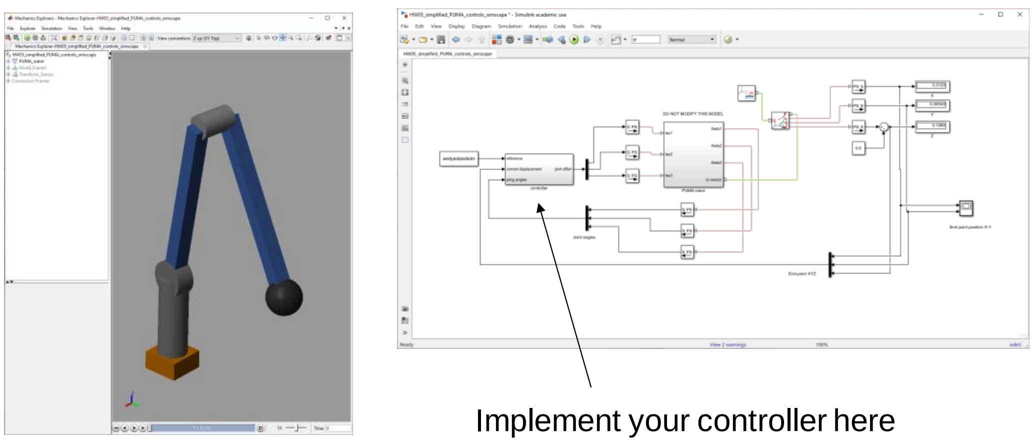
ME 6407 HW 5: Controls project

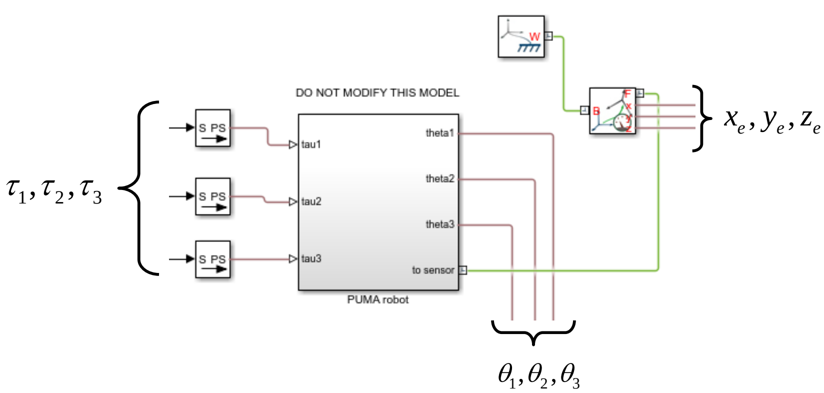
<u>Problem 1</u>: End-point control of a simplified PUMA class robot

Plan trajectories and implement a controller to position the end-point of the simplified PUMA robot with gravity, and draw symbols (1-1) and characters (1-2) on the X0-Y0 plane.

Robot system and controller:

- Simplified PUMA robot (that we used for HW02 and HW03) <u>with gravity</u>
- Inputs to the robot are <u>joint torques</u>. (you are NOT allowed to use the ideal joint motion control that SimScape Multibody provides. You should use joint torque control.)
- <u>Joint angles</u> and <u>3D end-point position (x-y-z)</u> are measurable.
- You can use any type of controller and tune control gains to satisfy the requirements. You can use different controllers and gains for Problems 1-1 and 1-2.
- You can change the initial posture defined in the setting file if needed.





The robot must satisfy the following tracking performance requirements:

1) The robot must complete the task <u>within 10 seconds</u>.
2) The maximum positioning error (instantaneous absolute error) must be <u>less than 3cm</u>.
3) The root mean squared error (RMSE) must be <u>less than 1cm</u>.
   NOTE: you can exclude the first 1 second period (t=0~1 sec) from the error evaluation due to possible large transient response. Evaluate 2) and 3) from t=1 to t=10.

Problem 1-1: Drawing of three circles

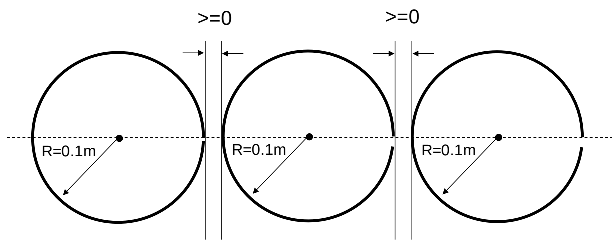


Figure 1-1: Problem 1-1 drawing task

Draw three circles with a radius of 0.1m aligned on a line. There should be zero or greater distances between the circles. Trajectory from one cycle to next needs to be planned, too.

Problem 1-2: Drawing of letters of your choice

Target word must contain at least six (6) characters, such as "GATECH", "ME6407", "ROBOTICS", "ATLANTA." You can freely choose upper and lower case alphabetical characters as well as numbers (no special or control characters. space does not count.). See a sample below. The manipulator can "write" all characters in a single stroke.

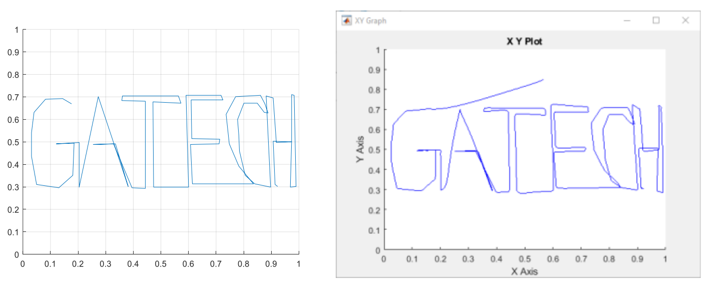


Figure 1-2: Problem 1-2 drawing task sample

Report (for each of p-1-1 & p-1-2):

1) Explanation of your controller design, block diagram(s), values of gains
2) Plots of reference trajectory and reproduced trajectory
3) Plot of joint torques versus time
4) Plot of the end-point position error versus time (indicate maximum error and RMSE)
5) Videos (use SimScape screen capturing function)

$$RMSE = \sqrt{\frac{\sum_N (e_x^2(k) + e_y^2(k) + e_z^2(k))}{N}}, (k = 1, \cdots N)$$

BONUS:

a) +10 points. Effective implementation of computational torque control
b) +10 points. For P1-1, complete the circle drawing task AS FAST AS POSSILE while still meeting the requirements 2) and 3).
c) +10 points: TA award. TA will recommend up to 5 assignments from the A/ROBO/AMZ/Q sections combined that reported excellent results and/or successful demonstration of challenging tasks for additional bonus points.

Manipulator model and link parameters: You can find geometric and material parameters of the manipulator links in the SimScape model if you wish to design a dynamics-based controller. You are not allowed to modify the dynamics of the manipulator (e.g., making links lighter or remove gravity).

MATLAB codes and SimScape models: You can use the materials listed below that are available on the course webpage.

HW05_simplified_PUMA_controls_setting.m (link parameter setting file)

HW05_simplified_PUMA_controls_simscape.slx (SimScape template model)

hw05_sample_gatechwriting1_video.mp4 and hw05_sample_circledrawing_video.mp4 (sample video clips)

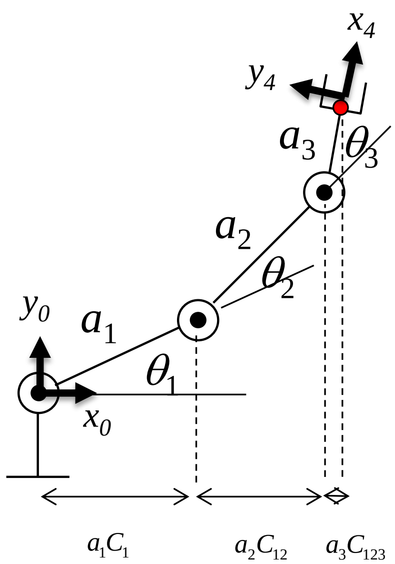
HW05_sample_trajectories.pdf (incomplete samples).

Grading: Total 30 points + 40 points + Bonus points

See Next Page for Problem 2

<u>Problem 2</u>: Planar redundant manipulator and its mainpulability

For the redundant 3R robot (the same robot as in (3) kinematics, (8) Controls Part 2) where the link lengths are $a_1 = 0.2, a_2 = 0.15, a_3 = 0.12$. The workspace of the manipulator is within a circle whose radius is 0.47 (0.2+0.15+0.12) before becoming the velocity singularity. For each of the end-point positions given in the table, determine joint angles so that the manipulability measure becomes the largest (i.e., optimal in terms of singularity avoidance).



Position (x,y)	Joint 1 (deg)	Joint 2 (deg)	Joint 3 (deg)	Manipulability w
(0.1,0)				
(0.2,0)				
(0.3,0)				
(0.4,0)				
(0.45,0)				

This problem can be done fully numerically. While you don't need to give a proof of optimality, you must show the process how your approach/algorithm is expected to maximize the manipulability measure.

Report (for each of p2):

1) Explanation of your approach to determine redundance joint angles
2) Joint angles and manipulability measure for each end-point positions
3) Screenshot of each posture (preferred)

Grading: Total 30 points

Documentation and submission: Create a SINGLE WORD or pdf file containing all descriptions, derivations, plots, and codes, and upload it to Canvas. Separately upload video files to Canvas. You should provide sufficient details of your approach and results (MATLAB codes, plots, tables with numbers, etc.) to justify your work. Answers without justification will not receive points. Note that graders are not always able to run your scripts to reproduce results if you don't show necessary numbers. You must document any assistance that you received from any person or any reference (see the syllabus for documentation).

Students who will receive TA awards (+10 points) may be asked to provide full MATLAB scripts and SimScape models for future sections.

EOF