

Assignment 1, EECS 397/600: DARPA Robotics Challenge

due before class, Tuesday, 9 September

On the lab computer, login as guest. Once you have completed this assignment, save your work somewhere (off the computer), then delete your work from the system. (Soon, you'll get your own account and this will not be necessary).

1) Test the minimal simulator and minimal controller:

In the `ros_workspace/hku_drc_class`, subdirectory “`catkin/src`”, create a new package called “`minimal_nodes`” (following the class notes). Using the code described in the class notes, create the source files `minimal_simulator.cpp` and `minimal_controller.cpp`. Edit `CMakeLists.txt` to add these two executables, then compile them with: `hku_make -d minimal_nodes`. Fix any typographical errors until these compile cleanly.

Run both of these nodes, together with `rqt_plot`. Display desired velocity, actual velocity and force stimulus, as per the figure on pg 28 of the notes. Stimulate the input to the controller via “`rostopic pub`”, as described on pg 27. Halt `rqt_plot` (using the “pause” button) and capture a transient (like that in the notes). Save a screen capture of this plot.

One way to get a screen capture is to use the “Gimp image editor”, which you can find under “applications/graphics”. With GIMP running, in the GNU Image Manip window, select: `file->create->screenshot...` In the “screenshot” pop-up window, click “snap”, then click on the window (`rqt_plot`) that you want to capture. The captured window will display. Then select `file->save as...` and choose a file name and folder. Name the file `fname.jpg`, and it will be stored in `jpg` format. Include this screen capture with your assignment upload.

2) Design a new node:

Create a new node (in the “`src`” directory) called “`minimal_commander.cpp`”. This routine should start up by prompting the user for an amplitude and a frequency, then it should compute a sinusoidal output of the specified amplitude and frequency. It should publish the resulting output to the topic “`vel_cmd`”. Your source code must include comments explaining the intent.

Edit the `CMakeLists.txt` file to include compilation of this new node in your modified package.

Create a launch file to launch all three of these nodes (simulator, controller, commander). Invoke your launch file to start up all 3 nodes. (Don't forget—you will need to have “`roscore`” running in some window).

Enter amplitude and frequency values in response to your program's prompts.

Run `rqt_plot`, as before, and capture a system response.

3) Submit assignment:

Using Blackboard, submit your (commented!) `minimal_commander.cpp` source code and your launch file. Also submit your `rqt_plot` screen captures. Comment on the performance. What would you say is the maximum frequency that your system can follow? (yes, this is subjective).