International Journal of Engineering Inventions
e-ISSN: 2278-7461, p-ISSN: 2319-6491
Volume 15, Issue 11 [November 2026] PP: 129-137

Trajectory Tracking Control of A 2-Dof Robot Manipulator Using PSO-Optimized ANFIS Trained on Fuzzy-Pd Control Data

Van-Luong Nguyen[1], Duy-Trung Nguyen[1], Mai-Phuong Dao-Thi[2], N. -K. Nguyen[1, *]

[1]*Faculty of Automation, Artificial Intelligence and Intelligent Systems, Electric Power University, Hanoi, Vietnam*

[2]*Faculty of Automation, School of Electrical and Electronic Engineering, Hanoi University of Industry, Hanoi, Vietnam*

[*]*Corresponding Author*

ABSTRACT: *Trajectory tracking in robot manipulators is challenging because of nonlinear dynamics and interactions between joint motions. This study investigates a learning-based control framework for a two-degree-of-freedom (2-DOF) robot manipulator by integrating an expert-designed fuzzy proportional–derivative (Fuzzy-PD) controller, an adaptive neuro-fuzzy inference system (ANFIS), and particle swarm optimization (PSO). The Fuzzy-PD controller first provides a baseline tracking policy that incorporates expert knowledge into the control response. State data and the corresponding control signals generated by this controller are collected to construct a supervised learning dataset for a five-layer ANFIS architecture. Through this learning process, ANFIS approximates the nonlinear mapping between the selected inputs and the reference control actions, transferring the baseline controller's behavior to a trainable neuro-fuzzy representation. PSO is incorporated into parameter optimization to explore the ANFIS search space and reduce dependence on the initialization and local search characteristics of conventional hybrid learning. The resulting controller is evaluated through numerical simulations of the 2-DOF manipulator performing trajectory tracking tasks. The reported responses indicate accurate reference tracking and rapid convergence of tracking errors under the simulated conditions. The framework offers a systematic approach to combining expert control knowledge, data-driven function approximation, and population-based optimization for nonlinear robotic control. Its practical value lies in reducing reliance on manual fuzzy-controller tuning while retaining a structured neuro-fuzzy representation of the control policy.*

NOMENCLATURE

Symbol	***Description***	***Unit***
J1, J2	*Moment of inertia of the two joints*	*kg.m2*
m1, m2	*Mass of the two joints*	*kg*
l1, l2	*Length of the two links*	*m*
lc1, lc2	*Distance from the center of mass to the axis of rotation for each link*	*m*
g	*Gravitational acceleration*	*m/s2*
θ1, θ2	*Rotation angle of joint 1 and joint 2*	*rad*
τ1, τ2	*Total torque at Link 1 and Link 2*	*N.m*

Date of Submission: 02-11-2026 Date of acceptance: 12-11-2026

I. INTRODUCTION

Accurate trajectory tracking is essential to robotic tasks that require coordinated motion along a prescribed path. Even a two-degree-of-freedom (2-DOF) manipulator exhibits configuration-dependent inertia, velocity-dependent coupling, and gravitational effects, making it a useful platform for investigating nonlinear control methods [1]. Friction, uncertain physical parameters, and external disturbances introduce additional challenges. Although proportional–derivative (PD) and proportional–integral–derivative (PID) controllers offer simple feedback structures, their performance depends on gain selection and the extent to which nonlinear dynamics are compensated [2]. Maintaining tracking accuracy across different motions therefore motivates control strategies that can represent nonlinear input–output relationships without relying exclusively on an exact inverse dynamic model.

Fuzzy logic provides a framework for incorporating qualitative operating knowledge into controller design. Zadeh's formulation of fuzzy sets allows variables to be represented through graded membership rather than binary classification [3]. Takagi and Sugeno subsequently established a fuzzy modeling approach that combines rule-based antecedents with parameterized consequent functions [4]. In manipulator control, these ideas allow feedback actions to vary nonlinearly with the selected error variables. Ahmad et al. investigated a PD-type fuzzy controller as part of composite schemes for motion tracking and vibration suppression in a flexible-joint manipulator [5]. Urrea et al. developed a fuzzy controller incorporating acceleration information for a 2-DOF manipulator and assessed its tracking performance against a conventional PID controller [6]. Such studies demonstrate the relevance of fuzzy feedback to robotic motion control. Nevertheless, selecting membership functions, scaling factors, and rule consequents remains a substantial design task, and a fixed fuzzy controller does not acquire a learning mechanism merely through the use of linguistic rules.

ANFIS offers a way to combine fuzzy inference with parameter learning from numerical data. In Jang's formulation, an adaptive network implements a fuzzy inference system and adjusts its parameters using input–output observations [7]. This structure makes it possible to approximate a nonlinear mapping while retaining an explicit connection to fuzzy rules. Kern et al. compared fuzzy and ANFIS controllers for a 3-DOF manipulator, reporting improved tracking indices for ANFIS under their evaluation conditions and emphasizing the importance of parameter tuning and training data [8]. The performance of a learned controller, however, depends on what the dataset represents. Learning an inverse dynamic model and learning the control policy of an expert controller are distinct tasks. In the latter case, the training targets are the expert's control actions, and the immediate learning objective is to reproduce that policy over the sampled operating region.

Parameter optimization is a further consideration in ANFIS design. Conventional hybrid learning combines least-squares estimation of consequent parameters with gradient-based adjustment of premise parameters. Its outcome can depend on initialization, learning settings, and the nonconvex optimization landscape. PSO, introduced by Kennedy and Eberhart, provides a derivative-free search method in which candidate solutions exchange information through individual and swarm experience [9]. Shoorehdeli et al. investigated ANFIS identification using PSO for antecedent training and an extended Kalman filter for consequent estimation [10]. Seydi et al. applied a modified PSO method to train ANFIS parameters for nonlinear system identification [11], while Liu et al. combined an improved quantum-behaved PSO algorithm with least-squares estimation [12]. These studies establish precedents for swarm-based neuro-fuzzy training. They also indicate that the parameter subset assigned to the optimizer must be specified explicitly. A population-based search can explore alternative parameter configurations, but it does not guarantee that the global optimum will be reached.

The use of swarm optimization in motion control has also been investigated beyond system identification. Selma et al. developed an ANFIS–PSO controller for quadrotor trajectory tracking and compared its simulated performance with ANFIS and PID alternatives [13]. For robot manipulators, Mahmoodabadi and Ziaei combined inverse dynamics with a PSO-optimized fuzzy PD controller for an RPP mechanism [14]. These contributions support the relevance of optimization-assisted intelligent control, although their plant structures and control architectures differ from the present application. Closed-loop properties also require separate attention: Li et al., for example, established boundedness and tracking results for robust adaptive backstepping controllers using radial basis function neural networks [15]. Their results concern a specific controller design and cannot be transferred directly to an ANFIS trained by supervised learning. Consequently, neither a low approximation error nor successful parameter optimization alone establishes closed-loop stability or robustness.

The present study focuses on the transfer of an expert-designed Fuzzy-PD control policy to a PSO-optimized ANFIS for trajectory tracking of a 2-DOF manipulator. The research question is whether a neuro-fuzzy controller learned from expert-generated data can maintain accurate tracking when used in the feedback loop. This focus distinguishes the learning task from inverse-dynamics identification and places the quality of the expert-generated mapping at the center of controller development. Since ANFIS–PSO combinations have already been reported, the contribution is framed around the particular data-generation, learning, and control

workflow rather than the combination of these methods alone.

The proposed framework first uses the Fuzzy-PD controller to generate paired observations of the selected state variables and corresponding control actions. These data provide supervised targets for a five-layer ANFIS that approximates the expert controller's nonlinear mapping. PSO is then used to search the selected ANFIS parameter space according to the prescribed learning objective. The trained controller is subsequently evaluated in closed-loop trajectory tracking simulations. This sequence connects expert knowledge with numerical learning while distinguishing the accuracy of policy approximation from the tracking behavior of the controlled manipulator.

The study brings together three elements: expert-guided generation of control data, neuro-fuzzy approximation of the resulting feedback policy, and PSO-based parameter selection. Its simulation assessment addresses reference tracking and transient error convergence for the 2-DOF system. The intended outcome is a systematic procedure for transferring a designed fuzzy control policy into a trainable representation and evaluating its closed-loop behavior. The resulting ANFIS retains fuzzy membership evaluation, rule activation, normalization, and consequent aggregation; its role is to parameterize and learn the control mapping, with performance conclusions confined to the conditions examined in simulation.

II. METHODOLOGIES

2.1. Kinematic and Dynamic Modeling of the 2-DOF Robot

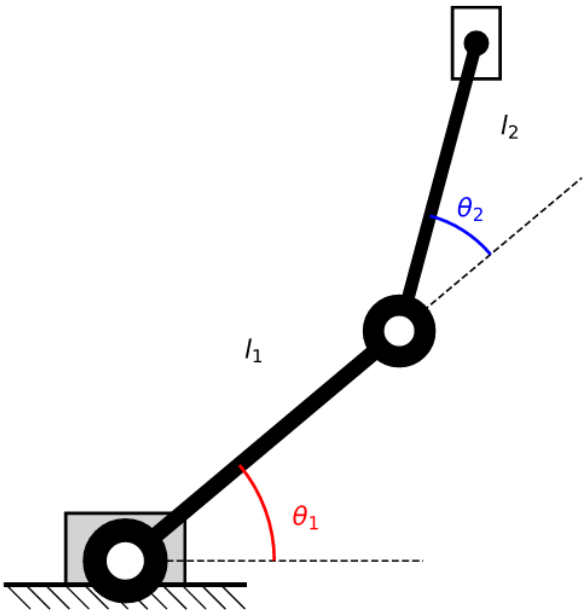


Fig. 1 Geometric model of a 2-DOF robot

The total kinetic energy of the system is:

$$K = \frac{1}{2}m_1\left(\dot{\theta}_1 l_{c1}\right)^2 + \frac{1}{2}J_1\dot{\theta}_1^{\,2} + \frac{1}{2}J_2\left(\dot{\theta}_1+\dot{\theta}_2\right)^2 + \frac{1}{2}m_2\left[\left(\dot{\theta}_1 l_1\right)^2 + \left(\left(\dot{\theta}_1+\dot{\theta}_2\right)l_{c2}\right)^2 + 2\dot{\theta}_1 l_1\left(\dot{\theta}_1+\dot{\theta}_2\right)l_{c2}\cos\theta_2\right] \quad (1)$$

The total potential energy of the system is:

$$P = m_1 g l_{c1}\sin\theta_1 + m_2 g\left[l_1\sin\theta_1 + l_{c2}\sin\left(\theta_1+\theta_2\right)\right] \quad (2)$$

The Lagrange equation corresponding to the two-degree-of-freedom robot:

$$L\left(\theta,\dot{\theta}\right)=K-P=\frac{1}{2}m_1\left(\dot{\theta}_1 l_{c1}\right)^2+\frac{1}{2}J_1\dot{\theta}_1^{\,2}+\frac{1}{2}J_2\left(\dot{\theta}_1+\dot{\theta}_2\right)^2-m_1 g l_{c1}\sin\theta-m_2 g\left[l_1\sin\theta_1+l_{c2}\sin\left(\theta_1+\theta_2\right)\right]$$
$$+\frac{1}{2}m_2\left[\left(\dot{\theta}_1 l_1\right)^2+\left(\left(\dot{\theta}_1+\dot{\theta}_2\right)l_{c2}\right)^2+2\dot{\theta}_1 l_1\left(\dot{\theta}_1+\dot{\theta}_2\right)l_{c2}\cos\theta_2\right] \tag{3}$$

with $s_1=\sin\left(\theta_1\right), c_1=sos\left(\theta_1\right), s_{12}=\sin\left(\theta_1+\theta_2\right)$

According to the Lagrange equation, the nonlinear equation is established

The total moment at Link 1 is:

$$\tau_1=\left(J_1+J_2+m_1 l_{c1}^2+m_2 l_1^2+m_2 l_{c2}^2+2m_2 l_1 l_{c2}\cos\theta_2\right)\ddot{\theta}_1+\left(J_2+m_2 l_{c2}^2+2m_2 l_1 l_{c2}\cos\theta_2\right)\ddot{\theta}_2$$
$$-l_1 l_{c2} m_2\sin\theta_2\left(2\dot{\theta}_1\dot{\theta}_2+\dot{\theta}_2^2\right)+\left(m_1+m_2\right)g l_{c1}\cos\theta_1+m_2 g l_{c2}\cos\left(\theta_1+\theta_2\right) \tag{4}$$

The total moment at Link 2 is:

$$\tau_2=\left(J_2+m_2 l_{c2}^2+2m_2 l_1 l_{c2}\cos\theta_2\right)\ddot{\theta}_1+\left(m_2 l_{c2}^2+J_2\right)\ddot{\theta}_2+m_2 l_1 l_{c2}\dot{\theta}_1^2\sin\theta_2+m_2 g l_{c2}\cos\left(\theta_1+\theta_2\right) \tag{5}$$

The dynamic equation consisting of the following matrices:

$$\tau=M\left(\theta\right)\ddot{\theta}+C\left(\theta,\dot{\theta}\right)\dot{\theta}+G\left(\theta\right) \tag{6}$$

where $M\left(\theta\right)$ *isthe inertia matrix,* $C\left(\theta,\dot{\theta}\right)$ *is the centripetal vector,* $G\left(\theta\right)$ *is the gravity vector.*

$$M\left(\theta\right)=\begin{bmatrix}J_1+J_2+m_1 l_{c1}^2+m_2\left(l_1^2+l_{c2}^2+2l_1 l_{c2}c_2\right) & J_2+m_2\left(l_{c2}^2+l_1 l_{c2}c_2\right)\\ J_2+m_2\left(l_{c2}^2+l_1 l_{c2}c_2\right) & J_2+m_2 l_{c2}^2\end{bmatrix} \tag{7}$$

$$C\left(\theta,\dot{\theta}\right)=\begin{bmatrix}-l_1 l_{c2}m_2 s_2\dot{\theta}_2 & -l_1 l_{c2}m_2 s_2\left(\dot{\theta}_1+\dot{\theta}_2\right)\\ l_1 l_{c2}m_2 s_2\dot{\theta}_2 & 0\end{bmatrix} \tag{8}$$

$$G\left(\theta\right)=\begin{bmatrix}\left(m_1+m_2\right)g l_{c1}c_1+m_2 g l_{c2}c_{12}\\ m_2 g l_{c2}c_{12}\end{bmatrix} \tag{9}$$

$$\ddot{\theta} = \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix}, \quad \tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}, \quad \dot{\theta} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \tag{10}$$

where: $s_1 = \sin(\theta_1), c_1 = \cos(\theta_1), s_{12} = \sin(\theta_1 + \theta_2), c_{12} = sos(\theta_1 + \theta_2)$

2.2. Design of the Expert PD-Fuzzy Controller

The ANFIS network is inherently based on supervised learning; therefore, the system requires an accurate Training Data baseline. To address this, a hybrid PD-Fuzzy controller is designed to act as the "expert controller" (instructor). The controller operates using a zero-order Sugeno inference mechanism.

The two primary input signals include the tracking error (E) and the derivative of the error (DE). The universe of discourse for each variable is normalized to the range of -1, 1 and partitioned into 5 fuzzy sets (NB, NS, ZE, PS, PB), utilizing a combination of triangular (trimf) and trapezoidal (trapmf) membership functions. Based on control experience, a matrix comprising 25 rules (5x5) is synthesized. The constant output characteristic of the Sugeno system ensures a smoothly smoothed Control Surface, eliminating abrupt variations in torque signals.

2.3. Integration of ANFIS and Particle Swarm Optimization

Following the experimental run of the PD-Fuzzy controller to extract input/output state data pairs, a 5-layer ANFIS network (comprising fuzzification, rule matching, normalization, consequent, and summation layers) is initialized. To overcome the conventional backpropagation algorithm's tendency to get trapped in local extrema, the Particle Swarm Optimization (PSO) algorithm is applied to search globally for the ideal weight set. Within the search space, each particle encodes a complete parameter structure of the ANFIS network, including the parameters shaping the membership functions in the antecedent part and the coefficients in the consequent part. At the t-th iteration, the velocity Vi and position Xi of the i-th individual are updated based on personal experience (Pbest) and swarm experience (Gbest):

$$v_{ij}(t+1) = w.v_{ij}(t) + c1.R1.(Pbest_{ij}(t) - x_{ij}(t)) + c2.R2.(Gbest_{ij}(t) - x_{ij}(t)) \tag{11}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{12}$$

This evolutionary process is monitored using the Root Mean Square Error (RMSE) between the predicted ANFIS output and the reference data from the PD-Fuzzy controller as the objective function. The algorithm iterates continuously until the RMSE approaches its minimum. The final outcome yields an independent ANFIS controller that entirely replaces the initially cumbersome fuzzy rule computations while preserving the system's excellent dynamic characteristics.

III. RESULTS AND DISCUSSION

To verify the reliability and performance of the proposed control architecture, the entire dynamic model of the 2-DOF manipulator and the algorithms were implemented in the MATLAB/Simulink environment. The evaluation process is divided into two primary phases: investigating the network's convergence during offline training and analyzing the closed-loop trajectory tracking response.

3.1. Convergence of the PSO-ANFIS Training Process

In the initial phase, the ANFIS network receives the sample dataset from the expert PD-Fuzzy controller. Replacing the classical backpropagation algorithm with Particle Swarm Optimization (PSO) thoroughly resolved the neural network's most significant barrier: entrapment in local minima.

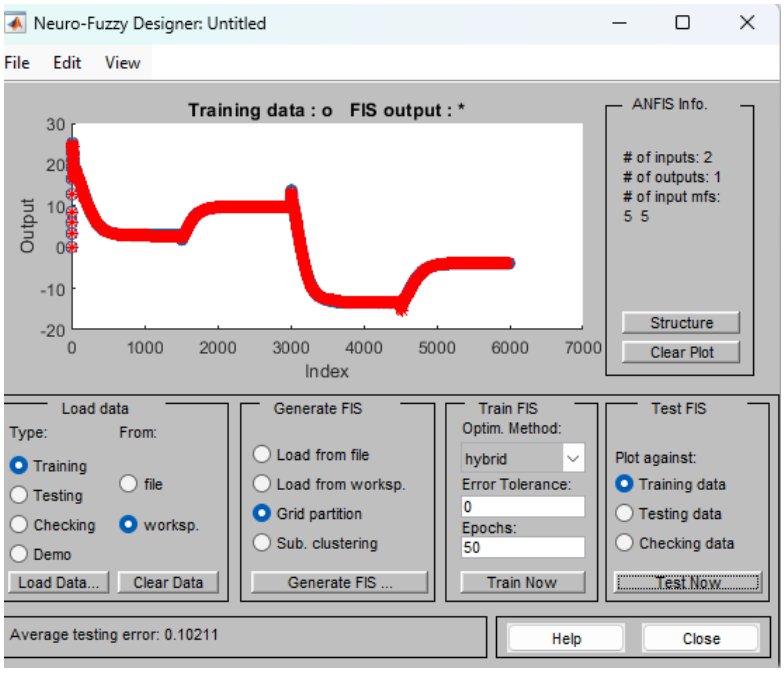


Fig. 2 Comparison between ANFIS output and reference data for Joint 1

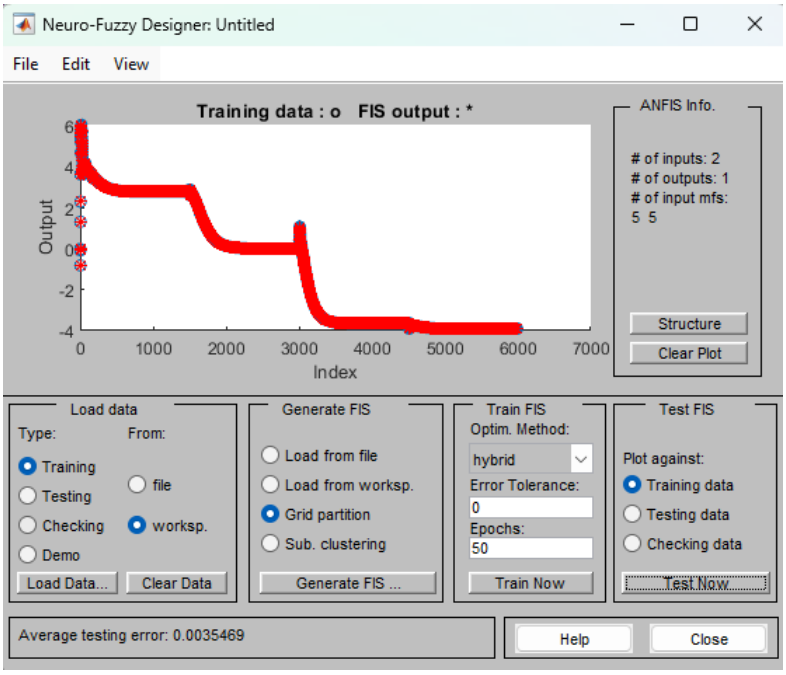


Fig. 3 Comparison between ANFIS output and reference data for Joint 2

The historical training data demonstrates PSO's superior global search speed. The objective function converges sharply, driving the Root Mean Square Error (RMSE) to a minimum after only 50 epochs (reaching 0.10211 for joint 1 and 0.003548 for joint 2). This near-zero error confirms that the network's weights have been successfully fine-tuned, perfectly approximating the expert's nonlinear control surface without encountering overfitting.

3.2. Analysis of Dynamic Characteristics and Trajectory Tracking Capability

To introduce rigorous dynamic stimuli, the Setpoint signal was designed as a multi-level step pulse. This signal type forces the joints to undergo abrupt changes in velocity and acceleration, thereby clearly exposing the nonlinear disturbance components generated internally within the mechanism.

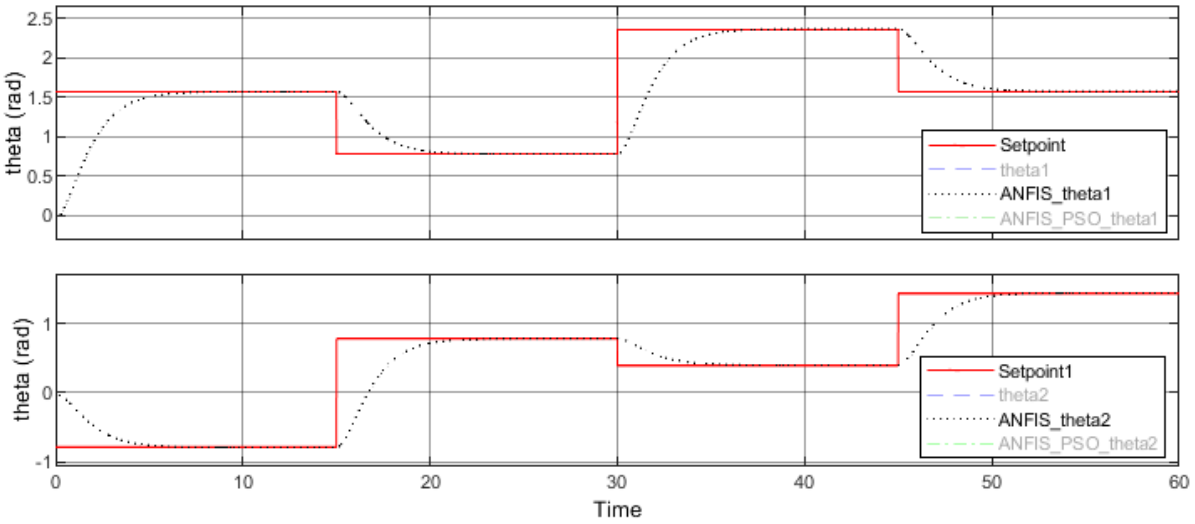


Fig. 4 Angular trajectory tracking response of the 2-DOF robot using conventional ANFIS

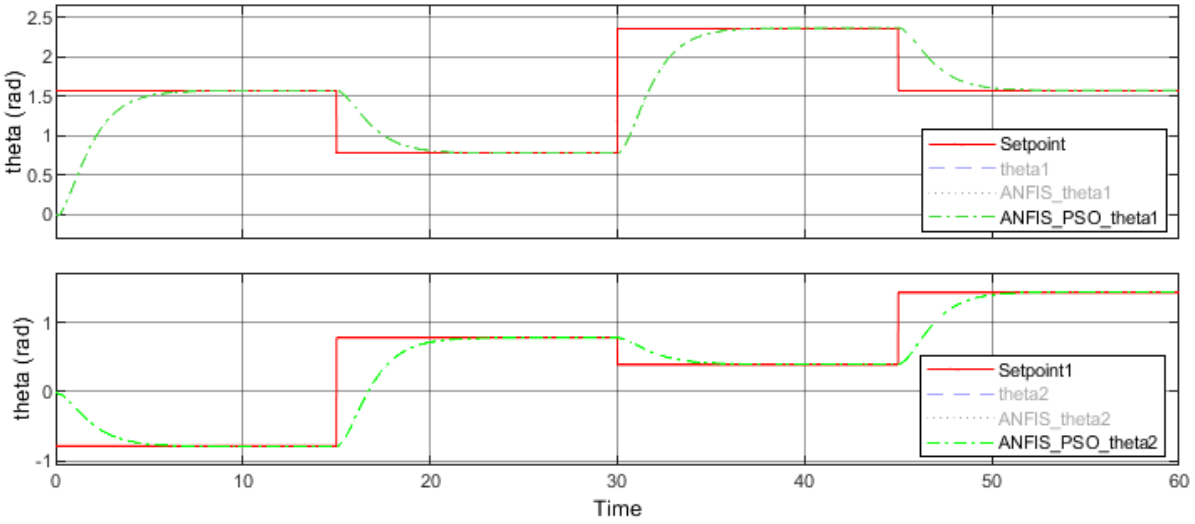


Fig. 5 Angular trajectory tracking response of the 2-DOF robot using PSO-optimized ANFIS

An in-depth analysis of the operational characteristics reveals that the most prominent feature is the actual response curve of both joints adhering perfectly to the required trajectory. The system handles the transient phase flawlessly: completely eliminating overshoot (Overshoot = 0%) and leaving no steady-state error upon entering the steady state. Dynamically, a 2-DOF manipulator is a multivariable system containing highly pronounced cross-coupling components. As joint 1 accelerates, it immediately generates moments of inertia and centripetal forces that act directly on the axis of joint 2, and vice versa. The fact that the rotation angle trajectory remains entirely unperturbed during phase transitions proves that the ANFIS-PSO architecture successfully learned the system's inverse dynamics. The controller proactively generated precise compensatory signals at the software level to cancel out these physical interactive forces. Looking further into applicability, this solution offers an absolute computational cost advantage. Instead of continuously executing fuzzification, inferencing across 25 rules, and performing cumbersome defuzzification as in the initial PD-Fuzzy, the current ANFIS-PSO network operates solely as a feed-forward structure utilizing static matrix multiplications. This streamlined architecture minimizes the control signal's phase delay, fully meeting the stringent criteria for embedding into real-time microprocessors in industrial robotics.

IV. CONCLUSION

This study investigated a trajectory tracking control framework for a two-degree-of-freedom (2-DOF) robot manipulator by integrating an expert-designed Fuzzy-PD controller, an adaptive neuro-fuzzy inference system (ANFIS), and particle swarm optimization (PSO). The Fuzzy-PD controller generated paired observations of the selected state variables and corresponding control actions, providing supervised training data for a five-layer ANFIS. PSO was employed to search the selected parameter space and improve the approximation of the expert control policy. This workflow establishes a systematic connection between knowledge-based controller design and data-driven neuro-fuzzy learning.The simulation results support the feasibility of using the trained ANFIS for closed-loop trajectory tracking under the investigated conditions. The reported responses indicate accurate reference following and smooth transient behavior, while the training results demonstrate the network’s ability to approximate the expert-generated control mapping. These findings should be interpreted separately: a small training root-mean-square error measures agreement with the reference control actions, whereas tracking performance depends on the interaction between the learned controller and the manipulator dynamics. The results therefore support the proposed workflow within the simulated operating region, without establishing guaranteed global optimality, general disturbance rejection, or closed-loop stability beyond the conditions examined.The principal value of the approach lies in transferring an expert-designed nonlinear feedback policy into a trainable neuro-fuzzy representation and using PSO to support parameter selection. The resulting controller retains fuzzy membership evaluation, rule activation, normalization, and consequent aggregation. Accordingly, any reduction in computational cost or control latency must be established through implementation measurements. Likewise, successful tracking of the coupled manipulator does not, by itself, demonstrate explicit identification or cancellation of individual inertial and Coriolis terms.

Future work will first examine generalization using trajectories, initial configurations, and operating conditions excluded from the training dataset. Comparisons with the expert Fuzzy-PD controller and an ANFIS trained using conventional hybrid learning will help distinguish the effects of policy approximation from those of PSO-based optimization. Repeated optimization runs should also be conducted to assess the variability of the resulting solutions. Further evaluation will consider payload changes, friction uncertainty, measurement noise, external disturbances, and actuator constraints, using tracking-error indices, control effort, and transient-response measures.Experimental validation on a physical 2-DOF manipulator will subsequently assess the effects of sampling, sensing, actuator dynamics, and computation time. State or disturbance observers may be incorporated to improve feedback quality and compensate for unmeasured effects. In parallel, stability analysis that explicitly accounts for approximation errors and bounded uncertainties will be needed to strengthen the theoretical basis of the controller. These developments will clarify the conditions under which the proposed framework can provide reliable tracking and support its progression toward real-time robotic implementation.

ACKNOWLEDGMENTS

This research is funded by Electric Power University under research project code ĐTNH.30/2026.

REFRENCES

[1]. J. J. Craig, Introduction to Robotics: Mechanics and Control, 3rd ed. Upper Saddle River, NJ, USA: Pearson Prentice Hall, 2005. https://search.worldcat.org/title/56352264

[2]. M. W. Spong, S. Hutchinson, and M. Vidyasagar, Robot Modeling and Control. Hoboken, NJ, USA: John Wiley & Sons, 2006. https://www.iri.upc.edu/people/thomas/Collection/details/1544.html

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

[4]. T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-15, no. 1, pp. 116–132, 1985. https://doi.org/10.1109/TSMC.1985.6313399

[5]. M. A. Ahmad, M. Z. M. Tumari, and A. N. K. Nasir, "Composite fuzzy logic control approach to a flexible joint manipulator," International Journal of Advanced Robotic Systems, vol. 10, Art. no. 58, 2013. https://doi.org/10.5772/52562

[6]. C. Urrea, J. Kern, and J. Alvarado, "Design and evaluation of a new fuzzy control algorithm applied to a manipulator robot," Applied Sciences, vol. 10, no. 21, Art. no. 7482, 2020. https://doi.org/10.3390/app10217482

[7]. J.-S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," IEEE Transactions on Systems, Man, and Cybernetics, vol. 23, no. 3, pp. 665–685, 1993. https://doi.org/10.1109/21.256541

[8]. J. Kern, D. Marrero, and C. Urrea, "Fuzzy control strategies development for a 3-DoF robotic manipulator in trajectory tracking," Processes, vol. 11, no. 12, Art. no. 3267, 2023. https://doi.org/10.3390/pr11123267

[9]. J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proceedings of ICNN'95 – International Conference on Neural Networks, vol. 4, 1995, pp. 1942–1948. https://doi.org/10.1109/ICNN.1995.488968

[10]. M. A. Shoorehdeli, M. Teshnehlab, and A. Khaki-Sedigh, "Training ANFIS as an identifier with intelligent hybrid stable learning algorithm based on particle swarm optimization and extended Kalman filter," Fuzzy Sets and Systems, vol. 160, no. 7, pp. 922–948, 2009. https://doi.org/10.1016/j.fss.2008.09.011

[11]. V. Seydi, M. A. Shoorehdeli, and M. Teshnehlab, "Training ANFIS structure with modified PSO algorithm," in 2007 Mediterranean Conference on Control & Automation, 2007, pp. 1–6. https://doi.org/10.1109/MED.2007.4433927

[12]. P. Liu, W. Leng, and W. Fang, "Training ANFIS model with an improved quantum-behaved particle swarm optimization algorithm," Mathematical Problems in Engineering, vol. 2013, Art. no. 595639, 2013. https://doi.org/10.1155/2013/595639

[13]. B. Selma, S. Chouraqui, and H. Abouaïssa, "Fuzzy swarm trajectory tracking control of unmanned aerial vehicle," Journal of Computational Design and Engineering, vol. 7, no. 4, pp. 435–447, 2020. https://doi.org/10.1093/jcde/qwaa036

[14]. M. J. Mahmoodabadi and A. Ziaei, "Inverse dynamics based optimal fuzzy controller for a robot manipulator via particle swarm optimization," Journal of Robotics, vol. 2019, Art. no. 5052185, 2019. https://doi.org/10.1155/2019/5052185

[15]. Y. Li, S. Qiang, X. Zhuang, and O. Kaynak, "Robust and adaptive backstepping control for nonlinear systems using RBF neural networks," IEEE Transactions on Neural Networks, vol. 15, no. 3, pp. 693–701, 2004. https://doi.org/10.1109/TNN.2004.826215