Determinant Formula and Error Formula for Tensor Pade-Type Approximation

Shimin Liu

***Abstract*—The determinant formula for tensor Padé-type approximation (TPTA) is proposed to compute the tensor exponential function $e^{\mathcal{A}t}$, where $\mathcal{A}$ represents an order-p ($p \geq 3$) tensor in n-dimensional Euclidean space. Furthermore, an error formula for TPTA is also derived. It is demonstrated that the determinant formula can be utilized to approximate any arbitrary tensor function defined by a power series. Numerical illustrations are presented to showcase the efficiency of the determinant formula.**

***Index Terms*—tensor Padé-type approximation, tensor exponential function, determinant formula, error formula, power series.**

I. Introduction

CONSIDER the tensor exponential function of the following form

$$e^{\mathcal{A}t} = \sum_{n=0}^{\infty} \frac{t^n}{n!}\mathcal{A}^n, \quad (1)$$

in which $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ is a tensor with order $p \geq 2$. This kind of tensor exponential function has a wide range of applications in quantum mechanics and materials mechanics [1, 2]. For instance, the utilization of the exponential function for second order tensors is commonly observed in numerical integration techniques applied to rate equations $\dot{\mathcal{X}} = \mathcal{A}\mathcal{X}$ and single crystal model $F = F^e F^p$ [3], where F^e and F^p represent the elastic and plastic of the crystal, respectively. More precisely, in the case of a solitary crystal possessing n_{syst} slip systems in total, the plastic part, F^p, is determined by the subsequent differential equation

$$\dot{F}^p F^{p-1} = \sum_{\alpha=1}^{n_{\text{syst}}} \dot{\gamma}^{\alpha} s_0^{\alpha} \otimes m_0^{\alpha}, \quad (2)$$

where $\dot{\gamma}^{\alpha}$ represents the proportion of system α in the overall non-elastic deformation rate. Both the slip and normal directions of system α are represented by the s_0^{α} and m_0^{α}, respectively. The tensor exponential function [4, 5] can be employed to discretize the aforementioned equation (2) implicitly. By employing this approach, we can acquire an exponential estimation implicitly for equation (2) as follows

$$F_{n+1}^p = \mathbf{exp}\left(\sum_{\alpha=1}^{n_{\text{syst}}} \Delta\gamma^{\alpha} s_0^{\alpha} \otimes m_0^{\alpha}\right) F_n^p.$$

Therefore, in order to calculate F_{n+1}^p recursively, we have to first calculate

$$\mathbf{exp}\left(\sum_{\alpha=1}^{n_{\text{syst}}} \Delta\gamma^{\alpha} s_0^{\alpha} \otimes m_0^{\alpha}\right).$$

For this issue, there exist several methods for computing the above tensor exponential function represented by tensor power series, For instance, Lu [6] proposed a method to compute the second-order tensor exponential function as follows

$$F(\mathcal{A}) = e^{\mathcal{A}} = \sum_{n=0}^{\infty} \frac{1}{n!}\mathcal{A}^n.$$

The method proposed in [6] demonstrates that $F(\mathcal{A})$ can be derived by taking the derivative of a function associated with the eigenvalues of the second tensor $\mathcal{A}$. However, this approach requires calculating the eigenvalues of $\mathcal{A}$, which involves a considerable amount of computation for high-dimensional problems [7, 8]. An alternative approach for the computation of the exponential function of a second tensor involves truncating infinite series [1]. The round-off and termination criterion have a constraining impact on the precision and efficiency of this approach. As $F(\mathcal{A})$ is a second-order tensor A's isotropic function [9], it is possible to express $F(\mathcal{A})$ directly using polynomial expressions of A [10, 11]. Recently, based on this observation, Gu and Liu [12] proposed a new algorithm for computing the higher order tensor exponential function represented by tensor power series, i.e., tensor Padé-type approximation (TPTA). TPTA is a rational expression with a tensor as its numerator and a scalar as its denominator. Although it is highly effective for high-order tensor power series, its computation is limited to order $(n - 1/n)$.

In this research, we propose an innovative numerical approach to calculate the exponential function $e^{\mathcal{A}t}$ associated with a pth order tensor $\mathcal{A}$ with $p \geq 3$. We demonstrate that $e^{\mathcal{A}t}$ can be approximated by a rational function featuring a tensor numerator with degree m and a scalar denominator with degree n, which remains non-zero at zero. Moreover, the power series expansion of this rational function in ascending powers of the variable aligns with the first $m+1$ terms of the series (1). This methodology draws inspiration from the concept presented in [13] to simplify complex multivariable systems with the use of tensor function. This function can be expanded into a series of powers with coefficients in tensor form. In this study, we have systematically expanded this method to encompass power series associated with tensor with order $p \geq 3$.

This paper is structured in the following manner. In Section II, we provide some preliminaries. In Section III, we introduce the definition of the tensor Padé-type approximation. In Section IV, we deduce the determinate formula and error formula for TPTA by using generalized linear functional. In Section V, numerical examples are provided and examined. Finally, we conclude the paper with final remarks in Section VI.

Manuscript received March 12, 2024; revised July 20, 2024.
This work was supported by the Youth Foundation of Changshu Institute of Technology (22ZK3764).
Shimin Liu is a lecturer of the School of Mathematics and Statistics, Changshu Institute of Technology, Suzhou, Jiangsu, 215500, China (e-mail: xqlsm1982@sina.com).

II. Preliminaries

How to expand the tensor exponential function $e^{\mathcal{A}t}$ into a power series representation with respect to pth order tensors is the primary problem in approximating the tensor exponential function. For order-2 symmetric tensor $\mathcal{A}$, the Cayley-Hamilton theorem [14] can be used to compute higher powers of $\mathcal{A}$. Nevertheless, this theorem does not hold true for tensors with an order greater than 3. In this section, we will employ the t-product [15], see also [16], of tensors to derive higher pth order $(p \geq 3)$ tensor powers. Initially, we present certain notations and fundamental definitions that will be utilized subsequently.

Let

$$\mathcal{A} = (a_{i_1 i_2 \cdots i_p}) \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$$

represent a pth order tensor $\mathcal{A}$. It is commonly accepted that a matrix can be classified as a tensor of the second order. We use $\mathcal{A}_i \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_{p-1}}$ to present the tensor with a dimension of $p-1$ that generated by fixing the pth index of $\mathcal{A}$ at i, $i = 1, \cdots, n_p$. For example, consider a fourth order tensor

$$\mathcal{A} = (a_{i_1 i_2 i_3 i_4}) \in \mathbb{R}^{2 \times 2 \times 3 \times 4},$$

then $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ and $\mathcal{A}_4$, which have the following form

$$\mathcal{A}_i = \left(\begin{array}{cc|cc|cc} a_{111i} & a_{121i} & a_{112i} & a_{122i} & a_{113i} & a_{123i} \\ a_{211i} & a_{221i} & a_{212i} & a_{222i} & a_{213i} & a_{223i} \end{array} \right)$$

are four $2 \times 2 \times 3$ tensors generated by fixing the 4th index of $\mathcal{A}$.

Definition 1. *([15]) Assume $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$, then the circulant tensor of $\mathcal{A}$ is defined as*

$$circ(\mathcal{A}) = \begin{pmatrix} \mathcal{A}_1 & \mathcal{A}_{n_p} & \mathcal{A}_{n_p-1} & \cdots & \mathcal{A}_2 \\ \mathcal{A}_2 & \mathcal{A}_1 & \mathcal{A}_{n_p} & \cdots & \mathcal{A}_3 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathcal{A}_{n_p} & \mathcal{A}_{n_p-1} & \cdots & \mathcal{A}_2 & \mathcal{A}_1 \end{pmatrix},$$

in which $\mathcal{A}_i \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_{p-1}}$, $i = 1, 2, \cdots, n_p$.

Besides, we use the symbol *unfold*$(\cdot)$ to denote a function that transforms an $n_1 \times n_2 \times \cdots \times n_p$ tensor into an $n_1 \times n_p \times n_2 \times \cdots \times n_{p-1}$ tensor using the following approach:

$$unfold(\mathcal{A}) = \begin{pmatrix} \mathcal{A}_1 \\ \mathcal{A}_2 \\ \vdots \\ \mathcal{A}_{n_p} \end{pmatrix}.$$

Similarly, we use the symbol *fold*$(\cdot)$ to denote the function that convert an $n_1 \times n_p \times n_2 \times \cdots \times n_{p-1}$ tensor into an $n_1 \times n_2 \times \cdots \times n_p$ tensor, i.e., $fold(unfold(\mathcal{A})) = \mathcal{A}$.

Following this, we shall present the t-product of two tensors.

Definition 2. *([15]) For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$, $\mathcal{B} \in \mathbb{R}^{n_2 \times l \times n_3 \cdots \times n_p}$, the t-product of $\mathcal{A}$ and $\mathcal{B}$ is denoted by*

$$\mathcal{A} * \mathcal{B} = fold(circ(\mathcal{A}) * unfold(\mathcal{B})).$$

Remark 1. *When $\mathcal{A}$ and $\mathcal{B}$ are second-order tensors, the t-product can be substituted with conventional matrix multiplication.*

Remark 2. *By the definition of the t-product, we have $\mathcal{A}^k = \mathcal{A} * \mathcal{A} * \cdots * \mathcal{A}$.*

Remark 3. *One notable attribute of the t-product is its ability to maintain the order of multiplication result for two tensors, unlike other tensor multiplications suggested in [17]. This unique characteristic influenced our decision to adopt the t-product as the preferred method for tensor multiplication.*

The tensor exponential function is a specialized mathematical operation applied to tensors, similar in concept to the conventional exponential function. It can be formulated as follows.

Definition 3. *Assume $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$. The function $e^{\mathcal{A}t}$ or $exp(\mathcal{A}t)$ represents the tensor exponential with respect to t, which can be expressed as the following power series*

$$e^{\mathcal{A}t} = \sum_{k=0}^{\infty} \frac{1}{k!} (\mathcal{A}t)^k,$$

in which $\mathcal{A}^0$ is an $n_1 \times n_2 \times \cdots \times n_p$ identity tensor $\mathcal{I}$.

Definition 4. *([15]) An identity tensor with dimensions $n \times n \times l_1 \times \cdots \times l_{p-2}$ can be defined as follows: $\mathcal{I}_1$ is an $(p-1)$-th order identity tensor with dimensions $n \times n \times l_1 \times \cdots \times l_{p-3}$, while $\mathcal{I}_j$ (for $j = 2, 3, \cdots, l_{p-2}$) represents an $(p-1)$-th order zero tensor.*

Obviously, the above tensor power series always converges for all $\mathcal{A}$ and t as k increase, so the exponential function in relation to the variable t is well-defined.

The tensor exponential function defined in Definition 3 satisfies the following key property, which will be utilized in the next section.

Property 1. *If $\mathcal{A} * \mathcal{B} = \mathcal{B} * \mathcal{A}$, then $e^{\mathcal{A}t} * e^{\mathcal{B}t} = e^{\mathcal{B}t} * e^{\mathcal{A}t} = e^{(\mathcal{A}+\mathcal{B})t}$.*

Proof.

$$\begin{aligned} e^{\mathcal{A}t} * e^{\mathcal{B}t} &= \left(\sum_{k=0}^{\infty} \frac{(\mathcal{A}t)^k}{k!} \right) * \left(\sum_{k=0}^{\infty} \frac{(\mathcal{B}t)^k}{k!} \right) \\ &= \sum_{m=0}^{\infty} \sum_{k=0}^{m} \frac{(\mathcal{A}t)^k}{k!} * \frac{(\mathcal{B}t)^{m-k}}{(m-k)!}. \end{aligned} \tag{3}$$

Since

$$\frac{1}{m!} C_m^k = \frac{1}{m!} \cdot \frac{m!}{k!(m-k)!} = \frac{1}{k!} \cdot \frac{1}{(m-k)!},$$

we can obtain

$$\frac{(\mathcal{A}t)^k}{k!} * \frac{(\mathcal{B}t)^{m-k}}{(m-k)!} = \frac{1}{m!} C_m^k (\mathcal{A}t)^k * (\mathcal{B}t)^{m-k}.$$

Furthermore, based on the assumption that $\mathcal{A} * \mathcal{B} = \mathcal{B} * \mathcal{A}$ and Binomial theorem, we can further simplify the above relation as follows

$$\begin{aligned} \sum_{k=0}^{m} \frac{(\mathcal{A}t)^k}{k!} * \frac{(\mathcal{B}t)^{m-k}}{(m-k)!} &= \sum_{k=0}^{m} \frac{1}{m!} C_m^k (\mathcal{A}t)^k * (\mathcal{B}t)^{m-k} \\ &= \frac{1}{m!} \sum_{k=0}^{m} C_m^k (\mathcal{A}t)^k * (\mathcal{B}t)^{m-k} \\ &= \frac{1}{m!} (\mathcal{A}t + \mathcal{B}t)^m. \end{aligned}$$

Substituting the above formula into the relation (3), we have

$$e^{\mathcal{A}t} * e^{\mathcal{B}t} = \sum_{m=0}^{\infty} \frac{1}{m!}(\mathcal{A}t+\mathcal{B}t)^m = e^{(\mathcal{A}+\mathcal{B})t} = e^{\mathcal{B}t} * e^{\mathcal{A}t}. \qquad \square$$

Just like the definition of matrix norm, the norm of a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ is defined as [18]

$$\|\mathcal{A}\| = \sqrt{\sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_p=1}^{n_p} a_{i_1 i_2 \cdots i_p}^2}. \tag{4}$$

Consequently, the inner product of two same dimension tensors $\mathcal{A}$, $\mathcal{B} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ is the sum of the products of their entries, i.e.,

$$(\mathcal{A}, \mathcal{B}) = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_p=1}^{n_p} a_{i_1 i_2 \cdots i_p} b_{i_1 i_2 \cdots i_p}.$$

Then, based on (4), it holds that $(\mathcal{A}, \mathcal{A}) = \|\mathcal{A}\|^2$.

III. Tensor Padé-type approximation

In this section, we will directly extend the definition of $(n-1/n)$ order TPTA [12] to the case of (m/n).

Consider a power series $f(x)$ with coefficients in tensor form:

$$f(x) = \mathcal{A}_0 + \mathcal{A}_1 x + \mathcal{A}_2 x^2 + \cdots + \mathcal{A}_n x^n + \cdots, \tag{5}$$

where $\mathcal{A}_i \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ and $x \in \mathbb{R}$. Let $\mathbf{P}$ represent the collection of scalar polynomials in a single real variable with coefficients from the field of real numbers $\mathbb{R}$.

Consider $\phi^{(q)} : \mathbf{P} \to \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_p}$ as a linear operator of a generalized nature on $\mathbf{P}$, with respect to t, defined as

$$\phi^{(q)}(t^i) = \mathcal{A}_{q+i}, i = 0, 1, 2, \cdots. \tag{6}$$

Next, we consider the polynomial $\mathcal{W}_q(x)$ associated to $v_n(x) = b_0 + b_1 x + \cdots + b_n x^n$ with $b_n \neq 0$ as follows

$$\mathcal{W}_q(x) = \phi^{(q)}\left(\frac{v_n(x) - v_n(t)}{x - t}\right), \quad q = m - n + 1. \tag{7}$$

Furthermore, based on the definitions (6) and (7), and by defining

$$\begin{aligned} \widetilde{v}_n(x) &= x^n v_n(x^{-1}), \\ \widetilde{\mathcal{W}}_q(x) &= x^{n-1} \mathcal{W}_q(x^{-1}), \end{aligned} \tag{8}$$

and

$$\mathcal{P}_{mn}(x) = \widetilde{v}_n(x) \sum_{i=0}^{m-n} \mathcal{A}_i x^i + x^{m-n+1} \widetilde{\mathcal{W}}_q(x), \quad m \geq n, \tag{9}$$

we have the ability to acquire the subsequent statement.

Theorem 1. *Let $\widetilde{v}_n(0) \neq 0$, then*

$$\mathcal{P}_{mn}(x)/\widetilde{v}_n(x) - f(x) = O(x^{m+1}).$$

Proof. Assume that

$$f_{m-n+1}(x) = \sum_{j=0}^{\infty} \mathcal{A}_{m-n+1+j} x^j,$$

then, through direct mathematical calculations, we have

$$x^{m-n+1} f_{m-n+1}(x) = f(x) - \sum_{i=0}^{m-n} \mathcal{A}_i x^i. \tag{10}$$

Expanding (7) and substituting it into (8), we can obtain

$$\widetilde{\mathcal{W}}_q(x) = \sum_{q=0}^{n-1} \left(\sum_{i=0}^{q} b_{n-q+i} \mathcal{A}_{m-n+1+i} \right) x^q.$$

Expanding $\widetilde{v}_n(x) f_{m-n+1}(x)$ directly yields

$$\begin{aligned} \widetilde{v}_n(x) f_{m-n+1}(x) &= x^n v_n(x^{-1}) \left(\sum_{j=0}^{\infty} \mathcal{A}_{m-n+1+j} x^j \right) \\ &= \left(\sum_{i=0}^{n} b_i x^{n-i} \right) \left(\sum_{j=0}^{\infty} \mathcal{A}_{m-n+1+j} x^j \right) \\ &= \sum_{q=0}^{\infty} \left(\sum_{i=0}^{q} b_{n-q+i} \mathcal{A}_{m-n+1+i} \right) x^q. \end{aligned}$$

Therefore, it can be inferred from equations (9) and (10) that

$$\begin{aligned} &\widetilde{v}_n(x) f(x) - \mathcal{P}_{mn}(x) \\ &= \widetilde{v}_n(x) \left(\sum_{i=0}^{m-n} \mathcal{A}_i x^i + x^{m-n+1} f_{m-n+1}(x) \right) \\ &\quad - \widetilde{v}(x) \sum_{i=0}^{m-n} \mathcal{A}_i x_i + x^{m-n+1} \widetilde{\mathcal{W}}_q(x) \\ &= x^{m-n+1} \left(\widetilde{v}_n(z) f_{m-n+1} - \widetilde{\mathcal{W}}_q(x) \right) \\ &= x^{m-n+1} \left\{ \sum_{q=n}^{\infty} (\sum_{i=0}^{q} b_{n-q+i} \mathcal{A}_{m-n+1+i}) x^q \right\} \\ &= O(x^{m+1}). \qquad \square \end{aligned}$$

Therefore, $R_{m,n}(x) = \mathcal{P}_{mn}(x)/\widetilde{v}_n(x)$ can be used as an estimation of the tensor series $f(x)$ defined in (5). It is referred to as an (m/n)-order tensor Padé-type approximation [12] for the provided series (5), and it is denoted as $(m/n)_f(x)$.

Remark 4. *If $\mathcal{A}_i, i = 1, 2, \cdots$ are second tensors, then the TPTA automatically reduces to matrix Padé-type approximation, see also [13, 19].*

Remark 5. *The generating polynomial $v_n(x)$ of $(m/n)_f(x)$ can be freely selected.*

IV. Error formula and Determinant formula for TPTA

On the basis of the definition of $(m/n)_f(x)$, we can further provide its error formula and determinant formula, while the $(n-1/n)$ order TPTA does not have these two formulas, which is also the main contribution of this article. The determinant formula can simplify the calculation of $(m/n)_f(x)$, while the error formula determines suitable values for m and n based on the desired level of accuracy for a specific problem.

A. Error formula for TPTA

For given m and n, using the definition of generalized linear functional $\phi^{(q)}$, we can get the following error expression of $(m/n)_f(x)$ for tensor power series $f(x)$.

Theorem 2. *Let $\widetilde{v}_n(0) \neq 0$, then for given m and n*

$$f(x) - (m/n)_f(x) = \frac{x^{m+1}}{\widetilde{v}_n(x)} \phi^{(m-n+1)} \left(\frac{v_n(t)}{1 - tx} \right).$$

Proof. As $\phi^{(m-n+1)}$ solely operates on t within $\mathbf{P}$, we can deduced from relations (8) and (9) that

$$\begin{aligned}
&\mathcal{P}_{mn}(x)\\
&= \widetilde{v}_n(x)\sum_{i=0}^{m-n}\mathcal{A}_i x^i + x^{m-n+1}\widetilde{\mathcal{W}}_q(x)\\
&= \widetilde{v}_n(x)\sum_{i=0}^{m-n}\mathcal{A}_i x^i\\
&\quad + \phi^{(m-n+1)}\left(\frac{x^{m+1}v_n(x^{-1}) - x^{m+1}v_n(t)}{1-tx}\right)\\
&= \widetilde{v}_n(x)\left[\sum_{i=0}^{m-n}\mathcal{A}_i x^i + x^{m-n+1}\phi^{(m-n+1)}\left(\frac{1}{1-tx}\right)\right]\\
&\quad - x^{m+1}\phi^{(m-n+1)}\left(\frac{v_n(t)}{1-tx}\right)\\
&= \widetilde{v}_n(x)f(x) - x^{m+1}\phi^{(m-n+1)}\left(\frac{v_n(t)}{1-tx}\right).
\end{aligned}$$

Then it immediately holds

$$f(x) - \mathcal{P}_{mn}(x)/\widetilde{v}_n(x) = \frac{x^{m+1}}{\widetilde{v}_n(x)}\phi^{(m-n+1)}\left(\frac{v_n(t)}{1-tx}\right). \qquad \square$$

B. Determinant formula for TPTA

Substituting the fact that

$$(1-tx)^{-1} = 1 + tx + (tx)^2 + \cdots + (tx)^n + \cdots,$$

into the above error formula, we can easily obtain

$$\begin{aligned}
f(x) &- (m/n)_f(x)\\
&= \frac{x^{m+1}}{\widetilde{v}_n(x)}\phi^{(m-n+1)}\left(\frac{v_n(t)}{1-tx}\right)\\
&= \frac{x^{m+1}}{\widetilde{v}_n(x)}\phi^{(m-n+1)}(v_n(t)\\
&\quad + v_n(t)tx + v_n(t)t^2x^2 + \cdots)\\
&= \frac{x^{m+1}}{\widetilde{v}_n(x)}(\phi^{(m-n+1)}(v_n(t))\\
&\quad + \phi^{(m-n+1)}(v_n(t)t)x\\
&\quad + \phi^{(m-n+1)}(v_n(t)t^2)x^2 + \cdots). \qquad (11)
\end{aligned}$$

If we enforce $v_n(t)$ satisfies $\phi^{(m-n+1)}(v_n(t)) = 0$, the first term of (11) vanishes, resulting in an approximation order of $m+2$. Additionally, if we set $\phi^{(m-n+1)}(tv_n(t)) = 0$, then the second term of (11) disappears as well, leading to an approximation order of $m+3$ and so forth. It should be noted that a rational function is essentially defined by its numerator and denominator with a multiplying factor [20]. It indicates that $(m/n)_f(x)$ relies on n constants. Therefore, the polynomial $v_n(t)$ satisfies

$$\phi^{(m-n+1)}(v_n(t)t^k) = 0, \quad k = 0, 1, 2, \cdots, n-1. \qquad (12)$$

The determination of the Hankel matrix with respect to the series $f(x)$ is denoted as

$$det(H_n(\mathcal{A}_{m-n+1})) = \begin{vmatrix} (\mathcal{A}_{m-n+1},\mathcal{A}_{m-n+1}) & \cdots & (\mathcal{A}_m,\mathcal{A}_{m-n+1})\\ (\mathcal{A}_{m-n+2},\mathcal{A}_{m-n+2}) & \cdots & (\mathcal{A}_{m+1},\mathcal{A}_{m-n+2})\\ \vdots & \ddots & \vdots\\ (\mathcal{A}_m,\mathcal{A}_m) & \cdots & (\mathcal{A}_{m+n-1},\mathcal{A}_m) \end{vmatrix}.$$

Then, we can conclude the following determinant formula for TPTA.

Theorem 3. *Assume* $det(H_n(\mathcal{A}_{m-n+1})) \neq 0$, *then*

$$(m/n)_f(x) = \mathcal{P}_{mn}(x)/q_{mn}(x),$$

here,

$$\mathcal{P}_{mn}(x) = \begin{vmatrix} H_n(\mathcal{A}_{m-n+1}) & \vec{\alpha}\\ \vec{\eta}^T & \sum_{i=0}^m \mathcal{A}_i x^i \end{vmatrix},$$
$$q_{mn}(x) = \begin{vmatrix} H_n(\mathcal{A}_{m-n+1}) & \vec{\alpha}\\ \vec{\xi}^T & 1 \end{vmatrix}, \qquad (13)$$

with

$$\begin{aligned}
\vec{\alpha} &= ((\mathcal{A}_{m+1},\mathcal{A}_{m-n+1}),\cdots,(\mathcal{A}_{m+n},\mathcal{A}_m))^T,\\
\vec{\eta}^T &= \left(\sum_{i=n}^m \mathcal{A}_{i-n}x^i, \sum_{i=n-1}^m \mathcal{A}_{i-n+1}x^i, \cdots, \sum_{i=1}^m \mathcal{A}_{i-1}x^i\right),\\
\vec{\xi}^T &= (x^n, x^{n-1}, \cdots, x).
\end{aligned}$$

Proof. From conditions (12), we know that,

$$\begin{cases} \phi^{(m-n+1)}(b_0 + b_1 t + \cdots + b_n t^n) = 0,\\ \phi^{(m-n+1)}(b_0 t + b_1 t^2 + \cdots + b_n t^{n+1}) = 0,\\ \cdots\cdots\cdots\\ \phi^{(m-n+1)}(b_0 t^{n-1} + b_1 t^n + \cdots + b_n t^{2n-1}) = 0. \end{cases}$$

By using the generalized linear functional on $\phi^{(q)}(t^i)$ defined in (6), we can further obtain

$$\begin{cases} b_0\mathcal{A}_{m-n+1} + b_1\mathcal{A}_{m-n+2} + \cdots + b_n\mathcal{A}_m = 0,\\ b_0\mathcal{A}_{m-n+2} + b_1\mathcal{A}_{m-n+3} + \cdots + b_n\mathcal{A}_{m+1} = 0,\\ \cdots\cdots\cdots\\ b_0\mathcal{A}_m + b_1\mathcal{A}_{m+1} + \cdots + b_n\mathcal{A}_{m+n} = 0. \end{cases}$$

Taking the inner product of the above n equations with $\mathcal{A}_{m-n+1}, \cdots, \mathcal{A}_{m+n}$, respectively yields

$$\begin{cases} b_0(\mathcal{A}_{m-n+1},\mathcal{A}_{m-n+1}) + \cdots + b_n(\mathcal{A}_{m-n+1},\mathcal{A}_m) = 0,\\ b_0(\mathcal{A}_{m-n+2},\mathcal{A}_{m-n+2}) + \cdots + b_n(\mathcal{A}_{m-n+2},\mathcal{A}_{m+1}) = 0,\\ \cdots\cdots\cdots\\ b_0(\mathcal{A}_{m+n},\mathcal{A}_m) + \cdots + b_n(\mathcal{A}_{m+n},\mathcal{A}_{m+n}) = 0. \end{cases}$$

By setting $b_n = 1$ and employing Cramer's rule, the values of $b_0, \cdots, b_{n-1}$ can be uniquely determined. Then, by means of (8), $q_{mn}(x)$ is holds. Now, we prove the quantity $\mathcal{P}_{mn}(x)$. By calculating $\widetilde{v}_n(x)f(x)$, we can obtain

$$\begin{aligned}
\widetilde{v}_n(x)f(x) &= a_0 + a_1 x + \cdots + a_m x^m + O(x^{m+1})\\
&= (b_0 + b_1 x + \cdots + b_n x^n)\left(\sum_{i=0}^m \mathcal{A}_i x^i\right)\\
&\quad + O(x^{m+1})
\end{aligned}$$

and

$$\begin{aligned}
\mathcal{P}_{mn}(x) &= b_0\mathcal{A}_0 + (b_0\mathcal{A}_1 + b_1\mathcal{A}_0)x + \cdots\\
&\quad + (b_0\mathcal{A}_m + b_1\mathcal{A}_{m-1} + \cdots + b_n\mathcal{A}_{m-n})x^m\\
&= \left(\sum_{i=0}^m \mathcal{A}_i x^i\right)b_0 + \left(\sum_{i=1}^m \mathcal{A}_{i-1}x^i\right)b_1 + \cdots\\
&\quad + \left(\sum_{i=n}^m \mathcal{A}_{i-n}x^i\right)b_n.
\end{aligned}$$

Hence, the determinant expression for $\mathcal{P}_{mn}(x)$ in (13) is established by utilizing the formula of $q_{mn}(x)$. $\square$

Example 1. *Consider a $2 \times 2 \times 2 \times 2$ tensor $\mathcal{A}$ of the form*

$$\mathcal{A} = \left(\begin{array}{cc|cc|cc|cc} 1 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 1 \end{array} \right).$$

Please seek the TPTA of type $(3/3)$ *for* $e^{\mathcal{A}x}$.

Now we will apply determinant formula (13) of TPTA to compute $(3/3)_{e^{\mathcal{A}x}}(x)$. Expanding $e^{\mathcal{A}x}$ into the tensor power series by means of t-product firstly, we can get

$$\begin{aligned} e^{\mathcal{A}x} = & \left(\begin{array}{cc|cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right) \\ & + \left(\begin{array}{cc|cc|cc|cc} 1 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 1 \end{array} \right) x \\ & + \left(\begin{array}{cc|cc|cc|cc} \frac{2}{3} & 0 & 0 & \frac{4}{9} & 0 & \frac{4}{9} & \frac{4}{9} & 0 \\ 0 & \frac{2}{3} & \frac{4}{9} & 0 & \frac{4}{9} & 0 & 0 & \frac{4}{9} \end{array} \right) x^2 \\ & + \cdots . \end{aligned}$$

Subsequently, we employ (4) to calculate $H_n(\mathcal{A}_{m-n+1})$ with $m = n = 3$ and acquire

$$H_3(\mathcal{A}_1) = \begin{pmatrix} \frac{8}{3} & \frac{56}{27} & \frac{112}{81} \\ \frac{56}{27} & \frac{976}{729} & \frac{488}{729} \\ \frac{515}{577} & \frac{486}{1093} & \frac{535}{3008} \end{pmatrix}.$$

Substituting it into (13), we can obtain

$$\begin{aligned} q_{33}(x) = & \begin{vmatrix} \frac{8}{3} & \frac{56}{27} & \frac{112}{81} & \frac{488}{729} \\ \frac{56}{27} & \frac{976}{729} & \frac{488}{729} & \frac{584}{2189} \\ \frac{515}{577} & \frac{486}{1093} & \frac{535}{3008} & \frac{151}{2548} \\ x^3 & x^2 & x & 1 \end{vmatrix} \\ = & \frac{23}{13806}x^3 - \frac{1249}{86337}x^2 + \frac{324}{6697}x - \frac{563}{9118} \end{aligned}$$

and

$$\begin{aligned} \mathcal{P}_{33}(x) = & (p_{11}|p_{12}|p_{13}|p_{14})\, x^3 + (p_{21}|p_{22}|p_{23}|p_{24})\, x^2 \\ & + (p_{31}|p_{32}|p_{33}|p_{34})\, x + (p_{41}|p_{42}|p_{43}|p_{44}), \end{aligned}$$

with

$$p_{11} = \begin{pmatrix} \frac{-46}{13465} & 0 \\ 0 & \frac{100}{10779} \end{pmatrix}, \quad p_{12} = \begin{pmatrix} 0 & \frac{-537}{171034} \\ \frac{-537}{171034} & 0 \end{pmatrix},$$

$$p_{13} = \begin{pmatrix} 0 & \frac{-537}{171034} \\ \frac{-537}{171034} & 0 \end{pmatrix}, \quad p_{14} = \begin{pmatrix} \frac{-537}{171034} & 0 \\ 0 & \frac{-310}{19579} \end{pmatrix},$$

$$p_{21} = \begin{pmatrix} \frac{-73}{10068} & 0 \\ 0 & \frac{-86}{2117} \end{pmatrix}, \quad p_{22} = \begin{pmatrix} 0 & \frac{-154}{13609} \\ \frac{-154}{13609} & 0 \end{pmatrix},$$

$$p_{23} = \begin{pmatrix} 0 & \frac{-154}{13609} \\ \frac{-154}{13609} & 0 \end{pmatrix}, \quad p_{24} = \begin{pmatrix} \frac{-154}{13609} & 0 \\ 0 & \frac{21}{1003} \end{pmatrix},$$

$$p_{31} = \begin{pmatrix} \frac{-125}{9352} & 0 \\ 0 & \frac{269}{9677} \end{pmatrix}, \quad p_{32} = \begin{pmatrix} 0 & \frac{-563}{27354} \\ \frac{-563}{27354} & 0 \end{pmatrix},$$

$$p_{33} = \begin{pmatrix} 0 & \frac{-563}{27354} \\ \frac{-563}{27354} & 0 \end{pmatrix}, \quad p_{34} = \begin{pmatrix} \frac{-563}{27354} & 0 \\ 0 & \frac{-563}{9118} \end{pmatrix},$$

$$p_{41} = \begin{pmatrix} \frac{-563}{9118} & 0 \\ 0 & \frac{-563}{9118} \end{pmatrix}, \quad p_{42} = p_{43} = p_{44} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

It can be readily confirmed that

$$\mathcal{P}_{33}(x) - q_{33}(x)e^{\mathcal{A}x} = O(x^4),$$

thus we have

$$(3/3)_{e^{\mathcal{A}x}}(x) = \mathcal{P}_{33}(x)/q_{33}(x).$$

V. Numerical experiments

Now we demonstrate the precision of the TPTA algorithm in calculating the tensor exponential function through two numerical examples. We take second-order tensor $\mathcal{A}$ from [6] and compute $e^{\mathcal{A}t}$ be means of (13) for specific values of t. All experiments were performed using MATLAB R2022a on an Intel(R) Core(TM) i5 CPU, @2.67 GHz 4.00 GB memory (Windows 11).

Example 2. *Consider a Jordan form*

$$\mathcal{A} = \begin{pmatrix} 0.5 & 1 & 0 \\ 0 & 0.5 & 1 \\ 0 & 0 & 0.5 \end{pmatrix},$$

the tensor exponential function $e^{\mathcal{A}t} = \sum_{n=0}^{\infty} \frac{\mathcal{A}^n t^n}{n!}$.

As the proposed method in [6], the exponential function

$$e^{\mathcal{A}t} = \begin{pmatrix} f(0.5t) & f'(0.5t) & \frac{1}{2}f''(0.5t) \\ 0 & f(0.5t) & f'(0.5t) \\ 0 & 0 & f(0.5t) \end{pmatrix},$$

where $f(x) = e^x$. We employed MATLAB function *funm* to compute $f(0.5t)$, $f'(0.5t)$ and $f''(0.5t)$, the exact tensor exponential $e^{\mathcal{A}t}$ is derived by calculating a Taylor polynomial of order 200 utilizing MATLAB's Symbolic Math Toolbox when t is a specific value. Figure 1 illustrates the Frobenius norm-based relative error in base-10 logarithm between TPTA of different orders and the true solution with respect to different t. We can derive the following observations based on the curve presented in Figure 1.

- MATLAB's function *funm* remains the most superior code in overall performance.
- The TPTA algorithm has proven to be highly efficient as well, since it can get very accurate value with only a few items of power series expansion. More precisely, $(7/7)_{e^{\mathcal{A}t}}$ is more accurate than *funm* in some points.

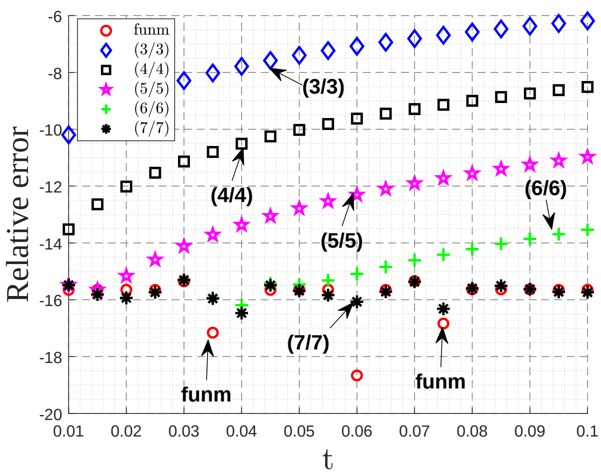

Fig. 1. Relative errors versus t for MATLAB's *funm* method and $(k/k)_{e^{\mathcal{A}t}}(t)$ method for Example 2.

Example 3. *Consider a tensor exponential function* $e^{\mathcal{A}t} = \sum_{n=0}^{\infty} \frac{\mathcal{A}^n t^n}{n!}$ *by randomly selected third-order tensor as follows:*

$$\mathcal{A} = \left(\begin{array}{ccc|ccc} 0.692 & 0.061 & 0.607 & 0.127 & 0.890 & 0.051 \\ 0.556 & 0.780 & 0.741 & 0.549 & 0.799 & 0.072 \\ 0.396 & 0.337 & 0.104 & 0.485 & 0.734 & 0.088 \end{array} \right).$$

TABLE I: Numerical results of $(k/k)_{e^{\mathcal{A}t}}(1)$ by using determinant formula with different k.

k	$(k/k)_{e^{\mathcal{A}t}}(1)$			
3	$(3/3)_{e^{\mathcal{A}_1 t}}(1) =$	3.7378586	2.3923452	2.0850578
		3.3171278	5.5389109	2.7445894
		2.2099211	2.6725809	2.2138140
	$(3/3)_{e^{\mathcal{A}_2 t}}(1) =$	1.8347122	3.398265	1.1709242
		3.3869955	4.3226504	1.7445016
		2.3237365	3.1926720	0.9297903
5	$(5/5)_{e^{\mathcal{A}_1 t}}(1) =$	3.6374269	1.9165248	1.5728379
		3.2657139	4.9379868	2.5381715
		1.9143609	2.4630834	2.1161953
	$(5/5)_{e^{\mathcal{A}_2 t}}(1) =$	1.9456368	3.9479852	1.2055781
		3.3374284	4.7641395	1.6892373
		2.1059645	2.8561139	1.9614427
7	$(7/7)_{e^{\mathcal{A}_1 t}}(1) =$	3.6592026	2.3405029	1.7387242
		3.3153829	5.3506815	2.5267307
		2.1217114	2.6484604	2.1923241
	$(7/7)_{e^{\mathcal{A}_2 t}}(1) =$	1.9299580	3.5269476	1.3478592
		3.3414563	4.4923726	1.8912948
		2.2381762	2.9804814	1.3265688
9	$(9/9)_{e^{\mathcal{A}_1 t}}(1) =$	3.6592464	2.3407492	1.7387693
		3.3154313	5.3509132	2.5267183
		2.1217480	2.6484855	2.1925419
	$(9/9)_{e^{\mathcal{A}_2 t}}(1) =$	1.9299684	3.5267631	1.3478380
		3.3414765	4.4922361	1.8913484
		2.2381759	2.9805069	1.3263993

In order to verify the validity of the determinant formula proposed in this article, we set an exact value $e^{\mathcal{A}} = (e^{\mathcal{A}_1}|e^{\mathcal{A}_2})$ and calculated it using MATLAB's Symbolic Math Toolbox as show below

$$e^{\mathcal{A}_1} = \begin{pmatrix} 3.6592465 & 2.3407504 & 1.7387649 \\ 3.3154318 & 5.3509139 & 2.5267149 \\ 2.1217493 & 2.6484899 & 2.1925401 \end{pmatrix},$$

$$e^{\mathcal{A}_2} = \begin{pmatrix} 1.9299693 & 3.5267634 & 1.3478442 \\ 3.3414779 & 4.4922379 & 1.8913536 \\ 2.2381769 & 2.9805050 & 1.3263996 \end{pmatrix}.$$

We utilize the determinant formula of TPTA to compute $(k/k)_{e^{\mathcal{A}t}}(1)$ with different k and report the values of $(k/k)_{e^{\mathcal{A}t}}(1)$ in Table I.

From this table, we can concluded that our algorithm is feasible and effectiveness for some higher orders, such as $k = 9$. For this case, there are already six decimal digits equal to the exact values.

VI. Conclusion

In this paper, in order to compute the tensor exponential function, we presented the error formula and determinant formula of tensor padé-type approximant (TPTA). The determinant formula is expressed as a tensor numerator divided by a scalar denominator, facilitating the computation of TPTA at any given order (m/n). This significantly broadens the applicability of the $(n-1/n)$ algorithm proposed in [12]. The performance and effectiveness of determinant formula for computing (m/n) order TPTA for tensor exponential function have been investigated by two numerical experiments. Generally speaking, the higher the order of TPTA, the better the approximation effect, but at the same time, it will also bring some instability, so determining an appropriate and stable order for TPTA is a subject of further research.

References

[1] E. A. de Souza Neto, "The exact derivative of the exponential of an unsymmetric tensor," *Comput. Methods Appl. Mech. Engrg.,* vol. 190, no. 18-19, pp. 2377-2383, 2001.

[2] A. Cuitino and M. Ortiz, "A material-independent method for extending stress update algorithms from small-strain plasticity to finite plasticity with multiplicative kinematics," *Eng. Comput.,* vol. 9, no. 4, pp. 437-451, 1992.

[3] Y. A. Hidayat, Sutandi, and E. Joelianto, "Integrated supply chain model of distribution regionalization and network optimization of white crystal sugar traditional market," *Engineering Letters,* vol. 29, no. 1, pp. 302-311, 2021.

[4] S. Dolgov and B. Khoromskij, "Simultaneousstate-time approximation of the chemical master equation using tensor product formats," *Numer. Linear Algebra Appl.,* vol. 22, no. 2, pp. 197-219, 2015.

[5] P. Gelß, S. Matera, and C. Schütte, "Solving the master equation without kinetic Monte Carlo: tensor train approximations for a CO oxidation model," *J. Comput. Phys.,* vol. 314, pp. 489-502, 2016.

[6] J. Lu, "Exact expansions of arbitrary tensor functions $F(A)$ and their derivatives," *Int. J. Solids Struct.,* vol. 41, no. 15, pp. 4261-4262, 2004.

[7] Y. Wang, K. Zhang, and H. Sun, "Criteria for strong H-tensors," *Front. Math. China,* vol. 11, no. 3, pp. 577-592, 2016.

[8] G. Wang, G. Zhou, and L. Caccetta, "Z-eigencvalue inclusion theorems for tensors," *Discrete Contin. Dyn. Syst. Ser. B,* vol. 22, no. 1, pp. 187-198, 2017.

[9] B. Medjahdi and H. Zoubir, "Affine factorable surfaces in the three-dimensional simply isotropic space," *IAENG International Journal of Applied Mathematics,* vol. 51, no. 2, pp. 366-371, 2021.

[10] Z. H. Guo, "Rates of stretch tensors," *J. Elasticity,* vol. 14, no. 3, pp. 263-267, 1984.

[11] T. C. T. Ting, "Determination of $C^{1/2}$, $C^{-1/2}$ and more general isotropic tensor functions of C," *J. Elasticity,* vol. 15, no. 3, pp. 319-323, 1985.

[12] C. Q. Gu and Y, Liu, "Tensor Padé-type approximant with application in computing tensor exponential function," *J. Funct. Spaces,* vol. 2018, Article ID 2835175, 10 pages, 2018.

[13] C. Q. Gu, "Matrix padé-type approximant and directional matrix padé approximant in the inner product space," *J. Comput. Appl. Math.,* vol. 164, no. 1, pp. 365-385, 2004.

[14] I. Mikhail, "Computation of the exponential and other isotropic tensor functions and their derivatives," *Comput. Methods Appl. Mech. Engrg.,* vol. 192, no. 35-36, pp. 3985-3999, 2003.

[15] C. D. Martin, S. Richard, and L. Betsy, "An order-p tensor factorization with applications in imaging," *SIAM J. Sci. Comput.,* vol. 35, no. 1, pp. 474-490, 2013.

[16] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover, "Tirdorder tensors as operators on matrices: a theoretical and computational frameworkwith applicationsin imaging," *SIAM J. Matrix Anal. Appl.,* vol. 34, no. 1, pp. 148-172, 2013.

[17] B. Soukaina and S. Nadir, "Tensor product and linear error block codes," *IAENG International Journal of Applied Mathematics,* vol. 51, no. 2, pp. 279-283, 2021

[18] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.,* vol. 51, no. 3, pp. 455-500, 2009.

[19] A. L. Tampos, J. E. C. Lope, and J. S. Hesthaven, "Accurate reconstruction of discontinuous functions using the singular Padé-Chebyshev method," *IAENG International Journal of Applied Mathematics,* vol. 42, no. 4, pp. 242-249, 2012.

[20] C. Brezinski, "From numerical quadrature to Padé approximation," Elsevier Science Publishers, 2010.