‘Mini’ Project #4

PLOTTING AND GRAPHS IN MATLAB FOR STEM APPLICATIONS

IDS 1380 INTRODUCTION TO STEM

FALL 2021

Overview: Mini Project #4

- **Why plot your data?**
  - "STEM professionals use graphing techniques to make information easier to understand. Graphs can also be used as a quick check to determine whether a computer solution is yielding expected results."[1]

- One of many 'helpful' references from the web
- Watch "How to Use Basic Plotting Functions" with link below
  - https://www.mathworks.com/videos/using-basic-plotting-functions-69018.html
- *Lots of plots!*
  - https://www.mathworks.com/products/matlab/plot-gallery.html
  - And https://www.mathworks.com/support/search.html?fq=asset_type_name:video%20category:matlab/2-and-3d-plots&page=1

- For this project, you will learn:
  - How to create simple x-y plots
  - Use MATLAB basic plotting functions
  - Solving Problems with MATLAB
    - MATLAB skills applicable to HW problems
      - Graphs

Deliverables for Mini-Project #4

Due: Wednesday, December 8, 2021, last day of class ☺

Work in again in *different* Peer Partnerships For Programming Feedback, P^{3}F

- Collaborate in groups of 3 to help each other with programming issues , syntax checks, etc., as needed
- Groups randomly assigned through Canvas

Mini-project activities – 5 parts

1- **Basic Plotting Skills** : Creating STEM graphs to 'spec' – Syntax and operations

2- **Practice More Plotting Skills with *Special* Format in MATLAB** – examples

3- **STEM application –Projectile motion & diver launching off a springboard**

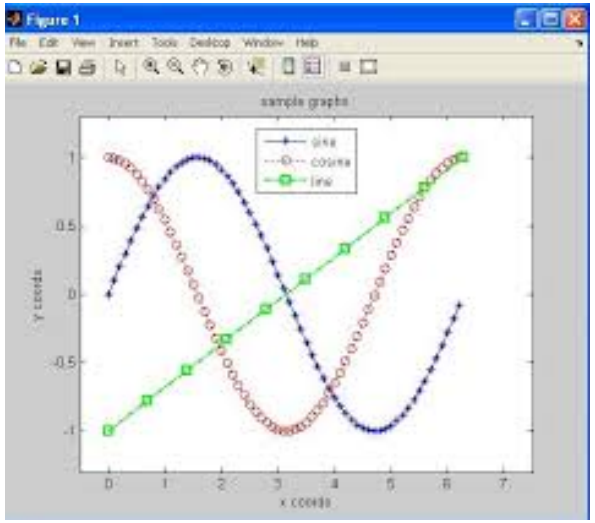
- Hand example from text similar to HW problem
- Example using MATLAB
- Use MATLAB example to create MATLAB code for your HW problem
- MATLAB code expanded on example using MATLAB

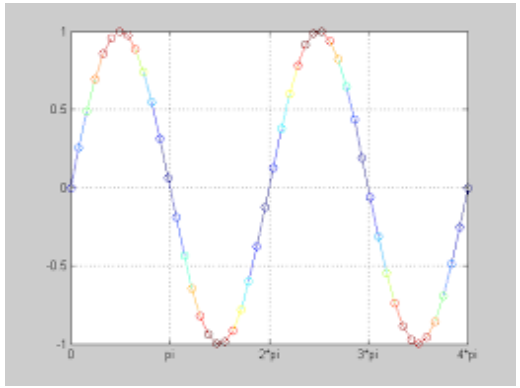
4- **STEM Application –Plotting derivatives**

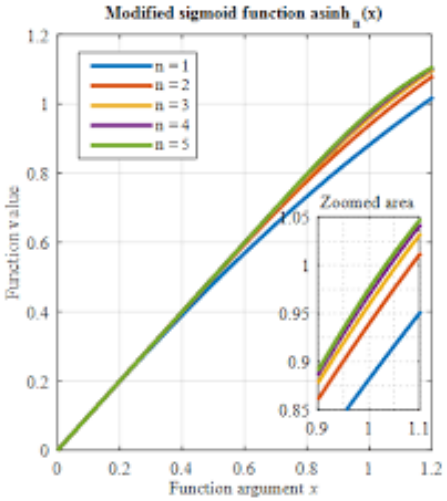
- Hand example from text similar to HW problem
- Example using MATLAB
- Use MATLAB example to create MATLAB code for your HW problem
- MATLAB code expanded on example using MATLAB

5- STEM Application –Plotting sinusoids for electrical circuits

- Hand example from text similar to HW problem
- Example using MATLAB
- Use MATLAB example to create MATLAB code for your HW problem
- MATLAB code expanded on example using MATLAB







Two-dimensional plots

The most useful plot of a STEM professional is the x-y plot. A set of ordered pairs is used to identify points on a two-dimensional graph; the points are then connected by straight lines. The values of x and y may be measured or calculated.

Generally, the **independent variable** is given the name x and is plotted on the x-axis, and the **dependent variable** given the name y and is plotted on the y-axis.

Example to get started! - plot command

Suppose a set of time versus distance data were obtained through measurement.

Values of time in vector called x → x=[0:2:18];

Values of distance in vector called y → y=[0, 0.33, 4.13, 6.29, 6.85, 11.19, 13.19, 13.96, 16.33, 18.17];

Plot these points → plot(x, y)

*> What shows up in your **Command Window** when you enter these 3 lines of MATLAB code?*

```
>> x=[0:2:18];

>> y=[0, 0.33, 4.13, 6.29, 6.85, 11.19, 13.19, 13.96, 16.33, 18.17];

>> plot(x,y)
```

Good STEM format for graphs & plots!

Now add titles, labels, and grids

```
>> x=[0:2:18];

>> y=[0, 0.33, 4.13, 6.29, 6.85, 11.19, 13.19, 13.96, 16.33, 18.17];

 >> plot(x,y)

>> title('Laboratory experiment 1')  % adds title to graph

>> xlabel('Time, t, s')      % adds label on x-axis, note text is enclosed by single quotes ' text '

>> ylabel('Distance, d, ft')     % adds label on x-axis

>> grid on     % turns grid lines on
```

MATLAB code-**notice colors** –
Text color changes to red when you type a single quote (‘), then changes to purple when you type the final single quote (‘).

```
x=[0:2:18];
y=[0, 0.33, 4.13, 6.29, 6.85, 11.19, 13.19, 13.96, 16.33, 18.17];
plot(x,y)
title('Laboratory experiment 1')  % adds title to graph
xlabel('Time,t, s') % adds label on x-axis, note text is enclosed by single quotes ' text '
ylabel('Distance, d, ft')     % adds label on x-axis
grid on     % turns grid lines on
```

Part 1- Basics Plotting Functions

Complete the following problems by writing the MATLAB code & 'copy & paste' graphs to worksheet:

Create plots of the following functions from $x = 0$ to 10.

(a) $y = e^x$
(b) $y = \sin(x)$
(c) $y = ax^2 + bx + c$, where $a = 5$, $b = 2$, and $c = 4$
(d) $y = \sqrt{x}$

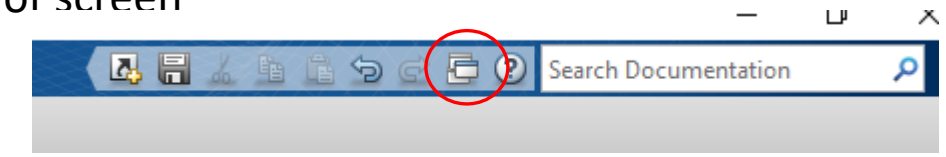
Each of your plots should include a title, an *x*-axis label, a *y*-axis label, and a grid.

plot	Creates an x–y plot	**plot(x,y)**
title	Adds a title to a plot	**title('My Graph')**
xlabel	Adds a label to the x-axis	**xlabel('Independent Variable')**
ylabel	Adds a label to the y-axis	**ylabel('Dependent Variable')**
grid	Adds a grid to the graph	**grid** **grid on** **grid off**
pause	Pauses the execution of the program, allowing the user to view the graph	**pause**
figure	Determines which figure will be used for the current plot	**figure** **figure(2)**
hold	Freezes the current plot, so that an additional plot can be overlaid	**hold on** **hold off**

Lose your figure when doing multiple 'Runs'?

→ Try the figure command

→ Look here in tool bar at top right of screen



Switch Windows - Under FIGURES, click Figure 1 (or the figure you want to view) !

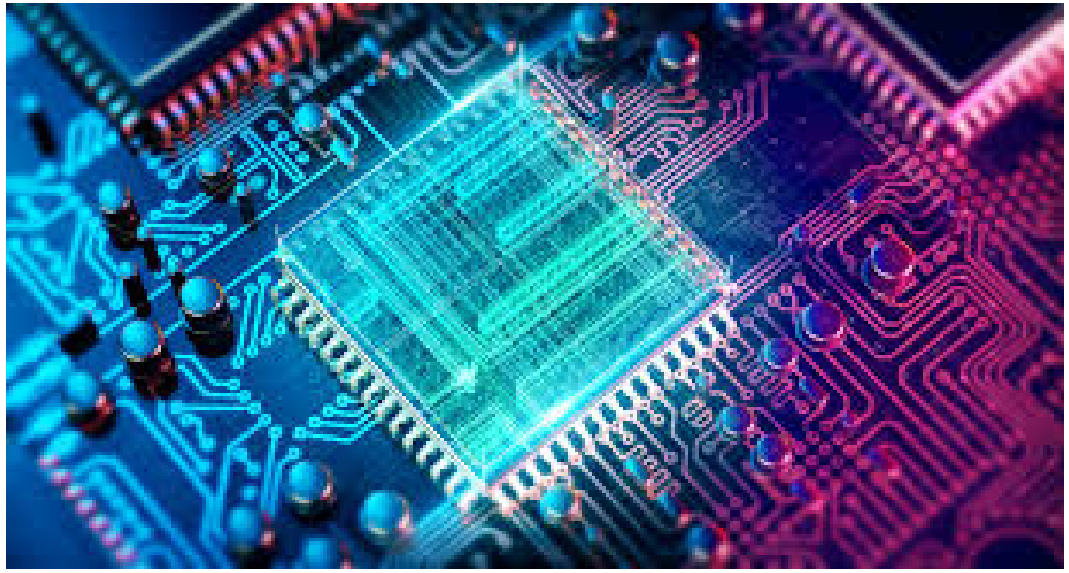
How to plot a multiple curve in MATLAB?

Two functions are given given:

$$y(t) = 4.43t - 4.905t^2$$
$$y1(t) = 2t^2 - 3t^3$$

The time is given t = 0 to 0.9 s

- Plot the function y(t) and y1(t) in a single plot.



A function is given: $y(t) = 4.43t - 4.905t^2$. The time is given t = 0 to 0.9 s

- Plot the function y(t).
- To plot the function we need data points for t; we will use **linspace** command for the data points

Initial data

Final data

No. of data points

```
clc                                  % clear command window
clear                                % Remove items from workspace, freeing up system memory
close all                            % close all previous figures
t = linspace(0,.9,100);              % time in [S]; we have used 100 data points
y = 4.43*t - 4.905*t.^2;             % the function given
```

As t is an array of data's, a dot is given before the ^2

The linewidth command is used make the curve thicker

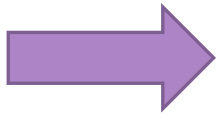
```
figure
plot(t,y,'linewidth',1.5)            % plot is used for drawing a curve
set(gca,'fontsize',12)               % gca: get current axis, fontsize: adjust the font of the axis
xlabel('Time (s)','fontsize',14)     % xlabel: labeling for x-axis
ylabel('y(t) (m)','fontsize',14)     % xlabel: labeling for y-axis
```

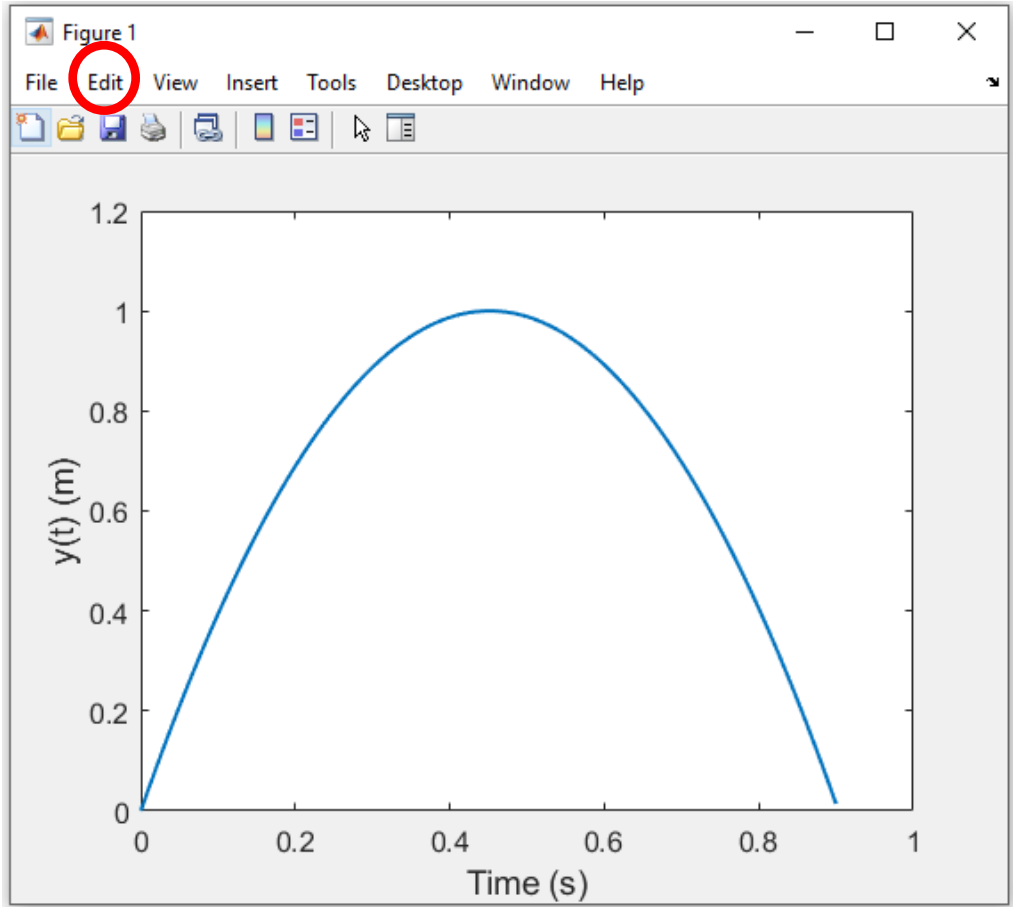
- If fontsize command is not used, the by default font size is 10, which is very small

Helpful hints on using MATLAB as tool for STEM applications

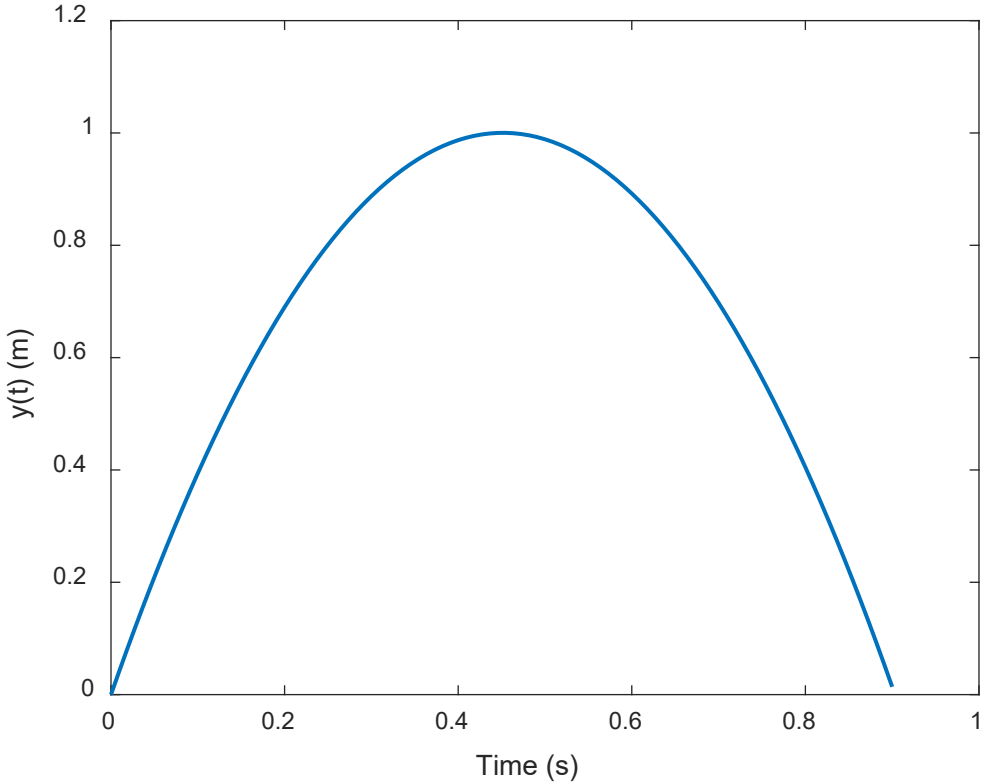
How to copy MATLAB figure to Word/PowerPoint document?

- Figure 1 *(Find it in the top toolbar)*
  - Edit
    - Copy figure
      - Paste to Word/PowerPoint

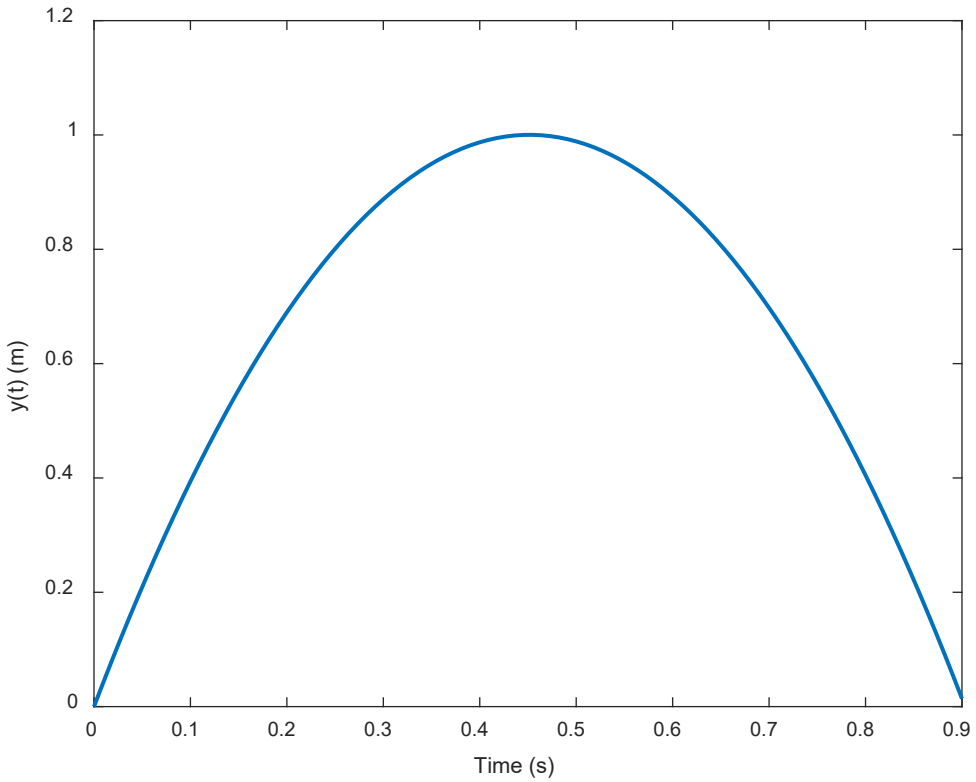




After copy from MATLAB to PowerPoint



For the font size of 14



For the by default font size of 10

Part 2- 'more that one line on a graph '– more commands

Plots with more than one line on graph –

→ layer plots on top of one another by using the **hold on** command

```
>> x=0: pi/100:2*pi;
>> y1=cos(x*4);
>> plot(x,y1)
>> y2=sin(x);
>> hold on;
>> plot (x, y2)
>> hold off  % command that stops the layering of  lines on the same plot, use as needed
>> plot(x, y1, x, y2)  % produces a graph with 2 lines of different colors , use hold on /hold off or this syntax
```

For code below, how many lines in the plot?
Use MATLAB to verify your answer!

```
X = 0:pi/100:2*pi;
Y1 = cos(X)*2;
Y2 = cos(X)*3;
Y3 = cos(X)*4;
Y4 = cos(X)*5;
Z = [Y1; Y2; Y3; Y4];
plot(X, Y1, X, Y2, X, Y3, X, Y4)
```

Try different variations of this MATLAB code in support of problems in part 2 and rest of mini-project #4.

Part 2- 'Adding format details '– more commands

** Try this example! **

```
x=[1:10];
y=[58.5, 63.8, 64.2, 67.3, 71.5, 88.3,90.1, 90.6, 89.5, 90.4];
plot(x,y, ':ok')
```

What happens? Use the table below to change the line, mark, and color options!

Line Type	Indicator	Point Type	Indicator	Color	Indicator
solid	-	point	.	blue	b
dotted	:	circle	o	green	g
dash-dot	-.	x-mark	x	red	r
dashed	--	plus	+	cyan	c
		star	*	magenta	m
		square	s	yellow	y
		diamond	d	black	k
		triangle down	v	white	w
		triangle up	^		
		triangle left	<		
		triangle right	>		
		pentagram	p		
		hexagram	h		

→ **Replace the plot command with this one. What happens?**

```
plot(x,y, ':ok', x,y*2,'--xr', x, y/2,'-b')
```

In Command Window type **>> help plot**

Command Window

```
    Example
       x = -pi:pi/10:pi;
       y = tan(sin(x)) - sin(tan(x));
       plot(x,y,'--rs','LineWidth',2,...
                      'MarkerEdgeColor','k',...
                      'MarkerFaceColor','g',...
                      'MarkerSize',10)

    See also plottools, semilogx, semilogy, loglog, plotyy, plot3, grid,
    title, xlabel, ylabel, axis, axes, hold, legend, subplot, scatter.

    Reference page for plot
    Other functions named plot
```

For everything you ever wanted to know about plotting in MATLAB, well, maybe too much for this project!

Initial data

Final data

No. of data points

```
clc                                  % clear command window
clear                                % Remove items from workspace, freeing up system memory
close all                            % close all previous figures
t = linspace(0,.9,100);              % time in [S]
y = 4.43*t - 4.905*t.^2;             % the function given
y1 = 2*t.^2 - 3*t.^3;                % the function given
```

As t is an array of data's, a dot is given before the ^2

b is for blue color

r is for red color

```
figure
plot(t,y,'b',t,y1,'r','linewidth',1.5) % plot is used for drawing a curve
set(gca,'fontsize',12)               % gca: get current axis, fontsize: adjust the font of the axis
xlabel('Time (s)','fontsize',14)     % xlabel: labeling for x-axis
ylabel('Position (m)','fontsize',14)  % xlabel: labeling for y-axis
legend('y(t)','y_1(t)')              % legend: used to distinguish the two curves
```

Helpful hints on using MATLAB as tool for STEM applications

Pay close attention on how to create 'independent' and 'dependent' variables! Above it's t ,and y & y1, respectively.

Part 2- Adding format details to your plot

Complete the following problems:

Plot the following functions on the same graph for x values from $-\pi$ to π, selecting spacing to create a smooth plot:

$$y_1 = \sin(x)$$
$$y_2 = \sin(2x)$$
$$y_3 = \sin(3x)$$

(*Hint*: Recall that the appropriate MATLAB® syntax for $2x$ is **2*x.**)

Adjust the plot created in Problem 5.3 so that:

- Line 1 is red and dashed.
- Line 2 is blue and solid.
- Line 3 is green and dotted.

Do not include markers on any of the graphs. In general, markers are included only on plots of measured data, not for calculated values.

Adjust the plot created in Problem so that the x-axis goes from -6 to $+6$.

- Add a legend.
- Add a text box describing the plots.

Helpful syntax:

legend('string1', 'string 2', etc)	Allows you to add a legend to your graph. The legend shows a sample of the line and lists the string you have specified.
text(x_coordinate,y_coordinate, 'string')	Allows you to add a text box to the graph. The box is placed at the specified *x*- and *y*-coordinates and contains the string value specified.

Example:

```
x=[1:10];
y=[58.5, 63.8, 64.2, 67.3, 71.5, 88.3,90.1, 90.6, 89.5, 90.4];
plot(x,y, ':ok', x,y*2,'--xr', x, y/2,'-b')
legend ('line 1', 'line 2', 'line 3')
text(2,100,'Label plots with the text commands')
title ('Example graph for mini-project #4')
```

Part 3. STEM application – projectile motion

A. Hand example from text similar to HW problem
B. Example using MATLAB
C. Use MATLAB example to create MATLAB code for your HW problem
D. MATLAB code expanded on example using MATLAB

Part 3 – STEM Applications with projectile motion

BALLISTICS

The range of an object (see Figure 5.12) shot at an angle θ with respect to the x-axis and an initial velocity v_0 is given by

$$R(\theta) = \frac{v^2}{g}\sin(2\theta) \qquad \text{for } 0 \le \theta \le \frac{\pi}{2} \text{(neglecting air resistance)}$$

Use $g = 9.9$ m/s^2 and an initial velocity of 100 m/s. Show that the maximum range is obtained at $\theta = \pi/4$ by computing and plotting the range for values of θ from

$$0 \le \theta \le \frac{\pi}{2}$$

in increments of 0.05.

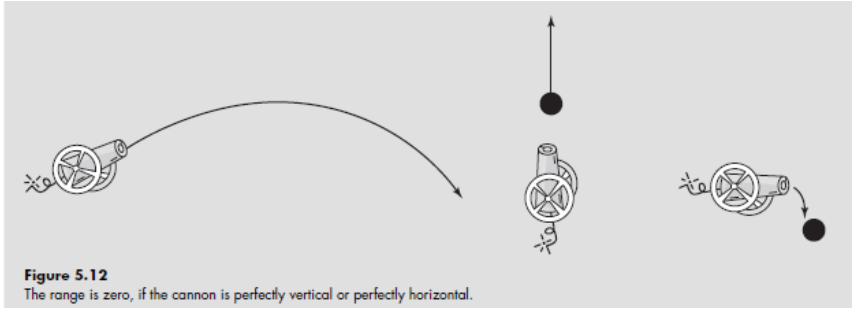


Figure 5.12
The range is zero, if the cannon is perfectly vertical or perfectly horizontal.

Repeat your calculations with an initial velocity of 50 m/s, and plot both sets of results on a single graph.

1. State the Problem
   Calculate the range as a function of the launch angle.
2. Describe the Input and Output

Input

$g = 9.9$ m/s^2
$\theta = 0$ to $\pi/2$, incremented by 0.05
$v_0 = 50$ m/s and 100 m/s

Output

Range R
Present the results as a plot.

3. Develop a Hand Example
   If the cannon is pointed straight up, we know that the range is zero, and if the cannon is horizontal, the range is also zero (see Figure 5.12).
   This means that the range must increase with the cannon angle up to some maximum and then decrease. A sample calculation at 45° ($\pi/4$ radians) shows that

$$R(\theta) = \frac{v^2}{g}\sin(2\theta)$$

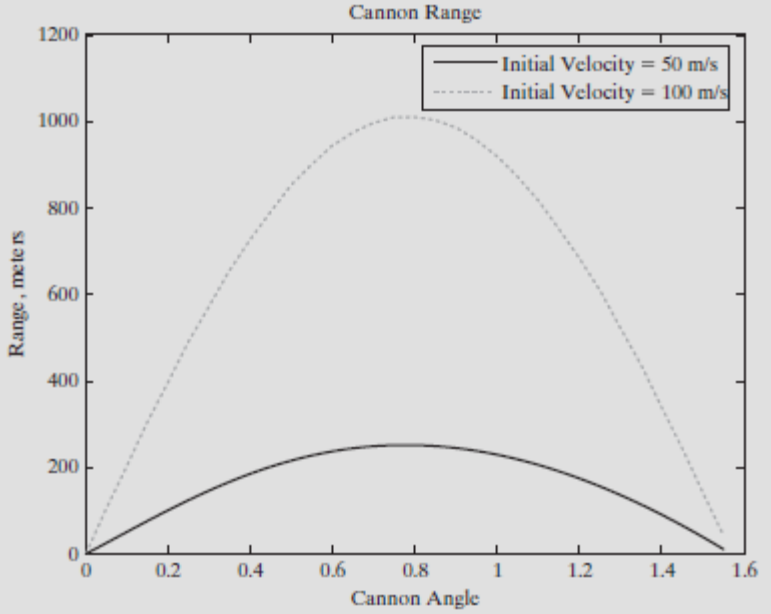
$$R\left(\frac{\pi}{4}\right) = \frac{100^2}{9.9}\sin\left(\frac{2\pi}{4}\right) = 1010 \text{ m when the initial velocity is 100 m/s}$$

Part 3 – STEM Applications with projectile motion

4. Develop a MATLAB® Solution

```
%Example 5.2
%The program calculates the range of a ballistic projectile
%
%Define the constants
  g = 9.9;
  v1 = 50;
  v2 = 100;
%Define the angle vector
  angle = 0:0.05:pi/2;
%Calculate the range
  R1 = v1^2/g*sin(2*angle);
  R2 = v2^2/g*sin(2*angle);
%Plot the results
  plot(angle,R1,angle,R2,':')
  title('Cannon Range')
  xlabel('Cannon Angle')
  ylabel('Range, meters')
  legend('Initial Velocity=50 m/s', 'Initial Velocity=100 m/s')
```

Notice that in the `plot` command, we requested MATLAB® to print the second set of data as a dashed line. A title, labels, and a legend were also added. The results are plotted in Figure 5.13.



5. Test the Solution

Compare the MATLAB® results with those from the hand example. Both graphs start and end at zero. The maximum range for an initial velocity of 100 m/s is approximately 1000 m, which corresponds well to the calculated value of 1010 m. Notice that both solutions peak at the same angle, approximately 0.8 radian. The numerical value for $\pi/4$ is 0.785 radian, confirming the hypothesis presented in the problem statement that the maximum range is achieved by pointing the cannon at an angle of $\pi/4$ radians (45°).

Part 3 – STEM Applications with projectile motion
Diver launching of a springboard!

A. Hand example from text similar to HW problem
B. Example using MATLAB
C. Use MATLAB example to create MATLAB code for your HW problem
D. MATLAB code expanded on example using MATLAB

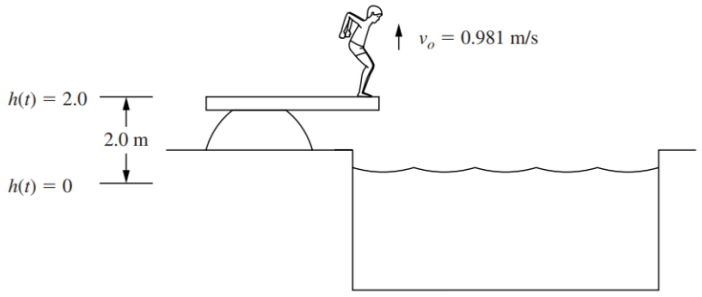


Figure P2.32 Diver jumping off a diving board.

Plot using MATLAB!

Use example to help create MATLAB code & properly labeled graph!

A diver jumps off a diving board 2.0 m above the water with an initial vertical velocity of 0.981 m/s as shown in Fig. P2.32. The height $h(t)$ above the water is given by

$$h(t) = -4.905\,t^2 + 0.981t + 2.0 \text{ m}.$$

(a) Find the time in seconds when the diver hits the water. Use both the quadratic formula and completing the square.

(b) Find the maximum height of the diver if it is known to occur at $t = 0.1$ s.

(c) Use the results of parts (a) and (b) to sketch the height $h(t)$ of the diver.

Part 4. Plotting derivatives in MATLAB

A. Hand example from text similar to HW problem
B. Example using MATLAB
C. Use MATLAB example to create MATLAB code for your HW problem
D. MATLAB code expanded on example using MATLAB

Part 4. Differential in MATLAB

Differentiation The derivative is an **instantaneous rate of Change**.

The derivation of a function y(t) is $\frac{dy}{dt}$

- y = 10sin(10*t); what is the derivative of the function y.
- To illustrate how to take derivatives using Symbolic Math Toolbox™ software, first create a symbolic expression:

```
syms t                % here t is the independent variable; syms is symbolic function of MATLAB
y = 10*sin(10*t);     % the function given
dy_dt = diff(y);      % diff is a built-in function used for differentiation
dy_dt


ans =
100*cos(10*t)
```

The MATLAB code for **derivatives**, *aka*, **differentiation**

```
clc                     % clear command window
clear                   % Remove items from workspace, freeing up system memory
syms t                  % here t is the independent variable; syms is symbolic function of MATLAB
y = 10*sin(10*t);       % the function given
dy_dt = diff(y);        % diff is a built-in function used for differentiation or derivative
dy_dt
```

Ans

```
dy_dt =

100*cos(10*t)
```

A function is given: $y(t) = 4.43t - 4.905t^2$. The time is given t = 0 to 0.9 s

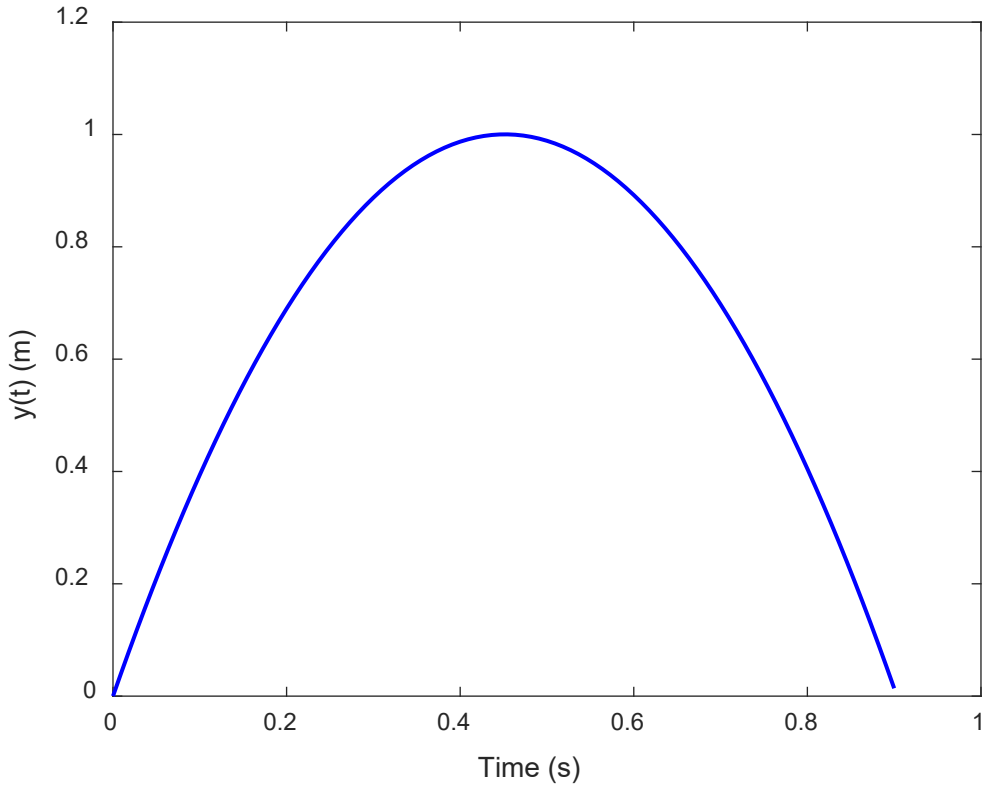
- Plot the function y(t).
- Plot the derivative of y(t); i.e., dy/dt *Yes, you need to 'differentiate' y(t).*

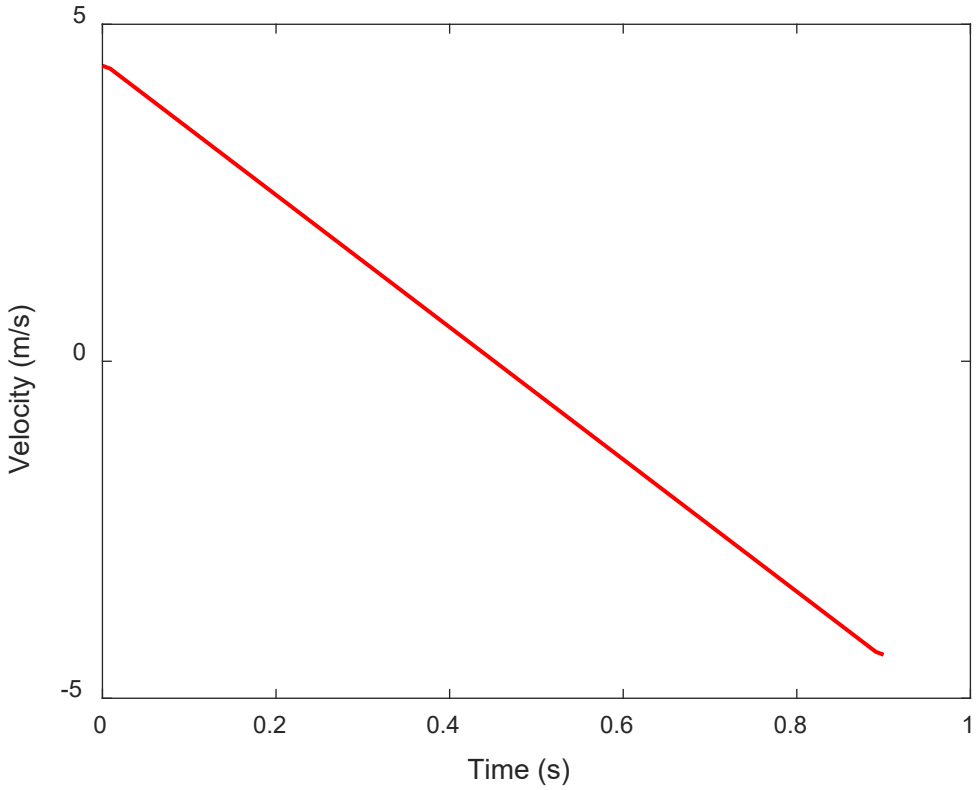
```
clc                                 % clear command window
clear                               % Remove items from workspace, freeing up system memory
close all                           % close all previous figures
t = linspace(0,.9,100);             % time in [S]
y = 4.43*t - 4.905*t.^2;            % the function given

figure
plot(t,y,'b','linewidth',1.5) % plot is used for drawing a curve
set(gca,'fontsize',12)              % gca: get current axis, fontsize: adjust the font of the
axis
xlabel('Time (s)','fontsize',14)  % xlabel: labeling for x-axis
ylabel('Position (m)','fontsize',14)  % xlabel: labeling for y-axis
legend('y(t)')

dy_dt = gradient(y,t);              % gradient is used to differentiate: dy_dt
% when numerical data are used to perform derivative, it is better to use gradient command
rather than diff command


figure
plot(t,dy_dt,'r','linewidth',1.5)
set(gca,'fontsize',12)
xlabel('Time (s)','fontsize',14)
ylabel('Velocity (m/s)','fontsize',14)
```





Here's the plots of distance y(t) versus time & Velocity versus time using the built in functions for derivatives in MATLAB!

Part 4 – STEM Applications with projectile motion

A. Hand example from text similar to HW problem
B. Example using MATLAB
C. Use MATLAB example to create MATLAB code for your HW problem
D. MATLAB code expanded on example using MATLAB

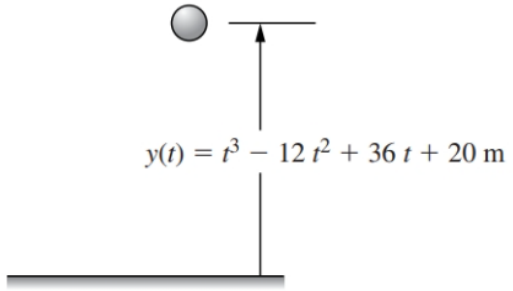


Figure P8.7 Motion of a particle in the vertical plane.

Plot using MATLAB!

Use example to help create MATLAB code & properly labeled graph!

8-7. The motion of a particle in the vertical plane is shown in Fig. P8.7. The height of the particle is given by

$$y(t) = t^3 - 12\,t^2 + 36\,t + 20 \text{ m}.$$

(a) Find the values of position and acceleration when the **velocity** is zero.

(b) Use your results in part (a) to sketch $y(t)$ for $0 \leq t \leq 9$ s.

Part 5. Electrical Circuits example

A. Hand example from text similar to HW problem
B. Example using MATLAB
C. Use MATLAB example to create MATLAB code for your HW problem
D. MATLAB code expanded on example using MATLAB

Part 5 – STEM Applications with electrical circuits

A. Hand example from text similar to HW problem
B. Example using MATLAB
C. Use MATLAB example to create MATLAB code for your HW problem
D. MATLAB code expanded on example using MATLAB

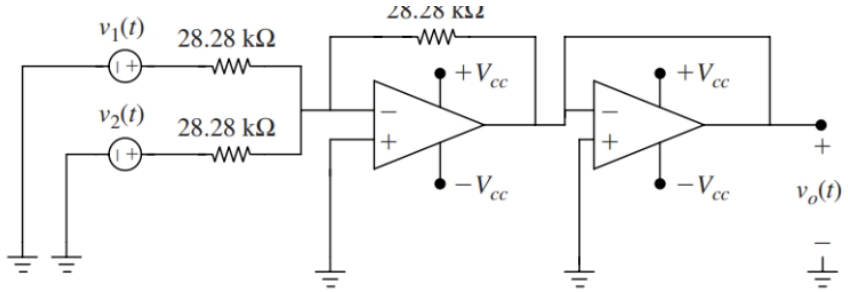


Figure P6.29 An Op–Amp circuit for problem P6-29.

Plot using MATLAB!

b) Plot one cycle of voltage $v_1(t)$ with and without time shift. (Two lines on the same graph)

6-29. Two voltages $v_1(t) = 10\sqrt{2}\sin\left(500\pi t - \frac{3\pi}{4}\right)$ V and $v_2(t) = 5\sin(500\pi t)$ V are applied to the Op–Amp circuit shown in Fig. P6.29.

(a) Write down the amplitude, frequency (in Hz), period (in seconds), phase angle or phase shift (in radians), and time shift (in seconds) of the voltage $v_1(t)$.

(b) Plot one cycle of the voltage $v_1(t)$, and indicate the earliest time after $t = 0$ when the voltage is $10\sqrt{2}$ V.

(c) The output voltage $v_o(t)$ is given by $v_o(t) = v_1(t) + v_2(t)$. Write $v_o(t)$ in the form $v_o(t) = M\sin(500\pi t + \theta)$ (i.e., find M and θ).

Deliverables for Mini-Project #4

Due: Wednesday, December 8, 2021, last day of class ☺

Work in again in *different* Peer Partnerships For Programming Feedback, P^{3}F

- Collaborate in groups of 3 to help each other with programming issues , syntax checks, etc., as needed
- Groups randomly assigned through Canvas

Mini-project activities – 5 parts

1- **Basic Plotting Skills** : Creating super STEM graphs to 'spec' – Syntax and operations

2- **Practice More Plotting Skills with *Special* Format in MATLAB** – examples

3- **STEM application –Projectile motion & diver launching off a springboard**

- Hand example from text similar to HW problem
- Example using MATLAB
- Use MATLAB example to create MATLAB code for your HW problem
- MATLAB code expanded on example using MATLAB

4- **STEM Application –Plotting derivatives**

- Hand example from text similar to HW problem
- Example using MATLAB
- Use MATLAB example to create MATLAB code for your HW problem
- MATLAB code expanded on example using MATLAB

5- STEM Application –Plotting sinusoids for electrical circuits

- Hand example from text similar to HW problem
- Example using MATLAB
- Use MATLAB example to create MATLAB code for your HW problem
- MATLAB code expanded on example using MATLAB

Reference

Selected materials from :

Moore, Holly.
MATLAB® for engineers / Holly Moore. — 3rd ed.
p. cm.
Includes index.
ISBN-13: 978-0-13-210325-1
ISBN-10: 0-13-210325-7
1. Engineering mathematics—Data processing. 2. MATLAB®. I. Title.
TA345.M585 2011
620.00151—dc23
2011022739