Homework 4

Problem 1 – Due 4pm Monday, Nov. 19. (Other problems due Tuesday, Nov. 20.)

To prepare for Lab 4, use your code from Prelab 3 and Lab 3, and design tractories for each of the three wheels over time to travel as required by the caption in Figure 1. Come up with at least 2 different solution trajectories (e.g., not necessarily an ellipse as the path). This is challenging because the robot must always point a laser pointer at a fixed target. You may refer to code from last year's "Lab 5" (which is Lab 4 this year) to help get started, but YOU MUST WRITE YOUR OWN MATLAB CODE. Simply copying the code from last year (or from a friend) is not allowed.

As in Lab 3, you must first calculate x, y, and phi trajectories for the robot over time. Then, approximate the derivatives, dx/dt, dy/dt, and dphi/dt over time. Use your Jacobian relationships to determine each of the three wheel velocities, and (finally) integrate these to create desired wheel trajectories, as functions of time. Turn in both all of your MATLAB code (including electronic copies, as requested by the TAs) and plots of trajectories.

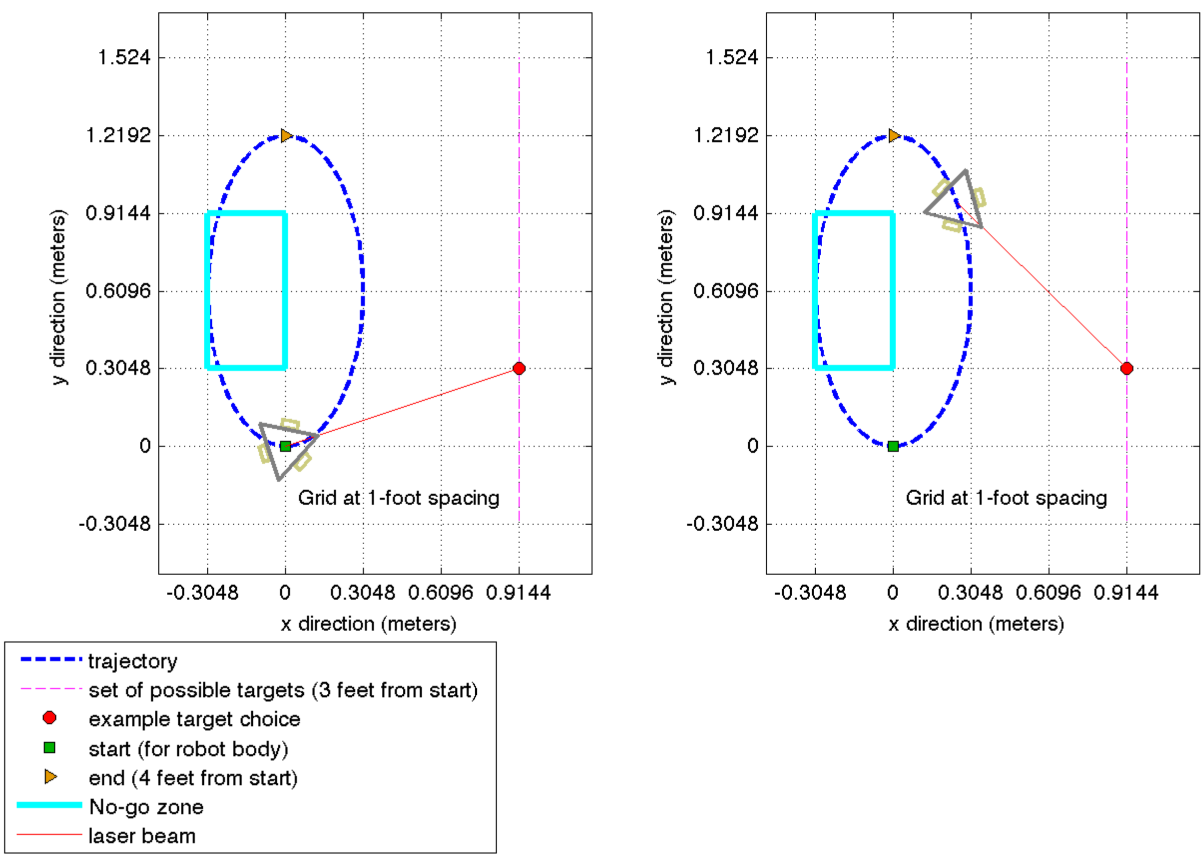


Figure 1. Frames from an "example trajectory" for the omnibot in Lab 4. The omnibot center of body must start and stop on the desired locations, four feet apart from one another. All parts of the robot must avoid the "no-go zone" shown. An ellipse is one solution; there are many other options! The omnibot must also aim a laser a fixed target location ANYWHERE along the dashed line 3-feet away from the start-to-end line, as shown. The laser is mounted as shown. The entire motion of the omnibot must take no more than 60 seconds, start to finish.

Problem 2 – Two-cart system.

You must turn in a copy of your MATLAB code for the parts which require MATLAB calculations and/or MATLAB plotting, below. As always, you must do your own work in writing this code. (Copying a solution someone else has written is of course not allowed.)

a) Using the MATLAB code template from Lecture 13, derive the equations of motion for the system shown in the schematic in Figure 2, below. You will need to download the function "fulldiff.m" from the homework website to complete this.
b) Write the equations of motion in "state space" format, using $X = [x_1 \quad x_2 \quad \dot{x}_1 \quad \dot{x}_2]^T$.
c) Using your equations from part a, solve for the four transfer functions, A(s) through D(s), shown at right in Figure 2. (Notice each block contains either a mechanical impedance or an inverse impedance.) Also determine the signs (+ or -) at each summing junction.
d) Solve symbolically BY HAND for the transfer function from F1 to X1, X1(s)/F1(s).
e) Solve symbolically BY HAND for the transfer function from F1 to X2, X2(s)/F1(s).
f) Now, use the values below in your solution within MATLAB to create Bode plots for your transfer functions from c and d, on the SAME SET OF AXES (clearly labeled).

```
m1 = 20; m2 = 1; k1 = 1200; k2 = 400; b2=20; % SI units for all
```

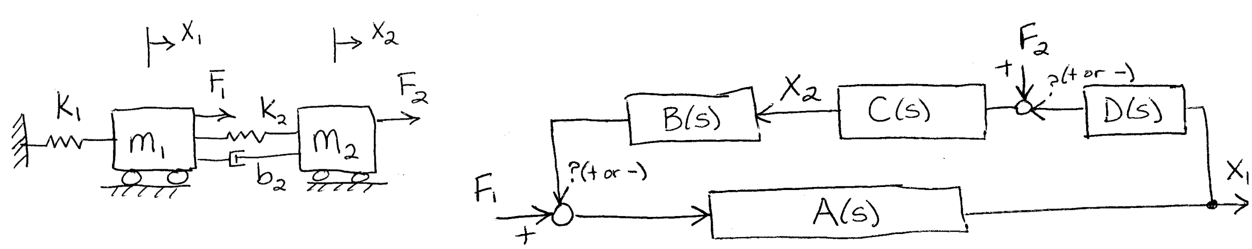


Figure 2. Schematic (left) and Block Diagram (right) for Problem 2.

g) Now, repeat c and d for the transfer functions from F2 to X1 and to X2 (instead of from F1). Use the same values shown in e, and again put both Bode plots (clearly labeled) on one set of axes (different from the axes used in c).

h) List poles and zeros for each of the 4 transfer functions.

Note: *Poles should simply be the SAME for all four transfer functions, since they are characteristic of the entire system. (Verify this, briefly.) For X1(s)/F1(s), the zeros correspond to poles of a system in which you hold m1 in place and then allow m2 to oscillate freely. For X2(s)/F2(s), the zeros correspond to poles for a system where you clamp m2 in place and then allow m1 to oscillate freely. Control for these two systems is known as a "collocated" feedback problem, because the actuation is applied ("located") in the same place sensing occurs: one feeds back the position of the mass to which force is applied.*

i) From your plots, comments on the relative stability and steady state error you would expect if you used a proportional controller with Kp=2,000 for each of the four transfer functions.

Problem 3 – Two-mass pulley system.

a) As in the previous problem, use MATLAB to derive the equations of motion for the system below, symbolically. The circles are a pulley with inertia J, such that x2=R2*theta and x1=R1*theta of the pulley, as it turns. Use x1 and x2 as generalized variables. Notes that gravity is present here, as well.
b) For the block diagram in Figure 3, as done in Problem 2b, solve for the missing transfer functions and the sign at the summing junction.
c) Solve symbolically BY HAND for the transfer function from F to X1, X1(s)/F(s).
d) Solve symbolically BY HAND for the transfer function from F to X2, X2(s)/F(s).
e) Now, use the values below in your solution to SOLVE FOR THE POLES AND ZEROS for your transfer functions from c and d. What pole(s) will dominate the response? Use MATLAB to calculate a 5-second step response for each transfer function, to verify your answer. Use only 5 seconds, so the transient response can be seen clearly.

```
m1=3.36; m2=1; J=0.1; k=686; b=24.5; R2=0.2; R1=0.5;  % SI units
```

f) Imagine m1 is no longer present and that x1 is the input to the system. Using your previous work, what are the natural frequency and damping ratio of the transfer function from X1 to X2, X2(s)/X1(s)? (Hint, you should get "nice" numbers…)

g) Now, instead imagine k is replaced with a solid cable (infinite stiffness), so the system has only one degree of freedom, instead of two. What is the transfer from F to X2, X2(s)/F(s), now? (Please do not use MATLAB to calculate the equation of motion. Instead calculate the total reflected inertia. This is just a first-order system. You should get "nice" numbers…)

HINT: Compare your answer in (g) with the answer in (e). How does a step response for (g) compare with (e)?

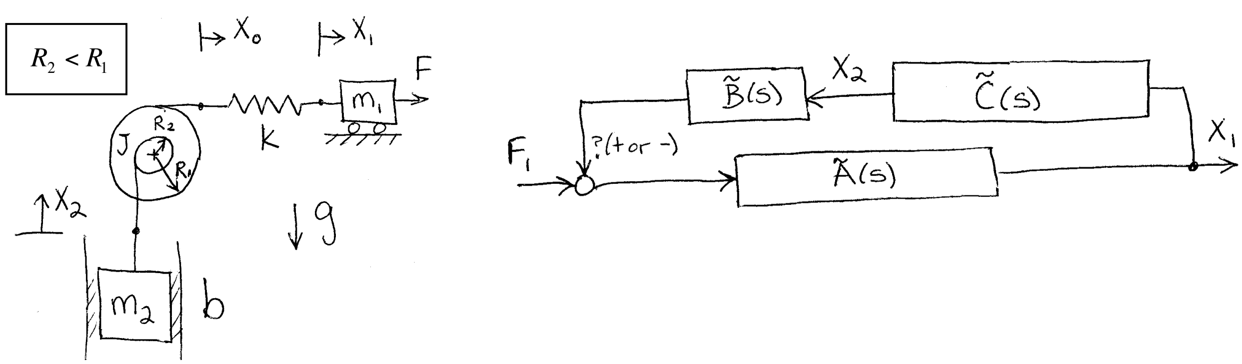
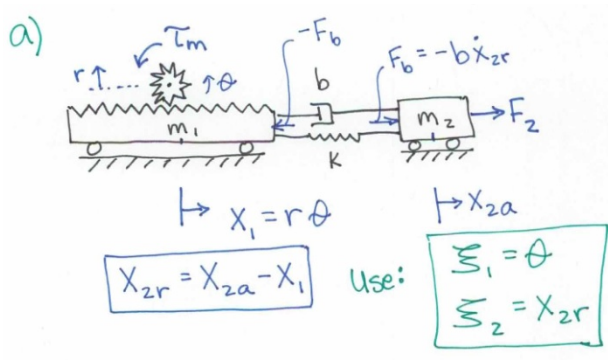
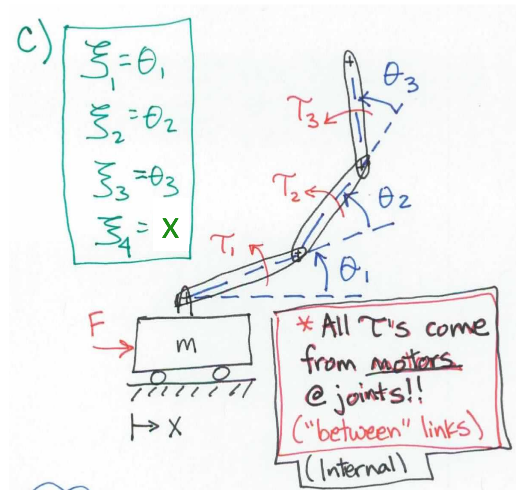
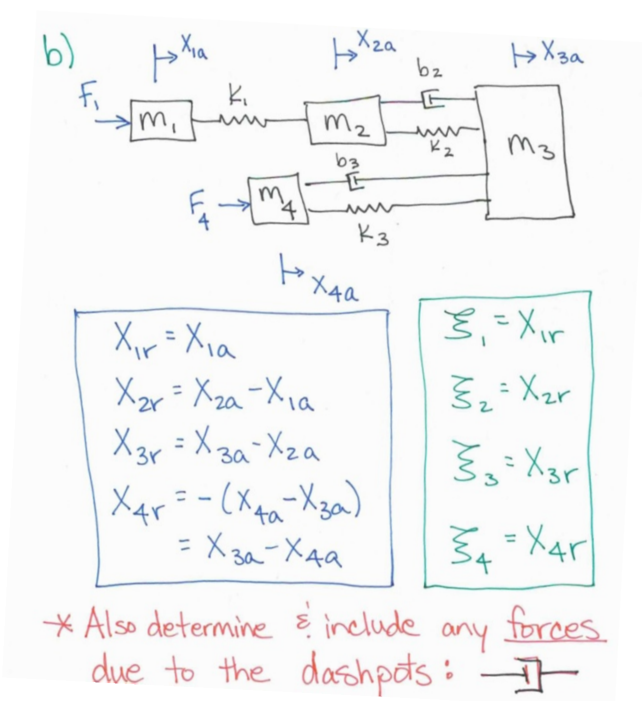


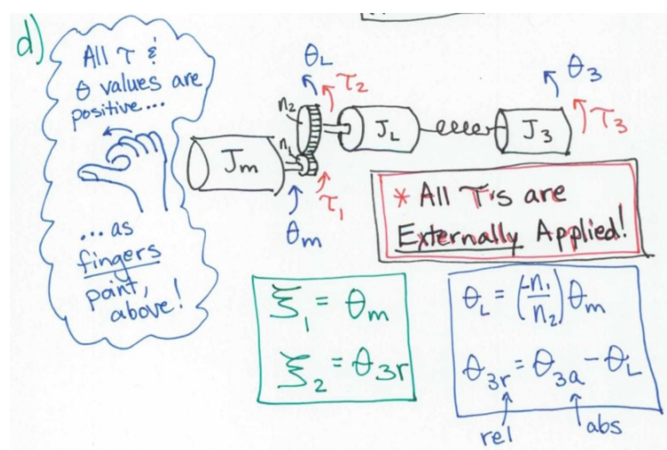
Figure 3. Schematic (left) and Block Diagram (right) for Problem 3. Note: $R_2 < R_1$.

Problem 4 – For each system and corresponding definition of generalized coordinates (GC's), identify the non-conservative forces ("big Xi", Ξi) associated with each GC, necessary in using the Lagrangian approach to develop equations of motion.









In **a**: non-conservative forces and torques include:

tau_m, F2, and Fb (due to dashpot, applied as equal and opposite forces on each mass).

In **b**: non-conservative forces and torques include:

tau1, tau2, tau3, and F.

In **c**: non-conservative forces and torques include:

F1, F4, and any damping forces (similar to part a) due to both b2 and b3.

In **d**: non-conservative forces and torques include:

tau1, tau2, and tau3. n1 and n2 are the numbers of teeth on the gears, as shown.

Note, when theta_m velocity is positive, theta_L velocity is negative!

Problem 5 – For the two-link system shown,

a) Write the Jacobian, *J*, and Jacobian transpose, J^T.

Assume we will use *relative* generalized coordinates, as shown, where theta1 is absolute but theta2 is relative to theta1.

b) Write the non-conservative forces, Ξ1 and Ξ2. Each should involve a mathematical expression that includes some subset of actuator torques, *tau1* and *tau2*, and the externally-applied "disturbance" forces, F_x and *Fy*, at the end effector.

c) Comment on how the Jacobian (or its transpose) describes the effect the disturbance forces will have on each Ξi, the total non-conservative torque affecting each equation of motion.