IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 2, 2020, accepted December 10, 2020, date of publication December 18, 2020,
date of current version December 31, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3045780

GMDS-ZNN Model 3 and Its Ten-Instant Discrete Algorithm for Time-Variant Matrix Inversion Compared With Other Multiple-Instant Ones

DONGQING WU[1,2,3,4], YUNONG ZHANG[2,3,4], (Member, IEEE), JINJIN GUO[2,3,4], ZHENYU LI[2,3,4], AND LIANGJIE MING[3,4,5]

[1]School of Computational Science, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China
[2]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China
[3]Guangdong Key Laboratory of Modern Control Technology, Guangzhou 510070, China
[4]Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou 510006, China
[5]School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China

Corresponding author: Yunong Zhang (zhynong@mail.sysu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61976230, in part by the Project Supported by Guangdong Province Universities and Colleges Pearl River Scholar Funded Scheme under Grant 2018, in part by the Research Fund Program of Guangdong Key Laboratory of Modern Control Technology under Grant 2017B030314165, in part by the Key-Area Research and Development Program of Guangzhou under Grant 202007030004, in part by the China Postdoctoral Science Foundation under Grant 2018M643306, and in part by the Fundamental Research Funds for the Central Universities under Grant 19lgpy227.

ABSTRACT The online time-variant matrix inversion problem has attracted extensive attention and study, because of its considerable appearance and application in scientific research and industrial production. For various control optimization problems, the demand for the high-precision and rapid-convergence of matrix inversion algorithm is increasing. It remains an ongoing challenge due to the rigorous requirements of precision and convergence of the algorithm. In this paper, on the basis of our previous works, by using Zhang neural network (ZNN) method, a continuous time-variant matrix inversion model, which is also a Getz-Marsden dynamic system (i.e., GMDS-ZNN model 3), is proposed. Besides, a general ten-instant Zhang et al discretization (ZeaD) formula is presented and investigated, with corresponding theoretical results being provided. Next, by applying this general formula to discretize the continuous time-variant matrix inversion model, a general ten-instant discrete time-variant matrix inversion (DTVMI) algorithm with the sixth-order precision is proposed. For comparison, four other DTVMI algorithms, with the second-, third-, fourth-, and fifth-order precisions, are also proposed and presented, respectively, by using other ZeaD formulas to discretize the continuous time-variant matrix inversion model. Besides, for the situation of coefficient matrix derivative being unknown, we provided the formula of estimating it with the fifth-order precision. With the help of the proposed matrix derivative estimation formula, the actual application field of GMDS-ZNN model 3 is expanded evidently. Finally, theoretical analyses and simulation experiment results highlight the effectiveness and accuracy of the proposed GMDS-ZNN model 3 and DTVMI algorithms.

INDEX TERMS Time-variant matrix inversion, Getz-Marsden dynamic system, ten-instant ZeaD formula, Zhang neural network, discrete algorithm.

I. INTRODUCTION

Solving the linear matrix equations, such as Lyapunov equation, Sylvester equation, and Stein equation, is considered to be an important issue widely encountered in a variety of science and engineering fields. One of the linear matrix equation sub-problems is the online matrix inversion problem. The solution of matrix inversion is one of the fundamental problems encountered in a variety of optimization problems, such as image reconstruction [1], nonlinear optimization [2], [3], and robot kinematics [4]. In general, the matrix inversion problem can be formulated as $AX = I$, where $A \in \mathbb{R}^{n\times n}$ and identity matrix $I \in \mathbb{R}^{n\times n}$ are constant matrices. $X \in \mathbb{R}^{n\times n}$ is the unknown matrix to be computed.

The associate editor coordinating the review of this manuscript and approving it for publication was Yongming Li.

Several methods were proposed and applied to solve this problem. Nowadays, due to the in-depth research in artificial intelligence, the neural-dynamic approach based on recurrent neural networks (RNN) has been considered to be an effective option for online optimization problem-solving. Methods based on RNN have manifested their high-speed parallel-processing nature and convenience of hardware implementation [5], [6]. They have been motivated as analog machines to solve optimization problems [7]–[10]. In 2001, aiming at the online solution of various time-variant problems, a new class of RNN, termed Zhang neural network (ZNN), has been proposed in [11]. The ZNN model is essentially based on an indefinite function termed Zhang function (ZF) which is served as error-monitoring. Since 2011, the ZFs have been found to speed up and consolidate the development of various ZNN models [12]–[15]. In [16], Getz and Marsden constructed a dynamical system to approximate the inverse of a time-variant matrix which was convergent exponentially toward the true time-variant inverse, and argued that all positive definite fixed matrix may be dynamically inversed in prescribed time and no initial guess is needed. In [16], a model named Getz-Marsden dynamic system (GMDS1) to inverse time-variant matrix was proposed with many characteristics being worth discussing and further investigating. Furthermore, based on the ingenious combination of the model in [16] and gradient neural network, GMDS-ZNN model 2 was presented in [17]. In accordance with previous works in [16], [18], we proved that the conventional GMDS1 is actually one of the explicit ZNN models, which is named as GMDS-ZNN model 1. In [18], we proposed the GMDS-ZNN model 2 variants based on ZNN design formula and ZF.

The above continuous models must be discretized to meet the needs of most practical applications. Therefore, the proposed GMDS-ZNN model 2 was discretized by different discretization formulas [18]–[20], specifically including Euler forward formula, Taylor-Zhang discretization formula and five-instant Zhang et al discretization (ZeaD) formula. In [21], a three-step general ZeaD formula was designed to approximate the first-order derivative of the target point and was used to discretize the continuous-time ZNN models for time-variant matrix inversion. The maximum steady-state residual errors of algorithms presented in [21] substantiate that the three-step general discrete ZNN algorithm is superior to Newton iteration and one-step discrete ZNN for the time-variant matrix inversion. In our previous research [18], [22], we conducted extensive numerical experiments, such that GMDS-ZNN model 1 and GMDS-ZNN model 2 were both presented and compared. Numerical results show that both models have fixed error pattern and are confirmed to be proportional to the sampling gap. In [18], [22], GMDS-ZNN model 2 is proved to have higher precision. Moreover, based on our best knowledge, we know that the precision of the discrete algorithm is strongly dependent on the accuracy of the ZeaD formula. In short, high-precision ZeaD formulas lead to high-precision discrete algorithms.

In the published works, multiple-instant discrete algorithms for time-variant matrix inversion were developed and applied. It should be noted that the highest order of residual error of those algorithms is only $O(\tau^5)$ and no previous works consider the situation of $\dot{A}$ being unknown. Therefore, it is necessary to explore the numerical difference formula with higher accuracy and the situation of $\dot{A}$ being unknown. In this paper, a continuous model for time-variant matrix inversion is proposed. To obtain high-precision discrete GMDS-ZNN model 3, a general ten-instant ZeaD formula is constructed and proposed. Secondly, with the help of the general ten-instant ZeaD formula, a general ten-instant DTVMI algorithm for matrix inversion is further proposed. At last, other multiple-instant algorithms, such as two-instant, four-instant, six-instant, along with eight-instant algorithms are also derived and presented by applying other ZeaD formulas.

The rest of this paper is organized into four sections. Continuous GMDS-ZNN model 3 is proposed in Section II. Section III presents discrete algorithms of GMDS-ZNN model 3. Two theorems are presented and proved to consolidate the convergence and stability of the proposed algorithm. In Section IV, two time-variant matrices are provided as the benchmark examples to verify the effectiveness and accuracy of the proposed algorithm. Eventually, Section V concludes this paper with final remarks.

The main contributions of this paper lie in the following facts.

1) By employing ZNN method, the GMDS-ZNN model 3 for matrix inversion is developed.
2) For digital hardware realization, a novel high-precision ten-instant DTVMI algorithm, i.e., DTVMI-V algorithm, is proposed and constructed with corresponding theoretical analyses.
3) For comparison, by exploiting the general ten-instant ZeaD formula and other multiple-instant ZeaD formulas, five DTVMI algorithms, with $O(\tau^2)$, $O(\tau^3)$, $O(\tau^4)$, $O(\tau^5)$, and $O(\tau^6)$ precisions are proposed and constructed, respectively.
4) The stability and convergence of the DTVMI-V algorithm are proved theoretically.
5) For the situation of $\dot{A}_k$ unknown, the formula of estimating $\dot{A}_k$ with truncation error $O(\tau^5)$ is provided. The formula expands the actual application field of GMDS-ZNN model 3 evidently.
6) Numerical experiment results indicate that the proposed DTVMI-V algorithm is effective and feasible for time-variant matrix inversion.

II. PROBLEM FORMATION AND SOLVERS

In this work, we consider the problem of time-variant matrix inversion in the form of

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

where $A(t)$ is a smoothly time-variant nonsingular matrix, I is the identity matrix of dimension n, and $X(t)$ is the time-variant unknown matrix to be obtained. Numerous efforts have been made to solve the computational problem of fast matrix inversion since the mid-1980s, accompanied by a series of algorithms were designed and applied [23]. The time complexity of those algorithms is usually convergent to $O(n^3)$ asymptotically. Such poor performance may not be suitable in most large-scale online applications. Therefore, a wealth of parallel-processing computational methods of matrix inversion have been proposed, developed, and applied, to meet the actual scientific and engineering requirements. The aim of the paper is to propose a new ten-instant discrete matrix inversion algorithm to compute $X(t)$ with the truncation error of $O(\tau^6)$. To monitor and accelerate the solving process of (1), we define the following scalar-valued nonnegative function [12], [24]:

$$\epsilon(t) = \frac{1}{2}\|X(t)A(t) - I\|_{\mathrm{F}}^2 \in \mathbb{R}, \tag{2}$$

where $\epsilon(t)$ is close to zero when $X(t)$ is close to $A^{-1}(t)$ with $t \to +\infty$. Besides, symbol $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius-norm of a matrix and I represents the identity matrix as before. Next, we adopt the gradient neural network (GNN) to obtain the continuous solution model for the time-variant matrix inversion (TVMI) problem [11], [25]:

$$\dot{X}(t) = -\gamma\partial\epsilon(t)/\partial X(t) = -\gamma(X(t)A(t) - I)A^{\mathrm{T}}(t), \tag{3}$$

where $\gamma \in \mathbb{R}^+$ is the convergence rate. The larger γ means the faster convergence rate. The superscript $^{\mathrm{T}}$ denote the transformation operator of matrix. For problem of complex matrix inversion, Zhang et al firstly defined an indefinite error-monitoring function $E(t) = A(t) - X^{-1}(t) \in \mathbb{R}^{n\times n}$, and then obtained the following Zhang neural network (ZNN) [24], [26]:

$$\dot{X}(t) = -X(t)\dot{A}(t)X(t) - \lambda(X(t)A(t)X(t) - X(t)), \tag{4}$$

where $\lambda \in \mathbb{R}^+$ denotes the reciprocal of a capacitance parameter, which should set as large as the maximum limitation of the hardware [27].

It is necessary to point out that (4) is the explicit form of GMDS model. Therefore, (4) is also termed GMDS-ZNN model 1. Referencing to [24], we combine (3) with (4) to get a continuous GNN-ZNN type (GZ-type) model by replacing the second term of the right-hand side of (4) with the right-hand side of (3). The final continuous GZ-type model is obtained as

$$\dot{X}(t) = -X(t)\dot{A}(t)X(t) - \gamma(X(t)A(t) - I)A^{\mathrm{T}}(t), \tag{5}$$

which is named continuous GMDS-ZNN model 3 (5). The continuous GMDS-ZNN model 3 (5) is actually the symmetric model of GMDS-ZNN model 2. It is worth noting that the continuous GMDS-ZNN model 3 (5) takes the advantage of both GNN (i.e., fast convergence rate) and ZNN (e.g., high convenience accuracy). Next, we will discuss the discrete algorithms of the continuous GMDS-ZNN model 3 (5).

III. DISCRETE ALGORITHMS

Since digital hardware requires discrete computational algorithm, we discretize the developed GMDS-ZNN model 3 (5) for investigating its features. The above-developed model is discretized by five different discretization formulas, such as Euler forward formula, Taylor-Zhang discretization formula (TZDF), six-instant ZeaD formula, eight-instant ZeaD formula, and ten-instant ZeaD formula.

A. EULER FORWARD FORMULA

Euler forward formula is a first-order numerical method for solving ordinary differential equations with given initial values. It is one of the most basic explicit methods for solving numerical ordinary differential equations. This formula has the first-order truncation error [28]. The formula is described as follows:

$$\dot{u}_k = \frac{1}{\tau}(u_{k+1} - u_k) + O(\tau), \tag{6}$$

where $\tau \in (0, 1)$ denotes the sampling gap, and $k = 0, 1, 2, \cdots$ denotes the updating index of u. Applying (6) to (5), we get the following DTVMI-I algorithm, which is shown as

$$X_{k+1} \doteq -\tau X_k\dot{A}_kX_k - h(X_kA_k - I)A_k^{\mathrm{T}} + X_k, \tag{7}$$

where symbol $\doteq$ denotes the right-hand side of (7) assigned to X_{k+1} as the estimation, $\dot{A}_k$ denotes $\dot{A}(t_k)$, and $h = \gamma\tau > 0$ denotes the step size.

B. TAYLOR-ZHANG DISCRETIZATION FORMULA

Compared with Euler forward formula, TZDF has second-order truncation error, whose precision is higher than Euler forward formula. TZDF is given as follows [29]:

$$\dot{u}_k = \frac{1}{2\tau}(2u_{k+1} - 3u_k + 2u_{k-1} - u_{k-2}) + O(\tau^2). \tag{8}$$

With (5) and (8), the four-instant DTVMI algorithm is named as DTVMI-II algorithm, which is expressed as follows:

$$\begin{aligned} X_{k+1} \doteq\ & -\tau X_k\dot{A}_kX_k - h(X_kA_k - I)A_k^{\mathrm{T}} \\ & + \frac{3}{2}X_k - X_{k-1} + \frac{1}{2}X_{k-2}, \end{aligned} \tag{9}$$

where $h > 0$ denotes the step size again as before.

C. SIX-INSTANT ZeaD FORMULA

According to [30], a six-instant ZeaD formula is shown as follows:

$$\begin{aligned} \dot{u}_k = \frac{1}{2\tau}\Big(& u_{k+1} - \frac{5}{24}u_k - \frac{1}{2}u_{k-1} - \frac{1}{4}u_{k-2} \\ & - \frac{1}{6}u_{k-3} + \frac{1}{8}u_{k-4}\Big) + O(\tau^3). \end{aligned} \tag{10}$$

With (5) and (10), the six-instant DTVMI algorithm is named as DTVMI-III algorithm, which is given as follows:

$$X_{k+1} \doteq -2\tau X_k\dot{A}_kX_k - 2h(X_kA_k - I)A_k^{\mathrm{T}}$$

TABLE 1. Coefficients of ten-instant ZeaD formulas.

No.	d_0	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9
1	175/432	737/7560	−35/144	−245/864	−35/216	35/288	343/2160	−35/864	−95/1008	35/864
2	350/823	9839/691320	−105/823	−280/823	−105/823	35/823	777/4115	70/2469	−935/5761	385/6584
3	700/1677	21587/469560	−280/1677	−560/1677	−70/559	140/1677	1148/8385	70/1677	−1810/11739	245/4472
4	100/237	947/66360	−20/237	−100/237	−10/79	40/237	88/1185	10/237	−230/1659	95/1896
5	350/873	31063/244440	−280/873	−175/873	−140/873	35/873	952/4365	−35/873	−670/6111	35/776
6	175/428	7051/89880	−175/856	−35/107	−105/856	35/428	819/4280	−35/642	−555/5992	35/856

$$+\frac{5}{24}X_k+\frac{1}{2}X_{k-1}+\frac{1}{4}X_{k-2}+\frac{1}{6}X_{k-3}-\frac{1}{8}X_{k-4}, \quad (11)$$

where $h>0$ denotes the step size again as before.

D. EIGHT-INSTANT ZeaD FORMULA

According to [30], an eight-instant ZeaD formula is described as follows:

$$\dot{u}_k=\frac{1}{\tau}\bigg(\frac{50}{111}u_{k+1}-\frac{51}{2220}u_k-\frac{20}{111}u_{k-1}-\frac{30}{111}u_{k-2}-\frac{10}{111}u_{k-3}+\frac{35}{444}u_{k-4}+\frac{44}{555}u_{k-5}-\frac{5}{111}u_{k-6}\bigg)+O(\tau^4). \quad (12)$$

With (5) and (12), the eight-instant DTVMI algorithm is named as DTVMI-IV algorithm, which is given as follows:

$$X_{k+1}\doteq-\tau X_k\dot{A}_kX_k-h(X_kA_k-I)A_k^{\mathrm{T}}+\frac{111}{50}\bigg(\frac{51}{2220}X_k+\frac{20}{111}X_{k-1}+\frac{30}{111}X_{k-2}+\frac{10}{111}X_{k-3}-\frac{35}{444}X_{k-4}-\frac{44}{555}X_{k-5}+\frac{5}{111}X_{k-6}\bigg). \quad (13)$$

E. GENERAL TEN-INSTANT ZeaD FORMULA

The general ten-instant ZeaD formula is first investigated and obtained in [31]. It is proved theoretically that the accuracy of this formula can reach order 6, and it is verified in solving the square root problem of time-variant matrix [31]. The general ten-instant formula is given as follows:

$$\dot{u}_k=\frac{1}{\tau}(d_0u_{k+1}+d_1u_k+d_2u_{k-1}+d_3u_{k-2}+d_4u_{k-3}+d_5u_{k-4}+d_6u_{k-5}+d_7u_{k-6}+d_8u_{k-7}+d_9u_{k-8})+O(\tau^5). \quad (14)$$

According to the theoretical model and derivation of the literature [31], we calculate and list six groups of the coefficients of ten-instant ZeaD formulas. These coefficients d_i are provided in Table 1. We use the ten-instant ZeaD formula (14) to discretize (5). The general ten-instant DTVMI algorithm named as DTVMI-V algorithm, is thus obtained:

$$X_{k+1}=\frac{1}{d_0}(-\tau X_k\dot{A}_kX_k-h(X_kA_k-I)A_k^{\mathrm{T}}-d_1X_k-d_2X_{k-1}-d_3X_{k-2}-d_4X_{k-3}-d_5X_{k-4}-d_6X_{k-5}-d_7X_{k-6}-d_8X_{k-7}-d_9X_{k-8})+O(\tau^6), \quad (15)$$

where $h=\tau\gamma$. It is worth mentioning that when h is fixed as a constant, the truncation error is consistent with $O(\tau^6)$. When h is not fixed and γ is a constant, the error is upgraded, which is consistent with $O(\tau^5)$.

Theorem 1: *With sampling gap $\tau\in(0,1)$, the maximal steady-state residual error $\lim_{k\to+\infty}\sup\|X_{k+1}A_{k+1}-I\|_{\mathrm{F}}$ of (15) is $O(\tau^6)$.*

Proof: According to Theorem 1 and Theorem 2 in [31], we know that the (15) is 0-stable, consistent, and convergent. Therefore, it converges with the order of its truncation error. Assume that Z_{k+1} is the exact solution of $Z_{k+1}A_{k+1}-I=O$. Based on (15), we have $X_{k+1}=Z_{k+1}+O(\tau^6)$ with $\tau\in(0,1)$, and further have

$$=\lim_{k\to+\infty}\sup\|X_{k+1}A_{k+1}-I\|_{\mathrm{F}}$$
$$=\lim_{k\to+\infty}\sup\|(Z_{k+1}+O(\tau^6))A_{k+1}-I\|_{\mathrm{F}}$$
$$=\lim_{k\to+\infty}\sup\|I+O(\tau^6)A_{k+1}-I\|_{\mathrm{F}}=O(\tau^6).$$

The proof is thus completed. ■

F. SITUATION OF $\dot{A}$ BEGING UNKNOWN

In (7), (9), (11), (13), and (15), $\dot{A}_k$ is known, but this condition may not meet in some application scenarios. One practical approach is to estimate $\dot{A}_k$ through the current and past sample points (e.g., $A_k, A_{k-1}, A_{k-2},\cdots$). According to [29], the general estimation of $\dot{A}_k$ is formulated as

$$\dot{A}_k=\frac{1}{\tau}\bigg(\sum_{i=1}^{n}\xi_{n-i+1}A_{k-i+1}\bigg)+O(\tau^q), \quad (16)$$

where n represents the number of sample points involved in estimating $\dot{A}_k$, ξ_{n-i+1} denotes the coefficient, and q represents the order of truncation error of (16). We aim to find all the coefficients (e.g., $\xi_n,\xi_{n-1},\cdots,\xi_1$) in (16) to estimate $\dot{A}_k$ under the given precision order $q=5$. The following theorem gives the estimation formula of $\dot{A}_k$.

Theorem 2: For the situation of $\dot{A}_k$ unknown, the estimation of $\dot{A}_k$ is given as

$$\dot{A}_k = \frac{1}{\tau}(\xi_7 A_k + \xi_6 A_{k-1} + \xi_5 A_{k-2} + \xi_4 A_{k-3} + \xi_3 A_{k-4} + \xi_2 A_{k-5} + \xi_1 A_{k-6}) + O(\tau^5), \quad (17)$$

where $\xi_1 = 0$, $\xi_2 = -1/5$, $\xi_3 = 5/4$, $\xi_4 = -10/3$, $\xi_5 = 5$, $\xi_6 = -5$, and $\xi_7 = 137/60$.

Proof: The general formula corresponding to (17) is shown as follows:

$$\dot{f}_k = \frac{1}{\tau}(\xi_7 f_k + \xi_6 f_{k-1} + \xi_5 f_{k-2} + \xi_4 f_{k-3} + \xi_3 f_{k-4} + \xi_2 f_{k-5} + \xi_1 f_{k-6}) + O(\tau^5). \quad (18)$$

The Taylor expansions of f_{k-1} through f_{k-6} are

$$f_{k-1} = f_k - \tau\dot{f}_k + \frac{\tau^2}{2!}\ddot{f}_k - \frac{\tau^3}{3!}\dddot{f}_k + \frac{\tau^4}{4!}f_k^{(4)} - \frac{\tau^5}{5!}f_k^{(5)} + O(\tau^5), \quad (19)$$

$$f_{k-2} = f_k - (2\tau)\dot{f}_k + \frac{(2\tau)^2}{2!}\ddot{f}_k - \frac{(2\tau)^3}{3!}\dddot{f}_k + \frac{(2\tau)^4}{4!}f_k^{(4)} - \frac{(2\tau)^5}{5!}f_k^{(5)} + O(\tau^5), \quad (20)$$

$$f_{k-3} = f_k - (3\tau)\dot{f}_k + \frac{(3\tau)^2}{2!}\ddot{f}_k - \frac{(3\tau)^3}{3!}\dddot{f}_k + \frac{(3\tau)^4}{4!}f_k^{(4)} - \frac{(3\tau)^5}{5!}f_k^{(5)} + O(\tau^5), \quad (21)$$

$$f_{k-4} = f_k - (4\tau)\dot{f}_k + \frac{(4\tau)^2}{2!}\ddot{f}_k - \frac{(4\tau)^3}{3!}\dddot{f}_k + \frac{(4\tau)^4}{4!}f_k^{(4)} - \frac{(4\tau)^5}{5!}f_k^{(5)} + O(\tau^5), \quad (22)$$

$$f_{k-5} = f_k - (5\tau)\dot{f}_k + \frac{(5\tau)^2}{2!}\ddot{f}_k - \frac{(5\tau)^3}{3!}\dddot{f}_k + \frac{(5\tau)^4}{4!}f_k^{(4)} - \frac{(5\tau)^5}{5!}f_k^{(5)} + O(\tau^5), \quad (23)$$

$$f_{k-6} = f_k - (6\tau)\dot{f}_k + \frac{(6\tau)^2}{2!}\ddot{f}_k - \frac{(6\tau)^3}{3!}\dddot{f}_k + \frac{(6\tau)^4}{4!}f_k^{(4)} - \frac{(6\tau)^5}{5!}f_k^{(5)} + O(\tau^5), \quad (24)$$

where $\dot{f}_k$, $\ddot{f}_k$, $\dddot{f}_k$, $f_k^{(4)}$, and $f_k^{(5)}$ represent the first-, second-, third-, fourth-, and fifth-order derivatives at $t = k\tau$, respectively. Substitute (19), (20), (21), (22), (23), and (24) into (18). Then, (18) is reformulated as

$$\eta_6 f_k - \eta_5 \dot{f}_k \tau + \frac{\eta_4}{2!}\tau^2\ddot{f}_k - \frac{\eta_3}{3!}\tau^3\dddot{f}_k + \frac{\eta_2}{4!}\tau^4 f_k^{(4)} - \frac{\eta_1}{5!}\tau^5 f_k^{(5)} = O(\tau^5), \quad (25)$$

where $\eta_6 = \xi_7 + \xi_6 + \xi_5 + \xi_4 + \xi_3 + \xi_2 + \xi_1$, $\eta_5 = 1 + \xi_6 + 2\xi_5 + 3\xi_4 + 4\xi_3 + 5\xi_2 + 6\xi_1$, $\eta_4 = \xi_6 + 2^2\xi_5 + 3^2\xi_4 + 4^2\xi_3 + 5^2\xi_2 + 6^2\xi_1$, $\eta_3 = \xi_6 + 2^3\xi_5 + 3^3\xi_4 + 4^3\xi_3 + 5^3\xi_2 + 6^3\xi_1$, $\eta_2 = \xi_6 + 2^4\xi_5 + 3^4\xi_4 + 4^4\xi_3 + 5^4\xi_2 + 6^4\xi_1$, $\eta_1 = \xi_6 + 2^5\xi_5 + 3^5\xi_4 + 4^5\xi_3 + 5^5\xi_2 + 6^5\xi_1$. To let the left-hand side of (25) converge to zero, the following linear system should hold:

$$\begin{cases} \eta_6 = \xi_7 + \xi_6 + \xi_5 + \xi_4 + \xi_3 + \xi_2 + \xi_1 = 0, \\ \eta_5 = 1 + \xi_6 + 2\xi_5 + 3\xi_4 + 4\xi_3 + 5\xi_2 + 6\xi_1 = 0, \\ \eta_4 = \xi_6 + 2^2\xi_5 + 3^2\xi_4 + 4^2\xi_3 + 5^2\xi_2 + 6^2\xi_1 = 0, \\ \eta_3 = \xi_6 + 2^3\xi_5 + 3^3\xi_4 + 4^3\xi_3 + 5^3\xi_2 + 6^3\xi_1 = 0, \\ \eta_2 = \xi_6 + 2^4\xi_5 + 3^4\xi_4 + 4^4\xi_3 + 5^4\xi_2 + 6^4\xi_1 = 0, \\ \eta_1 = \xi_6 + 2^5\xi_5 + 3^5\xi_4 + 4^5\xi_3 + 5^5\xi_2 + 6^5\xi_1 = 0. \end{cases} \quad (26)$$

The ordinary solution of (26) is

$$\begin{cases} \xi_7 = -\xi_1 + 137/60, \\ \xi_6 = 6\xi_1 - 5, \\ \xi_5 = -15\xi_1 + 5, \\ \xi_4 = 20\xi_1 - 10/3, \\ \xi_3 = -15\xi_1 + 5/4, \\ \xi_2 = 6\xi_1 - 1/5, \\ \xi_1 = \xi_1. \end{cases} \quad (27)$$

The particular solution of (26) is

$$\begin{cases} \xi_7 = 137/60, \\ \xi_6 = -5, \\ \xi_5 = 5, \\ \xi_4 = -10/3, \\ \xi_3 = 5/4, \\ \xi_2 = -1/5, \\ \xi_1 = 0. \end{cases} \quad (28)$$

Therefore, the formula with consistency is

$$\dot{f}_k = \frac{1}{60\tau}(137 f_k - 300 f_{k-1} + 300 f_{k-2} - 200 f_{k-3} + 75 f_{k-4} - 12 f_{k-5}) + O(\tau^5). \quad (29)$$

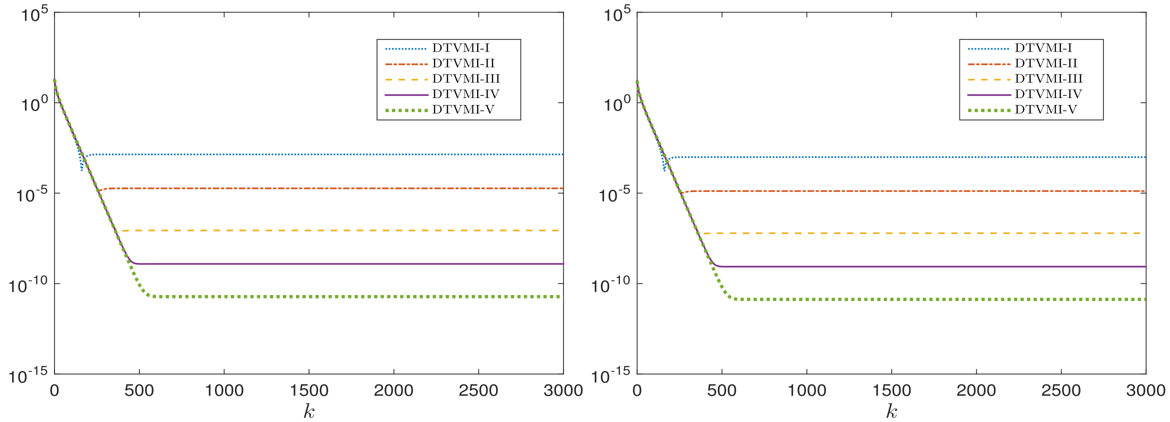
Next, the zero-stability of (29) is verified by the characteristic polynomial of (29) [32], [33]. The characteristic polynomial is shown as

$$\phi(\zeta) = 137\zeta^5 - 300\zeta^4 + 300\zeta^3 - 200\zeta^2 + 75\zeta - 12 = 0, \quad (30)$$

whose characteristic roots are $\zeta = 1$, $481/2290 \pm 2344/3463\mathrm{i}$, and $640/1663 \pm 107/660\mathrm{i}$. All the roots satisfy $|\zeta| \leq 1$ with $\zeta = 1$ being simple. The results satisfy the 0-stability condition. According to the Dahlquist equivalence theorem, (29) is also convergent with the order of its truncation error. The proof is thus completed. ■

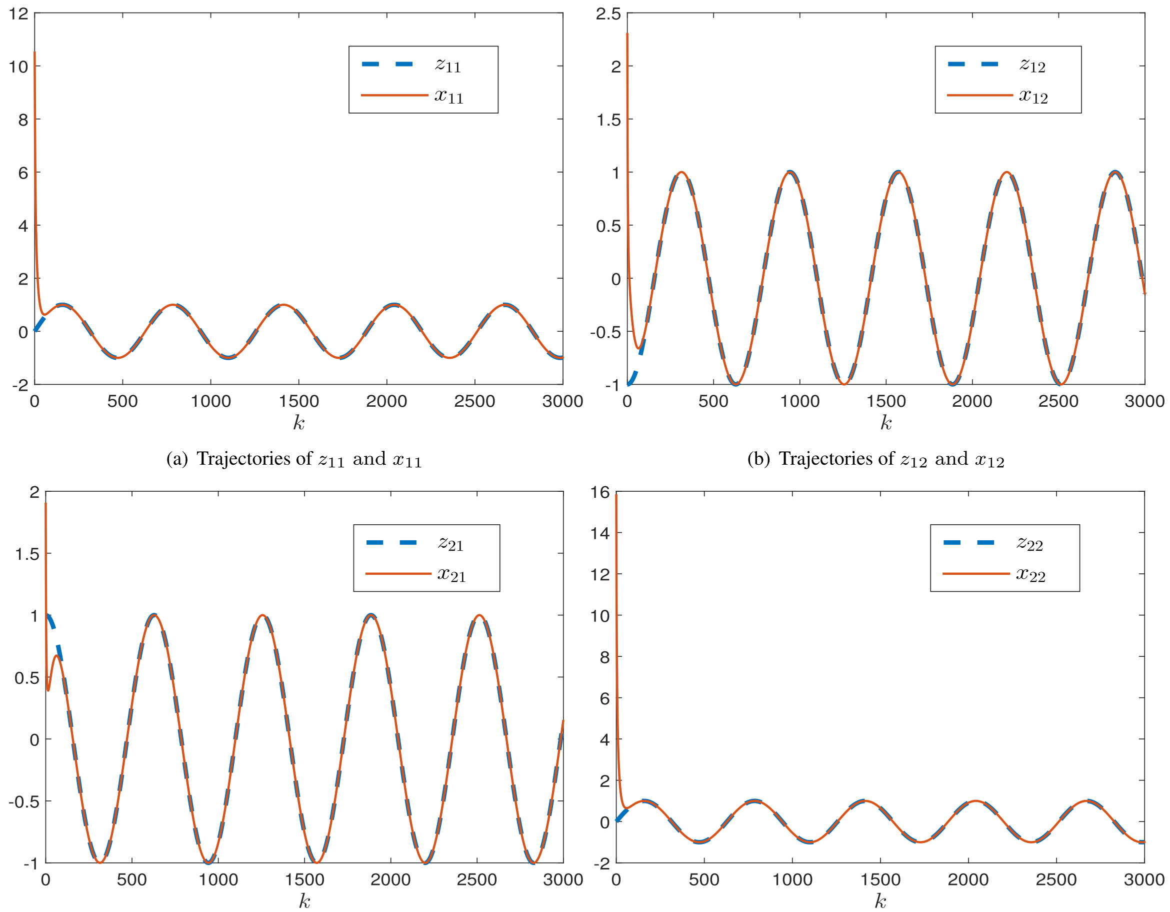
IV. NUMERICAL EXPERIMENTS AND VERIFICATION

In this section, numerical experiments are carried out to verify the efficacy of the presented GMDS-ZNN model 3 (5) on two time-variant matrix inversion examples. Furthermore, we verify the estimated $\dot{A}_k$ through the formula provided in (17).



(a) $\|X_k A_k - I\|_{\mathrm{F}}$ (b) $\|X_k - Z_k\|_{\mathrm{F}}$

FIGURE 1. **Residual error trajectories synthesized by five DTVMI algorithms for Example 1 ($\tau = 0.01$ s and $\gamma = 5.0$).**



(a) Trajectories of z_{11} and x_{11} (b) Trajectories of z_{12} and x_{12} (c) Trajectories of z_{21} and x_{21} (d) Trajectories of z_{22} and x_{22}

FIGURE 2. **Trajectories synthesized by Z_k and the solution of DTVMI-V algorithm** (15) **for Example 1 ($\tau = 0.01$ s and $\gamma = 5.0$).**

A. EXAMPLE 1

The first time-variant matrix is a 2×2 real matrix which is shown as follows:

$$A(t_k) = \begin{pmatrix} \sin(t_k) & \cos(t_k) \\ -\cos(t_k) & \sin(t_k) \end{pmatrix} \in \mathbb{R}^{2\times 2}. \quad (31)$$

To verify the computational results, we show the theoretical inversion of matrix (16):

$$Z(t_k) = A^{-1}(t_k) = \begin{pmatrix} \sin(t_k) & -\cos(t_k) \\ \cos(t_k) & \sin(t_k) \end{pmatrix} \in \mathbb{R}^{2\times 2}, \quad (32)$$

which is given for checking the correctness of the dynamic system solutions. In Fig. 1 (a), it shows the residual errors

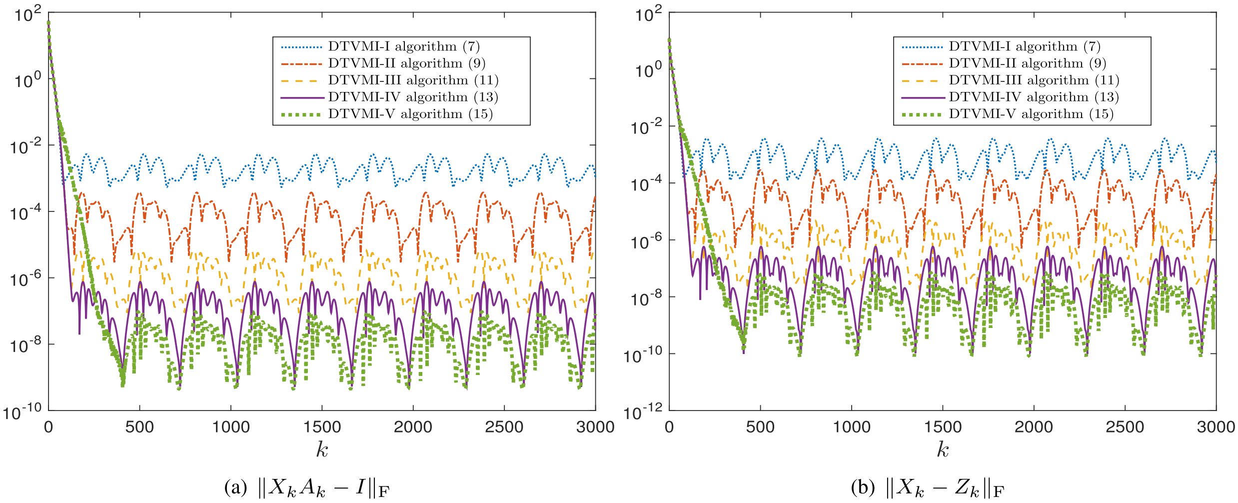


FIGURE 3. **Residual error trajectories synthesized by five DTVMI algorithms for Example 2 ($\tau = 0.01$ s and $\gamma = 1.0$).**

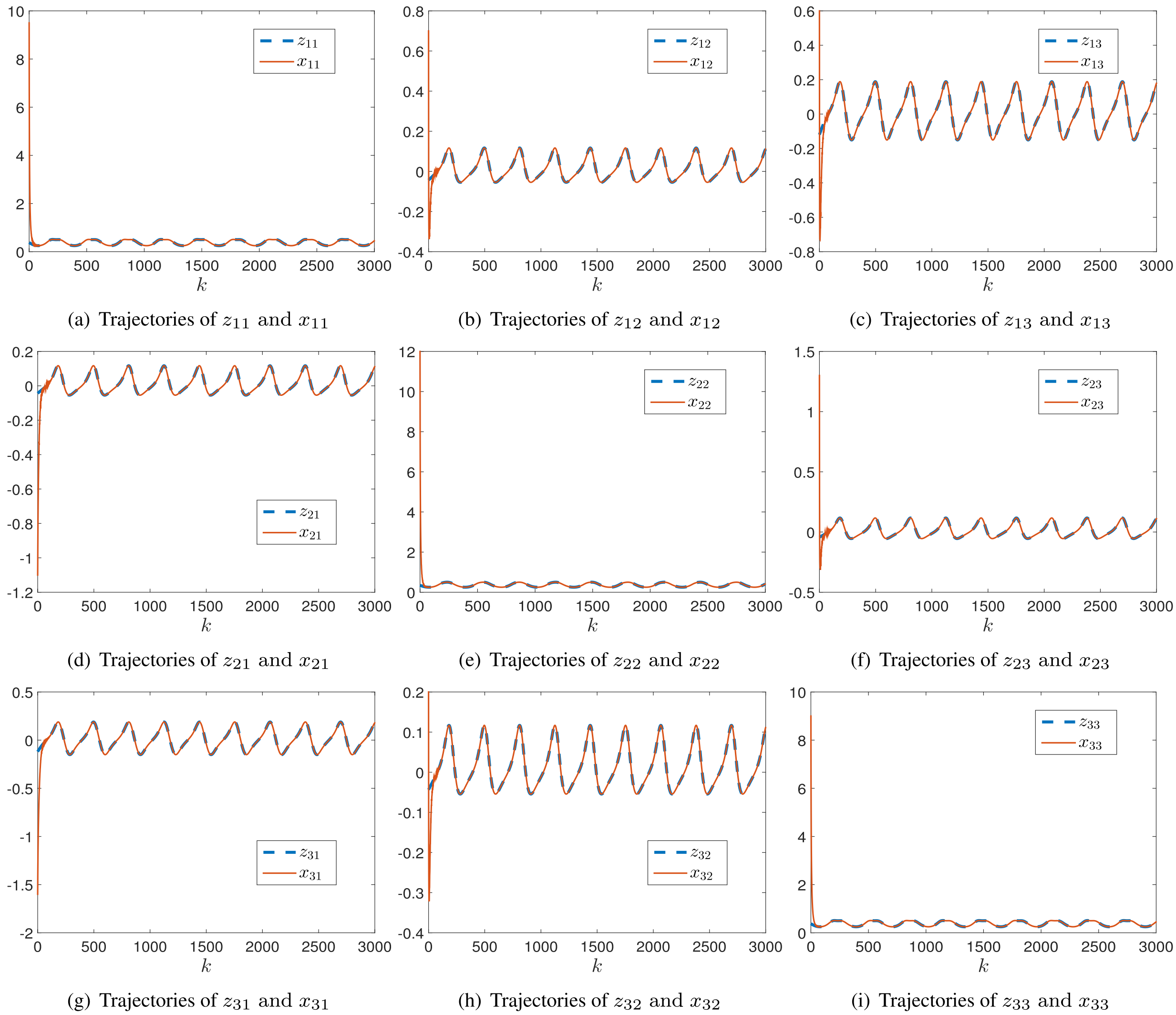


FIGURE 4. **Solution trajectories synthesized by Z_k and DTVMI-V algorithm (15) for Example 2 ($\tau = 0.01$ s and $\gamma = 1.0$).**

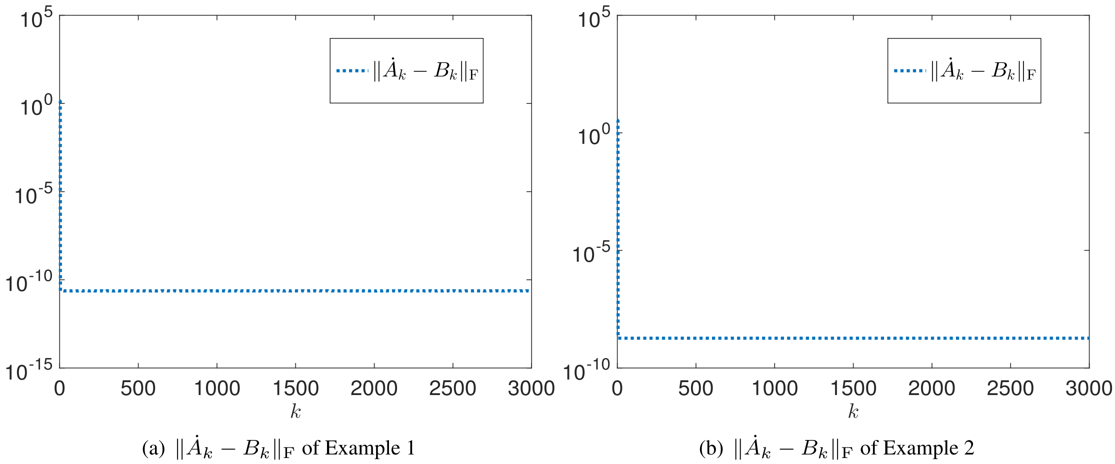


FIGURE 5. **Coefficient matrix derivative estimation errors $\|\dot{A}_k - B_k\|_F$.**

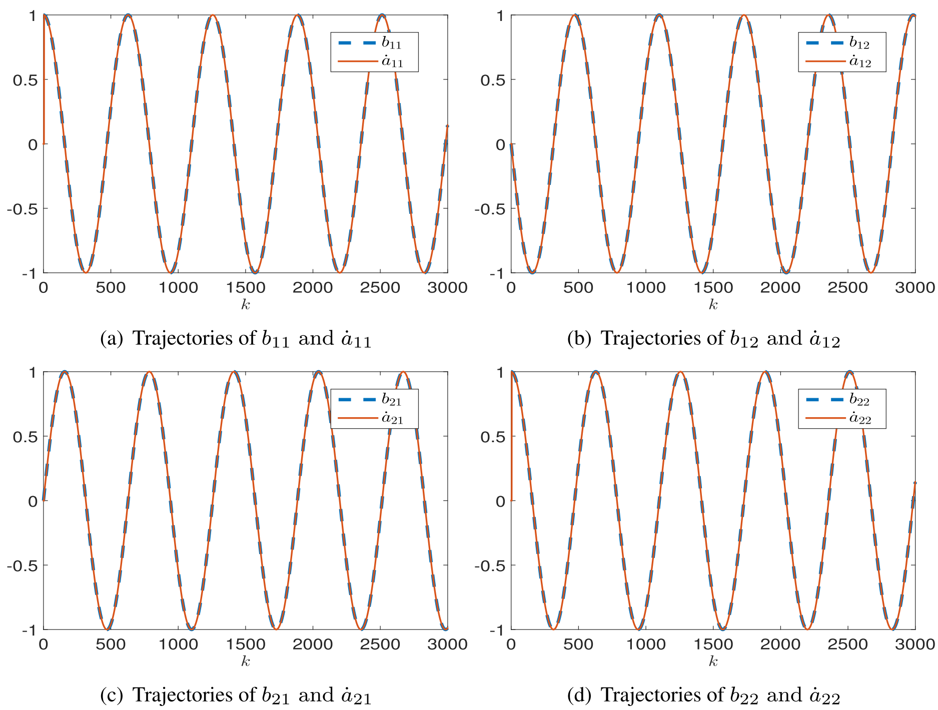
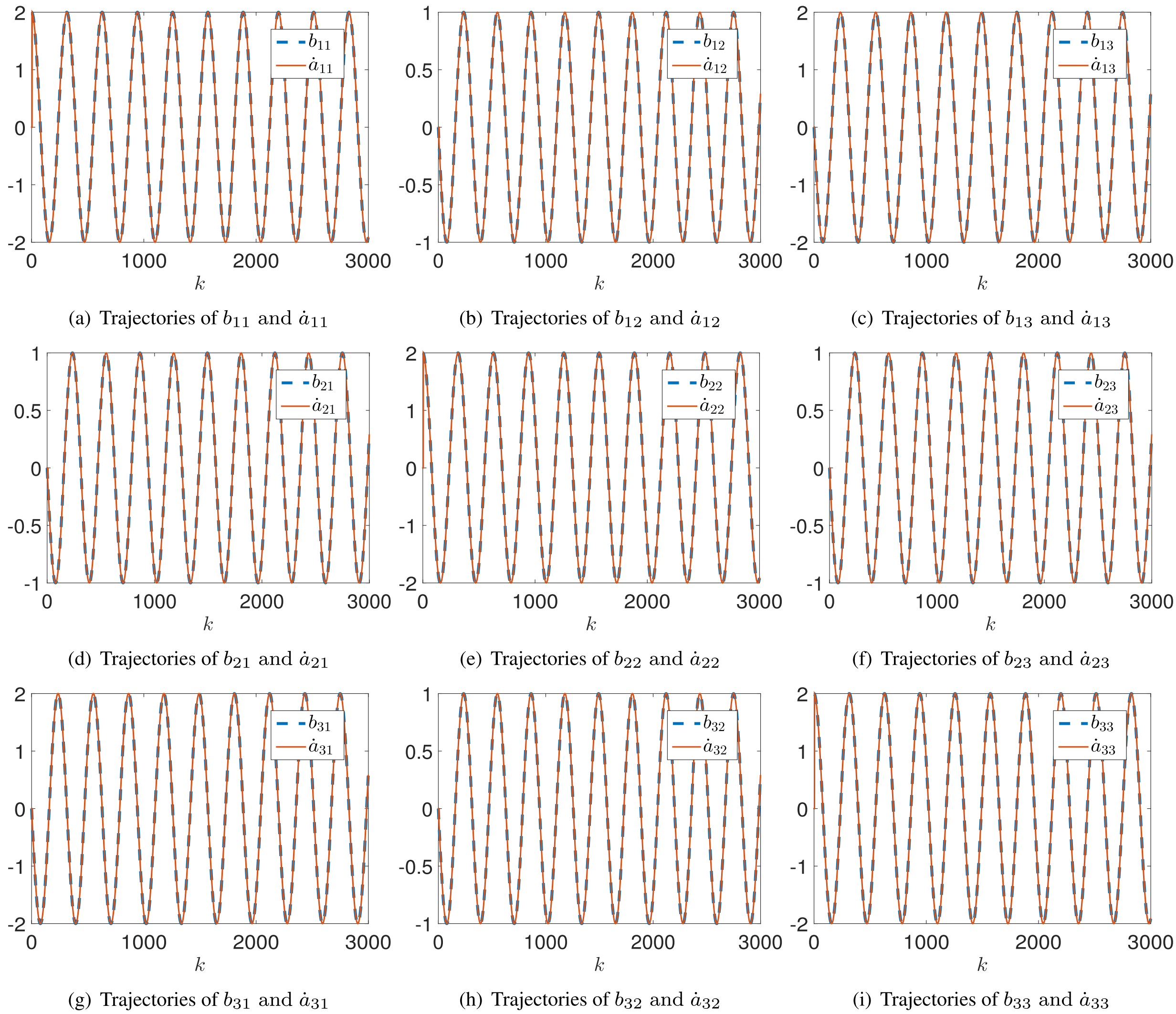


FIGURE 6. **Coefficient matrix derivative trajectories synthesized for Example 1.**

$\|X_k A_k - I\|_F$ of the five DTVMI algorithms, i.e., DTVMI-I algorithm (7), DTVMI-II algorithm (9), DTVMI-III algorithm (11), DTVMI-IV algorithm (13), and DTVMI-V algorithm (15), with $\tau = 0.01$ s and $h = 0.05$. In Fig. 1 (b), it illustrates the residual errors $\|X_k - Z_k\|_F$ of the same five DTVMI algorithms as before. The initial values of the example are arbitrarily set as

$$\begin{pmatrix} 10.5 & 2.3 \\ 1.9 & 15.8 \end{pmatrix}.$$

The step-size and task duration (i.e., final time) are uniformly set as $h = 0.05$ and $t_f = 30$ s. The values of the ten coefficients, which are applied in all these numerical experiments for DTVMI-V algorithm (15), are shown in the first row of Table 1. In Fig. 2, it depicts the numerical results of the DTVMI-V algorithm (15). From the trajectories of all entries in Fig. 2, the solution of the model coincides with the theoretical solution perfectly. In addition, we see that the residual error trajectories synthesized by different DTVMI algorithms quickly stabilized to the theoretical error level, after undergoing the initial hundreds of recursions. Moreover, to observe the variation law of residual errors synthesized by different DTVMI algorithms with different values of τ, with the help of MATLAB, more numerical experiments



(a) Trajectories of b_{11} and $\dot{a}_{11}$
(b) Trajectories of b_{12} and $\dot{a}_{12}$
(c) Trajectories of b_{13} and $\dot{a}_{13}$
(d) Trajectories of b_{21} and $\dot{a}_{21}$
(e) Trajectories of b_{22} and $\dot{a}_{22}$
(f) Trajectories of b_{23} and $\dot{a}_{23}$
(g) Trajectories of b_{31} and $\dot{a}_{31}$
(h) Trajectories of b_{32} and $\dot{a}_{32}$
(i) Trajectories of b_{33} and $\dot{a}_{33}$

FIGURE 7. **Coefficient matrix derivative trajectories synthesized for Example 2.**

have been carried out in Example 1. The result is exhibited in Table 2.

From Table 2, it is shown that the three values of r/τ for DTVMI-I algorithm (7) are approximately equal, which means the residual errors of (7) are almost directly proportional to $O(\tau^2)$ where r represents the residual error $\|X_kA_k - I\|_{\mathrm{F}}$. Similarly, from the values of r/τ^3, r/τ^4, r/τ^5, and r/τ^6 for (9), (11), (13), and (15), we can conclude that the residual errors of DTVMI-II algorithm (9), DTVMI-III algorithm (11), DTVMI-IV algorithm (13), and DTVMI-V algorithm (15), are almost directly proportional to $O(\tau^3)$, $O(\tau^4)$, $O(\tau^5)$ and $O(\tau^6)$, respectively. Thus, the correctness of Theorem 1 is further substantiated. Note that, in practical applications, we can obtain higher computational accuracy by simply setting the value of τ a little smaller. As shown in Table 2, when τ becomes smaller, such as $\tau = 0.008$ s, the accuracy of the calculation results of all DTVMI algorithms is improved, especially the magnitude of residual errors synthesized by DTVMI-V algorithm (15) is around 10^{-12}.

B. EXAMPLE 2

The second time-variant matrix is a 3×3 real matrix which is shown as follows:

$$A(t_k) = \begin{pmatrix} 3+\sin(2t_k) & 0.5\cos(2t_k) & \cos(2t_k) \\ 0.5\cos(2t_k) & 3+\sin(2t_k) & 0.5\cos(2t_k) \\ \cos(2t_k) & 0.5\cos(2t_k) & 3+\sin(2t_k) \end{pmatrix},$$

where $A(t_k) \in \mathbb{R}^{3\times 3}$. To verify the computational result, we propose the theoretical inversion of $A(t_k)$, which is named as $Z(t_k) = (z_{ij}(t_k))$. Because the $Z(t_k)$ is too complicated, it is omitted in this paper. The initial values of the example are arbitrarily set as

$$\begin{pmatrix} 9.5 & 0.7 & 0.6 \\ -1.1 & 12 & 1.3 \\ -1.6 & 0.2 & 9.0 \end{pmatrix}.$$

The step-size and task duration (i.e., final time) are uniformly set as $h = 0.01$ and $t_{\mathrm{f}} = 30$ s. The results of the numerical experiments for DTVMI-V algorithm (15), are

TABLE 2. Residual errors and ratios of DTVMI algorithms for Example 1 with different values of τ.

DTVMI Algorithm	τ (s)	r	r/τ	r/τ^2	r/τ^3	r/τ^4	r/τ^5	r/τ^6
	0.01	1.388E−03	1.388E−01	**1.388E+01**	1.388E+03	1.388E+05	1.388E+07	1.388E+09
DTTVMI-I algorithm (7)	0.009	1.128E−03	1.254E−01	**1.393E+01**	1.548E+03	1.720E+05	1.911E+07	2.123E+09
	0.008	8.943E−04	1.118E−01	**1.397E+01**	1.747E+03	2.183E+05	2.729E+07	3.411E+09
	0.01	1.849E−05	1.849E−03	1.849E−01	**1.849E+01**	1.849E+03	1.849E+05	1.849E+07
DTTVMI-II algorithm (9)	0.009	1.353E−05	1.503E−03	1.670E−01	**1.856E+01**	2.062E+03	2.291E+05	2.546E+07
	0.008	9.533E−06	1.192E−03	1.490E−01	**1.862E+01**	2.327E+03	2.909E+05	3.637E+07
	0.01	8.667E−08	8.667E−06	8.667E−04	8.667E−02	**8.667E+00**	8.667E+02	8.667E+04
DTTVMI-III algorithm (11)	0.009	5.707E−08	6.341E−06	7.046E−04	7.829E−02	**8.699E+00**	9.665E+02	1.074E+05
	0.008	3.575E−08	4.469E−06	5.586E−04	6.982E−02	**8.728E+00**	1.091E+03	1.364E+05
	0.01	1.220E−09	1.224E−07	1.224E−05	1.224E−03	1.224E−01	**1.224E+01**	1.224E+03
DTTVMI-IV algorithm (13)	0.009	7.257E−10	8.062E−08	8.958E−06	9.953E−04	1.106E−01	**1.229E+01**	1.365E+03
	0.008	4.040E−10	5.050E−08	6.312E−06	7.890E−04	9.863E−02	**1.233E+01**	1.541E+03
	0.01	1.897E−11	1.897E−09	1.897E−07	1.897E−05	1.897E−03	1.897E−01	**1.897E+01**
DTTVMI-V algorithm (15)	0.009	1.012E−11	1.125E−09	1.250E−07	1.388E−05	1.543E−03	1.714E−01	**1.905E+01**
	0.008	5.008E−12	6.260E−10	7.825E−08	9.782E−06	1.223E−03	1.528E−01	**1.910E+01**

Note: "r" represents the residual error $\|X_k A_k - I\|_\text{F}$, and "E" denotes "$\times 10$", such as "1.388E−03" denoting "1.388×10^{-03}".

shown in Fig. 3 and Fig. 4. The experimental results are as satisfactory as expected.

C. SITUATION OF $\dot{A}$ BEING UNKNOWN

In both Example 1 and Example 2, we apply (17) to verify the numerical accuracy of Theorem 2. Assume the ground-truth derivative of $\dot{A}(t)$ is $B(t) = (b_{ij}(t))$, whose discretized form is $B_k = (b_{ij}(t_k))$ when $t_k = k\tau$. Firstly, the ground-true derivative is calculated, so as to get B_k. Secondly, $\dot{A}_k$ is calculated by the formula provided in (17). Thirdly, the comparisons are made in two aspects. The first aspect is to observe the $\|\dot{A}_k - B_k\|_\text{F}$, which is depicted in Fig. 5. The second aspect is to compare every counterpart entry of B_k with $\dot{A}_k$. The results are shown in Fig. 6 and Fig. 7. According to these numerical results, the effectiveness and correctness of Theorem 2 are verified convincingly.

V. CONCLUSION

In this work, we present and investigate a general ten-instant ZeaD formula. In addition, the DTVMI-V algorithm (15) has been obtained by using the general ten-instant ZeaD formula to discretize GMDS-ZNN model 3 (5). The theoretical results are guaranteed the stability and convergence of the DTVMI-V algorithm (15) for time-variant matrix inversion. Furthermore, the simulation results have shown that all solutions of the DTVMI-V algorithm (15) are convergent to the theoretical solutions with the truncation error of $O(\tau^6)$, which is superior to the four other DTVMI algorithms. For the situation of $\dot{A}_k$ unknown, (17) has been provided and proved, which is capable of estimating $\dot{A}_k$ with the truncation error of $O(\tau^5)$. The cost is that five more temporary storage units for matrix A_{k-1} through A_{k-5} are needed to calculate the $\dot{A}_k$. The results of numerical simulation in both examples show that the convergence of $\|\dot{A}_k - B_k\|_\text{F}$ is satisfactory, reaching below 10^{-10} rapidly. At last, our future research direction may focus on the investigation of applying the ten-instant discrete algorithm to other time-variant problems solving.

REFERENCES

[1] R. J. Steriti and M. A. Fiddy, "Regularized image reconstruction using SVD and a neural network method for matrix inversion," *IEEE Trans. Signal Process.*, vol. 41, no. 10, pp. 3074–3077, Oct. 1993.

[2] N. Liu and S. Qin, "A novel neurodynamic approach to constrained complex-variable pseudoconvex optimization," *IEEE Trans. Cybern.*, vol. 49, no. 11, pp. 3946–3956, Nov. 2019.

[3] N. Liu and S. Qin, "A neurodynamic approach to nonlinear optimization problems with affine equality and convex inequality constraints," *Neural Netw.*, vol. 109, pp. 147–158, Jan. 2019.

[4] R. H. Sturges, "Analog matrix inversion (robot kinematics)," *IEEE J. Robot. Autom.*, vol. 4, no. 2, pp. 157–162, Apr. 1988.

[5] A. C. Tsoi and S. Tan, "Recurrent neural networks: A constructive algorithm, and its properties," *Neurocomputing*, vol. 15, nos. 3–4, pp. 309–326, Jun. 1997.

[6] K. Chen and C. Yi, "Robustness analysis of a hybrid of recursive neural dynamics for online matrix inversion," *Appl. Math. Comput.*, vol. 273, pp. 969–975, Jan. 2016.

[7] M. P. Kennedy and L. O. Chua, "Neural networks for nonlinear programming," *IEEE Trans. Circuits Syst.*, vol. 35, no. 5, pp. 554–562, May 1988.

[8] X. Liao and J. Yu, "Robust stability for interval hopfield neural networks with time delay," *IEEE Trans. Neural Netw.*, vol. 9, no. 5, pp. 1042–1045, Sep. 1998.

[9] S. Qin and X. Xue, "A two-layer recurrent neural network for nonsmooth convex optimization problems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1149–1160, Jun. 2015.

[10] S. Qin, X. Yang, X. Xue, and J. Song, "A one-layer recurrent neural network for pseudoconvex optimization problems with equality and inequality constraints," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3063–3074, Oct. 2017.

[11] Y. Zhang, D. Jiang, and J. Wang, "A recurrent neural network for solving Sylvester equation with time-varying coefficients," *IEEE Trans. Neural Netw.*, vol. 13, no. 5, pp. 1053–1063, Sep. 2002.

[12] B. Liao and Y. Zhang, "Different complex ZFs leading to different complex ZNN models for time-varying complex matrix inversion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 9, pp. 1621–1631, Sep. 2014.

[13] I. Stojanovic, P. Stanimirovic, I. Zivkovic, D. Gerontitis, and X.-Z. Wang, "ZNN models for computing matrix inverse based on hyperpower iterative methods," *Filomat*, vol. 31, no. 10, pp. 2999–3014, 2017.

[14] Z. Zhang, X. Deng, X. Qu, B. Liao, L.-D. Kong, and L. Li, "A varying-gain recurrent neural network and its application to solving online time-varying matrix equation," *IEEE Access*, vol. 6, pp. 77940–77952, 2018.

[15] Z. Qi and Y. Zhang, "New models for future problems solving by using ZND method, correction strategy and extrapolation formulas," *IEEE Access*, vol. 7, pp. 84536–84544, 2019.

[16] N. H. Getz and J. E. Marsden, "Dynamical methods for polar decomposition and inversion of matrices," *Linear Alg. Appl.*, vol. 258, pp. 311–343, Jun. 1997.

[17] Y. Zhang, K. Chen, and H.-Z. Tan, "Performance analysis of gradient neural network exploited for online time-varying matrix inversion," *IEEE Trans. Autom. Control*, vol. 54, no. 8, pp. 1940–1945, Aug. 2009.
[18] D. Guo and Y. Zhang, "Zhang neural network, Getz-Marsden dynamic system, and discrete-time algorithms for time-varying matrix inversion with application to robots' kinematic control," *Neurocomputing*, vol. 97, pp. 22–32, Nov. 2012.
[19] J. H. Mathews and K. K. Fink, *Numerical Methods Using MATLAB*. Englewood Cliffs, NY, USA: Prentice-Hall, 2004.
[20] Y. Zhang, L. Jin, D. Guo, Y. Yin, and Y. Chou, "Taylor-type 1-step-ahead numerical differentiation rule for first-order derivative approximation and ZNN discretization," *J. Comput. Appl. Math.*, vol. 273, pp. 29–40, Jan. 2015.
[21] C. Hu, X. Kang, and Y. Zhang, "Three-step general discrete-time zhang neural network design and application to time-variant matrix inversion," *Neurocomputing*, vol. 306, pp. 108–118, Sep. 2018.
[22] J. Li, G. Wu, C. Li, M. Xiao, and Y. Zhang, "GMDS-ZNN variants having errors proportional to sampling gap as compared with models 1 and 2 having higher precision," in *Proc. 5th Int. Conf. Syst. Informat. (ICSAI)*, Nanjing, China, Nov. 2018, pp. 728–733.
[23] Y. Zhang, Y. Xie, and H. Tan, "Time-varying Moore-Penrose inverse solving shows different Zhang functions leading to different ZNN models," in *Proc. 9th Interface Conf. Adv. Neural Netw.*, Shenyang, China, Jul. 2012, pp. 98–105.
[24] D. Zhang, P. He, N. Wang, B. Liao, and Y. Zhang, "GZ-type complex dynamic system solving online for time-varying complex matrix inverse," in *Proc. IEEE Int. Conf. Inf. Autom. (ICIA)*, Zhejiang, China, Aug. 2016, pp. 154–158.
[25] Y. Zhang and S. S. Ge, "Design and analysis of a general recurrent neural network model for time-varying matrix inversion," *IEEE Trans. Neural Netw.*, vol. 16, no. 6, pp. 1477–1490, Nov. 2005.
[26] Y. Zhang and J. Wang, "Global exponential stability of recurrent neural networks for synthesizing linear feedback control systems via pole assignment," *IEEE Trans. Neural Netw.*, vol. 13, no. 3, pp. 633–644, May 2002.
[27] Y. Zhang and C. Yi, Zhang, *Neural Networks and Neural-Dynamic Method*. New York, NY, USA: Nova Science, 2011.
[28] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*. New York, NY, USA: Springer, 2002.
[29] Y. Zhang, H. Gong, M. Yang, J. Li, and X. Yang, "Stepsize range and optimal value for Taylor-Zhang discretization formula applied to zeroing neurodynamics illustrated via future equality-constrained quadratic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 959–966, Mar. 2019.
[30] D. Guo, Z. Nie, and L. Yan, "Novel discrete-time Zhang neural network for time-varying matrix inversion," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 2301–2310, Aug. 2017.
[31] J. Chen, J. Guo, and Y. Zhang, "General ten-instant DTDMSR model for dynamic matrix square root finding," *Cybern. Syst.*, pp. 1–7, Oct. 2020. [Online]. Available: https://www.tandfonline.com/action/doSearch?AllField=10.1080%2F01969722.2020.1827794&SeriesKey=ucbs20, doi: 10.1080/01969722.2020.1827794.
[32] E. Suli and D. F. Mayers, *An Introduction to Numerical Analysis*. Cambridge, U.K.: Cambridge Univercity Press, 2003.
[33] D. F. Griffiths and D. J. Higham, *Numerical Methods for Ordinary Differential Equations: Initial Value Problems*. London, U.K.: Springer, 2010.

DONGQING WU received the M.S. degree in computer graphics from the Institute of Industrial Design and Graphics, South China University of Technology, Guangzhou, China, in 2005, and the Ph.D. degree in mechanical engineering from the School of Electromechanical Engineering, Guangdong University of Technology, Guangzhou, in 2019. He is currently a Visiting Scholar with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou. He is also an Associate Professor with the School of Computational Science, Zhongkai University of Agriculture and Engineering, Guangzhou. His current research interests include neural networks, geometry algorithm, and pattern recognition.

YUNONG ZHANG (Member, IEEE) received the B.S. degree in industrial electrical automation from the Huazhong University of Science and Technology, Wuhan, China, in 1996, the M.S. degree in control theory and control engineering from the South China University of Technology, Guangzhou, China, in 1999, and the Ph.D. degree in mechanical and automation engineering from the Chinese University of Hong Kong, Hong Kong, China, in 2003. He is currently a Professor with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou. Before joining Sun Yat-sen University, in 2006, he has been with the National University of Singapore, the University of Strathclyde, and the National University of Ireland at Maynooth, since 2003. His main research interests include robotics, neural networks, computation, and optimization.

JINJIN GUO received the B.E. degree in measurement technology and instrument from Nanchang University, Nanchang, China, in 2016, and the M.E. degree in control engineering from Sun Yat-sen University, Guangzhou, China, in 2018, where she is currently pursuing the Ph.D. degree in computer science and technology with the School of Computer Science and Engineering. Her main research interests include neural networks, numerical computation, and tracking control.

ZHENYU LI received the B.S. degree in computer science and technology from Sun Yat-sen University, Guangzhou, China, in 2019, where he is currently pursuing the M.S. degree in computer technology. His current research interests include time-varying problems solving, neural networks, and intelligent computing.

LIANGJIE MING received the B.S. degree in communication engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2018. He is currently pursuing the M.S. degree in detection technology and automatic equipment with Sun Yat-sen University, Guangzhou, China. His current research interests include neural networks, robotics, and computation.