Introduction to Numerical Analysis

1.1 Analysis versus Numerical Analysis

The word ***analysis*** in mathematics usually means who to solve a problem through equations. The solving procedures may include algebra, calculus, differential equations, or the like.

Numerical analysis is similar in that problems solved, but the only procedures that are used are arithmetic: add, subtract, multiply, divide and compare.

Differences between ***analytical solutions*** and ***numerical solutions***:

1) An analytical solution is usually given in terms of mathematical functions. The behavior and properties of the function are often apparent. However, a numerical solution is always an approximation. It can be plotted to show some of the behavior of the solution.

2) An analytical solution is not always meaningful by itself.

   Example: $\sqrt{3}$ as one of the roots of $x^3 - x^2 - 3x + 3 = 0$.

3) While the numerical solution is an approximation, it can usually be evaluated as accurate as we need. Actually, evaluating an analytic solution numerically is subject to the same errors.

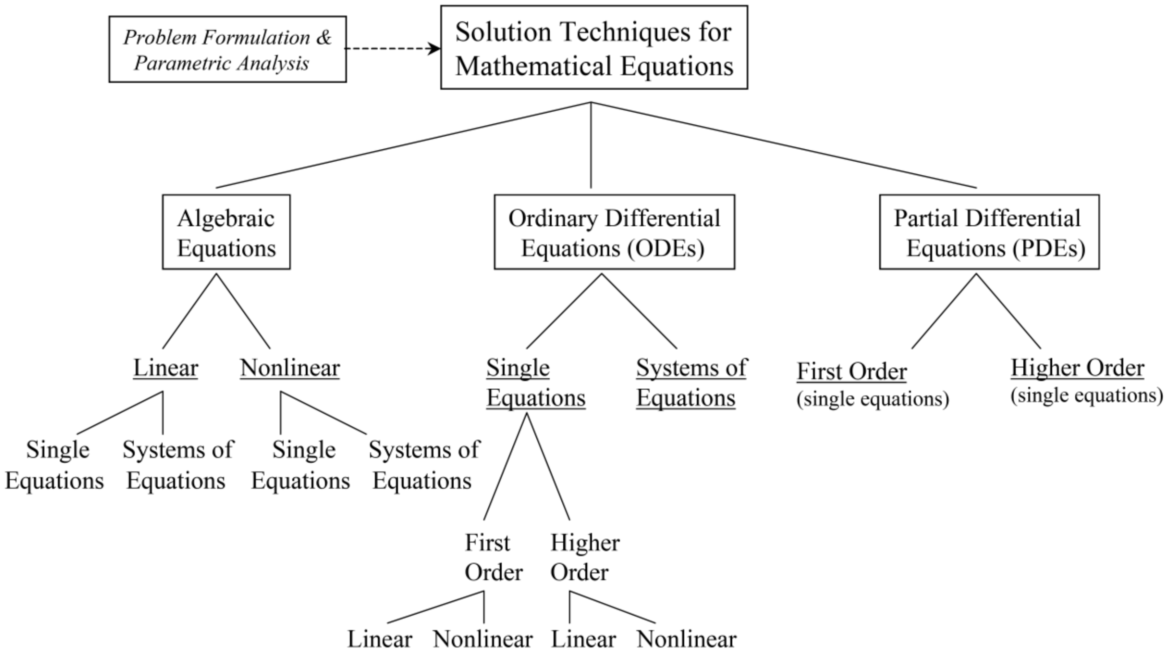
1.2 Computers and Numerical Analysis

Numerical Methods＋Programs —Computers→ Numerical Analysis

- As you will learn enough about many ***numerical methods***, you will be able to write ***programs*** to implement them.
- Programs can be written in any computer language. In this course all programs will be written in Matlab environment.
- Actually, writing programs is not always necessary. Numerical analysis is so important that extensive commercial software packages are available.

1.3 Types of Equations

The equations is divided into three main categories such as in below figure:-



1.4 Kinds of Errors in Numerical Procedures

The total error comprises of:

1) ***Model Error***: due to the mismatch between the physical situation and the mathematical model.

2) ***Data Error***: due to the measurements of doubtful accuracy.

3) ***Human Error***: due to human blunders.

4) ***Propagated Error***: the error in the succeeding steps of a process due to an occurrence of an earlier error.

5) ***Truncation Error***: the notion of truncation error usually refers to errors introduced when a more complicated mathematical expression is “replaced” with a more elementary formula. This formula itself may only be approximated to the true values, thus would not produce exact answers.

Example 1.1:

Truncation of an infinite series to a finite series to a finite number of terms leads to the truncation error. For example, the Taylor series of exponential function

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots + \frac{x^n}{n!}$$

If only four terms of the series are used, then

$$e^x \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!}$$

$$e^1 \approx 1 + 1 + \frac{1^2}{2!} + \frac{1^3}{3!} \quad = 2.66667$$

The truncation error would be the unused terms of the Taylor series, which then are

$$E_t = \frac{x^4}{4!} + \frac{x^5}{5!} + \Lambda \;= \frac{1^4}{4!} + \frac{1^5}{5!} + \Lambda \qquad \cong 0.0516152$$

Check a few Taylor series approximations of the number ex, for x = 1, n = 2, 3 and 4. Given that e1 = 2.718281.

Order of n	Approximation for ex	Absolute error	Percent relative error
2	2.500000	0.218281	8.030111%
3	2.666667	0.051614	1.898774%
4	2.708333	0.00995	0.365967%

6) ***Round-Off Error***: A **round-off error**, also called **rounding error**, is the difference between the calculated approximation of a number and its exact mathematical value **due** to rounding

Example 1.2:

Numbers such as π, e, or $\sqrt{3}$ cannot be expressed by a fixed number of decimal places. Therefore they cannot be represented exactly by the computer.

Consider the number π. It is irrational, i.e. it has infinitely many digits after the period: π = 3.1415926535897932384626433832795.....

The round-off error computer representation of the number π depends on how many digits are left out.

Let the true value for π is 3.141593.

Number of digits (Decimal digit)	Approximation for π	Absolute error	Percent relative error
1	3.1	0.041593	1.3239%
2	3.14	0.001593	0.0507%
3	3.142	0.000407	0.0130%

1.5 Errors in Numerical Procedures

There are two common ways to express the size of the error in a computed result: ***absolute error*** and ***relative error***.

- Absolute error = | true value – approximate value |, which is usually used when the magnitude of the true value is small.

- Relative error = $\frac{|\text{true value - approximate value}|}{|\text{true value}|}$, which is a desirable one.

While

$$\text{Percent relative error}, \varepsilon_t = \left|\frac{\text{true value} - \text{approximate value}}{\text{true value}}\right| \times 100\%$$

Interpolation and Approximation

What is interpolation?

Many times, data is given only at discrete points such as $(x_1, y_1), (x_2, y_2), \ldots\ldots\ldots (x_n, y_n), (x_{n+1}, y_{n+1})$. So, how then does one find the value of y at any other value of x?

Well, a continuous function $f(x)$ may be used to represent the n+1 data values with $f(x)$ passing through the n+1 point (Figure 2.1). Then we can find the value of y at any other value of x. This is called **interpolation**.

Of course, if x falls outside the range of x for which the data is given, it is no longer interpolation, but instead, is called **extrapolation**.

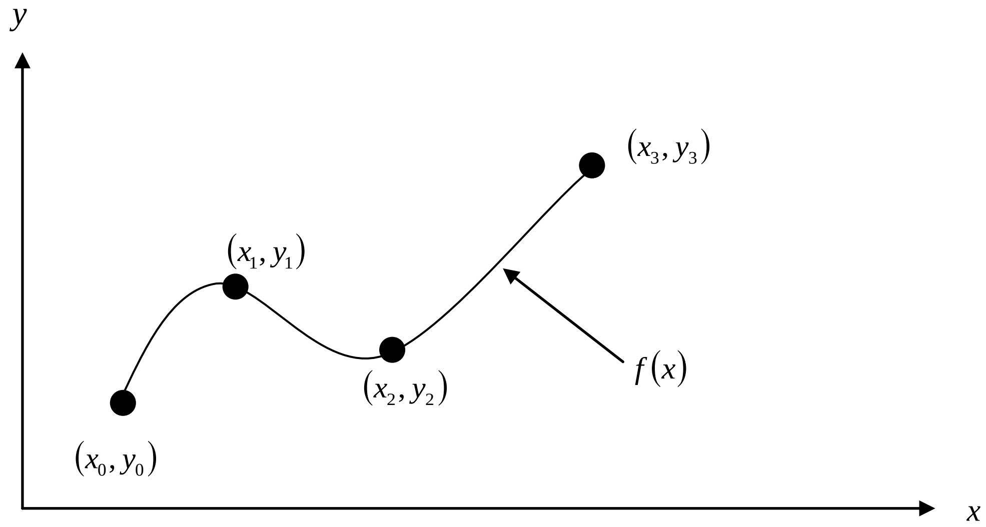


Figure 2.1 Interpolation of discrete data

For n+1 data points, there is one and only one polynomial of order n that passes through all the points. For example, there is only one straight line (that is, a first-order polynomial) that connects two points. Similarly, only one parabola connects a set of three points.

Polynomial Interpolation consists of determining the unique n^{th} order polynomial that fits n+1 data points. This polynomial then provides a formula to compute intermediate values.

One of the methods used to find this polynomial is called the Lagrange method of interpolation. Other methods include Newton's divided difference polynomial method and the direct method.

2.1 Lagrange Interpolating Polynomial

Consider a function $f(x)$ that passes through the two distinct points (x_0, $f(x_0)$) and (x_1, $f(x_1)$) as shown in Figure 2.2. The first order polynomial that approximates the function between these two points can be expressed as

$f(x) = a + bx$

Where a and b are constants. $f(x)$ can also be written in Lagrangian form as

$f(x) = c_0(x - x_1) + c_1(x - x_0)$

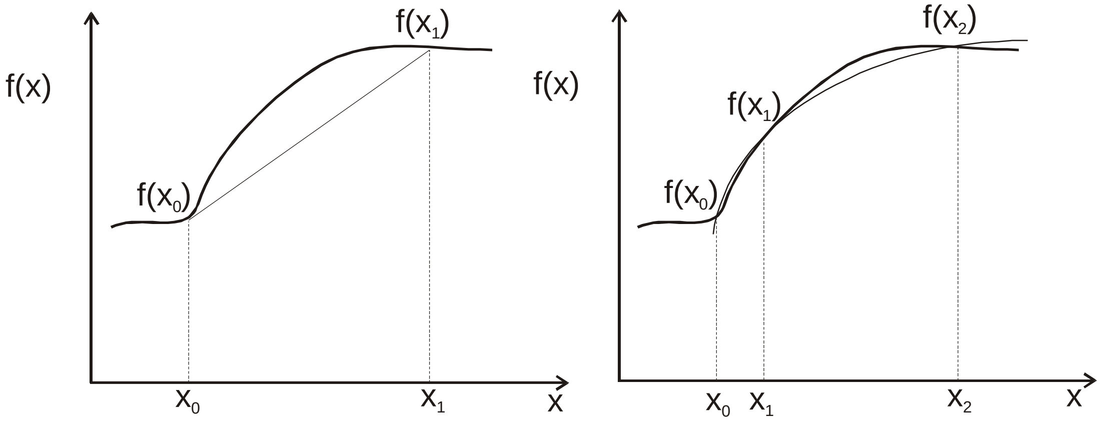


Figure 2.2 First and second order polynomial approximation.

I. Linear

By weighting the average of the two values used to produce the coordinates of the line the formula:

$$f_1(x) = L_1 f(x_1) + L_2 f(x_2) \qquad \text{where:} \qquad L_1 = \frac{x - x_2}{x_1 - x_2} \qquad L_2 = \frac{x - x_1}{x_2 - x_1}$$

Example 2.1

Compute a 4-decimal place value of ln 9.2 from ln 9.0 = 2.1972, ln 9.5 = 2.2513 by linear Lagrange interpolation and determine the error, using the exact value of ln 9.2 = 2.2192.

Solution:

$x_1 = 9.0$, $x_2 = 9.5$, $f_1 = \ln 9.0 = 2.1972$, $f_2 = \ln 9.5 = 2.2513$; hence we get

$$L_1(x) = \frac{x - 9.5}{-0.5} = -2.0(x - 9.5), \qquad L_1(9.2) = -2.0(-0.3) = 0.6$$

$$L_2(x) = \frac{x - 9.0}{0.5} = 2.0(x - 9.0), \qquad L_2(9.2) = 2 \times 0.2 = 0.4$$

$\ln 9.2 \approx p_1(9.2) = L_1(9.2) f_1 + L_2(9.2) f_2 = 0.6 \times 2.1972 + 0.4 \times 2.2513 = 2.2188$

The absolute error is $|2.2192 - 2.2188| = 0.0004$

II. Quadratic

By weighting the average of the three points that produce the parabola we can derive the formula:

$$f_2(x) = L_1 f(x_1) + L_2 f(x_2) + L_3 f(x_3)$$

where:

$$L_1 = \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} \qquad L_2 = \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} \qquad L_3 = \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}$$

Example 2.2

Compute ln 9.2 from the data in the previous example 2.1 and the additional third value ln 11.0 = 2.3979.

Solution:

$$L_1(x) = \frac{(x-9.5)(x-11.0)}{(9.0-9.5)(9.0-11.0)} = x^2 - 20.5x + 104.5 \qquad \Rightarrow L_1(9.2) = 0.5400$$

$$L_2(x) = \frac{(x-9.0)(x-11.0)}{(9.5-9.0)(9.5-11.0)} = -\frac{1}{0.75}(x^2 - 20x + 99) \qquad \Rightarrow L_2(9.2) = 0.4800$$

$$L_3(x) = \frac{(x-9.0)(x-9.5)}{(11.0-9.0)(11.0-9.5)} = \frac{1}{3}(x^2 - 18.5x + 85.5) \qquad \Rightarrow L_3(9.2) = -0.0200$$

ln 9.2 ≈ $p_2(9.2)$ = 0.5400 × 2.1972 + 0.4800 ×2.2513 – 0.0200×2.3979 = 2.2192.

The absolute error is $|2.2192 - 2.2192| = 0.0000$

III. General Lagrange Interpolating Polynomial

In general, the Lagrange polynomial can be represented as:

$$f_{n-1}(x) = \sum_{i=1}^{n} L_i(x) f(x_i) \qquad \text{where} \qquad L_i(x) = \prod_{\substack{j=1 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j}$$

Example 2.3

Find the Lagrange interpolation polynomial that takes the values prescribed below

x_k	0	1	2	4
$f(x_k)$	1	1	2	5

Solution

$$\boldsymbol{P_3(x)} = \sum_{k=0}^{3} L_{3,k}(x)\; f(x_k)$$

$$\boldsymbol{P_3(x)} = \frac{(x-1)(x-2)(x-4)}{(0-1)(0-2)(0-4)}(1) + \frac{(x-0)(x-2)(x-4)}{(1-0)(1-2)(1-4)}(1)$$

$$+ \frac{(x-0)(x-1)(x-4)}{(2-0)(2-1)(2-4)}(2) + \frac{(x-0)(x-1)(x-2)}{(4-0)(4-1)(4-2)}(5)$$

When working with grids having large numbers of intervals one typically assigns a set of low degree (n = 1, 2, or 3) basis functions to each adjacent set of n+1 = 2, 3, or 4 nodes.

Example 2.4

The following table gives the value of density of saturated water for various temperatures of saturated stream.

Temp ^oC (= T)	100	150	200	250
Density kg/m^3 (= d)	958	917	865	799

1) Use third order Lagrange interpolating polynomials to correlate density as a function of temperature.

2) Find the densities when the temperatures are 130^{o}C .

Solution

$$L_1 = \frac{(T-150)(T-200)(T-250)}{(100-150)(100-200)(100-250)} = -1.3333\times10^{-6}T^3 + 4\times10^{-3}T^2 - 0.1566T + 10$$

$$L_2 = \frac{(T-100)(T-200)(T-250)}{(150-100)(150-200)(150-250)} = 4\times10^{-6}T^3 - 2.2\times10^{-3}T^2 + 0.38T - 20$$

$$L_3 = \frac{(T-100)(T-150)(T-250)}{(200-100)(200-150)(200-250)} = -4\times10^{-6}T^3 + 2\times10^{-3}T^2 - 0.31T + 15$$

$$L_4 = \frac{(T-100)(T-150)(T-200)}{(250-100)(250-150)(250-200)} = 1.3333\times10^{-6}T^3 + 6\times10^{-4}T^2 - 0.08666 - 4$$

$$f_4(x) = L_1 f(x_1) + L_2 f(x_2) + L_3 f(x_3) + L_4 f(x_4) = -4\times10^{-6}T^3 - 4\times10^{-4}T^2 - 0.53T + 1019$$

$$f_4(130) = 934.5520$$

Example 2.5

Use Lagrange global interpolation by one polynomial and piecewise polynomial interpolation with quadratic for the following nodes.

x_k	0	1	2	4	5
$f(x_k)$	0	16	48	88	0

Solution

Global interpolation by one polynomial: $P(x) = \sum_{k=0}^{4} L_{4,k}(x) \;\; f(x_k)$

$$P_4(x) = \frac{(x-1)(x-2)(x-4)(x-5)}{(0-1)(0-2)(0-4)(0-5)}(0) + \frac{(x-0)(x-2)(x-4)(x-5)}{(1-0)(1-2)(1-4)(1-5)}(16)$$

$$+ \frac{(x-0)(x-1)(x-4)(x-5)}{(2-0)(2-1)(2-4)(2-5)}(48) + \frac{(x-0)(x-1)(x-2)(x-5)}{(4-0)(4-1)(4-3)(4-5)}(88) + 0$$

$$= -4.6667x^4 + 33.33x^3 - 59.3333x^2 + 46.6667x$$

Piecewise polynomial interpolation with quadratic

$$P_2(x) = \frac{(x-1)(x-2)}{(0-1)(0-2)}(0) + \frac{(x-0)(x-2)}{(1-0)(1-2)}(16) + \frac{(x-0)(x-1)}{(2-0)(2-1)}(48); \quad 0 \le x \le 2$$

$$= 8x + 8x^2$$

$$P_2(x) = \frac{(x-4)(x-5)}{(2-4)(2-5)}(48) + \frac{(x-2)(x-5)}{(4-2)(4-5)}(88) + \frac{(x-2)(x-4)}{(5-2)(5-4)}(0); \quad 2 \le x \le 5$$

$$= -280 + 236x - 36x^2$$

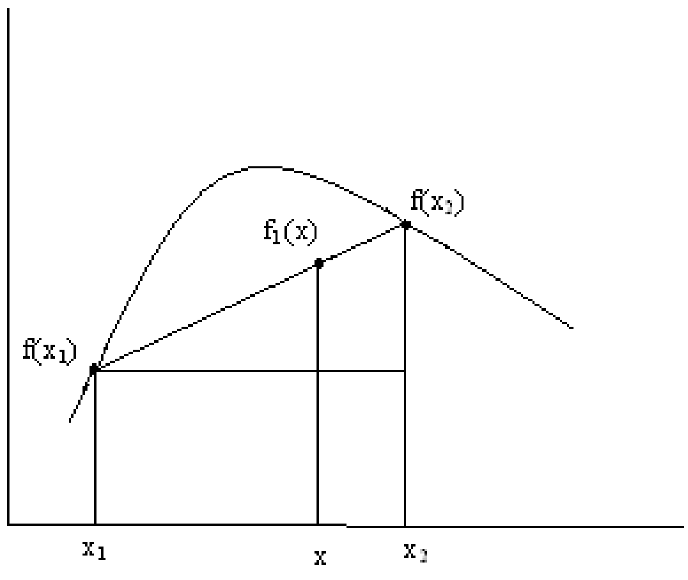
2.1 Newton Divided Difference Interpolating

The Lagrangian interpolation polynomials are useful in discussions on numerical integration. An alternative in interpolation is ‘*Newton’s Divided Difference Interpolation*’. It involves fewer arithmetical operations.

Another advantage of Newton’s rests with the following scenario. Suppose we need to improve the accuracy and increase the number of grid points. From the forms of Lagrange interpolation polynomials, all the terms have to be evaluated once again, and this is a huge amount of work if the number of points is large. Newton’s does not suffer from this drawback, and just one additional term needs to be computed.

I. Linear Interpolation

Consider the diagram below in which a curve is modeled (poorly) by $\overline{x_1 x_2}$:



Using similar triangles the slopes are the same and hence:

$$\frac{f_1(x) - f(x_1)}{x - x_1} = \frac{f(x_2) - f(x_1)}{x_2 - x_1}$$

And thus the coordinate on the curve at x_1 can be approximated by rearranging the above to become:

$$f_1(x) = f(x_1) + \frac{f(x_2) - f(x_1)}{x_2 - x_1}(x - x_1)$$

Example 2.6

Estimate the common logarithm of 10 using linear Newton's interpolation.

(a) Interpolate between log 8 = 0.9030900 and log 12 = 1.0791812.

(b) Interpolate between log 9 = 0.9542425 and log 11 = 1.0413927.

For each of the interpolations, compute the percent relative error based on the true value.

<u>Solution</u>

a) $f_1(10) = 0.90309 + \frac{1.0791812 - 0.90309}{12 - 8}(10 - 8) = 0.991136$

$$\varepsilon_t = \frac{1 - 0.991136}{1} \times 100\% = 0.886\%$$

b) $f_1(10) = 0.9542425 + \frac{1.0413927 - 0.9542425}{11 - 9}(10 - 9) = 0.997818$

$$\varepsilon_t = \frac{1 - 0.997818}{1} \times 100\% = 0.218\%$$

II. Quadratic Interpolation

To reduce the error, a quadratic interpolation that introduces some curvature into the interpolation is used. The form:

$$f_2(x) = b_1 + b_2(x - x_1) + b_3(x - x_1)(x - x_2)$$

Let $x = x_1$ to produce:

$$b_1 = f(x_1)$$

Let $x = x_2$ and use the previous identity to produce:

$$f(x_2) = f(x_1) + b_2(x_2 - x_1) + b_3(x - x_1)(x - x_2) \quad \Rightarrow \quad b_2 = \frac{f(x_2) - f(x_1)}{x_2 - x_1}$$

And again by substitution of b_1 and b_2 we derive that:

$$f(x_3) = f(x_1) + \frac{f(x_2) - f(x_1)}{x_2 - x_1}(x_3 - x_1) + b_3(x_3 - x_1)(x_3 - x_2) \Rightarrow b_3 = \frac{\frac{f(x_3) - f(x_2)}{x_3 - x_2} - \frac{f(x_2) - f(x_1)}{x_2 - x_1}}{x_3 - x_1}$$

$$f_2(x) = f(x_1) + \frac{f(x_2) - f(x_1)}{x_2 - x_1}(x - x_1) + \frac{\frac{f(x_3) - f(x_2)}{x_3 - x_2} - \frac{f(x_2) - f(x_1)}{x_2 - x_1}}{x_3 - x_1}(x - x_1)(x - x_2)$$

Example 2.7

Fit a second-order Newton's Interpolating polynomial to estimate log 10 using the data from Example 2.7 at x = 8, 9, and 11. Compute the true percent relative error.

Solution

First, order the points

$x_1 = 9$ $f(x_1) = 0.9542425$

$x_2 = 11$ $f(x_2) = 1.0413927$

$x_3 = 8$ $f(x_3) = 0.9030900$

$b_1 = 0.9542425$

$$b_2 = \frac{1.0413927 - 0.9542425}{11 - 9} = 0.0435751$$

$$b_3 = \frac{\frac{0.9030900 - 1.0413927}{8 - 11} - 0.0435751}{8 - 9} = \frac{0.0461009 - 0.0435751}{8 - 9} = -0.0025258$$

Substituting these values yields the quadratic formula

$$f_2(x) = 0.9542425 + 0.0435751(x-9) - 0.0025258(x-9)(x-11)$$

which can be evaluated at x = 10 for

$$f_2(10) = 0.9542425 + 0.0435751(10-9) - 0.0025258(10-9)(10-11) = 1.0003434$$

$$\varepsilon_t = \frac{1 - 1.0003434}{1} \times 100\% = 0.03434\%$$

III. General form of Newton Divided Difference Interpolating Polynomial

In general, if we find the finite differences defined as:

$$f[x_i, x_j] = \frac{f(x_i) - f(x_j)}{x_i - x_j}$$

$$f[x_i, x_j, x_k] = \frac{f[x_i, x_j] - f[x_j, x_k]}{x_i - x_k}$$

M

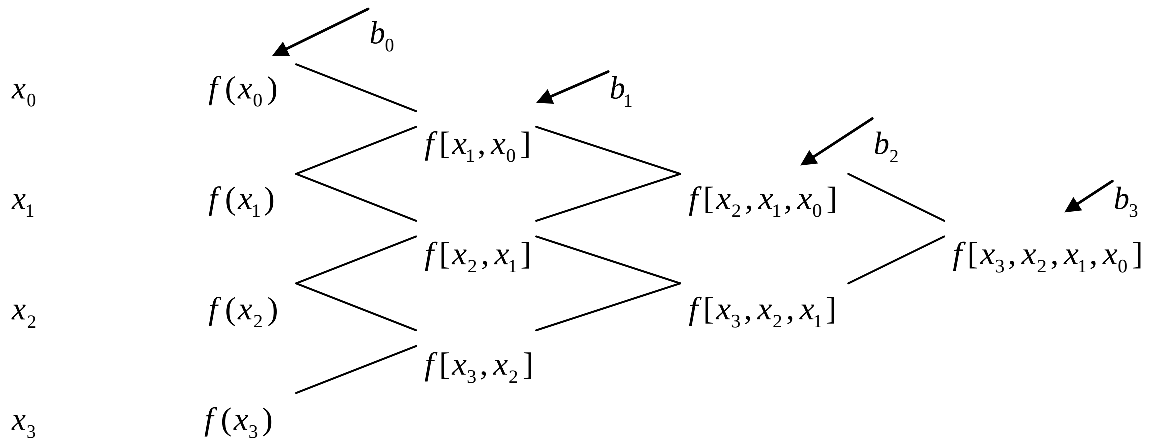
$$f[x_n, x_{n-1}, ..., x_1] = \frac{f[x_n, x_{n-1}, ..., x_2] - f[x_{n-1}, x_{n-2}, ..., x_1]}{x_n - x_1}$$

Then the general Newton Interpolating Polynomial of order n – 1with n data points is defined as:

$$f_{n-1}(x) = b_1 + b_2(x - x_1) + b_3(x - x_1)(x - x_2) + ... + b_n(x - x_1)(x - x_2)...(x - x_{n-1})$$

$$
\begin{aligned}
b_1 &= f(x_1) \\
b_2 &= f[x_2, x_1] \\
b_3 &= f[x_3, x_2, x_1] \\
&\text{M} \\
b_n &= f[x_n, x_{n-1}, \ldots, x_1]
\end{aligned}
$$

Where



For an example of a third order polynomial, given (x_0, y_0), (x_1, y_1), (x_2, y_2), and (x_3, y_3),

$$
\begin{aligned}
f_3(x) = f[x_0] &+ f[x_1, x_0](x - x_0) + f[x_2, x_1, x_0](x - x_0)(x - x_1) \\
&+ f[x_3, x_2, x_1, x_0](x - x_0)(x - x_1)(x - x_2)
\end{aligned}
$$

Example 2.8

The following table gives the value of density of saturated water for various temperatures of saturated stream.

Temp^o C (= T) : 100 150 200 250 300

Density kg/m^3 (= d) **:** 958 917 865 799 712

Using Newton divided difference interpolating find the densities when the temperatures are 130^oC and 275^oC respectively.

Solution

i	T	D	$f[x_{i+1},x_i]$	$f[x_{i+2},x_{i+1},x_i]$	$f[x_{i+3},x_{i+2},x_{i+1},x_i]$	$f[x_{i+4},x_{i+3},x_{i+2},x_{i+1},x_i]$
1	100	958				
			-0.8200			
2	150	917		-0.0022		
			-1.0400		-4×10^{-6}	
3	200	865		-0.0028		-2.6667×10^{-8}
			-1.3200		-9.333×10^{-6}	
4	250	799		-0.0042		
			-1.7400			
5	300	712				

$$P_4 = 958 - 0.8200 \times (T - 100) - 0.0022 \times (T - 100) * (T - 150) - 4 \times 10^{-6} \times (T - 100) * (T - 150) * (T - 200)$$
$$- 2.6667 \times 10^{-8} \times (T - 100) * (T - 150) * (T - 200) * (T - 250)$$

$$P_4 = 999 - 0.0167T - 0.0051T^2 + 1.4667 \times 10^{-5} T^3 - 2.6667 \times 10^{-8} T^4$$

$$P_4(130) = 934.6864 \text{ kg/m3}$$

$$P_4(275) = 758.7187 \text{ kg/m3}$$

Or by direct substitution

$$P_4(130) = 934.6864 \text{ kg/m3}$$

$P(130)= 958-0.82\times(130-100)-0.0022\times(130-100)\times(130-150) -4\times10^{-6}\times(130-100) \times (130-150) \times(130-200) -2.6667\times10^{-8}\times(130-100) \times(130-150)\times(130-200) \times(130-250)$
$= 934.6864$ kg/m^3

$P(275)=958-0.82\times(275-100)-0.0022\times(275-100)\times(275-150)-4\times10^{-6}\times(275-100) \times(275-150) \times(275-200) -2.6667\times10^{-8} \times(275-100)\times(275-150)\times(275-200) \times(275-250)$
$= 758.7188$ kg/m^3

3.0 What is regression?

Regression analysis gives information on the relationship between a response variable and one or more independent variables to the extent that information is contained in the data. The goal of regression analysis is to express the response variable as a function of the predictor variables.

Once regression analysis relationship is obtained, it can be used to predict values of the response variable, identify variables that most affect response, or verify hypothesized casual models of the response.

3.1 Linear regression

Linear regression is the most popular regression model. In this model we wish to predict response to n data points (x_1,y_1), (x_2,y_2), ..., (x_n, y_n) data by a regression model given by.

$$y = a_0 + a_1 x$$

Where a_0 and a_1 are the constants of the regression model.

A measure of goodness of fit, that is, how $a_0 + a_1 x$ predicts the response variable y is the magnitude of the residual, ε_i at each of the n data points.

$$\varepsilon_i = y_i - (a_0 + a_1 x_i)$$

Ideally, if all the residuals ε_i are zero, one may have found an equation in which all the points lie on the model. Thus, ***minimization*** of the residual is an objective of obtaining regression coefficients.

The most popular method to minimize the residual is the least squares method, where the estimates of the constants of the models are chosen such that the sum of the squared residuals is minimized, that is minimize $\sum_{i=1}^{n} \varepsilon_i^{\,2}$.

Let us use the least squares criterion where we minimize

$$S_r = \sum_{i=1}^{n} \varepsilon_i^{\,2} = \sum_{i=1}^{n} \left(y_i - a_0 - a_1 x_i\right)^2$$

S_r is called the sum of the square of the residuals.

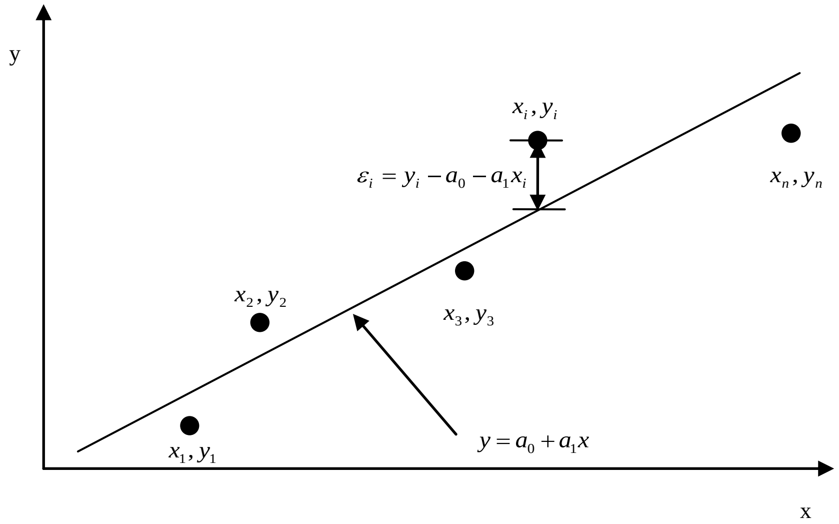


Figure 3.1 Linear regression of y vs. x data showing residuals at a typical point, x_i.

To find a_0 and a_1, we minimize S_r with respect to a_0 and a_1:

$$\frac{\partial S_r}{\partial a_0} = 2\sum_{i=1}^{n}(y_i - a_0 - a_1 x_i)(-1) = 0$$

$$\frac{\partial S_r}{\partial a_1} = 2\sum_{i=1}^{n}(y_i - a_0 - a_1 x_i)(-x_i) = 0$$

Giving

$$-\sum_{i=1}^{n} y_i + \sum_{i=1}^{n} a_0 + \sum_{i=1}^{n} a_1 x_i = 0$$

$$-\sum_{i=1}^{n} y_i x_i + \sum_{i=1}^{n} a_0 x_i + \sum_{i=1}^{n} a_1 x_i^2 = 0$$

Noting that $\sum_{i=1}^{n} a_0 = a_0 + a_0 + \ldots + a_0 = na_0$

$$na_0 + a_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i \qquad (3.1)$$

$$a_0 \sum_{i=1}^{n} x_i + a_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i \qquad (3.2)$$

Solving the above equations gives:

$$a_1 = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$

$$a_0 = \frac{\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i y_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$

Or from equation (3.2)

$$a_o = \frac{\sum_{i=1}^{n} y_i}{n} - \frac{a_1 \sum_{i=1}^{n} x_i}{n} = \bar{y} - a_1\bar{x}$$

Example 3.1

The following *y* vs. *x* data is given

x	1	7	13	19	25
y	1	49	169	361	625

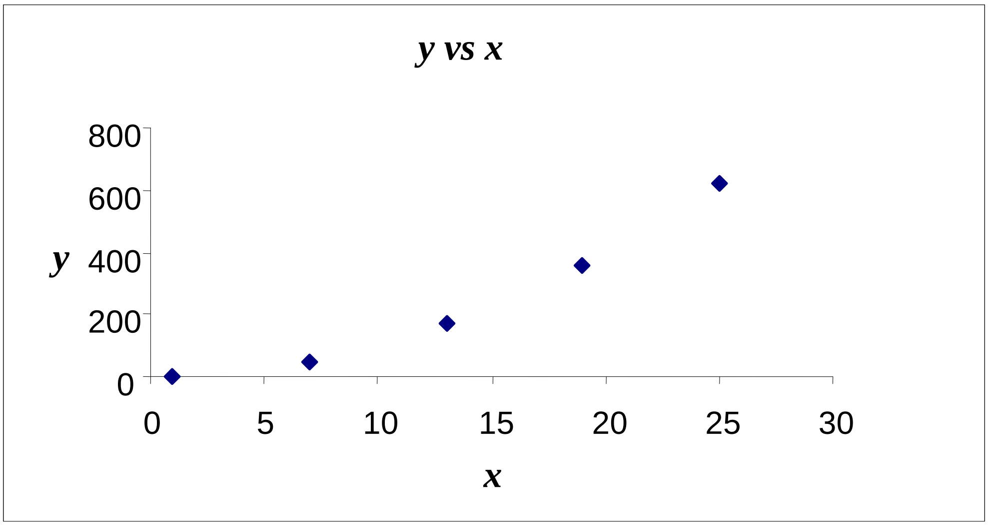


Figure 3.1 Data points of the y vs x data

Although $y = x^2$ is an exact fit to the data, a scientist thinks that $y = a_0 + a_1 x$ can explain the data. Find constants of the model, a_0, and a_1,

Solution

First find the constants of the assumed model

$$y = a_0 + a_1 x$$

$$a_0 = \bar{y} - a_1 \bar{x}$$

$$n = 5$$

$$\sum_{i=1}^{n} x_i y_i = \sum_{i=1}^{5} x_i y_i = 1\times 1 + 7\times 49 + 13\times 169 + 19\times 361 + 25\times 625 = 25025$$

$$\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{5} x_i^2 = 1^2 + 7^2 + 13^2 + 19^2 + 25^2 = 1205$$

$$\sum_{i=1}^{n} y_i = \sum_{i=1}^{5} y_i = 1 + 49 + 169 + 361 + 625 = 1205$$

$$\sum_{i=1}^{n} x_i = \sum_{i=1}^{5} x_i = 1+7+13+19+25 = 65$$

$$a_1 = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$

$$a_0 = \bar{y} - a_1 \bar{x}$$

$$a_1 = \frac{5(25025)-(65)(1205)}{5(1205)-(65)^2} = 26$$

$$a_0 = \bar{y} - a_1\bar{x} = \frac{1205}{5} - 26\frac{65}{5} = (241) - 26(13) = -97$$

This gives

$$y = a_0 + a_1 x$$

$$y = -97 + 26x$$

Example 3.2

The following table gives the value of density of saturated water for various temperatures of saturated stream.

Temp$^o C$ (= T)	100	150	200	250	300
Density kg/m^3 (= D)	958	917	865	799	712

a) Use curve fitting to fit the results to a first-order polynomial $D = A + BT$.

b) Find the densities when the temperatures are 130$^o C$ and 275$^o C$ respectively.

Solution:

a_0 and a_1 can be computed by constructing the following table:

T_i	D_i	T_i^2	$T_i D_i$
100	958	10000	95800
150	917	22500	137550
200	865	40000	173000
250	799	62500	199750
300	712	90000	213600
∑1000	4251	225000	819700

$$a_1 = \frac{5\times 819700 - 1000\times 4251}{5\times 225000 - (1000)^2} = -1.22$$

$$a_0 = \frac{4251}{5} - a_1\frac{1000}{5} = 1094.2$$

D=1094.2-1.22×T

To compare the predicted values to the experimental values:

T_i	D_i	D_i(estimated) D=1094.2-1.22×T
100	958	972.2
150	917	911.2
200	865	850.2
250	799	789.2
300	712	728.2

D(130)= 1094.2-1.22×130=935.6

D(175)= 1094.2-1.22×175=880.7

3.2 Polynomial Models

Given N data points (x_1, y_1), (x_2, y_2). . . , (x_N, y_N) use least squares method to regress the data to an n^{th} order polynomial.

In the development, we use n as the degree of the polynomial and N as the number of data pairs (x_i, y_i). We will always have $N > n+1$ in the following.

Assume the functional relationship for fitting

$$Y(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$$

with errors defined by

$$e_i = y_i - Y(x_i) = y_i - a_0 - a_1 x_i - a_2 x_i^2 - \cdots - a_n x_i^n ,$$

in which $i = 1, 2, 3, \ldots, N$.

We minimize the sum of error squares,

$$S = \sum_{i=1}^{N} e_i^2 = \sum_{i=1}^{N} (y_i - a_0 - a_1 x_i - a_2 x_i^2 - \cdots - a_n x_i^n)^2 .$$

At the minimum, all the first partial derivatives with respect to a_i's vanish. We have

$$\frac{\partial S}{\partial a_0} = 0 = 2\sum_{i=1}^{N} (y_i - a_0 - a_1 x_i - a_2 x_i^2 - \cdots - a_n x_i^n)(-1) ,$$

$$\frac{\partial S}{\partial a_1} = 0 = 2\sum_{i=1}^{N} (y_i - a_0 - a_1 x_i - a_2 x_i^2 - \cdots - a_n x_i^n)(-x_i) ,$$

$$\frac{\partial S}{\partial a_2} = 0 = 2\sum_{i=1}^{N} (y_i - a_0 - a_1 x_i - a_2 x_i^2 - \cdots - a_n x_i^n)(-x_i^2) ,$$

$$\vdots$$

$$\frac{\partial S}{\partial a_n} = 0 = 2\sum_{i=1}^{N} (y_i - a_0 - a_1 x_i - a_2 x_i^2 - \cdots - a_n x_i^n)(-x_i^n) ,$$

Rearrange them to get

$$a_0 N + a_1 \sum_{i=1}^{N} x_i + a_2 \sum_{i=1}^{N} x_i^2 + \cdots + a_n \sum_{i=1}^{N} x_i^n = \sum_{i=1}^{N} y_i ,$$

$$a_0\sum_{i=1}^{N}x_i + a_1\sum_{i=1}^{N}x_i^{\,2} + a_2\sum_{i=1}^{N}x_i^{\,3} + \cdots + a_n\sum_{i=1}^{N}x_i^{\,n+1} = \sum_{i=1}^{N}x_i y_i\ ,$$

$$a_0\sum_{i=1}^{N}x_i^{\,2} + a_1\sum_{i=1}^{N}x_i^{\,3} + a_2\sum_{i=1}^{N}x_i^{\,4} + \cdots + a_n\sum_{i=1}^{N}x_i^{\,n+2} = \sum_{i=1}^{N}x_i^{\,2} y_i\ ,$$

$$\vdots$$

$$a_0\sum_{i=1}^{N}x_i^{\,n} + a_1\sum_{i=1}^{N}x_i^{\,n+1} + a_2\sum_{i=1}^{N}x_i^{\,n+2} + \cdots + a_n\sum_{i=1}^{N}x_i^{\,2n} = \sum_{i=1}^{N}x_i^{\,n} y_i\ ,$$

or, in matrix form,

$$\begin{bmatrix} N & \sum_{i=1}^{N}x_i & \sum_{i=1}^{N}x_i^{\,2} & \cdots & \sum_{i=1}^{N}x_i^{\,n} \\ \sum_{i=1}^{N}x_i & \sum_{i=1}^{N}x_i^{\,2} & \sum_{i=1}^{N}x_i^{\,3} & \cdots & \sum_{i=1}^{N}x_i^{\,n+1} \\ \sum_{i=1}^{N}x_i^{\,2} & \sum_{i=1}^{N}x_i^{\,3} & \sum_{i=1}^{N}x_i^{\,4} & \cdots & \sum_{i=1}^{N}x_i^{\,n+2} \\ \vdots & & & & \\ \sum_{i=1}^{N}x_i^{\,n} & \sum_{i=1}^{N}x_i^{\,n+1} & \sum_{i=1}^{N}x_i^{\,n+2} & \cdots & \sum_{i=1}^{N}x_i^{\,2n} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{N}y_i \\ \sum_{i=1}^{N}x_i y_i \\ \sum_{i=1}^{N}x_i^{\,2} y_i \\ \vdots \\ \sum_{i=1}^{N}x_i^{\,n} y_i \end{bmatrix}. \qquad (3.3)$$

Equations (3.3) represent a linear system. However, this system is usually ill-conditioned and round-off errors can distort the solution of a_i's. Up to degree-3 or 4, the problem is not too great. It is very infrequent to use a degree higher than 4.

Example 3.3

Rotameter calibration data (flow rate versus Rotameter reading) are as follows:

Rotameter Reading R	10	30	50	70	90
Flow rate V(L/min)	20	52.1	84.6	118.3	151

a) Using curve fitting to fit the calibration data to second order polynomial.

b) Calculate the flowrate (V) at rotameter reading R=73.

Solution:

a) 2nd order polynomial

$$S_r = \sum_{i=1}^{n}\varepsilon_i^2 = \sum_{i=1}^{n}(y_i - a_o - a_1x_i - a_2x_i)^2$$

$$\frac{dS_r}{da_o} = 2\sum_{i=1}^{n}(y_i - a_o - a_1x_i - a_2x_i^2)\times(-1) = 0$$

$$\frac{dS_r}{da_1} = 2\sum_{i=1}^{n}(y_i - a_o - a_1x_i - a_2x_i^2)\times(-x_i) = 0 \qquad (1)$$

$$\frac{dS_r}{da_2} = 2\sum_{i=1}^{n}(y_i - a_o - a_1x_i - a_2x_i^2)\times(-x_i^2) = 0$$

Re arranging above equations

$$a_o n + a_1\sum_{i=1}^{n} x_i + a_2\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} y_i$$

$$a_o\sum_{i=1}^{n} x_i + a_1\sum_{i=1}^{n} x_i^2 + a_2\sum_{i=1}^{n} x_i^3 = \sum_{i=1}^{n} x_i y_i \qquad (2)$$

$$a_o\sum_{i=1}^{n} x_i^2 + a_1\sum_{i=1}^{n} x_i^3 + a_2\sum_{i=1}^{n} x_i^4 = \sum_{i=1}^{n} x_i^2 y_i$$

Making required table

	R	V	R^2	R^3	R^4	RV	R^2y
	10	20	100	1000	10000	200	2000
	30	52.1	900	27000	810000	1563	46890
	50	84.6	2500	125000	6250000	4230	211500
	70	118.3	4900	343000	2401000	8281	579670
	90	151	8100	729000	6561000	13590	1223100
$\sum$	250	426	16500	1225000	9669000	27864	2063160

By substitution in equation 2

$$5a_o + 250a_1 + 16500a_2 = 426$$

$$250a_o + 16500a_1 + 1225000a_2 = 27864$$

$$16500a_o + 1225000a_1 + 96690000a_2 = 2063160$$

Solving above equation simultaneously gives;

ao =3.8786 , a1 =1.5981 , a2 = 4.2857×10^{-4}

then

$$V = 3.8786 + 1.5981\times \mathrm{R} + 4.2857\times10^{-4}\times \mathrm{R}^2$$

B)

$$V(73) = 3.8786 + 1.5981\times 73 + 4.2857\times10^{-4}\times 73^2 = 122.83$$

3.3 Nonlinear Data

Whenever data from experimental tests are not linear, we need to fit to them some function other than a first-degree polynomial. Popular forms that are tried are the power form

$$y = ax^b$$

or the exponential form

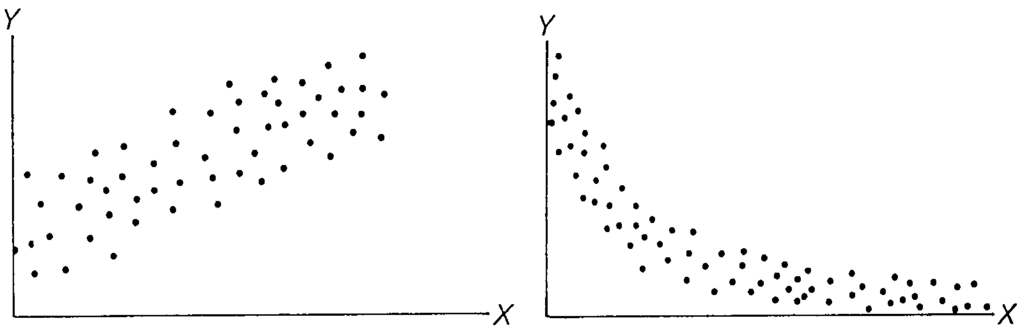
$$y = ae^{bx}.$$

Since such nonlinear equations are much more difficult to solve than linear equations, they are usually linearized by taking logarithms before determining the parameters:

$$\ln y = \ln a + b \ln x,$$

or

$$\ln y = \ln a + bx.$$



linear non-linear

Figure 3.2 Linear vs non-linear data

In cases when such linearization of the function is not desirable, or when no method of linearization can be discovered, graphical methods are frequently used; one merely plots the experimental values and sketches in a curve that seems to fit well.

Example 3.4

The progress of a homogeneous chemical reaction is followed and it is desired to evaluate the rate constant and the order of the reaction. The rate law expression for the reaction is known to follow the power function form $-r \ = kC^n$

Use the data provided in the table to obtain n and k.

C_A(gmol/l)	4	2.25	1.45	1.0	0.65	0.25	0.006
$-r_A$(gmol/l $\cdot$ s)	0.398	0.298	0.238	0.198	0.158	0.098	0.048

Solution

Taking the natural log of both sides of Equation, we obtain

$$\ln(-r\ ) = \ln(k) + n\ln(C\ )$$

Let

$$z = \ln(-r)$$
$$w = \ln(C)$$
$$a_0 = \ln(k) \quad \text{implying that} \quad k = e^{a_0}$$
$$a_1 = n$$

We get

$$z = a_0 + a_1 w$$

This is a linear relation between z and w, where

$$a_1 = \frac{n\sum_{i=1}^{n} w_i z_i - \sum_{i=1}^{n} w_i \sum_{i=1}^{n} z_i}{n\sum_{i=1}^{n} w_i^2 - \left(\sum_{i=1}^{n} w_i\right)^2}$$

$$a_0 = \left(\frac{\sum_{i=1}^{n} z_i}{n}\right) - a_1\left(\frac{\sum_{i=1}^{n} w_i}{n}\right)$$

Table: Kinetics rate law using power function

i	C	$-r$	w	z	$w \times z$	w^2
1	4	0.398	1.3863	-0.92130	-1.2772	1.9218
2	2.25	0.298	0.8109	-1.2107	-0.9818	0.65761
3	1.45	0.238	0.3716	-1.4355	-0.5334	0.13806
4	1	0.198	0.0000	-1.6195	0.0000	0.00000
5	0.65	0.158	-0.4308	-1.8452	0.7949	0.18557
6	0.25	0.098	-1.3863	-2.3228	3.2201	1.9218
7	0.006	0.048	-5.1160	-3.0366	15.535	26.173
$\sum_{i=1}^{7}$			-4.3643	-12.391	16.758	30.998

$$n = 7$$
$$\sum_{i=1}^{7} w_i = -4.3643$$
$$\sum_{i=1}^{7} z_i = -12.391$$
$$\sum_{i=1}^{7} w_i z_i = 16.758$$
$$\sum_{i=1}^{7} w_i^2 = 30.998$$

From above equations

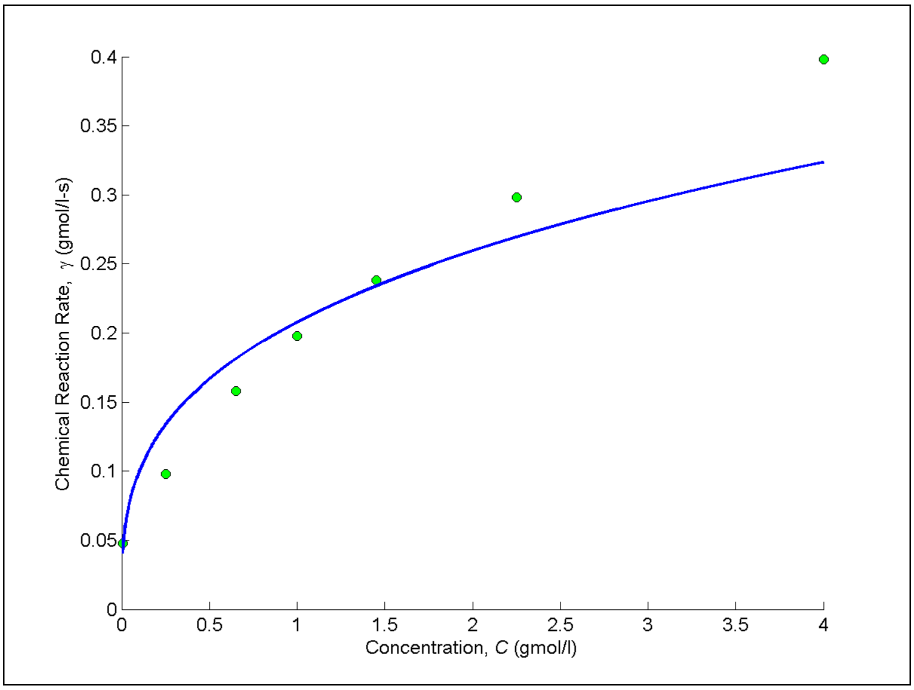
$$a_1 = \frac{7\times(16.758)-(-4.3643)\times(-12.391)}{7\times(30.998)-(-4.3643)^2}$$
$$= 0.31943$$
$$a_0 = \frac{-12.391}{7} - (.31943)\frac{-4.3643}{7}$$
$$= -1.5711$$

Then

$$k = e^{-1.5711}$$
$$= 0.20782$$
$$n = a_1$$
$$= 0.31941$$

Finally, the model of progress of that chemical reaction is

$$-r = 0.20782 \times C^{0.31941}$$



Example 3.5

It is suspected from theoretical considerations that the rate of water flow from a firehouse is proportional to some power of the nozzle pressure. Assume pressure data is more accurate. You are transforming the data.

Flow rate, F (gallons/min)	96	129	135	145	168	235
Pressure, p (psi)	11	17	20	25	40	55

What is the exponent b of the nozzle pressure in the regression model $F = ap^b$

Solution

The linearization of the above data is done as follows.

$$F = ap^b$$

$$\ln(F) = \ln(a) + b\ln(p)$$

$$z = a_0 + bx$$

Where

$$z = \ln(F)$$

$$x = \ln(p)$$

$$a_0 = \ln(a)$$

Implying

$$a = e^{a_0}$$

There is a linear relationship between z and x.

Linear regression constants are given by

$$b = \frac{n\sum_{i=1}^{n} x_i z_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} z_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$

$$a_0 = \frac{\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} z_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i z_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$

Since

$$\mathrm{n} = 6$$

$$\sum_{\mathrm{i}=1}^{6} \mathrm{x_i z_i} = \ln(11)\times\ln(96) + \ln(17)\times\ln(129) + \ln(20)\times\ln(135) + \ln(25)\times\ln(145) + \ln(40)\times\ln(168)$$

$$+ \ln(55)\times\ln(235) \quad = 96.208$$

$$\sum_{i=1}^{6} x_i = \ln(11) + \ln(17) + \ln(20) + \ln(25) + \ln(40) + \ln(55) = 19.142$$

$$\sum_{i=1}^{6} z_i = \ln(96) + \ln(129) + \ln(135) + \ln(145) + \ln(168) + \ln(235) = 29.890$$

$$\sum_{i=1}^{6} x_i^2 = (\ln(11))^2 + (\ln(17))^2 + (\ln(20))^2 + (\ln(25))^2 + (\ln(40))^2 + (\ln(55))^2 = 62.779$$

then

$$b = \frac{6\times 96.208 - 19.142\times 29.890}{6\times 62.779 - 19.142^2}$$

$$= \frac{577.25 - 572.15}{376.67 - 366.41}$$

$$= 0.49721$$

Example 3.6

The following data have been obtained for the decomposition of benzene diazonium chloride to chlorobenzene:

T (K)	313	319	323	328	333
k (s^{-1})	0.0043	0.0103	0.018	0.0355	0.0717

From this data, determine the pre-exponential factor *A* and activation energy *E*, assuming that the rate constant follows an Arrhenius form.

$$k = A\exp\left(\frac{-E}{RT}\right)$$

Solution:

$$\ln k = \ln A - \frac{E}{RT}$$

$$y = \ln k$$

$$x = 1/T$$

$$a_o = \ln A$$

$$a_1 = \frac{-E}{R}$$

We get

$$y = a_o + a_1 x$$

	T (K)	k (s^{-1})	x=1/T	y=ln k	x^2	xy
	313	0.0043	0.00319	-5.44914	1.02073e-05	-0.01741
	319	0.0103	0.00313	-4.57561	9.82695e-06	-0.01434
	323	0.018	0.00310	-4.01738	9.58506e-06	-0.01244
	328	0.0355	0.00305	-3.33822	9.29506e-06	-0.01018
	333	0.0717	0.00300	-2.63526	9.01803e-06	-0.00791
$\sum$			0.01548	-20.0156	4.79324e-05	-0.06228

$$a_1 = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2} = \text{-}14612$$

$$a_o = \bar{y} - a_1\bar{x} = 41.2272$$

$$a_o = \ln A = 41.2272 \Rightarrow$$

$$A = \exp(40.2272) = 8.0303\times 10^{17}$$

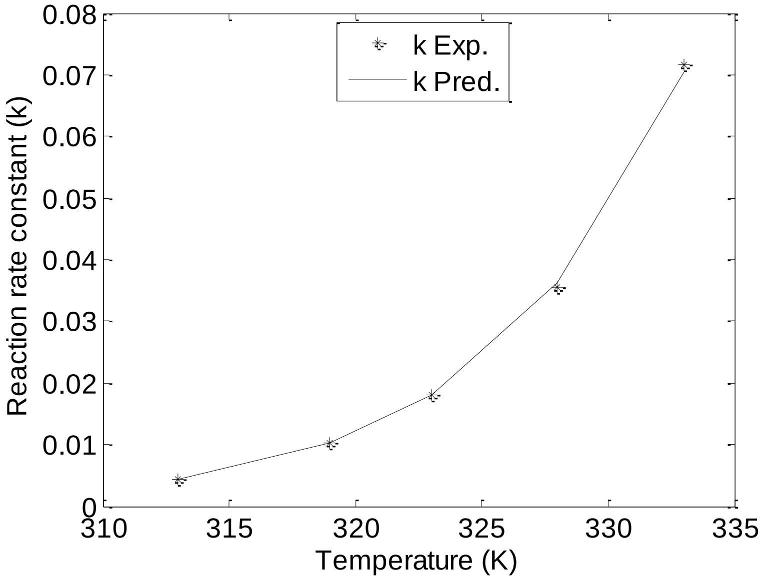
$$a_1 = -E/R \Rightarrow$$

$$E = -a_1 \times R = -(-14612)\times 8.314 = 121480$$

A Matlab program for solving example 3.6 is listed in Table 3.1.

Table (3.1) Matlab code and results for solution example (3.6)	
Matlab Code	**T=[313,319,323,328,333];** **K=[0.0043,0.0103,0.018,0.0355,0.0717];** **x=1./T;** **y=log(K);** **Poly=polyfit(x,y,1);** **E=-Poly(1)*8.314** **Ao=exp(Poly(2))**
Results	**E =** **1.2148e+05** **Ao =** **8.0303e+17**

The comparison between experimental and predicted k values is shown in below figure:



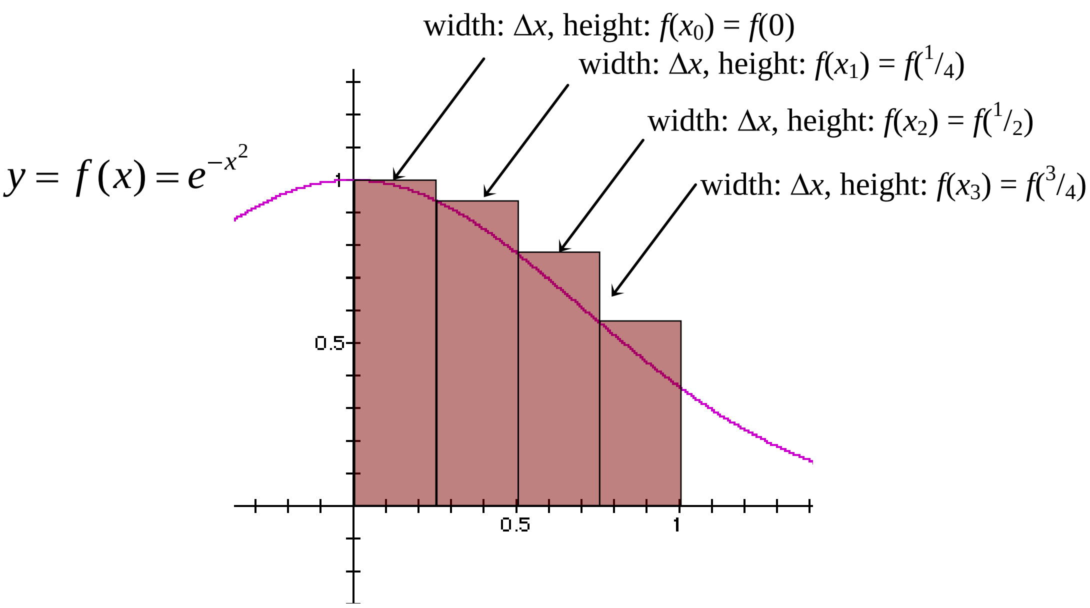
Numerical Integration Approximation.

Integration is the process of measuring the area under a function plotted on a graph. Sometimes, the evaluation of expressions involving these integrals can become daunting, if not indeterminate. For this reason, a wide variety of numerical methods have been developed to find the integral.

Here we discuss six different methods for approximating the value of a definite integral. Each method revolves around associating a definite integral with area under a curve. The first three use areas of rectangles, the fourth uses areas of trapezoids, and the final approximation technique uses areas of shapes that include a portion of a parabola.

4.1 Left-Endpoint Approximation

On each of the four subintervals shown below, we create a rectangle whose width is the length of the subdivision and whose height is determined by the function value at the ***left*** endpoint of each subdivision.



The sum of the areas of the four rectangles represents our approximation for the area under the curve and therefore represents an approximation for the value of the definite integral:

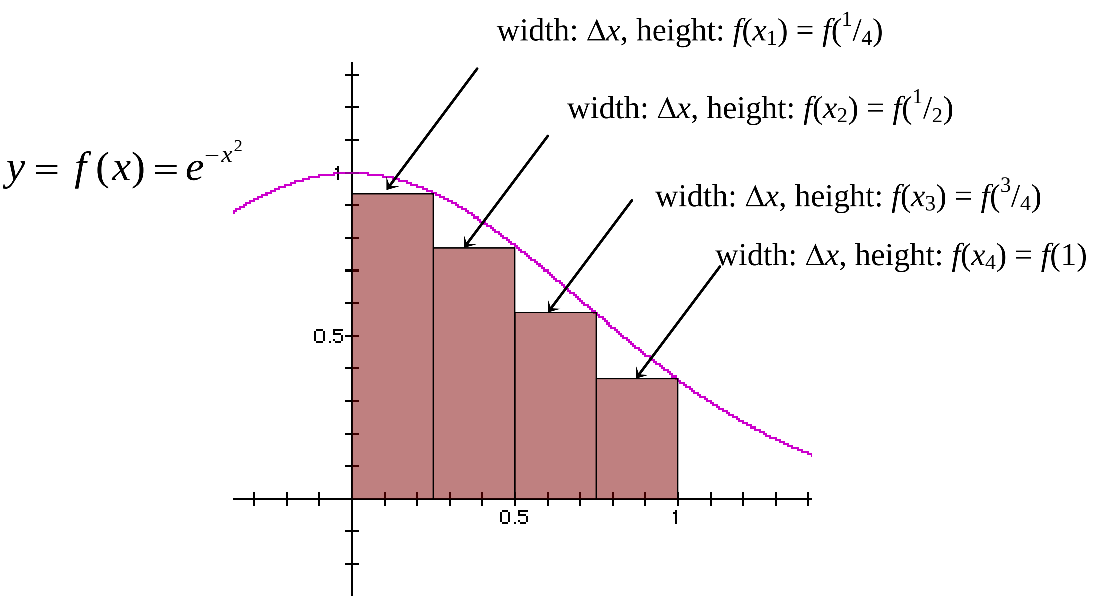
$$\int_0^1 e^{-x^2}dx \approx \Delta x \cdot f(x_0) + \Delta x \cdot f(x_1) + \Delta x \cdot f(x_2) + \Delta x \cdot f(x_3)$$
$$\approx \Delta x\left(f(x_0) + f(x_1) + f(x_2) + f(x_3)\right)$$
$$\approx \Delta x \sum_{i=0}^{3} f(x_i)$$

This same sequence of steps can be generalized for left-endpoint approximation of the definite integral $\int_a^b f(x)dx$ using n subdivisions:

$$\int_a^b f(x)dx \approx \Delta x \cdot f(x_0) + \Delta x \cdot f(x_1) + \Lambda + \Delta x \cdot f(x_{n-2}) + \Delta x \cdot f(x_{n-1})$$
$$\approx \Delta x\left(f(x_0) + f(x_1) + \Lambda + f(x_{n-2}) + f(x_{n-1})\right)$$
$$\approx \Delta x \sum_{i=0}^{n-1} f(x_i)$$

4.2 Right-Endpoint Approximation

Again we create rectangles whose widths are each the length of a subdivision, but here each height is determined by the function value at the ***right*** endpoint of each subinterval.



The sum of the areas of these four rectangles represents a right-endpoint approximation for the area under the curve and therefore is an approximation for the value of the definite integral:

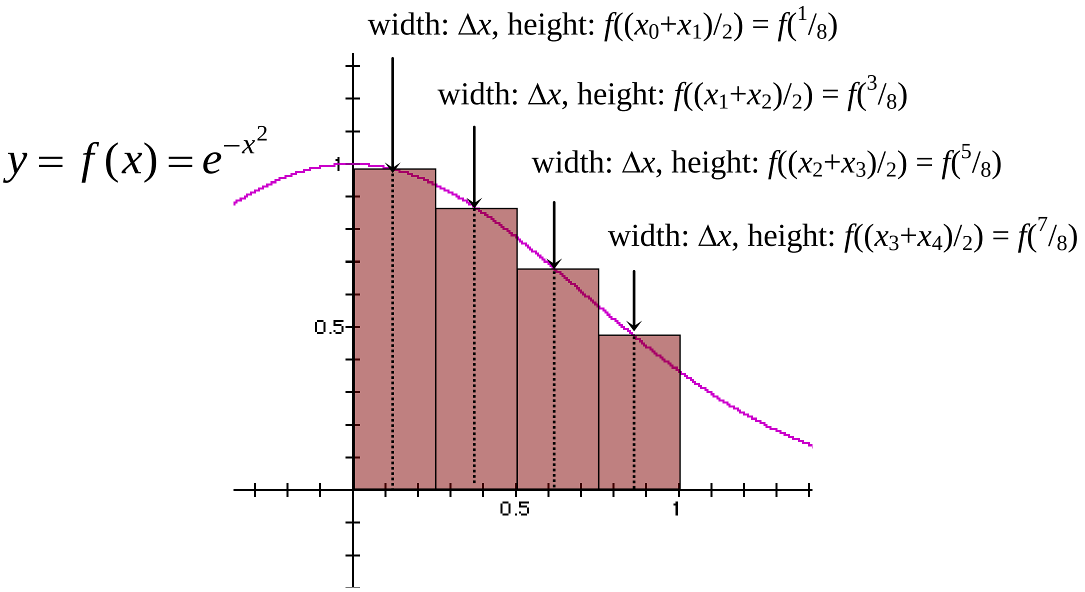
$$\int_0^1 e^{-x^2}dx \approx \Delta x\cdot f(x_1)+\Delta x\cdot f(x_2)+\Delta x\cdot f(x_3)+\Delta x\cdot f(x_4)$$
$$\approx \Delta x\big(f(x_1)+f(x_2)+f(x_3)+f(x_4)\big)$$
$$\approx \Delta x\sum_{i=1}^{4} f(x_i)$$

This same sequence of steps can be generalized for right-endpoint approximation of the definite integral $\int_a^b f(x)dx$ using n subdivisions:

$$\int_a^b f(x)dx \approx \Delta x\cdot f(x_1)+\Delta x\cdot f(x_2)+\Lambda\ +\Delta x\cdot f(x_{n-1})+\Delta x\cdot f(x_n)$$
$$\approx \Delta x\big(f(x_1)+f(x_2)+\Lambda\ +f(x_{n-1})+f(x_n)\big)$$
$$\approx \Delta x\sum_{i=1}^{n} f(x_i)$$

4.3 Midpoint Approximation

For a third time we create rectangles each of whose width is the length of the subdivision, but now each height is determined by the function value at the ***midpoint*** of each subdivision.



The sum of the areas of these four rectangles represents a midpoint approximation for the area under the curve and therefore is another approximation for the value of the definite integral:

$$\int_0^1 e^{-x^2}dx \approx \Delta x \cdot f\left(\frac{x_0+x_1}{2}\right)+\Delta x \cdot f\left(\frac{x_1+x_2}{2}\right)+\Delta x \cdot f\left(\frac{x_2+x_3}{2}\right)+\Delta x \cdot f\left(\frac{x_3+x_4}{2}\right)$$

$$\approx \Delta x\left(f\left(\frac{x_0+x_1}{2}\right)+ f\left(\frac{x_1+x_2}{2}\right)+ f\left(\frac{x_2+x_3}{2}\right)+ f\left(\frac{x_3+x_4}{2}\right)\right)$$

$$\approx \Delta x\sum_{i=0}^{3} f\left(\frac{x_i+x_{i+1}}{2}\right)$$

This same sequence of steps can be generalized for midpoint approximation of the definite integral $\int_a^b f(x)dx$ using n subdivisions:

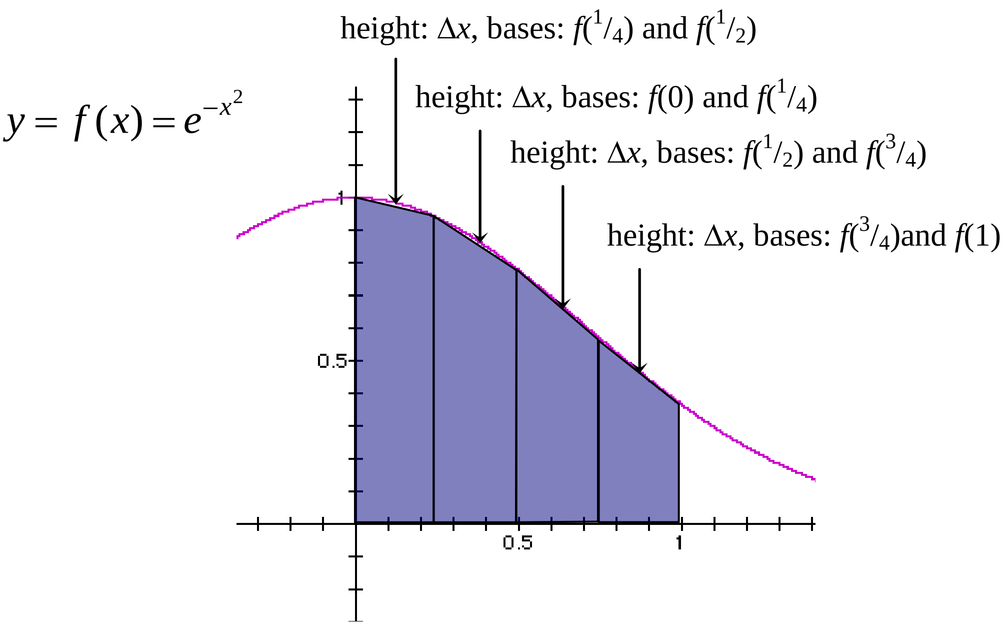
$$\int_a^b f(x)dx \approx \left(\Delta x \cdot f\left(\frac{x_0+x_1}{2}\right)\right)+\left(\Delta x \cdot f\left(\frac{x_1+x_2}{2}\right)\right)+\Lambda +\left(\Delta x \cdot f\left(\frac{x_{n-1}+x_n}{2}\right)\right)+\left(\Delta x \cdot f\left(\frac{x_n+x_{n+1}}{2}\right)\right)$$

$$\approx \Delta x\left(f\left(\frac{x_0+x_1}{2}\right)+ f\left(\frac{x_1+x_2}{2}\right)+\Lambda + f\left(\frac{x_{n-1}+x_n}{2}\right)+ f\left(\frac{x_n+x_{n+1}}{2}\right)\right)$$

$$\approx \Delta x\sum_{i=0}^{n-1} f\left(\frac{x_i+x_{i+1}}{2}\right)$$

4.4 Trapezoidal Rule

Trapezoidal rule is based on the Newton-Cotes formula that if one approximates the integrand by an n^{th} order polynomial, then the integral of the function is approximated by the integral of that n^{th} order polynomial. Integrating polynomials is simple and is based on the calculus formula. The height of each trapezoid is the length of the subdivision. The two bases of each trapezoid correspond to the values of the function at the endpoints of the subinterval on which the trapezoid has been drawn.



It may be useful to remove the first of these trapezoids and rotate it into a more conventional orientation as we calculate its area.

The sum of the areas of these four trapezoids represents an approximation for the area under the curve and therefore is one more approximation for the value of the definite integral:

$$\begin{aligned}\int_0^1 e^{-x^2}dx &\approx \left(\frac{1}{2}\Delta x\cdot\left(f(x_0)+f(x_1)\right)\right)+\left(\frac{1}{2}\Delta x\cdot\left(f(x_1)+f(x_2)\right)\right)+\\ &\left(\frac{1}{2}\Delta x\cdot\left(f(x_2)+f(x_3)\right)\right)+\left(\frac{1}{2}\Delta x\cdot\left(f(x_3)+f(x_4)\right)\right)\\ &\approx \frac{1}{2}\Delta x\left(f(x_0)+2f(x_1)+2f(x_2)+2f(x_3)+f(x_4)\right)\\ &\approx \frac{1}{2}\Delta x\sum_{i=0}^{3}\left(f(x_i)+f(x_{i+1})\right)\end{aligned}$$

This same sequence of steps can be generalized for trapezoid approximation of the definite integral $\int_a^b f(x)dx$ using n subdivisions:

$$\begin{aligned}\int_a^b f(x)dx &\approx \left(\frac{1}{2}\Delta x\cdot\left(f(x_0)+f(x_1)\right)\right)+\left(\frac{1}{2}\Delta x\cdot\left(f(x_1)+f(x_2)\right)\right)+\Lambda\\ &+\left(\frac{1}{2}\Delta x\cdot\left(f(x_{n-1})+f(x_{n-1})\right)\right)+\left(\frac{1}{2}\Delta x\cdot\left(f(x_{n-1})+f(x_n)\right)\right)\\ &\approx \frac{1}{2}\Delta x\left(f(x_0)+2f(x_1)+2f(x_2)+\Lambda\right.\\ &\left.+2f(x_{n-2})+2f(x_{n-1})+f(x_n)\right)\end{aligned}$$

Single Segment Trapezoidal Rule

$$\boxed{\int_a^b f(x)dx \approx \frac{1}{2}\Delta x\left(f(x_i)+f(x_{i+1})\right)}$$

Multiple Segments Trapezoidal Rule

$$\boxed{\int_a^b f(x)dx \approx \frac{1}{2}\Delta x\sum_{i=0}^{n-1}\left(f(x_i)+f(x_{i+1})\right)}$$

Example 4.1

Evaluate the integral $I = \int_0^1 \frac{dx}{\sqrt{1+x^2}}$ by trapezoidal rule dividing the interval [0, 1] into five equal parts.

Solution

$n = 5$

$\Delta x = \frac{1-0}{5} = 0.2$

x	0	0.2	0.4	0.6	0.8	1.0
$\frac{1}{\sqrt{1+x^2}}$	1.0	0.98058	0.92848	0.85749	0.78087	0.70711

From Trapezoidal Rule;

$$I = \frac{\Delta x}{2}[f(x_1) + 2(f(x_2) + f(x_3) + f(x_4) + f(x_5)) + f(x_6)]$$

$$= \frac{0.2}{2}[1 + 2(0.98058 + 0.92848 + 0.85749 + 0.78087) + 0.70711]$$

$$= 0.88016$$

Example 4.2

Use Multiple-segment Trapezoidal Rule to find the area under the curve $f(x) = \frac{300x}{1+e^x}$ from $x = 0$ to $x = 10$.

Solution

Using two segments, we get

$$\Delta x = \frac{10-0}{2} = 5$$

$$f(0) = \frac{300(0)}{1+e^0} = 0$$

$$f(5) = \frac{300(5)}{1+e^5} = 10.039$$

$$f(10) = \frac{300(10)}{1+e^{10}} = 0.136$$

$$\text{Area} = \frac{5}{2}\left[f(0) + 2f(5) + f(10)\right] = \frac{5}{2}\left[0 + 2(10.039) + 0.136\right] = 50.535$$

So what is the true value of this integral? $\int_0^{10} \frac{300x}{1+e^x} dx = 246.59$

Making the relative true error $|\in_t| = \left|\frac{246.59 - 50.535}{246.59}\right| \times 100\% = 79.506\%$

Table: Values obtained using Multiple-segment Trapezoidal Rule for $\int_0^{10} \frac{300x}{1+e^x} dx$

n	Approximate Value	E_t	$\lvert\in_t\rvert$
1	0.681	245.91	99.724%
2	50.535	196.05	79.505%
4	170.61	75.978	30.812%
8	227.04	19.546	7.927%
16	241.70	4.887	1.982%
32	245.37	1.222	0.495%
64	246.28	0.305	0.124%

Example 4.3

The average values of a function can be determined by:-

$$Cp_{mh} = \frac{\int_{T_1}^{T_2} \mathrm{C}p dT}{T_2 - T_1}$$

$C_p = 0.99403 + 1.617\times10^{-4}T + 9.7215\times10^{-8}T^2 - 9.5838\times10^{-11} T^3 + 1.9520\times10^{-14} T^4$

C_p in KJ/(Kg K)

Use this relationship to verity the average value of specific heat of dry air in the range from 300 K to 450 K:

1) Analytically

2) Numerically using five points Trapezoidal Rule

Solution

1) $$Cp_{mh} = \frac{\int_{300}^{450} 0.99403 + 1.617\times10^{-4}\mathrm{T} + 9.7215\times10^{-8}\mathrm{T}^2 - 9.5838\times10^{-11}\mathrm{T}^3 + 1.9520\times10^{-14}\mathrm{T}^4 dT}{450 - 300}$$

$$Cp_{mh} = \frac{\left. 0.99403\,\mathrm{T} + \frac{1.617\times10^{-4}}{2}\mathrm{T}^2 + \frac{9.7215\times10^{-8}}{3}\mathrm{T}^3 - \frac{9.5838\times10^{-11}}{4}\mathrm{T}^4 + \frac{1.9520\times10^{-14}}{5}\mathrm{T}^5 \right|_{300}^{450}}{450 - 300}$$

$$Cp_{mh} = \frac{465.73 - 306.18}{450 - 300} = 1.0637$$

2) $\Delta T = \frac{450 - 150}{4} = 37.5$

T	300	337.5	375	412.5	450
Cp	1.0489	1.0562	1.0637	1.0711	1.0785

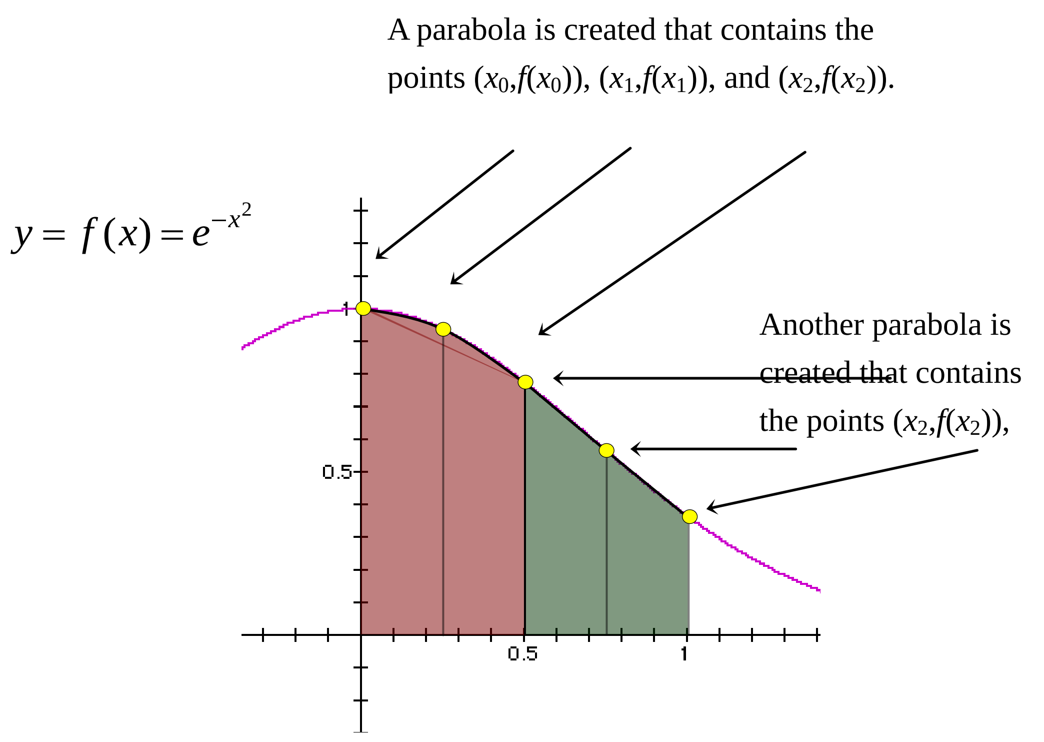
$$Cp_{mh} = \frac{(\mathrm{dT}/2) * (\mathrm{Cp}(1) + 2 * (\mathrm{Cp}(2) + \mathrm{Cp}(3) + \mathrm{Cp}(4)) + \mathrm{Cp}(5))}{T_2 - T_1}$$

$$= \frac{(37.5/2) * (1.0489 + 2 * (1.0562 + 1.0637 + 1.0711) + 1.0785)}{450 - 300} = 1.0637$$

$$\text{Realative Error \%} = \frac{\text{Analytical Solution - Numerical Solution}}{\text{Analytical Solution}}\% = \frac{1.0637 - 1.0637}{1.0637}\% = 0\%$$

4.5 Simpson's Rule (1/3 Simpson's Rule)

The final approximation technique we develop in this section is called Simpson's Rule. It is different from the first four methods because we are not creating polygons on each subinterval but rather we create a figure with a non-straight component to it. For this method, it is required that the number of subintervals be an **even** number.



Simpson's Rule uses pairs of subdivisions and creates over each pair a parabola that contains the points (x_{2i-2}, $f(x_{2i-2})$), (x_{2i-1}, $f(x_{2i-1})$), and (x_{2i}, $f(x_{2i})$) for i going from 1 to $^n/_2$. A shape is created using the resulting parabola, two vertical segments—one from (x_{2i-2},0) to (x_{2i-2}, $f(x_{2i-2})$) and one from (x_{2i+2},0) to (x_{2i+2}, $f(x_{2i+2})$)—and the segment on the x-axis with endpoints (x_{2i-2},0) and (x_{2i+2},0). The area of the resulting shape—such as of the red-shaded figure above or the green-shaded figure above-is calculated using the formula $\Delta x \cdot \frac{1}{3}\left(f(x_{2i-1}) + 4f(x_{2i}) + f(x_{2i+1})\right)$.

The sum of the areas of these shapes represents an approximation for the area under the curve and therefore is an approximation for the value of the definite integral:

$$\int_0^1 e^{-x^2}dx \approx \left(\Delta x \cdot \frac{1}{3}\left(f(x_0) + 4f(x_1) + f(x_2)\right)\right) + \left(\Delta x \cdot \frac{1}{3}\left(f(x_2) + 4f(x_3) + f(x_4)\right)\right)$$

This same sequence of steps can be generalized for the Simpson's Rule approximation of the definite integral $\int_a^b f(x)dx$ using n subdivisions:

$$\int_a^b f(x)dx \approx \left(\Delta x \cdot \frac{1}{3}\left(f(x_0) + 4f(x_1) + f(x_2)\right)\right) + \Lambda + \left(\Delta x \cdot \frac{1}{3}\left(f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)\right)\right)$$

$$\approx \Delta x \cdot \frac{1}{3}\left(f(x_0) + 4f(x_1) + f(x_2) + \Lambda + f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)\right)$$

Single Segment 1/3 Simpson's Rule

$$\int_a^b f(x)dx \approx \frac{\Delta x}{3}\left(f(x_0) + 4f(x_1) + f(x_2)\right)$$

Multiple Segment 1/3 Simpson's Rule

$$\int_a^b f(x)dx \approx \frac{\Delta x}{3}\sum_{i=1}^{n/2}\left(f(x_{2i-2}) + 4f(x_{2i-1}) + f(x_{2i})\right)$$

Example 4.4

Evaluate the integral $I = \int_0^{0.8} \frac{dx}{\sqrt{1+x^2}}$ by 1/3 Simpson's rule dividing the interval [0, 0.8] to 4 equal sub-intervals.

Solution

$n = 4$

$$\Delta x = \frac{0.8 - 0}{4} = 0.2$$

x	0	0.2	0.4	0.6	0.8
$\frac{1}{\sqrt{1+x^2}}$	1.0	0.91287	0.84515	0.79057	0.74536

From Simpson's 1/3rd Rule

$$I = \int_0^{0.8} f(x)dx = \frac{\Delta x}{3}[(f(x_0) + 4f(x_1) + f(x_2)) + (f(x_2) + 4f(x_3) + f(x_4))]$$

$$= \frac{\Delta x}{3}[f(x_0) + 4[f(x_1) + f(x_3)] + 2f(x_2) + f(x_4)]$$

$$= \frac{0.2}{3}[1.0 + 4[0.91287 + 0.79051] + 2 \times 0.84515 + 0.74536]$$
$$= 0.68329$$

4.6 Simpson's Rule (3/8 Simpson's Rule)

If we connect the points of the curve using a 3rd order Lagrange polynomial, the area under the curve can be approximated by the following formula:

$$\int_a^b f(x)\,dx \cong \frac{3\Delta x}{8}[f(x_0) + 3f(x_1) + 3f(x_2) + 2f(x_3) + 3f(x_4) + 3f(x_5)$$
$$+ 2f(x_6) + ... + 2f(x_{n-3}) + 3f(x_{n-2}) + 3f(x_{n-1}) + f(x_n)]$$

Single Segment 3/8 Simpson's Rule

$$\int_a^b f(x)dx \approx \frac{3\Delta x}{8}[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)]$$

Multiple Segment 3/8 Simpson's Rule

$$\int_a^b f(x)dx \approx \frac{3\Delta x}{8}\sum_{i=1}^{n/3}\left(f(x_{3i-3}) + 3f(x_{3i-2}) + 3f(x_{3i-1}) + f(x_{3i})\right)$$

Example 4.5

Evaluate the integral of the following tabular data with

(a) The trapezoidal rule.

(b) Simpson's rules.

x	0	0.1	0.2	0.3	0.4	0.5
F(x)	1	8	4	3.5	5	1

Solution

(a) Trapezoidal rule (n = 5):

$$I = \frac{0.1}{2}[1 + 2(8 + 4 + 3.5 + 5) + 1] = 2.15$$

(b) Simpson's rules (n = 5):

$$I = \frac{0.1}{3}[1 + 4(8) + 4] + 0.1\frac{3}{8}[4 + 3(3.5 + 5) + 1] = 1.233333 + 1.14375 = 2.377083$$

Example 4.6

The volume of is given by following expression:

$$V = \frac{F_{A0}}{CA_o}\int_0^{0.9} \frac{d\,x_A}{k(1 - x_A)}$$

with $k = 2.7 \times 10^7 \exp(-6500/T)\quad \min^{-1}$ and $\quad T = 325 + \dfrac{19000x_A}{120.35x_A + 143.75}$ using

$$F_{A0} = 1500\text{mol/min}, \quad CA_0 = 2.5 \text{ mol L}^{-1}$$

Calculate the volume of the reactor using Simpsons rule with five points (4 steps).

Solution

Xa	T	k	$\frac{1}{k(1-x_A)}$
0	325.0000	0.0557	17.9691
0.2250	350.0251	0.2325	5.5491
0.4500	368.2020	0.5816	3.1263
0.6750	382.0035	1.1005	2.7958
0.9000	392.8396	1.7597	5.6827

$$V = (1500/2.5)*(0.225/3)*(23.1031 + 4*7.1346 + 2*4.0195 + 4*3.5947 + 7.3063)$$
$$= 3661.4\text{ L}$$

A Matlab program for solving example 4.5 is listed in Table 4.1.

Table (4.1) Matlab code and results for solution example (4.5)	
Matlab Code	**Xa=0:0.225:0.9** **T=325+(19000*Xa)./(120.35*Xa+143.75)** **k=2.7e7*exp(-6500./T)** **f=1./(k.*(1-Xa))** **V=(1500/2.5)*(0.225/3)*(f(1)+4*f(2)+2*f(3)+4*f(4)+f(5))**
Results	**Xa =** **0 0.2250 0.4500 0.6750 0.9000** **T =** **325.0000 350.0251 368.2020 382.0035 392.8396** **k =** **0.0557 0.2325 0.5816 1.1005 1.7597** **f =** **17.9691 5.5491 3.1263 2.7958 5.6827** **V =** **2.8478e+03**

Numerical Differentiation

Numerical Differentiation is a method used to approximate the value of a derivative over a continuous region [a,b].

Let f(x) is a continuous function with step size h. There are forward, backward and centered difference methods to approximate the derivatives of f(x) at a point x_i.

5.1 Forward Difference Approximation of the First Derivative

We know

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

For a finite $'\Delta x'$.

$$f'(x) \cong \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

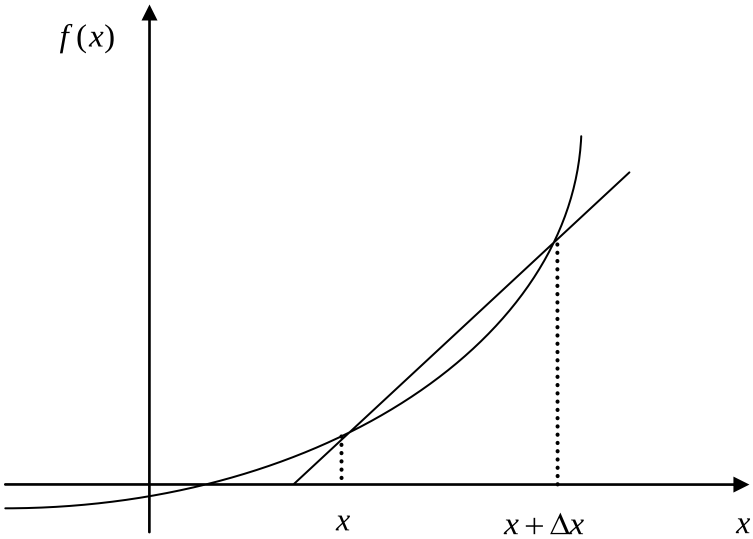


Figure 5.1: Graphical representation of forward difference approximation of first derivative

So if you want to find the value of $f'(x)$ at $x = x_i$, we may choose another point $'\Delta x'$ ahead as $x = x_{i+1}$. This gives

$$\boxed{f'(x_i) \cong \frac{f(x_{i+1}) - f(x_i)}{\Delta x}}$$

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

Example 5.1

The velocity of a rocket is given by $v(t)=2000\ln\left[\frac{14\times10^4}{14\times10^4-2100t}\right]-9.8t \quad , \quad 0\le t\le 30$

Where $'v'$ is given in m/s and $'t'$ is given in seconds.

Use forward difference approximation of the first derivative of $v(t)$ to calculate the acceleration at $t=16s$. Use a step size of $\Delta t=2s$.

Solution

$$a(t_i)\cong\frac{v(t_{i+1})-v(t_i)}{\Delta t}$$

$$t_i=16$$

$$\Delta t=2$$

$$t_{i+1}=t_i+\Delta t=16+2=18$$

$$a(16)=\frac{v(18)-v(16)}{2}$$

$$v(18)=2000\ln\left[\frac{14\times10^4}{14\times10^4-2100(18)}\right]-9.8(18)=453.02m/s$$

$$v(16)=2000\ln\left[\frac{14\times10^4}{14\times10^4-2100(16)}\right]-9.8(16)=392.07m/s$$

Hence

$$a(16)=\frac{v(18)-v(16)}{2}=\frac{453.02-392.07}{2}=30.475m/s^2$$

The exact value of $a(16)$ can be calculated by differentiating

$$a(t)=\frac{d}{dt}\left[2000\ln\left[\frac{14\times10^4}{14\times10^4-2100t}\right]-9.8t\right] \quad =2000\left(\frac{14\times10^4-2100t}{14\times10^4}\right)\frac{d}{dt}\left(\frac{14\times10^4}{14\times10^4-2100t}\right)-9.8$$

$$=2000\left(\frac{14\times10^4-2100t}{14\times10^4}\right)(-1)\left(\frac{14\times10^4}{(14\times10^4-2100t)^2}\right)(-2100)-9.8 \quad =29.674m/s^2$$

The absolute relative true error is

$$|\in_t|=\left|\frac{\text{True Value - Approximate Value}}{\text{True Value}}\right|\times100=\left|\frac{29.674-30.475}{29.674}\right|\times100=2.6993\%$$

5.2 Backward Difference Approximation of the First Derivative

We know

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

For a finite $'\Delta x'$,

$$f'(x) \cong \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

If $'\Delta x'$ is chosen as a negative number,

$$f'(x) \cong \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

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

This is a backward difference approximation as you are taking a point backward from x. To find the value of $f'(x)$ at $x = x_i$, we may choose another point $'\Delta x'$ behind as $x = x_{i-1}$. This gives

$$\boxed{f'(x_i) \cong \frac{f(x_i) - f(x_{i-1})}{\Delta x}}$$

$$= \frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}} \qquad \text{where} \qquad \Delta x = x_i - x_{i-1}$$

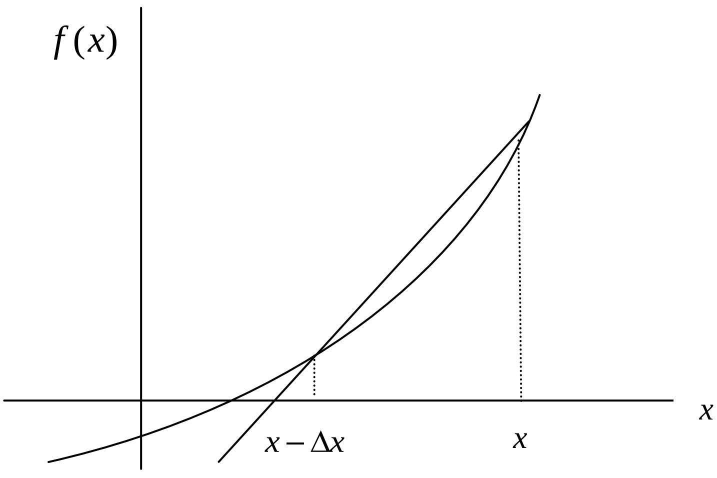


Figure 5.2 Graphical representation of backward difference approximation of first derivative

Example 5.2

The velocity of a rocket is given by

$$\nu(t) = 2000 \ln\left[\frac{14\times10^4}{14\times10^4 - 2100\,t}\right] - 9.8t \;,\; 0 \le t \le 30$$

Use backward difference approximation of the first derivative of $\nu(t)$ to calculate the acceleration at $t = 16s$. Use a step size of $\Delta t = 2s$.

Solution

$$a(t) \cong \frac{\nu(t_i) - \nu(t_{i-1})}{\Delta t}$$

$$t_i = 16$$

$$\Delta t = 2$$

$$t_{i-1} = t_i - \Delta t = 16 - 2 = 14$$

$$a(16) = \frac{\nu(16) - \nu(14)}{2}$$

$$\nu(16) = 2000\ln\left[\frac{14\times10^4}{14\times10^4 - 2100(16)}\right] - 9.8(16) = 392.07 m/s$$

$$\nu(14) = 2000\ln\left[\frac{14\times10^4}{14\times10^4 - 2100(14)}\right] - 9.8(14) = 334.24 m/s$$

$$a(16) = \frac{\nu(16) - \nu(14)}{2} = \frac{392.07 - 334.24}{2} = 28.915 m/s^2$$

The absolute relative true error is

$$|\in_t| = \left|\frac{29.674 - 28.915}{29.674}\right| \times 100 \quad = 2.557\%$$

5.3 Central Difference Approximation of the First Derivative

As shown above, both forward and backward divided difference approximation of the first derivative are accurate on the order of $0(\Delta x)$. Can we get better approximations?

Yes, another method to approximate the first derivative is called the **Central difference approximation of the first derivative.**

From Taylor series

$$f(x_{i+1}) = f(x_i) + f'(x_i)\Delta x + \frac{f''(x_i)}{2!}(\Delta x)^2 + \frac{f'''(x_i)}{3!}(\Delta x)^3 + \ldots \qquad (1)$$

$$f(x_{i-1}) = f(x_i) - f'(x_i)\Delta x + \frac{f''(x_i)}{2!}(\Delta x)^2 - \frac{f'''(x_i)}{3!}(\Delta x)^3 + \ldots \qquad (2)$$

Subtracting equation (2) from equation (1)

$$f(x_{i+1}) - f(x_{i-1}) = f'(x_i)(2\Delta x) + \frac{2f'''(x_i)}{3!}(\Delta x)^3 + \ldots$$

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2\Delta x} - \frac{f'''(x_i)}{3!}(\Delta x)^2 + \ldots$$

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2\Delta x} + 0(\Delta x)^2$$

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

Hence showing that we have obtained a more accurate formula as the error is of the order of $0(\Delta x)^2$.

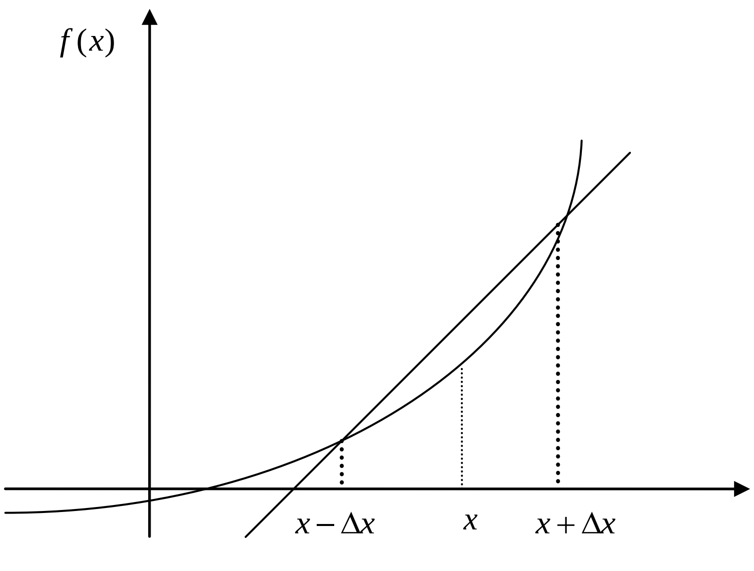


Figure 5.3 Graphical Representation of central difference approximation of first derivative.

Example 5.3

The velocity of a rocket is given by

$$v(t) = 2000\ln\left[\frac{14\times10^4}{14\times10^4 - 2100t}\right] - 9.8t, 0 \le t \le 30 .$$

Use central divided difference approximation of the first derivative of $v(t)$ to calculate the acceleration at $t = 16s$. Use a step size of $\Delta t = 2s$.

Solution

$$a(t_i) \cong \frac{v(t_{i+1}) - v(t_{i-1})}{2\Delta t}$$

$$t_i = 16$$

$$t_{i+1} = t_i + \Delta t = 16 + 2 = 18$$

$$t_{i-1} = t_i - \Delta t = 16 - 2 = 14$$

$$a(16) = \frac{v(18) - v(14)}{2(2)} = \frac{v(18) - v(14)}{4}$$

$$v(18) = 2000\ln\left[\frac{14\times10^4}{14\times10^4 - 2100(18)}\right] - 9.8(18) = 453.02 m/s$$

$$v(14) = 2000\ln\left[\frac{14\times10^4}{14\times10^4 - 2100(14)}\right] - 9.8(14) = 334.24 m/s$$

$$a(16) = \frac{v(18) - v(14)}{4} = \frac{453.02 - 334.24}{4} = 29.695 m/s^2$$

The absolute relative true error is

$$|\in_t| = \left|\frac{29.674 - 29.695}{29.674}\right| \times 100 = 0.070769\%$$

The results from the three difference approximations are given in Table 1.

Table 1 Summary of $a(16)$ using different divided difference approximations.

Type of Difference Approximation	$a(16)$ (m/s^2)	$\lvert\in_t\rvert\%$
Forward	30.475	2.6993
Backward	28.915	2.557
Central	29.695	0.070769

Clearly, the central difference scheme is giving more accurate results because the order of accuracy is proportional to the square of the step size.

5.4 Higher Order Derivatives

Example: Second order derivative:

Note that for the centered formulation, it is a derivation of a derivative:

$$f''(x) \cong \frac{\dfrac{f(x_{i+1}) - f(x_i)}{\Delta x} - \dfrac{f(x_i) - f(x_{i-1})}{\Delta x}}{\Delta x} = \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{(\Delta x)^2}$$

Forward
$$f''(x) \cong \frac{f(x_{i+2}) - 2f(x_{i+1}) + f(x_i)}{(\Delta x)^2}$$

Backward
$$f''(x) \cong \frac{f(x_i) - 2f(x_{i-1}) + f(x_{i-2})}{(\Delta x)^2}$$

Centered
$$f''(x) \cong \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{(\Delta x)^2}$$

I) Forward Difference Methods

First Derivative

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

Second Derivative

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

Third Derivative

$$f^{(3)}(x_i) = \frac{f(x_{i+3}) - 3f(x_{i+2}) + 3f(x_{i+1}) - f(x_i)}{(\Delta x)^3}$$

Fourth Derivative

$$f^{(4)}(x_i) = \frac{f(x_{i+4}) - 4f(x_{i+3}) + 6f(x_{i+2}) - 4f(x_{i+1}) + f(x_i)}{(\Delta x)^4}$$

II) Backward Difference Methods

First Derivative

$$f'(x_i) = \frac{f(x_i) - f(x_{i-1})}{\Delta x}$$

Second Derivative

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

Third Derivative

$$f^{(3)}(x_i) = \frac{f(x_i) - 3f(x_{i-1}) + 3f(x_{i-2}) - f(x_{i-3})}{(\Delta x)^3}$$

Fourth Derivative

$$f^{(4)}(x_i) = \frac{f(x_i) - 4f(x_{i-1}) + 6f(x_{i-2}) - 4f(x_{i-3}) + f(x_{i-4})}{(\Delta x)^4}$$

III) Central Difference Methods

First Derivative

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

Second Derivative

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

Third Derivative

$$f^{(3)}(x_i) = \frac{f(x_{i+2}) - 2f(x_{i+1}) + 2f(x_{i-1}) - f(x_{i-2})}{2(\Delta x)^3}$$

Fourth Derivative

$$f^{(4)}(x_i) = \frac{f(x_{i+2}) - 4f(x_{i+1}) + 6f(x_i) - 4f(x_{i-1}) + f(x_{i-2})}{(\Delta x)^4}$$

6.1 Linear Equation

$y = mx$ is an equation, in which variable y is expressed in terms of x and the constant m, is called Linear Equation. In Linear Equation exponents of the variable is always ‘one’.

6.2 Linear Equation in n variables:

$a_1x_1 + a_2x_2 + a_3x_3 + ... + a_nx_n = b$

Where $x_1, x_2, x_3, ..., x_n$ are variables and

$a_1, a_2, a_{13}, ..., a_n$ and b are constants.

6.3 System of Linear Equations:

A Linear System of m linear equations and n unknowns is:

$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + ... + a_{1n}x_n = b_1$

$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + ... + a_{2n}x_n = b_2$

$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + ... + a_{3n}x_n = b_3$

....

....

$a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + ... + a_{mn}x_n = b_m$

Where $x_1, x_2, x_3, ..., x_n$ are variables or unknowns and a’s and b’s are constants.

6.4 Augmented Matrix

System of linear equations:

$a_{11}x_1 + a_{12}x_2 + a_{13}x_{31} = b_1$

$a_{21}x_1 + a_{22}x_2 + a_{23}x_{31} = b_2$

$a_{31}x_1 + a_{32}x_2 + a_{33}x_{31} = b_3$

Can be written in the form of matrices product

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Or we may write it in the form AX=b,

Where A= $\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$, X = $\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$, b = $\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$

Augmented matrix is $[A:b] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix}$

Example 6.1:

Write the matrix and augmented form of the system of linear equations

3x – y + 6z = 6

x + y + z = 2

2x + y+4z = 3

Solution: Matrix form of the system is

$$\begin{bmatrix} 3 & -1 & 6 \\ 1 & 1 & 1 \\ 2 & 1 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 6 \\ 2 \\ 3 \end{bmatrix}$$

Augmented form is $[A:b] = \begin{bmatrix} 3 & -1 & 6 & 6 \\ 1 & 1 & 1 & 2 \\ 2 & 1 & 4 & 3 \end{bmatrix}$.

6.5 Methods for Solving System of Linear Equations

1. **Gaussian Elimination Method**
2. **Gauss -Jorden Elimination Method**

6.5.1 Gaussian Elimination.

Gaussian elimination is a general method of finding possible solutions to a linear system of equations.

Gaussian Elimination Method

Step 1. By using elementary row operations

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & A_{12} & A_{13} & B_1 \\ 0 & 1 & A_{23} & B_2 \\ 0 & 0 & 1 & B_3 \end{bmatrix}$$

Step 2. Find solution by back – substitutions.

Example 6.2:

Solve the system of linear equations by Gaussion- Elimination method

$$\begin{cases} x_1 + x_2 + x_3 = 3 \\ 2x_1 - x_2 - 2x_3 = 6 \\ 4x_1 + 2x_2 + 3x_3 = 7 \end{cases}$$

Solution:

Step 1.

Augmented matrix is $\left[\begin{array}{ccc\|c} 1 & 1 & 1 & 3 \\ 2 & -1 & -2 & 6 \\ 4 & 2 & 3 & 7 \end{array}\right]$	$R_2 = r_2 - 2r_1$ $R_3 = r_3 - 4r_1$
$\left[\begin{array}{ccc\|c} 1 & 1 & 1 & 3 \\ 0 & -3 & -4 & 0 \\ 0 & -2 & -1 & -5 \end{array}\right]$	$R_3 = 3r_3 - 2r_2$
$\left[\begin{array}{ccc\|c} 1 & 1 & 1 & 3 \\ 0 & -3 & -4 & 0 \\ 0 & 0 & 5 & -15 \end{array}\right]$	$R_2 = -r_2$ $R_3 = \frac{r_3}{5}$
$\left[\begin{array}{ccc\|c} 1 & 1 & 1 & 3 \\ 0 & 3 & 4 & 0 \\ 0 & 0 & 1 & -3 \end{array}\right]$	

Equivalent system of equations form is:

Step 2. Back Substitution

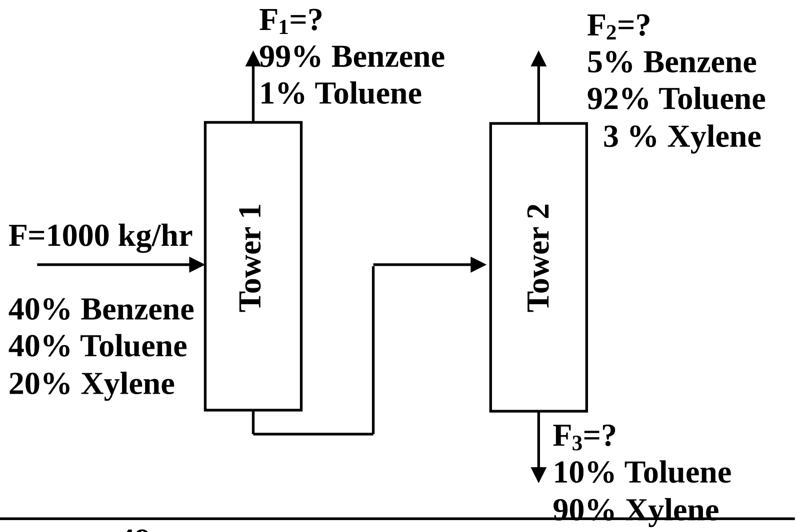
$$\begin{aligned} x_1 + x_2 + x_3 &= 3 \\ 3x_2 + 4x_3 &= 0 \\ x_3 &= -3 \end{aligned} \quad \Rightarrow \quad \begin{aligned} x_3 &= -3 \\ x_2 &= -4x_3/3 = 12/3 = 4 \\ x_1 &= 3 - x_2 - x_3 = 3 - 4 + 3 = 2 \end{aligned}$$

Solutions are $x_1 = 2, \; x_2 = 4, \; x_3 = -3$

Example 6.3:

For the below figure calculate the values of the unknown flow rates F_1, F_2 and F_3 by using Gaussion- Elimination method

.



Component material balance gives these three equations of three variables

$$F_1 + F_2 + F_3 = 1000$$
$$0.99F_1 + 0.05F_2 + 0F_3 = 400$$
$$0.01F_1 + 0.92F_2 + 0.1F_3 = 400$$

Augmented matrix is

$$\begin{bmatrix} 1 & 1 & 1 & 1000 \\ 0.99 & 0.05 & 0 & 400 \\ 0.01 & 0.92 & 0.1 & 400 \end{bmatrix} \qquad \begin{matrix} R_2 = r_2 - (0.99/1)\times r_1 \\ R_3 = r_3 - (0.01/1)\times r_1 \end{matrix}$$

$$\begin{bmatrix} 1 & 1 & 1 & 1000 \\ 0 & -0.94 & -0.99 & -590 \\ 0 & 0.91 & 0.09 & 390 \end{bmatrix} \qquad R_3 = r_3 - (0.91/(-0.94))r_2$$

$$\begin{bmatrix} 1 & 1 & 1 & 1000 \\ 0 & -0.94 & -0.99 & -590 \\ 0 & 0 & -0.8684 & -181.17 \end{bmatrix} \qquad \begin{matrix} R_2 = r_2/(-0.94) \\ R_3 = r_3/(-0.8684) \end{matrix}$$

$$\begin{bmatrix} 1 & 1 & 1 & 1000 \\ 0 & 1 & 1.0532 & 627.6596 \\ 0 & 0 & 1 & 208.6253 \end{bmatrix}$$

Equivalent system of equations form is:

$F_1 + F_2 + F_3 = 1000$

$F_2 + 1.0532F_3 = 627.6596$

$F_3 = 208.6253$

Step 2. Back Substitution

$F_3 = 208.6253$

$F_2 = 627.6596 - 1.0532F_3 = 627.6596 - 1.0532\times 208.6253 = 407.9354$

$F_1 = 1000 - F_2 - F_3 = 1000 - 208.6253 - 407.9354 = 383.4393$

6.5.2 Gauss - Jorden Elimination Method

Gauss - Jorden Method
By using elementary row operations

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & 0 & B_1 \\ 0 & 1 & 0 & B_2 \\ 0 & 0 & 1 & B_3 \end{bmatrix}$$

Example 6.4:
Solve the system of linear equations by Gauss-Jorden elimination method
$x_1 + x_2 + 2x_3 = 8$
$- x_1 - 2x_2 + 3x_3 = 1$
$3x_1 -7x_2 + 4x_3 = 10$

Solution:
Augmented matrix is

$$\begin{bmatrix} 1 & 1 & 2 & 8 \\ -1 & -2 & 3 & 1 \\ 3 & -7 & 4 & 10 \end{bmatrix} \quad R_2= r_2+r_1 \quad R_3= r_3-3r_1$$

$$\begin{bmatrix} 1 & 1 & 2 & 8 \\ 0 & -1 & 5 & 9 \\ 0 & -10 & -2 & -14 \end{bmatrix} \quad R_2=-r_2 \quad R_3=r_3-10r_2$$

$$\begin{bmatrix} 1 & 1 & 2 & 8 \\ 0 & 1 & -5 & -9 \\ 0 & 0 & -52 & -104 \end{bmatrix} \quad R_3=-r_3/52$$

$$\begin{bmatrix} 1 & 1 & 2 & 8 \\ 0 & 1 & -5 & -9 \\ 0 & 0 & 1 & 2 \end{bmatrix} \quad R_1=r_1-2r_3 \quad R_2=r_2+5r_3$$

$$\begin{bmatrix} 1 & 1 & 0 & 4 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix} \quad R_1=r_1-r_2$$

$$\begin{bmatrix} 1 & 0 & 0 & 3 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}$$

Equivalent system of equations form is:
$x_1 = 3$

$x_2 = 1$
$x_3 = 2$
is the solution of the system.

Example 6.5:
Total and component material balance on a system of distillation columns gives the flowing equations:-

$$F_1 + F_2 + F_3 + F_4 = 1690$$
$$0.4F_1 + 0.15F_2 + 0.25F_3 + 0.2F_4 = 412.5$$
$$0.25F_1 + 0.8F_2 + 0.3F_3 + 0.45F_4 = 701$$
$$0.08F_1 + 0.05F_2 + 0.45F_3 + 0.3F_4 = 487.3$$

Use Gauss - Jorden method to compute the four un-known's in above equations:-

Solution:
Augmented matrix is

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1690 \\ 0.4 & 0.15 & 0.25 & 0.2 & 412.5 \\ 0.25 & 0.8 & 0.3 & 0.45 & 701 \\ 0.08 & 0.05 & 0.45 & 0.3 & 487.3 \end{bmatrix}$$

$R_2 = r_2-(0.4/1)r_1$
$R_3 = r_3-(0.25/1)r_1$
$R_4 = r_4-(0.08/1)r_1$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1690 \\ 0 & -0.25 & -0.15 & -0.2 & -263.5 \\ 0 & 0.55 & 0.05 & 0.2 & 278.5 \\ 0 & -0.03 & 0.37 & 0.22 & 352.1 \end{bmatrix}$$

$R_3 = r_3-(0.55/(-0.25))r_2$
$R_4 = r_4-((-0.03)/(-0.025))r_2$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1690 \\ 0 & -0.25 & -0.15 & -0.2 & -263.5 \\ 0 & 0 & -0.28 & -0.24 & -301.2 \\ 0 & 0 & 0.388 & 0.244 & 383.72 \end{bmatrix}$$

$R_4 = r_4-((0.0388)/(-0.028))r_3$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1690 \\ 0 & -0.25 & -0.15 & -0.2 & -263.5 \\ 0 & 0 & -0.28 & -0.24 & -301.2 \\ 0 & 0 & 0 & -0.08857 & -33.657 \end{bmatrix}$$

$R_2 = r_2/(-0.25)$
$R_3 = r_3/(-0.028)$
$R_4 = r_4/(-0.0887)$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1690 \\ 0 & 1 & 0.6 & 0.8 & 1054 \\ 0 & 0 & 1 & 0.85714 & 1075.74 \\ 0 & 0 & 0 & 1 & 380 \end{bmatrix}$$

$R_1= r_1-r_4$
$R_2=r_2-(0.8/1)r_4$
$R_3=r_3-(0.85714/1)r_4$

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 1310 \\ 0 & 1 & 0.6 & 0 & 750 \\ 0 & 0 & 1 & 0 & 750 \\ 0 & 0 & 0 & 1 & 380 \end{bmatrix}$$

$R_1= r_1-r_3$
$R_2=r_2-((-0.6)/1)r_3$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 560 \\ 0 & 1 & 0 & 0 & 300 \\ 0 & 0 & 1 & 0 & 750 \\ 0 & 0 & 0 & 1 & 380 \end{bmatrix}$$

$R_1=r_1-r_2$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 260 \\ 0 & 1 & 0 & 0 & 300 \\ 0 & 0 & 1 & 0 & 750 \\ 0 & 0 & 0 & 1 & 380 \end{bmatrix}$$

Equivalent system of equations form:
$F_1 = 260$, $F_2 = 300$, $F_3 = 750$ and
$F_4 = 380$ is the solution of the system.

Example 6.6

Balance the following chemical equation:

$x_1\ P_2I_4 + x_2\ P_4 + x_3\ H_2O \rightarrow x_4\ PH_4I + x_5\ H_3PO_4$

Solution:

P balance: $2x_1+ 4x_2=x_4+ x_5$

I balance: $4x_1=x_4+ x_5$

H balance: $2x_3=4x_4+3x_5$

O balance: $x_3=4x_5$

Re-write these as homogeneous equations, each having zero on its right hand side:

$2x_1+ 4x_2- x_4- x_5= 0$

$4x_1- x_4 - x=0$

$2x_3- 4x_4- 3x_5=0$

$x_3- 4x_5= 0$

At this point, there are four equations in five unknowns. To complete the system, we define an auxiliary equation by arbitrarily choosing a value for one of the coefficients:

$x_1= 1$

We can easily solve the above equations to balance this reaction using MATLAB such in table 6.1

Table (6.1) Matlab code and results for solution example (6.6)	
Matlab Code	**A = [2 4 0 -1 -1** **4 0 0 -1 0** **0 0 2 -4 -3** **0 0 1 0 -4** **1 0 0 0 0];** **B= [0;0;0;0;1];** **X = A\B**
Results	**X =** **1.0000** **1.3000** **12.8000** **4.0000** **3.2000**

This does not yield integral coefficients, but multiplying by 10 will do the trick:

The balanced equation will be:

$10\ P_2I_4 + 13\ P_4 + 128\ H_2O \rightarrow 40\ PH_4I + 32\ H_3PO_4$

7.1 Jacobi Method

Let the given equation be

$$a_1x + b_1y + c_1z = d_1$$

$$a_2x + b_2y + c_2z = d_2$$

$$a_3x + b_3y + c_3z = d_3$$

If the given system of equation is diagonally dominant then

$$x^{(i+1)} = \frac{1}{a_1}\left(d_1 - b_1y^{(i)} - c_1z^{(i)}\right)$$

$$y^{(i+1)} = \frac{1}{b_2}\left(d_2 - a_2x^{(i)} - c_2z^{(i)}\right)$$

$$z^{(i+1)} = \frac{1}{c_3}\left(d_3 - a_3x^{(i)} - b_3y^{(i)}\right)$$

7.1.1 Condition for Jacobi method of converges:

The sufficient condition is

$$|a_1| \geq |b_1| + |c_1|$$

$$|b_2| \geq |a_2| + |c_2|$$

$$|c_3| \geq |a_3| + |b_3|$$

The absolute value of the diagonal element in each row of the coefficient matrix must be greater than the sum of the absolute values of the off-diagonal elements in the same row.

Example 7.1:

Use the Jacobi iteration method to obtain the solution of the following equations:

$6x_1 -2 x_2 + x_3 = 11$

$x_1 +2x_2 -5x_3 = -1$

$-2x_1 +7 x_2 +2x_3 = 5$

Solution

Step 1: Re-write the equations such that each equation has the unknown with largest coefficient on the left hand side:

$6x_1= 11+2 x_2-x_3$

$7x_2= 5+2x_1 -2x_3$

$5x_3 =1+x_1 +2x_2$

$$x_1 = \frac{2x_2 - x_3 + 11}{6}$$

$$x_2 = \frac{2x_1 - 2x_3 + 5}{7}$$
$$x_3 = \frac{x_1 + 2x_2 + 1}{5}$$

Step 2: Assume the initial guesses $x_1^0 = x_2^0 = x_3^0 = 0$ then calculate x_1^1, x_2^1 *and* x_3^1:

$$x_1^1 = \frac{2(x_2^0) - (x_3^0) + 11}{6} = \frac{2(0) - (0) + 11}{6} = 1.833$$
$$x_2^1 = \frac{2(x_1^0) - 2(x_3^0) + 5}{7} = \frac{2(0) - 2(0) + 5}{7} = 0.714$$
$$x_3^1 = \frac{(x_1^0) + 2(x_2^0) + 1}{5} = \frac{(0) + 2(0) + 1}{5} = 0.200$$

Step 3: Use the values obtained in the first iteration, to calculate the values for the 2nd iteration:

$$x_1^2 = \frac{2(x_2^1) - (x_3^1) + 11}{6} = \frac{2(0.714) - (0.200) + 11}{6} = 2.038$$
$$x_2^2 = \frac{2(x_1^1) - 2(x_3^1) + 5}{7} = \frac{2(1.833) - 2(0.200) + 5}{7} = 1.181$$
$$x_3^2 = \frac{(x_1^1) + 2(x_2^1) + 1}{5} = \frac{(1.833) + 2(0.714) + 1}{5} = 0.852$$

and so on for the next iterations so that the next values are calculated using the current values:

$$x_1^{i+1} = \frac{2(x_2^i) - (x_3^i) + 11}{6}$$
$$x_2^{i+1} = \frac{2(x_1^i) - 2(x_3^i) + 5}{7}$$
$$x_3^{i+1} = \frac{(x_1^i) + 2(x_2^i) + 1}{5}$$

The results for 9 iterations are:

	Unknowns		
Iter.	x_1	x_2	x_3
1	1.833	0.714	0.200
2	2.038	1.181	0.852
3	2.085	1.053	1.080
4	2.004	1.001	1.038
.	.	.	.
.	.	.	.
9	2.000	1.000	1.000

Example 7.2:

Solve the equations by Jacobi method

$$20x_1 + x_2 - 2x_3 = 17$$
$$3x_1 + 20x_2 - x_3 = -18$$
$$2x_1 - 3x_2 + 20x_3 = 25$$

Solution

Rewrite the given equation in the form:

$$x_1^{i+1} = \frac{1}{20}(17 - x_2^i + 2x_3^i)$$

$$x_2^{i+1} = \frac{1}{20}(-18 - 3x_1^i + x_3^i)$$

$$x_3^{i+1} = \frac{1}{20}(25 - 2x_1^i + 3x_2^i)$$

Using $x_1^0 = x_2^0 = x_3^0 = 0$, we obtain

$$x_1^1 = \frac{17}{20} = 0.85$$

$$x_2^1 = \frac{-18}{20} = -0.90$$

$$x_3^1 = \frac{25}{20} = 1.25$$

Putting these values on the right of equations to obtain

$$x_1^2 = \frac{1}{20}\left(17 - x_2^1 - 2x_3^1\right) = 1.02$$

$$x_2^2 = \frac{1}{20}\left(-18 - 3x_1^1 + x_3^1\right) = -0.965$$

$$x_3^2 = \frac{1}{20}\left(25 - 2x_1^1 + 3x_2^1\right) = 1.1515$$

These and further iterates are listed in the table below:

i	x_1^i	x_2^i	x_3^i
0	0	0	0
1	0.85	–0.90	1.25
2	1.02	–0.965	1.1515
3	1.0134	–0.9954	1.0032
4	1.0009	–1.0018	0.9993
5	1.0000	–1.0002	0.9996
6	1.0000	–1.0000	1.0000

The values in 5th and 6th iterations being practically the same, we can stop. Hence the solutions are:

$$x_1 = 1,\ x_2 = -1 \text{ and } x_3 = 1$$

7.2 Gauss-Seidel Method

If the given system of equation is diagonally dominant then

$$x^{(i+1)} = \frac{1}{a_1}\left(d_1 - b_1 y^{(i)} - c_1 z^{(i)}\right)$$

$$y^{(i+1)} = \frac{1}{b_2}\left(d_2 - a_2 x^{(i+1)} - c_2 z^{(i)}\right)$$

$$z^{(i+1)} = \frac{1}{c_3}\left(d_3 - a_3 x^{(i+1)} - b_3 y^{(i+1)}\right)$$

Example 7.3:

Use the Gauss-Seidel method to obtain the solution of the following equations:

$6x_1 - 2x_2 + x_3 = 11$ (1)

$x_1 + 2x_2 - 5x_3 = -1$ (2)

$-2x_1 + 7x_2 + 2x_3 = 5$ (3)

Solution

Step 1: Re-write the equations such that each equation has the unknown with largest `coefficient on the left hand side:

$x_1 = \dfrac{2x_2 - x_3 + 11}{6}$ from eq. (1)

$x_2 = \dfrac{2x_1 - 2x_3 + 5}{7}$ from eq. (3)

$x_3 = \dfrac{x_1 + 2x_2 + 1}{5}$ from eq. (2)

Step 2: Assume the initial guesses $x_2^0 = x_3^0 = 0$, then calculate x_1^1:

$$x_1^1 = \frac{2(x_2^0) - (x_3^0) + 11}{6} = \frac{2(0) - (0) + 11}{6} = 1.833$$

Use the updated value $x_1^1 = 1.833$ and $x_3^0 = 0$ to calculate x_2^1

$$x_2^1 = \frac{2(x_1^1) - 2(x_3^0) + 5}{7} = \frac{2(1.833) - 2(0) + 5}{7} = 1.238$$

Similarly, use $x_1^1 = 1.833$ and $x_2^1 = 1.238$ to calculate x_3^1

$$x_3^1 = \frac{(x_1^1) + 2(x_2^1) + 1}{5} = \frac{(1.833) + 2(1.238) + 1}{5} = 1.062$$

Step 3: Repeat the same procedure for the 2^{nd} iteration

$$x_1^2 = \frac{2(x_2^1) - (x_3^1) + 11}{6} = \frac{2(1.238) - (1.062) + 11}{6} = 2.069$$

$$x_2^2 = \frac{2(x_1^2) - 2(x_3^1) + 5}{7} = \frac{2(2.069) - 2(1.062) + 5}{7} = 1.002$$

$$x_3^2 = \frac{(x_1^2)+2(x_2^2)+1}{5} = \frac{(2.069)+2(1.002)+1}{5} = 1.015$$

and so on for the next iterations so that the next values are calculated using the current values:

$$x_1^{i+1} = \frac{2(x_2^i)-(x_3^i)+11}{6}$$

$$x_2^{i+1} = \frac{2(x_1^{i+1})-2(x_3^i)+5}{7}$$

$$x_3^{i+1} = \frac{(x_1^{i+1})+2(x_2^{i+1})+1}{5}$$

and continue the above iterative procedure until $[(x_k)^{i+1} - (x_k)^i]/(x_k)^{i+1} < \epsilon$ for i=1,2 and 3.

The procedure yields the exact solution after 5 iterations only:

	Unknown		
Iter.	x_1	x_2	x_3
1	1.833	1.238	1.062
2	2.069	1.002	1.015
3	1.998	0.995	0.998
4	1.999	1.000	1.000
5	2.000	1.000	1.000

Example 7.4:

Solve by Gauss – Seidel method, the equations:

$$20x_1 + x_2 - 2x_3 = 17$$
$$3x_1 + 20x_2 - x_3 = -18$$
$$2x_1 - 3x_2 + 20x_3 = 25$$

Solution

As before, we start with initial estimate $x_1^0 = x_2^0 = x_3^0 =$ 0. We write the given equation in the form

$$x_1^{i+1} = \frac{1}{20}(17 - x_2^i + 2x_3^i)$$

$$x_2^{i+1} = \frac{1}{20}(-18 - 3x_1^{i+1} + x_3^i)$$

$$x_3^{i+1} = \frac{1}{20}(25 - 2x_1^{i+1} + 3x_2^{i+1})$$

These and further iterates are listed in the table below:

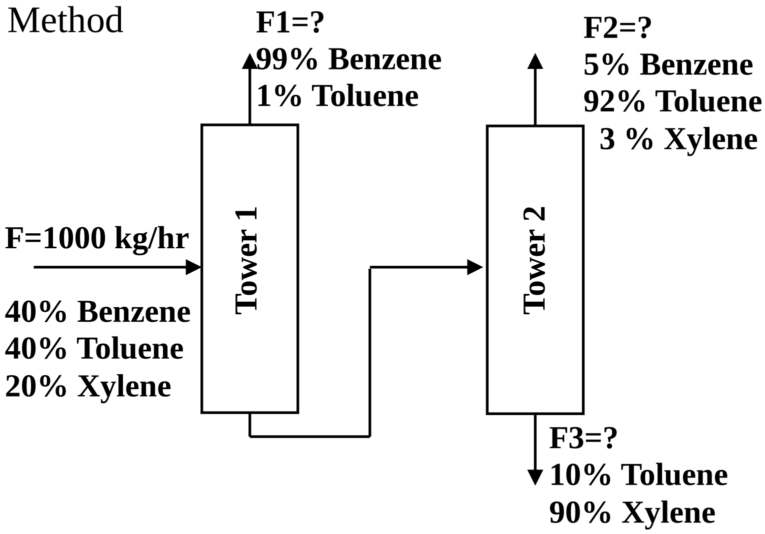
i	x_1^i	x_2^i	x_3^i
0	0	0	0
1	0.8500	–1.0275	1.0109
2	1.0025	–0.9998	0.9998
3	1.0000	–1.0000	1.0000

The value in the 2nd and 3rd iterations being particularly the same, we can stop. Hence the solutions is $x_1 = 1$, $x_2 = -1$ and $x_3 = 1$.

Example 7.5:

For the below figure calculate the values of the unknown flow rates F1, F2 and F3 by using Gauss-Seidel Method

.



Component material balance gives these three equations of three variables

$$0.99F_1 + 0.05F_2 + 0F_3 = 400$$
$$0.01F_1 + 0.92F_2 + 0.1F_3 = 400$$
$$0F_1 + 0.03F_2 + 0.9F_3 = 200$$

Re-arranging the above equations

$$F_1 = (400 - 0.05F_2)/0.99$$
$$F_2 = (400 - 0.01F_1 - 0.1F_3)/0.92$$
$$F_3 = (200 - 0.03F_2)/0.9$$

Starting with F1=F2=F3=1000/3

Iteration	F1	F2	F3
	333.3333	333.3333	333.3333
1.0000	387.2054	394.3420	209.0775
2.0000	384.1241	407.8815	208.6262
3.0000	383.4403	407.9380	208.6243
4.0000	383.4375	407.9383	208.6243
5.0000	383.4375	407.9383	208.6243

A Matlab program for solving the above equations using Gauss-Seidel method is listed in Table 7.1

Table (7.1) Matlab code and results for solution example (7.5)	
Matlab Code	**F1=333.33; F2=333.33; F3=333.33** **for i=1:4** **F1=(400-0.05*F2)/0.99;** **F2=(400-0.01*F1-0.1*F3)/0.92;** **F3=(200-0.03*F2)/0.9;** **disp([i, F1, F2, F3])** **end**
Results	**1.0000 387.2056 394.3423 209.0775** **2.0000 384.1241 407.8815 208.6262** **3.0000 383.4403 407.9380 208.6243** **4.0000 383.4375 407.9383 208.6243**

8.1 Graphical Methods

A simple method for obtaining a root of the equation $f(x) = 0$ is to plot the function and observe where it crosses the x axis. There is much available software that will facilitate making a plot of a function. We will use Matlab exclusively for the course notes; however you can use other software such as Excel or Matcad for your work.

Example 8.1

Solve $\frac{600}{x}(1 - e^{-0.15x}) = 50$ using the graphical method.

Solution

The function $f(x) = \frac{600}{x}(1 - e^{-0.15x}) - 50$ can be plotted in Figure 8.1 using the Matlab statements listed in table (8.1).

Table (8.1) Matlab code for solving example (8.1) using graphical method	
Matlab Code	**x=4:0.1:20;** **fx=600*(1-exp(-0.15*x))./x-50;** **plot(x,fx,[0 20],[0 0])** **xlabel('x');** **ylabel('f(x)')** **grid on; zoom on**

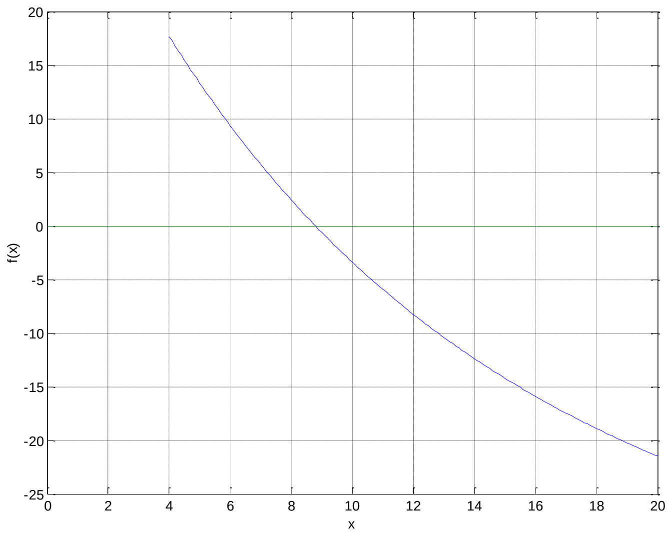


Figure 8.1 The graphical method for roots finding.

The Matlab Zoom on statement allows the function to be zoomed in at the cursor with left mouse click (right mouse click will zoom out). Each time you click the axes limits will be changed by a factor of 2 (in or out). You can zoom in as many times as necessary for the desired accuracy. Figure 8.2 shows the approximate root *x* to be 8.79

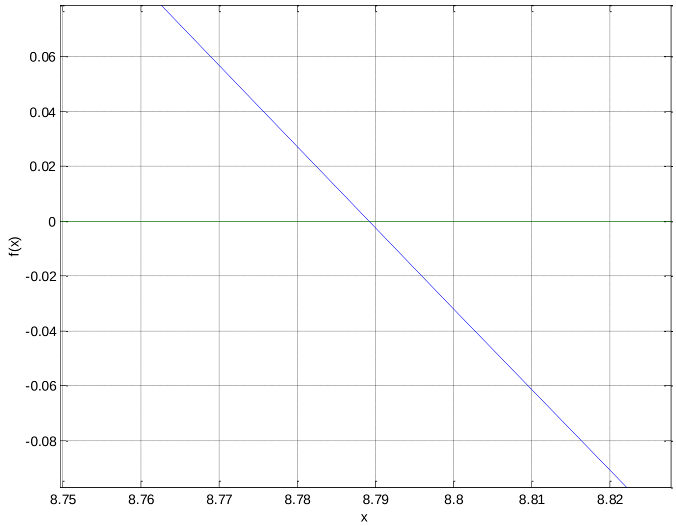


Figure 8.2 The graphical method for roots finding with Matlab Zoom on.

The plot of a function between x_1 and x_2 is important for understanding its behavior within this interval. More than one root can occur within the interval when $f(x_1)$ and $f(x_2)$ are on opposite sides of the *x* axis. The roots can also occur within the interval when $f(x_1)$ and $f(x_2)$ are on the same sides of the *x* axis. Since the functions that are tangent to the *x* axis satisfy the requirement $f(x) = 0$ at this point, the tangent point is called a *multiple root*.

8.2 The Bisection Method

The bisection method or interval halving can be used to determine the solution to f(x) = 0 on an interval [x_1 = a, x_2 = b] if f(x) is real and continuous on the interval and $f(x_1)$ and $f(x_2)$ have opposite signs. We assume for simplicity that the root in this interval is unique. The location of the root is then calculated as lying at the midpoint of the subinterval within which the functions have opposite signs. The process is repeated to any specified accuracy.

The procedure can be summarized in the following steps

Let $f(x_1)\, f(x_2) < 0$ on an interval $[x_1 = a, x_2 = b]$

Step 1 Let $x_x = \frac{1}{2}(x_1 + x_2)$; $f_1 = f(x_1)$; $f_2 = f(x_2)$

Step 2 Evaluate $f_x = f(x_x)$

If $f_x\, f_1 > 0$ then

$x_1 = x_x$; $f_1 = f_x$

else

$x_2 = x_x$; $f_2 = f_x$

end

Step 3 If abs$(x_2 - x_1)$ > an error tolerance, go back to **Step 1**

Figure 8.3 shows first three iterations x_3, x_4, and x_5 of the bisection method.

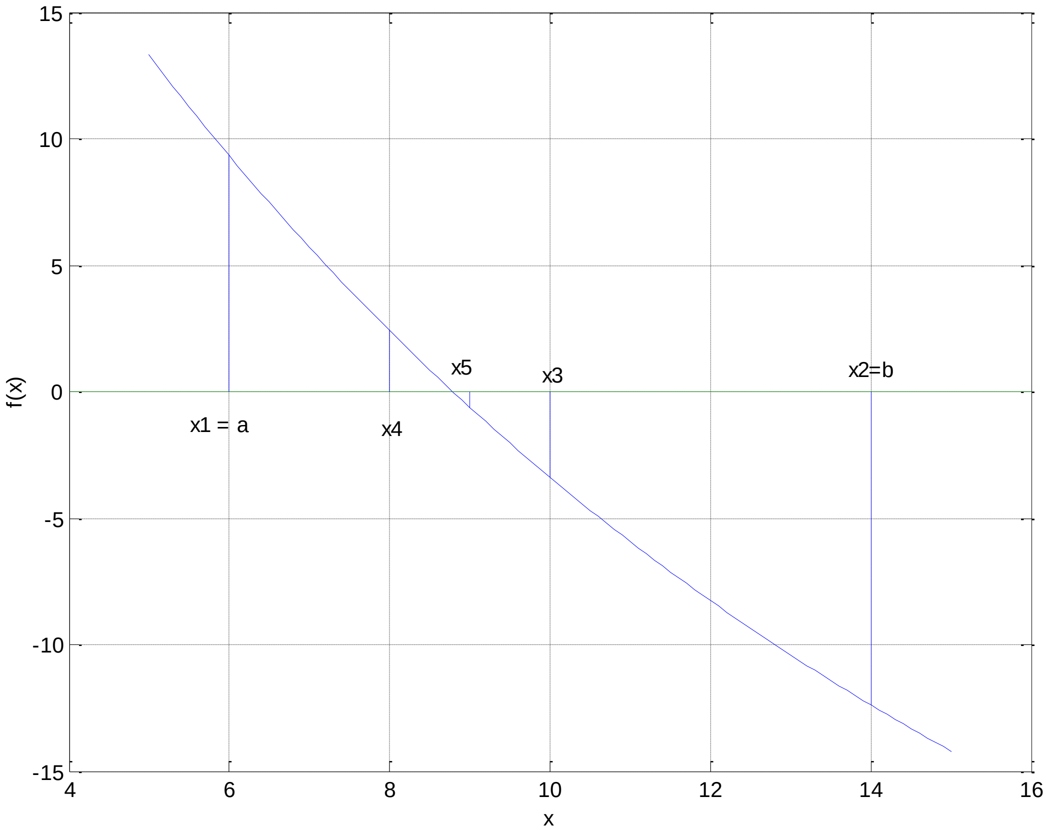


Figure 8.3 The first three iterations x_3, x_4, and x_5 of the bisection method.

$x_1 = 6 \Rightarrow f(x_1) = 9.3430$ and $x_2 = 14 \Rightarrow f(x_2) = -12.3910$

$x_3 = \frac{1}{2}(x_1 + x_2) = \frac{1}{2}(6 + 14) = 10 \Rightarrow f(x_3) = -3.3878$

$f(x_1)\, f(x_3) < 0 \Rightarrow x_4 = \frac{1}{2}(x_1 + x_3) = \frac{1}{2}(6 + 10) = 8 \Rightarrow f(x_4) = 2.4104$

$f(x_3)\, f(x_4) < 0 \Rightarrow x_5 = \frac{1}{2}(x_3 + x_4) = \frac{1}{2}(10 + 8) = 9 \Rightarrow f(x_5) = -0.6160$

Since $f(x_1)$ and $f(x_2)$ bracket the root and $x_3 = \frac{1}{2}(x_1 + x_2) = \frac{1}{2}(a + b)$, the error after the first iteration is less than or equal to $\frac{1}{2}(b - a)$.

A Matlab program for solving example 8.1 using bisection method is listed in Table 8.2 where the function $f(x)$ is an input to the program.

<table>
<tr><td colspan="2">Table (8.2) Matlab code and results for solving example (8.1) using bisection method</td></tr>
<tr><td>Matlab Code</td><td><pre>fun=inline('600*(1-exp(-0.15*x)) /x-50')
x1=6;
f1= fun (x1);
x2=14;
f2= fun (x2);
tol=1e-5;
for i=1:100
    x3=(x1+x2)/2;
    f3= fun(x3);
    if f1*f3<0
    x2=x3;
    f2=f3;
    else
    x1=x3;
    f1=f3;
    end
    if abs(x2-x1)<tol; break;end
end
x3</pre></td></tr>
<tr><td>Results</td><td><pre>x3 =
    8.7892</pre></td></tr>
</table>

The statement **Inline** is used to define the function at a given value of x.

Example 8.2

Use the bisection method to find the root of the equation x-cos(x) = 0 with a percent relative error $|\varepsilon_t| \leq 1\%$. (The exact value is 0.7391)

Solution

f(x)= x-cos(x)

We have seen before that there is a single root lies in the interval [0,1]. Therefore, we start with x_l=0 and x_2=1, then iterate using the same procedure followed in example 8.1 to get the following tabulated results:

Iter	X_l	X_2	X_3	$f(X_l)$	$f(X_2)$	$f(X_3)$	$\lvert\varepsilon_t\rvert$
1	0.0	1.0	0.5	-1.0000	0.4597	-0.3776	32.35
2	0.5	1.0	0.75	-0.3776	0.4597	0.0183	1.48
3	0.5	0.75	0.625	-0.3776	0.0183	-0.1860	15.44
4	0.625	0.75	0.6875	-0.1860	0.0183	-0.0853	6.98
5	0.6875	0.75	0.7188	-0.0853	0.0183	-0.0339	2.75
6	0.7188	0.75	0.7344	-0.0339	0.0183	-0.0079	0.64

Then x=0.7344

Example 8.3

The friction factor *f* depends on the Reynolds number Re for turbulent flow in smooth pipe according to the following relationship.

$$\frac{1}{\sqrt{f}} = -0.40 + \sqrt{3}\ln(\mathrm{Re}\sqrt{f})$$

Use the bisection method to compute f for Re = 25200 that lies between [0.001 ,0.1].

Solution

Re-write the above equation in the form

$$E(f) = -0.40 + \sqrt{3}\ln(25200\sqrt{f}) - \frac{1}{\sqrt{f}}$$

Iter	f_l	f_2	f_3	$E(f_l)$	$E(f_2)$	$E(f_3)$
1	0.0010	0.1000	0.0505	-20.4514	11.9973	10.1179
2	0.0010	0.0505	0.0258	-20.4514	10.1179	7.7528
3	0.0010	0.0258	0.0134	-20.4514	7.7528	4.7705
4	0.0010	0.0134	0.0072	-20.4514	4.7705	1.0841
5	0.0010	0.0072	0.0041	-20.4514	1.0841	-3.2373

6	0.0041	0.0072	0.0056	-3.2373	1.0841	-0.6453
7	0.0056	0.0072	0.0064	-0.6453	1.0841	0.2946
8	0.0056	0.0064	0.0060	-0.6453	0.2946	-0.1536

Then according to above table $f = 0.006$

8.3 Secant Method (Linear Interpolation Method)

The bisection method is generally inefficient, it requires more function evaluations in comparison with the secant method which is linear interpolation using the latest two points. Figure 8.4 shows graphically the root x_3 obtained from the intersection of the line $A\overline{B}$ with the x-axis.

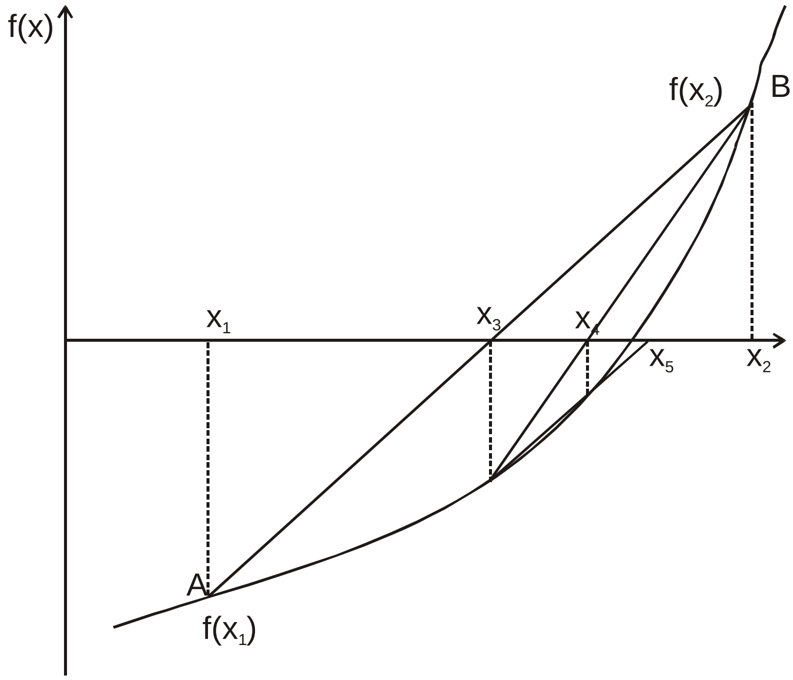


Figure 8.4 Graphical depiction of the secant method.

The intersection of the straight line with the x-axis can be obtained by using similar triangles $x_3 x_1 A$ and $x_3 x_2 A$ or by using linear interpolation with the following points.

x	x_1	x_3	x_2
$f(x)$	$f(x_1)$	0	$f(x_2)$

$$\frac{x_3 - x_2}{x_2 - x_1} = \frac{0 - f(x_2)}{f(x_2) - f(x_1)} \Rightarrow x_3 = x_2 - f(x_2)\frac{x_2 - x_1}{f(x_2) - f(x_1)}$$

The next guess is then obtained from the straight line through two points $[x_2, f(x_2)]$ and $[x_3, f(x_3)]$. In general, the guessed valued is calculated from the two previous points $[x_{n-1}, f(x_{n-1})]$ and $[x_n, f(x_n)]$ as

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

The secant method always uses the latest two points without the requirement that they bracket the root as shown in Figure 8.4 for points [x_3, $f(x_3)$] and [x_4, $f(x_4)$]. As a consequence, the secant method can sometime diverge. A Matlab program for solving example 8.1 using secant method is listed in Table 8.3

Table (8.3) Matlab code and results for solving example (8.1) using secant method

Matlab Code	<pre>fun=inline('600*(1-exp(-0.15*x))/x-50')
	tol=1e-5;
	x(1)=1;
	f(1)= fun(x(1));
	x(2)=14;
	f(2)= fun(x(2));
	for i=2:20
	    x(i+1)=x(i)-f(i)*(x(i)-x(i-1))/(f(i)-f(i-1))
	    f(i+1)= fun(x(i+1));
	    if abs(x(i+1)-x(i))<tol;
	            break;
	    end
	end
	x</pre>

Results	<pre>x =
	    1.0000
	   14.0000
	   10.4956
	    8.3724
	    8.8209
	    8.7897
	    8.7892
	    8.7892</pre>The last value of x vector is the solution of the equation

Example 8.4

Use secant method to estimate the root of $f(x)=e^{-x}-x$. with the two initial guesses x_0 = 0. and x_1 = 1.

Solution:

Iter	x	f(x)
Starting Values	0	1.0000
	1.0	-0.6321
1	0.6127	-0.0708
2	0.563838	0.00518
3	0.56717	-0.0000418

Example 8.5

Repeat Examples 8.4 using the secant method with the two initial guesses x_0 = 2. and x_1 = 3.

Solution:

Iter	x	f(x)
Starting Values	2	-1.8647
	3	-2.9502
1	0.2823	0.4718
2	0.6570	-0.1385
3	0.5719	-0.0075
4	0.5671	

This method converges with the required accuracy after 5 iterations.

Example 8.6

Use secant method with initial guesses T = 300 and T = 350 to calculate the bubble point of binary system (VCM 18 mol%, Water 82 mol%). The vapor pressure for this components is calculated by:

VCM $P^o_{vcm}=\exp(14.9601-1803.84/(T-43.15))$

Water $P^o_w =\exp(18.3036-3816.44/(T-46.13))$

Where: $K_i= P^o_i/P_t$

$P_t=760$

$y_i=K_i \times x_i$

At Bubble point $\sum y_i=\sum K_i \times x_i=1$

Solution

$$f(T)= K_i \times x_i -1 = e^{(14.9601-\frac{1803.84}{T-43.15})} \times \frac{0.18}{760} + e^{(18.3036-\frac{3816.44}{T-46.13})} \times \frac{0.82}{760} -1$$

Iter	T	f(T)
Initial Values	300.0000	-0.3086
	350.0000	1.4192
1	308.9299	-0.1132
2	311.9636	-0.0378
3	313.4866	0.0018
4	313.4157	-0.000028

Then at bubble point T=313.457 K

8.4 The Newton-Raphson Method

The *Newton-Raphson* method and its modification is probably the most widely used of all root-finding methods. Starting with an initial guess x_1 at the root, the next guess x_2 is the intersection of the tangent from the point $[x_1, f(x_1)]$ to the x-axis. The next guess x_3 is the intersection of the tangent from the point $[x_2, f(x_2)]$ to the x-axis as shown in Figure 8.5. The process can be repeated until the desired tolerance is attained.

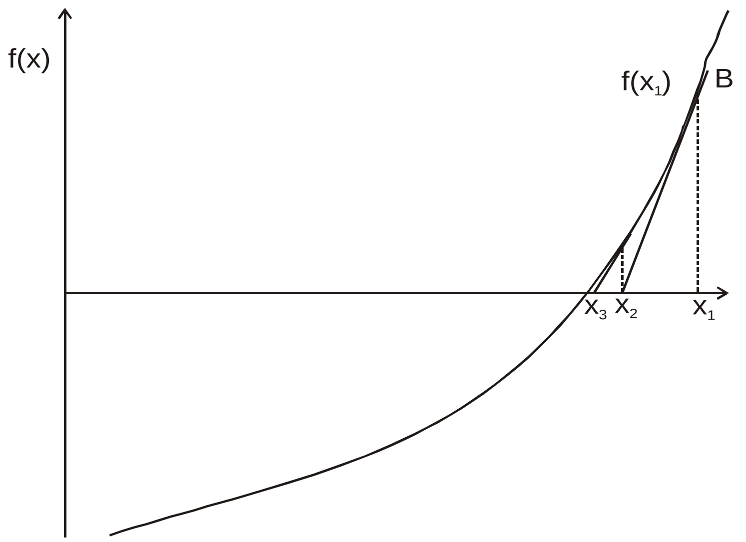


Figure 8.5 Graphical depiction of the *Newton-Raphson* method.

The derivative or slope $f(x_n)$ can be approximated numerically as

$$f'(x_n) = \frac{f(x_n+\Delta x)-f(x_n)}{\Delta x}$$

The *Newton-Raphson* method can be derived from the definition of a slope

$$f'(x_1) = \frac{f(x_1) - 0}{x_1 - x_2} \Rightarrow x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$$

In general, from the point $[x_n, f(x_n)]$, the next guess is calculated as

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

A Matlab program for solving example 8.1 using Newton- Raphson method is listed in Table 8.4 .

Table (8.4) Matlab code and results for solving example (8.1) using Newton-Raphson method	
Matlab Code	**f=inline('600*(1-exp(-0.15*x))/x-50');** **df=inline('(90.0*exp(-0.15*x))/x + (600*exp(-0.15*x) - 600)/x^2');** **tol=1e-5;** **x(1)=1;** **for i=2:20** **x(i+1)=x(i)-f(x(i))/df(x(i));** **if abs(x(i+1)-x(i))<tol;** **break;** **end** **end** **x'**
Results	**x =** **1.0000** **0.0031** **5.9278** **8.4473** **8.7840** **8.7892** **8.7892** The last value of x vector is the solution of the equation

Example 8.7

Use Newton-Raphson method to estimate the root of $f(x)=e^{-x}-x$. Show all details of the iterations. Hint: the root is located between 0 and 1.

Solution:

Iter	X_i	$f(X_i)$	$f'(X_i)$	X_{i+1}	$\lvert\varepsilon_a(\%)\rvert$
1	0.0	1.0	-2.0	0.5	100.00
2	0.5	0.1065	-1.6065	0.5663	11.71
3	0.5663	0.0013	-1.5676	0.5671	0.15
4	0.5671	0.0000	-1.5676	0.5671	0.00

Example 8.8

Repeat Example 8.7 starting with $x_o = 5$.

Solution:

Iter	X_i	$f(X_i)$	$f'(X_i)$	X_{i+1}	$\lvert\varepsilon_a(\%)\rvert$
1	5.0	-4.9933	-1.0067	0.04016	12351
2	0.04016	0.92048	-1.9606	0.5096	92.12
3	0.5096	0.0911	-1.6007	0.5665	10.04
4	0.5665	0.0010	-1.5675	0.5671	0.000

Example 8.9

Apply Newton-Raphson method to solve Redlich-Kwong equation which used to estimate the molar volume of saturated vapor of methyl chloride at 333.15 K and 13.76 bar

$$P=\frac{RT}{V-b}-\frac{a}{T^{0.5}V(V+b)}$$

If you know that:-

$A=1.5651\times10^{8}\ cm^{6}\ bar\ mol^{-2}\ K^{1/2}$

$b=44.891\ cm^{3}\ mol^{-1}$

$R=83.14\ cm^{3}.bar.K^{-1}.mol^{-1}$

Solution

$$f(V)=\frac{RT}{V-b}-\frac{a}{T^{0.5}V(V+b)}-P$$

$$f(V)=\frac{83.14\times333.15}{V-44.891}-\frac{1.5651\times10^{8}}{333.15^{0.5}V(V+44.891)}-13.76$$

$$f(V)=\frac{27698}{V-44.891}-\frac{8574764.131}{V(V+44.891)}-13.76$$

$$f'(V)=\frac{-27698}{(V-44.891)^2}+\frac{8574764.131}{V(V+44.891)^2}+\frac{8574764.131}{V^2(V+44.891)}$$

It’s better to start with ideal molar volume as initial value of V

$$V = \frac{RT}{P} = \frac{83.14 \times 333.15}{13.76} = 2012.943$$

Iter	V_i	$f(V_i)$	$f'(V_i)$	V_{i+1}
1	2012.943	-1.75623	-0.00512	1669.717
2	1669.717	0.291628	-0.00695	1711.673
3	1711.673	0.005731	-0.00668	1712.531
4	1712.531	0.00000231	-0.00667	1712.531

The molar volume equal to 1712.531 $cm^3\ mol^{-1}$

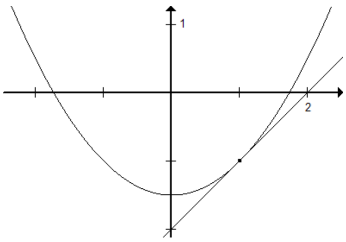
Solving System of Non-Linear Systems of Equations

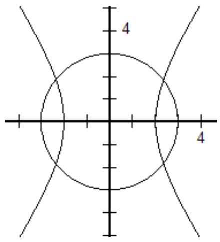
A ***nonlinear system of equations*** has at least one equation which is not first degree.

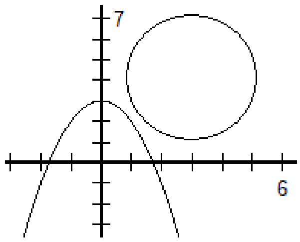
Examples: $x^2 + y^2 = 25$ $\quad$ $y = 3x^2 - 4x + 2$ $\quad$ $xy = 9$

$2x + 3y = 7$ $\quad$ $x^2 + y = 8$ $\quad$ $3x^2 - y^2 = 12$

The solutions of a nonlinear system are the points of intersection of the graphs of the equations. Some systems have one point of intersection; some have more than one point of intersection; and some have no points of intersection.







9.1 Analytical Methods for Solving Systems of Equations

Solutions of nonlinear systems of equations can be found using the substitution or the elimination method. The substitution method is preferable for a system with one linear equation. The elimination method is preferable in most, but not all, cases when both equations are nonlinear.

9.1.1. The Substitution Method

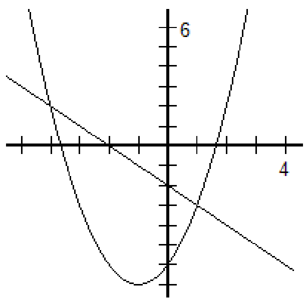
Solve one of the equations for a first degree variable. Substitute the resulting expression in for that variable in the other equation. Solve for the remaining variable. Back substitute to find the value(s) of the first variable. Write your solutions as ordered pairs.

Example 9.1

Solve $\quad x^2 + 2x = y + 6$

$\quad x + y = -2$

Solution



$x + y = -2 \;\rightarrow\; y = -2 - x \;\rightarrow\; x^2 + 2x = -2 - x + 6 \;\rightarrow$

$x^2 + 3x - 4 = 0 \;\rightarrow\; (x + 4)(x - 1) = 0 \;\rightarrow\; x = -4 \;\&\; x = 1$

$y = -2 - (-4) = 2 \quad \& \quad y = -2 - 1 = -3 \quad \rightarrow$

Solution: $(-4, 2)$ $\quad (1, -3)$

To check graphically, enter $y_1 = x^2 + 2x - 6$ $\quad \& \quad$ $y_2 = -2 - x$

Find the points of intersection.

9.1.2. The Elimination Method

Line the equations up vertically so like terms are underneath each other. If needed, multiply each equation by a number so that when the equations are added together one of the variables is eliminated. Solve for the remaining variable. Back substitute to find the value(s) of the eliminated variable. Write your solutions as ordered pairs.

Example 9.2

Solve $3x^2 + 5y^2 = 17$
$2x^2 - 3y^2 = 5$

Solution:

$3x^2 + 5y^2 = 17 \quad 3R_1 \quad \rightarrow \quad 9x^2 + 15y^2 = 51$

$2x^2 - 3y^2 = 5 \quad 5R_2 \quad \rightarrow \quad \underline{10x^2 - 15y^2 = 25}$

$19x^2 = 76$

$x^2 = 4 \quad \rightarrow \quad x = -2 \quad \& \quad x = 2$

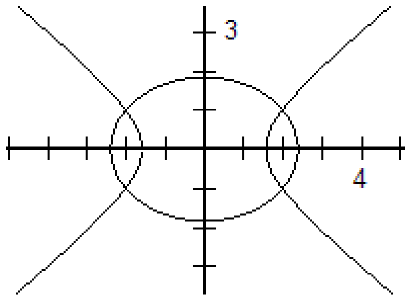
$3(-2)^2 + 5y^2 = 17 \quad \rightarrow \quad 12 + 5y^2 = 17 \quad \rightarrow$

$5y^2 = 5 \quad \rightarrow \quad y^2 = 1 \quad \rightarrow \quad y = -1 \quad \& \quad y = 1$

$3(2)^2 + 5y^2 = 17 \quad \rightarrow \quad 12 + 5y^2 = 17 \quad \rightarrow$

$5y^2 = 5 \quad \rightarrow \quad y^2 = 1 \quad \rightarrow \quad y = -1 \quad \& \quad y = 1$

Solutions: (–2, –1) (–2, 1) (2, –1) (2, 1)



9.2 Numerical Methods for Solving Systems of Equations

9.2.1 Fixed point iteration for systems of non-linear equations

Using an initial guess, solve for each variable in the system and use fixed-point iteration to estimate the solution.

One of the most important drawbacks of the fixed iteration method is that the convergence of the method is dependent on how the equations are formulated. May diverge quickly, in that case try solving for the variables in a different way.

It can be shown that sufficient convergence criteria for two equations are:

$$\left|\frac{\partial f_1}{\partial x_1}\right| + \left|\frac{\partial f_1}{\partial x_2}\right| < 1$$

and

$$\left|\frac{\partial f_2}{\partial x_1}\right| + \left|\frac{\partial f_2}{\partial x_2}\right| < 1$$

This represents a very restrictive criteria and that's why fixed point iteration method is not used to solve systems of non-linear equations.

Example 9.3

Solve using 5 iteration of successive substitution where x = y = 1.5 initially:

$$\begin{cases} x^2 + y^2 = 5 \\ y - x^2 = -1 \end{cases}$$

Solution

Using $x = \sqrt{5 - y^2}$ and y = $x^2 - 1$ it is easy to show the following iterations:

Iteration	x_n	y_n
0	1.5	1.5
1	1.658	1.75
2	1.392	0.9375
3	2.030	3.121
4	Non-real	Non-real
5		

So it is apparent that successive substitution this will not work using these formulas.

Using $y = \sqrt{5 - x^2}$ and $x = \sqrt{1 + y}$ we can just as easily show the following iterations:

Iteration	x_n	y_n
0	1.5	1.5
1	1.5811	1.5811
2	1.607	1.555
3	1.599	1.564
4	1.601	1.561
5	1.600	1.562

Which is rapidly converging on the true solution of $\left(\sqrt{\frac{1+\sqrt{17}}{2}}, \ \frac{-1+\sqrt{17}}{2} \right)$

As with the Jacobi iterative process, convergence is assured only for a system of diagonally dominant linear equations. For systems that are neither linear nor diagonally dominant, convergence is a function of the equations themselves as well as the values of x's chosen to start the iterations.

Example 9.4

As an example of applying the Jacobi method to a system of non-linear equations, consider the following system:

$$4\sqrt{x_1} - x_2 + x_3 = 7$$

$$4x_1 - 8(x_2)^2 + x_3 = -21$$

$$-2x_1 + x_2 + 5(x_3)^2 = 15$$

Solving the equations for each of the unknowns (x's), we have the following:

$$x_1 = \left(\frac{7+x_2-x_3}{4}\right)^2$$

$$x_2 = \sqrt{\frac{-21-4x_1-x_3}{-8}}$$

$$x_3 = \sqrt{\frac{15+2x_1-x_2}{5}}$$

Using these relationships in a Jacobi algorithm with starting values of $x_1 = x_2 = x_3 = 0$, we can show the convergence of the algorithm over ten iterations in the following table:

Itr	x_1	x_2	x_3
0	0	0	0
1	3.0625	1.620185	1.732051
2	2.9654	2.091114	1.975086
3	3.164866	2.086764	1.941118
4	3.191267	2.109519	1.961783
5	3.193133	2.113257	1.963314
6	3.195105	2.113523	1.963314
7	3.195343	2.113757	1.963501
8	3.195385	2.113790	1.963514
9	3.195403	2.113796	1.963516
10	3.195406	2.113798	1.963518

9.2.2 Newton-Raphson for solving systems of non-linear equations

The Newton-Raphson formula is the following:

$$x_{i+1} = x_i - \frac{f(x_i)}{f^{'}(x_i)}$$

This formula can be obtained using Taylor series expansion. We can do the same approach for a system of equations, but considering a Taylor series that account for the presence of both variables:

$$f_{1(i+1)} = f_{1(i)} + \left(x_{1(i+1)} - x_{1(i)}\right)\frac{\partial f_{1(i)}}{\partial x_1} + \left(x_{2(i+1)} - x_{2(i)}\right)\frac{\partial f_{1(i)}}{\partial x_2} + \ldots$$

and

$$f_{2(i+1)} = f_{2(i)} + \left(x_{1(i+1)} - x_{1(i)}\right)\frac{\partial f_{2(i)}}{\partial x_1} + \left(x_{2(i+1)} - x_{2(i)}\right)\frac{\partial f_{2(i)}}{\partial x_2} + \ldots$$

For the root estimate $f_{1(i+1)}$ and $f_{2(i+1)}$ must be equal zero.

Therefore:

$$\frac{\partial f_{1(i)}}{\partial x_1} x_{1(i+1)} + \frac{\partial f_{1(i)}}{\partial x_2} x_{2(i+1)} = -f_{1(i)} + x_{1(i)} \frac{\partial f_{1(i)}}{\partial x_1} + x_{2(i)} \frac{\partial f_{1(i)}}{\partial x_2}$$

and

$$\frac{\partial f_{2(i)}}{\partial x_1} x_{1(i+1)} + \frac{\partial f_{2(i)}}{\partial x_2} x_{2(i+1)} = -f_{2(i)} + x_{1(i)} \frac{\partial f_{2(i)}}{\partial x_1} + x_{2(i)} \frac{\partial f_{2(i)}}{\partial x_2}$$

Finally;

$$x_{1(i+1)} = x_{1(i)} - \frac{f_{1(i)} \frac{\partial f_{2(i)}}{\partial x_2} - f_{2(i)} \frac{\partial f_{1(i)}}{\partial x_2}}{\frac{\partial f_{1(i)}}{\partial x_1} \frac{\partial f_{2(i)}}{\partial x_2} - \frac{\partial f_{1(i)}}{\partial x_2} \frac{\partial f_{2(i)}}{\partial x_1}}$$

$$x_{2(i+1)} = x_{2(i)} - \frac{f_{2(i)} \frac{\partial f_{1(i)}}{\partial x_1} - f_{1(i)} \frac{\partial f_{2(i)}}{\partial x_1}}{\frac{\partial f_{1(i)}}{\partial x_1} \frac{\partial f_{2(i)}}{\partial x_2} - \frac{\partial f_{1(i)}}{\partial x_2} \frac{\partial f_{2(i)}}{\partial x_1}}$$

Which is an iterative method to solve the system of nonlinear equations.

Note also that the Newton-Raphson method can be generalized to solve N simultaneous equations.

Example 9.5

Solve the following system using Newton-Raphson method:

$$x^2 + y^2 - 8x - 4y + 11 = 0$$

$$x^2 + y^2 - 20x + 75 = 0$$

By tacking a starting point as (x=2; y=4) and $\varepsilon = 10^{-5}$.

Solution

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

$$y_{(i+1)} = y_{(i)} - \frac{f_{2(i)} \frac{\partial f_{1(i)}}{\partial x} - f_{1(i)} \frac{\partial f_{2(i)}}{\partial x}}{\frac{\partial f_{1(i)}}{\partial x} \frac{\partial f_{2(i)}}{\partial y} - \frac{\partial f_{1(i)}}{\partial y} \frac{\partial f_{2(i)}}{\partial x}}$$

Let $f_1 = x^2 + y^2 - 8x - 4y + 11$ and $f_2 = x^2 + y^2 - 20x + 75$

Thus: $\frac{\partial f_{1,i}}{\partial x} = 2x - 8$, $\frac{\partial f_{1,i}}{\partial y} = 2y - 4$, $\frac{\partial f_{2,i}}{\partial x} = 2x - 20$, and $\frac{\partial f_{2,i}}{\partial y} = 2y$

Hence when x =2 and y = 4 we find that:

$$\frac{\partial f_{1,0}}{\partial x} = 2(2) - 8 = -4, \quad \frac{\partial f_{1,0}}{\partial y} = 2(4) - 4 = 4, \quad \frac{\partial f_{2,0}}{\partial x} = 2(2) - 20 = -16, \quad \text{and} \quad \frac{\partial f_{2,0}}{\partial y} = 2(4) = 8$$

Also $f_{1,0} = 2^2 + 4^2 - 8(2) - 4(4) + 11 = -1$ and $f_{2,0} = 2^2 + 4^2 - 20(2) + 75 = 55$

$$\frac{\partial f_{1,0}}{\partial x}\frac{\partial f_{2,0}}{\partial y} - \frac{\partial f_{1,0}}{\partial y}\frac{\partial f_{2,0}}{\partial x} = (-4)(8) - (4)(-16) = 32$$

So for the first iteration we see that:

$$x_1 = 2 - \frac{(-1)(8) - (55)(4)}{32} \approx 9.1250$$

$$y_1 = 4 - \frac{(55)(-4) - (-1)(-16)}{32} \approx 11.3750$$

$$\varepsilon = \left|\frac{x_0 - x_1}{x_0}\right| = \frac{2 - 9.125}{2} = 3.5625$$

Now we find that iteration 2 produces:

$$\frac{\partial f_{1,1}}{\partial x} = 2(9.1250) - 8 = 10.25, \quad \frac{\partial f_{1,1}}{\partial y} = 2(11.375) - 4 = 18.75, \quad \frac{\partial f_{2,1}}{\partial x} = 2(9.1250) - 20 = -1.75,$$

and $\frac{\partial f_{2,1}}{\partial y} = 2(11.375) = 22.75$

Also $f_{1,1} = 9.125^2 + 11.375^2 - 8(9.125) - 4(11.375) + 11 = 105.1563$ and $f_{2,1} = 9.125^2 + 11.375^2 - 20(9.125) + 75 = 105.1563$

$$\frac{\partial f_{1,1}}{\partial x}\frac{\partial f_{2,1}}{\partial y} - \frac{\partial f_{1,1}}{\partial y}\frac{\partial f_{2,1}}{\partial x} = (10.25)(22.75) - (18.75)(-1.75) = 266$$

So for the second iteration we see that:

$$x_2 = 9.125 - \frac{(105.1563)(22.75) - (105.1563)(18.75)}{266} \approx 7.543703$$

$$y_2 = 11.375 - \frac{(105.1563)(10.25) - (105.1563)(-1.75)}{266} \approx 6.631109$$

$$\varepsilon = \left|\frac{x_0 - x_1}{x_0}\right| = \left|\frac{9.125 - 7.543703}{9.125}\right| = 0.173293$$

Itr	f_1	f_2	$\frac{\partial f_1}{\partial x}$	$\frac{\partial f_1}{\partial y}$	$\frac{\partial f_2}{\partial x}$	$\frac{\partial f_2}{\partial y}$	x	y	ε_s
							2	4	
1	-1	55	-4	4	-16	8	9.125	11.375	3.5625
2	105.1563	105.1563	10.25	18.75	-1.75	22.75	7.543703	6.631109	0.173293
3	25.005	25.005	7.087406	9.262218	-4.91259	13.26222	6.826694	4.480083	0.095047
4	5.141014	5.141014	5.653389	4.960166	-6.34661	8.960166	6.576327	3.728981	0.036675
5	0.626838	0.626838	5.152654	3.457963	-6.84735	7.457963	6.535955	3.607865	0.006139
6	0.016299	0.016299	5.07191	3.215731	-6.92809	7.215731	6.534848	3.604543	0.000169
7	1.23E-05	1.23E-05	5.069696	3.209087	-6.9303	7.209087	6.534847	3.604541	1.28×10^{-7}

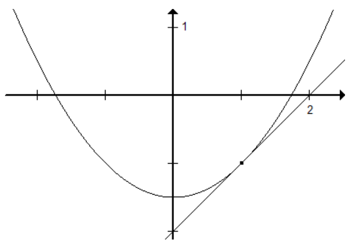
Solving System of Non-Linear Systems of Equations

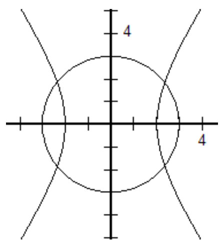
A ***nonlinear system of equations*** has at least one equation which is not first degree.

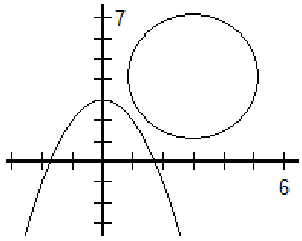
Examples: $x^2 + y^2 = 25$ $y = 3x^2 - 4x + 2$ $xy = 9$

$2x + 3y = 7$ $x^2 + y = 8$ $3x^2 - y^2 = 12$

The solutions of a nonlinear system are the points of intersection of the graphs of the equations. Some systems have one point of intersection; some have more than one point of intersection; and some have no points of intersection.







9.1 Analytical Methods for Solving Systems of Equations

Solutions of nonlinear systems of equations can be found using the substitution or the elimination method. The substitution method is preferable for a system with one linear equation. The elimination method is preferable in most, but not all, cases when both equations are nonlinear.

9.1.1. The Substitution Method

Solve one of the equations for a first degree variable. Substitute the resulting expression in for that variable in the other equation. Solve for the remaining variable. Back substitute to find the value(s) of the first variable. Write your solutions as ordered pairs.

Example 9.1

Solve $x^2 + 2x = y + 6$

$x + y = -2$

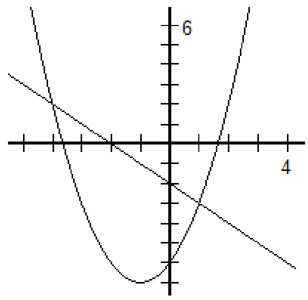
Solution

$x + y = -2 \rightarrow y = -2 - x \rightarrow x^2 + 2x = -2 - x + 6 \rightarrow$

$x^2 + 3x - 4 = 0 \rightarrow (x + 4)(x - 1) = 0 \rightarrow x = -4$ & $x = 1$

$y = -2 - (-4) = 2$ & $y = -2 - 1 = -3 \rightarrow$

Solution: $(-4, 2)$ $(1, -3)$



To check graphically, enter $y_1 = x^2 + 2x - 6$ & $y_2 = -2 - x$

Find the points of intersection.

9.1.2. The Elimination Method

Line the equations up vertically so like terms are underneath each other. If needed, multiply each equation by a number so that when the equations are added together one of the variables is eliminated. Solve for the remaining variable. Back substitute to find the value(s) of the eliminated variable. Write your solutions as ordered pairs.

Example 9.2

Solve $3x^2 + 5y^2 = 17$

$2x^2 - 3y^2 = 5$

Solution:

$3x^2 + 5y^2 = 17 \quad 3R_1 \quad \rightarrow \quad 9x^2 + 15y^2 = 51$

$2x^2 - 3y^2 = 5 \quad 5R_2 \quad \rightarrow \quad \underline{10x^2 - 15y^2 = 25}$

$19x^2 = 76$

$x^2 = 4 \quad \rightarrow \quad x = -2 \quad \& \quad x = 2$

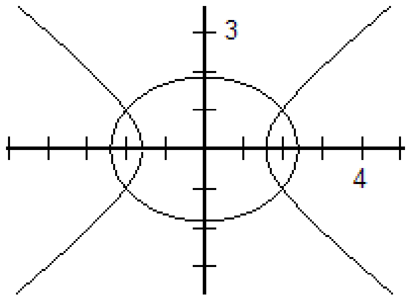
$3(-2)^2 + 5y^2 = 17 \quad \rightarrow \quad 12 + 5y^2 = 17 \quad \rightarrow$

$5y^2 = 5 \quad \rightarrow \quad y^2 = 1 \quad \rightarrow \quad y = -1 \quad \& \quad y = 1$

$3(2)^2 + 5y^2 = 17 \quad \rightarrow \quad 12 + 5y^2 = 17 \quad \rightarrow$

$5y^2 = 5 \quad \rightarrow \quad y^2 = 1 \quad \rightarrow \quad y = -1 \quad \& \quad y = 1$

Solutions: (–2, –1) (–2, 1) (2, –1) (2, 1)



9.2 Numerical Methods for Solving Systems of Equations

9.2.1 Fixed point iteration for systems of non-linear equations

Using an initial guess, solve for each variable in the system and use fixed-point iteration to estimate the solution.

One of the most important drawbacks of the fixed iteration method is that the convergence of the method is dependent on how the equations are formulated. May diverge quickly, in that case try solving for the variables in a different way.

It can be shown that sufficient convergence criteria for two equations are:

$$\left|\frac{\partial f_1}{\partial x_1}\right| + \left|\frac{\partial f_1}{\partial x_2}\right| < 1$$

and

$$\left|\frac{\partial f_2}{\partial x_1}\right| + \left|\frac{\partial f_2}{\partial x_2}\right| < 1$$

This represents a very restrictive criteria and that's why fixed point iteration method is not used to solve systems of non-linear equations.

Example 9.3

Solve using 5 iteration of successive substitution where x = y = 1.5 initially:

$$\begin{cases} x^2 + y^2 = 5 \\ y - x^2 = -1 \end{cases}$$

Solution

Using $x = \sqrt{5 - y^2}$ and y = x^2 – 1 it is easy to show the following iterations:

Iteration	x_n	y_n
0	1.5	1.5
1	1.658	1.75
2	1.392	0.9375
3	2.030	3.121
4	Non-real	Non-real
5		

So it is apparent that successive substitution this will not work using these formulas.

Using $y = \sqrt{5 - x^2}$ and $x = \sqrt{1 + y}$ we can just as easily show the following iterations:

Iteration	x_n	y_n
0	1.5	1.5
1	1.5811	1.5811
2	1.607	1.555
3	1.599	1.564
4	1.601	1.561
5	1.600	1.562

Which is rapidly converging on the true solution of $\left(\sqrt{\frac{1+\sqrt{17}}{2}},\ \frac{-1+\sqrt{17}}{2}\right)$

As with the Jacobi iterative process, convergence is assured only for a system of diagonally dominant linear equations. For systems that are neither linear nor diagonally dominant, convergence is a function of the equations themselves as well as the values of x's chosen to start the iterations.

Example 9.4

As an example of applying the Jacobi method to a system of non-linear equations, consider the following system:

$$4\sqrt{x_1} - x_2 + x_3 = 7$$

$$4x_1 - 8(x_2)^2 + x_3 = -21$$

$$-2x_1 + x_2 + 5(x_3)^2 = 15$$

Solving the equations for each of the unknowns (x's), we have the following:

$$x_1 = \left(\frac{7 + x_2 - x_3}{4}\right)^2$$

$$x_2 = \sqrt{\frac{-21 - 4x_1 - x_3}{-8}}$$

$$x_3 = \sqrt{\frac{15 + 2x_1 - x_2}{5}}$$

Using these relationships in a Jacobi algorithm with starting values of $x_1 = x_2 = x_3 = 0$, we can show the convergence of the algorithm over ten iterations in the following table:

Itr	x_1	x_2	x_3
0	0	0	0
1	3.0625	1.620185	1.732051
2	2.9654	2.091114	1.975086
3	3.164866	2.086764	1.941118
4	3.191267	2.109519	1.961783
5	3.193133	2.113257	1.963314
6	3.195105	2.113523	1.963314
7	3.195343	2.113757	1.963501
8	3.195385	2.113790	1.963514
9	3.195403	2.113796	1.963516
10	3.195406	2.113798	1.963518

9.2.2 Newton-Raphson for solving systems of non-linear equations

The Newton-Raphson formula is the following:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

This formula can be obtained using Taylor series expansion. We can do the same approach for a system of equations, but considering a Taylor series that account for the presence of both variables:

$$f_{1(i+1)} = f_{1(i)} + \left(x_{1(i+1)} - x_{1(i)}\right)\frac{\partial f_{1(i)}}{\partial x_1} + \left(x_{2(i+1)} - x_{2(i)}\right)\frac{\partial f_{1(i)}}{\partial x_2} + \ldots$$

and

$$f_{2(i+1)} = f_{2(i)} + \left(x_{1(i+1)} - x_{1(i)}\right)\frac{\partial f_{2(i)}}{\partial x_1} + \left(x_{2(i+1)} - x_{2(i)}\right)\frac{\partial f_{2(i)}}{\partial x_2} + \ldots$$

For the root estimate $f_{1(i+1)}$ and $f_{2(i+1)}$ must be equal zero.

Therefore:

$$\frac{\partial f_{1(i)}}{\partial x_1}x_{1(i+1)}+\frac{\partial f_{1(i)}}{\partial x_2}x_{2(i+1)}=-f_{1(i)}+x_{1(i)}\frac{\partial f_{1(i)}}{\partial x_1}+x_{2(i)}\frac{\partial f_{1(i)}}{\partial x_2}$$

and

$$\frac{\partial f_{2(i)}}{\partial x_1}x_{1(i+1)}+\frac{\partial f_{2(i)}}{\partial x_2}x_{2(i+1)}=-f_{2(i)}+x_{1(i)}\frac{\partial f_{2(i)}}{\partial x_1}+x_{2(i)}\frac{\partial f_{2(i)}}{\partial x_2}$$

Finally;

$$x_{1(i+1)}=x_{1(i)}-\frac{f_{1(i)}\frac{\partial f_{2(i)}}{\partial x_2}-f_{2(i)}\frac{\partial f_{1(i)}}{\partial x_2}}{\frac{\partial f_{1(i)}}{\partial x_1}\frac{\partial f_{2(i)}}{\partial x_2}-\frac{\partial f_{1(i)}}{\partial x_2}\frac{\partial f_{2(i)}}{\partial x_1}}$$

$$x_{2(i+1)}=x_{2(i)}-\frac{f_{2(i)}\frac{\partial f_{1(i)}}{\partial x_1}-f_{1(i)}\frac{\partial f_{2(i)}}{\partial x_1}}{\frac{\partial f_{1(i)}}{\partial x_1}\frac{\partial f_{2(i)}}{\partial x_2}-\frac{\partial f_{1(i)}}{\partial x_2}\frac{\partial f_{2(i)}}{\partial x_1}}$$

Which is an iterative method to solve the system of nonlinear equations.

Note also that the Newton-Raphson method can be generalized to solve N simultaneous equations.

Example 9.5

Solve the following system using Newton-Raphson method:

$$x^2+y^2-8x-4y+11=0$$

$$x^2+y^2-20x+75=0$$

By tacking a starting point as (x=2; y=4) and $\varepsilon=10^{-5}$.

Solution

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

$$y_{(i+1)}=y_{(i)}-\frac{f_{2(i)}\frac{\partial f_{1(i)}}{\partial x}-f_{1(i)}\frac{\partial f_{2(i)}}{\partial x}}{\frac{\partial f_{1(i)}}{\partial x}\frac{\partial f_{2(i)}}{\partial y}-\frac{\partial f_{1(i)}}{\partial y}\frac{\partial f_{2(i)}}{\partial x}}$$

Let $f_1=x^2+y^2-8x-4y+11$ and $f_2=x^2+y^2-20x+75$

Thus: $\frac{\partial f_{1,i}}{\partial x}=2x-8,\quad \frac{\partial f_{1,i}}{\partial y}=2y-4,\quad \frac{\partial f_{2,i}}{\partial x}=2x-20,$ and $\frac{\partial f_{2,i}}{\partial y}=2y$

Hence when x =2 and y = 4 we find that:

$$\frac{\partial f_{1,0}}{\partial x} = 2(2) - 8 = -4, \quad \frac{\partial f_{1,0}}{\partial y} = 2(4) - 4 = 4, \quad \frac{\partial f_{2,0}}{\partial x} = 2(2) - 20 = -16, \quad \text{and} \quad \frac{\partial f_{2,0}}{\partial y} = 2(4) = 8$$

Also $f_{1,0} = 2^2 + 4^2 - 8(2) - 4(4) + 11 = -1$ and $f_{2,0} = 2^2 + 4^2 - 20(2) + 75 = 55$

$$\frac{\partial f_{1,0}}{\partial x}\frac{\partial f_{2,0}}{\partial y} - \frac{\partial f_{1,0}}{\partial y}\frac{\partial f_{2,0}}{\partial x} = (-4)(8) - (4)(-16) = 32$$

So for the first iteration we see that:

$$x_1 = 2 - \frac{(-1)(8) - (55)(4)}{32} \approx 9.1250$$

$$y_1 = 4 - \frac{(55)(-4) - (-1)(-16)}{32} \approx 11.3750$$

$$\varepsilon = \left|\frac{x_0 - x_1}{x_0}\right| = \frac{2 - 9.125}{2} = 3.5625$$

Now we find that iteration 2 produces:

$$\frac{\partial f_{1,1}}{\partial x} = 2(9.1250) - 8 = 10.25, \quad \frac{\partial f_{1,1}}{\partial y} = 2(11.375) - 4 = 18.75, \quad \frac{\partial f_{2,1}}{\partial x} = 2(9.1250) - 20 = -1.75,$$

and $\frac{\partial f_{2,1}}{\partial y} = 2(11.375) = 22.75$

Also $f_{1,1} = 9.125^2 + 11.375^2 - 8(9.125) - 4(11.375) + 11 = 105.1563$ and $f_{2,1} = 9.125^2 + 11.375^2 - 20(9.125) + 75 = 105.1563$

$$\frac{\partial f_{1,1}}{\partial x}\frac{\partial f_{2,1}}{\partial y} - \frac{\partial f_{1,1}}{\partial y}\frac{\partial f_{2,1}}{\partial x} = (10.25)(22.75) - (18.75)(-1.75) = 266$$

So for the second iteration we see that:

$$x_2 = 9.125 - \frac{(105.1563)(22.75) - (105.1563)(18.75)}{266} \approx 7.543703$$

$$y_2 = 11.375 - \frac{(105.1563)(10.25) - (105.1563)(-1.75)}{266} \approx 6.631109$$

$$\varepsilon = \left|\frac{x_0 - x_1}{x_0}\right| = \left|\frac{9.125 - 7.543703}{9.125}\right| = 0.173293$$

Itr	f_1	f_2	$\frac{\partial f_1}{\partial x}$	$\frac{\partial f_1}{\partial y}$	$\frac{\partial f_2}{\partial x}$	$\frac{\partial f_2}{\partial y}$	x	y	ε_s
							2	4	
1	-1	55	-4	4	-16	8	9.125	11.375	3.5625
2	105.1563	105.1563	10.25	18.75	-1.75	22.75	7.543703	6.631109	0.173293
3	25.005	25.005	7.087406	9.262218	-4.91259	13.26222	6.826694	4.480083	0.095047
4	5.141014	5.141014	5.653389	4.960166	-6.34661	8.960166	6.576327	3.728981	0.036675
5	0.626838	0.626838	5.152654	3.457963	-6.84735	7.457963	6.535955	3.607865	0.006139
6	0.016299	0.016299	5.07191	3.215731	-6.92809	7.215731	6.534848	3.604543	0.000169
7	1.23E-05	1.23E-05	5.069696	3.209087	-6.9303	7.209087	6.534847	3.604541	1.28×10^{-7}

Solution of First-Order Ordinary Differential Equations

An equation that consists of derivatives is called a differential equation. Differential equations have applications in all areas of science and engineering. Mathematical formulation of most of the physical and engineering problems lead to differential equations. So, it is important for engineers and scientists to know how to set up differential equations and solve them. Differential equations are of two types

1) Ordinary differential equation (ODE).
2) Partial differential equations (PDE).

An ordinary differential equation is that in which all the derivatives are with respect to a single independent variable. Examples of ordinary differential equation include:

1) $\frac{dy}{dx} + y = \sin(x)$, $y(0) = 1$,

2) $\frac{d^2y}{dx^2} + 2\frac{dy}{dx} + y = 0$, $\frac{dy}{dx}(0) = 2$, $y(0) = 4$

3) $\frac{d^3y}{dx^3} + 3\frac{d^2y}{dx^2} + 5\frac{dy}{dx} + y = \sin x$, $\frac{d^2y}{dx^2}(0) = 12$, $\frac{dy}{dx}(0) = 2$, $y(0) = 4$

First order ordinary differential equations are of the form:

$$\frac{dx}{dt} = f(x,t) \text{ with } x(0) = x_0$$

On the left hand side is the derivative of the dependent variable *x* with respect to the independent variable *t*. On the right hand side, there is a function that may depend on both *x* and *t*.

Many differential equations cannot be solved exactly. Numerical methods have been developed to approximate solutions. Numerical analysis is a field in mathematics that is concerned with developing approximate numerical methods and assessing their accuracy, for instance for solving differential equations. We will discuss the most basic method such Taylor, Euler and Runge-Kutta methods.

10.1 Taylor Series Method

Function $y(x)$ can be expanded over a small interval *x* using the ***Taylor series*** from a start or reference point x

$$y(x+h) = y(x) + hy'(x) + \frac{1}{2!}h^2y''(x) + \frac{1}{3!}h^3y'''(x) + \frac{1}{4!}h^4y^{(4)}(x) + \Lambda \qquad (1)$$

Where $h_i = x_{i+1} - x_i = h$, a constant.

Example 10.1

Solve the following ordinary differential equation (ODE) using Taylor's method of order 2 with h= 0.2

$$\frac{dy}{dx} = y - x^2 + 1 \quad , \quad \text{for } 0 \le x \le 2 \quad , \quad \text{with } y(0) = 0.5$$

Solution

$$\frac{d^2y}{dx^2} = \frac{dy}{dx} - 2x = y - x^2 + 1 - 2x$$

$$y_{n+1} = y_n + (y_n - x_n^2 + 1)h + \frac{1}{2}(y_n - x_n^2 - 2x_n + 1)h^2$$

i	x	y
1	0	0.5000
2	0.2000	0.8300
3	0.4000	1.2158
4	0.6000	1.6521
5	0.8000	2.1323
6	1.0000	2.6486
7	1.2000	3.1913
8	1.4000	3.7486
9	1.6000	4.3061
10	1.8000	4.8463
11	2.0000	5.3477

10.2. Euler's Method

Euler's method is the simplest and least useful of these three methods. If we are solving a first-order differential equation of the form $\frac{dy}{dt} = f(t, y)$ with the initial condition $y(0)=A$, Euler's method begins by approximating the first derivative as

$$\frac{dy}{dt} \approx \frac{y(t + \Delta t) - y(t)}{\Delta t}$$

Setting this equal to $f(t,y)$ and solving for $y(t + \Delta t)$ yields the following algorithm for advancing the numerical solution of an ordinary differential equation:

$$\frac{dy}{dx} = f(x,y) \qquad y(0) = y_0$$

$$y_{n+1} = y_n + h \times f(x, y)$$

Using Euler's method we have the following consideration:

$$x_1 = x_0 + h \qquad y_1 = y_0 + h \cdot f(x_0, y_0)$$
$$x_2 = x_1 + h \qquad y_2 = y_1 + h \cdot f(x_1, y_1)$$
$$x_3 = x_2 + h \qquad y_3 = y_2 + h \cdot f(x_2, y_2)$$

M M

M M

M M

Exercise 10.2:

Apply Euler's method to approximate the solution of the initial value problem

$$\frac{dy}{dx} = 2y \quad \text{with} \quad y(0) = 5 \qquad (2)$$

Solution

We know that the analytical solution of equation (2) is , $y = 5\exp(2x)$. We numerically solve equation (2) using Euler's method with h=0.1 in the time interval [0, 0.5], and then check how well this method performs. We have $f(y) = 2y$. Then

$$x_0 = 0$$
$$x_1 = x_0 + h = 0 + 0.1 = 0.1$$
$$x_2 = x_1 + h = 0.1 + 0.1 = 0.2$$
$$x_3 = x_2 + h = 0.2 + 0.1 = 0.3$$
$$x_4 = x_3 + h = 0.3 + 0.1 = 0.4$$
$$x_5 = x_4 + h = 0.4 + 0.1 = 0.5$$

And

$$y_0 = 5$$
$$y_1 = y_0 + hf(y_0) = 5 + \underbrace{(0.1)}_{h}\underbrace{(2)(5)}_{f(y_0)} = 6$$
$$y_2 = y_1 + hf(y_1) = 6 + (0.1)(2)(6) = 7.2$$
$$y_3 = y_2 + hf(y_2) = 7.2 + (0.1)(2)(7.2) = 8.64$$
$$y_4 = y_3 + hf(y_3) = 8.64 + (0.1)(2)(8.64) = 10.368$$
$$y_5 = y_4 + hf(y_4) = 10.368 + (0.1)(2)(10.368) = 12.4416$$

We summarize this in the following table. If h=0.1, then

x	y	Exact	Difference
0	5	5	0
0.1	6	6.107014	0.107014
0.2	7.2	7.459123	0.259123
0.3	8.64	9.110594	0.470594
0.4	10.368	11.1277	0.759705
0.5	12.4416	13.59141	1.149809

The third column contains the exact values, $y = 5\exp(2x)$. The last column contains the absolute error after each step, computed as $|y - y_{Exact}|$. We see that when *h*=0.1, the numerical approximation is not very good after five steps. If we repeat the same approximation with a smaller value for *h*, say *h*=0.01, the following table results for the first five steps:

x	y	Exact	Difference
0	5	5	0
0.01	5.1	5.101007	0.001007
0.02	5.202	5.204054	0.002054
0.03	5.30604	5.309183	0.003143
0.04	5.412161	5.416435	0.004275
0.05	5.520404	5.525855	0.005451

Doing five steps only gets us to *x*=0.05. We can do more steps until we reach *x*=0.5. We find that the final point will be:

x	y	Exact	Difference
0.5	13.45794	13.59141	0.133469

Choosing a smaller value for *h* resulted in a better approximation at *x*=0.5 but also required more steps. One source of error in the approximation comes from the approximation itself.

10.3 Fourth order Runge-Kutta Method

To find numerical solution to the initial value problem $\frac{dy}{dx} = f(x, y),\ y(0) = y_0$ using Runge-Kutta method we have the following consideration:

$$\boxed{y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)h}$$

$$k_1 = f(x_i, y_i)$$

$$k_2 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1 h\right)$$

$$k_3 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_2 h\right)$$

$$k_4 = f(x_i + h, y_i + k_3 h)$$

This method gives more accurate result compared to Euler's method

Example 10.3:

Solve the following ordinary differential equation (ODE) using fourth order Runge-Kutta method to calculate y(x=0.2)

$$\frac{dy}{dx} = x + y \;; \quad y(0) = 1 \quad , \quad h = 0.1$$

Solution:

$k_1 = 0 + 1 = 1$

$k_2 = (0+0.05)+(1+1\times0.05)=1.10$

$k_3 = (0+0.05) +(1+1.1\times0.05)= 1.1050$

$k_4 = (0+0.1)+(1+1.1050\times0.1) = 1.2105$

$$y_1 = y_o + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$y(0.1) = 1 + \frac{0.1}{6} \times (1 + 2\times1.1 + 2\times1.105 + 1.2105) = 1.11034$$

$k_1 = 0.1 + 1.11034 = 1.21034$

$k_2 = (0.1+0.05)+(1.11034+1.21034\times0.05) = 1.3209$

$k_3 = (0.1+0.05) +(1.11034+1.3209\times0.05)= 1.3264$

$k_4 = (0.1+0.1)+(1.11034+1.3264\times0.1) = 1.4430$

$$y(0.2) = 1.11034 + \frac{0.1}{6} \times (1.21034 + 2\times1.3209 + 2\times1.3264 + 1.4430) = 1.2428$$

at x=0.2   y=1.2428

Example 10.4:

A ball at 1200 K is allowed to cool down in air at an ambient temperature of 300 K. Assuming heat is lost only due to radiation, the differential equation for the temperature of the ball is given by

$$\frac{dT}{dt} = -2.2067\times10^{-12}\left(T^4 - 81\times10^8\right) \quad , \quad T(0) = 1200\ \text{K}$$

where T is in K and t in seconds. Find the temperature at $t = 480$ seconds using Runge-Kutta 4th order method. Assume a step size of $h = 240$ seconds.

Solution

$$\frac{dT}{dt} = -2.2067 \times 10^{-12}\left(T^4 - 81\times10^8\right)$$

$$f\left(t,T\right) = -2.2067 \times 10^{-12}\left(T^4 - 81\times10^8\right)$$

$$T_{i+1} = T_i + \frac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right)$$

For $i = 0$, $t_0 = 0$, $T_0 = 1200\ K$

$$k_1 = f\left(t_0, T_0\right) = f\left(0,1200\right) = -2.2067 \times 10^{-12}\left(1200^4 - 81\times10^8\right) = -4.5579$$

$$k_2 = f\left(t_0 + \frac{1}{2}h, T_0 + \frac{1}{2}k_1h\right) = f\left(0 + \frac{1}{2}(240), 1200 + \frac{1}{2}(-4.5579)\times 240\right)$$

$$= f\left(120,653.05\right) = -2.2067 \times 10^{-12}\left(653.05^4 - 81\times10^8\right)$$

$$= -0.38347$$

$$k_3 = f\left(t_0 + \frac{1}{2}h, T_0 + \frac{1}{2}k_2h\right) = f\left(0 + \frac{1}{2}(240), 1200 + \frac{1}{2}(-0.38347)\times 240\right)$$

$$= f\left(120,1154.0\right) = -2.2067 \times 10^{-12}\left(1154.0^4 - 81\times10^8\right) = -3.8954$$

$$k_4 = f\left(t_0 + h, T_0 + k_3h\right) = f\left(0 + 240, 1200 + (-3.894)\times 240\right) = f\left(240,265.10\right)$$

$$= -2.2067 \times 10^{-12}\left(265.10^4 - 81\times10^8\right) = 0.069750$$

$$T_1 = T_0 + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$= 1200 + \frac{240}{6}\left(-4.5579 + 2(-0.38347) + 2(-3.8954) + (0.069750)\right)$$

$$= 675.65\,K$$

T_1 is the approximate temperature at $t=t_1$

$$t = t_0 + h = 0 + 240 = 240$$

For $i = 1, t_1 = 240, T_1 = 675.65\,K$

$$k_1 = f\left(t_1, T_1\right) = f\left(240,675.65\right) = -2.2067 \times 10^{-12}\left(675.65^4 - 81\times10^8\right)$$

$$= -0.44199$$

$$k_2 = f\left(t_1 + \frac{1}{2}h, T_1 + \frac{1}{2}k_1h\right) = f\left(240 + \frac{1}{2}(240), 675.65 + \frac{1}{2}(-0.44199)240\right)$$

$$= f\left(360,622.61\right) = -2.2067 \times 10^{-12}\left(622.61^4 - 81\times10^8\right)$$

$= -0.31372$

$$k_3 = f\left(t_1 + \frac{1}{2}h, T_1 + \frac{1}{2}k_2 h\right) = f\left(240 + \frac{1}{2}(240), 675.65 + \frac{1}{2}(-0.31372) \times 240\right)$$

$= f(360, 638.00)$

$= -2.2067 \times 10^{-12}(638.00^4 - 81 \times 10^8)$

$= -0.34775$

$$k_4 = f(t_1 + h, T_1 + k_3 h) = f(240 + 240, 675.65 + (-0.34775) \times 240) = f(480, 592.19)$$

$= 2.2067 \times 10^{-12}(592.19^4 - 81 \times 10^8)$

$= -0.25351$

$$T_2 = T_1 + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$= 675.65 + \frac{240}{6}(-0.44199 + 2(-0.31372) + 2(-0.34775) + (-0.25351))$$

$= 594.91\,K$

T_2 is the approximate temperature at time t_2

$t_2 = t_1 + h \quad = 240 + 240 = 480$

Table 1 and Figure 2 show the effect of step size on the value of the calculated temperature at $t = 480$ seconds.

Table 1 Value of temperature at time, $t = 480$s for different step sizes

Step size, h	T(480)	E_t	$\lvert \varepsilon_t \rvert$%
480	-90.278	737.85	113.94
240	594.91	52.660	8.1319
120	646.16	1.4122	0.21807
60	647.54	0.033626	0.0051926
30	647.57	0.00086900	0.00013419

Example 10.5

Using Matlab Commands solve the following equation using both Eular and Runge-Kutta method and to approximate the solution of the initial value problem $\frac{dy}{dx} = x + y, \quad y(0) = 1$ with step size h = 0.1.

Solution:

Eular	**Runge Kutta**

clear all, clc,format short
x(1)=0;
y(1)=1;
h=0.5
for i=1:5
x(i+1)=x(i)+h;
dy=x(i)+y(i);
y(i+1)=y(1)+h*dy;
end
y_exact= -1-x+2*exp(x);
error=y_exact-y
table=[x',y',y_exact',error']
	clear all, clc,format short
	x(1)=0;y(1)=1;h=0.1;
	f=inline('x+y');
	% f(x,y) = x+y
	for i=1:5
	x(i+1)=x(i)+h;
	k1 = f(x(i),y(i));
	k2 = f(x(i)+h/2,y(i)+k1*h/2);
	k3 = f(x(i) + h/2,y(i)+k2*h/2);
	k4 = f(x(i) + h,y(i)+k3*h);
	y(i+1)=y(i)+(1/6)*h*(k1 +2*k2 + 2*k3 +k4);
	end
	y_exact= -1-x+2*exp(x); error=y_exact-y
	table=[x',y',y_exact',error']

table = 0 1.0000 1.0000 0 0.1000 1.1000 1.1103 0.0103 0.2000 1.1200 1.2428 0.1228 0.3000 1.1320 1.3997 0.2677 0.4000 1.1432 1.5836 0.4404 0.5000 1.1543 1.7974 0.6431	table = 0 1.0000 1.0000 0 0.1000 1.1103 1.1103 0.0000 0.2000 1.2428 1.2428 0.0000 0.3000 1.3997 1.3997 0.0000 0.4000 1.5836 1.5836 0.0000 0.5000 1.7974 1.7974 0.0000

11.1 Integration two simultaneous first-order ordinary differential equations

Consider the following system of first-order ODE's describing the dependence of two dependent variables y and z on one independent variable x:

$$\frac{dy}{dx} = f(x, y, z)$$

$$\frac{dz}{dx} = g(x, y, z)$$

These two differential equations are coupled and must be integrated simultaneously because both equations involve both dependent variables.

Initial conditions are required giving the values of y and z at the initial value of x. The algorithm for 4th-order Runge-Kutta integration of two coupled ODEs is:

$$y_{i+1} = y_i + \frac{h}{6}(k_{11} + 2k_{21} + 2k_{31} + k_{41})$$

$$z_{i+1} = z_i + \frac{h}{6}(k_{12} + 2k_{22} + 2k_{32} + k_{42})$$

$$k_{11} = f(x_i, y_i, z_i)$$

$$k_{12} = g(x_i, y_i, z_i)$$

$$k_{21} = f(x_i + 0.5h, y_i + 0.5hk_{11}, z_i + 0.5hk_{12})$$

$$k_{22} = g(x_i + 0.5h, y_i + 0.5hk_{11}, z_i + 0.5hk_{12})$$

$$k_{32} = f(x_i + 0.5h, y_i + 0.5hk_{21}, z_i + 0.5hk_{22})$$

$$k_{32} = g(x_i + 0.5h, y_i + 0.5hk_{21}, z_i + 0.5hk_{22})$$

$$k_{41} = f(x_i + h, y_i + hk_{31}, z_i + hk_{32})$$

$$k_{42} = g(x_i + h, y_i + hk_{31}, z_i + hk_{32})$$

As example an exothermic reaction in unsteady-state continuous stirred tank reactor and exothermic reaction in a plug flow reactor with heat exchange through the reactor wall.

From the one and two ODE examples, you can extend the method to integration of three coupled ODE's. Three coupled ODE's would be encountered, for example, for reaction of gases in a steady-state non-isothermal plug flow reactor with significant pressure drop (dC/dx =, dT/dx =, and dP/dx=).

11.2 Integration of a system of first-order ordinary differential equations

$$\frac{dy_1}{dx} = f_1(x, y_1, y_2, \dots, y_m)$$

$$\frac{dy_2}{dx} = f_2(x, y_1, y_2, \dots, y_m)$$

....

....

$$\frac{dy_m}{dx} = f_m(x, y_1, y_2, \dots, y_m)$$

The solution of the above equations is:

$$\boxed{y_i^{n+1} = y_i^n + \frac{h}{6}(k_{1,i} + 2k_{2,i} + 2k_{3,i} + k_{4,i})}$$ Where i = 1, 2, ..., m and

$$k_{1,i} = f_i(x^n, y_1^n, y_2^n \dots, y_m^n)$$

$$k_{2,i} = f_i(x^n + \frac{h}{2}, y_1^n + \frac{hk_{1,1}}{2}, y_2^n + \frac{hk_{1,2}}{2} \dots, y_m^n + \frac{hk_{1,m}}{2})$$

$$k_{3,i} = f_i(x^n + \frac{h}{2}, y_1^n + \frac{hk_{2,1}}{2}, y_2^n + \frac{hk_{2,2}}{2} \dots, y_m^n + \frac{hk_{2,m}}{2})$$

$$k_{4,i} = f_i(x^n + h, y_1^n + hk_{3,1}, y_2^n + hk_{3,2} \dots, y_m^n + hk_{3,m})$$

The idea of the solution to a system of differential equations is similar to a solution of a single differential equation.

Example 11.1:

Using fourth order Runge-Kutta method with step size h = 0.1 solve

$$\frac{dy_1}{dx} = y_1 y_2 + x \quad , \quad y_1(0) = 1$$

$$\frac{dy_2}{dx} = xy_2 + y_1 \quad , \quad y_2(0) = -1$$

To calculate $y_1(0.1)$ and $y_2(0.1)$

Solution

At x = 0, $y_1 = 1$, $y_2 = -1$

$k_{1,1} = y_1 y_2 + x = (1)(-1) + 0 = -1$

$k_{1,2} = xy_2 + y_1 = (0)(-1) + 1 = 1$

$k_{2,1} = (y_1+0.5hk_{1,1})(y_2+0.5hk_{1,2}) + (x+0.5h) = (0.95)(-0.95) + 0.05 = -0.8525$

$k_{2,2} = (x+0.5h)(y_2+0.5hk_{1,2}) +(y_1+0.5hk_{1,1}) =(0.05)(-0.95) + 0.95 = 0.9025$

$k_{3,1} = (y_1+0.5hk_{2,1})(y_2+0.5hk_{2,2}) + (x+0.5h) =(0.9574)(-0.9549) + 0.05= -0.8642$

$k_{3,2} = (x+0.5h)(y_2+0.5hk_{2,2}) +(y_1+0.5hk_{2,1}) =(0.05)(-0.9549) + 0.9574 = 0.9096$

$k_{4,1} = (y_1+hk_{3,1})(y_2+hk_{3,2}) + (x+h) =(0.9136)(-0.9091) + 0.1 = -0.7305$

$k_{4,2} = (x+h)(y_2+hk_{3,2}) +(y_1+hk_{3,1}) =(0.1)(-0.9091) + 0.9136= 0.8227$

at x=0.1

$y_1(0.1) = y_1(0) + (h/6)(k_{1,1} + 2k_{2,1} + 2k_{3,1} + k_{4,1})$

$y_1(0.1) = 1 +(0.1/6) [(-1) + 2(-0.8525) + 2(-0.864) + (-0.730)]=$ **0.9139**

$y_2(0.1) = y_2(0) +(h/6) (k_{1,2} + 2k_{2,2} + 2k_{3,2} + k_{4,2})$

$y_2(0.1) = -1 + [(1) + 2(0.9025) + 2(0.909) + (0.823)]=$ **-0.9092**

Example 11.2

Use ode45 Matlab Command to solve the following first order system for y_1 and y_2 at $0 \le x \le 1$.

$$\frac{dy_1}{dx} = y_1 y_2 + x \qquad , y_1(0) = 1$$

$$\frac{dy_2}{dx} = xy_2 + y_1 \qquad , y_2(0) = -1$$

Using fourth order Runge-Kutta method with step size h = 0.1

Solution

The Matlab routines **ode45** can be used to solve the system. A Matlab function must be created to evaluate the slopes as a **column** vector. The function name in this example is ***exode(x, y)*** which must be saved first in the hard drive with the same name ***exode.m***.

```
function dydx = exode(x,y)
dydx (1,1)=y(1)*y(2)+x;
dydx (2,1)=x*y(2)+y(1);
```

The command **ode45** is then evaluated from the command windows. Matlab will set the step size to achieve a preset accuracy that can be changed by user.

The independent variable can also be specified at certain locations between the initial and final values and Matlab will provide the dependent value at these locations.

```
xspan=0:0.1:1;
[x,y]=ode45('exode',xspan,[1 , -1])
```

x =	y =	
0	1.0000	-1.0000
0.1000	0.9139	-0.9092
0.2000	0.8522	-0.8341
0.3000	0.8106	-0.7711
0.4000	0.7863	-0.7174
0.5000	0.7772	-0.6705
0.6000	0.7817	-0.6283
0.7000	0.7987	-0.5889
0.8000	0.8274	-0.5504
0.9000	0.8675	-0.5108
1.0000	0.9188	-0.4681

Exercise 11.3:

Let's consider a simple example of a model of a plug flow reactor that is described by a system of ordinary differential equations. A plug flow reactor is operated as shown in Figure (9.1) below.

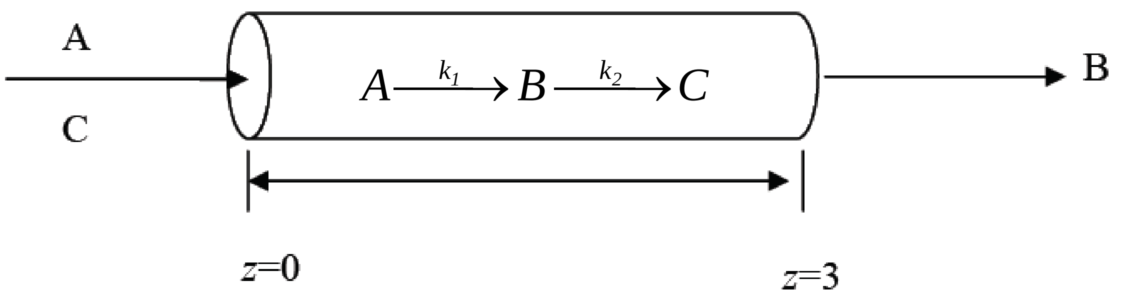


Figure (9.1) Isothermal plug flow reactor

The plug flow initially has only reactant A, the components A react to form component B. The mole balance for each component is given by the following differential equations

$$u\frac{dC_A}{dz} = -k_1 C_A$$

$$u\frac{dC_B}{dz} = k_1 C_A - k_2 C_B$$

$$u\frac{dC_C}{dz} = k_2 C_B$$

With the following initial values

C_A (z=0) =1 kmol/m^3 C_B (z=0) =0 C_C (z=0) =0 and k_1=2 k_2=3

If u=0.5 m/s and reactor length z=3 m. Solve the differential equations and plot the concentration of each species along the reactor length

Solution:

We'll start by writing the function defining the right hand side (RHS) of the ODEs. The following function file 'example3' is used to set up the ode solver.

```
function dC= Example4 ( z, C)
u = 0.5;
k1=2; k2=3;
dC(1,1) = -k1 *C(1) / u;
dC(2,1) = (k1 *C(1)-k2 *C(2)) / u;
dC(3,1) = k2 *C(2)/ u;
```

Now we'll write a main script file to call ode45. CA, CB and CC must be defined within the same matrix, and so by calling CA as C(1), CB as C(2) and CC as c(3), they are listed as common to matrix C.

The following run file is created to obtain the solution:

```
clear all, clc
[z , C] = ode45(' Example3', [0:0.1:3], [1 0 0])
plot (z,C(:,1),'k+-',z,C(:,2),'k*:',z,C(:,3),'kd-.')
xlabel ('Length (m)');
ylabel ('Concentrations (kmol/m^3) ');
legend ('A', 'B', 'C')
```

The produced plot is as in Figure (9.2)

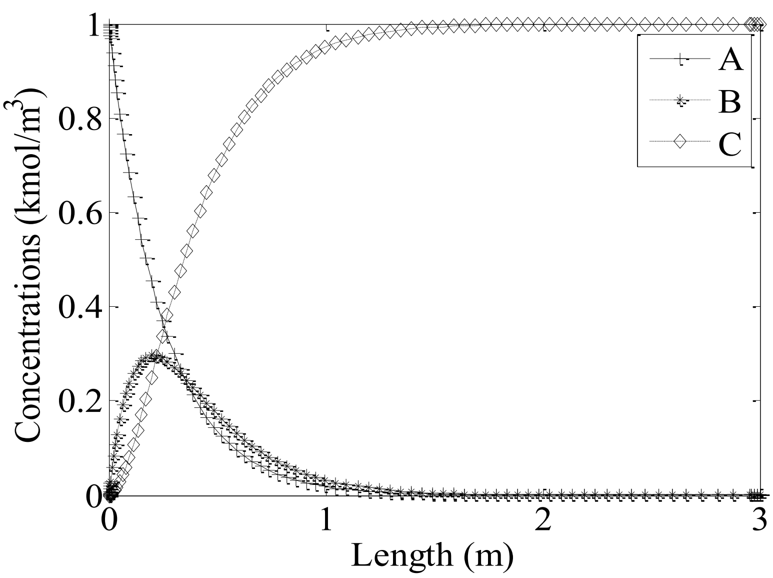


Figure (9.2): A, B and C concentrations along plug flow reactor

11.3 Solving Higher Order Ordinary Differential Equations

We have learned Euler's and Runge-Kutta methods to solve first order ordinary differential equations of the form

$$\frac{dy}{dx} = f(x, y),\ y(0) = y_0$$

What do we do to solve differential equations that are higher than first order? For example an n^{th} order differential equation of the form

$$a_n \frac{d^n y}{dx^n} + a_{n-1} \frac{d^{n-1} y}{dx^{n-1}} + \ldots + a_1 \frac{dy}{dx} + a_o y = f(x)$$

with $n-1$ initial conditions can be solved by assuming

$$y = z_1 \qquad (1)$$

$$\frac{dy}{dx} = \frac{dz_1}{dx} = z_2 \qquad (2)$$

$$\frac{d^2 y}{dx^2} = \frac{dz_2}{dx} = z_3 \qquad (3)$$

$$\vdots$$

$$\frac{d^{n-1} y}{dx^{n-1}} = \frac{dz_{n-1}}{dx} = z_n \qquad (n)$$

$$\frac{d^n y}{dx^n} = \frac{dz_n}{dx}$$

$$= \frac{1}{a_n}\left(-a_{n-1} \frac{d^{n-1} y}{dx^{n-1}} \ldots - a_1 \frac{dy}{dx} - a_0 y + f(x)\right)$$

$$= \frac{1}{a_n}\left(-a_{n-1} z_n \ldots - a_1 z_2 - a_0 z_1 + f(x)\right) \qquad (n+1)$$

The above Equations from (2) to (n+1) represent n first order differential equations as follows

$$\frac{dz_1}{dx} = z_2 = f_1(z_1, z_2, \ldots, x)$$

$$\frac{dz_2}{dx} = z_3 = f_2(z_1, z_2, \ldots, x)$$

$$\vdots$$

$$\frac{dz_n}{dx} = \frac{1}{a_n}\left(-a_{n-1} z_n \ldots - a_1 z_2 - a_0 z_1 + f(x)\right)$$

Each of the *n* first order ordinary differential equations is accompanied by one initial condition. These first order ordinary differential equations are simultaneous in nature but can be solved by the methods used for solving first order ordinary differential equations that we have already learned.

Higher Order Ordinary Differential Equations	System of first Order Ordinary Differential Equations
$\frac{d^2y}{dx^2}-2\frac{dy}{dx}+2y=e^{2x}\sin x$ with y(0)= -0.4 , $\frac{dy}{dx}(0)=-0.6$	$\frac{dy}{dx}=z \quad , \quad y(0)=-0.4$ $\frac{dz}{dx}=e^{2x}\sin x-2y+2z \quad , \quad z(0)=-0.6$
$\frac{d^3y}{dx^3}+2\frac{d^2y}{dx^2}-3\frac{dy}{dx}=x^2+y,$ with $\frac{d^2y}{dx^2}(0)=1, \quad \frac{dy}{dx}(0)=2, \quad y(0)=4$	$\frac{dy}{dx}=z, \quad y(0)=4$ $\frac{dz}{dx}=u, \quad z(0)=2$ $\frac{du}{dx}=x^2+y-2u+3z, \quad u(0)=1$
$\frac{d^4y}{dx^4}+-\frac{dy}{dx}+3y=\frac{y}{x},$ with $\frac{d^3y}{dx^3}(0)=0.5, \frac{d^2y}{dx^2}(0)=0.25, \frac{dy}{dx}(0)=1, y(0)=1$	$\frac{dy}{dx}=z, \quad y(0)=1$ $\frac{dz}{dx}=u, \quad z(0)=1$ $\frac{du}{dx}=v, \quad u(0)=0.25$ $\frac{dv}{dx}=\frac{y}{x}-3y+z, \quad v(0)=0.5$

Example 11.4

Re-write the following differential equation as a set of first order differential equations.

$$3\frac{d^2y}{dx^2}+2\frac{dy}{dx}+5y=e^{-x}, \; y(0)=5, \; y'(0)=7$$

Solution

The ordinary differential equation would be rewritten as follows. Assume

$$\frac{dy}{dx}=z, \quad \text{Then} \quad \frac{d^2y}{dx^2}=\frac{dz}{dx}$$

Substituting this in the given second order ordinary differential equation gives

$$3\frac{dz}{dx}+2z+5y=e^{-x}$$

$$\frac{dz}{dx}=\frac{1}{3}\left(e^{-x}-2z-5y\right)$$

The set of two simultaneous first order ordinary differential equations complete with the initial conditions then is

$$\frac{dy}{dx} = z,\ y(0) = 5$$

$$\frac{dz}{dx} = \frac{1}{3}\left(e^{-x} - 2z - 5y\right),\ z(0) = 7\,.$$

Now one can apply any of the numerical methods used for solving first order ordinary differential equations.

Example 11.5

Given the third-order ordinary differential equation and associated initial conditions

$$\frac{d^3y}{dx^3} + 3\frac{d^2y}{dx^2} + 5\frac{dy}{dx} + y = x^2 \quad , \quad y(0) = 4,\ \left.\frac{dy}{dx}\right|_{x=0} = 0.6\ ,\ \left.\frac{d^2y}{dx^2}\right|_{x=0} = 0.22$$

a. Write this differential equation as a system of first-order ordinary differential equations

b. Using fourth order Runge-Kutaa method to estimate y(0.1) and y(0.2) taking Δx=0.1

a)

$$\frac{dy}{dx} = z\,, \quad y(0) = 4$$

$$\frac{dz}{dx} = u\,, \quad z(0) = 0.6$$

$$\frac{du}{dx} = x^2 - 3u - 5z - y\,, \quad u(0) = 0.22$$

b) Solution (1)

First step of integration x=0 , y=4 , z=0.6 , u=0.22 , Δx=0.1

k11= z =0.6000

k21= u= 0.2200

k31=x2-3×u-5×z-y= 02-3×0.22-5×0.6-4=-7.6600

k12=z+0.5×Δx×k21=0.6+0.5×0.1×0.22= 0.6110

k22=(u+0.5×Δx×k31) =(0.22+0.5×0.1×(-7.6600)) = -0.1630

k32=(x+0.5×Δx)2 -3×(u+0.5×Δx×k31)-5×(z+0.5×Δx×k21)-(y+0.5×Δx×k11)

=(0+0.5×0.1) 2-3×(0.22+0.5×0.1×(-7.66))-5×(0.6+0.5×0.1×0.22)-(4+0.5×0.1×0.6)

= -6.5935

k13=z+0.5×Δx×k22=0.6+0.5×0.1×(-0.1630)= 0.5918

k23=u+0.5×Δx×k32 =0.22+0.5×0.1×(-6.5935) = -0.1097

k33=(x+0.5×Δx) 2 -3×(u+0.5×Δx×k32)-5×(z+0.5×Δx×k22)-(y+0.5×Δx×k12)

=(0+0.5×0.1)2-3×(0.22+0.5×0.1×(-6.5935))-5×(0.6+0.5×0.1×(-0.1630))-(4+0.5×0.1× 0.6110) = -6.6583

k14=z+Δx×k23=0.6+0.1×(-0.1097)= 0.5890

k24=u+Δx×k33=0.22+0.1×(-6.6583) = -0.4458

k34=(x+Δx) 2 -3×(u+Δx×k33)-5×(z+Δx×k23)-(y+Δx×k13)

=(0+0.1)2-3×(0.22+0.1×(-6.6583))-5×(0.6+0.1×(-0.1097))-(4+0.1×0.5918) = -5.6569

x=x+Δx=0+0.1= 0.1000

y=y+Δx/6×(k11+2×k12+2×k13+k14)=4+0.1/6×(0.6+2×0.6110+2×0.5918+0.5890) = 4.0599

z=z+Δx/6*(k21+2*k22+2*k23+k24)=0.6+0.1/6×(0.2200+2×(-0.1630)+2×(-0.1097)+(-0.4458))= 0.5871

u=u+Δx/6*(k31+2*k32+2*k33+k34)=u+0.1/6×(-7.6600+2×(-6.5935)+2×(-6.6583)+(-5.6569))=-0.4437

Then at x=0.1 , **y=4.0599**, $z = \frac{dy}{dx} = 0.5871$, $u = \frac{dz}{dx} = \frac{d^2y}{dx^2} = -0.4437$

Second step of integration x=0.1 , z=0.5871 , u=-0.4437 , Δx=0.1

k11= z = 0.5871

k21= u= -0.4437

k31=x2-3×u-5×z-y= 0.12-3×(-0.4437)-5×(0.5871)- 4.0599=-5.6546

k12=z+0.5×Δx×k21=0.5871+0.5×0.1×(-0.4437)= 0.5650

k22=(u+0.5×Δx×k31) =((-0.4437)+0.5×0.1×(-5.6546)) = -0.7264

k32=(x+0.5×Δx)2-3×(u+0.5×Δx×k31)-5×(z+0.5×Δx×k21)-(y+0.5×Δx×k11)

=(0.1+0.5×0.1)2-3×(-0.4437+0.5×0.1×(-5.6546))-5×(0.5871+0.5×0.1×(-0.4437))-(4.0599+0.5×0.1×0.5871) = -4.7124

k13=z+0.5×Δx×k22=0.5871+0.5×0.1×(-0.7264)= 0.5508

k23=u+0.5×Δx×k32 =-0.4437+0.5×0.1×(-4.7124) = -0.6793

k33=(x+0.5×Δx)2 -3×(u+0.5×Δx×k32)-5×(z+0.5×Δx×k22)-(y+0.5×Δx×k12)

=(0.1+0.5×0.1)2-3×(-0.4437+0.5×0.1×(-4.7124))-5×(0.5871+0.5×0.1×(-0.7264))-(4.0599+0.5×0.1×0.5650) = -4.7819

k14=z+Δx×k23=0.5871+0.1×(-0.6793)= 0.5192

k24=u+Δx×k33=-0.4437+0.1×(-4.7819) = -0.9219

k34=(x+Δx)2-3×(u+Δx×k33)-5×(z+Δx×k23)-(y+Δx×k13)

=(0.1+0.1)2-3×(-0.4437+0.1×(-4.7819))-5×(0.5871+0.1×(-0.6793))-(4.0599+0.1×0.5 508) = -3.9055

$x=x+\Delta x=0.1+0.1= 0.2$

$y=y+\Delta x/6\times(k11+2\times k12+2\times k13+k14)=$

$4.0599+0.1/6\times(0.5871+2\times0.5650+2\times0.5508+0.5192)= 4.1155$

$z=z+\Delta x/6*(k21+2*k22+2*k23+k24)=$

$0.5871+0.1/6\times((-0.4437)+2\times(-0.7264)+2\times(-0.6793)+(-0.9219))= 0.5175$

$u=u+\Delta x/6*(k31+2*k32+2*k33+k34) =$

$-0.4437 +0.1/6\times((-5.6546) +2\times(-4.7124)+2\times(-4.7819)+ (-3.9055))= -0.9195$

Then at x=0.2 , **y=4.1155**, $z = \frac{dy}{dx} = 0.5175$, $u = \frac{dz}{dx} = \frac{d^2y}{dx^2} = -0.9195$

Solution (2) Using Matlab:

We'll start by writing the function defining the right hand side (RHS) of the ODEs. The following function file 'Ex' is used to set up the ode solver.

```
function dq = Ex(x,q)
y=q(1);z=q(2);u=q(3);
dq(1,1)=z;
dq(2,1)=u;
dq(3,1)=x^2-3*u-5*z-y;
```

The following run file is created to obtain the solution:

```
clear all,clc,format compact
[x,q]=ode45('Ex',[0:0.1:0.2],[4,0.6,0.22])
y=q(:,1)
```

The produced results will be

```
x =
         0
    0.1000
    0.2000
q =
    4.0000    0.6000    0.2200
    4.0599    0.5871   -0.4437
    4.1155    0.5175   -0.9195
y =
    4.0000
    4.0599
    4.1155
```