Journal of Applied Analysis and Computation
Volume 16, Number 3, June 2026, 1276–1297
Website:http://www.jaac-online.com
DOI:10.11948/20250262

# QUASI-SYNCHRONIZATION OF COMPLEX-VALUED NEUTRAL-TYPE INERTIAL NEURAL NETWORKS WITH PROPORTIONAL DELAYS*

**Yanxia Hu[1], Xiaofang Meng[1], Zhouhong Li[2,†] and Jinde Cao[3,4]**

**Abstract** This article explores the quasi-synchronization of complex-valued neutral-type inertial neural networks with proportional delays. By constructing appropriate Lyapunov functions and utilizing inequality techniques, sufficient conditions for the quasi-synchronization of the error system are derived. The entire analysis employs a non-reduced order approach and a non-separation method, it is directly concerned with the original system. A control law is designed, and it is proven that, under certain conditions, the error between the response system and the driving system can be bounded, thereby achieving quasi-synchronization. Finally, the effectiveness of these findings is thoroughly validated through numerical examples, and the research results are applied to the processes of image encryption and decryption. In terms of innovation, this paper focuses on models with parameter mismatch. It can provide theoretical references for the research on parameter-mismatched systems, and its analysis methods can also be applied to parameter-matched systems. At the same time, it provides more efficient analysis tools for complex-valued neural networks and further advances the theoretical foundations of quasi-synchronization control.

**Keywords** Complex-valued neural networks, quasi-synchronization, inertial term, non-reduced order approach, non-separation method.

**MSC(2010)** 34K40, 34K26, 92B25.

## 1. Introduction

In the advancement of contemporary science and technology, a neural network is an information processing system that mimics the human brain's structure and function, and is crucial for information processing and intelligent computing. Due to its unique advantages, including associative memory, low error rate, strong self-learning ability, and parallel computing, it has been widely applied in various fields. For example, it has been successfully used in multiple

†The corresponding author.
[1]School of Statistics and Mathematics, Yunnan University of Finance and Economics, Kunming, Yunnan 650221, China
[2]Department of Mathematics, Yuxi Normal University, Yuxi, Yunnan 653100, China
[3]School of Mathematics, Southeast University, Nanjing 211189, China
[4]Purple Mountain Laboratories, Nanjing 211111, China
*This work is supported by Yunnan Fundamental Research Projects under Grant 202501AT070454, in part by the Key Laboratory of Complex Dynamics System and Application Analysis of the Department of Education of Yunnan Province, and the Special Basic Cooperative Research Programs of Yunnan Provincial Undergraduate Universities' Association under Grant 202401BA070001-158, and the National Natural Sciences Foundation of the People's Republic of China under Grant 62576098, and Key Project of Guangxi Natural Science Foundation under Grant 2025GXNSFDA069040 and the Yunnan Key Laboratory of Smart City in Cyberspace Security.
Email: yxhu828@163.com(Y. Hu), xfmeng@ynufe.edu.cn(X. Meng), zhouhli@yeah.net(Z. Li), jdcao@seu.edu.cn(J. Cao)

fields, including image recognition and processing [26, 32], medical analysis and diagnosis [3], state estimation [35], and automatic control [4], among others. In addition, as noted in [27], neural networks and their algorithms, being the cornerstone of artificial intelligence, form the foundation of deep learning and have permeated all aspects of life, with broad development prospects.

As early as 1986, Babcock and Westervelt began to focus on neural networks with inertia terms, and they first proposed the concept of inertial neural networks. They demonstrated inertia characteristics and dynamic models by introducing inductors into neural circuits, and in 1987, they proposed inertial neural networks with time delays [1]. Inertial neural networks are a special type of neural network expressed through second-order differential equations. The second-order term is referred to as the inertial term. Introducing the inertial term into the neural networks endows the system with more complex dynamic characteristics. The inertia term through the inductance in the circuit not only helps neurons to have a disordered search of memory, but also can accurately simulate the biological characteristics of neurons. Due to its fast convergence speed, strong storage capacity, and fault tolerance, it shows broad application prospects in multiple fields. In a large number of previous studies, most of which have equivalently transformed inertial neural networks into first-order neural networks through appropriate variable substitutions [15, 19, 34, 41, 43]. In [15], the research results were also applied to the field of secure communication. In [19], two kinds of mixed delays, namely time-varying delay and coupling delay, were considered. In [34], the fuzzy inertial neural networks were studied through intermittent control and applied to image encryption. Although the order reduction method is effective, it increases the difficulty of theoretical analysis. In recent years, the non-order reduction method has become a popular and novel approach for studying the dynamic behavior of inertial neural networks [7–9, 20, 42]. Compared to the traditional order reduction method, this method directly analyzes the original high-order inertial neural networks, which not only avoids increasing the system's dimension but also retains the system's complex dynamic characteristics completely.

In 1990, Clarke first proposed the complex-valued neural networks [6]. Since the complex-valued system defines neurons, connection weights, and activation functions in the sense of the complex domain. When dealing with complex data, it offers superior information storage capacity and faster computing speed compared to real-valued neural networks. Therefore, complex-valued neural networks have attracted widespread attention [5, 24, 44]. Complex-valued neural networks are widely used due to their superiority in handling more complex data, such as image processing, signal processing, etc. [16]. In previous studies, the synchronization and stability of complex-valued neural networks have been typically analyzed using the separation method, where the complex-valued system is decomposed into two real-valued subsystems. Then the real and imaginary parts were discussed separately [13, 23, 29, 40]. Although this method is feasible, it makes theoretical research more difficult. Therefore, the non-separation method has become a novel approach for analyzing the dynamic behavior of complex-valued neural networks. This method is more concise and retains the system characteristics more completely [8, 10, 21, 22].

It is well known that synchronization is crucial in the study of the dynamic behavior of neural networks, and it has significant implications for their applications in real-life scenarios, such as image encryption [17, 38] and communication security [28]. Time delays may have a significant impact on the synchronization and stability of a system. There are various types of time delays, including distributed time delay, leakage delay, proportional delay, and neutral-

type delay, among others. Unlike other time delays, proportional delay is an unbounded and monotonically increasing time delay. The unbounded characteristic has caused many problems in system dynamics [11]. Furthermore, neutral-type delays play a key role in automatic control and population dynamics. Therefore, it is considered to introduce neutral-type delays into neural networks [30]. Parameter mismatch is unavoidable in practical systems, complicating synchronization. Studying its effects can lead to more robust control strategies. Additionally, proportional lag and parameter mismatch further complicate the synchronization process [14]. Overcoming these challenges can advance theory and aid in the development of new control methods and algorithms for various applications.

Based on the above discussion, the research motivation of this article can be summarized as follows: Firstly, this paper focuses on the complex-valued neutral-type inertial neural network models. In addition, we considered systems with parameter mismatches, and dealing with these mismatches is a challenge we face. Meanwhile, due to the varying lengths of synchronization time, neural network synchronization is categorized into several types, including exponential, polynomial, and finite-time synchronization, among others. However, quasi-synchronization only requires convergence to an acceptable error range. This advantage makes it more valuable for both theoretical research and practical applications than other synchronization studies [12, 14, 18, 33, 39]. In [18], the authors propose a novel image encryption framework that integrates preassigned-time control and projective synchronization. In [33], quasi-synchronization is used instead of complete synchronization for analysis. Based on the above discussion, research on quasi-synchronization is meaningful. Secondly, for the quasi-synchronization research of complex-valued inertial neural networks, the inertial term is handled by the non-reduction method without separating the complex variables. Finally, constructing an appropriate Lyapunov function and applying inequality techniques simplifies the complex derivation process. The research results presented in this paper will provide new ideas and a theoretical foundation for future research.

However, no progress has been made in quasi-synchronization for complex-valued neutral-type inertial neural networks with proportional delays. Compared with previous studies, the main contributions of this paper are as follows:

(1) A proportional time delay is introduced into the complex-valued neutral inertial neural network, which expands the scope of application of the existing neural network model. The influence of parameter mismatch on synchronization is systematically studied, aligning more closely with practical applications.

(2) The non-reduced order approach is adopted to handle the inertia term, and the non-separation method to handle the complex-valued problem. The original system is analyzed directly, which retains the system characteristics more completely and overcomes the problem that traditional methods may lose information when dealing with complex systems.

(3) Under parameter mismatch and proportional time delay, complete synchronization in the strict sense is difficult to achieve; however, quasi-synchronization with bounded error can be achieved through a control strategy. The Lyapunov function, which is independent of time delay, is employed to overcome the analytical difficulties arising from the coupling of a complex-valued system, a neutral term, and a proportional time delay. The explicit relationship between the synchronization error bound and the proportional time delay parameter, as well as the degree of parameter mismatch, is derived, and the sufficient conditions for the network to achieve quasi-synchronization are provided.

(4) Numerical simulations have validated the applicability of the theoretical results for black-

and-white image encryption. Furthermore, these simulations demonstrate that our encryption algorithm provides robust security for image protection.

The paper is structured as follows: Section 2 covers the model description and preliminary work. Section 3 analyzes the quasi-synchronization conditions of complex-valued neutral-type inertial neural networks. Section 4 provides a numerical example to verify the validity of the results and gives its application in black and white image encryption, and Section 5 summarizes the conclusions.

**Notations.** The sets of all complex and $n$-dimensional complex are represented by $\mathbb{C}$ and $\mathbb{C}^n$, $\mathbb{N}$ represents the set of $\{1, 2, \cdots, n\}$. For any $z = z^R + \mathbf{i}z^I \in \mathbb{C}$, $z^R$ and $z^I$ respectively denote the real and imaginary parts of it, $z^*$ is the conjugate of $z$, $\|z\|_0 = \sqrt{zz^*}$ is the zero-norm of $z$. For $Z = (z_1, z_2, \cdots, z_n) \in \mathbb{C}^n$, $\|Z\| = \max\limits_{i\in\mathbb{N}} \|z_i\|_0$, $\tilde{z} = \|z\|_0^2$ and $\bar{z} = \|z - \hat{z}\|_0^2$. We will use the following notation for convenience: $\tilde{c}_{ij} = \|c_{ij}\|_0^2$, $\bar{c}_{ij} = \|c_{ij} - \hat{c}_{ij}\|_0^2$, $\tilde{d}_{ij} = \|d_{ij}\|_0^2$, $\bar{d}_{ij} = \|d_{ij} - \hat{d}_{ij}\|_0^2$, $\tilde{e}_{ij} = \|e_{ij}\|_0^2$, $\bar{e}_{ij} = \|e_{ij} - \hat{e}_{ij}\|_0^2$.

## 2. Model description and preliminary analysis

This section provides a model description of the driving system, response system, and error system, as well as the lemma definitions and conditions used in this paper.

Consider the following complex-valued neutral-type inertial neural networks with proportional delays as the driving system given by

$$\begin{aligned}\ddot{x}_i(t) = &-\hat{a}_i\dot{x}_i(t) - \hat{b}_i x_i(t) + \sum_{j=1}^{n} \hat{c}_{ij}\mathcal{G}_j(x_j(t)) + \sum_{j=1}^{n} \hat{d}_{ij}\mathcal{G}_j(x_j(\gamma_j t)) \\ &+ \sum_{j=1}^{n} \hat{e}_{ij}\mathcal{G}_j(\dot{x}_j(\gamma_j t)) + \mathcal{I}_i, \end{aligned} \tag{2.1}$$

where $i \in \mathbb{N}$; $x_i(t) \in \mathbb{C}$ are the state variables and the second derivative is inertial term, $\hat{a}_i > 0$ and $\hat{b}_i > 0$ are constants, $\hat{c}_{ij}$, $\hat{d}_{ij}$, $\hat{e}_{ij}$ denote complex-valued connection weights, $\mathcal{G}_j(\cdot)$ denotes the complex-valued activation functions, $\gamma_j$ is the proportional delay factor with $0 < \gamma_j \leq 1$, $\tilde{\gamma} = \min\limits_{j\in\mathbb{N}}\{\gamma_j\}$, $\mathcal{I}_i$ is the external input, the initial conditions of (2.1) are $x_i(s) = \varphi_i(s)$, $\dot{x}_i(s) = \psi_i(s)$, $s \in [\tilde{\gamma}t_0, t_0]$, in which $\varphi_i(s)$ and $\psi_i(s)$ are continuous.

The corresponding response system is given by

$$\begin{aligned}\ddot{y}_i(t) = &-a_i\dot{y}_i(t) - b_i y_i(t) + \sum_{j=1}^{n} c_{ij}\mathcal{G}_j(y_j(t)) + \sum_{j=1}^{n} d_{ij}\mathcal{G}_j(y_j(\gamma_j t)) \\ &+ \sum_{j=1}^{n} e_{ij}\mathcal{G}_j(\dot{y}_j(\gamma_j t)) + \mathcal{I}_i + u_i(t), \end{aligned} \tag{2.2}$$

where $a_i > 0$ and $b_i > 0$ are constants, $c_{ij}$, $d_{ij}$, $e_{ij}$ denote complex-valued connection weights, $u_i(t) \in \mathbb{C}$ is the control input that will be designed. The initial conditions of (2.2) are $y_i(s) = \tilde{\varphi}_i(s)$, $\dot{y}_i(s) = \tilde{\psi}_i(s)$, $s \in [\tilde{\gamma}t_0, t_0]$, in which $\tilde{\varphi}_i(s)$ and $\tilde{\psi}_i(s)$ are continuous.

Denote $z_i(t) = y_i(t) - x_i(t)$, the error system is given by

$$\ddot{z}_i(t) = -a_i\dot{z}_i(t) - b_i z_i(t) + \sum_{j=1}^{n} \left[c_{ij}\mathcal{F}_j(z_j(t)) + d_{ij}\mathcal{F}_j(z_j(\gamma_j t)) + e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\right]$$

$$-(a_i-\hat{a}_i)\dot{x}_i(t)-(b_i-\hat{b}_i)x_i(t)+\sum_{j=1}^{n}\big[(c_{ij}-\hat{c}_{ij})\mathcal{G}_j(x_j(t))$$
$$+(d_{ij}-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]+u_i(t), \tag{2.3}$$

where the initial conditions of (2.3) are $z_i(s)=\tilde{\varphi}_i(s)-\varphi_i(s)$, $\dot{z}_i(s)=\tilde{\psi}_i(s)-\psi_i(s)$, $\mathcal{F}_j(z_j)=\mathcal{G}_j(y_j)-\mathcal{G}_j(x_j)$.

**Remark 2.1.** In system (2.1), $\gamma_j t=(1-\gamma_j)t$ represents the proportional delay, and $\lim\limits_{t\to+\infty}\gamma_j t=+\infty$. The proportional delay is an unbounded, monotonically increasing delay that varies over time. Due to this characteristic, the derivation process of quasi-synchronization of a complex-valued neutral-type inertial neural network with proportional delay is quite difficult. The proportional delay is time-dependent, allowing the network's operating time to be controlled. The results of this paper provide theoretical support and new methods for subsequent related research.

Based on the description of network models (2.3), in this section, some relevant lemmas and definitions will be given.

**Remark 2.2.** Easy to know, for any $x,y\in\mathbb{C}$, the following equations can be obtained:

$$(x+y)^*=x^*+y^*,\ (xy)^*=y^*x^*.$$

**Lemma 2.1.** [37] *For $x,y\in\mathbb{C}$, $\varepsilon>0$, then, the following inequality holds:*

$$x^*y+y^*x\le\varepsilon x^*x+\frac{1}{\varepsilon}y^*y.$$

**Definition 2.1.** [39] For any positive constant $\phi$, if there exist $t>0$, satisfy $\|z(t)\|\le\phi$ and $\|\dot{z}(t)\|\le\phi$, the error system (2.3) is quasi-stable.

In this section, we assume the following conditions:

($H_1$) For any $j\in\mathbb{N}$, $x_j$, $y_j\in\mathbb{C}$, there exist some positive numbers $\mathcal{L}_j$, $\mathcal{M}_j$, satisfy the following inequalities:

$$\big[\mathcal{G}_j(y_j)-\mathcal{G}_j(x_j)\big]^*\big[\mathcal{G}_j(y_j)-\mathcal{G}_j(x_j)\big]\le\mathcal{L}_j(y_j-x_j)^*(y_j-x_j),$$
$$\mathcal{G}_j^*(x_j)\mathcal{G}_j(x_j)\le\mathcal{M}_j.$$

($H_2$) For any $x_i$, $\dot{x}_i\in\mathbb{C}$, there exist some positive numbers $\delta_i$, $\rho_i$, satisfy the following inequalities:

$$\|x_i\|_0\le\delta_i,\ \|\dot{x}_i\|_0\le\rho_i.$$

**Remark 2.3.** The conventional analysis approaches for complex-valued neural networks typically employ either the separation method or the non-separation method. Compared with directly dividing the original complex-valued systems into two real-valued subsystems of the real part and the imaginary part. The non-separation method not only retains the original system's characteristics completely, but also significantly reduces the complexity of the derivation. In this paper, we consider analyzing model (2.3) as a whole in the complex domain. By using the non-separation method, we have obtained the desired conclusion.

**Remark 2.4.** In the study of neural networks with inertial terms, many studies adopt the reduction method [15, 19, 34, 41, 43]. The reduction method requires the introduction of new parameters for variable substitution, increasing the system's dimension. More importantly, this method complicates the derivation and analysis. The controller designed in this paper is aimed at the original system rather than the first-order transformed system obtained by reduction. Therefore, the non-reduction method is more in line with the requirements of practical applications.

## 3. Quasi-synchronization

In this section, we will design a suitable controller based on the above description of the drive system, response system, and error system. Then, we will use the lemma and assumptions to obtain the quasi-stability of the error system (2.3).

To the driving system (2.1) and the response system (2.2) to be quasi-synchronized, the controller of the error system is designed as:

$$u_i(t) = -\alpha_i \dot{z}_i(t) - \beta_i z_i(t), \tag{3.1}$$

where $\alpha_i, \beta_i > 0$ are the gain coefficient to be designed.

**Theorem 3.1.** *Assumptions* $(H_1)$-$(H_2)$ *hold, further assumption is made as follows:*

$(H_3)$ $\mathcal{H}_i = \min\{\mathcal{A}_i, \mathcal{C}_i\} > 0$, $\mathcal{B}_i > \mathcal{H}_i > 0$, $\hat{a}_i > a_i$ *and* $\hat{b}_i > b_i$,

*where*

$$\begin{aligned}
\mathcal{A}_i &= a_i + \alpha_i + b_i + \beta_i - 1,\\
\mathcal{B}_i &= 2(b_i + \beta_i) + (a_i - \hat{a}_i) + (b_i - \hat{b}_i)\\
&\quad - \sum_{j=1}^{n} \left(\tilde{c}_{ij} + \tilde{d}_{ij} + \tilde{e}_{ij} + \bar{c}_{ij} + \bar{d}_{ij} + \bar{e}_{ij}\right) - 4n\mathcal{L}_i,\\
\mathcal{C}_i &= 2(a_i + \alpha_i) + (a_i - \hat{a}_i) + (b_i - \hat{b}_i)\\
&\quad - \sum_{j=1}^{n} \left(\tilde{c}_{ij} + \tilde{d}_{ij} + \tilde{e}_{ij} + \bar{c}_{ij} + \bar{d}_{ij} + \bar{e}_{ij}\right) - 2n\mathcal{L}_i - 2,
\end{aligned}$$

*then the driving system* (2.1) *and the response system* (2.2) *are quasi-synchronized under the controller* (3.1).

**Proof.** Consider the Lyapunov function

$$V(t) = \max_{i\in\mathbb{N}} \left[\left(z_i(t) + \dot{z}_i(t)\right)\left(z_i(t) + \dot{z}_i(t)\right)^*\right].$$

Calculating the derivative of $V(t)$ along the solution of (2.3), we obtain

$$\begin{aligned}
\dot{V}(t) &= \max_{i\in\mathbb{N}} \Big[\dot{z}_i(t)z_i^*(t) + z_i(t)\dot{z}_i^*(t) + 2\dot{z}_i(t)\dot{z}_i^*(t) + z_i(t)\ddot{z}_i^*(t)\\
&\quad + \ddot{z}_i(t)z_i^*(t) + \ddot{z}_i(t)\dot{z}_i^*(t) + \dot{z}_i(t)\ddot{z}_i^*(t)\Big]\\
&= \max_{i\in\mathbb{N}} \Big\{\dot{z}_i(t)z_i^*(t) + z_i(t)\dot{z}_i^*(t) + 2\dot{z}_i(t)\dot{z}_i^*(t) + z_i(t)\Big( - a_i\dot{z}_i(t) - b_i z_i(t)
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{j=1}^{n}\big[c_{ij}\mathcal{F}_j(z_j(t))+d_{ij}\mathcal{F}_j(z_j(\gamma_j t))+e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\big]-(a_i-\hat{a}_i)\dot{x}_i(t)\\
&-(b_i-\hat{b}_i)x_i(t)+\sum_{j=1}^{n}\big[(c_{ij}-\hat{c}_{ij})\mathcal{G}_j(x_j(t))+(d_{ij}-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))\\
&+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]+u_i(t)\Big)^{*}+\Big(-a_i\dot{z}_i(t)-b_i z_i(t)\\
&+\sum_{j=1}^{n}\big[c_{ij}\mathcal{F}_j(z_j(t))+d_{ij}\mathcal{F}_j(z_j(\gamma_j t))+e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\big]-(a_i-\hat{a}_i)\dot{x}_i(t)\\
&-(b_i-\hat{b}_i)x_i(t)+\sum_{j=1}^{n}\big[(c_{ij}-\hat{c}_{ij})\mathcal{G}_j(x_j(t))+(d_{ij}-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))\\
&+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]+u_i(t)\Big)z_i^{*}(t)+\Big(-a_i\dot{z}_i(t)-b_i z_i(t)\\
&+\sum_{j=1}^{n}\big[c_{ij}\mathcal{F}_j(z_j(t))+d_{ij}\mathcal{F}_j(z_j(\gamma_j t))+e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\big]-(a_i-\hat{a}_i)\dot{x}_i(t)\\
&-(b_i-\hat{b}_i)x_i(t)+\sum_{j=1}^{n}\big[(c_{ij}-\hat{c}_{ij})\mathcal{G}_j(x_j(t))+(d_{ij}-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))\\
&+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]+u_i(t)\Big)\dot{z}_i^{*}(t)+\dot{z}_i(t)\Big(-a_i\dot{z}_i(t)-b_i z_i(t)\\
&+\sum_{j=1}^{n}\big[c_{ij}\mathcal{F}_j(z_j(t))+d_{ij}\mathcal{F}_j(z_j(\gamma_j t))+e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\big]-(a_i-\hat{a}_i)\dot{x}_i(t)\\
&-(b_i-\hat{b}_i)x_i(t)+\sum_{j=1}^{n}\big[(c_{ij}-\hat{c}_{ij})\mathcal{G}_j(x_j(t))+(d_{ij}-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))\\
&+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]+u_i(t)\Big)^{*}\Big\}\\
&=\max_{i\in\mathbb{N}}\Big\{\big[1-(a_i+\alpha_i)-(b_i+\beta_i)\big]\dot{z}_i(t)z_i^{*}(t)+\big[1-(a_i+\alpha_i)-(b_i\\
&+\beta_i)\big]z_i(t)\dot{z}_i^{*}(t)+(2-2a_i-2\alpha_i)\dot{z}_i(t)\dot{z}_i^{*}(t)-2(b_i+\beta_i)z_i(t)z_i^{*}(t)\\
&+z_i(t)\Big(\sum_{j=1}^{n}\big[c_{ij}\mathcal{F}_j(z_j(t))+d_{ij}\mathcal{F}_j(z_j(\gamma_j t))+e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\big]\Big)^{*}\\
&+\Big(\sum_{j=1}^{n}\big[c_{ij}\mathcal{F}_j(z_j(t))+d_{ij}\mathcal{F}_j(z_j(\gamma_j t))+e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\big]\Big)z_i^{*}(t)\\
&+\dot{z}_i(t)\Big(\sum_{j=1}^{n}\big[c_{ij}\mathcal{F}_j(z_j(t))+d_{ij}\mathcal{F}_j(z_j(\gamma_j t))+e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\big]\Big)^{*}\\
&+\Big(\sum_{j=1}^{n}\big[c_{ij}\mathcal{F}_j(z_j(t))+d_{ij}\mathcal{F}_j(z_j(\gamma_j t))+e_{ij}\mathcal{F}_j(\dot{z}_j(\gamma_j t))\big]\Big)\dot{z}_i^{*}(t)\\
&-(a_i-\hat{a}_i)\big[\dot{x}_i(t)z_i^{*}(t)+z_i(t)\dot{x}_i^{*}(t)\big]-(b_i-\hat{b}_i)\big[x_i(t)z_i^{*}(t)+z_i(t)x_i^{*}(t)\big]\\
&-(a_i-\hat{a}_i)\big[\dot{x}_i(t)\dot{z}_i^{*}(t)+\dot{z}_i(t)\dot{x}_i^{*}(t)\big]-(b_i-\hat{b}_i)\big[x_i(t)\dot{z}_i^{*}(t)+\dot{z}_i(t)x_i^{*}(t)\big]
\end{aligned}
$$

$$
\begin{aligned}
&+z_i(t)\Bigg(\sum_{j=1}^{n}\big[(c_{ij}-\hat{c}_{ij})\mathcal{G}_j(x_j(t))+(d_{ij}-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))\\
&+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]\Bigg)^*+\Bigg(\sum_{j=1}^{n}\big[(c_{ij}-\hat{c}_{ij})\mathcal{G}_j(x_j(t))+(d_{ij}\\
&-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]\Bigg)z_i^*(t)+\dot{z}_i(t)\Bigg(\sum_{j=1}^{n}\big[(c_{ij}\\
&-\hat{c}_{ij})\mathcal{G}_j(x_j(t))+(d_{ij}-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]\Bigg)^*\\
&\times\Bigg(\sum_{j=1}^{n}\big[(c_{ij}-\hat{c}_{ij})\mathcal{G}_j(x_j(t))+(d_{ij}-\hat{d}_{ij})\mathcal{G}_j(x_j(\gamma_j t))\\
&+(e_{ij}-\hat{e}_{ij})\mathcal{G}_j(\dot{x}_j(\gamma_j t))\big]\Bigg)\dot{z}_i^*(t)\Bigg\}. \qquad (3.2)
\end{aligned}
$$

It can be obtained from (3.2), $(H_1)$ and Lemma 2.1 that

$$
\begin{aligned}
\dot{V}(t)\le\max_{i\in\mathbb{N}}\Bigg\{\Big[&-2(b_i+\beta_i)+\sum_{j=1}^{n}(\tilde{c}_{ij}+\tilde{d}_{ij}+\tilde{e}_{ij}+\bar{c}_{ij}+\bar{d}_{ij}+\bar{e}_{ij})\\
&+4n\mathcal{L}_i\Big]z_i(t)z_i^*(t)+\big[1-(a_i+\alpha_i)-(b_i+\beta_i)\big]\big(\dot{z}_i(t)z_i^*(t)\\
&+z_i(t)\dot{z}_i^*(t)\big)+\Big[2-2a_i-2\alpha_i+\sum_{j=1}^{n}(\tilde{c}_{ij}+\tilde{d}_{ij}+\tilde{e}_{ij}+\bar{c}_{ij}+\bar{d}_{ij}\\
&+\bar{e}_{ij})+2n\mathcal{L}_i\Big]\dot{z}_i(t)\dot{z}_i^*(t)-(a_i-\hat{a}_i)\big[\dot{x}_i(t)z_i^*(t)+z_i(t)\dot{x}_i^*(t)\big]\\
&-(b_i-\hat{b}_i)\big[x_i(t)z_i^*(t)+z_i(t)x_i^*(t)\big]-(a_i-\hat{a}_i)\big[\dot{x}_i(t)\dot{z}_i^*(t)\\
&+\dot{z}_i(t)\dot{x}_i^*(t)\big]-(b_i-\hat{b}_i)\big[x_i(t)\dot{z}_i^*(t)+\dot{z}_i(t)x_i^*(t)\big]+6n\mathcal{M}_i\Bigg\}. \qquad (3.3)
\end{aligned}
$$

From $\hat{a}_i>a_i$, $\hat{b}_i>b_i$ and (3.3), $\dot{V}(t)$ can be future written as

$$
\begin{aligned}
\dot{V}(t)\le\max_{i\in\mathbb{N}}\Bigg\{\Big[&-2(b_i+\beta_i)+(\hat{a}_i-a_i)+(\hat{b}_i-b_i)+\sum_{j=1}^{n}(\tilde{c}_{ij}+\tilde{d}_{ij}+\tilde{e}_{ij}\\
&+\bar{c}_{ij}+\bar{d}_{ij}+\bar{e}_{ij})+4n\mathcal{L}_i\Big]z_i(t)z_i^*(t)+\big[1-(a_i+\alpha_i)-(b_i\\
&+\beta_i)\big]\big(\dot{z}_i(t)z_i^*(t)+z_i(t)\dot{z}_i^*(t)\big)+\Big[2-2a_i-2\alpha_i+(\hat{a}_i-a_i)+(\hat{b}_i\\
&-b_i)+\sum_{j=1}^{n}(\tilde{c}_{ij}+\tilde{d}_{ij}+\tilde{e}_{ij}+\bar{c}_{ij}+\bar{d}_{ij}+\bar{e}_{ij})+2n\mathcal{L}_i\Big]\dot{z}_i(t)\dot{z}_i^*(t)\Bigg\}\\
&+\max_{i\in\mathbb{N}}\big[2(\hat{b}_i-b_i)x_i(t)x_i^*(t)+2(\hat{a}_i-a_i)\dot{x}_i(t)\dot{x}_i^*(t)+6n\mathcal{M}_i\big]. \qquad (3.4)
\end{aligned}
$$

It can be obtained from $(H_2)$, $(H_3)$ and (3.4) that

$$
\dot{V}(t)\le\max_{i\in\mathbb{N}}\Bigg\{\Big[-2(b_i+\beta_i)+(\hat{a}_i-a_i)+(\hat{b}_i-b_i)+\sum_{j=1}^{n}(\tilde{c}_{ij}+\tilde{d}_{ij}+\tilde{e}_{ij}
$$

$$
\begin{aligned}
&+\bar{c}_{ij}+\bar{d}_{ij}+\bar{e}_{ij})+4n\mathcal{L}_i\Big]z_i(t)z_i^*(t)+\big[1-(a_i+\alpha_i)\\
&-(b_i+\beta_i)\big]\big(\dot{z}_i(t)z_i^*(t)+z_i(t)\dot{z}_i^*(t)\big)+\Big[2-2a_i-2\alpha_i+(\hat{a}_i-a_i)\\
&+(\hat{b}_i-b_i)+\sum_{j=1}^{n}\big(\tilde{c}_{ij}+\tilde{d}_{ij}+\tilde{e}_{ij}+\bar{c}_{ij}+\bar{d}_{ij}+\bar{e}_{ij}\big)+2n\mathcal{L}_i\Big]\dot{z}_i(t)\dot{z}_i^*(t)\Big\}\\
&+\max_{i\in\mathbb{N}}\big[6n\mathcal{M}_i+2(\hat{a}_i-a_i)\rho_i^2+2(\hat{b}_i-b_i)\delta_i^2\big]\\
\le&\max_{i\in\mathbb{N}}\big\{-\mathcal{A}_i(\dot{z}_i(t)z_i^*(t)+z_i(t)\dot{z}_i^*(t))-\mathcal{B}_iz_i(t)z_i^*(t)-\mathcal{C}_i\dot{z}_i(t)\dot{z}_i^*(t)\big\}+\mathcal{D}\\
\le&\max_{i\in\mathbb{N}}\big\{-\mathcal{H}_i(z_i(t)+\dot{z}_i(t))(z_i(t)+\dot{z}_i(t))^*-(\mathcal{B}_i-\mathcal{H}_i)z_i(t)z_i^*(t)\big\}+\mathcal{D}\\
\le&\max_{i\in\mathbb{N}}\big\{-\mathcal{H}_i(z_i(t)+\dot{z}_i(t))(z_i(t)+\dot{z}_i(t))^*\big\}+\mathcal{D}\\
\le&-\mathcal{H}V(t)+\mathcal{D}.
\end{aligned}
\tag{3.5}
$$

From (3.5), we can obtain that

$$\dot{V}(t)+\mathcal{H}V(t)\le\mathcal{D}.$$

Then using integral factors,

$$\int_{t_0}^{t}\big(e^{\mathcal{H}t}\cdot V(t)\big)'\mathrm{d}t\le\mathcal{D}\int_{t_0}^{t}e^{\mathcal{H}t}\mathrm{d}t.$$

Then,

$$V(t)\cdot e^{\mathcal{H}t}\le V(t_0)\cdot e^{\mathcal{H}t_0}+\frac{\mathcal{D}}{\mathcal{H}}\cdot e^{\mathcal{H}t}.$$

Let $t_0\to-\infty$, then $e^{\mathcal{H}t_0}\to 0$

$$V(t)\le\frac{\mathcal{D}}{\mathcal{H}},\ t\ge t_0.$$

From $V(t)=\max\limits_{i\in\mathbb{N}}\big[(z_i(t)+\dot{z}_i(t))(z_i(t)+\dot{z}_i(t))^*\big]$, we can get that

$$\|z(t)\|^2\le\frac{\mathcal{D}}{\mathcal{H}}\tag{3.6}$$

and

$$\|\dot{z}(t)\|^2\le\frac{\mathcal{D}}{\mathcal{H}}.\tag{3.7}$$

From (3.6), it can obtain that

$$\|z(t)\|\le\sqrt{\frac{\mathcal{D}}{\mathcal{H}}}.$$

From (3.7), likewise,

$$\|\dot{z}(t)\|\le\sqrt{\frac{\mathcal{D}}{\mathcal{H}}},$$

where

$$\begin{aligned}
&\mathcal{D} = \max_{i\in\mathbb{N}}\left\{6n\mathcal{M}_i + 2(\hat{a}_i - a_i)\rho_i^2 + 2(\hat{b}_i - b_i)\delta_i^2\right\},\\
&\mathcal{H} = \min\left\{\min_{i\in\mathbb{N}}\mathcal{A}_i, \min_{i\in\mathbb{N}}\mathcal{C}_i\right\},\\
&\mathcal{A}_i = a_i + \alpha_i + b_i + \beta_i - 1,\\
&\mathcal{C}_i = 2(a_i + \alpha_i) + (a_i - \hat{a}_i) + (b_i - \hat{b}_i)\\
&\qquad - \sum_{j=1}^{n}\left(\tilde{c}_{ij} + \tilde{d}_{ij} + \tilde{e}_{ij} + \bar{c}_{ij} + \bar{d}_{ij} + \bar{e}_{ij}\right) - 2n\mathcal{L}_i - 2.
\end{aligned}$$

Therefore, according to the definition 2.1, the system (2.3) is quasi-stable. That is, the driving system (2.1) and the response system (2.2) are quasi-synchronized. This completes the proof. □

Furthermore, when $\hat{a}_i = a_i$, $\hat{b}_i = b_i$, $\hat{c}_{ij} = c_{ij}$, $\hat{d}_{ij} = d_{ij}$, $\hat{e}_{ij} = e_{ij}$, the systems (2.1) and (2.2) are simply synchronous. From this, we can draw the following inference.

**Corollary 3.1.** *Assumption* $(H_1)$ *hold, further assumption is made as follows*

$(H_4)$ $H_i = \min\{A_i, C_i\} > 0$, $B_i > H_i > 0$,

*where*

$$\begin{aligned}
&A_i = a_i + \alpha_i + b_i + \beta_i - 1,\\
&B_i = 2(b_i + \beta_i) - \sum_{j=1}^{n}\left(\tilde{c}_{ij} + \tilde{d}_{ij} + \tilde{e}_{ij}\right) - 4n\mathcal{L}_i,\\
&C_i = 2(a_i + \alpha_i) - \sum_{j=1}^{n}\left(\tilde{c}_{ij} + \tilde{d}_{ij} + \tilde{e}_{ij}\right) - 2n\mathcal{L}_i - 2,
\end{aligned}$$

*similar to the proof of Theorem 3.1, then the driving system* (2.1) *and the response system* (2.2) *are simply synchronous under the controller* (3.1).

**Remark 3.1.** So far, there are few studies on quasi-synchronization of complex-valued inertial neural networks. Therefore, our theoretical results extend and complement the research on complex-valued inertial neural networks. In addition, considering similar systems (2.1) and (2.2), but without inertial terms, we can use the same method to analyze them. The proof process is omitted.

**Remark 3.2.** This paper directly scales the proportional delay by using inequality techniques, simplifying the complex derivation for handling delays. If $\gamma_j = 1$, then systems (2.1) and (2.2) are drive-response systems without delays. The obtained criteria are also applicable to them.

**Remark 3.3.** There is inevitably a mismatch in circuit parameters between the driving and response systems. Numerous studies have examined complex neural networks with parameter mismatches [31, 33, 36], including mismatches in state connection weights, activation functions, and time delays, among others. This paper aims to study the mismatch in system connection weights.

# 4. Numerical examples and applications

## 4.1. Illustrative example

In this subsection, the effectiveness of the previous theoretical results in this paper will be proved by one example.

**Example 4.1.** Consider the following complex-valued neutral-type inertial neural networks with proportional delays as the driving system, as follows:

$$\ddot{x}_i(t) = -\hat{a}_i\dot{x}_i(t) - \hat{b}_i x_i(t) + \sum_{j=1}^{2}\hat{c}_{ij}\mathcal{G}_j(x_j(t)) + \sum_{j=1}^{2}\hat{d}_{ij}\mathcal{G}_j(x_j(\gamma_j t)) + \sum_{j=1}^{2}\hat{e}_{ij}\mathcal{G}_j(\dot{x}_j(\gamma_j t)) + \mathcal{I}_i. \tag{4.1}$$

The corresponding response system is as follows:

$$\ddot{y}_i(t) = -a_i\dot{y}_i(t) - b_i y_i(t) + \sum_{j=1}^{2}c_{ij}\mathcal{G}_j(y_j(t)) + \sum_{j=1}^{2}d_{ij}\mathcal{G}_j(y_j(\gamma_j t)) + \sum_{j=1}^{2}e_{ij}\mathcal{G}_j(\dot{y}_j(\gamma_j t)) + \mathcal{I}_i + u_i(t). \tag{4.2}$$

Let $i = 1, 2$, given the system (4.1) whose parameters can be shown by

$$\begin{aligned}
&\hat{a}_1 = 0.5,\ \hat{a}_2 = 1.2,\ \hat{b}_1 = 0.2,\ \hat{b}_2 = 1.5,\\
&\hat{c}_{11} = 0.25 + 0.3\mathbf{i},\ \hat{c}_{12} = 0.2 - 0.1\mathbf{i},\ \hat{c}_{21} = 0.2 + 0.1\mathbf{i},\ \hat{c}_{22} = 0.4 - 0.2\mathbf{i},\\
&\hat{d}_{11} = 0.35 + 0.2\mathbf{i},\ \hat{d}_{12} = 0.2 + 0.3\mathbf{i},\ \hat{d}_{21} = 0.3 + 0.1\mathbf{i},\ \hat{d}_{22} = 0.2 - 0.1\mathbf{i},\\
&\hat{e}_{11} = 0.25 + 0.35\mathbf{i},\ \hat{e}_{12} = 0.3 - 0.1\mathbf{i},\ \hat{e}_{21} = 0.1 + 0.1\mathbf{i},\ \hat{e}_{22} = 0.2 - 0.1\mathbf{i},
\end{aligned}$$

and the coefficients in system (4.2) are

$$\begin{aligned}
&a_1 = 0.4,\ a_2 = 1,\ b_1 = 0.1,\ b_2 = 1.2,\\
&c_{11} = 0.2 + 0.3\mathbf{i},\ c_{12} = 0.1 + 0.1\mathbf{i},\ c_{21} = 0.2 + 0.2\mathbf{i},\ c_{22} = 0.5 - 0.2\mathbf{i},\\
&d_{11} = 0.3 + 0.2\mathbf{i},\ d_{12} = 0.2 + 0.2\mathbf{i},\ d_{21} = 0.3 + 0.1\mathbf{i},\ d_{22} = 0.3 - 0.1\mathbf{i},\\
&e_{11} = 0.3 + 0.3\mathbf{i},\ e_{12} = 0.1 - 0.1\mathbf{i},\ e_{21} = 0.2 + 0.1\mathbf{i},\ e_{22} = 0.3 - 0.1\mathbf{i}.
\end{aligned}$$

Besides, the other parameters are expressed by $\delta_1 = \delta_2 = 1.5$, $\rho_1 = \rho_2 = 1.2$, choose $\alpha_1 = \alpha_2 = 1.5$ and $\beta_1 = \beta_2 = 2$ the delays are $\gamma_1 = \gamma_2 = 0.8$, the external input are $\mathcal{I}_1 = \mathcal{I}_2 = 0.5$, choose activation functions $\mathcal{G}_1(x) = \mathcal{G}_2(x) = 0.1\sin x^R + 0.2\mathbf{i}\sin x^I$, which comply with the assumptions with $\mathcal{M}_1 = \mathcal{M}_2 = 0.05$ and $\mathcal{L}_1 = \mathcal{L}_2 = 0.04$.

By the simple calculation,

$$\begin{aligned}
&\mathcal{A}_1 = a_1 + \alpha_1 + b_1 + \beta_1 - 1 = 3,\\
&\mathcal{A}_2 = a_2 + \alpha_2 + b_2 + \beta_2 - 1 = 4.7,\\
&\mathcal{B}_1 = 2(b_1 + \beta_1) + (a_1 - \hat{a}_1) + (b_1 - \hat{b}_1) - \sum_{j=1}^{2}(\tilde{c}_{1j} + \tilde{d}_{1j} + \tilde{e}_{1j}
\end{aligned}$$

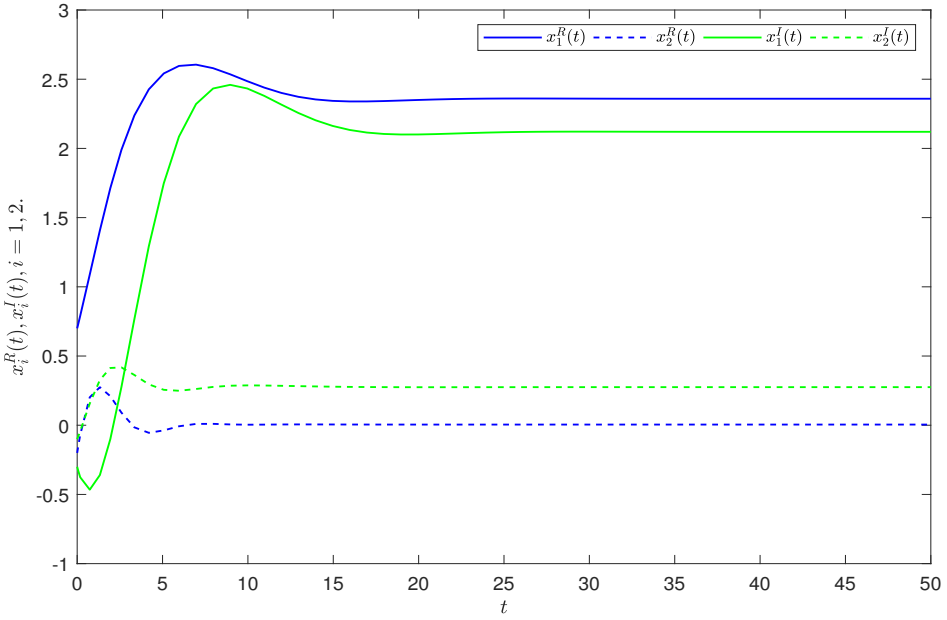


**Figure 1.** Trajectory diagram of the state variables $x_i^R(t)$, $x_i^I(t)$ of the drive system.

$$
\begin{aligned}
&+\bar{c}_{1j} + \bar{d}_{1j} + \bar{e}_{1j}) - 4n\mathcal{L}_1 = 3.61,\\
\mathcal{B}_2 &= 2(b_2 + \beta_2) + (a_2 - \hat{a}_2) + (b_2 - \hat{b}_2) - \sum_{j=1}^{2}(\tilde{c}_{2j} + \tilde{d}_{2j} + \tilde{e}_{2j}\\
&+\bar{c}_{2j} + \bar{d}_{2j} + \bar{e}_{2j}) - 4n\mathcal{L}_2 = 4.81,\\
\mathcal{C}_1 &= 2(a_1 + \alpha_1) + (a_1 - \hat{a}_1) + (b_1 - \hat{b}_1) - \sum_{j=1}^{2}(\tilde{c}_{1j} + \tilde{d}_{1j} + \tilde{e}_{1j}\\
&+\bar{c}_{1j} + \bar{d}_{1j} + \bar{e}_{1j}) - 2n\mathcal{L}_1 - 2 = 0.77,\\
\mathcal{C}_2 &= 2(a_2 + \alpha_2) + (a_2 - \hat{a}_2) + (b_2 - \hat{b}_2) - \sum_{j=1}^{2}(\tilde{c}_{2j} + \tilde{d}_{2j} + \tilde{e}_{2j}\\
&+\bar{c}_{2j} + \bar{d}_{2j} + \bar{e}_{2j}) - 2n\mathcal{L}_2 - 2 = 1.57,\\
\mathcal{D}_1 &= 6n\mathcal{M}_1 + 2(\hat{a}_1 - a_1)\rho_1^2 + 2(\hat{b}_1 - b_1)\delta_1^2 = 1.338,\\
\mathcal{D}_2 &= 6n\mathcal{M}_2 + 2(\hat{a}_2 - a_2)\rho_2^2 + 2(\hat{b}_2 - b_2)\delta_2^2 = 2.526.
\end{aligned}
$$

Thus, $\mathcal{D} = \max\{\mathcal{D}_1, \mathcal{D}_2\} = 2.526$, $\mathcal{A} = \min\{\mathcal{A}_1, \mathcal{A}_2\} = 3$, and $\mathcal{C} = \min\{\mathcal{C}_1, \mathcal{C}_2\} = 0.77$, $\mathcal{H} = \min\{\mathcal{A}, \mathcal{C}\} = 0.77$, therefore, $\sqrt{\frac{\mathcal{D}}{\mathcal{H}}} = \sqrt{\frac{2.526}{0.77}} \approx 3.281$. All the assumptions outlined in Theorem 3.1 are satisfied, thus establishing quasi-synchronization between system (4.1) and system (4.2). Figures 1-4 show the time trajectories of the real and imaginary parts of the state variables in the drive-response system, respectively. Figure 5 shows that, under the control of the controller, the system (4.1)-(4.2) can achieve synchronization.

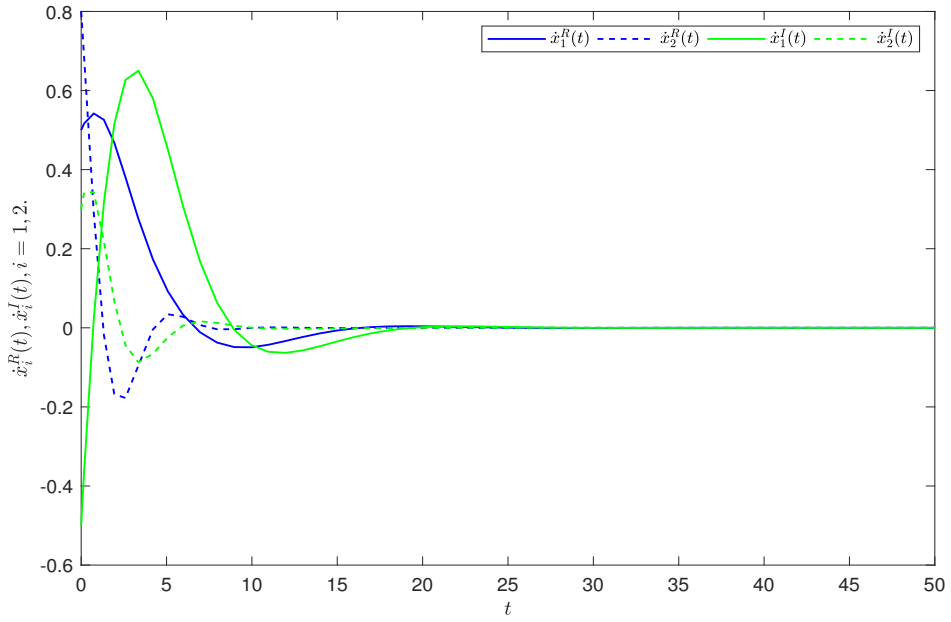


**Figure 2.** Trajectory diagram of the first-order state variables $\dot{x}_i^R(t)$, $\dot{x}_i^I(t)$ of the drive system.

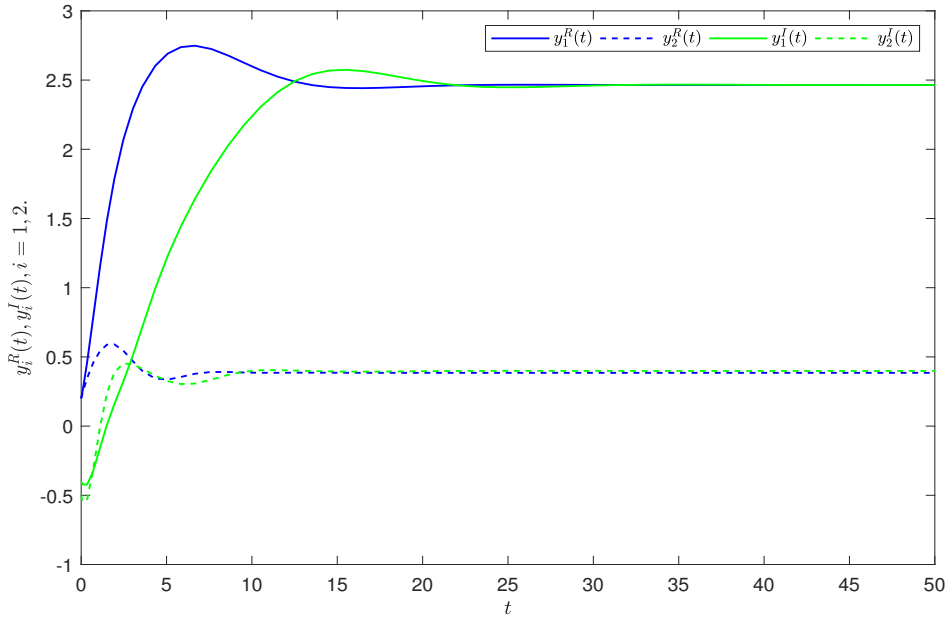


**Figure 3.** Trajectory diagram of the state variables $y_i^R(t)$, $y_i^I(t)$ of the response system.

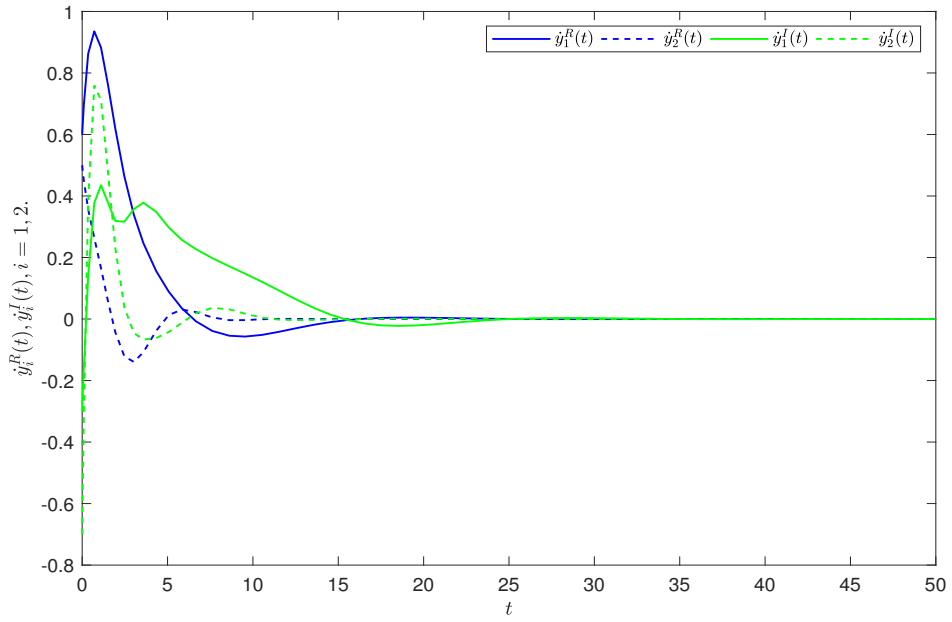


**Figure 4.** Trajectory diagram of the first-order state variables $\dot{y}_i^R(t)$, $\dot{y}_i^I(t)$ of the response system.

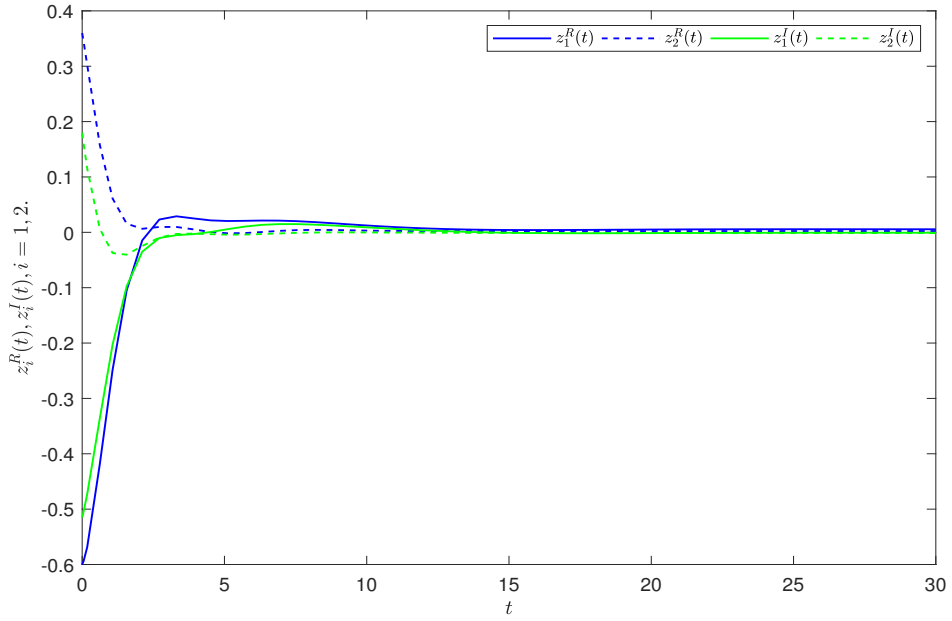


**Figure 5.** Trajectory diagram of the state variables $z_i^R(t)$, $z_i^I(t)$ of the error system.

**Remark 4.1.** Corollary 3.1 is the general case of parameter matches for the drive-response systems (2.1) and (2.2). Based on the proof of Theorem 3.1, by the same token, the conclusion of Corollary 3.1 also holds. Furthermore, using the exact numerical values as in Example 4.1, it can be concluded that the error system (2.3) is stable.

## 4.2. Application of image encryption

This part will encrypt and decrypt the image based on the parameters provided in the above numerical examples, utilizing the systems (4.1) and (4.2) from the article, along with the conclusions drawn in Theorem 3.1. As shown in Figure 6, the effects of the original image, the encrypted image, and the decrypted image are illustrated. The encryption effect is intuitively presented by analyzing the correlation and information entropy of adjacent pixels of the histogram.

### 4.2.1. Development of image encryption and decryption algorithm

The following outlines the algorithm design and specific steps for encrypting and decrypting color images.

**Step 1.** The color pixel matrix $\mathbf{A} = [a]_{m\times n}$ is acquired from original image, where $a_{pq} \in \{0, 1, \ldots, 255\}$, and $n$ represents the number of vertical pixels, while $m$ represents the number of horizontal pixels of the original image, respectively.

**Step 2.** Select the driver system to obtain the required encryption sequence

$$\{(x_1(t_u))\}_{u\in\{1,2,\ldots,m\}};\ \{(x_2(t_v))\}_{v\in\{1,2,\ldots,n\}},\ t_1 > T,$$

where $T$ refers to an instant when the synchronization error is sufficiently small.

**Step 3.** Get the driver system sequence matrix $\mathbf{B} = [b]_{m\times n}$, by:

$$b_{pq} = \mathrm{mod}\Big(\mathrm{floor}\left(\left\|x_1^R(t_u)\times x_1^I(t_v)\times x_2^R(t_u)\times x_2^I(t_v)\right\|\times 10^8\right), 256\Big),$$

where floor$(\cdot)$ function brings a value to the nearest integer approaching negative infinity, and mod$(\cdot,\cdot)$ signifies the modulo operator.

**Step 4.** Get the encrypted matrix $\mathbf{C} = [c_{pq}]_{m\times n}$, $c_{pq} = a_{pq} \oplus b_{pq}$, store the encrypted image's information, where $\oplus$ represents bitwise XOR.

Compared with encryption, decryption is the inverse process of encryption.

The following is a flowchart of the pseudo-code of the image encryption algorithm and the image encryption and decryption process.

### 4.2.2. Analysis of the histograms for both the original Image and the encrypted

As a key tool in image analysis, histograms can provide an intuitive presentation of the color frequency distribution in images. The histogram of the original image usually exhibits an uneven distribution, mainly when the frequency of specific color values occurs significantly higher. In Figure 7, we give a histogram of the original image and the encrypted image. By comparing the color frequency distributions of the two images, we can intuitively observe that the histogram of the original image exhibits irregular shapes and uneven features. In contrast, the histogram of the encrypted image presents a nearly uniform distribution. Our encryption effect is very significant.

**Algorithm 1** Chaotic Image Encryption

**Require:** Grayscale image matrix $\mathbf{A}$ (pixel values $\in [0, 255]$); Complex-valued neural network model

**Ensure:** Encrypted image matrix $\mathbf{C}$

1: Simulate complex-valued neural network to get chaotic sequences $\{x_1(t_u)\}$, $\{x_2(t_v)\}$, $T$ is synchronization error threshold)
2: Calculate chaotic intermediate matrix $\mathbf{B}$:
3: **for** each pixel $(p, q)$ **do**
4: $\quad b_{pq} = \mathrm{mod}\left(\mathrm{floor}\left(\left|x_1^R(t_u) \times x_1^I(t_u) \times x_2^R(t_v) \times x_2^I(t_v)\right| \times 10^8\right), 256\right)$
5: **end for**
6: Encrypt image matrix $\mathbf{C}$:
7: **for** each pixel $(p, q)$ **do**
8: $\quad c_{pq} = a_{pq} \oplus b_{pq}$
9: **end for**
10: **Output**: Encrypted image matrix $\mathbf{C}$

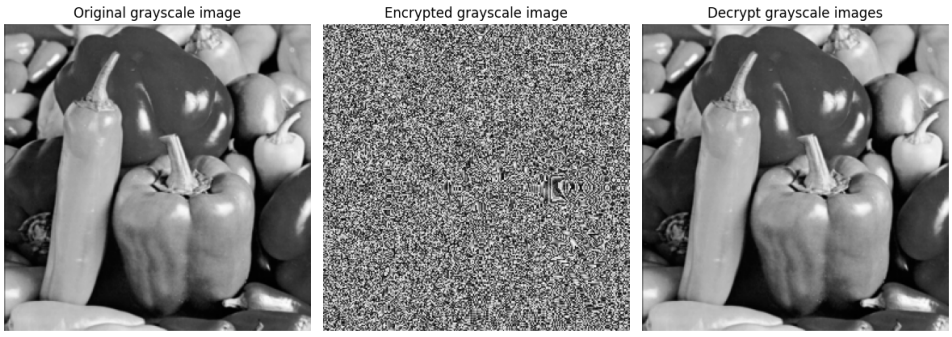


**Figure 6.** Original image, encrypted image, decrypted image.

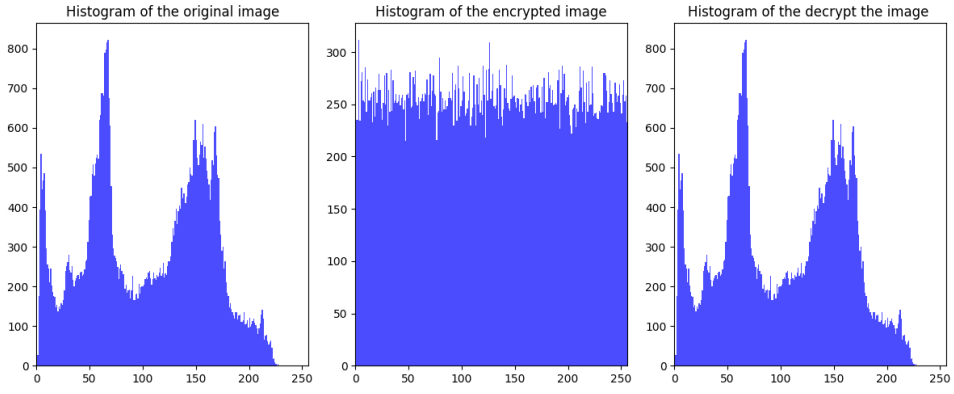


**Figure 7.** The histograms corresponding to the original image and the encrypted image.

### 4.2.3. Correlation-based analysis of the values of adjacent pixels

In addition to the above histogram analysis, the effect of image encryption can also be evaluated by the correlation of adjacent pixels. Among them, the correlation between neighboring pixel values encompasses horizontal, vertical, and diagonal adjacent pixel correlations. In an efficient encryption system, the closer the correlation of each adjacent pixel value in the ciphertext generated after the encryption process is zero, the better the encryption performance is. This situation poses a significant challenge for the attacker to extract sufficient information from the encrypted ciphertext, thereby ensuring that the image remains effectively protected. Next, we will use the following Pearson correlation coefficient formula [2]:

$$r_{xy} = \frac{\mathbf{cov}(x,y)}{\sqrt{d(x)}\sqrt{d(y)}},$$

where $r_{xy} \in [-1,1]$. When $r_{xy} = 1$, it indicates that there is a perfect positive correlation between $x$ and $y$; when $r_{xy} \in (0,1)$, it implies that there is a positive correlation between $x$ and $y$; when $r_{xy} \in (-1,0)$, it implies that there is a negatively correlation between $x$ and $y$; when $r_{xy} = -1$, it implies that there is a perfect negatively correlation between $x$ and $y$; when $r_{xy} = 0$, $x$ and $y$are not linearly correlated with each other. The covariance, variance, and mean are expressed as follows:

$$\begin{aligned}
&\mathbf{cov}(x,y) = \frac{1}{L}\sum_{p=1}^{L}\big(x_p - e(x)\big)\big(y_p - e(y)\big),\\
&d(x) = \frac{1}{L}\sum_{p=1}^{L}\big(x_p - e(x)\big)^2, \quad d(y) = \frac{1}{L}\sum_{p=1}^{L}\big(y_p - e(y)\big)^2,\\
&e(x) = \frac{1}{L}\sum_{p=1}^{L}x_p, \quad e(y) = \frac{1}{L}\sum_{p=1}^{L}y_p,
\end{aligned}$$

where $L$ represents the quantity of adjacent pixels within the image that are chosen to calculate the correlation. Figure 8 shows the correlation of the original and encrypted images in horizontal (**H**), vertical (**V**), and diagonal (**D**) directions. Table 1 presents the correlation coefficients between the original image and the encrypted image. It can be seen from Figure 8 and Table 1 that there is a strong correlation between adjacent pixel values in all directions of the original image. In contrast, the correlation of adjacent pixel values in all directions is significantly weakened, indicating that the encrypted image has a significant anti-interference effect.

**Table 1.** The correlation coefficient between the original image and the encrypted image.

| Image Type | **H** | **V** | **D** |
|---|---|---|---|
| Original image | 0.9704 | 0.9789 | 0.9483 |
| Encrypted image | −0.0313 | 0.0013 | −0.0086 |

### 4.2.4. The analysis regarding information entropy

Information entropy is an indicator of the degree of information disorder. The greater the disorder of the pixel values in the image, the closer the information entropy is to 8. In other

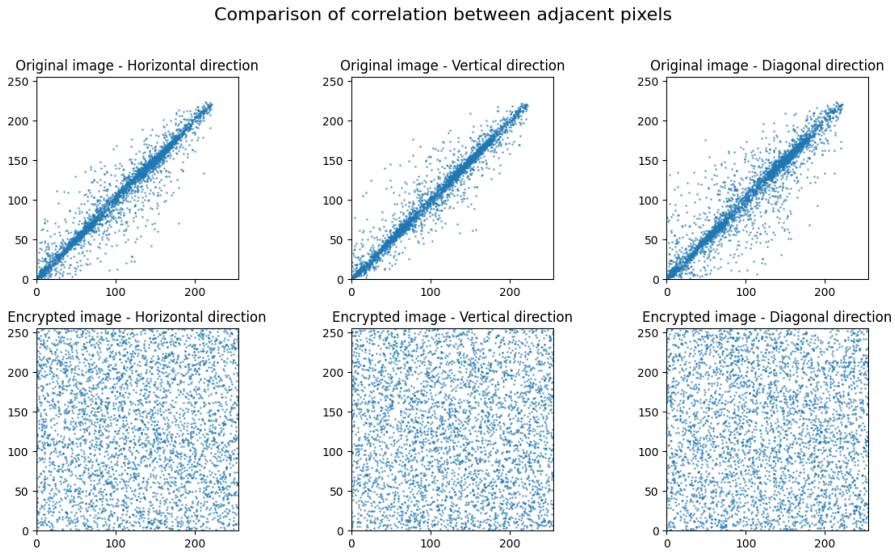


**Figure 8.** The correlation between the original image and the encrypted image in the horizontal, vertical, and diagonal directions.

words, the lower the probability of information leakage, the more effective the encryption is. Information entropy is calculated using the following formula [25]:

$$h(v) = \sum_{u=0}^{2^L-1} p(v_u) \log_2 \frac{1}{p(v_u)}$$

where, $p(v_u)$ represents the probability of $v_u$ occurrence.In theory, the closer the value of information entropy is to 8, the more complex and confusing the distribution of pixel values. As shown in Table 2, compared with the original image, the information entropy of the encrypted image is very close to 8, which indicates that the possibility of information leakage is extremely low.

**Table 2.** Information entropy values of the original and encrypted images.

| Image Type | Information Entropy |
|---|---|
| Original image | 7.5608 |
| Encrypted image | 7.9971 |

## 5. Conclusions and future works

This paper mainly studies the quasi-synchronization problem of complex-valued neutral inertial neural networks with proportional delay terms. Firstly, the original system was analyzed directly without introducing appropriate variable substitution, and the inertial neural network was equivalently transformed into a first-order differential system. The non-separation method is adopted to handle the complex-valued neural network. Compared with converting the complex-valued neural network equivalently into two real-valued neural networks, the non-separation

method is more concise and retains the system characteristics more completely. Secondly, under the design of a specific controller, by constructing an appropriate Lyapunov function and applying inequality techniques, the sufficient conditions for achieving quasi-synchronization of the error system are analyzed. Finally, the effectiveness of the conclusion is fully verified through numerical examples, and the research results are applied to the image encryption and decryption process. What is more noteworthy is that, compared with existing studies, the criterion for the quasi-synchronization of this system that we derived has a wider application range. Moreover, it provides a theoretical reference for the synchronization control of systems with parameter mismatches and expands the theoretical boundaries of complex-valued inertial neural networks. In the subsequent research, we plan to incorporate the fractional-order concept into the inertial neural network and extend the complex number domain to the quaternion domain. Then, we will study its dynamic behavior to address the practical problems it raises.

**CRediT authorship contribution statement. Yanxia Hu:** Writing-original draft, Formal analysis. **Xiaofang Meng:** Formal analysis, Writing–review & editing, Funding acquisition, Software. **Zhouhong Li:** Formal analysis, Writing–review & editing, Funding acquisition. **Jinde Cao:** Supervision, Validation, Writing–review & editing.

**Acknowledgments.** The authors are grateful to the editor and anonymous referees for their useful comments and suggestions.

**Declaration of competing interest.** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Data availability.** The authors declare that the manuscript has no associated data.

## References

[1] K. Babcock and R. Westervelt, *Stability and dynamics of simple electronic neural networks with added inertia*, Physica D: Nonlinear Phenomena, 1986, 23(1–3), 464–469.

[2] R. Boriga, A. C. Dăscălescu and I. Priescu, *A new hyperchaotic map and its application in an image encryption scheme*, Signal Processing: Image Communication, 2014, 29(8), 887–901.

[3] R. W. Brause, *Medical analysis and diagnosis by neural networks*, Medical Data Analysis, 2001, 1–13.

[4] F. Chowdhury, P. Wahi, R. Raina and S. Kaminedi, *A survey of neural networks applications in automatic control*, Proceedings of the 33rd Southeastern Symposium on System Theory (Cat. No. 01EX460), Athens, OH, USA, 2001, 349–353.

[5] J. Chu, M. Gao, X. Liu, et al., *Channel estimation based on complex-valued neural networks in IM/DD FBMC/OQAM transmission system*, Journal of Lightwave Technology, 2022, 40(4), 1055–1063.

[6] T. L. Clarke, *Generalization of neural networks to the complex plane*, 1990 IJCNN International Joint Conference on Neural Networks, San Diego, CA, USA, 1990, 435–440.

[7] R. Guo, S. Xu and C. K. Ahn, *Dissipative sliding-mode synchronization control of uncertain complex-valued inertial neural networks: Non-reduced-order strategy*, IEEE Transactions on Circuits and Systems I: Regular Papers, 2022, 70(2), 860–871.

[8] R. Guo, S. Xu and J. Guo, *Sliding-mode synchronization control of complex-valued inertial neural networks with leakage delay and time-varying delays*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2022, 53(2), 1095–1103.

[9] J. Han, G. Chen, L. Wang, et al., *Direct approach on fixed-time stabilization and projective synchronization of inertial neural networks with mixed delays*, Neurocomputing, 2023, 535, 97–106.

[10] W. He, F. Du and S. Zhang, *Non-separation method-based finite-time synchronization of discrete-time fractional-order complex-valued neural networks with time-varying delays*, Neurocomputing, 2025, 130732.

[11] S. Jia and L. Zhou, *Fixed-time stabilization of fuzzy neutral-type inertial neural networks with proportional delays*, ISA Transactions, 2024, 144, 167–175.

[12] G. Jiang, L. Wang, X. Zong, et al., *Practically predefined-time stabilization of stochastic fuzzy memristive neural networks under deception attacks*, IEEE Transactions on Cybernetics, 2025. DOI: 10.1109/TCYB.2025.3621274.

[13] J. Jin, J. Fang, C. Chen, et al., *A complex-valued time varying zeroing neural network model for synchronization of complex chaotic systems*, Nonlinear Dynamics, 2025, 113(6), 5471–5491.

[14] A. Kumar, S. Singh, S. Das and Y. Cao, *Projective quasi-synchronization of complex-valued recurrent neural networks with proportional delay and mismatched parameters via matrix measure approach*, Engineering Applications of Artificial Intelligence, 2023, 126, 106800.

[15] S. Lakshmanan, M. Prakash, C. P. Lim, et al., *Synchronization of an inertial neural network with time-varying delays and its application to secure communication*, IEEE Transactions on Neural Networks and Learning Systems, 2016, 29(1), 195–207.

[16] C. Lee, H. Hasegawa and S. Gao, *Complex-valued neural networks: A comprehensive survey*, IEEE/CAA Journal of Automatica Sinica, 2022, 9(8), 1406–1426.

[17] H. Li, L. Wang and Q. Lai, *Synchronization of a memristor chaotic system and image encryption*, International Journal of Bifurcation and Chaos, 2021, 31(16), 2150251.

[18] H. Li, L. Wang, X. Wan and C.-K. Zhang, *Error transmission of chaos-based image encryption: Application to smart grid*, IEEE Transactions on Industrial Informatics, 2025. DOI: 10.1109/TII.2025.3609139.

[19] W. Li, X. Gao and R. Li, *Stability and synchronization control of inertial neural networks with mixed delays*, Applied Mathematics and Computation, 2020, 367, 124779.

[20] X. Li, X. Li and C. Hu, *Some new results on stability and synchronization for delayed inertial neural networks based on non-reduced order method*, Neural Networks, 2017, 96, 91–100.

[21] H. Lin, Y. Shi, J. Guo and X. He, *Non-decomposition method for event-triggered finite-time synchronization control of complex-valued memristive neural networks*, Cognitive Neurodynamics, 2025, 19(1), 115.

[22] Y. Liu, Z. Zhang, X. Wang, et al., *Finite-time stabilization for fuzzy complex-valued neural networks with mixed delays via comparison approach*, IEEE Transactions on Fuzzy Systems, 2023, 32(2), 621–633.

[23] C. Long, G. Zhang and J. Hu, *Fixed-time synchronization for delayed inertial complex-valued neural networks*, Applied Mathematics and Computation, 2021, 405, 126272.

[24] A. G. Mullissa, C. Persello and J. Reiche, *Despeckling polarimetric SAR data using a multistream complex-valued fully convolutional network*, IEEE Geoscience and Remote Sensing Letters, 2021, 19, 1–5.

[25] A. Y. Niyat, M. H. Moattar and M. N. Torshiz, *Color image encryption based on hybrid hyper-chaotic system and cellular automata*, Optics and Lasers in Engineering, 2017, 90, 225–237.

[26] M. Egmont-Petersen, D. de Ridder and H. Handels, *Image processing with neural networks-a review*, Pattern Recognition, 2002, 35(10), 2279–2301.

[27] J. Schmidhuber, *Deep learning in neural networks: An overview*, Neural Networks, 2015, 61, 85–117.

[28] L. Shanmugam, P. Mani, R. Rajan and Y. H. Joo, *Adaptive synchronization of reaction-diffusion neural networks and its application to secure communication*, IEEE Transactions on Cybernetics, 2018, 50(3), 911–922.

[29] Y. Sheng, H. Gong and Z. Zeng, *Global synchronization of complex-valued neural networks with unbounded time-varying delays*, Neural Networks, 2023, 162, 309–317.

[30] Q. Song, L. Yang, Y. Liu and F. E. Alsaadi, *Stability of quaternion-valued neutral-type neural networks with leakage delay and proportional delays*, Neurocomputing, 2023, 521, 191–198.

[31] Z. Tang, J. H. Park and J. Feng, *Impulsive effects on quasi-synchronization of neural networks with parameter mismatches and time-varying delay*, IEEE Transactions on Neural Networks and Learning Systems, 2017, 29(4), 908–919.

[32] B. B. Traore, B. Kamsu-Foguem and F. Tangara, *Deep convolution neural network for image recognition*, Ecological Informatics, 2018, 48, 257–268.

[33] P. Wan and Z. Zeng, *Quasi-synchronization of timescale-type delayed neural networks with parameter mismatches via impulsive control*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2023, 53(7), 4254–4266.

[34] L. Wang, Y. Hu, C. Hu, et al., *Finite-time synchronization of delayed fuzzy inertial neural networks via intermittent control*, Neurocomputing, 2024, 574, 127288.

[35] Z. Wang, D. W. Ho and X. Liu, *State estimation for delayed neural networks*, IEEE Transactions on Neural Networks, 2005, 16(1), 279–284.

[36] K. Wu, M. Tang and H. Ren, *Mittag-Leffler projective synchronization of caputo fractional-order reaction-diffusion memristive neural networks with multi-type time delays*, Communications in Nonlinear Science and Numerical Simulation, 2025, 149, 108934.

[37] Z. Wu, G. Chen and X. Fu, *Synchronization of a network coupled with complex-variable chaotic systems*, Chaos: An Interdisciplinary Journal of Nonlinear Science, 2012, 22(2).

[38] S. Yan, Z. Gu, J. H. Park and X. Xie, *Synchronization of delayed fuzzy neural networks with probabilistic communication delay and its application to image encryption*, IEEE Transactions on Fuzzy Systems, 2022, 31(3), 930–940.

[39] J. Yang, H.-L. Li, J. Yang, et al., *Quasi-synchronization and complete synchronization of fractional-order fuzzy BAM neural networks via nonlinear control*, Neural Processing Letters, 2022, 54(4), 3303–3319.

[40] H. Zhang and X. Wang, *Complex projective synchronization of complex-valued neural network with structure identification*, Journal of the Franklin Institute, 2017, 354(12), 5011–5025.

[41] Z. Zhang, M. Chen and A. Li, *Further study on finite-time synchronization for delayed inertial neural networks via inequality skills*, Neurocomputing, 2020, 373, 15–23.

[42] Z. Zhang, S. Wang, X. Wang, et al., *Event-triggered synchronization for delayed quaternion-valued inertial fuzzy neural networks via nonreduced order approach*, IEEE Transactions on Fuzzy Systems, 2023, 31(9), 3000–3014.

[43] L. Zhou, Q. Zhu and T. Huang, *Global polynomial synchronization of proportional delayed inertial neural networks*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2023, 53(7), 4487–4497.

[44] L. Zhu, Y. Jiang, Y. Li, et al., *Conductivity prediction and image reconstruction of complex-valued multi-frequency electrical capacitance tomography based on deep neural network*, IEEE Transactions on Instrumentation and Measurement, 2021, 71, 1–10.

Received September 2025; Accepted December 2025; Available online December 2025.