Journal of Computational and Applied Mathematics 273 (2014) 29–40

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Taylor-type 1-step-ahead numerical differentiation rule for first-order derivative approximation and ZNN discretization

Yunong Zhang *, Long Jin, Dongsheng Guo, Yonghua Yin, Yao Chou

School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, PR China

HIGHLIGHTS

- A formula is proposed to approximate the first-order derivative.
- An optimal step length rule for the proposed formula is investigated.
- A Taylor-type ZNN model is derived for time-varying matrix inversion.

ARTICLE INFO

Article history:
Received 2 August 2013
Received in revised form 19 April 2014

Keywords:
First-order derivative
Numerical differentiation
ZNN discretization
Computational precision
Error analysis

ABSTRACT

In order to achieve higher computational precision in approximating the first-order derivative and discretize more effectively the continuous-time Zhang neural network (ZNN), a Taylor-type numerical differentiation rule is proposed and investigated in this paper. This rule not only greatly remedies some intrinsic weaknesses of the backward and central numerical differentiation rules, but also overcomes the limitation of the Lagrange-type numerical differentiation rules in ZNN discretization. In addition, a formula is proposed to obtain the optimal step-length of the Taylor-type numerical differentiation rule. Moreover, based on the proposed numerical differentiation rule, the stability, convergence and residual error of the Taylor-type discrete-time ZNN (DTZNN) are analyzed. Numerical experimental results further substantiate the efficacy and advantages of the proposed Taylor-type numerical differentiation rule for first-order derivative approximation and ZNN discretization.

1. Introduction

Numerical differentiation, which describes methods/algorithms for estimating the derivative of a mathematical function, is widely used to solve ordinary differential equations (ODEs) and partial differential equations (PDEs) in numerical analysis and engineering applications [1–4]. Various methods/algorithms have been presented and employed for first-order derivative approximation, such as the polynomial interpolation method [5], the finite difference method [6], the regularization method [7], the method of undetermined coefficients [8] and the method of Lagrange-type 1-step-ahead numerical differentiation [9]. In [10], Mboup et al. derived a numerical differentiation explicit rule yielding a pointwise derivative estimation for each given order. Ramm et al. presented a new approach to the construction of finite-difference methods and they showed how the multi-point differentiators can generate regularizing algorithms with a stepsize being a regularization parameter in [11]. However, considering the following four situations: (1) that the backward differentiation rules may not adapt to the fast variational rate of the first-order derivative of target point, (2) that the central differentiation

* Corresponding author. Tel.: +86 13060687155.
E-mail addresses: ynzhang@ieee.org, zhynong@mail.sysu.edu.cn, jallonzyn@sina.com (Y. Zhang).

http://dx.doi.org/10.1016/j.cam.2014.05.027
0377-0427/

rules cannot approximate the first-order derivative of the target points without enough number of data point in either side, (3) that the numerical differentiation is urgently needed to be inversely applicated to the discretization of continuous-time Zhang neural network (ZNN) [12] which is exploited for the online solution of time-varying problems, and (4) that a numerical differentiation rule does not necessarily generate a convergent and stable discrete-time ZNN (DTZNN) model (not to mention a high-precision one), a new kind of effective 1-step-ahead numerical differentiation rule is needed.

As mentioned above, a numerical differentiation rule for ZNN discretization does not necessarily generate a convergent and stable discrete-time method/algorithm. In view of this point, the performances of the counterpart DTZNN models, which are generated by Lagrange-type 1-step-ahead numerical differentiation rules [9] and Euler forward-difference rule, are analyzed in this paper. Moreover, a Taylor-type numerical differentiation rule is elaborately constructed for first-order derivative approximation. Then, the optimal step length is analyzed to minimize the total error of the Taylor-type numerical differentiation rule in the numerical computations. Furthermore, the Taylor-type numerical differentiation rule is adopted to discretize the continuous-time ZNN model. Both theoretical analysis and numerical experimental results demonstrate the efficacy and advantages of the proposed Taylor-type DTZNN model.

The rest of this paper is organized into five sections. Section 2 introduces the ZNN models for time-varying matrix inversion and proves the convergence and stability of the Euler-type DTZNN model as well as Newton iteration. In Section 3, the Taylor-type numerical differentiation rule is proposed, and the corresponding total error and optimal step length are also investigated. By applying the Taylor-type numerical differentiation rule to the discretization of continuous-time ZNN model, the Taylor-type DTZNN model is generated in Section 4, which is proved to be stable and convergent with $O(\tau^3)$ error pattern. Section 5 illustrates the numerical experimental results of a time-varying matrix inversion example, which are synthesized by Newton iteration, Euler-type and Taylor-type DTZNN models. Section 6 concludes this paper with final remarks. Before ending this introductory section, the main contributions of this paper lie in the following facts.

(1) This paper proposes and further investigates a Taylor-type numerical differentiation rule to approximate the first-order derivative of the target point, thereby remedying the intrinsic weaknesses of the backward and central differentiation rules.
(2) The stability and convergence of Euler-type DTZNN model and Newton iteration for time-varying matrix inversion are proved theoretically for the first time in this paper. Besides, it is also proved that the Lagrange-type DTZNN models using four to eight data points are neither stable nor convergent.
(3) The total error and optimal step length of the proposed Taylor-type numerical differentiation rule is further investigated in this paper, and the corresponding numerical experimental result demonstrates the efficacy of the optimal step-length formula.
(4) Based on the proposed Taylor-type numerical differentiation rule, a Taylor-type DTZNN model with $O(\tau^3)$ error pattern is derived for the first time, of which the stability and convergence are proved theoretically. In addition, numerical experimental results are provided to substantiate the superiority of the Taylor-type DTZNN model.

2. ZNN and its discretization

Zhang neural network, which is a special class of recurrent neural networks, has been proposed for the online solution of various time-varying problems [13,14]. In this paper, we take time-varying matrix inversion as an example. For possible digital hardware (e.g., digital circuit or digital computer) realization, a continuous-time ZNN model is required to be discretized into the counterpart DTZNN models. Thus, how to discretize a continuous-time ZNN in a computational inexpensive manner for a higher accuracy is a goal pursued by researchers.

2.1. Continuous-time ZNN models

Let us consider the following time-varying matrix inversion problem:

$$A(t)X(t) = I, \tag{1}$$

where $A(t) \in R^{n\times n}$ is a smoothly time-varying nonsingular matrix, $I \in R^{n\times n}$ is the identity matrix, and $X(t) \in R^{n\times n}$ is the unknown matrix to be obtained. Let $X^*(t) \in R^{n\times n}$ denote the time-varying theoretical solution of (1). To monitor and control the solving process of (1), we define the following matrix-valued indefinite error function:

$$E(t) = A(t)X(t) - I \in R^{n\times n}. \tag{2}$$

Then, by adopting the ZNN design formula $\dot{E}(t) = -\gamma E(t)$, the implicit-dynamics continuous-time ZNN model is given directly as [14]

$$A(t)\dot{X}(t) = -\dot{A}(t)X(t) - \gamma\big(A(t)X(t) - I\big), \tag{3}$$

where the design parameter $\gamma > 0 \in R$, $\dot{X}(t)$ and $\dot{A}(t)$ denote the time-derivatives of $X(t)$ and $A(t)$, respectively. In view of $A(t)$ being nonsingular at any time instant t and state matrix $X(t)$ of (3) converging to $A^{-1}(t)$, ZNN (3) can be reformulated as follows [14]:

$$\dot{X}(t) = -X(t)\dot{A}(t)X(t) - \gamma X(t)\big(A(t)X(t) - I\big), \tag{4}$$

which is the explicit-dynamics continuous-time ZNN model for time-varying matrix inversion.

Table 1
Lagrange-type 1-step-ahead differentiation rules using multiple data points.

Data points	1-step-ahead rule	Truncation error
4	$f'(x_2) \approx (2f(x_3) + 3f(x_2) - 6f(x_1) + f(x_0))/(6\tau)$	$O(\tau^3)$
5	$f'(x_3) \approx (3f(x_4) + 10f(x_3) - 18f(x_2) + 6f(x_1) - f(x_0))/(12\tau)$	$O(\tau^4)$
6	$f'(x_4) \approx (12f(x_5) + 65f(x_4) - 120f(x_3) + 60f(x_2) - 20f(x_1) + 3f(x_0))/(60\tau)$	$O(\tau^5)$
7	$f'(x_5) \approx (10f(x_6) + 77f(x_5) - 150f(x_4) + 100f(x_3) - 50f(x_2) + 15f(x_1) - 2f(x_0))/(60\tau)$	$O(\tau^6)$
8	$f'(x_6) \approx (60f(x_7) + 609f(x_6) - 1260f(x_5) + 1050f(x_4) - 700f(x_3) + 315f(x_2) - 84f(x_1) + 10f(x_0))/(420\tau)$	$O(\tau^7)$

Table 2
Multiple-point Lagrange-type DTZNN models.

L4-DTZNN	$X_{k+1} = -3/2X_k + 3X_{k-1} - 1/2X_{k-2} - 3\tau X_k\dot{A}_kX_k - 3hX_k(A_kX_k - I)$
L5-DTZNN	$X_{k+1} = -10/3X_k + 6X_{k-1} - 2X_{k-2} + 1/3X_{k-3} - 4\tau X_k\dot{A}_kX_k - 4hX_k(A_kX_k - I)$
L6-DTZNN	$X_{k+1} = -65/12X_k + 10X_{k-1} - 5X_{k-2} + 5/3X_{k-3} - 1/4X_{k-4} - 5\tau X_k\dot{A}_kX_k - 5hX_k(A_kX_k - I)$
L7-DTZNN	$X_{k+1} = -77/10X_k + 15X_{k-1} - 10X_{k-2} + 5X_{k-3} - 3/2X_{k-4} + 1/5X_{k-5} - 6\tau X_k\dot{A}_kX_k - 6hX_k(A_kX_k - I)$
L8-DTZNN	$X_{k+1} = -203/20X_k + 21X_{k-1} - 35/2X_{k-2} + 35/3X_{k-3} - 21/4X_{k-4} + 7/5X_{k-5} - 1/6X_{k-6} - 7\tau X_k\dot{A}_kX_k - 7hX_k(A_kX_k - I)$

2.2. Lagrange-type numerical differentiation rules

To remedy intrinsic weaknesses of the backward and central numerical differentiation rules, Zhang et al. [9] proposed a special kind of Lagrange-type 1-step-ahead numerical differentiation for approximating the first-order derivative of the target point. For comparison and also for reader's convenience, the differentiation rules using four to eight data points are directly given in Table 1. These rules not only achieve higher computational precision but also greatly remedy some intrinsic weaknesses of the central and backward numerical differentiation rules [9]. In view of the advantages of Lagrange-type 1-step-ahead differentiation rules for first-order derivative approximation, we attempt to apply these rules to the ZNN discretization for achieving a higher computational precision in the ensuing subsections.

2.3. DTZNN models

Before presenting the DTZNN models, the unique characteristics of a method for time-varying problem solving, which are different from those of static (time-invariant) problem solving, are provided as follows.

(1) Computation is performed based on the present and/or previous data but for future use. That is, the solution X_{k+1} to the time-varying problem at the time instant t_{k+1} should be calculated before the time instant t_{k+1}, and, at the time instant t_{k+1}, the system uses the value X_{k+1} which has been calculated. So, how to develop the discrete-time model without using future data is the key point for the time-varying problem solving. In other words, the discrete-time model for the time-varying problem solving should be causal.
(2) Computation consumes time inevitably at every single time instant. In the widely-used numerical discretization methods (e.g., Runge–Kutta methods [3,15]), at every single computational time instant, many precalculations have to be done before applying the final updating formula. But time is precious for the time-varying problem solving in practice. So, how to design a relatively simple discrete-time model with less calculation time is important. In other words, the discrete-time model for the time-varying problem solving should satisfy the requirement of real-time computation.

In order to discretize the continuous-time ZNN model (4), we usually refer to the Euler forward-difference rule. From [14], the Euler-type DTZNN model is directly given as

$$X_{k+1} = X_k - \tau X_k\dot{A}_kX_k - hX_k(A_kX_k - I), \tag{5}$$

where $\tau > 0 \in R$ denotes the sampling gap, k denotes the updating index (with $k = 0, 1, 2, \ldots$) and $h = \tau\gamma > 0$. In general, we denote $X_k = X(t = k\tau)$. In addition, $A(t)$ and $\dot{A}(t)$ are discretized by the standard sampling method, of which the sampling gap is also $\tau = t_{k+1} - t_k$. For convenience and also for consistency with X_k, we use A_k standing for $A(t = k\tau)$ and $\dot{A}_k$ standing for $\dot{A}(t = k\tau)$.

It is worth noting that, with term $\tau X_k\dot{A}_kX_k$ being omitted and $h = 1$, the Euler-type DTZNN model (5) reduces to Newton iteration as follows:

$$X_{k+1} = X_k - X_k(A_kX_k - I). \tag{6}$$

Thus, Newton iteration (6) can be viewed as a special case of the Euler-type DTZNN model (5) for time-varying matrix inversion.

Based on the standard sampling method, by employing Lagrange-type 1-step-ahead numerical differentiation rule listed in Table 1, the corresponding Lagrange-type DTZNN models for solving time-varying matrix inversion problem (1) are directly deduced in Table 2.

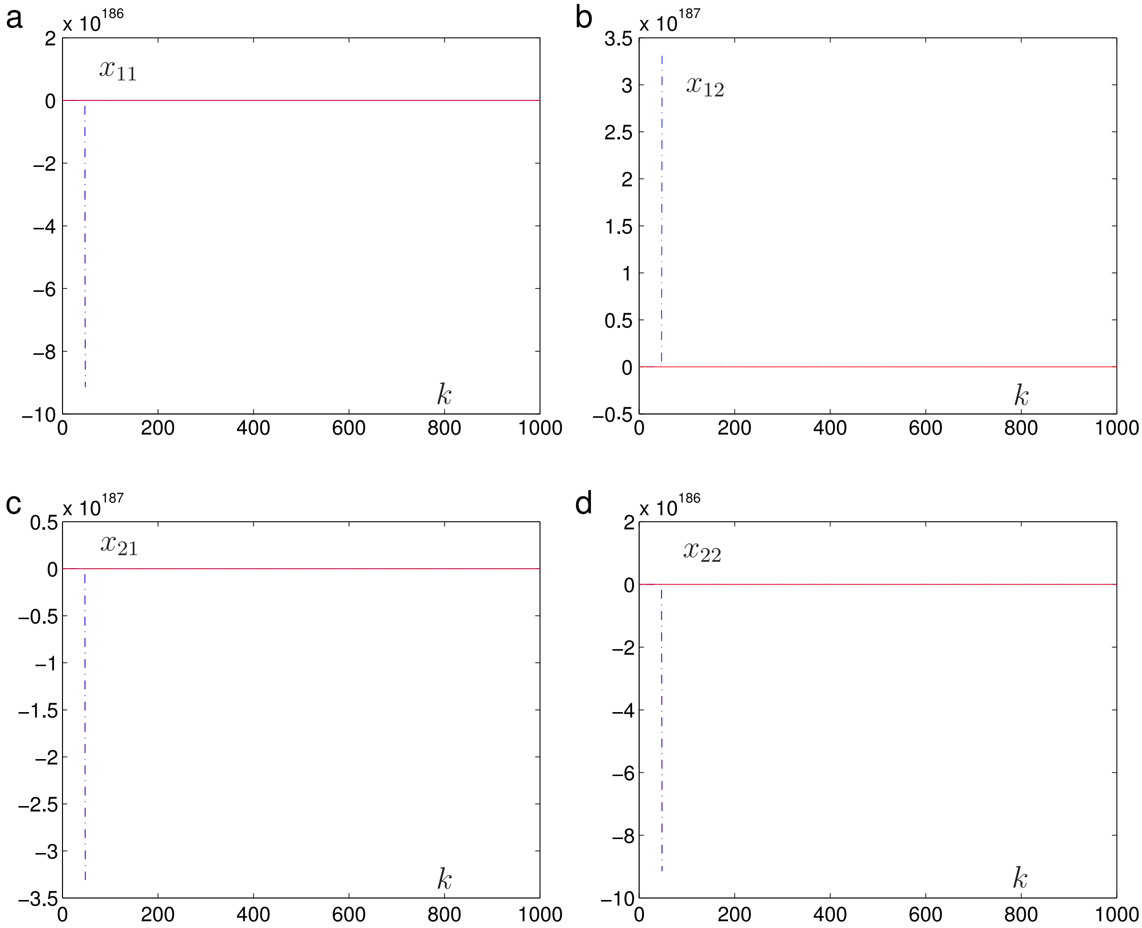

Fig. 1. Divergence phenomenon arising in solving process of using L4-DTZNN model with $h = 0.3$, $\tau = 0.01$ and initial state being $[0, -1; 1, 0] \in R^{2\times 2}$, where the solid curves correspond to the theoretical solution and the dash-dotted curves correspond to the L4-DTZNN solution. (a) x_{11}-profile. (b) x_{12}-profile. (c) x_{21}-profile. (d) x_{22}-profile.

Remark 1. During the solving process of time-varying matrix inversion problem (1), computation has to be performed based on the present and/or previous data, e.g., at time instant t_k, we can use known information (e.g., X_k, X_{k-1}, A_k and $\dot{A}_k$) instead of unknown information (e.g., A_{k+1} and $\dot{A}_{k+1}$) for computing X_{k+1}. Thus, the objective of this ZNN discretization is, through the present and/or previous data, to find the unknown matrix X_{k+1} during $[t_k, t_{k+1})$, such that (1) holds true at each time instant. Note that, at the start of the computation (i.e., at time instant $t_0 = 0$), we cannot compute $X(t_0)$ based on previous data; thus X_0 can be randomly generated or directly set as X_0^* which is computed in advance for achieving better performance, where X_0^* is the theoretical solution of (1) at time instant $t_0 = 0$.

2.4. Divergence phenomenon

In this subsection, numerical experiments are performed to investigate the performance of the presented Lagrange-type DTZNN models shown in Table 2 using four to eight data points for time-varying matrix inversion. For illustration, we use the following time-varying matrix:

$$A(t) = \begin{bmatrix} \sin(t) & \cos(t) \\ -\cos(t) & \sin(t) \end{bmatrix} \in R^{2\times 2}. \tag{7}$$

The time-varying theoretical inverse of (7) is

$$X^*(t) = A^{-1}(t) = \begin{bmatrix} \sin(t) & -\cos(t) \\ \cos(t) & \sin(t) \end{bmatrix} \in R^{2\times 2},$$

which is given for checking the correctness of the DTZNN solutions.

As illustrated in Fig. 1, the divergence phenomenon occurs in the solving process, which means that the solution generated by the L4-DTZNN model cannot converge to the theoretical solution. Note that, the similar divergence phenomena occur in the solving process of using other Lagrange-type DTZNN models for time-varying matrix inversion of (7). By contrast, from [14], we know that Euler-type DTZNN model (5) is effective for the time-varying matrix inversion of (7). These phenomena inspire us to theoretically investigate the convergence and stability of the presented DTZNN models.

2.5. Theoretical analyses and explanations

Before investigating the performance of Euler- and Lagrange-type DTZNN models, the following preliminaries, i.e., Definitions 1–3, are provided to lay a basis for further discussion [16].

Definition 1. An N step method $\sum_{j=0}^{N}\alpha_j x_{k+j} = \tau\sum_{j=0}^{N}\beta_j f_{k+j}$ can be checked 0-stability by determining the roots of the characteristic polynomial $P_N(\theta) = \sum_{j=0}^{N}\alpha_j\theta^j$. If the roots of $P_N(\theta) = 0$ are such that

- all roots lie in the unit disk, i.e., $|\theta| \le 1$; and,
- any root on the unit circle (i.e., $|\theta| = 1$) is simple (i.e., not multiple);

then the N step method is 0-stable. In addition, the 0-stability is sometimes called Dahlquist stability or root stability.

Definition 2. An N step method is said to be consistent of order p if its truncation error is $O(\tau^p)$ with $p > 0$ for the smooth exact solution.

Definition 3. An N step method is convergent, i.e., $x_{[(t-t_0)/\tau]} \to x^*(t)$, for all $t \in [t_0, t_{\text{final}}]$, as $\tau \to 0$, if and only if the method is consistent and 0-stable. In other words, consistency plus 0-stability leads to convergence. In particular, a 0-stable consistent method converges with the order of its truncation error.

Based on the above three definitions, we have the following theoretical results about the Euler-type DTZNN model (5).

Theorem 1. *Euler-type DTZNN model (5) is 0-stable.*

Proof. According to the characteristic polynomial in Definition 1, the characteristic polynomial of Euler-type DTZNN model (5) can be derived as

$$P_1(\theta) = \theta - 1 = 0,$$

which has only one root (i.e., $\theta = 1$) on the unit disk. Therefore, the Euler-type DTZNN model (5) is 0-stable. The proof is thus completed. □

Theorem 2. *Euler-type DTZNN model (5) is consistent and convergent, which converges with the order of its truncation error being $\mathbf{O}(\tau^2)$ for all $t_k \in [t_0, t_{\text{final}}]$, where $\mathbf{O}(\tau^2)$ denotes a matrix with every element being $O(\tau^2)$.*

Proof. In view of the Euler forward-difference rule, we have

$$\dot{X}_k = \frac{X_{k+1} - X_k}{\tau} + \mathbf{O}(\tau). \tag{8}$$

By exploiting (8) to discretize (4), we can get the following equation:

$$X_{k+1} = X_k - \tau X_k\dot{A}_k X_k - hX_k(AX_k - I) + \mathbf{O}(\tau^2). \tag{9}$$

Note that, dropping $\mathbf{O}(\tau^2)$ of (9) yields exactly Euler-type DTZNN model (5), and thus the truncation error of Euler-type DTZNN model (5) is $\mathbf{O}(\tau^2)$. Therefore, Euler-type DTZNN model (5) is both 0-stable and consistent. According to Definitions 2 and 3, it can be concluded that Euler-type DTZNN model (5) is consistent and convergent, which converges with the order of its truncation error being $\mathbf{O}(\tau^2)$ for all $t_k \in [t_0, t_{\text{final}}]$. The proof is thus completed. □

Theorem 3. *Consider time-varying matrix inversion problem (1). The steady-state residual error $\lim_{k\to\infty}\|A_kX_k - I\|_\text{F}$ of the Euler-type DTZNN model (5) is $O(\tau^2)$, where symbol $\|\cdot\|_\text{F}$ denotes the Frobenius norm of a matrix.*

Proof. In view of Definition 3, Theorems 1 and 2, it can be concluded that $X_k^* + \mathbf{O}(\tau^2) = X_k$ with k large enough. Then, we can obtain

$$\|A_kX_k - I\|_\text{F} = \|A_k(X_k^* + \mathbf{O}(\tau^2)) - I\|_\text{F} = \|A_kX_k^* - I + A_k\mathbf{O}(\tau^2)\|_\text{F}.$$

Thus, we can further have

$$\|A_kX_k - I\|_\text{F} = \|A_k\mathbf{O}(\tau^2)\|_\text{F} = O(\tau^2),$$

which now completes the proof. □

Proposition 1. *The* L4-DTZNN *model is not convergent.*

Table 3
Root outside the unit disk of the characteristic polynomial corresponding to the Lagrange-type DTZNN models listed in Table 2.

Data points	4	5	6	7	8
Root	−2.6861	−4.6979	−6.9614	−9.4127	−12.0244

Proof. According to the characteristic polynomial in Definition 1, the characteristic polynomial of the L4-DTZNN model can be derived as

$$P_1(\theta) = \theta^3 + 1.5\theta^2 - 3\theta + 0.5 = 0,$$

which has three roots [i.e., $\theta_1 = 1$, $\theta_2 = (-2.5 + \sqrt{8.25})/2 \approx 0.1861$ and $\theta_3 = (-2.5 - \sqrt{8.25})/2 \approx -2.6861$]. Evidently, θ_3 is outside the unit disk. According to Definition 1, the L4-DTZNN model is not guaranteed to be 0-stable. Therefore, the L4-DTZNN model is not convergent. The proof is thus completed. □

Besides, for completeness, Table 3 shows the root outside the unit disk of the characteristic polynomial corresponding to the L5-, L6-, L7- and L8-DTZNN models. From Table 3, we can have the conclusion that the L5-, L6-, L7- and L8-DTZNN models (listed in Table 2) are not convergent. In other words, the Lagrange-type numerical differentiation rules using four to eight data points cannot be successfully applied to the ZNN discretization.

Remark 2. By following Theorems 1–3, it can be similarly proved that Newton iteration (6) is consistent and convergent, which converges with the order of truncation error $\mathbf{O}(\tau)$ and with the steady-state residual error $\lim_{k\to\infty} \|A_k X_k - I\|_F$ being $O(\tau)$.

3. Taylor-type numerical differentiation

In order to achieve higher computational precision in approximating the first-order derivative of the target point and be applied to ZNN discretization successfully, a Taylor-type numerical differentiation rule is proposed in this section. This rule not only greatly remedies some intrinsic weaknesses of the backward and central numerical differentiation rules, but also overcomes the limitation of the Lagrange-type numerical differentiation rules in ZNN discretization.

3.1. New effective $O(\tau^2)$ rule

A Taylor series is an infinite sum of terms that are calculated from the values of derivatives of a function at a single point. In scientific and engineering fields, the partial sums can be accumulated until an approximation to the function is obtained that achieves the specified accuracy. Therefore, a Taylor type numerical differentiation rule is constructed in this subsection for first-order derivative approximation by eliminating the second-order derivative, which can achieve higher computational precision in the application of ZNN discretization. The ensuing theorem and proof are presented for the construction of the Taylor-type 1-step-ahead numerical differentiation rule.

Theorem 4. *The Taylor-type* 1*-step-ahead numerical differentiation rule is formulated as* $f'(x_2) \approx (2f(x_3) - 3f(x_2) + 2f(x_1) - f(x_0))/2\tau$, *which has a truncation error of* $O(\tau^2)$.

Proof. Based on the Taylor expansion [17], we have the following rule:

$$f(x_3) = f(x_2 + \tau) = f(x_2) + \tau f'(x_2) + \frac{\tau^2}{2!} f^{(2)}(x_2) + \frac{\tau^3}{3!} f^{(3)}(c_1), \tag{10}$$

where c_1 lies between x_2 and x_3. Then, we rewrite (10) as

$$f'(x_2) = \frac{f(x_3) - f(x_2)}{\tau} - \frac{\tau}{2!} f^{(2)}(x_2) - \frac{\tau^2}{3!} f^{(3)}(c_1). \tag{11}$$

Similarly, one can obtain

$$f(x_1) = f(x_2 - \tau) = f(x_2) - \tau f'(x_2) + \frac{\tau^2}{2!} f^{(2)}(x_2) - \frac{\tau^3}{3!} f^{(3)}(c_2),$$

and

$$f(x_0) = f(x_2 - 2\tau) = f(x_2) - 2\tau f'(x_2) + \frac{4\tau^2}{2!} f^{(2)}(x_2) - \frac{8\tau^3}{3!} f^{(3)}(c_3),$$

with c_2 and c_3 lying in interval (x_1, x_2) and interval (x_0, x_2), respectively. Then, the above two rules can be rewritten as

$$f'(x_2) = \frac{f(x_2) - f(x_1)}{\tau} + \frac{\tau}{2!} f^{(2)}(x_2) - \frac{\tau^2}{3!} f^{(3)}(c_2), \tag{12}$$

and

$$f'(x_2) = \frac{f(x_2) - f(x_0)}{2\tau} + \tau f^{(2)}(x_2) - \frac{2\tau^2}{3} f^{(3)}(c_3). \tag{13}$$

Let (11) plus (13), then subtract (12). Thus, we develop the following new 1-step-ahead difference rule based on Taylor series:

$$f'(x_2) = \frac{2f(x_3) - 3f(x_2) + 2f(x_1) - f(x_0)}{2\tau} + \tau^2 \left(-\frac{1}{3!} f^{(3)}(c_1) - \frac{2}{3} f^{(3)}(c_3) + \frac{1}{3!} f^{(3)}(c_2) \right),$$

which can be further rewritten as

$$f'(x_2) = \frac{2f(x_3) - 3f(x_2) + 2f(x_1) - f(x_0)}{2\tau} + O(\tau^2). \tag{14}$$

Dropping $O(\tau^2)$ of (14) yields exactly Taylor-type 1-step-ahead numerical differentiation rule as below:

$$f'(x_2) \approx \frac{2f(x_3) - 3f(x_2) + 2f(x_1) - f(x_0)}{2\tau}. \tag{15}$$

The proof is thus completed. □

Based on Theorem 4, a new effective $O(\tau^2)$ rule is obtained for first-order derivative approximation, which is expected to be applied to ZNN discretization for a higher accuracy in comparison with the Euler-type DTZNN model (which is investigated in the ensuing section).

3.2. Optimal step-length formula

It is well known that the step length τ brings about a great influence on the total error. In general, a smaller step length τ theoretically leads to a smaller total error. However, due to the existence of round-off errors in the numerical computations, a smaller step length τ does not necessarily generate a smaller total error. Thus, the optimal step length is generally expected for obtaining a relatively higher accuracy in practical applications. In this subsection, for the proposed Taylor-type 1-step-ahead numerical differentiation rule (15), we investigate the corresponding optimal step length via the following theorem.

Theorem 5. *The optimal step-length formula of the proposed Taylor-type* 1*-step-ahead numerical differentiation rule* (15) *is* $(2\varepsilon/M)^{1/3}$*, where* ε *denotes the maximum absolute value of round-off errors of* $f(x_m)$ *(with* $m = 0, 1, 2, 3$*) in the numerical computations, and* M *denotes the maximum absolute value of* $f^{(3)}(c_i)$ *(with* $i = 1, 2, 3$*).*

Proof. The round-off errors in the numerical computations can be simplified as the following equations:

$$\begin{aligned} f(x_3) &= y_3 + \varepsilon_3, \\ f(x_2) &= y_2 + \varepsilon_2, \\ f(x_1) &= y_1 + \varepsilon_1, \\ f(x_0) &= y_0 + \varepsilon_0, \end{aligned}$$

where $f(x_3), f(x_2), f(x_1)$ and $f(x_0)$ are approximated by numerical values y_3, y_2, y_1 and y_0, respectively. In addition, $\varepsilon_3, \varepsilon_2, \varepsilon_1$ and ε_0 are the corresponding round-off errors.

According to the Taylor-type 1-step-ahead numerical differentiation rule (15), we can obtain

$$f'(x_2) = \frac{2y_3 - 3y_2 + 2y_1 - y_0}{2\tau} + E(f, \tau),$$

in which

$$E(f, \tau) = \frac{2\varepsilon_3 - 3\varepsilon_2 + 2\varepsilon_1 - \varepsilon_0}{2\tau} - \left(\frac{\tau^2 f^{(3)}(c_1)}{6} - \frac{\tau^2 f^{(3)}(c_2)}{6} + \frac{4\tau^2 f^{(3)}(c_3)}{6} \right).$$

Evidently, the total error term $E(f, \tau)$ contains two parts, i.e., a part due to round-off errors and a part due to truncation errors.

In view of ε denoting the maximum absolute value of round-off errors of $f(x_m)$ (with $m = 0, 1, 2, 3$) and M denoting the maximum absolute value of $f^{(3)}(c_i)$ (with $i = 1, 2, 3$), we have

$$|E(f, \tau)| \leq \frac{4\varepsilon}{\tau} + M\tau^2. \tag{16}$$

Table 4
Errors in approximation to $f'(x) = -\sin(x)$ using Taylor-type 1-step-ahead numerical differentiation rule (15) with different values of step length τ.

τ	Error	τ	Error
7×10^{-1}	3.3119×10^{-4}	7×10^{-4}	8.3674×10^{-8}
3×10^{-1}	8.8291×10^{-3}	3×10^{-4}	1.6770×10^{-8}
1×10^{-1}	1.4809×10^{-3}	1×10^{-4}	5.1036×10^{-9}
7×10^{-2}	7.6189×10^{-4}	7×10^{-5}	2.2474×10^{-9}
3×10^{-2}	1.4874×10^{-4}	3×10^{-5}	1.1770×10^{-8}
1×10^{-2}	1.7013×10^{-5}	1×10^{-5}	5.4893×10^{-8}
7×10^{-3}	8.3719×10^{-6}	7×10^{-6}	4.5125×10^{-8}
3×10^{-3}	1.5463×10^{-6}	3×10^{-6}	2.1179×10^{-7}
1×10^{-3}	1.7210×10^{-7}	1×10^{-6}	1.0452×10^{-6}

Thus, the value of τ that minimizes the right-hand side of formula (16) is

$$\tau = \left(\frac{2\varepsilon}{M}\right)^{1/3}, \tag{17}$$

which now completes the proof. □

3.3. Numerical verification

To analyze the total errors of Taylor-type 1-step-ahead numerical differentiation rule (15) and substantiate the efficacy of the optimal step-length formula (17), we exploit the following function as the target function:

$$f(x) = \cos(x).$$

We use $x = \pi/12$ as the target point in the following numerical experiment. Additionally, eighteen different values of step length τ are investigated to verify the efficacy of the Taylor-type 1-step-ahead numerical differentiation rule. Besides, in the numerical experiment, $\varepsilon = 5 \times 10^{-13}$. Note that, for the target function $f(x) = \cos(x)$, $|f^{(3)}(x)| = |\sin(x)| \leq 1 = M$ (with symbol $|\cdot|$ denoting the absolute value of a scalar), then, from (17), we can get the optimal step length $\tau = 1 \times 10^{-4}$ in this example. The errors between the first-order derivative of target-function values and the first-order derivative of the approximation values via (15) at the target point are presented in Table 4. As observed from Table 4, the minimal error in approximation to $f'(x) = -\sin(x)$ using (15) occurs approximately at the step length $\tau = 7 \times 10^{-5}$ or $\tau = 1 \times 10^{-4}$. These results substantiate that the above optimal step-length formula (17) for the Taylor-type 1-step-ahead numerical differentiation rule is effective and accurate.

Remark 3. The noise corruption is an important issue for the derivative estimation in the numerical analysis, of which the sources can be classified into two categories: (1) the bias term stemming from the truncation errors of the Taylor expansion; (2) the variance term generated from the noisy environment [10]. As shown in Theorem 5, the total error term $E(f, \tau)$ of Taylor-type 1-step-ahead numerical differentiation rule (15) contains two parts, i.e., the part due to round-off errors corresponding to the variance term, and the part due to truncation errors corresponding to the bias term. The Gaussian distribution, being a commonly occurring probability distribution in statistics, is often used in the natural and social sciences for random variables whose distributions are not known. Therefore, the zero-mean Gaussian white noise is a typical noise for us to investigate. Three-sigma (3σ) rule in statistics indicates that nearly all (99.73%) of the values lie within three standard deviations of the mean in a Gaussian distribution [18]. Thus, for the zero-mean Gaussian white noises, it is reasonable to assume that all values of the noises lie within $[-3\sigma, 3\sigma]$. Then the optimal step length for the Taylor-type 1-step-ahead numerical differentiation rule (15) influenced by zero-mean Gaussian white noises can be readily derived as $\tau = ((2\varepsilon + 6\sigma)/M)^{1/3}$.

4. Taylor-type ZNN discretization

In order to achieve a higher accuracy in ZNN discretization, a new DTZNN model based on the proposed Taylor-type numerical differentiation rule (15) is developed and investigated in this section.

By following the similar steps given in Section 2.3, a new Taylor-type DTZNN model can be developed for time-varying matrix inversion as

$$X_{k+1} = -\tau X_k \dot{A}_k X_k - hX_k(A_k X_k - I) + \frac{3}{2}X_k - X_{k-1} + \frac{1}{2}X_{k-2}. \tag{18}$$

For such a Taylor-type DTZNN model (18), we have the following theorems to guarantee its stable and convergent properties.

Theorem 6. *Taylor-type DTZNN model (18) is 0-stable.*

Table 5
Comparisons among three discrete-time models in terms of MSSRE.

τ	Model	MSSRE	τ	Model	MSSRE
0.1	NI (6)	0.1414	0.05	NI (6)	7.0703×10^{-2}
	EZ (5)	0.0224		EZ (5)	5.8127×10^{-3}
	TZ (18)	2.9783×10^{-3}		TZ (18)	3.8743×10^{-4}
0.01	NI (6)	1.4142×10^{-2}	0.005	NI (6)	7.0710×10^{-3}
	EZ (5)	2.3557×10^{-4}		EZ (5)	5.8917×10^{-5}
	TZ (18)	3.1424×10^{-6}		TZ (18)	3.9279×10^{-7}
0.001	NI (6)	1.4142×10^{-3}	0.002	NI (6)	2.8284×10^{-3}
	EZ (5)	2.3570×10^{-6}		EZ (5)	9.4279×10^{-6}
	TZ (18)	3.1428×10^{-9}		TZ (18)	2.5144×10^{-8}

Proof. According to Definition 1, the characteristic polynomial of Taylor-type DTZNN model (18) can be derived as

$$P_1(\theta) = \theta^3 - 1.5\theta^2 + \theta - 0.5 = 0,$$

which has three roots on or in the unit disk, i.e., $\theta_1 = 1$, $\theta_2 = 0.25 + i0.6614$ and $\theta_3 = 0.25 - i0.6614$. Therefore, Taylor-type DTZNN model (18) is 0-stable. The proof is thus completed. □

Theorem 7. *Taylor-type DTZNN model (18) is consistent and convergent, which converges with the order of truncation error being $\mathbf{O}(\tau^3)$ for all $t_k \in [t_0, t_{\text{final}}]$.*

Proof. In view of (14), we have the following equation:

$$X_{k+1} = -\tau X_k \dot{A}_k X_k - hX_k(A_k X_k - I) + \frac{3}{2}X_k - X_{k-1} + \frac{1}{2}X_{k-2} + \mathbf{O}(\tau^3). \tag{19}$$

Note that, dropping $\mathbf{O}(\tau^3)$ of (19) yields exactly Taylor-type DTZNN model (18), and thus the truncation error of Taylor-type DTZNN model (18) is $\mathbf{O}(\tau^3)$. Therefore, according to Definition 2, the Taylor-type DTZNN model (18) is consistent of order 3. Together with Theorem 6, Taylor-type DTZNN model (18) is both 0-stable and consistent. Finally, in view of Definition 3, it can be concluded that Taylor-type DTZNN model (18) is consistent and convergent, which converges with the order of truncation error being $\mathbf{O}(\tau^3)$ for all $t_k \in [t_0, t_{\text{final}}]$. The proof is thus completed. □

Theorem 8. *Consider time-varying matrix inversion problem (1). The steady-state residual error $\lim_{k\to\infty} \|A_k X_k - I\|_{\text{F}}$ of Taylor-type DTZNN model (18) is $O(\tau^3)$.*

Proof. In view of Definition 3, Theorems 6 and 7, it can be concluded that $X_k^* = X_k + \mathbf{O}(\tau^3)$ with k large enough. Then, we obtain

$$\|A_k X_k - I\|_{\text{F}} = \|A_k(X_k^* + \mathbf{O}(\tau^3)) - I\|_{\text{F}} = \|A_k X_k^* - I + A_k\mathbf{O}(\tau^3)\|_{\text{F}}.$$

Consequently, we can further have

$$\|A_k X_k - I\|_{\text{F}} = \|A_k\mathbf{O}(\tau^3)\|_{\text{F}} = O(\tau^3),$$

which now completes the proof. □

5. Numerical experiments and verifications

In this section, numerical experiments are performed to demonstrate the efficacy of Taylor-type DTZNN model (18) for time-varying matrix inversion. For illustration and comparison, we reuse the time-varying matrix given in Section 2.4 [i.e., the one depicted in (7)]. The corresponding numerical experimental results are shown in Figs. 2 and 3 as well as Table 5.

As shown in Fig. 2, starting from eight randomly-generated initial states within $[-0.3, 0.3] \in R^{2\times2} + A^{-1}(0)$, neural states of Taylor-type DTZNN model (18) all converge to the theoretical solution rapidly. Note that the figures generated by Euler-type DTZNN model (5) and Newton iteration (6) are similar to Fig. 2, and are thus omitted.

To compare more overall and clearly the aforementioned three discrete-time models for time-varying matrix inversion, the corresponding residual errors are visualized in Fig. 3. It can be observed from Fig. 3(a) that, starting with an initial state (i.e., $X_0 = [0.2, -0.8; 0.8, 0.2]$), the residual errors of Taylor-type and Euler-type DTZNN models converge to zero rapidly, whereas the Newton iteration has an obvious lagging error. As illustrated in Fig. 3(b), the steady-state residual error (SSRE), i.e., $\|A_k X_k - I\|_{\text{F}}$ with k large enough, synthesized by Taylor-type DTZNN model (18) is of order 10^{-6}. In addition, the SSRE synthesized by Euler-type DTZNN model (5) is of order 10^{-4} in contrast to that synthesized by Newton iteration (6) being of order 10^{-2}. Besides, the residual errors of three discrete-time models with sampling gap τ being 0.1 and 0.001 are shown in Fig. 3(c) and (d), respectively. These comparative results substantiate the effectiveness of the proposed Taylor-type numerical differentiation rule for ZNN discretization in obtaining higher solution precision. For further investigation,

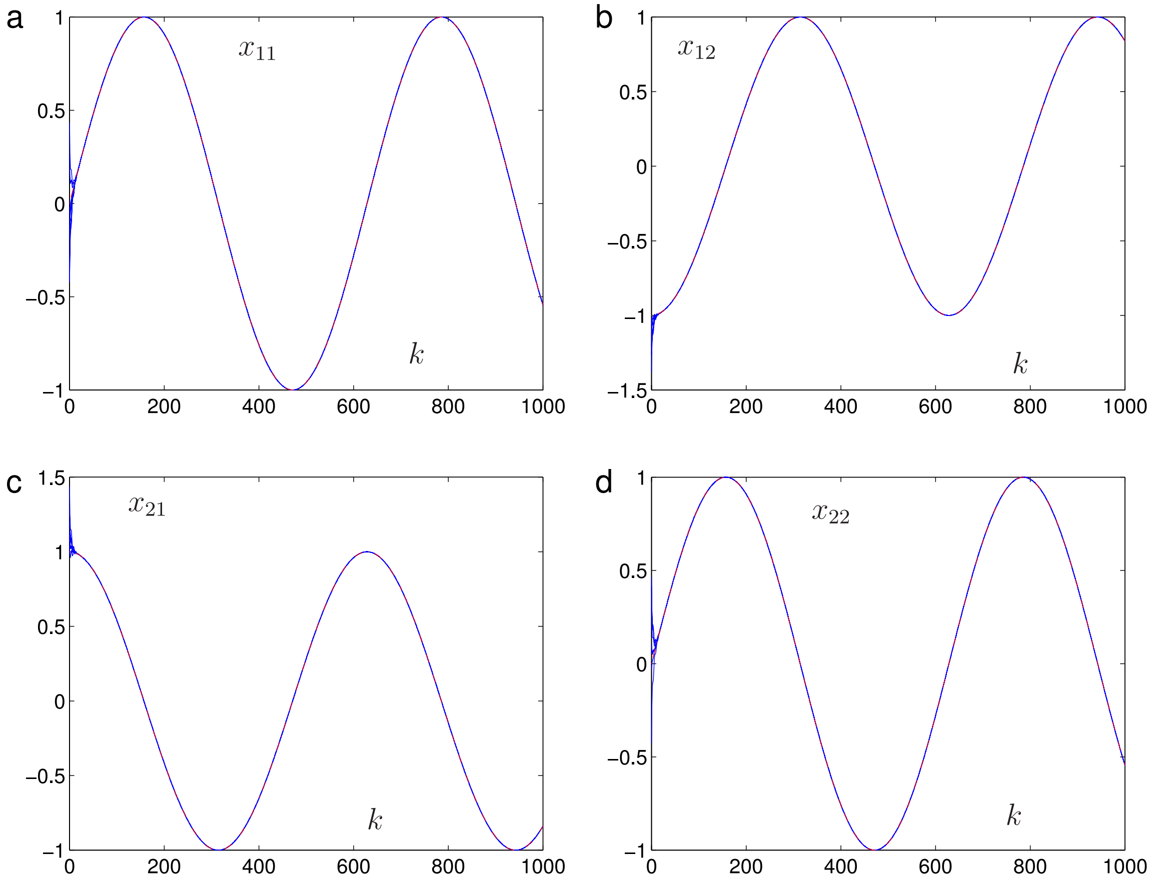

Fig. 2. Neural state X_k synthesized by Taylor-type DTZNN model (18) with $h = 0.3$ and $\tau = 0.01$, where the dash-dotted curves correspond to the theoretical solution and the solid curves correspond to the neural-network solutions. (a) x_{11}-profile. (b) x_{12}-profile. (c) x_{21}-profile. (d) x_{22}-profile.

more detailed maximum SSRE (MSSRE) data of the models are shown in Table 5 with respect to different values of sampling gap τ. From Table 5, the following important facts are summarized, which further demonstrate the theoretical analyses completely.

- First, the MSSRE of Newton iteration (6) without utilizing time derivative of time-varying coefficient changes in an $O(\tau)$ manner. For example, it can be observed from Table 5 that the MSSREs of Newton iteration (6) are of order 10^{-1}, 10^{-2} or 10^{-3}, corresponding to $\tau = 0.1$, 0.01 or 0.001, respectively.
- Second, the MSSRE of Euler-type DTZNN model (5), which utilizes time derivative of time-varying coefficient and is discretized via Euler difference, changes in an $O(\tau^2)$ manner. This fact is clearly shown in Table 5.
- Third, the MSSRE of Taylor-type DTZNN model (18), which utilizes time derivative of time-varying coefficient and is discretized via the proposed Taylor-type 1-step-ahead numerical differentiation rule (15), changes in an $O(\tau^3)$ manner. For example, as seen from Table 5, the MSSREs of Taylor-type DTZNN model (18) are of order 10^{-3}, 10^{-6} or 10^{-9}, corresponding to $\tau = 0.1$, 0.01 or 0.001, respectively.

Besides, it is worth investigating the performance of the three discrete-time models in a noisy environment. In view of $\tau = ((2\varepsilon + 6\sigma)/M)^{1/3}$ in Remark 3, the zero-mean Gaussian white noises can be counted as having no significant impact on the MSSRE of Taylor-type DTZNN model (18) for standard deviation $\sigma < (\tau^3 M - 2\varepsilon)/6$. Note that, for time-varying matrix inversion problem (7), we have $M = 1$ and $\varepsilon = 5 \times 10^{-13}$. Therefore, the maximal values of σ which do not influence the performance of Taylor-type DTZNN model (18) are 1.6667×10^{-4} and 1.6667×10^{-7} corresponding to using sampling gap $\tau = 0.1$ and 0.01, respectively. Means of MSSREs of three discrete-time models with zero-mean Gaussian white noises for 10 trials are shown in Fig. 4. As seen from the figure, starting with the same initial state $X_0 = [0.2, -0.8; 0.8, 0.2]$, the performance of Taylor-type DTZNN model (18) is not reduced for $\sigma < 10^{-4}$ with $\tau = 0.1$, and for $\sigma < 10^{-7}$ with $\tau = 0.01$. In addition, for standard deviation $\sigma > (\tau^3 M - 2\varepsilon)/6$, it can be observed from the figure that the noises have remarkable impacts on the performance of Taylor-type DTZNN model (18), which means that some noise reduction technologies can be considered for a better performance, such as [10]. Moreover, Fig. 4 shows that, even with the large noise, the performance of each DTZNN model is better than that of Newton iteration (6), especially Taylor-type DTZNN model (18).

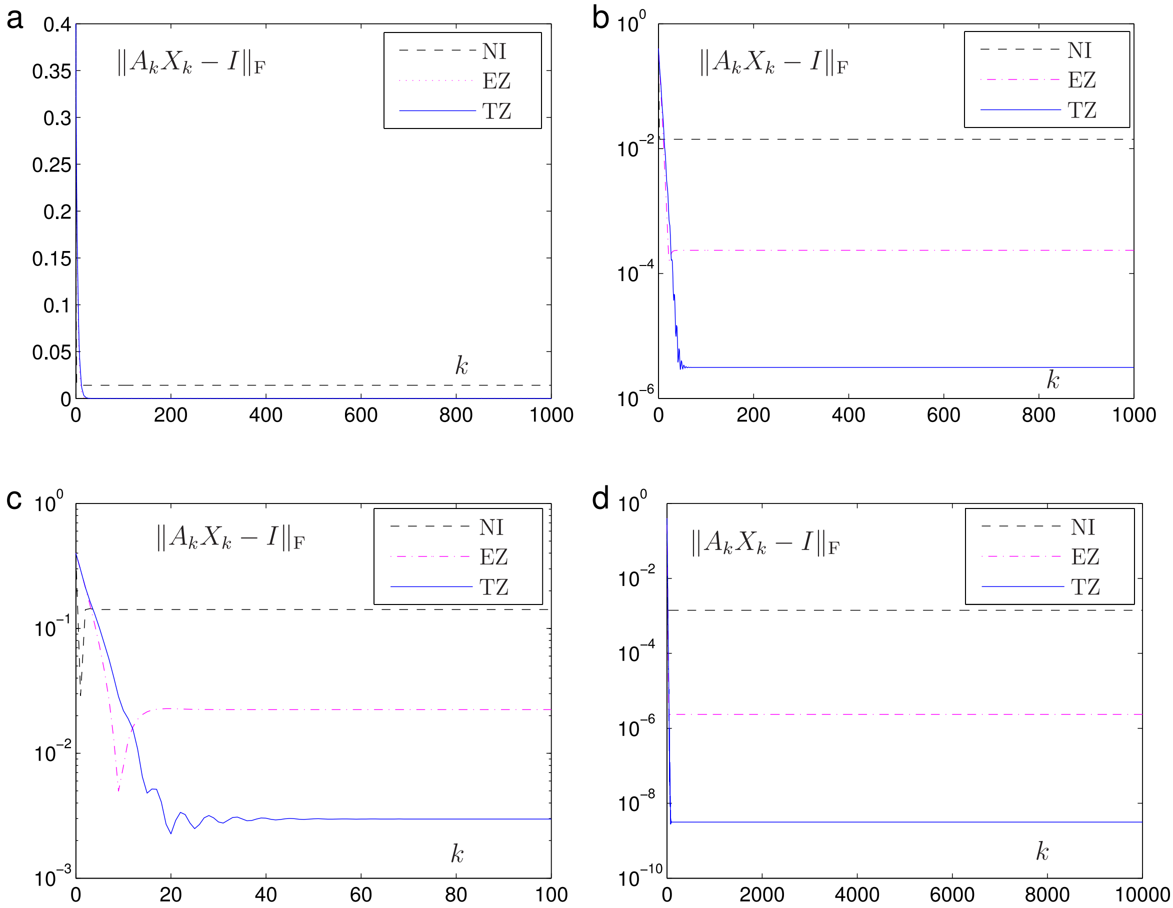

Fig. 3. Residual errors $\|A_kX_k - I\|_F$ of three discrete-time models with $h = 0.3$, where NI, EZ and TZ represent Newton iteration (6), Euler-type DTZNN model (5) and Taylor-type DTZNN model (18), respectively. (a) The residual errors with $\tau = 0.01$. (b) The semilog plot of the residual errors with $\tau = 0.01$. (c) The semilog plot of the residual errors with $\tau = 0.1$. (d) The semilog plot of the residual errors with $\tau = 0.001$.

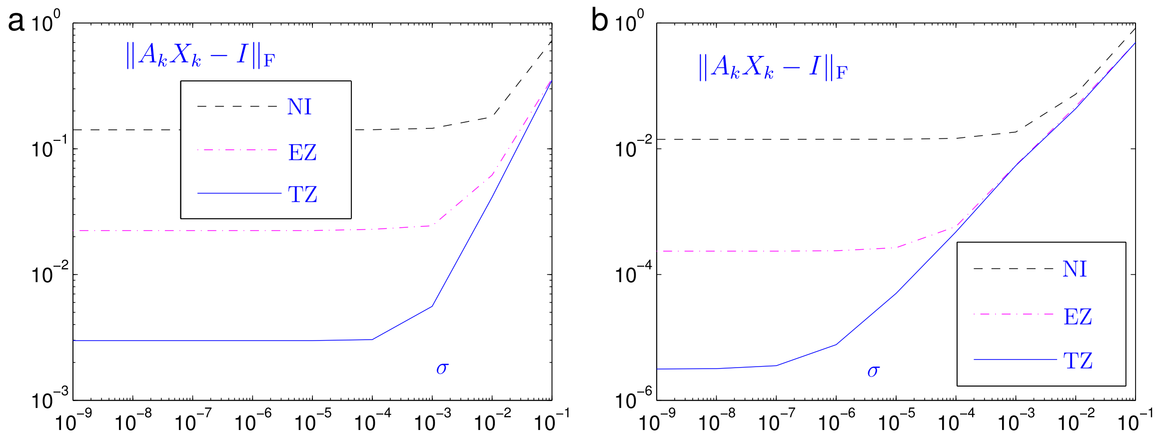

Fig. 4. Means of MSSREs of three discrete-time models with $h = 0.3$ in a noisy environment for 10 trials. (a) $\tau = 0.1$. (b) $\tau = 0.01$.

6. Conclusions

In this paper, in order to achieve higher computational precision in approximating the first-order derivative and discretize the continuous-time ZNN model [e.g., (3)] effectively, a Taylor-type numerical differentiation rule (15) has been proposed and investigated. Compared with the Lagrange-type numerical differentiation rule, the Taylor-type one not only discretizes the continuous-time ZNN model successfully, but also achieves a higher computational precision. In addition, a formula [i.e., (17)] has been proposed to obtain the optimal step length for the Taylor-type numerical differentiation rule (15). Moreover, the stability and convergence of three discrete-time models [i.e., Newton iteration (6), Euler-type DTZNN model (5) and Taylor-type DTZNN model (18)] have been investigated. Theoretical analyses have shown that the SSREs of Newton iteration (6), Euler-type DTZNN model (5) and Taylor-type DTZNN model (18) have an $O(\tau)$, $O(\tau^2)$ and $O(\tau^3)$ pattern, respectively. Finally, numerical experimental results have further demonstrated the efficacy and advantages of the proposed Taylor-type numerical differentiation rule (15) for first-order derivative approximation and ZNN discretization.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grants 61075121 and 60935001, and also by the 2012 Scholarship Award for Excellent Doctoral Student Granted by Ministry of Education of China under Grant 3191004.

References

[1] Y.N. Zhang, D.C. Jiang, J. Wang, A recurrent neural network for solving Sylvester equation with time-varying coefficients, IEEE Trans. Neural Netw. 13 (5) (2002) 1053–1063.
[2] Y.N. Zhang, S.S. Ge, Design and analysis of a general recurrent neural network model for time-varying matrix inversion, IEEE Trans. Neural Netw. 16 (6) (2005) 1477–1490.
[3] S.C. Chapra, R.P. Canale, Numerical Methods for Engineers, third ed., McGraw-Hill, New York, 1998.
[4] T.N. Krishnamurti, L. Bounoua, An Introduction to Numerical Weather Prediction Technique Techniques, CRC Press, Boca Raton, FL, 1995.
[5] M. Gentile, A. Sommariva, M. Vianello, Polynomial interpolation and cubature over polygons, J. Comput. Appl. Math. 235 (17) (2011) 5232–5239.
[6] L. Gavete, F. Urena, J.J. Benito, E. Salete, A note on the dynamic analysis using the generalized finite difference method, J. Comput. Appl. Math. 252 (2013) 132–147.
[7] L.J. Wang, X. Han, J. Liu, J.J. Chen, An improved iteration regularization method and application to reconstruction of dynamic loads on a plate, J. Comput. Appl. Math. 235 (14) (2011) 4083–4094.
[8] J.P. Li, General explicit difference formulas for numerical differentiation, J. Comput. Appl. Math. 183 (1) (2005) 29–52.
[9] Y.N. Zhang, Y. Chou, J.H. Chen, Z.Z. Zhang, L. Xiao, Presentation, error analysis and numerical experiments on a group of 1-step-ahead numerical differentiation formulas, J. Comput. Appl. Math. 239 (2013) 406–414.
[10] M. Mboup, C. Join, M. Filess, Numerical differentiation with annihilators in noisy environment, Numer. Algorithms 50 (2009) 439–467.
[11] A.G. Ramm, A.B. Smirnova, On stable numerical differentiation, Math. Comp. 70 (2001) 1131–1153.
[12] Y.N. Zhang, B.G. Mu, H.C. Zheng, Link between and comparison and combination of Zhang neural network and quasi-Newton BFGS method for time-varying quadratic minimization, IEEE Trans. Cybernet. 42 (2) (2013) 490–503.
[13] Y.N. Zhang, C.F. Yi, Zhang Neural Networks and Neural-Dynamic Method, Nova Science Publishers, New York, 2011.
[14] D.S. Guo, Y.N. Zhang, Zhang neural network, Getz–Marsden dynamic system, and discrete-time algorithms for time-varying matrix inversion with application to robots' kinematic control, Neurocomputing 97 (2012) 22–32.
[15] K. Burrage, P.M. Burrage, Low rank Runge–Kutta methods, symplecticity and stochastic Hamiltonian problems with additive noise, J. Comput. Appl. Math. 236 (16) (2012) 3920–3930.
[16] D.F. Griffiths, D.J. Higham, Numerical Methods for Ordinary Differential Equations: Initial Value Problems, Springer, England, 2010.
[17] J.H. Mathews, K.D. Fink, Numerical Methods Using MATLAB, fourth ed., Prentice-Hall, Inc., Englewood Cliffs, NJ, 2005.
[18] R. Durrett, Probability: Theory and Examples, Wadsworth and Brooks, Pacific Grove, 1991.