Finite-time Convergent Complex-Valued Neural Networks for the Time-varying Complex Linear Matrix Equations

Xuezhong Wang, Lu Liang and Maolin Che,

***Abstract*—In this paper, we propose two complex-valued neural networks for solving a time-varying complex linear matrix equation by constructing two new types of nonlinear activation functions. Theoretically, we prove that the complex-valued neural networks are globally stable in the sense of Lyapunov stability theory. The solution of the complex-valued neural networks converges to the theoretical solution of the time-varying complex linear matrix equation in finite time. Compared with existing real-valued neural networks for solving time-varying complex linear matrix equations, the complex-valued neural nerworks can avoid redundant computation in a double real-valued space and thus has a low model complexity and storage capacity. Numerical simulations are presented to show the effectiveness of the complex-valued neural networks.**

***Index Terms*—Time-varying complex linear matrix equation, finite time convergence, weighted sign-bi-power activation function, complex-valued neural network.**

I. Introduction

Let us consider the smoothly time-varying matrices $A(t) \in \mathbb{C}^{n\times n}$ and $B(t) \in \mathbb{C}^{n\times p}$, the goal of this paper is to find a time-varying matrix $X(t) \in \mathbb{C}^{n\times p}$ such that the following time-varying complex linear matrix equation holds true:

$$A(t)X(t) = B(t), \quad \text{with} \quad t \in [0, +\infty). \tag{1}$$

Without loss of generality, $A(t)$ and $B(t)$ are assumed to be known, and their time derivatives $\dot{A}(t)$ and $\dot{B}(t)$ are assumed to be known or could be estimated.

Throughout this paper, we use $\|A\|_F$, $A^\top$, A^H, $\Re(A)$ and $\Im(A)$ to denote the Frobenius norm, the transpose, the complex conjugated transpose, the real part and the imaginary part of a given matrix $A \in \mathbb{C}^{m\times n}$, respectively. This notation is consistently used for lower-order parts of a given structure. For example, the entry with row index i and column index j in a matrix A, i.e., A_{ij}, is symbolized by a_{ij} (also $(\mathbf{x})_i = x_i$). Hence, we use $|A| = (|a_{kj}|)$, $\Theta(A) = (\Theta(a_{kj}))$ and $\exp(A) = (\exp(a_{kj}))$ denote the element-wise modulus, the element-wise argument and the element-wise exponential of the matrix $A \in \mathbb{C}^{m\times n}$, respectively. For two given matrices $A, B \in \mathbb{C}^{m\times n}$, $A \circ B$ denotes the Hadamard product of matrices A and B, i.e., $(A \circ B)_{ij} = a_{ij}b_{ij}$.

Matrix equations are arisen in control theory, signal processing, model reduction, image restoration, ordinary and partial differential equations and several applications in science and engineering. There are various approaches either direct methods or iterative methods to evaluate the solution of these equations [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. Numerous of numerical algorithms were presented for finding the approximate solution of linear matrix equations by using different techniques such as Taylor's series, homotopy, quadrature formulas, interpolation and decomposition [2], [3], [8], [9], [10]. Bai et al. [11] proposed a modification of the Hermitian and skew-Hermitian splitting iteration method for solving a broad class of complex symmetric linear systems; Axelsson et al. [12] introduced a real valued iterative methods for solving complex linear systems; Wang et al. [13] presented a preconditioned modification of the Hermitian and skew-Hermitian splitting iteration method for solving complex symmetric linear systems; Each iteration of their method requires the solution of two linear systems with real symmetric coefficient matrices. Ding et al. [2], [14], [15] extend the classical iterative methods (such as the Jacobi and Gauss-Seidel iterative methods) for the system of linear equations $Ax = b$ to solve the system of linear matrix equations (such as the generalized Sylvester matrix equation). It has been proven that the Jacobi and Gauss-Seidel iterative algorithms complete the calculation within finite steps of iteration and has a time complexity $O(n^3)$ [16]. Evidently, such serial processing algorithms performed on digital computers may not be efficient enough in large-scale online applications. Especially, when applied to online solution of time-varying linear matrix equations, these related iterative algorithms should be fulfilled within every sampling period and the algorithms fail when the sampling rate is too high to allow the algorithms to complete the calculation in a single sampling period, not to mention more challenging situations [17].

Recently, many authors have shown great interest for solving linear matrix equations on the basis of gradient-based neural networks (GNNs) [2], [9], [18] or Zhang neural networks (ZNNs) [17], [19], [20]. The GNN approach uses the Frobenius norm of the error matrix as the performance criterion and defines a neural network evolving along the negative gradient-descent direction. In the time-varying case, the Frobenius norm of the error matrix cannot decrease to zero even after infinite time due to the lack of velocity compensation of time-varying coefficients [20]. Zhang neural networks (ZNNs) are developed for solving online time-varying problems. Their dynamic is designed based on an indefinite error-monitoring function instead of a usually norm-based energy function [18], [21], [22], [23], [24], [25], [26]. Compared with GNNs, a prominent advantage of the

X. Wang is with the School of Mathematics and Statistics, Hexi University, Zhangye, 734000, P. R. of China. E-mail:xuezhongwang77@126.com.

L. Lu is with the School of Energy Science and Engineering, Central South University, Changsha, 410083, P. R. of China. E-mail: Lianglu612@yeah.net.

M. Che is with the School of Economic Mathematics, Southwest University of Finance and Economics, Chengdu, 611130, P. R. of China. E-mail:chnmaolinche@gmail.com.

(Advance online publication: 7 November 2018)

ZNN solution lies in that the lagging error diminishes to zero exponentially as time t goes on [18], [21], [22], [23], [24], [25], [26]. It is well known that the design of ZNN is based on a matrix or vector-valued indefinite error function and an exponent-type formula, which makes every entry/element of the error function exponentially converge to zero. By defining different Zhang functions, a series of ZNN models can be proposed for solving the same time-varying problem [17], [20].

Xiao and Zhang [20] have proposed two real-valued finite time convergence ZNN models to compute the time-varying real linear matrix equation. A finite-time convergent ZNN models with the Li function for the online solution of the time-varying complex linear matrix equation was studied in [17]. Since any complex matrix can be treated as the combination of its real and imaginary parts, the time-varying complex linear matrix equation (1) can be rewritten as [17]:

$$\begin{aligned}&[\Re(A(t))+\iota\Im(A(t))]\,[\Re(X(t))+\iota\Im(X(t))]\\&=\Re(B(t))+\iota\Im(B(t)),\end{aligned}\tag{2}$$

where ι ($\iota^2=-1$) is the imaginary unit.

Considering that the real and the imaginary parts of the left-side and right-side of the equation (2) always holds equal, the following time-varying matrix equations can be further derived from equation (2) as

$$\left\{\begin{array}{rcl}\Re(A(t))\Re(X(t))-\Im(A(t))\Im(X(t)) &=& \Re(B(t)),\\ \Re(A(t))\Im(X(t))+\Im(A(t))\Re(X(t)) &=& \Im(B(t)),\end{array}\right.$$

which can be equivalently expressed in a compact matrix form as

$$\begin{aligned}&\begin{bmatrix}\Re(A(t)) & -\Im(A(t))\\ \Im(A(t)) & \Re(A(t))\end{bmatrix}\begin{pmatrix}\Re(X(t))\\ \Im(X(t))\end{pmatrix}\\&=\begin{pmatrix}\Re(B(t))\\ \Im(B(t))\end{pmatrix}\in\mathbb{R}^{2n\times 2p}.\end{aligned}\tag{3}$$

Hence, the n-dimensional time-varying complex linear matrix equation (1) has been transferred to the $2n$-dimensional time-varying real linear matrix equation (3). We note that we need to design a $2n$-dimensional real-valued neural network for solving the matrix equation (3), which adds much workload and is computationally inefficient.

Thus, the main motivation and the novelty of us, in this paper, is to propose a complex-valued neural network for solving the matrix equation (1), where this neural network can avoid redundant computation in a double real-valued space and reduce a low model complexity and storage capacity.

This paper is organized as follows. In Section II, we recall some preliminary results. Complex-valued neural network models with the weighted sign-bi-power activation functions for online solution of the time-varying complex linear matrix equations are presented in Section III. Convergence properties of the complex-valued neural network models will be discussed in Section IV. Gradient-based neural network models will be presented in section V. Illustrative numerical examples are presented in Section VI.

Before ending this section, the main contributions of this paper are summarized and listed as follows:

1) This paper focuses on solving time-varying complex linear matrix equations in complex domain rather than conventionally investigated static or time-varying linear matrix equations in real domain.
2) Two types of activation functions are constructed and two finite-time convergent complex-valued neural networks are proposed and investigated for online solution of time-varying complex linear matrix equations in complex domain.
3) The paper carries out an in-depth theoretical analysis for our proposed ZNN models. It is theoretically proved that our models can converge to the theoretical solution of time-varying complex linear matrix equations with in finite time. In addition, the upper bound of the convergence time are derived analytically via Lyapunov theory.

II. Preliminary

By Euler's formula, a complex number $\alpha\in\mathbb{C}$ can be represented as $\alpha=|\alpha|\exp(\iota\theta)$, where $\theta\in(-\pi,\pi]$ is the augment of the number α. Meanwhile, we can also rewrite a complex matrix $A\in\mathbb{C}^{m\times n}$ as $|A|\circ\exp(\iota\Theta(A))$.

The following two lemmas are needed to analyze the convergence and stability of the complex-valued neural networks.

Lemma II.1. [27] *The following identity holds for arbitrary time-varying complex matrix $Z(t)\in\mathbb{C}^{m\times p}$:*

$$\frac{\mathrm{d}Z^H(t)}{\mathrm{d}t}=\left(\frac{\mathrm{d}Z(t)}{\mathrm{d}t}\right)^H.$$

Lemma II.2. [27] *For any two time-varying complex matrices $Y(t),Z(t)\in\mathbb{C}^{m\times p}$, the next identity is satisfied:*

$$\frac{\mathrm{d}(Y(t)Z(t))}{\mathrm{d}t}=\frac{\mathrm{d}Y(t)}{\mathrm{d}t}Z(t)+Y(t)\frac{\mathrm{d}Z(t)}{\mathrm{d}t}.$$

For a given matrix $A\in\mathbb{R}^{m\times n}$, the function $\mathcal{F}(A)$ is defined to be element-wise applicable, odd and monotonically increasing i.e., $\mathcal{F}(A)=(f(a_{kj}))$, with an odd and monotonically increasing function $f(\cdot)$, where

$$f(a_{kj}) \;=\; \tfrac{1}{2}k_1\mathrm{Lip}^{\sigma}(a_{kj})+\tfrac{1}{2}k_2\mathrm{Lip}^{\frac{1}{\sigma}}(a_{kj})+\tfrac{1}{2}k_3a_{kj}$$

with $\sigma\in(0,1)$ and

$$\mathrm{Lip}^{\sigma}(a_{kj})=\left\{\begin{array}{ll}a_{kj}^{\sigma}, & a_{kj}>0,\\ 0, & a_{kj}=0,\\ -a_{kj}^{\sigma}, & a_{kj}<0.\end{array}\right.$$

Now, we construct two new activation functions to analyze the complex-valued neural networks for solving the equation (1). For a given complex matrix $A=\Re(A)+\iota\Im(A)\in\mathbb{C}^{m\times n}$, two types of the weighted sign-bi-power activation functions $\Psi_k(A)=(\psi_k(e_{ij}))$ $(k=1,2)$ are as follows:

(a) Type I activation function array is defined by

$$\Psi_1(\Re(A)+\iota\Im(A))=\mathcal{F}(\Re(A))+\iota\mathcal{F}(\Im(A)).\tag{4}$$

(b) Type II activation function array is defined by the expression

$$\Psi_2(\Re(A)+\iota\Im(A))=\mathcal{F}(\Gamma)\circ\exp(\iota\Theta),\tag{5}$$

where $\Gamma=|A|\in\mathbb{R}^{m\times n}$ (resp. $\Theta=\Theta(A)\in\mathbb{R}^{m\times n}$) denotes element-wise modulus (resp. element-wise arguments) of the complex matrix A.

III. NEURAL NETWORK MODELS BASED ON ZNN

Here, the nonlinear methods of ZNN design for finite-time convergent complex-valued ZNN models are presented. Then, by exploiting this method, two finite time convergent ZNN models are first proposed for time-varying complex linear matrix equations based on two basic ZFs. For presentation convenience, such two ZNN models are termed ZNN-I model and ZNN-II model.

As usual, the time derivative of the complex function $E(t)$ is denoted by $\dot{E}(t)$. The complex-valued neural network model is developed by employing three basic steps from [17], [20]. An application of these steps in our case is described in the following.

Step 1. (Choose a convenient Zhang Function). The first step assumes definition of a proper fundamental matrix-valued error-monitoring function (Zhang Function, or ZF shortly) is defined as follows

$$E(t) = A(t)X(t) - B(t). \tag{6}$$

In order to ensure the existence of the unique complex and time-varying theoretical solution at any time instant $t \in [0, +\infty)$. A solution of (6) can be found in the possibility of adding a bias term λI with a positive scalar $\lambda \in \mathbb{R}$, where $I \in \mathbb{R}^{n\times n}$ is identical matrix. This replacement of a singular matrix by a well-conditioned matrix is known as Tikhnov regularization method [28]. Therefore, it is reasonable to define the following complex function as the fundamental error-monitoring function (called ZFL1):

$$E(t) = (A(t) + \lambda I)\, X(t) - B(t). \tag{7}$$

Step 2. (Define Zhang design formula). In the second step, with the aim to achieve global convergence of $E(t)$ to zero, it is necessary to use the general design pattern

$$\dot{E}(t) := \frac{\mathrm{d}E(t)}{\mathrm{d}t} = -\gamma\Psi_k(E(t)),\ k = 1, 2, \tag{8}$$

where the design parameter $\gamma > 0$ corresponds to the inductance parameter or reciprocal of a capacitance parameter, and $\Psi_k(\cdot)$ $(k = 1, 2)$ denotes an especially constructed activation-function matrix mapping of neural networks.

In this paper, we apply the weighted sign-bi-power activation function [29] to accelerate the ZNN to finite-time convergence to the theoretical solution of the time-varying complex linear matrix equation.

Step 3. (Generate a ZNN model). In the last step, the dynamic equation of a complex neural network model for computing the time-varying complex-valued linear matrix equation can be established by expanding (8). The complex matrix-valued error-monitoring function $E(t)$ defined in (7) possesses the following time derivative:

$$\dot{E}(t) = \dot{A}(t)X(t) + (A(t) + \lambda I)\dot{X}(t) - \dot{B}(t). \tag{9}$$

Combining (8) and (9), we can obtain the following implicit dynamic equation of ZNN model:

$$\begin{aligned} A(t)\dot{X}(t) \quad = \quad & \dot{B}(t) - \dot{A}(t)X(t) - \gamma\Psi_k[(A(t) + \lambda I)X(t) \\ & -B(t)],\ k = 1, 2. \end{aligned} \tag{10}$$

For presentation convenience, if $k = 1$, we call above dynamic equation as ZNN-I model. If $k = 2$, we call above dynamic equation as ZNN-II model.

IV. CONVERGENCE ANALYSIS

In this section, we prove that both ZNN-I and ZNN-II can be globally convergent to the time-varying theoretical solution of equation (1).

A. Convergence of the model ZNN-I

The convergence performances in finite time as the weighted sign-bi-power activation function of the ZNN-I model, defined on the basis of (4), is investigated in this subsection. The following result is valid for the ZNN-I model with the type I weighted sign-bi-power activation function.

Theorem IV.1. *Given smoothly time-varying complex matrices $A(t) \in \mathbb{C}^{n\times n}$ and $B(t) \in \mathbb{C}^{n\times p}$. If the Type I activation function is used, then the state matrix $X(t) \in \mathbb{C}^{n\times p}$ of the neural network* (10)*, starting from an arbitrary initial state $X(0) \in \mathbb{C}^{n\times p}$, converges to the theoretical solution $X^*(t) \in \mathbb{C}^{n\times p}$ of the equation* (1) *in finite time:*

$$t_f \le \begin{cases} \dfrac{2\ln\left[1+\frac{k_1}{k_3}|e^+(0)|^{1-\sigma}\right]}{\gamma k_3(1-\sigma)}, & \text{if } |e^+(0)| < 1; \\ \dfrac{2\sigma\ln\left[\frac{k_2+k_3}{k_3}|e^+(0)|^{\frac{1-\sigma}{\sigma}}+k_2\right]}{\gamma k_3(1-\sigma)} + \dfrac{2\sigma\ln\left[1+\frac{k_1}{k_3}\right]}{\gamma k_3(1-\sigma)}, & \text{if } |e^+(0)| \ge 1; \end{cases}$$

where $|e^+(0)|$ is the largest element in the matrix $|E(0)| = |A(0)X(0) - B(0)|$.

Proof. Let $\widetilde{X}(t) = X(t) - X^*(t)$ denote the difference between the time-varying solution $X(t)$ generated by the neural network (10) and the time-varying theoretical solution $X^*(t)$ of the equation (1). The time derivative of

$$(A(t) + \lambda I)\, X^*(t) - B(t) = 0,$$

can be expressed as

$$\dot{A}(t)X^*(t) + (A(t) + \lambda I)\, \dot{X}^*(t) - \dot{B}(t) = 0. \tag{11}$$

Substitution $X^*(t) = X(t) - \widetilde{X}(t)$ into (11) leads to

$$\dot{A}(t)\left(X(t) - \widetilde{X}(t)\right) + (A(t)+\lambda I)\left(\dot{X}(t) - \dot{\widetilde{X}}(t)\right) - \dot{B}(t) = 0.$$

The last equation is equivalent to

$$\begin{aligned} &\dot{A}(t)X(t) + (A(t) + \lambda I)\dot{X}(t) - \dot{B}(t) = \dot{A}(t)\widetilde{X}(t) \\ &+(A(t) + \lambda I)\dot{\widetilde{X}}(t). \end{aligned}$$

Since $\widetilde{X}(t) = X(t) - X^*(t)$, it can be verified that $\widetilde{X}(t)$ is the solution ensuring dynamics (10) with the initial state $\widetilde{X}(0) = X(0) - X^*(0)$.

By using (9) in conjunction with (10), it is possible to conclude

$$\dot{A}(t)\widetilde{X}(t) + (A(t) + \lambda I)\dot{\widetilde{X}}(t) = \dot{E}(t). \tag{12}$$

It is possible to verify that

$$\begin{aligned} E_X(t) &= (A(t) + \lambda I)\widetilde{X}(t) \\ &= (A(t) + \lambda I)(t)X(t) - (A(t) + \lambda I)X(t)^* \\ &= (A(t) + \lambda I)X(t) - B(t) \\ &= E(t). \end{aligned}$$

Then (12) can be rewritten as the following Zhang design formula:

$$\dot{E}_X(t) = \dot{E}(t) = -\gamma\Psi_1(E(t)).$$

According to the definition of $\Psi_1(\cdot)$, the following two equivalent formulae in the real numbers domain appear:

$$\Re(\dot{E}(t)) = -\gamma\mathcal{F}(\Re(E(t)))$$

and

$$\Im(\dot{E}(t)) = -\gamma\mathcal{F}(\Im(E(t))).$$

We construct the following Lyapunov function:

$$L(t) = \frac{\|E(t)\|_F^2}{2} = \frac{\mathrm{Tr}\left(E(t)^{\mathrm{H}}E(t)\right)}{2},$$

where $\mathrm{Tr}(P) = \sum_{i=1}^{n} p_{ii}$ for any matrix $P \in \mathbb{C}^{n\times n}$. Since $E(t) = \Re(E(t)) + \iota\Im(E(t))$, the time derivative of $L(t)$ satisfies the following identities:

$$\begin{aligned}
\frac{\mathrm{d}L(t)}{\mathrm{d}t} &= \frac{\mathrm{Tr}\left(\dot{E}(t)^{\mathrm{H}}E(t)+E(t)^{\mathrm{H}}\dot{E}(t)\right)}{2} \\
&= -\frac{1}{2}\gamma\mathrm{Tr}\left\{\left(\mathcal{F}\left(\Re\left(E(t)\right)\right)^{\mathrm{T}} - \iota\mathcal{F}\left(\Im\left(E(t)\right)\right)^{\mathrm{T}}\right)\right. \\
&\quad \cdot\left(\Re\left(E(t)\right) + \iota\Im\left(E(t)\right)\right) + \left(\Re\left(E(t)\right)^{\mathrm{T}} -\right. \\
&\quad \left.\left.\iota\Im\left(E(t)\right)^{\mathrm{T}}\right)\left(\mathcal{F}(\Re\left(E(t)\right))\right)^{\mathrm{T}} + \iota\mathcal{F}(\Im\left(E(t)\right))\right\} \\
&= -\gamma\mathrm{Tr}\left\{\Re\left(E(t)\right)^{\mathrm{T}}\mathcal{F}\left(\Re\left(E(t)\right)\right)\right. \\
&\quad \left.+\Im\left(E(t)\right)^{\mathrm{T}}\mathcal{F}\left(\Im\left(E(t)\right)\right)\right\}.
\end{aligned}$$

Since $\mathcal{F}(\cdot)$ is odd and monotonically increasing, we conclude

$$\Re\left(E(t)\right)^{\mathrm{T}}\mathcal{F}\left(\Re\left(E(t)\right)\right) + \Im\left(E(t)\right)^{\mathrm{T}}\mathcal{F}\left(\Im\left(E(t)\right)\right) \geq 0,$$

and then $\frac{\mathrm{d}L(t)}{\mathrm{d}t} \leq 0$. According to the Lyapunov stability theory,

$$E(t) = (A(t) + \lambda I)\,X(t) - B(t)$$

is globally convergent to zero matrix, regardless of the initial value. That is to say, as $t \to \infty$, we have

$$X(t) \to \left(A(t) + \lambda I\right)^{-1} B(t).$$

In view of $\lambda \to 0$, the state matrix $X(t)$ globally converges to the time-varying theoretical solution of (1) starting from arbitrary initial state $X(0)$. Next, it is necessary to prove the finite-time convergent performance of the ZNN-I model.

The initial value of the matrix valued error function $E(t)$ is

$$E(0) = (A(0) + \lambda I)X(0) - B(0).$$

We define

$$|e^{+}(0)| = \max\left\{|E(0)|\right\}.$$

Similar to [17], one can verify inequalities

$$|e^{+}(t)| \geq |e_{ij}(t)|, \quad -|e^{+}(t)| \leq |e_{ij}(t)| \leq |e^{+}(t)|,$$

for all possible values of indices $i \in \{1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, m\}$. This means that $|e_{ij}(t)|$ converges to zero for all possible $i \in \{1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, m\}$ when $|e^{+}(t)|$ reach zero. In other words, the convergence time of ZNN-I model (10) is bounded by t_f^{+} of the dynamics of $|e^{+}(t)|$, where t_f^{+} represent the convergence time of the dynamics of $|e^{+}(t)|$.

To estimate t_f^{+}, we can begin with the following formula:

$$|\dot{e}^{+}(t)| = -\gamma\psi_1\left(|e^{+}(t)|\right).$$

Another Lyapunov function candidate is defined as

$$l_{+}(t) = |e^{+}(t)|^2.$$

Since $l_{+}(t) \geq 0$, the time derivative of $l_{+}(t)$ is equal to

$$\begin{aligned}
\dot{l}_{+}(t) &= -2\gamma|e^{+}(t)|\psi_1\left(|e^{+}(t)|\right) \\
&= -\gamma\left(k_1|e^{+}(t)|^{\sigma+1} + k_2|e^{+}(t)|^{\frac{1}{\sigma}+1} + k_3|e^{+}(t)|^2\right) \\
&= -\gamma\left(k_1 v(t)^{\frac{\sigma+1}{2}} + k_2 v(t)^{\frac{\sigma+1}{2\sigma}} + k_3 v(t)\right).
\end{aligned}$$

For such a differential equation, if $v(0) = |e^{+}(0)|^2 \leq 1$, according to the Lemma 3 of [29], there exists t_f^{+} satisfying

$$t_f^{+} \leq \frac{2\ln\left[1 + \frac{k_1}{k_3}|e^{+}(0)|^{1-\sigma}\right]}{\gamma k_3(1-\sigma)},$$

such that all $|e_{ij}(t)| = 0$ when $t_f > t_f^{+}$.

If $v(0) = |e^{+}(0)|^2 \geq 1$, according to the Lemma 3 of [29], there exists t_f^{+} satisfying

$$t_f^{+} \leq \frac{2\sigma\ln\left[\frac{k_2+k_3}{k_3}|e^{+}(0)|^{\frac{1-\sigma}{\sigma}+k_2}\right]}{\gamma k_3(1-\sigma)} + \frac{2\sigma\ln\left[1 + \frac{k_1}{k_3}\right]}{\gamma k_3(1-\sigma)},$$

such that all $|e_{ij}(t)| = 0$ when $t_f > t_f^{+}$.

The above results mean that, if the Type I activation function is adopted, neural state $X(t)$ of the neural network (10) with Ψ_1 converges to the theoretical solution $X^{*}(t)$ of linear matrix equation (1) in finite time t_f. □

B. Convergence of the model ZNN-II

In the following, we investigate the convergence of the complex neural network model ZNN-II, defined by (10) for $k = 2$. The following result can be verified about the complex-valued neural network model ZNN-II based on a type II activation function.

Theorem IV.2. *Given smoothly time-varying complex matrices* $A(t) \in \mathbb{C}^{n\times n}$ *and* $B(t) \in \mathbb{C}^{n\times p}$*. If the Type II activation function is used, then the state matrix* $X(t) \in \mathbb{C}^{n\times p}$ *of the neural network* (10)*, starting from an arbitrary initial state* $X(0) \in \mathbb{C}^{n\times p}$*, converges to the theoretical solution* $X^{*}(t) \in \mathbb{C}^{n\times p}$ *of the equation* (1) *in finite time:*

$$t_f \leq \begin{cases} \dfrac{2\ln\left[1+\frac{k_1}{k_3}|e^{+}(0)|^{1-\sigma}\right]}{\gamma k_3(1-\sigma)}, & \text{if } |e^{+}(0)| < 1; \\ \dfrac{2\sigma\ln\left[\frac{k_2+k_3}{k_3}|e^{+}(0)|^{\frac{1-\sigma}{\sigma}+k_2}\right]}{\gamma k_3(1-\sigma)} + \dfrac{2\sigma\ln\left[1+\frac{k_1}{k_3}\right]}{\gamma k_3(1-\sigma)}, & \text{if } |e^{+}(0)| \geq 1; \end{cases}$$

where $|e^{+}(0)|$ *is the largest element in the matrix* $|E(0)| = |A(0)X(0) - B(0)|$.

Proof. Analogically as in the proof of Theorem IV.1, the error dynamics is given by

$$\dot{E}(t) = -\gamma\Psi_2(E(t)),$$

where $E(t) = (A(t) + \lambda I)\,X(t) - B(t)$. According to the definition of $\Psi_2(\cdot)$, immediately follows

$$\Psi_2(E(t)) = \mathcal{F}(|E(t)|) \circ \exp(\iota\Theta(E(t))).$$

We construct the following Lyapunov function:

$$L(t) = \frac{\|E(t)\|_F^2}{2} = \frac{\mathrm{Tr}\left(E(t)^{\mathrm{H}}E(t)\right)}{2},$$

which further implies

$$\begin{aligned}\frac{\mathrm{d}L(t)}{\mathrm{d}t} &= \frac{\mathrm{Tr}\left(E(t)^{\mathrm{H}}\dot{E}(t)+\dot{E}(t)^{\mathrm{H}}E(t)\right)}{2}\\ &= -\frac{1}{2}\gamma\mathrm{Tr}\left(E(t)^{\mathrm{H}}\mathcal{H}_2\left(E(t)\right)+E(t)\mathcal{H}_2\left(E(t)\right)^{\mathrm{H}}\right)\\ &= -\frac{1}{2}\gamma\mathrm{Tr}\left(E(t)^{\mathrm{H}}\mathcal{H}_2\left(E(t)\right)+\left(E(t)^{\mathrm{H}}\mathcal{H}_2\left(E(t)\right)\right)^{\mathrm{H}}\right)\\ &= -\gamma\mathrm{Tr}\left(\Re\left(E(t)^{\mathrm{H}}\mathcal{H}_2\left(E(t)\right)\right)\right)\\ &= -\gamma\mathrm{Tr}\left\{\Re\left[E(t)^{\mathrm{H}}\mathcal{F}\left(|E(t)|\right)\circ\exp\left(\iota\Theta(E(t))\right)\right]\right\}.\end{aligned}$$

Since $E(t) = |E(t)| \circ \exp(\iota\Theta(E(t)))$, one can verify

$$\begin{aligned}\frac{\mathrm{d}L(t)}{\mathrm{d}t} \;=\; &-\gamma\mathrm{Tr}\left\{\Re[\exp\left(-\iota\Theta\left(E(t)^{\mathrm{H}}\right)\circ\left|E(t)^{\mathrm{H}}\right|\right)\right.\\ &\left.\cdot\left(\mathcal{F}\left(|E(t)|\right)\circ\exp\left(\iota\Theta\left(E(t)\right)\right)\right)]\right\}.\end{aligned}$$

Since $\mathcal{F}(\cdot)$ is monotonically increasing, we conclude $\mathcal{F}(|E(t)|) \geq 0$ for $E(t) \neq 0$. As a result, $L(t)$ is negative definite. According to the Lyapunov stability theory, the matrix $E(t) = (A(t)+\lambda I)\,X(t) - B(t)$ globally converges to the zero matrix from arbitrary initial value. Similarly as in the proof of Theorem IV.1, we conclude that the state matrix $X(t)$ globally converges to the time-varying theoretical solution of (1) starting from arbitrary initial state $X(0)$.

Because ZNN-I model (10), for $k = 2$, is derived by using the intrinsically nonlinear method of ZNN design similar to ZNN-I model, we also have

$$\dot{E}_X(t) = \dot{E}(t) = -\gamma\Psi_2(E(t)).$$

Therefore, the proof of finite time convergence can be generalized from the proof of Theorem IV.1 and is thus omitted. □

V. Comparison verification

We compare in this section the ZNN model with the conventional GNN model for the same online time-varying complex linear matrix equations solving task. We have the following GNN models for time-varying complex linear matrix equations solving.

First, the following the scalar-valued nonnegative energy function is defined

$$E(t) = \frac{\|(A(t)+\lambda I)X(t) - B(t)\|_F^2}{2}.$$

Then, a complex-valued gradient algorithm is designed to evolve along a negative gradient descent direction of this energy function until the minimum point can be reached. Obviously, the negative gradient of this energy function can be derived as follows.

$$-\frac{\partial(E(t))}{\partial(X(t))} = -(A(t)+\lambda I)^{\mathrm{H}}\big((A(t)+\lambda I)X(t) - B(t)\big).$$

Third, by using the above negative gradient, we can construct the following complex-valued GNN model for online solution of the complex-valued linear equation system.

$$\frac{dX(t)}{dt} = -\gamma\big(A(t)+\lambda I\big)^{\mathrm{H}}\Psi_k\big((A(t)+\lambda I)X(t) - B(t)\big), \tag{13}$$

where $\Psi_k(\cdot)$ $(k = 1, 2)$ are two types of activation functions defined in section II. Design parameter $\gamma > 0$ is used to scale the convergence rate of the fully complex-valued GNN model, and complex state vector $X(t) \in \mathbb{C}^{n\times p}$, starting from any initial state $X(0) \in \mathbb{C}^{n\times p}$.

VI. Numerical examples

In this section, we show that we can use the neural network (10) with different activation functions to solve the time-varying complex linear equations (1) via several examples. Meanwhile, we also compare our neural networks (10) with the gradient-based neural network (13) via an example. the computations are implemented in Matlab Version 2013a on a laptop with Intel Core i5-4200M CPU (2.50GHz) and 7.89GB RAM. All ordinary differential equations are solved by the Matlab function “ode23” or “ode45”.

A. Numerical tests based on ZNN

Example 1 [17]. Consider the following time-varying matrices $A(t)$ and $B(t)$ with

$$A(t) = \sin(4t) + \iota\cos(4t),$$

$$B(t) = [\sin(3t) + 1 - \iota\cos(2t) \quad \cos(6t) + \iota(\sin(5t) + 2)].$$

Numerical values of $X(t)$ in different time points are computed numerically via the formula

$$X(t) = (A(t)+\lambda I)^{-1}B(t), \tag{14}$$

If we take initial vector $X(0) = (1, 1)^\top$ and choose the weighted sign-bi-power activation function as $f(\cdot)$ with $\sigma = \frac{1}{3}$, then state variables trajectories of real part and imaginary part of the model ZNN-II (10) with $\gamma = 20$ and $\lambda = 10^{-2}$ are shown in Figure 1 (a) and Figure 2 (a) and in Figure 1 (b) and Figure 2 (b), respectively, where the red curves denote the theoretical solution computed by (14) and blue curves denote the solution computed by the model ZNN-II.

Trajectories of residual errors $\|\,(A(t)+\lambda I)\,X(t) - B(t)\|_F$, generated by using the complex model ZNN-II with $\lambda = 10^{-3}$ and $\gamma = 20$ is shown in Figure 3 (a) and $\lambda = 10^{-5}$ and $\gamma = 2\times 10^6$ is shown in Figure 3 (b), respectively.

It is seen from Figure 1 and Figure 2 that state variables $X(t)$ of ZNN-II converges directly to the theoretical solution of (1) within a rather short time. In addition, Figure 3 (a) and (b) show the transient convergence behavior of $\|\,(A(t)+\lambda I)\,X(t) - B(t)\|_F$ synthesized by ZNN-II.

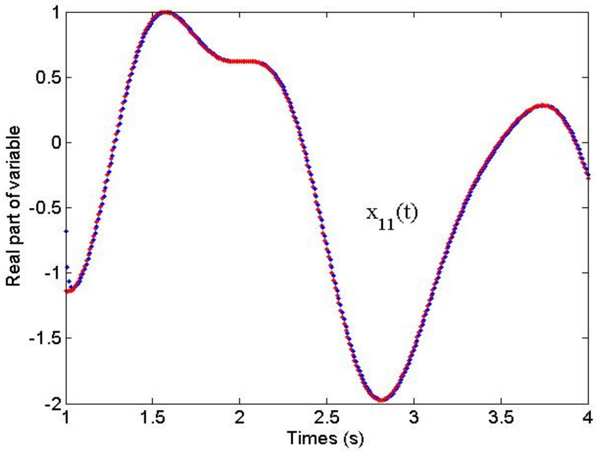
Example 2. Consider the time-varying matrix

$$A(t) = \begin{pmatrix} 2\sin(t) & 0 & 1 & \sin(t)\\ 0 & \cos(t) & 2\sin(t) & 1\\ \cos(t) & 1 & 0 & 2\cos(t)\\ 1 & 2\cos(t) & \sin(t) & 0 \end{pmatrix} + \iota\begin{pmatrix} \cos(t) & \sin(t) & 0 & \cos(t)\\ \sin(t) & \cos(t) & 2\cos(t) & \sin(t)\\ \sin(t) & t & 0 & 2\sin(t)\\ 1 & 2\sin(t) & \cos(t) & 0 \end{pmatrix}.$$

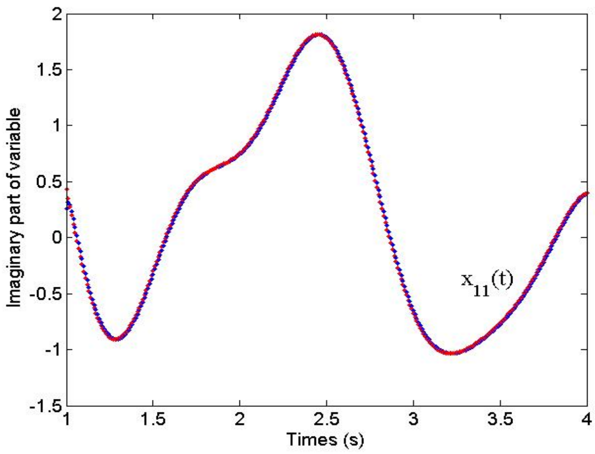
and

$$B(t) = (\sin(t)+\iota\cos(2t) \quad \sin(2t) \quad \iota\sin(3t) \quad \sin(4t)+\iota t)^\top.$$

If we take initial vector $X(0) = (0, 0, 0, 0)^\top$ and choose the weighted sign-bi-power activation function as $f(\cdot)$ with $\sigma = \frac{1}{3}$, then state variables trajectories of real part and imaginary part of the model ZNN-I (10) with $\gamma = 200$ and $\lambda = 10^{-3}$ are shown in Figure 4 (a) and (b), respectively.

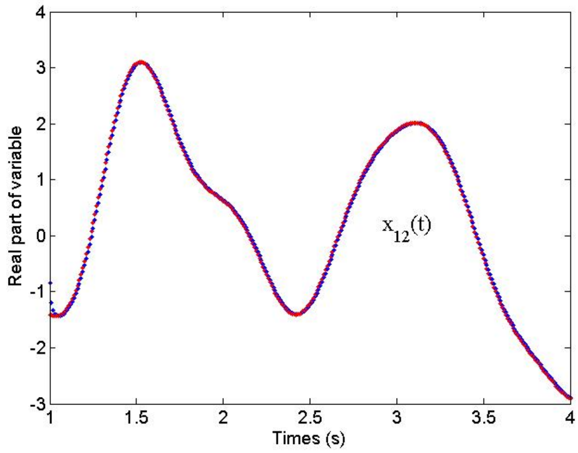


(a) The first column.

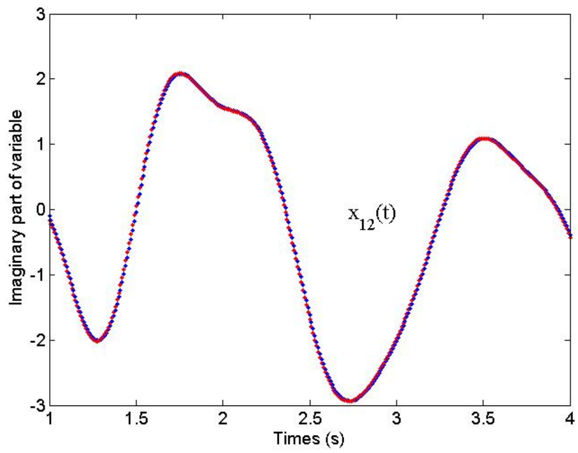


(b) The first column.

Fig. 1. Trajectories of the state variables of the model ZNN-II in Example 1.



(a) The second column.



(b) The second column.

Fig. 2. Trajectories of the state variables of the model ZNN-II in Example 1.

The red curves denote the theoretical solution computed by (14) and blue curves denote the solution computed by the model ZNN-I.

Trajectories of residual errors $\|\,(A(t)+\lambda I)\,X(t) - B(t)\|_F$, generated by using the complex model ZNN-I with $\lambda = 10^{-3}$ and $\gamma = 200$ is shown in Figure 5 (a) and $\lambda = 10^{-6}$ and $\gamma = 2\times 10^{7}$ is shown in Figure 5 (b), respectively.

It is seen from Figure 1 (a) and (b) that state variables $X(t)$ of ZNN-I converges directly to the theoretical solution of (1) within a rather short time. In addition, Figure 5 (a) and (b) show the transient convergence behavior of $\|\,(A(t)+\lambda I)\,X(t) - B(t)\|_F$ synthesized by ZNN-I.

B. Numerical tests based on GNN

To show our ZNN models's superiority, we compare in this subsection the ZNN model with the conventional GNN model for the same online time-varying complex linear matrix equations solving task.

Example 3. We consider the time-varying matrix in Example 2

$$A(t) = \begin{pmatrix} 2\sin(t) & 0 & 1 & \sin(t) \\ 0 & \cos(t) & 2\sin(t) & 1 \\ \cos(t) & 1 & 0 & 2\cos(t) \\ 1 & 2\cos(t) & \sin(t) & 0 \end{pmatrix}.$$

$$+\iota \begin{pmatrix} \cos(t) & \sin(t) & 0 & \cos(t) \\ \sin(t) & \cos(t) & 2\cos(t) & \sin(t) \\ \sin(t) & 1 & 0 & 2\sin(t) \\ 1 & 2\sin(t) & \cos(t) & 0 \end{pmatrix}.$$

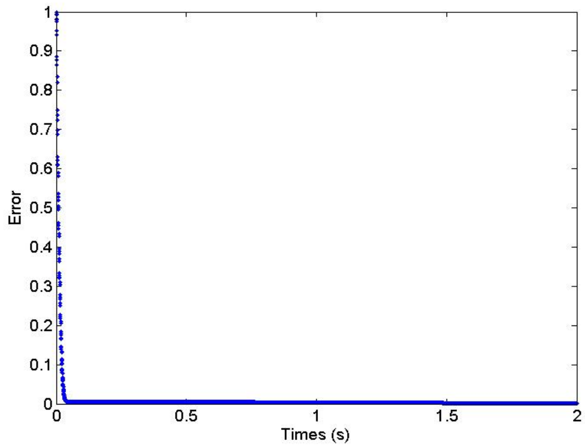
and

$$B(t) = (\sin(t)+\iota\cos(2t) \quad \sin(2t) \quad \iota\sin(3t) \quad \sin(4t)+\iota t)^{\top}.$$

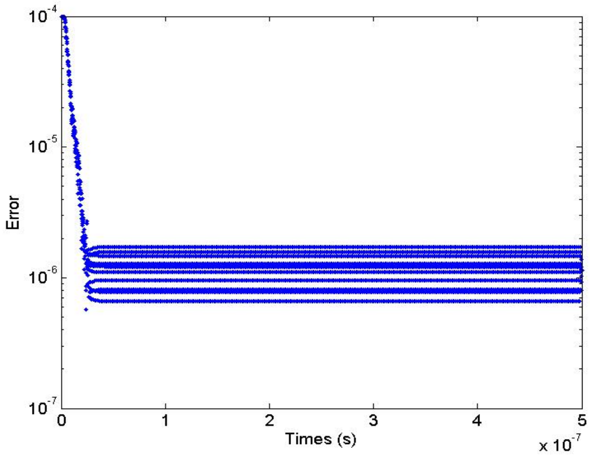
Take the initial vector $v(0) = (0,0,0,0)^{\mathrm{T}}$ and choose the weighted sign-bi-power activation function as $f(\cdot)$ with $\sigma = \frac{1}{3}$. State variables trajectories of the GNN model (13), for $k = 1$, with $\gamma = 200$ and $\lambda = 10^{-6}$ are graphically illustrated in Figure 6 (a) and (b), respectively.

From the Figures 6, we can see that ZNN state $X(t)$ can always converge to the theoretical solution of (1), whereas GNN state $X(t)$ does not fit well with the theoretical solution of (1).

In fact, the ZNN superiority comes from the fact that ZNN exploits the time-derivative information of matrix $A(t)$ and

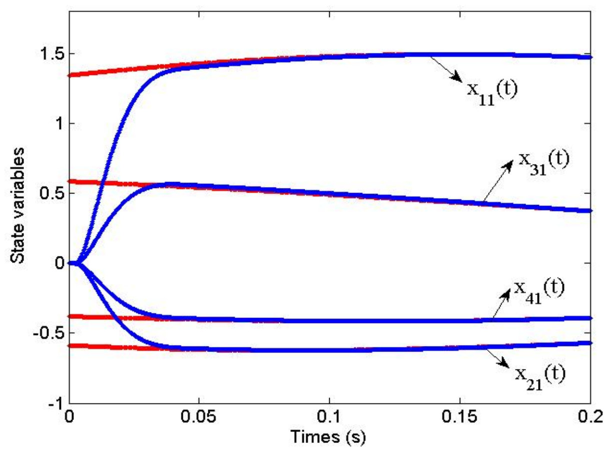


(a) $\gamma = 20$

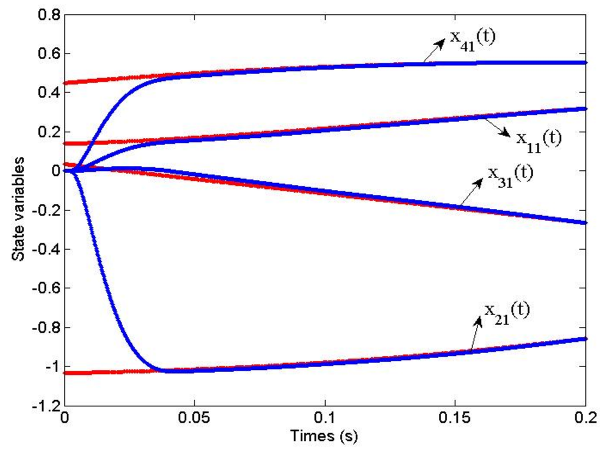


(b) Semilog plot with $\gamma = 2 \times 10^6$

Fig. 3. Trajectories of the residual errors of the model ZNN-II in Example 1.



(a) The real part of the first column.



(b) The imaginary part of the first column.

Fig. 4. Trajectories of the state variables of the model ZNN-I in Example 2.

$B(t)$ during the real-time inverting process, which ensures that ZNN could globally exponentially converge to the exact solution of the time-varying complex linear matrix equations. In contrast, GNN has not exploited such important information, and thus it may not be effective on solving such time-varying complex linear matrix equations.

VII. Conclusion and further considerations

In this paper, we have proposed two complex-valued neural networks for solving the matrix equation (1) under certain conditions. To achieve this goal, we have designed two new complex-valued activation functions based on the weighted sign-bi-power activation function. We have proved that the solution of our neural networks can converge to the theoretical solution of the matrix equation (1) in finite times. For comparative purposes, the conventional gradient-based neural networks (GNN) are also developed and exploited for solving such a time-varying complex linear matrix equation. The computer simulation results verify the superiorness of the ZNN models, as compared with the GNN models for solving time-varying complex linear matrix equations in complex domain.

Sometimes, the coefficient matrix $A(t)$ in (1) may be not a square matrix, that is, $A(t) \in \mathbb{C}^{m\times n}$ with $m \neq n$. Without loss of generality, suppose that $m > n$. Hence, the future work is to solving this matrix equation $A(t)X(t) = B(t)$, where all entries of $A(t)$ and $B(t)$ are smooth functions with respect to $t \geq 0$.

References

[1] J. B. Erway, R. F. Marcia, and J. A. Tyson, “Generalized diagonal pivoting methods for tridiagonal systems without interchanges,” *IAENG International Journal of Applied Mathematics*, vol. 40, no. 4, pp. 269–275, 2010.

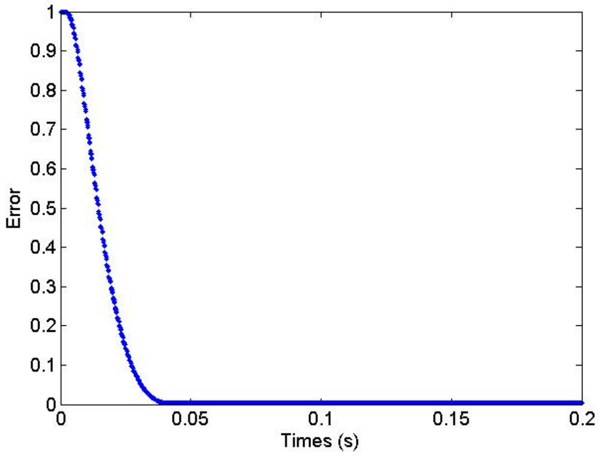
[2] F. Ding and T. Chen, “Gradient based iterative algorithms for solving a class of matrix equations,” *IEEE Transactions on Automatic Control*, vol. 50, no. 8, pp. 1216–1221, 2005.

[3] G. X. Huang, F. Yin, and K. Guo, “An iterative method for the skew-symmetric solution and the optimal approximate solution of the matrix equation AXB =C, AXB = C mathcontainer loading mathjax,” *Journal of Computational and Applied Mathematics*, vol. 212, no. 2, pp. 231–244, 2008.

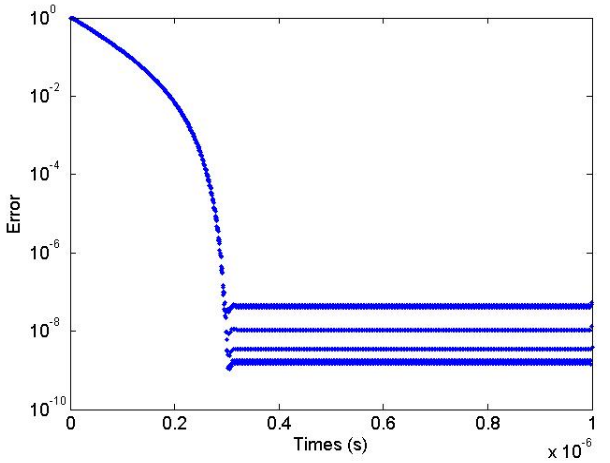
[4] J. Kurzak, A. Buttari, and J. Dongarra, “Solving systems of linear equations on the cell processor using cholesky factorization,” *IEEE Transactions on Parallel and Distributed Systems*, 2008.

[5] G. N. S. Prasanna and B. R. Musicus, “Generalized multiprocessor scheduling and applications to matrix computations,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 7, no. 6, pp. 650–664, 1996.

[6] M. U. Rehman, C. Vuik, and G. Segal, “Preconditioners for the steady incompressible navier-stokes problem,” *IAENG International Journal of Applied Mathematics*, vol. 38, no. 4, pp. 223–232, 2008.

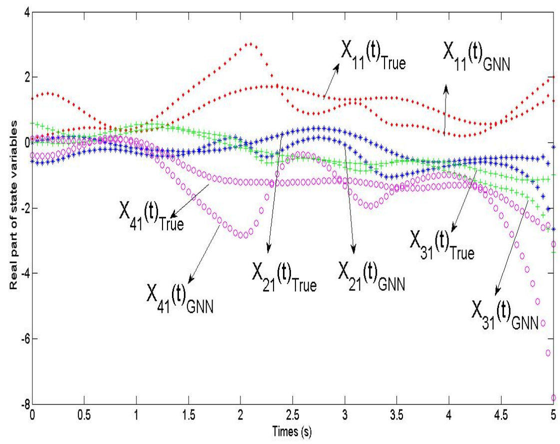


(a) $\gamma = 200$

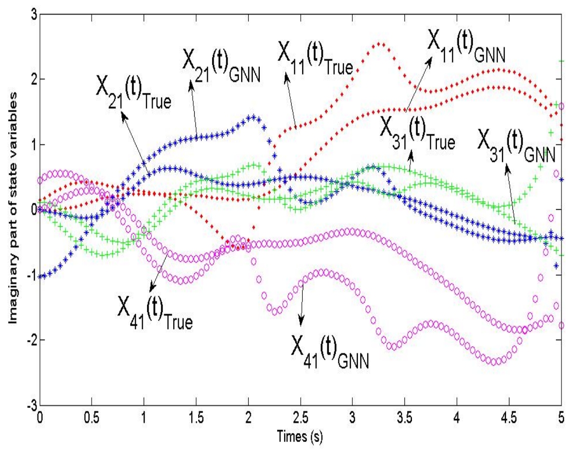


(b) Semilog plot with $\gamma = 2 \times 10^7$

Fig. 5. Trajectories of the residual errors of the model ZNN-I in Example 2.



(a) The real part of state variables.



(b) The imaginary part of state variables.

Fig. 6. Trajectories of the state variables of the model GNN, for $k = 1$, in Example 3.

[7] X. Wang and Z. Wang, "Preconditioned parallel multisplitting usaor methods for h-matrices linear systems," *IAENG International Journal of Applied Mathematics*, vol. 44, no. 1, pp. 45–52, 2014.

[8] A. G. Wu, L. Lv, and M. Z. Hou, "Finite iterative algorithms for extended sylvester-conjugate matrix equations," *Mathematical and Computer Modelling*, vol. 54, no. 9C10, pp. 2363–2384, 2011.

[9] B. Zhou and G. Duan, "An explicit solution to the matrix equation AX-XF = BY," *Linear Algebra and its Applications*, vol. 402, pp. 345–366, 2005.

[10] B. Zhou, G. R. Duan, and Z. Y. Li, "Gradient based iterative algorithm for solving coupled equations," *Systems and Control Letters*, vol. 58, no. 5, pp. 327–333, 2009.

[11] Z. Z. Bai, M. Benzi, and F. Chen, "Modified hss iteration methods for a class of complex symmetric linear systems," *Computing*, vol. 87, no. 3-4, pp. 93–111, 2010.

[12] O. Axelsson and A. Kucherov, "Real valued iterative methods for solving complex symmetric linear systems," *Numerical Linear Algebra with Applications*, vol. 7, no. 4, pp. 197–218, 2015.

[13] T. Wang, Q. Zheng, and L. Lu, "A new iteration method for a class of complex symmetric linear systems," *Journal of Computational and Applied Mathematics*, vol. 325, pp. 188–197, 2017.

[14] F. Ding, P. X. Liu, and J. Ding, "Iterative solutions of the generalized sylvester matrix equations by using the hierarchical identification principle," *Applied Mathematics and Computation*, vol. 197, no. 1, pp. 41–50, 2008.

[15] F. Ding and H. Zhang, "Gradient-based iterative algorithm for a class of the coupled matrix equations related to control systems," *Iet Control Theory and Applications*, vol. 8, no. 15, pp. 1588–1595, 2014.

[16] A. K. Gupta, *Numerical Methods Using Matlab*. Prentice Hall,, 1999.

[17] L. Xiao, "A finite-time convergent neural dynamics for online solution of time-varying linear complex matrix equation," *Neurocomputing*, vol. 167, no. C, pp. 254–259, 2015.

[18] D. Guo, C. Yi, and Y. Zhang, "Zhang neural network versus gradient-based neural network for time-varying linear matrix equation solving," *Neurocomputing*, vol. 74, no. 17, pp. 3708–3712, 2011.

[19] Q. Zhang and X. Wang, "Complex-valued neural network for hermitian matrices," *Engineering Letters*, vol. 25, no. 3, pp. 312–320, 2017.

[20] L. Xiao and Y. Zhang, "From different zhang functions to various ZNN models accelerated to finite-time convergence for time-varying linear matrix equation," *Neural Processing Letters*, vol. 39, no. 3, pp. 309–326, 2013.

[21] X. Z. Wang, Y. Wei, and P. S. Stanimirović, "Complex neural network models for time-varying Drazin inverse," *Neural Computation*, pp. 1–24, 2016.

[22] L. Xiao and Y. Zhang, "Zhang neural network versus gradient neural network for solving time-varying linear inequalities," *IEEE Transactions on Neural Networks*, vol. 22, no. 10, pp. 1676–84, 2011.

[23] ——, "Two new types of zhang neural networks solving systems of time-varying nonlinear inequalities," *IEEE Transactions on Circuits and Systems I*, vol. 59, no. 10, pp. 2363–2373, 2012.

[24] Y. Zhang and S. S. Ge, "Design and analysis of a general recurrent neural network model for time-varying matrix inversion," *IEEE Transactions on Neural Networks*, vol. 16, no. 6, pp. 1477–1490, 2005.

[25] Y. Zhang, D. Jiang, and J. Wang, "A recurrent neural network for solving sylvester equation with time-varying coefficients," *IEEE Transactions on Neural Networks*, vol. 13, no. 5, pp. 1053–63, 2002.

[26] Y. Zhang, W. Ma, and B. Cai, "From zhang neural network to newton iteration for matrix inversion," *IEEE Transactions on Circuits and Systems I*, vol. 56, no. 7, pp. 1405–1415, 2009.

[27] A. Hjorungnes and D. Gesbert, "Complex-valued matrix differentia-

tion: Techniques and key results," *IEEE Trans. Signal Process.*, vol. 55, no. 6p1, pp. 2740–2746, 2007.

[28] G. H. Golub, P. C. Hansen, and D. P. O'Leary, "Tikhonov regularization and total least squares," *SIAM. J. Matrix Anal. Appl.*, vol. 21, no. 1, pp. 185–194, 1999.

[29] Y. Shen, P. Miao, Y. Huang, and Y. Shen, "Finite-time stability and its application for solving time-varying sylvester equation by recurrent neural network," *Neural Processing Letters*, vol. 42, no. 3, pp. 763–784, 2015.