ECE 4560
Assignment 9

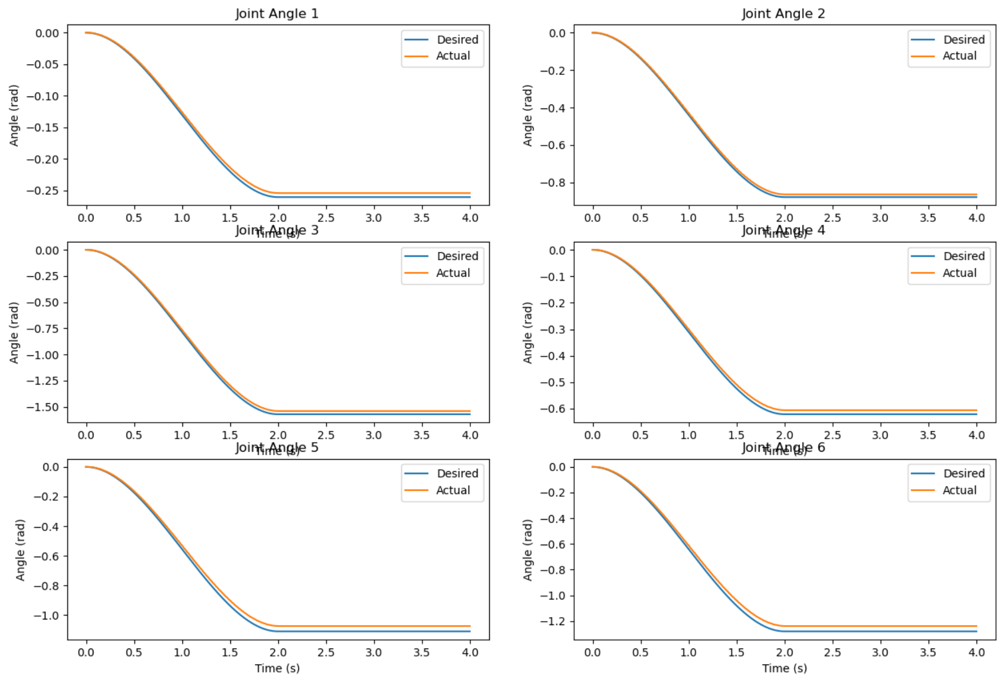
Due: November 7th, 11:59pm

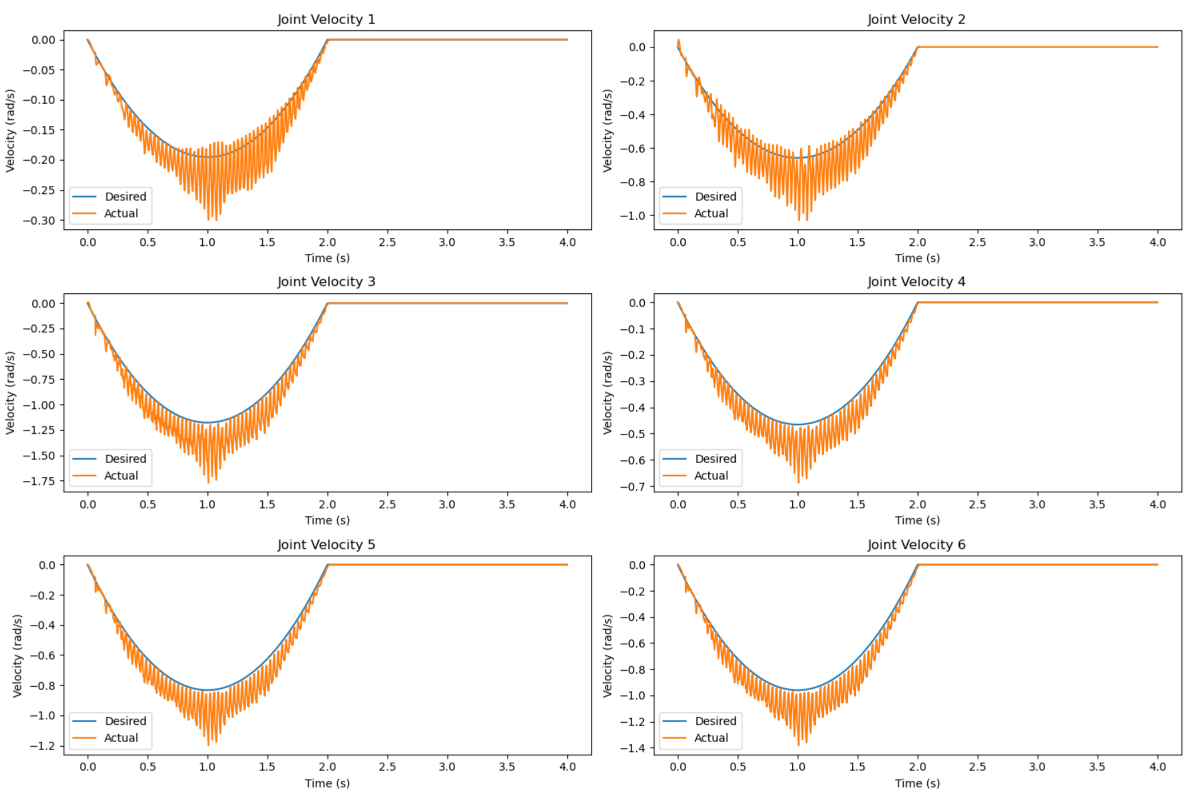
---

1. This week's homework will be implementing what we've learned in class on the UR5e robot arm in MuJoCo. This will use the same repository as in the last few homework assignments: https://github.gatech.edu/mtucker34/Mujoco-Example.

   (a) (6 points) The first half of the assignment is to program a cubic spline to move between an initial set of joint angles and a final set of joint angles. Specifically, modify lines 30 and 36 of the 'test-assignment9.py' python script. Once you're done, run the script and observe the motion of the robot. The simulation will time out after 'time_period + 2' seconds. Feel free to play with the final configuration and the time period to see how the motion of the robot changes. Once you have run the simulation, it will write a file called 'simulation_data.csv'. To plot the trajectory that was run as stored in this file, run the python script 'plot_trajectory.py'. Please submit a copy of this plot with your homework report, and write down the equations you used for the coefficients of your cubic spline, and the general cubic spline formula. A screenshot of your code is sufficient too.

   An example of the plot that you should obtain is below:


Joint Angle 1
Joint Angle 2
Joint Angle 3
Joint Angle 4
Joint Angle 5
Joint Angle 6
Angle (rad)
Time (s)
Desired
Actual


Joint Velocity 1
Joint Velocity 2
Joint Velocity 3
Joint Velocity 4
Joint Velocity 5
Joint Velocity 6
Velocity (rad/s)
Time (s)
Desired
Actual

Notice that the velocity tracking has a decent bit of chatter. This is likely due to sub-optimally tuned control gains. You are welcome to play with the gains (listed in control_utils.py) to improve the performance!

(b) (6 points) Second, please compute the spatial manipulator Jacobian using the formula:

$$J^s = \begin{bmatrix} \xi_1 & \xi_2' & \xi_3' & \xi_4' & \xi_5' & \xi_6' \end{bmatrix}$$

where the ξ_i' will be calculated using the formula:

$$\xi_i' = \mathrm{Ad}_{e^{\hat{\xi}_1\theta_1}\cdots\hat{\xi}_{i-1}\theta_{i-1}}\xi_i$$

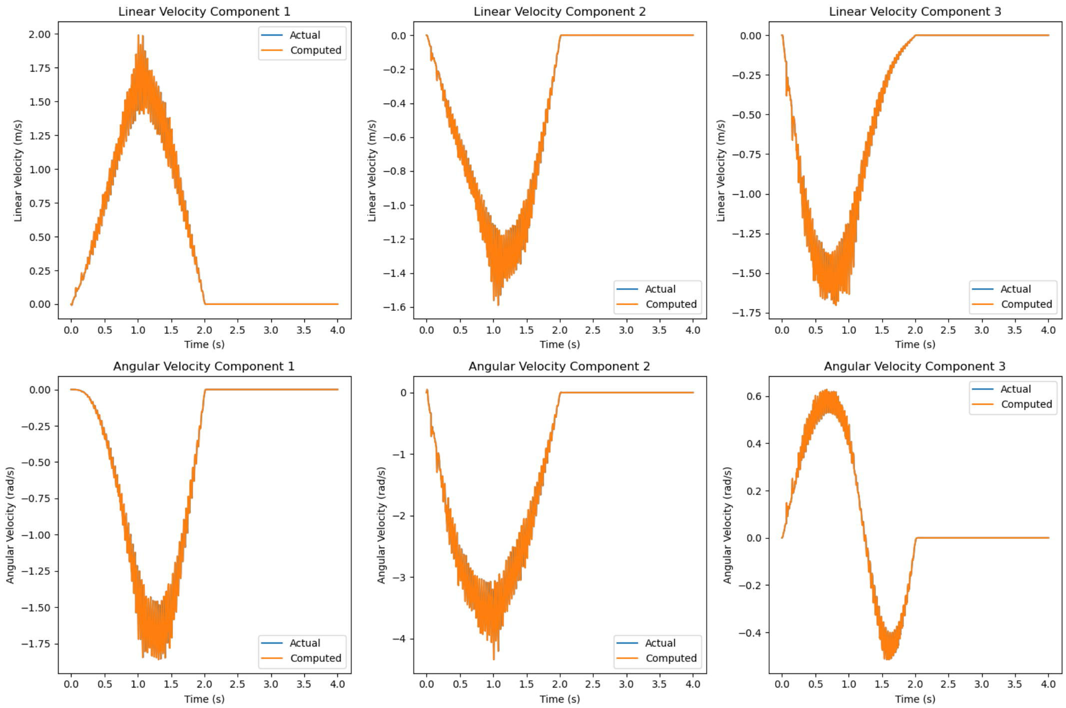
Because we are using this formula, you can use the same ξ that you calculated for the forward kinematics. Specifically, to compute the jacobian, please fill in lines '46-64' of 'jacobian.py'. When you are done, run 'test-assignment9.py' again to obtain an updated simulation_dat.csv file. Then, plot the plotting script "plot_Jacobian.py". This script will plot the spatial twist velocities as calculated using the formula:

$$\xi^s = J^s \dot{q}$$

as compared to the actual spatial velocities that are obtained using a velocity sensor and gyroscope located at the end-effector on the robot. The code is transforming these body twist velocities into spatial twist velocities using the formula:

$$\xi^s = \mathrm{Ad}_{g_e}\xi^b$$

If you've calculated your spatial Jacobian correctly, you should see that the two lines on the plots overlap:



For your homework writeup, please attach a screenshot of both your code as well as the plot that you obtain.

2. **LAB COMPONENT:** This week we will explore the implementation of cubic spline trajectories for our robot motions. For details, please refer to the course website.

   (a) (3 points) As usual, for full credit you will need both a writeup submitted with your submitted homework assignment, as well as a robotic demonstration either shown in person to the TAs or with a video (preferably a youtube link) emailed to Prof. Tucker and the TAs.