AR523: Robot Manipulators

Assignment 2: Force control of a robotic arm

Overview:

- This assignment focuses on implementing and analyzing force control strategies for robotic manipulators. You will explore admittance and impedance control schemes.

- The simulation environment is PyBullet (Fig. 1), which provides realistic collision checking and kinematics utilities. You may refer the following for learning PyBullet: PyBullet Official Documentation or Workshop on Pybullet by IIT-BHU robotics club

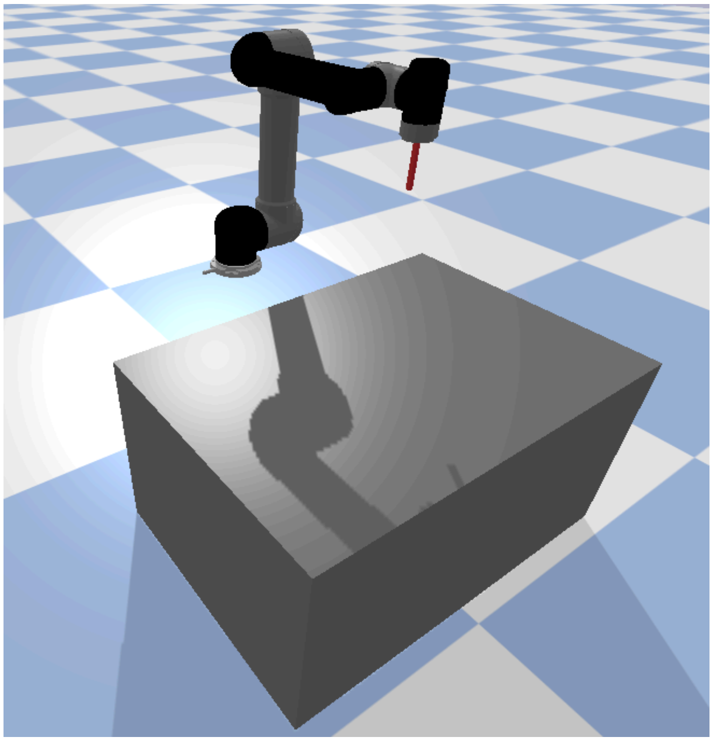

Fig 1: PyBullet simulation environment

- Please start the assignment on time. A viva-based interaction will be conducted for evaluation. Refer to lecture notes for theoretical background.

We hope that this assignment deepens your understanding of motion planning for manipulators!

Exercises:

The starter code and specific tasks to be carried out are provided in the form of a Github repository at this link. Please follow the description provided. If you find any kind of error in basic code, please correct yourself

The assignment has 6 exercises

- Part 1: Admittance control for static interaction
- Part 2: Admittance control for dynamic interaction
- Part 3: Impedance control for static interaction
- Part 4: Impedance control for dynamic interaction
- Part 5: Analysis and Comparison
- Part 6: Evaluation on unseen environment

Static Interaction: The end-effector interacts with the environment at a fixed point (e.g., maintaining contact force on a stationary surface).

Dynamic Interaction: The end-effector moves along a continuous path (e.g., circular motion) while maintaining a desired contact force with the environment.

For **parts 1-4**, you are expected to deliver the following:

- Code implementation for both the controllers and the interactions
- Plots of motion trajectories and contact forces (real-time plotting preferred)

For **part 5**, prepare a comparative analysis between admittance and impedance control schemes. Discuss their advantages, limitations, and suitable application domains. Additionally, you can also use any metrics to compare them based on your understanding. Some are provided here for reference: Force tracking error, Settling time, Motion smoothness

For **Part 6**, your planner will be tested on a different environment, where surface topology will be varied. The goal is to evaluate adaptability and robustness of your implementation.

Extra Credits:

- Extend your implementation to handle interactions with non-planar surfaces.
- Implement online adaptation of virtual mass, damping, or stiffness based on contact conditions.

Instructions:

- This assignment is to be done individually or in pairs (working in pairs is recommended and will be rewarded). The choice is yours. If you choose to work in a pair, try pairing with someone you have not paired in Assignment 1 (new pairs will be rewarded and old pairs will be penalized).
- We will create a google group for discussion. Please post your queries and also help others by responding to their queries (This will be rewarded too!)
- Please submit it on google form as a single zip file named *<A1_StudentID>.zip* or *<A1_StudentID1_StudentID2>.zip*. The zip file should contain a full code, running instructions, and analysis in the PDF file.
- The **submission date** is 5:00 pm IST on Tuesday, 11th November, 2025. Late submission will incur a daily 10% score adjustment for up to two days.
- This assignment will carry ~5-15% **weightage** overall. As mentioned, the instructor will provide the final weightage after closure of all assignments.
- Your first submission within the prescribed time period is final. Please do not ask for replacing the files later (e.g., on account of any incorrect upload). Such tampering beyond the submission time is not fair to other students in class who submit on time.
- The assignment must be done only from your own original efforts. Do not use any existing implementation/report from online sources. Do not violate the academic honesty code.
- Please write your implementation without use of AI code generators (e.g., GPT). Taking assistance from such tools will take away the opportunity to deeply understand the working of the algorithms. Hence, it is important to do this exercise on your own. A standard AI code generated solution will be part of automated code similarity checking, Submissions that resemble similarity beyond acceptance to standard AI-generated code will be flagged and treated as external assistance beyond your own effort.