Model-free analysis of complex systems using delayed-feedback echo state network

Wenjing Guo [1], Shuang Li [1], Jianming Liu [2], Eric Li [3] and Xu Xu [1,*]
[1] *College of Mathematics, Jilin University, 2699 Qianjin Street, Changchun 130012, China*
[2] *School of Mathematics, China University of Mining and Technology, Xuzhou 221116, China*
[3] *School of Computing, Engineering & Digital Technologies, Teesside University, Middlesbrough TS1 3BX, United Kingdom*

(Received 24 September 2025; accepted 16 June 2026; published 8 July 2026)

Investigating the dynamical behavior of complex systems from data poses formidable challenges in the field of nonlinear science. This paper proposes a delayed-feedback echo state network (df-ESN) model specifically designed for modeling and analyzing complex systems from data. The df-ESN introduces the delayed feedback of the reservoir state to reflect the finite transmission speed of signals among reservoir units. The delayed feedback term links the current reservoir state to its value m steps ago, thereby significantly enhancing the memory capacity. By adopting the Lyapunov-Krasovskii stability method, we establish delay-dependent stability criteria to ensure the global echo state property (ESP). Furthermore, the definition of local echo state property (local ESP) is proposed to weaken the restrictive constraints imposed by the conventional ESP condition. The df-ESN also offers a strategy to determine the optimal parameters in the delayed feedback mechanism via a grid search technique, thus overcoming the challenges of determining optimal reservoir parameters in the ESN. This work provides a comprehensive theoretical understanding for the memory capability, local/global echo state property, and the impact of the noise on the model performance. Various nonlinear dynamical problems are conducted to verify the effectiveness of df-ESN, including chaotic time series prediction, high-dimensional spatiotemporal system analysis, bifurcation diagram reconstruction, and basin of attraction prediction. Moreover, df-ESN is used to study the model-free dynamical analysis of double soliton solutions of the nonlinear Schrödinger equation. The results underscore the capability of the proposed df-ESN as a model-free strategy for analyzing the complex systems, and they contribute to a deeper understanding of the underlying theoretical aspects of the model.

DOI: 10.1103/chvg-gfy8

I. INTRODUCTION

In real-world scenarios, numerous complex systems consist of a large number of interacting individuals and exhibit sophisticated dynamic behaviors. In conventional modeling approaches, the governing equations of such systems are usually derived based on their inherent physical laws, such as momentum principles or energy conservation laws. Nevertheless, many nonlinear systems are overly complex to formulate accurate mathematical models, which imposes substantial challenges on theoretical analysis and investigation.

The data-driven approach is the methodology that relies primarily on data rather than on explicit rules or expert heuristics to make decisions, generate insights, or build predictive models. The core idea of a data-driven method lies in analyzing and mining the information and patterns within the data, thereby better understanding the problem, predicting outcomes, or assisting in decision-making [1,2].

Machine learning (ML) is the computational engine and the key technology that operationalizes data-driven thinking. It encompasses various algorithms and approaches that allow computers to improve their performance on tasks through experience (or data). These methods include supervised learning, unsupervised learning, and reinforcement learning [3,4]. As a core technique within machine learning, artificial neural networks (ANNs) enable systems to learn from data without explicit programming for each task [5]. The ANNs are especially effective for tasks involving large datasets and intricate patterns, such as image recognition [6], natural language processing [7], and predictive analytics [8,9].

The echo state network (ESN) is a special type of artificial neural network distinguished by a fixed, randomly initialized reservoir of sparsely connected neurons and a trainable linear readout layer. As a representative model of reservoir computing [10], the ESN has some advantages compared with the conventional ANNs: (i) The fixed reservoir performs a high-dimensional nonlinear expansion of inputs, with no need for gradient-based training of recurrent weights. (ii) A spectral radius less than 1 ensures fading memory and inherent stability, naturally avoiding the vanishing/exploding gradient problem. (iii) Only the readout weights are learned via linear/ridge regression, reducing training to a convex optimization problem and making ESNs orders of magnitude faster to train than fully recurrent models such as long short-term memory models (LSTMs) or gated recurrent units (GRUs) [11].

Recently, the ESN models have been applied in many fields of science and engineering, including pattern recognition [12–14], time-series classification [15–17], image processing

*Contact author: xuxu@jlu.edu.cn

 ©2026 American Physical Society

[18–23], signal processing [24], and the chaotic time series prediction [25–28].

Jaeger and Haas [29] demonstrated that the ESN achieved a very high prediction accuracy on the Mackey-Glass chaotic time series, with an improvement of 2400 times compared to traditional methods. Reference [30] proposed a parallel reservoir structure for predicting high-dimensional systems and verified its effectiveness on the Kuramoto-Sivashinsky chaotic system. The study in Ref. [31] employed the ESN with a parameter-aware reservoir, which not only successfully predicts the bifurcation diagram, but also captures system behaviors across a wide range of parameters and identifies attractor basins in multistable systems. Reference [32] proposed a local information flow ESN for initial-value-aware prediction, capable of modeling dynamic systems with varying initial states and detecting attractor basins of the systems. Hybrid architectures like attention-enhanced reservoir computing (AERC) [33] further improve long-term dependency modeling via attention-weighted reservoir states.

Despite the advantages of the ESNs, they still face limitations in modeling complex temporal dependencies. (i) Finding an appropriate reservoir structure and hyperparameter set can be time-consuming and may require expert knowledge, and the poor choices can lead to unstable or suboptimal behavior. (ii) Because the reservoir is untrained and fixed, ESNs often struggle to retain long-range temporal dependencies, especially when compared with fully trained architectures such as LSTM or GRU. (iii) The linear readout layer and random reservoir may be insufficient for tasks that require sophisticated nonlinear feature extraction or hierarchical representations. (iv) For highly intricate or high-dimensional spatiotemporal problems, ESNs typically underperform deeper, fully trained recurrent architectures.

Recently, some improved ESN models have been proposed by integrating past information, such as the leaky echo state network (L-ESN) [34], the hierarchical delayed memory echo state network [35], the single-node delay-based echo state network [36], and the delayed self-feedback echo state network [37]. These models primarily emphasize structural modifications to enhance memory capacity, although most of them lack a comprehensive theoretical analysis of the impact of the parameters on the model performance.

This paper proposes an echo state network with a delay feedback mechanism, termed `df`-ESN, as a data-driven framework for analyzing the complex dynamics of nonlinear systems. The key contributions are as follows:

(i) *Introducing a delayed feedback mechanism*: The `df`-ESN introduces the delayed feedback of the reservoir into the ESN to effectively simulate the delay effect of biological neurons, which improve the memory capability.

(ii) *Reducing computational and design complexity*: A two-stage grid searching technique is proposed to identify the optimal values of feedback gain and delay in the delayed mechanism, effectively circumventing the challenges typically faced by the existing ESN models in determining the optimal reservoir parameters.

(iii) *Providing a theoretical understanding*: By analyzing the echo state property, the memory capability, and the noise robustness, the `df`-ESN investigates the comprehensive theoretical analysis of the model, and provides an in-depth understanding of the impact of the parameters on model performance.

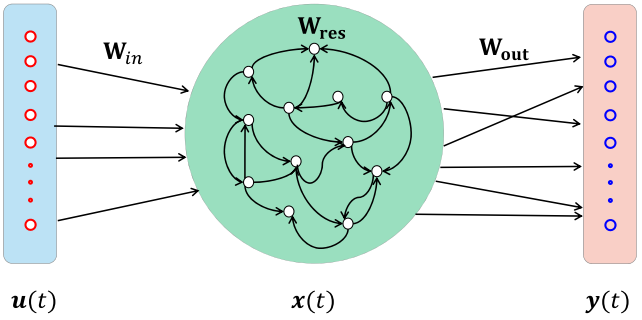


FIG. 1. Schematic diagram of an ESN.

II. ECHO STATE NETWORKS

As shown in Fig. 1, an ESN consists of an input layer, an output layer, and a reservoir layer containing a large number of sparsely and randomly connected neurons. Denote the input and the reservoir state at time t as $\mathbf{u}_t$ and $\mathbf{x}_t$, respectively. Then the reservoir state at time t is [10]

$$\mathbf{x}_{t+1} = f(\mathbf{W}_{\mathrm{in}}\mathbf{u}_{t+1} + \mathbf{W}_{\mathrm{res}}\mathbf{x}_t), \tag{1}$$

where $\mathbf{u}_t \in \mathbb{R}^{d_{\mathrm{in}}}$, $\mathbf{x}_t \in \mathbb{R}^N$, $\mathbf{W}_{\mathrm{in}} \in \mathbb{R}^{N\times d_{\mathrm{in}}}$, $\mathbf{W}_{\mathrm{res}} \in \mathbb{R}^{N\times N}$, d_{in} is the input dimension, N is the size of the reservoir, $\mathbf{W}_{\mathrm{in}}$ is the input-reservoir weight matrix, $\mathbf{W}_{\mathrm{res}}$ is the reservoir-reservoir weight matrix, and $f(.)$ is the activation function.

The output of the network is $\mathbf{y}_t = \mathbf{W}_{\mathrm{out}}[\mathbf{u}_t; \mathbf{x}_t]$, where $\mathbf{W}_{\mathrm{out}} \in \mathbb{R}^{d_{\mathrm{out}}\times(d_{\mathrm{in}}+N)}$, $\mathbf{y}_t \in \mathbb{R}^{d_{\mathrm{out}}}$, d_{out} is the dimension of output and $\mathbf{W}_{\mathrm{out}}$ is the reservoir-to-output weight matrix, which is trained using a simple rule such as linear regression. It is noted that only the reservoir-to-output weight matrix $\mathbf{W}_{\mathrm{out}}$ is trained and all other parameters are fixed at the initial construction.

During training, $\mathbf{x}_t$ are stacked into a state collection $\mathbf{X}$, and the desired output $\mathbf{y}_*(t)$ of the ESN are stacked into a vector $\mathbf{Y}$. When the activation function of the output units is the identity function, $\mathbf{W}_{\mathrm{out}}$ can be calculated by $\mathbf{W}_{\mathrm{out}} = \mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \xi\mathbf{I})^{-1}$, where $\mathbf{I}$ is the identity matrix, and ξ is Tikhonov regularization parameter.

In the prediction stage, the predicted output ($\mathbf{y}_t$) is fed back into the reservoir input, i.e., ($\mathbf{u}_{t+1} = \mathbf{y}_t$) for a free-running prediction. Specifically, after $\mathbf{W}_{\mathrm{out}}$ is obtained through training, the predicted $\mathbf{y}_{t+1}$ can be derived from the following equation:

$$\mathbf{x}_{t+1} = f(\mathbf{W}_{\mathrm{in}}\mathbf{y}_t + \mathbf{W}_{\mathrm{res}}\mathbf{x}_t), \quad \mathbf{y}_{t+1} = \mathbf{W}_{\mathrm{out}}[\mathbf{y}_t; \mathbf{x}_{t+1}]. \tag{2}$$

To improve the performance of ESNs, various enhanced variants have been developed. The leaky echo state network (L-ESN) [34] is proposed by introducing a critical leakage rate parameter α ($0 < \alpha \leqslant 1$) into the ESN, resulting in the following modified equation:

$$\mathbf{x}_{t+1} = (1-\alpha)\mathbf{x}_t + \alpha f(\mathbf{W}_{\mathrm{in}}\mathbf{u}_{t+1} + \mathbf{W}_{\mathrm{res}}\mathbf{x}_t). \tag{3}$$

By tuning the leakage rate α, L-ESN can effectively adapt to input signals with varying temporal characteristics.

The deep echo state network (deep-ESN) [38] improves representational ability by stacking multiple reservoir layers

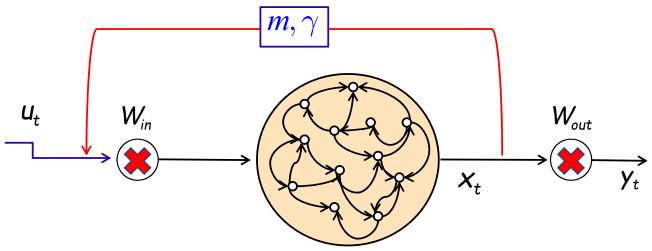


FIG. 2. Diagrammatic sketch of the df-ESN in training.

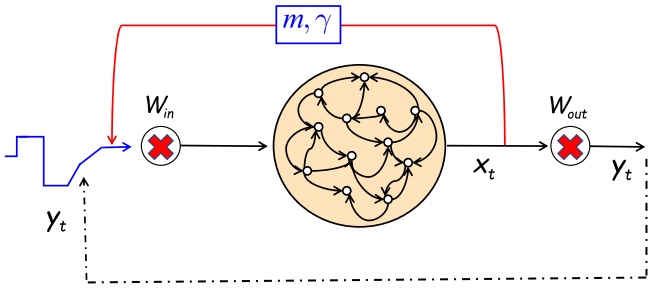


FIG. 3. Diagrammatic sketch of df-ESN for prediction.

hierarchically. The input signal $\mathbf{u}_t$ drives the first layer, whose output feeds into subsequent layers. Each layer's hidden state is updated similarly to ESN or L-ESN, and the final output $\mathbf{y}_t$ is derived from the topmost or combined reservoir states.

Attention-enhanced reservoir computing (AERC) [33] improves the model's performance by integrating an attention mechanism into the output layer. It dynamically generates attention weights $\mathbf{W}_{\text{att},t}$ from the current reservoir state via a lightweight feedforward network F:

$$\mathbf{W}_{\text{att},t} = F(\mathbf{W}_{\text{net}}, \mathbf{x}_t), \quad \mathbf{y}_t = \mathbf{W}_{\text{att},t}\mathbf{x}_t,$$

where $\mathbf{W}_{\text{net}}$ denotes the trainable parameters of F. This structure enables input-dependent adaptive readout weights, enhancing the model's predictive adaptability.

III. DELAYED FEEDBACK ECHO STATE NETWORK

In this paper, we propose a delayed feedback echo state network (df-ESN) by introducing the delayed feedback of the reservoir into the ESN. The reservoir dynamics of df-ESN is described by

$$\mathbf{x}_{t+1} = \gamma\mathbf{x}_{t-m} + f(\mathbf{W}_{\text{in}}\mathbf{u}_{t+1} + \mathbf{W}_{\text{res}}\mathbf{x}_t), \tag{4}$$

where $\mathbf{x}_{t-m} = \mathbf{x}(t-m)$ is the delayed feedback of the reservoir with feedback gain γ and delay steps $m \geqslant 0$. The diagrammatic sketch of df-ESN is shown in Fig. 2.

In Eq. (4), the delayed feedback $\gamma\mathbf{x}(t-m)$ successfully simulates the finite transmission speed between the biological neurons. When $\gamma = 0$, the model reverts to the standard ESN model, and when $m = 0$, its behavior is similar to that in the L-ESN model. So, the proposed df-ESN model extends the standard ESN model, and exhibits flexibility in the model configuration.

Compared with existing ESN variants, the proposed df-ESN exhibits the following superior features:

(i) *Enhanced dynamic modeling capability*: By introducing a delayed feedback term $\gamma x(t-m)$ to embed historical state information, the df-ESN effectively strengthens its temporal memory, enriches system dynamics, and improves the nonlinear fitting ability. It enables more precise modeling for delay-involved complex systems such as biological and chemical processes.

(ii) *Improved stability and robustness*: The embedded time-delay mechanism can stabilize the reservoir dynamics, thereby enhancing the network's robustness against noise and external disturbances.

(iii) *Low-cost parameter tuning*: For the df-ESN, only the delay parameter needs to be optimized via a grid search to obtain the best performance, while reservoir parameters are just coarsely initialized and require no further fine-tuning. This design effectively reduces computational burden and tedious trial-and-error work.

The training procedure of the proposed df-ESN is consistent with that of a standard ESN, except that the reservoir state $\mathbf{x}_t$ is updated via Eq. (4). In the prediction stage, the value of $\mathbf{y}(t)$ is fed back into the reservoir input for a free-running prediction as shown in Fig. 3. Specifically, after $\mathbf{W}_{\text{out}}$ is obtained through training, the predicted $\mathbf{y}_{t+1}$ can be derived as

$$\mathbf{x}_{t+1} = \gamma\mathbf{x}(t-m) + f(\mathbf{W}_{\text{in}}\mathbf{y}_t + \mathbf{W}_{\text{res}}\mathbf{x}_t),$$
$$\mathbf{y}_{t+1} = \mathbf{W}_{\text{out}}[\mathbf{y}_t; \mathbf{x}_{t+1}].$$

A. Global/local echo state property

The echo state property (ESP) is an important property for ensuring that the state of the reservoir is uniquely determined by the history of the input data, but not by the initial values of the reservoir state. Specifically, if the network has the ESP, then for any two initial values of the reservoir state $\mathbf{x}(0)$ and $\mathbf{x}'(0)$, $\| \mathbf{x}(t) - \mathbf{x}'(t) \| \to 0$ holds true for a sufficiently large time.

Assumption 1 (Lipschitz condition for the activation function). The nonlinear activation function $f(\cdot)$ satisfies the global Lipschitz condition on a set $D \subseteq \mathbb{R}^N$, i.e., $\|f(x) - f(x')\| \leqslant L_f\|x - x'\|$, with $f(0) = 0$ and Lipschitz constant $L_f > 0$ [e.g., $L_f = 1$ for $\tanh(\cdot)$].

Let $\mathbf{x}_t^{(1)}$ and $\mathbf{x}_t^{(2)}$ be two state sequences generated by Eq. (4) with the same input sequence $\{\mathbf{u}_t\}$ but different initial conditions. Define the error state as $\mathbf{e}_t = \mathbf{x}_t^{(1)} - \mathbf{x}_t^{(2)}$, and $\widetilde{\mathbf{x}}^{(i)} = \mathbf{W}_{\text{in}}\mathbf{u}_{t+1} + \mathbf{W}_{\text{res}}\mathbf{x}_t^{(i)} (i = 1, 2)$. Subtracting the dynamics of the two sequences yields the error system:

$$\mathbf{e}_{t+1} = \gamma\left(\mathbf{x}_{t-m}^{(1)} - \mathbf{x}_{t-m}^{(2)}\right) + f(\widetilde{\mathbf{x}}^{(1)}) - f(\widetilde{\mathbf{x}}^{(2)}). \tag{5}$$

Taking the norm of both sides and applying the Lipschitz condition, we obtain

$$\|\mathbf{e}_{t+1}\| \leqslant |\gamma|\|\mathbf{e}_{t-m}\| + L_f\|\mathbf{W}_{\text{res}}\|\|\mathbf{e}_t\|, \tag{6}$$

where $\|\mathbf{W}_{\text{res}}\|$ is the spectral norm (largest singular value) of the matrix $\mathbf{W}_{\text{res}}$.

According to the stability theory, the ESP implies the stability of the trivial equilibrium of error system (5). In this section, the Lyapunov-Krasovski stability is used to give the delay-dependent ESP conditions for any initial conditions of the reservoir states.

Theorem 1. Consider the *df*-ESN with the activation function $f(\cdot)$ satisfing Assumption 1. If there exists a constant $\mu > 0$ such that

$$|\gamma| + L_f \|\mathbf{W}_{\text{res}}\| + \mu \text{m} < 1, \tag{7}$$

then the system satisfies the global echo state property. That is, for any initial reservoir states, the error system is globally uniformly asymptotically stable, and thus $\lim_{t\to\infty} \|\mathbf{e}_t\| = 0$.

The proof of *Theorem 1* can be found in Appendix A.

Parameter μ is chosen to suppress large delayed perturbations while satisfying the stability condition. For large delays, γ and $\|\mathbf{W}_{\text{res}}\|$ are reduced to maintain a valid stability budget, meaning that μ, even if small in magnitude, will be effective in providing the required delay penalty.

In practice, ESN reservoir matrices are usually rescaled by their spectral radius ρ instead of the spectral norm, leveraging the strong correlation between the two quantities for random non-normal matrices. Adjusting the spectral radius to a stability-conducive range ensures the spectral norm remains bounded, meeting the ESP conditions. So, in this section, we use the spectral radius ρ to derive the conditions for the ESP, as given in Theorem 1: $|\gamma| + L_f \rho + \mu m < 1$.

Remark 1. From Eq. (7), increasing the time delay step m leads to diminishing returns. Theorem 1 indicates that a large delay necessitates a feedback gain $|\gamma|$ far less than 1 to guarantee the ESP. Therefore, a tradeoff must be struck between the incorporation of large delay information and the maintenance of adequate delay strength. In addition, in the implementation of the algorithm, if the delay step size m is set to an excessively large value, the model will attempt to mine patterns from (potentially noisy) early input data. This will cause the output weights W_{out} to fit spurious correlations that exist only in the training data, thereby impairing the model's generalization ability and potentially inducing overfitting.

Theorem 1 provide the theoretical results for the global asymptotic stability of the `df`-ESN. However, in real-world problems, a key aspect of evaluating the stability of ESN's running involves quantifying the impact of small disturbances.

Definition 1 (local ESP [39]). The ESN has a local echo state property if for a perturbation $\mathbf{x}'(t_0) = \mathbf{x}(t_0) + \delta$ at time t_0, $\lim_{t\to\infty} \| \mathbf{x}(t) - \mathbf{x}'(t) \| \to 0$ holds true for a sufficiently small δ.

Definition 2 (Weakly nonlinear systems). An n-dimensional nonlinear system $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x})$ is said to be weakly nonlinear in the neighborhood of the equilibrium point $\boldsymbol{x} = \boldsymbol{0}$ if $\boldsymbol{f}(\boldsymbol{x})$ can be expressed as $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{R}(\boldsymbol{x})$, where $\boldsymbol{A}$ is the constant $n \times n$ Jacobian matrix of $\boldsymbol{f}(\boldsymbol{x})$ evaluated at $\boldsymbol{x} = \boldsymbol{0}$, $\boldsymbol{f} = (f_1, f_2, \ldots, f_n)^{\text{T}}$, $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^{\text{T}}$, and the vector function $\boldsymbol{R}(\boldsymbol{x})$ satisfies the higher-order infinitesimal condition: $\lim_{\|x\|\to 0} \|\boldsymbol{R}(\boldsymbol{x})\|/\|\boldsymbol{x}\| = 0$.

The local ESP guarantees that the reservoir state is robust for *infinitesimally small* perturbations of the any initial conditions. It differs from the global ESP requiring that the same attractor remains invariant for arbitrarily large perturbations to the system. In practical applications, local ESP has been proven to be a satisfactory criterion when training and predicting are aimed at the same attractor [39].

When considering the infinitesimally small perturbations and the weakly nonlinear systems, the error dynamics can be rigorously approximated by their linearized counterpart, since higher-order terms become negligible for sufficiently small perturbations. Thus, the local asymptotic stability of the error system can be fully characterized by the linearized system.

Denoting $\widetilde{\mathbf{x}}(t) = \mathbf{W}_{\text{in}}\mathbf{u}_{t+1} + \mathbf{W}_{\text{res}}\mathbf{x}_t$, and using Eq. (4), we have $\mathbf{x}_{t+1} = \gamma \mathbf{x}_{t-m} + f(\widetilde{\mathbf{x}}(t))$. If we denote $\delta t > 0$ as the time step, then

$$(\delta t)^{-1}(\mathbf{x}_{t+1} - \mathbf{x}_t) = (\delta t)^{-1}[\gamma \mathbf{x}_{t-m} + f(\widetilde{\mathbf{x}}_t) - \mathbf{x}_t]. \tag{8}$$

When $\delta t \to 0$, by rescaling the time variable, we can derive

$$\dot{\mathbf{x}}_t = -\mathbf{x}_t + \gamma \mathbf{x}_{t-\tau} + f(\widetilde{\mathbf{x}}_t), \tag{9}$$

where $\tau = m\delta t$ is the time delay. The derivation of Eq. (9) is given in Appendix B.

To analyze the local ESP, let $\mathbf{x}_t$ and $\mathbf{z}_t$ be two arbitrarily close reservoir states. Denoting $\mathbf{e}_t = \mathbf{x}_t - \mathbf{z}_t$, we then have

$$\dot{\mathbf{e}}_t = -\mathbf{e}_t + \gamma \mathbf{e}_{t-\tau} + f(\widetilde{\mathbf{x}}_t) - f(\widetilde{\mathbf{z}}_t). \tag{10}$$

The linearized equation of the error system (10) is then given by

$$\dot{\mathbf{e}}_t = -\mathbf{e}_t + \gamma \mathbf{e}_{t-\tau} + \sigma \mathbf{W}_{\text{res}}\mathbf{e}_t, \tag{11}$$

where $\sigma = f'(s)$ is the derivative of the activation function, and s lies between $\widetilde{\mathbf{x}}(t)$ and $\widetilde{\mathbf{z}}(t)$. We now have the following theorem for the local ESP.

Theorem 2. Consider Eq. (10) with weak nonlinearity satisfying Definition 2. Assume that the activation function $f(.)$ is continuously differentiable with $f(0) = 0$. Denote $\sigma = f'(s)$, where s lies between $\widetilde{\mathbf{x}}(t)$ and $\widetilde{\mathbf{z}}(t)$. If the spectral radius of the reservoir matrix satisfies $\rho < 1$, then *df*-ESN has local ESP if one of the following conditions holds:

(i) $|\gamma| + \rho\sigma < 1$, for any $\tau > 0$.

(ii) $|\gamma| + \rho\sigma > 1$ with $\gamma < 0$, and $\tau \in [0, \tau^*)$, where τ^* is given in Eq. (C7).

The proof of Theorem 2 can be seen in Appendix C.

The value of σ determines the critical threshold for the local ESP. Let $\sigma_{\max} = \sup_s f'(s)$ denote the global maximum of the activation function derivative. Using this upper bound yields the most conservative stability condition, as it assumes the strongest possible amplification from the reservoir nonlinearity, ensuring stability for all admissible values of $f'(s)$ over the entire state space.

Corollary 1. Under the conditions of Theorem 2, if $f(\cdot) = \tanh(\cdot)$, then $\sigma_{\max} = 1$, and thus *df*-ESN satisfies the local ESP if one of the following conditions holds:

(i) $|\gamma| + \rho < 1$, for any $\tau > 0$.

(ii) $|\gamma| + \rho > 1$ with $\gamma < 0$, and $\tau \in [0, \tau^*)$, where τ^* is given in Eq. (C7).

Remark 2. Condition (i) of Corollary 1 is analogous to the global ESP in the standard ESN. In contrast, condition (ii) relaxes the global ESP condition due to $|\gamma| + \rho > 1$ with $\gamma < 0$. This differs greatly from some existing ESN models, which only use positive parameters.

Using the activation function $f(\cdot) = \tanh(\cdot)$, and $\rho = 0.7$, Fig. 4 illustrates the regions of the local ESP for different values of σ. The three colored regions correspond to $\sigma = 1$ (green), $\sigma = 0.5$ (red), and $\sigma = 0$ (blue), respectively, verifying the core conclusion that the larger the σ, the smaller the local stability region of the system. Specifically, $\sigma_{\max} = 1$ corresponds to the narrowest local stability region with the most conservative local stability results.

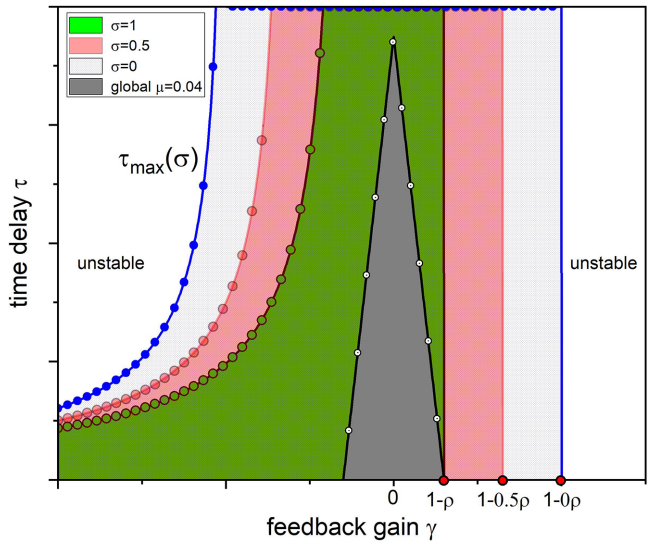


FIG. 4. Local ESP for different values of parameter σ. Clearly, $\sigma_{\max} = 1$ corresponds to the narrowest stability region, which is appropriate for scenarios requiring high stability and strict avoidance of instability risks. The global stability condition in Theorem 1 leads to more conservative outcomes (gray grid areas).

Furthermore, the gray grid area in the figure represents the stability region derived from the global stability condition in Theorem 1 with $\mu = 0.04$. Reducing the value of μ can expand the global stability region; however, even in the limiting case of $\mu \to 0$, the maximum attainable global stability region, defined by $|\gamma| < 1 - \rho$ and $\tau \geqslant 0$, remains significantly smaller than the local stability regions derived from the ESP analysis. This confirms that the global stability condition yields a more conservative outcome than the local ESP-based conditions.

Next, we analyze the noise robustness of the proposed `df`-ESN and quantify how internal perturbations propagate through the delayed feedback loop. We derive the noise-to-state transfer function and its corresponding H_∞ norm in the following theorem.

Theorem 3. Suppose the *df*-ESN with internal noise $\mathbf{Q}_t$ satisfies the delay-dependent stability condition: there exists $\mu > 0$ such that $|\gamma| + \beta + \mu m < 1$, $\beta = L_f \|\mathbf{W}_{\text{res}}\|$. Assume the activation function $f(\cdot)$ satisfies the global Lipschitz condition. Then the noise transfer function from $\mathbf{Q}_t$ to the state deviation $\mathbf{e}_t$ is

$$G(z) = \frac{z^m}{z^{m+1} - \beta z^m - |\gamma|},$$

which is stable. Its H_∞ norm satisfies $\|G\|_\infty < 1/(\mu m) + 1$. In particular, the low-frequency noise gain is bounded by

$$|G(1)| = \frac{1}{1 - \beta - |\gamma|} < \frac{1}{\mu m}.$$

The proof of Theorem 3 is given in Appendix D.

At zero frequency, $|G(1)| = 1/(1 - \beta - |\gamma|) > 1$, so low-frequency noise is amplified. However, by keeping $|\gamma|$ and β sufficiently small, the gain can be limited. Hence the stability condition also serves as a design guideline: a larger m or a more "strict" choice of μ encourages the use of smaller $|\gamma| + \beta$, which is the direct mechanism for suppressing noise amplification.

For comparison, consider the L-ESN model with internal noise $\mathbf{Q}_t$:

$$\mathbf{x}_{t+1} = (1 - \alpha)\mathbf{x}_t + \alpha f(\mathbf{W}_{\text{in}}\mathbf{u}_{t+1} + \mathbf{W}_{\text{res}}\mathbf{x}_t + \mathbf{Q}_t). \tag{12}$$

A straightforward calculation yields

$$G_L(z) = \frac{1}{z - \lambda_L}, \tag{13}$$

where $\lambda_L = 1 - \alpha + \alpha\beta$, and H_∞ norm $\|G_L\|_\infty = 1/(1 - \lambda_L)$ is a finite constant. This confirms the inherent low-pass filtering characteristic that L-ESN suppresses high-frequency noise but exhibits a fixed amplification of low-frequency noise.

Its low-frequency gain is $G_L(1) = 1/(1 - \lambda_L)$, which is directly tied to the memory length: longer memory requires $\lambda_L \to 1$, inevitably causing large noise amplification. In contrast, `df`-ESN decouples memory length (controlled by m) from the noise gain (controlled by $|\gamma|$ and β). Therefore, for the same memory requirement, `df`-ESN can achieve a much lower noise amplification by keeping $|\gamma|$ and β small while setting m appropriately. This makes `df`-ESN strictly superior in noisy environments where both long memory and low noise sensitivity are desired.

B. Memory capacity

When $|f(.)| < 1$, from Eq. (4) we have

$$\|\mathbf{x}_{t+1}\| \leqslant \|\mathbf{W}_{\text{in}}\mathbf{u}_{t+1}\| + \rho\|\mathbf{x}_t\| + |\gamma|\|\mathbf{x}_{t-m}\|.$$

Repeatedly using Eq. (4) on the above equation yields

$$\begin{aligned}\|\mathbf{x}_{t+1}\| &\leqslant \|\mathbf{W}_{\text{in}}\mathbf{u}_{t+1}\| + \rho\|\mathbf{W}_{\text{in}}\mathbf{u}_t\| + |\gamma|\|\mathbf{W}_{\text{in}}\mathbf{u}_{t-m}\| \\ &\quad + \rho|\gamma|\|\mathbf{W}_{\text{in}}\mathbf{u}_{t-m}\| + \cdots \\ &= \sum_{l=0}^{+\infty}\sum_{k=0}^{l} C_l^k \rho^{l-k}|\gamma|^k \|\mathbf{W}_{\text{in}}\mathbf{u}_{t-(km+l)}\|.\end{aligned}$$

Clearly, the state of the reservoir depends on all the input states before current moment. In addition, if $0 < \rho < 1$ and $|\gamma| < 1$, then $\lim_{l\to+\infty}\rho^l = 0$, $\lim_{l\to+\infty}|\gamma|^l = 0$. Thus, there exists a positive integer S such that when $i > S$, $\rho^i \to 0$ and $|\gamma|^i \to 0$. Therefore, it can be obtained that

$$\begin{aligned}\|\mathbf{x}_{t+1}\| &\leqslant \sum_{l=0}^{S}\sum_{k=0}^{l} C_l^k \rho^{l-k}|\gamma|^k \|\mathbf{W}_{\text{in}}\mathbf{u}_{t-(km+l)}\| \\ &\quad + o(\rho^S + |\gamma|^S),\end{aligned}$$

where $o(\cdot)$ is a higher-order infinitesimal. So we have

$$||\mathbf{x}_{t+1}|| \leqslant \rho^S\|W_{\text{in}}\mathbf{u}_{t-S}\| + |\gamma|^S\|\mathbf{W}_{\text{in}}\mathbf{u}_{t-S(m+1)}\| + \cdots \tag{14}$$

From Eq. (14), it is clear that the reservoir state in the standard ESN (i.e., when $\gamma = 0$) at time t remembers the input states of the previous S time steps as shown in the first term on the right-hand side of Eq. (14). However, the reservoir state of the `df`-ESN can remember the input states of the previous $S(m + 1)$ time steps as shown in the second term on the right-hand side of Eq. (14). This clearly indicates the superior memory ability of the `df`-ESN compared with that of ESN as shown Fig. 5.

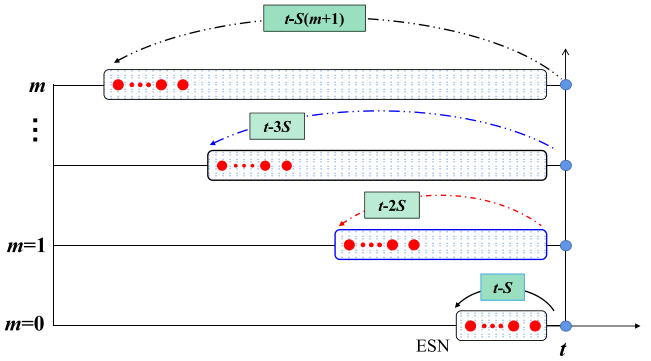


FIG. 5. Comparison of the memory performance of ESN and df-ESN. Clearly, the standard ESN (i.e., when $\gamma = 0$) at time t remembers the input states of the previous S time steps; however, the df-ESN can remember the input states of the previous $S(m+1)$ time steps. This clearly indicates the superior memory ability of the df-ESN.

Next, to examine the ability of the models to extract information from input signals, we consider the memory capacity (MC) proposed by Jaeger in Ref. [40].

Definition 3. Let $\mathbf{u}(t) \in U$ $(-\infty < t < +\infty, U \subset \mathbb{R})$ be a one-dimensional input signal. Let the reservoir matrix and the input matrix be $\mathbf{W}_{\text{res}}$ and $\mathbf{W}_{\text{in}}$, respectively. For the delay step k, let the output weight be $\mathbf{W}_{\text{out}}^{k}$ and the output be $\mathbf{y}_k$. The overall memory capacity (MC) of the ESN model and the k-step delayed short-term memory capacity (MC_k) are as follows:

$$\begin{aligned} \text{MC} &= \sum_{k=1}^{\infty} \text{MC}_k, \\ \text{MC}_k &= \max_{W_{\text{out}}^{k}} d\left[\mathbf{W}_{\text{out}}^{k}\right](\mathbf{u}(t-k), \mathbf{y}_k(t)), \\ d\left[\mathbf{W}_{\text{out}}^{k}\right] &= \frac{\text{cov}^2(\mathbf{u}(n-k), \mathbf{y}_k(n))}{\sigma^2(\mathbf{u}(n))\sigma^2(\mathbf{y}_k(n))}, \end{aligned} \tag{15}$$

where cov represents the covariance, σ^2 represents the variance, $\mathbf{u}(n-k)$ is the input data, $\mathbf{y}_k(n)$ is the output data, and k is the delay time step.

To compute the MC numerically, we randomly generate 5000 input data located in the interval $[-0.5, 0.5]$, where the input sequence is random, with no correlation between the inputs at different times. The first 500 data are used for warm-up, the next 4000 are used for training, and the remaining data are used for computing MC. Figure 6(a) shows a comparison of the MC_k curves between ESN and df-ESN when the spectral radius $\rho = 0.95, 0.5$ and the reservoir size $N = 100$.

In df-ESN, we set the parameters as $\gamma = -0.1$ and $m = 1$. Obviously, the MC_k of ESN exhibits almost 100% recall ability ($\text{MC}_k > 0.95$) for input delays up to 8, followed by steep slopes descending in a clifflike manner, as shown by the black square curve. When the input delay is greater than 15, the MC_k is 0.003 very close to 0. This clearly indicates the short-term memory capacity of the ESN. Unlike ESN, df-ESN exhibits a better MC_k even if for a larger input delay K. Ultimately, the memory capacity stabilizes around $\text{MC}_\infty \approx 0.20$ when

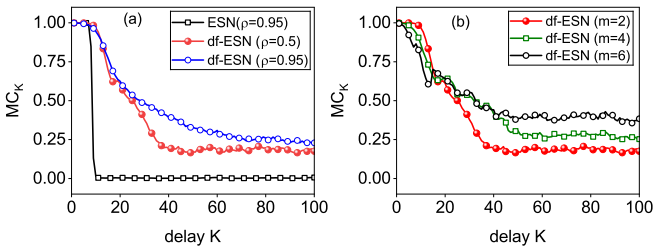


FIG. 6. Comparison of the memory capacity between the ESN and df-ESN, where (a) $\rho = 0.95, 0.5$; (b) for different delay steps m. Clearly, the proposed df-ESN can produce good MC even for a relatively small spectral radius. In addition, the selected m can affect the memory capacity.

$\rho = 0.5$, and $\text{MC}_\infty \approx 0.24$ when $\rho = 0.95$, as shown by the red solid circle curve and the blue hollow curve, which is much greater than $\text{MC}_\infty \approx 0.003$ in ESN as shown by the black square curve. The aim of choosing $\rho = 0.5$ is to show that the proposed df-ESN can produce good MC even for a relatively small spectral radius. This is a strong distinction from the standard ESN model, which requires the spectral radius of W_{res} to be very to close 1.

Figure 6(b) presents the memory capacity curves of df-ESN under different delay step m. Clearly, for the different delays m, df-ESN achieves the MC_k with a certain amplitude: $\text{MC}_k \approx 0.20$ for $m = 2$, $\text{MC}_k \approx 0.28$ for $m = 4$, and $\text{MC}_k \approx 0.38$ for $m = 6$. So, in tasks that require capturing long-term dependencies on historical information, the df-ESN with appropriate delay parameters is a good model to improve the memory ability.

C. Two-stage optimization strategy

In echo state networks, the performance is highly sensitive to multiple interdependent hyperparameters, such as spectral radius, input scaling, reservoir size, and sparsity. However, there exists no universal theoretical guideline for their optimal selection, often requiring extensive trial-and-error tuning.

The df-ESN fundamentally redefines hyperparameter selection, replacing black-box tuning with the efficient two-stage grid search framework. In practice, we first perform a rough preliminary estimation of the core reservoir hyperparameters. We then construct a uniform grid on the $m - \gamma$ plane by discretizing m and γ as

$$m_i = m_{\min} + i\,\frac{m_{\max} - m_{\min}}{M},\ \gamma_j = \gamma_{\min} + j\,\frac{\gamma_{\max} - \gamma_{\min}}{M_\gamma},$$

where $i = 0, 1, \ldots, M$, $j = 0, 1, \ldots, M_\gamma$.

This generates a set of grid points (m_i, γ_j). We compute the short-term RMSE at each grid point, select the pair (m, γ) with the lowest RMSE as the optimal combination, and then use these parameters to predict the long-term system evolution.

The two-stage optimization strategy eliminates the need for exhaustive fine-tuning of the core reservoir parameters, requiring only coarse initial estimates. This design circumvents the high-dimensional, costly optimization typically imposed by the full reservoir parameter space. Instead, df-ESN performs a lightweight grid search over just two critical hyperparameters,

TABLE I. Complexity comparison between standard ESN and df-ESN.

Item	Standard ESN	df-ESN	Additional cost
Memory (training)	$O(TN + N^2)$	$O(TN + N^2 + mN)$	$O(mN)$
Memory (prediction)	$O(N^2)$	$O(N^2 + mN)$	$O(mN)$
Training – forward (per step)	$O(\eta N^2 + Nd_{\text{in}})$	$O(\eta N^2 + Nd_{\text{in}} + N)$	$O(N)$
Training – solving $\mathbf{W}_{\text{out}}$	$O(TN^2 + N^3)$	same	0
Prediction (per step)	$O(\eta N^2 + Nd_{\text{out}})$	$O(\eta N^2 + Nd_{\text{out}} + N)$	$O(N)$

Notes: N = reservoir size, T = training steps, η = sparsity (nonzero fraction), m = delay steps, d_{in}, d_{out} = input/output dimensions. The added $O(N)$ per step and $O(mN)$ memory are lower-order terms for all practical settings.

m and γ, effectively compressing the search space to a simple two-dimensional domain.

D. Computational and memory complexity

We analyze the additional computational and memory cost introduced by $\gamma\mathbf{x}(t-m)$, compared with the standard ESN. The main results are summarized in Table I.

Memory footprint. The df-ESN uses *only a single* delayed state $\mathbf{x}(t-m)$ at each time step, not the *entire* history of m steps. To retrieve $\mathbf{x}(t-m)$ efficiently, we maintain a circular buffer of length $m+1$ that stores the most recent reservoir states. Therefore, the extra memory required is exactly $O(mN)$. For typical parameter settings (such as $N \leqslant 2000$, $m \leqslant 10$), this adds is negligible compared to the $O(N^2)$ storage of $\mathbf{W}_{\text{res}}$ or the $O(TN)$ storage of the state collection matrix during training.

Computational overhead. At each time step, the extra operation is the vector addition $\gamma\mathbf{x}(t-m)$, which costs $O(N)$ floating-point operations. This per-step overhead is asymptotically dominated by $\mathbf{W}_{\text{res}}\mathbf{x}_t$, which requires $O(\eta N^2)$ operations, where η is the sparsity of $\mathbf{W}_{\text{res}}$ (typically $\eta = 0.01$ to 0.1). Hence, on a per-step basis, the delayed feedback does *not* change the overall complexity class of the model. The total training complexity for both standard ESN and df-ESN is $O(T\eta N^2 + N^3)$, where the $O(N^3)$ term comes from solving the linear system for the output weights $\mathbf{W}_{\text{out}}$ via ridge regression. The prediction complexity per step is $O(\eta N^2)$. These dominating terms are identical for the two models.

Scaling to large N and m. If the required delay is very large ($m \gg 10$), the linear overhead $O(mN)$ might become noticeable. In such cases, one can employ either *sparse delay* (store only every kth state and interpolate) or *hierarchical delay* (cascade multiple short-delay reservoirs). For extremely high-dimensional inputs (e.g., millions of degrees of freedom), a standard practice is to first reduce the input dimension using PCA, random projection, or an autoencoder; the complexity analysis with respect to N and m remains unchanged.

Overall, the delayed feedback in df-ESN introduces only linear overheads that are negligible for the reservoir sizes and delay lengths typically used in practice; therefore, df-ESN preserves the scalability of standard ESN and is suitable for large-scale complex systems.

IV. NUMERICAL EXPERIMENTS

This section conducts some numerical experiments to assess the effectiveness of the proposed df-ESN model. To ensure a fair and consistent comparison across all baseline models, the unified hyperparameter tuning protocol is adopted in this work. For all compared models, including the standard ESN, leaky ESN (L-ESN), deep ESN, and the proposed df-ESN, the same reservoir size, spectral radius search range, training sequence length, and input scaling factor are used during both hyperparameter optimization and final performance evaluation.

In each experiment, the elements in matrix W_{res} satisfy the random uniform distribution with values between –1 and 1. Both the input $\mathrm{u}(t)$ and desired output $\mathrm{y}_*(t)$ are normalized using the mean 0 and standard deviation 1. For the df-ESN model, we optimize the parameters γ and m via *grid-searching technique* as shown in Sec. III C. In contrast, all hyperparameters of the baseline models are optimized via Grid Search or Heuristic Algorithms to identify the best-performing configurations.

The following root mean square error (RMSE) is used to measure the performance of the models,

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}[y_*(t) - y(t)]^2},$$

where $y_*(t)$, $y(t)$, and T, respectively, represent the desired values, predicted values, and the prediction time steps.

A. Three-dimensional chaotic system

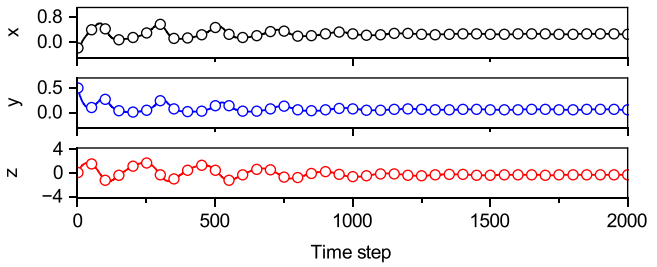
We consider a three-dimensional dynamical system, which is described by [41]

$$\begin{aligned} \dot{x} &= yz + r, \\ \dot{y} &= x^2 - y, \\ \dot{z} &= 1 - 4x, \end{aligned} \tag{16}$$

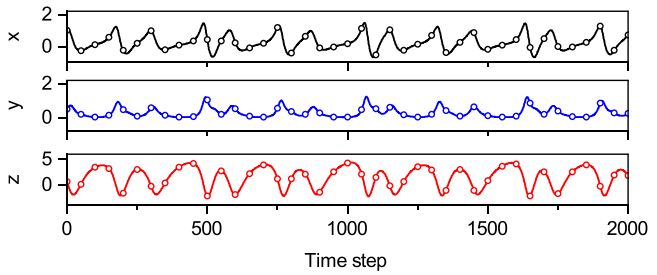
where r is a parameter.

1. Prediction on the basin of attraction

The numerical results indicate that when $r = 0.2$, system (16) exhibits multistability states under different initial values (x_0, y_0, z_0). When the initial condition is (0.2, 0.5, 0.6), the system exhibits amplitude death, as illustrated in Fig. 7(a) [42,43]; in contrast, when the initial condition is (1.0, 0.5, 1.5), the system remains in an oscillatory state, as shown in Fig. 7(b). These two phenomena are widely used in physical, chemical, biological, and physiological systems, and have been extensively studied recently [44–46].



(a) Amplitude death at initial value (0.2,0.5,0.6).



(b) Oscillation at initial value (1.0,0.5,1.5)

FIG. 7. Multistability of system (16): (a) amplitude death, (b) oscillation.

In this experiment, we consider the basin of attraction in the $x - z$ plane($[x_0, z_0] \in [-1, 1.5] \times [-2, 2]$) under the fixed value of $y_0 = 0.5$. The basin of attraction refers to all initial states in phase space whose trajectories converge to a given attractor over time. The true basin of attraction is shown in Fig. 8. The central red area represents initial states inducing amplitude death, and the surrounding white region denotes those sustaining system oscillation.

The initial-value-aware scheme [32], which details are in Appendix E, is used for prediction of basins of attraction. We use the data under five different sets of initial values: (x_0, z_0)= (0.7,0.0), (0.3, 0.0), (–0.3, 0.0), (0.0, 1.2), and (0, –0.6) as the training data. There are 5000 training data points for each set of initial values, resulting in a total of 25 000 training

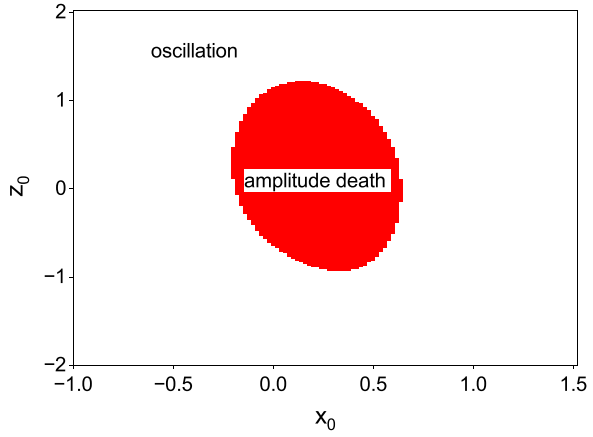


FIG. 8. Basin of the attraction of Eq. (16). The central red area represents initial states inducing amplitude death, and the surrounding white region denotes those sustaining system oscillation.

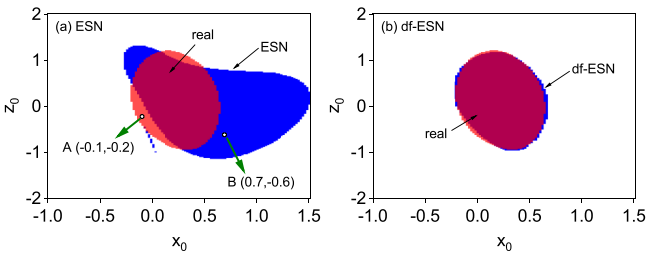
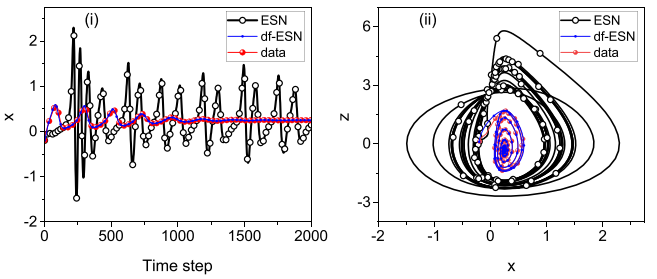


FIG. 9. Basin of attraction prediction: (a) ESN and (b) df-ESN. Blue regions denote the basin predicted by the models, while red regions represent the true basin of amplitude death in the system. The df-ESN prediction aligns perfectly with the real basin.

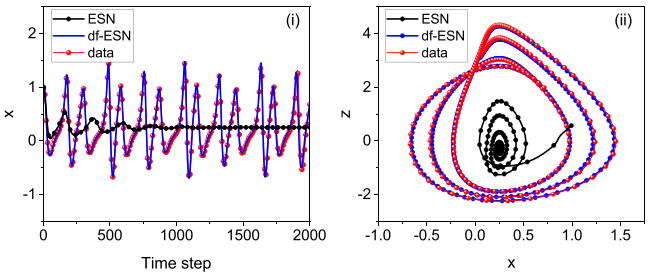
data points. Both input and output are three-dimensional data $(x(k), y(k), z(k))$ at time step k, where $k = 1, 2, \ldots, 25\,000$.

Figure 9 shows the basin of attraction predicted by the ESN and df-ESN. The parameters used in the df-ESN are reservoir size $N = 1000$, regularization parameter $\xi = 10^{-8}$, radius $\rho = 0.5$, sparsity $\eta = 0.1$, feedback gain $\gamma = -0.2$, and delay step $m = 3$. The parameters in ESN model are: $N = 1000$, $\rho = 0.9$, $\eta = 0.2$, and $\xi = 10^{-3}$. Compared with the real attraction basin in Fig. 8, the result predicted by df-ESN achieves high consistency. Moreover, the prediction accuracy of df-ESN in Fig. 9(b) is obviously superior to that in Fig. 9(a).

As shown in Fig. 9(a), initial state A lies outside the blue region, implying the standard ESN predicts an oscillatory response. Nevertheless, A belongs to the actual red attraction basin, where the real system exhibits amplitude death. This reveals obvious prediction deviation, which is further verified by the time-domain and phase-plane results in Fig. 10(a). The



(a) Predicted solutions for initial value A (-0.1,-0.2).



(b) Predicted solutions for initial value B (0.7,-0.6).

FIG. 10. Time history and phase diagram of the system when initial values are $A(-0.1, -0.2)$ and $B(0.7, -0.6)$. Clearly, the df-ESN can give the predicted solutions for different initial values which align perfectly with the real solution, while ESN cannot.

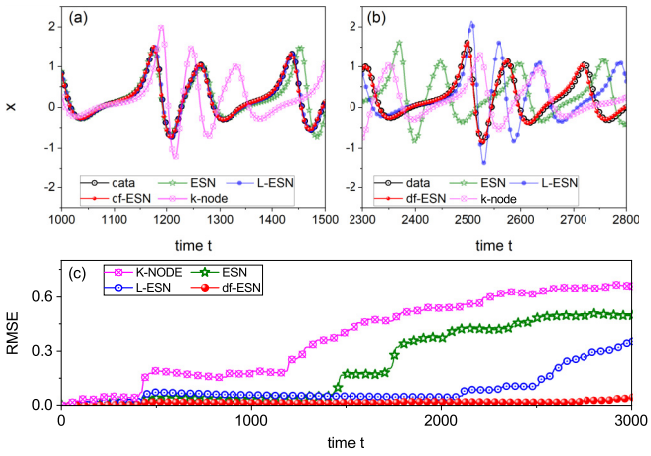


FIG. 11. Prediction performance of the different models. (a) and (b) x-direction trajectories at 1000–1500 and 2300–3000 steps; (c) RMSE evolution. Clearly, the df-ESN exhibit the same long-term dynamical behavior as actual chaotic systems.

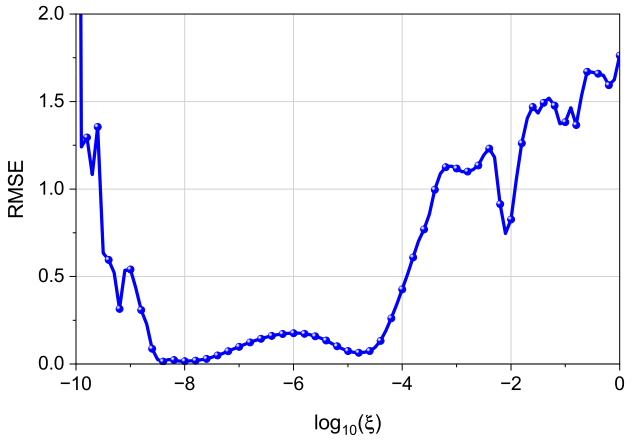


FIG. 12. RMSE curves of chaotic time series under different Tikhonov regularization parameters $\boldsymbol{\xi}$. Weak regularization causes singularity and overfitting, while excessively strong regularization restrains temporal dynamics capture and modeling performance.

same mismatch occurs at initial state B, where ESN yields amplitude death while the real system maintains oscillation [see Fig. 10(b)]. In contrast, the proposed df-ESN produces predictions highly consistent with actual responses under various initial conditions, and can accurately characterize the inherent dynamics of the system.

The favorable prediction performance of df-ESN on the basin of attraction mainly benefits from its delay-feedback mechanism allowing the network to effectively capture long-range correlation and nonstationary features. The delay mechanism also assists the reservoir in retaining historical dynamic information and maintaining stable evolution trajectories, thus better distinguishing diverse attraction domains in nonlinear systems.

2. *Chaotic time-series prediction*

When $r = 0$, system (16) exhibits chaotic solutions with initial value $(x_0, y_0, z_0) = (1, 1, 1)$. Using the Runge-Kutta method [27], we generate the time series data containing 5500 sampling points, where the first 500 samples are used for warm up, the following 2000 data are as the input $\mathrm{u}(t)$ for training the reservoir, the remaining data are as the desired output $\mathrm{y}_*(t)$. Both input and output are three-dimensional data (x_k, y_k, z_k), $k = 1, 2, \ldots, 5500$. The df-ESN is configured with $N = 500$, $\xi = 10^{-8}$, $\rho = 0.5$, $\eta = 0.1$, $\gamma = -0.3$, and $m = 3$.

Figure 11 presents the prediction results from ESN, L-ESN, df-ESN, and the physics-informed K-NODE method [47]. For the ESN, the reservoir size is set to $N = 500$, spectral radius $\rho = 0.95$, sparsity $\eta = 0.06$, and regularization coefficient $\xi = 10^{-6}$. For the L-ESN model, the configuration is $N = 500$, $\rho = 1.0$, $\eta = 0.01$, leakage factor $\alpha = 0.4$, and $\xi = 10^{-7}$. In the K-NODE, Eq. (16) is incorporated as physical prior knowledge, with a lookahead coefficient of $\alpha = 3$, a learning rate of $lr = 0.01$, and a total of 1000 training epochs.

To intuitively visualize the performance discrepancies among these methods, Fig. 11 characterizes the prediction results in the x-direction. Specifically, subplot (a) displays the predicted trajectories over the time step range of 1000–1500, subplot (b) shows the prediction outcomes for time steps 2300–2800, and subplot (c) depicts the RMSE of predictions across all time steps from 0 to 3000. The prediction results in the y and z directions exhibit consistent trends with those in the x-direction, and thus are omitted for brevity.

Clearly, the ESN achieves satisfactory accuracy with RMSE values lower than 0.1 within the first 1500 time steps, but diverges from the actual data as time evolves. The L-ESN delivers more accurate predictions than the ESN within 2500 time steps; however, it still produces considerably larger errors than the proposed df-ESN, which maintains an RMSE below 10^{-2} throughout the entire 3000-time-step prediction period. As a typical neural-ODE-based model, K-NODE inherently amplifies tiny initial errors during each numerical integration step, which degrades its prediction performance for chaotic systems. By contrast, benefiting from delayed feedback mechanism, df-ESN achieves stable long-term prediction performance.

Next, we investigated the impact of the regularization parameter ξ on the system performance, where ξ varies from 0 to 10^{-10}, while all other parameters are kept fixed. The results as shown in Fig. 12 indicate that the df-ESN can achieve a good balance between fitting and generalization performance when $\log_{10}(\xi)$ is approximately in the range of -8.5 to -4.5.

From a spectral analysis perspective, ξ regulates the eigenvalues of the covariance matrix $\mathbf{X}\mathbf{X}^\top$ by shifting them to $\lambda_i + \xi$. If ξ is too small, it fails to elevate the near-zero eigenvalues, leading to singular or ill-conditioned matrix inversion and causing model overfitting. Conversely, if ξ is too large, it will obscure the informative eigenvalues, excessively constrain the dynamic characteristics of the reservoir, and fail to fully utilize its expressive capacity. Therefore, a moderate $\boldsymbol{\xi}$ is required to improve matrix inversion stability while preserving informative eigenvalues for reservoir dynamic features.

3. *Two-stage optimization strategy*

We numerically verified the efficiency of the two-stage grid search method using system (16) for 500-step predictions, and obtained the following results.

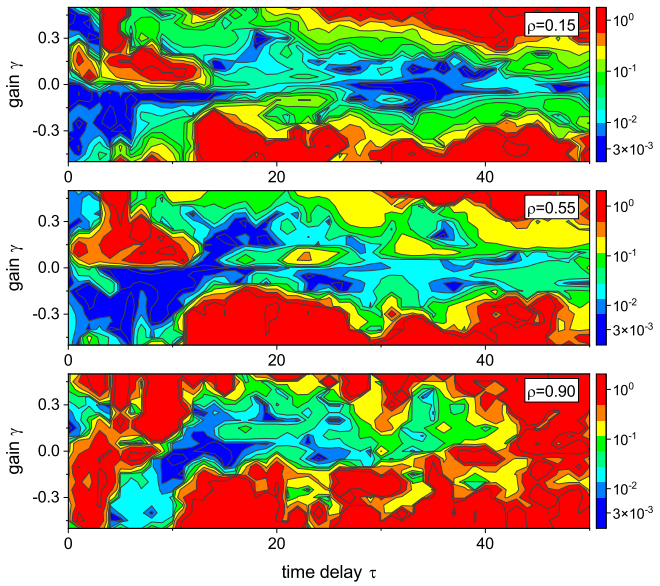


FIG. 13. Colormaps of the predicted RMSE in the m-γ plane for different spectral radii ρ. Darker (bluer) colors represent smaller prediction errors. For each ρ, we can obtain a proper parameter pair (m, γ) to yield low prediction error.

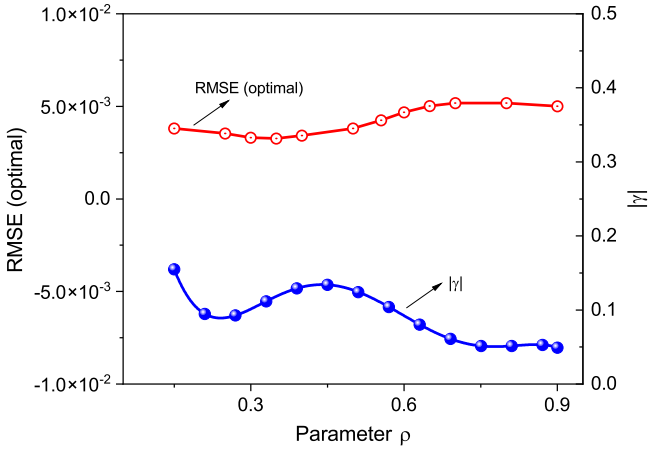


FIG. 14. The optimal prediction RMSE and corresponding feedback gain γ as functions of the spectral radius ρ. The results demonstrate that `df`-ESN maintains strong robustness against reservoir parameter variations, and that the coupling between ρ and (m, γ) is weak.

(1) *Wide low-error region in* (m, γ) *plane.* As shown in Fig. 13, the `df`-ESN maintains low prediction error over a broad range of (m, γ) values (blue regions) for different ρ (0.15, 0.55, 0.90). Importantly, near-optimal performance is achievable across all tested ρ, confirming that suitable (m, γ) pairs exist for different reservoir configurations. This indicates low sensitivity of the `df`-ESN to the exact choice of (m, γ); a coarse 2D grid search is sufficient to locate good parameters, and fine-tuning is not critical.

Furthermore, as the time delay m increases, the high-accuracy region of the feedback gain gradually narrows. This trend is consistent with the prediction in Theorem 1, which implies diminishing returns with larger delays. For large delays, the model automatically reduces the gain to mitigate overfitting, balancing complexity and generalization.

(2) *Weak coupling between* ρ *and* (m, γ). Figure 14 shows how the optimal RMSE varies with the spectral radius ρ. For each ρ in the tested range, we can obtain a proper parameter pair (m, γ) to yield low prediction error. The optimal RMSE changes slightly within the range from 4.1×10^{-3} to 5.0×10^{-3}.

To quantitatively evaluate the stability of prediction performance, we utilize the mean relative deviation and the coefficient of variation (CV), defined as $\mathrm{CV} = \sigma/\mu \times 100\%$, where μ and σ denote the mean and standard deviation of the RMSE dataset, respectively. The optimal RMSE deviates from the mean by at most $+13.30\%$ and -12.32%, and the corresponding CV is 10.25%. Meanwhile, the optimal feedback gain γ is confined to a narrow interval of 0.05 to 0.15.

These observations reveal that the prediction error fluctuates moderately as ρ changes, implying a weak correlation between the spectral radius ρ and the optimized parameters (m, γ). Accordingly, the sequential tuning strategy—fixing ρ first and then optimizing (m, γ)—is reliable and will not miss near-optimal parameter combinations.

4. Parameter tuning for many hyperparameters

If the hyperparameters of both the reservoir and the delayed feedback are optimized simultaneously, the dimensionality of the search space will be substantially increased, rendering the grid search method impractical. Heuristic algorithms (like Bayesian optimization [48] and PSO [49]) work well for reducing search space and improving efficiency in high-dimensional problems. We compare Grid Search and Bayesian Optimization for tuning the four-dimensional hyperparameters (ρ, η, m, γ) in `df`-ESN, with the results shown in Table II, where the computational efficiency is defined by $(\mathrm{Time} \times \mathrm{RMSE})^{-1} \times 100\%$. Clearly, the Grid Search achieves an RMSE of the same order of magnitude as Bayesian optimization. However, its computational time is considerably higher resulting in a computational efficiency of only 0.49, which is far lower than the 6.26 achieved by the heuristic algorithm.

However, in low-dimensional spaces, the performance of the Heuristic algorithms is far inferior to that of conventional algorithms. Table III give a comparison of the computational efficiency using Bayesian Optimization and Grid Search for two key hyperparameters. These two hyperparameters include: ρ and ξ for ESN, ρ and α for L-ESN, m and γ for `df`-ESN; and other parameters are fixed. Clearly, although Bayesian optimization yields lower RMSE, it also requires substantially more computational time. For the `df`-ESN, Bayesian Optimization achieves the lowest error (0.039), but Grid Search $(m = 3, \gamma = -0.3)$ attains 80% accuracy (RMSE $= 0.067$) with 12.5% computation time and 4.64 times higher efficiency. Consequently, across these models, grid search achieves significantly higher computational efficiency than Bayesian optimization in two-dimensional searching space.

Next, to verify the stability of ESN models under different parameter configurations, we conduct a comparative

TABLE II. Performance comparison of Grid Search and Bayesian Optimization in 4D parameter space.

Method	Hyperparameter	Time	RMSE	Efficiency
Grid search	$\rho = 0.5, \eta = 0.1,$ $m = 3.0, \gamma = -0.3.$	3040.24	0.0675	0.49
Bayesian opt.	$\rho = 0.6583, \eta = 0.1489,$ $m = 1.0, \gamma = -0.3745.$	529.62	0.03017	6.26

analysis of the ESN, L-ESN, and df-ESN under a unified hyperparameter tuning protocol. First, two sets of reservoir parameters for the standard ESN are obtained through heuristic algorithm optimization. Subsequently, these core hyperparameters of the reservoir are fixed for all models. On this basis, a grid search method is further adopted to fine-tune the leakage rate α of L-ESN, as well as the delay step m and feedback gain γ of df-ESN. Detailed parameter settings and RMSE of 2000-step predictions for each method are presented in Table IV and Fig. 15.

Compared with their respective separately optimized results, all models exhibit a slight decrease in prediction accuracy. Nevertheless, the performance degradation of df-ESN is significantly lower than that of both L-ESN and the ESN, as evidenced by its RMSE values of 0.027 (or 0.033), which are far smaller than those of ESN and L-ESN. It is noted that L-ESN demonstrates a higher degradation because its leakage rate α has a stronger coupling relationship with reservoir parameters; consequently, the shared parameters adopted in the unified tuning protocol fail to match its optimal parameter combination. In contrast, the delayed feedback mechanism inherent in df-ESN alleviates such parameter coupling, thereby endowing it with better adaptability to the unified hyperparameter tuning framework.

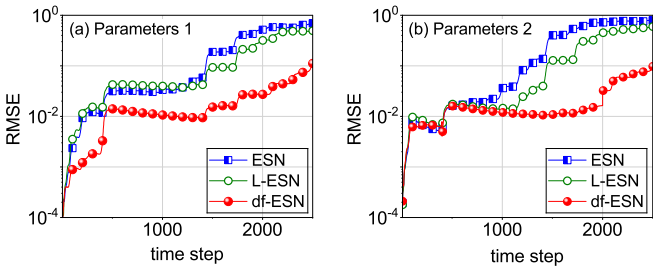


FIG. 15. Prediction RMSE of different models using two distinct sets of reservoir parameters. The df-ESN achieves the slowest error rise with minimal errors, demonstrating that its delayed feedback mechanism significantly improves model stability and adaptability.

Although two-stage grid search is employed in this work, we acknowledge its computational limitations for large-scale systems. To address this, several strategies can be adopted to reduce the search space. (i) Heuristic initialization based on the characteristic time scales of the target system can effectively narrow the search range for the delay parameters. (ii) An adaptive grid search approach, which first uses a coarse grid to locate promising regions and then refines the search within those regions, can reduce the number of evaluations. (iii) Bayesian optimization serves as a powerful alternative, as it intelligently samples the parameter space to find optimal configurations with significantly fewer trials. These methods make the df-ESN framework more computationally feasible for large-scale and high-dimensional applications.

B. Kuramoto-Sivashinsky system

The Kuramoto-Sivashinsky system (KS system) is a high-order partial differential equation that describes the evolution of nonlinear waves in spatially extended systems, and it exhibits spatiotemporal chaos behavior making it extremely be difficult to predict the spatiotemporal behavior [50–52]. In this experiment, we consider the one-dimensional KS system [30,53] as follows:

$$\frac{\partial u}{\partial t} = -\frac{\partial^4 u}{\partial x^4} - \frac{\partial^2 u}{\partial x^2} - u\frac{\partial u}{\partial x}, \tag{17}$$

with boundary and initial conditions:

$$\begin{aligned} &u(0,t) = u(L,t) = \left.\frac{\partial u}{\partial x}\right|_{x=0} = \left.\frac{\partial u}{\partial x}\right|_{x=L} = 0, \\ &u(x,0) = u_0(x), \end{aligned} \tag{18}$$

where $u(x,t)$ is a function of the spatial variable $x \in [0, L]$ and the time variable $t > t_0 = 0$. All variables are nondimensionalized with the characteristic time scale set to unity. The initial condition $u_0(x)$ is typically a small random perturbation or a superposition of multiple sinusoidal modes. The specific form $u_0(x)$ does not affect the long-term chaotic behavior of the KS equation.

TABLE III. Computational efficiency in two-dimensional space for various searching strategies.

Model	Method	Hyperparameter	Time	RMSE	Efficiency
ESN	Grid Search	$\rho = 0.95, \xi = 10^{-6}$	34.52	0.72	4.02
	Bayesian Opt.	$\rho = 0.48, \xi = 10^{-6}$	599.41	0.30	0.56
L-ESN	Grid Search	$\rho = 1, a = 0.4$	32.65	0.55	5.57
	Bayesian Opt.	$\rho = 0.91, a = 0.59$	174.66	0.25	2.29
df-ESN	Grid Search	$m = 3, \gamma = -0.3$	43.83	0.067	34.05
	Bayesian Opt.	$m = 2, \gamma = -0.16$	349.56	0.039	7.34

TABLE IV. Parameters setting and RMSE for 2000-step predictions under two sets of parameters.

Models	ESN	L-ESN	df-ESN
Parameter_1		$N = 450,\ \rho = 0.95$ $\xi = 10^{-6},\ \eta = 0.06$ $\alpha = 0.9$	$(m, \gamma) = (2, 0.05)$
RMSE_1	0.618	0.321	0.027
Parameter_2		$N = 500,\ \rho = 0.78$ $\xi = 10^{-5},\ \eta = 0.10$ $\alpha = 0.94$	$(m, \gamma) = (2, -0.3)$
RMSE_2	0.715	0.433	0.033

1. Spatiotemporal dynamics

In the numerical simulations, the domain length is set to $L = 22$, and the system is spatially discretized with a step size $\Delta x = 0.1$, yielding a uniform spatial discretization of 221 grid points over the interval $[0, L]$. The time domain is discretized as $t_i = t_0 + i\Delta t$ with a time step $\Delta t = 0.25$. At each discrete time step t_i, the system input is given by the state variable u evaluated at these 221 spatial points, so that the input corresponds to 221-dimensional time-series data. In this study, the numerical solution of Eq. (17) is obtained using the comsol software package [54]. The KS system exhibits distinct convective stripe patterns as shown in Fig. 16.

We adopt the 17 500 sampling data to train ESN models, in which the first 2500 samples for warm-up, the following 15 000 data for training, and the remaining data for testing. We perform comparative prediction experiments on ESN, L-ESN and deep-ESN, AERC [33], a neural operator MNO [55] and df-ESN for the KS system. For the deep-ESN, we employed a two-layer reservoir structure and the hidden state is updated similarly to L-ESN. The parameters for all models are $N = 1000$, $\xi = 10^{-4}$, $\eta = 0.1$. The other parameters are $\rho = 0.90$, $\alpha = 0.91$ for deep-ESN; $\rho = 0.85$ for ESN; $\rho = 0.92$, $\alpha = 0.90$ for L-ESN; $\rho = 0.5$, $\gamma = -0.1$, $m = 3$ for df-ESN. The AERC is configured with $\rho = 0.85$, $\alpha = 0.92$ and an attention hidden dimension of 64. We train MNO with parameters: 4 Fourier layers, 32 Fourier modes, channel width 32, learning rate 0.001 (halved every 10 epochs), and 500 epochs.

Figure 17 illustrates the evolution pattern of the KS system in the t-x plane, where the horizontal axis represents the evolution time step, the vertical axis is the spatial position, and the colors represent the predicted value of $u(t)$. The dashed rectangles highlight the deviations between the predicted solutions obtained by the data-driven models and the ground truth. Figure 18 illustrates the corresponding errors with the greener area indicating the smaller error. Apparently, the df-ESN predicted solutions accurately track the real system evolution up to 800 time steps, which is substantially longer than that of baseline models: ESN (around 200 steps), AERC (approximately 320 steps), deep-ESN (roughly 400 steps), L-ESN (about 480 steps), and MNO (nearly 600 steps).

Figure 19 compares the RMSE evolution between the df-ESN with some state-of-art approaches, namely PINN [56], Neural ODE [57], MNO [55], and AERC [33]. For Neural ODE and PINN, three-layer and four-layer fully connected networks with a hidden size of 128 are adopted, respectively, with a consistent learning rate of 10^{-3}. For all compared models, the same reservoir size, spectral radius search range, training sequence length, and input scaling factor are used during both hyperparameter optimization and final performance evaluation via grid search, heuristic algorithms and manual tuning.

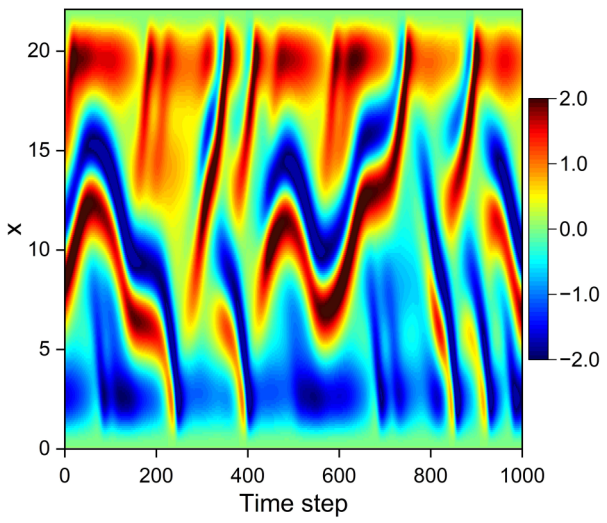


FIG. 16. The pattern of the KS system in the t-x plane.

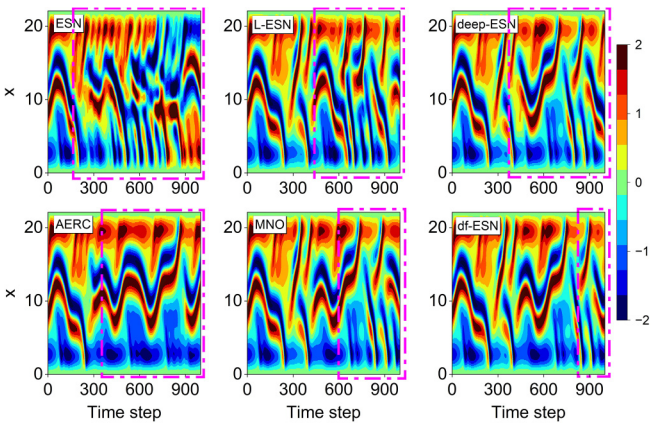


FIG. 17. Predicted spatiotemporal pattern of KS system in the t-x plane. The dashed rectangles highlight the deviations of the solutions between the data-driven models with the ground truth.

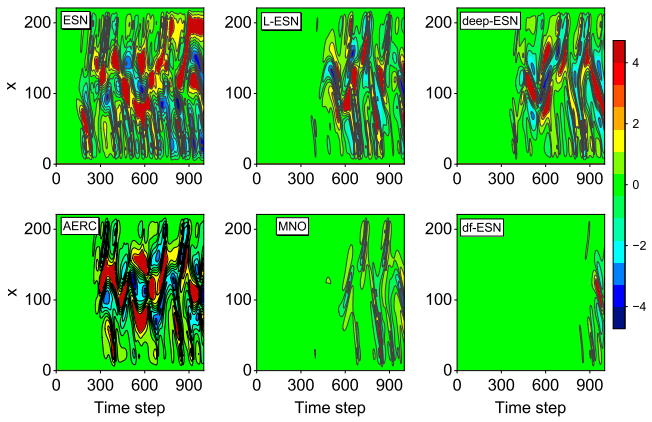


FIG. 18. Pattern of the predicted errors of the KS system between the data-driven models with the ground truth.

While exhibiting slightly higher RMSE in the initial short-term phase, the df-ESN model delivers exceptional long-term robustness and sustained high precision. The RMSE stabilizes around 10^{-2} throughout the entire prediction window of 800 time steps. In contrast, all competing methods show clear limitations: (i) the deep-ESN and AERC models demonstrate outstanding short-term precision, achieving RMSE values are much lower than 10^{-2} within the first 200 time steps. However, their performance degrades rapidly as the prediction horizon extends. (ii) PINN exhibits the poorest overall performance, with RMSE rising rapidly from the early stage and achieving 10^{0} after about 500 steps. Its multilayer feed-forward structure lacks inherent temporal memory, making it unsuitable for modeling the sensitive dynamics of chaotic systems. (iii) Neural ODE suffers from early error divergence, with RMSE stabilizing at a high level after 600 steps. Its limited nonlinear expressivity fails to capture the complex chaotic behavior in the system, resulting in poor long-term stability. (iv) The MNO outperforms PINN and Neural ODE but shows a steady error increase throughout the prediction

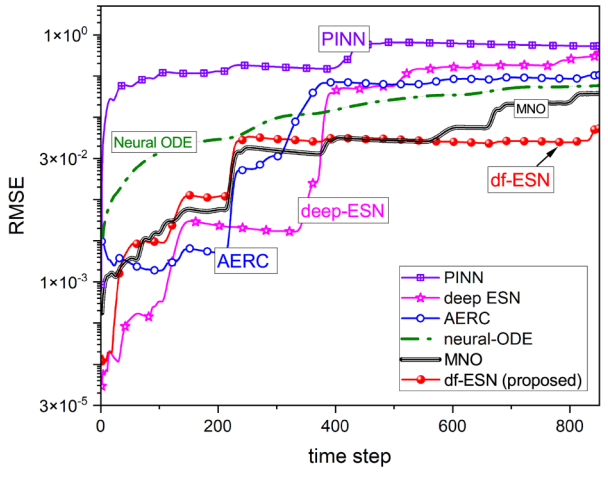


FIG. 19. Prediction performance of various machine learning models for the KS system. Clearly, the df-ESN achieves the lowest and most stable RMSE, significantly outperforming some state-of-art approaches.

TABLE V. Parameter configuration and RMSE of the KS system under different noise intensities.

Intensity	Methods	ρ	η	α	m	γ	RMSE
Noise_0.01	ESN	0.6	0.15				0.848
	L-ESN	0.8	0.03	0.2			0.106
	deep-ESN	0.5	0.1	0.95			0.126
	df-ESN	0.5	0.1		4	0.6	0.043
Noise_0.05	ESN	0.9	0.01				0.850
	L-ESN	0.5	0.02	0.1			0.164
	deep-ESN	0.95	0.1	0.9			0.308
	df-ESN	0.5	0.1		5	0.7	0.051
Noise_0.1	ESN	0.8	0.15				0.864
	L-ESN	0.9	0.03	0.15			0.462
	deep-ESN	1	0.1	0.95			0.360
	df-ESN	0.5	0.1		2	0.7	0.094

horizon. This is because it lacks dynamic memory modeling capability, resulting in severe memory fading for time-series evolution. Therefore, compared with thsee baseline models, df-ESN maintains superior long-term stability and high precision for spatiotemporal chaotic systems by introducing the delay feedback mechanism.

2. *Robust to noise*

To quantitatively evaluate the noise robustness of the df-ESN against the baseline methods, we inject zero-mean Gaussian noise into the normalized data. Three noise levels are adopted with standard deviations of 0.01, 0.05 and 0.1, representing low, medium and high noise intensities, respectively. We employ the ESN, L-ESN, deep-ESN, and df-ESN to compute the errors under a consistent prediction horizon of 400 time steps. For all experimental, we set $N = 1000$ and $\xi = 10^{-4}$. The remaining parameter settings and corresponding results are summarized in Table V.

Clearly, the df-ESN achieves the best performance across all scenarios: under low noise, the RMSE (0.043) is significantly lower than that of ESN (0.848), L-ESN (0.106), and deep-ESN (0.126); at medium noise, df-ESN still maintains the lowest RMSE of 0.051, while other models show a notable increase in prediction error; even under high noise, df-ESN yields the smallest RMSE of 0.094, demonstrating superior noise robustness compared to the other three methods. Figure 20 also exhibits the superior noise robustness of the df-ESN compared to the baseline models.

Next, we investigate the effect of feedback gain γ on internal noise amplification. Experiments are conducted on the KS system with the following parameters: $\rho = 0.5$, $\xi = 10^{-4}$, and $m = 3$. Zero-mean Gaussian noise with standard deviation 0.001 is added to the state update to simulate internal disturbances. The activation function is $f(.) = \tanh(.)$, and thus the Lipschitz constant is $L_f = 1$, and $\beta \approx \rho = 0.5$ in Theorem 3, giving the stability boundary $|\gamma| \to 0.5$. Four representative feedback gains are selected: $\gamma = -0.1, -0.3, -0.5, -0.6$. For each γ, we investigate the evolution of the error ($\text{RMSE}_{\text{noise}}$) between the noisy prediction data and the noise-free prediction data. The results are illustrates in Fig. 21.

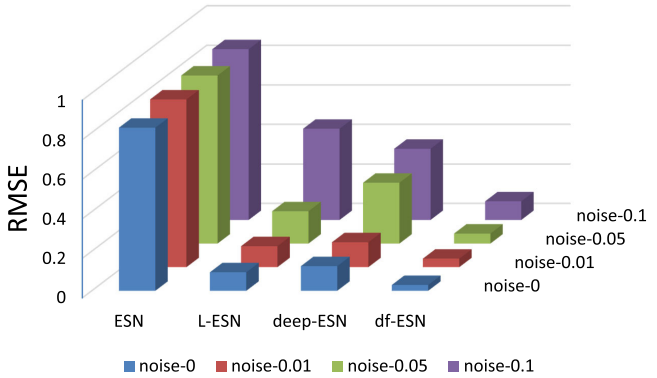


FIG. 20. The prediction errors under different noise levels. Across different noise intensities, the df-ESN model consistently maintains the lowest prediction error, demonstrating superior noise robustness and prediction stability.

When $\gamma = -0.1$ and -0.3 which is far away from the stability boundary, the RMSE remains consistently low over the prediction horizon, indicating effective internal noise be negligible. However, as γ approaches the stability boundary, noise amplification becomes evident: for $\gamma = -0.5$, the RMSE begins to rise sharply after approximately 200 steps, while for $\gamma = -0.6$ (beyond the boundary), the error grows even earlier and more rapidly, severely degrading prediction performance. This result is consistent with the H_∞ norm analysis in Theorem 3. The df-ESN achieve a much lower internal noise amplification when $|\gamma|$ lies far from the stability boundary, which ensures stable and robust performance with practical parameter configurations.

To evaluate whether the delayed feedback mechanism genuinely captures the system dynamics rather than fitting to noise, Fig. 22 presents the power spectral density (PSD) analysis of the predicted signals and corresponding errors under different delay parameters ($m = 3$ and 10) and noise scenarios (input noise and internal noise), where $\gamma = -0.1$ for all cases. Both types of noise are zero-mean Gaussian white noise: the input noise has a standard deviation of 0.05, and the internal noise has a standard deviation of 0.001. For both noise scenarios, the system parameters are $(m, \gamma) = (3, -0.1)$.

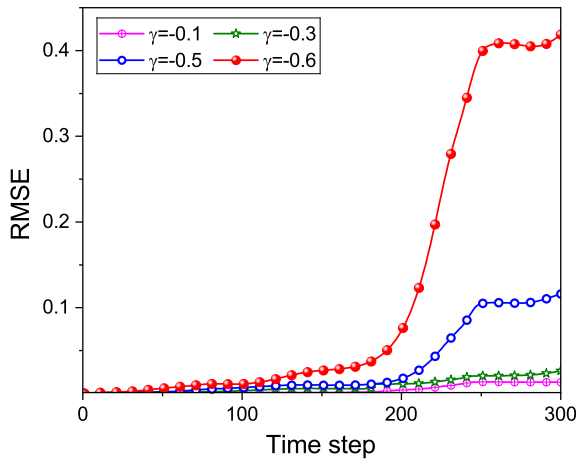


FIG. 21. Evolution of the RMSE between noisy and noise-free predictions under different feedback gains. It demonstrates that internal noise can be negligible when γ is far from the stability boundary.

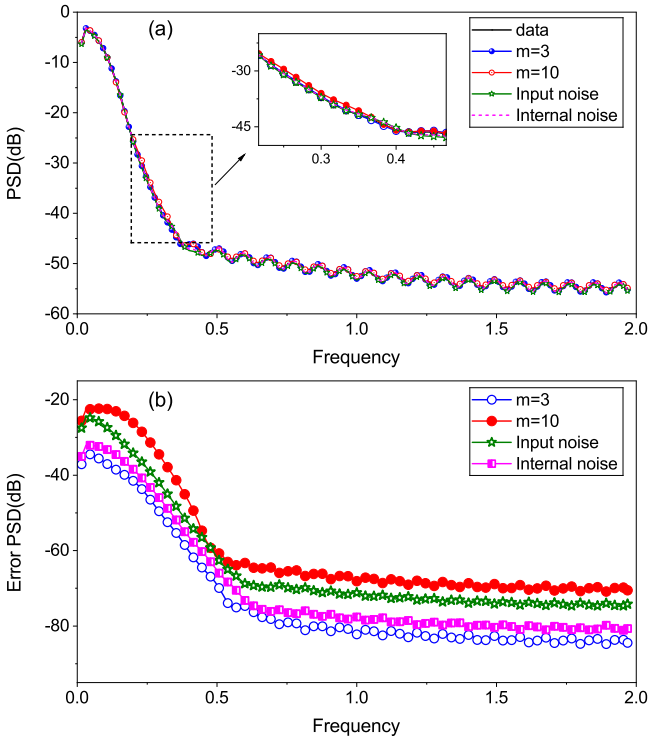


FIG. 22. Power Spectral Density (PSD) analysis of signals and prediction errors under different delay and noise conditions for KS system. (a) PSD of true/predicted signals. (b) PSD of prediction error.

As shown in Fig. 22(a), all spectra have a low-pass shape with power concentrated at low frequencies, decaying rapidly at high frequencies. The spectral shape and peak position remain unchanged under all test conditions, with no spurious components. Notably, the PSD under noisy conditions nearly overlaps the noise-free case, confirming strong robustness against external and internal disturbances. This demonstrates high fidelity in reproducing the KS system's inherent frequency structure.

Figure 22(b) shows that the error spectra are flat across all frequencies, indicating that errors arise from the intrinsic difficulty of long-term dynamics, not overfitting to high-frequency noise. Increasing the delay to $m = 10$ raises the average error power, but the PSD still shows no spurious peaks or high-frequency elevation. This suggests performance degradation is due to memory dilution (feedback arrives too late for slow dominant motions) rather than noise overfitting. Even under noisy conditions, the error PSD remains free of isolated peaks, ruling out harmful synchronization or overfitting, and further demonstrating robustness.

C. Reconstruction of bifurcation diagrams

We consider a three-dimensional system [58]:

$$\dot{x} = y, \quad \dot{y} = rz, \quad \dot{z} = -z - x^3 - zx^2 + xy^2 - xyz, \tag{19}$$

where r is a parameter.

In this experiment, the parameter-aware scheme [59] based on the df-ESN (details are provided in Appendix F) is

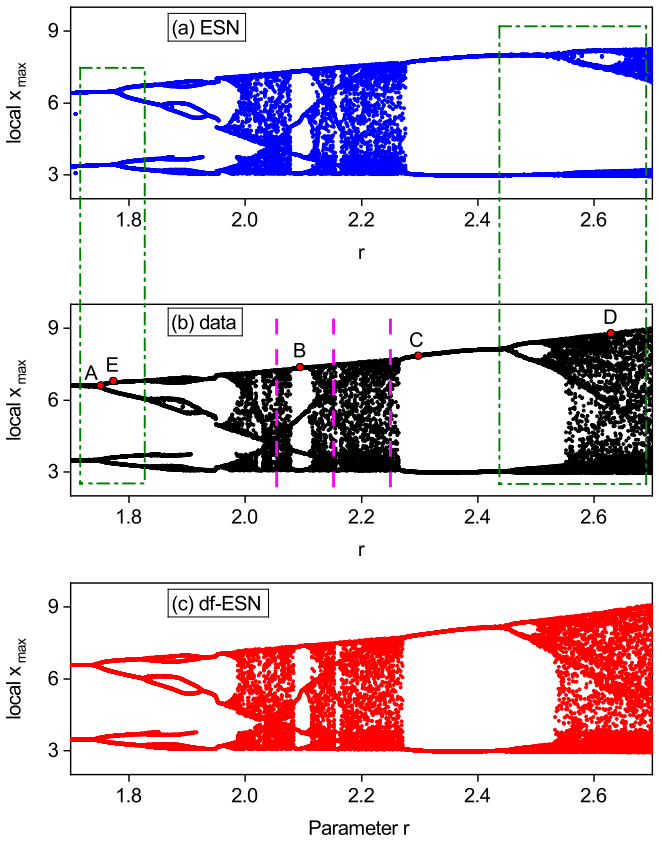


FIG. 23. Bifurcation diagrams using the ESN and df-ESN. The dashed rectangles highlight the deviations of the solutions between the ESN with the ground truth. The df-ESN achieves a favorable approximation to actual system solutions across broad parameter intervals.

employed to analyze the system with varying parameters. For a system exhibiting distinct dynamics over a fixed parameter set, the training data are collected from the system under a small number of parameters; once the ESN is sufficiently trained, the system states under other parameter values can be inferred using the trained model. When the parameter varies over a continuous range within the parameter set, the bifurcation diagram can be predicted from the inferred results. In the parameter-aware scheme, the parameters are introduced into the data form as follows: $\widetilde{data} = [\widehat{data}, p]^{\mathrm{T}}$, where, $\widehat{data}$ are the raw data, p is a parameter.

Using the Runge-Kutta method with time step $\triangle t = 0.05$, we compute the numerical solutions of Eq. (19) with three parameters $r = 2.05,\ 2.15,\ 2.15$ to generate the training data [see the purple dashed lines in Fig. 23(b)]. We set 5000 training data points for each value of r, resulting in a total of 15 000 training data points. The parameters of the df-ESN are set as $N = 1000$, $\xi = 10^{-7}$, $\rho = 0.5$, $\eta = 0.1$, $\gamma = 0.1$, and $m = 3$.

Figure 23 shows the bifurcation diagram of the system, where the horizontal axis represents the values of parameter r and the vertical axis represents the local maximum value of x. Figures 23(a) and 23(c), respectively, present the bifurcation diagrams obtained using ESN and the df-ESN. The parameters in ESN model are $N = 1000$, $\rho = 0.9$, $\eta = 0.1$, and $\xi = 10^{-7}$. By comparing them with the real bifurcation diagram as shown in Fig. 23(b), ESN performs poorly for

TABLE VI. Predicted dynamics at some values of r.

Parameter r	True Dynamics	ESN	df-ESN
1.76	Period-4	Period-2	Period-4
2.10	Period-4	Period-4	Period-4
2.30	Period-2	Period-2	Period-2
2.62	Chaos	multiperiod	Chaos

reconstructing the period-doubling bifurcation diagrams in some values of r, especially when r is far away from the three training parameters. In contrast, the bifurcation diagram predicted by the df-ESN match the real bifurcation diagram very well for all parameter $r \in [1.7, 2.7]$.

We further evaluate the model's prediction performance at several representative values of r, as summarized in Table VI. Notably, system exhibits the chaotic dynamics at all training parameters ($r = 2.05, 2.15, 2.25$), and the model is never trained on periodic orbits. Nevertheless, both ESN models successfully reconstruct the real periodic behaviors at point B ($r = 2.10$) and point C ($r = 2.30$) near the training set, demonstrating that the model does not merely memorize training trajectories but learns the intrinsic mapping between system parameters and dynamical responses.

Conversely, at the unseen period-4 point E ($r = 1.76$) and the chaotic point D ($r = 2.62$), the df-ESN produces predictions consistent with the true dynamics (see Figs. 24 and 25), whereas the ESN exhibits clear deviations. These results confirm that the df-ESN can reliably extrapolate to unseen periodic windows and chaotic regimes beyond the parameter range of the training data.

Figure 25 illustrates the model prediction on both sides of the critical bifurcation point A at approximately $r = 1.75$. At $r = 1.74$ on the left of point A and $r = 1.76$ on the right of point A, the df-ESN correctly captures the period-2 and period-4 dynamical states, respectively. In comparison, the ESN fails to predict the period-4 solution at $r = 1.76$. This further confirms that the df-ESN is capable of resolving critical bifurcation structures even at parameter values unseen in the training set.

In Fig. 26, two error metrics are utilized to evaluate the generalization capability of df-ESN. The first is $\mathrm{RMSE}_{\mathrm{seq}}$ which is calculated using predicted and real sequences within the first 100 time steps for each parameter setting. It is affected by initial phase differences and reflects short-term trajectory prediction accuracy. The second is bifurcation diagram distribution error, denoted as $\mathrm{RMSE}_{\mathrm{bif}}$. We extract local maxima from time series data, sort the data, select identical

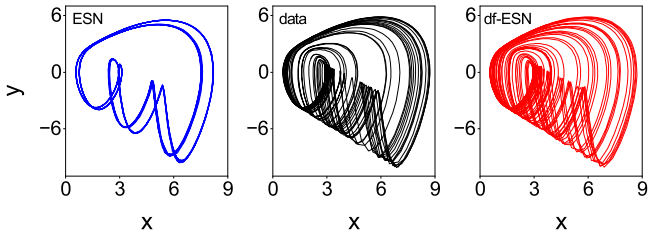
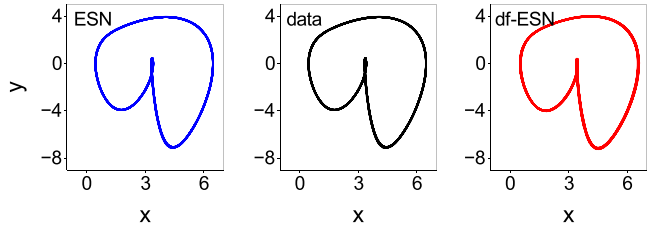
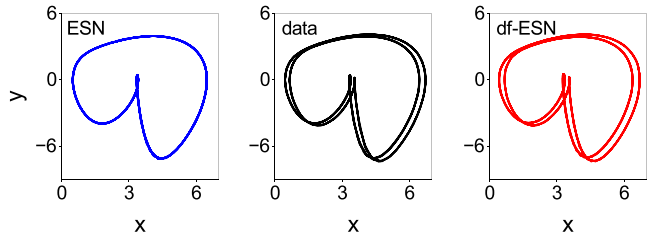


FIG. 24. Illustrates the dynamical behavior when $r = 2.62$. Clearly, the df-ESN correctly captures the real chaotic solutions, while the ESN cannot.



(a) Prediction dynamics of the models when $r = 1.74 < 1.75$.



(b) Prediction dynamics of the models when $r = 1.76 > 1.75$.

FIG. 25. Illustrates the dynamical behavior of the system near the critical bifurcation point A. This verifies that the df-ESN can resolve critical bifurcation structures at parameter values not encountered during training.

quantile points, and compute their RMSE. This metric is immune to temporal offset and misalignment, verifying whether the model captures the overall statistical features and dynamic state distribution, rather than merely accurate pointwise prediction.

Notably, the $\mathrm{RMSE}_{\mathrm{bif}}$ is strongly correlated with the dynamical complexity of the system. The df-ESN maintains consistently low error in periodic regimes, where the dynamics are simple and stable. Local error peaks appear only

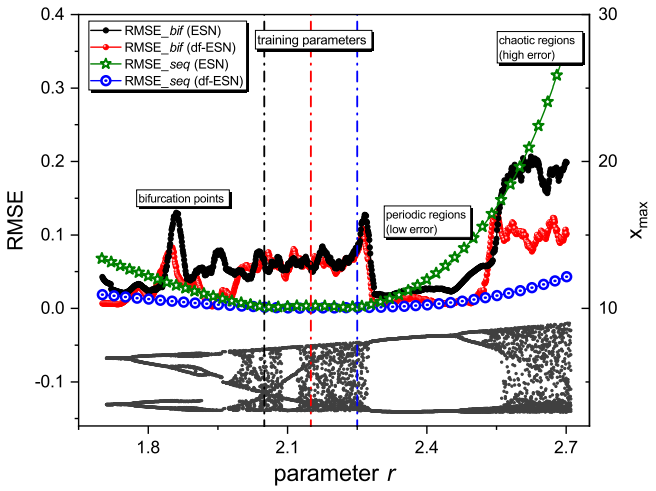


FIG. 26. Comparison of the prediction errors $\mathrm{RMSE}_{\mathrm{bif}}$ and $\mathrm{RMSE}_{\mathrm{seq}}$ with respect to the parameter r. The df-ESN exhibits consistently low and stable $\mathrm{RMSE}_{\mathrm{seq}}$ across the entire parameter range. While $\mathrm{RMSE}_{\mathrm{bif}}$ remains low over most intervals, it rises sharply around bifurcations and chaotic regions, and these error variations faithfully capture the system's dynamical features.

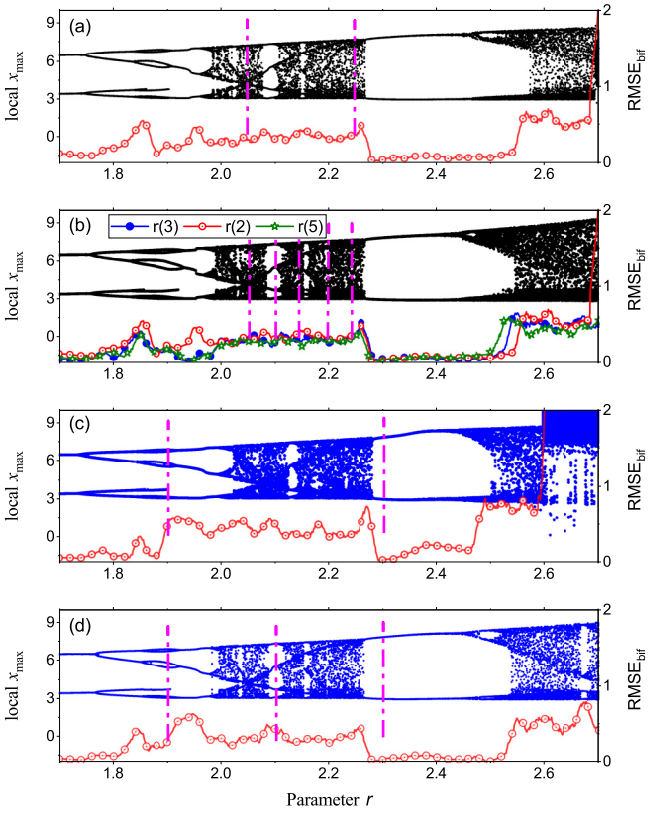


FIG. 27. Influence of training parameter density and dynamical regime on bifurcation diagram reconstruction: (a) and (b) chaotic training data (2 vs 5 parameters); (c) and (d) periodic training data (2 vs 3 parameters). Left axis: local $x_{\max}$; right axis: $\mathrm{RMSE}_{\mathrm{bif}}$; magenta lines: training parameters.

near critical bifurcation points, where the system undergoes topological transitions, or in chaotic regions, where the attractor structure is highly complex and sensitive to parameter changes. This results confirms its ability to capture long term statistical features of the system, rather than being limited to interpolation between known training points.

The key finding is that $\mathrm{RMSE}_{\mathrm{seq}}$ of the ESN (green-star curve) shows low error only near the training parameters, with a rapid and significant increase in error as the distance to these points grows, especially in the extrapolated chaotic regime for $r > 2.5$. This clealy shows that the ESN struggles to generalize to new dynamical regimes. In contrast, the $\mathrm{RMSE}_{\mathrm{seq}}$ of the *df*-ESN (blue circular curve) maintains consistently low and stable error across the entire parameter range, even far from the training set. This indicates that the df-ESN successfully learns the fundamental structure of the system, including periodic orbits, bifurcation transitions, and chaotic attractors, rather than merely memorizing the training data. The superior performance highlights the improved generalization ability of *df*-ESN.

Next, Fig. 27 demonstrates the influence of training parameter selection on the generalization ability of df-ESN. In Fig. 27(b), training with five chaotic parameters (2.05, 2.10, 2.15, 2.20, 2.25) yields superior results compared to adopting only two parameters ($r = 2.05, 2.25$) in Fig. 27(a)

and three parameters in Fig. 23. Increasing the quantity of training parameters enables the model to capture more system characteristics and boost prediction accuracy. Nevertheless, the performance gain is limited when expanding parameters from three to five, implying that moderate training parameters are adequate without excessive augmentation.

In Fig. 27(c), training with periodic parameters $r = 1.9$ and 2.3 enables rough bifurcation prediction, yet the wide parameter interval causes higher RMSE for $r > 2.5$. Further introducing $r = 2.1$ into the training set in Fig. 27(d) effectively optimizes the prediction performance.

A comparison between Figs. 27(a)–27(d) shows that the optimal strategy is to adopt appropriate dense sampling of chaotic data, which ensures low reconstruction error and strong generalization. Sparse parameter sampling or pure periodic data training will cause serious reconstruction failure in chaotic regions, which can be compensated by increasing the sampling density.

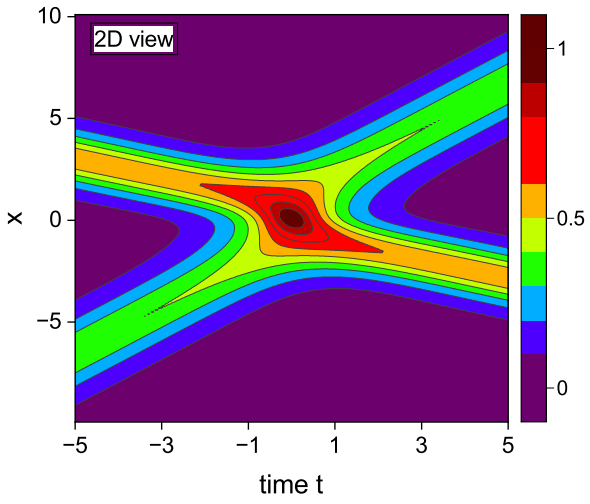


FIG. 28. Double-solitons in the Schrödinger equation.

D. Solitons in the Schrödinger equation

In this section, we demonstrate the effectiveness of the df-ESN in predicting solitons governed by the Schrödinger equation. The Schrödinger equation is pivotal in the field of integrable system theory and has extensive applications in various domains, notably in space plasma physics and nonlinear optics [60]. As a partial differential equation with complex coefficients, the Schrödinger equation is inherently highly nonlinear, making the prediction of spatiotemporal dynamics particularly challenging.

We consider a nonlinear Schrödinger equation described by [60]

$$\mathrm{i}u_t + a_1 u_{xx} + a_2|u|^2 u - \mathrm{i}\left(a_3 u_{xxx} + a_4|u|^2 u_x + a_5 u|u|^2{}_x\right) = 0, \tag{20}$$

with boundary and initial conditions:

$$u(x_0, t) = q_1(t),\ u(x, t_0) = q_2(x),$$

where $a_1 = 1/2,\ a_2 = 4,\ a_3 = 1/4,\ a_4 = 6,\ a_5 = -6;\ u = u(x, t)$ is the complex-valued solutions depending on a spatial variable $x \in [-L_1, L_2]$ and time $t \in [-T, T]$, $\mathrm{i}^2 = -1$, the subscripts "x" and "t" denote the partial derivatives with respect to x and t, respectively.

Equation (20) has the double soliton solutions $u = G/F$, where F and G are written in an explicit form [60]:

$$\begin{aligned} G &= (\alpha_1 + \alpha_3)\cosh(A_2 e^{\mathrm{i}B_1}) + (\alpha_2 + \alpha_4)\cosh(A_1 e^{\mathrm{i}B_2}) \\ &\quad + \mathrm{i}\alpha_5[\sinh(A_2 e^{\mathrm{i}B_1}) - \sinh(A_1 e^{\mathrm{i}B_2})], \\ F &= b_1\cosh(A_2 + A_1) + b_2\cosh(A_2 - A_1) \\ &\quad + b_3\cos(B_2 - B_1). \end{aligned} \tag{21}$$

The parameters in Eq. (21) can be found in Ref. [60].

A soliton, also termed as a solitary wave, is a wave that maintains its shape when it moves at a constant speed. The 2D view of the solitary waves from Eq. (20) is shown in Fig. 28, where the horizontal axis signifies the evolution time, the vertical axis represents the spatial position, and colors indicate the values of $|u|$. From this figure, a remarkable feature of the double solitary waves is their ability to cross each other without any alteration in their properties. Their magnitude, shape, and velocity remain perfectly conserved after the interaction.

From Eqs. (20) and (21), it is clear that the interaction of two solitary waves occurs at $t = 0$. In this experiment, our goal is to accurately predict the propagation behavior of the two waves both before and after their collision using the df-ESN. By solving Eq. (20) in the spatial region [–10,10] and the time region [–5,5], we generate the sampling data of this double solitons.

1. Two solitary waves before collision

With step size Δx=0.1, and a time step size $\Delta t = 0.001$, we generate the data of the double solitons at a grid resolution of 200×10000. Before collision, we selected 2000 data for training with $t \in [-5, -3)$ and 3000 data for testing where $t \in [-3, 0]$ to predict the dynamics near the collision point. Using the grid-searching technique in Sec. III C, we determine a set of optimal parameters $\gamma = -0.3$ and $m = 2$, and other parameters in the df-ESN are $N = 500$, $\xi = 10^{-10}$, and $\rho = 0.5$, $\eta = 0.1$.

Figures 29(a) and 29(b) illustrate the evolution patterns of $|u|$ using the df-ESN and ESN, where the colors indicate the predicted values of $|u|$ at (t_i, x_j) with $t_i \in [-5, 0]$ and $x_j \in [-10, 10]$. The parameters in ESN models are $N = 500$, $\rho = 0.9$, $\eta = 0.2$, and $\xi = 10^{-6}$. Figures 29(c) and 29(d) visualize the corresponding errors between the predicted and real data, with the lighter-colored areas denoting the smaller errors. Clearly, for the short-term evolution from predicting time $t = -3$, ESN provides a relatively accurate approximated solution, as evidenced by the pale purple areas in Fig. 29(d), but diverges from the actual data as time evolves. However, the predicted solutions from df-ESN accurately tracks the real data for the long term evolution even around the collision point when $t = 0$ as shown in Figs. 29(a) and 29(c).

Figure 30 illustrates the prediction of the two solitary waves in the $x - |u|$ plane for some times, where the black circle, red solid curve and blue dotted curve represent the results from the data, df-ESN, and ESN, respectively. Clearly,

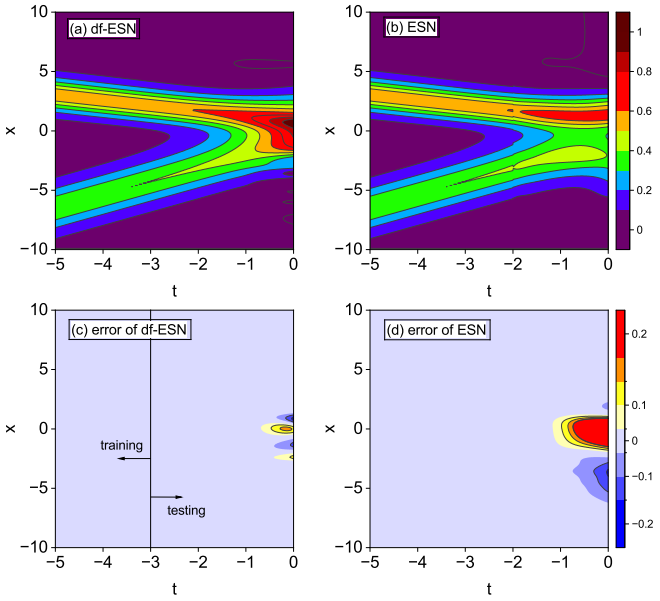


FIG. 29. Prediction of solitary waves before collision. (a) and (b) Evolution patterns; (c) and (d) prediction errors.

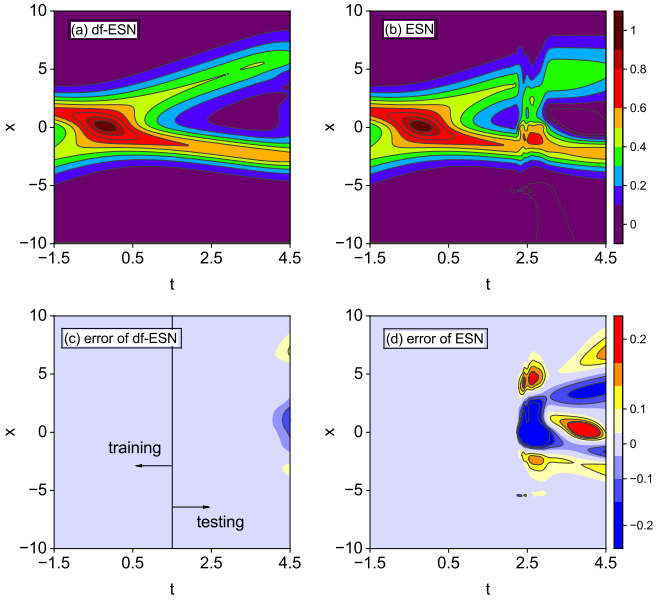


FIG. 31. Prediction of the solitary waves after collision. (a) and (b) Evolution patterns; (c) and (d): evolution error with the grayer areas denoting the smaller error.

when $t = -2$ and -1, df-ESN succeeds in capturing the double-peak solitary waves as shown in Figs. 30(a) and 30(b), and also predicting the occurrence of the single-peak parent entity when $t = 0$ [see Fig. 30(c)]. In contrast, ESN fails to accurately predict such behavior when the evolution time is relatively long as shown the blue dotted curves in Figs. 30(b) and 30(c).

Figure 30(d) shows the predicted solution of the solitary waves using df-ESN when $t = -3$, -2, -1, and 0. Clearly, for each time moment, a fast solitary wave (with smaller amplitude) chases after another slow solitary wave (with bigger amplitude), and the peaks of the two waves gradually

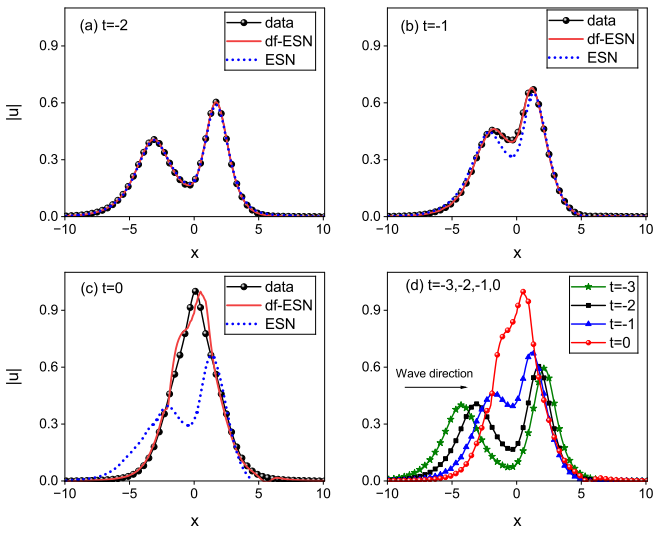


FIG. 30. The prediction for two solitary waves in the $(x - |u|)$ plane: (a) $(t = -2)$; (b) $(t = -1)$; (c) $(t = 0)$; and (d) $(t = -3)$, (-2), (-1), (0). Clearly, for the long-term evolution, df-ESN can capture the solitary waves with two peaks and a single parent entity.

approach each other with an increase of time. Finally, these two waves collide with each other and merge as a parent entity as shown by the single-peak wave at $t = 0$. This prediction process is consistent with the evolution of the actual system [60] which indicates the ability of the df-ESN in predicting the dynamical behavior of the large nonlinear system.

2. Two solitary waves after collision

To investigate the solitary waves after collision, we selected 3000 data for training with $t \in [-1.5, 1.5)$ and 3000 data for testing where $t \in [1.5, 4.5]$. We set the reservoir parameters as $N = 500$, $\xi = 10^{-10}$, $\rho = 0.5$, $\eta = 0.1$, $\gamma = -0.3$ and $m = 2$. The selected parameters in ESN models are: $N = 500$, $\rho = 1$, $\eta = 0.1$, and $\xi = 10^{-5}$.

Figures 31(a) and 31(b) illustrates the system's evolution patterns, and Figs. 31(c) and 31(d) show the predicted errors between the predicted and actual solutions using df-ESN and ESN. Clearly, for the short-term evolution from starting time $t = 1.5$, ESN provides a relatively accurate approximate solution, as evidenced by the gray areas in Fig. 31(d), but diverges from the actual data as time evolves. However, the predicted two-solitary waves from df-ESN accurately tracks the actual data for the long term evolution even when the evolution time is about $t = 4.5$ as shown in Fig. 31(c).

Figure 32 illustrates the prediction of the two solitary waves in the $x - |u|$ plane after collision for some discrete times. Clearly, when $t = 2$, df-ESN succeeds in capturing the double-peak solitary waves as shown in Fig. 32(a). Additionally, compared to the solitary waves before the collision at $t = -2$, the two solitary waves post-collision retain their original speeds and shapes unchanged. However, their positions are switched such that they move in opposite directions, leading to monotonic separation of wave peaks, as depicted in Fig. 32(b).

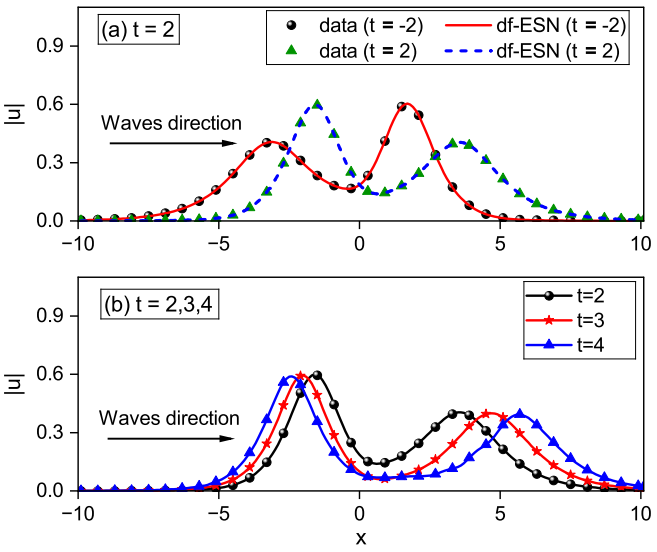


FIG. 32. The prediction for two solitary waves in the $x - |u|$ plane: (a) $t = 2, -2$ and (b) $t = 2, 3, 4$. Clearly, the two solitary waves postcollision retain their original speeds and shapes unchanged, but their positions are exchanged in a way that the slow wave now chases after the fast wave, which resulted in two peaks gradually moving away.

The results verify that *df*-ESN achieves accurate prediction of soliton dynamics described by the nonlinear Schrödinger equation (NLSE). Beyond algorithmic innovation, this study provides a promising solution for practical scientific and engineering issues. Concretely, the proposed method has wide application potential. In optical communications, it realizes rapid and precise prediction of soliton evolution and interaction, facilitating the design of high-speed long-distance fiber communication systems to enhance signal stability and bandwidth utilization. In fluid dynamics and nonlinear physics, it efficiently characterizes complex soliton behaviors, supporting investigations on diverse wave phenomena (e.g., soliton pulse control via black phosphorus saturable absorbers [61]) and promoting further exploration in nonlinear dynamics research. Accordingly, *df*-ESN is not only a theoretical model but also an effective tool with broad practical application prospects.

3. Analysis of delay precision and temporal resolution

Due to the sensitivity of soliton dynamics to phase and timing, we propose a fractional delay extension to the `df`-ESN using linear interpolation to refine the delay mechanism:

$$\tilde{\mathbf{x}}(t-m) = (1-\chi)\mathbf{x}_{t-\lfloor m\rfloor} + \chi\mathbf{x}_{t-\lceil m\rceil},$$

where $\chi = m - \lfloor m\rfloor$, m can take any positive real value, with $\lfloor m\rfloor$ (floor function) and $\lceil m\rceil$ (ceiling function) denoting downward and upward rounding, respectively. The fractional delay approach offers greater flexibility in aligning the model's memory structure with the characteristic time scales of soliton interactions, theoretically improving temporal precision.

To quantitatively assess prediction accuracy, we take the waveform at the collision moment $t = 0$ as the reference and employ three complementary metrics [62]: (i) The Matthews correlation coefficient (MCC), which measures the morphological similarity between predicted and true waveforms; its value ranges from -1 to 1, where a value closer to 1 indicates a higher morphological consistency between the two waveforms. (ii) The intersection over union (IoU), which quantifies the spatial overlap between the two waveforms; a higher IoU implies better spatial coincidence. (iii) The spatial shift, which measures the deviation in peak position between the predicted and true waveforms; it quantifies how much the predicted peak drifts spatially relative to the real one, normalized by the soliton width for physical interpretability.

TABLE VII. Prediction accuracy versus time step Δt at $t = 0$.

Δt	MCC	Spatial shift	Relative offset	IoU
0.1	0.965	–0.4	10.5%	0.861
0.1 (fractional)	0.976	–0.3	7.9%	0.861
0.01	0.981	–0.2	5.3%	0.933
0.01 (fractional)	0.986	–0.2	5.3%	0.939
0.001	0.992	–0.1	2.6%	0.947
0.001 (fractional)	0.985	–0.1	2.6%	0.949
0.0001	0.993	–0.1	2.6%	0.949

Table VII lists the prediction accuracy at $t = 0$ under varying time steps Δt. As Δt decreases from 0.1 to 0.001, MCC rises from 0.965 to 0.992, IoU increases from 0.861 to 0.947, and spatial shift drops from -0.4 (10.5% soliton width) to -0.1 (2.6%). At coarse temporal resolution ($\Delta t = 0.1$), fractional delay surpasses integer delay; yet their performance difference gradually diminishes with smaller Δt, and they produce nearly identical results at $\Delta t = 0.001$. Further reducing Δt brings only trivial accuracy improvement, indicating performance saturation.

In small time-step scenarios, integer delay can well characterize phase-related dynamics and reach comparable precision to fractional delay, where the tiny spatial shift is physically negligible, thus balancing prediction accuracy and computational cost. Nevertheless, discrete delay inevitably suffers from gradual phase error accumulation over long-term prediction. For systems with low temporal resolution or high phase sensitivity, fractional delay or continuous-time frameworks such as neural ODE are more favorable to attain better prediction performance.

V. CONCLUSIONS AND DISCUSSIONS

This paper proposes an echo state network with delayed mechanism (`df`-ESN) for dynamcial analysis of the complex systems. By integrating delayed state feedback into the reservoir, `df`-ESN explicitly models the influence of past inputs on the current state. This design significantly enhances memory retention and improves long-term predicted behavior of the complex systems. Mathematically, we establish a comprehensive analytical framework for analyzing the dynamical behavior, and systematically studied the synergistic interplay between feedback gain and delay steps. This approach ensures model stability while optimizing prediction precision, effec-

tively addressing the well-known challenges associated with parameter tuning in ESN.

Experimental results demonstrate that df-ESN not only achieves superior long-term prediction accuracy in chaotic systems but also significantly enhances the precision of both attraction basins prediction and bifurcation diagram reconstruction. Furthermore, we extend the application of df-ESN to the modeling of soliton evolution. The experimental results using df-ESN not only confirms the model's capability in modeling short-term soliton behavior, but also highlights the inherent challenges in capturing long-term dynamics for systems with complex spatiotemporal nonlinearities. These results highlight the computational strength and potential of the proposed df-ESN for its applications in complex systems analysis.

We summarize the distinct advantages of the proposed df-ESN over existing ESN models including hierarchical delayed memory ESN and L-ESN as follows:

(i) *Structural and theoretical superiority.* Different from hierarchical delayed memory ESN that captures multi-scale temporal features via layer stacking, df-ESN embeds a unified delayed feedback structure within a single reservoir. Its complete theoretical framework verifies local and global echo state properties, guaranteeing network stability and rationality.

(ii) *Essential mechanism difference from L-ESN.* L-ESN adjusts memory decay via leaky integration, while df-ESN incorporates delayed internal states into reservoir updating to fully utilize historical information. It constructs abundant temporal dynamics and explicitly captures long-range dependencies, instead of only tuning state attenuation.

(iii) *Promoted memory capacity.* Both theoretical analysis and experimental results prove that embedded delayed states effectively boost the memory capacity of df-ESN against conventional leaky and hierarchical delayed models. Such improvement originates from the inherent delayed feedback loop for historical information retention and utilization, rather than simple feedback regulation.

We offer the following additional discussions on the proposed df-ESN model.

(i) ESN models equipped with nonlinear readouts enhance representation capacity [63–66]. They maintain original reservoir dynamics and employ nonlinear mapping to strengthen feature decoding and break linear limitations. In comparison, df-ESN optimizes reservoir dynamics and improves the quality of the dynamical representation itself using delayed feedback, which effectively captures long-range dependencies and initial condition sensitivity. The two methods are complementary: delayed feedback generates superior reservoir states, while nonlinear readout efficiently extracts predictive information from these states.

(ii) By explicitly introducing time-delayed reservoir states, df-ESN fully exploits historical temporal information. This delay mechanism endows the model with abundant dynamic characteristics, greatly improving its representation and expression ability. The embedded delay structure in df-ESN brings distinctive advantages absent in standard ESN, enabling it to better capture complex system dynamics.

(iii) The df-ESN integrates delayed feedback loops into reservoir dynamic equations, regarding delay as an optimizable internal structure instead of external preprocessing or fixed physical attribute. This design differs from Ref. [67], which adopts fixed delay parameters only for input preprocessing and phase space reconstruction without modifying reservoir internal dynamics. It also distinguishes from the model in Ref. [68], where delay is determined by physical topology and remains unadjustable for performance improvement. Furthermore, this work establishes a complete theoretical framework, investigates memory enhancement and echo state characteristics, and verifies model stability theoretically, which is another core distinction from the above researches.

(iv) Although df-ESN is a purely data-driven model requiring no prior physical knowledge, the optimal delay m derived from K-S experiments always ranges from 1 to 10. Linear stability analysis yields the maximum unstable wave number $k_{\max} \approx 0.707$ and maximum growth rate $\sigma_{\max} = 0.25$, corresponding to a linear characteristic time $\tau_{\text{lin}} = 4$ dimensionless units, i.e., 16 discrete steps with $\Delta t = 0.25$. The smaller optimal m indicates the learned delay does not characterize pure exponential growth, but reflects the advection timescale of dominant spatial patterns or instability saturation time. For soliton dynamics, the optimal m is less than 3, matching the instantaneous interaction feature of solitons. Mapping optimal hyperparameters (m, γ) to physical quantities such as Lyapunov time and advection velocity is a worthy research direction, which can further evolve df-ESN from a black-box predictor into a data-driven system identification tool. This study mainly concentrates on prediction performance and feature extraction, while in-depth physical interpretation is reserved for future work.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants No. 12572009 and No. 12072128.

W.G.: Conceptualization, Methodology, Investigation, and Software; S.L.: Methodology and Software; J.L.: Investigation and Software; E.L.: Investigation and Draft Preparation; X.X.: Conceptualization, Methodology, Investigation, and Reviewing.

The authors declare that there is no conflict of interest.

DATA AVAILABILITY

There are no publicly available research data or software supporting this manuscript. Requests for further information or data should be sent to the authors.

APPENDIX A: PROOF OF THEOREM 1

Proof. We construct the Lyapunov-Krasovskii functional:

$$\begin{aligned} V(t) &= V_1(t) + V_2(t) + V_3(t) \\ &= \|\mathbf{e}_t\| + |\gamma| \sum_{k=t-m}^{t-1} \|\mathbf{e}_k\| + \mu \sum_{\theta=-m}^{-1} \sum_{k=t+\theta}^{t-1} \|\mathbf{e}_k\|, \end{aligned} \tag{A1}$$

where $\mu > 0$ is a positive constant to be determined, and m is the delay step, $V_1(t) = \|\mathbf{e}_t\|$, $V_2(t) = |\gamma| \sum_{k=t-m}^{t-1} \|\mathbf{e}_k\|$, and $V_3(t) = \mu \sum_{\theta=-m}^{-1} \sum_{k=t+\theta}^{t-1} \|\mathbf{e}_k\|$.

It is clear that $V(t) \geqslant 0$, and $V(t) = 0$ if $\mathbf{e}(s) = \mathbf{0}$ for all $s \in [t-\tau, t]$. The $V_3(t)$ acts as a "memory penalty" on the error energy over the past m time units. A larger m imposes a stronger penalty on historical errors.

We compute the forward difference of $V(t)$ along the error system (5): $\Delta V(t) = V(t+1) - V(t)$. The difference of each term is computed as follows:

$$\Delta V_1(t) = \|\mathbf{e}_{t+1}\| - \|\mathbf{e}_t\|,$$

$$\begin{aligned}\Delta V_2(t) &= |\gamma| \left(\sum_{k=t+1-m}^{t} \|\mathbf{e}_k\| - \sum_{k=t-m}^{t-1} \|\mathbf{e}_k\| \right) \\ &= |\gamma| (\|\mathbf{e}_t\| - \|\mathbf{e}_{t-m}\|),\end{aligned}$$

$$\begin{aligned}\Delta V_3(t) &= \mu \sum_{\theta=-m}^{-1} \left(\sum_{k=t+1+\theta}^{t} \|\mathbf{e}_k\| - \sum_{k=t+\theta}^{t-1} \|\mathbf{e}_k\| \right) \\ &= \mu \sum_{\theta=-m}^{-1} (\|\mathbf{e}_t\| - \|\mathbf{e}_{t+\theta}\|) \\ &= m\mu \|\mathbf{e}_t\| - \mu \sum_{k=t-m}^{t-1} \|\mathbf{e}_k\|.\end{aligned}$$

Combining these results, the total difference is

$$\begin{aligned}\Delta V(t) = \|\mathbf{e}_{t+1}\| - \|\mathbf{e}_t\| + |\gamma| (\|\mathbf{e}_t\| - \|\mathbf{e}_{t-m}\|) \\ + m\mu \|\mathbf{e}_t\| - \mu \sum_{k=t-m}^{t-1} \|\mathbf{e}_k\|.\end{aligned}$$

Substituting the error bound (6) into the above expression yields

$$\Delta V(t) \leqslant (L_f \|\mathbf{W}_{\text{res}}\| + |\gamma| + m\mu - 1) \|\mathbf{e}_t\| - \mu \sum_{k=t-m}^{t-1} \|\mathbf{e}_k\|.$$

Since $\mu > 0$ and $\sum_{k=t-m}^{t-1} \|\mathbf{e}_k\| \geqslant 0$, the last term is nonpositive. Therefore, a sufficient condition for $\Delta V(t) < 0$ for all non-zero $\mathbf{e}_t$ is $L_f \|\mathbf{W}_{\text{res}}\| + |\gamma| + m\mu < 1$.

By the Lyapunov-Krasovskii stability theorem for time-delay systems, $V(t)$ is a positive-definite, radially unbounded functional with strictly negative-definite derivative (except at the origin). So the error system is globally uniformly asymptotically stable. Thus, $\lim_{t\to\infty} \|\mathbf{e}(t)\| = 0$ for any initial condition. This completes the proof. ■

APPENDIX B: DERIVATION OF EQ. (9)

We introduce a new time scale as $T = t/\delta t$, and then $\mathrm{d}T/\mathrm{d}t = 1/\delta t$. When $\delta t \to 0$, it is easy from (8) to have

$$\begin{aligned}\frac{dx(t)}{dt} &= \frac{dx(t)}{dT}\frac{dT}{dt} = \frac{dx(t)}{dT}\frac{1}{\delta t} \\ &= \frac{1}{\delta t}[-x(t) + \gamma x(t-m) + f(\tilde{x}(t))].\end{aligned}$$

Define $x(t) = x(T\delta t) := y(T)$. Then we have

$$\frac{dy(T)}{dT} = -y(T) + \gamma y(T-m) + f(\tilde{y}(T)).$$

We simply set $t = T$ and $x(t) = y(T)$, and Eq. (9) follows accordingly. ■

APPENDIX C: PROOF OF THEOREM 2

Proof. The characteristic equation of Eq. (11) is

$$\det(\lambda) = \prod_{j=1}^{n} (\lambda + 1 - \sigma_j - \gamma \mathrm{e}^{-\lambda\tau}) = 0, \tag{C1}$$

where $\sigma_j = \rho_j \sigma$ with the jth eigenvalue ρ_j of $\mathbf{W}_{\text{res}}$.

In Eq. (C1), replacing ρ_j with spectral radius $\rho = \max_j |\rho_j|$ yields a conservative sufficient stability condition, because stability under the worst-case perturbation ρ ensures stability for all ρ_j with smaller magnitudes. In this case, we consider the following equation:

$$\mathcal{Q}(\lambda) = \lambda + \kappa - \gamma \exp(-\lambda\tau), \tag{C2}$$

where $\kappa = 1 - \rho\sigma > 0$ if $\rho\sigma < 1$.

When $\tau = 0$, Eq. (C2) has root $\lambda = \gamma - \kappa$ with having a negative real part when $\gamma < \kappa$.

If $\gamma > \kappa$, it is easy to see that $\mathcal{Q}(0) = \kappa - \gamma < 0$, and $\mathcal{Q}(\lambda \to +\infty) > 0$. Because $\mathcal{Q}(\lambda)$ is a continuous function of λ, there exists at least a $\lambda^* > 0$ such that $\mathcal{Q}(\lambda^*) = 0$. This fact implies that the trivial equilibrium of Eq. (10) is always unstable for $\gamma > \kappa$.

When $\gamma < \kappa$, assume $\lambda = \mathrm{i}\omega$ $(\omega > 0)$ is the root of Eq. (C2). Substituting it into Eq. (C2) gives

$$\gamma \cos(\omega\tau) = \kappa, \ \gamma \sin(\omega\tau) = -\omega \tag{C3}$$

and

$$\gamma^2 = \omega^2 + \kappa^2, \ \ \tan(\omega\tau) = -\omega/\kappa. \tag{C4}$$

From the first equation of Eq. (C4), if $|\gamma| < \kappa$, Eq. (C4) has no positive real root ω implying that all the zeros of $\mathcal{Q}(\lambda)$ have the negative real parts for any $\tau \geqslant 0$. This is a delay-independent stability criterion; the system remains stable for any time delay (even an infinitely large time delay).

If $\gamma < -\kappa$, from Eq. (C4), we have $\omega = \sqrt{\gamma^2 - \kappa^2}$, and a series of critical time delays:

$$\tau_j = \frac{1}{\omega}[\arctan(-\omega/\kappa) + (j+1)\pi], \ j = 0, \pm 1, \ldots . \tag{C5}$$

We now show how the number of positive real parts for the eigenvalues changes when τ crosses each critical value τ_j. From Eq. (C2), it is easy to calculate the derivative of λ with respect to τ at $\lambda = \mathrm{i}\omega$ as

$$\lambda'|_{i\omega} = \left.\frac{-\gamma\lambda}{\tau\gamma + \exp(\lambda\tau)}\right|_{i\omega} = \frac{-\mathrm{i}\omega\gamma}{\cos(\omega\tau) + \tau\gamma + \mathrm{i}\sin(\omega\tau)}$$

and thus

$$\mathrm{Re}(\lambda'|_{i\omega}) = \omega^2 > 0. \tag{C6}$$

In derivation of the above equation, we have used the result of Eq. (C3).

Equation (C6) indicates that system adds a pair of eigenvalues with positive real parts for each crossing of τ to the right of τ_j. It is noted that Eq. (C2) with $\tau = 0$ has no roots with positive real part under the condition $\gamma < -\kappa$.

So, all roots of Eq. (C2) have negative real parts when $\tau \in [0, \tau_* \triangleq \min_j(\tau_j))$, where

$$\tau_* = \frac{1}{\omega}[\arctan(-\omega/\kappa) + \pi]. \tag{C7}$$

This completes the proof. ■

APPENDIX D: PROOF OF THEOREM 3

Proof. Consider two trajectories with identical external inputs: the noise-free trajectory $\mathbf{x}_t^0$ satisfying

$$\mathbf{x}_{t+1}^0 = \gamma \mathbf{x}_{t-m}^0 + f\big(\mathbf{W}_{\text{in}}\mathbf{u}_{t+1} + \mathbf{W}_{\text{res}}\mathbf{x}_t^0\big),$$

and the noisy trajectory $\mathbf{x}_t$ satisfying

$$\mathbf{x}_{t+1} = \gamma \mathbf{x}_{t-m} + f\big(\mathbf{W}_{\text{in}}\mathbf{u}_{t+1} + \mathbf{W}_{\text{res}}\mathbf{x}_t + \mathbf{Q}_t\big),$$

where $\mathbf{Q}_t$ represents internal noise.

Subtracting the two equations and applying the Lipschitz condition yields

$$\|\mathbf{e}_{t+1}\| \leqslant |\gamma|\|\mathbf{e}_{t-m}\| + \beta\|\mathbf{e}_t\| + L_f\|\mathbf{Q}_t\|,$$

where $\mathbf{e}_t = \mathbf{x}_t - \mathbf{x}_t^0$ and $\beta = L_f\|\mathbf{W}_{\text{res}}\|$.

Define $E_t = \|\mathbf{e}_t\|$ and $Q_t = L_f\|\mathbf{Q}_t\|$. For worst-case analysis, we treat the inequality as equality:

$$E_{t+1} = \beta E_t + |\gamma| E_{t-m} + Q_t. \tag{D1}$$

Applying the z-transform to Eq. (D1) under zero initial conditions yields

$$zE(z) = \beta E(z) + |\gamma| z^{-m} E(z) + Q(z),$$

where $E(z) = \sum_{t\geqslant 0} E_t z^{-t}$, $Q(z) = \sum_{t\geqslant 0} Q_t z^{-t}$.

Solving for $E(z)$ yields the transfer function as

$$G(z) = E(z)/Q(z) = \frac{z^m}{z^{m+1} - \beta z^m - |\gamma|}.$$

The characteristic equation of the homogeneous system ($Q_t = 0$) is $z^{m+1} - \beta z^m - |\gamma| = 0$. Under the given condition $|\gamma| + \beta + \mu m < 1$, all poles lie inside the unit circle (Theorem 1). Hence $G(z)$ is stable.

Next, we consider the noise gain bound. The frequency response is obtained by substituting $z = e^{j\omega}$ with $j^2 = -1$. The H_∞ norm $\|G\|_\infty = \sup_\omega |G(e^{j\omega})|$ measures the worst-case noise amplification.

At zero frequency ($z = 1$), we have

$$|G(1)| = \frac{1}{1 - \beta - |\gamma|} < \frac{1}{\mu m}.$$

For any ω, a crude bound is $|G(e^{j\omega})| \leqslant 1/(1 - \beta - |\gamma|)$ because $|e^{j\omega(m+1)} - \beta e^{j\omega m} - |\gamma|| \geqslant 1 - \beta - |\gamma|$. More conservatively, numerical experiments and simple analysis show that the maximum gain never exceeds $1/(1-\beta-|\gamma|)+1$. Since $1/(1 - \beta - |\gamma|) \leqslant 1/(\mu m)$, we obtain $\|G\|_\infty \leqslant 1/(\mu m) + 1$. This bound accounts for potential resonant peaks that may slightly increase the noise gain beyond the DC value. This completes the proof. ■

APPENDIX E: INITIAL-VALUE-AWARE SCHEME

In many real-world applications, the systems may start from various initial configurations, and understanding its behavior under different starting points is essential. The initial-value-aware `df`-ESN modle addresses this issue by training on data generated from a few initial values of the system. This enables the model to learn a more general representation of the system dynamics, making it capable of predicting not only the training initial conditions but also unseen initial states. Such a capability is crucial for applications where the initial state may vary or is not precisely known in advance.

(1) *Training data construction*. The training data comprises p distinct time series, each initiated from a unique initial condition $\mathbf{v}_i(0)$, where $i = 1, 2, \ldots, p$. These time series represent the temporal evolution of the same underlying deterministic system, but starting from different initial states. In practice, the initial values $\mathbf{v}_i(0)$ may be sampled from a predefined region of the system's state space, ensuring a diverse representation of possible starting configurations. All time series are assumed to have the same length L_{train}, allowing for consistent processing and alignment during model training.

For each time series, the input provided to the model is a concatenation of two components:

$$\{\mathbf{v}_i(0)\}_{q=0}^{L_0} \bigoplus \{\mathbf{v}_i(t)\}_{t=1}^{L_{\text{train}}-L_0}, \tag{E1}$$

where

(i) $\{\mathbf{v}_i(0)\}_{q=0}^{L_0}$: A sequence of repeated initial condition vectors (typically of dimension d), used to initialize the reservoir states. The repetition of the initial state over L_0 time steps helps stabilize the internal dynamics of the echo state network.

(ii) $\{\mathbf{v}_i(t)\}_{t=1}^{L_{\text{train}}-L_0}$: The actual dynamic trajectory of the system, representing its time evolution from the initial state $\mathbf{v}_i(0)$, typically of dimension d as well.

The concatenated input thus has a total length of L_{train} and provides the model with both the starting configuration and the subsequent evolution of the system.

(2) *Reservoir state matrix generation*. After independently processing each input sequence through the `df`-ESN model, for each input sequence, the reservoir of the `df`-ESN model generates a sequence of high-dimensional hidden states. From this sequence, the states from time step $L_1 + 1$ to L_{train} are extracted to form the time-dependent state matrix:

$$\mathbf{X}_i = [\mathbf{x}_i(L_1 + 1), \ldots, \mathbf{x}_i(L_{\text{train}})], \tag{E2}$$

where each $\mathbf{x}_i(t) \in \mathbb{R}^N$ is the reservoir state vector at time t, and N is the size of the reservoir (number of neurons). The overall state matrix is constructed by vertically stacking the individual state matrices from all p time series:

$$\mathbf{X} = [\mathbf{X}_1; \ldots; \mathbf{X}_p], \tag{E3}$$

resulting in a matrix of size $[p \times (L_{\text{train}} - L_1)] \times N$. This aggregated state matrix is then used to train the output weight matrix $\mathbf{W}_{\text{out}}$ using the method described in Sec. III.

(3) *Prediction procedure*. For a new initial value $\mathbf{v}_j(0)$, the input sequence is initialized with $\mathbf{v}_j(0)$ for the first L_0 time steps. The corresponding reservoir states $\mathbf{x}_j(q)$, with $q = 1, \ldots, L_0$, are computed using the reservoir dynamics. The first predicted output is obtained as

$$\mathbf{v}_j(L_0 + 1) = \mathbf{W}_{\text{out}}\mathbf{x}_j(L_0). \tag{E4}$$

For subsequent time steps ($t > L_0$), the prediction is performed iteratively by using the previously predicted value as the input to the model. The initial condition is fixed only during the first L_0 steps, after which the model relies solely on its learned internal representations. This auto-regressive mechanism allows the model to generate multistep predictions without requiring the true future states.

APPENDIX F: PARAMETERS-AWARE SCHEME

When the initial condition of a dynamical system is fixed, variations in system parameters—such as control gains, external forces, or coupling coefficients—can induce significant changes in the system's behavior, including transitions between periodic, quasiperiodic, and chaotic regimes. Understanding and predicting these parameter-dependent behaviors is critical in many scientific and engineering applications, such as tuning a physical system, exploring bifurcation diagrams, or analyzing sensitivity to external inputs. The parameters-aware `df`-ESN model can capture the relationship between system parameters and the resulting dynamics, even when those parameter configurations have not been explicitly encountered during training. By training on data generated under a variety of parameter values but a fixed initial condition, the model learns to generalize across the parameter space, enabling prediction of system states under previously unseen parameter settings.

(1) *Training data construction*. The training data consists of time series generated from the same deterministic system, with a fixed initial condition $\mathbf{v}_0$, but under different parameter values p_i, where $i = 1, 2, \ldots, g$. Each time series reflects the system's temporal evolution under a specific parameter configuration p_i. The parameter values p_i may be sampled from a predefined range or grid, covering the region of interest in the parameter space. All time series are assumed to have the same length L_{train}, ensuring uniformity in the training data structure.

For each parameter value p_i, the input sequence is formed by concatenating:

$$\{[\mathbf{v}_0; p_i]\}_{q=0}^{L_0} \bigoplus \{[\mathbf{v}_i(t); p_i]\}_{t=1}^{L_{\text{train}}-L_0}, \tag{F1}$$

where

(i) $\{[\mathbf{v}_0; p_i]\}_{q=0}^{L_0}$: a sequence in which the initial state $\mathbf{v}_0$ and the parameter p_i are concatenated (typically forming a vector of dimension $d + m$, with d the state dimension and m the parameter dimension), repeated over L_0 time steps for reservoir initialization.

(ii) $\{[\mathbf{v}_i(t); p_i]\}_{t=1}^{L_{\text{train}}-L_0}$: the dynamic trajectory of the system under parameter p_i, starting from $\mathbf{v}_0$.

This input structure ensures that the model is exposed to both the initial configuration and the parameter value throughout the input window, enabling it to learn parameter-dependent dynamics.

(2) *State matrix generation*. For each input sequence, the reservoir produces a sequence of hidden states, from which the segment corresponding to time steps $L_1 + 1$ to L_{train} is extracted to form:

$$\mathbf{X}_i = [\mathbf{x}_i(L_1 + 1), \ldots, \mathbf{x}_i(L_{\text{train}})], \tag{F2}$$

with $\mathbf{x}_i(t) \in \mathbb{R}^N$ denoting the reservoir state at time t. The overall state matrix is obtained by stacking all individual state matrices:

$$\mathbf{X} = [\mathbf{X}_1; \ldots; \mathbf{X}_g], \tag{F3}$$

resulting in a matrix of size $[g \times (L_{\text{train}} - L_1)] \times N$, which is used to train the output weight matrix $\mathbf{W}_{\text{out}}$ via the method referenced in the main text [e.g., Eq. (4)].

(3) *Prediction procedure*. For a new parameter value p_j, the input is initialized as $[\mathbf{v}_0; p_j]$ for the first L_0 time steps. The reservoir states $\mathbf{x}_j(q)$ ($q = 1, \ldots, L_0$) are computed using the model dynamics. The first predicted state is given by

$$\mathbf{v}_j(L_0 + 1) = \mathbf{W}_{\text{out}}\mathbf{x}_j(L_0). \tag{F4}$$

For subsequent time steps ($t > L_0$), the prediction is performed using the previously predicted value as input, while the parameter setting is fixed only for the initial L_0 steps. This approach enables the model to produce iterative, autoregressive predictions across the desired time horizon, capturing the system's response to the specified parameter value without requiring retraining.

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

[2] K. Champion, B. Lusch, J. N. Kutz, and S. L. Brunton, Data-driven discovery of coordinates and governing equations, Proc. Natl. Acad. Sci. USA **116**, 22445 (2019).

[3] M. I. Jordan and T. M. Mitchell, Machine learning: Trends, perspectives, and prospects, Science **349**, 255 (2015).

[4] P. Domingos, A few useful things to know about machine learning, Commun. ACM **55**, 78 (2012).

[5] Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, Nature (London) **521**, 436 (2015).

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM **60**, 84 (2017).

[7] P. M. Nadkarni, L. Ohno-Machado, and W. W. Chapman, Natural language processing: An introduction, J. Am. Med. Inf. Assoc. **18**, 544 (2011).

[8] S.-X. Lun, X.-S. Yao, H.-Y. Qi, and H.-F. Hu, A novel model of leaky integrator echo state network for time-series prediction, Neurocomputing **159**, 58 (2015).

[9] R. P. Masini, M. C. Medeiros, and E. F. Mendes, Machine learning advances for time series forecasting, J. Econ. Surv. **37**, 76 (2023).

[10] H. Jaeger, The "echo state" approach to analysing and training recurrent neural networks-with an erratum note, GMD Technical Report No. 148, German National Research Center for Information Technology, Bonn, Germany, 2001.

[11] F. M. Shiri, T. Perumal, N. Mustapha, and R. Mohamed, A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU, J. Artif. Intell. **6**, 301 (2023).

[12] L. Sun, B. Liu, J. Tao, and Z. Lian, Multimodal cross-and self-attention network for speech emotion recognition, in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, Piscataway, NJ, 2021), pp. 4275–4279.

[13] H. Ibrahim, C. K. Loo, and F. Alnajjar, Speech emotion recognition by late fusion for bidirectional reservoir computing with random projection, IEEE Access **9**, 122855 (2021).

[14] H. Ibrahim, C. K. Loo, and F. Alnajjar, Bidirectional parallel echo state network for speech emotion recognition, Neural Comput. Appl. **34**, 17581 (2022).

[15] F. M. Bianchi, S. Scardapane, S. Løkse, and R. Jenssen, Reservoir computing approaches for representation and classification of multivariate time series, IEEE Trans. Neural Netw. Learn. Syst. **32**, 2169 (2021).

[16] Z. Huang, C. Yang, X. Chen, X. Zhou, G. Chen, T. Huang, and W. Gui, Functional deep echo state network improved by a bi-level optimization approach for multivariate time series classification, Appl. Soft Comput. **106**, 107314 (2021).

[17] H. Peng, X. Xiong, M. Wu, J. Wang, Q. Yang, D. Orellana-Martín, and M. J. Pérez-Jiménez, Reservoir computing models based on spiking neural p systems for time series classification, Neural Netw. **169**, 274 (2024).

[18] J. Sun, L. Li, and H. Peng, An image classification method based on echo state network, in *2021 International Conference on Neuromorphic Computing (ICNC)* (IEEE, Piscataway, NJ, 2021), pp. 165–170.

[19] S. D. Gardner, M. R. Haider, L. Moradi, and V. Vantsevich, A modified echo state network for time independent image classification, in *2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)* (IEEE, Piscataway, NJ, 2021), pp. 255–258.

[20] H. Duan and X. Wang, Echo state networks with orthogonal pigeon-inspired optimization for image restoration, IEEE Trans. Neural Netw. Learn. Syst. **27**, 2413 (2016).

[21] J. Jeon, P. Kim, B. Jang, and Y. Kim, PDE-guided reservoir computing for image denoising with small data, Chaos **31**, 073103 (2021).

[22] A. Souahlia, A. Belatreche, A. Benyettou, and K. Curran, An experimental evaluation of echo state network for colour image segmentation, in *2016 International Joint Conference on Neural Networks (IJCNN)* (IEEE, Piscataway, NJ, 2016), pp. 1143–1150.

[23] A. Souahlia, A. Belatreche, A. Benyettou, Z. Ahmed-Foitih, E. Benkhelifa, and K. Curran, Echo state network-based feature extraction for efficient color image segmentation, Concurrency Comput. Part. Exper. **32**, e5719 (2020).

[24] H. Kim and J. Jeong, Decoding electroencephalographic signals for direction in brain-computer interface using echo state network and Gaussian readouts, Comput. Biol. Med. **110**, 254 (2019).

[25] S. Shahi, F. H. Fenton, and E. M. Cherry, Prediction of chaotic time series using recurrent neural networks and reservoir computing techniques: A comparative study, Mach. Learn. Appl. **8**, 100300 (2022).

[26] H. H. Ren, Y. L. Bai, M. H. Fan, L. Ding, X. X. Yue, and Q. H. Yu, Constructing polynomial libraries for reservoir computing in nonlinear dynamical system forecasting, Phys. Rev. E **109**, 024227 (2024).

[27] J. Liu, X. Xu, and E. Li, A minimum complexity interaction echo state network, Neural Comput. Appl. **36**, 4013 (2024).

[28] J. Liu, X. Xu, and E. Li, An echo state network with interacting reservoirs for modeling and analysis of nonlinear systems, Nonlin. Dyn. **112**, 8341 (2024).

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

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

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

[32] J. Liu, X. Xu, W. Guo, and E. Li, Learning hidden dynamics using echo state network, Nonlin. Dyn. **113**, 14181 (2025).

[33] F. Köster, K. Kanno, and A. Uchida, Attention-enhanced reservoir computing as a multiple-dynamical-system approximator, Phys. Rev. Appl. **24**, 054068 (2025).

[34] H. Jaeger, M. Lukoševičius, D. Popovici, and U. Siewert, Optimization and applications of echo state networks with leaky-integrator neurons, Neural Netw. **20**, 335 (2007).

[35] X. Na, W. Ren, and X. Xu, Hierarchical delay-memory echo state network: A model designed for multi-step chaotic time series prediction, Eng. Appl. Artif. Intell. **102**, 104229 (2021).

[36] D. W. Liedji, J. H. Talla Mbé, and G. Kenné, Chaos recognition using a single nonlinear node delay-based reservoir computer, Eur. Phys. J. B **95**, 18 (2022).

[37] X. Xu, J. Liu, and E. Li, Delayed self-feedback echo state network for long-term dynamics of hyperchaotic systems, Phys. Rev. E **109**, 064210 (2024).

[38] C. Gallicchio, A. Micheli, and L. Pedrelli, Deep reservoir computing: A critical experimental analysis, Neurocomputing **268**, 87 (2017).

[39] G. Wainrib and M. N. Galtier, A local echo state property through the largest Lyapunov exponent, Neural Netw. **76**, 39 (2016).

[40] H. Jaeger, Short term memory in echo state networks, GMD Technical Report No. 152, German National Research Center for Information Technology, Bonn, Germany, 2002.

[41] Z. Wang, I. Moroz, Z. Wei, and H. Ren, Dynamics at infinity and a Hopf bifurcation arising in a quadratic system with coexisting attractors, Pramana **90**, 12 (2018).

[42] G. Saxena, A. Prasad, and R. Ramaswamy, Amplitude death: The emergence of stationarity in coupled nonlinear systems, Phys. Rep. **521**, 205 (2012).

[43] A. Koseska, E. Volkov, and J. Kurths, Oscillation quenching mechanisms: Amplitude vs. oscillation death, Phys. Rep. **531**, 173 (2013).

[44] S. Panahi and Y.-C. Lai, Adaptable reservoir computing: A paradigm for model-free data-driven prediction of critical transitions in nonlinear dynamical systems, Chaos **34**, 051501 (2024).
[45] X. Xu and J. Liu, Stability, bifurcation and dynamics in a network with delays, Int. J. Bifurcation Chaos **34**, 2450025 (2024).
[46] R. Xiao, L.-W. Kong, Z.-K. Sun, and Y.-C. Lai, Predicting amplitude death with machine learning, Phys. Rev. E **104**, 014205 (2021).
[47] T. Z. Jiahao, M. A. Hsieh, and E. Forgoston, Knowledge-based learning of nonlinear dynamics and chaos, Chaos **31**, 111101 (2021).
[48] J. Snoek, H. Larochelle, and R. P. Adams, Practical Bayesian optimization of machine learning algorithms, in *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2* (Curran Associates, Inc., Red Hook, NY, 2016), pp. 2951–2959.
[49] D. Wang, D. Tan, and L. Liu, Particle swarm optimization algorithm: An overview, Soft Comput. **22**, 387 (2018).
[50] Y. Kuramoto and T. Tsuzuki, Persistent propagation of concentration waves in dissipative media far from thermal equilibrium, Prog. Theor. Phys. **55**, 356 (1976).
[51] D. M. Michelson and G. I. Sivashinsky, Nonlinear analysis of hydrodynamic instability in laminar flames— II. Numerical experiments, Acta Astronaut. **4**, 1207 (1977).
[52] D. Wilczak and P. Zgliczyński, A geometric method for infinite-dimensional chaos: Symbolic dynamics for the Kuramoto-Sivashinsky PDE on the line, J. Diff. Eq. **269**, 8509 (2020).
[53] P. R. Vlachas, W. Byeon, Z. Y. Wan, T. P. Sapsis, and P. Koumoutsakos, Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks, Proc. R. Soc. A **474**, 20170844 (2018).
[54] COMSOL multiphysics® v. 6.1, https://www.comsol.com.
[55] Z. Li, M. Liu-Schiaffini, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, Learning chaotic dynamics in dissipative systems, Adv. Neural Inf. Proc. Syst. **35**, 16768 (2022).
[56] T. Wang, G. Liu, E. Li, and X. Xu, Adaptive deep physics-informed neural network with dual-nested activation for solving complex partial differential equations, Comput. Methods Appl. Mech. Eng. **444**, 118125 (2025).
[57] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, Neural ordinary differential equations, Adv. Neural Inf. Proc. Syst. **31**, 6571 (2018).
[58] Z. Wang, D. Veeman, M. Zhang, H. Natiq, R. Yang, and I. Hussain, A symmetric oscillator with multi-stability and chaotic dynamics: Bifurcations, circuit implementation, and impulsive control, Eur. Phys. J.: Spec. Top. **231**, 2153 (2022).
[59] L.-W. Kong, H.-W. Fan, C. Grebogi, and Y.-C. Lai, Machine learning prediction of critical transition and system collapse, Phys. Rev. Res. **3**, 013090 (2021).
[60] Z. Xu, L. Li, Z. Li, and G. Zhou, Modulation instability and solitons on a cw background in an optical fiber with higher-order effects, Phys. Rev. E **67**, 026603 (2003).
[61] Y. Chen, G. Jiang, S. Chen, Z. Guo, X. Yu, C. Zhao, H. Zhang, Q. Bao, S. Wen, D. Tang, *et al.*, Mechanically exfoliated black phosphorus as a new saturable absorber for both q-switching and mode-locking laser operation, Opt. Express **23**, 12823 (2015).
[62] T. Fawcett, An introduction to ROC analysis, Pattern Recognit. Lett. **27**, 861 (2006).
[63] J. Pathak, Z. Lu, B. R. Hunt, M. Girvan, and E. Ott, Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data, Chaos **27**, 121102 (2017).
[64] J. Herteux and C. Räth, Breaking symmetries of the reservoir equations in echo state networks, Chaos **30**, 123142 (2020).
[65] A. Ohkubo and M. Inubushi, Reservoir computing with generalized readout based on generalized synchronization, Sci. Rep. **14**, 30918 (2024).
[66] D. Prosperino, H. Ma, and C. Rth, Tailored minimal reservoir computing: On the bidirectional connection between nonlinearities in the model and in data, Chaos **35**, 093105 (2025).
[67] K. Nakai and Y. Saiki, Machine-learning construction of a model for a macroscopic fluid variable using the delay-coordinate of a scalar observable, Discrete Contin. Dyn. Syst. S **14**, 1079 (2021).
[68] S. Iacob, M. Freiberger, and J. Dambre, Distance-based delays in echo state networks, in *International Conference on Intelligent Data Engineering and Automated Learning* (Springer, Cham, 2022), Vol. 13756, pp. 211–222.