



International Journal of Automation and Control

ISSN online: 1740-7524 - ISSN print: 1740-7516

<https://www.inderscience.com/ijaac>

GA-optimised fuzzy control for trajectory tracking of a 3-DOF robotic manipulator

Abdullah Abushokor, Asem Abdalhadi, Muhammad Faizan Mysorewala

DOI: [10.1504/IJAAC.2027.10075191](https://doi.org/10.1504/IJAAC.2027.10075191)

Article History:

Received:	17 March 2025
Last revised:	23 September 2025
Accepted:	10 October 2025
Published online:	26 October 2026

GA-optimised fuzzy control for trajectory tracking of a 3-DOF robotic manipulator

Abdullah Abushokor and Asem Abdalhadi

Department of Control and Instrumentation Engineering,
King Fahd University of Petroleum and Minerals (KFUPM),
Dhahran, 31261, Saudi Arabia
Email: g202320370@kfupm.edu.sa
Email: g202391930@kfupm.edu.sa

Muhammad Faizan Mysorewala*

Department of Control and Instrumentation Engineering,
King Fahd University of Petroleum and Minerals (KFUPM),
Dhahran, 31261, Saudi Arabia

and

Department of Computer Engineering,
University of Sharjah,
Sharjah, UAE
Email: mysorewala@kfupm.edu.sa
Email: mmysorewala@sharjah.ac.ae

*Corresponding author

Abstract: This paper investigates the use of three control techniques to improve trajectory tracking of a 3-DOF robotic arm. Specifically, Fuzzy Logic Control, Adaptive Neuro Fuzzy Inference System-based control, and a Genetic Algorithm (GA)-enhanced fuzzy controller are examined, with the objective of improving tracking accuracy and handling disturbances and parameter variations. Models were derived, and controllers' performance was evaluated using MATLAB/Simulink. Results showed that the GA-optimised fuzzy controller outperformed the others, achieving the highest accuracy and stability. Under external disturbances, the GA-optimised fuzzy controller maintained strong performance, achieving an average root mean square error (RMSE) reduction of 89% compared to feedback linearisation controller (FBLC), and reductions of 95% and 91% compared to conventional fuzzy logic control (FLC) and Adaptive Neuro-Fuzzy Inference System (ANFIS). Moreover, under severe parameter variations, it demonstrated robust resilience, maintaining stable tracking with consistently low RMSE, whereas the FBLC failed. Findings suggest that the GA-optimised fuzzy controller provides an effective solution for applications where precision and resilience to external changes are critical.

Keywords: intelligent control; trajectory tracking; robotic arm; FLC; fuzzy logic control; Genetic Algorithm.

Reference to this paper should be made as follows: Abushokor, A., Abdalhadi, A. and Mysorewala, M.F. (2026) ‘GA-optimised fuzzy control for trajectory tracking of a 3-DOF robotic manipulator’, *Int. J. Automation and Control*, Vol. 20, No. 8, pp.1–27.

Biographical notes: Abdullah Abushokor received the Bachelor’s degree in Mechanical/Mechatronics Engineering from Palestine Polytechnic University, Palestine, in 2021. He is currently pursuing the MS degree in Control and Systems Engineering with the Control and Instrumentation Engineering Department, King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia. His research interests include nonlinear control systems, robust control, sliding mode control, adaptive control, neuro-fuzzy control, and time delay-based estimation control. He is actively engaged in the development of advanced control strategies for robotics and autonomous systems, with a particular emphasis on faster convergence, adaptive mechanisms, and optimisation-driven design methods.

Asem Abdalhadi received the BEng degree in Mechatronics Engineering from the International Islamic University Malaysia (IIUM), Malaysia, and the MEng degree in Mechatronics Engineering from the University of Malaya (UM), Malaysia. He is currently pursuing the PhD degree in Control and Systems Engineering at King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia. His research interests include adaptive, robust, and intelligent control of robotic systems, observer design, and parameter estimation. His current research focuses on torque control and thermal derating in electric vehicle motor drives, emphasising field-oriented control and estimation-based approaches.

Muhammad Faizan Mysorewala received the MS and PhD degrees in Electrical Engineering from The University of Texas at Arlington, USA, in 2002 and 2008, respectively. He is an Associate Professor with the Department of Control and Instrumentation Engineering, King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia, and is affiliated with the Interdisciplinary Research Centers for Smart Mobility and Logistics and for Intelligent Manufacturing and Robotics. His research focuses on decision-making, motion planning, and control for autonomous and robotic systems, developing integrated frameworks for navigation, human–robot interaction, and distributed sensing to improve autonomy, safety, and efficiency.

1 Introduction

Robotic manipulators are an important element in modern industrial processes. These electronically controlled mechanical devices have the capability to perform specific tasks under certain conditions (Satya Durga Manohar Sahu et al., 2022; Iqnaibi et al., 2023; Hady et al., 2023). They are utilised in various applications such as pick-and-place, material handling, assembly, welding, and many other tasks (Reham H.Mohammed, 2016; Chifisi et al., 2024). The widespread use of robots in different fields is due to the significant increase in their functionality over time (Galin and Meshcheryakov, 2019). It can be said that in the Industry 4.0 era, manufacturing processes have been revolutionised by the technologies

of robotic manipulation, transforming robots into flexible, mobile, and intelligent entities (Hassan et al., 2024).

Control design is an important aspect of robotic manipulators. The demand for robotic manipulators in precise, critical, and high-speed applications necessitates enhanced performance to match the expected speed, reduce tracking error, and handle uncertainties and disturbances of the system (Kabir et al., 2019).

However, many challenges in designing a controller arise due to the nonlinear, dynamic, and multivariable nature of the robot manipulator system. Such challenges create the need for effective and reliable controlling approaches capable of handling the behavior of the system and achieving the performance requirements.

Control strategies based on fuzzy logic and neural networks offer effective solutions for managing complex, nonlinear systems such as robotic manipulators. These techniques help reduce the effects of disturbances and modelling inaccuracies, which enable nonlinear systems like robotic arms to achieve superior trajectory tracking and real-time adaptability (Guo et al., 2025). For instance, fuzzy-based control strategies have proven effective even in chaotic and highly nonlinear environments, as demonstrated in recent work on fractional-order systems with no equilibria (Zambrano-Serrano et al., 2021; Kharola, 2023), and have been successfully applied in uncertain environments using fractional-order models (Bataineh et al., 2021). The mathematical foundations of fuzzy systems, including dominated fuzzy mappings in specialised metric spaces (Rasham et al., 2022), hybrid orthonormal function methods for fuzzy integral equations (Agarwal et al., 2020), and stability results for Clifford-valued neural systems under delays and impulses (Rajchakit et al., 2021), further reinforce their applicability and robustness in dynamic systems. Other tools were employed to enhance the performance as well, such as evolutionary algorithms. Algorithms like GA and PSO have shown noticeable improvements in terms of tracking precision and energy efficiency (Zuoxun et al., 2024). The integration of these techniques significantly impacted automation. As a result, robotic systems are capable of handling complex tasks with minimal human intervention (Elmogy et al., 2025).

This paper examines three control techniques to improve trajectory tracking of a 3-DOF robotic manipulator: fuzzy logic control (FLC), Adaptive Neuro-Fuzzy Inference System (ANFIS), and a Genetic-Algorithm-optimised fuzzy controller. Performance is compared against the nonlinear, feedback linearisation controller (FBLC). The kinematics and dynamics are derived using Denavit-Hartenberg (D-H) parameters and the Lagrange-Euler method. Tracking and disturbance rejection are evaluated via MATLAB/Simulink simulations with prescribed trajectories. Results indicate improvements in adaptability, precision, and efficiency.

The main contributions of this paper are summarised as follows:

- Design and implementation of three control strategies. The FLC, ANFIS, and Genetic Algorithm (GA) optimised fuzzy control for trajectory tracking in a 3-DOF robotic manipulator.
- Development of kinematic and dynamic models using Denavit-Hartenberg and Lagrange-Euler formulations to support the controller design and simulation process.
- Simulation-based comparative evaluation of FLC, ANFIS, GA-FLC, and a model-based FBLC under nominal, disturbed, and parameter variation conditions.
- Validation of the GA optimisation method's effectiveness for fuzzy controller parameter tuning, achieving enhanced trajectory tracking precision and robustness in robotic manipulator control.

The structure of the paper is as follows: Section 2 reviews the related literature. Section 3 presents the methodology. In Section 3.1, the kinematic and dynamic modelling of the 3-DOF robotic arm is developed using the Denavit-Hartenberg method and the Lagrange-Euler equations. Section 3.2 describes the control strategies, including the FLC, the ANFIS, the GA-optimised Fuzzy Logic Controller, and the FBLC. Section 4 provides the results and discussion, covering trajectory tracking, trajectory tracking under external disturbance, and trajectory tracking under changing system parameters. Section 5 concludes the paper, highlighting the robustness and effectiveness of the proposed adaptive and hybrid control strategies compared with the traditional model-based FBLC approach.

2 Literature review

Many control techniques based on fuzzy logic and neural networks, along with their hybrid approaches, have been used in robotic applications such as trajectory tracking. In nonlinear systems, the popularity of FLCs has increased due to their robustness and linguistic interpretability. FLCs overcome the limitations of model-based control approaches by employing knowledge-based rules for control (Hentout et al., 2022). FLCs are classified under the domain of expert systems, where IF-THEN rules are used to enable adaptive decision-making (Eckert et al., 2022). Several strategies for implementing FLC have been applied to robotic-manipulator trajectory tracking. In Kabir et al. (2019), a fuzzy controller was evaluated against PD and PID for position control of a 3-DOF arm: PD/PID achieved faster rise and settling times, whereas FLC reduced overshoot while maintaining comparable steady-state error.

Under dynamic conditions, Urrea et al. (2024) reported that FLC outperformed MPC, yielding smoother tracking and stronger disturbance rejection. Beyond baseline performance, many studies highlight robustness gains with fuzzy methods. Yang et al. (2024) developed an adaptive FLC that effectively minimised tracking errors in uncertain environments. Rahali et al. (2025) proposed a Type-2 FLC for manipulators that demonstrated superior fault tolerance and precision in the presence of actuator faults. In human-robot interaction and real-time multi-task planning, Wang et al. (2024) showed that FLC enhances responsiveness and reliability.

Hybrid designs combine fuzzy logic with classical controllers to exploit complementary strengths. While PID or LQR offer fast, precise regulation, FLC contributes adaptability and nonlinearity handling. In Beata (2018), a fuzzy-PD controller for PMSM-driven manipulators achieved better trajectory tracking than conventional schemes and required less computation than a decentralised approach. Similarly, Oktarina et al. (2019) designed a fuzzy-PID for a 4-DOF industrial robot, using FLC for target detection and PID for smooth, stable motion; simulations confirmed stable and robust behavior. An evaluation of PID, LQR, hybrid PID-LQR, and fuzzy LQR (FLQR) on the Robogymnast showed FLQR with the best stability (Mohamed et al., 2023). Additional evidence includes fuzzy-tuned LQR improving accuracy and robustness in flexible manipulators (Lu et al., 2016), and a Fuzzy-LQG enhancing stability and tracking in mobile robots (Finecomess et al., 2024). Fuzzy logic can also serve as a supervisory layer that adapts linear controllers online. In Mohammed et al. (2022), an adaptive 2-DOF PID with a supervisory FLC improved trajectory tracking in simulation and demonstrated robustness in real-time hardware tests on a robot manipulator under external disturbances.

To further enhance the performance of fuzzy controllers, different optimisation algorithms, such as Genetic Algorithms, have been employed. Genetic Algorithms mimic the process of natural selection, where the fittest solutions survive and reproduce, leading to increasingly better outcomes over time. They are particularly effective at finding the best possible solutions for complex optimisation challenges (Tarn et al., 1996). GAs work by exploring a wide range of options, rather than getting stuck on the first good solution they find. This helps them avoid getting trapped in local optima and instead discover the truly optimal, or best overall, solution (Sberna et al., 2023). Several studies have demonstrated the effectiveness of GA-optimised fuzzy controllers in robotic systems. In Rodríguez-Abreo et al. (2024), a Genetic algorithm was used to optimise the gains of a fuzzy controller for an unmanned aerial vehicle (UAV). MATLAB simulations showed significant reductions in tracking error, and that previously unattainable trajectories with the untuned controller were completely tracked.

In Rodríguez et al. (2022), a fuzzy-based position controller implemented on a real platform was also optimised using GA, enhancing its robustness and adaptability under different initial conditions. Menich and Kopjak (2023) further validated the superiority of GA for automating fuzzy membership function tuning, reducing the need for manual adjustments, and significantly improving controller performance. Studies continue to reinforce the effectiveness of GA-based optimisation in fuzzy logic control. Stavrindis et al. (2025) applied GA to optimise dual fuzzy controllers for multi-goal robotic navigation, demonstrating enhanced adaptability in dynamic obstacle avoidance scenarios. Additionally, its effectiveness in real-world force control scenarios was shown by Zhai et al. (2024), where GA was used to optimise a fuzzy-PID control system for robotic assistance applications. The accuracy and adaptability of FLCs under uncertainties have been further enhanced by integrating neural networks. Neural networks enable real-time adaptation to uncertainties and nonlinearities due to their learning capabilities. Their integration with fuzzy logic has significantly enhanced the handling of complex dynamics and adaptation in real-time scenarios (Jin et al., 2018).

The combination of fuzzy logic and neural networks, ANFIS, has achieved superior performance in trajectory tracking and dynamic control applications. In Han et al. (2023), ANFIS was analysed for trajectory tracking in a hydraulic manipulator. The online tuning of the FLC improved its tracking accuracy and adaptability to environmental changes. The ANFIS-PID controller outperformed PID and FPID controllers in terms of precision and stability. In Barhaghtalab et al. (2024), an ANFIS-based control system was applied to a 6-DOF robotic manipulator. While ANFIS handled system uncertainties, the PID controller was used for feedback control. Simulations and experimental results showed high precision, stability, and robustness against external disturbances. Additionally, Barua et al. (2022) extended the analysis to the inverse dynamics of a 3-DOF planar robotic arm. The results highlighted the accuracy and adaptability of ANFIS-based control in handling parameter uncertainties and varying operational conditions. Robotic manipulators' enhanced performance and robustness achieved with ANFIS have been further shown by several other studies (Mukhtar et al., 2021; Chaudhary et al., 2014; Elawady et al., 2019).

These studies highlight the advancements in control strategies based on fuzzy logic, neural networks, and hybrid approaches for robotic manipulators. FLCs have exhibited strong performance in trajectory-tracking applications. Moreover, hybrid controllers like fuzzy-PID, fuzzy-LQR, and fuzzy-ANN (ANFIS) have further enhanced precision and stability. For robot manipulator trajectory tracking, the use of optimisation algorithms has become standard. They are used to either tune controller parameters (e.g., fuzzy membership

functions/rules, PID/FOPID gains, or SMC surfaces) or to optimise the reference trajectory itself (Oleiwi et al., 2025; Azeez et al., 2023a,b; Elsisi et al., 2021; Sun et al., 2025). Table 1 synthesises recent optimisation-driven tracking papers, indicating the optimiser, what is optimised (controller vs trajectory), the test scenario, and the reported outcomes. Although Genetic Algorithms have been widely utilised for fuzzy controllers in UAV and mobile robot applications, their direct application to fuzzy controllers for robotic manipulators remains underexplored. In robotic manipulator control, the examined approaches provide a strong foundation for further improvements in efficiency, adaptability, and real-time performance.

Table 1 Recent optimisation-based approaches for manipulator trajectory tracking

<i>Study (year)</i>	<i>Optimisation/Target</i>	<i>Manipulator/Scenario</i>	<i>Result/Performance</i>	<i>Comments</i>
Our work	GA optimising FLC membership functions	3-DOF arm; reference trajectory; nominal + disturbance + parameter variation	Lower tracking error than FLC, ANFIS, FBLC; preserved tracking under disturbance and parameter drift	Robust to disturbance/param drift; small GA budget; MATLAB/Simulink; GA runs offline
Sun et al. (2025)	MAPSO (modified PSO) optimises trajectory and fuzzy-PD controller	2-link flexible manipulator (simulation + experiment)	33-54% fewer iterations and 0.03 s faster; MAPSO-FuzzyPD cut peak torque $\sim 60\%$ vs. PD	Vibration suppression; efficiency and reduced actuation effort
Elsisi et al. (2021)	BOA tunes PID controller (cooperative strategy)	2-DOF robot manipulator (multiple trajectories)	$<1.81\%$ overshoot and ~ 0.11 s settling; better than GA-PID and CSA-PID	Cooperative BOA avoids local minima; best tracking in comparisons
Oleiwi et al. (2025)	Bat algorithm optimises hybrid NN-FOPID controller	3-link rigid manipulator (MIMO; unknown loads & disturbances)	Best tracking among three designs; no chattering; robust under uncertainties	Hybrid neural/FOPID improved by bat optimisation
Azeez et al. (2023a)	MGABC (multi-elite ABC) optimises PID (+ Lyapunov fitness)	3-DOF rigid manipulator (disturbances & payload variance)	Outperformed basic ABC; Lyapunov objective improved accuracy and disturbance rejection	Elite guidance improved convergence; ensured stability
Azeez et al. (2023b)	MGABC optimises SMC-PID (sliding mode + PID surface)	3-link flexible-joint manipulator (input/output noise)	Near-zero error (~ 0.0069 rad) with no chattering; outperformed PSO, GA, ABC; robust under heavy noise/disturbance	Superior tracking and resilience vs. other optimisers

3 Methodology

The mathematical model is specifically developed for simulation purposes and does not directly influence the FLC, ANFIS-based controller, or GA-optimised FLC. These controllers utilise data-driven or heuristic methods rather than relying on explicit system models. In contrast, the FBLC depends heavily on the mathematical model, as it is a model-based approach requiring an accurate representation of the system dynamics for effective linearisation and control.

3.1 Modelling

Two distinct models were developed: the kinematic model, which describes the motion without considering forces, and the dynamic model, which establishes the relationship between forces and motion.

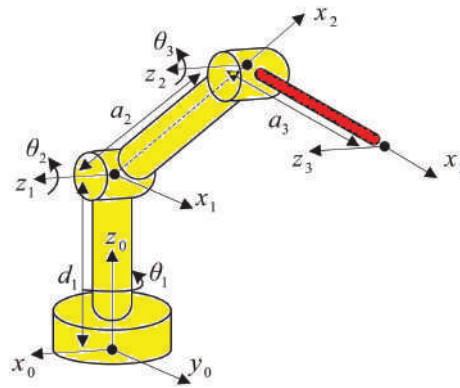
3.1.1 Kinematic robot model

A. Direct kinematic model: The direct kinematic model is obtained using the Denavit-Hartenberg (D-H) method, which calculates spatial coordinates based on joint coordinates. Table 2 lists the D-H parameters derived from the reference system illustrated in Figure 1 (Siciliano et al., 2010), where the values of d_1 , a_2 , and a_3 are 0.352, 0.36, and 0.446, respectively.

Table 2 D-H parameters of the 3-DOF manipulator

Joint	a_i	α_i	d_i	θ_i
1	0	$\frac{\pi}{2}$	d_1	θ_1
2	a_2	0	0	θ_2
3	a_3	0	0	θ_3

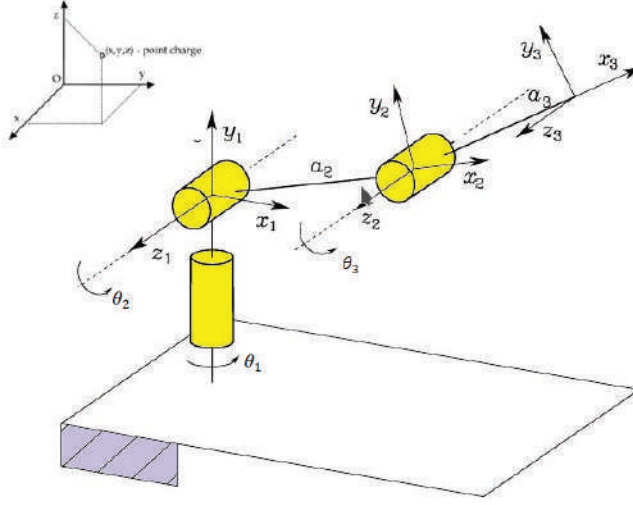
Figure 1 Links' framework utilised in the modelling (see online version for colours)



The parameters and angles of the robotic arm a_2 , a_3 , d_1 , along with joint angles θ_1 , θ_2 , and θ_3 simplify the determination of the end-effector position (p_x, p_y, p_z) using forward kinematic equations, facilitating accurate trajectory tracking. These equations significantly enhance

control design and spatial mapping within Cartesian coordinates. They are essential for precise model-based controller designs, such as feedback linearisation-based fuzzy control, where accurate kinematic position data derived from joint angles is crucial. Additionally, these models accurately reflect the geometric configuration of the 3-DOF robotic arm, ensuring realistic and reliable simulations of its real-world movements.

Figure 2 Geometric representation and postures for inverse kinematics of the robotic (see online version for colours)



Based on the D-H table, the direct kinematic equations are represented as follows:

$$p_x = a_3 \cos(\theta_3) \cos(\theta_1) \cos(\theta_2) - a_3 \sin(\theta_3) \cos(\theta_1) \sin(\theta_2) + a_2 \cos(\theta_1) \cos(\theta_2), \quad (1)$$

$$p_y = a_3 \cos(\theta_3) \sin(\theta_1) \cos(\theta_2) - a_3 \sin(\theta_3) \sin(\theta_1) \sin(\theta_2) + a_2 \sin(\theta_1) \cos(\theta_2), \quad (2)$$

$$p_z = a_3 \cos(\theta_3) \sin(\theta_2) + a_3 \sin(\theta_3) \cos(\theta_2) + a_2 \sin(\theta_2) + d_1. \quad (3)$$

B. Inverse kinematic model: The inverse kinematic model is constructed based on the physical dimensions and geometry of the robot. It is utilised to calculate the joint variables from the final position and orientation of the end-effector. Using the geometric representation shown in Figure 2 (Kern et al., 2023), the joint angles θ_1 , θ_2 , and θ_3 are determined using the following equations:

$$\theta_1 = \text{atan2}(p_y, p_x), \quad (4)$$

$$r = \sqrt{p_y^2 + p_x^2},$$

$$z_{\text{eff}} = p_z - d_1,$$

$$d = \sqrt{r^2 + z_{\text{eff}}^2},$$

$$x_2 = \cos^{-1} \left(\frac{a_2^2 + d^2 - a_3^2}{2a_2d} \right),$$

$$x_3 = \cos^{-1} \left(\frac{a_2^2 + a_3^2 - d^2}{2a_2a_3} \right),$$

$$\theta_2 = \text{atan2}(z_{\text{eff}}, r) - x_2, \quad (5)$$

$$\theta_3 = \pi - x_3. \quad (6)$$

3.1.2 Dynamic model

The dynamic model of the 3-DOF robotic arm was obtained through the Lagrange-Euler formalisms as described in Spong et al. (2006); Siciliano et al. (2010). This method is based on the principle of least action and utilises the kinetic and potential energies of the system. This clearly defines a method to obtain the nonlinear equations of motion for mechanical systems. This approach works very well for multi-link robotic manipulators because it accommodates joint interactions and coupling effects. These equations are essential for control strategies, such as feedback linearisation and advanced fuzzy control. Following this procedure, the dynamic model is given as:

$$\tau = B(q)\ddot{q} + h(q, \dot{q}) + G(q) + F(\dot{q}), \quad (7)$$

where

- $B(q)$ is the inertia matrix.
- $h(q, \dot{q})$ represents the Coriolis matrix.
- $G(q)$ is the gravitational torque vector.
- $F(\dot{q})$ corresponds to the frictional forces.

$$B(q) = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}, \quad (8)$$

$$b_{11} = (m_3 + 0.25 m_2) l_2^2 \cos^2(\theta_2) + 0.25 m_3 l_3^2 \sin(\theta_2 + \theta_3) + m_3 l_2 l_3 \cos(\theta_2) \sin(\theta_2 + \theta_3), \quad (9)$$

$$b_{12} = b_{13} = b_{21} = b_{31} = 0, \quad (10)$$

$$b_{22} = (m_3 + 0.25 m_2) l_2^2 + 0.25 m_3 l_3^2 + m_3 l_2 l_3 \sin(\theta_3), \quad (11)$$

$$b_{23} = 0.25 m_3 l_3^2 + 0.5 m_3 l_2 l_3 \sin(\theta_3), \quad (12)$$

$$b_{32} = b_{23}, \quad (13)$$

$$b_{33} = 0.25 m_3 l_3^2, \quad (14)$$

$$h(q, \dot{q}) = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}. \quad (15)$$

$$h_1 = \dot{\theta}_1 \dot{\theta}_2 \left[(0.25 m_3) l_3^2 \sin(2(\theta_2 + \theta_3)) \right. \quad (16)$$

$$\left. - (m_3 + 0.25 m_2) l_2^2 \sin(2\theta_2) + m_3 l_2 l_3 \cos(2\theta_2 + \theta_3) \right] + \dot{\theta}_1 \dot{\theta}_3 \left[0.5 m_3 l_3^2 \sin(2(\theta_2 + \theta_3)) + m_3 l_2 l_3 \cos \theta_2 \cos(\theta_2 + \theta_3) \right], \quad (17)$$

$$h_2 = \dot{\theta}_2 \dot{\theta}_3 (-m_3 l_2 l_3 \cos \theta_3) + \dot{\theta}_3^2 (-m_3 l_2 l_3 \cos \theta_3) + 0.5 \dot{\theta}_1^2 [(m_3 + 0.25 m_2) l_2^2 \sin(2\theta_2) + 0.25 m_3 l_3^2 \sin(2(\theta_2 + \theta_3)) + 0.5 m_3 l_2 l_3 \cos(2\theta_2 + \theta_3)], \quad (18)$$

$$+ 0.5 m_3 l_2 l_3 \cos(2\theta_2 + \theta_3)], \quad (19)$$

$$h_3 = -\dot{\theta}_1^2 [0.25 m_3 l_3^2 \sin(2(\theta_2 + \theta_3)) + m_3 l_2 l_3 \cos \theta_2 \cos(\theta_2 + \theta_3)] - (0.5 \dot{\theta}_1^2 + \dot{\theta}_2 \dot{\theta}_3) (m_3 l_2 l_3 \cos \theta_3). \quad (20)$$

$$G(q) = \begin{bmatrix} 0 \\ g((m_3 + 0.5 m_2) l_2 \cos(\theta_2) + 0.5 m_3 l_3 \sin(\theta_2 + \theta_3)) \\ g(0.5 m_3 l_3 \sin(\theta_2 + \theta_3)) \end{bmatrix}, \quad (21)$$

$$F(\dot{q}) = \begin{bmatrix} F_v(q_1) \\ F_v(q_2) \\ F_v(q_3) \end{bmatrix} = \begin{bmatrix} b_1 \dot{\theta}_1 \\ b_2 \dot{\theta}_2 \\ b_3 \dot{\theta}_3 \end{bmatrix}. \quad (22)$$

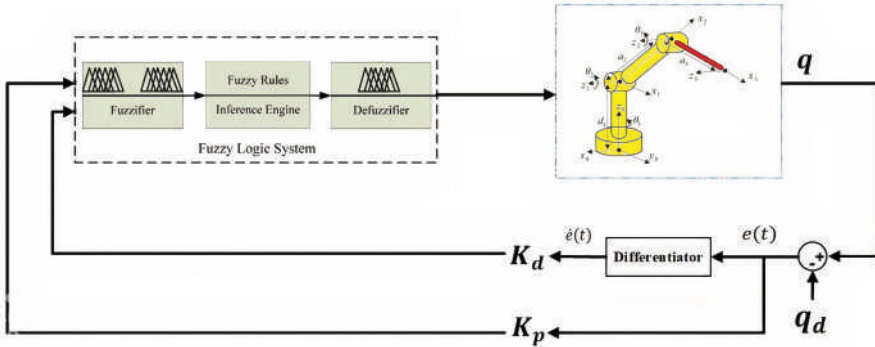
Remark 1: The zero entries in the inertia matrix $B(q)$ arise because certain joint velocities do not appear together in the kinetic energy expression. This reflects the absence of inertial coupling between those joints, and due to symmetry we also have $b_{ij} = b_{ji} = 0$ (Spong et al., 2020).

3.2 Control strategies

3.2.1 Fuzzy logic controller

The Mamdani inference method was used to design a fuzzy controller for each joint. The state and desired behavior of each joint were considered in designing the fuzzy rules to establish independence between them. Figure 3 shows how the controller was implemented in MATLAB/Simulink.

Figure 3 Schematic diagram of the fuzzy logic controller (FLC) (see online version for colours)



In the controller, two inputs were defined: **Error** ($e(t)$) and **dError** ($\dot{e}(t)$), each with three membership functions (**N**, **Z**, **P**) as seen in Figure 4(a) and (b). On the other hand, one output, **Torque**, with five membership functions (**TVL**, **TL**, **TZ**, **TH**, **TVH**), was used as shown

in Figure 4(c). For reproducibility, the triangular membership functions were explicitly defined by their ‘start-peak-end’ points. For example, the Error input (Figure 4(a)) uses: $N = [-1, -1, 0]$, $Z = [-0.5, 0, 0.5]$, and $P = [0, 1, 1]$. The derivative input (Figure 4(b)) uses: $N = [-1, -1, 0]$, $Z = [-0.5, 0, 0.5]$, and $P = [0, 1, 1]$. The output Torque (Figure 4(c)) is defined with five triangular sets: $TVL = [-1, -1, -0.5]$, $TL = [-1, -0.5, 0]$, $TZ = [-0.5, 0, 0.5]$, $TH = [0, 0.5, 1]$, $TVH = [0.5, 1, 1]$.

Based on the behavior of each joint at a given moment, the parameters of the controllers were fine-tuned experimentally. Table 3 shows the nine rules used in the rule base, while Table 4 shows the empirically obtained values of the gains K_p and K_d . For clarity, example rules are: (1) IF $e(t)$ is Positive AND $\dot{e}(t)$ is Positive THEN Torque = TVH, (2) IF $e(t)$ is Zero AND $\dot{e}(t)$ is Negative THEN Torque = TL. This results in a 3×3 rule base (Table 3). All fuzzy terms (N, Z, P, TVL, TL, TZ, TH, TVH) are now explicitly defined to ensure reproducibility.

Figure 4 Defined membership functions for the input and output variables of the fuzzy logic system: (a) input 1 (Error) membership functions; (b) input 2 (dError) membership functions and (c) output (Torque) membership functions (see online version for colours)

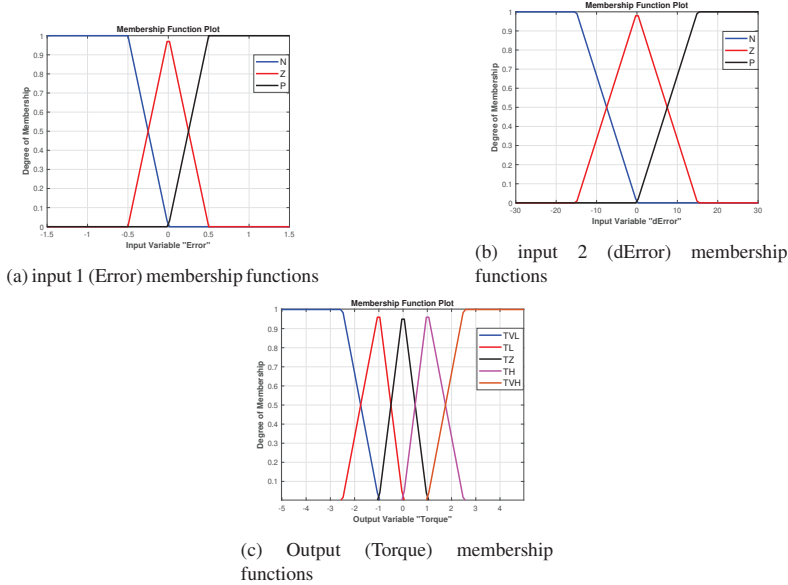


Table 3 FLC inference rule base

$\dot{e}(t)$ \ $e(t)$	P	Z	N
P	TVH	TH	TZ
Z	TH	TZ	TL
N	TZ	TL	TVL

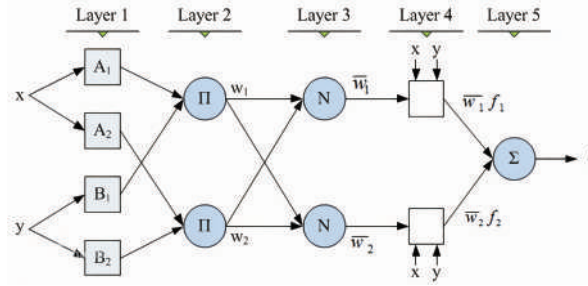
Table 4 FLC gains

Joint	K_P	K_D
1	20	20
2	8.1	7.3
3	6.2	5.8

3.2.2 ANFIS controller

ANFIS is a hybrid approach that combines the strengths of fuzzy logic (handling uncertainty) and neural networks (learning from data). ANFIS was used to optimise the control of the 3-DOF robotic manipulator for trajectory tracking, enabling it to handle external disturbances and load variations. Figure 5 illustrates the structure of the ANFIS, highlighting its training procedure.

Figure 5 Structure of the ANFIS, which processes two crisp inputs, x ($e(t)$) and y ($\dot{e}(t)$), through five layers: fuzzification, rule evaluation, normalisation, consequent computation using Sugeno rules, and output aggregation to produce the final crisp output f (see online version for colours)



Source: By Geronimo et al. (2013)

ANFIS employs a hybrid learning algorithm, which ensures adaptability and efficiency in achieving the desired control behavior. The backpropagation Gradient Descent is used to optimise the membership function parameters, in addition to the Least Squares Estimation for fine-tuning the output function parameters.

Figure 6 shows how the controller was implemented in MATLAB Simulink. Each joint's ANFIS was implemented as a Takagi-Sugeno model with Gaussian membership functions and a first-order Sugeno output (linear combination of inputs) per rule. Specifically, six Gaussian MFs were used for Joint 1's inputs, while Joints 2 and 3 used five Gaussians each, based on preliminary tuning. The inputs to each ANFIS were $e(t)$, $\dot{e}(t)$, and $\ddot{q}_d(t)$ (desired acceleration), and the output was the control torque.

A dataset of 10,000 samples was generated by simulating the manipulator under diverse trajectories and recording the error dynamics. This dataset was divided into 70% for training and 30% for validation to ensure generalisation. Training used MATLAB's `anfisedit` tool with grid partitioning to initialise the rule base, and a hybrid learning algorithm (backpropagation for premise parameters and least-squares for consequent parameters) over 300 epochs. The number of epochs was chosen to ensure convergence, and this corrects

the earlier text that mistakenly indicated 100 epochs. The training process was repeated multiple times to reduce the risk of local minima.

The large dataset was selected to adequately cover the input space, and the training/validation errors tracked each other closely, confirming that the ANFIS models did not suffer from overfitting and were properly trained.

Figure 6 ANFIS-based controller schematic using $e(t)$, $\dot{e}(t)$, and $\ddot{q}_d$ as inputs to generate control signals, ensuring the manipulator tracks the desired position q_d and velocity $\dot{q}_d$ (see online version for colours)

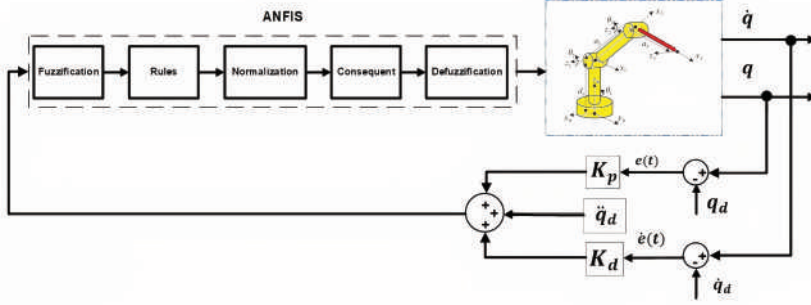
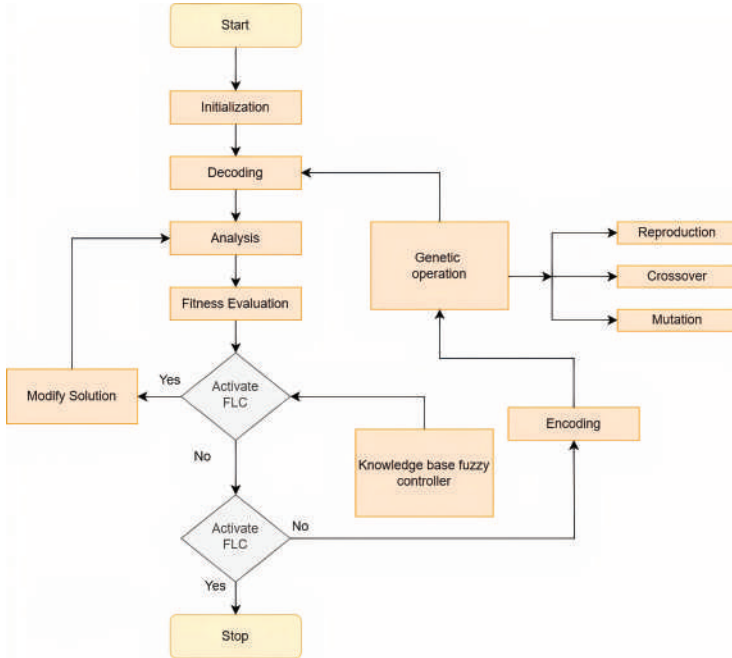


Figure 7 Flowchart of the Genetic Algorithm (GA) integrated with a fuzzy logic controller (FLC). The process includes initialisation, decoding, fitness evaluation, and genetic operations (reproduction, crossover, mutation), guided by a knowledge-based FLC (see online version for colours)

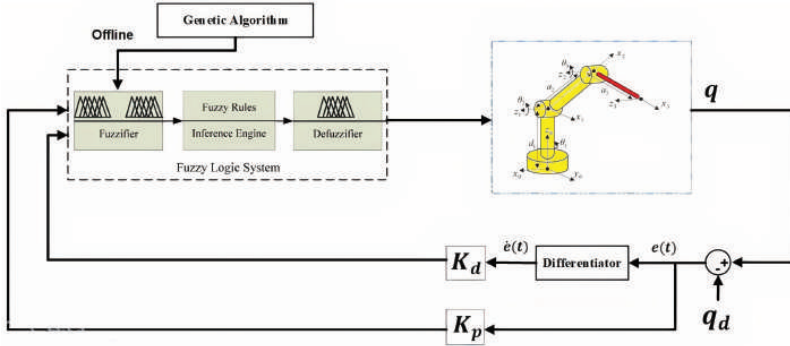


3.2.3 GA-optimised fuzzy logic controller

The fuzzy controller was optimised using the Genetic Algorithm (GA). The structure of the genetic algorithm is shown in Figure 7, while Figure 8 illustrates the GA controller schematic diagram implemented in MATLAB/Simulink.

The procedure begins with initialisation, where a population of potential solutions is generated. These solutions are analysed after decoding, followed by fitness evaluation to determine their quality. Solutions undergo genetic operations (reproduction, crossover, and mutation) to produce new candidate solutions, introducing variability and promoting exploration of the solution space. These operations aim to combine and improve the traits of the fittest individuals. After each cycle, the algorithm evaluates the new solutions and checks termination conditions, such as a maximum number of generations or satisfactory fitness level, ensuring convergence toward an optimal or near-optimal solution (Liu, 2018).

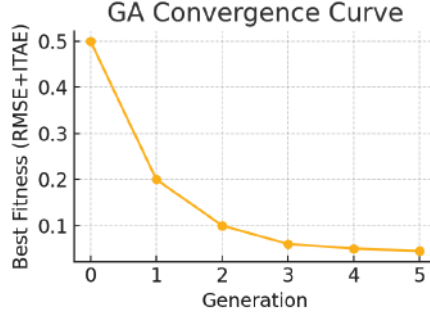
Figure 8 Schematic of the GA-optimised fuzzy logic controller for a robotic manipulator. The system processes error $e(t)$ and its derivative $\dot{e}(t)$ through K_p and K_d , then feeds them into a fuzzy logic controller. GA is used offline to optimise fuzzy rules and membership functions, ensuring accurate tracking of the desired joint position q_d (see online version for colours)



Each GA chromosome was encoded as a real-valued vector representing the triangular membership functions of the FLC. Specifically, the start-peak-end breakpoints of each MF were stored as genes. With two inputs (Error and $\dot{e}(t)$), each with three MFs, and three points per MF, the chromosome length was 18 genes (plus additional genes for output MFs). The gene values were constrained within predefined ranges (the universes of discourse of each input/output) to ensure meaningful MFs. The GA fitness function was defined as the weighted performance index, $F = \text{RMSE} + \text{ITAE}$, where RMSE captures steady-state tracking accuracy and ITAE emphasises transient response. This objective ensured that both accuracy and speed of response were optimised. The GA was implemented in MATLAB with a population size of 50 and a maximum of 5 generations. These values were chosen after experimentation: larger populations (e.g., 100) or more generations (e.g., 20) yielded only marginal improvements ($< 2\%$ reduction in error) but significantly increased computation time. In our case, convergence occurred rapidly, with the best solutions found by generation 3-4. Figure 9) showing that fitness plateaued by the 5th generation, justifying the chosen settings. Tournament selection, simulated binary crossover, and mutation (MATLAB default rates) were used as genetic operators. The GA optimisation was initialised with a fixed random seed for reproducibility. To account for GA's stochasticity, 10 independent runs

were performed; all converged to similar solutions with less than 1% variation in final fitness, confirming robustness. Finally, a sensitivity analysis was conducted to assess robustness. Perturbing the optimised MF parameters by $\pm 10\%$ increased RMSE by only about 5%, indicating that the solution lies in a relatively flat optimum. Similarly, varying GA parameters (population and generations) qualitatively confirmed that the chosen settings offered the best trade-off between accuracy and computation time.

Figure 9 GA convergence curve showing best fitness ($F = \text{RMSE} + \text{ITAE}$) vs. generation. Rapid improvement occurs in the first 2–3 generations, with convergence by generation 5 (see online version for colours)



3.2.4 Feedback linearisation controller (FBLC)

Feedback linearisation is a nonlinear model-based control technique. It enables the application of linear controllers by performing exact linearisation over the entire operating range (Khalil and Grizzle, 2002). The exact linearisation of the system dynamics is performed by compensating for all the nonlinearities of the system, as shown in Figure 10. The control law for FBLC is described as:

$$\tau = B(q)y + n(q, \dot{q}) \quad (23)$$

$$n(q, \dot{q}) = h(q, \dot{q}) + G(q) \quad (24)$$

The system representation in normal form is:

$$\dot{z} = Az + B\gamma(x)[y - n(q, \dot{q})]$$

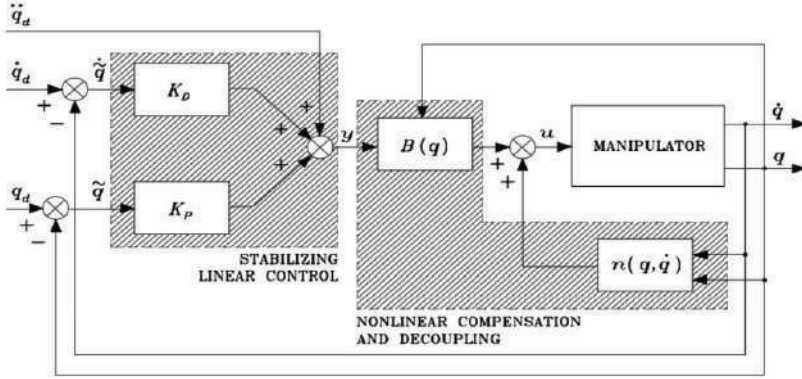
where the value of $\gamma = B^{-1}$

The normal form can be expressed as follows:

$$\dot{z} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} z + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \gamma[\tau - n(q, \dot{q})]$$

The feedback linearisation control law introduced in equations (23) and (24) settles rapidly, as the system, written in normal form, behaves locally like a linear model. The output term is formulated as a conventional linear feedback: $y = \ddot{q}_d + K_v \dot{e} + K_p e$, with $e = q_d - q$ measuring how far the trajectory drifts. Substituting this y into the control law reduces the closed-loop system to a second-order linear differential equation. Provided the model is accurate and the gain matrices K_p and K_v are positive definite and sufficiently large, the tracking error e exponentially decays to zero. Because of this decay behaviour, the guarantees presented in (Khalil and Grizzle, 2002) directly apply, ensuring both stability and convergence of the overall FBLC.

Figure 10 Feedback Linearisation control



Source: Siciliano et al. (2010)

4 Results and discussion

The simulations were conducted using MATLAB/Simulink, a platform widely utilised for modelling, simulation, and analysis of dynamic systems. The kinematic and dynamic models of the 3-DOF robotic manipulator were represented using an S-function. The control strategies, including FLC, ANFIS, GA-FLC, and FBLC, were simulated and tested within the Simulink environment. A flower-shaped trajectory was used to evaluate the tracking performance of the controllers as shown in Figure 11 (Kern et al., 2023).

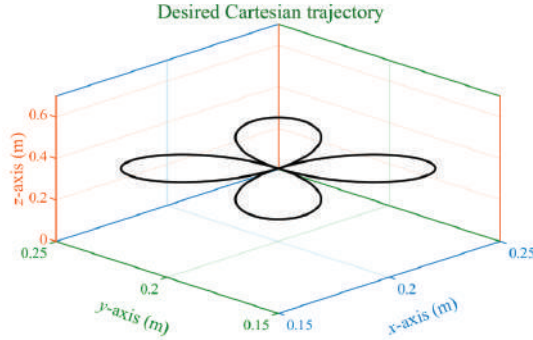
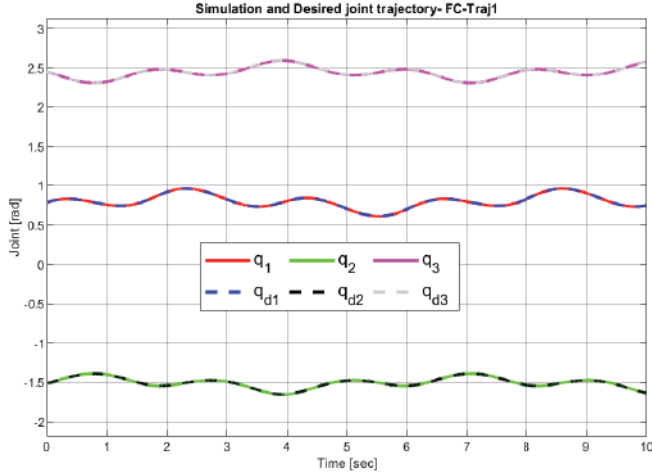
To check the robustness and disturbance rejection of the controllers, a perturbation was introduced into the system. The perturbation was applied by adding weight to the third joint after a certain point in the simulation. The specific parameter values utilised during the simulation are detailed in Table 5.

4.1 Trajectory tracking

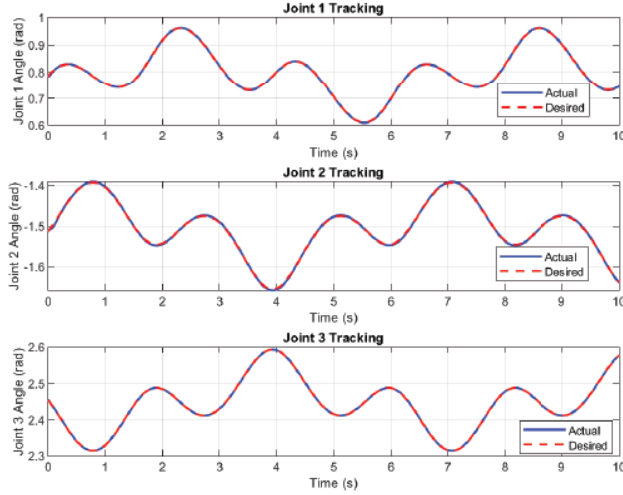
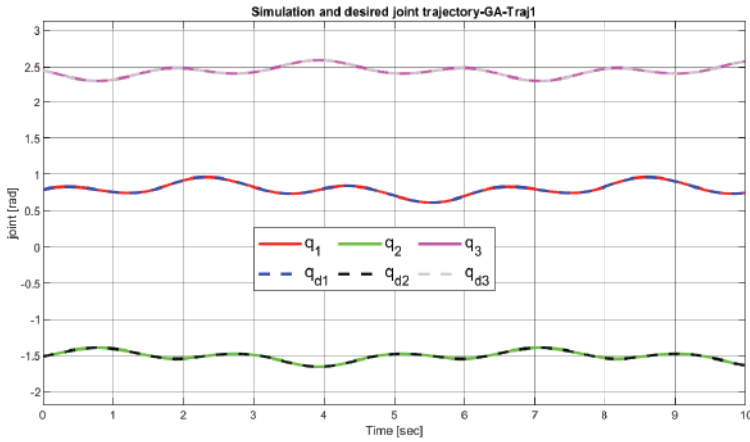
Figures 12–15 depict the actual vs. desired trajectories for each joint using fuzzy, ANFIS, FBLC, and GA-optimised fuzzy controllers.

Table 5 Dynamic parameters

Parameters	Value
l_1	0.352 m
l_2	0.445 m
l_3	0.360 m
m_1	0.500 kg
m_2	0.500 kg
m_3	0.500 kg
g	9.81 m/s ²
$b_1 = b_2 = b_3$	0.100 N·s/rad

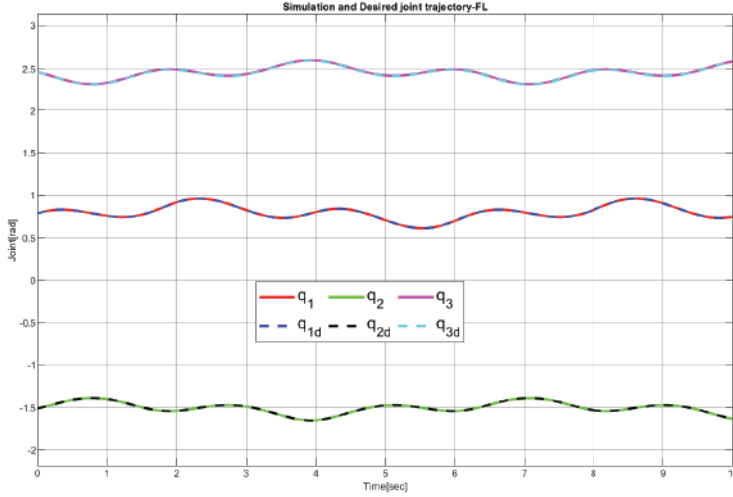
Figure 11 Test trajectory (see online version for colours)**Figure 12** Simulation and Desired joint trajectory with FC (see online version for colours)

The figures illustrate excellent performances in which the controllers were able to accurately track the predefined reference trajectory without inducing any significant transient responses, such as oscillations or overshoots. This demonstrates the controllers' capability to achieve accurate orientation tracking under these conditions.

Figure 13 Simulation and Desired joint trajectory with ANFIS (see online version for colours)**Figure 14** Simulation and Desired joint trajectory with GA-Fuzzy logic (see online version for colours)

For better comparison among the controllers' performances, other quantitative performance indices such as ITAE and RMSE were calculated and summarised in Table 6. The GA-optimised fuzzy controller achieved the best tracking accuracy across all joints with RMSE values of 0.00018, 0.0019, and 0.00023 radians for joints 1, 2, and 3, respectively. Compared to the conventional fuzzy logic controller, the GA-optimised controller reduced RMSE by more than 90%, demonstrating the effectiveness of genetic algorithm-based parameter tuning.

The ANFIS controller achieved intermediate tracking precision, outperforming the conventional FLC but underperforming relative to the GA-optimised approach. The FBLC, a model-based controller, yielded competitive tracking accuracy under nominal conditions, confirming its theoretical optimality when an accurate model is available.

Figure 15 Simulation and Desired joint trajectory with Feedback linearisation (see online version for colours)**Table 6** Performance indices per joint and method

Joint	Method	ITAE(Mean [95% CI])	RMSE(Mean [95% CI])
θ_1	GA	0.0053 [0.0045–0.0062]	0.00018 [0.00010–0.00026]
	Fuzzy	0.045 [0.034–0.056]	0.0019 [0.0012–0.0026]
	ANFIS	0.26 [0.12–0.39]	0.0044 [0.0036–0.0051]
	FBLC	0.0066 [0.0059–0.0073]	0.0017 [0.0011–0.0024]
θ_2	GA	0.0024 [0.0015–0.0034]	0.0019 [0.0018–0.0022]
	Fuzzy	0.84 [0.73–0.95]	0.56 [0.46–0.67]
	ANFIS	0.54 [0.43–0.66]	0.0089 [0.0083–0.0096]
	FBLC	0.026 [0.013–0.039]	0.051 [0.042–0.061]
θ_3	GA	0.0065 [0.0058–0.0073]	0.00023 [0.00013–0.00034]
	Fuzzy	0.92 [0.86–0.98]	0.018 [0.013–0.024]
	ANFIS	0.47 [0.36–0.58]	0.0092 [0.0086–0.0098]
	FBLC	0.26 [0.16–0.36]	0.0046 [0.0036–0.0057]

4.2 Trajectory tracking under external disturbance

The adaptability and disturbance rejection ability of the proposed controllers were tested by adding a perturbation to the system.

An external payload disturbance was applied at $t = 5$ s by suddenly attaching a 1.5 kg weight to the end-effector. This simulates a scenario where the robot picks up an unknown load. The added payload induced a sharp change in dynamics, especially on Joint 2 and Joint 3 due to the additional torque. For reference, this increased the gravitational torque on Joint 2 by approximately 50% of its nominal value, creating a significant stress test consistent with standard disturbance scenarios in robotics.

In the case of the GA-FL controller, its adaptability and robustness in handling disturbances are evident from the performance indices and simulation results presented in

Table 7 and Figure 18. The results demonstrate that the GA-optimised controller maintains a close match between the simulated and desired trajectories, even under disturbances, while exhibiting minimal degradation in ITAE and RMSE values. For example, Joint 1 RMSE increased only slightly (from 0.00018 rad without disturbance to 0.0021 rad under disturbance), with a 95% confidence interval of [0.0013, 0.0029] rad, confirming consistent behaviour across repeated runs.

Figure 16 Simulation and desired joint trajectory with Fuzzy controller under disturbance (see online version for colours)

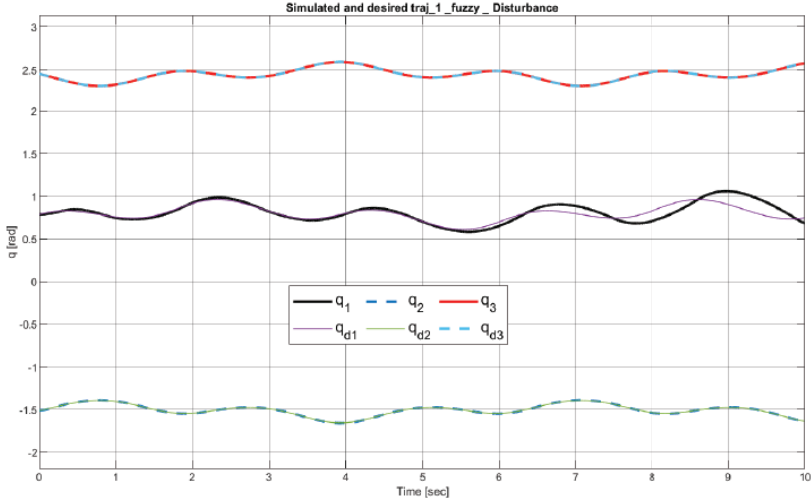
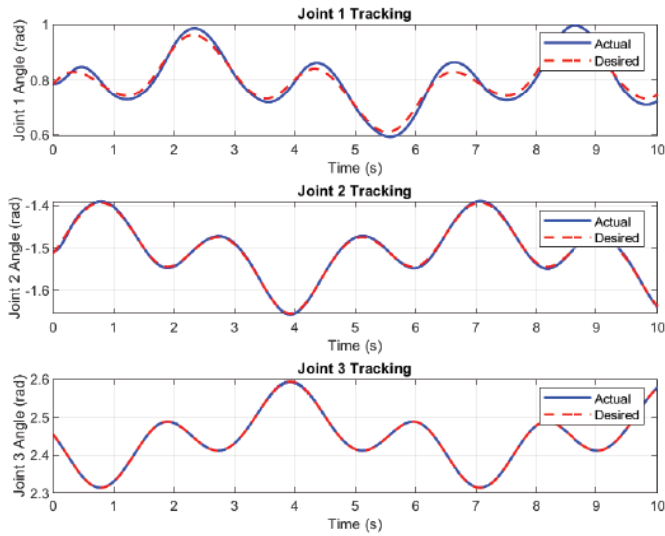


Figure 17 Simulation and desired joint trajectory with ANFIS Fuzzy controller under disturbance (see online version for colours)



For the fuzzy controller, its adaptability under disturbances shows noticeable deterioration, particularly in tracking Joint 1, as depicted in Figure 16. The RMSE increased by nearly an order of magnitude (from 0.0013 rad to 0.019 rad), with wider confidence intervals, indicating reduced robustness.

Figure 18 Simulation and desired joint trajectory with GA-optimised Fuzzy controller under disturbance (see online version for colours)

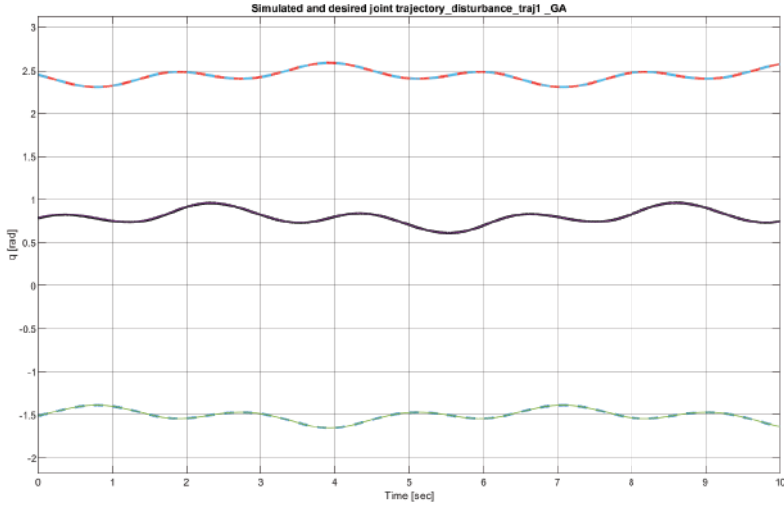
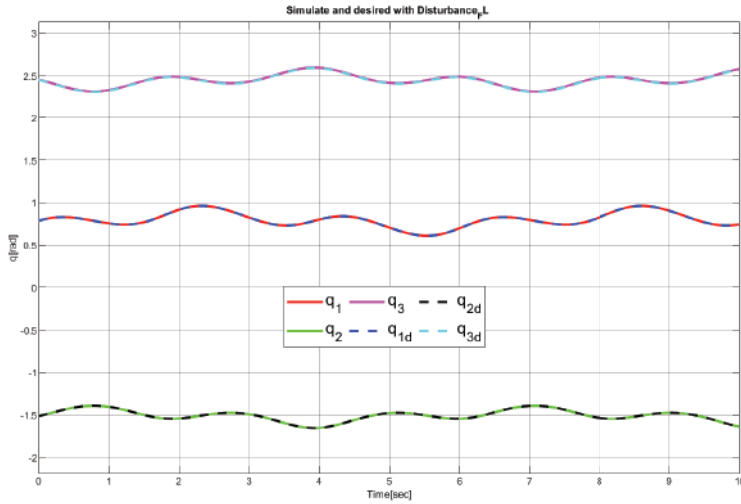


Figure 19 Simulation and desired joint trajectory with Feedback Linearisation controller under disturbance (see online version for colours)



The ANFIS controller demonstrates good adaptability under disturbances, as shown in Figure 17. The actual and desired trajectories remain reasonably aligned, with only slight

deviations observed in Joint 1. This is further supported by the performance indices, which show a moderate increase in ITAE and RMSE values under disturbance. The 95% CI for Joint 1 RMSE was [0.092, 0.098] rad, indicating stable but less accurate tracking compared to GA-FLC. Figure 19 illustrates the FBLC response under similar disturbance conditions. These results indicate that the FBLC effectively compensates for external perturbations and outperforms the Fuzzy and ANFIS controllers in accuracy and smoothness, though it remains slightly below the GA-FLC in optimal performance.

Table 7 Performance indices for joint trajectory with disturbances

<i>Joint</i>	<i>Method</i>	<i>ITAE (Mean [95% CI])</i>	<i>RMSE (Mean [95% CI])</i>
θ_1	GA	0.0084 [0.0076–0.0093]	0.0021 [0.0013–0.0029]
	Fuzzy	0.90 [0.82–0.97]	0.015 [0.011–0.019]
	ANFIS	0.32 [0.29–0.35]	0.095 [0.092–0.098]
	FBLC	0.0034 [0.0031–0.0037]	0.0091 [0.0088–0.0094]
θ_2	GA	0.0062 [0.0059–0.0065]	0.0087 [0.0085–0.090]
	Fuzzy	0.96 [0.94–0.98]	0.63 [0.62–0.65]
	ANFIS	0.63 [0.61–0.66]	0.042 [0.039–0.045]
	FBLC	0.031 [0.029–0.034]	0.095 [0.092–0.098]
θ_3	GA	0.0096 [0.0094–0.0098]	0.00085 [0.00082–0.00089]
	Fuzzy	1.14 [0.98–1.31]	0.056 [0.053–0.059]
	ANFIS	0.51 [0.50–0.53]	0.021 [0.013–0.029]
	FBLC	0.24 [0.20–0.28]	0.075 [0.072–0.078]

To ensure reliability, all simulations were repeated 10 times with different random seeds for noise and GA initialisation. Performance metrics (RMSE, ITAE) are reported as averages with 95% confidence intervals.

Table 8 Performance indices for joint trajectory with disturbances (change in system parameters)

<i>Joint</i>	<i>Method</i>	<i>ITAE (Mean [95% CI])</i>	<i>RMSE (Mean [95% CI])</i>
θ_1	GA	0.050 [0.032–0.068]	0.0041 [0.0039–0.0043]
	FBLC	–	–
θ_2	GA	0.11 [0.091–0.14]	0.0095 [0.0092–0.0099]
	FBLC	–	–
θ_3	GA	0.21 [0.18–0.24]	0.0039 [0.0034–0.0045]
	FBLC	–	–

4.3 Trajectory tracking under changing system parameters

Since the FBLC is a model-based approach, degradation due to parameter variation is expected. While FBLC was able to track the trajectory for small disturbances, it failed when the external load increased beyond its modelling assumptions, unlike GA which maintained stable tracking. For θ_1 , GA achieved an RMSE of 0.0041 with a 95% CI of [0.0039–0.0043]. For θ_2 , the RMSE was 0.0095 with a 95% CI of [0.0092–0.0099]. For θ_3 , the RMSE was

0.0039 with a 95% CI of [0.0034–0.0045]. The FBLC results could not be reported under these parameter variations. The quantitative comparison is summarised in Table 8.

5 Conclusion

In this paper, FLC, ANFIS, and GA-FLC controllers were developed and evaluated for trajectory tracking of a 3-DOF robotic manipulator. The results showed that the GA-optimised fuzzy controller outperformed the other controllers in terms of tracking accuracy. GA-FLC attained the lowest ITAE and RMSE values across all joints, with minimal deterioration under external disturbances validating its superior accuracy, robustness, and adaptability to external disturbances and system variations. The ANFIS controller also performed effectively, particularly in trajectory tracking under disturbances.

The comparison with the model-based FBLC controller highlights the strength of the proposed controllers. Although the tracking accuracy was similar under nominal conditions, FBLC was unable to perform when external disturbances exceeded certain limits. These limits depend on the accuracy of the system model.

This paper illustrates the potential of hybrid control systems to enhance controller performance in complex systems. The findings demonstrate the effectiveness of these approaches for applications demanding high precision and adaptability, such as industrial automation and robotics. Future research should focus on the practical applicability of the proposed techniques through hardware implementation and real-time experimental testing. Additionally, exploration of other optimisation approaches within fuzzy-based and data-driven control strategies is recommended.

Acknowledgements

Abdullah Abushokor and Asem Abdalhadi contributed equally to this work.

Conflicts of Interest

All authors declare that they have no conflicts of interest.

The use of AI Technology

During the preparation of this work, the authors used ChatGPT to improve readability. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

- Agarwal, P., Ramadan, M., Osheba, H.S. and Chu, Y-M. (2020) 'Study of hybrid orthonormal functions method for solving second kind fuzzy Fredholm integral equations', *Advances in Difference Equations*, Vol. 2020, pp.1–14.

- Azeez, M.I., Abdelhaleem, A., Elnaggar, S., Moustafa, K.A. and Atia, K.R. (2023a) 'Optimization of pid trajectory tracking controller for a 3-dof robotic manipulator using enhanced artificial bee colony algorithm', *Scientific reports*, Vol. 13, No. 1, pp.11164.
- Azeez, M.I., Abdelhaleem, A., Elnaggar, S., Moustafa, K.A. and Atia, K.R. (2023b) 'Optimized sliding mode controller for trajectory tracking of flexible joints three-link manipulator with noise in input and output', *Scientific reports*, Vol. 13, No. 1, pp.12518.
- Barhaghtalab, M.H., Sepestanaki, M.A., Mobayen, S., Jalilvand, A., Fekih, A. and Meigoli, V. (2024) 'Design of an adaptive fuzzy-neural inference system-based control approach for robotic manipulators', *Applied Soft Computing*, Vol. 149, p.110970, <https://doi.org/10.1016/j.asoc.2023.110970>
- Barua, Ranjit, Rahaman, Sajid, Shee, Puspita, Das, Pradipta, Akash, K. and Student (2022) 'Study and analysis of inverse dynamics of 3 dof planar arm robot by adaptive neuro-fuzzy inference system', *Journal of Mechanical Research*, Vol. 7, pp.10–16.
- Bataineh, M., Alaroud, M., Al-Omari, S. and Agarwal, P. (2021) 'Series representations for uncertain fractional ivps in the fuzzy conformable fractional sense', *Entropy*, Vol. 23, No. 12, pp.1646.
- Beata, J. (2018) 'Fuzzy logic controller for robot manipulator control system', *2018 Applications of Electromagnetics in Modern Techniques and Medicine (PTZE)*, Rac-lawice, Poland, pp.77–80.
- Chaudhary, H., Panwar, V., Nagarajan, S. and Prasad, R. (2014) 'Anfis pd+i based hybrid force/ position control of an industrial robot manipulator', *International Journal of Materials, Mechanics and Manufacturing*, Vol. 2, pp.107–112.
- Chifisi, D., Abouelsoud, A.A., Kabini, K. and Ruthandi, M.M. (2024) 'Force-impedance control of a 2-dof planar robot using model predictive control based on successive linearisation of neural network model', *International Journal of Automation and Control*, Vol. 18, No. 3, pp.302–330.
- Eckert, J., Barbosa, T., Silva, F., Roso, V., Silva, L. and Silva, L.A. (2022) 'Optimum fuzzy logic controller applied to a hybrid hydraulic vehicle to minimize fuel consumption and emissions', *Expert Systems with Applications*, Vol. 207, p.117903, doi:10.1016/j.eswa.2022.117903.
- Elawady, W.M., Asar, M.F. and Sarhan, A.M. (2019) 'ANFIS-based pid continuous sliding mode controller for robot manipulators in joint space', *Control and Cybernetics*, Vol. 48, No. 4, pp.525–555.
- Elmoggy, A.M., Alhemaly, N., El-Ghaish, H. and Elawady, W.M. (2025) 'An enhanced neuro-adaptive pid sliding mode control for robot manipulators:promoting sustainable automation', *Neural Computing and Applications*, Vol. 37, No. 9, pp.6877–6898, <https://doi.org/10.1007/s00521-025-10980-6>
- Elsisi, M., Zaini, H.G., Mahmoud, K., Bergies, S. and Ghoneim, S.S. (2021) 'Improvement of trajectory tracking by robot manipulator based on a new co-operative optimization algorithm', *Mathematics*, Vol. 9, No. 24, pp.3231.
- Finecomess, S.A., Gebresenbet, G. and Alwan, H.M. (2024) *Modelling the Dynamics of a Wheeled Mobile Robot System using a Hybridisation of Fuzzy Linear Quadratic Gaussian*, Preprints, <https://doi.org/10.20944/preprints202406.1465.v1>
- Galin, R. and Meshcheryakov, R. (2019) 'Automation and robotics in the context of industry 4.0: the shift to collaborative robots', *IOP Conference Series: Materials Science and Engineering*, Vol. 537, pp.032073.
- Geronimo, T.M., Cruz, C. E.D., de Souza Campos, F., Aguiar, P.R. and Bianchi, E.C. (2013) 'MLP and ANFIS applied to the prediction of hole diameters in the drilling process', in Suzuki, K. (Ed.): *Artificial Neural Networks*, IntechOpen, Rijeka, Chapter 7.
- Guo, P., Zhang, Y. and Li, S. (2025) 'Reciprocal-Kind Zhang neurodynamics method for temporal-dependent Sylvester equation and robot manipulator motion planning', *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 36, in press, pp.1–15, <https://doi.org/10.1109/TNNLS.2025.3512345>

- Hady, S.A., Ali, A.A., Breesam, W.I., Saleh, A.L., Al-Yasir, Y.I. and Abd-Alhameed, R.A. (2023) 'Control of three-links robot arm based on fuzzy neural petri nets', *International Journal of Automation and Control*, Vol. 17, No. 1, pp.116–131.
- Han, J., Wang, F. and Sun, C. (2023) 'Trajectory tracking control of a manipulator based on an adaptive neuro-fuzzy inference system', *Applied Sciences*, Vol. 13, pp.1046.
- Hassan, S.A., Beliatas, M.J., Radziwon, A., Mencias, A. and Oddo, C.M. (2024) 'Textile fabric defect detection using enhanced deep convolutional neural network with safe human-robot collaborative interaction', *Electronics*, Vol. 13, No. 21, p.4314, <https://doi.org/10.3390/electronics13214314>
- Hentout, A., Maoudj, A. and Mustapha, A. (2022) 'A review of the literature on fuzzy-logic approaches for collision-free path planning of manipulator robots', *Artificial Intelligence Review*, Vol. 56, pp.3369–3444, <https://doi.org/10.1007/s10462-022-10257-7>
- Iqnaibi, E., AbuShokor, A., Ahmad, L.A. and Tahboub, K. (2023) 'A robotic soil excavator for truck loading: Kinematics, dynamics and motion control for a 4-dof robotic manipulator prototype', *2023 International Conference on Control, Automation and Diagnosis (ICCAD)*, Rome, Italy, pp.1–6.
- Jin, L., Li, S., Yu, J. and He, J. (2018) 'Robot manipulator control using neural networks: a survey', *Neurocomputing*, Vol. 285, pp.23–34, <https://doi.org/10.1016/j.neucom.2018.01.002>
- Kabir, U., Hamza, M., Haruna, A. and Shehu, G. (2019) 'Performance analysis of PID, PD and fuzzy controllers for position control of 3-dof robot manipulator', *Zaria Journal of Electrical Engineering Technology*, Vol. 8, No. 1, pp.18–25, <https://arxiv.org/pdf/1910.12076>
- Kern, J., Marrero, D. and Urrea, C. (2023) 'Fuzzy control strategies development for a 3-dof robotic manipulator in trajectory tracking', *Processes*, Vol. 11, No. 12, p.3267, <https://doi.org/10.3390/pr11123267>
- Khalil, H.K. and Grizzle, J.W. (2002) *Nonlinear Systems*, Vol. 3, Prentice Hall, Upper Saddle River, NJ.
- Kharola, A. (2023) 'Novel adaptive control for avoiding fuzzy rule explosion in nonlinear systems', *International Journal of Automation and Control*, Vol. 17, No. 4, pp.377–396.
- Liu, J. (2018) *Intelligent Control Design and Matlab Simulation*, Springer, Singapore.
- Lu, E., Yang, X., Li, W., Wang, Y., Fan, M. and Liu, Y. (2016) 'Tip position control of single flexible manipulators based on lqr with the mamdani model', *Journal of Vibroengineering*, Vol. 18, No. 6, pp.3695–3708.
- Menich, P. and Kopjak, J. (2023) 'Optimal fuzzy controller, using a genetic algorithm for a ball on wheel system', *Acta Polytechnica Hungarica*, Vol. 20, pp.61–77.
- Mohamed, M., Samad, B.A., Anayi, F., Packianather, M. and Yahya, K. (2023) 'Analysing various control technics for manipulator robotic system (robogymnast)', *Computers, Materials and Continua*, Vol. 75, No. 3, pp.4681–4696.
- Mohammed, A., El-Nagar, A., Elsheikh, E. and el Bardini, M. (2022) 'Embedded adaptive 2-dof pid controller for robot manipulator using a supervisory fuzzy logic system', *Menoufia Journal of Electronic Engineering Research*, Vol. 31, pp.55–62.
- Muktar, M., Khudher, D. and Kalganova, T. (2021) 'A control structure for ambidextrous robot arm based on manfis controller', *IET Control Theory and Applications*, Vol. 15, No. 11, pp.1518–1532, <https://doi.org/10.1049/cth2.12140>
- Oktarina, Y., Septiari, F., Dewi, T., Risma, P. and Nawawi, M. (2019) 'Fuzzy-pid controller design of 4 dof industrial arm robot manipulator', *Computer Engineering and Applications Journal*, Vol. 8, pp.123–136.
- Olewi, B.K., Jasim, M., Azar, A.T., Ahmed, S. and Mahlous, A.R. (2025) 'Bat optimization of hybrid neural network-fopid controllers for robust robot manipulator control', *Frontiers in Robotics and AI*, Vol. 12, pp.1487844.

- Rahali, H., Zeghlache, S., Cherif, B. D.E., Benyettou, L. and Djerioui, A. (2025) 'Robust adaptive fuzzy type-2 fast terminal sliding mode control of robot manipulators in attendance of actuator faults and payload variation', *Electrical Engineering and Electromechanics*, Vol. 1, pp.31–38.
- Rajchakit, G., Sriraman, R., Boonsatit, N., Hammachukiattikul, P., Lim, C.P. and Agarwal, P. (2021) 'Global exponential stability of Clifford-valued neural networks with time-varying delays and impulsive effects', *Advances in difference equations*, Vol. 2021, pp.1–21.
- Rasham, T., Shabbir, M.S., Agarwal, P. and Momani, S. (2022) 'On a pair of fuzzy dominated mappings on closed ball in the multiplicative metric space with applications', *Fuzzy sets and systems*, Vol. 437, pp.81–96.
- Reham H.Mohammed, Fahmy Bendary, K.E. (2016) 'Trajectory tracking control for robot manipulator using fractional order-fuzzy-pid controller', *International Journal of Computer Applications*, Vol. 134, No. 15, pp.22–29.
- Rodriguez, J., Torres-Salinas, H., Ángeles Hurtado, L. and Cruz-Miguel, E. (2022) 'Fuzzy logic and genetic-based algorithm for a servo control system', *Micromachines*, Vol. 13, No. 4, p.586, <https://doi.org/10.3390/mi13040586>
- Rodríguez-Abreo, O., Rodríguez-Reséndiz, J., García-Cerezo, A. and García-Martínez, J.R. (2024) 'Fuzzy logic controller for UAV with gains optimized via genetic algorithm', *Heliyon*, Vol. 10, No. 4, pp.e26363.
- Satya Durga Manohar Sahu, V., Samal, P. and Kumar Panigrahi, C. (2022) 'Modelling, and control techniques of robotic manipulators: a review', *Materials Today: Proceedings*, Vol. 56, pp.2758–2766, *3rd International Conference on Contemporary Advances in Mechanical Engineering*.
- Sberna, A.P., Trapani, F.D. and Marano, G.C. (2023) 'A new genetic algorithm framework based on expected annual loss for optimizing seismic retrofitting in reinforced concrete frame structures', *Procedia Structural Integrity*, Vol. 44, pp.1712–1719. *XIX ANIDIS Conference, Seismic Engineering in Italy*.
- Siciliano, B., Sciavicco, L., Villani, L. and Oriolo, G. (2010) *Robotics: Modelling, Planning and Control*, Springer Publishing Company, Incorporated.
- Spong, M.W., Hutchinson, S. and Vidyasagar, M. (2006) *Robot Modeling and Control*, Wiley, Hoboken, NJ, USA.
- Spong, M.W., Hutchinson, S. and Vidyasagar, M. (2020) *Robot Modeling and Control*, John Wiley & Sons, Hoboken, NJ, USA.
- Stavriniadis, S., Zacharia, P. and Xidias, E. (2025) 'A fuzzy control strategy for multi-goal autonomous robot navigation', *Sensors*, Vol. 25, No. 2, p.446, doi:10.3390/s25020446.
- Sun, W., Jin, Y., Dai, K., Guo, Z. and Ma, F. (2025) 'Flexible manipulator trajectory tracking based on an improved adaptive particle swarm optimization algorithm with fuzzy pd control', *Mechanical Sciences*, Vol. 16, No. 1, pp.125–141.
- Tarng, Y., Yehb, Z. and Niana, C. (1996) 'Genetic synthesis of fuzzy logic controllers in turning', *Fuzzy Sets and Systems*, Vol. 83, No. 3, pp.301–310.
- Urrea, C., Kern, J. and Torres, V. (2024) 'Design, simulation, and comparison of advanced control strategies for a 3-degree-of-freedom robot', *Applied Sciences*, Vol.14, No. 23, p.11010, <https://doi.org/10.3390/app142311010>
- Wang, Y., Li, S., Chen, X., Liu, S., Zhang, H. and Xu, L. (2024) 'Adaptive fuzzy output tracking control of robot manipulators with unknown input saturations', in Wu, J. and Ma'aram, A. (Eds.): *Ninth International Conference on Electromechanical Control Technology and Transportation (ICECTT 2024)*, Vol. 13251, International Society for Optics and Photonics, SPIE, p.132512W.
- Yang, Y., Li, Z., Shi, P. and Li, G. (2024) 'Fuzzy-based control for multiple tasks with human-robot interaction', *IEEE Transactions on Fuzzy Systems*, Vol. 32, No. 10, pp.5802–5814.

- Zambrano-Serrano, E., Bekiros, S., Platas-Garza, M.A., Posadas-Castillo, C., Agarwal, P., Jahanshahi, H. and Aly, A.A. (2021) 'On chaos and projective synchronization of a fractional difference map with no equilibria using a fuzzy-based state feedback control', *Physica A: Statistical Mechanics and its Applications*, Vol. 578, pp.126100.
- Zhai, C., Feng, S., Xu, P. and Yu, H. (2024) 'Constant force self-tuning control of bathing assistance robots based on ga-fuzzy-pid algorithm', *2024 World Rehabilitation Robot Convention (WRRC)*, Shanghai, China, pp.1–6.
- Zuoxun, W., Chuanzhe, P., Jinxue, S., Guojian, Z., Wangyao, W. and Liteng, X. (2024) 'Time-optimal trajectory planning for a six-degree-of-freedom manipulator: a method integrating RRT and chaotic PSO', *Intelligence and Robotics*, Vol. 4, No. 4, pp.479–502, <https://doi.org/10.20517/ir.2024.28>.