Neural Networks 101 (2018) 33–46

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Delay-dependent dynamical analysis of complex-valued memristive neural networks: Continuous-time and discrete-time cases

Jinling Wang [a], Haijun Jiang [a,*], Tianlong Ma [b], Cheng Hu [a]

[a] College of Mathematics and System Sciences, Xinjiang University, Urumqi, 830046, Xinjiang, China
[b] College of Basic Research, Qinghai University, Xining, 810016, Qinghai, China

ARTICLE INFO

Article history:
Received 22 October 2017
Received in revised form 21 December 2017
Accepted 30 January 2018
Available online 8 February 2018

Keywords:
Memristor
Complex-valued neural networks
Discontinuous activation functions
Matrix inequalities
Delay-dependent stability

ABSTRACT

This paper considers the delay-dependent stability of memristive complex-valued neural networks (MCVNNs). A novel linear mapping function is presented to transform the complex-valued system into the real-valued system. Under such mapping function, both continuous-time and discrete-time MCVNNs are analyzed in this paper. Firstly, when activation functions are continuous but not Lipschitz continuous, an extended matrix inequality is proved to ensure the stability of continuous-time MCVNNs. Furthermore, if activation functions are discontinuous, a discontinuous adaptive controller is designed to acquire its stability by applying Lyapunov–Krasovskii functionals. Secondly, compared with techniques in continuous-time MCVNNs, the Halanay-type inequality and comparison principle are firstly used to exploit the dynamical behaviors of discrete-time MCVNNs. Finally, the effectiveness of theoretical results is illustrated through numerical examples.

1. Introduction

Theoretically, the memristor as the fourth basic circuit element, was firstly postulated by Chua (1971) based on the integral theory of fundamental circuit in 1971. It has the unique electrical characteristics relative to resistor, capacitor and inductor. In 2008, researchers at HP's Laboratory implemented the physical model (Strukov, Snider, Stewart, & Williams, 2008) of the memristor, which means that it opens up new horizons for further development on circuit design. In 2009, the adaptive behavior of cells, which was similar to the property of the memristor, was proposed by means of the single-celled amoeba experiment (Pershin, Fontaine, & Ventra, 2009). Based on the experimental verification, more research results show that artificial neural networks with variable weights constructed by the memristor can better simulate human brain like associative memory functions. Further, these experiments check that memristive neural networks have the following advantages. As an analog element, memristor can realize the continuous update of synaptic weights and structure huge-scale integrated neural networks. In addition, the memristive neural network has larger storage capacities, stronger learning and memory abilities, and better information processing abilities in virtue of combining the advantages of memristor and cross array. Hence, in recent years, the study on the memristive neural network has become a hot spot in many fields (Corinto, Ascoli, & Gilli, 2011; Guo, Yang, & Wang, 2016; Wen, Zeng, & Huang, 2013; Yang, Guo, & Wang, 2015; Yang & Ho, 2016; Zhang, Shen, Yin, & Sun, 2013).

On the one hand, the in-depth analysis and study on complex-valued neural networks (CNNs) have been carried out in the complex plane, due to their potential engineering applications. Compared with real-valued neural networks (RNNs), the CNNs provide natural and reasonable ways to keep the physical characteristics of primitive problems in the complex domain (Xia & Feng, 2006). However, as an extension of RNNs, the main challenges we face are how to address the problems of complex-valued states and connection weights, especially complex-valued activation functions. Based on the Liouville's theorem, the activation function in CNNs cannot be both bounded and analytic while it is usually chosen to be a smooth bounded function in RNNs. Nowadays, the main approaches to analyze complex-valued activation functions can be categorized two ways. One is to separate it into the real and imaginary components (Hu & Wang, 2015; Nagamani & Ramasamy, 2015; Wang, Duan, Huang, Wang, & Li, 2016; Zhou & Song, 2013). The other way does not need to divide into two parts but should satisfy the Lipschitz continuity (Fang & Sun, 2014). However, these two methods have their own shortcomings which lead to the limitation of activation functions. For example, some complex-valued activation functions cannot be divided into two parts, and some are discontinuous. As is known to us, when the system is discontinuous, it is difficult to ensure the stability of systems (Abdurahman & Jiang, 2016; Aubin & Cellina, 1984; Aubin & Frankowska, 1990; Cai & Huang, 2017; Filippov, 1988; Wang, Li, & Huang, 2014). Therefore, combining with these cases, a novel

* Corresponding author.
E-mail address: jianghai@xju.edu.cn (H. Jiang).

https://doi.org/10.1016/j.neunet.2018.01.015
0893-6080/

mapping function is proposed to handle these problems and relax restrictions of complex-valued activation functions in this paper. Furthermore, if the activation function is discontinuous, a discontinuous adaptive controller is further designed to stabilize the MCVNNs.

On the other hand, it is clear that the delay-dependent stability of neural networks (NNs) is less conservative than delay-independent ones, since time-delay phenomena are often encountered in various practical situations and may have negative effect on system stability (Cao, Li, & Han, 2006; Liu, Wang, Tang, & Qiu, 2017; Mu & Chen, 2016; Park, Kwon, Park, Lee, & Cha, 2015; Seuret & Gouaisbaut, 2013; Seuret, Gouaisbaut, & Fridman, 2013; Xu, Lam, Ho, & Zou, 2005; Yang, Cao, & Lu, 2013; Yang, Guo, & Wang, 2016). Meanwhile, a series of results have been acquired on the delay-dependent stability of continuous-time NNs via the Lyapunov–Krasovskii functionals and linear matrix inequalities (LMIs). Overall, the major challenges are how to extend a less restrictive inequality and slack variables than ever before. In this paper, an extended matrix inequality is proved to guarantee the delay-dependent stability of continuous-time MCVNNs.

Actually, in many real-world applications, discrete-time counterparts of NNs are more applicable to image processing, pattern recognition and computer simulation. A multitude of numerical schemes can be used to obtain discrete-time counterparts of continuous-type NNs, for instance, Euler scheme, Runge–Kutta scheme. For the discrete case, the structure, properties and dynamics of NNs have been changed greatly. Nowadays, discrete-time NNs have been in the spotlight, and some results have been obtained based on summation inequalities and Lyapunov functions (Seuret, Gouaisbaut, & Fridman, 2015; Shao & Han, 2011; Wu & Zeng, 2012; Zhang, He, Jiang, Wang, & Wu, 2017; Zhang, Wang, & Liu, 2014). However, there is little work on the discrete-time MCVNNs since its complexity and discontinuity. Hence, the dynamical behaviors of discrete-time MCVNNs are also analyzed in this article. Compared with continuous-time MCVNNs, some new methods and conclusions are given to fill the gap of discrete-time MCVNNs.

Motivated by the above discussion, the aim of this paper is to consider the delay-dependent stability of continuous-time and discrete-time MCVNNs, and deal with the above-mentioned problems. The main contributions of this paper include four aspects:

(1) Different from previous works (Fang & Sun, 2014; Hu & Wang, 2015; Nagamani & Ramasamy, 2015; Wang et al., 2016; Zhou & Song, 2013), a novel linear mapping function is constructed to address the problem of MCVNNs in the complex plane, no matter whether complex-valued activation functions can be divided into the real and imaginary parts.

(2) In contrast with Seuret et al., (2013, 2015), two extended matrix inequalities with less restriction are given to guarantee the delay-dependent stability of both continuous-time and discrete-time MCVNNs via the Lyapunov–Krasovskii functional. Meanwhile, the assumption of complex-valued activation functions in MCVNNs is less conservative than Abdurahman & Jiang (2016), Cai & Huang (2017), Corinto et al. (2011) and Zhang et al. (2013).

(3) When activation functions are discontinuous, it is difficult to ensure the stability of the system, and a discontinuous adaptive controller is further designed to stabilize the continuous-time MCVNNs.

(4) Different from the methods of continuous-time cases, the Halanay-type inequality and comparison principle are firstly introduced to investigate dynamical behaviors of discrete-time MCVNNs.

The paper is organized as follows. Section 2 gives model description and preliminaries. In Section 3, sufficient conditions are obtained to ensure that the equilibrium point of the continuous-time system uniquely exists and is globally asymptotically stable. Similarly, based on a discontinuous adaptive controller, we show more solicitude on the case of activation functions which are discontinuous and also obtain its stability. The major objectives are to make the qualitative analysis on the discrete-time MCVNNs in Section 4. In Section 5, some examples with numerical simulation are given to demonstrate the effectiveness of the obtained results. The conclusions are given in Section 6.

Notation. Throughout this paper, let $\mathbb{Z}$, $\mathbb{Z}^+$, $\mathbb{R}$, $\mathbb{C}$ denote the set of all integers, positive integers, real numbers and complex numbers, respectively. $\mathbb{R}^n$ and $\mathbb{C}^n$ denote the n-dimensional Euclidean and unitary space. $\mathbb{R}^{n\times n}$ and $\mathbb{C}^{n\times n}$ are the set of $n\times n$ real matrix and the set of $n\times n$ complex matrix. $\mathcal{C}=\mathcal{C}([-\tau_2,0],\mathbb{R}^n)$ denotes function mapping $[-\tau_2,0]$ into $\mathbb{R}^n$. D^T denotes the transpose of matrix $D\in\mathbb{R}^{n\times n}$, E denotes the identity matrix. $\mathrm{i}=\sqrt{-1}$ represents the imaginary unit. $\|\cdot\|$ represents the Euclidean 2-norm. $\|z_t\|_C=\max_{\mu\in[0,\tau_2]}\{\|z(t-\mu)\|\}$, $\|z_n\|_C=\max_{\mu\in[0,l_2]_Z}\{\|z(n-\mu)\|\}$. For any vectors $x,y\in\mathbb{R}^n$, $x\le y$ represents $x_i\le y_i$ $(i=1,\dots,n)$. For $c,d\in\mathbb{Z}$ and $c\le d$, we denote the discrete interval $[c,d]_Z=\{c,c+1,\dots,d-1,d\}$. If $d=\infty$, then $[c,\infty)_Z=\{c,c+1,\dots\}$. Besides, $[r]$ denotes the integer part of the real number r.

2. Model description and preliminaries

In this section, a class of memristor-based neural networks (MNNs) are introduced systematically. By Kirchhoff's current law, the ith subsystem of MNNs can be written as

$$\dot{v}_i(t)=-d_iv_i(t)+\sum_{j=1}^{n}a_{ij}(v_i(t))f_j(v_j(t))+\sum_{j=1}^{n}b_{ij}(v_i(t))\times f_j(v_j(t-\tau(t)))+u_i,\ t\ge 0, \tag{1}$$

where $i\in\mathfrak{L}=\{1,2,\dots,n\}$, n corresponds to the number of units in the neural network; $v_i(t)$ is the voltage of the capacitor $\mathfrak{C}_i$; $d_i>0$ represents the neuron self-inhibitions; $f_j(v_j(t)), f_j(v_j(t-\tau(t)))$ are the functions without and with time delays; $\tau(t)$ corresponds to the time delay and $0\le\tau_1\le\tau(t)\le\tau_2$; u_i denotes the external input or bias, $a_{ij}(\cdot), b_{ij}(\cdot)$ are the memristor-based weights given by

$$a_{ij}(v_i(t))=\frac{\mathfrak{M}_{ij}}{\mathfrak{C}_i}\times \mathrm{sgn}_{ij},\quad b_{ij}(v_i(t))=\frac{\tilde{\mathfrak{M}}_{ij}}{\mathfrak{C}_i}\times \mathrm{sgn}_{ij},$$

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

where $\mathfrak{M}_{ij}$ is the memristor between the feedback function $f_j(v_j(t))$ and $v_i(t)$; $\tilde{\mathfrak{M}}_{ij}$ is the memristor between the feedback function $f_j(v_j(t-\tau(t)))$ and $v_i(t)$.

In this article, we will consider complex-valued networks due to its extensive applications. In the complex domain, complex-valued states, connection weights, and activation functions exist in the MNNs, and MCVNNs with time delays can be written as follows from (1)

$$\dot{z}_i(t)=-d_iz_i(t)+\sum_{j=1}^{n}a_{ij}(z_i(t))f_j(z_j(t))+\sum_{j=1}^{n}b_{ij}(z_i(t))\times f_j(z_j(t-\tau(t)))+u_i. \tag{2}$$

Eq. (2) also can be rewritten as the following matrix form

$$\begin{aligned}\dot{z}(t)&=-Dz(t)+A(z(t))f(z(t))+B(z(t))f(z(t-\tau(t)))+u,\\ &=F(t,z(t),z(t-\tau(t))),\end{aligned} \tag{3}$$

where $z(t)=(z_1(t),z_2(t),\dots,z_n(t))^T\in\mathbb{C}^n$; $D=\mathrm{diag}(d_1,d_2,\dots,d_n)^T\in\mathbb{R}^{n\times n}$; $A(z(t))=(a_{ij}(z_i(t)))\in\mathbb{C}^{n\times n}$, $B(z(t))=(b_{ij}(z_i(t)))\in\mathbb{C}^{n\times n}$; $f(\cdot)=(f_1(\cdot),f_2(\cdot),\dots,f_n(\cdot))^T:\mathbb{C}^n\to\mathbb{C}^n$; and $u=(u_1,u_2,\dots,u_n)^T\in\mathbb{C}^n$. As is well known, based on the feature of memristor