Two DTZNN Models of $O(\tau^4)$ Pattern for Online Solving Dynamic System of Linear Equations: Application to Manipulator Motion Generation

ZHONGBO SUN$^{1,2}$, YONGBAI LIU$^2$, LIN WEI$^3$, KEPING LIU$^2$, LONG JIN$^3$, (Member, IEEE), AND LUQUAN REN$^1$

$^1$Key Laboratory of Bionic Engineering of Ministry of Education, Jilin University, Changchun 130025, China
$^2$Department of Control Engineering, Changchun University of Technology, Changchun 130012, China
$^3$School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China

Corresponding author: Keping Liu (liukeping@ccut.edu.cn)

The work was supported in part by the National Natural Science Foundation of China under Grant 61873304, Grant 11701209, and Grant 51875047, in part by the China Postdoctoral Science Foundation Funded Project under Grant 2018M641784 and Grant 2019T120240, in part by the Key Science and Technology Projects of Jilin Province, China, under Grant 20190302025GX, Grant 20170204067GX, and Grant 20180201105GX, in part by the Industrial Innovation Special Funds Project of Jilin Province, China, under Grant 2018C038-2, in part by the Jilin Engineering Laboratory for Intelligence Robot and Visual Measurement Technology under Grant 2019C010, and in part by the Fundamental Research Funds for the Central Universities under Grant lzujbky-2019-89.

ABSTRACT In this paper, firstly, a high accuracy one-step-ahead numerical differentiation formula with $O(\tau^4)$ pattern is proposed for discretization. Meanwhile, a high precision first-order derivative formula of the backward difference rule with $O(\tau^4)$ pattern error is given to approximate the derivative information. Then, two high accuracy discrete-time zeroing-type models (HADTZTM) with $O(\tau^4)$ pattern, i.e., HADTZTM with derivative information known (HADTZTM-K) and HADTZTM with derivative information unknown (HADTZTM-U), are developed, analyzed and investigated for online solving the dynamic system of linear equations (DSLEs). In addition, the 0-stability, consistency, and convergence of the HADTZTM-K and HADTZTM-U are verified for DSLEs. From a theoretical/numerical viewpoint, the classical models are revisited and analyzed for online solving DSLEs. Ultimately, simulation experiment including an application to the path-tracking of the four-link planar manipulator is conducted to demonstrate the efficiency and superiority of the HADTZTM-K and HADTZTM-U, where the HADTZTM-U overcomes the difficulty of derivative information unknown in practical applications.

INDEX TERMS Dynamic system of linear equations (DSLEs), discrete-time zeroing neural network (DTZNN), backward difference formula, theoretical results, steady-state residual error.

I. INTRODUCTION

The problem of solving linear equations has been widely researched by many scholars in the scientific and engineering fields as a fundamental mathematical one, including research on rapidity and accuracy of solutions under overdetermined, under-determined and normal conditions. Therefore, there are many applications for solving linear equation problems, for instance, robotics [1], [2], machine learning [3], optimization [4] etc. Many traditional methods such as Newton-Raphson iteration [5], gradient-related methods [3] and algorithms based on homotopic perturbation [6], [7], by reason of these proposed algorithms do not utilize time-varying derivative information in the design process. Therefore, these approaches are theoretically devoted to solve the static (or time-invariant) linear equations. Note that there are a large number of practical problems which should be formulated as dynamic (or time-varying) system of linear equations (DSLEs) [8]. That is to say, static solvers cannot exhibit satisfactory performance when applied to solve dynamic linear equations. Although a few algorithms have been developed to solve time-varying linear equations [14], the existing ones still have large errors in the application of DSLEs with high time-dependence such as robots [9], localization [10], and signal processing [11], etc. Therefore, research on the stability, accuracy and rapidity of algorithms

The associate editor coordinating the review of this manuscript and approving it for publication was Yan-Jun Liu.
still have a wide range of significance. In this paper, two high accuracy neural-network-based discrete-time models are designed to solve DSLEs with time-varying coefficients. The theoretical analysis and example simulations demonstrate the superior performance of the models in real-world applications.

In recent years, neural networks have been widely applied in various fields, both in science and engineering [12], [13], [14]–[19]. As a significant application, the kinematic control of robot redundant is frequently solved by the neural network method [20]. Aiming at eliminating the joint drift phenomenon of robot manipulator, Xie et al. have exploited the gradient descent method to construct the dynamic neural network, which would compel joint angle error to rapidly converge to zero [21]. This method has been extended to the research of the manipulator with unknown structure and achieved remarkable results [22]. In addition, according to the game theory, the manipulability optimization method of manipulators has been proposed in [23], which is assisted by a specially designed recurrent neural network with the great noise resistance. In practical applications, considering the convenience of algorithm implementation in hardware, Ferreira et al. have creatively proposed a classic gradient-based neural network (GNN) with parallel processing capabilities to solving the DSLEs [3]. However, although GNN has the potential capacity to handle problems in parallel, it just be theoretically and essentially designed for solving the static problems. Therefore, Zhang et al. proposed a new zeroing neural network (ZNN) based on recurrent neural network for online solving the mathematical problem in the engineering and science [24]. Based on the strong parallel processing ability and exponential convergence advantages of ZNN models in scientific applications, Zhang et al. have utilized several different functions to design different continuous-time zeroing-type neural network models (CTZTM) for solving DSLEs with time-varying coefficients [25], [26]. But there is no doubt that the similar continuous models can hardly be directly realized on digital computers and applied in real-world applications. Therefore, in order to facilitate the applications on hardware and make up for the shortcomings of the lack of accuracy of existing models, two discrete-time ZNN based models are designed for online solving DSLEs in this paper.

In this paper, a high accuracy first-order derivative approximation formula is firstly proposed and used for discretization. Sequentially, a novel high accuracy discrete-time zeroing-type neural network model with known derivative information (HADTZTM-K) is designed, analyzed and investigated for online solving DSLEs. Meanwhile, the 0-stability, consistency and convergence of the HADTZTM-K are theoretically proved for the first time. Two numerical examples and an illustrative application to a four-link planar redundant manipulator verify that the HADTZTM-K has a high accuracy with the order of residual error $O(\tau^4)$ pattern. However, for some real-world applications, the derivative information of HADTZTM-K will be difficult to know or obtain [27]. To overcome this disadvantage, the backward-difference rule with $O(\tau^4)$ pattern precision will be used to approximate the derivative information in the HADTZTM-K [28], and then a novel high accuracy discrete-time zeroing-type model with unknown derivative information (HADTZTM-U) is developed for online solving DSLEs in practical applications. The theoretical analysis and numerical simulation results demonstrate the efficiency and superiority of the proposed HADTZTM-K and HADTZTM-U for online solving DSLEs.

The remainder of this paper is organized as follows. In Section II, the problem formulation of DSLEs and a new Taylor-type difference rule are introduced and analyzed to design the models of HADTZTM-K and HADTZTM-U as a foundation. Section III develops, investigates and theoretically proves the superiority of HADTZTM-K and HADTZTM-U for online solving DSLEs. In Section IV, three models, including the discrete-time Newton-Raphson-type model (DTNTM), the discrete-time Euler-type neural network model (DTETM), and the discrete-time zeroing-type model (DTZTM) are presented for comparison with the HADTZTM-K and HADTZTM-U. To demonstrate the superior performance of the proposed models, two numerical examples and a four-link planar manipulator example are illustrated in Section V. In the end, Section VI concludes the future works of this paper. The main contributions of this paper are as follows.

1) This paper proposes two discrete-time zeroing type models, which overcome the shortcomings of traditional solvers that can only solve static problems, and facilitate hardware implementation.

2) Theoretical analyses and results demonstrate that the proposed HADTZTM-K and HADTZTM-U models with the higher accuracy of $O(\tau^4)$ pattern are superior compared with DTNTM, DTETM and DTZTM for online solving DSLEs.

3) Two numerical simulation results, together with an illustrative application to a four-link planar manipulator, which substantiate the efficacy and superiority of the proposed HADTZTM-K and HADTZTM-U for online solving DSLEs in real time.

II. PROBLEM FORMULATION AND PRELIMINARIES

To build the foundation for further investigation, the problem formulation of DSLEs is mathematically presented in this section. Meanwhile, a high accuracy first-order derivative approximation with the $O(\tau^3)$ truncation error pattern is given as the basis for the HADTZTM-K and HADTZTM-U.

A. PROBLEM FORMULATION

Considering the mathematically problem formulation of the continuous time-varying DSLEs:

$$A(t)x(t) = b(t), \quad t \in [0, +\infty).$$

(1)

where the coefficient matrix $A(t) \in \mathbb{R}^{n \times n}$ is defined as a measurable, full column or row rank matrix with time-varying coefficients, $b(t) \in \mathbb{R}^n$ denotes a time-varying vector, and
\(X(t) \in \mathbb{R}^n\) signifies the unknown time-varying solutions of (1) to be obtained.

As a basis for the design of discrete-time models, the corresponding discrete-time form of the DSLEs (1) deserves to be mentioned as follows:

\[A_{k+1}X_{k+1} = B_{k+1},\]

where matrices \(A_{k+1}\) and \(B_{k+1}\) represent the sample value of \(A(t)\) and \(B(t)\) at sampling time \(t = (k + 1)\tau\), respectively; \(\tau\) denotes the sampling interval; \(X_{k+1}\) will be obtained in the \([k\tau, (k + 1)\tau)\). Therefore, utilizing the previous or real-time data, the time-varying solution \(X_{k+1}\) of linear equations will be updated online in each computational interval.

**B. HIGH ACCURACY TAYLOR-TYPE DIFFERENCE RULE**

To obtain a higher accuracy discretization formula in comparison with the Euler-type and Newton-Raphson-type models, a novel effective \(O(\tau^3)\) formula is generalized for first-order derivative estimation. Furthermore, to compensate the lack of accuracy of existing models, the higher precision first-order derivative approximation formula with \(O(\tau^3)\) pattern will be utilized for discretization. According the general first-order derivative approximation formula in [29]

\[\dot{f}_k = \frac{1}{\xi \tau} (\sum_{j=1}^{6} c_j f_{k-j+2}) + O(\tau^m),\]  

where \(f_{k+i} = f((k + i)\tau); c_j \neq \mathbb{R}, (j = 1, 2, \ldots n)\) and \(\xi \neq 0 \in \mathbb{R}\) are the coefficients; \(O(\tau^m)\) is the truncation error. The left-hand side first-order derivative \(\dot{f}_k\) will be discretized into six parts, which include \(f_{k+1}, f_k, f_{k-1}, f_{k-2}, f_{k-3}, f_{k-4}\). By using the Taylor expansion and solving the system of linear equations in Appendix A, a novel high accuracy Taylor-type first-order derivative approximation formula is designed to guarantee that the HADTZTM-K and HADTZTM-U have high precision solution with \(O(\tau^4)\) pattern in software and hardware applications. In Theorem 1, the \(S^m[a, b]\) signifies the set of all functions \(f\) and its first \(r\)th derivatives are continuous on interval \([a, b]\).

**Theorem 1:** Assume that \(f \in S^4[a, b]\) and that \(t_{k-4}, t_{k-3}, t_{k-2}, t_{k-1}, t_k, t_{k+1} \in [a, b]\), then

\[\dot{f}_k \approx \frac{24\dot{f}_{k+1} - 5\dot{f}_k - 12\dot{f}_{k-1} - 6\dot{f}_{k-2} - 4\dot{f}_{k-3} + 3\dot{f}_{k-4}}{48\tau}\]  

which has a truncation error of \(O(\tau^3)\).

**Proof:** The detailed certification process is shown in Appendix A.

**C. CONTINUOUS-TIME ZEROING-TYPE NEURAL NETWORK MODEL**

As a preliminary foundation for the establishment of HADTZTM-K and HADTZTM-U, in [25], [26], considering the over-determined, under-determined and normal conditions of DSLEs (2), the CTZTM is designed by using zeroing formula and error functions. Combined with the above high accuracy first-order derivative approximation formula (4), a discrete-time zeroing-type model is obtained to solve DSLEs in real time. According to the design process in [17], [26], the design steps of the CTZTM model is as follows.

**Step 1:** Define the vector-formed error function as the zero finding function which is

\[e(t) = A(t)X(t) - B(t).\]

**Step 2:** Choose the zeroing dynamics design formula (ZDF) and activation function as follows:

\[\dot{e}(t) = \frac{de(t)}{dt} = -\gamma F(e(t)).\]

Here, the linear activation function \(F(e(t)) = e(t)\) will be utilized to simplify the ZDF. Therefore, the ZDF can be

\[\dot{e}(t) = -\gamma e(t).\]

**Step 3:** Combining the equation (5) with (6), the CTZTM (7) can be obtained to online solving the DSLEs (2).

\[\dot{X}(t) = A^+(t)(\dot{B}(t) - A(t)X(t) - \gamma(A(t)\dot{X}(t) - B(t))),\]

where the parameter \(\gamma\) can be selected in accordance with the performance of the hardware, which is designed to ensure high convergence rate of the CTZTM (7). On the premise of (1), the \(A^+(t) = (A^T(t)A(t))^{-1}A^T(t)\) represents the Moore-Penrose inverse of \(A(t)\).

**Theoerem 2:** The problem formulation and preliminaries have been introduced for the design of HADTZTM-K and HADTZTM-U.

**III. DISCRETE-TIME ZEROING-TYPE MODELS AND THEORETICAL ANALYSES**

In this section, it is seriously considered that the case where the derivative information is known and unknown, and the lack of accuracy of the existing model. Then, two novel high accuracy discrete-time zeroing-type neural network models with derivative information known and unknown, termed as HADTZTM-K and HADTZTM-U, are developed, analyzed and investigated for online solving DSLEs (2).

**A. HADTZTM-K AND HADTZTM-U**

Utilizing the Taylor-type difference rule to discretize the CTZTM (7) model, the HADTZTM-K and HADTZTM-U will be designed for online solving DSLEs (2).

Firstly, based on high accuracy first-order derivative approximation formula (4), the equation (4) can be described as follows.

\[\dot{X}_k = \frac{24X_{k+4} - 5X_k - 12X_{k-1} - 6X_{k-2} - 4X_{k-3} + 3X_{k-4}}{48\tau}\]

which has a truncation error of \(O(\tau^3)\).

Assuming that the derivative information is known in CTZTM (7), then, the formula (8) will be employed to discretize the CTZTM (7) model. As a result of discretization,
the HADTZTM-K is
\[
X_{k+1} = \frac{5}{24}X_k + \frac{1}{2}X_{k-1} + \frac{1}{4}X_{k-2} + \frac{1}{6}X_{k-3} - \frac{1}{8}X_{k-4} + 2A_k^+(\tau B_k - \tau A_k X_k - h(A_k X_k - B_k)),
\]
where the parameter \( h = \gamma \tau > 0 \in \mathbb{R} \) signifies the step size, which needs to be set in accordance with the performance of HADTZTM-K (9) in the simulation process. However, the \( A_k \) and \( B_k \) of the model (9) may be difficult to be acquired directly in real-world applications. Thus, in [28], to overcome the situation of the unknown derivative information, the first-order derivative formula (10) of the backward-difference rule is applied to approximate the derivative information of the model (9). Thereby, the HADTZTM-U can be described as below:
\[
X_{k+1} = \frac{5}{24}X_k + \frac{1}{2}X_{k-1} + \frac{1}{4}X_{k-2} + \frac{1}{6}X_{k-3} - \frac{1}{8}X_{k-4} + 2A_k^+(\frac{25}{12}B_k - 4B_{k-1} + 3B_{k-2} - \frac{4}{3}B_{k-3} - h(A_k X_k - B_k)),
\]
where step size \( h = \gamma \tau > 0 \in \mathbb{R} \), and \( \tau \) denotes the sampling gap.

So far, two high accuracy discrete-time zeroing-type neural network models have been designed for online solving the DSLEs. Compared with the traditional continuous-time zeroing-type model, the proposed HADTZTM-K (9) and HADTZTM-U (11) can be widely applied to the digital hardware. Meanwhile, two novel discrete-time models, which not only simultaneously combine higher precision and better real-time performance than the existing discrete-time models, but also overcome the challenging problem that the derivative information is difficult to be obtained in real-world applications.

B. THEORETICAL ANALYSIS AND RESULTS

Based on several definitions described below, the proposed HADTZTM-K (9) and HADTZTM-U (11) are certified and analyzed for the 0-stability, consistence, and convergence as follows.

Definition 1: An N step model \( \sum_{i=0}^{N} \alpha_i X_{k+i} = \tau \sum_{i=0}^{N} \beta_i X_{k+i} \) can be proved to be 0-stability by calculating and checking the roots of characteristic polynomial \( P_N(\delta) = \sum_{i=0}^{N} \alpha_i \delta^i \), if the roots of \( P_N(\delta) = 0 \) lie in/on the unit disk (i.e., \(|\delta| \leq 1 \) and \(|\delta| = 1 \)).

Definition 2: An N step model can be testified to be consistent with order \( m \), if the model has a truncation error of \( O(\tau^m) \) when finding a smoothly exact solution, where \( m > 0 \).

Definition 3: An N step model is convergent, i.e., \( \lim_{\tau \to 0} X^*(\tau) = X^*(\tau) \) for all \( \tau \to 0 \), if and only if the model is consistent and 0-stability.

According to the above definitions, the theorems about the HADTZTM-K (9) and HADTZTM-U (11) models are proved.

Theorem 2: The HADTZTM-K (9) is 0-stable.

Proof: Based on the Definition 1, the characteristic polynomial of HADTZTM-K (9) can be described as
\[
P(\delta) = \delta_5 - \frac{5}{24} \delta_4 - \frac{1}{2} \delta_3 - \frac{1}{4} \delta_2 - \frac{1}{6} \delta + \frac{1}{8} = 0,
\]
where the five roots can be calculated as follows:
\[
\delta_1 = 1, \quad \delta_2 = -0.7627, \quad \delta_3 = 0.3833, \quad \delta_4 = -0.2062 + 0.6206i, \quad \delta_5 = -0.2062 - 0.6206i,
\]
where all the characteristic roots are located in/on the unit disk, and \( i \) signifies the imaginary unit. Therefore, according to the Definition 1, the proof about 0-stability of HADTZTM-K (9) is accomplished.

Theorem 3: The HADTZTM-K (9) is consistent and convergent, which converges with the 4-order of the \( O(\tau^4) \) pattern residual error for all \( \forall t_k \in [t_0, t_f] \).

Proof: According to the Definition 2, Definition 3 and Theorem 1, equation (8) can be rewritten as
\[
X_k = \frac{24X_{k+1} - 5X_k - 12X_{k-1} - 6X_{k-2}}{48\tau} = \frac{-4X_{k-3} - 3X_{k-4}}{48\tau} + O(\tau^3). \quad (12)
\]
Utilizing the equation (12) to discretize the CTZTM (7), the HADTZTM-K (9) with the \( O(\tau^4) \) pattern residual error can be developed as
\[
X_{k+1} = \frac{5}{24}X_k + \frac{1}{2}X_{k-1} + \frac{1}{4}X_{k-2} + \frac{1}{6}X_{k-3} - \frac{1}{8}X_{k-4} + 2A_k^+(\tau B_k - \tau A_k X_k - h(A_k X_k - B_k)) + O(\tau^4). \quad (13)
\]
Comparing equation (13) and equation (9), it can be observed that the HADTZTM-K (9) can be obtained if the \( O(\tau^4) \) pattern truncation error of (13) is dropped. Therefore, based on the Definition 2, HADTZTM-K (9) is consistent with 4-order of the \( O(\tau^4) \) pattern truncation error of (13). Then, according to Definition 3, the HADTZTM-K (9) can be certified to be consistent and convergent, which converges with the 4-order of \( O(\tau^4) \) truncation error for all \( t_k \in [t_0, t_f] \).

Theorem 4: HADTZTM-K (9) has \( O(\tau^4) \) order of steady-state residual error \( \lim_{\tau \to 0} ||A_k X_k - B_k||_F \) in solving the problem of DSLEs (2), where \(|\cdot|_F \) denotes the Frobenius norm of a matrix.
**Proof:** Let $X_k^* \in \mathbb{R}^n$ be the exact solution of (2), based on the above Definitions, the solution of the HADTZTM-K (9) can be described as $X_k = X_k^* + O(\tau^4)$ when $k \to \infty$. Thus

$$||A_k X_k - B_k||_F = ||A_k (X_k^* + O(\tau^4)) - B_k||_F = ||A_k X_k^* - B_k + A_k O(\tau^4)||_F.$$  

On account of $X_k^*$ being the exact solution of (2), thus, $A_k X_k^* = B_k$. Sequentially, the above results can be further given by

$$||A_k X_k - B_k||_F = ||A_k O(\tau^4)||_F = O(\tau^4).$$

Therefore, the steady-state residual error of HADTZTM-K (9) is

$$\lim_{k \to \infty} ||A_k X_k - B_k||_F = O(\tau^4).$$

Hereto, the proof is completed.

Similarly, due to Definition 1, Definition 2 and Definition 3, the theoretical analyses of the HADTZTM-U (11) are developed and investigated as follows.

**Theorem 5:** The HADTZTM-U (11) is 0-stable.

**Proof:** In light of the Definition 1, the characteristic polynomial of HADTZTM-U (11) is listed below

$$P_5(\delta) = \delta^5 - \frac{5}{24}\delta^4 - \frac{1}{2}\delta^3 - \frac{1}{4}\delta^2 + \frac{1}{6}\delta + \frac{1}{8} = 0$$

where the five roots can be seen as follows, $\delta_1 = 0.3833$, $\delta_2 = 1$, $\delta_3 = -0.7627$, $\delta_4 = -0.2062 + 0.6206i$, and $\delta_5 = -0.2062 - 0.6202i$, which satisfy the conditions of Definition 1. Thus, the 0-stable theoretical analysis of the HADTZTM-U (11) is finished.

**Theorem 6:** The HADTZTM-U (11) is consistent and convergent, which has a $O(\tau^4)$ pattern residual error and convergent to the 4-order of $O(\tau^4)$.

**Proof:** According to the previous Definitions, employing the equation (12) to discretize the CTZTM model (7) and utilizing the equation (10) to estimate the unknown derivative information. Therefore, the HADTZTM-U (11) with $O(\tau^4)$ pattern residual error can be given by

$$X_{k+1} = \frac{5}{24} X_k + \frac{1}{2} X_{k-1} + \frac{1}{4} X_{k-2} + \frac{1}{6} X_{k-3} - \frac{1}{8} X_{k-4}$$

$$+ 2A_k^+(\frac{25}{12} B_k - 4B_{k-1} + 3B_{k-2} - \frac{4}{3} B_{k-3})$$

$$+ \frac{1}{4} B_{k-4} + O(\tau^4) - (\frac{25}{12} A_k - 4A_{k-1} - 3A_{k-2} - \frac{4}{3} A_{k-3} + \frac{1}{4} A_{k-4} + O(\tau^4))X_k$$

$$- h(A_k X_k - B_k)) + O(\tau^4).$$  

(14)

The formula (14) can be further rewritten as

$$X_{k+1} = \frac{5}{24} X_k + \frac{1}{2} X_{k-1} + \frac{1}{4} X_{k-2} + \frac{1}{6} X_{k-3} - \frac{1}{8} X_{k-4}$$

$$+ 2A_k^+(\frac{25}{12} B_k - 4B_{k-1} + 3B_{k-2} - \frac{4}{3} B_{k-3})$$

$$+ \frac{1}{4} B_{k-4} - (\frac{25}{12} A_k - 4A_{k-1} + 3A_{k-2} - \frac{4}{3} A_{k-3} + \frac{1}{4} A_{k-4} + O(\tau^4))X_k$$

$$- h(A_k X_k - B_k)) + O(\tau^4).$$  

(15)

where the term $2A_k^+(1 - X_k) k A_k^+ (1 - X_k) O(\tau^4)$ can be absorbed by $O(\tau^4)$, therefore, the term $2A_k^+ (1 - X_k) O(\tau^4)$ will change in accordance with the $O(\tau^4)$ pattern. It means that the term $2A_k^+ (1 - X_k) O(\tau^4)$ can be rewritten as $O(\tau^4)$. In summary, the HADTZTM-U (11) has a truncation error of $O(\tau^4)$ pattern and convergent to the 4-order of the $O(\tau^4)$ pattern residual error for all $t_k \in [t_0, t_f]$. Dropping the error term of the (15), the HADTZTM-U (11) will be obtained to solving the problem (2). Based on the Definition 3, the proof about consistency and convergence of the HADTZTM-U (11) are completed.

To sum up, the 0-stability, consistency and convergence of the HADTZTM-K (9) and HADTZTM-U (11) have been investigated for online solving DSLEs (2). From the perspective of theoretical analysis, the proposed two novel models not only show higher accuracy, but also possess outstanding real-time performance when solving the DSLEs. In order to demonstrate the superior performance of the above two models in the numerical simulations, three comparative models are developed, analyzed and investigated in the following sections.

**IV. DTZTM, DTETM AND DTNTM**

In this section, according to [14], the theoretical analysis and results of the DTZTM, DTETM and DTNTM models are presented for contrasting with the models of HADTZTM-K (9) and HADTZTM-U (11) for online solving DSLEs (2).

**A. DTZTM**

Utilizing the taylor-type first-order derivative approximation (16) with $O(\tau^2)$ pattern to discretize CTZTM (7), the DTZTM will be obtained as follows.

$$\dot{X}_k = \frac{2X_{k+1} - 3X_k + 2X_{k-1} - X_{k-2}}{2\tau} + O(\tau^2)$$  

(16)

which has an $O(\tau^2)$ pattern truncation error, thus, the DTZTM with error terms can be given by

$$X_{k+1} = 1.5X_k - X_{k-1} + 0.5X_{k-2} + A_k^+(\tau \dot{B}_k - \dot{A}_k X_k$$

$$- h(A_k X_k - B_k)) + O(\tau^3).$$  

(17)

Getting rid of the $O(\tau^3)$ term on the right of (17), the DTZTM can be generalized as

$$X_{k+1} = 1.5X_k - X_{k-1} + 0.5X_{k-2} + A_k^+(\tau \dot{B}_k - \dot{A}_k X_k$$

$$- h(A_k X_k - B_k))$$  

(18)

where the parameter $h = \gamma\tau$ is the step size and $\tau$ is the sampling gap.

**Theorem 7:** The DTZTM (18) is 0-stable, consistent and convergent, which has a residual error of $O(\tau^3)$ pattern and converges with the 3-order of $O(\tau^3)$ pattern residual error.
**Proof:** According to [14] and Definitions 1, the roots of characteristic polynomial \( P_3(\delta) = \delta^3 - 1.5\delta^2 + \delta - 0.5 \) of the (18) are located in the unit circle, where \( \delta_1 = 0.25 + 0.6614i, \delta_2 = 0.25 - 0.6614i \) and \( \delta_3 = 1 \). Based on the Definitions 2-3, it can be summarized as the model of DTZTM (18) is 0-stable, consistent and convergent to the 3-order of \( O(\tau^3) \) pattern residual error for all \( t_k \in [t_0, t_f] \).

**B. DTETM AND DTNTM**

Owing to [28], the Euler-type first-order derivative approximation formula with \( O(\tau) \) pattern will be selected to discretize the CTZTM (7), thus, the DTETM can be expressed as

\[
\dot{X}_k = \frac{X_{k+1} - X_k}{\tau} + O(\tau)
\]

where \( h = \tau \gamma \) is the step size. Note that the DTNTM can be obtained if the parameter \( h = 1 \) and the derivative information \( A_k \) and \( B_k \) are dropped from (19), which is given by

\[
\dot{X}_{k+1} = X_k - A_k^T (A_k X_k - B_k)
\]

Contrastively, it can be summarized as the DTNTM (20) is a particular case of the DTETM (19). Here, it is worth declaring that the DTZTM (18), DTETM (19) and DTNTM (20) are presented for comparing with the HADTZTM-K (9) and HADTZTM-U (11). Due to the page limit, it is unnecessary to analyze the theory of the three comparative models in detail. Therefore, for the presented DTETM (19) and DTNTM (20), the detailed proof of the following theorems 8-9 can be found in [14].

**Theorem 8:** The DTETM (19) is 0-stable, consistent and convergent, which has a residual error of \( O(\tau^2) \) and converges with the 2-order of \( O(\tau^3) \) pattern residual error when \( t_k \in [t_0, t_f] \).

**Theorem 9:** The DTNTM (20) is 0-stable, consistent and convergent, which the residual error is \( O(\tau) \) pattern for all \( t_k \in [t_0, t_f] \).

In conclusion, DTZTM (18), DTETM (19) and DTNTM (20) have different solution precision which are \( O(\tau^3), O(\tau^2) \) and \( O(\tau) \), respectively, in solving discrete-time DSLEs. Obviously, the existing highest accuracy of the \( O(\tau^3) \) pattern cannot guarantee the high accuracy in practical applications. In this and the previous section, the 0-stability, consistency, and convergence of the proposed five models have been analyzed for online solving DSLEs (2). However, it is more necessary to verify the performance of the proposed models in real-world applications. Therefore, in the next section, the previous models will be utilized to online solving the DSLEs involved in Example 1-3. For the reader’s convenience and comparison, the aforementioned five models are listed in Table 1.

| Model          | Formula                                                                 |
|----------------|-------------------------------------------------------------------------|
| HADTZTM-K      | \[
X_{k+1} = \frac{5}{24} X_k + \frac{1}{2} X_{k-1} + \frac{1}{2} X_{k-2} + \frac{1}{8} X_{k-3} - \frac{1}{8} X_{k-4} - 2A_k^T (\tau B_k - \tau A_k X_k - h(A_k X_k - B_k))
\] |
| HADTZTM-U      | \[
X_{k+1} = \frac{5}{24} X_k + \frac{1}{2} X_{k-1} + \frac{1}{2} X_{k-2} + \frac{1}{8} X_{k-3} - \frac{1}{8} X_{k-4} + 2A_k^T (\tau B_k - 4B_k - 3B_{k-2} - 2B_{k-3} - 5B_{k-4} - \frac{25}{2} A_k - A_{k+1} + 3A_k - 2A_k - 3A_k - 4A_{k+1}) X_k - h(A_k X_k - B_k))
\] |
| DTZTM          | \[
X_{k+1} = 1.5X_k - X_{k-1} + 0.5X_{k-2} + A_k^T (\tau B_k - \tau A_k X_k - h(A_k X_k - B_k))
\] |
| DTETM          | \[
X_{k+1} = X_k - A_k^T (\tau B_k - \tau A_k X_k - h(A_k X_k - B_k))
\] |
| DTNTM          | \[
X_{k+1} = X_k - A_k^T (A_k X_k - B_k)
\] |

**V. NUMERICAL SIMULATIONS AND APPLICATION**

In this section, two numerical simulations and an illustrative application to a four-link planar manipulator are provided to further verify the excellent performance of the proposed HADTZTM-K (9) and HADTZTM-U (11) for online solving the DSLEs. Meanwhile, the presented DTZTM (18), DTETM (19) and DTNTM (20) will also be applied to compare with the proposed HADTZTM-K (9) and HADTZTM-U (11).

It is worth pointing out that the initial value of the model is an important part in the simulation process. For the proposed models, HADTZTM-K (9) and HADTZTM-U (11) require five initial states to start the iteration, while the DTZTM (18) need three initial states. In order to choose a more reasonable initial value to compensate for the impact of initial value setting on model performance, based on the initial state \( X_0 \), the DTETM (19) will be utilized to generate four states \( \{X_1, X_2, X_3, X_4\} \) as the initial states for HADTZTM-K (9), HADTZTM-U (11) and DTZTM (18). The four initial states can be obtained as follows

\[
\begin{align*}
X_1 &= X_0 + A_k^T (\tau B_0 - \tau A_0 X_0 - h(A_0 X_0 - B_0)) \\
X_2 &= X_1 + A_k^T (\tau B_1 - \tau A_1 X_1 - h(A_1 X_1 - B_1)) \\
X_3 &= X_2 + A_k^T (\tau B_2 - \tau A_2 X_2 - h(A_2 X_2 - B_2)) \\
X_4 &= X_3 + A_k^T (\tau B_3 - \tau A_3 X_3 - h(A_3 X_3 - B_3))
\end{align*}
\]

Hence, \( X_0, X_1, X_2 \) will be the initial states of DTZTM (18), and the HADTZTM-K (9) and HADTZTM-U (11) will iterate from five initial states \( \{X_0, X_1, X_2, X_3, X_4\} \). In addition, by reason of the derivative information of the two numerical simulations can be obtained by MATLAB software. Therefore, HADTZTM-K will be adopted to solving the two numerical simulations involved through Example 1 to 2 compared with DTZTM (18), DTETM (19) and DTNTM (20). Then, HADTZTM-U (11) will be employed to deal with the path-tracking problem of the four-link planar manipulator involved in Example 3.
Two DTZNN Models of $O((\tau^4))$ Pattern for Online Solving Dynamic System of Linear Equations

A. NUMERICAL EXAMPLE 1

In this example, considering the linear equations under normal conditions ($m = n$), the $A_k$ and $B_k$ can be given by

$$A_k = \begin{bmatrix} 3 + \sin(5t_k) & \cos(5t_k) & 0.5 \cos(5t_k) \\ \cos(5t_k) & 3 + \sin(5t_k) & \cos(5t_k) \\ 0.5 \cos(5t_k) & \cos(5t_k) & 3 + \sin(5t_k) \end{bmatrix} \in \mathbb{R}^{3 \times 3}$$

$$B_k = \begin{bmatrix} \cos(5t_k) & \sin(5t_k) & -\sin(5t_k) \end{bmatrix}^T \in \mathbb{R}^3$$

The above four models (9), (18), (19) and (20) will be applied to solve the DSLEs involved in Example 1, where the parameter $h = 0.5$ and the sampling gap is set to $\tau = 0.01s$, 0.001s and 0.0001s. Then, starting from five randomly generated initial state $X_0 \in [-1.2, 1.2]^3$, the corresponding numerical results are shown in Figure. 1. Assuming the theoretical solution is unknown, as can be seen from the Figure. 1 that the four discrete-time models are iterated from five different initial state, and the corresponding solution will converge together in a certain time. However, under the premise that the theoretical solution is unknown, the results in Figure. 1 cannot be directly considered as theoretical solutions. Therefore, combining with the steady-state residual errors of the models in Figure. 2, it can be summed up as the above four
models converge to the theoretical solution with different precision.

In order to demonstrate the performance of the model in solving the DSLEs, the residual errors $\|e(k\tau)\|_F = \|A_kX_k - B_k\|_F$ is utilized to reflect the performance of the model. The residual error of the models can be seen in Figure 2. In addition, the maximal steady-state residual errors (MSSREs) of the above models are listed in Table 2 for further comparison. As shown in Figure 2(b), the HADTZTM-K (9) achieves the highest convergence order of $10^{-8}$ pattern, which is higher than those of DTZTM (18), DTETM (19) and DTNTM (20) ($10^{-6}, 10^{-4}$ and $10^{-2}$). Comparing the Figure 2(a) with Figure 2(b) and Figure 2(c), it can be observed that steady-state residual errors of HADTZTM-K (9), DTZTM (18), DTETM (19) and DTNTM (20) change in the pattern of $O(\tau^4)$, $O(\tau^2)$ and $O(\tau)$, respectively. Thus, based on the Fig. 1, Figure 2 and Table 2, the performance of the above four models can be summarized as follows.

1) Because there is no derivative information $\dot{A}_k$ and $\dot{B}_k$ in the DTNTM (20), the steady-state residual error of the presented DTNTM (20) changes in an $O(\tau)$ pattern as the sampling gap $\tau$ decreases from $\tau = 0.01s$, to $0.001s$ and $0.0001s$. It can be seen from Fig. 1(d) that the results are converge to the theoretical solution in the form of an $O(\tau^{-3})$ pattern residual error.

2) For the presented DTETM (19), its steady-state residual error changes in the pattern of $O(\tau^2)$ as the sampling gap $\tau$ changes by 10 times. As shown in Figure 1(c), the results are converge to the theoretical solution in the form of an $O(\tau^{-5})$ pattern residual error.

3) For the comparison model DTZTM (18), its steady-state residual error is order of $10^{-5}, 10^{-8}, 10^{-11}$ as the sampling gap $\tau$ decreases from $\tau = 0.01s$, to $0.001s$ and $0.0001s$. As it can be seen in Figure 1(b), the results are converge to the theoretical solution in the form of an $O(\tau^{-5})$ pattern residual error.

4) The steady-state residual error of the proposed HADTZTM-K (9) changes in the pattern of $O(\tau^4)$ as the sampling gap $\tau$ changes from $\tau = 0.01s$, to $0.001s$ and $0.0001s$. Specifically, Figure 1(a) reveals that the results of HADTZTM-K (9) will converge to the theoretical solution in the form of an $O(\tau^4)$ pattern.
Theoretical solution in the form of an \(O(\tau^{-10})\) pattern residual error.

In summary, the computational performance of HADTZTM-K (9) is superior to DTZTM (18), DTETM (19) and DTNTM (20). Table 2 reveals that the MSSREs of the above four models decrease with the parameter \(h\) increasing from \(h = 0.1\) to 0.6. Moreover, as the sampling gap changes by 10 times, the steady-state residual error of the proposed HADTZTM-K (9) not only can be changed in an \(O(\tau^{-4})\) pattern, but also the average computing time per updating is less than sampling gap of \(\tau = 10^{-2}\) s, \(\tau = 10^{-3}\) s and \(\tau = 10^{-4}\) s. In other words, HADTZTM-K (9), a higher accuracy solver, can be utilized to solving the DSLEs in real-time.

**B. NUMERICAL EXAMPLE 2**

In this example, the following under-determined DSLEs with \(m = 8\) and \(n = 9\) are designed to test the proposed HADTZTM-K (9) as:

\[
A_k = \begin{bmatrix}
a_{11}(t_k) & a_{12}(t_k) & \ldots & a_{1n}(t_k) \\
a_{21}(t_k) & a_{22}(t_k) & \ldots & a_{2n}(t_k) \\
& \vdots & & \vdots \\
a_{m1}(t_k) & \ldots & \ldots & a_{mn}(t_k)
\end{bmatrix} \in \mathbb{R}^{8 \times 9}
\]

\[
B_k = \begin{bmatrix} b_1(t_k) & b_2(t_k) & \ldots & b_m(t_k) \end{bmatrix}^T \in \mathbb{R}^8
\]

where the \(a_{mn}(t_k)\) and \(b_m(t_k)\) are

\[
a_{mn}(t_k) = \begin{cases} 
m + \sin(t_k), & m=n \\
\cos(t_k)/(m-n), & m>n \\
\cos(t_k)/(n-m), & m<n
\end{cases}
\]

\[
b_m(t_k) = \begin{cases} 
\cos(t_k), & m \text{ is odd} \\
\sin(t_k), & m \text{ is even}
\end{cases}
\]

In this example, the models of HADTZTM-K (9), DTZTM (18), DTETM (19) and DTNTM (20) will be selected to solve the above DSLEs. Starting from five randomly generated initial state \(X_0 \in [-1.2, 1.2]^9\) and \(h = 0.3\), the corresponding numerical results are shown in Figure 3. The numerical simulation results of HADTZTM-K (9) with \(h = 0.3\) and \(\tau = 0.001 s\) are displayed in Figure 3. Assuming the theoretical solution to be unknown, as we can see from the Figure 3 that HADTZTM-K (9) starts from five initial states, and the corresponding solutions will also coincide together in a certain period of time. Similarly, the steady-state error \(|e(k\tau)|_F = \|A_kX_k - B_k\|_F\) in Figure 4 will be calculated to illustrate the solution of the HADTZTM-K (9) can converge to the theoretical solution. Due to space limitations, the numerical simulation results of

**FIGURE 3.** State trajectories of the HADTZTM-K (9) using \(h = 0.3\) and \(\tau = 0.001 s\) to solve for the DSLEs involved in Example 2.
the DTZTM (18), DTETM (19) and DTNTM (20) to online solving the time-varying DSLEs involved in Example 2 are not shown in this section.

According to the Figure. 3, Figure. 4 and Table 3, the performance of the above four models can be summarized as follows.

1) For the presented DTNTM (20), with $h = 1$ and $\tau = 0.01s, 0.001s, 0.0001s$, its steady-state residual error changes in the pattern of $O(\tau^{-4})$. Looking at Table 3, with $\tau = 0.01s, 0.001s$ and $0.0001s$ the numerical simulation result in Figure. 3 will converge to the theoretical solution in the form of an $O(\tau^{-11})$ pattern residual error.

2) For the comparison DTZTM (18), its steady-state residual error changes in an $O(\tau^{-3})$ pattern as the sampling gap $\tau$ changes by 10 times, which is greater than the order of $O(\tau^{-4})$. Similarly, it can be summarized from Table 3 that the simulation results of DTZTM (18) with $\tau = 0.001s$ and $h = 0.3$ will converge to the theoretical solutions in the form of an $O(\tau^{-3})$ pattern residual error.

3) The steady-state residual error of the HADTZTM-K (9) with $h = 0.2, 0.3$ changes in the pattern of $O(\tau^{-4})$ as the sampling gap $\tau$ decreases from $\tau = 0.01s, 0.001s$ and $0.0001s$. In addition, the numerical simulation result in Figure. 3 will converge to the theoretical solution in the form of an $O(\tau^{-11})$ pattern residual error.

To sum up, the computational performance of the HADTZTM-K (9) is superior to the model of DTZTM (18), DTETM (19) and DTNTM (20). In addition, it can be seen from Table 3 that the MSSREs of the DTZTM (18), DTETM (19) and DTNTM (20) will decrease with the parameter $h$ changing from $h = 0.1$ to $0.6$. However, what is different from Example 1 is that the steady-state residual error of the HADTZTM-K (9) will not decrease gradually with the parameter $h$ increasing from $h = 0.1$ to $h = 0.6$ for online solving the DSLEs involved in Example 2. Note
follows:

that, only in the case of $h = 0.2$ and $h = 0.3$, the MSSREs of the HADTZTM-K (9) changes in the pattern of $O(\tau^3)$ as the sampling gap $\tau$ changes by 10 times. That means choosing an appropriate $h$ is an important step when the HADTZTM-K (9) is used to solve under-determined DSLEs. Therefore, it can be concluded from multiple numerical simulations that the HADTZTM-K (9) has outstanding performance at $h = 0.3$. Furthermore, the average computing time per updating $10^{-5}$ is less than sampling gap of $\tau = 10^{-2}s$, $\tau = 10^{-3}s$ and $\tau = 10^{-4}s$. In conclusion, the HADTZTM-K (9) can be utilized to solving the under-determined DSLEs with higher-accuracy in real-time.

C. APPLICATION ON PLANAR MANIPULATOR

In this example, the proposed HADTZTM-K (9) and HADTZTM-U (11) are applied to a four-link planar manipulator for path-tracking. In fact, the path-tracking problem of a planar manipulator can be transformed into solving the DSLEs online. From the [14], [30], it can be obtained as follows:

$$J(q(t))\dot{q}(t) = \dot{r}_d(t) + k(r_d(t) - \Phi(q(t))) \quad (21)$$

where the $q(t)$ and $\dot{q}(t)$ denote the joint-angle vector and velocity vector, and $J(q(t)) = \partial\Phi(q(t))/\partial q(t) \in \mathbb{R}^{2 \times 4}$ is the Jacobian matrix of the four-link planar manipulator, where the $\Phi(\cdot)$ is the forward kinematics function. The term of $\dot{r}_d(t) + k(r_d(t) - \Phi(q(t)))$ signifies the end-effector position vector with position-error feedback, where the $r_d(t)$ and $\dot{r}_d(t)$ represent the desired end-effector position vector and velocity vector, and the parameter $k \in \mathbb{R}^+$ is the position-error feedback gain. Meanwhile, the $r(t)$ can be calculated by utilizing the formula $\Phi(q(t)) = r(t)$. In this application, the four-link planar manipulator is applied to track a three-leaf-clover path. The task will start from the initial joint-angle vector $q(0) = [0, \pi/2, 0, \pi/2]^T$ rad and complete the clover trajectory tracking during $T = 10s$. In addition, the parameters can be selected as follows: step size is $h = 0.3$, sampling gap is $\tau = 0.001s$ and position-error feedback gain is $k = 15$.

As shown in Figure. 5, specifically, Figure. 5(a) shows the motion trajectories synthesized by the proposed HADTZTM-K (9). It can be seen from Figure. 5(b) and Figure. 5(c) that the joint-angle vector $q$ and velocity vector $\dot{q}$ synthesized by HADTZTM-K (9) are stable and smooth. Due to the space limitation, it is worth mentioning that the motion trajectories, joint-angle vector $q$ and velocity vector $\dot{q}$ of the HADTZTM-U (11), DTZTM (18) are similar to the Figure. 5(a)-5(c), therefore, it will be omitted.

In addition, from a qualitative perspective, see Figure. 5(d)-5(f), the position-error vector of the HADTZTM-K (9) and HADTZTM-U (11) are smaller than the DTZTM (18) with $h = 0.3$. Note that the HADTZTM-K (9) has excellent performance when solving the under-determined DSLEs involved in Example 2 at $h = 0.3$, therefore, the parameters can be chosen as $h = 0.3$ and $\tau = 0.001s$ to complete the three-leaf-clover path tracking of the
TABLE 4. The maximal position-error of the HADTZTM-K (9), HADTZTM-U (11) and DTZTM (18) models with different \( h \) when the four-link planar manipulator with its end-effector tracking a three-leaf-clover path.

| Model       | Step size | \( \epsilon_x \)     | \( \epsilon_y \)      |
|-------------|-----------|-----------------------|-----------------------|
| HADTZTM-K   | \( h=0.1 \) | 5.2466 x 10^{-5}      | 5.6217 x 10^{-5}      |
|             | \( h=0.2 \) | 2.8236 x 10^{-5}      | 3.0582 x 10^{-5}      |
|             | \( h=0.3 \) | 2.0163 x 10^{-5}      | 2.2032 x 10^{-5}      |
| HADTZTM-U   | \( h=0.1 \) | 5.1379 x 10^{-5}      | 5.8148 x 10^{-5}      |
|             | \( h=0.2 \) | 2.7626 x 10^{-5}      | 3.1534 x 10^{-5}      |
|             | \( h=0.3 \) | 1.9715 x 10^{-5}      | 2.2664 x 10^{-5}      |
| DTZTM       | \( h=0.1 \) | 1.5086 x 10^{-4}      | 1.6166 x 10^{-4}      |
|             | \( h=0.3 \) | 5.967 x 10^{-5}       | 6.3346 x 10^{-5}      |
|             | \( h=0.5 \) | 3.903 x 10^{-5}       | 4.3675 x 10^{-5}      |
|             | \( h=0.7 \) | 3.1448 x 10^{-5}      | 3.5244 x 10^{-5}      |
|             | \( h=0.9 \) | 2.7030 x 10^{-5}      | 3.0560 x 10^{-5}      |
|             | \( h=1 \)  | 2.5483 x 10^{-5}      | 2.8921 x 10^{-5}      |

VI. CONCLUSION

In this paper, firstly, it is subtly designed the high accuracy Taylor-type difference rule (4) and the first-order derivative of the backward difference rule with \( O(\tau^4) \) pattern residual error. Then, HADTZTM-K (9) and HADTZTM-U (11) are proposed, analyzed and investigated for online solving the DSLEs (2) in comparison with DTZTM (18), DTETM (19) and DTNTM (20). In addition, two numerical examples have illustrated that the HADTZTM-K (9) is superior to the three existing comparative models, and then an illustrative application to a four-link planar manipulator for path-tracking has also demonstrated that HADTZTM-K (9) has the minimum position-error. It meant that, when the derivative information is known, HADTZTM-K (9) has high solution accuracy, stability and real-time performance in both numerical simulations and real-world applications. Considering the derivative information may be difficult to obtain in practical applications, HADTZTM-U (11) has also been applied to the path-tracking task of the four-link planar manipulator. The numerical results have proved that HADTZTM-U (11) still has excellent performance when dealing with the case where the derivative information is unknown.

For the future direction, the performance of the proposed models in a noisy environment could be discussed. Therefore, it will be further improved that the accuracy and anti-interference performance of the models should be considered and analyzed for the DSLEs. In addition, it is worth...
mentioning that the proposed two high accuracy discrete-time models will also be utilized to online solving zero finding problems [31–33], rehabilitation robots [34], trajectory planning [35], [36] and switched nonlinear systems [37–40] etc.

APPENDIX A

1. Proof of Theorem 1.

According to the literature [29], the \( \dot{f}_k \) in formula (3) will be discretized via \( f_{k+1}, f_k, f_{k-1}, f_{k-2}, f_{k-3}, f_{k-4} \). Assume that \( f \in S^4[a, b] \) and that \( t_{k-4}, t_{k-3}, t_{k-2}, t_{k-1}, t_k, f_k, f_{k-1} \in [a, b] \). Then, the five equations can be obtained as follows.

\[
\begin{align*}
    f_{k+1} &= f_k + \tau \dot{f}_k + \frac{\tau^2}{2} \ddot{f}_k + \frac{\tau^3}{6} \dddot{f}_k + O(\tau^4) \\
    f_{k-1} &= f_k - \tau \dot{f}_k + \frac{\tau^2}{2} \ddot{f}_k - \frac{\tau^3}{6} \dddot{f}_k + O(\tau^4) \\
    f_{k-2} &= f_k - 2\tau \dot{f}_k + 2\tau^2 \ddot{f}_k - \frac{4\tau^3}{3} \dddot{f}_k + O(\tau^4) \\
    f_{k-3} &= f_k - 3\tau \dot{f}_k + 9\tau^2 \ddot{f}_k - \frac{9\tau^3}{2} \dddot{f}_k + O(\tau^4) \\
    f_{k-4} &= f_k - 4\tau \dot{f}_k + 8\tau^2 \ddot{f}_k - \frac{32\tau^3}{3} \dddot{f}_k + O(\tau^4)
\end{align*}
\]

In order to eliminate the \( \dddot{f}_k \) and \( \ddot{f}_k \) terms of (22)-(26), it can be transformed into a problem for solving system of homogeneous linear equations, which is given by

\[
\begin{align*}
    c_1 + c_2 + c_3 + c_4 + c_5 + c_6 &= 0 \\
    \frac{1}{2} c_1 + \frac{1}{2} c_3 + 2c_4 + \frac{5}{2} c_5 + 8c_6 &= 0 \\
    \frac{1}{6} c_1 - \frac{1}{6} c_3 - \frac{4}{3} c_4 - \frac{2}{3} c_5 - \frac{32}{3} c_6 &= 0
\end{align*}
\]

Solving the (27), it can be obtained the following results

\[
\begin{align*}
    c_1 &= 2k_1 + 9k_2 + 24k_3 \\
    c_2 &= 3k_1 + 8k_2 + 15k_3 \\
    c_3 &= -6k_1 - 18k_2 - 40k_3 \\
    c_4 &= k_1, c_5 = k_2, c_6 = k_3
\end{align*}
\]

and the parameter \( \xi \) in the (3) can be calculated as follows.

\[
\xi = c_1 - c_3 - 2c_4 - 3c_5 - 4c_6 = 6k_1 + 24k_2 + 60k_3
\]

Substituting (28) and \( \xi \) into (3), the (3) can be rewritten as

\[
\begin{align*}
    \dot{f}_k = &\left((2k_1 + 9k_2 + 24k_3)f_{k+1} + (3k_1 + 8k_2 + 15k_3)f_k \\
    &- (6k_1 + 18k_2 + 40k_3)f_{k-1} + k_1f_{k-2} + k_2f_{k-3} \\
    &+ k_3f_{k-4}\right)/(6k_1 + 24k_2 + 60k_3) + O(\tau^3)
\end{align*}
\]

Based on the Definition 1, utilizing the (29) to discretize the CTZTM model (7) to obtain the HADTZTM-K (9) and HADTZTM-U (11) models. Then, choosing the appropriate \( k_1, k_2, k_3 \) to make the equations (9) and (11) satisfy the 0-stability. In this paper, it can be chose the case as follows: \( k_1 = -6, k_2 = -4, \) and \( k_3 = 3. \) Then, the specific high precision first-order derivative approximation formula can be given by

\[
\begin{align*}
    \dot{f}_k &= \frac{24f_{k+1} - 5f_k - 12f_{k-1} - 6f_{k-2} - 4f_{k-3}}{48\tau} + \frac{3f_{k-4}}{48\tau} + O(\tau^3)
\end{align*}
\]

by eliminating \( O(\tau^3) \) of (30), then, equation (4) can be obtained as

\[
f_k \approx \frac{24f_{k+1} - 5f_k - 12f_{k-1} - 6f_{k-2} - 4f_{k-3} + 3f_{k-4}}{48\tau}
\]

Therefore, the proof is completed.

ACKNOWLEDGMENT

The authors would like to thank the editors and anonymous reviewers for their valuable suggestions and constructive comments which have really helped them improve very much the presentation and quality of the paper.

REFERENCES

[1] Z. Zhang, L. Zheng, J. Yu, Y. Li, and Z. Yu, “Three recurrent neural networks and three numerical methods for solving a repetitive motion planning scheme of redundant robot manipulators,” IEEE/ASME Trans. Mechatronics, vol. 22, no. 3, pp. 1423–1434, Jun. 2017.

[2] L. Jin, S. Li, X. Luo, Y. Li, and B. Qin, “Neural dynamics for cooperative control of redundant robot manipulators,” IEEE Trans Ind. Informat., vol. 14, no. 9, pp. 3812–3821, Sep. 2018.

[3] L. V. Ferreira, E. Kaszkurewicz, and A. Bhaya, “Solving systems of linear equations via gradient systems with discontinuous right-hand sides: Application to LS-SVM,” IEEE Trans. Neural Netw., vol. 16, no. 2, pp. 501–505, Mar. 2005.

[4] L. Jin, S. Li, H. M. La, and X. Luo, “Manipulability optimization of redundant manipulators using dynamic neural networks,” IEEE Trans. Ind. Electron., vol. 64, no. 6, pp. 4710–4720, Jun. 2017.

[5] Y. Saad and H. A. V. D. Vorst, “Iterative solution of linear systems in the 20th century,” J. Comput. Appl. Math., vol. 123, nos. 1–2, pp. 1–33, 2000.

[6] E. Yusufoğlu, “An improvement to homotopy perturbation method for solving system of linear equations,” Comput. Math. Appl., vol. 58, nos. 11–12, pp. 2231–2235, Dec. 2009.

[7] M. T. Kajani and F. T. Kajani, “A new approach based on homotopy analysis method for solving system of linear equations,” Austral. J. Basic Appl. Sci., vol. 4, no. 12, pp. 6575–6579, 2010.

[8] B. Qiu, Y. Zhang, and Z. Yang, “New discrete-time ZNN models for least-squares solution of dynamic linear equation system with time-varying rank-deficient coefficient,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 11, pp. 5767–5776, Nov. 2018.

[9] F. Caccavale, S. Chiaverini, and B. Siciliano, “Second-order kinematic control of robot manipulators with jacobian damped least-squares inverse: Theory and experiments,” IEEE/ASME Trans. Mechatronics, vol. 2, no. 3, pp. 188–194, Sep. 1997.

[10] L. Jin, J. Yan, X. Du, X. Xiao, and D. Fu, “RNN for solving time-variant generalized Sylvester equation with applications to robotics and acoustic source localization,” IEEE Trans Ind. Informat., to be published, doi: 10.1109/tii.2020.2964817.

[11] 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.

[12] L. Jin, S. Li, B. Hu, M. Liu, and J. Fu, “A noise-suppressing neural algorithm for solving the time-varying system of linear equations: A control-based approach,” IEEE Trans Ind. Informat., vol. 15, no. 1, pp. 236–246, Jan. 2019.

[13] S. Li, J. He, Y. Li, and M. U. Rafique, “Distributed recurrent neural networks for cooperative control of manipulators: A game-theoretic perspective,” IEEE Trans. Neural Netw. Learn. Syst., vol. 28, no. 2, pp. 415–426, Feb. 2017.
y

Z. Sun et al.: Two DTZNN Models of $O(\varepsilon^+)$ Pattern for Online Solving Dynamic System of Linear Equations

[14] L. Jin, Y. Zhang, and B. Qiu, “Neural network-based discrete-time Z-type model of high accuracy in noisy environments for solving dynamic system of linear equations,” Neural Comput. Appl., vol. 29, no. 11, pp. 1217–1232, Oct. 2016.

[15] Y.-J. Liu, J. Li, S. Tong, and C. L. P. Chen, “Neural network control-based adaptive learning design for nonlinear systems with full-state constraints,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 7, pp. 1562–1571, Jul. 2016.

[16] L. Jin, S. Li, J. Yu, and J. He, “Robot manipulator control using neural networks: A survey,” Neurocomputing, vol. 285, pp. 23–34, Apr. 2018.

[17] L. Jin, S. Li, B. Liao, and Z. Zhang, “Zeroing neural networks: A survey,” Neurocomputing, vol. 267, pp. 597–604, Dec. 2017.

[18] L. Tang and D. Li, “Time-varying barrier Lyapunov function-based adaptive fuzzy control for switched systems with unknown dead zone,” IEEE Access, vol. 7, pp. 110928–110935, 2019.

[19] L. Tang and D. Li, “Time-varying barrier Lyapunov function based adaptive neural controller design for nonlinear pure-feedback systems with unknown hysteresis,” Int. J. Control, Autom. Syst., vol. 17, no. 7, pp. 1642–1654, Jul. 2019.

[20] Y. N. Zhang, Y. Wang, and L. Jin, “Complex-valued discrete-time neural dynamics for perturbed time-dependent complex quadratic programming with applications,” IEEE Trans. Neural Netw. Learn. Syst., to be published, doi: 10.1109/TNNLS.2019.2944992.

[21] Z. Xie, L. Jin, X. Du, X. Xiao, H. Li, and S. Li, “On generalized RMP scheme for redundant robot manipulators aided with dynamic neural networks and nonconvex bound constraints,” IEEE Trans Ind. Informat., vol. 15, no. 9, pp. 5172–5181, Sep. 2019.

[22] Z. Xie, L. Jin, X. Luo, S. Li, and X. Xiao, “A data-driven cyclic-motion generation scheme for kinematic control of redundant manipulators,” IEEE Trans. Control Syst. Technol., to be published, doi: 10.1109/TCST.2019.2963017.

[23] J. Zhang, L. Jin, and L. Cheng, “RNN for perturbed manipulability optimization of manipulators based on a distributed scheme: A game-theoretic perspective,” IEEE Trans. Neural Netw. Learn. Syst., to be published, doi: 10.1109/TNNLS.2020.2963998.

[24] Y. Zhang, W. Ma, and B. Cai, “From zhang neural networks to newton iteration for matrix inversion,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 56, no. 7, pp. 1405–1415, Jul. 2009.

[25] Y. N. Zhang, Y. Wang, and L. Jin, “Different ZFs leading to various ZNN models illustrated via online solution of time-varying underdetermined systems of linear equations with robotic application,” in Proc. Int. Conf. Adv. Neural Netw. Berlin, Germany: Springer-Verlag, 2013, pp. 481–488.

[26] Y. Zhang, W. Li, D. Guo, B. Mu, and H. Zheng, “Different zhang functions leading to various ZNN models illustrated solving the time-varying overdetermined system of linear equations,” in Proc. IEEE 3rd Int. Conf. Inf. Sci. Technol. (ICIST), Mar. 2013, pp. 771–776.

[27] L. Jin and Y. Zhang, “Discrete-time Zhang neural network of $O(\varepsilon^2)$ pattern for time-varying matrix pseudo-inversion with application to manipulator motion generation,” Neurocomputing, vol. 142, pp. 165–173, Oct. 2014.

[28] J. H. Mathews and K. D. Fink, Numerical Methods Using MATLAB, Upper Saddle River, NJ, USA: Prentice-Hall, 2005.

[29] B. Qiu and Y. Zhang, “Two new discrete-time neurodynamic algorithms applied to online future matrix inversion with nonsingular or sometimes-singular coefficient,” IEEE Trans. Cybern., vol. 49, no. 6, pp. 2032–2045, Jun. 2019.

[30] Y. X. Li, J. Lin, Y. Wang, L. Xiao, and J. Zhang, “Zeroing neural networks: A survey,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 8, pp. 1519–1531, Aug. 2019.

[31] L. Jin, S. Li, B. Liao, and Z. Zhang, “Zeroing neural networks: A survey,” Neurocomputing, vol. 267, pp. 597–604, Dec. 2017.

[32] Y. Zhang, W. Li, D. Guo, B. Mu, and H. Zheng, “Different zhang functions leading to various ZNN models illustrated solving the time-varying overdetermined system of linear equations,” in Proc. IEEE 3rd Int. Conf. Inf. Sci. Technol. (ICIST), Mar. 2013, pp. 771–776.

[33] L. Jin and Y. Zhang, “Discrete-time Zhang neural network of $O(\varepsilon^2)$ pattern for time-varying matrix pseudo-inversion with application to manipulator motion generation,” Neurocomputing, vol. 142, pp. 165–173, Oct. 2014.

[34] J. Zhang, P. Fiers, K. A. Witte, R. W. Jackson, K. L. Poggensee, C. G. Atkeson, and S. H. Collins, “Human-in-the-loop optimization of exoskeleton assistance during walking,” Science, vol. 356, no. 6344, pp. 1280–1284, Jun. 2017.

[35] P. Shen, X. Zhang, and Y. Fang, “Complete and time-optimal path-constrained trajectory planning with torque and velocity constraints: Theory and applications,” IEEE/ASME Trans. Mechatronics, vol. 23, no. 2, pp. 735–746, Apr. 2018.

[36] X. Zhang, X. Chen, F. Farzadpour, and Y. Fang, “A visual distance approach for multicamera deployment with coverage optimization,” IEEE/ASME Trans. Mechatronics, vol. 23, no. 3, pp. 1007–1018, Jun. 2018.

[37] Y. Qi, L. Jin, Y. Wang, L. Xiao, and J. Zhang, “Complex-valued discrete-time neural dynamics for perturbed time-dependent complex quadratic programming with applications,” IEEE Trans. Neural Netw. Learn. Syst., to be published, doi: 10.1109/TNNLS.2019.2944992.

[38] Z. Xie, L. Jin, X. Du, X. Xiao, H. Li, and S. Li, “On generalized RMP scheme for redundant robot manipulators aided with dynamic neural networks and nonconvex bound constraints,” IEEE Trans Ind. Informat., vol. 15, no. 9, pp. 5172–5181, Sep. 2019.

[39] Z. Xie, L. Jin, X. Luo, S. Li, and X. Xiao, “A data-driven cyclic-motion generation scheme for kinematic control of redundant manipulators,” IEEE Trans. Control Syst. Technol., to be published, doi: 10.1109/TCST.2019.2963017.

[40] J. Zhang, L. Jin, and L. Cheng, “RNN for perturbed manipulability optimization of manipulators based on a distributed scheme: A game-theoretic perspective,” IEEE Trans. Neural Netw. Learn. Syst., to be published, doi: 10.1109/TNNLS.2020.2963998.

[41] Y. Zhang, W. Ma, and B. Cai, “From zhang neural network to newton iteration for matrix inversion,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 56, no. 7, pp. 1405–1415, Jul. 2009.

[42] Y. Zhang, W. Li, D. Guo, B. Mu, and H. Zheng, “Different zhang functions leading to various ZNN models illustrated solving the time-varying overdetermined system of linear equations,” in Proc. IEEE 3rd Int. Conf. Inf. Sci. Technol. (ICIST), Mar. 2013, pp. 771–776.

[43] L. Jin and Y. Zhang, “Discrete-time Zhang neural network of $O(\varepsilon^2)$ pattern for time-varying matrix pseudo-inversion with application to manipulator motion generation,” Neurocomputing, vol. 142, pp. 165–173, Oct. 2014.

[44] J. H. Mathews and K. D. Fink, Numerical Methods Using MATLAB, Upper Saddle River, NJ, USA: Prentice-Hall, 2005.

[45] B. Qiu and Y. Zhang, “Two new discrete-time neurodynamic algorithms applied to online future matrix inversion with nonsingular or sometimes-singular coefficient,” IEEE Trans. Cybern., vol. 49, no. 6, pp. 2032–2045, Jun. 2019.

[46] Y. X. Li, J. Lin, Y. Wang, L. Xiao, and J. Zhang, “Zeroing neural networks: A survey,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 8, pp. 1519–1531, Aug. 2019.

[47] L. Jin, S. Li, B. Liao, and Z. Zhang, “Zeroing neural networks: A survey,” Neurocomputing, vol. 267, pp. 597–604, Dec. 2017.

[48] Y. Zhang, W. Li, D. Guo, B. Mu, and H. Zheng, “Different zhang functions leading to various ZNN models illustrated solving the time-varying overdetermined system of linear equations,” in Proc. IEEE 3rd Int. Conf. Inf. Sci. Technol. (ICIST), Mar. 2013, pp. 771–776.

[49] L. Jin and Y. Zhang, “Discrete-time Zhang neural network of $O(\varepsilon^2)$ pattern for time-varying matrix pseudo-inversion with application to manipulator motion generation,” Neurocomputing, vol. 142, pp. 165–173, Oct. 2014.
KEPING LIU received the B.S. degree in control theory and control engineering from the Changchun University of Technology, Changchun, China, in 1993, and the M.S. and Ph.D. degrees in control theory and control engineering with Jilin University, Changchun, in 1999 and 2002, respectively. He is currently a Full Professor with the Department of Control Engineering, Changchun University of Technology. He is also an Academic Visitor with Osaka University, Osaka, Japan, in 2005. His research interests include complex system modeling, optimization and the control, neural networks, intelligent industrial robots systems, and intelligent transportation systems.

LONG JIN (Member, IEEE) received the B.E. degree in automation and the Ph.D. degree in information and communication engineering from Sun Yat-sen University, Guangzhou, China, in 2011 and 2016, respectively. He is currently a Full Professor with the School of Information Science and Engineering, Lanzhou University, Lanzhou, China. Before joining Lanzhou University, in 2017, he was a Postdoctoral Fellow at the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. His main research interests include neural networks, robotics, and intelligent information processing.

LUQUAN REN received the B.S. and M.S. degrees from the Jilin University of Technology, Changchun, China, in 1967 and 1981, respectively. He is currently a Professor with Jilin University, a member of the Chinese Academy of Sciences (CAS), the Vice-Director of the Academic Board, Jilin University, and the Director of the Academic Board, Key Laboratory of Bionic Engineering, Ministry of Education, China. He has published two books, and about 300 academic articles and reports that were indexed by others more than 900 times. His research interests include the adaptability of living creatures to nature, bionic design and manufacturing aiming at the engineering components, and the basic theory of engineering bionics, such as bionic non-smooth surface, bionic materials, bionic electric-osmosis, bionic flexibility, bionic configuration, and coupling bionics.

***