

Introduction to MATLAB

Dr./ Ahmed Mohamed Nagib Elmekawy

Mechanical Engineering department, Alexandria university, Egypt

Spring 2017

Chapter 1

An Overview of MATLAB

The Default MATLAB Desktop:

The screenshot displays the MATLAB Desktop environment. The interface is divided into several panes:

- Current Directory:** Shows a list of files in the 'MyMATLAB...' directory, including 'actuator.m', 'cost.m', 'CubicEval.m', 'cubicsprin...', 'dcmotor.m', 'ethanol.m', 'Falling_Sp...', 'friction1.mdl', 'half-sine.m', 'height.m', 'perfspecs.m', 'plantcost.m', 'speed.m', 'TorqueDat...', 'trapezoid.m', and 'trebuchet...'. The 'trapezoid.m' file is selected, and its description 'trapezoid.m Trapezoidal voltage profile' is shown below.
- Command Window:** Contains the following MATLAB commands and their outputs:

```
>> clear
>> A=2*5^3
A =
    250
>> B=exp(0.005*A)
B =
    3.4903
>> C=4*sqrt(A+B^3)
C =
    68.4130
>> D=5*log10(1000)
D =
    15
>> x = 0:0.02:9;
>> y = A*sin(3*x);
>> plot(x,y)
```
- Workspace:** Displays a table of variables in the workspace:

Name	Value	Min	Max
A	250	250	250
B	3.4903	3.4903	3.4903
C	68.4130	68.4130	68.4130
D	15	15	15
x	<1x451 double>	0	9
y	<1x451 double>	-249.9...	249.9...
- Command History:** Shows a list of commands entered in the Command Window, including 'clear', 'A=2*5^3', 'B=exp(0.05*A)', 'C=4*sqrt(A+B^3)', 'D=5*log10(1000)', 'x = 0:0.03:9', 'clc', 'plot(x,y)', and a second set of 'clear' and assignment commands.

Scalar arithmetic operations

Symbol	Operation	MATLAB form
\wedge	exponentiation: a^b	a^b
$*$	multiplication: ab	$a*b$
$/$	right division: $a/b = \frac{a}{b}$	a/b
\backslash	left division: $a \backslash b = \frac{b}{a}$	$a \backslash b$
$+$	addition: $a + b$	$a+b$
$-$	subtraction: $a - b$	$a-b$

Example 1.1

Type the following in command windows:

```
>> 8/10
```

```
ans =
```

```
    0.8000
```

```
>> 5*ans
```

```
ans =
```

```
    4
```

```
>> r=8/10
```

```
r =
```

```
    0.8000
```

```
>> r
```

```
r =
```

```
    0.8000
```

```
>> s=20*r
```

```
s =
```

```
   16
```

Variables

- You do not need to declare the variable names and sizes at the beginning of your program as Matlab declare them automatically.
- You should avoid using variable names that Matlab uses as a function or command such as:

plot	end
if	for
while	pi
sin	cos

- You can check to see if a command, function or file name already exists by using the `exist('name')` command.
- Lower and upper case variables are different.

Example 1.2

Type the following in command windows:

```
>> Y=5
```

```
Y =
```

```
5
```

```
>> y=4
```

```
y =
```

```
4
```

```
>> x=y+Y
```

```
x =
```

```
9
```

Order of precedence

Precedence	Operation
First	Parentheses, evaluated starting with the innermost pair.
Second	Exponentiation, evaluated from left to right.
Third	Multiplication and division with equal precedence, evaluated from left to right.
Fourth	Addition and subtraction with equal precedence, evaluated from left to right.

Example 1.3 - Precedence

Type the following in command windows:

```
>> 8 + 3*5
```

```
ans =
```

```
23
```

```
>> 8 + (3*5)
```

```
ans =
```

```
23
```

```
>> (8 + 3) *5
```

```
ans =
```

```
55
```

```
>>4^2 - 12 - 8/4*2
```

```
ans =
```

```
0
```

```
>>4^2 - 12 - 8/(4*2)
```

```
ans =
```

```
3
```


Commands for managing the work session

Command	Description
<code>clc</code>	Clears the Command window.
<code>clear</code>	Removes all variables from memory.
<code>clear var1 var2</code>	Removes the variables <code>var1</code> and <code>var2</code> from memory.
<code>exist('name')</code>	Determines if a file or variable exists having the name 'name'.
<code>quit</code>	Stops MATLAB.
<code>who</code>	Lists the variables currently in memory.
<code>whos</code>	Lists the current variables and sizes, and indicates if they have imaginary parts.
<code>:</code>	Colon; generates an array having regularly spaced elements.
<code>,</code>	Comma; separates elements of an array.
<code>;</code>	Semicolon; suppresses screen printing; also denotes a new row in an array.
<code>...</code>	Ellipsis; continues a line.

Special variables and constants

Command	Description
ans	Temporary variable containing the most recent answer.
eps	Specifies the accuracy of floating point precision.
i,j	The imaginary unit $\sqrt{-1}$.
Inf	Infinity.
NaN	Indicates an undefined numerical result.
pi	The number π .

Complex Number Operations

- The number $c_1 = 1 - 2i$ is entered as follows:

$$c1 = 1-2i.$$

- An asterisk is not needed between i or j and a number, although it is required with a variable, such as $c2 = 5 - i*c1$.

- Be careful. The expressions

$$y = 7/2*i$$

and

$$x = 7/2i$$

give two different results:

$$y = (7/2)i = 3.5i$$

and

$$x = 7/(2i) = -3.5i.$$

Example 1.4

Type the following in command windows:

```
>> i = 5
```

```
i =
```

```
5
```

```
>> s = 3 + 7i
```

```
w =
```

```
3.0000 + 7.0000i
```

```
>> w = 3 + 7 * i
```

```
w =
```

```
38
```

Numeric display formats:

The `format` command controls how numbers appear on the screen. The table below gives the variants of this command. MATLAB uses many significant figures in its calculations, but we rarely need to see all of them. The default MATLAB display format is the short format, which uses four decimal digits. You can display more by typing `format long`, which gives 16 digits. To return to the default mode, type `format short`.

Command	Description and example
<code>format short</code>	Four decimal digits (the default); 13.6745.
<code>format long</code>	16 digits; 17.27484029463547.
<code>format short e</code>	Five digits (four decimals) plus exponent; 6.3792e+03.
<code>format long e</code>	16 digits (15 decimals) plus exponent; 6.379243784781294e-04.

Example 1.5

Type the following in command windows:

```
>> pi
```

```
ans =      3.1416
```

```
>> format long
```

```
>> pi
```

```
ans =      3.141592653589793
```

```
>> format short
```

```
>> pi
```

```
ans =      3.1416
```

Arrays

- The numbers 0, 1, 2, ..., 10 can be assigned to the variable u by typing $u = 0:10$.
- To compute $w = 5 \sin u$ for $u = 0, 1, 2, \dots, 10$, the session is;

```
>>u = 0:10;  
>>w = 5*sin(u);  
>>plot(u,w)
```

- The single line, $w = 5 \sin(u)$, computed the formula $w = 5 \sin u$ 11 times.

Arrays

- The numbers 0, 0.1, 0.2, ..., 10 can be assigned to the variable u by typing `u = 0:0.1:10`.
- To compute $w = 5 \sin u$ for $u = 0, 0.1, 0.2, \dots, 10$, the session is;

```
>>q = 0:0.1:10;  
>>r = 5*sin(u);  
>>plot(u,w)
```

- The single line, `w = 5*sin(u)`, computed the formula $w = 5 \sin u$ 101 times.

Array Index

```
>>u (7)           ← Gets the 7th element of u  
ans =  
    0.6000
```

```
>>w (7)           ← Gets the 7th element of w  
ans =  
    2.8232
```

- Use the `length` function to determine how many values are in an array.

```
>>m = length(w)  
m =  
    101
```

Polynomial Roots

To find the roots of $x^2 - 5x + 6 = 0$, the session is

```
>>a = [1, -5, 6];
```

```
>>roots(a)
```

```
ans =
```

```
    3.0000
```

```
    2.0000
```

The roots are $x = 3$ and $x = 2$.

Polynomial Roots

To find the roots of $x^3 - 7x^2 + 40x - 34 = 0$, the session is

```
>>a = [1, -7, 40, -34];  
>>roots(a)  
ans =  
    3.0000 + 5.0000i  
    3.0000 - 5.0000i  
    1.0000
```

The roots are $x = 1$ and $x = 3 \pm 5i$.

Some commonly used mathematical functions

Function	MATLAB syntax ¹
e^x	<code>exp(x)</code>
\sqrt{x}	<code>sqrt(x)</code>
$\ln x$	<code>log(x)</code>
$\log_{10} x$	<code>log10(x)</code>
$\cos x$	<code>cos(x)</code>
$\sin x$	<code>sin(x)</code>
$\tan x$	<code>tan(x)</code>
$\cos^{-1} x$	<code>acos(x)</code>
$\sin^{-1} x$	<code>asin(x)</code>
$\tan^{-1} x$	<code>atan(x)</code>

¹The MATLAB trigonometric functions use radian measure.

Example 1.6

Type the following in command windows:

```
>> x=sin(0)
```

```
x =      0
```

```
>> x=sin(90)
```

```
x =    0.8940
```

```
>> x=sind(0)
```

```
x =      0
```

```
>> x=sind(90)
```

```
x =      1
```

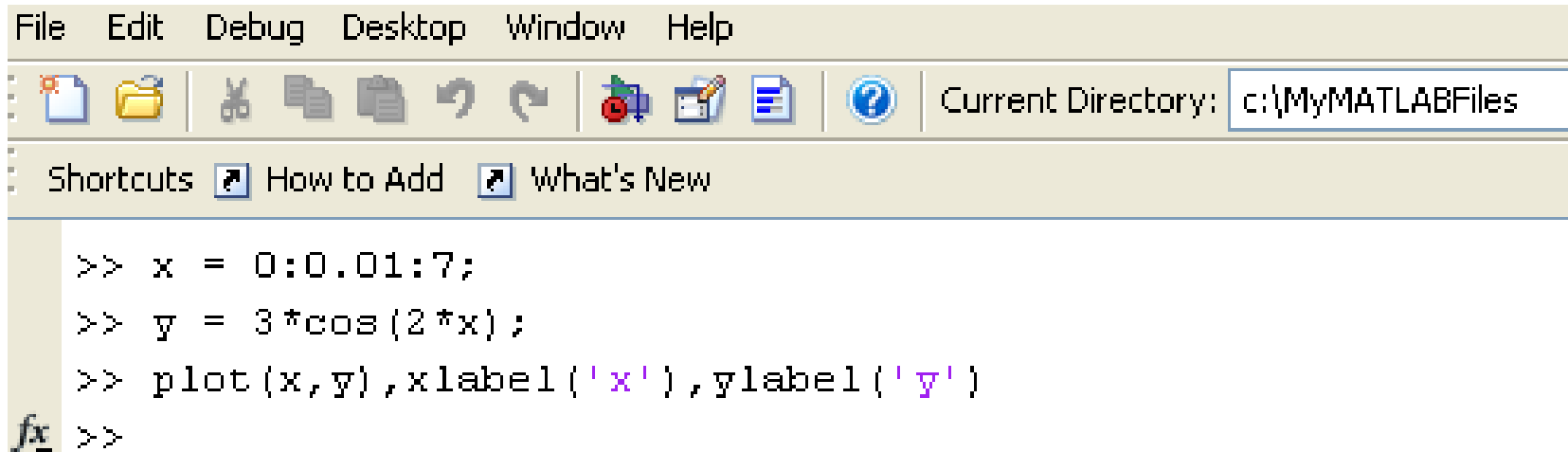
M-files

- Once you close Matlab program, All the commands you enter in the command windows is not saved and can not be used later.
- M-files can be saved and used later.
- From now all the programs will be written in M-files.
- The name of a script file must begin with a letter, and may include digits and the underscore character, up to 63 characters.
- Do not give a M-file the same name as a variable.
- Do not give a M-file the same name as a MATLAB command or function. You can check to see if a command, function or file name already exists by using the `exist` command.

M-files

- To create an M-file:
Go to File \implies New

An example of an m-file is shown below:



The screenshot shows the MATLAB Command Window interface. The menu bar includes File, Edit, Debug, Desktop, Window, and Help. The toolbar contains icons for file operations and a 'Current Directory' field showing 'c:\MyMATLABFiles'. Below the toolbar are shortcuts for 'How to Add' and 'What's New'. The Command Window displays the following MATLAB code being executed:

```
>> x = 0:0.01:7;  
>> y = 3*cos(2*x);  
>> plot(x,y),xlabel('x'),ylabel('y')  
fx >>
```

Comments in Matlab

The comment symbol may be put anywhere in the line. MATLAB ignores everything to the right of the % symbol. For example,

```
>>% This is a comment.  
>>x = 2+3 % So is this.  
x =  
    5
```

Note that the portion of the line before the % sign is executed to compute x.

Comments in Matlab

Commenting is used to organize the code and making it easy to understand the code later.

The comment symbol may be put anywhere in the line. MATLAB ignores everything to the right of the % symbol. For example,

```
1      % Program to solve Example 3-1
2
3      %Inputs
4 -    m=5; % mass
5 -    c=0; %damping
6 -    k=16; %stiffness
```

Note that the portion of the line before the % sign is executed.

The beginning of the program

We will start our program by clearing the variables from the memory, clearing any previous graph and clearing the command windows (If an error occurs, it will appear right away in the command window)

```
1 % Example 1
2 - clear all; clf; clc
3
```

How does Matlab interact when we write a text in the command window?

When you type `problem1` in command window,

1. MATLAB first checks to see if `problem1` is a variable and if so, displays its value.
2. If not, MATLAB then checks to see if `problem1` is one of its own commands, and executes it if it is.
3. If not, MATLAB then looks in the current directory for a file named `problem1.m` and executes `problem1` if it finds it.
4. If not, MATLAB then searches the directories in its search path, in order, for `problem1.m` and then executes it if found.

System, directory, and file commands

Table 1.3–2, Page 23

Command	Description
<code>addpath dirname</code>	Adds the directory <code>dirname</code> to the search path.
<code>cd dirname</code>	Changes the current directory to <code>dirname</code> .
<code>dir</code>	Lists all files in the current directory.
<code>dir dirname</code>	Lists all the files in the directory <code>dirname</code> .
<code>path</code>	Displays the MATLAB search path.
<code>pathtool</code>	Starts the Set Path tool.
<code>pwd</code>	Displays the current directory.
<code>rmpath dirname</code>	Removes the directory <code>dirname</code> from the search path.
<code>what</code>	Lists the MATLAB-specific files found in the current working directory. Most data files and other non-MATLAB files are not listed. Use <code>dir</code> to get a list of all files.
<code>what dirname</code>	Lists the MATLAB-specific files in directory <code>dirname</code> .