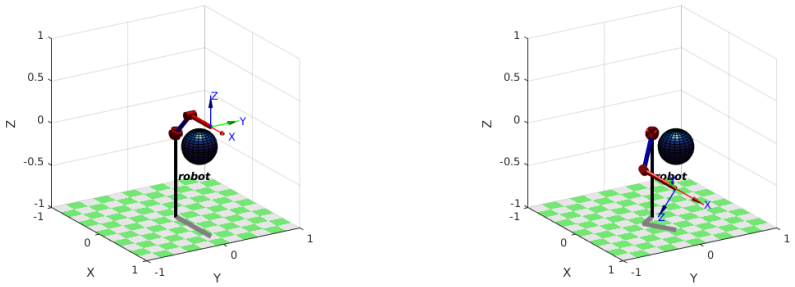
CS4610/CS5335: Homework 2

Out: 10/18/17, Due: 10/24/17

Please turn in this homework to Rob Platt via email on the due date. HW PA Q1, Q2, and Q3 should be submitted in the form of a set of three files named Q1.m, Q2.m, Q3.m. All this should be zipped up into a single file and emailed to me.

Have a look at the accompanying zip file. Stub files for the questions are provided to you. You should implement each of these. Once implemented, you should be able to run "hw2(X)" in order to run code for question "X". hw2.m is given to you and should not need to be modified. The only thing you need to do is to insert code into the stub functions in Q1.m, Q2.m, and Q3.m.



(a) (b)

Figure 1: Illustration of Q2. (a) initial arm configuration. (b) end effector in desired position.

PA Q1: Write a function that evaluates whether a straight-line path in configuration space between $q1$ and $q2$ is free or not. You should evaluate the line at 10 evenly spaced points between $q1$ and $q2$ in c-space. Your function will take as input a robot (encoded as a SerialLink class). You should use *robotCollision* (that function should be in the code provided for this homework) in your implementation. You will use this function in Q2 and Q3.

PA Q2: In this question, you're going to use an RRT to find a collision free path from a given start configuration to a goal configuration where the end effector is at a desired position. As in Q1, your function will take as input a robot (encoded as a SerialLink class), a starting arm configuration (encoded as a 1x4 vector of joint angles), and a desired position (encoded as a 3x1 vector). The output should be a series of milestones that achieve a collision free path. This function should work for arbitrary desired positions. You should create a single RRT rooted at the start. Figure 1 shows the starting configuration and a goal configuration on a path found by my code. You should use this command to solve for the IK solution at the goal: ROB.IKINE(TRANSL(XGOAL),ZEROS(1,4),[1,1,1,0,0,0]);.

PA Q3: Write a function that does trajecotry smoothing (as covered in the RRT slides) on an arbitrary trajectory. The input should be an $n \times 4$

sequence of milestones. The output should be an $m \times 4$, $m \leq n$, smoothed sequence of milestones.