Short Papers

Exponential Adaptive Lag Synchronization of Memristive Neural Networks via Fuzzy Method and Applications in Pseudorandom Number Generators

Shiping Wen, Zhigang Zeng, Tingwen Huang, and Yide Zhang

Abstract—**This paper investigates the problem of exponential lag synchronization control of memristive neural networks (MNNs) via the fuzzy method and applications in pseudorandom number generators. Based on the knowledge of memristor and recurrent neural networks, the model of MNNs is established. Then, considering the state-dependent properties of memristor, a fuzzy model of MNNs is employed to provide a new way of analyzing the complicated MNNs with only two subsystems, and update laws for the connection weights of slave systems and controller gain are designed to make the slave systems exponentially lag synchronized with the master systems. Two examples about synchronization problems are presented to show the effectiveness of the obtained results, and an application of the obtained theory is also given in the pseudorandom number generator.**

Index Terms—**Adaptive lag synchronization, fuzzy model, memristor, neural networks, pseudorandom number generator (PRNG).**

I. Introduction

Although current digital computers can now possess computing speed and complexity to emulate the brain functionality of animals like a spiders, mice, and cats [1], [2], the associated energy dissipation in the system grows exponentially along the hierarchy of animal intelligence, as the sequential processing of fetch, decode, and execution of instructions through the classical von Neumann bottleneck of conventional digital computers has resulted in less-efficient machines as their ecosystems have grown to be increasingly complex [3]. For example, to perform certain cortical simulations at the cat scale even at 83 times slower firing rate, the IBM team in [1] has to employ Blue Gene/P (BG/P): a super computer equipped with 147456 CPUs and 144 TBs of main memory. On the other hand, the human brain contains more than 100 billion neurons, and each neuron has more than 20 000 synapses. Efficient circuit implementation of synapses, therefore, is especially important to build a brain-like machine. However, since shrinking the current transistor size is very difficult, introducing a more efficient approach is essential for further development of neural network implementations.

In 2008, the Williams group announced a successful fabrication of a very compact and nonvolatile nano scale memory called the memristor [4]. It was postulated by Chua [5] as the fourth basic circuit element in electrical circuits. It is based on the nonlinear characteristics of charge and flux. By supplying a voltage or current to the memristor, its resistance can be altered [6]. This way, the memristor remembers information. Several examples of successful multichip networks of spiking neurons have been recently proposed [7]–[9]; however, there are still a number of practical problems that hinder the development of truly large-scale, distributed, massively parallel networks of very large scale integration (VLSI) neurons, such as how to set the weight of individual synapses in the network. It is well-known that changes in the synaptic connections between neurons are widely believed to contribute to memory storage, and the activity-dependent development of neural networks. These changes are thought to occur through correlated-based, or Hebbian plasticity.

In addition, we notice neural networks have been widely studied in recent years, for their immense application prospective [10]–[23]. Many applications have been developed in different areas such as combinatorial optimization, knowledge acquisition, and pattern recognition. Recently, the problem of lag synchronization of coupled neural networks, which is one of the hottest research fields of complex networks, has been a challenging issue because of its potential application such as information science, biological systems, and so on [24]–[36].

On the other hand, synchronization problem of neural networks has attracted great attention because of its potential applications in many fields such as secure communications, biological systems, information science, image encryption, and pseudorandom number generator (PRNG) [37], [38]. Currently, a wide variety of synchronization phenomena have been investigated, such as complete synchronization [39]–[41], generalized synchronization [42], phase synchronization [43], and lag synchronization [44]. In the case of real applications, it is very hard to directly get the identical parameters of the master and slave systems. Therefore, adaptive synchronization may be a good choice for such cases. It is worth mentioning that in connected electronic networks, the occurrence of time delay is unavoidable because of finite signal transmission times, switching speeds, and some other reasons. Thus, the complete synchronization of neural networks is hard to implement effectively and it is more reasonable to consider the lag synchronization problem.

However, to the best of the authors' knowledge, the research on global exponential lag adaptive synchronization of memristive neural networks is still an open problem that deserves further investigation. To shorten sup gap, we investigate the problem of global exponential lag adaptive synchronization for a class of memristive neural networks with time-varying delays. The main contributions of this paper can be summarized as follows: 1) A model of MNNs is established in accordance with the memristor-based electronic circuits; 2) a fuzzy model of memristive neural networks is employed to give a new way to analyze the complicated MNNs with only two subsystems; 3) update laws are designed for the connection weights of slave systems and controller gain to make the slave systems exponentially lag synchronized with the master systems; and 4) a simulation example is presented to show the applications of the obtained results in the PRNG.

Manuscript received August 13, 2013; revised October 15, 2013; accepted November 16, 2013. Date of publication December 11, 2013; date of current version. This work was supported by the Natural Science Foundation of China under Grant 61125303, National Basic Research Program of China (973 Program) under Grant 2011CB710606, Research Fund for the Doctoral Program of Higher Education of China under Grant 20100142110021, the Excellent Youth Foundation of Hubei Province of China under Grant 2010CDA081, and the National Priority Research Project NPRP 4-451-2-168, funded by Qatar National Research Fund.

S. P. Wen, Z. G. Zeng, and Y. D. Zhang are with the School of Automation, Huazhong University of Science and Technology, and Key Laboratory of Image Processing and Intelligent Control of Education Ministry of China, Wuhan, Hubei, 430074, China (e-mail: wenshiping226@126.com; zgzeng527@126.com; edwardchang@hust.edu.cn).

T. W. Huang is with Texas A & M University at Qatar, Doha 23874, Qatar (e-mail: tingwen.huang@qatar.tamu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TFUZZ.2013.2294855

1063-6706 © 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

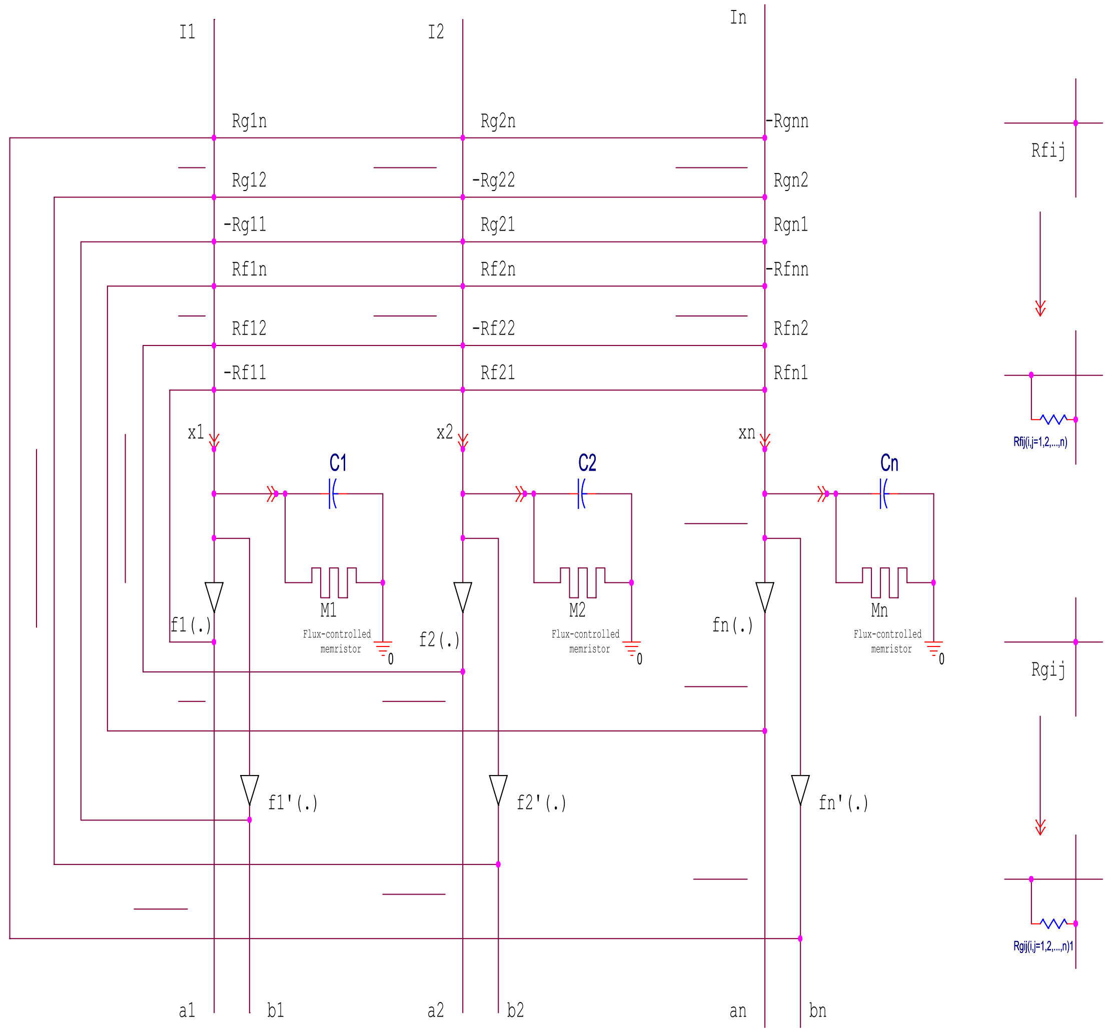


Fig. 1. Circuit of memristive network, where $x_i(.)$ is the state of the ith subsystem; $f_j(.)$ is the amplifier; R_{fij} is the connection resistor between the amplifier $f_j(.)$ and state $x_i(.)$; M_i and C_i are the memristor and capacitor; I_i is the external input; and a_i, b_i are the outputs $i, j = 1, 2, \dots, n$.

A. Circuit of Memristive Neural Networks

The memristive neural network can be implemented by VLSI circuits as shown in Fig. 1. f_j is the activation function, $\tau_j(t)$ is the time-varying delay, for the i-th subsystem, $x_i(t)$ is the voltage of the capacitor C_i, $f_j(x_j(t)), f_j(x_j(t-\tau_j(t)))$ are the functions of $x_i(t)$ with or without time-varying delays respectively, R_{fij} is the resistor between the feedback function $f_j(x_j(t))$ and $x_i(t)$, R_{gij} is the resistor between the feedback function $f_j(x_j(t-\tau_j(t)))$ and $x_i(t)$, M_i is the memristor parallel to the capacitor C_i, and I_i is an external input or bias, where $i, j = 1, 2, \dots, n$.

The memductance of the memristors can be depicted as in Fig. 2 [45], which are bounded. Thus, by Kirchoff's current law, the equation of the i-th subsystem is written as follows:

$$\begin{aligned} C_i\dot{x}_i(t) = \; & -\left[\sum_{j=1}^{n}\left(\frac{1}{R_{fij}}+\frac{1}{R_{gij}}\right)+W_i(x_i(t))\right]x_i(t) \\ & +\sum_{j=1}^{n}\frac{\text{sign}_{ij}f_j(x_j(t))}{R_{fij}} \\ & +\sum_{j=1}^{n}\frac{\text{sign}_{ij}f_j(x_j(t-\tau_j(t)))}{R_{gij}}+I_i \end{aligned} \tag{1}$$

where

$$\text{sign}_{ij} = \begin{cases} 1, & i \neq j \\ -1, & i = j \end{cases}$$

and W_i are the memductances of the memristors M_i, and

$$W_i(x_i(t)) = \begin{cases} W_i', & x_i(t) \leq 0 \\ W_i'', & x_i(t) > 0. \end{cases}$$

Therefore

$$\begin{aligned} \dot{x}_i(t) = & -d_i(x_i(t))x_i(t) + \sum_{j=1}^{n} a_{ij}f_j(x_j(t)) \\ & + \sum_{j=1}^{n} b_{ij}f_j(x_j(t-\tau_j(t))) + s_i \end{aligned} \tag{2}$$

where

$$a_{ij} = \frac{\text{sign}_{ij}}{C_iR_{fij}}, b_{ij} = \frac{\text{sign}_{ij}}{C_iR_{gij}}, s_i = \frac{I_i}{C_i}$$

$$\begin{aligned} d_i(x_i(t)) &= \frac{1}{C_i}\left[\sum_{j=1}^{n}\left(\frac{1}{R_{fij}}+\frac{1}{R_{gij}}\right)+W_i(x_i(t))\right] \\ &= \begin{cases} d_{1i}, & x_i(t) \leq 0 \\ d_{2i}, & x_i(t) > 0. \end{cases} \end{aligned}$$

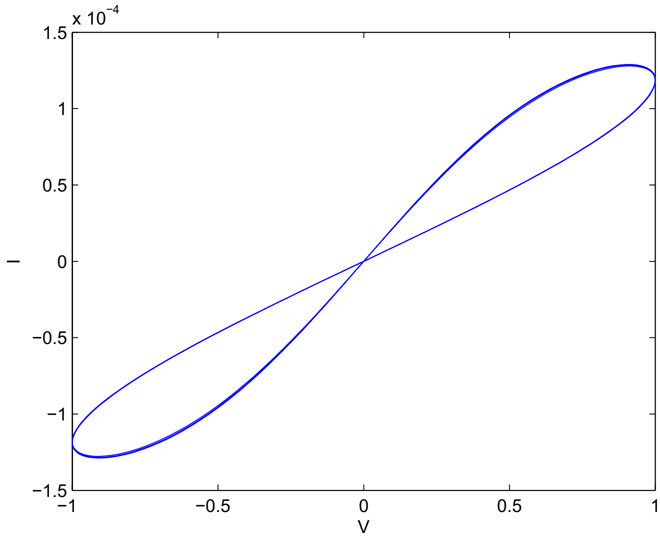


Fig. 2. Typical *I-V* characteristic of memristor [45]. The pinched hysteresis loop occurs because of the nonlinear relationship between the memristance current and voltage. The memristor exhibits the feature of pinched hysteresis, which means that a lag occurs between the application and the removal of a field and its subsequent effect, just like the neurons in the human brain.

Then, we can get

$$\dot{x}(t) = -D(x(t))x(t) + Af(x(t)) + Bf(x(t-\tau(t))) + s \quad (3)$$

where

$$\begin{aligned} D(x(t)) &= \operatorname{diag}\{d_1(x_1(t)), d_2(x_2(t)), \ldots, d_n(x_n(t))\} \\ A &= [a_{ij}]_{n\times n}, B = [b_{ij}]_{n\times n}, s = (s_1, s_2, \ldots, s_n)^T \\ f(x(t)) &= (f_1(x_1(t)), \ldots, f_n(x_n(t)))^{\mathrm{T}} \\ f(x(t-\tau(t))) &= (f_1(x_1(t-\tau_1(t))), \ldots, f_n(t-\tau_n(t)))^{\mathrm{T}}. \end{aligned}$$

B. Fuzzy Model of Memristive Neural Networks

To solve the problem about nonlinear control, fuzzy logic has attracted much attention as a powerful tool. Among various kinds of fuzzy methods, the Takagi–Sugeno fuzzy systems are widely accepted as a useful tool for design and analysis of fuzzy control system [46]–[53]. Currently, some control methods for memristive systems have been proposed [54], in which the number of the linear subsystems is decided by how many minimum nonlinear terms should be linearized in original system. Then, the memristive neural network (2) can be exactly represented by the fuzzy model as follows:

Rule 1: IF $x_i(t)$ is N_{1i}, THEN

$$\begin{aligned} \dot{x}_i(t) = &-d_{1i}x_i(t) + \sum_{j=1}^{n} a_{ij}f_j(x_j(t)) \\ &+ \sum_{j=1}^{n} b_{ij}f_j(x_j(t-\tau_j(t))) + s_i \end{aligned}$$

Rule 2: IF $x_i(t)$ is N_{2i}, THEN

$$\begin{aligned} \dot{x}_i(t) = &-d_{2i}x_i(t) + \sum_{j=1}^{n} a_{ij}f_j(x_j(t)) \\ &+ \sum_{j=1}^{n} b_{ij}f_j(x_j(t-\tau_j(t))) + s_i \end{aligned}$$

where N_{1i} is $x_i(t) \le 0$, and N_{2i} is $x_i(t) > 0$. With a center-average defuzzier, the over fuzzy system is represented as

$$\begin{aligned} \dot{x}_i(t) = &-\sum_{l=1}^{2} \vartheta_{li}(t)d_{li}x_i(t) + \sum_{j=1}^{n} a_{ij}f_j(x_j(t)) \\ &+ \sum_{j=1}^{n} b_{ij}f_j(x_j(t-\tau_j(t))) + s_i \end{aligned} \quad (4)$$

where

$$\vartheta_{1i}(t) = \begin{cases} 1, & x_i(t) \le 0, \\ 0, & x_i(t) > 0, \end{cases} \qquad \vartheta_{2i}(t) = \begin{cases} 0, & x_i(t) \le 0 \\ 1, & x_i(t) > 0. \end{cases}$$

When the system becomes complicated with n memristors, there are 2^n subsystems (according to 2^n fuzzy rules) and 2^n equations in the T–S fuzzy system. If n is large, the number of linear subsystems in the T–S fuzzy system is huge. For this problem, Li and Ge proposed a fuzzy modeling method and applied in the lag synchronization problem of two totally different chaotic systems [55]. Based on this work, a new fuzzy model is proposed to simplify memristive systems, in which only two subsystems are included. Furthermore, through this model, the idea of PDC can be applied to achieve between subsystems. Therefore, system (4) can be represented by

$$\begin{aligned} \dot{x}(t) = &-\sum_{l=1}^{2} \Pi_l(t)D_l x(t) + Af(x(t)) \\ &+ Bf(x(t-\tau(t))) + s \end{aligned} \quad (5)$$

where $\Pi_l(t) = \operatorname{diag}\{\vartheta_{l1}(t), \ldots, \vartheta_{ln}(t)\}$, $\sum_{l=1}^{2} \vartheta_{li}(t) = 1, i = 1, \ldots, n, l = 1, 2$, and

$$D_l = \operatorname{diag}\{d_{l1}, d_{l2}, \ldots, d_{ln}\}.$$

The initial conditions of system (5) is in the form of $x(t) = \phi(t) \in \mathcal{C}([-\tau, 0], \mathbb{R}^n)$, $\tau = \max_{1\le i\le n}\{\tau_i(t)\}$.

II. PRELIMINARIES

Denote $u = (u_1, \ldots, u_n)^T$, $|u|$ as the absolute-value vector; i.e., $|u| = (|u_1|, |u_2|, \ldots, |u_n|)^T$, $\|x\|_p$ as the p-norm of the vector x with $p, 1 \le p < \infty$. $\|x\|_\infty = \max_{i\in\{1,2,\ldots,n\}}|x_i|$ is the vector infinity norm. Denote $\|D\|_p$ as the p-norm of the matrix D with p. Denote $\mathcal{C}$ as the set of continuous functions. In addition, we assume the following throughout the paper:

A1. For $i \in \{1, 2, \ldots, n\}$, the activation function f_i is Lipschitz continuous, and $\forall r_1, r_2 \in \mathbb{R}$, there exists real number ι_i such that

$$0 \le \frac{f_i(r_1) - f_i(r_2)}{r_1 - r_2} \le \iota_i$$

where $f_i(0) = 0, r_1, r_2 \in \mathbb{R}$, and $r_1 \ne r_2$.

A2. For $i \in \{1, 2, \ldots, n\}$, the time-varying delay $\tau_i(t)$ satisfies the following inequalities:

$$\begin{aligned} 0 \le \tau_i(t) &\le \tau \\ \dot{\tau}_i(t) &\le \mu. \end{aligned} \quad (6)$$

In this paper, we consider system (5) as the master system, and through electronic inductors, the values of memristor will be presented in the corresponding slave system; then, the slave system is given as

$$\begin{aligned} \dot{z}_i(t) = &-\sum_{l=1}^{2} \vartheta_{li}(t)d_{li}z_i(t) + \sum_{j=1}^{n} \tilde{a}_{ij}(t)f_j(z_j(t)) \\ &+ \sum_{j=1}^{n} \tilde{b}_{ij}(t)f_j(z_j(t-\tau_j(t))) + s_i + u_i(t) \end{aligned} \quad (7)$$

or in compact form

$$\dot{z}(t) = -\sum_{l=1}^{2} \Pi_l(t) D_l z(t) + \tilde{A}(t) f(z(t)) + \tilde{B}(t) f(z(t-\tau(t))) + s + u(t), t \geq 0 \tag{8}$$

where $\tilde{A}(t) = (\tilde{a}(t))_{n\times n}$, $\tilde{B}(t) = (\tilde{b}(t))_{n\times n}$ are unknown connection weights, $z(t) = (z_1(t), z_2(t), \ldots, z_n(t))^T$, $\Pi_l(t)$ is related to the master system, and $u(t) = (u_1(t), u_2(t), \ldots, u_n(t))^T$ is the control input with the following form:

$$u_i(t) = \varrho_i(t)(z_i(t) - x_i(t-\nu)) \tag{9}$$

where $\varrho_i(t)$ is the adaptive control gain which needs to be designed, and the initial condition of system (8) is in the form of $z(t) = \psi(t) \in \mathcal{C}([-\tau, 0], \mathbb{R}^n)$.

Definition 1: The master system (4) and slave system (7) are said to be globally exponentially synchronized with lag ν, if there exist positive constants λ and μ, such that

$$\|z(t) - x(t-\nu)\| \leq \omega e^{-\lambda t}, t \geq 0. \tag{10}$$

If $\nu = 0$, the synchronization is complete synchronization.

Notation: The notation used here is fairly standard. The superscript "T" stands for matrix transposition, $\mathbb{R}^n$ denotes the n-dimensional Euclidean space, and $\mathbb{R}^{m\times n}$ is the set of all real matrices of dimension $m \times n$, I and 0 represent the identity matrix and zero matrix, respectively. Matrices, if their dimensions are not explicitly stated, are assumed to be compatible for algebraic operations. Denote $u = (u_1, \ldots, u_n)^T$, $|u|$ as the absolute-value vector, i.e., $|u| = (|u_1|, |u_2|, \ldots, |u_n|)^T$, $\|x\|_p$ as the p-norm of the vector x with $p, 1 \leq p < \infty$. $\|x\|_\infty = \max_{i\in\{1,2,\ldots,n\}} |x_i|$ is the vector infinity norm. Denote $\|D\|_p$ as the p-norm of the matrix D with p. Denote $\mathcal{C}$ as the set of continuous functions.

III. Main Results

In practice, lag exists, when the synchronization happens between the master and slave systems, which can be characterized as $z(t) = x(t - \upsilon)$ for some constant lag time $\upsilon > 0$. The lag synchronization error between the master and slave systems can be presented as

$$e(t) = z(t) - x(t-\nu). \tag{11}$$

Then, we can get the adaptive lag synchronization algorithm for memristive neural networks with unknown connection weights of the slave systems.

Theorem 1: System (7) will be globally exponentially synchronized with system (4) with lag ν, if the connection weights of system (7) $\tilde{a}_{ij}(t)$, $\tilde{b}_{ij}(t)$ and controller gain $\varrho_i(t)$, $i, j = 1, 2, \ldots, n$, are adapted in accordance with the following update law:

$$\begin{aligned}
\dot{\tilde{a}}_{ij}(t) &= -\varpi_{ij} f_j(z_j(t)) \text{sign}(e_i(t)) e^{\varepsilon t} \\
\dot{\tilde{b}}_{ij}(t) &= -\alpha_{ij} f_j(z_j(t-\tau_j(t))) \text{sign}(e_i(t)) e^{\varepsilon t} \\
\dot{\varrho}_i(t) &= -\beta_i |e_i(t)| e^{\varepsilon t}
\end{aligned} \tag{12}$$

where ϖ_{ij}, α_{ij} and β_i, $i, j = 1, 2, \ldots, n$ are arbitrary positive constants.

Proof: Let $V_i(t) = e^{\varepsilon t}|e_i(t)|$, and $d_i^- = \min_{l=1,2}\{d_{li}\}$. Calculating the derivative of $V_i(t)$ along systems (4) and (7), we can get

$$\begin{aligned}
\dot{V}_i(t) \leq e^{\varepsilon t} \Bigg\{ & (\varepsilon + \varrho_i(t) - d_i^-)|e_i(t)| \\
&+ \left\| \sum_{j=1}^{n} a_{ij}(f_j(z_j(t)) - f_j(x_j(t-\nu))) \right\| \\
&+ \left\| \sum_{j=1}^{n} b_{ij}(f_j(z_j(t-\tau_j(t))) - f_j(x_j(t-\tau_j(t)-\nu))) \right\| \\
&+ \sum_{j=1}^{n} (\tilde{a}_{ij}(t) - a_{ij}) f_j(z_j(t)) \text{sign}(e_i(t)) \\
&+ \sum_{j=1}^{n} (\tilde{b}_{ij}(t) - b_{ij}) f_j(z_j(t-\tau_j(t))) \text{sign}(e_i(t)) \Bigg\} \\
\leq e^{\varepsilon t} \Bigg\{ & (\varepsilon + \varrho_i(t) - d_i^-)|e_i(t)| + \sum_{j=1}^{n} |a_{ij}| \iota_j |e_j(t)| \\
&+ \sum_{j=1}^{n} |b_{ij}| \iota_j |e_j(t-\tau_j(t))| \\
&+ \sum_{j=1}^{n} (\tilde{a}_{ij}(t) - a_{ij}) f_j(z_j(t)) \text{sign}(e_i(t)) \\
&+ \sum_{j=1}^{n} (\tilde{b}_{ij}(t) - b_{ij}) f_j(z_j(t-\tau_j(t))) \text{sign}(e_i(t)) \Bigg\}.
\end{aligned} \tag{13}$$

Define a Lyapunov functional as

$$\begin{aligned}
V(t) = \sum_{i=1}^{n} \Bigg\{ & V_i(t) + \sum_{j=1}^{n} |b_{ij}| \iota_j e^{\varepsilon \tau_j(t)} \int_{t-\tau_j(t)}^{t} |e_j(s)| e^{\varepsilon s} ds \\
&+ \sum_{j=1}^{n} \frac{1}{2\varpi_{ij}} (\tilde{a}_{ij}(t) - a_{ij})^2 + \sum_{j=1}^{n} \frac{1}{2\alpha_{ij}} (\tilde{b}_{ij}(t) - b_{ij})^2 \\
&+ \sum_{j=1}^{n} \frac{1}{2\beta_i} (\varrho_i(t) + \omega_i)^2 \Bigg\}
\end{aligned} \tag{14}$$

where ω_i is a constant, and $\tilde{a}_{ij}(t)$, $\tilde{b}_{ij}(t)$, and $\varrho_i(t)$ are adapted by the update law (12). Then

$$\begin{aligned}
\dot{V}(t) = \sum_{i=1}^{n} \Bigg\{ & \dot{V}_i(t) + e^{\varepsilon t} \sum_{j=1}^{n} |b_{ij}| \iota_j (e^{\varepsilon \tau_j(t)} |e_j(t)| - |e_j(t-\tau_j(t))|) \\
&+ \sum_{j=1}^{n} \frac{1}{\varpi_{ij}} (\tilde{a}_{ij}(t) - a_{ij}) \dot{\tilde{a}}_{ij}(t) + \sum_{j=1}^{n} \frac{1}{\alpha_{ij}} (\tilde{b}_{ij}(t) - b_{ij}) \dot{\tilde{b}}_{ij}(t) \\
&+ \frac{1}{\beta_i} (\varrho_i(t) + \omega_i) \dot{\varrho}_i(t) \Bigg\} \\
\leq \sum_{i=1}^{n} \Bigg\{ & \dot{V}_i(t) + e^{\varepsilon t} \sum_{j=1}^{n} |b_{ij}| \iota_j (e^{\varepsilon \tau} |e_j(t)| - |e_j(t-\tau_j(t))|) \\
&+ \sum_{j=1}^{n} \frac{1}{\varpi_{ij}} (\tilde{a}_{ij}(t) - a_{ij}) \dot{\tilde{a}}_{ij}(t) + \sum_{j=1}^{n} \frac{1}{\alpha_{ij}} (\tilde{b}_{ij}(t) - b_{ij}) \dot{\tilde{b}}_{ij}(t) \\
&+ \frac{1}{\beta_i} (\varrho_i(t) + \omega_i) \dot{\varrho}_i(t) \Bigg\}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{n} \Bigg\{ \dot{V}_i(t) + e^{\varepsilon t} \sum_{j=1}^{n} |b_{ij}| \iota_j (e^{\varepsilon\tau} |e_j(t)| - |e_j(t-\tau_j(t))|) \\
&\quad - \sum_{j=1}^{n} (\tilde{a}_{ij}(t) - a_{ij}) f_j(z_j(t)) \mathrm{sign}(e_i(t)) e^{\varepsilon t} \\
&\quad - \sum_{j=1}^{n} (\tilde{b}_{ij}(t) - b_{ij}) f_j(z_j(t-\tau_j(t))) \mathrm{sign}(e_i(t)) e^{\varepsilon t} \\
&\quad - (\varrho_i(t) + \omega_i) |e_i(t)| e^{\varepsilon t} \Bigg\}. \qquad (15)
\end{aligned}
$$

Combining the derivatives (13) and (15)

$$
\begin{aligned}
\dot{V}(t) &\le e^{\varepsilon t} \sum_{i=1}^{n} \Bigg\{ (\varepsilon - \omega_i - d_i^-) |e_i(t)| \\
&\quad + \sum_{j=1}^{n} (|a_{ij}| + |b_{ij}| e^{\varepsilon\tau}) \iota_j |e_j(t)| \Bigg\} \\
&= e^{\varepsilon t} \sum_{i=1}^{n} \left(\varepsilon - d_i^- + \sum_{j=1}^{n} (|a_{ij}| + |b_{ij}| e^{\varepsilon\tau}) \iota_j - \omega_i \right) |e_i(t)|. \qquad (16)
\end{aligned}
$$

Set

$$\omega_i > -d_i^- + \sum_{j=1}^{n} (|a_{ij}| + |b_{ij}| e^{\varepsilon\tau}) \iota_j. \qquad (17)$$

If we let ε be small enough, we can get

$$e^{\varepsilon t} \sum_{i=1}^{n} \left(\varepsilon - d_i^- + \sum_{j=1}^{n} (|a_{ij}| + |b_{ij}| e^{\varepsilon\tau}) \iota_j - \omega_i \le 0. \qquad (18)$$

Hence

$$\dot{V}(t) \le 0 \qquad (19)$$

and with (13) and (15), we can get

$$e^{\varepsilon t} \sum_{i=1}^{n} |e_i(t)| \le V(t) \le V(0) \qquad (20)$$

where

$$
\begin{aligned}
V(0) &\le \sum_{i=1}^{n} \Bigg\{ \left(1 + \sum_{j=1}^{n} |b_i j| \iota_j \tau e^{\varepsilon\tau} \right) \max_{s \in [-\tau+\nu, \nu]} |\varphi(s) - \phi(s)| \\
&\quad + F_i(0) \Bigg\}
\end{aligned}
$$

in which

$$
\begin{aligned}
F_i(0) &= \sum_{j=1}^{n} \frac{1}{2\varpi_{ij}} (\tilde{a}_{ij}(0) - a_{ij})^2 + \sum_{j=1}^{n} \frac{1}{2\alpha_{ij}} (\tilde{b}_{ij}(0) - b_{ij})^2 \\
&\quad + \sum_{j=1}^{n} \frac{1}{2\beta_i} (\varrho_i(0) + \omega_i)^2 \Bigg\} \\
&\equiv \Phi.
\end{aligned}
$$

Consequently, the following inequality holds:

$$\sum_{i=1}^{n} |e_i(t)| \le \Phi e^{-\varepsilon t}, t \ge 0. \qquad (21)$$

This completes the proof.

When the connection weights $\tilde{a}_{ij}(t)$ and $\tilde{b}_{ij}(t)$ are known, then we can get update law of the adaptive controller gain as follows.

Corollary 1: System (7) will be globally exponentially synchronized with system (4) with lag ν, if controller gains $\varrho_i(t), i = 1, 2, \ldots, n$ are adaptive iterating in accordance with the following update law:

$$\dot{\varrho}_i(t) = -\beta_i |e_i(t)| e^{\varepsilon t} \qquad (22)$$

where β_i, $i = 1, 2, \ldots, n$ are arbitrary positive constants.

The proof is the same as in Theorem 1, and therefore, it is omitted.

IV. Numerical Examples

In this section, several numerical examples are utilized to demonstrate the effectiveness and applications of the obtained results.

Example 1: Consider memristive system (5) with

$$A = \begin{bmatrix} 2 & -0.1 \\ -5 & 4.5 \end{bmatrix}, B = \begin{bmatrix} -1.5 & -0.1 \\ -0.2 & -4 \end{bmatrix}$$

$$f_i(x_i) = \tanh(x_i), \tau_i(t) = 1, s_i = 0, i = 1, 2.$$

Let

$$d_1(x_1(t)) = \begin{cases} 0.9, & x_1(t) \le 0 \\ 1.1, & x_1(t) > 0, \end{cases} \quad d_2(x_2(t)) = \begin{cases} 1.1, & x_2(t) \le 0 \\ 0.9, & x_2(t) > 0. \end{cases}$$

The initial values of master system (5) are set to be $[0.4 \;\; 0.6]$. And the dynamical behaviors of this system are shown as in Fig. 3, which are chaotic and can be used in secure communications.

Without loss of generality, let

$$\tilde{A}(t) = A, \tilde{B}(t) = \begin{bmatrix} \tilde{b}_{11}(t) & -0.1 \\ -0.2 & \tilde{b}_{22}(t) \end{bmatrix}$$

in which

$$\tilde{b}_{11}(0) = \tilde{b}_{22}(0) = 1.$$

Set $\varepsilon = 0.02$, $\varrho_i(0) = 1$, $\alpha_{11} = \alpha_{22} = 0.9$, $\beta_1 = \beta_2 = 0.8$, the lag time $\nu = 1.5$, and the initial values of the slave system is set to be $[-0.5 \;\; -0.5]$. Then, we can get the simulation results as shown in Fig. 4.

Example 2: Consider memristive system (5) with

$$A = \begin{bmatrix} 1 + \pi/4 & 20 \\ 0.1 & 1 + \pi/4 \end{bmatrix}, B = \begin{bmatrix} -1.3\sqrt{2}\pi/4 & 0.1 \\ 0.1 & -1.3\sqrt{2}\pi/4 \end{bmatrix}$$

$$f_i(x_i) = 0.5(|x_i + 1| - |x_i - 1|), \tau_i(t) = 1, s_i = 0, i = 1, 2.$$

Let

$$d_1(x_1(t)) = \begin{cases} 1.0, & x_1(t) \le 0 \\ 1.2, & x_1(t) > 0, \end{cases} \quad d_2(x_2(t)) = \begin{cases} 1.2, & x_2(t) \le 0 \\ 1.0, & x_2(t) > 0. \end{cases}$$

The initial values of master system (5) are set to be $[0.2 \;\; -0.2]$, and the dynamical behaviors of this system are shown in Fig. 5.

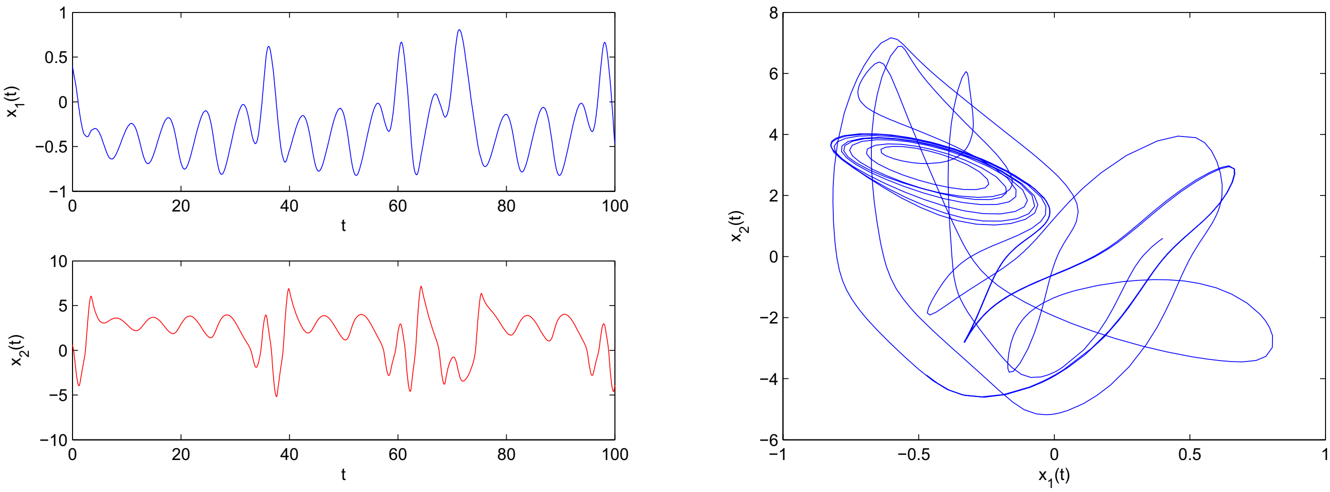
Let

$$\tilde{A}(t) = A, \tilde{B}(t) = B$$

and $\varepsilon = 0.01$, $\varrho_i(0) = 0.6$, $\beta_1 = \beta_2 = 0.8$, the lag time $\nu = 1.5$, and the initial values of the slave system are set to be $[-0.5 \;\; 0.5]$. Then, we can get the simulation results as shown in Fig. 6.

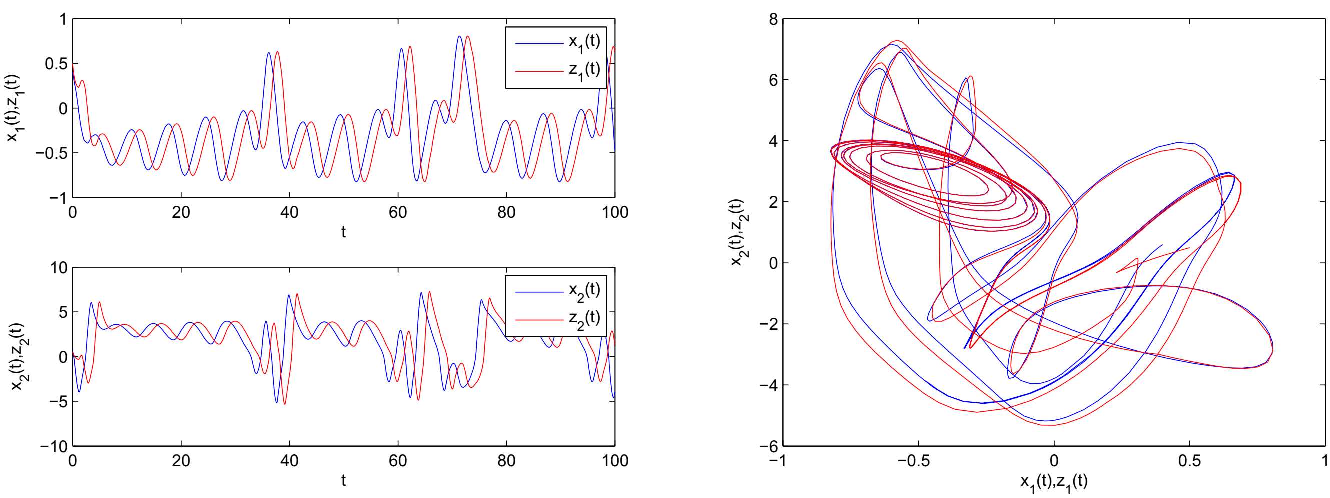
V. Memristive Neural Networks in the Psuedorandom Number Generator

Based on the above discussion, this section will discuss the applications of the exponential lag synchronization between MNNs in the field of the PRNG.



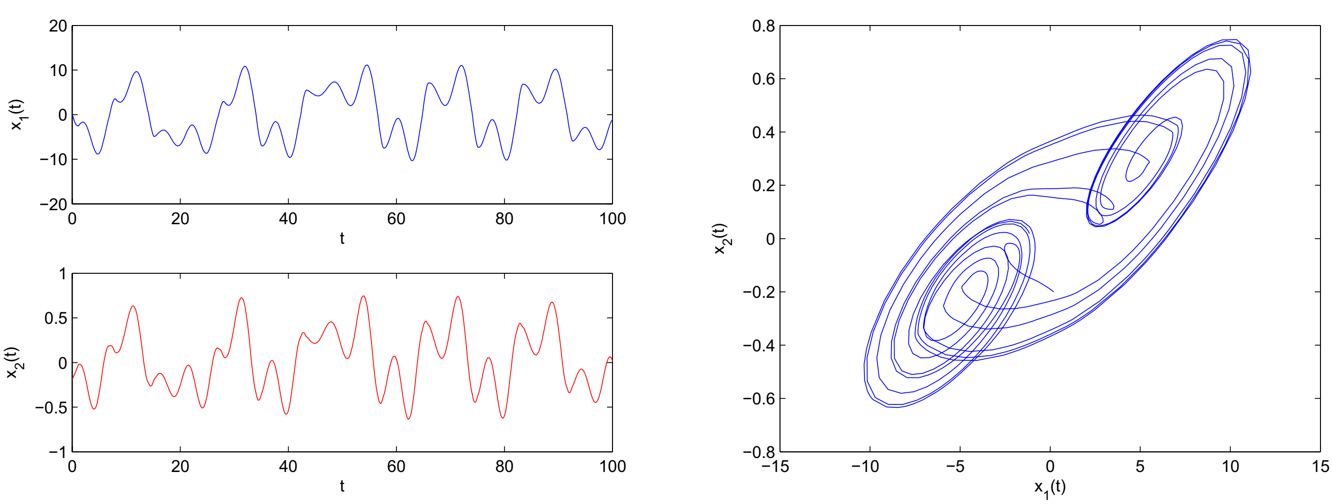
(a) (b)

Fig. 3. Transient behavior of memristive system (4).



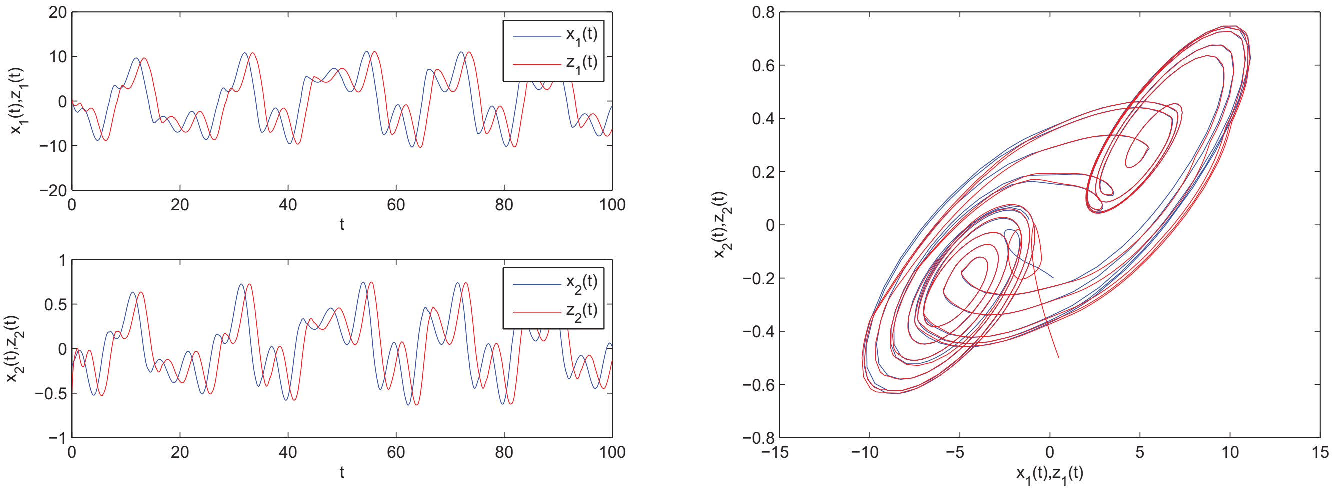
(a) (b)

Fig. 4. State trajectories of master system (4) and slave system (7) when lag time $\nu = 1.5$.



(a) (b)

Fig. 5. Transient behavior of memristive system (4).



(a) (b)

Fig. 6. State trajectories of master system (4) and slave system (7) when lag time $\nu = 1.5$.

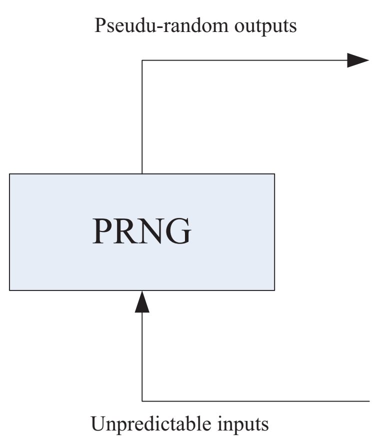


Fig. 7. Black-box view of a PRNG.

As random number generation plays an important role in cryptography and software testing, PRNG are intended to be general-purpose vehicles for the creation of random data used in these areas as in Fig. 7 [65]. In addition, many different methods exist to generate pseudorandom numbers like Blum-Blum-Shub, Mersenne Twister algorithms, etc. It is well known that pseudorandomness is the basis for cryptography and is essential for the achievement of any cryptographic function such as encryption, authentication, and identification. Neural networks can be used to generate random numbers as they are highly nonlinear mathematical systems. Based on the dynamics of neural networks, pseudorandom numbers are generated via neural plasticity.

Meanwhile, it is important to produce a perfect random number generator that gets a series of independent identically distributed continuous random variables in $[0 \;\; 1]$ [66]. One can produce a perfect random number generator only using nondeterministic physical phenomena. It is a practical way to employ a computer to produce a random-looking sequence of numbers in the way of a recursive rule. However, there exist unavoidable problems such as numerical algorithms are deterministic, the sequence of numbers cannot be "really random." To solve such limitations, chaotic systems provide a clue to produce random number generators as the deterministic systems may have a time evolution that appears rather "irregular" with the typical features of genuine random processes.

In this paper, utilizing the complex dynamics of chaotic MNNs and the algorithms of synchronization control, memristive neural networks are used to generate pseudorandom numbers to achieve encryption and decryption functions.

If we define a pseudorandom number sequence $k(t) = h(y_1(t), y_2(t)), t \in [t\text{start}, \text{tend}]$, $[t\text{start}, \text{tend}]$ is the operating interval, and

$$h(y_1(t), y_2(t)) = \begin{cases} 1, & y_1(t) \le y_2(t) \\ 0, & y_1(t) > y_2(t) \end{cases} \tag{23}$$

where

$$y_1(t) = \frac{x_1(t)}{\max_{t\in[t\text{start},\text{tend}]}\{x_1(t)\}}, y_2(t) = \frac{x_2(t)}{\max_{t\in[t\text{start},\text{tend}]}\{x_2(t)\}}.$$

Then, we can get the PRNG by the chaotic MNNs in Examples 1 and 2 as in Fig. 8(a) and (b), respectively.

Let $s(t)$ be the transmitted signal, which is operated with the signals generated by PRNG, we can get the encrypted signals as follows:

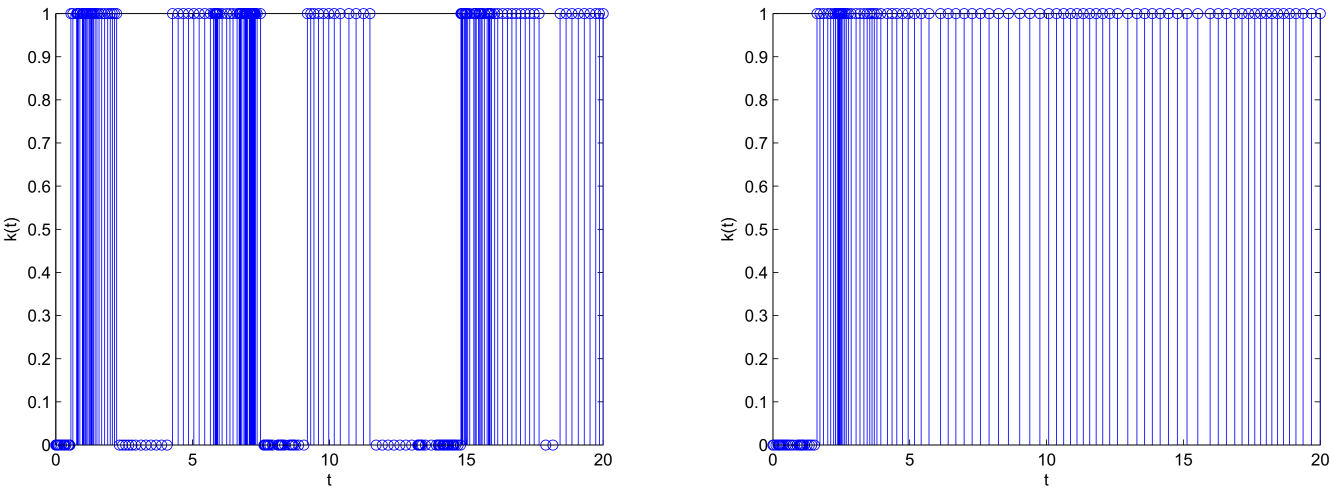
$$p(t) = s(t) \otimes k(t). \tag{24}$$

The original signal and correspondingly encrypted signals by systems in Examples 1 and 2 are shown in Figs. 9 and 10, respectively.

Remark 1: From these simulation results, it is obvious that the encrypted signals produced by the PRNG is quite different from the original signals because of the chaotic properties of the MNNs, and they can be easily retrieved through the synchronization of the chaotic MNNs in the receipt termination.

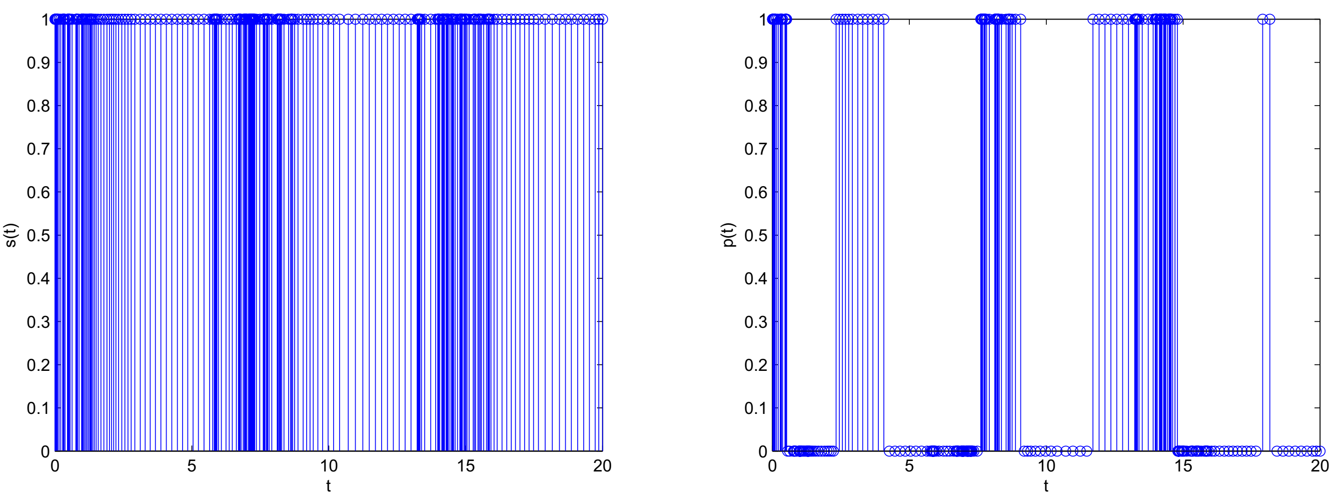
Remark 2: The decryption process is the same as the encryption process and as the existence of lag time, the decryption PRNG should be adopted after ν/h signals, where h is the length of the iterative step.

Remark 3: As the kernel of a new generation of cipher dreams, the hardware implementation of PRNGs based on memristive neural networks will come true in the future for their great applications in the field of the signal communication. Meanwhile, it is meaningful to investigate the design of the algorithm of digital image encryption and decryption via PRNGs based on memristive neural networks.



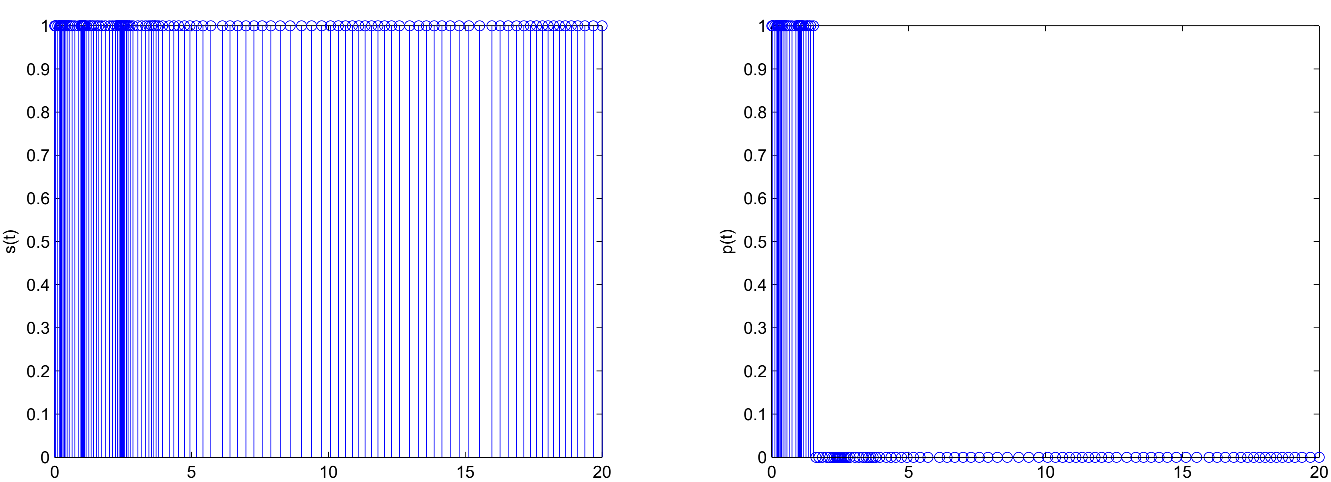
(a) (b)

Fig. 8. (a) PRNG produced by chaotic memristive neural networks in Example 1. (b) PRNG produced by chaotic memristive neural networks in Example 2.



(a) (b)

Fig. 9. (a) Original signal. (b) Encrypted signals by the chaotic memristive neural network in Example 1.



(a) (b)

Fig. 10. (a) Original signal. (b) Encrypted signals by the chaotic memristive neural network in Example 2.

VI. Conclusion

In this paper, the problem of exponential lag adaptive synchronization control of MNNs was investigated via fuzzy method and applied in a pseudorandom number generator. A model of fuzzy MNNs was established with only two subsystems, and the update laws for the connection weights of slave systems and controller gain are designed to make the slave systems exponentially lag synchronized with the master systems. Illustrative examples were given to demonstrate the effectiveness of the obtained results, which can be extended into the field of PRNG as an encryption method.

In the future, there are some issues that deserve further investigation, such as 1) how to design the optimal update laws of the connection weights of slave systems to achieve desired results; 2) how to extend the applications of memristive neural networks into the fields of optimal computation, biological systems, and secure communications; and 3) how to deal with the problem of synchronization of memristive neural networks with discrete and distributed time-varying delays.

References

[1] R. Ananthanarayanan, S. Eser, H. Simon, and D. Modha, presented at the IEEE/ACM Conf. High Perform. Netw. Comput., Portland, OR, USA, Nov. 2009.

[2] L. Smith, *Handbook of Nature-Inspired and Innovative Computing: Integrating Classical Models with Emerging Technologies*. New York, NY, USA: Springer-Verlag, pp. 433–475.

[3] S. Jo, T. Chang, I. Ebong, B. Bhadviya, P. Mazumder, and W. Lu, "Nanoscale memristor device as synapse in neuromorphic systems," *Nanotech. Lett.*, vol. 10, pp. 1297–1301, 2010.

[4] D. Strukov, G. Snider, D. Stewart, and R. Williams, "The missing memristor found," *Nature*, vol. 453, pp. 80–83, 2008.

[5] L. Chua, "Memristor—The missing circuit element," *IEEE Trans. Circuits Theory*, vol. CT-18, no. 5, pp. 507–519, Sep. 1971.

[6] M. Sharifiy and Y. Banadaki, "General spice models for memristor and application to circuit simulation of memristor-based synapses and memory cells," *J. Circuits Syst. Comput.*, vol. 19, pp. 407–424, 2010.

[7] T. Choi, B. Shi, and K. Boahen, "An on–off orientation selective address event representation image transceiver chip," *IEEE Trans. Circuits Syst. I*, vol. 51, no. 2, pp. 342–353, Feb. 2004.

[8] G. Indiveri, "A neuromorphic VLSI device for implementing 2-D selective attention systems," *IEEE Trans. Neural Netw.*, vol. 12, no. 6, pp. 1455–1463, Nov. 2001.

[9] S. Liu and R. Douglas, "Temporal coding in a silicon network of integrate-and-fire neurons," *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1305–1314, Sep. 2004.

[10] G. Bao, S. Wen, and Z. Zeng, "Robust stability analysis of interval fuzzy Cohen–Grossberg neural networks with piecewise constant argument of generalized type," *Neural Netw.*, vol. 33, pp. 32–41, 2012.

[11] J. Cao, D. Huang, and Y. Qu, "Global robust stability of delayed recurrent neural networks," *Chaos Solit. Fract.*, vol. 23, pp. 221–229, 2005.

[12] Q. Song, "Synchronization analysis in an array of asymmetric neural networks with time-varying delays and nonlinear coupling," *Appl. Math. Comput.*, vol. 216, pp. 1605–1613, 2010.

[13] Q. Song, Z. Zhao, and J. Yang, "Passivity and passification for stochastic Takagi–Sugeno fuzzy systems with mixed time-varying delays," *Neurocomput.*, vol. 122, pp. 330–337, Dec. 2013.

[14] C. Li and G. Feng, "Delay-interval-dependent stability of recurrent neural networks with time-varying delay," *Neurocomput.*, vol. 72, pp. 1179–1183, 2009.

[15] M. Forti and P. Nistri, "Global convergence of neural networks with discontinous neuron activations," *IEEE Trans. Circuits Syst. I*, vol. 50, no. 11, pp. 1421–1435, Nov. 2003.

[16] C. Li, G. Feng, and X. Liao, "Stabilization of nonlinear system via periodically intermittent control," *IEEE Trans. Circuit Syst. II*, vol. 54, no. 11, pp. 1019–1023, Nov. 2007.

[17] Y. Xia, Z. Yang, and M. Han, "Lag synchronization of unknown chaotic delayed yang-yang-type fuzzy neural networks with noise perturbation based on adaptive control and parameter identification," *IEEE Trans. Neural Netw.*, vol. 20, no. 7, pp. 1165–1180, Jul. 2009.

[18] Y. Shen and J. Wang, "An improved algebraic criterion for global exponential stability of recurrent neural networks with time-varying delays," *IEEE Trans. Neural Netw.*, vol. 19, no. 3, pp. 528–531, Mar. 2008.

[19] Z. Zeng and J. Wang, "Global exponential stability of recurrent neural networks with time-varying delays in the presence of strong external stimuli," *Neural Netw.*, vol. 19, pp. 1528–1537, 2006.

[20] Z. Zeng, J. Wang, and X. Liao, "Global asmptotic stability and global exponential stability of neural networks with unbounded time-varying delays," *IEEE Trans. Circuits Syst. II*, vol. 52, no. 3, pp. 168–173, Mar. 2005.

[21] B. Pizzileo, L. Kang, G. W. Irwin, and W. Zhao, "Improved structure optimization for fuzzy-neural networks," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 6, pp. 1076–1089, Dec. 2012.

[22] R. J. Wai and Y. W. Lin, "Adaptive moving-target tracking control of a vision-based mobile robot via a dynamic petri recurrent fuzzy neural network," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 4, pp. 688–701, Aug. 2013.

[23] G.-D. Wu and P. H. Huang, "A vectorization-optimization-method-based type-2 fuzzyneural network for noisy data classification," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 1, pp. 1–15, Feb. 2013.

[24] J. Fu, H. G. Zhang, T. D. Mag, and Q. L. Zhang, "On passivity analysis for stochastic neural networks with interval time-varying delay," *Neurocomput.*, vol. 73, pp. 795–801, 2010.

[25] H. G. Zhang, L. L. Cai, and Z. Bien, "A fuzzy basis function vector-based multivariable adaptive controller for nonlinear systems," *IEEE Trans. Syst. Man Cybern. B*, vol. 30, no. 1, pp. 210–217, Feb. 2000.

[26] H. G. Zhang and D. R. Liu, *Fuzzy Modeling and Fuzzy Control*. Boston, MA, USA: Birkhäuser, 2006.

[27] H. G. Zhang, T. D. Ma, G. B. Huang, and Z. L. Wang, "Robust global exponential synchronization of uncertain chaotic delayed neural networks via dual-stage impulsive control," *IEEE Trans. Syst. Man Cybern. B*, vol. 40, no. 3, pp. 831–844, Jun. 2010.

[28] P. Balasubramaniam, K. Kalpana, and R. Rakkiyappan, "Linear matrix inequality approach for synchronization control of fuzzy cellular neural networks with mixed time delays," *Chin. Phys. B*, vol. 21, p. 048402, 2012.

[29] O. Kwon, M. Park, S. Lee, J. Park, and E. Cha, "Stability for neural networks with time-varying delays via some new approaches," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 2, pp. 181–193, Feb. 2013.

[30] Z. Wu, P. Shi, H. Su, and J. Chu, "Sampled-data synchronization of chaotic systems with time delays," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 24, no. 3, pp. 410–421, Mar. 2013.

[31] G. Wu and P. Huang, "A vectorization-optimization-method-based type-2 fuzzy neural network for noisy data classification," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 1, pp. 1–15, Feb. 2013.

[32] C. Juang, T. Chen, and W. Cheng, "Speedup of implementing fuzzy neural networks with high-dimensional inputs through parallel processing on graphic processing units," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 4, pp. 717–728, Aug. 2011.

[33] J. Cao, G. Chen, and P. Li, "Global synchronization in an array of delayed neural networks with hybrid coupling," *IEEE Trans. Syst. Man Cybern. B*, vol. 38, no. 2, pp. 488–498, Apr. 2008.

[34] M. Park, O. Kwon, J. Park, S. Lee, and E. Cha, "Synchronization criteria for coupled neural networks with interval time-varying delays and leakage delay," *Appl. Math. Comput.*, vol. 218, pp. 6762–6775, 2012.

[35] J. Li, H. Kazemian, and M. Afzal, "Neural network approaches for noisy language modeling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 11, pp. 1773–1784, Nov. 2013.

[36] H. Zhang, T. Ma, G. Huang, and Z. Wang, "Robust global exponential synchronization of uncertain chaotic delayed neural networks via dualstage impulsive control," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 40, no. 3, pp. 831–844, Jun. 2010.

[37] L. Chua and T. Roska, *Cellular Networks and Visual Computing: Foundations and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[38] G. Chen and X. Dong, *From Chaos to Order: Methodolgies, Perspectives, and Applications*. Singapore: World Scientific, 1998.

[39] Y. Xia, Z. Yang, and M. Han, "Synchronization schemes for coupled identical Yang-Yang type fuzzy cellular neural networks," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 14, pp. 3645–3659, 2009.

[40] Q. Song, "Design of controller on synchronization of chaotic neural networks with mixed time-varying delays," *Neurocomput.*, vol. 72, pp. 3288–3295, 2009.

[41] Q. Zhu and J. Cao, "Adaptive synchronization under almost every initial data for stochastic neural networks with time-varying delays and

distributed delays," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 16, pp. 2139–2159, 2011.

[42] M. Sun, C. Zeng, and L. Tian, "Linear generalized synchronization between two complex networks," *Commun. Nonlinear Sci. Numer.*, vol. 15, pp. 2162–2167, 2010.

[43] F. Breve, L. Zhao, M. Quiles, and E. Macau, "Chaotic phase synchronization and desynchronization in an oscillator network for object selection," *Neural Netw.*, vol. 22, pp. 728–737, 2009.

[44] W. Yu and J. Cao, "Adaptive Q-S (lag, anticipated, and complete) time-varying synchronization and parameters identification of uncertain delayed neural networks," *Chaos*, vol. 16, 023119, 2006.

[45] N. Gergel-Hackett, B. Hamadani, J. Suehle, C. Richter, C. Hacker, and D. Gundlach, "A flexible solution-processed memristor," *IEEE Electr. Device Lett.*, vol. 30, no. 7, pp. 706–708, Jul. 2009.

[46] C. Chiu and T. Chiang, "Robust output regulation of T-S fuzzy systems with multiple time-varying state and input delays," *IEEE Trans. Fuzzy Syst.*, vol. 17, no. 4, pp. 962–975, Aug. 2009.

[47] J. Dong, Y. Wang, and G. Yang, "Control synthesis of continuous-time T-S fuzzy systems with local nonlinear models," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 39, no. 5, pp. 1245–1258, Oct. 2009.

[48] X. Liu and S. Zhong, "T-S fuzzy model-based impulsive control of chaotic systems with exponential decay rate," *Phys. Lett. A*, vol. 370, pp. 260–264, 2007.

[49] C. Park and Y. Cho, "T-S model based indirect adaptive fuzzy control using online parameter estimation," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 34, no. 6, pp. 2293–2302, Dec. 2004.

[50] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modelling and control," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Jan./Feb. 1985.

[51] L. G. Wu, Z. G. Feng, and W. X. Zheng, "Exponential stability analysis for delayed neural networks with switching parameters: Average dwell time approach," *IEEE Trans. Neural Netw.*, vol. 21, no. 9, pp. 1396–1407, Sep. 2010.

[52] L. G. Wu, Z. G. Feng, and J. Lam, "Stability and synchronization of discrete-time neural networks with switching parameters and time-varying delays," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 12, pp. 1957–1972, Dec. 2013.

[53] L. G. Wu, X. J. Su, P. Shi, and J. B. Qiu, "A new approach to stability analysis and stabilization of discrete-time T-S fuzzy time-varying delay systems," *IEEE Trans. Syst. Man Cybern. B*, vol. 41, no. 1, pp. 273–286, Feb. 2011.

[54] A. Wu and Z. Zeng, "Exponential stabilization of memristive neural networks with time delays," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 12, pp. 1919–1929, Dec. 2012.

[55] S. Li and Z. Ge, "Fuzzy modeling and synchronization of two totally different chaotic systems via novel fuzzy model," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 41, no. 4, pp. 1015–1026, Aug. 2011.

[56] E. Fridman, A. Seuret, and J. P. Richard, "Robust sampled-data stabilization of linear systems: An input delay approach," *Automatica*, vol. 40, pp. 1441–1446, 2004.

[57] W. Zhao and Y. Tan, "Harmless delay for global exponential stability of Cohen–Grossberg neural networks," *Math. Comput. Simul.*, vol. 74, pp. 47–57, 2007.

[58] S. Arik, "Global robust stability analysis of neural networks with discrete time delays," *Chaos Solit. Fract.*, vol. 26, pp. 1407–1414, 2005.

[59] N. Ozcan and S. Arik, "Global robust stability analysis of neural networks with multiple time delays," *IEEE Trans. Circuits Syst. I*, vol. 53, no. 1, pp. 166–176, Jan. 2006.

[60] S. Wen and Z. Zeng, "Dynamics analysis of a class of memristor-based recurrent networks with time-varying delays in the presence of strong external stimuli," *Neural Process. Lett.*, vol. 35, pp. 47–59, 2012.

[61] A. Wu, S. Wen, and Z. Zeng, "Synchronization control of a class of memristor-based recurrent neural networks," *Inform. Sci.*, vol. 183, pp. 106–116, 2012.

[62] H. Zhang, Y. Xie, Z. Wang, and C. Zheng, "Adaptive synchronization between two different chaotic neural networks with time delay," *IEEE Trans. Neural Network*, vol. 18, no. 6, pp. 1841–1845, Nov. 2007.

[63] Y. Liu, Z. D. Wang, J. Liang, and X. Liu, "Synchronization and state estimation for discrete-time complex networks with distributed delays," *IEEE. Trans. Syst. Man Cybern. B Cybern.*, vol. 38, no. 5, pp. 1314–1325, Oct. 2008.

[64] S. Zhu, Y. Shen, and G. Chen, "Exponential Passivity of neural networks with time-varying delay and uncertainty," *Phys. Lett. A*, vol. 375, pp. 136–142, 2010.

[65] L. Blum, M. Blum, and M. Shub, "A simple unpredictable Pseudo-random number generator," *SIAM J. Comput.*, vol. 15, pp. 364–383, 1986.

[66] M. Falcioni, L. Palatella, and S. Pigolotti, "Properties making a chaotic system a good pseudo random number generator," *Phys. Rev. E*, vol. 72, 016220, 2005.

Short Papers

Exponential Adaptive Lag Synchronization of Memristive Neural Networks via Fuzzy Method and Applications in Pseudorandom Number Generators

Shiping Wen, Zhigang Zeng, Tingwen Huang, and Yide Zhang

***Abstract*—This paper investigates the problem of exponential lag synchronization control of memristive neural networks (MNNs) via the fuzzy method and applications in pseudorandom number generators. Based on the knowledge of memristor and recurrent neural networks, the model of MNNs is established. Then, considering the state-dependent properties of memristor, a fuzzy model of MNNs is employed to provide a new way of analyzing the complicated MNNs with only two subsystems, and update laws for the connection weights of slave systems and controller gain are designed to make the slave systems exponentially lag synchronized with the master systems. Two examples about synchronization problems are presented to show the effectiveness of the obtained results, and an application of the obtained theory is also given in the pseudorandom number generator.**

***Index Terms*—Adaptive lag synchronization, fuzzy model, memristor, neural networks, pseudorandom number generator (PRNG).**

I. Introduction

Although current digital computers can now possess computing speed and complexity to emulate the brain functionality of animals like a spiders, mice, and cats [1], [2], the associated energy dissipation in the system grows exponentially along the hierarchy of animal intelligence, as the sequential processing of fetch, decode, and execution of instructions through the classical von Neumann bottleneck of conventional digital computers has resulted in less-efficient machines as their ecosystems have grown to be increasingly complex [3]. For example, to perform certain cortical simulations at the cat scale even at 83 times slower firing rate, the IBM team in [1] has to employ Blue Gene/P (BG/P): a super computer equipped with 147456 CPUs and 144 TBs of main memory. On the other hand, the human brain contains more than 100 billion neurons, and each neuron has more than 20 000 synapses. Efficient circuit implementation of synapses, therefore, is especially important to build a brain-like machine. However, since shrinking the current transistor size is very difficult, introducing a more efficient approach is essential for further development of neural network implementations.

In 2008, the Williams group announced a successful fabrication of a very compact and nonvolatile nano scale memory called the memristor [4]. It was postulated by Chua [5] as the fourth basic circuit element in electrical circuits. It is based on the nonlinear characteristics of charge and flux. By supplying a voltage or current to the memristor, its resistance can be altered [6]. This way, the memristor remembers information. Several examples of successful multichip networks of spiking neurons have been recently proposed [7]–[9]; however, there are still a number of practical problems that hinder the development of truly large-scale, distributed, massively parallel networks of very large scale integration (VLSI) neurons, such as how to set the weight of individual synapses in the network. It is well-known that changes in the synaptic connections between neurons are widely believed to contribute to memory storage, and the activity-dependent development of neural networks. These changes are thought to occur through correlated-based, or Hebbian plasticity.

In addition, we notice neural networks have been widely studied in recent years, for their immense application prospective [10]–[23]. Many applications have been developed in different areas such as combinatorial optimization, knowledge acquisition, and pattern recognition. Recently, the problem of lag synchronization of coupled neural networks, which is one of the hottest research fields of complex networks, has been a challenging issue because of its potential application such as information science, biological systems, and so on [24]–[36].

On the other hand, synchronization problem of neural networks has attracted great attention because of its potential applications in many fields such as secure communications, biological systems, information science, image encryption, and pseudorandom number generator (PRNG) [37], [38]. Currently, a wide variety of synchronization phenomena have been investigated, such as complete synchronization [39]–[41], generalized synchronization [42], phase synchronization [43], and lag synchronization [44]. In the case of real applications, it is very hard to directly get the identical parameters of the master and slave systems. Therefore, adaptive synchronization may be a good choice for such cases. It is worth mentioning that in connected electronic networks, the occurrence of time delay is unavoidable because of finite signal transmission times, switching speeds, and some other reasons. Thus, the complete synchronization of neural networks is hard to implement effectively and it is more reasonable to consider the lag synchronization problem.

However, to the best of the authors' knowledge, the research on global exponential lag adaptive synchronization of memristive neural networks is still an open problem that deserves further investigation. To shorten sup gap, we investigate the problem of global exponential lag adaptive synchronization for a class of memristive neural networks with time-varying delays. The main contributions of this paper can be summarized as follows: 1) A model of MNNs is established in accordance with the memristor-based electronic circuits; 2) a fuzzy model of memristive neural networks is employed to give a new way to analyze the complicated MNNs with only two subsystems; 3) update laws are designed for the connection weights of slave systems and controller gain to make the slave systems exponentially lag synchronized with the master systems; and 4) a simulation example is presented to show the applications of the obtained results in the PRNG.

Manuscript received August 13, 2013; revised October 15, 2013; accepted November 16, 2013. Date of publication December 11, 2013; date of current version. This work was supported by the Natural Science Foundation of China under Grant 61125303, National Basic Research Program of China (973 Program) under Grant 2011CB710606, Research Fund for the Doctoral Program of Higher Education of China under Grant 20100142110021, the Excellent Youth Foundation of Hubei Province of China under Grant 2010CDA081, and the National Priority Research Project NPRP 4-451-2-168, funded by Qatar National Research Fund.

S. P. Wen, Z. G. Zeng, and Y. D. Zhang are with the School of Automation, Huazhong University of Science and Technology, and Key Laboratory of Image Processing and Intelligent Control of Education Ministry of China, Wuhan, Hubei, 430074, China (e-mail: wenshiping226@126.com; zgzeng527@126.com; edwardchang@hust.edu.cn).

T. W. Huang is with Texas A & M University at Qatar, Doha 23874, Qatar (e-mail: tingwen.huang@qatar.tamu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TFUZZ.2013.2294855

1063-6706 © 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

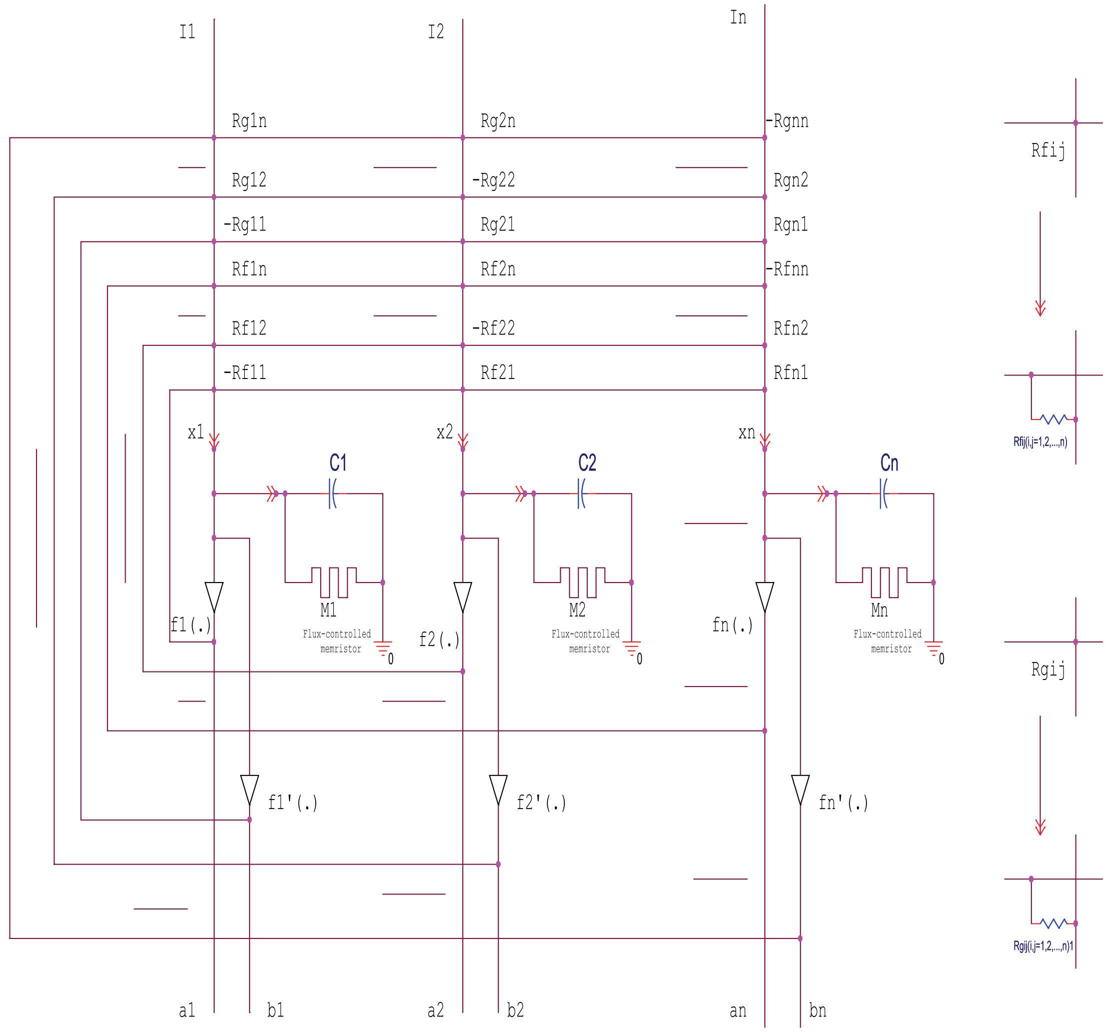


Fig. 1. Circuit of memristive network, where $x_i(.)$ is the state of the ith subsystem; $f_j(.)$ is the amplifier; R_{fij} is the connection resistor between the amplifier $f_j(.)$ and state $x_i(.)$; M_i and C_i are the memristor and capacitor; I_i is the external input; and a_i, b_i are the outputs $i, j = 1, 2, \dots, n$.

A. Circuit of Memristive Neural Networks

The memristive neural network can be implemented by VLSI circuits as shown in Fig. 1. f_j is the activation function, $\tau_j(t)$ is the time-varying delay, for the i-th subsystem, $x_i(t)$ is the voltage of the capacitor C_i, $f_j(x_j(t)), f_j(x_j(t-\tau_j(t)))$ are the functions of $x_i(t)$ with or without time-varying delays respectively, R_{fij} is the resistor between the feedback function $f_j(x_j(t))$ and $x_i(t)$, R_{gij} is the resistor between the feedback function $f_j(x_j(t-\tau_j(t)))$ and $x_i(t)$, M_i is the memristor parallel to the capacitor C_i, and I_i is an external input or bias, where $i, j = 1, 2, \dots, n$.

The memductance of the memristors can be depicted as in Fig. 2 [45], which are bounded. Thus, by Kirchoff's current law, the equation of the i-th subsystem is written as follows:

$$
\begin{aligned}
C_i\dot{x}_i(t) = \quad & -\left[\sum_{j=1}^{n}\left(\frac{1}{R_{fij}}+\frac{1}{R_{gij}}\right)+W_i(x_i(t))\right]x_i(t) \\
& + \sum_{j=1}^{n}\frac{\mathrm{sign}_{ij} f_j(x_j(t))}{R_{fij}} \\
& + \sum_{j=1}^{n}\frac{\mathrm{sign}_{ij} f_j(x_j(t-\tau_j(t)))}{R_{gij}} + I_i
\end{aligned}
\tag{1}
$$

where

$$
\mathrm{sign}_{ij} = \begin{cases} 1, & i \neq j \\ -1, & i = j \end{cases}
$$

and W_i are the memductances of the memristors M_i, and

$$
W_i(x_i(t)) = \begin{cases} W_i', & x_i(t) \leq 0 \\ W_i'', & x_i(t) > 0. \end{cases}
$$

Therefore

$$
\begin{aligned}
\dot{x}_i(t) = & -d_i(x_i(t))x_i(t) + \sum_{j=1}^{n} a_{ij} f_j(x_j(t)) \\
& + \sum_{j=1}^{n} b_{ij} f_j(x_j(t-\tau_j(t))) + s_i
\end{aligned}
\tag{2}
$$

where

$$
a_{ij} = \frac{\mathrm{sign}_{ij}}{C_i R_{fij}}, b_{ij} = \frac{\mathrm{sign}_{ij}}{C_i R_{gij}}, s_i = \frac{I_i}{C_i}
$$

$$
\begin{aligned}
d_i(x_i(t)) &= \frac{1}{C_i}\left[\sum_{j=1}^{n}\left(\frac{1}{R_{fij}}+\frac{1}{R_{gij}}\right)+W_i(x_i(t))\right] \\
&= \begin{cases} d_{1i}, & x_i(t) \leq 0 \\ d_{2i}, & x_i(t) > 0. \end{cases}
\end{aligned}
$$

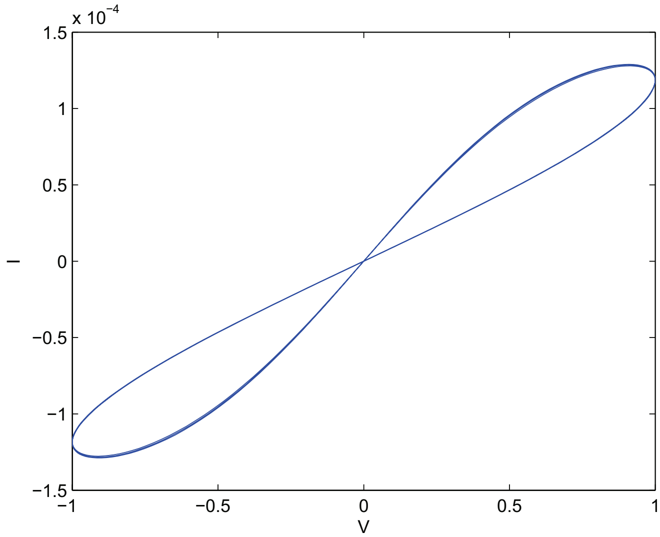


Fig. 2. Typical *I-V* characteristic of memristor [45]. The pinched hysteresis loop occurs because of the nonlinear relationship between the memristance current and voltage. The memristor exhibits the feature of pinched hysteresis, which means that a lag occurs between the application and the removal of a field and its subsequent effect, just like the neurons in the human brain.

Then, we can get

$$\dot{x}(t) = -D(x(t))x(t) + Af(x(t)) + Bf(x(t-\tau(t))) + s \quad (3)$$

where

$$\begin{aligned} D(x(t)) &= \mathrm{diag}\{d_1(x_1(t)), d_2(x_2(t)), \dots, d_n(x_n(t))\} \\ A &= [a_{ij}]_{n\times n}, B = [b_{ij}]_{n\times n}, s = (s_1, s_2, \dots, s_n)^T \\ f(x(t)) &= (f_1(x_1(t)), \dots, f_n(x_n(t)))^{\mathrm{T}} \\ f(x(t-\tau(t))) &= (f_1(x_1(t-\tau_1(t))), \dots, f_n(t-\tau_n(t)))^{\mathrm{T}}. \end{aligned}$$

B. Fuzzy Model of Memristive Neural Networks

To solve the problem about nonlinear control, fuzzy logic has attracted much attention as a powerful tool. Among various kinds of fuzzy methods, the Takagi–Sugeno fuzzy systems are widely accepted as a useful tool for design and analysis of fuzzy control system [46]–[53]. Currently, some control methods for memristive systems have been proposed [54], in which the number of the linear subsystems is decided by how many minimum nonlinear terms should be linearized in original system. Then, the memristive neural network (2) can be exactly represented by the fuzzy model as follows:

Rule 1: IF $x_i(t)$ is N_{1i}, THEN

$$\begin{aligned} \dot{x}_i(t) = &-d_{1i}x_i(t) + \sum_{j=1}^{n} a_{ij}f_j(x_j(t)) \\ &+ \sum_{j=1}^{n} b_{ij}f_j(x_j(t-\tau_j(t))) + s_i \end{aligned}$$

Rule 2: IF $x_i(t)$ is N_{2i}, THEN

$$\begin{aligned} \dot{x}_i(t) = &-d_{2i}x_i(t) + \sum_{j=1}^{n} a_{ij}f_j(x_j(t)) \\ &+ \sum_{j=1}^{n} b_{ij}f_j(x_j(t-\tau_j(t))) + s_i \end{aligned}$$

where N_{1i} is $x_i(t) \le 0$, and N_{2i} is $x_i(t) > 0$. With a center-average defuzzier, the over fuzzy system is represented as

$$\begin{aligned} \dot{x}_i(t) = &-\sum_{l=1}^{2} \vartheta_{li}(t)d_{li}x_i(t) + \sum_{j=1}^{n} a_{ij}f_j(x_j(t)) \\ &+ \sum_{j=1}^{n} b_{ij}f_j(x_j(t-\tau_j(t))) + s_i \end{aligned} \quad (4)$$

where

$$\vartheta_{1i}(t) = \begin{cases} 1, & x_i(t) \le 0, \\ 0, & x_i(t) > 0, \end{cases} \quad \vartheta_{2i}(t) = \begin{cases} 0, & x_i(t) \le 0 \\ 1, & x_i(t) > 0. \end{cases}$$

When the system becomes complicated with n memristors, there are 2^n subsystems (according to 2^n fuzzy rules) and 2^n equations in the T–S fuzzy system. If n is large, the number of linear subsystems in the T–S fuzzy system is huge. For this problem, Li and Ge proposed a fuzzy modeling method and applied in the lag synchronization problem of two totally different chaotic systems [55]. Based on this work, a new fuzzy model is proposed to simplify memristive systems, in which only two subsystems are included. Furthermore, through this model, the idea of PDC can be applied to achieve between subsystems. Therefore, system (4) can be represented by

$$\begin{aligned} \dot{x}(t) = &-\sum_{l=1}^{2} \Pi_l(t)D_l x(t) + Af(x(t)) \\ &+ Bf(x(t-\tau(t))) + s \end{aligned} \quad (5)$$

where $\Pi_l(t) = \mathrm{diag}\{\vartheta_{l1}(t), \dots, \vartheta_{ln}(t)\}$, $\sum_{l=1}^{2} \vartheta_{li}(t) = 1, i = 1, \dots, n, l = 1, 2$, and

$$D_l = \mathrm{diag}\{d_{l1}, d_{l2}, \dots, d_{ln}\}.$$

The initial conditions of system (5) is in the form of $x(t) = \phi(t) \in \mathcal{C}([-\tau, 0], \mathbb{R}^n)$, $\tau = \max_{1\le i\le n}\{\tau_i(t)\}$.

II. PRELIMINARIES

Denote $u = (u_1, \dots, u_n)^T$, $|u|$ as the absolute-value vector; i.e., $|u| = (|u_1|, |u_2|, \dots, |u_n|)^T$, $\|x\|_p$ as the p-norm of the vector x with $p, 1 \le p < \infty$. $\|x\|_\infty = \max_{i\in\{1,2,\dots,n\}}|x_i|$ is the vector infinity norm. Denote $\|D\|_p$ as the p-norm of the matrix D with p. Denote $\mathcal{C}$ as the set of continuous functions. In addition, we assume the following throughout the paper:

A1. For $i \in \{1, 2, \dots, n\}$, the activation function f_i is Lipschitz continuous, and $\forall r_1, r_2 \in \mathbb{R}$, there exists real number ι_i such that

$$0 \le \frac{f_i(r_1) - f_i(r_2)}{r_1 - r_2} \le \iota_i$$

where $f_i(0) = 0, r_1, r_2 \in \mathbb{R}$, and $r_1 \ne r_2$.

A2. For $i \in \{1, 2, \dots, n\}$, the time-varying delay $\tau_i(t)$ satisfies the following inequalities:

$$\begin{aligned} 0 \le \tau_i(t) \le \tau \\ \dot{\tau}_i(t) \le \mu. \end{aligned} \quad (6)$$

In this paper, we consider system (5) as the master system, and through electronic inductors, the values of memristor will be presented in the corresponding slave system; then, the slave system is given as

$$\begin{aligned} \dot{z}_i(t) = &-\sum_{l=1}^{2} \vartheta_{li}(t)d_{li}z_i(t) + \sum_{j=1}^{n} \tilde{a}_{ij}(t)f_j(z_j(t)) \\ &+ \sum_{j=1}^{n} \tilde{b}_{ij}(t)f_j(z_j(t-\tau_j(t))) + s_i + u_i(t) \end{aligned} \quad (7)$$

or in compact form

$$\dot{z}(t) = -\sum_{l=1}^{2} \Pi_l(t) D_l z(t) + \tilde{A}(t) f(z(t)) + \tilde{B}(t) f(z(t-\tau(t))) + s + u(t), t \geq 0 \tag{8}$$

where $\tilde{A}(t) = (\tilde{a}(t))_{n\times n}$, $\tilde{B}(t) = (\tilde{b}(t))_{n\times n}$ are unknown connection weights, $z(t) = (z_1(t), z_2(t), \ldots, z_n(t))^T$, $\Pi_l(t)$ is related to the master system, and $u(t) = (u_1(t), u_2(t), \ldots, u_n(t))^T$ is the control input with the following form:

$$u_i(t) = \varrho_i(t)(z_i(t) - x_i(t-\nu)) \tag{9}$$

where $\varrho_i(t)$ is the adaptive control gain which needs to be designed, and the initial condition of system (8) is in the form of $z(t) = \psi(t) \in \mathcal{C}([-\tau, 0], \mathbb{R}^n)$.

Definition 1: The master system (4) and slave system (7) are said to be globally exponentially synchronized with lag ν, if there exist positive constants λ and μ, such that

$$\|z(t) - x(t-\nu)\| \leq \omega e^{-\lambda t}, t \geq 0. \tag{10}$$

If $\nu = 0$, the synchronization is complete synchronization.

Notation: The notation used here is fairly standard. The superscript "T" stands for matrix transposition, $\mathbb{R}^n$ denotes the n-dimensional Euclidean space, and $\mathbb{R}^{m\times n}$ is the set of all real matrices of dimension $m \times n$, I and 0 represent the identity matrix and zero matrix, respectively. Matrices, if their dimensions are not explicitly stated, are assumed to be compatible for algebraic operations. Denote $u = (u_1, \ldots, u_n)^T$, $|u|$ as the absolute-value vector, i.e., $|u| = (|u_1|, |u_2|, \ldots, |u_n|)^T$, $\|x\|_p$ as the p-norm of the vector x with $p, 1 \leq p < \infty$. $\|x\|_\infty = \max_{i\in\{1,2,\ldots,n\}} |x_i|$ is the vector infinity norm. Denote $\|D\|_p$ as the p-norm of the matrix D with p. Denote $\mathcal{C}$ as the set of continuous functions.

III. Main Results

In practice, lag exists, when the synchronization happens between the master and slave systems, which can be characterized as $z(t) = x(t-\upsilon)$ for some constant lag time $\upsilon > 0$. The lag synchronization error between the master and slave systems can be presented as

$$e(t) = z(t) - x(t-\nu). \tag{11}$$

Then, we can get the adaptive lag synchronization algorithm for memristive neural networks with unknown connection weights of the slave systems.

Theorem 1: System (7) will be globally exponentially synchronized with system (4) with lag ν, if the connection weights of system (7) $\tilde{a}_{ij}(t)$, $\tilde{b}_{ij}(t)$ and controller gain $\varrho_i(t)$, $i, j = 1, 2, \ldots, n$, are adapted in accordance with the following update law:

$$\begin{aligned}
\dot{\tilde{a}}_{ij}(t) &= -\varpi_{ij} f_j(z_j(t)) \mathrm{sign}(e_i(t)) e^{\varepsilon t}\\
\dot{\tilde{b}}_{ij}(t) &= -\alpha_{ij} f_j(z_j(t-\tau_j(t))) \mathrm{sign}(e_i(t)) e^{\varepsilon t}\\
\dot{\varrho}_i(t) &= -\beta_i |e_i(t)| e^{\varepsilon t}
\end{aligned} \tag{12}$$

where ϖ_{ij}, α_{ij} and β_i, $i, j = 1, 2, \ldots, n$ are arbitrary positive constants.

Proof: Let $V_i(t) = e^{\varepsilon t}|e_i(t)|$, and $d_i^- = \min_{l=1,2}\{d_{li}\}$. Calculating the derivative of $V_i(t)$ along systems (4) and (7), we can get

$$\begin{aligned}
\dot{V}_i(t) \leq e^{\varepsilon t} \Bigg\{ &(\varepsilon + \varrho_i(t) - d_i^-)|e_i(t)|\\
&+ \left\| \sum_{j=1}^{n} a_{ij}(f_j(z_j(t)) - f_j(x_j(t-\nu))) \right\|\\
&+ \left\| \sum_{j=1}^{n} b_{ij}(f_j(z_j(t-\tau_j(t))) - f_j(x_j(t-\tau_j(t)-\nu))) \right\|\\
&+ \sum_{j=1}^{n} (\tilde{a}_{ij}(t) - a_{ij}) f_j(z_j(t)) \mathrm{sign}(e_i(t))\\
&+ \sum_{j=1}^{n} (\tilde{b}_{ij}(t) - b_{ij}) f_j(z_j(t-\tau_j(t))) \mathrm{sign}(e_i(t)) \Bigg\}\\
\leq e^{\varepsilon t} \Bigg\{ &(\varepsilon + \varrho_i(t) - d_i^-)|e_i(t)| + \sum_{j=1}^{n} |a_{ij}|\iota_j|e_j(t)|\\
&+ \sum_{j=1}^{n} |b_{ij}|\iota_j|e_j(t-\tau_j(t))|\\
&+ \sum_{j=1}^{n} (\tilde{a}_{ij}(t) - a_{ij}) f_j(z_j(t)) \mathrm{sign}(e_i(t))\\
&+ \sum_{j=1}^{n} (\tilde{b}_{ij}(t) - b_{ij}) f_j(z_j(t-\tau_j(t))) \mathrm{sign}(e_i(t)) \Bigg\}.
\end{aligned} \tag{13}$$

Define a Lyapunov functional as

$$\begin{aligned}
V(t) = \sum_{i=1}^{n} \Bigg\{ &V_i(t) + \sum_{j=1}^{n} |b_{ij}|\iota_j e^{\varepsilon\tau_j(t)} \int_{t-\tau_j(t)}^{t} |e_j(s)| e^{\varepsilon s} ds\\
&+ \sum_{j=1}^{n} \frac{1}{2\varpi_{ij}} (\tilde{a}_{ij}(t) - a_{ij})^2 + \sum_{j=1}^{n} \frac{1}{2\alpha_{ij}} (\tilde{b}_{ij}(t) - b_{ij})^2\\
&+ \sum_{j=1}^{n} \frac{1}{2\beta_i} (\varrho_i(t) + \omega_i)^2 \Bigg\}
\end{aligned} \tag{14}$$

where ω_i is a constant, and $\tilde{a}_{ij}(t)$, $\tilde{b}_{ij}(t)$, and $\varrho_i(t)$ are adapted by the update law (12). Then

$$\begin{aligned}
\dot{V}(t) = \sum_{i=1}^{n} \Bigg\{ &\dot{V}_i(t) + e^{\varepsilon t} \sum_{j=1}^{n} |b_{ij}|\iota_j (e^{\varepsilon\tau_j(t)}|e_j(t)| - |e_j(t-\tau_j(t))|)\\
&+ \sum_{j=1}^{n} \frac{1}{\varpi_{ij}} (\tilde{a}_{ij}(t) - a_{ij}) \dot{\tilde{a}}_{ij}(t) + \sum_{j=1}^{n} \frac{1}{\alpha_{ij}} (\tilde{b}_{ij}(t) - b_{ij}) \dot{\tilde{b}}_{ij}(t)\\
&+ \frac{1}{\beta_i} (\varrho_i(t) + \omega_i) \dot{\varrho}_i(t) \Bigg\}\\
\leq \sum_{i=1}^{n} \Bigg\{ &\dot{V}_i(t) + e^{\varepsilon t} \sum_{j=1}^{n} |b_{ij}|\iota_j (e^{\varepsilon\tau}|e_j(t)| - |e_j(t-\tau_j(t))|)\\
&+ \sum_{j=1}^{n} \frac{1}{\varpi_{ij}} (\tilde{a}_{ij}(t) - a_{ij}) \dot{\tilde{a}}_{ij}(t) + \sum_{j=1}^{n} \frac{1}{\alpha_{ij}} (\tilde{b}_{ij}(t) - b_{ij}) \dot{\tilde{b}}_{ij}(t)\\
&+ \frac{1}{\beta_i} (\varrho_i(t) + \omega_i) \dot{\varrho}_i(t) \Bigg\}
\end{aligned}$$

$$= \sum_{i=1}^{n} \Bigg\{ \dot{V}_i(t) + e^{\varepsilon t} \sum_{j=1}^{n} |b_{ij}| \iota_j (e^{\varepsilon\tau} |e_j(t)| - |e_j(t - \tau_j(t))|) - \sum_{j=1}^{n} (\tilde{a}_{ij}(t) - a_{ij}) f_j(z_j(t)) \mathrm{sign}(e_i(t)) e^{\varepsilon t} - \sum_{j=1}^{n} (\tilde{b}_{ij}(t) - b_{ij}) f_j(z_j(t - \tau_j(t))) \mathrm{sign}(e_i(t)) e^{\varepsilon t} - (\varrho_i(t) + \omega_i) |e_i(t)| e^{\varepsilon t} \Bigg\}. \tag{15}$$

Combining the derivatives (13) and (15)

$$\dot{V}(t) \le e^{\varepsilon t} \sum_{i=1}^{n} \Bigg\{ (\varepsilon - \omega_i - d_i^-) |e_i(t)| + \sum_{j=1}^{n} (|a_{ij}| + |b_{ij}| e^{\varepsilon\tau}) \iota_j |e_j(t)| \Bigg\} = e^{\varepsilon t} \sum_{i=1}^{n} \left(\varepsilon - d_i^- + \sum_{j=1}^{n} (|a_{ij}| + |b_{ij}| e^{\varepsilon\tau}) \iota_j - \omega_i \right) |e_i(t)|. \tag{16}$$

Set

$$\omega_i > -d_i^- + \sum_{j=1}^{n} (|a_{ij}| + |b_{ij}| e^{\varepsilon\tau}) \iota_j. \tag{17}$$

If we let ε be small enough, we can get

$$e^{\varepsilon t} \sum_{i=1}^{n} \left(\varepsilon - d_i^- + \sum_{j=1}^{n} (|a_{ij}| + |b_{ij}| e^{\varepsilon\tau}) \iota_j - \omega_i \le 0. \tag{18}$$

Hence

$$\dot{V}(t) \le 0 \tag{19}$$

and with (13) and (15), we can get

$$e^{\varepsilon t} \sum_{i=1}^{n} |e_i(t)| \le V(t) \le V(0) \tag{20}$$

where

$$V(0) \le \sum_{i=1}^{n} \Bigg\{ \left(1 + \sum_{j=1}^{n} |b_i j| \iota_j \tau e^{\varepsilon\tau} \right) \max_{s \in [-\tau+\nu, \nu]} |\varphi(s) - \phi(s)| + F_i(0) \Bigg\}$$

in which

$$F_i(0) = \sum_{j=1}^{n} \frac{1}{2\varpi_{ij}} (\tilde{a}_{ij}(0) - a_{ij})^2 + \sum_{j=1}^{n} \frac{1}{2\alpha_{ij}} (\tilde{b}_{ij}(0) - b_{ij})^2 + \sum_{j=1}^{n} \frac{1}{2\beta_i} (\varrho_i(0) + \omega_i)^2 \Bigg\} \equiv \Phi.$$

Consequently, the following inequality holds:

$$\sum_{i=1}^{n} |e_i(t)| \le \Phi e^{-\varepsilon t}, t \ge 0. \tag{21}$$

This completes the proof.

When the connection weights $\tilde{a}_{ij}(t)$ and $\tilde{b}_{ij}(t)$ are known, then we can get update law of the adaptive controller gain as follows.

Corollary 1: System (7) will be globally exponentially synchronized with system (4) with lag ν, if controller gains $\varrho_i(t), i = 1, 2, \ldots, n$ are adaptive iterating in accordance with the following update law:

$$\dot{\varrho}_i(t) = -\beta_i |e_i(t)| e^{\varepsilon t} \tag{22}$$

where $\beta_i, i = 1, 2, \ldots, n$ are arbitrary positive constants.

The proof is the same as in Theorem 1, and therefore, it is omitted.

IV. Numerical Examples

In this section, several numerical examples are utilized to demonstrate the effectiveness and applications of the obtained results.

Example 1: Consider memristive system (5) with

$$A = \begin{bmatrix} 2 & -0.1 \\ -5 & 4.5 \end{bmatrix}, B = \begin{bmatrix} -1.5 & -0.1 \\ -0.2 & -4 \end{bmatrix}$$

$$f_i(x_i) = \tanh(x_i), \tau_i(t) = 1, s_i = 0, i = 1, 2.$$

Let

$$d_1(x_1(t)) = \begin{cases} 0.9, & x_1(t) \le 0 \\ 1.1, & x_1(t) > 0, \end{cases} \quad d_2(x_2(t)) = \begin{cases} 1.1, & x_2(t) \le 0 \\ 0.9, & x_2(t) > 0. \end{cases}$$

The initial values of master system (5) are set to be $[0.4\ \ 0.6]$. And the dynamical behaviors of this system are shown as in Fig. 3, which are chaotic and can be used in secure communications.

Without loss of generality, let

$$\tilde{A}(t) = A, \tilde{B}(t) = \begin{bmatrix} \tilde{b}_{11}(t) & -0.1 \\ -0.2 & \tilde{b}_{22}(t) \end{bmatrix}$$

in which

$$\tilde{b}_{11}(0) = \tilde{b}_{22}(0) = 1.$$

Set $\varepsilon = 0.02$, $\varrho_i(0) = 1$, $\alpha_{11} = \alpha_{22} = 0.9$, $\beta_1 = \beta_2 = 0.8$, the lag time $\nu = 1.5$, and the initial values of the slave system is set to be $[-0.5\ \ -0.5]$. Then, we can get the simulation results as shown in Fig. 4.

Example 2: Consider memristive system (5) with

$$A = \begin{bmatrix} 1 + \pi/4 & 20 \\ 0.1 & 1 + \pi/4 \end{bmatrix}, B = \begin{bmatrix} -1.3\sqrt{2}\pi/4 & 0.1 \\ 0.1 & -1.3\sqrt{2}\pi/4 \end{bmatrix}$$

$$f_i(x_i) = 0.5(|x_i + 1| - |x_i - 1|), \tau_i(t) = 1, s_i = 0, i = 1, 2.$$

Let

$$d_1(x_1(t)) = \begin{cases} 1.0, & x_1(t) \le 0 \\ 1.2, & x_1(t) > 0, \end{cases} \quad d_2(x_2(t)) = \begin{cases} 1.2, & x_2(t) \le 0 \\ 1.0, & x_2(t) > 0. \end{cases}$$

The initial values of master system (5) are set to be $[0.2\ \ -0.2]$, and the dynamical behaviors of this system are shown in Fig. 5.

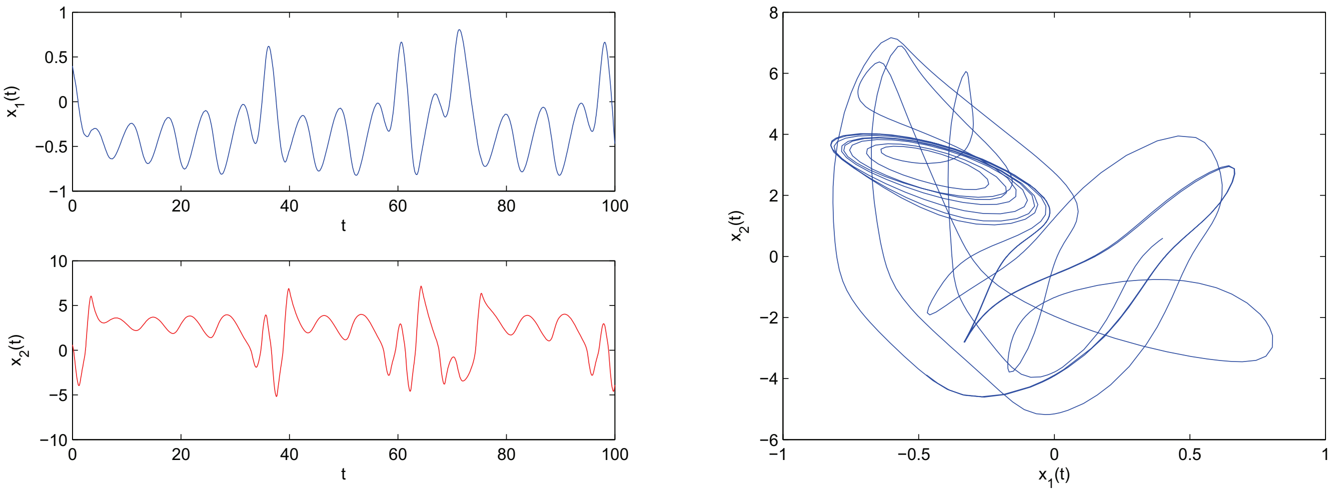
Let

$$\tilde{A}(t) = A, \tilde{B}(t) = B$$

and $\varepsilon = 0.01$, $\varrho_i(0) = 0.6$, $\beta_1 = \beta_2 = 0.8$, the lag time $\nu = 1.5$, and the initial values of the slave system are set to be $[-0.5\ \ 0.5]$. Then, we can get the simulation results as shown in Fig. 6.

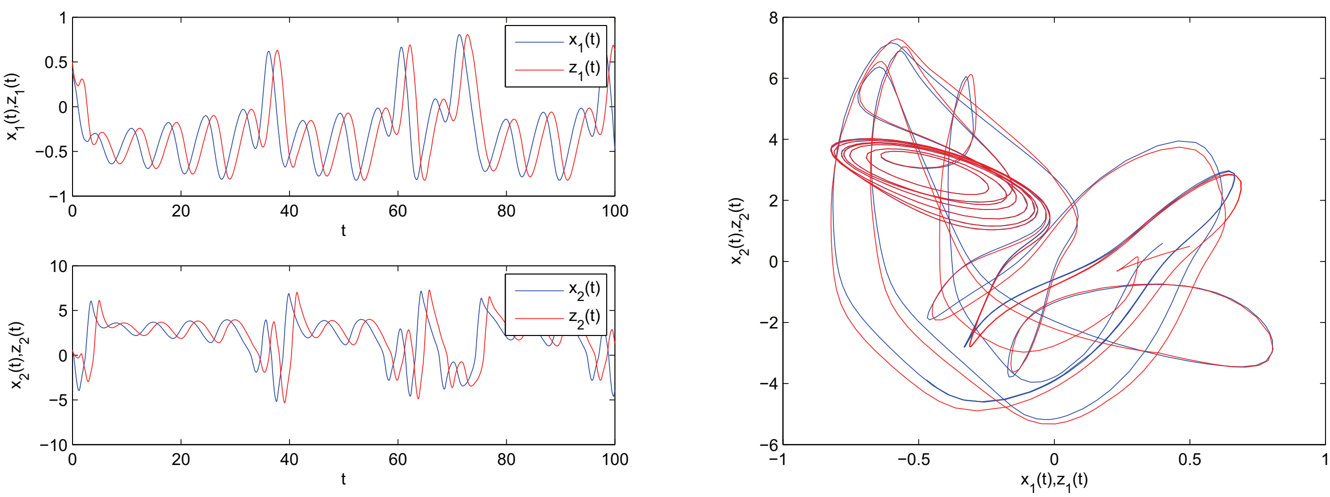
V. Memristive Neural Networks in the Psuedorandom Number Generator

Based on the above discussion, this section will discuss the applications of the exponential lag synchronization between MNNs in the field of the PRNG.



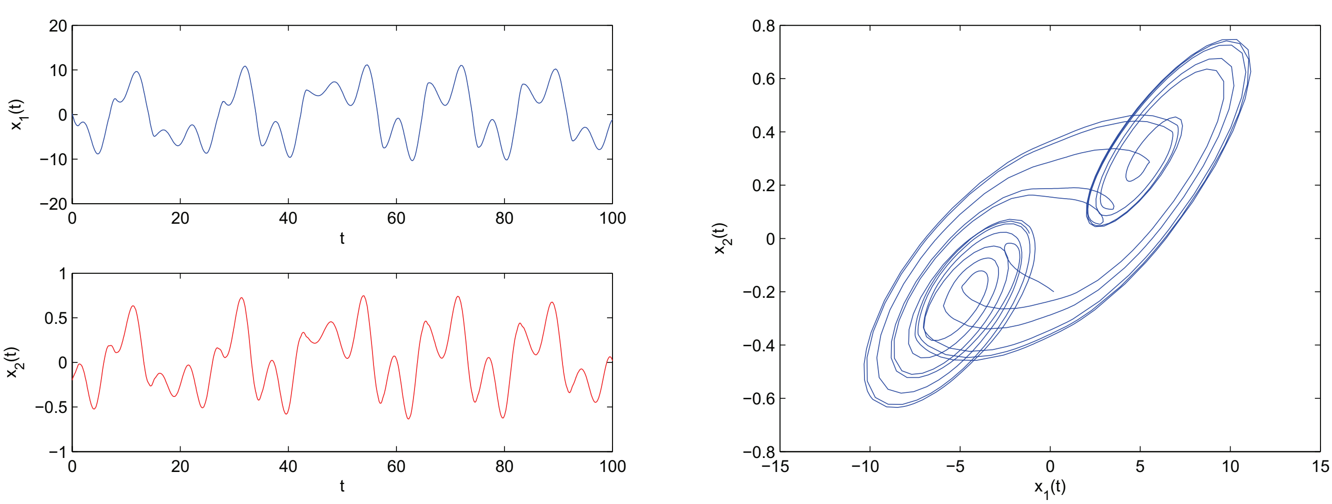
(a) (b)

Fig. 3. Transient behavior of memristive system (4).



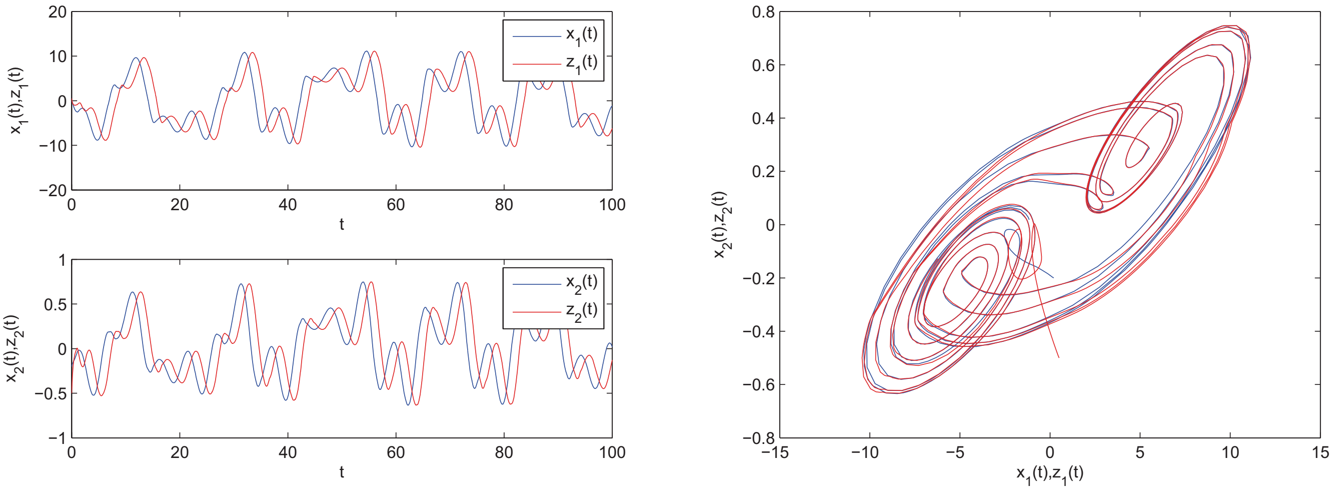
(a) (b)

Fig. 4. State trajectories of master system (4) and slave system (7) when lag time $\nu = 1.5$.



(a) (b)

Fig. 5. Transient behavior of memristive system (4).



(a) (b)

Fig. 6. State trajectories of master system (4) and slave system (7) when lag time $\nu = 1.5$.

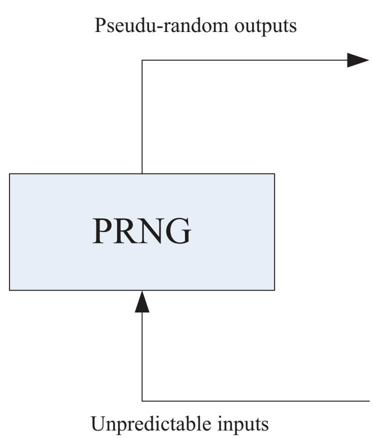


Fig. 7. Black-box view of a PRNG.

As random number generation plays an important role in cryptography and software testing, PRNG are intended to be general-purpose vehicles for the creation of random data used in these areas as in Fig. 7 [65]. In addition, many different methods exist to generate pseudorandom numbers like Blum-Blum-Shub, Mersenne Twister algorithms, etc. It is well known that pseudorandomness is the basis for cryptography and is essential for the achievement of any cryptographic function such as encryption, authentication, and identification. Neural networks can be used to generate random numbers as they are highly nonlinear mathematical systems. Based on the dynamics of neural networks, pseudorandom numbers are generated via neural plasticity.

Meanwhile, it is important to produce a perfect random number generator that gets a series of independent identically distributed continuous random variables in $[0\ \ 1]$ [66]. One can produce a perfect random number generator only using nondeterministic physical phenomena. It is a practical way to employ a computer to produce a random-looking sequence of numbers in the way of a recursive rule. However, there exist unavoidable problems such as numerical algorithms are deterministic, the sequence of numbers cannot be "really random." To solve such limitations, chaotic systems provide a clue to produce random number generators as the deterministic systems may have a time evolution that appears rather "irregular" with the typical features of genuine random processes.

In this paper, utilizing the complex dynamics of chaotic MNNs and the algorithms of synchronization control, memristive neural networks are used to generate pseudorandom numbers to achieve encryption and decryption functions.

If we define a pseudorandom number sequence $k(t) = h(y_1(t), y_2(t)), t \in [t\text{start, tend}]$, $[t\text{start, tend}]$ is the operating interval, and

$$h(y_1(t), y_2(t)) = \begin{cases} 1, & y_1(t) \le y_2(t) \\ 0, & y_1(t) > y_2(t) \end{cases} \tag{23}$$

where

$$y_1(t) = \frac{x_1(t)}{\max_{t\in[t\text{start,tend}]}\{x_1(t)\}}, y_2(t) = \frac{x_2(t)}{\max_{t\in[t\text{start,tend}]}\{x_2(t)\}}.$$

Then, we can get the PRNG by the chaotic MNNs in Examples 1 and 2 as in Fig. 8(a) and (b), respectively.

Let $s(t)$ be the transmitted signal, which is operated with the signals generated by PRNG, we can get the encrypted signals as follows:

$$p(t) = s(t) \otimes k(t). \tag{24}$$

The original signal and correspondingly encrypted signals by systems in Examples 1 and 2 are shown in Figs. 9 and 10, respectively.

Remark 1: From these simulation results, it is obvious that the encrypted signals produced by the PRNG is quite different from the original signals because of the chaotic properties of the MNNs, and they can be easily retrieved through the synchronization of the chaotic MNNs in the receipt termination.

Remark 2: The decryption process is the same as the encryption process and as the existence of lag time, the decryption PRNG should be adopted after ν/h signals, where h is the length of the iterative step.

Remark 3: As the kernel of a new generation of cipher dreams, the hardware implementation of PRNGs based on memristive neural networks will come true in the future for their great applications in the field of the signal communication. Meanwhile, it is meaningful to investigate the design of the algorithm of digital image encryption and decryption via PRNGs based on memristive neural networks.

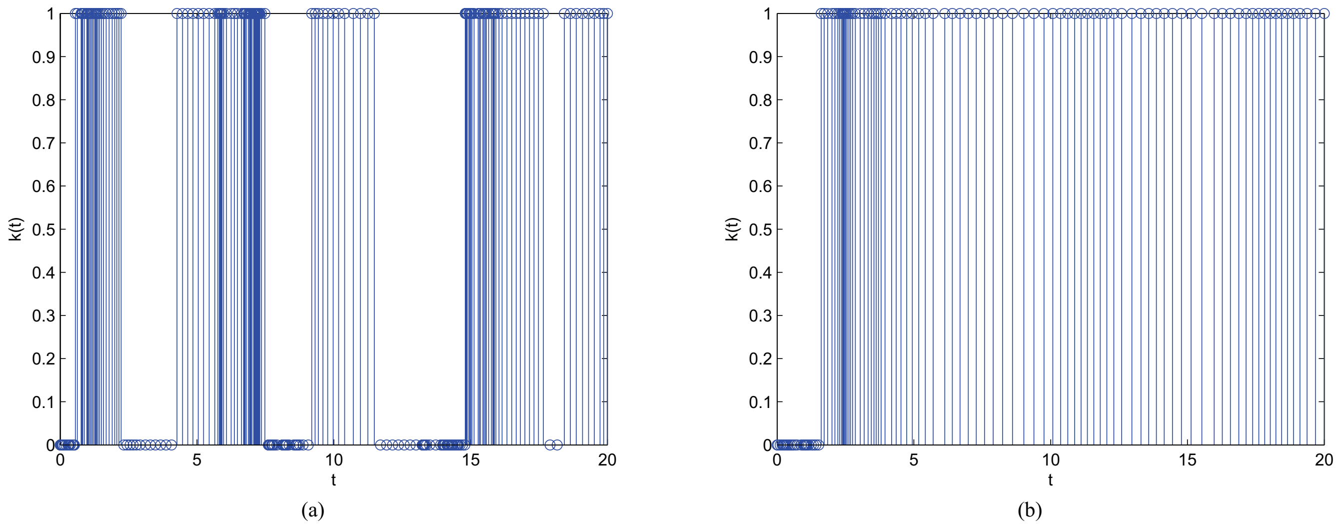


Fig. 8. (a) PRNG produced by chaotic memristive neural networks in Example 1. (b) PRNG produced by chaotic memristive neural networks in Example 2.

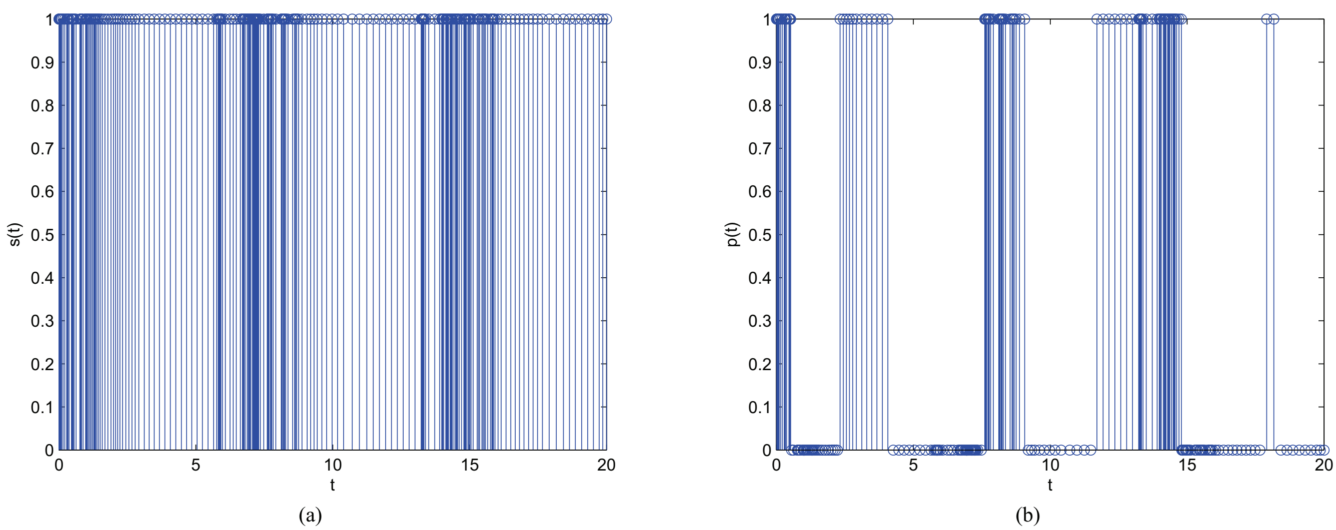


Fig. 9. (a) Original signal. (b) Encrypted signals by the chaotic memristive neural network in Example 1.

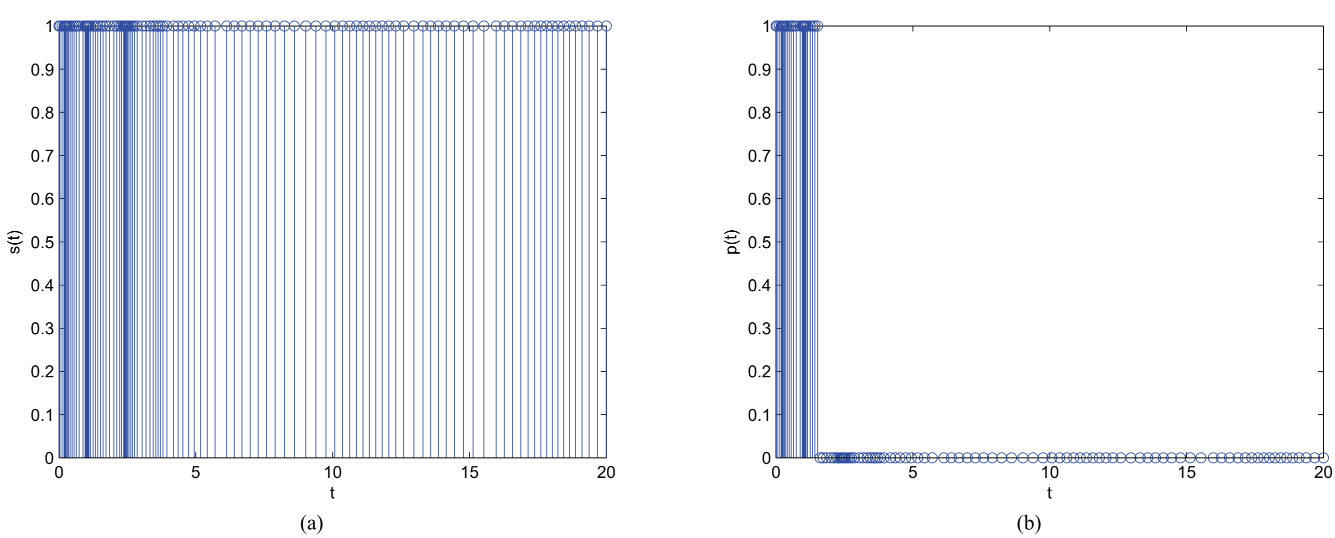


Fig. 10. (a) Original signal. (b) Encrypted signals by the chaotic memristive neural network in Example 2.

VI. Conclusion

In this paper, the problem of exponential lag adaptive synchronization control of MNNs was investigated via fuzzy method and applied in a pseudorandom number generator. A model of fuzzy MNNs was established with only two subsystems, and the update laws for the connection weights of slave systems and controller gain are designed to make the slave systems exponentially lag synchronized with the master systems. Illustrative examples were given to demonstrate the effectiveness of the obtained results, which can be extended into the field of PRNG as an encryption method.

In the future, there are some issues that deserve further investigation, such as 1) how to design the optimal update laws of the connection weights of slave systems to achieve desired results; 2) how to extend the applications of memristive neural networks into the fields of optimal computation, biological systems, and secure communications; and 3) how to deal with the problem of synchronization of memristive neural networks with discrete and distributed time-varying delays.

References

[1] R. Ananthanarayanan, S. Eser, H. Simon, and D. Modha, presented at the IEEE/ACM Conf. High Perform. Netw. Comput., Portland, OR, USA, Nov. 2009.

[2] L. Smith, *Handbook of Nature-Inspired and Innovative Computing: Integrating Classical Models with Emerging Technologies*. New York, NY, USA: Springer-Verlag, pp. 433–475.

[3] S. Jo, T. Chang, I. Ebong, B. Bhadviya, P. Mazumder, and W. Lu, "Nanoscale memristor device as synapse in neuromorphic systems," *Nanotech. Lett.*, vol. 10, pp. 1297–1301, 2010.

[4] D. Strukov, G. Snider, D. Stewart, and R. Williams, "The missing memristor found," *Nature*, vol. 453, pp. 80–83, 2008.

[5] L. Chua, "Memristor—The missing circuit element," *IEEE Trans. Circuits Theory*, vol. CT-18, no. 5, pp. 507–519, Sep. 1971.

[6] M. Sharifiy and Y. Banadaki, "General spice models for memristor and application to circuit simulation of memristor-based synapses and memory cells," *J. Circuits Syst. Comput.*, vol. 19, pp. 407–424, 2010.

[7] T. Choi, B. Shi, and K. Boahen, "An on–off orientation selective address event representation image transceiver chip," *IEEE Trans. Circuits Syst. I*, vol. 51, no. 2, pp. 342–353, Feb. 2004.

[8] G. Indiveri, "A neuromorphic VLSI device for implementing 2-D selective attention systems," *IEEE Trans. Neural Netw.*, vol. 12, no. 6, pp. 1455–1463, Nov. 2001.

[9] S. Liu and R. Douglas, "Temporal coding in a silicon network of integrate-and-fire neurons," *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1305–1314, Sep. 2004.

[10] G. Bao, S. Wen, and Z. Zeng, "Robust stability analysis of interval fuzzy Cohen–Grossberg neural networks with piecewise constant argument of generalized type," *Neural Netw.*, vol. 33, pp. 32–41, 2012.

[11] J. Cao, D. Huang, and Y. Qu, "Global robust stability of delayed recurrent neural networks," *Chaos Solit. Fract.*, vol. 23, pp. 221–229, 2005.

[12] Q. Song, "Synchronization analysis in an array of asymmetric neural networks with time-varying delays and nonlinear coupling," *Appl. Math. Comput.*, vol. 216, pp. 1605–1613, 2010.

[13] Q. Song, Z. Zhao, and J. Yang, "Passivity and passification for stochastic Takagi–Sugeno fuzzy systems with mixed time-varying delays," *Neurocomput.*, vol. 122, pp. 330–337, Dec. 2013.

[14] C. Li and G. Feng, "Delay-interval-dependent stability of recurrent neural networks with time-varying delay," *Neurocomput.*, vol. 72, pp. 1179–1183, 2009.

[15] M. Forti and P. Nistri, "Global convergence of neural networks with discontinous neuron activations," *IEEE Trans. Circuits Syst. I*, vol. 50, no. 11, pp. 1421–1435, Nov. 2003.

[16] C. Li, G. Feng, and X. Liao, "Stabilization of nonlinear system via periodically intermittent control," *IEEE Trans. Circuit Syst. II*, vol. 54, no. 11, pp. 1019–1023, Nov. 2007.

[17] Y. Xia, Z. Yang, and M. Han, "Lag synchronization of unknown chaotic delayed yang-yang-type fuzzy neural networks with noise perturbation based on adaptive control and parameter identification," *IEEE Trans. Neural Netw.*, vol. 20, no. 7, pp. 1165–1180, Jul. 2009.

[18] Y. Shen and J. Wang, "An improved algebraic criterion for global exponential stability of recurrent neural networks with time-varying delays," *IEEE Trans. Neural Netw.*, vol. 19, no. 3, pp. 528–531, Mar. 2008.

[19] Z. Zeng and J. Wang, "Global exponential stability of recurrent neural networks with time-varying delays in the presence of strong external stimuli," *Neural Netw.*, vol. 19, pp. 1528–1537, 2006.

[20] Z. Zeng, J. Wang, and X. Liao, "Global asmptotic stability and global exponential stability of neural networks with unbounded time-varying delays," *IEEE Trans. Circuits Syst. II*, vol. 52, no. 3, pp. 168–173, Mar. 2005.

[21] B. Pizzileo, L. Kang, G. W. Irwin, and W. Zhao, "Improved structure optimization for fuzzy-neural networks," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 6, pp. 1076–1089, Dec. 2012.

[22] R. J. Wai and Y. W. Lin, "Adaptive moving-target tracking control of a vision-based mobile robot via a dynamic petri recurrent fuzzy neural network," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 4, pp. 688–701, Aug. 2013.

[23] G.-D. Wu and P. H. Huang, "A vectorization-optimization-method-based type-2 fuzzyneural network for noisy data classification," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 1, pp. 1–15, Feb. 2013.

[24] J. Fu, H. G. Zhang, T. D. Mag, and Q. L. Zhang, "On passivity analysis for stochastic neural networks with interval time-varying delay," *Neurocomput.*, vol. 73, pp. 795–801, 2010.

[25] H. G. Zhang, L. L. Cai, and Z. Bien, "A fuzzy basis function vector-based multivariable adaptive controller for nonlinear systems," *IEEE Trans. Syst. Man Cybern. B*, vol. 30, no. 1, pp. 210–217, Feb. 2000.

[26] H. G. Zhang and D. R. Liu, *Fuzzy Modeling and Fuzzy Control*. Boston, MA, USA: Birkhäuser, 2006.

[27] H. G. Zhang, T. D. Ma, G. B. Huang, and Z. L. Wang, "Robust global exponential synchronization of uncertain chaotic delayed neural networks via dual-stage impulsive control," *IEEE Trans. Syst. Man Cybern. B*, vol. 40, no. 3, pp. 831–844, Jun. 2010.

[28] P. Balasubramaniam, K. Kalpana, and R. Rakkiyappan, "Linear matrix inequality approach for synchronization control of fuzzy cellular neural networks with mixed time delays," *Chin. Phys. B*, vol. 21, p. 048402, 2012.

[29] O. Kwon, M. Park, S. Lee, J. Park, and E. Cha, "Stability for neural networks with time-varying delays via some new approaches," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 2, pp. 181–193, Feb. 2013.

[30] Z. Wu, P. Shi, H. Su, and J. Chu, "Sampled-data synchronization of chaotic systems with time delays," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 24, no. 3, pp. 410–421, Mar. 2013.

[31] G. Wu and P. Huang, "A vectorization-optimization-method-based type-2 fuzzy neural network for noisy data classification," *IEEE Trans. Fuzzy Syst.*, vol. 21, no. 1, pp. 1–15, Feb. 2013.

[32] C. Juang, T. Chen, and W. Cheng, "Speedup of implementing fuzzy neural networks with high-dimensional inputs through parallel processing on graphic processing units," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 4, pp. 717–728, Aug. 2011.

[33] J. Cao, G. Chen, and P. Li, "Global synchronization in an array of delayed neural networks with hybrid coupling," *IEEE Trans. Syst. Man Cybern. B*, vol. 38, no. 2, pp. 488–498, Apr. 2008.

[34] M. Park, O. Kwon, J. Park, S. Lee, and E. Cha, "Synchronization criteria for coupled neural networks with interval time-varying delays and leakage delay," *Appl. Math. Comput.*, vol. 218, pp. 6762–6775, 2012.

[35] J. Li, H. Kazemian, and M. Afzal, "Neural network approaches for noisy language modeling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 11, pp. 1773–1784, Nov. 2013.

[36] H. Zhang, T. Ma, G. Huang, and Z. Wang, "Robust global exponential synchronization of uncertain chaotic delayed neural networks via dualstage impulsive control," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 40, no. 3, pp. 831–844, Jun. 2010.

[37] L. Chua and T. Roska, *Cellular Networks and Visual Computing: Foundations and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[38] G. Chen and X. Dong, *From Chaos to Order: Methodolgies, Perspectives, and Applications*. Singapore: World Scientific, 1998.

[39] Y. Xia, Z. Yang, and M. Han, "Synchronization schemes for coupled identical Yang-Yang type fuzzy cellular neural networks," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 14, pp. 3645–3659, 2009.

[40] Q. Song, "Design of controller on synchronization of chaotic neural networks with mixed time-varying delays," *Neurocomput.*, vol. 72, pp. 3288–3295, 2009.

[41] Q. Zhu and J. Cao, "Adaptive synchronization under almost every initial data for stochastic neural networks with time-varying delays and

distributed delays," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 16, pp. 2139–2159, 2011.

[42] M. Sun, C. Zeng, and L. Tian, "Linear generalized synchronization between two complex networks," *Commun. Nonlinear Sci. Numer.*, vol. 15, pp. 2162–2167, 2010.

[43] F. Breve, L. Zhao, M. Quiles, and E. Macau, "Chaotic phase synchronization and desynchronization in an oscillator network for object selection," *Neural Netw.*, vol. 22, pp. 728–737, 2009.

[44] W. Yu and J. Cao, "Adaptive Q-S (lag, anticipated, and complete) time-varying synchronization and parameters identification of uncertain delayed neural networks," *Chaos*, vol. 16, 023119, 2006.

[45] N. Gergel-Hackett, B. Hamadani, J. Suehle, C. Richter, C. Hacker, and D. Gundlach, "A flexible solution-processed memristor," *IEEE Electr. Device Lett.*, vol. 30, no. 7, pp. 706–708, Jul. 2009.

[46] C. Chiu and T. Chiang, "Robust output regulation of T-S fuzzy systems with multiple time-varying state and input delays," *IEEE Trans. Fuzzy Syst.*, vol. 17, no. 4, pp. 962–975, Aug. 2009.

[47] J. Dong, Y. Wang, and G. Yang, "Control synthesis of continuous-time T-S fuzzy systems with local nonlinear models," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 39, no. 5, pp. 1245–1258, Oct. 2009.

[48] X. Liu and S. Zhong, "T-S fuzzy model-based impulsive control of chaotic systems with exponential decay rate," *Phys. Lett. A*, vol. 370, pp. 260–264, 2007.

[49] C. Park and Y. Cho, "T-S model based indirect adaptive fuzzy control using online parameter estimation," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 34, no. 6, pp. 2293–2302, Dec. 2004.

[50] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modelling and control," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Jan./Feb. 1985.

[51] L. G. Wu, Z. G. Feng, and W. X. Zheng, "Exponential stability analysis for delayed neural networks with switching parameters: Average dwell time approach," *IEEE Trans. Neural Netw.*, vol. 21, no. 9, pp. 1396–1407, Sep. 2010.

[52] L. G. Wu, Z. G. Feng, and J. Lam, "Stability and synchronization of discrete-time neural networks with switching parameters and time-varying delays," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 12, pp. 1957–1972, Dec. 2013.

[53] L. G. Wu, X. J. Su, P. Shi, and J. B. Qiu, "A new approach to stability analysis and stabilization of discrete-time T-S fuzzy time-varying delay systems," *IEEE Trans. Syst. Man Cybern. B*, vol. 41, no. 1, pp. 273–286, Feb. 2011.

[54] A. Wu and Z. Zeng, "Exponential stabilization of memristive neural networks with time delays," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 12, pp. 1919–1929, Dec. 2012.

[55] S. Li and Z. Ge, "Fuzzy modeling and synchronization of two totally different chaotic systems via novel fuzzy model," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 41, no. 4, pp. 1015–1026, Aug. 2011.

[56] E. Fridman, A. Seuret, and J. P. Richard, "Robust sampled-data stabilization of linear systems: An input delay approach," *Automatica*, vol. 40, pp. 1441–1446, 2004.

[57] W. Zhao and Y. Tan, "Harmless delay for global exponential stability of Cohen–Grossberg neural networks," *Math. Comput. Simul.*, vol. 74, pp. 47–57, 2007.

[58] S. Arik, "Global robust stability analysis of neural networks with discrete time delays," *Chaos Solit. Fract.*, vol. 26, pp. 1407–1414, 2005.

[59] N. Ozcan and S. Arik, "Global robust stability analysis of neural networks with multiple time delays," *IEEE Trans. Circuits Syst. I*, vol. 53, no. 1, pp. 166–176, Jan. 2006.

[60] S. Wen and Z. Zeng, "Dynamics analysis of a class of memristor-based recurrent networks with time-varying delays in the presence of strong external stimuli," *Neural Process. Lett.*, vol. 35, pp. 47–59, 2012.

[61] A. Wu, S. Wen, and Z. Zeng, "Synchronization control of a class of memristor-based recurrent neural networks," *Inform. Sci.*, vol. 183, pp. 106–116, 2012.

[62] H. Zhang, Y. Xie, Z. Wang, and C. Zheng, "Adaptive synchronization between two different chaotic neural networks with time delay," *IEEE Trans. Neural Network*, vol. 18, no. 6, pp. 1841–1845, Nov. 2007.

[63] Y. Liu, Z. D. Wang, J. Liang, and X. Liu, "Synchronization and state estimation for discrete-time complex networks with distributed delays," *IEEE. Trans. Syst. Man Cybern. B Cybern.*, vol. 38, no. 5, pp. 1314–1325, Oct. 2008.

[64] S. Zhu, Y. Shen, and G. Chen, "Exponential Passivity of neural networks with time-varying delay and uncertainty," *Phys. Lett. A*, vol. 375, pp. 136–142, 2010.

[65] L. Blum, M. Blum, and M. Shub, "A simple unpredictable Pseudo-random number generator," *SIAM J. Comput.*, vol. 15, pp. 364–383, 1986.

[66] M. Falcioni, L. Palatella, and S. Pigolotti, "Properties making a chaotic system a good pseudo random number generator," *Phys. Rev. E*, vol. 72, 016220, 2005.