Journal of Robotics and Control (JRC)
Volume 6, Issue 4, 2025
ISSN: 2715-5072, DOI: 10.18196/jrc.v6i4.26460

Neural Network-Based Adaptive Robust Fractional PID Control for Robotic Systems

Yen-Vu Thi [1*], Nan-Wang Yao [2], Hai-Nguyen Huu [3], Cuong-Pham Van [4], Tung-Ngo Manh [5]
[1, 3, 5] Faculty of Automation, School of Electrical and Electronic Engineering, Hanoi University of Industry, Hanoi, Vietnam
[4] Faculty of Electrical Engineering, School of Electrical and Electronic Engineering, Hanoi University of Industry, Hanoi, Vietnam
[2] College of Electrical and information Engineering, Hunan University, Hunan Changsha, China
[4] ThaiBinh University, Thai Binh, Vietnam
Email: [1] yenvt@haui.edu.vn, [2] yaonan@hnu.edu.cn, [3] hainh@haui.edu.vn, [4] cuongpv0610@haui.edu.vn, [5] tung_nm@haui.edu.vn
*Corresponding Author

***Abstract*—This paper proposed an Adaptive Robust Fractional Oder PID controller based on neural networks (ARFONNs) in order to improve the trajectory tracking of Robotic systems. Robots are nonlinear objects with uncertain models, they are always affected by noise in the working process such as the payload variation, nonlinear friction, external disturbances, ect. To address this problem of robot, a proposed controller inherits the advantages of neural network, adaptive method and sliding mode controller to achieve fast and accurate control. The neural network controller has simple architecture, better approximation for the unknown dynamic of robotic systems, and fast training capability. Moreover, due to its robust nature, Sliding Mode Control (SMC) is a widely adopted nonlinear control approach. Furthermore, the quality of the robot control system is improved based on combining the flexibility of Fractional Order PID. The adaptive laws of the ARFONNs are defined by selecting a suitable Lyapunov function to the control system obtain global stability. In addition, Simulation and experimental results of the ARFONNs controller are conducted on a two-link Cleaning and Detecting Robot. The simulation and experimental results have compared with the Adaptive Robust Neural networks (ARNNs) and The neural networks controller (NNs) to demonstrate the stability and robustness as well as the performance of the ARFONNs controller.**

Keywords—Trajectory Tracking Control; Adaptive Sliding Mode Controller; Fractional Order PID; Neural Network-Based Control; A Two-Link Clearning and Detecting Robot (CDR).

I. INTRODUCTION

The industry 4.0 revolution is happening strongly all over the world and Robots play an important role. The problem of robots has attracted the attention of many scientists. Robots are MIMO objects with strong nonlinearity, uncertain parameters and are subject to external disturbance during the working robot process. To improve the working efficiency of robots, besides enhancing accuracy in mechanical assembly steps, control also plays a crucial role. Therefore, designing a suitable controller is a challenging problem to solve. To deal with this problem, many solutions have been researched and proposed in [1]-[23] such as PID, adaptive controller, sliding mode controller, bacstepping controller, etc. Adaptive controller is the problem of syzthesizing controller to keep the system stable, even if unwanted disturbances, structural changes or unknown parameters of the control object occur during the working process. When there is a change in the object, the controller will automatically adjust the structure and parameters to ensure constant system quality [3]-[5]. Sliding mode control is known as a simple sustainable nonlinear control method, effective. This control method is less sensitive to variation of system parameters, has good resistance to interference, and reacts quickly. However, the difficulty in designing a sliding mode controller requires knowing clearly the mathematical model of the object as well as the upper limit of the uncertain components of the model. Besides, there is always a phenomenon of frequency oscillation around the sliding surface [22]-[23]. In recent years, intelligent controllers based on fuzzy logic and neural networks to control the position of robot manipulators have received attention [24]-[40]. Fuzzy logic control has been applied and has proven successful for the approximation nonlinear systems [41]-[57]. In [49], the W. Chang and fellow-workers proposed an adaptive backstepping controller based on fuzzy logic for flexible robot manipulators. The authors approximated the unknown system by fuzzy logic. With the proposed controller, all variables within the closed – loop system are guaranteed to stay bounded, and the system output can track the reference signal with high accuracy. In [57], to cope with uncertainties in serial robotic manipulators, the authors proposed an adaptive robust sliding mode control scheme in the task space using a fuzzy logic–based method. The proposed controller was built without prior knowledge of the robotic system, and the fuzzy controller's rules were designed based on human experience and expertise to achieve good performance under uncertainties. However, human experience and expertise may not be sufficient, and designing appropriate membership functions and fuzzy rules remains a challenging task. To deal with this challenge, neural network-based controllers were proposed for controlling robot manipulators [58]-[69]. To reject the unknown dynamics of robot model, the weights of the neural network were updated by online training rules during the working process of the control system. In [76], the authors developed a robust neural network-based output feedback control scheme for robot manipulators, where joint velocities were not directly measured but estimated using a neural network. The online learning laws of weight neural network were defined by using the Lyapunov theorem. So, the robustness and stability of

proposed controllers were improved. In [70], an adaptive control approach employing an RBF neural network was developed for a robotic manipulator with uncertainties. The application of RBFNN to the control system has helped to improve the approximation accuracy of uncertain dynamics and enhance control performance. In cases where the inputs move beyond the approximation domain of the RBFNN, the controller – incorporating either local or global bias-reverts to PID control to drive the states back and strengthen the robustness of the adaptive RBFNN controller. In [71], a neural network controller using RBF functions was proposed to solve with the uncertainties model in the biped robot system. In this control, the unknown dynamic behavior of the bipedal robot was approximated using a neural network framework. The robustness and stability of the proposed controller for biped robot systems were achieved and proved based on Lyapunov theorem.

Recently, fractional-order differential equations have garnered significant attention and applications due to their capacity to accurately model a wide range of nonlinear systems. In addition, fractional order calculus theory and stability have also been applied to control systems [77]-[80]. In [81], the advanced version of the PID controller is the parallel form of the Fraction Order PID (FOPID). Two additional parameters were provided, which may make FOPID more robust due to its fractional degree structure. The FOPID has become a popular controller in robotics applications, it is rarely used for soft robotics [82], [83]. In [84], the authors proposed a controller that combines the FOPID with linear matrix inequality technique and genetic algorithm to improve current tracking in servo systems, positioning it as a feasible alternative to traditional proportional-integral controllers. The study in [85] focused on constructing an optimal fuzzy immune FOPID controller for robot tracking of a 3DOF robot manipulator. In [86], A fractional-order active disturbance rejection controller was introduced for time-delay systems to enable independent control of both servo and regulation actions. Experimental results demonstrated that the proposed controller successfully achieved this independent control and outperformed conventional time-delay controllers in terms of speed tracking, load disturbance rejection, and robustness against variations in plant parameters.

In this study, a neural network-based adaptive fractional-order PID controller is developed to ameliorate the control quality for the robot manipulator system. The proposed controller operates more flexibly by adjusting two parameters of the Fractional-order PID control compared to the classic PID controller [74]-[75]. When the proposed controller is used to the CDR, tracking efficiency and convergence speed are improved.

This paper is structured as follows: Section 2 presents Dynamic of robot. RBF neural network model is described in Section 3. Section 4 presents Adaptive FONNs controller system. The simulation and experiment results of Cleaning and Detecting robot are proposed. Finally, Section 6 shows concluding remarks

II. DYNAMIC OF ROBOT

To design a controller for the robot system, it is first necessary to formulate the robot's dynamic equations by Lagrange or Newton-Euler. According to Reference [74], the dynamic of n-link robot manipulators with uncertain models and external disturbances is shown as follows:

$$M_{FO}(\delta)\ddot{\delta} + V_{FO}(\delta,\dot{\delta})\dot{\delta} + G_{FO}(\delta) + F_{FO}(\dot{\delta}) + \tau_{FOd} = \tau_{FO} \quad (1)$$

where $\delta \in R^{n\times 1}$ is the position vectors of joint (Rad), $(\dot{\delta},\ddot{\delta}) \in R^{n\times 1}$ are the velocity vectors and acceleration vectors of joint (rad/s, rad/s^2). $M_{FO}(\delta) \in R^{n\times n}$ is the symmetric inertial Matrix, $V_{FO}(\delta,\dot{\delta}) \in R^{n\times n}$ is the vector of Coriolis and Centripetal forces, $G_{FO}(\delta) \in R^{n\times 1}$ vector containing Gravity forces and torques, $F_{FO}(\dot{\delta}) \in R^{n\times 1}$ is a vector of friction term, $\tau_{FOd} \in R^{n\times 1}$ is the unknown disturbances input, and $\tau_{FO} \in R^{n\times 1}$ is the joints's torque input.

Each component of the robot's dynamic model $M_{FO}(\delta), V_{FO}(\delta,\dot{\delta}), G_{FO}(\delta), F_{FO}(\dot{\delta})$ contain both known and unknown parts. The known components are derived from the nominal physical parameters of the robot. However, uncertainties such as modeling errors, parameter variations, and external disturbances introduce unknown elements into these matrices, especially when masses, link lengths, or center-of-mass positions are inaccurately estimated.

Assumption: the dynamics of robot manipulators (1) has known estimation terms and unknown uncertain terms.

$$\begin{aligned}&\left(\hat{M}_{FO0}(\delta) + \Delta M_{FO}(\delta)\right)\ddot{\delta} \\ &\quad + \left(\hat{V}_{FO0}(\delta,\dot{\delta}) + \Delta V_{FO}(\delta,\dot{\delta})\right)\dot{\delta} \\ &\quad + \left(\hat{G}_{FO0}(\delta) + \Delta G_{FO}(\delta)\right) \\ &\quad + \left(\hat{F}_{FO0}(\dot{\delta}) + \Delta F_{FO}(\dot{\delta})\right) + \tau_{FOd} \\ &\quad = \tau_{FO}\end{aligned} \quad (2)$$

where $\hat{M}_{FO0}(\delta), \hat{V}_{FO0}(\delta,\dot{\delta}), \hat{G}_{FO0}(\delta), \hat{F}_{FO0}(\dot{\delta})$ are the nominal terms, and $\Delta M_{FO}(\delta), \Delta V_{FO}(\delta,\dot{\delta}), \Delta G_{FO}(\delta)$, and $\Delta F_{FO}(\dot{\delta})$ are the unknown parts.

The dynamics of robot manipulator (2) can be rewritten as follows:

$$\begin{aligned}&\hat{M}_{FO0}(\delta)\ddot{\delta} + \Delta M_{FO}(\delta)\ddot{\delta} + \hat{V}_{FO0}(\delta,\dot{\delta})\dot{\delta} \\ &\quad + \Delta V_{FO}(\delta,\dot{\delta})\dot{\delta} + \hat{G}_{FO0}(\delta) \\ &\quad + \Delta G_{FO}(\delta) + \hat{F}_{FO0}(\dot{\delta}) + \Delta F_{FO}(\dot{\delta}) \\ &\quad + \tau_{FOd} = \tau_{FO} \\ &\hat{M}_{FO0}(\delta)\ddot{\delta} + \hat{V}_{FO0}(\delta,\dot{\delta})\dot{\delta} + \hat{G}_{FO0}(\delta) + \hat{F}_{FO0}(\dot{\delta}) \\ &\quad = \tau_{FO} - \Delta M_{FO}(\delta)\ddot{\delta} \\ &\quad - \Delta V_{FO}(\delta,\dot{\delta})\dot{\delta} - \Delta G_{FO}(\delta) \\ &\quad - \Delta F_{FO}(\dot{\delta}) - \tau_{FOd}\end{aligned} \quad (3)$$

III. RBF NEURAL NETWORK MODEL

The RBFNNs controller has structure as shown in Fig 1, which comprises three layers: Input, Hidden and Output layer.

$\delta, \dot{\delta}$ are input signals of the input layer. $\Theta_M, \Theta_V, \Theta_G$, and Θ_F denote the hidden layer outputs. $W_{M_{FO}}, W_{V_{FO}}, W_{G_{FO}}$, and $W_{F_{FO}}$ are ideal weight value of RBF. $\widehat{W}_{M_{FO}}, \widehat{W}_{V_{FO}}, \widehat{W}_{G_{FO}}$, and $\widehat{W}_{F_{FO}}$ are estimates of $W_{M_{FO}}, W_{V_{FO}}, W_{G_{FO}}$, and $W_{F_{FO}}$, respectively. $\Delta M_{FO}(\delta), \Delta V(\delta, \dot{\delta}), \Delta G_{FO}(\delta)$, and $\Delta F_{FO}(\dot{\delta})$ are the output values of RBFNNs controller

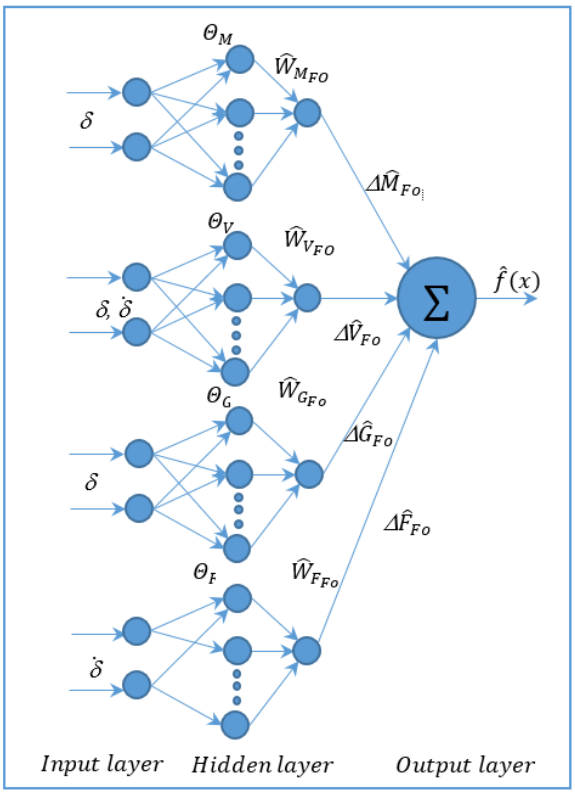

Fig. 1. Structure of RBF neural networks

The output values of the RBFNNs controller are respectively defined as follows:

$$\begin{aligned} \Delta M_{FO}(\delta) &= W_{M_{FO}}^T \Theta_M(\delta) + \Xi_{M_{FO}} \\ \Delta V_{FO}(\delta, \dot{\delta}) &= W_{V_{FO}}^T \Theta_V(\delta, \dot{\delta}) + \Xi_{V_{FO}} \\ \Delta G_{FO}(\delta) &= W_{G_{FO}}^T \Theta_G(\delta) + \Xi_{G_{FO}} \\ \Delta F_{FO}(\dot{\delta}) &= W_{F_{FO}}^T \Theta_F(\dot{\delta}) + \Xi_{F_{FO}} \end{aligned} \tag{4}$$

where; $\Xi_{M_{FO}}, \Xi_{V_{FO}}, \Xi_{G_{FO}}$, and $\Xi_{F_{FO}}$ are the error of $\Delta M_{FO}(\delta), \Delta V(\delta, \dot{\delta}), \Delta G_{FO}(\delta)$, and $\Delta F_{FO}(\dot{\delta})$, respectively. $\Delta\widehat{M}_{FO}(\delta)\Delta\hat{V}_{,}(\delta, \dot{\delta}), \Delta\widehat{G}_{FO}(\delta)$, and $\Delta\widehat{F}_{FO}(\dot{\delta})$ denote the estimated values of the $\Delta M_{FO}(\delta), \Delta V(\delta, \dot{\delta}), \Delta G_{FO}(\delta)$, and $\Delta F_{FO}(\dot{\delta})$, respectively.

The output of the jth hidden neuron will determine as follow:

$$\Theta_j = exp\left(-\|x - c_j\|^2 / 2d_j^2\right)$$

c_j, d_j are center and width of the j-th neuron. c selects by K-means clustering method or randomly from training data. A single value d is determined based on the distances between centers, the distances to nearest neighbors, or learning widths during training.

The initial weight value of RBF will be taken based on the designer's experience and then they will be updated during the system's working process by the online update laws.

The values of $\Delta\widehat{M}_{FO}(\delta)\Delta\hat{V}_{,}(\delta, \dot{\delta}), \Delta\widehat{G}_{FO}(\delta)$, and $\Delta\widehat{F}_{FO}(\dot{\delta})$ defined, respectively as:

$$\begin{aligned} \Delta\widehat{M}_{FO}(\delta) &= \widehat{W}_{M_{FO}}^T \Theta_M(\delta) \\ \Delta\hat{V}_{FO}(\delta, \dot{\delta}) &= \widehat{W}_{V_{FO}}^T \Theta_V(\delta, \dot{\delta}) \\ \Delta\hat{G}_{FO}(\delta) &= \widehat{W}_{G_{FO}}^T \Theta_G(\delta) \\ \Delta\hat{F}_{FO}(\dot{\delta}) &= \widehat{W}_{F_{FO}}^T \Theta_F(\dot{\delta}) \end{aligned} \tag{5}$$

Where $\widehat{W}_{M_{FO}}, \widehat{W}_{V_{FO}}, \widehat{W}_{G_{FO}}$, and $\widehat{W}_{F_{FO}}$ are estimate of $W_{M_{FO}}, W_{V_{FO}}, W_{G_{FO}}$, and $W_{F_{FO}}$, respectively.

IV. ARFONNs Controller System

This section, the ARFONNs controller is designed to improve the error tracking of robot manipulators. The proposed controller inherits the advantages of neural network, adaptive method and sliding mode controller. Substituting (4) into (3), we have:

$$\begin{aligned} \widehat{M}_{FO0}(\delta)\ddot{\delta} + \hat{V}_{FO0}(\delta, \dot{\delta})\dot{\delta} + \hat{G}_{FO0}(\delta) + \hat{F}_{FO0}(\dot{\delta}) \\ = \tau_{FO} + F(t) + \Gamma \end{aligned} \tag{6}$$

With $F(t) = -W_{M_{FO}}^T \Theta_M(\delta)\ddot{\delta} - W_{V_{FO}}^T \Theta_V(\delta, \dot{\delta})\dot{\delta} - W_{G_{FO}}^T \Theta_G(\delta)\text{-}W_{F_{FO}}^T \Theta_F(\dot{\delta})$, $\Gamma = -\Xi_{M_{FO}}\ddot{\delta} - \Xi_{V_{FO}}\dot{\delta} - \Xi_{G_{FO}} - \Xi_{F_{FO}} - \tau_{FOd}$

The FOPID controller is defined as:

$$\tau_{PID} = k_p\varepsilon(t) + k_i \frac{d^{-\beta}}{dt}\varepsilon(t) + k_d \frac{d^{\alpha}}{dt}\varepsilon(t) \tag{7}$$

where k_p, k_d, k_i are the constants of proportionality, differentiation, and integration respectively. β, α are real numbers.

Definitions of the tracking error and sliding surface are give as follows:

$$\varepsilon(t) = \delta_d - \delta \tag{8}$$

$$\begin{aligned} s(t) = \frac{d\varepsilon(t)}{dt} + k_p \int_0^t \varepsilon(t) + k_i \frac{d^{-\beta-1}}{dt}\varepsilon(t) \\ + k_d \frac{d^{\alpha-1}}{dt}\varepsilon(t) \end{aligned} \tag{9}$$

where $s(t)$ indicates the error over time, and k_p, k_d, k_i respectively denote the proportional, derivative, and integral gains. β, α are two additional parameters of the fractional order controller, and they are positive constants. The derivative of $s(t)$ and substituting (6), (2), we have:

$$\begin{aligned} \dot{s}(t) &= \ddot{\varepsilon}(t) + k_p\dot{\varepsilon}(t) + k_i \frac{d^{-\beta}}{dt}\varepsilon(t) + k_d \frac{d^{\alpha}}{dt}\varepsilon(t) \\ &= \ddot{\delta}_d - \ddot{\delta} + k_p\dot{\varepsilon}(t) \\ &\quad + k_i \frac{d^{-\beta}}{dt}\varepsilon(t) + k_d \frac{d^{\alpha}}{dt}\varepsilon(t) \\ \dot{s}(t) &= \ddot{\delta}_d - \widehat{M}^{-1}{}_{FO0}\left(\tau_{FO} + F(t) - \hat{V}_{FO0}\dot{\delta}\right. \\ &\quad \left. - \hat{G}_{FO0}(\delta) - \hat{F}_{FO0}\right) + k_p\dot{\varepsilon}(t) \\ &\quad + k_i \frac{d^{-\beta}}{dt}\varepsilon(t) + k_d \frac{d^{\alpha}}{dt}\varepsilon(t) \end{aligned} \tag{10}$$

The ARFONNs controller is defined as:

$$\begin{aligned}\tau_{FO} = \widehat{M}_{FO0}\ddot{\delta}_d + \widehat{V}_{FO0}\dot{\delta} + \widehat{G}_{FO0}(\delta) + \widehat{F}_{FO0} \\ + \widehat{M}_{FO0}\tau_{PID} + \widehat{M}_{FO0}\tau_{smc} \\ + \widehat{M}_{FO0}\hat{F}(t)\end{aligned} \quad (11)$$

With τ_{smc} is a sliding model control term, $\hat{F}(t)$ is considered an approximation of $F(t)$, and $\hat{F}(t)$ is defined as follows:

$$\begin{aligned}\hat{F}(t) = -\widehat{W}_{M_{FO}}^T\Theta_M(\delta)\ddot{\delta} - \widehat{W}_{V_{FO}}^T\Theta_V(\delta,\dot{\delta})\dot{\delta} - \widehat{W}_{G_{FO}}^T\Theta_G(\delta) \\ - \widehat{W}_{F_{FO}}^T\Theta_F(\dot{\delta})\end{aligned}$$

The robust controller is designed as:

$$\begin{aligned}\tau_{smc} = \frac{s}{\|s\|}\left(\frac{k_M W_{M_{FO}}^2}{4} + \frac{k_V W_{V_{FO}}^2}{4} + \frac{k_G W_{G_{FO}}^2}{4}\right. \\ \left. + \frac{k_F W_{F_{FO}}^2}{4}\right) + k_P\, sgn(s) \\ = \frac{s}{\|s\|}\Upsilon + k_P\, sgn(s)\end{aligned} \quad (12)$$

where $\Upsilon = \frac{k_M W_{M_{FO}}^2}{4} + \frac{k_V W_{V_{FO}}^2}{4} + \frac{k_G W_{G_{FO}}^2}{4} + \frac{k_F W_{F_{FO}}^2}{4}$ and $k_P > \Gamma$.

The online update laws of FONNs are proposed as:

$$\begin{cases}\dot{\widehat{W}}_{M_{FO}} = -F_M\Theta_M(\delta)\ddot{\delta} + k_M F_M\|s\|\widehat{W}_{M_{FO}} \\ \dot{\widehat{W}}_{V_{FO}} = -F_V\Theta_V(\delta,\dot{\delta})\dot{\delta} + k_V F_V\|s\|\widehat{W}_{V_{FO}} \\ \dot{\widehat{W}}_{G_{FO}} = -F_G\Theta_G(\delta) + k_G F_G\|s\|\widehat{W}_{G_{FO}} \\ \dot{\widehat{W}}_{F_{FO}} = -F_F\Theta_F(\delta,\dot{\delta},\ddot{\delta}) + k_F F_F\|s\|\widehat{W}_{F_{FO}}\end{cases} \quad (13)$$

where $k_M, k_V, k_G, k_F, F_M, F_V, F_G, F_F$ are the constant diagonal constant matrices and which are positive.

Consider an n-link robotic manipulator described by equation (3). When the ARFONNs update law follows (13) and the SMC robust controller is defined as (12), it can be ensured that the tracking error and all system parameters converge to zero. The ARFONNs controller is defined as (11). Stability of the control system can be ensured based on Lyapunov analysis, provided that the Lyapunov function is positive definite and its derivative is negative semidefinite. Thus, ensuring the stability of the overall control system requires satisfying these conditions. Choosing a Lyapunov function as:

$$\begin{aligned}L(t) = \frac{1}{2}s^T s + \frac{1}{2}tr(\widetilde{W}_{M_{FO}}^T F_M^{-1}\widetilde{W}_{M_{FO}}) \\ + \frac{1}{2}tr(\widetilde{W}_{V_{FO}}^T F_V^{-1}\widetilde{W}_{V_{FO}}) \\ + \frac{1}{2}tr(\widetilde{W}_{G_{FO}}^T F_G^{-1}\widetilde{W}_{G_{FO}}) + \frac{1}{2}tr(\widetilde{W}_{F_{FO}}^T F_F^{-1}\widetilde{W}_{F_{FO}})\end{aligned} \quad (14)$$

The derivative of $L(t)$ is:

$$\begin{aligned}\dot{L}(t) = s^T\dot{s} - tr(\widetilde{W}_{M_{FO}}^T F_M^{-1}\dot{\widehat{W}}_{M_{FO}}) \\ - tr(\widetilde{W}_{V_{FO}}^T F_V^{-1}\dot{\widehat{W}}_{V_{FO}}) \\ - tr(\widetilde{W}_{G_{FO}}^T F_G^{-1}\dot{\widehat{W}}_{G_{FO}}) \\ - tr(\widetilde{W}_{F_{FO}}^T F_F^{-1}\dot{\widehat{W}}_{F_{FO}})\end{aligned} \quad (15)$$

Applying (8) and (9) into (15), we have (16):

$$\begin{aligned}&\dot{L}(t) \\ &= s^T\begin{pmatrix}-\tau_{PID} - \tau_{smc} + \tilde{F}(t) + k_p\varepsilon(t) + k_i\frac{d^{-\beta}}{dt}\varepsilon(t) \\ +k_d\frac{d^{\alpha}}{dt}\varepsilon(t)\end{pmatrix} \\ &-tr(\widetilde{W}_{M_{FO}}^T F_M^{-1}\dot{\widehat{W}}_{M_{FO}}) - tr(\widetilde{W}_{V_{FO}}^T F_V^{-1}\dot{\widehat{W}}_{V_{FO}}) \\ &-tr(\widetilde{W}_{G_{FO}}^T F_G^{-1}\dot{\widehat{W}}_{G_{FO}}) - tr(\widetilde{W}_{F_{FO}}^T F_F^{-1}\dot{\widehat{W}}_{F_{FO}})\end{aligned} \quad (16)$$

with $\tilde{F}(t) = -\widetilde{W}_{M_{FO}}^T\Theta_M(\delta)\ddot{\delta} - \widetilde{W}_{V_{FO}}^T\Theta_V(\delta,\dot{\delta})\dot{\delta} - \widetilde{W}_{G_{FO}}^T\Theta_G(\delta) - \widetilde{W}_{F_{FO}}^T\Theta_F(\delta,\dot{\delta},\ddot{\delta})$,

Substituting (6) into (15), we have:

$$\begin{aligned}\dot{L}(t) = s^T(-\tau_{smc} + \tilde{F}(t)) - tr(\widetilde{W}_{M_{FO}}^T F_M^{-1}\dot{\widehat{W}}_{M_{FO}}) \\ - tr(\widetilde{W}_{V_{FO}}^T F_V^{-1}\dot{\widehat{W}}_{V_{FO}}) \\ - tr(\widetilde{W}_{G_{FO}}^T F_G^{-1}\dot{\widehat{W}}_{G_{FO}}) \\ - tr(\widetilde{W}_{F_{FO}}^T F_F^{-1}\dot{\widehat{W}}_{F_{FO}})\end{aligned} \quad (17)$$

By using (13), (17) can be rewritten as:

$$\begin{aligned}\dot{L}(t) = -s^T\tau_{smc} + s^T\Gamma - tr(\widetilde{W}_{M_{FO}}^T k_M\|s\|\widehat{W}_{M_{FO}}) \\ - tr(\widetilde{W}_{V_{FO}}^T k_V\|s\|\widehat{W}_{V_{FO}}) \\ - tr(\widetilde{W}_{G_{FO}}^T k_G\|s\|\widehat{W}_{G_{FO}}) \\ - tr(\widetilde{W}_{F_{FO}}^T k_F\|s\|\widehat{W}_{F_{FO}})\end{aligned} \quad (18)$$

$$\Gamma = -\Xi_{M_{FO}}\ddot{\delta} - \Xi_{V_{FO}}\dot{\delta} - \Xi_{G_{FO}} - \Xi_{F_{FO}} - \tau_{FOd}$$

Where

$$\widetilde{W}_{M_{FO}} = W_{M_{FO}} - \widehat{W}_{M_{FO}}; \widetilde{W}_{V_{FO}} = W_{V_{FO}} - \widehat{W}_{V_{FO}};$$
$$\widetilde{W}_{G_{FO}} = W_{G_{FO}} - \widehat{W}_{G_{FO}}; \widetilde{W}_{F_{FO}} = W_{F_{FO}} - \widehat{W}_{F_{FO}}$$

Applying (18) becomes:

$$\begin{aligned}\dot{L}(t) = -s^T\tau_{smc} + s^T\Gamma + k_M\|s\|tr\widetilde{W}_{M_{FO}}^T(W_{M_{FO}} \\ - \widetilde{W}_{M_{FO}}) + k_V\|s\|tr\widetilde{W}_{V_{FO}}^T(W_{V_{FO}} \\ - \widetilde{W}_{V_{FO}}) + k_G\|s\|tr\widetilde{W}_{G_{FO}}^T(W_{G_{FO}} \\ - \widetilde{W}_{G_{FO}}) + k_F\|s\|tr\widetilde{W}_{F_{FO}}^T(W_{F_{FO}} \\ - \widetilde{W}_{F_{FO}})\end{aligned} \quad (19)$$

$$\begin{aligned}tr\widetilde{W}^T(W - \widetilde{W}) = (\widetilde{W}, W) - \|\widetilde{W}\|^2 \\ \le \|\widetilde{W}\|\|W\| - \|\widetilde{W}\|^2\end{aligned}$$

By using

We have:

$$\begin{aligned}\dot{L}(t) \le -s^T\tau_{smc} + s^T\Gamma \\ + k_M\|s\|\left(\|\widetilde{W}_{M_{FO}}\|\|W_{M_{FO}}\| - \|\widetilde{W}_{M_{FO}}\|^2\right) \\ + k_V\|s\|\left(\|\widetilde{W}_{V_{FO}}\|\|W_{V_{FO}}\| - \|\widetilde{W}_{V_{FO}}\|^2\right) \\ + k_G\|s\|\left(\|\widetilde{W}_{G_{FO}}\|\|W_{G_{FO}}\| - \|\widetilde{W}_{G_{FO}}\|^2\right) \\ + k_F\|s\|\left(\|\widetilde{W}_{F_{FO}}\|\|W_{F_{FO}}\| - \|\widetilde{W}_{F_{FO}}\|^2\right)\end{aligned} \quad (20)$$

Substituting (12) into (20), equation (20) becomes (21):

$$\dot{L}(t) \le -s^T\left(\frac{s}{\|s\|}\Upsilon + k_P\, sgn(s)\right) + s^T\Gamma + k_M\|s\|\left(\|\widetilde{W}_{M_{FO}}\|\|W_{M_{FO}}\| - \|\widetilde{W}_{M_{FO}}\|^2\right) + k_V\|s\|\left(\|\widetilde{W}_{V_{FO}}\|\|W_{V_{FO}}\| - \|\widetilde{W}_{V_{FO}}\|^2\right) + k_G\|s\|\left(\|\widetilde{W}_{G_{FO}}\|\|W_{G_{FO}}\| - \|\widetilde{W}_{G_{FO}}\|^2\right) + k_F\|s\|\left(\|\widetilde{W}_{F_{FO}}\|\|W_{F_{FO}}\| - \|\widetilde{W}_{F_{FO}}\|^2\right)$$

$$\dot{L}(t) \le -s^T\left(\frac{k_M W_{M_{FO}}^2}{4} + \frac{k_V W_{V_{FO}}^2}{4} + \frac{k_G W_{G_{FO}}^2}{4} + \frac{k_F W_{F_{FO}}^2}{4}\right) + k_M\|s\|\left(\|\widetilde{W}_{M_{FO}}\|\|W_{M_{FO}}\| - \|\widetilde{W}_{M_{FO}}\|^2\right) + k_V\|s\|\left(\|\widetilde{W}_{V_{FO}}\|\|W_{V_{FO}}\| - \|\widetilde{W}_{V_{FO}}\|^2\right) + k_G\|s\|\left(\|\widetilde{W}_{G_{FO}}\|\|W_{G_{FO}}\| - \|\widetilde{W}_{G_{FO}}\|^2\right) + k_F\|s\|\left(\|\widetilde{W}_{F_{FO}}\|\|W_{F_{FO}}\| - \|\widetilde{W}_{F_{FO}}\|^2\right) \quad (21)$$

$$\dot{L}(t) \le -k_M\|s\|\left(\|\widetilde{W}_{M_{FO}}\| - \frac{W_{M_{FO}}}{2}\right)^2 - k_V\|s\|\left(\|\widetilde{W}_{V_{FO}}\| - \frac{W_{V_{FO}}}{2}\right)^2 - k_G\|s\|\left(\|\widetilde{W}_{G_{FO}}\| - \frac{W_{G_{FO}}}{2}\right)^2 - k_F\|s\|\left(\|\widetilde{W}_{F_{FO}}\| - \frac{W_{F_{FO}}}{2}\right)^2$$

$$\dot{L}(t) \le 0 \quad (22)$$

Consequently, $\dot{L}(t) \le 0, s$ tends to zero as $t \to \infty$, which guarantees the global stability of the control system.

V. Simulation and Experimental Results of ARFONNs

A. Simulation Results of ARFONNs

In this section, the ARFONNs will be simulated on Matlab- Simulink for a cleaning and detecting robot (CDR) to demonstrate the effectiveness of the proposed controller. The parameters of the CDR model (Fig. 2) are given as:

$$M_{FO} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}; V_{FO} = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix};$$
$$G_{FO} = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix}; F_{FO} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}; \tau_{FOd} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$$

here: $M_{11} = (m_1 + m_2)l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2\, cos(\delta_2)$

$M_{12} = M_{21} = m_2 l_2^2 + m_2 l_1 l_2\, cos(\delta_2); M_{22} = m_2 l_2^2$

$V_{11} = -m_2 l_1 l_2\, sin(\delta_2)\,\dot{\delta}_2; \quad V_{12} = -m_2 l_1 l_2\, sin(\delta_2)\,(\dot{\delta}_1 + \dot{\delta}_2)$

$V_{21} = m_2 l_1 l_2\, sin(\delta_2)\,\dot{\delta}_1; V_{22} = 0$

$G_1 = m_2 g l_2\, cos(\delta_2)\,(\delta_1 + \delta_2) + (m_1 + m_2) g l_2\, cos(\delta_2)$

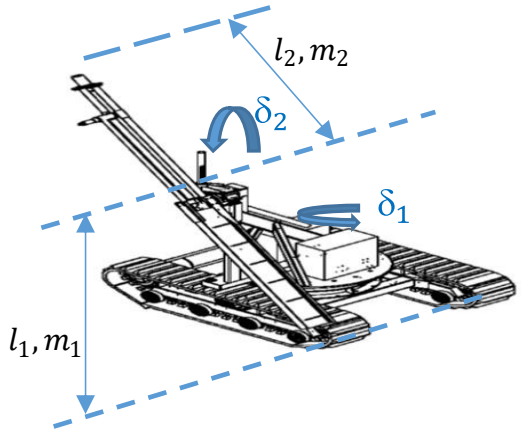

Fig. 2. The cleaning and detecting robot manipulator

$$G_2 = m_2 g l_2\, cos(\delta_2)\,(\delta_1 + \delta_2)$$

$$F_{FO}(\dot{\delta}) = \begin{bmatrix} F_1(\dot{\delta}_1) \\ F_2(\dot{\delta}_2) \end{bmatrix} = \begin{bmatrix} 20\dot{\delta}_1 + 5_{sgn}(\dot{\delta}_1) \\ 20\dot{\delta}_2 + 5_{sgn}(\dot{\delta}_2) \end{bmatrix}; \quad \tau_{FOd} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} 0.5\, sin(20t) \\ 0.5\, sin(20t) \end{bmatrix}$$

where m_1, m_2 are links masses. l_1, l_2 are links lengths; $g = 9{,}8(m/s^2)$ is acceleration of gravity. The CDR parameters are given as:

$$m_1 = 2kg; m_2 = 1kg; l_1 = 0{,}8m; l_2 = 1m$$

The desired position trajectories of the robot will be chosen by: $\delta_d = [\delta_{d1}\ \delta_{d2}]^T = [0.1\, sin(t), 0.1\, sin(t)]^T$ and Initial position joints and initial velocities of joints respectively

$\delta_0 = [\delta_{01}\ \delta_{02}]^T = [0.0\ 0.0]^T(rad)$ and $\dot{\delta}_0 = [\dot{\delta}_{01}\ \dot{\delta}_{02}]^T = [0.0\ 0.0]^T$ (rad/s).

The structure of ARFONNs can be characterized by n=4 nodes. This is based on experience and testing during simulation. The proposed controller parameters are chosen as:

$$k_p = diag[60, 60]; k_d = diag[70, 70];$$
$$k_i = diag[0.2, 0.2]; \beta = 0.7;\ \alpha = 0.02$$

$F_M = F_V = F_G = F_F = 10; k_M = k_V = k_G = k_F = 0.01;$

The mean square error (MSE) of the position tracking response is defined to quantify the respective control performance as:

$$MSE = \frac{1}{T}\sum_{t=1}^{T}[\delta(t) - \delta_d(t)]^2$$

In which, T denotes the total number of sampling instants. The normalized mean square error (NMSE) of the position tracking response, calculated using a per-unit value of 1 rad, is employed to evaluate the control performance.

We simulated the system in two cases. The first simulation case assumes that CDR does not carry a load. The simulated results show in Fig. 3.

The second simulation case, a 0.5 kg payload is added to the mases of the two links CDR, whereas the desired input trajectories and the other parameters are unchanged from the first simulation case.

The simulated result of the proposed ARFONNs, ARNNs [74] and NNs [75] are shown in Fig. 3 and Fig. 4. The simulated comparison NMSE values of controllers are also depicted in Table I. According to these simulated results, the ARFONNs, ARNNs and NNs can provide the good tracking positon. However, the proposed scheme achieves faster convergence of tracking errors compared to the ARNNs and NNs methods. In addition, according to the NMSE values presented in Table I, the proposed ARFONNs controller demonstrates superior position tracking performance compared to the ARNNs, and NNs schemes. The torque control inputs of the ARFONNs, ARNNs, and NNs are illustrated in Fig. 3 for case 1, and Fig. 4 for case 2. In case 1 and 2, the control torque inputs for joint 1 produced by methods show good performance. While the proposed ARFONNs method demonstrates better torque inputs performance for joint 2 under parameter variation conditions compared to ARNNs, and NNs. Additionally, based on tracking errors of the links, it is observed that the ARFONNs exhibit the smallest tracking errors. This indicates that the ARFONNs provide better position tracking compared to the NNs and ARNNs. Furthermore, with the updated parameters in the dynamic structure of the ARFONNs and the adjustment of the number of law nodes, the approximation capability of the ARFONNs is superior to that of the NNs and ARNNs systems. The performance of the approximating functions and estimated parameters is excellent, demonstrating the correctness of the proposed method, including the robustness of the control system parameters.

TABLE I. SIMULATION RESULT COMPARISONS OF ARFONNS, ARNNS AND NNS SCHEMES

NMSE ($x10^{-4}$)	Case 1		Case 2	
	Link1	Link2	Link1	Link2
ARFONNs	0.2254	0.0405	0.4024	0.0862
ARNNs	1.1258	0.0605	1.8956	0.1825
NNs	5.5126	1.7456	11.6281	4.6281

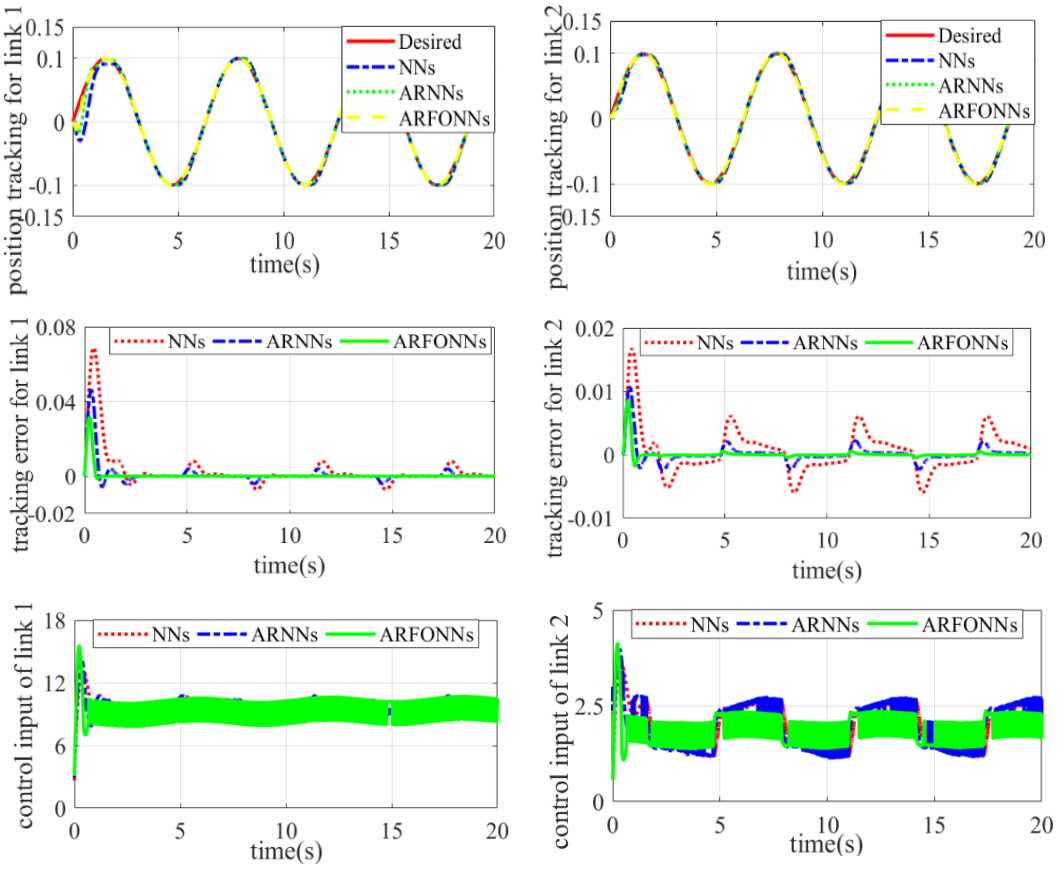

Fig. 3. Simulated results of position tracking, tracking errors, control efforts of NNs, ARNNs and ARFONNs in the first simulated case

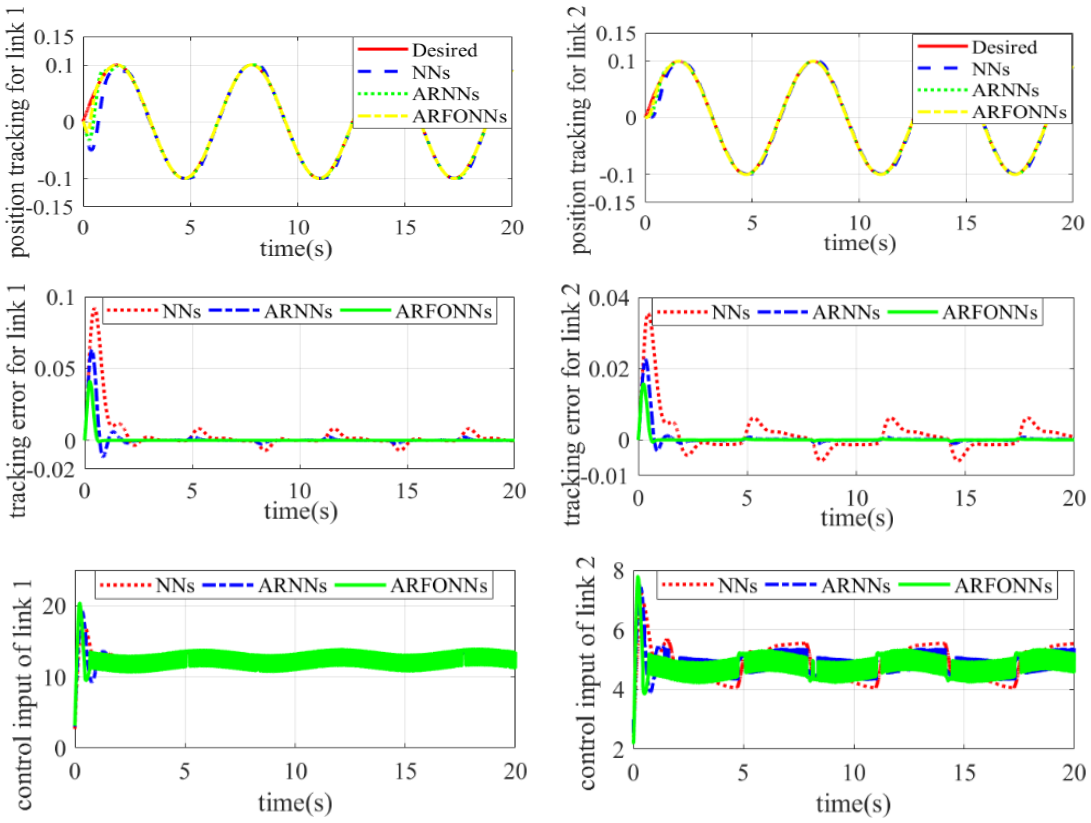

Fig. 4. Simulated results of position tracking, tracking errors, control efforts of NNs, ARNNs and the ARFONNs control system in the second simulated case

B. Experiment results

In this section, we have used the ARFONNs for CDR which are moveable in the condenser water chamber in our lab of intelligent automation technology (Fig. 5). An autonomous robot has been researched and tested to address the issue of condenser cleaning. Condensers are critical components in a wide range of industrial sectors, including thermal power plants, nuclear power plants, and other industrial facilities. A condenser system typically consists of tens of thousands of small-aperture tubes that function as cold sources, helping to lower turbine exhaust temperatures and steam pressure within the thermodynamic cycle, thereby enhancing overall thermal efficiency. Over time, and due to the harsh working environment, these condensers accumulate increasing amounts of dirt and debris. This buildup not only reduces the thermal efficiency of the cycle but can also lead to corrosion of the condenser pipes. As a result, the development of CDR is essential. In this experiment, the electrical control system of the CDR, as illustrated in Fig. 6, is designed based on a distributed control architecture that incorporates a multi-parallel processing system. The central unit of the main controller is connected to several co-processor modules via a CAN bus. These co-processor modules form the internal network responsible for managing the robot's pose structure. This design enables the robot to achieve high-speed information processing and real-time communication capabilities. The CDR robot operates in both manual and automatic modes. In manual mode, an operator controls and supervises the system via remote control and cameras. In automatic mode, the CDRM autonomously verifies operating conditions, locates condenser tube positions, and optimizes cleaning parameters-including pressure, duration, and cycle count-based on input from cameras and sensors (e.g., cooling water, flow, pressure, and flow direction sensors). A high-pressure water jet is then used for cleaning. After each cycle, the system continues to monitor condenser tube cleanliness.

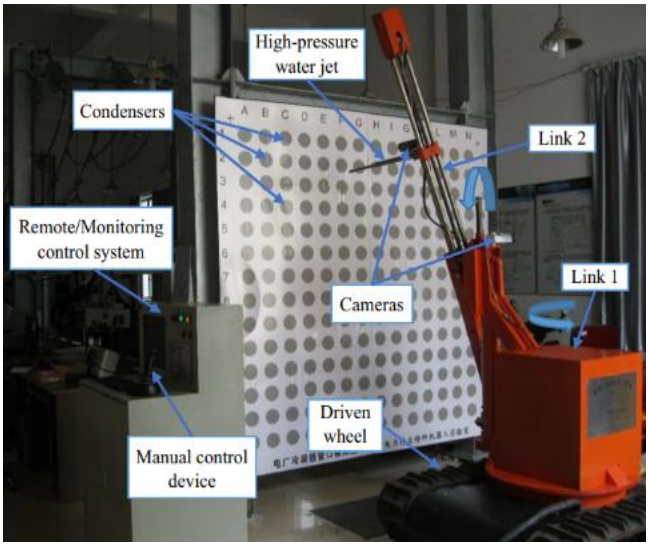

Fig. 5. Mechanical structure of the CDR

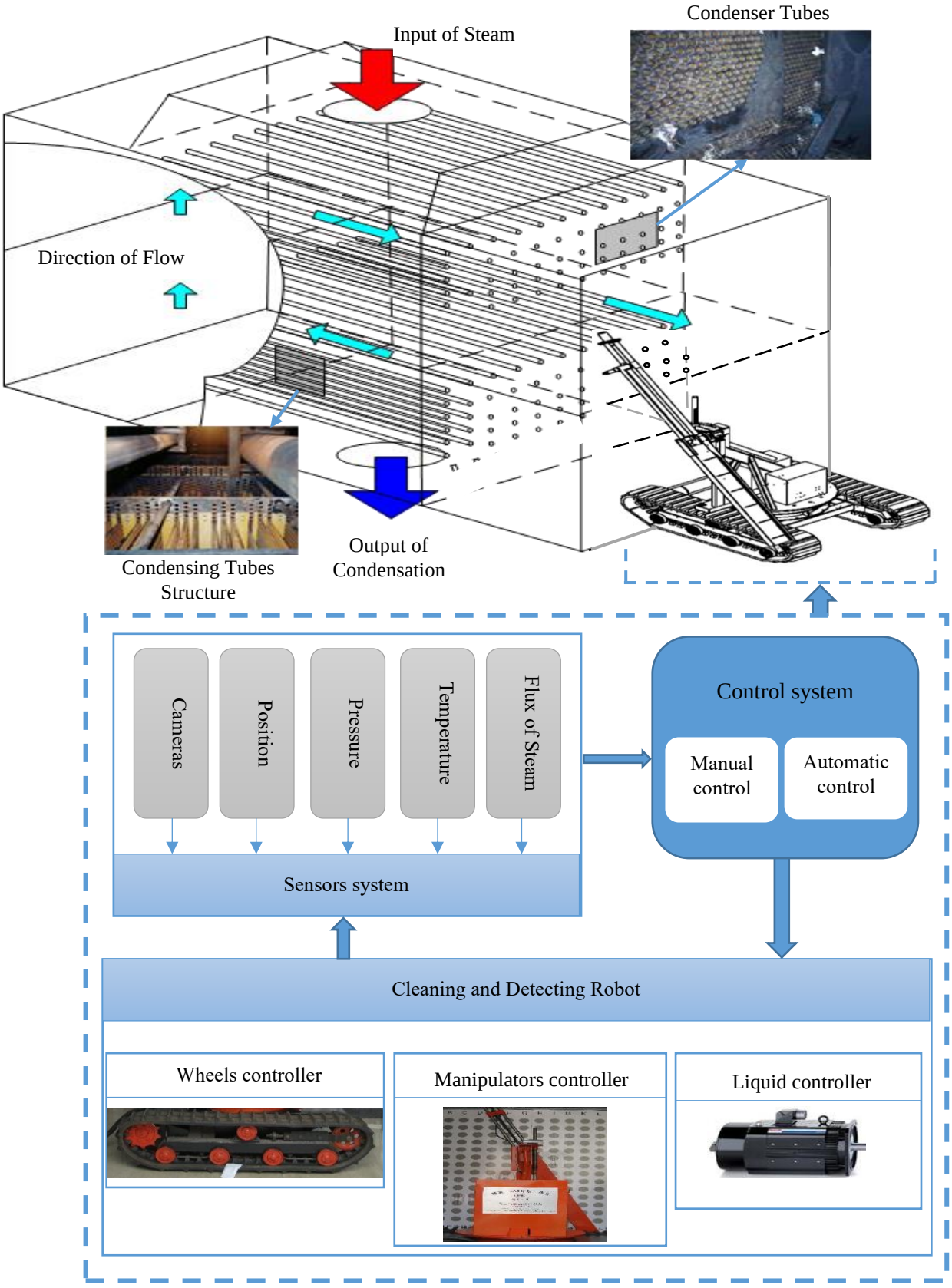

Fig. 6. Electrical control structure of CDR

The first experiment case assumes that CDR does not carry a load. The desired input trajectories and the other parameters are the same as in the first simulation case. The experiment results show in Fig. 7.

The second experiment case assumes that 0.5kg payload is added in the masses of two links CDR, the desired input trajectories and the other parameters are the same as in the first experiment case. The experiment results show in Fig. 8.

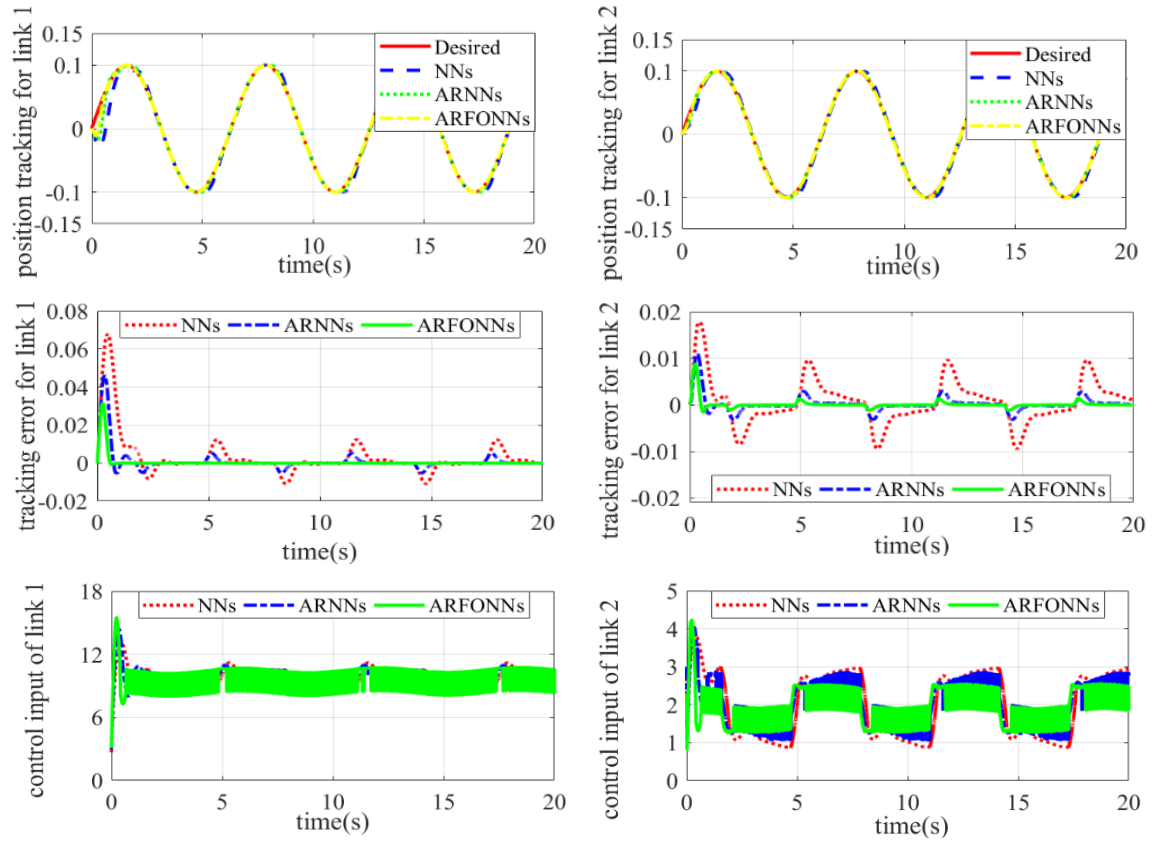

Fig. 7. Experiment results of position tracking, tracking errors, control efforts of NNs, ARNNs and the ARFONNs control system in the first case

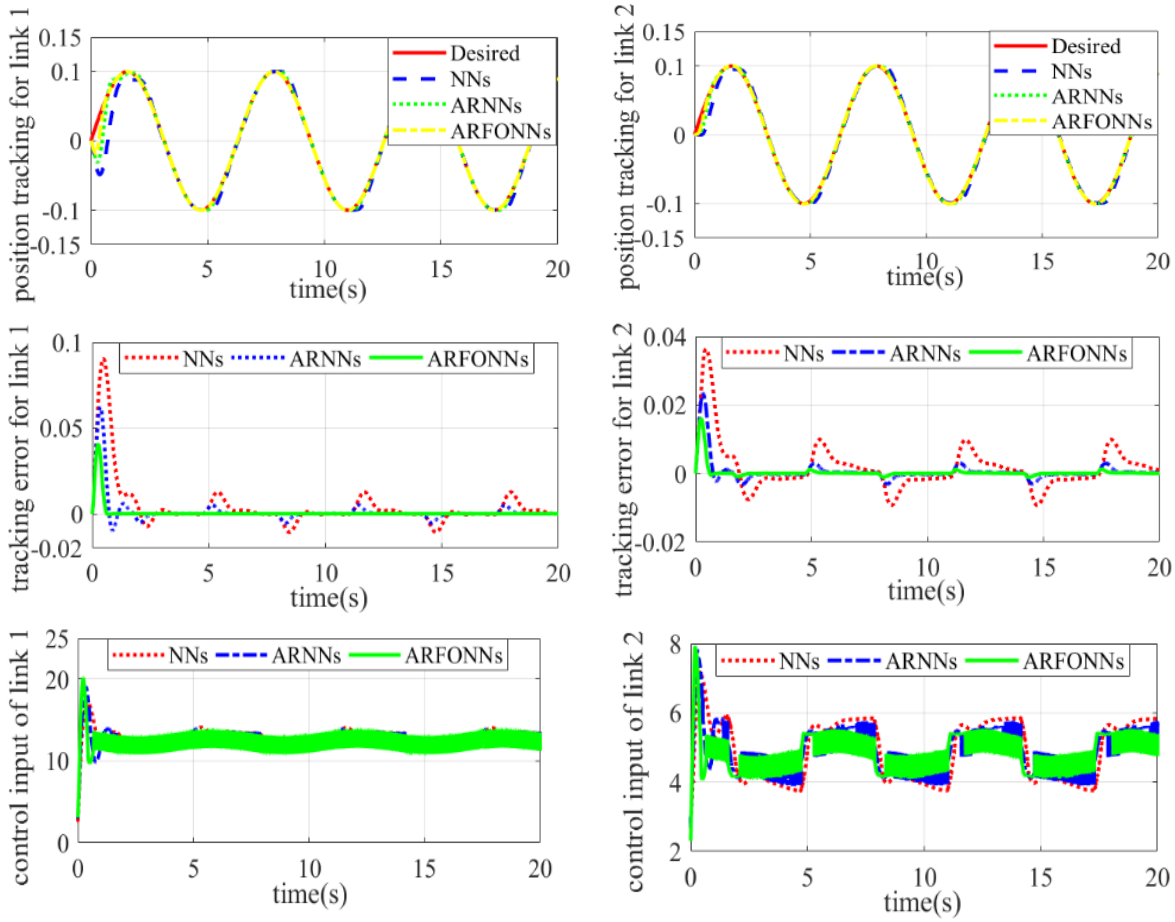

Fig. 8. Experiment results of position tracking, tracking errors, control efforts of NNs, ARNNs and the ARFONNs control system in the second case

In the experimental work, the ARNNs and the proposed control scheme outperform the conventional NNs based system. The performance of the NNs control system is highly dependent on the choice of RBF and initial parameters, necessitating careful selection for optimal outcomes. The first case, Fig. 7 is represented tracking errors and control voltages for the NNs, ARNNs, and the ARFONNs strategies. In the second case, Fig. 8 illustrates the tracking errors and control voltages for the NNs, ARNNs, and ARFONNs strategies. A comparison of these figures shows that while both NNs and ARNNs achieve acceptable tracking performance, the ARFONNs strategy demonstrates faster error convergence. This improvement is also described by the NMSE results in Table II. However, due to dependencies on measurement devices and programming implementation, the results in Table II are not as favorable as those in Table I. Nevertheless, the robustness of the ARFONNs controller under parameter variations is clear evident when comparing the tracking performance and control signals in Fig. 7 and Fig. 8. Both simulation and experimental results confirm that the proposed ARFONNs controller offers superior control performance and robustness compared to the ARNNs and NNs controllers under varying conditions.

TABLE II. EXPERIMENT RESULT COMPARISONS OF ARFONNS, ARNNS AND NNS SCHEMES

NMSE ($x10^{-4}$)	Case 1		Case 2	
	Link1	Link2	Link1	Link2
ARFONNs	0.4586	0.0547	0.8965	0.1852
ARNNs	1.9125	0.1625	2.1754	0.3205
NNs	13.2031	3.9161	17.5781	7.0508

VI. CONCLUSIONS

A robust adaptive fraction odder PID control using neural networks has been proposed to control for CDR. The ARFONNs controller is a combination of the adaptive mothed, Fraction odder PID controller, sliding mode robust term and neural networks to help CDR achieve the high exact position tracking. The unknown dynamic of CDR has been approximated by the RBFNNs and flexible of Fraction odder PID is helped to improve the performance and robustness of the control system. All the parameters of the ARFONNs are defined based on Lyapunov stability theory. From the Simulation and experimental results conducted on the CDR, it is evident that the performance of the ARFONNs control has been significantly improved. The ARFONNs controller can also be put into practice to control other nonlinear systems and uncertain models, such as AC servos and MMR systems. However, the selection of the parameters of the fractional order PID controller like β, α is based on the authors' experience and corrections during simulation and experiment. Therefore, we can use the optimization algorithm to determine those parameters to reduce the amount of calculation.

REFERENCES

[1] P. Chotikunnan and R. Chotikunan, "Dual Design PID Contrller for Robotic Manipulator Application," *Journal of Robotics and Control*, vol. 4, no. 1, pp. 23-34, 2023.

[2] H. R. Nohooji, "constrained neural adaptive PID control for robot manipualtors," *Journal of the Franklin Institute*, vol. 357, no. 7, pp. 3907-3923, 2020.

[3] I. Carlucho, M. D. Paula, and G. G. Acosta, "An adatpvie deep reinforcement learning approach for MIMO PID control of mobile robots," *ISA Transactions*, vol. 102, pp. 280-294, 2020.

[4] J. Lee, P. H. Chang, B. Yu, and M. Jin, "An Adaptive PID Control for Robot Manipulators Under Substantial Payload Variations," in *IEEE Access*, vol. 8, pp. 162261-162270, 2020.

[5] M. Hou, L. Jia, and Z. Wang, "Adaptive Parameter Approaching Law-Based Sliding Mode Control for Wheeled Robots," in *IEEE Access*, vol. 13, pp. 14881-14890, 2025.

[6] Y. Xie, H. Li, S. Wang, S. Zheng, and P. Shi, "Practical Adaptive Backstepping Control for Performance and State Constrained Systems of Cable-Driven Manipulators," in *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 9, pp. 5176-5188, 2024.

[7] Z. Chen, X. Ren, M. Bernabei, V. Mainardi, G. Ciuti, and C. Stefanini, "A Hybrid Adaptive Controller for Soft Robot Interchangeability," in *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 875-882, 2024.

[8] C. Wang, H. Zhan, Q. Guo, and T. Li, "Adaptive Dynamic Programming-Based Fixed-Time Optimal Control for Wheeled Mobile Robot," in *IEEE Robotics and Automation Letters*, vol. 10, no. 1, pp. 176-183, 2025.

[9] K. Lu, S. Han, J. Yang, and H. Yu, "Inverse Optimal Adaptive Tracking Control of Robotic Manipulators Driven by Compliant Actuators," in *IEEE Transactions on Industrial Electronics*, vol. 71, no. 6, pp. 6139-6149, 2024.

[10] Y. Zhang, L. Kong, S. Zhang, X. Yu, and Y. Liu, "Improved Sliding Mode Control for a Robotic Manipulator With Input Deadzone and Deferred Constraint," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 12, pp. 7814-7826, 2023.

[11] R. -D. Xi, X. Xiao, T. -N. Ma, and Z. -X. Yang, "Adaptive Sliding Mode Disturbance Observer Based Robust Control for Robot Manipulators Towards Assembly Assistance," in *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6139-6146, 2022.

[12] H. Hu *et al*., "Robust Adaptive Control of a Bimanual 3T1R Parallel Robot With Gray-Box-Model and Prescribed Performance Function," in *IEEE/ASME Transactions on Mechatronics*, vol. 29, no. 1, pp. 466-475, 2024.

[13] M. Zhang, Z. Zhang, and M. Sun, "Adaptive Tracking Control of Uncertain Robotic Manipulators," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 71, no. 5, pp. 2734-2738, 2024.

[14] L. Wang, W. Sun, S. -F. Su, and X. Zhao, "Adaptive Asymptotic Tracking Control for Flexible-Joint Robots With Prescribed Performance: Design and Experiments," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 6, pp. 3707-3717, 2023.

[15] Z. Liu *et al*., "A Novel Faster Fixed-Time Adaptive Control for Robotic Systems With Input Saturation," in *IEEE Transactions on Industrial Electronics*, vol. 71, no. 5, pp. 5215-5223, 2024.

[16] Y. Hu, H. Yan, H. Zhang, M. Wang, and L. Zeng, "Robust Adaptive Fixed-Time Sliding-Mode Control for Uncertain Robotic Systems With Input Saturation," in *IEEE Transactions on Cybernetics*, vol. 53, no. 4, pp. 2636-2646, 2023.

[17] S. Lu and J. Zhao, "Research on Tracking Control of Circular Trajectory of Robot Based on the Variable Integral Sliding Mode PD Control Algorithm," in *IEEE Access*, vol. 8, pp. 204194-204202, 2020.

[18] Y. Li, L. Zheng, Y. Wang, E. Dong, and S. Zhang, "Impedance Learning-Based Adaptive Force Tracking for Robot on Unknown Terrains," in *IEEE Transactions on Robotics*, vol. 41, pp. 1404-1420, 2025.

[19] S. Chen, W. He, Z. Zhao, Y. Feng, Z. Liu, and K. -S. Hong, "Adaptive Control of a Flexible Manipulator with Unknown Hysteresis and Intermittent Actuator Faults," in *IEEE/CAA Journal of Automatica Sinica*, vol. 12, no. 1, pp. 148-158, 2025.

[20] C. Li, F. Liu, Y. Wang, and M. Buss, "Concurrent Learning-Based Adaptive Control of an Uncertain Robot Manipulator With Guaranteed Safety and Performance," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 5, pp. 3299-3313, 2022.

[21] H. Ma, Q. Zhou, H. Li, and R. Lu, "Adaptive Prescribed Performance Control of A Flexible-Joint Robotic Manipulator With Dynamic Uncertainties," in *IEEE Transactions on Cybernetics*, vol. 52, no. 12, pp. 12905-12915, 2022.

[22] S. Islam and X. P. Liu, "Robust Sliding Mode Control for Robot Manipulators," in *IEEE Transactions on Industrial Electronics*, vol. 58, no. 6, pp. 2444-2453, 2011.

[23] K. Li, R. Wen, "Robust Control of a Walking Robot System and Controller Design," *Procedia Engineering*, vol. 174, pp. 947-955, 2017.

[24] A. Green and J. Z. Sasiadek, "Heuristic design of a fuzzy controller for a flexible robot," in *IEEE Transactions on Control Systems Technology*, vol. 14, no. 2, pp. 293-300, 2006.

[25] S. Kang, P. X. Liu, and H. Wang, "Adaptive Fuzzy Finite-Time Command Filtering Control for Flexible-Joint Robot Systems Against Multiple Actuator Constraints," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 12, pp. 4554-4558, 2023.

[26] Y. Zhu, J. Liu, J. Yu, and Q. -G. Wang, "Command Filtering-Based Adaptive Fuzzy Control of Flexible-Joint Robots With Time-Varying Full-State Constraints," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 71, no. 2, pp. 682-686, 2024.

[27] S. Xu and B. He, "Robust Adaptive Fuzzy Fault Tolerant Control of Robot Manipulators With Unknown Parameters," in *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 9, pp. 3081-3092, 2023.

[28] K. Li and S. Dong, "Predefined-Time Event-Triggered Adaptive Fuzzy Formation Control for Nonholonomic Multirobot Systems," in *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 12, pp. 6731-6743, 2024.

[29] C. Zhu, C. Yang, Y. Jiang, and H. Zhang, "Fixed-Time Fuzzy Control of Uncertain Robots With Guaranteed Transient Performance," in *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 3, pp. 1041-1051, 2023.

[30] G. Lin, J. Yu, and J. Liu, "Adaptive Fuzzy Finite-Time Command Filtered Impedance Control for Robotic Manipulators," in *IEEE Access*, vol. 9, pp. 50917-50925, 2021.

[31] Y. Jiang, Y. Wang, Z. Miao, J. Na, Z. Zhao, and C. Yang, "Composite-Learning-Based Adaptive Neural Control for Dual-Arm Robots With Relative Motion," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 3, pp. 1010-1021, 2022.

[32] S. -J. Kim, M. Jin, and J. -H. Suh, "A Study on the Design of Error-Based Adaptive Robust RBF Neural Network Back-Stepping Controller for 2-DOF Snake Robot's Head," in *IEEE Access*, vol. 11, pp. 23146-23156, 2023.

[33] Z. Chen, Y. Liu, W. He, H. Qiao, and H. Ji, "Adaptive-Neural-Network-Based Trajectory Tracking Control for a Nonholonomic Wheeled Mobile Robot With Velocity Constraints," in *IEEE Transactions on Industrial Electronics*, vol. 68, no. 6, pp. 5057-5067, 2021.

[34] G. Li, X. Chen, J. Yu, and J. Liu, "Adaptive Neural Network-Based Finite-Time Impedance Control of Constrained Robotic Manipulators With Disturbance Observer," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 3, pp. 1412-1416, 2022.

[35] G. D. Khan, "Adaptive Neural Network Control Framework for Industrial Robot Manipulators," in *IEEE Access*, vol. 12, pp. 63477-63483, 2024.

[36] F. Wang, Y. Yu, X. Li, J. Luo, and J. Zhong, "Adaptive Neural Network Synchronous Tracking Control for Teleoperation Robots Under Event-Triggered Mechanism," in *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 10010-10017, 2024.

[37] H. Ma, H. Ren, Q. Zhou, H. Li, and Z. Wang, "Observer-Based Neural Control of N-Link Flexible-Joint Robots," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 4, pp. 5295-5305, 2024.

[38] B. Zhang, T. Wu, and T. Wang, "Adaptive Neural Network Control for Exoskeleton Motion Rehabilitation Robot With Disturbances and Uncertain Parameters," in *IEEE Access*, vol. 11, pp. 75497-75507, 2023.

[39] Z. Liu, O. Zhang, Y. Gao, Y. Zhao, Y. Sun, and J. Liu, "Adaptive Neural Network-Based Fixed-Time Control for Trajectory Tracking of Robotic Systems," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 1, pp. 241-245, 2023.

[40] C. Zhu, Y. Jiang, and C. Yang, "Fixed-Time Neural Control of Robot Manipulator With Global Stability and Guaranteed Transient Performance," in *IEEE Transactions on Industrial Electronics*, vol. 70, no. 1, pp. 803-812, 2023.

[41] J. Su, H. -K. Lam, J. Liu, and J. Yu, "Fuzzy Observer-Based Command Filtered Adaptive Control of Flexible Joint Robots With Time-Varying Output Constraints," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 71, no. 9, pp. 4251-4255, 2024.

[42] S. Diao, W. Sun, S. -F. Su, and J. Xia, "Adaptive Fuzzy Event-Triggered Control for Single-Link Flexible-Joint Robots With Actuator Failures," in *IEEE Transactions on Cybernetics*, vol. 52, no. 8, pp. 7231-7241, 2022.

[43] M. Van, Y. Sun, S. Mcllvanna, M. -N. Nguyen, M. O. Khyam, and D. Ceglarek, "Adaptive Fuzzy Fault Tolerant Control for Robot Manipulators With Fixed-Time Convergence," in *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 9, pp. 3210-3219, 2023.

[44] S. Ling, H. Wang, and P. X. Liu, "Adaptive Fuzzy Tracking Control of Flexible-Joint Robots Based on Command Filtering," in *IEEE Transactions on Industrial Electronics*, vol. 67, no. 5, pp. 4046-4055, 2020.

[45] S. Bian, J. M. Garibaldi, and Z. Li, "Reshaping Wearable Robots Using Fuzzy Intelligence: Integrating Type-2 Fuzzy Decision, Intelligent Control, and Origami Structure," in *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 11, pp. 3741-3761, 2023.

[46] F. Zhang and P. Huang, "Fuzzy-Based Adaptive Super-Twisting Sliding-Mode Control for a Maneuverable Tethered Space Net Robot," in *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 7, pp. 1739-1749, 2021.

[47] Y. Pan, P. Du, H. Xue, and H. -K. Lam, "Singularity-Free Fixed-Time Fuzzy Control for Robotic Systems With User-Defined Performance," in *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 8, pp. 2388-2398, 2021.

[48] J. Zhou, E. Liu, X. Tian, and Z. Li, "Adaptive Fuzzy Backstepping Control Based on Dynamic Surface Control for Uncertain Robotic Manipulator," in *IEEE Access*, vol. 10, pp. 23333-23341, 2022.

[49] W. Chang, Y. Li, and S. Tong, "Adaptive Fuzzy Backstepping Tracking Control for Flexible Robotic Manipulator," in *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 12, pp. 1923-1930, 2021.

[50] M. Van and S. S. Ge, "Adaptive Fuzzy Integral Sliding-Mode Control for Robust Fault-Tolerant Control of Robot Manipulators With Disturbance Observer," in *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 5, pp. 1284-1296, 2021.

[51] Y. Li, S. Dong, and K. Li, "Fixed-Time Command Filter Fuzzy Adaptive Formation Control for Nonholonomic Multirobot Systems With Unknown Dead-Zones," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 11, pp. 17305-17316, 2024.

[52] M. Zhai, S. Diao, T. Yang, Q. Wu, Y. Fang, and N. Sun, "Adaptive Fuzzy Control for Underactuated Robot Systems With Inaccurate Actuated States and Unavailable Unactuated States," in *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 1566-1578, 2025.

[53] W. Yuan, Y. -H. Liu, C. -Y. Su, and F. Zhao, "Whole-Body Control of an Autonomous Mobile Manipulator Using Model Predictive Control and Adaptive Fuzzy Technique," in *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 3, pp. 799-809, 2023.

[54] B. M. Yilmaz, E. Tatlicioglu, A. Savran, and M. Alci, "Self-Adjusting Fuzzy Logic Based Control of Robot Manipulators in Task Space," in *IEEE Transactions on Industrial Electronics*, vol. 69, no. 2, pp. 1620-1629, 2022.

[55] G. Chen, Y. Jiang, K. Guo, and L. Wang, "Speed Tracking Control for Unmanned Driving Robot Vehicle Based on Fuzzy Adaptive Sliding Mode Control," in *IEEE Transactions on Vehicular Technology*, vol. 71, no. 12, pp. 12617-12625, Dec. 2022.

[56] L. Teng, M. A. Gull, and S. Bai, "PD-Based Fuzzy Sliding Mode Control of a Wheelchair Exoskeleton Robot," in *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 5, pp. 2546-2555, 2020.

[57] X. Yin, L. Pan, and S. Cai, "Robust adaptive fuzzy sliding mode trajectory tracking control for serial robotic manipulators," *Robotics and Computer-Integrated Manufacturing*, vol. 72, p. 101884, 2021.

[58] C. Yang, G. Peng, L. Cheng, J. Na, and Z. Li, "Force Sensorless Admittance Control for Teleoperation of Uncertain Robot Manipulator Using Neural Networks," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 5, pp. 3282-3292, 2021.

[59] G. Chen, J. Jiang, L. Wang, and W. Zhang, "Clutch Mechanical Leg Neural Network Adaptive Robust Control of Shift Process for Driving Robot With Clutch Transmission Torque Compensation," in *IEEE*

Transactions on Industrial Electronics, vol. 69, no. 10, pp. 10343-10353, 2022.

[60] Q. Yang, F. Zhang, and C. Wang, "Deterministic Learning-Based Neural PID Control for Nonlinear Robotic Systems," in *IEEE/CAA Journal of Automatica Sinica*, vol. 11, no. 5, pp. 1227-1238, 2024.

[61] X. Wang, B. Xu, Y. Cheng, H. Wang, and F. Sun, "Robust Adaptive Learning Control of Space Robot for Target Capturing Using Neural Network," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 10, pp. 7567-7577, 2023.

[62] Z. Cui, J. Li, W. Li, X. Zhang, P. W. Y. Chiu, and Z. Li, "Fast Convergent Antinoise Dual Neural Network Controller With Adaptive Gain for Flexible Endoscope Robots," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 7, pp. 9095-9108, July 2024.

[63] D. X. Ba, N. T. Thien, and J. Bae, "A Novel Iterative Second-Order Neural-Network Learning Control Approach for Robotic Manipulators," in *IEEE Access*, vol. 11, pp. 58318-58332, 2023.

[64] N. Hassan and A. Saleem, "Neural Network-Based Adaptive Controller for Trajectory Tracking of Wheeled Mobile Robots," in *IEEE Access*, vol. 10, pp. 13582-13597, 2022.

[65] C. Liu, G. Peng, K. Zhao, J. Li, and C. Yang, "Neural Learning-Based Adaptive Force Tracking Control for Robots With Finite-Time Prescribed Performance Under Varying Environments," in *IEEE Transactions on Industrial Electronics*, vol. 71, no. 12, pp. 16338-16347, 2024.

[66] Y. Fan, C. Yang, H. Zhan, and Y. Li, "Neuro-Adaptive-Based Predefined-Time Smooth Control for Manipulators With Disturbance," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 54, no. 8, pp. 4605-4616, 2024.

[67] H. -T. Nguyen and C. C. Cheah, "Analytic Deep Neural Network-Based Robot Control," in *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 4, pp. 2176-2184, 2022.

[68] P. SharafianArdakani, M. A. Hanafy, I. Kondaurova, A. Ashary, M. M. Rayguru, and D. O. Popa, "Adaptive User Interface With Parallel Neural Networks for Robot Teleoperation," in *IEEE Robotics and Automation Letters*, vol. 10, no. 2, pp. 963-970, 2025.

[69] X. Liang, Z. Yao, W. Deng, and J. Yao, "Adaptive Neural Network Finite-Time Tracking Control for Uncertain Hydraulic Manipulators," in *IEEE/ASME Transactions on Mechatronics*, vol. 30, no. 1, pp. 645-656, 2025.

[70] Q. Liu, D. Li, S. S. Ge, R. Ji, Z. Ouyang, and K. P. Tee, "Adaptive bias RBF neural network control for a robotic manipulator," *Neurocomputing*, vol. 447, pp. 213-223, 2021.

[71] C. Sun, W. He, W. Ge, and C. Chang, "Adaptive Neural Network Control of Biped Robots," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 2, pp. 315-326, 2017.

[72] B. Li and X. Zhao, "Neural Network-Based Adaptive Sliding Mode Control for T-S Fuzzy Fractional Order Systems," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 12, pp. 4549-4553, 2023.

[73] T. Kathuria, V. Kumar, K. P. S. Rana, and A. T. Azar, "Control of a three link manipulator Using Fractional Order PID controller," *Fractional Order Systems*, pp.477-510, 2018

[74] P. Van Cuong and W. Y. Nan, "Adaptive trajectory tracking neural network control with robust compensator for robot manipulators," *Neural Comput & Applic*, vol. 27, pp. 525–536, 2016.

[75] M. Sato and K. Ishii, "A neural network based controller for a wheel type mobile robot," *International congress series*, vol. 1291, pp. 261-264, 2006.

[76] Y. H. Kim and F. L. Lewis, "Neural network output feedback control of robot manipulators," in *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 301-309, April 1999.

[77] J. Wu, Z. Xu, Y. Zhang, C.-Y. Su, and Y. Wang, "Modeling and tracking control of dielectric elastomer actuators based on fractional calculus," *ISA Trans*., vol. 138, pp. 687–695, Jul. 2023.

[78] X. Yi, R. Guo, and Y. Qi, "Stabilization of chaotic systems with both uncertainty and disturbance by the UDE-based control method," *IEEE Access*, vol. 8, pp. 62471–62477, 2020.

[79] R. Peng, C. Jiang, and R. Guo, "Stabilization of a class of fractional order systems with both uncertainty and disturbance," *IEEE Access*, vol. 9, pp. 42697–42706, 2021.

[80] N. Ullah, F. Nisar, and A. A. Alahmadi, "Closed loop control of photo voltaic emulator using fractional calculus," *IEEE Access*, vol. 8, pp. 28880–28887, 2020.

[81] N. Zhuo-Yun, Z. Yi-Min, W. Qing-Guo, L. Rui-Juan, and X. Lei-Jun, "Fractional-order PID controller design for time-delay systems based on modified Bode's ideal transfer function," *IEEE Access*, vol. 8, pp. 103500–103510, 2020.

[82] J. Muñoz, R. de Santos-Rico, L. Mena, and C. A. Monje, "Humanoid head camera stabilization using a soft robotic neck and a robust fractional order controller," *Biomimetics*, vol. 9, no. 4, 2024.

[83] M. Iskandar, P. Azahar, A. Irawan, and R. M. T. R. Ismail, "Adjustable convergence rate prescribed performance with fractional-order PID controller for servo pneumatic actuated robot positioning," *Cogn. Robot.*, vol. 3, pp. 93–106, 2023.

[84] H. Fan, H. Wei, D. Xu, and Y. Liu, "Closed-Loop Iterative Optimized Fractional-Order PID Current Control of PMSM," in *IEEE Transactions on Industrial Informatics*, vol. 21, no. 2, pp. 1120-1129, Feb. 2025.

[85] W. Adris Shutnan *et al.*, "Modeling and Control of a 3DOF Robot Manipulator Using Artificial Fuzzy-Immune FOPID Controller," in *IEEE Access*, vol. 12, pp. 153074-153088, 2024.

[86] P. Chen, H. Gan, Y. Liu, and Y. Luo, "Active Disturbance Rejection Fractional-Order Independent Control of Time Delay Systems With Application to Air Floating Motion Platform," in *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1-10, 2024.