Chapter 1

Zhang Neural Networks for Online Solution of Time-Varying Linear Inequalities

Dongsheng Guo, Laicheng Yan and Yunong Zhang

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/62732

Abstract

In this chapter, a special type of recurrent neural networks termed "Zhang neural network" (ZNN) is presented and studied for online solution of time-varying linear (matrix-vector and matrix) inequalities. Specifically, focusing on solving the time-varying linear matrix-vector inequality (LMVI), we develop and investigate two different ZNN models based on two different Zhang functions (ZFs). Then, being an extension, by defining another two different ZFs, another two ZNN models are developed and investigated to solve the time-varying linear matrix inequality (LMI). For such ZNN models, theoretical results and analyses are presented as well to show their computational performances. Simulation results with two illustrative examples further substantiate the efficacy of the presented ZNN models for time-varying LMVI and LMI solving.

Keywords: Zhang neural network (ZNN), Zhang function (ZF), time-varying linear inequalities, design formulas, theoretical results

1. Introduction

In recent years, linear inequalities have played a more and more important role in numerous fields of science and engineering applications [1–9], such as obstacle avoidance of redundant robots [1, 2], robustness analysis of neural networks [3] and stability analysis of fuzzy control systems [5]. They, including linear matrix-vector inequality (LMVI) and linear matrix inequality (LMI), have now been considered as a powerful formulation and design technique for solving a variety of problems [7–11]. Due to their important roles, lots of numerical algorithms and neural networks have been presented and studied for online solution of linear inequalities [7–17]. For example, an iterative method was presented by Yang *et al*. for linear inequalities solving [12]. In [13], three continuous-time neural networks were developed by Cichocki and Bargiela to solve the system of linear inequalities. Besides, a gradient-based neural network and a simplified neural network were investigated respectively in [10] and [11] for solving a class of LMI problems (e.g., Lyapunov matrix inequalities and algebraic Riccati matrix inequalities).

It is worth pointing out that most of the reported approaches are designed intrinsically to solve the time-invariant (or say, static) linear inequalities. In view of the fact that many systems in science and engineering applications are time-varying, the resultant linear inequalities may be time-varying ones (i.e., the coefficients are time-varying). Generally speaking, to solve a time-varying problem, based on the assumption of the short-time invariance, such a time-varying problem can be treated as a time-invariant problem within a small time period [8]. The corresponding approaches (e.g., numerical algorithms and neural networks) are thus designed for solving the problem at each single time instant. Note that, as for this common way used to solve the time-varying problem, the time-derivative information (or say, the change trend) of the time-varying coefficients is not involved. Due to the lack of the consideration of such an important information, the aforementioned approaches may be less effective, when they are exploited directly to solve time-varying problems [7–9, 17–19].

Aiming at solving time-varying problems (e.g., time-varying matrix inversion and time-varying quadratic program), a special type of recurrent neural networks termed Zhang neural network (ZNN) has been formally proposed by Zhang *et al*. since March 2001 [7–9, 17–21]. According to Zhang *et al*.'s design method, the design of a ZNN is based on an indefinite Zhang function (ZF), with the word "indefinite" meaning that such a ZF can be positive, zero, negative or even lower-unbounded. By exploiting methodologically the time-derivative information of time-varying coefficients involved in the time-varying problems, the resultant ZNN models can thus solve the time-varying problems effectively and efficiently (in terms of avoiding the lagging errors generated by the conventional approaches) [18, 19]. For better understanding and to lay a basis for further investigation, the concepts of ZNN and ZF [18] are presented as follows.

Concept 1. Being a special type of recurrent neural networks, 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 conventional gradient 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.

Concept 2. Zhang function (ZF) is the design basis of ZNN. It differs from the common error/energy functions in the study of conventional approaches. Specifically, compared with the conventional norm-based scalar-valued positive or at least lower-bounded energy function, ZF can be bounded, unbounded or even lower-unbounded (in a word, indefinite). Besides, corresponding to a vector- or matrix-valued problem to be solved, ZF can be vector- or matrix-valued to monitor the solving process fully.

In this chapter, focusing on time-varying linear (matrix-vector and matrix) inequalities solving, we present four different ZNN models based on four different ZFs. Specifically, by defining the first two different ZFs, the corresponding two ZNN models are developed and investigated for solving the time-varying LMVI. Then, being an extension, by defining another two different ZFs, another two ZNN models are developed and investigated to solve the time-varying LMI. For such ZNN models, theoretical results and analyses are also presented to show their computational performances. Simulation results with two illustrative examples further substantiate the efficacy of the presented ZNN models for time-varying LMVI and LMI solving.

2. Preliminaries

As mentioned in Concept 2, the ZF is the design basis for deriving ZNN models to solve time-varying LMVI and LMI. Thus, for presentation convenience, in this chapter, the ZF is denoted by $E(t)$ with $\dot{E}(t)$ being the time derivative of $E(t)$. Based on the ZF, the design procedure of a ZNN model for time-varying LMVI/LMI solving is presented as follows [18, 19].

1. Firstly, an indefinite ZF is defined as the error-monitoring function to monitor the solving process of time-varying LMVI/LMI.
2. Secondly, to force the ZF (i.e., $E(t)$) converge to zero, we choose its time derivative (i.e., $\dot{E}(t)$) via the ZNN design formula (including its variant).
3. Finally, by expanding the ZNN design formula, the dynamic equation of a ZNN model is thus established for time-varying LMVI/LMI solving.

In order to derive different ZNN models to solve time-varying LMVI and LMI, the following two design formulas (being an important part in the above ZNN design procedure) are exploited in this chapter [7–9, 17–21]:

$$\dot{E}(t) = -\gamma\mathcal{F}(E(t)), \tag{1}$$

$$\dot{E}(t) = -\gamma\mathrm{SGN}(E_0)\odot\mathcal{F}(E(t)), \tag{2}$$

where $\gamma > 0 \in R$, being the reciprocal of a capacitance parameter, is used to scale the convergence rate of the solution, and $\mathcal{F}(\cdot)$ denotes the activation-function array. Note that, in general, design parameter γ should be set as large as the hardware system would permit, or selected appropriately for simulation purposes [22]. 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 and hyperbolic-sine activation functions [19, 23]. Furthermore, $E_0 = E(t = 0)$

denotes the initial error, and the unipolar signum function $\mathrm{sgn}(\cdot)$, being an element of $\mathrm{SGN}(\cdot)$, is defined as

$$\mathrm{sgn}(c) = \begin{cases} 1, & \text{if} \quad c > 0, \\ 0, & \text{if} \quad c \leq 0. \end{cases}$$

Besides, the multiplication operator $\odot$ is the Hadamard product [24] and is defined as follows:

$$U \odot V = \begin{bmatrix} u_{11}v_{11} & u_{12}v_{12} & \cdots & u_{1n}v_{1n} \\ u_{21}v_{21} & u_{22}v_{22} & \cdots & u_{2n}v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{m1}v_{m1} & u_{m2}w_{m2} & \cdots & u_{mn}v_{mn} \end{bmatrix} \in R^{m\times n}.$$

Note that, as for the presented design formulas (1) and (2), the former is the original ZNN design formula proposed by Zhang *et al.* to solve the time-varying Sylvester equation [20], while the latter is the variant of such a ZNN design formula constructed elaborately for time-varying linear inequalities solving [9]. Thus, for presentation convenience and better understanding, (1) is called the original design formula, while (2) is called the variant design formula for time-varying LMVI and LMI solving in this chapter.

Remark 1. For the variant design formula (2), when the initial error $E_0 > 0$, it reduces to $\dot{E}(t) = -\gamma\,\mathcal{F}\,(E(t))$, which is exactly the original design formula (1) for various time-varying problems solving [17–21]. In this case (i.e., $\dot{E}(t) = -\gamma\,\mathcal{F}\,(E(t))$), different convergence performances of $E(t)$ can be achieved by choosing different activation function arrays [17–21, 23]. For example, (2) reduces to $\dot{E}(t) = -\gamma E(t)$ with a linear activation function array used and with $E_0 > 0$. Evidently, its analytical solution is $E(t) = \exp(-\gamma t)E_0$, which means that $E(t)$ is globally and exponentially convergent to zero with rate γ. By following the previous successful researches [17–21, 23], superior convergence property of $E(t)$ can be achieved by exploiting nonlinear activation functions, e.g., the power-sigmoid and hyperbolic-sine activation functions [23]. In addition, the convergence property can be further improved by increasing the γ value. Therefore, in the case of $E_0 > 0$, the global and exponential convergence property is guaranteed for $E(t)$. Note that, in the case of $E_0 \leq 0$, (2) reduces to $\dot{E}(t) = 0$, meaning that $E(t) = E_0$ as time t evolves. In this situation, there is no need to investigate the convergence performance of $E(t)$ with the different activation function arrays and different γ values used.

According to the presented design formulas (1) and (2), by defining different ZFs (i.e., $E(t)$ with different formulations), different ZNN models are thus developed and investigated for time-varying LMVI and LMI solving.

3. Time-varying linear matrix-vector inequality

In this section, we introduce two different ZFs and develop the resultant ZNN models for time-varying linear matrix-vector inequality (LMVI) solving. Then, theoretical results and analyses are provided to show the computational performances of such two ZNN models.

Specifically, the following problem of time-varying LMVI [7, 9] is considered in this chapter:

$$A(t)x(t) \le b(t), \tag{3}$$

in which $A(t) \in R^{n \times n}$ and $b(t) \in R^n$ are smoothly time-varying matrix and vector, respectively. In addition, $x(t) \in R^n$ is the unknown time-varying vector to be obtained. The objective is to find a feasible solution $x(t)$ such that (3) holds true for any time instant $t \ge 0$. Note that, for further discussion, $A(t)$ is assumed to be nonsingular at any time instant $t \in [0, +\infty]$ in this chapter.

3.1. ZFs and ZNN models

In this subsection, by defining two different ZFs, two different ZNN models are developed and investigated for time-varying LMVI solving.

3.1.1. The first ZF and ZNN model

To monitor and control the process of solving the time-varying LMVI (3), the first ZF is defined as follows [7]:

$$E(t) = A(t)x(t) + \Lambda^2(t) - b(t) \in R^n, \tag{4}$$

where $\Lambda^2(t) = \Lambda(t) \odot \Lambda(t)$ with the time-varying vector $\Lambda(t)$ being $\Lambda(t) = [\lambda_1(t), \lambda_2(t), \cdots, \lambda_n(t)]^{\mathrm{T}} \in R^n$. In view of the fact that $\Lambda^2(t) \ge 0$, when $E(t) = 0$, then we have

$$A(t)x(t) - b(t) = -\Lambda^2(t) \le 0.$$

That is to say, time-varying LMVI (3) solving can be equivalent to solving the time-varying equation $A(t)x(t) + \Lambda^2(t) - b(t) = 0$. For further discussion, the following diagonal matrix $D(t)$ is defined:

$$D(t) = \begin{bmatrix} \lambda_1(t) & 0 & \cdots & 0 \\ 0 & \lambda_2(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n(t) \end{bmatrix} \in R^{n\times n},$$

which yields

$$\Lambda^2(t) = D(t)\Lambda(t) \text{ and } \dot{\Lambda}^2(t) = \frac{\mathrm{d}\Lambda^2(t)}{\mathrm{d}t} = 2D(t)\dot{\Lambda}(t).$$

with $\dot{\Lambda}(t)$ being the time derivative of $\Lambda(t)$.

On the basis of ZF (4), by exploiting the original design formula (1), the dynamic equation of a ZNN model is established as follows:

$$A(t)\dot{x}(t) + 2D(t)\dot{\Lambda}(t) = -\dot{A}(t)x(t) + \dot{b}(t) - \gamma\mathcal{F}(A(t)x(t) + \Lambda^2(t) - b(t)), \tag{5}$$

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

$$\begin{bmatrix} A(t) & 2D(t) \end{bmatrix} \begin{bmatrix} \dot{x}(t) \\ \dot{\Lambda}(t) \end{bmatrix} = \begin{bmatrix} -\dot{A}(t) & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \Lambda(t) \end{bmatrix} + \dot{b}(t) - \gamma\mathcal{F}(\begin{bmatrix} A(t) & D(t) \end{bmatrix} \begin{bmatrix} x(t) \\ \Lambda(t) \end{bmatrix} - b(t)). \tag{6}$$

By defining the augmented vector $y(t) = [x^{\mathrm{T}}(t), \Lambda^{\mathrm{T}}(t)]^{\mathrm{T}} \in R^{2n}$, (6) is further rewritten as follows:

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

with $\dot{y}(t)$ being the time derivative of $y(t)$, and the augmented matrices are being defined as below:

$$C(t) = \begin{bmatrix} A^{\mathrm{T}}(t) \\ 2D^{\mathrm{T}}(t) \end{bmatrix}^{\mathrm{T}} \in R^{n\times 2n}, P(t) = \begin{bmatrix} -\dot{A}^{\mathrm{T}}(t) \\ 0 \end{bmatrix}^{\mathrm{T}} \in R^{n\times 2n} \text{ and } Q(t) = \begin{bmatrix} A^{\mathrm{T}}(t) \\ D^{\mathrm{T}}(t) \end{bmatrix}^{\mathrm{T}} \in R^{n\times 2n}.$$

In order to make (7) more computable, we can reformulate (7) to the following explicit form:

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

where $C^{\dagger}(t) = C^{\mathrm{T}}(t)(C(t)C^{\mathrm{T}}(t))^{-1} \in R^{2n \times n}$ denotes the right pseudoinverse of $C(t)$ and the MATLAB routine "pinv" is used to obtain $C^{\dagger}(t)$ at each time instant in the simulations. Therefore, based on ZF (4), ZNN model (8) is obtained for time-varying LMVI solving. Besides, for better understanding and potential hardware implementation, ZNN model (8) is expressed in the ith (with $i = 1,2,\cdots,2n$) neuron form as

$$y_i = \int \sum_{k=1}^{n} c_{ik} \Big(\sum_{j=1}^{m} p_{kj} y_j + \dot{b}_k - \gamma f\Big(\sum_{j=1}^{m} q_{kj} y_j - b_k\Big)\Big) \mathrm{d}t,$$

where y_i denotes the ith neuron of (8), $m = 2n$ and $f(\cdot)$ is a processing element of $\mathcal{F}(\cdot)$. In addition, time-varying weights c_{ik}, p_{kj} and q_{kj} denote the ikth element of $C^{\dagger}(t)$, the kjth element of $P(t)$ and kjth element of $Q(t)$, respectively. Moreover, time-varying thresholds $\dot{b}_k$ and b_k denote, respectively, the kth elements of $\dot{b}(t)$ and $b(t)$. Thus, the neural-network structure of (8) is shown in **Figure 1**.

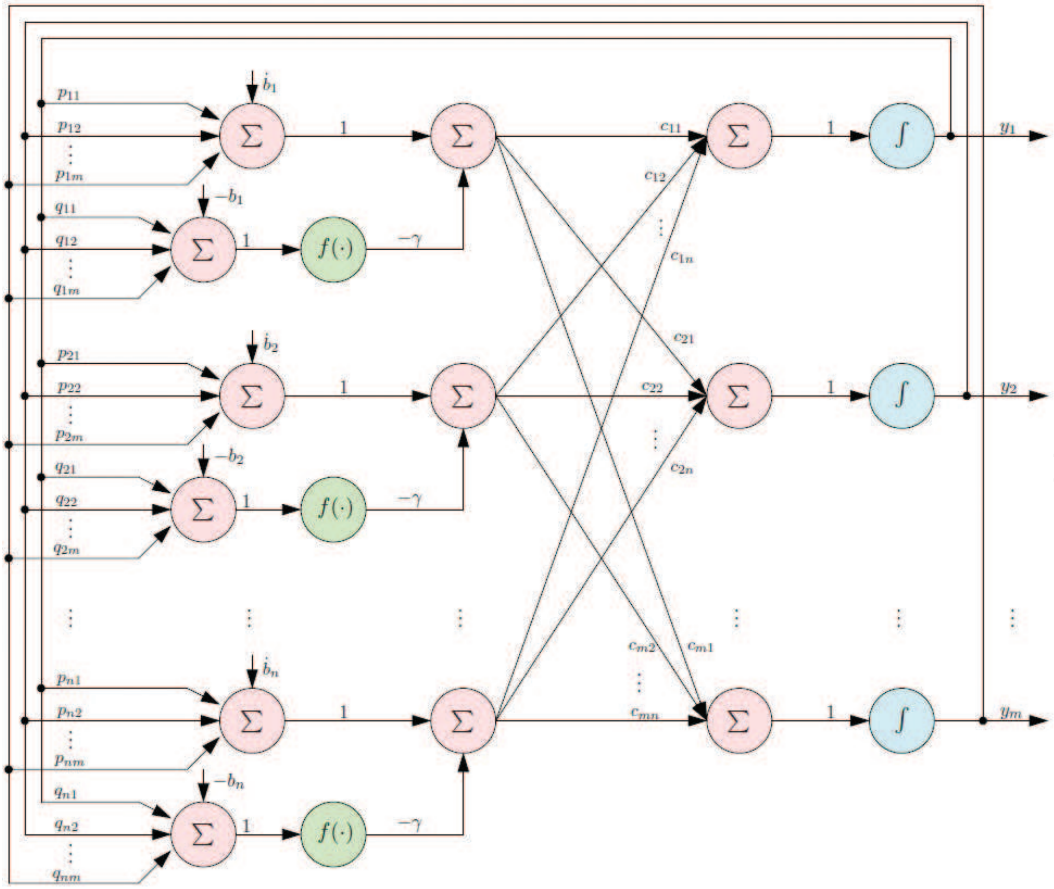


Figure 1. Structure of the neurons in ZNN model (8) for time-varying LMVI (3) solving.

3.1.2. The second ZF and ZNN model

Being different from the first ZF (4), the second ZF is defined as follows [9]:

$$E(t) = A(t)x(t) - b(t) \in R^n. \tag{9}$$

On the basis of such a ZF, by exploiting the variant design formula (2), another ZNN model is developed as follows:

$$A(t)\dot{x}(t) = -\dot{A}(t)x(t) + \dot{b}(t) - \gamma \mathrm{SGN}(E_0) \odot \mathcal{F}(A(t)x(t) - b(t)), \tag{10}$$

where the initial error $E_0 = E(t = 0) = A(0)x(0) - b(0)$. Therefore, based on ZF (9), ZNN model (10) is obtained for time-varying LMVI solving. Besides, for better understanding and potential hardware implementation, the block diagram of ZNN model (10) is shown in **Figure 2**, where $I \in R^{n \times n}$ denotes the identity matrix.

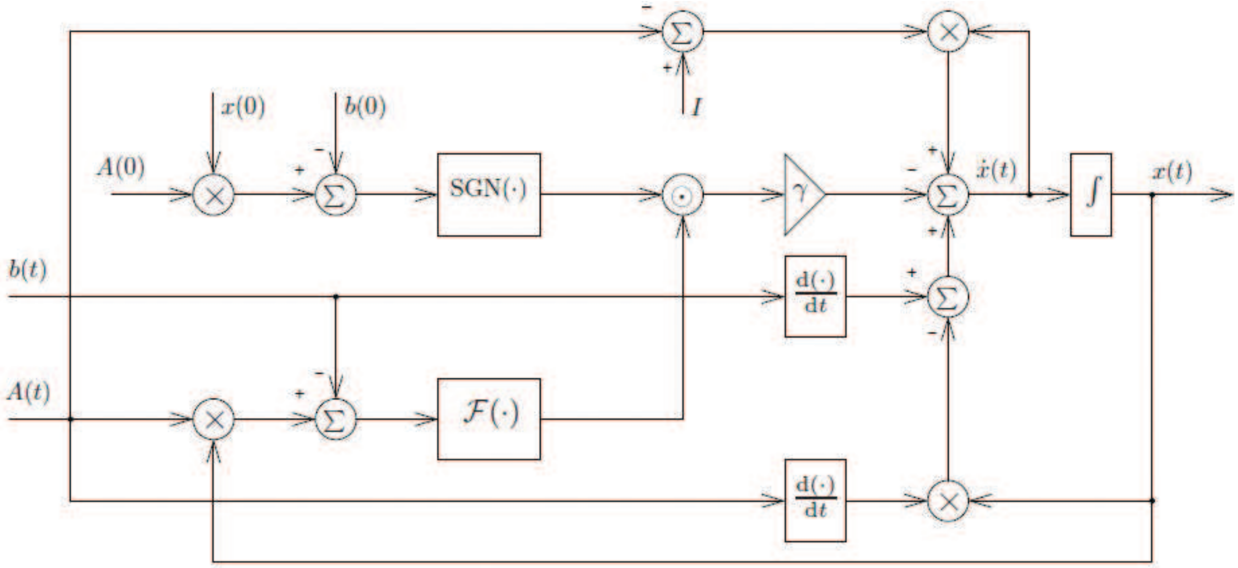


Figure 2. Block diagram of ZNN model (10) for time-varying LMVI (3) solving.

3.2. Theoretical results and analyses

In this subsection, theoretical results and analyses of the presented ZNN models (8) and (10) for solving the time-varying LMVI (3) are provided via the following theorems.

Theorem 1. Given a smoothly time-varying nonsingular coefficient matrix $A(t) \in R^{n \times n}$ and a smoothly time-varying coefficient vector $b(t) \in R^n$ in (3), 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 time-varying LMVI (3).

Proof: To lay a basis for discussion, we define $x^*(t)$ as a theoretical time-varying solution of (3), i.e., $A(t)x^*(t) \le b(t)$. Then, a time-varying vector $\Lambda^*(t)$ would exist, which results in the time-varying matrix-vector equation as follows:

$$A(t)x^*(t) + \Lambda^{*2}(t) = b(t). \tag{11}$$

By differentiating (11) with respect to time t, we have

$$A(t)\dot{x}^*(t) + \dot{A}(t)x^*(t) + \dot{\Lambda}^{*2}(t) = \dot{b}(t), \tag{12}$$

with $\dot{x}^*(t)$ and $\dot{\Lambda}^{*2}(t)$ being respectively the time derivatives of $x^*(t)$ and $\Lambda^{*2}(t)$. Based on (7), (11) and (12), we further have

$$\begin{aligned} &A(t)(\dot{x}(t) - \dot{x}^*(t)) + \dot{A}(t)(x(t) - x^*(t)) + \dot{\Lambda}^2(t) - \dot{\Lambda}^{*2}(t) \\ &= -\gamma\mathcal{F}(A(t)(x(t) - x^*(t)) + \Lambda^2(t) - \Lambda^{*2}(t)), \end{aligned}$$

which is rewritten as

$$\dot{\tilde{E}}(t) = -\gamma\mathcal{F}(\tilde{E}(t)), \tag{13}$$

where $\tilde{E}(t) = A(t)(x(t) - x^*(t)) + \Lambda^2(t) - \Lambda^{*2}(t) \in R^n$ with $\dot{\tilde{E}}(t)$ being the time derivative of $\tilde{E}(t)$.

As for (13), its compact form of a set of n decoupled differential equations is written as follows:

$$\dot{\tilde{e}}_i(t) = -\gamma f(\tilde{e}_i(t)), \tag{14}$$

where $i = 1,2,\cdots,n$. To analyze (14), we define a Lyapunov function candidate $v_i(t) = \tilde{e}_i^2(t)/2 \ge 0$ with its time derivative being

$$\dot{v}_i(t) = \frac{\mathrm{d}v_i(t)}{\mathrm{d}t} = \tilde{e}_i(t)\dot{\tilde{e}}_i(t) = -\gamma\tilde{e}_i(t)f(\tilde{e}_i(t)).$$

Since $f(\cdot)$ is a monotonically increasing odd activation function, i.e., $f(-\tilde{e}_i(t)) = -f(\tilde{e}_i(t))$, then we have

$$\tilde{e}_i(t) f(\tilde{e}_i(t)) \begin{cases} > 0, \text{ if } \tilde{e}_i(t) \neq 0, \\ = 0, \text{ if } \tilde{e}_i(t) = 0, \end{cases}$$

which guarantees the negative definiteness of $\dot{v}_i(t)$. That is to say, $\dot{v}_i(t) < 0$ for $\tilde{e}_i(t) \neq 0_,$ while $\dot{v}_i(t) = 0$ for $\tilde{e}_i(t) = 0$ only. By Lyapunov theory, $\tilde{e}_i(t)$ converges to zero for any $i \in \{1,2,\cdots,n\}$, thereby showing that $\tilde{E}(t)$ is convergent to zero as well.

Besides, based on (11), we have $A(t)x^*(t) + \Lambda^{*2}(t) = b(t)$. Then, $\tilde{E}(t)$ is rewritten as $\tilde{E}(t) = A(t)x(t) + \Lambda^2(t) - b(t)$, which is equivalent to $A(t)x(t) - b(t) = -\Lambda^2(t) + \tilde{E}(t)$. Note that, as analyzed previously, $\tilde{E}(t) \to 0$ with time $t \to +\infty$. Thus, as time evolves,

$$A(t)x(t) - b(t) = -\Lambda^2(t) + \tilde{E}(t) \to -\Lambda^2(t).$$

Since $-\Lambda^2(t) \leq 0$ (i.e., each element is less than or equal to zero), then we have $A(t)x(t) - b(t) \leq 0$. This implies that $x(t)$ (being the first n elements of $y(t)$ of (8)) would converge to a time-varying vector which satisfies the time-varying LMVI (3); i.e., $x(t) \to x^*(t)$ to make (3) hold true. In summary, the presented ZNN model (8) generates an exact time-varying solution of the time-varying LMVI (3). The proof is thus completed. □

Theorem 2. Given a smoothly time-varying nonsingular coefficient matrix $A(t) \in R^{n \times n}$ and a smoothly time-varying coefficient vector $b(t) \in R^n$ in (3), if a monotonically increasing odd activation function array $\mathcal{F}(\cdot)$ is used, then ZNN model (10) generates an exact time-varying solution of the time-varying LMVI (3).

Proof: Consider ZNN model (10), which is derived from the variant design formula (2). Thus, there are three cases as follows.

1. If the randomly generated initial state $x(0) \in R^n$ is outside the initial solution set $S(0)$ of (3), i.e., $E_0 > 0$ in (10), based on Remark 1 and the previous work [9], the global and exponential convergence of the error function $E(t)$ is achieved (or say, $E(t) = A(t)x(t) - b(t) \to 0$ globally and exponentially). This also means that the neural state $x(t)$ of ZNN model (10) is convergent to the theoretical time-varying solution of the matrix-vector equation $A(t)x(t) - b(t) = 0$. Note that $A(t)x(t) - b(t) = 0$ (i.e., $A(t)x(t) = b(t)$) is a special case of $A(t)x(t) \leq b(t)$. Therefore, ZNN model (10) is effective on solving the time-varying LMVI (3), in terms of $x(t)$ being convergent to the time-varying solution set $S(t)$ of (3).
2. If $x(0)$ is inside $S(0)$ of (3), i.e., $E_0 \leq 0$ in (10), based on Remark 1, the error function $E(t)$ would remain E_0 with $t \to +\infty$. That is, $E(t) = E_0 \leq 0$, no matter how time t evolves. In this situation, ZNN model (10) is still effective on solving the time-varying LMVI (3), in terms of its neural state $x(t)$ always being inside $S(t)$ of (3).

3. If some elements of $x(0)$ are inside $S(0)$ of (3) while the others are outside $S(0)$, i.e., some elements of E_0 are greater than zero while the rest elements of E_0 are less than or equal to zero, then, (i) for the elements of $E(t)$ that have positive initial values (i.e., their initial values are greater than zero), they can be convergent to zero globally and exponentially; and (ii) for the rest elements of $E(t)$, they can be always equal to their initial values that are less than or equal to zero. In view of the fact that, as time evolves, each element of $E(t) = A(t)x(t) - b(t)$ is less than or equal to zero, ZNN model (10) is thus effective on solving the time-varying LMVI (3).

By summarizing the above analyses, the time-varying LMVI (3) is solved effectively via ZNN model (10), in the sense that such a model can generate an exact time-varying solution of (3). The proof is thus completed. □

Remark 2. On the basis of two different ZFs (i.e., (4) and (9)), two different ZNN models (i.e., (8) and (10)) are obtained for online solution of the time-varying LMVI (3). Note that the former aims at solving (3) aided with equality conversion (i.e., from inequality to equation) and the original design formula (1), while the latter focuses on solving (3) directly with the aid of the variant design formula (2). The resultant ZNN model (8) is depicted in an explicit dynamics (i.e., $\dot{y}(t) = \cdots$), and ZNN model (10) is depicted in an implicit dynamics (i.e., $A(t)\dot{x}(t) = \cdots$). As analyzed above and as demonstrated by the simulation results shown in Section 5, such two ZNN models are both effective on solving the time-varying LMVI (3). In summary, two different approaches for time-varying LMVI solving have been discovered and presented in this chapter; i.e., one is based on the variant of the original ZNN design formula, and the other is based on the conversion from inequality to equation. This can be viewed as an important breakthrough on (time-varying or static) inequalities solving [7–9].

4. Time-varying linear matrix inequality

In this section, being an extension, by defining another two different ZFs, another two ZNN models are developed and investigated for time-varying linear matrix inequality (LMI) solving.

Specifically, the following problem of time-varying LMI is considered [9]:

$$A(t)X(t) \leq B(t), \tag{15}$$

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. Note that (15) is a representative time-varying LMI problem which is studied here. The design approaches presented in this chapter (more specifically, summarized in Remark 2) can be directly extended to solve other types of time-varying LMIs [8, 10, 11].

4.1. The first ZF and ZNN model

In order to solve the time-varying LMI (15), the first ZF is defined as follows:

$$E(t) = A(t)X(t) + \Lambda^2(t) - B(t) \in R^{m\times n}, \tag{16}$$

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

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

In addition, for matrices $\Lambda(t)$ and $\Lambda^2(t)$, we have

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

where operator $\mathrm{vec}(\cdot) \in R^{mn}$ generates a column vector obtained by stacking all column vectors of a matrix together [8, 18, 19]. 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 ith (with $i = 1,\cdots,n$) 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 $u(t) = \mathrm{vec}(X(t)) \in R^{mn}$, $v(t) = \mathrm{vec}(\Lambda(t)) \in R^{mn}$ and $w(t) = \mathrm{vec}(B(t)) \in R^{mn}$, ZF (16) is reformulated as $E(t) = M(t)u(t) - w(t) + D(t)v(t) \in R^{mn}$, 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 [18, 19]. Thus, on the basis of (16), by exploiting the original design formula (1), we have

$$C(t)\dot{y}(t) = P(t)y(t) + \dot{w}(t) - \gamma\mathcal{F}(Q(t)y(t) - w(t)), \tag{17}$$

where the augmented vector $y(t) = [u^{\mathrm{T}}(t), v^{\mathrm{T}}(t)]^{\mathrm{T}} \in R^{2mn}$, and $\dot{y}(t) \in R^{2mn}$ and $\dot{w}(t) \in R^{mn}$ are the time derivatives of $y(t)$ and $w(t)$, respectively. In addition, the augmented matrices are defined as

$$C(t) = \begin{bmatrix} M^{\mathrm{T}}(t) \\ 2D^{\mathrm{T}}(t) \end{bmatrix}^{\mathrm{T}} \in R^{mn\times 2mn}, P(t) = \begin{bmatrix} -N^{\mathrm{T}}(t) \\ 0 \end{bmatrix}^{\mathrm{T}} \in R^{mn\times 2mn}$$
$$\text{and } Q(t) = \begin{bmatrix} M^{\mathrm{T}}(t) \\ D^{\mathrm{T}}(t) \end{bmatrix}^{\mathrm{T}} \in R^{mn\times 2mn},$$

where $N(t) = \dot{M}(t) = I \otimes \dot{A}(t) \in R^{mn \times mn}$ with $\dot{A}(t)$ being the time derivative of $A(t)$.

Similarly, to make (17) more computable, we can reformulate (17) as the following explicit form:

$$\dot{y}(t) = C^{\dagger}(t)P(t)y(t) + C^{\dagger}(t)\dot{w}(t) - \gamma C^{\dagger}(t)\mathcal{F}(Q(t)y(t) - w(t)), \tag{18}$$

where $C^{\dagger}(t) = C^{\mathrm{T}}(t)(C(t)C^{\mathrm{T}}(t))^{-1} \in R^{2mn \times mn}$. Therefore, based on ZF (16), ZNN model (18) is obtained for time-varying LMI solving. Note that the neural-network structure of (18) is similar to the one shown in **Figure 1**, and is thus omitted here. Besides, as for ZNN model (18), we have the following theoretical result, with the related proof being generalized from the proof of Theorem 1 and being left to interested readers to complete as a topic of exercise.

Corollary 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 (15), if a monotonically increasing odd activation function array $\mathcal{F}(\cdot)$ is used, then ZNN model (18) generates an exact time-varying solution of the time-varying LMI (15).

4.2. The second ZF and ZNN model

In this subsection, being different from the first ZF (16), the second ZF is defined as follows:

$$E(t) = A(t)X(t) - B(t) \in R^{m\times n}. \tag{19}$$

On the basis of such a ZF, by exploiting the variant design formula (2), the following ZNN model for time-varying LMI solving is developed:

$$A(t)\dot{X}(t) = -\dot{A}(t)X(t) + \dot{B}(t) - \gamma \mathrm{SGN}(E_0) \odot \mathcal{F}(A(t)X(t) - B(t)), \tag{20}$$

where the initial error $E_0 = E(t = 0) = A(0)X(0) - B(0)$. Note that, due to similarity to the block diagram of (10), the block diagram of ZNN model (20) is omitted. Besides, as for ZNN model (20), we have the following theoretical result, of which the proof is generalized from the proof of Theorem 2 (and is also left to interested readers to complete as a topic of exercise).

Corollary 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 (15), if a monotonically increasing odd activation function array $\mathcal{F}(\cdot)$ is used, then ZNN model (20) generates an exact time-varying solution of the time-varying LMI (15).

5. Simulative verifications

In this section, one illustrative example is first simulated for demonstrating the efficacy of the presented ZNN models (8) and (10) for solving the time-varying LMVI (3). Then, another illustrative example is provided for substantiating the efficacy of the presented ZNN models (18) and (20) for solving the time-varying LMI (15).

Example 1 In the first example, the following smoothly time-varying coefficient matrix $A(t)$ and coefficient vector $b(t)$ of (3) are designed to test ZNN models (8) and (10):

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

$$\text{and } b(t) = \begin{bmatrix} \sin(3t)+1 \\ \cos(3t)+2 \\ \sin(3t)+\cos(3t)+3 \end{bmatrix} \in R^3.$$

The corresponding simulation results are shown in **Figures 3** through **9**.

Specifically, **Figures 3** and **4** illustrate the state trajectories synthesized by ZNN model (8) using $\gamma = 1$ and the power-sigmoid activation function. As shown in **Figures 3** and **4**, starting from five randomly generated initial states, the $x(t)$ trajectories (being the first 3 elements of $y(t)$ in (8)) and the $\Lambda(t)$ trajectories (being the rest elements of $y(t)$) are time-varying. In addition, **Figure 5** presents the characteristics of residual error

$\| Q(t)y(t) - b(t) \|_2 = \| A(t)x(t) + \Lambda^2(t) - b(t) \|_2$ (with symbol $\| \cdot \|_2$ denoting the two norm of a vector), from which we can observe that the residual errors of ZNN model (8) (corresponding to **Figures 3** and **4**) are all convergent to zero. This means that the $x(t)$ and $\Lambda(t)$ solutions shown in **Figures 3** and **4** are the time-varying solutions of $A(t)x(t) + \Lambda^2(t) - b(t) = 0$. In view of $-\Lambda^2(t) \leq 0$, such a solution of $x(t)$ is an exact solution of the time-varying LMVI (3), i.e., $A(t)x(t) \leq b(t)$. For better understanding, the profiles of the testing error function $\varepsilon(t) = A(t)x(t) - b(t)$ (i.e., ZF (9)) are illustrated in **Figure 6**. As shown in the figure, all the elements of $\varepsilon(t)$ are less than or equal to zero, thereby meaning that the $x(t)$ solution satisfies $A(t)x(t) \leq b(t)$ (being an exact time-varying solution of (3)). These simulation results substantiate the efficacy of ZNN model (8) for time-varying LMVI solving. Besides, **Figure 7** shows the simulation results synthesized by ZNN model (8) using different γ values (i.e., $\gamma = 1$ and $\gamma = 10$) and different activation functions (i.e., linear, hyperbolic-sine and power-sigmoid activation functions). As seen from **Figure 7**, the residual errors all converge to zero, which means that ZNN model (8) solves the time-varying LMVI (3) successfully. Note that, from **Figure 7**, we have a conclusion that superior computational performance of ZNN model (8) can be achieved by increasing the γ value and choosing a suitable activation function.

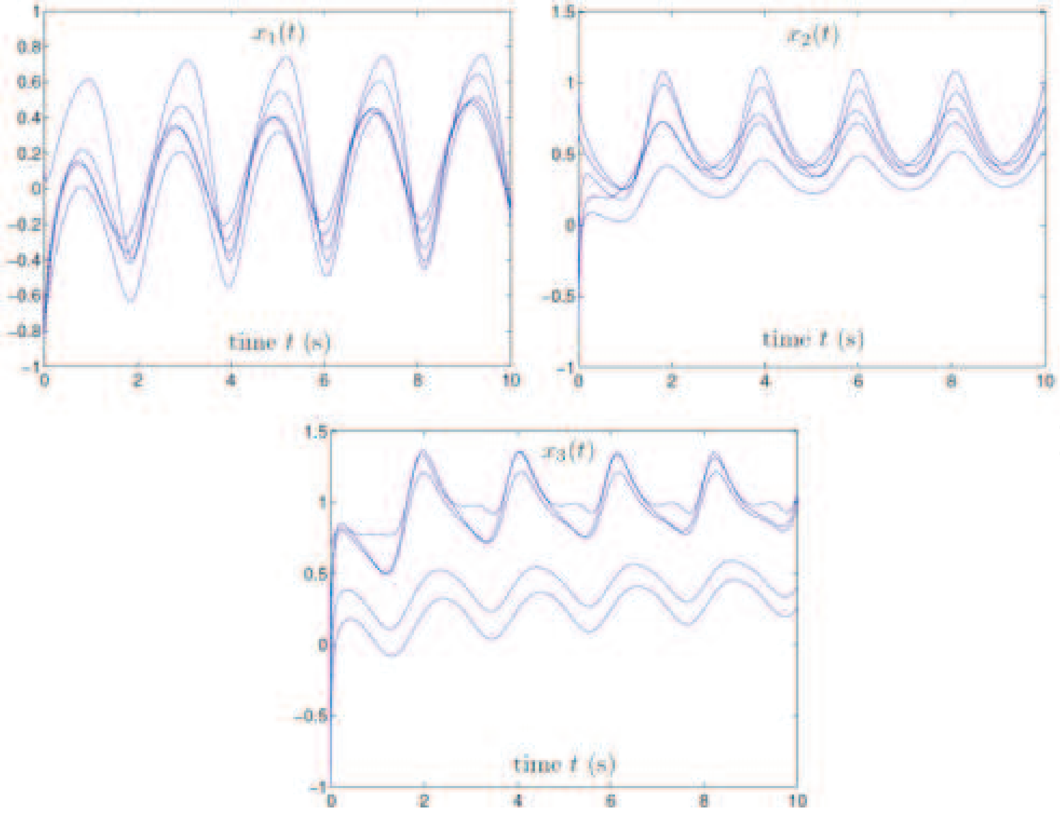


Figure 3. State trajectories of $x(t) \in R^3$ synthesized by ZNN model (8) with $\gamma = 1$ and the power-sigmoid activation function used for time-varying LMVI (3) solving.

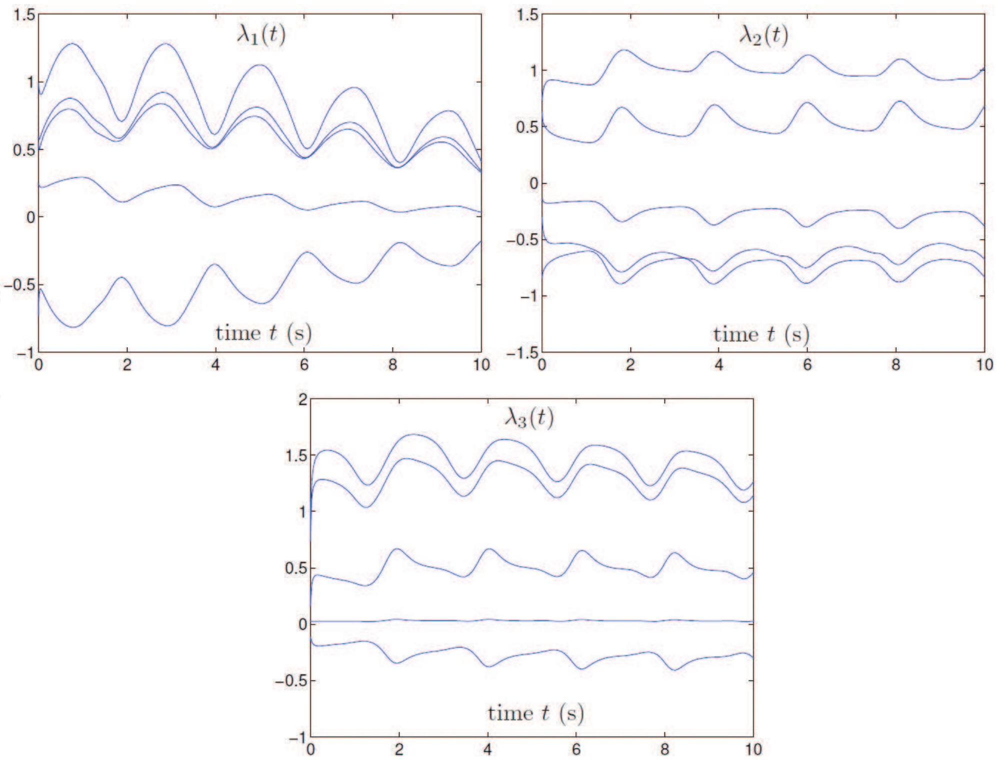


Figure 4. State trajectories of $\Lambda(t) \in R^3$ synthesized by ZNN model (8) with $\gamma = 1$ and the power-sigmoid activation function used for time-varying LMVI (3) solving.

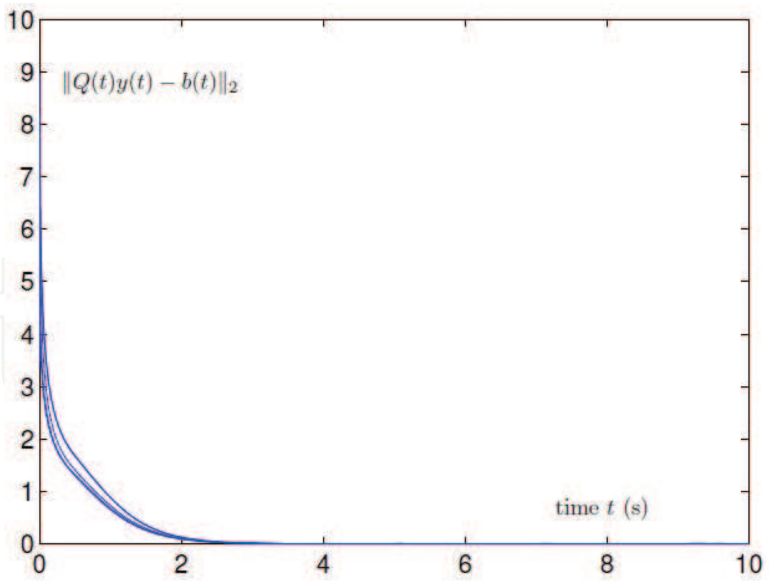


Figure 5. Residual errors $\| Q(t)y(t) - b(t) \|_2$ of ZNN model (8) for time-varying LMVI (3) solving.

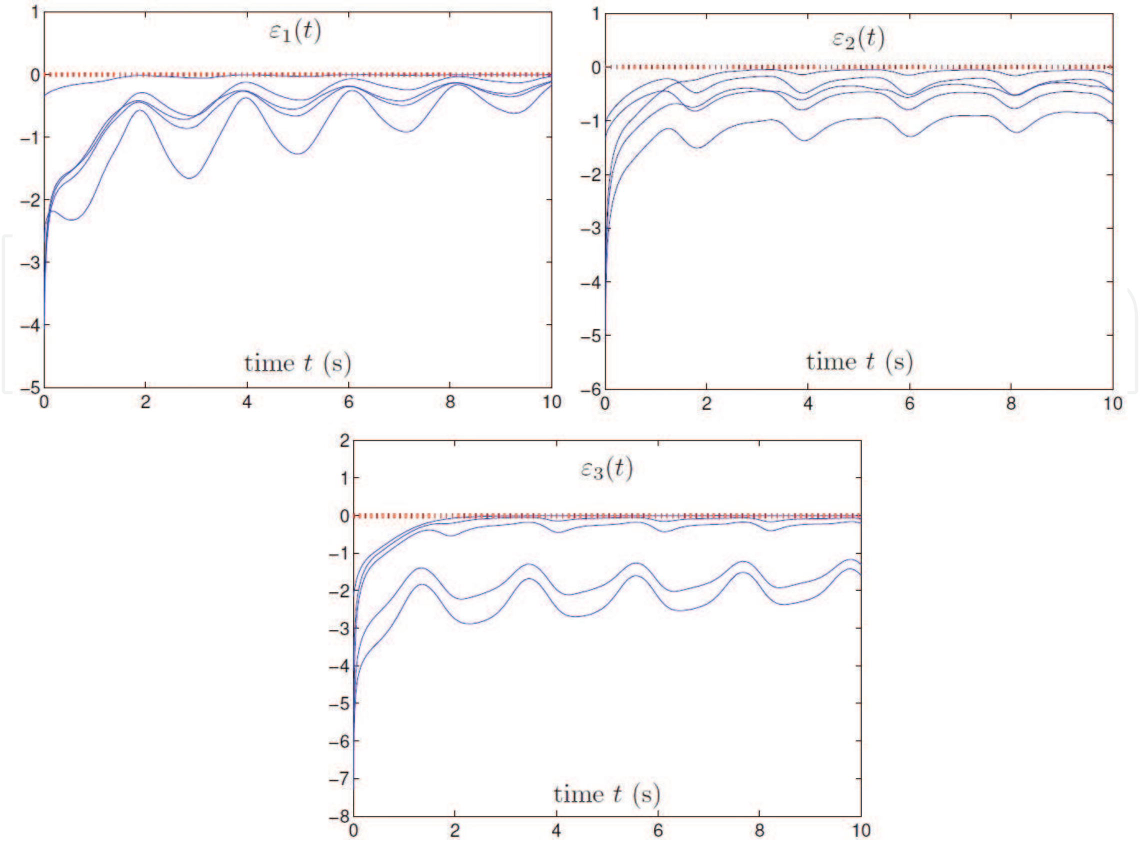
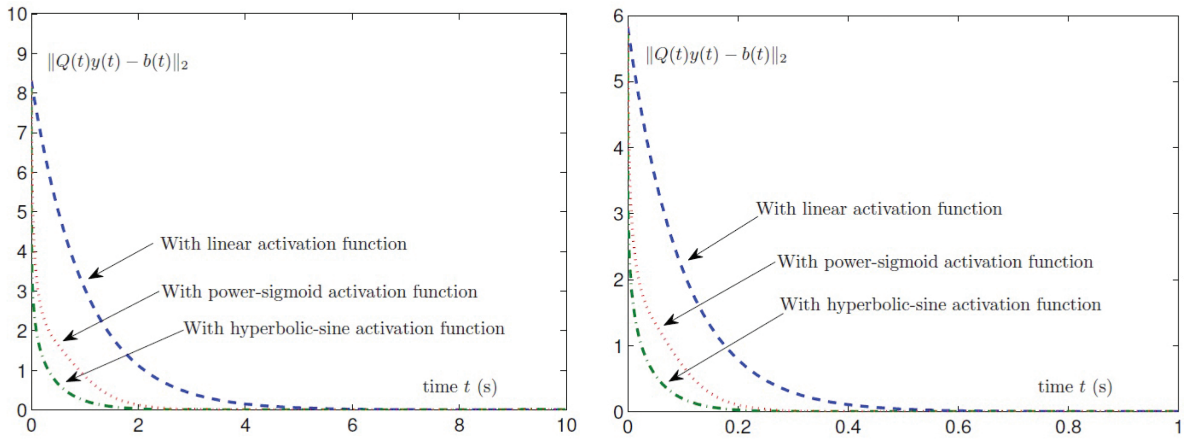


Figure 6. Profiles of $\varepsilon(t) = A(t)x(t) - b(t) \in R^3$ synthesized by ZNN model (8) with $\gamma = 1$ and the power-sigmoid activation function used for time-varying LMVI (3) solving.



(a) With $\gamma = 1$

(b) With $\gamma = 10$

Figure 7. Residual errors $\| Q(t)y(t) - b(t) \|_2$ of ZNN model (8) with γ fixed and different activation functions used for time-varying LMVI (3) solving.

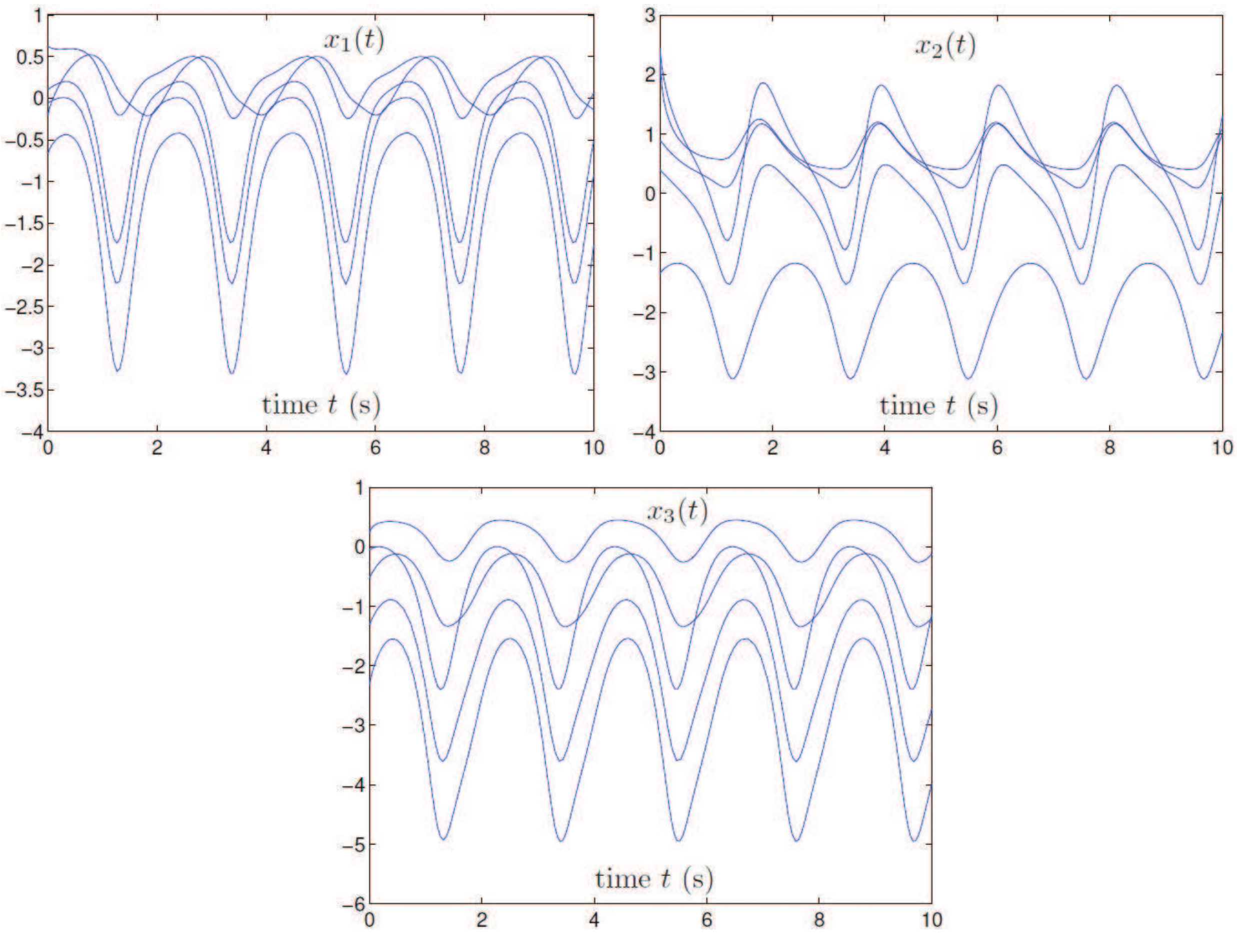


Figure 8. State trajectories of $x(t) \in R^3$ synthesized by ZNN model (10) with $\gamma = 1$ and the power-sigmoid activation function used for time-varying LMVI (3) solving.

It is worth pointing out here that, in general, it may be difficult to know whether the initial state $x(0)$ used for simulation/application is outside the initial solution set $S(0)$ of the time-varying LMVI (3) or not. Thus, as for ZNN model (10), we focus on investigating its computational performance when some elements of $x(0)$ are outside $S(0)$ while the others are inside $S(0)$. In this case, some elements of the initial error $E_0 = A(0)x(0) - b(0)$ are greater than zero, while the rest are less than or equal to zero. The corresponding simulation results synthesized by ZNN model (10) using $\gamma = 1$ and the power-sigmoid activation function are illustrated in **Figures 8** and **9**. As shown in **Figure 8**, starting from five randomly generated initial states, the $x(t)$ trajectories of ZNN model (10) are time-varying. Besides, from **Figure 9** which shows the profiles of the testing error function $\varepsilon(t) = A(t)x(t) \leq b(t)$, we can observe that the elements of $\varepsilon(t)$ with positive initial values are convergent to zero, while the rest elements remain at their initial values. This result implies that the $x(t)$ solutions shown in **Figure 6** are the time-varying solutions of (3), i.e., $A(t)x(t) \leq b(t)$, thereby showing the efficacy of ZNN model (10) for time-varying LMVI solving. That is, ZNN model (10) generates an exact time-varying solution of the time-varying LMVI (3). Note that the computational performance of ZNN model (10) can be improved by increasing the value of γ and choosing a suitable activation function (which is similar to that of ZNN model (8)). Being a topic of exercise, the corresponding simulative verifications of ZNN model (10) are left for interested readers.

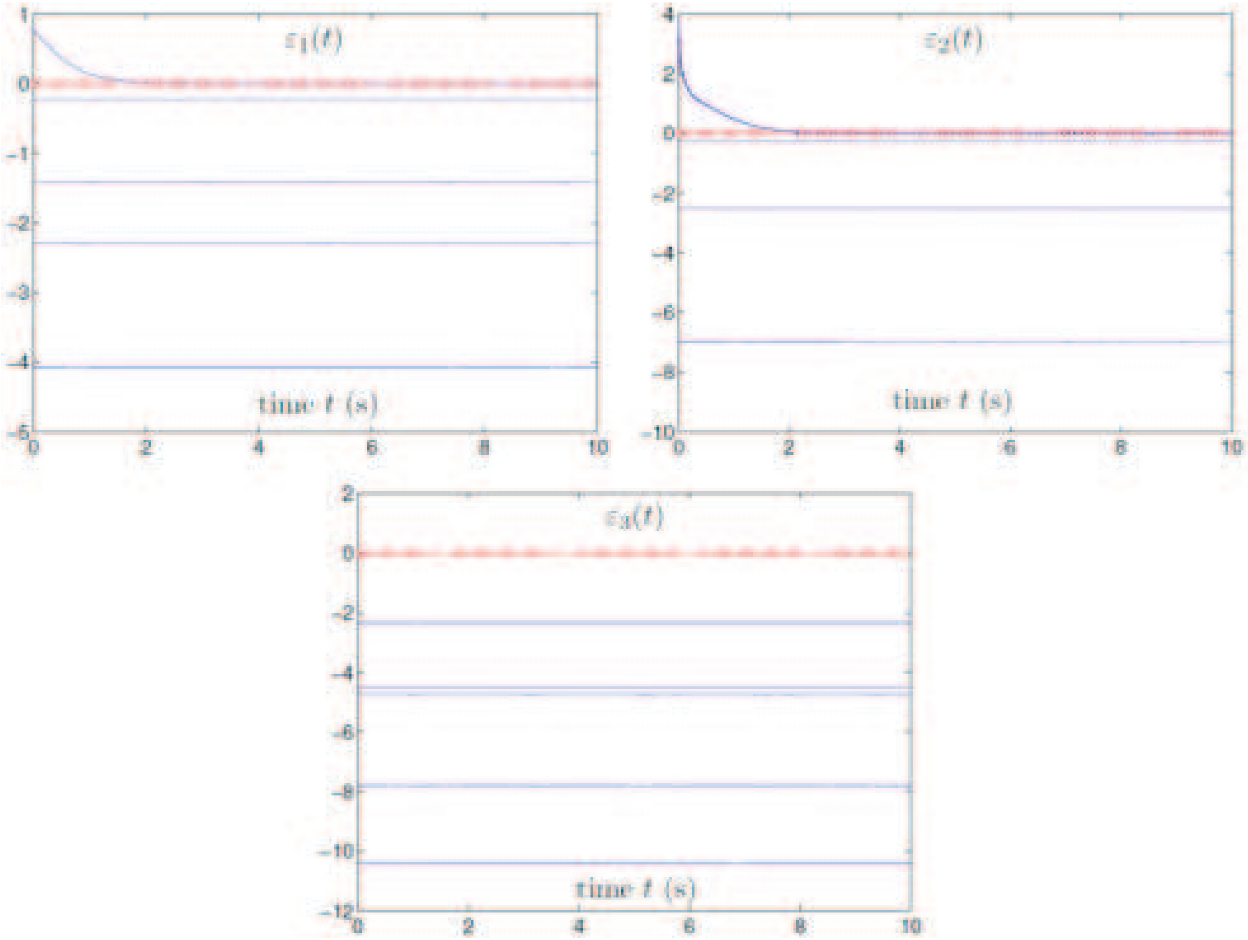


Figure 9. Profiles of $\varepsilon(t) = A(t)x(t) - b(t) \in R^3$ synthesized by ZNN model (10) with $\gamma = 1$ and the power-sigmoid activation function used for time-varying LMVI (3) solving.

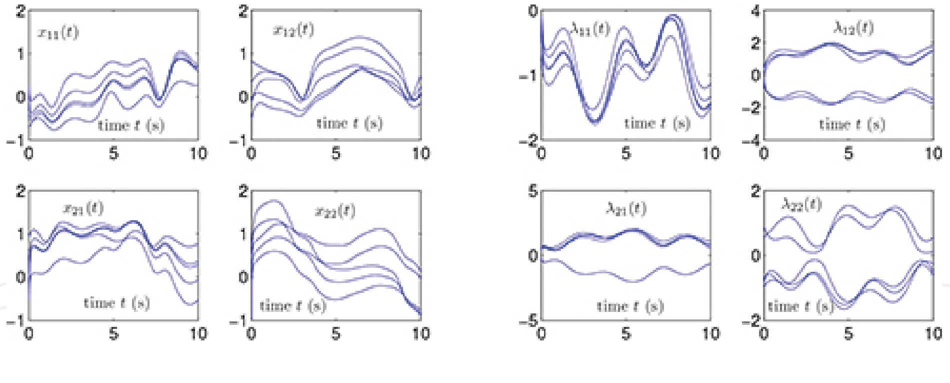
In summary, the above simulation results (i.e., **Figures 3** through **9**) have substantiated that the presented ZNN models (8) and (10) are both effective on time-varying LMVI solving.

Example 2 In the second example, the following smoothly time-varying coefficient matrices $A(t)$ and $B(t)$ of (15) are designed to test ZNN models (18) and (20):

$$A(t) = \begin{bmatrix} \sin(10t) & \cos(10t) \\ -\cos(10t) & \sin(10t) \end{bmatrix} \in R^{2\times 2} \text{ and}$$

$$B(t) = \begin{bmatrix} \cos(10t)+1 & \sin(10t)+1.5 \\ -\sin(10t)+1.5 & -\cos(10t)+1 \end{bmatrix} \in R^{2\times 2}.$$

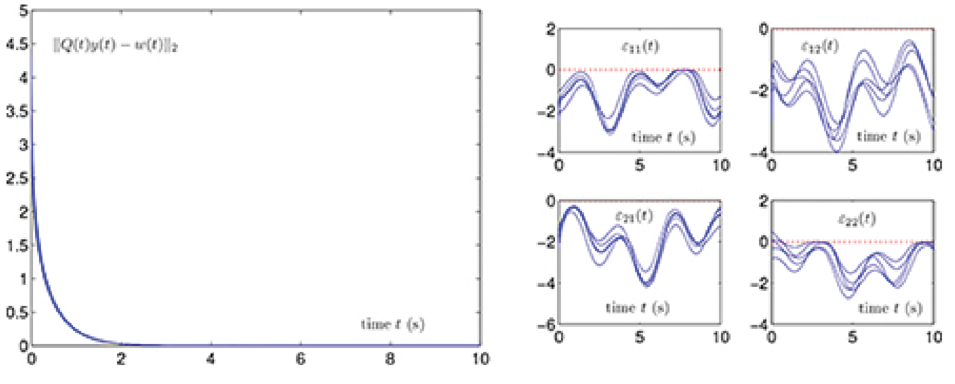
The corresponding simulation results are illustrated in **Figures 10** through **13**.



(a) State Trajectories of $X(t) \in R^{2\times 2}$

(b) State Trajectories of $\Lambda(t) \in R^{2\times 2}$

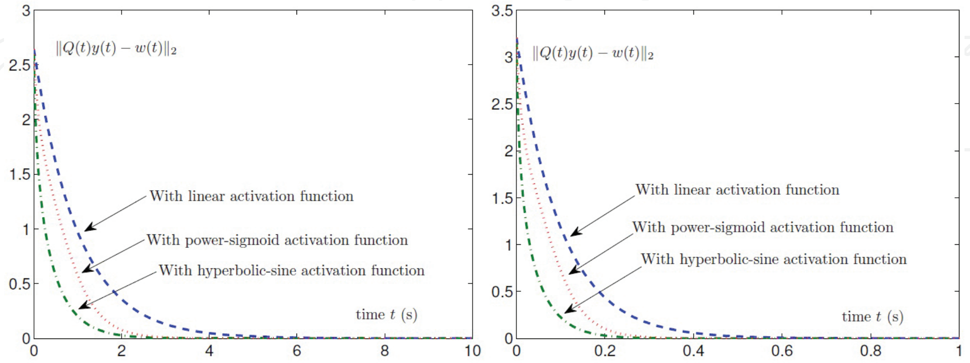
Figure 10. Neural states synthesized by ZNN model (18) with $\gamma = 1$ and the hyperbolic-sine activation function used for time-varying LMI (15) solving.



(a) Residual errors $\|Q(t)y(t) - w(t)\|_2$

(b) Profiles of $\varepsilon(t) \in R^{2\times 2}$

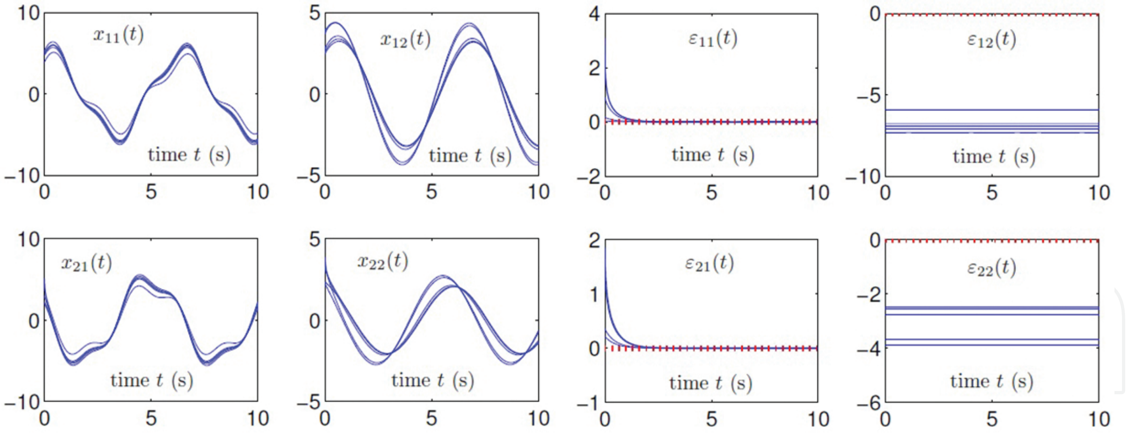
Figure 11. Profiles of residual errors $\| Q(t)y(t) - w(t) \|_2$ and $\varepsilon(t) = A(t)X(t) - B(t)$ synthesized by ZNN model (18) with $\gamma = 1$ and the hyperbolic-sine activation function used for time-varying LMI (15) solving.



(a) With $\gamma = 1$

(b) With $\gamma = 10$

Figure 12. Residual errors $\| Q(t)y(t) - w(t) \|_2$ of ZNN model (18) with γ fixed and different activation functions used for time-varying LMI (15) solving.



(a) State Trajectories of $X\ t\ \in R^{2\times}$ (b) State Trajectories of $\Lambda(t) \in R^{2\times 2}$

Figure 13. Simulation results synthesized by ZNN model (20) with $\gamma = 1$ and the hyperbolic-sine activation function used for time-varying LMI (15) solving.

On one hand, as synthesized by ZNN model (18) using $\gamma = 1$ and the hyperbolic-sine activation function, **Figure 10** shows the trajectories of $X(t)$ (being the first 4 elements of $y(t)$ in (18)) and $\Lambda(t)$ (being the rest elements of $y(t)$), which are time-varying. In addition, **Figure 11(a)** shows the characteristics of residual error $\| Q(t)y(t) - w(t) \|_2 = \| A(t)X(t) + \Lambda^2(t) - B(t) \|_F$ (with symbol $\| \cdot \|_F$ denoting the Frobenius norm of a matrix), from which we can observe that the residual errors of ZNN model (18) all converge to zero. This means that the solutions of $X(t)$ and $\Lambda(t)$ shown in **Figure 10** are the time-varying solutions of $A(t)X(t) + \Lambda^2(t) - B(t) = 0$. That is, $X(t)$ satisfies $A(t)X(t) = B(t) - \Lambda^2(t) \leq B(t)$, showing that such a solution is an exact time-varying solution of the time-varying LMI (15). For better understanding, **Figure 11(b)** shows the profiles of the testing error function $\varepsilon(t) = A(t)X(t) - B(t)$, from which we can observe that all the elements of $\varepsilon(t)$ are less than or equal to zero. These simulation results substantiate the efficacy of ZNN model (18) for time-varying LMI solving. Besides, **Figure 12** shows the simulation results synthesized by ZNN model (18) using different γ values and different activation functions. As seen from **Figure 12**, the residual errors all converge to zero, which means that the time-varying LMI (15) is solved successfully via ZNN model (18). Note that, as for ZNN model (18), its computational performance can be improved by increasing the γ value and choosing a suitable activation function (as shown in **Figure 12**).

On the other hand, as synthesized by ZNN model (20) using $\gamma = 1$ and the hyperbolic-sine activation function, **Figure 13** shows the related simulation results, where some elements of the initial state $X(0)$ are outside the initial solution set $S(0)$ of the time-varying LMI (15) while the others are inside $S(0)$. From **Figure 13(a)**, we can observe that the $X(t)$ trajectory of ZNN model (20) is time-varying. In addition, as shown in **Figure 13(b)**, the errors $\varepsilon_{11}(t)$ and $\varepsilon_{21}(t)$ (being the elements of the testing error function $\varepsilon(t) = A(t)X(t) - B(t)$) converge to zero, and the errors $\varepsilon_{12}(t)$ and $e_{22}(t)$ are always equal to $\varepsilon_{12}(0) < 0$ and $\varepsilon_{22}(0) < 0$. This means that the

$X(t)$ solution shown in **Figure 13(a)** is the time-varying solution of (15), i.e., $A(t)X(t) \leq B(t)$, thereby showing the efficacy of ZNN model (20). That is, ZNN model (20) generates an exact time-varying solution of the time-varying LMI (15). Besides, the investigations on the computational performance of (10) using different γ values and different activation functions are left to interested readers to complete as a topic of exercise.

In summary, the above simulation results (i.e., **Figures 10** through **13**) have substantiated that the presented ZNN models (18) and (20) are both effective on time-varying LMI solving.

6. Summary

In this chapter, by exploiting two design formulas (1) and (2), based on different ZFs (i.e., (4), (9), (16) and (19)), four different ZNN models (i.e., (8), (10), (18) and (20)) have been developed and investigated to solve the time-varying LMVI (3) and time-varying LMI (15). For such ZNN models, theoretical results and analyses have also been presented to show their computational performances. Simulation results with two illustrative examples have further substantiated the efficacy of the presented ZNN models for time-varying LMVI and LMI solving.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (with number 61473323 and 61403149), and also by the Scientific Research Funds of Huaqiao University.

Author details

Dongsheng Guo[1*], Laicheng Yan[1] and Yunong Zhang[2]

*Address all correspondence to: gdongsh2008@126.com; gdongsh@hqu.edu.cn

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

2 School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China

References

[1] Guo D, Zhang Y. Acceleration-level inequality-based MAN scheme for obstacle avoidance of redundant robot manipulators. IEEE Transactions on Industrial Electronics. 2014; 61:6903–6914. DOI: 10.1109/TIE.2014.2331036

[2] Guo D, Zhang Y. A new inequality-based obstacle-avoidance MVN scheme and its application to redundant robot manipulators. IEEE Transactions on Systems, Man, and Cybernetics, Part C. 2012; 42:1326–1340. DOI: 10.1109/TSMCC.2012.2183868

[3] Jing X. Robust adaptive learning of feedforward neural networks via LMI optimizations. Neural Networks. 2012; 31:33–45. DOI: 10.1016/j.neunet.2012.03.003

[4] Wang Z, Zhang H, Jiang B. LMI-based approach for global asymptotic stability analysis of recurrent neural networks with various delays and structures. IEEE Transactions on Neural Networks. 2011; 22:1032–1045. DOI: 10.1109/TNN.2011.2131679

[5] Kim E, Kang HJ, Park M. Numerical stability analysis of fuzzy control systems via quadratic programming and linear matrix inequalities. IEEE Transactions on Systems, Man, and Cybernetics, Part A. 1999; 29:333–346. DOI: 10.1109/3468.769752

[6] Xiao L, Zhang Y. Different Zhang functions resulting in different ZNN models demonstrated via time-varying linear matrix-vector inequalities solving. Neurocomputing. 2013; 121:140–149. DOI: 10.1016/j.neucom.2013.04.041

[7] Guo D, Zhang Y. ZNN for solving online time-varying linear matrix-vector inequality via equality conversion. Applied Mathematics and Computation. 2015; 259:327–338. DOI: 10.1016/j.amc.2015.02.060

[8] Guo D, Zhang Y. Zhang neural network for online solution of time-varying linear matrix inequality aided with an equality conversion. IEEE Transactions on Neural Networks and Learning Systems. 2014; 25:370–382. DOI: 10.1109/TNNLS.2013.2275011

[9] Guo D, Zhang Y. A new variant of the Zhang neural network for solving online time-varying linear inequalities. Proceedings of the Royal Society A. 2012; 468:2255–2271. DOI: 10.1098/rspa.2011.0668

[10] Lin C, Lai C, Huang T. A neural network for linear matrix inequality problems. IEEE Transactions on Neural Networks. 2000; 11:1078–1092. DOI: 10.1109/72.870041

[11] Cheng L, Hou ZG, Tan M, A simplified neural network for linear matrix inequality problems. Neural Processing Letters. 2009; 29:213–230. DOI: 10.1007/s11063-009-9105-5

[12] Yang K, Murty KG, Mangasarian OL. New iterative methods for linear inequalities. Journal of Optimization Theory and Applications. 1992; 72:163–185. DOI: 10.1007/BF00939954

[13] Cichocki A, Bargiela, A. Neural networks for solving linear inequality systems. Parallel Computing. 1997; 22:1455–1475. DOI: 10.1016/S0167-8191(96)00065-8

[14] Xia Y, Wang J, Hung DL. Recurrent neural networks for solving linear inequalities and equations. IEEE Transactions on Circuits and Systems - I. 1999; 46:452–462. DOI: 10.1109/81.754846

[15] Zhang Y. A set of nonlinear equations and inequalities arising in robotics and its online solution via a primal neural network. Neurocomputing. 2006; 70:513–524. DOI: 10.1016/j.neucom.2005.11.006

[16] Hu X. Dynamic system methods for solving mixed linear matrix inequalities and linear vector inequalities and equalities. Applied Mathematics and Computation. 2010; 216:1181–1193. DOI: 10.1016/j.amc.2010.02.010

[17] Xiao L, Zhang Y. Zhang neural network versus gradient neural network for solving time-varying linear inequalities. IEEE Transactions on Neural Networks. 2011; 22:1676–1684. DOI: 10.1109/TNN.2011.2163318

[18] Zhang Y, Guo D. Zhang Functions and Various Models. Heidelberg: Springer-Verlag; 2015. 236 p.

[19] Zhang Y, Yi C. Zhang Neural Networks and Neural-Dynamic Method. New York: Nova Science Publishers; 2011. 261 p.

[20] Zhang Y, Jiang D, Wang J. A recurrent neural network for solving Sylvester equation with time-varying coefficients. IEEE Transactions on Neural Networks. 2002; 13:1053–1063. DOI: 10.1109/TNN.2002.1031938

[21] Zhang Y, Ge SS. Design and analysis of a general recurrent neural network model for time-varying matrix inversion. IEEE Transactions on Neural Networks. 2005; 16:1477–1490. DOI: 10.1109/TNN.2005.857946

[22] Mead C. Analog VLSI and Neural Systems. Boston, USA: Addison-Wesley, Reading; 1989. 371 p.

[23] Li Z, Zhang Y. Improved Zhang neural network model and its solution of time-varying generalized linear matrix equations. Expert Systems with Applications. 2010; 37:7213–7218. DOI: 10.1016/j.eswa.2010.04.007

[24] Liu S, Trenkler G. Hadamard, Khatri-Rao, Kronecker and other matrix products. International Journal of Cooperative Information Systems. 2008; 4:160–177.