Exact and Approximate Numbers:

The numbers that arise in technical applications are better described as "exact numbers" because there is not the sort of uncertainty in their values that was described above. They are the result of counting discrete items. For example, one dozen eggs is exactly 12 eggs, never 12.1 eggs or 11.9 eggs.

Approximate number is defined as a number approximated to the exact number and there is always a difference between the exact and approximate numbers.
For example, $3, 6, 9$ are exact numbers as they do not need any approximation.
But, $\sqrt{2}, \pi, \sqrt{3}$ are approximate numbers as they cannot be expressed exactly by a finite digits. They can be written as $1.414, 3.1416, 1.7320$ etc. which are only approximations to the true values.

Rules for Rounding Off Numbers

(i) Discard all the digits to the right of the $n\ th$ place, if the $(n+1)th$ digitis less than 5, leave the $n\ th$ digit unchanged. If the $(n+1)th$ digit is greater than 5 add one to the $n\ th$ digit.
Ex: If 27.73 is rounded off to three decimal places, the result is 27.7, since the digit 3 is being dropped. If 27.76 is rounded off to three decimal places, the value is 27.8, since the digit 6 is being dropped.

(ii) If the discarded digit is exactly 5 then leave the $n\ th$ digit unaltered if it's an even number and add one to the $n\ th$ digit if it's an odd number.
Ex: If 27.75 is rounded off to three significant figures, the value 27.8 results, since mber only the digit 5 is being dropped.
If 9.2652 is rounded off to three significant figures, the value 9.27 results, since the digit 5 is being dropped.

Significant figures

The digits which are used to represent a number are called significant figures. Thus, $1, 2, \ldots, 9$ are always significant and 0 is significant except if it is used to fix decimal places or to discard digits or to fill the unknown places. Zeros between two significant digits are always significant.

Ex:

Number	Significant figures
100	1
0.00123	3
10.23	4

Types of errors

1. **Inherent Errors**:The errors that are already present in the statement of the problem before its solution are called inherent errors. We cannot control this kind of errors.

   Ex:

x	0	1	2
y	1.13	**1.001**	0.98

   Printing mistake:

x	0	1	2
y	1.13	**1.01**	0.98

2. **Computational Errors:**There are two types of computational errors such as

(a) Round-off Errors: It arises when a calculated number is rounded off to a fixed number of digits; the difference between the exact and the rounded off number is called Round-off error. We can minimize this error by taking more decimal places during the calculation and at the last step round- off the number upto its desired accuracy.

Ex: Let the exact number $= 29.3257$
Its 3 decimal places approximation $= 29.326$
Therefore, the Round-off error$= (29.3257 \sim 29.326)$

(b) Truncation Errors: If approximation is used or infinite process be replaced by finite one during calculation, then the errors involved are called Truncation Errors. We can minimize this error by taking more decimal places during the computation or by taking more terms in the infinite expansion.

Ex:$cosx = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots$
Let this infinite series be truncated to $cosx = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} = C$(say)
Then the truncation error $= (cosx \sim C)$

Computation of errors

1. **Error:**Let $V_T =$ True value
   and$V_A =$ Approximate value of a number. Then the error is
   $$E = V_T - V_A.$$

2. **Absolute Error:**The absolute error is denoted by E_A and defined by
   $$E_A = |V_T - V_A|.$$

3. **Relative Error:** The relative error is denoted by E_R and defined by

$$E_R = \frac{E_A}{V_T} = \frac{|V_T - V_A|}{V_T}.$$

4. **Percentage Error:** The percentage error is denoted by E_P and defined by

$$\boldsymbol{E_P = E_R \times 100\% = \frac{|V_T - V_A|}{V_T} \times 100\%}$$

Problem: Approximate $\frac{2}{3}$ to 4 significant figures and find the absolute error, relative error and percentage error.

Interpolation

Let the analytic formula of $f(x)$ is not known but the values of $f(x)$ are known for $(n+1)$ distinct values of x, say, $x_0, x_1, x_2, \dots, x_n$ (node points) and the corresponding entries are given by $y_0 = f(x_0),\ y_1 = f(x_1),\ \dots, y_n = f(x_n)$. Our problem is to compute the value of (x) , at least approximately, for a given node point x lies in the vicinity of the above given values of the node points. The process by which we can find the required value of $f(x)$ for any other value of x in the interval $[x_0, x_n]$ is called Interpolation. When x lies slightly outside the interval $[x_0, x_n]$ then the process is called Extrapolation.

Newton's Forward Interpolation Formula

Let a function $f(x)$ is known for $(n+1)$ distinct and equispaced node points $x_0,\ x_1,\ \dots,\ x_{r-1},\ x_r,\ x_{r+1}, \dots,\ x_{n-1}, x_n$ such that $x_r = x_0 + rh, r = 0, 1, 2, \dots, n$ and h is the step length. The corresponding entries are given by $y_0 = f(x_0),\ y_1 = f(x_1),\ \dots, y_n = f(x_n)$.

Now, $x_n - x_r = x_0 + nh - (x_0 + rh)$

$$= (n-r)h \dots\dots\dots\dots\dots\dots\dots\dots (1)$$

Our objective is to find a polynomial $P(x)$ of degree less than or equal to n such that $P(x)$ replaces $f(x)$ on the set of node points x_r , $r = 0(1)n$ ie,

$P(x_r) = f(x_r)$, $r = 0(1)n$ ……(2)

Let us take the form of $P(x)$ as

$$P(x) = A_0 + A_1(x - x_0) + A_2(x - x_0)(x - x_1) + A_3(x - x_0)(x - x_1)$$

$(x - x_2) + \dots + A_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}) \dots\dots\dots\dots\dots\dots\dots (3)$

The constants $A_j, (j = 0(1)n)$ are to be determined by using equation (2).

Putting $x = x_0$ in equation (3) we get,

$$P(x_0) = A_0$$

or, $y_0 = A_0$

Putting $x = x_1$ in equation (3) we get,

$$P(x_1) = A_0 + A_1(x_1 - x_0)$$

$$\text{or, } y_1 = y_0 + A_1(h)$$

$$\text{or, } A_1 = \frac{y_1 - y_0}{h} = \frac{\Delta y_0}{h}.$$

Putting $x = x_2$in equation (3) we get,

$$P(x_2) = A_0 + A_1(x_2 - x_0) + A_2(x_2 - x_0)(x_2 - x_1)$$

or, $y_2 = y_0 + \left(\frac{y_1 - y_0}{h}\right)(2h) + A_2(2h)(h)$

or, $2h^2 A_2 = y_2 - y_0 + 2(y_1 - y_0)$

or, $A_2 = \frac{\Delta^2 y_0}{2h^2}$

Putting $x = x_3$in equation (3) we get,

$$P(x_3) = A_0 + A_1(x_3 - x_0) + A_2(x_3 - x_0)(x_3 - x_1) + A_3(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)$$

or, $y_3 = y_0 + \left(\frac{y_1 - y_0}{h}\right)(3h) + \frac{y_2 - 2y_1 + y_0}{2h^2}(3h)(2h) + A_3(3h)(2h)(h)$

or, $3!\, h^3 A_3 = y_3 - 3y_1 + 3y_2 - y_0 = \Delta^3 y_0$

or, $A_3 = \frac{\Delta^3 y_0}{3! h^3}$

Similarly, $A_r = \frac{\Delta^r y_0}{r! h^r}$.

Substituting $A_r{}'$s value in (3) we have,

$$f(x) \approx P(x) = y_0 + (x - x_0)\frac{\Delta y_0}{h} + (x - x_0)(x - x_1)\frac{\Delta^2 y_0}{2!\, h^2} + (x - x_0)(x - x_1)$$

$$(x - x_2)\frac{\Delta^3 y_0}{3!\, h^3} + \cdots + (x - x_0)(x - x_1)\cdots(x - x_{n-1})\frac{\Delta^n y_n}{n!\, h^n}$$

This formula is known as Newton's Forward Interpolation Formula. It is used mainly in computing $f(x)$ when x lies in the beginning of the table.

Example: Find the value of $f(0.5)$suitable interpolation formula from the following table:

x	0	1	2	3
$f(x)$	1	2	11	34

Ans: 0.875.

Newton's Backward Interpolation Formula

Let a function $f(x)$ is known for $(n+1)$ distinct and equispaced node points $x_0,\ x_1,\ \dots,\ x_{r-1},\ x_r,\ x_{r+1}, \dots,\ x_{n-1}, x_n$ such that $x_r = x_0 + rh, r = 0, 1, 2, \dots, n$ and h is the step length. The corresponding entries are given by $y_0 = f(x_0),\ y_1 = f(x_1),\ \dots, y_n = f(x_n)$.

Now, $x_{n-r} - x_n = x_0 + (n-r)h - (x_0 + nh)$

$= (-r)h$ .....(1)

Our objective is to find a polynomial $P(x)$ of degree less than or equal to n such that $P(x)$ replaces $f(x)$ on the set of node points x_r , $r = 0(1)n$ ie,

$P(x_r) = f(x_r)$, $r = 0(1)n$ .....(2)

Let us take the form of $P(x)$ as

$$P(x) = B_n + B_{n-1}(x - x_n) + B_{n-2}(x - x_n)(x - x_{n-1}) + B_{n-3}(x - x_n)(x - x_{n-1})$$

$(x - x_{n-2}) + \cdots + B_0(x - x_n)(x - x_{n-1}) \cdots (x - x_1)$ .....(3)

The constants $B_j, (j = 0(1)n)$ are to be determined by using equation (2).

Putting $x = x_n$ in equation (3) we get,

$$P(x_n) = B_n$$

or, $y_n = B_n$

Putting $x = x_{n-1}$ in equation (3) we get,

$$P(x_{n-1}) = B_n + B_{n-1}(x_{n-1} - x_n)$$

or, $y_{n-1} = y_n + B_{n-1}(-h)$

or, $B_{n-1} = \frac{\Delta y_{n-1}}{h}$.

Putting $x = x_{n-2}$in equation (3) we get,

$$P(x_{n-2}) = B_n + B_{n-1}(x_{n-2} - x_n) + B_{n-2}(x_{n-2} - x_n)(x_{n-2} - x_{n-1})$$

or, $y_{n-2} = y_n + \frac{\Delta y_{n-1}}{h}(-2h) + B_{n-2}(-2h)(-h)$

or, $B_{n-2} = \frac{\Delta^2 y_{n-2}}{2h^2}$

Putting $x = x_3$in equation (3) we get,

$$P(x_{n-3}) = B_n + B_{n-1}(x_{n-3} - x_n) + B_{n-2}(x_{n-3} - x_n)(x_{n-3} - x_{n-1}) + B_{n-3}(x_{n-3} - x_n)(x_{n-3} - x_{n-1})(x_{n-3} - x_{n-2})$$

or, $y_{n-3} = y_n + \left(\frac{\Delta y_{n-1}}{h}\right)(-3h) + \frac{\Delta^2 y_{n-2}}{2h^2}(-3h)(-2h) + B_{n-3}(-3h)(-2h)(-h)$

or, $B_{n-3} = \frac{\Delta^3 y_{n-3}}{3!h^3}$

Similarly, $A_{n-r} = \frac{\Delta^r y_{n-r}}{r!h^r}$.

Substituting $B_r{}'$ s value in (3) we have,

$$f(x) \approx P(x) = y_n + (x - x_n)\frac{\Delta y_{n-1}}{h} + (x - x_n)(x - x_{n-1})\frac{\Delta^2 y_{n-2}}{2!\,h^2} + (x - x_n)(x - x_{n-1})$$

$$(x - x_{n-2})\frac{\Delta^3 y_{n-3}}{3!\,h^3} + \cdots + (x - x_n)(x - x_{n-1})\cdots(x - x_0)\frac{\Delta^n y_0}{n!\,h^n}$$

This formula is known as Newton's Backward Interpolation Formula. It is used mainly in computing $f(x)$ when x lies in the end of the table.

Example: Find the value of $f(0.5)$using suitable interpolation formula from the following table:

x	0	1	2	3	4	5
$f(x)$	0	3	8	15	24	35

Ans:29.25

Lagrange's Interpolation Formula

Let a function $f(x)$ is known for $(n+1)$ distinct but not necessarily equispaced node points $x_0,\ x_1,\ \ldots,\ x_{r-1},\ x_r,\ x_{r+1}, \ldots,\ x_{n-1}, x_n$and the corresponding entries are given by $y_0 = f(x_0)$, $y_1 = f(x_1),\ \ldots, y_n = f(x_n)$. Then the Laplace's Interpolation formula is given by

$$L_n(x) = \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)}y_0 + \frac{(x-x_0)(x-x_2)\dots(x-x_n)}{(x_1-x_0)(x_1-x_2)\dots(x_1-x_n)}y_1 + \cdots + \frac{(x-x_0)(x-x_1)\dots(x-x_{n-1})}{(x_n-x_0)(x_n-x_1)\dots(x_n-x_{n-1})}y_n$$

Example: Find the value of $f(2)$using Lagrange's Interpolation from the following table:

x	0	1	3	4
$f(x)$	-12	0	6	12

Newton's Divided Difference Interpolation

Let a function $f(x)$ is known for $(n+1)$ distinct but not necessarily equispaced node points $x_0,\ x_1,\ \dots,\ x_{r-1},\ x_r,\ x_{r+1}, \dots,\ x_{n-1}, x_n$ and the corresponding entries are given by $y_0 = f(x_0),\ y_1 = f(x_1),\ \dots, y_n = f(x_n)$. Then the Newton's divided difference Interpolation formula is given by

$$f(x) = f(x_0) + (x-x_0)f[x_0,x_1] + (x-x_0)(x-x_1)f[x_0,x_1,x_2] + \cdots + (x-x_0)(x-x_1)\dots(x-x_{n-1})f[x_0,x_1,\dots,x_{n-1}] + R_{n+1}(x)$$

Where $R_{n+1}(x)$ is the remainder term.

Example: Find the value of f(2)using Newton's Divided Difference Interpolation formula from the following table:

x	-1	1	2	3
f(x)	-21	15	12	3

Numerical Integration

Numerical integration is the study of how the numerical value of an integral can be found. Also called quadrature, which refers to finding a square whose area is the same as the area under a curve, it is one of the classical topics of **numerical analysis**. Of central interest is the process of approximating a definite integral from values of the integrand when exact mathematical integration is not available.

Trapezoidal Rule

Let $I = \int_a^b f(x)dx$

The simplest quadrature rule in wide use is the Trapezoidal rule where the integrand is approximated by a linear polynomial. It is a two point formula ie, n (no. of interval)$= 1$. Therefore there are only two functional values $f(a) = y_0 = f(x_0)$ and $f(b) = y_1 = f(x_1)$ where $b - a = h$. Like many other methods, it has both a geometric and an analytic derivation. The idea of the geometric derivation is to approximate the area under the curve $y = f(x)$from $x = a$to $x = b$by the area of the trapezoid bounded by the points $(a, 0), (b, 0), [a, f(a)]$, and $[b, f(b)]$.This gives

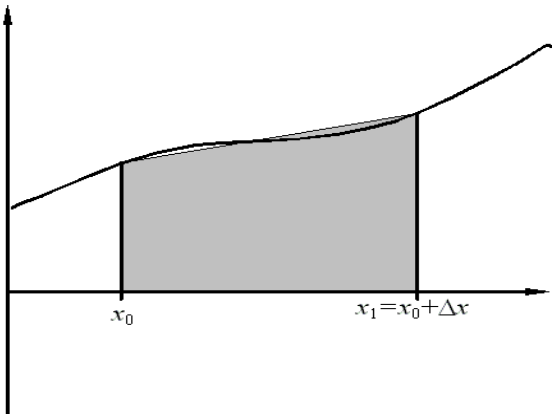
$$I_T = \int_a^b f(x)dx = \frac{h}{2}[f(a) + f(b)] = \frac{h}{2}[y_0 + y_1]$$

This formula is known as Trapezoidal rule for numerical integration. The error is given by

$E_T = -\frac{h^3}{12}f''(\xi)$where$a < \xi < b$.

$= -\frac{h}{12}[y_{-1} - 2y_0 + y_1]$.

Graphical Interpretation: In Trapezoidal rule the actual value of the integration is approximated by the area of the trapezium shown in the following figure.



Composite Trapezoidal Rule

If we divide the range of integration into n equal sub-intervals by $(n+1)$ points $a = x_0, x_1, x_2, \dots, x_n = b$ where $x_i = x_0 + ih$ $(i = 0(1)n)$; then if Trapezoidal rule is applied to each of the intervals $[x_0, x_0+h]$, $[x_0+h, x_0+2h]$, ... , $[x_0+\overline{n-1}h$, $x_0+nh]$then we get,

$$I_C^T = \int_a^b f(x)dx$$

$$= \int_{x_0}^{x_0+nh} f(x)dx$$

$$= \int_{x_0}^{x_0+h} f(x)dx + \int_{x_0+h}^{x_0+2h} f(x)dx + \cdots + \int_{x_0+\overline{n-1}h}^{x_0+nh} f(x)dx$$

$$= \frac{h}{2}[f(x_0)+f(x_0+h)] + \frac{h}{2}[f(x_0+h)+f(x_0+2h)] + \cdots + \frac{h}{2}[f(x_0+\overline{n-1}h)+f(x_0+nh)]$$

$$= \frac{h}{2}[\{f(x_0)+f(x_0+nh)\} + 2\{f(x_0+h)+f(x_0+2h)+\cdots+f(x_0+\overline{n-1}h)\}]$$

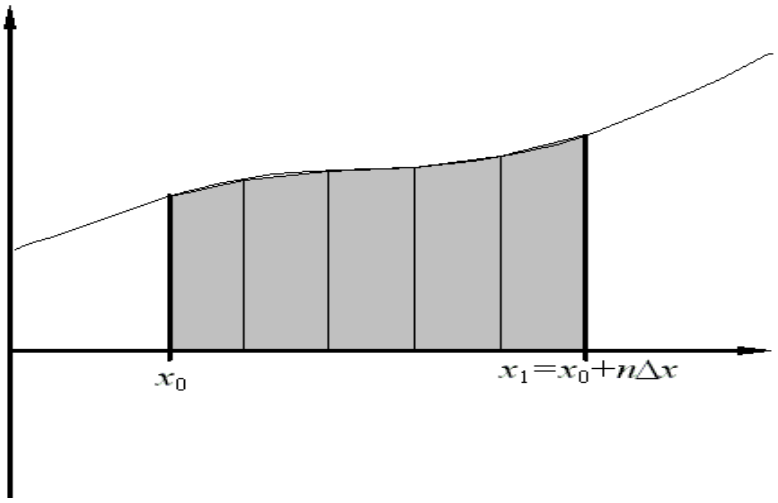
$$= \frac{h}{2}[(y_0+y_n) + 2(y_1+y_2+\cdots+y_{n-1})]$$

$= \frac{h}{2} \times$ [(Sum of first and last ordinates)$+2 \times$ (Sum of the all other ordinates)]

This formula is called Composite Trapezoidal rule for numerical integration. The error is given by

$$E_C^T = -\frac{h}{12}[y_{-1} + y_n - (y_0 + y_{n-1})] .$$

Graphical Interpretation: In Composite Trapezoidal rule the actual value of the integration is approximated by the sum of the area of n trapeziums shown in the following figure.



Simpson's $\frac{1}{3}rd$ Rule

Let $I = \int_a^b f(x)dx$

Simpson's 1/3 rule is an extension of Trapezoidal rule where the integrand is approximated by a second order polynomial.It is a three point formula ie, n (no. of interval)$= 2$. Therefore there are only three functional values $(a) = y_0 = f(x_0)$, $y_1 = f(x_1)$ and $f(b) = y_2 = f(x_2)$ where $h = \frac{b-a}{2}$ and $x_i = x_0 + ih$ $(i = 0(1)n)$. The Simpson's 1/3 Rule quadrature formula is given by

$$I_S = \int_a^b f(x)dx = \frac{h}{3}[f(x_0) + 4f(x_1) + f(x_2)]$$

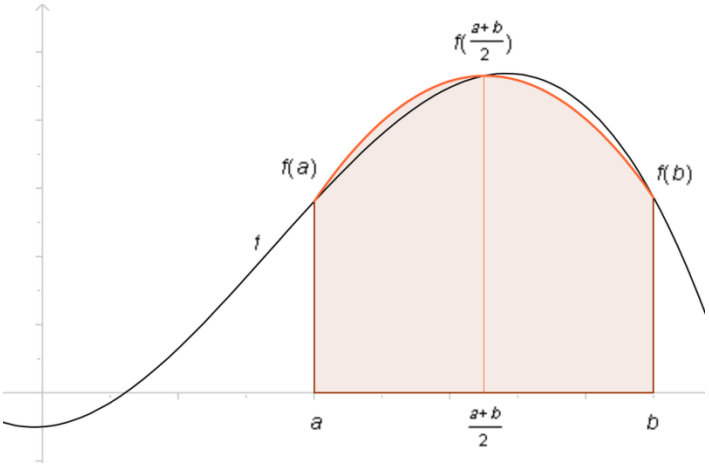
$$= \frac{h}{3}[y_0 + 4y_1 + y_2]$$

The error occurs in this formula is given by

$E_S = -\frac{h^5}{12}f^{IV}(\xi)$where$a <\xi< b$.

$= -\frac{h}{90}[y_{-1} - 4y_0 + 6y_1 - 4y_2 + y_3]$.

Graphical Interpretation: Simpson's $\frac{1}{3}rd$ rule approximates the actual value of the integration by the shaded area shown in the following figure.



Composite Simpson's $\frac{1}{3}rd$ Rule

If we divide the range of integration into $n = 2m$ (even) equal sub-intervals by $(n+1)$ points $a = x_0, x_1, x_2, \dots, x_n = b$ where $x_i = x_0 + ih\ (i = 0(1)n)$; then if Simpson's $\frac{1}{3}rd$ rule is applied to each of the intervals $[x_0, x_0 + 2h]$, $[x_0 + 2h, x_0 + 4h]$, ... , $[x_0 + \overline{n-2}h\ , x_0 + nh]$then we get,

$$I_C^S = \int_a^b f(x)dx$$

$$= \int_{x_0}^{x_0+nh} f(x)dx$$

$$= \int_{x_0}^{x_0+2h} f(x)dx + \int_{x_0+h}^{x_0+4h} f(x)dx + \cdots + \int_{x_0+\overline{n-2}h}^{x_0+nh} f(x)dx$$

$$= \frac{h}{3}[f(x_0) + 4f(x_0+h) + f(x_0+2h)] + \frac{h}{3}[f(x_0+2h) + 4f(x_0+3h) + f(x_0+4h)] + \cdots$$
$$+ \frac{h}{3}[f(x_0+\overline{n-2}h) + 4f(x_0+\overline{n-1}h) + f(x_0+nh)]$$

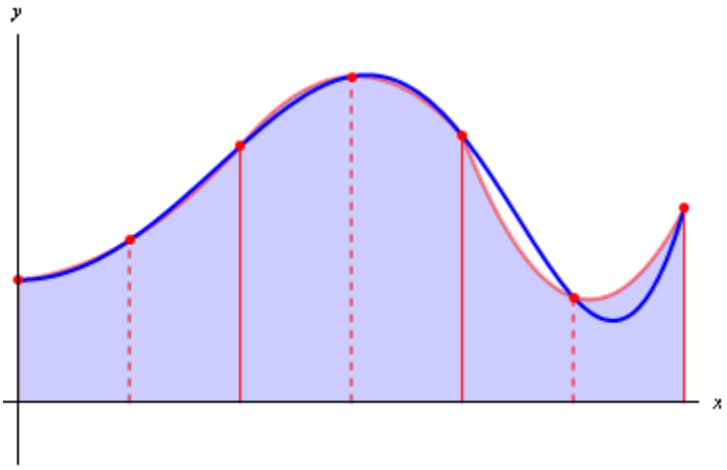
$$= \frac{h}{3}[(y_0 + y_n) + 4(y_1 + y_3 + \cdots + y_{n-1}) + 2(y_2 + y_4 + \cdots + y_{n-2})]$$

$= \frac{h}{3} \times$ [(Sum of first and last ordinates)$+4 \times$ (Sum of the all odd ordinates)$+2$(Sum of all even ordinates)]

This formula is called Composite Simpson's $\frac{1}{3}rd$ rule for numerical integration. The error is given by

$$E_C^S = -\frac{h}{90}[y_{-1} + y_5 - 4(y_0 + y_4) + 7(y_1 + y_3) - 8y_2].$$

Graphical Interpretation*:* Composite Simpson's $\frac{1}{3}rd$ rule approximates the actual value of the integration by the shaded (blue) region shown in the following figure.



Degree of Precision: A quadrature formula is said to have degree of precision $k(>0)$, if it is exact, ie, the error is zero for any arbitrary polynomial of degree less or equal to k, but there is at least one polynomial of degree $(k+1)$ for which it is not exact.

The degree of precision of Trapezoidal rule$= 1$.

The degree of precision of Simpson's $\frac{1}{3}rd$ rule$= 3$.

Problem1: Evaluate $\int_3^7 x^2 log x dx$ taking $n = 10$ using Trapezoidal rule.

Problem2: Evaluate $\int_0^1 x^2 e^x dx$ taking $n = 12$using Simpson's $\frac{1}{3}rd$ rule.

.

System of linear algebraic equations.

A general system of linear equations with n equations and n variables can be written in the form

$$\left.\begin{array}{l} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1,n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2,n}x_n = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3,n}x_n = b_3 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{n,1}x_1 + a_{n,2}x_2 + a_{n,3}x_3 + \cdots + a_{n,n}x_n = b_n \end{array}\right\} ----(1)$$

Where the coefficients a_{ij} $(i,j = 1,2,3,\cdots,n)$ and $b_i(i = 1,2,3,\cdots,n)$ are given constants.

This system of equations can be written inmatrix notation as $AX = b$, where

$A = [a_{ij}]_{n\times n} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{bmatrix}$ is the coefficient matrix , $b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \cdots \\ b_n \end{bmatrix}$ is called right hand side vector which are known and $X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_n \end{bmatrix}$ is called the solution vector to be determined.

Note: (i) If all the $b_i = 0, (i = 1,2,3,\cdots,n)$ then the system is called homogeneous otherwise the system is non-homogeneous.

(ii)If the diagonal elements of the coefficient matrix A satisfies the conditions $|a_{ii}| > \sum_{j=1,i\neq j}^{n} |a_{ij}|$, $i = 1,2,3,\cdots,n$ then the system will be called strictly diagonally dominant.

Gauss elimination method for system:

(i) It's a direct method of finding the solutions.

(ii) To apply this method we shall assume a square system, that is, number of equations must be equal to number of variables.

Let us consider the following system of n-equation in n- variables as

$$\left.\begin{array}{l} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1,n-1}^{(1)}x_{n-1} + a_{1,n}^{(1)}x_n = b_1^{(1)} \\ a_{21}^{(1)}x_1 + a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \cdots + a_{2,n-1}^{(1)}x_{n-1} + a_{2,n}^{(1)}x_n = b_2^{(1)} \\ a_{31}^{(1)}x_1 + a_{32}^{(1)}x_2 + a_{33}^{(1)}x_3 + \cdots + a_{3,n-1}^{(1)}x_{n-1} + a_{3,n}^{(1)}x_n = b_3^{(1)} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{n-1,1}^{(1)}x_1 + a_{n-1,2}^{(1)}x_2 + a_{n-1,3}^{(1)}x_3 + \cdots + a_{n-1,n-1}^{(1)}x_{n-1} + a_{n-1,n}^{(1)}x_n = b_{n-1}^{(1)} \\ a_{n,1}^{(1)}x_1 + a_{n,2}^{(1)}x_2 + a_{n,3}^{(1)}x_3 + \cdots + a_{n,n-1}^{(1)}x_{n-1} + a_{n,n}^{(1)}x_n = b_n^{(1)} \end{array}\right\} ----(2)$$

Where $a_{ij}^{(1)}, (i,j = 1,2,3,\cdots,n)$ and $b_i^{(1)}, (i = 1,2,3,\cdots,n)$ are prescribed constants. Let $a_{11}^{(1)} \neq 0$. Now multiplying the first equation successively by

$-\frac{a_{21}^{(1)}}{a_{11}^{(1)}}(= m_{21}), -\frac{a_{31}^{(1)}}{a_{11}^{(1)}}(= m_{31}), -\frac{a_{41}^{(1)}}{a_{11}^{(1)}}(= m_{41}), \cdots, -\frac{a_{n-1,1}^{(1)}}{a_{11}^{(1)}}(= m_{n-1,1}),\ -\frac{a_{n,1}^{(1)}}{a_{11}^{(1)}}(= m_{n,1})$ and adding respectively with 2nd ,3rd ,4th up to nth equation of the system we get

$$\left.\begin{array}{r} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1,n-1}^{(1)}x_{n-1} + a_{1,n}^{(1)}x_n = b_1^{(1)} \\ a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 + \cdots + a_{2,n-1}^{(2)}x_{n-1} + a_{2,n}^{(2)}x_n = b_2^{(2)} \\ a_{32}^{(2)}x_2 + a_{33}^{(2)}x_3 + \cdots + a_{3,n-1}^{(2)}x_{n-1} + a_{3,n}^{(2)}x_n = b_3^{(2)} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{n-1,2}^{(2)}x_2 + a_{n-1,3}^{(2)}x_3 + \cdots + a_{n-1,n-1}^{(2)}x_{n-1} + a_{n-1,n}^{(2)}x_n = b_{n-1}^{(2)} \\ a_{n,2}^{(2)}x_2 + a_{n,3}^{(2)}x_3 + \cdots + a_{n,n-1}^{(2)}x_{n-1} + a_{n,n}^{(2)}x_n = b_n^{(2)} \end{array}\right\} ----(3)$$

$$\text{Where,}\quad \begin{array}{l} a_{22}^{(2)} = a_{22}^{(1)} - \frac{a_{21}^{(1)}\cdot a_{12}^{(1)}}{a_{11}^{(1)}}, a_{23}^{(2)} = a_{23}^{(1)} - \frac{a_{21}^{(1)}\cdot a_{13}^{(1)}}{a_{11}^{(1)}}, \cdots\cdots\cdots \\ a_{32}^{(2)} = a_{32}^{(1)} - \frac{a_{31}^{(1)}\cdot a_{12}^{(1)}}{a_{11}^{(1)}}, a_{33}^{(2)} = a_{33}^{(1)} - \frac{a_{31}^{(1)}\cdot a_{13}^{(1)}}{a_{11}^{(1)}}, \cdots\cdots\cdots \\ a_{n,2}^{(2)} = a_{n,2}^{(1)} - \frac{a_{n,1}^{(1)}\cdot a_{12}^{(1)}}{a_{11}^{(1)}}, a_{n,3}^{(2)} = a_{n,3}^{(1)} - \frac{a_{n,1}^{(1)}\cdot a_{13}^{(1)}}{a_{11}^{(1)}}, \cdots\cdots\cdots \end{array}$$

Etc.

It is clear from the system (3) that except the first equation, the rest $(n-1)$

Equations are free from the unknown x_1. Again assuming $a_{22}^{(2)} \neq 0$, multiplying second equation of the system (3) successively by

$-\frac{a_{32}^{(2)}}{a_{22}^{(2)}}(= m_{32}), -\frac{a_{42}^{(2)}}{a_{22}^{(2)}}(= m_{42}), \cdots, -\frac{a_{n-1,2}^{(2)}}{a_{22}^{(2)}}(= m_{n-1,2}), -\frac{a_{n,2}^{(2)}}{a_{22}^{(2)}}(= m_{n,2})$ and adding respectively to 3rd, 4th, ..., $(n-1)$th and nth equation of the system (3) we get,

$$\left.\begin{array}{r} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1,n-1}^{(1)}x_{n-1} + a_{1,n}^{(1)}x_n = b_1^{(1)} \\ a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 + \cdots + a_{2,n-1}^{(2)}x_{n-1} + a_{2,n}^{(2)}x_n = b_2^{(2)} \\ a_{33}^{(3)}x_3 + \cdots + a_{3,n-1}^{(3)}x_{n-1} + a_{3,n}^{(3)}x_n = b_3^{(3)} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{n-1,3}^{(3)}x_3 + \cdots + a_{n-1,n-1}^{(3)}x_{n-1} + a_{n-1,n}^{(3)}x_n = b_{n-1}^{(3)} \\ a_{n,3}^{(3)}x_3 + \cdots + a_{n,n-1}^{(3)}x_{n-1} + a_{n,n}^{(3)}x_n = b_n^{(3)} \end{array}\right\} ----(4)$$

Here also we observe that the 3rd, 4th up to nth equations of the system (4) are free from unknowns x_1, x_2.

Repeating the same process of elimination of the unknowns, lastly we get a system of equations which is equivalent to the system (2) as:

$$\left.\begin{aligned} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1,n-1}^{(1)}x_{n-1} + a_{1,n}^{(1)}x_n &= b_1^{(1)} \\ a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 + \cdots + a_{2,n-1}^{(2)}x_{n-1} + a_{2,n}^{(2)}x_n &= b_2^{(2)} \\ a_{33}^{(3)}x_3 + \cdots + a_{3,n-1}^{(3)}x_{n-1} + a_{3,n}^{(3)}x_n &= b_3^{(3)} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ a_{n-1,n-1}^{(n-1)}x_{n-1} + a_{n-1,n}^{(n-1)}x_n &= b_{n-1}^{(n-1)} \\ a_{n,n}^{(n)}x_n &= b_n^{(n)} \end{aligned}\right\} ----(5)$$

The non-zero (by assumption) co-efficients $a_{11}^{(1)}, a_{22}^{(2)}, a_{33}^{(3)}, \cdots, a_{n,n}^{(n)}$ of the above set of equations are known as pivots and the corresponding equations are known as pivotal equations.

Now we can get easily the solution of system of equations (5) as follows. First we find x_n from the n-th equation, then x_{n-1} from the $(n-1)$-th equation substituting the value of x_n and then successively we get all he value of the unknowns $x_1, x_2, x_3, \cdots, x_n$. This process is known as back substitution.

Gauss Jacobi's Iteration Method:

Let us consider the following system of linear equations as follows

$$\left.\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1,n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2,n}x_n &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3,n}x_n &= b_3 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ a_{n,1}x_1 + a_{n,2}x_2 + a_{n,3}x_3 + \cdots + a_{n,n}x_n &= b_n \end{aligned}\right\} ----(6)$$

Where the coefficients a_{ij} $(i, j = 1,2,3, \cdots, n)$ and $b_i (i = 1,2,3, \cdots, n)$ are given constants.

Note: (i) Gauss Jacobi's method of iteration is an iterative method or indirect method, it is based on finding the successive better approximations of the unknowns of the system of equations, using iteration formula.

(ii)The convergence of iteration depends on the sufficient conditions that the system must be diagonally dominant , that is the coefficient matrix should be diagonally dominant that is $|a_{ii}| \geq \sum_{j=1, i\neq j}^{n} |a_{ij}|, \quad i = 1,2,3, \cdots, n.$

(iii)In this method also we shall find the solution of a square system.

Iteration formula:

The system (6) is diagonally dominant $(a_{ii} \neq 0, i = 1,2,3, \cdots, n)$ so it can be written as

$$\left.\begin{aligned}
x_1 &= \frac{1}{a_{11}}\left[b_1 - a_{12}x_2 - a_{13}x_3 - a_{14}x_4 - \cdots - a_{1,n-1}x_{n-1} - a_{1,n}x_n\right] \\
x_2 &= \frac{1}{a_{22}}\left[b_2 - a_{21}x_1 - a_{23}x_3 - a_{24}x_4 - \cdots - a_{2,n-1}x_{n-1} - a_{2,n}x_n\right] \\
x_3 &= \frac{1}{a_{33}}\left[b_3 - a_{31}x_1 - a_{32}x_2 - a_{34}x_4 - \cdots - a_{3,n-1}x_{n-1} - a_{3,n}x_n\right] \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
x_n &= \frac{1}{a_{n,n}}\left[b_n - a_{n,1}x_1 - a_{n,2}x_2 - a_{n,3}x_3 - \cdots - a_{n,n-2}x_{n-2} - a_{n,n-1}x_{n-1}\right]
\end{aligned}\right\} ----(7)$$

In this method, the iterations are generated by formulae known as iteration formulae as follows:

$$\left.\begin{aligned}
x_1^{(k+1)} &= \frac{1}{a_{11}}\left[b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - a_{14}x_4^{(k)} - \cdots - a_{1,n-1}x_{n-1}^{(k)} - a_{1,n}x_n^{(k)}\right] \\
x_2^{(k+1)} &= \frac{1}{a_{22}}\left[b_2 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - a_{24}x_4^{(k)} - \cdots - a_{2,n-1}x_{n-1}^{(k)} - a_{2,n}x_n^{(k)}\right] \\
x_3^{(k+1)} &= \frac{1}{a_{33}}\left[b_3 - a_{31}x_1^{(k)} - a_{32}x_2^{(k)} - a_{34}x_4^{(k)} - \cdots - a_{3,n-1}x_{n-1}^{(k)} - a_{3,n}x_n^{(k)}\right] \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
x_n^{(k+1)} &= \frac{1}{a_{n,n}}\left[b_n - a_{n,1}x_1^{(k)} - a_{n,2}x_2^{(k)} - a_{n,3}x_3^{(k)} - \cdots - a_{n,n-2}x_{n-2}^{(k)} - a_{n,n-1}x_{n-1}^{(k)}\right]
\end{aligned}\right\} ----(8)$$

Where initial guess $x_i^{(0)}$, $(i = 1,2,3,\cdots,n)$ being taken arbitrarily.

Here also the number of iterations $k$required depends up on the desired degree of accuracy.

If an error ε_s be tolerated in s th iteration, then the test for convergence is given by $\left|x_i^{(s+1)} - x_i^s\right| < \varepsilon_s$, for $k \geq s$.

Gauss-Seidel Iteration Method:

This method is also an indirect method for finding solution of a system of linear equations. This method is almost identical with Gauss Jacobi's method, except in considering the iteration formula. The sufficient condition for convergence of Gauss Seidel method is that the system of equations must be strictly diagonally dominant. That is the coefficient matrix $A = \left[a_{ij}\right]_{n\times n}$ be such that $|a_{ii}| > \sum_{j=1,i\neq j}^{n} |a_{ij}|, \quad i = 1,2,3,\cdots,n.$

We consider a system of strictly diagonally dominant equations as:

$$\left.\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1,n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2,n}x_n &= b_2 \\
a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3,n}x_n &= b_3 \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
a_{n,1}x_1 + a_{n,2}x_2 + a_{n,3}x_3 + \cdots + a_{n,n}x_n &= b_n
\end{aligned}\right\} ----(9)$$

As the system is diagonally dominant therefore $a_{ii} \neq 0, (i = 1,2,3,\cdots,n)$. Like Gauss Jacobi's method, the system of equation can be written in the form

$$\left.\begin{aligned}
x_1 &= \frac{1}{a_{11}}\left[b_1 - a_{12}x_2 - a_{13}x_3 - a_{14}x_4 - \cdots - a_{1,n-1}x_{n-1} - a_{1,n}x_n\right] \\
x_2 &= \frac{1}{a_{22}}\left[b_2 - a_{21}x_1 - a_{23}x_3 - a_{24}x_4 - \cdots - a_{2,n-1}x_{n-1} - a_{2,n}x_n\right] \\
x_3 &= \frac{1}{a_{33}}\left[b_3 - a_{31}x_1 - a_{32}x_2 - a_{34}x_4 - \cdots - a_{3,n-1}x_{n-1} - a_{3,n}x_n\right] \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
x_{n-1} &= \frac{1}{a_{n-1,n-1}}\left[b_{n-1} - a_{n-1,1}x_1 - a_{n-1,2}x_2 - a_{n-1,3}x_3 - \cdots - a_{n-1,n-2}x_{n-2} - a_{n-1,n}x_n\right] \\
x_n &= \frac{1}{a_{n,n}}\left[b_n - a_{n,1}x_1 - a_{n,2}x_2 - a_{n,3}x_3 - \cdots - a_{n,n-2}x_{n-2} - a_{n,n-1}x_{n-1}\right]
\end{aligned}\right\} - (10)$$

Now after considering an initial guess, $x_1 = x_1^{(0)}, x_2 = x_2^{(0)}, x_3 = x_3^{(0)}, \cdots x_n = x_n^{(0)}$ (usually the initial values of the unknowns are taken to be $x_i^0 = 0, i = 1,2,3,\cdots,n$), The successive iteration scheme called iteration formula of Gauss-Seidel method , are as follows:

$$\left.\begin{aligned}
x_1^{(k+1)} &= \frac{1}{a_{11}}\left[b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - a_{14}x_4^{(k)} - \cdots - a_{1,n-1}x_{n-1}^{(k)} - a_{1,n}x_n^{(k)}\right] \\
x_2^{(k+1)} &= \frac{1}{a_{22}}\left[b_2 - a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - a_{24}x_4^{(k)} - \cdots - a_{2,n-1}x_{n-1}^{(k)} - a_{2,n}x_n^{(k)}\right] \\
x_3^{(k+1)} &= \frac{1}{a_{33}}\left[b_3 - a_{31}x_1^{(k+1)} - a_{32}x_2^{(k+1)} - a_{34}x_4^{(k)} - \cdots - a_{3,n-1}x_{n-1}^{(k)} - a_{3,n}x_n^{(k)}\right] \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
x_n^{(k+1)} &= \frac{1}{a_{n,n}}\left[b_n - a_{n,1}x_1^{(k+1)} - a_{n,2}x_2^{(k+1)} - a_{n,3}x_3^{(k+1)} - \cdots - a_{n,n-2}x_{n-2}^{(k+1)} - a_{n,n-1}x_{n-1}^{(k+1)}\right]
\end{aligned}\right\}$$
$$- (11)$$

the number of iterations $k$required depends up on the desired degree of accuracy.

If an error ε_s be tolerated in s th iteration, then the test for convergence is given by $\left|x_i^{(s+1)} - x_i^s\right| < \varepsilon_s$, for $k \geq s$.

Matrix Inversion Method:

Let us consider a system of n-linear equations with n-unknown as :

$$AX = b - - - -(12)$$

where

$$A = \left[a_{ij}\right]_{n\times n} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \cdots \\ b_n \end{bmatrix} \text{ and } X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_n \end{bmatrix}.$$

Multiplying the system of equations (12) by the inverse of the matrix A, A^{-1} we get $X = A^{-1}b$, provided A^{-1} exists that is $|A| \neq 0$, that is A is non singular Matrix.

In general A^{-1} is defined as $A^{-1} = \frac{Adj(A)}{Det(A)}$.

But finding the inverse as well as solution of the system of equation by this method is a tremendous task. Thus, to find in easier method we have Gauss Jordan's Matrix Inversion method.

Gauss Jordan's Matrix Inversion method.

In this method we shall find the inverse of a matrix without calculating the determinant.

In this method we shall write the augmented matrix of a quare matrix *A* by writing a unit matrix *I* of same order as that of *A* side by side. Then we shall transfer the matrix *A* to a unit matrix by number of steps equal to order of the matrix and then the matrix so obtained to which the unit matrix is transferred is the inverse of the matrix *A*.

Without the loss of generality let us consider a 4×4 matrix *A* of the following form:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}, \quad |A| \neq 0$$

Now the Augmented matrix of A is formed by an unit matrix *I* as

$$[A:I] \sim \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & \vdots & 1 & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & \vdots & 0 & 1 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & a_{34} & \vdots & 0 & 0 & 1 & 0 \\ a_{41} & a_{42} & a_{43} & a_{44} & \vdots & 0 & 0 & 0 & 1 \end{bmatrix}$$

Now dividing the first row (pivot row) by a_{11}(pivot element) then multiplying successively by a_{21}, a_{31}, a_{41} and subtracting from 2nd , 3rd and 4th row we get

$$\sim \begin{bmatrix} 1 & \vdots & a'_{11} & a'_{12} & a'_{13} & a'_{14} & \vdots & 0 & 0 & 0 \\ 0 & \vdots & a'_{21} & a'_{22} & a'_{23} & a'_{24} & \vdots & 1 & 0 & 0 \\ 0 & \vdots & a'_{31} & a'_{32} & a'_{33} & a'_{34} & \vdots & 0 & 1 & 0 \\ 0 & \vdots & a'_{41} & a'_{42} & a'_{43} & a'_{44} & \vdots & 0 & 0 & 1 \end{bmatrix}$$

Where

$$a'_{11} = \frac{a_{12}}{a_{11}}, a'_{12} = \frac{a_{13}}{a_{11}}, a'_{13} = \frac{a_{14}}{a_{11}}, a'_{14} = \frac{1}{a_{11}}$$

$$a'_{21} = a_{22} - a_{21}\frac{a_{12}}{a_{11}}, a'_{22} = a_{23} - a_{21}\frac{a_{13}}{a_{11}}, a'_{23} = a_{24} - a_{21}\frac{a_{14}}{a_{11}}, a'_{24} = 0 - a_{21}\frac{1}{a_{11}}$$

$$a'_{31} = a_{32} - a_{31}\frac{a_{12}}{a_{11}}, a'_{32} = a_{33} - a_{31}\frac{a_{13}}{a_{11}}, a'_{33} = a_{34} - a_{31}\frac{a_{14}}{a_{11}}, a'_{34} = 0 - a_{31}\frac{1}{a_{11}}$$

$$a'_{41} = a_{42} - a_{41}\frac{a_{12}}{a_{11}}, a'_{42} = a_{43} - a_{41}\frac{a_{13}}{a_{11}}, a'_{43} = a_{44} - a_{41}\frac{a_{14}}{a_{11}}, a'_{24} = 0 - a_{41}\frac{1}{a_{11}}$$

Then we divide the second row by a'_{21} and then multiplying successively by $a'_{11}, a'_{31}, a'_{41}$ and subtracting from 1st 3rd and 4th row we get

$$\sim\begin{bmatrix} 1 & 0 & \vdots & a''_{11} & a''_{12} & a''_{13} & a''_{14} & \vdots & 0 & 0 \\ 0 & 1 & \vdots & a''_{21} & a''_{22} & a''_{23} & a''_{24} & \vdots & 0 & 0 \\ 0 & 0 & \vdots & a''_{31} & a''_{32} & a''_{33} & a''_{34} & \vdots & 1 & 0 \\ 0 & 0 & \vdots & a''_{41} & a''_{42} & a''_{43} & a''_{44} & \vdots & 0 & 1 \end{bmatrix}$$

Where,

$$a''_{11} = a'_{12} - a'_{11}\frac{a'_{22}}{a'_{21}}, a''_{12} = a'_{13} - a'_{11}\frac{a'_{23}}{a'_{21}}, a''_{13} = a'_{14} - a'_{11}\frac{a'_{24}}{a'_{21}}, a''_{14} = 0 - a'_{11}\frac{1}{a'_{21}}$$

$$a''_{21} = \frac{a'_{22}}{a'_{21}}, a''_{22} = \frac{a'_{23}}{a'_{21}}, a''_{23} = \frac{a'_{24}}{a'_{21}}, a''_{24} = \frac{1}{a'_{21}}$$

$$a''_{31} = a'_{32} - a'_{31}\frac{a'_{22}}{a'_{21}}, a''_{32} = a'_{33} - a'_{31}\frac{a'_{23}}{a'_{21}}, a''_{33} = a'_{34} - a'_{31}\frac{a'_{24}}{a'_{21}}, a''_{34} = 0 - a'_{31}\frac{1}{a'_{21}}$$

$$a''_{41} = a'_{42} - a'_{41}\frac{a'_{22}}{a'_{21}}, a''_{42} = a'_{43} - a'_{41}\frac{a'_{23}}{a'_{21}}, a''_{43} = a'_{44} - a'_{41}\frac{a'_{24}}{a'_{21}}, a''_{44} = 0 - a'_{41}\frac{1}{a'_{21}}$$

Again dividing the thirs row by a''_{31}, then multiplying the 3rd,row by $a''_{11}, a''_{21}, a''_{41}$ successively and then subtracting from 1st 2nd and 4th row we get,

$$\sim\begin{bmatrix} 1 & 0 & 0 & \vdots & a'''_{11} & a'''_{12} & a'''_{13} & a'''_{14} & \vdots & 0 \\ 0 & 1 & 0 & \vdots & a'''_{21} & a'''_{22} & a'''_{23} & a'''_{24} & \vdots & 0 \\ 0 & 0 & 1 & \vdots & a'''_{31} & a'''_{32} & a'''_{33} & a'''_{34} & \vdots & 0 \\ 0 & 0 & 0 & \vdots & a'''_{41} & a'''_{42} & a'''_{43} & a'''_{44} & \vdots & 1 \end{bmatrix}$$

$$a'''_{11} = a''_{12} - a''_{11}\frac{a''_{32}}{a''_{31}}, a'''_{12} = a''_{13} - a''_{11}\frac{a''_{33}}{a''_{31}}, a'''_{13} = a''_{14} - a''_{11}\frac{a''_{34}}{a''_{31}}, a'''_{14} = 0 - a''_{11}\frac{1}{a''_{31}}$$

$$a'''_{21} = a''_{22} - a''_{21}\frac{a''_{32}}{a''_{31}}, a'''_{22} = a''_{23} - a''_{21}\frac{a''_{33}}{a''_{31}}, a'''_{23} = a''_{24} - a''_{21}\frac{a''_{34}}{a''_{31}}, a'''_{24} = 0 - a''_{21}\frac{1}{a''_{31}}$$

$$a'''_{31} = \frac{a''_{32}}{a''_{31}}, a'''_{32} = \frac{a''_{33}}{a''_{31}}, a'''_{33} = \frac{a''_{34}}{a''_{31}}, a'''_{34} = \frac{1}{a''_{31}}$$

$$a'''_{41} = a''_{42} - a''_{41}\frac{a''_{32}}{a''_{31}}, a'''_{42} = a''_{43} - a''_{41}\frac{a''_{33}}{a''_{31}}, a'''_{43} = a''_{44} - a''_{41}\frac{a''_{34}}{a''_{31}}, a'''_{44} = 0 - a''_{41}\frac{1}{a''_{31}}$$

Finally We divide the forth row by a'''_{41}, then multiplying successively by $a'''_{11}, a'''_{21}, a'''_{31}$ and then subtracting from 1st, 2nd, 3rd row, we get

$$\sim\begin{bmatrix}1 & 0 & 0 & 0 & \vdots & a_{11}'''' & a_{12}'''' & a_{13}'''' & a_{14}''''\\ 0 & 1 & 0 & 0 & \vdots & a_{21}'''' & a_{22}'''' & a_{23}'''' & a_{24}''''\\ 0 & 0 & 1 & 0 & \vdots & a_{31}'''' & a_{32}'''' & a_{33}'''' & a_{34}''''\\ 0 & 0 & 0 & 1 & \vdots & a_{41}'''' & a_{42}'''' & a_{43}'''' & a_{44}''''\end{bmatrix}$$

Where

$$a_{11}'''' = a'''_{12} - a'''_{11}\frac{a'''_{42}}{a'''_{41}},\ a_{12}'''' = a'''_{13} - a'''_{11}\frac{a'''_{43}}{a'''_{41}}, a_{13}'''' = a'''_{14} - a'''_{11}\frac{a'''_{44}}{a'''_{41}},\ a_{14}''''$$
$$= 0 - a'''_{11}\frac{1}{a'''_{41}}$$

$$a_{21}'''' = a'''_{22} - a'''_{21}\frac{a'''_{42}}{a'''_{41}}, a_{22}'''' = a'''_{23} - a'''_{21}\frac{a'''_{43}}{a'''_{41}}, a_{23}'''' = a'''_{24} - a'''_{21}\frac{a'''_{44}}{a'''_{41}}, a_{24}''''$$
$$= 0 - a'''_{21}\frac{1}{a'''_{41}}$$

$$a_{31}'''' = a'''_{32} - a'''_{31}\frac{a'''_{42}}{a'_{41}}, a_{32}'''' = a'''_{33} - a'''_{31}\frac{a'''_{43}}{a'''_{41}}, a_{33}'''' = a'''_{34} - a'''_{31}\frac{a'''_{44}}{a'''_{41}}, a_{34}''''$$
$$= 0 - a'''_{31}\frac{1}{a'''_{41}}$$

$a_{41}'''' = \frac{a'''_{42}}{a'''_{41}}, a_{42}'''' = \frac{a'''_{43}}{a'''_{41}}, a_{43}'''' = \frac{a'''_{44}}{a'''_{41}}, a_{44}'''' = \frac{1}{a'''_{41}}$.

Thus the required matrix ,

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

Is the required Inverse of the matrix

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

Matrix Factorization or LU Factorization Method:

Matrix factorization method is a powerful and minimum labour method to compute the unknowns in a system of linear equations. In this method we shall express the coefficient matrix of a system of linear equation as product to two square matrix of same order, one is upper triangular and another is lower triangular matrix.

Let us consider a system of linear equation be of the form

$$AX = b - -(1)$$

where A is the coefficient matrix, b is the column vector with known constants and the column vector with unknowns is X.

Let U and L be upper and lower triangular matrices such that

$$A = LU - - - (2).$$

Therefore we have

$$LUX = b - - - (3).$$

Let us set

$$UX = Y - - - (4)$$

where Y is again column vector of unknows.

Now the system reduces to

$$LY = b - - - - (5)$$

Now by forward substitution we can solve equation (5) for Y.

Then by back substitution we can solve equation (4) for X.

For example let us consider a system of 3 equations with 3 unknowns in the following form:

$$\left.\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &= b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3 \end{aligned}\right\} - - - (6)$$

Here the coefficient matrix A is given by

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

And the column vector of unknowns $X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$, and the column vector of constant $b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$

So the system is $AX = b$

Now let us consider the lower triangular matrix L and upper triangular matrix U of order 3×3 of the form

$$L = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \text{ and } U = \begin{bmatrix} 1 & u_{12} & u_{13} \\ 0 & 1 & u_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

So, if $A = LU$, then

$$LU = \begin{bmatrix} l_{11} & l_{11}u_{12} & l_{11}u_{13} \\ l_{21} & l_{21}u_{12} + l_{22} & l_{21}u_{13} + l_{22}u_{23} \\ l_{31} & l_{31}u_{12} + l_{32} & l_{31}u_{13} + l_{32}u_{23} + l_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} - - - (8)$$

From (8) we can easily find the values of l_{ij} and u_{ij} for $i, j = 1,2,3$.

Now the given system can be written as

$$LUX = b - - - (9)$$

Let us set

$$UX = Y - - - (10)$$

where $Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$, y_1, y_2, y_3 are unknowns.

Now the system reduces to

$$LY = b - - - - (11)$$

That is

$$\begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \times \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

That is

$$\left.\begin{aligned} l_{11}y_1 &= b_1 \\ l_{21}y_1 + l_{22}y_2 &= b_2 \\ l_{31}y_1 + l_{31}y_2 + l_{33}y_2 &= b_3 \end{aligned}\right\} - - - - (12)$$

By forward substitution we can easily solve the system (12) for y_1, y_2, y_3.

Now equation (10) gives

$$\begin{bmatrix} 1 & u_{12} & u_{13} \\ 0 & 1 & u_{23} \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

That is

$$\left.\begin{aligned} x_1 + u_{12}x_2 + u_{13}x_3 &= y_1 \\ x_2 + u_{23}x_3 &= y_2 \\ x_3 &= y_3 \end{aligned}\right\} - - - (13)$$

Now by back substitution we can find the unknowns x_1, x_2, x_3 from equation (13).

Numerical Solution of Ordinary Differential Equations:

Let us consider a 1st order ordinary differential equation as initial value problem as follows:

$$\frac{dy}{dx} = f(x,y), y(x_0) = y_0 ----(1)$$

Our objective is to solve the given initial value problem. If $f(x,y)$ has only a very simple form, then we can find the analytical solution for the given differential equations. But there are enormous collection of problems which arise in real life situation (engineering, science, technology or industry) which has complicated form or the solution of the differential equation is very laborious. In these cases, the numerical methods give at least an approximate value of y at some prescribed value of x.

Euler Method:

It's a simple and single step but crude numerical method for solving an ordinary initial value differential equation, where the solution will be obtained as a set of tabulated values of variables x and y.

Let us consider the a first order 1st degree differential equation as (1). Let us suppose that we have to find the approximate value of y say y_n when $x = x_n$. We divide the range $[x_0, x_n]$ into n-equal sub intervals by the points $x_0, x_1, x_2, \cdots, x_{r-1}, x_r, x_{r+1}, \cdots, x_n$, where $x_r = x_0 + rh$, $(r = 1,2,3, \cdots, n)$ and $h = x_r - x_{r-1}$=step length.

Assuming $f(x,y) \simeq f(x_{r-1}, y_{r-1})$ in $(x_{r-1} \leq x \leq x_r)$ and integrating (1) in the range $[x_{r-1}, x_r]$ we get the Euler iteration formula as:

$\int_{x_{r-1}}^{x_r} dy = \int_{x_{r-1}}^{x_r} f(x,y)dx$,

or, $y_r = y_{r-1} + \int_{x_{r-1}}^{x_r} f(x,y)dx$, $----(2)$

or, $y_r \simeq y_{r-1} + f(x_{r-1}, y_{r-1}) \int_{x_{r-1}}^{x_r} dx = y_{r-1} + h\, f(x_{r-1}, y_{r-1}), ---- (3)$

Using formula (3) at the mesh point $x_r (r = 1,2,3, \cdots, n)$, we get, the successive approximations to y as follows:

$$\left.\begin{array}{ll} y_1 = y_0 + hf(x_0, y_0) & = y(x_1) \\ y_2 = y_1 + hf(x_1, y_1) & = y(x_2) \\ y_3 = y_2 + hf(x_2, y_2) & = y(x_3) \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ y_n = y_{n-1} + hf(x_{n-1}, y_{n-1}) & = y(x_n) \end{array}\right\} ----(4)$$

Note: This method depends the step length h and smaller the length h yield a better approximate result. So the method is too tedious to get result up to desired degree of accuracy.

Modified Euler Method:

This method gives us a rapid and moderately accurate result upto desired degree of accuracy. Starting with initial value $y(x_r) = y_r$ an approximate value of $y_r^{(0)}$can be computed by Euler formula (3).

$$y_r^{(0)} = y_{r-1} + \int_{x_{r-1}}^{x_r} f(x,y)dx \simeq y_{r-1} + h\, f(x_{r-1}, y_{r-1}) - - - -(5)$$

Where $f(x,y)$is replaced by $f(x_{r-1}, y_{r-1})$ in $[x_{r-1} \leq x \leq x_r]$.

Instead of considering $f(x,y) \simeq f(x_{r-1}, y_{r-1})$, if we use the Trapezoidal rule in the range $[x_{r-1} \leq x \leq x_r]$, we get ,

$$y_r^{(0)} = y_{r-1} + \frac{h}{2}\,[f(x_{r-1}, y_{r-1}) + f(x_r, y_r)] - - - -(6)$$

No replacing $f(x_r, y_r)$ by its moderate approximate value $f\left(x_r, y_r^{(0)}\right)$ at the end point of the interval $[x_{r-1}, x_r]$, we get, the first approximations to $y_r = y(x_r)$ as

$$y_r^{(1)} = y_{r-1} + \frac{h}{2}\left[f(x_{r-1}, y_{r-1}) + f\left(x_r, y_r^{(0)}\right)\right] - - - -(7)$$

In this manner, considering successive approximations to y_r, we get the iteration formula as

$$y_r^{(n)} = y_{r-1} + \frac{h}{2}\left[f(x_{r-1}, y_{r-1}) + f\left(x_r, y_r^{(n-1)}\right)\right] - - - -(8)$$

Where $y_r^{(n)}$ is the n-th approximation to y_r. Thus $y_r^{(n)} \simeq y_r$.

Note: In any two consecutive approximation of y_r, say, $y_r^{(k-1)}$ and $y_r^{(k)}$ if $\left|y_r^{(k)} - y_r^{(k-1)}\right| < \varepsilon$ where ε is the error of precession, then we conclude $y_r^{(k-1)} \simeq y_r^{(k)} \simeq y_r$.

Taylor's Series Method:

Its also a simple and useful numerical method for solving an ordinary differential equation given in equation

$$y'(x) = \frac{dy}{dx} = f(x,y), y(x_0) = y_0 - - - -(9)$$

Where $f(x,y)$ is simple in nature and is sufficiently differentiable with respect to x and y.

Taking the step length h, sufficiently small, the exact solution of (9), $y(x)$ can be expanded by Taylor's series about x_0 as:

$$y(x) \simeq y(x_0 + h) = y(x_0) + hy'^{(0)} + \frac{h^2}{2\,!}y''^{(x_0)} + \frac{h^3}{3\,!}y'''^{(x_0)} + \frac{h^4}{4\,!}y^{iv}(x_0) + \cdots ----(10)$$

The values of the derivatives required in the equation (10) can be computed by taking successive differentiations of the implicit function $f(x, y)$as follows:

$$\left.\begin{aligned} y'(x) &= f(x, y) \\ y''(x) &= f_x + f_y.y'(x) \\ y'''(x) &= f_{xx} + f_{xy}.y'(x) + \{f_{xy} + f_{yy}.y'(x)\}y'(x) + f_y.y''(x) \\ &= f_{xx} + 2f_{xy}.y'(x) + f_{yy}\{y'(x)\}^2 + f_y.y''(x)\ , \end{aligned}\right\} ----(11)$$

(Assuming $f_{xy} = f_{yx}$)

So on and then substituting x_0 for x and the corresponding values of derivatives for x_0.

Note: To get the solution $y(x)$ up to a desired degree of accuracy $y(x_{r+1}), (r = 1,2,3, \cdots)$ through the sequence of values of x, $x_r = x_{r-1} + h, (r = 1,2,3, \cdots)$, we compute $y(x_r)$ for $x_r = x_{r-1} + h$ first and the values of the derivative from (11) for x_r , then we compute $y(x_{r+1})$ fro $x_{r+1} = x_r + h$.

R-K/Runge-Kutta Method:

Second Order Runge Kutta Method:

Second order Runge Kutta method is based on the Taylor's Series Method. To derive the computational formula, Let us consider a differential equation as

$$y'(x) = \frac{dy}{dx} = f(x, y), y(x_0) = y_0 ----(12)$$

Now taking the derivatives of y we get

$$\left.\begin{aligned} y'' &= f_x + f_y.y' = f_x + f.f_y \\ y''' &= f_{xx} + f_{xy}.y' + (f_x + f_y.y')f_y + f[f_{yx} + f_{yy}.y'] \\ &= f_{xx} + f_{xy}.f + f_x.f_y + f_y^2.f + f.f_{yx} + f_{yy}.f^2 \\ &= f_{xx} + 2f.f_{xy} + f_x.f_y + f_y^2.f + f_{yy}.f^2 \end{aligned}\right\} ---(13)$$

Again, by Taylor's Series, we have

$$\begin{aligned} y_1 = y(x_0 + h) &= y(x_0) + hy'(x_0) + \frac{h^2}{2\,!}y''(x_0) + O(h^3) \\ &= (y)_0 + h(y')_0 + \frac{h^2}{2\,!}(y'')_0 + O(h^3) \\ &= y_0 + h(f)_0 + \frac{h^2}{2\,!}[f_x + f.f_y]_0 + O(h^3) \\ &= y_0 + h(f)_0 + \frac{h^2}{2\,!}\left[(f_x)_0 + (f)_0.(f_y)_0\right] + O(h^3) ---(14) \end{aligned}$$

Where the scripts '0' denotes the values of the functions at (x_0, y_0).

In this method, The solution of (12) is taken, with the step length h (small) as

$$y_1 = y(x_0 + h) = y_0 + k ----(15)$$

Where $\left.\begin{matrix} k = \alpha k_1 + \beta k_2 \\ k_1 = h\, f(x_0, y_0) = h(f)_0 \\ k_2 = h\,[f(x_0 + mh, y_0 + nk_1)] \end{matrix}\right\} ---(16)$

Where α, β, m, n are constants and are evaluated such that (16) agrees Taylor's Series (14) upto including term containing h^2.

Again from (16) we have

$$\begin{aligned} k_2 &= h\left[f(x_0, y_0) + \{mhf_x + nk_1 f_y\}_{(x_0, y_0)}\right] \\ &= h\left[(f)_0 + mh(f_x)_0 + n.h\,(f)_0.(f_y)_0\right] \\ &= h(f)_0 + h^2\left\{m(f_x)_0 + n\,(f)_0.(f_y)_0\right\} \end{aligned}$$

Substituting in (15) we have

$$\begin{aligned} y_1 = y(x_0 + h) &= y_0 + \alpha.h(f)_0 + \beta.h(f)_0 + h^2\beta\left\{m(f_x)_0 + n\,(f)_0.(f_y)_0\right\} \\ &= y_0 + (\alpha + \beta)h(f)_0 + h^2\left\{m\beta(f_x)_0 + n\,\beta(f)_0.(f_y)_0\right\} ----(17) \end{aligned}$$

Now comparing (17) with (14) we get

$$\alpha + \beta = 1, m\beta = \frac{1}{2}, n\beta = \frac{1}{2},$$

$$\text{So, } m = n$$

Now taking $m = n = 1$ we get $\alpha = \beta = \frac{1}{2}$.

Thus the computational formulafor Runge-Kutta method of order 2 reduces to

$$\left.\begin{matrix} y_1 = y(x_0 + h) = y_0 + k \\ k = \frac{1}{2}(k_1 + k_2) \\ k_1 = hf(x_0, y_0) \\ k_2 = hf(x_0 + h, y_0 + k) \end{matrix}\right\} ---(18)$$

The error in this formula is $O(h^3)$.

Fourth Order Runge Kutta Method:

The computational formula for fourth order Runge-Kutta method can be derived in similar manner as in second order by considering terms up to h^4, as follows:

$$\left.\begin{aligned} y_1 &= y(x_0+h) = y_0 + k \\ k &= \frac{1}{6}(k_1 + 2k_2 + 2\,k_3 + k_4) \\ k_1 &= h\,f(x_0, y_0) \\ k_2 &= h\,f\left(x_0 + \frac{h}{2}, y_0 + \frac{k_1}{2}\right) \\ k_3 &= h\,f\left(x_0 + \frac{h}{2}, y_0 + \frac{k_2}{2}\right) \\ k_4 &= h\,f(x_0 + h, y_0 + k_3) \end{aligned}\right\} --- (19)$$

Where the error is $O(h^5)$.

Milne's Predictor-Corrector Method:

It is a multi step method , that is to compute y_{n+1} a knowledge of preceding values of y and y' is essentially required. These values of y to be computed by any one of the self starting method like Taylor's series method, Euler Method, Runge-Kutta Method,W.E. Milne uses two types of quadrature formula (i) open type formula to derive the Predictor formula and (ii) Closed-type quadrature formula to derive corrector formula.

The Predictor Formula is given by: $\bar{y}_{n+1} = y_{n-3} + \frac{4h}{3}[2y'_{n-2} - y_{n-1} + 2\,y'_n]$

The corrector Formula is given by: $y_{n+1} = y_{n-1} + \frac{h}{3}[y'_{n-1} + 4y'_n + y'_{n+1}]$

Computational Procedure:

Step I: Compute y'_{n-2}, y'_{n-1}, y'_n by the given differential equation $y'_r = f(x_r, y_r)$.

Step II: Compute $\bar{y}_{n+1}$ by the predictor formula

Step III: Compute $y'_{n+1}$by the given differential equation, by using the predicted value $\bar{y}_{n+1}$ obtained in Step II.

Step IV: Using Predicted value y'_{n+1}obtained in Step III, compute y_{n+1} by the corrector formula.

Step V: Compute $D_{n+1} = corrected\ value\ (y_{n+1}) - Predicted\ value\ (\bar{y}_{n+1})$. If D_{n+1} is very small then proceed for the next interval and D_{n+1} is not sufficiently small, then reduce, the value of h by taking its half etc.

Solution of Equations

Algebraic and Transcendental Equations

f(x) = 0 is called an algebraic equation if the corresponding f (x) is a polynomial. An example is$7x^2 + 2x + 1 = 0$. f(x) is called transcendental equation if the f (x) contains trigonometric, or exponential or logarithmic functions Examples of transcendental equations are sin x – x =0

There are two types of methods available to find the roots of algebraic and transcendental equations of the form f (x) = 0.

1. Direct Methods: Direct methods give the exact value of the roots in a finite number of steps. We assume here that there are no round off errors. Direct methods determine all the roots at the same time.

2. Indirect or Iterative Methods: Indirect or iterative methods are based on the concept of successive approximations. The general procedure is to start with one or more initial approximation to the root and obtain a sequence of iterates k x which in the limit converges to the actual or true solution to the root. Indirect or iterative methods determine one or two roots at a time. The indirect or iterative methods are further divided into two categories: bracketing and open methods. The bracketing methods require the limits between which the root lies, whereas the open methods require the initial estimation of the solution. Bisection and False position methods are two known examples of the bracketing methods. Among the open methods, the Newton-Raphson is most commonly used. The most popular method for solving a non-linear equation is the Newton-Raphson method and this method has a high rate of convergence to a solution.

Methods such as the bisection method and the false position method of finding roots of a nonlinear equation $f(x) = 0$ require bracketing of the root by two guesses. Such methods are called *bracketing methods*. These methods are always convergent since they are based on reducing the interval between the two guesses so as to zero in on the root of the equation.

In the Newton-Raphson method, the root is not bracketed. In fact, only one initial guess of the root is needed to get the iterative process started to find the root of an equation. The method hence falls in the category of *open methods*. Convergence in open methods is not guaranteed but if the method does converge, it does so much faster than the bracketing methods.

What is the bisection method and what is it based on?

One of the first numerical methods developed to find the root of a nonlinear equation $f(x) = 0$ was the bisection method (also called *binary-search* method). The method is based on the following theorem.

Theorem

An equation $f(x) = 0$, where $f(x)$ is a real continuous function, has at least one root between x_ℓ and x_u if $f(x_\ell) f(x_u) < 0$ (See Figure 1).

Note that if $f(x_\ell) f(x_u) > 0$, there may or may not be any root between x_ℓ and x_u (Figures 2 and 3). If $f(x_\ell) f(x_u) < 0$, then there may be more than one root between x_ℓ and x_u (Figure 4).

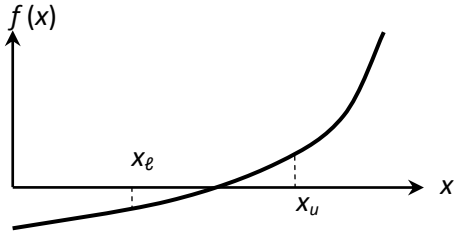


Figure 1 At least one root exists between the two points if the function is real, continuous, and changes sign.

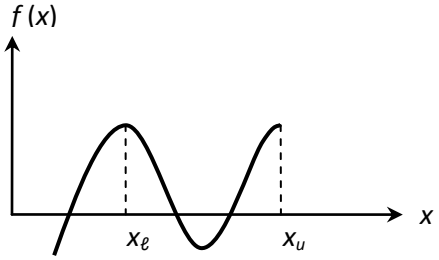


Figure 2 If the function $f(x)$ does not change sign between the two points, roots of the equation $f(x) = 0$ may still exist between the two points.

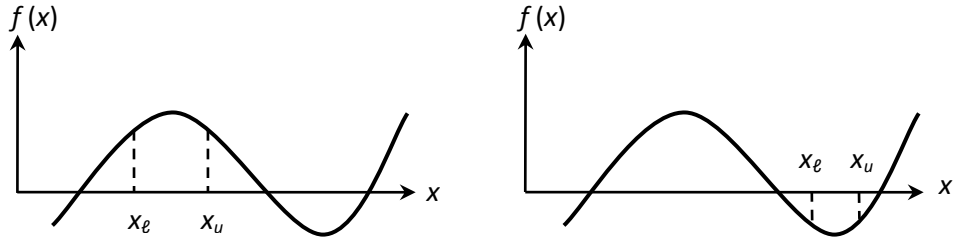


Figure 3 If the function $f(x)$ does not change sign between two points, there may not be any roots for the equation $f(x) = 0$ between the two points.

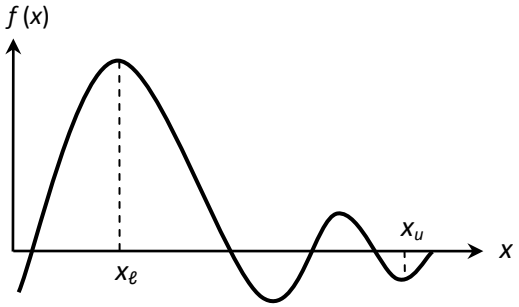


Figure 4 If the function $f(x)$ changes sign between the two points, more than one root for the equation $f(x) = 0$ may exist between the two points.

Since the method is based on finding the root between two points, the method falls under the category of bracketing methods.

Since the root is bracketed between two points, x_ℓ and x_u, one can find the mid-point, x_m between x_ℓ and x_u. This gives us two new intervals

1. x_ℓ and x_m, and
2. x_m and x_u.

Is the root now between x_ℓ and x_m or between x_m and x_u? Well, one can find the sign of $f(x_\ell) f(x_m)$, and if $f(x_\ell) f(x_m) < 0$ then the new bracket is between x_ℓ and x_m, otherwise, it is between x_m and x_u. So, you can see that you are literally halving the interval. As one repeats this process, the width of the interval $[x_\ell, x_u]$ becomes smaller and smaller, and you can zero in to the root of the equation $f(x) = 0$. The algorithm for the bisection method is given as follows.

Algorithm for the bisection method

The steps to apply the bisection method to find the root of the equation $f(x) = 0$ are

1. Choose x_ℓ and x_u as two guesses for the root such that $f(x_\ell) f(x_u) < 0$, or in other words, $f(x)$ changes sign between x_ℓ and x_u.
2. Estimate the root, x_m, of the equation $f(x) = 0$ as the mid-point between x_ℓ and x_u as

$$x_m = \frac{x_\ell + x_u}{2}$$

3. Now check the following

a) If $f(x_\ell) f(x_m) < 0$, then the root lies between x_ℓ and x_m; then $x_\ell = x_\ell$ and $x_u = x_m$.

b) If $f(x_\ell) f(x_m) > 0$, then the root lies between x_m and x_u; then $x_\ell = x_m$ and $x_u = x_u$.

c) If $f(x_\ell) f(x_m) = 0$; then the root is x_m. Stop the algorithm if this is true.

4. Find the new estimate of the root

$$x_m = \frac{x_\ell + x_u}{2}$$

5. Repeat the process until the difference of the new estimated root and the previous root is negligible.

Advantages of Bisection Method

a) The bisection method is always convergent. Since the method brackets the root, the method is guaranteed to converge.

b) As iterations are conducted, the interval gets halved. So one can guarantee the decrease in the error in the solution of the equation.

Drawbacks of Bisection Method

a) The convergence of bisection method is slow as it is simply based on halving the interval.

b) If one of the initial guesses is closer to the root, it will take larger number of iterations to reach the root.

c) If a function $f(x)$ is such that it just touches the x-axis (Figure 3.8) such as $f(x) = x^2 = 0$ it will be unable to find the lower guess, x_ℓ, and upper guess, x_u, such that $f(x_\ell) f(x_u) < 0$

False position method

The false position method uses this property:
A straight line joins $f(x_i)$ and $f(x_u)$. The intersection of this line with the x-axis represents an improvement estimate of the root. This new root can be computed as:

$$\frac{f(x_l)}{x_r - x_l} = \frac{f(x_u)}{x_r - x_u}$$

$$\Rightarrow x_r = x_u - \frac{f(x_u)(x_l - x_u)}{f(x_l) - f(x_u)}$$ This is called the false-position formula

Then, x_r replaces the initial guess for which the function value has the same sign as $f(x_r)$

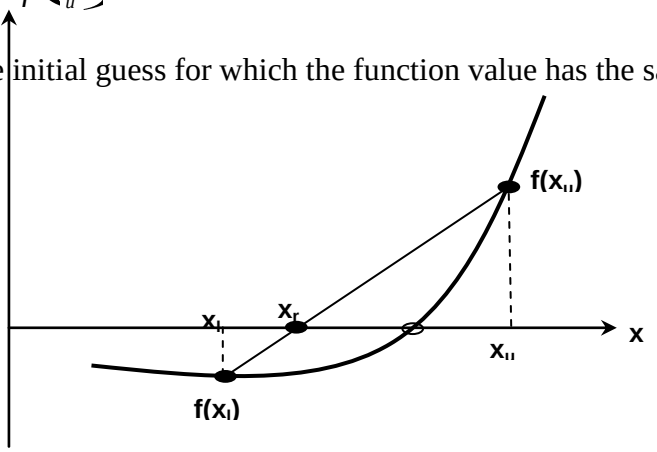


Figure 5. False-position method.

Although, the false position method is an improvement of the bisection method. In some cases, the bisection method will converge faster and yields to better results (see Figure.5).

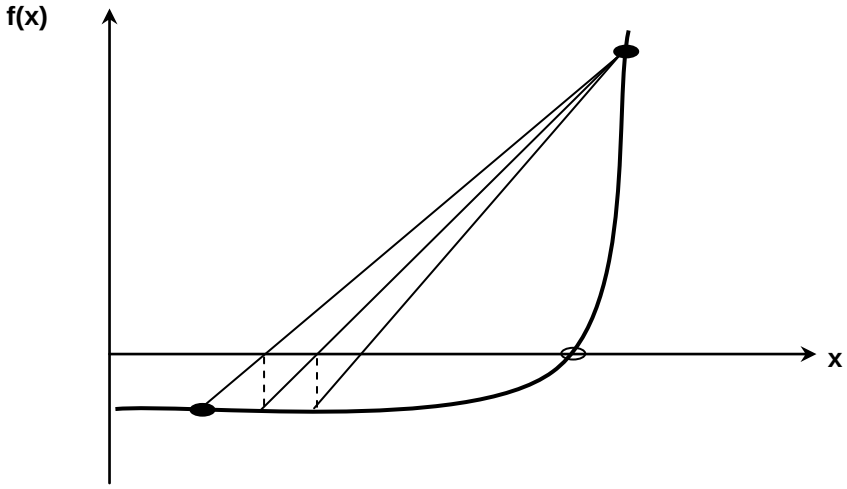


Figure 6. Slow convergence of the false-position method.

Newton-Raphson method

Newton-Raphson method is based on the principle that if the initial guess of the root of f(x)=0 is at x_i, then if one draws the tangent to the curve at $f(x_i)$, the point x_{i+1} where the tangent crosses the x-axis is an improved estimate of the root (Figure 3.12).

Using the definition of the slope of a function, at $x = x_i$

$$f'(x_i) = = \frac{f(x_i) - 0}{x_i - x_{i+1}} \quad \Rightarrow x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

This equation is called the Newton-Raphson formula for solving nonlinear equations of the form $f(x) = 0$ So starting with an initial guess, x_i, one can find the next guess, x_{i+1}, by using the above equation. One can repeat this process until one finds the root within a desirable tolerance.

Algorithm

The steps to apply using Newton-Raphson method to find the root of an equation f(x) = 0 are

1. Evaluate f′(x) symbolically
2. Use an initial guess of the root, x_i, to estimate the new value of the root x_{i+1} as

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

3. Repeat the process until the difference of the new estimated root and the previous root is negligible.