ECE4560 - Homework #7

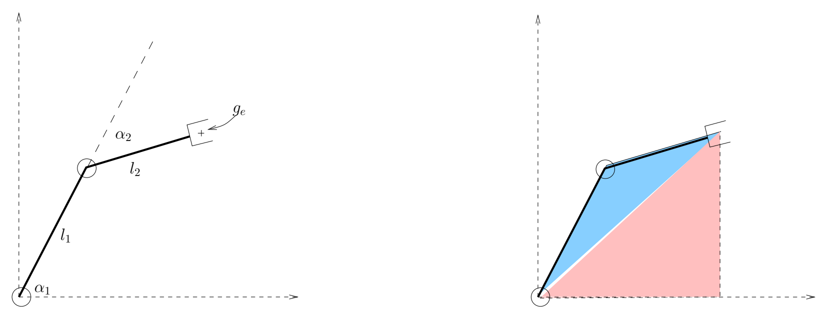
Due: Oct. 16 2017

Problem 1. (10 pts) The *inverse kinematics* of a manipulator are the equations that provide the manipulator joint angles given the end effector configuration. In this simple problem, let's consider that our only interest is in the position of the manipulator end-effector (orientation is not important) for the manipulator in Figure 1

(a) What are the forward kinematics of the planar manipulator? What is $d_e(\alpha) = (x_e(\alpha), y_e(\alpha))^T$.

(b) The inverse kinematics for this can be solved using the law of cosines and properties of right triangles, as hinted at in Figure 1(b). Suppose that $l_1 = 1$ and $l_2 = 0.5$, solve for the inverse kinematics given that $d_e = (1.3595, 0.2113)^T$. The inverse kinematics essentially asks: given the end-effector configuration of $(x_e, y_e)^T$, what are α_1 and α_2?

(c) When you solve the above problem, there will be cases where the computation cannot be performed. What cases of (x, y) cause the solution to not exist?



(a) Manipulator Geometry. (b) Geometry of Inverse Kinematics.

Figure 1: Simple planar manipulator with two revolute joints.

Problem 2. [10 pts] Using the $SE(2)$ class exponent and logarithm as example stubs, code in the exponent and logarithm for the $SE(3)$ class (they should be static member functions). Verify your solution by exponentiating the following twist

$$\xi = (2, 1, -3, \pi/10, -\pi/4, \pi/9)$$

for 2 seconds, and then taking the logarithm of the result. You should get the same thing back out. The group element you get should be

```
>> g = SE3.exp([2; 1; -3; pi/10; -pi/4; pi/9], 2)

ans =

   -0.1084   -0.7389   -0.6650    4.6977
   -0.0015    0.6691   -0.7432    4.5839
    0.9941   -0.0795   -0.0737   -0.8141
         0         0         0    1.0000

>> log(ans, 2)

ans =

    2.0000
```

```
 1.0000
-3.0000
 0.3142
-0.7854
 0.3491
```

You can also do this by hand as a test of your abilities to actually compute the exponent and logarithm.

Problem 3. (30 pts) From the piktul manipulator thread, work out Module 1, Adventure 2. AT this point, it is just pencil and paper. The following week will be the manipulator implementation. It would probably be best to actually code the answer up in the inverse kinematics member function (which may be called `invkin` or something like that, within the `piktul` class).

Provide the joint angles associated to the following two $SE(2)$ configurations:

$$\left[\begin{array}{cc|c} 0.707 & 0.707 & 3.165 \\ -0.707 & 0.707 & -7.811 \\ \hline \multicolumn{2}{c|}{0} & 1 \end{array}\right] \quad \text{and} \quad \left[\begin{array}{cc|c} 0.259 & -0.966 & 7.328 \\ 0.966 & 0.259 & 2.828 \\ \hline \multicolumn{2}{c|}{0} & 1 \end{array}\right] \tag{1}$$

with the respective heights, 0.5 and 0.25. You can always test the output by applying your forward kinematics routine to the found joint angles.

Problem 3. (30 pts) For the turtlebot groups, work out the first two enumerated items in the "Turtlebot: Sensing Part 1" module. Turn in the finite state machine associated to the bumper logic, as well as the code associated to the finite state machine. Naturally, you should also demo the final product.

Problem 3. (30 pts) For the biped track, work out Module Set 1, Part 2. The adventure explores the center of mass calculations for the robot. These will be important because the center of mass plays a big role is assessing the static stability (e.g. balance) of the robot. Ideally, the center of mass is always over the support region defined by the feet in contact with the ground. Submission details will be at the wiki page.

Problem 3. (30 pts)**.** If you are working on a group project, then your group contact will review the submissions associated to the prior tasks and provide a new set of tasks. You should work out whatever baby steps this group contact assigns the group as opposed to the lab problem. It is your responsibility to turn in the results to the group contact. There should be sufficient output for them to assess progress.