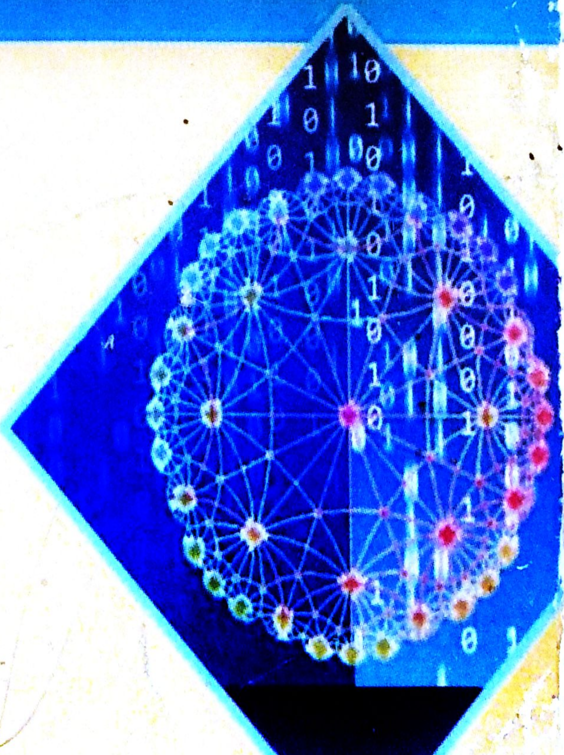
Numerical Methods for Scientists and Engineers

3rd Edition



Contents

Preface

The objective of this third edition is the same as in previous two editions: to provide a broad coverage of various numerical techniques, that are widely used for solving many important problems in engineering, science and technology with the help of computational tools.

I have updated the previous edition by adding new material as suggested by my old colleagues and students. Two new chapters: Chapter 12, Boundary Value Problems and Chapter 13, Approximation of Functions, have been added.

The text now has thirteen chapters. Chapter 1 is unchanged. Chapter 2 has been updated by adding a new section explaining Bairstow method, to find the complex roots of a polynomial with real coefficients and another section with a method for solving a system of nonlinear equations. Chapter 3 remains unchanged.

In Chapter 4, Gerschgorin's theorem is included to get the bounds for eigenvalues of a square matrix, while Chapter 5 has been repeated as such.

Chapter 6 has been updated by including a new section on Hermite interpolation. In Chapter 7, a new section, where Gaussian quadrature type of formula has been derived for evaluating multiple integrals, has been added. Chapter 8 is unchanged except that the concept of stability is introduced and illustrated. Chapters 9 to 11 remain unchanged.

Chapter 12 is new to this edition. Here, finite difference method, shooting method, and weighted residual methods such as Galerkin and collocation methods are discussed in detail. Finally, Chapter 13 is also new to this edition. Here, approximation of a real continuous function by a polynomial, using least squares, Chebyshev and Pade approximations are discussed extensively. In addition, Fourier series approximation is also given with an introduction to Fast Fourier Transform.

I wish to thank all my old colleagues, friends and students whose feedback has helped me to improve over previous two editions.

I also wish to thank the publisher, Prentice-Hall of India, for their careful processing of the manuscript, both at the editorial and production stages.

K. SANKARA RAO

Chapter 1

Basics in Computing

1.1 INTRODUCTION

We begin this chapter with some of the basic concepts of representation of numbers on computers and errors introduced during computation. Problem-solving using computers and the steps involved are also discussed in brief.

Many of the available digital computing systems fall mainly under four categories: personal computers, workstations, mainframe computers and super computers, based on their speed, cost and facilities. Mainframe and super computers being costly and are used only for research and development purposes. These computers involve large-scale programmes with huge data. In the new millennium, computers with storage capacities of several hundred billion words and capable of making 15 billion calculations a second are made available at select installations over the globe.

With the availability of such powerful digital computers and vastly improved numerical methods, scientists and engineers will be able to develop models that can be used for numerous purposes: weather prediction, effect of solar storms on Earth, the performance of an aircraft as a whole, some aspects of space flight simulations and many more practical problems meant for the welfare of the mankind.

Personal computers with local area networking (LAN) and pentium processors can meet most of the demands of teaching, project evaluations, computer-aided design and almost all business applications. In fact, many computer installations provide the user with the necessary software of routine nature in the name of utility sub-programmes.

1.2 REPRESENTATION OF NUMBERS

It is known that in our daily life, we use numbers based on the decimal system. In this system, we use ten symbols 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 and the number 10 is called the base of the system. Thus, when a base N is given, we need N different symbols 0, 1, 2, . . . , $(N-1)$ to represent an arbitrary number. The number systems commonly used in computers are shown as in Table 1.1.

Table 1.1 Number Systems

Base, N	Number
2	Binary
8	Octal
10	Decimal
16	Hexadecimal

Thus, if a number system has only two symbols 0 and 1, then, its base is 2 and so on. In general, an arbitrary real number, a can be written as

$$a = a_m N^m + a_{m-1} N^{m-1} + \cdots + a_1 N^1 + a_0 + a_{-1} N^{-1} + \cdots + a_{-m} N^{-m}$$

In binary system, it has the form,

$$a = a_m 2^m + a_{m-1} 2^{m-1} + \cdots + a_1 2^1 + a_0 + a_{-1} 2^{-1} + \cdots + a_{-m} 2^{-m}$$

In hexadecimal system, we write it as

$$a = a_m 16^m + a_{m-1} 16^{m-1} + \cdots + a_1 16^1 + a_0 + a_{-1} 16^{-1} + \cdots + a^{-m} 16^{-m}$$

For example, the value of the decimal number 1729 is represented and calculated as

$$(1729)_{10} = 1 \times 10^3 + 7 \times 10^2 + 2 \times 10^1 + 9 \times 10^0$$

While the decimal equivalent of binary number 1.0011001 is

$$1 \times 2^0 + 0 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} + 0 \times 2^{-5} + 0 \times 2^{-6} + 1 \times 2^{-7}$$

$$= 1 + \frac{1}{8} + \frac{1}{16} + \frac{1}{128} = (1.1953125)_{10}$$

Electronic computers use binary system whose base is 2. The two symbols used in this system are 0 and 1, which are called *binary digits* or simply *bits*. The internal representation of any data within a computer is in binary form. However, we prefer data input and output of numerical results in decimal system. Within the computer, the arithmetic is carried out in binary form. Infact, there is a built-in circuit design in every computer, which converts decimal input to binary and binary result to decimal output, and carry-out binary addition, subtraction, multiplication and division. For example, the method of converting decimal to binary equivalent can be seen as follows: we divide the given decimal number by 2 and the resulting successive quotients and continue to do so till the quotient becomes zero. Then, the binary equivalent of the decimal number is obtained as a string of remainders and the process can be seen through examples.

Similarly, for converting a decimal fraction into its binary equivalent, we multiply it by 2, the resultant integer part gives the most significant bit of the binary fraction. We keep multiplying by 2 and extract the next significant digit, and the process is continued until the fractional part becomes zero.

Example 1.1 Convert the decimal number 47 into its binary equivalent.

Solution

		Remainder
2	47	↓
2	23	1
2	11	1
2	5	1
2	2	1
2	1	0
	0	1 ⟵ Most significant bit

Thus,

$$(47)_{10} = (101111)_2$$

Example 1.2 Find the binary equivalent of the decimal fraction 0.7625.

Solution

	Product	Integer part	
0.7625 × 2	1.5250	1	← Most significant digit
0.5250 × 2	1.0500	1	
0.05 × 2	0.1	0	
0.1 × 2	0.2	0	
0.2 × 2	0.4	0	
0.4 × 2	0.8	0	
0.8 × 2	1.6	1	
0.6 × 2	1.2	1	
0.2 × 2	0.4	0	Repeated hereafter

Therefore,

$$(0.7625)_{10} = (0.11000011(0011))_2$$

Suppose, we consider 8 bits only, that is, $(0.11000011)_2$; its decimal value is equal to $(0.7617187)_{10}$. While if we take 12 bits, its decimal equivalent is $(0.76245)_{10}$.

Example 1.3 Convert $(59)_{10}$ into binary and then into octal.

Solution

		Remainder
2	59	
2	29	1
2	14	1
2	7	0
2	3	1
2	1	1
	0	1 ⟵ Most significant bit

Thus,

$$(59)_{10} = (111011)_2$$

Now, if we group the binary digits such that each group contains three bits each, we can easily go from binary to octal. Thus,

$$(111011)_2 = 111\ 011 = (73)_8$$

1.2.1 Floating-point Representation

In general, two types of arithmetic operations are carried out in computers: integer arithmetic and floating point arithmetic. However, most scientific and engineering calculations are essentially carried out in floating point arithmetic. For example, an n digit floating point number in base b can be represented as

$$a = \pm(.d_1d_2, \ldots, d_n)_b b^e$$

where, $(.d_1d_2, \ldots, d_n)_b$ is called *mantissa* and e is the *exponent*. Let us consider a binary number as

$$.10110101 \times 2^{11} = .10110101E01011$$

Here, .10110101 is called the mantissa and 1011 is the exponent. The precision of floating point numbers on any computer is determined by the number of digits used in the mantissa, which in general, varies.

Computers having 48 bits to represent single precision real number, in general, allocate 1 bit for sign, 7 bits for exponent and 40 bits for mantissa. Thus, the range is limited from 2.939E−39 to 1.701E+38. The numerical precision in this case is 11 decimal digits. Similarly, those computers having 32 bits to represent a single precision real number, in general, allocate 1 bit for sign, 7 bits for exponent and 24 bits for mantissa. In this case, the limit ranges from 2.939E−39 to 1.701E+38. The numerical precision in this case is only six decimal digits.

When 64-bit double precision arithmetic is used, the computer output may be accurate even up to 16 decimal digits.

1.3 ERRORS IN COMPUTATIONS

Numerically, computed solutions are subject to certain errors. It may be fruitful to identify the error sources and their growth while classifying the errors in numerical computation. There are essentially three error sources: inherent errors, local round-off errors and local truncation errors.

1.3.1 Inherent Errors

It is that quantity of error which is present in the statement of the problem itself, before finding its solution. It arises due to the simplified assumptions made in the mathematical modelling of a problem. It can also arise when the data is obtained from certain physical measurements of the parameters of the problem.

1.3.2 Local Round-off Errors

Every computer has a finite word length, and therefore, it is possible to store only a fixed number of digits of a given input number. Since computers store information in binary form, storing an exact decimal number in its binary form into the computer memory gives an error. This error is computer-dependent. Also, at the end of computation of a particular problem, the final results in the computer, which is obviously in binary form, should be converted into decimal form — a form understandable to the user — before their print out. Therefore, an additional error is committed at this stage too. This error is called *local round-off error*.

For example, in Section 1.2, we have noted that

$$(0.7625)_{10} = (0.11000011\ (0011))_2$$

If a particular computer system has a word length of 12 bits only, then the decimal number 0.7625 is stored in the computer memory in binary form as 0.110000110011. However, it is equivalent to 0.76245. Thus, in storing the number 0.7625, we have committed an error equal to 0.00005, which is the round-off error; inherent with the computer system considered. Thus, we define the *error* as

$$\text{Error} = \text{True value} - \text{Computed value}$$

Now, in order to determine the accuracy of an approximate solution, errors are measured in different ways. *Absolute error*, denoted by |Error|, while, the *relative error* is defined as

$$\text{Relative error} = \frac{|\text{Error}|}{|\text{True value}|}$$

For example, consider the value of $\sqrt{2} = (1.414213\ldots)$ up to four decimal places, then

$$\sqrt{2} = 1.4142 + \text{Error}$$

Hence, we get

$$\text{Absolute error} = |\text{Error}| = 0.00001$$

$$\text{Relative error} = \frac{0.00001}{1.4142}$$

We are aware of the fact that $\sqrt{2}$ is irrational. However, widely used value up to four decimal digits is taken as the true value for the computation of relative error. These error measures are generally used in numerical analysis for measuring the accuracy of the results.

When a number N is written in floating point form with t digits, say, in base 10 as

$$N = (.d_1 d_2 \ldots d_t)10^e$$

we say that the number N has t significant digits. Here, d_1 is called the *most significant digit*. For example, 0.3 agrees with 1/3 to one significant digit, while 0.3333 agrees with 1/3 to four significant digits.

1.3.3 Local Truncation Error

It is generally easier to expand a function into a power series using Taylor series expansion and evaluate it by retaining the first few terms. For example, we may approximate the function $f(x) = \cos x$ by the series

$$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots + (-1)^n \frac{x^{2n}}{(2n)!} + \cdots \tag{1.1}$$

In fact, it is an infinite series expansion. If we use only the first three terms to compute cos x for a given x, we get an approximate answer. Here, the error is due to truncating the series. Suppose, we retain the first n terms, the *truncation error* (TE) is given by

$$\text{TE} \leq \frac{x^{2n+2}}{(2n+2)!} \tag{1.2}$$

It may be noted that the TE is independent of the computer used.

If we wish to compute cos x for $|x| < \pi/2$ accurate with five significant digits, the question is, how many terms in the expansion (1.1) are to be included? In this situation

$$\frac{x^{2n+2}}{(2n+2)!} < .5 \times 10^{-5} = 5 \times 10^{-6}$$

Taking logarithm on both sides, we get

$$(2n + 2) \log x - \log [(2n + 2)!] < \log_{10} 5 - 6 \log_{10} 10 = 0.699 - 6 = -5.3$$

or

$$\log [(2n + 2)!] - (2n + 2) \log x > 5.3$$

We can observe that, for x in the interval $[-\pi/2, \pi/2]$, the above inequality is satisfied for $n = 7$. Hence, seven terms in the expansion (1.1) are required to get the value of cos x, with the prescribed accuracy, in the interval $[-\pi/2, \pi/2]$. In this example, the truncation error is given by

$$\text{TE} \leq \frac{x^{16}}{16!}$$

1.4 PROBLEM-SOLVING USING COMPUTERS

In order to solve a given problem using a computer, the major steps involved are:

(i) Choosing an appropriate numerical method,

(ii) Designing an algorithm,

(iii) Programming and debugging, and

(iv) Computer execution.

These steps are briefly explained as follows.

We define the numerical method as a mathematical formula for finding the solution to a given problem. There may be many methods available to solve the same problem. For example, in Chapter 2, we shall present various computer-based numerical methods, such as, bisection method, regula-falsi method, method of iteration, Newton–Raphson method, Muller's method, Graeffe's root squaring method, etc., for solving an algebraic or transcendental equation. One should choose an appropriate method, which suits best in the given situation, as a first step.

Once we chose a particular method for solving a problem, we should write down the sequence of steps to be followed in order, precisely and unambiguously, to obtain the solution. This is called *designing an algorithm.*

Now, the flow chart for the algorithm is drawn and then translated into a programming language, which we call *computer programme.* This programme should be debugged and free from coding errors. The choice of the languages is for the user to decide. It can be FORTRAN, BASIC, COBOL, Pascal, C, etc.

As a last step, the computer programme is fed to a personal computer or to a mainframe computer, with the necessary data through an input unit. Then, the central processing unit (CPU) of the computing system interprets the programme steps and executes them if the programme is free from coding errors. When it encounters output statement, the numerical answers to the problem are sent to the output unit chosen by the user; it may be a printer. This completes the problem-solving task using a computer.

Chapter 2

Solution of Algebraic and Transcendental Equations

2.1 INTRODUCTION

One of the basic problems in science and engineering is the computation of roots of an equation in the form, $f(x) = 0$. The equation $f(x) = 0$ is called an *algebraic equation*, if it is purely a polynomial in x; it is called a *transcendental equation* if $f(x)$ contains trigonometric, exponential or logarithmic functions. For example,

$$x^3 + 5x^2 - 6x + 3 = 0$$

is an algebraic equation, whereas

$$M = E - e \sin E \qquad \text{and} \qquad ax^2 + \log (x - 3) + e^x \sin x = 0$$

are transcendental equations.

To find the solution of an equation $f(x) = 0$, we find those values of x for which $f(x) = 0$ is satisfied. Such values of x are called the *roots* of $f(x) = 0$. Thus a is a root of an equation $f(x) = 0$, if and only if, $f(a) = 0$.

Before, we develop various numerical methods, we shall list below some of the basic properties of an algebraic equation:

(i) Every algebraic equation of nth degree, where n is a positive integer, has n and only n roots.

(ii) Complex roots occur in pairs. That is, if $(a + ib)$ is a root of $f(x) = 0$, then $(a - ib)$ is also a root of this equation.

(iii) If $x = a$ is a root of $f(x) = 0$, a polynomial of degree n, then $(x - a)$ is a factor of $f(x)$. On dividing $f(x)$ by $(x - a)$ we obtain a polynomial of degree $(n - 1)$.

(iv) Descartes rule of signs: The number of positive roots of an algebraic equation $f(x) = 0$ with real coefficients cannot exceed the number of changes in sign of the coefficients in the polynomial $f(x) = 0$. Similarly, the number of negative roots of $f(x) = 0$ cannot exceed the number of changes in the sign of the coefficients of $f(-x) = 0$. For example, consider an equation

$$x^3 - 3x^2 + 4x - 5 = 0$$

As there are three changes in sign, also, the degree of the equation is three, and hence the given equation will have all the three positive roots.

(v) Intermediate value property: If $f(x)$ is a real valued continuous function in the closed interval $a \le x \le b$. If $f(a)$ and $f(b)$ have opposite signs, then the graph of the function $y = f(x)$ crosses the x-axis at least once; that is $f(x) = 0$ has at least one root ξ such that $a < \xi < b$.

Broadly speaking, all the known numerical methods for solving either a transcendental equation or an algebraic equation can be classified into two groups: *direct methods* and *iterative methods*. Direct methods require no knowledge of the initial approximation of a root of the equation $f(x) = 0$, while iterative methods do require first approximation to initiate iteration. How to get the first approximation? We can find the approximate value of the root of $f(x) = 0$ either by a *graphical method* or by an *analytical method* as explained below:

Graphical method

Often, the equation $f(x) = 0$ can be rewritten as $f_1(x) = f_2(x)$ and the first approximation to a root of $f(x) = 0$ can be taken as the abscissa of the point of intersection of the graphs of $y = f_1(x)$ and $y = f_2(x)$. For example, consider,

$$f(x) = x - \sin x - 1 = 0$$

It can be written as $x - 1 = \sin x$. Now, we shall draw the graphs of

$$y = x - 1 \quad \text{and} \quad y = \sin x$$

as shown in Fig. 2.1. The approximate value of the root is found to be 1.9.

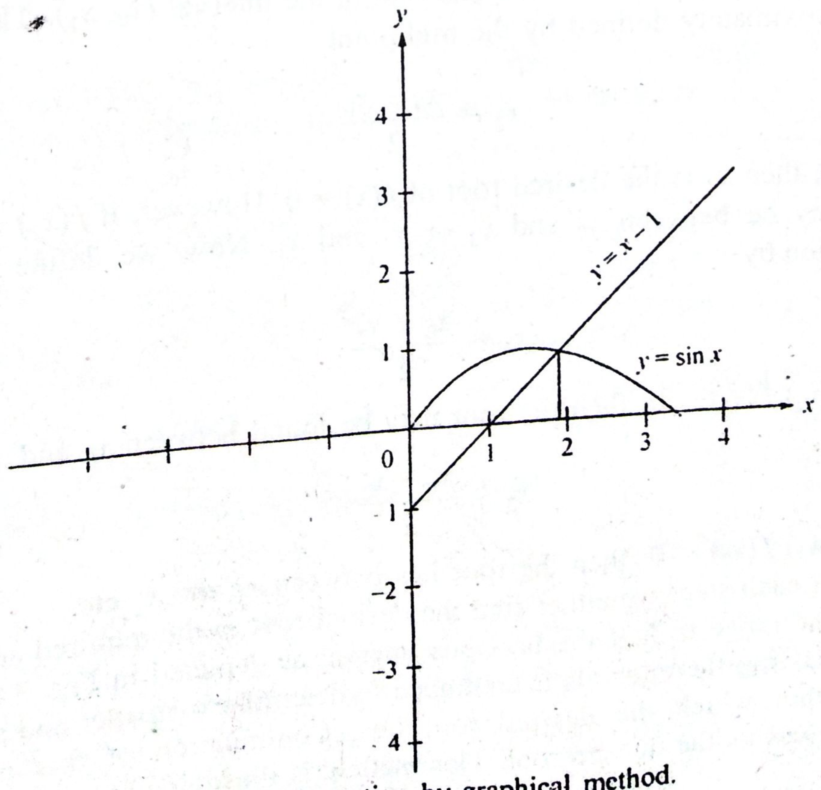


Fig. 2.1 Illustration by graphical method.

Analytical method

This method is based on 'intermediate value property'. We shall illustrate it through an example. Let,

$$f(x) = 3x - \sqrt{1 + \sin x} = 0$$

We can easily verify

$$f(0) = -1$$

$$f(1) = 3 - \sqrt{1 + \sin\left(1 \times \frac{180}{\pi}\right)} = 3 - \sqrt{1 + 0.84147} = 1.64299$$

We observe that $f(0)$ and $f(1)$ are of opposite signs. Therefore, using intermediate value property we infer that there is at least one root between $x = 0$ and $x = 1$. This method is often used to find the first approximation to a root of either transcendental equation or algebraic equation. Hence, in analytical method, we must always start with an initial interval (a, b), so that $f(a)$ and $f(b)$ have opposite signs.

2.2 BISECTION METHOD

This method is due to Bolzano. Suppose, we wish to locate the root of an equation $f(x) = 0$ in an interval, say (x_0, x_1). Let $f(x_0)$ and $f(x_1)$ are of opposite signs, such that $f(x_0)\, f(x_1) < 0$.

Then the graph of the function crosses the x-axis between x_0 and x_1, which guarantees the existence of at least one root in the interval (x_0, x_1). The desired root is approximately defined by the mid-point

$$x_2 = \frac{x_0 + x_1}{2}$$

If $f(x_2) = 0$, then x_2 is the desired root of $f(x) = 0$. However, if $f(x_2) \neq 0$, then the root may be between x_0 and x_2 or x_2 and x_1. Now, we define the next approximation by

$$x_3 = \frac{x_0 + x_2}{2}$$

provided $f(x_0)\, f(x_2) < 0$, then the root may be found between x_0 and x_2 or by

$$x_3 = \frac{x_1 + x_2}{2}$$

provided $f(x_1)\, f(x_2) < 0$, then the root lies between x_1 and x_2 etc.

Thus, at each step, we either find the desired root to the required accuracy or narrow the range to half the previous interval as depicted in Fig. 2.2. This process of halving the intervals is continued to determine a smaller and smaller interval within which the desired root lies. Continuation of this process eventually gives us the desired root. This method is illustrated in the following example.

given by Eq. (2.2). This method can best be understood through the following examples.

Example 2.2 Use the Regula-Falsi method to compute a real root of the equation $x^3 - 9x + 1 = 0$,

(i) if the root lies between 2 and 4
(ii) if the root lies between 2 and 3.

Comment on the results.

Solution Let $f(x) = x^3 - 9x + 1$.

(i) $f(2) = -9$ and $f(4) = 29$. Since $f(2)$ and $f(4)$ are of opposite signs, the root of $f(x) = 0$ lies between 2 and 4. Taking $x_1 = 2$, $x_2 = 4$ and using Regula-Falsi method, the first approximation is given by

$$x_3 = x_2 - \frac{x_2 - x_1}{f(x_2) - f(x_1)} f(x_2) = 4 - \frac{2 \times 29}{38} = 2.47368$$

and $f(x_3) = -6.12644$. Since $f(x_2)$ and $f(x_3)$ are of opposite signs, the root lies between x_2 and x_3. The second approximation to the root is given as

$$x_4 = x_3 - \frac{x_3 - x_2}{f(x_3) - f(x_2)} f(x_3) = 2.73989$$

and $f(x_4) = -3.090707$. Now, since $f(x_2)$ and $f(x_4)$ are of opposite signs, the third approximation is obtained from

$$x_5 = x_4 - \frac{x_4 - x_2}{f(x_4) - f(x_2)} f(x_4) = 2.86125$$

and $f(x_5) = -1.32686$. This procedure can be continued till we get the desired result. The first three iterations are shown as in the table.

n	x_{n+1}	$f(x_{n+1})$
2	2.47368	−6.12644
3	2.73989	−3.090707
4	2.86125	−1.32686

(ii) $f(2) = -9$ and $f(3) = 1$. Since $f(2)$ and $f(3)$ are of opposite signs, the root of $f(x) = 0$ lies between 2 and 3. Taking $x_1 = 2$, $x_2 = 3$ and using Regula-Falsi method, the first approximation is given by

$$x_3 = x_2 - \frac{x_2 - x_1}{f(x_2) - f(x_1)} f(x_2) = 3 - \frac{1}{10} = 2.9$$

and $f(x_3) = -0.711$. Since $f(x_2)$ and $f(x_3)$ are of opposite signs, the root lies between x_2 and x_3. The second approximation to the root is given as

$$x_4 = x_3 - \frac{x_3 - x_2}{f(x_3) - f(x_2)} f(x_3) = 2.94156$$

and $f(x_4) = -0.0207$. Now, we observe that $f(x_2)$ and $f(x_4)$ are of opposite signs, the third approximation is obtained from

$$x_5 = x_4 - \frac{x_4 - x_2}{f(x_4) - f(x_2)} f(x_4) = 2.94275$$

and $f(x_5) = -0.0011896$. This procedure can be continued till we get the desired result. The first three iterations are shown as in the table.

n	x_{n+1}	$f(x_{n+1})$
2	2.9	−0.711
3	2.94156	−0.0207
4	2.94275	−0.0011896

From the above computations, we observe that the value of the root as a third approximation is evidently different in both the cases, while the value of x_5, when the interval considered is (2, 3), is closer to the root. Hence, an important observation in this method is that the interval (x_1, x_2) chosen initially in which the root of the equation lies must be sufficiently small.

Example 2.3 Use Regula-Falsi method to find a real root of the equation

$$\log x - \cos x = 0$$

accurate to four decimal places after three successive approximations.

Solution Given $f(x) = \log x - \cos x$. We observe that

$$f(1) = 0 - 0.5403 = -0.5403$$

and

$$f(2) = 0.69315 + 0.41615 = 1.1093$$

Since $f(1)$ and $f(2)$ are of opposite signs, the root lies between $x_1 = 1$, $x_2 = 2$. The first approximation is obtained from

$$x_3 = x_2 - \frac{x_2 - x_1}{f(x_2) - f(x_1)} f(x_2) = 2 - \frac{1.1093}{1.6496} = 1.3275$$

and

$$f(x_3) = 0.2833 - 0.2409 = 0.0424$$

Now, since $f(x_1)$ and $f(x_3)$ are of opposite signs, the second approximation is obtained as

$$x_4 = 1.3275 - \frac{(.3275)\,(.0424)}{0.0424 + 0.5403} = 1.3037$$

and

$$f(x_4) = 1.24816 \times 10^{-3}$$

Similarly, we observe that $f(x_1)$ and $f(x_4)$ are of opposite signs, so, the third approximation is given by

$$x_5 = 1.3037 - \frac{(0.3037)\,(0.001248)}{0.001248 + 0.5403} = 1.3030$$

and

$$f(x_5) = 0.62045 \times 10^{-4}$$

Hence, the required real root is 1.3030.

Example 2.4 Using Regula-Falsi method, find the real root of the following equation correct to three decimal places:

$$x \log_{10} x = 1.2$$

Solution Let $f(x) = x \log_{10} x - 1.2$. We observe that $f(2) = -0.5979$, $f(3) = 0.2314$. Since $f(2)$ and $f(3)$ are of opposite signs, the real root lies between $x_1 = 2$, $x_2 = 3$. The first approximation is obtained from

$$x_3 = x_2 - \frac{x_2 - x_1}{f(x_2) - f(x_1)} f(x_2) = 3 - \frac{0.2314}{0.8293} = 2.72097$$

and $f(x_3) = -0.01713$. Since $f(x_2)$ and $f(x_3)$ are of opposite signs, the root of $f(x) = 0$ lies between x_2 and x_3. Now, the second approximation is given by

$$x_4 = x_3 - \frac{x_3 - x_2}{f(x_3) - f(x_2)} f(x_3) = 2.7402$$

and $f(x_4) = -3.8905 \times 10^{-4}$. Thus, the root of the given equation correct to three decimal places is 2.740.

2.4 METHOD OF ITERATION

The method of iteration can be applied to find a real root of the equation $f(x) = 0$ by rewriting the same in the form,

$$x = \phi(x) \tag{2.3}$$

For example, $f(x) = \cos x - 2x + 3 = 0$. It can be rewritten as

$$x = \frac{1}{2}(\cos x + 3) = \phi(x)$$

Let $x = \xi$ is the desired root of Eq. (2.3). Suppose x_0 is its initial approximation. The first and successive approximations to the root can be obtained as

$$\left.\begin{aligned} x_1 &= \phi(x_0) \\ x_2 &= \phi(x_1) \\ &\vdots \\ x_{n+1} &= \phi(x_n) \end{aligned}\right\} \tag{2.4}$$

Definition 2.1 Let $\{x_i\}$ be the sequence obtained by a given method and let $x = \xi$ denotes the root of the equation $f(x) = 0$. Then, the method is said to be *convergent*, if and only if

$$\operatorname*{Lt}_{n\to\infty} |x_n - \xi| = 0$$

The convergence of the above sequence to the root is stated as in Theorem 2.1.

Theorem 2.1 Support $x = \xi$ be a root of the equation $f(x) = 0$, which can be rewritten as $x = \phi(x)$, contained in an interval I. Also, let $\phi(x)$ and $\phi'(x)$ be continuous in I. Then, if $|\phi'(x)| < 1$ for all x in I, the iterative process defined by $x_{n+1} = \phi(x_n)$ converges to the root $x = \xi$, if and only if, the initially chosen approximation $x_0 \in I$.

This method is illustrated through the following examples.

Example 2.5 Use the method of iteration to determine the real root of the equation $e^{-x} = 10x$ correct to four decimal places.

Solution Let $f(x) = e^{-x} - 10x = 0$, we observe that $f(0) = 1$ and $f(1) = -9.6321$. Since $f(0) < f(1)$ numerically, the root is near to $x = 0$. Now, we shall rewrite the given equation in the form

$$x = \frac{1}{10}e^{-x} = \phi(x)$$

Therefore,

$$\phi'(x) = -\frac{1}{10}e^{-x}$$

and

$$|\phi'(x)| = \frac{1}{10}e^{-x} = \frac{1}{10\,e^{x}} < 1$$

for all x in (0, 1). Hence, the method of iteration can be applied. Thus, we start with the initial value $x_0 = 0$, then

$$x_1 = \phi(x_0) = \frac{1}{10} = 0.1, \qquad f(x_1) = -0.09516$$

Similarly, the successive approximations are

$$x_2 = \phi(x_1) = \frac{1}{10}e^{-0.1} = \frac{0.904837}{10} = 0.09048, \quad f(x_2) = 0.00869$$

$$x_3 = \phi(x_2) = 0.091349, \qquad f(x_3) = -7.90877 \times 10^{-4}$$

$$x_4 = \phi(x_3) = 0.091274, \qquad f(x_4) = 2.75784 \times 10^{-5}.$$

Hence, the required root is 0.0913.

Example 2.6 Find a real root of the equation

$$f(x) = x^3 + x^2 - 1 = 0$$

by the method of iteration.

Solution We observe that $f(0) = -1$, $f(1) = 1$ which shows that there is a real root between $x = 0$ and $x = 1$. To find the real root, we rewrite the equation in the form

$$x^2(x + 1) = 1 \quad \text{or} \quad x = \frac{1}{\sqrt{x+1}} = \phi(x)$$

Therefore,

$$\phi'(x) = -\frac{1}{2(x+1)^{3/2}}$$

We note that $|\phi'(x)| < 1$, for all x in (0, 1). Hence, the method of iteration is applicable here.

Taking the initial value $x_0 = 1$, we successively obtain the following values:

$$\begin{aligned}
x_1 &= \phi(x_0) = 1/\sqrt{2} = 0.70711, & f(x_1) &= -0.14644\\
x_2 &= \phi(x_1) = 0.76537, & f(x_2) &= 0.03414\\
x_3 &= \phi(x_2) = 0.75263, & f(x_3) &= 7.2213 \times 10^{-3}\\
x_4 &= \phi(x_3) = 0.75536, & f(x_4) &= 1.55658 \times 10^{-3}\\
x_5 &= \phi(x_4) = 0.75477, & f(x_5) &= -3.44323 \times 10^{-4}\\
x_6 &= \phi(x_5) = 0.7549, & f(x_6) &= 7.38295 \times 10^{-5}
\end{aligned}$$

Hence, the required root is 0.7549.

Note: The given equation can be rewritten in many ways. Suppose, we rewrite

$$x^2 = 1 - x^3 \quad \text{or} \quad x = (1 - x^3)^{1/2} = \phi(x)$$

Then

$$|\phi'(x)| = \frac{3x^2}{2(1 - x^3)^{1/2}}$$

if we take $x = 1$, in the interval (0, 1), $|\phi'(x)| = \infty$, then the condition $|\phi'(x)| < 1$ is violated.

2.5 NEWTON–RAPHSON METHOD

This is a very powerful method for finding the real root of an equation in the form, $f(x) = 0$. Suppose, x_0 is an approximate root of $f(x) = 0$. Let $x_1 = x_0 + h$, where h is small, be the exact root of $f(x) = 0$, then $f(x_1) = 0$. Now, expanding $f(x_0 + h)$ by Taylor's theorem, we get

$$f(x_0 + h) = f(x_0) + h f'(x_0) + \frac{h^2}{2} f''(x_0) + \cdots = 0 \tag{2.5}$$

Since h is small, we neglect terms containing h^2 and its higher powers, then

$$f(x_0) + h f'(x_0) = 0 \quad \text{or} \quad h = \frac{-f(x_0)}{f'(x_0)}$$

Therefore, a better approximation to the root is given by

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Still better and successive approximations $x_2, x_3, \ldots, x_n$ to the root can obviously be obtained from the iteration formula,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{2.6}$$

This is known as Newton–Raphson iteration formula, which has the following geometrical interpretation:

Suppose, the graph of the function $y = f(x)$ crosses the x-axis at α (*see* Fig. 2.4), then $x = \alpha$ is the root of the equation $f(x) = 0$.

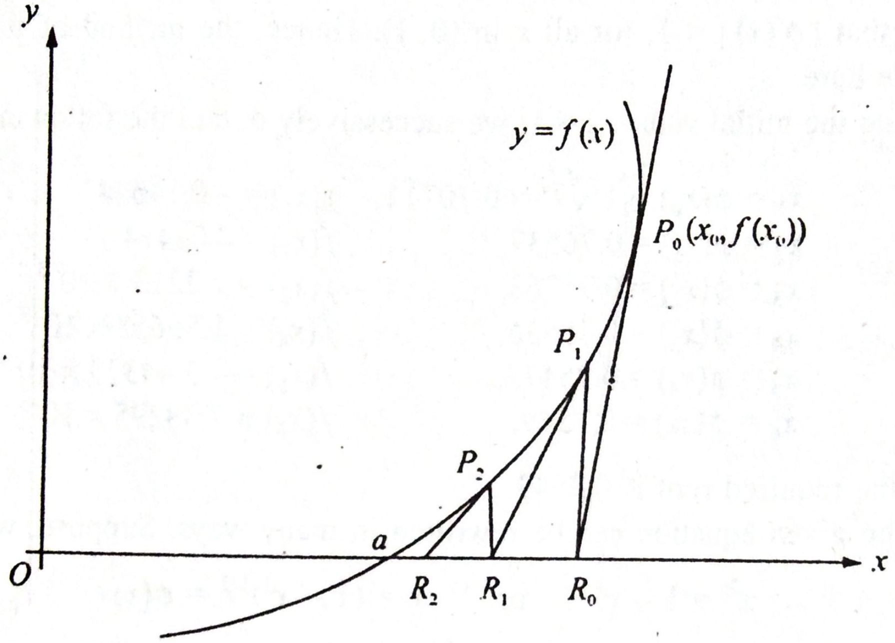


Fig. 2.4 Geometrical interpretation of Newton–Raphson method.

Let x_0 be a point closer to the root α, then the equation of the tangent at $P_0(x_0, f(x_0))$ is

$$y - f(x_0) = f'(x_0)\,(x - x_0) \tag{2.7}$$

This tangent cuts the x-axis at R_0 $(x_1, 0)$. Therefore,

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \tag{2.8}$$

which is a first approximation to the root α. If P_1 is a point on the curve corresponding to x_1, then the tangent at P_1 cuts the x-axis at $R_1(x_2, 0)$, which is still closer to α, than x_1. Therefore, x_2 is a second approximation to the root. Continuing this process, we arrive at the root α, very rapidly, which is evident from Fig. 2.4. Thus, in this method, we have replaced the part of the curve between the point P_0 and x-axis by a tangent to the curve at P_0 and so on. In order to illustrate this method, we shall consider the following examples.

Example 2.7 Find the real root of the equation $xe^x - 2 = 0$ correct to two decimal places, using Newton–Raphson method.

Solution Given $f(x) = xe^x - 2$, we have

$$f'(x) = xe^x + e^x \text{ and } f''(x) = xe^x + 2e^x$$

clearly, we have

$$f(0) = -2 \quad \text{and} \quad f(1) = e - 2 = 0.71828$$

Hence, the required root lies in the interval (0, 1) and is nearer to 1.

Also, $f'(x)$ and $f''(x)$ do not vanish in (0, 1) and $f(x)$ and $f''(x)$ will have the same sign at $x = 1$. Therefore, we take the first approximation $x_0 = 1$, and using Newton–Raphson method, we get

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = \frac{e+2}{2e} = 0.867879$$

and

$$f(x_1) = 6.71607 \times 10^{-2}$$

The second approximation is

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = 0.867879 - \frac{0.06716}{4.44902} = 0.85278$$

and

$$f(x_2) = 7.655 \times 10^{-4}$$

Thus, the required root is 0.853.

Example 2.8 Find a real root of the equation $x^3 - x - 1 = 0$ using Newton–Raphson method, correct to four decimal places.

Solution Let $f(x) = x^3 - x - 1$, then we observe that $f(1) = -1, f(2) = 5$. Therefore, the root lies in the interval (1, 2). We also observe

$$f'(x) = 3x^2 - 1, \qquad f''(x) = 6x$$

and

$$f(1) = -1, \quad f''(1) = 6, \quad f(2) = 5, \quad f''(2) = 12$$

Since $f(2)$ and $f''(2)$ are of the same sign, we choose $x_0 = 2$ as the first approximation to the root. The second approximation is computed using Newton–Raphson method as

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 2 - \frac{5}{11} = 1.54545 \quad \text{and} \quad f(x_1) = 1.14573$$

The successive approximations are

$$x_2 = 1.54545 - \frac{1.14573}{6.16525} = 1.35961, \qquad f(x_2) = 0.15369$$

$$x_3 = 1.35961 - \frac{0.15369}{4.54562} = 1.32579, \qquad f(x_3) = 4.60959 \times 10^{-3}$$

$$x_4 = 1.32579 - \frac{4.60959 \times 10^{-3}}{4.27316} = 1.32471, \qquad f(x_4) = -3.39345 \times 10^{-5}$$

$$x_5 = 1.32471 + \frac{3.39345 \times 10^{-5}}{4.26457} = 1.324718, \qquad f(x_5) = 1.823 \times 10^{-7}$$

Hence, the required root is 1.3247.

Convergence of Newton–Raphson method

To examine the convergence of Newton–Raphson formula (2.6), that is,

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

We compare it with the general iteration formula $x_{n+1} = \phi(x_n)$, and thus obtain

$$\phi(x_n) = x_n - \frac{f(x_n)}{f'(x_n)}$$

In general, we write it as

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

We have already noted in Theorem 2.1 that the iteration method converges if $|\phi'(x)| < 1$. Therefore, Newton–Raphson formula (2.6) converges, provided

$$|f(x)f''(x)| < |f'(x)|^2 \tag{2.9}$$

in the interval considered. Newton–Raphson formula therefore converges, provided the initial approximation x_0 is chosen sufficiently close to the root and $f(x), f'(x)$ and $f''(x)$ are continuous and bounded in any small interval containing the root.

Definition 2.2 Let

$$x_n = \alpha + \varepsilon_n, \qquad x_{n+1} = \alpha + \varepsilon_{n+1}$$

where α is a root of $f(x) = 0$. If we can prove that $\varepsilon_{n+1} = K\varepsilon_n^p$, where K is a constant and ε_n is the error involved at the n th step, while finding the root by an iterative method, then the rate of convergence of the method is p.

We can now establish that Newton–Raphson method converges quadratically.

Let

$$x_n = \alpha + \varepsilon_n, \qquad x_{n+1} = \alpha + \varepsilon_{n+1}$$

where α is a root of $f(x) = 0$ and ε_n is the error involved at the nth step, while finding the root by Newton–Raphson formula (2.6). Then, Eq. (2.6) gives,

$$\alpha + \varepsilon_{n+1} = \alpha + \varepsilon_n - \frac{f(\alpha + \varepsilon_n)}{f'(\alpha + \varepsilon_n)}$$

i.e.

$$\varepsilon_{n+1} = \varepsilon_n - \frac{f(\alpha + \varepsilon_n)}{f'(\alpha + \varepsilon_n)} = \frac{\varepsilon_n f'(\alpha + \varepsilon_n) - f(\alpha + \varepsilon_n)}{f'(\alpha + \varepsilon_n)}$$

Using Taylor's expansion, we get

$$\varepsilon_{n+1} = \frac{1}{f'(\alpha) + \varepsilon_n f''(\alpha) + \cdots}\left\{\varepsilon_n\,[f'(\alpha) + \varepsilon_n f''(\alpha) + \cdots]\right.$$

$$\left. - \left[f(\alpha) + \varepsilon_n f'(\alpha) + \frac{\varepsilon_n^2}{2} f''(\alpha) + \cdots\right]\right\}.$$

Since α is a root, $f(\alpha) = 0$. Therefore, the above expression simplifies to

$$\varepsilon_{n+1} = \frac{\varepsilon_n^2}{2} f''(\alpha) \frac{1}{f'(\alpha) + \varepsilon_n f''(\alpha)}$$

$$= \frac{\varepsilon_n^2}{2} \frac{f''(\alpha)}{f'(\alpha)} \left[1 + \varepsilon_n \frac{f''(\alpha)}{f'(\alpha)}\right]^{-1}$$

$$= \frac{\varepsilon_n^2}{2} \frac{f''(\alpha)}{f'(\alpha)} \left[1 - \varepsilon_n \frac{f''(\alpha)}{f'(\alpha)}\right]$$

or

$$\varepsilon_{n+1} = \frac{\varepsilon_n^2}{2} \cdot \frac{f''(\alpha)}{f'(\alpha)} + O\left(\varepsilon_n^3\right)$$

On neglecting terms of order ε_n^3 and higher powers, we obtain

$$\varepsilon_{n+1} = K\varepsilon_n^2 \tag{2.10}$$

where

$$K = \frac{f''(\alpha)}{2f'(\alpha)} \tag{2.11}$$

It shows that Newton–Raphson method has second order convergence or converges quadratically.

Example 2.9 Set up Newton's scheme of iteration for finding the square root of a positive number N.

Solution The square root of N can be carried out as a root of the equation $x^2 - N = 0$. Let $f(x) = x^2 - N$. By Newton's method, we have

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

In this problem, $f(x) = x^2 - N, f'(x) = 2x$. Therefore,

$$x_{n+1} = x_n - \frac{x_n^2 - N}{2x_n} = \frac{1}{2}\left(x_n + \frac{N}{x_n}\right) \tag{2.12}$$

Example 2.10 Evaluate $\sqrt{12}$, by Newton's formula.

Solution Since $\sqrt{9} = 3$, $\sqrt{16} = 4$, we take $x_0 = (3 + 4)/2 = 3.5$. Using Eq. (2.12), we have

$$x_1 = \frac{1}{2}\left(x_0 + \frac{N}{x_0}\right) = \frac{1}{2}\left(3.5 + \frac{12}{3.5}\right) = 3.4643$$

$$x_2 = \frac{1}{2}\left(3.4643 + \frac{12}{3.4643}\right) = 3.4641$$

$$x_3 = \frac{1}{2}\left(3.4641 + \frac{12}{3.4641}\right) = 3.4641$$

Hence, $\sqrt{12} = 3.4641$.

Example 2.11 Obtain the Newton–Raphson extended formula

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} - \frac{1}{2}\frac{[f(x_0)]^2}{[f'(x_0)]^3} f''(x_0)$$

for finding the root of the equation $f(x) = 0$.

Solution Expanding $f(x)$ by Taylor's series, in the neighbourhood of x_0, we obtain after retaining the first order term only

$$0 = f(x) = f(x_0) + (x - x_0) f'(x_0) + \cdots$$

Which gives

$$x = x_0 - \frac{f(x_0)}{f'(x_0)}$$

This is the first approximation to the root. Therefore,

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \tag{2.13}$$

Again, expanding $f(x)$ by Taylor's series and retaining up to second order term, we have

$$0 = f(x) = f(x_0) + (x - x_0) f'(x_0) + \frac{(x - x_0)^2}{2} f''(x_0)$$

Therefore,

$$f(x_1) = f(x_0) + (x_1 - x_0) f'(x_0) + \frac{(x_1 - x_0)^2}{2} f''(x_0) = 0$$

Using Eq. (2.13), the above equation reduces to the form

$$f(x_0) + (x_1 - x_0) f'(x_0) + \frac{1}{2}\frac{[f(x_0)]^2}{[f'(x_0)]^2} f''(x_0) = 0$$

Thus, the Newton–Raphson extended formula is given by

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} - \frac{1}{2}\frac{[f(x_0)]^2}{[f'(x_0)]^3} f''(x_0) \tag{2.14}$$

This is also known as Chebyshev's formula of third order.

2.6 MULLER'S METHOD

In Muller's method, $f(x) = 0$ is approximated by a second degree polynomial; that is by a quadratic equation that fits through three points in the vicinity of a root.

The roots of this quadratic equation are then assumed to be approximated to the roots of the equation $f(x) = 0$. This method is iterative in nature and does not require the evaluation of derivatives as in Newton–Raphson method. This method can also be used to determine both real and complex roots of $f(x) = 0$.

Suppose, x_{i-2}, x_{i-1}, x_i be any three distinct approximations to a root of $f(x) = 0$. Let $f(x_{i-2}) = f_{i-2}$, $f(x_{i-1}) = f_{i-1}$ and $f(x_i) = f_i$. Noting that any three distinct points in the (x, y)-plane uniquely, determine a polynomial of second degree. A general polynomial of second degree is given by

$$f(x) = ax^2 + bx + c \tag{2.15}$$

Suppose, it passes through the points (x_{i-2}, f_{i-2}), (x_{i-1}, f_{i-1}) and (x_i, f_i) as shown in Fig. 2.5, then the following equations will be satisfied:

$$a x_{i-2}^2 + b x_{i-2} + c = f_{i-2} \tag{2.16}$$

$$a x_{i-1}^2 + b x_{i-1} + c = f_{i-1} \tag{2.17}$$

$$a x_i^2 + b x_i + c = f_i \tag{2.18}$$

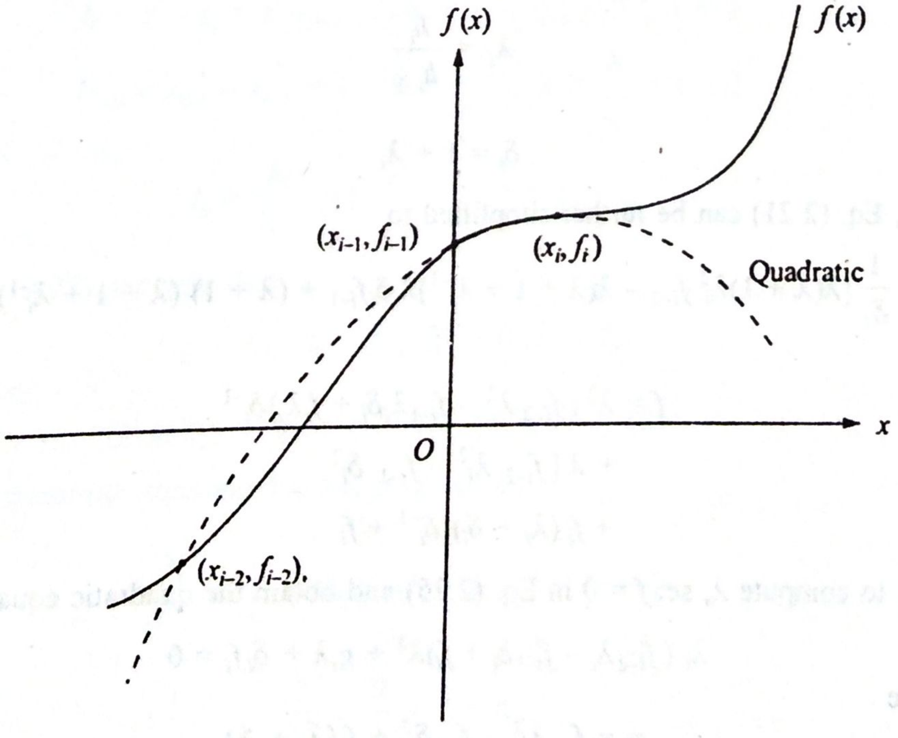


Fig. 2.5 Quadratic polynomial.

Eliminating a, b, c in Eqs. (2.15)–(2.18), we obtain

$$\begin{vmatrix} x^2 & x & 1 & f \\ x_{i-2}^2 & x_{i-2} & 1 & f_{i-2} \\ x_{i-1}^2 & x_{i-1} & 1 & f_{i-1} \\ x_i^2 & x_i & 1 & f_i \end{vmatrix} = 0$$

which can be written conveniently as

$$f = \frac{(x - x_{i-1})(x - x_i)}{(x_{i-2} - x_{i-1})(x_{i-2} - x_i)} f_{i-2} + \frac{(x - x_{i-2})(x - x_i)}{(x_{i-1} - x_{i-2})(x_{i-1} - x_i)} f_{i-1}$$

$$+ \frac{(x - x_{i-2})(x - x_{i-1})}{(x_i - x_{i-2})(x_i - x_{i-1})} f_i \qquad (2.19)$$

Equation (2.19), obviously is a second degree polynomial. Now, introducing the notation

$$h = x - x_i, \qquad h_i = x_i - x_{i-1}, \qquad h_{i-1} = x_{i-1} - x_{i-2} \qquad (2.20)$$

The above equation can be written as

$$f = \frac{(h + h_i)h}{-h_{i-1}(-h_{i-1} - h_i)} f_{i-2} + \frac{(h + h_i + h_{i-1})h}{(h_{i-1})(-h_i)} f_{i-1}$$

$$+ \frac{(h + h_i + h_{i-1})(h + h_i)}{(h_i + h_{i-1})h_i} f_i \qquad (2.21)$$

We further define,

$$\lambda = \frac{h}{h_i} = \frac{x - x_i}{x_i - x_{i-1}} \qquad (2.22)$$

$$\lambda_i = \frac{h_i}{h_{i-1}} \qquad (2.23)$$

and

$$\delta_i = 1 + \lambda_i \qquad (2.24)$$

Thus, Eq. (2.21) can be further simplified to

$$f = \frac{1}{\delta_i}[\lambda(\lambda + 1)\lambda_i^2 f_{i-2} - \lambda(\lambda + 1 + \lambda_i^{-1})\lambda_i \delta_i f_{i-1} + (\lambda + 1)(\lambda + 1 + \lambda_i^{-1})\lambda_i f_i]$$

or

$$f = \lambda^2 (f_{i-2}\lambda_i^2 - f_{i-1}\lambda_i\delta_i + f_i\lambda_i)\delta_i^{-1}$$

$$+ \lambda[f_{i-2}\lambda_i^2 - f_{i-1}\delta_i^2$$

$$+ f_i(\lambda_i + \delta_i)]\delta_i^{-1} + f_i \qquad (2.25)$$

Now, to compute λ, set $f = 0$ in Eq. (2.25) and obtain the quadratic equation as

$$\lambda_i(f_{i-2}\lambda_i - f_{i-1}\delta_i + f_i)\lambda^2 + g_i\lambda + \delta_i f_i = 0 \qquad (2.26)$$

where

$$g_i = f_{i-2}\lambda_i^2 - f_{i-1}\delta_i^2 + f_i(\lambda_i + \delta_i) \qquad (2.27)$$

A direct solution of Eq. (2.26) leads to loss of accuracy, and therefore, to obtain maximum accuracy we rewrite Eq. (2.26) as follows:

$$\frac{f_i\delta_i}{\lambda^2} + \frac{g_i}{\lambda} + \lambda_i(f_{i-2}\lambda_i - f_{i-1}\delta_i + f_i) = 0 \qquad (2.28)$$

so that,

$$\frac{1}{\lambda} = \frac{-g_i \pm [g_i^2 - 4f_i\delta_i\lambda_i(f_{i-2}\lambda_i - f_{i-1}\delta_i + f_i)]^{1/2}}{2f_i\delta_i}$$

or

$$\lambda = \frac{-2f_i\delta_i}{g_i \pm [g_i^2 - 4f_i\delta_i\lambda_i(f_{i-2}\lambda_i - f_{i-1}\delta_i + f_i)]^{1/2}} \tag{2.29}$$

Here, the positive sign must be so chosen that the denominator becomes largest in magnitude. Using

$$x_{i+1} = x_i + h_i\lambda \tag{2.30}$$

we can get a better approximation to the root.

Example 2.12 Find the root of the equation $x^3 - x - 1 = 0$ using Muller's method.

Solution Let $f(x) = x^3 - x - 1$, then $f(0) = -1, f(1) = -1, f(2) = 5$. Therefore, a root lies between 1 and 2 and is close to 1. Muller's method can be conveniently started by taking

$$x_{i-2} = 0, \qquad x_{i-1} = 1 \qquad \text{and} \qquad x_i = 2$$

Correspondingly, we get

$$f_{i-2} = -1, \qquad f_{i-1} = -1 \qquad \text{and} \qquad f_i = 5$$

We define

$$h = x - x_i = x - 2, \qquad h_i = x_i - x_{i-1} = 2 - 1 = 1$$

$$h_{i-1} = x_{i-1} - x_{i-2} = 1 - 0 = 1, \qquad \lambda = \frac{h}{h_i} = x - 2$$

$$\lambda_i = \frac{h_i}{h_{i-1}} = 1, \qquad \delta_i = 1 + \lambda_i = 2$$

From Eq. (2.27), we note that

$$g_i = f_{i-2}\,\lambda_i^2 - f_{i-1}\,\delta_i^2 + f_i\,(\lambda_i + \delta_i)$$

which gives,

$$g_i = (-1)\,(1) - (-1)\,(2)^2 + (5)\,(3) = 18$$

Now, the quadratic equation for $1/\lambda$ is given by

$$\frac{f_i\delta_i}{\lambda^2} + \frac{g_i}{\lambda} + \lambda_i\,(f_{i-2}\lambda_i - f_{i-1}\,\delta_i + f_i) = 0$$

That is,

$$\frac{5}{\lambda^2} + \frac{9}{\lambda} + 3 = 0$$

which gives

$$\frac{1}{\lambda} = \frac{-9 \pm \sqrt{21}}{10}$$

Taking negative sign, so that the numerator is the largest value in magnitude, we get $\lambda = -0.736236$. Therefore, the first approximation to the required root is given by

$$x_{i+1} = x_i + \lambda h_i = 2 - 0.736236 = 1.26376$$

2.7 GRAEFFE'S ROOT SQUARING METHOD

This method is particularly attractive for finding all the roots of a polynomial equation. We shall illustrate this method by considering a polynomial of third degree. However, it may be noted that this method is applicable for higher degree polynomials too.

Consider a polynomial of third degree

$$f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 \tag{2.31}$$

On putting $x = -x$, we get

$$f(-x) = a_0 - a_1x + a_2x^2 - a_3x^3 \tag{2.32}$$

On multiplying Eq. (2.31) with Eq. (2.32), we get

$$f(x)\,f(-x) = a_3^2t^3 - (a_2^2 - 2a_1a_3)\,t^2 + (a_1^2 - 2a_0a_2)\,t - a_0^2 \tag{2.33}$$

Here, we have replaced x^2 by t. It may be observed that, the roots of Eq. (2.33) are indeed the squares or 2^i $(i = 1)$ powers of the original roots. Here $i = 1$ indicates that squaring is done once.

Now, Eq. (2.33) can again be squared and this squaring process is repeated as many times as required. After each squaring, the coefficients become large and overflow is possible as i increases.

Suppose, we have squared the given polynomial i times, then we can estimate the value of the roots by evaluating 2^i root of

$$\left|\frac{a_i}{a_{i-1}}\right|, \qquad i = 1, 2, \ldots, n$$

where n is the degree of the given polynomial. The proper sign of each root can be determined by recalling the original equation. This method of course fails, when the roots of the given polynomial are repeated. This technique could be understood better from the following example.

Example 2.13 Using Graeffe root squaring method, find all the roots of the equation

$$x^3 - 6x^2 + 11x - 6 = 0$$

Solution Using Graeffe root squaring method, the first three squared polynomials are as under:

For $i = 1$, the polynomial is

$$x^3 - (36 - 22)\,x^2 + (121 - 72)\,x - 36 = x^3 - 14x^2 + 49x - 36 \tag{1}$$

For $i = 2$, the polynomial is

$$x^3 - (196 - 98)\,x^2 + (2401 - 1008)\,x - 1296 = x^3 - 98x^2 + 1393x - 1296 \tag{2}$$

For $i = 3$, the polynomial is

$$x^3 - (9604 - 2786)x^2 + (1940449 - 254016)x - 1679616$$
$$= x^3 - 6818x^2 + 1686433x - 1679616 \quad (3)$$

The roots of polynomial (1) are

$$\sqrt{\frac{36}{49}} = 0.85714, \quad \sqrt{\frac{49}{14}} = 1.8708, \quad \sqrt{\frac{14}{1}} = 3.7417$$

Similarly, the roots of polynomial (2) are

$$\sqrt[4]{\frac{1296}{1393}} = 0.9821, \quad \sqrt[4]{\frac{1393}{98}} = 1.9417, \quad \sqrt[4]{\frac{98}{1}} = 3.1464$$

Still better estimates of the roots obtained from polynomial (3) are

$$\sqrt[8]{\frac{1679616}{1686433}} = 0.99949, \quad \sqrt[8]{\frac{1686433}{6818}} = 1.99143, \quad \sqrt[8]{\frac{6818}{1}} = 3.0144$$

It may be observed that the exact values of the roots of the given polynomial are 1, 2 and 3.

2.8 BAIRSTOW METHOD

To find the complex roots of a polynomial

$$f(x) = a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \cdots + a_{n-1}x + a_n = 0 \quad (2.34)$$

where the coefficients a_i are real, we present below a more convenient iterative method which is due to Bairstow. Since complex roots always occur in pairs as $\alpha \pm i\beta$, the corresponding real quadratic polynomial is

$$[x - (\alpha + i\beta)][x - (\alpha - i\beta)] = x^2 - 2\alpha x + (\alpha^2 + \beta^2)$$

which is of the form $x^2 - px - l$. Thus, if we can find quadratic factors, we can avoid complex arithmetic in view of the fact that quadratic factors will always have real coefficients. Also, it may be noted that in any hand computation, performing arithmetic operations of complex numbers is a tedious job. Now, the division of $f(x)$ by the above quadratic factor can be expressed in the form

$$f(x) = (x^2 - px - l)\, Q_{n-2}(x) + r(x) \quad (2.35)$$

where $Q_{n-2}(x)$ is a polynomial of degree $(n - 2)$ and $r(x)$ is the remainder. Thus, we may write

$$f(x) = (x^2 - px - l)(b_0x^{n-2} + b_1x^{n-3} + \cdots + b_{n-3}x + b_{n-2}) + b_{n-1}(x - p) + b_n \quad (2.36)$$

If $(x^2 - px - l)$ were an exact factor of $f(x)$, the remainder $b_{n-1}(x - p) + b_n = 0$. This particular form is chosen only for convenience of later simplifications. This form imply that $b_{n-1} = 0$ and $b_n = 0$, which means that both b_{n-1} and b_n depend on p and l. In other words, the problem is to determine p and l such that

$$b_{n-1}(p, l) = 0, \qquad b_n(p, l) = 0 \tag{2.37}$$

Using Taylor series expansion in terms of $(p^* - p)$ and $(l^* - l)$ and retaining only the first order terms, we get from Eq. (2.37), the following pair of equations:

$$b_{n-1}(p^*, l^*) = b_{n-1}(p, l) + \frac{\partial b_{n-1}}{\partial p}(p^* - p) + \frac{\partial b_{n-1}}{\partial l}(l^* - l)$$

and

$$b_n(p^*, l^*) = b_n(p, l) + \frac{\partial b_n}{\partial p}(p^* - p) + \frac{\partial b_n}{\partial l}(l^* - l) \tag{2.38}$$

Here, we consider (p^*, l^*) as a point at which the remainder is zero and take

$$p^* - p = \Delta p, \quad l^* - l = \Delta l$$

Then, the above pair becomes

$$b_{n-1}(p^*, l^*) = 0 = b_{n-1} + \frac{\partial b_{n-1}}{\partial p}\Delta p + \frac{\partial b_{n-1}}{\partial l}\Delta l$$

$$b_n(p^*, l^*) = 0 = b_n + \frac{\partial b_n}{\partial p}\Delta p + \frac{\partial b_n}{\partial l}\Delta l \tag{2.39}$$

where, all the terms on the right hand side are to be evaluated at (p, l). The solution of these equations gives us the corrections Δp and Δl, which of course require the evaluation of partial derivatives. This procedure is repeated with the updated values of p and l.

In order to compute the coefficients b_i, p and l we equate the coefficients of like powers of x in Eq. (2.36) and get

$$\left.\begin{array}{ll}
a_0 = b_0, & \text{that is } b_0 = a_0 \\
a_1 = b_1 - pb_0, & \text{that is } b_1 = a_1 + pb_0 \\
a_2 = b_2 - pb_1 - lb_0, & \text{that is } b_2 = a_2 + pb_1 + lb_0 \\
\vdots & \\
a_i = b_i - pb_{i-1} - lb_{i-2}, & \text{that is } b_i = a_i + pb_{i-1} + lb_{i-2} \\
\vdots & \\
\multicolumn{2}{l}{\text{Coefficient of } x \text{ gives } b_{n-1} - pb_{n-2} - lb_{n-3} = a_{n-1}, \text{ that is}} \\
 & b_{n-1} = a_{n-1} + pb_{n-2} + lb_{n-3} \\
\multicolumn{2}{l}{\text{Coefficient of constant gives } b_n - pb_{n-1} - lb_{n-2} = a_n, \text{ that is}} \\
 & b_n = a_n + pb_{n-1} + lb_{n-2}
\end{array}\right\} \tag{2.40}$$

This set of equations is equivalent to the recurrence relation

$$b_i = a_i + pb_{i-1} + lb_{i-2}, \quad i = 2, 3, \ldots, n \tag{2.41}$$

If we set $b_{-1} = 0$, $b_{-2} = 0$, the above recurrence relation holds for $i = 0, 1, 2, \ldots n$. Of course, the coefficient b_i depend on the numbers p and l.

Bairstow showed that the required partial derivatives in Eq. (2.38) can be obtained from the b's in just the same way that the b's are obtained from a's as in Eq. (2.40). Taking the partial derivatives of Eqs. (2.40) with respect to p and l and letting

$$\frac{\partial b_{i+1}}{\partial p} = C_i \tag{2.42}$$

we arrive at the following relations

$$\frac{\partial b_0}{\partial p} = \frac{\partial a_0}{\partial p} = 0 \qquad \frac{\partial b_0}{\partial l} = \frac{\partial a_0}{\partial l} = 0$$

$$\frac{\partial b_1}{\partial p} = p\frac{\partial b_0}{\partial p} + b_0 = b_0 = C_0 \qquad \frac{\partial b_1}{\partial l} = p\frac{\partial b_0}{\partial l} = 0$$

$$\frac{\partial b_2}{\partial p} = p\frac{\partial b_1}{\partial p} + b_1 + l\frac{\partial b_0}{\partial p} = C_1 \qquad \frac{\partial b_2}{\partial l} = p\frac{\partial b_1}{\partial l} + l\frac{\partial b_0}{\partial l} + b_0 = b_0 = C_0$$

Therefore, $C_1 = b_1 + pc_0$.

$$\frac{\partial b_3}{\partial p} = p\frac{\partial b_2}{\partial p} + b_2 + l\frac{\partial b_1}{\partial p} = C_2 \qquad \frac{\partial b_3}{\partial l} = p\frac{\partial b_2}{\partial l} + l\frac{\partial b_1}{\partial l} + b_1 = b_1 + pC_0 = C_1$$

Therefore, $C_2 = b_2 + pC_1 + lC_0$

$$\frac{\partial b_{n-1}}{\partial p} = p\frac{\partial_{n-2}}{\partial p} + b_{n-2} + l\frac{\partial b_{n-3}}{\partial p} = C_{n-2}$$

$$\frac{\partial b_{n-1}}{\partial l} = p\frac{\partial b_{n-2}}{\partial l} + l\frac{\partial b_{n-3}}{\partial l} + b_{n-3} = b_{n-3} + pC_{n-4} + lC_{n-5} = C_{n-3}$$

Therefore,

$$C_{n-2} = b_{n-2} + pC_{n-3} + lC_{n-4} \tag{2.43}$$

Utilizing these results, Eqs. (2.39) can now be written as

$$\left.\begin{aligned} -b_{n-1} &= C_{n-2}\,\Delta p + C_{n-3}\,\Delta l \\ -b_n &= C_{n-1}\,\Delta p + C_{n-2}\,\Delta l \end{aligned}\right\} \tag{2.44}$$

These are the central equations for Newton's iteration. Solving these equations for Δp and Δl, we find

$$\left.\begin{aligned} \Delta p &= \frac{b_n C_{n-3} - b_{n-1} C_{n-2}}{C_{n-2}^2 - C_{n-1}\,C_{n-3}}, \\ \Delta l &= \frac{b_{n-1} C_{n-1} - b_n C_{n-2}}{C_{n-2}^2 - C_{n-1}\,C_{n-3}}. \end{aligned}\right\} \tag{2.45}$$

For illustration of the method, we consider the following example

Example 2.14 Find the quadratic factors of

$$x^4 - 1.1x^3 + 2.3x^2 + 0.5x + 3.3 = 0$$

Using $(x^2 + x + 1)$ as a starting factor (Gerald et. al)

Solution: Comparing the given starting factor with the standard factor $(x^2 - px - l)$, we note that $p = -1$, $l = -1$. We are also given the data as $a_0 = 1$, $a_1 = -1.1$, $a_2 = 2.3$, $a_3 = 0.5$, $a_4 = 3.3$. Now, using Eq. (2.40). we compute b's as

$$b_0 = a_0 = 1,\ b_1 = a_1 + pb_0 = -1.1 + (-1)(1) = -2.1 (= b_{n-3})$$

$$b_2 = a_2 + pb_1 + lb_0 = 2.3 + (-1)(-2.1) + (-1)(1) = 3.4\ (= b_{n-2})$$

$$b_3 = a_3 + pb_2 + lb_1 = 0.5 + (-1)(3.4) + (-1)(-2.1) = -0.8\ (= b_{n-1})$$

$$b_4 = a_4 + pb_3 + lb_2 = 3.3 + (-1)(-0.8) + (-1)(3.4) = 0.7\ (= b_n)$$

Finally, using Eq. (2.43), we compute C's as

$$C_0 = b_0 = 1$$

$$C_1 = b_1 + pC_0 = (-2.1) + (-1)(1) = -3.1 \qquad (= C_{n-3})$$

$$C_2 = b_2 + pC_1 + lC_0 = 3.4 + (-1)(-3.1) + (-1)(1) = 5.5 \qquad (= C_{n-2})$$

$$C_3 = b_3 + pC_2 + lC_1 = -0.8 + (-1)(5.5) + (-1)(-3.1) = -3.2 \qquad (= C_{n-1})$$

Substituting these values of b's and C's in Eq. (2.45), we get after first iteration that

$$\Delta p = \frac{(0.7)(-3.1) - (-0.8)(5.5)}{(5.5)^2 - (-3.2)(-3.1)} = \frac{2.23}{20.33} = 0.11$$

$$\Delta l = \frac{(-0.8)(-3.2) - (0.7)(5.5)}{(5.5)^2 - (-3.2)(-3.1)} = \frac{-1.29}{20.33} = -0.06$$

Thus, as a first approximation, we have

$$p^* = (-1) + \Delta p = (-1) + (0.11) = -0.89$$

$$l^* = (-1) + \Delta l = (-1) + (-0.06) = -1.06$$

Now, starting with $p = -0.89$, $l = -1.06$ and repeating the above steps, we get

$$\Delta p = -0.01, \qquad \Delta l = -0.04 \quad \text{and}$$

$$p^* = -0.89 - 0.01 = -0.9, \quad l^* = -1.06 - 0.04 = -1.1$$

Hence, the factor after the second iteration is $(x^2 + 0.9x + 1.1)$. Finally, the quadratic factors of the given equation is found to be

$$(x^2 + 0.9x + 1.1)(x^2 - 2x + 3)$$

2.9 SYSTEM OF NON-LINEAR EQUATIONS

The general problem is to solve a system of n non-linear equations in n unknowns. That is, to solve

$$\left.\begin{aligned} f_1(x_1, x_2, \ldots, x_n) &= 0 \\ f_2(x_1, x_2, \ldots, x_n) &= 0 \\ &\vdots \\ f_n(x_1, x_2, \ldots, x_n) &= 0 \end{aligned}\right\} \qquad (2.46)$$

Most of the known methods are of iterative type, where we start with initial guess and improve the solution iteratively, until a required tolerance is achieved. By taking the gradients of all the variables, we get a function matrix called the Jacobian defined as

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

It is essential that this Jacobian be non-singular for the existence of a solution. It may also be noted that the value of the Jacobian changes from iteration to iteration. Suppose, we denote the vector of components $(x_1, x_2, \ldots, x_n)$ by X and the vector $(f_1, f_2, \ldots, f_n)$ by F, then the system (2.46) can be written as

$$F(X) = 0 \qquad (2.47)$$

The task is to solve the system (2.47). We present below a simple Newton's method by considering a system in two variables such as

$$\begin{aligned} f(x, y) &= 0 \\ g(x, y) &= 0 \end{aligned} \qquad (2.48)$$

Suppose, we choose (x_0, y_0) as the initial (guess) approximation and let h, k are the quantities to be determined such that

$$\begin{aligned} f(x_0 + h,\ y_0 + k) &= 0 \\ g(x_0 + h,\ y_0 + k) &= 0 \end{aligned} \qquad (2.49)$$

Using Taylor series expansion, we get

$$f(x_0 + h, y_0 + k) = f(x_0, y_0) + \left(h\frac{\partial f}{\partial x} + k\frac{\partial f}{\partial y}\right)\Bigg|_{(x_0, y_0)} + \text{H.O.T} = 0$$

$$g(x_0 + h, y_0 + k) = g(x_0, y_0) + \left(h\frac{\partial g}{\partial x} + k\frac{\partial g}{\partial y}\right)\Bigg|_{(x_0, y_0)} + \text{H.O.T} = 0$$

Neglecting the higher order terms (H.O.T) and solving the above system, we get the approximate values of h and k as

$$h = \left.\frac{-f\dfrac{\partial g}{\partial y} + g\dfrac{\partial f}{\partial y}}{J}\right|_{(x_0, y_0)} \tag{2.50}$$

$$k = \left.\frac{-g\dfrac{\partial f}{\partial x} + f\dfrac{\partial g}{\partial x}}{J}\right|_{(x_0, y_0)}$$

where J is the Jacobian defined by

$$J = \begin{vmatrix} f_x & f_y \\ g_x & g_y \end{vmatrix} \tag{2.51}$$

Once h and k are computed, we get the first approximate solution in the form

$$\left.\begin{aligned} x_1 &= x_0 + h \\ y_1 &= y_0 + k \end{aligned}\right\} \tag{2.52}$$

which suggests an iteration

$$\left.\begin{aligned} x_{n+1} &= x_n + h \\ y_{n+1} &= y_n + k \end{aligned}\right\} \tag{2.53}$$

Observe that we need to guess a good initial approximation (x_0, y_0) for convergence of iteration, which we may get graphically. The generalisation to a system of equations in n variables is of course straight forward. Here, follows an example for illustration.

Example 2.15 Solve the following system of equations

$$f(x, y) = x^3 - 3xy^2 - 2x + 2 = 0$$

$$g(x, y) = 3x^2y - y^3 - 2y = 0$$

taking $x_0 = y_0 = 1$, as the initial approximation.

Solution In the present example, we have

$$f(x, y) = x^3 - 3xy^2 - 2x + 2$$

$$g(x, y) = 3x^2y - y^3 - 2y \tag{1}$$

Therefore,

$$f_x = 3x^2 - 3y^2 - 2, \qquad f_y = -6xy$$

$$g_x = 6xy, \qquad g_y = 3x^2 - 3y^2 - 2 \tag{2}$$

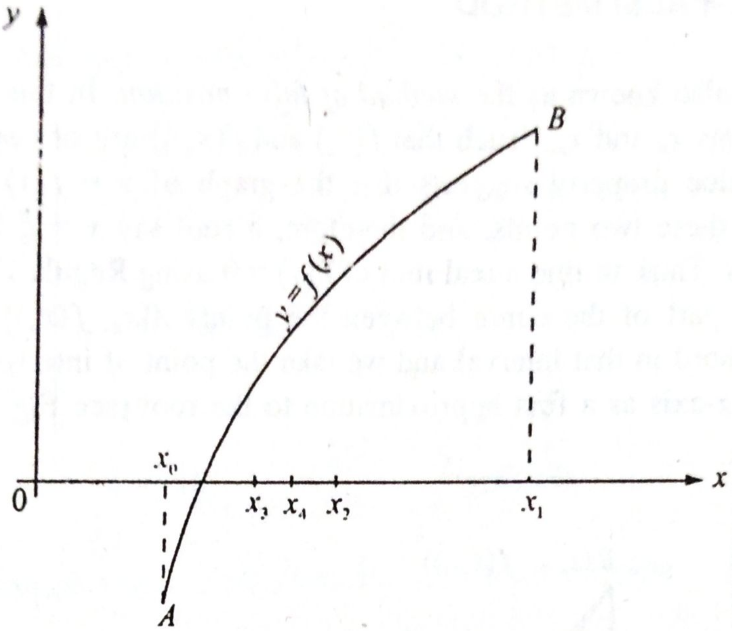


Fig. 2.2 Geometrical illustration of bisection method.

Example 2.1 Solve $x^3 - 9x + 1 = 0$ for the root between $x = 2$ and $x = 4$ by the bisection method.

Solution Given $f(x) = x^3 - 9x + 1$. We can verify $f(2) = -9$, $f(4) = 29$. Therefore, $f(2)\,f(4) < 0$ and hence the root lies between 2 and 4. Let $x_0 = 2$, $x_1 = 4$. Now, we define

$$x_2 = \frac{x_0 + x_1}{2} = \frac{2 + 4}{2} = 3$$

as a first approximation to a root of $f(x) = 0$ and note that $f(3) = 1$, so that $f(2)\,f(3) < 0$. Thus, the root lies between 2 and 3. We further define,

$$x_3 = \frac{x_0 + x_2}{2} = \frac{2 + 3}{2} = 2.5$$

and note that $f(x_3) = f(2.5) < 0$, so that $f(2.5)\,f(3) < 0$. Therefore, we define the mid-point,

$$x_4 = \frac{x_3 + x_2}{2} = \frac{2.5 + 3}{2} = 2.75, \text{ etc.}$$

Similarly, we find that

$$x_5 = 2.875 \quad \text{and} \quad x_6 = 2.9375$$

and the process can be continued until the root is obtained to the desired accuracy. These results are presented in the table.

n	x_n	$f(x_n)$
2	3	1.0
3	2.5	–5.875
4	2.75	–2.9531
5	2.875	–1.1113
6	2.9375	–0.0901

Assume the initial approximation $(x_0, y_0) = (1, 1)$ so that

$$f(x_0, y_0) = -2, \qquad g(x_0, y_0) = 0$$

$$f_x\Big|_{(x_0,y_0)} = -2, \quad f_y\Big|_{(x_0,y_0)} = -6$$

$$g_x\Big|_{(x_0,y_0)} = 6, \quad g_y\Big|_{(x_0,y_0)} = -2$$

then the Jacobian

$$J\Big|_{(x_0, y_0)} = \begin{vmatrix} -2 & -6 \\ 6 & -2 \end{vmatrix} = 40 \neq 0$$

Hence, the solution exists. Now, from Eq. (2.50) we compute

$$h = \frac{[(2)(-2) + (0)(-6)]}{40} = -\frac{1}{10} = -0.1$$

$$k = \frac{[(0)(-2) + (-2)(6)]}{40} = -\frac{3}{10} = -0.3$$

Thus, the first approximation is

$$x_1 = x_0 + h = 1 - 0.1 = 0.9, \; y_1 = y_0 + k = 1 - 0.3 = 0.7 \qquad (3)$$

Similarly, we find

$$f(x_1, y_1) = 0.729 - 1.323 - 1.8 + 2 = -0.394$$

$$g(x_1, y_1) = 1.701 - 0.343 - 1.4 = -0.042$$

$$f_x = 2.43 - 1.47 - 2 = -1.04, \quad f_y = -3.78$$

$$g_x = 3.78, \; g_y = 2.43 - 1.47 - 2 = -1.04$$

Therefore,

$$J = \begin{vmatrix} -1.04 & -3.78 \\ 3.78 & -1.04 \end{vmatrix} = 15.37 \neq 0$$

$$h = \frac{(0.394)(-1.04) + (-0.042)(-3.78)}{15.37} = -0.0163$$

$$k = \frac{(0.042)(-1.04) + (-0.394)(3.78)}{15.37} = -0.0997$$

Hence, the second approximation is

$$x_2 = x_1 + h = 0.9 - 0.0163 = 0.8837$$

$$y_2 = y_1 + k = 0.7 - 0.0997 = 0.6003$$

This is the solution of the given system after two iterations. At this point, we may note that

$$f(x_2, y_2) = -0.03265$$

and

$$g(x_2, y_2) = -0.01055.$$

Further continuation gives

$$f_x\Big|_{(x_2,y_2)} = 1.2617, \quad f_y\Big|_{(x_2,y_2)} = -3.1829$$

$$g_x\Big|_{(x_2,y_2)} = 3.1829, \quad g_y\Big|_{(x_2,y_2)} = 1.2617$$

and

$$J = \begin{vmatrix} f_x & f_y \\ g_x & g_y \end{vmatrix} = 11.7227 \neq 0$$

which yield

$$h = 0.006379, \qquad k = -0.0773$$

Thus, the third approximation is

$$x_3 = x_2 + h = 0.8901, \qquad y_3 = y_2 + k = 0.5926$$

and

$$f(x_3, y_3) = -0.0127, \quad g(x_3, y_3) = 0.0152$$

The three iterations are tabulated as

i	x_i	y_i	$f(x_i, y_i)$	$g(x_i, y_i)$
1	0.9	0.7	–0.394	–0.042
2	0.8837	0.6003	–0.0327	–0.0106
3	0.8901	0.5926	–0.0127	0.0152

EXERCISES

2.1 Find the real root of the equation, $x^3 - 3x - 5 = 0$ by the bisection method.

2.2 Find the real root of the equation, $x^3 + x - 3 = 0$, using Regula-Falsi method, correct to four places of decimal.

2.3 Find the real root of the equation $x^6 - x^4 - x^3 - 1 = 0$, which lies between 1.4 and 1.5, correct to four places of decimal, by the method of false position, obtained after three successive approximations.

2.4 Use Regula-Falsi method to find the real roots of the equation $x^3 - \sin x + 1 = 0$ correct to four decimal places after three successive approximations between (–2, –1).

2.5 Explain the method of false position for finding a real root of the equation $f(x) = 0$, and hence derive the general formula.

2.6 Use Regula-Falsi method to compute the root of the equation $\cos x - xe^x = 0$.

2.7 Find the root of the equation $2x = \cos x + 3$, correct to three decimal places using iteration method.

2.8 Find a root of the equation $x \log_{10} x = 4.77$ by Newton–Raphson method, correct to two decimal places.

2.9 Explain the Newton–Raphson method to find a root of the equation $f(x) = 0$, and hence derive its iteration formula.

2.10 Geometrically explain Newton–Raphson method to find a root of the equation $f(x) = 0$ and hence derive the general formula.

2.11 Obtain the real root of the equation $x^3 - 3x - 5 = 0$ using Newton–Raphson method, after third iteration.

2.12 Find a real root of the equation, $x^4 - x - 10 = 0$ using Newton–Raphson method correct to four decimal places.

2.13 Apply Newton–Raphson method to determine a root of the equation $\cos x = x\, e^x$ correct to three decimal places, using the initial approximation, $x_0 = 1$.

2.14 Set up the Newton's scheme of iteration for finding the p-th root of a positive number N.

2.15 Obtain the cube root of 12 using Newton–Raphson iteration.

2.16 Find the first approximation of the root of the equation $x^3 - 3x - 5 = 0$ using Muller's method, which lies between 2 and 3.

2.17 Find the first approximation to the root of the equation

$$f(x) = \sin x - \frac{x}{2} = 0$$

near $x = 2.0$, using Muller's method.

2.18 Using Graeffe's root squaring method, find the roots of the equation $x^3 - 4x^2 + 3x + 1 = 0$ with the help of a calculator.

2.19 Using the method of false position, find the root of $x \sin x - 1 = 0$ which lies in the interval (0, 2).

2.20 Find the quadratic factors of

$$x^4 - 5.7x^3 + 26.7x^2 - 42.21x + 69$$

Using Bairstows method with $(x^2 - 1.5x + 4.3)$ as a starting factor.

2.21 Using Bairstows method, find the quadratic factors of the polynomial

$$2x^4 + 7x^3 - 4x^2 + 29x + 14$$

with $(x^2 + 5x + 2)$ as a starting factor.

2.22 Find the solution of

$$f(x, y) = x^3 - 3xy^2 + 1 = 0$$

$$g(x, y) = 3x^2y - y^3 = 0$$

taking (1, 1) as the initial approximation using Newtons method.

2.23 Using Newtons method, find the solution of

$$f(x, y) = 4x^2 + y^2 + 2xy - y - 2 = 0$$

$$g(x, y) = 2x^2 + 3xy + y^2 - 3 = 0$$

taking (0.4, 0.9) as the initial approximation.