Unique non-negative definite solution of the Time-Varying Algebraic Riccati Equations with Applications to Stabilization of LTV Systems

Theodore E. Simos[a,b,c,d,e], Vasilios N. Katsikis[f,*], Spyridon D. Mourtas[f], Predrag S. Stanimirović[g]

[a] *College of Applied Mathematics, Chengdu Univ. of Information Technology, Chengdu 610225, P.R. China*
[b] *Scientific and Educational Center "Digital Industry", South Ural State University, 76 Lenin Ave., 454 080 Chelyabinsk, Russia*
[c] *Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan*
[d] *Data Recovery Key Laboratory of Sichuan Province, Neijiang Normal Univ., Neijiang, China*
[e] *Section of Mathematics, Dept. of Civil Engineering, Democritus Univ. of Thrace, Xanthi, Greece*
[f] *Department of Economics, Division of Mathematics and Informatics, National and Kapodistrian University of Athens, Sofokleous 1 Street, 10559 Athens, Greece*
[g] *University of Niš, Faculty of Sciences and Mathematics, Višegradska 33, 18000 Niš, Serbia*

Abstract

In the context of infinite-horizon optimal control problems, the algebraic Riccati equations (ARE) arise when the stability of linear time-varying (LTV) systems is investigated. Using the zeroing neural network (ZNN) approach to solve the time-varying eigendecomposition-based ARE (TVE-ARE) problem, the ZNN model (ZNNTVE-ARE) for solving the TVE-ARE problem is introduced as a result of this research. Since the eigendecomposition approach is employed, the ZNNTVE-ARE model is designed to produce only the unique nonnegative definite solution of the time-varying ARE (TV-ARE) problem. It is worth mentioning that this model follows the principles of the

*Corresponding author
Email addresses: tsimos.conf@gmail.com (Theodore E. Simos), vaskatsikis@econ.uoa.gr (Vasilios N. Katsikis), spirosmourtas@gmail.com (Spyridon D. Mourtas), pecko@pmf.ni.ac.rs (Predrag S. Stanimirović)

ZNN method, which converges exponentially with time to a theoretical time-varying solution. The ZNNTVE-ARE model can also produce the eigenvector solution of the continuous-time Lyapunov equation (CLE) since the Lyapunov equation is a particular case of ARE. Moreover, this paper introduces a hybrid ZNN model for stabilizing LTV systems in which the ZNNTVE-ARE model is employed to solve the continuous-time ARE (CARE) related to the optimal control law. Experiments show that the ZNNTVE-ARE and HFTZNN-LTVSS models are both effective, and that the HFTZNN-LTVSS model always provides slightly better asymptotic stability than the models from which it is derived.

Keywords: Algebraic Riccati equations, Zeroing neural network, Eigendecomposition, Linear time-varying systems

1. Introduction

In applied mathematics, science, and various engineering problems, algebraic Riccati equations (ARE) have been the focus of significant interest (see [1, 2, 3, 4, 5, 6, 7]) since their pervasiveness to optimal linear system control and filtering theory introduced by Kalman in [8]. For instance, the robustness on a finite-time horizon of an uncertain linear time-varying (LTV) system is evaluated in [2]. Therein, the main emphasis is on the robust induced gains and limits on the feasible range of states. The analysis conditions are given as differential equations of Riccati type and differential linear matrix inequalities in two equivalent forms, leveraged by a new algorithm. In [5], a new algorithm for adjustive and attentive iterative learning control (ILC), in which the stability of the control updates is probabilistically analyzed, is proposed. The resulting learning controller can be applied effectively using a recursive approach. Applications on anthropomorphic robot arms demonstrate improved and more robust tracking performance without careful adaptation over derivative control and high-gain proportional, model-free, and model-based ILC.

In this paper, we focus only on the unique nonnegative definite solution of the time-varying ARE (TV-ARE) and, hence, the eigenvector solution of the TV-ARE is considered. In this way, the time-varying eigendecomposition-based ARE (TVE-ARE) problem is defined and tackled by the zeroing (or Zhang) neural network (ZNN) method, which originated from Hopfield neural network (HNN). The ZNN method is an excellent solution for addressing

different time-varying problems and has an exponential convergence performance, according to the models introduced in [9, 10, 11, 12, 13, 14, 15, 16]. The main feature of ZNN models is the ability to change their convergence speed by adjusting the design parameter value (see [17, 18, 19, 20, 21, 22]). Many Zeroing neural networks dynamical systems whose structure allows finite time convergence have been created. Some of them are defined with constant gain parameters, such as finite-time zeroing neural networks (FTZNN) [14, 23, 24, 25, 27], while some models are defined by varying-parameter finite-time zeroing neural networks (VPFTZNN) [28]. In contrast, different nonlinear activation functions have been applied in the known Zhang neural network (ZNN) dynamics [29, 30, 31, 32] to increase its convergence speed in a finite time. A noise-handling design for solving continuous-time matrix inversion was proposed in [33].

By applying the ZNN method to solve the TVE-ARE problem, the ZNNTVE-ARE model is introduced. Furthermore, since the Lyapunov equation is a special case of ARE, the ZNNTVE-ARE model can also produce the eigenvector solution of the continuous-time Lyapunov equation (CLE). In addition, a hybrid ZNN model, which comprises of the ZNNTVE-ARE model and called HEZNN-LTVSS, is developed and investigated for the LTV systems stabilization. The experiments show that the ZNNTVE-ARE and HFTZNN-LTVSS models are both effective and that the HFTZNN-LTVSS model always gives slightly better asymptotic stability than the models from which it is derived. It is worth noting that a ZNN model for solving the TV-ARE problem is introduced in [34]. The model therein is different from the ZNNTV-ARE model presented here in the following two ways:
(1) Unlike the model presented in [34], the ZNNTVE-ARE model uses an eigendecomposition-based approach, which leads to a different ZNN design.
(2) The ZNNTVE-ARE model is designed to produce only the unique nonnegative definite solution of the TV-ARE, whereas the model presented in [34] produces a random solution of the solution set.

Our research key contributions in this paper are illustrated as follows.
(1) A ZNN dynamical system for the TVE-ARE problem solving is introduced. As far as we know, the ZNN design for finding only the unique nonnegative definite solution of ARE is an original approach.
(2) The ZNN dynamical system to solve the TVE-ARE problem can also produce the eigenvector solution of the CLE.
(3) A hybrid ZNN model for the LTV systems stabilization is developed and investigated.

(4) Illustrative examples are carried out to check the applicability and effectiveness of the proposed explicit models.
(5) In addition to the numerical examples for solving the TVE-ARE problem, three examples for stabilizing LTV systems are presented.

The overall arrangement of parts in the paper is based on the following hierarchy. Section 2 introduces preliminaries upon the LTV systems stabilization, TV-ARE, and the eigendecomposition of a matrix. Section 3 defines the ZNNTVE-ARE model, which is based on the ZNN dynamics. Section 4 comprises the hybrid ZNN model for stabilizing LTV systems, which is marked as HEZNN-LTVSS and incorporates the ZNNTVE-ARE models. Section 5 presents six different experiments, two of which include applications to solve TV-ARE, one includes application to solve CLE, and the last three include applications to stabilization of LTV systems. The efficacy of the proposed models is demonstrated by the associated simulation analysis. Finally, section 6 contains the concluding comments.

2. Problem Formulation

Consider the following linear time-varying (LTV) system:

$$\dot{\mathrm{x}}(t) = P(t)\mathrm{x}(t) + L(t)w(t), \quad \mathrm{x}(0) = \mathrm{x}_0. \tag{2.1}$$

where $P(t) \in \mathbb{R}^{n\times n}$, $\mathrm{x}(t) \in \mathbb{R}^n$, $L(t) \in \mathbb{R}^{n\times m}$ and $w(t) \in \mathbb{R}^m$. The infinite horizon quadratic cost is given by

$$J(w) = 1/2 \int_0^\infty [\mathrm{x}^{\mathrm{T}}(t)S(t)\mathrm{x}(t) + w^{\mathrm{T}}(t)K(t)w(t)]\mathrm{d}t, \tag{2.2}$$

where $()^{\mathrm{T}}$ stands for the transpose matrix, $S(t) \in \mathbb{R}^{n\times n}$ is a nonnegative definite matrix and $K(t) \in \mathbb{R}^{m\times m}$ is a positive definite matrix. The feedback control law that minimizes (2.2) is

$$w(t) = -E(t)\mathrm{x}(t), \tag{2.3}$$

where $E(t) = K^{-1}(t)L^{\mathrm{T}}(t)X(t) \in \mathbb{R}^{m\times n}$. Note that the superscript $()^{-1}$ implies the inverse operator, and $X(t) \in \mathbb{R}^{n\times n}$ is the solution to the following continuous-time ARE (CARE):

$$P^{\mathrm{T}}(t)X(t)+X(t)P(t)-X(t)L(t)K^{-1}(t)L^{\mathrm{T}}(t)X(t)+S(t)=\mathbf{0}, \tag{2.4}$$

in which $\mathbf{0} \in \mathbb{R}^{n\times n}$ denotes the zero matrix. The stability properties of the controller (2.3) are presented in [35, 36].

2.1. Eigenvector solution

By setting $H(t) = L(t)K^{-1}(t)L^{\mathrm{T}}(t) \in \mathbb{R}^{n\times n}$, (2.4) is converted to the following general form of time-varying ARE (TV-ARE):

$$P^{\mathrm{T}}(t)X(t) + X(t)P(t) - X(t)H(t)X(t) + S(t) = \mathbf{0}, \tag{2.5}$$

where $H(t)$, $S(t)$ are symmetric and nonnegative definite matrices. It is known that the equation (2.5) has a unique nonnegative definite solution under these assumptions (see [37]). Inhere, we are only focused on a unique nonnegative definite solution $X(t)$.

Now consider the Hamiltonian matrix

$$W(t) = \begin{bmatrix} P(t) & -H(t) \\ -S(t) & -P^{\mathrm{T}}(t) \end{bmatrix} \in \mathbb{R}^{2n\times 2n}.$$

By performing eigendecomposition on $W(t)$ we get:

$$W(t)V(t) = V(t)\Lambda(t), \tag{2.6}$$

where $V(t)$ is the $2n \times 2n$ modal matrix of eigenvector columns related to the form of each system mode and $\Lambda(t)$ is a diagonal matrix with entries $\lambda_i(t)$, $i = 1, 2, \ldots, 2n$ on the main diagonal. Note that $\lambda_i(t)$ denotes the ith eigenvalue of $W(t)$. Furthermore, $W(t)$ includes divergent and convergent mode pairs, with eigenvalues equal in magnitude and with opposite signs.

Let $\Lambda_1(t)$ be the $n \times n$ diagonal matrix with nonzero entries $\lambda_i(t)$ which possess negative real parts, and $\Lambda_2(t)$ be the $n \times n$ diagonal matrix whose diagonal entries $\lambda_i(t)$ have positive real parts. Then (2.6) can be decomposed in appropriate blocks. As presented in [38, 39],

$$W(t)\begin{bmatrix} V_{11}(t) & V_{12}(t) \\ V_{21}(t) & V_{22}(t) \end{bmatrix} = \begin{bmatrix} V_{11}(t) & V_{12}(t) \\ V_{21}(t) & V_{22}(t) \end{bmatrix}\begin{bmatrix} \Lambda_1(t) & \mathbf{0} \\ \mathbf{0} & \Lambda_2(t) \end{bmatrix}, \tag{2.7}$$

where $V_{11}(t), V_{12}(t), V_{21}(t), V_{22}(t)$ are $n \times n$ blocks of $V(t)$. According to [40], to derive in a steady-state solution of (2.5) that induces an explicit solution of the optimal-control problem for the LTV of (2.1) in terms of the eigenvectors of $W(t)$, we need to eliminate the unstable modes in the optimal control problem formulation. Since the unstable modes arise from $\Lambda_2(t)$, we can eliminate $\Lambda_2(t)$ in (2.7) and, hence, the following holds for the stable modes

$$\begin{bmatrix} P(t) & -H(t) \\ -S(t) & -P^{\mathrm{T}}(t) \end{bmatrix}\begin{bmatrix} V_{11}(t) \\ V_{21}(t) \end{bmatrix} = \begin{bmatrix} V_{11}(t) \\ V_{21}(t) \end{bmatrix}\Lambda_1(t), \tag{2.8}$$

or, equivalently,

$$P(t)V_{11}(t) - H(t)V_{21}(t) = V_{11}(t)\Lambda_1(t), \tag{2.9}$$

$$-S(t)V_{11}(t) - P^{\mathrm{T}}(t)V_{21}(t) = V_{21}(t)\Lambda_1(t). \tag{2.10}$$

As presented in [39], on eliminating the eigenvalues $\Lambda_1(t)$ gives a positive definite solution $X(t)$ of (2.5) under the transformation

$$X(t) = V_{21}(t)V_{11}^{-1}(t). \tag{2.11}$$

2.2. Eigenvector solution of Lyapunov equation

For the free system (2.1), the stability problem can be defined by the direct method of Lyapunov, which states that the symmetric matrix $X(t)$ related to the continuous-time equation

$$P^{\mathrm{T}}(t)X(t) + X(t)P(t) + S(t) = \mathbf{0} \tag{2.12}$$

should be positive definite for any symmetric positive definite matrix S to ensure asymptotic stability of (2.1). As presented in [38], a different solution can be appointed in terms of the eigenvalue problem determined by

$$\begin{bmatrix} P(t) & \mathbf{0} \\ -S(t) & -P^{\mathrm{T}}(t) \end{bmatrix} \begin{bmatrix} V_{11}(t) \\ V_{21}(t) \end{bmatrix} = \begin{bmatrix} V_{11}(t) \\ V_{21}(t) \end{bmatrix} \Lambda_1(t), \tag{2.13}$$

where the n eigenvalues of $\Lambda_1(t)$, with considered negative real parts, are related to the matrix $P(t)$, and $V_{11}(t)$, and $V_{21}(t)$ are the corresponding eigenvector components for the $2n \times 2n$ system (2.13). Hence, (2.11) determines the positive definite symmetric solution of the CLE given by (2.12). Note that CLE is a linear matrix equation that arises from the TV-ARE.

2.3. Time-varying eigendecomposition-based ARE problem

Based on the previous analysis, the following group of matrix equations

$$\begin{cases} P(t)V_{11}(t) - H(t)V_{21}(t) = V_{11}(t)\Lambda_1(t), \\ -S(t)V_{11}(t) - P^{\mathrm{T}}(t)V_{21}(t) = V_{21}(t)\Lambda_1(t), \\ X(t)V_{11}(t) = V_{21}(t), \end{cases} \tag{2.14}$$

is assumed in finding the eigenvector solution to TV-ARE and CLE. Note that TV-ARE becomes a CLE when $H(t) = \mathbf{0}$.

Following the methodology proposed in [41, 42], which does not separate the real and imaginary components, we have the problem of finding four complex matrices instead of the problem of finding eight real matrices. As a result of solving (2.14), we acquire $V_{11}(t)$, $V_{21}(t)$, $\Lambda_1(t)$, and $X(t)$.

3. Solving TVE-ARE via zeroing neural networks

The ZNN evolution was introduced by Zhang *et al.* in 2001, [43], and has been researched and developed as an essential class of recurrent neural networks since then. In addition, ZNN has been studied theoretically and has been proven to be an effective and reliable tool for solving time-varying problems. According to the ZNN design under the linear activation function (see [9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22]), the error matrix $Z(t)$ can be defined and then can be dynamically regulated by the following evolution formula

$$\dot{Z}(t) = -\beta Z(t), \tag{3.1}$$

where $\dot{(\ )}$ denotes the time derivative and $\beta > 0$ is the ZNN design parameter. It is shown that the greater the value assigned to β, the higher the convergence rate. From (3.1), $Z(t)$ is compelled to exponential convergence to the null matrix. We intend to formulate a continuous-time TVE-ARE approach that is based upon the transformations performed in section 2. In order to acquire $V_{11}(t)$, $V_{21}(t)$, $\Lambda_1(t)$, and $X(t)$ satisfying (2.14), the following error matrix equation group is constructed:

$$\begin{cases} Z_1(t)=P(t)V_{11}(t)-H(t)V_{21}(t)-V_{11}(t)\Lambda_1(t), \\ Z_2(t)=S(t)V_{11}(t)+P^{\mathrm{T}}(t)V_{21}(t)+V_{21}(t)\Lambda_1(t), \\ Z_3(t)=X(t)V_{11}(t)-V_{21}(t). \end{cases} \tag{3.2}$$

Substituting $Z_i(t), i = 1, 2, 3$, defined in (3.2) in place of $Z(t)$ into (3.1), one obtains

$$\begin{cases} -\beta_1 Z_1(t) = \dot{P}(t)V_{11}(t) + P(t)\dot{V}_{11}(t) - \dot{H}(t)V_{21}(t) - H(t)\dot{V}_{21}(t) \\ \qquad\qquad -\dot{V}_{11}(t)\Lambda_1(t) - V_{11}(t)\dot{\Lambda}_1(t), \\ -\beta_2 Z_2(t) = \dot{S}(t)V_{11}(t) + S(t)\dot{V}_{11}(t) + \dot{P}^{\mathrm{T}}(t)V_{21}(t) + P^{\mathrm{T}}(t)\dot{V}_{21}(t) \\ \qquad\qquad +\dot{V}_{21}(t)\Lambda_1(t) + V_{21}(t)\dot{\Lambda}_1(t), \\ -\beta_3 Z_3(t) = \dot{X}(t)V_{11}(t) + X(t)\dot{V}_{11}(t) - \dot{V}_{21}(t). \end{cases} \tag{3.3}$$

For plainness, all ZNN design parameters are set to a single value β, i.e., $\beta_1 = \beta_2 = \beta_3 = \beta$.

To acquire a dynamic model that is simple and transparent, and that can promptly calculate $V_{11}(t)$, $V_{21}(t)$, $\Lambda_1(t)$, and $X(t)$, (3.3) requires to be simplified by applying the Kronecker product $\otimes$ and the vectorization technique $\mathrm{vec}(\cdot)$ as bellow:

$$\begin{cases} \mathrm{vec}(-\beta\, Z_1(t) - \dot{P}(t)V_{11}(t) + \dot{H}(t)V_{21}(t)) = (I \otimes P(t))\mathrm{vec}(\dot{V}_{11}(t)) \\ \qquad -(I \otimes H(t))\mathrm{vec}(\dot{V}_{21}(t)) - (\Lambda_1^{\mathrm{T}}(t) \otimes I)\mathrm{vec}(\dot{V}_{11}(t)) - (I \otimes V_{11}(t))\mathrm{vec}(\dot{\Lambda}_1(t)), \\ \mathrm{vec}(-\beta\, Z_2(t) - \dot{S}(t)V_{11}(t) - \dot{P}^{\mathrm{T}}(t)V_{21}(t)) = (I \otimes S(t))\mathrm{vec}(\dot{V}_{11}(t)) \\ \qquad +(I \otimes P^{\mathrm{T}}(t))\mathrm{vec}(\dot{V}_{21}(t)) + (\Lambda_1^{\mathrm{T}}(t) \otimes I)\mathrm{vec}(\dot{V}_{21}(t)) + (I \otimes V_{21}(t))\mathrm{vec}(\dot{\Lambda}_1(t)), \\ \mathrm{vec}(-\beta\, Z_3(t) = (V_{11}^{\mathrm{T}}(t) \otimes I)\mathrm{vec}(\dot{X}(t)) + (I \otimes X(t))\mathrm{vec}(\dot{V}_{11}(t)) - (I \otimes I)\mathrm{vec}(\dot{V}_{21}(t)), \end{cases} \tag{3.4}$$

where $I \in \mathbb{R}^{n \times n}$ denotes the $n \times n$ identity matrix.

Because the matrix $\Lambda_1(t)$ is diagonal, we only need to acquire the elements of $\Lambda_1(t)$ placed on its main diagonal. Hence, by separating these elements from $\Lambda_1(t)$ and placing them into the vector $\mathbf{c}(t)$, it is sufficient to use $\dot{\mathbf{c}}(t)$ instead of $\dot{\Lambda}_1(t)$. In this way, $\Lambda_1(t)$ is compelled to be a diagonal matrix and, simultaneously, the dimensions of (3.4) are reduced.

Thus, employing the n main diagonal elements of $\Lambda_1(t)$, we obtain

$$\mathrm{vec}(\dot{\Lambda}_1(t)) = M\dot{\mathbf{c}}(t) \tag{3.5}$$

which substitutes $\mathrm{vec}(\dot{\Lambda}_1(t))$ in (3.4). In addition, the vector $\dot{\mathbf{c}}(t) \in \mathbb{C}^n$ is determined by piling the elements placed along the main diagonal of $\dot{\Lambda}_1(t)$.

Moreover, the matrix $M \in \mathbb{R}^{n^2\times n}$ can be computed applying Alg. 1, where zeros$(\cdot)$, floor$(\cdot)$ and mod$(\cdot)$ represent the standard MATLAB functions [44].

Algorithm 1 Procedure for computing the matrix M.

Input: The number n of columns (or rows) of the matrix $\Lambda_1(t) \in \mathbb{C}^{n\times n}$.

Put M =zeros(n^2, n)
for $l = 1 : n^2$ **do**
  Put g =floor$(\frac{l-1}{n}) + 1$ and h =mod$(l - 1, n) + 1$
  if $g == h$ **then**
    Put $M(l, h) = 1$
  end if
end for
return M

Output: The generated matrix M.

By employing the matrix M, (3.4) can be reformulated as below:

$$\begin{cases} \mathrm{vec}(-\beta\, Z_1(t) - \dot{P}(t)V_{11}(t) + \dot{H}(t)V_{21}(t)) = (I\otimes P(t))\mathrm{vec}(\dot{V}_{11}(t)) \\ \qquad -(I\otimes H(t))\mathrm{vec}(\dot{V}_{21}(t)) - (\Lambda_1^{\mathrm{T}}(t)\otimes I)\mathrm{vec}(\dot{V}_{11}(t)) - (I\otimes V_{11}(t))M\dot{\mathbf{c}}(t), \\ \mathrm{vec}(-\beta\, Z_2(t) - \dot{S}(t)V_{11}(t) - \dot{P}^{\mathrm{T}}(t)V_{21}(t)) = (I\otimes S(t))\mathrm{vec}(\dot{V}_{11}(t)) \\ \qquad +(I\otimes P^{\mathrm{T}}(t))\mathrm{vec}(\dot{V}_{21}(t)) + (\Lambda_1^{\mathrm{T}}(t)\otimes I)\mathrm{vec}(\dot{V}_{21}(t)) + (I\otimes V_{21}(t))M\dot{\mathbf{c}}(t), \\ \mathrm{vec}(-\beta\, Z_3(t) = (V_{11}^{\mathrm{T}}(t)\otimes I)\mathrm{vec}(\dot{X}(t)) + (I\otimes X(t))\mathrm{vec}(\dot{V}_{11}(t)) - (I\otimes I)\mathrm{vec}(\dot{V}_{21}(t)), \end{cases} \tag{3.6}$$

We define the following matrices

$$N(t) = \begin{bmatrix} N_{11}(t) & N_{12}(t) & N_{13}(t) & N_{14}(t) \\ N_{21}(t) & N_{22}(t) & N_{23}(t) & N_{24}(t) \\ N_{31}(t) & N_{32}(t) & N_{33}(t) & N_{34}(t) \end{bmatrix}, \quad \dot{D}(t) = \begin{bmatrix} \mathrm{vec}(\dot{V}_{11}(t)) \\ \mathrm{vec}(\dot{V}_{21}(t)) \\ \dot{\mathbf{c}}(t) \\ \mathrm{vec}(\dot{X}(t)) \end{bmatrix},$$

$$F(t) = \big[F_1(t), F_2(t), F_3(t)\big]^{\mathrm{T}} \in \mathbb{C}^{3n^2},$$

where $N(t) \in \mathbb{C}^{3n^2\times(3n^2+n)}$, $\dot{D}(t) \in \mathbb{C}^{3n^2+n}$ and

- $N_{11}(t)=I\otimes P(t)-\Lambda_1^{\mathrm{T}}(t)\otimes I$
- $N_{12}(t)=-I\otimes H(t)$
- $N_{21}(t)=I\otimes S(t)$
- $N_{31}(t)=I\otimes X(t)$
- $N_{34}(t)=V_{11}^{\mathrm{T}}(t)\otimes I$
- $N_{33}(t)=\hat{\mathbf{0}}$
- $N_{22}(t)=I\otimes P^{\mathrm{T}}(t)+\Lambda_1^{\mathrm{T}}(t)\otimes I$
- $N_{13}(t)=-(I\otimes V_{11}(t))M$
- $N_{23}(t)=(I\otimes V_{21}(t))M$
- $N_{32}(t)=-I\otimes I$
- $N_{14}(t)=N_{24}(t)=\tilde{\mathbf{0}}$

with

$$\begin{cases} N_{kj}(t)\in\mathbb{R}^{n^2\times n} & ,k=1,2,3 \text{ and } j=3 \\ N_{kj}(t)\in\mathbb{R}^{n^2\times n^2} & ,\text{else} \end{cases},$$

and

- $F_1(t)=\mathrm{vec}(-\beta Z_1(t)-\dot{P}(t)V_{11}(t)+\dot{H}(t)V_{21}(t))$
- $F_2(t)=\mathrm{vec}(-\beta Z_2(t)-\dot{S}(t)V_{11}(t)-\dot{P}^{\mathrm{T}}(t)V_{21}(t))$
- $F_3(t)=\mathrm{vec}(-\beta Z_3(t))$

with $F_1(t),F_2(t),F_3(t)\in\mathbb{R}^{n^2}$. Then (3.6) is be translated as

$$N(t)\dot{D}(t)=F(t). \tag{3.7}$$

Note that $\tilde{\mathbf{0}}\in\mathbb{R}^{n^2\times n^2}$ and $\hat{\mathbf{0}}\in\mathbb{R}^{n^2\times n}$ denote the zero matrix. The corresponding continuous-time TVE-ARE model is acquired by the best approximate solution of (3.7)

$$\dot{D}(t)=N^{\dagger}(t)F(t), \tag{3.8}$$

where $()^{\dagger}$ is a mark for the pseudoinverse operator. Note that {(3.6),(3.8)} is the proposed ZNNTVE-ARE model and the solution $D(t)$ of (3.8) can be efficiently generated by employing an `ode` MATLAB solver. Of course, there are the costs of calculating the pseudoinverse of the matrix $N(t)$. Moreover, after rearranging the elements of the vector $D(t)$, the reconstruction of $V_{11}(t)$, $V_{21}(t)$, $\Lambda_1(t)$, and $X(t)$ can be accomplished. The exponential convergence of the ZNNTVE-ARE model {(3.6),(3.8)} to the theoretical time-varying unique nonnegative solution of TV-ARE (2.5) is proved in Theorem 3.1.

Theorem 3.1. *Let $P(t),H(t),S(t)\in\mathbb{R}^{n\times n}$ be differentiable and $H(t)$, $S(t)$ be symmetric and nonnegative definite matrices. The ZNNTVE-ARE model* {(3.6),(3.8)} *starting from any initial value $D(0)$, converges exponentially to the theoretical time-varying unique nonnegative solution of TV-ARE* (2.5).

Proof. In order to obtain the solution $D(t)$ of TV-ARE (2.5), the error matrix equation group is defined as in (3.2), based on the ZNN design. After that, by adopting the linear design formula for zeroing (3.2), the model (3.3) is obtained. From [45, Theorem 1], each error matrix equation in the error matrix equation group (3.3) converges to the theoretical solution when $t \to \infty$. As a result, the solution of (3.3) converges to the theoretical solution of (2.5) when $t \to \infty$. In addition, from the derivation process of (3.8), we know that it is actually another form of (3.3). The proof is thus completed. □

4. Hybrid Eigendecomposition-based ZNN Model for LTV Systems Stabilization

To stabilize LTV systems two models based on heuristic techniques are introduced in [46]. The two models are the frozen-time Riccati equation (FTRE), which is based on the state-dependent Riccati Equation (SDRE) method, and the forward propagating Riccati equation (FPRE), which solves the differential-algebraic Riccati equation forward in time rather than backward in time as in optimal control. Influenced by these two models, the HEZNN-LTVSS model, which is based on the ZNNTVE-ARE model, is introduced and investigated in this paper to stabilize LTV systems.

It is obvious that, after the replacement $H(t) = L(t)K^{-1}(t)L^{\mathrm{T}}(t)$, the equation (2.5) can be approached through the ZNNTVE-ARE {(3.6),(3.8)} model. After the solution $X(t)$ of (2.5) is obtained, then the state feedback control law of (2.3) can be determined. As a consequence, the initial problem (2.1) can be solved. That is, (2.1) can be rewritten as

$$\dot{\mathrm{x}}(t) = P(t)\mathrm{x}(t)+L(t)\big(-K^{-1}(t)L^{\mathrm{T}}(t)X(t)\mathrm{x}(t)\big),$$

and, equivalently,

$$\dot{\mathrm{x}}(t) = \big(P(t) - H(t)X(t)\big)\mathrm{x}(t). \tag{4.1}$$

In the following theorem 4.1, considering the general infinite-horizon nonlinear regulator problem of minimizing (2.2) concerning the state x and the control w subject to the nonlinear differential constraint (2.1), the stability of the SDRE method is proved. Also, note that $\mathbb{C}^k$ denotes the space of continuous functions that have continuous first k derivatives.

Theorem 4.1. *Consider the general infinite-horizon nonlinear regulator problem of minimizing* (2.2) *with respect to the state* x *and the control* w *subject*

to the nonlinear differential constraint (2.1). *In addition to* $P(\mathrm{x})$, $L(\mathrm{x})$, $S(\mathrm{x})$, *and* $K(\mathrm{x})$ *belonging to* $\mathbb{C}^k$, *assume that the state dependent coefficient parameterization* $P(\mathrm{x})$ *is smooth and is both a stabilisable and detectable parameterization of the nonlinear system. Then the SDRE method produces a closed loop solution which is locally asymptotically stable.*

Proof. Note that the closed loop solution is given by (4.1). That is,

$$\begin{aligned}\dot{\mathrm{x}} &= \big(P(\mathrm{x}) - L(\mathrm{x})K^{-1}(\mathrm{x})L^{\mathrm{T}}(\mathrm{x})X(\mathrm{x})\big)\mathrm{x} \\ &= P_c(\mathrm{x})\mathrm{x}\end{aligned}$$

and that the closed loop matrix

$$P_c(\mathrm{x}) = P(\mathrm{x}) - L(\mathrm{x})K^{-1}(\mathrm{x})L^{\mathrm{T}}(\mathrm{x})X(\mathrm{x})$$

is guaranteed to be stable at every point x from the Riccati equation theory. Under the smoothness assumptions, $X(\mathrm{x})$ and hence $P_c(\mathrm{x})$ are also smooth. Expanding the matrix $P_c(\mathrm{x})$ into the partial Taylor series expansion about zero yields for some neighborhood about the origin

$$\dot{\mathrm{x}} \approx P_c(\mathrm{x})\mathrm{x} + \psi(\mathrm{x}) \cdot \|\mathrm{x}\|$$

where $\psi(\mathrm{x})$ is of order k and

$$\lim_{\|\mathrm{x}\|\to 0} \psi(\mathrm{x}) = 0.$$

In a smaller neighborhood about the origin, the linear term which has a constant stable coefficient matrix dominates the higher-order term yielding local asymptotic stability. □

Combining (3.8) and (4.1), the Hybrid Eigendecomposition ZNN for LTV Systems Stabilization (HEZNN-LTVSS) model is acquired as presented below:

$$\begin{bmatrix} F(t) \\ \big(P(t) - H(t)X(t)\big)\mathrm{x}(t) \end{bmatrix} = \begin{bmatrix} N(t) & \check{\mathbf{0}} \\ \bar{\mathbf{0}} & I \end{bmatrix} \begin{bmatrix} \dot{D}(t) \\ \dot{\mathrm{x}}(t) \end{bmatrix}, \tag{4.2}$$

where $\check{\mathbf{0}}$ is a $3n^2 \times n$ zero matrix and $\bar{\mathbf{0}}$ is a $n \times (3n^2 + n)$ zero matrix, respectively. An explicit form of the dynamics (4.2) is equivalent to

$$\begin{bmatrix} \dot{D}(t) \\ \dot{\mathrm{x}}(t) \end{bmatrix} = \begin{bmatrix} N(t) & \check{\mathbf{0}} \\ \bar{\mathbf{0}} & I \end{bmatrix}^{\dagger} \begin{bmatrix} F(t) \\ \big(P(t) - H(t)X(t)\big)\mathrm{x}(t) \end{bmatrix}. \tag{4.3}$$

Note that (4.3) is the proposed HEZNN-LTVSS model that can be effectively solved by applying an `ode` MATLAB solver.

In the following theorem 4.2, considering the general infinite-horizon nonlinear regulator problem of minimizing (2.2) with respect to the state x and the control w subject to the nonlinear differential constraint (2.1), the stability of the HEZNN-LTVSS model (4.3) is proved.

Theorem 4.2. *Consider the general infinite-horizon nonlinear regulator problem of minimizing* (2.2) *with respect to the state* x *and the control* w *subject to the nonlinear differential constraint* (2.1). *In addition to* $P(\mathrm{x})$, $L(\mathrm{x})$, $S(\mathrm{x})$, *and* $K(\mathrm{x})$ *belonging to* $\mathbb{C}^k$, *assume that the state dependent coefficient parameterization* $P(\mathrm{x})$ *is smooth and is both a stabilisable and detectable parameterization of the nonlinear system. Then the HEZNN-LTVSS model* (4.3) *produces a closed loop solution which is locally asymptotically stable.*

Proof. Since the HEZNN-LTVSS model (4.3) comprises of the ZNNTVE-ARE model {(3.6),(3.8)} and the SDRE method, it can be derived from Theorem 3.1 and Theorem 4.1 that the HEZNN-LTVSS model (4.3) produces a closed-loop solution which is locally asymptotically stable. □

5. Illustrative Examples

This section presents and comments six examples. Three of these examples demonstrate the effectiveness of the ZNNETV-ARE model, and the other three contain applications of the HEZNN-LTVSS to solve LTV systems. Additionally, it is necessary to explain a few important parameters and symbols, and some additional details must be given as preliminaries for the coming examples.

1) The time interval is restricted to $[0, 10]$ during the computation, which indicates that the starting time is $t_0 = 0$ and the ending time is $t_f = 10$.

2) $\|\cdot\|_F$ means the matrix Frobenius norm.

3) The error $\|\text{TV-ARE}\|_F$ in Figs. 1d and 2d means

$$\left\|P^{\mathrm{T}}(t)X(t)+X(t)P(t)-X(t)H(t)X(t)+S(t)\right\|_F.$$

Furthermore, the error $\|\text{CLE}\|_F$ in Fig. 3d stands for

$$\left\|P^{\mathrm{T}}(t)X(t)+X(t)P(t)+S(t)\right\|_F.$$

4) In all examples of this section, the ZNN design parameter is set to $\beta = 100$.

5) The `ode15s` MATLAB solver with default settings is employed for acquiring the ZNNTVE-ARE solution {(3.6),(3.8)} and the HEZNN-LTVSS solution (4.3).

6) With the exception of example 5.3, in which $X^{\star}(t)$ denotes the theoretical solution, examples 5.1 and 5.2 include the optimal eigenvector solution (see [38, 39]) denoted by $X^{\star}(t)$.

7) The superscript $()^{ZNN}$ denotes the solution produced by the ZNNETV-ARE in Figs. 1-3, and the solution produced by the HEZNN-LTVSS in Figs. 4 and 5, while the superscript $()^{FTRE}$ and $()^{FPRE}$ in Figs. 4 and 5 denote the solution produced by the FTRE and FPRE models presented in [46].

8) The notation *ZNN*, *FTRE* and *FPRE* in the legend of Figs. 4 and 5 denote the solutions produced by the HEZNN-LTVSS, FTRE and FPRE models, respectively.

9) In the legend of figures, the subscripts r and i denote the real and imaginary parts, respectively.

5.1. Example 1

Assume the input matrices $P(t)$, $H(t)$ and $S(t)$ as follows:

$$P(t) = \begin{bmatrix} 2+\sin(t) & 0 \\ -4-\sin(t) & 4+\sin(2t) \end{bmatrix}, \quad H(t) = \begin{bmatrix} 3-\cos(2t) & 0 \\ 0 & 5-\sin(2t) \end{bmatrix},$$
$$S(t) = \begin{bmatrix} 0 & 3-\cos(t) \\ 3-\cos(t) & 0 \end{bmatrix}.$$

Given the round number to the nearest integer of the eigenvalues with negative real part, we set $\mathbf{c}(0) = [-3,-3]^{\mathrm{T}} + [-1,1]^{\mathrm{T}}\imath$. Then, for the initial values

$$V_{11}(0) = V_{21}(0) = \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix} + \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}\imath, \quad X(0) = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix},$$

the results of ZNNTVE-ARE are depicted in Figs. 1a-1d.

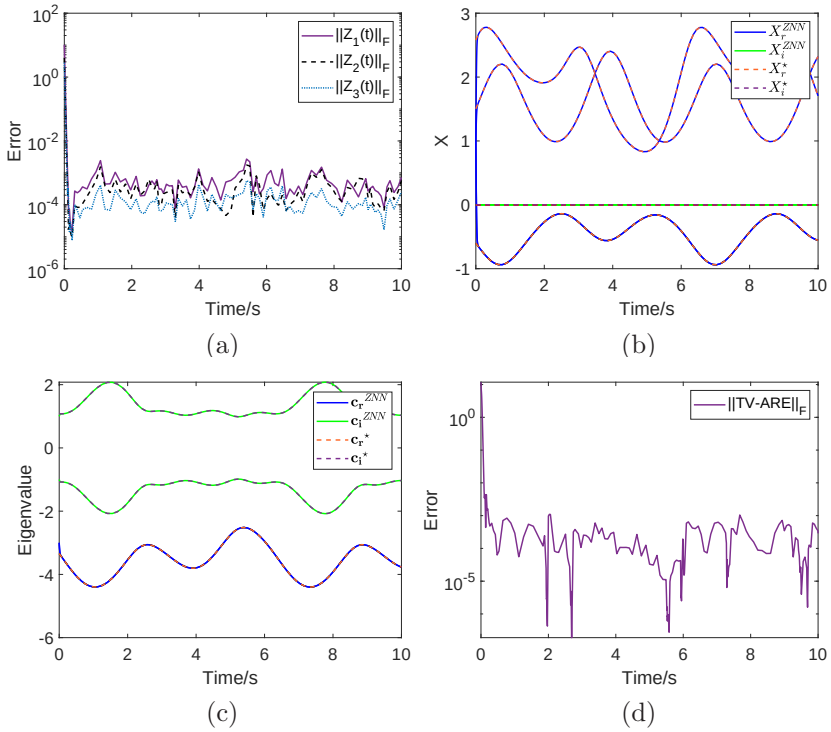

Figure 1: Performance of ZNNTVE-ARE for solving example 5.1.
(1a-1d): The errors $\|Z_i(t)\|_F$, i=1, 2, 3, the solution $X(t)$, the eigenvalues and the error $\|\text{TV-ARE}\|_F$ in example 5.1.

More precisely, Fig. 1a depicts the trajectories of $\|Z_i(t)\|_F$, i=1, 2, 3, of ZNNTVE-ARE. Fig. 1b depicts the trajectories of the ZNNTVE-ARE's solution $X^{ZNN}(t)$ decomposed into $X_r^{ZNN}(t)$ (real part) and $X_i^{ZNN}(t)$ (real part), along with the optimal eigenvector solution $X^{\star}(t)$, decomposed into $X_r^{\star}(t)$ (real part) and $X_i^{\star}(t)$ (imaginary part). Fig. 1c comprises of the eigenvalues produced by the ZNNTVE-ARE, which are decomposed into $\mathbf{c_r}^{ZNN}(t)$ (real part) and $\mathbf{c_i}^{ZNN}(t)$ (imaginary part), along with the actual eigenvalues of the problem, decomposed into $\mathbf{c_r}^{\star}(t)$ (real part) and $\mathbf{c_i}^{\star}(t)$ (imaginary part). Fig. 1d shows the error $\|\text{TV-ARE}\|_F$ produced by the ZNNTVE-ARE's solution $X^{ZNN}(t)$.

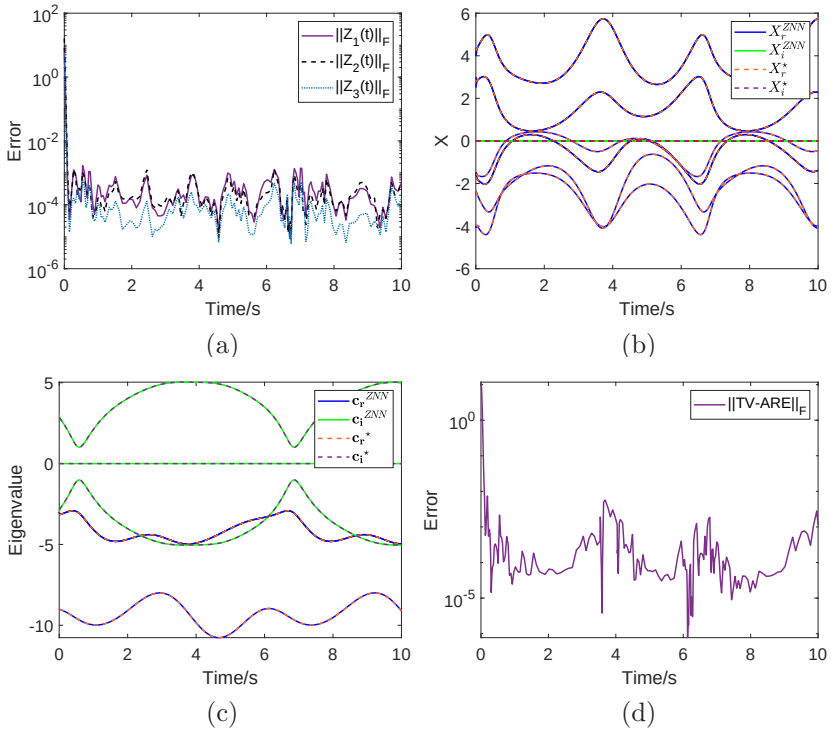

Figure 2: Performance of ZNNTVE-ARE for solving example 5.2.
(2a-2d): The errors $\|Z_i(t)\|_F$, i=1, 2, 3, the solution $X(t)$, the eigenvalues and the error $\|\text{TV-ARE}\|_F$ in example 5.2.

5.2. Example 2

Assume the input matrices $P(t)$, $H(t)$ and $S(t)$ of dimensions 3×3 as follows:

$$P(t) = \begin{bmatrix} 3+\sin(t) & 5-\sin(t) & 4-\cos(2t) \\ 5-\sin(t) & 4+\sin(2t) & -3-\sin(t) \\ 4-\cos(2t) & -3-\sin(t) & 4 \end{bmatrix},$$

$$H(t) = \begin{bmatrix} 8+3\sin(t) & 4-\sin(t) & 1+\cos(2t) \\ 4-\sin(t) & 1/2+\cos(t) & 5+\cos(t) \\ 1+\cos(2t) & 5+\cos(t) & 3/2+\sin(2t) \end{bmatrix},$$

$$S(t) = \begin{bmatrix} -8+2\sin(t) & -8+\cos(t) & 3/2 \\ -8+\cos(t) & 3 & 6-\cos(t) \\ 3/2 & 6-\cos(t) & 5+\sin(2t) \end{bmatrix}.$$

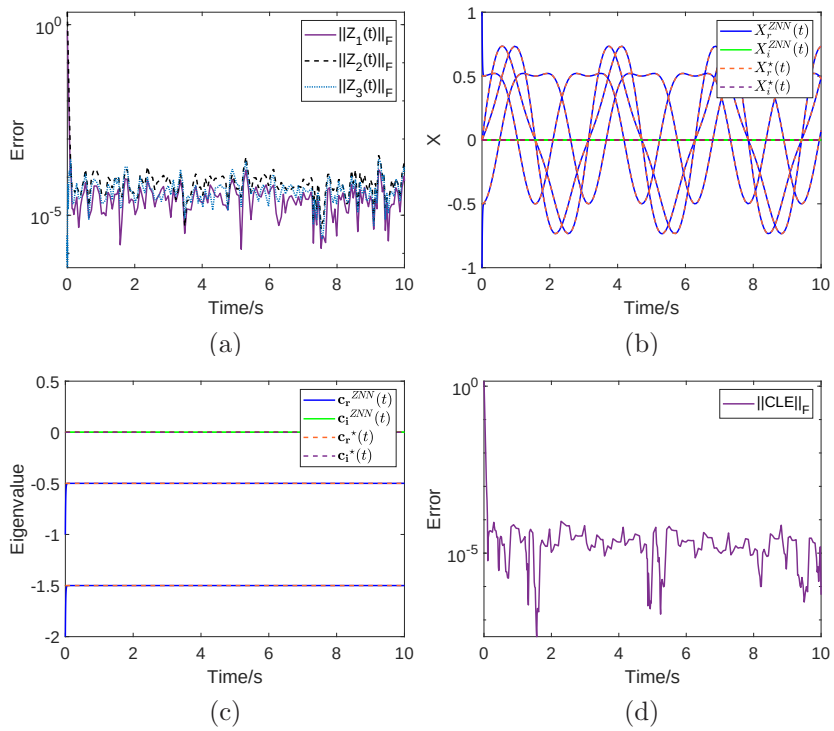

Figure 3: Performance of ZNNTVE-ARE for solving example 5.3.
(3a-3d): The errors $\|Z_i(t)\|_F$, i=1, 2, 3, the solution $X(t)$, the eigenvalues and the error $\|\mathrm{CLE}\|_F$ in example 5.3.

Given the round number to the nearest integer of the eigenvalues with a negative real part, we set $\mathbf{c}(0) = [-9, -3, -3]^{\mathrm{T}} + [0, -3, 3]^{\mathrm{T}}\imath$. Then, for the initial values

$$V_{11}(0)=\begin{bmatrix} 0 & -1 & -1 \\ -1 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}+\begin{bmatrix} 0 & 1 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\imath,$$

$$V_{21}(0)=\begin{bmatrix} 0 & -1 & -1 \\ -1 & -1 & -1 \\ -1 & 1 & 1 \end{bmatrix}+\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & -1 \end{bmatrix}\imath, \quad X(0)=\begin{bmatrix} -3/2 & -3/2 & 5/2 \\ -3/2 & -5/2 & 4 \\ 3 & 4 & -4 \end{bmatrix},$$

the results of ZNNTVE-ARE are depicted in Figs. 2a-2d.

More precisely, Fig. 2a depicts the trajectories of $\|Z_i(t)\|_F$, i=1, 2, 3, of ZNNTVE-ARE. Fig. 2b depicts the trajectories of the ZNNTVE-ARE's solution $X^{ZNN}(t)$, decomposed into $X_r^{ZNN}(t)$ and $X_i^{ZNN}(t)$, along with the

optimal eigenvector solution $X^{\star}(t)$, decomposed into $X_r^{\star}(t)$ and $X_i^{\star}(t)$. Fig. 2c comprises of the eigenvalues produced by the ZNNTVE-ARE, which are decomposed into $\mathbf{c_r}^{ZNN}(t)$ and $\mathbf{c_i}^{ZNN}(t)$, along with the actual eigenvalues of the problem, decomposed into $\mathbf{c_r}^{\star}(t)$ and $\mathbf{c_i}^{\star}(t)$. Fig. 2d shows the error $\|\text{TV-ARE}\|_F$ produced from the ZNNTVE-ARE's solution $X^{ZNN}(t)$.

5.3. Example 3

Consider the input 2×2 matrices $P(t)$, $S(t)$ and $H(t)$ as bellow:

$$P(t) = \begin{bmatrix} -1/2\cos(2t) - 1 & 1/2\sin(2t) \\ 1/2\sin(2t) & 1/2\cos(2t) - 1 \end{bmatrix}, \quad S(t) = \begin{bmatrix} \sin(2t) & -\cos(2t) \\ \cos(2t) & \sin(2t) \end{bmatrix}.$$

and $H(t) = \mathbf{0}$. Since $H(t)$ is a zero matrix, ZNNTVE-ARE reduces to a CLE. The theoretical solution of the particular CLE is the following:

$$X^{\star}(t) = \begin{bmatrix} \frac{-(-2+\cos(2t))\sin(2t)}{3} & \frac{(1-2\cos(2t))(2+\cos(2t))}{6} \\ \frac{(2-\cos(2t))(1+2\cos(2t))}{6} & \frac{\sin(2t)(2+\cos(2t))}{3} \end{bmatrix}.$$

Given the round number to the nearest integer of the eigenvalues with a negative real part, we set $\mathbf{c}(0) = [-2, -1]^{\mathrm{T}}$. Then the results of ZNNTVE-ARE generated by the initial values

$$V_{11}(0) = I, \quad V_{21}(0) = X(0) = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

are depicted in Figs. 3a-3d.

More precisely, Fig. 3a depicts the trajectories of $\|Z_i(t)\|_F$, i=1, 2, 3, of ZNNTVE-ARE. Fig. 3b depicts the trajectories of the ZNNTVE-ARE's solution $X^{ZNN}(t)$, decomposed into $X_r^{ZNN}(t)$ and $X_i^{ZNN}(t)$, along with the theoretical solution $X^{\star}(t)$, decomposed into $X_r^{\star}(t)$ and $X_i^{\star}(t)$. Fig. 3c comprises of the eigenvalues produced by the ZNNTVE-ARE, which are decomposed into $\mathbf{c_r}^{ZNN}(t)$ and $\mathbf{c_i}^{ZNN}(t)$, along with the actual eigenvalues of the problem, decomposed into $\mathbf{c_r}^{\star}(t)$ and $\mathbf{c_i}^{\star}(t)$. Fig. 3d shows the error $\|\text{CLE}\|_F$ produced from the ZNNTVE-ARE's solution $X^{ZNN}(t)$.

5.4. Example 4

Consider the stabilization of a 2-dimensional LTV system. The open-loop system is described by

$$\mathrm{x}(t) = \begin{bmatrix} \mathrm{x}_1(t) \\ \mathrm{x}_2(t) \end{bmatrix}, \quad P(t) = \begin{bmatrix} 0 & 1 \\ -2 + \cos(\pi t) & \frac{1}{4}\sin(\pi t) \end{bmatrix},$$

and $L(t) = [0, \frac{1}{2}]^{\mathrm{T}}$. Fig. 4a depicts the open-loop response, that is unstable, and Fig. 4b depicts the phase portrait of the open-loop response. Furthermore, we set $S(t) = I$, $K(t) = 0.01$ and, hence, $H(t) = L(t)K^{-1}(t)L^{\mathrm{T}}(t)$. Given the round number to the nearest integer of the eigenvalues with a negative real part, we set $\mathbf{c}(0) = [-5, -1]^{\mathrm{T}}$. Then, for the initial values

$$V_{11}(0)=\begin{bmatrix} -1/4 & 1/2 \\ 1 & -1/2 \end{bmatrix}, \quad V_{21}(0)=\begin{bmatrix} 0 & 1/2 \\ 1/4 & 0 \end{bmatrix}, \quad X(0)=I, \quad \mathrm{x}(0)=[3,1]^{\mathrm{T}},$$

the results of HEZNN-LTVSS model are shown in Figs. 4c and 4d, in which Fig. 4c presents the close-loop responses, and Fig. 4d presents the phase portrait of the close-loop responses.

5.5. Example 5

In this example, we consider the stabilization of a 2-dimension LTV system. The open-loop system is described by

$$\mathrm{x}(t) = \begin{bmatrix} \mathrm{x}_1(t) \\ \mathrm{x}_2(t) \end{bmatrix}, \quad P(t) = \begin{bmatrix} 1/2\cos(t) & 1/2\cos(t)\sin(t) \\ -\sin(t) & -5+\sin(t) \end{bmatrix},$$

and $L(t) = [2, 1]^{\mathrm{T}}$. Fig. 5a depicts the open-loop response, that is unstable, and Fig. 5b depicts the phase portrait of the open-loop response. Furthermore, we set $S(t) = I$, $K(t) = 1$ and, hence, $H(t) = L(t)K^{-1}(t)L^{\mathrm{T}}(t)$. We set $\mathbf{c}(0) = [-5, -2]^{\mathrm{T}}$. Then, for the initial values

$$V_{11}(0)=\begin{bmatrix} 0 & 3/4 \\ -1 & -1/3 \end{bmatrix}, \quad V_{21}(0)=\begin{bmatrix} 0 & 1/2 \\ 0 & 0 \end{bmatrix}, \quad X(0)=I, \quad \mathrm{x}(0)=[3,1]^{\mathrm{T}},$$

the results of HEZNN-LTVSS model are shown in Figs. 5c and 5d, in which Fig. 5c presents the close-loop responses, and Fig. 5d presents the phase portrait of the close-loop responses.

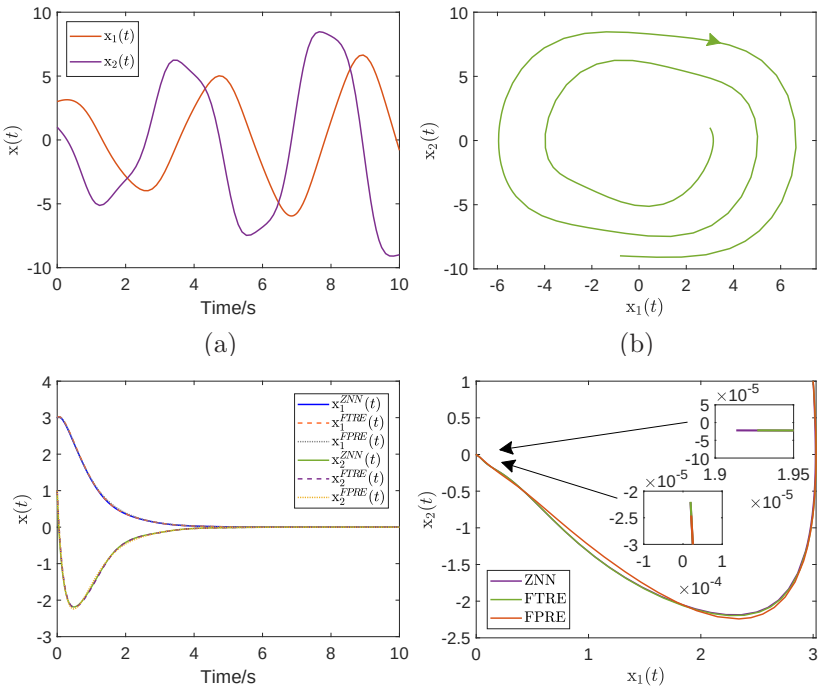

Figure 4: Performance of the HEZNN-LTVSS, FTRE and FPRE models for solving example 5.4.
(4a-4d): The open-loop response and their phase portrait, the close-loop responses and their phase portraits, respectively, of the system included in example (5.4).

5.6. Example 6

In this example, we consider the stabilization of a 4-dimensional LTV system. The open-loop system is described by

$$\mathrm{x}(t) = \begin{bmatrix} \mathrm{x}_1(t) \\ \mathrm{x}_2(t) \\ \mathrm{x}_3(t) \\ \mathrm{x}_4(t) \end{bmatrix}, \quad L(t) = \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix},$$

$$P(t) = \begin{bmatrix} 3+\sin(t) & 5-\sin(t) & 4-\cos(2t) & 0 \\ 5-\sin(t) & 4+\sin(2t) & -3-\sin(t) & 0 \\ 4-\cos(2t) & -3-\sin(t) & 4 & 0 \\ 0 & 0 & 0 & 3+\sin(t) \end{bmatrix},$$

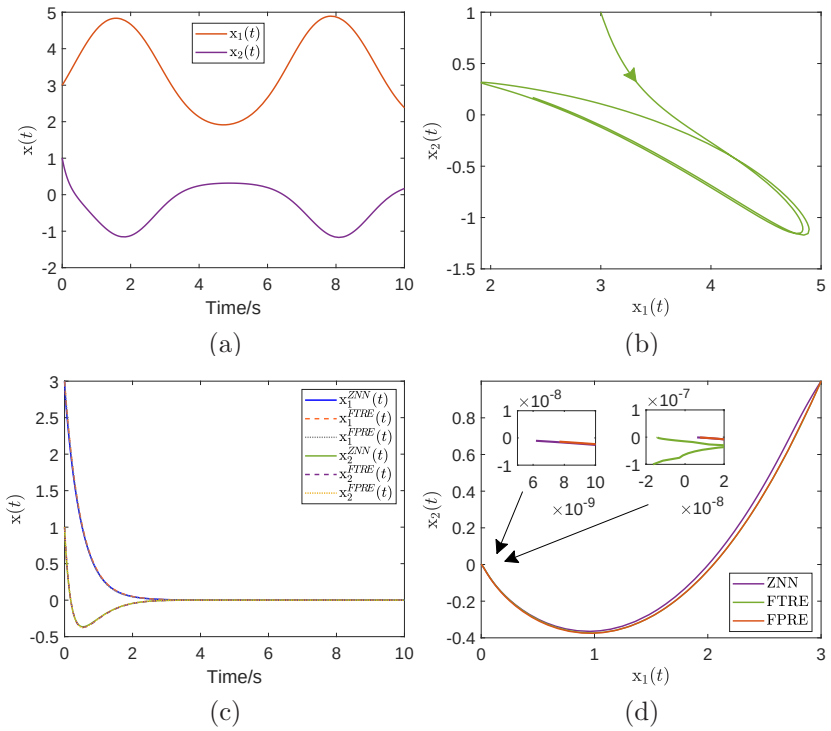

Figure 5: Performance of the HEZNN-LTVSS, FTRE and FPRE models for solving example 5.5.
(5a-5d): The open-loop response and their phase portrait, the close-loop responses and their phase portraits, respectively, of the system included in example (5.5).

Fig. 6a depicts the open-loop response, that is unstable, and Fig. 5b depicts the phase portrait of the open-loop response. Furthermore, we set $S(t) = I$, $K(t) = 1$ and, hence, $H(t) = L(t)K^{-1}(t)L^{\mathrm{T}}(t)$. We set $\mathbf{c}(0), V_{11}(0), V_{21}(0)$ the round number of their optimal value. Then, for the initial values

$$\mathrm{x}(0) = [0.02, 0.03, 0.04, 0.05]^{\mathrm{T}},$$

the results of the HEZNN-LTVSS model are shown in Figs. 6d-6f, in which Fig. 6d presents the close-loop responses, and Figs. 6e-6f present the phase portrait of the close-loop responses.

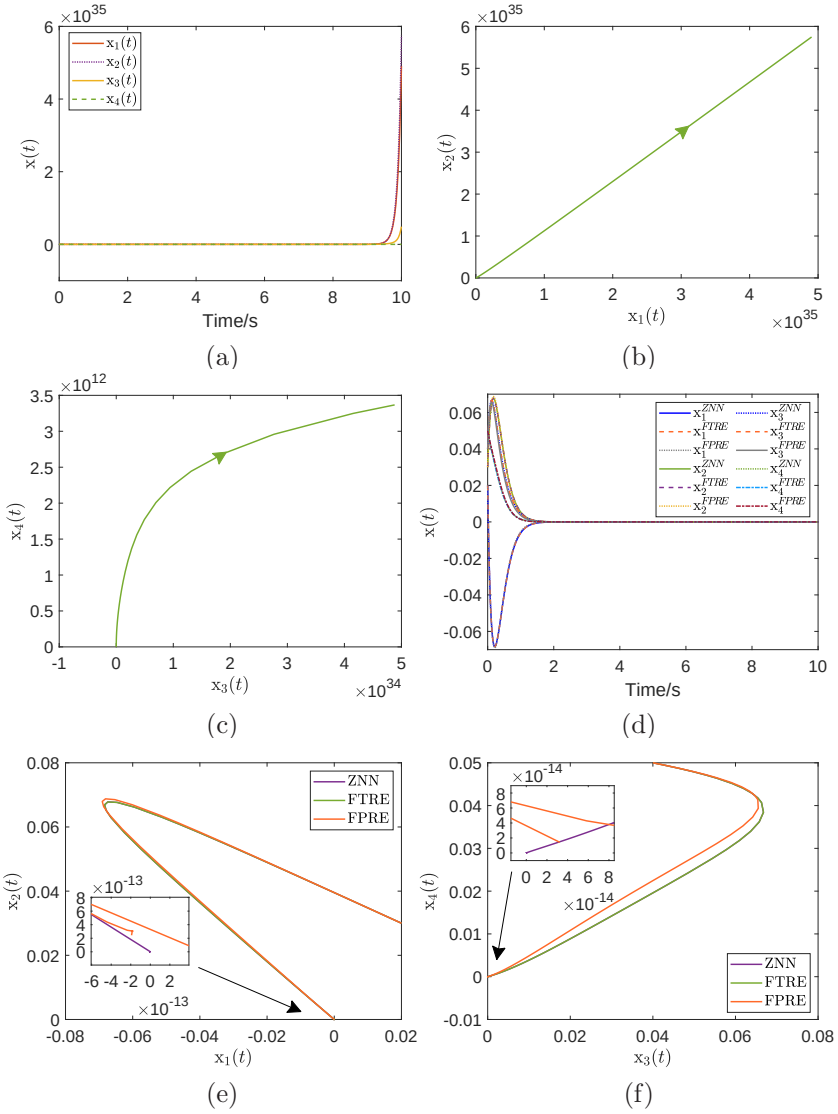

Figure 6: Performance of the HEZNN-LTVSS, FTRE and FPRE models for solving example 5.6.
(6a-6f): The open-loop response and their phase portrait, the close-loop responses and their phase portraits, respectively, of the system included in example (5.6).

5.7. Analysis of Obtained Numerical Results

In this subsection, the results produced by the ZNN model in examples 5.1-5.6 are commented on and analyzed. The TV-ARE problem (2.5) is

approached by the ZNNTVE-ARE model in examples 5.1-5.2, while (2.12) is approached by the ZNNTVE-ARE model in example 5.3, and (2.1) is approached by the HEZNN-LTVSS model in examples 5.4-5.6.

For the examples 5.1-5.3, Figs. 1a, 2a, 3a depict the trajectories of $\|Z_i(t)\|_F$, i=1, 2, 3, measured in ZNNTVE-ARE. Since all the errors therein rapidly converge to zero, (2.14) is satisfied. Furthermore, Figs. 1d, 2d show the error $\|\text{TV-ARE}\|_F$, and Fig. 3d shows the error $\|\text{CLE}\|_F$. Therein, we observe that the errors, which are produced from the ZNNTVE-ARE's solution $X^{ZNN}(t)$, rapidly converge to zero. That is, (2.5) is satisfied in examples 5.1-5.2 and (2.12) is also satisfied in example 5.3. In addition, Figs. 1b, 2b, 3b depict the trajectories of the ZNNTVE-ARE's solutions $X^{ZNN}(t)$ along with the trajectories of the optimal eigenvector solutions $X^{\star}(t)$. Therein, all the solutions are decomposed into the real and imaginary parts, and we observe that the ZNNTVE-ARE's solutions are identical to the optimal eigenvector solutions in examples 5.1-5.2 and to the theoretical solution in example 5.3. Also, Figs. 1c, 2c, 3c comprise of the eigenvalues produced by the ZNNTVE-ARE along with the corresponding actual eigenvalues. Therein, all the eigenvalues are decomposed into the real and imaginary parts, and we observe that the eigenvalues produced by the ZNNTVE-ARE are identical to the corresponding actual eigenvalues.

For the examples 5.4-5.6, Figs. 4a, 5a, 6a show the open-loop responses of the LTV systems of examples 5.4-5.6, respectively, and Figs. 4b, 5b, 6b, 6c show their phase portrait. Therein, we observe that the LTV systems are unstable. In addition, Figs. 4c, 5c, 6d show the close-loop responses of the LTV systems generated by the HEZNN-LTVSS, FTRE and FPRE models of examples 5.4-5.6, respectively, where we observe that the LTV systems are stabilized. Figs. 4d, 5d, 6e, 6f show their phase portrait, where we observe that the HFTZNN-LTVSS model provides always a slightly better asymptotic stability than the FTRE and FPRE models.

Overall, we can conclude that the ZNNTVE-ARE model works excellently in solving the TV-ARE (2.5) and the CLE (2.12). Moreover, the HEZNN-LTVSS model also works excellently in stabilizing the LTV system (2.1). Note that the value of the ZNN design parameter β regulates the convergence speed.

6. Conclusion

This paper introduces the ZNN approach for solving the TV-ARE problem based on the eigendecomposition approach. In line with the ZNN method, an appropriate error matrix was determined to solve the TV-ARE problem, and on that basis, the ZNNTVE-ARE model was proposed. Since the Lyapunov equation is a a particular case of ARE, the ZNNTVE-ARE model can also produce the eigenvector solution of CLE. In addition, a hybrid ZNN model for the LTV systems stabilization is provided. This model is called HEZNN-LTVSS and comprises the ZNNTVE-ARE model to solve the CARE related to the optimal control law. Six explicit examples, two of which contain an applications to TV-ARE problems, one contains application to CLE problem, and the last three contain applications to LTV system stabilization problems, are efficiently solved by the proposed models. In this way, the efficacy of the proposed models for solving the TV-ARE, CLE, and LTV systems stabilization problems are examined. The inference is that the ZNNTVE-ARE and HEZNN-LTVSS models are usable and effective in solving the TV-ARE, CLE as well as in LTV systems stabilization problems, and that the HFTZNN-LTVSS model always provides slightly better asymptotic stability than the models from which it is derived.

Some future study areas can be identified.

1. The ZNNTVE-ARE and HEZNN-LTVSS streams accelerated by a nonlinear activation function can be investigated. In this direction, there is a possibility of studying nonlinear ZNNTVE-ARE and HEZNN-LTVSS flows with a terminal convergence.
2. One of the perspectives for future work is the development of ZNNTVE-ARE and HEZNN-LTVSS streams with finite-time convergence.
3. The proposed ZNNTVE-ARE and HEZNN-LTVSS models have the disadvantage of being noise-intolerant because all noise types substantially impact the accuracy of the proposed ZNN approaches. As a result, future research could focus on adapting these models to an integration-enhanced and noise-handling ZNN class of dynamical systems.

Acknowledgements. Predrag S. Stanimirović is supported by the Ministry of Education, Science and Technological Development, Republic of Serbia (No. 451-03-68/2022-14/200124).
Predrag S. Stanimirović is supported by the Science Fund of the Republic of Serbia, (No. 7750185, Quantitative Automata Models: Fundamental Problems and Applications - QUAM).

References

[1] X. Dong, G. Hu, Time-varying formation tracking for linear multiagent systems with multiple leaders, IEEE Transactions on Automatic Control 62 (7) (2017) 3658–3664.

[2] P. Seiler, R. M. Moore, C. Meissen, M. Arcak, A. Packard, Finite horizon robustness analysis of LTV systems using integral quadratic constraints, Automatica 100 (2019) 135–143.

[3] B. Qin, H. Sun, J. Ma, W. Li, T. Ding, Z. Wang, A. Y. Zomaya, Robust H_∞ control of doubly fed wind generator via state-dependent Riccati equation technique, IEEE Transactions on Power Systems 34 (3) (2019) 2390–2400.

[4] L. Wang, Numerical algorithms of the discrete coupled algebraic Riccati equation arising in optimal control systems, Mathematical Problems in Engineering 2020 (2020) 1–8. doi:10.1155/2020/1841582.

[5] O. Koç, G. Maeda, J. Peters, Optimizing the execution of dynamic robot movements with learning control, IEEE Transactions on Robotics 35 (4) (2019) 909–924. doi:10.1109/TRO.2019.2906558.

[6] J. Liu, J. Zhang, F. Luo, Newton's method for the positive solution of the coupled algebraic Riccati equation applied to automatic control, Comp. Appl. Math. 39 (2020) 113. doi:10.1007/s40314-020-01143-5.

[7] J. Liu, L. Wang, Y. Bai, New estimates of upper bounds for the solutions of the continuous algebraic Riccati equation and the redundant control inputs problems, Automatica 116 (2020) 108936. doi:10.1016/j.automatica.2020.108936.

[8] R. E. Kalman, Contributions to the theory of optimal control, Boletin de la Sociedad Matematica Mexicana 5 (1960) 102–119.

[9] Y. Zhang, C. Yi, Zhang neural networks and neural-dynamic method, Nova Science Publishers, Inc., New York, 2011.

[10] B. Liao, Y. Zhang, Different complex ZFs leading to different complex ZNN models for time-varying complex generalized inverse matrices, IEEE Transactions on Neural Networks and Learning Systems 25 (9) (2014) 1621–1631.

[11] 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.

[12] P. S. Stanimirović, V. N. Katsikis, S. Li, Hybrid GNN-ZNN models for solving linear matrix equations, Neurocomputing 316 (2018) 124–134.

[13] P. S. Stanimirović, V. N. Katsikis, S. Li, Higher-order ZNN dynamics, Neural Processing Letters 51 (2020), 697–721.

[14] L. Xiao, B. Liao, S. Li, Z. Zhang, L. Ding, L. Jin, Design and analysis of FTZNN applied to the real-time solution of a nonstationary Lyapunov equation and tracking control of a wheeled mobile manipulator, IEEE Transactions on Industrial Informatics 14 (1) (2018) 98–105.

[15] P. S. Stanimirović, V. N. Katsikis, D. Gerontitis, A new varying-parameter design formula for solving time-varying problems, Neural Processing Letters (2020). doi:10.1007/s11063-020-10386-6.

[16] V. N. Katsikis, S. D. Mourtas, P. S. Stanimirović, Y. Zhang, Solving complex-valued time-varying linear matrix equations via QR decomposition with applications to robotic motion tracking and on angle-of-arrival localization, IEEE Transactions on Neural Networks and Learning Systems (2021) Pub Date: 2021-01-29. doi:10.1109/tnnls.2021.3052896.

[17] L. Jin, Y. Zhang, S. Li, Y. Zhang, Modified ZNN for time-varying quadratic programming with inherent tolerance to noises and its application to kinematic redundancy resolution of robot manipulators, IEEE Transactions on Industrial Electronics 63 (11) (2016) 6978–6988. doi:10.1109/TIE.2016.2590379.

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

[19] Y. Zhang, L. Jin, D. Guo, Y. Yin, Y. Chou, Taylor-type 1-step-ahead numerical differentiation rule for first-order derivative approximation and ZNN discretization, Journal of Computational and Applied Mathematics 273 (2015) 29–40.

[20] P. S. Stanimirović, V. N. Katsikis, Z. Zhang, S. Li, J. Chen, M. Zhou, Varying-parameter Zhang neural network for approximating some expressions involving outer inverses, Optimization Methods and Software 35 (6) (2019) 1–27.

[21] P. S. Stanimirović, V. N. Katsikis, S. Li, Integration enhanced and noise tolerant ZNN for computing various expressions involving outer inverses, Neurocomputing 329 (2019) 129–143.

[22] V. N. Katsikis, S. D. Mourtas, P. S. Stanimirović, Y. Zhang, Continuous-time varying complex QR decomposition via zeroing neural dynamics, Neural Processing Letters 53 (2021) 3573–3590.

[23] L. Xiao, *A new design formula exploited for accelerating Zhang neural network and its application to time-varying matrix inversion*, Theor. Comput. Sci. **647** (2016) 50–58.

[24] Y. Kong, H. Lu, Y. Xue, H. Xia, *Terminal neural computing: finite convergence and its applications*, Neurocomputing **217** (2016) 133–141.

[25] L. Xiao, Y. Zhang, K. Li, B. Liao, Z. Tan, *A novel recurrent neural network and its finite-time solution to time-varying complex matrix inversion*, Neurocomputing **331** (2019) 483–492.

[26] L. Xiao, B. Liao, S. Li, Z. Zhang, L. Ding, L. Jin, *Design and analysis of FTZNN applied to the real-time solution of a nonstationary Lyapunov equation and tracking control of a wheeled mobile manipulator*, IEEE Transactions on Industrial Informatics **14** (2017) 98–105.

[27] J. Jin, L. Xiao, M. Lu, J. Li, *Design and analysis of two FTRNN models with application to time-varying Sylvester equation*, IEEE Access **7** (2019) 58945–58950.

[28] D. Gerontitis, R. Behera, P. Tzekis, P. Stanimirović, *A family of varying-parameter finite-time zeroing neural networks for solving time-varying Sylvester equation and its application*, Journal of Computational and Applied Mathematics 403 (2022) doi: 10.1016/j.cam.2021.113826.

[29] J. Dai, Y. Li, L. Xiao, L. Jia, Q. Liao, and J. Li, *Comprehensive study on complex-valued ZNN models activated by novel nonlinear functions for dynamic complex linear equations*, Information Sciences 561 (2021) 101–114.

[30] J. Jin, and J. Gong, *A noise-tolerant fast convergence ZNN for dynamic matrix inversion*, International Journal of Computer Mathematics **98** (2021) 2202–2219.

[31] L. Xiao, *A nonlinearly activated neural dynamics and its finite-time solution to time-varying nonlinear equation*, Neurocomputing 173 (2016) 1983–1988.

[32] 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.

[33] L. Jin, Y. Zhang, S. Li, Integration-enhanced Zhang neural network for real-time-varying matrix inversion in the presence of various kinds of noises, IEEE Transactions on Neural Networks and Learning Systems, 27 (2016) 2615–2627.

[34] H. Liu, T. Wang, D. Guo, Design and validation of zeroing neural network to solve time-varying algebraic Riccati equation, IEEE Access 8 (2020) 211315–211323. doi:10.1109/ACCESS.2020.3039253.

[35] C. P. Mracek, J. R. Cloutier, Control designs for the nonlinear benchmark problem via the state-dependent Riccati equation method, International Journal of robust and nonlinear control 8 (4-5) (1998) 401–433.

[36] E. B. Erdem, A. G. Alleyne, Design of a class of nonlinear controllers via state dependent Riccati equations, IEEE Transactions on Control Systems Technology 12 (1) (2004) 133–137.

[37] W. M. Wonham, On a matrix Riccati equation of stochastic control, SIAM Journal on Control 6 (4) (1968) 681–697.

[38] S. Marshall, H. Nicholson, Optimal control of linear multivariable systems with quadratic performance criteria, in: Proceedings of the Institution of Electrical Engineers, Vol. 117, IET, 1970, pp. 1705–1713.

[39] A. Laub, A Schur method for solving algebraic Riccati equations, IEEE Transactions on Automatic Control 24 (6) (1979) 913–921.

[40] J. E. Potter, Matrix quadratic solutions, SIAM Journal on Applied Mathematics 14 (3) (1966) 496–501.

[41] S. Li, Y. Li, Nonlinearly activated neural network for solving time-varying complex Sylvester equation, IEEE Transactions on Cybernetics 44 (8) (2014) 1397–1407. doi:10.1109/TCYB.2013.2285166.

[42] J. Dai, Y. Li, L. Xiao, L. Jia, Q. Liao, J. Li, Comprehensive study on complex-valued ZNN models activated by novel nonlinear functions for dynamic complex linear equations, Information Sciences 561 (2021) 101–114. doi:10.1016/j.ins.2020.12.078.

[43] Y. Zhang, J. Wang, Recurrent neural networks for nonlinear output regulation, Automatica 37 (8) (2001) 1161–1173. doi:10.1016/s0005-1098(01)00092-9.

[44] A. K. Gupta, Numerical methods using MATLAB, MATLAB solutions series, Berkley: Springer Press., New York, NY, 2014.

[45] Y. Zhang, S. S. Ge, Design and analysis of a general recurrent neural network model for time-varying matrix inversion, IEEE Transactions on Neural Networks 16 (6) (2005) 1477–1490.

[46] A. Prach, O. Tekinalp, D. S. Bernstein, A numerical comparison of frozen-time and forward-propagating Riccati equations for stabilization of periodically time-varying systems, in: Proc. American Control Conf, 2014, pp. 5633–5638.