Chaotic time series prediction by noisy echo state network

Aren Shinozaki\textsuperscript{1}, Takaya Miyano\textsuperscript{1a)}, and Yoshihiko Horio\textsuperscript{2b)}

\textsuperscript{1} Department of Mechanical Engineering, Ritsumeikan University
1–1–1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan

\textsuperscript{2} Research Institute of Electrical Communication, Tohoku University
2–1–1 Aoba-ku, Sendai, Miyagi 980-8577, Japan

\textsuperscript{a)} tmiyano@se.ritsumei.ac.jp
\textsuperscript{b)} horio@riec.tohoku.ac.jp

Received January 28, 2020; Revised April 25, 2020; Published October 1, 2020

Abstract: We have applied noisy echo state networks to the short-term forecasting of hyperchaotic and chaotic time series. The hyperchaotic time series were generated using the augmented Lorenz equations as a star network of \(Q\) nonidentical Lorenz systems and a four-dimensional Lorenz system. The echo state networks were used mainly in the recursive forecasting mode, wherein the output value of the network, i.e., the predicted value, at the current time step was recursively fed back to the input node at the next time step of prediction. The addition of external noise to the reservoir network has been found to considerably improve the fidelity of the geometrical structures of the chaotic attractors reconstructed from the predicted time series. We discuss these observations on the basis of Ueda’s theory of chaos.

Key Words: reservoir computing, echo state network, hyperchaos, time series analysis

1. Introduction

Reservoir computing is a new approach to constructing recurrent neural networks. In particular, the echo state network (ESN) is a practical method for implementing a reservoir computer \cite{1–3} and has been shown to be effective, for example, for chaotic time series prediction \cite{4–10}. When applied to time series prediction, the ESN usually has three layers, i.e., the input layer consisting of a single node from which input data are generated, the reservoir layer consisting of sparsely coupled multiple nodes (reservoir network), and the output layer consisting of a single node from which predicted values are provided. The sigmoid function and hyperbolic tangent function are often used as the activate function of the reservoir nodes. In the reservoir network, the signals initiated by input data transmit along the closed paths and generate dynamic patterns with fading memory, referred to as long short-term memory. These patterns are used as the library patterns to make predictions at the output node.

A remarkable characteristic of the ESN is that only the weight coefficients (output weight coefficients) for connecting the reservoir nodes to the output node need to be optimized using a supervised learning algorithm. The weight coefficients (input weight coefficients) connecting the input node to...
the reservoir nodes and those between reservoir nodes (reservoir weight coefficients) are initially set to appropriate values and then kept fixed during the training of the output weight coefficients. Remarkably, the input and reservoir weight coefficients need not be updated even when the ESN is applied to other time series. In this sense, the reservoir network appears to be a universal simulator of nonlinear dynamical behaviors.

In a previous report [11], on the basis of the criterion for physical reality proposed by Einstein, Podolsky, and Rosen in their seminal paper on the entanglement of a pair of particles [12], the authors argued that chaotic dynamical behavior might represent a piecewise deterministic process with intervening stochastic processes such as experimental perturbations and numerical roundoff, and hence, might be of no physical reality. Instead, chaotic attractors have physical reality in the sense that their geometrical structures can be reproducibly observed with a probability of unity. These observations appear to be consistent with Ueda’s theory of chaos [13, 14], wherein chaotic dynamical behavior represents a manifestation of random transitions, triggered by experimental perturbations or numerical roundoff, between (infinitely many) unstable periodic orbits accompanying (infinitely many) homoclinic points in the chaotic attractors [15]. Ueda referred to such dynamical behavior as randomly transitional oscillations [13].

Following the above observation, when applying an ESN to chaotic time series prediction, we can expect that random noise added to the reservoir network enhances random transitions between dynamic patterns generated in the network and will improve the degree of fidelity of the chaotic attractor simulated by the network. That is, the addition of random dynamical noise to the reservoir nodes is expected to be effective for intentionally inducing randomly transitional oscillations, which were inferred by Ueda to be the origin of observable chaotic behavior, in the reservoir network.

In this study, we test our expectation by applying a noisy ESN to the short-term time series prediction of chaotic and hyperchaotic time series. In particular, we use the augmented Lorenz (AL) equations [16–18] and four-dimensional (4D) Lorenz equations [19] as hyperchaotic flow models. The AL equations were originally developed as a simplified dynamical model of large-scale circulation in a turbulent Rayleigh–Bénard convective hydrodynamic flow [16] and have recently been applied as a pseudorandom number generator to construct a chaos-based stream cipher. As will be shown in this paper, the AL model consisting of five nonidentical Lorenz subsystems has four positive Lyapunov exponents and generates a high-dimensional hyperchaotic attractor. The 4D Lorenz system was introduced by Wang and Wang and were shown to have two positive Lyapunov exponents. Both hyperchaotic systems inherit the characteristics of the original Lorenz equations, especially the double-scroll structure of its chaotic attractor.

Jaeger has reported that the addition of external noise to the reservoir nodes may be effective in time series prediction by an ESN, although he did not show its reason [1]. Recently, Estébanez et al. conducted an experiment on the nonlinear forecasting of chaotic time series generated by the Rössler equations using a hardware-implemented reservoir computer. They reported that the addition of external noise to the reservoir network was effective for improving the degree of fidelity of the Rössler attractor reconstructed from the predicted time series, although they did not discuss their experimental results in terms of Ueda’s theory of chaos [9]. Similar results obtained using a photonic reservoir computer were reported by Antonik et al. [5]. In this study, we examine the effect of the addition of external noise on the predictive performance of an ESN through numerical experiments and discuss our results in terms of Ueda’s theory of chaos.

2. Mathematical preliminaries for ESN

Given a time series \( \{u(t)\}_{t=0}^{N-1} \) consisting of \( N \) data points, an ESN makes predictions about the values one time step into the future using the following equations.

\[
\begin{pmatrix}
x_1(t) \\
\vdots \\
x_n(t)
\end{pmatrix} = \tanh[U(t)] ,
\]
where

\[
U(t) = \begin{pmatrix}
  w_1^{in} \\
  \vdots \\
  w_n^{in}
\end{pmatrix} u(t) + \begin{pmatrix}
  w_{11} & \ldots & w_{1n} \\
  \vdots & \ddots & \vdots \\
  w_{n1} & \ldots & w_{nn}
\end{pmatrix} \begin{pmatrix}
  x_1(t-1) \\
  \vdots \\
  x_n(t-1)
\end{pmatrix} + \begin{pmatrix}
  w_1^{ofb} \\
  \vdots \\
  w_n^{ofb}
\end{pmatrix} y(t-1) + \begin{pmatrix}
  b_1 \\
  \vdots \\
  b_n
\end{pmatrix},
\]

\[
y(t) = \sum_{i=1}^{n} [w_i^{out1} x_i(t) + w_i^{out2} x_i^2(t)].
\]

Here, \(x_i(t)\) denotes the (output) value at time \(t\) of the \(i\)th reservoir node \(x_i\) with \(i\) from 1 to \(n\), \(n\) denotes the number of the reservoir nodes, \(y(t)\) denotes the output value at time \(t\) of the ESN, \(w_i^{in}\), \(w_{ij}\), and \(w_i^{ofb}\) with \(i\) and \(j\) from 1 to \(n\) are appropriately chosen weight coefficients, and \(b_i\) with \(i\) from 1 to \(n\) represents the bias terms given as pseudorandom numbers subject to a uniform distribution.

We can add a random time series to \(x_i(t)\), although it can be defined using some nonlinear function of \(x_i(t)\) [3]. In this study, \(y(t)\) is given as Eq. (3), as was used in [7]. This form of \(y(t)\) appears to be effective for reducing the prediction error.

ESNs are usually given sparse weight coefficients of \(w_{ij}\). For example, we express the rate of the weight coefficients \(w_{ij}\) having nonzero values as per\(W = k\) with \(0 \leq k \leq 1\). Then, \(100 \times k\) % of a total of \(n \times n\) coefficients have nonzero values and the remaining \(w_{ij}\) values are set to zero. Likewise, we can define per\(W^{in}\), per\(W^{ofb}\), per\(W^{out1}\), per\(W^{out2}\), and per\(B\) for \(w_i^{in}\), \(w_i^{ofb}\), \(w_i^{out1}\), \(w_i^{out2}\), and \(b_i\), respectively. We also denote the rate of \(w_{ij}\) having negative values among those having nonzero values as per\(negW\). For example, \(100 \times k_{-}\times k\) coefficients \(w_{ij}\) have negative values when per\(negW = k_{-}\) with \(0 \leq k_{-} \leq 1\). Likewise, we can define per\(negW^{in}\), per\(negW^{ofb}\), and per\(negB\).

The weight coefficients \(w_{ij}\) are determined so as to satisfy the following conditions.

\[
0 < \rho(W) < 1, \quad \rho(W) = \max(\alpha_i)
\]

(4)

Here, \(\alpha_i\) with \(i\) from 1 to \(n\) is the \(i\)th eigenvalue of the matrix \(W = (w_{ij})\) and \(\rho(W)\) is called the spectrum radius.

The output values \(y(t)\) are interpreted as predictions one time step into the future, denoted as \(\hat{u}(t+1)\), for \(u(t+1)\). That is,

\[
\hat{u}(t+1) = y(t).
\]

(5)

The learning parameters \(w_i^{out1}\) and \(w_i^{out2}\) with \(i\) from 1 to \(n\) are optimized by a recursive least-squares (RLS) method [20] so as to minimize the error functional defined as

\[
E = \sum_{t=1}^{M} \left[ d(t) - \sum_{i=1}^{n} [w_i^{out1} x_i(t) + w_i^{out2} x_i^2(t)] \right]^2 + \gamma \sum_{i=1}^{n} \left[ (w_i^{out1})^2 + (w_i^{out2})^2 \right],
\]

(6)

where \(\{d(t)\}_{t=1}^{M}\) is a time series for training \(w_i^{out1}\) and \(w_i^{out2}\), and \(\gamma\) is a regularization parameter.

In this study, ESNs are used in two forecasting modes, i.e., direct forecasting mode and recursive forecasting mode. In the direct forecasting mode, the input value at a prediction time step of \(t\) is
given as \( u(t) \), which is a standard usage of ESNs for time series prediction. When using an ESN in the recursive forecasting mode, a current output \( \hat{u}(t+1) = y(t) \) at time \( t \) is recursively input to the input node as the next input value at a prediction time step of time \( t + 1 \). The recursive forecasting mode will be mainly used to examine the effect of the addition of external noise to the reservoir network.

3. Numerical experiments and results

The first case study is concerned with a hyperchaotic time series that was generated by numerically integrating the AL equations [16–18] defined as

\[
\begin{align*}
X &= \sigma \left( \sum_{k=1}^{K} \frac{Y_k}{m_k^2} - X \right), \\
\dot{Y}_k &= r_k - m_k X Z_k - Y_k, \\
\dot{Z}_k &= m_k X Y_k - Z_k, \\
\dot{r}_k &= R_0 m_k^2 \Phi_k s_k, \\
\Phi_1 &= \phi - \frac{1}{2} \sin(2\phi), \\
\Phi_{k>1} &= \frac{\sin(m_k - 1)\phi}{m_k - 1} - \frac{\sin(m_k + 1)\phi}{m_k + 1}, \\
s_k &= \sin(m_k\phi),
\end{align*}
\]  

(7) \quad (8) \quad (9) \quad (10) \quad (11) \quad (12) \quad (13)

where \( k = 1, \ldots, Q, Q = 5, \sigma = 25, R_0 = 3185, \) and \( \phi = 0.36 \) [rad]. These parameter settings generate a hyperchaotic time series with four positive Lyapunov exponents \( (\lambda_1 = 1.888, \lambda_2 = 0.868, \lambda_3 = 0.367, \) and \( \lambda_4 = 0.061) \) from which the Kaplan–Yorke (KY) dimension (the Lyapunov dimension) is estimated to be 10.014. A fourth-order Runge–Kutta method was used by the numerical integration to generate a hyperchaotic time series with four positive Lyapunov exponents.

The ESNs are specified by the parameter settings summarized in Table I. For the noisy ESN, \( \xi(t) \) lying between \( -\xi_0 \) and \( \xi_0 \) subject to a uniform distribution were added to the bias terms \( b_i \). In the first case study, \( \xi_0 \) was set to \( 7 \times 10^{-5} \). For the noiseless ESN, \( \xi_0 \) was set 0. The first \( 9 \times 10^5 \) data points were used to optimize \( u_i^{\text{out}1} \) and \( u_i^{\text{out}2} \) of the noiseless and noisy ESNs in the direct forecasting mode using an RLS algorithm with \( \gamma_0 = 1 \times 10^{-6} \). Then, predictions one time step into the future were made for the remaining \( 10^5 \) data points using both the noiseless and noisy ESNs.

When using the ESN in the recursive forecasting mode, the forecasting started at \( t_0 = 9 \times 10^5 \) and subsequently made \( 10^5 \) predictions. At the initial stage of forecasting, an actual value of \( u(t_0) \) was input to the input node, and then the predicted values \( \hat{u}(t+1) (t \geq t_0) \) were recursively input to the input node instead of the actual value \( u(t+1) \). To examine the effect of the addition of the noise terms \( \xi(t) \), we made predictions \( \hat{u}(t+1) \) using both the noiseless and noisy ESNs and compared the predicted structure of the hyperchaotic attractor with the actual structure.

Figure 1 shows a three-dimensional (3D) plot of the actual values of \( X(t), X(t+1), \) and \( X(t+2) \) for the AL equations. It represents a 3D projection of the \((2Q+1)\)-dimensional hyperchaotic attractor. Despite the high dimensional dynamical system having \( 2Q + 1 = 11 \) degrees of freedom, the 3D plot exhibits the double scroll structure similar to that of the original Lorenz system.

Table II summarizes the learning and prediction errors of the noiseless and noisy ESNs. The errors were estimated as the root-mean-square error between \( \hat{u}(t+1) \) and \( u(t+1) \) normalized by the standard deviation of \( u(t+1) \) (NRMSE). For both the noiseless and noisy ESNs, the prediction error is of the same order as the learning error, which implies that the ESNs incur no overtraining. However, the learning and prediction errors of the noisy ESN are considerably larger than those of the noiseless ESN, as expected.

Figure 2 shows the time series \( X(t) \) predicted by the noisy ESN running in the recursive forecasting
Table I. Parameter settings of the ESNs.

| Parameters | Values |
|------------|--------|
| \( n \)   | 200    |
| \( w_{in}^{in} \) | lying between \(-0.8\) and \(0.8\) |
| \( w_{ij} \) | lying between \(-0.1\) and \(0.1\) |
| \( w_{ofb}^{in} \) | lying between \(-0.8\) and \(0.8\) |
| \( w_{out1}^{in} \) | initially set to 0 |
| \( w_{out2}^{in} \) | initially set to 0 |
| \( perW^{in} \) | 0.2    |
| \( perW \) | 0.2    |
| \( perW^{ofb} \) | 0.2    |
| \( perW^{out1} \) | 0.5    |
| \( perW^{out2} \) | 0.5    |
| \( pernegW^{in} \) | 0.2    |
| \( pernegW \) | 0.2    |
| \( pernegW^{ofb} \) | 0.2    |
| \( b_i \) | lying between \(-\xi_0\) and \(\xi_0\) |
| \( perB \) | 0.5    |
| \( pernegB \) | 0.5    |
| \( \rho(w_{ij}) \) | 0.9992 |
| \( \gamma \) | \(\gamma_0\) |
| \( \xi_i(t) \) | 0 or lying between \(-\xi_0\) and \(\xi_0\) |

Fig. 1. 3D plot of actual \(X(t) - X(t + 1) - X(t + 2)\) for the AL equations.

Table II. Learning and prediction errors (NRMSE) in the direct forecasting mode for the chaotic time series.

| Time series | Errors  | noiseless ESN | noisy ESN  |
|-------------|---------|---------------|------------|
| AL, X       | Learning | \(8.15 \times 10^{-6}\) | \(2.86 \times 10^{-4}\) |
|             | Prediction | \(5.89 \times 10^{-6}\) | \(2.88 \times 10^{-4}\) |
| 4D Lorenz, X | Learning | \(2.33 \times 10^{-4}\) | \(7.82 \times 10^{-4}\) |
|             | Prediction | \(1.95 \times 10^{-4}\) | \(7.81 \times 10^{-4}\) |
| Lorenz, X   | Learning | \(6.40 \times 10^{-6}\) | \(2.84 \times 10^{-4}\) |
|             | Prediction | \(7.65 \times 10^{-6}\) | \(2.69 \times 10^{-4}\) |
| Rössler, X  | Learning | \(1.80 \times 10^{-7}\) | \(2.50 \times 10^{-3}\) |
|             | Prediction | \(1.70 \times 10^{-3}\) | \(2.30 \times 10^{-3}\) |

mode and the corresponding actual counterpart. The predicted time series reproduces only the initial part \((0 \leq t < 0.2\), \(t_0\) is reset to 0 in Fig. 2 for convenience) of the actual series and rapidly separates from the actual series at \(t \geq 0.2\) owing to the sensitive dependence on infinitesimal differences in the initial conditions. Nevertheless, the predicted time series as a whole reproduces the geometrical
Fig. 2. Predicted (red traces) and the corresponding actual (black traces) time series for the AL equations. Predictions one time step into the future were made for the $X$ variable using a noisy ESN running in the recursive forecasting mode.

Fig. 3. 3D plot of predicted $X(t) - X(t+1) - X(t+2)$ for the AL equations. Predictions one time step into the future were made for the $X$ variable using a noisy ESN running in the recursive forecasting mode.

For comparison, we used a noiseless ESN running in the recursive forecasting mode and conducted similar time series prediction for the variable $X$ of the AL equations. The predicted time series is totally different from the actual series (Fig. 4). The 3D plot of $X(t) - X(t+1) - X(t+2)$ generated from the predicted time series exhibits a closed loop indicative of a limit cycle, as shown in Fig. 5.

The second case study is concerned with a hyperchaotic time series that was generated by the numerical integration of a 4D Lorenz system [19] defined as

\begin{align}
\dot{X} &= a(Y - X) + W , \\
\dot{Y} &= X - Y - XZ , \\
\dot{Z} &= XY - bZ , \\
\dot{W} &= -YZ + rW ,
\end{align}

where $a$, $b$, $c$, and $r$ are bifurcation parameters set to 10, 8/3, 28, and -1, respectively. These parameter settings yield two positive Lyapunov exponents ($\lambda_1 = 0.3561$ and $\lambda_2 = 0.1565$) from which the KY dimension is estimated to be 3.034.

The numerical integration was conducted in much the same way as in the first case study at a time width of $1 \times 10^{-3}$ with initial conditions of $X = 0.1$, $Y = 0.1$, $Z = 0.1$, and $W = 0.1$. A hyperchaotic time series consisting of $10^6$ data points was obtained at a sampling time step of 10, i.e., a sampling
Fig. 4. Predicted (red traces) and the corresponding actual (black traces) time series for the AL equations. Predictions one time step into the future are made for the $X$ variable using a noiseless ESN running in the recursive forecasting mode.

Fig. 5. 3D plot of predicted $X(t) - X(t+1) - X(t+2)$ for the AL equations. Predictions one time step into the future are made for the $X$ variable using a noiseless ESN running in the recursive forecasting mode.

time interval of $1 \times 10^{-2}$.

The ESNs are specified by the parameter settings summarized in Table I with $\xi_0 = 7 \times 10^{-5}$ and $\gamma_0 = 1 \times 10^{-3}$. The first $9 \times 10^5$ data points were used to optimize $w_{\text{out}1}$ and $w_{\text{out}2}$ of the noiseless and noisy ESNs in the direct forecasting mode using an RLS algorithm. Then, predictions of $\hat{u}(t+1)$ were made for the remaining $10^5$ data points using both the noiseless and noisy ESNs. The recursive forecasting was conducted in much the same way as in the first case study.

The predictive performance of direct forecasting is summarized in Table II. Similar results to those in Fig. 1 to Fig. 5 are shown in Figs. 6–10, respectively. Note that the noisy and noiseless ESNs share the same reservoir network as that used in the first case study.

As has been observed in the case of the AL equations, both the noiseless and noisy ESNs incur no overtraining, and the time series predicted by the noisy ESN running in the recursive forecasting mode reproduces the geometrical structure of the hyperchaotic attractor with sufficient fidelity, whereas the noiseless ESN cannot reproduce it and instead yields a closed loop indicative of a limit cycle, as shown in Fig. 10.

The third example is a chaotic time series generated by the numerical integration of the original Lorenz equations:

\begin{align*}
\dot{X} &= \sigma(Y - X), \\
\dot{Y} &= RX - Y - XZ,
\end{align*}

(18) 
(19)
Fig. 6. 3D plot of actual $X(t) - X(t+1) - X(t+2)$ for the 4D Lorenz equations.

Fig. 7. Predicted (red traces) and the corresponding actual (black traces) time series for the 4D Lorenz equations. Predictions one time step into the future were made for the $X$ variable using a noisy ESN running in the recursive forecasting mode.

Fig. 8. 3D plot of predicted $X(t) - X(t+1) - X(t+2)$ for the 4D Lorenz equations. Predictions one time step into the future were made for the $X$ variable using a noisy ESN running in the recursive forecasting mode.

\[
\dot{Z} = XY - bZ ,
\]

(20) where $\sigma$ (the Prandtl number), $R$ (the Rayleigh number), and $b$ are bifurcation parameters set to 10, 28, and 8/3, respectively. These parameter settings yield a single positive Lyapunov exponent ($\lambda_1 = 0.9056$) from which the KY dimension is estimated to be 2.062.
Fig. 9. Predicted (red traces) and the corresponding actual (black traces) time series for the 4D Lorenz equations. Predictions one time step into the future are made for the $X$ variable using a noiseless ESN running in the recursive forecasting mode.

Fig. 10. 3D plot of predicted $X(t) - X(t+1) - X(t+2)$ for the 4D Lorenz equations. Predictions one time step into the future are made for the $X$ variable using a noiseless ESN running in the recursive forecasting mode.

The numerical integration was conducted in much the same way as in the first case study at a time width of $1 \times 10^{-3}$ with the initial conditions of $X = 0$, $Y = -0.01$, and $Z = 9$. A chaotic time series consisting of $10^5$ data points was obtained at a sampling time step of 25, i.e., a sampling time interval of $2.5 \times 10^{-2}$.

The ESNs are specified by the parameter settings summarized in Table 1 with $\xi_0 = 1 \times 10^{-5}$ and $\gamma_0 = 1 \times 10^{-8}$. The first $9 \times 10^4$ data points were used to optimize $w^\text{out1}_i$ and $w^\text{out2}_i$ of the noiseless and noisy ESNs in the direct forecasting mode using an RLS algorithm. Then, predictions of $\hat{u}(t+1)$ were made for the remaining $10^4$ data points using both the noiseless and noisy ESNs. In recursive forecasting, forecasting started at $t_0 = 9 \times 10^4$ and subsequently made $10^5$ predictions of $\hat{u}(t+1)$.

The predictive performance of direct forecasting is summarized in Table II. Similar results to those shown in Figs. 1, 3, and 5 are shown in Figs. 11–13, respectively. The noisy and noiseless ESNs also have the same reservoir network as that used in the first case study.

Both the noiseless and noisy ESNs incur no overtraining, and the time series predicted by the noisy ESN running in the recursive forecasting mode reproduces the geometrical structure of the chaotic attractor with sufficient fidelity, whereas the noiseless ESN yields a closed loop totally different from the actual structure, as shown in Fig. 13.

The fourth example is a chaotic time series generated by the numerical integration of the Rössler equations:
Fig. 11. 3D plot of actual $X(t) - X(t+1) - X(t+2)$ for the original Lorenz equations.

Fig. 12. 3D plot of predicted $X(t) - X(t+1) - X(t+2)$ for the original Lorenz equations. Predictions one time step into the future were made for the $X$ variable using a noisy ESN running in the recursive forecasting mode.

Fig. 13. 3D plot of predicted $X(t) - X(t+1) - X(t+2)$ for the original Lorenz equations. Predictions one time step into the future are made for the $X$ variable using a noiseless ESN running in the recursive forecasting mode.

\begin{align*}
\dot{X} &= -Y - Z, \\
\dot{Y} &= X + aY, \\
\dot{Z} &= b + Z(X - c),
\end{align*}

where $a$, $b$, and $c$ are bifurcation parameters set to 0.2, 0.2, and 5.7, respectively. These parameter settings yield a single positive Lyapunov exponent of $\lambda_1 = 0.071$ from which the KY dimension is estimated to be 2.013.

The numerical integration was conducted in much the same way as in the first case study at a time width of $1 \times 10^{-3}$ with the initial conditions of $X = 0.1$, $Y = 0.1$, and $Z = 0.1$. A chaotic time
series consisting of $10^5$ data points was obtained at a sampling time step of 200, i.e., a sampling time interval of $2 \times 10^{-1}$.

The ESNs are specified by the parameter settings summarized in Table I with $\xi_0 = 1 \times 10^{-6}$ and $\gamma_0 = 1 \times 10^{-5}$. The optimizations of $w_1^{out1}$ and $w_1^{out2}$ of the noiseless and noisy ESNs running in the direct forecasting mode were conducted in much the same way as in the third case study. In direct forecasting, predictions of $\hat{u}(t+1)$ were made for $10^4$ data points using both the noiseless and noisy ESNs. In recursive forecasting, forecasting started at $t_0 = 9 \times 10^4$ and subsequently made $10^5$ predictions of $\hat{u}(t+1)$.

The predictive performance of direct forecasting is summarized in Table II. Similar results to those shown in Figs. 1, 3, and 5 are shown in Figs. 14–16, respectively. The noisy and noiseless ESNs also
have the same reservoir network as that used in the first case study. Similar tendencies to those in
the first to third case studies are seen in Table II and in Figs. 14–16.

4. Discussion

In the case of the hyperchaotic time series generated from the AL equations, despite the high KY
dimension and a large prediction error, the noisy ESN running in the recursive forecasting mode has
been found to yield short-term predictions from which the geometrical structure of the hyperchaotic
attractor is reproduced in the 3D plot with sufficient fidelity. Although such a structure represents
the projection of a high-dimensional (i.e., an 11-dimensional) attractor onto the 3D space, we may
consider that the noisy ESN running in the recursive forecasting mode is capable of capturing the
entire geometrical structure of the hyperchaotic attractor inherent in the AL equations. In contrast,
the noiseless ESN running in the recursive forecasting mode could not reproduce the attractor from
the predicted time series. Similar observations have been obtained for the 4D Lorenz system as a
hyperchaotic flow model.

For the chaotic time series of the original Lorenz and Rössler systems, the noisy ESNs running
in the recursive forecasting mode have also reproduced the geometrical structures of the chaotic
attractors that were reconstructed from the predicted time series, whereas the noiseless ESNs could
not reproduce the attractors.

Thus, we conclude that the external noise added to the reservoir network is effective for reproducing
the hyperchaotic and chaotic attractors despite causing the deterioration of the predictive precision.

About fifty years ago, Ueda and his colleagues found chaos in an electronic circuit simulating a
forced Duffing oscillator [13]. They also conducted a mathematical analysis of the ordinary differential
equation governing the forced Duffing oscillator on the basis of Birkhoff’s theory of dynamical
systems [15]. According to Ueda’s theory of chaos, the dynamical behavior subject to the exact solu-
tions of a chaotic flow may not be observed by actual and/or numerical experiments because of the
sensitive dependence of the chaotic flow on infinitesimal perturbations caused by measurements and
numerical roundoff. Rather, the observed chaotic behavior is ascribed to randomly transitional oscil-
lations between minimal sets such as infinitely many and dense unstable periodic orbits contained in
chaotic attractors. In Ueda’s view of chaos, observed chaotic behavior represents a piecewise determin-
istic process with intervening stochastic processes of external perturbations caused by measurements,
not an entirely deterministic process. Ueda also proposed the mathematical concept of the substan-
tial stability of chaotic attractors, which ensures the reproducible observability of the geometrical
structures of the attractors.

Our numerical results appear to be consistent with Ueda’s theory of chaos. In the reservoir network,
many limit cycle oscillations with long short-term memory can be generated, which may simulate
unstable periodic oscillations that reside in the chaotic attractor. The noise added to the reservoir
nodes is interpreted to act as the external perturbations that cause random transitions between limit
cycle oscillations temporarily induced in the reservoir network. Such randomly transitional oscillations
may simulate the hyperchaotic and chaotic behaviors. In the case of the noiseless ESNs running in
the recursive forecasting mode, the randomly transitional oscillations are not abundantly induced in
the reservoir network. Eventually, the predicted time series cannot yield the attractors.

In this study, the range of random noise and the regularization parameters, i.e., $\xi_0$ and $\gamma_0$, respecti-
vatively, were determined by a trial-and-error method to achieve a good predictive performance since
we have no mathematical algorithm to optimize $\xi_0$ and $\gamma_0$. The present settings of $\xi_0$ and $\gamma_0$
appear to have no clear correlation with the KY dimension nor the Kolmogorov–Sinai entropy defined as
the sum of positive Lyapunov exponents. In terms of Ueda’s theory of chaos, we conjecture that the
optimal value of $\xi_0$ should be related to the fine structure of the chaotic attractor that reflects how
the minimal sets, e.g., unstable periodic orbits, are densely packed in the attractor. However, this
issue has not been resolved and remains to be investigated.

An interesting question might be whether the predictive method proposed in this paper can be used
to distinguish chaos from stochastic processes. We infer that our method may be effective because both
the noisy and noiseless ESNs provide a similar structure of the 3D plot, i.e., an indefinite geometrical
structure, for stochastic processes. The external noise added to the reservoir network has no critical effect on simulating randomly transitional oscillations when the noisy ESN handles a stochastic time series.

5. Conclusion
We applied ESNs running in the recursive forecasting mode to the short-term prediction of hyperchaotic and chaotic time series. The ESNs were found to be capable of reproducing the geometrical structures of the hyperchaotic and chaotic attractors from the predicted time series, although the addition of external noise to the reservoir network deteriorated the predictive precision. These observations were interpreted on the basis of Ueda’s theory of chaos, wherein the added noise is considered to trigger random transitions between unstable oscillations with fading memory in the reservoir network. In a future paper, we will report a method of quantitatively evaluating the degree of fidelity of the predicted geometrical structures of hyperchaotic and chaotic attractors to actual structures.

Acknowledgments
We are thankful to Dr. Claudio Mirasso and Dr. Piotr Antonik for their helpful advice. One of the authors (T. M.) is also thankful to Dr. Yoshisuke Ueda for stimulating discussions. This study was partly supported by JSPS KAKENHI Grant Number JP18H03307.

References
[1] H. Jaeger, “The ‘echo state’ approach to analysing and training recurrent neural networks – with an Erratum note,” German National Research Center for Information Technology GMD Report, no. 148, 2001.
[2] H. Jaeger, M. Lukosevicius, D. Popovici, and U. Siewert, “Optimization and applications of echo state networks with leaky-integrator neurons,” Neural Networks, vol. 20, pp. 335–352, 2007.
[3] M. Lukosevicius and H. Jaeger, “Reservoir computing approaches to recurrent neural network training,” Comput. Sci. Rev., vol. 3, pp. 127–149, 2009.
[4] J. Pathak, Z. Lu, B.R. Hunt, M. Girvan, and E. Ott, “Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data,” Chaos, vol. 27, pp. 121102–1–121102–9, 2017.
[5] P. Antonik, M. Haelterman, and S. Massar, “Brain-inspired signal processor for generating periodic patterns and emulating chaotic systems,” Phys. Rev. Appl., vol. 7, pp. 054014–1–054014–16, 2017.
[6] Z. Lu, B.R. Hunt, and E. Ott, “Attractor reconstruction by machine learning,” Chaos, vol. 28, pp. 061104–1–061104–9, 2018.
[7] J. Pathak, Z. Lu, B.R. Hunt, M. Girvan, and E. Ott, “Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach,” Phys. Rev. Lett., vol. 120, pp. 024102–1–024102–5, 2018.
[8] D. Canaday, A. Griffith, and D.J. Gauthier, “Rapid time series prediction with a hardware-based reservoir computer,” Chaos, vol. 28, pp. 123119–1–123119–12, 2018.
[9] I. Estébanez, I. Fischer, and M.C. Soriano, “Constructive role of noise for high-quality replication of chaotic attractor dynamics using a hardware-based reservoir computer,” Phys. Rev. Appl., vol. 12, pp. 034058–1–034058–10, 2019.
[10] A. Shinozaki, K. Shiozawa, K. Kajita, T. Miyano, and Y. Hori, “Short-term prediction of hyperchaotic flow using echo state network,” The 2019 International Joint Conference on Neural Networks, paper N-20022, 2019.
[11] K. Shiozawa and T. Miyano, “Symbolic diffusion entropy rate of chaotic time series as a surrogate measure for the largest Lyapunov exponent,” Phys. Rev. E, vol. 100, pp. 032221–1–032221–6, 2019.
[12] A. Einstein, B. Podolsky, and N. Rosen, “Can quantum-mechanical description of physical reality be considered complete?” Phys. Rev., vol. 47, pp. 777–780, 1935.
[13] Y. Ueda, The Road to Chaos (Aerial Press, Santa Cruz, 1992).
[14] Y. Ueda, “Basin-filling Peano omega-branches and structural stability of a chaotic attractor,” NOLTA, vol. 5, pp. 252–258.
[15] G.D. Birkhoff, Dynamical Systems (American Mathematical Society, Providence, Rhode Island, 1927).
[16] K. Cho, T. Miyano, and T. Toriyama, “Chaotic gas turbine subject to augmented Lorenz equations,” Phys. Rev. E, vol. 86, pp. 036308–1–036308–12, 2012.
[17] K. Cho and T. Miyano, “Chaotic cryptography using augmented Lorenz equations aided by quantum key distribution,” IEEE Trans. Circuits Syst. I, vol. 62, pp. 478–487, 2015.
[18] K. Cho and T. Miyano, “Design and test of pseudorandom number generator using a star network of Lorenz oscillators,” Int. J. Bifurcation Chaos, vol. 27, pp. 1750184–1–1750184–14, 2017.
[19] X. Wang and M. Wang, “A hyperchaos generated from Lorenz system,” Physica A, vol. 387, pp. 3751–3758, 2008.
[20] W. Liu, I. Park, Y. Wang, and J.C. Príncipe, “Extended kernel recursive least squares algorithm,” IEEE Trans. Signal Process., vol. 57, pp. 3801–3814, 2009.