

Design and Analysis of a Novel Integral Recurrent Neural Network for Solving Time-Varying Sylvester Equation

Zhijun Zhang^{ID}, Senior Member, IEEE, Lunan Zheng^{ID}, Student Member, IEEE,
Hui Yang^{ID}, and Xilong Qu

Abstract—To solve a general time-varying Sylvester equation, a novel integral recurrent neural network (IRNN) is designed and analyzed. This kind of recurrent neural networks is based on an error-integral design equation and does not need training in advance. The IRNN can achieve global convergence performance and strong robustness if odd-monotonically increasing activation functions [i.e., the linear, bipolar-sigmoid, power, or sigmoid-power activation functions (SP-AFs)] are applied. Specifically, if linear or bipolar-sigmoid activation functions are applied, the IRNN possess exponential convergence performance. The IRNN has finite-time convergence property by using power activation function. To obtain faster convergence performance and finite-time convergence property, an SP-AF is designed. Furthermore, by using the discretization method, the discrete IRNN model and its convergence analysis are also presented. Practical application to robot manipulator and computer simulation results with using

Manuscript received March 29, 2019; revised August 9, 2019 and August 18, 2019; accepted August 26, 2019. This work was supported in part by the National Natural Science Foundation under Grant 61976096, Grant 61603142, Grant 61633010, and Grant 61733005, in part by the Guangdong Foundation for Distinguished Young Scholars under Grant 2017A030306009, in part by the Guangdong Special Support Program under Grant 2017TQ04X475, in part by the Science and Technology Program of Guangzhou under Grant 201707010225, in part by the Fundamental Research Funds for Central Universities under Grant x2zdD2182410, in part by the Scientific Research Starting Foundation of South China University of Technology, in part by the National Key Research and Development Program of China under Grant 2017YFB1002505, in part by the National Key Basic Research Program of China (973 Program) under Grant 2015CB351703, in part by the Guangdong Natural Science Foundation under Grant 2014A030312005, in part by the Research Project of Education Department of Hunan Province under Grant 18A441 and Grant 18C0958, in part by the Open Research Fund of Hunan Provincial Key Laboratory of Network Investigational Technology under Grant 2017WLZC006, and in part by the Key Project of Educational Planning of Hunan Province under Grant XJK18DJA1. This article was recommended by Associate Editor Y. Xia. (Corresponding author: Xilong Qu.)

Z. Zhang is with the School of Automation Science and Engineering, South China University of Technology, Guangzhou 510640, China, and also with the Hunan Provincial Key Laboratory of Finance and Economics Big Data Science and Technology, School of Information Technology and Management, Hunan University of Finance and Economics, Changsha 410205, China (e-mail: drzhangzhijun@gmail.com).

L. Zheng is with the School of Automation Science and Engineering, South China University of Technology, Guangzhou 510640, China (e-mail: aulnzheng@mail.scut.edu.cn).

H. Yang is with the Key Laboratory of Advanced Control and Optimization of Jiangxi Province, School of Electrical and Automation Engineering, East China Jiaotong University, Nanchang 330013, China (e-mail: yhsuo@263.net).

X. Qu is with the School of Information Technology and Management, Hunan University of Finance and Economics, Changsha 410006, China (e-mail: quxilong@126.com).

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

Digital Object Identifier 10.1109/TCYB.2019.2939350

different activation functions and design parameters have verified the effectiveness, stability, and reliability of the proposed IRNN.

Index Terms—Computer simulations, convergence and robustness, recurrent neural networks (RNNs), time-varying equation solving.

I. INTRODUCTION

ARTIFICIAL neural network, as a kind of mathematical model of parallel distributed information processing approach by imitating the nervous system of animals [1], has attracted the attention of many researchers and engineers in the past few decades [2]–[6]. As a crucial kind of artificial neural network, the recurrent neural network (RNN) has been applied to many fields extensively, such as speech recognition [7]; nonconvex optimization [8]; algebraic problems [9]–[11]; time-varying problems [12], [13]; unmanned aerial vehicles [14], [15]; and robotics [16], [17]. Since Hopfield first proposed a method for studying the stability of an RNN with fixed weights by using energy function in the 1980s [18], this kind of research problems about the stability of fixed weights RNNs has obtained a lot of attention. The RNNs with fixed weights and feedback structure (such as Hopfield network [19]; Cohen–Grossberg network [20]; BAM neural network [21]; gradient-based neural network (GNN) [22], [23]; and zeroing neural network (ZNN) [6], [10], [11], [13]) can be used in optimal computation, associative memory, and pattern recognition. For solving the optimizing problems, especially the algebraic problems, the GNN and ZNN are two kinds of effective methods.

Sylvester equation (i.e., $AX - XB + C = 0$) is a kind of ubiquitous significant linear equation in matrix theory and algebra. It can be applied to analysis and synthesis of mathematics, dynamic control system, or computer vision problems, for example, linear least-squares regression [24]; optimization [25]; control system [14], [26]; image fusion [27]; clustering [28]; etc. For solving the time-invariant problems, such as matrix inversion and Sylvester equation, GNNs are proved to be effective [15], [22]. However, the GNN cannot track the time-varying state variables of a time-varying problem [29]. For solving the time-varying problems, ZNN, as an implicit dynamics method by considering the time derivative of the error function, is proposed in recent

years [6], [10], [11], [13], [30]. To improve the noise-tolerant ability of the RNN, an integral term was incorporated into the traditional differential neural network (i.e., ZNN) and then the noise-tolerant ZNN (NTZNN) was proposed [5], [6]. Because of the integral term, compared with the ZNN, NTZNN not only has the globally exponential convergence property but also has the capability of noise suppressing. Since the NTZNN is a differential neural network with an integral term, it can be seen as a mixed neural network. In this article, an integral RNN (IRNN) is proposed by just considering the integral function. However, the analyses to IRNN show that it has better convergence performance than the NTZNN.

The motivation of this article is listed as follows.

Motivation 1 (Stronger Anti-Interference Capability): In practical applications, disturbances and model errors always exist. The IRNN has a better capacity of resisting disturbance with its innate integral structure. This is the most important difference compared with the differential neural networks.

Motivation 2 (Novel Network Structure): With more and more complicated application requirements, many kinds of RNNs with different network structures and properties are needed. It is necessary to design more effective neural network for solving the time-varying problems and for improving the realizability and reliability of the neural networks. Inspired by the differential neural networks [5], [6], [10], [11], [13], [30], through dialectical thinking, a novel IRNN is proposed. Being different from the traditional RNNs based on the differential equation form, the IRNN is constructed by error-integral design equation and can obtain better convergence performance.

Motivation 3 (Better Convergence Property): Compared with the NTZNN method [5], [6], because of the novel network structure, the IRNN has faster convergence speed.

The remainder of this article is organized as follows. In Section II, the problem formulation of the Sylvester equation is presented and a novel IRNN is proposed. In Section III, the global convergence performance and detailed convergence properties by using different activation functions of the proposed IRNN are proved theoretically. Robustness analysis of the IRNN is presented in Section IV. A discrete IRNN formula is proposed in Section V and its convergence and error analysis are given. For illustrating the effectiveness, reliability, and stability of the IRNN, computer simulation results based on MATLAB Simulink are shown in Section VI. The practical application to the motion planning problem is presented in Section VII for verifying the practicability of the IRNN. A briefly summary is given in Section VIII. Before ending this section, the main contributions are summarized as follows.

- 1) A novel IRNN for solving the time-varying Sylvester equations is proposed, analyzed, and proved to be effective. To design the IRNN, an error-integral design equation is presented.
- 2) Global convergence performance of the IRNN with an odd-monotonically increasing activation function is proved and the influences of design parameters are discussed.
- 3) The convergence properties (i.e., exponential convergence performance by using the linear or

bipolar-sigmoid activation functions and finite-time convergence property by using a power activation function) and upper bounds of the IRNN are analyzed and proved. A new sigmoid-power activation function (SP-AF) is designed, and the faster and finite-time convergence of the IRNN with this activation function is proved.

- 4) The robustness of the IRNN is discussed in consideration of the model-implementation error.
- 5) Discrete IRNN is proposed and its convergence and error analysis are proved.
- 6) Simulation results further verify the convergence and robustness theorems proposed in this article, and further illustrate the effectiveness and reliability of the IRNN for solving a practical Sylvester equation.

II. PROBLEM FORMULATION AND THE IRNN MODEL

First, to lay a basis for the discussion, the problem formulation and the preliminaries are presented. Second, the design procedure and the model of the IRNN are illustrated.

A. Problem Formulation

A smoothly time-varying Sylvester equation is

$$A(t)X(t) - X(t)B(t) + C(t) = 0, \quad \forall t \in [0, +\infty) \quad (1)$$

where t denotes the time variable. The time-varying coefficient matrices $A(t) = (a_{ij}(t))_{m \times m}$, $B(t) = (b_{ij}(t))_{n \times n}$, and $C(t) = (c_{ij}(t))_{m \times n}$ as well as their time derivatives matrices $\dot{A}(t)$, $\dot{B}(t)$, and $\dot{C}(t)$ are assumed to be known or can be estimated accurately. If such a Sylvester equation holds true, that is, the unique solution $X(t) = X^*(t) \in \mathbb{R}^{m \times n}$ of the Sylvester equation exists, it is expected to find the accurate solution as fast as possible.

To solve the Sylvester equation (1) and monitor the time-varying process, the matrix-type error function is defined as

$$E(t) := A(t)X(t) - X(t)B(t) + C(t). \quad (2)$$

The time-varying unique solution $X^*(t)$ of the time-varying Sylvester equation (1) could be obtained if the error function (2) equals to zero or converges to zero.

B. Integral Recurrent Neural Network Model

A novel IRNN based on the error-integral design equation is proposed in this section. The error-integral neural network design equation can be formulated as

$$E(t) = E_0 - \int_0^t E(\tau)d\tau \quad (3)$$

where $E_0 = E(0) = A(0)X(0) - X(0)B(0) + C(0)$ is the initial value of the error function. $A(0)$, $B(0)$, and $C(0)$ are the initial coefficient matrices and $X(0) = X_0$ is the presupposed state matrix. With the consideration of the error function (2), the integral of the error will increase and $E_0 - \int_0^t E(\tau)d\tau$ will decrease, which implies that $E(t)$ will decrease, too.

To illustrate the design procedure more clearly, the error-integral design equation is reformulated equivalently as

$$\int_0^t E(\tau)d\tau = E_0 - E(t). \quad (4)$$

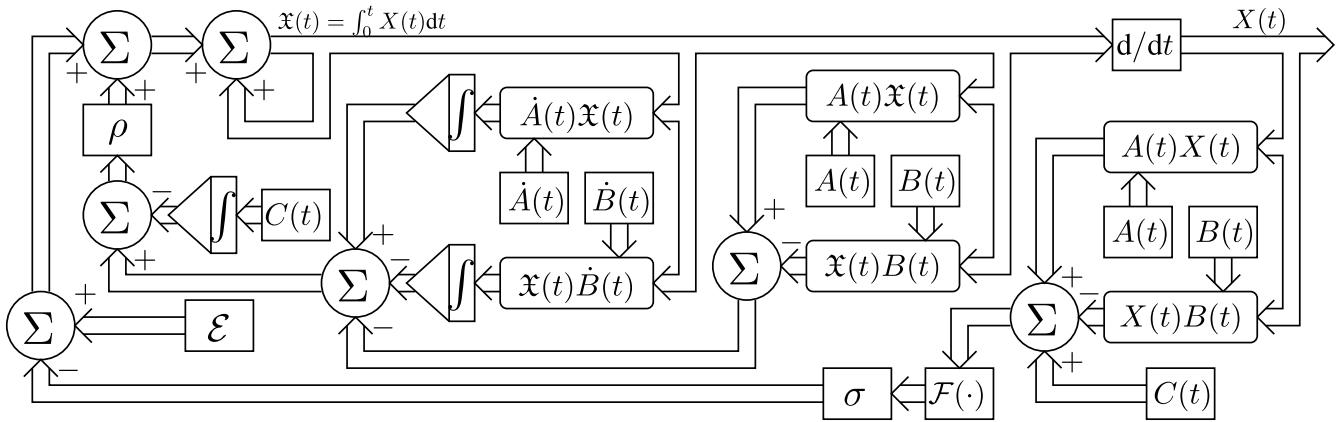


Fig. 1. Block diagram of the IRNN model (9) for solving the Sylvester equation.

For obtaining better convergence and robustness properties, inspired by neural network thought [1], activation functions are considered and (4) can be reshaped as

$$\rho \int_0^t E(\tau) d\tau = \mathcal{E} - \sigma \mathcal{F}(E(t)) \quad (5)$$

where $\mathcal{F}(\cdot) : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$ denotes the activation function array; and σ and ρ are the positive scalar-valued parameters which are used to scale the convergence rate of the neural network. It is worth mentioning that ρ can adjust the residual error of the IRNN. Since $\int_0^0 E(\tau) d\tau = 0$, the parameter \mathcal{E} should satisfy the initial condition, that is, $\mathcal{E} = \sigma \mathcal{F}(E(0))$.

Substituting (2) into (5), we have

$$\begin{aligned} \rho \int_0^t (A(\tau)X(\tau) - X(\tau)B(\tau) + C(\tau)) d\tau \\ = \mathcal{E} - \sigma \mathcal{F}(A(t)X(t) - X(t)B(t) + C(t)). \end{aligned} \quad (6)$$

According to the definition of integral, we can define

$$\Xi(t) = \int_0^t X(\tau) d\tau \quad (7)$$

where $\dot{\Xi}(t) = X(t)$. Considering the chain rule about derivative [31], we have $d(A(t)\Xi(t))/dt = \dot{A}(t)\Xi(t) + A(t)\dot{\Xi}(t)$, that is, $A(t)X(t) = d(A(t)\Xi(t))/dt - \dot{A}(t)\Xi(t)$. Similarly, we have $X(t)B(t) = d(\Xi(t)B(t))/dt - \Xi(t)\dot{B}(t)$. Therefore, the design formula (6) can be reformulated as

$$\begin{aligned} \rho \int_0^t (A(\tau)X(\tau) - X(\tau)B(\tau) + C(\tau)) d\tau \\ = \rho \int_0^t \left(\frac{d(A(\tau)\Xi(\tau))}{d\tau} - \dot{A}(\tau)\Xi(\tau) - \frac{d(\Xi(\tau)B(\tau))}{d\tau} \right. \\ \left. + \Xi(\tau)\dot{B}(\tau) + C(\tau) \right) d\tau \\ = \rho \int_0^t (-\dot{A}(\tau)\Xi(\tau) + \Xi(\tau)\dot{B}(\tau)) d\tau \\ + \rho A(t)\Xi(t) - \rho \Xi(t)B(t) + \rho \int_0^t C(\tau) d\tau \\ = \mathcal{E} - \sigma \mathcal{F}(A(t)X(t) - X(t)B(t) + C(t)) \end{aligned} \quad (8)$$

that is, the following IRNN model can be obtained:

$$\begin{aligned} \rho A(t)\Xi(t) - \rho \Xi(t)B(t) \\ = \mathcal{E} - \sigma \mathcal{F}(A(t)X(t) - X(t)B(t) + C(t)) \\ + \rho \int_0^t (\dot{A}(\tau)\Xi(\tau) - \Xi(\tau)\dot{B}(\tau)) d\tau - \rho \int_0^t C(\tau) d\tau. \end{aligned} \quad (9)$$

The above matrix-type implicit equation is reshaped to be the vector-type one by taking advantage of the Kronecker product [32]. The vector-type IRNN model is

$$\begin{aligned} \rho M(t)\vec{x}(t) = \vec{\epsilon} - \sigma \mathcal{F}(M(t)\vec{x}(t) + \vec{c}(t)) \\ + \rho \int_0^t \dot{M}(\tau)\vec{x}(\tau) d\tau - \rho \int_0^t \vec{c}(\tau) d\tau \end{aligned} \quad (10)$$

where $M(t) = I_n \otimes A(t) - B^T(t) \otimes I_m$, $\vec{x}(t) = \text{VEC}(\Xi(t))$, $\vec{x}(t) = \text{VEC}(X(t))$, $\vec{c}(t) = \text{VEC}(C(t))$, and $\vec{\epsilon} = \text{VEC}(\mathcal{E})$. The large matrix $\mathcal{S} \otimes \mathcal{G}$ with the Kronecker product operator \otimes is formulated as

$$\mathcal{S} \otimes \mathcal{G} := \begin{bmatrix} s_{11}\mathcal{G} & \cdots & s_{1q}\mathcal{G} \\ \vdots & \ddots & \vdots \\ s_{p1}\mathcal{G} & \cdots & s_{pq}\mathcal{G} \end{bmatrix} \in \mathbb{R}^{pw \times qv} \quad (11)$$

where $\mathcal{S} \in \mathbb{R}^{p \times q}$ and $\mathcal{G} \in \mathbb{R}^{w \times v}$. Matrices $I_n \in \mathbb{R}^{n \times n}$ and $I_m \in \mathbb{R}^{m \times m}$ denote the identity matrices and $\text{VEC}(\cdot)$ is the vectorization operation by putting all the column vectors of the input matrix into one reconstructed column vectors together.

According to the above discussion, the matrix-type IRNN formulation (9) and the vector-type equation (10) are obtained. The block diagram of the IRNN (9) for solving the Sylvester equation is presented in Fig. 1. The symbols Σ and \int in Fig. 1 denote the summation and integral operator, respectively. The state solution $X^*(t)$ of the IRNN will be obtained through a differential operation d/dt . In addition, the block diagram is beneficial to the network implementation and the IRNN can be realized by following the mentioned design idea.

III. CONVERGENCE ANALYSIS

Following the IRNN design formulation in Section II, the convergence properties are presented in this section.

Theorem 1: Consider the time-varying Sylvester equation (1) with time-varying matrices $A(t)$, $B(t)$, and $C(t)$. If

the unique solution of the Sylvester equation exists and an odd-monotonically increasing activation function $\mathcal{F}(\cdot)$ is used, the state matrix $X(t)$ with the random initial state X_0 of the IRNN (9) can globally converge to the time-varying theoretical solution $X^*(t)$ of the Sylvester equation (1) asymptotically.

Proof: As is shown in Section II, (5) is equivalent to the implicit equation (9). In addition, the state solution $X(t)$ (i.e., the output of the IRNN) will converge to the unique theoretical solution $X^*(t)$ if the error function $E(t)$ converges to zero. Therefore, the convergence problem of the IRNN (9) is equivalent to the convergence problem of (5). To discuss this conveniently, the scalar type of (5) is

$$\rho \int_0^t e_{ij}(\tau) d\tau = \varepsilon_{ij} - \sigma f(e_{ij}(t)) \quad (12)$$

where $i \in \{1, 2, \dots, m\}$, $j \in \{1, 2, \dots, n\}$, e_{ij} and ε_{ij} are the (i, j) th elements of the error function $E(t)$ and initial condition E_0 . $f(\cdot)$ is the element of activation function $\mathcal{F}(\cdot)$. According to the calculus and the solving process of the integral function, $w_{ij}(t) = \int_0^t e_{ij}(\tau) d\tau$ is defined and $\dot{w}_{ij}(t) = e_{ij}(t)$. The scalar-type equation (12) is rewritten as

$$\rho w_{ij}(t) = \varepsilon_{ij} - \sigma f(\dot{w}_{ij}(t)). \quad (13)$$

For proving the global convergence property of the IRNN, the Lyapunov function candidate is defined as $v_{ij}(t) = (\rho w_{ij}(t) - \varepsilon_{ij})^2/2$, $\forall i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\}$. The time derivative of the Lyapunov function candidate is $\dot{v}_{ij}(t) = (\rho w_{ij}(t) - \varepsilon_{ij})\rho \dot{w}_{ij}(t)$. According to the definition of $w_{ij}(t)$ and (13), the time derivative $\dot{v}_{ij}(t)$ is converted to the following formula:

$$\dot{v}_{ij}(t) = (\rho w_{ij}(t) - \varepsilon_{ij})\rho \dot{w}_{ij}(t) = -\sigma \rho f(e_{ij}(t))e_{ij}(t). \quad (14)$$

Since each activation subfunction $f(\cdot)$ is odd-monotonically increasing, $f(e_{ij}(t))e_{ij}(t) > 0$ for any $e_{ij}(t) \neq 0$ and $f(e_{ij}(t))e_{ij}(t) = 0$ if and only if $e_{ij}(t) = 0$. It means that $\dot{v}_{ij}(t) = -\sigma \rho f(e_{ij}(t))e_{ij}(t) < 0$ for any $t > 0$, where σ and ρ are the positive real numbers. The Lyapunov function $v_{ij}(t)$ will approach to zero and $\rho w_{ij}(t)$ will converge to ε_{ij} if $t \rightarrow +\infty$. In addition, $f(\dot{w}_{ij}(t)) = f(e_{ij}(t)) = 0$ when $e_{ij}(+\infty) = 0$, that is, the scalar-type error function $e_{ij}(t)$ will converge to 0 when $t \rightarrow +\infty$ for any $i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\}$. So that, $X(t)$ will converge to the unique theoretical solution $X^*(t)$ of the Sylvester equation (1) asymptotically.

To summarize, the proof of the global asymptotical convergence theorem (Theorem 1) is completed. ■

Next, the convergence properties of the IRNN with different activation functions, that is, linear type (L-AF), bipolar-sigmoid type (BS-AF), and power type (P-AF) are proved and analyzed. Since the properties of the IRNN with different activation functions are different, the proofs of the theorems about convergence are divided into three parts.

Theorem 2: Given time-varying matrices $A(t)$, $B(t)$, and $C(t)$ of (1), if a linear activation function array is used, then the state matrix of the IRNN (9), starting from any initial random state X_0 , globally converges to the unique time-varying theoretical solution $X^*(t)$. In addition, the convergence rate of the IRNN (9) is exponential convergence with the convergence rate ρ/σ .

Proof: To discuss computation accuracy, the error based on the Frobenius norm is considered and shown as follows:

$$\|X(t) - X^*(t)\|_F = \|\tilde{X}(t)\|_F = \|\tilde{x}(t)\|_2 \quad (15)$$

where $\tilde{X}(t) = X(t) - X^*(t)$ and $\tilde{x}(t) = \text{VEC}(\tilde{X}(t)) = \vec{x}(t) - \vec{x}^*(t)$. $\|\tilde{X}(t)\|_F \rightarrow 0$ is equivalent to $\|X(t) - X^*(t)\|_F \rightarrow 0$. It means that $X(t)$ will converge to $X^*(t)$ if $\|\tilde{X}(t)\|_F \rightarrow 0$. According to the Kronecker product, the error function (2) can be rewritten as $\text{VEC}(E(t)) = \vec{e}(t) = M(t)\vec{x}(t) + \vec{c}(t) = M(t)(\vec{x}(t) - \vec{x}^*(t)) = M(t)\tilde{x}(t)$, where $M(t)\vec{x}^*(t) + \vec{c}(t) = 0$. Matrix $M(t)$ is invertible if the Sylvester equation satisfies the regularity condition in [32]. Besides, $\|M^{-1}(t)\|_F = \sqrt{\|\text{SVD}(M^{-1}(t))\|_2^2} = \varphi > 0$, where $\text{SVD}(\cdot)$ is a vector containing all the singular values of a matrix. According to the theory of matrices, (15) satisfies

$$\begin{aligned} \|\tilde{x}(t)\|_2 &= \|M^{-1}(t)\vec{e}(t)\|_2 \leq \|M^{-1}(t)\|_F \|\vec{e}(t)\|_2 \\ &= \|M^{-1}(t)\|_F \|E(t)\|_F = \varphi \sqrt{\sum_{i=1}^m \sum_{j=1}^n e_{ij}^2(t)}. \end{aligned} \quad (16)$$

For the L-AF $f(u) = u$, where $u \in \mathbb{R}$, (13) can be formulated as

$$\rho w_{ij}(t) = \varepsilon_{ij} - \sigma \dot{w}_{ij}(t). \quad (17)$$

First, the initial condition is discussed. When $t = 0$, the integral item $w_{ij}(0) = \int_0^0 e_{ij}(\tau) d\tau = 0$ and $\varepsilon_{ij} - \sigma \dot{w}_{ij}(0) = \varepsilon_{ij} - \sigma e_{ij}(0) = 0$, that is, $e_{ij}(0) = \varepsilon_{ij}/\sigma$.

Second, according to the calculus and the solving process of the differential equation, the solution of (17) with L-AF is

$$w_{ij}(t) = -\frac{\varepsilon_{ij}}{\rho} \exp\left(-\frac{\rho}{\sigma}t\right) + \frac{\varepsilon_{ij}}{\rho} = \int_0^t e_{ij}(\tau) d\tau. \quad (18)$$

The time derivative of the $w_{ij}(t)$ is

$$\dot{w}_{ij}(t) = \frac{\varepsilon_{ij}}{\sigma} \exp\left(-\frac{\rho}{\sigma}t\right) = e_{ij}(t). \quad (19)$$

Substituting (19) into inequality (16), we have

$$\|\tilde{X}(t)\|_F \leq \varphi \sqrt{\sum_{i=1}^m \sum_{j=1}^n \left(\frac{\varepsilon_{ij}}{\sigma} \exp\left(-\frac{\rho}{\sigma}t\right)\right)^2} = \varphi \kappa_1 \exp\left(-\frac{\rho}{\sigma}t\right) \quad (20)$$

where $\kappa_1 = \sqrt{\sum_{i=1}^m \sum_{j=1}^n \varepsilon_{ij}^2 / \sigma^2} = \sqrt{\sum_{i=1}^m \sum_{j=1}^n e_{ij}^2(0)} = \|E(0)\|_F$. It implies that the solving process via using the IRNN with L-AF possesses exponential convergence. The convergence rate of the IRNN is ρ/σ . To obtain the faster speed, ρ should be set as large as possible and σ needs to set smaller.

To summarize, the proof of the convergence theorem (Theorem 2) of the IRNN with a linear activation function is complete. ■

Theorem 3: Given time-varying matrices $A(t)$, $B(t)$, and $C(t)$ of (1), the state matrix of the IRNN (9) globally converges to the theoretical solution $X^*(t)$ if a bipolar-sigmoid activation function array $f(u) = (1 - \exp(-\xi u))/(1 + \exp(-\xi u))$ ($\xi \geq 1 \in \mathbb{Z}$) is used. Three convergence properties are listed as follows.

- 1) The IRNN (9) possesses convergence rate $\rho\varrho_{\max}(t)/2\sigma\xi$, where $\varrho_{\max}(t) = \max_{1 \leq i \leq m, 1 \leq j \leq n} \varrho_{ij}(t)$ is the maximum value of $\varrho_{ij}(t)$ and $\varrho_{ij}(t)$ is a time-varying parameter defined between $(\exp(-\xi e_{ij}(0)) + 1)^2/\exp(-\xi e_{ij}(0))$ and $\lim_{t \rightarrow \infty} (\exp(-\xi e_{ij}(t)) + 1)^2/\exp(-\xi e_{ij}(t))$.
- 2) The minimum value of the convergence rate of the IRNN (9) is $2\rho/\sigma\xi$ when $t \rightarrow +\infty$.
- 3) There exists a threshold value \mathfrak{M} satisfying that if $e_{ij}(t) > \mathfrak{M} \forall i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\}$, the convergence speed with BS-AF is faster than that of the IRNN with L-AF.

Proof: For the bipolar-sigmoid activation function $f(u) = (1 - \exp(-\xi u))/(1 + \exp(-\xi u))$, where $\xi \geq 1 \in \mathbb{Z}$ and $u \in \mathbb{R}$ are scalar parameters, (13) is reformulated as

$$\rho w_{ij}(t) = \varepsilon_{ij} - \sigma \frac{1 - \exp(-\xi \dot{w}_{ij}(t))}{1 + \exp(-\xi \dot{w}_{ij}(t))} \quad (21)$$

where $\varepsilon_{ij} = \sigma(1 - \exp(-\xi e_{ij}(0)))/(1 + \exp(-\xi e_{ij}(0)))$ and the initial state is $e_{ij}(0) = -\ln((\sigma - \varepsilon_{ij})/(\sigma + \varepsilon_{ij}))/\xi$. To solve the differential equation (21), its derivation is

$$\begin{aligned} \frac{d(\rho w_{ij}(t))}{dt} &= \rho e_{ij}(t) = \frac{d\left(\varepsilon_{ij} - \sigma \frac{1 - \exp(-\xi \dot{w}_{ij}(t))}{1 + \exp(-\xi \dot{w}_{ij}(t))}\right)}{dt} \\ &= -2\sigma\xi \left(\frac{\exp(-\xi e_{ij}(t))}{(1 + \exp(-\xi e_{ij}(t)))^2} \right) \frac{de_{ij}(t)}{dt}. \end{aligned} \quad (22)$$

The differential from of (22) is

$$dt = \frac{-2\sigma\xi}{\rho} \left(\frac{\exp(-\xi e_{ij}(t))}{(1 + \exp(-\xi e_{ij}(t)))^2 e_{ij}(t)} \right) d(e_{ij}(t)). \quad (23)$$

Define $l_{ij}(t) = \exp(-\xi e_{ij}(t)) > 0$, we have

$$dt = \frac{-2\sigma\xi}{\rho} \left(\frac{1}{(1 + l_{ij}(t))^2 \ln(l_{ij}(t))} \right) dl_{ij}(t). \quad (24)$$

Calculating the integral of above equation, we have

$$\begin{aligned} \int dt &= t = \int \frac{-2\sigma\xi}{\rho} \left(\frac{1}{(1 + l_{ij}(t))^2 \ln(l_{ij}(t))} \right) dl_{ij}(t) \\ &= \frac{-2\sigma\xi}{\rho} \int \left(\frac{1}{(1 + l_{ij}(t))^2 \ln(l_{ij}(t))} \right) dl_{ij}(t). \end{aligned} \quad (25)$$

Because of the difficulty for calculating the integral of the right-hand side, we assume that the integral function of $1/((1 + l_{ij}(t))^2 \ln(l_{ij}(t)))$ is $\mathcal{V}(l_{ij}(t))$. The aim of the following discussion is to find the upper bound of $\mathcal{V}(l_{ij}(t))$. Since $\ln(l_{ij}(t)) > 0$ when $l_{ij}(t) > 1$ and $\ln(l_{ij}(t)) \leq 0$ when $l_{ij}(t) \leq 1$, the following discussion will be divided into two parts. Consider the integrand function $1/(1 + x)^2 \ln(x)$ and its integral function $\mathcal{V}(x) = \int(1/(1 + x)^2 \ln(x))$.

1) When $x \in [\theta, \eta]$ with $0 < \theta < \eta \leq 1$, $\ln(x) < 0$. According to the Newton–Leibniz formulas, we have

$$\int_{\theta}^{\eta} \left(\frac{1}{(1 + x)^2 \ln(x)} \right) dx = \mathcal{V}(\eta) - \mathcal{V}(\theta) \quad (26)$$

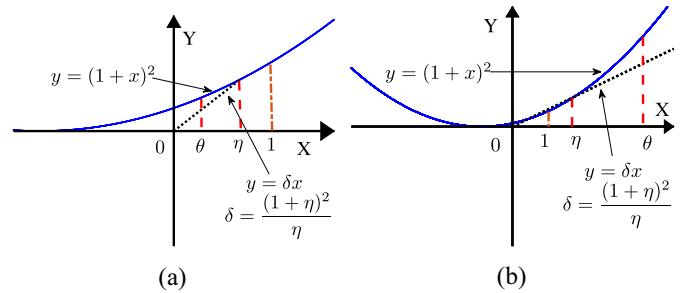


Fig. 2. Function curves $y = (1 + x)^2$ and $y = \delta x$ with $\delta = (1 + \eta)^2/\eta$ in two cases. (a) $0 < \theta < x < \eta \leq 1$. (b) $1 < \eta < x < \theta$.

where $\mathcal{V}(\theta)$ is a scalar value when $x = \theta$. By reformulating (26), the integral function of $(1/(1 + x)^2 \ln(x))$ is

$$\mathcal{V}(\eta) = \int_{\theta}^{\eta} \left(\frac{1}{(1 + x)^2 \ln(x)} \right) dx + \mathcal{V}(\theta). \quad (27)$$

The integrand function $1/((1 + x)^2 \ln(x)) < 0$ when $x \in [\theta, \eta]$ with $0 < \theta < \eta \leq 1$. It means that the integral of the integrand function is smaller than 0. Consider the denominator product factor $(1 + x)^2$ of the integrand function. There exists a parameter $\delta = (1 + \eta)^2/\eta$ satisfying $(1 + x)^2 \geq \delta x$ when $x \in [\theta, \eta]$ [Fig. 2 (a)]. Then, the following inequality is obtained:

$$\begin{aligned} \mathcal{V}(\eta) &= \int_{\theta}^{\eta} \left(\frac{1}{(1 + x)^2 \ln(x)} \right) dx + \mathcal{V}(\theta) \\ &\geq \frac{1}{\delta} \int_{\theta}^{\eta} \left(\frac{1}{x \ln(x)} \right) dx + \mathcal{V}(\theta) = \frac{1}{\delta} (\mathcal{W}(\eta) - \mathcal{W}(\theta)) \\ &\quad + \mathcal{V}(\theta) \\ &= \frac{1}{\delta} (\ln(|\ln(\eta)|) + \Gamma - \ln(|\ln(\theta)|) - \Gamma) + \mathcal{V}(\theta) \\ &= \frac{1}{\delta} (\ln(|\ln(\eta)|) + \mathcal{U}(\theta)) \end{aligned} \quad (28)$$

where $\mathcal{W}(x) = \ln(|\ln(x)|) + \Gamma$ is the integral function of $1/(x \ln(x))$, Γ is a constant value, and $\mathcal{U}(\theta) = -\ln(|\ln(\theta)|) + \delta \mathcal{V}(\theta)$. For each parameter $x \in (0, 1]$ (by replacing η in (28) with x), we have

$$\mathcal{V}(x) \geq \frac{1}{\delta} (\ln(|\ln(x)|) + \mathcal{U}(\theta)), \delta = (1 + x)^2/x. \quad (29)$$

2) When $x \in [\eta, \theta]$ with $1 < \eta < \theta$, $\ln(x) > 0$. According to the Newton–Leibniz formulas, we have

$$\int_{\eta}^{\theta} \frac{1}{(1 + x)^2 \ln(x)} dx = \mathcal{V}(\theta) - \mathcal{V}(\eta). \quad (30)$$

By reformulating (30), we have

$$\mathcal{V}(\eta) = \int_{\eta}^{\theta} \left(-\frac{1}{(1 + x)^2 \ln(x)} \right) dx + \mathcal{V}(\theta). \quad (31)$$

Similar to situation 1), there exists a parameter $\delta = (1 + \eta)^2/\eta$ satisfying $(1 + x)^2 \geq \delta x$ when $x \in [\eta, \theta]$ [see Fig. 2(b)]. The following inequality is obtained:

$$\begin{aligned} \mathcal{V}(\eta) &\geq \frac{1}{\delta} \int_{\eta}^{\theta} \left(-\frac{1}{x \ln(x)} \right) dx + \mathcal{V}(\theta) \\ &= \frac{1}{\delta} (\mathcal{W}(\eta) - \mathcal{W}(\theta)) + \mathcal{V}(\theta) = \frac{1}{\delta} (\ln(|\ln(\eta)|) + \mathcal{U}(\theta)). \end{aligned} \quad (32)$$

For each parameter $x \in (1, +\infty)$, we have

$$\mathcal{V}(x) \geq \frac{1}{\delta}(\ln(|\ln(x)|) + \mathcal{U}(\theta)) \quad (33)$$

where $\delta = (1+x)^2/x$. The upper bound of $\mathcal{V}(x)$ in 2) is similar to that of the integral function $\mathcal{V}(x)$ in 1).

To summarize, the general upper bound of the integral function $\mathcal{V}(x)$ is obtained. Therefore, (25) is rewritten as

$$\begin{aligned} -\rho t &= \int \left(\frac{1}{(1+l_{ij}(t))^2 \ln(l_{ij}(t))} \right) dl_{ij}(t) \\ &= \mathcal{V}(l_{ij}(t)) \geq \frac{1}{\varrho_{ij}(t)} (\ln(|\ln(l_{ij}(t))|) + \mathcal{U}(l_{ij}(0))) \\ &= \frac{1}{\varrho_{ij}(t)} (\ln(|\ln(l_{ij}(t))|) + \ln(h)) \end{aligned} \quad (34)$$

where $\varrho_{ij}(t) := (1+l_{ij}(t))^2/l_{ij}(t) = l_{ij}(t) + 2 + 1/l_{ij}(t)$ is a time-varying parameter and $h = \exp(\mathcal{U}(l_{ij}(0)))$, that is, the solution of (34) is

$$|e_{ij}(t)| \leq \frac{h}{\xi} \exp\left(-\frac{\rho \varrho_{ij}(t)}{2\sigma\xi} t\right). \quad (35)$$

The convergence rate of the IRNN of the (i, j) th element $e_{ij}(t)$ is $\rho \varrho_{ij}(t)/2\sigma\xi$. Since $l_{ij}(t) = \exp(-\xi e_{ij}(t))$ and the IRNN possesses global asymptotical convergence property, $\varrho_{ij}(t)$ is defined between $(\exp(-\xi e_{ij}(0)) + 1)^2/\exp(-\xi e_{ij}(0))$ and $\lim_{t \rightarrow \infty} (\exp(-\xi e_{ij}(t)) + 1)^2/\exp(-\xi e_{ij}(t))$. Substituting (35) into inequality (16), we have

$$\begin{aligned} \|\tilde{X}(t)\|_F &\leq \varphi \sqrt{\sum_{i=1}^m \sum_{j=1}^n \left(\frac{h}{\xi} \exp\left(-\frac{\rho \varrho_{ij}(t)}{2\sigma\xi} t\right) \right)^2} \\ &\leq \varphi \kappa_2 \exp\left(-\frac{\rho \varrho_{\max}(t)}{2\sigma\xi} t\right) \end{aligned} \quad (36)$$

where

$$\kappa_2 = \sqrt{\sum_{i=1}^m \sum_{j=1}^n \left(\frac{\exp(\mathcal{U}(l_{ij}(0)))}{\xi} \right)^2} \quad (37)$$

and $\varrho_{\max}(t) = \max_{1 \leq i \leq m, 1 \leq j \leq n} \varrho_{ij}(t)$. The convergence rate of the IRNN is $\rho \varrho_{\max}(t)/2\sigma\xi$. In addition, $\varrho_{ij}(t)$ will achieve the minimum value if $l_{ij}(t) = 1$. It means that the IRNN possesses faster speed when the initial error $|e_{ij}(0)|$ is large. Furthermore, $\varrho_{ij}(t)$ will converge to 4 if $t \rightarrow \infty$, that is

$$|e_{ij}(t)| \leq \frac{h}{\xi} \exp\left(-\frac{2\rho}{\sigma\xi} t\right), \text{ if } t \rightarrow \infty. \quad (38)$$

It implies that the convergence rate of the IRNN with BS-AF is at least $2\rho/\sigma\xi$, which is just a half of L-AF if $\xi = 4$.

Furthermore, (35) is similar to (19) and the comparison about the convergence performance is discussed as below. We assume that the IRNN with L-AF and BS-AF start in the same initial state. The convergence rate of the IRNN with L-AF is ρ/σ . Consider the following inequality:

$$\frac{\rho \varrho_{ij}(t)}{2\sigma\xi} > \frac{\rho}{\sigma}. \quad (39)$$

According to the definition of $\varrho_{ij}(t)$, we have $(1+l_{ij}(t))^2/l_{ij}(t) > 2\xi$, and the solution of the inequality is

$$\begin{cases} l_{ij}(t) \in (0, +\infty), & \text{if } \xi = 1 \text{ or } \xi = 2 \\ l_{ij}(t) \in (0, L_a) \cup (L_b, +\infty), & \text{otherwise} \end{cases}$$

where $L_a = \xi - 1 - \sqrt{\xi^2 - 2\xi}$ and $L_b = \xi - 1 + \sqrt{\xi^2 - 2\xi}$. Since $l_{ij}(t) = \exp(-\xi e_{ij}(t))$, we have

$$\begin{cases} e_{ij}(t) \in (-\infty, +\infty), & \text{if } \xi = 1 \text{ or } \xi = 2 \\ e_{ij}(t) \in (-\infty, \mathfrak{E}_b) \cup (\mathfrak{E}_a, +\infty), & \text{otherwise} \end{cases}$$

where $\mathfrak{E}_b = -\ln(L_b)/\xi$ and $\mathfrak{E}_a = -\ln(L_a)/\xi$. If $e_{ij}(t) < \mathfrak{E}_b$ or $e_{ij}(t) > \mathfrak{E}_a$, $\forall i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\}$, the convergence speed of the IRNN with BS-AF will be faster than that of the IRNN with L-AF. That is to say, there exists a threshold value \mathfrak{M} satisfying that if $e_{ij}(t) > \mathfrak{M} \forall i \in \{1, 2, \dots, m\}, j \in \{1, 2, \dots, n\}$, the convergence speed with BS-AF is faster than that of the IRNN with L-AF.

To summarize, the proof of the convergence theorem (Theorem 3) of the IRNN with BS-AF is complete. ■

Theorem 4: Given time-varying matrices $A(t)$, $B(t)$, and $C(t)$ of (1), $X(t)$ of the IRNN (9) globally converges to the theoretical solution $X^*(t)$ if a power activation function $f(u) = u^\mu$ ($\mu \geq 3$ is an odd number and $x \in \mathbb{R}$) is used. Two convergence properties are listed as follows.

- 1) The IRNN with P-AF possesses finite-time convergence and the finite convergence time of the IRNN is $T_{\max} = \sigma \mu \kappa_3 / \rho(\mu - 1)$.
- 2) The convergence speed of the IRNN with P-AF is larger than that of the IRNN with L-AF in the same computational error value $|e_{ij}(t)| \leq \sqrt[\mu-1]{1/\mu}$.

Proof: 1) For P-AF $f(u) = u^\mu$, where $\mu \geq 3$ is an odd number and $x \in \mathbb{R}$, (13) is rewritten as

$$\rho w_{ij}(t) = \varepsilon_{ij} - \sigma \left(\frac{dw_{ij}(t)}{dt} \right)^\mu \quad (40)$$

where $\varepsilon_{ij} = \sigma e_{ij}^\mu(0)$ and the initial state $e_{ij}(0) = \sqrt[\mu]{\varepsilon_{ij}/\sigma}$. Equation (40) is reformulated as

$$dt = \left(\frac{\varepsilon_{ij} - \rho w_{ij}(t)}{\sigma} \right)^{-\frac{1}{\mu}} d(w_{ij}(t)). \quad (41)$$

The integral of (41) is

$$\begin{aligned} t &= \int dt = \int \left(\frac{\varepsilon_{ij} - \rho w_{ij}(t)}{\sigma} \right)^{-\frac{1}{\mu}} d(w_{ij}(t)) \\ &= -\frac{\sigma\mu}{\rho(\mu-1)} \left(\frac{\varepsilon_{ij} - \rho w_{ij}(t)}{\sigma} \right)^{\frac{\mu-1}{\mu}} + \mathfrak{J} \end{aligned} \quad (42)$$

where $\mathfrak{J} = \sigma\mu e_{ij}^{\mu-1}(0)/\rho(\mu-1)$ is the parameter satisfying the initial condition $t = 0$. The solution of (42) is

$$w_{ij}(t) = \frac{\sigma e_{ij}^\mu(0)}{\rho} - \frac{\sigma}{\rho} \left(-\frac{\rho(\mu-1)}{\sigma\mu} t + e_{ij}^{\mu-1}(0) \right)^{\frac{\mu}{\mu-1}}. \quad (43)$$

The computational error $e_{ij}(t)$ of the (i, j) th element is

$$e_{ij}(t) = -\frac{\sigma\mu}{\rho(\mu-1)} \left(-\frac{\rho(\mu-1)}{\sigma\mu} t + e_{ij}^{\mu-1}(0) \right)^{\frac{1}{\mu-1}}. \quad (44)$$

Three cases should be considered.

- 1) When $-\rho(\mu-1)t/(\sigma\mu) + e_{ij}^{\mu-1}(0) = 0$, we can obtain $t = \sigma\mu e_{ij}^{\mu-1}(0)/\rho(\mu-1) = T_{ij}$.
- 2) When $t < T_{ij}$, since $-\rho(\mu-1)t/(\sigma\mu) + e_{ij}^{\mu-1}(0)$ is a monotone decreasing function about t and, thus, $-\rho(\mu-1)t/(\sigma\mu) + e_{ij}^{\mu-1}(0) > 0$. In this condition, the solution of (44) exists.
- 3) When $t > T_{ij}$, we can obtain $-\rho(\mu-1)t/(\sigma\mu) + e_{ij}^{\mu-1}(0) < 0$. Evidently, there is no solution in the domain of the real number to (44) with odd number $\mu \geq 3$ when $-\rho(\mu-1)t/(\sigma\mu) + e_{ij}^{\mu-1}(0) < 0$. Therefore, the real part of the complex solution of (44) is zero. That is to say, $e_{ij}(t) = 0$ when $t > T_{ij}$ in the domain of the real number.

Substituting (44) into inequality (16), we have

$$\|\tilde{X}(t)\|_F \begin{cases} \leq \frac{\varphi\sqrt{mn}\sigma\mu}{\rho(\mu-1)}(-\kappa_4 t + \kappa_5)^{\frac{1}{\mu-1}}, & \text{if } t \leq T_{\max} \\ = 0, & \text{otherwise} \end{cases} \quad (45)$$

where $\kappa_4 = \rho(\mu-1)/\sigma\mu$, $\kappa_5 = \max_{1 \leq i \leq m, 1 \leq j \leq n} e_{ij}^{\mu-1}(0)$, and $T_{\max} = \sigma\mu\kappa_3/\rho(\mu-1)$. It implies that the finite convergence time T_{\max} is obtained and the finite-time convergence of the IRNN with a power activation function is proved.

2) To discuss the convergence speed of the IRNN with P-AF, the time derivative of $e_{ij}(t)$ is considered.

First, (40) is reformulated as

$$\dot{e}_{Pij}(t) = \dot{w}_{Pij}(t) = \left(-\frac{1}{\sigma} (\rho w_{Pij}(t) - \varepsilon_{Pij}) \right)^{\frac{1}{\mu}} \quad (46)$$

where $e_{Pij}(t)$ denotes the error function $e_{ij}(t)$ when the IRNN uses a P-AF. The time derivative of (46) is

$$\begin{aligned} \dot{e}_{Pij}(t) &= -\frac{\rho}{\mu\sigma} \left(-\frac{1}{\sigma} (\rho w_{Pij}(t) - \varepsilon_{Pij}) \right)^{\frac{1-\mu}{\mu}} e_{Pij}(t) \\ &= -\frac{\rho}{\mu\sigma} e_{Pij}^{1-\mu}(t) e_{Pij}(t) = -\frac{\rho}{\mu\sigma} e_{Pij}^{2-\mu}(t). \end{aligned} \quad (47)$$

Second, the computational error $e_{Lij}(t)$ of the IRNN for solving the Sylvester equation with L-AF (17) is

$$\dot{e}_{Lij}(t) = \dot{w}_{Lij}(t) = -\frac{1}{\sigma} (\rho w_{Lij}(t) - \varepsilon_{Lij}) \quad (48)$$

where $e_{Lij}(t)$ denotes the error function $e_{ij}(t)$ when the IRNN uses an L-AF. The time derivative is

$$\dot{e}_{Lij}(t) = -\frac{\rho}{\sigma} \dot{w}_{Lij}(t) = -\frac{\rho}{\sigma} e_{Lij}(t). \quad (49)$$

Suppose that the computational error $e_{Lij}(t) = e_{Pij}(t) = e_{ij}(t)$, we consider the following inequality:

$$\frac{\rho}{\mu\sigma} e_{ij}^{2-\mu}(t) \geq \frac{\rho}{\sigma} e_{ij}(t). \quad (50)$$

The solution of the above inequality is $|e_{ij}(t)| \leq \sqrt[\mu-1]{1/\mu} < 1$. It means that if $|e_{ij}(t)| \leq \sqrt[\mu-1]{1/\mu} < 1$, the absolute value of $\dot{e}_{Lij}(t)$ is smaller than $\dot{e}_{Pij}(t)$, that is, the convergence speed of the IRNN with P-AF is larger than L-AF in the same computational error value $|e_{Lij}(t)| = |e_{Pij}(t)| \leq \sqrt[\mu-1]{1/\mu} < 1$.

In summary, the proof of the convergence theorem (Theorem 4) of the IRNN with P-AF is complete. ■

Theorem 5: Based on Theorems 3 and 4, a faster reconstructed piecewise activation function (called SP-AF) is obtained

$$f(u) = \begin{cases} u^\mu, & \text{if } u < 1 \\ \frac{1 - \exp(-\xi u)}{1 + \exp(-\xi u)}, & \text{otherwise} \end{cases} \quad (51)$$

with suitable parameters μ and ξ . Given time-varying matrices $A(t)$, $B(t)$, and $C(t)$ of (1), the state matrix of the IRNN (9) globally converges to the theoretical solution $X^*(t)$ if an SP-AF is used. The convergence speed of the activation function (51) is faster than that of the linear activation function. What is more, this activation function possesses finite-time convergence property.

Proof: Consult the proof of Theorems 3 and 4. ■

The proof of the convergence properties of the IRNN (9) with different activation functions are complete. It is worth mentioning that the larger ρ or the smaller σ are used, the faster convergence speed of the IRNN is obtained.

IV. ROBUSTNESS ANALYSIS

In practical applications, some outside interferences and model errors may exist and would lead to model uncertainty. For instance, the high-order residual errors of circuit components or truncating/roundoff errors in digital realization are the common causes for model-implementation errors in neural network realization [33]. The robustness performance of the proposed IRNN is discussed in this section.

The implicit equation of the IRNN with model-implementation errors $\Delta\Phi(t)$ is written as

$$\begin{aligned} &\rho A(t)\mathfrak{X}(t) - \rho \mathfrak{X}(t)B(t) \\ &= \mathcal{E} - \sigma \mathcal{F}(A(t)X(t) - X(t)B(t) + C(t)) + \Delta\Phi(t) \\ &\quad + \rho \int_0^t (\dot{A}(\tau)\mathfrak{X}(\tau) - \mathfrak{X}(\tau)\dot{B}(\tau))d\tau - \rho \int_0^t C(\tau)d\tau \end{aligned} \quad (52)$$

where $\Delta\Phi(t) \in \mathbb{R}^{m \times n}$ is the model-implementation error. Theorem 6 about robustness is proved as follows.

Theorem 6: Consider the perturbed IRNN (P-IRNN) model [e.g., with unknown smooth model-implementation error $\Delta\Phi(t)$] in (52) for solving the Sylvester equation (1). If $\|\Delta\dot{\Phi}(t)\|_F \leq \zeta$ and $\|M^{-1}(t)\|_F \leq \varphi$ for any $t \in [0, \infty)$, then the computational error $\|X(t) - X^*(t)\|_F$ will be bounded by $(mn + \sqrt{mn})\varphi\zeta/2\rho$, where $M(t) = I \otimes A(t) - B^T(t) \otimes I$.

Proof: Matrix-type equation (52) can be transformed to the following vector-type equation:

$$\begin{aligned} \rho M(t)\vec{x}(t) &= \vec{\epsilon} - \sigma \mathcal{F}(M(t)\vec{x}(t) + \vec{c}(t)) + \Delta\vec{\phi}(t) \\ &\quad + \rho \int_0^t \dot{M}(\tau)\vec{x}(\tau)d\tau - \rho \int_0^t \vec{c}(\tau)d\tau \end{aligned} \quad (53)$$

where $\Delta\vec{\phi}(t) = \text{VEC}(\Delta\Phi(t))$. Similar to the derivation process of the IRNN in Section II, the P-IRNN model (53) is

$$\rho \vec{w}(t) = \rho \int_0^t \vec{e}(\tau)d\tau = \vec{\epsilon} - \sigma f(\vec{e}(t)) + \Delta\vec{\phi}(t) \quad (54)$$

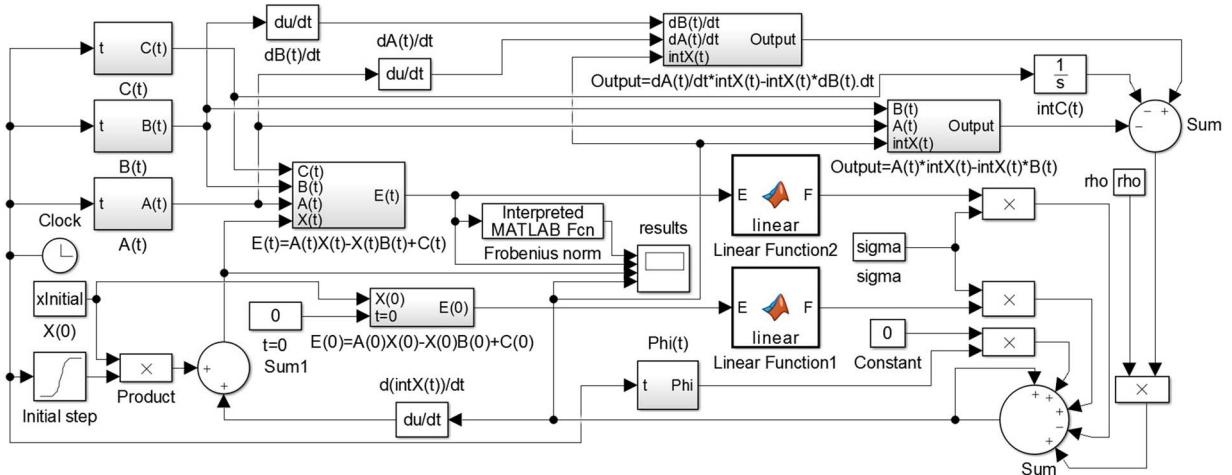


Fig. 3. MATLAB Simulink program of the IRNN model (9) with linear activation function for solving the Sylvester equation.

where $\vec{w}(t) = \int_0^t \vec{e}(\tau) d\tau$ and $\vec{e}(\tau) = \text{VEC}(E(t))$. Lyapunov function candidate is defined as

$$v(t) = \frac{(\rho \vec{w}(t) - \vec{\epsilon} - \Delta \vec{\phi}(t))^T (\rho \vec{w}(t) - \vec{\epsilon} - \Delta \vec{\phi}(t))}{2} \quad (55)$$

where superscript T denotes the transpose of a vector. The time derivative of the Lyapunov function (55) is

$$\begin{aligned} \dot{v}(t) &= (\rho \vec{w}(t) - \vec{\epsilon} - \Delta \vec{\phi}(t))(\rho \dot{\vec{w}}(t) - \Delta \dot{\vec{\phi}}(t)) \\ &= -\sigma f(\vec{e}(t))(\rho \vec{e}(t) - \Delta \dot{\vec{\phi}}(t)) \\ &\leq -\sum_{i=1}^n \sum_{j=1}^n \sigma f(|e_{ij}(t)|) \{ \rho |e_{ij}(t)| - \zeta \} = \mathcal{Q} \end{aligned} \quad (56)$$

where $\|\Delta \dot{\vec{\phi}}(t)\|_2 = \|\Delta \dot{\Phi}(t)\|_F < \zeta$. In the equation of the IRNN with model-implementation errors, two cases exist

$$\begin{cases} \rho |e_{ij}(t)| \geq \zeta, \forall i \in \{1, \dots, m\}, j \in \{1, \dots, n\} \\ \rho |e_{ij}(t)| < \zeta, \exists i \in \{1, \dots, m\}, j \in \{1, \dots, n\}. \end{cases}$$

According to the Lyapunov stability theory, the following two properties of the P-IRNN (52) [with model-implementation error $\Delta \dot{\Phi}(t)$] can be obtained.

1) For the first case, the time derivative function (56) $\dot{v}(t) \leq \mathcal{Q} \leq 0$, that is, the Lyapunov function (55) decreases. When $\dot{v}(t) < 0$, the residual error $\vec{e}(t)$ would decrease to zero if the odd-monotonically increasing activation function $f(\cdot)$ is used, and when $\dot{v}(t) = 0$, the residual error $\vec{e}(t)$ would converge to a steady value.

2) For the second case, the following discussion is divided into two situations due to nondeterminacy of the sign of \mathcal{Q} . If $\mathcal{Q} < 0$, the performance of the IRNN is similar to case 1), that is, $\dot{v}(t) \leq \mathcal{Q} \leq 0$. Otherwise, $\mathcal{Q} > 0$ and the sign of $\dot{v}(t)$ is unknown. In this situation, we can discuss directly the residual errors. Inspired by [34], the upper bound of the error function $|e_{ij}(t)|$ of the P-IRNN is

$$|e_{ij}(t)| \leq \frac{(1 + \sqrt{mn})\zeta}{2\rho}. \quad (57)$$

Substituting inequality (57) into inequality (16), the upper bound of the computational error $\tilde{X}(t) = X(t) - X^*(t)$ is

$$\|\tilde{X}(t)\|_F \leq \varphi \sqrt{\sum_{i=1}^m \sum_{j=1}^n \left(\frac{(1 + \sqrt{mn})\zeta}{2\rho} \right)^2} = \frac{(mn + \sqrt{mn})\varphi\zeta}{2\rho}. \quad (58)$$

According to Theorems 2–4, ρ should be set large and σ should be set small. It means that large ρ will lead to small upper bound. For obtaining the better convergence and robust property, ρ and σ should be set as suitable values.

To summarize, the computational error $\|\tilde{X}(t)\|_F$ would converge toward 0 or a steady value in the first case. Otherwise, the upper bound of the computational error $\|\tilde{X}(t)\|_F$ is $(mn + \sqrt{mn})\varphi\zeta/2\rho$.

Thus, the robustness theorem is proved. ■

V. DISCRETE IRNN AND SOFTWARE REALIZATION

The realization of the IRNN is discussed in this section. First, as for the continuous RNN, it is more suitable for hardware realization such as electric circuits. The IRNN can be realized by MATLAB Simulink program (Fig. 3). If we want to use this IRNN, we can import the matrices data into the Simulink module and obtain the output solution. What is more, we can generate the code of the IRNN by Simulink. Second, as for software programming by using the computer, discretization of the continuous model is needed. Some iteration methods for solving the integral equation (9) (a Volterra integral equation) can be used. Third, the IRNN can be realized by using the ZNN method. Defining a new error function based on the IRNN design formula (5), we have

$$\mathbb{E}(t) = \rho \int_0^t E(\tau) d\tau + \sigma \mathcal{F}(E(t)) - \mathcal{E}. \quad (59)$$

Consider the ZNN formula $\dot{\mathbb{E}}(t) = -\gamma \mathbb{F}(\mathbb{E}(t))$, we have

$$\rho E(t) + \sigma \mathcal{F}(E(t)) \dot{E}(t) = -\gamma \mathbb{F}(\mathbb{E}(t)) \quad (60)$$

where $\mathbb{F}(\cdot)$ is the activation function of the ZNN method, $\mathcal{F}(E(t)) = \partial \mathcal{F}(E(t))/\partial E(t)$. Furthermore, (60) can be

rewritten as

$$\dot{E}(t) = \frac{-\gamma \mathbb{F}(\rho \int_0^t E(\tau) d\tau + \sigma \mathcal{F}(E(t)) - \mathcal{E}) - \rho E(t)}{\sigma \mathcal{F}(E(t))}. \quad (61)$$

Based on the vectorization operation and Kronecker product [32], (61) can be written as

$$\begin{aligned} \dot{\vec{x}}(t) &= \mathcal{R}(t) \left(-\dot{M}(t)\vec{x}(t) - \dot{\vec{c}}(t) \right. \\ &\quad \left. - \frac{\gamma \mathbb{F}(\rho \int_0^t \vec{e}(\tau) d\tau + \sigma \mathcal{F}(\vec{e}(t)) - \vec{\varepsilon}) + \rho \vec{e}(t)}{\sigma \mathcal{F}(\vec{e}(t))} \right) \end{aligned} \quad (62)$$

where $\mathcal{R}(t) = M^{-1}(t)$ is the inverse matrix of $M(t)$. According to [13], $\mathcal{R}(t)$ can be approximated by the quasi-Newton BFGS method. By using some numerical differential method, such as the Euler method or Runge–Kutta method, the discrete realization of (62) can be obtained. For example, when the Euler method is used, we have

$$\begin{aligned} \vec{x}_{k+1} &= \vec{x}_k + \mathcal{R}_k \Delta \left(-\dot{M}_k \vec{x}_k - \dot{\vec{c}}_k - \mathcal{Z}_k \right) \\ \mathcal{Z}_k &= \frac{\gamma \mathbb{F}(\rho \sum_{i=1}^k \vec{e}_i \Delta + \sigma \mathcal{F}(\vec{e}_k) - \vec{\varepsilon}) + \rho \vec{e}_k}{\sigma \mathcal{F}(\vec{e}_k)} \end{aligned} \quad (63)$$

where Δ is the step length, the subscript k means the k -step data of the matrices and vectors. In computer, we can obtain the solution \vec{x}_{k+1} by using the information from k step, and (63) can be realized.

According to the Dahlquist convergence theorem [35], if and only if a linear multistep method (LMM) is both consistent and zero-stable, the LMM is convergent.

Definition 1 (Zero-Stability (Root Condition) [35]): If all the roots ς of the characteristic polynomial $\mathcal{P}(\cdot)$ of an LMM are within ($\varsigma < 1$) or on the boundary of the unit circle ($\varsigma = 1$ and should be simple root), such an LMM is zero-stable.

Definition 2 (Consistency [35]): An LMM is consistent if there exists parameter p , such that the linear difference operator \mathcal{L} of the LMM is equal to $\mathcal{O}(\Delta^{p+1})$, that is

$$\mathcal{L} = \mathcal{O}(\Delta^{p+1}) \quad (64)$$

where $\mathcal{O}(\cdot)$ is the standard notation for the index of the remainder, Δ is the step length, and p is a positive integer.

Theorem 7: The discrete equation (63) is both zero-stable and consistent with $\mathcal{O}(\Delta^2)$, that is, the convergence of the discrete IRNN model (63) can be guaranteed.

Proof: As for the discrete equation (63), the characteristic polynomial is $\mathcal{P}(\varsigma) = \varsigma - 1$, that is, (63) is zero-stable.

Based on (61) and the Euler method, we have

$$\frac{\vec{e}_{k+1} - \vec{e}_k}{\Delta} + \mathcal{O}(\Delta) = -\mathcal{Z}_k. \quad (65)$$

Furthermore, the above equation can be rewritten as

$$\vec{e}_{k+1} = -\mathcal{Z}_k \Delta + \vec{e}_k - \mathcal{O}(\Delta^2). \quad (66)$$

According to the Talyor series expansion [36], the k -step error function \vec{e}_{k+1} can be expressed as

$$\begin{aligned} \vec{e}_{k+1} &= \vec{e}_k + \frac{\partial \vec{e}_k}{\partial \vec{x}_k} (\vec{x}_{k+1} - \vec{x}_k) + \frac{\partial \vec{e}_k}{\partial t_k} \Delta + \mathcal{O}(\Delta^2) \\ &= \vec{e}_k + M_k (\vec{x}_{k+1} - \vec{x}_k) + (\dot{M}_k \vec{x}_k + \dot{\vec{c}}_k) \Delta + \mathcal{O}(\Delta^2) \end{aligned} \quad (67)$$

where $\mathcal{O}(\Delta^2)$ contains the second-order and higher-order terms. Substituting (67) into (66), we have

$$\vec{x}_{k+1} = -\mathcal{R}_k \Delta (\mathcal{Z}_k + M_k \vec{x}_k + \dot{\vec{c}}_k) + \vec{x}_k - \mathcal{O}(\Delta^2). \quad (68)$$

The difference operator \mathcal{L} is

$$\mathcal{L} = \vec{x}_{k+1} - \vec{x}_k + \mathcal{R}_k \Delta (\dot{M}_k \vec{x}_k + \dot{\vec{c}}_k + \mathcal{Z}_k) = \mathcal{O}(\Delta^2). \quad (69)$$

That is, the discrete equation (63) is consistent.

In conclusion, according to the Dahlquist theorem [35], the convergence of the discrete IRNN (63) is guaranteed. ■

VI. COMPUTER SIMULATION VERIFICATIONS

To verify the effectiveness of the IRNN (9), a time-varying Sylvester equation with the following time-varying coefficient matrices $A(t)$, $B(t)$, and $C(t)$ is considered, where

$$\begin{aligned} A(t) &= \begin{bmatrix} \frac{S^3 + 10S + 2C^3}{20} & \frac{S^2 C + 10C - 2SC^2}{20} \\ \frac{S^2 C - 10C - 2SC^2}{20} & \frac{SC^2 + 10S - 2S^2 C}{20} \end{bmatrix} \\ B(t) &= \begin{bmatrix} \frac{S}{10} & 0 \\ 0 & \frac{C}{5} \end{bmatrix}, C(t) = \begin{bmatrix} \frac{S^2}{10} - 1 & \frac{C^2}{5} \\ \frac{SC}{10} & \frac{SC}{5} - 1 \end{bmatrix} \end{aligned}$$

where $S = \sin t$ and $C = \cos t$. The theoretical solution of the Sylvester equation with the above matrices is shown as follows:

$$X^*(t) = \begin{bmatrix} 2 \sin t & -2 \cos t \\ 2 \cos t & 2 \sin t \end{bmatrix}.$$

The state solution $X(t)$ of the proposed IRNN (9) is

$$X(t) = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}, \text{ with } X(0) = \begin{bmatrix} x_a & x_b \\ x_c & x_d \end{bmatrix}$$

where x_{ij} is the (i, j) th element of the solution $X(t)$ and the initial elements x_a , x_b , x_c , and x_d are generated randomly in the computer simulation. In addition, all the elements of the initial state are within $[-10, +10]$. Supposed that the solving process of the Sylvester equation (1) by using the IRNN (9) is complete if the norm-based error $\|X(t) - X^*(t)\|_F$ is smaller than 0.1% of the maximal random absolute value of the initial state, that is, $\|X(t) - X^*(t)\|_F \leq 0.1\% \times \sqrt{4 \times 10^2} = 0.02$. The MATLAB Simulink program of the IRNN model with L-AF for solving the Sylvester equation is shown in Fig. 3.

A. Convergence of the IRNN

First, for illustrating the global convergence performance of the IRNN (9) with different design parameters ρ and σ , the simulation results are shown in Figs. 4 and 5. To verify the effectiveness of the IRNN (9), the experiment is repeated ten times. The total convergence time of one situation is defined as

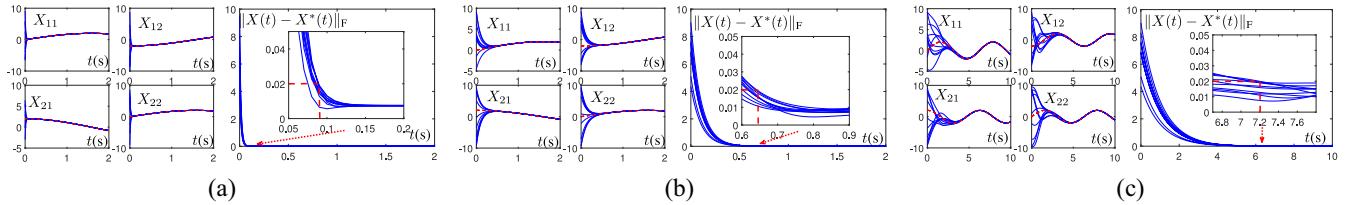


Fig. 4. Online solutions of the Sylvester equation (1) and the computational errors $\|X(t) - X^*(t)\|_F$ of the IRNN (9) with different σ . (a) $\sigma = 0.01, \rho = 1$. (b) $\sigma = 0.1, \rho = 1$. (c) $\sigma = 1, \rho = 1$.

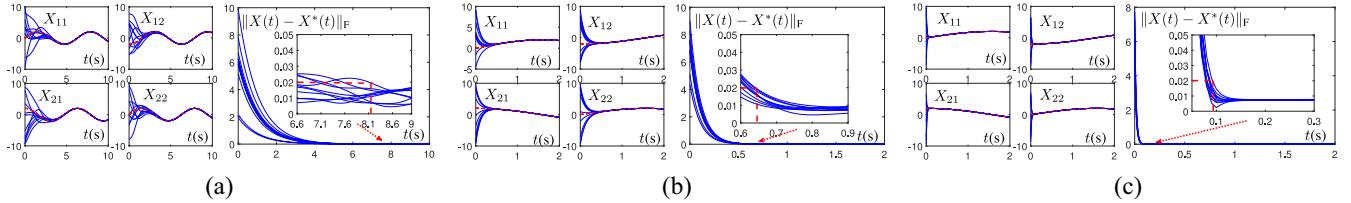


Fig. 5. Online solutions of the Sylvester equation (1) and the computational errors $\|X(t) - X^*(t)\|_F$ of the IRNN (9) with different ρ . (a) $\sigma = 0.1, \rho = 0.1$. (b) $\sigma = 0.1, \rho = 1$. (c) $\sigma = 0.1, \rho = 10$.

the time that all the errors are smaller than 0.02. The figures in the left-hand side of each subgraph present the online solutions of the IRNN (9). The computational errors $\|X(t) - X^*(t)\|_F$ are illustrated in the right-hand side of each subfigure. Blue solid lines and red dashed lines in Figs. 4 and 5 denote the convergence results of the IRNN and the theoretical solution, respectively. As is shown in Fig. 4(a), all the errors are smaller than 0.02 after $t > 0.089$ s, that is, the convergence process of the IRNN (9) with design parameters $\sigma = 0.01$ and $\rho = 1$ is complete and the convergence time is 0.089 s. The larger σ we set, the longer time the IRNN (9) possesses. When $\sigma = 0.1$ and $\rho = 1$, the convergence time is 0.641 s [Fig. 4(b)], which is nearly eight times larger than $\sigma = 0.01$. Furthermore, $\|X(t) - X^*(t)\|_F$ of the IRNN with $\sigma = 1$ and $\rho = 1$ approach to the threshold value 0.02 when $t = 7.185$ s. Consider the simulation results of the IRNN (9) by changing the design parameter σ . From Fig. 5, we find that the convergence time of the IRNN with the fixed σ will decrease if ρ increases (8.097 s when $\rho = 0.1$, 0.641 s when $\rho = 1$, and 0.092 s when $\rho = 10$). Theorem 1 and the convergence of the IRNN with different design parameters σ and ρ have been verified. More detailed convergence time of the IRNN with different design parameters is listed in Table I. According to the simulation results listed in Table I, the convergence time will be smaller and the convergence speed will be faster if ρ/σ is set larger.

Second, the results in Fig. 6 have verified that different activation functions will lead to different performances, and the IRNN with the designed SP-AF can obtain the best convergence among the L-AF, BS-AF, P-AF, and SP-AF. The convergence design parameters ρ is set to 10 and σ is set to 0.1. Fig. 6(a) is the convergence results by using an L-AF. Exponential convergence performance is achieved and the convergence time is $t = 0.067$ s. If a BS-AF is used [Fig. 6(b)], the speed is much faster than that of the IRNN with L-AF at the beginning, but the speed will decrease while the error decrease to a small value. From Fig. 6(b), the convergence time is $t = 0.081$ s, which is longer than using an

TABLE I
CONVERGENCE TIME OF THE IRNN (9) WITH LINEAR ACTIVATION FUNCTION AND DIFFERENT DESIGN PARAMETERS

Parameter σ	Parameter ρ	ρ/σ	Convergence time t
0.01	1	100	0.089s
0.05	1	20	0.334s
0.1	1	10	0.641s
0.5	1	2	3.427s
1	1	1	7.185s
0.1	0.1	1	8.097s
0.1	0.5	5	1.319s
0.1	1	10	0.641s
0.1	5	50	0.153s
0.1	10	100	0.092s

L-AF. The IRNN with a P-AF possesses opposite convergence performance compared with using a BS-AF. In Fig. 6(c), the speed of using a P-AF is much slower than that of using an L-AF in the beginning. The speed becomes faster while the error is decreasing and the convergence time is $t = 5.250$ s. For obtaining a better performance, a novel SP-AF (51) is designed, and the speed is the fastest among these four kinds of activation functions. The convergence time of the IRNN with SP-AF is $t = 0.016$ s. Thus, Theorems 2–4 and the convergence properties of the IRNN with different activation functions have been verified.

B. Robustness of the Perturbed IRNN

Consider the following model-implementation error:

$$\Phi(t) = \begin{bmatrix} 0.02 \cos 4t & 0.04 \cos 4t \\ 0.06 \sin 4t & 0.02 \sin 4t \end{bmatrix}.$$

The simulation results about the robustness performance by using an L-AF are illustrated in Fig. 7. From Fig. 7(a)–(c), the residual errors of the IRNN (52) with fixed $\rho = 1$ are become smaller if the design parameter σ increases. However, for obtaining faster speed, σ should be set as small as possible. It means that the parameter σ needs to be set as a

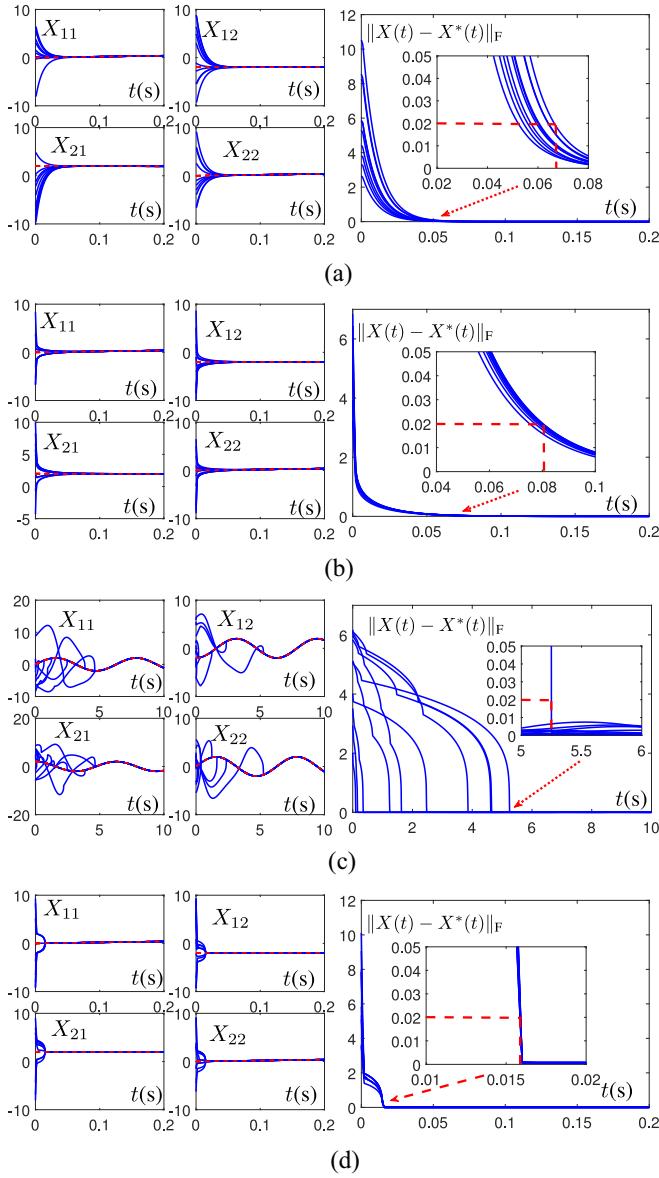


Fig. 6. Online solutions of the Sylvester equation (1) and the computational errors $\|X(t) - X^*(t)\|_F$ of the IRNN (9) with different activation function ($\sigma = 0.1$ and $\rho = 10$). (a) Linear activation function. (b) Bipolar-sigmoid activation function. (c) Power activation function. (d) SP-AF.

suitable value. In addition, the residual errors in Fig. 7(d)–(f) are become smaller while parameter ρ is increasing, that is, the larger ρ we set, the faster speed and smaller residual errors the IRNN possesses. To compare the robustness of the IRNN, a performance indicator is needed. We define the maximal residual error (MRE) as the maximal value of the computational error if $\|X(t) - X^*(t)\|_F$ is oscillation (i.e., the convergence process is complete and the computational error oscillates near a fixed value). More detailed simulation results about the MRE is listed in Table II. According to the simulation results in Table II, all the computational errors in the table are very small, which verifies the strong robustness of the P-IRNN (52) for solving the time-varying Sylvester equation.

Note 1: Take the IRNN with L-AF as an example.

Convergence: According to Theorem 2, the convergence rate of $\tilde{X}(t)$ is ρ/σ , which means that both larger ρ and smaller σ

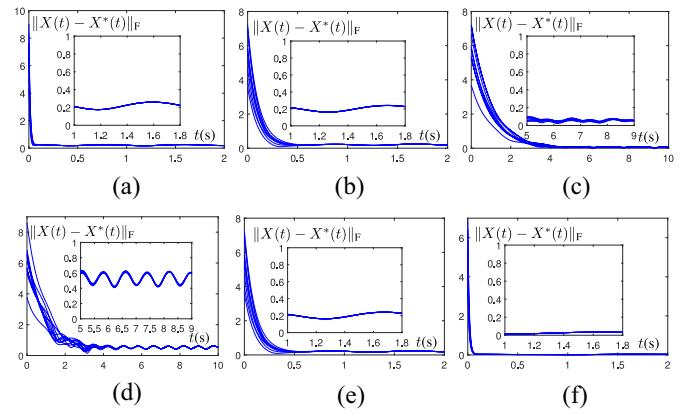


Fig. 7. Robust performance of the P-IRNN (52) with an L-AF and design parameters ρ and σ . (a) $\sigma = 0.01$, $\rho = 1$. (b) $\sigma = 0.1$, $\rho = 1$. (c) $\sigma = 1$, $\rho = 1$. (d) $\sigma = 0.1$, $\rho = 1$. (e) $\sigma = 0.1$, $\rho = 1$. (f) $\sigma = 0.1$, $\rho = 10$.

TABLE II
MAXIMAL RESIDUAL ERRORS OF THE P-IRNN (52) WITH L-AF
AND DIFFERENT DESIGN PARAMETERS

Parameter σ	Parameter ρ	ρ/σ	MRE
0.01	1	100	0.259
0.05	1	20	0.254
0.1	1	10	0.240
0.5	1	2	0.131
1	1	1	0.083
0.1	0.1	1	0.612
0.1	0.5	5	0.407
0.1	1	10	0.240
0.1	5	50	0.058
0.1	10	100	0.034

will speed up the convergence process. The convergence experiment (e.g., Table I and Figs. 4 and 5) shown in Section VI-A also verify the proof of Theorem 2.

Robustness: According to Theorem 6, when the P-IRNN is considered, larger ρ will lead to smaller $\tilde{X}(t)$ (i.e., lower MRE). The robustness experiment (e.g., Table II and Fig. 7) also verify the proof of Theorem 2. Furthermore, from Table II, when σ increases, the residual error $\tilde{X}(t)$ will decrease. Thus, for achieving the better anti-interference ability of the IRNN, larger ρ and σ are needed.

Consequently, when the IRNN is used in practical application, ρ and σ should be chosen as large as possible.

C. Large Dimension Situation

To illustrate the effectiveness of the IRNN for solving the large dimension problem, a Sylvester equation with large dimension is considered in this section. Without loss of generality, we assume that dimensions satisfying $m = n = \bar{h}$ and theoretical solution of the Sylvester equation is $X^*(t)$ with $x_{ij}(t) = 10\xi_1 \sin(200\xi_2 t + \xi_3)$. Each element in matrices $A(t)$ and $B(t)$ is $a_{ij}(t) = 10\Lambda_1 \sin(10\Lambda_2 t + \Lambda_3) + 20\vartheta(i, j)$ and $b_{ij}(t) = 10\Xi_1 \sin(10\Xi_2 t + \Xi_3) - 20\vartheta(i, j)$, respectively. $\xi_1-\xi_3$, $\Lambda_1-\Lambda_3$, and $\Xi_1-\Xi_3$ are created by using a random function within range $[-1, 1]$. $\vartheta(i, j)$ is a two-valued function. When $i = j$, $\vartheta(i, j) = 1$; otherwise, $\vartheta(i, j) = 0$. The corresponding $C(t)$ is determined through $C(t) = -A(t)X^*(t) + X^*(t)B(t)$.

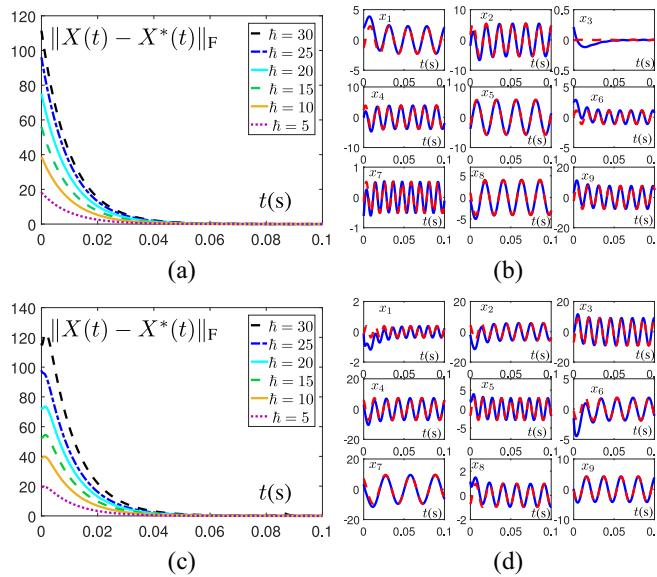


Fig. 8. Comparisons when considering different dimension $m = n = \hbar$. (a) Computational errors $\|X(t) - X^*(t)\|_F$ without disturbances. (b) Solutions of the Sylvester equation without disturbances when $\hbar = 30$. (c) Computational errors $\|X(t) - X^*(t)\|_F$ with disturbances. (d) Solutions of the Sylvester equation with disturbances when $\hbar = 30$.

The simulation when considering different dimensions of $A(t)$ and $B(t)$ are presented in Fig. 8. From Fig. 8(a), although the larger dimensions of the Sylvester equation will lead to the larger initial error, the convergence of the IRNN will be guaranteed. Furthermore, the solutions of the IRNN without considering the constant disturbance when $\hbar = 30$ will converge to the theoretical solutions [solutions x_1 - x_9 are the first nine elements of $X(t)$]. When the constant disturbance $\Phi(t)$ with $\phi_{ij}(t) = 1$ is considered, although the convergence process is disturbed and the dimensions of the equation increase, the computational errors will always converge to zero. Furthermore, the solutions of the IRNN with considering the disturbance will also converge to the theoretical solution.

D. Compared With the State-of-the-Art Methods

1) *Compared With ZNN*: The IRNN has better capacity of resisting disturbance. Consider the ZNN formula [10], [11], [13], [30]

$$\dot{E}(t) = -\gamma \mathcal{F}(E(t)) + \Delta\Phi(t) \quad (70)$$

where γ is a positive parameter which can be used to scale the convergence rate. Based on (70), the following implicit dynamic equation will be obtained as:

$$\begin{aligned} A(t)\dot{X}(t) - \dot{X}(t)B(t) \\ = -\dot{A}(t)X(t) + X(t)\dot{B}(t) - \dot{C}(t) - \gamma \mathcal{F}(A(t)X(t) - X(t)B(t) + C(t)) \\ + \Delta\Phi(t). \end{aligned} \quad (71)$$

Furthermore, the implicit dynamic equation (71) can be transformed into the vector-type formulation as

$$\begin{aligned} M(t)\dot{\vec{x}}(t) &= -\gamma \mathcal{F}(M(t)\vec{x}(t) + \vec{c}(t)) \\ &- \dot{M}(t)\vec{x}(t) - \dot{\vec{c}}(t) + \Delta\vec{\phi}(t). \end{aligned} \quad (72)$$

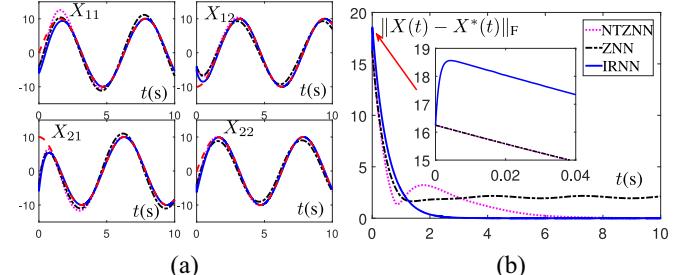


Fig. 9. Comparisons among the IRNN ($\sigma = 1$, $\rho = 2$), ZNN ($\gamma = 2$), and NTZNN ($\gamma = 2$, $\lambda = 1$) when $\Delta\Phi(t) = [0.2 - 0.2; 0.2 - 0.2]$ and $\mathcal{F}(E(t)) = E(t)$. (a) Online solutions of the Sylvester equation. (b) Computational errors $\|X(t) - X^*(t)\|_F$.

TABLE III
CONVERGENCE TIME OF THE IRNN, ZNN, AND NTZNN WITH L-AF
AND DIFFERENT DESIGN PARAMETERS

Parameter	Convergence Rate	IRNN Time t	ZNN Time t	NTZNN Time t
$\rho/\sigma = 1, \gamma = 1, \lambda = 1$	1	6.328s	$+\infty$	11.83s
$\rho/\sigma = 2, \gamma = 2, \lambda = 1$	2	3.162s	$+\infty$	6.961s
$\rho/\sigma = 4, \gamma = 4, \lambda = 1$	4	1.644s	$+\infty$	6.284s
$\rho/\sigma = 8, \gamma = 8, \lambda = 1$	8	0.819s	$+\infty$	5.560s

According to [34], the upper bound of the residual error $\|\tilde{X}(t)\|_F = \|X(t) - X^*(t)\|_F$ of the ZNN is

$$\|\tilde{X}(t)\|_F \leq \frac{\varphi\varpi(mn + \sqrt{mn})}{2\chi\gamma} \quad (73)$$

where ϖ is the upper bound satisfying $\|\Delta\Phi(t)\|_F \leq \varpi$.

When $\Delta\Phi(t)$ has direct component, $0 < \|\Delta\Phi(t)\|_F \leq \varpi$. Furthermore, according to Theorem 2, we know that $0 < \|M^{-1}(t)\|_F \leq \varphi$. That is to say, $\varpi > 0$ and $\varphi > 0$. From inequality (73), the upper bound of $\|\tilde{X}(t)\|_F$ will not approach to 0 unless γ is $+\infty$. Thus, as for disturbance with direct component, the traditional ZNN cannot solve it effectively.

The robustness Theorem 6 shows that the upper bound of the residual error $\tilde{X}(t)$ of the IRNN is

$$\|\tilde{X}(t)\|_F \leq \frac{(mn + \sqrt{mn})\varphi\xi}{2\rho} \quad (74)$$

Even if the disturbance $\Delta\Phi(t)$ has direct component, it will be suppressed by using the IRNN.

Simulation comparisons between the IRNN and the ZNN are illustrated in Fig. 9. Parameter γ of ZNN is set as 2, which means that the convergence rate of the ZNN with L-AF is 2. Equivalently, the parameters of the IRNN with L-AF are set as $\rho = 2$ and $\sigma = 1$ for obtaining the same rate $\rho/\sigma = 2$.

From Fig. 9, the state solution $X(t)$ of the Sylvester equation by using the ZNN (70) (black dashed lines) cannot converge to the theoretical solution $X^*(t)$. However, the residual error $\|X(t) - X^*(t)\|_F$ of the IRNN [blue solid line in Fig. 9(b)] can converge to 0, that is, the IRNN can work very well even the direct component disturbance exists.

2) *Compared With NTZNN*: The IRNN has better convergence properties. Consider the NTZNN formula [5], [6]

$$\dot{E}(t) = -\gamma \mathcal{F}(E(t)) - \lambda \int_0^t E(\tau) d\tau + \Delta\Phi(t). \quad (75)$$

Based on [5], by adding the integral term, the NTZNN can overcome the shortcoming of the ZNN when facing direct

TABLE IV
MAXIMAL RESIDUAL ERRORS OF THE P-IRNN, ZNN, AND NTZNN
WITH L-AF AND DIFFERENT DESIGN PARAMETERS

Parameter	Convergence Rate	IRNN MRE	ZNN MRE	NTZNN MRE
$\rho/\sigma = 1, \gamma = 1, \lambda = 1$	1	0.003	0.139	1.383
$\rho/\sigma = 2, \gamma = 2, \lambda = 1$	2	0.003	0.095	0.869
$\rho/\sigma = 4, \gamma = 4, \lambda = 1$	4	0.003	0.070	0.472
$\rho/\sigma = 8, \gamma = 8, \lambda = 1$	8	0.003	0.067	0.243

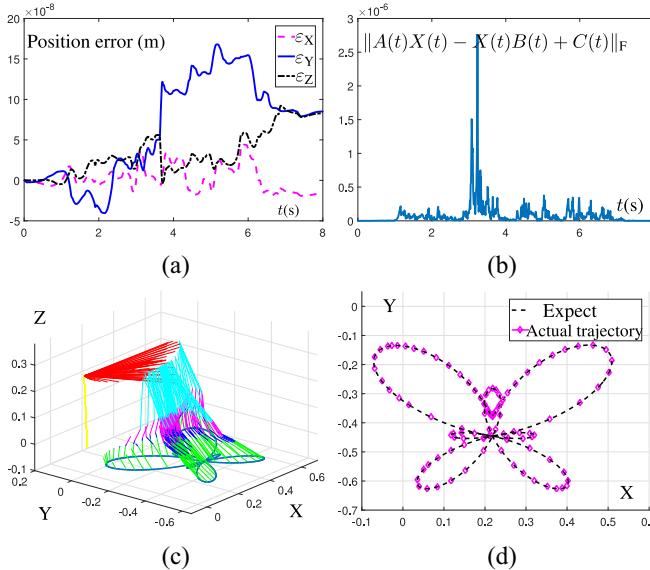


Fig. 10. Practical application to the motion planning of robot manipulators. (a) Position errors of the end-effector. (b) Computational errors $\|A(t)X(t) - X(t)B(t)\|_F$. (c) Tracking trajectories of the robot manipulator. (d) Expected path and end-effector trajectories.

component disturbance. However, due to the integral term of the NTZNN, overshoot will be occurred and convergence process will slow down compared with the ZNN.

According to the theoretical analysis presented in Section IV, the IRNN can overcome the direct component disturbance and achieve the same convergence speed as ZNN. It means that the IRNN has better performance than the NTZNN.

Simulation comparisons between the IRNN and the NTZNN are illustrated in Fig. 9. The residual error $\|X(t) - X^*(t)\|_F$ of the NTZNN (purple dotted lines) can converge to 0. However, the convergence process is slower than the IRNN along with the growth of residual error. Because of the overshoot of the state solution, the residual error of the NTZNN will not be monotone decreasing and the convergence process will slow down. In contrast, compared with the NTZNN, the IRNN has faster speed. Furthermore, the IRNN has the capability to overcome the direct component disturbance as well.

Some numerical comparisons among ZNN, NTZNN and the proposed IRNN on the convergence time and maximal residual errors have been illustrated in Tables III and IV. Those results with different convergence rates show that the proposed IRNN has faster convergence speed and stronger robustness.

According to the above comparisons, the IRNN has better properties for overcoming the direct component disturbance than the ZNN and NTZNN.

VII. APPLICATION

The application to robot kinematic problems (i.e., the motion planning problem of a redundant robot manipulator) is illustrated as follows. Consider the following forward-kinematics equation of a redundant robot manipulator:

$$r(t) = \mathfrak{F}(\Theta(t)) \quad (76)$$

where $\mathfrak{F}(\cdot)$ is a nonlinear mapping function between the robot manipulator end-effector position $r(t)$ in Cartesian space and joint angle $\Theta(t)$ in joint angle space. Due to the nonlinearity and redundancy of (76), the common solving method is to convert the position layer equation (76) to the velocity layer equation

$$J(\Theta(t))\dot{\Theta}(t) = \dot{r}(t) \quad (77)$$

where matrix $J(\Theta(t)) = \partial \mathfrak{F}(\Theta(t))/\partial \Theta(t)$ is the Jacobian matrix. Thus, the inverse kinematics problem of the robot manipulator is transformed to the solving problem of (77). Furthermore, $\dot{\Theta}(t)$ will be solved if $\dot{r}(t)$ and $J^{-1}(\Theta(t))$ [i.e., the inversion of $J(\Theta(t))$] are known. Actually, the target trajectory $r(t)$ is known if the control task is known, which means $\dot{r}(t)$ is known. Therefore, the key problem is how to obtain $J^{-1}(\Theta(t))$. However, $J(\Theta(t))$ is not a square matrix and the pseudoinverse $J^\dagger(\Theta(t))$ is considered. In fact, $J^\dagger(\Theta(t))$ is the solution of the following Sylvester equation:

$$A(t)X(t) - X(t)B(t) + C(t) = 0 \quad (78)$$

where $A(t) = 0$, $B(t) = -J(\Theta(t))J^T(\Theta(t))$ and $-J^T(\Theta(t))$. Then, the proposed IRNN can be used for solving (78). Computer simulation is presented in Fig. 10. Fig. 10(a) shows the tracking errors in x -, y -, and z -axes between the robot manipulator and the target position when the IRNN method is used. All the position errors ε_X , ε_Y , and ε_Z are all within $[-0.5 \times 10^{-7}, 2 \times 10^{-7}]$. What is more, the computational error $\|A(t)X(t) - X(t)B(t) + C(t)\|_F$ is smaller than 3×10^{-6} , which means the IRNN has capability to solve the inverse kinematics problem. Fig. 10(c) shows that the running process of the robot manipulator JACO² for tracking a butterfly trajectory. Fig. 10(d) presents that the expected path and the actual control trajectory are almost the same.

VIII. CONCLUSION

A novel IRNN with different odd-monotonically increasing activation functions has been proposed and discussed in this article. This kind of integral neural network is a new effective method for solving the time-varying Sylvester equation problem. Global convergence has been proved by using the Lyapunov function and the influence of different design parameters is analyzed theoretically. Exponential convergence of the proposed IRNN with a linear activation function has been verified. Detailed convergence performance by using bipolar-sigmoid, power, or SP-AFs is proved. It is worth mentioning that the IRNN with a power or SP-AF can achieve finite convergence time. Furthermore, the discrete IRNN has been proposed and its convergence analysis has been presented. Practical application to robot manipulator and computer simulation results have verified the effectiveness,

stability, accuracy, and robustness of the proposed IRNN for solving the time-varying Sylvester equation.

The future research work can be conducted from the following three aspects. First, such an IRNN design method can be used to various time-varying mathematics problems, such as time-varying quadratic programming, time-varying matrix inversion, time-varying inequalities, and so on. Second, the proposed IRNN can be used to some practical control systems or pattern recognition problems, such as robot manipulators, unmanned aerial vehicles, and computer vision. Third, the discrete form and the practical realization by using some hardware components of the IRNN will be considered.

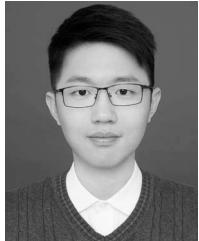
REFERENCES

- [1] S. S. Haykin, *Neural Networks and Learning Machines*. New York, NY, USA: Prentice-Hall, 2009.
- [2] S. Lu, X. Wang, G. Zhang, and X. Zhou, "Effective algorithms of the Moore-Penrose inverse matrices for extreme learning machine," *Intell. Data Anal.*, vol. 19, no. 4, pp. 743–760, 2015.
- [3] H. Chen *et al.*, "Standard plane localization in fetal ultrasound via domain transferred deep neural networks," *IEEE J. Biomed. Health Inform.*, vol. 19, no. 5, pp. 1627–1636, 2015.
- [4] H. Zhong, C. Miao, Z. Shen, and Y. Feng, "Comparing the learning effectiveness of BP, ELM, I-ELM, and SVM for corporate credit ratings," *Neurocomputing*, vol. 128, no. 5, pp. 285–295, 2014.
- [5] L. Jin, Y. Zhang, S. Li, and Y. Zhang, "Noise-tolerant ZNN models for solving time-varying zero-finding problems: A control-theoretic approach," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 992–997, Feb. 2017.
- [6] L. Jin, Y. Zhang, and S. Li, "Integration-enhanced Zhang neural network for real-time-varying matrix inversion in the presence of various kinds of noises," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 12, pp. 2615–2627, Dec. 2016.
- [7] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2013, pp. 6645–6649.
- [8] G. Li, Z. Yan, and J. Wang, "A one-layer recurrent neural network for constrained nonconvex optimization," *Neural Netw.*, vol. 61, no. 2, pp. 10–21, 2015.
- [9] Z. Zhang and L. Zheng, "A complex varying-parameter convergent-differential neural-network for solving online time-varying complex Sylvester equation," *IEEE Trans. Cybern.*, vol. 49, no. 10, pp. 3627–3639, Oct. 2019.
- [10] S. Li, S. Chen, and 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.*, vol. 37, no. 2, pp. 189–205, 2013.
- [11] L. Xiao and B. Liao, "A convergence-accelerated Zhang neural network and its solution application to Lyapunov equation," *Neurocomputing*, vol. 193, pp. 213–218, Jun. 2016.
- [12] Z. Zhang, L. Zheng, J. Weng, Y. Mao, W. Lu, and L. Xiao, "A new varying-parameter recurrent neural-network for online solution of time-varying Sylvester equation," *IEEE Trans. Cybern.*, vol. 48, no. 11, pp. 3135–3148, Nov. 2018.
- [13] Y. Zhang, B. Mu, and H. Zheng, "Link between and comparison and combination of Zhang neural network and quasi-Newton BFGS method for time-varying quadratic minimization," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 490–503, Apr. 2013.
- [14] L. Zheng and Z. Zhang, "Convergence and robustness analysis of novel adaptive multilayer neural dynamics-based controllers of multirotor UAVs," *IEEE Trans. Cybern.*, to be published.
- [15] Z. Zhang, J. Yu, Y. Li, and X. Zhang, "A new neural-dynamic control method of position and angular stabilization for autonomous quadrotor UAVs," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2016, pp. 850–855.
- [16] Z. Zhang, Z. Li, Y. Zhang, Y. Luo, and Y. Li, "Neural-dynamic-method-based dual-arm CMG scheme with time-varying constraints applied to humanoid robots," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 12, pp. 3251–3262, Dec. 2015.
- [17] B. Liao, Y. Zhang, and L. Jin, "Taylor $o(h^3)$ discretization of ZNN models for dynamic equality-constrained quadratic programming with application to manipulators," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 2, pp. 225–237, Feb. 2016.
- [18] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci. USA*, vol. 79, no. 8, pp. 2554–2558, 1982.
- [19] T. Wang, S. Zhao, W. Zhou, and W. Yu, "Finite-time state estimation for delayed Hopfield neural networks with Markovian jump," *Neurocomputing*, vol. 156, pp. 193–198, May 2015.
- [20] X. Nie, W. X. Zheng, and J. Cao, "Multistability of memristive Cohen-Grossberg neural networks with non-monotonic piecewise linear activation functions and time-varying delays," *Neural Netw.*, vol. 71, pp. 27–36, Nov. 2015.
- [21] R. Sakthivel, P. Vadivel, K. Mathiyalagan, A. Arunkumar, and M. Sivachitra, "Design of state estimator for bidirectional associative memory neural networks with leakage delays," *Inf. Sci.*, vol. 296, pp. 263–274, Mar. 2015.
- [22] Y. Chen, C. Yi, and D. Qiao, "Improved neural solution for the Lyapunov matrix equation based on gradient search," *Inf. Process. Lett.*, vol. 113, nos. 22–24, pp. 876–881, 2013.
- [23] B. Zhou, Z.-Y. Li, G. R. Duan, and Y. Wang, "Weighted least squares solutions to general coupled Sylvester matrix equations," *J. Comput. Appl. Math.*, vol. 224, no. 2, pp. 759–776, 2009.
- [24] M. Harker and P. O'Leary, "Least squares surface reconstruction from gradients: Direct algebraic methods with spectral, tikhonov, and constrained regularization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2011, pp. 2529–2536.
- [25] I. Kuzmanović and N. Truhar, "Optimization of the solution of the parameter-dependent Sylvester equation and applications," *J. Comput. Appl. Math.*, vol. 237, no. 1, pp. 136–144, 2013.
- [26] G. M. Flagg and S. Gugercin, "On the ADI method for the Sylvester equation and the optimal-H₂ points," *Appl. Numer. Math.*, vol. 64, pp. 50–58, Feb. 2013.
- [27] Q. Wei, N. Dobigeon, and J.-Y. Tourneret, "Fast fusion of multi-band images based on solving a Sylvester equation," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4109–4121, Nov. 2015.
- [28] H. Hu, Z. Lin, J. Feng, and J. Zhou, "Smooth representation clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3834–3841.
- [29] W. Ma, Y. Zhang, and J. Wang, "MATLAB Simulink modeling and simulation of Zhang neural networks for online time-varying Sylvester equation solving," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2008, pp. 285–289.
- [30] L. Xiao and Y. Zhang, "Two new types of Zhang neural networks solving systems of time-varying nonlinear inequalities," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 10, pp. 2363–2373, Oct. 2012.
- [31] H. P. Greenspan and D. J. Benney, *Calculus: An Introduction to Applied Mathematics*. Singapore: McGraw-Hill, 1970.
- [32] Y. Zhang, D. Jiang, and J. Wang, "A recurrent neural network for solving Sylvester equation with time-varying coefficients," *IEEE Trans. Neural Netw.*, vol. 13, no. 5, pp. 1053–1063, Sep. 2002.
- [33] C. Mead and M. Ismail, *Analog VLSI Implementation of Neural Systems*, vol. 80. New York, NY, USA: Springer, 2012.
- [34] Y. Zhang, G. Ruan, K. Li, and Y. Yang, "Robustness analysis of the Zhang neural network for online time-varying quadratic optimization," *J. Phys. A Math. Theor.*, vol. 43, no. 24, 2010, Art. no. 245202.
- [35] D. F. Griffiths and D. J. Higham, *Numerical Methods for Ordinary Differential Equations: Initial Value Problems*. London, U.K.: Springer, 2010.
- [36] Y. Zhang, Z. Li, D. Guo, Z. Ke, and P. Chen, "Discrete-time ZD, GD and NI for solving nonlinear time-varying equations," *Numer. Algorithms*, vol. 64, no. 4, pp. 721–740, 2013.



Zhijun Zhang (M'12–SM'19) received the Ph.D. degree from Sun Yat sen University, Guangzhou, China, in 2012.

He was a Post-Doctoral Research Fellow with the Institute for Media Innovation, Nanyang Technological University, Singapore, from 2013 to 2015. Since 2015, he has been an Associate Professor with the School of Automation Science and Engineering, South China University of Technology, Guangzhou, and also with the Hunan Provincial Key Laboratory of Finance and Economics Big Data Science and Technology, School of Information Technology and Management, Hunan University of Finance and Economics, Changsha, China. His current research interests include robotics, neural networks, and human–robot interaction.



Lunan Zheng (S'19) received the B.S. degree in automation from the South China University of Technology, Guangzhou, China, in 2017, where he is currently pursuing the Ph.D. degree in pattern recognition and intelligence system with the School of Automation Science and Engineering.

His current research interests include neural networks, machine learning, and robotics.



Xilong Qu received the Ph.D. degree in traffic information engineering and control from Southwest Jiaotong University, Chengdu, China, in 2006.

He is currently the Dean of the School of Information Technology and Management, Hunan University of Finance and Economics, Changsha, China. His current research interests include networked manufacture, agile supply chain, and integration of distributed system.



Hui Yang received the M.Sc. and Ph.D. degrees from Northeastern University, Shenyang, China, in 1988 and 2004, respectively.

He is a Professor with the School of Electrical and Automation Engineering, East China Jiaotong University, Nanchang, China. His current research interests include intelligent transportation system control, complex system modeling, control and optimization, and process industry integrated automation technology and applications.