**CPSC 4420/6420 Artificial Intelligence.**

**Fall 2023 Term Project**

**Due Date: Dec/05/2023**

---

**Reinforcement Learning for UAVs with Manipulator Arms**

**Introduction:**

This project draws inspiration from the paper "Actuator Trajectory Planning for UAVs with Overhead Manipulator using Reinforcement Learning.". This project involves developing a control system for a UAV (Unmanned Aerial Vehicle) with an overhead manipulator, applying reinforcement learning to achieve accurate trajectory tracking. The challenge is to design, implement, and analyze a system that can adapt to a dynamic environment, using a variety of learning algorithms and techniques.

**Objective:**

To apply reinforcement learning in a complex robotics scenario, focusing on trajectory planning and manipulation using a UAV.

**Detailed Problem Formulation:**

1. **Base and Target Trajectory Definition:**
    - Establish a base trajectory for the UAV and a desired target trajectory for the manipulator's end-effector. You are given the initial the end-effector target trajectory and the UAV base trajectory (ee_path.npy, base_path.npy).
    - In the project, you are only required to use the kinematic equations to model the manipulator arm's movement, you can also develop a full dynamic model for a more realistic action space formulation which would increase the robustness of your model.
2. **Markov Decision Process (MDP) Formulation:**
    - Define the state space, incorporating UAV and manipulator positions, orientations, and velocities.
    - Detail the action space, including motor commands (angles if you opted to only use the kinematic model) and manipulator movements.
    - Develop the transition model.
    - Create a reward function that incentivizes trajectory adherence, smooth movement, and obstacle avoidance.

3. **Application of Reinforcement Learning Methods:**
    - Implement a basic Reinforcement learning method like SARSA and Q-learning to develop initial control policies and check the validity of your MDP formulation.
    - Explore a more advanced technique such as Deep Q-Network (DQN), Proximal Policy Optimization (PPO), Soft Actor Critic (SAC) or any algorithm you believe is suitable for improved performance.
    -
4. **Results and Visualization:**
    - Develop visual presentations and simulations to demonstrate how the UAV and its manipulator operate under different policies. You can use pygame for example.
    - Investigate the effects of model parameters (Episodes, Learning rate, ...etc) on the model performance.
    - Introduce controlled noise in the motor system (action phase) to simulate real-world unpredictability and observe its impact.
5. **Optional Extensions for Creativity (Extra Credit):**
    - Graduate students are highly encouraged (and undergrads are also welcome) to explore ideas in other related topics such as robust reinforcement learning, action-critic methods, inverse learning, learning under partial and inaccurate observation, imitation learning, etc. and apply the results to this problem.
    - Implement computer vision techniques to track the position of the end-effector.
    - Introduce computer vision techniques for state representation and target trajectory identification.
    - Extend the manipulator arm to include more joints (e.g., a 3-joint arm) for a more complex control challenge.
    - Introduce drone rotations (roll,pitch,yaw) to the model for a more realistic simulation.
    - You are highly encouraged to simulate this project using a simulation platform such as ROS/Gazebo, Unity, PyBullet.

**Deliverables:**

- **Project Report:** Documenting the approach, methodology, challenges, and findings.
- **Code and Simulations:** Complete codebase along with simulation results.
- **Presentations:** A comprehensive presentation showcasing the project's outcomes, including visual aids and demonstrations.
- **Readme:** A readme file with details and instructions on your code.

**Model Parameters:**

Following are the model parameters required to develop your model.

The UAV is equipped with an overhead two-link manipulator as shown below.

Assume the arms are 50cm each and are massless.

The joint angle range is $(40° \ \ to \ \ 90°)$ for q1 and $(-70° \ \ to \ \ -20°)$ for q2.

Assume the drone angle of attach $\alpha = 0$

You can add your assumptions if you opted to include the model dynamics. Please contact Hazim ([halzorg@clemson.edu](mailto:halzorg@clemson.edu)) to assist you with these assumptions.

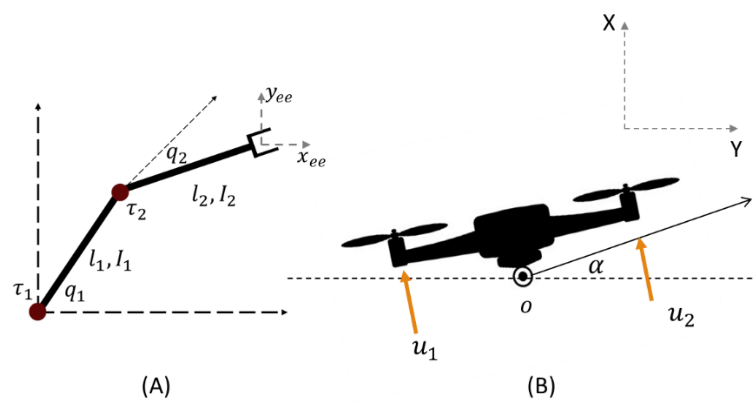


*Figure 1 Manipulator Equipped UAV schematics*

Where $\tau_1 \tau_2$ are the joint torques, $q_1 q_2$are the joint angles, and $l_1 l_2$are the arm lengths.

(ee_path.npy) and (base_path.npy) are provided for your use.

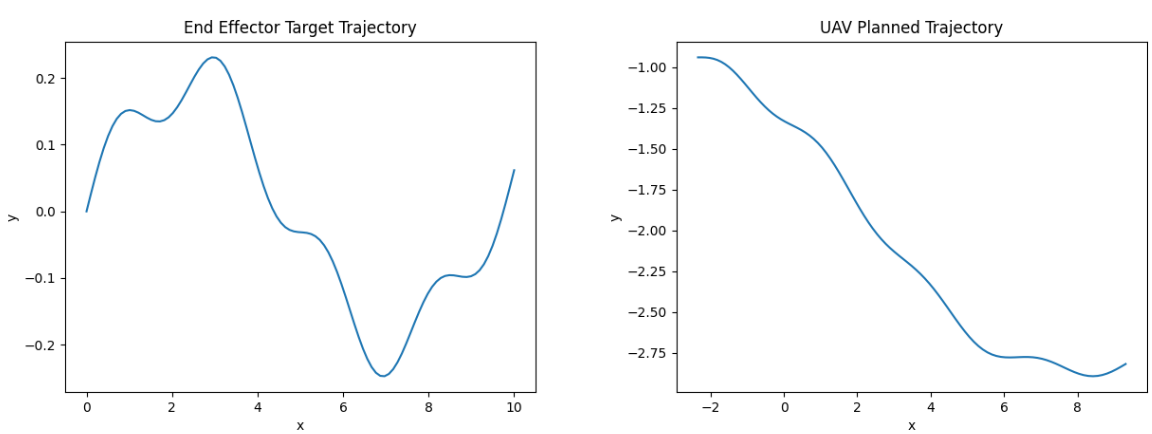


*Figure 2 left EE trajectory, right UAV trajectory.*