



Li-Function Activated Zhang Neural Network for Online Solution of Time-Varying Linear Matrix Inequality

Dongsheng Guo¹ · Xinjie Lin¹

Published online: 18 June 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

In the previous work, a typical recurrent neural network termed Zhang neural network (ZNN) has been developed for various time-varying problems solving. Based on the previous work, by exploiting a special activation function (i.e., Li activation function), the resultant ZNN model is presented and investigated in this paper for online solution of time-varying linear matrix inequality (TVLMI). For such a Li-function activated ZNN (LFAZNN) model, theoretical results are provided to show its excellent computational performance on solving the TVLMI. That is, the presented LFAZNN model has the property of finite-time convergence. Comparative simulation results with two illustrative examples further substantiate the efficacy of the presented LFAZNN model for TVLMI solving.

Keywords Zhang neural network · Li activation function · Finite-time convergence · Theoretical results · Time-varying linear matrix inequality

1 Introduction

Recently, linear matrix inequality (LMI) has played a more and more important role in science and engineering areas [1–6]. The LMI-based approaches have been designed and used in many fields, such as stability analysis of fuzzy control system, robustness analysis of neural network, and gain scheduled controller design. The LMI has been now considered as a powerful formulation and design technique for solving different types of problems [1–13]. For example, Lyapunov matrix inequality is constructed to analyze the stability and performance of delayed dynamical systems [6–9]. As a powerful tool, the LMI-based technique can also be applicable for solving the system identification problems, the nonconvex multi-criterion optimal control problems, and the inverse problems of optimal control [9,13]. Generally

This work is supported by the National Natural Science Foundation of China with number 61603143, the Quanzhou City Science and Technology Program of China with number 2018C111R, the Promotion Program for Young and Middle-aged Teacher in Science and Technology Research of Huaqiao University with number ZQN-YX402, and the Scientific Research Funds of Huaqiao University with number 15BS410.

✉ Dongsheng Guo
gdongsh@hqu.edu.cn; gdongsh2008@126.com

¹ College of Information Science and Engineering, Huaqiao University, Xiamen 361021, China

speaking, these problems are transformed into the related LMI problems, which are then solved by using neural-network models and/or numerical algorithms [7–9,13]. Given its importance, an effective model/algorithm for solving a class of LMIs is worth developing and investigating.

To solve a class of LMIs (e.g., Lyapunov matrix inequalities and algebraic Riccati matrix inequalities), the gradient-based neural networks (which can be viewed as the common way) were developed by Lin et al. in [7] and [8]. In [9], a simplified neural network for a class of LMI problems solving was presented by Cheng et al. Besides, the dynamic system methods were provided by Hu [14] for solving mixed LMIs and linear vector inequalities and equalities. Note that the intrinsic design of these neural networks and methods is for time-invariant LMIs solving. As many systems in engineering applications are time-varying, the corresponding LMIs may also be time-varying ones (i.e., the coefficients always change as time t evolves). When the aforementioned neural networks and methods are exploited directly for time-varying LMIs solving, the time-derivative information of coefficients is not involved. They may be less effective and efficient due to the lack of consideration for such an important information [13,15–17].

Since March 2001, a typical recurrent neural network termed Zhang neural network (ZNN) has been formally proposed by Zhang et al. for various time-varying problems solving [16, 18]. The general procedure of the ZNN design is presented in the Appendix. For better understanding, the concept of ZNN [13] is given as follows.

Concept Being a typical recurrent neural network, Zhang neural network (ZNN) has been developed and studied since 2001. It originates from the research of Hopfield-type neural networks and is a systematic approach for time-varying problems solving. Such a ZNN is different from the traditional gradient-based neural network(s) in terms of the problem to be solved, indefinite error function, exponent-type design formula, dynamic equation, and the utilization of time-derivative information.

By following the inspirational work of Zhang et al. [16,18], two different ZNN models for time-varying LMI (TVLMI) solving were proposed and investigated in the authors' previous work [13,17]. Specifically, in [17], the first ZNN model was developed by exploiting the variant of the ZNN design formula. In [13], the second ZNN model was developed by means of the conversion from inequality to equality. For such two ZNN models, the former is depicted in an implicit dynamics, while the latter is depicted in an explicit dynamics. Theoretical and simulative results were also provided in [17] and [13] to further show the efficacy of the presented two ZNN models for TVLMI solving. By following the researches of [13] and [17], another ZNN model was developed by Sun et al. in [19] for solving time-varying Stein matrix inequality (which, to some extent, shows the significance of the ZNN research community). All of the results in [13,17] and [19] also indicate that different choices of activation functions result in different convergence performances for a ZNN model. That is, the activation function affects greatly the ZNN convergence.

In [20], a special activation function termed Li activation function was designed by Li et al. to accelerate the ZNN model to finite-time convergence. The recent research on ZNN [21–24] has further revealed that the finite-time convergence of ZNN models can be achieved by exploiting a specifically-designed activation function. Motivated by the idea in [20], this paper presents a ZNN model with finite-time convergence on TVLMI solving by employing the Li activation function. Evidently, the convergence property of such a model is different from that of the existing ZNN model [13,17,19] using the previously-investigated activation functions (e.g., power-sigmoid and hyperbolic-sine activation functions). It is also worth pointing out here that at present the research on TVLMI solving is rare (though it can be an appealing topic), not to mention a computational model with finite-time convergence.

Being an extension of the successful work [13,17,19], this paper presents and investigates the Li-function activated ZNN (LFAZNN) model for online solution of TVLMI. Specifically, by defining an indefinite matrix-valued error function and by using the ZNN design formula, the ZNN model depicted in an explicit dynamics is developed for TVLMI solving. For such a ZNN model, employing Li activation function [20] results in the LFAZNN model. Then, it is theoretically proven that the presented LFAZNN model has the property of finite-time convergence. Simulation results with two illustrative examples further substantiate the efficacy of the presented LFAZNN model for TVLMI solving.

The rest of this paper is organized into four sections. Section 2 presents the formulations of the TVLMI problem and the ZNN model. In Sect. 3, the LFAZNN model is presented and investigated for TVLMI solving, together with theoretical results showing the computational performance of such an LFAZNN model. In Sect. 4, as synthesized by the presented LFAZNN model, simulation results are illustrated. Section 5 concludes this paper with final remarks. Before ending this section, it is worth summarizing and listing the main contributions of this paper as follows.

- This paper presents the Li-function activated ZNN (LFAZNN) model for online solution of time-varying linear matrix inequality (TVLMI). Note that Li activation function is different from the activation functions investigated in [13,17,19].
- This paper gives theoretical result to show the finite-time convergence of the presented LFAZNN model. This is the first time to provide the finite-time exact solution to TVLMI.
- This paper illustrates comparative simulation results to further substantiate the efficacy of the presented LFAZNN model for TVLMI solving.

2 Time-Varying Linear Matrix Inequality

In this section, the formulation of the TVLMI problem is presented. Then, by defining an indefinite matrix-valued error function and by using the ZNN design formula [13,15–19], the resultant ZNN model is developed for TVLMI solving.

Specifically, in this paper, the following TVLMI problem is considered:

$$A(t)X(t) \leq B(t), \tag{1}$$

where $A(t) \in R^{m \times m}$ and $B(t) \in R^{m \times n}$ are smoothly time-varying matrices, and $X(t) \in R^{m \times n}$ is the unknown matrix to be obtained. The objective of this paper is to find a feasible solution $X(t)$ such that (1) holds true for any time instant $t \geq 0$. Note that, in (1), the symbol “ \leq ” (between two matrices) is satisfied, if every element of one matrix is less than or equal to the corresponding element of the other matrix. Besides, to lay a basis for further discussion, $A(t)$ is assumed to be nonsingular at any time instant $t \in [0, +\infty)$ in this paper.

To solve the TVLMI (1), the indefinite matrix-valued error function is defined as follows:

$$E(t) = A(t)X(t) - B(t) + \Lambda^2(t) \in R^{m \times n}, \tag{2}$$

where $\Lambda^2(t) = \Lambda(t) \odot \Lambda(t)$ with the time-varying matrix $\Lambda(t)$ being

$$\Lambda(t) = \begin{bmatrix} \lambda_{11}(t) & \lambda_{12}(t) & \cdots & \lambda_{1n}(t) \\ \lambda_{21}(t) & \lambda_{22}(t) & \cdots & \lambda_{2n}(t) \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{m1}(t) & \lambda_{m2}(t) & \cdots & \lambda_{mn}(t) \end{bmatrix} \in R^{m \times n}.$$

Note that the multiplication operator \odot is the Hadamard product [25]. Thus, the time-varying matrix $\Lambda^2(t)$ in (2) is written as below:

$$\Lambda(t) = \begin{bmatrix} \lambda_{11}^2(t) & \lambda_{12}^2(t) & \cdots & \lambda_{1n}^2(t) \\ \lambda_{21}^2(t) & \lambda_{22}^2(t) & \cdots & \lambda_{2n}^2(t) \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{m1}^2(t) & \lambda_{m2}^2(t) & \cdots & \lambda_{mn}^2(t) \end{bmatrix} \in R^{m \times n},$$

of which each element is greater than or equal to zero, i.e., $\Lambda^2(t) \geq 0$. Evidently, when $E(t) = 0$, the following result is obtained:

$$A(t)X(t) - B(t) = -\Lambda^2(t) \leq 0.$$

That is to say, solving the TVLMI (1) can be equivalent to solving the time-varying matrix equation $A(t)X(t) - B(t) + \Lambda^2(t) = 0$.

Besides, for further discussion, the relationship between matrices $\Lambda(t)$ and $\Lambda^2(t)$ is reformulated as

$$\text{vec}(\Lambda^2(t)) = D(t)\text{vec}(\Lambda(t)),$$

where operator $\text{vec}(\cdot) \in R^{mn}$ generates a column vector obtained by stacking all column vectors of a matrix [13,15,16]. In addition, the diagonal matrix $D(t)$ is defined as follows:

$$D(t) = \begin{bmatrix} \tilde{\lambda}_1(t) & 0 & \cdots & 0 \\ 0 & \tilde{\lambda}_2(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \tilde{\lambda}_n(t) \end{bmatrix} \in R^{mn \times mn},$$

with the i th (with $i = 1, \dots, n$) block matrix being

$$\tilde{\lambda}_i(t) = \begin{bmatrix} \lambda_{1i}(t) & 0 & \cdots & 0 \\ 0 & \lambda_{2i}(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_{mi}(t) \end{bmatrix} \in R^{m \times m}.$$

By defining $v(t) = \text{vec}(\Lambda(t)) \in R^{mn}$, the time derivative of $\text{vec}(\Lambda^2(t))$ is formulated as

$$\frac{d(\text{vec}(\Lambda^2(t)))}{dt} = 2D(t)\dot{v}(t),$$

with $\dot{v}(t)$ being the time derivative of $v(t)$. Furthermore, by defining $u(t) = \text{vec}(X(t)) \in R^{mn}$ and $w(t) = \text{vec}(B(t)) \in R^{mn}$, the error function (2) is reformulated as follows:

$$\tilde{E}(t) = M(t)u(t) - w(t) + D(t)v(t) \in R^{mn}, \tag{3}$$

where $M(t) = I \otimes A(t) \in R^{mn \times mn}$ with $I \in R^{n \times n}$ being the identity matrix and \otimes denoting the Kronecker product [13,15,16].

To make the error function $\tilde{E}(t)$ converge to zero, the time derivative of $\tilde{E}(t)$, i.e., $\dot{\tilde{E}}(t)$, is chosen via the following form (i.e., the so-called ZNN design formula) [13,15–19]:

$$\dot{\tilde{E}}(t) = -\gamma \mathcal{F}(\tilde{E}(t)), \tag{4}$$

where design parameter $\gamma > 0 \in R$, being the reciprocal of a capacitance parameter [26], is used to scale the convergence rate of the solution, and $\mathcal{F}(\cdot)$ denotes the activation-function

array. Note that, in general, the value of γ should be set as large as the hardware system would permit, or selected appropriately for simulation purposes [26]. In addition, function $f(\cdot)$, being a processing element of $\mathcal{F}(\cdot)$, can be any monotonically increasing odd activation function, e.g., the linear, power-sigmoid, power-sum and hyperbolic-sine activation functions [13,15–19].

Based on (3) and (4), the following dynamic equation of a ZNN model is established:

$$M(t)\dot{u}(t) + 2D(t)\dot{v}(t) = -\dot{M}(t)u(t) + \dot{w}(t) - \gamma\mathcal{F}(M(t)u(t) - w(t) + D(t)v(t)), \tag{5}$$

where $\dot{u}(t)$, $\dot{M}(t)$ and $\dot{w}(t)$ are the time derivatives of $u(t)$, $M(t)$ and $w(t)$, respectively. As for (5), it is reformulated as

$$\begin{bmatrix} M(t) & 2D(t) \end{bmatrix} \begin{bmatrix} \dot{u}(t) \\ \dot{v}(t) \end{bmatrix} = \begin{bmatrix} -\dot{M}(t) & 0 \end{bmatrix} \begin{bmatrix} u(t) \\ v(t) \end{bmatrix} + \dot{w}(t) - \gamma\mathcal{F}\left(\begin{bmatrix} M(t) & D(t) \end{bmatrix} \begin{bmatrix} u(t) \\ v(t) \end{bmatrix} - w(t)\right) \tag{6}$$

Thus, by defining the augmented vector $y(t) = [u^T(t), v^T(t)]^T \in R^{2mn}$, (6) is further rewritten as follows:

$$W(t)\dot{y}(t) = Q(t)y(t) + \dot{w}(t) - \gamma\mathcal{F}(P(t)y(t) - w(t)), \tag{7}$$

where $\dot{y}(t)$ is the time derivative of $y(t)$. In addition, the augmented matrices $W(t) \in R^{mn \times 2mn}$, $Q(t) \in R^{mn \times 2mn}$ and $P(t) \in R^{mn \times 2mn}$ are defined respectively as below:

$$W(t) = \begin{bmatrix} M^T(t) \\ 2D^T(t) \end{bmatrix}^T, \quad Q(t) = \begin{bmatrix} -\dot{M}^T(t) \\ 0 \end{bmatrix}^T \quad \text{and} \quad P(t) = \begin{bmatrix} M^T(t) \\ D^T(t) \end{bmatrix}^T.$$

To make (7) more computable, we can reformulate (7) as the following explicit form:

$$\dot{y}(t) = W^\dagger(t)Q(t)y(t) + W^\dagger(t)\dot{w}(t) - \gamma W^\dagger(t)\mathcal{F}(P(t)y(t) - w(t)), \tag{8}$$

where $W^\dagger(t) = W^T(t)(W(t)W^T(t))^{-1} \in R^{2mn \times mn}$. Therefore, based on the error function (2), the ZNN model (8) is obtained for TVLMI solving. For (8), the following theoretical result is provided, of which the proof can be generalized from [13,17–19].

Theorem 1 *Given a smoothly time-varying nonsingular coefficient matrix $A(t) \in R^{m \times m}$ and a smoothly time-varying coefficient matrix $B(t) \in R^{m \times n}$ in (1), if a monotonically increasing odd activation function array $\mathcal{F}(\cdot)$ is used, then ZNN model (8) generates an exact time-varying solution of the TVLMI (1).*

3 Li-Function Activated ZNN Model

In this section, by exploiting Li activation function in [20], the Li-function activated ZNN (LFAZNN) model is developed and investigated for TVLMI solving.

Specifically, Li activation function originated from [20] is presented as follows:

$$\psi(\tilde{e}_i) = \frac{1}{2}\text{Lip}^\tau(\tilde{e}_i) + \frac{1}{2}\text{Lip}^{1/\tau}(\tilde{e}_i),$$

where \tilde{e}_i is the i th element of $\tilde{E}(t)$ depicted in (3) and $\tau \in (0, 1)$. In addition, $\text{Lip}^\tau(\cdot)$ is defined as

$$\text{Lip}^\tau(\tilde{e}_i) = \begin{cases} |\tilde{e}_i|^\tau, & \text{if } \tilde{e}_i > 0, \\ 0, & \text{if } \tilde{e}_i = 0, \\ -|\tilde{e}_i|^\tau, & \text{if } \tilde{e}_i < 0, \end{cases}$$

with symbol $|\cdot|$ denoting the absolute value of a scalar.

Thus, by exploiting Li activation function array $\Psi(\cdot)$ [with $\psi(\cdot)$ being its element], the following LFAZNN model for TVLMI solving is established:

$$\dot{y}(t) = W^\dagger(t)Q(t)y(t) + W^\dagger(t)\dot{w}(t) - \gamma W^\dagger(t)\Psi(P(t)y(t) - w(t)). \tag{9}$$

As for such an LFARNN model (9), the theoretical result on the general convergence property is presented as follows, which can be generalized from Theorem 1.

Corollary *Given a smoothly time-varying nonsingular coefficient matrix $A(t) \in R^{m \times m}$ and a smoothly time-varying coefficient matrix $B(t) \in R^{m \times n}$ in (1), the presented LFAZNN model (9) generates an exact time-varying solution of the TVLMI (1).*

As to the presented LFAZNN model (9), its the neural-network structure is given in Fig. 1. In such a figure, the i th (with $i = 1, 2, \dots, 2mn$) neuron of (9) is written as follows:

$$y_i = \int \sum_{k=1}^{\mu} \hat{w}_{ik} \left(\sum_{j=1}^{\nu} q_{kj}y_j + \dot{w}_k - \gamma \psi \left(\sum_{j=1}^{\nu} p_{kj}y_j - w_k \right) \right) dt,$$

where $\nu = 2\mu = 2mn$. \hat{w}_{ik} , q_{kj} , and p_{kj} denote time-varying weights that correspond to the ik th element of $W^\dagger(t)$, the kj th element of $Q(t)$, and kj th element of $P(t)$. \dot{w}_k and w_k denote time-varying thresholds that correspond to the k th elements of $\dot{w}(t)$ and $w(t)$, respectively.

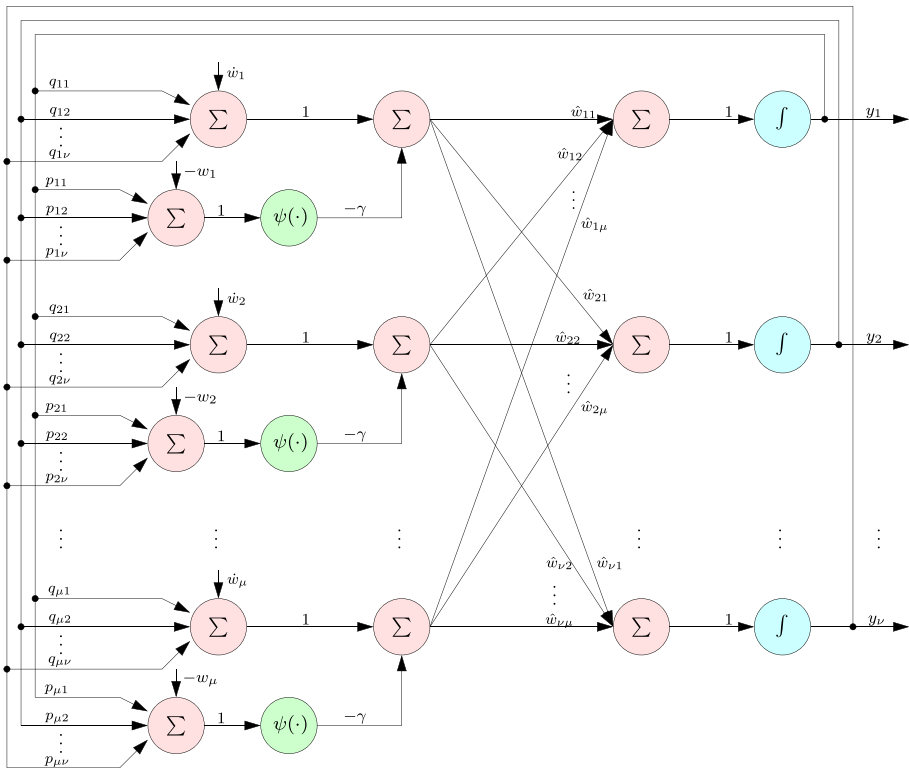


Fig. 1 Structure of the neurons in the presented LFAZNN model (9) for TVLMI solving

As shown in Fig. 1, there can be three layers for the structure of the presented LFAZNN model (9). The i th output of the first layer is $\sum_{j=1}^v q_{kj}y_j + \dot{w}_k - \gamma\psi(\sum_{j=1}^v p_{kj}y_j - w_k)$, whereas the i th output of the second layer is $\sum_{k=1}^{\mu} \hat{w}_{ik}(\sum_{j=1}^v q_{kj}y_j + \dot{w}_k - \gamma\psi(\sum_{j=1}^v p_{kj}y_j - w_k))$, i.e., \dot{y}_i (the time-derivative of y_i). The third layer possesses the integral action, and thus the i th output of such a layer is y_i . Given an initial state, the state of $\{y_1, y_2, \dots, y_v\}$ in Fig. 1 evolves as time elapses, and finally reaches an equilibrium state. At this time, the output of the third layer is the solution of the TVLMI (1), showing that the presented LFAZNN model (9) can generate a time-varying solution of (1).

Besides, as for the presented LFAZNN model (9), it has the property of finite-time convergence for TVLMI solving.

Theorem 2 *Given a smoothly time-varying nonsingular coefficient matrix $A(t) \in R^{m \times m}$ and a smoothly time-varying coefficient matrix $B(t) \in R^{m \times n}$ in (1), the error function $E(t)$ of the presented LFAZNN model (9) can converge to zero within finite time, which corresponds to the finite-time convergence of $X(t)$ [whose vectorization form is the first part of the state vector $y(t)$ of (9)] to a time-varying theoretical solution of TVLMI (1).*

Proof As for the presented LFAZNN model (9), it can be reformulated as follows:

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

where $\tilde{E}(t)$ is depicted in (3). Be defining $\tilde{e}_i(t)$ as the i th element of $\tilde{E}(t)$ (with $i = 1, 2, \dots, mn$), we have the following decoupled differential equation:

$$\dot{\tilde{e}}_i(t) = -\gamma\psi(\tilde{e}_i(t)),$$

where $\dot{\tilde{e}}_i(t)$ denotes the time derivative of $\tilde{e}_i(t)$.

For further discussion, $\tilde{E}(0) = M(0)u(0) - w(0) + D(0)v(0)$ is denoted as the initial value of $\tilde{E}(t)$ in (3). In addition, $\tilde{e}^+(t)$ is denoted as the element in $\tilde{E}(t)$ with the largest initial value $\tilde{e}^+(0) = \max\{\tilde{E}(0)\}$, and $\tilde{e}^-(t)$ is denoted as the element in $\tilde{E}(t)$ with the smallest initial value $\tilde{e}^-(0) = \min\{\tilde{E}(0)\}$.

Based on the comparison lemma [20], since $\tilde{e}^+(0) = \max\{\tilde{E}(0)\} \geq \tilde{e}_i(0)$ with $i = 1, 2, \dots, mn$, then we have $\tilde{e}^+(t) \geq \tilde{e}_i(t)$ with $t \geq 0$. Similarly, we have $\tilde{e}^-(t) \leq \tilde{e}_i(t)$ with $t \geq 0$. Thus, for all possible i , the following result is obtained, as time t goes on:

$$\tilde{e}^-(t) \leq \tilde{e}_i(t) \leq \tilde{e}^+(t),$$

which means that $\tilde{e}_i(t)$ converges to zero when both $\tilde{e}^+(t)$ and $\tilde{e}^-(t)$ reach zero. That is to say, for the presented LFAZNN model (9), its convergence time t_c is bounded by the larger one between the dynamics of $\tilde{e}^+(t)$ and that of $\tilde{e}^-(t)$. In mathematics, $t_c \leq \max\{t_c^+, t_c^-\}$, where t_c^+ and t_c^- denote the convergence time corresponding to $\tilde{e}^+(t)$ and $\tilde{e}^-(t)$.

Evidently, t_c^+ and t_c^- should be calculated first so as to obtain t_c . To calculate t_c^+ , we have

$$\dot{\tilde{e}}^+(t) = -\gamma\psi(\tilde{e}^+(t)) \text{ with } \tilde{e}^+(0) = \max\{\tilde{E}(0)\}. \tag{10}$$

Defining a Lyapunov function candidate $L(t) = |\tilde{e}^+(t)|^2$, we have its time derivative as

$$\begin{aligned} \dot{L}(t) &= -2\gamma\tilde{e}^+(t)\psi(\tilde{e}^+(t)) = -\gamma\left(|\tilde{e}^+(t)|^{\tau+1} + |\tilde{e}^+(t)|^{\frac{1}{\tau}+1}\right) \\ &\leq -\gamma|\tilde{e}^+(t)|^{\tau+1} = -\gamma L^{\frac{\tau+1}{2}}(t). \end{aligned}$$

By solving $\dot{L}(t) \leq -\gamma L^{\frac{\tau+1}{2}}(t)$ with the initial condition $L(0) = |\tilde{e}^+(0)|^2$, the following result is obtained:

$$L^{\frac{1-\tau}{2}}(t) \begin{cases} \leq |\tilde{e}^+(0)|^{1-\tau} - \frac{\gamma t(1-\tau)}{2}, & \text{if } t \leq \frac{2|\tilde{e}^+(0)|^{1-\tau}}{\gamma(1-\tau)}, \\ = 0, & \text{if } t > \frac{2|\tilde{e}^+(0)|^{1-\tau}}{\gamma(1-\tau)}. \end{cases}$$

This implies that $L(t)$ converges to zero at most after a time period $2|\tilde{e}^+(0)|^{1-\tau}/\gamma(1-\tau)$. Therefore, $\tilde{e}^+(t)$ reaches zero for $t > 2|\tilde{e}^+(0)|^{1-\tau}/\gamma(1-\tau)$, i.e., the convergence time for $\tilde{e}^+(t)$ is $t_c^+ < 2|\tilde{e}^+(0)|^{1-\tau}/\gamma(1-\tau)$. By following the similar analysis, we have $t_c^- < 2|\tilde{e}^-(0)|^{1-\tau}/\gamma(1-\tau)$. Thus, the convergence time for $\tilde{E}(t)$ is obtained as follows:

$$t_c < \max \left\{ \frac{2|\tilde{e}^-(0)|^{1-\tau}}{\gamma(1-\tau)}, \frac{2|\tilde{e}^+(0)|^{1-\tau}}{\gamma(1-\tau)} \right\}.$$

Note that $\tilde{E}(t)$ is the vectorization form of the error function $E(t)$ depicted in (2). By summarizing the above analysis, the error function $E(t)$ of the presented LFAZNN model (9) can converge to zero within finite time t_c , which corresponds to the finite-time convergence of $X(t)$ [whose vectorization form is the first part of the state vector $y(t)$ of (9)] to a time-varying theoretical solution of TVLMI (1). The proof in thus completed. \square

4 Simulative Verifications

In the previous section, the LFAZNN model (9) and its finite-time convergence property have been presented and analyzed to solve TVLMI (1). In this section, simulation results with two illustrative examples are provided to substantiate the efficacy of the presented LFAZNN model (9) for TVLMI solving.

Example 1 In the first example, let us consider TVLMI (1) with smoothly time-varying coefficient matrices $A(t)$ and $B(t)$ being as follows:

$$A(t) = \begin{bmatrix} \sin(t) & \cos(t) \\ -\cos(t) & \sin(t) \end{bmatrix} \in R^{2 \times 2} \text{ and } B(t) = \begin{bmatrix} \cos(2t) & \sin(2t) + 1 \\ -\sin(2t) + 1 & -\cos(2t) \end{bmatrix} \in R^{2 \times 2}.$$

The simulation results synthesized by the presented LFAZNN model (9) are illustrated in Figs. 2, 3, 4 and 5.

Specifically, Fig. 2 shows the neural states of the presented LFAZNN model (9) with $\tau = 0.2$ and $\gamma = 1$. As seen from Fig. 2, the trajectories of $X(t)$ [being the first four elements of $y(t)$ in (9)] and $\Lambda(t)$ [being the rest elements of $y(t)$], starting from five randomly-generated initial states, are time-varying. In addition, Fig. 3a shows the characteristics of residual error $\|P(t)y(t) - w(t)\|_2 = \|A(t)X(t) - B(t) + \Lambda^2(t)\|_F$ (with $\|\cdot\|_2$ denoting the two norm of a vector and $\|\cdot\|_F$ denoting the Frobenius norm of a matrix), from which we can observe that the residual errors of the presented LFAZNN model (9) are all convergent to zero. This means that the solutions of $X(t)$ and $\Lambda(t)$ shown in Fig. 2 are the time-varying solutions to $A(t)X(t) - B(t) + \Lambda^2(t) = 0$. Given that $-\Lambda^2(t) \leq 0$, $A(t)X(t) = B(t) - \Lambda^2(t) \leq B(t)$, showing that the $X(t)$ solution is an exact time-varying solution of TVLMI (1). For better understanding, Fig. 3b illustrates the profiles of the testing error function $\varepsilon(t) = A(t)X(t) - B(t)$. As shown in Fig. 3b, all the elements of $\varepsilon(t)$ are less than or equal to zero, i.e., $A(t)X(t) - B(t) \leq 0$. Evidently, these simulation results substantiate the efficacy of the presented LFAZNN model (9) for TVLMI solving. Besides, it is worth pointing out here that, as shown in Fig. 3a, instead of converging exponentially to zero, the residual errors

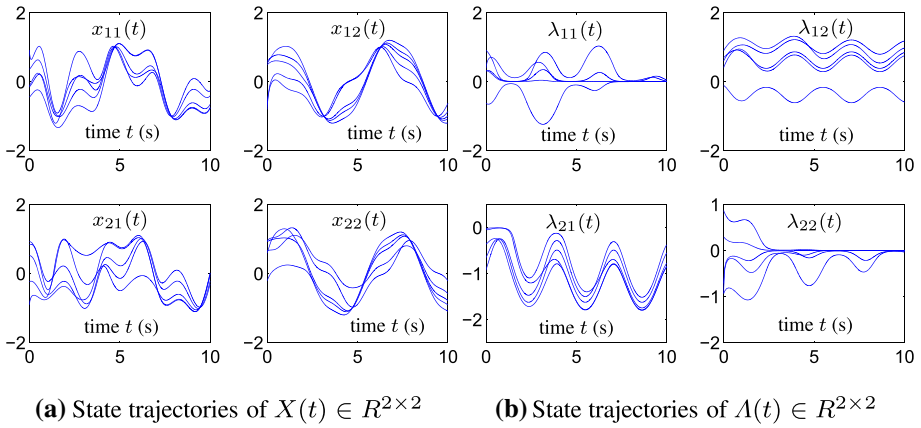


Fig. 2 Neural states of the presented LFAZNN model (9) with $\tau = 0.2$ and $\gamma = 1$ for TVLMI (1) solving

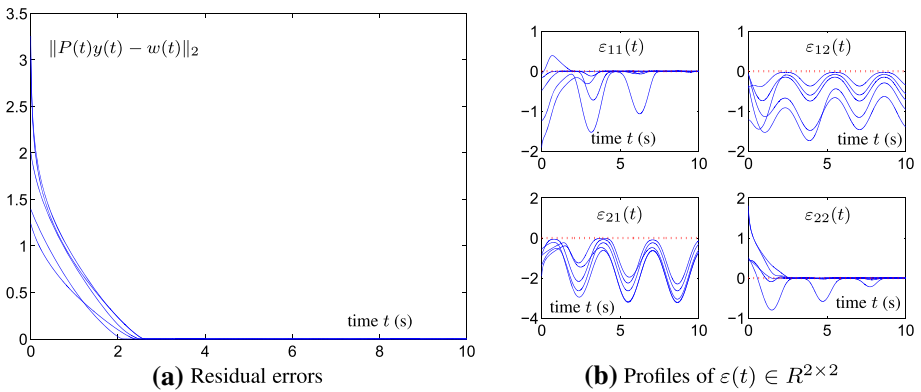


Fig. 3 Profiles of residual error $\|P(t)y(t) - w(t)\|_2$ and testing error function $\varepsilon(t) = A(t)X(t) - B(t)$ synthesized by the presented LFAZNN model (9) with $\tau = 0.2$ and $\gamma = 1$ for TVLMI (1) solving

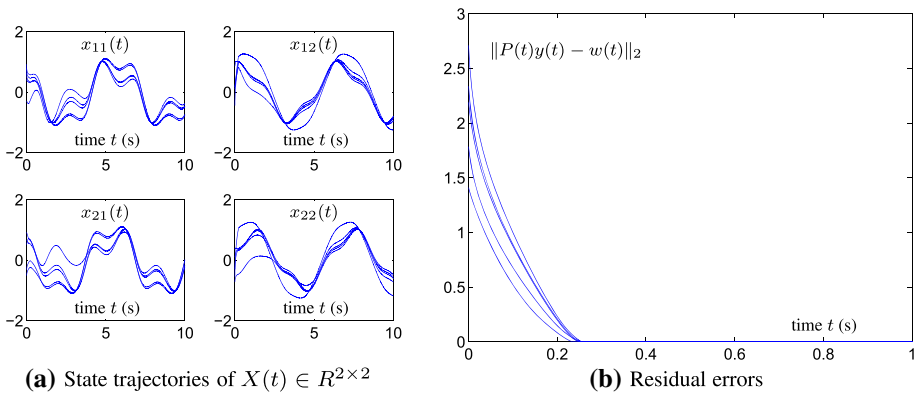


Fig. 4 Simulation results synthesized by the presented LFAZNN model (9) with $\tau = 0.2$ and $\gamma = 10$ for TVLMI (1) solving

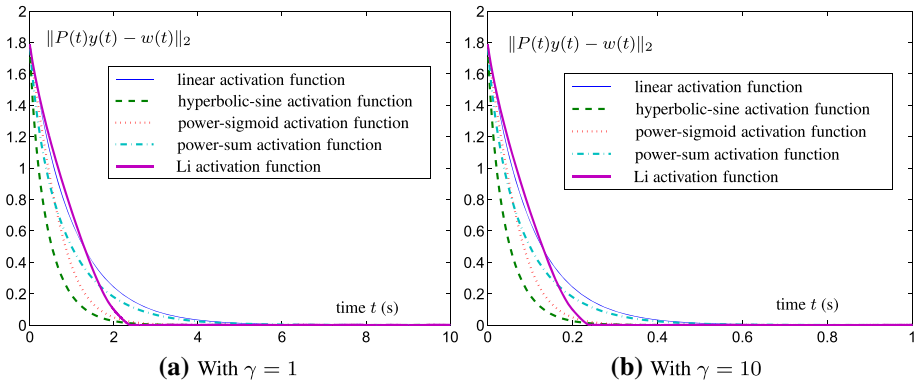


Fig. 5 Profiles of residual error $\|P(t)y(t) - w(t)\|_2$ synthesized by the ZNN model (8) using different types of activation functions and the LFAZNN model (9) with different γ values for TVLMI (1) solving

of (9) decrease directly to zero within finite time. This result of finite-time convergence is quite from the convergence results in the previous work [13,17,19].

By increasing the value of γ (i.e., from 1 to 10), the presented LFAZNN model (9) is simulated and the related results are shown in Fig. 4. As observed from such a figure [especially, Fig. 4b], the problem of TVLMI (1) is solved effectively via (9) in terms of residual errors converging to zero rapidly. This statement that the $X(t)$ trajectories in Fig. 4a satisfy $A(t)X(t) \leq B(t)$; i.e., they are the solutions to TVLMI (1). These results substantiate again the efficacy of the presented LFAZNN model (9) for TVLMI solving. Furthermore, a comparison of Figs. 3a and 4b shows that the convergence performance of (9) is improved, as the value of γ increases. This result indicates that superior computational performance of the presented LFAZNN model (9) (including its finite-time convergence) can be achieved by increasing γ .

Besides, for comparison and to provide an intuitive result on finite-time convergence, the ZNN model (8) using different types of activation functions (i.e., the linear, hyperbolic-sine, power-sigmoid, power-sum, and Li activation functions [13,15–20]) are simulated to solve TVLMI (1). The corresponding results are shown in Fig. 5, where $\tau = 0.2$ for the LFAZNN model (9). Evidently, these results indicate that the TVLMI (1) is solved successfully via (8); i.e., the residual errors are all convergent to zero. In addition, the following results are observed and summarized from Fig. 5.

- The convergence performance of the ZNN model (8) [including the LFAZNN model (9)] is improved effectively by increasing the value of γ . Thus, the important role of γ is confirmed again.
- The ZNN model (8) using the hyperbolic-sine, power-sigmoid and power-sum activation functions possesses better convergence than the one using the linear activation function.
- The LFAZNN model (9) possesses a finite-time convergence, i.e., the corresponding residual error decreases directly to zero within finite time instead of converging to zero when $t \rightarrow \infty$.

Based on these results, the LFAZNN model (9) is advantageous over the ZNN model (8) using the previously-investigated activation functions [13,15–19] because it has the property of finite-time convergence on TVLMI solving. Notably, using Li activation function, the ZNN model for solving other time-varying problems [21–24] still processes the finite-time

convergence. In addition, based on this activation function, more activation functions can be constructed to further improve the ZNN performance [27,28]. This can be viewed as the generalizability of Li activation function when comparing to those in [13,15–19].

In summary, the above simulation results (i.e., Figs. 2, 3, 4 and 5) have substantiated the efficacy of the presented LFAZNN model (9) for online solution of TVLMI (1), and more importantly, its property of finite-time convergence (in terms of the residual error decreasing directly to zero within finite time).

Example 2 In the second example, we investigate how the value of parameter τ affects the convergence performance of the presented LFAZNN model (9). The time-varying coefficient matrices $A(t)$ and $B(t)$ of TVLMI (1) in this example are designed as follows:

$$A(t) = \begin{bmatrix} 3 + \sin(3t) & \cos(3t)/2 & \cos(3t) \\ \cos(3t)/2 & 3 + \sin(3t) & \cos(3t)/2 \\ \cos(3t) & \cos(3t)/2 & 3 + \sin(3t) \end{bmatrix} \in \mathbb{R}^{3 \times 3},$$

$$B(t) = \begin{bmatrix} \sin(3t) & \cos(3t) \\ \cos(4t) + 1 & \sin(4t) + 1 \\ \sin(5t) + \cos(5t) & \sin(5t) \cos(5t) \end{bmatrix} \in \mathbb{R}^{3 \times 2}.$$

The corresponding simulation results are illustrated in Figs. 6 and 7.

Specifically, as synthesized by the presented LFAZNN model (9) with $\tau = 0.4$ and $\gamma = 5$, Fig. 6a shows the trajectories of $X(t)$ [being the first six elements of $y(t)$ in (9)] that are time-varying. Figure 6b shows the characteristics of residual error $\|P(t)y(t) - w(t)\|_2 = \|A(t)X(t) - B(t) + \Lambda^2(t)\|_F$, which indicates that the residual errors of (9) decrease directly to zero within finite time. These results substantiate the efficacy (as well as the finite-time convergence property) of the presented LFAZNN model (9) for TVLMI solving.

By changing the value of τ (i.e., $\tau = 0.6$ and $\tau = 0.8$), the presented LFAZNN model (9) is simulated to solve the aforementioned TVLMI. The related results are shown in Fig. 7, which indicate that (9) is effective on solving the TVLMI (1) in terms of the residual errors being convergent to zero rapidly and within finite time. The finite-time convergence property of (9) for TVLMI solving is thus substantiated once again. Figs. 6b and 7 are also compared, and the finding shows that the convergence times of (9) in these figures are about 0.7277 s,

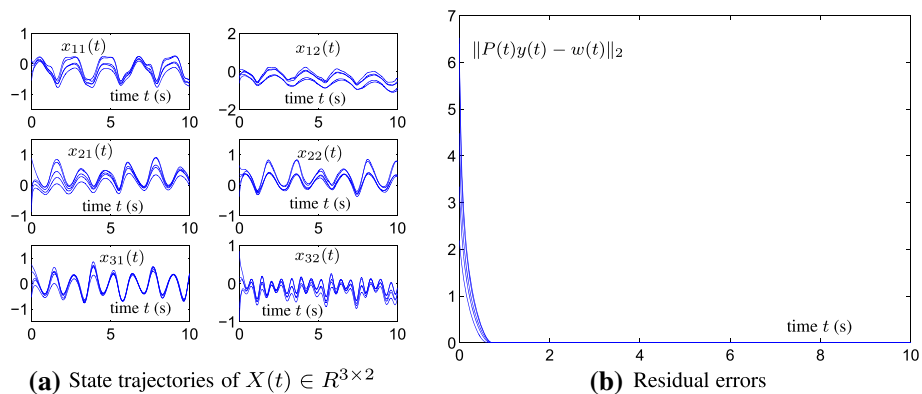


Fig. 6 Simulation results synthesized by the presented LFAZNN model (9) with $\tau = 0.4$ and $\gamma = 5$ for TVLMI (1) solving

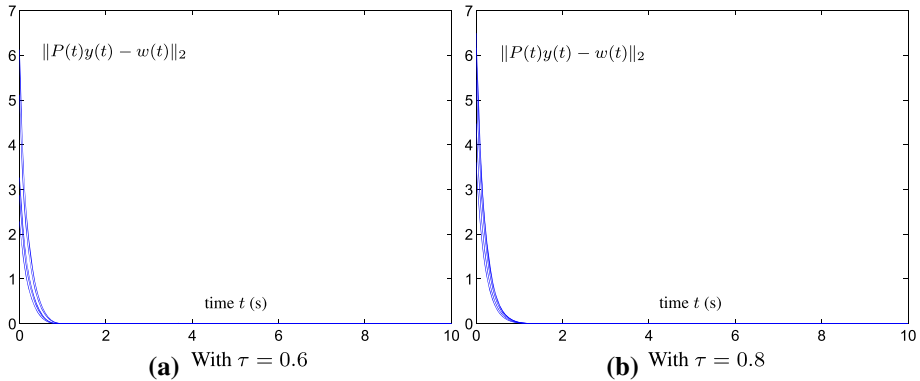


Fig. 7 Profiles of residual error $\|P(t)y(t) - w(t)\|_2$ synthesized by the presented LFAZNN model (9) with $\gamma = 5$ and different values of τ for TVLMI (1) solving

0.8778 s, and 1.5071 s, respectively. This scenario suggests that decreasing the value of τ can enhance the finite-time convergence performance of the presented LFAZNN model (9). Therefore, τ in (9) should be set appropriately small [20] to ensure the residual error of (9) converging to zero in a short period of time.

Remark It follows from Figs. 3, 4, 5, 6 and 7 that the convergence time of the presented LFAZNN model (9) can be expedited by increasing the value of γ , and can be further shortened by decreasing the value of τ . Thus, to achieve superior convergence for (9), parameter γ should be set appropriately large, while parameter τ should be set appropriately small.

5 Conclusions

In this paper, by exploiting Li activation function, the LFAZNN model (9) has been presented and investigated for online solution of TVLMI (1). Then, theoretical results have been provided to show the excellent computational performance of the presented LFAZNN model (9). That is, such an LFAZNN model has the property of finite-time convergence on solving TVLMI. Comparative simulation results with two illustrative examples have been illustrated to further substantiate the efficacy of the presented LFAZNN model (9) for TVLMI solving, as well as its finite-time convergence property (in terms of the residual error decreasing directly to zero within finite time).

Acknowledgements The authors would like to thank the editors and anonymous reviewers for their valuable suggestions and constructive comments which have really helped the authors improve very much the presentation and quality of this paper.

Appendix: Procedure of ZNN Design

In this appendix, the general procedure of the ZNN design [13,15–19] for time-varying problems solving is presented.

Specifically, as to a time-varying problem $F(t) = 0 \in R^{m \times n}$ to be solved, the following matrix/vector-valued error function $E(t) \in R^{m \times n}$ is firstly defined:

$$E(t) := F(t).$$

Next, the time derivative $\dot{E}(t)$ of $E(t)$ is selected such that $E(t)$ can globally and exponentially converge to zero. More specifically, $\dot{E}(t)$ can be selected via the ZNN design formula [13, 15–19] as follows:

$$\dot{E}(t) = -\gamma \mathcal{F}(E(t)), \quad (11)$$

where $\gamma > 0 \in R$ denotes the parameter that is used to scale the convergence rate of the solution, and $\mathcal{F}(\cdot) : R^{m \times n} \rightarrow R^{m \times n}$ denotes the monotonically-increasing odd activation-function array. Finally, by expanding the ZNN design formula (11), the differential equation of a ZNN model is thus established for solving a specific time-varying problem.

References

1. Kim E, Kang HJ, Park M (1999) Numerical stability analysis of fuzzy control systems via quadratic programming and linear matrix inequalities. *IEEE Trans Syst Man Cybern Part A* 29(4): 333–346
2. Chesi G, Garulli A, Tesi A, Vicino A (2003) Solving quadratic distance problems: an LMI-based approach. *IEEE Trans Autom Control* 48(2):200–212
3. Chesi G (2010) LMI techniques for optimization over polynomials in control: a survey. *IEEE Trans Autom Control* 55(11):2500–2510
4. Jing X (2012) Robust adaptive learning of feedforward neural networks via LMI optimizations. *Neural Netw* 31:33–45
5. Boyd S, Ghaoui LE, Feron E, Balakrishnan V (1994) *Linear matrix inequalities in system and control theory*. SIAM, Philadelphia
6. Liao X, Chen G, Sanchez EN (2002) Delay-dependent exponential stability analysis of delayed neural networks: an LMI approach. *Neural Netw* 15(7):855–866
7. Lin C-L, Lai C-C, Huang T-H (2000) A neural network for linear matrix inequality problems. *IEEE Trans Neural Netw* 11(5):1078–1092
8. Lin C-L, Huang T-H (2000) A novel approach solving for linear matrix inequalities using neural networks. *Neural Process Lett* 11(2):153–169
9. Cheng L, Hou ZG, Tan M (2009) A simplified neural network for linear matrix inequality problems. *Neural Process Lett* 29(3):213–230
10. Su T-J, Huang M-Y, Hou C-L, Lin Y-J (2010) Cellular neural networks for gray image noise cancellation based on a hybrid linear matrix inequality and particle swarm optimization approach. *Neural Process Lett* 32(2):147–165
11. Tan M (2016) Stabilization of coupled time-delay neural networks with nodes of different dimensions. *Neural Process Lett* 43(1):255–268
12. Wu H, Zhang X, Xue S, Wang L, Wang Y (2016) LMI conditions to global Mittag–Leffler stability of fractional-order neural networks with impulses. *Neurocomputing* 193:148–154
13. Guo D, Zhang Y (2014) Zhang neural network for online solution of time-varying linear matrix inequality aided with an equality conversion. *IEEE Trans Neural Netw Learn Syst* 25(2):370–382
14. Hu X (2010) Dynamic system methods for solving mixed linear matrix inequalities and linear vector inequalities and equalities. *Appl Math Comput* 216(4):1181–1193
15. Zhang Y, Guo D (2015) *Zhang functions and various models*. Springer, Heidelberg
16. Zhang Y, Yi C (2011) *Zhang neural networks and neural-dynamic method*. Nova Science Publishers, New York
17. Guo D, Zhang Y (2012) A new variant of the Zhang neural network for solving online time-varying linear inequalities. *Proc R Soc A*. 468(2144):2255–2271
18. Zhang Y, Jiang D, Wang J (2002) A recurrent neural network for solving Sylvester equation with time-varying coefficients. *IEEE Trans Neural Netw* 13(5):1053–1063
19. Sun J, Zhang S, Wang K (2016) Zhang neural networks for a set of linear matrix inequalities with time-varying coefficient matrix. *Inf Process Lett* 116(10):603–610

20. Li S, Chen S, Liu B (2013) 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(2):189–205
21. Zhang Y, Ding Y, Qiu B, Zhang Y, Li X (2017) Signum-function array activated ZNN with easier circuit implementation and finite-time convergence for linear systems solving. *Inf Process Lett* 124:30–34
22. Jin L, Li S, Wang H, Zhang Z (2018) Nonconvex projection activated zeroing neurodynamic models for time-varying matrix pseudoinversion with accelerated finite-time convergence. *Appl Soft Comput* 62:840–850
23. Xiao L, Liao B, Li S, Zhang Z, Ding L, Jin L (2018) Design and analysis of FTZNN applied to the real-time solution of a nonstationary Lyapunov equation and tracking control of a wheeled mobile manipulator. *IEEE Trans Ind Inform* 14(1):98–105
24. Xiao L, Liao B, Li S, Chen K (2018) Nonlinear recurrent neural networks for finite-time solution of general time-varying linear matrix equations. *Neural Netw* 98:102–113
25. Liu S, Trenkler G (2008) Hadamard, Khatri-Rao, Kronecker and other matrix products. *Int J Coop Inf Syst* 4(1):160–177
26. Mead C (1989) *Analog VLSI and neural systems*. Addison-Wesley, Boston, USA
27. Xiao L, Li S, Yang J, Zhang Z (2018) A new recurrent neural network with noise-tolerance and finite-time convergence for dynamic quadratic minimization. *Neurocomputing* 285:125–132
28. Lv X, Xiao L, Tan Z (2019) Improved Zhang neural network with finite-time convergence for time-varying linear system of equations solving. *Inf Process Lett* 147:88–93

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.