Journal of Electrical Engineering, Vol. 77, No. 1, 2026, pp. 33-40
https://reference-global.com/journal/JEE

Performance evaluation of an adaptive fractional-order fuzzy PID controller on three-link robotic manipulator system

Sweta, Vinay Kumar Deolia, Jitendra Kumar

Because of their adaptability, three-link robotic manipulator (TLRM) devices are frequently used in industrial automation; yet, their operational complexity makes precise control difficult. In order to improve trajectory tracking along the X and Y directions, an adaptive fractional-order fuzzy PID (AFOFPID) controller is suggested. The optimisation criterion is a weighted mixture of the integral of absolute error (IAE) and the integral of absolute changes in controller output (IACCO), and the controller parameters are adjusted using the Cuckoo Search Algorithm (CSA). To manage changes in system dynamics, the AFOFPID combines a fuzzy logic structure with an adaptive mechanism, and the fractional-order component enhances stability and robustness even further.

Keywords: AFOFPID, CSA, three-link robotic manipulator (TLRM), IAE, fuzzy logic control

1 Introduction and literature assessment

Three-link robust robotic manipulators (TLRM) systems are nonlinear in nature, meaning that controlling them is still a major challenge because of this. While classical proportional-integral-derivative (PID) controllers account for around 90% of applications in industrial process control, they are not always suitable for complicated, nonlinear systems such as robotic manipulators. According to a review by Kumar et al. (2020), and Gupta et al. (2024) conventional PID controllers are mostly useful in linear systems and have limitations when it comes to regulating nonlinear systems. Due to this, sophisticated control schemes like fuzzy PID (FPID) have been developed to meet the demands of increasingly complicated systems [1-4, 5, 6].

The literature has examined several nonlinear control strategies for managing complicated nonlinear systems, including sliding mode control (SMC) and model reference adaptive control (MRAC), self-tuning regulators, and gain scheduling. However, as demonstrated by techniques like SMC, MRAC, and self-tuning regulators, these strategies frequently need a precise mathematical model of the plant in order to be designed. Furthermore, gain scheduling can become unnecessarily complicated in systems with several operating points, which makes it challenging to apply in nonlinear systems with wide operating ranges. As such, these methods might not provide the best control possible for systems such as robotic manipulators in both servo and regulatory mode operations [5].

Researchers have looked for more efficient control strategies for nonlinear systems in response to these difficulties. A new paradigm for intelligent control was established by Zadeh with the introduction of fuzzy logic. By utilising linguistic variables, fuzzy logic controllers (FLCs) offer a powerful tool for creating controllers that function on the basis of expert knowledge and logical reasoning as opposed to exact mathematical models. The language synthesis of fuzzy controllers was first accomplished by Mamdani et al., who reported the first successful deployment of an FLC on a laboratory-scale process. The main benefit of FLCs is that they can operate without a precise mathematical structure of the TLRM system, which is one of the reasons why the technical community has come to accept them so widely [7, 8]. Feedback-controlled complicated nonlinear systems perform better when their control technique is optimised. Scholars are always trying to come up with clever adaptive control schemes that can handle both regulatory and servo functions in robotic manipulator systems. Here are a few noteworthy new studies in this field [9]. Fuzzy logic was used by Mudi and Paul to create robust self-tuning fuzzy PD (STFPD) and fuzzy PI (STFPI) controllers that dynamically adjusted gain in real-time based on error and the rate of error change. This approach is built on the adaptive nature of fuzzy control. Tested on both linear and nonlinear second-order systems, these controllers showed significant gains in control efficacy over traditional fuzzy logic controllers (FLCs) [10]. By incorporating a robust self-tuning fuzzy PD+I controller, Malki et al. (2024) improved adaptive control even further. This controller was created

ECE Department, GLA University, Mathura (281406), India
sweta.gla@gla.ac.in, vinaykumar.deolia@gla.ac.in, singhjitendra86@gmail.com

https://doi.org/10.2478/jee-2026-0004, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X

to manage a flexible joint robotic arm under different load uncertainty. In contrast to conventional controllers, the controller's gains were changed in real-time as a non-linear function of the input signals, improving resilience, response time, and lowering overshoot [11].

The substantial potential that adaptive fractional-order fuzzy PID (AFOFPID) controllers represent for the industrial robotics industry is highlighted by this study. Fractional-order controllers' (FOCs) advantages are highlighted in the literature study, especially the additional degrees of freedom (DOF) they provide when PID control may not be sufficient. Adding adaptive methods to fuzzy systems – like self-tuning, for example – improves FOC performance even more. Control gets complicated because the system is nonlinear and MIMO-coupled. On the other hand, adaptive fuzzy logic improves the AFOFPID controller's ability to effectively reduce outside noise and sensor disturbances while following a trajectory. Numerous experiments show that the AFOFPID controller outdoes FOFPID, FOPID, and PID controllers in terms of trajectory tracking and positional movement in the X-axis and Y-axis for TLRMS for angular position.

This document is organised as follows: An introduction and overview of the literature assessment are given in Section 1, and problem formulation of TLRM system is covered in Section 2. The recommended controller structure design and its tuning using the Cuckoo Search Algorithm (CSA) are presented in Section 3, whereas Section 4 discusses a thorough analysis of various controllers of the simulation results and analysis. The study is finally concluded in Section 5.

2 Dynamic system modelling

This section describes a TLRM system with 3-DOF, as seen in Fig. 1. A frictionless ball bearing is used to join the second link (L2) to the end of the first link (L1), and a frictionless pivot is used to install the L1 on a stationary base. In a similar manner, frictionless ball bearings are used to link the L3 and L2. For further explanation to guarantee smooth operation and decrease wear, the system uses ball bearings at each joint and frictionless pivots. While the second and third links are joined by frictionless ball bearings that enable smooth rotation, the first link is fixed to a hard foundation. This design minimises maintenance, lowers resistance, and improves precision. In order to keep the robotic manipulator system accurate and effective, frictionless components must be used.

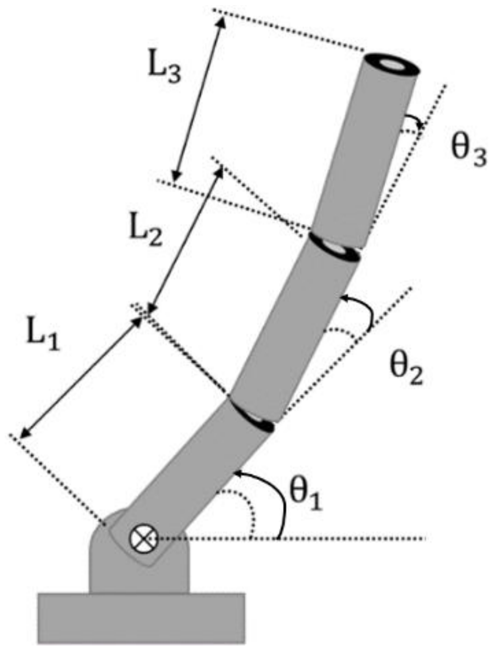


Fig. 1. Diagram of a TLRM System [4]

Table 1. Mass (kg), length (m) and gravity (m/s^2) parameters used in TLRM system [4]

Parameters	Units	L1	L2	L3
Mass	kg	m_1: 0.1	m_2: 0.1	m_3: 0.1
Length	m	L_1: 0.8	L_2: 0.4	L_3: 0.4
Gravity	m/s^2	9.8		

The state space modelling of TLRM System is shown in Eqn. (1) [12, 13].

$$\begin{bmatrix} N_{11} & N_{12} & N_{13} \\ N_{21} & N_{22} & N_{23} \\ N_{31} & N_{32} & N1_{33} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} + \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \tag{1}$$

In Eqn. (1), $\ddot{\theta}_i$ is the second-order derivative of the angular position for three links, $\dot{\theta}_i^2$ is a centrifugal term in which $i = 1, 2, 3$. $\dot{\theta}_i \dot{\theta}_j$ is the Coriolis term in which $(i \neq j)$, and θ_i is differentiation of potential energy stored in links.

3 AFOFPID controller design and tuning of controller gains using CSA

The purpose of the AFOFPID controller is to improve control in robotic systems by utilising fuzzy logic to integrate adaptive features. With the use of a adaptive fuzzy logic mechanism, it integrates the outputs of fuzzy PD of fractional order (FOFPD) and fuzzy PI of fractional order (FOFPI) to enable real-time modifications to output gains. This flexibility is attained by altering a parameter α in the range of 0 to 1, which enables the controller to react quickly to shifting system circumstances. For L1, L2, and L3, the design includes five gains for each: input and output gains. A similar structure is applied to the L1, L2, and L3.

Conventional PID controllers have several drawbacks, including the assumption of a linear relationship between input and output, which degrades performance in nonlinear systems; fixed parameters, which increases the controller's sensitivity to noise and decreases its adaptability to changing dynamics; FOPID controllers, which incorporate fuzzy logic but can be difficult to tune; and FOFPID controllers, which incorporate fuzzy logic but increase system complexity and may not dynamically adjust to changing conditions [14-16].

On the other hand, the AFOFPID controller improves upon these drawbacks by providing real-time flexibility and improved trajectory tracking and disturbance rejection capabilities. Because it dynamically modifies the parameter α in response to the state of the system, it is appropriate for both nonlinear along with MIMO systems.

A method for approximating fractional-order transfer functions is the Oustaloup method. In order to facilitate simpler implementation and analysis, it entails representing a differential equation of fractional order as a sequence of terms of integer order. Through the use of fractional calculus techniques, the AFOFPID controller can improve control strategies while preserving the advantages of conventional integer-order PID control. Because of this, systems exhibiting fractional dynamics perform better, which makes it especially helpful for intricate robotic applications.

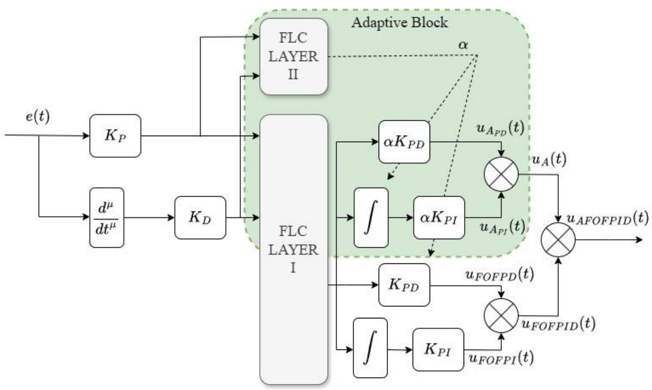


Fig. 2. AFOFPID control structure

The general mathematical formulation of traditional FOPI controller for the links of TLRMS is presented in the velocity form as [17]

$$\frac{d^\mu u(t)}{dt^\mu} = K_P \frac{d^\mu e(t)}{dt^\mu} + K_I e(t) \tag{2}$$

where proportional and integrals gains are K_P and K_I, respectively. The error and controller output of the FOPI controller is given as $e(t)$ and $u(t)$, respectively along with the fractional order differentiator term μ.

Equation (2) can be changed as

$$K_P e(t) + K_D \frac{d^\mu e(t)}{dt^\mu} = \frac{d^\mu u(t)}{dt^\mu} \tag{3}$$

A new gain constant K'_{PI} is added on the right-hand side in the above equation for enhanced degree of freedom for total control.

$$K_P e(t) + K_D \frac{d^\mu e(t)}{dt^\mu} = K'_{PI} \frac{d^\mu u(t)}{dt^\mu} \tag{4}$$

In the above equation the right-hand side shows the formulation of FOFPI controller for link of TLRMS. The error term $K_P e(t)$ and rate change in error term $K_D \frac{d^\mu e(t)}{dt^\mu}$ are the input to the FOFPI controller and the output is given as $K'_{PI} \frac{d^\mu u(t)}{dt^\mu}$. Again, reorganizing Eqn. (4) gets

$$K_P e(t) + K_D \frac{d^\mu e(t)}{dt^\mu} = u'_{FOFPI}(t) \tag{5}$$

where

$$u'_{FOFPI}(t) = K'_{PI} \frac{d^\mu u(t)}{dt^\mu} \tag{6}$$

The equation shows the control action of FOFPI controller. In order to achieve the output from the FOFPI controller a FOI term is utilized. Hence, the absolute output from the FOFPI controller is yielded as below

$$u_{FOFPI}(t) = K'_{PI} \left(\frac{d^{-\lambda}}{dt^{-\lambda}} \left(\frac{d^\mu u(t)}{dt^\mu} \right) \right) \tag{7}$$

where, λ is the FOI term.

The general mathematical formulation of traditional FOPD controller for the links of TLRMS is presented as

$$u(t) = K_P e(t) + K_D \frac{d^\mu u(t)}{dt^\mu} \tag{8}$$

The error and controller output of the FOPD controller are given as $e(t)$ and $u(t)$ along with the fractional order differentiator term μ.

Equation (8) can be changed as

$$K_P e(t) + K_D \frac{d^\mu u(t)}{dt^\mu} = u(t) \tag{9}$$

A new gain constant K'_{PD} is added in the right-hand side in Eqn. (9) for enhanced degree of freedom for total control

$$K_P e(t) + K_D \frac{d^\mu e(t)}{dt^\mu} = K'_{PD} u(t) \tag{10}$$

In the above equation right-hand side shows the formulation of FOFPD controller for link of TLRMS. The error term $K_P e(t)$ and rate change in error term $K_D \frac{d^\mu e(t)}{dt^\mu}$ are the input to the FOFPD controller and the output is given as $K'_{PD} u(t)$. Again, reorganizing Eqn. (10) gets

$$K_P e(t) + K_D \frac{d^\mu e(t)}{dt^\mu} = u_{FOFPD}(t) \tag{11}$$

where

$$u_{FOFPD}(t) = K'_{PD} u(t) \tag{12}$$

Equation (12) shows the control action of FOFPD controller.

It is observed from equation (2) to equation (12) that the mathematical formulation of the velocity form of the FOFPI controller and the position form of the FOFPD controller are fairly comparable resulting only difference in the output term. The control signal is incremental in velocity form and direct in position form. As shown in Fig. 2, the shared architecture of the FOFPI and FOFPD controllers was used to create the control action in order to simplify the fuzzy controller's design. Therefore, the FOFPI and FOFPD controllers can be combined in parallel to create the FOFPID controller, as shown in Fig. 2. The AFOFPID controller's final design equation is as follows:

$$K_P e(t) + K_D \frac{d^\mu u(t)}{dt^\mu} = u_{FOFPD}(t) + u_{FOFPI}(t) \tag{13}$$

Re-organizing Eqn. (13) gives

$$K_P e(t) + K_D \frac{d^\mu u(t)}{dt^\mu} = \\ = K'_{PD} u(t) + K'_{PI} \left(\frac{d^{-\lambda}}{dt^{-\lambda}} \left(\frac{d^\mu u(t)}{dt^\mu} \right) \right) \tag{14}$$

$$K_P e(t) + K_D \frac{d^\mu u(t)}{dt^\mu} = u_{FOFPID}(t) \tag{15}$$

where proportional and derivative gains are K_P and K_D as input gains and K'_{PD} and K'_{PI} are output gains of the FOFPID controller, respectively. $K_P e(t)$ and $K_D \frac{d^\mu u(t)}{dt^\mu}$ are inputs to the fuzzy controller whereas $u_{FOFPID}(t)$ is the combined output of FOFPD and FOFPI controllers as follows.

$$u_{FOFPID}(t) = u_{FOFPD}(t) + u_{FOFPI}(t) \tag{16}$$

$$u_{FOFPID}(t) = \\ = K'_{PD} u(t) + K'_{PI} \left(\frac{d^{-\lambda}}{dt^{-\lambda}} \left(\frac{d^\mu u(t)}{dt^\mu} \right) \right) \tag{17}$$

To design AFOFPID controller, adaptive block is included with the output gains of fuzzy controller, so re-organizing equation (17) as follows,

$$K_P e(t) + K_D \frac{d^\mu u(t)}{dt^\mu} = \\ = \alpha K_{PD} u(t) + \alpha K_{PI} \left(\frac{d^{-\lambda}}{dt^{-\lambda}} \left(\frac{d^\mu u(t)}{dt^\mu} \right) \right) \tag{18}$$

where $K'_{PI} = \alpha K_{PI}$ and $K'_{PD} = \alpha K_{PD}$. The adaptive value α is dynamical at runtime using adaptive block of FLC-II which contributes to the adaptive nature of AFOFPID controller.

Adaptive application on output gains of fuzzy controller is given as algebraic sum as follows

$$u_{A_{PD}}(t) + u_{A_{PI}}(t) = u_A(t) \tag{19}$$

After combining the effect of α on the output gains, gives a novel AFOFPID controller as shown in Eqn. (19).

$$u_{FOFPID}(t) + u_A(t) = \\ = \alpha K_{PD} u(t) + \alpha K_{PI} \left(\frac{d^{-\lambda}}{dt^{-\lambda}} \left(\frac{d^\mu u(t)}{dt^\mu} \right) \right) \tag{20}$$

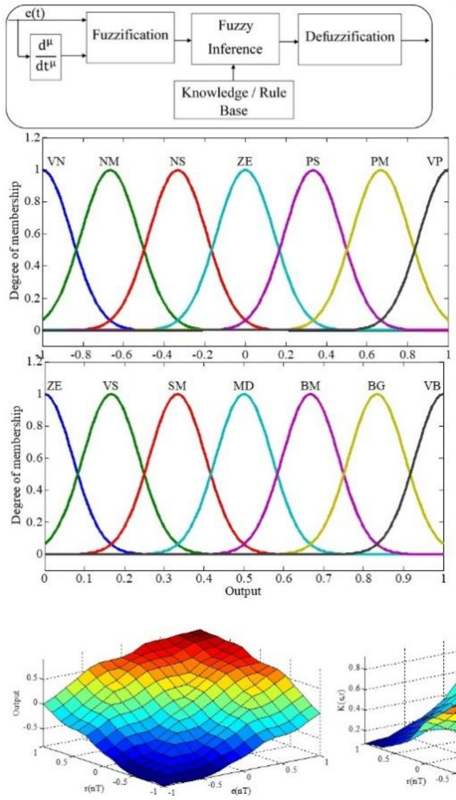
Re-organizing Eqn. (20) we get

$$u_{AFOFPID}(t) = \\ = \alpha K_{PD} u(t) + \alpha K_{PI} \left(\frac{d^{-\lambda}}{dt^{-\lambda}} \left(\frac{d^\mu u(t)}{dt^\mu} \right) \right) \tag{21}$$

Additionally, the output gains K_{PD} and K_{PI} vary from 0 to K_{PD} and 0 to K_{PI}, respectively, while the parameter α fluctuates within the range $\alpha \in (0, 1)$. The fuzzy logic controller (FLC) at Layer II, also known as the adaptive block, generates the value of α, imparting intelligence and adaptability to the AFOFPID controller. The AFOFPID controller of links of TLRMS have five main gains: K_P and K_D are input gains to the FLC Layer I and FLC Layer II, respectively, while K_{PD} and K_{PI} are output gains from the FLC Layer I and α is the output gain from the FLC Layer II. The additional fractional gains are λ and μ. All three Links use a similar control structure, with different controller gains designed for each [18-20].

Fuzzification, knowledge base (KB), fuzzy inference process, and defuzzification blocks are the four main parts of a basic fuzzy logic controller (FLC) structure, as shown in Fig. 3. The FLC uses Gaussian membership functions (MFs) and runs on a two-dimensional rule base. In this work, Gaussian MFs are selected because of their smooth output, which improves the resilience and reliability of the controller, even if other MF types, such as triangular, trapezoidal, and sigmoidal, can be used in FLC design. The Gaussian MFs utilized for the FLCs' input and output variables are shown in Fig. 3. The FLCs' output variable ranges vary from layer to layer. The range for the Layer-I FLC is [–1, 1], and for the Layer-II FLC, it is [0, 1]. The input and output MF acronyms used in both layers are shown in Fig. 3 The FLC uses the center of gravity method for de-fuzzification and a Mamdani inference mechanism. Furthermore, a maximum operator is used for aggregation and a minimum operator is used for implication. For the FLC Layer I and FLC Layer II (adaptive block), which are shown in Fig. 3, respectively, different sets of rules are contained in the KB, which is an essential part of FLC design. Figure 3 (bottom) also displays the various three-dimensional surface plots that are produced by these rule sets for Layer I and Layer II FLCs. To create smoother surface plots, the rule base from has been modified in this work. System instability may result from rapid variations in output gain brought on by irregular surfaces. The controller can be made more efficient by fine-tuning the surface plots, especially for systems like robotic manipulators that exhibit open-loop behavior that is intrinsically unstable [21, 22].

Eight essential gains, i.e., proportional gain K_P, integral gain K_I, derivative gain K_D, FOI gain λ, FOD gain μ, and the adaptive gain α and fuzzy output gains (K_{PI}, K_{PD} must be adjusted in the design of an AFOFPID controller in order to obtain optimal control performance



Input MFs	
VN	Very Negative
NM	Negative Medium
NS	Negative Small
ZE	Zero
PS	Positive Small
PM	Positive Medium
VP	Very Positive
Output MFs	
ZE	Zero
VS	Very Small
SM	Small
MD	Medium
BM	Big Medium
BG	Big
VB	Very Big

Fig. 3. Fuzzy inference block, Input and Output MFs, and surface plots for FLC layer I and FLC Layer II (top to bottom)

in a robotic system. With the benefit of the using CSA, these modifications guarantee that the controller will continue to respond to system dynamics with high accuracy and robustness. Cuckoo-inspired nesting behaviour serves as the model for CSA, which minimises an objective function J_{min} balancing the weighted sum of integrals of absolute change in controller output ($w_2 * IACCO$) and weighted sum of integrals of absolute error ($w_1 * IAE$). Compared to other techniques like Genetic Algorithms (GA) and Particle Swarm Optimisation (PSO), this procedure enables the AFOFPID controller to adaptively tweak itself for improved performance, particularly in avoiding local optima and attaining faster convergence. Since there isn't a proven tuning technique for intelligent controllers like the AFOFPID in the literature, CSA is a great option for determining the right gains. Because it is so good at striking a balance between investigation and exploitation in intricate, multifaceted problems, CSA shines. Lévy flights are another feature of the algorithm that improves its search performance by adding erratic, scale-free search patterns. The AFOFPID controller is especially well-suited for sophisticated robotic manipulator applications because of this method's ability to produce a more precise, robust, and reliable control system [4, 14, 23-25].

$$J_{min} = w_1 * IAE + w_2 * IACCO \tag{22}$$

Here, IAE is the product of w_1 and IACCO is the product w_2 [6].

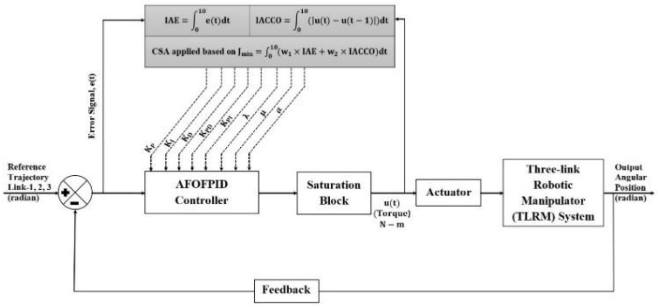


Fig. 4. Closed loop control configuration of CSA-tuned AFOFPID controller incorporated into TLRM system

4 Simulation outcomes and evaluation of comparative performance

All the simulation results and comparative performances are realized in MATLAB/SIMULINK environment. The computer included a 64-bit operating system, 16 GB of RAM, and an Intel Core i7 CPU working at 2.7 GHz. Using a sample rate of one millisecond, the fourth-order Runge-Kutta method was used to solve the ordinary differential equation (ODE). The study used the minimum objective function criterion J_{min}, to assess the performance of several controllers, including AFOFPID, FOFPID, FOPID, and conventional PID. The same set of parameters were used to optimise each controller. The findings, which are shown in Tab. 2 and Figs. 5-7, show that the AFOFPID controller outperformed the others in terms of trajectory tracking and control precision, as shown by its lowest J_{min} value. Because of its adaptive fractional-order design, which allowed it to perform better than the other controllers in terms of resilience, stability, and accuracy, the AFOFPID controller is a great choice for controlling complicated systems like a TLRM system. The findings show that in these kinds of applications, the flexibility and precision of the AFOFPID controller offer more dependable and efficient control. As Tab. 2 and Fig. 5 reveal, the AFOFPID controller outperformed both FOFPID, FOPID, and PID controllers by reaching the lowest J_{min} value, demonstrating greater control accuracy and trajectory tracking capabilities in simulations. This performance improvement is further supported by Fig. 6 (a), (b), and (c) for their reference tracking capability, where end efector, i.e., Link-3 (L3), is showcased for FOPID, FOFPID, and AFOFPID controllers. Similarly, Fig. 7 (a), (b), and (c) demonstrate error curves of end efector, i.e., L3, for FOPID, FOFPID, and AFOFPID controllers, respecttively. With improved stability, accuracy, and robustness within torque ranges of [–10, 10] N-m, the AFOFPID controller is a more dependable and accurate choice for managing intricate TLRM systems due to its utilisation of FO operators and adaptive control capabilities.

The proposed AFOFPID controller at L3 exhibits the most effective performance, characterized by the gain values K_P = 207.98, K_I = 65.76, K_D = 66.92, K_{PD} = 201.28, K_{PI} = 99.92, with fractional orders λ = 0.78 and μ = 0.36, and an adaptive factor α = 0.21, indicating improved control accuracy and dynamic response compared to other levels.

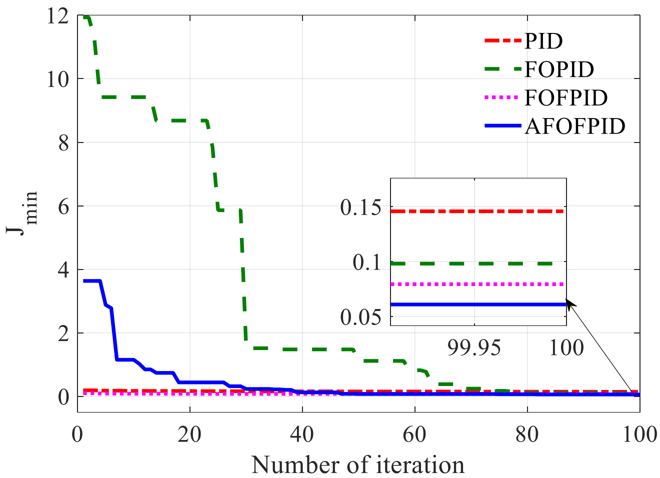


Fig. 5. J_{min} curve for AFOFPID, FOFPID, FOPID, and PID controllers

Table 2. J_{min} values for IAE and IACCO optimization metrics for AFOFPID, FOFPID, FOPID, and PID controllers

Controller	IAE Values			IACCO Values			J_{min}
	L1	L2	L3	L1	L2	L3	
AFOFPID	0.02	0.02	0.01	1.04	0.82	0.33	0.06
FOFPID	0.03	0.02	0.01	1.33	0.92	0.36	0.08
FOPID	0.03	0.06	0.05	1.69	1.50	1.11	0.10
PID	0.05	0.08	0.09	2.68	1.78	1.18	0.15

The results of the simulation demonstrate that when it comes to operating the TLRM System, the AFOFPID controller – which is optimised using the J_{min} objective function – performs better than PID, FOPID, and FOFPID controllers. The AFOFPID controller provides precise and effective control, particularly in controlling torque limits and following intended trajectories, by greatly improving reference-tracking capabilities. The controller minimises the impact of outside disturbances while achieving smooth trajectory tracking. The J_{min} performance value of the AFOFPID controller over FOFPID, FOPID, and PID controllers by 25%, 40%, and 60%, respectively, demonstrating the controller's efficacy in providing precise and dependable control for the TLRMS.

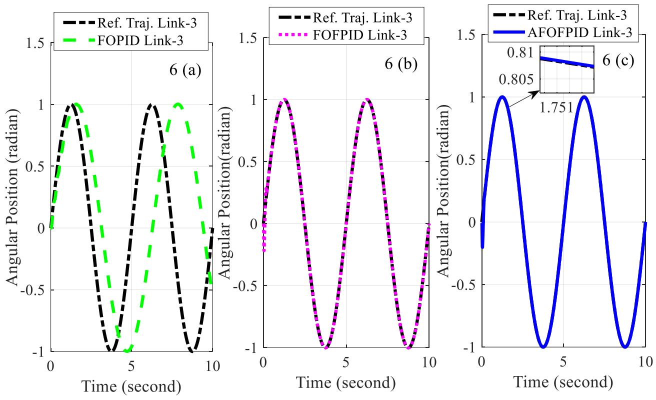


Fig. 6. Reference tracking curves of L3 for (a) FOPID, (b) FOFPID, and (c) AFOFPID controllers

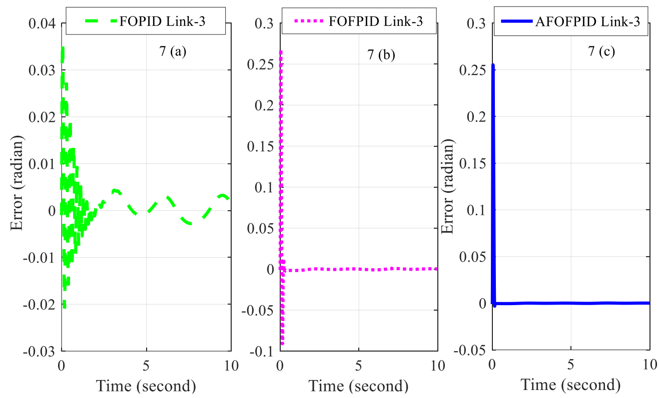


Fig. 7. Error curve for trajectory tracking analysis of L3 for (a) FOPID, (b) FOFPID, and (c) AFOFPID controllers

5 Conclusion and future works

Three-link robotic manipulators (TLRM) and other nonlinear multi-input multi-output (MIMO) systems are still difficult to control. In contrast to PID, FOPID, and FOFPID controllers, this work presented an adaptive fractional-order fuzzy PID (AFOFPID) controller as a reliable substitute that exhibits better reference tracking and overall system performance. The outcomes demonstrate how well it handles nonlinear MIMO dynamics, with objective function reductions of roughly 25%, 40%, and 60% over FOFPID, FOPID, and PID, respectively. The potential of AFOFPID in sophisticated robotics and control systems may be further established by future research that expands this methodology to more intricate robotic manipulators and real-time applications.

References

[1] K. Jayaswal, D. K. Palwalia, and S. Kumar, “Analysis of robust control method for the flexible manipulator in reliable operation of medical robots during COVID-19 pandemic,” *Microsyst. Technol.*, vol. 9, no. Romanov, 2020, doi: 10.1007/s00542-020-05028-9.

[2] K. Jayaswal, D. K. Palwalia, and S. Kumar, “Performance investigation of PID controller in trajectory control of two-link robotic manipulator in medical robots,” *J. Interdiscip. Math.*, vol. 24, no. 2, pp. 467–478, Feb. 2021, doi: 10.1080/09720502.2021.1893444.

[3] P. Tripathi, J. Kumar, and V. K. Deolia, “Trajectory Tracking Analysis of Fractional-Order Nonlinear PID Controller for Single Link Robotic Manipulator System BT - Proceedings of Fourth International Conference on Computer and Communication Technologies,” 2023, pp. 443–452.

[4] Sweta, J. Kumar, and V. K. Deolia, “Investigation of Fractional-Order Nonlinear PID Controller for Three-link Robotic Manipulator System,” in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1–7, doi: 10.1109/ICCCNT56998.2023.10307476.

[5] D. Gupta, V. Goyal, and J. Kumar, “Comparative performance analysis of fractional-order nonlinear PID controller for complex surge tank system: tuning through machine learning control approach,” *Multimed. Tools Appl.*, 2024, doi: 10.1007/s11042-024-18427-1.

[6] J. Kumar, V. Kumar, and K. P. S. Rana, “Fractional-order self-tuned fuzzy PID controller for three-link robotic manipulator system,” *Neural Comput. Appl.*, vol. 32, no. 11, pp. 7235–7257, 2020, doi: 10.1007/s00521-019-04215-8.

[7] L. A. Zadeh, “Toward a Theory of Fuzzy Systems. Aspect Network and System Theory,” *New York Rinehart Winst.*, 1971.

[8] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller,” *Int. J. Man. Mach. Stud.*, vol. 7, no. 1, pp. 1–13, 1975, doi: https://doi.org/10.1016/S0020-7373(75)80002-2.

[9] K. Khorasani, “Adaptive control of flexible joint robots,” in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, 1991, pp. 2127–2134 vol.3, doi: 10.1109/ROBOT.1991.131942.

[10] R. K. Mudi and N. R. Pal, “A note on fuzzy PI-type controllers with resetting action,” *Fuzzy Sets Syst.*, vol. 121, no. 1, pp. 149–159, 2001, doi: https://doi.org/10.1016/S0165-0114(99)00143-8.

[11] H. A. Malki, D. Misir, D. Feigenspan, and G. Chen, “Fuzzy PID control of a flexible-joint robot arm with uncertainties from time-varying loads,” *IEEE Trans. Control Syst. Technol.*, vol. 5, no. 3, pp. 371–378, 1997, doi: 10.1109/87.572133.

[12] A. Koivo and T.-H. Guo, “Adaptive linear controller for robotic manipulators,” *IEEE Trans. Automat. Contr.*, vol. 28, no. 2, pp. 162–171, 1983, doi: 10.1109/TAC.1983.1103211.

[13] J. Kumar, D. Gupta, and V. Goyal, “Nonlinear PID Controller for Three-Link Robotic Manipulator System: A Comprehensive Approach BT - Proceedings of International Conference on Communication and Artificial Intelligence,” 2022, pp. 137–152.

[14] R. Sharma, K. P. S. Rana, and V. Kumar, “Performance analysis of fractional order fuzzy PID controllers applied to a robotic manipulator,” *Expert Syst. Appl.*, vol. 41, no. 9, pp. 4274–4289, 2014, doi: 10.1016/j.eswa.2013.12.030.

[15] C. Yeroglu and N. Tan, “Classical controller design techniques for fractional order case,” *ISA Trans.*, vol. 50, no. 3, pp. 461–472, 2011, doi: https://doi.org/10.1016/j.isatra.2011.03.004.

[16] J. Kumar, V. Goyal, and D. Gupta, "Reliable Incorporation of Robust Fractional-Order Nonlinear PID Controller into Integrated Power System Using Gray Wolf Optimization Method BT - Advances in Data and Information Sciences," 2023, pp. 333–343.

[17] V. Kumar, K. P. S. Rana, J. Kumar, and P. Mishra, "Self-tuned robust fractional order fuzzy PD controller for uncertain and nonlinear active suspension system," *Neural Comput. Appl.*, vol. 30, no. 6, pp. 1827–1843, 2018, doi: 10.1007/s00521-016-2774-x.

[18] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. I," *IEEE Trans. Syst. Man. Cybern.*, vol. 20, no. 2, pp. 404–418, 1990, doi: 10.1109/21.52551.

[19] A. Mourad and Z. Youcef, "Fuzzy-PI Controller Tuned With ICA: applied to 2 DOF robot control trajectory," in *2022 XXVIII International Conference on Information, Communication and Automation Technologies (ICAT)*, 2022, pp. 1–6, doi: 10.1109/ICAT54566.2022.9811113.

[20] A. Agrawal, V. Goyal, and P. Mishra, "Comparative Study of Fuzzy PID and PID Controller Optimized with Spider Monkey Optimization for a Robotic Manipulator System," *Recent Adv. Comput. Sci. Commun. (Formerly Recent Patents Comput. Sci.*, vol. 14, no. 4, 2021.

[21] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965, doi: https://doi.org/10.1016/S0019-9958(65)90241-X.

[22] J. Kumar, V. Kumar, and K. P. S. Rana, "A fractional order fuzzy PD+I controller for three-link electrically driven rigid robotic manipulator system," *J. Intell. Fuzzy Syst.*, vol. 35, pp. 5287–5299, 2018, doi: 10.3233/JIFS-169812.

[23] X.-S. Yang and S. Deb, "Cuckoo search: recent advances and applications," *Neural Comput. Appl.*, vol. 24, no. 1, pp. 169–174, 2014, doi: 10.1007/s00521-013-1367-1.

[24] K. Deb, "An introduction to genetic algorithms," *Sadhana*, vol. 24, no. 4, pp. 293–315, 1999, doi: 10.1007/BF02823145.

[25] R. M. Ramadan and R. F. Abdel-Kader, "Particle swarm optimization for human face recognition," in *2009 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 2009, pp. 579–584, doi: 10.1109/ISSPIT.2009.5407518.

Received 11 November 2025