Introduction to Engineering MATLAB – 7 Plotting

Agenda
- Plotting
- Basic 2-D plots

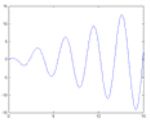
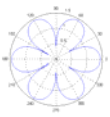
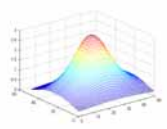
Note: when working through this lesson on your own, type each example step by step in the command window to learn the effects of each individual step. Then create the script files shown for each example to see how it's run all at once.

1

MAKING X-Y PLOTS

MATLAB has many functions and commands that can be used to create various types of plots.

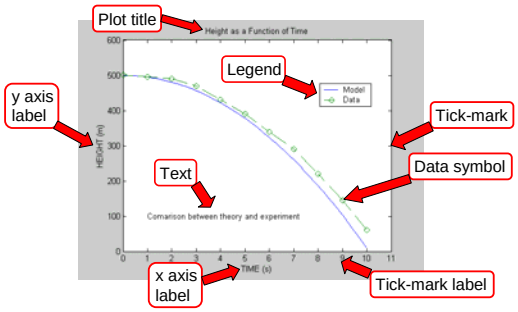
In our class we will only create two dimensional x –y plots.

2

EXAMPLE OF A 2-D PLOT

The plot displays the height as a function of time for a falling object.



3

TWO-DIMENSIONAL PLOT COMMAND

The basic 2-D plot command is:

plot(x,y) where x is a vector (one dimensional array), and y is a vector. Both vectors must have the same number of elements.

❖ The plot command creates a single curve with the x values on the *abscissa* (horizontal axis) and the y values on the *ordinate* (vertical axis).

❖ The curve is made from segments of lines that connect the points that are defined by the x and y coordinates of the elements in the two vectors.

4

CREATING THE X AND Y VECTORS

❖ If data is given, the information is entered as the elements in the vectors x and y.

❖ If the values of y are determined by a function from the values of x, then a vector x is created first, and then the values of y are calculated for each value of x. The spacing (difference) between the elements of x must be such that the plotted curve will show the details of the function.

5

CREATING A PLOT OF POPULATION GROWTH

Data from Matlab # 2:

Year	1984	1986	1988	1990	1992	1994	1996
Population	127	130	136	145	158	178	211

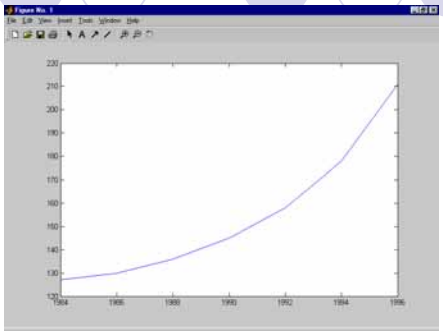
A plot can be created by the commands shown below. This can be done in the command window, or by writing and running a script file.

```
>> year = [1984 1986 1988 1990 1992 1994 1996];
>> pop =[127 130 136 145 158 178 211];
>> plot(year,pop)
```

Once the plot command is executed, the Figure Window opens with the following plot.

6

A PLOT OF POPULATION GROWTH



7

CREATING A PLOT OF A FUNCTION

Consider: $y = 3^{-0.6x}\sin(5x)$ for $0 \le x \le 5$

A script file for plotting the function is:

```
%  A script file that creates a plot of
%  the function: 3^(-0.6x)*sin(5x)
x = [0:0.01:5];
y = 3.^(-0.6*x).*sin(5*x);
plot(x,y)
```

Creating a vector with spacing of 0.01

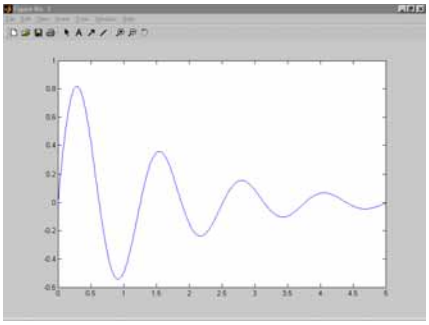
Calculating a value of *y* for each *x*

Once the plot command is executed, the Figure Window opens with the following plot.

8

A PLOT OF A FUNCTION

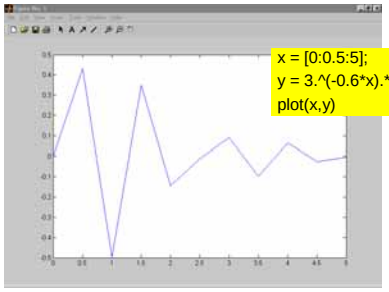
$y = 3^{-0.6x}\sin(5x)$ for $0 \le x \le 5$



9

CREATING A PLOT OF A FUNCTION

If the vector x is created with large spacing, the graph is not accurate. Below is the previous plot with spacing of 0.5.



```
x = [0:0.5:5];
y = 3.^(-0.6*x).*sin(5*x);
plot(x,y)
```

10

FORMATTING PLOTS

A plot can be formatted to have a required appearance.

With formatting you can:

- Add title to the plot.
- Add labels to axes.
- Change line type, thickness, and color of the plot line.
- Change range of the axes.
- Add legend.
- Add text blocks.
- Add grid.

11

FORMATTING PLOTS

There are two methods to format a plot:

1. Formatting commands.
   In this method commands, that make changes or additions to the plot, are entered after the plot command. This can be done when working in the command window, or as part of a program in a script file.

2. Formatting the plot in the figure window.
   In this method the plot is formatted by clicking on the plot and using the menu to make changes or add details.

12

FORMATTING PLOTS

The plot command can include options that define the line (color, style, color) of line that is used in plotting the curve.

plot(x,y,'color_linestyle_marker')

Color Code	Color	Line Code	Line Type	Marker Code	Marker
y	yellow	-	solid	.	point
m	magenta	:	dotted	o	circle
c	cyan	-.	Dashdot	x	x-mark
r	red	--	dashed	+	plus
g	green			*	star
b	blue			s	square
w	white			d	diamond
k	black			v	triangle

13

EXAMPLES OF THE PLOT COMMAND

plot(x,y,'r') Plots *y* vs. *x* with a red solid (default) line.

plot(x,y,'--') Plots *y* vs. *x* with a black (default) dashed line.

plot(x,y,'g:') Plots *y* vs. *x* with a green dotted line.

plot(x,y,'+') Plots *y* vs. *x* as points marked with + (no line connects the points)

plot(x,y,'b*-.') Plots *y* vs. *x* with points marked with stars and blue dashdot line that connects the points.

14

FORMTTING COMMANDS

title('string') Add the string as a title at the top of the plot.

xlabel('string') Add the string as a label to the x-axis.

ylabel('string') Add the string as a label to the y-axis.

axis([xmin xmax ymin ymax]) Sets the min and max limits of the x- and y-axes.

15

FORMTTING COMMANDS

legend('string 1','string 2','string 3')

Creates a legend using the strings to label various curves (when several curves are in one plot). The location of the legend is specified by the mouse.

text(x,y,'string')

Places the string (text) on the plot at coordinate x,y relative to the plot axes.

gtext('string')

Places the string (text) on the plot. When the command executes the figure window pops up and the text location is clicked with the mouse.

16

EXAMPLE OF FORMATTED PLOT

Below is a script file of a formatted population growth plot

```
year = [1984 1986 1988 1990 1992 1994 1996];
pop =[127 130 136 145 158 178 211];
plot(year,pop,'g--s')
xlabel('YEAR')
ylabel('POPULATION (MILLIONS)')
title('Population Growth From 1984 to 1996')
axis([1982 1998 100 250])
text(1984,220,'Data from Reference [3]')
```

Creating a vector of the years numbers

Creating a vector with the population data

Plotting pop vs. year with a dashed green line and square markers

Labels for the axes

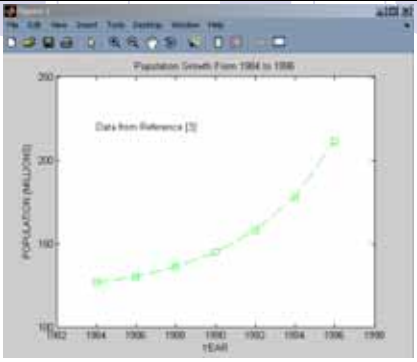
Title for the plot

Setting limits of the axes

Creating text

17

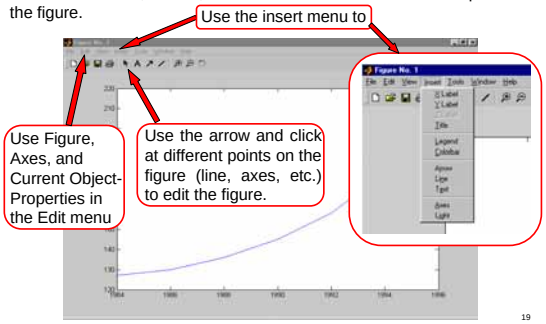
POPULATION GROWTH PLOT AFTER FORMATTING



18

FORMATTING A PLOT IN THE FIGURE WINDOW

Once a figure window is open, the figure can be formatted interactively. You can add lines, arrows, and text to draw attention to certain parts of the figure.



19