Supplementary Material - Getting Started With the First Programming Assignment

Presented by Lucy Tang, PhD student.

This MATLAB demonstration will get you started with working on the first programming assignment using our simulation environment.

The first thing you have to do is change the current folder that MATLAB is working in to the one containing the project files. Next, open the file ***runsim***. You can do this by double clicking the file name in the current folder window or by using the **open** command in the command window.

```
>> open runsim.m
```

To run this file, you can type **runsim** in the command window.

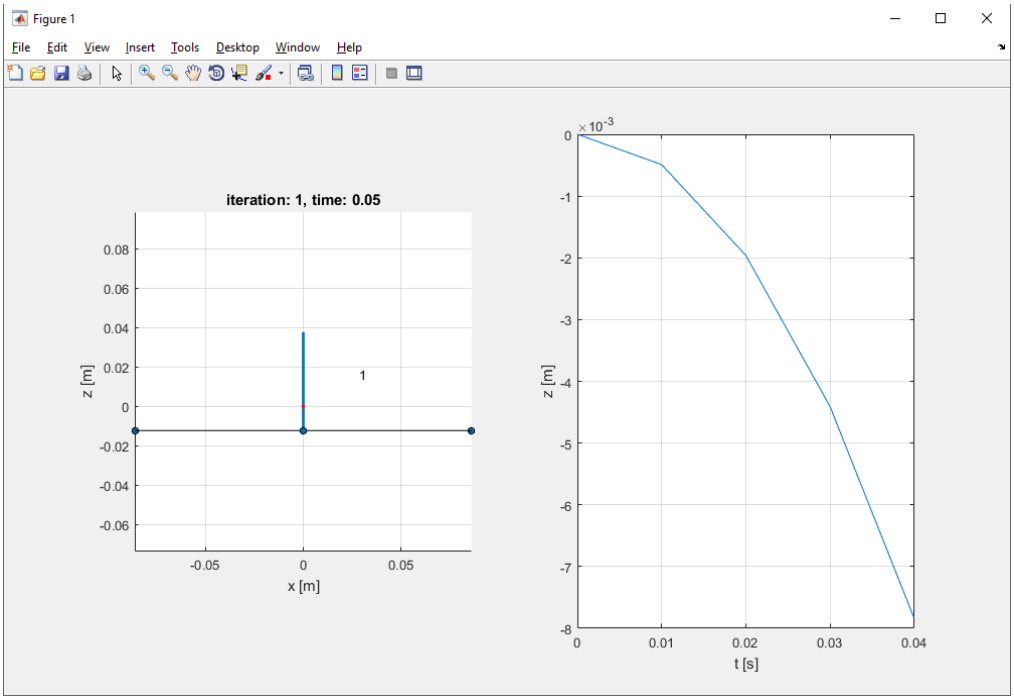
```
>> runsim
```

You can exit execution of this file prematurely by pressing Ctrl+C.

You can use the command **clc** to clear the command window.

An alternate way to run the program is to click the green run button in the editor:

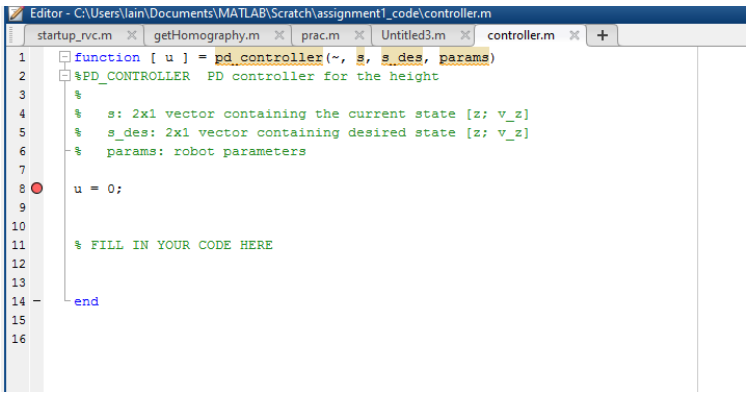
When you run the code, you will see an animation quadrotor on the left and a plot of the height over time on the right. Since initially there is no controller running, you should see a quadrotor falling straight downwards:



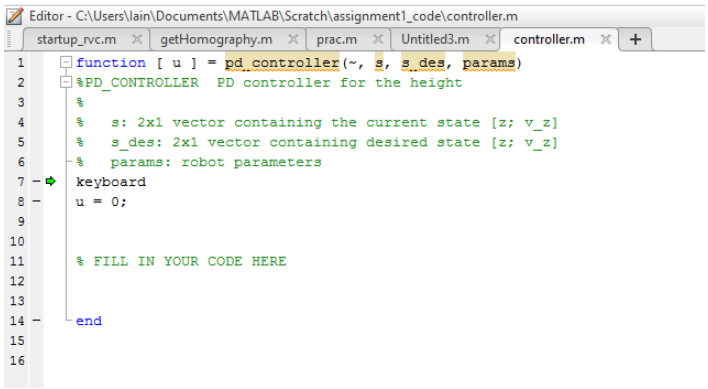
The plot on the right similarly shows that the height Z is decreasing as time increases. Make sure that this code can run successfully.

To complete assignment one, you have to modify the function `controller.m`. When we open the file, we see that the input, u, is currently zero. That means, we're applying no thrust to the quadrotor. To pass the assignment, you need to calculate an appropriate value for u that allows the quadrotor to control its height using the given input parameters.

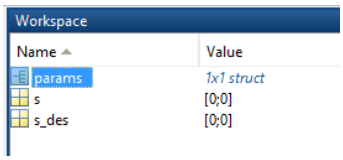
A useful way to debug your code is to use **Breakpoints**. You can set a breakpoint by clicking the line in the code where you want the execution to stop. A red dot will appear indicating the presence of the breakpoint:



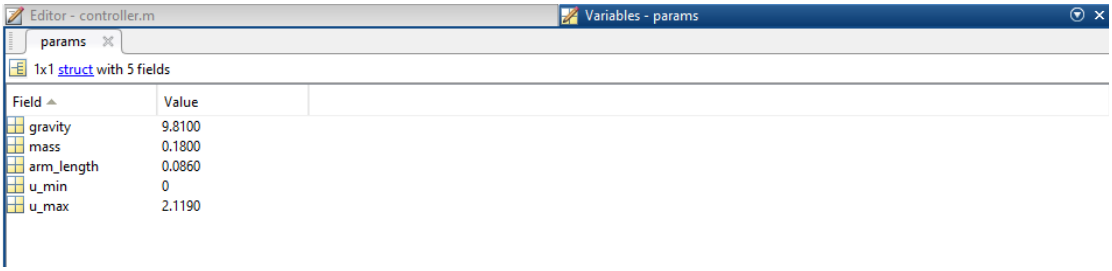
Now, when the function runsim is run the execution will stop at the breakpoint. Another way to do this is to use the **keyboard** command. Just type in the command **keyboard** before the line of code at which we want execution to stop. Again, when we run the code, it stops at the designated line:



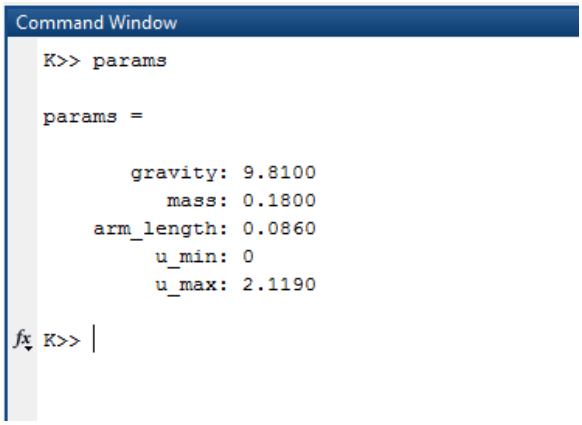
Now, we can explore what variables are currently defined at this point in the code. To find out the value of a variable, we can double click its name in the workspace:



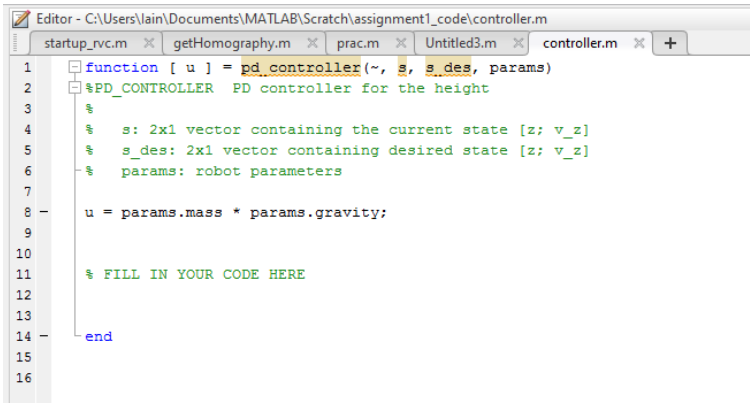
This displays the values:



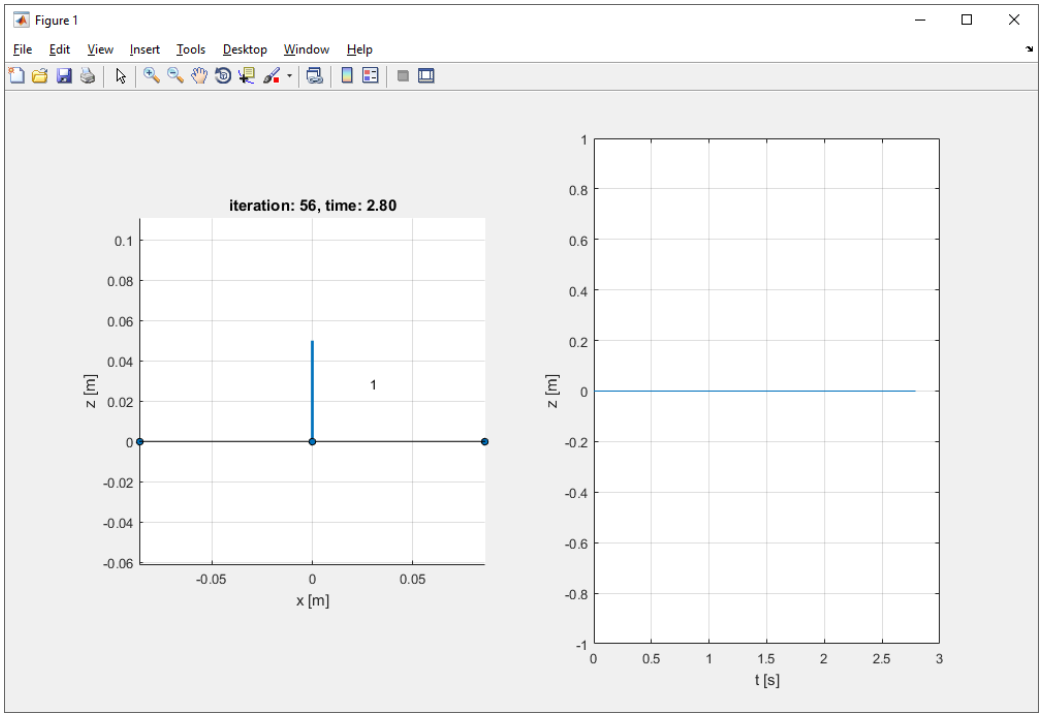
Alternatively, we can just type the variable name in the command window. Here, we see that the parameters that have been defined in the `params` variable are the mass of the quadrotor, the gravity constant and a few other constants:



If we go to the editor and set the control input, u, exactly equal to the weight of the quadrotor, mg:



and run the simulator:



we see that the robot's height remains constant. This is because the controller is applying enough just enough thrust to cancel out the force of gravity, allowing the robot to hover in place. (Don't forget to remove any break points that you set when you finish debugging your code).

To complete the remainder of the assignment, you'll need to follow the instructions in the assignment instructions pdf file.