Numerical Analysis

Using MATLAB and Spreadsheets

Steven T. Karris

Orchard Publications
Fremont, California

Numerical Analysis

using MATLAB and Spreadsheets

Steven T. Karris

Students and working professionals will find ***Numerical Analysis using MATLAB and Spreadsheets***, to be a concise and easy-to-learn text. It provides complete, clear, and detailed explanations of the principal numerical analysis methods and well known functions used in science and engineering. These are illustrated with many practical examples.

This text includes the following chapters:
• Introduction to MATLAB • Root Approximations and Partial Fraction Expansion • Sinusoids and Complex Numbers • Matrices and Determinants • Review of Differential Equations • Power Series • Finite Differences and Interpolation • Linear and Parabolic Regression • Solution of Differential Equations by Numerical Methods • Integration by Numerical Methods • Difference Equations • The Gamma and Beta Functions • Bessel, Legendre, and Chebyshev Polynomials • Optimization Methods

Each chapter contains numerous practical applications supplemented with detailed instructions for using MATLAB and/or Microsoft Excel to obtain quick solutions.

Steven T. Karris is the president and founder of Orchard Publications. He earned a bachelors degree in electrical engineering at Christian Brothers University, Memphis, Tennessee, a masters degree in electrical engineering at Florida Institute of Technology, Melbourne Florida, and has done post-master work at the latter. He is a registered professional engineer in California and Florida. He has over 30 years of professional engineering experience in industry. In addition, he has over 25 years of teaching experience that he acquired at several educational institutions as an adjunct professor. He is currently with UC Berkeley Extension.

Orchard Publications, Fremont, California

Visit us on the Internet
www.orchardpublications.com
or email us: info@orchardpublications.com

ISBN 0-9709511-1-6

$42.95 U.S.A.

ISBN 0-9709511-1-6

Table of Contents

Chapter 16

Optimization Methods

Chapter 1

Introduction to MATLAB

This chapter is an introduction of the basic MATLAB commands and functions, procedures for naming and saving the user generated files, comment lines, access to MATLAB's Editor/Debugger, finding the roots of a polynomial, and making plots. Several examples are provided with detailed explanations.

1.1 Command Window

To distinguish the screen displays from the user commands, important terms and MATLAB functions, we will use the following conventions:

Click: Click the left button of the mouse

`Courier Font:` Screen displays

Helvetica Font: User inputs at MATLAB's command window prompt `EDU>>`[*]

Helvetica Bold: MATLAB functions

Times Bold Italic: Important terms and facts, notes, and file names

When we first start MATLAB, we see the toolbar on top of the *command screen* and the prompt `EDU>>`. This prompt is displayed also after execution of a command; MATLAB now waits for a new command from the user. We can use the *Editor/Debugger* to write our program, save it, and return to the command screen to execute the program as explained below.

To use the Editor/Debugger:

1. From the *File* menu on the toolbar, we choose *New* and click on *M-File*. This takes us to the *Editor Window* where we can type our *code* (list of statements) for a new file, or open a previously saved file. We must save our program with a file name which starts with a letter. ***Important!*** MATLAB is *case sensitive*, that is, it distinguishes between upper- and lower-case letters. Thus, *t* and *T* are two different characters in MATLAB language. The files that we create are saved with the file name we use and the extension ***.m***; for example, ***myfile01.m***. It is a good practice to save the code in a file name that is descriptive of our code content. For instance, if the code performs some matrix operations, we ought to name and save that file as ***matrices01.m*** or any other similar name. We should also use a separate disk to backup our files.

* *EDU>> is the MATLAB prompt in the Student Version.*

Example 1.12

The volume V of a right circular cone of radius r and height h is given by

$$V = \frac{1}{3}\pi r^2 h \tag{1.14}$$

Plot the volume of the cone as r and h vary on the intervals $0 \le r \le 4$ and $0 \le h \le 6$ meters.

Solution:

The volume of the cone is a function of both the radius r and the height h, that is, $V = f(r, h)$

The three-dimensional plot is created with the following MATLAB code where, as in the previous example, in the second line we have used the dot multiplication, division, and exponentiation. As mentioned in the footnote of the previous page, this topic will be explained in Section 1.8.

```
[R,H]=meshgrid(0: 4, 0: 6);                    % Creates R and H matrices from vectors r and h
V=(pi .* R .^ 2 .* H) ./ 3;  mesh(R, H, V)
xlabel('x-axis, radius r (meters)'); ylabel('y-axis, altitude h (meters)');
zlabel('z-axis, volume (cubic meters)'); title('Volume of Right Circular Cone'); box on
```

The three-dimensional plot of Figure 1.6, shows how the volume of the cone increases as the radius and height are increased.

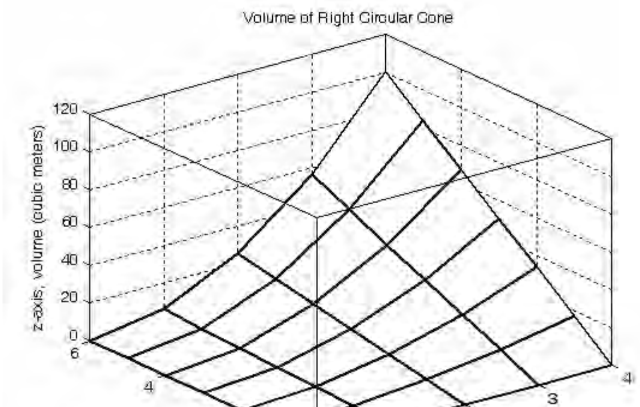


Figure 1.6. Volume of a right circular cone.

This, and the plot of Figure 1.5, are rudimentary; MATLAB can generate very sophisticated and impressive three-dimensional plots. The MATLAB User's manual contains more examples.

Chapter 2

Root Approximations

This chapter is an introduction to Newton's and bisection methods for approximating roots of linear and non-linear equations. Several examples are presented to illustrate practical solutions using MATLAB and spreadsheets.

2.1 Newton's Method for Root Approximation

Newton's (*or Newton-Raphson*) method can be used to approximate the roots of any linear or non-linear equation of any degree. This is an iterative (repetitive procedure) method and it is derived with the aid of Figure 2.1.

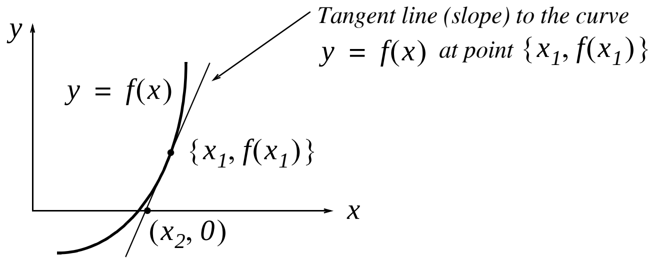

Tangent line (slope) to the curve
$y = f(x)$ at point $\{x_1, f(x_1)\}$
y
$y = f(x)$
$\{x_1, f(x_1)\}$
x
$(x_2, 0)$

Figure 2.1. Newton's method for approximating real roots of a function

We assume that the slope is neither zero nor infinite. Then, the slope (first derivative) at $x = x_1$ is

$$f'(x_1) = \frac{y - f(x_1)}{x - x_1}$$

$$y - f(x_1) = f'(x_1)(x - x_1) \tag{2.1}$$

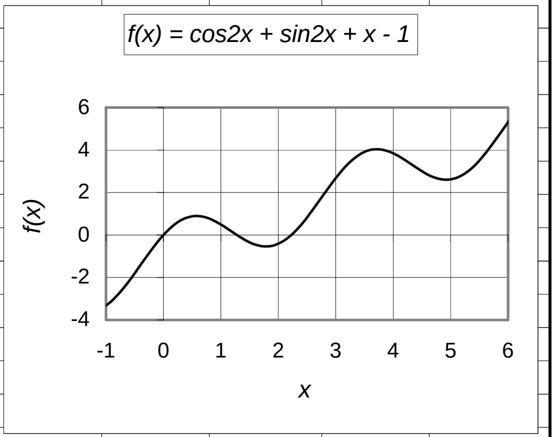
The slope crosses the $x - axis$ at $x = x_2$ and $y = 0$. Since this point $[x_2, f(x_2)] = (x_2, 0)$ lies on the slope line, it satisfies (2.1). By substitution,

$$0 - f(x_1) = f'(x_1)(x_2 - x_1)$$

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} \tag{2.2}$$

and in general,

$$\boxed{x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}} \tag{2.3}$$



	A	B	C	D	E	F	G
1	x	f(x)					
2	-1.00	-3.325					
3	-0.90	-3.101					
4	-0.80	-2.829					
5	-0.70	-2.515					
6	-0.60	-2.170					
7	-0.50	-1.801					
8	-0.40	-1.421					
9	-0.30	-1.039					
10	-0.20	-0.668					
11	-0.10	-0.319					
12	0.00	0.000					
13	0.10	0.279					
14	0.20	0.510					
15	0.30	0.690				x	f(x)
16	0.40	0.814		Real Root at	→	0.00	0.000
17	0.50	0.882		Real Root between		1.20	0.138
18	0.60	0.894				1.30	-0.041
19	0.70	0.855		Real Root between		2.20	-0.059
20	0.80	0.770				2.30	0.194

Figure 2.8. Graph for Example 2.5

We can obtain more accurate approximations using Excel's ***Goal Seek*** feature. We use *Goal Seek* when we know the desired result of a single formula, but we do not know the input value which satisfies that result. Thus, if we have the function $y = f(x)$, we can use *Goal Seek* to set the dependent variable y to the desired value (goal) and from it, find the value of the independent variable x which satisfies that goal. In the last three examples our goal was to find the values of x for which $y = f(x) = 0$.

To illustrate the *Goal Seek* feature, we will use it to find better approximations for the non-zero roots of Example 2.5. We do this with the following steps:

1. We copy range A24:B24 (or A25:B25) to two blank cells, say J1 and K1, so that J1 contains 1.20 and K1 contains 0.138 (or 1.30 and –0.041 if range A25:B25 was copied). We increase the accuracy of Columns J and K to 5 decimal places by clicking on *Format, Cells, Numbers* tab.
2. From the *Tools* drop menu, we *click* on *Goal Seek*, and when the *Goal Seek* dialog box appears, we make the following entries:

   Set cell: K1
   To value: 0

Chapter 3

Sinusoids and Phasors

This chapter is an introduction to alternating current waveforms. The characteristics of sinusoids are discussed and the frequency, phase angle, and period are defined. Voltage and current relationships are expressed in sinusoidal terms. Phasors which are rotating vectors in terms of complex numbers are also introduced and their relationships to sinusoids are derived.

3.1 Alternating Voltages and Currents

The waveforms shown in Figure 3.1 may represent alternating currents or voltages.

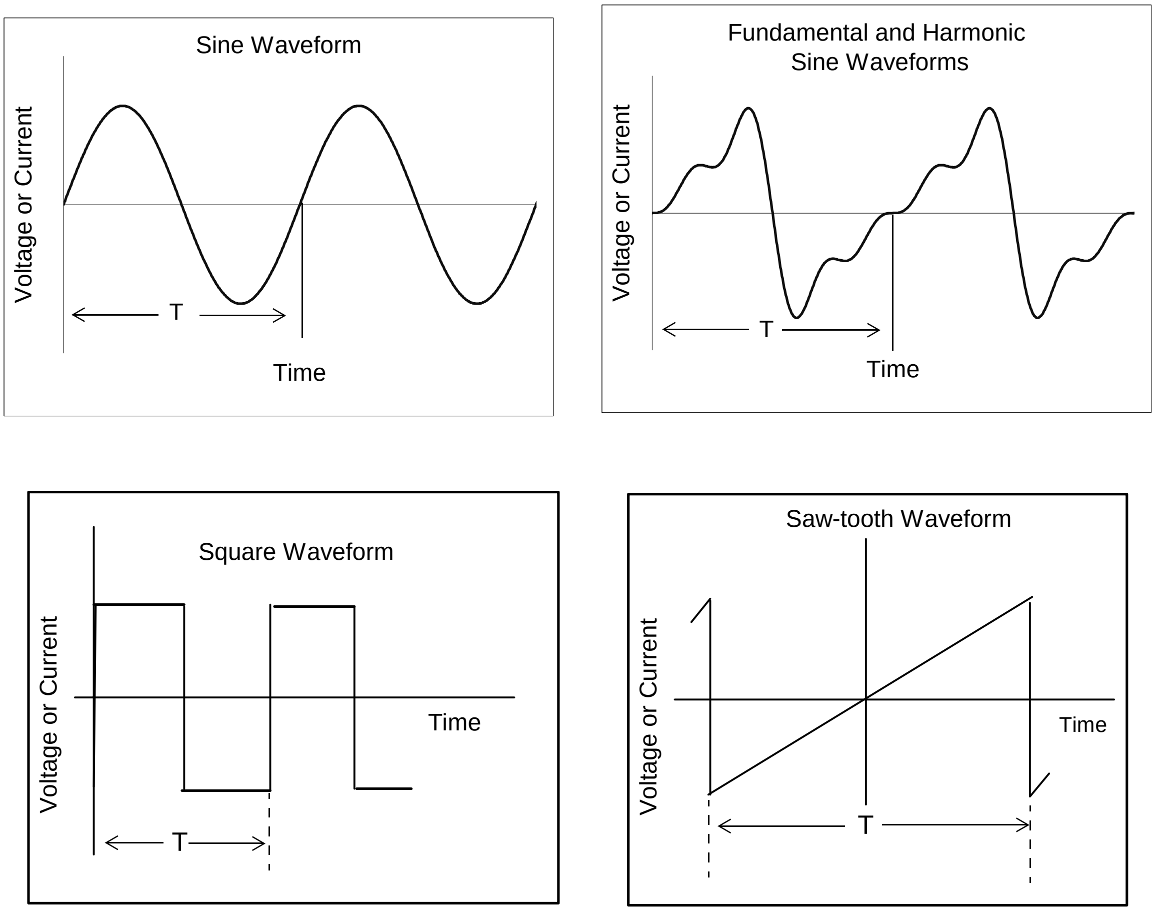


Figure 3.1. Examples of alternating voltages and currents

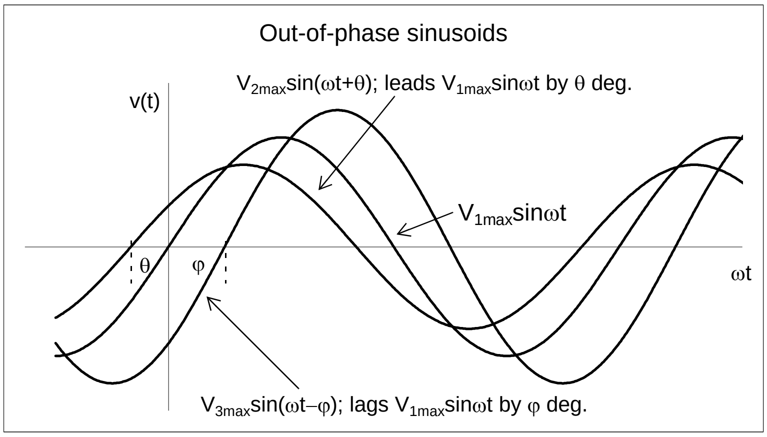

Out-of-phase sinusoids
V2maxsin(ωt+θ); leads V1maxsinωt by θ deg.
v(t)
V1maxsinωt
θ
φ
ωt
V3maxsin(ωt−φ); lags V1maxsinωt by φ deg.

Figure 3.4. Out-of-phase sinusoids

We must remember that when we say that one sinusoid *leads* or *lags* another sinusoid, these are of the same frequency. Obviously, two sinusoids of different frequencies can never be in phase.

It is convenient to express the phase angle in degrees rather than in radians in a sinusoidal function. For example, it is acceptable to express

$$v(t) = 100\sin(2000\pi t - \pi/6)$$

as

$$v(t) = 100\sin(2000\pi t - 30°)$$

since the subtraction inside the parentheses needs not to be performed.

When two sinusoids are to be compared in terms of their phase difference, these must first be written either both as cosine functions, or both as sine functions, and should also be written with positive amplitudes. We should remember also that *a negative amplitude implies $180°$ phase shift.*

Example 3.1

Find the phase difference between the sinusoids

$$i_1 = 120\cos(100\pi t - 30^\circ)$$

and

$$i_2 = -6\sin(100\pi t - 30^\circ)$$

Solution:

We recall that the minus (–) sign indicates a $\pm 180°$ phase shift, and that the sine function lags the cosine by $90°$. Then,

$$-\sin x = \sin(x \pm 180^\circ) \text{ and } \sin x = \cos(x - 90^\circ)$$

Chapter 4

Matrices and Determinants

This chapter is an introduction to matrices and matrix operations. Determinants, Cramer's rule, and Gauss's elimination method are introduced. Some definitions and examples are not applicable to subsequent material presented in this text, but are included for subject continuity, and reference to more advance topics in matrix theory. These are denoted with a dagger (†) and may be skipped.

4.1 Matrix Definition

A *matrix* is a rectangular array of numbers such as those shown below.

$$\begin{bmatrix} 2 & 3 & 7 \\ 1 & -1 & 5 \end{bmatrix} \qquad or \qquad \begin{bmatrix} 1 & 3 & 1 \\ -2 & 1 & -5 \\ 4 & -7 & 6 \end{bmatrix}$$

In general form, a matrix A is denoted as

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{bmatrix} \tag{4.1}$$

The numbers a_{ij} are the *elements* of the matrix where the index i indicates the row, and j indicates the column in which each element is positioned. Thus, a_{43} indicates the element positioned in the fourth row and third column.

A matrix of m rows and n columns is said to be of $m \times n$ *order matrix*.

If $m = n$, the matrix is said to be a *square matrix of order* m (or n). Thus, if a matrix has five rows and five columns, it is said to be a square matrix of order 5.

In a square matrix, the elements $a_{11}, a_{22}, a_{33}, \dots, a_{nn}$ are called the *main diagonal elements*. Alternately, we say that the matrix elements $a_{11}, a_{22}, a_{33}, \dots, a_{nn}$, are located on the *main diagonal*.

Example 4.8

Compute the determinant of A using the elements of the first row.

$$A = \begin{bmatrix} 1 & 2 & -3 \\ 2 & -4 & 2 \\ -1 & 2 & -6 \end{bmatrix} \tag{4.25}$$

Solution:

$$detA = 1\begin{bmatrix} -4 & 2 \\ 2 & -6 \end{bmatrix} - 2\begin{bmatrix} 2 & 2 \\ -1 & -6 \end{bmatrix} - 3\begin{bmatrix} 2 & -4 \\ -1 & 2 \end{bmatrix} = 1 \times 20 - 2 \times (-10) - 3 \times 0 = 40$$

Check with MATLAB:

```
A=[1  2  –3; 2  –4  2; –1  2  –6];  det(A)      %  Define matrix A and compute detA
```

```
ans =
    40
```

We must use the above procedure to find the determinant of a matrix A of order 4 or higher. Thus, a fourth-order determinant can first be expressed as the sum of the products of the elements of its first row by its cofactor as shown below.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = a_{11}\begin{bmatrix} a_{22} & a_{23} & a_{24} \\ a_{32} & a_{33} & a_{34} \\ a_{42} & a_{43} & a_{44} \end{bmatrix} - a_{21}\begin{bmatrix} a_{12} & a_{13} & a_{14} \\ a_{32} & a_{33} & a_{34} \\ a_{42} & a_{43} & a_{44} \end{bmatrix} \tag{4.26}$$

$$+a_{31}\begin{bmatrix} a_{12} & a_{13} & a_{14} \\ a_{22} & a_{23} & a_{24} \\ a_{42} & a_{43} & a_{44} \end{bmatrix} - a_{41}\begin{bmatrix} a_{12} & a_{13} & a_{14} \\ a_{22} & a_{23} & a_{24} \\ a_{32} & a_{33} & a_{34} \end{bmatrix}$$

Determinants of order five or higher can be evaluated similarly.

Example 4.9

Compute the value of the determinant

$$A = \begin{bmatrix} 2 & -1 & 0 & -3 \\ -1 & 1 & 0 & -1 \\ 4 & 0 & 3 & -2 \\ -3 & 0 & 0 & 1 \end{bmatrix} \tag{4.27}$$

Chapter 5

Differential Equations, State Variables, and State Equations

This chapter is a review of ordinary differential equations and an introduction to state variables and state equations. Solutions of differential equations with numerical methods will be discussed in Chapter 9.

5.1 Simple Differential Equations

In this section we present two simple examples to show the importance of differential equations in engineering applications.

Example 5.1

The current and voltage in a capacitor are related by

$$i_C(t) = C\frac{dv_C}{dt} \tag{5.1}$$

where $i_C(t)$ is the current through the capacitor, $v_C(t)$ is the voltage across the capacitor, and the constant C is the capacitance in farads (F). For this example $C = 1\ F$ and the capacitor is being charged by a constant current I. Find the voltage v_C across this capacitor as a function of time given that the voltage at some reference time $t = 0$ is V_0.

Solution:

It is given that the current, as a function of time, is constant, that is,

$$i_C(t) = I = constant \tag{5.2}$$

By substitution of (5.2) into (5.1) we get

$$\frac{dv_C}{dt} = I$$

and by separation of the variables,

$$dv_C = Idt \tag{5.3}$$

Integrating both sides of (5.3) we get

$$v_C(t) = It + k \tag{5.4}$$

where k represents the constants of integration of both sides.

$$y(t) = y_N + y_F = k_1 e^{-t} + k_2 e^{-3t} - 3e^{-2t} \tag{5.40}$$

The constants k_1 and k_2 are evaluated from the given initial conditions. For this example,

$$y(0) = 1 = k_1 e^0 + k_2 e^0 - 3e^0$$

or

$$k_1 + k_2 = 4 \tag{5.41}$$

Also,

$$y'(0) = -1 = \left.\frac{dy}{dt}\right|_{t=0} = \left.-k_1 e^{-t} - 3k_2 e^{-3t} + 6e^{-2t}\right|_{t=0}$$

or

$$-k_1 - 3k_2 = -7 \tag{5.42}$$

Simultaneous solution of (5.41) and (5.42) yields $k_1 = 2.5$ and $k_2 = 1.5$. By substitution into (5.40), we get

$$y(t) = y_N + y_F = 2.5e^{-t} + 1.5e^{-3t} - 3e^{-2t} \tag{5.43}$$

Check with MATLAB:

```
y=dsolve('D2y+4*Dy+3*y=3*exp(-2*t)', 'y(0)=1', 'Dy(0)=-1')
```

```
y =
(-3*exp(-2*t)*exp(t)+3/2*exp(-3*t)*exp(t)+5/2)/exp(t)
```

```
pretty(y)
```

```
     -3 exp(-2 t) exp(t) + 3/2 exp(-3 t) exp(t) + 5/2
     ------------------------------------------------
                          exp(t)
```

```
ezplot(y,[0 8])
```

The plot is shown in Figure 5.2

Example 5.8

Find the total solution of the ODE

$$\frac{d^2y}{dt^2} + 6\frac{dy}{dt} + 9y = 0 \tag{5.44}$$

subject to the initial conditions $y(0) = -1$ and $y'(0) = 1$

Chapter 6

Fourier, Taylor, and Maclaurin Series

This chapter is an introduction to Fourier and power series. We begin with the definition of sinusoids that are harmonically related and the procedure for determining the coefficients of the trigonometric form of the series. Then, we discuss the different types of symmetry and how they can be used to predict the terms that may be present. Several examples are presented to illustrate the approach. The alternate trigonometric and the exponential forms are also presented. We conclude with a discussion on power series expansion with the Taylor and Maclaurin series.

6.1 Wave Analysis

The French mathematician Fourier found that any *periodic* waveform, that is, a waveform that repeats itself after some time, can be expressed as a series of harmonically related sinusoids, i.e., sinusoids whose frequencies are multiples of a *fundamental* frequency (or first harmonic). For example, a series of sinusoids with frequencies $1\ MHz$, $2\ MHz$, $3\ MHz$, and so on, contains the fundamental frequency of $1\ MHz$, a second harmonic of $2\ MHz$, a third harmonic of $3\ MHz$, and so on. In general, any periodic waveform $f(t)$ can be expressed as

$$\begin{aligned} f(t) &= \frac{1}{2}a_0 + a_1 \cos\omega t + a_2 \cos 2\omega t + a_3 \cos 3\omega t + a_4 \cos 4\omega t + \dots \\ &\quad + b_1 \sin\omega t + b_2 \sin 2\omega t + b_3 \sin 3\omega t + b_4 \sin 4\omega t + \dots \end{aligned} \tag{6.1}$$

or

$$f(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty}(a_n \cos n\omega t + b_n \sin n\omega t) \tag{6.2}$$

where the first term $a_0/2$ is a constant, and represents the DC (average) component of $f(t)$. Thus, if $f(t)$ represents some voltage $v(t)$, or current $i(t)$, the term $a_0/2$ is the average value of $v(t)$ or $i(t)$.

The terms with the coefficients a_1 and b_1 together, represent the fundamental frequency component ω[*]. Likewise, the terms with the coefficients a_2 and b_2 together, represent the second harmonic component 2ω, and so on.

* *We recall that* $k_1 \cos\omega t + k_2 \sin\omega t = k\cos(\omega t + \theta)$ *where* θ *is a constant.*

$$f_n(x) = 1 + 2\left(x - \frac{\pi}{4}\right) + 2\left(x - \frac{\pi}{4}\right)^2 + \dots \tag{6.130}$$

We can also obtain a Taylor series expansion with the MATLAB **taylor(f,n,a)** function where **f** is a symbolic expression, **n** produces the first n terms in the series, and **a** defines the Taylor approximation about point a. A detailed description can be displayed with the **help taylor** command. For example, the following code will compute the first 8 terms of the Taylor series expansion of $y = f(x) = tan x$ about $a = \pi/4$.

```
x=sym('x'); y=tan(x); z=taylor(y,8,pi/4); pretty(z)
```

```
                                 2                   3                      4
1 + 2x - 1/2 pi + 2(x - 1/4 pi) + 8/3(x - 1/4 pi) + 10/3(x - 1/4 pi)

       64             5   244             6   2176             7
     + -- (x - 1/4 pi)  + --- (x - 1/4 pi)  + ---- (x - 1/4 pi)
       15                 45                  315
```

Example 6.13

Express the function

$$y = f(t) = e^t \tag{6.131}$$

in a Maclaurin's series.

Solution:

A Maclaurin's series has the form of (6.132), that is,

$$f(x) = f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \dots + \frac{f^{(n)}(0)}{n!}x^n \tag{6.132}$$

For this function, we have $f(t) = e^t$ and thus $f(0) = 1$. Since all derivatives are e^t, then, $f'(0) = f''(0) = f'''(0) = \dots = 1$ and therefore,

$$f_n(t) = 1 + t + \frac{t^2}{2!} + \frac{t^3}{3!} + \dots \tag{6.133}$$

MATLAB displays the same result.

```
t=sym('t'); fn=taylor(exp(t)); pretty(fn)
```

```
                 2         3          4           5
      1 + t + 1/2 t  + 1/6 t  + 1/24 t  + 1/120 t
```

Chapter 7

Finite Differences and Interpolation

This chapter begins with finite differences and interpolation which is one of its most important applications. Finite Differences form the basis of numerical analysis as applied to other numerical methods such as curve fitting, data smoothing, numerical differentiation, and numerical integration. We will discuss these applications in this and the next three chapters.

7.1 Divided Differences

Consider the continuous function $y = f(x)$ and let $x_0, x_1, x_2, \ldots, x_{n-1}, x_n$ be some values of x in the interval $x_0 \le x \le x_n$. It is customary to show the independent variable x, and its corresponding values of $y = f(x)$ in tabular form as in Table 7.1.

TABLE 7.1 The variable x and $y = f(x)$ in tabular form

x	$f(x)$
x_0	$f(x_0)$
x_1	$f(x_1)$
x_2	$f(x_2)$
$\ldots$	$\ldots$
x_{n-1}	$f(x_{n-1})$
x_n	$f(x_n)$

Let x_i and x_j be any two, not necessarily consecutive values of x, within this interval. Then, the *first divided difference* is defined as:

$$f(x_i, x_j) = \frac{f(x_i) - f(x_j)}{x_i - x_j} \tag{7.1}$$

Likewise, the *second divided difference* is defined as:

$$f(x_i, x_j, x_k) = \frac{f(x_i, x_j) - f(x_j, x_k)}{x_i - x_k} \tag{7.2}$$

Solution:

We let the origin be at $(x_0, y_0) = (0, 0)$, and the plot in the intervals $-10 \le x \le 10$ and $-10 \le y \le 10$. Then, we write and execute the following code.

```
% This is the code for Example_7_14
x=–10: 0.25: 10;                    % Define interval in increments of 0.25
y=x;                                 % y must have same number of points as x
[X,Y]=meshgrid(x,y);                % Create X and Y matrices
Z=X.^3+Y.^3–3.*X.*Y;
mesh(X,Y,Z);                        % Generate mesh plot
xlabel('x'); ylabel('y'); zlabel('z');
title('Plot for the Function of Example 7.14');
z_int=interp2(X,Y,Z, -1,2,'cubic');
fprintf(' \n')
fprintf('Interpolated Value of z at x = –1 and y = 2 is z = %4.2f \n',z_int)
fprintf(' \n')
```

The plot for the function of this example is shown in Figure 7.10.

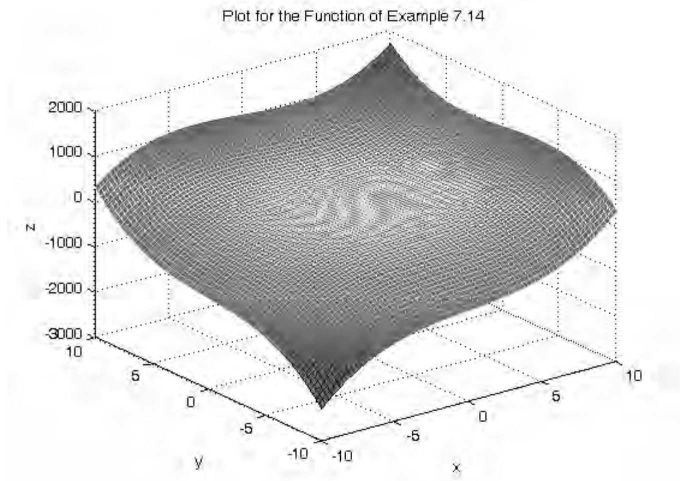


Figure 7.10. Plot for Example 7.14

```
Interpolated Value of z at x = -1 and y = 2 is z = 13.00
```

Example 7.15

A land surveyor measured and recorded the data below for a rectangular undeveloped land which lies approximately 500 meters above sea level.

```
500.08 500.15 500.05 500.08 500.14 500.13 500.09 500.15
500.12 500.01 500.11 500.18 500.15 500.12 500.05 500.15
```

Chapter 8

Linear and Parabolic Regression

This chapter is an introduction to regression and procedures for finding the best curve to fit a set of data. We will discuss linear and parabolic regression, and regression with power series approximations. We will illustrate their application with several examples.

8.1 Curve Fitting

Curve fitting is the process of finding equations to approximate straight lines and curves that best fit given sets of data. For example, for the data of Figure 8.1, we can use the equation of a straight line, that is,

$$y = mx + b \tag{8.1}$$

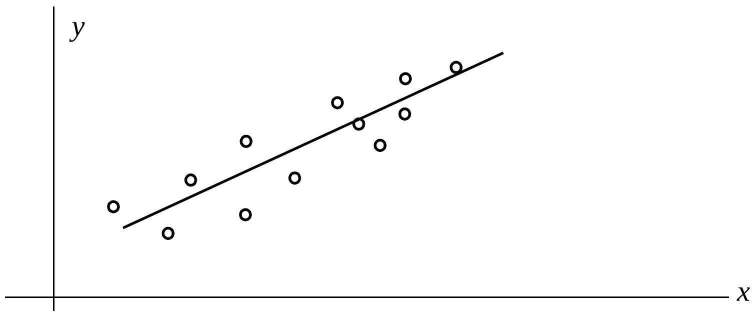


Figure 8.1. Straight line approximation.

For Figure 8.2, we can use the equation for the quadratic or parabolic curve of the form

$$y = ax^2 + bx + c \tag{8.2}$$

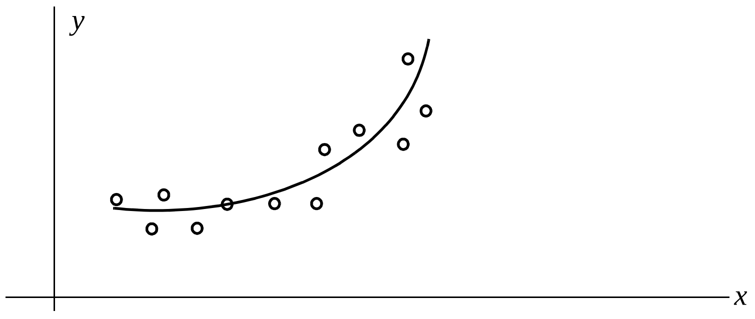


Figure 8.2. Parabolic line approximation

In finding the best line, we normally assume that the data, shown by the small circles in Figures 8.1 and 8.2, represent the independent variable x, and our task is to find the dependent variable

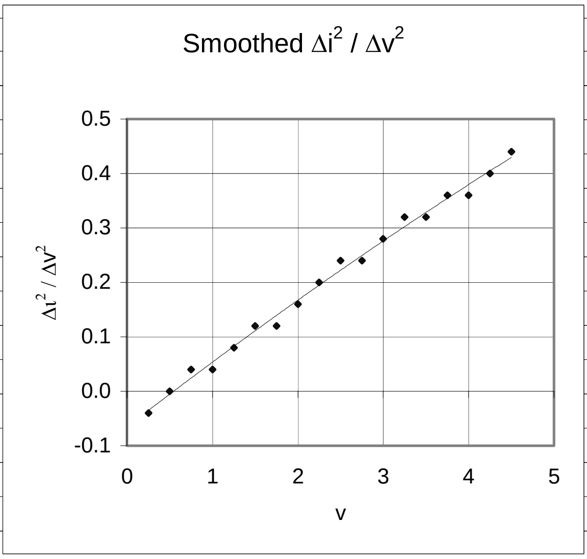
	A	B	C
1		**Computed**	
2	**Volts**	**$\Delta i^2 / \Delta v^2$**	
3			
4	0.00	-0.04	
5	0.25	-0.04	
6	0.50	0.00	
7	0.75	0.04	
8	1.00	0.04	
9	1.25	0.08	
10	1.50	0.12	
11	1.75	0.12	
12	2.00	0.16	
13	2.25	0.20	
14	2.50	0.24	
15	2.75	0.24	
16	3.00	0.28	
17	3.25	0.32	
18	3.50	0.32	
19	3.75	0.36	
20	4.00	0.36	From this plot, $(\Delta i^2 / \Delta v^2)\,|_{v=0} = i''(0) = -0.08$
21	4.25	0.40	
22	4.50	0.44	

Figure 8.14. Plot to obtain smoothed data of $\Delta i^2 / \Delta v^2$ in Example 8.6

```
%
plot(a,q); title('milliamps vs volts, n=3');...
xlabel('v'); ylabel('ma')                  % Plot the polynomial
% Display actual, smoothed and % error values
ma_smooth=polyval(p,v);                    % Calculate the values of the fitted polynomial
ma_exper = ma;
% The following statement computes the percent error between the
%  smoothed polynomial and the experimental (given) data
error=(ma_smooth-ma_exper).*100./(ma_exper+eps);
%
y=zeros(21,4);                             % Construct a 21 x 4 matrix of zeros
y(:,1)=v';                                 % 1st column of matrix
y(:,2)=ma_exper';                          % 2nd column of matrix
y(:,3)=ma_smooth';                         % 3rd column of matrix
y(:,4)=abs(error)';                        % 4th column of matrix
fprintf(' \n');                            % Insert line
% continued on the next page
```

Chapter 9

Solution of Differential Equations by Numerical Methods

This chapter is an introduction to several methods that can be used to obtain approximate solutions of differential equations. Such approximations are necessary when no exact solution can be found. The Taylor, Runge-Kutta, Adams', and Milne's methods are discussed.

9.1 Taylor Series Method

We recall from Chapter 6 that the Taylor series expansion about point a is

$$y_n = f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \ldots + \frac{f^{(n)}(a)}{n!}(x-a)^n \tag{9.1}$$

Now, if $x_1 > a$ is a value close to a, we can find the approximate value y_1 of $f(x_1)$ by using the first $k+1$ terms in the Taylor expansion of $f(x_1)$ about $x = a$. Letting $h_1 = x - a$ in (9.1), we get:

$$y_1 = y_0 + y'_0 h_1 + \frac{1}{2!}y''_0 h_1^2 + \frac{1}{3!}y'''_0 h_1^3 + \frac{1}{4!}y_0^{(4)} h_1^4 + \ldots \tag{9.2}$$

Obviously, to minimize the error $f(x_1) - y_1$ we need to keep h_1 sufficiently small.

For another value $x_2 > x_1$, close to x_1, we repeat the procedure with $h_2 = x_2 - x_1$; then,

$$y_2 = y_1 + y'_1 h_2 + \frac{1}{2!}y''_1 h_2^2 + \frac{1}{3!}y'''_1 h_2^3 + \frac{1}{4!}y_1^{(4)} h_2^4 + \ldots \tag{9.3}$$

In general,

$$\boxed{y_{i+1} = y_i + y'_i h_{i+1} + \frac{1}{2!}y''_i h_{i+1}^2 + \frac{1}{3!}y'''_i h_{i+1}^3 + \frac{1}{4!}y_i^{(4)} h_{i+1}^4 + \ldots} \tag{9.4}$$

Example 9.1

Use the Taylor series method to obtain a solution of

$$y' = -xy \tag{9.5}$$

correct to four decimal places for values $x_0 = 0.0$, $x_1 = 0.1$, $x_2 = 0.2$, $x_3 = 0.3$, $x_4 = 0.4$, and $x_5 = 0.5$ with the initial condition $y(0) = 1$.

For $x_5 = 0.5$ (9.18) yields

$$y = e^{-0.125} = 0.8825$$

and we observe that this value is in close agreement with the value of (9.17).

We can verify the analytical solution of Example 9.1 with MATLAB's dsolve(s) function using the following code:

```
syms x y z
z=dsolve('Dy=–x*y','y(0)=1','x')
```

```
z =
exp(-1/2*x^2)
```

The procedure used in this example, can be extended to apply to a second order differential equation

$$y'' = f(x, y, y') \tag{9.19}$$

In this case, we need to apply the additional formula

$$y'_{i+1} = y'_i + y''_i h + \frac{1}{2!}y'''_i h^2 + \frac{1}{3!}y^{(4)}_i h^3 + \dots \tag{9.20}$$

9.2 Runge-Kutta Method

The *Runge-Kutta method* is the most widely used method of solving differential equations with numerical methods. It differs from the Taylor series method in that we use values of the first derivative of $f(x, y)$ at several points instead of the values of successive derivatives at a single point.

For a Runge-Kutta method of order 2, the following formulas are applicable.

$$\boxed{\begin{gathered} k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + h, y_n + h) \\ y_{n+1} = y_n + \frac{1}{2}(k_1 + k_2) \\ \textit{For Runge-Kutta Method of Order 2} \end{gathered}} \tag{9.21}$$

When higher accuracy is desired, we can use order 3 or order 4. The applicable formulas are as follows:

Chapter 10

Integration by Numerical Methods

This chapter is an introduction to numerical methods for integrating functions which are very difficult or impossible to integrate using analytical means. We will discuss the trapezoidal rule that computes a function $f(x)$ with a set of linear functions, and Simpson's rule that computes a function $f(x)$ with a set of quadratic functions.

10.1 The Trapezoidal Rule

Consider the function $y = f(x)$ for the interval $a \le x \le b$, shown in Figure 10.1.

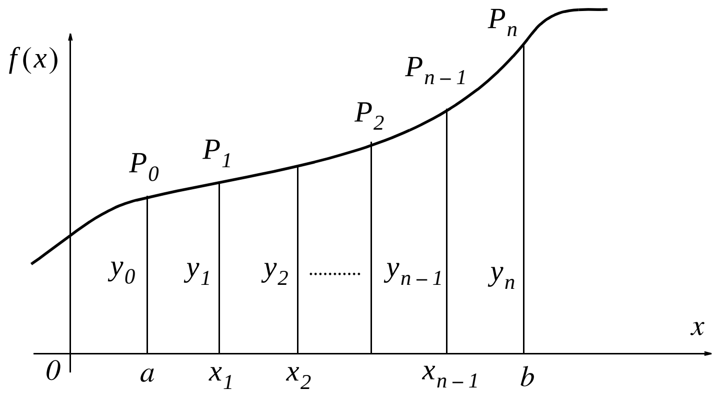


Figure 10.1. Integration by the trapezoidal rule

To evaluate the definite integral $\int_a^b f(x)dx$, we divide the interval $a \le x \le b$ into n subintervals each of length $\Delta x = \frac{b-a}{n}$. Then, the number of points between $x_0 = a$ and $x_n = b$ is $x_1 = a + \Delta x, x_2 = a + 2\Delta x, \ldots, x_{n-1} = a + (n-1)\Delta x$. Therefore, the integral from a to b is the sum of the integrals from a to x_1, from x_1 to x_2, and so on, and finally from x_{n-1} to b. The total area is

$$\int_a^b f(x)dx = \int_a^{x_1} f(x)dx + \int_{x_1}^{x_2} f(x)dx + \ldots + \int_{x_{n-1}}^{b} f(x)dx = \sum_{k=1}^{n} \int_{x_{k-1}}^{x_k} f(x)dx$$

The integral over the first subinterval, can now be approximated by the area of the trapezoid

$$\begin{aligned} y_0 &= \alpha h^2 - \beta h + \gamma && (a) \\ y_1 &= \gamma && (b) \\ y_2 &= \alpha h^2 + \beta h + \gamma && (c) \end{aligned} \qquad (10.13)$$

We can now evaluate the coefficients α, β, γ and express (10.12) in terms of h, y_0, y_1 and y_2. This is done with the following procedure.

By substitution of *(b)* of (10.13) into *(a)* and *(c)* and rearranging we get

$$\alpha h^2 - \beta h = y_0 - y_1 \qquad (10.14)$$

$$\alpha h^2 + \beta h = y_2 - y_1 \qquad (10.15)$$

Addition of (10.14) with (10.15) yields

$$2\alpha h^2 = y_0 - 2y_1 + y_2 \qquad (10.16)$$

and by substitution into (10.12) we get

$$Area|_{-h}^{h} = \frac{1}{3}h(2\alpha h^3 + 6\gamma) = \frac{1}{3}h[(y_0 - 2y_1 + y_2) + 6y_1] \qquad (10.17)$$

or

$$Area|_{-h}^{h} = \frac{1}{3}h(y_0 + 4y_1 + y_2) \qquad (10.18)$$

Now, we can apply (10.18) to successive segments of any curve $y = f(x)$ in the interval $a \le x \le b$ as shown on the curve of Figure 10.5.

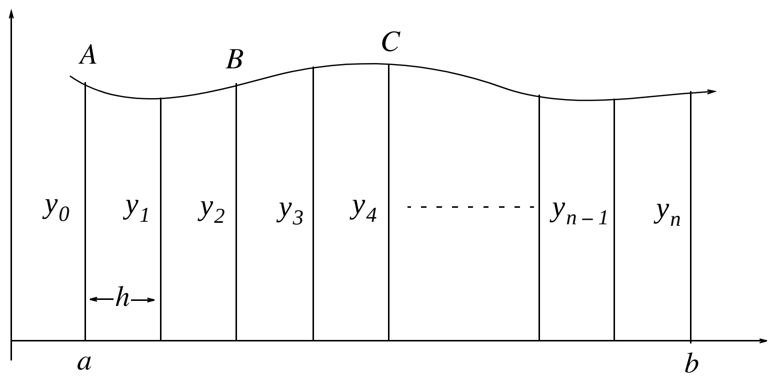


Figure 10.5. Simpson's rule of integration by successive segments

From Figure 10.5, we see that each segment of width $2h$ of the curve can be approximated by a

Chapter 11

Difference Equations

This chapter is an introduction to difference equations. The discussion is limited to linear difference equations with constant coefficients. The Fibonacci numbers are defined, and a practical example in electric circuit theory is given at the end of this chapter.

11.1 Definition, Solutions, and Applications

Difference equations are used in numerous applications such as engineering, mathematics, physics, and other sciences. A difference equation defines the relationship between the values y_k of a function, and the discrete set of the independent variables x_k. For example, the relation

$$\begin{aligned} y(n) + b_{1'}y(n-1) + b_{1'}y(n-1) + b_{2'}y(n-2) + \dots + b_{k'}y(n-k) \\ = a_0x(n) + a_{1'}x(n-1) + a_{2'}x(n-2) + \dots + a_{k'}x(n-k) \end{aligned} \tag{11.1}$$

is a linear difference equation with constant coefficients, and describes the relationship of a discrete input $x(n)$ and the corresponding discrete output $y(n)$ in a linear and time invariant* system. with constant coefficients a_i and b_i.

In (11.1), the difference order k was chosen to be the same on both sides. However, in most cases certain coefficients a_i and b_i are zero and thus, the order k for the left and right sides will not always be the same.

The general form of a linear, constant coefficient difference equation has the form

$$(a_0E^r + a_1E^{r-1} + a_{r-1}E + a_r)y = \phi(x) \tag{11.2}$$

where a_k represents a constant coefficient and E is an operator similar to the D operator in ordinary differential equations. The E operator increases the argument of a function by one interval h, and r is a positive integer that denotes the order of the difference equation.

In terms of the interval h, the difference operator E is

$$Ef(x_k) = f(x_k + h) = f(x_{k+h}) \tag{11.3}$$

* *A time invariant system is defined as one in which the input-output relationship does not change with time. That is, if an input x produces an output $y = f(x)$ at some time t_0, the same input x will produce the same output y at any other time. All systems in this text are assumed to be time invariant.*

11.2 Fibonacci Numbers

The *Fibonacci numbers* are solutions of the difference equation

$$y_{x+2} = y_{x+1} + y_x \tag{11.33}$$

that is, in a series of numbers, each number after the second, is the sum of the two preceding numbers.

Example 11.4

Given that $y_0 = 0$ and $y_1 = 1$, compute the first 12 Fibonacci numbers.

Solution:

For $x = 0, 1, 2, 3$ and so on, we obtain the Fibonacci numbers

$$1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, \ldots$$

We will conclude this chapter with an application to electric circuit analysis.

Example 11.5

For the electric network of Figure 11.1, derive an expression for the voltage V_x at each point P_x where $x = 0, 1, 2, \ldots, n$, given that the voltage V_0 at point P_0 is known.

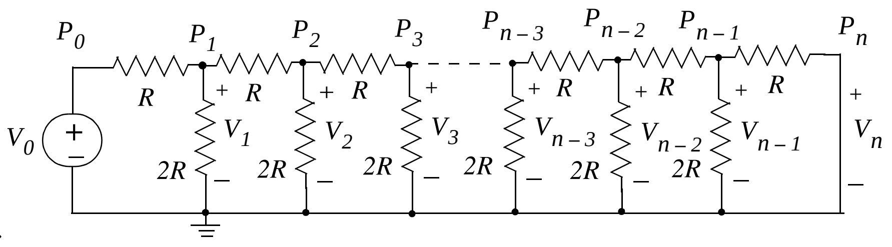


Figure 11.1. Electric network for Example 11.5

Solution:

We need to derive a difference equation that relates the unknown voltage V_x to the known voltage V_0. We start by drawing part of the circuit as shown in Figure 11.2, and we denote the voltages and currents as indicated.

Chapter 12

Partial Fraction Expansion

This chapter is an introduction to partial fraction expansion methods. In elementary algebra we learned how to combine fractions over a common denominator. Partial fraction expansion is the reverse process and splits a rational expression into a sum of fractions having simpler denominators.

12.1 Partial Fraction Expansion

The partial fraction expansion method is used extensively in integration and in finding the inverses of the Laplace, Fourier, and Z transforms. This method allows us to decompose a rational polynomial into smaller rational polynomials with simpler denominators, from which we can easily recognize their integrals or inverse transformations. In the subsequent discussion we will discuss the partial fraction expansion method and we will illustrate with several examples. We will also use the MATLAB **residue(r,p,k)** function which returns the residues (coefficients) **r** of a partial fraction expansion, the poles **p** and the direct terms **k**. There are no direct terms if the highest power of the numerator is less than that of the denominator.

Let

$$F(s) = \frac{N(s)}{D(s)} \tag{12.1}$$

where $N(s)$ and $D(s)$ are polynomials and thus (12.1) can be expressed as

$$F(s) = \frac{N(s)}{D(s)} = \frac{b_m s^m + b_{m-1}s^{m-1} + b_{m-2}s^{m-2} + \ldots + b_1 s + b_0}{a_n s^n + a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \ldots + a_1 s + a_0} \tag{12.2}$$

The coefficients a_k and b_k for $k = 0,\ 1,\ 2,\ \ldots,\ n$ are real numbers and, for the present discussion, we have assumed that the highest power of $N(s)$ is less than the highest power of $D(s)$, i.e., $m < n$. In this case, $F(s)$ is a *proper rational function*. If $m \ge n$, $F(s)$ is an *improper rational function.*

It is very convenient to make the coefficient a_n of s^n in (12.2) unity; to do this, we rewrite it as

$$F(s) = \frac{N(s)}{D(s)} = \frac{\frac{1}{a_n}(b_m s^m + b_{m-1}s^{m-1} + b_{m-2}s^{m-2} + \ldots + b_1 s + b_0)}{s^n + \frac{a_{n-1}}{a_n}s^{n-1} + \frac{a_{n-2}}{a_n}s^{n-2} + \ldots + \frac{a_1}{a_n}s + \frac{a_0}{a_n}} \tag{12.3}$$

```
     1         1
```

The direct terms $k = [1 \quad 1]$ are the coefficients of the s term and the constant in (2.54).

12.2 Alternate Method of Partial Fraction Expansion

The partial fraction expansion method can also be performed by the *equating the numerators procedure* thereby making the denominators of both sides the same, and then equating the numerators. We assume that the degree on the numerator $N(s)$ is less than the degree of the denominator. If not, we first perform a long division and then work with the quotient and the remainder as before.

We also assume that the denominator $D(s)$ can be expressed as a product of real linear and quadratic factors. If these assumptions prevail, we let $s-a$ be a linear factor of $D(s)$ and we suppose that $(s-a)^m$ is the highest power of $s-a$ that divides $D(s)$. Then, we can express $F(s)$ as

$$F(s) = \frac{N(s)}{D(s)} = \frac{r_1}{s-a} + \frac{r_2}{(s-a)^2} + \dots \frac{r_m}{(s-a)^m} \tag{12.36}$$

Next, let $s^2 + \alpha s + \beta$ be a quadratic factor of $D(s)$ and suppose that $(s^2 + \alpha s + \beta)^n$ is the highest power of this factor that divides $F(s)$. Now, we perform the following steps:

1. To this factor, we assign the sum of n partial fractions as shown below.

$$\frac{r_1 s + k_1}{s^2 + \alpha s + \beta} + \frac{r_2 s + k_2}{(s^2 + \alpha s + \beta)^2} + \dots + \frac{r_n s + k_n}{(s^2 + \alpha s + \beta)^n} \tag{12.37}$$

2. We repeat Step 1 for each of the distinct linear and quadratic factors of $D(s)$.
3. We set the given $F(s)$ equal to the sum of these partial fractions.
4. We multiply each term of the right side by the appropriate factor to make the denominators of both sides equal.
5. We arrange the terms of both sides in decreasing powers of s.
6. We equate the coefficients of corresponding powers of s.
7. We solve the resulting equations for the residues.

Example 12.7

Express $F_7(s)$ of (12.38) below as a sum of partial fractions using the equating the numerators procedure.

Chapter 13

The Gamma and Beta Functions and Distributions

This chapter is an introduction to the gamma and beta functions and their distributions used with many applications in science and engineering. They are also used in probability, and in the computation of certain integrals.

13.1 The Gamma Function

The *gamma function*, denoted as $\Gamma(n)$, is also known as *generalized factorial function.* It is defined as

$$\boxed{\Gamma(n) = \int_0^{\infty} x^{n-1} e^{-x} dx} \tag{13.1}$$

and this improper* integral converges (approaches a limit) for all $n > 0$.

We will derive the basic properties of the gamma function and its relation to the well known factorial function

$$n! = n(n-1)(n-2)\ldots 3 \cdot 2 \cdot 1 \tag{13.2}$$

We will evaluate the integral of (13.1) by performing integration by parts using the relation

$$\int u dv = uv - \int v du \tag{13.3}$$

Letting

$$u = e^{-x} \quad and \quad dv = x^{n-1} \tag{13.4}$$

we get

$$du = -e^{-x} dx \quad and \quad v = \frac{x^n}{n} \tag{13.5}$$

Then, with (13.3), we write (13.1) as

* *Improper integrals are two types and these are:*

a. $\int_a^b f(x)dx$ *where the limits of integration* a *or* b *or both are infinite*

b. $\int_a^b f(x)dx$ *where* $f(x)$ *becomes infinite at a value* x *between the lower and upper limits of integration inclusive.*

$$\boxed{n\Gamma(n) = \Gamma(n+1)} \tag{13.11}$$

It is convenient to use (13.10) for $n < 0$, and (13.11) for $n > 0$.

From (13.10), we see that $\Gamma(n)$ becomes infinite as $n \to 0$.

For $n = 1$, (13.1) yields

$$\Gamma(1) = \int_0^{\infty} e^{-x} dx = -e^{-x}\Big|_0^{\infty} = 1 \tag{13.12}$$

Thus, we have derived the important relation,

$$\Gamma(1) = 1 \tag{13.13}$$

From the recurring relation of (13.11), we obtain

$$\begin{aligned} \Gamma(2) &= 1 \cdot \Gamma(1) = 1 \\ \Gamma(3) &= 2 \cdot \Gamma(2) = 2 \cdot 1 = 2! \\ \Gamma(4) &= 3 \cdot \Gamma(3) = 3 \cdot 2 = 3! \end{aligned} \tag{13.14}$$

and in general

$$\boxed{\Gamma(n+1) = n! \quad for\ n = 1, 2, 3, \ldots} \tag{13.15}$$

The formula of (13.15) is a very useful relation; it establishes the relationship between the $\Gamma(n)$ function and the factorial $n!$.

We must remember that, whereas the factorial $n!$ is defined only for zero (recall that $0! = 1$) and positive integer values, the gamma function exists (is continuous) everywhere *except* at 0 and *negative integer numbers*, that is, $-1, -2, -3$, and so on. For instance, when $n = -0.5$, we can find $\Gamma(-0.5)$ in terms of $\Gamma(0.5)$, but if we substitute the numbers $0, -1, -2, -3$ and so on in (13.11), we get values which are not consistent with the definition of the $\Gamma(n)$ function, as defined in that relation.

Stated in other words, *the $\Gamma(n)$ function is defined for all positive integers and positive fractional values, and for all negative fractional, but not negative integer values.*

We can use MATLAB's **gamma(n)** function to plot $\Gamma(n)$ versus n. This is done with the code below which produces the plot shown in Figure 13.1.

```
n=–4: 0.05: 4; g=gamma(n); plot(n,g); axis([–4  4  –6  6]); grid;
title('The Gamma Function'); xlabel('n'); ylabel('Gamma(n)')
```

Figure 13.1 shows the plot of the function $\Gamma(n)$ versus n.

Chapter 14

Orthogonal Functions and Matrix Factorizations

This chapter is an introduction to orthogonal functions. We begin with orthogonal lines and functions, orthogonal trajectories, orthogonal vectors, and we conclude with the factorization methods LU, Cholesky, QR, and singular value decomposition.

14.1 Orthogonal Functions

Orthogonal functions are those which are perpendicular to each other. Mutually orthogonal systems of curves and vectors are of particular importance in physical problems. From analytic geometry and elementary calculus we know that two lines are orthogonal if the product of their slopes is equal to minus one. This is shown in Figure 14.1.

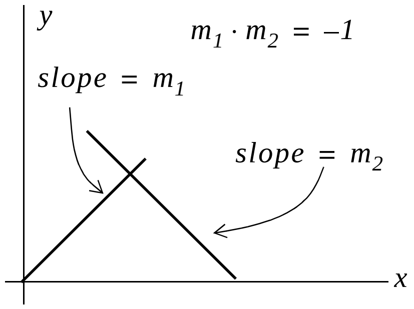


Figure 14.1. Orthogonal lines

Orthogonality applies also to curves. Figure 14.2 shows the angle between two curves C_1 and C_2.

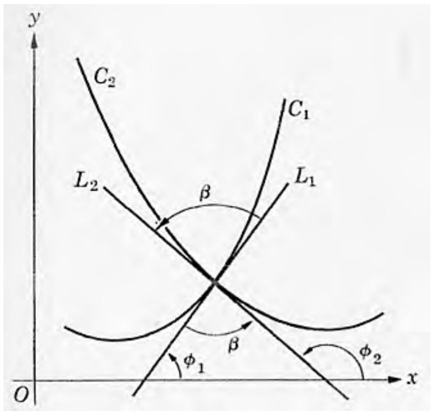


Figure 14.2. Orthogonal curves

```
C =
   -0.4027    0.0000    0.9153
    0.0000    1.0000    0.0000
   -0.9153    0.0000   -0.4027
```

We observe that the vectors of the C matrix produced by MATLAB are different from those we derived with the Gram-Schmidt orthogonalization procedure. The reason for this difference is that the orthogonalization process is not unique, that is, we may find different values depending on the process being used. As shown below, the vectors produced by MATLAB also satisfy the condition $C \cdot C^T = I$.

```
I=C*C'
```

```
I =
    1.0000   -0.0000    0.0000
   -0.0000    1.0000   -0.0000
    0.0000   -0.0000    1.0000
```

14.5 The LU Factorization

In matrix computations, computers use the so-called matrix factorization methods to decompose a matrix A into a product of other smaller matrices. The *LU factorization method* decomposes a matrix A into a lower triangular matrix L and an upper triangular matrix U so that $A = L \cdot U$. In Chapter 4 we saw how the method of Gaussian elimination proceeds by systematically removing the unknowns from a system of linear equations.

Consider the following 3×3 lower triangular case.

$$\begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

The unknowns are found from

$$\begin{aligned} x_1 &= b_1/L_{11} \\ x_2 &= (b_2 - L_{21}x_1)/L_{22} \\ x_3 &= (b_3 - L_{31}x_1 - L_{31}x_2)/L_{33} \end{aligned} \tag{14.16}$$

provided that $L_{11} \cdot L_{22} \cdot L_{33} \neq 0$.

For the upper triangular case, the unknowns are written in reverse order. Thus, to solve

Chapter 15

Bessel, Legendre, and Chebyshev Functions

This chapter is an introduction to some very interesting functions. These are special functions that find wide applications in science and engineering. They are solutions of differential equations with variable coefficients and, under certain conditions, satisfy the orthogonality principle.

15.1 The Bessel Function

The *Bessel functions*, denoted as $J_n(x)$, are used in engineering, acoustics, aeronautics, thermodynamics, theory of elasticity and others. For instance, in the electrical engineering field, they are used in frequency modulation, transmission lines, and telephone equations.

Bessel functions are solutions of the differential equation

$$\boxed{x^2\frac{d^2y}{dx^2}+x\frac{dy}{dx}+(x^2-n^2)y = 0} \tag{15.1}$$

where n can be any number, positive or negative integer, fractional, or even a complex number. Then, the form of the general solution of (15.1) depends on the value of n.

Differential equations with variable coefficients, such as (15.1), cannot be solved in terms of familiar functions as those which we encountered in ordinary differential equations with constant coefficients. The usual procedure is to derive solutions in the form of infinite series, and the most common are the *Method of Frobenius* and the *Method of Picard*. It is beyond the scope of this book to derive the infinite series which are approximations to the solutions of these differential equations; these are discussed in advanced mathematics textbooks. Therefore, we will accept the solutions without proof.

Applying the method of Frobenius to (15.1), we obtain the infinite power series

$$J_n(x) = \sum_{k=0}^{\infty}(-1)^k\cdot\left(\frac{x}{2}\right)^{n+2k}\cdot\frac{1}{k!\cdot\Gamma(n+k+1)} \qquad n\ge 0 \tag{15.2}$$

This series is referred to as *Bessel function of order* n where n is any positive real number or zero. If in (15.2), we replace n with $-n$, we get the relation

$$J_{-n}(x) = \sum_{k=0}^{\infty}(-1)^k\cdot\left(\frac{x}{2}\right)^{-n+2k}\cdot\frac{1}{k!\cdot\Gamma(-n+k+1)} \tag{15.3}$$

```
x = 0.00: 0.05: 10.00; v = besselj(0,x); w = besselj(1,x); z = besselj(2,x);
plot(x,v,x,w,x,z); grid; title('Bessel Functions of the First Kind'); xlabel('x'); ylabel('Jn(x)');
text(0.95, 0.85, 'J0(x)'); text(2.20, 0.60, 'J1(x)'); text(4.25, 0.35, 'J2(x)')
```

The plots for $J_0(x)$, $J_1(x)$ and $J_2(x)$ are shown in Figure 15.1.*

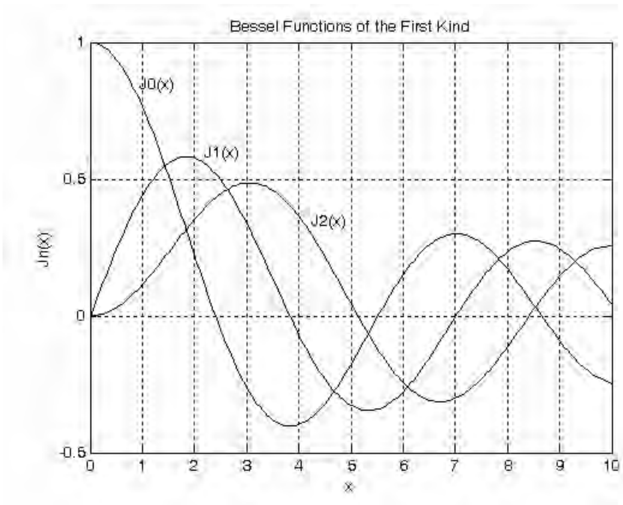


Figure 15.1. Plots of $J_0(x)$, $J_1(x)$ and $J_2(x)$ using MATLAB

We can also use Excel to plot these series as shown in Figure 15.2.

The definition of a Bessel function of the first kind will be explained shortly.

The x-axis crossings in the plot of Figures 15.1 and 15.2 show the first few roots of the $J_0(x)$, $J_1(x)$, and $J_2(x)$ series. However, all $J_n(x)$ are infinite series and thus, it is a very difficult and tedious task to compute all roots of these series. Fortunately, tables of some of the roots of $J_0(x)$ and $J_1(x)$ are shown in math tables.

The equations $J_0(x) = 0$ and $J_1(x) = 0$ exhibit some interesting characteristics. The most noteworthy are:

1. They have no complex roots
2. Each has an infinite number of distinct real roots

* *In Frequency Modulation (FM), x is denoted as β and it is called **modulation index**. The functions $J_0(\beta)$, $J_1(\beta)$, $J_2(\beta)$ and so on, represent the carrier, first sideband, second sideband etc. respectively.*

Chapter 16

Optimization Methods

This chapter introduces three methods for maximizing or minimizing some function in order to achieve the optimum solution. These methods are topics discussed in detail in a branch of mathematics called *operations research* that is concerned with financial and engineering economic problems. Our intent here is to introduce these methods with the basic ideas. We will discuss linear programming, dynamic programming, and network analysis and we will illustrate these with some simple but practical examples.

16.1 Linear Programming

In linear* programming we seek to maximize or minimize a particular quantity, referred to as the *objective*, which is dependent on a finite number of variables. These variables may or may not be independent of each another, and in most cases are subject to certain conditions or limitations referred to as *constraints*.

Example 16.1

The ABC Semiconductor Corporation produces microprocessors (μPs) and memory (RAM) chips. The material types, A and B, required to manufacture the μPs and $RAMs$ and the profits for each are shown in Table 16.1.

TABLE 16.1 Data for Example 16.1

	Parts of Material Types	
	μPs	$RAMs$ (1000s)
Semiconductor Material A	3	2
Semiconductor Material B	5	10
Profit	\$25.00 per unit	\$20.00 per 1000

Due to limited supplies of silicon, phosphorus and boron, its product mix at times of high consumer demand, is subject to limited supplies. Thus, ABC Semiconductor can only buy 450 parts of Material A, and 1000 parts of Material B. This corporation needs to know what combination of μPs and $RAMs$ will maximize the overall profit.

* *A linear program is one in which the variables form a linear combination,i.e., are linearly related. All other programs are considered non-linear.*

Therefore, the minimum cost is 15 and it is achieved through path $a \to c \to e \to h \to m$, as shown in Figure 16.5

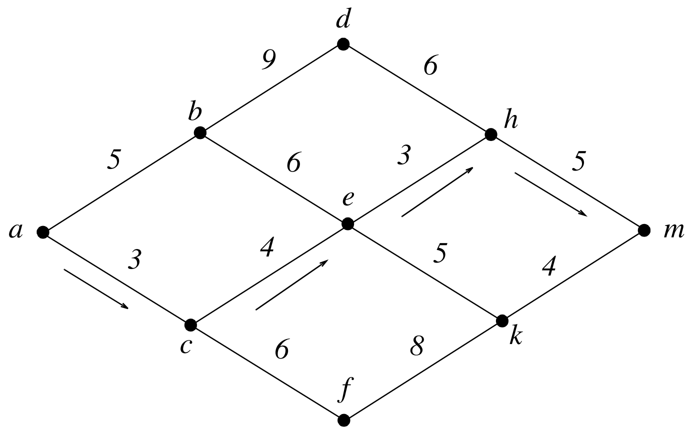


Figure 16.5. Line graph showing the minimum cost for Example 16.2

Example 16.3

On the line graph of Figure 16.6, node A represents an airport in New York City and nodes B through L several airports throughout Europe and Asia. All flights originate at A and fly eastward. A salesman must leave New York City and be in one of the airports H, J, K, or L at the shortest possible time. The encircled numbers represent waiting times in hours at each airport. The numbers in squares show the hours he must travel by an automobile to reach his destination, and the numbers beside the line segments indicated the flight times, also in hours. Which airport should he choose (H, J, K, or L) to minimize his total travel time, and in how many hours after departure from A will he reach his destination?

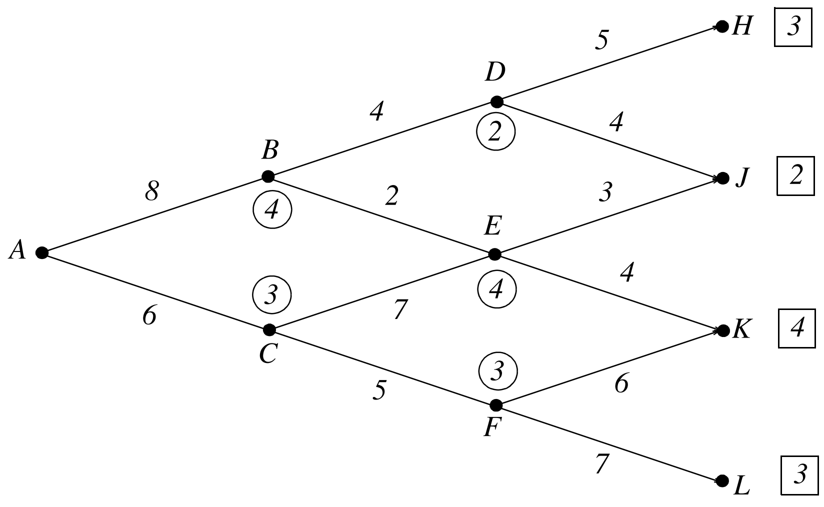


Figure 16.6. Line graph for Example 16.3

Index