Filomat 34:15 (2020), 5059–5071
https://doi.org/10.2298/FIL2015059G

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

Tracking Control of Ship Course System Using New Six-Step ZeaD (Zhang et al Discretization) Formula with High Precision

Jinjin Guo[a,b], Yunong Zhang[a,b], Binbin Qiu[a,b]

[a] *School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China*
[b] *Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou 510006, China*

Abstract. In this paper, firstly, a new six-step Zhang et al discretization (SSZeaD) formula is proposed, which is with the truncation error proportional to the fourth power of sampling period. Then, the SSZeaD formula is used to discretize a ship course system (SCS) for tracking control, and thus the SSZeaD-type SCS model is developed. For comparison purposes, the classical Euler forward formula (EFF) with the truncation error proportional to the first power of sampling period is also used to discretize the SCS, and thus the EFF-type SCS model is obtained. Besides, there is an important parameter called stepsize, which is closely related to the stability and the precision of the above two discrete-time models. In view of the importance of the stepsize, the effective stepsize domains of these two discrete-time models are confirmed by theoretical analyses. Finally, numerical experimental results well verify the higher tracking precision of the SSZeaD-type SCS model as compared with the EFF-type SCS model.

1. Introduction

Tracking control is often deemed as a fundamental issue, which has been analyzed and applied in various kinds of scientific and engineering fields [1–12], such as robotic kinematics [1, 13–16]. For achieving the tracking control purpose of a system, a controller needs to be designed to force the error between the actual output and the desired output as close to zero as possible [17, 18]. In [19], the feedback linearization method is used to enhance the tracking performance of a linear hydraulic-actuator by adjusting the operator input as well as reducing and nearly eliminating the load dependence of the tracking response. In [20], the backstepping method is used to construct the adaptive neural tracking controller for a class of nonlinear non-strict-feedback systems. In [21], by adopting zeroing neural dynamics (or termed, zeroing neurodynamics, ZN) method [22–27], a ZN controller is designed to solve the tracking control problem of general-form single-input single-output nonlinear system illustrated by a ship course system (SCS). In fact, the above-mentioned ZN method is proposed by Zhang et al [1, 28–30], and is designed mainly for solving different time-dependent problems such as tracking control problem. More specifically, the ZN method is a novel class of recurrent neurodynamic method [1, 23], which well integrates neural networks and dynamic

2010 *Mathematics Subject Classification*. Primary 34H05; Secondary 65C20, 93C55
Keywords. ZeaD (Zhang et al discretization), Six-step ZeaD (SSZeaD), Euler forward formula (EFF), Stepsize domains
Received: 30 October 2018; Revised: 19 November 2018; Accepted: 09 December 2018
Communicated by Bolin Liao
Corresponding author: Binbin Qiu
Research is supported by the National Natural Science Foundation of China (with numbers 61976230 and 62006254).
Email addresses: guojj2017@126.com (Jinjin Guo), zhynong@mail.sysu.edu.cn (Yunong Zhang), qiubb6@mail.sysu.edu.cn (Binbin Qiu)

systems into a whole, and fully exploits their both advantages, e.g., parallel computing and dynamic updating [1, 31–35]. It is worth noting that we have on the foundation of [21] done further research in tracking control problem of the SCS in this paper.

All the above literatures are focused on the research of continuous-time models. However, it may be a difficult task for directly adopting the continuous-time models to handle the corresponding discrete-time problems [36, 37]. Since digital computers and technologies are widely used in numerous scientific disciplines and industrial applications, e.g., modern industrial control system [38], many signals are made up of discrete-time variables rather than continuous-time ones [1]. Besides, the discrete-time models are more suitable to represent practical engineering problems as compared with the continuous-time ones [1, 36]. For this purpose, it is necessary to develop and investigate the corresponding discrete-time models instead of the above continuous-time models for discrete-time problems solving.

In general, in terms of numerically solving continuous-time models [39, 40], various numerical differentiation methods have been developed to approximate the first-order derivative [41, 42]. For instance, the classical Euler forward formula (EFF) is usually considered as the first and also the simplest one-step-ahead finite difference formula that has been proposed in 1755 and widely applied for decades [1, 43, 44]. In recent years, a class of one-step-ahead time-discretization method using Taylor expansion and derivation has been proposed and applied by Zhang et al since 2014 [44–48], which is referred to as ZeaD (Zhang et al discretization) method in this paper. Specially, the EFF can be regarded as a one-step ZeaD formula.

In this paper, in order to achieve higher precision in approximating the first-order derivative and discretizing more effectively the continuous-time models [46], a six-step ZeaD (SSZeaD) formula with $O(g^4)$ precision is proposed (where g is the sampling period). Then, it is used to discretize the SCS, and thus the SSZeaD-type SCS model is developed. For comparison purposes, the EFF with $O(g)$ precision is also used to discretize the SCS, and thus the EFF-type SCS model is obtained. Besides, we mainly concentrate on the stability and the tracking precision of the SSZeaD-type SCS model and the EFF-type SCS model for tracking control. There is an important parameter termed stepsize in the above two discrete-time models, which is closely related to their stability and precision [46]. If the value of the stepsize is outside its effective domain, no matter how small the sampling period is, the tracking errors of discrete-time models are impossible to achieve convergence, which means the failure of problems solving. Therefore, in a proper domain of the stepsize (that is, the effective stepsize domain), it makes sense for the discrete-time problems solving.

The rest of this paper is divided into five sections. In Section 2, a SSZeaD formula with $O(g^4)$ precision is proposed with the corresponding proof. In Section 3, two different discrete-time SCS models (i.e., the SSZeaD-type SCS model and the EFF-type SCS model) are proposed for comparative analyses. In Section 4, the detailed proofs are provided about the effective stepsize domains of such two different discrete-time SCS models. Section 5 presents the numerical experimental results, which accord with the theoretical results. Section 6 summarizes the whole paper. Before ending this section, it is worth pointing out that the main contributions of this paper are the following.

- In view of the importance of discretizing continuous-time models, the SSZeaD formula with $O(g^4)$ precision is proposed to discretize the SCS. For comparison purposes, the EFF with $O(g)$ precision is also used to discretize the SCS.
- As the stepsize is closely related to the stability and the precision of discrete-time models, the effective stepsize domains of the SSZeaD-type SCS model and the EFF-type SCS model are confirmed, respectively.
- The higher tracking precision of the SSZeaD-type SCS model is illustrated for tracking different desired outputs of the SCS, as compared with the EFF-type SCS model.

For readers' convenience, the acronyms used in this paper are summarized as below.

Acronyms

SCS: Ship course system.
ZN: Zeroing neurodynamics.

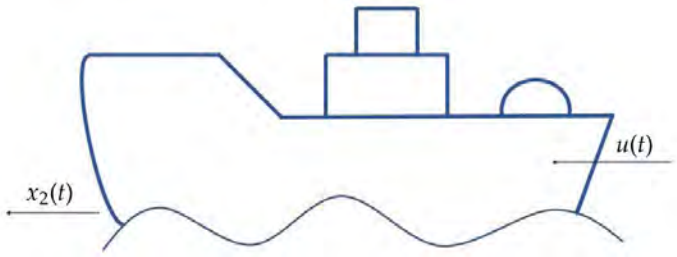


Figure 1: Diagrammatic drawing of simplified SCS.

ZeaD: Zhang et al discretization.
SSZeaD: Six-step Zhang et al discretization.
EFF: Euler forward formula.

2. SSZeaD Formula

In this section, we propose the high-precision SSZeaD formula, which is presented with the corresponding proof below.

Theorem 2.1. *Assume that the first five derivatives of arbitrary function r are continuous on $[t_0, t_f]$, and that t_{k-5}, t_{k-4}, t_{k-3}, t_{k-2}, t_{k-1}, t_k, $t_{k+1} \in [t_0, t_f]$. Thereinto, t_{k+i} is denoted as $(k+i)g$, with $k \in \mathbb{N}$ representing the updating index and $g \in \mathbb{R}^+$ denoting the sampling period. Then, the SSZeaD formula is proposed as*

$$\dot{r}(t_k) = \frac{20r(t_{k+1})}{51g} + \frac{3r(t_k)}{17g} - \frac{71r(t_{k-1})}{204g} - \frac{13r(t_{k-2})}{34g} + \frac{3r(t_{k-3})}{34g} + \frac{13r(t_{k-4})}{102g} - \frac{11r(t_{k-5})}{204g} + O(g^4). \tag{1}$$

Proof According to Taylor expansion, we have the expressions of $r(t_{k+1})$, $r(t_{k-1})$, $r(t_{k-2})$, $r(t_{k-3})$, $r(t_{k-4})$ and $r(t_{k-5})$ as below:

$$r(t_{k+1}) = r(t_k) + g\dot{r}(t_k) + \frac{g^2}{2}\ddot{r}(t_k) + \frac{g^3}{6}r^{(3)}(t_k) + \frac{g^4}{24}r^{(4)}(t_k) + O(g^5), \tag{2}$$

$$r(t_{k-1}) = r(t_k) - g\dot{r}(t_k) + \frac{g^2}{2}\ddot{r}(t_k) - \frac{g^3}{6}r^{(3)}(t_k) + \frac{g^4}{24}r^{(4)}(t_k) + O(g^5), \tag{3}$$

$$r(t_{k-2}) = r(t_k) - 2g\dot{r}(t_k) + 2g^2\ddot{r}(t_k) - \frac{4}{3}g^3r^{(3)}(t_k) + \frac{2}{3}g^4r^{(4)}(t_k) + O(g^5), \tag{4}$$

$$r(t_{k-3}) = r(t_k) - 3g\dot{r}(t_k) + \frac{9}{2}g^2\ddot{r}(t_k) - \frac{9}{2}g^3r^{(3)}(t_k) + \frac{27}{8}g^4r^{(4)}(t_k) + O(g^5), \tag{5}$$

$$r(t_{k-4}) = r(t_k) - 4g\dot{r}(t_k) + 8g^2\ddot{r}(t_k) - \frac{32}{3}g^3r^{(3)}(t_k) + \frac{32}{3}g^4r^{(4)}(t_k) + O(g^5), \tag{6}$$

and

$$r(t_{k-5}) = r(t_k) - 5g\dot{r}(t_k) + \frac{25}{2}g^2\ddot{r}(t_k) - \frac{125}{6}g^3r^{(3)}(t_k) + \frac{625}{24}g^4r^{(4)}(t_k) + O(g^5). \tag{7}$$

Based on the algebraic operations "$[(2)\times 80 - (3)\times 71 - (4)\times 78 + (5)\times 18 + (6)\times 26 - (7)\times 11]/204$", the following equation is obtained:

$$\dot{r}(t_k) = \frac{20r(t_{k+1})}{51g} + \frac{3r(t_k)}{17g} - \frac{71r(t_{k-1})}{204g} - \frac{13r(t_{k-2})}{34g} + \frac{3r(t_{k-3})}{34g} + \frac{13r(t_{k-4})}{102g} - \frac{11r(t_{k-5})}{204g} + O(g^4).$$

The proof is thus completed. □

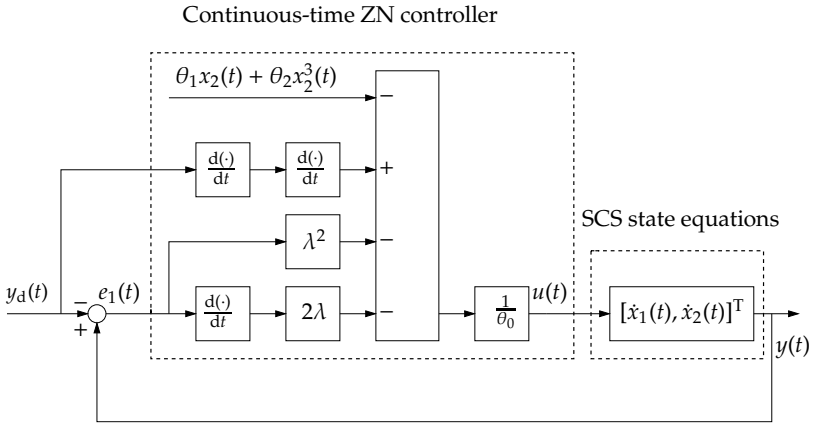


Figure 2: Realization of continuous-time ZN controller (9) for tracking control of SCS (8), where SCS state equations represent problem solving process.

3. Two Different Discrete-Time SCS Models

In this section, we mainly propose two different discrete-time SCS models (i.e., the SSZeaD-type SCS model and the EFF-type SCS model) for comparative analyses. Before developing discrete-time SCS models, we provide a brief introduction to the SCS. The schematic diagram of the simplified SCS is shown in Figure 1. The system equations of SCS [21] can be expressed as

$$\begin{cases} \dot{x}_1(t) = x_2(t), \\ \dot{x}_2(t) = \theta_1 x_2(t) + \theta_2 x_2^3(t) + \theta_0 u(t), \\ y(t) = x_1(t), \end{cases} \tag{8}$$

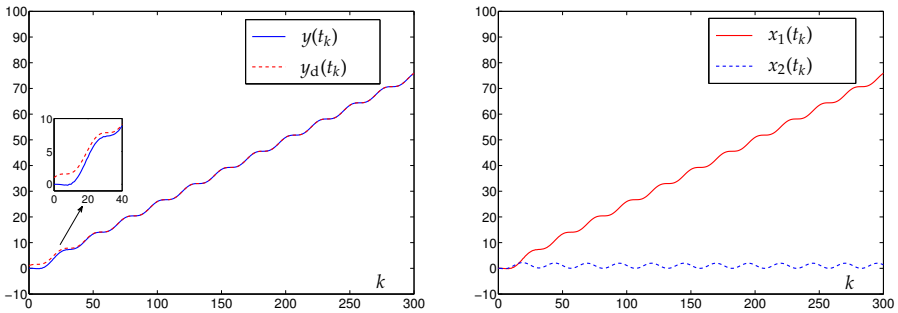
where $x_1(t)$ represents the displacement and $x_2(t)$ represents the velocity. Besides, $u(t)$ represents the control input; $y(t)$ represents the system output, i.e., the displacement; $\theta_0, \theta_1, \theta_2$ represent constant parameters. The continuous-time ZN controller designed in [21] is directly presented by adopting the ZN method as follows:

$$u(t) = \frac{\ddot{y}_{\mathrm{d}}(t) + 2\lambda\dot{y}_{\mathrm{d}}(t) + \lambda^2 y_{\mathrm{d}}(t) - 2\lambda x_2(t) - \lambda^2 x_1(t) - \theta_1 x_2(t) - \theta_2 x_2^3(t)}{\theta_0}, \tag{9}$$

where $y_{\mathrm{d}}(t)$ is the desired output, $\dot{y}_{\mathrm{d}}(t)$ is the first-order time derivative of $y_{\mathrm{d}}(t)$, and $\ddot{y}_{\mathrm{d}}(t)$ is the second-order time derivative of $y_{\mathrm{d}}(t)$. Besides, the design parameter $\lambda \in \mathbb{R}^+$ is for adjusting the convergence rate of the ZN method [21–25, 49].

For a better understanding of continuous-time ZN controller (9) acting on the tracking control of SCS (8), its realization is represented as a control system shown in Figure 2. From this figure, we can find that the continuous-time ZN controller (9) takes effect through the state equations of SCS (8), which makes the tracking error (or termed, error function) $e_1(t)$ as close to zero as possible.

By adopting the SSZeaD formula (1) to discretize (8), the following SSZeaD-type SCS model can be



(a) Actual output $y(t_k)$ and desired output $y_d(t_k)$

(b) State variables $x_1(t_k)$ and $x_2(t_k)$

(c) Trajectory of control input $u(t_k)$

(d) Absolute tracking error $|e(t_k)|$

Figure 3: Performance of ZN controller (11) for SSZeaD-type SCS model (10) tracking desired output $y_d(t_k) = \cos(t_k) + t_k$, with $g = 0.25$ s.

developed with a truncation error of order $O(g^5)$:

$$\begin{cases} x_1(t_{k+1}) = -9x_1(t_k)/20 + 71x_1(t_{k-1})/80 + 39x_1(t_{k-2})/40 - 9x_1(t_{k-3})/40 \\ \qquad - 13x_1(t_{k-4})/40 + 11x_1(t_{k-5})/80 + 51g\dot{x}_1(t_k)/20 + O(g^5) \\ \qquad = -9x_1(t_k)/20 + 71x_1(t_{k-1})/80 + 39x_1(t_{k-2})/40 - 9x_1(t_{k-3})/40 \\ \qquad - 13x_1(t_{k-4})/40 + 11x_1(t_{k-5})/80 + 51gx_2(t_k)/20 + O(g^5), \\ x_2(t_{k+1}) = -9x_2(t_k)/20 + 71x_2(t_{k-1})/80 + 39x_2(t_{k-2})/40 - 9x_2(t_{k-3})/40 \\ \qquad - 13x_2(t_{k-4})/40 + 11x_2(t_{k-5})/80 + 51g\dot{x}_2(t_k)/20 + O(g^5) \\ \qquad = 51g\Big(\theta_1 x_2(t_k) + \theta_2 x_2^3(t_k) + \theta_0 u(t_k)\Big)/20 - 9x_2(t_k)/20 + 71x_2(t_{k-1})/80 \\ \qquad + 39x_2(t_{k-2})/40 - 9x_2(t_{k-3})/40 - 13x_2(t_{k-4})/40 + 11x_2(t_{k-5})/80 + O(g^5), \\ y(t_{k+1}) = x_1(t_{k+1}). \end{cases} \tag{10}$$

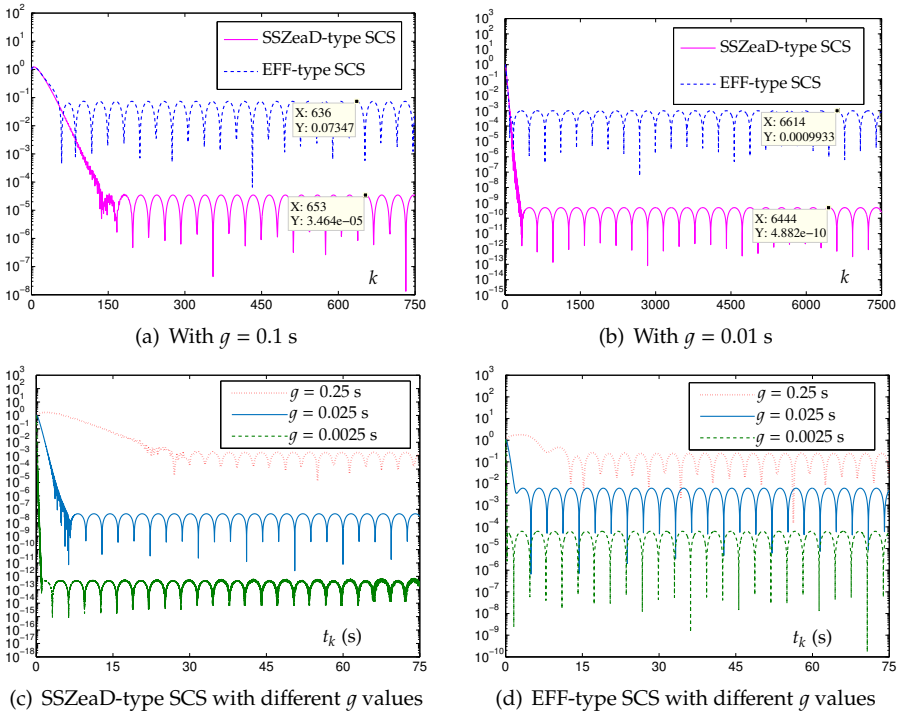
Accordingly, with t_k being an arbitrary time instant, the continuous-time ZN controller can be discretized as

$$u(t_k) = \Big(\ddot{y}_d(t_k) + 2h\dot{y}_d(t_k)/g + h^2 y_d(t_k)/g^2 - 2hx_2(t_k)/g - h^2 x_1(t_k)/g^2 - \theta_1 x_2(t_k) - \theta_2 x_2^3(t_k)\Big)/\theta_0, \tag{11}$$

where $h = \lambda g$ with h denoting the stepsize. In reality, the discrete-time ZN controller (11) can achieve the tracking control of the SSZeaD-type SCS model (10), which is verified by numerical experiments in the ensuing Section 5.

The conventional EFF [1, 43, 48, 50] is shown as

$$\dot{r}(t_k) = \frac{r(t_{k+1}) - r(t_k)}{g} + O(g). \tag{12}$$



(a) With $g = 0.1$ s (b) With $g = 0.01$ s

(c) SSZeaD-type SCS with different g values (d) EFF-type SCS with different g values

Figure 4: Absolute tracking errors $|e(t_k)|$ for desired output $y_d(t_k) = \cos(t_k) + t_k$ when using SSZeaD-type SCS model (10) and EFF-type SCS model (13), respectively.

It is also used to discretize (8) and the EFF-type SCS model is thus obtained as below with a truncation error of order $O(g^2)$:

$$\begin{cases} x_1(t_{k+1}) &= x_1(t_k) + gx_2(t_k) + O(g^2), \\ x_2(t_{k+1}) &= (1 + g\theta_1)x_2(t_k) + g\theta_2 x_2^3(t_k) + g\theta_0 u(t_k) + O(g^2), \\ y(t_{k+1}) &= x_1(t_{k+1}). \end{cases} \tag{13}$$

To lay a basis for further discussion, it is necessary to present the following important results on the convergence of discrete-time models.

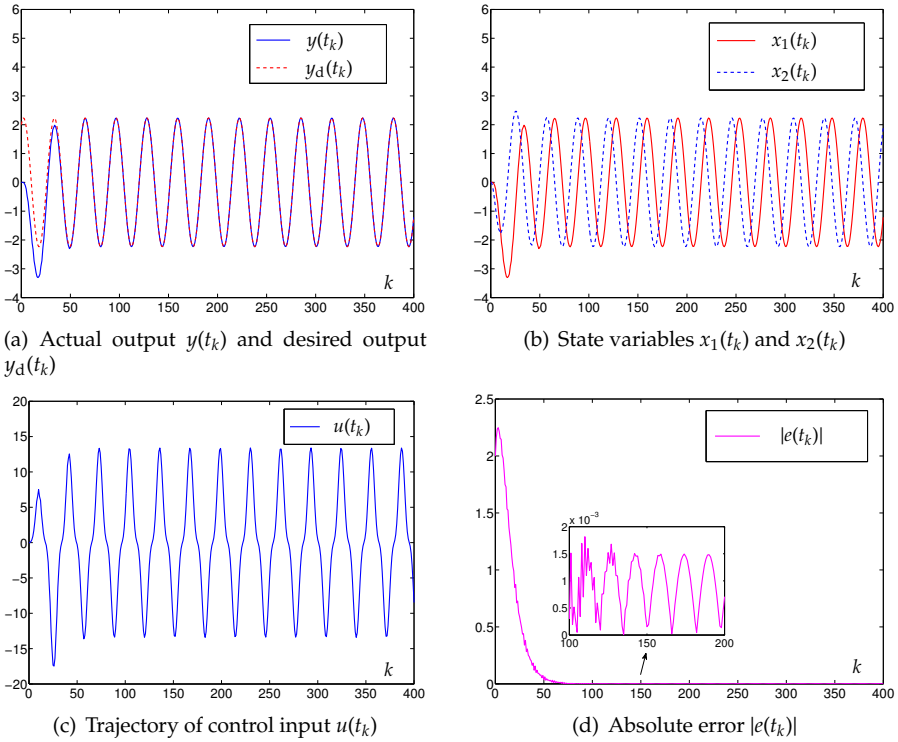
Result 1 [51, 52] An N-step method $\sum_{j=0}^{N} \alpha_j \psi_{m+j} = g \sum_{j=0}^{N} \beta_j f_{m+j}$ is zero-stable, whose sufficient and necessary condition is that root condition is satisfied. That is, the roots of characteristic polynomial $\rho_N(\xi) = \sum_{j=0}^{N} \alpha_j \xi^j$ should satisfy $|\xi| \leq 1$ with $|\xi| = 1$ being simple. In addition, the zero stability is also called Dahlquist stability or root stability.

Result 2 [52, 53] An N-step method is said to be consistent (i.e., has consistency) of order p if the truncation error for the exact solution is of order $O(g^{p+1})$ where $p > 0$.

Result 3 [51] An N-step method is convergent, i.e., $\psi_{[t/g]} \to \psi^*(t)$, for all $t \in [0, t_f]$, as $g \to 0$, if and only if the method is zero-stable and consistent. Thereinto, $\psi_{[t/g]}$ is the actual solution with $[t/g]$ being the largest integer not greater than t/g, and $\psi^*(t)$ is the theoretical solution. That is, zero stability plus consistency means convergence, which is also known as Dahlquist equivalence theorem.

Result 4 [51, 52] A zero-stable consistent method converges with the order of its truncation error.

According to above four results, we can obtain the following two theorems.



(a) Actual output $y(t_k)$ and desired output $y_d(t_k)$

(b) State variables $x_1(t_k)$ and $x_2(t_k)$

(c) Trajectory of control input $u(t_k)$

(d) Absolute error $|e(t_k)|$

Figure 5: Performance of ZN controller (11) for SSZeaD-type SCS model (10) tracking desired output $y_d(t_k) = \sin(t_k) + 2\cos(t_k)$, with $g = 0.2$ s.

Theorem 3.1. *The SSZeaD-type SCS model (10) is zero-stable, and the EFF-type SCS model (13) is also zero-stable.*

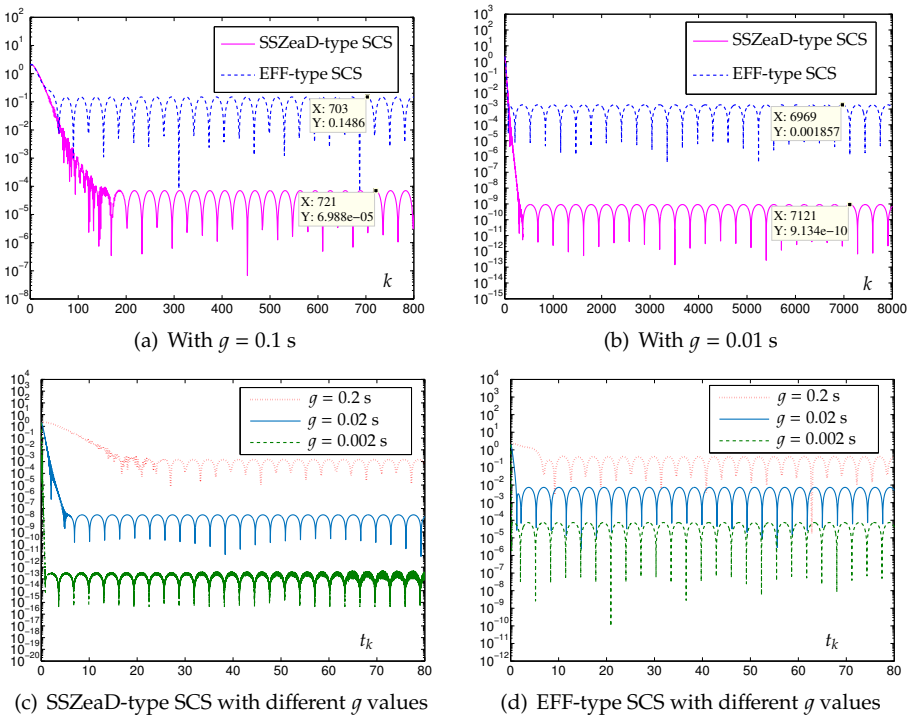
Proof The characteristic polynomial of the SSZeaD-type SCS model (10) can be expressed as

$$\rho_6(\xi) = \xi^6 + \frac{9}{20}\xi^5 - \frac{71}{80}\xi^4 - \frac{39}{40}\xi^3 + \frac{9}{40}\xi^2 + \frac{13}{40}\xi - \frac{11}{80}. \tag{14}$$

The roots of characteristic polynomial (14) are $\xi_1 = 1$, $\xi_2 = -0.7039 + 0.6585\mathrm{i}$, $\xi_3 = -0.7039 - 0.6585\mathrm{i}$, $\xi_4 = -0.8027$, $\xi_5 = 0.3802 + 0.1995\mathrm{i}$, and $\xi_6 = 0.3802 - 0.1995\mathrm{i}$, where i denotes the imaginary unit. Evidently, these roots are on or in the unit circle, and thus the SSZeaD-type SCS model (10) is zero-stable. Similarly, the root of characteristic polynomial of the EFF-type SCS model (13) is $\xi_1 = 1$, which is on the unit circle. Therefore, the EFF-type SCS model (13) is also zero-stable. The proof is thus completed. □

Theorem 3.2. *The SSZeaD-type SCS model (10) is consistent and convergent, which converges with the truncation error of order $O(g^5)$, and the EFF-type SCS model (13) is also consistent and convergent, which converges with the truncation error of order $O(g^2)$.*

Proof From the SSZeaD-type SCS model (10), we know that the truncation error is of order $O(g^5)$; thus, on the basis of Result 1 and Result 2, the SSZeaD-type SCS model (10) is zero-stable and consistent. Moreover, taking Result 3 and Result 4 into consideration, we obtain that the SSZeaD-type SCS model (10) converges with the truncation error of order $O(g^5)$. In the same manner, on the grounds of Result 1 and Result 2, the EFF-type SCS model (13) is zero-stable and consistent. Therefore, in light of Result 3 and Result 4, the EFF-type SCS model (13) converges with the truncation error of order $O(g^2)$. The proof is thus completed.□



(a) With $g = 0.1$ s (b) With $g = 0.01$ s

(c) SSZeaD-type SCS with different g values (d) EFF-type SCS with different g values

Figure 6: Tracking errors $|e(t_k)|$ for desired output $y_d(t_k) = \sin(t_k) + 2\cos(t_k)$ using SSZeaD-type SCS model (10) and EFF-type SCS model (13), respectively.

4. Effective Stepsize Domains

The stepsize is an important parameter since it determines the stability and the precision of discrete-time models. In this section, we present detailed proofs about effective stepsize domains of the SSZeaD-type SCS model (10) and the EFF-type SCS model (13).

Proposition 4.1. *The effective stepsize domain of the SSZeaD-type SCS model (10) is* $(0, 8/51)$.

Proof The first error function $e_1(t) = y(t) - y_d(t) = x_1(t) - y_d(t)$ is constructed, and the ZN design formula [49] is presented as below:

$$\dot{e}_1(t) = -\lambda e_1(t). \tag{15}$$

Then, we use the SSZeaD formula (1) to discretize the ZN design formula (15) directly, and the following equation can be obtained:

$$\tfrac{20}{51}e_1(t_{k+1}) + \tfrac{51h+9}{51}e_1(t_k) - \tfrac{71}{204}e_1(t_{k-1}) - \tfrac{39}{102}e_1(t_{k-2}) + \tfrac{9}{102}e_1(t_{k-3}) + \tfrac{13}{102}e_1(t_{k-4}) - \tfrac{11}{204}e_1(t_{k-5}) + O(g^5) = 0. \tag{16}$$

The characteristic equation of (16) is

$$\frac{20}{51}\xi^6 + \frac{51h+9}{51}\xi^5 - \frac{71}{204}\xi^4 - \frac{39}{102}\xi^3 + \frac{9}{102}\xi^2 + \frac{13}{102}\xi - \frac{11}{204} = 0. \tag{17}$$

In order to calculate the effective stepsize domain, (16) must be stable, i.e., six roots of (17) all inside a unit circle. For six-order equation (17) with parameter h, it may be difficult to solve all six roots inside a unit circle. However, by the application of bilinear transformation $\xi = (1+s)/(1-s)$ [48], we just make sure

Table 1: Routh stability criterion table for equation (18).

s^6	L_1	$136-255h$	$255h+408$	$51h$
s^5	L_2	544	$204h+102$	0
s^4	L_3	$\frac{20808h^2+29376h-16320}{102h-41}$	$51h$	
s^3	L_4	$\frac{249696h^3+80784h^2-32016h+20400}{1530h^2-615h+200}$	0	
s^2	L_5	$51h$		
s^1	L_6	0		
s^0	L_7			

$L_1 = 8-51h$

$L_2 = 82-204h$

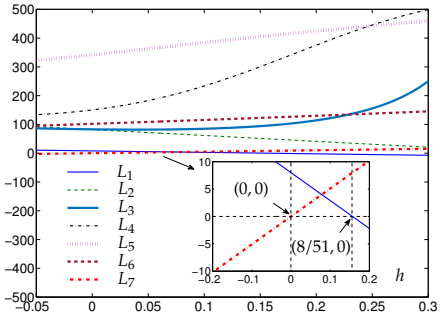
$L_3 = \frac{-26010h^2+10455h-3400}{102h-41}$

$L_4 = \frac{-249696h^3+580176h^2+2976h+30080}{1530h^2-615h+200}$

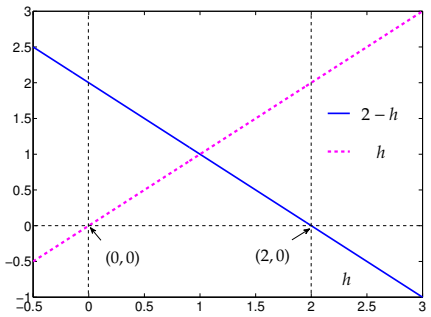
$L_5 = \frac{-795906h^4-2910519h^3-13640409h^2-816765h-642600}{15606h^3-36261h^2-186h-1880}$

$L_6 = \frac{21973248h^4+41478912h^3+30022512h^2+3285760h+1285200}{15606h^4+57069h^3+267459h^2+16015h+12600}$

$L_7 = 51h$



(a) $L_1 \sim L_7$ by SSZeaD-type SCS model (10)



(b) $2-h$ and h by EFF-type SCS model (13)

Figure 7: Trajectories of seven functions in the first column of Table 1 corresponding to SSZeaD-type SCS model (10), and trajectories of two functions corresponding to EFF-type SCS model (13).

that all coefficients of the first column in Routh stability criterion table are greater than zero [48]. Through the bilinear transformation, the following equation is gained:

$$(8-51h)s^6+(82-204h)s^5+(136-255h)s^4+544s^3+(255h+408)s^2+(204h+102)s+51h=0. \tag{18}$$

The Routh stability criterion table corresponding to equation (18) is presented in Table 1. According to Routh stability criterion [54], we finally obtain $0<h<8/51$. That is, the effective stepsize domain of the SSZeaD-type SCS model (10) is $(0, 8/51)$. The proof is thus completed. □

Proposition 4.2. *The effective stepsize domain of the EFF-type SCS model (13) is* $(0, 2)$.

Proof We use the EFF (12) to discretize the ZN design formula (15) directly, and the following equation can be obtained:

$$e_1(t_{k+1})-(1-h)e_1(t_k)+O(g^2)=0. \tag{19}$$

Table 2: Maximum steady-state absolute tracking errors of SSZeaD-type SCS model (10) and EFF-type SCS model (13) equipped with discrete-time ZN controller (11) using different values of stepsize h and sampling period g when tracking desired output $y_{\mathrm{d}}(t_k) = \sin(t_k) + 2\cos(t_k)$.

Stepsize	Sampling period	Model (10)	Model (13)
$h = 0.09$	$g = 0.2$ s	1.5828×10^{-3}	4.3240×10^{-1}
	$g = 0.02$ s	3.8192×10^{-8}	9.7322×10^{-3}
	$g = 0.002$ s	5.4415×10^{-13}	9.9360×10^{-5}
$h = 0.11$	$g = 0.2$ s	1.5210×10^{-3}	4.1874×10^{-1}
	$g = 0.02$ s	3.1494×10^{-8}	8.0211×10^{-3}
	$g = 0.002$ s	4.9943×10^{-13}	8.1300×10^{-5}
$h = 0.13$	$g = 0.2$ s	1.4555×10^{-3}	4.0297×10^{-1}
	$g = 0.02$ s	2.6770×10^{-8}	6.8155×10^{-3}
	$g = 0.002$ s	5.3946×10^{-13}	6.8796×10^{-5}
$h = 0.16$	$g = 0.2$ s	6.9327×10^{1}	3.7718×10^{-1}
	$g = 0.02$ s	∞	5.5572×10^{-3}
	$g = 0.002$ s	∞	5.5898×10^{-5}
$h = 2$	$g = 0.2$ s	∞	1.6959×10^{3}
	$g = 0.02$ s	∞	∞
	$g = 0.002$ s	∞	∞

Note that "∞" means that the item is above the order of 10^{13} in the table.

The characteristic equation of (19) is $\xi - 1 + h = 0$. In the same manner, by applying the bilinear transformation $\xi = (1 + s)/(1 - s)$, the following equation is obtained:

$$(2 - h)s + h = 0.$$

According to Routh stability criterion [54], we figure out $0 < h < 2$, which is also consistent with that in [55–57]. That is, the effective stepsize domain of the EFF-type SCS model (13) is $(0, 2)$. The proof is thus completed. □

5. Numerical Experiments and Comparisons

In this section, we further investigate the tracking precision of the SSZeaD-type SCS model (10) and the EFF-type SCS model (13) as well as the effectiveness of the designed controller applying to the tracking control of SCS. The numerical experiments are conducted with the constant parameters $\theta_0 = \theta_1 = \theta_2 = 100$ selected.

The task duration is set as 75 s and stepsize h is set as 0.1. The corresponding numerical results are presented in Figure 3 and Figure 4. Thereinto, Figure 3 describes the performance of controller (11) for the SSZeaD-type SCS model (10) tracking the desired output $y_{\mathrm{d}}(t_k) = \cos(t_k) + t_k$, with $g = 0.25$ s. After computational time is approximately 10 s (i.e., $k = 40$), the actual output $y(t_k)$ nearly completely tracks the desired output $y_{\mathrm{d}}(t_k)$ in Figure 3(a). Figure 3(b) describes the trajectories of discrete-time state variables $x_1(t_k)$ and $x_2(t_k)$. The control input $u(t_k)$ is presented in Figure 3(c), and absolute tracking error $|e(t_k)| = |y(t_k) - y_{\mathrm{d}}(t_k)|$ is presented in Figure 3(d). Besides, Figure 4 describes absolute tracking errors $|e(t_k)|$ when using the SSZeaD-type SCS model (10) and the EFF-type SCS model (13), respectively. Specifically, in Figure 4, we attempt to choose several different values of sampling period for comparison study, i.e., $g = 0.1$ s, 0.01 s, 0.25 s, 0.025 s and 0.0025 s. In Figure 4(a), the steady-state absolute tracking errors of the EFF-type SCS model (13) and the SSZeaD-type SCS model (10) are of orders 10^{-2} and 10^{-5}, respectively,

when $g = 0.1$ s. Comparatively, in Figure 4(b), the steady-state absolute tracking errors of the EFF-type SCS model (13) and the SSZeaD-type SCS model (10) are of orders 10^{-4} and 10^{-10}, respectively, when $g = 0.01$ s. This reflects the fact that the SSZeaD-type SCS model (10) has higher tracking precision as compared with the EFF-type SCS model (13). In Figure 4(c), the steady-state absolute tracking errors of the SSZeaD-type SCS model (10) are of orders 10^{-3}, 10^{-8} and 10^{-13}, when $g = 0.25$ s, 0.025 s and 0.0025 s, respectively. That is, as the value of g decreases by 10 times, the steady-state absolute tracking error of the SSZeaD-type SCS model (10) reduces by about 10^5 times, which approximately presents the $O(g^5)$ pattern. Comparatively, in Figure 4(d), the steady-state absolute tracking errors of the EFF-type SCS model (13) are of orders 10^{-1}, 10^{-3} and 10^{-5}, when $g = 0.25$ s, 0.025 s and 0.0025 s, respectively. In other words, as the value of g decreases by 10 times, the steady-state absolute tracking error of the EFF-type SCS model (13) reduces by about 10^2 times, which approximately presents the $O(g^2)$ pattern.

The task duration is set as 80 s and stepsize h is set as 0.12 with numerical results presented in Figure 5 and Figure 6. Besides, the desired output is $y_d(t_k) = \sin(t_k) + 2\cos(t_k)$. Thereinto, Figure 5 describes the performance of controller (11) for the SSZeaD-type SCS model (10), which is similar to that in Figure 3. Figure 6 describes absolute tracking errors $|e(t_k)|$ when using the SSZeaD-type SCS model (10) and the EFF-type SCS model (13), respectively, which is similar to that in Figure 4. Similarly, in Figure 6, we choose several different values of sampling period to undertake a comparative study, such as $g = 0.1$ s, 0.01 s, 0.2 s, 0.02 s and 0.002 s. In Figure 6(a), the steady-state absolute tracking errors of the EFF-type SCS model (13) and the SSZeaD-type SCS model (10) are of orders 10^{-2} and 10^{-5}, respectively, when $g = 0.1$ s. Comparatively, in Figure 6(b), the steady-state absolute tracking errors of the EFF-type SCS model (13) and the SSZeaD-type SCS model (10) are of orders 10^{-4} and 10^{-10}, respectively, when $g = 0.01$ s. The steady-state absolute tracking errors of the SSZeaD-type SCS model (10) are of orders 10^{-3}, 10^{-8} and 10^{-13}, when $g = 0.2$ s, 0.02 s and 0.002 s, respectively, in Figure 6(c), which approximately presents the $O(g^5)$ pattern. The steady-state absolute tracking errors of the EFF-type SCS model (13) are of orders 10^{-1}, 10^{-3} and 10^{-5}, when $g = 0.2$ s, 0.02 s and 0.002 s, respectively, in Figure 6(d), which approximately presents the $O(g^2)$ pattern. To sum up, the performance of the SSZeaD-type SCS model (10) in comparison with the EFF-type SCS model (13) is greater.

Besides, the maximum steady-state absolute tracking errors of the SSZeaD-type SCS model (10) and the EFF-type SCS model (13) are listed in Table 2, when tracking the desired output $y_d(t_k) = \sin(t_k) + 2\cos(t_k)$, with the different values of stepsize and sampling period. From Table 2, we observe that the maximum steady-state absolute tracking error of the SSZeaD-type SCS model (10) approximately presents the $O(g^5)$ pattern, when the value of stepsize is in $(0, 8/51)$. Besides, the maximum steady-state absolute tracking error of the EFF-type SCS model (13) approximately presents the $O(g^2)$ pattern, when the value of stepsize is in $(0, 2)$. However, when the value of stepsize is outside $(0, 8/51)$ (such as 0.16 and 2), the approximate $O(g^5)$ pattern of the SSZeaD-type SCS model (10) is no longer valid, which indirectly verifies the effectiveness of Proposition 4.1. Meanwhile, when the value of stepsize is outside $(0, 2)$ (such as 2), the approximate $O(g^2)$ pattern of the EFF-type SCS model (13) is no longer valid, which indirectly verifies the effectiveness of Proposition 4.2.

It is worth pointing out that the trajectories of seven functions $L_1 \sim L_7$ in the first column of Table 1 corresponding to the SSZeaD-type SCS model (10), and trajectories of two functions $2-h$ and h corresponding to the EFF-type SCS model (13) are shown in Figure 7(a) and Figure 7(b), respectively. From Figure 7(a), we can find the intersection of $L_1 \sim L_7$ being greater than zero, of which h is $(0, 8/51)$. That is, the effective stepsize domain of the SSZeaD-type SCS model (10) is $(0, 8/51)$, which accords with the obtained theoretical result in Proposition 4.1. From Figure 7(b), we can find that the intersection of $2 - h$ and h greater than zero is $(0, 2)$. In other words, the effective stepsize domain of the EFF-type SCS model (13) is $(0, 2)$, which conforms to the obtained theoretical result in Proposition 4.2.

6. Conclusion

In this paper, the proposed six-step Zhang et al discretization (SSZeaD) formula (1) with $O(g^4)$ precision has been used to discretize the ship course system (SCS) (8), and thus the SSZeaD-type SCS model (10) has been developed. For comparison, Euler forward formula (EFF) (12) with $O(g)$ precision has been also used to

discretize the SCS (8), and thus the EFF-type SCS model (13) has been obtained. Besides, the effective stepsize domains of these two discrete-time SCS models have been confirmed by the corresponding theoretical analyses. Finally, numerical experimental results have further validated the higher tracking precision of the SSZeaD-type SCS model (10) as compared with the EFF-type SCS model (13) in effective stepsize domains. One of our future research directions is to generalize the design method of the SSZeaD-type SCS model (10) with the proposed SSZeaD formula (1) to solve other discrete-time engineering problems. Developing the discretization-effective specific and general forms of multiple-step ZeaD formulas with higher precision can also be an interesting future research direction. Besides, the stepsize domain confirmation and optimum of the general SSZeaD-type SCS model can also be an interesting research topic, which will be studied in the future work.

References

[1] B. Qiu, Y. Zhang, Two new discrete-time neurodynamic algorithms applied to online future matrix inversion with nonsingular or sometimes-singular coefficient, IEEE Transactions on Cybernetics 49 (2019) 2032–2045.
[2] L. Xiao, B. Liao, S. Li, L. Ding, L. Jin, Design and analysis of FTZNN applied to real-time solution of nonstationary Lyapunov equation and tracking control of wheeled mobile manipulator, IEEE Transactions on Industrial Informatics 14 (2018) 98–105.
[3] Y.-J. Liu, S. Tong, Adaptive NN tracking control of uncertain nonlinear discrete-time systems with nonaffine dead-zone input, IEEE Transactions on Cybernetics 45 (2015) 497–505.
[4] A. M. Mohammed, S. Li, Dynamic neural networks for kinematic redundancy resolution of parallel Stewart platforms, IEEE Transactions on Cybernetics 46 (2016) 1538–1550.
[5] Z. Li, W. Ma, Z. Yin, H. Guo, Tracking control of time-varying knee exoskeleton disturbed by interaction torque, ISA Transactions 71 (2017) 458–466.
[6] Y. Zhang, J. Wang, Y. Xia, A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits, IEEE Transactions on Neural Networks 14 (2003) 658–667.
[7] B. Qiu, Y. Zhang, Z. Yang, New discrete-time ZNN models for least-squares solution of dynamic linear equation system with time-varying rank-deficient coefficient, IEEE Transactions on Neural Networks and Learning Systems 29 (2018) 5767–5776.
[8] B. Qiu, Y. Zhang, J. Guo, Z. Yang, X. Li, New five-step DTZD algorithm for future nonlinear minimization with quartic steady-state error pattern, Numerical Algorithms 81 (2019) 1043–1065.
[9] H. Yildirim, N. Kuruoglu, On the 3-parameter spatial motions in lorentzian 3-space, Filomat 32 (2018) 1183–1192.
[10] D. Chen, Y. Zhang, Jerk-level synchronous repetitive motion scheme with gradient-type and zeroing-type dynamics algorithms applied to dual-arm redundant robot system control, International Journal of Systems Science 48 (2017) 2713–2727.
[11] Y. Zhang, S. Li, J. Gui, X. Luo, Velocity-level control with compliance to acceleration-level constraints: a novel scheme for manipulator redundancy resolution, IEEE Transactions on Industrial Informatics 14 (2018) 921–930.
[12] Y. Shi, B. Qiu, D. Chen, J. Li, Y. Zhang Proposing and validation of a new four-point finite-difference formula with manipulator application, IEEE Transactions on Industrial Informatics 14 (2018) 1323–1333.
[13] S. Li, J. He, Y. Li, M. U. Rafique, Distributed recurrent neural networks for cooperative control of manipulators: A game-theoretic perspective, IEEE Transactions on Neural Networks and Learning Systems 28 (2017) 415–426.
[14] B. Liao, Q. Xiang, Discrete-time noise-suppressing Zhang neural network for dynamic quadratic programming with application to manipulators, Engineering Letters 25 (2017) 431–437.
[15] D. Chen, Y. Zhang, Minimum jerk norm scheme applied to obstacle avoidance of redundant robot arm with jerk bounded and feedback control, IET Control Theory and Applications 10 (2016) 1896–1903.
[16] Y. Zhang, S. Chen, S. Li, Z. Zhang, Adaptive projection neural network for kinematic control of redundant manipulators with unknown physical parameters, IEEE Transactions on Industrial Electronics 65 (2018) 4909–4920.
[17] Y. Zhang, B. Qiu, B. Liao, Z. Yang, Control of pendulum tracking (including swinging up) of IPC system using zeroing-gradient method, Nonlinear Dynamics 89 (2017) 1–25.
[18] X. Li, S. Song, Y. Guo, Robust finite-time tracking control for Euler-Lagrange systems with obstacle avoidance. Nonlinear Dynamics 93 (2018) 443–451.
[19] N. D. Manring, L. Muhi, R. C. Fales, V. S. Mehta, J. Kuehn, J. Peterson, Using feedback linearization to improve the tracking performance of a linear hydraulic-actuator, Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME 140 (2018) 1–7.
[20] Q. N. Li, R. N. Yang, Z. C. Liu, Adaptive tracking control for a class of nonlinear non-strict-feedback systems, Nonlinear Dynamics 88 (2017) 1537–1550.
[21] J. Li, M. Mao, Y. Zhang, D. Chen, Y. Yin, ZD, ZG and IOL controllers and comparisons for nonlinear system output tracking with DBZ problem conquered in different relative-degree cases, Asian Journal of Control 19 (2017) 1–14.
[22] Y. Zhang, D. Jiang, J. Wang, A recurrent neural network for solving Sylvester equation with time-varying coefficients, IEEE Transactions on Neural Networks 13 (2002) 1053–1063.
[23] L. Xiao, A finite-time recurrent neural network for solving online time-varying Sylvester matrix equation based on a new evolution formula, Nonlinear Dynamics 90 (2017) 1581–1591.
[24] B. Liao, Y. Zhang, From different ZFs to different ZNN models accelerated via Li activation functions to finite-time convergence for time-varying matrix pseudoinversion, Neurocomputing 133 (2014) 512–522.

[25] L. Xiao, R. Lu, Finite-time solution to nonlinear equation using recurrent neural dynamics with a specially-constructed activation function, Neurocomputing 151 (2015) 246–251.
[26] K. Chen, Implicit dynamic system for online simultaneous linear equations solving, Electronics Letters 49 (2013) 101–102.
[27] C. Yi, Y. Chen, X. Lan, Comparison on neural solvers for the Lyapunov matrix equation with stationary & nonstationary coefficients, Applied Mathematical Modelling 37 (2013) 2495–2502.
[28] Y. Zhang, L. Xiao, Z. Xiao, M. Mao, Zeroing Dynamics, Gradient Dynamics, and Newton Iterations, CRC Press, Boca Raton, 2015.
[29] K. Chen, Z. Zhang, A primal neural network for online equality-constrained quadratic programming, Cognitive Computation 10 (2018) 381–388.
[30] Y. Zhang, J. Guo, B. Qiu, Y. Shi, Z. Yang, New formula ZD4I*g*S_Q applied to solving future nonlinear systems of equations with abundant numerical experiment verification, In: Proceedings of the 17th International Conference on Control, Automation and Systems (2017) 9–14.
[31] K. Chen, Improved neural dynamics for online Sylvester equations solving, Information Processing Letters 116 (2016) 455–459.
[32] M. Liu, B. Liao, L. Ding, L. Xiao, Performance analyses of recurrent neural network models exploited for online time-varying nonlinear optimization, Computer Science and Information Systems 13 (2016) 691–705.
[33] L. Xiao, B. Liao, A convergence-accelerated Zhang neural network and its solution application to Lyapunov equation, Neurocomputing 193 (2016) 213–218.
[34] K. Chen, C. Yi, Robustness analysis of a hybrid of recursive neural dynamics for online matrix inversion, Applied Mathematics and Computation 273 (2016) 969–975.
[35] K. Chen, Recurrent implicit dynamics for online matrix inversion, Applied Mathematics and Computation 219 (2013) 10218–10224.
[36] D.-J. Li, D.-P. Li, Adaptive control via neural output feedback for a class of nonlinear discrete-time systems in a nested interconnected form, IEEE Transactions on Cybernetics 48 (2018) 2633–2642.
[37] Y.-J. Liu, S. Tong, Optimal control-based adaptive NN design for a class of nonlinear discrete-time block-triangular systems, IEEE Transactions on Cybernetics 46 (2016) 2670–2680.
[38] V. Lobov, K. Lobova, Automated control system of industrial dust suppression process, Metallurgical and Mining Industry 7 (2015) 53–59.
[39] Y.-J. Liu, Y. Gao, S. Tong, Y. Li, Fuzzy approximation-based adaptive backstepping optimal control for a class of nonlinear discrete-time systems with dead-zone, IEEE Transactions on Fuzzy Systems 24 (2016) 16–28.
[40] I. Svarc, R. Matousek, Discrete methods for continuous-time control systems, In: Proceedings of the 13th WSEAS international conference on Systems (2009) 232–235.
[41] H. C. Nejad, M. Farshad, O. Khayat, F. N. Rahatabad, Performance verification of a fuzzy wavelet neural network in the first order partial derivative approximation of nonlinear functions, Neural Processing Letters 43 (2016) 219–230.
[42] Y. Chen, C.-M. Chen, Numerical simulation with the second order compact approximation of first order derivative for the modified fractional diffusion equation, Applied Mathematics and Computation 320 (2018) 319–330.
[43] D. Guo, Z. Nie, L. Yan, Novel discrete-time Zhang neural network for time-varying matrix inversion, IEEE Transactions on Systems, Man, and Cybernetics: Systems 47 (2017) 2301–2310.
[44] L. Jin, Y. Zhang, Discrete-time Zhang neural network of $O(\tau^3)$ pattern for time-varying matrix pseudoinversion with application to manipulator motion generation, Neurocomputing 142 (2014) 165–173.
[45] B. Liao, Y. Zhang, 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 27 (2016) 225–237.
[46] Y. Zhang, L. Jin, D. Guo, Y. Yin, Y. Chou, Taylor-type 1-step-ahead numerical differentiation rule for first-order derivative approximation and ZNN discretization, Journal of Computational and Applied Mathematics 273 (2015) 29–40.
[47] D. Guo, X. Lin, Z. Su, S. Sun, Z. Huang, Design and analysis of two discrete-time ZD algorithms for time-varying nonlinear minimization, Numerical Algorithms 77 (2018) 23–36.
[48] C. Hu, X. Kang, Y. Zhang, Three-step general discrete-time Zhang neural network design and application to time-variant matrix inversion, Neurocomputing 306 (2018) 108–118.
[49] Y. Zhang, B. Qiu, L. Jin, D. Guo, Z. Yang, Infinitely many Zhang functions resulting in various ZNN models for time-varying matrix inversion with link to Drazin inverse, Information Processing Letters 115 (2015) 703–706.
[50] Q. Xiang, B. Liao, L. Xiao, L. Lin, S. Li, Discrete-time noise-tolerant Zhang neural network for dynamic matrix pseudoinversion, Soft Computing 23 (2019) 755–766.
[51] J. H. Mathews, K. D. Fink, Numerical Methods Using MATLAB, Prentice-Hall, New Jersey, 2005.
[52] E. Suli, D. F. Mayers, An Introduction to Numerical Analysis, Cambridge University Press, Oxford, 2003.
[53] D. F. Griffiths, D. J. Higham, Numerical Methods for Ordinary Differential Equations: Initial Value Problems, Springer, London, 2010.
[54] K. Ogata, Modern Control Engineering, Prentice-Hall, Inc, Englewood Cliffs, NJ, 2001.
[55] Y. Zhang, H. Qiu, C. Peng, Y. Shi, H. Tan, Simply and effectively proved square characteristics of discrete-time ZD solving systems of time-varying nonlinear equations, In: Proceedings of IEEE International Conference on Information and Automation (2015) 1457–1462.
[56] L. Jin, Y. Zhang, Discrete-time Zhang neural network for online time-varying nonlinear optimization with application to manipulator motion generation, IEEE Transactions on Neural Networks and Learning Systems 26 (2015) 1525–1531.
[57] L. Jin, Y. Zhang, Continuous and discrete Zhang dynamics for real-time varying nonlinear optimization, Numerical Algorithms 73 (2016) 115–140.