|  |
| --- |
| * <MG2FN> |

Part I Review of MATLAB

# Introduction

|  |
| --- |
| * [MG2FN\Ch01](MG2FN/Ch01) |

In this introductory chapter a short MATLAB tutorial is provided.

This tutorial describes the basic MATLAB commands needed.

Note that there are numerous free MATLAB tutorials on the internet – check references [1-28].

Also, you may want to consult many of the excellent books on the subject – check references [29- 47].

This chapter may be skipped on a first reading of the book.

In this tutorial it is assumed that you have started MATLAB on your computer system successfully and that you are now ready to type the commands at the MATLAB prompt (which is denoted by double arrows “>>”).

For installing MATLAB on your computer system, check the web links provided at the end of the book.

# Symbolic Computing

|  |
| --- |
| * [MG2FN\Ch02](MG2FN/Ch02) |

In this chapter, the basics of symbolic computing are illustrated with MATLAB.

Symbolic computing means performing algebraic and other mathematical operations without resorting to numerical computations.

This type of computing is also called computer algebra.

Computer programs and software that are used in symbolic computations are called computer algebra systems.

Many such competing systems are available today such as MAPLE, MATHEMATICA, MACZYMA, etc.

It should be noted that MATLAB is not basically a computer algebra system but can be used for this purpose with the aid of the MATLAB Symbolic Math Toolbox.

This toolbox uses the MAPLE engine and helps in converting MATLAB into a computer algebra system.

Some of the operations that we will use in subsequent chapters can be easily implemented using symbolic computing.

Thus the use of the MATLAB Symbolic Math Toolbox will become essential in subsequent chapters.

Therefore, the use of this toolbox with various symbolic computing exercises will be illustrated briefly in this chapter.

Symbolic arithmetic operations may also be performed in MATLAB using the MATLAB Symbolic Math Toolbox.

For the next examples, this toolbox needs to be installed with MATLAB.

In order to perform symbolic operations in MATLAB, we need to define symbolic numbers[1] using the sym command.

For example, ½ is defined as a symbolic number as follows:

Notice in the above example how the final answer was cast in the form of a fraction without decimal digits and without calculating a numerical value.

In the addition of the two fractions above, MATLAB finds their common denominator and adds them by the usual procedure for rational numbers.

Notice also in the above four example outputs that symbolic answers are not indented but numerical answers are indented.

You may also use symbolic variables[2] in MATLAB.

For example, the variables x and y are defined as symbolic variables as follows:

It is seen thus that the value of the volume of the sphere is 41.2162 (no units are used in this example).

Finally, the variables x and this can be defined as symbolic variables without assigning a numerical value as follows:

## Solving Equations with the MATLAB Symbolic Math Toolbox

The MATLAB Symbolic Math Toolbox can be used to solve algebraic equations in MATLAB.

In addition, systems of linear and nonlinear algebraic equations can also be solved using this toolbox.

There are special commands in this toolbox for solving equations – in particular the MATLAB command solve.

Please note that the command roots that was discussed earlier in this chapter cannot be used for the symbolic solution of algebraic equations.

It needs to be replaced with the solve command.

The use of the MATLAB command solve will be illustrated in this section with several examples.

Consider the following linear algebraic symbolic equation that we need to solve for the variable x in terms ofthe constants a and b:

# Solving Equations

|  |
| --- |
| * [MG2FN\Ch03](MG2FN/Ch03) |

In this chapter we discuss how to solve algebraic equations using MATLAB.

We will discuss both linear and nonlinear algebraic equations.

We will also discuss systems of simultaneous linear and nonlinear algebraic equations.

First, let us solve the following simple linear equation for the variable x:

The solution ofthe above equation may be obvious to the reader but we will show how to solve it using MATLAB.

The easiest method to do it in MATLAB would be to consider the left-hand-side of the equation as a polynomial of first degree and then find the roots of the polynomial using the MATLAB command roots.

First, we write the above equation as a polynomial as follows:

We enter the coefficients of the above polynomial in a vector in MATLAB then use the roots command to find the root which is the solution to the above linear equation in x. Here are the needed commands:

## Solving Equations with the MATLAB Symbolic Math Toolbox

# MATLAB Functions

|  |
| --- |
| * [MG2FN\Ch04](MG2FN/Ch04) |

In this chapter we introduce the basics of programming in MATLAB.

The material presented in this chapter is not intended to be comprehensive or exhaustive but merely an introduction[6] to programming in MATLAB.

In MATLAB programming, the list of commands and instructions are usually stored in a text file called an M-file (short for MATLAB file).

These M-files can be of two kinds – either script files or function files.

These files can be created or opened from the File menu by clicking on New or Open, then clicking on M-File.

These files typically have the .m extension that is associated with MATLAB.

We will discuss script files first.

The examples presented are simple.

The reader can write more complicated examples based on the material presented here.

Script files are used to store MATLAB scripts.

A script is defined in MATLAB as a sequence of MATLAB commands.

A script can span several lines.

For example, here is a MATLAB script:

# Graphs in MATLAB

|  |
| --- |
| * [MG2FN\Ch05](MG2FN/Ch05) |

In this chapter we explain how to plot graphs in MATLAB.

Both two dimensional and three-dimensional graphs are presented.

A graph in MATLAB appears in its own window (not on the command line).

First, we will consider two-dimensional or planar graphs.

To plot a two-dimensional graph, we need two vectors.

Here is a simple example using two vectors x and y along with the MATLAB command plot:

Part II Fibonacci Numbersand The Golden Ratio

# Fibonacci Numbers

|  |
| --- |
| * [MG2FN\Ch06](MG2FN/Ch06) |

The Fibonacci sequence is a sequence of numbers where each term of the sequence is obtained by adding the previous two terms.

In addition, this special sequence starts with the numbers 1 and 1.

Thus, the whole Fibonacci sequence can be generated starting with these two numbers and the above special rule.

The numbers in this sequence are called Fibonacci numbers.

These numbers were first obtained by the Italian mathematician Leonardo of Pisa (also called Leonardo Fibonacci) in the thirteenth century.

Next, we will generate the first few terms of the Fibonacci sequence.

We will start with 1 and 1[19].

Adding these two terms brings us to 2 which is the third Fibonacci number.

Then we add 2 and 1 to obtain 3 which is the fourth Fibonacci number.

Next, we add 3 and 2 to obtain 5 which is the fifth Fibonacci number.

Then, we add 5 and 3 to obtain 8 which is the sixth Fibonacci number.

This is followed by adding 8 and 5 to obtain 13 which is the seventh Fibonacci number.

This process is illustrated interactively in MATLAB as follows:

# The Golden Ratio

|  |
| --- |
| * [MG2FN\Ch07](MG2FN/Ch07) |

In this chapter, we will continue our study of the Fibonacci sequence and illustrate its precise relationship to what is called the Golden Ratio.

Let us calculate the ratio of each two consecutive numbers in the Fibonacci sequence.

Here are the first few terms of the Fibonacci sequence: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ……

Let us start by calculating the ratio of the first two terms.

The first two terms are 1 and 1; their ratio is 1/1 = 1.

The next two consecutive terms are 1 and 2; their ratio is 2/1 = 2.

Next are the numbers 2 and 3; their ratio is 3/2 = 1.5.

The ratio of 3 and 5 is 5/3 = 1.6667.

Next are the numbers 5 and 8 whose ratio is 8/5 = 1.6.

This followed by the numbers 8 and 13 whose ratio is 13/8 = 1.625.

Next are 13 and 21 whose ratio is 21/13 = 1.6154.

This is followed by the numbers 21 ad 34; their ratio is 34/21 = 1.6190.

Next are the numbers 34 and 55 whose ratio is 55/34 = 1.6176.

The ratio of the next two numbers in the sequence is 89/55 = 1.6182.

Finally, the last two terms in the sequence above have the ratio 144/89 = 1.6180.

As can be seen from the above calculations, the ratio of any two consecutive Fibonacci numbers seems to approach the number 1.6180 as the numbers increase.

In order to ascertain this observation, the above calculations are easily implemented using MATLAB.

First, let us perform the above calculations again using MATLAB interactively as follows:

# Properties of the Golden Ratio

|  |
| --- |
| * [MG2FN\Ch08](MG2FN/Ch08) |

In this chapter, we will study certain properties of the Golden Ratio in detail.

In addition, we will study a nice formula that is used to compute Fibonacci numbers using the value of the Golden Ratio.

First, let us calculate the inverse of the Golden Ratio , i.e. let us calculate numerically the value of .

This computation is performed in MATLAB as follows:

Next, Next, we execute the above MATLAB function several times and compare its results with those of the exact function fib(n) as follows (we see that we obtain exact results for higher values of n):or higher values of n):

Thus, for the tenth Fibonacci number, we obtain almost an exact result.

We obtain also an almost exact result for the 30th Fibonacci number as shown above.

The larger the value of n, the more accurate is the Fibonacci number which is obtained.

In the next chapter, we study another sequence of interesting numbers that is closely related to the Fibonacci sequence.

# Lucas Numbers

|  |
| --- |
| * [MG2FN\Ch09](MG2FN/Ch09) |

In this chapter we will study Lucas numbers and the Lucas sequence of numbers.

These numbers are closely associated with Fibonacci numbers.

Lucas numbers are generated using the same rule as that of Fibonacci numbers, i.e. the next number is obtained by adding the previous two numbers.

However, instead of starting with the numbers 1 and 1, we start with the numbers 2 and 1.

Thus the third Lucas number is 3.

The fourth Lucas number is 4.

The fifth Lucas number is 7.

Thus, the first few Lucas numbers are 2, 1, 3, 4, 7, 11, 18, 29, 47, 76, 123, ….

This sequence of numbers is called the Lucas sequence.

These numbers are generated using MATLAB interactively as follows:

p118.

As expected, the slope of the line is almost as before.

The angle of the line is calculated as tan-1(0.4812) = 25.7o, almost exactly as before.

It should also be noted that the slope and angle of the Lucas numbers are exactly the same as the slope and angle of Fibonacci numbers (see Chapter 6).

It seems that this slope (or angle) is a constant that is associated with these types of sequences.

An alternative method to plot the logarithms of Lucas numbers is to use the MATLAB command semilogy in association with the MATLAB function lucas(n).

The graph obtained in this way will also be an almost straight line but will be slightly different from the graphs obtained here.

# Generalizations of Fibonacci Numbers

# Random Fibonacci Numbers References

# MATLAB Books

* Gilat, A., MATLAB: An Introduction with Applications, Second Edition, John Wiley & Sons, 2004.
* Pratap, R., Getting Started with MATLAB 7: An Introduction for Scientists and Engineers, Oxford University Press, 2005.
* Hanselmann, D. and Littlefield, B., Mastering MATLAB 7, Prentice Hall, 2004.
* Palm, W., Introduction to MATLAB 7 for Engineers, McGraw-Hill, 2004.
* Moore, H., MATLAB for Engineers, Prentice Hall, 2006.
* Chapman, S., MATLAB Programming for Engineers, Thomson Engineering, 2004.
* Davis, T. and Sigmon, K., MATLAB Primer, Seventh Edition, Chapman & Hall, 2004.
* Higham, D. and Higham, N., MATLAB Guide, Second Edition, SIAM, 2005.
* King, J., MATLAB for Engineers, Addison-Wesley, 1988.
* Etter, D., Introduction to MATLAB for Engineers and Scientists, Prentice Hall, 1995.
* Magrab, E. et al., An Engineer’s Guide to MATLAB, Prentice Hall, 2000.
* Etter, D., Kuncicky, D. and Hull, D., Introduction to MATLAB 6, Prentice Hall, 2001.
* Recktenwald, G., Introduction to Numerical Methods and MATLAB: Implementation and Applications, Prentice Hall, 2001.
* Biran, A. and Breiner, M., MATLAB 5 for Engineers, Addison-Wesley, 1999.
* Part-Enander, E. and Sj oberg, A., The MATLAB 5 Handbook, Addison-Wesley, 1999.
* Etter, D., Engineering Problem Solving with MATLAB, Prentice Hall, 1993.
* Chen, K., Giblin, P, and Irving, A., Mathematical Explorations with MATLAB, Cambridge University Press, 1999.
* Mathews, J. and Fink, K., Numerical Methods Using MATLAB, Third Edition, Prentice Hall, 1999.
* Fausett, L., Applied Numerical Analysis Using MATLAB, Prentice Hall, 1999.

# Books on Fibonacci Numbers

* Posamentier, A. S., The Fabulous Fibonacci Numbers, Prometheus Books, 2007.
* Vaj da, S., Fibonacci and Lucas Numbers, and the Golden Section, Dover Publications, 2007.
* Vorobiev, N. N. and Martin, M., Fibonacci Numbers, Birkhauser, 2003.
* Dunlap, R. A., The Golden Ratio and Fibonacci Numbers, World Scientific Publishing, 1997.
* Olsen, S., The Golden Section: Nature’s Greatest Secret, Walker and Company, 2006.
* Livio, M., The Golden Ratio: The Story of PHI, the World’s Most Astonishing Number, Broadway, 2003.
* Koshy, T., Fibonacci and Lucas Numbers with Applications, Wiley -Interscience, 2001.
* Dobson, E. D., Understanding Fibonacci Numbers, Traders Press, 1994.
* Hoggatt, V. E. and Bicknell, M., Primer on Fibonacci Numbers, Fibonacci Association, 1972.

# Web Links for Fibonacci Numbers

* Fibonacci Numbers at Wolfram MathWorld

<http://mathworld.wolfram.com/FibonacciNumber.html>

* Fibonacci Numbers at Wikipedia

<http://en.wikipedia.org/wiki/Fibonacci_number>

* Fibonacci Numbers and the Golden Section

<http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fib.html>

* The Mathematical Magic of Fibonacci Numbers

<http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fibmaths.html>

* The Fibonacci Sequence – Math is Fun

<http://www.mathsisfun.com/numbers/fibonacci-sequence.html>

* Generalizations of Fibonacci Numbers from Wikipedia

<http://en.wikipedia.org/wiki/Tribonacci_number>

* Random Fibonacci Sequences from Wikipedia

<http://en.wikipedia.org/wiki/Viswanath%27s_constant>

* Broken link
* The Golden Mean and the Phy sics of Aesthetics by Subhash Kak

[http://arxiv.org/abs/phy sics/0411195](http://arxiv.org/abs/phy%20sics/0411195)

* Stepping Bey ond Fibonacci Numbers – from Ivars Peterson’s MathTrek

<http://www.maa.org/mathland/mathtrek_09_30_02.html>

* Viswanath’s Constant from PlanetMath

<http://planetmath.org/encyclopedia/ViswanathsConstant.html>

* Divakar Viswanath, “Random Fibonacci Sequences and the Number 1.13198824…”, Mathematics of Computation, Vol. 69, 2000, pp. 1131-1155.

# Installation of MATLAB

# Footnotes

#