Neurocomputing 351 (2019) 111–122

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Echo state networks regulated by local intrinsic plasticity rules for regression

Xinjie Wang[a,b], Yaochu Jin[b,c,*], Kuangrong Hao[a,b]

[a] Engineering Research Center of Digitized Textile and Apparel Technology, Ministry of Education, Donghua University, Shanghai 201620, PR China
[b] College of Information Sciences and Technology, Donghua University, Shanghai 201620, PR China
[c] Department of Computer Science, University of Surrey, Guildford, GU2 7XH, UK

ARTICLE INFO

Article history:
Received 16 September 2018
Revised 4 February 2019
Accepted 15 March 2019
Available online 18 April 2019

Communicated by Yang Tang

Keywords:
Echo state networks
Local intrinsic plasticity
Covariance matrix adaptation evolution strategy
Regression problems

ABSTRACT

Intrinsic plasticity, as a biologically inspired unsupervised learning rule, is used for adapting the intrinsic excitability of the reservoir neurons. Existing intrinsic plasticity rules can only select a set of fixed rule parameters for the whole reservoir neurons, which affects the learning performance due to the lack of flexibility in providing intrinsic plasticity. In this paper, we present an echo state network (ESN) with local intrinsic plasticity rule built by different reservoir neurons which can adopt the intrinsic plasticity rule with different rule parameters to adjust its intrinsic excitability. And the covariance matrix adaptation evolution strategy is used to search and select the rule parameters corresponding to different reservoir neurons. Compared with several state-of-the-art ESN models and an ESN with the global plasticity rule, the proposed local intrinsic plasticity rule is able to achieve much better performance in some benchmark prediction tasks.

© 2019 Published by Elsevier B.V.

1. Introduction

Echo state networks are a new type of recurrent neural networks that have been widely used for time series prediction [1,2] and pattern recognition [3,4]. A basic ESN model can be regarded as a three-layer neural network, with a fully connected input layer, a large-scale sparse recurrent connected hidden layer (i.e., reservoir) and a simple trainable linear output layer [5]. The original ESN only trains the connection weights of the output layer, which overcomes some inherent weaknesses in recurrent neural networks, such as high computational complexity and inefficiency in training.

The core of the ESN is a large-scale sparse recurrent connected hidden layer (usually known as the reservoir), which is used as an intermediary for information storage and processing. However, the fixed connection weights of the reservoir also limit the learning performance of the ESN. Therefore, much research has been dedicated to the optimization of the reservoir. Rodan and Tino [6] proposed a deterministically constructed simple cycle reservoir (SCR) to eliminate the randomness of the reservoir structure. It was shown that SCR can obtain comparable performance to standard ESN while minimizing the complexity of the reservoir topology. In addition, small world effect [7] and scale-free property [8] can also be used to build the reservoir topology. Moreover, the performance of the ESN can be improved by modifying the state update equation of reservoir. Based on the standard ESN, Jaeger [9] presented a leaky integrator ESN (LESN) that can better adapt to the network input signal and further increase the short-term memory capacity of the reservoir. To deal with the time warping problem in dynamic pattern recognition, a time-warping invariant ESN is proposed by modifying the state update equation of reservoir [10]. Qiao et al. [11] presented a growing ESN (GESN) whose reservoir is composed of a few independent sub-reservoirs, which can overcome the coupling effects between the neurons within a single reservoir. Neuroevolution has been shown to be effective in the optimization of neural networks [12–14]. For instance, the number of internal neurons and the spectral radius of the reservoir are optimized using a genetic algorithm to improve the accuracy for time series prediction [15].

During the past few years, computational modeling of neural plasticity, which improves the adaptation of randomized learning models to the input signal, is a common approach to optimizing the reservoir neurons [16,17]. Neural plasticity mainly consists of synaptic plasticity and non-synaptic plasticity (intrinsic plasticity), i.e., rules for adjusting weights and activation functions.

The present research of neuroscience focuses primarily on synaptic plasticity, which is able to learn the structural information of input signal based on the activities stimulated by the neurons

* Corresponding author at: Department of Computer Science, University of Surrey, Guildford, GU2 7XH, UK.
E-mail address: yaochu.jin@surrey.ac.uk (Y. Jin).

https://doi.org/10.1016/j.neucom.2019.03.032
0925-2312/© 2019 Published by Elsevier B.V.

[16]. Biological experiments show that a single neuron also has the ability to alter its intrinsic excitability to match the received synaptic input by changing voltage gated ion channels [18–20]. This new neural plasticity mechanism is known as intrinsic plasticity (IP). At the same time, findings in neurobiology experiments indicate that responses of neurons to natural scenes are approximately exponential spike-count distributions in both primary visual cortex and the inferior temporal cortex [21]. Note that exponential distributions have the highest information entropy among all non-negative distributions while keeping a same mean [20,22].

However, until now, a deep understanding of adjustment process and the underlying mechanism of IP in neural circuits still lacks [20]. In addition to experimental research, computational modeling of the IP offers a novel option to better understand these neural plasticity mechanisms. Inspired by neuroscience and information theory, Triesch [20] proposed an IP learning rule built by computational modeling. Triesch assumed that the output distribution of the single sigmoid neuron is an exponential distribution with a fixed mean, and the IP rule can be derived by minimizing the Kullback–Leiber (KL) divergence between the actual output distribution and the desired exponential distribution of neurons.

In fact, Triesch's rule is closely related to the information maximization rule proposed by Bell and Sejnowski [23]. However, information maximization rule does not assume any form of the desired distribution. When the sensitive region of the activation function is used to respond to the high probability region of the input probability density function, the maximum mutual information between the input and output of an individual neuron can be achieved.

Compared with information maximization rule, Triesch's rule has an extra effect which helps neurons maintain a more suitable firing rate level for various synaptic input distributions [20]. However, this rule also decreases its sensitivity and discriminative ability, because most neurons work in the insensitive regions of the sigmoid function. In a follow-up study, Li [24] pointed out that the Weibull distribution is more appropriate for describing the real activity distribution of a single biological neuron. Li's rule makes more neurons operate in the sensitive region of the sigmoid function, while keeping the neuronal averaged output values low but not too low. In addition, The Weibull distribution can represent or approximate some different distributions by adjusting the rule parameters. For instance, the exponential distribution, the Rayleigh distribution, and the Gaussian distribution.

It should be pointed out that these studies are based on the desired output distribution of the neuron, which is a non-negative distribution [20]. Based on the above study, Schrauwen et al. [17] and Wardermann and Steil [25] extended the range of the desired distribution to [−1,1], using a more common hyperbolic tangent function and the Gaussian output distribution as the activation function and the desired distribution.

Existing IP rules for altering the intrinsic excitability of the reservoir neurons are mainly applied in the pre-training process of the ESN to achieve global plasticity optimization of the activation function within the reservoir [17,26]. The parameters of the IP rules such as the learning rate, the scale parameter and the shape parameter in the whole reservoir neurons are the same in the global plasticity optimization. Note, however, that the output distributions of real biological neurons may have different forms in different brain regions and species [24]. Therefore, existing IP rules apparently lack a certain degree of flexibility.

To further improve the flexibility of the IP rule and the learning performance of the ESN, we propose a local IP rule, where the learning rates, scale parameters and shape parameters for different neurons can be different. To this end, the covariance matrix adaptation evolution strategy (CMA-ES) [27] algorithm is used for optimizing the above parameters.

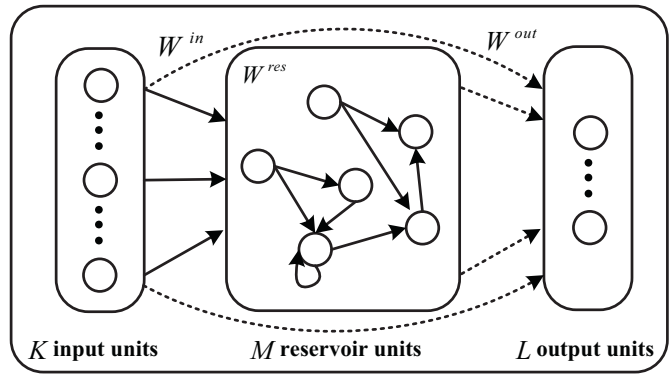

Fig. 1. The basic structure of ESN model.

The main contributions of this paper are summarized as follows. First, an evolution strategy is used to generate local IP rules so that different neurons within the reservoir can adopt different parameters in the plasticity rules or even different types of plasticity rules. Second, we show that IP rules with negative learning rates, which are called anti-IP rules, will emerge by means of evolution and co-exist with IP rules with positive learning rates. Third, our empirical results on time series benchmark problems, including the sunspot series problem, the tenth order nonlinear autoregressive moving average (NARMA) system, and the Henon map system, demonstrate that the proposed local IP rules consisting of both IP rules and anti-IP rules are able to consistently enhance the learning performance of the ESN. Finally, the proposed method is compared with the original ESN [1], simple cycle reservoir [6] (SCR), leaky integrator ESN [9] (LESN) and growing ESN [11] (GESN) on the above benchmark problems. The comparative results confirm that the ESN with local plasticity rules is able to outperform the state-of-the-art ESN models.

2. Related work

2.1. Echo state network

ESN is a relatively new recurrent neural network that simulates the large-scale sparse recurrent connection of cerebral cortex neurons. Compared with the traditional recurrent neural network, ESN has the advantage that the training process of the network is significantly simplified, and the good generalization ability is guaranteed. Therefore, it has great development potential and application value.

The basic ESN model includes an input layer, a reservoir and an output layer. The most special feature of ESN is adopting a lot of randomly and sparsely connected reservoir units as the hidden layer for high dimension nonlinear representation of the input information. In general, the connection weights of the input layer and reservoir are generated by random initialization [28]. Only the connection weights of the output layer need to be trained by a simple linear regression method.

The linear regression method is usually used to train the connection weights of output layer which has the advantages of simple training algorithm and global optimization.

The basic structure of ESN model without output feedback connections is shown in Fig. 1. It is composed of an input layer, a reservoir and an output layer. The unit number of input, reservoir and output are denoted as K, M and L, respectively. W^{in}, W^{res} and W^{out} represent the connection weights of input layer-to-reservoir, internal reservoir, and input layer and reservoir-to-output layer (dashed arrows), respectively. The input states $x^{in}(t+1)$ and

activation states $x(t+1)$ of the reservoir at time $t+1$ are represented by Eqs. (1) and (2) [29].

$$x^{in}(t+1) = W^{in}u(t+1) + W^{res}x(t) \tag{1}$$

$$x(t+1) = f(a_i(t+1)x^{in}(t+1) + b_i(t+1)) \tag{2}$$

where $u(t+1)$ represents the input of the ESN at time $t+1$. And the sigmoid function is used as activation function of the reservoir units. The parameters $a_i(t+1)$ and $b_i(t+1)$ respectively are the gain and the bias of the ith reservoir neuron's activation function at time $t+1$. Usually, the gain is 1 and the bias is 0 in the standard sigmoid function.

According to the activation states of the reservoir, the activation states $y(t+1)$ of output layer at time $t+1$ are as follows:

$$y(t+1) = W^{out}[x(t+1), u(t+1)] \tag{3}$$

The output weights W^{out} are determined by a learning algorithm which usually minimizes the difference between the actual output y and the desired output y^{desired} of the network.

In this work, the normalized root mean square error (NRMSE) is adopted to be an evaluation function which is as shown in (4) [11]:

$$NRMSE(W^{out}) = \sqrt{\frac{\sum_{t=1}^{T}\left(y^{desired}(t) - y(t)\right)^2}{T\sigma^2}} \tag{4}$$

where σ^2 represents the variance of the desired outputs. NRMSE is the normalized RMSE (root mean square error), and the smaller the NRMSE value, the better performance in terms of learning ability or prediction accuracy. Under the condition that the NRMSE of the desired output and the actual output is minimum, the formula for calculating the least-squares solution of the output weights is as follows:

$$W^{out} = (X^{\mathrm{T}}X)^{-1}X \cdot Y \tag{5}$$

where $X = [x(1), ..., x(T), u(1), ..., u(T)]$, for the training of the output weights W^{out}, the concatenation of the reservoir states and input vectors and desired output of the network in each moment will be stored in X and Y over time $t = 1, 2, ..., T$, respectively.

2.2. Intrinsic plasticity

IP is a common physiological phenomenon observed in many species and brain regions [20]. IP plays an important role in maintaining the activity level of individual neurons within a certain range of homeostatic control while maximizing information transmission to downstream targets [30,31].

Homeostatic characteristic can effectively maintain a stable metabolic cost of the system. Maximizing information transmission can maximize the mutual information between neurons input and output, so that the neuron's response curve can match the synaptic input distribution. In other words, IP rule can efficiently represent the synapse input in the condition of a stable metabolic cost. However, neither Triesch's rule [30] nor the information maximization rule [31] could efficiently resolve the contradiction between minimizing the firing rate and maximizing the sensitivity in the IP model. In further study, Li pointed out that Weibull distribution is more appropriate in describing the real activity distribution of a single biological neuron [24].

The goal of IP rule is to obtain an approximately desired distribution of the firing rate level while obeying constraints on its metabolic costs. In order to investigate the impacts of IP mechanism on functions and dynamics of neural circuits [24], we choose Li's rule as an adaptation rule for the following experiment and theoretical analysis. The desired output distribution of the Li's rule is also based on the non-negative distribution, therefore the sigmoid function is employed as the activation function of the neurons.

$$y = f(x) = \frac{1}{1 + e^{-(ax+b)}} \tag{6}$$

The probability density function of the Weibull distribution is as follows [32]:

$$f(x;\alpha,\beta) = \begin{cases} \frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1} e\left(-\left(\frac{x}{\alpha}\right)^{\beta}\right) & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{7}$$

where $\alpha > 0$ and $\beta > 0$ respectively represent the scale parameter and the shape parameter of the Weibull distribution. Shape parameter β is the most important parameter which determines the basic shape of the probability density function. Scale parameter α plays the role of scaling the curve, but does not affects the shape of the distribution. In addition, other types of distribution can be represented or approximated by changing the shape parameter of the Weibull distribution, for example, the exponential distribution, the Rayleigh distribution, the Gaussian distribution, etc. [33].

Specifically, when $0 < \beta < 1$, the probability density function curve tends to be infinite as x approaches zero which is strictly decreasing (see Fig. 2(a)). When $\beta = 1$, the Weibull distribution becomes the same as the exponential distribution, as shown in Fig. 2(b), the Weibull distribution reduces to the Rayleigh distribution when β=2, as shown in Fig. 2(d). When $1 < \beta < 3.6$, the top-left part of the Weibull distribution is much less than the exponential distribution. The distribution looks very much like an exponential distribution with a droop-head, refer to Fig. 2(c). Recent research finding also shows that the firing-rate distribution may have a droop-head [24]. So the exponential-like distribution with a droop-head is considered to be more appropriate in describing the real firing-rate distribution used as the desired output distribution in this work. When $\beta > 3.6$, the Weibull distribution looks like a Gaussian distribution, except that the top part of the distribution leans more to the right side than the normal Gaussian distribution, see Fig. 2(f). In particular, when β is about 3.6, the Weibull distribution can represent the Gaussian distribution, as shown in Fig. 2(e).

The derivation of the Li's IP rule is similar to initial IP rule which is also based on the KL divergence. For a sigmoid neuron model, the IP rule is derived by minimizing the KL divergence between actual output distribution and the Weibull distribution.

The gain a and bias b of the sigmoidal neuron are updated by stochastic gradient descent rule at each time step which is as follows:

$$\Delta a_i(t+1) = \frac{\eta}{a_i(t)} + \eta\Delta b_i(t)x_i^{in}(t) \tag{8}$$

$$\Delta b_i(t+1) = \eta\left[\beta(1 - x_i(t)) - x_i(t) - \frac{\beta}{\alpha^{\beta}}x_i(t)^{\beta}(1 - x_i(t))\right] \tag{9}$$

$$a_i(t+1) = a_i(t) + \Delta a_i(t+1) \tag{10}$$

$$b_i(t+1) = b_i(t) + \Delta b_i(t+1) \tag{11}$$

where $\Delta a_i(t+1)$ and $\Delta b_i(t+1)$ are the change of the gain and the bias of the ith reservoir neuron's activation function at the time t, respectively. η is the learning rate of IP rule. $\alpha > 0$ and $\beta > 0$ represent the shape parameter and the scale parameter of the Weibull distribution, respectively.

3. Local intrinsic plasticity rule

Existing IP rules for altering the intrinsic excitability of the reservoir neurons are mainly applied in the pre-training process of

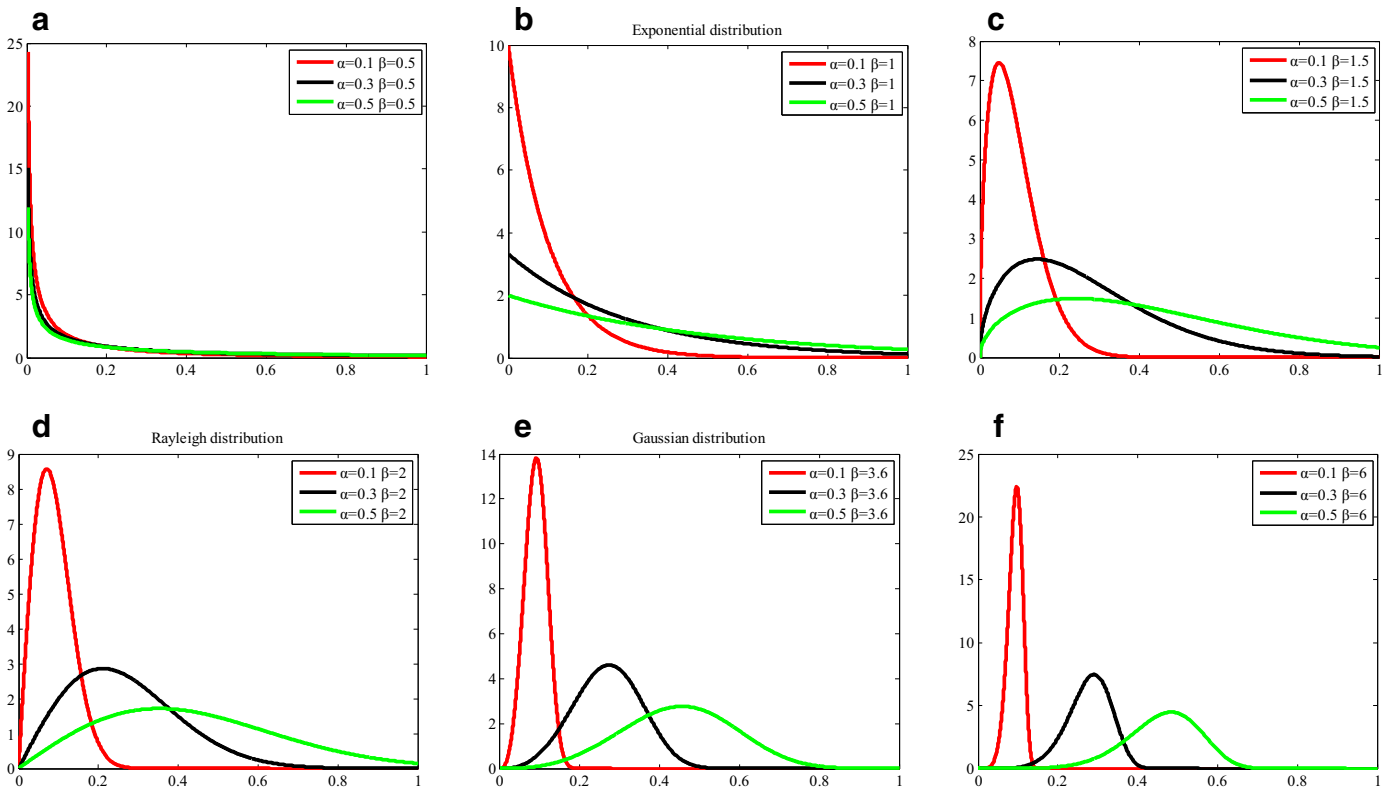

Fig. 2. The Weibull distribution with different shape parameters and scale parameters.

the network to achieve the global plasticity optimization of neuron's activation function. But global IP rule lacks of certain degree of flexibility for different output distributions [24].

To further improve the flexibility of IP rule and the learning performance of ESN, we propose a local plasticity rule that different neurons within the reservoir can adopt the IP rule with different learning rates, scale parameters and shape parameters to alter its intrinsic excitability in this paper. The CMA-ES algorithm is used for optimizing the above parameters. From the point of view of a single neuron, the most efficient representation is that different neurons can use different IP rules to represent different inputs as much as possible, so that neurons can carry and transmit more information [34]. A diagram of the pre-training and training the ESN with the IP rule is showed in Fig. 3.

The local IP learning rule is presented in the following Eqs. (12) and (13):

$$\Delta a_i(t+1) = \frac{\eta_i}{a_i(t)} + \eta_i \Delta b_i(t) x_i^{in}(t) \tag{12}$$

$$\Delta b_i(t+1) = \eta_i \beta_i (1 - x_i(t)) - \eta_i x_i(t) - \eta_i \frac{\beta_i}{\alpha_i^{\beta_i}} x_i(t)^{\beta_i} (1 - x_i(t)) \tag{13}$$

where $\Delta a_i(t+1)$ and $\Delta b_i(t+1)$ are the change of the gain and the bias of the ith reservoir neuron's activation function at the time t, respectively. α_i and β_i represent the IP rule's shape parameter and the scale parameter of the ith reservoir neuron, respectively. η_i is the IP rule's learning rate of the ith reservoir neuron. For the gain and the bias update formula, the local IP learning rule is the same as the global IP learning rule.

The only difference between the global IP rule and the local IP rule, shown from the above formulas, is that the local plasticity rule applies corresponding parameter rules to different reservoir neuron.

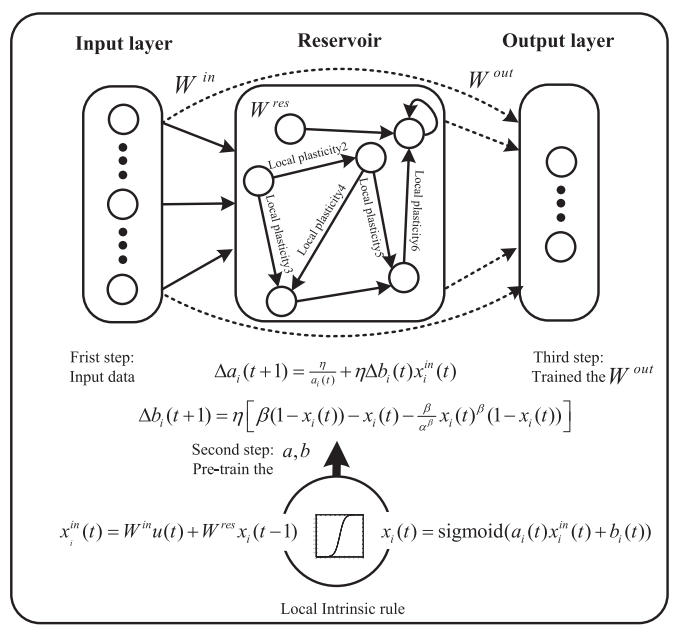

Fig. 3. Pre-training and training the ESN with the IP rule.

In recent years, neuroevolution has been popular for optimization of neural networks [35]. Here, we use an evolutionary algorithm to construct local IP learning rules. The CMA-ES algorithm is widely used in dealing with continuous problems, the detailed introduction to CMA-ES algorithm can be seen in [27]. The core idea of CMA-ES algorithm is to handle the problems of dependencies and scaling between variables by adjusting the covariance

Algorithm 1 Local IP rule.

1: Initialize ESN, generation $g = 0$.
2: Input data through the input layer.
3: Pre-train gain a and bias b of the reservoir neuron by using global IP.
4: Train the readout weights by the LSE method.
5: Create initial populations (learning rate η_i^0, scale parameter α_i^0, shape parameter β_i^0)
6: **Repeat**
7: Input data.
8: Update gain and bias of the ith reservoir neuron by training data.
$\Delta a_i^g(t+1) = \frac{\eta_i^g}{a_i^g(t)} + \eta_i^g \Delta b_i^g(t) x_i^{in}(t)$
$\Delta b_i^g(t+1) = \eta_i^g \beta_i^g (1 - x_i(t)) - \eta_i^g x_i(t) - \eta_i^g \frac{\beta_i^g}{\alpha_i^{g\beta_i^g}} x_i(t)^{\beta_i^g} (1 - x_i(t))$
7: Calculate training NRMSE of ESN (Fitness value) by using the local IP.
7: Calculate testing NRMSE of ESN by using the local IP.
9: Select parent, perform recombination, and generate offspring η_i^g, α_i^g and β_i^g.
10: Output and save η_i^g, α_i^g, β_i^g and testing NRMSE.
11: $g = g + 1$
12: **Until** the stopping criterion is met (given number of generations)

matrix in the normal distribution. Therefore, we use the CMA-ES algorithm to select different rule parameters of different neuron within the reservoir. The NRMSE on the training data is used as the fitness function of the evolutionary algorithm. The population sizes of the parent and offspring are set to 20 and 10, respectively. The termination criterion is the predefined maximum number of iteration which is set to 5000. The pseudo code of the local IP rule is presented in Algorithm 1.

4. Experiments

In order to verify the effectiveness of the local IP rule on prediction problems, tenth-order NARMA system, Henon map system and sunspot series prediction problem are used as the benchmark.

For the setting of connection weight between the layers, the connection weights W^{in} are randomly generated over the interval $[-0.1, 0.1]$. The connection weights of internal reservoir W^{res} are scaled to ensure the stability of the network and to satisfy the necessary condition of the echo state property which is that the spectral radius of W^{res} ($\rho = \rho max(W^{res})$) must always be smaller than 1 [2]. The NRMSE is used to evaluate the performance of the network. It should be noted that rule parameters of the global IP are optimized by CMA-ES under the same condition of parameter settings for a fair comparison.

4.1. NARMA system

The NARMA system is discrete time system which is also one of the common task for nonlinear time series prediction [6,11,29]. In this paper, we choose tenth-order NARMA system as prediction task. The equation of tenth-order NARMA system is expressed as follows:

$$x(n+1) = 0.3x(n) + 0.05x(n)\sum_{i=0}^{9} x(n-i) + 1.5u(n-9)u(n) + 0.1 \quad (14)$$

where $u(n)$ and $x(n)$, respectively, are the input and output of the system at time n. $u(n)$ is randomly generated from a uniform distribution within the range of 0 and 0.5. The initial state of the system $x(1)$, $x(2)$,…,$x(10)$ is all set to 0. And 5200 samples are generated by formula (14). The initial 1000 training data are only used for washout of the network, the following 1800 training data are used to pre-train, and the rest samples are used for test. For the NARMA system, $u(n)$ and $x(n)$ are used as network input and output, respectively.

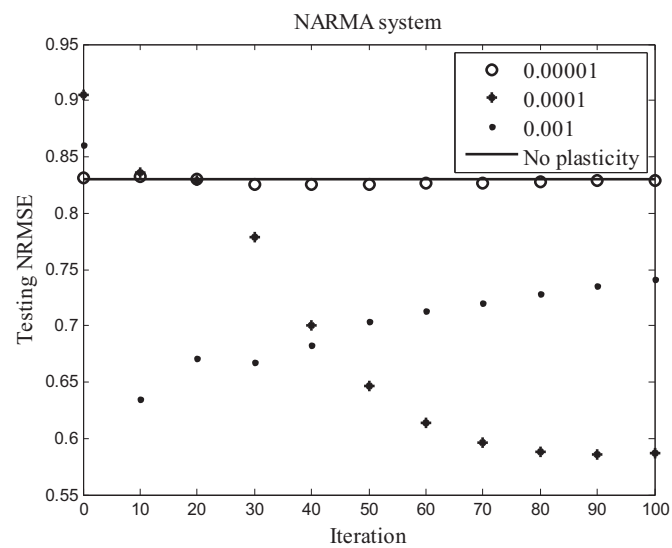

Fig. 4. The NRMSE of different learning rates of the IP rule on the test data.

The learning rate parameter of the IP rule plays an important role in the learning performance of the network. For a large learning rate parameter, it will easily lead to the network divergence. However, the weakness of too small learning rate parameter is the slow convergence speed. Therefore, the learning rate parameter should be selected as large as possible while meeting the convergence request.

In this part, we firstly test several of different learning rate parameters of global IP rule. And the appropriate learning rate parameter is determined by comparing the convergence and the learning performance of the network, and then it will be used for the initial learning rate of the local IP rule. As discussed in Section 2.2, an exponential-like distribution with a droop-head is considered to be appropriate for describing the firing-rate distribution of the scale and shape parameters. Therefore, the scale and the shape parameters of the global IP rule are set in the range of [0.1, 1] and [1, 3.6], respectively. For the NARMA system, the initial scale parameter and the shape parameter of the local IP rule are set to 0.3 and 1.5, as recommended in [24].

From Fig. 4 we can see that learning rate must be a small positive constant in order to guarantee the stable convergence of the network. Therefore, for NARMA system, the initial learning rate of the local IP is set to 0.00001. In addition, we also compare

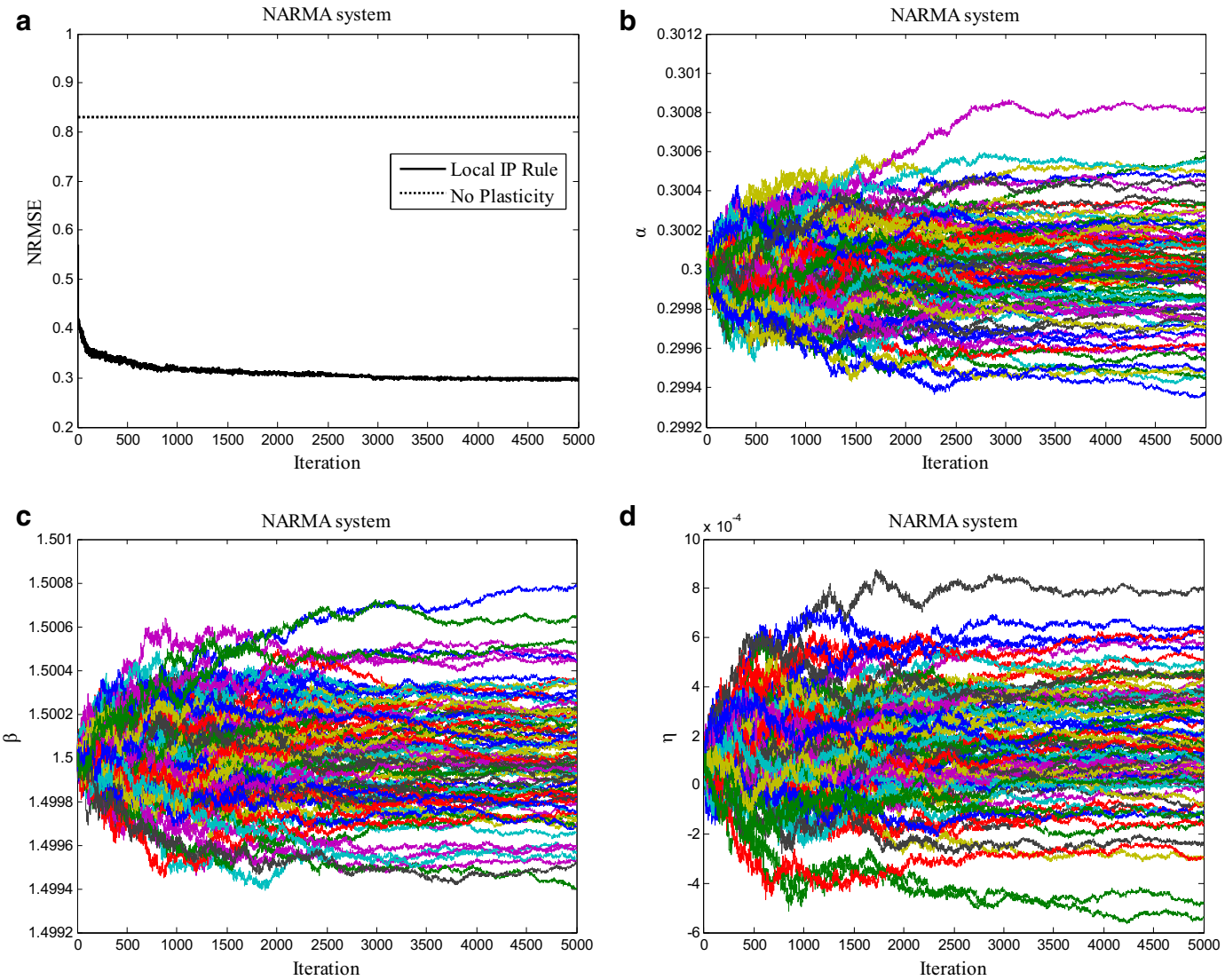

Fig. 5. The convergence curves of the NRMSE, scale parameter α, shape parameter β and learning rate η that applied local IP rule for the NARMA system.

various learning rates of global IP rule for other benchmarks. For the Henon map system and the sunspot series task, the initial learning rates are set to 0.000001 and 0.0001.

The training NRMSE of the best individual on the NARMA system over the generations are shown in Fig. 5(a). Compared to the ESN with no plasticity, the ESN with local IP rule can significantly improve the prediction accuracy of the network.

In addition, scale parameter α, see Fig. 5(b), shape parameter β (see Fig. 5(c)) and learning rate η (see Fig. 5(d)) also tend to be convergent in the optimization process. The scale parameter α and shape parameter β corresponding to each neuron seem different. Nevertheless, the density function of each neuron is approximately an exponential distribution with a droop-head. Such a distribution is believed to be more appropriate in describing the real activity distribution of a single biological neuron [24]. From Fig. 5(b) and (c), we can see that the optimized scale and shape parameters are still close to 0.3 and 1.5, respectively.

For learning rate η, the learning rate is set to a small positive constant in the previous research of the IP learning rule. However, we found that in addition to positive learning rates, there also exist negative learning rates in the local IP rule. The numbers of positive and negative learning rates are also roughly the same. It is very common for a learning rate to be negative in the research of the synaptic plasticity, such as anti-Hebb [36] and anti-Oja [37]. These synaptic plasticity learning rules have also been shown to improve the learning performance. Similar to these anti-synaptic plasticity learning rules, an IP with a negative learning rate is called an anti-IP rule in this paper.

The experimental results show that the proposed local IP rule is able to simultaneously take the role of IP rule and anti-IP rule into account to enhance the learning performance of the network. The prediction results based on the ESN with local IP for the NARMA system are shown in Fig. 6. As IP learning rules do not adjust the weights of the reservoir neurons, the spectral radius of the reservoir remains unchanged.

The output weights distribution is a commonly used analysis index in reservoir computing. As pointed out in [38], large output weights may result in poor generalization capabilities and require that the reservoir states should be represented with a high precision. From Fig. 7, we can see that the output weights of the ESN with the global IP rule are widely distributed, while those of the ESN with the local IP rule fall in a much smaller range.

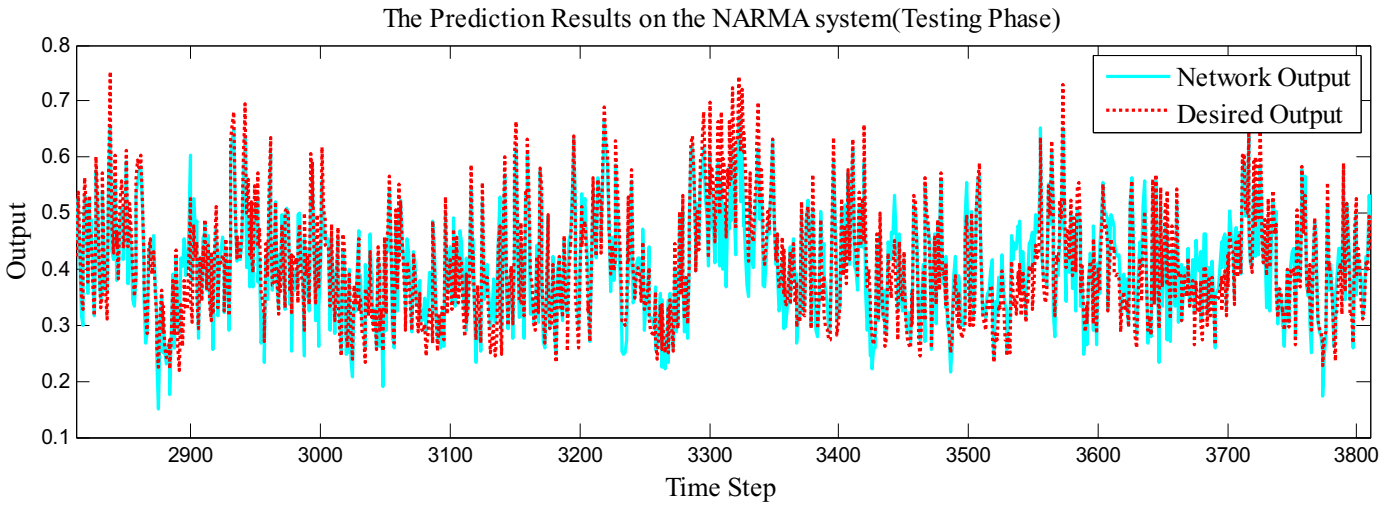

Fig. 6. The prediction results based on the ESN with local IP for the NARMA system.

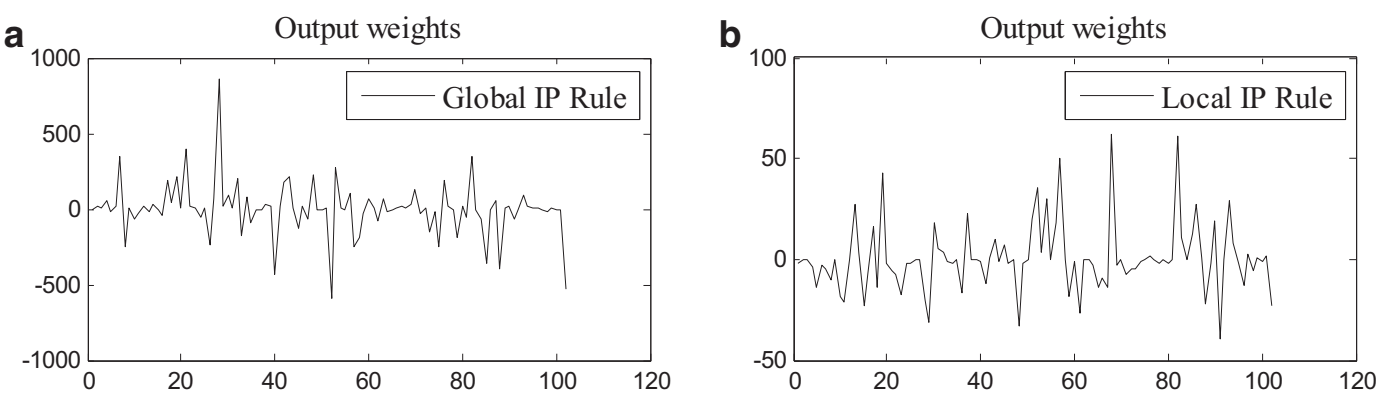

Fig. 7. The NARMA system: output weights distribution comparison of two IP rules. (a) Global IP rule (b) local IP rule.

Table 1
Some network parameters and prediction results of different methods for NARMA system.

Approach	Testing NRMSE (Mean)	Testing NRMSE (Std.)	Reservoir size	Spectral radius	Sparsity
Local IP	**0.3329**	**7.6661e−05**	100	0.8234	0.05
Global IP	0.5044	6.1721e−04	100	0.8234	0.05
GESN	0.8289	3.4566e−04	100	0.8620	0.05
LESN	0.6377	1.2991 e−03	100	0.8162	0.05
OESN	0.8306	7.8666e−04	100	0.8234	0.05
SCR	0.8321	7.5598e−04	100	0.7779	0.005

In addition, the proposed method is compared with some other existing methods in the same reservoir size and sparsity. The network parameters and the mean values and variance of the testing NRMSE are listed in Table 1. From Table 1 we can see that the ESN with global IP rules has better prediction performance compared to the ESN, SCR, LESN and GESN. The ESN using local IP rule can significantly improve the prediction accuracy of the network, and above methods are run independently 30 times. It should be noted that the computational complexity of the ESN with evolved local IP rules is higher than ESN models without using any plasticity or using a global IP rule. It typically takes tens of minutes for evolving the plasticity rules using the training dataset for the NARMA problem.

4.2. Henon map prediction

The Henon map system is a typical chaotic system that is inspired by the Lorenz attractor [39]. Because of the symmetry of chaotic system, Henon map system is commonly used in information encryption [40] and system identification [41]. The Henon map system is given as below:

$$\begin{cases} u_1(t+1) = u_2(t) - au_1^2(t) + 1 \\ u_2(t+1) = bu_1(t) \end{cases} \tag{15}$$

where $a = 1.4$ and $b = 0.3$. The system emerges chaotic phenomenon, and the initial state of this system is set to $u_1(1)$=$u_2(1)$=0. $u_1(t)$ and $u_2(1)$ are used to predict $u_1(t+1)$. The initial 1250

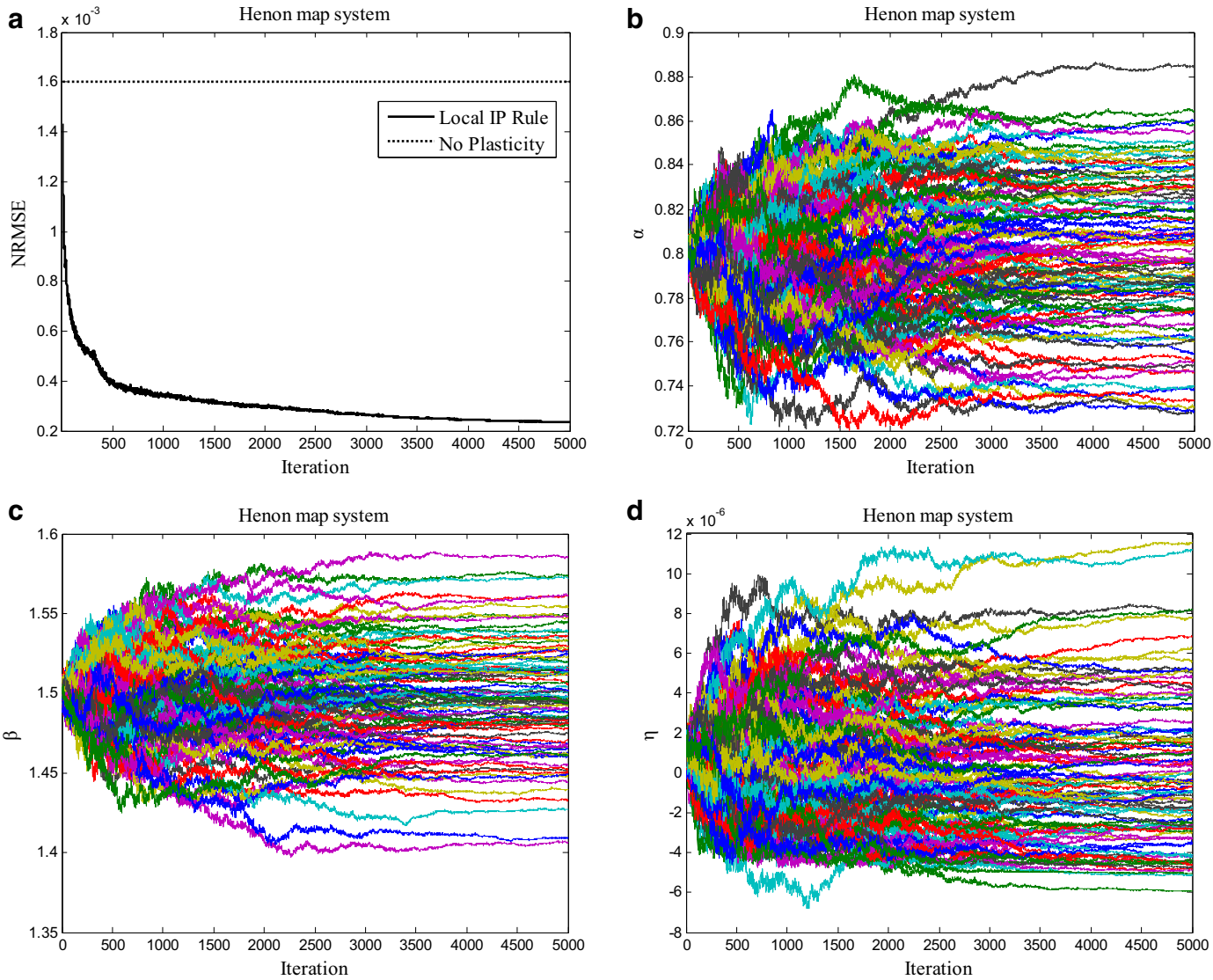

Fig. 8. The convergence curves of the NRMSE, scale parameter α, shape parameter β and learning rate η that applied local IP rule for the Henon map system.

training data are only used for washout of the network, the following 200 training data are used to pre-train the gain and the bias of the reservoir neuron's activation function and update the output weights, the rest of the samples are used as the testing set, the initial scale parameter and the shape parameter of the local IP rule are set to 0.8 and 1.5.

For the Henon system, similar results are obtained. The convergence curves of the NRMSE, scale parameter, shape parameter and learning rate using the local IP rule are shown in Fig. 8. Compared to the ESN with no plasticity, the ESN with local IP rule can significantly improve the prediction accuracy of the network, see Fig. 8(a).

In addition, scale parameter α, as shown in Fig. 8(b), shape parameter β (see Fig. 8(c)) and learning rate η (see Fig. 8(d)) also tend to be convergent in the optimization process. The density function of each neuron is also approximately exponential distribution but with a droop-head [24]. For learning rate η, we also found that in addition to positive learning rates, there also exist negative learning rates in the local IP rule. We also found that the coexisting phenomenon of IP rule and anti-IP rule is exist in local IP rule. The prediction results based on the ESN with local IP for the Henon system are shown in Fig. 9.

Similarly, we can find from Fig. 10 that the output weights of the ESN with a global IP rule are much larger than those of local IP rules, implying that the ESN with local rules has better generalization capabilities.

In addition, the proposed method is compared with some other existing methods on the premise of same reservoir size and sparsity. The network parameters and the mean values and variance of the testing NRMSE are listed in Table 2. From Table 2 we can see that the ESN with global IP rules has better prediction performance compared to the ESN, SCR, and LESN. The ESN using local IP rule can significantly improve the prediction accuracy of the network, and above methods are run independently 30 times.

4.3. Sunspot series prediction

The number of sunspots is the basic parameter of solar activity level, which is also closely related to geomagnetic disturbance, satellite telecommunication and weather variation. The sunspot series prediction task is also a common benchmark task for time series prediction [6,11,42]. In this section, we used the smoothed monthly mean sunspot numbers from the National Oceanic and Atmospheric Administration (NOAA) as benchmark task.

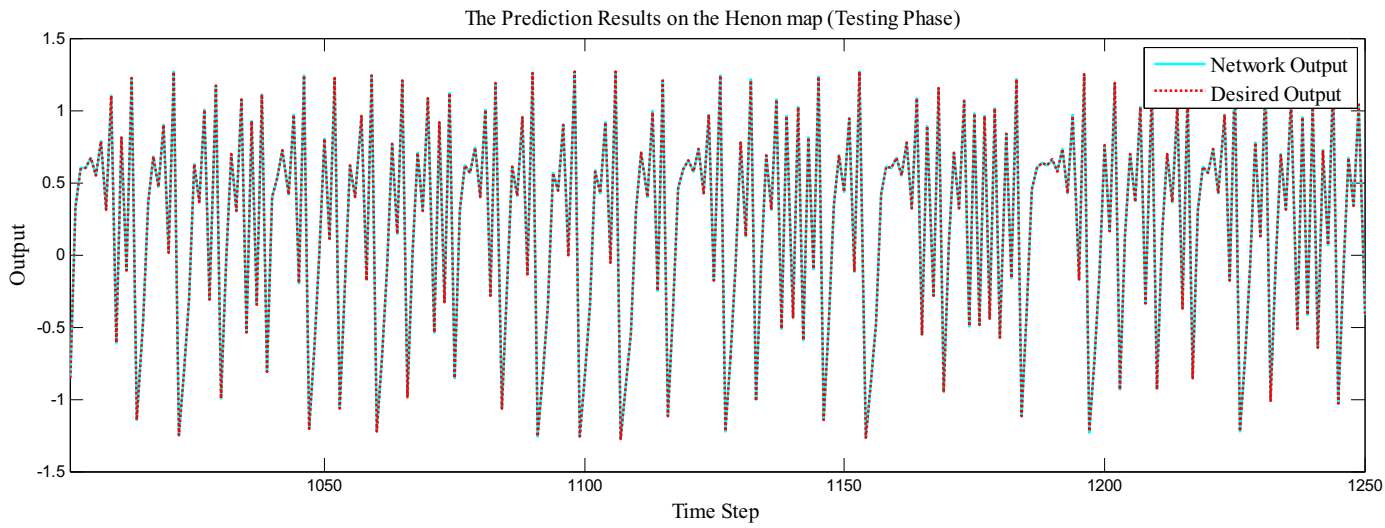

Fig. 9. The prediction results based on the ESN with local IP for the Henon map system.

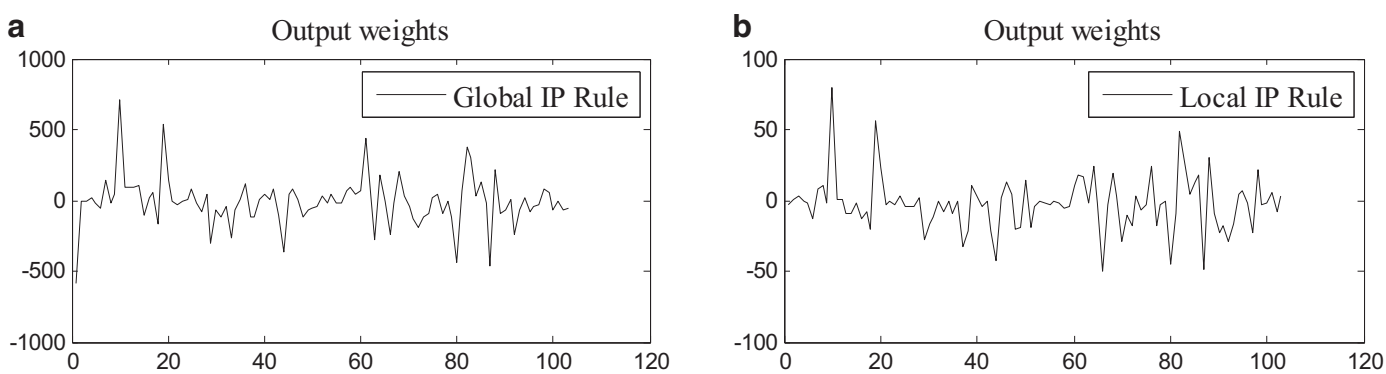

Fig. 10. The Henon map system: output weights distribution comparison of two IP rules. (a) Global IP rule (b) local IP rule.

Table 2
Some network parameters and prediction results of different methods for the Henon map system.

Approach	Testing NRMSE (Mean)	Testing NRMSE (Std.)	Reservoir size	Spectral radius	Sparsity
Local IP	**2.1399 e−04**	**6.1103e−05**	100	0.8563	0.05
Global IP	0.0011	4.3215e−04	100	0.8563	0.05
GESN	0.0008	5.5118e−04	100	0.8560	0.05
LESN	0.0014	1.8906e−04	100	0.8396	0.05
OESN	0.0016	6.2112e−04	100	0.8563	0.05
SCR	0.0012	2.8135e−04	100	0.7822	0.005

The sunspots dataset includes 3174 samples of monthly mean sunspots activity numbers from 1750 until 2013 [43]. In this part, the initial 500 training data are only used for washout of the network, the following 1500 training data are used to pre-train the gain and the bias of the reservoir neuron's activation function and update the output weights, the rest of the samples are used as the testing set. For the sunspots dataset, previous data $y(t-20)$,$y(t-10)$, and current data $y(t)$ are used to predict the network output of the next time $y(t+1)$. Therefore, the number of input and output neurons is 3 and 1, respectively.

For the sunspot series, the initial scale parameter and the shape parameter of the local IP rule are set to 0.3 and 1.5. The convergence curves of the NRMSE, α, β and η using the local IP rule is shown in Fig. 11. Compared to the ESN with no plasticity, the ESN with local IP rule can significantly improve the prediction accuracy of the network (see Fig. 11(a)). And the density function of each neuron is also approximately exponential distribution but with a droop-head (see Fig. 11(b) and (c)) [24]. For learning rate η, we also found that in addition to positive learning rates, there also exist negative learning rates in the local IP rule (see Fig. 11(d)). And we also found that IP rule and anti-IP rule can coexist in local IP rule. The prediction results based on the ESN with local IP for the sunspots dataset is shown in Fig. 12. In addition, we can find from Fig. 13 that the output weights of the ESN with a global IP rule are larger than those of the ESN with local IP rules.

Compared with several state-of-the-art ESN models, the network parameters and the mean values and variance of the testing NRMSE are listed in Table 3. From Table 3, we can see that the

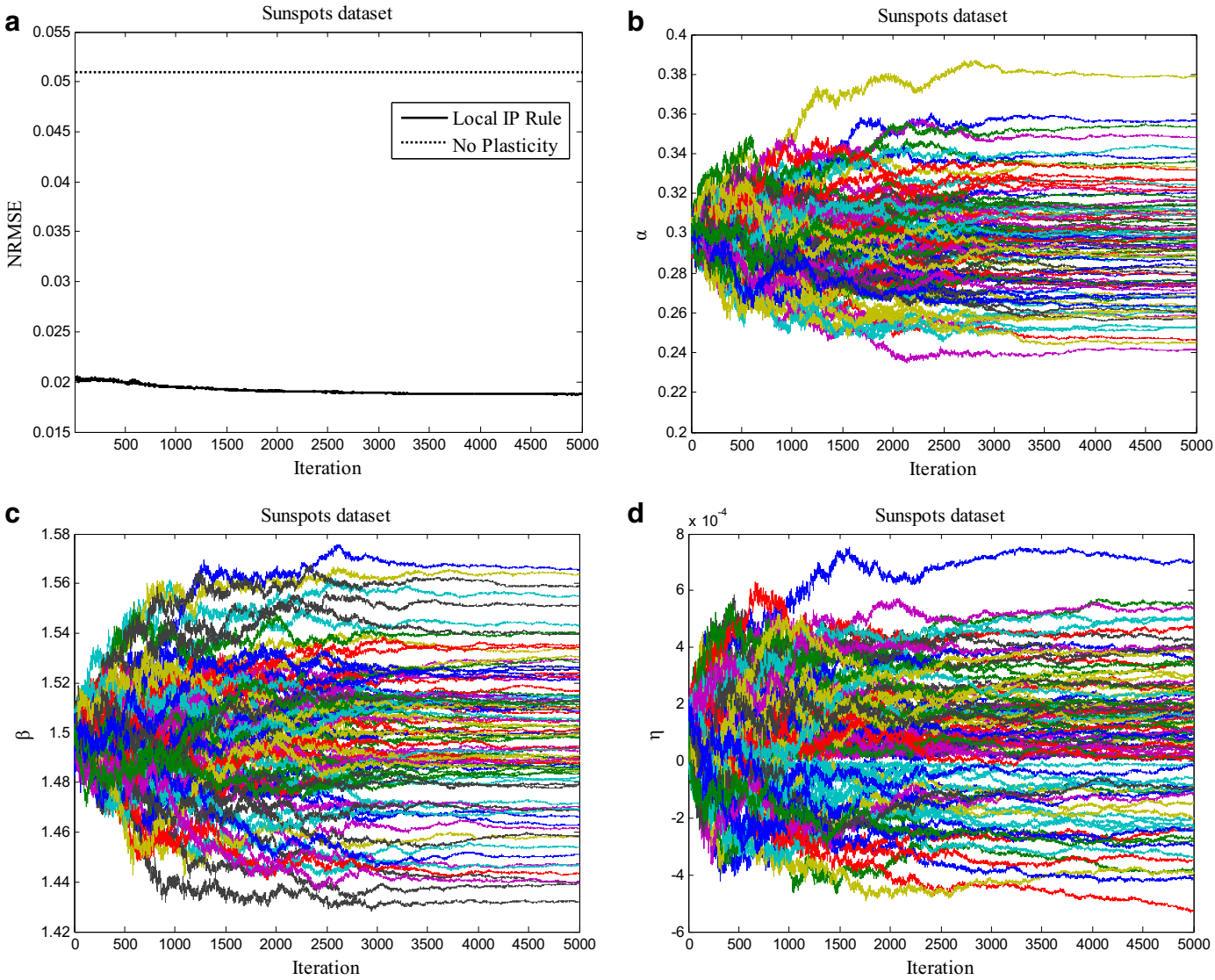

Fig. 11. The convergence curves of the NRMSE, scale parameter α, shape parameter β, learning rate η of ESN that applied local IP rule for the sunspots dataset.

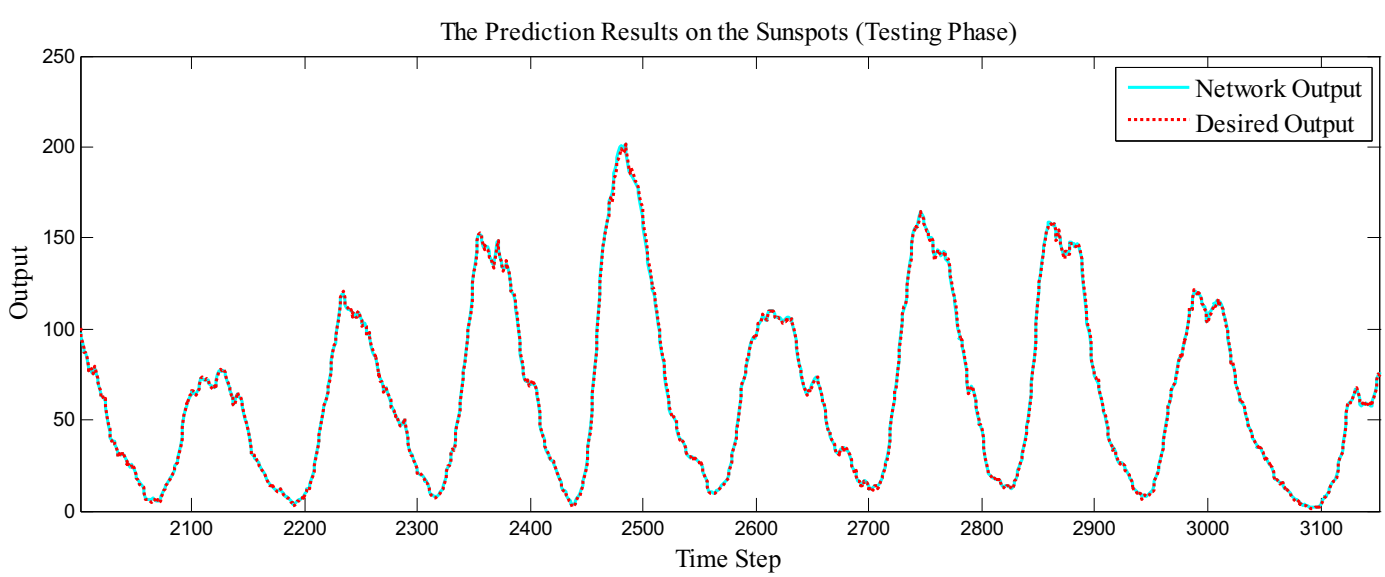

Fig. 12. The prediction results based on the ESN with local IP for the sunspots dataset.

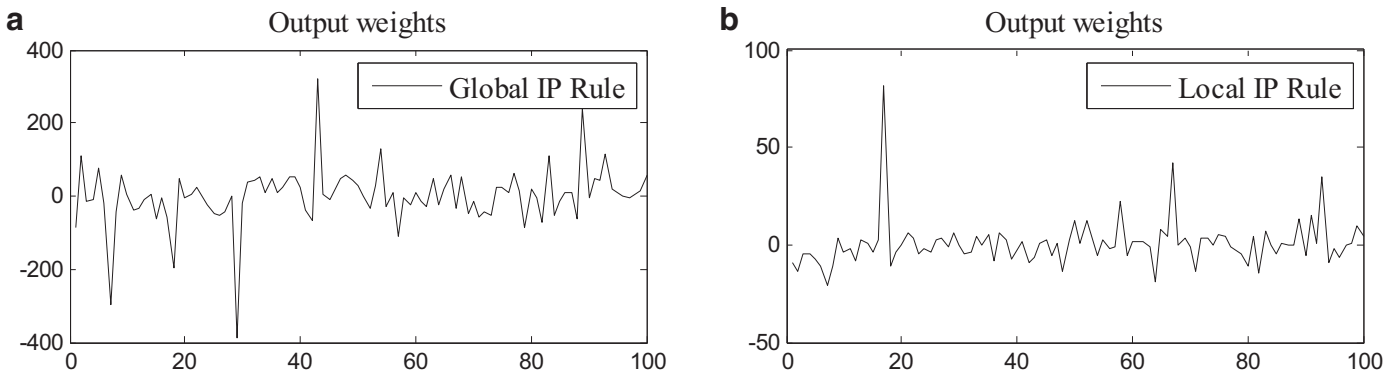

Fig. 13. The sunspot series: output weights distribution comparison of two IP rules. (a) Global IP rule (b) local IP rule.

Table 3
Some network parameters and prediction results of different methods for sunspot series prediction.

Approach	Testing NRMSE (Mean)	Testing NRMSE (Std.)	Reservoir size	Spectral radius	Sparsity
Local IP	**0.0198**	**0.0002**	100	0.7829	0.05
Global IP	0.0277	0.0041	100	0.7829	0.05
GESN	0.0335	0.0057	100	0.8743	0.05
LESN	0.0404	0.0047	100	0.8286	0.05
OESN	0.0510	0.0539	100	0.7829	0.05
SCR	0.0421	0.0031	100	0.7495	0.005

ESN with global plasticity rules has better prediction performance to the SCR, LESN, OESN and GESN. The ESN with local IP rule can significantly improve the prediction accuracy of the network.

5. Conclusion

The goal of IP rule is to obtain an approximately desired distribution of the firing rate level while obeying constraints on its metabolic costs. Therefore, the form of expectation distribution is vital to network performance. Existing IP rule for altering the intrinsic excitability of the reservoir neurons can only select a set of fixed rule parameters for the whole reservoir neurons. The lack of flexibility in intrinsic plasticity learning rules will inevitably affect the learning performance.

In this paper, we present an ESN with local IP rule built by different reservoir neurons which can adopt the IP rule with different rule parameters to adjust its intrinsic excitability, such as scale parameter, shape parameter and learning rate. The Weibull distribution is used as desired distribution in that it can represent or approximate some different distributions by adjusting the rule parameters.

Experimental results show that the density function of each neuron is also approximately exponential distribution but with a droop-head which is more appropriate in describing the real activity distribution of a single biological neuron. For learning rate η, we found that in addition to positive learning rates, there also exist negative learning rates in local IP rules, which can be considered as anti-intrinsic plasticity. The experimental results demonstrated that the proposed local IP rule is able to simultaneously take the role of IP rule and anti-IP rule into account to enhance the learning performance of the network. Although a deep understanding of the major roles of the anti-IP rule in learning still lacks and its underlying biological mechanisms remain elusive, increasing findings in computational modeling of neural plasticity may help us gain more insights into the neuronal intrinsic excitability.

Compared with several state-of-the-art ESN models and the ESN with a global plasticity rule, the proposed local IP rule is able to achieve much better performance in some prediction tasks.

The present research on computational modeling still focuses on one particular type of neural plasticity. Future work will aim to model the synergies between intrinsic and synaptic plasticity, which may contribute to additional improvements of the learning performance. In addition, the role of plasticity in deep ESNs will also be of great interest for modeling big data.

Acknowledgments

This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant CUSF-DH-D-2018101 and Grant 2232016D-32, in part by the National Nature Science Foundation of China under Grant 61473078, Grant 61503075, and Grant 61603090, in part by the International Collaborative Project of the Shanghai Committee of Science and Technology under Grant 16510711100, in part by the Agricultural Project of the Shanghai Committee of Science and Technology under Grant 16391902800, and in part by the Shanghai Science and Technology Promotion Project from Shanghai Municipal Agriculture Commission under Grant 2016-1-5-12.

References

[1] H. Jaeger, H. Haas, Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication, Science 304 (2004) 78–80.
[2] H. Jaeger, The “echo state” approach to analysing and training recurrent neural networks, überwacht. Lern. (2001).
[3] L. Mici, X. Hinaut, S. Wermter, Activity recognition with echo state networks using 3D body joints and objects category, in: Proceedings of the Twenty–forth European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges, Belgium, i6doc.com publication, 2016, pp. 465–470. ESANN 2016, April 27, 2016–April 29, 2016.
[4] R. Fourati, B. Ammar, C. Aouiti, et al., Optimized echo state network with intrinsic plasticity for EEG-based emotion recognition, in: Proceedings of the Twenty-forth International Conference on Neural Information and Processing, Guangzhou, China, Springer Verlag, 2017, pp. 718–727.
[5] M. Han, M. Xu, Laplacian echo state network for multivariate time series prediction, IEEE Trans. Neural Netw. Learn. Syst. 29 (2018) 238–244.
[6] A. Rodan, P. Tino, Minimum complexity echo state network, IEEE Trans. Neural Netw. 22 (2011) 131–144.
[7] Z.D. Deng, Y. Zhang, Collective behavior of a small-world recurrent neural system with scale-free distribution, IEEE Trans. Neural Netw. 18 (2007) 1364–1375.

[8] N. Jianjun, D. Zhidong, Synchronization enhancement of a scale-free highly clustered network, in: Proceedings of the 2008 Seventh World Congress on Intelligent Control and Automation, 2008, pp. 1996–2000.
[9] H. Jaeger, M. Lukosevicius, D. Popovici, et al., Optimization and applications of echo state networks with leaky-integrator neurons, Neural Netw. 20 (2007) 335–352.
[10] M.L.S. Cius, P. Dan, H. Jaeger, et al., Time warping invariant echo state networks, (2006).
[11] J. Qiao, F. Li, H. Han, et al., Growing echo-state network with multiple subreservoirs, IEEE Trans. Neural Netw. Learn. Syst. 28 (2017) 391–404.
[12] N. Chouikhi, R. Fdhila, B. Ammar, et al., Single- and multi-objective particle swarm optimization of reservoir structure in echo state network, in: Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), 2016, pp. 440–447.
[13] Y.C. Jin, R.J. Wen, B. Sendhoff, Evolutionary multi-objective optimization of spiking neural networks, in: J. MarquesDeSa, L.A. Alexandre, W. Duch, D.P. Mandic (Eds.), Proceedings of the Artificial Neural Networks – Icann 2007, Pt 1, Springer-Verlag Berlin, Berlin, 2007, p. 370.
[14] M. Han, Y. He, J. Wang, A multi-objective optimization model for alloy addition in BOS process based on ESN and modified MOPSO, in: Proceedings of the Second International Conference on Mathematics and Computers in Sciences and in Industry, Sliema, Malta, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 283–288. MCSI 2015, August 17, 2015–August 19, 2015.
[15] S.S. Zhong, X.L. Xie, L. Lin, et al., Genetic algorithm optimized double-reservoir echo state network for multi-regime time series prediction, Neurocomputing 238 (2017) 191–204.
[16] M.H. Yusoff, J. Chrol-Cannon, Y. Jin, Modeling neural plasticity in echo state networks for classification and regression, Inf. Sci. 364 (2016) 184–196.
[17] B. Schrauwen, M. Wardermann, D. Verstraeten, et al., Improving reservoirs using intrinsic plasticity, Neurocomputing 71 (2008) 1159–1171.
[18] Y. Li, C. Li, Synergies between intrinsic and synaptic plasticity based on information theoretic learning, PLoS One 8 (2013) e62894.
[19] W. Zhang, D.J. Linden, The other side of the engram: experience-driven changes in neuronal intrinsic excitability, Nat. Rev. Neurosci. 4 (2003) 885–900.
[20] J. Triesch, A gradient rule for the plasticity of a neuron's intrinsic excitability, in: W. Duch, J. Kacprzyk, S. Zadrozny (Eds.), Proceedings of the Fifteenth International Conference on Artificial Neural Network, Warsaw, Poland, Springer-Verlag Berlin, 2005, pp. 65–70.
[21] R. Baddeley, L.F. Abbott, M.C.A. Booth, et al., Responses of neurons in primary and inferior temporal visual cortices to natural scenes, Proc. R. Soc. B – Biol. Sci. 264 (1997) 1775–1783.
[22] W.B. Levy, R.A. Baxter, Energy efficient neural codes, Neural Comput. 8 (1996) 531–543.
[23] A.J. Bell, T.J. Sejnowski, An information-maximization approach to blind separation and blind deconvolution, Neural Comput. 7 (1995) 1129–1159.
[24] C.G. Li, A Model of neuronal intrinsic plasticity, IEEE Trans. Auton. Ment. Dev. 3 (2011) 277–284.
[25] M. Wardermann, J. Steil, Intrinsic plasticity for reservoir learning algorithms, in: Proceedings of the Fifteenth European Symposium on Artificial Neural Networks, Bruges, Belgium, d-side publication, 2007, pp. 513–518. ESANN 2007, April 25, 2007–April 27, 2007.
[26] F.Z. Xue, Q. Li, H.J. Zhou, et al., Reservoir computing with both neuronal intrinsic plasticity and multi-clustered structure, Cogn. Comput. 9 (2017) 400–410.
[27] N. Hansen, The CMA evolution strategy: a tutorial, Vu Le (2005).
[28] H. Jaeger, Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the echo state network approach, Ger. Nat. Res. Center Inf. Technol. Sankt August. Ger. Tech. Rep. 159 (2002).
[29] H. Jaeger, Adaptive nonlinear system identification with echo state networks, in: Proceedings of the Conference on Advance Neural Information Processing Systems, Neural Information Processing Systems Foundation, Vancouver, BC, Canada, 2003, pp. 593–600.
[30] J. Triesch, Synergies between intrinsic and synaptic plasticity mechanisms, Neural Comput. 19 (2007) 885–909.
[31] M. Stemmler, C. Koch, How voltage-dependent conductances can adapt to maximize the information encoded by neuronal firing rate, Nat. Neurosci. 2 (1999) 521–527.
[32] W. Weibull, A statistical distribution function of wide applicability, J. Appl. Mech. 13 (1951) 293–297.
[33] G.O. Silva, E.M.M. Ortega, G.M. Cordeiro, The beta modified Weibull distribution, Lifetime Data Anal. 16 (2010) 409–430.
[34] P. Foldiak, Forming sparse representations by local anti-Hebbian learning, Biol. Cybern. 64 (1990) 165–170.
[35] M. Han, K. Zhong, T. Qiu, et al., Interval type-2 fuzzy neural networks for chaotic time series prediction: a concise overview, IEEE Trans. Cybern. (2018) 1–12.
[36] J. Chrol-Cannon, Y.C. Jin, Computational modeling of neural plasticity for self-organization of neural networks, Biosystems 125 (2014) 43–54.
[37] S. Babinec, J. Pospichal, Improving the prediction accuracy of echo state neural networks by anti-Oja's learning, Springer-Verlag Berlin, Oporto, Portugal, 2007, pp. 19–28.
[38] H. Jaeger, Reservoir riddles: suggestions for echo state network research, in: Proceedings of the IEEE International Joint Conference on Neural Networks, IJCNN 2005, 3, IEEE, 2005, pp. 1460–1462.
[39] M. Henon, A two-dimensional mapping with a strange attractor, Commun. Math. Phys. (West Germany) 50 (1976) 69–77.
[40] J. Li, H. Liu, Colour image encryption based on advanced encryption standard algorithm with two-dimensional chaotic map, IET Inf. Secur. 7 (2013) 265–270.
[41] H. Dedieu, M.J. Ogorzalek, Identifiability and identification of chaotic systems based on adaptive synchronization, IEEE Trans. Circuits Syst. I: Fundam. Theory Appl. 44 (1997) 948–962.
[42] C.L. Yang, J.F. Qiao, H.G. Han, et al., Design of polynomial echo state networks for time series prediction, Neurocomputing 290 (2018) 148–160.
[43] National Oceanic And Atmospheric Administration. Sunspot Numbers. [Online], Available: https://www.ngdc.noaa.gov/stp/space-weather/solar-data/solar-indices/sunspot-numbers/depricated/international/tables/, (2014).

Xinjie Wang was born in Suzhou, Anhui Province, China, in 1991. He is currently working toward the Ph.D. degree with the College of Information Sciences and Technology, Donghua University, Shanghai, China. His current research interests include computational modeling of plasticity and data driven evolutionary optimization.

Yaochu Jin received the B.Sc., M.Sc., and Ph.D. degrees from Zhejiang University, Hangzhou, China, in 1988, 1991, and 1996, respectively, and the Dr.-Ing. degree from Ruhr University Bochum, Germany, in 2001. He is a Professor in Computational Intelligence, Department of Computer Science, University of Surrey, Guildford, U.K. He was also a Finland Distinguished Professor, University of Jyvaskyla, Finland and a Changjiang Distinguished Professor, Northeastern University, China. His main research interests include evolutionary computation, machine learning, computational neuroscience, and evolutionary developmental systems, with their application to data-driven optimization and decision-making, self-organizing swarm robotic systems, and bioinformatics. He has (co)authored over 300 peer-reviewed journal and conference papers and has been granted eight patents on evolutionary optimization. Dr Jin is the Editor-in-Chief of the IEEE Transactions on Cognitive and Developmental Systems and Complex and Intelligent Systems. He is an IEEE Distinguished Lecturer (2013–2015, 2017–2019) and was Vice President for Technical Activities of the IEEE Computational Intelligence Society (2014–2015). He was the recipient of the Best Paper Award of the 2010 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, and the 2014 and 2017 IEEE Computational Intelligence Magazine Outstanding Paper Award. He is a Fellow of IEEE.

Kuangrong Hao (M'17) received the B.S. and M.S. degrees in mechanical engineering from Hebei University of Technology, Tianjin, China in 1984 and 1989, respectively. She received the M.S. and Ph.D. degrees in applied mathematics and computer sciences from Ecole Normale Supérieur de Cachan, France in 1991, and Ecole Nationale des Ponts et Chaussées, Paris, France in 1995, respectively. She is currently a Professor at the College of Information Sciences and Technology, Donghua University, Shanghai, China. She has published more than 200 technical papers, more than 50 granted invention patents, and five research monographs. Her scientific interests include intelligent perception, intelligent systems and network intelligence, robot control, and intelligent optimization of textile industrial process.