A strictly predefined-time convergent and anti-noise fractional-order zeroing neural network for solving time-variant quadratic programming in kinematic robot control

Yi Yang[a,b], Xiao Li[c], Xuchen Wang[a,b], Mei Liu[a,b], Junwei Yin[d], Weibing Li[e], Richard M. Voyles[f] and Xin Ma[a,b,*]

[a] *Multi-Scale Medical Robotics Center, The Chinese University of Hong Kong, Hong Kong 999077, China*
[b] *Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong 999077, China*
[c] *Department of Mechanical and Electrical Engineering, China University of Petroleum (East China), Qingdao 266580, China*
[d] *School of Mechanical Engineering, Dalian Jiaotong University, Dalian 116028, China*
[e] *School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China*
[f] *School of Engineering Technology, Purdue University, West Lafayette, IN 47907, USA*

ARTICLE INFO

Keywords:
strictly predefined-time convergent
fractional-order zeroing neural network
time-variant quadratic programming
kinematic control
noise resilience

ABSTRACT

This paper proposes a strictly predefined-time convergent and anti-noise fractional-order zeroing neural network (SPTC-AN-FOZNN) model, meticulously designed for addressing time-variant quadratic programming (TVQP) problems. This model marks the first variable-gain ZNN to collectively manifest strictly predefined-time convergence and noise resilience, specifically tailored for kinematic motion control of robots. The SPTC-AN-FOZNN advances traditional ZNNs by incorporating a conformable fractional derivative in accordance with the Leibniz rule, a compliance not commonly achieved by other fractional derivative definitions. It also features a novel activation function designed to ensure favorable convergence independent of the model's order. When compared to five recently published recurrent neural networks (RNNs), the SPTC-AN-FOZNN, configured with $0 < \alpha \leq 1$, exhibits superior positional accuracy and robustness against additive noises for TVQP applications. Extensive empirical evaluations, including simulations with two types of robotic manipulators and experiments with a Flexiv Rizon robot, have validated the SPTC-AN-FOZNN's effectiveness in precise tracking and computational efficiency, establishing its utility for robust kinematic control.

1. Introduction

Quadratic programming (QP) problems are central to diverse scientific and engineering domains, particularly artificial intelligence and robotic kinematic control (Boyd & Vandenberghe, 2004; Jin et al., 2018; Zhang & Jin, 2017). These challenges are often formulated as time-variant quadratic programming (TVQP) problems (Boyd & Vandenberghe, 2004; Chong & Żak, 2013). The ubiquity of QPs has catalyzed the development of advanced algorithms for their resolution (Mandal, 2023). Recurrent neural networks (RNNs), known for their time-series processing capability, have become prominent solvers for these problems due to their ability to operate in parallel hardware realizations (Huang et al., 2016; Li, 2018; Lu et al., 2019). However, traditional Gradient Neural Networks (GNNs), a subset of RNNs, are typically aligned with time-invariant problems and have shown limitations like lagging-behind errors when applied to TVQP challenges (Lu et al., 2019; Zhang & Yi, 2011).

In response to these deficiencies, Zeroing Neural Network (ZNN) models have been developed, providing a reliable framework for solving TVQP problems (Jin et al., 2017; Liu et al., 2023). ZNN models have exhibited exponential convergence in kinematic control of robots, with the choice of activation functions significantly affecting their convergence performance (Li et al., 2021; Qi et al., 2022). This has led to the establishment of finite-time, fixed-time, and predefined-time convergent ZNNs (Becerra et al., 2018; Chen et al., 2020; Li et al., 2013; Li, 2018; Li et al., 2019, 2021, 2020; Xiao et al., 2023). Moreover, variable gains have been identified as a factor influencing the convergence rate and precision of ZNNs (Li & Zhang, 2011). Nevertheless, the use of large constant gains or progressively increasing variable gains poses significant challenges in hardware implementations, primarily due to limited power budgets and other practical constraints (Zhang & Yi, 2011). Additionally, the susceptibility of ZNNs to noises (Li et al., 2024, 2019), such as measurement and computational round-off inaccuracies in hardware realizations, further complicates their deployment in real-world settings, thus highlighting a crucial gap in their practicality and robustness under variable-gain conditions.

Building upon the distinctive characteristics of the PTC-FOZNN model (Yang et al., 2024a), which shows enhanced convergence traits under a time-shrinking gain with a suitably designed activation function, this paper proposes, for the first time, the strictly predefined-time convergent and anti-noise fractional-order zeroing neural network (SPTC-AN-FOZNN) model to address TVQP problems. The SPTC-AN-FOZNN innovatively incorporates a novel activation function that ensures noise resilience and strictly predefined-time convergence, independent of the model's order. Numerical validations highlight the SPTC-AN-FOZNN's superior

[*] Corresponding author
Email address: xinma001@cuhk.edu.hk (X. Ma)

convergence performance compared to five recently established RNN models. Furthermore, its utility as an inverse-kinematics solver for robotic motion planning tasks has been strictly confirmed. The paper's contributions include:

(1) This research signifies a measured advancement in the development of strictly predefined-time convergent and noise-resilient activation functions for a class of hardware-friendly variable-gain ZNNs, known in our framework as FOZNNs. This results in a novel SPTC-AN-FOZNN model for successful tackling of TVQP problems.

(2) A rigorous proof of the SPTC-AN-FOZNN's strictly predefined-time convergent and noise-tolerant characteristics, demonstrating its practical applicability in kinematic control of robotic manipulators.

(3) Superior convergence precision and robustness against additive noises of the SPTC-AN-FOZNN when compared to five other RNN models, validated through an illustrative numerical example and complementary empirical evaluations, including simulations and experiments with a Franka Emika Panda robot and a Flexiv Rizon robot.

The paper is organized as follows: Section 2 discusses TVQP problems and relevant preliminary theory in the formulation of traditional RNNs. Section 3 details the SPTC-AN-FOZNN and its convergence analysis. Section 4 presents numerical validations, showcasing the model's efficacy for TVQPs. Section 5 applies the SPTC-AN-FOZNN to robotic kinematic control, illustrating its practical application through simulations and experiments. Section 6 concludes the paper.

2. Preliminary theory

2.1. *Problem formulation and preliminaries*

A continuous-time RNN model can be mathematically represented as follows (Haykin, 1999; Yi & Tan, 2004):

$$\dot{x} = \mathcal{R}(x(t), t), t \in [0, \infty) \tag{1}$$

where $x(t) \in \mathbb{R}^n$ denotes the state vector of the system, and $x_0 = x(0)$ specifies the system's initial state. The function $\mathcal{R}(\cdot)$ represents a proper dynamic functional. Assume that the origin, $x(t) = 0$ serves as the equilibrium state of the system, the following definitions relevant to the convergence theory can be outlined,

Definition 1. *(Bhat & Bernstein, 2000) The origin of the system (1) is defined as locally finite-time stable if a nonempty open set Ω around the origin, along with a locally bounded settling-time function $T : \Omega\backslash\{0\} \rightarrow \mathbb{R}_+ \cup \{0\}$ exists to ensure that any trajectory $x(t, x_0)$ originating from an initial state $x_0 \in \Omega\backslash\{0\}$ converges to the origin for all $t \geq T(x_0)$.*

Definition 2. *(Polyakov, 2012) The origin of the system (1) is defined as locally fixed-time stable when it is not only locally finite-time stable but also satisfies the condition where a constant $T_{\max} > 0$ exists such that $T(x_0) \leq T_{\max}$ for all $x_0 \in \Omega$. Furthermore, the origin of system 1 is said to be globally fixed-time stable if $\Omega = \mathbb{R}^n$.*

Definition 3. *(Sanchez-Torres et al., 2015) The origin of the system (1) with a predefined constant $t_c > 0$ is defined as locally predefined-time stable when it is not only locally fixed-time stable but also satisfies the condition where $T(x_0) \leq t_c$ for all $x_0 \in \Omega$. Furthermore, the origin of system (1) is said to be globally predefined-time stable if $\Omega = \mathbb{R}^n$.*

Definition 4. *(Becerra et al., 2018) The origin of the system (1) is termed as weakly predefined-time stable if it is predefined-time stable with $T(x_0) \leq t_c$ for all $x_0 \in \Omega$.*

Definition 5. *(Becerra et al., 2018) The origin of the system (1) is termed as strictly predefined-time stable if it is predefined-time stable with $\sup_{x_0 \in \Omega} T(x_0) = t_c$.*

Remark 1. According to the above definitions, it is known that an RNN model that achieves predefined-time convergence (PTC) also satisfies fixed-time convergence (FIXTC) and finite-time convergence (FNITC) conditions. The strictly predefined-time convergence (SPTC) characteristic of the RNN model guarantees consistent performance and dependability. In constrast to the weakly PTC (WPTC), the SPTC guarantees that the model converges exactly at a user-prescribed time t_c.

The integration of noise resistance with predefined-time convergence in the RNN model enhances its robustness against disturbances and sensor noise, critical in dynamic settings. This robustness ensures that the RNN consistently meets strict timing constraints and maintains reliable performance, even under variable conditions. Such capabilities are essential in precision-dependent applications like robotic surgery and autonomous vehicle navigation, where safety and effectiveness hinge on accurate timing and stable operation. Together, these features significantly broaden the practical utility and reliability of RNN models.

Given the reasons in Remark 1, this work explores a strictly predefined-time convergent and anti-noise recurrent neural solution to the following TVQP problem:

$$\begin{array}{ll} \min & x^{\mathrm{T}}(t)H(t)x(t)/2 + \rho^{\mathrm{T}}(t)x(t) \\ \text{s.t.} & A(t)x(t) = b(t) \\ & C(t)x(t) \leq d(t) \end{array} \tag{2}$$

where $H(t) \in \mathbb{R}^{n\times n}$ is positive semi-definite, $A(t) \in \mathbb{R}^{m\times n}$ and $C(t) \in \mathbb{R}^{p\times n}$ are matrices of full row rank, and $\rho(t) \in \mathbb{R}^n$, $b(t) \in \mathbb{R}^m$ and $d(t) \in \mathbb{R}^p$ are vectors of proper dimensions. If the solution to (2) exists, it is termed as the Karush-Kuhn-Tucker (KKT) point for the TVQP problem (2).

Lemma 1. *(Nazemi & Nazemi, 2014) $x^*(t) \in \mathbb{R}^n$ is the KKT point for the TVQP problem (2) if for every $\tau \rightarrow 0_+$ there exist the Lagrangian multipliers $\phi^*(t) \in \mathbb{R}^m$ and $\varphi^*(t) \in \mathbb{R}^p$ satisfying*

$$\begin{cases} H(t)x^*(t) + \rho(t) + A^{\mathrm{T}}(t)\phi^*(t) + C^{\mathrm{T}}(t)\varphi^*(t) = 0 \\ A(t)x^*(t) - b(t) = 0 \\ \psi^{\tau}_{FB}(d(t) - C(t)x^*(t), \varphi^*(t)) = 0 \end{cases} \tag{3}$$

where ψ^{τ}_{FB} represents the perturbed Fischer-Burmeister (FB) function.

The perturbed FB function is defined as (Nazemi & Nazemi, 2014):

$$\psi^{\tau}_{FB}(u,v) = u + v - \sqrt{u \odot u + v \odot v + \tau}, \tau \to 0_{+} \quad (4)$$

where $u, v \in \mathbb{R}^n$ are two vectors of same dimensions. $\tau \in \mathbb{R}^n$ denotes a small perturbation term, the symbol $\odot$ represents the element-wise product operation, and it is worthwhile mentioning the square root also applies to the vector element-wisely.

In accordance with the locally Lipschitz continuity condition detailed in Effati & Nazemi (2006), this study asserts the uniqueness of the optimal solution for the TVQP problem (2). Supported by Lemma 1, the TVQP problem (2) can be effectively resolved via determining the solution to a time-variant quasi-linear equation (TVQLE):

$$f(y(t), t) = P(t)y(u) + q(t) = 0 \quad (5)$$

where $y(t) = [x^{*\mathrm{T}}(t), \phi^{*\mathrm{T}}(t), \varphi^{*\mathrm{T}}(t)]^{\mathrm{T}} \in \mathbb{R}^{n+m+p}$, and

$$P(t) = \begin{bmatrix} H(t) & A^{\mathrm{T}}(t) & C^{\mathrm{T}}(t) \\ A(t) & 0 & 0 \\ -C(t) & 0 & I \end{bmatrix}, q(t) = \begin{bmatrix} \rho(t) \\ -b(t) \\ d(t) - n(t) \end{bmatrix}$$

with $n(t) = \sqrt{m(t) \odot m(t) + \varphi^{*}(t) \odot \varphi^{*}(t) + \tau}$ and $m(t) = d(t) - C(t)x^{*}(t)$.

Thus, a vector $y(t)$ satisfying $f(y(t), t) = 0$ is recognized as a solution to (3), and the first n elements of $y(t)$ correspond to the solution for the TVQP problem (2). The solution to the TVQLE (5) is achievable through the application of two classes of RNNs, which we will explore in detail in the subsequent section.

2.2. Traditional GNN and ZNN models

The following discussion includes an examination of two prevalent RNN models: the GNN model (LeCun et al., 1998) and the ZNN model (Zhang et al., 2002).

2.2.1. GNN model

For the scalar cost function $e(t) = (f(y(t), t))^2/2$, the dynamics of the GNN model is expressed as follows:

$$\dot{y}(t) = -\gamma \nabla e(y) = -\gamma \frac{\partial e(t)}{\partial y(t)} = -\gamma M^{\mathrm{T}}(t) f(y(t), t) \quad (6)$$

where $\gamma > 0$ denotes the learning rate or gain factor, $\nabla e(y) := \partial e(t)/\partial y(t)$ stands for the gradient of the cost function with respect to $y(t)$, and $M(t)$ represents the coefficient matrix in (8).

Remark 2. To improve convergence performance, innovative GNN variants have been developed (Liao et al., 2020; Yang et al., 2024d; Yu et al., 2024). A prime example is the fractional-order GNN (FO-GNN) model (Yang et al., 2024d), which is distinguished by its high accuracy and accelerated convergence rates. Nonetheless, the FO-GNN model presents significant limitations: the convergence time cannot be explicitly predefined by users, and its resistance to additive noise remains inadequate.

Remark 3. In scenarios involving time-variant systems, the GNN model (6) requires frequent recomputation at each time step, which can lead to persistent residual errors or delayed convergence, particularly when the computational iterations are limited (Li et al., 2021). In contrast, ZNNs employ the evolutionary dynamics involving the derivative information of the vector-valued error function, theoretically eradicates the residual errors common in GNN applications.

2.2.2. ZNN model

The traditional ZNN model is represented by the RNN described in (7), which is designed to strategically drive the residual error $\epsilon(t) = f(y(t), t)$ towards zero,

$$\dot{\epsilon}(t) = -\gamma \Phi(\epsilon(t)) \quad (7)$$

where $\gamma > 0$ represents the learning rate or gain factor, and $\Phi(\cdot) : \mathbb{R}^{n+m+p} \to \mathbb{R}^{n+m+p}$ signifies the activation function.

The evolution dynamics in (7) can be expanded to produce an alternative formulation for the ZNN model, i.e.,

$$M(t)\dot{y}(t) = -N(t)y(t) - \sigma(t) - \gamma \Phi(\epsilon(t)) \quad (8)$$

where

$$M(t) = \begin{bmatrix} H(t) & A^{\mathrm{T}}(t) & C^{\mathrm{T}}(t) \\ A(t) & 0 & 0 \\ \left(\Pi_1(t) - I\right) C(t) & 0 & I - \Pi_2(t) \end{bmatrix}$$

$$N(t) = \begin{bmatrix} \dot{H}(t) & \dot{A}^{\mathrm{T}}(t) & \dot{C}^{\mathrm{T}}(t) \\ \dot{A}(t) & 0 & 0 \\ \left(\Pi_1(t) - I\right) \dot{C}(t) & 0 & 0 \end{bmatrix}$$

$$\sigma(t) = \begin{bmatrix} \dot{\rho}(t) \\ -\dot{b}(t) \\ \left(I - \Pi_1(t)\right) \dot{d}(t) \end{bmatrix}$$

with $\Pi_1(t) = \mathrm{diag}(m(t) \oslash n(t))$ and $\Pi_2(t) = \mathrm{diag}(\varphi^{*}(t) \oslash n(t))$, where $\oslash$ denotes the Hadamard division and diag() represents the diagonalization operation.

In hardware realization of an RNN, extraneous noises are inevitably present. A typical noise-perturbed ZNN used for solving the TVQLE (5) can be characterized as follows,

$$M(t)\dot{y}(t) = -N(t)y(t) - \sigma(t) - \gamma \Phi(\epsilon(t)) + \delta(t) \quad (9)$$

where $\delta(t) \in \mathbb{R}^{n+m+p}$ represents additive noises, including computational roundoff errors and states' measurement inaccuracies.

In practical scenarios, robotic and mechatronic systems encounter various types of random noise. Typically, computational errors such as truncation and discretization present at lower frequencies, while inaccuracies in state measurement manifest as high-frequency disturbances. Furthermore, the limits of these noises are often not well-defined. Consequently, the following assumption is integral to our analysis:

Assumption 1. *The additive noise is presumed bounded, that is, $\|\delta(t)\|_\infty \leq \Delta$, with Δ representing a constant upper bound for the noise magnitude.*

Remark 4. This assumption is rooted in the operational norms of conventional mechatronic systems, which are not subject to infinitely large or unbounded noise levels. The noise boundaries are typically specified in machine datasheets, expressed in units like decibels or signal-to-noise ratios, which provide tangible metrics essential for our investigations. Furthermore, more complex noise conditions, such as mixed-frequency noises, can be treated as linear combinations of the noise conditions discussed above. We have no doubt that the proposed model in our work is also capable of effectively tackling these more complex noisy systems.

Prior to exploring the design of the fractional-order neural model, we present a foundational definition and lemma.

Definition 6. *(Khalil et al., 2014) The conformable fractional derivative of a n-th order differentiable function $h(t)$ is defined as:*

$$W_\alpha(h)(t) = \lim_{s\to 0} \frac{h^{(n)}(t + st^{n+1-\alpha}) - h^{(n)}(t)}{s} \tag{10}$$

where $n < \alpha \leq n+1$ represents the order of the conformable fractional derivative operator with $n \geq 0$ being an integer.

A function h is considered α-differentiable at a point $t > 0$ if the operator $W_\alpha(h)(t)$ is defined at that point. Furthermore, when $0 < \alpha \leq 1$, the conformable fractional derivative exhibits specific characteristics.

Lemma 2. *(Khalil et al., 2014) For $0 < \alpha \leq 1$, if f and g are two α-differentiable functions at a point $t > 0$, then*

(a) $W_\alpha(af + bg) = aW_\alpha(f) + bW_\alpha(g)$, *for all* $a, b \in \mathbb{R}$.

(b) $W_\alpha(t^c) = ct^{c-\alpha}$ *for all* $c \in \mathbb{R}$.

(c) $W_\alpha(C) = 0$ *for all constant function* $f(t) \equiv C$.

(d) $W_\alpha(fg) = fW_\alpha(g) + gW_\alpha(f)$.

(e) $W_\alpha(f/g) = (gW_\alpha(f) - fW_\alpha(g))/g^2$.

(f) $W_\alpha(f)(t) = t^{1-\alpha}f'(t)$ *for any* $f(t) \in C^1(-\infty, +\infty)$.

Remark 5. Traditional definitions of fractional derivatives, such as Riemann-Liouville and Caputo's definitions, typically do not adhere to the Leibniz rule (Yang et al., 2024b, 2025), impacting the applicability of properties (d), (e), and (f) outlined in Lemma 2. In contrast, the conformable fractional derivative, as defined in (10), maintains compliance with the Leibniz rule. This adherence enhances its suitability for extending traditional recurrent neural network models into the fractional-order domain. Moreover, conformable fractional derivatives, unlike traditional definitions that involve complex integral operations, offer a more straightforward formulation that directly generalizes the integer-order derivative (Yang et al., 2024c). This simplifies the model's computational requirements and increases precision by reducing computational complexity and improving accuracy when evaluating transient states between consecutive integer-order states. This efficiency is particularly beneficial in neural network controllers tasked with managing highly dynamic and non-linear system behaviors.

3. Scheme design and convergence analysis

This section elaborates on the design of the SPTC-AN-FOZNN model. It also provides theoretical justifications to support model's strictly predefined-time convergence and its noise resilience.

3.1. Design of SPTC-AN-FOZNN model

A fractional-order ZNN (FOZNN) model is formulated by substituting $\dot{\epsilon}(t)$ in (7) with conformable fractional derivative of $\epsilon(t)$, i.e.,

$$W_\alpha(\epsilon)(t) = t^{1-\alpha}\dot{\epsilon}(t) = -\gamma\Phi(\epsilon(t)) \tag{11}$$

with its expanded expression given as

$$M(t)\dot{y}(t) = -N(t)y(t) - \sigma(t) - \gamma t^{\alpha-1}\Phi(\epsilon(t)) \tag{12}$$

where the order α is prescribed within $0 < \alpha \leq 1$, and the term $t^{\alpha-1}$ is derived from the property (f) associated with the conformable fractional derivative as outlined in Lemma 2. Then, the noise-perturbed FOZNN model is presented as

$$M(t)\dot{y}(t) = -N(t)y(t) - \sigma(t) - \gamma t^{\alpha-1}\Phi(\epsilon(t)) + \delta(t) \tag{13}$$

To confer the strictly predefined-time convergence attribute upon the FOZNN model (11) with or without external noises, we propose the following predefined-time stabilizer to serve as the activation function,

$$\Phi(x) = \begin{cases} \frac{1}{\gamma t_c^{\alpha-1}}\left(\frac{x}{t_c - t} + \frac{\gamma\|x\|^2}{(t_c-t)^2}\frac{x}{\|x\|}\right), t < t_c \\ \frac{1}{\gamma t^{\alpha-1}}\left(\frac{x}{t_p - t} + \left(\zeta + \frac{\gamma t^{\alpha-1}\|x\|^2}{(t_p-t)^2}\right)\frac{x}{\|x\|}\right), t \geq t_c \end{cases} \tag{14}$$

where $t_p = (1 - \exp(-\pi/(2\sqrt{\zeta(\zeta - \Delta)})))t_c$ with $t_c > 0$ being the predefined time, and ζ being an arbitrary positive number greater than Δ. Then, (12) and (13), combined with (14), establish the SPTC-AN-FOZNN model for solving the TVQP problem (2).

3.2. Main results in theoretical analysis

Theorems presented herein guarantee the strictly predefined-time convergence of FOZNNs (12) and (13) under the activation function (14).

Theorem 1. *For the TVQP problem (2) or equivalently, the TVQLE (5), if the predefined-time stabilizer (14) is utilized, the neural state vector $y(t)$ for the non-noise FOZNN model (12), originating from an initial condition y_0 sufficiently close to the theoretical initial state y_0^*, can converge to the theoretical solution $y^*(t)$ in strictly predefined time t_c.*

Table 1
Representative RNN models published recently and their convergence characteristics.

Model	Gain	Formulation or activation function	Convergence ability	Noise
FO-GNN (Yang et al., 2024d)	$\gamma(t)=\gamma$	$\dot{y}(t)=-\gamma\dfrac{M^{\mathrm{T}}(t)\,\epsilon(t)}{\Gamma(2-\alpha)}\odot\left\lvert y_i-y_{i-1}+\varepsilon\right\rvert^{1-\alpha}$	INFTC	No
PRAGNN (Yu et al., 2024)	$\gamma(t)=\gamma$	$\dot{y}(t)=-k(t)M^{\mathrm{T}}(t)\epsilon(t)-\gamma_2 M^{\mathrm{T}}(t)\epsilon(t)-\gamma_3\dfrac{M^{\mathrm{T}}(t)\epsilon(t)}{\lVert M^{\mathrm{T}}(t)\epsilon(t)\rVert}$	WPTC	Yes
SPTC-NT-ZNN (Li et al., 2024)	$\gamma(t)=\gamma$	$\Phi(x)=\begin{cases} x/(t_c-t), t<t_c \\ x+\lvert x\rvert^p\,\mathrm{sign}(x)+\xi\,\mathrm{sign}(x), t\geq t_c\end{cases}$	SPTC	Yes
NIFZNN (Chen et al., 2024)	$\gamma(t)=\gamma$	$\dot{y}=-\dfrac{M^{\mathrm{T}}(t)\epsilon(t)}{\lVert M^{\mathrm{T}}(t)\epsilon(t)\rVert^2}\left(\epsilon^{\mathrm{T}}(t)\dfrac{\partial\epsilon(t)}{\partial t}+\gamma\dfrac{\lVert\epsilon(t)\rVert^2}{2}\right)$	INFTC	No
PTC-FOZNN (Yang et al., 2024a)	$\gamma(t)=\gamma t^{\alpha-1}$	$\Phi(x)=\dfrac{\pi}{2\kappa\gamma t_c^{\alpha}}\left(\lVert x\rVert^{1-\kappa}+\lVert x\rVert^{1+\kappa}\right)\dfrac{x}{\lVert x\rVert}$	WPTC	No
SPTC-AN-FOZNN	$\gamma(t)=\gamma t^{\alpha-1}$	$\Phi(x)=\begin{cases}\dfrac{1}{\gamma t_c^{\alpha-1}}\left(\dfrac{x}{t_c-t}+\dfrac{\gamma\lVert x\rVert^2}{(t_c-t)^2}\dfrac{x}{\lVert x\rVert}\right), t<t_c \\ \dfrac{1}{\gamma t^{\alpha-1}}\left(\dfrac{x}{t_p-t}+\left(\zeta+\dfrac{\gamma t^{\alpha-1}\lVert x\rVert^2}{(t_p-t)^2}\right)\dfrac{x}{\lVert x\rVert}\right), t\geq t_c\end{cases}$	SPTC for $0<\alpha\leq 1$	Yes

Note: "INFTC", "SPTC" and "WPTC" means convergence in infinite time, strictly, and weakly predefined time, respectively. Besides, the listed activation function should be combined with (9) or (13) to give the formulation of the specific model.

Proof. For $t<t_c$, integrating the designed predefined-time stabilizer (14) into the FOZNN model (12) yields:

$$\dot{\epsilon}(t)=-\frac{t^{\alpha-1}}{t_c^{\alpha-1}}\left(\frac{\epsilon(t)}{t_c-t}+\gamma\frac{\lVert\epsilon(t)\rVert^2}{\left(t_c-t\right)^2}\frac{\epsilon(t)}{\lVert\epsilon(t)\rVert}\right) \tag{15}$$

Design a Lyapunov functional candidate as $V(t)=\lVert\epsilon(t)\rVert/(t_c-t)$. For $t<t_c$, the time derivative of $V(t)$ is

$$\begin{aligned}\dot{V}(t)&=\frac{\epsilon^{\mathrm{T}}(t)}{\lVert\epsilon(t)\rVert\left(t_c-t\right)}\dot{\epsilon}(t)+\frac{\lVert\epsilon(t)\rVert}{\left(t_c-t\right)^2}\\&=-\frac{t^{\alpha-1}}{\left(t_c-t\right)t_c^{\alpha-1}}\left(\frac{\lVert\epsilon(t)\rVert}{t_c-t}+\gamma\frac{\lVert\epsilon(t)\rVert^2}{\left(t_c-t\right)^2}\right)+\frac{\lVert\epsilon(t)\rVert}{\left(t_c-t\right)^2}\\&\leq-\frac{\gamma V^2}{t_c-t}\leq 0\end{aligned} \tag{16}$$

The Lyapunov stability theorem suggests that the origin is globally finite-time stable. Subsequently, integrating on both sides of the differential inequality (16) yields:

$$\ln\frac{t_c}{t_c-t}=\int_0^t\frac{\mathrm{d}\tau}{t_c-\tau}\leq\int_{V(0)}^{V(t)}\frac{\mathrm{d}V}{-\gamma V^2}=\frac{1}{\gamma}\left(\frac{1}{V(t)}-\frac{1}{V(0)}\right) \tag{17}$$

Then, (17) implies that $V(t)\to 0$ as $t\to t_c$. Thus, one can obtain that $\lVert\epsilon(t)\rVert=(t_c-t)V(t)\to 0$ as $t\to t_c$. For $t\geq t_c$, substituting the activation function (14) into the non-noise FOZNN model (12) produces:

$$\dot{\epsilon}(t)=-\left(\frac{\epsilon(t)}{t_p-t}+\left(\zeta+\gamma t^{\alpha-1}\frac{\lVert\epsilon(t)\rVert^2}{\left(t_p-t\right)^2}\right)\frac{\epsilon(t)}{\lVert\epsilon(t)\rVert}\right) \tag{18}$$

For $t\geq t_c$, we consider a Lyapunov functional candidate $V(t)=\lVert\epsilon(t)\rVert/(t-t_p)$, and obtains that

$$\begin{aligned}\dot{V}(t)&=\frac{\epsilon^{\mathrm{T}}(t)}{\lVert\epsilon(t)\rVert\left(t-t_p\right)}\dot{\epsilon}(t)-\frac{\lVert\epsilon(t)\rVert}{\left(t-t_p\right)^2}\\&=\frac{1}{t_p-t}\left(\frac{\lVert\epsilon(t)\rVert}{t_p-t}+\gamma t^{\alpha-1}\frac{\lVert\epsilon(t)\rVert^2}{\left(t_p-t\right)^2}+\zeta\right)-\frac{\lVert\epsilon(t)\rVert}{\left(t-t_p\right)^2}\\&\leq-\frac{\gamma t^{\alpha-1}V^2}{t-t_p}-\frac{\zeta}{t-t_p}<0\end{aligned} \tag{19}$$

which is always negative, indicating that $\epsilon(t)=0$ is strictly maintained for all $t>t_c$. This suggests that the non-noise FOZNN model (12) with the activation function (14) exhibits strictly predefined-time convergence. ■

Theorem 2. *For the TVQP problem (2) or equivalently, the TVQLE (5), if the predefined-time stabilizer (14) is utilized, the neural state vector $y(t)$ for the noise-perturbed FOZNN model (13), originating from an initial condition y_0 sufficiently close to the theoretical initial state y_0^*, can converge to the theoretical solution $y^*(t)$ in strictly predefined time t_c.*

Proof. For $t < t_c$, substituting the designed predefined-time stabilizer (14) into the FOZNN model (13) yields:

$$\dot{\epsilon}(t) = -\frac{t^{\alpha-1}}{t_c^{\alpha-1}}\left(\frac{\epsilon(t)}{t_c - t} + \gamma\frac{\|\epsilon(t)\|^2}{\left(t_c - t\right)^2}\frac{\epsilon(t)}{\|\epsilon(t)\|}\right) + \delta(t) \quad (20)$$

Design a Lyapunov functional candidate as $V(t) = \|\epsilon(t)\|/(t_c - t)$. For $t < t_c$, the time derivative of $V(t)$ is

$$\begin{aligned}\dot{V}(t) =& \frac{\epsilon^{\mathrm{T}}(t)}{\|\epsilon(t)\|\left(t_c - t\right)}\dot{\epsilon}(t) + \frac{\|\epsilon(t)\|}{\left(t_c - t\right)^2}\\ =& -\frac{t^{\alpha-1}}{\left(t_c - t\right)t_c^{\alpha-1}}\left(\frac{\|\epsilon(t)\|}{t_c - t} + \gamma\frac{\|\epsilon(t)\|^2}{\left(t_c - t\right)^2}\right)\\ &+ \frac{\epsilon^{\mathrm{T}}(t)\delta(t)}{\|\epsilon(t)\|\left(t_c - t\right)} + \frac{\|\epsilon(t)\|}{\left(t_c - t\right)^2}\\ \leq& -\frac{\gamma V^2}{t_c - t} + \frac{\Delta}{t_c - t}\end{aligned} \quad (21)$$

The Lyapunov stability theorem suggests that the origin is globally finite-time stable. Subsequently, integrating on both sides of the differential inequality (21) gives:

$$\begin{aligned}\ln\frac{t_c}{t_c - t} = \int_0^t \frac{\mathrm{d}\tau}{t_c - \tau} &\leq \int_{V(0)}^{V(t)} \frac{\mathrm{d}V}{\Delta - \gamma V^2}\\ &= \frac{1}{2\sqrt{\gamma\Delta}}\ln\frac{|(u(0) - 1)(u(t) + 1)|}{|(u(0) + 1)(u(t) - 1)|}\end{aligned} \quad (22)$$

where $u(t) = V(t)\sqrt{\gamma/\Delta}$. Then, from (22), one can observe that $u(t) \to 1$ and $V(t) \to \sqrt{\Delta/\gamma}$ as $t\pi t_c$. Thus, one can deduce that $\|\epsilon(t)\| = (t_c - t)V(t) \to 0$ as $t\pi t_c$.

For $t \geq t_c$, substituting the activation function (14) into the noise-perturbed FOZNN model (13) produces:

$$\dot{\epsilon}(t) = -\left(\frac{\epsilon(t)}{t_p - t} + \left(\zeta + \gamma t^{\alpha-1}\frac{\|\epsilon(t)\|^2}{\left(t_p - t\right)^2}\right)\frac{\epsilon(t)}{\|\epsilon(t)\|}\right) + \delta(t) \quad (23)$$

For $t \geq t_c$, we consider a Lyapunov functional candidate $V(t) = \|\epsilon(t)\|/(t - t_p)$, and obtains that

$$\begin{aligned}\dot{V}(t) =& \frac{\epsilon^{\mathrm{T}}(t)}{\|\epsilon(t)\|\left(t - t_p\right)}\dot{\epsilon}(t) - \frac{\|\epsilon(t)\|}{\left(t - t_p\right)^2}\\ =& \frac{1}{t_p - t}\left(\frac{\|\epsilon(t)\|}{t_p - t} + \gamma t^{\alpha-1}\frac{\|\epsilon(t)\|^2}{\left(t_p - t\right)^2} + \zeta\right) - \frac{\|\epsilon(t)\|}{\left(t - t_p\right)^2}\\ &+ \frac{\epsilon^{\mathrm{T}}(t)\delta(t)}{\|\epsilon(t)\|\left(t - t_p\right)}\\ \leq& -\frac{\gamma t^{\alpha-1}V^2}{t - t_p} - \frac{\zeta - \Delta}{t - t_p} < 0\end{aligned} \quad (24)$$

which is always negative, indicating that $\epsilon(t) = 0$ is strictly maintained for all $t > t_c$. This suggests that the noise-perturbed FOZNN model (13) with the activation function (14) exhibits strictly predefined-time convergence. ■

Remark 6. The SPTC-AN-FOZNN is evaluated against a range of recently developed RNN models as detailed in Table 1. While not the pioneer in achieving strictly predefined-time convergence and noise resilience—attributes also seen in the SPTC-NT-ZNN model (Li et al., 2024)—the SPTC-AN-FOZNN introduces two notable innovations. Firstly, it employs a fractional-order approach, utilizing a time-diminishing gain factor, $\gamma(t) = \gamma t^{\alpha-1}$. This approach enhances energy efficiency in hardware implementations, especially as the system approaches the steady state, by reducing power consumption as $t \to \infty$. Secondly, unlike the SPTC-NT-ZNN, whose piecewise activation function may induce fluctuations in residual errors at the transition point t_c, the SPTC-AN-FOZNN uses a structurally coherent piecewise function that ensures a smoother transition in error dynamics at critical time points. These enhancements not only contribute to greater energy efficiency but also ensure more stable convergence behavior, underscoring the SPTC-AN-FOZNN's advancement over existing models. However, it is worthwhile noting that the computational cost of the SPTC-AN-FOZNN model, similar to the SPTC-NT-ZNN and PTC-FOZNN, lags behind the FO-GNN, PRAGNN, and NIFZNN models, which benefit from simpler operations with lower algorithmic complexity.

Remark 7. The strictly predefined-time convergence characteristics speculated in the Theorem 1 and Theorem 2 can be extended to $\alpha > 1$ for the SPTC-AN-FOZNN model. Consider $\alpha > 1$, the first derivative of $V(t)$ in (16) becomes:

$$\begin{aligned}\dot{V}(t) =& -\frac{t^{\alpha-1}}{\left(t_c - t\right)t_c^{\alpha-1}}\left(\frac{\|\epsilon(t)\|}{t_c - t} + \gamma\frac{\|\epsilon(t)\|^2}{\left(t_c - t\right)^2}\right) + \frac{\|\epsilon(t)\|}{\left(t_c - t\right)^2}\\ \leq& -\frac{t}{\left(t_c - t\right)t_c}\left(\frac{\|\epsilon(t)\|}{t_c - t} + \gamma\frac{\|\epsilon(t)\|^2}{\left(t_c - t\right)^2}\right) + \frac{\|\epsilon(t)\|}{\left(t_c - t\right)^2}\\ \leq& \frac{-\gamma t V^2}{(t_c - t)t_c} + \frac{V}{t_c}\end{aligned} \quad (25)$$

Note that the above differential inequality implies that $V(t) \to 0$ as $t \to 0$. This is because as t gets very close to t_c, the first term $-\gamma t V^2/((t_c - t)t_c)$ might dominate due to the small value in the denominator. A key aspect to consider is how $V(t)$ behaves to prevent the term from diverging. If $V(t)$ remains bounded and doesn't tend to zero, this term could lead to a scenario where $\dot{V}(t)$ becomes very large negatively (since V^2 in the numerator could not offset the rapid decrease in $t_c - t$ in the denominator), pushing $V(t)$ to decrease rapidly. However, if $V(t)$ tends to zero faster than $t_c - t$ tends to zero, the term could remain bounded.

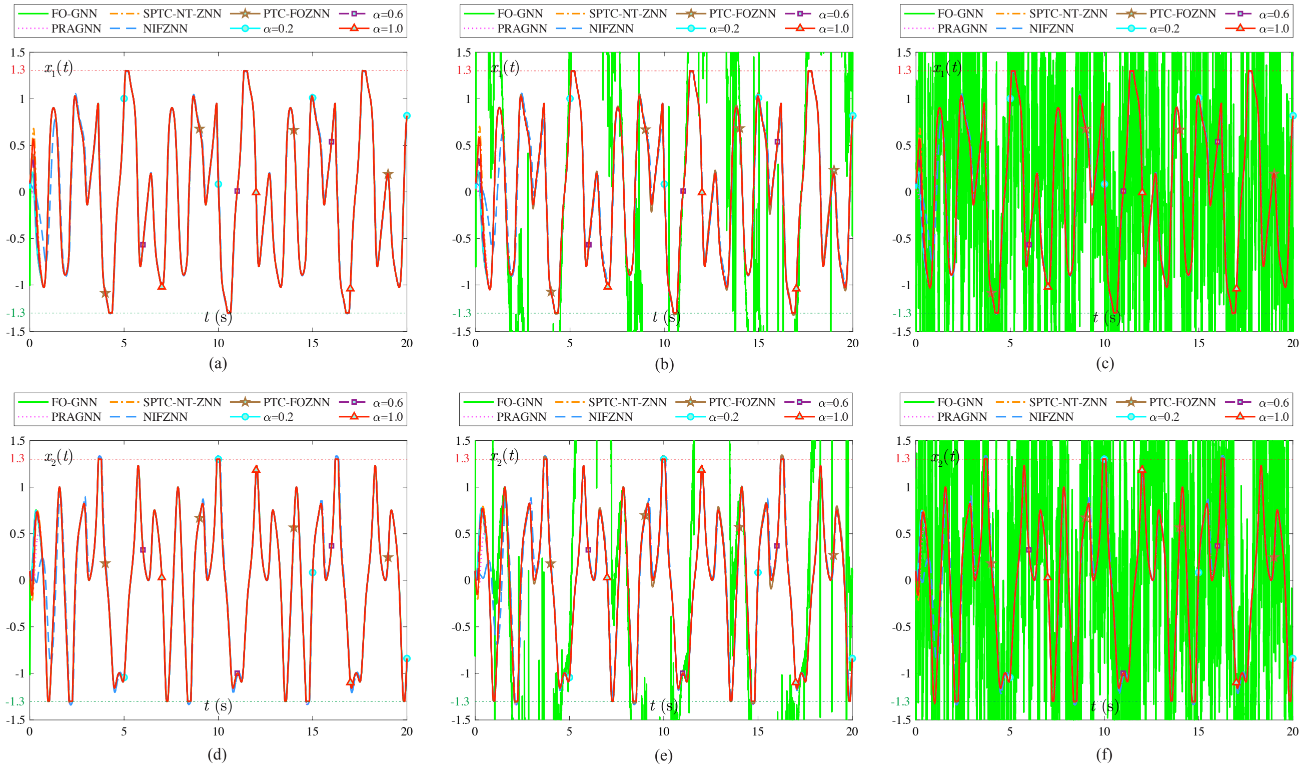


Figure 1: Profiles of neural states x_1 and neural state x_2 across six models for solving the TVQP problem (26) under the following noise conditions: (a) and (d) without additive noise; (b) and (e) a low-frequency noise $\delta(t) = 0.2\cos(t)$; (c) and (f) a high-frequency random noise $\delta(t) = 0.5\bar{n}(t)$, where $\bar{n}(t)$ is a white noise signal bounded by 1. (Our SPTC-AN-FOZNN model takes three distinct α values)

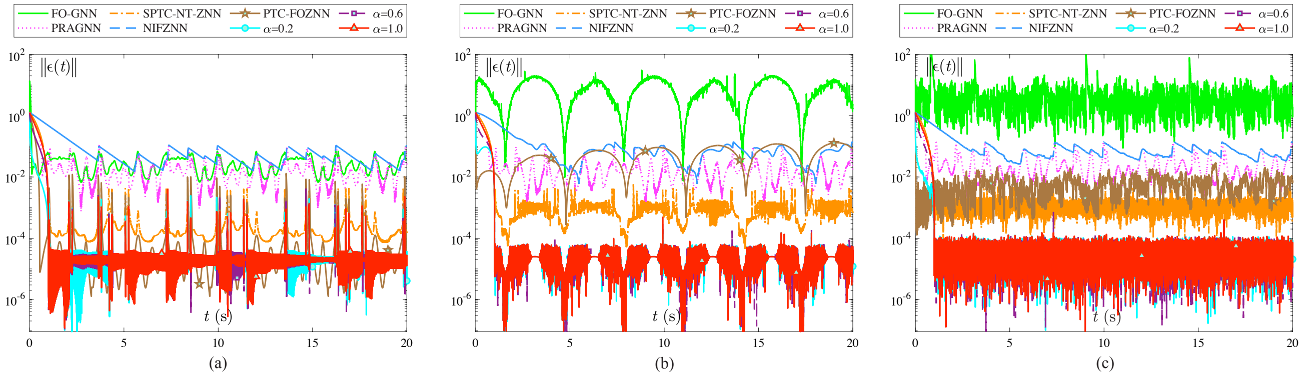


Figure 2: Profiles of the residual error $\|\epsilon(t)\|$ across six models for solving the TVQP problem (26) under the following noise conditions: (a) without additive noise; (b) a low-frequency noise $\delta(t) = 0.2\cos(t)$; (c) a high-frequency random noism $\delta(t) = 0.5\bar{n}(t)$, where $\bar{n}(t)$ is a white noise signal bounded by 1. (Our SPTC-AN-FOZNN odel takes three distinct α values.)

4. Numerical validation

The efficacy of the proposed SPTC-AN-FOZNN model is evaluated through a comparative study with five recent RNN models, including the FO-GNN model (Yang et al., 2024d), PRAGNN model (Yu et al., 2024), SPTC-NT-ZNN model (Li et al., 2024), NIFZNN model (Chen et al., 2024), and the PTC-FOZNN model (Yang et al., 2024a). This assessment is performed on the following TVQP problem:

$$
\begin{aligned}
\min \quad & (\sin(t)/8 + 1/2)x_1^2(t) + (\cos(t)/8 + 1/2)x_2^2(t) \\
& + \cos(t)x_1(t)x_2(t)/2 + \cos(3t)x_1(t) \\
& + \sin(3t)x_2(t) \qquad\qquad\qquad\qquad (26) \\
\text{s.t.} \quad & \cos(4t)x_1(t) + \sin(4t)x_2(t) = \sin(2t) \\
& -1.3 \le x_1(t), x_2(t) \le 1.3
\end{aligned}
$$

where the coefficients associated to the compact form (2) are $A = [\cos(4t), \sin(4t)]$, $b = \sin(2t)$, $d = [1.3, 1.3, 1.3, 1.3]^{\mathrm{T}}$, $C = [I, -I]^{\mathrm{T}}$, and

$$
H = \begin{bmatrix} \sin(t)/4+1 & \cos(t)/2 \\ \cos(t)/2 & \cos(t)/4+1 \end{bmatrix}, \rho = \begin{bmatrix} \cos(3t) \\ \sin(3t) \end{bmatrix}
$$

where I denotes an identity matrix. All six models under consideration are discretized using the forward Euler method, with a discrete step size $\Delta t = 1 \times 10^{-3}$ seconds, ensuring a consistent evaluation framework. The models employ a constant gain factor $\gamma = 2$, a predefined time $t_c = 1$ s, and parameters $p = 0.5$, $\kappa = 0.5$, $\Delta = \|\delta(t)\|_\infty$, $\zeta = 5\Delta$, $\gamma_2 = 0$, $\gamma_3 = \zeta$, and $\xi = \zeta/\gamma$ are applied to all neural models. Additionally, $\tau = 1 \times 10^{-8}$ is set for the perturbed FB function.

In an enhancement to the standard configuration, our SPTC-AN-FOZNN model introduces a variable gain $\gamma(t) = \gamma t^{\alpha-1}$ where $0 < \alpha \le 1$, aimed at reducing power consumption in hardware implementations through quick mitigation of the model's gain. In this simulation, we consider the three types of additive noise simulated in the computation and hardware implementation/measurements, namely, $\delta(t) = 0$, $\delta(t) = 0.2\cos(t)$, and $\delta(t) = 0.5\bar{n}(t)$, with $\bar{n}(t)$ being a white noise signal bounded by 1. The neural state profiles for all models, under the influence of three types of noise conditions, are depicted in Fig. 1. Fig. 2 showcases the residual error profiles for the six models, both in noise-free conditions and under the two specified noise scenarios. It is noteworthy that the SPTC-AN-FOZNN model, particularly with three distinct values of α, demonstrates superior precision and enhanced noise immunity compared to its counterparts. Notably, this model achieves convergence (e.g., $\|\epsilon(t)\| \sim 10^{-5}$) within the strictly predefined time 1 second for all three noise scenarios, reducing residual error by over 90% relative to the SPTC-NT-ZNN and other RNN models. These findings affirm the SPTC-AN-FOZNN model's exceptional efficacy in addressing general TVQP problems.

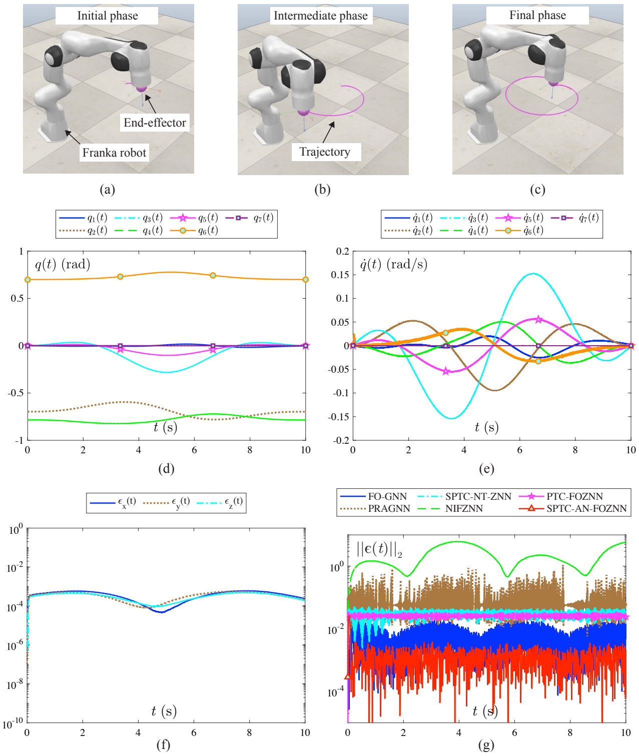


Figure 3: The snapshots of the (a) initial, (b) intermediate, and (c) final phases for a simulated Franka Emika Panda robot during tracking a heart-shaped path, with the rendered (d) joint angles, (e) joint velocities, (f) absolute tracking errors by the SPTC-AN-FOZNN model, and (g) residual errors rendered by six different neural models with a bounded random noise $\delta(t) = \cos(t) + \bar{n}(t)$.

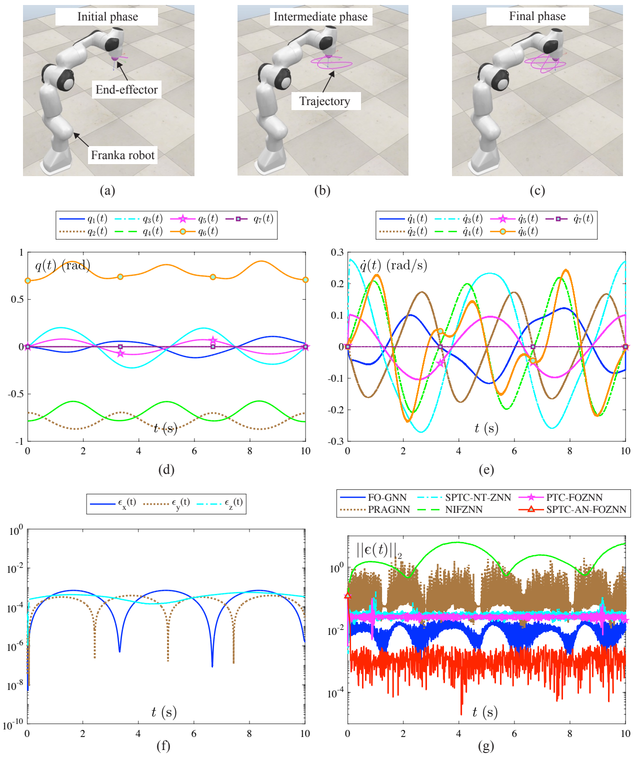
Further analysis reveals that in high dynamic range environments, where signal variability can be intense and unpredictable, the SPTC-AN-FOZNN model maintains optimal performance due to its adaptive gain adjustment, $\gamma(t)$. This adjustment allows the model to respond more efficiently to sudden changes in signal amplitude, thereby ensuring faster convergence rates and substantially reduced error margins. Additionally, in scenarios characterized by rapid system state changes, the model's ability to quickly adapt its gain in response to the state's dynamics proves invaluable. These attributes are particularly beneficial in applications demanding stringent real-time performance and robust noise immunity, such as in autonomous robotics and adaptive signal processing. Such applications often face scenarios where the rapid and accurate adjustment of control actions is critical to system stability and operational success, underscoring the practical relevance and superior performance of the SPTC-AN-FOZNN model.

5. Kinematic control of robotic manipulators

This section presents the application of the SPTC-AN-FOZNN model to the kinematic motion control of robotic manipulators through the formulation of the TVQP problem below (Li et al., 2021; Zhang & Jia, 2023):

$$
\begin{aligned}
\min \quad & \dot{q}^{\mathrm{T}}(t)\dot{q}(t)/2 + \rho^{\mathrm{T}}(t)\dot{q}(t) \\
\text{s.t.} \quad & J(t)\dot{q}(t) = \dot{w}(t) \\
& q^- \le q(t) \le q^+ \qquad\qquad\qquad (27) \\
& \dot{q}^- \le \dot{q}(t) \le \dot{q}^+
\end{aligned}
$$

where $q(t), \dot{q}(t) \in \mathbb{R}^n$ represent the joint angles and velocities of the robotic manipulator, respectively. The vector $\rho(t) = \iota(q(t) - q_0)$ aids the repetitive motion of the robotic manipulator, with $\iota \in \mathbb{R}$ being a positive constant and q_0 the



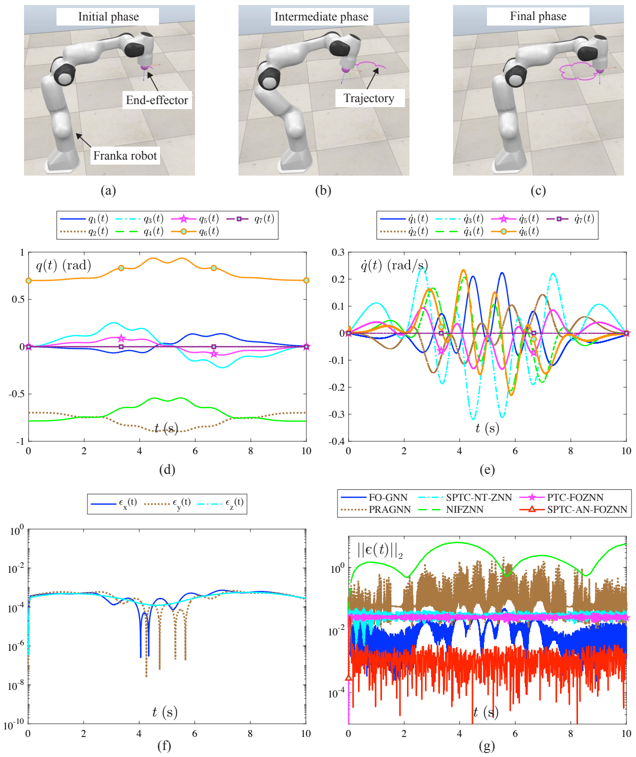


Figure 4: The snapshots of the (a) initial, (b) intermediate, and (c) final phases for a simulated Franka Emika Panda robot during tracking a Lissajous curve, with the rendered (d) joint angles, (e) joint velocities, (f) absolute tracking errors by the SPTC-AN-FOZNN model, and (g) residual errors rendered by six different neural models with a bounded random noise $\delta(t) = \cos(t) + \bar{n}(t)$.

Figure 5: The snapshots of the (a) initial, (b) intermediate, and (c) final phases for a simulated Franka Emika Panda robot during tracking a five-petal-plum-shaped path, with the rendered (d) joint angles, (e) joint velocities, (f) absolute tracking errors by the SPTC-AN-FOZNN model, and (g) residual errors rendered by six different neural models with a bounded random noise $\delta(t) = \cos(t) + \bar{n}(t)$.

initial joint angle. $J(t) \in \mathbb{R}^{3\times n}$ is the Jacobian matrix, and $w(t) \in \mathbb{R}^3$ signifies the desired trajectory of the end-effector. The bounds q^-, q^+, $\dot{q}^-$ and $\dot{q}^+$ specify the permissible ranges for joint angles and velocities.

The problem is transformed into the compact form (2) with $x(t) = \dot{q}(t)$, $H(t) = I$, $A(t) = J(t)$, $b(t) = \dot{w}(t)$, $C(t) = [I, -I]^{\mathrm{T}}$, and $d(t) = [d^{+\mathrm{T}}, -d^{-\mathrm{T}}]$. According to Li et al. (2021), extra functions are incorporated to ensure the smoothness of $d(t)$ at boundary points, leading to:

$$d^- = \begin{cases} \dot{q}^-, & \text{if } q(t) \in \left[\xi_1, q^+\right] \\ \dot{q}^- \left(1 - \varrho_1(q(t))\right), & \text{if } q(t) \in \left[q^-, \xi_1\right] \end{cases}$$
$$d^+ = \begin{cases} \dot{q}^+, & \text{if } q(t) \in \left[q^-, \xi_2\right] \\ \dot{q}^+ \left(1 - \varrho_2(q(t))\right), & \text{if } q(t) \in \left[\xi_2, q^+\right] \end{cases} \quad (28)$$

where $\varrho_1(x) = (\sin(0.5\pi(\sin(0.5\pi(x-\xi_1)/\xi_3))^2))^2$, $\varrho_2(x) = (\sin(0.5\pi(\sin(0.5\pi(x-\xi_2)/\xi_4))^2))^2$, $\xi_1 = \kappa_1 q^-$, $\xi_2 = \kappa_2 q^+$, $\xi_3 = q^- - \xi_1$, and $\xi_4 = q^+ - \xi_2$, with $0 < \kappa_1, \kappa_2 < 1$ being two positive constants.

5.1. Simulation validation

To enhance the convincibility of the simulation validation, two different types of robotic manipulators, namely the Franka Emika Panda robot (Haddadin et al., 2022) and the Flexiv Rizon robot (Murtaza & Hutchinson, 2022), are utilized to execute the tracking of three trajectory types:

- Heart-shaped curve:

$$\begin{cases} x(t) = a(2\sin(\theta) - \sin(2\theta)), \\ y(t) = a(2\cos(\theta) - \cos(2\theta)) - a, \\ z(t) = z_0, \end{cases} \quad (29)$$

where a is a scaling factor, z_0 is a constant, and $\theta = 2\pi t/T$ with T being the total simulation time.

- Lissajous curve:

$$\begin{cases} x(t) = A\sin(2\pi a t/T + \delta), \\ y(t) = B\sin(2\pi b t/T), \\ z(t) = z_0, \end{cases} \quad (30)$$

where $A = 0.06$ and $B = 0.06$ are amplitudes, $a = 3$ and $b = 2$ are frequency factors, and $\delta = \pi/2$ is the phase difference.

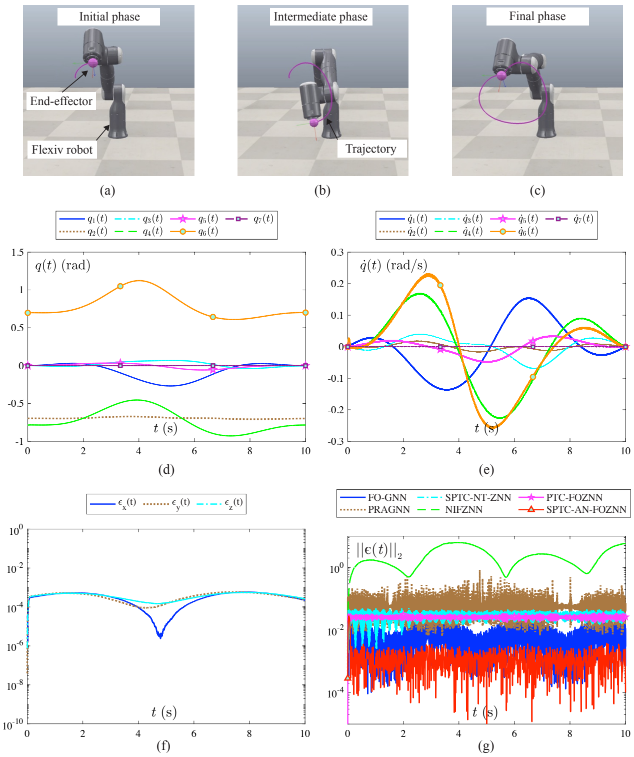


Figure 6: The snapshots of the (a) initial, (b) intermediate, and (c) final phases for a simulated Flexiv Rizon robot during tracking a heart-shaped path, with the rendered (d) joint angles, (e) joint velocities, (f) absolute tracking errors by the SPTC-AN-FOZNN model, and (g) residual errors rendered by six different neural models with a bounded random noise $\delta(t) = \cos(t) + \bar{n}(t)$.

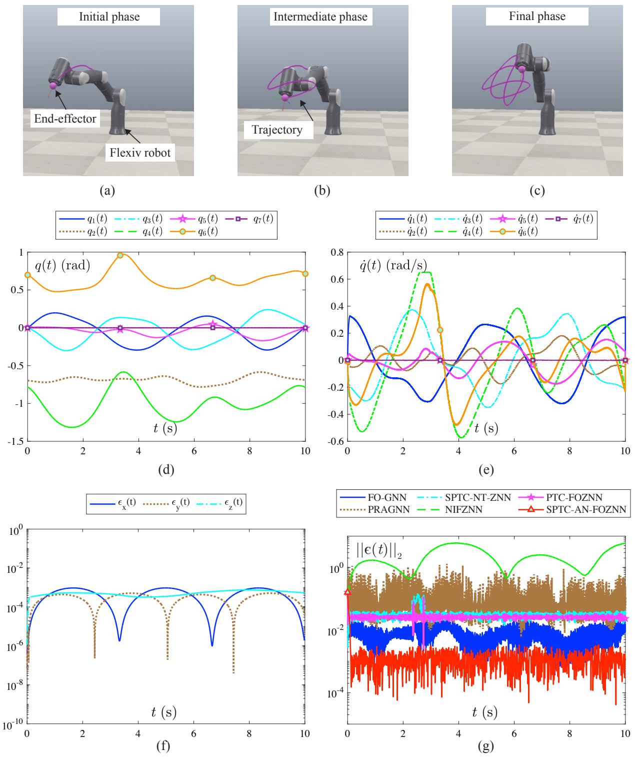


Figure 7: The snapshots of the (a) initial, (b) intermediate, and (c) final phases for a simulated Flexiv Rizon robot during tracking a Lissajous curve, with the rendered (d) joint angles, (e) joint velocities, (f) absolute tracking errors by the SPTC-AN-FOZNN model, and (g) residual errors rendered by six different neural models with a bounded random noise $\delta(t) = \cos(t) + \bar{n}(t)$.

- Five-petal-plum-shaped curve:

$$\begin{cases} x(t) = r\left(\cos(\phi) + \dfrac{\cos(n\phi)}{n}\right), \\ y(t) = r\left(\sin(\phi) + \dfrac{\sin(n\phi)}{n}\right), \\ z(t) = z_0, \end{cases} \tag{31}$$

  where $\phi = 2\pi \sin(\pi t/2/T) \sin(\pi t/2/T)$, $r = 0.1$ is the radius, and $n = 5$ is the number of lobes.

These trajectories provide a comprehensive test bed to evaluate the performance of the robotic arms under various dynamic conditions. The choice of complex trajectories like heart-shaped, Lissajous, and five-petal plum-shaped paths allows for rigorous assessment of the control algorithms in handling intricate movement patterns, crucial for real-world robotic applications.

The simulations are operationalized within the the CoppeliaSim's virtual environment (Rohmer et al., 2013), tackling a kinematic motion planning task with the noise-perturbed neural network controller. Specifically, the SPTC-AN-FOZNN model, configured with $\alpha = 0.5$ and a predefined convergence time of $t_c = 0.01$ s, addresses the control challenges posed by a random noise $\delta(t) = \cos(t) + \bar{n}(t)$, where $\bar{n}(t)$ represents the white noise bounded by 1.

In the context of motion planning, the robot's end-effector position is specified with precision, while the other degrees of freedom remain unconstrained. The range of joint angles for the two types of robotic manipulators is set as follows:

- Lower limit (q^-):

  $$-[161°, 131.5°, 172.5°, 107°, 172.5°, 82.5°, 172.5°]^{\mathrm{T}}$$

- Upper limit (q^+):

  $$+[161°, 131.5°, 172.5°, 155°, 172.5°, 262.5°, 172.5°]^{\mathrm{T}}$$

These limits are consistent with the specified joint angle range for the real robot arm. Additionally, the joint-velocity bounds are established at $\dot{q}^- = -0.65$ rad/s and $\dot{q}^+ = 0.65$ rad/s, with a scaling factor $\imath$ set to 1. This setup ensures that the robot operates within safe and efficient kinematic parameters.

The subplots (a)-(c) in Fig. 3-8 highlight the simulation results, illustrating the end-effector's precision in tracking the designated path. The subplots (d) and (e) detail the joint angles and velocities, confirming their cycling back to initial positions, thereby substantiating the SPTC-AN-FOZNN

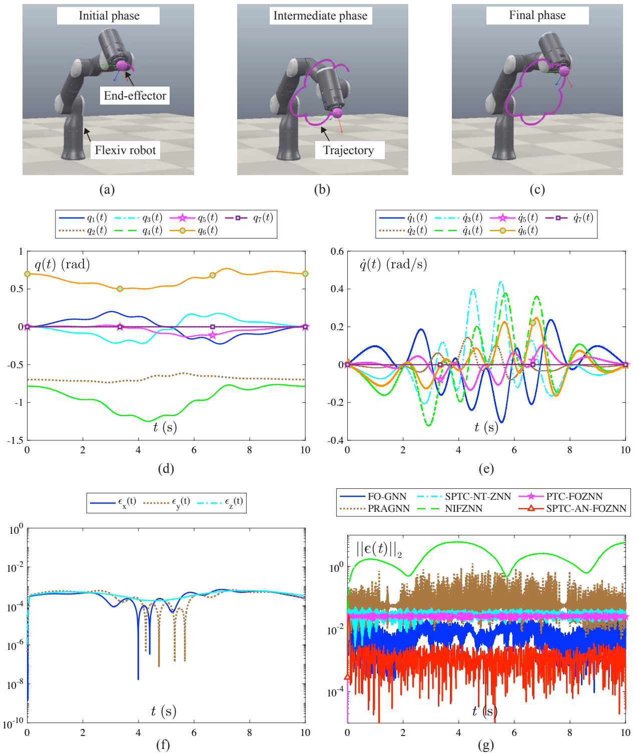


Figure 8: The snapshots of the (a) initial, (b) intermediate, and (c) final phases for a simulated Flexiv Rizon robot during tracking a five-petal-plum-shaped path, with the rendered (d) joint angles, (e) joint velocities, (f) absolute tracking errors by the SPTC-AN-FOZNN model, and (g) residual errors rendered by six different neural models with a bounded random noise $\delta(t) = \cos(t) + \bar{n}(t)$.

model's capability in precise kinematic motion planning. This model effectively maintains all joint movements within specified limits, with tracking errors in three dimensions kept around 10^{-4} meters. Additionally, the subplot (g) in Fig. 3-8 showcases the residual errors $\epsilon(t) = f(y(t), t)$ across six different RNN models, verifying the SPTC-AN-FOZNN's compliance with predefined-time convergence as theorized in Section 3.2. The comparative analysis underlines its superior tracking precision and robustness against additive noises, reinforcing its suitability for complex robotic kinematic control applications.

5.2. *Experimental validation*

To evaluate the performance of the SPTC-AN-FOZNN model, the experimental setup incorporates the Flexiv Rizon robot system, as illustrated in the schematic diagram of Fig. 9. The robot is controlled via a workstation running the Robot Operating System (ROS), which communicates with the control interface through an Ethernet connection. Joint positions (q) and velocities ($\dot{q}$) are measured using transducers, including optical encoders for angular positions of each joint, an inertial measurement unit (IMU) for dynamic joint angular velocity feedback, and an external calibrated camera for additional measurement of the end-effector's tracking position. These sensor readings are treated as the real joint angles rather than the joint angles obtained through resolution methods. The experimental configuration considers environmental factors such as the variability in sensor readings and external disturbances. The noise model employed in the tests consists of random white noise with a sufficiently large upper bound to simulate real-world operational uncertainties. These configurations ensure the robustness of the model in scenarios involving unpredictable system dynamics and environmental disturbances.

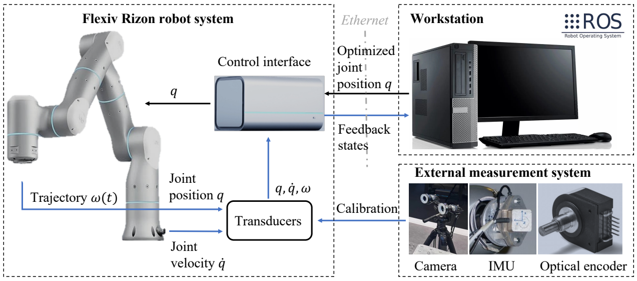


Figure 9: Schematic diagram for the experimental design of the Flexiv Rizon robot's kinematic control system.

The Flexiv Rizon robot is programmed to perform cyclical motions along a five-petal-plum-shaped trajectory, as demonstrated in Fig. 10(a). The measured joint angles and velocities, depicted in Fig. 10(c) and (d), confirm the cyclical nature of the motions, with the robot consistently returning to its initial states. Fig. 10(b) portrays the three-dimensional trajectory of the robot's end-effector relative to the reference path, highlighting the achieved path-tracking accuracy ranging from 10^{-4} and 10^{-3} meters, thus aligning with the order of positional precision ($\sim$ 0.5 mm) calibrated in our lab and claimed in the Flexiv Rizon robot's datasheet. In addition, the joint torques measured during the experiment are presented in Fig. 10(f), indicating stable and smooth motion of the robot, crucial for successful path-tracking. These results collectively affirm the SPTC-AN-FOZNN's effectiveness and its applicability in robotic motion control tasks.

6. Conclusion

This paper develops the SPTC-AN-FOZNN model, tailored for resolving TVQP problems, with a particular emphasis on its deployment in kinematic motion control of robots. Theoretical analysis confirmed that the SPTC-AN-FOZNN model achieves strictly predefined-time convergence and exhibits anti-noise characteristics. Numerical assessments involving an illustrative TVQP example across six distinct RNN models, demonstrated that the SPTC-AN-FOZNN model outperforms other RNNs in terms of convergence precision and robustness. The practicability of the SPTC-AN-FOZNN model is further underscored by its successful application to the kinematic motion control of a Franka Emika Panda robot and a Flexiv Rizon robot.

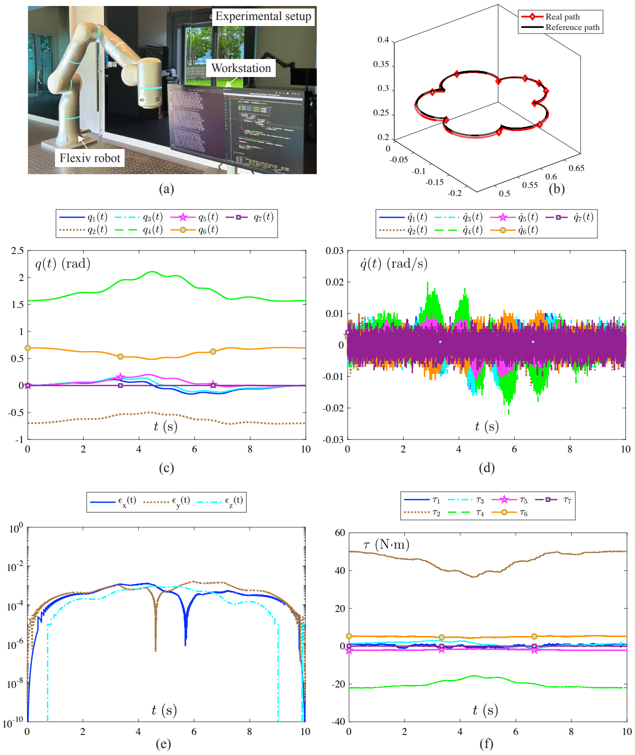


Figure 10: (a) Image of the experimental configuration, (b) comparison of the actual trajectory with the designed five-petal plum-shaped path, and the profiles of (c) joint angles, (d) joint velocities, (e) absolute tracking errors and (f) real-time joint torques produced by the SPTC-AN-FOZNN model on the actual experimental platform.

This marks a seminal demonstration of a strictly predefined-time convergent ZNN model with time-diminishing variable gain. This innovation suggests promising future directions for more energy-efficient ZNN hardware architectures exhibiting strictly user-prescribed-time convergence.

While effective, the current model has limitations that require further exploration. Its reliance on fixed parameters, like α values and noise properties, may restrict its use in varied environments or complex systems. It also assumes predictable and uniform noise, which is often unrealistic in dynamic, real-world scenarios. Future work could focus on adaptive parameter tuning to improve the model's responsiveness to abrupt changes in system dynamics or noise conditions.

Funding

This work was supported by the National Natural Science Foundation of China under Grant 52205032, in part by the Shun Hing Institute of Advanced Engineering, The Chinese University of Hong Kong, and in part by Research Grants Council of Hong Kong (Ref. No. 14204423).

Ethical approval

Not required.

CRediT authorship contribution statement

xx: Conceptualization, Methodology, Investigation, Writing – review & editing. **xx:** Software, Investigation, Writing– original draft. **xx:** Visualization, Supervision, Funding acquisition. **xx:** review. **xx:** Investigation. **xx:** Investigation, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper

References

Becerra, H. M., Vazquez, C. R., Arechavaleta, G., & Delfin, J. (2018). Predefined-time convergence control for high-order integrator systems using time base generators. *IEEE Trans. Control Syst. Technol.*, 26(5), 1866–1873.

Bhat, S. P. & Bernstein, D. S. (2000). Finite-time stability of continuous autonomous systems. *SIAM J. Control Optim.*, 38(3), 751–766.

Boyd, S. & Vandenberghe, L. (2004). *Convex Optimization*. Cambridge, U.K.: Cambridge University Press.

Chen, D., Li, S., Lin, F. J., & Wu, Q. (2020). New super-twisting zeroing neural-dynamics model for tracking control of parallel robots: A finite-time and robust solution. *IEEE Trans. Cybern.*, 50(6), 2651–2660.

Chen, J., Pan, Y., Zhang, Y., Li, S., & Tan, N. (2024). Inverse-free zeroing neural network for time-variant nonlinear optimization with manipulator applications. *Neural Networks*, 178, 106462.

Chong, E. K. P. & Żak, S. H. (2013). *An Introduction to Optimization*. John Wiley & Sons, Ltd, 4th edition.

Effati, S. & Nazemi, A. R. (2006). Neural network models and its application for solving linear and quadratic programming problems. *Appl. Math. Comput.*, 172(1), 305–331.

Haddadin, S., Parusel, S., Johannsmeier, L., Golz, S., Gabl, S., Walch, F., Sabaghian, M., Jahne, C., Hausperger, L., & Haddadin, S. (2022). The Franka Emika Robot: A Reference Platform for Robotics Research and Education. *IEEE Robotics and Automation Magazine*, 29(2), 46–64.

Haykin, S. (1999). *Neural networks: a comprehensive foundation second edition*, volume 13.

Huang, X., Lou, X., & Cui, B. (2016). A novel neural network for solving convex quadratic programming problems subject to equality and inequality constraints. *Neurocomputing*, 214, 23–31.

Jin, L., Li, S., Liao, B., & Zhang, Z. (2017). Zeroing neural networks: A survey. *Neurocomputing*, 267, 597–604.

Jin, L., Li, S., Xiao, L., Lu, R., & Liao, B. (2018). Cooperative motion generation in a distributed network of redundant robot manipulators with noises. *IEEE Trans. Syst. Man, Cybern. Syst.*, 48(10), 1715–1724.

Khalil, R., Al Horani, M., Yousef, A., & Sababheh, M. (2014). A new definition of fractional derivative. *J. Comput. Appl. Math.*, 264, 65–70.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2323.

Li, S., Chen, S., & Liu, B. (2013). Accelerating a recurrent neural network to finite-time convergence for solving time-varying sylvester equation by using a sign-bi-power activation function. *Neural Process. Lett.*, 37(2), 189–205.

Li, W. (2018). A recurrent neural network with explicitly definable convergence time for solving time-variant linear matrix equations. *IEEE Trans. Ind. Informatics*, 14(12), 5289–5298.

Li, W., Guo, C., Ma, X., & Pan, Y. (2024). A strictly predefined-time convergent and noise-tolerant neural model for solving linear equations with robotic applications. *IEEE Trans. Ind. Electron.*, 71(1), 798–809.

Li, W., Liao, B., Xiao, L., & Lu, R. (2019). A recurrent neural network with predefined-time convergence and improved noise tolerance for dynamic matrix square root finding. *Neurocomputing*, 337, 262–273.

Li, W., Ma, X., Luo, J., & Jin, L. (2021). A strictly predefined-time convergent neural solution to equality- and inequality-constrained time-variant quadratic programming. *IEEE Trans. Syst. Man, Cybern. Syst.*, 51(7), 4028–4039.

Li, W., Xiao, L., & Liao, B. (2020). A finite-time convergent and noise-rejection recurrent neural network and its discretization for dynamic nonlinear equations solving. *IEEE Trans. Cybern.*, 50(7), 3195–3207.

Li, Z. & Zhang, Y. (2011). Time-varying quadratic programming by zhang neural network equipped with a time-varying design parameter $\gamma(t)$. In *Advances in Neural Networks*, volume 6675 LNCS of *PART 1* (pp. 101–108).: Springer, Berlin, Heidelberg.

Liao, S., Liu, J., Xiao, X., Fu, D., Wang, G., & Jin, L. (2020). Modified gradient neural networks for solving the time-varying sylvester equation with adaptive coefficients and elimination of matrix inversion. *Neurocomputing*, 379, 1–11.

Liu, M., Wu, H., Shi, Y., & Jin, L. (2023). High-order robust discrete-time neural dynamics for time-varying multilinear tensor equation with m-tensor. *IEEE Trans. Ind. Informatics*, 19(9), 9457–9467.

Lu, H., Jin, L., Luo, X., Liao, B., Guo, D., & Xiao, L. (2019). Rnn for solving perturbed time-varying underdetermined linear system with double bound limits on residual errors and state variables. *IEEE Trans. Ind. Informatics*, 15(11), 5931–5942.

Mandal, P. K. (2023). A review of classical methods and nature-inspired algorithms (nias) for optimization problems. *Results Control Optim.*, 13, 100315.

Murtaza, M. A. & Hutchinson, S. (2022). Consensus in Operational Space for Robotic Manipulators with Task and Input Constraints. In *Proceedings - IEEE International Conference on Robotics and Automation* (pp. 10148–10154).: Institute of Electrical and Electronics Engineers Inc.

Nazemi, A. & Nazemi, M. (2014). A gradient-based neural network method for solving strictly convex quadratic programming problems. *Cognit. Comput.*, 6(3), 484–495.

Polyakov, A. (2012). Nonlinear feedback design for fixed-time stabilization of linear control systems. *IEEE Trans. Automat. Contr.*, 57(8), 2106–2110.

Qi, Y., Jin, L., Luo, X., & Zhou, M. (2022). Recurrent neural dynamics models for perturbed nonstationary quadratic programs: A control-theoretical perspective. *IEEE Trans. Neural Networks Learn. Syst.*, 33(3), 1216–1227.

Rohmer, E., Singh, S. P. N., & Freese, M. (2013). V-rep: A versatile and scalable robot simulation framework. In *IEEE International Conference on Intelligent Robots and Systems* (pp. 1321–1326).

Sanchez-Torres, J. D., Sanchez, E. N., & Loukianov, A. G. (2015). Predefined-time stability of dynamical systems with sliding modes. In *Proceedings of the American Control Conference*, volume 2015-July (pp. 5842–5846).

Xiao, L., Li, L., Tao, J., & Li, W. (2023). A predefined-time and anti-noise varying-parameter znn model for solving time-varying complex stein equations. *Neurocomputing*, 526, 158–168.

Yang, Y., Li, W., Zhou, J., Huang, J., Hu, J., Voyles, R. M., & Ma, X. (2024a). Ptc-foznn: A strictly predefined-time convergent fractional-order recurrent neural network for solving time-variant quadratic programming. In *2024 IEEE International Conference on Control and Automation (ICCA 2024)*.

Yang, Y., Ma, X., Zhang, H. H., & Voyles, R. M. (2024b). Stabilization for a class of partially observable uncertain fractional-order nonlinear systems with time-varying delays and disturbance. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 54(12), 7341–7355.

Yang, Y., Voyles, R. M., Zhang, H. H., & Nawrocki, R. A. (2025). Fractional-order spike-timing-dependent gradient descent for multi-layer spiking neural networks. *Neurocomputing*, 611, 128662.

Yang, Y., Wang, X., Voyles, R. M., & Ma, X. (2024c). A predefined-time convergent and noise-tolerant zeroing neural network model for time variant quadratic programming with application to robot motion planning. *Tsinghua Science and Technology*.

Yang, Y., Zhu, P., Li, W., Voyles, R. M., & Ma, X. (2024d). A fractional-order gradient neural solution to time-variant quadratic programming with application to robot motion planning. *IEEE Transactions on Industrial Electronics*, 71(12), 16579–16589.

Yi, Z. & Tan, K. K. (2004). *Convergence Analysis of Recurrent Neural Networks*, volume 13. Boston, MA: Springer US.

Yu, D., Zhang, G., & Zhang, T. (2024). Predefined-time robust adaptive gradient neural network for solving linear time-varying equations and its applications. *Expert Syst. Appl.*, 255, 124546.

Zhang, Y. & Jia, Y. (2023). Motion planning of redundant dual-arm robots with multicriterion optimization. *IEEE Syst. J.*, 17(3), 4189–4199.

Zhang, Y., Jiang, D., & Wang, J. (2002). A recurrent neural network for solving sylvester equation with time-varying coefficients. *IEEE Trans. Neural Networks*, 13(5), 1053–1063.

Zhang, Y. & Jin, L. (2017). *Robot Manipulator Redundancy Resolution*. Wiley.

Zhang, Y. & Yi, C. (2011). *Zhang Neural Networks and Neural-Dynamic Method*. Nova Science Pub Inc; UK ed. edition.