MCTA 4362 Machine Learning: Mini Project

Design of Intelligent Controllers Using Reinforcement Learning for Control Applications [20 marks]

1) This project is group work of max 4 persons, and the total mark is 20.
2) You can use any programming software (e.g. Python, Matlab, Unity3D(C#), Scilab, Octave, FreeMat, etc).
3) Presentation Venue: Advanced Multi-Agent System Lab, E0 Building, Level 2.
4) Upload your code and simulation file to your own Github Repository and share your Github link together with your report to the Google Drive link (link will be given later).

Background

Classical controllers such as Proportional (P), Proportional-Integral (PI), and Proportional-Integral-Derivative (PID) are widely used in control systems due to their simplicity and ease of implementation. These methods work well in linear and well-modeled environments but tend to struggle when systems are nonlinear, time-varying, or subject to unpredictable disturbances.

Reinforcement Learning (RL) has emerged as a powerful tool for control problems where an accurate model of the plant is difficult to obtain. By interacting with the environment, an RL agent can learn optimal control policies through trial and error, making it suitable for a wide range of applications in control engineering.

Problem Statement

Many real-world systems require intelligent and adaptive control strategies beyond conventional PID methods. Your task is to select any suitable plant or control system (e.g., DC motor, inverted pendulum, ball and beam, temperature control, magnetic levitation, etc.) and:

- Design a Reinforcement Learning-based controller for it
- Compare the performance with a classical P/PI/PID controller
- Demonstrate the strengths and limitations of both approaches

This open-ended control problem gives you the flexibility to choose a plant and RL method that aligns with your group's interest and creativity.

Objectives

1) To model and simulate a control system (plant of your choice).
2) To implement Reinforcement Learning algorithms (e.g., Q-learning, DQN, PPO, etc.) to train an intelligent controller.
3) To compare the RL controller against classical control methods (P, PI, PID) using appropriate performance metrics.

Instructions: By using your knowledge and intuition, design an intelligent controller for realizing this intelligent system. You need to do the followings:

1) **System Selection and Modelling:**
   - Choose a dynamic system (plant) suitable for control (e.g., robotic arm, temperature system, DC motor, etc.)
   - Model the system using differential equations or Simulink blocks
   - Define the control objective: tracking, regulation, stabilization, etc.
2) **RL-based Controller Development:**
   - Choose an appropriate RL algorithm
   - Train an agent to control the selected system using trial-and-error learning
   - Tune rewards, state/action representations, and simulation environments
   - **BONUS MARKS (up to 5): GUI implementation or hardware testing (if applicable).**
3) **Evaluation and Comparison:**
   - Compare your RL controller against a traditional controller (P, PI, PID)
   - Evaluate performance based on:
     - Rise time
     - Overshoot
     - Settling time
     - Robustness to disturbances
     - Mean Squared Error (MSE), etc.
4) **Documentation and Presentation:** Clearly explain your methodology, simulations, and analysis in your report and prepare a presentation summarizing the project and results.

Deliverables

1) Final Project Report:
   - Present a comprehensive report documenting the project implementation, including introduction, RL modeling, simulation, and analysis of results.
2) Presentation:
   - Deliver an engaging presentation summarizing the project, highlighting the key aspects, methodology, and findings.
3) Code and Documentation:
   - Submit well-documented code that demonstrates the implementation of the methods and simulation techniques used in the project.
4) Individual Contributions:
   - Each team member should submit a summary of their individual contributions to the project.