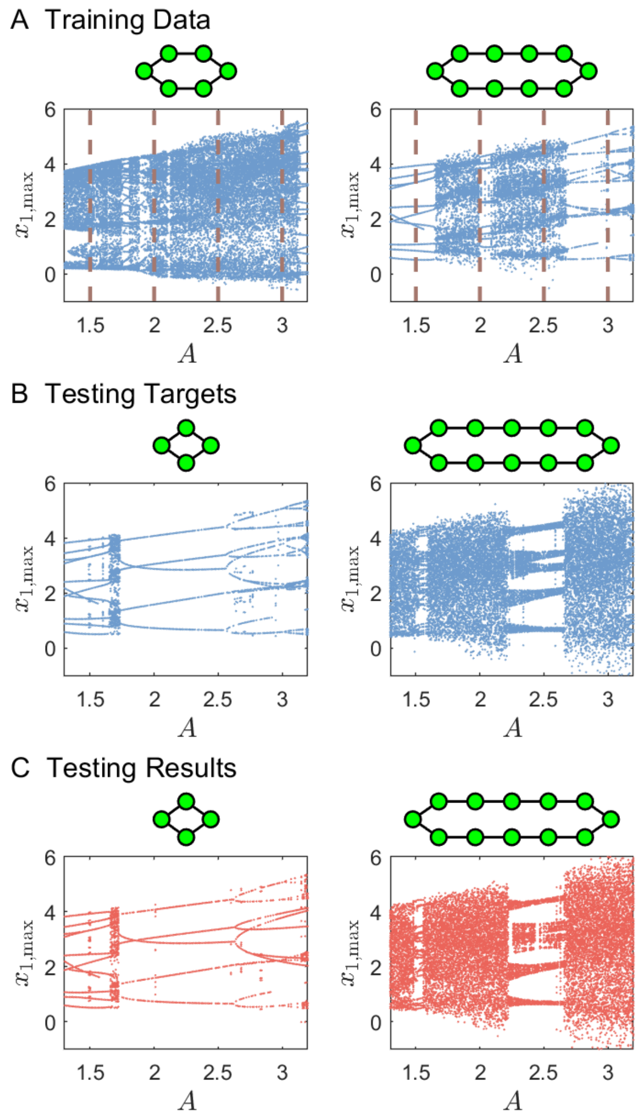

Fig. 4. Demonstration of extrapolability of digital twin in system size. (A) The digital twin is trained using time series from the Lorenz-96 climate networks of size $m = 6$ and $m = 10$. The target climate system is subject to a sinusoidal driving $f(t) = A\sin(\omega t) + F$, and the training time series data are from the A values marked by the eight vertical orange dashed lines. (B) The true bifurcation diagrams of the target climate network of size $m = 4$ and $m = 12$. (C) The corresponding digital-twin generated bifurcation diagrams, where the twin consists of $m/2$ parallel RC networks, each taking input from two nodes in the target system and from the nodes in the network that are coupled to the two nodes.

series: A =1.5, 2.0, 2.5, and 3.0, as marked by the vertical orange dashed lines in Figs. 4(A) and 4(B). As in Fig. 2, the digital twin consists of $m/2$ parallel RC networks, each of size $D_r = 1,500$. The optimized hyperparameter values are determined to be $d = 927$, $\lambda = 0.71$, $k_{\text{in}} = 0.076$, $k_c = 0.078$, $\alpha = 0.27$, $\beta = 1 \times 10^{-11}$, and $\sigma_{\text{noise}} = 3 \times 10^{-3}$. Then we consider climate networks of two different sizes: $m = 4$ and $m = 12$, and test if the trained digital twin can be adapted to the new systems. For the network of size $m = 4$, we keep only two parallel RC networks for the digital twin. For $m = 12$, we add one additional RC network to the trained digital twin for $m = 10$, so the new twin consists of six parallel RC networks of the same hyperparameter values. The true bifurcation diagrams for the climate system of sizes $m = 4$ and $m = 12$ are shown in Fig. 4(B) (the left and right panels, respectively). The corresponding bifurcation diagrams generated by the adapted digital twins are shown in Fig. 4(C), which agree with the ground truth reasonably well, demonstrating that our RC based digital twin possesses certain extrapolability in system size.

Discussion

We have articulated the principle of creating digital twins for nonlinear dynamical systems based on RCs that are recurrent neural networks. In general, RC is a powerful neural network framework that does not require backpropagation during training but only a linear regression is needed. This feature makes the development of digital twins based on RC computationally efficient. We have demonstrated that a well-trained RC network is able to serve as a digital twin for systems subject to external, time-varying driving. The twin can be used to anticipate possible critical transitions or regime shifts in the target system as the driving force changes, thereby providing early warnings for potential catastrophic collapse of the system. We have used a variety of examples from different fields to demonstrate the workings and the anticipating power of the digital twin, which include the Lorenz-96 climate network of different sizes (in the main text), a driven chaotic CO_2 laser system (SM 1), and an ecological system (SM 2). For low-dimensional nonlinear dynamical systems, a single RC network is sufficient for the digital twin. For high-dimensional systems such as the climate network of a relatively large size, parallel RC networks can be integrated to construct the digital twin. At the level of the detailed state evolution, our recurrent neural network based digital twin is essentially a dynamical twin system that evolves in parallel to the real system, and the evolution of the digital twin can be corrected from time to time using sparse feedback of data from the target system (SM 3). In cases where direct measurements of the target system are not feasible or are too costly, the digital twin provides a way to assess the dynamical evolution of the target system. At the qualitative level, the digital twin can faithfully reproduce the attractors of the target system, e.g., chaotic, periodic, or quasiperiodic, without the need of state updating. In addition, we show that the digital twin is able to accurately predict a critical bifurcation point and the average lifetime of transient chaos that occurs after the bifurcation, even under a driving signal that is different from that during the training (SM 6). The issue of robustness against dynamical and observational noises in the training data has also been treated (SM 7).

To summarize, our RC based digital twins are capable of performing the following tasks: (1) extrapolating certain dynamical evolution of the target system beyond the training parameter regime, (2) making long-term continual forecasting of nonlinear dynamical systems under nonstationary external driving with sparse state updates, (3) inferring the existence of hidden variables in the system and reproducing/predicting their dynamical evolution, (4) adapting to external driving of different waveform, and (5) extrapolating the global bifurcation behaviors to systems of different sizes.

Our design of the digital twins for nonlinear dynamical systems can be extended in a number of ways.

1. Online learning. Online or continual learning is a recent trend in machine-learning research. Unlike the approach of batch learning, where one gathers all the training data in one place and does the training on the entire data set (the way by which training is conducted for our work), in an online learning environment, one evolves the machine learning model incrementally with the flow of data. For each training step, only the newest inputted training data is used to update the machine learning model. When a new data set is available, it is not necessary to train the model over again on the entire data set accumulated so far, but only on the new set. This can result in a significant reduction in the computational complexity. Previously, an online learning approach to RC known as the FORCE learning was developed (45). An attempt to deal with the key problem of online learning termed "catastrophic forgetting" was made in the context of RC (46). Further investigation is required to see if these methods can be exploited for creating digital twins through online learning.

2. Beyond reservoir computing. Second, the potential power of recurrent neural network based digital twin may be further enhanced by using more sophisticated recurrent neural network models depending on the target problem. We use the RC networks because they are relatively simple yet powerful enough for both low- and high-dimensional dynamical systems. Schemes such as knowledge-based hybrid RC (47) or ODE-nets (48) are worth investigating.

3. Reinforcement learning. Is it possible to use digital twins to make reinforcement learning feasible in situations where the target system cannot be "disturbed"? Particularly, reinforcement learning requires constant interaction with the target system during training so that the machine can learn from its mistakes and successes. However, for a real-world system, these interactions may be harmful, uncontrollable, and irreversible. As a result, reinforcement learning algorithms are rarely applied to safety-critical systems (49). In this case, digital twins can be beneficial. By building a digital twin, the reinforcement learning model does not need to interact with the real system, but with its simulated replica for efficient training. This area of research is called model-based reinforcement learning (50).

4. Potential benefits of noise. A phenomenon uncovered in our study is the beneficial role of dynamical noise in the target system. As briefly discussed in Fig. 3, adding dynamic noise in the training dataset enhances the digital twin's ability to extrapolate the dynamics of the target system with different waveform of driving. Intuitively, noise can facilitate the exploration of the phase space of the target nonlinear system. A systematic study of the interplay between dynamical noise and the performance of the digital twin is worthy.

5. Extrapolability. The demonstrated extrapolability of our digital twin, albeit limited, may open the door to forecasting the behavior of large systems using twins trained on small systems. Much research is needed to address this issue.

6. Spatiotemporal dynamical systems with multistability. We have considered digital twins for a class of coupled dynamical systems: the Lorenz-96 climate model. When developing digital twins for spatiotemporal dynamical systems, two issues can arise. One is the computational complexity associated with such high-dimensional systems. We have demonstrated that parallel reservoir computing provides a viable solution. Another issue is multistability. Spatiotemporal dynamical systems in general exhibit extremely rich dynamical behaviors such as chimera states (51–59). To develop digital twins of spatiotemporal dynamical systems with multiple coexisting states requires that the underlying recurrent neural networks possess certain memory capabilities. To develop methods to incorporate memories into digital twins is a problem of current interest.

Materials and Methods

Methods. The basic construction of the digital twin of a nonlinear dynamical system (61) is illustrated in Fig. 5. It is essentially a recurrent RC neural network with a control mechanism, which requires two types of input signals: the observational time series for training and the control signal $f(t)$ that remains in both the training and self-evolving phase. The hidden layer hosts a random or complex network of artificial neurons. During the training, the hidden recurrent layer is driven by both the input signal $\mathbf{u}(t)$ and the control signal $f(t)$. The neurons in the hidden layer generate a high-dimensional nonlinear response signal. Linearly combining all the responses of these hidden neurons with a set of trainable and optimizable parameters yields the output signal. Specifically, the digital twin consists of four components: (i) an input subsystem that maps the low-dimensional (D_{in}) input signal into a (high) D_r-dimensional signal through the weighted $D_r \times D_{\text{in}}$ matrix $\mathcal{W}_{\text{in}}$, (ii) a reservoir network of N neurons characterized by $\mathcal{W}_r$, a weighted network matrix of dimension $D_r \times D_r$, where $D_r \gg D_{\text{in}}$, (iii) an readout subsystem that converts the D_r-dimensional signal from the reservoir network into an D_{out}-dimensional signal through the output weighted matrix $\mathcal{W}_{out}$, and (iv) a controller with the matrix $\mathcal{W}_c$. The matrix $\mathcal{W}_r$ defines the structure of the reservoir neural network in the hidden layer, where the dynamics of each node are described by an internal state and a nonlinear hyperbolic tangent activation function.

The matrices $\mathcal{W}_{\text{in}}$, $\mathcal{W}_c$, and $\mathcal{W}_r$ are generated randomly prior to training, whereas all elements of $\mathcal{W}_{\text{out}}$ are to be determined through training. Specifically, the state updating equations for the training and self-evolving phases are, respectively,

$$\begin{aligned}\mathbf{r}(t+\Delta t) &= (1-\alpha)\mathbf{r}(t) \\ &\quad + \alpha \tanh\left[\mathcal{W}_r\mathbf{r}(t) + \mathcal{W}_{\text{in}}\mathbf{u}(t) + \mathcal{W}_c f(t)\right], \qquad [2]\\ \mathbf{r}(t+\Delta t) &= (1-\alpha)\mathbf{r}(t) \\ &\quad + \alpha \tanh\left[\mathcal{W}_r\mathbf{r}(t) + \mathcal{W}_{\text{in}}\mathcal{W}_{\text{out}}\mathbf{r}'(t) + \mathcal{W}_c f(t)\right], \qquad [3]\end{aligned}$$

where $\mathbf{r}(t)$ is the hidden state, $\mathbf{u}(t)$ is the vector of input training data, Δt is the time step, the vector $\tanh(\mathbf{p})$ is defined to be $[\tanh(p_1), \tanh(p_2), \ldots]^T$ for a vector $\mathbf{p} = [p_1, p_2, ...]^T$, and α is the leakage factor. During the training, several trials of data are typically used under different driving signals so that the digital twin can "sense, learn, and mingle" the responses of the target system to gain the ability to extrapolate a response to a new driving signal that has never been encountered before. We input these trials of training data, i.e., a few pairs of $\mathbf{u}(t)$ and the associated $f(t)$, through the matrices $\mathcal{W}_{\text{in}}$ and $\mathcal{W}_c$ sequentially. Then we record the state vector $\mathbf{r}(t)$ of the neural network during the entire training phase as a matrix $\mathcal{R}$. We also record all the desired output, which is the one-step prediction result $\mathbf{v}(t) = \mathbf{u}(t+\Delta t)$, as the matrix $\mathcal{V}$. To make the readout nonlinear and to avoid unnecessary symmetries in the system (24, 62), we change the matrix $\mathcal{R}$ into $\mathcal{R}'$ by squaring the entries of even dimensions in the states of the hidden layer. [The vector $(\mathbf{r}'(t)$ in Eq. Eq. (3) is defined in a similar way.] We carry out a linear regression between $\mathcal{V}$ and $\mathcal{R}'$, with a ℓ-2 regularization coefficient β, to determine the readout matrix:

$$\mathcal{W}_{\text{out}} = \mathcal{V} \cdot \mathcal{R}'^T(\mathcal{R}' \cdot \mathcal{R}'^T + \beta\mathcal{I})^{-1}. \qquad [4]$$

To achieve acceptable learning performance, optimization of hyperparameters is necessary. The four widely used global optimization methods are genetic algorithm (63–65), particle swarm optimization (66, 67), Bayesian optimization (68, 69), and surrogate optimization (70–72). We use the surrogate optimization (the algorithm

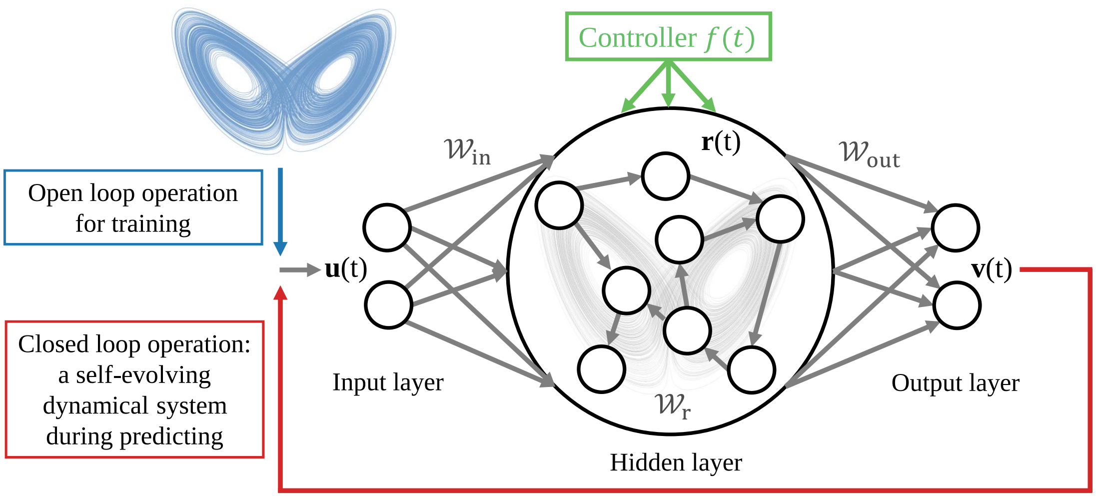

Fig. 5. Basic structure of the digital twin of a chaotic system. It consists of three layers: the input layer, the hidden recurrent layer, an output layer, as well as a controller component. The input matrix $\mathcal{W}_{\text{in}}$ maps the D_{in}-dimensional input chaotic data to a vector of much higher dimension D_r, where $D_r \gg D_{\text{in}}$. The recurrent hidden layer is characterized by the $D_r \times D_r$ weighted matrix $\mathcal{W}_r$. The dynamical state of the i^{th} neuron in the reservoir is r_i, for $i = 1, \ldots, D_r$. The hidden-layer state vector is $\mathbf{r}(t)$, which is an embedding of the input (60). The output matrix $\mathcal{W}_{\text{out}}$ readout the hidden state into the D_{out}-dimensional output vector. The controller provides an external driving signal $f(t)$ to the neural network. During training, the vector $\mathbf{u}(t)$ is the input data, and the blue arrow exists during the training phase only. In the predicting phase, the output vector $\mathbf{v}(t)$ is directly fed back to the input layer, generating a closed-loop, self-evolving dynamical system, as indicated by the red arrow connecting $\mathbf{v}(t)$ to $\mathbf{u}(t)$. The controller remains on in both the training and predicting phases.

surrogateopt in Matlab). The hyperparameters that are optimized include d - the average degree of the recurrent network in the hidden layer, λ - the spectral radius of the recurrent network, k_{in} - the scaling factor of $\mathcal{W}_{\text{in}}$, k_c - the scaling of $\mathcal{W}_{\text{c}}$, c_0 - the bias in Eq. Eq. (2) and Eq. (3), α - the leakage factor, and β - the ℓ-2 regularization coefficient. The RC network is validated using the same driving $f(t)$ as in the training phase, but driving signals with different amplitudes and frequencies are used in the testing phase. Prior to making predictions, the RC network is initialized using random short segments of the training data, so no data from the target system under the testing driving signals $f(t)$ is required. To produce the bifurcation diagram, sufficiently long transients in the dynamical evolution of the RC network are disregarded.

ACKNOWLEDGMENTS. We thank Z.-M. Zhai for discussions. This work was supported by the Army Research Office through Grant No. W911NF-21-2-0055 and by the U.S.-Israel Energy Center managed by the Israel-U.S. Binational Industrial Research and Development (BIRD) Foundation.

1. EJ Eric J. Tuegel, AR Ingraffea, TG Eason, SM Spottswood, Reengineering aircraft structural life prediction using a digital twin. *Int. J. Aerosp. Eng.* **2011**, 154798 (2011).
2. F Tao, Q Qi, Make more digital twins. *Nature* **573**, 274–277 (2019).
3. A Rasheed, O San, T Kvamsdal, Digital twin: Values, challenges and enablers from a modeling perspective. *IEEE Access* **8**, 21980–22012 (2020).
4. K Bruynseels, FS de Sio, J van den Hoven, Digital twins in health care: Ethical implications of an emerging engineering paradigm. *Front. Gene.* **9**, 31 (2018).
5. SM Schwartz, K Wildenhaus, A Bucher, B Byrd, Digital twins and the emerging science of self: Implications for digital health experience design and "small" data. *Front. Comp. Sci.* **2**, 31 (2020).
6. R Laubenbacher, JP Sluka, JA Glazier, Using digital twins in viral infection. *Science* **371**, 1105–1106 (2021).
7. P Voosen, Europe builds 'digital twin' of earth to hone climate forecasts. *Science* **370**, 16–17 (2020).
8. P Bauer, B Stevens, W Hazeleger, A digital twin of earth for the green transition. *Nat. Clim. Chang.* **11**, 80–83 (2021).
9. YC Lai, T Tél, *Transient Chaos - Complex Dynamics on Finite Time Scales.* (Springer, New York), (2011).
10. K McCann, P Yodzis, Nonlinear dynamics and population disappearances. *Ame. Nat.* **144**, 873–879 (1994).
11. A Hastings, et al., Transient phenomena in ecology. *Science* **361**, eaat6412 (2018).
12. M Dhamala, YC Lai, Controlling transient chaos in deterministic flows with applications to electrical power systems and ecology. *Phys. Rev. E* **59**, 1646–1655 (1999).
13. YC Lai, C Grebogi, J Kurths, Modeling of deterministic chaotic systems. *Phys. Rev. E* **59**, 2907–2910 (1999).
14. YC Lai, C Grebogi, Modeling of coupled chaotic oscillators. *Phys. Rev. Lett.* **82**, 4803–4806 (1999).
15. WX Wang, R Yang, YC Lai, V Kovanis, C Grebogi, Predicting catastrophes in nonlinear dynamical systems by compressive sensing. *Phys. Rev. Lett.* **106**, 154101 (2011).
16. WX Wang, YC Lai, C Grebogi, Data based identification and prediction of nonlinear and complex dynamical systems. *Phys. Rep.* **644**, 1–76 (2016).
17. YC Lai, Finding nonlinear system equations and complex network structures from data: A sparse optimization approach. *Chaos* **31**, 082101 (2021).
18. H Jaeger, The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *Ger. Natl. Res. Cent. for Inf. Technol. GMD Tech. Rep.* **148**, 13 (2001).
19. W Mass, T Nachtschlaeger, H Markram, Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neur. Comp.* **14**, 2531–2560 (2002).
20. H Jaeger, H Haas, Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science* **304**, 78–80 (2004).
21. ND Haynes, MC Soriano, DP Rosin, I Fischer, DJ Gauthier, Reservoir computing with a single time-delay autonomous Boolean node. *Phys. Rev. E* **91**, 020801 (2015).
22. L Larger, et al., High-speed photonic reservoir computing using a time-delay-based architecture: Million words per second classification. *Phys. Rev. X* **7**, 011015 (2017).
23. J Pathak, Z Lu, B Hunt, M Girvan, E Ott, Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data. *Chaos* **27**, 121102 (2017).
24. Z Lu, et al., Reservoir observers: Model-free inference of unmeasured variables in chaotic systems. *Chaos* **27**, 041102 (2017).
25. J Pathak, B Hunt, M Girvan, Z Lu, E Ott, Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Phys. Rev. Lett.* **120**, 024102 (2018).
26. TL Carroll, Using reservoir computers to distinguish chaotic signals. *Phys. Rev. E* **98**, 052209 (2018).
27. K Nakai, Y Saiki, Machine-learning inference of fluid variables from data using reservoir computing. *Phys. Rev. E* **98**, 023111 (2018).
28. ZS Roland, U Parlitz, Observing spatio-temporal dynamics of excitable media using reservoir computing. *Chaos* **28**, 043118 (2018).
29. A Griffith, A Pomerance, DJ Gauthier, Forecasting chaotic systems with very low connectivity reservoir computers. *Chaos* **29**, 123108 (2019).
30. J Jiang, YC Lai, Model-free prediction of spatiotemporal dynamical systems with recurrent neural networks: Role of network spectral radius. *Phys. Rev. Res.* **1**, 033056 (2019).
31. G Tanaka, et al., Recent advances in physical reservoir computing: A review. *Neu. Net.* **115**, 100–123 (2019).
32. H Fan, J Jiang, C Zhang, X Wang, YC Lai, Long-term prediction of chaotic systems with machine learning. *Phys. Rev. Res.* **2**, 012080 (2020).
33. C Zhang, J Jiang, SX Qu, YC Lai, Predicting phase and sensing phase coherence in chaotic systems with machine learning. *Chaos* **30**, 083114 (2020).
34. C Klos, YFK Kossio, S Goedeke, A Gilra, RM Memmesheimer, Dynamical learning of dynamics. *Phys. Rev. Lett.* **125**, 088103 (2020).
35. LW Kong, HW Fan, C Grebogi, YC Lai, Machine learning prediction of critical transition and system collapse. *Phys. Rev. Res.* **3**, 013090 (2021).
36. D Patel, D Canaday, M Girvan, A Pomerance, E Ott, Using machine learning to predict sta-

tistical properties of non-stationary dynamical processes: System climate, regime transitions, and the effect of stochasticity. *Chaos* **31**, 033149 (2021).

37. JZ Kim, Z Lu, E Nozari, GJ Pappas, DS Bassett, Teaching recurrent neural networks to infer global temporal structure from local examples. *Nat. Mach. Intell.* **3**, 316–323 (2021).
38. H Fan, LW Kong, YC Lai, X Wang, Anticipating synchronization with machine learning. *Phys. Rev. Resesearch* **3**, 023237 (2021).
39. LW Kong, H Fan, C Grebogi, YC Lai, Emergence of transient chaos and intermittency in machine learning. *J. Phys. Complex.* **2**, 035014 (2021).
40. E Bollt, On explaining the surprising success of reservoir computing forecaster of chaos? the universal machine learning dynamical system with contrast to var and dmd. *Chaos* **31**, 013108 (2021).
41. DJ Gauthier, E Bollt, A Griffith, WA Barbosa, Next generation reservoir computing. *Nat. Commun.* **12**, 1–8 (2021).
42. TL Carroll, Optimizing memory in reservoir computers. *Chaos* **32**, 023123 (2022).
43. EN Lorenz, Predictability: A problem partly solved in *Proc. Seminar on Predictability*. Vol. 1, (1996).
44. C Van den Broeck, J Parrondo, R Toral, R Kawai, Nonequilibrium phase transitions induced by multiplicative noise. *Phys. Rev. E* **55**, 4084 (1997).
45. D Sussillo, LF Abbott, Generating coherent patterns of activity from chaotic neural networks. *Neuron* **63**, 544–557 (2009).
46. T Kobayashi, T Sugino, Continual learning exploiting structure of fractal reservoir computing in *International Conference on Artificial Neural Networks*. (Springer), pp. 35–47 (2019).
47. J Pathak, et al., Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model. *Chaos* **28**, 041101 (2018).
48. RT Chen, Y Rubanova, J Bettencourt, DK Duvenaud, Neural ordinary differential equations. *Adv. Neu. Info. Proc. Sys.* **31** (2018).
49. F Berkenkamp, M Turchetta, AP Schoellig, A Krause, Safe model-based reinforcement learning with stability guarantees. *arXiv preprint arXiv:1705.08551* (2017).
50. TM Moerland, J Broekens, CM Jonker, Model-based reinforcement learning: A survey. *arXiv preprint arXiv:2006.16712* (2020).
51. Y Kuramoto, D Battogtokh, Coexistence of coherence and incoherence in nonlocally coupled phase oscillators. *Nonlin. Phenom. Complex Syst.* **5**, 380–385 (2002).
52. DM Abrams, SH Strogatz, Chimera states for coupled oscillators. *Phys. Rev. Lett.* **93**, 174102 (2004).
53. I Omelchenko, Y Maistrenko, P Hövel, E Schöll, Loss of coherence in dynamical networks: Spatial chaos and chimera states. *Phys. Rev. Lett.* **106**, 234102 (2011).
54. MR Tinsley, S Nkomo, K Showalter, Chimera and phase-cluster states in populations of coupled chemical oscillators. *Nat. Phys.* **8**, 662 (2012).
55. AM Hagerstrom, et al., Experimental observation of chimeras in coupled-map lattices. *Nat. Phys.* **8**, 658 (2012).
56. I Omelchenko, OE Omel'chenko, P Hövel, E Schöll, When nonlocal coupling between oscillators becomes stronger: Patched synchrony or multichimera states. *Phys. Rev. Lett.* **110**, 224101 (2013).
57. I Omelchenko, A Zakharova, P Hövel, J Siebert, E Schöll, Nonlinearity of local dynamics promotes multi-chimeras. *Chaos* **25**, 083104 (2015).
58. I Omelchenko, OE Omel'chenko, A Zakharova, E Schöll, Optimal design of tweezer control for chimera states. *Phys. Rev. E* **97**, 012216 (2018).
59. LW Kong, YC Lai, Scaling law of transient lifetime of chimera states under dimension-augmenting perturbations. *Phys. Rev. Res.* **2**, 023196 (2020).
60. A Hart, J Hook, J Dawes, Embedding and approximation theorems for echo state networks. *Neu. Net.* **128**, 234–247 (2020).
61. (year?) The codes of this work are shared at github.com/lw-kong/Digital_Twin_2021.
62. J Herteux, C Räth, Breaking symmetries of the reservoir equations in echo state networks. *Chaos* **30**, 123142 (2020).
63. DE Goldberg, *Genetic Algorithms*. (Pearson Education India), (2006).
64. AR Conn, NI Gould, P Toint, A globally convergent augmented lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM J. Numer. Anal.* **28**, 545–572 (1991).
65. A Conn, N Gould, P Toint, A globally convergent lagrangian barrier algorithm for optimization with general inequality constraints and simple bounds. *Math. Comput.* **66**, 261–288 (1997).
66. J Kennedy, R Eberhart, Particle swarm optimization in *Proceedings of ICNN'95-International Conference on Neural Networks*. (IEEE), Vol. 4, pp. 1942–1948 (1995).
67. E Mezura-Montes, CAC Coello, Constraint-handling in nature-inspired numerical optimization: past, present and future. *Swarm Evol. Comput.* **1**, 173–194 (2011).
68. MA Gelbart, J Snoek, RP Adams, Bayesian optimization with unknown constraints. *arXiv preprint arXiv:1403.5607* (2014).
69. J Snoek, H Larochelle, RP Adams, Practical bayesian optimization of machine learning algorithms in *NeurIPS*. pp. 2951–2959 (2012).
70. HM Gutmann, A radial basis function method for global optimization. *J. Glob. Optim.* **19**, 201–227 (2001).
71. RG Regis, CA Shoemaker, A stochastic radial basis function method for the global optimization of expensive functions. *INFORMS J. Comput.* **19**, 497–509 (2007).
72. Y Wang, CA Shoemaker, A general stochastic algorithmic framework for minimizing expensive black box objective functions based on surrogate models and sensitivity analysis. *arXiv preprint arXiv:1410.6271* (2014).

Supporting Information for

Digital twins of nonlinear dynamical systems

Ling-Wei Kong, Yang Weng, Bryan Glaz, Mulugeta Haile and Ying-Cheng Lai

Ying-Cheng Lai
E-mail: Ying-Cheng.Lai@asu.edu

This PDF file includes:

Supporting text
Figs. S1 to S10
SI References

Supporting Information Text

1. A driven chaotic laser system

We consider the single-mode, class B, driven chaotic CO_2 laser system (1–4) described by

$$\frac{du}{dt} = -u[f(t) - z], \tag{1}$$

$$\frac{dz}{dt} = \epsilon_1 z - u - \epsilon_2 z u + 1, \tag{2}$$

where the dynamical variables u and z are proportional to the normalized intensity and the population inversion, $f(t) = A\cos(\Omega t + \phi)$ is the external sinusoidal driving signal of amplitude A and frequency Ω, ϵ_1 and ϵ_2 are two parameters. Chaos is common in this laser system (1, 2, 4). For example, for $\epsilon_1 = 0.09$, $\epsilon_2 = 0.003$, and $A = 1.8$, there is a chaotic attractor for $\Omega < \Omega_c \approx 0.912$, as shown by a sustained chaotic time series in Fig. S1(a1). The chaotic attractor is destroyed by a boundary crisis (5) at Ω_c. For $\Omega > \Omega_c$, there is transient chaos, after which the system settles into periodic oscillations, as exemplified in Fig. S1(a2). Suppose chaotic motion is desired. The crisis bifurcation at Ω_c can then be regarded as a kind of system collapse.

To build a digital twin for the chaotic laser system, we use the external driving signal as the natural control signal for the RC network. Different from the examples in the main text, here the driving frequency Ω, instead of the driving amplitude A, serves as the bifurcation parameter. Assuming observational data in the form of time series are available for several values of Ω in the regime of a chaotic attractor, we train the RC network using chaotic time series collected from four values of $\Omega < \Omega_c$: $\Omega = 0.81$, 0.84, 0.87, and 0.90. The training parameter setting is as follows. For each Ω value in the training set, the training and validation lengths are $t = 2,000$ and $t = 83$, respectively, where the latter corresponds to approximately five Lyapunov times. The "warming up" length is $t = 0.5$. The time step of the reservoir system is $\Delta t = 0.05$. The size of the random RC network is $D_r = 800$. The optimal hyperparameter values are determined to be $d = 151$, $\lambda = 0.0276$, $k_{\text{in}} = 1.18$, $k_c = 0.113$, $\alpha = 0.33$, and $\beta = 2 \times 10^{-4}$.

Figures S1(A1) and S1(A2) show two representative time series from the laser model (the ground truth) for $\Omega = 0.905 < \Omega_c$ and $\Omega = 0.925 > \Omega_c$, respectively. The one in panel (A1) is associated with sustained chaos (pre-critical) and the other in panel (A2) is characteristic of transient chaos with a final periodic attractor (post-critical). The corresponding time series generated by the digital twin are shown in Figs. S1(B1) and S1(B2), respectively. It can be seen that the training aided by the control signal enables the digital twin to correctly capture the dynamical climate of the target system, e.g., sustained or transient chaos. The true return maps in the pre-critical and post-critical regimes are shown in Figs. S1(C1) and S1(C2), respectively, and the corresponding maps generated by the digital twin are shown in Figs. S1(D1) and S1(D2). In the pre-critical regime, an invariant region (the green dashed square) exists on the return map in which the trajectories are confined, leading to sustained chaotic motion, as shown in Figs. S1(C1) and S1(D1). Within the invariant region in which the chaotic attractor lives, the digital twin captures the essential dynamical features of the attractor. Because the training data are from the chaotic attractor of the target system, the digital twin fails to generate the portion of the real return map that lies outside the invariant region, which is expected because the digital twin has never been exposed to the dynamical behaviors that are not on the chaotic attractor. In the post-critical regime, a "leaky" region emerges, as indicated by the red arrows in Figs. S1(C2) and S1(D2), which destroys the invariant region and leads to transient chaos. The remarkable feature is that the digital twin correctly assesses the existence of the leaky region, even when no such information is fed into the twin during training. From the point of view of predicting system collapse, the digital twin is able to anticipate the occurrence of the crisis and transient chaos. A quantitative result of these predictions are demonstrated in 6.

As indicated by the predicted return maps in Figs. S1(D1) and S1(D2), the digital twin is unable to give the final state after the transient, because such state must necessarily lie outside the invariant region from which the training data are originated. In particular, the digital twin is trained with time series data from the chaotic attractors prior to the crisis. With respect to Figs. S1(D1) and S1(D2), the digital twin can learn the dynamics within the dash green box in the plotted return maps, but is unable to predict the dynamics outside the box, as it has never been exposed to these dynamics.

A comparison of the real and predicted bifurcation diagram is demonstrated in Fig. S2. The strong resemblance between them indicate the power of the digital twin in extrapolating the correct global behavior of the target system. Moreover, this demonstrates that not only can this approach extrapolate with various driving amplitudes A (as demonstrated in the main text), but the approach can also work with varying driving frequencies Ω.

2. A driven chaotic ecological system

We study a chaotic driven ecological system that models the annual blooms of phytoplankton under seasonal driving (6). Seasonality plays a crucial role in ecological systems and epidemic spreading of infectious diseases (7), which is usually modeled as a simple periodic driving force on the system. The dynamical equations of this model in the dimensionless form are (6):

$$\frac{dN}{dt} = I - f(t)NP - qN, \tag{3}$$

$$\frac{dP}{dt} = f(t)NP - P, \tag{4}$$

where N represents the level of the nutrients, P is the biomass of the phytoplankton, the Lotka-Volterra term NP models the phytoplankton uptake of the nutrients, I represents a small and constant nutrient flow from external sources, q is the sinking rate of the nutrients to the lower level of the water unavailable to the phytoplankton, and $f(t)$ is the seasonality term: $f(t) = A\sin(\omega_{\text{eco}} t)$. The parameter values are (6): $I = 0.02$, $q = 0.0012$, and $\omega_{\text{eco}} = 0.19$.

Climate change can dramatically alter the dynamics of this ecosystem (8). We consider the task of forecasting how the system behaves if the climate change causes the seasonal fluctuation to be more extreme. In particular, suppose the training data are measured from the system when it behaves normally under a driving signal of relatively small amplitude, and we wish to predict the dynamical behaviors of the system in the future when the amplitude of the driving signal becomes larger (due to climate change). The training parameter setting is as follows. The size of the RC network is $D_r = 600$ with $D_{\text{in}} = D_{\text{out}} = 2$. The time step of the evolution of the network dynamics is $\Delta t = 0.1$. The training and validation lengths for each value of the driving amplitude A in the training are $t = 1,500$ and $t = 500$, respectively. The optimized hyperparameters of the RC are $d = 350$, $\lambda = 0.42$, $k_{\text{in}} = 0.39$, $k_c = 1.59$, $\alpha = 0.131$, and $\beta = 1 \times 10^{-7.5}$.

Figure S3 shows the results of our digital twin approach on this ecological model to learn from the dynamics under a few different values of the driving amplitude to generate the correct response of the system to a driving signal of larger amplitude. In particular, the training data are collected with the driving amplitude $A = 0.35, 0.4, 0.45$ and 0.5, all in the chaotic regions. Figures S3(A1) and S3(A2) show the true attractors of the system for $A = 0.45$ and 0.56, respectively, where the attractor is chaotic in the former case (within the training parameter regime) and periodic in the latter (outside the training regime). The corresponding attractors generated by the digital twin are shown in Figs. S3(B1) and S3(B2). The digital twin can not only replicate the chaotic behavior in the training data [Fig. S3(B1)] but also predict the transition to a periodic attractor under a driving signal with larger amplitudes (more extreme seasonality), as shown in Fig. S3(B2). In fact, the digital twin can faithfully produce the global dynamical behavior of the system, both inside and outside the training regime, as can be seen from the nice agreement between the ground-truth bifurcation diagram in Fig. S3(C) and the diagram generated by the digital twin in Fig. S3(D).

3. Continual forecasting under non-stationary external driving with sparse real-time data

The three examples (Lorenz-96 climate network in the main text, the driven CO_2 laser and the ecological system) have demonstrated that our RC based digital twin is capable of extrapolating and generating the correct statistical features of the dynamical trajectories of the target system such as the attractor and bifurcation diagram. That is, the digital twin can be regarded as a "twin" of the target system only on a statistical sense. In particular, from random initial conditions the digital twin can generate an ensemble of trajectories, and the statistics calculated from the ensemble agree with those of the original system. At the level of individual trajectories, if a target system and its digital twin start from the same initial condition, the trajectory generated by the twin can stay close to the true trajectory only for a short period of time (due to chaos). However, with infrequent state updates, the trajectory generated by the twin can shadow the true trajectory (in principle) for an arbitrarily long period of time (9), realizing *continual forecasting* of the state evolution of the target system.

In data assimilation for numerical weather forecasting, the state of the model system needs to be updated from time to time (10–12). This idea has recently been exploited to realize long-term prediction of the state evolution of chaotic systems using RC (9). Here we demonstrate that, even when the driving signal is non-stationary, the digital twin can still generate the correct state evolution of the target system. As a specific example, we use the chaotic ecosystem in Eqs. (3-4) with the same RC network trained in Sec. 2. Figure S4(A) shows the non-stationary external driving $f(t) = A(t)\sin(\omega_{\text{eco}} t)$ whose amplitude $A(t)$ increases linearly from $A(t = 0) = 0.4$ to $A(t = 2500) = 0.6$ in the time interval $[0, 2500]$. Figure S4(B) shows the true (blue) and digital-twin generated (red) time evolution of the nutrient abundance. Due to chaos, without state updates, the two trajectories diverge from each other after a few cycles of oscillation. However, even with rare state updates, the two trajectories can stay close to each other for any arbitrarily long time, as shown in Fig. S4(C). In particular, there are 800 time steps involved in the time interval $[0, 2500]$ and the state of the digital twin is updated 20 times, i.e., 2.5% of the available time series data. We will discuss the results further discussion in the next section.

4. Continual forecasting with hidden dynamical variables

In real-world scenarios, usually not all the dynamical variables of a target system are accessible. It is often the case that only a subset of the dynamical variables can be measured and the remaining variables are inaccessible or hidden from the outside world. Can a digital twin still make continual forecasting in the presence of hidden variables based on the time series data from the accessible variables? Also, Can the digital twin do this without knowing that there exists some hidden variables before training? In general, when there are hidden variables, the reservoir network needs to sense their existence, encode them in the hidden state of the recurrent layer, and constantly update them. As such, the recurrent structure of reservoir computing is necessary, because there must be a place for the machine to store and restore the implicit information that it has learned from the data. Compared with the cases where complete information about the dynamical evolution of all the observable is available, when there are hidden variables, it is significantly more challenging to predict the evolution of a target system driven by an non-stationary external signal using sparse observations of the accessible variables.

As an illustrative example, we again consider the ecosystem described by Eqs. (3) and (4). We assume that the dynamical variable N (the abundance of the nutrients) is hidden and $P(t)$, the biomass of the phytoplankton, is externally accessible. Despite the accessibility to $P(t)$, we assume that it can be measured only occasionally. That is, only sparsely updated data of the variable $P(t)$ is available. It is necessary that the digital twin is able to learn some equivalent of $N(t)$ as the time evolution of $P(t)$ also depends on the value $N(t)$, and to encode the equivalent in the reservoir network. In an actual application, when the digital twin is deployed, knowledge about the existence of such a hidden variable is not required.

Figure S5 presents a representative resulting trial, where Fig. S5(A) shows the non-stationary external driving signal $f(t)$ (the same as the one in Fig. S4(A)). Figure S5(B) shows, when the observable $P(t)$ is not updated with the real data, the predicted time series (red) $P(t)$ diverges from the true time series (blue) after about a dozen oscillations. However, if $P(t)$ is updated to the digital twin with the true values at the times indicated by the purple vertical lines in Fig. S5(C), the predicted time series $P(t)$ matches the ground truth for a much longer time. The results suggest that the existence of the hidden variable does not significantly impede the performance of continual forecasting.

The results in Fig. S5 motivate the following questions. First, has the reservoir network encoded information about the hidden variable? Second, suppose it is known that there is a hidden variable and the training dataset contains this variable, can its evolution be inferred with only rare updates of the observable during continual forecasting? Previous results (13–15) suggested that reservoir computing can be used to infer the hidden variables in a nonlinear dynamical system. Here we show that, with a segment of the time series of $N(t)$ used only for training an additional readout layer, our digital twin can forecast $N(t)$ with only occasional inputs of the observable time series $P(t)$. In particular, the additional readout layer for $N(t)$ is used only for extracting information about $N(t)$ from the reservoir network and its output is never injected back to the reservoir. Consequently, whether this additional task of inferring $N(t)$ is included or not, the trained output layer for $P(t)$ and the forecasting results of $P(t)$ are not altered.

Figure S5(D) shows that, when the observable $P(t)$ is not updated with the real data, the digital twin can to infer the hidden variable $N(t)$ for several oscillations. If $P(t)$ is updated with the true value at the times indicated by the purple vertical lines in Fig. S5(C), the dynamical evolution of the hidden variable $N(t)$ can also be accurately predicted for a much longer period of time, as shown in Fig. S5(E). It is worth emphasizing that during the whole process of forecasting and monitoring, no information about the hidden variable $N(t)$ is required - only sparse data points of the observable $P(t)$ are used.

The training and testing settings of the digital twin for the task involving a hidden variable are as follows. The input dimension of the reservoir is $D_{\text{in}} = 1$ because there is a single observable $\log_{10} P(t)$. The output dimension is $D_{\text{out}} = 2$ with one dimension of the observable $\log_{10} P(t + \Delta t)$ in addition to one dimension of the hidden variable $N(t + \Delta t)$. Because of the higher memory requirement in dealing with a hidden variable, a somewhat larger reservoir network is needed, so we use $D_r = 1,000$. The times step of the dynamical evolution of the neural network is $\Delta t = 0.1$. The training and validating lengths for each value of the driving amplitude in the training are $t = 3,500$ and $t = 350$, respectively. Other optimized hyperparameters of the reservoir are $d = 450$, $\lambda = 1.15$, $k_{\text{in}} = 0.32$, $k_c = 3.1$, $\alpha = 0.077$, $\beta = 1 \times 10^{-8.3}$, and $\sigma_{\text{noise}} = 10^{-3.0}$.

It is also worth noting that Figs. S4 and S5 have demonstrated the ability of the digital twin to extrapolate beyond the parameter regime of the target system from which the training data are obtained. In particular, the digital twin was trained only with time series under stationary external driving of the amplitude $A = 0.35, 0.4, 0.45$, and 0.5. During the testing phase associated with both Figs. S4 and S5, the external driving is non-stationary with its amplitude linearly increasing from $A = 0.4$ to $A = 0.6$. The second half of the time series $P(t)$ and $N(t)$ in Figs. S4 and S5 are thus beyond the training parameter regime.

The results in Figs. S4 and S5 help legitimize the terminology "digital twin," as the reservoir computers subject to the external driving are dynamical twin systems that evolve "in parallel" to the corresponding real systems. Even when the target system is only partially observable, the digital twin contains both the observable and hidden variables whose dynamical evolution is encoded in the recurrent neural network in the hidden layer. The dynamical evolution of the output is constantly (albeit infrequently) corrected by sparse feedback from the real system, so the output trajectory of the digital twin shadows the true trajectory of the target system. Suppose one wishes to monitor a variable in the target system, it is only necessary to read it from the digital twin instead of making more (possibly costly) measurements on the real system.

5. Digital twins under external driving with varied waveform

In the main text, it is demonstrated that dynamical noise added to the driving signal during the training can be beneficial. Figure S6 presents a comparison between the noiseless training and the training with dynamical noise of a strength $\delta_{\text{DB}} = 3 \times 10^{-3}$ (as in the main text). The ground-truth bifurcation diagram is shown in Fig. S6(A) and three examples with different reservoir neural networks for the noiseless (B1, B2, B3) and noisy (C1, C2, C3) training schemes are shown. All the settings other than the noise level are the same as that in Fig. 3 in the main text. Though there is still a fluctuation in predicted results, adding dynamical noise into the training data can produce bifurcation diagrams that are in general closer to the ground truth than without noise.

To further demonstrate the beneficial role of noise, we test the additive training noise scheme using the ecological system. The training process and hyperparameter values of the digital twin are identical to these in Ref. [2]. A dynamical noise of amplitude $\delta_{\text{DB}} = 3 \times 10^{-4}$ is added to the driving signal $f(t)$ during training in the same way as in Fig. 3 in the main text. During testing, the driving signals is altered to

$$f_{\text{test}}(t) = A_{\text{test}} \sin(\omega_{\text{eco}} t) + \frac{A_{\text{test}}}{2} \sin(\frac{\omega_{\text{eco}}}{2} t + \Delta\phi) \tag{5}$$

where $\omega_{\text{eco}} = 0.19$. Two sets of testing signals $f_{\text{test}}(t)$ are used, with $A_{\text{test}} = 0.3$ and 0.4, respectively. Figure S7 show the true and predicted bifurcation diagrams of $\log_{10} P_{max}$ versus $\Delta\phi$ for $A_{\text{test}} = 0.3$ (left column) and $A_{\text{test}} = 0.4$ (right column). It can be seen that the bifurcation diagrams generated by the digital twin with the aid of training noise are remarkably accurate. We also find that, for this ecological system, the amplitude δ_{DB} of the dynamical noise during training does not have a significant effect on the predicted bifurcation diagram. A plausible reason is that the driving signal $f(t)$ is a multiplicative term in the system equations.

6. Quantitative characterization of the performance of digital twins

In the main text, a quantitative measure of the overlapping rate between the target and predicted spanning regions is introduced to measure the performance of the digital twins, where a spanning region is the smallest simply connected region that encloses the entire attractor. In a two-dimensional projection, we divide a large reference plane into pixels of size 0.05×0.05. All the pixels through which the system trajectory crosses and those surrounded by the trajectory belong to the spanning region, and the regions covering the true attractor of the target system and predicted attractor can be compared. In particular, all the pixels that belong to one spanned region but not to the other are counted and the number is divided by the total number of pixels in the spanned region of the true attractor. This gives RESR, the relative error of the spanned regions, as described in the main text. While this measure is effective in most cases, near a bifurcation (e.g., near the boundary of a periodic window), large errors can arise because a small parameter mismatch can lead to a characteristically different attractor. To reduce the error, we test the attractors at three nearby parameter values, e.g., A and $A \pm \Delta A$ with $\Delta A = 0.005$, and choose the smallest RESR values among the three.

A direct comparison between the predicted bifurcation diagram with the ground truth is difficult given the rich information a bifurcation diagram can provide. To better quantify the here we provide another measure to quantify the performance of digital twins, we employ another measure (besides RESR). In particular, for a bifurcation diagram, the parameter values at which the various bifurcations occur are of great interest, as they define the critical points at which characteristic changes in the system can occur. To be concrete, we focus on the crisis point at which sustained chaotic motion on an attractor is destroyed and replaced by transient chaos. To characterize the performance of the digital twins in extrapolating the dynamics of the target system, we examine the errors in the predicted critical bifurcation point and in the average lifetime of the chaotic transient after the bifurcation.

As an illustrative example, we take the driven chaotic laser system in SM 1, where a crisis bifurcation occurs at the critical driving frequency $\Omega_c \approx 0.912$ at which the chaotic attractor of the system is destroyed and replaced by a non-attracting chaotic invariant set leading to transient chaos. We test to determine if the digital twin can faithfully predict the crisis point based only on training data from the parameter regime of a chaotic attractor. Let $\hat{\Omega}_c$ be the digital-twin predicted critical point. Figure S8(A) shows the distribution of $\hat{\Omega}_c$ obtained from 200 random realizations of the reservoir neural network. Despite the fluctuations in the predicted $\hat{\Omega}_c$, their average value is $\langle\hat{\Omega}_c\rangle = 0.914$, which is close to the true value $\Omega_c = 0.912$. A relative error ε_Ω of $\hat{\Omega}_c$ can be defined as

$$\varepsilon_\Omega = \frac{|\Omega_c - \hat{\Omega}_c|}{D(\Omega_c, \{\Omega_{\text{train}}\})}, \tag{6}$$

where $D(\Omega_c, \{\Omega_{\text{train}}\})$ denotes the minimal distance from Ω_c to the set of training parameter points $\{\Omega_{\text{train}}\}$, i.e., the difference between Ω_c and the closest training point. For the driven laser system, we have $D(\Omega_c, \{\Omega_{\text{train}}\}) \approx 10\%$.

The second quantity is the lifetime $\tau_{\text{transient}}$ of transient chaos after the crisis bifurcation (16, 17), as shown in Fig. S8(B). The average transient lifetime is the inverse of the slope of the linear regression of predicted data points in Fig. S8(B), which is $\langle\tau\rangle \approx 0.8 \times 10^3$. Compared with the true value $\langle\tau\rangle \approx 1.2 \times 10^3$, we see that the digital twin is able to predict the average chaotic transient lifetime to within the same order of magnitude. Considering that key to the transient dynamics is the small escaping region in Fig. S1(D2), which is sensitive to the inevitable training errors, the performance can be deemed as satisfactory.

7. Robustness of digital twin against combined dynamical/observational noises

Can our RC based digital twins withstand the influences of different types of noises? To address this question, we introduce dynamical and observational noises in the training data, which are modeled as additive Gaussian noises. Take the six-dimensional Lorenz-96 system from the main text as an example. Figure S9(A) shows the true bifurcation diagram under different amplitudes of external driving, where the vertical dashed lines specify the training points. Figures S9(B1) and S9(B2) show two realizations of the bifurcation diagram generated by the digital twin under both dynamical and observational noises of amplitudes $\sigma_{\text{dyn}} = 10^{-2}$ and $\sigma_{\text{ob}} = 10^{-2}$. Two bifurcation diagrams for noise amplitudes of an order of magnitude larger: $\sigma_{\text{dyn}} = 10^{-1}$ and $\sigma_{\text{ob}} = 10^{-1}$, are shown in Figs. S9(C1) and S9(C2). It can be seen that the additional noises have little effect on the performance of the digital twin in generating the bifurcation diagram.

8. Periodic windows of a high period: effect of long transients

Figure 1 in the main text demonstrates that the digital twin is able to predict many details of a bifurcation diagram but it fails to generate a relatively large periodic window about $A = 3.2$. A closer examination of the dynamics of the target Lorenz-96 system reveals that the periodic attractor in the window has the period 21 with a rather complicated structure, as shown in Fig. S10(A) in a two-dimensional projection. The digital twin predicts a chaotic attractor, as shown in Fig. S10(B). The reason that the digital twin fails to predict the periodic attractor lies in the long transient of the trajectory before it reaches the final attractor, as shown in Fig. S10(C). A comparison between Figs. S10(B) and S10(C) indicates that what the digital twin has predicted is in fact the transient behavior in the periodic window. The implication is that the digital twin has in fact faithfully captured the dynamical climate of the target system.

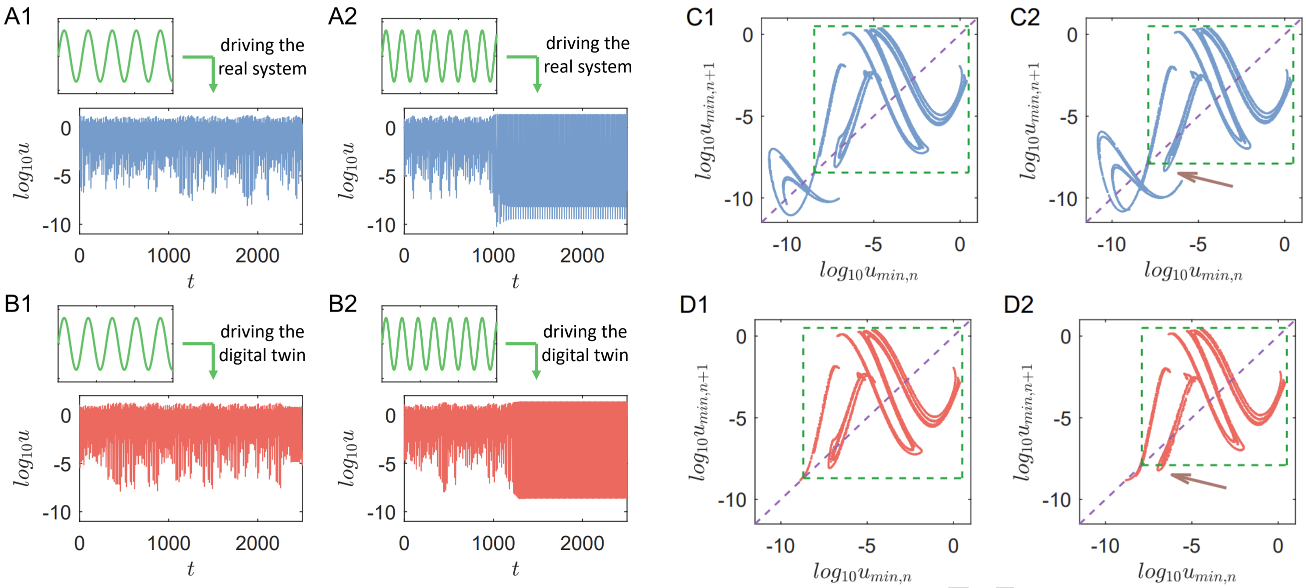

Fig. S1. Performance of digital twin of a driven $\mathrm{CO_2}$ laser system to extrapolate system dynamics under different driving frequencies. (A1, A2) True sustained and transient chaotic time series of $\log_{10} u(t)$ of the target system, for driving frequencies $\Omega = 0.905 < \Omega_c$ and $\Omega = 0.925 > \Omega_c$, respectively. The sinusoidal driving signal $f(t)$ is schematically illustrated. In (A1), the system exhibits sustained chaos. In (A2), the system settles into a periodic state after transient chaos. (B1, B2) The corresponding time series generated by the digital twin. In both cases, the dynamical behaviors generated by the digital twin agree with the ground truth in (A1, A2): sustained chaos in (B1) and transient chaos to a periodic attractor in (B2). (C1,C2) The return maps constructed from the local minima of $u(t)$ from the true dynamics, where the green dashed square defines an interval that contains the chaotic attractor in (C1) or a non-attracting chaotic set due to the escaping region (marked by the brown arrow) leading to transient chaos in (C2). (D1,D2) The return maps generated by the digital twin for the same values of Ω as in (C1,C2), respectively, which agree with the ground truth. The escaping region is successfully predicted in (D2).

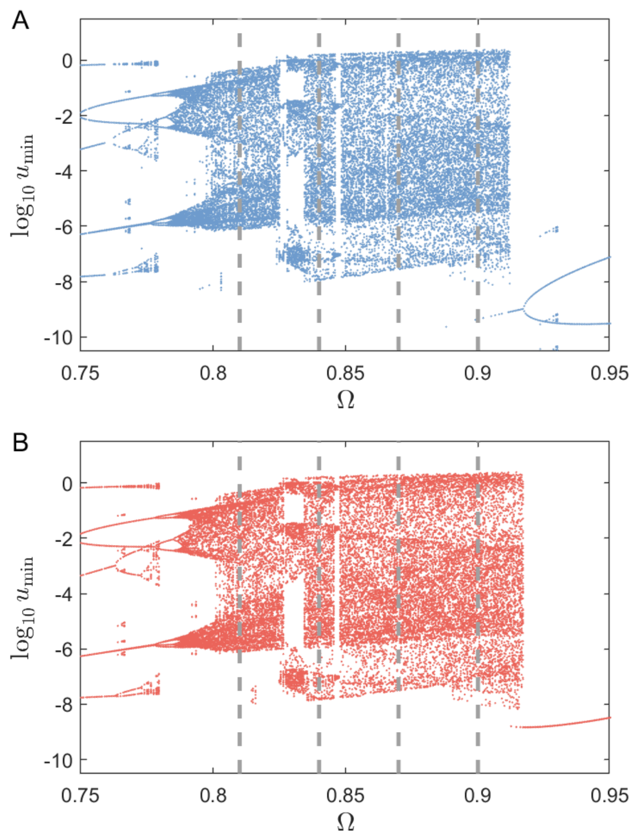

Fig. S2. Comparison of the real (A) and predicted (B) bifurcation diagrams of the driven laser system with varying driving frequencies Ω. The four vertical grey dashed lines indicate the values of driving frequencies Ω used for training the RC neural network. The strong resemblance between the two bifurcation diagrams indicates the power of the digital twin in extrapolating the correct global behavior of the target system, and demonstrates that not only can this approach extrapolate system dynamics to various driving amplitudes A, but also to varying driving frequency Ω.

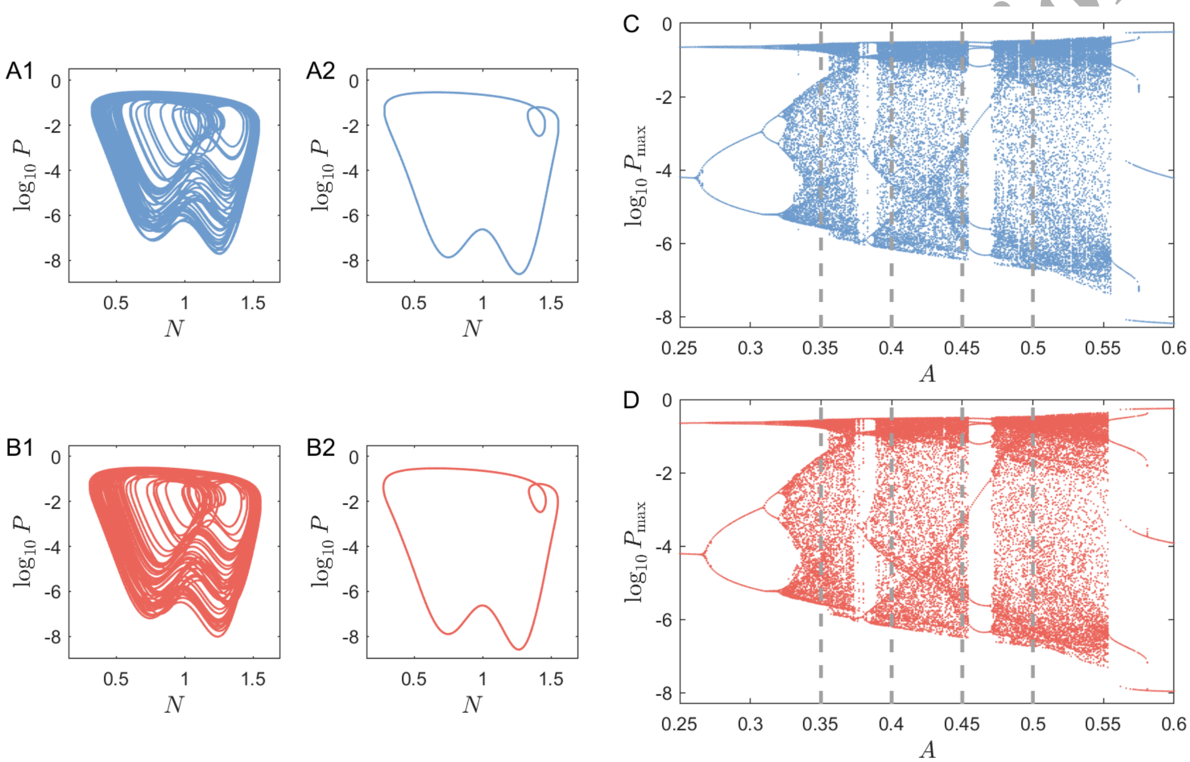

Fig. S3. Performance of the digital twin of an ecological model about blooms of phytoplankton with seasonality. The effect of seasonality is modeled by a sinusoidal driving signal $f(t) = A\sin(\omega_{\rm eco} t)$. (A1, A2) Chaotic and periodic attractors of this system in the $(N, \log_{10} P)$ plane for $A = 0.45$ and $A = 0.56$, respectively. (B1, B2) The corresponding attractors generated by the digital twin under the same driving signals $f(t)$ as in (A1, A2). The digital twin has successfully extrapolated the periodical behavior outside the chaotic training region. (C) The ground-truth bifurcation diagram of the target system. (D) The digital-twin generated bifurcation diagram. In (C) and (D), the four vertical grey dashed lines indicate the values of driving amplitudes A used for training the RC network. The strong resemblance between the two bifurcation diagrams indicates the power of the digital twin in extrapolating the correct global behavior of the target system.

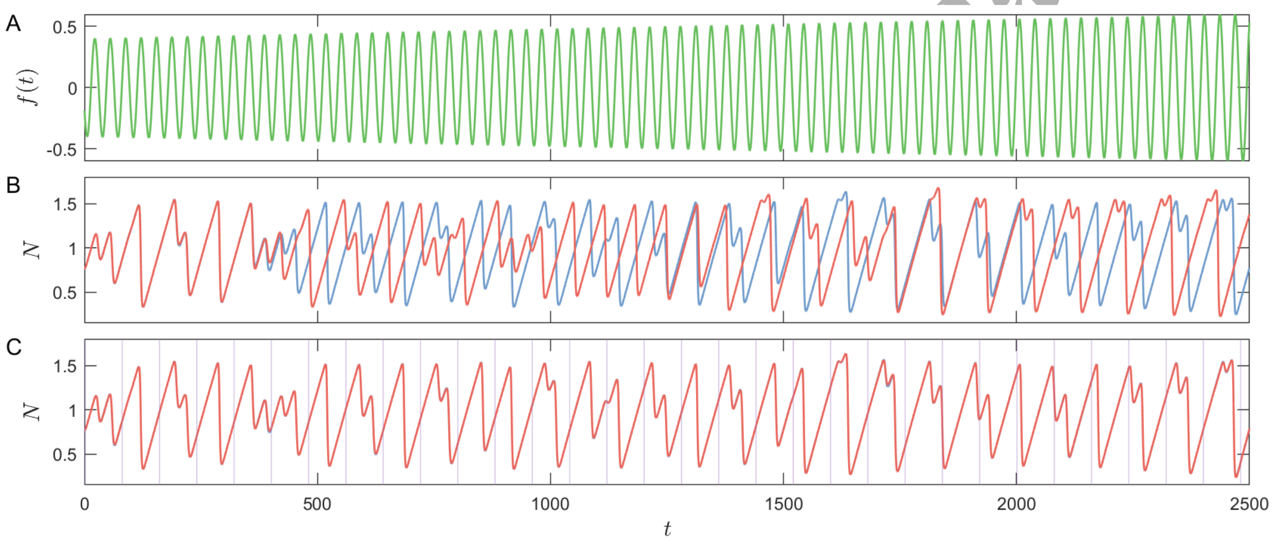

Fig. S4. Continual forecasting of the chaotic ecological system under non-stationary external driving $f(t)$ and with sparse updates of the dynamical variables. (A) A nonstationary sinusoidal driving signal $f(t)$ whose amplitude increases with time. The task for the digital twin is to forecast the response of the chaotic target system under this driving signal for a relatively long term. (B) The trajectory generated by the digital twin (red) in comparison with the true trajectory (blue). For $t \in [0, 400]$, the two trajectories match each other with small errors, but the digital-twin generated trajectory begins to deviate from the true trajectory at $t \sim 400$ (due to chaos). (C) With only sparse updates from real data at times indicated by the vertical lines (2.5% of the time steps in the given time interval), the digital twin can make relatively accurate predictions for a long term, demonstrating the ability to perform continual forecasting.

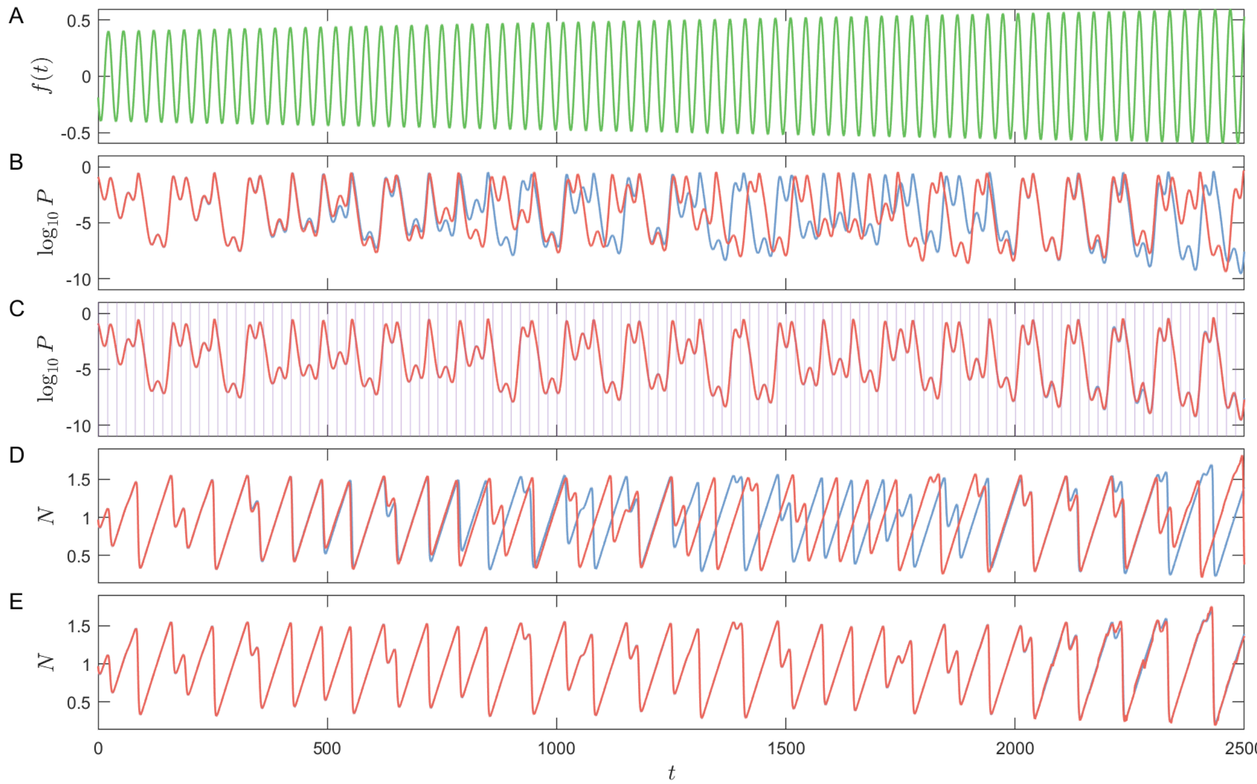

Fig. S5. Continual forecasting and monitoring of a hidden dynamical variable in the chaotic ecological system under non-stationary external driving with sparse updates from the observable. The system is described by Eqs. (3) and (4). The dynamical variable $N(t)$ is hidden, and the other variable $P(t)$ is externally accessible but only sparsely sampled measurement of it can be performed. (A) The non-stationary sinusoidal driving signal $f(t)$ with a time-varying amplitude. (B) Digital-twin generated time evolution of the accessible variable $P(t)$ (red) in comparison with the ground truth (blue) in the absence of any state update of $P(t)$. The predicted time evolution quickly diverges from the true behavior. (C) With sparse updates of $P(t)$ at the times indicated by the purple vertical lines (10% of the times steps), the digital twin is able to make an accurate forecast of $P(t)$. (D) Digital-twin generated time evolution of the hidden variable $N(t)$ (red) in comparison with the ground truth (blue) in the absence of any state update of $P(t)$. (E) Accurate forecasting of the hidden variable $N(t)$ with sparse updates of $P(t)$.

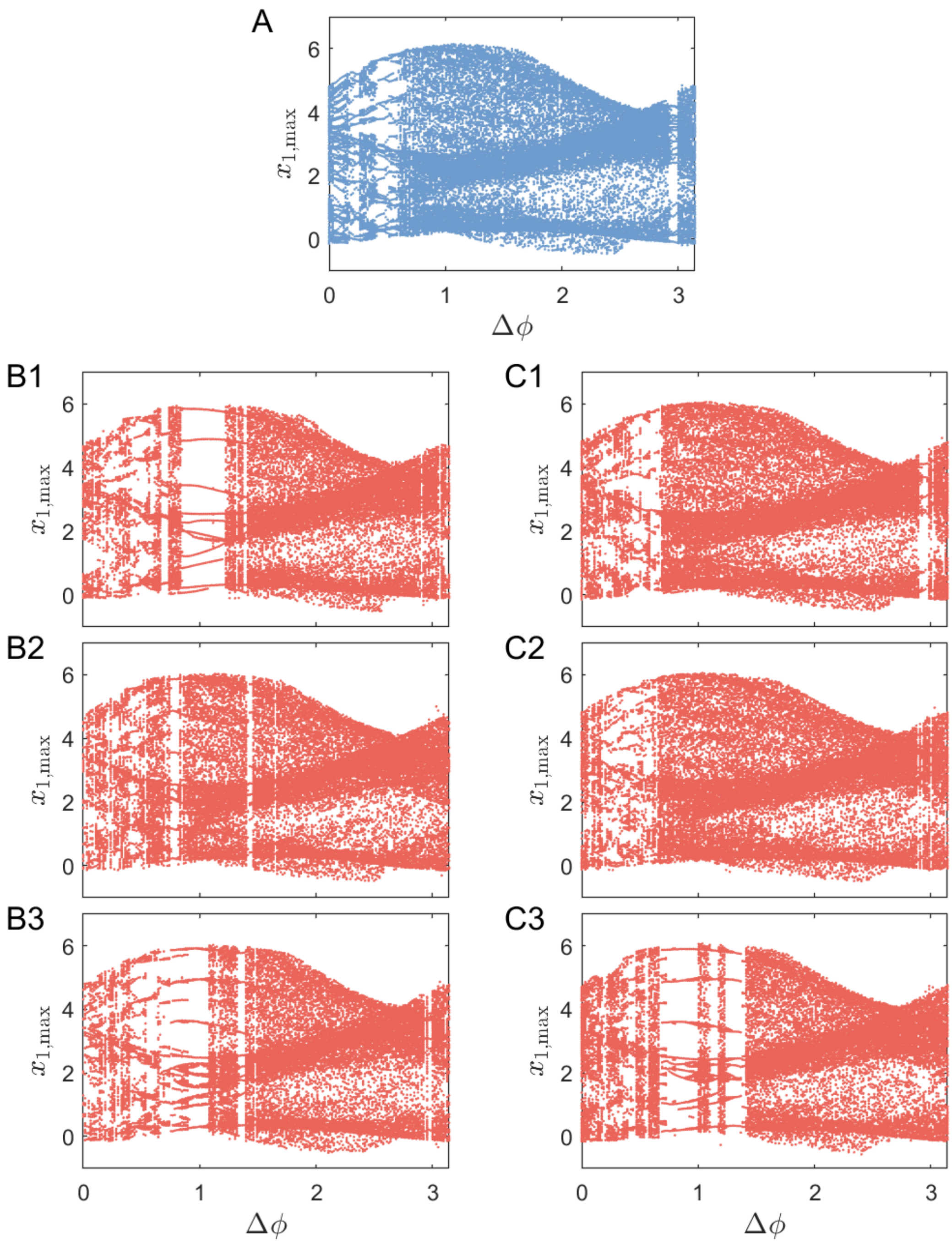

Fig. S6. Comparisons of the prediction performance between the noiseless (left) and noisy (right) cases on the task of predicting under external driving with different waveform. The target system is a six-dimensional Lorenz-96 system. Panel (A) shows the true bifurcation diagram. Panels (B1-B3) show the prediction results without any dynamical noise in the training data with three realizations of the reservoir network. Panels (C1-C3) show the prediction results with dynamical noise of a strength $\delta_{\text{DB}} = 3 \times 10^{-3}$ in the training data. The settings are the same as that in Fig. 3 in the main text.

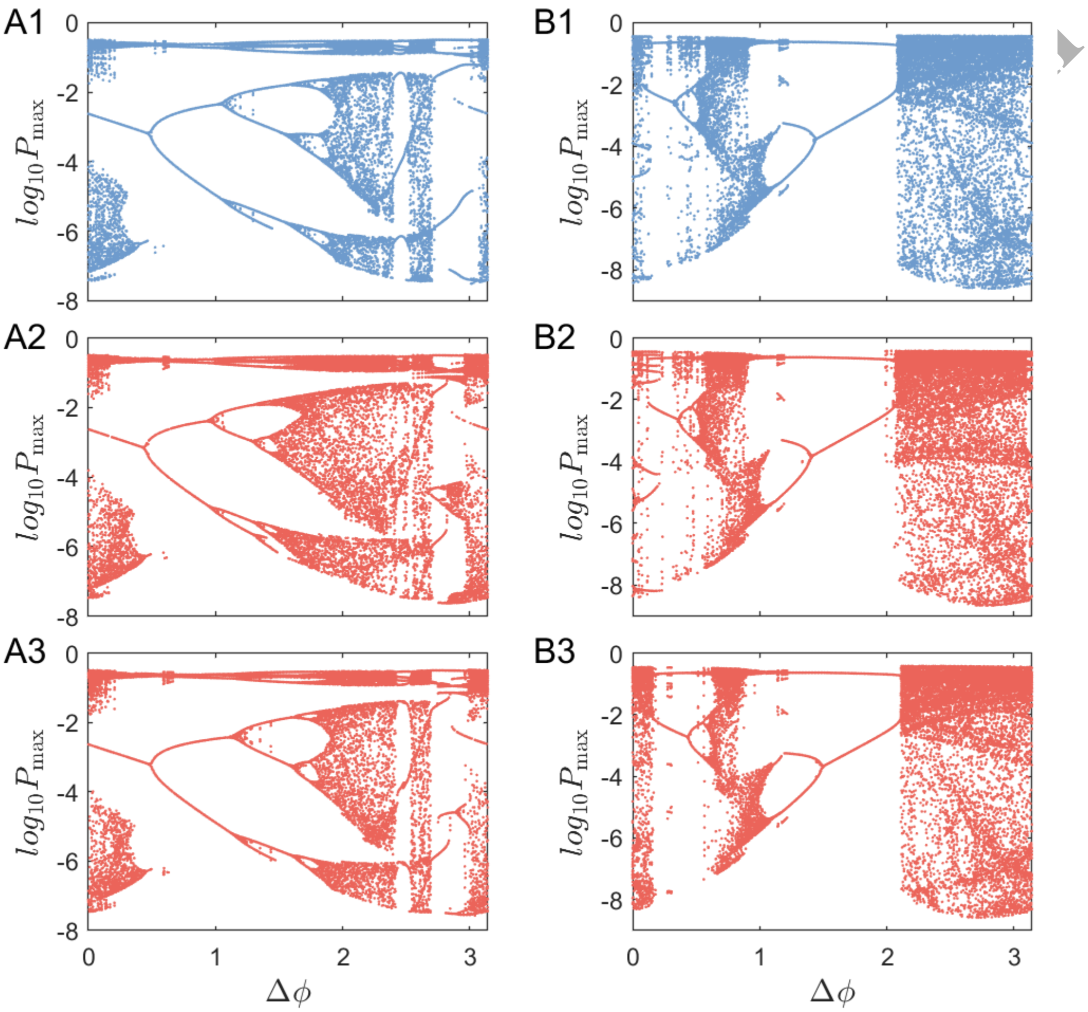

Fig. S7. Performance of the digital twin with the ecological model under driving signals with waveform different from the training set. The testing driving signals are described by Eq. 5 while the training driving signals are sinusoidal waves with small dynamical noise. (A1) The real bifurcation diagram for $A_{\text{test}} = 0.3$. (A2, A3) Predicted bifurcation diagrams for $A_{\text{test}} = 0.3$ with two random realizations of the reservoir networks. (B1-B3) Same as (A1-A3) but with $A_{\text{test}} = 0.4$.

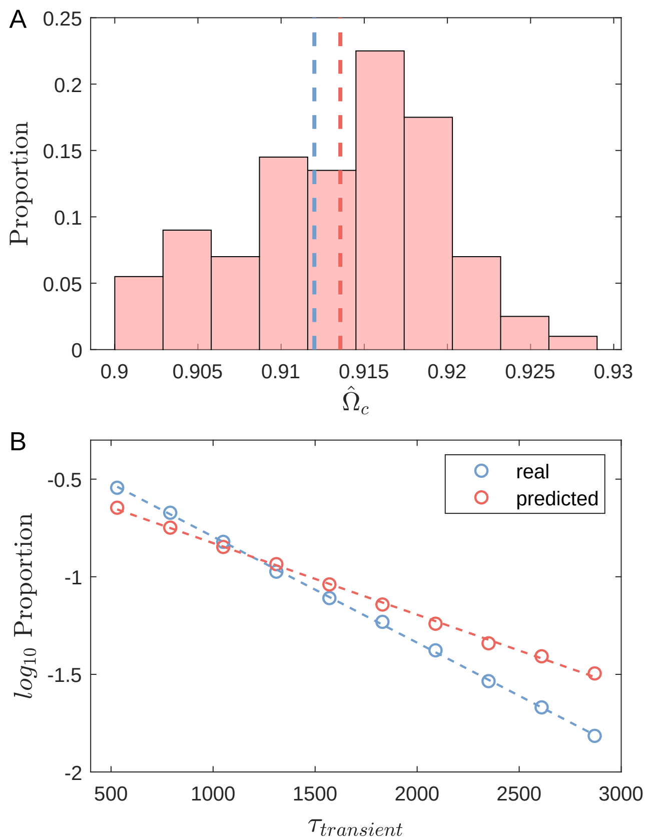

Fig. S8. Quantitative performance of the digital twin for a chaotic driven laser system. (A) Distribution of the predicted values of the crisis bifurcation point $\hat{\Omega}_c$, at which a chaotic attractor is destroyed and replaced by a non-attracting chaotic invariant set leading to transient chaos. The blue and red vertical dashed lines denote the true value $\Omega_c \approx 0.912$ and the average predicted value $\langle \hat{\Omega}_c \rangle$, respectively, where 200 random realizations of the reservoir neural network are used to generate this distribution. Despite the fluctuations in the predicted crisis point, the ensemble average value of the prediction is quite close to the ground truth. (B) Exponential distribution of the lifetime of transient chaos slightly beyond the crisis point: true (blue) and predicted (red) behaviors. The predicted distribution is generated using 100 random reservoir realizations, each with 200 random initial 'warming up" data.

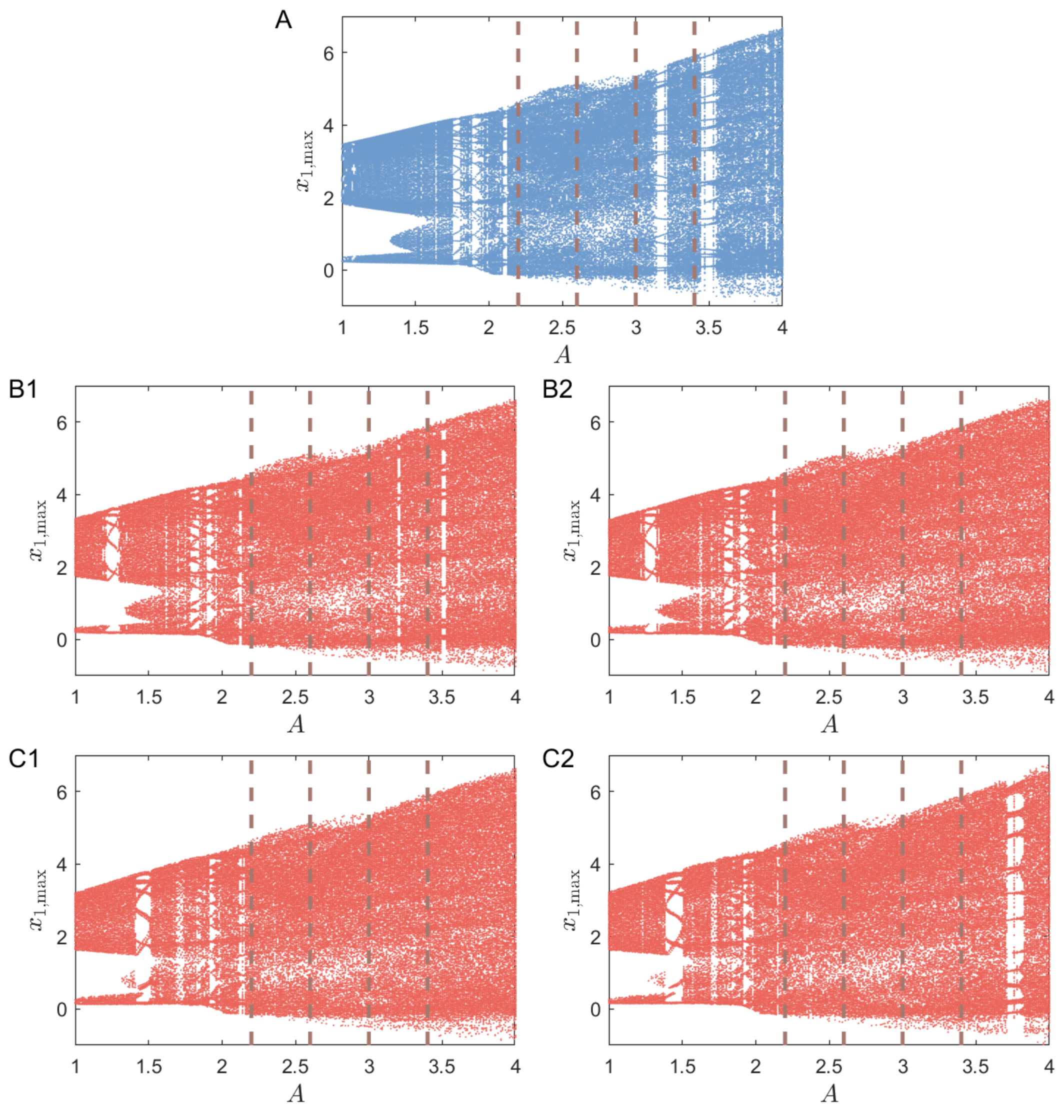

Fig. S9. Robustness of digital twin against combined dynamical and observational noises. The setting is the same as that in Fig. 1 in the main text, except with additional noises in the training data. (A) A true bifurcation diagram of the six-dimensional Lorenz-96 system. (B1, B2) Two examples of the bifurcation diagram predicted by the digital twin with training data under dynamical noise of amplitude $\sigma_{\text{dyn}} = 10^{-2}$ and observational noise of amplitude $\sigma_{\text{ob}} = 10^{-2}$. (C1, C2) Two examples of the predicted bifurcation diagrams under the two kinds of noise with $\sigma_{\text{dyn}} = 10^{-1}$ and $\sigma_{\text{ob}} = 10^{-1}$. Both the dynamical and observational noises are additive Gaussian processes. It can be seen that though larger additional noises make the predicted details less accurate, the general shapes of the predicted results are not harmed significantly. The settings of the training data and reservoir neural networks are the same as those in Fig. 2 in the main text. The dynamical noises are added to the dynamical equations of the state variables. There is no noise in the sinusoidal external driving.

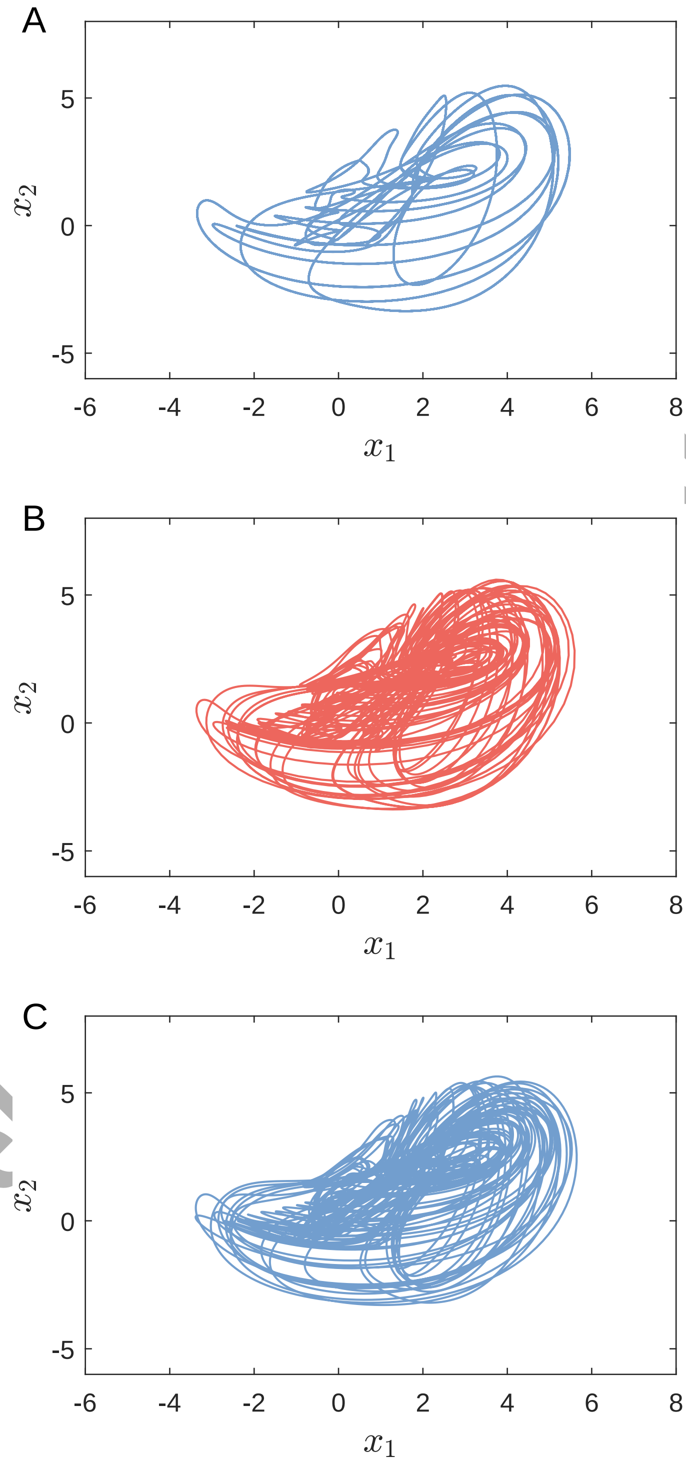

Fig. S10. Origin of the failure of the digital twin in predicting the periodic window in Fig. 1(C) in the main text. (A) A two-dimensional portrait of the periodic attractor of period-21 in the Lorenz-96 system for $A = 3.2$. (B) The digital-twin predicted chaotic attractor. (C) The transient behavior of the target Lorenz-96 system for $A = 3.2$. The remarkable resemblance between (B) and (C) suggests that the trained digital twin has faithfully captured the dynamical climate of the target system.

References

1. D Dangoisse, P Glorieux, D Hennequin, Laser chaotic attractors in crisis. *Phys. Rev. Lett.* **57**, 2657 (1986).
2. D Dangoisse, P Glorieux, D Hennequin, Chaos in a CO_2 laser with modulated parameters: experiments and numerical simulations. *Phys. Rev. A* **36**, 4775 (1987).
3. HG Solari, E Eschenazi, R Gilmore, JR Tredicce, Influence of coexisting attractors on the dynamics of a laser system. *Opt. Commun.* **64**, 49–53 (1987).
4. IB Schwartz, Sequential horseshoe formation in the birth and death of chaotic attractors. *Phys. Rev. Lett.* **60**, 1359 (1988).
5. C Grebogi, E Ott, JA Yorke, Crises, sudden changes in chaotic attractors and chaotic transients. *Phys. D* **7**, 181–200 (1983).
6. A Huppert, B Blasius, R Olinky, L Stone, A model for seasonal phytoplankton blooms. *J. Theo. Biol.* **236**, 276–290 (2005).
7. L Stone, R Olinky, A Huppert, Seasonal dynamics of recurrent epidemics. *Nature* **446**, 533–536 (2007).
8. M Winder, U Sommer, Phytoplankton response to a changing climate. *Hydrobiologia* **698**, 5–16 (2012).
9. H Fan, J Jiang, C Zhang, X Wang, YC Lai, Long-term prediction of chaotic systems with machine learning. *Phys. Rev. Res.* **2**, 012080 (2020).
10. E Kalnay, *Atmospheric Modeling, Data Assimilation and Predictability.* (Cambridge university press), (2003).
11. M Asch, M Bocquet, M Nodet, *Data Assimilation: Methods, Algorithms, and Applications.* (SIAM), (2016).
12. A Wikner, et al., Using data assimilation to train a hybrid forecast system that combines machine-learning and knowledge-based components. *Chaos* **31**, 053114 (2021).
13. Z Lu, et al., Reservoir observers: Model-free inference of unmeasured variables in chaotic systems. *Chaos* **27**, 041102 (2017).
14. ZS Roland, U Parlitz, Observing spatio-temporal dynamics of excitable media using reservoir computing. *Chaos* **28**, 043118 (2018).
15. T Weng, H Yang, C Gu, J Zhang, M Small, Synchronization of chaotic systems and their machine-learning models. *Phys. Rev. E* **99**, 042203 (2019).
16. LW Kong, HW Fan, C Grebogi, YC Lai, Machine learning prediction of critical transition and system collapse. *Phys. Rev. Res.* **3**, 013090 (2021).
17. LW Kong, H Fan, C Grebogi, YC Lai, Emergence of transient chaos and intermittency in machine learning. *J. Phys. Complex.* **2**, 035014 (2021).