Proposed Project :

Optimization of PI Controller Parameters for DC Motor Speed Control Using Particle Swarm Optimization (PSO) and Genetic Algorithm (GA)

1. Problem Statement

Controlling the speed of a Direct Current (DC) motor with high precision and stability is essential in numerous applications, such as robotics, automation, and industrial processes. The Proportional-Integral (PI) controller is widely used for this purpose due to its simplicity and reliability. However, the performance of a PI controller strongly depends on the correct tuning of its parameters (K_p and K_i). Conventional tuning techniques, such as Ziegler–Nichols or trial-and-error methods, often fail to provide satisfactory performance in systems affected by nonlinearities, parameter variations, and external disturbances.

To overcome these limitations, metaheuristic optimization algorithms, such as Particle Swarm Optimization (PSO) and Genetic Algorithm (GA), can be used to automatically and efficiently determine the optimal PI parameters that improve dynamic performance.

This project aims to develop an optimized speed control strategy for a DC motor using PSO and GA and to validate the results experimentally through an Arduino-based implementation.

2. Objectives

The main goal of this project is to design and implement an optimized PI controller for DC motor speed regulation using PSO and GA.

Specific Objectives :

1. **Modeling:** Develop the mathematical model and transfer function of a DC motor that relates input voltage to angular speed.
2. **Controller Design:** Implement a PI controller to regulate the motor speed and analyze its performance using conventional tuning methods.
3. **Optimization:** Apply PSO and GA algorithms to find the optimal values of $K_p$and $K_i$that minimize error-based performance indices (e.g., IAE, ISE, ITAE).
4. **Simulation:** Develop a MATLAB/Simulink model integrating the DC motor, PI controller, and optimization algorithms to compare classical and optimized control performances.
5. **Experimental Validation:** Implement the optimized PI controller on a real DC motor using Arduino hardware and validate the simulation results experimentally.
6. **Performance Comparison:** Evaluate the improvement in speed response, settling time, overshoot, and steady-state error for the optimized controllers (PSO and GA) compared to conventional methods.

3. Simulation and Experimental Validation

Simulation Phase (MATLAB/Simulink):

- Develop the DC motor model using differential equations and obtain its transfer function.
- Design a standard PI controller and simulate its performance for different load conditions.

- Implement PSO and GA algorithms to optimize $K_p$and $K_i$based on a selected objective function (e.g., minimizing ITAE).
- Compare the time-domain responses (rise time, overshoot, settling time, steady-state error) for:
    - Conventional PI
    - PSO-optimized PI
    - GA-optimized PI
- Analyze robustness and disturbance rejection capabilities.

Experimental Validation (Arduino Platform):

- Use Arduino to implement the optimized PI control in real time.
- Measure motor speed using an encoder or Hall-effect sensor.
- Interface the motor with a driver module (e.g., L298N) and collect data via serial communication.
- Compare the experimental results with the simulation outcomes to evaluate accuracy and real-world applicability.
- Assess controller performance under different operating conditions (load torque variations, voltage fluctuations, etc.).

4. List of Materials

Category	Component	Description / Function
Control Hardware	**Arduino Uno or Mega**	Microcontroller board for implementing the controller
Motor System	**DC Motor (12V or 24V)**	Main actuator for speed control
Motor Driver	**L298N / L293D Module**	Interface circuit to control motor direction and speed
Speed Sensor	**Rotary Encoder or Hall Sensor**	For real-time measurement of motor speed
Power Supply	**12V DC Power Adapter**	To power the motor and Arduino system
Signal Conditioning	**Resistors, Capacitors**	Used for filtering and protection
Data Acquisition	**USB Cable / Serial Interface**	For communication between Arduino and MATLAB
Simulation Software	**MATLAB/Simulink**	Modeling, control design, and optimization environment

Category	Component	Description / Function
Breadboard & Wires	**Connecting Components**	For prototyping and circuit assembly
Load (optional)	**Variable Mechanical Load**	To test controller robustness under varying torque

5. Expected Outcomes

- Optimized PI controller parameters (K_p, K_i) obtained through PSO and GA.
- Enhanced speed tracking performance with reduced overshoot, faster response, and minimal steady-state error.
- Demonstrated improvement of metaheuristic optimization techniques over classical tuning.
- Successful hardware validation confirming simulation results.