An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.15, No.4, pp.728-737 (2025)
https://doi.org/10.36922/IJOCTA025180091

RESEARCH ARTICLE

Model reference adaptive control based time delay estimation with RBF neural network for robot manipulators

Saim Ahmed[1,2*], Ahmad Taher Azar[1,2], and Ibraheem Kasim Ibraheem[3,4]

[1]College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia
[2]Automated Systems and Computing Lab (ASCL), Prince Sultan University, Riyadh, Saudi Arabia
[3]Department of Electrical Engineering, College of Engineering, University of Baghdad, Baghdad, Iraq
[4]Department of Electronics and Communication Engineering, College of Engineering, Uruk University, Baghdad, Iraq
sahmed@psu.edu.sa, aazar@psu.edu.sa, ibraheemki@coeng.uobaghdad.edu.iq

ARTICLE INFO

Article History:
Received: May 1, 2025
Revised: July 4, 2025
Accepted: July 18, 2025
Published Online: September 4, 2025

Keywords:
Model reference adaptive control,
Time delay estimation
Neural network
Robotic manipulator

AMS Classification 2010:
93D05; 93C95; 93C10; 70E60

ABSTRACT

In this paper, model-reference adaptive control (MRAC) with neural network (NN) and time delay estimation (TDE) is proposed for controlling a robotic manipulator. With more than two degrees of freedom (DoF) of the robot, the formulation of a known regression matrix is tedious and also difficult to compute for the different robotic systems. Therefore, this work introduces MRAC based on TDE with NN (MRAC-NNTDE) to achieve high-control performance without prior knowledge of the regression matrix and offers a model-free scheme. Firstly, MRAC is applied to adjust the control gains, then TDE is implemented to estimate the unknown dynamical robotic system, and NN is employed to deal with the TDE estimation error. The overall stability of the robotic dynamics is investigated using the Lyapunov theorem. In the end, computer simulations are compared to validate the effectiveness of the proposed scheme.

1. Introduction

One of the most well-known control techniques in the field of control science and engineering is model reference adaptive control (MRAC). Depending on the current state of the closed-loop system, MRAC either estimates the unknown parameters or updates the control gain.[1,2] Both linear and nonlinear systems have been widely controlled using this control technique.[3] The adaptive scheme has been employed with well-known advanced control techniques, for instance, H_∞ control, sliding mode control (SMC), proportional-integral-derivative control (PID), neural network (NN), and fuzzy logic control schemes, etc.[4–9] In addition, various MRAC schemes have been designed to improve joint position tracking performance.[10,11]

Numerous robotics technologies make use of control system analysis and design.[12–14] In which, MRAC has been widely used in robotic and aviation applications throughout the past few decades.[15] Furthermore, in the presence of nonlinearities and outside disturbances, it is utilized to precisely track the joint position of uncertain robotic manipulators.[3] Because it requires a known regression matrix to deal with the unknown dynamics of robotic systems,[16] it is quite difficult to calculate this known matrix for robotic manipulators with high degrees of freedom (DOF). To address this challenge, researchers have proposed diverse adaptive control

*Corresponding Author

methodologies that do not necessitate a known regression matrix. One such approach involves the utilization of neural networks to approximate the unknown dynamics of robotic systems.[17] These neural network-based adaptive controllers have demonstrated promising outcomes in managing complex, high-degree-of-freedom robotic manipulators.[18] However, the implementation of these advanced control techniques frequently requires substantial computational resources and meticulous tuning of hyperparameters.[19]

In the literature, various notable model-free control types of research have been proposed using the time delay estimation (TDE) scheme.[20] The TDE approach has offered an appropriate and highly efficient estimation method for unknown uncertain dynamical systems with external disturbances and provides an easy implementation of the model-free approach.[21]

Thanks to the TDE scheme, model-free control can be developed. The general TDE concept is to estimate the unknown dynamical systems by inserting a constant delay.[22] Moreover, TDE has been extensively integrated with numerous controllers, such as sliding mode control (SMC), adaptive control, fuzzy control, NN, and PID control schemes, to obtain accurate estimation and robustness precisely.[23–27] As stated, TDE provides robust estimation and offers satisfactory performance, accurately and efficiently in terms of joint position tracking, better convergence, and small steady-state error.

Control development in the context of TDE with adaptive control has shown great interest by researchers in exploiting the advantages of both techniques.[28,29] Therefore, several model-independent control techniques related to TDE with adaptive control have been proposed for robot manipulators where the unknown dynamics are estimated using TDE, and the estimation error is dealt with using adaptive gain.[30,31] To name a few, bounded nonlinearities have been taken into consideration; Jin et al. proposed a controller that combines TDE and adaptive control using ideal velocity feedback to drive the robot manipulators toward their desired trajectory.[32] Aiming to control the time-varying dynamics of robotic manipulators, an SMC-based TDE was given to control time-varying dynamics to observe the desired trajectory and knowledge of unknown dynamics known by TDE. In,[29] fault-tolerant control using TDE and nonsingular fast terminal SMC (NFTSMC) has been proposed to estimate the unknown dynamics under actuator faults, and robustness and fast convergence have been obtained by NFTSMC. One of the main factors contributing to TDE's prominence is its ease of use and ability to estimate the dynamics of unknown systems. Thus, TDE is a good substitute for model-based control approaches.

The following are the primary contributions of this work:

- Using MRAC and TDE, a model-free control is designed to avoid the regression matrix and provide robust, accurate, and exact tracking.
- The unknown dynamics of the robotic manipulator are estimated using TDE, and the state gains are updated using MRAC, and the TDE error is compensated using NN.
- The Lyapunov stability criterion is used to establish and carry out the closed-loop system's overall asymptotic stability analysis.
- For uncertain robotic manipulators with unknown external disturbances, MRAC using NNTDE is a novel approach.

The other sections are arranged as follows. In Section 2, the preliminary is presented. Section 3 presents the design of the proposed MRAC-NNTDE method and the stability analysis of the overall system. To show the efficacy of the proposed method, simulation results are depicted in Section 4, and their discussion is given in Section 5. Finally, this work concludes in Section 6.

2. Preliminaries

2.1. Definition 1[33]

The perturbed dynamical system with function $\mathfrak{z}(t)$ is defined as

$$\dot{\mathfrak{z}}(t) = \mathfrak{g}(\mathfrak{z}) + \mathfrak{h}(\mathfrak{z})\mathfrak{u}(t) + \mathfrak{p}(t). \tag{1}$$

where $\mathfrak{g}(\mathfrak{z})$ is the unknown function, $\mathfrak{h}(\mathfrak{z})$ is a distribution matrix, $\mathfrak{p}(t)$ shows unknown external disturbance, and $\mathfrak{u}(t)$ represents the control input. When known and unknown functions are separated, Equation (1) can be expressed as

$$\psi(\mathfrak{z}, t) = \mathfrak{g}(\mathfrak{z}) + \mathfrak{p}(t) = \dot{\mathfrak{z}}(t) - \mathfrak{h}(\mathfrak{z})\mathfrak{u}(t). \tag{2}$$

One can calculate the TDE estimation of unknown dynamics as

$$\begin{aligned} \hat{\psi}(\mathfrak{z}, t) &\triangleq \hat{\mathfrak{g}}(\mathfrak{z}) + \hat{\mathfrak{p}}(t) \triangleq \mathfrak{g}(\mathfrak{z})_{(t-\mathfrak{d})} + \mathfrak{p}_{(t-\mathfrak{d})} \\ &= \dot{\mathfrak{z}}_{(t-\mathfrak{d})} - \mathfrak{h}(\mathfrak{z})_{(t-\mathfrak{d})}\mathfrak{u}_{(t-\mathfrak{d})}. \end{aligned} \tag{3}$$

where $\mathfrak{d}$ used as constant time delay and $\hat{\psi}(\mathfrak{z}, t)$ denoted the estimated unknown dynamics.

3. Control design

This section provides the construction of the suggested model-free control scheme for the uncertain robotic manipulator with external disturbance utilizing the MRAC integrated TDE and NN technique. Then, the overall system stability is established using the Lyapunov analysis.

The following equation describes the dynamics of n-DOF robotic manipulator[34]

$$\mathcal{M}(q)\ddot{q} + \mathcal{V}(q,\dot{q})\dot{q} + \mathcal{G}(q) + \mathcal{D}(t) = \mathcal{T}(t). \quad (4)$$

where $q, \dot{q}$ and $\ddot{q} \in \Re^m$ represent the vectors of the joint's position, velocity, and acceleration, respectively. $\mathcal{M}(q) \in \Re^{m\times m}$ is the symmetric and positive definite inertia matrix, $\mathcal{V}(q,\dot{q}) \in \Re^{m\times m}$ is coriolis/centripetal, $\mathcal{G}(q) \in \Re^m$ is gravitational force, the control torque is denoted by $\mathcal{T}(t) \in \Re^m$, and external disturbances are represented by $\mathcal{D}(t) \in \Re^m$.

3.1. MRAC with TDE scheme

This subsection presents the design of MRAC-TDE. To represent in the state-space form, the dynamics of the robot manipulator (4) can be expressed as follows:

$$\bar{\mathcal{M}}\ddot{q} - \bar{\mathcal{M}}\ddot{q} + \mathcal{M}(q)\ddot{q} + \mathcal{V}(q,\dot{q})\dot{q} + \mathcal{G}(q) + \mathcal{D}(t) = \mathcal{T}(t). \quad (5)$$

$$\Rightarrow \ddot{q} + \chi(q,\dot{q},\ddot{q}) = \bar{\mathcal{M}}^{-1}\mathcal{T}(t). \quad (6)$$

where $\chi(q,\dot{q},\ddot{q}) = \bar{\mathcal{M}}^{-1}((\mathcal{M}(q) - \bar{\mathcal{M}})\ddot{q} + \mathcal{V}(q,\dot{q})\dot{q} + \mathcal{G}(q) + \mathcal{D}(t))$ is the unknown uncertain dynamics and external disturbance, and $\bar{\mathcal{M}} > 0$ is diagonal matrix.

In order to develop the MRAC-based TDE method, Equation (6) can be expressed in the form of state space:

$$\dot{x} = Ax + B\left[\bar{\mathcal{M}}^{-1}\mathcal{T}(t) - \chi(x,\dot{x})\right]. \quad (7)$$

where $x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}_{n\times 1}$, $A = \begin{bmatrix} 0 & \mathcal{I}_{m\times m} \\ 0 & 0 \end{bmatrix}_{n\times n}$, $B = \begin{bmatrix} 0 \\ \mathcal{I}_{m\times m} \end{bmatrix}_{n\times m}$, and $\mathcal{I}$ is identity matrix.

Therefore, the control design of MRAC-TDE is given as follows:

$$\mathcal{T}(t) = \bar{\mathcal{M}}(\hat{\vartheta}^T x + \hat{\upsilon}^T r + \hat{\chi}). \quad (8)$$

where $\hat{\vartheta}$, $\hat{\upsilon}$, and $\hat{\chi}$ are adaptive gains, $\hat{\chi} \triangleq \chi(x(t-d), \dot{x}(t-d)) = \bar{\mathcal{M}}^{-1}\mathcal{T}(t-d) - \ddot{q}(t-d)$ is the estimation of $\hat{\chi}$ using the delayed value of Equation (6), and d is the constant delay.

The dynamics reference model for MRAC is given by,

$$\dot{x}_p = A_p x_p + B_p r. \quad (9)$$

with $A_p = A + B\vartheta^T$ is Hurwitz matrix, $B_p = B\upsilon^T$, $x_p = \begin{bmatrix} q_d, \dot{q}_d \end{bmatrix}^T$ is the reference input and $r \in \Re^m$ is the desired reference. Moreover, $\vartheta \in \Re^{n\times m}$ and $\upsilon \in \Re^{m\times m}$ are the unknown gains.

$$\dot{x} = Ax + B\left[(\hat{\vartheta}^T x + \hat{\upsilon}^T r) + \Delta\xi\right]. \quad (10)$$

where $\Delta\xi = \hat{\chi} - \chi(x,\dot{x})$, $\|\Delta\xi\| \le \beta$ and $\beta > 0$. Equation (10) can be expressed as

$$\dot{x} = (A + B\hat{\vartheta}^T)x + B\left[\hat{\upsilon}^T r + \Delta\xi\right]. \quad (11)$$

The dynamics of tracking error $e = x - x_p$, $\dot{e} = \dot{x} - \dot{x}_p$ using Equations (9) and (11) can be computed as

$$\dot{e} = \left[A + B\hat{\vartheta}^T\right]x + B\hat{\upsilon}^T r + B\Delta\xi - A_p x_p - B_p r \pm A_p x. \quad (12)$$

Equation (12) can be expressed as

$$\dot{e} = A_p e + \left[A + B\hat{\vartheta}^T - A_p\right]x + B\hat{\upsilon}^T r + B\Delta\xi - B_p r. \quad (13)$$

Using A_p and B_p, Equation (13) can be written as

$$\dot{e} = A_p e + B(\hat{\vartheta}^T - \vartheta^T)x + B(\hat{\upsilon}^T - \upsilon^T)r + B\Delta\xi. \quad (14)$$

Equation (14) can be represented as

$$\dot{e} = A_p e + B\tilde{\vartheta}^T x + B\tilde{\upsilon}^T r + B\Delta\xi. \quad (15)$$

where $\tilde{\vartheta} = \hat{\vartheta} - \vartheta$, $\tilde{\upsilon} = \hat{\upsilon} - \upsilon$.

To investigate the stability of the closed-loop model, the Lyapunov theorem is applied; thus, the Lyapunov function is selected as

$$\begin{aligned} V(e,\tilde{\vartheta},\tilde{\upsilon}) &= e^T P e + trace\left(\tilde{\vartheta}^T \varphi_1^{-1}\tilde{\vartheta}\right) \\ &+ trace\left(\tilde{\upsilon}^T \varphi_2^{-1}\tilde{\upsilon}\right). \end{aligned} \quad (16)$$

where $\varphi_1^T = \varphi_1 > 0 \in \Re^{n\times n}$, $\varphi_2^T = \varphi_2 > 0 \in \Re^{m\times m}$, $P^T = P > 0 \in \Re^{n\times n}$ are symmetric and +ve definite matrices, and *trace* represents the diagonal elements sum.

By taking a derivative of Equation (16) and then substituting to Equation (15), one gets

$$\begin{aligned} \dot{V}(e,\tilde{\vartheta},\tilde{\upsilon}) &= e^T(PA_p + A_p^T P)e \\ &+ 2e^T PB\tilde{\vartheta}^T x + trace\left(2\tilde{\vartheta}^T \varphi_1^{-1}\dot{\hat{\vartheta}}\right) \\ &+ 2e^T PB\tilde{\upsilon}^T r + trace\left(2\tilde{\upsilon}^T \varphi_2^{-1}\dot{\hat{\upsilon}}\right) \\ &+ 2e^T PB\Delta\xi. \end{aligned} \quad (17)$$

Equation (17) can be expressed as

$$\begin{aligned} \dot{V}(e,\tilde{\vartheta},\tilde{\upsilon}) &= -e^T Q e \\ &+ 2trace\left(\tilde{\vartheta}^T(xe^T PB + \varphi_1^{-1}\dot{\hat{\vartheta}})\right) \\ &+ 2trace\left(\tilde{\upsilon}^T(re^T PB + \varphi_2^{-1}\dot{\hat{\upsilon}})\right) \\ &+ 2e^T PB\Delta\xi. \end{aligned} \quad (18)$$

Thus, the adaptive parameters $\hat{\vartheta}$ and $\hat{\upsilon}$ can be computed as

$$\begin{aligned} \dot{\hat{\vartheta}} &= -\varphi_1 xe^T PB, \\ \dot{\hat{\upsilon}} &= -\varphi_2 re^T PB. \end{aligned} \quad (19)$$

By substituting Equation (19) into Equation (18), one obtains

$$\dot{V}(e, \tilde{\vartheta}, \tilde{\upsilon}) = -e^T(Q)e + 2e^T PB\Delta\xi \tag{20}$$

Since P is a matrix that is symmetric +ve definite and A_p is Hurwitz; thus, we get the Lyapunov equation $PA_p + A_p^T P = -Q$ and $Q > 0$ is also a +ve definite symmetric matrix.

To find the asymptotic stability of the MRAC-TDE approach, one can get the following equation:

$$\begin{aligned} \dot{V}(e, \tilde{\vartheta}, \tilde{\upsilon}) &= -e^T(Q)e + 2e^T PB\Delta\xi \\ &\leq -\lambda_m(Q)\|e\|^2 + 2\|e\|\,\|PB\|\,\|\Delta\xi\|. \end{aligned} \tag{21}$$

$$\|e\| \leq \frac{2\|PB\|\,\|\Delta\xi\|}{\lambda_m(Q)}. \tag{22}$$

$$\Rightarrow \dot{V}(e, \tilde{\vartheta}, \tilde{\upsilon}) < 0.$$

Hence, the closed MRAC-TDE system is asymptotically stable.

3.2. MRAC-NNTDE control design

In this section, MRAC-TDE with a neural network (NN) has been described. Here, NN is used to estimate the TDE estimation error to obtain precise and robust tracking performance.

To design the MRAC-NNTDE proposed scheme, we utilize Equation (7) given by

$$\dot{x} = Ax + B\left[\bar{\mathcal{M}}^{-1}\mathcal{T}(t) - \chi(x, \dot{x})\right]. \tag{23}$$

The proposed control input is developed as

$$\mathcal{T}(t) = \bar{\mathcal{M}}\left[\hat{\vartheta}^T x + \hat{\upsilon}^T r + \hat{\chi} - Ke + \mathcal{T}(t)_{nn}\right]. \tag{24}$$

where $K \in \Re^{m\times n}$.

$$\dot{x} = Ax + B\left[\begin{array}{l} \hat{\vartheta}^T x + \hat{\upsilon}^T r + \hat{\chi} - Ke - \chi(x, \dot{x}) \\ +\mathcal{T}(t)_{nn} \end{array}\right]. \tag{25}$$

The above equation can be expressed as

$$\dot{x} = Ax + B\left[\hat{\vartheta}^T x + \hat{\upsilon}^T r + \Delta\xi - Ke + \mathcal{T}(t)_{nn}\right]. \tag{26}$$

where we consider

$$\Delta\xi = W^{*T} h(e) + \varepsilon \tag{27}$$

with $h_j(e_i) = exp\left(-\frac{\|e_i - c_j\|^2}{2b_j^2}\right)$, $h = [h_1,\, h_2,\, \cdots\, h_n]^T$, and ε is the approximation error.

Moreover, the NN approximation law can be computed as

$$\begin{aligned} \mathcal{T}(t)_{nn} &= \Delta\hat{\xi}(t) = -\hat{W}^T h(e), \\ \dot{\hat{W}} &= \varphi_3 h(e) e^T PB. \end{aligned} \tag{28}$$

By substituting $\mathcal{T}(t)_{nn}$ from Equation(28) and $\Delta\xi$ from Equation (27) into Equation (26), one gets

$$\dot{x} = Ax + B\left[\begin{array}{l} \hat{\vartheta}^T x + \hat{\upsilon}^T r + W^{*T} h(e) + \varepsilon \\ -\hat{W}^T h(e) - Ke \end{array}\right]. \tag{29}$$

Equation (29) can be written as

$$\dot{x} = Ax + B\left[\hat{\vartheta}^T x + \hat{\upsilon}^T r - \tilde{W}^T h(e) + \varepsilon - Ke\right]. \tag{30}$$

where $\tilde{W}^T = \hat{W}^T - W^{*T}$.

The tracking error dynamics $\dot{e} = \dot{x} - \dot{x}_p$ using Equations (9) and (30) is defined as

$$\begin{aligned} \dot{e} = &\left[A + B\hat{\vartheta}^T\right] x + B\hat{\upsilon}^T r - B\tilde{W}^T h(e) - BKe \\ &+ B\varepsilon - A_p x_p - B_p r \pm A_p x. \end{aligned} \tag{31}$$

When one solves Equation (31), it yields

$$\dot{e} = A_p e + B\tilde{\vartheta}^T x + B\tilde{\upsilon}^T r - B\tilde{W}^T h(e) - BKe + B\varepsilon. \tag{32}$$

where $\tilde{\vartheta} = \hat{\vartheta} - \vartheta$ and $\tilde{\upsilon} = \hat{\upsilon} - \upsilon$ are the adaptive errors, and the ϑ and υ are the unknown gains. In Figures 1 and 2, the comprehensive diagram of the suggested scheme using MRAC, TDE, neural network, and a robotic system is depicted, and the neural network architecture is given, respectively.

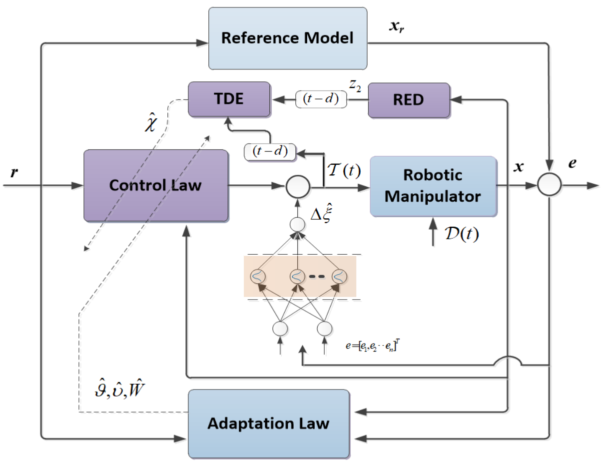

Figure 1. Control input under uncertain dynamics

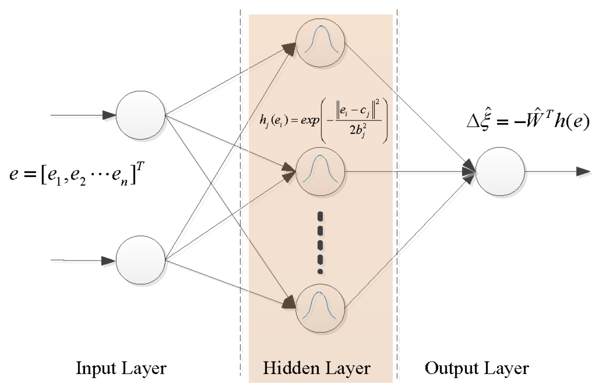

Figure 2. Control input under uncertain dynamics

Remark 1 To successfully implement the designed controller, we can effectively work with the obtainable joint position, even without known velocity and acceleration. We can confidently estimate velocity and acceleration by employing the robust observer, ensuring optimal performance. The following are the estimates of $\dot{q}$ and $\ddot{q}$ [35]

$$\begin{cases} \frac{dz_0}{dt} = z_1 + \varphi_0, \\ \frac{dz_1}{dt} = z_2 + \varphi_1, \\ \frac{dz_2}{dt} = \varphi_2. \end{cases} \tag{33}$$

where $z_0 = \hat{q}$, $z_1 = \dot{\hat{q}}$, $z_2 = \ddot{\hat{q}}$, $\alpha > 0$ and $\varphi_j(e) = \alpha|e|^{\frac{2-j}{3}} sgn(e)$ when $j = 0, 1, 2$.

3.3. Stability investigation

This subsection establishes the overall stability of the suggested MRAC-NNTDE scheme by applying the Lyapunov theorem. Thus, the Lyapunov candidate is chosen as

$$\begin{aligned} V(e, \tilde{\vartheta}, \tilde{\upsilon}, \tilde{W}) = e^T Pe + trace\left(\tilde{\vartheta}^T \varphi_1^{-1} \tilde{\vartheta}\right) \\ + trace\left(\tilde{\upsilon}^T \varphi_2^{-1} \tilde{\upsilon}\right) + trace\left(\tilde{W}^T \varphi_3^{-1} \tilde{W}\right). \end{aligned} \tag{34}$$

where $\varphi_3^T = \varphi_3 > 0 \in \Re$.

$$\begin{aligned} &\dot{V}(e, \tilde{\vartheta}, \tilde{\upsilon}, \tilde{W}) = e^T (PA_p + A_p^T P)e \\ &-2e^T PBKe + 2e^T PB\varepsilon \\ &+2e^T PB\tilde{\vartheta}^T x + trace\left(2\tilde{\vartheta}^T \varphi_1^{-1} \dot{\tilde{\vartheta}}\right) \\ &+2e^T PB\tilde{\upsilon}^T r + trace\left(2\tilde{\upsilon}^T \varphi_2^{-1} \dot{\tilde{\upsilon}}\right) \\ &-2e^T PB\tilde{W}^T h(e) + trace\left(2\tilde{W}^T \varphi_3^{-1} \dot{\tilde{W}}\right). \end{aligned} \tag{35}$$

Then, (35) can be written as

$$\begin{aligned} &\dot{V}(e, \tilde{\vartheta}, \tilde{\upsilon}, \tilde{W}) = -e^T Qe - 2e^T PBKe + 2e^T PB\varepsilon \\ &+2trace\left(\tilde{\vartheta}^T (xe^T PB + \varphi_1^{-1} \dot{\tilde{\vartheta}})\right) \\ &+2trace\left(\tilde{\upsilon}^T (re^T PB + \varphi_2^{-1} \dot{\tilde{\upsilon}})\right) \\ &+2trace\left(\tilde{W}^T (-h(e)e^T PB + \varphi_3^{-1} \dot{\tilde{W}})\right). \end{aligned} \tag{36}$$

Therefore, using adaptive laws given in Equations (19) and (28), one can compute

$$\dot{V}(e, \tilde{\vartheta}, \tilde{\upsilon}, \tilde{W}) \leq -e^T Qe - 2e^T PBKe + 2e^T PB\varepsilon. \tag{37}$$

As $2e^T PB(\varepsilon - Ke)$ with $\varepsilon \leq Ke$, we get

$$\dot{V}(e, \tilde{\vartheta}, \tilde{\upsilon}, \tilde{W}) \leq -e^T Qe. \tag{38}$$

The Lyapunov function $V(e, \tilde{\vartheta}, \tilde{\upsilon}, \tilde{W})$ is positive definite, and the derivative would be $\dot{V}(e, \tilde{\vartheta}, \tilde{\upsilon}, \tilde{W}) \leq 0$. Thus, it is concluded that $\tilde{\vartheta}$, $\tilde{\upsilon}$ and $\tilde{W}$ are all bounded according to Barbalat's Lemma [36]. Taking the integration of Equation (38) from 0 to ∞, we have

$$\begin{aligned} &Q \int_0^\infty e(\sigma)^T e(\sigma) d\sigma \\ &\leq -\int_0^\infty \dot{V}(e(\sigma), \tilde{\vartheta}(\sigma), \tilde{\upsilon}(\sigma), \tilde{W}(\sigma)) d\sigma \\ &= V_0 - V_\infty < \infty. \end{aligned} \tag{39}$$

The bounded function V is non-increasing with bounded initial value V_0; therefore, it may be stated that $\lim_{t\to\infty} \int_0^t e(\sigma)^T e(\sigma) d\sigma$ and $\dot{e}$ is bounded. Following this, the tracking error asymptotically converges to zero.

4. Numerical results

In order to validate the suggested model-free scheme based on MRAC, neural network, and TDE (MRAC-NNTDE), this section first looks at the robotic manipulator's uncertain 2-DOF dynamics under unknown external disturbances. Subsequently, computer simulations are used and compared with adaptive fixed-time sliding mode control (AfTSMC) [37] to show how effective the suggested scheme is in validating the MRAC-TDE program.

4.1. 2-DOF robotic manipulator dynamics

In order to assess the efficacy of the MRAC-NNTDE control method, this study examines the dynamic behavior of a 2-DOF robotic manipulator under uncertainty and external disturbances.[38] $\mathcal{M}(q)$, $\mathcal{V}(q, \dot{q})$, $\mathcal{G}(q)$, and $\mathcal{D}(q, \dot{q})$ are provided in the following manner:

$$\mathcal{M}(q) = \begin{bmatrix} \mathcal{M}(q)_{11} & \mathcal{M}(q)_{12} \\ \mathcal{M}(q)_{21} & \mathcal{M}(q)_{22} \end{bmatrix},$$

with $\mathcal{M}(q)_{11} = l_2^2 m_2 + 2l_1 l_2 m_2 c_2 + l_1^2 (m_1 + m_2) + I_1$,
$\mathcal{M}(q)_{12} = \mathcal{M}(q)_{21} = l_2^2 m_2 + l_1 l_2 m_2 c_2$,
$\mathcal{M}(q)_{22} = l_2^2 m_2 + I_2$,

$$\mathcal{V}(q, \dot{q}) = \begin{bmatrix} -m_2 l_1 l_2 s_2 \dot{q}_2^2 - 2m_2 l_1 l_2 s_2 \dot{q}_1 \dot{q}_2 \\ m_2 l_1 l_2 s_2 \dot{q}_2^2 \end{bmatrix},$$

$$\mathcal{G}(q) = \begin{bmatrix} m_2 l_2 g c_{12} + (m_1 + m_2) l_1 g c_1 \\ m_2 l_2 g c_{12} \end{bmatrix}.$$

where $c_i = cos(q_i)$, $s_i = sin(q_i)$, $c_{ij} = cos(q_i + q_j)$ and $g = 0.98$.

While $\mathcal{D}(q, \dot{q}) = \begin{bmatrix} 0.5\dot{q}_1 + sin(1.5q_1) \\ 1.3\dot{q}_2 - 1.8sin(2q_2) \end{bmatrix}$.

The 2-DOF robot manipulator diagram is given in Figure 3. The suitable values of the parameters l_i, m_i, and I_i are listed in Table 1.

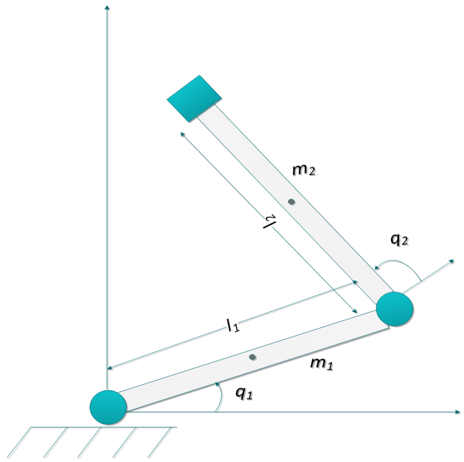

Figure 3. 2-DOF Robotic manipulator

Table 1. Proposed control parameters.

	$link_1$	$link_2$
$length l_i(m)$	1	1
$mass m_i(kg)$	0.5	0.5
$inertia I_i(kg.m^2)$	5	5

4.2. MRAC-NNTDE application on 2-DOF robot manipulator

The following appropriate values are chosen for the proposed control (24) and the dynamics model (4) in order to guarantee high-performance tracking using our suggested model-free control approach: $r_1(t) = r_2(t) = step$, $\mathcal{D}(t) = [sin(t) + 0.5\dot{q}_1 + sin(1.5q_1), sin(t) + 1.3\dot{q}_2 + 1.8sin(2q_2)]^T$, $A_p = diag(-10, -10, -10, -10)$, $B_p = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T$, $K = 10^5 \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$, $\bar{\mathcal{M}} = diag(0.0001, 0.0001)$, $d = 0.001$, $x_{p1}(0) = x_{p2}(0) = -0.1$. The initial conditions and adaptation law parameters (19,28) are also chosen as $\varphi_1 = 0.01 \times diag(1, 1, 1, 1)$, $\varphi_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\varphi_3 = 5$, $\hat{\vartheta}(0) = \hat{\upsilon}(0) = 1$ and $\hat{W}(0) = 0$. The initial values of joints are given as $q_1(0) = 1.5$ and $q_2(0) = 2.1$. These parameter selections are crucial for achieving the desired tracking performance in the model-free control technique.

MRAC-NNTDE is evaluated on the uncertain dynamics of a 2-DOF robotic manipulator under disturbances to demonstrate the effectiveness of the suggested approach. As a result, their comparison of the simulation results is shown in Figures 4-7, respectively, for position tracking, tracking error, control inputs, and TDE estimation of unknown dynamics. Moreover, the adaptive gains to compensate for the unknown dynamics are given in Figures 8-10.

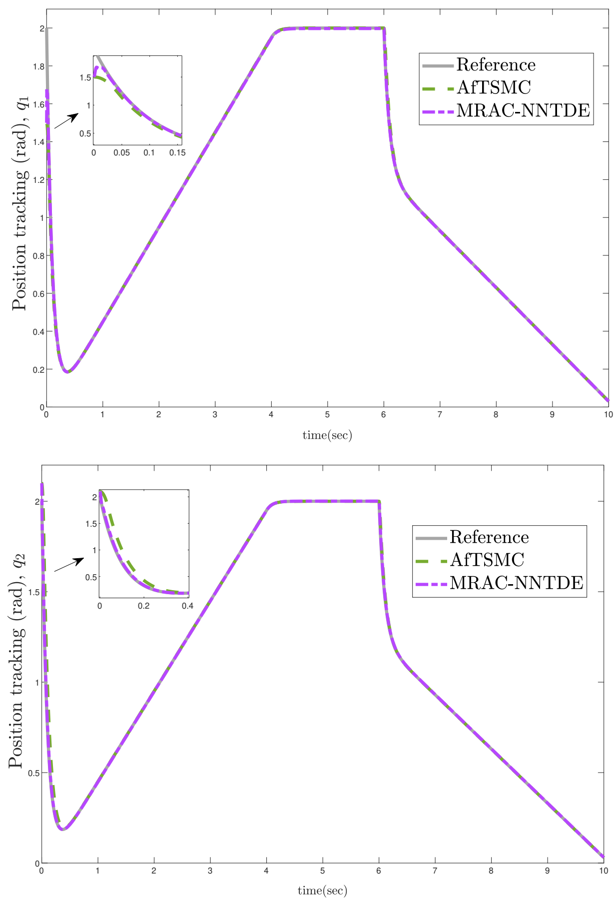

Figure 4. Position tracking

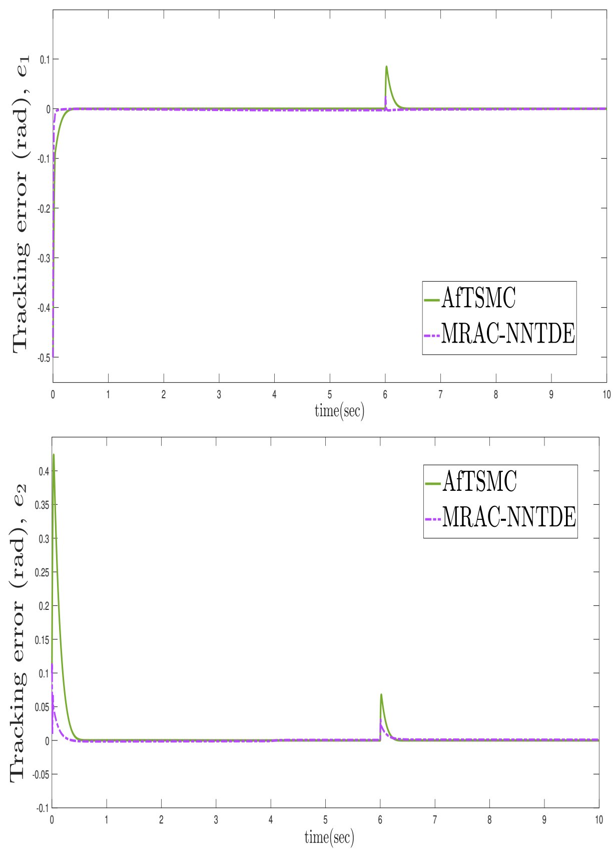

Figure 5. Tracking error

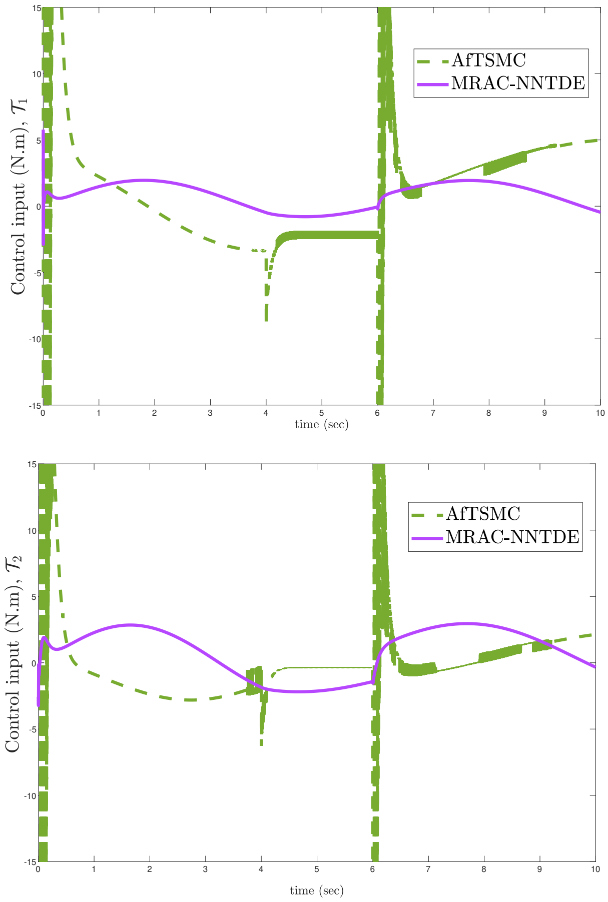

Figure 6. Control torque input

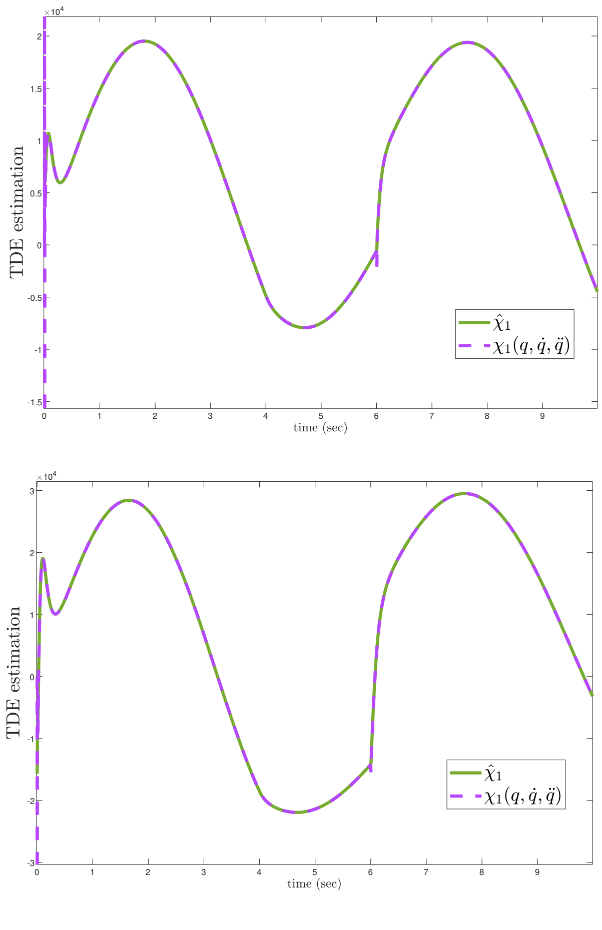

Figure 7. Estimation using TDE

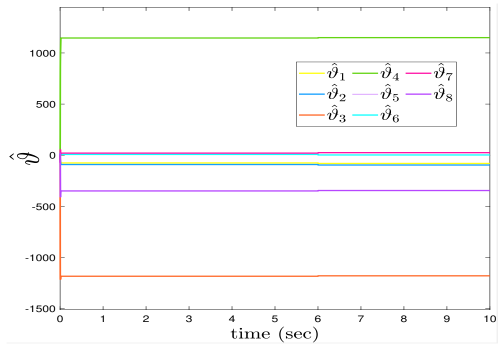

Figure 8. Adaptive gain

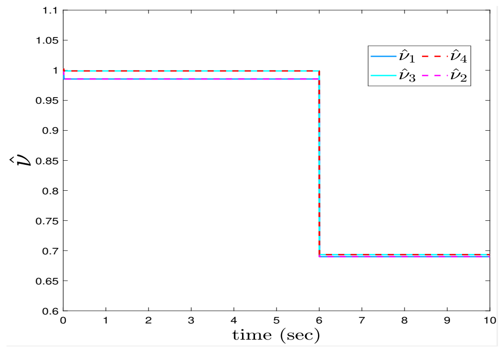

Figure 9. Adaptive gain

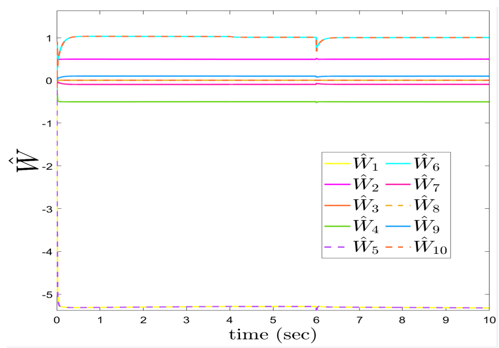

Figure 10. Adaptive gain

The results clearly depict the enhanced performance of MRAC-NNTDE in addressing uncertain dynamics and disturbances. Figures 4-5 illustrate the precise position tracking and errors achieved by the designed method, while Figures 6 and 7 depict the efficient control inputs and precise TDE estimation of unknown dynamics, respectively. The adaptability of the method is also clear from Figures 8 to 10, demonstrating the adaptation of adaptive gains utilized to overcome the unknown dynamics during simulation. It is clear from the acquired simulation results that computer simulations guarantee the overall stability of the system with adequate performance in the face of external disturbances. Furthermore, without the need for a regression matrix, MRAC-NNTDE achieves a rapid convergence speed and extremely robust tracking performances for unknown and unpredictable dynamics of robotic manipulators. The proposed MRAC-NNTDE approach demonstrates superior performance in handling uncertainties and disturbances in robotic manipulator control. The capability of the proposed method in rapid convergence, as well as good tracking with no need for a regression matrix, shows its efficiency and flexibility. These findings support that MRAC-NNTDE is a promising approach for controlling complex robotic systems in realistic applications where uncertain dynamics and external disturbances prevail.

Remark 2 The proposed MRAC scheme adaptively adjusts the control gains, TDE estimates the unknown system dynamics, and a neural network compensates for the estimation error. Compared to the existing control scheme, this approach provides superior performance in handling nonlinearities and uncertainties, resulting in enhanced tracking and robustness.

5. Discussion

The proposed MRAC-NNTDE technique is intended to control the nonlinear dynamical robotic systems under external disturbances. The goal of this study is to increase tracking and transient features while retaining the robustness of a closed-loop system. The MRAC-NNTDE approach aims to achieve quick convergence, and the Lyapunov technique was utilized to demonstrate stability.

The simulation results demonstrate that the MRAC-NNTDE technique efficiently handles the dynamics of a robotic manipulator with external disturbances. Figures 3-4 exhibit the variables q_1, q_2, and the tracking errors e_1, e_2, indicating the successful control performance and the capacity of the proposed system to reduce the tracking error to zero quickly. The control input in Figure 5 has desirable characteristics such as smoothness and satisfactory tracking performance, efficiently suppressing the effects of external disturbances. The benefits of the MRAC-NNTDE approach, including improved responsiveness, reduced tracking error, and enhanced control of nonlinear dynamics, are emphasized by the simulation that graphically demonstrates and confirms the theoretical analysis.

Constraints of the given controller parameters and stability proofs are discussed in this study. In order to achieve the stability of the overall system and convergence of error in a specified time, the appropriate parameters for the MRAC-NNTDE technique were selected. The study allows for easier selection of acceptable values and speeds up convergence by emphasizing the necessity of choosing the right value of parameters to obtain stability and convergence.

In summary, the proposed MRAC-NNTDE approach possesses great promise for improving the tracking quality and resilience of nonlinear systems, particularly against external disturbances. With wide relevance to numerous control engineering problems, the paper provides original contributions to the design and application of the MRAC-NNTDE method.

6. Conclusion

For the uncertain robot manipulator's unknown dynamics under unknown external disturbances, the proposed MRAC-NNTDE scheme is introduced. TDE replaces the known regression matrix and steady-state error to estimate unknown dynamics, resulting in model-free control and high-tracking performance in terms of convergence. Simulation results of the proposed scheme, which is applied to a 2-DOF robotic manipulator, demonstrate the performance of MRAC-NNTDE. The resulting simulations demonstrate that the MRAC-NNTDE has effectively achieved better control inputs, including robust compensation of uncertain unknown dynamics under external perturbations, minimal steady-state error, precise joint position tracking, and rapid response speed. For further research work, the proposed control scheme can be used with fractional-order control for the practical application of the robotic system.

Acknowledgments

This paper is derived from a research grant funded by the Research, Development, and Innovation Authority (RDIA), Kingdom of Saudi Arabia, with grant number 13382-psu-2023-PSNU-R-3-1-EI-. The authors would like to acknowledge Prince Sultan University, Riyadh, Saudi Arabia, for their support of this publication. This research is supported by the Automated Systems and Computing Lab (ASCL), Prince Sultan University, Riyadh, Saudi Arabia.

Funding

This work is funded by Prince Sultan University, Riyadh, Saudi Arabia.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Author contributions

Conceptualization: Saim Ahmed, Ahmad Taher Azar
Formal analysis: All authors
Investigation: Saim Ahmed, Ahmad Taher Azar
Methodology: Saim Ahmed, Ahmad Taher Azar
Writing – original draft: Saim Ahmed
Writing – review & editing: All authors

Availability of data

Not applicable.

AI tools statement

All authors confirm that no AI tools were used in the preparation of this manuscript.

References

1. Ajel AR. Robust model reference adaptive control for tail-sitter vtol aircraft. *Actuators.* 2021;10(7).
2. Lavretsky E. Adaptive control: Introduction, overview, and applications. *Lecture notes IEEE Robust Adapt. Control Workshop* 208; 2008.
3. Dan Z, Bin W. A review on model reference adaptive control of robotic manipulators. *Annu Rev Control.* 2017;43:188–198.
4. Xiao Shunli LJ, Yangmin L. A model reference adaptive pid control for electromagnetic actuated micro-positioning stage. In *2012 IEEE International Conference on Automation Science and Engineering (CASE)*, IEEE, 2012;97–102.
5. Goléa N, Goléa A, Kadjoudj M. Nonlinear model reference adaptive control using takagi-sugeno fuzzy systems. *J Intell Fuzzy Syst.* 2006;17(1):47–57.
6. Guo L, Parsa L. Model reference adaptive control of five-phase ipm motors based on neural network. *IEEE Trans Ind Electron.* 2011;59(3):1500–1508.
7. Wu C, Zhao J. H∞ adaptive tracking control for switched systems based on an average dwell-time method. *Int J Syst Sci.* 2015;46(14):2547–2559.
8. Fang Y, Fei J, Ma K. Model reference adaptive sliding mode control us- ing rbf neural network for active power filter. *Int J Electr Power Energy Syst.* 2015;73:249–258.
9. Uçak K, Arslantürk BN. Adaptive mimo fuzzy pid controller based on peak observer. *Int J Optim Control Theor Appl.* 2023;13(2).
10. Ahmed S, Wang H, Tian Y. Modification to model reference adaptive control of 5-link exoskeleton with gravity compensation. In *2016 35th Chinese Control Conference (CCC)*, 2016;6115–6120.
11. Nguyen N, Summers E. On time delay margin estimation for adaptive control and robust modification adaptive laws. In *AIAA Guidance, navigation, and control conference* 2011;6438.
12. Raja MM, Vijayakumar V, Veluvolu KC. Improved order in hilfer fractional differential systems: Solvability and optimal control problem for hemivariational inequalities. *Chaos Solitons Fractals.* 2024;188:115558.
13. Ma Y-K, Raja MM, Nisar KS, Shukla A, Vijayakumar V. Results on controllability for sobolev type fractional dif- ferential equations of order 1¡ r¡ 2 with finite delay. *AIMS Math.* 2022;7(6):10215–10233.
14. Raja MM, Vijayakumar V, Veluvolu KC, Shukla A, Nisar KS. Existence and optimal control results for caputo fractional delay clark's subdifferential inclusions of order order $r \in (1, 2)$ with sectorial operators. *Optim Control Appl Methods.* 2024;45(4):1832–1850.
15. Lavretsky E, Kevin W, Howe D. *Robust and adaptive control with aerospace applications*. England: Springer- Verlag London, 2013.
16. Lewis FL, Dawson DM, Abdallah CT. *Robot manipulator control: Theory and practice*. CRC Press, 2003.
17. Zhou X, Tian Y, Wang H. Neural network state observer-based robust adaptive fault-tolerant quantized iterative learning control for the rigid-flexible coupled robotic systems with unknown time delays. *Appl Math Comput.* 2022;430:127286.
18. Zhou X, Wang H, Wu K, Zheng G. Fixed-time neural network trajectory tracking control for the rigid-flexible coupled robotic mechanisms with large beam- deflections. *Appl Math Model.* 2023;118:665–691.
19. Ghazi, Farah F., and Luma NM Tawfiq. Design optimal neural network based on new LM training algorithm for solving 3D-PDEs. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*. 2024;14(3):249-260.
20. Wang Y, Yan F, Chen J, Ju F, Chen B. A new adaptive time-delay control scheme for cable-driven manipulators. *IEEE Trans Ind Informat.* 2018;15(6):3469–3481.
21. Wang Y, Li B, Yan F, Chen B. Practical adaptive fractional-order nonsingular terminal sliding mode control for a cable-driven manipulator. *Int. J. Robust Nonlinear Control* 2019;29(5):1396–1417.
22. Youcef-Toumi K, Wu S-T. Input/output linearization using time delay control. 1992;114(1):10–19.
23. Long Y, Liu X, Yao C, Song E. Model-free adaptive full-order sliding mode control with time delay estimation of high- pressure common rail system. *J Vib Control.* 2024;10775463241256225.
24. Ahmed S, Azar AT, Ibraheem IK. Model-free scheme using time delay estimation with fixed-time fsmc for the non- linear robot dynamics. *AIMS Math.* 2024;9(4):9989–10009.
25. Yadegari H, Beyramzad J, Khanmirza E. Magnetorquers-based satellite attitude control using interval type-ii fuzzy terminal sliding mode control with time delay estimation. *Adv Space Res* 2022;69(8):3204–3225.

26. Han S, Wang H, Tian Y, Christov N. Time-delay estimation based com- puted torque control with robust adaptive rbf neural network compensator for a reha- bilitation exoskeleton. *ISA Trans.* 2020;97:171–181.
27. Ahmed S, Azar AT. Enhanced tracking control for n-dof robotic manipula- tors: A fixed-time terminal sliding mode approach with time delay estimation. *Results Eng.* 2024;24:102904.
28. Li G, Ma X, Li Y. Adaptive sliding mode control based on time-delay estimation for underactuated 7-dof tower crane. *IEEE Trans Syst Man Cybern Syst.* 2025.
29. Van M, Ge SS, Ren H. Finite time fault tolerant control for robot ma- nipulators using time delay estimation and continuous nonsingular fast terminal sliding mode control. *IEEE Trans Cybern* 2016;47(7):1681–1693.
30. Choi J, Kwon W, Lee YS, Han S. Adaptive time-delay estimation error compensation for application to robot manipulators. *Control Eng Pract.* 2024;151:106029.
31. Jin M, Kang SH, Chang PH, Lee J. Robust control of robot manipulators using inclusive and enhanced time delay control. *IEEE/ASME Trans Mechatronics.* 2017;22(5):2141–2152.
32. Jin M, Kang SH, Chang PH. Robust compliant motion control of robot with nonlinear friction using time-delay es- timation. *IEEE Trans Ind Electron.* 2008;55(1):258–269.
33. Hsia TC, Gao L. Robot manipulator control using decentralized linear time-invariant time-delayed joint controllers. In *Proceedings IEEE International Conference on Robotics and Automation*, IEEE, 1990;2070–2075.
34. Yavuz M, Öztürk M, Yaşkıran B. Comparison of fractional order sliding mode controllers on robot manipulator. *ICAM'24* 2024;188.
35. Reichhartinger M, Spurgeon S, Forstinger M, Wipfler M. A robust exact differentiator toolbox for matlab®/simulink®. *IFAC-PapersOnLine*, 2017;50(1):1711–1716.
36. Asl SBF, Moosapour SS. Adaptive backstepping fast terminal sliding mode controller design for ducted fan engine of thrust-vectored aircraft. *Aerosp. Sci. Technol.* 2017;71:521–529.
37. Ahmed S, Azar AT, Tounsi M. Adaptive fault tolerant non-singular sliding mode control for robotic manipulators based on fixed-time control law. *Actuators.* 2022;11(12):353.
38. Jasim Mohamed, Mohamed, et al. Hybrid controller with neural network PID/FOPID operations for two-link rigid robot manipulator based on the zebra optimization algorithm. *Frontiers in Robotics and AI.* 2024;11:1386968.

Saim Ahmed *received the B.S. degree in electronics from the Sir Syed University of Science and Technology, Pakistan, in 2009, the M.E. degree in industrial control and automation from Hamdard University, Pakistan, in 2013, and the Ph.D. degree in control science and engineering from Nanjing University of Science and Technology, China, in 2019. He is currently a Postdoctoral Researcher with the Automated Systems and Computing Laboratory (ASCL), Prince Sultan University, Saudi Arabia. His research interests include the theory and applications of adaptive control, sliding mode control, time delay control, robotic manipulators, and non-linearities and their compensation..*

https://orcid.org/0000-0002-2302-705X

Ahmad Taher Azar *is currently a Full Professor with Prince Sultan University, Riyadh, Saudi Arabia, where he is also the Leader of the Automated Systems and Computing Laboratory (ASCL). He has expertise in artificial intelligence, control theory and applications, robotics, machine learning, computational intelligence, and dynamic system modeling. He has authored/coauthored over 450 research papers in prestigious peer-reviewed journals, book chapters, and conference proceedings. He is currently an Editor of IEEE SYSTEMS JOURNAL, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, Human Centric Computing and Information Sciences (Springer), and Engineering Applications of Artificial Intelligence (Elsevier).*

https://orcid.org/0000-0002-7869-6373

Ibraheem Kasim Ibraheem *was born in Baghdad, Iraq, in 1976. He received the B.S. degree in electrical engineering from the University of Baghdad, Baghdad, Iraq, in 1998 and M.Sc. Ph.D degrees in Computer Control Engineering from the same university and department, in 2001 and 2007, respectively. In 2012, he became a Professor. His research interests include Active disturbance rejection control, power control, Robotics, signal processing, nonlinear control, and intelligent control applications.*

http://orcid.org/0000-0001-7009-3634