Neurocomputing 407 (2020) 465–479

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Time-varying generalized tensor eigenanalysis via Zhang neural networks

Changxin Mo [a,1], Xuezhong Wang [b,2], Yimin Wei [c,*,3]

[a] School of Mathematical Sciences, Fudan University, Shanghai 200433, PR China
[b] School of Mathematics and Statistics, Hexi University, Zhangye 734000, PR China
[c] School of Mathematical Sciences and Key Laboratory of Mathematics for Nonlinear Sciences, Fudan University, Shanghai 200433, PR China

ARTICLE INFO

Article history:
Received 26 October 2019
Revised 10 February 2020
Accepted 24 April 2020
Available online 19 May 2020
Communicated by Long Cheng

AMS subject classifications:
15A18
15A69
65H17
65L12
92B20

Keywords:
Time-varying tensor
Eigenvalue and eigenvector
Z-eigenvalue
H-eigenvalue
Time-varying matrix
Generalized eigenvalues
Zhang neural network
Zhang dynamics model

ABSTRACT

Eigenanalysis of matrices with parameters has a long history. When the parameter is time, or the matrix is time-dependent, the Zhang neural networks for the time-varying matrix problem have been developed in recent years. Motivated by tensor generalized eigenvalues and the Zhang dynamics method, we investigate the time-varying eigenpair of symmetric tensors. A continuous Zhang dynamics model is given to compute the tensor eigenpairs, such as the H- and Z-eigenpairs. In order to accelerate the convergence, a modified Zhang dynamics model is also presented. Moreover, the generalized tensor/matrix eigenpairs could also be computed by the two proposed models. Theoretical analysis of the convergence and robustness are provided. We also test some numerical examples which illustrate that the two proposed models are effective.

1. Introduction

Eigenvalues and eigenvectors (eigenpairs) are essential in numerical linear algebra and of central importance for many topics of physics and engineering. Many researches has performed in this field since the 18th century. Generally, most researchers are concerned with the eigenpairs of real or complex matrices with fixed entries, i.e., without any parameter associated with each entry. Suppose that A is a square matrix of size n, nonzero $\mathbf{x}$ is a vector in $\mathbb{C}^n$, and λ is a scalar in $\mathbb{C}$, then we say λ is an eigenvalue with eigenvector $\mathbf{x}$, if

$$A\mathbf{x} = \lambda\mathbf{x}.$$

The matrix A can be thought of as a linear transformation, in which case we can consider the general case, from $\mathbb{C}^n$ into $\mathbb{C}^n$. Eigenpairs are always utilized in many area, such as vibration analysis, graph theory, control theory, electric circuits, advanced dynamics and quantum mechanics; see references [10,20,37,49,60].

How do we go about finding the eigenvalues of a matrix? Theoretically, we seek roots of the characteristic polynomial $p_A(t) = \det(tI - A)$.However, in view of the unsolvability of polynomials of degree more than four by the Galois theory [28], direct and

* Corresponding author.
E-mail addresses: cxmo16@fudan.edu.cn (C. Mo), ymwei@fudan.edu.cn (Y. Wei).

[1] This author is supported in part by Promotion Program of Excellent Doctoral Research, Fudan University (SSH6281011/001) and the National Natural Science Foundations of China under grant 11771099.
[2] This author is supported by National Natural Science Foundation of China under grant 11771099, Natural Science Foundation of Gansu Province and Innovative Ability Promotion Project in Colleges and Universities of Gansu Province 2019B-146.
[3] This author is supported by the National Natural Science Foundations of China under grant 11771099 and Innovation Program of Shanghai Municipal Education Commission.

https://doi.org/10.1016/j.neucom.2020.04.115
0925-2312/

iterative methods are proposed and analyzed. The monographs by Wilkinson et al. [61] and Demmel [16] introduce many results about those methods, such as the Jacobi type method, the power method, QR and QL methods, the inverse iteration, Rayleigh Quotient iteration, Divide-and-Conquer method and so on. Up to now a wide range of results about matrix eigenvalue have been presented by many researchers; see the classical monographs, such as *Matrix Computations* [21], *Matrix Analysis* [23], *Matrix Perturbation Theory*[46], *Introduction to Linear Algebra* [47] for more details.

1.1. Parametric or time-varying problem

Those problems that involve matrices without parameters are called non-parametric matrix problems, or static-time or time-invariant problems, since the matrices keep the same as time evolves. What would happen if the entries of the matrix are dependent of parameters? Just like

$$A(p)\mathbf{x}(p) = \lambda(p)\mathbf{x}(p)$$

where the components of p can be some perturbation parameters, or simply the components of the matrix $A(p)$. As far as we know, the eigenvalues problem of matrices depending on parameters could be dated back to 1937 [44]. Moreover, the sensitivity analysis of eigenpairs of matrices dependent on parameters are very important in many engineering applications such as the optimum design of dynamical structures, model updating, damage detecting, quantum mechanics, diffraction grating theory, medical imaging, and social network theory. Many researchers consider matrices depending on one parameter; for example, Lancaster [32]. The classical monograph *Perturbation Theory for Linear Operators* by Kato [27] gave a systematic presentation of perturbation theory for linear operators. Matrices dependent on several parameters have also been studied by Sun [48], Chu [14], Mailybaev [35] and others.

If the parameter is time, then they are called time-varying problems which are of great importance in engineering, such as for robot optimization processes. Previous efforts to find the eigenstructure of time-varying matrix flows, mostly via differential equations solvers and path following continuation go back to 1991 [2], where the smoothness of the eigendata and SVD data of a time-dependent matrix flow has been explored in detail; see [2,35,45,50,67] and the references therein. Moreover, are the aforementioned methods reliable and effective if we perform them to solve the time-varying matrix problems? Unfortunately, these static methods are not effective and they can only work approximately with sizable residual errors and are not desirable [64]. Therefore, some new approaches are needed to deal with these challenges.

1.2. Zhang neural networks

A special class of powerful dynamic methods, which are designed specifically for time-varying problems, has been developed by Zhang et al. [65,68]. These methods are usually called the Zhang neural networks (ZNN) or zeroing neural networks; see references [50,65,68] for more details. So far, the Zhang neural networks have been applied to both time-varying and time-invariant problems, such as linear systems, matrix equations, matrix inversions and pseudoinverse [51,59], generalized eigenvalue problems [53], matrix minimization problems [65,68], Yang-Baxter-like matrix equation [63], the Takagi factorization [54], geometric measures of entanglement in multipartite pure states [8], the tensor complementarity problem [52], the tensor singular value decomposition [36,56] and the best rank-one approximations of tensors [7]. Generally speaking, the ZNN methods have two key parts. One is the error function which decays exponentially to zero over time. The other is that it uses the one-step ahead convergent finite difference equations, rather than the standard ODE initial value problem solvers. The former guarantees the convergence and efficiency of the methods, and the latter implies that it may be implemented on-chip which makes it more practical [68]. In practice, the first-order time derivatives have been used to construct the model and then discretized with one-step ahead backward differentiation formulas. This algorithm for matrix time-varying problems is effective. For the tensor case, Wang et al. [55] used this idea to solve multi-linear system [17,19] and compute time-varying Drazin inverses [42,57].

ZNN methods which build on the original ideas of Zhang and Wang [66] have been studied since 2001. Compared with other gradient-based neural networks (GNN) which are designed intrinsically for problems with constant matrices and/or vectors, ZNN methods are new approaches tailored to time-varying problems. Different from conventional GNN, ZNN models make full use of the time-derivative information of the coefficient matrices and vectors during its real-time solving process, while GNN models have not exploited such important information, thus may not be effective on solving the time-varying problems. Moreover, implicit dynamic equations in ZNN could preserve physical parameters in the coefficient/mass matrix and describe the usual and unusual parts of a dynamic system in the same form, and in this sense implicit systems which ZNN used are much superior to the systems represented by explicit dynamics used by GNN. On the other hand, as a predictive approach rather than adapting to the change of the coefficient matrix in a posterior passive manner, ZNN performs better in catching the exact solution. Many examples are given to show the superiority of the ZNN in the time-varying problems; see chapters 2 to 8 of [68].

Recently, based on the idea of the first-order time derivatives, a new continuous Zhang dynamics (ZD) model has been proposed to compute the eigenpairs of a time-varying real symmetric matrix by Zhang et al. [67]. Shortly afterwards, a new ZD model, based on the look-ahead finite difference equations, has been studied by Uhlig and Zhang [50]. The convergence of the two ZD models is studied and numerical experiments demonstrate the efficiency of those algorithms for computing the time-varying matrix eigenpairs.

1.3. Tensor eigenvalue problem

Tensor eigenvalues have attracted considerable attention when Qi [39] and Lim [34] proposed the concepts of eigenvalue and eigenvector in 2005. Tensors are generalizations of matrices to higher dimensions and can consequently be treated as multidimensional fields. Usually, we call a matrix an order-two tensor and vector an order-one tensor. For more information, refer to the tensor monographs [9,40,41,58]. Unlike the matrix case, there are two different real eigenvalue definitions of tensors, the *H*-eigenvalue and *Z*-eigenvalue. Firstly, we give some notations.

Let $\mathbb{R}$ and $\mathbb{C}$ be the real and complex fields, respectively. We use small letters $a, b, \ldots$ for scalars, small bold letters $\mathbf{x}, \mathbf{y}, \ldots$ for vectors, capital letters $A, B, \ldots$ for matrices and calligraphic letters $\mathcal{A}, \mathcal{B}, \ldots$ for tensors. An m-order n-dimensional real tensor $\mathcal{A} = (a_{i_1 \cdots i_m})$, where $a_{i_1 \cdots i_m} \in \mathbb{R}$ with $i_j = 1, 2, \ldots, n$ for $j = 1, 2, \ldots, m$, is usually denoted by $\mathcal{A} \in \mathbb{R}^{[m,n]}$. The notation $\mathcal{A}(t) = (a_{i_1 \cdots i_m}(t))$ denotes a tensor flow and $\mathcal{A}(t) \in \mathbb{R}^{[m,n]}$ means that it is an m-order n-dimensional tensor flow and $a_{i_1 \cdots i_m}(t) \in \mathbb{R}$ for each time t.

We are now ready for the definitions of *H*- and *Z*-eigenvalues.

Definition 1.1. ([34,39]). We call λ an *H*-eigenvalue of the tensor $\mathcal{A} \in \mathbb{R}^{[m,n]}$, if there exists a nonzero vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)^\top \in \mathbb{R}^n$ such that

$$\mathcal{A}\mathbf{x}^{m-1} = \lambda \mathbf{x}^{[m-1]},$$

in which $\mathcal{A}\mathbf{x}^{m-1}$ is a vector in $\mathbb{R}^n$ and

$$\left(\mathcal{A}\mathbf{x}^{m-1}\right)_i = \sum_{i_2,\ldots,i_m=1}^{n} a_{ii_2\cdots i_m} x_{i_2}\cdots x_{i_m}$$

for all i and $\mathbf{x}^{[m-1]}$ is a vector in $\mathbb{R}^n$ with entries $\left(\mathbf{x}^{[m-1]}\right)_i = x_i^{m-1}$ for all i. We call λ an Z-eigenvalue of the tensor $\mathcal{A} \in \mathbb{R}^{[m,n]}$, if there exists a nonzero vector $\mathbf{x} \in \mathbb{R}^n$ such that

$$\begin{cases} \mathcal{A}\mathbf{x}^{m-1} = \lambda\mathbf{x}, \\ \mathbf{x}^\top\mathbf{x} = 1. \end{cases}$$

The vector $\mathbf{x}$ is called the corresponding eigenvector associated with λ.

However, the definitions of H- and Z-eigenvalues have been unified by Chang et al. [6] to a generalized tensor eigenvalue framework. It is worth noting that the generalized tensor eigenvalue is also an extension of the generalized matrix eigenvalue which is essential in scientific and engineering computations. Generally, the generalized tensor eigenvalue problem is to find the number λ and the vector $\mathbf{x}$ satisfying

$$\mathcal{A}\mathbf{x}^{m-1} = \lambda\mathcal{B}\mathbf{x}^{m-1}. \tag{1.1}$$

Similar to the matrix case, the generalized eigenvalue of the tensor pair has been defined by Ding and Wei [18]. The tensor pair $\{\mathcal{A},\mathcal{B}\}$ is called a singular tensor pair if $\det(\beta\mathcal{A} - \alpha\mathcal{B}) = 0$ for all $(\alpha,\beta) \in \mathbb{C}_{1,2} = \{(\alpha,\beta) \neq (0,0);\ \alpha,\beta \in \mathbb{C}\}$. The tensor pair $\{\mathcal{A},\mathcal{B}\}$ is called a regular tensor pair if $\det(\beta\mathcal{A} - \alpha\mathcal{B}) \neq 0$ for some $(\alpha,\beta) \in \mathbb{C}_{1,2}$. Moreover, when $\mathcal{B}$ is nonsingular, then $\beta \neq 0$ if (α,β) is an eigenvalue of $\{\mathcal{A},\mathcal{B}\}$, and in this case $\lambda = \alpha/\beta \in \mathbb{C}$ is called an eigenvalue of the tensor pair $\{\mathcal{A},\mathcal{B}\}$.

The computing of eigenpairs of real tensors has been studied by many researchers. For example, Cui et al. [15] propose an optimization algorithm to compute all real eigenvalues, and corresponding real eigenvectors, of a real symmetric tensor. Kolda and Mayo [30] provide an adaptive shifted power method for computing the generalized tensor eigenpairs. Chen et al. [11] present a homotopy method to compute tensor eigenpairs and a MATLAB software package, TenEig, has been developed.

Motivated by the idea of tensor generalized eigenvalues [18] and the ZD method proposed by Zhang et al. [67] for time-varying real symmetric matrices, we consider the following time-varying generalized tensor eigenpair in this paper.

Problem Statement: Given a real time-varying tensor pair $\{\mathcal{A}(t),\mathcal{B}(t)\}$ in which both $\mathcal{A}(t)$ and $\mathcal{B}(t)$ are m-order n-dimensional real tensors for each fixed time t, our objective is to find a number flow $\lambda(t) \in \mathbb{C}$ and a nonzero vector flow $\mathbf{x}(t) \in \mathbb{C}^n$ such that

$$\mathcal{A}(t)\mathbf{x}^{m-1}(t) = \lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t), \quad t \in [0,\infty), \tag{1.2}$$

in which $\mathcal{A}(t)\mathbf{x}^{m-1}(t)$ is a vector (depending on time t) defined similarly as the time-invariant case, that is

$$\left(\mathcal{A}(t)\mathbf{x}^{m-1}(t)\right)_i = \sum_{i_2=1}^{n}\cdots\sum_{i_m=1}^{n} a_{ii_2\cdots i_m}(t) x_{i_2}(t)\cdots x_{i_m}(t), \quad i = 1,2,\cdots,n.$$

In this case, we call the flow $\lambda(t)$ an eigenvalue of the tensor pair $\{\mathcal{A}(t),\mathcal{B}(t)\}$, and the nonzero $\mathbf{x}(t)$ is the corresponding eigenvector associated with $\lambda(t)$.

(1) The above time-varying generalized tensor eigenpair reduces to the time-varying real symmetric matrix proposed by Zhang et al. in [67] if $m = 2, \mathcal{B}(t)$ is the identity matrix and $\mathcal{A}(t)$ is symmetric.

(2) If we choose $\mathcal{B}(t)$ as the identity tensor $\mathcal{I}$, where its entries are $\delta_{i_1,\ldots,i_m}$ for $i_1,\ldots,i_m = 1,2,\ldots,n$, $\delta_{i_1,\ldots,i_m} = 1$ if $i_1 = \cdots = i_m$ and zero otherwise, defined by Qi in [39], then $\mathcal{B}(t)\mathbf{x}^{m-1}(t) = \mathcal{I}\mathbf{x}^{m-1}(t) = \mathbf{x}^{[m-1]}(t) = \left(x_1^{m-1}(t),\cdots,x_n^{m-1}(t)\right)^\top$. In this case, the above time-varying generalized tensor eigenpair can be rewritten by

$$\mathcal{A}(t)\mathbf{x}^{m-1}(t) = \lambda(t)\mathbf{x}^{[m-1]}(t)$$

which is equivalent to computing the corresponding H-eigenpairs of the time-varying tensor for each fixed t.

(3) In order to compute the Z-eigenpairs of the time-varying tensor for each fixed t, we have to add the condition $\|\mathbf{x}(t)\|_2 = 1$, and we just need to consider

$$\mathcal{A}(t)\mathbf{x}^{m-1}(t) = \lambda(t)\mathbf{x}(t).$$

Kolda and Mayo [29] define a symmetric even identity tensor $\mathcal{E}$ such that $\mathcal{B}(t)\mathbf{x}^{m-1}(t) = \mathcal{E}\mathbf{x}^{m-1}(t) = \|\mathbf{x}\|^{m-2}\mathbf{x} = \mathbf{x}$ and give the explicit expression. Actually, $\mathcal{E}$ could be nonsymmetric, as is given in [39] and then [38] in which it is called the Z-identity tensor. In a word, we only need to consider $\mathcal{A}(t)\mathbf{x}^{m-1}(t) = \lambda(t)\mathbf{x}(t)$ if we want to compute the Z-eigenpairs.

1.4. Contributions and outline

The main contributions of this paper are listed as follows:

(1) Time-varying generalized eigenanalysis for matrix/tensor is considered. We try to compute the generalized eigenpairs for time-varying matrix-pencil and H- and Z-eigenpairs for time-varying tensor. The theoretical results about convergence and robustness are given.
(2) Two dynamics models are given. By using different activation functions, the modified continuous Zhang dynamics model could usually obtain better convergence results compared with the original continuous Zhang dynamics model. Numerical examples are also tested to illustrate its superiority.

The rest of this paper is organized as follows. In Section 2, we provide preliminaries and introduce basic notations. In Section 3, the continuous Zhang dynamics is used to construct the model for the time-varying generalized tensor eigenpair and their convergence and robustness properties are analyzed. A modified model has also been proposed. In Section 4, we test some numerical experiments to illustrate the effectiveness of the two proposed models. We conclude our paper in Section 5.

2. Preliminaries

A tensor $\mathcal{A} \in \mathbb{R}^{[m,n]}$ is called symmetric [5,39] if its entries are invariant under any permutation of their indices. A tensor flow $\mathcal{A}(t) \in \mathbb{R}^{[m,n]}$ is called symmetric if its symmetric for each fixed t and it is called diagonal if all the off-diagonal entries are zero, that is, those $\mathcal{A}(t)(i_1,\ldots,i_m)$ with $\delta_{i_1,\ldots,i_m} = 0$. A tensor $\mathcal{A}$ is called nonsingular if $\det(\mathcal{A}) \neq 0$ where $\det(\mathcal{A})$ denotes the determinant of tensor $\mathcal{A}$ [24,39].

Next, we give the definitions of eigenvalue and eigenvector of a time-varying real tensor pair $\{\mathcal{A}(t),\mathcal{B}(t)\}$.

Definition 2.1. Let $\mathcal{A}(t)$ and $\mathcal{B}(t)$ be two m-order n-dimensional time-varying tensors on $\mathbb{R}$. Assume that both $\mathcal{A}(t)\mathbf{x}^{m-1}(t)$ and $\mathcal{B}(t)\mathbf{x}^{m-1}(t)$ are not identically zero. If the following n-system time-dependent equations

$$[\mathcal{A}(t) - \lambda(t)\mathcal{B}(t)]\mathbf{x}^{m-1}(t) = 0, \tag{2.1}$$

i.e.,

$$\sum_{i_2,\ldots,i_m=1}^{n}[\mathcal{A}(i,i_2,\ldots,i_m)(t)-\lambda(t)\mathcal{B}(i,i_2,\ldots,i_m)(t)]x_{i_2}(t)\cdots x_{i_m}(t)=0,\quad i=1,2,\ldots,n,$$

possess a solution, then we say that $(\lambda(t),x(t))$ is an eigenvalue–eigenvector of $\mathcal{A}(t)$ relative to $\mathcal{B}(t)$. Here $\lambda(t)$ is called a $\mathcal{B}_t$-eigenvalue of $\mathcal{A}(t)$, and $\mathbf{x}$ is called a $\mathcal{B}_t$-eigenvector of $\mathcal{A}(t)$ corresponding to $\lambda(t)$.

If $\mathcal{B}(t)\equiv\mathcal{I}$, then we call $\lambda(t)$ an H_t-eigenvalue if (2.1) has a real solution $\mathbf{x}(t)$, i.e., $\mathcal{A}(t)$ has a real eigenvector.

If $\mathcal{B}(t)\equiv\mathcal{E}$ such that $\mathcal{E}\mathbf{x}^{m-1}(t)=\mathbf{x}(t)$ for even-order, and in odd-order case if $\mathcal{A}(t)\mathbf{x}^{m-1}(t)=\lambda(t)\mathbf{x}(t)$, then we call $\lambda(t)$ an Z_t-eigenvalue if they have a real solution $\mathbf{x}(t)$ satisfying $\|\mathbf{x}(t)\|_2=1$.

Remark 2.1. If $\mathcal{A}(t)$ and $\mathcal{B}(t)$ are time-invariant, then the definition is equivalent to that of the generalized tensor eigenpair [6, Definition 2.1]. Moreover, if the order is two, then it is just the matrix generalized eigenvalue.

Time-varying symmetric or Hermitian matrices have been considered by Uhlig [50] and Zhang et. [67]. In this paper, for simplicity, we only consider the real and symmetric tensor flows $\mathcal{A}(t)$ and $\mathcal{B}(t)$.

We assume that the entries of $\mathcal{A}(t)$ and $\mathcal{B}(t)$ are analytic, and we also assume that $\mathcal{B}(t)$ is nonsingular for all time t [18] and (2.1) has at least one real first-order differentiable solution, since we need this in constructing the model. Two lemmas are provided.

Lemma 2.1. *Suppose that $(\lambda(t),\mathbf{x}(t))$ is an eigenvalue–eigenvector of $\mathcal{A}(t)$ relative to $\mathcal{B}(t)$, then $(\lambda(t)),k\mathbf{x}(t))$ is also an eigenvalue-eigenvector of $\mathcal{A}(t)$ relative to $\mathcal{B}(t)$ for $k\neq 0$.*

Proof. For a given tensor flow $\mathcal{C}(t)$ and $\mathbf{y}(t)=k\mathbf{x}(t)$, we have

$$\mathcal{C}(t)\mathbf{y}^{m-1}(t)=k^{m-1}\mathcal{C}(t)\mathbf{x}^{m-1}(t).$$

It is easy to see that

$$\begin{aligned}\mathcal{A}(t)\mathbf{y}^{m-1}(t)&=k^{m-1}\mathcal{A}(t)\mathbf{x}^{m-1}(t)=k^{m-1}\lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t)\\&=\lambda(t)\mathcal{B}(t)\mathbf{y}^{m-1}(t),\end{aligned}$$

and we complete the proof. □

Remark 2.2. The above lemma illustrates that it is reasonable to assume that the norm of the $\mathcal{B}_t$-eigenvector is a constant. For the Z_t-eigenpair case, we have the natural condition that the Euclidean norm of the eigenvector is a constant.

Now, we give the definitions of scalar product and Frobenius norm of time-varying tensor flows.

Definition 2.2. Given two time-varying tensor flows $\mathcal{A}(t)$ and $\mathcal{B}(t)$ and for each t we have $\mathcal{A}(t),\mathcal{B}(t)\in\mathbb{R}^{I_1\times I_2\times\cdots\times I_m}$. The scalar product of $\mathcal{A}(t)$ and $\mathcal{B}(t)$, denoted by $\langle\mathcal{A}(t),\mathcal{B}(t)\rangle$, is defined as

$$\langle\mathcal{A}(t),\mathcal{B}(t)\rangle:=\sum_{i_1}\sum_{i_2}\cdots\sum_{i_m}a_{i_1i_2\cdots i_m}(t)b_{i_1i_2\cdots i_m}(t).$$

The Frobenius norm of $\mathcal{A}(t)$ is then defined as $\|\mathcal{A}(t)\|_F:=\sqrt{\langle\mathcal{A}(t),\mathcal{A}(t)\rangle}$.

Lemma 2.2. *Let the tensor flow $\mathcal{A}(t)\in\mathbb{R}^{[m,n]}$ and $\mathbf{x}(t)\in\mathbb{R}^n$ for each t, then*

$$\|\mathcal{A}(t)\mathbf{x}^{m-1}(t)\|_2\leqslant\|\mathcal{A}(t)\|_F\cdot\|\mathbf{x}(t)\|_2^{m-1}.$$

Proof. The proof is similar to that of [62, Lemma 2.1]. □

3. Zhang neural network formulation

We firstly give one continuous Zhang dynamics model for the time-varying generalized tensor eigenpair (1.2), and its convergence and robustness properties will be studied. In order to accelerate convergence of the continuous Zhang dynamics model, a modified Zhang dynamics model is presented.

3.1. Continuous Zhang dynamics model

To investigate the time-varying generalized tensor eigenpair (1.2), we define the vector-valued time-varying function which is always used in the Zhang neural network method as follows:

$$\mathbf{e}_1(t)=\mathcal{A}(t)\mathbf{x}^{m-1}(t)-\lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t).\tag{3.1}$$

We assume that the time-varying generalized tensor eigenpair (3.1) always has at least one real solution and we only consider the real case, thus we have $\mathbf{e}_1(t)\in\mathbb{R}^n$ for each t.

According to the main idea of the ZNN method, in order to force the entries of $\mathbf{e}_1(t)$ to converge quickly to zero, we stipulate that

$$\dot{\mathbf{e}}_1(t)=\frac{\mathrm{d}\mathbf{e}_1(t)}{\mathrm{d}t}=-\eta_1\mathbf{e}_1(t),\tag{3.2}$$

which is also called as the ZD design formula [67], and $\eta_1\gg 0$ is the design parameter which could be choose as large as possible if the hardware permits. Note that

$$\begin{aligned}\frac{\mathrm{d}\mathbf{e}_1(t)}{\mathrm{d}t}&=\dot{\mathcal{A}}(t)\mathbf{x}^{m-1}(t)+(m-1)\mathcal{A}(t)\mathbf{x}^{m-2}(t)\dot{\mathbf{x}}(t)\\&\quad-\left(\dot{\lambda}(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t)+\lambda(t)\dot{\mathcal{B}}(t)\mathbf{x}^{m-1}(t)+(m-1)\lambda(t)\mathcal{B}(t)\mathbf{x}^{m-2}(t)\dot{\mathbf{x}}(t)\right).\end{aligned}$$

Combining the above equality with (3.2) and rearranging the items, we obtain the following differential equation for variables $\mathbf{x}$ and λ,

$$\begin{aligned}&((m-1)(\mathcal{A}(t)-\lambda(t)\mathcal{B}(t))\mathbf{x}^{m-2}(t))\dot{\mathbf{x}}(t)-\mathcal{B}(t)\mathbf{x}^{m-1}(t)\dot{\lambda}(t)\\&=-\eta_1\left(\mathcal{A}(t)\mathbf{x}^{m-1}(t)-\lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t)\right)-\dot{\mathcal{A}}(t)\mathbf{x}^{m-1}(t)+\lambda(t)\dot{\mathcal{B}}(t)\mathbf{x}^{m-1}(t).\end{aligned}\tag{3.3}$$

On the other hand, in order to keep the eigenvector $\mathbf{x}(t)$ from becoming zero by chance in computation, by Lemma 2.1, we stipulate that the length of $\mathbf{x}(t)$ always remains the same constant, that is

$$\|\mathbf{x}(t)\|_2^2=\mathbf{x}^\top(t)\mathbf{x}(t)=1\tag{3.4}$$

for all times t. Another error function could be established as follows,

$$e_2(t)=\|\mathbf{x}(t)\|_2^2-1.\tag{3.5}$$

We can see that $e_2(t)\in\mathbb{R}$ for each fixed time t. Again, the ZD design formula is applied for quick convergence of the entries of $e_2(t)$,

$$\dot{e}_2(t)=\frac{\mathrm{d}e_2(t)}{\mathrm{d}t}=-\eta_2e_2(t),\tag{3.6}$$

where $\eta_2(\gg 0)$ is the design parameter and could be chosen similarly as η_1. Similarly, by (3.5) and (3.6), we have

$$2\mathbf{x}^\top(t)\dot{\mathbf{x}}(t)=-\eta_2\left(\mathbf{x}^\top(t)\mathbf{x}(t)-1\right).\tag{3.7}$$

An implicit-dynamics continuous ZD model for the time-varying generalized tensor eigenpair is obtained by combining Eqs. (3.3) and (3.7). Denote $\mathbf{z}^\top(t)=[\mathbf{x}^\top(t),\lambda(t)]$, then we get

$$M(t,\mathbf{z})\dot{\mathbf{z}}(t)=\mathbf{q}(t,\mathbf{z}), \tag{3.8}$$

where

$$M(t,\mathbf{z})=\begin{bmatrix}(m-1)(\mathcal{A}(t)-\lambda(t)\mathcal{B}(t))\mathbf{x}^{m-2}(t)) & -\mathcal{B}(t)\mathbf{x}^{m-1}(t)\\ 2\mathbf{x}^{\top}(t) & 0\end{bmatrix}\in\mathbb{R}^{(n+1)\times(n+1)},$$

$$\dot{\mathbf{z}}(t)=\begin{bmatrix}\dot{\mathbf{x}}(t)\\ \dot{\lambda}(t)\end{bmatrix}\in\mathbb{R}^{n+1},$$

and

$$\mathbf{q}(t,\mathbf{z})=\begin{bmatrix}-\eta_1(\mathcal{A}(t)\mathbf{x}^{m-1}(t)-\lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t))-\dot{\mathcal{A}}(t)\mathbf{x}^{m-1}(t)+\lambda(t)\dot{\mathcal{B}}(t)\mathbf{x}^{m-1}(t)\\ -\eta_2(\mathbf{x}^{\top}(t)\mathbf{x}(t)-1)\end{bmatrix}\in\mathbb{R}^{n+1}.$$

Three special cases of the model (3.8) will be obtained for different $\mathcal{A}(t)$ and $\mathcal{B}(t)$.

(1) *The matrix case.* If we choose the order of $\mathcal{A}(t)$ and $\mathcal{B}(t)$ as two, and let $\mathcal{B}(t)$ be the identity matrix, then the model (3.8) reduces to the model (7) of [67]. With minor correction, we have

$$\begin{aligned}M(t,\mathbf{z})&=\begin{bmatrix}A(t)-\lambda(t)\mathcal{I} & -\mathbf{x}(t)\\ -\mathbf{x}^{\top}(t) & 0\end{bmatrix},\\ \mathbf{q}(t,\mathbf{z})&=\begin{bmatrix}-\eta_1(A(t)-\lambda(t)I)\mathbf{x}(t)-\dot{A}(t)\mathbf{x}(t)\\ \frac{\eta_2}{2}(\mathbf{x}^{\top}(t)\mathbf{x}(t)-1)\end{bmatrix}.\end{aligned} \tag{3.9}$$

(2) *The Z_t-eigenpair tensor case.* As $\mathcal{A}(t)$ is symmetric, then $\left[\mathcal{A}(t)\mathbf{x}(t)^{m-2}\right]_{ij}=\left[\mathcal{A}(t)\mathbf{x}(t)^{m-2}\right]_{ji}$ since $[\mathcal{A}(t)]_{iji_3\ldots i_n}=[\mathcal{A}(t)]_{jii_3\ldots i_n}$, which means that $M(t)$ could be a symmetric time-varying matrix if we change (3.7) as follows,

$$-\mathbf{x}^{\top}(t)\dot{\mathbf{x}}(t)=\frac{\eta_2}{2}\left(\mathbf{x}^{\top}(t)\mathbf{x}(t)-1\right). \tag{3.10}$$

In this case, we have

$$M(t,\mathbf{z})=\begin{bmatrix}(m-1)\mathcal{A}(t)\mathbf{x}^{m-2}(t)-\lambda(t)I & -\mathbf{x}(t)\\ -\mathbf{x}^{\top}(t) & 0\end{bmatrix}\in\mathbb{R}^{(n+1)\times(n+1)}$$

and

$$\mathbf{q}(t,\mathbf{z})=\begin{bmatrix}-\eta_1\left(\mathcal{A}(t)\mathbf{x}^{m-1}(t)-\lambda(t)\mathbf{x}(t)\right)-\dot{\mathcal{A}}(t)\mathbf{x}^{m-1}(t)\\ \frac{\eta_2}{2}(\mathbf{x}^{\top}(t)\mathbf{x}(t)-1)\end{bmatrix}\in\mathbb{R}^{n+1}.$$

(3) *The H_t-eigenpair tensor case.* Let $\mathcal{B}(t)\equiv\mathcal{I}$ where $\mathcal{I}$ is the identity tensor. Then we have

$$M(t,\mathbf{z})=\begin{bmatrix}(m-1)(\mathcal{A}(t)-\lambda(t)\mathcal{I})\mathbf{x}^{m-2}(t) & -\mathbf{x}^{[m-1]}(t)\\ -\mathbf{x}^{\top}(t) & 0\end{bmatrix}\in\mathbb{R}^{(n+1)\times(n+1)} \tag{3.11}$$

and

$$\mathbf{q}(t,\mathbf{z})=\begin{bmatrix}-\eta_1\left(\mathcal{A}(t)\mathbf{x}^{m-1}(t)-\lambda(t)\mathbf{x}^{m-1}(t)\right)-\dot{\mathcal{A}}(t)\mathbf{x}^{m-1}(t)\\ \frac{\eta_2}{2}(\mathbf{x}^{\top}(t)\mathbf{x}(t)-1)\end{bmatrix}\in\mathbb{R}^{n+1}.$$

Now we can use model (3.8) to compute the H_t-eigenpairs.

Remark 3.1. We stipulate that the length of the eigenvector is always one in (3.4). Actually, we can choose any positive number as the length in any norm as we like, except that we need to compute the Z_t-eigenpairs. Moreover, there is another condition that could be used as an alternative, that is, let $\mathbf{y}$ be a fixed vector such that $\mathbf{y}^{\top}\mathbf{x}=1$ [13,14]. However, this needs more study and we will not consider it in this paper.

3.2. Theoretical analysis

Bordering for a singular matrix removes the singularity and can be dated back to 1935 [3]. Bordered matrices can be used to deal with the problem of computing the eigenvalues or pseudoinverse [51] of the matrix and approximate solution of homogenous systems. The paper by Blattner [1] used this technical to compute the eigenvalues and found the projections associated with a square matrix. Eric Chu gave more details in [13] and applied it to calculate stationary distribution vectors of ergodic Markov chains.

Let A be a sqaure matrix, and λ is an eigenvalue with right-eigenvector $\mathbf{x}$ and left-eigenvector $\mathbf{y}$, that is, $A\mathbf{x}=\lambda\mathbf{x},\mathbf{y}^{*}A=\lambda\mathbf{y}^{*}$. Furthermore, suppose that λ is a *simple* eigenvalue, without loss of generality, we add the restriction $\mathbf{y}^{*}\mathbf{x}=1$. Note that $A-\lambda I$ is singular, by using the right-eigenvector $\mathbf{x}$ and left-eigenvector $\mathbf{y}$, we obtain one bordered matrix

$$M=\begin{bmatrix}A-\lambda I & \mathbf{x}\\ \mathbf{y}^{*} & 0\end{bmatrix}.$$

It has been proved in [1,22] that the bordered matrix M is nonsingular. If A is symmetric, then an eigenvalue could share the same right-eigenvector and left-eigenvector. Thus we get the similar result for the time-varying case.

Lemma 3.1. *Let $A(t)$ be an n-by-n time-varying real symmetric matrix flow with simple eigenvalue flows $\lambda(t)$, and corresponding unitized real eigenvector flow $\mathbf{x}(t)$ for each fixed t, then the bordered matrix*

$$M_t=\begin{bmatrix}A(t)-\lambda(t)I & -\mathbf{x}(t)\\ -\mathbf{x}^{\top}(t) & 0\end{bmatrix}$$

is nonsingular.

Similarly, we also have the result for the generalized eigenvalue of matrix pair $(A(t),B(t))$.

Lemma 3.2. *Let $A(t)$ be an n-by-n time-varying real symmetric matrix flow and $B(t)$ is symmetric positive definite for each t. If $\lambda(t)$ is the simple eigenvalue flows with corresponding unit real eigenvector flow $\mathbf{x}(t)$ for each fixed t, then the bordered matrix*

$$M_t=\begin{bmatrix}A(t)-\lambda(t)B(t) & -\mathbf{x}(t)\\ -\mathbf{x}^{\top}(t) & 0\end{bmatrix}$$

is nonsingular.

Since the coefficient matrix in model (3.8) is not symmetric, we consider another condition about the eigenvector which is different from (3.4), that is,

$$\mathcal{B}(t)\mathbf{x}^{m}(t)=1.$$

We suppose that $\mathcal{B}(t)$ is symmetric positive definite in which case the eigenvector $\mathbf{x}(t)$ will not equal zero. The error function is

$$e_3(t)=\mathcal{B}(t)\mathbf{x}^{m}(t)-1.$$

Again, we can see that $e_3(t)\in\mathbb{R}$ for each fixed time t and the ZD design formula is applied for quick convergence, thus we have

$$\dot{e}_3(t)=\frac{\mathrm{d}e_3(t)}{\mathrm{d}t}=-\eta_3e_3(t)$$

i.e.,

$$-\left(\mathcal{B}(t)\mathbf{x}^{m-1}(t)\right)^{\top}\dot{\mathbf{x}}(t)=\frac{\eta_3}{m}(\mathcal{B}(t)\mathbf{x}^{m}(t)-1)+\dot{\mathcal{B}}(t)\mathbf{x}^{m}(t)$$

where $\eta_3\gg0$ is the design parameter and could be chosen similarly as η_1.

$$M_{\mathcal{B}}(t,\mathbf{z})\dot{\mathbf{z}}(t) = \mathbf{q}(t,\mathbf{z}),$$

where

$$M_{\mathcal{B}}(t,\mathbf{z}) = \begin{bmatrix} (m-1)(\mathcal{A}(t)-\lambda(t)\mathcal{B}(t))\mathbf{x}^{m-2}(t)) & -\mathcal{B}(t)\mathbf{x}^{m-1}(t) \\ -(\mathcal{B}(t)\mathbf{x}^{m-1}(t))^{\top} & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)},$$

and

$$\mathbf{q}(t,\mathbf{z}) = \begin{bmatrix} -\eta_1(\mathcal{A}(t)\mathbf{x}^{m-1}(t)-\lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t)) - \dot{\mathcal{A}}(t)\mathbf{x}^{m-1}(t)+\lambda(t)\dot{\mathcal{B}}(t)\mathbf{x}^{m-1}(t) \\ \frac{\eta_3}{m}(\mathcal{B}(t)\mathbf{x}^m(t)-1)+\dot{\mathcal{B}}(t)\mathbf{x}^m(t) \end{bmatrix} \in \mathbb{R}^{n+1}.$$

Giving a different condition, we can see that the coefficient matrix $M(t,\mathbf{z})$could be symmetric. By the Householder transformation [21], we can see that there exists a nonsingular U such that

$$U^{\top}(-\mathcal{B}(t)\mathbf{x}^{m-1}) = (1,0,\ldots,0)^{\top}.$$

Here we can let $-\mathbf{x}$ be the first column of U and $U(:2,n)^{\top}U(:2,n) = I_{n-1}$ and $U(:2,n)^{\top}(\mathcal{B}(t)\mathbf{x}^{m-1}) = \mathbf{0}$. Thus we have

$$U^{\top}(-\mathcal{B}\mathbf{x}^{m-2})U = \begin{bmatrix} 1 & \mathbf{0}^{\top} \\ \mathbf{0} & \widetilde{B} \end{bmatrix}$$

since $\mathcal{B}\mathbf{x}^m = 1$. On the other hand, we assume that $\lambda(t)$ is the simple eigenvalue flow, then we have

$$U^{\top}(\mathcal{A}\mathbf{x}^{m-2})U = \begin{bmatrix} \lambda & \mathbf{0}^{\top} \\ \mathbf{0} & \widetilde{A} \end{bmatrix}.$$

Denote $U_M = \begin{bmatrix} U & \mathbf{0} \\ \mathbf{0}^{\top} & 1 \end{bmatrix}$, then we can see that

$$U_M^{\top}M_{\mathcal{B}}U_M = \begin{bmatrix} & & 1 \\ & (m-1)\left(\widetilde{A}-\lambda\widetilde{B}\right) & \\ 1 & & \end{bmatrix}.$$

This means that the matrix $\widetilde{A}-\lambda\widetilde{B}$ is nonsingular which implies that $M_{\mathcal{B}}$is nonsingular. Which we only need to verify the invertibility of $\widetilde{A}-\lambda\widetilde{B}$. Without loss of generality, we assume that the matrix $M_{\mathcal{B}}$is nonsingular.

Remark 3.2. The invertibility of coefficient matrix $M(t,\mathbf{z})$ in (3.8) is satisfied for the matrix case. For generality, the authors suppose that it is nonsingular in [67]. Similarly, we also assume that it is nonsingular in the tensor case. Because of the complicity in the nonsymmetric matrix case [50], we only consider the symmetric tensors. We conjecture that the symmetric condition can be weakened, but this will not be covered in this paper.

The differential-algebraic system of equations (DAEs) with the general form of differential equations for vector-valued functions $\mathbf{x}$ in one independent variable t,

$$F(\dot{\mathbf{x}}(t),\mathbf{x}(t),t) = 0, \tag{3.12}$$

has been explored since the age of Cauchy. Theoretical results and methods for linear DAEs, quasilinear DAEs and nonlinear DAEs are abundant; see [25,31,33,43] for more details.

We continue our investigation of the Zhang dynamics model by the non-autonomous system (3.8). This non-autonomous system can be made autonomous upon replacing it by the system

$$\begin{cases} \dot{t} = 1, \\ M(t,\mathbf{z})\dot{\mathbf{z}}(t) = \mathbf{q}(t,\mathbf{z}), \end{cases}$$

and choosing $(t,\mathbf{z}) := \mathbf{u}$ as a new variable. Thus the system (3.8) may be rewritten as

$$M(\mathbf{u})\dot{\mathbf{u}} = \mathbf{q}(\mathbf{u}), \tag{3.13}$$

where $M(\mathbf{u}) = \begin{bmatrix} 1 & 0 \\ 0 & M(t,\mathbf{z}) \end{bmatrix}$ and $\mathbf{q}(\mathbf{u}) := \begin{bmatrix} 1 \\ \mathbf{q}(t,\mathbf{z}) \end{bmatrix}$. If $M(t,\mathbf{z})$ is nonsingular, then $M(\mathbf{u})$is nonsingular, and thus

$$\dot{\mathbf{u}} = M^{-1}(\mathbf{u})\mathbf{q}(\mathbf{u}) \tag{3.14}$$

which is not implicit anymore. By giving the initial values, we only need to consider the following classical explicit system of ordinary differential equations:

$$\begin{cases} \mathbf{u}(t) = f(t,\mathbf{u}(t)), \\ \mathbf{u}(0) = \mathbf{u}_0. \end{cases}$$

With only mild condition that f is continuous, then the solution of the above equation exists. As for the uniqueness, the Lipschitz continuous of f about $\mathbf{u}$ is needed. Moreover, the solution is analytic if f is also analytic; referred to [25,31,33,43] or [12, Theorems 1.183, 1.184 and 1.2].

3.3. Convergence analysis

We are ready for the convergence result.

Theorem 3.1. *Let $\mathcal{A}(t)$ and $\mathcal{B}(t)$ be two m-order n-dimension real symmetric time-varying tensor flows and $\mathcal{B}(t)$ is nonsingular for each t, and the initial values $\lambda(t_0) \in \mathbb{R}$ and $\mathbf{x}(t_0) \in \mathbb{R}^n$ are given. If the matrix $M(t,\mathbf{z})$ in continuous ZD model (3.8) is nonsingular for all t, then the ZD model (3.8) converges to a $\mathcal{B}_t$-eigenvalue $\lambda^*(t)$ and the corresponding $\mathcal{B}_t$-eigenvector $\mathbf{x}^*(t)$ of $\mathcal{A}(t)$.*

Proof. Recall that we have let $\dot{\mathbf{e}}_1(t) = -\eta_1\mathbf{e}_1(t)$. In order to prove the convergence, we firstly define a Lyapunov-candidate-function V as follows,

$$V(\mathbf{e}_1) = \frac{\mathbf{e}_1^{\top}(t)\mathbf{e}_1(t)}{2},$$

where $\mathbf{e}_1(t)$ is the vector-valued error function defined in the previous section.

Note that $V(\mathbf{e}_1) = 0$ if $\mathbf{e}_1(t) = 0$ and $V(\mathbf{e}_1)$ is positive for any other values of $\mathbf{e}_1(t)$, thus the Lyapunov-candidate-function V is positive definite (in the sense of dynamical systems). We use the notation $\dot{V}(\mathbf{e}_1)$ to denote the time derivative of the Lyapunov-candidate-function V, we can see that

$$\dot{V}(\mathbf{e}_1) = \frac{d}{dt}V(\mathbf{e}_1(t)) = \frac{\partial V}{\partial \mathbf{e}_1}\cdot\frac{d\mathbf{e}_1}{dt} = \mathbf{e}_1^{\top}(t)\cdot\dot{\mathbf{e}}_1(t) = -\eta_1\mathbf{e}_1^{\top}(t)\mathbf{e}_1(t).$$

Similarly, it is not hard to find that $\dot{V}(\mathbf{e}_1) = 0$ if $\mathbf{e}_1(t) = 0$ and $\dot{V}(\mathbf{e}_1) < 0$ if $\mathbf{e}_1(t) \neq 0$ since η_1 is positive, which means that $\dot{V}(\mathbf{e}_1)$ is negative definite. Thus $\mathbf{e}_1(t) = \mathcal{A}(t)\mathbf{x}^{m-1}(t) - \lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t)$ converges to zero as time t evolves by the Lyapunov stability theory. Similarly, for $\dot{e}_2(t) = -\eta_2 e_2(t)$, we know that $e_2(t) = \|\mathbf{x}(t)\|_2^2 - 1$ converges to zero also as time t evolves. Moreover, the eigenvector is of length one. □

Let us consider the convergence rate of the proposed model.

Theorem 3.2. *Suppose that the conditions in Theorem 3.1 hold, the tuple $(\lambda(t), \mathbf{x}(t))$ generated by the continuous ZD model (3.8) converges to an eigenpair of $\mathcal{A}(t)$ in exponential rate, if t is sufficient large.*

Proof. The continuous-time Zhang neural network (3.8) is derived from the dynamic equation $\dot{\mathbf{e}} = -\eta\mathbf{e}(t)$ with the initial condition $\mathbf{e}(t_0)$, which has the following explicit solution,

$$\mathbf{e}(t) = \mathbf{e}(t_0)\exp(-\eta t).$$

Thus $\mathbf{e}(t)$ converges exponentially to zero with convergent rate η. Therefore, for the two functions $\dot{\mathbf{e}}_1(t) = -\eta_1\mathbf{e}_1(t)$ and $\dot{e}_2(t) = -\eta_2 e_2(t)$, we can see that $\mathbf{e}_i(t)$ converges exponentially to zero with convergent rate η_i for $i = 1, 2$. By definition of the error function $e_2(t)$ in (3.5), we obtain that $\mathbf{x}(t)$ converges to a vector with length one which means that it is nonzero. On the other hand, convergence of the error function $\mathbf{e}_1(t)$ in (3.1) to zero implies that

$$\mathcal{A}(t)\mathbf{x}^{m-1}(t) = \lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t).$$

By Definition 2.1, we get the result. □

The following corollary could be obtained from the above two theorems. It is just the matrix case given by Zhang et al. [67].

Corollary 3.1. *Let $\mathcal{A}(t)$ be a two-order n-dimensional real symmetric time-varying tensor flow and $\mathcal{B}(t)$ is the identity matrix $\mathcal{I}$. The initial values $\lambda(t_0) \in \mathbb{R}$ and $\mathbf{x}(t_0) \in \mathbb{R}^n$ are given. If the matrix $M(t, \mathbf{z})$ in continuous ZD model (3.8) is nonsingular for all t, then the continuous ZD model (3.8) converges to an eigenvalue $\lambda^*(t)$ of $\mathcal{A}(t)$ exponentially with the corresponding eigenvector $\mathbf{x}^*(t)$.*

By Lemma 3.1, we can give one more result for the matrix case. Although it is a special case of [67], it is more concrete.

Theorem 3.3. Let $A(t)$ be an n-by-n time-varying real symmetric matrix flows with simple eigenvalue flows and the initial values $\lambda(t_0) \in \mathbb{R}$ and $\mathbf{x}(t_0) \in \mathbb{R}^n$ are given. Then the continuous ZD model (3.8) converges to an eigenvalue $\lambda^*(t)$ of $\mathcal{A}(t)$ exponentially with the corresponding eigenvector $\mathbf{x}^*(t)$.

3.4. Robustness analysis

We give the robustness result in this subsection.

Theorem 3.4. *Suppose that the original symmetric time-varying tensor flows $\mathcal{A}(t)$ in continuous ZD model (3.8) is perturbed by symmetric time-varying noise $\Delta\mathcal{A}(t)$ of small intensity, i.e., the input symmetric time-varying tensor is actually $\hat{\mathcal{A}}(t) = \mathcal{A}(t) + \Delta\mathcal{A}(t)$, and $\mathcal{B}(t)$ is known. Starting from the given initial states $\hat{\lambda}(t_0)$ and $\hat{\mathbf{x}}(t_0)$, then the steady-state residual error*

$$\lim_{t\to\infty}\|\mathcal{A}(t)\hat{\mathbf{x}}^{m-1}(t) - \hat{\lambda}(t)\mathcal{B}(t)\hat{\mathbf{x}}^{m-1}(t)\|_2$$

of (3.8) with the noise $\Delta\mathcal{A}(t)$ is uniformly upper bounded by $\sup_t\{\|\Delta\mathcal{A}(t)\|_F\}$. Furthermore, $\|\mathcal{A}(t)\hat{\mathbf{x}}^{m-1}(t) - \hat{\lambda}(t)\mathcal{B}(t)\hat{\mathbf{x}}^{m-1}(t)\|_2$ will converge exponentially to or stay within the error bound.

Proof. We define

$$E(t) = \mathcal{A}(t)\hat{\mathbf{x}}^{m-1}(t) - \hat{\lambda}(t)\mathcal{B}(t)\hat{\mathbf{x}}^{m-1}(t),$$

then we have

$$E(t) = \hat{\mathcal{A}}(t)\hat{\mathbf{x}}^{m-1}(t) - \hat{\lambda}(t)\mathcal{B}(t)\hat{\mathbf{x}}^{m-1}(t) - \Delta\mathcal{A}(t)\hat{\mathbf{x}}^{m-1}(t), \tag{3.15}$$

since $\mathcal{A}(t) = \hat{\mathcal{A}}(t) - \Delta\mathcal{A}(t)$. Note that

$$\hat{\mathcal{A}}(t)\hat{\mathbf{x}}^{m-1}(t) - \hat{\lambda}(t)\mathcal{B}(t)\hat{\mathbf{x}}^{m-1}(t) := \hat{\mathbf{e}}_1(t) = \hat{\mathbf{e}}_1(t_0)\cdot\exp(-\eta_1 t)$$

where η_1 is positive and it approaches zero as t approaches infinity. Therefore, by Theorem 3.1, the steady-state of the above error function is

$$\lim_{t\to\infty}E(t) = \lim_{t\to\infty}\left(-\Delta\mathcal{A}(t)\hat{\mathbf{x}}^{m-1}(t)\right).$$

By Lemma 2.2, we have

$$\begin{aligned}\lim_{t\to\infty}\|E(t)\|_2 &= \lim_{t\to\infty}\| - \Delta\mathcal{A}(t)\hat{\mathbf{x}}^{m-1}(t)\|_2 \leqslant \lim_{t\to\infty}\|\Delta\mathcal{A}(t)\|_F\|\hat{\mathbf{x}}(t)\|_2^{m-1}\\ &\leqslant \sup_t\{\|\Delta\mathcal{A}(t)\|_F\}\end{aligned}$$

since $\hat{\mathbf{x}}^\top(t)\hat{\mathbf{x}}(t) - 1 := \hat{e}_2(t) = \hat{e}_2(t_0)\exp(-\eta_2 t)$ approaches zero as t approaches infinity.

By the triangle inequality of the vector norm, from (3.15), we have

$$\begin{aligned}\|E(t)\|_2 &\leqslant \|\hat{\mathcal{A}}(t)\hat{\mathbf{x}}^{m-1}(t) - \hat{\lambda}(t)\mathcal{B}(t)\hat{\mathbf{x}}^{m-1}(t)\|_2 + \|\Delta\mathcal{A}(t)\hat{\mathbf{x}}^{m-1}(t)\|_2\\ &\leqslant \|\hat{\mathbf{e}}_1(t_0)\cdot\exp(-\eta_1 t)\|_2 + \sup_t\{\|\Delta\mathcal{A}(t)\|_F\}\|\hat{\mathbf{x}}(t)\|_2^{m-1}\\ &\leqslant \|\hat{\mathbf{e}}_1(t_0)\cdot\exp(-\eta_1 t)\|_2 + \sup_t\{\|\Delta\mathcal{A}(t)\|_F\}\left(\sqrt{|\hat{e}_2(t_0)\exp(-\eta_2 t) + 1|}\right)^{m-1}\\ &\leqslant \|\hat{\mathbf{e}}_1(t_0)\cdot\exp(-\eta_1 t)\|_2 + \sup_t\{\|\Delta\mathcal{A}(t)\|_F\}\left(1 + \sqrt{|\hat{e}_2(t_0)\exp(-\eta_2 t)|}\right)^{m-1},\end{aligned}$$

which means that

$$\|E(t)\|_2 - \sup_t\{\|\Delta\mathcal{A}(t)\|_F\} \leqslant \|\hat{\mathbf{e}}_1(t_0)\cdot\exp(-\eta_1 t)\|_2 + \sup_t\{\|\Delta\mathcal{A}(t)\|_F\}\cdot f(|\hat{e}_2(t_0)\exp(-\eta_2 t)|).$$

Thus the function $f(|\hat{e}_2(t_0)\exp(-\eta_2 t)|) = (1 + \sqrt{|\hat{e}_2(t_0)\exp(-\eta_2 t)|})^{m-1} - 1$ and can be computed by using the binomial formula. Each term of $f(|\hat{e}_2(t_0)\exp(-\eta_2 t)|)$ consists of the factor $\exp(-\eta_2 t)$ and approaches zero as t approaches infinity. Therefore, $\|\mathcal{A}(t)\hat{\mathbf{x}}^{m-1}(t) - \hat{\lambda}(t)\mathcal{B}(t)\hat{\mathbf{x}}^{m-1}(t)\|_2$ converges exponentially to or stay within the error bound $\sup_t\{\|\triangle\mathcal{A}(t)\|_F\}$. □

3.5. Modified continuous ZD model

Activation functions are always used in the recurrent neural networks for solving the time-varying matrix problems, such as matrix inversion and Sylvester equations [68]. Using activation functions to accelerate the convergence, we propose a modified continuous ZD model for the time-varying generalized tensor eigenpair (1.2).

In (3.2) and (3.6) of Section 3.1, we stipulate that

$$\dot{\mathbf{e}}_1(t) = \frac{\mathrm{d}\mathbf{e}_1(t)}{\mathrm{d}t} = -\eta_1\mathbf{e}_1(t), \quad \dot{e}_2(t) = \frac{\mathrm{d}e_2(t)}{\mathrm{d}t} = -\eta_2 e_2(t). \tag{3.16}$$

In order to accelerate the convergent process, we modify the above Eqs. (3.16) to

$$\begin{aligned}\dot{\mathbf{e}}_1(t) &= \frac{\mathrm{d}\mathbf{e}_1(t)}{\mathrm{d}t} = -\eta_1\exp(\beta_1 t)\Phi(\mathbf{e}_1(t)),\\ \dot{e}_2(t) &= \frac{\mathrm{d}e_2(t)}{\mathrm{d}t} = -\eta_2\exp(\beta_2 t)\Phi(e_2(t))\end{aligned} \tag{3.17}$$

in which $\beta_1, \beta_2 \geqslant 0$. Here $\Phi(\cdot)$ is a monotonically-increasing odd activation-function and $\Phi(\mathbf{y}) = (\Phi(y_1), \Phi(y_2), \cdots, \Phi(y_n))^\top$.

Remark 3.3. Generally speaking, any monotonically-increasing odd activation-function $\Phi(\cdot)$ could be used for model construction. Some known activation-functions, such as linear activation function, bipolar-sigmoid activation function, power activation function and power-sigmoid activation function, have been used by researchers [68].

Three activation functions are used in the numerical examples:

(1) Linear activation function: $\Phi(u) = u$.
(2) Power activation function: $\Phi(u) = u^p$ where integer $p \geqslant 3$. We set $p = 5$ below.
(3) Bipolar-sigmoid activation function: $\Phi(u) = \frac{1-\exp(-\xi u)}{1+\exp(-\xi u)}$ with $\xi \geqslant 1$. We set $\xi = 1$ below.

We only change the right-hand side of the first-order time derivatives, then the coefficient matrix of the modified continuous ZD model remains the same as in (3.8). However, the right-hand side changes and we denote it by $\mathbf{q}_m(t,\mathbf{z})$ in which

$$\mathbf{q}_m(t,\mathbf{z}) = \begin{bmatrix} -\eta_1 \exp(\beta_1 t)\Phi(\mathbf{e}_1(t)) - \dot{\mathcal{A}}(t)\mathbf{x}^{m-1}(t) + \lambda(t)\dot{\mathcal{B}}(t)\mathbf{x}^{m-1}(t) \\ -\eta_2 \exp(\beta_2 t)\Phi(e_2(t)) \end{bmatrix} \in \mathbb{R}^{n+1}.$$

Thus, we get the general formula of the modified continuous ZD model,

$$M(t,\mathbf{z})\dot{\mathbf{z}}(t) = \mathbf{q}_m(t,\mathbf{z}). \tag{3.18}$$

It is obvious that by choosing $\beta_1 = \beta_2 = 0$ and $\Phi(\cdot)$ as the linear activation function, the previous model (3.8) could be retrieved from (3.18). In this case, the previous model (3.8) obtains convergence in exponential rate if certain conditions are satisfied.

Now we suppose that $\Phi(\cdot)$ is the linear activation function and both β_1 and β_2 are positive, then we deduce a special case of (3.18):

$$M(t,\mathbf{z})\dot{\mathbf{z}}(t) = \mathbf{q}_{m_\ell}(t,\mathbf{z}) \tag{3.19}$$

where

$$\mathbf{q}_{m_\ell}(t,\mathbf{z}) = \begin{bmatrix} -\eta_1 \exp(\beta_1 t)\big(\mathcal{A}(t)\mathbf{x}^{m-1}(t) - \lambda(t)\mathcal{B}(t)\mathbf{x}^{m-1}(t)\big) - \dot{\mathcal{A}}(t)\mathbf{x}^{m-1}(t) + \lambda(t)\dot{\mathcal{B}}(t)\mathbf{x}^{m-1}(t) \\ -\eta_2 \exp(\beta_2 t)(\mathbf{x}^\top(t)\mathbf{x}(t) - 1) \end{bmatrix}.$$

For this case, we have the following result.

Theorem 3.5. *Let $\mathcal{A}(t)$ and $\mathcal{B}(t)$ be two m-order n-dimensional real symmetric time-varying tensor flows and $\mathcal{B}(t)$ is nonsingular for each t, and the initial values $\lambda(t_0) \in \mathbb{R}$ and $\mathbf{x}(t_0) \in \mathbb{R}^n$ are given. If the matrix $M(t,\mathbf{z})$ in model (3.19) is nonsingular for all t, then the modified continuous ZD model (3.19) converges to a $\mathcal{B}_t$-eigenvalue $\lambda^*(t)$ of $\mathcal{A}(t)$ with the corresponding $\mathcal{B}_t$-eigenvector $\mathbf{x}^*(t)$ in super-exponential rate.*

Proof. To prove the global convergence, similarly as Theorem 3.1, we define a Lyapunov-candidate-function V,

$$V(\mathbf{e}_1) = \frac{\mathbf{e}_1^\top(t)\mathbf{e}_1(t)}{2},$$

where $\mathbf{e}_1(t)$ is the vector-valued error function defined in (3.17). Consequently, we have

$$\dot{V}(\mathbf{e}_1) = \mathbf{e}_1^\top(t)\cdot\dot{\mathbf{e}}_1(t) = -\eta_1 \exp(\beta_1 t)\mathbf{e}_1^\top(t)\mathbf{e}_1(t).$$

We can see that $V(\mathbf{e}_1)$ is positive definite and $\dot{V}(\mathbf{e}_1)$ is negative definite since $\eta_1 > 0$ and $\exp(\beta_1 t)$ is always positive for any t. Thus $\mathbf{e}_1(t)$ converges globally to zero as time t evolves by using the Lyapunov stability theory. The convergent result for $e_2(t)$ could be obtained similarly.

Note that $\dot{\mathbf{e}}_1(t) = -\eta_1 \exp(\beta_1 t)\mathbf{e}_1(t)$ and $\dot{e}_2(t) = -\eta_2 \exp(\beta_2 t)e_2(t)$. We suppose that the initial conditions are $\mathbf{e}_1(t_0)$ and $e_2(t_0)$ and by the classical technique of ODE, we have

$$\mathbf{e}_1(t) = \mathbf{e}_1(t_0)\exp\left(-\frac{\eta_1}{\beta_1}(\exp(\beta_1 t) - 1)\right),$$
$$e_2(t) = e_2(t_0)\exp\left(-\frac{\eta_2}{\beta_2}(\exp(\beta_2 t) - 1)\right).$$

In this case, we can see that the model will converge faster than the previous one in (3.8) and we get the super-exponential rate. $\square$

Remark 3.4. Three corollaries for the matrix eigenvalues, the tensor H- and Z-eigenvalues, could be concluded as in Section 3.3, and we omit them here.

Remark 3.5. In the above theorem, the linear activation function is used. For the nonlinear activation functions, some numerical results are shown in the next section.

4. Numerical examples

We present some illustrative examples to illustrate the effectiveness of the models proposed in previous sections for the time-varying tensor eigenpair.

We discuss the solution of (3.8) in sub Section 3.1. As for the methods, a vast number of numerical discretizations schemes exist for DAEs, and most of them are originally designed for ODEs, such as backward differentiation formulas methods or Runge–Kutta methods or Rosenbrock-Wanner methods [26]. All MATLAB ODE solvers can solve systems of equations of the form like (3.14) or the one like (3.13) with a mass $M(\mathbf{u})$, i.e., $M(t,\mathbf{x})$. Even though the mass matrix is singular, we can use “ODE15s” and “ODE23t” to solve it. Usually, the “ODE15s” solver is the a first choice for most stiff problems. In view of this, in the simulation of the corresponding continuous ZD model and modified continuous ZD model, the MATLAB solver “ODE15s” is used.

We briefly introduce some procedures below. Before the algorithm is implemented, the starting time t_0 and initial values $\lambda(t_0)$ and $\mathbf{x}(t_0)$ should be given. Since we know the explicit expression of $\mathcal{A}(t)$, it is easy to generate the mass matrix of (3.8). Here, the time derivative is required, and we compute it by using the MATLAB routine “diff”. By using the MATLAB routine “odeset”, we can get the value of the mass matrix, i.e.,

$$\mathrm{options} = \mathrm{odeset}('\mathrm{Mass}', @\mathrm{mM})$$

where “mM” is the mass matrix. Similarly, we can obtain the right-hand side of (3.8) and (3.18). The main step is

$$[\mathrm{t}, \mathrm{x}] = \mathrm{ode15s}(@\mathrm{RHS}, \mathrm{Tspan}, \mathrm{x}, \mathrm{options})$$

where RHS denotes the right-hand side of the model (3.8) and Tspan is the time interval. Note that the mass matrix of the modified model (3.18) is the same as (3.8), and the only difference is the right-hand side. We just need to change the right-hand function in implementing the algorithm. For detailed step-by-step MATLAB code, one can refer to [68] in which the authors show it explicitly.

The proposed models only track one of the eigenpairs, therefore we need to test with different initial values. Also, there are some parameters involved in the model (3.8), one can choose the appropriate values for experimental purposes.

Firstly, an example in which the associated tensor is of order two from [67], i.e., the matrix case, is given. Correspondingly, we choose $\mathcal{B}(t)$ as the identity matrix.

Example 4.1. (Example 4.1 of [67]) The matrix flow $A(t)$ is given by

$$A(t) = \begin{bmatrix} \sin(t) & \exp(\sin(t)) \\ \exp(\sin(t)) & \cos(t) \end{bmatrix}.$$

Moreover, its analytic eigenvalues are

$$\begin{cases} \lambda_1^*(t) = \frac{\sin(t)+\cos(t)-\sqrt{(\sin(t)-\cos(t))^2+4\exp(2\sin(t))}}{2}, \\ \lambda_2^*(t) = \frac{\sin(t)+\cos(t)+\sqrt{(\sin(t)-\cos(t))^2+4\exp(2\sin(t))}}{2}. \end{cases}$$

By using the continuous ZD model (3.8), the eigenvalues can be obtained successfully and the trajectories results are shown in Fig. 1 of [67], with the appropriate initial values and parameters. Moreover, the solution error and residual error are also given; see Fig. 2 of [67]. Next, we apply the modified continuous ZD model (3.18) to study the same problem. We choose two different initial values, and we set the parameters as $\eta_1 = \eta_2 = 10$ and $\beta_1 = \beta_2 = \frac{1}{100}$. We can see that the modified continuous ZD model tracks the accurate eigenvalues successfully, as illustrated in Fig. 2.

In order to see the convergence rate of the residual error $\text{err}(t) = \|A(t)\mathbf{x}(t) - \lambda(t)\mathbf{x}(t)\|_2$ more explicit, we choose the time interval $[0,3]$ and the parameters are $\eta_1 = \eta_2 = 10$ and $\beta_1 = \beta_2 = 10.16$. The initial values are $\mathbf{x}(0) = (1,0)^\top$ and $\lambda(0) = 4, \mathbf{x}(0) = (1,0)^\top$ and $\lambda(0) = 0$. The numerical results are illustrated in Fig. 3, from which we can see that the convergence of the modified model is faster than the original model proposed by Zhang et al. [67].

The linear activation function is used in the above example. Now we apply a nonlinear activation function to show the effectiveness of the modified model. The bipolar-sigmoid activation function is chosen as an example. Initial values and parameters remain the same. The numerical results are illustrated in Fig. 4, from which we can see that the convergence of the modified model is also faster than the original model proposed in [67]. Eigenvectors obtained from the modified continuous ZD model (3.18) are also shown in Table 1.

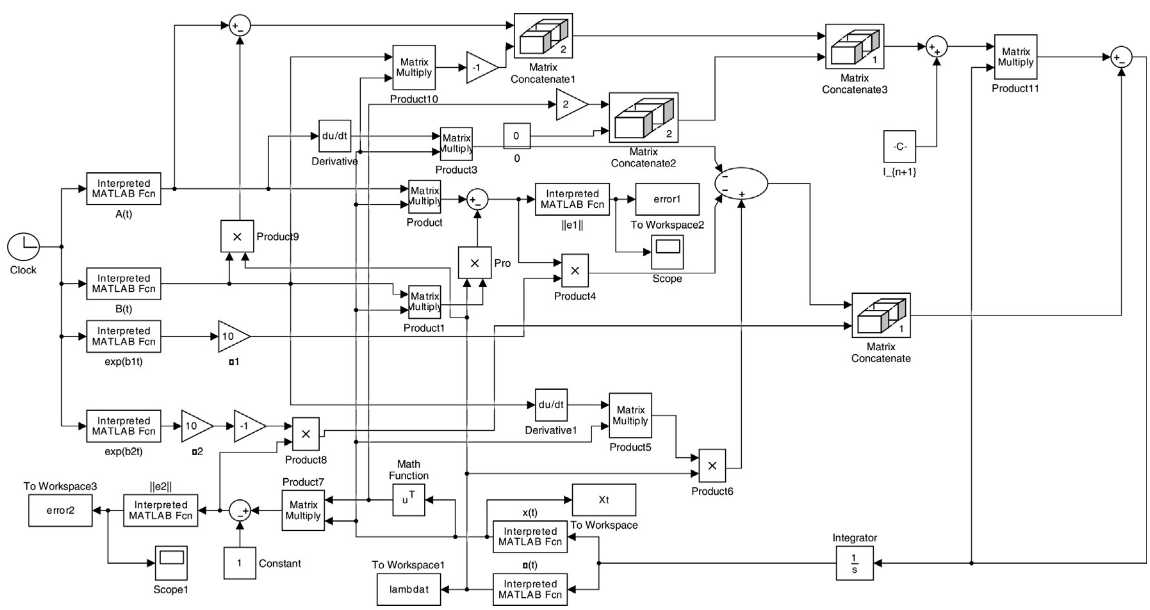
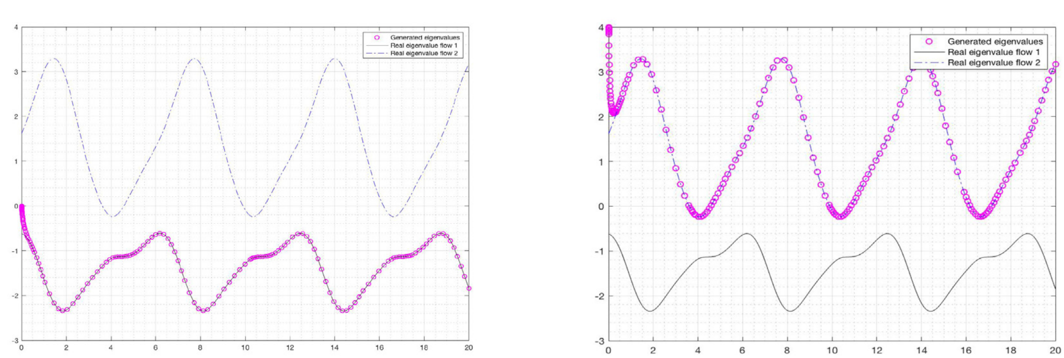
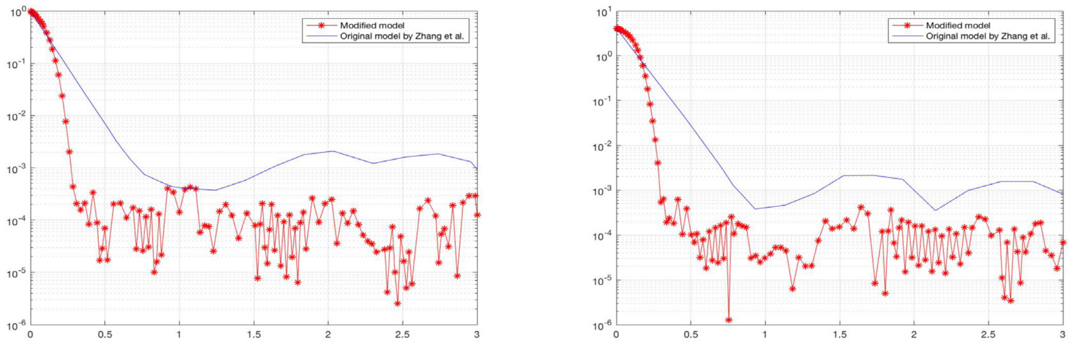


Fig. 1. Simulink model of (3.18) with linear activation function.



(a) $\mathbf{x}(0) = (1,0)^\top$, $\lambda(0) = 0$. (b) $\mathbf{x}(0) = (1,0)^\top$, $\lambda(0) = 4$.

Fig. 2. Trajectories of the two different eigenvalues of matrix $\mathcal{A}(t)$ by modified model (3.18).



(a) $\mathbf{x}(0) = (1, 0)^\top$, $\lambda(0) = 0$.

(b) $\mathbf{x}(0) = (1, 0)^\top$, $\lambda(0) = 4$.

Fig. 3. Residual error synthesized by model (3.8) and modified model (3.18) with linear activation function.

Example 4.2. Consider a cubic symmetric tensor $\mathcal{A}(t) \in \mathbb{R}^{[3,3]}$ which is depending on t with all zero entries except that

$$\begin{cases} \mathcal{A}(1,1,1) = (t-20)^2 + 2, \\ \mathcal{A}(1,2,2) = \mathcal{A}(2,1,2) = \mathcal{A}(2,2,1) = \cos(t-20), \\ \mathcal{A}(1,3,3) = \mathcal{A}(3,1,3) = \mathcal{A}(3,3,1) = -t + 21, \end{cases}$$

and $\mathcal{B}(t)$ is chosen as the tensor $\mathcal{E}$. Model (3.8) is used to compute the Z-eigenvalues and also eigenvectors. For different initial eigenpairs, the result is illustrated in Fig. 5(a) and (b). When $t = 20, \mathcal{A}(t)$ has two different Z-eigenvalues, $\lambda_1 = 2 \text{and} \lambda_2 = -2$ with eigenvectors $(1,0,0)^\top$ and $(-1,0,0)\top$ respectively; see Example 5.7 of [4] and Example 4.9 of [15]. By Fig. 5(a) and (b), we can see that if $t = 20$, we get two Z-eigenvalues, 2 and -2, thus the results obtained by the proposed model are reasonable. The figure of the residual error of Fig. 5(a) and (b) by the proposed model (3.8) and model (3.18) is given in Fig. 6. In these tests, the parameters β_1 and β_2 in the modified model (3.18) are chosen as 0.001.

Chen et al. [11] proposed a MATLAB software package, TenEig, to compute the tensor Z-eigenvalues, and we use it to verify the results in Fig. 5 where the horizontal axis denotes the time t and vertical axis denotes the Z-eigenvalue. By computing the Z-eigenvalues of the tensors $\mathcal{A}(t)$ in different time t, we find that the results are almost the same which shows that these models are effective in computing Z-eigenvalues.

Moreover, $\lambda_1^*(t) = (t-20)^2 + 2$ and $\lambda_2^*(t) = -(t-20)^2 - 2$ are always the accurate eigenvalue flows of tensor $\mathcal{A}(t)$, and their eigenvectors are $(1,0,0)^\top$ and $(-1,0,0)^\top$ respectively. By verifying the values of $(t-20)^2 + 2$ and $-(t-20)^2 - 2$ in different t, we can see that the results are reasonable.

Example 4.3. We have a symmetric real tensor $\mathcal{A}(t) \in \mathbb{R}^{[4,3]}$ depending on t with all entries being zeros except that

$$\begin{cases} a_{1111} = (t-20)^2 + 2, a_{2222} = 3 * \exp(t-20), a_{3333} = -\cos(t-20) + 6, \\ a_{1123} = a_{1213} = a_{1231} = a_{2113} = a_{2131} = a_{2311} = \sin(t-20) + 1, \\ a_{1132} = a_{1312} = a_{1321} = a_{3112} = a_{3121} = a_{3211} = -\cos(t-20) + 2. \end{cases}$$

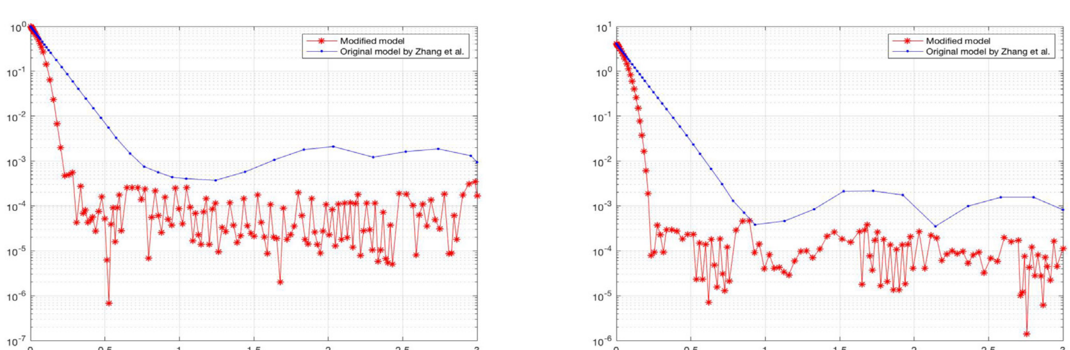
When $t = 20, \mathcal{A}x^4 = 2x_1^4 + 3x_2^4 + 5x_3^4 + 12x_1^2x_2x_3$ which was discussed in Example 3 of [39] and Example 4.3 of [15]. Moreover, from Example 4.3 of [15], it has five different H-eigenvalues

$$-1.3952, \quad 2.0000, \quad 3.0000, \quad 5.0000, \quad 7.4505.$$

By giving different initial values of the eigenvectors and eigenvalues, we can compute the above H-eigenvalues (and its corresponding eigenvectors). The initial values for computing the five H-eigenvalues are chosen as

$$\begin{cases} (\mathbf{x}_0^\top; \lambda_0) = \left(0.6, -0.5, -\sqrt{1 - 0.6^2 - 0.5^2}; -2\right), \\ (\mathbf{x}_0^\top; \lambda_0) = (1, 0, 0; 0), \\ (\mathbf{x}_0^\top; \lambda_0) = (0, 1, 0; 2), \\ (\mathbf{x}_0^\top; \lambda_0) = (0, 0, 1; 6), \\ (\mathbf{x}_0^\top; \lambda_0) = \left(0.6, 0.5, \sqrt{1 - 0.6^2 - 0.5^2}; 7\right). \end{cases}$$

We use model (3.8) to compute these H-eigenvalues and there trajectories are shown on the left of Fig. 7. With appropriate parameters $\beta_1 = \beta_2 = 0.001$, we also successfully track the trajectories by model (3.18), as shown on the right of Fig. 7. We show the residual error of two eigenvalue flows generated by the modified model (3.18), as shown in Fig. 8.



(a) $\mathbf{x}(0) = (1, 0)^\top$, $\lambda(0) = 0$.

(b) $\mathbf{x}(0) = (1, 0)^\top$, $\lambda(0) = 4$.

Fig. 4. Residual error synthesized by model (3.8) and modified model (3.18) with bipolar-sigmoid activation function.

Table 1
Partial selected eigenvectors of the time-varying matrix in different time t with initial values $\mathbf{x}(0) = (1,0)^\top$ and $\lambda(0) = 0$ by the modified model (3.18).

time t	Accurate eigenvalue	Accurate eigenvector	Approximate eigenvector
2.034	−2.313	$(0.606,-0.795)^\top$	$(0.607,-0.795)^\top$
5.062	−1.049	$(-0.963,0.270)^\top$	$(0.963,-0.270)^\top$
8.118	−2.344	$(0.622,-0.783)^\top$	$(0.622,-0.783)^\top$
11.009	−1.119	$(-0.951,0.309)^\top$	$(0.951,-0.309)^\top$
14.159	−2.276	$(0.638,-0.770)^\top$	$(0.638,-0.770)^\top$
17.090	−1.131	$(-0.929,0.369)^\top$	$(0.930,-0.368)^\top$
20	−1.844	$(0.671,-0.742)^\top$	$(0.671,-0.742)^\top$

Example 4.4. In this example, we consider the generalized eigenvalue case. For simplicity, order two tensors are chosen, i.e., the generalized matrix eigenvalue problem. Let

$$A(t) = \begin{pmatrix} \sin(t)+2 & \exp(\sin(t)) & 0 & -\exp(\sin(t)) \\ \exp(\sin(t)) & \cos(t)-2 & 0 & 1 \\ -\exp(\sin(t)) & 0 & -0.12t^2+2.4t-7 & 0 \\ -\exp(\sin(t)) & 1 & 0 & 1/(t+1) \end{pmatrix},$$

which is a submatrix of $A_s(t)$ [50, Page 425]. $B(t)$ is a diagonal matrix given, by MATLAB notation, as follows,

$$B(t) = \mathrm{diag}\left(\left[\cos(t)+2, \exp(\sin(t)), 2, t^2+1\right]\right).$$

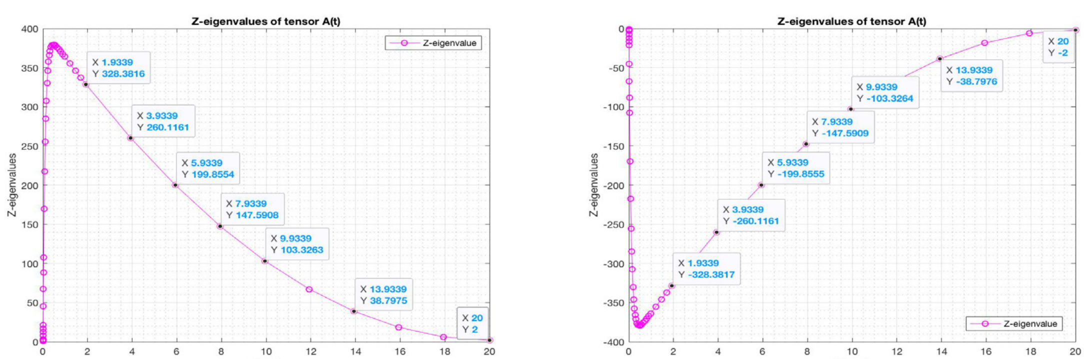
Time interval is chosen as $[0,20]$. Using the '*eig*' function in MATLAB, we find that the four generalized eigenvalues of matrix pencil $\{A(20), B(20)\}$ are $-3.49999, -1.09248, 0.00136, 1.66198$, which are used to verify the proposed models. The initial values for computing the four eigenvalues are chosen as

$$\begin{cases} (\mathbf{x}_0^\top;\lambda_0) = (0,0,1,0;1), \\ (\mathbf{x}_0^\top;\lambda_0) = \left(0.2,-0.5,0,\sqrt{1-0.2^2-(-0.5)^2};-1\right), \\ (\mathbf{x}_0^\top;\lambda_0) = \left(0.2,0.5,0,\sqrt{1-0.2^2-0.5^2};0\right), \\ (\mathbf{x}_0^\top;\lambda_0) = \left(-0.5,-0.2,0,\sqrt{1-0.5^2-0.2^2};2\right). \end{cases}$$

We use the models (3.8) and (3.18) to compute the eigenvalues flows. We choose the parameters $\eta_1 = \eta_2 = 10$ and $\beta_1 = \beta_2 = 1$. The trajectories of the four different eigenvalues of the matrix pencil $\{A(t), B(t)\}$ by the models (3.8) and (3.18) with the linear activation function are shown in Fig. 9. The residual errors of the two models are given in Fig. 10 from which we can see that the modified model is more reliable. Moreover, another activation function, the bipolar-sigmoid activation function (BSAF), is applied, and the results are shown in Fig. 11. The accuracy is similar as that with the linear activation function but better than that by the original model (3.8).

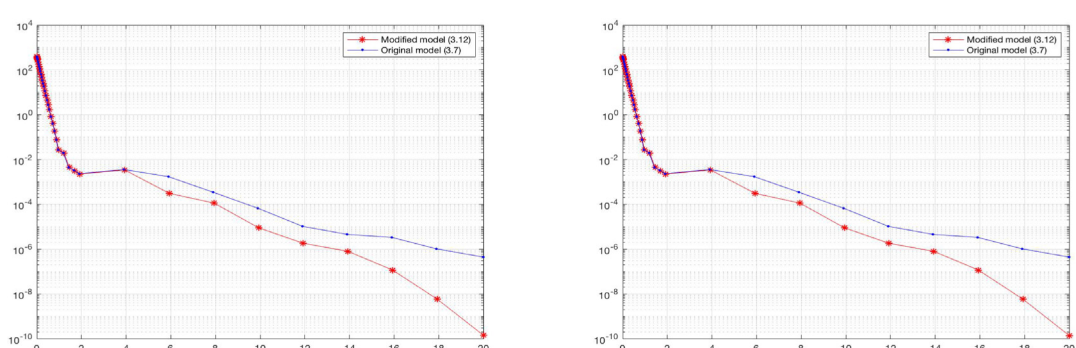
Remark 4.1. Not all nonlinear activation functions are better than the linear activation function. In the numerical examples, the power activation function is also used, but we find that sometimes it is not better than the linear one. Therefore, choosing an appropriate activation function is also an important issue.

Remark 4.2. The concrete Simulink models of this example correspond to three different activation functions. Designing the Simulink model, we write the equation $M(t,\mathbf{z})\dot{\mathbf{z}}(t) = \mathbf{q}(t,\mathbf{z})$ into $(I + M(t,\mathbf{z}))\dot{\mathbf{z}}(t) - \mathbf{q}(t,\mathbf{z}) = \dot{\mathbf{z}}(t)$where I is the identity matrix with



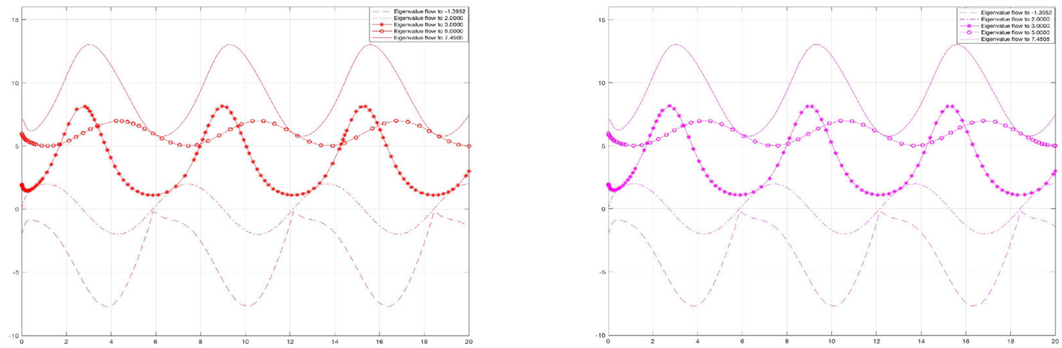
(a) $\mathbf{x} = (1,0,0)^\top$, $\lambda = 1$. (b) $\mathbf{x} = (-1,0,0)^\top$, $\lambda = -1$.

Fig. 5. Trajectories of the two different Z-eigenvalues of tensor $\mathcal{A}(t)$ with $\eta_1 = \eta_2 = 10$.



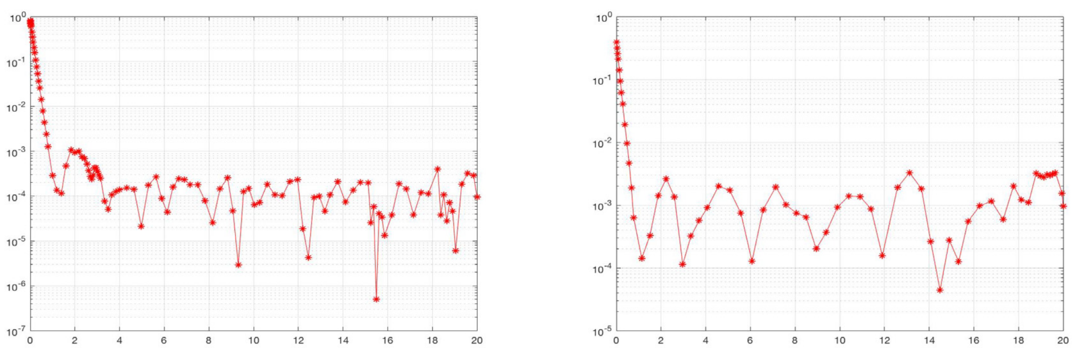
(a) $\mathbf{x} = (1,0,0)^\top$, $\lambda = 1$. (b) $\mathbf{x} = (-1,0,0)^\top$, $\lambda = -1$.

Fig. 6. Residual error synthesized by model (3.8) and modified model (3.18).



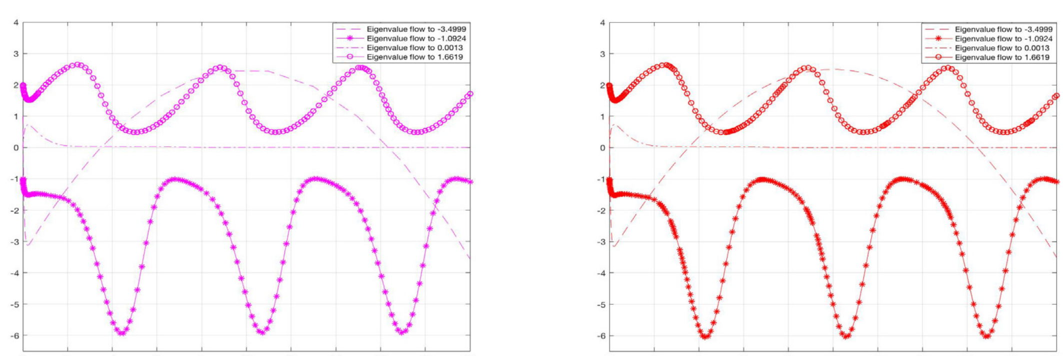
(a) Model (3.8) (b) Modified model (3.18)

Fig. 7. Trajectories of the five different H-eigenvalues of tensor $\mathcal{A}(t)$ by model (3.8) and modified model (3.18).



(a) $(\mathbf{x}_0^\top; \lambda_0) = (1,0,0;0)$ (b) $(\mathbf{x}_0^\top; \lambda_0) = (0,0,1;6)$

Fig. 8. Figure 8: Residual error synthesized by modified model (3.18) for Example 4.3.



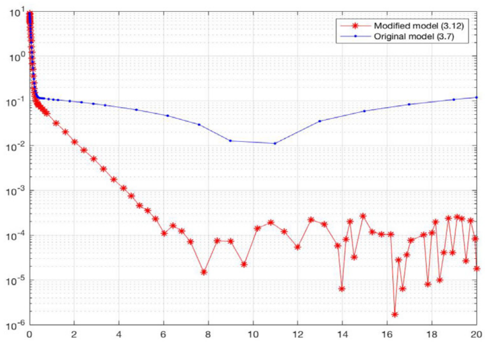
(a) Model (3.8) (b) Modified model (3.18)

Fig. 9. Trajectories of the four different generalized eigenvalues of matrix pencil $\{A(t), B(t)\}$ by model (3.8) and modified model (3.18).

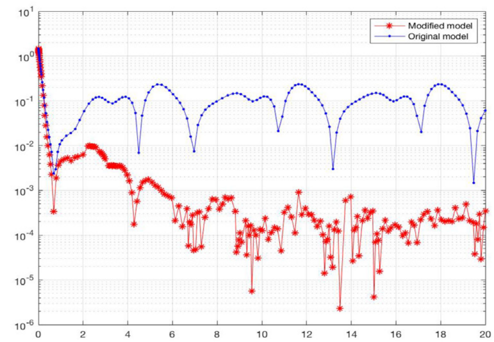
the same size as $M(t, \mathbf{z})$. Before running the model, we need to set $n = \text{length}(A(t))$ in the command window. As for the tensors case, by giving $\mathcal{A}(t)$ and $\mathcal{B}(t)$, we can get the coefficient matrix $M(t, \mathbf{z})$ and the right-hand vector $\mathbf{q}(t, \mathbf{z})$, and then we perform the model (3.18) to compute it. Different initial values are needed for different results.

5. Conclusions and remarks

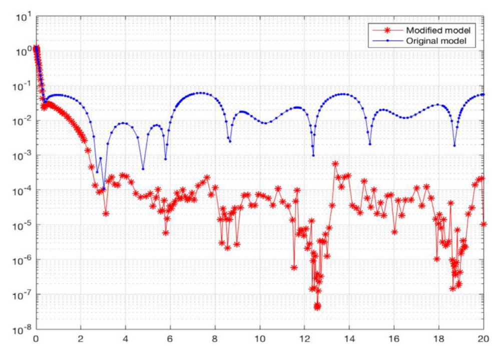
In this paper, two Zhang dynamics models, the continuous Zhang dynamics model and the modified continuous Zhang dynamics model, have been presented to explore the time-varying generalized tensor eigenpairs. The time-varying general-



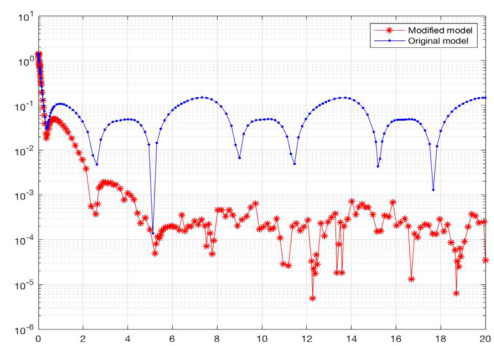
(a) $(\mathbf{x}_0^\top; \lambda_0) = (0, 0, 1, 0; 1)$



(b) $(\mathbf{x}_0^\top; \lambda_0) = (0.2, -0.5, 0, \sqrt{1 - 0.2^2 - (-0.5)^2}; -1)$

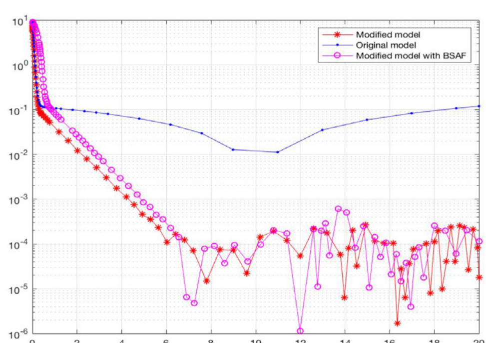


(c) $(\mathbf{x}_0^\top; \lambda_0) = (0.2, 0.5, 0, \sqrt{1 - 0.2^2 - 0.5^2}; 0)$

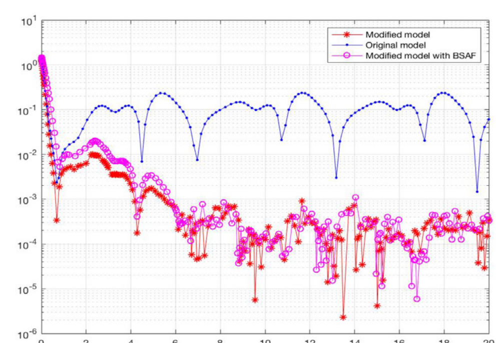


(d) $(\mathbf{x}_0^\top; \lambda_0) = (-0.5, -0.2, 0, \sqrt{1 - 0.5^2 - 0.2^2}; 2)$

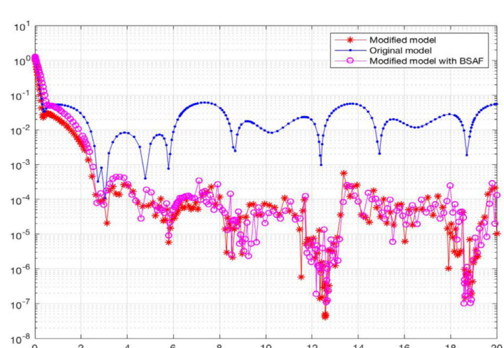
Fig. 10. Residual error synthesized by model (3.8) and modified model (3.18) for Example 4.4.



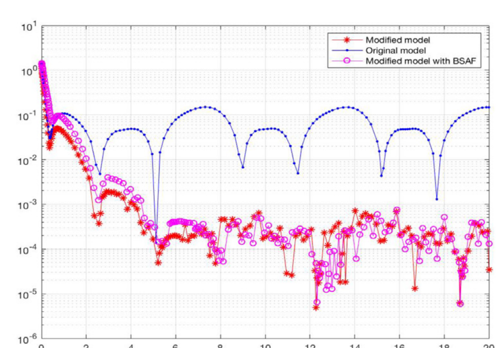
(a) $(\mathbf{x}_0^\top; \lambda_0) = (0, 0, 1, 0; 1)$



(b) $(\mathbf{x}_0^\top; \lambda_0) = (0.2, -0.5, 0, \sqrt{1 - 0.2^2 - (-0.5)^2}; -1)$



(c) $(\mathbf{x}_0^\top; \lambda_0) = (0.2, 0.5, 0, \sqrt{1 - 0.2^2 - 0.5^2}; 0)$



(d) $(\mathbf{x}_0^\top; \lambda_0) = (-0.5, -0.2, 0, \sqrt{1 - 0.5^2 - 0.2^2}; 2)$

Fig. 11. Residual error synthesized by model (3.8) and modified model (3.18) for Example 4.4.

ized tensor/matrix eigenpairs could also be computed by the proposed model. Theoretical analysis results of convergence and robustness are given. Finally, numerical examples illustrate the efficiency of the two proposed Zhang dynamics models. Moreover, the theoretical and numerical results substantiate that the modified continuous Zhang dynamics models could get better results .

It is worth pointing out that the time-varying tensor eigenpair is difficult, even for the general time-varying matrix case. More theoretical study is required to get a better understanding of the eigenanalysis for time-varying tensors. Uhlig and Zhang [50] posed many open problems for the matrix case, and the same problems for time-varying tensor of higher order would be a bigger challenge. Bordered matrices in the matrix case are singular only for specific borders, and the probability of it being singular is zero. In other words, the generic case is always nonsingular. How about the bordered matrices for the tensor case? It is an open question for the future research.

CRediT authorship contribution statement

Changxin Mo: Data curation, Software, Writing - original draft. **Xuezhong Wang:** Writing - review & editing, Validation. **Yimin Wei:** Conceptualization, Methodology, Investigation.

Declaration of Competing Interest

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.

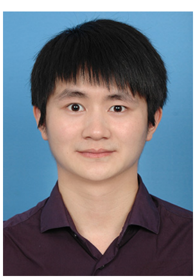
Acknowledgments

The authors would like to thank the handling editor and two referees for their detailed comments which greatly improve the manuscript. We would like to thank Prof. Frank Uhlig for his recent papers on time-varying matrix and Prof. Eric King-Wah Chu, Dr. Weiyang Ding, Prof. Jun Yan, Prof. Predrag S. Stanimirović and Prof. Dimitrios Gerontitis for their valuable suggestions and help. The first author would also like to thank Miss Chengxia Zu for her help and useful references.

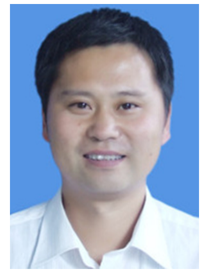
References

[1] J.W. Blattner, Bordered matrices, J. Soc. Ind. Appl. Math. 10 (1962) 528–536.
[2] A. Bunse-Gerstner, R. Byers, V. Mehrmann, N.K. Nichols, Numerical computation of an analytic singular value decomposition of a matrix valued function, Numer. Math. 60 (1991) 1–39.
[3] C. Carathéodory, Variationsrechnung und Partielle Differentialgleichungen erster Ordnung, B.G. Teubner Verlagsgesellschaft, Leipzig, 1935.
[4] D. Cartwright, B. Sturmfels, The number of eigenvalues of a tensor, Linear Algebra Appl. 438 (2013) 942–952.
[5] K.C. Chang, K.J. Pearson, T. Zhang, Perron-Frobenius theorem for nonnegative tensors, Commun. Math. Sci. 6 (2008) 507–520.
[6] K.C. Chang, K.J. Pearson, T. Zhang, On eigenvalue problems of real symmetric tensors, J. Math. Anal. Appl. 350 (2009) 416–422.
[7] M. Che, A. Cichocki, Y. Wei, Neural networks for computing best rank-one approximations of tensors and its applications, Neurocomputing 267 (2017) 114–133.
[8] M. Che, L. Qi, Y. Wei, G. Zhang, Geometric measures of entanglement in multipartite pure states via complex-valued neural networks, Neurocomputing 313 (2018) 25–38.
[9] M. Che, Y. Wei, Theory and Computation of Complex Tensors and its Applications, Springer, Singapore, 2020.
[10] C.-K. Chen, S.-H. Ho, Application of differential transformation to eigenvalue problems, Appl. Math. Comput. 79 (1996) 173–188.
[11] L. Chen, L. Han, L. Zhou, Computing tensor eigenvalues via homotopy methods, SIAM J. Matrix Anal. Appl. 37 (2016) 290–319.
[12] C. Chicone, Ordinary Differential Equations with Applications, Texts in Applied Mathematics, vol. 34, Springer-Verlag, New York, 1999.
[13] K.-W.E. Chu, Bordered matrices, singular systems, and ergodic Markov chains, SIAM J. Sci. Statist. Comput. 11 (1990) 688–701.
[14] K.-W.E. Chu, On multiple eigenvalues of matrices depending on several parameters, SIAM J. Numer. Anal. 27 (1990) 1368–1385.
[15] C.-F. Cui, Y.-H. Dai, J. Nie, All real eigenvalues of symmetric tensors, SIAM J. Matrix Anal. Appl. 35 (2014) 1582–1601.
[16] J.W. Demmel, Applied Numerical Linear Algebra, vol. 56, SIAM, 1997. .
[17] W. Ding, L. Qi, Y. Wei, $\mathcal{M}$-tensors and nonsingular $\mathcal{M}$-tensors, Linear Algebra Appl. 439 (2013) 3264–3278.
[18] W. Ding, Y. Wei, Generalized tensor eigenvalue problems, SIAM J. Matrix Anal. Appl. 36 (2015) 1073–1099.
[19] W. Ding, Y. Wei, Solving multi-linear systems with $\mathcal{M}$-tensors, J. Sci. Comput. 68 (2016) 689–715.
[20] M. Fiedler, A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory, Czechoslovak Math. J. 25 (1975) 619–633.
[21] G.H. Golub, C.F. Van Loan, Matrix Computations, 4nd ed., Johns Hopkins University Press, Baltimore, MD, USA, 2013.
[22] M.R. Hestenes, Inversion of matrices by biorthogonalization and related results, J. Soc. Ind. Appl. Math. 6 (1958) 51–90.
[23] R.A. Horn, C.R. Johnson, Matrix Analysis, second ed., Cambridge University Press, New York, 2012.
[24] S. Hu, Z.-H. Huang, C. Ling, L. Qi, On determinants and eigenvalue theory of tensors, J. Symbolic Comput. 50 (2013) 508–531.
[25] A. Ilchmann, T. Reis, eds., Surveys in Differential-Algebraic Equations. I, Differential-Algebraic Equations Forum, Springer, Heidelberg, 2013.
[26] A. Ilchmann (Ed.), Surveys in Differential-Algebraic Equations. IV, Differential-Algebraic Equations Forum, Springer, Cham, 2017.
[27] T. Kato, Perturbation Theory for Linear Operators, Springer, New York, 1966.
[28] B.M. Kiernan, The development of Galois theory from Lagrange to Artin, Arch. History Exact Sci. 8 (1971) 40–154.
[29] T.G. Kolda, J.R. Mayo, Shifted power method for computing tensor eigenpairs, SIAM J. Matrix Anal. Appl. 32 (2010) 1095–1124.
[30] T.G. Kolda, J.R. Mayo, An adaptive shifted power method for computing generalized tensor eigenpairs, SIAM J. Matrix Anal. Appl. 35 (2014) 1563–1581.
[31] R. Lamour, R. März, C. Tischendorf, Differential-Algebraic Equations: a Projector based Analysis, Differential-Algebraic Equations Forum, Springer, Heidelberg, 2013.
[32] P. Lancaster, On eigenvalues of matrices dependent on a parameter, Numer. Math. 6 (1964) 377–387.
[33] X.J. Li, J.M. Yong, Optimal Control Theory for Infinite-Dimensional Systems, Systems & Control: Foundations & Applications, Birkhäuser Boston Inc, Boston, MA, 1995.
[34] L.H. Lim, Singular values and eigenvalues of tensors: a variational approach, in IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, 2006, pp. 129–132. .
[35] A.A. Mailybaev, Computation of multiple eigenvalues and generalized eigenvectors for matrices dependent on parameters, Numer. Linear Algebra Appl. 13 (2006) 419–436.
[36] Y. Miao, L. Qi, Y. Wei, Generalized tensor function via the tensor singular value decomposition based on the t-product, Linear Algebra Appl. 590 (2020) 258–303.
[37] W. Michiels, S.-I. Niculescu, Stability and Stabilization of Time-Delay Systems: An Eigenvalue-based Approach, SIAM, 2007.
[38] C. Mo, C. Li, X. Wang, Y. Wei, Z-eigenvalues based structured tensors: $\mathcal{M}_z$-tensors and strong $\mathcal{M}_z$-tensors, Comput. Appl. Math. 38 (2019) 175.
[39] L. Qi, Eigenvalues of a real supersymmetric tensor, J. Symbolic Comput. 40 (2005) 1302–1324.
[40] L. Qi, H. Chen, Y. Chen, Tensor Eigenvalues and Their Applications, Springer, Singapore, 2018.
[41] L. Qi and Z. Luo, Tensor Analysis: Spectral Theory and Special Tensors, vol. 151, SIAM, 2017..
[42] S. Qiao, X.-Z. Wang, Y. Wei, Two finite-time convergent Zhang neural network models for time-varying complex matrix Drazin inverse, Linear Algebra Appl. 542 (2018) 101–117.
[43] P.J. Rabier, W.C. Rheinboldt, Theoretical and numerical analysis of differential-algebraic equations, in Handbook of numerical analysis, vol. VIII, Handb. Numer. Anal., VIII, North-Holland, Amsterdam, 2002, pp. 183–540. .
[44] F. Rellich, Störungstheorie der spektralzerlegung, Math. Ann. 113 (1937) 600–619.
[45] P. Sirkovic, D. Kressner, Subspace acceleration for large-scale parameter-dependent hermitian eigenproblems, SIAM J. Matrix Anal. Appl. 37 (2016) 695–718.
[46] G. Stewart, J.-G. Sun, Matrix Perturbation Theory, 1990, Academic Press, 1990.
[47] G. Strang, Introduction to Linear Algebra, fifth ed., Wellesley-Cambridge Press, Wellesley, MA, 2016.
[48] J.-G. Sun, Eigenvalues and eigenvectors of a matrix dependent on several parameters, J. Comput. Math 3 (1985) 351–364.
[49] W. Thomson, Theory of Vibration with Applications, CRC Press, 2018.
[50] F. Uhlig, Y. Zhang, Time-varying matrix eigenanalyses via Zhang Neural Networks and look-ahead finite difference equations, Linear Algebra Appl. 580 (2019) 417–435.
[51] G. Wang, Y. Wei, S. Qiao, Generalized Inverses: Theory and Computations, Springer, Singapore, Science Press, Beijing, 2018.
[52] X. Wang, M. Che, L. Qi, Y. Wei, Modified gradient dynamic approach to the tensor complementarity problem, Optim. Methods Software 35 (2020) 394–415.

[53] X. Wang, M. Che, Y. Wei, Recurrent neural network for computation of generalized eigenvalue problem with real diagonalizable matrix pair and its applications, Neurocomputing 216 (2016) 230–241.
[54] X. Wang, M. Che, Y. Wei, Complex-valued neural networks for the Takagi vector of complex symmetric matrices, Neurocomputing 223 (2017) 77–85.
[55] X. Wang, M. Che, Y. Wei, Neural networks based approach solving multi-linear systems with $\mathcal{M}$-tensors, Neurocomputing 351 (2019) 33–42.
[56] X. Wang, M. Che, Y. Wei, Tensor neural network models for tensor singular value decompositions, Comput. Optim. Appl. (2020).
[57] X. Wang, Y. Wei, P.S. Stanimirović, Complex neural network models for time-varying Drazin inverse, Neural Comput. 28 (2016) 2790–2824.
[58] Y. Wei, W. Ding, Theory and Computation of Tensors, Elsevier, Academic Press, 2016.
[59] Y. Wei, P. Stanimirović, M. Petković, Numerical and Symbolic Computations of Generalized Inverses, World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2018.
[60] S. Weinberg, Testing quantum mechanics, Ann. Phys. 194 (1989) 336–386.
[61] J.H. Wilkinson, C. Reinsch, and F.L. Bauer, Handbook for Automatic Computation: Linear Algebra (Grundlehren Der Mathematischen Wissenschaften, vol. 186), Springer-Verlag, 1986. .
[62] Z.-J. Xie, X.-Q. Jin, Y. Wei, Tensor methods for solving symmetric $\mathcal{M}$-tensor systems, J. Sci. Comput. 74 (2019) 412–425.
[63] H. Zhang, L. Wan, Zeroing neural network methods for solving the yang-baxter-like matrix equation, Neurocomputing (2020) 409–418.
[64] Y. Zhang, S.S. Ge, Design and analysis of a general recurrent neural network model for time-varying matrix inversion, IEEE Trans. Neural Networks 16 (2005) 1477–1490.
[65] Y. Zhang, D. Guo, Zhang Functions and Various Models, Springer-Verlag, Berlin Heidelberg, Dordrecht London, 2015.
[66] Y. Zhang, J. Wang, Recurrent neural networks for nonlinear output regulation, Automatica J. IFAC 37 (2001) 1161–1173.
[67] Y. Zhang, M. Yang, C. Li, F. Uhlig, H. Hu, New continuous ZD model for computation of time-varying eigenvalues and corresponding eigenvectors, Preprint, (2018). .
[68] Y. Zhang, C. Yi, Zhang Neural Networks and Neural-Dynamic Method, Nova Science Publishers, New York, 2011.



Changxin Mo received his B.S. degree in Mathematics from Sun Yat-Sen University, Guangdong,China, in 2016. He is currently pursuing the Ph.D. degree in Computational Mathematics at Fudan University, Shanghai, China. His current research interests include numerical and multilinear linear algebra, neural networks.



Xuezhong Wang received the Ph.D in Computational Mathematics from School of Mathematical Sciences at Fudan University in 2018, B.S. degree in Mathematics from Northwest Normal University, Lanzhou,China, in 2000, and M.S. degree in Computational Mathematics from UESTC, Chengdu,China, in 2007, respectively. He is currently a Professor at the School of Mathematics and Statistics, Hexi University. His current research interests include neural network and numerical algebra.

Yimin Wei received the B.S. degree in computational mathematics from Shanghai Normal University of Shanghai in China, and Ph.D. degree in computational mathematics from Fudan University of Shanghai in China, 1991 and 1997, respectively. He was a Lecturer with the Department of Mathematics, Fudan University, from 1997 to 2000, a Visiting Scholar with the Division of Engineering and Applied Science, Harvard University, Boston, MA, USA, from 2000 to 2001, and an Associate Professor with the School of Mathematical Sciences, Fudan University, from 2001 to 2005. He is currently a Professor with the School of Mathematical Sciences, Fudan University. His current research interests include numerical linear algebra and its applications, sensitivity in computational linear control, and perturbation analysis of the matrix and the tensor. He is author of more than 150 technical journal papers and 5 monographs published by Elsevier, Springer, World Scientifc and Science Press.