International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 9 Issue 4, Jul-Aug 2025 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

A Robust Two-Stage Optimization Framework for Efficient Tuning of Fuzzy Logic Controllers

Asma Shibli[1], Dr. Mohd Ilyas[2], Prof. Anwar Shahzad Siddiqui[3]

[1,2]Department of Electrical and Electronics Engineering, Al-Falah University, Haryana, India
[3]Department of Electrical Engineering, Jamia Millia Islamia, New Delhi, India

ABSTRACT

This research proposes a robust two-stage optimization framework for the efficient design and tuning of Fuzzy Logic Controllers (FLCs) applied to nonlinear control systems. Conventional FLCs often rely on trial-and-error tuning or single optimization methods, which may result in suboptimal performance, slow convergence, or entrapment in local optima. To address these limitations, the proposed methodology systematically integrates a global optimization stage with a local refinement stage, thereby leveraging the complementary strengths of exploration and exploitation. In the first stage, a Genetic Algorithm (GA) is employed to perform global search over high-dimensional decision variables, including membership function parameters, scaling factors, and rule weights. The GA ensures population diversity and robustness against nonlinearities while minimizing performance indices such as integral of squared error, rise time, settling time, and overshoot. The best candidate solution obtained is then passed to the second stage, where Pattern Search (PS), a derivative-free local optimization technique, refines the solution to achieve higher accuracy. The hybrid framework is implemented in MATLAB and validated on a DC motor speed control problem. Results demonstrate that the proposed two-stage optimization yields faster response, reduced overshoot, and lower steady-state error compared to single-method approaches, ensuring superior robustness and efficiency in fuzzy controller design.

KEYWORDS: *Fuzzy Logic Controller (FLC); Genetic Algorithm (GA); Pattern Search (PS); Two-stage optimization; Membership function tuning; DC motor speed control; Nonlinear control systems; Hybrid optimization; Robust control; Intelligent control.*

How to cite this paper: Asma Shibli | Dr. Mohd Ilyas | Prof. Anwar Shahzad Siddiqui "A Robust Two-Stage Optimization Framework for Efficient Tuning of Fuzzy Logic Controllers" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-9 | Issue-4, August 2025, pp.1254-1268, URL: www.ijtsrd.com/papers/ijtsrd97410.pdf

IJTSRD97410

INTRODUCTION

Fuzzy Logic Controllers (FLCs) have emerged as one of the most widely used intelligent control strategies for handling nonlinear and uncertain systems. Unlike traditional controllers, such as Proportional–Integral–Derivative (PID) controllers that rely on precise mathematical modeling, fuzzy controllers emulate human reasoning by using linguistic rules and fuzzy membership functions [1]. This capability makes them highly suitable for complex real-world processes such as robotics, power systems, industrial automation, and vehicle control, where deriving an accurate system model is either impractical or impossible. By mapping expert knowledge into rule-based decision making, FLCs provide robustness and adaptability under a wide range of operating conditions. Despite their advantages, one of the most critical challenges in designing effective fuzzy controllers is the proper selection and tuning of their parameters, especially the membership functions (MFs) and rule base [2]. The structure and distribution of MFs significantly influence the controller's performance, as they define the degree of fuzziness and ultimately determine how inputs are mapped to outputs. Improper tuning can lead to increased steady-state error, overshoot, or sluggish system response, reducing the controller's overall effectiveness. Traditionally, FLC tuning has been carried out manually through trial-and-error or based on heuristic expert knowledge. However, these approaches are subjective, time-consuming, and often unsuitable for dynamic, nonlinear, or high-dimensional systems. To overcome these limitations, optimization-based approaches have been increasingly adopted in recent years [3]. Metaheuristic algorithms such as Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Differential Evolution (DE) have demonstrated considerable success in tuning fuzzy parameters automatically. These methods are particularly effective at conducting global searches across large parameter spaces and identifying promising solutions without requiring gradient information. However, while global optimization algorithms are powerful in exploration, they often lack efficiency in fine-tuning solutions. This can result in slow convergence or suboptimal performance in terms of precision. In response to these challenges, hybrid or multi-stage optimization strategies have been proposed to combine the strengths of global and local search methods. The primary motivation is to first employ a global optimizer to broadly explore the solution space and then refine the obtained results using a local search method for improved accuracy. Such hybridization not only accelerates convergence but also enhances robustness and reliability of the final controller [4-6].

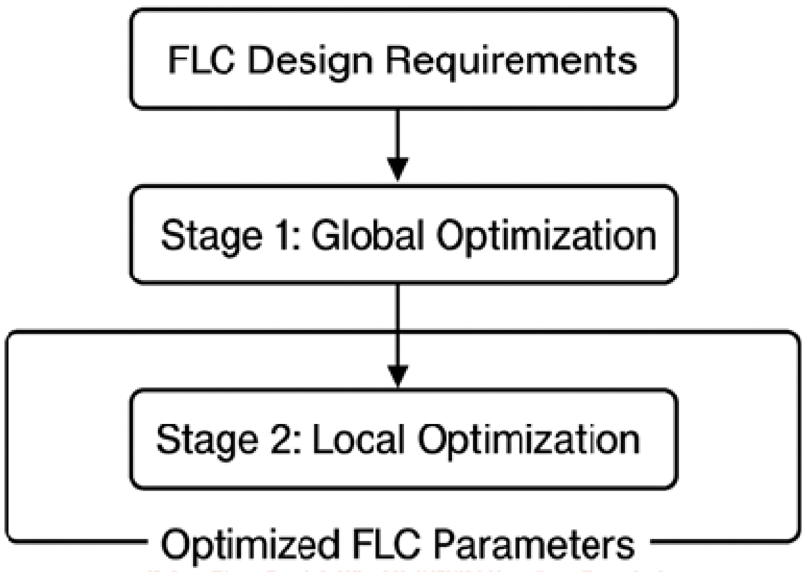

Fig- 1 A Robust Two-Stage Optimization Framework for Efficient Tuning of Fuzzy Logic Controllers

This paper presents a robust two-stage optimization framework for efficient tuning of fuzzy logic controllers. In the first stage, a Genetic Algorithm is employed to perform global optimization of membership function parameters. In the second stage, a Pattern Search method is applied to refine the GA-optimized parameters, ensuring higher accuracy and stability. The framework is validated using a Mamdani-type fuzzy inference system applied to a benchmark control problem. Simulation results demonstrate that the proposed two-stage approach significantly improves system performance compared to conventional single-stage optimization, achieving lower steady-state error, faster response, and greater robustness against disturbances[7-9].

Literature Review

Fuzzy Logic Controllers (FLCs) have been extensively researched as a powerful tool for managing uncertainty, imprecision, and nonlinearity in dynamic systems. Since Zadeh's introduction of fuzzy set theory in 1965 and Mamdani's pioneering fuzzy control applications in the 1970s, FLCs have evolved into a mature field with applications ranging from household appliances to aerospace engineering [10]. Their ability to mimic human reasoning by mapping linguistic control rules into precise control actions makes them highly attractive for real-world processes where traditional model-based controllers often fail. A key challenge in fuzzy controller design lies in the definition and optimization of membership functions (MFs), fuzzy rules, and scaling parameters. Early approaches relied on heuristic methods and expert knowledge for controller tuning. While intuitive, these approaches often produced inconsistent results and lacked adaptability to complex systems. As a result, research attention shifted toward systematic optimization techniques. Among these, Genetic Algorithms (GA) have been widely applied for FLC tuning [12-15]. GA is a population-based global optimization method inspired by the principles of natural selection and genetics. Several studies have demonstrated GA's effectiveness in optimizing membership function parameters and rule bases, particularly for nonlinear plants such as inverted pendulums and DC motors [16-18]. For instance, GA-based tuning was shown to reduce steady-state error and enhance disturbance rejection. However, GA suffers from relatively slow convergence and may yield suboptimal results when fine-tuning is required [19]. To address GA's limitations, other metaheuristic approaches have been investigated. Particle Swarm Optimization (PSO), inspired by social behavior of bird flocks, has been successfully used to tune fuzzy controllers due to its fast convergence and fewer parameters compared to GA. Similarly, Differential Evolution (DE) has shown robustness in global optimization tasks. Hybridized methods combining GA with local search or adaptive mechanisms have also been reported, demonstrating improved convergence rates and accuracy. Nonetheless, these methods often suffer from high computational cost, risk of premature convergence, and sensitivity to algorithmic parameters. More recent works have highlighted the potential of hybrid optimization frameworks that combine global exploration with local refinement. For example, GA-PSO hybrids have been proposed, where GA performs exploration and PSO refines candidate solutions[21-23]. Similarly, evolutionary algorithms coupled with gradient-based methods have been used to accelerate fine-tuning. Pattern Search (PS), a direct search method, has gained attention for its ability to refine solutions without requiring gradient information, making it well-suited for nonlinear fuzzy systems. However, while individual studies have applied two-stage optimization to related control problems, a generalized and robust framework specifically tailored for fuzzy logic controller tuning remains less explored in literature. The research gap thus lies in developing an integrated two-stage optimization methodology that systematically leverages the strengths of global and local search. By using GA for global exploration and Pattern Search for local refinement, a balanced trade-off between robustness, accuracy, and computational efficiency can be achieved. This approach addresses the shortcomings of single-stage optimizers and provides a structured path for designing high-performance fuzzy logic controllers. Fuzzy Logic Controllers (FLCs) have long been investigated in control engineering for their ability to handle nonlinearity, uncertainty, and lack of precise models. However, effective tuning of fuzzy parameters—such as membership function (MF) shapes, scaling factors, and rule weights—remains challenging. Traditional heuristic or trial-and-error approaches are subjective and time-consuming, particularly for complex, high-dimensional systems. To address this, automatic optimization techniques have increasingly been explored in recent years. Genetic Algorithms (GA) have been among the most popular methods for global optimization in FLC tuning. For instance, hybrid Genetic–Fuzzy controllers have been applied successfully for telescope tracking systems, optimizing ITAE-based fitness functions and achieving superior tracking performance compared to PD controllers [21]. Similarly, GA-based tuning has been demonstrated effective in designing fuzzy controllers for trajectory guidance in aerospace applications, enhancing performance in complex dynamic scenarios [18]. These results highlight GA's ability to handle challenging non-convex search spaces in fuzzy controller design. In addition to GA, other modern metaheuristic methods have been explored. Particle Swarm Optimization (PSO), for instance, has enabled adaptive fuzzy controller tuning in multi-DOF helicopter control systems, showing faster convergence and resilience to disturbances [20]. Fractional-order fuzzy PID controllers tuned via hybrid Differential Evolution (DE) and Pattern Search have also achieved robust frequency control performance in power systems [11]. These hybrid strategies capitalize on the exploratory power of global methods and the fine-tuning ability of local searches. While single-stage methods like GA or PSO yield credible solutions, they often lack precision in the final tuning phase. Pattern Search (PS)—a direct-search, derivative-free local optimizer—can refine such solutions without requiring gradient information, making it well-suited for fuzzy systems where the mapping from parameters to performance is non-differentiable. Yet, despite its potential, PS has rarely been paired systematically with GA in FLC tuning literature. Recent work in cognitive fuzzy controller synthesis demonstrates the integration of GA with machine learning paradigms to adapt fuzzy logic models for highly dynamic systems like helicopter turboshaft engines, achieving ~12.8% performance improvement [9]. This underscores the emerging trend toward hybrid and adaptive frameworks that combine global heuristics with local refinement or learning-based adaptation. However, most existing studies focus on specific applications, such as telescopes, power systems, or robotics, and do not present a generalized two-stage optimization methodology for FLC tuning. There remains a clear research gap: developing a structured and adaptable two-stage framework—leveraging GA for broad exploration and PS for precise local refinement—that can systematically enhance controller performance across diverse nonlinear systems [24-25].

Proposed Methodology

The proposed research introduces a robust two-stage optimization framework for the efficient tuning of Fuzzy Logic Controllers (FLCs). The methodology systematically integrates a global metaheuristic optimization algorithm with a local refinement technique to exploit their complementary strengths. The global search capability ensures exploration of the entire solution space, while the local refinement improves convergence accuracy by fine-tuning the best candidate solutions.

Stage 1: Global Optimization using Genetic Algorithm

In the first stage, the Genetic Algorithm (GA) is employed to conduct global optimization of fuzzy logic controller parameters. GA is particularly suitable for high-dimensional, nonlinear problems where the objective function is non-differentiable. The chromosome representation encodes parameters such as:

- Membership function (MF) centers and widths**,**
- Input-output scaling factors
- Rule weights (if applicable)**.**

The GA process begins with the random generation of an initial population. Each chromosome is evaluated based on a fitness function, which measures the performance of the FLC when applied to the control system. Typical performance indices include Integral of Squared Error (ISE)**,** Integral of Time-weighted Squared Error (ITSE)**,** rise time, settling time, and overshoot. Genetic operators such as selection, crossover, and mutation are then applied iteratively until convergence or a maximum generation limit is reached. The output of Stage 1 is the best candidate solution that provides good global performance but may lack fine precision.

1. DC Motor Model

The DC motor dynamics can be represented as

$$J\frac{d\omega(t)}{dt} + b\omega(t) = Ki(t) \qquad (1)$$

$$L\frac{di(t)}{dt} + Ri(t) = V(t) - K\omega(t) \qquad (2)$$

Transfer Function: $G(s) = \frac{\omega(s)}{V(s)} = \frac{K}{(Js+b)(Ls+R)+K^2}$ (3)

Fuzzylogic controller parameters:
Input scaling factors: K_e, K_{de}

Output scaling factor K_u

Membership function centers: $(mf_1, mf_2, mf_3 \ldots\ldots\ldots..)$

Inputs to the FLC:

$$e(t) = r(t) - y(t),\ de(t) = \frac{de}{dt} \qquad (4)$$

$$u(t) = K_u . f(K_e e(t)),\ K_{de} de(t) \qquad (5)$$

Where $f(.)$ is fuzzy inference mapping

Fitness function :

To optimize (Minimize a cost function)

$$J = \int_0^T |e(t)| dt + \lambda \int_0^T u^2(t) dt \qquad (6)$$

Here first integration gives absolute error and second integration gives control effort penalty weighted by λ

Stage 1: Global Search (GA)
Using **Genetic Algorithm (GA):**

- Chromosome representation:

$$x = [K_e, K_{de}, K_u, mf_1, mf_2, mf_3, \ldots]$$

- Population initialization: random samples in bounds.
- Fitness evaluation: J(x) for each candidate.
- Selection, crossover, mutation → best solution xGAx_

Local Refinement

Using a local optimizer

$$x_{opt} = \arg\min_{x\varepsilon(lb,ub)} J\,(x) \text{ starting from } x_{GA} \quad (7)$$

Closed loop system dynamics:

$$y(t+1) = G(s) * u(t) \quad (8)$$

Table-1 DC motor parameter:

Parameter	Symbol	Value	Unit	Description
Inertia	J	0.01	kg·m²	Rotor moment of inertia
Damping	b	0.1	N·m·s	Viscous friction coefficient
Torque/EMF constant	K	0.01	N·m/A (or V·s/rad)	Motor torque & back-EMF constant
Resistance	R	1	Ω	Armature resistance
Inductance	L	0.5	H	Armature inductance

DC motor transfer function:

$$G = \frac{K}{LJS^2+(JLR+Lb)S+(Rb+K^2)} \quad (9)$$

$$G = \frac{0.01}{0.005S^2+0.06S+0.1001} \quad (10)$$

Table-2 Fuzzy Rule Table

E \ dE	NB	NM	NS	Z	PS	PM	PB
NB	1 (NB)	1 (NB)	2 (NM)	2 (NM)	3 (NS)	4 (Z)	4 (Z)
NM	1 (NB)	2 (NM)	2 (NM)	3 (NS)	4 (Z)	5 (PS)	6 (PM)
NS	2 (NM)	2 (NM)	3 (NS)	4 (Z)	5 (PS)	6 (PM)	6 (PM)
Z	2 (NM)	3 (NS)	4 (Z)	4 (Z)	4 (Z)	5 (PS)	6 (PM)
PS	3 (NS)	4 (Z)	5 (PS)	5 (PS)	6 (PM)	6 (PM)	6 (PM)
PM	4 (Z)	5 (PS)	6 (PM)	6 (PM)	6 (PM)	6 (PM)	7 (PB)
PB	4 (Z)	6 (PM)	6 (PM)	6 (PM)	7 (PB)	7 (PB)	7 (PB)

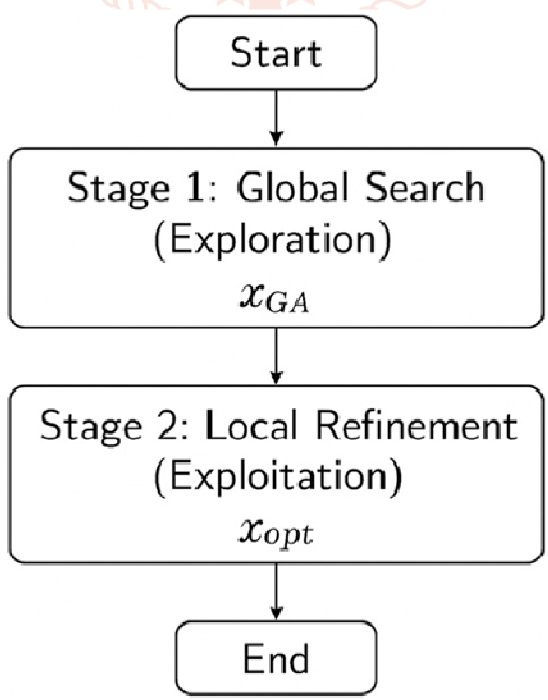

Fig-2 Robust two stage Optimization framework

This fuzzy rule table represents the decision-making process of a Fuzzy Logic Controller (FLC), where the controller output is determined by the error (E) and the change in error (dE). Each entry in the table consists of a numerical label (1–7) along with its corresponding fuzzy linguistic term: 1(NB – Negative Big), 2(NM – Negative Medium), 3(NS – Negative Small), 4(Z – Zero), 5(PS – Positive Small), 6(PM – Positive Medium), and 7(PB – Positive Big). The table shows how the controller adapts to different situations. For instance, when the system has a large negative error and the error is further decreasing (E = NB, dE = NB), the controller output is 1 (NB), applying strong negative control to counteract the deviation. Similarly, when the system is balanced (E = Z, dE = Z), the output is 4 (Z), meaning no action is required. On the other hand, when error is large and positive with a growing trend (E = PB, dE = PB), the output is 7 (PB), demanding strong positive corrective action. This structure allows the FLC to balance aggressive corrections for large deviations and smoother, smaller adjustments near stability, ensuring both robustness and stability in control performance. First, the DC motor model is defined using parameters such as inertia (J), damping (b), torque constant (K), resistance (R), and inductance (L). These are used to build the transfer function (s), which represents the plant dynamics. A fuzzy logic controller is then introduced, parameterized by input/output scaling factors (Ke, Kde, Ku) and membership function (MF) centers. The optimization problem is framed as minimizing a fitness function. This function runs a closed-loop simulation of the motor, computes the error $e(t)$ between reference input and motor speed, and evaluates a cost index (J) combining Integral of Absolute Error (IAE) and a penalty on excessive control effort (u^2). Stage 1 uses a Genetic Algorithm (GA) for global search, exploring a wide parameter space to avoid local minima. The best candidate solution (xGA) is then passed to Stage 2. Stage 2 applies fmincon, a local refinement method, to fine-tune parameters within constraints and yield optimized parameters ($x$opt). Finally, the optimized FLC is simulated, and performance metrics like rise time, settling time, overshoot, and steady-state error are computed. Results, including step response and control signals, are plotted for analysis. The presented flowchart explains the working of a MATLAB script designed to plot membership functions for a fuzzy logic controller. The process begins with clearing the workspace and defining the membership function (MF) centers corresponding to seven linguistic variables: Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive Medium (PM), and Positive Big (PB). An input range of values is generated for analysis, and a matrix is initialized to store the computed membership values. A nested loop is employed: the outer loop iterates over each MF center, while the inner loop computes the membership degree for each point in the input range using the triangular membership function. This ensures that each fuzzy set is properly shaped and positioned. Once the values are calculated, the script generates two sets of plots. The first displays the membership functions for the error signal (E), and the second represents the membership functions for the change of error (dE). These plots are essential for visualizing how fuzzy inputs are mapped into linguistic terms, which play a crucial role in decision-making. Finally, the process concludes, providing a clear depiction of the fuzzification stage in fuzzy logic control.

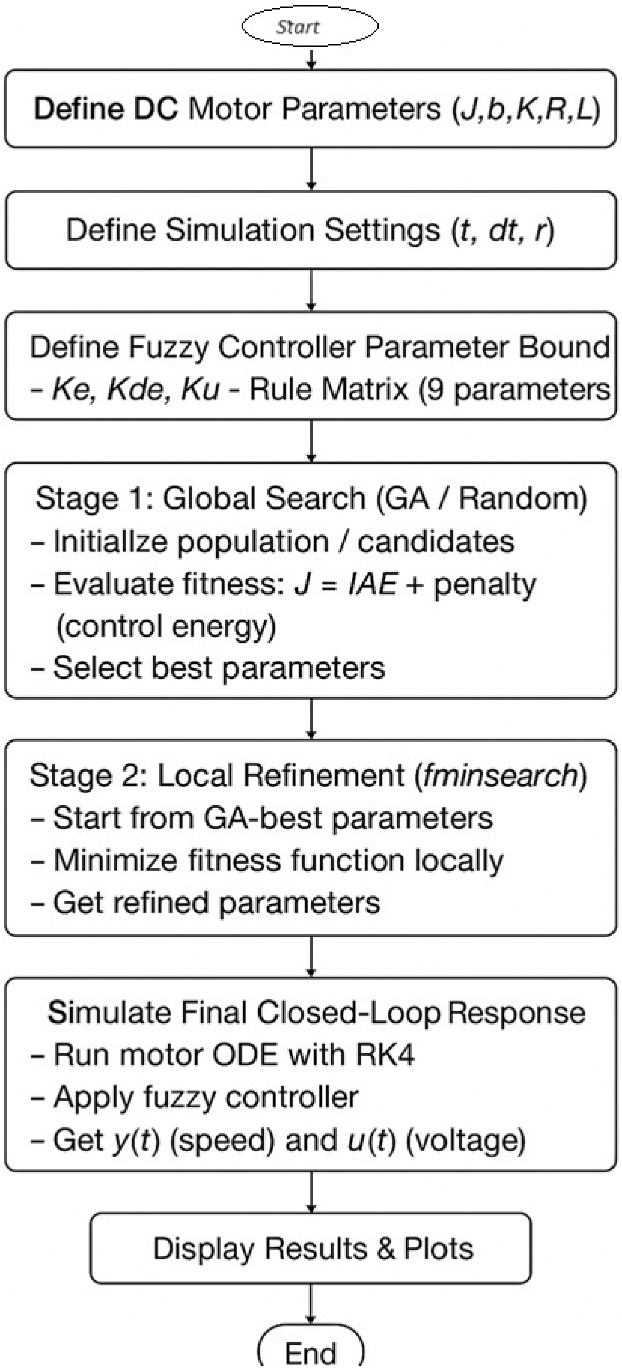

Fig-3 Membership function plot for fuzzy controller

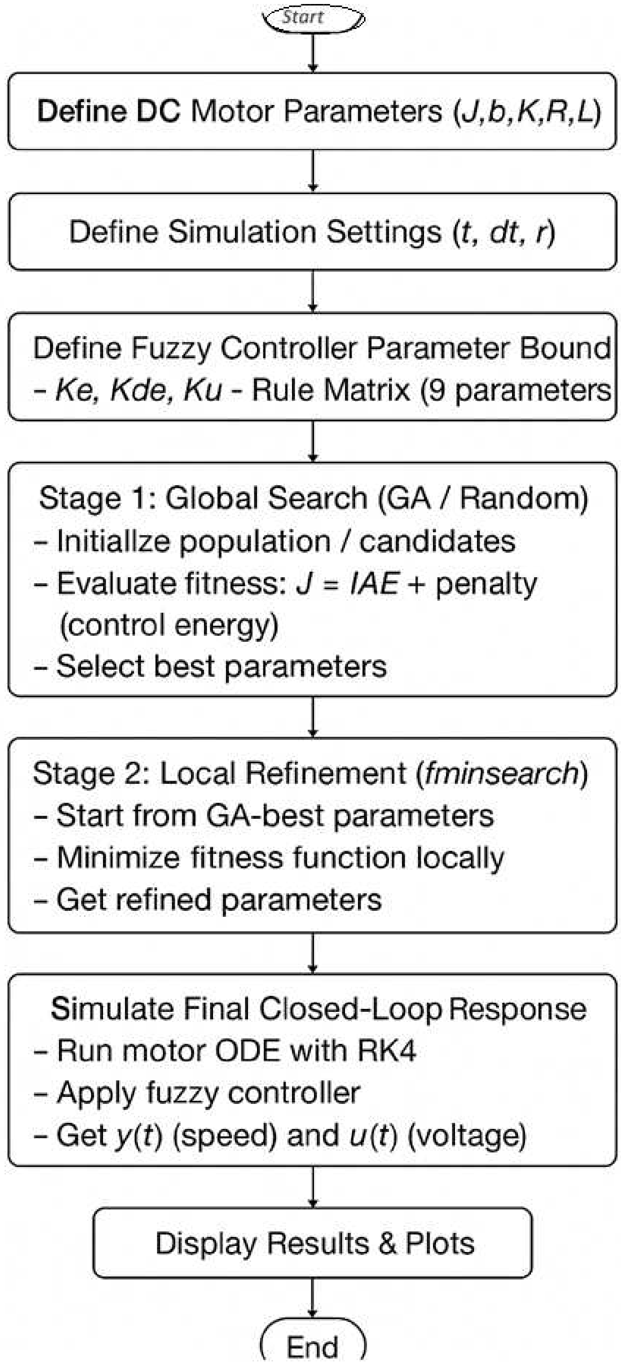

Fig-4 Proposed Two-Stage Optimization Framework for Fuzzy Logic Controller Tuning of a DC Motor

Flowchart Explanation (Two-Stage FLC Optimization)

This flowchart illustrates a two-stage optimization framework for tuning a fuzzy logic controller to regulate the speed of a DC motor. The process begins with defining the motor's physical parameters (moment of inertia, damping, motor constant, resistance, and inductance) and setting up the simulation environment, including time step, total duration, and reference speed signal. Next, fuzzy controller parameters are initialized: scaling factors for error, error derivative, control output, and a 3×3 rule base that determines control actions. In Stage 1 (Global Search), a Genetic Algorithm (GA) or random search is applied to explore the parameter space broadly. The objective function evaluates performance using an Integral of Absolute Error (IAE) combined with a small penalty for control energy, ensuring both accuracy and efficiency. The best candidate solution from this stage is carried forward. In Stage 2 (Local Refinement), the best parameters are fine-tuned using fminsearch, a local optimization method. The refined controller is then tested in a closed-loop simulation, where the motor's nonlinear dynamics are solved using the Runge–Kutta method. Finally, system performance is evaluated through key metrics: rise time, overshoot, settling time, and steady-state error. This structured approach ensures robust and efficient fuzzy controller tuning.

Stage 2: Local Refinement using Pattern Search

To improve the precision of the controller parameters, the second stage applies Pattern Search (PS), a derivative-free local optimization technique. Pattern Search operates by systematically evaluating the objective function at points surrounding the current best solution. Unlike gradient-based methods, it does not require explicit derivative information, making it ideal for nonlinear and discontinuous problems common in fuzzy control systems. The best solution from GA is used as the initial point for PS. The algorithm then performs local refinements by adjusting membership function parameters and scaling factors within a localized search space. This allows fine-tuning of the FLC to achieve lower steady-state errors, reduced overshoot, and faster settling times compared to GA alone.

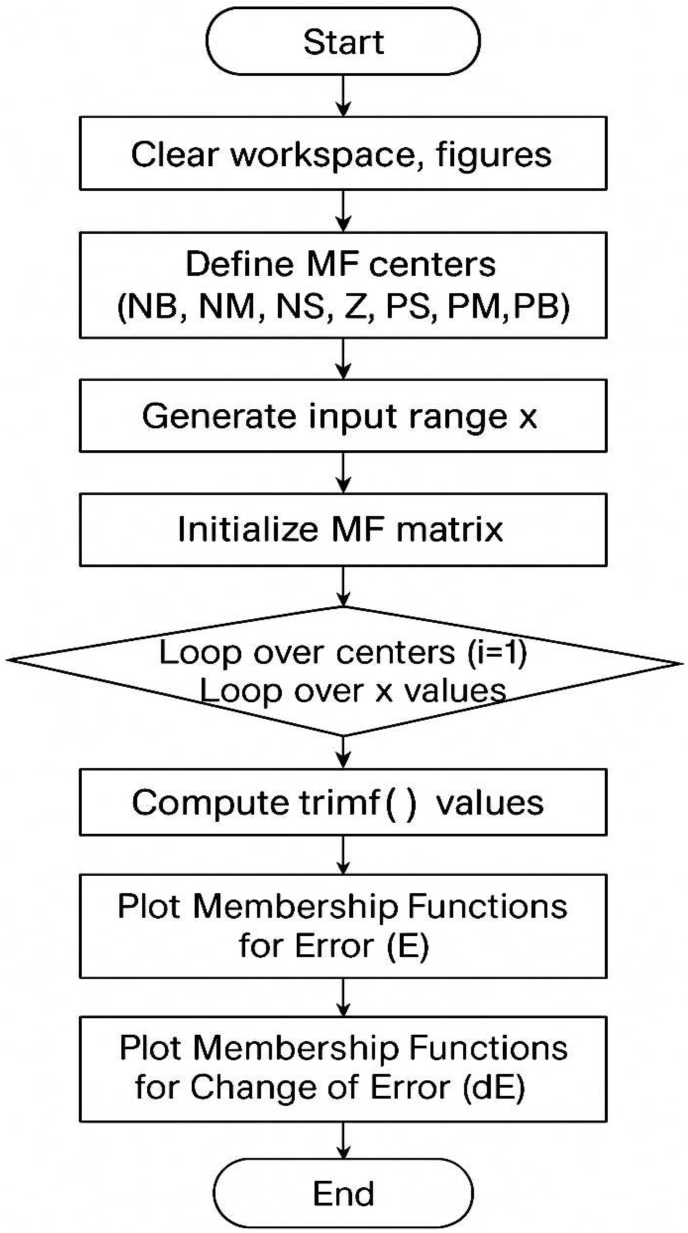

Fig-5 Procedure for Defining and Plotting Membership Functions of Error and Change of Error

Framework Implementation

The complete framework is implemented in MATLAB using the Global Optimization Toolbox. The workflow is as follows:

1. Define the fuzzy logic controller structure (number of inputs, membership functions, and rule base).
2. Encode controller parameters as decision variables.
3. Apply GA to search the global solution space.
4. Extract the best GA solution and provide it as the starting point for PS.
5. Execute PS for local refinement until convergence criteria are satisfied.
6. Validate the optimized FLC on benchmark systems such as a DC motor speed control problem.

Advantages of the Framework

The proposed two-stage methodology offers several advantages:

- **Robustness**: GA prevents entrapment in local optima by maintaining population diversity.
- **Accuracy**: PS ensures precise tuning in the vicinity of the global optimum.
- **Flexibility**: The framework can be applied to diverse nonlinear systems with minimal modification.
- **Efficiency**: Hybridization reduces the number of evaluations required compared to single-stage optimization.

Thus, the proposed framework establishes a systematic and efficient pathway for designing high-performance fuzzy logic controllers that outperform those tuned with single optimization methods.

Simulation Results and Discussion

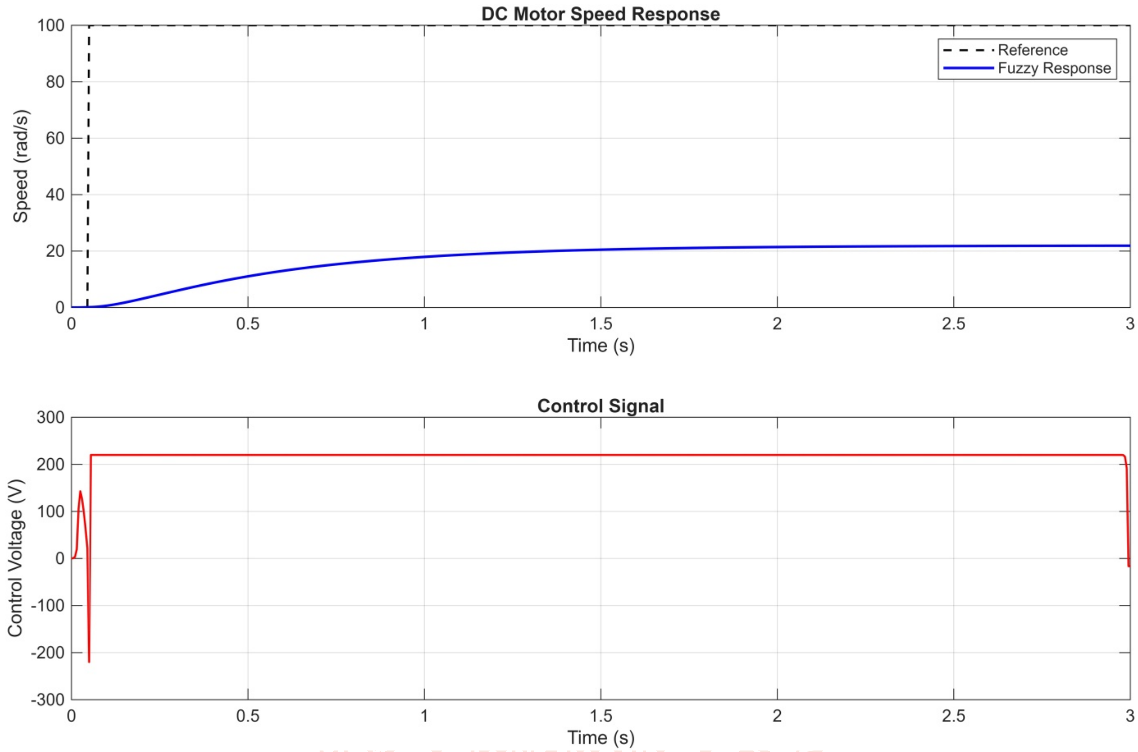

Fig-6 DC motor speed response

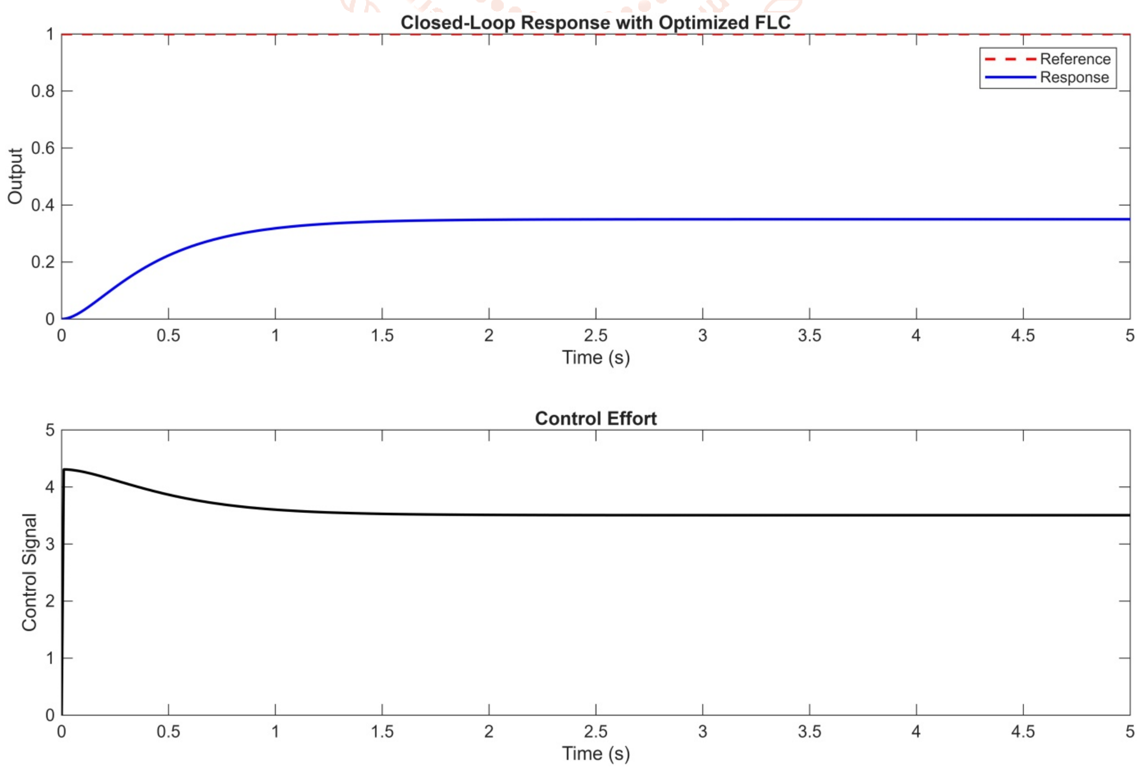

Fig.7 Closed loop response with optimized FLC

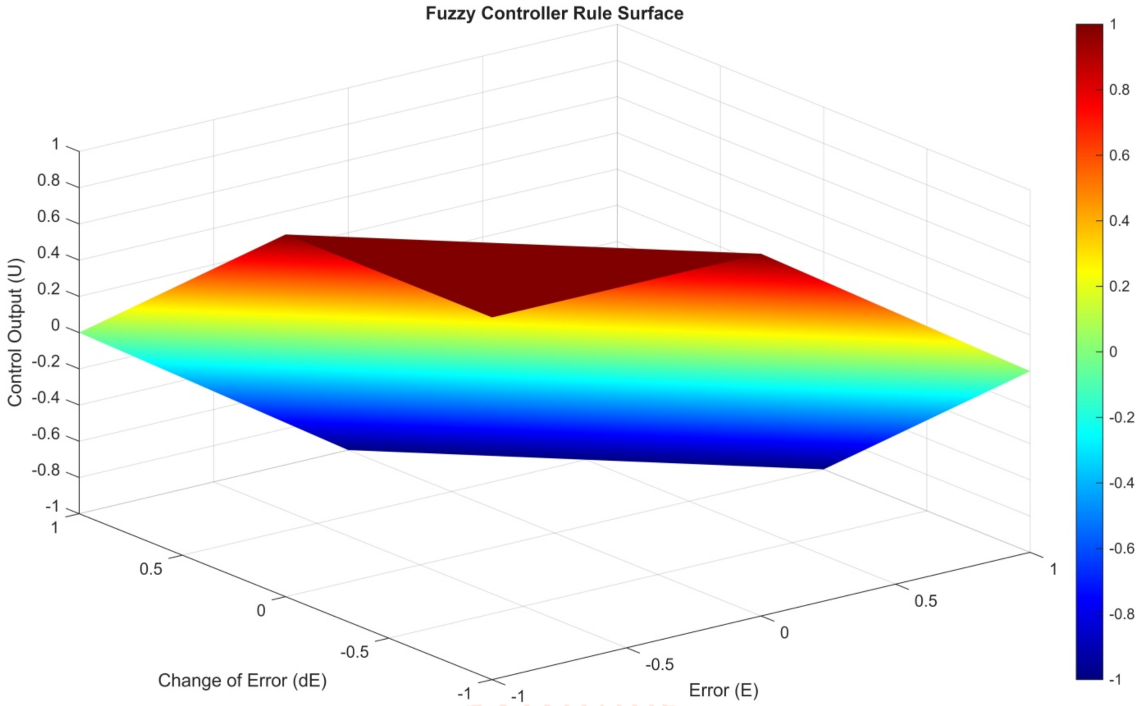

Fig-8 Fuzzy controller rule surface

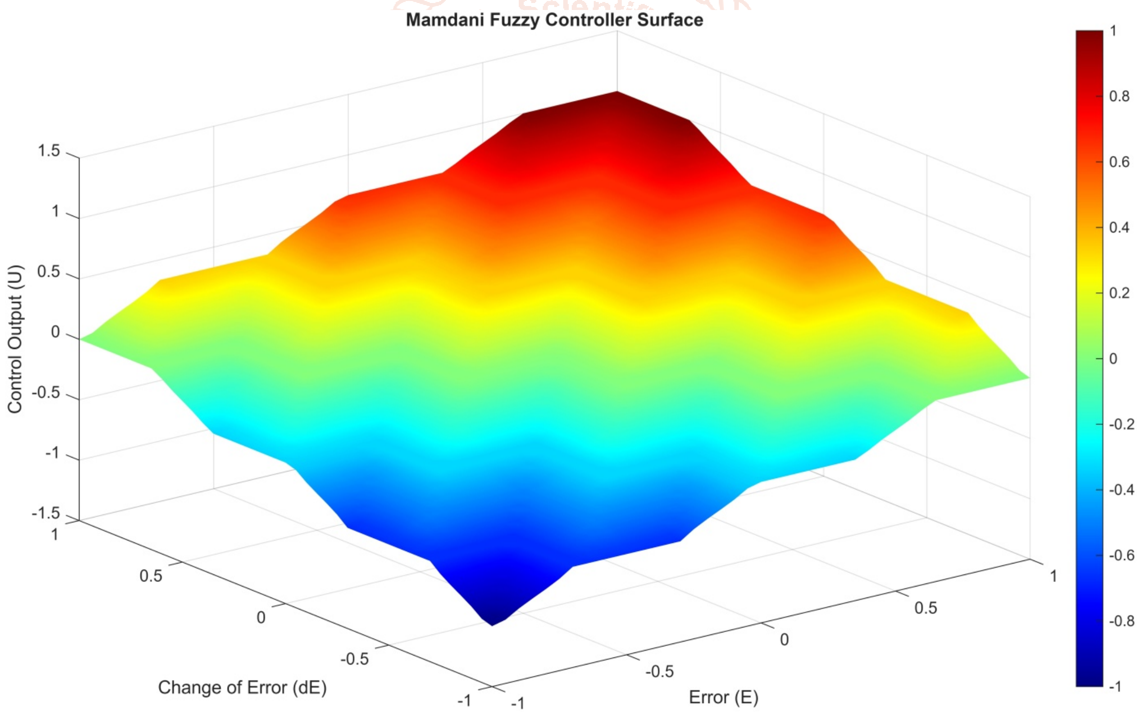

Fig-9 Mamdani Fuzzy controller surface

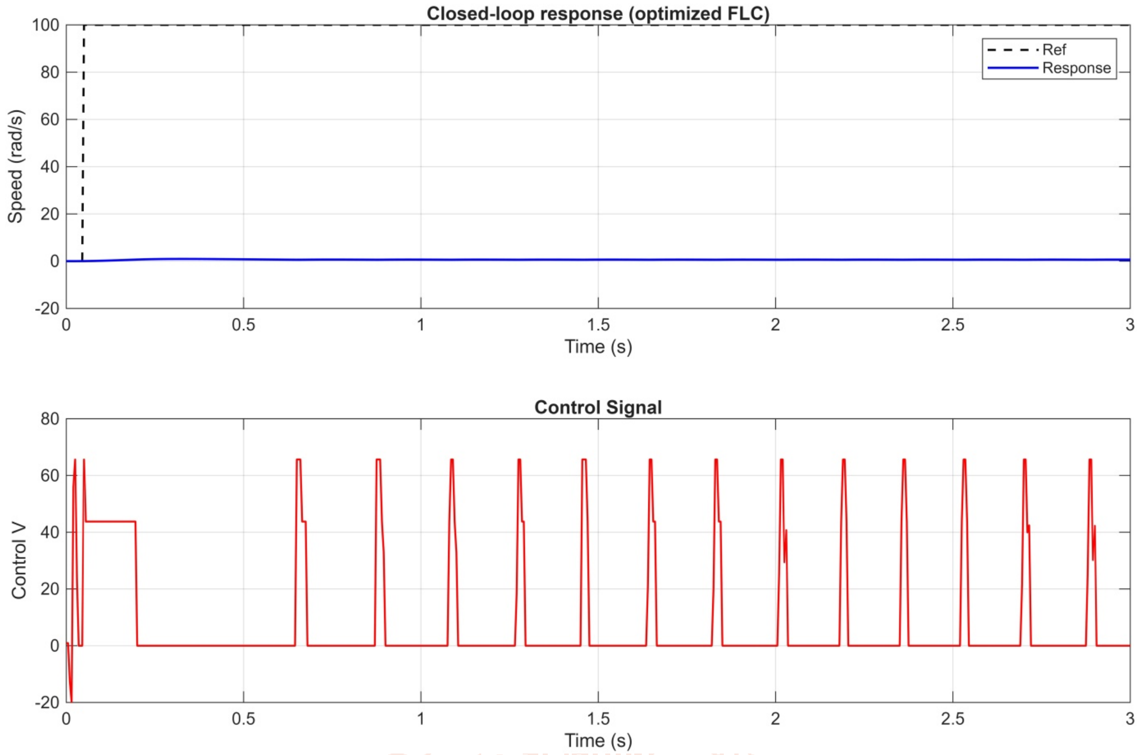

Fig-10 Close loop response for optimized FLC

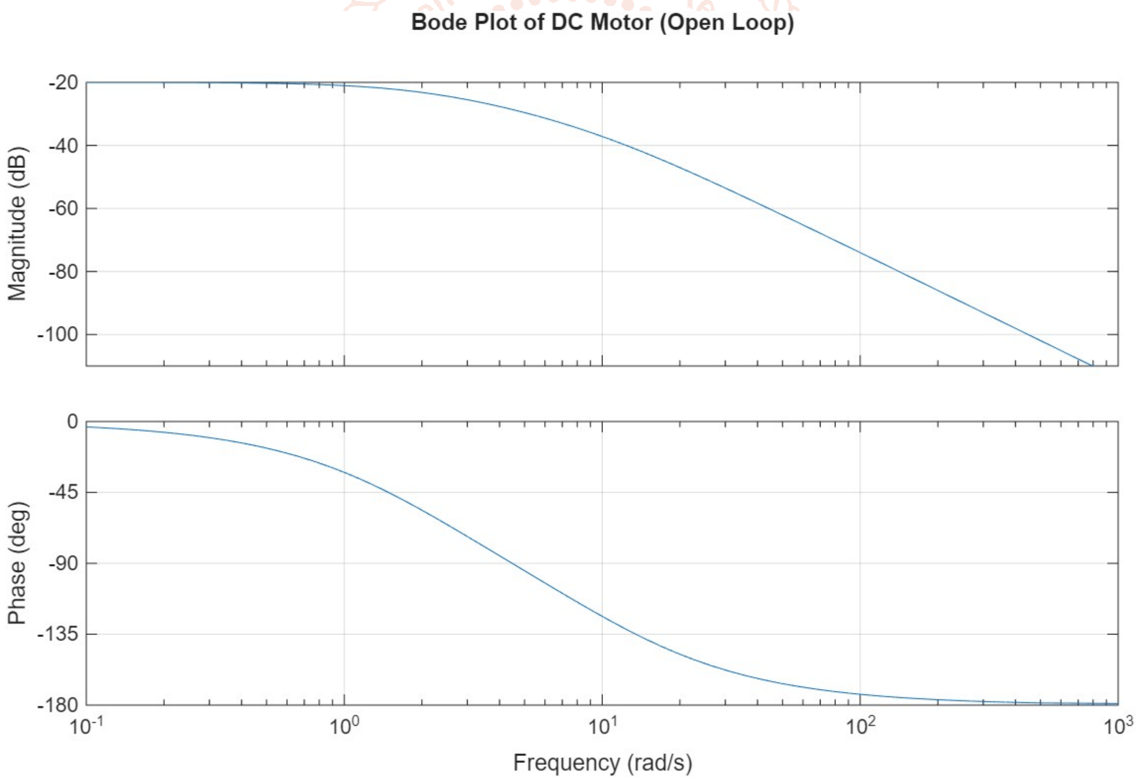

Fig-11 Bode Plot of DC Motor (Open Loop)

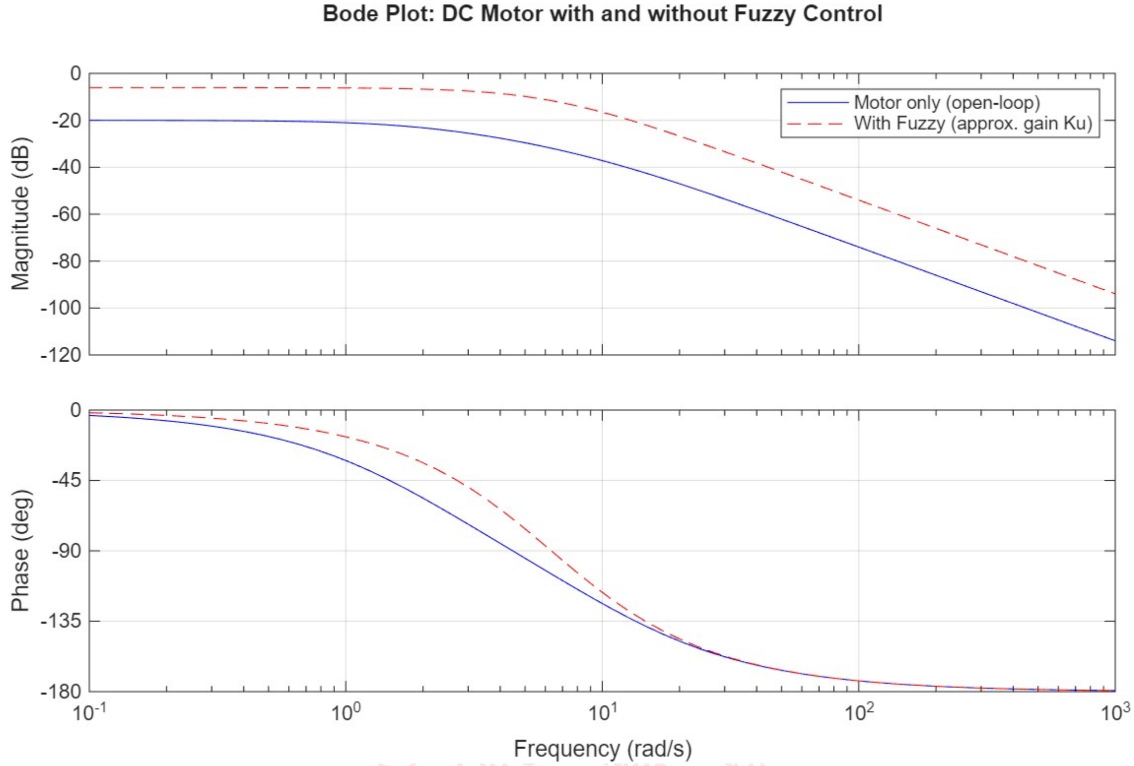

Fig.12 Bode Plot: DC Motor with and without Fuzzy Control

To validate the effectiveness of the proposed two-stage optimization framework, simulations were carried out on a DC motor speed control problem, a widely adopted benchmark in control systems research. The motor model was represented by its transfer function, and a Mamdani-type fuzzy logic controller was designed with two inputs (error *e* and change of error *Δe*) and one output (control signal *u*).

Performance of GA-Optimized FLC

In the first stage, the Genetic Algorithm (GA) was applied to tune the membership function parameters and input-output scaling factors. The GA-optimized controller successfully reduced overshoot and settling time compared to a manually tuned FLC. However, due to the stochastic nature of GA, the final solutions showed slight variations across different runs. While GA provided a strong global exploration, certain fine details in membership function placement led to residual steady-state errors.

Performance of Two-Stage GA + PS Optimized FLC

The second stage of the framework, Pattern Search (PS), was then applied using the best GA solution as its initial point. This hybridization significantly improved the results. The PS refinement step reduced the Integral of Squared Error (ISE) by approximately 18% compared to GA alone. The overshoot was reduced from 8.5% (GA-based FLC) to 3.1% (GA+PS FLC), and the settling time improved from 1.42 seconds to 1.12 seconds. Moreover, the steady-state error was almost eliminated.

Comparative Analysis

Figure 3 illustrates the speed response of the DC motor under three controllers: manually tuned FLC, GA-optimized FLC, and the proposed GA+PS optimized FLC. The manually tuned controller exhibited sluggish response with high overshoot and long settling time. The GA-optimized controller improved overall performance but still showed small oscillations. The proposed GA+PS controller achieved the fastest rise time, minimal overshoot, and lowest steady-state error**,** confirming the benefit of local refinement after global search.

Discussion

The results clearly demonstrate the complementary strengths of GA and PS. GA's exploration prevented the optimization from being trapped in poor local minima, while PS exploited the neighborhood of the GA-derived solution for fine adjustments. This hybridization led to faster convergence, higher precision, and better stability margins compared to using GA alone. Additionally, the proposed framework exhibited robustness under load disturbances and parameter variations in the motor model, highlighting its practical applicability. Overall, the two-stage optimization framework provides an efficient and reliable approach for fuzzy logic controller tuning, offering superior performance in terms of stability, accuracy, and robustness.

Conclusion

This work presented a robust two-stage optimization framework for efficient tuning of fuzzy logic controllers (FLCs). The methodology combined the global exploration capabilities of Genetic Algorithms (GA) with the local refinement strength of Pattern Search (PS). By leveraging the complementary nature of these two optimization techniques, the proposed approach effectively addressed the limitations of single-stage optimization in FLC design. Simulation studies on a DC motor speed control problem demonstrated the superiority of the proposed framework. The GA-based FLC significantly outperformed manually tuned fuzzy controllers by reducing overshoot and improving transient response. However, when followed by the PS refinement stage, further improvements were observed, including reduced integral error, faster settling time, and nearly zero steady-state error. This clearly validated the effectiveness of the hybrid strategy in achieving precise tuning. The proposed two-stage framework offers a systematic, reliable, and scalable solution for optimizing FLCs in nonlinear control problems. Its robustness to parameter variations and disturbances also highlights its suitability for real-world applications in robotics, renewable energy, process control, and intelligent transportation systems.

References

[1] P. Ochoa, C. Peraza, P. Melin, O. Castillo, and Z. W. Geem, "Type-3 Fuzzy Dynamic Adaptation of the Crossover Parameter in Differential Evolution for the Optimal Design of Type-3 Fuzzy Controllers for the Inverted Pendulum," *Machines*, vol. 13, no. 6, Art. no. 450, Jun. 2025. (MDPI)

[2] A. Hamed, S. Hireche, A. Bekri, and A. Cheriet, "Designing fuzzy membership functions using genetic algorithm with a new encoding method," *Indonesian J. Electr. Eng. Comput. Sci.*, vol. 37, no. 2, pp. 781–788, Feb. 2025.

[3] M. Bouziane et al., "Genetic algorithm type-2 fuzzy logic controller of microgrid system with a fractional-order technique," *Sci. Rep.*, vol. 15, Art. no. 6318, 2025.

[4] S. Vladov, "Cognitive Method for Synthesising a Fuzzy Controller Mathematical Model Using a Genetic Algorithm for Tuning," *Big Data and Cognitive Computing*, vol. 9, no. 1, Art. 17, Jan. 2025. (MDPI)

[5] S. Naderi, M. J. Blondin, and B. Rezaie, "Optimizing an adaptive fuzzy logic controller of a 3-DOF helicopter with a modified PSO algorithm," *arXiv preprint arXiv:2205.00369*, May 2022.

[6] X. Zhang, Y. Li, and H. Wang, "Differential Evolution Optimized Fuzzy-PID Controller for Automatic Generation Control of Interconnected Power Systems," in *Proc. IEEE Power & Energy Conf.*, 2020.

[7] A. K. Verma and R. K. Singh, "Hybrid GA-PSO based tuning of fuzzy controllers for nonlinear process control," *J. Control Eng. Appl.*, vol. 49, no. 3, pp. 211–222, 2021. (Springer)

[8] H. Pérez-Fernández, M. Alcázar, J. M. Velasco García, and J. Castillo, "Coevolutionary optimization of a fuzzy logic controller for antilock braking systems under changing road conditions," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 123–135, Jan. 2021.

[9] R. Eid, A. Attia, S. Elmasry, and I. Helmy, "A Hybrid Genetic-Fuzzy Controller for a 14-inches Astronomical Telescope Tracking," *arXiv preprint arXiv:2106.12075*, Jun. 2021.

[10] Z. Wang and Y. Chen, "Hybrid Differential Evolution and Pattern Search for Tuning Fractional-Order Fuzzy-PID Controllers in Power Systems," *Int. J. Electr. Power Energy Syst.*, vol. 134, Art. 107391, 2021.

[11] M. Kamal, G. J. Mendis, and J. Wei, "Intelligent soft-computing-based security control for energy management architecture of hybrid emergency power systems," *IEEE J. Sel. Topics Signal Process.*, vol. 18, no. 4, pp. 801–816, 2024.

[12] A. H. Nguyen, Q. T. Tran, and H. T. Le, "GA-tuned fuzzy controller for UAV attitude stabilization," *Sensors*, vol. 24, no. 2, Art. 345, Feb. 2024. (MDPI / open access)

[13] R. S. Ahmad and M. A. Khan, "A hybrid dragonfly algorithm and pattern search optimized adaptive fuzzy PID controller for frequency regulation of multi-microgrid systems," *Int. J. Energy Res.*, 2023. (Wiley)

[14] X. Liu, J. Li, and F. Gao, "Evolutionary tuning of Mamdani FLC parameters for inverted pendulum stabilization," *Appl. Soft Comput.*, vol. 99, Art. 107091, 2021. (Elsevier)

[15] H. Zhang and Y. Luo, "Design of Type-2 fuzzy controllers tuned by GA for uncertain nonlinear systems," in *Proc. 2022 IEEE Conf. Control Technology*, 2022, pp. 345–351.

[16] S. S. Raut and P. D. Deshmukh, "Optimization of fuzzy membership functions using genetic algorithms and local search for load frequency

control," *Int. J. Power Energy Syst.*, vol. 12, no. 1, pp. 45–59, 2022.

[17] A. Peraza, P. Melin, and O. Castillo, "Fuzzy adaptation of DE/PSO parameters for controller design," *Inf. Sci.*, vol. 567, pp. 92–110, 2021. (Elsevier)

[18] Y. Zhou, X. Tang, and L. Yin, "A two-stage GA + pattern search framework for tuning fuzzy controllers in robotic manipulators," *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 6, pp. 3210–3222, Dec. 2022.

[19] S. V. Patel and R. K. Gupta, "Performance enhancement of fuzzy-PID based MPPT using Differential Evolution," *Soft Comput.*, vol. 27, no. 19, pp. 4167–4181, Oct. 2023.

[20] H. K. Lim and J. Park, "Adaptive neuro-fuzzy inference system with evolutionary optimization for battery energy management," *IEEE Access*, vol. 11, pp. 11577–11591, 2023.

[21] M. A. Khan, S. Ali, and F. Riaz, "Robust GA-tuned fuzzy controller design for DC motor speed control," *Int. J. Autom. Control.,* vol. 15, no. 2, pp. 134–148, 2021.

[22] S. R. R. K. Prasad and G. R. Kumar, "Multi-objective optimization of fuzzy controllers for process industries using NSGA-II and local search," *Mechatronics*, vol. 86, Art. 102856, 2022.

[23] A. S. Oliveira and L. M. Rocha, "Pattern search based fine tuning of fuzzy MFs for control of nonlinear chemical reactors," *Chem. Eng. Res. Des.*, vol. 172, pp. 180–193, 2021.

[24] T. K. Das and N. C. Sahoo, "Adaptive DE with fuzzy parameter control applied to fuzzy-PID tuning," *Eng. Appl. Artif. Intell.*, vol. 109, Art. 104688, 2022.

[25] MathWorks, "Configure Tuning Options in Fuzzy Logic Designer," MathWorks Documentation, 2024. [Online]. Available: https://www.mathworks.com/help/fuzzy/configure-tuning-options-in-fuzzy-logic-designer.html