MATLAB Template Code Walkthrough

MMAN1300 : Lab Report 2 | Written by Daniel Wong

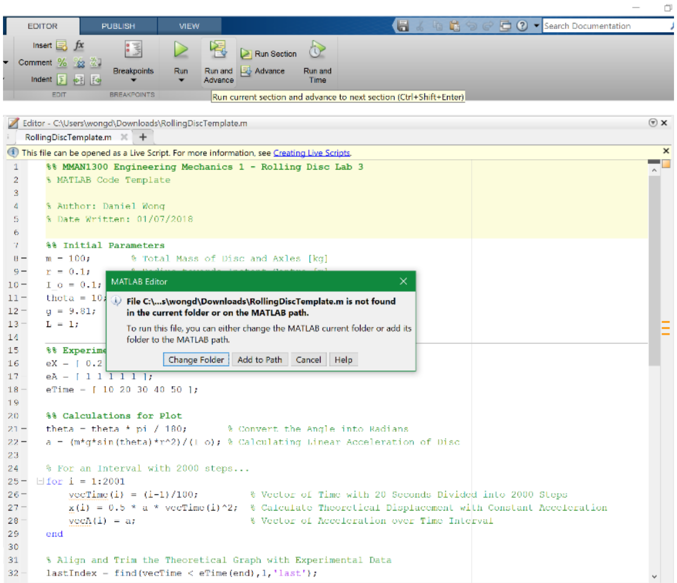
This is a brief guide into how to use MATLAB and program your own calculations and plots with the template provided.

Obtaining MATLAB

All UNSW students are able to acquire a free, student license. Simply head to this website and follow the instructions. Make sure during installation that you include the 'Curve Fitting Toolbox' as a package.

Setting up your Program

Once installed, you can open up the template code. Give the program a test run by clicking 'Run' in the top menu. It'll ask you to add the file to the directory for MATLAB. Click 'add to path' and the code should run.



Now let's see how to modify and write new code.

Initial Parameters

Here, we setup any constants for pending calculations. We are storing these numeric values inside variables, which we can call and reference by their variable name. Change these values according to your calculations and the apparatus.

```
%% Initial Parameters
m = 100;          % Total Mass of Disc and Axles [kg]
r = 0.1;          % Radius towards Instant Centre [m]
I_o = 0.1;        % Moment of Inertia about Instant Centre [mm^4]
theta = 10;       % Slope Angle of Ramp [Degrees]
g = 9.81;         % Acceleration due to Gravity [ms^-2]
L = 1;            % Length of Angled Slope [m]
```

Experimental Data

We can store our data in multiple ways. For small amounts of easy-to-write data, we can create variables to store arrays of values. Change the values here according to what you have recorded. These can be plotted.

```
%% Experimental Data at Recorded Intervals
eX = [ 0.2, 0.4, 0.6, 0.8, 1.0 ]; % Experimental Displacement
eA = [ 1, 1, 1, 1, 1 ];           % Experimental Acceleration
eTime = [ 10, 20, 30, 40, 50 ];   % Measured Time
```

Calculations for Plot

This section performs the calculations for the angle, acceleration and plotting values.

1. The first section calculates the acceleration of the disc.
2. The second section creates a time vector consisting of 20 seconds split over 2000 intervals. The displacement is then calculated.
3. The third section limits the theoretical displacement and acceleration values based on the limits of your experimental time measurements. This ensures that both graphs are plotted on the same scale.

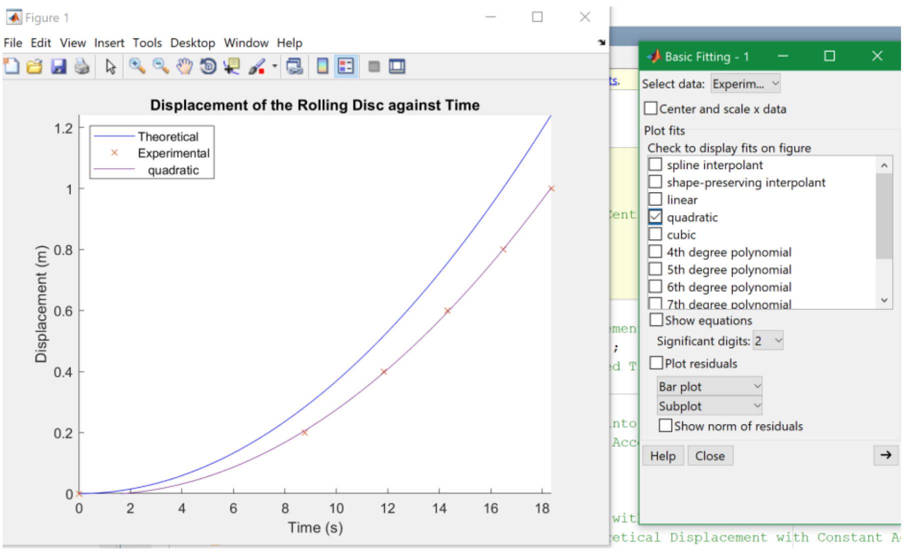
```
%% Calculations for Plot
a = (m*g*sind(theta)*r^2)/(I_o); % Calculating Linear Acceleration of Disc

vecTime = linspace(0,20,2000);  % Vector of Time with 20 Seconds Divided into 2000 Steps
x = 0.5 * a .* vecTime .^ 2; % Calculate Theoretical Displacement with Constant Acceleration

% Align and Trim the Theoretical Graph with Experimental Data
lastIndex = find(vecTime < eTime(end),1,'last');
vecTime = vecTime(1:lastIndex);
x = x(1:lastIndex);
```

Plotting Experimental and Theoretical Graphs for Displacement

This section plots the graphs for your report. Using the *plot* command, we can set a particular style for lines, colour and whether to include x's in our graph. We can also set axis limits, create a legend and labels for axes.

```
%% Plotting the Graph of Displacement against Time
figure(1);
hold on;
% Plotting Curves
plot_tX = plot(vecTime, x, 'Color', [0 0 1], 'LineStyle', '-');   % Plotting Theoretical Curve
% Note that with Experimental Data, you can use curve fitting tools to plot
% lines of best fit within the graph viewer.
plot_eX = plot([0 eTime],[ 0 eX] , 'x'); % Plotting Experimental Data
axis( [0 eTime(end) 0 x(end)] );    % Defining Axes Boundaries
lgd1 = legend([plot_tX, plot_eX],{'Theoretical', 'Experimental'}, 'Location','NorthWest');    % Legend
% Graph Labels
xlabel('Time (s)');
ylabel('Displacement (m)');
title('Displacement of the Rolling Disc against Time');
```

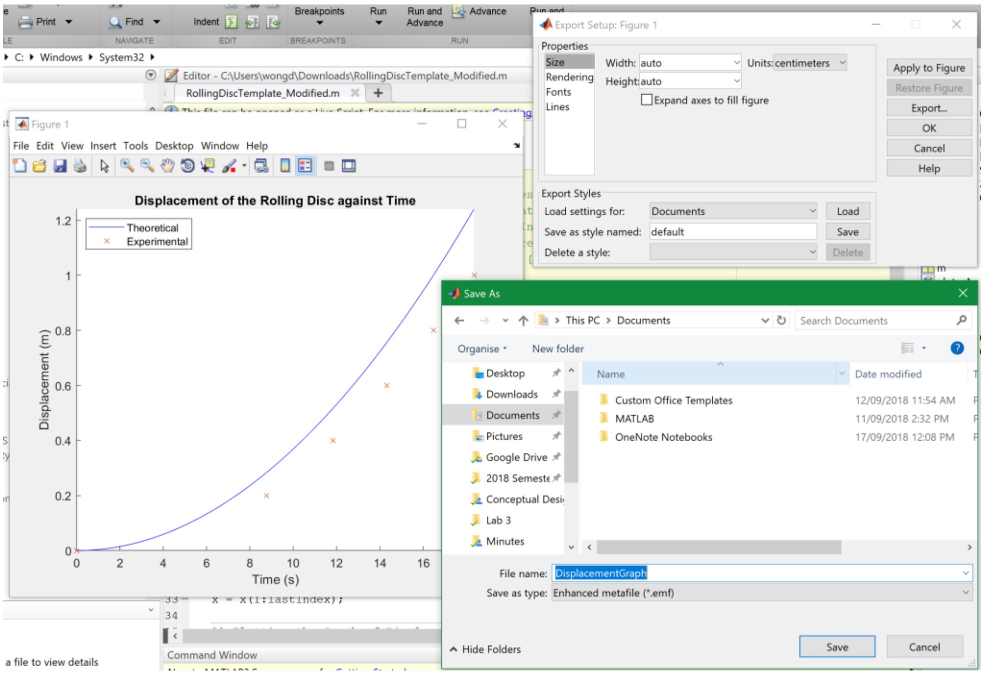
Once you hit 'run', a graph of the plot should appear. Let's learn how to add a line of best fit. Click [**Tools -> Basic Fitting**] and select a type of fitting curve for your experimental data.



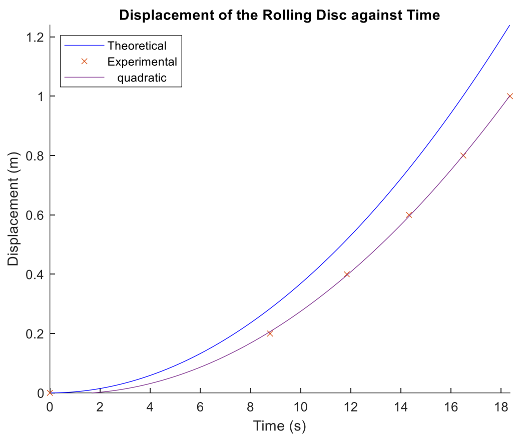
This is what you should see. A line of best fit for your experimental data plotted in conjunction with a theoretical plot.

Exporting the Graphs into your Report

You can export your graphs in high quality without needing a screenshotting tool. Inside the graph view, go to [**File -> Export Setup… -> Export…**]



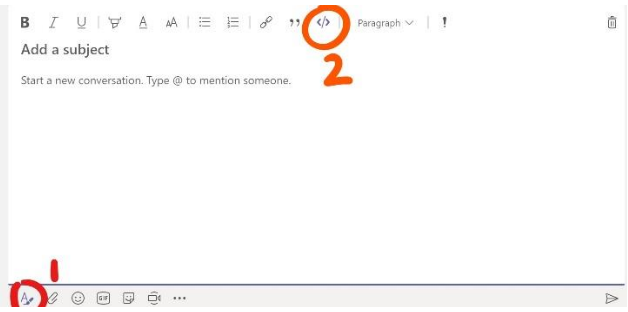
Save your file as an 'Enhnaced metafile (*.emf) and insert into your word document. This is the result you should get.

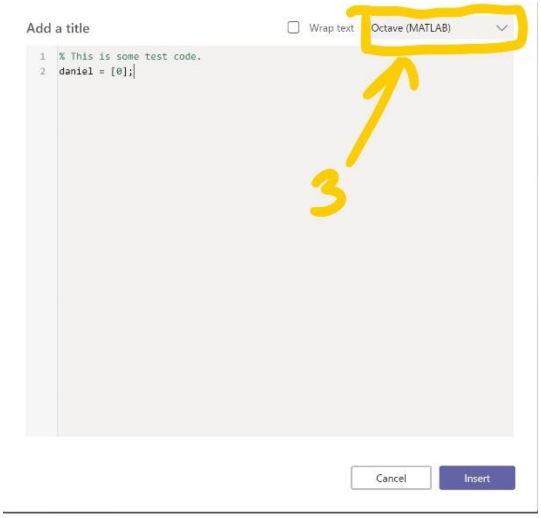


Seeking Assistance

Keep in mind that any errors or new things you want to learn, MATLAB has a large online database of documentation for getting guides on how to perform these commands. Please reference to them before asking around in the forums.

If you do wish to post to Teams, it has an inbuilt space for entering code in a separately formatted segment. Simply click around the bottom textbox [**Format -> Code Snippet -> Select Octave (MATLAB)**] as your code format.





Conclusion

Hopefully this brief guide has been useful towards teaching you the basics of MATLAB. Use the template as a guide for making beautiful graphs for this report, and hopefully future reports as well!

Best of luck with your reports. 😊
~ Daniel Wong