Discrete-time zeroing neural network for solving time-varying Sylvester-transpose matrix inequation via exp-aided conversion

Yunong Zhang[a], Yihong Ling[a], Shuai Li[b], Min Yang[a], Ning Tan[a,*]

[a] *School of Information Science and Technology, Sun Yat-sen University, Guangzhou, 510006, China*

[b] *Department of Electrical and Electronic Engineering, Swansea University, Wales, SA1 8EN, UK*

Abstract

Time-varying linear matrix equations and inequations have been widely studied in recent years. Time-varying Sylvester-transpose matrix inequation, which is an important variant, has not been fully investigated. Solving the time-varying problem in a constructive manner remains a challenge. This study considers an exp-aided conversion from time-varying linear matrix inequations to equations to solve the intractable problem. On the basis of zeroing neural network (ZNN) method, a continuous-time zeroing neural network (CTZNN) model is derived with the help of Kronecker product and vectorization technique. The convergence property of the model is analyzed. Two discrete-time ZNN models are obtained with the theoretical analyses of truncation error by using two Zhang et al.'s discretization (ZeaD) formulas with different precision to discretize the CTZNN model. The comparative numerical experiments are conducted for two discrete-time ZNN models, and the corresponding numerical results substantiate the convergence and effectiveness of two ZNN discrete-time models.

Keywords: Zeroing neural network, Time-varying Sylvester-transpose matrix inequation, ZeaD formula, Discrete-time model, Exp-aided conversion

[☆] This work is supported by the National Natural Science Foundation of China (grant 61976230), by Shenzhen Science and Technology Plan Project (grant JCYJ20170818154936083), by the Fundamental Research Funds for the Central Universities (grant 19lgpy221), and by the CCF-Tencent Open Fund.

[*] Corresponding author

E-mail Address: tann5@mail.sysu.edu.cn

1. Introduction

Linear matrix equations (LMEs) have been studied widely in recent years because of their important roles in some fields, such as image processing [1], linear system analysis [2], and control theory [3]. Specifically, Wei et al. [1] proposed a robust fast fusion of multiband images method with significantly decreased computational complexity through explicitly solving an underlying Sylvester equation. The minimum norm robust pole assignment problem for linear time-invariant systems was converted to an unconstrained minimization problem in [2] by using a Sylvester equation-based parametrization. A parametric design algorithm was designed in [3] for solving periodic Luenberger function observer problems based on two classes of discrete periodic Sylvester matrix equations. Several numerical methods/algorithms have been proposed and investigated to solve LMEs and their variants [4–8]. For example, Peng et al. [5] proposed an iteration method to obtain the symmetric solutions and optimal approximation solution of the matrix equation. In [7], a fuzzy Sylvester matrix equation was transferred into two crisp Sylvester matrix equations. Thus, the method can use numerical methods to solve the problem.

Time-varying problems have been attracting considerable attention due to their emergence in various applications [9–13]. Neural network methods [14–19] have also been considered valid alternatives to solve time-varying problems because of their efficiency, accuracy, and distributed fashion. Gradient-based neural network (GBNN) [14–16], which is a conventional scheme, has been fully studied to solve various important issues, such as time-varying matrix inversion [15] and LMEs [16]. However, the residual error always maintains a relatively large value when GBNN is used to solve a time-varying problem [17]. By contrast, zeroing neural network (ZNN) [17–19] perfectly exploits the time derivatives of time-varying parameters. The resulting residual error exponentially converges to zero when solving a time-varying problem. ZNN has also been widely used to systematically solve diverse time-varying issues, such as time-varying linear equations or inequalities [20–23], matrix inverse or generalized inverse [24, 25], and quadratic optimization [26].

Time-varying LMEs (TVLMEs), which combine the properties of time-varying problem and LMEs and are more difficult than time-invariant LMEs, have been also widely studied [20–22, 27–31]. Xiao [27] proposed a finite-time convergent ZNN method to solve a time-varying complex LME. The proposed method makes the error between the theoretical solution and the actual solution converge to zero in a finite time. Zhang et al. [31] proposed a varying-parameter ZNN that converges considerably faster than original ZNN to solve time-varying complex Sylvester equations. Compared with solving TVLMEs, solving time-varying linear matrix inequations (TVLMIs) is more challenging due to their intractable nature. Notably, linear ma-

trix inequality conventionally indicates a generalized inequality $A \succeq 0$, which denotes that A is positive semidefinite. For comparison, we define linear matrix inequation as a common inequality $A \geq 0$, which denotes that every element of A is nonnegative. Nearly all neural network methods are designed for solving TVLMEs but not for TVLMIs. A conversion inspired by a square-function-designed auxiliary matrix was considered by Guo et al. [32] to reformulate TVLMI as TVLME for solving TVLMI. The TVLME was solved by the ZNN model with a remarkable performance.

Time-varying Sylvester-transpose matrix inequation (TVSTMI), which has not been investigated before, is challenging because it contains not only the unknown matrix-valued variable but also its transpose. This study proposes another conversion inspired by an exponential-function-designed auxiliary matrix, which is different from the conversion proposed by Guo et al., to solve the problem. With the aid of ZNN design formula [19], a continuous-time zeroing neural network (CTZNN) model is derived. Along with the direction on improving continuous-time models to facilitate hardware implementation, a class of effective discretization formulas termed ZeaD formulas [33–36] is considered to discretize the CTZNN model. Thus, two discrete-time ZNN models with different accuracies are obtained to solve the TVSTMI through using Euler discretization formula [35] and six-instant ZeaD formula [36] to discretize the CTZNN model.

The remainder of the paper is organized as follows. In Section 2, the problem formulation of TVSTMI is described, and the exp-aided conversion from TVLMI to TVLME is investigated. In Section 3, the CTZNN model for solving problem is derived and proposed, and the theoretical analysis is also given to prove the convergence of the ZNN model. Section 4 presents two discrete-time ZNN models and their theoretical truncation errors. In Section 5, numerical experiments are conducted to compare the discrete-time ZNN models and a static model for solving TVSTMI, for substantiating the convergence and effectiveness of the former ones. Section 6 concludes this study with final remarks. The main contributions of this study are summarized below.

- Different from previous work that solves time-varying Sylvester matrix equations, this study first investigates the complicated TVSTMI with both the unknown matrix-valued variable and its transpose.

- A new conversion inspired by an exponential-function-designed auxiliary matrix is first proposed to transform TVLMI to TVLME. Then, the CTZNN model for solving TVSTMI with convergence analysis is obtained.

- By using two ZeaD formulas with different precision, the CTZNN model is

effectively discretized into two discrete-time ZNN models with truncation error analyses.

- Numerical experiments are conducted to compare the discrete-time ZNN models and the static model. The obtained relative results substantiate the superiority of two discrete-time ZNN models for solving TVSTMI.

2. Problem formulation and conversion

This section presents the formulation of TVSTMI, and displays a convenient conversion from time-varying matrix inequation to time-varying matrix equation for further derivation.

2.1. Problem formulation

First, the problem formulation of TVSTMI can be presented as

$$A(t)X(t)B(t) + C(t)X^{\mathrm{T}}(t)D(t) \leq E(t). \tag{1}$$

Notably, parameter matrices $A(t), B(t), C(t)$, $D(t)$, and $E(t) \in \mathbb{R}^{n\times n}$ are smoothly time-varying, and $X(t) \in \mathbb{R}^{n\times n}$ is the objective matrix to be solved for. This study aims to obtain a feasible $X(t)$ that satisfies the inequation. Compared with static Sylvester-transpose matrix inequation, TVSTMI (1) is a more challenging work. The experiential scheme for solving time-varying problems is that a continuous-time model is considered to be obtained by solving the derivative of $X(t)$ [15, 17] given necessary initial states [e.g., $X(t_0)$ with t_0 being the initial time]. Moreover, the continuous-time model is discretized effectively by ZeaD formulas [33–36] to construct a discrete-time model for easy hardware implementation.

Remark 1. *For randomly given parameter matrices $A(t)$, $B(t)$, $C(t)$, $D(t)$, and $E(t)$, the time-varying inequation may be unsolvable at some instants. Therefore, suitable parameter matrices are given to make the inequation solvable at any time instant t for obtaining a continuous solution $X(t)$. Thus, the hypothesis is made that the matrix $J(t) = B^{\mathrm{T}}(t) \otimes A(t) + (D^{\mathrm{T}}(t) \otimes C(t))P$ is always nonsingular with the help of a permutation matrix $P \in \mathbb{R}^{n^2\times n^2}$ [4]. This condition is also sufficient for the solvability of time-varying Sylvester-transpose matrix equation. The following discussion is oriented to the limited solvable problem.*

2.2. From inequation to equation with an auxiliary matrix

Numerous neural network methods, such as ZNN, perform directly well on TVLME. However, to the best of our knowledge, solving TVLMI is difficult. Hence, in this study, a new conversion from inequation to equation inspired by an auxiliary matrix is investigated to solve the challenging TVSTMI problem. For better understanding, $f(X(t),t)$ is defined to describe problem (1), which is formulated as

$$f(X(t),t) = A(t)X(t)B(t) + C(t)X^{\mathrm{T}}(t)D(t) - E(t) \leq \mathbf{0}. \tag{2}$$

An auxiliary matrix $Z(t) \in \mathbb{R}^{n\times n}$ with nonnegative elements, is introduced to convert TVLMI to TVLME. Then, a newly formed equation is obtained through applying the conversion with the aid of $Z(t)$:

$$A(t)X(t)B(t) + C(t)X^{\mathrm{T}}(t)D(t) - E(t) + Z(t) = \mathbf{0}. \tag{3}$$

For better constructing $Z(t)$, another simpler time-varying matrix $Y(t) \in \mathbb{R}^{n\times n}$ is introduced as follows:

$$Y(t) = \begin{bmatrix} y_{1,1}(t) & y_{1,2}(t) & \cdots & y_{1,n}(t) \\ y_{2,1}(t) & y_{2,2}(t) & \cdots & y_{2,n}(t) \\ \vdots & \vdots & \ddots & \vdots \\ y_{n,1}(t) & y_{n,2}(t) & \cdots & y_{n,n}(t) \end{bmatrix}.$$

Notably, every element $y_{i,j}(t)$ of $Y(t)$ has a domain being $\mathbb{R}$ and a continuous first-order time derivative with $i,j = 1,2,\ldots,n$. In addition, the image of the composite function $\exp(y_{i,j}(t))$ is $\mathbb{R}^+$, with $\exp(\cdot)$ being an exponential function. Accordingly, $Z(t)$ can be effortlessly constructed via $Y(t)$. Its expression is presented as follows:

$$Z(t) = \mathcal{F}(Y(t)) = \begin{bmatrix} \exp(y_{1,1}(t)) & \exp(y_{1,2}(t)) & \cdots & \exp(y_{1,n}(t)) \\ \exp(y_{2,1}(t)) & \exp(y_{2,2}(t)) & \cdots & \exp(y_{2,n}(t)) \\ \vdots & \vdots & \ddots & \vdots \\ \exp(y_{n,1}(t)) & \exp(y_{n,2}(t)) & \cdots & \exp(y_{n,n}(t)) \end{bmatrix},$$

where $\mathcal{F}(\cdot) : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ is a matrix-variable mapping. Evidently, every element of $Z(t)$, $z_{i,j}(t) = \exp(y_{i,j}(t))$, is a nonnegative variable. The time derivatives of $Y(t)$ and $Z(t)$ are investigated for further discussion. First, $\dot{Y}(t)$, which is the time derivative of $Y(t)$, can be easily presented as

$$\dot{Y}(t) = \begin{bmatrix} \dot{y}_{1,1}(t) & \dot{y}_{1,2}(t) & \cdots & \dot{y}_{1,n}(t) \\ \dot{y}_{2,1}(t) & \dot{y}_{2,2}(t) & \cdots & \dot{y}_{2,n}(t) \\ \vdots & \vdots & \ddots & \vdots \\ \dot{y}_{n,1}(t) & \dot{y}_{n,2}(t) & \cdots & \dot{y}_{n,n}(t) \end{bmatrix}.$$

In addition, $\dot{Z}(t)$, which is the time derivative of $Z(t)$ and expressed as a Hadamard product [37] of $\dot{Y}(t)$ and $Z(t)$, is obtained as

$$\dot{Z}(t) = \dot{Y}(t) \circ Z(t) = \begin{bmatrix} \dot{y}_{1,1}(t)\exp(y_{1,1}(t)) & \cdots & \dot{y}_{1,n}(t)\exp(y_{1,n}(t)) \\ \vdots & \ddots & \vdots \\ \dot{y}_{n,1}(t)\exp(y_{n,1}(t)) & \cdots & \dot{y}_{n,n}(t)\exp(y_{n,n}(t)) \end{bmatrix}.$$

For converted equation problem (3), $X(t)$ and $Y(t)$ [$Z(t)$ can be constructed by $Y(t)$] are considered as the objective matrix-valued variables. After solving time-varying equation (3), the solutions of $X(t)$ and $Y(t)$ are obtained. Time-varying inequation (2) is solved as well. Converted equation (3) can be reformulated as

$$f(X(t), t) = A(t)X(t)B(t) + C(t)X^{\mathrm{T}}(t)D(t) - E(t) = -Z(t) \leq \mathbf{0}.$$

In other words, $X(t)$ obtained by solving converted equation (3) is also a feasible solution of original inequation (2).

Remark 2. *The auxiliary matrix $Z(t)$ is introduced to convert time-varying inequation to time-varying equation, which transforms the problem to the easy one. Many feasible matrices can be used for constructing effective conversion from time-varying matrix inequation to time-varying matrix equation. Except for $Z(t)$, which is inspired by construction function $\mathcal{F}(Y(t)) = [\exp(y_{i,j})]|^{i,j=1,2,\ldots n}$, the auxiliary matrix can be constructed by other functions with properties consisting of domain being $\mathbb{R}$, image being $(0, +\infty)$ or $[0, +\infty)$, and differentiability. Two other construction functions are introduced here: the square function*

$$\mathcal{F}_1(Y(t)) = Y(t) \circ Y(t),$$

where $\circ$ is the Hadamard product operation, and the hyperbolic cosine function

$$\mathcal{F}_2(Y(t)) = \frac{\mathcal{F}(Y(t)) + \mathcal{F}(-Y(t))}{2} - \mathcal{I},$$

where $\mathcal{I}$ is the matrix of which each element is assigned as 1. Notably, different construction functions may cause different effects on numerical experiments.

3. Proposed CTZNN model

In this section, a CTZNN model is proposed to solve the converted time-varying Sylvester-transpose matrix equation (3). To apply the ZNN method [19, 27], a matrix-valued error function is defined as

$$\epsilon(t) = A(t)X(t)B(t) + C(t)X^{\mathrm{T}}(t)D(t) - E(t) + Z(t) \in \mathbb{R}^{n\times n}, \tag{4}$$

and its time derivative is obtained as

$$\begin{aligned}\dot{\epsilon}(t) = &\dot{A}(t)X(t)B(t) + A(t)\dot{X}(t)B(t) + A(t)X(t)\dot{B}(t) + \dot{C}(t)X^{\mathrm{T}}(t)D(t) \\ &+ C(t)\dot{X}^{\mathrm{T}}(t)D(t) + C(t)X^{\mathrm{T}}(t)\dot{D}(t) - \dot{E}(t) + \dot{Z}(t).\end{aligned} \tag{5}$$

The ZNN method is used to zero out every element of error function (4), and the formula $\dot{\epsilon}(t) = -\gamma\epsilon(t)$ is utilized to complete the convergence of error function (4), which is also presented as follows:

$$\dot{\epsilon}(t) = -\gamma(A(t)X(t)B(t) + C(t)X^{\mathrm{T}}(t)D(t) - E(t) + Z(t)). \tag{6}$$

Notably, γ is a predefined positive parameter for adjusting the convergence speed. In general, a large γ signifies a fast convergence speed. By substituting (5) into (6), the following equation is obtained:

$$\begin{aligned}A(t)\dot{X}(t)B(t) + C(t)\dot{X}^{\mathrm{T}}(t)D(t) + \dot{Z}(t) = &-\dot{A}(t)X(t)B(t) - A(t)X(t)\dot{B}(t) \\ &- \dot{C}(t)X^{\mathrm{T}}(t)D(t) - C(t)X^{\mathrm{T}}(t)\dot{D}(t) + \dot{E}(t) - \lambda\epsilon(t).\end{aligned} \tag{7}$$

In the differential equation (7), $\dot{X}(t)$, $\dot{X}^{\mathrm{T}}(t)$, and $\dot{Z}(t)$, which are the time derivatives of the objective matrix variables, are arranged in the left part Other variables are arranged in the right part. For the convenience of presentation, with $Q(t)$ defined as

$$Q(t) = -\dot{A}(t)X(t)B(t) - A(t)X(t)\dot{B}(t) - \dot{C}(t)X^{\mathrm{T}}(t)D(t) - C(t)X^{\mathrm{T}}(t)\dot{D}(t) + \dot{E}(t).$$

We define $U(t) = Q(t) - \lambda\epsilon(t)$, which consists of all matrix variables of the right part. Therefore, equation (7) is reformulated as

$$A(t)\dot{X}(t)B(t) + C(t)\dot{X}^{\mathrm{T}}(t)D(t) + \dot{Z}(t) = U(t).$$

The Kronecker product [4] and vectorization technique are used to solve the above-mentioned equation by reformulating it as

$$\begin{aligned}(B^{\mathrm{T}}(t) \otimes A(t))\mathrm{vec}(\dot{X}(t)) + (D^{\mathrm{T}}(t) \otimes C(t))\mathrm{vec}(\dot{X}^{\mathrm{T}}(t)) \\ + \mathrm{vec}(\dot{Z}(t)) = \mathrm{vec}(U(t)),\end{aligned} \tag{8}$$

where sign $\otimes$ denotes the Kronecker product operation, and $\mathrm{vec}(\cdot)$ is a vectorization operation on matrices. For example, given a matrix $M = [\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_n]$ with $\mathbf{m}_i$ being the ith column vector, where $i = 1, 2, \ldots, n$, we obtain $\mathrm{vec}(M) = [\mathbf{m}_1^{\mathrm{T}}, \mathbf{m}_2^{\mathrm{T}}, \ldots, \mathbf{m}_n^{\mathrm{T}}]^{\mathrm{T}}$. $\mathrm{vec}(\dot{Z}(t))$ can also be transformed into the form of a matrix

multiplying a vector to separate the time-derivative variable and the general variable, which is resolved as follows:

$$\mathrm{vec}(\dot{Z}(t)) = \mathrm{vec}(Z(t) \circ \dot{Y}(t)) = F(t)\mathrm{vec}(\dot{Y}(t)) \in \mathbb{R}^{n^2},$$

where $F(t) \in \mathbb{R}^{n^2 \times n^2}$ is a diagonal matrix consisting of all elements of $Z(t)$. $F(t)$ is formulated as

$$F(t) = \begin{bmatrix} Z_1'(t) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & Z_2'(t) & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & Z_n'(t) \end{bmatrix},$$

where $Z_j'(t) \in \mathbb{R}^{n \times n}$, with $j = 1, 2, \ldots, n$, is also a diagonal matrix presented as

$$Z_j'(t) = \begin{bmatrix} z_{1,j}(t) & 0 & \cdots & 0 \\ 0 & z_{2,j}(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & z_{n,j}(t) \end{bmatrix}.$$

Notably, $z_{i,j}(t)$, with $i, j = 1, 2, \ldots, n$, is the (i, j)th element of matrix $Z(t)$. By defining $G(t) = B^{\mathrm{T}}(t) \otimes A(t)$ and $H(t) = D^{\mathrm{T}}(t) \otimes C(t)$, equation (8) can be briefly written as

$$G(t)\mathrm{vec}(\dot{X}(t)) + H(t)\mathrm{vec}(\dot{X}^{\mathrm{T}}(t)) + F(t)\mathrm{vec}(\dot{Y}(t)) = \mathrm{vec}(U(t)).$$

To resolve $\mathrm{vec}(\dot{X}^{\mathrm{T}}(t))$, a useful lemma [4] is introduced as follows.

Lemma 1. *For any matrix $X \in \mathbb{R}^{m \times n}$,*

$$\mathrm{vec}(X^{\mathrm{T}}) = P_{(m,n)}\mathrm{vec}(X),$$

where $P_{(m,n)}$ is uniquely determined by the integers m and n.

Thus, with $P = P_{(n,n)}$, $\mathrm{vec}(\dot{X}^{\mathrm{T}}(t))$ is resolved into $P\mathrm{vec}(\dot{X}(t))$, and the following equation is obtained:

$$(G(t) + H(t)P)\mathrm{vec}(\dot{X}(t)) + F(t)\mathrm{vec}(\dot{Y}(t)) = \mathrm{vec}(U(t)). \tag{9}$$

For brief presentation, we define

$$\mathbf{x}(t) = \mathrm{vec}(X(t)), \quad \mathbf{y}(t) = \mathrm{vec}(Y(t)), \quad \dot{\mathbf{x}}(t) = \mathrm{vec}(\dot{X}(t)),$$
$$\dot{\mathbf{y}}(t) = \mathrm{vec}(\dot{Y}(t)), \quad \mathbf{u}(t) = \mathrm{vec}(U(t)), \quad J(t) = G(t) + H(t)P.$$

The simpler formulation of (9) is

$$J(t)\dot{\mathbf{x}}(t) + F(t)\dot{\mathbf{y}}(t) = \mathbf{u}(t),$$

which can be rewritten in the matrix-vector form:

$$\begin{bmatrix} J(t) & F(t) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{y}}(t) \end{bmatrix} = \mathbf{u}(t). \tag{10}$$

With $K(t) = [J(t) \quad F(t)] \in \mathbb{R}^{n^2 \times 2n^2}$ and $\mathbf{w}(t) = [\mathbf{x}^{\mathrm{T}}(t) \quad \mathbf{y}^{\mathrm{T}}(t)]^{\mathrm{T}} \in \mathbb{R}^{2n^2}$, the CTZNN model is obtained by solving a system of differential equations shown as follows:

$$\dot{\mathbf{w}}(t) = K^{\dagger}(t)\mathbf{u}(t), \tag{11}$$

where $\dot{\mathbf{w}}(t)$ is the time derivative of $\mathbf{w}(t)$, and $K^{\dagger}(t) = K^{\mathrm{T}}(t)(K(t)K^{\mathrm{T}}(t))^{-1}$ denotes the right pseudoinverse of $K(t)$ (i.e., Moore-Penrose inverse). In the online solving processing, a discretization formula can be used to obtain $X(t)$ at next time instant through computing $\dot{\mathbf{w}}(t)$ and $\mathbf{w}(t)$ at the current and previous time instants.

Remark 3. *As shown in formulation (10), column number is twice as many as row number for coefficient matrix $K(t) = [J(t) \quad F(t)]$ being of row full rank, which means that the solution to (10) is not unique. The right pseudoinverse is considered in this study to obtain one of feasible solutions of $\mathbf{x}(t)$ and $\mathbf{y}(t)$. Except for the CTZNN model, other models using different methods (e.g., other generalized inverses) to solve (10) are also welcome if the obtained solution still satisfies inequation (2).*

Notably, the CTZNN model based on ZNN has remarkable properties consisting of convergence and stability, which can guarantee the solution satisfying equation (3) after a short computing time. The convergence theorem of the CTZNN model is given, and the corresponding proof is presented to demonstrate the validity.

Theorem 1. *Let smoothly time-varying matrices $A(t), B(t), C(t)$, and $D(t)$ satisfy the hypothesis mentioned in Remark 1. CTZNN model (11) converges to one of the feasible solutions of inequation (2) as $t \to +\infty$.*

PROOF. Let $\hat{X}(t) \in \mathbb{R}^{n\times n}$ and $\hat{Z}(t) \in \mathbb{R}^{n\times n}$ be the solution obtained by the CTZNN model. The right pseudoinverse is used to obtain $\hat{X}(t)$ [the matrix form of the first half part of $\mathbf{w}(t)$]. Accordingly, the following expression is equivalent to equation (11):

$$\begin{aligned}&\dot{A}(t)\hat{X}(t)B(t) + A(t)\dot{\hat{X}}(t)B(t) + A(t)\hat{X}(t)\dot{B}(t) + \dot{C}(t)\hat{X}^{\mathrm{T}}(t)D(t)\\ &+C(t)\dot{\hat{X}}^{\mathrm{T}}(t)D(t) + C(t)\hat{X}^{\mathrm{T}}(t)\dot{D}(t) - \dot{E}(t) + \dot{\hat{Z}}(t)\\ &= -\lambda(A(t)\hat{X}(t)B(t) + C(t)\hat{X}^{\mathrm{T}}(t)D(t) - E(t) + \hat{Z}(t)).\end{aligned}$$

With $\hat{\epsilon}(t) = A(t)\hat{X}(t)B(t) + C(t)\hat{X}^{\mathrm{T}}(t)D(t) + \hat{Z}(t) - E(t) \in \mathbb{R}^{n\times n}$, the above-mentioned equation can be reformulated as

$$\dot{\hat{\epsilon}}(t) = -\lambda\hat{\epsilon}(t).$$

The compact form of this equation can be presented as

$$\dot{\hat{\epsilon}}_{i,j}(t) = -\lambda\hat{\epsilon}_{i,j}(t), \tag{12}$$

where $\hat{\epsilon}_{i,j}(t)$ is the (i,j)th element of $\hat{\epsilon}(t)$, with $i,j = 1,2\ldots,n$, and $\dot{\hat{\epsilon}}_{i,j}(t)$ is its time derivative. A Lyapunov function candidate is defined to analyze the subsystem (12):

$$l_{i,j}(t) = \hat{\epsilon}_{i,j}^2(t)/2.$$

The time derivative of this equation is obtained as

$$\dot{l}_{i,j}(t) = \dot{\hat{\epsilon}}_{i,j}(t)\hat{\epsilon}_{i,j}(t) = -\lambda\hat{\epsilon}_{i,j}^2(t).$$

With design parameter $\lambda > 0$, we have

$$\dot{l}_{i,j}(t) = \begin{cases} > 0, & \text{if} \quad \hat{\epsilon}_{i,j}(t) > 0,\\ = 0, & \text{if} \quad \hat{\epsilon}_{i,j}(t) = 0.\end{cases}$$

Lyapunov stability theory [19] posits that equilibrium point $l_{i,j}(t) = 0$ is globally asymptotically stable, that is, $l_{i,j}(t)$ globally converges to zero for any $i,j \in \{1,2,\ldots,n\}$. Therefore, $\hat{\epsilon}(t)$ globally converges to zero matrix as well [i.e., $\hat{\epsilon}(t) \to \mathbf{0}$, as $t \to +\infty$]. Thus, as time evolves, we have

$$A(t)\hat{X}(t)B(t) + C(t)\hat{X}^{\mathrm{T}}(t)D(t) - E(t) = \hat{\epsilon}(t) - \hat{Z}(t) \to -\hat{Z}(t).$$

As defined previously, $\hat{Z}(t) \geq \mathbf{0}$. Thus, we have

$$f(\hat{X}(t),t) = A(t)\hat{X}(t)B(t) + C(t)\hat{X}^{\mathrm{T}}(t)D(t) - E(t) \leq \mathbf{0}.$$

The aforementioned discussion substantiates that the solution $\hat{X}(t)$ satisfies inequation (2), as time $t \to +\infty$. Accordingly, CTZNN model (11) converges to one of the feasible solutions of inequation (2) as $t \to +\infty$, and the proof is completed. □

4. Two different discrete-time models based on ZeaD formulas

Two effective discrete-time ZNN models are proposed based on different ZeaD formulas [33–36] to verify the effectiveness of discretization of the CTZNN model and facilitate the implementation of digital circuits [17]. ZeaD formulas feature one-step-ahead approximation, that is, approximating $\dot{x}(t_k)$ by $x(t_{k+1})$, $x(t_k)$, $x(t_{k-1})$, …, where k denotes the updating index. The discrete-time models based on different ZeaD formulas have different accuracies in solving problem. Thus, the discrete-time ZNN models are numerically experimented to show their results of the accuracy holding on theories investigated by the previous ZeaD formulas work.

The Euler discretization formula [35] (also viewed as the first and simplest one of ZeaD formulas) using two instants is presented as

$$\dot{x}(t_k) = \frac{x(t_{k+1}) - x(t_k)}{\tau} + O(\tau), \tag{13}$$

where $t_k = k\tau, k = 0, 1, 2, \ldots$, with $\tau < 1$ representing the sampling period, and $O(\tau)$ denotes the truncation error. By using Euler discretization formula (13) to discretize the CTZNN model, the following equation is obtained:

$$\mathbf{w}(t_{k+1}) = \mathbf{w}(t_k) + \tau K^{\dagger}(t_k)\mathbf{u}(t_k) + \mathbf{O}(\tau^2),$$

where $\mathbf{O}(\tau^2) \in \mathbb{R}^{2n^2}$ denotes a vector with every element being $O(\tau^2)$. We define $\hat{\mathbf{u}}(t_k) = \tau\mathbf{u}(t_k) = \mathrm{vec}(\tau Q(t_k) - h\epsilon(t_k))$, where $h = \tau\lambda$, which is termed step size and must be fixed with a proper value [38]. The Euler discrete-time ZNN (EDZNN) model is formulated as

$$\mathbf{w}_{k+1} \doteq \mathbf{w}_k + K_k^{\dagger}\hat{\mathbf{u}}_k, \tag{14}$$

where $\mathbf{w}_k$ denotes $\mathbf{w}(t_k)$, and $\mathbf{w}_{k+1}$, $K_k^{\dagger}$, $\hat{\mathbf{u}}_k$ have similar denotations. In addition, $\doteq$ denotes the computational assignment operation.

A discretization formula proposed in [36] uses six instants to approximate the time derivative. The discretization formula with high precision, also termed six-instant ZeaD formula, is shown as

$$\dot{x}(t_k) = \frac{x(t_{k+1})}{2\tau} - \frac{5x(t_k)}{48\tau} - \frac{x(t_{k-1})}{4\tau} - \frac{x(t_{k-2})}{8\tau} - \frac{x(t_{k-3})}{12\tau} + \frac{x(t_{k-4})}{16\tau} + O(\tau^3). \tag{15}$$

Similarly, using six-instant ZeaD formula (15) to discretize the CTZNN model yields

$$\begin{aligned}\mathbf{w}(t_{k+1}) =& \frac{5}{24}\mathbf{w}(t_k) + \frac{1}{2}\mathbf{w}(t_{k-1}) + \frac{1}{4}\mathbf{w}(t_{k-2}) + \frac{1}{6}\mathbf{w}(t_{k-3}) - \frac{1}{8}\mathbf{w}(t_{k-4}) \\ &+ 2K^{\dagger}(t_k)\hat{\mathbf{u}}(t_k) + \mathbf{O}(\tau^4),\end{aligned}$$

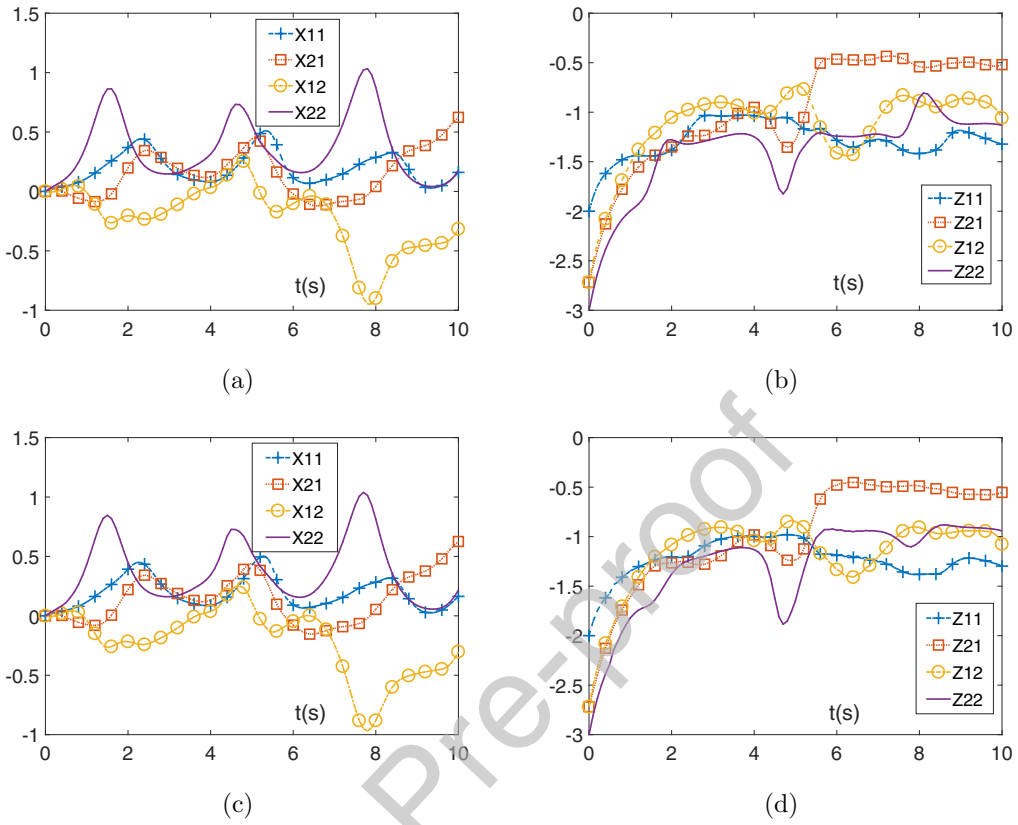


Figure 1: Profiles of elements of solution $\{X_k\}$ and objective function $\{f_k\}$ with sampling period $\tau = 0.1$ s, where (a) and (b) are synthesized by EDZNN model while (c) and (d) are synthesized by 6IDZNN model.

and the six-instant discrete-time ZNN (6IDZNN) model is formulated as

$$\mathbf{w}_{k+1} \doteq \frac{5}{24}\mathbf{w}_k + \frac{1}{2}\mathbf{w}_{k-1} + \frac{1}{4}\mathbf{w}_{k-2} + \frac{1}{6}\mathbf{w}_{k-3} - \frac{1}{8}\mathbf{w}_{k-4} + 2K_k^{\dagger}\hat{\mathbf{u}}_k, \tag{16}$$

where the notations (e.g., $\mathbf{w}_k$) have the same meanings as those in EDZNN (14).

Remark 4. *With step size h set in the effective domain and in accordance with [35] and [36], the EDZNN and 6IDZNN models are proven to be zero-stable, consistent, and convergent with the orders of truncation error being $\mathbf{O}(\tau^2)$ and $\mathbf{O}(\tau^4)$, respectively.*

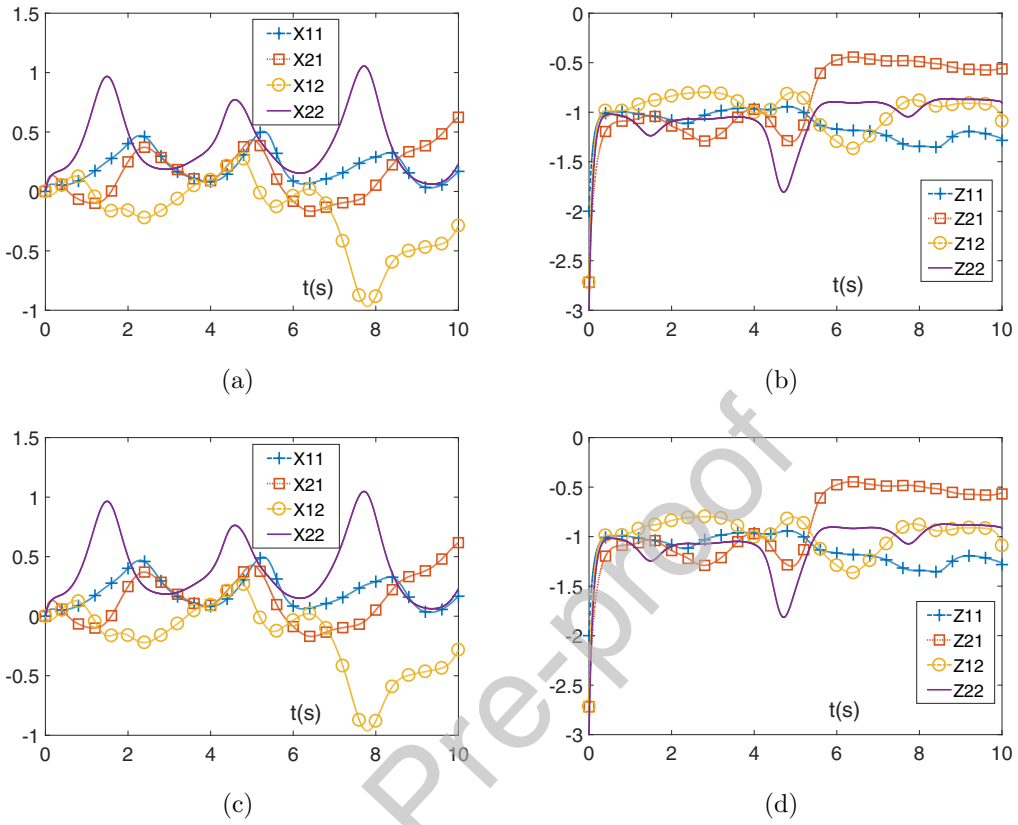


Figure 2: Profiles of elements of solution $\{X_k\}$ and objective function $\{f_k\}$ with sampling period $\tau = 0.01$ s, where (a) and (b) are synthesized by EDZNN model while (c) and (d) are synthesized by 6IDZNN model.

With residual error defined as

$$\|A_k X_k B_k + C_k X_k^{\mathrm{T}} D_k - E_k + Z_k\|_{\mathrm{F}},$$

the maximal steady-state residual error, abbreviated as MSSRE, is defined as

$$\lim_{k\to+\infty} \sup \|A_k X_k B_k + C_k X_k^{\mathrm{T}} D_k - E_k + Z_k\|_{\mathrm{F}},$$

where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm. MSSRE is the largest value of residual error after a model converges.

Theorem 2. *The MSSRE synthesized by the EDZNN model is of order* $O(\tau^2)$.

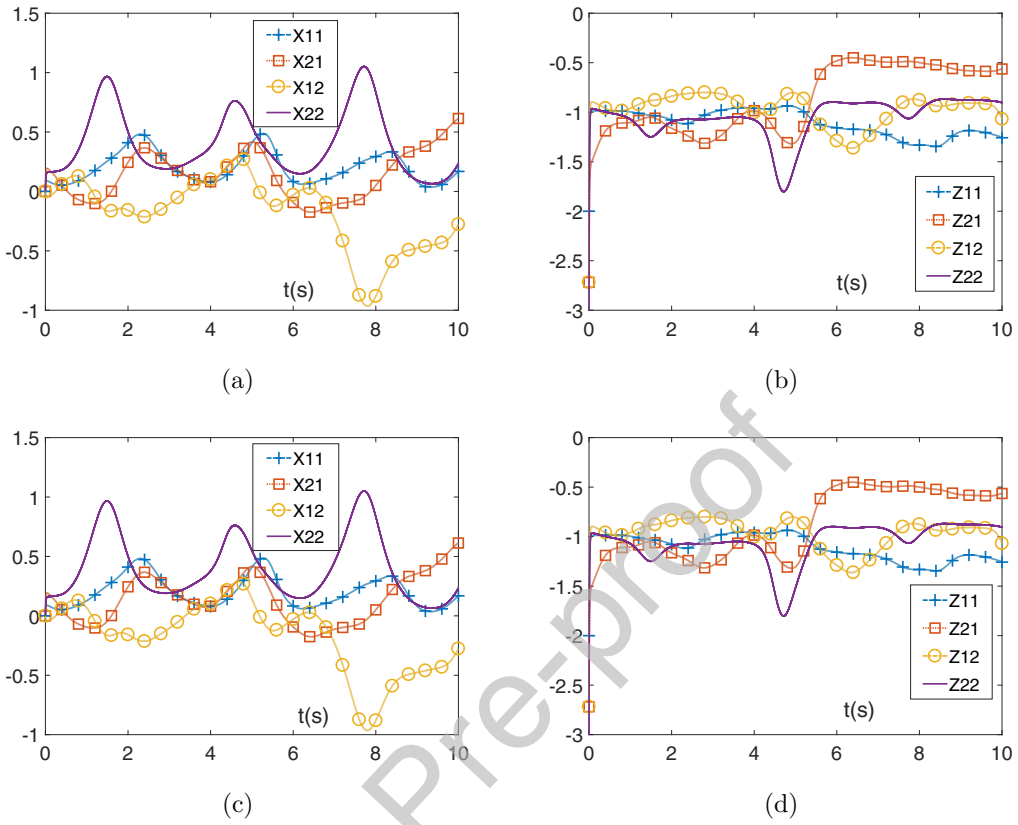


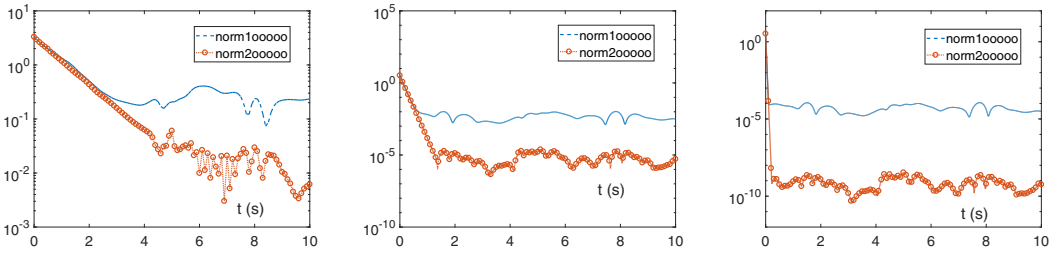
Figure 3: Profiles of elements of solution $\{X_k\}$ and objective function $\{f_k\}$ with sampling period $\tau = 0.001$ s, where (a) and (b) are synthesized by EDZNN model while (c) and (d) are synthesized by 6IDZNN model.

PROOF. Let X_k^* and Y_k^* be the theoretical solution of the CTZNN model, with $Z_k^* = \exp(Y_k^*)$, the following equation holds true:

$$\lim_{k\to+\infty} \sup \|A_k X_k^* B_k + C_k X_k^{*\mathrm{T}} D_k - E_k + Z_k^*\|_{\mathrm{F}} = 0.$$

Let X_k and Y_k be the solution obtained by the EDZNN model. We define that $X_k = X_k^* + \mathcal{O}(\tau^2)$ and $Y_k = Y_k^* + \mathcal{O}(\tau^2)$, where $\mathcal{O}(\tau^2) \in \mathbb{R}^{n\times n}$ consists of elements being $O(\tau^2)$. With $Z_k = \mathcal{F}(Y_k)$, the following equation is obtained:

$$\begin{aligned}
&\lim_{k\to+\infty} \sup \|A_k X_k B_k + C_k X_k^{\mathrm{T}} D_k - E_k + Z_k\|_{\mathrm{F}} \\
&= \lim_{k\to+\infty} \sup \|A_k (X_k^* + \mathcal{O}(\tau^2)) B_k + C_k (X_k^{*\mathrm{T}} + \mathcal{O}(\tau^2)) D_k - E_k + \mathcal{F}(Y_k^* + \mathcal{O}(\tau^2))\|_{\mathrm{F}}.
\end{aligned} \tag{17}$$



(a) (b) (c)

Figure 4: Residual errors synthesized by EDZNN and 6IDZNN models with different values of sampling period: (a) $\tau = 0.1$ s; (b) $\tau = 0.01$ s; (c) $\tau = 0.001$ s.

Notably, the Taylor expansion [39] of $\exp(x)$ is read as $\exp(x) = 1 + x + O(x^2)$. We have $\mathcal{F}(\mathcal{O}(\tau^2)) = \mathcal{I} + \mathcal{O}(\tau^2) + \mathcal{O}(\tau^4)$, where $\mathcal{I} \in \mathbb{R}^{n\times n}$ denotes the matrix with each element being 1. With $\mathcal{O}(\tau^4)$ absorbed into $\mathcal{O}(\tau^2)$, we have $\mathcal{F}(\mathcal{O}(\tau^2)) = \mathcal{I} + \mathcal{O}(\tau^2)$. Moreover, $\mathcal{F}(Y_k^* + \mathcal{O}(\tau^2))$ has the property of being resolved into $\mathcal{F}(Y_k^*) \circ \mathcal{F}(\mathcal{O}(\tau^2))$. Accordingly, the following transformation is obtained:

$$\mathcal{F}(Y_k^* + \mathcal{O}(\tau^2)) = \mathcal{F}(Y_k^*) \circ \mathcal{F}(\mathcal{O}(\tau^2)) = Z_k^* \circ (\mathcal{I} + \mathcal{O}(\tau^2)) = Z_k^* + Z_k^* \circ \mathcal{O}(\tau^2).$$

With $\lim_{k\to+\infty} \sup\|\cdot\|$ defined as $\|\cdot\|_{\mathrm{s}}$, the MSSRE (17) is deduced as

$$\begin{aligned}
&\|A_k X_k B_k + C_k X_k^{\mathrm{T}} D_k - E_k + Z_k\|_{\mathrm{s}} \\
\leq &\|A_k \mathcal{O}(\tau^2) B_k + C_k \mathcal{O}(\tau^2) D_k + Z_k^* \circ \mathcal{O}(\tau^2)\|_{\mathrm{s}} + \|A_k X_k^* B_k + C_k X_k^{*\mathrm{T}} D_k - E_k + Z_k^*\|_{\mathrm{s}} \\
= &\|A_k \mathcal{O}(\tau^2) B_k + C_k \mathcal{O}(\tau^2) D_k + Z_k^* \circ \mathcal{O}(\tau^2)\|_{\mathrm{s}} \\
\leq &\|A_k\|_{\mathrm{s}} \|\mathcal{O}(\tau^2)\|_{\mathrm{s}} \|B_k\|_{\mathrm{s}} + \|C_k\|_{\mathrm{s}} \|\mathcal{O}(\tau^2)\|_{\mathrm{s}} \|D_k\|_{\mathrm{s}} + \|Z_k^* \circ \mathcal{O}(\tau^2)\|_{\mathrm{s}} \\
\leq &(\|A_k\|_{\mathrm{s}} \|B_k\|_{\mathrm{s}} + \|C_k\|_{\mathrm{s}} \|D_k\|_{\mathrm{s}} + \|Z_k^*\|_{\mathrm{s}}) O(\tau^2).
\end{aligned}$$

The matrices A_k, B_k, C_k, and D_k are uniformly norm bounded. Notably, $\|Z_k^*\|_{\mathrm{s}} \leq \max(\|Z^*(t)\|_{\mathrm{F}})$, $t \in [0, +\infty)$, where $\|Z^*(t)\|_{\mathrm{F}}$ is free from τ. Thus, the MSSRE synthesized by the EDZNN model (14) is $O(\tau^2)$. The proof is thus completed. □

Theorem 3. *The MSSRE synthesized by the 6IDZNN model is of order $O(\tau^4)$.*

Proof. The proof can be generalized from the proof of Theorem 2. □

5. Numerical experiments and verification

In this section, two comparative numerical experiments are given to verify the convergence, effectiveness, and accuracy of two discrete-time ZNN models. For guaranteeing the solution of problem existing, the hypothesis is considered to design the

Table 1: Detailed information of residual errors synthesized by EDZNN and 6IDZNN models with sampling period $\tau = 0.1$, 0.01, or 0.001 s, where convergence time is obtained as the time instant, from initial time 0 s, when the monotonic decrease of residual error ceases.

	τ (s)	MSSRE	Convergence time (s)	MSSRE pattern
EDZNN	0.1	4.066×10^{-1}	3.800	$O(\tau^2)$
	0.01	1.045×10^{-2}	1.240	
	0.001	1.146×10^{-4}	0.142	
6IDZNN	0.1	6.037×10^{-2}	4.600	$O(\tau^4)$
	0.01	2.400×10^{-5}	1.370	
	0.001	3.407×10^{-9}	0.227	

parameter matrices that $J(t) = B^{\mathrm{T}}(t) \otimes A(t) + (D^{\mathrm{T}}(t) \otimes C(t))P$ is not singular. Accordingly, parameter matrices $A(t)$, $B(t)$, $C(t)$, $D(t)$, and $E(t)$ are presented as follows:

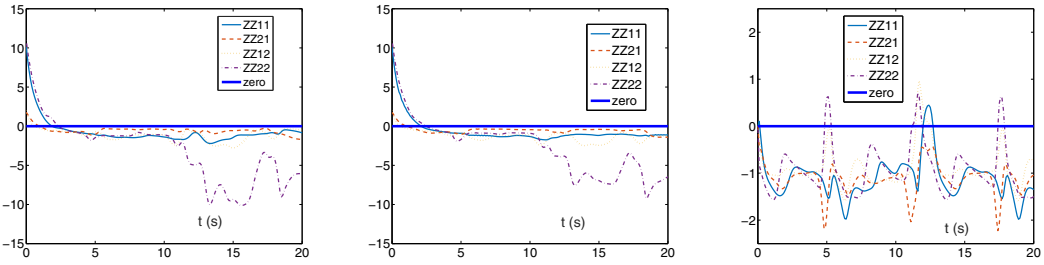
$$A(t) = \begin{bmatrix} \exp(\cos(t)) + 1 & \sin(t) \\ \cos(t) & \exp(\sin(t)) + 1 \end{bmatrix}, \quad B(t) = \begin{bmatrix} \sin(2t) + 2 & \sin(t) \\ \cos(t) & \cos(2t) + 2 \end{bmatrix},$$

$$C(t) = \begin{bmatrix} \exp(\sqrt{2}\sin(t + \frac{\pi}{4})) + 2 & \frac{1}{2}\sin(t) \\ \frac{1}{2}\cos(t) & \exp(\sqrt{2}\sin(\frac{\pi}{4} - t)) + 1 \end{bmatrix},$$

$$D(t) = \begin{bmatrix} \sin(2t) + 1 & \cos(t) \\ \sin(t) & \cos(2t) + 1 \end{bmatrix}, \quad E(t) = \begin{bmatrix} \exp(\sin(t)) + 1 & \exp(\cos(\frac{t}{2})) \\ \exp(\cos(\frac{t}{2})) & \exp(\sin(t)) + 2 \end{bmatrix}.$$

5.1. Simulation of discrete-time ZNN models

The initial states are set without loss of generality to $X_0 = [0, 0; 0, 0]$ and $Y_0 = [0, 0; 0, 0]$ to perform the numerical experiments for discrete-time ZNN models. For desirable convergence of two discrete-time models, the step size h must be set in effective domain, and $h = 0.1$ in this study. Besides, the value of sampling period τ is set to 0.1, 0.01, or 0.001 s (i.e., the value of λ is set to 1, 10, or 100) to substantiate MSSRE conforming to the correct order. The experimental duration is set to 10 s.



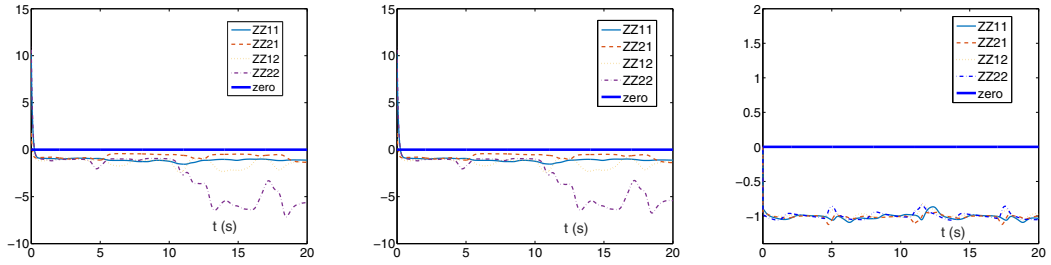
(a) (b) (c)

Figure 5: Profiles of elements of objective function $\{f_k\}$ synthesized by three models with sampling period $\tau = 0.1$ s by (a) EDZNN model, (b) 6IDZNN model, and (c) static model.

The numerical results synthesized by the EDZNN and 6IDZNN models are displayed in Fig. 1 through Fig. 4 and Table 1. As shown in Fig. 1 through Fig. 3, $\{X_k\}$ [i.e., $\{X(t)\}$, with $t = k\tau$ and $k = 1, 2 \ldots$] synthesized by two discrete-time ZNN models converge to slightly different values, and each element of $\{f_k\}$ is lower than zero, which substantiates the convergence and effectiveness of two discrete-time ZNN models for solving TVSTMI. The comparative results of residual errors are shown in Fig. 4 for verifying the accuracy of two discrete-time ZNN models. Detailed information of residual errors is presented in Table 1. From Fig. 4 and Table 1, we can conclude that, with a small sampling period τ, that is, a large design parameter λ, indicates that the convergence time of residual error is correspondingly less. After the curves of numerical results are stable, the MSSREs synthesized by the EDZNN and 6IDZNN models are 4.066×10^{-1} and 6.037×10^{-2} with $\tau = 0.1$ s, 1.045×10^{-2} and 2.400×10^{-5} with $\tau = 0.01$ s, and 1.146×10^{-4} and 3.407×10^{-9} with $\tau = 0.001$ s, respectively. The MSSREs of the EDZNN model converge with order $O(\tau^2)$, and those of the 6IDZNN model converge with order $O(\tau^4)$. The above-mentioned analyses substantiate the accuracy of two discrete-time ZNN models conforming to Theorem 2, Theorem 3, and the ZeaD formula theories [35, 36]. That is, the 6IDZNN model with small MSSREs produces the highly similar results to the theoretical results produced by the CTZNN model compared with the EDZNN model. Thus the 6IDZNN model is likely to meet the requirement.

5.2. Comparisons between discrete-time models and static model

The initial states are set to $X_0 = [1, 0; 0, 1]$, $Y_0 = [0, 0; 0, 0]$, and parameter $h = 0.1$ to perform the numerical experiments for discrete-time ZNN models differently. For comparison, we consider a static model for solving TVSTMI. First, when $t = t_k$ with



(a) (b) (c)

Figure 6: Profiles of elements of objective function $\{f_k\}$ synthesized by three models with sampling period $\tau = 0.01$ s by (a) EDZNN model, (b) 6IDZNN model, and (c) static model.

k being the updating step and $t_k = k\tau$, we have the following equation with constant coefficients:

$$A(t_k)\overline{X}_k B(t_k) + C(t_k)\overline{X}_k^{\mathrm{T}} D(t_k) - E(t_k) + S = \mathbf{0},$$

where $\overline{X}_k$ is the objective solution and S is the slack matrix variable with every element being positive or zero. Corresponding to initial state $Y_0 = [0, 0; 0, 0]$ of discrete-time models, S is set as $\mathcal{F}(Y_0)$, that is, $s_{i,j} = 1$. Then, $\overline{X}_k$ can be obtained through solving the time-invariant Sylvester-transpose matrix equation as follows:

$$A(t_k)\overline{X}_k B(t_k) + C(t_k)\overline{X}_k^{\mathrm{T}} D(t_k) + G(t_k) = \mathbf{0},$$

where $G(t_k) = -E(t_k) + S$. The original problem (1) is time-varying. Thus, when the theoretical solution $\overline{X}_k$ of the above-mentioned equation is computed, time variable t varies and is within $(t_k, +\infty)$. For the consistency of sampling period, solution $\overline{X}_k$ should be verified by TVSTMI at $t = (k+1)\tau$, that is, $X_{k+1} = \overline{X}_k$, we obtain the static model for TVSTMI (1) with a defect of delay.

The value of sampling period τ is set to 0.1 or 0.01 s (i.e., the value of λ is set as 1 or 10). The experimental duration is set to 20 s. The numerical results synthesized by the EDZNN, 6IDZNN and static models are displayed in Figs. 5 and 6. As shown in Fig. 5, one can see that the values of elements of the objective function synthesised by the discrete-time ZNN models decrease in the first few seconds and remain below-zero after 2.4 s. However, some values of elements of the objective function synthesized by the static model are larger than zero periodically. This finding implies the static model is inappropriate for TVSTMI with $\tau = 0.1$ s. As shown in Fig. 6, the values of objective function synthesized by discrete-time models remain below-zero after a short time and have fast convergence speed. In addition, the values of elements of objective function synthesized by the static model are below-zero. That is, the static

model is not particularly suitable for the time-varying problems especially when each update takes a long computation time. The discrete-time ZNN models effectively overcome the delayed phenomenon usually existing in conventional static methods for solving time-varying problems, which has been considerably studied [17, 40].

6. Conclusions

In this paper, we have presented the problem formulation of TVSTMI. The exp-aided conversion has been proposed to transform TVLMI to TVLME to solve the TVSTMI. TVLME can be handled effectively by neural network methods. Then, the CTZNN model based on the ZNN method has been derived with the aid of Kronecker product and vectorization technique. The convergence of the CTZNN model has been analyzed as well. Moreover, the EDZNN and 6IDZNN models have been derived using Euler discretization formula and six-instant ZeaD formula to discretize the CTZNN model. The MSSREs of two ZNN discrete-time models have been proven with the orders being $O(\tau^2)$ and $O(\tau^4)$, respectively. Furthermore, illustrative and comparative examples have been provided for performing the numerical experiments of two discrete-time ZNN models and the static model. The corresponding numerical results have substantiated the convergence, effectiveness, and superiority of two discrete-time ZNN models. The derivation and analyses have been discussed under a relatively strong condition of non-singularity (Remark 1). Thus, new results may be inappropriate for some singular situations. This topic should be given additional discussions before using the proposed method. Thus, solving the problem under weak conditions is one of our future directions.

Reference

References

[1] Q. Wei, N. Dobigeon, J. Tourneret, J. Bioucas-Dias, S. Godsill, R-FUSE: Robust fast fusion of multiband images based on solving a Sylvester equation, IEEE Signal Processing Letters 23 (11) (2016) 1632–1636.

[2] A. Varga, Robust pole assignment via sylvester equation based state feedback parametrization, in: CACSD. Conference Proceedings. IEEE International Symposium on Computer-Aided Control System Design, 2000, pp. 13–18.

[3] L. Lv, L. Zhang, On the periodic Sylvester equations and their applications in periodic luenberger observers design, Journal of the Franklin Institute 353 (5) (2016) 1005–1018.

[4] M. Hajarian, Matrix iterative methods for solving the Sylvester-transpose and periodic Sylvester matrix equations, Journal of the Franklin Institute 350 (10) (2013) 3328–3341.

[5] Y. Peng, X. Hu, L. Zhang, An iteration method for the symmetric solutions and the optimal approximation solution of the matrix equation AXB=C, Applied Mathematics and Computation 160 (3) (2005) 763–777.

[6] N. Huang, C. Ma, The modified conjugate gradient methods for solving a class of generalized coupled Sylvester-transpose matrix equations, Computers and Mathematics with Applications 67 (8) (2014) 1545–1558.

[7] Q. He, L. Hou, J. Zhou, The solution of fuzzy Sylvester matrix equation, Soft Computing 22 (19) (2018) 6515–6523.

[8] H. Zhang, H. Yin, New proof of the gradient-based iterative algorithm for the Sylvester conjugate matrix equation, Computers and Mathematics with Applications 74 (12) (2017) 3260–3270.

[9] L. Jin, Y. Zhang, Continuous and discrete Zhang dynamics for real-time varying nonlinear optimization, Numerical Algorithms 73 (1) (2016) 115–140.

[10] D. Chen, Y. Zhang, Jerk-level synchronous repetitive motion scheme with gradient-type and zeroing-type dynamics algorithms applied to dual-arm redundant robot system control, International Journal of Systems Science 48 (7) (2017) 1–15.

[11] L. Jin, S. Li, L. Xiao, R. Lu, B. Liao, Cooperative motion generation in a distributed network of redundant robot manipulators with noises, IEEE Transactions on Systems, Man, and Cybernetics: Systems 48 (10) (2018) 1715–1724.

[12] W. He, H. Huang, S. S. Ge, Adaptive neural network control of a robotic manipulator with time-varying output constraints, IEEE Transactions on Cybernetics 47 (10) (2017) 3136–3147.

[13] L. Jin, S. Li, X. Luo, Y. Li, B. Qin, Neural dynamics for cooperative control of redundant robot manipulators, IEEE Transactions on Industrial Informatics 14 (9) (2018) 3812–3821.

[14] L. Jin, S. Li, B. Hu, RNN models for dynamic matrix inversion: A control-theoretical perspective, IEEE Transactions on Industrial Informatics 14 (1) (2018) 189–199.

[15] Y. Zhang, Y. Shi, K. Chen, C. Wang, Global exponential convergence and stability of gradient-based neural network for online matrix inversion, Applied Mathematics and Computation 215 (3) (2009) 1301–1306.

[16] P. S. Stanimirović, M. D. Petković, Gradient neural dynamics for solving matrix equations and their applications, Neurocomputing 306 (2018) 200–212.

[17] L. Jin, S. Li, B. Liao, Z. Zhang, Zeroing neural networks: A survey, Neurocomputing 267 (2017) 597–604.

[18] Y. Zhang, D. Jiang, J. Wang, A recurrent neural network for solving Sylvester equation with time-varying coefficients, IEEE Transactions on Neural Networks 13 (5) (2002) 1053–1063.

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

[20] Y. Shen, P. Miao, Y. Huang, Y. Shen, Finite-time stability and its application for solving time-varying Sylvester equation by recurrent neural network, Neural Processing Letters 42 (3) (2015) 763–784.

[21] L. Xiao, A finite-time recurrent neural network for solving online time-varying Sylvester matrix equation based on a new evolution formula, Nonlinear Dynamics 90 (3) (2017) 1581–1591.

[22] L. Xiao, B. Liao, S. Li, K. Chen, Nonlinear recurrent neural networks for finite-time solution of general time-varying linear matrix equations, Neural Networks 98 (2018) 102–113.

[23] L. Xiao, Y. Zhang, Different Zhang functions resulting in different ZNN models demonstrated via time-varying linear matrix-vector inequalities solving, Neurocomputing 121 (2013) 140–149.

[24] S. Qiao, X. Wang, Y. Wei, Two finite-time convergent Zhang neural network models for time-varying complex matrix Drazin inverse, Linear Algebra and its Applications 542 (2018) 101–117.

[25] M. D. Petković, P. S. Stanimirović, V. N. Katsikis, Modified discrete iterations for computing the inverse and pseudoinverse of the time-varying matrix, Neurocomputing 289 (2018) 155–165.

[26] Y. Zhang, G. Ruan, K. Li, Y. Yang, Robustness analysis of the Zhang neural network for online time-varying quadratic optimization, Journal of Physics A Mathematical General 43 (24) (2010) 245202.

[27] L. Xiao, A finite-time convergent neural dynamics for online solution of time-varying linear complex matrix equation, Neurocomputing 167 (2015) 254–259.

[28] Z. Li, Y. Zhang, Improved Zhang neural network model and its solution of time-varying generalized linear matrix equations, Expert Systems with Applications 37 (10) (2010) 7213–7218.

[29] L. Xiao, Y. Zhang, From different Zhang functions to various ZNN models accelerated to finite-time convergence for time-varying linear matrix equation, Neural Processing Letters 39 (3) (2014) 309–326.

[30] C. Yi, Y. Zhang, D. Guo, A new type of recurrent neural networks for real-time solution of Lyapunov equation with time-varying coefficient matrices, Mathematics and Computers in Simulation 92 (2013) 40–52.

[31] Z. Zhang, L. Zheng, A complex varying-parameter convergent-differential neural-network for solving online time-varying complex sylvester equation, IEEE Transactions on Cybernetics 49 (10) (2019) 3627–3639.

[32] D. Guo, Y. Zhang, 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 25 (2) (2014) 370–382.

[33] Y. Shi, B. Qiu, D. Chen, J. Li, Y. Zhang, Proposing and validation of a new four-point finite-difference formula with manipulator application, IEEE Transactions on Industrial Informatics 14 (4) (2018) 1323–1333.

[34] C. Hu, X. Kang, Y. Zhang, Three-step general discrete-time Zhang neural network design and application to time-variant matrix inversion, Neurocomputing 306 (2018) 108–118.

[35] L. Xiao, Y. Zhang, Solving time-varying nonlinear inequalities using continuous and discrete-time Zhang dynamics, International Journal of Computer Mathematics 90 (5) (2013) 1114–1127.

[36] D. Guo, Z. Nie, L. Yan, Novel discrete-time Zhang neural network for time-varying matrix inversion, IEEE Transactions on Systems, Man, and Cybernetics: Systems 47 (8) (2017) 2301–2310.

[37] R. Horn, C. Johnson, Matrix Analysis, Cambridge University Press, Inc., New York, 2013.

[38] Y. Zhang, H. Gong, M. Yang, J. Li, X. Yang, Stepsize range and optimal value for Taylor-Zhang discretization formula applied to zeroing neurodynamics illustrated via future equality-constrained quadratic programming, IEEE Transactions on Neural Networks and Learning Systems 30 (3) (2019) 959–966.

[39] J. H. Mathews, K. D. Fink, Numerical Methods Using MATLAB, 4th Edition, Prentice Hall, New Jersey, 2004.

[40] Y. Zhang, Z. Li, D. Guo, Z. Ke, P. Chen, Discrete-time ZD, GD and NI for solving nonlinear time-varying equations, Numerical Algorithms 64 (4) (2013) 721–740.

Yunong Zhang
Sun Yat-sen University, Guangzhou, China

Photo: 

Biography

Yunong Zhang (S'02-M'03) received the B.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 1996, the M.S. degree from the South China University of Technology, Guangzhou, China, in 1999, and the Ph.D. degree from the Chinese University of Hong Kong, Hong Kong, in 2003. He joined Sun Yat-sen University, Guangzhou, in 2006, where he is currently a Professor with the School of Data and Computer Science. He had been with the National University of Ireland, Maynooth, Ireland, the University of Strathclyde, Glasgow, U.K., and the National University of Singapore, Singapore, since 2003. His current research interests include neural networks, robotics, computation, and optimization.

Yihong Ling
Sun Yat-sen University, Guangzhou, China

Photo: 

Biography

Yihong Ling received the B.S. degree in software engineering from South China Normal University, Foshan, China, in 2017. He is currently pursuing the M.S. degree in computer science and technology at Sun Yat-sen University, GuangZhou, China. His current research interests include neural networks, robotics, and computation.

Shuai Li
Swansea University, Wales, UK

Photo: 

Biography

Shuai Li received B.E. degree in precision mechanical engineering from Hefei University of Technology, Hefei, China, in 2005, M.E. degree in automatic control engineering from University of Science and Technology of China, Hefei, in 2008, and Ph.D. degree in electrical and computer engineering from Stevens Institute of Technology, Hoboken, NJ, USA, in 2014. He is currently an Associate Professor (Reader) at Swansea University, Wales, UK, leading the Robotic Lab, conducting research on robot manipulation and impedance control, multi-robot coordination, distributed control, intelligent optimization and control, and legged robots. Dr. Li is the founding Editor-in-Chief of International Journal of Robotics and Control and the General Co-Chair of 2018 International Conference on Advanced Robotics and Intelligent Control.

Min Yang
Sun Yat-sen University, Guangzhou, China

Photo: 

Biography

Min Yang received the B.E. degree in automation from Sun Yat-sen University, Guangzhou, China, in 2017. He is currently pursuing the Ph.D. degree in information and communication engineering at the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China. His current research interests include neural networks, robotics, and numerical computation.

Ning Tan
Sun Yat-sen University, Guangzhou, China

Photo:

Biography

Ning Tan presently is associate professor at Sun Yat-sen University, Guangzhou, China. He received the Ph.D. degree in automation from the Department of Automatic Control and Micro-Mechatronic Systems, Université de Franche-Comté/Franche-Comté Electronics Mechanics Thermal Science and Optics–Sciences and Technologies Institute, Besançon, France. From 2014 to 2018, he held a post-doctoral position and research fellow position at the Singapore University of Technology and Design and National University of Singapore, Singapore. His research interests include soft robotics, bioinspired design, modular and reconfigurable mechanisms, and micro-nano robotics.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: