CS545–Introduction to Robotics

Homework Assignment 3 (Due April 17)

In the following problems, you will need the NAO simulator and Matlab for CLMCPLOT for other visualizations. You find relevant files and information under:

http://www-clmc.usc.edu/Teaching/TeachingIntroductionToRoboticsHomework

IMPORTANT: In your solutions of the homework, also provide intermediate steps how you derived the solution to a problem.

1. (125 Points) In this homework, you are supposed to create a complete point-to-point reaching behavior for the NAO robot. First, you will need to create a planning system for point-to-point movements for the endeffector (right hand) of the robot. Second, you will need to implement the plan on the robot simulator and visualize the outcome to confirm that it works.

a) For planning, you are to create a 5th order spline system (a "minimum jerk spline"). The general 5th order spline equation is:

$$x(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5$$

Assume the movement duration is τ, and the movement starts at t=0. Assume that at start the movement is at x_0, and it is supposed to reach x_f at time t=τ. Determine all the constants c_0 to c_5 as a function of x_0, x_f, and their derivatives, and τ. At start and end, velocities and accelerations are $\dot{x}(0) = \dot{x}_0, \dot{x}(\tau) = \dot{x}_f, \ddot{x}(0) = \ddot{x}_0, \ddot{x}(\tau) = \ddot{x}_f$.

b) Using matlab, implement a planning system that creates a trajectory of 2 seconds duration, starting at position x=0 and going to a target at x=2, with zero velocity and acceleration boundary conditions. Use a time step of 0.01s to compute the movement plan. Provide a print-out of your matlab code, and plots of position, velocity, and acceleration for the entire trajectory.

c) A useful way to implement the minimum jerk planning system is by creating a function that takes as input variables the current state $x(t), \dot{x}(t), \ddot{x}(t)$, the remaining time to go τ_{togo}, and the target state $x_f, \dot{x}_f, \ddot{x}_f$. The output of the function would be $x(t+\Delta t), \dot{x}(t+\Delta t), \ddot{x}(t+\Delta t)$, i.e., the planned state one time increment ahead. This function can be used to plan the next desired state given the current desired state, which is useful in a control loop of a robot. Note that the time-to-go τ_{togo} needs to be decreased every iteration of the control loop by Δt. Implement this function in matlab, and create the results of b) using this incremental planning system. Compare the results of b) and c) and comment on the differences and similarities.

d) The webpage has a file min_jerk_task.cpp. Modify this file to implement your min jerk function from matlab. Run the task using the "setTask" command in the blue task_servo window. You will automatically collect a data file, and you should save it to disk with "saveData". Visualize the trajectories of all 5 right arm DOFs in CLMCPLOT in matlab, i.e., print the data traces R_SFE_th and R_SFE_des_th in the top most chart of clmcplot, and then the data traces for R_SAA, R_HR, R_EB, and R_WR in the next 4 charts. Create a print-out of your visualization and include it in your homework. Comment on the quality of tracking for the 5 DOFs. Repeat the same for the plot, this time using joint velocities, and then another plot using joint accelerations.

e) Add three additional targets to you C-program such that the right arm creates approximately a square in Cartesian space. Note that due to the joint-space movements, the hand movement in Cartesian space is curved. Your square should end at the same point where it started. Collect data of your movement and save it such that it can be visualized with CLMCPLOT. In clmcplot, click the "PhasePlot" button, and then click RIGHT_HAND_x and then RIGHT_HAND_y. This creates a window that plot RIGHT_HAND_x in the horizontal axis, and RIGHT_HAND_y in the vertical axis. Provide this plot in your homework and comment on what is good or bad about your realization of the square. Repeat this plot for RIGHT_HAND_x and RIGHT_HAND_z and provide this print-out as well. Comment on the quality of your square from this view.