

Computational Performance of Echo State Networks with Dynamic Synapses

Ryota Mori¹(✉), Gouhei Tanaka^{1,2,3}, Ryosho Nakane², Akira Hirose²,
and Kazuyuki Aihara^{1,2,3}

¹ Graduate School of Information Science and Technology,
The University of Tokyo, Tokyo 113-8656, Japan
{mori,gouhei,aihara}@sat.t.u-tokyo.ac.jp

² Graduate School of Engineering, The University of Tokyo, Tokyo 113-8656, Japan
nakane@cryst.t.u-tokyo.ac.jp, ahirose@ee.t.u-tokyo.ac.jp

³ Institute of Industrial Science, The University of Tokyo, Tokyo 153-8505, Japan

Abstract. The echo state network is a framework for temporal data processing, such as recognition, identification, classification and prediction. The echo state network generates spatiotemporal dynamics reflecting the history of an input sequence in the dynamical reservoir and constructs mapping from the input sequence to the output one in the readout. In the conventional dynamical reservoir consisting of sparsely connected neuron units, more neurons are required to create more time delay. In this study, we introduce the dynamic synapses into the dynamical reservoir for controlling the nonlinearity and the time constant. We apply the echo state network with dynamic synapses to several benchmark tasks. The results show that the dynamic synapses are effective for improving the performance in time series prediction tasks.

Keywords: Echo state networks · Reservoir computing · Dynamic synapses · Short-term synaptic plasticity · Time series prediction · Recurrent neural networks

1 Introduction

The echo state network (ESN) is a computational framework for processing time series data, consisting of two parts: the dynamical reservoir (DR) and the readout [1]. The DR is often constructed with sparsely connected recurrent neural networks (RNNs), which play the role to map the input time series into nonlinear spatiotemporal dynamics generated by the DR. The dynamics of the DR is a function of the input history, and therefore, the activation states in the DR contain the information of the input data. The readout is used to make a mapping from the activation states in the DR to the output time series. In the readout, the outputs of the ESN are often created by a linear combination of the activation states of the DR.

The feature of the ESN is that only the connection weights in the readout part are trained. The input connection weights and the internal connection weights

in the DR are all fixed in advance. Therefore, in terms of computation time, the ESN is advantageous compared with the RNNs where all the weights are adjusted [2]. Particularly when the readout is a linear transformation, it is easy to obtain the weights that minimize the difference between the network output sequence and the desired output sequence by any linear regression method.

The ESNs have been successfully applied to a variety of tasks such as time series prediction, system identification, system control, adaptive filtering, noise reduction, function generation, and pattern classification. However, the ESN with the neuron-based reservoir is not good at dealing with slowly changing time series data whose time constant is smaller than that of the neuron unit. A time delay is required to handle the slow dynamics in the sample data. Although a delay line can be realized by connecting neurons in a chain in an unidirectional way, more time delay requires more neurons. Here, to change the nonlinearity and the time constant of the DR in another way, we introduce dynamic synapses into the DR.

Dynamic synapse, also called short-term synaptic plasticity, refers to the synapse in which the efficiency of synaptic transmission changes transiently due to the changes in the calcium concentration and the release of neurotransmitters [3]. The short-term synaptic plasticity persists for only several hundred milliseconds. Mongillo *et al.* theoretically showed that short-term facilitation in the prefrontal cortex is implicated in working memory [4]. Their simulations indicate that a population activity can be reactivated by weak nonspecific excitatory inputs as long as the synapses remain facilitated. Thus, dynamic synapses can store the history of the past neural activities for about one second as the changes in the synaptic transmission efficiency. Therefore, they can process information in accordance with the past neural activity. Hence dynamic synapses may play an important functional role in time series processing in the brain.

In this study, we propose the ESN with dynamic synapses and investigate its computational performance in time series prediction and memory capacity. We perform numerical experiments on several benchmark tasks to evaluate the effectiveness of the dynamic synapses in information processing. This research is important for understanding the dynamical behavior of the brain as well as realizing bio-inspired / energy-efficient information processing systems.

2 Methods

2.1 Models

(1) Echo State Network. In this study, we use the ESN with dynamic synapses. Our ESN consists of K input layer neurons $\mathbf{I}(t) = [I_i(t)]_{1 \leq i \leq K}$, randomly connected N hidden layer neurons $\mathbf{H}(t) = [h_i(t)]_{1 \leq i \leq N}$ (dynamical reservoir), and L output layer neurons $\mathbf{Y}(t) = [y_i(t)]_{1 \leq i \leq L}$. The $N \times K$ input weight matrix $\mathbf{W}^{in} = [w_{i,j}^{in}]$ is created as a uniform random matrix in the range $[-0.5, 0.5]$, and the $N \times N$ internal weight matrix $\mathbf{W} = [w_{ij}]$ is created as a uniform sparse random matrix in the range $[-0.5, 0.5]$. \mathbf{W} is normalized by the spectral radius represented as $\alpha_{sd} (< 1)$. We set the sparseness of the internal

weight matrix \mathbf{W} at 15%. The reservoir state $\mathbf{H}(t)$ is generated by the following difference equation:

$$\mathbf{H}(t+1) = \mathbf{f}(\mathbf{W}^{in}\mathbf{I}(t+1) + \mathbf{W}\mathbf{H}(t)), \quad (1)$$

where \mathbf{f} denotes the component-wise application of the unit's activation function f . We use the sigmoid function $f(s) = 1/(1 + \exp(-s))$, the hyperbolic tangent $f(s) = \tanh(s)$, and the linear function $f(s) = s$. The output $\mathbf{Y}(t)$ is generated by the following difference equation:

$$\mathbf{Y}(t+1) = \mathbf{f}^{out}(\mathbf{W}^{out}\mathbf{H}(t)), \quad (2)$$

where \mathbf{W}^{out} is an $L \times (K + N + L)$ output weight matrix calculated by using input-output training data pairs [2]. We use the output activation function $f^{out}(s) = \tanh(s)$.

(2) Dynamic Synapse. The short-term plasticity of dynamic synapses is caused by quantitative alteration of the releasable neurotransmitters and the calcium concentration [5,6]. The dynamics of dynamic synapses are described by the following two equations for the variables x_i representing the ratio of the releasable neurotransmitters and u_i representing the calcium concentration of neuron i ($i = 1, \dots, N$):

$$x_i(t+1) = x_i(t) + \frac{1 - x_i(t)}{\tau_D} - x_i(t)u_i(t)h_i(t), \quad (3)$$

$$u_i(t+1) = u_i(t) + \frac{U_{se} - u_i(t)}{\tau_F} + U_{se}(1 - u_i(t))h_i(t), \quad (4)$$

where τ_D and τ_F are time constants for the dynamics of x_i and u_i , respectively. If no action potential comes to the presynaptic terminal, x_i and u_i recover to the steady state level 1 and U_{se} , respectively. Here, the efficiency of synaptic transmission is proportional to $x_j(t)u_j(t)$. Therefore, when we innovate dynamic synapses, the strength of the connection from the j th neuron to the i th neuron is redefined as $D_{ij}(t) = w_{ij}x_j(t)u_j(t)/U_{se}$. This standard dynamic synapse model requires $0 < h_i$. Therefore, this model cannot use tanh units and linear units, and instead we use the sigmoid units in this model.

The standard dynamic synapse model reproduces faithfully the experimental results. However, this model is not necessarily suitable for the neural network. For example, (1) this dynamic synapse model is not superset of the static synapse. (2) Dynamics of synaptic efficacy is not monotonic. (3) Dynamic synapse is complex because it includes two variables. To solve these problem, I devised a new univariate dynamic synapse model as follows:

$$e_i(t+1) = e_i(t) + \frac{1 - e_i(t)}{\tau_i} + a_i(e_i(t) - m_i)(M_i - e_i(t))h_i(t), \quad (5)$$

where $e_i(t)$ is the synaptic efficacy of neuron i , and a_i determines the rate of change of synaptic efficacy, m_i and M_i are the minimum and maximum values of

synaptic efficacy, respectively ($0 < m_i < 1$, $1 < M_i$). Now w_{ij} can be described by $D_{ij}(t) = w_{ij}e_j(t)$. In this model, if $a_i > 0$, then the short-term facilitation occurs. If $a_i < 0$, then the short-term depression occurs. If $a_i = 0$, then synapses become static. This model can use tanh units.

2.2 Tasks

(1) Memory Capacity. In order to evaluate the short-term memory capacity of ESN with dynamic synapses, we calculated the Memory Capacity (MC) [2, 7] of the network. We consider an ESN with a single input unit $I(t)$ and many output units $\{y_k(t); k = 1, 2, \dots\}$. The input $I(t)$ is a random signal generated by sampling from a uniform distribution in the interval $[-0.5, 0.5]$. Training signal $d_k(t) = I(t-k)$ are delayed versions of input signal $I(t)$. Memory Capacity (MC) of an ESN is defined as follows:

$$\text{MC} = \sum_{k=1}^{\infty} \max_{\mathbf{w}^{out}} r^2(I(t-k), y_k(t)), \quad (6)$$

where

$$r^2(I(t-k), y_k(t)) = \frac{\text{cov}^2(I(t-k), y_k(t))}{\sigma^2(I(t-k))\sigma^2(y_k(t))} \quad (7)$$

is the determination coefficient (*cov* denotes the covariance and σ^2 denotes the variance).

(2) NARMA 10 Time Series. NARMA (Nonlinear autoregressive moving average) is a generalized version of the autoregressive moving average model, where the regression is nonlinear. NARMA is often used in many studies to evaluate the performance of time series processing of RNNs. The NARMA 10 time series, which includes 10 steps time lag, is generated by the following recurrence relation:

$$y(t) = \alpha y(t-1) + \beta y(t-1) \sum_{i=1}^n y(t-i) + \gamma I(t-n)I(t-1) + \delta, \quad (8)$$

where $\alpha = 0.3$, $\beta = 0.05$, $\gamma = 1.5$, $\delta = 0.1$, $n = 10$ [8]. The input $I(t)$ is a signal generated by randomly sampling from a uniform distribution in $[0, 0.5]$. The task is to predict $y(t)$ from $I(t)$.

(3) NARMA 20 Time Series. The NARMA 20 time series includes 20 steps time lag. This is a more difficult task than the NARMA10 because of the longer history. The NARMA 20 has an additional nonlinear transformation by tanh to confine the signal in a finite range. The NARMA 20 is generated by the following recurrence relation:

$$y(t) = \tanh \left(\alpha y(t-1) + \beta y(t-1) \sum_{i=1}^n y(t-i) + \gamma I(t-n)I(t-1) + \delta \right), \quad (9)$$

where $n = 20$ and the rest of conditions are the same as NARMA 10.

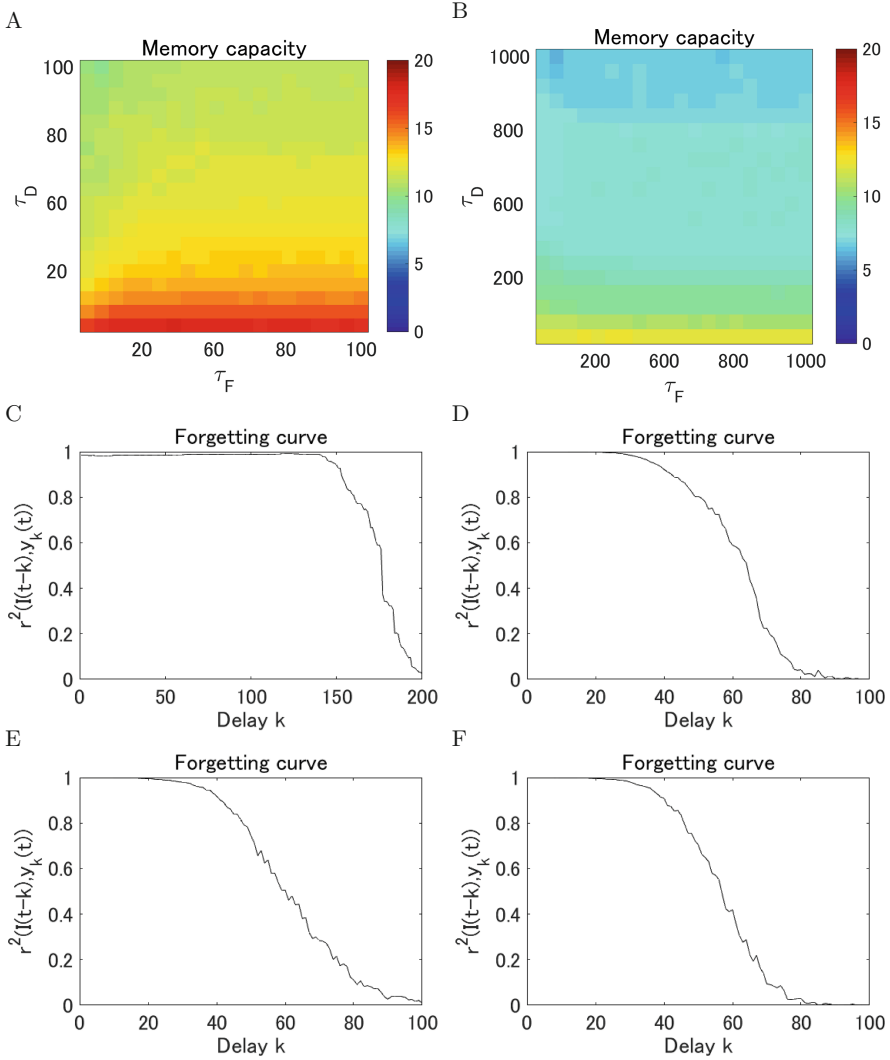


Fig. 1. Comparison of Memory Capacity between the static synapse and the dynamic synapse. We set $N = 400$ and $\alpha_{sd} = 0.99$. We use the standard dynamic synapse model in A–B, and the univariate dynamic synapse model in C–F. **A, B.** Dependence of Memory Capacity on the time constant. We ran 20 trials with different initial weights, and we calculate the average value. We use the standard dynamic synapse model **C**. Forgetting curves of linear DR with static synapses. **D.** Forgetting curves of the tanh DR with static synapses. **E.** Forgetting curves of the linear DR with dynamic synapses. **F.** Forgetting curves of the tanh DR with dynamic synapses.

3 Results

First, the memory capacity of the ESN with dynamic synapses was evaluated in comparison with the ESN without static synapses. In this task, we set $\alpha_{sd} = 0.99$, $N = 400$, and $w_{ij}^{in} \in [-0.1, 0.1]$. The activation function of the neuron units in the dynamic reservoir is given by the linear type, the tanh type or the sigmoid type. Figure 1 shows the results of the numerical experiments on the memory capacity task. Figures 1A and B show how the memory capacity depends on the delay parameters τ_D and τ_F in the dynamic synapses when using sigmoid units. These results show that a smaller value of τ_D gives a larger memory capacity and the value of τ_F is not influential on the capacity. Figures 1C–F show the forgetting curves which indicate how much input history can be embedded in the spatiotemporal dynamics of the dynamic reservoir. As the length of the delay k is increased, the determination coefficient tends to decrease. In this task, we use the univariate dynamic synapse model. We set $a_i \in [0, 0.1]$ (random), $\tau_i \in [1, 20]$ (random), $m_i \in [0, 0.1]$ (random), and $M_i \in [1, 4]$ (random). Figures 1C and D for linear units show that the memory capacity is much decreased by introducing the dynamic synapses. This means that the linearity of the original dynamic reservoir, which is favorable for the transmission of the input data without transformation, is lost by the nonlinearity of the dynamic synapses. In the case of tanh units, the decrease in the memory capacity is relatively small

Table 1. Comparison of the performance between the static synapse and the dynamic synapse. We ran 20 trials with different initial weights, and we calculate the average value and the standard deviation. In these tasks, we use the univariate dynamic synapse model. **A.** Memory Capacity of the ESN. We set $\alpha_{sd} = 0.99$, $N = 400$, $a_i \in [0, 0.1]$ (random), $\tau_i \in [1, 20]$ (random), $m_i \in [0, 0.1]$ (random), and $M_i \in [1, 4]$ (random). **B.** NARMA task. We set $\alpha_{sd} = 0.8$, and use the tanh units. The rest of conditions are the same as those in A.

A

task	activation function	static synapse	dynamic synapse
MC	linear DR	159.3380 ± 8.1009	49.8540 ± 3.0186
	tanh DR	56.1503 ± 2.9915	48.9405 ± 4.7646

B

task	measure		static synapse	dynamic synapse
NARMA10	NRMSE	training	0.13920.0220	0.0768±0.0155
		testing	0.2452±0.0294	0.1541±0.0267
	RMSE	training	0.0204±0.0033	0.01080.0020
		testing	0.0359±0.0049	0.0218±0.0050
NARMA20	NRMSE	training	0.3511±0.0641	0.2997±0.0743
		testing	0.5514±0.0503	0.4966±0.0842
	RMSE	training	0.0148±0.0028	0.0127±0.0033
		testing	0.0232±0.0020	0.0210±0.0040

as shown in Figs. 1E and F. Table 1A summarizes the statistical results for 20 trials in the above four cases. Overall, the dynamic synapses are not effective for improving the memory capacity due to their highly nonlinear property.

Next, NARMA time series prediction performance of the ESN with dynamic synapses was evaluated in comparison with the ESN without static synapses. In this task, we use the univariate dynamic synapse model. We set $\alpha_{sd} = 0.8$, $N = 400$, $w_{ij}^m \in [-0.3, 0.3]$, $a_i \in [0, 0.1]$ (random), $\tau_i \in [1, 20]$ (random), $m_i \in [0, 0.1]$ (random), and $M_i \in [1, 4]$ (random). Table 1B summarizes the statistical results for 20 trials. In the NARMA 10 task, we normalize output to range $[-0.5, 0.5]$. As a result, we found that the dynamic synapse reduce the prediction error about 39.3–47.1% in NARMA 10 and about 9.5–14.2% in NARMA 20. Figure 2 shows the time series of training signals (NARMA 10, NARMA 20) and output signals.

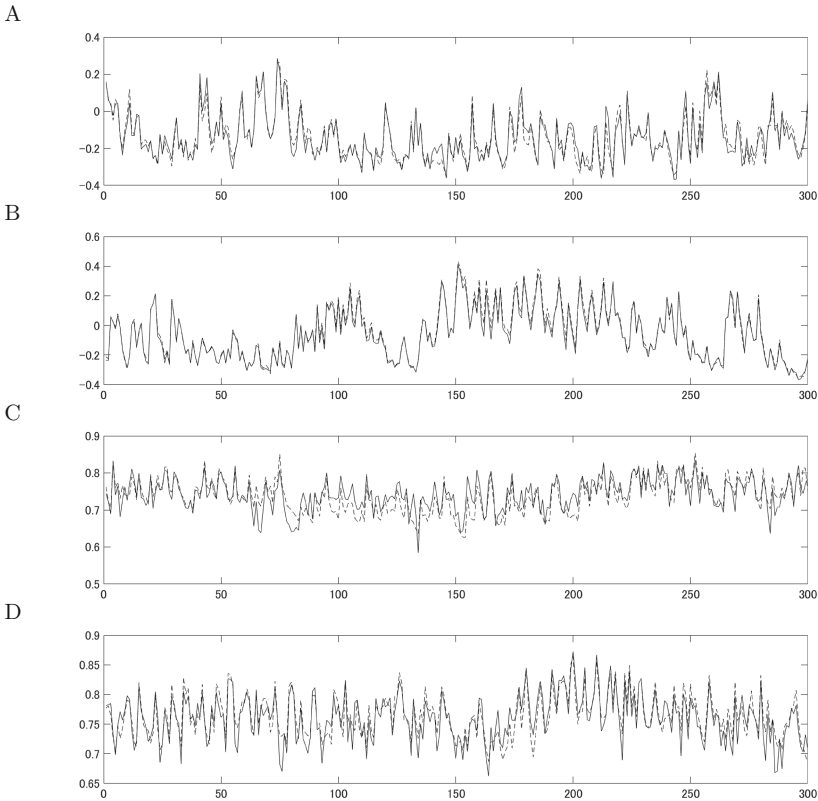


Fig. 2. Training signal (the dotted line) vs Output signal (the solid line). The activation function of the internal units is given by $f = \tanh$. **A.** NARMA10 time series with the static synapses. **B.** NARMA10 time series with the dynamic synapses. **C.** NARMA20 time series with static synapses. **D.** NARMA20 time series with the dynamic synapses

4 Conclusion

We have proposed an echo state network incorporating dynamic synapses which can change the nonlinearity and the time constant of the dynamic reservoir. Numerical experiments were performed to evaluate the effect of dynamic synapses on the computational ability of the echo state network. In the memory capacity task, the dynamic synapses are not effective for improving the performance. This is because the linear dynamics, which is advantageous for the memory capacity task, is broken by the dynamic synapses. In the time series prediction tasks with NARMA 10 and NARMA 20, the dynamic synapses can reduce the prediction error. These tasks require highly nonlinear dynamics, which can be brought about by the dynamic synapses. In this way, the dynamic synapses are suited for relatively difficult tasks that require a dynamical reservoir generating highly nonlinear dynamics. Further numerical experiments for other tasks, however, are necessary to fully reveal the effect of the dynamic synapses on computational ability of the echo state network. In particular, the prediction of time series with slow dynamics is an interesting task to understand how dynamic synapses control the time constant.

Acknowledgments. This work was partially supported by JSPS KAKENHI Grant Number 16K00326 (GT), 26280093 (KA).

References

1. Jaeger, H.: The “echo state” approach to analysing and training recurrent neural networks. Technical Report 148, GMD - German National Research Institute for Computer Science (2001)
2. Jaeger, H.: Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the “echo state network” approach. GMD-Forschungszentrum Informationstechnik (2002)
3. Markram, H., Tsodyks, M.: Redistribution of synaptic efficacy between neocortical pyramidal neurons. *Nature* **382**, 807–810 (1996)
4. Mongillo, G., Barak, O., Tsodyks, M.: Synaptic theory of working memory. *Science* **319**, 1543 (2008)
5. Tsodyks, M., Markram, H.: The neural code between neocortical pyramidal neurons depends on neurotransmitter release probability. *Proc. Natl. Acad. Sci. USA* **94**, 719–723 (1997)
6. Tsodyks, M., Markram, H.: Differential signaling via the same axon of neocortical pyramidal neurons. *Proc. Natl. Acad. Sci. USA* **95**, 5323–5328 (1998)
7. Jaeger, H.: Short term memory in echo state networks. GMD-Report 152, German National Research Institute for Computer Science (2002)
8. Goudarzi, A., Banda, P., Lakin, M.R., Teuscher, C., Stefanovic, D.: A Comparative Study of Reservoir Computing for Temporal Signal Processing, [arXiv:1401.2224v1](https://arxiv.org/abs/1401.2224v1) [cs.NE] (2014)
9. Schrauwen, B., Verstraeten, D., Van Campenhout, J.: An overview of reservoir computing: theory, applications and implementations. In: Proceedings of the 15th European Symposium on Artificial Neural Networks, pp. 471–482 (2007)