Explicit Linear Dual-Multistep Methods Applied to ZNN Illustrated via Discrete Time-Dependent Linear and Nonlinear Inequalities System Solving

Jinjin Guo*‡§, Binbin Qiu*‡§, Liangjie Ming†‡§ and Yunong Zhang*‡§

*School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China

†School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China

‡Research Institute of Sun Yat-sen University in Shenzhen, Shenzhen 518057, China

§Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou 510006, China

Email: zhynong@mail.sysu.edu.cn, ynzhang@ieee.org

***Abstract*—In this work, time-dependent linear and nonlinear inequalities system (TDLNIS) is studied and solved. First, using zeroing neural network (ZNN) method twice, a continuous time-dependent ZNN (CTDZNN) model is proposed to solve the continuous TDLNIS. Subsequently, explicit linear dual-multistep methods, i.e., explicit linear dual-4-step, dual-3-step, and dual-2-step methods, are presented and studied. Afterwards, by applying the explicit linear dual-4-step method to the proposed CTDZNN model, a 4-step discrete time-dependent ZNN (4S-DTDZNN) model is proposed to solve the discrete TDLNIS. For comparison, 3-step discrete time-dependent ZNN (3S-DTDZNN) and 2-step discrete time-dependent ZNN (2S-DTDZNN) models are also developed for solving the discrete TDLNIS. In addition, theoretical analyses and results indicate the effectiveness and superiority of the proposed 4S-DTDZNN model. Finally, numerical experimental results further substantiate the effectiveness and superiority of the proposed 4S-DTDZNN model.**

***Keywords*—Time-dependent linear and nonlinear inequalities system, zeroing neural network, explicit linear dual-multistep methods, discrete time-dependent zeroing neural network model.**

I. Introduction

Inequality is the mathematical modeling of unequal relation, which is the basis of further mathematical study and an important tool for mastering modern scientific technology [1], [2]. Some researches on the extensions and applications of inequality have been performed over the past few decades [3], [4], [5]. For example, Reference [3] formulated an impulsive delay differential inequality and obtained an estimated decay rate of the inequality solutions. Reference [4] derived a new stability criteria with delay dependence in regard to linear matrix inequalities for load frequency control systems. Reference [5] presented two new sufficient conditions on global asymptotic synchronization for the drive-response inertial delayed neural networks by using constructed integrating inequality and inequality techniques.

Compared with static (or saying, time-invariant) inequality, time-dependent one is more complicated, because it is required to acquire the solution at each instant of time so as to satisfy the real-time computational requirement [6], [7], [8]. In terms of solving time-dependent inequality problems, zeroing neural network (or termed, Zhang neural network, ZNN) method, is a great alternative [2], [9], [10], [11], [12]. The ZNN is a special class of recurrent neural network [6], [7], [11], and it inherits the merits of conventional neural networks, e.g., parallel computing [6]. Existing literatures indicate that the ZNN method is also effective for solving other time-dependent problems [6], [7], [8], [13], such as time-dependent matrix inversion [8], [13]. References [9], [10], [11], and [12] mainly studied time-dependent linear inequality or time-dependent nonlinear inequality by adopting the ZNN method once. Differing from the study subjects in [9], [10], [11], [12], this work considers time-dependent linear inequality and time-dependent nonlinear inequality as a whole, i.e., time-dependent linear and nonlinear inequalities system (TDLNIS). Then, by adopting the ZNN method twice, a continuous time-dependent ZNN (CTDZNN) model is proposed to solve the continuous TDLNIS.

Considering the fact that analog/continuous variables to be processed by computer must be converted into digital/discrete ones [6], [8], developing discrete models/algorithms is essential for solving the corresponding discrete time-dependent problems (including discrete TDLNIS). Generally speaking, discrete models can be developed by adopting time-discretization (or saying generally, numerical differentiation) formulas, such as Euler forward formula and Zhang *et al.* discretization (also termed, Zhang time-discretization, ZTD) formulas, to discretize continuous models [6], [14], [15], [16], [17], [18]. For instance, Reference [6] proposed a 7-step ZTD formula, and utilized it to develop a 7-step ZTD-type discrete time-dependent ZNN (DTDZNN) model for solving discrete time-dependent different-layer nonlinear and linear equations. Reference [15] proposed and studied 3-step ZTD-type DTDZNN models to solve discrete time-dependent equality-constrained quadratic programming problem. In addition to ZTD formulas, References [17] and [18] presented the 4-step Adams-Bashforth (AB) method, and used it to acquire 4-step AB-type DTDZNN models for discrete time-dependent matrix inversion, matrix pseudoinversion, and nonlinear minimization. On the basis of the previous-mentioned CTDZNN model, this work further presents and studies an explicit linear dual-4-step method. By applying the method to the proposed CTDZNN model, a 4-step DTDZNN (4S-DTDZNN) model is thus proposed to solve the discrete TDLNIS. For comparison, explicit linear dual-3-step and dual-2-step methods are also presented and studied. By applying them to the proposed CTDZNN

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

model, 3-step DTDZNN (3S-DTDZNN) and 2-step DTDZNN (2S-DTDZNN) models are developed.

The remainder of this work is organized into six sections. The discrete TDLNIS is introduced in Section II. A CTDZNN is proposed to solve the continuous TDLNIS in Section III. The explicit linear dual-4-step, dual-3-step, and dual-2-step methods are presented and studied, respectively, and then by applying them to the proposed CTDZNN model, the corresponding DTDZNN models are proposed in Section IV. The theoretical analyses and results of the DTDZNN models are provided for solving the discrete TDLNIS in Section V. Two numerical examples are provided to validate the effectiveness of DTDZNN models and the superiority of the 4S-DTDZNN model in Section VI. The work is summed up with final remarks in Section VII. Note that the main contributions and novelties of this work are listed as follows.

1) A CTDZNN model is proposed to solve the continuous TDLNIS.
2) An explicit linear dual-4-step method is first applied to combine with the proposed CTDZNN model, and thus a 4S-DTDZNN model with high precision is proposed to solve the discrete TDLNIS.
3) Comparative numerical experimental results substantiate the effectiveness (or saying, validity) and superiority of the proposed 4S-DTDZNN model.

II. Problem Formulation

The discrete TDLNIS is formulated as the following expression group, with $\mathbf{x}_{k+1}=\mathbf{x}(t_{k+1})\in\mathbb{R}^n$ to be acquired during computational interval $[t_k,t_{k+1})=[k\iota,(k+1)\iota)\subseteq[0,t_\mathrm{f}]$:

$$\begin{cases} W_{k+1}\mathbf{x}_{k+1}\le\mathbf{v}_{k+1}, & (1)\\ \psi(\mathbf{x}_{k+1},t_{k+1})\le\mathbf{0}, & (2)\end{cases}$$

in which ι denotes the length of sampling period, and t_f denotes the final instant of time. Besides, $W_{k+1}\in\mathbb{R}^{r\times n}$ is a time-dependent full-row-rank matrix with $r\le n$; $\mathbf{v}_{k+1}\in\mathbb{R}^r$ and $\psi(\mathbf{x}_{k+1},t_{k+1})\in\mathbb{R}^l$ are time-dependent vectors with $l\le n$. W_{k+1}, $\mathbf{v}_{k+1}$, and $\psi(\mathbf{x}_{k+1},t_{k+1})$ are assumed to be generated from $W(t)$, $\mathbf{v}(t)$, and $\psi(\mathbf{x}(t),t)$, respectively, by sampling at t_{k+1}. We need to acquire the future unknown solution $\mathbf{x}_{k+1}$ during $[t_k,t_{k+1})$ based upon the already known data information, such as W_k, $\mathbf{v}_k$, and $\mathbf{x}_k$. Thus, the stringent real-time computational requirement is guaranteed [6], [8].

To solve the discrete TDLNIS (1)-(2), the corresponding continuous TDLNIS needs to be first studied, which is formulated as follows (i.e., the so-called continuation technique):

$$\begin{cases} W(t)\mathbf{x}(t)\le\mathbf{v}(t), & (3)\\ \psi(\mathbf{x}(t),t)\le\mathbf{0}, & (4)\end{cases}$$

with $\mathbf{x}(t)$ denoting the unknown time-dependent solution of the continuous TDLNIS (3)-(4).

III. CTDZNN Model

By adopting the ZNN method twice, a CTDZNN model is proposed to solve the continuous TDLNIS (3)-(4) in this section.

First of all, by introducing a time-dependent nonnegative vector $\mathbf{y}^{\cdot 2}(t)=[y_1^2(t),y_2^2(t),\cdots,y_r^2(t)]^\mathrm{T}\in\mathbb{R}^r$, with the superscript $^\mathrm{T}$ denoting the transpose operator, (3) can be converted into an equality as below:

$$W(t)\mathbf{x}(t)-\mathbf{v}(t)+\mathbf{y}^{\cdot 2}(t)=\mathbf{0},$$

where slack variable vector $\mathbf{y}(t)=[y_1(t),y_2(t),\cdots,y_r(t)]^\mathrm{T}\in\mathbb{R}^r$ is unknown and needs to be acquired in the solution process of (3)-(4). Afterwards, a vector-valued error function (or termed, zeroing function) is defined [2], [6], [10], [13]:

$$\mathbf{e}(t)=W(t)\mathbf{x}(t)-\mathbf{v}(t)+\mathbf{y}^{\cdot 2}(t), \tag{5}$$

where $\mathbf{y}^{\cdot 2}(t)$ is equivalent to $\Lambda(t)\mathbf{y}(t)$, with $\Lambda(t)=\mathrm{diag}\{y_1(t),y_2(t),\cdots,y_r(t)\}\in\mathbb{R}^{r\times r}$ [10]. The following equation can be obtained by applying the ZNN method, i.e., the ZNN design formula, $\dot{\mathbf{e}}(t)=-\lambda\mathbf{e}(t)$, to (5):

$$\begin{aligned}&W(t)\dot{\mathbf{x}}(t)+2\Lambda(t)\dot{\mathbf{y}}(t)=-\dot{W}(t)\mathbf{x}(t)+\dot{\mathbf{v}}(t)\\&-\lambda(W(t)\mathbf{x}(t)-\mathbf{v}(t)+\mathbf{y}^{\cdot 2}(t)),\end{aligned} \tag{6}$$

in which the design parameter $\lambda>0$. About inequality (4), another vector-valued error function is defined as

$$\hat{\mathbf{e}}(t)=[\hat{e}_1(t),\hat{e}_2(t),\cdots,\hat{e}_l(t)]^\mathrm{T}\in\mathbb{R}^l,$$

where $\hat{e}_i(t)=(\max\{0,\psi_i(\mathbf{x}(t),t)\})^2/2$ with $i=1,2,\cdots,l$ [9], [12]. Applying the ZNN method once more, one can obtain

$$J(\mathbf{x}(t),t)\dot{\mathbf{x}}(t)=-\frac{1}{2}\lambda\max\{\mathbf{0},\psi(\mathbf{x}(t),t)\}-\dot{\psi}_t(\mathbf{x}(t),t), \tag{7}$$

where

$$J(\mathbf{x}(t),t)=\begin{bmatrix}\frac{\partial\psi_1(\mathbf{x}(t),t)}{\partial x_1(t)} & \frac{\partial\psi_1(\mathbf{x}(t),t)}{\partial x_2(t)} & \cdots & \frac{\partial\psi_1(\mathbf{x}(t),t)}{\partial x_n(t)}\\ \frac{\partial\psi_2(\mathbf{x}(t),t)}{\partial x_1(t)} & \frac{\partial\psi_2(\mathbf{x}(t),t)}{\partial x_2(t)} & \cdots & \frac{\partial\psi_2(\mathbf{x}(t),t)}{\partial x_n(t)}\\ \vdots & \vdots & \ddots & \vdots\\ \frac{\partial\psi_l(\mathbf{x}(t),t)}{\partial x_1(t)} & \frac{\partial\psi_l(\mathbf{x}(t),t)}{\partial x_2(t)} & \cdots & \frac{\partial\psi_l(\mathbf{x}(t),t)}{\partial x_n(t)}\end{bmatrix},$$

and

$$\dot{\psi}_t(\mathbf{x}(t),t)=\left[\frac{\partial\psi_1(\mathbf{x}(t),t)}{\partial t},\frac{\partial\psi_2(\mathbf{x}(t),t)}{\partial t},\cdots,\frac{\partial\psi_l(\mathbf{x}(t),t)}{\partial t}\right]^\mathrm{T},$$

with $J(\mathbf{x}(t),t)\in\mathbb{R}^{l\times n}$ and $\dot{\psi}_t(\mathbf{x}(t),t)\in\mathbb{R}^l$. Let $\mathbf{z}(t)=[\mathbf{x}^\mathrm{T}(t),\mathbf{y}^\mathrm{T}(t)]^\mathrm{T}\in\mathbb{R}^{n+r}$, and combine (6) with (7) together. We further have

$$Q(t)\dot{\mathbf{z}}(t)=\mathbf{q}(t),$$

in which

$$Q(t)=\begin{bmatrix}W(t) & 2\Lambda(t)\\ J(\mathbf{x}(t),t) & O_{l\times r}\end{bmatrix}\in\mathbb{R}^{(r+l)\times(n+r)},$$

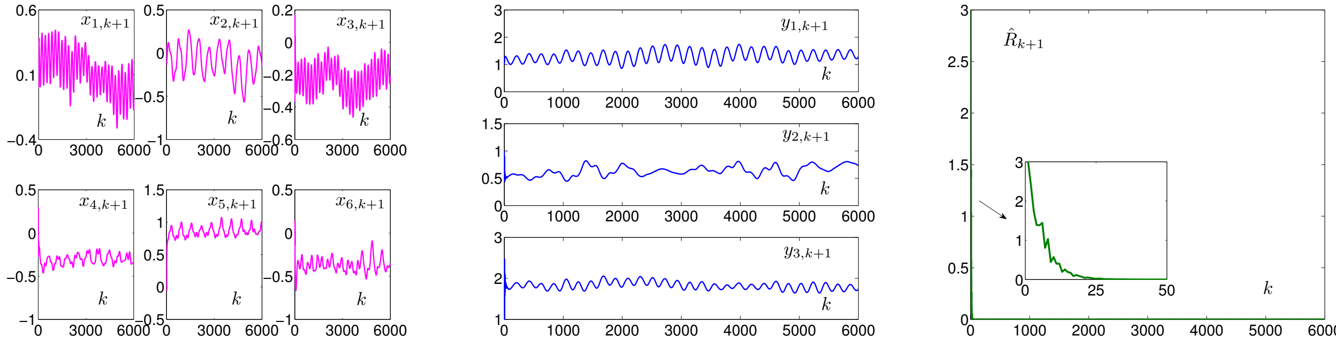
and

$$\mathbf{q}(t)=\begin{bmatrix}\dot{\mathbf{v}}(t)-\dot{W}(t)\mathbf{x}(t)-\lambda(W(t)\mathbf{x}(t)-\mathbf{v}(t)+\mathbf{y}^{\cdot 2}(t))\\ -\lambda\max\{\mathbf{0},\psi(\mathbf{x}(t),t)\}/2-\dot{\psi}_t(\mathbf{x}(t),t)\end{bmatrix}\in\mathbb{R}^{r+l}.$$

$\dot{\mathbf{z}}(t)=[\dot{\mathbf{x}}^\mathrm{T}(t),\dot{\mathbf{y}}^\mathrm{T}(t)]^\mathrm{T}\in\mathbb{R}^{n+r}$ denotes the first-order time derivative of $\mathbf{z}(t)$, and $O_{l\times r}$ denotes an $l\times r$ zero matrix. By assuming that $Q(t)$ is of row full rank, the following CTDZNN model is ultimately developed:

$$\dot{\mathbf{z}}(t)=Q^\dagger(t)\mathbf{q}(t), \tag{8}$$

with the superscript † denoting the pseudo-inverse operator.

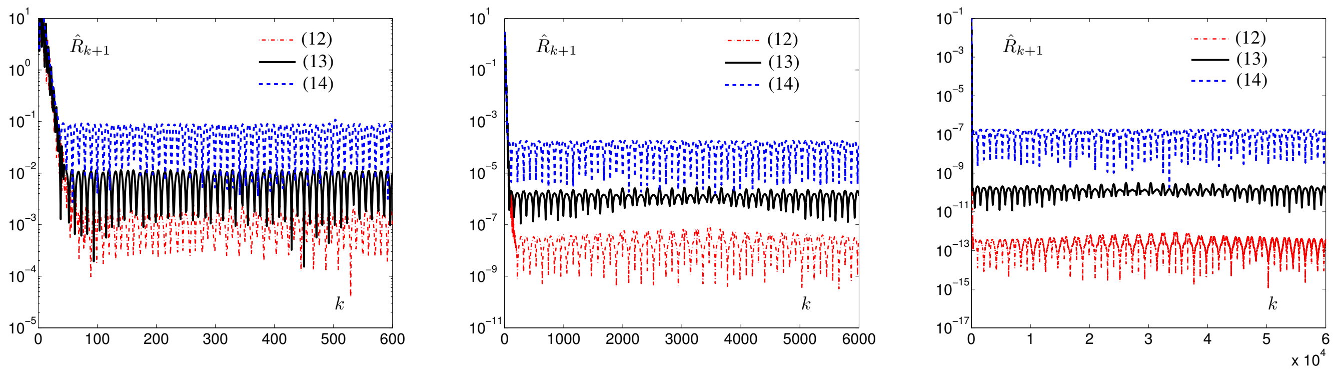


(a) Elemental trajectories of $\mathbf{x}_{k+1}$

(b) Elemental trajectories of $\mathbf{y}_{k+1}$

(c) Trajectory of $\hat{R}_{k+1}$

Fig. 1. Trajectories of $\mathbf{x}_{k+1} = [x_{1,k+1}, x_{2,k+1}, x_{3,k+1}, x_{4,k+1}, x_{5,k+1}, x_{6,k+1}]^{\mathrm{T}}$, $\mathbf{y}_{k+1} = [y_{1,k+1}, y_{2,k+1}, y_{3,k+1}]^{\mathrm{T}}$, and $\hat{R}_{k+1}$ generated by 4S-DTDZNN model (12) with $\iota = 0.01$, respectively, when solving discrete TDLNIS in Example 1.



(a) With $\iota = 0.1$

(b) With $\iota = 0.01$

(c) With $\iota = 0.001$

Fig. 2. Trajectories of $\hat{R}_{k+1}$ synthesized by 4S-DTDZNN model (12), 3S-DTDZNN model (13), and 2S-DTDZNN model (14), respectively, with different values of ι, when solving discrete TDLNIS in Example 1.

IV. Explicit Linear Dual-Multistep Methods and DTDZNN Models

In this section, explicit linear dual-multistep methods, i.e., explicit linear dual-4-step, dual-3-step, and dual-2-step methods are presented and studied. Afterwards, the explicit linear dual-multistep methods are applied to the proposed CTDZNN model (8), with the corresponding DTDZNN models proposed.

First, we have the following lemma with its proof given in Appendix A, developed essentially with the aid of ZTD-involved techniques [2], [6], [8], [15], [18].

Lemma 1: With $\iota \in (0, 1)$, the explicit linear dual-4-step (i.e., in terms of indices of variables and derivatives) method is presented as (with $\mathcal{O}(\iota^5)$ as its truncation error [14], [15]):

$$\begin{aligned}\varsigma_{k+1} &= \varsigma_k - \frac{5}{6}\varsigma_{k-1} + \frac{5}{7}\varsigma_{k-2} + \frac{5}{42}\varsigma_{k-3} \\ &+ \frac{\iota}{126}(285\dot{\varsigma}_k - 256\dot{\varsigma}_{k-1} + 263\dot{\varsigma}_{k-2} - 46\dot{\varsigma}_{k-3}) + \mathcal{O}(\iota^5).\end{aligned} \tag{9}$$

For comparison purposes, the explicit linear dual-3-step and dual-2-step methods are also respectively presented as

$$\begin{aligned}\varsigma_{k+1} &= \frac{10}{7}\varsigma_k - \frac{8}{7}\varsigma_{k-1} + \frac{5}{7}\varsigma_{k-2} \\ &+ \frac{\iota}{28}(47\dot{\varsigma}_k - 32\dot{\varsigma}_{k-1} + 21\dot{\varsigma}_{k-2}) + \mathcal{O}(\iota^4),\end{aligned} \tag{10}$$

and

$$\varsigma_{k+1} = \frac{3}{2}\varsigma_k - \frac{1}{2}\varsigma_{k-1} + \frac{\iota}{4}(5\dot{\varsigma}_k - 3\dot{\varsigma}_{k-1}) + \mathcal{O}(\iota^3), \tag{11}$$

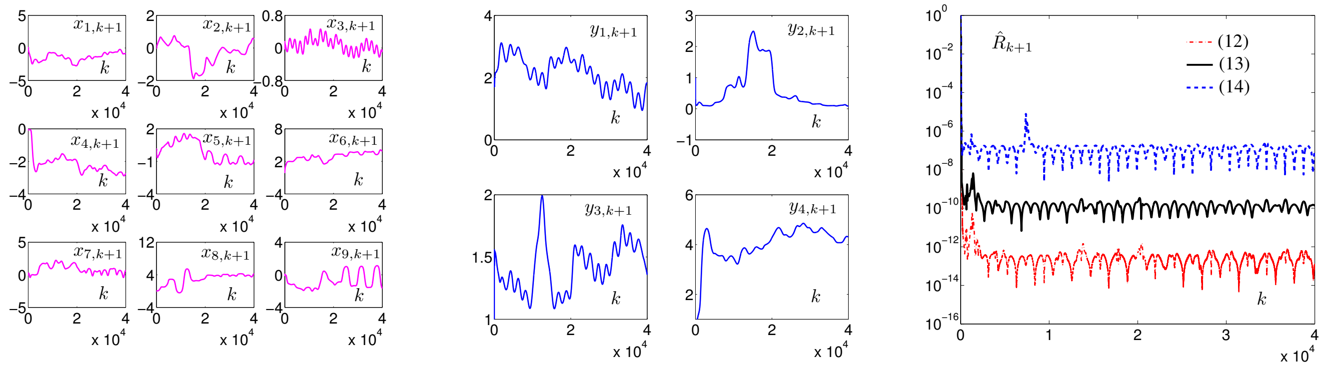
where $\mathcal{O}(\iota^4)$ and $\mathcal{O}(\iota^3)$ as the truncation errors, respectively.

Then, by applying the explicit linear dual-4-step method (9) to the CTDZNN model (8), the following 4S-DTDZNN model is proposed:

$$\begin{aligned}\mathbf{z}_{k+1} &\doteq \mathbf{z}_k - \frac{5}{6}\mathbf{z}_{k-1} + \frac{5}{7}\mathbf{z}_{k-2} + \frac{5}{42}\mathbf{z}_{k-3} \\ &+ \frac{\iota}{126}(285\dot{\mathbf{z}}_k - 256\dot{\mathbf{z}}_{k-1} + 263\dot{\mathbf{z}}_{k-2} - 46\dot{\mathbf{z}}_{k-3}),\end{aligned} \tag{12}$$

where $\doteq$ stands for assigning one result on the right to a variable on the left, and $\dot{\mathbf{z}}_k = Q_k^\dagger \mathbf{q}_k$ with $Q_k^\dagger$ denoting the pseudo-inverse matrix of $Q(t)$ at t_k (i.e., using the conventional static pseudo-inverse operation). Besides, the truncation error of (12) is $\mathbf{O}(\iota^5)$ with every entry being $\mathcal{O}(\iota^5)$. About the proposed 4S-DTDZNN model (12), four initial state vectors, i.e., $\mathbf{z}_0$, $\mathbf{z}_1$, $\mathbf{z}_2$, and $\mathbf{z}_3$, are necessary to start it up. Thereinto, $\mathbf{z}_0$ is relatively arbitrarily set, and the remaining three initial state vectors can be generated by $\mathbf{z}_{k+1} \doteq \mathbf{z}_k + \iota\dot{\mathbf{z}}_k$ with $k = 0$, 1, and 2.

Similarly, by respectively applying the explicit linear dual-3-step method (10) and the explicit linear dual-2-step method



(a) Elemental trajectories of $\mathbf{x}_{k+1}$

(b) Elemental trajectories of $\mathbf{y}_{k+1}$

(c) Trajectory of $\hat{R}_{k+1}$

Fig. 3. Trajectories of $\mathbf{x}_{k+1} = [x_{1,k+1}, x_{2,k+1}, x_{3,k+1}, x_{4,k+1}, x_{5,k+1}, x_{6,k+1}, x_{7,k+1}, x_{8,k+1}, x_{9,k+1}]^{\mathrm{T}}$, $\mathbf{y}_{k+1} = [y_{1,k+1}, y_{2,k+1}, y_{3,k+1}, y_{4,k+1}]^{\mathrm{T}}$, and $\hat{R}_{k+1}$ generated by 4S-DTDZNN model (12) with $\iota = 0.001$, respectively, when solving discrete TDLNIS in Example 2.

(11) to the CTDZNN model (8), the following 3S-DTDZNN and 2S-DTDZNN models are developed:

$$\mathbf{z}_{k+1} \doteq \frac{10}{7}\mathbf{z}_k - \frac{8}{7}\mathbf{z}_{k-1} + \frac{5}{7}\mathbf{z}_{k-2} + \frac{\iota}{28}(47\dot{\mathbf{z}}_k - 32\dot{\mathbf{z}}_{k-1} + 21\dot{\mathbf{z}}_{k-2}), \tag{13}$$

and

$$\mathbf{z}_{k+1} \doteq \frac{3}{2}\mathbf{z}_k - \frac{1}{2}\mathbf{z}_{k-1} + \frac{\iota}{4}(5\dot{\mathbf{z}}_k - 3\dot{\mathbf{z}}_{k-1}), \tag{14}$$

with the corresponding truncation errors being $\mathbf{O}(\iota^4)$ and $\mathbf{O}(\iota^3)$, respectively.

V. Theoretical Analyses and Results

In this section, the theoretical analyses and results of the DTDZNN models are provided for solving the discrete TDLNIS (1)-(2).

Theorem 1: With $\iota \in (0, 1)$, the 4S-DTDZNN model (12) is 0-stable, consistent, and convergent, and it converges with the order of truncation error being $\mathbf{O}(\iota^5)$.

Proof: The proof is given in Appendix B. ■

Corollary 1: With $\iota \in (0, 1)$, the 3S-DTDZNN model (13) and the 2S-DTDZNN model (14) are 0-stable, consistent, and convergent, and they converge with the orders of truncation errors being $\mathbf{O}(\iota^4)$ and $\mathbf{O}(\iota^3)$, respectively.

Defining the total residual error as $\hat{R}_{k+1} = \|W_{k+1}\mathbf{x}_{k+1} - \mathbf{v}_{k+1} + \mathbf{y}_{k+1}^2\|_2 + \|\max\{\mathbf{0}, \psi(\mathbf{x}_{k+1}, t_{k+1})\}\|_2$, with $\|\cdot\|_2$ denoting the 2-norm of a vector, one has the following theorem.

Theorem 2: With $\iota \in (0, 1)$, the total maximal steady-state residual error (TMSSRE) $\lim_{k\to+\infty}\sup \hat{R}_{k+1}$ synthesized by the 4S-DTDZNN model (12) is $\mathcal{O}(\iota^5)$.

Proof: The proof is given in Appendix C. ■

Corollary 2: With $\iota \in (0, 1)$, the TMSSREs synthesized by the 3S-DTDZNN model (13) and the 2S-DTDZNN model (14) are $\mathcal{O}(\iota^4)$ and $\mathcal{O}(\iota^3)$, respectively.

VI. Numerical Experiments and Results

In this section, two numerical examples are provided to validate the effectiveness of DTDZNN models and the superiority of the 4S-DTDZNN model (12), specific as follows.

Example 1: One considers the following discrete TDLNIS with $\mathbf{x}_{k+1}$ to be obtained during computational interval $[t_k, t_{k+1})$, of which the entries (or saying, elements) of coefficient matrix W_k and vector $\mathbf{v}_k$ are respectively

$$w_{i,j}(t_k) = \begin{cases} \dfrac{\cos(0.1(i-j)t_k)}{i-j}, & \text{if } i > j \\ \cos(0.1it_k) + 2i, & \text{if } i = j \\ \dfrac{\sin(0.1(j-i)t_k)}{j-i}, & \text{if } i < j \end{cases} \tag{15}$$

and

$$v_i(t_k) = \begin{cases} \cos(3t_k) + 2, & \text{if } i \text{ is odd} \\ \sin(t_k), & \text{if } i \text{ is even} \end{cases} \tag{16}$$

with $i = 1$, 2, and 3 and $j = 1, 2, \cdots, 6$. Besides, $\psi(\mathbf{x}_k, t_k) \leq \mathbf{0}$ is presented via the following expression group:

$$\begin{cases} x_1(t_k)x_2(t_k) - 1/(t_k+1)^3 + \cos(t_k)x_3(t_k) \\ \quad + x_4(t_k) - x_5^2(t_k) \leq 0, \\ -\sin(t_k)x_2(t_k) - \exp(-2t_k) + x_3^2(t_k) \\ \quad + \exp(-t_k)\sin(2t_k) + 2x_5(t_k)x_6(t_k) \leq 0. \end{cases}$$

The task duration is $T = 60$ s, the sampling period is $\iota = 0.01$, and the design parameter is $\lambda = 20$. The initial state vector is $\mathbf{x}_0 = [0, 0, 0, 0, 0, 0]^{\mathrm{T}}$, and the initial slack vector is $\mathbf{y}_0 = [1, 1, 1]^{\mathrm{T}}$. The corresponding numerical results are shown in Fig. 1. Thereinto, Figs. 1(a) and 1(b) show the elemental trajectories of $\mathbf{x}_{k+1}$ and $\mathbf{y}_{k+1}$, respectively. Besides, Fig. 1(c) shows the trajectory of $\hat{R}_{k+1}$ (i.e., the total residual error). It converges toward zero quickly and the convergence time is approximately 0.25 s. Distinctly, the 4S-DTDZNN model (12) is able to solve the above discrete TDLNIS effectively.

To provide further evidence on the superiority of the 4S-DTDZNN model (12), the 3S-DTDZNN model (13) and the 2S-DTDZNN model (14) are adopted to solve the discrete TDLNIS as well. The trajectories of $\hat{R}_{k+1}$ are shown in Fig. 2, with $\iota\lambda = 0.2$. As indicated in the figure, when ι varies from 0.1 to 0.01 to 0.001, the TMSSREs synthesized by (12) vary from 10^{-3} to 10^{-8} to 10^{-13}. Comparatively, the TMSSREs synthesized by (12) and (14) vary from 10^{-2} to 10^{-6} to 10^{-10} and from 10^{-2} to 10^{-4} to 10^{-7}, respectively. That is, the TMSSREs synthesized by (12), (13), and (14)

TABLE I. TMSSREs SYNTHESIZED BY DTDZNN MODELS IN EXAMPLE 2

Sampling period	Model (12)	Model (13)	Model (14)
$\iota = 0.1$	2.70×10^{-3}	1.10×10^{-1}	1.35×10^{-1}
$\iota = 0.05$	1.49×10^{-4}	1.20×10^{-3}	1.90×10^{-2}
$\iota = 0.01$	6.72×10^{-8}	2.75×10^{-6}	2.14×10^{-4}
$\iota = 0.005$	2.46×10^{-9}	2.13×10^{-7}	2.30×10^{-5}
$\iota = 0.001$	7.36×10^{-13}	2.86×10^{-10}	2.24×10^{-7}

approximately vary in the manners of $\mathcal{O}(\iota^5)$, $\mathcal{O}(\iota^4)$, and $\mathcal{O}(\iota^3)$, respectively. The superiority of the 4S-DTDZNN model (12) is substantiated.

Example 2: One considers another discrete TDLNIS, of which $w_{i,j}(t_k)$ and $v_i(t_k)$ with $i = 1, 2, 3$, and 4 and $j = 1, 2, \cdots, 9$ are defined as those in Example 1 corresponding to the entries of W_k and $\mathbf{v}_k$, respectively. Furthermore, $\psi(\mathbf{x}_k, t_k) \leq \mathbf{0}$ is presented as follows:

$$\begin{cases} \cos(t_k)x_1(t_k) - 1/(t_k+1)^2 \\ + x_4(t_k) - x_5^3(t_k) + x_7^2(t_k) \leq 0, \\ \sin(t_k)x_1(t_k)x_2(t_k) - 3\exp(-t_k) + \exp(-t_k)\sin(4t_k) \\ + x_3^2(t_k) + 2\cos(t_k)x_5(t_k)x_6(t_k) + x_8(t_k)x_9(t_k) \leq 0, \\ 2x_2(t_k) - \exp(-t_k)\sin(4t_k) - \cos(t_k) \\ + x_6(t_k)x_7(t_k) - 2x_9^2(t_k) \leq 0. \end{cases}$$

The relevant parameters and initial values are respectively set as $T = t_{\mathrm{f}} - 0 = 40$ s, $\iota = 0.001$, $\lambda = 200$, $\mathbf{x}_0 = [0,0,0,0,0,0,0,0,0]^{\mathrm{T}}$, and $\mathbf{y}_0 = [2,1,1,1]^{\mathrm{T}}$. The corresponding numerical results are displayed in Fig. 3. Specifically, the elemental trajectories of $\mathbf{x}_{k+1}$ and $\mathbf{y}_{k+1}$ are respectively displayed in Figs. 3(a) and 3(b). Besides, the trajectories of $\hat{R}_{k+1}$ synthesized by (12), (13), and (14) are displayed in Fig. 3(c), of which the TMSSREs are of orders 10^{-13}, 10^{-10}, and 10^{-7}, respectively. It is evident that the three DTDZNN models can effectively solve the discrete TDLNIS, with the 4S-DTDZNN model (12) having the best computational performance.

Moreover, Table I displays the TMSSREs synthesized by the three DTDZNN models, with $\iota\lambda = 0.2$ and different values of ι. The table indicates that, when τ decreases by a factor of 10, the TMSSRE synthesized by (12) approximately improves the precision by a factor of 10^5. Meanwhile, the TMSSREs synthesized by (13) and (14) approximately improve the precision by factors of 10^4 and 10^3, respectively. These findings substantiate that the computational precision of (12), (13), or (14) is approximately $\mathcal{O}(\iota^5)$, $\mathcal{O}(\iota^4)$, or $\mathcal{O}(\iota^3)$, respectively.

VII. CONCLUSION

In this work, we have studied the TDLNIS. First of all, we have proposed the CTDZNN model (8) for solving the continuous TDLNIS (3)-(4) by using the ZNN method twice. Subsequently, we have respectively presented and studied the explicit linear dual-4-step method (9), the explicit linear dual-3-step method (10), and the explicit linear dual-2-step method (11). Afterwards, we have proposed the 4S-DTDZNN model (12) for solving the discrete TDLNIS (1)-(2) by applying the explicit linear dual-4-step method (9) to the proposed CTDZNN model (8). For comparison purposes, we have also developed the 3S-DTDZNN model (13) and the 2S-DTDZNN model (14) for solving the discrete TDLNIS (1)-(2). In addition, theoretical analyses and results have indicated the effectiveness and superiority of the proposed 4S-DTDZNN model (12). Finally, two specific numerical examples have been given to further validate the effectiveness and superiority of the proposed 4S-DTDZNN model (12). The research on general explicit linear dual-multistep methods combining with CTDZNN models for solving various discrete time-dependent problems can be future work.

ACKNOWLEDGMENT

This work is supported in part by the National Natural Science Foundation of China under Grant 61976230, in part by the China Postdoctoral Science Foundation under Grant 2018M643306, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2019A1515012128, in part by the Fundamental Research Funds for the Central Universities under Grant 19lgpy227, and in part by the Shenzhen Science and Technology Plan Project under Grant JCYJ20170818154936083.

APPENDIX A

According to the Taylor expansion theorem [6], [14], the following seven equations are yielded:

$$\begin{aligned} \varsigma_{k+1} = \varsigma((k+1)\iota) = \varsigma_k + \iota\dot{\varsigma}_k + \frac{\iota^2}{2}\ddot{\varsigma}_k \\ + \frac{\iota^3}{6}\dddot{\varsigma}_k + \frac{\iota^4}{24}\varsigma_k^{(4)} + \frac{\iota^5}{120}\varsigma_k^{(5)} + \mathcal{O}(\iota^6), \end{aligned} \tag{17}$$

$$\begin{aligned} \varsigma_{k-1} = \varsigma((k-1)\iota) = \varsigma_k - \iota\dot{\varsigma}_k + \frac{\iota^2}{2}\ddot{\varsigma}_k \\ - \frac{\iota^3}{6}\dddot{\varsigma}_k + \frac{\iota^4}{24}\varsigma_k^{(4)} - \frac{\iota^5}{120}\varsigma_k^{(5)} + \mathcal{O}(\iota^6), \end{aligned} \tag{18}$$

$$\begin{aligned} \varsigma_{k-2} = \varsigma((k-2)\iota) = \varsigma_k - 2\iota\dot{\varsigma}_k + 2\iota^2\ddot{\varsigma}_k \\ - \frac{4\iota^3}{3}\dddot{\varsigma}_k + \frac{2\iota^4}{3}\varsigma_k^{(4)} - \frac{4\iota^5}{15}\varsigma_k^{(5)} + \mathcal{O}(\iota^6), \end{aligned} \tag{19}$$

$$\begin{aligned} \varsigma_{k-3} = \varsigma((k-3)\iota) = \varsigma_k - 3\iota\dot{\varsigma}_k + \frac{9\iota^2}{2}\ddot{\varsigma}_k \\ - \frac{9\iota^3}{2}\dddot{\varsigma}_k + \frac{27\iota^4}{8}\varsigma_k^{(4)} - \frac{81\iota^5}{40}\varsigma_k^{(5)} + \mathcal{O}(\iota^6), \end{aligned} \tag{20}$$

$$\begin{aligned} \dot{\varsigma}_{k-1} = \dot{\varsigma}((k-1)\iota) = \dot{\varsigma}_k - \iota\ddot{\varsigma}_k + \frac{\iota^2}{2}\dddot{\varsigma}_k \\ - \frac{\iota^3}{6}\varsigma_k^{(4)} + \frac{\iota^4}{24}\varsigma_k^{(5)} + \mathcal{O}(\iota^5), \end{aligned} \tag{21}$$

$$\begin{aligned} \dot{\varsigma}_{k-2} = \dot{\varsigma}((k-2)\iota) = \dot{\varsigma}_k - 2\iota\ddot{\varsigma}_k + 2\iota^2\dddot{\varsigma}_k \\ - \frac{4\iota^3}{3}\varsigma_k^{(4)} + \frac{2\iota^4}{3}\varsigma_k^{(5)} + \mathcal{O}(\iota^5), \end{aligned} \tag{22}$$

$$\begin{aligned} \dot{\varsigma}_{k-3} = \dot{\varsigma}((k-3)\iota) = \dot{\varsigma}_k - 3\iota\ddot{\varsigma}_k + \frac{9\iota^2}{2}\dddot{\varsigma}_k \\ - \frac{9\iota^3}{2}\varsigma_k^{(4)} + \frac{27\iota^4}{8}\varsigma_k^{(5)} + \mathcal{O}(\iota^5). \end{aligned} \tag{23}$$

Let us multiply (17), (18), (19), (20), (21), (22), and (23) by 1, $5/6$, $-5/7$, $-5/42$, $256\iota/126$, $-263\iota/126$, and $46\iota/126$, respectively. Subsequently, the following equation can be obtained by adding these results together:

$$
\begin{aligned}
&\varsigma_{k+1} = \varsigma_k - \frac{5}{6}\varsigma_{k-1} + \frac{5}{7}\varsigma_{k-2} + \frac{5}{42}\varsigma_{k-3} \\
&+ \frac{\iota}{126}(285\dot{\varsigma}_k - 256\dot{\varsigma}_{k-1} + 263\dot{\varsigma}_{k-2} - 46\dot{\varsigma}_{k-3}) + \mathcal{O}(\iota^5),
\end{aligned}
$$

which is just the explicit linear dual-4-step (i.e., the indices of variables and derivatives from $k-3$ to $k+1$) method (9). Hence, the proof is completed. ■

APPENDIX B

Based on [14], the first and second characteristic polynomials of the 4S-DTDZNN model (12) are presented as

$$
\begin{cases}
\varrho(\gamma) = \gamma^4 - \gamma^3 + \frac{5}{6}\gamma^2 - \frac{5}{7}\gamma - \frac{5}{42}, \\
\sigma(\nu) = \frac{1}{126}(285\nu^3 - 256\nu^2 + 263\nu - 46).
\end{cases}
$$

There are three roots, namely, $\gamma_1 = -0.1396$, $\gamma_2 = 0.0698 + 0.9208\mathrm{i}$, and $\gamma_3 = 0.0698 - 0.9208\mathrm{i}$ inside the unit circle, and only one root, namely, $\gamma_4 = 1$, on the unit circle. Evidently, the first (or saying, left) characteristic polynomial $\varrho(\gamma)$ satisfies the root condition [6], [14]; hence the 4S-DTDZNN model (12) is 0-stable. Besides, $\varrho(1) = 0$ and $\varrho'(1) = \sigma(1) = 41/21 \neq 0$ are obtained, indicating that the 4S-DTDZNN model (12) is consistent [14]. In accordance with the definition of consistency of order $\mathbf{O}(\iota^\mu)$ in [14], one knows that the 4S-DTDZNN model (12) is consistent of order $\mathbf{O}(\iota^5)$. Considering the fact that 0-stability plus consistency guarantees convergence [14], [18], the 4S-DTDZNN model (12) is convergent, and its convergence order is $\mathbf{O}(\iota^5)$. Hence, the proof is completed. ■

APPENDIX C

Suppose $\mathbf{x}^*_{k+1}$ to be the theoretical solution of the discrete TDLNIS (1)-(2). Based on Theorem 1, $\mathbf{x}_{k+1} = \mathbf{x}^*_{k+1} + \mathbf{O}(\iota^5)$. Thereafter, the following expression is obtained:

$$
\begin{aligned}
&\lim_{k\to+\infty}\sup \|W_{k+1}\mathbf{x}_{k+1} - \mathbf{v}_{k+1} + \mathbf{y}^{.2}_{k+1}\|_2 \\
&= \lim_{k\to+\infty}\sup \|W_{k+1}(\mathbf{x}^*_{k+1} + \mathbf{O}(\iota^5)) - \mathbf{v}_{k+1} + \mathbf{y}^{.2}_{k+1}\|_2 \\
&= \lim_{k\to+\infty}\sup \|W_{k+1}\mathbf{O}(\iota^5)\|_2 = \mathcal{O}(\iota^5),
\end{aligned}
$$

with W_{k+1} being uniformly bounded. When $\psi(\mathbf{x}_{k+1}, t_{k+1}) < \mathbf{0}$, $\|\max\{\mathbf{0}, \psi(\mathbf{x}_{k+1}, t_{k+1})\}\|_2 = 0$ is obtained. The TMSSRE synthesized by the 4S-DTDZNN model (12) is $\mathcal{O}(\iota^5) + 0 = \mathcal{O}(\iota^5)$. When $\psi(\mathbf{x}_{k+1}, t_{k+1}) \geq \mathbf{0}$, $\psi(\mathbf{x}^*_{k+1}, t_{k+1}) = \mathbf{0}$ is obtained, and the following expression is further obtained:

$$
\begin{aligned}
&\lim_{k\to+\infty}\sup \|\max\{\mathbf{0}, \psi(\mathbf{x}_{k+1}, t_{k+1})\}\|_2 \\
&= \lim_{k\to+\infty}\sup \|\max\{\mathbf{0}, \psi(\mathbf{x}^*_{k+1} + \mathbf{O}(\iota^5), t_{k+1})\}\|_2 \\
&= \lim_{k\to+\infty}\sup \|\psi(\mathbf{x}^*_{k+1} + \mathbf{O}(\iota^5), t_{k+1})\|_2 \\
&= \lim_{k\to+\infty}\sup \left\|\frac{\partial\psi(\mathbf{x}^*_{k+1}, t_{k+1})}{\partial\mathbf{x}^*_{k+1}}\mathbf{O}(\iota^5) + \mathbf{O}(\iota^{10})\right\|_2 \\
&= \mathcal{O}(\iota^5),
\end{aligned}
$$

with $\partial\psi(\mathbf{x}^*_{k+1}, t_{k+1})/\partial\mathbf{x}^*_{k+1}$ being uniformly bounded. The TMSSRE synthesized by the 4S-DTDZNN model (12) is $\mathcal{O}(\iota^5) + \mathcal{O}(\iota^5) = \mathcal{O}(\iota^5)$. Hence, the proof is completed. ■

REFERENCES

[1] F. Xu, Z. Li, Z. Nie, H. Shao, and D. Guo, "Zeroing neural network for solving time-varying linear equation and inequality systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 8, pp. 2346–2357, 2019.

[2] Y. Zhang, Z. Qi, B. Qiu, M. Yang, and M. Xiao, "Zeroing neural dynamics and models for various time-varying problems solving with ZLSF models as minimization-type and Euler-type special cases," *IEEE Computational Intelligence Magazine*, vol. 14, no. 3, pp. 52–60, 2019.

[3] X. Li and M. Bohner, "An impulsive delay differential inequality and applications," *Computers and Mathematics with Applications*, vol. 64, no. 6, pp. 1875–1881, 2012.

[4] F. Yang, J. He, and D. Wang, "New stability criteria of delayed load frequency control systems via infinite-series-based inequality," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 1, pp. 231–240, 2018.

[5] Z. Zhang and L. Ren, "New sufficient conditions on global asymptotic synchronization of inertial delayed neural networks by using integrating inequality techniques," *Nonlinear Dynamics*, vol. 95, no. 2, pp. 905–917, 2019.

[6] J. Guo, B. Qiu, J. Chen, and Y. Zhang, "Solving future different-layer nonlinear and linear equation system using new eight-node DZNN model," *IEEE Transactions on Industrial Informatics*, to be published, doi: 10.1109/TII.2019.2933748.

[7] H. Lu, L. Jin, X. Luo, B. Liao, D. Guo, and L. Xiao, "RNN for solving perturbed time-varying underdetermined linear system with double bound limits on residual errors and state variables," *IEEE Transactions on Industrial Informatics*, to be published, doi: 10.1109/TII.2019.2909142.

[8] B. Qiu and Y. Zhang, "Two new discrete-time neurodynamic algorithms applied to online future matrix inversion with nonsingular or sometimes-singular coefficient," *IEEE Transactions on Cybernetics*, vol. 49, no. 6, pp. 2032–2045, 2019.

[9] L. Xiao and Y. Zhang, "Zhang neural network versus gradient neural network for solving time-varying linear inequalities," *IEEE Transactions on Neural Networks*, vol. 22, no. 10, pp. 1676–1684, 2011.

[10] D. Guo and Y. Zhang, "ZNN for solving online time-varying linear matrix-vector inequality via equality conversion," *Applied Mathematics and Computation*, vol. 259, pp. 327–338, 2015.

[11] D. Guo and Y. Zhang, "Zhang neural network for online solution of time-varying linear matrix inequality aided with an equality conversion," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 370–382, 2014.

[12] L. Xiao and Y. Zhang, "Two new types of Zhang neural networks solving systems of time-varying nonlinear inequalities," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 59, no. 10, pp. 2363–2373, 2012.

[13] D. Guo, B. Qiu, Z. Ke, Z. Yang, and Y. Zhang, "Case study of Zhang matrix inverse for different ZFs leading to different nets," in: *Proceedings of the International Joint Conference on Neural Networks*, pp. 2764–2769, 2014.

[14] D. F. Griffiths and D. J. Higham, *Numerical Methods for Ordinary Differential Equations: Initial Value Problems*. London, UK: Springer, 2010.

[15] B. Liao, Y. Zhang, and L. Jin, "Taylor $O(h^3)$ discretization of ZNN models for dynamic equality-constrained quadratic programming with application to manipulators," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 2, pp. 225–237, 2016.

[16] W. Li, L. Xiao, and B. Liao, "A finite-time convergent and noise-rejection recurrent neural network and its discretization for dynamic nonlinear equations solving," *IEEE Transactions on Cybernetics*, to be published, doi: 10.1109/TCYB.2019. 2906263.

[17] M. D. Petkovic, P. S. Stanimirovic, and V. N. Katsikis, "Modified discrete iterations for computing the inverse and pseudoinverse of the time-varying matrix," *Neurocomputing*, vol. 289, pp. 155–165, 2018.

[18] B. Qiu, Y. Zhang, J. Guo, Z. Yang, and X. Li, "New five-step DTZD algorithm for future nonlinear minimization with quartic steady-state error pattern," *Numerical Algorithms*, vol. 81, no. 3, pp. 1043–1065, 2019.