2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

Memory Regressor Extended Echo State Networks for Nonlinear Dynamics Modeling

Kai Hu[2], Qian Wang[1], Tian Shi[3], Kohei Nakajima[4], and Yongping Pan[1]

Abstract— **Echo state network (ESN) implements an alternative paradigm called reservoir computing to train recurrent neural networks (RNNs), where internal weights are randomly generated and kept fixed, and only readout weights need to be trained, which greatly reduces the training complexity of RNNs. ESN not only facilitates the practical implementation of RNNs but also shows superior performance over fully trained RNNs across a range of applications. However, the conventional ESN suffers from the drawbacks of stringent conditions for weight convergence and slow convergence speed. This paper proposes a memory regressor extended learning method to update the readout weights of ESNs. By constructing and incorporating a generalized prediction error based on regressor extension and filtering, the capacity of ESN to utilize historical data can be greatly improved. In the discrete-time domain, it is proven that exponential convergence of readout weights is achieved under a condition termed interval excitation that is strictly weaker than the classical condition of persistent excitation. Simulation results on modeling a 10th-order nonlinear autoregressive moving-average (NARMA) system have revealed that the proposed approach accelerates weight convergence speed almost ten times higher compared to the conventional ESN.**

I. Introduction

Complex nonlinear systems, characterized by their typically chaotic and unpredictable dynamics, exhibit multiple interacting spatiotemporal scales that pose challenges to classical numerical methods in terms of prediction and control [1]. Recurrent neural networks (RNNs) are a type of neural network architecture that is well-suited for modeling and analyzing spatiotemporal data. RNNs can capture temporal dependencies and can be extended to incorporate spatial information, making them effective in handling complex spatiotemporal patterns [2]. By utilizing recurrent connections, RNNs can process sequential data and retain information from previous time steps, allowing them to capture the temporal dynamics of a nonlinear system. This makes them particularly useful for tasks such as time series prediction [3], speech recognition [4], natural language processing [5], and robotics [6], where the order and context of the data are important.

This work was supported in part by the Fundamental Research Funds for the Central Universities, Sun Yat-sen University, China, under Grant No. 23lgzy004. (*Corresponding author: Yongping Pan*).

[1]Q. Wang and Y. Pan are with the School of Advanced Manufacturing, Sun Yat-sen University, Shenzhen 518100, China wangqq666@mail2.sysu.edu.cn; panyongp@mail.sysu.edu.cn

[2]K. Hu is with the School of Artificial Intelligence, Sun Yat-sen University, Zhuhai 519000, China hukai8@mail2.sysu.edu.cn

[3]T. Shi is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China shit23@mail2.sysu.edu.cn

[4]K. Nakajima is with the Graduate School of Information Science and Technology, University of Tokyo, Tokyo 113-8656, Japan k-nakajima@isi.imi.i.u-tokyo.ac.jp

The two mainstream training methods for RNNs currently are backpropagation and reservoir computing (RC). Some of the earlier approaches explore the potential of RNNs in capturing temporal dynamics through the utilization of low-dimensional RNNs [7], [8], but its effectiveness is limited by the fact that gradients may vanish or explode during the learning process, which can impede the network ability to learn and predict. Some work [9], [10] extend long short-term memory to model and predict nonlinear systems, which partially alleviates the issues. However, to the best of our knowledge, these approaches have been used only on low-dimensional and low-complexity systems. Several recent pieces of literature demonstrate promising results for high-complexity systems [11], [12]. Nevertheless, backpropagation and other applied techniques can significantly increase the training complexity of RNNs, leading to slower convergence speed.

RC offers a powerful approach to train RNNs, providing insights into complex systems that exhibit high-dimensional nonlinear dynamics. This study primarily concentrates on echo state networks (ESNs) in the RC framework. ESNs typically consist of three layers, i.e., input, reservoir, and output layers, where the reservoir layer is the core component that maps input signals into a higher dimensional space through nonlinear transformations. This high-dimensional representation effectively captures the spatiotemporal dynamic characteristics of complex nonlinear systems. It should be noted that the reservoir layer is not a conventional hidden layer in artificial neural networks, as the parameters of the reservoir are randomly initialized and remain unchanged throughout the entire training process. Only the readout layer needs to be trained to enable the ESN model to perform specific tasks. This structural design gives ESNs the advantages of fast training and lower computational complexity [13]. The universal approximation properties of ESNs have been shown in [14], [15]. Zhang et al. [16] provided an analytical explanation for the observation that echo states are obtained in practice when the spectral radius of the reservoir weight matrix is smaller than one.

The utilization of the pseudo-inverse is a common method for training ESNs. Nevertheless, this method can potentially result in ill-posed problems and large output weights, which may weaken the generalization capability of ESNs. As a result, the behavior of the trained ESN may differ significantly when tested with data that deviates slightly from training data [17]. Truncated singular value decomposition [18] and Tikhonov-type regularization [19] have been proposed as regularization techniques for learning readout weights. Training algorithms based on gradient descent [1], [20] and recursive least squares (RLS) [21], [22] have demonstrated good results in modeling

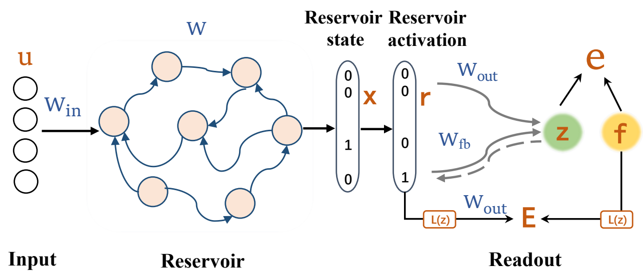


Fig. 1. The structure and computational process of MRE-ESN.

nonlinear systems. However, these methods do not further process prediction errors, resulting in ESNs not fully harnessing the historical information across time scales provided by recurrent connections in the reservoir. Some research has made improvements to RLS-based FORCE learning using memory regressor extension (MRE), but none of these studies provide proof regarding parameter convergence [23]–[25].

The convergence of parameter estimation is guaranteed by a widely recognized condition termed persistent excitation (PE) [26]. However, PE requires sufficient training data containing rich spectrum information all the time, which is difficult to satisfy in practice. Even if the PE condition is fulfilled, the speed of parameter convergence in gradient-based estimation largely depends on the excitation strength, leading to a slow learning pace [27]. This paper presents an MRE-enhanced learning method for training readout weights of the ESN. A generalized prediction error is defined to retain past excitation information by using regressor extension and filtering, and both the general and generalized prediction errors act synchronously on the weight-updating process. We prove in the discrete-time domain that our method can converge under a relaxed interval excitation (IE) condition that greatly relaxes PE. Compared with gradient-based estimation, this approach offers improved convergence during training, manifested as faster convergence speed and relaxed excitation requirement.

Notations: $\mathbb{R}$, $\mathbb{R}^+$, $\mathbb{R}^n$, and $\mathbb{R}^{m\times n}$ denote the spaces of real numbers, positive real numbers, real n-vectors, and real $m\times n$-matrices, respectively, $\mathbb{N}^+$ denotes the set of positive natural numbers, $\|\boldsymbol{x}\|$ denotes the Euclidean norm of $\boldsymbol{x}\in\mathbb{R}^n$, $\Omega_c := \{\boldsymbol{x}| \|\boldsymbol{x}\| \le c\}$ denotes a ball of radius $c\in\mathbb{R}^+$, I refers to an identity matrix and x_i refers to the ith element of $\boldsymbol{x}$, L_∞ refers to the space of bounded signals, where i, m, $n\in\mathbb{N}^+$.

II. The Proposed Method

A. Echo State Network

A block diagram of the proposed MRE-ESN is shown in Fig. 1, where the reservoir consists of a collection of recurrently connected neurons, and the connectivity structure is usually random and sparse. The overall dynamics of the reservoir are driven by current inputs and past information, where dynamics containing high-dimensional information are parsed out at the readout layer. It is worth noting that the system modeling problem discussed in this study is in a discrete context. At the time step k, the reservoir state $\boldsymbol{x}(k)\in\mathbb{R}^N$ is determined by a combination of input stimuli, internal neuronal activity, and feedback loop, which can be expressed by

$$\boldsymbol{x}(k) = \boldsymbol{\phi}\big(W_{\text{in}}\boldsymbol{u}(k) + W\boldsymbol{x}(k-1) + W_{\text{fb}}z(k-1)\big) \quad (1)$$

where $\boldsymbol{\phi}(k) = \tanh(\boldsymbol{x}(k))$ is an activation function, $W_{\text{in}}\in\mathbb{R}^{N\times P}$ is a connection matrix between the input and reservoir, $\boldsymbol{u}(k)\in\mathbb{R}^P$ is the input, $W\in\mathbb{R}^{N\times N}$ is a connection matrix within the reservoir, and $W_{\text{fb}}\in\mathbb{R}^N$ is a connection vector from the readout layer to reservoir. According to the concepts of ESNs, these matrices are randomly generated and remain unchanged during operations.

The sole trainable weight $\hat{W}_{\text{out}}\in\Omega_{c_{\text{w}}}\in\mathbb{R}^N$ represents the connection vector between the reservoir and the readout layer, where $c_{\text{w}}\in\mathbb{R}^+$ is a certain constant. The reservoir activation $\boldsymbol{r}(k)\in\mathbb{R}^N$ is given by the weighted sum of its historical value and current reservoir state as follows:

$$\boldsymbol{r}(k) = (1-\alpha)\boldsymbol{r}(k-1) + \alpha\boldsymbol{x}(k) \quad (2)$$

where $\alpha\in(0,1]$ is a constant leaky rate used to control the learning process of reservoir activation. The network output $z(k)\in\mathbb{R}$ is represented as follows:

$$z(k) = \boldsymbol{r}^T(k)\hat{W}_{\text{out}}(k-1). \quad (3)$$

At each time step k, the sampled $z(k)$ utilizes the readout weight $\hat{W}_{\text{out}}(k-1)$ from the previous $k-1$ as $\hat{W}_{\text{out}}(k)$ has not yet been updated at this point.

Consider a nonlinear dynamics problem that aims to enable $z(k)$ to replicate the target dynamics

$$f(k) = \boldsymbol{r}^T(k)W_{\text{out}} \quad (4)$$

where $W_{\text{out}}\in\mathbb{R}^N$ is an ideal weight. Let $\tilde{W}_{\text{out}} := W_{\text{out}} - \hat{W}_{\text{out}}\in\mathbb{R}^N$ be a weight estimation error. At the time step k, a prediction error $e(k)\in\mathbb{R}$ is given as follows:

$$e(k) = f(k) - \boldsymbol{r}^T(k)\hat{W}_{\text{out}}(k-1). \quad (5)$$

After obtaining (5), updating $\hat{W}_{\text{out}}$ using the gradient descent approach is straightforward. However, this approach has certain limitations. First, the convergence of $\hat{W}_{\text{out}}$ depends on the PE condition for training data and network activations. In addition, the convergence speed is relatively slow.

B. MRE-Enhanced Method

To solve the aforementioned limitations, an MRE-enhanced algorithm is proposed to train ESNs as follows. We construct a new extended regression equation via linear filtering operators with memory. Let $L(z) := \frac{\lambda}{1-(1-\lambda)z^{-1}}$ denote a discrete-time stable filter. In order to facilitate the theoretical analysis, we provide the following definitions.

Definition 1: A bounded signal $\Phi(k)\in\mathbb{R}^N$ is of PE if there exists $\sigma\in\mathbb{R}^+$ such that $L(z)[\Phi(k)\Phi^T(k)] \ge \sigma I, \forall k\ge 0$.

Definition 2: A bounded signal $\Phi(k)\in\mathbb{R}^N$ is of IE if there exist $k_c,\sigma\in\mathbb{R}^+$ such that $L(z)[\Phi(k)\Phi^T(k)] \ge \sigma I, k = k_c$.

Define an excitation matrix to be $Q(k) := L(z)[\boldsymbol{r}(k)\boldsymbol{r}^T(k)]$. Multiplying each side of (4) by $\boldsymbol{r}(k)$ and applying $L(z)$ and $Q(k)$, one gets an extended prediction equation

$$Y(k) := L(z)[\boldsymbol{r}(k)f(k)]. \quad (6)$$

Based on (6), define a generalized prediction error

$$E(k) := \begin{cases} Y(k) - Q(k)\hat{W}_{\rm out}(k),\ k < k_{\rm c} \\ Y(k_{\rm e}) - Q(k_{\rm e})\hat{W}_{\rm out}(k),\ k \geq k_{\rm c} \end{cases} \tag{7}$$

with $k_{\rm e} := \arg\max_{\tau \in [k_{\rm c}, k]\cap \mathbb{N}} \sigma_{\min}(Q(\tau))$. The current maximal exciting strength is given by $\sigma_{\rm c}(k) := \sigma_{\min}(Q(k_{\rm e})),\ k \geq k_{\rm c}$. The readout weight $\hat{W}_{\rm out}$ is updated by

$$\begin{aligned}\hat{W}_{\rm out}(k+1) =& \hat{W}_{\rm out}(k) + \frac{1}{2}R^{-1}(k)\boldsymbol{r}(k)e(k) \\ &+ \frac{1}{2}\Lambda^{-1}(k)E(k)\end{aligned} \tag{8}$$

where $R(k) := I + \boldsymbol{r}(k)\boldsymbol{r}^T(k)$ and

$$\Lambda(k) := \begin{cases} I + Q(k),\ k < k_{\rm c} \\ I + Q(k_{\rm e}),\ k \geq k_{\rm c} \end{cases}$$

are matrix normalization gains.

By incorporating the original prediction error and the generalized prediction error, as shown in (8), our learning algorithm has the superiority of utilizing the past activation history of the extended regressor. Theoretically, we will demonstrate that the proposed method can converge under the IE condition, which is less stringent than the PE condition, as follows.

Theorem 1: Let $[0, k_{\rm f})$ with $k_{\rm f} \in \mathbb{R}^+$ denote the maximal iteration set of the existence of solutions of the system (2). For any given initial readout weight $\hat{W}_{\rm out}(0) \in \Omega_{c_{\rm w}}$, the MRE-enhanced parameter update law in (8) guarantees that:

1) $\hat{W}_{\rm out}(k)$ and $e(k)$ are of $L_\infty, \forall k \leq k_{\rm f}$;
2) $\tilde{W}_{\rm out}(k) \to \mathbf{0}$ exponentially at $k \geq k_{\rm c}$ as $k \to \infty$, if the IE condition $Q(k_{\rm c}) \geq \sigma I$ in Definition 2 is satisfied for certain constants $k_{\rm c}, \sigma \in \mathbb{R}^+$.

Proof. Based on (8), $\tilde{W}_{\rm out}$ is represented by

$$\begin{aligned}\tilde{W}_{\rm out}(k+1) =& \hat{W}_{\rm out}(k+1) - \hat{W}_{\rm out}(k) \\ =& -\frac{1}{2}R^{-1}(k)\boldsymbol{r}(k)e(k) - \frac{1}{2}\Lambda^{-1}(k)E(k) \\ =& \Big[I - \frac{1}{2}R^{-1}(k)\boldsymbol{r}(k)\boldsymbol{r}^T(k) \\ &- \frac{1}{2}\Lambda^{-1}(k)Q(k)\Big]\tilde{W}_{\rm out}(k).\end{aligned} \tag{9}$$

Consider a Lyapunov function candidate

$$V(k) = \tilde{W}_{\rm out}^T(k)\tilde{W}_{\rm out}(k). \tag{10}$$

According to (9) and (10), one has

$$\begin{aligned}V(k+1) =& \tilde{W}_{\rm out}^T(k+1)\tilde{W}_{\rm out}(k+1) \\ =& \tilde{W}_{\rm out}^T(k)[I - \frac{1}{2}R^{-1}(k)\boldsymbol{r}(k)\boldsymbol{r}^T(k) \\ &- \frac{1}{2}\Lambda^{-1}(k)Q(k)]^2\tilde{W}_{\rm out}(k).\end{aligned} \tag{11}$$

Bringing $R^{-1}(k)$ and $\Lambda(k)$ into (11), one gives

$$0 < I - \frac{1}{2}R^{-1}(k)\boldsymbol{r}(k)\boldsymbol{r}^T(k) - \frac{1}{2}\Lambda^{-1}(k)Q(k) < I \tag{12}$$

to get $V(k+1) \leq V(k)$, which implies $\tilde{W}_{\rm out}(k), \hat{W}_{\rm out}(k) \in L_\infty$, and hence, $e(k) \in L_\infty$ on $k \in [0, k_{\rm f})$.

Next, consider the convergence problem under the IE condition for $k \geq k_{\rm c}$, i.e., there exist $k_{\rm c}, \sigma \in \mathbb{R}^+$ such that $Q(k_{\rm c}) \geq \sigma I$. It follows from (12) that

$$\begin{aligned}&\Big[I - \frac{R^{-1}(k)\boldsymbol{r}(k)\boldsymbol{r}^T(k)}{2} - \frac{(I + Q(k_{\rm e}))^{-1}Q(k_{\rm e})}{2}\Big]^2 \\ \leq& \Big[I - \frac{1}{2}(I + Q(k_{\rm e}))^{-1}Q(k_{\rm e})\Big]^2 \\ =& \Big[\big(I + Q(k_{\rm e})\big)^{-1}\big(I + Q(k_{\rm e}) - \frac{1}{2}Q(k_{\rm e})\big)\Big]^2 \\ =& \big(I + Q(k_{\rm e})\big)^{-2}\big(I + \frac{1}{2}Q(k_{\rm e})\big)^2 \\ =& \big(I + Q(k_{\rm e})\big)^{-2}\Big[\frac{1}{2}I + \frac{1}{2}\big(I + Q(k_{\rm e})\big)\Big]^2 \\ =& \frac{1}{4}[I + (I + Q(k_{\rm e}))^{-1}]^2.\end{aligned} \tag{13}$$

As $Q(k_{\rm e}) \geq Q(k_{\rm c}) \geq \sigma I$, one has

$$\begin{aligned}V(k+1) \leq& \frac{1}{4}\big[1 + (1+\sigma)^{-1}\big]^2\tilde{W}_{\rm out}^T(k)\tilde{W}_{\rm out}(k) \\ =& \frac{1}{4}\big[1 + (1+\sigma)^{-1}\big]^2 V(k)\end{aligned} \tag{14}$$

with $0 < \frac{1}{4}[1 + (1+\sigma)^{-1}]^2 < 1$, which implies that $\tilde{W}_{\rm out}(k) \to 0$ exponentially for $k \geq k_{\rm c}$. □

III. Numerical Verification

In this section, numerical verification is conducted to show the benefits of the proposed MRE-based learning method. The task is training the readout weight $\hat{W}_{\rm out}$ to make the ESN output $z(k)$ follow a nonlinear dynamic model. We validate the superiority of the proposed method on the 10th-order nonlinear autoregressive moving average (NARMA) system.

A. Simulation Settings

Simulations are carried out in the MATLAB software, where the average results of 20 runs are presented for each trial. We randomly initialize the input weight $W_{\rm in}$ and the feedback weight $W_{\rm fb}$ following uniform distributions with $W_{\rm in}, W_{\rm fb} \sim U(-1, 1)$. The reservoir connection weight W is scaled by a chaotic degree $g \in \mathbb{R}^+$ to make the reservoir chaotic [28]. The computational capabilities of ESNs may be enhanced if reservoir connections are at the edge of chaos [29].

The reservoir consists of $N = 600$ neurons. The hyperparameter $p \in (0, 1)$ determines the sparsity of interconnections between neurons within the reservoir. If the value is too small, it will lead to a less expressive network. If the value is too large, it will make recognizing network output patterns difficult. We employ grid search to determine the ideal connectivity sparsity p within the reservoir. The baseline is the original gradient descent method in [30], which is a common form of the gradient descent method:

$$\hat{W}_{\rm out}(k+1) = \hat{W}_{\rm out}(k) - \frac{\boldsymbol{r}(k)}{\gamma + |\boldsymbol{r}(k)|}e(k) \tag{15}$$

where $\gamma \in \mathbb{R}^+$ is the learning rate. This method solely relies on the original prediction error.

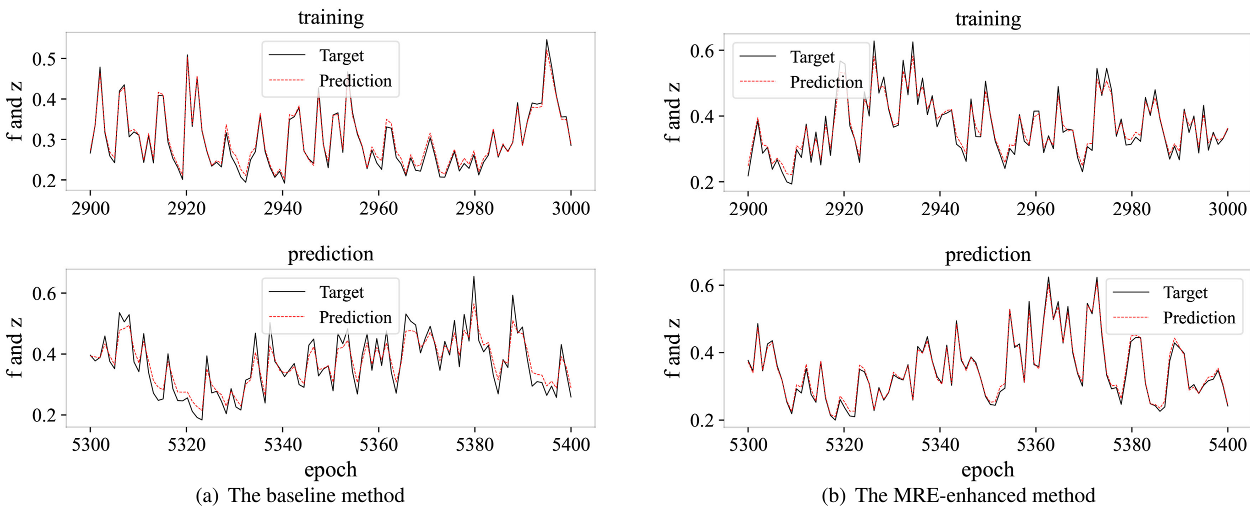


Fig. 2. Comparison of ESN fitting performance using baseline and the MRE-enhanced methods in the training and prediction phases. Top: Fitting curve of the last 100 training epochs. Bottom: Fitting curve of the last 100 testing epochs.

B. 10th-Order NARMA System

The 10th-order NARMA system, which exhibits complex behavior and nonlinearity, is often used as a benchmark in nonlinear system identification and control. Here, we use it to evaluate the performance of the proposed method in terms of learning and prediction. The system is described by

$$f(k+1) = 0.3f(k) + 0.05f(k)\sum_{i=0}^{9} f(k-i) + 1.5v(k-9)v(k) + 0.1 \tag{16}$$

where $v(k) \sim U(0, 0.5)$ and $f(0) = 0$.

For fair comparisons, we set most of the simulation parameters the same as in [31], and select the parameters p, g, and γ by a simple grid search. We generate 5,400 samples using (16), with 3,000 samples allocated for training and 2,400 samples for testing. After several rounds of the grid search, set $p = 0.12$, $g = 4.0$ for the proposed method and $p = 0.10$, $g = 3.5$, $\gamma = 2.0$ for the baseline. The readout weight $W_{\text{out}}(0)$ is initialized to $\mathbf{0}$. Similar to [31], normalized root mean square error (NRMSE) as follows:

$$\text{NRMSE} = \left(\sum_{i=1}^{n}(f_i - z_i)^2 / \sum_{i=1}^{n}(f_i - \overline{f})^2\right)^{\frac{1}{2}}$$

is applied as the metric to evaluate the performance.

C. Simulation Results

Simulation results of the baseline are shown in Fig. 2(a), where the ESN output $z(k)$ gradually matches the target $f(k)$ when the training starts. After training, the ESN approximates the target system with low accuracy. Simulation results of the proposed method are shown in Fig. 2(b). For better visualization, we only present the last 100 epochs of the training and prediction phases. The ESN output $z(k)$ rapidly approaches the target $f(k)$, resulting in an overall improved fitting performance. After training, the learned ESN keeps precisely generating the target output on its own.

Simulation results about convergence speed are shown in Fig. 3. For the baseline, the readout weight norm $\|\hat{W}_{\text{out}}\|$ does not converge to a constant after learning, and the NRMSE continues exhibiting a decreasing trend even after the final 3,000 epochs of training. For the proposed method, $\|\hat{W}_{\text{out}}\|$ converges to a constant after about 1,500 epochs. In the later stages of training, the NRMSE remains relatively stable. To further illustrate the convergence of the readout weight $\hat{W}_{\text{out}}$ in individual neurons, we randomly select 8 neurons to observe their evolution. Fig. 4(a) illustrates that during the entire training process of the baseline, the 8 elements of $\hat{W}_{\text{out}}$ change rapidly in the early stages and then gradually stabilize. Nevertheless, they do not converge to constants. Fig. 4(b) indicates that the 8 elements of $\hat{W}_{\text{out}}$ essentially stabilize and converge after the 1,500th epoch for the proposed method.

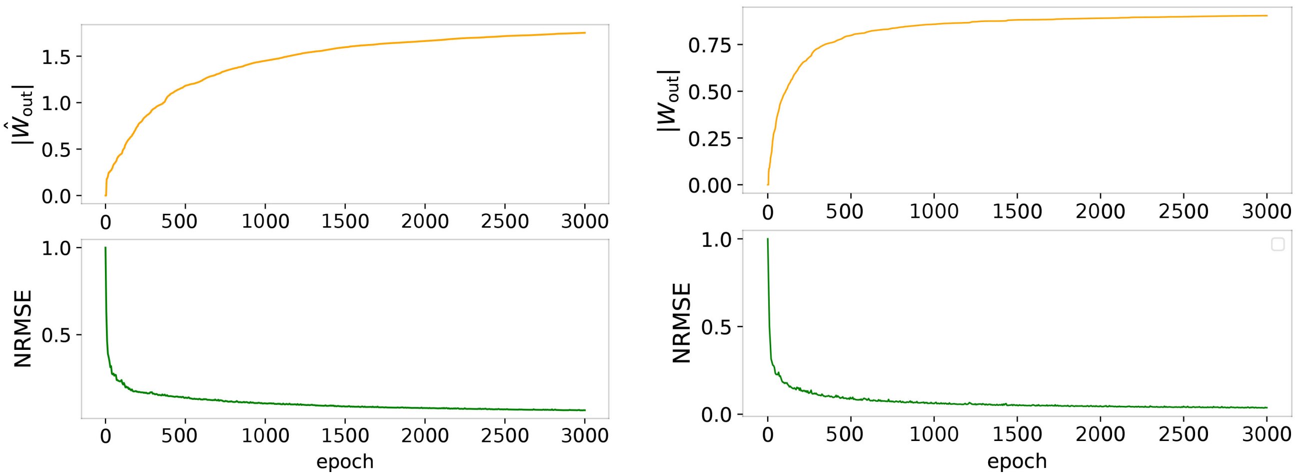
It is evident that the proposed method exhibits significantly faster convergence compared to the baseline without the generalized prediction error. With additional training, the baseline is expected to reach a state of basic convergence around the 14,000th epoch. The specific comparison results are displayed in Table I. The proposed method not only outperforms the baseline in modeling accuracy but also achieves a convergence speed almost 10 times faster than the baseline.

IV. Conclusions

This paper has proposed an MRE learning method for ESNs and has shown exceptional modeling performance for nonlinear dynamical systems. In particular, the proposed method can achieve parameter convergence under the IE condition that

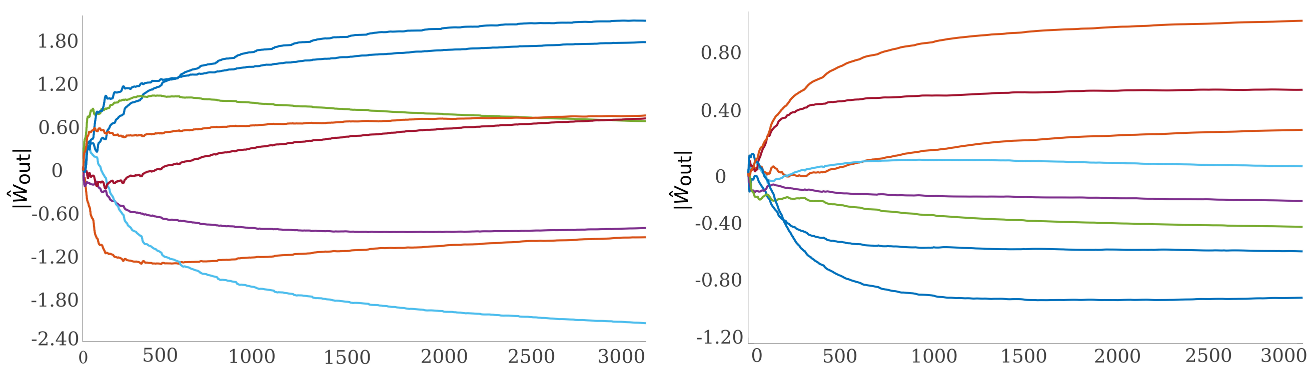
TABLE I
Simulation Comparisons on Nonlinear Dynamics Modelling

Methods	RNMSE	Convergence Time
Gradient descent method in [30]	6.27e-04	158.60s
The proposed method	**8.75e-05**	**16.56s**



(a) The baseline method (b) The MRE-enhanced method

Fig. 3. Comparison of ESN training performance using baseline and the MRE-enhanced methods. Top: The norm change of the readout weight W_{out}. Bottom: The variation in performance (NRMSE) of nonlinear dynamics modeling.



(a) The baseline method (b) The MRE-enhanced method

Fig. 4. Comparison of the evolution in the readout weight W_{out} for eight randomly selected readout neurons using baseline and MRE methods. The horizontal axis of the coordinate system refers to the training time step.

relaxes the strict PE condition. In simulation validation, the proposed method unequivocally exhibits superior performance in modeling nonlinear dynamical systems and achieves almost 10 times faster convergence than the conventional gradient descent method on the 10th NARMA nonlinear system.

References

[1] P. R. Vlachas, J. Pathak, B. R. Hunt, T. P. Sapsis, M. Girvan, E. Ott, and P. Koumoutsakos, "Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics," *Neural Netw.*, vol. 126, pp. 191–217, 2020.

[2] X. Shu, L. Zhang, G.-J. Qi, W. Liu, and J. Tang, "Spatiotemporal co-attention recurrent neural networks for human-skeleton motion prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3300–3315, 2022.

[3] K. Bandara, C. Bergmeir, and H. Hewamalage, "LSTM-MSNet: Leveraging forecasts on sets of related time series with multiple seasonal patterns," *IEEE Trans. Neural. Netw. Learn. Syst.*, vol. 32, no. 4, pp. 1586–1599, 2021.

[4] Y. Lin, D. Guo, J. Zhang, Z. Chen, and B. Yang, "A unified framework for multilingual speech recognition in air traffic control systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 8, pp. 3608–3620, 2021.

[5] R. Gurunath, A. H. Alahmadi, D. Samanta, M. Z. Khan, and A. Alahmadi, "A novel approach for linguistic steganography evaluation based on artificial neural networks," *IEEE Access*, vol. 9, pp. 120 869–120 879, 2021.

[6] T. G. Thuruthel, B. Shih, C. Laschi, and M. T. Tolley, "Soft robot perception using embedded soft sensors and recurrent neural networks," *Sci. Robot.*, vol. 4, no. 26, 2019.

[7] J. L. Elman, "Finding structure in time," *Cognit. sci.*, vol. 14, no. 2, pp. 179–211, 1990.

[8] W. E. Faller and S. J. Schreck, "Unsteady fluid mechanics applications of neural networks," *J. Aircr.*, vol. 34, no. 1, pp. 48–55, 1997.

[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] F. A. Gers, D. Eck, and J. Schmidhuber, "Applying LSTM to time series predictable through time-window approaches," in *Proc. Int. Conf. Artif. Neural netw.*, London, UK, 2001, pp. 669–676.

[11] 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: Math. Phys. Eng. Sci.*, vol. 474, no. 2213, p. 20170844, 2018.

[12] T. Simpson, N. Dervilis, P. Couturier, N. Maljaars, and E. Chatzi, "Reduced order modeling of non-linear monopile dynamics via an AE-LSTM scheme," *Front. Energy Res.*, vol. 11, p. 1128201, 2023.

[13] I. B. Yildiz, H. Jaeger, and S. J. Kiebel, "Re-visiting the echo state property," *Neural Netw.*, vol. 35, pp. 1–9, 2012.

[14] L. Grigoryeva and J.-P. Ortega, "Echo state networks are universal," *Neural Netw.*, vol. 108, pp. 495–508, 2018.

[15] L. Gonon and J.-P. Ortega, "Reservoir computing universality with stochastic inputs," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 1, pp. 100–112, 2020.

[16] B. Zhang, D. J. Miller, and Y. Wang, "Nonlinear system modeling with random matrices: Echo state networks revisited," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 1, pp. 175–182, 2012.

[17] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *Science*, vol. 304, no. 5667, pp. 78–80, 2004.

[18] Per, Christian, and Hansen, "The truncated SVD as a method for regularization," *BIT Numerical Math.*, vol. 27, no. 4, pp. 534–553, 1987.

[19] J. Colinas, Y. Goussard, and J.-J. Laurin, "Application of the tikhonov regularization technique to the equivalent magnetic currents near-field technique," *IEEE Trans. Antennas Propag.*, vol. 52, no. 11, pp. 3122–3132, 2004.

[20] M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Comput. Sci. Rev.*, vol. 3, no. 3, pp. 127–149, 2009.

[21] D. Sussillo and L. F. Abbott, "Generating coherent patterns of activity from chaotic neural networks," *Neuron*, vol. 63, no. 4, pp. 544–557, 2009.

[22] C. Zhang, Q. Song, H. Zhou, Y. Ou, H. Deng, and L. T. Yang, "Revisiting recursive least squares for training deep neural networks," *arXiv preprint arXiv:2109.03220*, 2021.

[23] R. Wu, Z. Li, and Y. Pan, "Adaptive echo state network robot control with guaranteed parameter convergence," in *Proc. Int. Robot. Appli.*, Yantai, China, 2021, pp. 587–595.

[24] R. Wu, K. Nakajima, and Y. Pan, "Performance improvement of FORCE learning for chaotic echo state networks," in *Proc. Int. Conf. Neural Inf. Process.*, Bali, Indonesia, 2021, pp. 262–272.

[25] Y. Li, K. Hu, K. Nakajima, and Y. Pan, "Composite FORCE learning of chaotic echo state networks for time-series prediction," in *Proc. Chinese Control Conf.*, Hefei, China, 2022, pp. 7355–7360.

[26] S. Sastry and M. Bodson, "Adaptive control: Stability, convergence and robustness," Englewood Cliffs, NJ, USA, 1989.

[27] Y. Pan, T. Sun, Y. Liu, and H. Yu, "Composite learning from adaptive backstepping neural network control," *Neural Netw.*, vol. 95, pp. 134–142, 2017.

[28] H. Jaeger, "The "echo state" approach to analysing and training recurrent neural networks," *GMD German Nat. Res. Inst. Comput. Sci., Tech. Rep.*, vol. 148, no. 34, p. 13, Jan. 2001.

[29] J. Boedecker, O. Obst, J. T. Lizier, N. M. Mayer, and M. Asada, "Information processing in echo state networks at the edge of chaos," *Theory Biosci.*, vol. 131, pp. 205–213, 2012.

[30] R. Ortega, S. Aranovskiy, A. A. Pyrkin, A. Astolfi, and A. A. Bobtsov, "New results on parameter estimation via dynamic regressor extension and mixing: Continuous and discrete-time cases," *IEEE Trans. Autom. Control*, vol. 66, no. 5, pp. 2265–2272, 2021.

[31] J. Qiao, F. Li, H. Han, and W. Li, "Growing echo-state network with multiple subreservoirs," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 391–404, 2016.