Reconstructive Reservoir Computing to Detect Anomaly in Time-series Signals

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY-NC-SA 4.0

SUBMISSION DATE / POSTED DATE

30-12-2021 / 05-01-2022

CITATION

Kato, Junya; Tanaka, Gouhei; Nakane, Ryosho; Hirose, Akira (2022): Reconstructive Reservoir Computing to Detect Anomaly in Time-series Signals. TechRxiv. Preprint. <https://doi.org/10.36227/techrxiv.17704682.v1>

DOI

[10.36227/techrxiv.17704682.v1](https://doi.org/10.36227/techrxiv.17704682.v1)

Reconstructive Reservoir Computing to Detect Anomaly in Time-series Signals

Junya Kato, Gouhei Tanaka, *Member, IEEE*, Ryosho Nakane, *Member, IEEE*, and Akira Hirose, *Fellow, IEEE*

Abstract—We propose reconstructive reservoir computing (RRC) for anomaly detection working for time-series signals. This paper investigates its fundamental properties with experiments employing echo state networks (ESNs). The RRC model is a reconstructor to replicate a normal input time-series signal with no delay or a certain delay ($\text{delay} \geq 0$). In its anomaly detection process, we evaluate instantaneous reconstruction error defined as the difference between input and output signals at each time. Experiments with a sound dataset from industrial machines demonstrate that the error is low for normal signals while it becomes higher for abnormal ones, showing successful anomaly detection. It is notable that the RRC models' behavior is very different from that of conventional anomaly detection models, that is, those based on forecasting ($\text{delay} < 0$). The error of the proposed reconstructor is explicitly lower than that of a forecaster, resulting in superior distinction between normal and abnormal states. We show that the RRC model is effective over a large range of reservoir parameters. We also illustrate the distribution of the output weights optimized through a training to discuss their roles in the reconstruction. Then, we investigate the influence of the neuronal leaking rate and the delay time-shift amount on the transient response and the reconstruction error, showing high effectiveness of the reconstructor in anomaly detection. The proposed RRC will play a significant role for anomaly detection in the present and future sensor network society.

Index Terms—Anomaly detection, reconstruction, auto-encoder, reservoir computing, echo state network

I. INTRODUCTION

Modern society seriously requires anomaly detection technology. For example, an automatic alerting system will be widely useful, where a security camera captures an abnormal footprint to generate an alert. Our society needs adaptive techniques using neural networks learning in the artificial intelligence (AI) framework. However, it is usually difficult to obtain abnormal teacher signals for their rare occasion. Thus, effective training methods without them for neural networks are indispensable.

To deal with static signals, some conventional methods use auto-encoder (AE) for anomaly detection [1]–[3]. AE extracts the features useful for anomaly detection out of the signals. For this purpose, they feed identical input and output teacher signals in the training. They observe the changes in

the extracted features. They employ an additional classifier to detect abnormal signals from the feature changes. A few conventional methods use reconstruction errors indicating the difference between the input and output signals. That is, when a reconstruction error is large, they decide that the signal is abnormal [4], [5]. In such static signal processing, there is no idea of time difference in the signals.

However, anomaly detection for time-series signals increases its significance. In factories, for example, it is important to detect abnormal motors by watching their time-series signals such as sound and vibration. In these days, many sensors are attached at the edge of computer networks. Time-series sensor signals are sent to a data center for anomaly detection. Layered neural networks, including deep learning networks, can actually deal with time-series signals by developing them spatially [6], [7]. On the other hand, it is often difficult for a network to process multiple time-series signals obtained from many sensors. This is because the number of neurons should be large, resulting in a long training time. In addition, signal mixing often causes confusion in the training. A vast number of sensor signals would burden the network with heavy traffic. However, deep learning at the edge of network would not be the solution because of the long training time and the huge power consumption [8], [9].

Some targets of anomaly detection may gradually change their own signal property, even though they are still normal. To deal with such targets, engineers should consider the use of on-line training, that is, simultaneous training and testing. Updating along drifting normal signals will enable a neural network to detect sudden abnormal signals quickly. In this situation, fast training is required.

Reservoir computing, including echo state networks (ESNs), is expected to achieve on-line fast training [10], [11]. It is a class of neural networks, which processes time-series signals rapidly with shorter training time than that of fully trainable recurrent neural networks. It can also be implemented directly utilizing physical phenomena [12] such as photonics [13] and spin waves [14]–[16]. In the near future, our society will use reservoir computing with power efficient devices at the edge of sensor networks. Some conventional methods extract important features from signals by using ESNs for the purpose of anomaly detection [17]–[21]. A part of them extract and classify features simultaneously [22], [23]. However, most of such systems need an additional classifier, which spoils the advantage of the short training time in the ESNs [24].

Other conventional methods for anomaly detection train recurrent neural networks or ESNs to forecast future signals [25]–[27]. In general, recurrent neural networks including

This work was supported in part by the New Energy and Industrial Technology Development Organization (NEDO) as a project JPNP16007.

J.Kato, R.Nakane and A.Hirose are with Department of Electrical Engineering and Information Systems, The University of Tokyo, Tokyo 113-8656, Japan. (e-mail: jkato@eis.t.u-tokyo.ac.jp, nakane@cryst.t.u-tokyo.ac.jp, ahirose@ee.t.u-tokyo.ac.jp)

G.Tanaka is with International Research Center for Neurointelligence (IRCIN), The University of Tokyo, Tokyo 113-0033, Japan. (e-mail: gtanaka@g.ecc.u-tokyo.ac.jp)

Fig. 1. Structure of reconstructive reservoir computing (RRC) model for anomaly detection dealing with time-series signals.

ESNs have often been used for forecasting time-series signals [28], [29]. In these methods, a large forecasting error is the sign of anomaly [30]–[33]. Here, we pay attention to the fact that we deal with time-series signals by using a neural network having various amount of delays in itself and then, we should manipulate the temporal relationship between input and output signals explicitly.

This paper proposes reconstructive reservoir computing (RRC) for anomaly detection based on reconstruction errors. The RRC model is a reconstructor to replicate a normal input time-series signal with no delay or a certain delay. We employ ESNs to reconstruct past or present input signals. This is a reconstruction task. When an instantaneous reconstruction error, indicating the difference between the RRC model output and the expected output signal, is over a threshold, we decide that the signal is abnormal. This paper presents that our reconstruction method shows lower error for normal data than conventional forecasting methods. Thus, our RRC method is more suitable for anomaly detection.

The following sections introduce our reconstruction method and experimental settings to deal with time-series data of machine vibration. The experimental results show that our proposal is effective over a large range of parameters. We observe the distribution of output neuron weights optimized in training to discuss their roles. We also investigate the relationship between the leaking rate of neurons and the properties of transient response. The leak rate is influential to the quality of anomaly detection differently in the reconstruction and forecasting methods. We find that the reconstruction method is clearly different from the conventional forecasting methods. This difference is directly caused by the delay, or memory, inherent in the reservoir.

This paper is composed as follows. In Section II, we explain the structure of the proposed system as well as its dynamics of training and anomaly detection. In Section III, we present and discuss experimental results. We conclude this paper in Section IV.

Fig. 2. Temporal relationship between the input signal $s_{in}(n)$ and the teacher signal $s_{in}(n - n_{shift})$ where n_{shift} is the amount of the discrete time shift. When $n_{shift} \geq 0$, the task is reconstruction while, when $n_{shift} \leq -1$, it is forecasting.

II. PROPOSAL OF RECONSTRUCTIVE RESERVOIR COMPUTING FOR ANOMALY DETECTION

A. System structure

Figure 1 displays the structure of our RRC model for anomaly detection. Its forward signal processing is expressed as

$$\mathbf{x}(n) = (1 - \alpha)\mathbf{x}(n-1) + \alpha f_a(\mathbf{W}_{\text{res}}\mathbf{x}(n-1) + \mathbf{W}_{\text{in}}\mathbf{s}_{\text{in}}(n)), \quad (1)$$

$$\mathbf{s}_{\text{out}}(n) = f_a(\mathbf{W}_{\text{out}}\mathbf{x}(n)), \quad (2)$$

$$\mathbf{s}_{\text{in}}(n) \equiv [s_{\text{in}1}(n) \dots s_{\text{in}i}(n) \dots s_{\text{in}I}(n)]^T, \quad (3)$$

$$\mathbf{x}(n) \equiv [x_1(n) \dots x_j(n) \dots x_J(n)]^T, \quad (4)$$

$$\mathbf{s}_{\text{out}}(n) \equiv [s_{\text{out}1}(n) \dots s_{\text{out}k}(n) \dots s_{\text{out}K}(n)]^T, \quad (5)$$

where $[\cdot]^T$ represents transposition of matrices, and the nonlinear function f_a is the activation function working on each element of the vectors. There are multiple terminals in both input and output layers, where I , J and K denote the numbers of the input terminals, the reservoir neurons, and the output neurons, respectively. Let t_0 and $n = 1, 2, \dots, N$ be a unit discrete time and the index for the discrete time, respectively. The input signal $\mathbf{s}_{\text{in}}(n) \in \mathbb{R}^I$ at time $t \equiv nt_0$ is fed to the neurons in the reservoir through a weight matrix $\mathbf{W}_{\text{in}} \equiv [w_{\text{in}ji}] \in \mathbb{R}^{J \times I}$. The weight matrix $\mathbf{W}_{\text{res}} \equiv [w_{\text{res}jj'}] \in \mathbb{R}^{J \times J}$ represents connection weights between neurons in the reservoir, where $\mathbf{x}(n) \in \mathbb{R}^J$ designates neural output signals in the reservoir. Nonlinearity of the activation function f_a converts input signals to higher-dimensional signals $\mathbf{x}(n)$. A leaking rate α decreases the signals remaining in the reservoir neurons. The output layer derives output signals $\mathbf{s}_{\text{out}}(t) \in \mathbb{R}^K$ from the signals of reservoir neurons through a multiplication with the weight matrix $\mathbf{W}_{\text{out}} \equiv [w_{\text{out}kj}] \in \mathbb{R}^{K \times J}$.

B. Optimizing the output layer weights

Only the weight matrix in the output layer is optimized. The other weights are fixed and not trained. Figure 2 shows how to feed input and output teacher signals. We use a normal signal $\mathbf{s}_{\text{in}}(n)$ as the input teacher signal. We also employ $\mathbf{s}_{\text{in}}(n - n_{\text{shift}})$, which is the normal signal with some discrete time shifts n_{shift} , as the output teacher signal. We define n_{shift} in such a manner that a positive n_{shift} corresponds to delay. At every discrete time, we calculate root mean square error (RMSE) $E(n)$, which is 2- norm of a difference between temporary output signals $\mathbf{s}_{\text{out}}(n)$ and the teacher output signals while training, that is,

$$E(n) \equiv \|\mathbf{s}_{\text{out}}(n) - \mathbf{s}_{\text{in}}(n - n_{\text{shift}})\|_2. \quad (6)$$

This is an instantaneous reconstruction error, and we train reservoir to reduce it. The optimized output weights can be obtained by updating them at every discrete time with the gradient method, or if the signals are statistically stationary, the optimized weights can also be calculated as

$$\mathbf{W}_{\text{out}}^{\text{opt}} = f_a^{-1}(\mathbf{S}_{\text{in}}(n_{\text{shift}}))\mathbf{X}^\dagger, \quad (7)$$

where $\mathbf{X} \equiv [\dots \mathbf{x}(n) \dots] \in \mathbb{R}^{J \times N}$ and $\mathbf{S}_{\text{in}}(n_{\text{shift}}) \equiv [\dots \mathbf{s}_{\text{in}}(n - n_{\text{shift}}) \dots] \in \mathbb{R}^{K \times N}$ are the matrices that consist of

Fig. 3. (a) A part of the waveform of a normal data (normal data #00000018) and (b) the spectrogram of the whole normal data (normal data #00000018).

time sequences of $\mathbf{x}(n)$ and $\mathbf{s}_{\text{in}}(n - n_{\text{shift}})$, respectively, $[\cdot]^\dagger$ denotes pseudo inverse, N is the discrete time corresponding to the signal duration, and f_a^{-1} means inverse of f_a working on every element. Here, as Fig. 2 shows, when $n_{\text{shift}} \geq 0$, we train reservoir to generate the past or present input signals by using its internal delay. We call this task reconstruction. On the other hand, when $n_{\text{shift}} \leq -1$, the trained reservoir outputs the future signal. That is the forecasting method, which is conventional. In the next section, we describe the detail of this temporal relationship.

C. Anomaly detection

We detect an abnormal signal as follows. First, we train an ESN. Then, we feed a test input $\mathbf{s}_{\text{in}}(n)$ into input terminals and obtain an output signal $\mathbf{s}_{\text{out}}(n)$ from the output layer. After that, we calculate an instantaneous error $E(n)$ between the input signal $\mathbf{s}_{\text{in}}(n - n_{\text{shift}})$ with a time shift n_{shift} and the output signal $\mathbf{s}_{\text{out}}(n)$. Only when $E(n)$ is over a threshold, we decide that $\mathbf{s}_{\text{in}}(n - n_{\text{shift}})$ is abnormal. We explain the training and the testing in detail below.

Figure 2 displays flows of teacher and test signals, as we have seen above. First, we feed the teacher input signal $\mathbf{s}_{\text{in}}(n)$ and obtain the output signal $\mathbf{s}_{\text{out}}(n)$. We calculate $E(n)$ between the output signal and the teacher input signal $\mathbf{s}_{\text{in}}(n - n_{\text{shift}})$ which has a time shift n_{shift} . We train reservoir to lower this $E(n)$. Since the initial state of the reservoir $\mathbf{x}(0)$

Fig. 4. (a) Input spectrogram (normal data #00000011) and (b) its instantaneous reconstruction error $E(n)$ where 10 [s] long is equal to about $N = 1.6 \times 10^5$ long in discrete time.

affects the output, we discard the corresponding part of $x(n)$ ($n = 1$ to n_0) to avoid this influence in the training.

After the training, we feed normal or abnormal signals for testing. Then, we calculate $E(n)$ between $s_{out}(n)$ and $s_{in}(n - n_{shift})$. Here, when $n_{shift} = 0$, we call the task reconstruction because the input signal $s_{in}(n)$ is used with $x(n-1)$ to produce itself $s_{in}(n)$ as an output. When $n_{shift} \geq 0$, the task is also reconstruction because the target signal of the reconstruction is already echoing in the reservoir. On the other hand, the task is forecasting when $n_{shift} \leq -1$.

Thresholds for anomaly detection depends on situations. If we could use abnormal signals as teacher signals for a binary classification task, the thresholds were set to maximize F-measure calculated as a harmonic average of recall = $\frac{TP}{TP+FN}$ and precision = $\frac{TP}{TP+FP}$ where TP, FN, FP denote true positive, false negative and false positive, respectively. However, we should set the threshold without using abnormal data.

As we mentioned in Section I, normal signals may drift. In this situation, an on-line training is useful practically. We can train the model with an appropriate time constant by the gradient method or the pseudo inverse matrix method. Then, we update the model along drifting normal signals and detect sudden abnormal signals simultaneously.

Fig. 5. (a) Input spectrogram (abnormal data #00000000) and (b) its instantaneous error $E(n)$ where 10 [s] long is equal to about $N = 1.6 \times 10^5$ long in discrete time.

TABLE I
SIGNAL AND PARAMETERS FOR PREPROCESSING

Sampling frequency	16 [kHz]
Length of data	About 10 [s]
SNR	6 [dB]
Window function for STFT	hann window
Window size	256 sample (=16 [ms])
Window shift	1 sample (=62.5 [μ s])
Number of bands(bins)	129

III. EXPERIMENT AND ITS RESULT

A. Experimental settings

For our experiment, we use the MIMII dataset [34], a part of which was used in DCASE (Detection and Classification of Acoustic Scenes and Events) 2020 CHALLENGE Task2, Unsupervised Detection of Anomalous Sounds for Machine Condition Monitoring [35]. This dataset consists of waveforms of machine sound and vibration. Every data is a 10 minute-long waveform, and the sampling rate is 16 kHz. The waveform is composed of not only sound and vibration of machines but also those of circumstances. Each data does not have the same magnitude of sound and vibration. We use a set of fan data, whose signal-to-noise ratio (SNR) is 6 dB, for this experiment. The dataset of fan (6 dB) has 1,015 normal

Fig. 6. (a) Mean and (b) standard deviation of F-measure versus spectral radius and the number of neurons in the reservoir when we adopt the 90-percentile instantaneous reconstruction error $E(n)$ for normal data as a threshold to try the anomaly detections for 10 times of random weight settings.

data and 361 abnormal data. Figure 3(a) shows a part of the waveform of a normal data (normal data #00000018 [34]).

We preprocess each data as follows by using parameters shown in Table I. First, we normalize the waveform so that its mean is 0 and its standard deviation is 1. Next, we obtain a spectrogram by short-time Fourier transform (STFT). Figure 3(b) displays the spectrogram of the whole normal data (normal data #00000018). A unit discrete time t_0 is 1 data sample duration. Thus, $t_0 = 1/(16 \text{ [kHz]}) = 62.5 \text{ } [\mu\text{s}]$. Each data is about 10 [s] long, which is equal to about $N = 1.6 \times 10^5$ long in discrete time.

We determine the default reservoir parameters as follows. We set the number of reservoir neurons $J = 800$, the connectivity of reservoir neurons 5%, and the weights of neurons (except those in the output layer) at random obeying uniform distribution. The spectral radius of $\mathbf{W}_{\text{res}}$ is set at 1.35. We adopt $f_a(u) \equiv \tanh(u) = (e^u - e^{-u})/(e^u + e^{-u})$ as activation function of the neurons. The number of the input terminals is $I = 129$, which is the same as that of frequency bins of the spectrogram. The number of the output neurons is also $K = 129$. The connectivity between the input terminal and the reservoir is full. We set the leaking rate α at unity.

B. Anomaly detection experiment

1) *Training*: Here, we use the normal data (normal data #00000018) as the teacher data. We adopt pseudo inverse matrix method for the training. We start training after feeding the first n_0 signals ($n_0 = 100$ for this experiment) to avoid the

Fig. 7. Weights of output layer neurons, $\mathbf{W}_{\text{out}}$, obtained by the training and represented in various colormap ranges (a) $[-0.1, 0.1]$, (b) $[-0.05, 0.05]$, (c) $[-0.01, 0.01]$ and (d) $[-0.005, 0.005]$ where the vertical axis shows the output neuron number while the horizontal axis the reservoir neuron number.

influence of the initial reservoir state. In this paper, $\mathbf{x}(0) = \mathbf{0}$ for all experiments. The data may become short corresponding to n_{shift} because some signals are discarded where a part of the output signal does not have its counterpart of the teacher signal. We set $n_{\text{shift}} = 1$ except in Section III-E and Section III-F.

2) *Anomaly detection by reconstruction*: After training, we feed another normal data (normal data #00000011 to show a typical case) and calculate $E(n)$. Figure 4 shows (a) the input spectrogram and (b) $E(n)$. The instantaneous error $E(n)$ presents low values. This spectrogram contains high frequency components in the former 1/3 due to environmental sound and vibration, while in the latter 2/3 period they diminish. However, $E(n)$ does not change so much.

Figure 5 displays the results when we feed an abnormal data (abnormal data #00000000). Figure 5(a) and (b) show the input spectrogram and $E(n)$, respectively. As we can see, $E(n)$ has many peaks and its graph lifts off the horizontal axis. Thus, by setting the threshold appropriately for $E(n)$, we can achieve the anomaly detection.

Fig. 8. Instantaneous error $E(n)$ when only the weights $\mathbf{W}_{\text{out}} = [w_{\text{out } kj}]$ having small absolute values of elements ($|w_{\text{out } kj}| < 0.05$) are working (others are replaced to 0) to test (a) normal and (b) abnormal data as well as $E(n)$ only those having large absolute values ($|w_{\text{out } kj}| \geq 0.05$) are working (others are replaced to 0) to test (c) normal and (d) abnormal data.

In the above discussion, we evaluated the instantaneous error $E(n)$. However, when we do not need any instantaneous decision, we can also carry out the anomaly detection by using averaged $E(n)$ for more reliable detection.

C. F-measure and its dependency on the reservoir parameters

Figure 6 shows how the spectral radius and the number of neurons in the reservoir affect F-measure. We try anomaly detection 10 times and calculate the mean and the standard deviation of F-measure. In every experiment, we set different random weights for $\mathbf{W}_{\text{in}} \equiv [w_{\text{in } ji}]$ and $\mathbf{W}_{\text{res}} = [w_{\text{res } jj'}]$. We also adopt the 90-percentile instantaneous reconstruction error of normal data as threshold. In other words, we allow the false positive rate to be 10 %. When the error is over the threshold, we decide that the signal is abnormal.

When F-measure is closer to unity, the performance of anomaly detection is higher. Figure 6(a) shows high F-measure over a large range of the spectral radius (up to 2.0) and the number of neurons in the reservoir (over 100). From Fig. 6(b), we find that the standard deviation increases sharply as the number of the reservoir neurons decreases, in particular under

100. The connection density is 5 %. Thus, when there are 10 neurons in the reservoir, the number of neuron connections is $10 \times 10 \times 0.05 = 5$. This means that there are only 5 non-zero elements in $\mathbf{W}_{\text{res}}$. The number discreteness of the neurons and the randomness of the weights sometimes cause instability in anomaly detection with such a few neurons.

The above mentioned results explicitly show the high stability of the proposed RRC against the reservoir parameters. In the following subsections, we set the spectral radius and the number of neurons at 1.35 and 800, respectively, to investigate the influence of the time shift n_{shift} on the RRC response.

D. Role of the connection weights in the output layer

Figure 7 (colormap) presents the weights in the trained $\mathbf{W}_{\text{out}}$ by color. The vertical axis shows the output neuron number while the horizontal axis the reservoir neuron number. The ranges of the color bars are (a) $[-0.1, 0.1]$, (b) $[-0.05, 0.05]$, (c) $[-0.01, 0.01]$, (d) $[-0.005, 0.005]$, but all the colormaps express the same $\mathbf{W}_{\text{out}}$. That is, we present the colormaps in multiple scales, since the dynamic range is so large. We can observe about 10 red or blue vertical stripes in

Fig. 7(a), which has the largest range of color representation. Thus, these 10 out of 800 neurons in the reservoir contribute greatly to producing the output. As we narrow the range, lower weights appear in Fig. 7(b)-(d). In Fig. 7(d), almost every weights are colored, but still we can see a stripe pattern in the colormap. Thus, some neurons in the reservoir convey signals strongly to the output layer while others do not. This result means self-organization of a structure similar to the bottleneck in a sand-clock-type AE network, widely used for static data, which we discuss in detail below.

Figure 8 shows $E(n)$ when only the elements having small absolute values $|w_{out\ kj}| < 0.05$ are used (others are replaced to 0) to test (a) normal data and (b) abnormal data, where $\mathbf{W}_{out} = [w_{out\ kj}]$. Figure 8 also shows $E(n)$ when the elements with only large absolute values $|w_{out\ kj}| \geq 0.05$ are used (others are replaced to 0) to test (c) normal data or (d) abnormal data. Every $E(n)$ becomes larger than those in Fig. 4(b) and Fig. 5(b). In the case of (a) and (b) (using the non-dominant weights), the floors rise drastically. They are so similar that it is difficult to carry out anomaly detection. Even in the case of (c) and (d) (using the dominant weights), every $E(n)$ lifts up clearly, resulting in almost the same level with larger ripples. This situation is not suitable for anomaly detection anymore.

Comparing these results with those in Fig. 4(b), we can see that even non-dominant weights are helpful to decrease $E(n)$ for normal data. Here, the role of $\mathbf{W}_{out}$ is analyzed experimentally as follows. The stripe pattern in Fig. 7 presents that there are two types of neurons in the reservoir. One is dominant neurons that determine a major part of the output signal. The other is non-dominant neurons that tune the output signal in detail. One possible interpretation is that, with the dominant neurons, the reservoir works like an AE, which usually deals with static data processing. However, the non-dominant neurons are also necessary for time-series data processing. The results suggest that the reservoir needs both types of neurons to reconstruct such time-series data since non-dominant neurons also convey past signals helpful for the time-series reconstruction.

E. Transient response and its dependency on the time shift and the leaking rate

For observing a transient response of the reservoir, we concatenate the teacher data s_{in} (normal data #00000018) and another data t_{in} (either normal or abnormal data) to compose a testing data, which is 20 [s] long because each teacher data and another data is 10 [s] long. Figure 9 shows the flows of signals when we feed this new testing data. In this experiment, we consider variations in the time shift n_{shift} and the leaking rate α . Here, we observe their influence on $E(n)$ especially when the testing signal switches from s_{in} to t_{in} (transient response).

Figure 10 shows the instantaneous reconstruction error $E(n)$ when the leaking rate is (a), (b) $\alpha = 1$, (c), (d) $\alpha = 0.3$, (e), (f) $\alpha = 0.1$ and (g), (h) $\alpha = 0.01$ where $n_{shift} = 0$. In (a), (c), (e) and (g), we use the testing data consisting of the teacher data (normal data #00000018) and the normal data

Fig. 9. Flows of signals when we feed the testing data consisting of the teacher data s_{in} and another data t_{in} in order to observe the transient response when the testing signal changes from s_{in} to t_{in} .

(normal data #00000011). In (b), (d), (f) and (h), the testing data consists of the identical teacher (normal data #00000018) and the abnormal data (abnormal data #00000000). When the leaking rate α is smaller, more past signals remain in the reservoir. Thus, $E(n)$ increases sharply when the test starts ($t = 0$ [s]) (start transient) and the signal switches from s_{in} to t_{in} ($t = 10$ [s]) (switch transient). In $t = 10 - 20$, the floor is higher when we test the abnormal data than the normal data. Thus, it is possible to achieve the anomaly detection.

We change n_{shift} and conduct the experiments with the same testing data consisting of the teacher and the normal data. We focus on $E(n)$ around $t = 10$ [s] ($n = 159744$ in discrete time) where the signal switches. In Fig. 11, (a), (c), (e) and (g) show $E(n)$ at the switch transient. We pay attention to the peak of $E(n)$ to draw peak-value curves versus n_{shift} in (b), (d), (f) and (h). Again we set (a), (b) $\alpha = 1$, (c), (d) $\alpha = 0.3$, (e), (f) $\alpha = 0.1$ and (g), (h) $\alpha = 0.01$. First, we focus on (a), (c), (e) and (g). When we use the conventional forecasting method ($n_{shift} \leq -1$), $E(n)$ rises before the switching ($n = 159744$ in discrete time), remains almost at the same value, and starts decreasing just after the switching ($n = 159745$). On the other hand, when we use

Fig. 10. Instantaneous reconstruction error $E(n)$ when the leaking rate is (a), (b) $\alpha = 1$, (c), (d) $\alpha = 0.3$, (e), (f) $\alpha = 0.1$ and (g), (h) $\alpha = 0.01$ where $n_{\text{shift}} = 0$. In (a), (c), (e), (g), we use the testing data consisting of the teacher data (normal data #00000018) and the normal data (normal data #00000011) while in (b), (d), (f), (h), the testing data consisting of the teacher (normal data #00000018) and the abnormal data (abnormal data #00000000).

Fig. 11. Instantaneous error $E(n)$ around $t=10$ [s] ($n = 159744$ in discrete time) when (a) $\alpha = 1$, (c) $\alpha = 0.3$, (e) $\alpha = 0.1$ and (g) $\alpha = 0.01$ as well as the peak value of $E(n)$ versus n_{shift} when (b) $\alpha = 1$, (d) $\alpha = 0.3$, (f) $\alpha = 0.1$ and (h) $\alpha = 0.01$.

Fig. 12. Instantaneous error $E(n)$ in dB for normal data (orange) and abnormal data (blue) when (a) $n_{\text{shift}} = -1$ (conventional forecasting method) and (b) $n_{\text{shift}} = 15$ (reconstruction method), respectively.

Fig. 13. Difference between means of $20\log_{10}(E(n))$ for normal and abnormal data (see in Fig. 12) versus n_{shift} .

the reconstruction method ($n_{\text{shift}} \geq 0$), $E(n)$ starts increasing from $n = 159745$. In addition, by comparing (b), (d), (f) and (h) where we sweep n_{shift} from -10 to 10, we find that the peak value decreases drastically at $n_{\text{shift}} = 0$, and the value is mostly low for $n_{\text{shift}} \geq 1$. Thus, the transient response is different between the conventional forecasting methods and the reconstruction method. When we try anomaly detection, $E(n)$ at a transient time should be low because both components of testing data are normal data. Thus, the reconstruction method can achieve better anomaly detection than the conventional forecasting methods. We also find that the amount of the time shift n_{shift} resulting in the minimum peak of $E(n)$ becomes larger as α gets smaller.

In Fig. 11, we can see that the duration of high $E(n)$ is related with the types of the input and the expected output signals. In the conventional forecasting methods, the system should generate unknown signal t_{in} before its input signal switches from s_{in} to t_{in} . Thus, $E(n)$ rises just before the input signal switches. After the switching, both the testing signal and the expected output signal become the same signal series t_{in} , and $E(n)$ starts to decrease. When we use the reconstruction method, the situation is different. Even after the input signal switches to t_{in} , the expected output signal is still s_{in} for a while. Thus, right after the input signal switches, $E(n)$ starts to increase. However, when the expected output signal switches to t_{in} , $E(n)$ drops sharply.

In the above results, we find that the difference between the conventional forecasting methods and the reconstruction method is not only the sign of n_{shift} . There is the significant difference in the transient response. In the next subsection, we see another difference that influences the total performance in anomaly detection.

F. Reconstruction and forecasting errors for anomaly detection and their dependency on the time shift

Figure 12 shows the reconstruction error $E(n)$ in dB (i.e. $20\log_{10}(E(n))$) for the normal data (orange) and the abnormal data (blue) when (a) $n_{\text{shift}} = -1$ and (b) $n_{\text{shift}} = 15$, respectively. The leaking rate is $\alpha = 0.01$ in this experiment. Though the corresponding curves in (a) and (b) look very similar, both curves in (b) are placed lower than those in (a), respectively. There, it is notable that the curve for the normal data (orange) in (b) shifts down more largely.

We investigate further the relationship between $E(n)$ and n_{shift} . We sweep n_{shift} to calculate the means of $20\log_{10}(E(n))$ for the normal and abnormal data. Then we calculate the difference between the two means. Figure 13 shows the difference of the means versus n_{shift} . As n_{shift} increases from a negative value (the conventional forecasting methods), the difference of the means increases. We observe the peak of the difference at $n_{\text{shift}} = 22$ (a reconstruction method). We obtain 1.3 dB larger difference when we set $n_{\text{shift}} = 22$ than $n_{\text{shift}} = -1$. This large difference is greatly helpful for anomaly detection because the quality of anomaly detection depends on the small overlap of the probability distributions for normal and abnormal inputs.

IV. CONCLUSION

In this paper, we proposed the reconstructive reservoir computing to effectively achieve anomaly detection for time-series signals. We train the ESN model to obtain an expected output signal identical to its input signal with a certain time shift. By setting an appropriate time shift, we can utilize the past signal because there is time delay inherent in the reservoir. For anomaly detection, we calculate the reconstruction error that shows the difference between the shifted input and the corresponding output signals. When the reconstruction error is larger than a threshold, we decide that the signal is abnormal. We demonstrated that our reconstruction method is different from the conventional forecasting methods. This paper focuses on the time shift between the input and the output signals. We investigated the influence of the time shift and the leaking rate on the reconstruction properties. Through our experiments, we found that the RRC model is effective for anomaly detection over a large range of its parameters. We observed the connection weights in the trained output layer and found that only a few dominant neurons in the reservoir decide a major part of the output signal while other non-dominant neurons tune the detail. Our research also showed that the reconstruction method is better for anomaly detection than the conventional forecasting methods as follows. When we investigate the transient response, the peak value of instantaneous error $E(n)$ becomes larger in the conventional methods than in the reconstruction method. We calculated the means of $E(n)$ for normal and abnormal data. Then, we found that the difference of the means becomes larger when the leaking rate is small and the discrete time shift is a moderate positive number (reconstruction). Our proposal is based on reservoir computing having such an advantage as fast training, resulting in low energy consumption. We can use our methods practically for the edge computing where anomaly detection plays significant roles. Thus, our RRC will be crucial for our sensor-network society in the near future.

REFERENCES

- [1] Y. Xia, X. Cao, F. Wen, G. Hua, and J. Sun, "Learning discriminative reconstructions for unsupervised outlier removal," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [2] A. S. Hashmi and T. Ahmad, "GP-ELM-RNN: Garson-pruned extreme learning machine based replicator neural network for anomaly detection," *Journal of King Saud University-Computer and Information Sciences*, 2019.
- [3] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, "A survey of deep learning-based network anomaly detection," *Cluster Computing*, vol. 22, no. 1, pp. 949–961, 2019.
- [4] B. Thompson, R. Marks, J. Choi, M. El-Sharkawi, H. Ming-Yuh, and C. Bunje, "Implicit learning in autoencoder novelty assessment," in *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)*, vol. 3, 2002, pp. 2878–2883 vol.3.
- [5] S. Chaurasia, S. Goyal, and M. Rajput, "Outlier detection using autoencoder ensembles: A robust unsupervised approach," in *2020 International Conference on Contemporary Computing and Applications (IC3A)*, 2020, pp. 76–80.
- [6] M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, ser. MLSDA'14. New York, NY, USA: Association for Computing Machinery, 2014, p. 4–11. [Online]. Available: <https://doi.org/10.1145/2689746.2689747>
- [7] H. Shao, H. Jiang, H. Zhao, and F. Wang, "A novel deep autoencoder feature learning method for rotating machinery fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 95, pp. 187–204, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0888327017301607>
- [8] S. Hawkins, H. He, G. Williams, and R. Baxter, "Outlier detection using replicator neural networks," in *International Conference on Data Warehousing and Knowledge Discovery*. Springer, 2002, pp. 170–180.
- [9] E. Marchi, F. Vesperini, S. Squartini, and B. Schuller, "Deep recurrent neural network-based autoencoders for acoustic novelty detection," *Computational intelligence and neuroscience*, vol. 2017, 2017.
- [10] 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), Bonn, Germany, Technical Report 148, 13, 2001.
- [11] W. Maass, T. Natschlager, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," *Neural Computation*, vol. 14, no. 11, pp. 2531–2560, 2002.
- [12] G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, and A. Hirose, "Recent advances in physical reservoir computing: A review," *Neural Networks*, vol. 115, pp. 100–123, 2019.
- [13] Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, and S. Massar, "Optoelectronic reservoir computing," *Scientific reports*, vol. 2, p. 287, 2012.
- [14] R. Nakane, G. Tanaka, and A. Hirose, "Reservoir computing with spin waves excited in a garnet film," *IEEE Access*, vol. 6, pp. 4462 – 4469, 2018.
- [15] T. Ichimura, R. Nakane, G. Tanaka, and A. Hirose, "A numerical exploration of signal detector arrangement in a spin-wave reservoir computing device," *IEEE Access*, vol. 9, pp. 72 637–72 646, 2021.
- [16] R. Nakane, A. Hirose, and G. Tanaka, "Spin waves propagating through a stripe magnetic domain structure and their applications to reservoir computing," *Physical Review Research*, vol. 3, p. 033243, September 2021.
- [17] N. Gianniotis, S. D. Kügler, P. Tiño, and K. L. Polsterer, "Model-coupled autoencoder for time series visualisation," *Neurocomputing*, vol. 192, pp. 139–146, 2016, advances in artificial neural networks, machine learning and computational intelligence. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925231216002587>
- [18] L. Sun, B. Jin, H. Yang, J. Tong, C. Liu, and H. Xiong, "Unsupervised EEG feature extraction based on echo state network," *Information Sciences*, vol. 475, pp. 1–17, 2019.
- [19] S. Zhang, Z. Sun, M. Wang, J. Long, Y. Bai, and C. Li, "Deep fuzzy echo state networks for machinery fault diagnosis," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 7, pp. 1205–1218, 2020.
- [20] J. Long, Z. Sun, C. Li, Y. Hong, Y. Bai, and S. Zhang, "A novel sparse echo autoencoder network for data-driven fault diagnosis of delta 3-d printers," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 3, pp. 683–692, 2020.
- [21] R. Liu, B. Reimer, S. Song, B. Mehler, and E. Solovey, "Unsupervised fNIRS feature extraction with CAE and ESN autoencoder for driver cognitive load classification," *Journal of Neural Engineering*, vol. 18, no. 3, p. 036002, 2021.
- [22] R. Fourati, B. Ammar, C. Aouiti, J. Sanchez-Medina, and A. M. Alimi, "Optimized echo state network with intrinsic plasticity for eeg-based emotion recognition," in *International Conference on Neural Information Processing (ICONIP) 2017*. Springer International Publishing, 2017, pp. 718–727.
- [23] R. Fourati, B. Ammar, J. Sanchez-Medina, and A. M. Alimi, "Unsupervised learning in reservoir computing for eeg-based emotion recognition," *IEEE Transactions on Affective Computing*, pp. 1–1, 2020.
- [24] N. Chouikhi, B. Ammar, A. Hussain, and A. M. Alimi, "Bi-level multi-objective evolution of a multi-layered echo-state network autoencoder for data representations," *Neurocomputing*, vol. 341, pp. 195–211, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925231219303145>
- [25] S. Suh, D. H. Chae, H.-G. Kang, and S. Choi, "Echo-state conditional variational autoencoder for anomaly detection," in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 1015–1022.
- [26] E. López, C. Valle, H. Allende, E. Gil, and H. Madsen, "Wind power forecasting based on echo state networks and long short-term memory," *Energies*, vol. 11, no. 3, p. 526, 2018. [Online]. Available: <https://www.mdpi.com/1996-1073/11/3/526>
- [27] T. Kieu, B. Yang, C. Guo, and C. S. Jensen, "Outlier detection for time series with recurrent autoencoder ensembles," in *Proceedings*

- of the Twenty-Eighth International Joint Conference on Artificial Intelligence, *IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 7 2019, pp. 2725–2732. [Online]. Available: <https://doi.org/10.24963/ijcai.2019/378>
- [28] Q. Yang, Y. Zhou, Y. Yu, J. Yuan, X. Xing, and S. Du, “Multi-step-ahead host load prediction using autoencoder and echo state networks in cloud computing,” *The Journal of Supercomputing*, vol. 71, pp. 3037–3053, 2015.
 - [29] Q. Ma, L. Shen, and G. W. Cottrell, “DeePr-ESN: A deep projection-encoding echo-state network,” *Information Sciences*, vol. 511, pp. 152–171, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0020025519309053>
 - [30] A. E. Bezer and A. Hirose, “Proposal of a human heartbeat detection/monitoring system employing chirp z-transform and time-sequential neural prediction,” in *International Conference on Neural Information Processing (ICONIP) 2016*, 2016, pp. 510–516.
 - [31] O. Obst, X. R. Wang, and M. Prokopenko, “Using echo state networks for anomaly detection in underground coal mines,” in *2008 International Conference on Information Processing in Sensor Networks (IPSN 2008)*, 2008, pp. 219–229.
 - [32] Q. Chen, A. Zhang, T. Huang, Q. He, and Y. Song, “Imbalanced dataset-based echo state networks for anomaly detection,” *Neural Computing and Applications*, vol. 32, pp. 3685–3694, 2020.
 - [33] H. Nguyen, K. Tran, S. Thomassey, and M. Hamad, “Forecasting and anomaly detection approaches using LSTM and LSTM autoencoder techniques with the applications in supply chain management,” *International Journal of Information Management*, vol. 57, p. 102282, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S026840122031481X>
 - [34] H. Purohit, R. Tanabe, K. Ichige, T. Endo, Y. Nikaido, K. Suefusa, and Y. Kawaguchi, “Mimii dataset: Sound dataset for malfunctioning industrial machine investigation and inspection,” in *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019.
 - [35] DCASE Community, “DCASE 2020 Challenge,” 2020. [Online]. Available: <http://dcase.community/challenge2020/>