Numer Algor (2016) 73:115–140
DOI 10.1007/s11075-015-0088-1

ORIGINAL PAPER

Continuous and discrete Zhang dynamics for real-time varying nonlinear optimization

Long Jin[1,2,3] · **Yunong Zhang**[1,2,3]

Received: 30 June 2014 / Accepted: 8 December 2015 / Published online: 16 December 2015

Abstract Online solution of time-varying nonlinear optimization problems is considered an important issue in the fields of scientific and engineering research. In this study, the continuous-time derivative (CTD) model and two gradient dynamics (GD) models are developed for real-time varying nonlinear optimization (RTVNO). A continuous-time Zhang dynamics (CTZD) model is then generalized and investigated for RTVNO to remedy the weaknesses of CTD and GD models. For possible digital hardware realization, a discrete-time Zhang dynamics (DTZD) model, which can be further reduced to Newton-Raphson iteration (NRI), is also proposed and developed. Theoretical analyses indicate that the residual error of the CTZD model has an exponential convergence, and that the maximum steady-state residual error (MSSRE) of the DTZD model has an $O(\tau^2)$ pattern with τ denoting the sampling

This work is supported by the National Natural Science Foundation of China (with numbers 61473323), by the Foundation of Key Laboratory of Autonomous Systems and Networked Control, Ministry of Education, China (with number 2013A07), and also by the Science and Technology Program of Guangzhou, China (with number 2014J4100057). Besides, kindly note that both authors of the paper are jointly of the first authorship.

✉ Yunong Zhang
zhynong@mail.sysu.edu.cn

Long Jin
jinlongsysu@foxmail.com

[1] School of Information Science and Technology, Sun Yat-sen University (SYSU), Guangzhou 510006, China

[2] SYSU-CMU Shunde International Joint Research Institute, Shunde 528300, China

[3] Key Laboratory of Autonomous Systems and Networked Control, Ministry of Education, Guangzhou 510640, China

gap. Simulation and numerical results further illustrate the efficacy and advantages of the proposed CTZD and DTZD models for RTVNO.

Keywords Time-varying nonlinear optimization · Derivative method · Gradient dynamics (GD) · Zhang dynamics (ZD) · Discrete-time

1 Introduction

The online solution of optimization problems is considered a basic and important issue, and has been widely encountered in the scientific and engineering fields [1–9]. As one of the important branches, nonlinear optimization has been analyzed and widely applied in the engineering fields. Given its fundamental roles, numerous algorithms have been proposed and investigated to solve nonlinear optimization problems [7–12]. A family of scaled conjugate gradient algorithms for large-scale minimization was defined in [10] and compared with well known algorithms. In [11], Dai and Liao proposed a conjugate gradient method based on the secant condition, although their method did not necessarily generate a descent search direction. In [12], Yasushi and Hiroshi proposed conjugate gradient methods based on secant conditions that generate descent search directions, they also proved the convergence properties of these methods. By proposing a three-term conjugate gradient algorithm for large-scale optimization using subspace minimization technique, Andrei [3] proved that the search directions are descent, and that they satisfy the Dai-Liao conjugacy condition discussed in [11]. However, these algorithms were generalized and developed for solving static nonlinear optimization problems; thus, they might not be effective in handling time-varying optimization problems. Time-varying optimization is different from the static one as the former changes with the time; for example, the objective function relates to time t that is a unidirectional uniform (or say, even) stream parameter. Thus, the time derivative is important in obtaining an accurate solution for time-varying optimization.

Traditional numerical algorithms aiming at static optimization assume that the optimization problem does not change during the computational time. Thus, the calculated solution is directly used in the optimization problem after the calculation. A computational method designed intrinsically for static optimization, such as the Newton-Raphson iteration (NRI) and the methods presented in [7–12], can be viewed as a traditional numerical algorithm. In addition, the dynamical approach, specifically the neural dynamics originating from the neural network, has several potential advantages in real-time applications, including self-adaptation, parallel processing, distributed storage, and hardware implementation [13–17]. For example, for solving the optimization problem, gradient dynamics (GD), a typical dynamical approach, is often exploited by defining an ordinary differential equation (ODE), such that the solution of the optimization problem corresponds to a stable equilibrium point of the dynamical ODE system [18]. Therefore, the solution of the dynamical system forms a continuous trajectory, which begins from the initial point and ends at the solution of the original optimization problem. In addition to those common advantages, changing

with the problem variation and exploiting the time derivative, Zhang dynamics has predictive power to some extent. Thus, in this study, a novel continuous-time Zhang dynamics (CTZD) model is generalized, developed, and investigated for real-time varying nonlinear optimization (RTVNO). Moreover, for potential digital hardware (e.g., digital computer or digital circuit) realization, a discrete-time Zhang dynamics (DTZD) model, which can be further reduced to NRI as a special case of the DTZD model, is developed and studied for RTVNO.

The rest of this study is organized into five sections. In Section 2, the problem formulation of RTVNO is presented. Then, the continuous-time derivative (CTD) model and two GD models are developed as the bases of solution and comparison. The CTD model is sensitive to the initial value and the perturbations and is unable to solve RTVNO in a robust manner. Meanwhile, the GD models do not exploit the time derivative, thereby inducing lagging errors. To remedy the weaknesses of the CTD and GD models, the CTZD model is proposed. For the potential digital hardware realization, the DTZD model is developed and studied for RTVNO in Section 3. In Section 4, theoretical analyses indicate that the residual error of the CTZD model has an exponential convergence, and that the maximum steady-state residual error (MSSRE) of the DTZD model has an $O(\tau^2)$ pattern, with τ denoting the sampling gap. Theoretical analyses in this section also illustrate that the MSSRE of any method, which is designed intrinsically for solving static optimization problems and now employed for solving online discrete-time RTVNO, is $O(\tau)$. In Section 5, the simulation and numerical results of the RTVNO examples, which are synthesized by the proposed CTZD and DTZD models as well as other models comparatively, are presented. In Section 6, the conclusion of this study is presented, along with the final remarks.

2 Problem formulation and continuous-time solutions

To build the basis for further investigation, the problem formulation is presented initially in this section. Subsequently, the CTD, GD and CTZD models are presented, along with their design procedures.

2.1 Problem formulation

The aim is to solve, for each $t \in [0, +\infty)$, the problem given by

$$\min_{\mathbf{x}(t)\in\mathbb{R}^n} f(\mathbf{x}(t), t) \in \mathbb{R}, \tag{1}$$

where time-varying nonlinear mapping function $f(\cdot,\cdot) : \mathbb{R}^n \times [0, +\infty) \to \mathbb{R}$ is second-order differentiable and bounded below. We need to find the time-varying optimization solution $\mathbf{x}(t) \in \mathbb{R}^n$ in real time t, such that the smooth RTVNO function depicted in (1) achieves its minimum value at each time instant.

To obtain the online solution of RTVNO (1), we primarily define $\mathbf{g}(\mathbf{x}(t), t)$ as

$$\begin{aligned}\mathbf{g}(\mathbf{x}(t), t) &= \frac{\partial f(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} = \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \cdots, \frac{\partial f}{\partial x_n}\right]^{\mathrm{T}} \\ &= [g_1(\mathbf{x}(t), t), g_2(\mathbf{x}(t), t), \cdots, g_n(\mathbf{x}(t), t)]^{\mathrm{T}} \in \mathbb{R}^n, \end{aligned} \tag{2}$$

where $\partial f/\partial x_i = \partial f(\mathbf{x}(t), t)/\partial x_i(t) = g_i(\mathbf{x}(t), t)$, $\forall i \in \{1, 2, \cdots, n\}$, with $\mathbf{g}(\cdot, \cdot)$ denoting the differentiable nonlinear mapping function derived from $f(\mathbf{x}(t), t)$, and with superscript $^{\mathrm{T}}$ denoting the transpose operator of a vector or matrix. In addition, we define the time-varying set

$$\Omega^*(t) = \{(t, \mathbf{x}^*(t)) | \partial f(\mathbf{x}^*(t), t)/\partial \mathbf{x}^*(t) = 0\}$$

for time instant $t \in [0, +\infty)$. To achieve the desired path $\mathbf{x}^*(t)$ of RTVNO (1), $\mathbf{g}(\mathbf{x}(t), t)$ should be set to zero.

2.2 CTD and GD models

To solve RTVNO (1), an intuitive method is to exploit the derivative approach. Specifically, to achieve the desired path $\mathbf{x}^*(t)$ on which $\mathbf{g}(\mathbf{x}(t), t) = 0$, the derivative of the gradient $\mathbf{g}(\mathbf{x}(t), t)$ with respect to time t should be zero for each time instant $t \in [0, +\infty)$, that is,

$$0 = \frac{\mathrm{d}\mathbf{g}(\mathbf{x}(t), t)}{\mathrm{d}t} = \frac{\partial \mathbf{g}(\mathbf{x}(t), t)}{\partial t} + \frac{\partial \mathbf{g}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} \frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \dot{\mathbf{g}}_t(\mathbf{x}(t), t) + H(\mathbf{x}(t), t) \frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t}, \tag{3}$$

where Hessian matrix $H(\mathbf{x}(t), t)$ and time-derivative vector $\dot{\mathbf{g}}_t(\mathbf{x}(t), t)$ are defined, respectively, as

$$H(\mathbf{x}(t), t) = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} & \cdots & \frac{\partial g_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_n}{\partial x_1} & \frac{\partial g_n}{\partial x_2} & \cdots & \frac{\partial g_n}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_n} \end{bmatrix} \in \mathbb{R}^{n \times n},$$

$$\dot{\mathbf{g}}_t(\mathbf{x}(t), t) = \frac{\partial \mathbf{g}(\mathbf{x}(t), t)}{\partial t} = \frac{\partial^2 f(\mathbf{x}(t), t)}{\partial \mathbf{x}(t) \partial t} \in \mathbb{R}^n,$$

with g_i and x_i denoting respectively the ith elements of $\mathbf{g}(\mathbf{x}(t), t)$ and $\mathbf{x}(t)$, $\forall i \in \{1, \cdots, n\}$. In this study, we assume that Hessian matrix $H(\mathbf{x}(t), t)$ is nonsingular for any t. By rearranging equation (3), we obtain the CTD model expressed as

$$\dot{\mathbf{x}}(t) = -H^{-1}(\mathbf{x}(t), t) \frac{\partial^2 f(\mathbf{x}(t), t)}{\partial \mathbf{x}(t) \partial t}, \quad \text{with } \mathbf{x}(0) = \mathbf{x}^*(0). \tag{4}$$

Therefore, assuming that equation (4) satisfies the Lipschitz condition [19] and that Hessian matrix $H(\mathbf{x}(t), t)$ is positive definite, then the solution to equation (4) corresponds to the theoretical solution to RTVNO (1), which is concluded from the ensuing Theorem 1.

Theorem 1 *Assume that the n-variable bounded below function* $f(\mathbf{x}(t), t)$ *(with* $\mathbf{x}(t) \in \mathbb{R}^n$ *and* $n \geqslant 2$*) has the second order continuous partial derivatives in the neighborhood of point* $\mathrm{p}^* = \mathbf{x}^*(t_\mathrm{p}) = [x_1^*(t_\mathrm{p}), x_2^*(t_\mathrm{p}), \cdots, x_n^*(t_\mathrm{p})]^\mathrm{T}$ *at time instant* t_p*, which satisfies* $f_{x_i}(\mathrm{p}^*, t_\mathrm{p}) = \partial f(\mathbf{x}(t), t)/\partial x_i(t)|_{t=t_\mathrm{p},\ \mathbf{x}(t_\mathrm{p})=\mathrm{p}^*} = 0,\ \forall i \in \{1, 2, \cdots, n\}$*. If the corresponding Hessian matrix* $H(\mathrm{p}^*, t_\mathrm{p})$ *is positive definite, then* $f(\mathbf{x}(t), t)$ *achieves its minimum at point* p^* *of time instant* t_p*.*

Proof The second order Taylor expansion of $f(\mathbf{x}(t), t)$ with Lagrange remainder at time instant t_p around point p^* is formulated as follows [19]:

$$\begin{aligned} f(\mathbf{x}(t_\mathrm{p}), t_\mathrm{p}) &= f(\mathrm{p}^*, t_\mathrm{p}) + \nabla^\mathrm{T} f(\mathrm{p}^*, t_\mathrm{p})\Delta_\mathbf{x} + \frac{1}{2}\Delta_\mathbf{x}^\mathrm{T}\nabla^2 f(\mathrm{p}^*, t_\mathrm{p})\Delta_\mathbf{x} + r(\mathrm{p}^*, t_\mathrm{p}) \\ &= f(\mathrm{p}^*, t_\mathrm{p}) + \frac{1}{2}\Delta_\mathbf{x}^\mathrm{T}\nabla^2 f(\mathrm{p}^*, t_\mathrm{p})\Delta_\mathbf{x} + r(\mathrm{p}^*, t_\mathrm{p}) \end{aligned}$$

with

$$r(\mathrm{p}^*, t_\mathrm{p}) = \frac{1}{3!}\left(\frac{\partial}{\partial x_1(t)}\Delta_{x_1} + \cdots + \frac{\partial}{\partial x_n(t)}\Delta_{x_n}\right)^3 f(\mathbf{x}(t), t)|_{t=t_\mathrm{p},\ \mathbf{x}(t)=\theta_\mathbf{x}},$$

where $\Delta_\mathbf{x} = \mathbf{x}(t_\mathrm{p}) - \mathrm{p}^*$ with Δ_{x_i} being its ith element. Meanwhile, the ith element of $\theta_\mathbf{x}$ is $(x_i^*(t_\mathrm{p}) + \theta_i(x_i(t_\mathrm{p}) - x_i^*(t_\mathrm{p})))$ with $0 < \theta_i < 1,\ \forall i \in \{1, 2, \cdots, n\}$. Note that term $\nabla^\mathrm{T} f(\mathrm{p}^*, t_\mathrm{p})\Delta_\mathbf{x} = 0$ because $f_{x_i}(\mathrm{p}^*, t_\mathrm{p}) = 0, \forall i \in \{1, 2, \cdots, n\}$. For simplicity and clarity, we have the following definitions:

$$\begin{aligned} \Delta_f(\mathrm{p}^*, t_\mathrm{p}) &= f(\mathbf{x}(t_\mathrm{p}), t_\mathrm{p}) - f(\mathrm{p}^*, t_\mathrm{p}), \\ q(\mathrm{p}^*, t_\mathrm{p}) &= \Delta_\mathbf{x}^\mathrm{T}\nabla^2 f(\mathrm{p}^*, t_\mathrm{p})\Delta_\mathbf{x} = \Delta_\mathbf{x}^\mathrm{T} H(\mathrm{p}^*, t_\mathrm{p})\Delta_\mathbf{x}. \end{aligned}$$

Subsequently, we obtain $\Delta_f(\mathrm{p}^*, t_\mathrm{p}) = (\mathrm{p}^*, t_\mathrm{p})/2 + r(\mathrm{p}^*, t_\mathrm{p})$. When $\mathbf{x}(t_\mathrm{p}) \to \mathrm{p}^*$, $r(\mathrm{p}^*, t_\mathrm{p}) \to 0$. Term $r(\mathrm{p}^*, t_\mathrm{p})$ is considerably smaller than $q(\mathrm{p}^*, t_\mathrm{p})/2$. Thus, the sign of $q(\mathrm{p}^*, t_\mathrm{p})$ is consistent with that of $\Delta_f(\mathrm{p}^*, t_\mathrm{p})$. By contrast, $q(\mathrm{p}^*, t_\mathrm{p})$ is a quadratic term related to $\Delta_\mathbf{x}$; thus, when $H(\mathrm{p}^*, t_\mathrm{p})$ is positive definite, $q(\mathrm{p}^*, t_\mathrm{p}) > 0$ always holds true. In addition, for $\Delta_f(\mathrm{p}^*, t_\mathrm{p}) > 0$, we have $f(\mathbf{x}(t_\mathrm{p}), t_\mathrm{p}) > f(\mathrm{p}^*, t_\mathrm{p})$. Hence, if the corresponding Hessian matrix $H(\mathrm{p}^*, t_\mathrm{p})$ is positive definite, then function $f(\mathbf{x}(t), t)$ achieves its minimum at point p^* of time instant t_p. The proof is complete. □

The gradient-descent method is a conventional approach that is frequently used to solve the static optimization problem. For comparative purposes, two models of gradient dynamics are developed and exploited to solve RTVNO (1). By following gradient-descent design method, we could initially define a norm-based or square-based energy function, such as $\varepsilon = \|\partial f(\mathbf{x})/\partial\mathbf{x}\|_2^2/2$. Subsequently, a typical continuous-time adaptation rule based on the negative-gradient information leads to the differential equation given by

$$\dot{\mathbf{x}}(t) = -\gamma\frac{\partial\varepsilon}{\partial\mathbf{x}} = -\gamma H^\mathrm{T}(\mathbf{x}(t), t)\frac{\partial f(\mathbf{x}(t), t)}{\partial\mathbf{x}(t)} = -\gamma H(\mathbf{x}(t), t)\frac{\partial f(\mathbf{x}(t), t)}{\partial\mathbf{x}(t)}, \qquad (5)$$

where design parameter $\gamma > 0$ is used to scale the convergence rate of the GD model, which should be selected appropriately for numerical stability. In addition, another GD model can be directly derived from $\varepsilon = f(\mathbf{x})$, and is formulated as

$$\dot{\mathbf{x}}(t) = -\gamma \frac{\partial \varepsilon}{\partial \mathbf{x}} = -\gamma \frac{\partial f(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}. \tag{6}$$

For presentation convenience, GD models (5) and (6) are termed GD-1 model and GD-2 model, respectively, in this study. Assuming that GD-1 model (5) [or GD-2 model (6)] satisfies Lipschitz condition [19] and that Hessian matrix $H(\mathbf{x}(t), t)$ is positive definite, Theorem 1 proves that the solution to the GD-1 model (5) [or GD-2 model (6)] corresponds to the theoretical solution to RTVNO (1). CTD model (4) and GD-1 model (5) [or GD-2 model (6)] are investigated further by comparing their respective performances for the online solution of RTVNO (1). On the one hand, CTD model (4) exploits the time derivative [i.e., term $\partial^2 f(\mathbf{x}(t), t)/\partial \mathbf{x}(t)\partial t$] during the real-time solution process, whereas GD-1 model (5) [or GD-2 model (6)] does not employ such important information. As a result, GD-1 model (5) [or GD-2 model (6)] adapts to the change of coefficients in a posterior passive manner, making them unsuitable for solving RTVNO (1) in a predictive and accurate manner. On the other hand, GD-1 model (5) [or GD-2 model (6)] exploits the error-feedback information [i.e., term $\partial f(\mathbf{x}(t), t)/\partial \mathbf{x}(t)$] during the real-time solution process, whereas CTD model (4) does not. Supposing that $\mathbf{x}_0$ is not found accurately in advance for various reasons, including round-off errors, then $\mathbf{g}(\mathbf{x}_0, t_0) \neq 0$, and $\mathbf{g}(\mathbf{x}(t), t)$ would in general be propagated and remain nonzero at every t as a result of solving equation (4). Even if $\mathbf{x}_0$ is found accurately, perturbations or computational errors (e.g., truncation or round-off errors) may still occur in the problem data during the process, resulting in $\mathbf{g}(\mathbf{x}(t), t) \neq 0$ for all $t \geq 0$, in general. From the viewpoint of control, the norm of the error of a method can be interpreted as a measure of the distance of the current solution $\mathbf{x}(t)$ from the zero of $\mathbf{g}(\mathbf{x}(t), t)$, and the error can be used as an input to obtain a robust solution despite the occurrence of perturbations or computational errors [20–22]. Therefore, the error-feedback information is indispensable in achieving a robust and accurate solution for the online solution of RTVNO (1). Thus, a new method combining the advantages of the CTD model (4) and GD-1 model (5) [or GD-2 model (6)] while remedying their weaknesses is required.

2.3 CTZD model

The CTZD model proposed in this subsection combines the error-feedback information and the time-derivative information, thereby obtaining good robustness and high accuracy for the online solution of RTVNO (1).

To monitor and control the solving process of RTVNO (1), we define the following vector-valued indefinite error-function as

$$\mathbf{e}(t) = [e_1(t), e_2(t), \cdots, e_n(t)]^{\mathrm{T}} = \mathbf{g}(\mathbf{x}(t), t), \tag{7}$$

where $e_j(t) = g_j(\mathbf{x}(t), t)$ is the jth element of $\mathbf{e}(t)$, $\forall j \in \{1, 2, \cdots, n\}$. Evidently, if the error-function $\mathbf{e}(t)$ converges to zero, then the solution $\mathbf{x}(t)$ converges to the

desired path $\mathbf{x}^*(t)$. To make every element of $\mathbf{e}(t)$ converge to zero, a CTZD design formula is adopted as

$$\frac{\mathrm{d}\mathbf{e}(t)}{\mathrm{d}t} = -\gamma\Phi(\mathbf{e}(t)), \quad \text{i.e.,} \quad \frac{\mathrm{d}\mathbf{g}(\mathbf{x}(t),t)}{\mathrm{d}t} = -\gamma\Phi(\mathbf{g}(\mathbf{x}(t),t)), \tag{8}$$

where design parameter $\gamma > 0$ is defined the same way as before. In addition, $\Phi(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ denotes a vector array of activation function, and the element of $\Phi(\cdot)$ is denoted by $\phi(\cdot)$. As proven in the ensuing Theorems 2 and 3, CTZD design formula (8) is asymptotically stable (specifically, exponentially stable) and drives the residual error $\mathbf{e}(t)$ during the real-time process to zero in an exponential manner. Generally speaking, any monotonically increasing odd activation function $\phi(\cdot)$ can be used to construct the dynamics model. Different choices of activation function $\phi(\cdot)$ may lead to different convergence performance. In this study, three activation functions (i.e., the linear, power-sigmoid and hyperbolic sine activation functions) are applied to construct the CTZD model. These functions are expressed as

– the linear (li) activation function:

$$\phi_{\mathrm{li}}(e_i) = e_i;$$

– the power-sigmoid (ps) activation function (with $p = 3$ and $\xi = 4$):

$$\phi_{\mathrm{ps}}(e_i) = \begin{cases} \frac{1+\exp(-\xi)}{1-\exp(-\xi)}\frac{1-\exp(-\xi e_i)}{1+\exp(-\xi e_i)}, & \text{if } |e_i| < 1, \\ e_i^p, & \text{if } |e_i| \geqslant 1; \end{cases}$$

– and the hyperbolic sine (hs) activation function (with $m = 3$):

$$\phi_{\mathrm{hs}}(e_i) = \frac{\exp(e_i m)}{2} - \frac{\exp(-e_i m)}{2}.$$

Expanding CTZD design formula (8) obtains the differential equation given by

$$H(\mathbf{x}(t),t)\dot{\mathbf{x}}(t) = -\gamma\Phi(\mathbf{g}(\mathbf{x}(t),t)) - \dot{\mathbf{g}}_t(\mathbf{x}(t),t). \tag{9}$$

If Hessian matrix $H(\mathbf{x}(t),t)$ is nonsingular, the above dynamics can be rewritten as

$$\begin{aligned} \dot{\mathbf{x}}(t) &= -H^{-1}(\mathbf{x}(t),t)\left(\gamma\Phi(\mathbf{g}(\mathbf{x}(t),t)) + \dot{\mathbf{g}}_t(\mathbf{x}(t),t)\right) \\ &= -H^{-1}(\mathbf{x}(t),t)\left(\gamma\Phi\left(\frac{\partial f(\mathbf{x}(t),t)}{\partial \mathbf{x}(t)}\right) + \frac{\partial^2 f(\mathbf{x}(t),t)}{\partial \mathbf{x}(t)\partial t}\right), \end{aligned} \tag{10}$$

where $\mathbf{x}(t)$, starting from randomly-generated initial condition $\mathbf{x}(0) \in \mathbb{R}^n$, denotes the state of the dynamics corresponding to $\mathbf{x}^*(t) \in \mathbb{R}^n$ of RTVNO (1). Solving CTZD model (10) instead of CTD model (4) is then expected to take care of the $\mathbf{e}(t)$ incurred during the process, by driving it to zero asymptotically (specifically, exponentially). In addition, with the use of error-feedback information and time-derivative information, CTZD model (10) is expected to solve RTVNO (1) in a robust and accurate manner. If Hessian matrix $H(\mathbf{x}(t),t)$ is positive definite, then $\mathbf{x}(t)$ corresponds to the theoretical solution to RTVNO (1).

As stated in Section 2.2, CTD model (4) and GD model (5) [or (6)] should satisfy Lipschitz condition for the online solution of RTVNO (1). By contrast, the proposed CTZD model (10) does not require such an assumption because it satisfies

the Lipschitz condition naturally. For simplicity and for improved understanding, we investigate CTZD design formula (8) for the scalar case, that is, $\dot{e}_i(t) = -\gamma\phi(e_i(t))$ with $e_i(t)$ being the ith element of $\mathbf{e}(t)$. In the ensuing Theorems 2 and 3, as proven by Lyapunov theory, the dynamic system $\dot{e}_i(t) = -\gamma\phi(e_i(t))$ possesses the global exponential convergence property. The resultant $e_i(t)$ of such a dynamic system, starting from any randomly-generated initial state $e_i(0)$, exponentially converges to zero. Hence, for a given initial state $e_i(0)$, $e_i(t)$ is bounded within interval $[e_i(0), 0]$ [corresponding to $e_i(0) < 0$] or $[0, e_i(0)]$ [corresponding to $e_i(0) > 0$]. For the dynamic system $\dot{e}_i(t) = -\gamma\phi(e_i(t))$, $e_i(t)$ is bounded within interval $[-|e_i(0)|, |e_i(0)|]$ (where symbol $|\cdot|$ denotes the absolute value of a scalar). Based on the aforementioned analysis, we further obtain the Lipschitz constant L for the general activation function situation, which is given by

$$L = \max\left\{\phi_{e_i}(e_i) = \frac{\partial\phi(e_i)}{\partial e_i}\right\}, \quad \text{with } e_i \in [-|e_i(0)|, |e_i(0)|]\,.$$

That is, as the activation function $\phi(\cdot)$ exploited in this work satisfies Lipschitz condition, the dynamic system $\dot{e}_i(t) = -\gamma\phi(e_i(t))$ with any initial value $e_i(0)$ has a unique zero-converging solution $e_i(t)$, of which the existence and uniqueness of the CTZD solution are guaranteed naturally.

Specifically, based on these results, we have the following corresponding Lipschitz constants for the linear, power-sigmoid, and hyperbolic-sine activation functions employed in this work:

– For the linear activation function,

$$L_{\mathrm{li}} = \max\{\phi_{e_i}(e_i) = 1\} = 1.$$

– For the power-sigmoid activation function,

$$\begin{aligned} L_{\mathrm{ps}} &= \begin{cases} \max\left\{\phi_{e_i}(e_i) = \frac{1+\exp(-4)}{1-\exp(-4)}\frac{8\exp(-4e_i)}{(1+\exp(-4e_i))^2}\right\}, & \text{if } |e_i| < 1, \\ \max\left\{\phi_{e_i}(e_i) = 3e_i^2\right\}, & \text{if } |e_i| \geqslant 1; \end{cases} \\ &= \begin{cases} 2.0746, & \text{if } |e_i(0)| < 1, \\ 3e_i^2(0), & \text{if } |e_i(0)| \geqslant 1. \end{cases} \end{aligned}$$

– For the hyperbolic-sine activation function,

$$\begin{aligned} L_{\mathrm{hs}} &= \max\left\{\phi_{e_i}(e_i) = 3(\exp(3e_i) + \exp(-3e_i))/2\right\} \\ &= \frac{3(\exp(3|e_i(0)|) + \exp(-3|e_i(0)|))}{2}. \end{aligned}$$

Each activation function satisfies Lipschitz condition; hence, the existence and uniqueness of the CTZD solution are guaranteed naturally. This means that, with a given initial condition, the proposed CTZD model (10) derived from CTZD design formula (8) generates a unique solution, which corresponds to the theoretical solution to RTVNO (1), provided that Hessian matrix $H(\mathbf{x}(t), t)$ is positive definite.

3 Discrete-time solutions

For potential digital hardware (e.g., digital computer or digital circuit) realization, a DTZD model is generalized and developed in this section. Consider the following discrete-time RTVNO with computational time interval $[t_k, t_{k+1}) \in [0, +\infty)$ given by

$$\min_{\mathbf{x}(t_{k+1})\in\mathbb{R}^n} f\left(\mathbf{x}(t_{k+1}), t_{k+1}\right) \in \mathbb{R} \tag{11}$$

where $f\left(\mathbf{x}(t_{k+1}), t_{k+1}\right)$ is assumed to be generated or measured from the smoothly time-varying signal $f\left(\mathbf{x}(t), t\right)$ by sampling at time instant $t = (k+1)\tau$ (denoted as t_{k+1}). In addition, $\tau > 0$ denotes the sampling gap, and $k = 0, 1, 2, \cdots$ denotes the update index.

In the online solution process of discrete-time RTVNO (11), computation has to be performed based on the present and/or previous data. For example, at time instant t_k, only the known information can be used, such as $\mathbf{x}(t_k)$, $f(\mathbf{x}(t_k), t_k)$ and its derivatives, instead of unknown information, such as $f(\mathbf{x}(t_{k+1}), t_{k+1})$ and its derivatives, to compute the unknown vector $\mathbf{x}(t_{k+1})$ during the computational time interval $[t_k, t_{k+1})$. Thus, through the present and/or previous data, the objective is to determine the unknown vector $\mathbf{x}(t_{k+1})$ of the unknown function $f(\mathbf{x}(t_{k+1}), t_{k+1})$ during $[t_k, t_{k+1})$, such that (11) achieves its minimal value at each time instant. At the start of the computation (i.e., at time instant $t_0 = 0$), we could not compute $\mathbf{x}(t_0)$ based on pervious data; thus $\mathbf{x}(t_0)$ can be randomly generated or directly set as $\mathbf{x}^*(t_0)$, which is computed in advance for achieving better performance, where $\mathbf{x}^*(t_0)$ is the theoretical solution of (11) at time instant $t_0 = 0$.

Corresponding to Section 2.3, we define an updating function $\mathbf{u}(\mathbf{x}, t)$ as

$$\mathbf{u}(\mathbf{x}(t), t) = -H^{-1}(\mathbf{x}(t), t)\left(\gamma\Phi\left(\frac{\partial f(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}\right) + \frac{\partial^2 f(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)\partial t}\right).$$

Subsequently, to discretize the CTZD model, we use Euler method [24] and obtain

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_k) + \tau\mathbf{u}(\mathbf{x}(t_k), t_k),\ \ t_{k+1} = t_k + \tau.$$

Exploiting the linear activation function array, we obtain the equation given by

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_k) - H^{-1}(\mathbf{x}(t_k), t_k)\left(h\mathbf{g}(\mathbf{x}(t_k), t_k) + \tau\dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k)\right), \tag{12}$$

where step-size $h = \tau\gamma > 0$. Unfortunately, in numerous engineering applications, determining the analytical form or numerical value of $\dot{\mathbf{g}}_t(\mathbf{x}(t), t)$ may be difficult (or even impossible). Thus, $\dot{\mathbf{g}}_t(\mathbf{x}(t), t)$ may generally be estimated from $\mathbf{g}(\mathbf{x}(t), t)$ by employing Euler backward-difference rule expressed as

$$\dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k) \approx \frac{\mathbf{g}\left(\mathbf{x}(t_k), t_k\right) - \mathbf{g}\left(\mathbf{x}(t_k), t_{k-1}\right)}{\tau}. \tag{13}$$

From CTZD model (10), we derive the DTZD model for RTVNO (11) as

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_k) - H^{-1}(\mathbf{x}(t_k), t_k)\left((h+1)\mathbf{g}(\mathbf{x}(t_k), t_k) - \mathbf{g}(\mathbf{x}(t_k), t_{k-1})\right). \tag{14}$$

Without $\mathbf{g}(\mathbf{x}(t_0), t_{-1})$, we could not approximate $\dot{\mathbf{g}}_t(\mathbf{x}(t_0), t_0)$ using (13). Therefore, we simply set $\dot{\mathbf{g}}_t(\mathbf{x}(t_0), t_0) = 0$ and use $\mathbf{x}(t_1) = \mathbf{x}(t_0) - hH^{-1}(\mathbf{x}(t_0), t_0)\mathbf{g}(\mathbf{x}(t_0), t_0)$ to initialize DTZD model (14). If the power-sigmoid or the hyperbolic sine activation function array is applied to construct DTZD model, then we have

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_k) - H^{-1}(\mathbf{x}(t_k), t_k)\left(h\Phi(\mathbf{g}(\mathbf{x}(t_k), t_k)) + \mathbf{g}(\mathbf{x}(t_k), t_k) - \mathbf{g}(\mathbf{x}(t_k), t_{k-1})\right), \tag{15}$$

where $\Phi = \Phi_{\mathrm{ps}}$ or $\Phi = \Phi_{\mathrm{hs}}$. In addition, if NRI model [24] is exploited to solve RTVNO (11), we have

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_k) - H^{-1}(\mathbf{x}(t_k), t_k)\mathbf{g}(\mathbf{x}(t_k), t_k), \tag{16}$$

which can be viewed as a special case of DTZD model (14), with $h = 1$ and without using the approximation of $\dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k)$. Moreover, the discrete form of CTD model (4) via Euler method [i.e., the discrete-time derivative (DTD) model] can be obtained correspondingly as

$$\mathbf{x}(t_{k+1}) = \mathbf{x}(t_k) - H^{-1}(\mathbf{x}(t_k), t_k)\left(\mathbf{g}(\mathbf{x}(t_k), t_k) - \mathbf{g}(\mathbf{x}(t_k), t_{k-1})\right) \tag{17}$$

with $\mathbf{x}(t_0) = \mathbf{x}^*(0)$. Therefore, by exploiting the approximation of time-derivative information and the error-feedback information in a combinative manner, DTZD model (14) is expected to have better performance than NRI model (16) and DTD model (17).

4 Theoretical analyses and results

In this section, theoretical analyses and results of CTZD model (10) and DTZD model (14) for solving RTVNO are presented as follows.

Theorem 2 *Consider RTVNO* (1)*. Suppose that the corresponding Hessian matrix is positive definite. If monotonically increasing odd activation-function array* $\Phi(\cdot)$ *is employed, then state* $\mathbf{x}(t) \in \mathbb{R}^n$ *of CTZD model* (10)*, starting from randomly generated initial state* $\mathbf{x}(0) \in \mathbb{R}^n$*, converges to the theoretical solution to RTVNO* (1) *as* $t \to +\infty$.

Proof Define a Lyapunov function candidate [25] as

$$\begin{aligned} V(\mathbf{x}(t), t) &= \frac{1}{2}\|\mathbf{g}(\mathbf{x}(t), t)\|_2^2 = \frac{1}{2}\mathbf{g}^{\mathrm{T}}(\mathbf{x}(t), t)\mathbf{g}(\mathbf{x}(t), t) \\ &= \frac{1}{2}\sum_{j=1}^{n} g_j^2(\mathbf{x}(t), t) = \frac{1}{2}\sum_{j=1}^{n}\left(\frac{\partial f(\mathbf{x}(t), t)}{\partial x_j(t)}\right)^2 \geqslant 0, \end{aligned} \tag{18}$$

which guarantees the positive-definiteness of Lyapunov function candidate $V(\mathbf{x}(t), t)$; that is, $V(\mathbf{x}(t), t) > 0$ for any $g_j(\mathbf{x}(t), t) \neq 0$, and $V(\mathbf{x}(t), t) = 0$ only

for each $g_j(\mathbf{x}(t),t)=0$, with $j\in\{1,2,\cdots,n\}$. The time derivative $\dot{V}(\mathbf{x}(t),t)$ along the element trajectories of CTZD model (10) becomes

$$\begin{aligned}\dot{V}(\mathbf{x}(t),t) &= \frac{\mathrm{d}V(\mathbf{x}(t),t)}{\mathrm{d}t}=\sum_{j=1}^{n} g_j(\mathbf{x}(t),t)\frac{\mathrm{d}g_j(\mathbf{x}(t),t)}{\mathrm{d}t}=-\gamma\sum_{j=1}^{n} g_j(\mathbf{x}(t),t)\phi(g_j(\mathbf{x}(t),t))\\ &= -\gamma\sum_{j=1}^{n}\frac{\partial f(\mathbf{x}(t),t)}{\partial x_j(t)}\phi\left(\frac{\partial f(\mathbf{x}(t),t)}{\partial x_j(t)}\right). \end{aligned} \tag{19}$$

As the element of $\Phi(\cdot)$ [i.e., $\phi(\cdot)$] is a monotonically increasing odd function, we get

$$\phi\left(\frac{\partial f(\mathbf{x}(t),t)}{\partial x_j(t)}\right)\begin{cases} >0, & \text{if } \partial f(\mathbf{x}(t),t)/\partial x_j(t)>0,\\ =0, & \text{if } \partial f(\mathbf{x}(t),t)/\partial x_j(t)=0,\\ <0, & \text{if } \partial f(\mathbf{x}(t),t)/\partial x_j(t)<0.\end{cases}$$

Hence we have

$$\frac{\partial f(\mathbf{x}(t),t)}{\partial x_j(t)}\phi\left(\frac{\partial f(\mathbf{x}(t),t)}{\partial x_j(t)}\right)\begin{cases} >0, & \text{if } \partial f(\mathbf{x}(t),t)/\partial x_j(t)\neq 0,\\ =0, & \text{if } \partial f(\mathbf{x}(t),t)/\partial x_j(t)=0,\end{cases}$$

which guarantees the final negative-definiteness of $\dot{V}(\mathbf{x}(t),t)$. The Lyapunov function candidate (18) is positive definite and its time derivative (19) is negative definite, thus satisfying the requirement of Lyapunov theory. Hence, based on Lyapunov theory, starting from randomly generated initial state $\mathbf{x}(0)$, state $\mathbf{x}(t)$ of CTZD model (10) converges to the desired path $\mathbf{x}^*(t)\in\mathbb{R}^n$. According to Theorem 1, if the corresponding Hessian matrix $H(\mathbf{x}(t),t)$ is positive definite, then state $\mathbf{x}(t)\in\mathbb{R}^n$ of CTZD model (10) converges to the theoretical solution to RTVNO (1) as $t\to+\infty$. The proof is complete. □

Theorem 3 *In addition to Theorem 2, CTZD model* (10) *possesses the following properties.*

1) *If the linear activation function array is exploited, the exponential convergence with rate* γ *in terms of* $\partial f(\mathbf{x}(t),t)/\partial\mathbf{x}(t)\to 0$ *is achieved for CTZD model* (10).
2) *If the power-sigmoid activation function array is exploited, superior convergence is achieved for CTZD model* (10)*, compared with the convergence obtained using the linear activation function array.*
3) *If the hyperbolic sine activation function array is exploited, superior convergence to the exponential convergence with rate* $m\gamma$ $(m=3)$ *is achieved for CTZD model* (10)*, compared with the one obtained using the power-sigmoid activation function array.*

Proof We focus on the convergence properties of CTZD model (10) by using the aforementioned three types of activation function array $\Phi(\cdot)$.

1) If CTZD model (10) is activated by the linear activation function array, then

$$\mathrm{d}\mathbf{g}(\mathbf{x}(t),t)/\mathrm{d}t = -\gamma\Phi_{\mathrm{li}}\left(\mathbf{g}(\mathbf{x}(t),t)\right) = -\gamma\mathbf{g}(\mathbf{x}(t),t) = -\gamma\frac{\partial f(\mathbf{x}(t),t)}{\partial\mathbf{x}(t)}. \quad (20)$$

From (20), we obtain

$$\frac{\partial f(\mathbf{x}(t),t)}{\partial\mathbf{x}(t)} = \mathbf{g}(\mathbf{x}(t),t) = \exp(-\gamma t)\mathbf{g}(\mathbf{x}(0),0) = \exp(-\gamma t)\frac{\partial f(\mathbf{x}(0),0)}{\partial\mathbf{x}(0)}, \quad (21)$$

which proves that CTZD model (10) possesses the exponential convergence with rate γ in terms of $e_j(t) = g_j(\mathbf{x}(t),t) = \partial f(\mathbf{x}(t),t)/\partial x_j(t) \to 0$.

2) Next, the power-sigmoid activation function array is discussed. The following two sub-cases are considered in the application of the power-sigmoid activation function array to CTZD model (10).

2.a) For $|\partial f(\mathbf{x}(t),t)/\partial x_j(t)| \geqslant 1$, the power function $\phi(e_j(t)) = e_j^p(t)$ with $p = 3$ is exploited specifically for this error range. Reviewing equations (18) and (19), we have

$$\dot{V}_{\mathrm{ps}}(\mathbf{x}(t),t) = -\gamma\sum_{j=1}^{n} g_j\phi_{\mathrm{ps}}\left(g_j\right) = -\gamma\sum_{j=1}^{n} g_j^{p+1}$$

$$\begin{cases} = -\gamma\sum_{j=1}^{n} g_j^2 = -\gamma\sum_{j=1}^{n} g_j\phi_{\mathrm{li}}\left(g_j\right) = \dot{V}_{\mathrm{li}}(t), \ \forall|g_j| = 1, \\ < -\gamma\sum_{j=1}^{n} g_j^2 = -\gamma\sum_{j=1}^{n} g_j\phi_{\mathrm{li}}\left(g_j\right) = \dot{V}_{\mathrm{li}}(t), \ \exists|g_j| > 1, \end{cases}$$

where $\dot{V}_{\mathrm{ps}}(\mathbf{x}(t),t)$ and $\dot{V}_{\mathrm{li}}(\mathbf{x}(t),t)$ denote $\dot{V}(\mathbf{x}(t),t)$ activated by the power-sigmoid activation function array and by the linear activation function array, respectively [note that, in this proof, g_j stands for $\partial f(\mathbf{x}(t),t)/\partial x_j(t)$ due to the space limitation]. Therefore, if the power-sigmoid activation function array over error range $|g_j| > 1$ is exploited, superior convergence is achieved for CTZD model (10), compared with the convergence obtained using the linear activation function array, in which the exponential convergence rate is γ.

2.b) For $|\partial f(\mathbf{x}(t),t)/\partial x_j(t)| < 1$, the bipolar-sigmoid function is applied for this error range. Reviewing Lyapunov function candidate $V(\mathbf{x}(t),t) = \|\partial f(\mathbf{x}(t),t)/\partial\mathbf{x}(t)\|_2^2/2$ and equation (19) again over the error range, we prove with $\xi = 4$ that

$$\dot{V}_{\mathrm{ps}}(\mathbf{x}(t),t) = -\gamma\sum_{j=1}^{n} g_j\phi_{\mathrm{ps}}\left(g_j\right) < -\gamma\sum_{j=1}^{n} g_j^2 = -\gamma\sum_{j=1}^{n} g_j\phi_{\mathrm{li}}\left(g_j\right) = \dot{V}_{\mathrm{li}}(t), \forall|g_j| < 1,$$

which implies that, if the power-sigmoid activation function array is applied over error range $|g_j| < 1$, superior convergence is achieved for CTZD model (10), in contrast with the result obtained using the linear activation function array, in which the exponential convergence rate is γ.

The analysis of the two sub-cases shows that the power-sigmoid activation function array is utilized, and hence, superior convergence is achieved

(to the exponential convergence of using the linear activation function array with rate γ).

3) Finally, the hyperbolic sine activation function array with $m = 3$ is discussed. Applying Taylor expansion to the hyperbolic sine activation function, we obtain

$$\begin{aligned}\phi_{\mathrm{hs}}\left(g_j\right) &= \left(\frac{\exp(3g_j)}{2} - \frac{\exp(-3g_j)}{2}\right) = \frac{1}{2}\left(\sum_{l=0}^{+\infty}\frac{(3g_j)^l}{l!} - \sum_{l=0}^{+\infty}\frac{(-3g_j)^l}{l!}\right)\\ &= \sum_{l=1}^{+\infty}\frac{(3g_j)^{2l-1}}{(2l-1)!} > 3g_j. \end{aligned} \tag{22}$$

Reviewing equations (19) and (22), for any $|g_j| \neq 0$, we have

$$\begin{aligned}\dot{V}_{\mathrm{hs}}(t) &= -\gamma\sum_{j=1}^{n} g_j\phi_{\mathrm{hs}}\left(g_j\right) < -\gamma\sum_{j=1}^{n} 3g_j^2 = 3\dot{V}_{\mathrm{li}}(\mathbf{x}(t),t),\\ \dot{V}_{\mathrm{hs}}(t) &= -\gamma\sum_{j=1}^{n} g_j\phi_{\mathrm{hs}}\left(g_j\right) < -\gamma\sum_{j=1}^{n} g_j\phi_{\mathrm{ps}}\left(g_j\right) = \dot{V}_{\mathrm{ps}}(t),\end{aligned}$$

which, together with $V(\mathbf{x}(t),t) = \|\partial f(\mathbf{x}(t),t)/\partial\mathbf{x}(t)\|_2^2/2$, implies that, superior convergence can be achieved for CTZD model (10) if we use the hyperbolic sine activation function array, in contrast with the result obtained by using the linear activation function array and the power-sigmoid activation function array. In addition, the convergence of $\partial f(\mathbf{x}(t),t)/\partial\mathbf{x}(t) \to 0$ using the hyperbolic sine activation function array is faster than the exponential convergence with rate $m\gamma$ (where $m = 3$). As such, the proof is complete. □

Remark 1 As shown in Theorem 3, the residual error of CTZD model (10) for RTVNO (1) with a linear activation function array is $\mathbf{e}(t) = \mathbf{e}(0)\exp(-\gamma t)$, where $\mathbf{e}(0)$ is the initial value of $\mathbf{e}(t)$. Generally, a fast convergence rate is expected, which requires that γ should be set sufficiently large. In addition, after the time period of $4/\gamma$ seconds, $|e_i(t)|$ [being the absolute value of the ith element of $\mathbf{e}(t)$] would be less than 1.85 % of $|e_i(0)|$, $\forall i \in \{1,2,\cdots,n\}$. This means that, with $\gamma = 400$, $|e_i(t)|$ is less than $0.0185 \times |e_i(0)|$ at $t = 0.01$ second, and less than $4.25 \times 10^{-18} \times |e_i(0)|$ at $t = 0.1$ second. Note that floating-point numbers have some limited precision in a computer. For example, the spacing of floating-point numbers, "eps", in MATLAB environment is of order 10^{-16} (i.e., 2^{-52}). Thus, this result is appropriate in practical applications. This means that when the residual error is less than $4.25\times10^{-18}\times|e_i(0)|$ at $t = 0.1$ second, the time-varying optimization of $f(\mathbf{x}(t),t)$ is solved in a pointwise and accurate manner in real time t.

Remark 2 From Theorem 1, we know that the positive definiteness of Hessian matrix $H(\mathbf{x}^*(t),t)$ guarantees that function $f(\mathbf{x}(t),t)$ achieves its minimum at $\mathbf{x}^*(t)$. However, during the real-time solution process, it may be difficult or even impossible to have $\mathbf{x}^*(t)$ and compute Hessian matrix $H(\mathbf{x}^*(t),t)$ to judge its positive definiteness. Considering the fact that residual error $\mathbf{g}(\mathbf{x}(t),t)$ converges exponentially to zero, it

is practicable and reasonable to have $\partial f(\mathbf{x}(t), t)/\partial \mathbf{x}(t) = 0$ [i.e., $\mathbf{x}(t)$ satisfies the assumption of $f_{x_i}(\mathrm{p}^*) = 0$ in Theorem 1] and to compute Hessian matrix $H(\mathbf{x}(t), t)$ as an estimate of $H(\mathbf{x}^*(t), t)$ to judge its positive definiteness.

Theorem 4 *Consider discrete-time RTVNO* (11)*. For the general case of step-size* $h \in (0, 2)$*, the MSSRE of DTZD model* (14)*,* $\lim_{k\to\infty} \|\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})\|_2$*, is* $O(\tau^2)$*, where* $\|\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})\|_2$ *denotes the two-norm of* $\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})$*. In addition, for the special case of* $h = 1$*, the maximum residual error* $\|\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})\|_2$ *of DTZD model* (14) *from the second update is theoretically* $O(\tau^2)$*.*

Proof Corresponding to equation (12), DTZD model (14) utilizes the simple approximation of $\dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k)$ via $(\mathbf{g}(\mathbf{x}(t_k), t_k) - \mathbf{g}(\mathbf{x}(t_k), t_{k-1}))/\tau$ when $k \geqslant 1$. We know that

$$\frac{\mathbf{g}(\mathbf{x}(t_k), t_k) - \mathbf{g}(\mathbf{x}(t_k), t_{k-1})}{\tau} = \dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k) + \mathbf{O}(\tau),$$

where $\mathbf{O}(\tau)$ denotes a vector with each element being $O(\tau)$. Therefore, for DTZD model (14), $\widetilde{\mathbf{x}}(t_k) = \mathbf{x}(t_{k+1}) - \mathbf{x}(t_k)$ can be written as

$$\widetilde{\mathbf{x}}(t_k) = -H^{-1}(\mathbf{x}(t_k), t_k)\left(\left(h\mathbf{g}(\mathbf{x}(t_k), t_k) + \tau(\dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k) + \mathbf{O}(\tau))\right)\right). \tag{23}$$

Using Taylor expansion, we have

$$\begin{aligned}\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1}) &= \mathbf{g}(\mathbf{x}(t_k) + \widetilde{\mathbf{x}}(t_k), t_k + \tau) \\ &= \mathbf{g}(\mathbf{x}(t_k), t_k) + H(\mathbf{x}(t_k), t_k)\widetilde{\mathbf{x}}(t_k) + \dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k)\tau + \mathbf{O}(\tau^2),\end{aligned} \tag{24}$$

where $\mathbf{O}(\tau\widetilde{\mathbf{x}}(t_k))$ and $\mathbf{O}(\|\widetilde{\mathbf{x}}(t_k)\|_2^2)$ are absorbed into $\mathbf{O}(\tau^2)$, as they are on the same order of magnitude [26]. Substituting (23) into (24), we obtain

$$\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1}) = (1 - h)\mathbf{g}(\mathbf{x}(t_k), t_k) + \mathbf{O}(\tau^2). \tag{25}$$

Then, we have

$$\begin{aligned}\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1}) &= (1 - h)\mathbf{g}(\mathbf{x}(t_k), t_k) + \mathbf{O}(\tau^2) \\ &= (1 - h)\left((1 - h)\mathbf{g}(\mathbf{x}(t_{k-1}), t_{k-1}) + \mathbf{O}(\tau^2)\right) + \mathbf{O}(\tau^2) \\ &= (1 - h)^2\mathbf{g}(\mathbf{x}(t_{k-1}), t_{k-1}) + (1 - h)\mathbf{O}(\tau^2) + \mathbf{O}(\tau^2) \\ &\quad\vdots \\ &= (1 - h)^k\mathbf{g}(\mathbf{x}(t_1), t_1) + (1 - h)^{k-1}\mathbf{O}(\tau^2) + \cdots + \mathbf{O}(\tau^2) \\ &= (1 - h)^k\mathbf{g}(\mathbf{x}(t_1), t_1) + \frac{(1 - h)^k - 1}{1 - h - 1}\mathbf{O}(\tau^2) \\ &= (1 - h)^k\mathbf{g}(\mathbf{x}(t_1), t_1) + \frac{1 - (1 - h)^k}{h}\mathbf{O}(\tau^2).\end{aligned} \tag{26}$$

Note that, as mentioned before, we use $\mathbf{x}(t_1) = \mathbf{x}(t_0) - hH^{-1}(\mathbf{x}(t_0), t_0)\mathbf{g}(\mathbf{x}(t_0), t_0)$ for the initialization of DTZD model (14). For equation (26), as $0 < h < 2$, we have $-1 < 1 - h < 1$ and $\lim_{k\to\infty}(1 - h)^k = 0$. Thus, from (26), we have

$$\lim_{k\to\infty} \|\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})\|_2 = \left\|\frac{1}{h}\mathbf{O}(\tau^2)\right\|_2 = O(\tau^2).$$

In addition, it follows from (25) that, if $h = 1$, then residual error $\|\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})\|_2$ of DTZD model (14) from the second update is $O(\tau^2)$. Thus, the proof is complete. □

Theorem 5 *The maximum residual error $\|\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})\|_2$ of NRI model (16) for solving discrete-time RTVNO (11) is $O(\tau)$.*

Proof Reviewing and simplifying (24) with $\widetilde{\mathbf{x}}(t_k) = \mathbf{x}(t_{k+1}) - \mathbf{x}(t_k)$ and NRI model (16), we have

$$\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1}) = \dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k)\tau + \mathbf{O}(\tau^2),$$
$$\|\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})\|_2 = \|\dot{\mathbf{g}}_t(\mathbf{x}(t_k), t_k)\tau + \mathbf{O}(\tau^2)\|_2 = O(\tau),$$

which is also in view of the second-order differentiability of $f(\mathbf{x}(t), t)$, i.e., the continuity and finiteness of $\dot{\mathbf{g}}_t(\mathbf{x}(t), t)$. The proof is complete. □

Various methods have been proposed for the fast and accurately solution of the static optimization problem, and for this reason, the time-derivative information is not utilized during the real-time solution process. Thus, we have the following theorem, which reveals that the MSSRE of any method designed intrinsically to solve the static optimization problem and employed for the online solution of discrete-time RTVNO (11) is $O(\tau)$.

Theorem 6 *Suppose that a traditional method (i.e., a method designed intrinsically to solve the static optimization problem that does not utilize the time-derivative information) converges to the optimal solution to a static optimization problem within computational time interval $[0, \tau)$. If the traditional method is employed for the online solution of discrete-time RTVNO (11), the MSSRE of the traditional method is $O(\tau)$.*

Proof As always, assume that the time derivative of $\mathbf{x}^*(t)$ exists, and $\mathrm{d}x_i^*(t_k)/\mathrm{d}t = \delta_i(t_k)$ at time instant $t = k\tau$, with $x_i^*(t)$ being the ith element of vector $\mathbf{x}^*(t)$ and $\delta_i(t_k)$ being the finite time-derivative value of $x_i^*(t_k)$. Then, $\lim_{\tau\to 0} \widetilde{x}_i^*(t_k)/\tau = \mathrm{d}x_i^*(t_k)/\mathrm{d}t = \delta_i(t_k)$. Thus, $\widetilde{x}_i^*(t_k) \approx \delta_i(t_k)\tau$. That is, $\widetilde{x}_i^*(t_k)$ changes in an $O(\tau)$ pattern. In mathematics, $\widetilde{x}_i^*(t_k) = O(\tau)$ and $\widetilde{\mathbf{x}}^*(t_k) = \mathbf{O}(\tau)$. Note that, within computational time interval $[k\tau, (k+1)\tau)$, the traditional method converges to the optimal solution $\mathbf{x}^*(t_k)$ to the discrete-time RTVNO (11) sampled at time instant $t = k\tau$, i.e., $\mathbf{x}(t_{k+1}) = \mathbf{x}^*(t_k)$. Moreover, at time instant $t = (k+1)\tau$, the difference between the solution generated by the traditional method and the optimal solution is $\mathbf{x}(t_{k+1}) - \mathbf{x}^*(t_{k+1}) = \mathbf{x}^*(t_k) - \mathbf{x}^*(t_{k+1}) = -\widetilde{\mathbf{x}}^*(t_k) = \mathbf{O}(\tau)$. Using Taylor expansion, and considering the second-order differentiability of $f(\mathbf{x}(t), t)$ [i.e., the continuity and finiteness of $H(\mathbf{x}^*(t_{k+1}), t_{k+1})$], we have

$$\begin{aligned}\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1}) &= \mathbf{g}(\mathbf{x}^*(t_{k+1}) + \mathbf{O}(\tau), t_{k+1}) \\ &= \mathbf{g}(\mathbf{x}^*(t_{k+1}), t_{k+1}) + H(\mathbf{x}^*(t_{k+1}), t_{k+1})\mathbf{O}(\tau) + \mathbf{O}(\tau^2) \\ &= \mathbf{g}(\mathbf{x}^*(t_{k+1}), t_{k+1}) + \mathbf{O}(\tau).\end{aligned}$$

Given that $\mathbf{g}(\mathbf{x}^*(t_{k+1}), t_{k+1}) = 0$, we further have

$$\|\mathbf{g}(\mathbf{x}(t_{k+1}), t_{k+1})\|_2 = \|\mathbf{O}(\tau)\|_2 = O(\tau).$$

The proof is complete. □

As such, from Theorem 6, the MSSRE of each method in [7–12, 20] has an $O(\tau)$ pattern for solving online discrete-time RTVNO (11).

5 Simulative and numerical verification

In this section, computer simulations and numerical experiments are provided to verify the efficacy of the proposed CTZD and DTZD models for the RTVNO. These simulations are carried out in MATLAB version 7.6.0 environment using a personal digital computer equipped with a central processing unit of 2.20-GHz Inter(R) Core(TM) 2 Duo E4500, 2.0-GB memory and a Microsoft Windows XP Professional operating system.

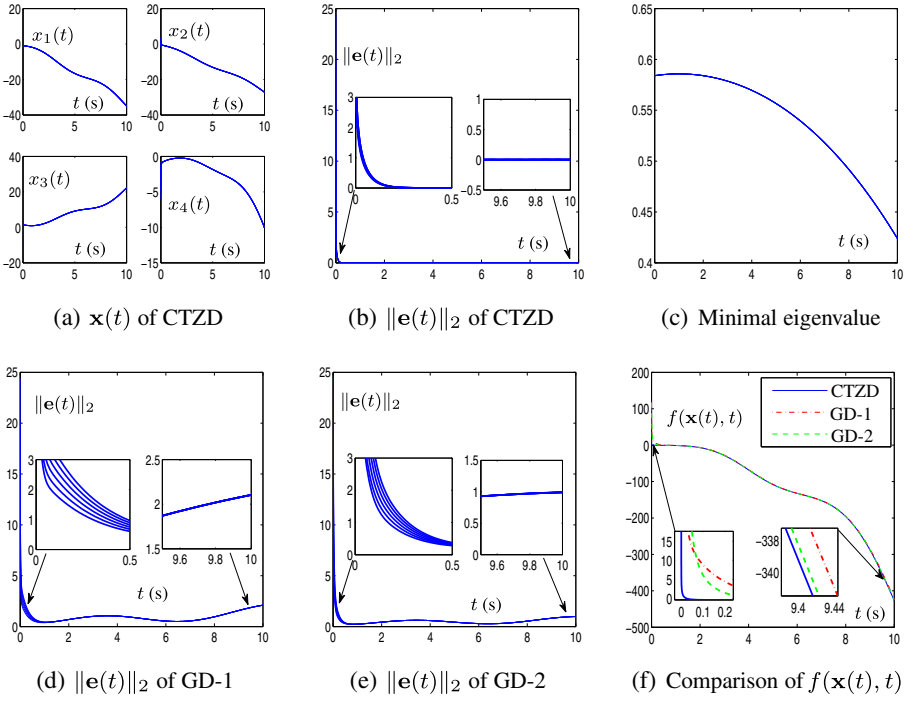
5.1 CTZD, GD and CTD for solving RTVNO

To illustrate and compare the aforementioned exponential convergence of the CTZD model for the RTVNO, in this subsection, one example of the RTVNO formulation is handled by using CTZD model (10), GD models (5) and (6) as well as CTD model (4), which is expressed as

$$\begin{aligned}\min_{\mathbf{x}(t)\in\mathbb{R}^4} f(\mathbf{x}(t), t) = {} & (x_1(t)+t)^2 + (x_2(t)+t)^2 + (x_3(t) - \exp(-t))^2 \\ & +0.1(t-1)x_3(t)x_4(t) - (x_1(t) + \ln(0.1t+1))(x_2(t) + \sin(t)) \\ & +(x_1(t) + \sin(t))x_3(t) + (x_4(t) + \exp(-t))^2. \end{aligned} \tag{27}$$

In addition, all the continuous-time models in this study are simulated using MATLAB routine "ode45" [19].

Figure 1 illustrates the simulation results of the comparison between CTZD model (10) and GD models (5) and (6) in terms of their solutions to RTVNO (27). All states $\mathbf{x}(t)$ start from initial states $\mathbf{x}(0) = [0.5 * j, 4 - j, j - 8, j - 6]^{\mathrm{T}}$, where $j = 0, 1, \cdots, 5$. Specifically, Fig. 1a illustrates the element trajectories of state $\mathbf{x}(t)$ of CTZD model (10) using the hyperbolic sine activation function array over time interval [0 10] s. To investigate the convergence performance of CTZD model (10) and GD models (5) and (6), the residual error $\|\mathbf{e}(t)\|_2 = \|\partial f(\mathbf{x}(t), t)/\partial \mathbf{x}(t)\|_2$ is monitored during the solving process. The residual errors of CTZD model (10) for solving RTVNO (27) are shown in Fig. 1b. As shown in the figure, all the residual errors of CTZD model (10) converge rapidly to zero. In addition, the minimal eigenvalue of $H(\mathbf{x}(t), t)$, as shown in Fig 1c, is larger than zero during the solving process. This means that RTVNO (27) has its time-varying minimum trajectory over time interval [0 10] s. Results of the residual errors of GD models (5) and (6) are shown in Fig. 1d and e, respectively. Starting with different initial states $\mathbf{x}(0)$, all residual errors of CTZD model (10) converge rapidly to zero, while GD models (5)



(a) $\mathbf{x}(t)$ of CTZD (b) $\|\mathbf{e}(t)\|_2$ of CTZD (c) Minimal eigenvalue

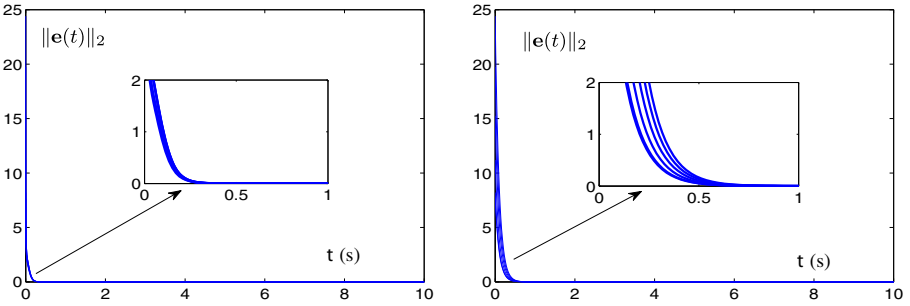
(d) $\|\mathbf{e}(t)\|_2$ of GD-1 (e) $\|\mathbf{e}(t)\|_2$ of GD-2 (f) Comparison of $f(\mathbf{x}(t), t)$

Fig. 1 Comparison of CTZD model (10) and GD models (5) and (6) for solving RTVNO (27) with $\gamma = 10$, where the CTZD model is activated by the hyperbolic sine function array

and (6) display relatively large lagging errors. In addition, Fig. 1f shows the comparison of $f(\mathbf{x}(t), t)$ generated by CTZD model (10) and GD models (5) and (6). As can be seen, $f(\mathbf{x}(t), t)$ generated by CTZD model (10) is smaller than those generated by GD models (5) and (6). An analysis of Fig. 1b, c and f shows that $\mathbf{x}(t)$ of CTZD model (10) achieves the time-varying minimum solution of the objective function depicted in RTVNO (27) in real time t.

Furthermore, as seen from Figs. 1, 2 and 3 as well as in Table 1, in using different activation functions, CTZD model (10) displays different performances on the convergence of the residual error, even with the same value of the design parameter γ. Specifically speaking, the performance of the power-sigmoid activation function array is better than that of the linear activation function array; nevertheless, the hyperbolic sine activation function array has the best performance among the three activation function arrays, which further validates Theorem 3. In addition, as shown in Table 1, the convergence time of CTZD model (10) can be expedited from order 10^{-3} to 10^{-9} s, as design parameter γ is increased from 10^3 to 10^9. The above results substantiate the analysis that CTZD model (10) has an exponential convergence property, which can be expedited by increasing the value of γ.

Next, CTD model (4) and CTZD model (10) are utilized to solve RTVNO (27) for further investigation and comparison. The corresponding simulative and numerical results are shown in Fig. 4. As seen from Fig. 4a, starting with $\mathbf{x}^*(0)$, the residual

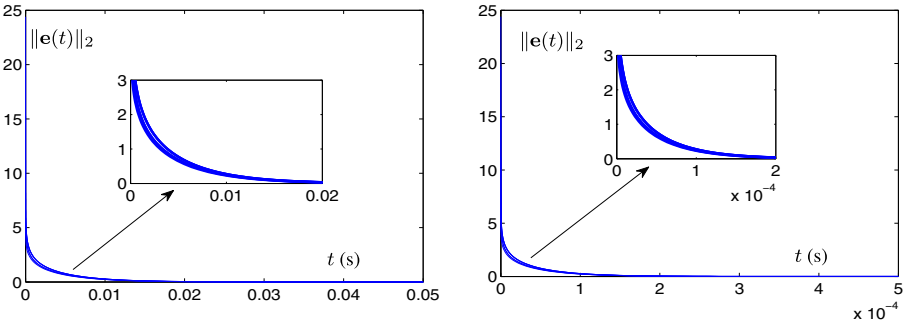


(a) $\|\mathbf{e}(t)\|_2$ by using Φ_{ps} (b) $\|\mathbf{e}(t)\|_2$ by using Φ_{li}

Fig. 2 Residual errors of CTZD model (10) using different activation function arrays and $\gamma = 10$ for solving RTVNO (27)

errors of CTD model (4) and CTZD model (10) are both relatively small. However, as time t increases, the residual error of CTD model (4) becomes slightly larger than that of CTZD model (10). This is evidently attributed to the lack of error feedback in CTD model (4). Thus, the performance of the CTD model (4) may grow worse as computational errors accumulate. Meanwhile, observed from Fig. 4b, the residual error of CTD model (4) does not converge to zero, while that of CTZD model (10) converges rapidly to zero with initial state $\mathbf{x}(0) \neq \mathbf{x}^*(0)$. These comparative results have illustrated once again the efficacy of the proposed CTZD model for time-varying nonlinear optimization.

Remark 3 Zhang dynamics, where the state dimension can be one or more, has been derived from the Zhang neural network (which originated from the research of Hopfield neural network). Zhang dynamics is viewed as a systematic approach to the online solution of time-varying problems, including the scalar situation; it differs from the conventional GD in terms of the problem addressed, error function,



(a) $\|\mathbf{e}(t)\|_2$ of CTZD with $\gamma = 100$ (b) $\|\mathbf{e}(t)\|_2$ of CTZD with $\gamma = 10^4$

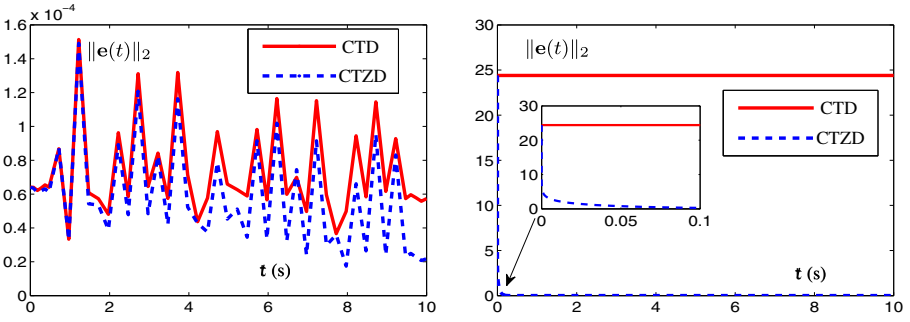
Fig. 3 Residual errors of CTZD model (10) using hyperbolic sine activation functions with different values of γ for solving RTVNO (27)

Table 1 Average convergence times (in seconds) of CTZD model (10) solving RTVNO (27) using different activation function arrays with varying values of γ to achieve precision $\|\mathbf{e}(t)\|_2 < 0.01$

	$\gamma = 10^3$	$\gamma = 10^5$	$\gamma = 10^7$	$\gamma = 10^9$
Linear	0.0078	7.8172×10^{-5}	7.8915×10^{-7}	7.9920×10^{-9}
Power-sigmoid	0.0035	3.4515×10^{-5}	3.4410×10^{-7}	3.4395×10^{-9}
Hyperbolic sine	0.0027	2.7475×10^{-5}	2.7505×10^{-7}	2.7530×10^{-9}

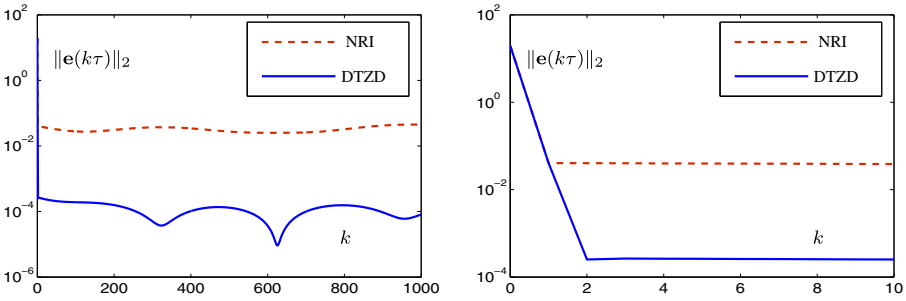
design formula, dynamic equation and the utilization of time derivatives. Thus, A comparison of CTZD model (10) and GD models (5) and (6) is provided below.

1) The design of CTZD model (10) is based on the elimination of each element of a vector-valued indefinite error function. In contrast, the design of GD models (5) and (6) is based on the elimination of the whole norm-based or square-based scalar-valued nonnegative or, at least, lower-bounded energy function [25, 27].
2) The design method of GD models (5) and (6) are intrinsically made for static nonlinear optimization with constant coefficients. Thus, it can only approximately approach the theoretical solution of a time-varying problem. In contrast, the design method of CTZD model (10) is a new method intrinsically made for time-varying problems [23]. Thus, it converges to the exact theoretical solution of RTVNO.
3) GD models (5) and (6) do not utilize the time-derivative information of RTVNO (1). Thus, they may not be effective enough in solving an RTVNO problem. In contrast, CTZD model (10) methodically and systematically exploits the time-derivative information of the objective function depicted in (1) during its real-time solving process. This is why CTZD model (10) converges to the theoretical solution of an RTVNO [28].
4) CTZD model (10) exponentially converges to the time-varying theoretical solution. In contrast, GD models (5) and (6) generate only approximate results for the time-varying theoretical solution with larger lagging errors [29].



(a) With $\mathbf{x}(0) = \mathbf{x}^*(0)$ (b) With $\mathbf{x}(0) \neq \mathbf{x}^*(0)$

Fig. 4 Residual errors of CTD model (4) and CTZD model (10) (with $\gamma = 10$ and linear activation functions) for solving RTVNO (27), where each initial state in subfigure **b** is $\mathbf{x}(0) = [0, 4, -8, -6]^{\mathrm{T}}$



(a) Results of 1000 updates (b) First 10 updates of subfigure (a)

Fig. 5 Residual errors of DTZD model (14) (with $h = 1$) and NRI model (16) for solving RTVNO (27)

5) The investigation of different activation functions (e.g., the power-sigmoid function and the hyperbolic sine function) leads to more choices for better performance. In addition, the convergence speed of CTZD model (10) using nonlinear activation functions is faster than that obtained when using linear activation functions. In contrast, GD models (5) and (6) lack such activation functions. Thus, the application of activation function can be seen as an advantage. Owing to the design parameter γ (being a multiplier of the exponential convergence rate), another effective factor can be expedited for the convergence of the CTZD model (10).

5.2 DTZD, NRI and DTD for solving RTVNO

In this subsection, three time-varying optimization problems are solved as three examples to illustrate the efficacy and advantages of DTZD model (14) for the RTVNO solving. The results are then compared with those of NRI model (16) and DTD model (17).

5.2.1 Example 1

In this example, RTVNO (27) is considered, and its residual errors synthesized by DTZD and NRI models are shown in Fig. 5. As seen from Fig. 5a, starting with the initial state $[0.5, 3, -7, -5]^{\mathrm{T}}$, the MSSRE synthesized by DTZD model (14) is of order 10^{-4}. In addition, the MSSRE synthesized by NRI model (16) is of order 10^{-2}, which is roughly 100 times larger than that of DTZD model (14). Figure 5b further shows that DTZD model (14) only needs two updates to converge to order 10^{-4}. These results substantiate the important role of time-derivative information in obtaining a more precise solution.

Table 2 shows more detailed MSSRE data of the two models for solving RTVNO (27) with respect to different values of sampling gap τ and step-size h. The following important facts are summarized from the table. First, the MSSREs of DTZD model (14) roughly change in an $O(\tau^2)$ pattern. Second, the MSSREs of NRI model (16)

Table 2 MSSREs of DTZD model (14) and NRI model (16) for solving RTVNO (27)

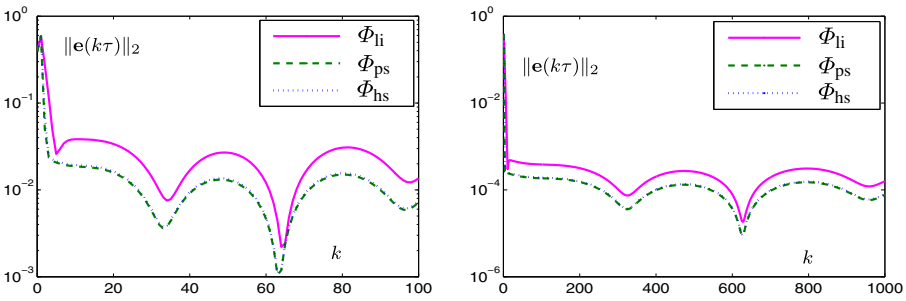
	Parameters				
		Sampling gap (s)			
Model	Step-size	$\tau = 0.1$	$\tau = 0.01$	$\tau = 0.001$	$\tau = 0.0001$
DTZD model (14)	$h = 0.50$	0.0421	3.8840×10^{-4}	3.8834×10^{-6}	3.8833×10^{-8}
	$h = 0.75$	0.0279	2.5901×10^{-4}	2.5815×10^{-6}	2.5814×10^{-8}
	$h = 1.00$	0.0236	1.9420×10^{-4}	1.9419×10^{-6}	1.9418×10^{-8}
	$h = 1.25$	0.0185	1.5533×10^{-4}	1.5481×10^{-6}	1.5481×10^{-8}
	$h = 1.50$	0.0157	1.2955×10^{-4}	1.3038×10^{-6}	1.3038×10^{-8}
	$h = 1.75$	0.0155	1.1594×10^{-4}	1.1583×10^{-6}	1.1582×10^{-8}
	$h = 2$	1.1765	1.1023	2.8313	5.7615
NRI model (16)		0.4485	0.0450	0.0045	4.4871×10^{-4}

change in an $O(\tau)$ pattern. Third, step-size h affects the convergence performance of DTZD model (14).

In addition, the corresponding numerical results in terms of computational time [including the MSSRE and the average computing time per update (ACTPU)] with respect to different values of sampling gap τ are listed in Table 3. As seen from the table, the MSSREs synthesized by DTZD model (14) are much smaller than those by NRI model (16), whereas each computational time of those models is of the same order. As such, the proposed DTZD model is an effective method for improving the accuracy of the solution of software computation and/or digital hardware implementation for RTVNO.

Table 3 Comparison between DTZD model (14) (with $h = 1$) and NRI model (16) in terms of MSSRE and ACTPU for solving RTVNO (27)

Sampling gap τ (s)	Model	MSSRE	ACTPU (s)
$\tau = 0.1$	DTZD model (14)	0.0236	1.9095×10^{-3}
	NRI model (16)	0.4485	1.9039×10^{-3}
$\tau = 0.01$	DTZD model (14)	1.9420×10^{-4}	1.8346×10^{-3}
	NRI model (16)	0.0450	1.8048×10^{-3}
$\tau = 0.007$	DTZD model (14)	9.6322×10^{-5}	1.8534×10^{-3}
	NRI model (16)	0.0315	1.8436×10^{-3}
$\tau = 0.005$	DTZD model (14)	4.9929×10^{-5}	1.8370×10^{-3}
	NRI model (16)	0.0225	1.8325×10^{-3}
$\tau = 0.002$	DTZD model (14)	9.0214×10^{-6}	1.8162×10^{-3}
	NRI model (16)	0.0090	1.7886×10^{-3}

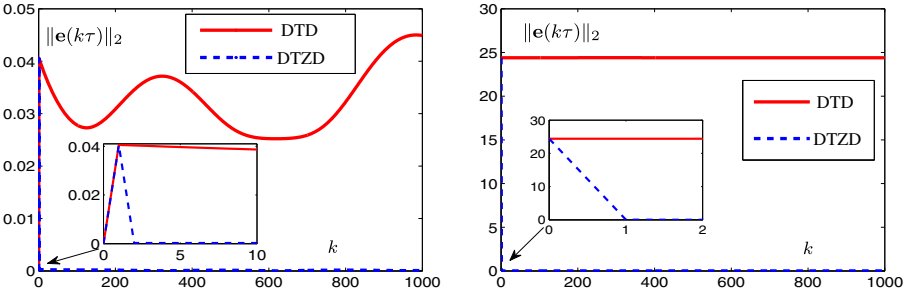


(a) With $\tau = 0.1$ s (b) With $\tau = 0.01$ s

Fig. 6 Residual errors of DTZD model (15) (with $h = 0.5$) and with different activation functions for solving RTVNO (27)

The power-sigmoid and the hyperbolic sine activation function arrays are developed for solving RTVNO (27). Figure 6 consists of two subfigures: the left subfigure presents the residual errors of DTZD model (15) with $\tau = 0.1$ s, whereas the right one presents those of DTZD model (15) with $\tau = 0.01$ s. The residual errors of DTZD model (15) with the power-sigmoid and the hyperbolic sine activation function arrays are slightly less than that obtained with the linear activation function array. This finding illustrates that the power-sigmoid and the hyperbolic sine activation function arrays can achieve relatively better performance for the DTZD model.

Moreover, DTZD model (14) and DTD model (17) are utilized to solve RTVNO (27), and the corresponding numerical results are shown in Fig. 7. As seen from Fig. 7a, starting with $\mathbf{x}^*(0)$, the residual error of DTZD model (14) converges to near zero with two updates and remains relatively small, whereas the residual error of DTD model (17) is much larger and of order 10^{-2}. As update index k increases, the residual error of DTD model (17) increases due to the lack of error feedback. In addition, as seen from Fig. 7b, the residual error of DTD model (17) with initial state $\mathbf{x}(0) \neq \mathbf{x}^*(0)$ does not converge to zero, whereas that of DTZD model (14) converges to



(a) With $\mathbf{x}(0) = \mathbf{x}^*(0)$ (b) With $\mathbf{x}(0) \neq \mathbf{x}^*(0)$

Fig. 7 Residual errors of DTZD model (14) (with $h = 1$) and DTD model (17) for solving RTVNO (27), where $\tau = 0.01$ and each initial state in subfigure **b** is $\mathbf{x}(0) = [0, 4, -8, -6]^{\mathrm{T}}$

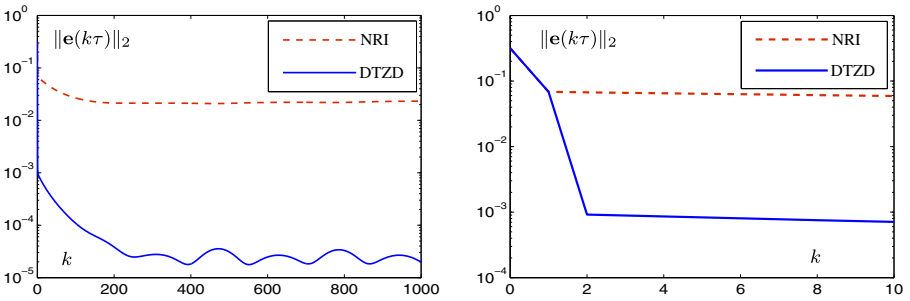
zero (or say rigorously, near zero) with only one or two updates. Therefore, the DTD model (17) is highly sensitive to the initial value (as well as the perturbations), thus limiting its usage in practical applications. These comparative results have illustrated once again the efficacy of the proposed DTZD model for time-varying nonlinear optimization.

5.2.2 Example 2

In this example, we consider the RTVNO given by

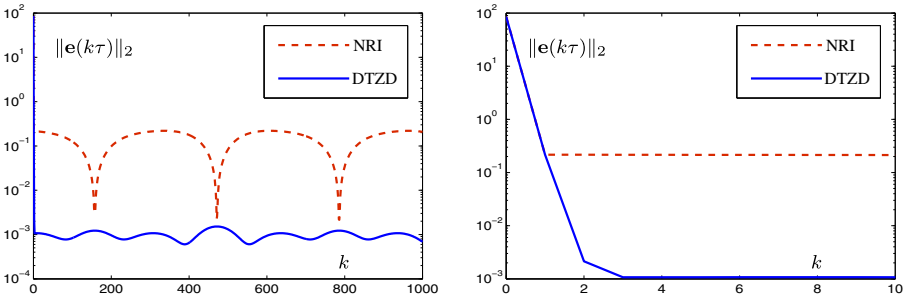
$$\begin{aligned}\min_{\mathbf{x}(t)\in\mathbb{R}^{10}} f(\mathbf{x}(t),t) \;=\; & (4+\sin(t))x_1^2(t)+2(x_2(t)-\exp(-t))^2+2x_3^2(t)+x_4^2(t)\\ & -\cos(t)x_3(t)x_4(t)+2x_3(t)\exp(-t)x_2(t)+(1+\cos(t))x_5^2(t)\\ & +(x_9(t)-\exp(-t))^2+x_8^2(t)+\log(0.1t+1)x_9(t)x_{10}(t)\\ & +2x_6^2(t)+2x_7^2(t)+x_1(t)x_7(t)+(x_{10}(t)-t)^2\\ & -\sin(t)x_7(t)x_8(t)+2x_7(t)-\exp(-t)x_6(t)x_5(t). \qquad (28)\end{aligned}$$

In this and in the subsequent examples, the corresponding solution $\mathbf{x}(t)$, the minimal eigenvalue of the Hessian matrix $H(\mathbf{x}(t),t)$ and the generated $f(\mathbf{x}(t),t)$ are omitted due to the complexity of the RTVNO and the space limitation. Therefore, we only present the residual errors synthesized by DTZD model (14) and NRI model (16), with the initial state being $[0.1, 1.4, -0.8, -0.4, 0, 0, -0.5, 0, 1, 0]^{\mathrm{T}}$. The numerical experimental results are shown in Fig. 8. As seen from the figure, the MSSRE synthesized by DTZD model (14) is of order 10^{-5}, whereas that synthesized by NRI model (16) is of order 10^{-2}, which is around 1000 times larger than that of DTZD model (14). In addition, as observed from Fig. 8b, the residual error of DTZD model (14) achieves the order of 10^{-4} (specifically, 8.7×10^{-4}) after two updates. Note that the residual error of DTZD model (14) from update index $k=200$ to $k=1000$ is finally of order 10^{-5}.



(a) Results of 1000 updates (b) First 10 updates of subfigure (a)

Fig. 8 Residual errors of DTZD model (14) (with $h=1$) and NRI model (16) for solving RTVNO (28)



(a) Results of 1000 updates (b) First 10 updates of subfigure (a)

Fig. 9 Residual errors of DTZD model (14) (with $h = 1$) and NRI model (16) for solving RTVNO (29) with dimension $n = 30$

5.2.3 Example 3

In this example, a more complicated problem, which originates from equation (22) in [4], is considered

$$\min_{\mathbf{x}(t)\in\mathbb{R}^n} f(\mathbf{x}(t), t) = \sum_{i=1}^{\frac{n}{2}} 4x_{2i-1}^2 + 2\sin(t)x_{2i-1}x_{2i} + 2x_{2i}^2 - 22x_{2i-1} - 2x_{2i}. \quad (29)$$

The corresponding numerical experimental results, synthesized by DTZD model (14) and NRI model (16) starting with $[0, \cdots, 0]^{\mathrm{T}} \in \mathbb{R}^n$, are shown in Fig. 9 and Table 4. Results shown in Fig. 9a and b are similar to those shown in Figs. 5 and 8, which

Table 4 Comparison between DTZD model (14) (with $h = 1$) and NRI model (16) in terms of MSSRE and ACTPU for solving RTVNO (29)

Sampling gap τ (s)	Dimension n	Model	MSSRE	ACTPU (s)
$\tau = 0.1$	10	DTZD model (14)	0.0878	2.1815×10^{-3}
		NRI model (16)	1.2834	2.1762×10^{-3}
$\tau = 0.01$	10	DTZD model (14)	8.7716×10^{-4}	2.1045×10^{-3}
		NRI model (16)	0.1270	2.0428×10^{-3}
$\tau = 0.1$	20	DTZD model (14)	0.1241	5.0671×10^{-3}
		NRI model (16)	1.8149	4.9381×10^{-3}
$\tau = 0.01$	20	DTZD model (14)	1.200×10^{-3}	4.9745×10^{-3}
		NRI model (16)	0.1779	4.8377×10^{-3}
$\tau = 0.1$	30	DTZD model (14)	0.1521	9.0315×10^{-3}
		NRI model (16)	2.1740	8.9771×10^{-3}
$\tau = 0.01$	30	DTZD model (14)	1.5142×10^{-3}	8.9817×10^{-3}
		NRI model (16)	0.2195	8.6771×10^{-3}

substantiate the superiority of the proposed DTZD model (14) again. Meanwhile, Table 4 lists the corresponding MSSRE and ACTPU data, with respect to different values of sampling gap τ and dimension n. As seen from the table, the MSSREs synthesized by DTZD model (14) are much smaller than those by NRI model (16) even for the relatively larger scale application, whereas their computational times are of the same order. Several important facts are summarized from the table. First, the MSSRE of DTZD model (14) changes in an $O(\tau^2)$ pattern. Second, the MSSRE of NRI model (16) changes in an $O(\tau)$ pattern. Third, as dimension n increases, the ACTPU increases. Restricted by the current computer operating environment, the ACTPU of DTZD model (14) for solving RTVNO (29) with dimension $n = 30$ is close to the sampling gap $\tau = 0.01$ s.

6 Conclusions

In this study, the CTD model and two GD models (i.e., GD-1 and GD-2) have been presented and employed for the RTVNO solving. By following Zhang et al's design method, the CTZD model has been proposed, generalized and investigated to combine the advantages of CTD and GD models while remedying their weaknesses. Moreover, the DTZD model has been developed for potential digital hardware realization. Besides, a bridge has been built to connect the NRI model and DTZD model. Theoretical analyses have shown that the residual error of the CTZD model has an exponential convergence, and that the MSSRE of the DTZD model has an $O(\tau^2)$ pattern with τ denoting the sampling gap. The MSSRE of any method designed intrinsically for solving the static optimization problem and employed for the online discrete-time RTVNO solving is shown to have an $O(\tau)$ pattern. Moreover, simulative and numerical results further illustrated the efficacy and advantages of the proposed CTZD and DTZD models for RTVNO solving. Considering that the Hessian matrix $H(\mathbf{x}(t), t)$ involved in the proposed CTZD and DTZD models is required to be invertible in the online solution process of the RTVNO problem, the investigation of new models for RTVNO with a singular Hessian matrix may be an interesting but difficult research topic in the future.

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

References

1. Courvoisier, Y., Gander, M.J.: Optimization of Schwarz waveform relaxation over short time windows. Numer. Algor **64**(2), 221–243 (2013)
2. Amini, K., Ahookhosh, M., Nosratipour, H.: An inexact line search approach using modified nonmonotone strategy for unconstrained optimization. Numer. Algor **66**(1), 49–78 (2014)
3. Andrei, N.: An accelerated subspace minimization three-term conjugate gradient algorithm for unconstrained optimization. Numer. Algor **65**(4), 859–874 (2014)

4. Martínez, J.M., Prudente, L.F.: Handling infeasibility in a large-scale nonlinear optimization algorithm. Numer. Algor **60**(2), 263–277 (2012)
5. Zhang, Y., Ge, S.S., Lee, T.H.: A unified quadratic programming based dynamical system approach to joint torque optimization of physically constrained redundant manipulators. IEEE Trans. Syst., Man, Cybern. B, Cybern **34**(5), 2126–2132 (2004)
6. Ahookhosh, M., Amini, K.: An efficient nonmonotone trust-region method for unconstrained optimization. Numer. Algor **59**(4), 523–540 (2012)
7. Wang, F., Jian, J., Wang, C.: A model-hybrid approach for unconstrained optimization problems. Numer. Algor **66**(4), 741–759 (2014)
8. Gaviano, M., Lera, D.: Properties and numerical testing of a parallel global optimization algorithm. Numer. Algor **60**(4), 613–629 (2012)
9. Cai, X., Wang, G., Zhang, Z.: Complexity analysis and numerical implementation of primal-dual interior-point methods for convex quadratic optimization based on a finite barrier. Numer. Algor **62**(2), 289–306 (2013)
10. Birgin, E.G., Martínez, J.M.: A spectral conjugate gradient method for unconstrained optimization. Appl. Math. Opt **43**(2), 117–128 (2001)
11. Dai, Y., Liao, L.: New conjugacy conditions and related nonlinear conjugate gradient methods. Appl. Math. Opt **43**(1), 87–101 (2001)
12. Narushima, Y., Yabe, H.: Conjugate gradient methods based on secant conditions that generate descent search directions for unconstrained optimization. J. Appl. Math. Comput. **236**(17), 4303–4317 (2012)
13. Xia, Y., Sun, C.: A novel neural dynamical approach to convex quadratic program and its efficient applications. Neural Netw. **22**(1), 1463–1470 (2009)
14. Liao, B., Zhang, Y.: Different complex ZFs leading to different complex ZNN models for time-varying complex generalized inverse matrices. IEEE Trans. Neural Netw. Learn. Syst. **25**(9), 1621–1631 (2014)
15. Xiao, L., Lu, R.: Finite-time solution to nonlinear equation using recurrent neural dynamics with a specially-constructed activation function. Neurocomputing **151**(1), 246–251 (2015)
16. Barbarosou, M.P., Maratos, N.G.: A nonfeasible gradient projection recurrent neural network for equality-constrained optimization problems. IEEE Trans. Neur. Netw. **19**(10), 1665–1677 (2008)
17. Cao, J., Mao, X., Luo, Q.: Neurodynamic system theory and applications. Abstr. Appl. Anal. **2013**(1), 1–1 (2013)
18. Liao, L., Qi, H., Qi, L.: Neurodynamical optimization. J. Global Optim. **28**(2), 175–195 (2004)
19. Mathews, J.H., Fink, K.D.: Numerical Methods Using MATLAB. Prentice-Hall, Englewood Cliffs, NJ (2005)
20. Bhaya, A., Kaszkurewicz, E.: Control Perspectives on Numerical Algorithms and Matrix Problems. SIAM, Philadelphia, PA (2006)
21. Bhaya, A., Kaszkurewicz, E.: A control-theoretic approach to the design of zero finding numerical methods. IEEE Trans. Autom. Control **52**(6), 1014–1026 (2007)
22. Bhaya, A., Kaszkurewicz, E.: Iterative methods as dynamical systems with feedback control. In: Proceedings of IEEE Conference on Decision and Control, pp. 2374–2380 (2003)
23. Zhang, Y., Yi, C.: Zhang Neural Networks and Neural-Dynamic Method. Nova, New York (2011)
24. Zhang, Y., Li, Z., Guo, D., Ke, Z., Chen, P.: Discrete-time ZD, GD and NI for solving nonlinear time-varying equations. Numer. Algor. **64**(4), 721–740 (2013)
25. Zhang, Y., Xiao, L., Ruan, G., Li, Z.: Continuous and discrete time Zhang dynamics for time-varying 4th root finding. Numer. Algor. **57**(1), 35–51 (2011)
26. Piepmeier, J.A., McMurray, G.V., Lipkin, H.: A dynamic quasi-Newton method for uncalibrated visual servoing. In: Proceedings of IEEE Conference on Robotics and Automation, pp. 1595–1600 (1999)
27. Zhang, Y., Ge, S.S.: Design and analysis of a general recurrent neural network model for time-varying matrix inversion. IEEE Trans. Neur. Netw. **16**(6), 1477–1490 (2005)
28. Xiao, L., Zhang, Y.: Zhang neural network versus gradient neural network for solving time-varying linear inequalities. IEEE Trans. Neur. Netw. **22**(10), 1676–1684 (2011)
29. Zhang, Y., Chen, K., Tan, H.: Performance analysis of gradient neural network exploited for online time-varying matrix inversion. IEEE Trans. Autom. Control **54**(8), 1940–1945 (2009)