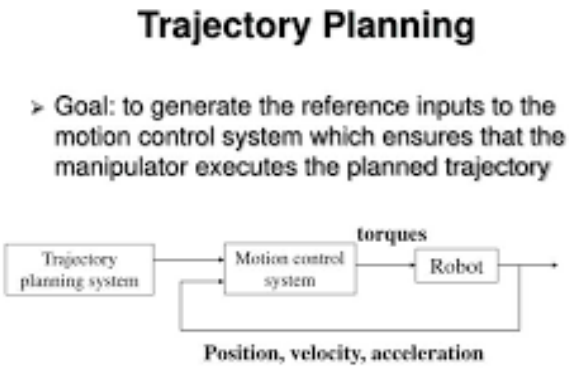
WEEK 4

We'll be starting with Week 4 of the course. This week deals with Introduction to Trajectory Planning.

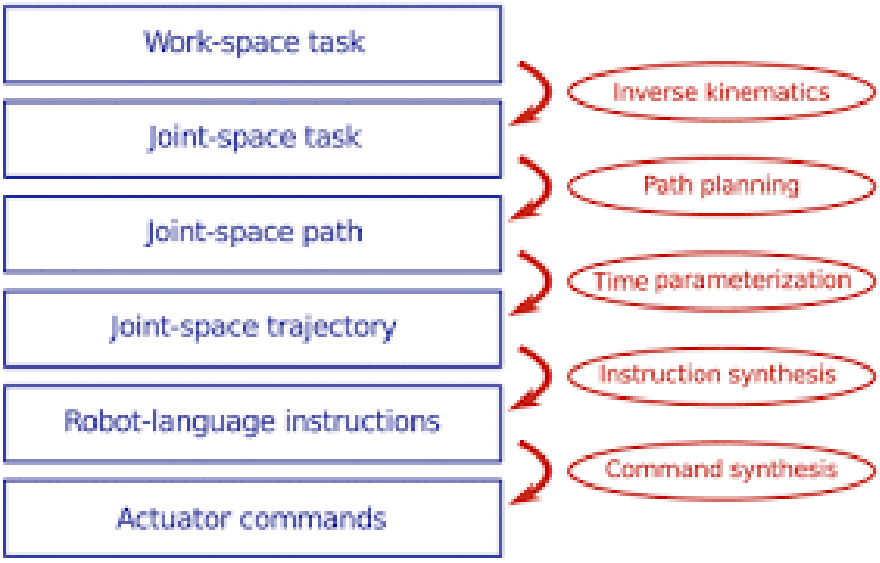
INTRODUCTION TO TRAJECTORY PLANNING

Trajectory planning is moving from point A to point B while avoiding collisions over time. This can be computed in both discrete and continuous methods. Trajectory planning is a major area in robotics as it gives way to autonomous vehicles.

Trajectory planning is sometimes referred to as motion planning and erroneously as path planning. Trajectory planning is distinct from path planning in that it is parametrized by time. Essentially trajectory planning encompasses path planning in addition to planning how to move based on velocity, time, and kinematics.



Tasks of robot control can be classified in different ways. For example, different path planning strategies can be used in the case of different situations. There are two types of constraints that must be considered in path planning. First, the motion of a robot can be restricted by obstacles and obstacle constraints have to be used. On the other hand, there can be some kind of constraints for path selection. These constraints are known as path constraints.



The main path planning tasks for a robot are as follows:

- grasping and releasing objects - moving from place to place
- following previously specified paths - following moving objects
- working with other manipulators
- exerting forces (i.e. pushing, pulling and holding)
- exerting torques
- collecting data
- using tools

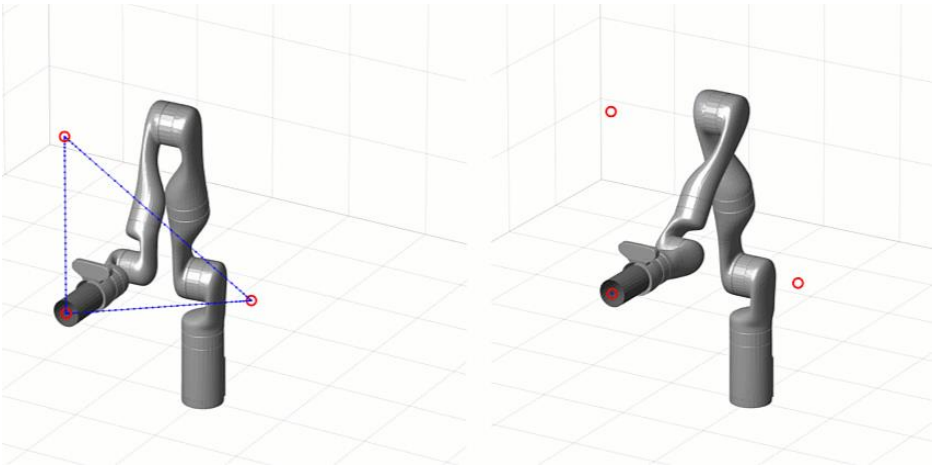
RESOURCES ON TRAJECTORY PLANNING

1. Trajectory Planning : Lecture Series on Robotics by Prof. P.Seshu, Department of Mechanical Engineering,IIT Bombay.
2. Trajectory Planning for Robot Manipulators - YouTube by MATLAB
3. Trajectory Planning for Robot Manipulators - MATLAB & Simulink (mathworks.com)
4. TrajectoryPlanningJoints.pdf By Prof. Alessandro De Luca

USING RST TO SIMULATE JOINT SPACE TRAJECTORIES IN MATLAB ON A 2D ROBOT MANIPULATOR

The following tutorial gives a detailed description on using different methods to simulate joint space trajectory on 2D Robot manipulator

Simulate Joint-Space Trajectory Tracking in MATLAB - MATLAB & Simulink - MathWorks India



ASSIGNMENT

Taking the above tutorial as reference your task is to generate and simulate interpolated joint trajectories to move from an initial to a desired end-effector pose, On the robot manipulator that you created in week 3 assignment. The timing of the trajectories is based on an approximate desired end of arm tool (EOAT) speed.

timeStep = 0.01; % seconds #time steps for trajectory planning

toolSpeed = 0.1; % m/s #approximate desired end of arm tool speed

Generate both Task space trajectory and Joint space trajectory and Visualize and compare both the trajectories in the end.

Hint: For task space trajectories compute task-space trajectory waypoints via interpolation between initial and final position. Read about the function `transformtraj`

You are allowed to set any initial and final end effector pose, but it should be clearly mentioned in your code with proper comments.

You need to submit a .m file with a small video of your simulation.