

Bo YANG, Zhidong DENG

# An extended SHESN with leaky integrator neuron and inhibitory connection for Mackey-Glass prediction

© Higher Education Press and Springer-Verlag Berlin Heidelberg 2011

**Abstract** Echo state network (ESN) proposed by Jaeger in 2001 has remarkable capabilities of approximating dynamics for complex systems, such as Mackey-Glass problem. Compared to that of ESN, the scale-free highly-clustered ESN, i.e., SHESN, which state reservoir has both small-world phenomenon and scale-free feature, exhibits even stronger approximation capabilities of dynamics and better echo state property. In this paper, we extend the state reservoir of SHESN using leaky integrator neurons and inhibitory connections, inspired from the advances in neurophysiology. We apply the extended SHESN, called e-SHESN, to the Mackey-Glass prediction problem. The experimental results show that the e-SHESN considerably outperforms the SHESN in prediction capabilities of the Mackey-Glass chaotic time-series. Meanwhile, the interesting complex network characteristic in the state reservoir, including the small-world property and the scale-free feature, remains unchanged. In addition, we unveil that the original SHESN may be unstable in some cases. However, the proposed e-SHESN model is shown to be able to address the flaw through the enhancement of the network stability. Specifically, by using the ridge regression instead of the linear regression, the stability of e-SHESN could be much more largely improved.

**Keywords** echo state network (ESN), e-SHESN, Mackey-Glass problem, small-world phenomenon, scale-free distribution, ridge regression

## 1 Introduction

With the development of neuroscience, more and more

Received April 28, 2011; accepted August 13, 2011

Bo YANG, Zhidong DENG (✉)

State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Beijing 100084, China  
E-mail: michael@tsinghua.edu.cn

computational models have been proposed to explain lots of experimental phenomena. In the field of artificial neural network, more useful and interesting inspirations from neurophysiology and neuroanatomy are used for the existing computational models, such as spiking networks [1–3] and Kohonen network [4,5], in order to improve their performance on specific problems. Among these models, recurrent neural network (RNN) is one of the most successful models [6]. In 2001, Jaeger proposed echo state network (ESN) for function approximation, chaotic time-series prediction, and modeling of nonlinear dynamic system [7]. The ESN is similar to liquid state machine on the reservoir architecture [8], which is widely exploited in chaotic system prediction and pattern matching [9,10]. The ESN contains a “rich” state reservoir as a hidden layer, of which the reservoir connection weight is assigned randomly. For the training of ESN, we just need to adjust the output weight using linear regression algorithm [11]. The ESN partially reflects some features of learning mechanisms in biological brains. It is amazing that such a simple model with completely random reservoir state can approximate such the chaotic systems as Mackey-Glass dynamics [12] and laser time-series [13] with rather high accuracy. Specifically, the accuracy of predicting the Mackey-Glass chaotic time-series with ESN is significantly improved by a factor of 2400 over previous techniques [9], thanks to its rich state reservoir that causes the network to have large memory capacity. Besides, Jaeger gave a formal expression of the network short-term memory capacity [7,14]. He had proven that echo state property holds if and only if spectral radius of reservoir weight matrix is less than 1 [7]. Considering the memory capacity of network may increase with the spectral radius according to Ref. [7]; however, this sufficient condition restricts largely further improvement of approximation capabilities of this model.

Recently, the modified model, called SHESN (scale-free highly-clustered ESN), was proposed by Deng and Zhang [13]. The SHESN model can largely relax the sufficient condition given by Jaeger [7], which means that the

spectral radius of reservoir weight matrix could be much greater than 1 [13]. They designed the network to generate a naturally evolving state reservoir according to incremental growth rules that account for the following features: 1) short characteristic path length, 2) high clustering coefficient, 3) scale-free distribution, and 4) hierarchical and distributed architecture. In the reservoir of SHESN, neurons are divided into different domains, each of which contains one backbone neurons and lots of local ones. This network structure reflects spatial distribution features of biological brain neurons, and has some natural characteristics of biological neuronal systems in many aspects, such as power law [15,16], small-world property [17], and hierarchical architecture. In Ref. [13], the authors applied the SHESN to benchmark problems such as Mackey-Glass prediction problem and laser time-series problem as well. The experimental results achieved by SHESN indicate that the prediction accuracy of nonlinear dynamical system is significantly raised.

On the basis of all the aforementioned results, we present an extended SHESN model, called e-SHESN. It modifies the reservoir using leaky integrator neurons and inhibitory connections. By applying our new model to the Mackey-Glass prediction problem, the experimental results show that the e-SHESN, which state reservoir preserves both small-world property and scale-free feature, is capable of enhancing the approximating and predicting capabilities of the network.

The rest of this paper is organized as follows. In Sect. 2, we present an e-SHESN and explore the collective behavior of this new biologically inspired model, including small-world property and scale-free feature. We then apply it to the Mackey-Glass prediction problem and compare the results of e-SHESN to SHESN in the next section. In Sect. 4, we discuss how to improve the stability of the proposed model. Finally, we draw conclusions and provide some open problems.

## 2 Material and method

### 2.1 Network structure

The evolving state reservoir of SHESN is generated according to incremental growth rules. This procedure is described as the following four steps [13].

1) Initialization of a grid plane of a state reservoir: Suppose that the number of internal neurons in a new state reservoir is  $n$ , where  $n = L \times L$ . To achieve power law, the internal neurons with a size of  $n$  are incrementally assigned on a grid plane divided into  $L \times L$  squares by using a stochastic dynamic growth model proposed in Ref. [16]. Note that different internal neurons cannot be placed at the same square and are not allowed beyond the grid plane. For instance, a 1000-unit reservoir whose neurons are placed

on a  $300 \times 300$  grid plane is used in the later experiment.

2) Generation of backbone neurons and synaptic connections associated: We divide all the internal neurons into backbone neurons and local neurons. Approximately 1% of the internal neurons in the state reservoir are backbone ones in our experiment. Specifically, we randomly generate the coordinates of each backbone neuron on the  $L \times L$  grid plane. Furthermore, we define a domain as the set of internal neurons that comprises one backbone neuron and a number of local neurons around this backbone. The spatial distribution of the backbone neurons, however, must satisfy two restrictions. One is that different backbone neurons are not allowed to be placed at the same square of the grid plane. The other is that the minimum distance between any two backbone neurons must be greater than a certain threshold such that the resulting domains could be separated from each other. For instance, we set this threshold to be 30 in later experiment. After that, the backbone neurons are fully connected to each other that are described later.

3) Incremental growth of new local neurons: For each local neuron, we randomly select one of the backbone neurons and put the local neuron into the domain associated with the backbone neuron. The local neurons are placed in the domain according to the bounded Pareto heavy-tailed distribution [18]:

$$P(\rho) = \frac{\gamma k^\gamma}{1 - (k/Q)^\gamma} \rho^{-(\gamma+1)}, \quad k \leq \rho \leq Q, \quad (1)$$

where  $\gamma$  denotes the shape parameter,  $k$  ( $Q$ ) the minimum (maximum) value, and  $\rho$  the distance between the backbone neuron and the local neuron in the same domain. Here, we assign the same value to these parameters that are mentioned in Ref. [5].

4) Generation of the synaptic connections for the new local neuron by using local preferential attachment rules: Based on the preferential attachment rules [15], any newly added local neurons always prefer to connect to the neurons that already have many synaptic connections. More precisely, the probability that a new local neuron is connected to an existing neuron is proportional to the outdegree of the latter one. Typically, a new strategy called a local preferential attachment is adopted to assign the synaptic connection weight, which is described in detail in Ref. [13].

By following the above steps, we can build the reservoir of e-SHESN. For the time-series prediction problems such as Mackey-Glass systems, no input units are attached to the reservoir. Figure 1 shows the topological structure of our e-SHESN model.

### 2.2 Extension of SHESN using leaky integrator neurons and inhibitory connections

Neither SHESN nor ESN could have working memory in

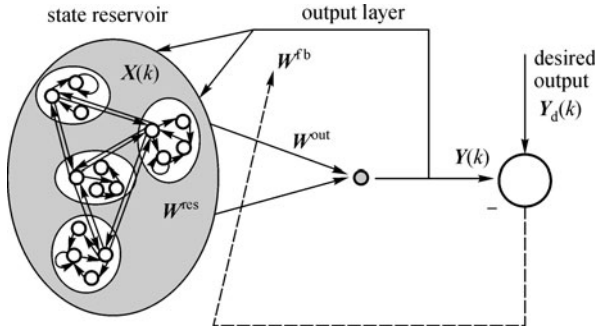


Fig. 1 e-SHESN network with no input units

the reservoir. Their output depends only on their previous value. Thus, according to Jaeger, it is not suitable for the standard sigmoid network to learn the output sequence of slowly and continuously changing dynamical systems [7]. For SHESN, the prediction ability can be further enhanced by maintaining working memory in the reservoir. Great deals are known about the biophysical mechanisms responsible for generating neuronal activity, and these provide a basis for constructing neuron models [19]. Leaky integrator neuron is a simple but nevertheless useful neuron model. By added a “leaky” term, the neuron can get time-independent working memory.

To improve the ability of SHESN for learning those slowly and continuously changing dynamical systems, leaky integrator neurons are added into the state reservoir, which is as similar as the model proposed by Jaeger [7]. For continuous task, the equation of the leaky integrator neuron model is

$$f(W^{\text{res}}X + W^{\text{fb}}Y + V) - \alpha X = \tau \frac{dX}{dt}, \quad (2)$$

where  $\tau$  denotes a time constant,  $\alpha$  the leaky decay rate, and  $V$  a white noise vector added to the activation function of the internal units in the e-SHESN.  $W^{\text{res}}$  represents the reservoir weight matrix and  $W^{\text{fb}}$  is the feedback connection weight one.  $X$  and  $Y$  stand for the state vector of the internal units and the output vector, respectively.  $f$  is the activation function (typically the sigmoid function). Note that there is no input signal in this equation. For discrete time task, the network of e-SHESN should be updated approximately according to

$$X(k+1) = \left(1 - \frac{\delta\alpha}{\tau}\right)X(k) + \frac{\delta}{\tau}f(W^{\text{res}}X(k) + W^{\text{fb}}Y(k) + V(k)), \quad (3)$$

where  $X(k)$  denotes the activation vector of the internal units at time step  $k$  in the updating process, and  $\delta$  the stepsize.

Biological neural networks are generally constructed by

excitatory and inhibitory neurons [20]. Recently, one of the neurophysiology experiments shows that there are approximately 80% excitatory neurons and 20% inhibitory neurons in the biological neural network [21,22]. In the reservoir of e-SHESN, four-fifth of the neurons are designed to be excitatory neurons and the other neurons are inhibitory ones. In addition, we also assign synaptic connection weights for the two kinds of neurons. For instance, an excitatory neuron connects to the other neurons with positive connection weight value sampled randomly from the uniform distribution over  $[0, 0.25]$ , while an inhibitory one over  $[-0.25, 0]$ . Figure 2 shows the spatial distribution of 1000 internal neurons on  $300 \times 300$  grid plane. The reservoir contains 80% excitatory neurons (circle) and 20% inhibitory neurons (cross).

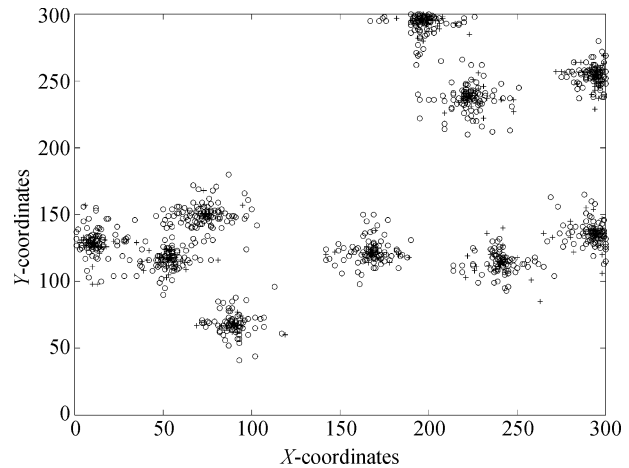


Fig. 2 1000 neurons are placed on a  $300 \times 300$  grid plane according to the naturally incremental growth rules. There are 80% excitatory neurons (circle) and 20% inhibitory neurons (cross) in the reservoir

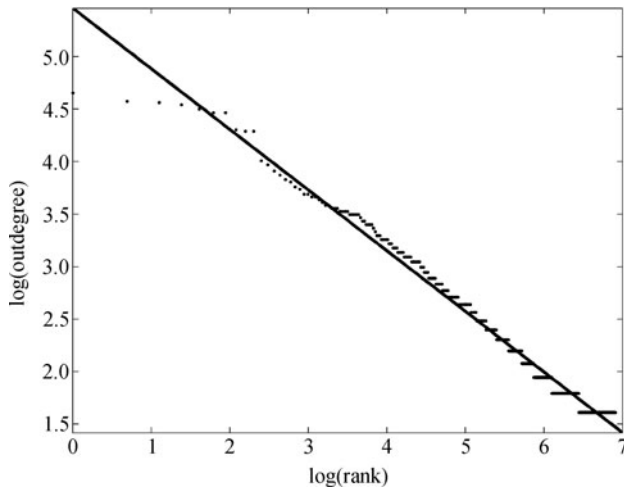
### 2.3 Complexities analysis of reservoir: Keeping small-world property and scale-free feature

The significant contribution in SHESN is that the reservoir has both the small-world property and the scale-free feature. Below is the confirmation that this characteristic still holds for our e-SHESN.

The average characteristic path length and the clustering coefficient are often used to characterize the small-world phenomenon of complex network topology [17]. For the e-SHESN, we make calculation of the sparse connectivity of the state reservoir to be 0.980%. The average characteristic path length and the clustering coefficient of the e-SHESN are computed to be 3.5121 and 0.4238, respectively, which indicates that the e-SHESN maintains the small-world property again.

Then, let us investigate the scale-free feature of the e-SHESN. It is well known that the power laws are free of

any characteristic scale. Networks that have power law degree distributions are called scale-free networks [16]. Correlation coefficient is often used to characterize the scale-free feature. The closer the correlation coefficient is to 1, the closer the structure obeys the power law [16]. To obtain the correlation coefficient of our e-SHESN, we make a fitting linear plot on the log-log plot of neurons rank versus neurons outdegree, both of which are defined in Ref. [13]. As shown in Fig. 3, we make calculation of the correlation coefficient to be 0.9912 with the  $p$ -value equal to 0 and the absolute slope value of the fitting linear plot to be 0.578. Thus, it indicates that the scale-free feature for the e-SHESN holds.



**Fig. 3** Log-log plot of outdegree of neurons versus rank. The correlation coefficient is 0.9912 with a  $p$ -value of 0. The absolute value of fitting linear plot slope is 0.578

### 3 Applying e-SHESN to Mackey-Glass prediction problem

In this section, we examine the performance of the e-SHESN with Mackey-Glass time-series approximation and prediction task. The network used here was prepared by following the procedure that has been introduced in the previous section. The 1000-unit network with single output unit was adopted. The internal neurons were placed on the  $300 \times 300$  grid plane according to the naturally incremental growth rules described above. The network states were updated according to Eq. (3) with parameters  $\delta = 0.1$ ,  $\tau = 2.5$ , and  $\alpha = 0.85$ . The spectral radius of the reservoir weight matrix was set to 2.1, which is the same as the value in Ref. [13]. The feedback connection matrix  $\mathbf{W}^{\text{fb}}$  was sampled randomly from the uniform distribution over  $[-0.5, 0.5]$ . To compare the result with that of SHESN, a noise signal  $\mathbf{V}(k)$  randomly generated from the uniform distribution over  $[-0.0008, 0.0008]$  was fed into the reservoir.

#### 3.1 Data set preparations and test criteria

The differential equation of Mackey-Glass system is given below:

$$\frac{dx}{dt} = \frac{0.2x(t-\tau)}{1+x(t-\tau)^{10}} - 0.1x(t), \quad (4)$$

where  $x$  represents a state, and  $\tau$  a time delay. For comparison to SHESN, data sets 1 and 2 that were described in Ref. [13] were employed here. Each data set had 4000 points. Table 1 lists the number of points and different activation functions used in each data set. According to the output range of sigmoid function used by the output unit, the absolute value of the network output cannot be greater than 1. Thus, the sequences are then shifted and squashed into  $[-1, 1]$  by a transformation  $y \mapsto f(y)$  in order to calculate the output weight matrix.

**Table 1** Data sets for Mackey-Glass sequence approximation and prediction

name	$\tau$	points	transformation
data set 1	17	4000	$\tanh(x-1)$
data set 2	30	4000	$0.3\tanh(x-1)+0.2$

For convenience, we used the first 3000 points of each data set to train the network, respectively. The other 1000 points of each data set were used for testing. During the training phase, the first 1000 steps were discarded for initial transient and the output matrix was calculated with the remaining 2000 steps.

In the test phase, the criterion used here is to evaluate the accuracy of the network for all points in the data sets. Actually, the average mean-square-error (MSE) for all the points were computed after accomplishing each 100 independent trails for SHESN and e-SHESN, respectively.

#### 3.2 Training e-SHESN by supervised learning

As described in Ref. [7], the output weight matrix is adjusted just by a simple linear regression during the training phase. For the update of the output weight, we used the generalized inverse matrix approach, which is the same to Ref. [13].

Since there are no input units attached to the reservoir, the feedback signal must be “teacher-forced” during the learning phase. Otherwise, the network must “guess” the sequence using its own actual output, which may result in the occurrence of an unstable state reservoir. Thus, during the training phase, Eq. (3) can be rewritten as follows:

$$\begin{aligned} \mathbf{X}(k+1) &= \left(1 - \frac{\delta\alpha}{\tau}\right)\mathbf{X}(k) \\ &+ \frac{\delta}{\tau} \cdot \mathbf{f}\left(\mathbf{W}^{\text{res}}\mathbf{X}(k) + \mathbf{W}^{\text{fb}}\mathbf{Y}_d(k) + \mathbf{V}(k)\right), \quad (5) \end{aligned}$$

where  $Y_d(k)$  is the vector of teacher signals at time step  $k$ .

Compared to  $1.13 \times 10^{-5}$  and  $1.31 \times 10^{-2}$ , we achieved the average training MSEs of  $1.20 \times 10^{-5}$  for data set 1 and  $1.17 \times 10^{-3}$  for data set 2, respectively, after completing 100 independent trails.

### 3.3 Approximation of Mackey-Glass sequence

We applied the e-SHESN to the Mackey-Glass sequence approximation task. Note that the network state is still updated according to Eq. (5), which means that the feedback is still “teacher-forced”. After 100 independent trials, the average MSEs for data sets 1 and 2 were  $1.21 \times 10^{-3}$  and  $1.14 \times 10^{-5}$ , respectively, which compare to  $2.1 \times 10^{-3}$  and  $1.4 \times 10^{-4}$  obtained by SHESN. We can observe that the performance of the e-SHESN is better than SHESN. Figure 4 shows the 500-step subsequences of the network output sequences and the teacher sequences from 2000 to 2500 after completing one of the 100 independent trials. Data sets 1 and 2 were used in Figs. 4(a) and 4(b), respectively. From Fig. 4, we can see that the sequences are matched quite well in shapes, except some points, which indicates that e-SHESN has also the remarkable ability of approximating dynamics similar to SHESN.

### 3.4 Prediction of Mackey-Glass sequence

The ability of chaotic prediction is one of the most important characteristics of SHESN (and ESN). During training phase, the network can “remember” the chaotic attractor of dynamics due to rich states of the reservoir [7]. Here, we consider the Mackey-Glass sequence prediction task. Before doing this task, Eq. (5) should be modified because there is no feedback from the teacher sequences in

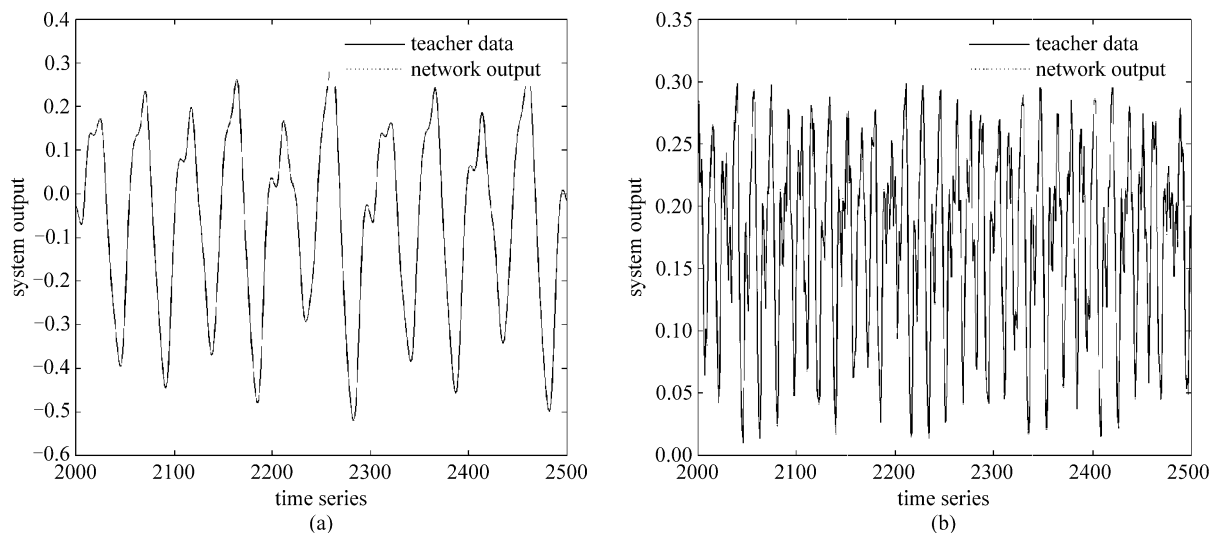
the prediction task, which means that the network should run on itself without teacher sequence. Thus, we fed the network output signal back into the reservoir instead of the teacher sequence. More precisely, the network state should be updated according to Eq. (3) during the prediction phase.

Figure 5 shows the 300-step subsequences of the network output and the teacher subsequences from steps 3000 to 3300 after one of 100 independent trials were done. Data sets 1 and 2 were used in Figs. 5(a) and 5(b), respectively. The average MSE for data sets 1 and 2 were  $9.43 \times 10^{-2}$  and  $1.22 \times 10^{-1}$ , respectively. It can be observed from Fig. 5(a) that the sequences match quite well in shapes. For data set 2, the profiles of both generally make the same, and the error is acceptable.

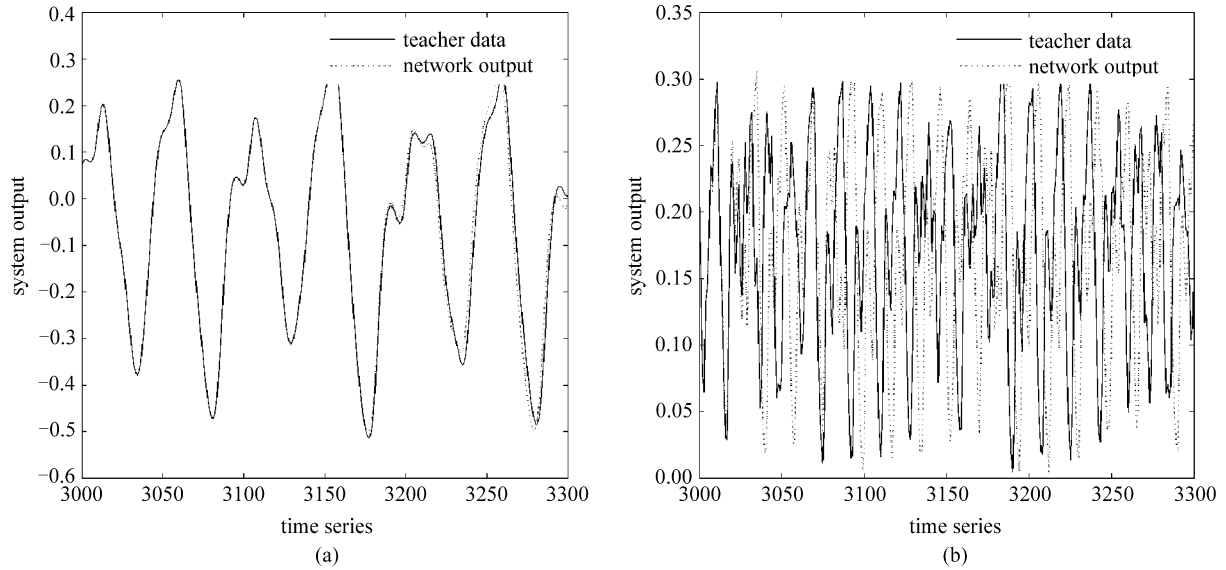
It can be concluded that the reservoir can preserve such a chaotic sequence in a network way; however, the ability degrades when the complexity of chaotic system dynamics increases. From another perspective, we can investigate the attractors of the predicted sequence. We revealed the dynamics of attractors by plotting the trajectory of the point sets  $(Y_d(k), Y_d(k+5))$ . Figure 6 shows the attractors that were obtained from the teacher sequence and network output sequence when the time delay was 30. It is obvious that the network could still recall the attractor of the dynamics if the system is rather chaotic, which demonstrates that the e-SHESN has remarkable capability of predicting chaotic sequence.

## 4 Discussion

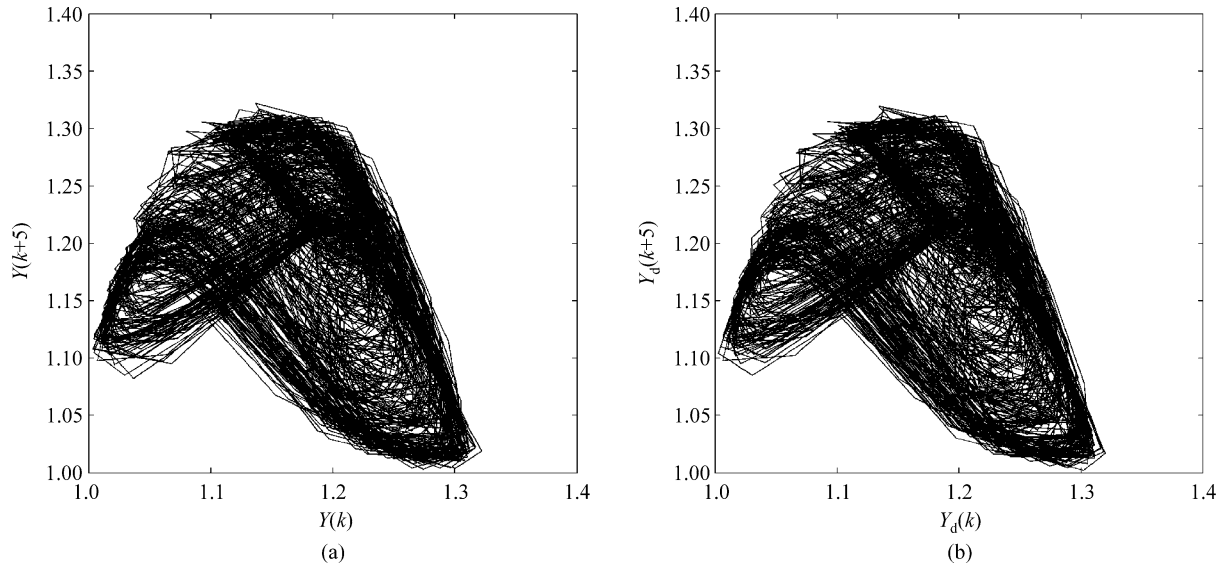
From the Mackey-Glass sequence task, we can see the remarkable capability of approximating and predicting



**Fig. 4** 500-step teacher subsequence (solid) and training network output subsequences (dashed). (a) e-SHESN runs with data set 1; (b) e-SHESN runs with data set 2



**Fig. 5** 300-step teacher subsequence (solid) and training network output subsequences (dashed). (a) e-SHESN runs with data set 1; (b) e-SHESN runs with data set 2



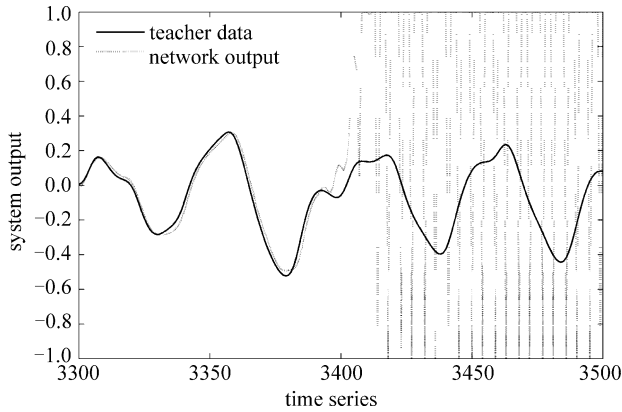
**Fig. 6** Attractors of teacher and network output sequence. Network runs with data set 2. (a) Attractor of teacher sequence; (b) attractor of network output sequence

complex dynamical systems using the e-SHESN. With no formal representation, we can conclude that this excellent ability is due to the leaky integrator neurons that are added into the state reservoir. The reason is that the integrator neurons can preserve time-independent working memory. This memory gives the network ability to recall the complex dynamics such as chaotic attractor of a dynamical system.

When we apply the SHESN to the prediction task, however, the network output sequence will become unstable in some of the independent trials. Figure 7 shows one of these kinds of cases (the network runs on data

set 1), where the curves of the teacher (solid) and the network output (dashed) are apart from each other from time step 3400 on. It can be readily observed that the absolute value of the network output is up to one.

To understand how this case happens, let us consider the structure of the e-SHESN. As no input unit is attached to the reservoir, some signals should be fed into the reservoir in order to maintain the echo state property. More precisely, the network states are mapped to the network output through output weight matrix. As we mentioned before, the network has echo state property if and only if the network state is uniquely determined by any



**Fig. 7** Network output becomes unstable. Output plot (dashed) diverges from actual Mackey-Glass sequence (solid)

left-infinite input sequence [7]. Especially in the e-SHESN, the network input is zero but the network state changes from time to time. This seems to be in conflict with the definition of echo state property. In fact, the feedback acts as the network input in the e-SHESN. However, we cannot feed the network output signals back into the reservoir during training phase because we want the network to “remember” the dynamical system attractor. Otherwise, the network should “guess” the teacher sequence, which could lead to an unstable state reservoir because there are only excitatory connections in the reservoir of SHESN. During the prediction phase, we must feed the network output back into the reservoir because the network must run on itself. As a result, the network may fall into the unstable loop in some cases.

According to Jaeger, the main method to improve the robustness and generalization of reservoir computing is noise injection. This approach actually “simulates” what would happen if a small “mistake” is happened and thus forces the reservoir to be able to recover from its own “mistakes”. This technology can lead an increase of stability of the network; however, the precision of the prediction decreases at the same time when more noises are added into the network. Recently, a new method of calculating output weight matrix is introduced by Wyffels, Schrauwen, and Stroobandt [23]. In their paper, they use

ridge regression to minimize the MSE of the network. According to the canonical ridge regression, the matrix solution of e-SHESN described above is

$$\omega_{\text{ridge}} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{Y}_d, \quad (6)$$

where  $\mathbf{X}$  denotes the reservoir states matrix, and  $\mathbf{Y}_d$  the desired output of the network.  $\lambda$  is the regularization parameter, which determines the allowed readout weights norm. Thus, we should optimize this parameter for each reservoir and cannot reuse or fix to some arbitrary value.

Meanwhile, we guess that the structure of the reservoir can affect the stability of the e-SHESN. We can improve this situation by adding inhibitory connections into the reservoir. That is just what we have done in this paper. Theoretically, inhibitory connections can produce a degenerative effect throughout network state updating. In addition, we guess that only the inhibitory neurons perform a vital role in preventing the network from being unstable, not the leaky integrator neurons.

As a result, we do an extra comparative test on the SHESN and e-SHESN. Table 2 compares the average unstable frequency and the onset of the unstable state of SHESN, SHESN with leaky integrator neurons, and e-SHESN without leaky integrator neurons. All of the three models employ the classic linear regression method to calculate output weight matrices. Apparently, it is the inhibitory neurons that significantly improve the stability or robustness of the network rather than the leaky integrator neurons. Table 3 compares the average unstable frequency and the onset of the unstable state of SHESN, e-SHESN, and e-SHESN without leaky integrating neurons when the ridge regression is used to train the network. From Table 3, the decrease of the network’s instability is more spectacular. Thus, we can conclude that an appropriate regression method can make better performance on increasing the reservoir’s stability than a well-designed structure with noise injected.

## 5 Conclusions

In this paper, we propose an e-SHESN model using leaky

**Table 2** Comparison of the two networks on Mackey-Glass prediction (using linear regression method)

	SHESN	SHESN (with leaky integration neurons)	e-SHESN (without leaky integration neurons)
average frequency	13/50	11/50	2/50
average onset of unstable state	343/1000	352/1000	551/1000

**Table 3** Comparison of the two networks on Mackey-Glass prediction (using ridge regression method)

	SHESN	e-SHESN	e-SHESN(without leaky integration neurons)
average frequency	0/50	0/50	0/50
average onset of unstable state	0/1000	0/1000	0/1000

integrator neurons and inhibitory connections, inspired by neurophysiology evidences. The addition of leaky integrator neurons enhances abilities of the network to have integration of information in the phase of network state propagation. Meanwhile, both excitatory neurons and inhibitory ones, which proportion is kept to be 4:1 according to neuroanatomical results, are introduced into the reservoir of our e-SHESN. The investigation of biological characteristics of the e-SHESN indicates that both the small-world property and the scale-free feature remain preserved. For the Mackey-Glass prediction problem, the experimental results show that the e-SHESN considerably outperforms the SHESN in capabilities of approximating and predicting complex dynamical systems like chaotic timeseries. In other words, the strange attractors of the Mackey-Glass chaotic timeseries are precisely kept in the network reservoir. Interestingly, we also reveal the flaw in the network structure of the original SHESN, which may lead to an unstable state in the SHESN. It is by adding inhibitory neurons that our e-SHESN is capable of improving the network stability because it seems to result in a degenerative effect during the network state propagation. Additionally, the stability of e-SHESN could be significantly enhanced by using the ridge regression instead of the linear regression. There still remain open problems. For example, it is required to have strict theoretic proof for the increased network memory capacity by adding leaky integrator neurons into the state reservoir. Moreover, it is unclear why it happens with the unstable state in SHESN. What is the reason that the e-SHESN can avoid the unstable states occurred. Finally, research along this line and practical applications to other complex dynamics problems are in progress.

**Acknowledgements** The authors would like to thank Zhenbo CHENG for helpful discussion. This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 90820305 and 60775040).

## References

- Seung H S. Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. *Neuron*, 2003, 40(6): 1063–1073
- Izhikevich E M. Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, 2003, 14(6): 1569–1572
- Pavlidis N G, Tasoulis D K, Plagianakos V P, Vrahatis M N. Spiking neural network training using evolutionary algorithms. In: *Proceedings of IEEE International Joint Conference on Neural Networks*. 2005, 4: 2190–2194
- Oja M, Kaski S, Kohonen T. Bibliography of self-organizing map (SOM) papers: 1998–2001 addendum. *Neural Computing Surveys*, 2002, 3(1): 1–156
- Kohonen T. *Self-Organization and Associative Memory*. 3rd ed. New York, NY: Springer-Verlag, 1989
- Bodén M. A guide to recurrent neural networks and backpropagation. *SICS Technical Report T2002:03*, 2002
- Jaeger H. The “echo state” approach to analyzing and training recurrent neural networks. *GMD Technical Report 148*, 2001
- Maass W, Natschläger T, Markram H. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 2002, 14(11): 2531–2560
- Jaeger H, Haas H. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 2004, 304(5667): 78–80
- Fette G, Eggert J. Short term memory and pattern matching with simple echo state networks. *Lecture notes in Computer Science*, 2005, 3696: 13–18
- Jaeger H. A tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the “echo state network” approach. *GMD Report 159*, 2002
- Mackey M C, Glass L. Oscillation and chaos in physiological control systems. *Science*, 1977, 197(4300): 287–289
- Deng Z D, Zhang Y. Collective behavior of a small-world recurrent neural system with scale-free distribution. *IEEE Transactions on Neural Networks*, 2007, 18(5): 1364–1375
- Jaeger H. Short term memory in echo state networks. *GMD Technical Report 152*, 2002
- Barabasi A L, Albert R. Emergence of scaling in random networks. *Science*, 1999, 286(5439): 509–512
- Medina A, Matta I, Byers J. On the origin of power laws in Internet topologies. *ACM SIGCOMM Computer Communication Review*, 2000, 30(2): 18–28
- Watts D J, Strogatz S H. Collective dynamics of ‘small-world’ networks. *Nature*, 1998, 393(6684): 440–442
- Crovella M, Harchol-Balter M, Murta C. Task assignment in a distributed system: Improving performance by unbalancing load. In: *Proceedings of ACM Conference on Measurement and Modeling of Computer Systems*. 1998, 268–269
- Dayan P, Abbott L F. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. Cambridge: The MIT Press, 2001
- Kohonen T. *Self-Organization and Associative Memory*. 3rd ed. New York, NY: Springer-Verlag, 1989
- Connors B W, Gutnick M J. Intrinsic firing patterns of diverse neocortical neurons. *Trends in Neurosciences*, 1990, 13(3): 99–104
- Abbott L F. Lapicque’s introduction of the integrate-and-fire model neuron (1907). *Brain Research Bulletin*, 1999, 50(5–6): 303–304
- Wyffels F, Schrauwen B, Stroobandt D. Stable output feedback in reservoir computing using ridge regression. *Lecture Notes in Computer Science*, 2008, 5163: 808–817