International Journal of Computational Intelligence Systems
Vol. **13(1)**, 2020, *pp.* 663–671
DOI: https://doi.org/10.2991/ijcis.d.200527.001; ISSN: 1875-6891; eISSN: 1875-6883
https://www.atlantis-press.com/journals/ijcis/

Research Article

A Neural Network for Moore–Penrose Inverse of Time-Varying Complex-Valued Matrices

Yiyuan Chai[1], Haojin Li[2], Defeng Qiao[2], Sitian Qin[2,*], Jiqiang Feng[1]

[1]*Shenzhen Key Laboratory of Advanced Machine Learning and Application, College of Mathematics and Statistics, Shenzhen University, Shenzhen, 518060, China*
[2]*Department of Mathematics, Harbin Institute of Technology, Weihai, 264209, China*

ARTICLE INFO

Article History
Received 01 Mar 2020
Accepted 17 May 2020

Keywords
Zhang neural network
Moore–Penrose inverse
Finite-time convergence
Noise suppression

ABSTRACT

The Moore–Penrose inverse of a matrix plays a very important role in practical applications. In general, it is not easy to immediately solve the Moore–Penrose inverse of a matrix, especially for solving the Moore–Penrose inverse of a complex-valued matrix in time-varying situations. To solve this problem conveniently, in this paper, a novel Zhang neural network (ZNN) with time-varying parameter that accelerates convergence is proposed, which can solve Moore–Penrose inverse of a matrix over complex field in real time. Analysis results show that the state solutions of the proposed model can achieve super convergence in finite time with weighted sign-bi-power activation function (WSBP) and the upper bound of the convergence time is calculated. A related noise-tolerance model which possesses finite-time convergence property is proved to be more efficient in noise suppression. At last, numerical simulation illustrates the performance of the proposed model as well.

1. INTRODUCTION

The Moore–Penrose inverse of a matrix is widely used in scientific and engineering fields as one of the basic problems encountered extensively, such as pattern recognition [1], optimization [2], and signal processing [3]. In general, it is not easy to immediately and accurately solve the Moore–Penrose inverse in numerical simulation. Because of its wide applications, the calculation of Moore–Penrose inverse has been studied a lot, and many algorithms for constant matrix have been proposed, such as continuous matrix square algorithm [4], Newton iteration method [5] and Greyville recursion [6]. However, in the process of solving matrix inverse kinematics problems of online control of redundant robots [7], the real-time calculation problem of the matrix Moore–Penrose inverse is required to be solved. Although there have been many effective methods for solving the matrix Moore–Penrose inverse, as far as we know, most of the research is based on the real field and few people generalize this problem to the complex field. In fact, solving the Moore–Penrose inverse of the complex-valued matrix is also often involved in the abovementioned fields, and it also plays a pivotal role in practical applications.

Parallel computing is widely used to solve linear and nonlinear problems [7–16] for its superior performance in complex large-scale online applications. Neural network is considered as an effective alternative to scientific computing [17,18] due to its parallel distribution and easy hardware implementation. Some recursive neural networks [19–21] have been constructed to compute the pseudo-inverse of matrices.

Zhang neural network (ZNN), as a special kind of recurrent neural network, is different from gradient neural networks (GNNs) that use a passive tracking method. ZNN performs better when involving the time derivative information of time-varying coefficients. It has the advantages of fast convergence rate in solving time-varying problems. And it is widely used in dynamic problems such as nonlinear dynamic optimization [22], motion control of robot redundant arms [23]. In 2013, a sign-bi-power activation function that accelerates ZNN to converge in finite time was proposed by Li *et al.* [24]. Then, Shen *et al.* [25] tried to accelerate the convergence rate of ZNN by constructing a tunable activation function. In addition, robustness is also an influence property because noise interference is unavoidable in practical application. Thus, inspired by xiao *et al.* [26], one noise-tolerance ZNN model which possesses finite-time convergence property is proposed in this paper.

It has been shown that ZNN has been proven to own superior property in many studies (see [27–29]). Therefore, for the time-varying full-rank matrix Moore–Penrose inversion problem, we choose to construct a ZNN to solve the time-varying optimization problem. The contributions of this paper are listed as follows:

(a) To the best of our knowledge, there is little research on solving time-varying Moore–Penrose inverse problems over complex field with ZNN model. Compared with the existing results [30], we generalize the problem to the complex field and solve the Moore–Penrose inverse in finite time.

* *Corresponding author. Email: qinsitian@163.com*

(b) A time-varying parameter is utilized in the design formula, which can effectively reduce the convergence time of the model solution.

(c) An improved ZNN model which is proved to be more efficient in noise suppression compared with traditional ZNN model is proposed, and the value of its error function converges to zero in finite time.

The remains of this paper are summarized as follows: Some definitions and preliminaries about generalized inverse and complex analysis are introduced in Section 2. The ZNN model for right Moore–Penrose inverse of a matrix with special activation function is constructed and its Lyapunov stability and convergence in finite time are proved in Section 3. In addition, an improved ZNN model for matrix Moore–Penrose inverse which has the ability to suppress noise is introduced. It is worth mentioning that the improved ZNN model can not only effectively suppress noise but also its solution can reach a finite time convergence under the acceleration of a special activation function. In Section 4, numerical examples of time-vary complex-valued problems are given to demonstrate the validity of our results.

2. PROBLEM DESCRIPTION AND PRELIMINARIES

In order to lay the foundation for further discussion, some definitions about the matrix Moore–Penrose inverse are introduced. In this paper, $\mathbb{C}^{m\times n}$ means $m\times n$-dimensional linear space over complex fields and $\mathbb{R}^{2n}$ means 2n-dimensional real vector space. A^H means conjugate transpose of matrix A.

Definition 1. [7,31] For a matrix $A\in\mathbb{C}^{m\times n}$, if $X\in\mathbb{C}^{n\times m}$ satisfies one or more of the following equations:

$$\begin{aligned} AXA &= A,\\ XAX &= X,\\ (AX)^H &= AX,\\ (XA)^H &= XA, \end{aligned}$$

then X is called the generalized inverse of A. And if X satisfies all of the equations above, then X is called the Moore–Penrose inverse of A, which is denoted by A^+.

If $rank(A) = min\{m,n\}$, AA^H (or A^HA) is nonsingular. Then the unique Moore–Penrose inverse $A^+\in\mathbb{C}^{m\times n}$ for matrix A could be rewritten as [30]

$$A^+ := \begin{cases} A^H(AA^H)^{-1}, & \text{if } m<n,\\ A^{-1}, & \text{if } m=n,\\ (A^HA)^{-1}A^H, & \text{if } m>n, \end{cases} \tag{1}$$

where the three equations from top to bottom respectively represent the right Moore–Penrose inverse, inverse and the left Moore–Penrose inverse.

From (1), the Moore–Penrose inverse of matrix A depends on $(AA^H)^{-1}$. However, calculating $(AA^H)^{-1}$ in high-dimensional case is difficult, so it is hard to calculate the Moore–Penrose inverse of matrix A at high dimension.

A^+ can not be gotten by (1) when $rank(A) < min\{m,n\}$, so, in this paper, the Moore–Penrose inverses of $A(t)\in\mathbb{C}^{m\times n}$ are only considered to be full-rank.

Definition 2. [32] For any $Z\in\mathbb{C}^n$, the linear function $\varphi:\mathbb{C}^n\to\mathbb{R}^{2n}$ is defined as

$$\varphi(Z) = \begin{pmatrix} \mathbf{Re}(Z)\\ \mathbf{Im}(Z)\end{pmatrix}\in\mathbb{R}^{2n}.$$

From the definition of $\varphi(Z)$, it is clear that

Proposition 1. *[32] For any $\lambda\in\mathbb{R}$ and $Z,Z_1,Z_2\in\mathbb{C}^n$, it holds that*

$$\begin{aligned} \varphi(\lambda Z) &= \lambda\varphi(Z);\\ \varphi(Z_1\pm Z_2) &= \varphi(Z_1)\pm\varphi(Z_2);\\ Z^HZ &= \varphi^T(Z)\varphi(Z). \end{aligned}$$

3. MODEL DESIGN AND CONVERGENCE ANALYSIS

In this section, we will propose a ZNN model and an improved noise-tolerance ZNN model to solve right time-varying Moore–Penrose inverse of matrices. Their stability and convergence are discussed as well.

3.1. A ZNN Model for Right Time-Varying Moore–Penrose Inverse

In this subsection, we will propose a ZNN model to solve the right Moore–Penrose inverse of time-varying matrix $A(t)$ with full row-rank. Hence, for (1), if $rank(A) = m$, the right Moore–Penrose inverse $A^+(t)$ satisfies

$$A^+(t)A(t)A^H(t) = A^H(t). \tag{2}$$

For solving $A^+(t)$, we first let $X(t) = A^+(t)$.

The corresponding error function [8,33] could be defined as

$$e(t) = X(t)A(t)A^H(t) - A^H(t), \tag{3}$$

then motivated by Zhang *et al.* [34], the following design formula is introduced

$$\dot{e}(t) = -\gamma_1 e^{\alpha_1 t}\Phi(e(t)), \tag{4}$$

where $\gamma_1,\alpha_1>0$ and $\Phi(\cdot):\mathbb{C}^{n\times m}\to\mathbb{C}^{n\times m}$ is activation function array. For any $z\in\mathbb{C}^{n\times m}$, the ijth element of the function $\Phi(\mathbf{z})$ is defined as

$$(\Phi(z))_{ij} = \phi\left(\mathbf{Re}(z_{ij})\right) + \mathbf{i}\phi\left(\mathbf{Im}(z_{ij})\right), \tag{5}$$

where $\phi(\cdot):\mathbb{R}\to\mathbb{R}$ is a monotonically increasing odd function constructed as [35]

$$\phi(x) = \lambda_1 sgn^k(x) + \lambda_2 sgn^{\frac{1}{k}}(x) + \lambda_3 x. \tag{6}$$

Here $\lambda_1, \lambda_2, \lambda_3 > 0,\ k \in (0,1)$,

$$sgn^k(x) = \begin{cases} |x|^k, & x > 0 \\ 0, & x = 0 \\ -|x|^k, & x < 0. \end{cases}$$

Combining (3) with (4), ZNN model is written as follows:

$$\begin{aligned} \dot{X}(t)A(t)A^H(t) &= -\gamma_1 e^{\alpha_1 t}\Phi\left(X(t)A(t)A^H(t) - A^H(t)\right) \\ &\quad -X(t)\left(\dot{A}(t)A^H(t) + A(t)\dot{A}^H(t)\right) + \dot{A}^H(t), \end{aligned}$$

which can be modified as

$$\begin{aligned} \dot{X}(t) &= \dot{X}(t)\left(I - A(t)A^H(t)\right) - \gamma_1 e^{\alpha_1 t}\Phi\left(X(t)A(t)A^H(t)\right. \\ &\quad \left. -A^H(t)\right) - X(t)\left(\dot{A}(t)A^H(t) + A(t)\dot{A}^H(t)\right) + \dot{A}^H(t). \end{aligned} \tag{7}$$

Theorem 2. *For any nonzero complex-valued activation function array $\Phi(\cdot)$ defined as (5), the zero solution of the complex ZNN design formula (4) remains stable in the sense of Lyapunov.*

Proof. For convenience, design formula (4) could be written as subsystems:

$$\dot{e}_{ij} = -\gamma_1 e^{\alpha_1 t}\left(\Phi\left(e(t)\right)\right)_{ij}, \tag{8}$$

where $i \in \{1, 2, \ldots, n\},\ j \in \{1, 2, \ldots, m\}$.

The Lyapunov function is defined as

$$v_{ij}(t) = e_{ij}^H(t)e_{ij}(t) = \varphi^T\left(e_{ij}(t)\right)\varphi\left(e_{ij}(t)\right), \tag{9}$$

where $e_{ij}(t)$ means the ijth element of $e(t)$. It's obvious that $v_{ij}(t) \geqslant 0$. $v_{ij}(t) = 0$ if and only if $e_{ij}(t) = 0$. That is $\mathbf{Re}(e_{ij}(t)) = 0$ and $\mathbf{Im}(e_{ij}(t)) = 0$.

The derivative with respect to time is

$$\begin{aligned} \dot{v}_{ij}(t) &= 2\varphi^T\left(e_{ij}(t)\right)\varphi\left(\dot{e}_{ij}(t)\right) \\ &= -2\gamma_1 e^{\alpha_1 t}\varphi^T\left(e_{ij}(t)\right)\varphi\left[\left(\Phi\left(e(t)\right)\right)_{ij}\right] \\ &= -2\gamma_1 e^{\alpha_1 t}\left[\mathbf{Re}\left(e_{ij}(t)\right)\phi\left(\mathbf{Re}\left(e_{ij}(t)\right)\right)\right. \\ &\quad \left. +\mathbf{Im}\left(e_{ij}(t)\right)\phi\left(\mathbf{Im}\left(e_{ij}(t)\right)\right)\right]. \end{aligned}$$

Since $\phi(\cdot)$ is a monotonically increasing odd function, it holds that

$$\begin{aligned} \mathbf{Re}\left(e_{ij}(t)\right)\phi\left(\mathbf{Re}\left(e_{ij}(t)\right)\right) &\geqslant 0 \\ \mathbf{Im}\left(e_{ij}(t)\right)\phi\left(\mathbf{Im}\left(e_{ij}(t)\right)\right) &\geqslant 0. \end{aligned}$$

So we can get that $\dot{v}_{ij} \leqslant 0$. And $\dot{v}_{ij} = 0$ if and only if $e_{ij}(t) = 0$. By the Lyapunov theory [36], the zero solution of (4) is stable in the sense of Lyapunov.

It is worth mentioning that under the acceleration of time-varying parameter $e^{\alpha_1 t}$, the solution of our proposed model not only keeps stable, but also achieves finite time convergence. The upper bound of convergent time of state solution $X(t)$ will be calculated and this can be described by the following theorem.

Theorem 3. *From any initial value $X(0) = X_0$, the solution of the ZNN model (7) with the activation function (6) will converge to the theoretical solution $A^+(t)$ in finite time.*

Proof. For the Lyapunov function v_{ij} in (9) and the ZNN model (7) with the activation function (6), we can get

$$\begin{aligned} \dot{v}_{ij}(t) &= -2\gamma_1 e^{\alpha_1 t}\left[\mathbf{Re}\left(e_{ij}(t)\right)\phi\left(\mathbf{Re}\left(e_{ij}(t)\right)\right)\right. \\ &\quad \left. +\mathbf{Im}\left(e_{ij}(t)\right)\phi\left(\mathbf{Im}\left(e_{ij}(t)\right)\right)\right] \\ &= -2\gamma_1 e^{\alpha_1 t}\mathbf{Re}\left(e_{ij}(t)\right)\left[\lambda_1 sgn^k\left(\mathbf{Re}\left(e_{ij}(t)\right)\right)\right. \\ &\quad \left. +\lambda_2 sgn^{\frac{1}{k}}\left(\mathbf{Re}\left(e_{ij}(t)\right)\right) + \lambda_3\mathbf{Re}\left(e_{ij}(t)\right)\right] \\ &\quad -2\gamma_1 e^{\alpha_1 t}\mathbf{Im}\left(e_{ij}(t)\right)\left[\lambda_1 sgn^k\left(\mathbf{Im}\left(e_{ij}(t)\right)\right)\right. \\ &\quad \left. +\lambda_2 sgn^{\frac{1}{k}}\left(\mathbf{Im}\left(e_{ij}(t)\right)\right) + \lambda_3\mathbf{Im}\left(e_{ij}(t)\right)\right] \\ &= -2\gamma_1 e^{\alpha_1 t}\left(\lambda_1\left[|\mathbf{Re}(e_{ij}(t))|^{k+1} + |\mathbf{Im}(e_{ij}(t))|^{k+1}\right]\right. \\ &\quad +\lambda_2\left[|\mathbf{Re}(e_{ij}(t))|^{\frac{k+1}{k}} + |\mathbf{Im}(e_{ij}(t))|^{\frac{k+1}{k}}\right] \\ &\quad \left. +\lambda_3\left[|\mathbf{Re}(e_{ij}(t))|^2 + |\mathbf{Im}(e_{ij}(t))|^2\right]\right) \\ &\leqslant -2\gamma_1 e^{\alpha_1 t}\left(\lambda_1\left[\left(|\mathbf{Re}(e_{ij}(t))|^2\right)^{\frac{k+1}{2}}\right.\right. \\ &\quad \left. + \left(|\mathbf{Im}(e_{ij}(t))|^2\right)^{\frac{k+1}{2}}\right] \\ &\quad \left. +\lambda_3\left[|\mathbf{Re}(e_{ij}(t))|^2 + |\mathbf{Im}(e_{ij}(t))|^2\right]\right) \\ &\leqslant -2\gamma_1 e^{\alpha_1 t}\left(\lambda_1\left[|\mathbf{Re}(e_{ij}(t))|^2 + |\mathbf{Im}(e_{ij}(t))|^2\right]^{\frac{k+1}{2}}\right. \\ &\quad \left. +\lambda_3\left[|\mathbf{Re}(e_{ij}(t))|^2 + |\mathbf{Im}(e_{ij}(t))|^2\right]\right) \\ &= -2\gamma_1 e^{\alpha_1 t}\left(\lambda_1 v_{ij}^{\frac{k+1}{2}}(t) + \lambda_3 v_{ij}(t)\right). \end{aligned}$$

That is,

$$\dot{v}_{ij}(t) \leqslant -2\gamma_1\lambda_1 e^{\alpha_1 t}v_{ij}^{\frac{k+1}{2}}(t) - 2\gamma_1\lambda_3 e^{\alpha_1 t}v_{ij}(t).$$

From the above inequality, we can conclude that $v_{ij}(t)$ is monotonically decreasing and positive definite, which means that $v_{ij}(t) = 0$ for every $t \geqslant T_{ij}$ when $v_{ij}(T_{ij}) = 0$. That is to say, $e_{ij}(t) = 0$ for every $t \geqslant T_{ij}$. So, without loss of generality, we suppose $v_{ij}(t) > 0$ for $t \in (0, T_{ij})$. Following inequality is obtained

$$v_{ij}^{-\frac{k+1}{2}}(t)\dot{v}_{ij}(t) \leqslant -2\gamma_1\lambda_1 e^{\alpha_1 t} - 2\gamma_1\lambda_3 e^{\alpha_1 t}v_{ij}^{\frac{1-k}{2}}(t).$$

for $t \in (0, T_{ij})$. Multiply both sides by $e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)e^{\alpha_1 t}}$ and we have

$$\begin{aligned} &\mathrm{d}\left(e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)e^{\alpha_1 t}}v_{ij}^{\frac{1-k}{2}}(t)\right) \\ &\leqslant -(1-k)\gamma_1\lambda_1 e^{\alpha_1 t + \frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)e^{\alpha_1 t}}\mathrm{d}t. \end{aligned}$$

Integrating both sides of this inequality from 0 to t for $t \in (0, T_{ij})$, one has

$$e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)e^{\alpha_1 t}} v_{ij}^{\frac{1-k}{2}}(t) - e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)} v_{ij}^{\frac{1-k}{2}}(0) \leqslant -\frac{\lambda_1}{\lambda_3} e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)e^{\alpha_1 t}} + \frac{\lambda_1}{\lambda_3} e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)}.$$

Since $v_{ij}(t) > 0$, we can get

$$\frac{\lambda_1}{\lambda_3} e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)e^{\alpha_1 t}} \leqslant e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)} v_{ij}^{\frac{1-k}{2}}(0) + \frac{\lambda_1}{\lambda_3} e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)}.$$

Therefore,

$$t \leqslant T_{ij} \triangleq \frac{1}{\alpha_1} \ln\left[\frac{\alpha_1}{\lambda_3(1-k)} ln\left(\frac{\lambda_3}{\lambda_1} e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)} v_{ij}^{\frac{1-k}{2}}(0) + e^{\frac{1}{\alpha_1}\gamma_1\lambda_3(1-k)}\right)\right].$$

The upper bound on the convergence time of $X(t)$ is $\tilde{T}_1$, which $\tilde{T}_1 = max\{T_{ij} \mid i = 1, \ldots, n,\ j = 1, \ldots, m\}$.

As is proved above, the state solution of ZNN model can converge to the theoretical solution $A^+(t)$ in finite time.

It is worth noting that compared with the methods proposed in other references [8,30], not only do we generalize the problem to the complex domain, but also the state solution of the model proposed in this paper can achieve finite time convergence.

Remark 1. ZNN model for left time-varying Moore–Penrose inverse of matrices can be obtained similarly. The error function could be defined as

$$\varepsilon(t) = A^H(t)A(t)X(t) - A^H(t), \tag{10}$$

and the following design formula is introduced

$$\dot{\varepsilon}(t) = -\gamma_2 e^{\alpha_2 t}\Phi(\varepsilon(t)), \tag{11}$$

where $\gamma_2, \alpha_2 > 0$. The corresponding ZNN model can be written as

$$\begin{aligned} A^H(t)A(t)\dot{X}(t) = & -\gamma_2 e^{\alpha_2 t}\Phi\left(A^H(t)A(t)X(t) - A^H(t)\right) \\ & - \left(\dot{A}^H(t)A(t) + A^H(t)\dot{A}(t)\right)X(t) + \dot{A}^H(t), \end{aligned}$$

which can be rewritten as

$$\begin{aligned} \dot{X}(t) = & \left(I - A^H(t)A(t)\right)\dot{X}(t) - \gamma_2 e^{\alpha_2 t}\Phi\left(A(t)^H A(t)X(t)\right. \\ & \left. -A^H(t)\right) - \left(\dot{A}^H(t)A(t) + A^H(t)\dot{A}(t)\right)X(t) + \dot{A}^H(t). \end{aligned} \tag{12}$$

The convergence analysis is similar to the proof of Theorems 2 and 3, thus omitted here.

3.2. An Improved Matrix Inversion ZNN Model

The analysis about ZNN model for right time-varying Moore–Penrose inverse in Section 3.1 does not concern the problem of external noise, which means the state solution of the ZNN model (7) and (12) may be unstable once the external noise appears. In this section, we propose an improved matrix inversion ZNN model which can also be more efficient in noise suppression.

We introduce a design formula [26] for $e(t)$ mentioned in (3) as follows:

$$\begin{aligned} \dot{e}(t) = & -\delta_1 e^{\alpha_1 t}\Phi\left(e(t)\right) \\ & -\delta_2\Phi\left(e(t) + \delta_1\int_0^t e^{\alpha_1\tau}\Phi(e(\tau))\mathrm{d}\tau\right), \end{aligned} \tag{13}$$

where $\delta_1, \delta_2 > 0$. By employing the design formula (13), following improved ZNN model for solving right Moore–Penrose inverse is obtained

$$\begin{aligned} \dot{X}(t)A(t)A^H(t) = & -\delta_1 e^{\alpha_1 t}\Phi\left(X(t)A(t)A^H(t) - A^H(t)\right) \\ & -\delta_2\Phi\left(X(t)A(t)A^H(t) - A^H(t)\right. \\ & \left. +\delta_1\int_0^t e^{\alpha_1\tau}\Phi(X(\tau)A(\tau)A^H(\tau) - A^H(\tau))\mathrm{d}\tau\right) \\ & -X(t)\left(\dot{A}(t)A^H(t) + A(t)\dot{A}^H(t)\right) + \dot{A}^H(t). \end{aligned} \tag{14}$$

The following theorem ensures the state solution of the ZNN model (14) converges to the exact solution in finite time.

Theorem 4. *From any initial state $X(0)$, the state solution $X(t)$ of the ZNN model (14) with the activation function $\Phi(\cdot)$ will converge to the exact solution $A^+(t)$.*

Proof. We firstly introduce an intermediate variable $r(t) = e(t) + \delta_1\int_0^t e^{\alpha_1\tau}\Phi(e(\tau))\mathrm{d}\tau$, it is clear that

$$\dot{r}(t) = \dot{e}(t) + \delta_1 e^{\alpha_1 t}\Phi(e(t)) = -\delta_2\Phi(r(t)).$$

Define a Lyapunov function as

$$w_{ij}(t) = r_{ij}^H(t)r_{ij}(t) = \varphi^T\left(r_{ij}(t)\right)\varphi\left(r_{ij}(t)\right),$$

where $r_{ij}(t)$ means the ijth element of $r(t)$. The derivative with respect to time is

$$\begin{aligned} \dot{w}_{ij}(t) = & 2\varphi^T\left(r_{ij}(t)\right)\varphi\left(\dot{r}_{ij}(t)\right) \\ = & -2\delta_2\varphi^T\left(r_{ij}(t)\right)\varphi\left[\left(\Phi\left(r(t)\right)\right)_{ij}\right] \\ = & -2\delta_2\left[\mathbf{Re}\left(r_{ij}(t)\right)\phi\left(\mathbf{Re}\left(r_{ij}(t)\right)\right)\right. \\ & \left. +\mathbf{Im}\left(r_{ij}(t)\right)\phi\left(\mathbf{Im}\left(r_{ij}(t)\right)\right)\right] \leqslant 0. \end{aligned} \tag{15}$$

Note that $w_{ij}(t) \geqslant 0$ and $\dot{w}_{ij}(t) \leqslant 0$. The fact that $w_{ij}(t) = 0$ holds if and only if $r_{ij}(t) = 0$ and $\dot{w}_{ij}(t) = 0$ holds if and only if $r_{ij}(t) = 0$. According to Lyapunov theory and analysis in Theorem 3, we know that there exists $\tilde{T}_2$, such that $r_{ij}(t) = 0$ for $t > \tilde{T}_2$. In this situation, (13) is further written as

$$\dot{e}(t) = -\delta_1 e^{\alpha_1 t}\Phi(e(t)). \tag{16}$$

It is similar to (8). According to the proof of Theorem 2, $e_{ij}(t)$ converges to zero finally. That is to say, $X(t)$ will converge to exact state solution finally.

Next, the upper bound of convergent time of state solution $X(t)$ will be calculated and this can be described by the following theorem.

Theorem 5. *Starting from any initial state $X(0)$, the state solution $X(t)$ of the ZNN model (14) with activation function $\Phi(\cdot)$ will converge to the exact solution $A^{+}(t)$ in finite time.*

Proof. According to (15), we can further get

$$\begin{aligned}
\dot{w}_{ij}(t) &= -2\delta_2 \mathbf{Re}\left(r_{ij}(t)\right)\left[\lambda_1 sgn^{k}\left(\mathbf{Re}\left(r_{ij}(t)\right)\right)\right. \\
&\quad \left.+\lambda_2 sgn^{\frac{1}{k}}\left(\mathbf{Re}\left(r_{ij}(t)\right)\right)+\lambda_3 \mathbf{Re}\left(r_{ij}(t)\right)\right] \\
&\quad -2\delta_2 e^{\alpha_1 t}\mathbf{Im}\left(r_{ij}(t)\right)\left[\lambda_1 sgn^{k}\left(\mathbf{Im}\left(r_{ij}(t)\right)\right)\right. \\
&\quad \left.+\lambda_2 sgn^{\frac{1}{k}}\left(\mathbf{Im}\left(r_{ij}(t)\right)\right)+\lambda_3 \mathbf{Im}\left(r_{ij}(t)\right)\right] \\
&= -2\delta_2\left(\lambda_1\left[|\mathbf{Re}(r_{ij}(t))|^{k+1}+|\mathbf{Im}(r_{ij}(t))|^{k+1}\right]\right. \\
&\quad +\lambda_2\left[|\mathbf{Re}(r_{ij}(t))|^{\frac{k+1}{k}}+|\mathbf{Im}(r_{ij}(t))|^{\frac{k+1}{k}}\right] \\
&\quad \left.+\lambda_3\left[|\mathbf{Re}(r_{ij}(t))|^{2}+|\mathbf{Im}(r_{ij}(t))|^{2}\right]\right) \\
&\leqslant -2\delta_2\left(\lambda_1\left[\left(|\mathbf{Re}(r_{ij}(t))|^{2}\right)^{\frac{k+1}{2}}\right.\right. \\
&\quad \left.+\left(|\mathbf{Im}(r_{ij}(t))|^{2}\right)^{\frac{k+1}{2}}\right] \\
&\quad \left.+\lambda_3\left[|\mathbf{Re}(r_{ij}(t))|^{2}+|\mathbf{Im}(r_{ij}(t))|^{2}\right]\right) \\
&\leqslant -2\delta_2\left(\lambda_1\left[|\mathbf{Re}(r_{ij}(t))|^{2}+|\mathbf{Im}(r_{ij}(t))|^{2}\right]^{\frac{k+1}{2}}\right. \\
&\quad \left.+\lambda_3\left[|\mathbf{Re}(r_{ij}(t))|^{2}+|\mathbf{Im}(r_{ij}(t))|^{2}\right]\right) \\
&= -2\delta_2\left(\lambda_1 w_{ij}^{\frac{k+1}{2}}(t)+\lambda_3 w_{ij}(t)\right).
\end{aligned}$$

Multiplying $e^{\delta_2\lambda_3(1-k)t}$ on both sides, we have

$$\mathrm{d}\left(e^{\delta_2\lambda_3(1-k)t}w_{ij}^{\frac{1-k}{2}}(t)\right) \leqslant -(1-k)\delta_2\lambda_1 e^{\delta_2\lambda_3(1-k)t}\mathrm{d}t,$$

which means that

$$t \leqslant T'_{ij} \triangleq \frac{1}{\delta_2\lambda_3(1-k)}\ln\left(1+\frac{\lambda_3}{\lambda_1}w_{ij}^{\frac{1-k}{2}}(0)\right).$$

Thus, we get $\tilde{T}_2 = \max\{T'_{ij} \mid i=1,\ldots,n,\ j=1,\ldots,m\}$. It follows that when $t \geqslant \tilde{T}_2$, we have $r_{ij}(t)=0$. In this situation, (13) reduces to (16). Hence, we obtain the upper bound of convergence time as follows:

$$\begin{aligned}
t \leqslant T''_{ij} \triangleq \frac{1}{\alpha_1} ln\left[\frac{\alpha_1}{\lambda_3(1-k)} ln\left(\frac{\lambda_3}{\lambda_1}e^{\frac{1}{\alpha_1}\delta_1\lambda_3(1-k)}v_{ij}^{\frac{1-k}{2}}(0)\right.\right. \\
\left.\left.+e^{\frac{1}{\alpha_1}\delta_1\lambda_3(1-k)}\right)\right],
\end{aligned}$$

which is similar to the proof of Theorem 3. Define $\tilde{T}_3 = max\left\{T''_{ij} \mid i=1,\ldots,n, j=1,\ldots,m\right\}$, we get the final convergence time is $\tilde{T}_N = \tilde{T}_2 + \tilde{T}_3$. The proof is complete.

To verify noise-tolerance ability, we add a constant external noise to design formula (13) which is denoted by η, and thus the following noise-polluted design formula is obtained

$$\begin{aligned}
\dot{e}(t) &= -\delta_1 e^{\alpha_1 t}\Phi\left(e(t)\right) \\
&\quad -\delta_2\Phi\left(e(t)+\delta_1\int_0^t e^{\alpha_1\tau}\Phi(e(\tau))\mathrm{d}\tau\right)+\eta,
\end{aligned} \tag{17}$$

and the noise-polluted ZNN model can be written as

$$\begin{aligned}
\dot{X}(t)A(t)A^{H}(t) &= -\delta_1 e^{\alpha_1 t}\Phi\left(X(t)A(t)A^{H}(t)-A^{H}(t)\right) \\
&\quad -\delta_2\Phi\left(X(t)A(t)A^{H}(t)-A^{H}(t)\right. \\
&\quad \left.+\delta_1\int_0^t e^{\alpha_1\tau}\Phi(X(\tau)A(\tau)A^{H}(\tau)-A^{H}(\tau))\mathrm{d}\tau\right) \\
&\quad -X(t)\left(\dot{A}(t)A^{H}(t)+A(t)\dot{A}^{H}(t)\right)+\dot{A}^{H}(t)+\eta.
\end{aligned} \tag{18}$$

Theorem 6. *For noise-polluted model (18), the state solution $X(t)$ starting from any initial state $X(0)$ will converge to the exact solution $A^{+}(t)$ with activation function $\Phi(\cdot)$.*

Proof. As defined in Theorem 4, $r(t) = e(t)+\delta_1\int_0^t e^{\alpha_1\tau}\Phi(e(\tau))\mathrm{d}\tau$. It can be written as a subsystem and its derivative with respect to time is

$$\dot{r}_{ij}(t) = \left(-\delta_2\Phi(r(t))+\eta\right)_{ij}.$$

Define a Lyapunov function as

$$W(t) = \left(\delta_2\Phi(r(t))-\eta\right)_{ij}^{H}\left(\delta_2\Phi(r(t))-\eta\right)_{ij}.$$

It is obvious that $W(t) \geqslant 0$. The fact that $W(t)=0$ holds if and only if $\left(\delta_2\Phi(r(t))-\eta\right)_{ij}=0$. Note that function array $\Phi(\cdot)$ are monotonically increasing odd functions, and its derivative about time can be calculated as

$$\begin{aligned}
\dot{W}(t) &= \left(\left(\delta_2\frac{\partial(\Phi(r(t)))_{ij}}{\partial r_{ij}}\dot{r}_{ij}(t)\right)^{H}\left(\delta_2\Phi(r(t))-\eta\right)_{ij}\right. \\
&\quad \left.+\left(\delta_2\Phi(r(t))-\eta\right)_{ij}^{H}\left(\delta_2\frac{\partial(\Phi(r(t)))_{ij}}{\partial r_{ij}}\dot{r}_{ij}(t)\right)\right) \\
&= -\left[\left(\delta_2\frac{\partial(\Phi(r(t)))_{ij}}{\partial r_{ij}}\right)^{H}+\left(\delta_2\frac{\partial(\Phi(r(t)))_{ij}}{\partial r_{ij}}\right)\right] \\
&\quad \left(\delta_2\Phi(r(t))-\eta\right)_{ij}^{H}\left(\delta_2\Phi(r(t))-\eta\right)_{ij} \\
&= -2\mathbf{Re}\left(\delta_2\frac{\partial(\Phi(r(t)))_{ij}}{\partial r_{ij}}\right) \\
&\quad \varphi^{T}\left(\left(\delta_2\Phi(r(t))-\eta\right)_{ij}\right)\varphi\left(\left(\delta_2\Phi(r(t))-\eta\right)_{ij}\right) \\
&\leqslant 0.
\end{aligned}$$

That is to say, r_{ij} will converge to the equilibrium according to Lyapunov theory. In other words, it means that $\lim_{t\to+\infty} \dot{r}_{ij}(t) = \lim_{t\to+\infty} (\delta_2\Phi(r(t)) - \eta)_{ij} = 0$. In this situation, it can be obtained that $\dot{e}_{ij}(t) = -\delta_1 e^{\alpha_1 t}(\Phi(e(t)))_{ij}$ which is similar to (8) due to $\dot{e}_{ij}(t) = \dot{r}_{ij}(t) - \delta_1 e^{\alpha_1 t}(\Phi(e(t)))_{ij}$. According to the proof of Theorem 2, $e_{ij}(t)$ converges to zero finally. That is to say, $X(t)$ will converge to the exact state solution finally.

4. NUMERICAL EXAMPLES

In this section, numerical examples are presented to illustrate the effectiveness of the proposed ZNN model.

Example 1. Consider the right Moore–Penrose inverse of the following complex time-varying matrix:

$$A(t) = \begin{pmatrix} A_1(t) & A_2(t) & A_3(t) \end{pmatrix},$$

where

$$A_1(t) = \begin{pmatrix} \sin(t) + \mathbf{i}\cos(t) \\ \cos(t) \end{pmatrix},$$

$$A_2(t) = \begin{pmatrix} \cos(t) + \mathbf{i}\sin(t) \\ \sin(t) + \mathbf{i}\cos(t) \end{pmatrix},$$

$$A_3(t) = \begin{pmatrix} \sin(t) \\ \cos(t) + \mathbf{i}\sin(t) \end{pmatrix}.$$

The Zhang neural network (ZNN) model (7) is used to simulate the Moore–Penrose inverse of the time-varying matrix in Example 1. Here, we take $\lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 1, \gamma_1 = 1, \alpha_1 = 1$ and $k = 0.5$. The results of the state solution are shown in Figure 1.

We divide the elements in the same position into real part and imaginary part and put them in the same diagram. There are four different initial values in every diagram. It can be seen from Figure 1 that starting from any four given initial values, the solution curves of model (7) converges to one state corresponding to the elements of $X(t)$ respectively and after that, the curves keep converging. This also illustrate the effectiveness of our proposed model for solving the matrix Moore–Penrose inverse problem.

In order to further illustrate the validity of the conclusion, we plot the trajectory of the error function and the results are presented in Figure 2. We also divide the elements in the same position into real part and imaginary part and put them in the same diagram. It can be seen that starting from the same four initial values as before, the real part and imaginary part of each element rapidly converge to zero, indicating that the state solution can converge well to the theoretical solution. Compared with the traditional ZNN model proposed in another reference [30] that also solves the matrix Moore–Penrose inverse, the results in Figure 3 show that the variable parameter $e^{\alpha_1 t}$ can indeed accelerate the convergence with the same initial values, and the convergence time of the model solution decreases as the parameter α_1 becomes larger.

Example 2. Consider the left Moore-Penrose inverse of the following complex time-varying matrix:

$$A = \begin{pmatrix} sin(t) + \mathbf{i}cos(t) & cos(t) + \mathbf{i}sin(t) \\ -cos(t) & sin(t) \\ sin(t) + \mathbf{i}cos(t) & cos(t) + \mathbf{i}sin(t) \end{pmatrix}.$$

The ZNN model (12) is used to simulate the Moore–Penrose inverse of the time-varying matrix in Example 2. In this case, taking $\lambda_1 = 1$, $\lambda_2 = 1, \lambda_3 = 1, \gamma_2 = 1, \alpha_2 = 1$ and $k = 0.5$.

Choosing four different initial values arbitrarily in every diagram, we divide the elements in the same position into real part and

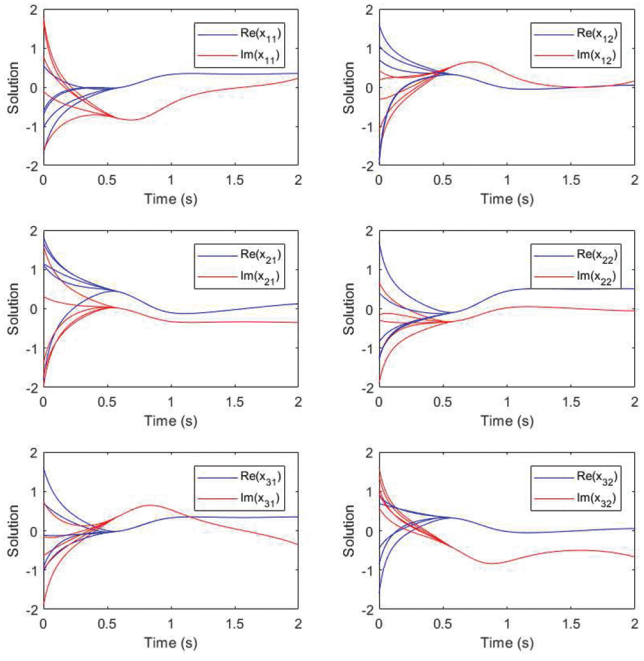

Figure 1 State trajectories of ZNN models (7) for Example 1.

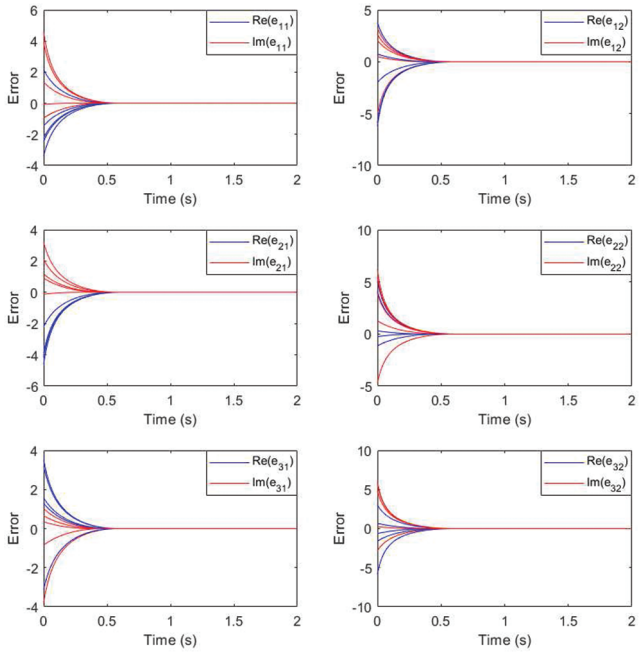

Figure 2 State trajectories of error function (3) for Example 1.

imaginary part. In order to reflect the convergence speed more intuitively, we put them in the same diagram. Similarly, as shown in Figures 4 and 5, we can observe that starting from any four given initial values, the solution curves of model (12) converges to one state corresponding to the elements of $X(t)$ respectively. That is to say, our model effectively solves the proposed problem of time-varying Moore–Penrose inverse.

Example 3. Consider the right Moore–Penrose inverse of the following complex time-varying matrix with noise:

$$A(t) = \left(A_1(t)\ A_2(t)\ A_3(t) \right)$$

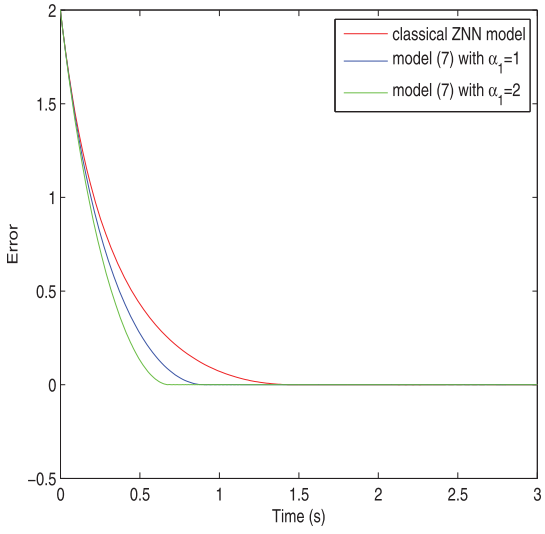

Figure 3 Influence of parameters on convergence time for Example 1.

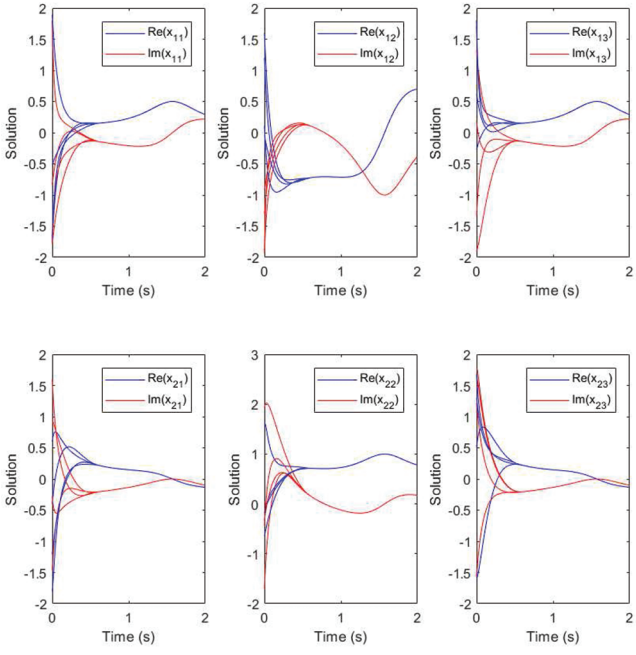

Figure 4 State trajectories of Zhang neural network (ZNN) models (12) for Example 2.

where

$$A_1(t) = \begin{pmatrix} \sin(t) + \mathbf{i}\cos(t) \\ \cos(t) \end{pmatrix},$$

$$A_2(t) = \begin{pmatrix} \cos(t) + \mathbf{i}\sin(t) \\ \sin(t) + \mathbf{i}\cos(t) \end{pmatrix},$$

$$A_3(t) = \begin{pmatrix} \sin(t) \\ \cos(t) + \mathbf{i}\sin(t) \end{pmatrix}.$$

We first examine the stability of the model without interference from external noise. In order to further illustrate the validity of the conclusion, we plot the trajectory of the error function and it is shown in Figure 6. Obviously, every element in the error function, whether real or imaginary, will eventually converge to zero, which effectively proves that the model (14) can solve the time-varying Moore–Penrose inverse of matrices. Next, we randomly choose a set of constant noise values and compare the convergence of the error curves under this noise interference. It is clear that the error curves which are represented by the blue solid line and the blue dotted line in Figure 7 of model (14) both converges to zero faster. However, the error curves of model (7), which are the curves represented by red in Figure 7, converge slowly under the interference of external noise. In other words, model (14) does better in noise suppression.

5. CONCLUSION

In this paper, for solving time-varying Moore–Penrose inverse over complex fields, a new ZNN model is proposed. The solution of the ZNN model (7) and the solution of the ZNN model (12) are proved to be stable in the sense of Lyapunov. Furthermore, for

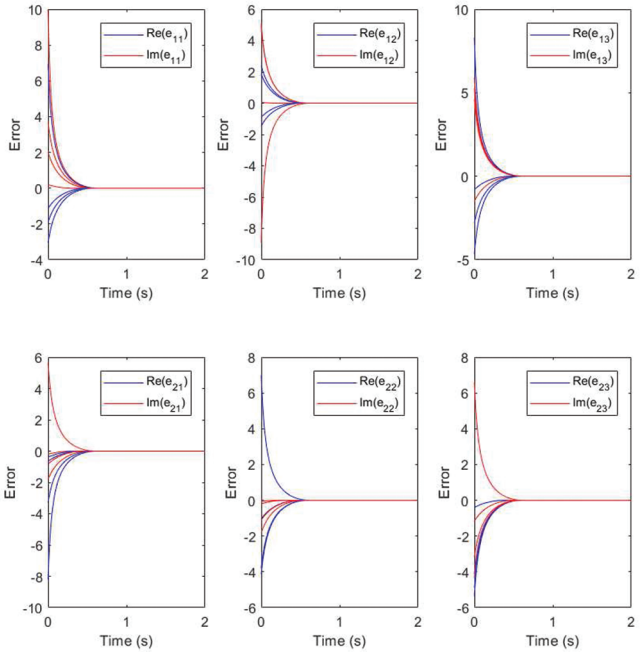

Figure 5 State trajectories of error function (10) for Example 2.

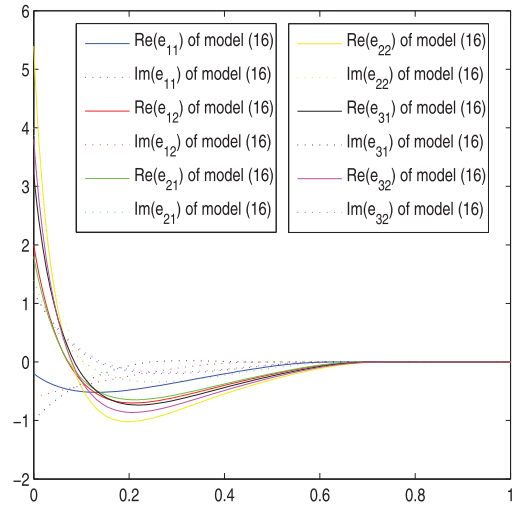

Figure 6 | Trajectories of error function without noise for Example 3.

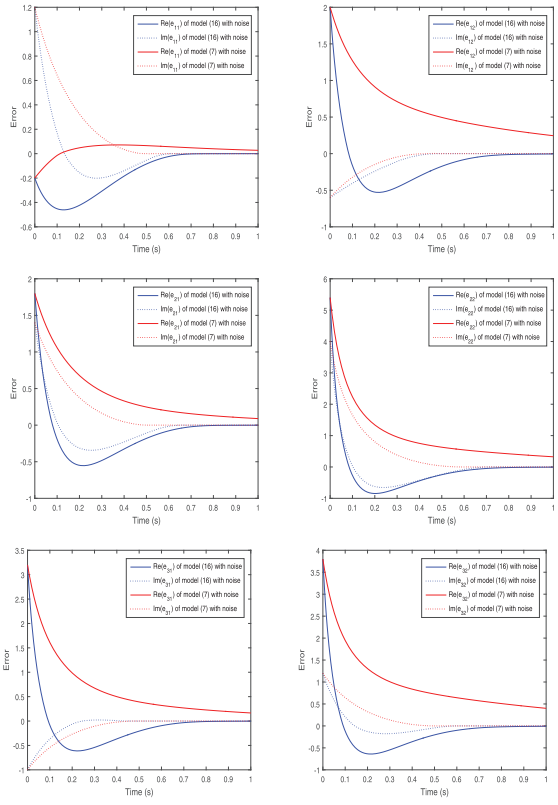

Figure 7 | Trajectories of error function with noise for Example 3.

any initial value, the state solution of ZNN will converge to the theoretical solution in real time. Compared with existing results, our model converges faster because of the new activation function. The improved ZNN model (14) is proven to be more efficient in noise suppression. In addition, models with faster convergence rate and models for solving Moore–Penrose inverses of rank-deficient matrix would be one of our future research directions, which might achieve rich results under unremitting efforts.

CONFLICT OF INTEREST

The authors declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled.

AUTHORS' CONTRIBUTIONS

Yiyuan Chai: Conceptualization and Methodology; Haojin Li: Writing- Original draft preparation; Defeng Qiao: Computer programs; Sitian Qin: Supervision, Validation and Investigation; Jiqiang Feng: Writing- Reviewing and Editing.

ACKNOWLEDGMENTS

This research is supported by the National Natural Science Foundation of China (61773136, 11871178).

REFERENCES

[1] J. Liu, S. Chen, X. Tan, D. Zhang, Efficient pseudoinverse linear discriminant analysis and its nonlinear form for face recognition, Int. J. Pattern Recognit. Artif. Intell. 21 (2007), 1265–1278.
[2] H.H. Bauschke, J.M. Borwein, X. Wang, Fitzpatrick functions and continuous linear monotone operators, Siam J. Optim. 18 (2007), 789–809.
[3] T. Yahagi, A deterministic approach to optimal linear digital equalizers, IEEE Trans. Acoustics Speech Signal Process. 31 (1983), 491–500.
[4] L. Chen, E.V. Krishnamurthy, I. Macleod, Generalised matrix inversion and rank computation by successive matrix powering, Parallel Comput. 20 (1994), 297–311.
[5] Y. Wei, J. Cai, M.K. Ng, Computing moore-penrose inverses of toeplitz matrices by newton's iteration, Math. Comput. Model. 40 (2004), 181–191.
[6] J. Zhou, Y. Zhu, X.R. Li, Z. You, Variants of the greville formula with applications to exact recursive least squares, SIAM J. Matrix Anal. Appl. 24 (2002), 150–164.
[7] J. Wang, Recurrent neural networks for computing pseudoinverses of rank-deficient matrices, Siam J. Sci. Comput. 18 (1997), 1479–1493.
[8] Y. Zhang, S.S. Ge, Design and analysis of a general recurrent neural network model for time-varying matrix inversion, IEEE Trans. Neural Netw. 16 (2005), 1477–1490.
[9] Y. Wu, X. Sun, X. Zhao, T. Shen, Optimal control of boolean control networks with average cost: a policy iteration approach, Automatica. 100 (2019), 378–387.
[10] M. Toyoda, Y. Wu, On optimal time-varying feedback controllability for probabilistic boolean control networks, IEEE Trans. Neural Netw. Learn. Syst. 31 (2020), 2202–2208.
[11] S. Chen, Y. Wu, M. Macauley, X.-M. Sun, Monostability and bistability of boolean networks using semi-tensor products, IEEE Trans. Control Netw. Syst. 6 (2019), 1379–1390.
[12] N. Liu, S. Qin, A novel neurodynamic approach to constrained complex-variable pseudoconvex optimization, IEEE Trans. Cybern. 49 (2019), 3946–3956.

[13] D. Li, M. Han, J. Wang, Chaotic time series prediction based on a novel robust echo state network, IEEE Trans. Neural Netw. Learn. Syst. 23 (2012), 787–799.
[14] S. Qin, Y. Liu, X. Xue, F. Wang, A neurodynamic approach to convex optimization problems with general constraint, Neural Netw. 84 (2016), 113–124.
[15] S. Qin, J. Feng, J. Song, X. Wen, C. Xu, A one-layer recurrent neural network for constrained complex-variable convex optimization, IEEE Trans. Neural Netw. Learn. Syst. 29 (2018), 534–544.
[16] S. Qin, X. Xue, A two-layer recurrent neural network for nonsmooth convex optimization problems, IEEE Trans. Neural Netw. Learn. Syst. 26 (2015), 1149–1160.
[17] N. Smaoui, A model for the unstable manifold of the bursting behavior in the 2d navier–stokes flow, SIAM J. Sci. Comput. 23 (2001), 824–839.
[18] P.V. Den Driessche, X. Zou, Global attractivity in delayed hopfield neural network models, Siam J. Appl. Math. 58 (1998), 1878–1890.
[19] L. Falong, B. Zheng, Neural network approach to computing matrix inversion, Appl. Math. Comput. 47 (1992), 109–120.
[20] J. Song, Y. Yam, Complex recurrent neural network for computing the inverse and pseudo-inverse of the complex matrix, Appl. Math. Comput. 93 (1998), 195–205.
[21] P.S. Stanimirovic, I.S. ivkovic, Y. Wei, Recurrent neural network approach based on the integral representation of the drazin inverse, Neural Comput. 27 (2015), 2107–2131.
[22] L. Jin, S. Li, Nonconvex function activated zeroing neural network models for dynamic quadratic programming subject to equality and inequality constraints, Neurocomputing. 267 (2017), 107–113.
[23] D. Guo, Y. Zhang, Li-function activated znn with finite-time convergence applied to redundant-manipulator kinematic control via time-varying jacobian matrix pseudoinversion, Appl. Soft Comput. 24 (2014), 158–168.
[24] S. Li, S. Chen, B. Liu, Accelerating a recurrent neural network to finite-time convergence for solving time-varying sylvester equation by using a sign-bi-power activation function, Neural Process. Lett. 37 (2013), 189–205.
[25] Y. Shen, P. Miao, Y. Huang, Y. Shen, Finite-time stability and its application for solving time-varying sylvester equation by recurrent neural network, Neural Process. Lett. 42 (2015), 763–784.
[26] L. Xiao, Design and analysis of robust nonlinear neural dynamics for solving dynamic nonlinear equation within finite time, Nonlinear Dyn. 96 (2019), 2437–2447.
[27] D. Guo, C. Yi, Y. Zhang, Zhang neural network *versus* gradient-based neural network for time-varying linear matrix equation solving, Neurocomputing. 74 (2011), 3708–3712.
[28] S. Ding, M. Yang, M. Mao, L. Xiao, Y. Zhang, Complex ZNN and GNN models for time-varying complex quadratic programming subject to equality constraints, in World Congress on Intelligent-Control and Automation, Guilin, China, 2016, pp. 210–215.
[29] X. Lin, B. Liao, A convergence-accelerated zhang neural network and its solution application to lyapunov equation, Neurocomputing. 193 (2016), 213–218.
[30] Y. Zhang, Y. Yang, N. Tan, B. Cai, Zhang neural network solving for time-varying full-rank matrix moore–penrose inverse, Computing. 92 (2011), 97–121.
[31] J.A. Fill, D.E. Fishkind, The moore–penrose generalized inverse for sums of matrices, SIAM J. Matrix Anal. Appl. 21 (1999), 629–635.
[32] Q. Ma, S. Qin, T. Jin, Complex zhang neural networks for complex-variable dynamic quadratic programming, Neurocomputing. 330 (2019), 56–69.
[33] Y. Zhang, D. Jiang, J. Wang, A recurrent neural network for solving sylvester equation with time-varying coefficients, IEEE Trans. Neural Netw. 13 (2002), 1053–1063.
[34] Z. Zhang, L. Zheng, J. Weng, Y. Mao, W. Lu, L. Xiao, A new varying-parameter recurrent neural-network for online solution of time-varying sylvester equation, IEEE Trans. Syst. Man Cybern. 48 (2018), 3135–3148.
[35] L. Xiao, A nonlinearly activated neural dynamics and its finite-time solution to time-varying nonlinear equation, Neurocomputing. 173 (2016), 1983–1988.
[36] Y. Zhang, C. Yi, Zhang Neural Networks and Neural-Dynamic Method, Nova Science Publishers, New York, NY, USA, 2011.