NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Prof. Jayati Sarkar (course co-ordinator)

Feedback

- Email (jayati@chemical.iitd.ac.in)

Books

Text books:

- S.C. Chapra & R.P. Canale, "Numerical Methods for Engineers with Personal Computer Applications", McGraw Hill Book Company, 1985.
- R.L. Burden & J. D. Faires, "Numerical Analysis".

Reference books:

- Atkinson, K.E., "An Introduction to Numerical Analysis", John Wiley & Sons, 1978.
- Gupta, S. K., "Numerical Methods for Engineers, New Academic Science, 2012.
- W. H. et al., "Numerical Recipes in C: The Art of Scientific Computing, 3rd Edition, Cambridge University Press, 2007.

Journals: https://www.aps.org, https://www.sciencedirect.com/

- International Journal for Numerical Methods in Engineering
- Physics of Fluid
- International Journal of Numerical Methods for Heat & Fluid Flow
- Journal of Non-Newtonian Fluid Mechanics

Policies

- Lectures will be carried out in asynchrono mode and ppts with video lectures will be uploaded in Impartus in Moodle.

- After every 5^{th} lecture the lecture on the stipulated lecture day will be converted into a live –session, where part of it can be converted into lectures, part of it as help-sessions to clear doubt and/or part to conduct quizzes.

- Do not use your mobile phones in live-sessions.

Practicals

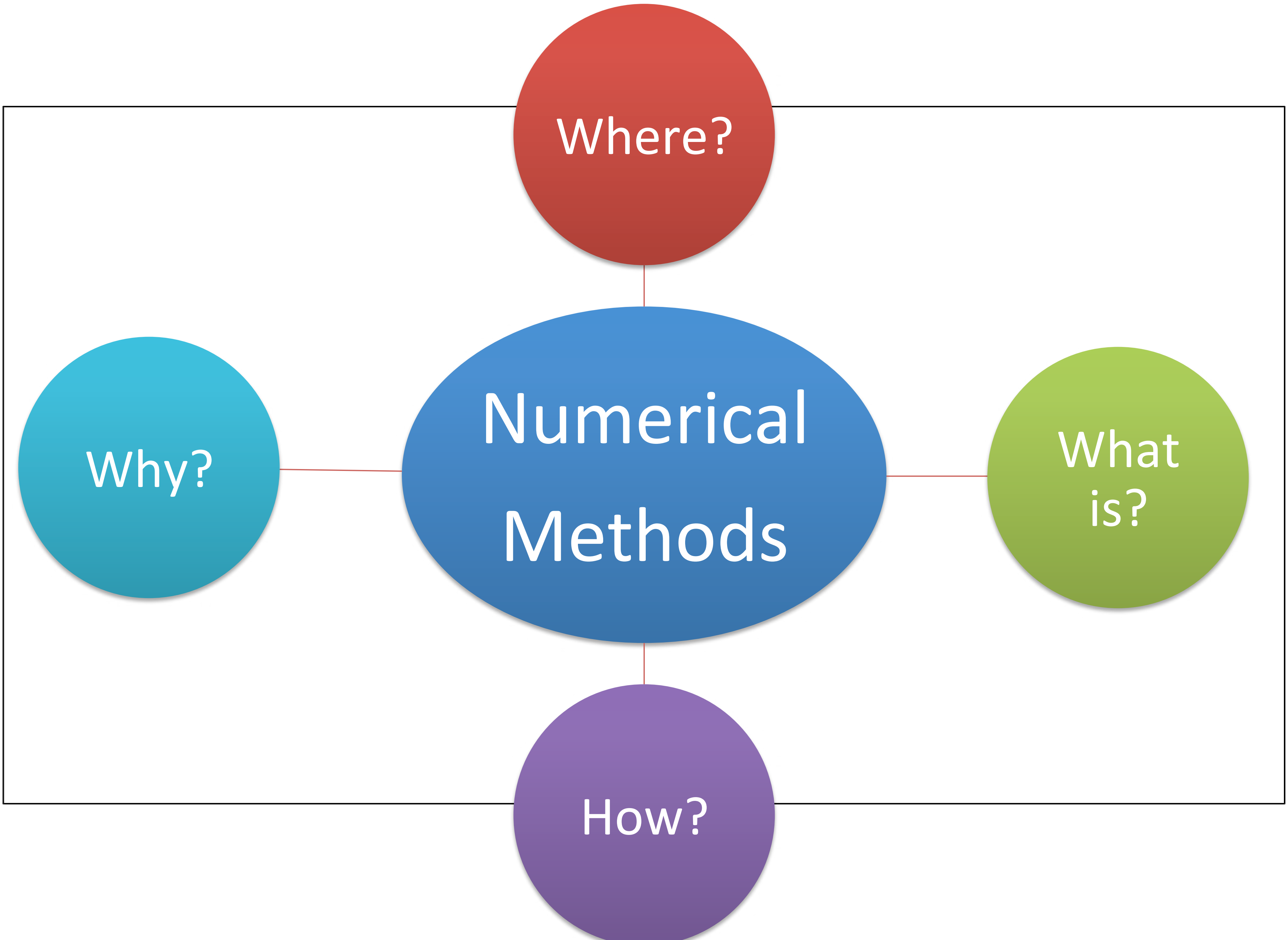
- Practical starts from next week.
- Days of practical
- Reminder: Please bring your calculators to the practicals?
- Practical sheets will be uploaded on Moodle.
- Codes/Reports to be submitted on Moodle.
- All announcements related to the course will be via course mailing lists
- For coding C/C++ language will be only admissible.

Marking scheme

- Quiz – 10 %
- Minor – 20 %
- Major – 40 %
- Term Paper-15 %
- Practical Submissions-15 %
- All exams will be closed book

Course Outline

1. INTRODUCTION,APPROXIMATION AND ROUND OFF ERRORS
2 LINEAR ALGEBRAIC EQUATIONS
3. ROOT FINDING AND SOLUTION OF NON-LINEAR EQUATIONS
4. FUNCTIONS, INTERPOLATION, APPROXIMATION, REGRESSION
5. NUMERICAL INTEGRATION AND DIFFERENTIATION
6. ORDINARY DIFFERENTIAL EQUATIONS
7. PARTIAL DIFFERENTIAL EQUATIONS

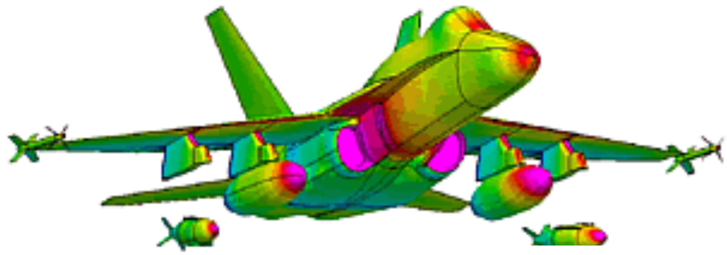


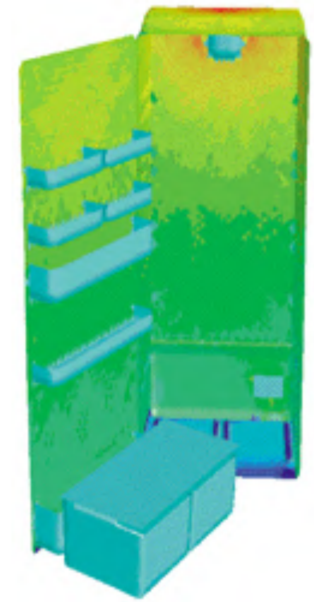
Where?
Why?
Numerical Methods
What is?
How?

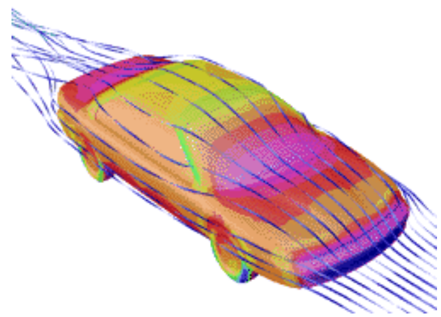
Where do we need numerical methods

In Engineering

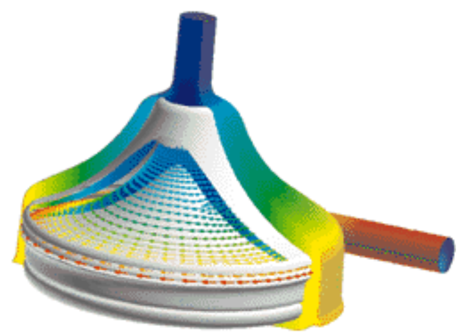
Civil, Mechanical, chemical in all engineering problems

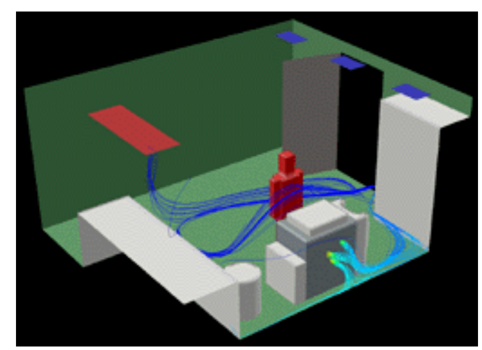






Numerical methods play a very important role. Dedicated software like **ANSYS/FLUENT/ABACUS** are used to solve computer aided design/ computational fluid dynamics problem.





Where do we need numerical methods

- **In life sciences and medicines**

  Differential modeling of cancer network

  Travelling wave analysis of tumor-immune interaction with immunotherapy

  Population dynamics determining algae growth. Therapy Planning and 3-D imaging

- **Social science**

  Correlating different aspects of growth and economy

- **Business and finance**

  Predicting stock market and stock trading

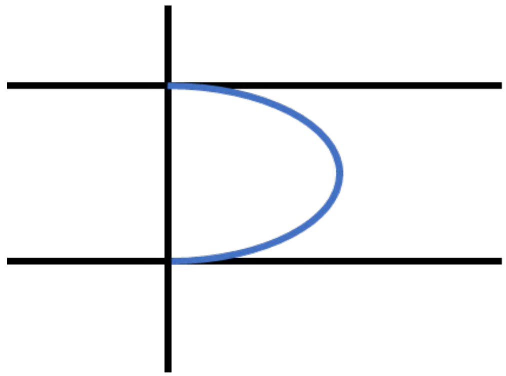
Why do we need numerical methods

Why?

$$\int f(x)dx$$

$f(x)$ need to be cast in particular form to be sorted or else no analyzed/ closed solution

Real life problems have closed solution only under extreme simplifications



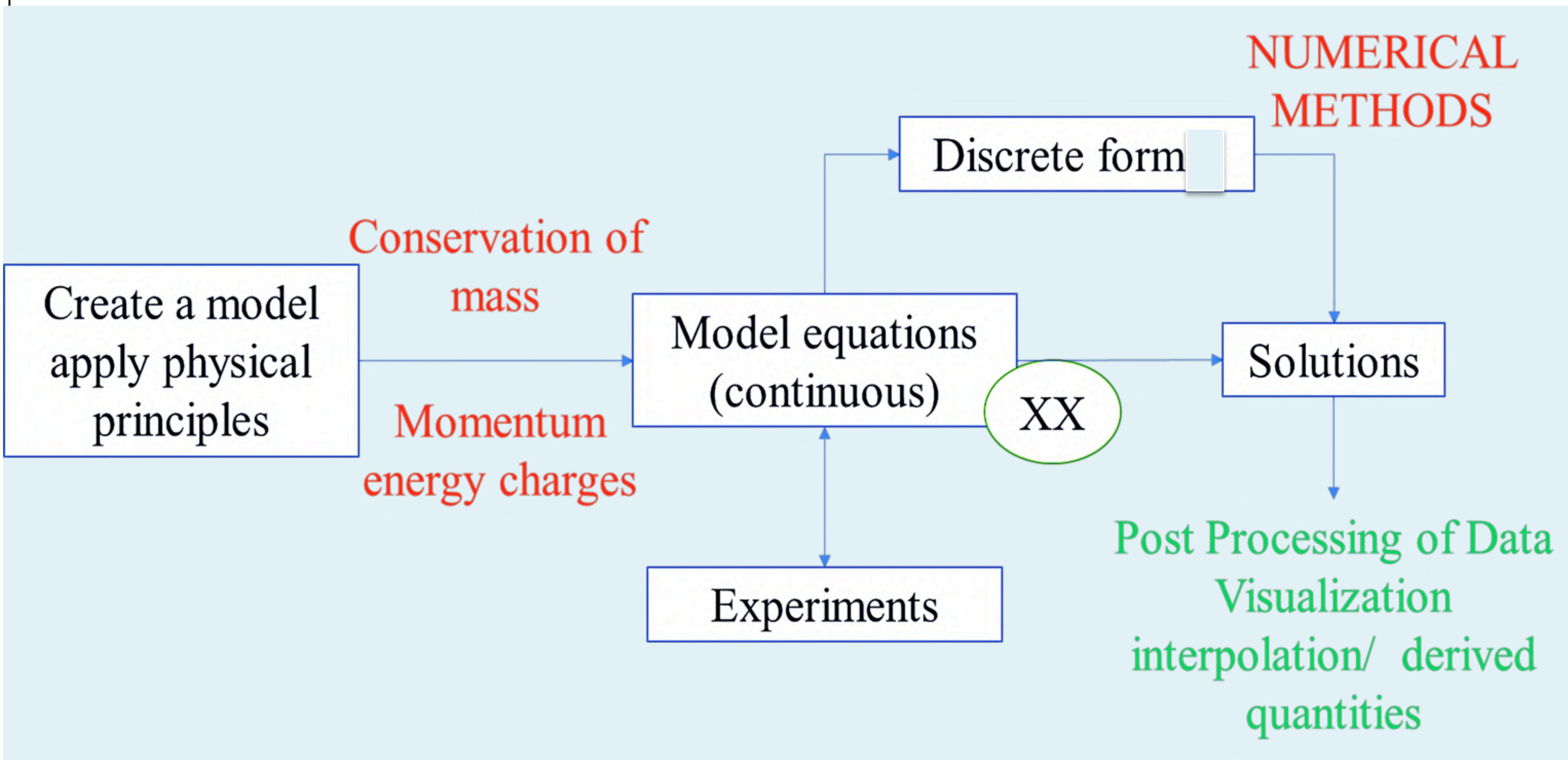
Newtonian
isothermal

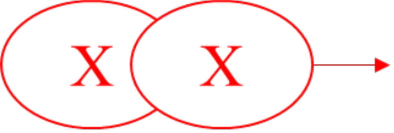
Steady State
irrotational

General Navier Stokes eg. →Are Coupled 2nd order PDES→No analyzed solution

Numerical Methods/ Computational techniques

Use computers to solve problems by step-wise, repeated and iterative solution methods, which otherwise would be tedious or unsolvable by hand calculations.





ALGEBRIC EQUATIONS

$$a_{11}x_1 + a_{12}x_2 = b_1$$

$$a_{21}x_1 + a_{22}x_2 = b_2$$

(linear if $a_{ij} = const, nonliner\ if\ a_{ij} = a_{ij}\ (x_i)$)

ODE (IVP/BVP)

$$\frac{dy}{dx} = f(x, y)$$

PDE

$$Au_{x,x} + 2Bu_{x,y} + C_{y,y} + \cdots \ldots \ldots = 0::::::::\ D = B^2 - AC$$

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2} \ldots\ldots we\ have\ eq^n.\ (D > 0 :: hyperbolic\ PDE)$$

$$\frac{\partial u}{\partial t} = k \frac{\partial^2 u}{\partial x^2} \ldots \ldots (D = 0 :: parabolic\ PDE)$$

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y) \ldots (D < 0 :: Elliptic\ eq^n)$$

Example of Iterative Method

Henon Approximation $\sqrt{S}$

(BABYLONIAN METHOD ~1750BC)

$$\boldsymbol{x^{[i+1]} = \frac{1}{2}\left(x^{[i]} + \frac{S}{x^{[i]}}\right)}$$

$$f(x) = x^{[i]} - S = 0$$

$$x^{[i+1]} = x^{[i]} - \frac{f\left(x^{[i]}\right)}{f'(x^{[i]})} \quad (Newton\ Raphson\ Method)$$

$$= x^{[i]} - \frac{\left(x^{[i]}\right)^2 - S}{2x^{[i]}} = \frac{\left(x^{[i]}\right)^2 + S}{2x^{[i]}} = \frac{1}{2}(x^{[i]} + \frac{S}{x^{[i]}})$$

Example of Iterative Method

S=3: Finding value of $\sqrt{3}$

Iteration Number $i = 0$ $x^{[0]}$=0.1

$$x^{[1]} = \frac{1}{2}\left(x^{[0]} + \frac{3}{x^{[0]}}\right) = \frac{1}{2}\left(0.1 + \frac{3}{0.1}\right) = 0.5\times(30.1) = 15.05$$

Iteration Number $i = 1$ $x^{[1]}$=15.05

$$x^{[2]} = \frac{1}{2}\left(x^{[1]} + \frac{3}{x^{[1]}}\right) = \frac{1}{2}\left(15.05 + \frac{3}{15.05}\right) = 0.5\times(30.1) = 7.6247$$

Iteration Number $i = 2$ $x^{[2]}$= 7.6247

$$x^{[3]} = \frac{1}{2}\left(x^{[2]} + \frac{3}{x^{[2]}}\right) = \frac{1}{2}\left(7.6247 + \frac{3}{7.62475}\right) = 0.5\times(8.018) = 4.009$$

Iterative approximate Henon Approximation $\sqrt{S}$ ::::::$x^{[i+1]} = \frac{1}{2}\left(x^{[i]} + \frac{S}{x^{[i]}}\right)$

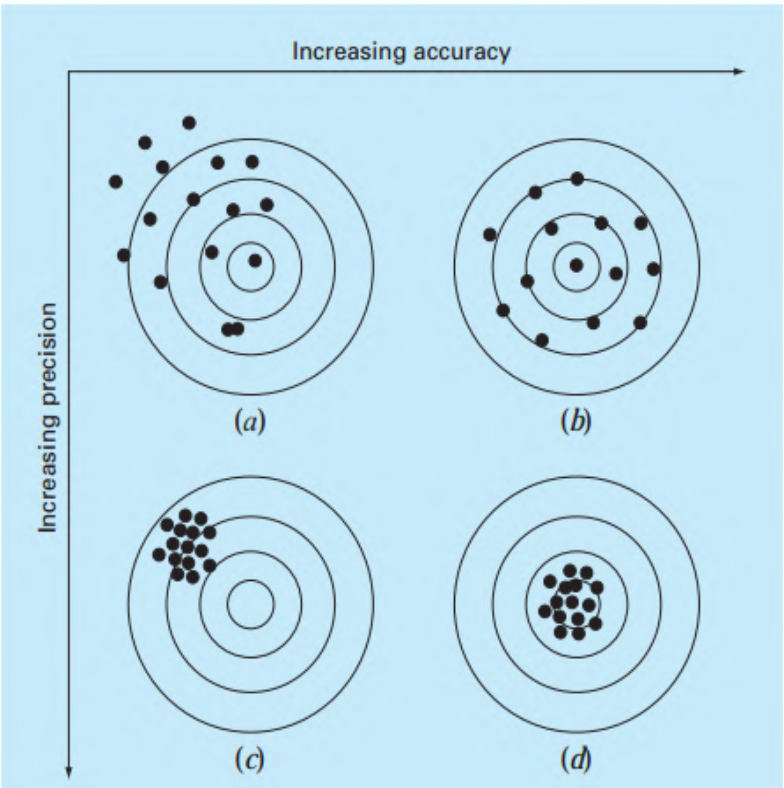
$\sqrt{3} = 1.732050808$

$[i]$	$x^{[i]}$	$3/x^{[i]}$	$x^{[i+1]}$=0.5*($x^{[i]} + 3/x^{[i]}$)
0	**0.1**	30	**15.08**
1	**15.05**	0.199	**7.6247**
2	**7.6247**	0.393	**4.009**
3	**4.009**	0.748	**2.3787**
4	**2.3787**	1.2612	**1.81997**
5	**1.8199**	1.6484	**1.73205**
6	**1.7341**	1.72993	**1.73205**
7	**1.73205**	1.732049	**1.73205**
8	**1.73205**	1.73205	**1.73205**
9	**1.73205**	1.73205	**1.73205**

Accuracy and Precision

Accuracy: refers to how closely a computer or measured value agrees with true value

Precision: refers to how closely individual computed or measured values are to each other



(a) & (c) though tightly grouped are both inaccurate or biased systematic deviation from truth.

(b) & (d) are equally accurate (centered around bull's eye), the latter is more precise because the shots are tightly grouped.

Numerical methods should be sufficiently accurate or unbiased and should be accurately precise.

Error Definitions

Numerical errors arise from the use of approximation to represent exact mathematical operations and quantities

Truncation error

which results when approximations are used to represent exact mathematical procedures

Round off errors

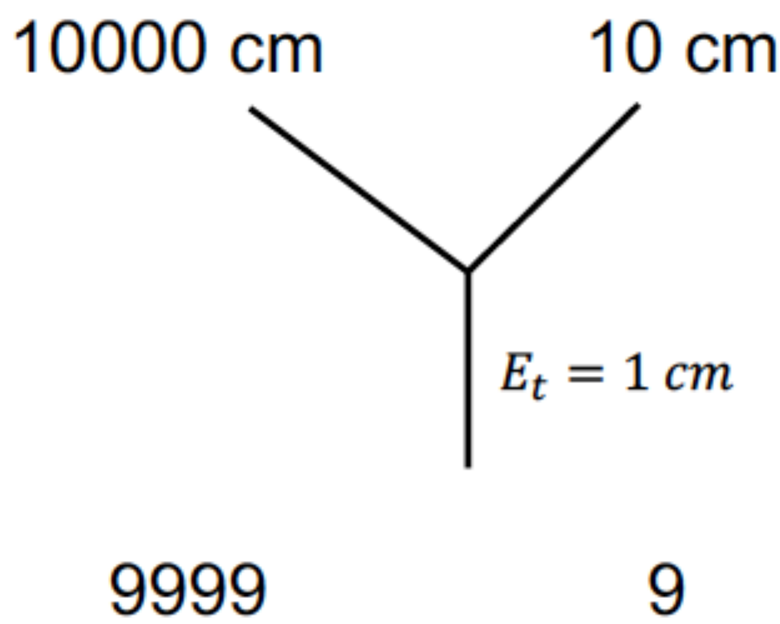
which results when numbers having limited significant figures are used to represent exact numbers

True Value= Approximation + Error

E_T = True Value – Approximation

Shortcomings

True Value - It takes no account of the order of magnitude of the value under consideration



Relative error (normalize the error with the true value)

$$\varepsilon_t = \frac{true\ error}{true\ value} \times 100\%$$

$$\boxed{\begin{array}{ll} \varepsilon_t = \dfrac{1}{10000} \times 100\% & = \dfrac{1}{10} \times 100\% \\ = 0.01\% & = 10\% \end{array}}$$

.

Shortcomings

Approximation - True or analytical values are rarely available

Error can be **normalized** by using the best available estimate of the true value to the approximation itself

$$\varepsilon_a = \left(\frac{\text{approximate error}}{\text{approximation}}\right) \times 100\%$$

$$= \frac{\text{current approximation} - \text{previous approximation}}{\text{current approximation}} \times 100\%$$

In simulations we are mostly interested in the absolute value of the error

$|\varepsilon_a| < \varepsilon_s$

$\quad\quad\quad\hookrightarrow$ prescribed tolerance level

Scarbarough, $\varepsilon_s = (0.5 \times 10^{2-n})$ %

if the criteria is followed, result is correct to atleast n significant figures.

Truncation error:

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

$$e_t^{(1)} = 2.718282 \ (\boldsymbol{true\ value})$$

$$e_2^{(1)} = 1 + 1 = 2$$

$$e_3^{(1)} = 1 + 1 + \frac{1}{2} = 2.5$$

$$e_4^{(1)} = 1 + 1 + \frac{1}{2} + \frac{1}{3} = 2.617778$$

For number of terms =2 $E_t = \left|e_t^{(1)} - e_2^{(1)}\right| = |2.718282 - 2| = 0.718282$

For number of terms =2 $\varepsilon_t = \left|\frac{e_t^{(1)} - e_2^{(1)}}{e_t^{(1)}}\right| \times 100 = \left|\frac{2.718282 - 2}{2.718282}\right| \times 100 = 26.424116$

For number of terms =2 $\varepsilon_a = \left|\frac{e_2^{(1)} - e_1^{(1)}}{e_2^{(1)}}\right| \times 100 = \left|\frac{2-1}{2}\right| \times 100 = 50$

For number of terms =3 $E_t = \left|e_t^{(1)} - e_3^{(1)}\right| = |2.718282 - 2.5| = 0.218282$

For number of terms =3 $\varepsilon_t = \left|\frac{e_t^{(1)} - e_3^{(1)}}{e_t^{(1)}}\right| \times 100 = \left|\frac{2.718282 - 2.5}{2.718282}\right| \times 100 = 8.030$

For number of terms =3 $\varepsilon_a = \left|\frac{e_3^{(1)} - e_2^{(1)}}{e_3^{(1)}}\right| \times 100 = \left|\frac{2.5-2}{2.5}\right| \times 100 = 20$

$e_t^{(1)}=2.718282$

Terms	$Result$	$\varepsilon_t(100\%)$	$\varepsilon_a(100\%)$
1	**1**	**63.2120582**	
2	**2**	**26.42411641**	**100**
3	**2.5**	**8.030145511**	**40**
4	**2.666666667**	**1.898821878**	**12.5**
5	**2.708333333**	**0.36599097**	**3.076923077**
6	**2.716666667**	**0.059424789**	**0.613496933**
7	**2.718055556**	**0.008330425**	**0.102197241**
8	**2.718253968**	**0.00103123**	**0.01459854**

Iterative approximate Henon Approximation $\sqrt{S}$ ::::::$x^{[i+1]} = \frac{1}{2}\left(x^{[i]} + \frac{S}{x^{[i]}}\right)$

$$\sqrt{3} = 1.732050808$$

$[i]$	$x^{[i]}$	$3/x^{[i]}$	$x^{[i+1]}$=0.5*($x^{[i]}$ + $3/x^{[i]}$)	$\varepsilon_t{}^{[i+1]} = \left\lvert \frac{x^{[i+1]} - \sqrt{3}}{\sqrt{3}} \right\rvert \times 100\%$	$\varepsilon_t{}^{[i+1]} = \left(\frac{x^{[i+1]} - x^{[i]}}{x^{[i+1]}} \right) \times 100\%$
0	**0.1**	30	**15.08**	**770.6442059**	**99.33687003**
1	**15.05**	0.199	**7.6247**	**340.2122597**	**97.38481514**
2	**7.6247**	0.393	**4.009**	**131.4597229**	**90.18957346**
3	**4.009**	0.748	**2.3787**	**37.33430853**	**68.53743641**
4	**2.3787**	1.2612	**1.81997**	**5.076016942**	**30.69995659**
5	**1.8199**	1.6484	**1.73205**	**4.6625E-05**	**5.07202448**
6	**1.7341**	1.72993	**1.73205**	**4.6625E-05**	**0.11835686**
7	**1.73205**	1.732049	**1.73205**	**4.6625E-05**	**0**
8	**1.73205**	1.73205	**1.73205**	**4.6625E-05**	**0**
9	**1.73205**	1.73205	**1.73205**	**4.6625E-05**	**0**

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Prof. Jayati Sarkar

Round off errors:

- Originate from the fact that computers can retain only a fixed number of significant figures during calculation.

- Irrational numbers like $\sqrt{7}$, e, π etc. cannot be expressed by a fixed number of significant figures.

- Also computers use a base 2 representation so they cannot precisely represent base 10 numbers

- The discrepancy introduced by omission of significant figures.

Computer representation of numbers

Fundamental unit whereby information is stored consists of a string of binary digits: bits

Base 10 systems uses 10 digits:- 1, 2,3 4, 5, 6, 7, 8, 9,0 to represent numbers

Place and face values are multiplied to get the number

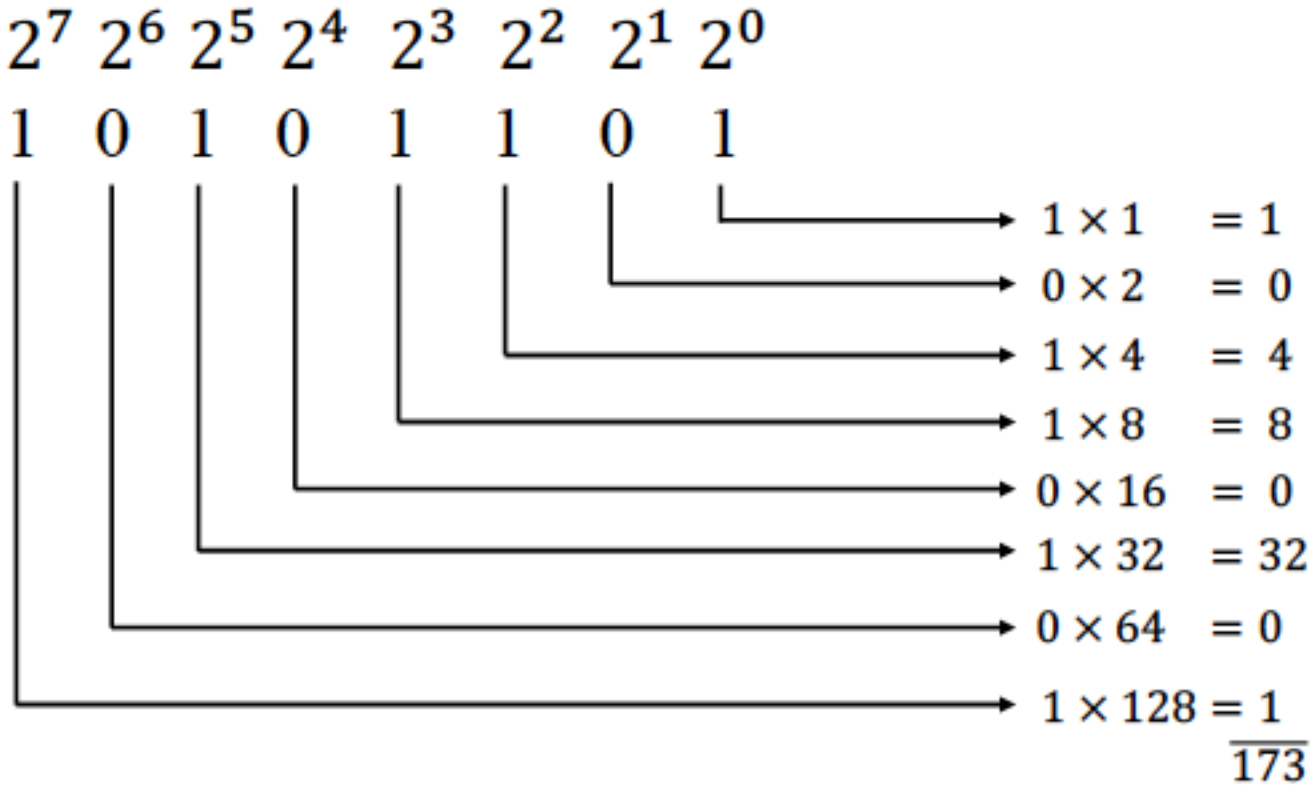
86409→

$$(8\times10^4)+(6\times10^3)+(4\times10^2)+(0\times10^1)+(9\times10^0)$$

10^4	10^3	10^2	10^1	10^0
8	6	4	0	9

$$\begin{aligned}
9\times1 &= & 9\\
0\times10 &= & 00\\
4\times100 &= & 400\\
6\times1000 &= & 6000\\
8\times10000 &= & 80000\\
\hline
& & 86409
\end{aligned}$$

- Computers represent numbers in binary base 2 system



Integer representation in a computer

- Signed magnitude approach

  First bit represents sign

  0 for positive (+) number

  1 for negative (-) integer

Integer representation in a computer

For 16 bit computer

0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	1

Sign

<u>Number</u>

Range of base 10 that can be represented on a 16 bit computer

0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

$= 2^{14} \times 1 + 2^{13} \times 1 + 2^{12} \times 1 + 2^{11} \times 1 + 2^{10} \times 1 + 2^{9} \times 1 + \ldots\ldots + 2^{1} \times 1 + 2^{0} \times 1$

$= 2^{15} - 1$

$= 32767$ range is from - 32767 to 32767

Since 1000000000000000 ~ is redundant as +0 & -0 are same , so one more is added to -ve integer that is -32768 to 32767.

Integer numbers above and below this cannot be stored

FLOATING POINT REPRESENTATION

Fractional values can be represented by floating point representation

$m \, . \, b^{e}$

where m = the mantissa, b = the base of the number system being used, and e = the exponent

Fractional part is called mantissa or significand.

$156.78 \rightarrow 0.15678 \times 10^{3}$

$1/34 = 0.029411765$

↪for decimal system 10 base

0.0294×10^{0}

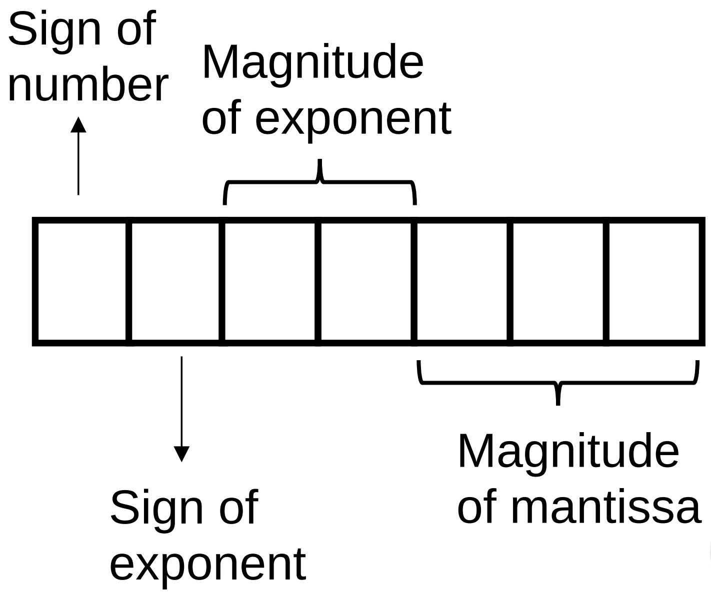
0.2941×10^{-1}

$$\boxed{1/b \le m < 1}$$

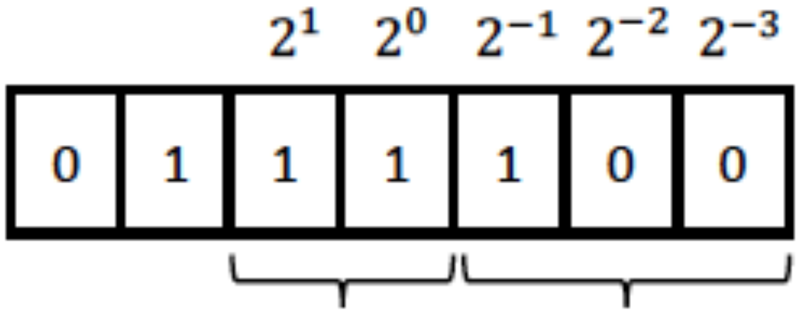
In 10 base system $0.1 \le m < 1$

In 2 base system $0.5 \le m < 1$

For a machine that stores 7 bit



Smallest Number



Magnitude of mantissa

$= 1\times2^{-1}+0\times2^{-2}+0\times2^{-3}$

$= 0.5$

Magnitude of exponent

$= 1 \times 2^1 + 2^0 = 3$

Not 0111000 Not admissible

001

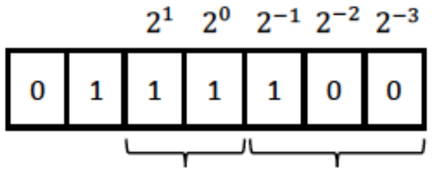
010

011

Exponent: $1 \times 2^1 + 2^0 = 3$

Magnitude: $1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3} = 0.5$

Number: $0.5 \times 2^{-3} = 0.0625$



$0111100 = (1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3}) \times 2^{-3} = (0.06250)_{10}$

$0111101 = (1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}) \times 2^{-3} = (0.078125)_{10}$

$0111110 = (1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3}) \times 2^{-3} = (0.093750)_{10}$

$0111111 = (1 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3}) \times 2^{-3} = (0.109375)_{10}$

$\boxed{\Delta x_1 = 0.015625}$

$0110100 = (1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3}) \times 2^{-2} = (0.125000)_{10}$

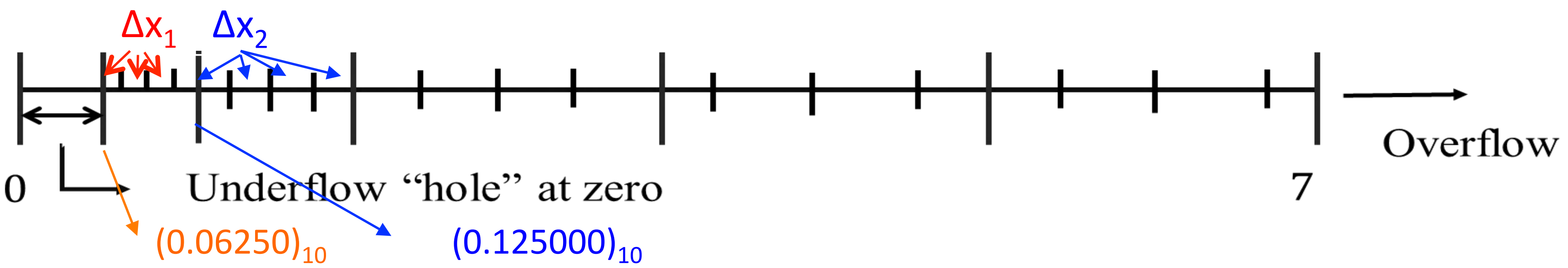
$0110101 = (1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}) \times 2^{-2} = (0.156250)_{10}$

$0110110 = (1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3}) \times 2^{-2} = (0.187500)_{10}$

$0110111 = (1 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3}) \times 2^{-2} = (0.218750)_{10}$

$\boxed{\Delta x_2 = 0.03125}$

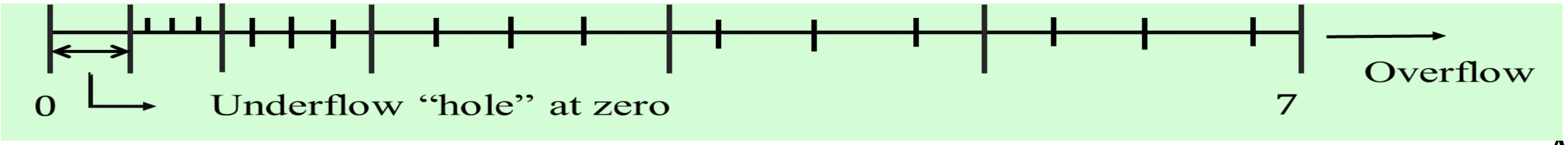
Max: $0011111 = (1 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3}) \times 2^{3} = 7$

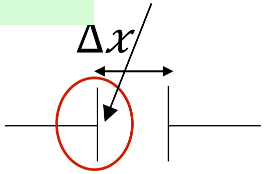


There is a limited range of quantities that may be represented

- Large +ve and -ve quantities cannot be represented → **overflow error**
- In addition very small quantities near to zero cannot be represented → **underflow error**

→There are <u>Only</u> a finite number of quantities that can be represented within the range



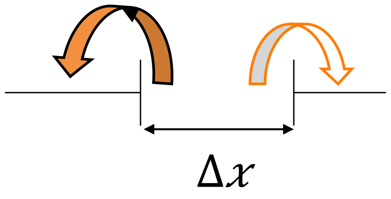


→The degree of precision is limited

If the value of fractional number falls here, Value it takes is that of lower one

So, error is always biased → This is called "CHOPPING OFF".

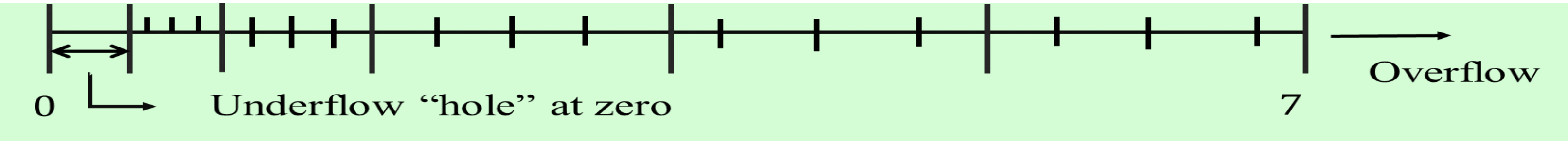
So, maximum E_t is ~(Δx)



→ Alternative is to round off the error

Depending on the value it is represented as the nearest allowable nu

So, maximum E_t is rounding off ~(Δx/2)



The interval between the numbers Δx increases as the number grows in magnitude

→It allows floating-point representation to preserve significant digits

→Quantisizing errors $\propto$ magnitude of number it represents

$|\Delta x|/|x| < \varepsilon$ Chopping

$|\Delta x|/|x| < \varepsilon/2$ Round off error

ε → machine epsilon

$= b^{(1-t)}$ (no of significant digit in mantissa)

→Though round off errors are important → testing convergence

- No. of significant digits carried on most computers allows most engineering problems to be performed with more than acceptable precision

Computer with IEEE format

~24 bits from mantissa →10^{-38} to 10^{39}

Double precision ~15 to16 decimal digits of precision

~10^{-308} to 10^{308}

Common arithmetic operations

Addition

- $0.1557 \times 10^1 + 0.4381 \times 10^{-1}$
- Hypothetical decimal computer with 4-digit mantissa and 1 digit exponent

The decimal of the mantissa of the smallest number is to be shifted by

$(1-(-1)) = 2$ places

$$
\begin{array}{l}
0.1557 \quad \times 10^1 \\
\underline{0.004381 \times 10^1} \\
0.1600\underline{81} \times 10^1
\end{array}
$$

$\downarrow$

- Chopped off- the last two digits shifted to the right is essentially lost

More Acute when adding two vastly different magnitude number (4000 & 0.001)

$$
\begin{array}{l}
\Rightarrow 0.4000 \quad \times 10^4 \\
\quad \underline{0.0000001 \times 10^4} \\
\quad 0.4000\underline{001} \times 10^4 \quad \Rightarrow \quad 0.4000 \times 10^4 \text{ (Chopped off)}
\end{array}
$$

$\downarrow$

as if no addition has been done

Subtraction

28.86 from 36.41

$\Rightarrow$ 0.3641×10^2

$\underline{0.2886 \times 10^2}$

0.0995×10^2

$0.\ 995\underline{0} \times 10^1$ (Addition of zero)

$\rightarrow$ Subtracting two close numbers 0.7641×10^3 from 0.7642×10^3

$\Rightarrow$ 0.7642×10^3

$\underline{0.7641 \times 10^3}$

0.0001×10^3

$0.1\underline{000} \times 10^0$

$\downarrow$

Will appear as a significant digit in subsequent computation

Multiplication

$0.1363 \times 10^3 \times 0.6423 \times 10^{-1}$

Multiplication of two n digit mantissa will yield 2n digit results,

$\rightarrow$ Most computers hold intermediate results in a double length register

Exponents are added, mantises are multiplied

0.08754549×10^2

$\Rightarrow 0.8754\underline{549} \times 10^1$

Chopped off

$0.8754 \times 10^2 \Rightarrow$ loosing precision

NUMERICAL METHODS IN CHEMICAL ENGINEERING

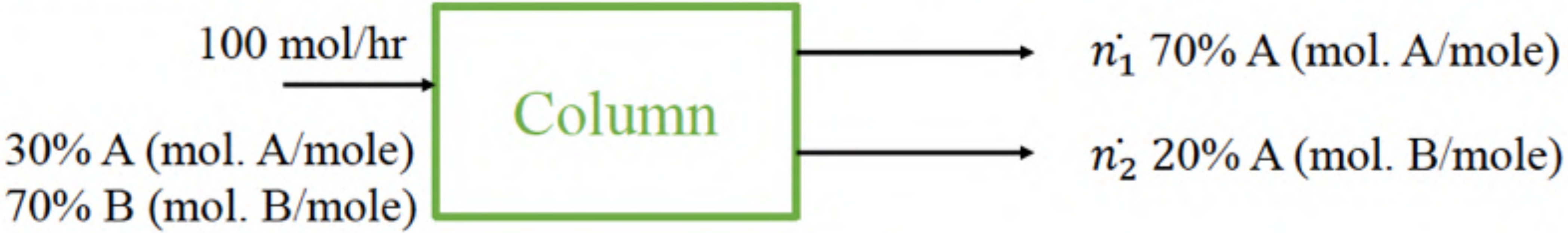
CLL-113

Solution of Linear Equations

Prof. Jayati Sarkar

Material Balance

In all types of separation processes



$\dot{n_0} = 100\frac{mol}{hr}$ $\qquad \dot{n_1} = ?$ $\qquad \dot{n_2} = ?$

$100\frac{mol}{hr} = \dot{n_1} + \dot{n_2}$ (what goes in and comes out)

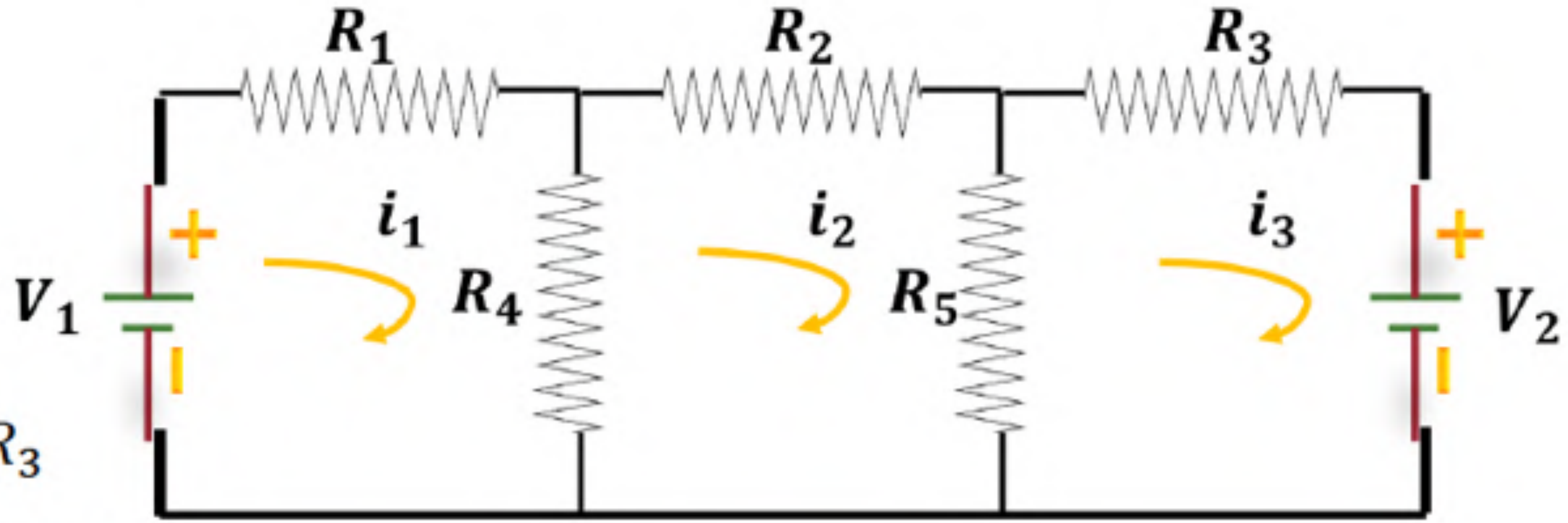
Species balance : A $\frac{30\ mol\ A}{100\ mol} \times 100\frac{mol}{hr} = \dot{n_1} \times \frac{70}{100} + \dot{n_2} \times \frac{20}{100}$

$30 = 0.7\dot{n_1} + 0.2\ \dot{n_2}$

$$\boxed{\begin{aligned} 100 &= \dot{n_1} + \dot{n_2} \\ 30 &= 0.7\dot{n_1} + 0.2\ \dot{n_2} \end{aligned}}$$

$$\underset{A}{\begin{bmatrix} 1 & 1 \\ 0.7 & 0.2 \end{bmatrix}} \underset{x}{\begin{bmatrix} n_1 \\ n_2 \end{bmatrix}} \underset{=}{=} \underset{b}{\begin{bmatrix} 100 \\ 30 \end{bmatrix}}$$

CIRCUIT DIAGRAM



$$V_1 = i_1 R_1 + (i_1 - i_2) R_4 + 0 R_3$$

$$0 = i_2 R_2 + (i_2 - i_3) R_5 + (i_2 - i_1) R_4$$

$$-V_2 = i_3 R_3 + (i_3 - i_2) R_5$$

$$\begin{bmatrix} (R_1 + R_4) & -R_4 & 0 \\ -R_4 & (R_2 + R_4 + R_5) & -R_5 \\ 0 & -R_5 & (R_3 + R_5) \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix} = \begin{bmatrix} V_1 \\ 0 \\ -V_2 \end{bmatrix}$$

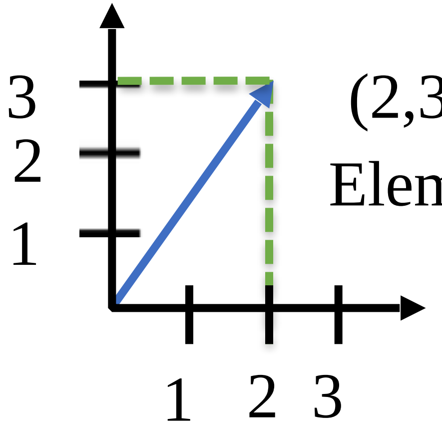
A x = b

$x \in R^2 \longleftarrow$ dim

Scalar, Vector, Matrices

Scalar: T=500°C :Magnitude

Vector : Magnitude and direction (velocity, acceleration)



(2,3)

Element of

A point in 2-dimension space

Represented as ordered set of scalar

$$x = \begin{bmatrix} 2 \\ 3 \end{bmatrix}_{2\times 1}$$

Rows Column

Norm $= \sqrt{2^2 + 3^2} = \sqrt{13}$

$$x_5 = \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix}_{n\times 1}$$

Matrix: Rectangular array of numbers

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

Liner operations addition / subtraction
Scalar/Matrix multiplication rules,
Eigen value/ vectors,
Rank of a matrix by Echelon method

A Linear Equation

$$4x + 5y + 7z + 3a = 15$$

Coefficients

Variables

- 4 equations to solve this
- It resembles 3 dimensional hyper space::::::::::1 degree is lost

- Not guaranteed that will have a solution
- A n×n matrix will have a unique solution when

Rank of the matrix should be $= n$

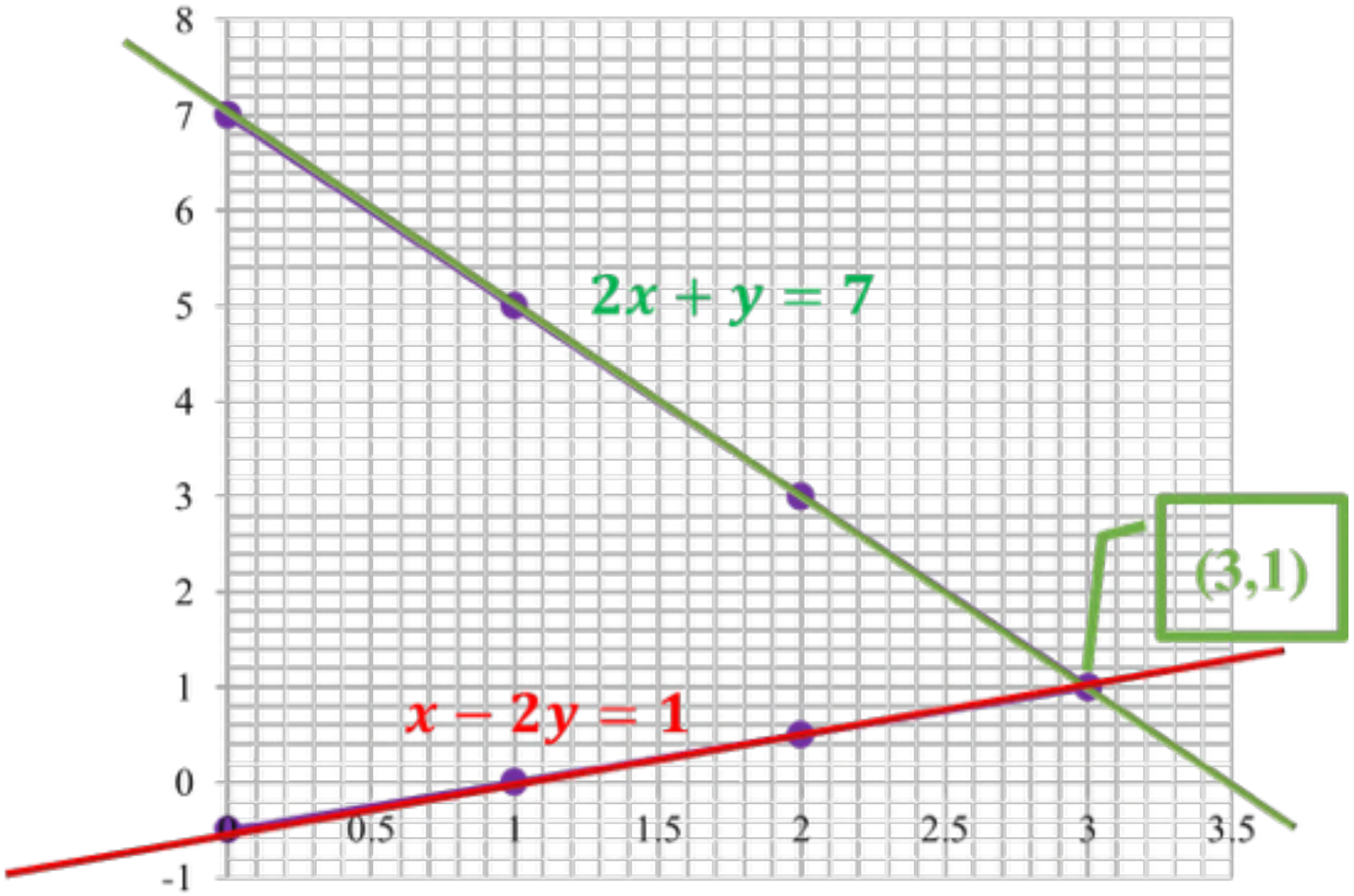
- Two Equations
- Two unknows

$$x - 2y = 1$$

$$2x + y = 7$$

$$\begin{bmatrix} 1 & -2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 7 \end{bmatrix}$$

A x = b



No. of rows : No. of Equations
No. of Columns: No. of unknowns

Determinant methods Not used in beyond n=3

C RAMER'S RULE

D = Determinant of Matrix

D_i = Determinant of A_i where

A_i denoted by replacing i^{th} column with vector b

$$D = \begin{vmatrix} 1 & -2 \\ 2 & 1 \end{vmatrix} = 5$$

$$D_1 = \begin{vmatrix} 1 & -2 \\ 7 & 1 \end{vmatrix} = 15 \qquad D_2 = \begin{vmatrix} 1 & 1 \\ 2 & 7 \end{vmatrix} = 5$$

A unique solution exits of D≠0

$$x = \frac{D_1}{D} = \frac{15}{5} = 3 \qquad y = \frac{D_2}{D} = \frac{5}{5} = 1$$

Solution = (3,1)

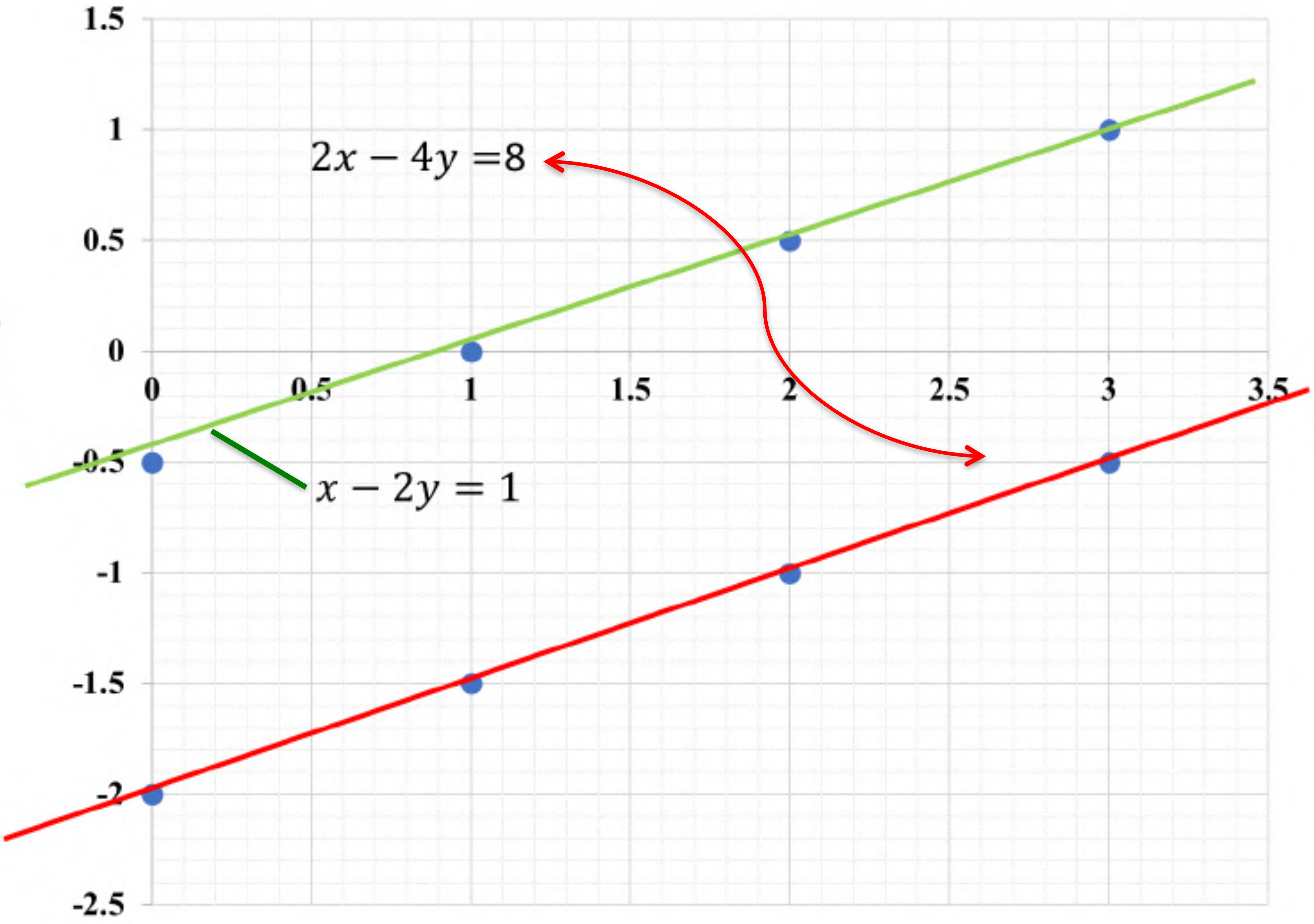
$$x - 2y = 1$$

$$2x - 4y = 8$$

D=1×(-4)-(2)×(-2)=0

Parallel lines

No Solutions

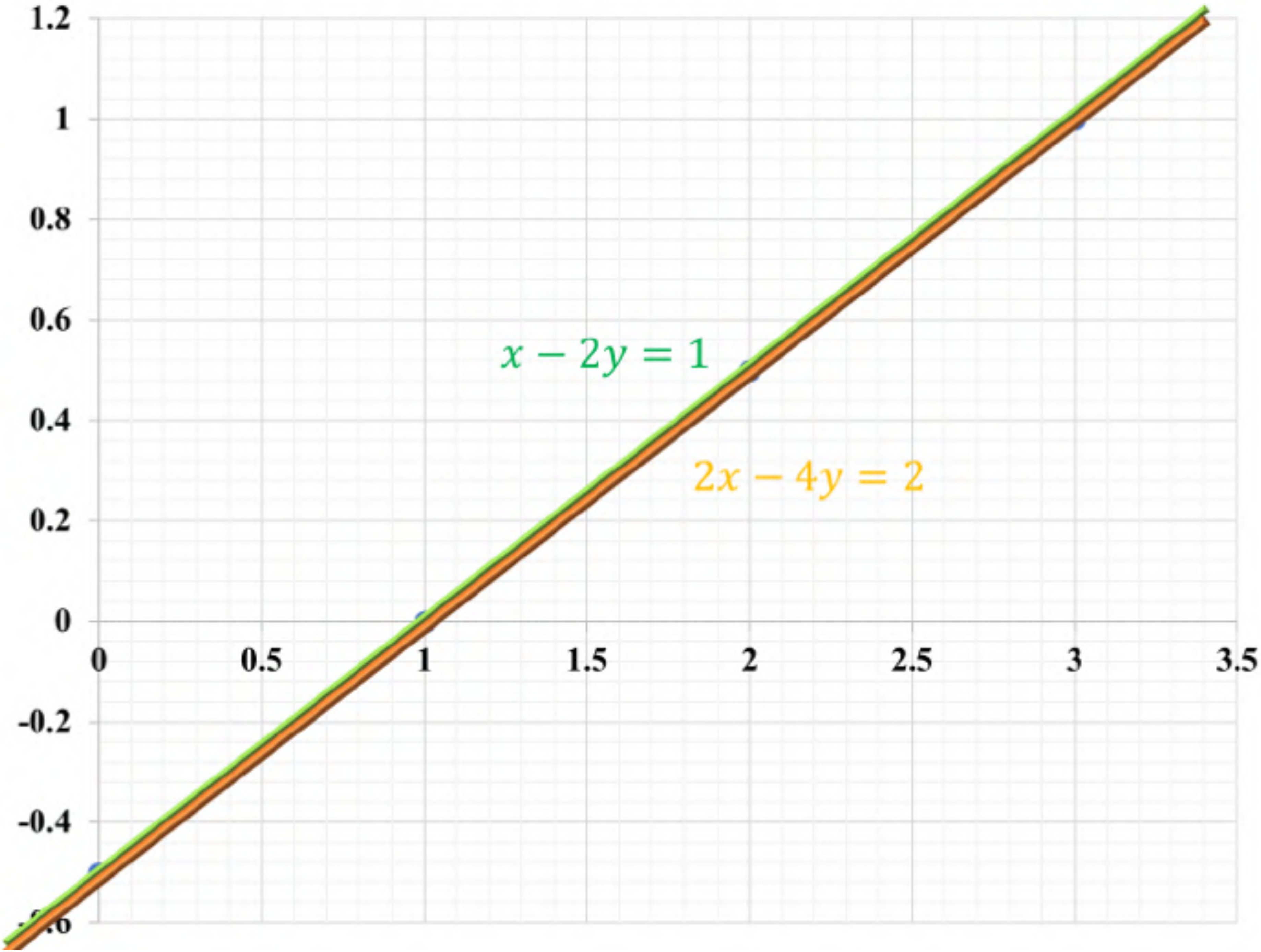


$$x - 2y = 1$$

$$2x - 4y = 2$$

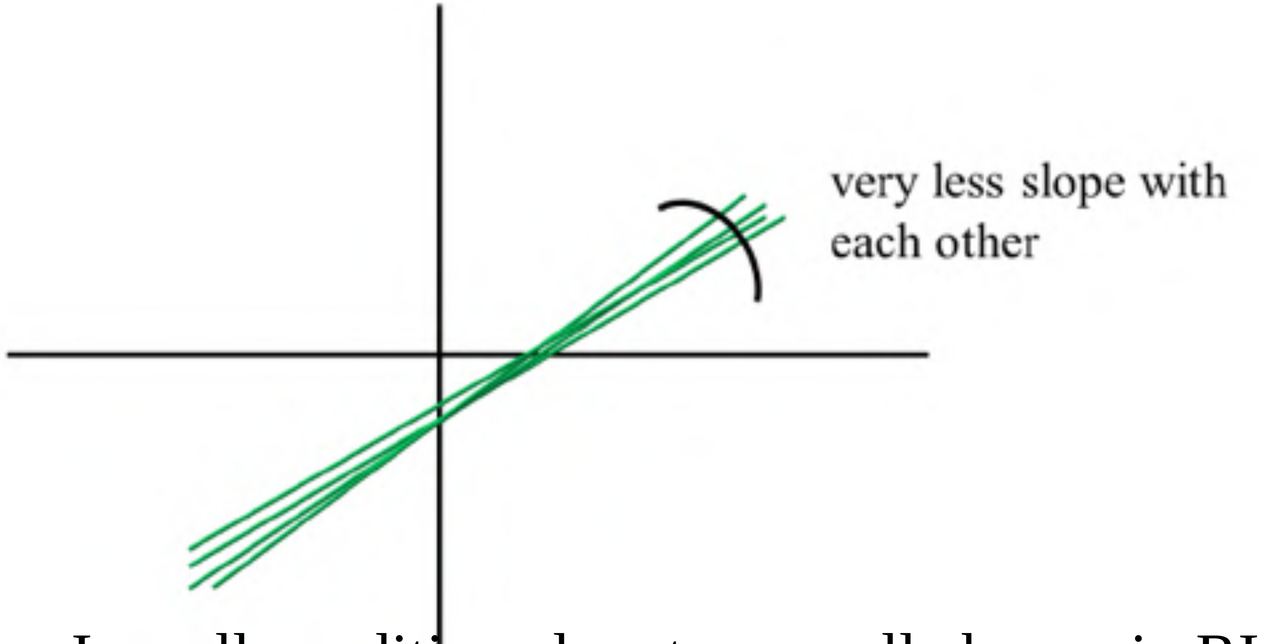
D=1×(-4)-(2)×(-2)=0

Same line
Infinite Solutions



$$x - 2y = 1 \qquad \Longrightarrow \quad y = -1$$
$$2x - 3.999y = 1.999 \qquad \Longrightarrow \quad x = -1$$

$$x - 2y = 1 \qquad \Longrightarrow \quad y = 0$$
$$2x - 3.999y = 2 \qquad \Longrightarrow \quad x = 1.0$$



- In well conditioned system small change in RHS caused small change in solutions
- In ill conditioned system small change in RHS caused large change in solutions

$$A = \begin{bmatrix} 1 & -2 \\ 2 & -3.999 \end{bmatrix} \rightarrow Eigen\ value$$

$$\lambda_2 = -0.00033348$$
$$\& \ \lambda_1 = -2.99867$$

$$the\ ratio = \frac{|\lambda_1|}{|\lambda_2|} = 8992$$

Highest to lowest Eigen Value

Very difficult to solve Stiff equations

Interpretation

$$x - 2y = 1$$

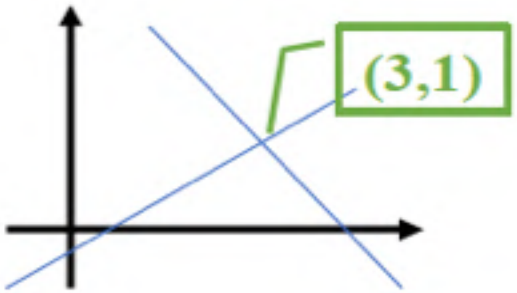
$$2x + y = 7$$

$\Longrightarrow$

Matrix Representation

$$\begin{bmatrix} 1 & -2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 7 \end{bmatrix}$$

$$A \quad x \quad = \quad b$$



Row picture

Intersection of lines/planes/Hyper surfaces==➔concept becomes difficult as n increases

Alternate from [column picture]

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} x + \begin{bmatrix} -2 \\ 1 \end{bmatrix} y = \begin{bmatrix} 1 \\ 7 \end{bmatrix}$$

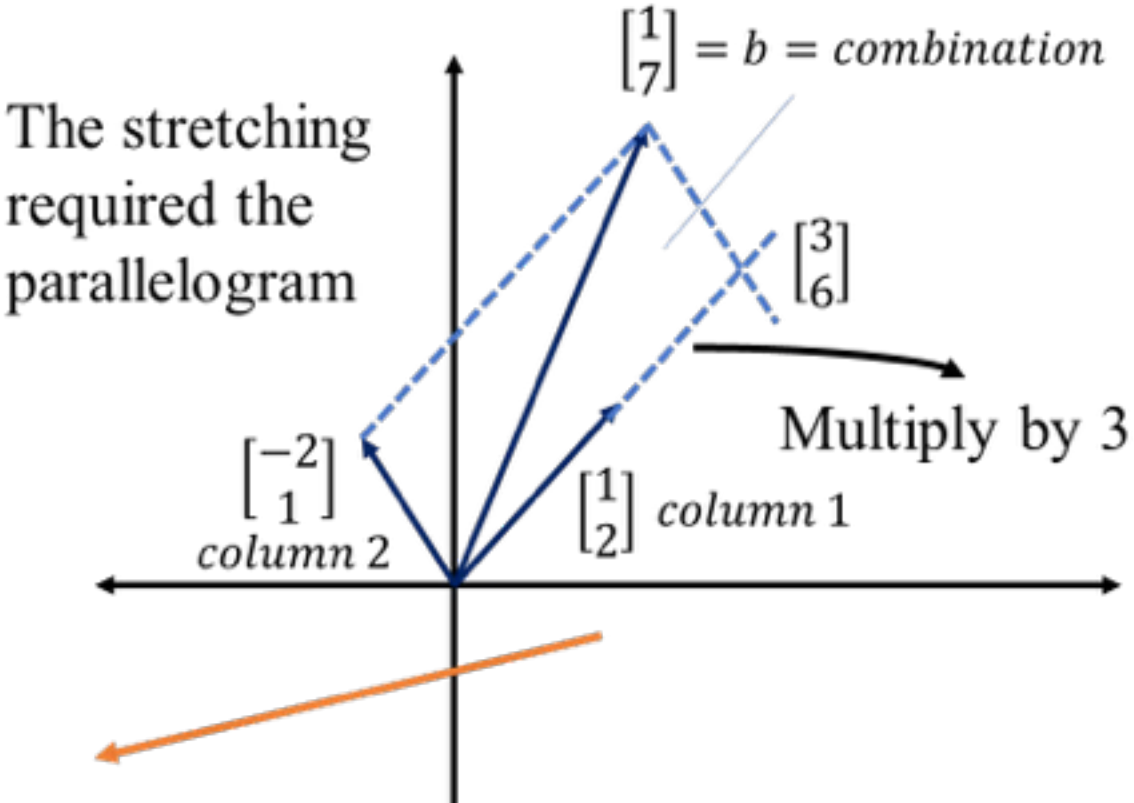
$$ux + wy = b \dots\dots\dots (\epsilon)$$

Linear Combination of two vectors is the resultant vector

Requirement to find the scalar that solve equation

A combination

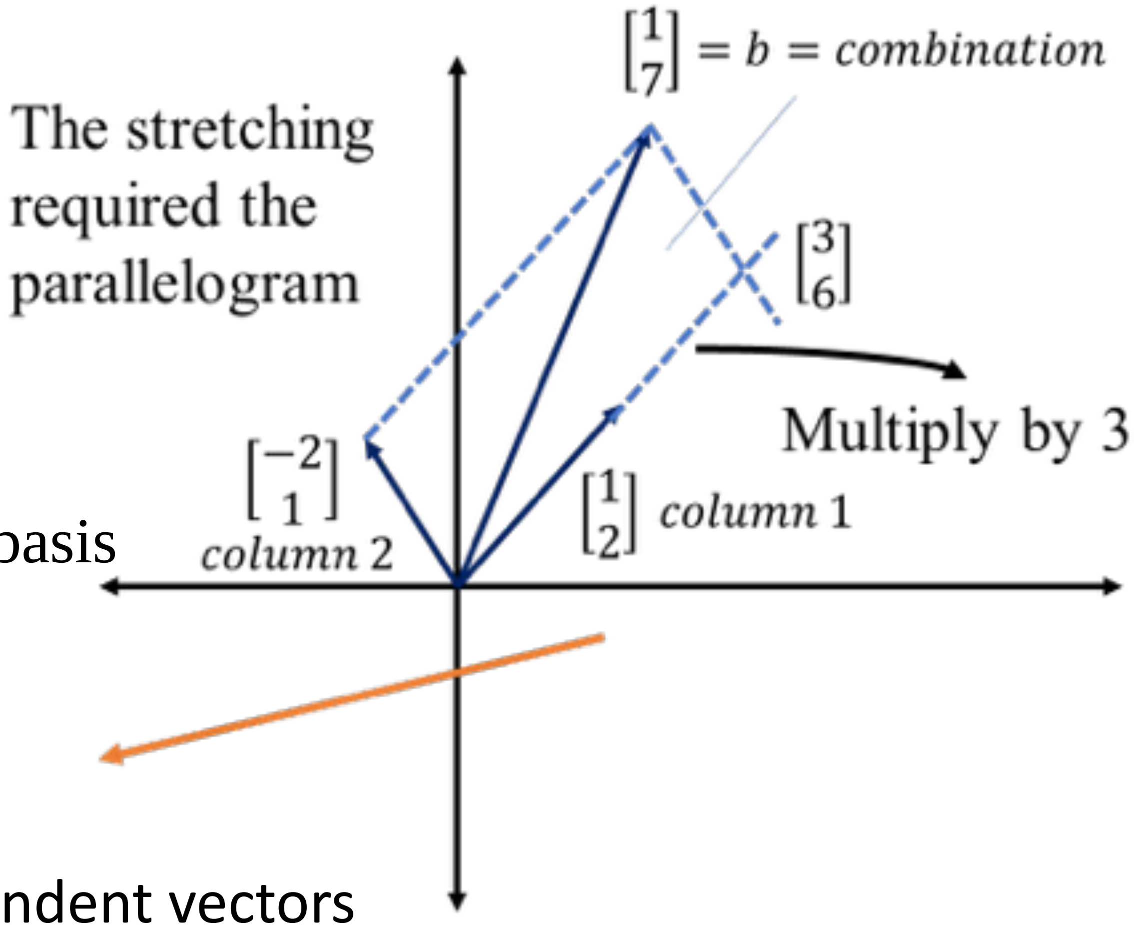
3(column 1) + 1(column 2) = Column b



VECTOR SPACE

[VECTOR] & → satisfies rules to vector additions, subtraction & multiplication

R^2 › *Space* $V = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ $W = \begin{bmatrix} -2 \\ 1 \end{bmatrix}$ $b = \begin{bmatrix} 1 \\ 7 \end{bmatrix}$



If V & W are linearly independent they form the basis for that particular vector space

R^n space there can be at most n linearly independent vectors

If we consider vector

$$V = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \& \quad m = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 4 \\ 8 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} 4 \\ 8 \end{bmatrix}$$

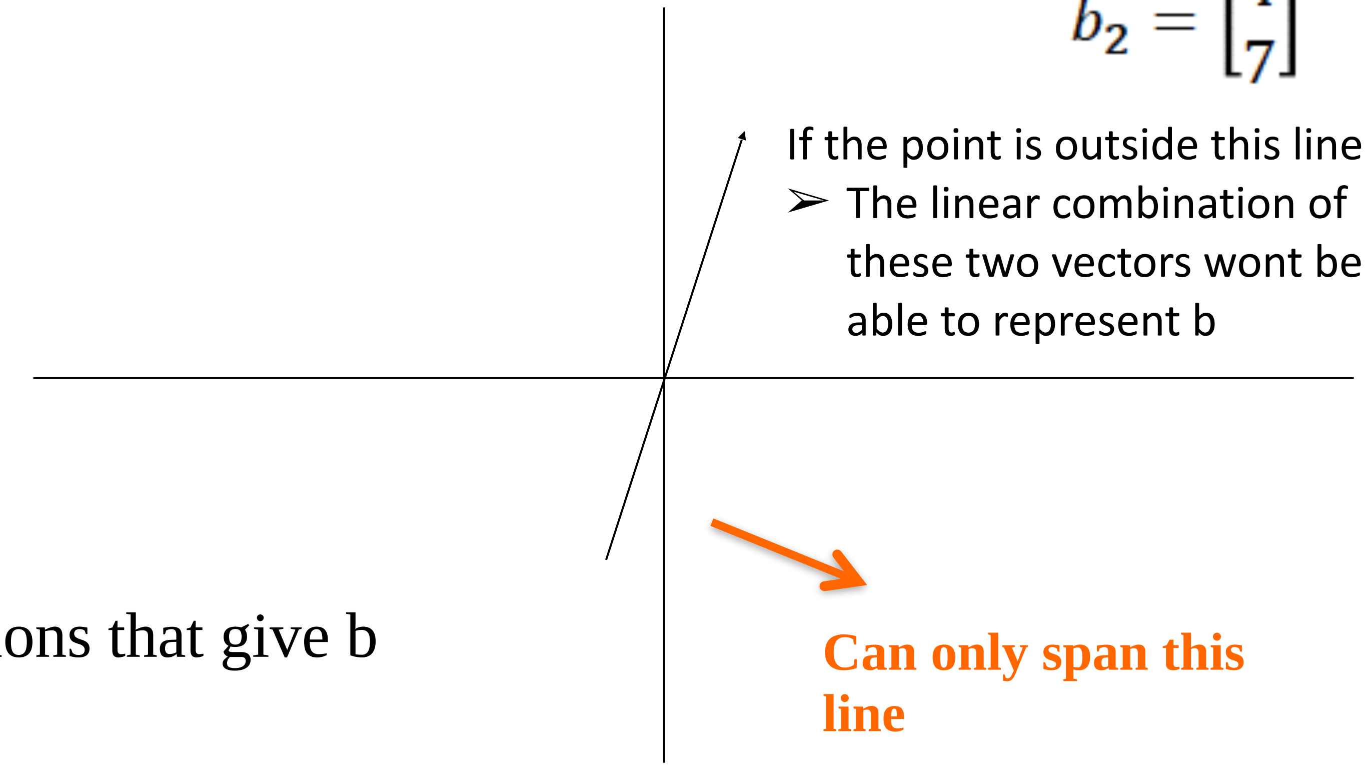
In this case there are infinite combinations that give b

➢As base vectors

NOT linearly independent

2V=m

$$b_2 = \begin{bmatrix} 4 \\ 7 \end{bmatrix}$$



If the point is outside this line

- ➢ The linear combination of these two vectors wont be able to represent b

Can only span this line

In case of linearly independent vectors

Rank $\{[A|b]\}$ = rank {A} = n

Same as saying $|D| \neq 0$

→ Unique solution

$$V = \begin{bmatrix}1\\2\end{bmatrix};\ W = \begin{bmatrix}2\\4\end{bmatrix};\ b_1 = \begin{bmatrix}4\\8\end{bmatrix}\ Infinite\ Solution$$

$$V = \begin{bmatrix}1\\2\end{bmatrix};\ W = \begin{bmatrix}2\\4\end{bmatrix};\ b_2 = \begin{bmatrix}4\\7\end{bmatrix}$$ No solution

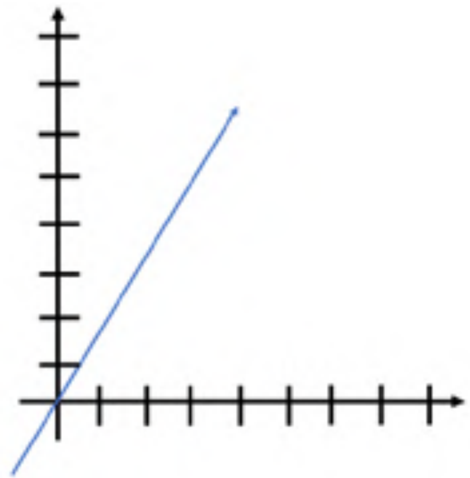
$|D| = 0\ Singular\ Matrix$

b_1 Rank of $\{[A|b]\}$ = rank {A} < n Infinite solution

b_2 Rank of $\{[A|b]\}$ > rank {A} No solution

Poorly conditioned matrix

$$\begin{bmatrix}1 & 2\\2 & 3.999\end{bmatrix}$$



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

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

.

.

$$a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \ ... + a_{nn}x_n = b_n$$

Cramer's Rule & matrix inversion are not preferred for higher N specially when $|A| \sim 0$ *or* ill conditionate matrix

➢ Direct methods

- Gauss Elimination, Gauss Jordon Elimination, Sparse method Thomas Algorithm

➢ Indirect methods/ Iterative Methods

- Jacobi, Gauss-seidel & relaxation methods

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Solution of Linear Equations

Prof. Jayati Sarkar

General $n \times n$ system

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

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

  .

  .

  $a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \ \dots + a_{nn}x_n = b_n$

$$\begin{bmatrix} a_{11} & a_{12} \cdots & a_{1n} \\ a_{21} & a_{22} \cdots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

$$\mathrm{A} \qquad \mathrm{x} \quad = \quad \mathrm{b}$$

- What does the solution signify?
- Row format
- Column format
- Unique solution?

Example of Elimination

$$x_1 - 2x_2 = 1 \ldots\ldots\ldots (1)$$

$$2x_1 + x_2 = 7 \ldots\ldots\ldots (2)$$

$$\begin{bmatrix} 1 & -2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 7 \end{bmatrix}$$

$$\mathrm{A} \quad \mathrm{x} \quad = \quad \mathrm{b}$$

Eliminate x_1

$$eq_2^n + eq_1^n(-2) = eq_3^n$$

$$0x_1 + x_2 + 4x_2 = 7 - 2$$

$$0x_1 + 5x_2 = 5 \ldots\ldots. (3)$$

$$5x_2 = 5$$

$$x_2 = 1$$

Back substitution in eq_1^n

$$x_1 - 2 \times 1 = 1$$

$$x_1 = 1 + 2 = 3$$

Are operations on original equations justified?

$$\begin{bmatrix} 1 & -2 : 1 \\ 2 & 1 : 7 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & -2 : 1 \\ 0 & 5 : 5 \end{bmatrix}$$

- $Solution = [3,1]$

$$L_1 \equiv x_1 - 2x_2 = 1$$
$$L_2 \equiv 2x_1 + x_2 = 7$$

$$\begin{aligned} \text{LHS} = \alpha L_1 + \beta L_2 &= \alpha(x_1 - 2x_2) + \beta(2x_1 + x_2) \\ &= \alpha(3-2) + \beta(6+1) \\ &= \alpha + 7\beta \\ &= RHS \end{aligned}$$

$$RHS = 1\alpha + 7\beta$$

$Solution \begin{bmatrix} 3 \\ 1 \end{bmatrix}$ *lies on* L_1 & L_2 *also lies on linear combination of* $\alpha L_1 + \beta L_2$ *this help us to take linear combinations of eq. 1 & eq. 2 to get eq. 3*

Gauss Elimination

$$A^{(0)}x = b^{(0)}$$

$$A^{(0)} = \begin{bmatrix} a^0{}_{11} & a^0{}_{12} & a^0{}_{13} & & a^0{}_{1n} \\ a^0{}_{21} & a^0{}_{22} & a^0{}_{23} & & a^0{}_{2n} \\ & & & & \\ & & & & \\ a^0{}_{n1} & a^0{}_{n2} & a^0{}_{n3} & & a^0{}_{nn} \end{bmatrix}; \mathrm{b} = \begin{bmatrix} b^0{}_1 \\ b^0{}_2 \\ \\ \\ b^0{}_n \end{bmatrix}$$

$$\boxed{eq.2 = eq.2 + \left(-\frac{a_{21}}{a_{11}}\right) eq.1}$$

- $\left(a_{21} - \frac{a_{21}}{a_{11}}.a_{11}\right)x_1 + \left(a_{22} - \frac{a_{21}}{a_{11}}.a_{12}\right)x_2 + \left(a_{23} - \frac{a_{21}}{a_{11}}.a_{13}\right)x_3 + \cdots = \left(b_2 - \frac{a_{21}}{a_{11}}b_1\right)$
- $a_{22}^1 x_2 + a_{23}^1 x_3 + \cdots a_{2n}^1$ $x_n =$ b_2^1
- Continue doing it for $i = 2\ to\ N$

Step k=1

Pivoted Element

Pivoting row k=1

$$
A^{(1)} = \begin{bmatrix}
a_{11}^{(0)} & a_{12}^{(0)} & \cdots & a_{1n}^{(0)} \\
0 & \left[a_{22}^{(0)} - \dfrac{a_{21}^{(0)}}{a_{11}^{(0)}} \times a_{12}^{(0)}\right] & \cdots & \left[a_{2n}^{(0)} - \dfrac{a_{21}^{(0)}}{a_{11}^{(0)}} \times a_{1n}^{(0)}\right] \\
0 & \left[a_{32}^{(0)} - \dfrac{a_{31}^{(0)}}{a_{11}^{(0)}} \times a_{12}^{(0)}\right] & \cdots & \left[a_{3n}^{(0)} - \dfrac{a_{31}^{(0)}}{a_{11}^{(0)}} \times a_{1n}^{(0)}\right] \\
. & . & \cdots & . \\
0 & \left[a_{n2}^{(0)} - \dfrac{a_{n1}^{(0)}}{a_{11}^{(0)}} \times a_{12}^{(0)}\right] & \cdots & \left[a_{nn}^{(0)} - \dfrac{a_{n1}^{(0)}}{a_{11}^{(0)}} \times a_{1n}^{(0)}\right]
\end{bmatrix}
$$

$$
b^{(1)} = \begin{bmatrix}
b_1^{(0)} \\
b_2^{(0)} - \dfrac{a_{21}^{(0)}}{a_{11}^{(0)}} \times b_1^{(0)} \\
b_3^{(0)} - \dfrac{a_{31}^{(0)}}{a_{11}^{(0)}} \times b_1^{(0)} \\
. \qquad . \\
b_n^{(0)} - \dfrac{a_{n1}^{(0)}}{a_{11}^{(0)}} \times b_1^{(0)}
\end{bmatrix}
$$

At the end of Step k=1

$$A^{(1)} = \begin{bmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \ldots\ldots\ldots\ldots a_{1n}^{(0)} \\ \mathbf{0} & a_{22}^{(1)} & \ldots\ldots\ldots\ldots a_{2n}^{(1)} \\ \mathbf{0} & a_{32}^{(1)} & \ldots\ldots\ldots\ldots a_{3n}^{(1)} \\ . & . & \ldots\ldots\ldots\ldots \quad . \\ \mathbf{0} & a_{n2}^{(1)} & \ldots\ldots\ldots\ldots a_{nn}^{(1)} \end{bmatrix} \qquad b^{(1)} = \begin{bmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(1)} \\ . \\ b_n^{(1)} \end{bmatrix}$$

k=1

$$i = 2, \ldots \ldots \ldots \ldots \ldots . n$$

$$j = 2, \ldots \ldots \ldots \ldots \ldots . n$$

$$a_{ij}^{(1)} = a_{ij}^{(0)} - \frac{a_{i1}^{(0)}}{a_{11}^{(0)}} \times a_{1j}^{(0)}$$

$$b_i^{(1)} = b_i^{(0)} - \frac{a_{i1}^{(0)}}{a_{11}^{(0)}} \times b_1^{(0)}$$

Step k=2

Pivoted Element

Pivoting row

k=2

$$A^{(2)} = \begin{bmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \cdots\cdots\cdots & a_{1n}^{(0)} \\ \mathbf{0} & a_{22}^{(1)} & \cdots\cdots\cdots & a_{2n}^{(1)} \\ \mathbf{0} & \mathbf{0} & \left[a_{33}^{(1)} - \dfrac{a_{32}^{(1)}}{a_{22}^{(1)}} \times a_{23}^{(1)}\right] \cdots\cdots & \left[a_{3n}^{(1)} - \dfrac{a_{32}^{(1)}}{a_{22}^{(1)}} \times a_{2n}^{(1)}\right] \\ . & . & \cdots\cdots & . \\ \mathbf{0} & \mathbf{0} & \left[a_{n3}^{(1)} - \dfrac{a_{n2}^{(1)}}{a_{22}^{(1)}} \times a_{23}^{(1)}\right] \cdots\cdots & \left[a_{nn}^{(1)} - \dfrac{a_{n2}^{(1)}}{a_{22}^{(1)}} \times a_{2n}^{(1)}\right] \end{bmatrix}$$

$$b^{(2)} = \begin{bmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(1)} - \dfrac{a_{32}^{(1)}}{a_{22}^{(1)}} \times b_2^{(1)} \\ . \qquad . \\ b_n^{(1)} - \dfrac{a_{n2}^{(1)}}{a_{22}^{(1)}} \times b_2^{(1)} \end{bmatrix}$$

At the end of Step k=2

$$A^{(2)} = \begin{bmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \ldots\ldots\ldots\ldots a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & \ldots\ldots\ldots\ldots a_{2n}^{(1)} \\ 0 & 0 & a_{33}^{(2)} \ldots\ldots\ldots a_{3n}^{(2)} \\ . & . & \ldots\ldots\ldots\ldots \quad . \\ 0 & 0 & a_{n2}^{(2)} \ldots\ldots\ldots a_{nn}^{(2)} \end{bmatrix} \qquad b^{(2)} = \begin{bmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(2)} \\ . \\ b_n^{(2)} \end{bmatrix}$$

k=2

$$i = 3, \ldots\ldots\ldots\ldots\ldots n$$

$$j = 3, \ldots\ldots\ldots\ldots\ldots n$$

$$a_{ij}^{(2)} = a_{ij}^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}} \times a_{2j}^{(1)}$$

$$b_i^{(2)} = b_i^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}} \times b_2^{(1)}$$

For general Step k

K is the pivoting row and a_{kk}^{k-1} is the pivoting element

$$k = 1, 2, \ldots.(n-1)$$

k=k

$$i = (k+1), \quad \ldots\ldots\ldots.n$$

$$j = (k+1), \quad \ldots\ldots\ldots.n$$

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} \times a_{kj}^{(k-1)}$$

$$b_{i}^{(k)} = b_{i}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} \times b_{k}^{(k-1)}$$

Back Substitution

At the end of Step k=n-1

$$A^{(n-1)} = \begin{bmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \ldots\ldots\ldots\ldots\ldots a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & \ldots\ldots\ldots\ldots\ldots a_{2n}^{(1)} \\ 0 & 0 & a_{33}^{(2)} \ldots\ldots\ldots\ldots a_{3n}^{(2)} \\ . & . & \ldots\ldots\ldots\ldots \quad . \\ 0 & 0 & 0 \ldots. a_{n-1n-1}^{(n-2)} a_{n-1n}^{(n-2)} \\ 0 & 0 & 0 \ldots\ldots\ldots\ldots\ldots a_{nn}^{(n-1)} \end{bmatrix}$$

$$b^{(n-1)} = \begin{bmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(2)} \\ . \\ b_n^{(n-1)} \end{bmatrix}$$

Back substitution

$$\boxed{x_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}} \qquad x_i = \frac{b_i^{(i-1)} - \sum_{j=i+1}^{n} a_{ij}^{(i-1)} \times x_j}{a_{ii}^{(i-1)}}}$$

$$i = (n-1), (n-2), \ldots 1$$

Solution by steps of Gauss Elimination

$$3x_1 - 0.1x_2 - 0.2x_3 = 7.85 \ldots\ldots(1)$$
$$0.1x_1 + 7x_2 - 0.3x_3 = -19.3 \ \ldots\ldots(2)$$

$$0.3x_1 - 0.2x_2 + 10x_3 = 71.4 \ldots\ldots(3)$$

$$A^{(0)} = \begin{bmatrix} 3 & -0.1 & 0.2 \\ 0.1 & 7 & -0.3 \\ 0.3 & -0.2 & 10 \end{bmatrix} \Longrightarrow A^{(1)} = \begin{bmatrix} 3 & -0.1 & 0.2 \\ 0 & 7.00333 & -0.29333 \\ 0 & -0.19000 & 10.02000 \end{bmatrix} \Longrightarrow A^{(2)} = \begin{bmatrix} 3 & -0.1 & 0.2 \\ 0 & 7.00333 & -0.29333 \\ 0 & 0 & 10.01204 \end{bmatrix}$$

$$R_2 = R_2 + \alpha_{21}R_1$$
$$R_3 = R_3 + \alpha_{31}R_1$$
$$\alpha_{21} = -\frac{0.1}{3}$$
$$\alpha_{31} = -\frac{0.3}{3}$$

$$R_3 = R_3 + \alpha_{32}R_2$$
$$\alpha_{32} = -\frac{-0.19}{7.00333}$$

$$b^{(0)} = \begin{bmatrix} 7.85 \\ -19.3 \\ 71.4 \end{bmatrix} \Rightarrow b^{(1)} = \begin{bmatrix} 7.85 \\ -19.56167 \\ 70.615 \end{bmatrix} \Rightarrow b^{(2)} = \begin{bmatrix} 7.85 \\ -19.56167 \\ 70.08929 \end{bmatrix}$$

Back substitution

$$x_3 = \frac{70.084}{10.012} = 7.0$$
$$x_2 = -2.5$$
$$x_1 = 3.0$$

- $x = \begin{bmatrix} 3.0 \\ -2.5 \\ 7.0 \end{bmatrix}$

What happens if pivot element become zero or very small number

Lots of roundoff errors

$$0.0003x_1 + 3.0000x_2 = 2.0001$$

$$1.0000x_1 + 1.0000x_2 = 1.0000$$

$$x_1 = \frac{1}{3}$$

$$x_2 = \frac{2}{3}$$

$$R_2 = R_2 + \left(-\frac{1}{0.0003}\right) R_1$$

$$-9999x_2 = -6666$$

$$x_1 = \frac{2.0001 - 3(\, x_2 \,)}{0.0003}$$

Significant figure	x_2	x_1	$\lvert\varepsilon_t\rvert\%$
3	0.667	-3	1000
4	0.6667	0	100
5	0.66667	0.3	10
6	0.666667	0.33	1
7	0.6666667	0.333	0[illegible]

Solving in reverse way

$$1.0000x_1 + 1.0000x_2 = 1.0000$$

$$0.0003x_1 + 3.0000x_2 = 2.0001$$

$$x_1 = \frac{1}{3}$$

$$x_2 = \frac{2}{3}$$

Significant figure	x_2	x_1	$\lvert\varepsilon_t\rvert\%$
3	0.667	0.333	0.1
4	0.6667	0.3333	0.01
5	0.66667	0.33333	0.001
6	0.666667	0.333333	0.0001
7	0.6666667	0.3333333	0.00001

$$x_1 + x_2 + x_3 = 4$$
$$2x_1 + 2x_2 + 3x_3 = 9$$
$$3x_1 + 4x_2 + 2x_3 = 9$$

Step 1

$$x_1 + x_2 + x_3 = 4$$
$$0x_1 + 0x_2 + x_3 = 1$$
$$0x_1 + x_2 - x_3 = -3$$

$$\underset{A}{\begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 3 \\ 3 & 4 & 2 \end{bmatrix}} \quad \underset{B}{\begin{bmatrix} 4 \\ 9 \\ 9 \end{bmatrix}}$$

$R_2 - 2R_1$
$R_3 - 3R_1$

$$\underset{A}{\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & -1 \end{bmatrix}} \quad \underset{B}{\begin{bmatrix} 4 \\ 1 \\ -3 \end{bmatrix}}$$

Diagonal term
become 0 close to 0

Step 2

$$\alpha_{32} = -\frac{a_{32}}{a_{22}} = \frac{1}{0}$$

Could not conti

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Solution of Linear Equations

Pivoting, FLOPS, GJ, LU Decomposition

Prof. Jayati Sarkar

Pivoting

- p stores current pivot rows
- Big stores current pivot element
- First task to see if element below in the column is higher than the pivot element
- New largest element & its row number is stores in big & p

If the previous search ends with $p \neq k$ then switching of row required

p= k
big=$|a_{k,k}|$
FOR(ii =k+1, n)
{
dummy =$|a_{ii,k}|$
IF (dummy > big)
big =dummy
p= ii
END IF
}

IF ($p \neq k$)
FOR (jj =k, n)
{
dummy =$a_{p,jj}$
$a_{p,jj}=a_{k,jj}$
$a_{k,jj}$=dummy
}
dummy =b_p
$b_p=b_k$
b_k =dummy
END IF

Pivoting

- p stores current pivot rows
- Big stores current pivot element
- First task to see if element below in the column is higher than the pivot element
- New largest element & its row number is stores in big & p

If the previous search ends with $p \neq k$ then switching of row required

```
p= k
big=|a_{k,k}|
FOR( ii =k+1, n)
{
dummy =|a_{ii,k}|
IF (dummy > big)
big =dummy
p= ii
END IF
}
```

```
IF (p≠k)
FOR (jj =k, n)
{
dummy =a_{p,jj}
a_{p,jj}=a_{k,jj}
a_{k,jj}=dummy
}
dummy =b_p
b_p=b_k
b_k =dummy
END IF
```

FLOPS involved in Gauss Elimination

Cramer's rule $\rightarrow N^2 \times N!$ Number of operation execution time α amount of floating point operation (FLOPS)

Some quantities that facilitate operation counting

$$\sum_{i=1}^{m} Cf(i) = C\sum_{i=1}^{m} f(i) \qquad \sum_{i=1}^{m} f(i) + g(i) = \sum_{i=1}^{m} f(i) + \sum_{i=1}^{m} g(i)$$

$$\sum_{i=1}^{m} 1 = 1 + 1 + \cdots \ldots \ldots .. + 1 = m \qquad \sum_{i=k}^{m} 1 = m - k + 1$$

$$\sum_{i=1}^{m} i = 1 + 2 + 3 \ldots \ldots \ldots .. + m = \frac{m(m+1)}{2} = m^2 + \theta(m)$$

$$\sum_{i=1}^{m} i^2 = 1^2 + 2^2 + 3^2 \ldots \ldots \ldots .. + m^2 = \frac{m(m+1)(2m+1)}{2} = \frac{m^2}{3} + \theta(m^2)$$

FLOPS IN STEP k=1 for Gauss Elimination

$$A^{(1)} = \begin{bmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \cdots\cdots\cdots & a_{1n}^{(0)} \\ \boxed{0} & \boxed{\left[a_{22}^{(0)} - \frac{a_{21}^{(0)}}{a_{11}^{(0)}} \times a_{12}^{(0)}\right]} & \boxed{\cdots\cdots\cdots} & \boxed{\left[a_{2n}^{(0)} - \frac{a_{21}^{(0)}}{a_{11}^{(0)}} \times a_{1n}^{(0)}\right]} \\ 0 & \left[a_{32}^{(0)} - \frac{a_{31}^{(0)}}{a_{11}^{(0)}} \times a_{12}^{(0)}\right] & \cdots\cdots\cdots & \left[a_{3n}^{(0)} - \frac{a_{31}^{(0)}}{a_{11}^{(0)}} \times a_{1n}^{(0)}\right] \\ . & . & \cdots\cdots\cdots & . \\ 0 & \left[a_{n2}^{(0)} - \frac{a_{n1}^{(0)}}{a_{11}^{(0)}} \times a_{12}^{(0)}\right] & \cdots\cdots\cdots & \left[a_{nn}^{(0)} - \frac{a_{n1}^{(0)}}{a_{11}^{(0)}} \times a_{1n}^{(0)}\right] \end{bmatrix} \quad b^{(1)} = \begin{bmatrix} b_1^{(0)} \\ \boxed{b_2^{(0)} - \frac{a_{21}^{(0)}}{a_{11}^{(0)}} \times b_1^{(0)}} \\ b_3^{(0)} - \frac{a_{31}^{(0)}}{a_{11}^{(0)}} \times b_1^{(0)} \\ . \quad\quad . \\ b_n^{(0)} - \frac{a_{n1}^{(0)}}{a_{11}^{(0)}} \times b_1^{(0)} \end{bmatrix}$$

Calculation per row

Division $\alpha_{21} = \frac{a_{21}^{(0)}}{a_{11}^{(0)}}$	**1**	
Subtraction	LHS=n-1	RHS=1
Multiplication	LHS=n-1	RHS=1

Total Number of Addition/Subtraction per row=n-1+1=n

Total Number of Multiplication/ Division per row=1+n-1+1=n+1

In step k=1

No. of middle loops: $\sum_{i=2}^{n} 1 = n - 2 + 1 = n - 1$

Total Number of Addition/Subtraction=n×(

Total Number of Multiplication/Division=(n+1)×(n-1)

Outer Loop (k)	Middle Loop (i)	Addition/ Subtraction FLOPS	Multiplication/ Division FLOPS
1	2,n	(n-1)×n	(n-1)×(n+1)
2	3,n	(n-2)×(n-1)	(n-2)×(n)
:	:	:	:
k	k+1,n	(n-k)×(n-k+1)	(n-k)×(n-k+2)
n-1	n,n	(1)×(2)	(1)×(3)

Total addition/Subtraction flop

$$\sum_{k=1}^{n-1}(n-k)(n+1-k) = \sum_{k=1}^{n-1}[n(n+1)-k(2n+1)+k^2]$$

$$= n(n+1)\sum_{k=1}^{n-1}1-(2n+1)\sum_{k=1}^{n-1}k+\sum_{k=1}^{n-1}k^2$$

$$= n(n+1)(n-1)-\frac{(2n+1)(n-1)\,n}{2}+\frac{(n-1)n(2n-1)}{6}$$

$$= [n^3+\theta(n)]-[n^3+\theta(n)^2]+\left[\frac{1}{3}n^3+\theta(n)^2\right]$$

$$= \frac{n^3}{3}+\theta(n)$$

Total multiplication/division flop $\sum_{k=1}^{n-1}(n-k)(n+2-k) = \sum_{k=1}^{n-1} n(n+2)-k(2n+2)+k^2$

$$= n(n+2)(n-1)-\frac{(2n+2)(n-1)(n)}{2}+\frac{(n-1)(n)(2n-1)}{6}$$

$$= [n^3+\theta(n)^2]-[n^3+\theta(n)]+\left[\frac{n^3}{3}+\theta(n)^2\right] = \frac{n^3}{3}+\theta(n)^2$$

$$Summing \rightarrow \frac{2n^3}{3}+\theta(n)^2\,FLOPS$$

Back substitution

$$x_i = \frac{b_i^{(i-1)} - \sum_{j=i+1}^{n} a_{ij}^{(i-1)} \times x_j}{a_{ii}^{(i-1)}}, \quad x_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}}$$

i	Back Substitution	Multiplication+ Division	Addition/ Subtraction
n-1	$x_{n-1} = \left(b_{n-1}^{(n-2)} - a_{n-1,n}^{(n-2)} \times x_n\right) / a_{n-1,n-1}^{(n-2)}$	1+1	1
n-2	$x_{n-2} = \left(b_{n-2}^{(n-3)} - a_{n-2,n-1}^{(n-3)} \times x_{n-1} - a_{n-2,n}^{(n-3)} \times x_n\right) / a_{n-2,n-2}^{(n-3)}$	2+1	2
:		:	:
n-i		i+1	i
:		:	:
n-(n-1)		(n-1)+1	(n-1)
Total		$\boxed{1 + \sum^{1} (i+1)}$	$\boxed{\sum^{1} i}$

Total number of FLOPS in Back Substitution

Multiplication + Division

$$1+\sum_{i=n-1}^{1}(i+1)=1+\sum_{i=n-1}^{1}1+\sum_{i=n-1}^{1}i$$

$$=1+(n-1)+\frac{n(n-1)}{2}$$

$$=\frac{n(n+1)}{2}$$

Addition + Subtraction

$$\sum_{i=n-1}^{1}(i)=\frac{n(n-1)}{2}$$

Total

$$=\frac{n(n+1)}{2}+\frac{n(n-1)}{2}=n^2+O(n)$$

Elimination Back Substitution

Total number of FLOPS in Gauss Elimination

$$=\frac{2n^3}{3}+n^2+O(n^2)+O(n)=\frac{2n^3}{3}+n^2+O(n^2)$$

n	Elimination	Back Substitution	Total Flops	$2n^3/3$	Percent Due to Elimination
10	705	100	805	667	87.58%
100	671550	10000	681550	666667	98.53%
1000	6.67×10^8	1×10^6	6.68×10^8	6.67×10^8	99.85%

Variations of Gauss Elimination: Gauss Jordon

$$3x_1 - 0.1x_2 \ + 0.2x_3 = 7.85....(1)$$
$$0.1x_1 + 7x_2 - 0.3x_3 = -19.3...(2)$$
$$0.3x_1 - 0.2x_2 + 10x_3 = 71.4...(3)$$

$$A^{(0)} = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0.1 & 7 & -0.3 \\ 0.3 & -0.2 & 10 \end{bmatrix} \Rightarrow A^{*(0)} = \begin{bmatrix} 1 & -0.03333 & -0.06667 \\ 0.1 & 7 & -0.3 \\ 0.3 & -0.2 & 10 \end{bmatrix} \Rightarrow A^{(1)} = \begin{bmatrix} 1 & -0.03333 & -0.06667 \\ 0 & 7.00333 & -0.29333 \\ 0 & -0.19000 & 10.02000 \end{bmatrix}$$

Normalize Pivot row

$$\boxed{R_1 = R_1 / a_{11}}$$

$$R_2 = R_2 + \alpha_{21} R_1$$
$$R_3 = R_3 + \alpha_{31} R_1$$
$$\alpha_{21} = -0.1/1$$
$$\alpha_{31} = -0.3/1$$

$$b^{(0)} = \begin{bmatrix} 7.85 \\ -19.3 \\ 71.4 \end{bmatrix} \Rightarrow b^{*(0)} = \begin{bmatrix} 2.61667 \\ -19.3 \\ 71.4 \end{bmatrix} \Rightarrow b^{(1)} = \begin{bmatrix} 2.61667 \\ -19.56167 \\ 70.615 \end{bmatrix}$$

$$A^{(1)} = \begin{bmatrix} 1 & -0.03333 & -0.06667 \\ 0 & 7.00333 & -0.29333 \\ 0 & -0.19000 & 10.02000 \end{bmatrix}$$

Normalize Pivot row

$\boxed{R_2 = R_2 / a_{22}}$

$$A^{*(1)} = \begin{bmatrix} 1 & -0.03333 & -0.06667 \\ 0 & 1.00000 & -0.0419 \\ 0 & -0.19000 & 10.02000 \end{bmatrix}$$

$R_1 = R_1 + \alpha_{12} R_2$

$R_3 = R_3 + \alpha_{32} R_2$

$\alpha_{12} = -(-0.0333/1)$

$\alpha_{32} = -(-0.19/1)$

$$A^{(2)} = \begin{bmatrix} 1 & 0 & -0.0681 \\ 0 & 1 & -0.0419 \\ 0 & 0 & 10.0120 \end{bmatrix}$$

$$b^{(1)} = \begin{bmatrix} 2.61667 \\ -19.56167 \\ 70.615 \end{bmatrix} \Rightarrow b^{*(1)} = \begin{bmatrix} 2.61667 \\ -2.7932 \\ 70.615 \end{bmatrix} \Rightarrow b^{(2)} = \begin{bmatrix} 2.5237 \\ -2.7932 \\ 70.0843 \end{bmatrix}$$

$$A^{(2)} = \begin{bmatrix} 1 & 0 & -0.0681 \\ 0 & 1 & -0.0419 \\ 0 & 0 & 10.0120 \end{bmatrix}$$

Normalize Pivot row

$\boxed{R_3 = R_3 / a_{33}}$

$$A^{*(2)} = \begin{bmatrix} 1 & 0 & -0.0681 \\ 0 & 1 & -0.0419 \\ 0 & 0 & 1 \end{bmatrix}$$

$R_1 = R_1 + \alpha_{13} R_3$

$R_2 = R_3 + \alpha_{23} R_3$

$\alpha_{13} = -(-0.0681/1)$

$\alpha_{23} = -(-0.0419/1)$

$$A^{(3)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$b^{(2)} = \begin{bmatrix} 2.5237 \\ -2.7932 \\ 70.0843 \end{bmatrix} \Rightarrow b^{*(2)} = \begin{bmatrix} 2.5237 \\ -2.7932 \\ 7.0 \end{bmatrix} \Rightarrow b^{(3)} = \begin{bmatrix} 3.0 \\ -2.5 \\ 7.0 \end{bmatrix}$$

$$A^3 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3.000 \\ -2.5 \\ 7.0 \end{bmatrix}$$

$$I \quad \boldsymbol{x} \quad = \quad b$$

So b contains the solution of x

No need to do a back substitution

So approx. 50% more time consuming than GE

GAUSS JORDON used to find A^{-1}

Number of flops $= n^3 + n^2 - n$

$= n^3 + \theta(n^2)$

$$AA^{-1} = I$$

Steps of Gauss Jordon

$$\boxed{[A \vdots I \vdots b] \Rightarrow [I \vdots A^{-1} \vdots x]}$$

$$\boxed{[A \vdots I] \Rightarrow [I \vdots A^{-1}]}$$

$$\left[\begin{array}{ccc|ccc} 3 & -0.1 & -0.2 & 1.0 & 0.0 & 0.0 \\ 0.1 & 7 & -0.3 & 0.0 & 1.0 & 0.0 \\ 0.3 & -0.2 & 10 & 0.0 & 0.0 & 1.0 \end{array}\right]$$

$$[A][x] \Rightarrow [b]$$

$$[x] \Rightarrow [A]^{-1}[b]$$

Variations of Gauss Elimination: LU Decomposition

$$[A][x] \Rightarrow [b]$$

$$[L]\underbrace{[U][x]}_{[y]} \Rightarrow [b]$$

$$[L][y] \Rightarrow [b]$$

Forward Substitution

$$\begin{bmatrix} 1 & 0 & 0 & 0\ldots\ldots\ldots \\ -\alpha_{21} & 1 & 0 & 0\ldots\ldots\ldots \\ -\alpha_{31} & -\alpha_{32} & 1 & 0\ldots\ldots\ldots \\ -\alpha_{41} & -\alpha_{42} & -\alpha_{43} & 0\ldots\ldots\ldots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ \vdots \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ \vdots \end{bmatrix}$$

Original b value

$$y_1 = b_1$$

$$y_2 = b_2 - \alpha_{21} y_1$$

$$y_i = b_i - \sum_{j=1}^{i-1} \alpha_{ij} y_j$$

$$[U][x] = [y] \Rightarrow [x]$$

Back Substitution

$$\text{Number of steps} = \frac{N^3}{3} - \frac{N}{3} + MN^2$$

Number of b vectors to so

Variations of Gauss Elimination: LU Decomposition

$$[A][x] \Rightarrow [b]$$

$$[L]\underbrace{[U][x]}_{[y]} \Rightarrow [b]$$

$$[L][y] \Rightarrow [b]$$

Forward Substitution

$$\begin{bmatrix} 1 & 0 & 0 & 0\ldots\ldots\ldots \\ -\alpha_{21} & 1 & 0 & 0\ldots\ldots\ldots \\ -\alpha_{31} & -\alpha_{32} & 1 & 0\ldots\ldots\ldots \\ -\alpha_{41} & -\alpha_{42} & -\alpha_{43} & 0\ldots\ldots\ldots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ \vdots \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ \vdots \end{bmatrix}$$

Original b value

$$y_1 = b_1$$

$$y_2 = b_2 - \alpha_{21} y_1$$

$$y_i = b_i - \sum_{j=1}^{i-1} \alpha_{ij} y_j$$

$$[U][x] = [y] \Rightarrow [x]$$

Back Substitution

$$\text{Number of steps} = \frac{N^3}{3} - \frac{N}{3} + MN^2$$

Number of b vectors to so

Variations of Gauss Elimination: LU Decomposition

$$A^{(0)} = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0.1 & 7 & -0.3 \\ 0.3 & -0.2 & 10 \end{bmatrix} \qquad A^{(1)} = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0 & 7.00333 & -0.29333 \\ 0 & -0.19000 & 10.02000 \end{bmatrix} \qquad A^{(2)} = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0 & 7.00333 & -0.29333 \\ 0 & 0 & 10.01201 \end{bmatrix}$$

$$U = \begin{bmatrix} 3 & -0.1 & -0.2 \\ 0 & 7.00333 & -0.29333 \\ 0 & 0 & 10.01201 \end{bmatrix}$$

$$\alpha_{21} = -\frac{a_{21}^{(0)}}{a_{11}^{(0)}} = -\frac{0.1}{3}$$

$$\alpha_{31} = -\frac{a_{31}^{(0)}}{a_{11}^{(0)}} = -\frac{0.3}{3}$$

$$\alpha_{32} = -\frac{a_{32}^{(1)}}{a_{22}^{(1)}} = \frac{0.19}{7.00333}$$

$$L = \begin{bmatrix} 1 & 0 & 0 \\ -\alpha_{21} & 1 & 0 \\ -\alpha_{31} & -\alpha_{32} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0.0333 & 1 & 0 \\ 0.1 & -0.02713 & 1 \end{bmatrix}$$

$$LU = \begin{bmatrix} 1 & 0 & 0 \\ 0.0333 & 1 & 0 \\ 0.1 & -0.02713 & 1 \end{bmatrix} \begin{bmatrix} 3 & -0.1 & 0.2 \\ 0 & 7.0033 & -0.2933 \\ 0 & 0 & 10.01209 \end{bmatrix}$$

$$A = \begin{bmatrix} 3 & -0.1 & 0.2 \\ 0.0999 & 7 & -0.3 \\ 0.3 & -0.2 & 9.99996 \end{bmatrix}$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

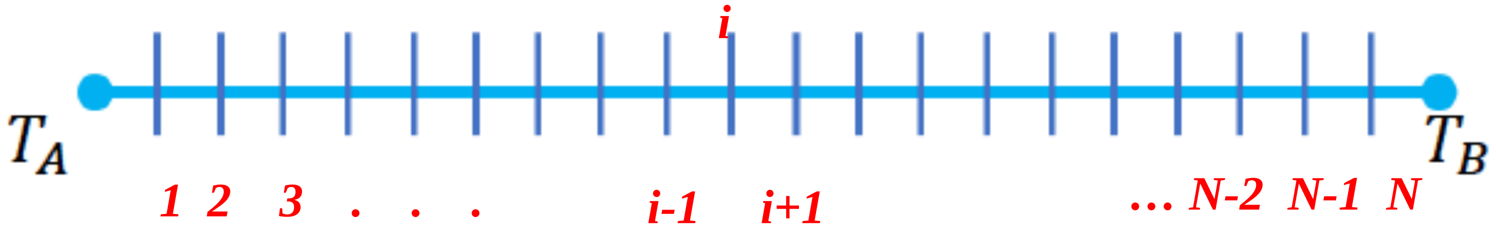
CLL-113

Solution of Linear Equations

Thomas Algorithm, Iterative Methods (GS, JI), SOR

Prof. Jayati Sarkar

Heat Condition



$$-K\frac{d^2T}{dx^2}=S$$

$$-\frac{T_{i+1}-2T_i+T_{i-1}}{\Delta x^2}=S_i^*/\mathrm{K}$$

$$i=1$$

$$-(T_2-2T_1+T_A)=S_i^*\Delta x^2/\mathrm{K} \longrightarrow -T_2+2T_1-T_A=S_i^*\Delta x^2\ /\mathrm{K}$$

$$2T_1-T_A=\frac{S_i^*}{K}\Delta x^2+T_2$$

$$i=2\ to\ N=1$$

$$-(T_{i+1}-2T_i+T_{i-1})=\frac{S_i^*}{K}\Delta x^2 \longrightarrow -T_{i+1}+2T_i-T_{i-1}=\frac{S_i^*}{K}\Delta x^2$$

$$i=\ N$$

$$-(T_B-2T_N+T_{N-1})=\frac{S_N^*}{K}\Delta x^2 \longrightarrow -T_B+2T_N-T_{N-1}=\frac{S_N^*}{K}\Delta x^2$$

$$2T_N-T_{N-1}=\frac{S_N^*}{K}\Delta x^2+T_B$$

diagonal

super diagonal

$$
\begin{bmatrix}
-2 & 1 & 0 & 0 & \cdots & 0 \\
1 & -2 & 1 & 0 & \cdots & 0 \\
0 & 1 & -2 & 1 & \cdots & 0 \\
0 & 0 & 0 & 1 & -2 & 1 \\
0 & 0 & 0 & 0 & 1 & -2
\end{bmatrix}
\begin{bmatrix}
T_1 \\ T_2 \\ \vdots \\ T_{N-1} \\ T_N
\end{bmatrix}
=
\begin{bmatrix}
-\dfrac{S_1}{K}\Delta x^2 - T_A \\
-\dfrac{S_2}{K}\Delta x^2 \\
-\dfrac{S_3}{K}\Delta x^2 \\
-\dfrac{S_{N-1}}{K}\Delta x^2 \\
-\dfrac{S_N}{K}\Delta x^2 - T_B
\end{bmatrix}
$$

sub diagonal

Matrices can have banded structure also for GENERAL matrix

FLOPS~N^3	N	FLOPS
	10	1000
	100	10,00000
	1000	100000000 0
	In TDMA FLOPS~N	

Banded Matrix Structure::::Tri-diagonal Matrix

Thomas Algorithm

$$\begin{array}{r} \\ \mathbf{0} \rightarrow \\ \\ \\ \\ \\ \\ \end{array}
\begin{bmatrix} b_1 & c_1 & 0 & 0 & 0 & 0 & 0 \\ a_2 & \boxed{b_2} & c_2 & 0 & 0 & 0 & 0 \\ 0 & a_3 & b_3 & c_3 & 0 & 0 & 0 \\ 0 & 0 & a_4 & b_4 & c_4 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\ 0 & 0 & 0 & 0 & 0 & a_N & b_N \end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \\ x_{N-1} \\ x_N \end{bmatrix}
=
\begin{bmatrix} d_1 \\ \boxed{d_2} \\ d_3 \\ d_4 \\ \vdots \\ d_{N-1} \\ d_N \end{bmatrix}$$

Thomas Algorithm

Step k=1

$$
\left[A^{(0)}\right],\left[d^{(0)}\right]
\begin{bmatrix}
b_1 & c_1 & 0 & 0 & 0 & 0 & 0 \\
a_2 & b_2 & c_2 & 0 & 0 & 0 & 0 \\
0 & a_3 & b_3 & c_3 & 0 & 0 & 0 \\
0 & 0 & a_4 & b_4 & c_4 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\
0 & 0 & 0 & 0 & 0 & a_N & b_N
\end{bmatrix},
\begin{bmatrix}
d_1 \\ d_2 \\ d_3 \\ d_4 \\ \vdots \\ d_{N-1} \\ d_N
\end{bmatrix}
$$

$$c_1' = \frac{c_1}{b_1}, d_1' = \frac{d_1}{b_1}$$

2 Multiplications/Divisions

1. Normalize pivot row with pivot element

$$
\left[A^{*(0)}\right],\left[d^{*(0)}\right]
\begin{bmatrix}
1 & c_1' & 0 & 0 & 0 & 0 & 0 \\
a_2 & b_2 & c_2 & 0 & 0 & 0 & 0 \\
0 & a_3 & b_3 & c_3 & 0 & 0 & 0 \\
0 & 0 & a_4 & b_4 & c_4 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\
0 & 0 & 0 & 0 & 0 & a_N & b_N
\end{bmatrix},
\begin{bmatrix}
d_1' \\ d_2 \\ d_3 \\ d_4 \\ \vdots \\ d_{N-1} \\ d_N
\end{bmatrix}
$$

2. Elimination Step R2=R2-R1*a2

$$
\left[A^{(1)}\right],\left[d^{(1)}\right]
\begin{bmatrix}
1 & c_1' & 0 & 0 & 0 & 0 & 0 \\
0 & b_2' & c_2 & 0 & 0 & 0 & 0 \\
0 & a_3 & b_3 & c_3 & 0 & 0 & 0 \\
0 & 0 & a_4 & b_4 & c_4 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\
0 & 0 & 0 & 0 & 0 & a_N & b_N
\end{bmatrix},
\begin{bmatrix}
d_1' \\ d_2' \\ d_3 \\ d_4 \\ \vdots \\ d_{N-1} \\ d_N
\end{bmatrix}
$$

$$b_2' = b_2 - c_1' a_2$$
$$d_2' = d_2 - d_1' a_2$$

2 Multiplications/Divisions

2 Additions/Subtractions

Number of FLOPs involved in step k=1 is [illegible]6
No middle loop

Thomas Algorithm

Step k=2

$$
\left[A^{(1)}\right],\left[d^{(1)}\right]
\begin{bmatrix}
1 & c_1' & 0 & 0 & 0 & 0 & 0 \\
0 & b_2' & c_2 & 0 & 0 & 0 & 0 \\
0 & a_3 & b_3 & c_3 & 0 & 0 & 0 \\
0 & 0 & a_4 & b_4 & c_4 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\
0 & 0 & 0 & 0 & 0 & a_N & b_N
\end{bmatrix},
\begin{bmatrix}
d_1' \\ d_2' \\ d_3 \\ d_4 \\ \vdots \\ d_{N-1} \\ d_N
\end{bmatrix}
$$

1. Normalize pivot row with pivot element

$$
c_2' = \frac{c_2}{b_2'},\ d_2'' = \frac{d_2'}{b_2'}
$$

2 Multiplications/Divisions

$$
\left[A^{*(1)}\right],\left[d^{*(1)}\right]
\begin{bmatrix}
1 & c_1' & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & c_2' & 0 & 0 & 0 & 0 \\
0 & a_3 & b_3 & c_3 & 0 & 0 & 0 \\
0 & 0 & a_4 & b_4 & c_4 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\
0 & 0 & 0 & 0 & 0 & a_N & b_N
\end{bmatrix},
\begin{bmatrix}
d_1' \\ d_2'' \\ d_3 \\ d_4 \\ \vdots \\ d_{N-1} \\ d_N
\end{bmatrix}
$$

2. Elimination Step R3=R3-R2*a3

$$
\left[A^{*(2)}\right],\left[d^{*(2)}\right]
\begin{bmatrix}
1 & c_1' & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & c_2' & 0 & 0 & 0 & 0 \\
0 & 0 & b_3' & c_3 & 0 & 0 & 0 \\
0 & 0 & a_4 & b_4 & c_4 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\
0 & 0 & 0 & 0 & 0 & a_N & b_N
\end{bmatrix},
\begin{bmatrix}
d_1' \\ d_2'' \\ d_3' \\ d_4 \\ \vdots \\ d_{N-1} \\ d_N
\end{bmatrix}
$$

$$
b_3' = b_3 - c_2' a_3
$$

$$
d_3' = d_3 - d_2'' a_3
$$

2 Multiplications/Divisions

2 Additions/Subtractions

Number of FLOPs involved in step k=2 is [illegible]6

No middle loop

Thomas Algorithm

After Step k=N-1

$$
\left[A^{(N-1)}\right],\left[d^{(N-1)}\right]
\begin{bmatrix}
1 & c_1' & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & c_2' & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & c_3' & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & c_4' & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 1 & c_{N-1}' \\
0 & 0 & 0 & 0 & 0 & 0 & b_N'
\end{bmatrix},
\begin{bmatrix}
d_1' \\ d_2'' \\ d_3'' \\ d_4'' \\ \vdots \\ d_{N-1}'' \\ d_N'
\end{bmatrix}
$$

Number of FLOPs involved in Elimination=6×(N-1)

Backward Sweep

$$x_N = \frac{d_N'}{b_N'}$$

Number of FLOPs involved : 1 Div

$$x_i = d_i'' - c_i' \times x_{i+1}$$

Number of FLOPs involved : (1 Mult+ 1 Subtraction)×(N-1)

i=N-1, N-2, …1

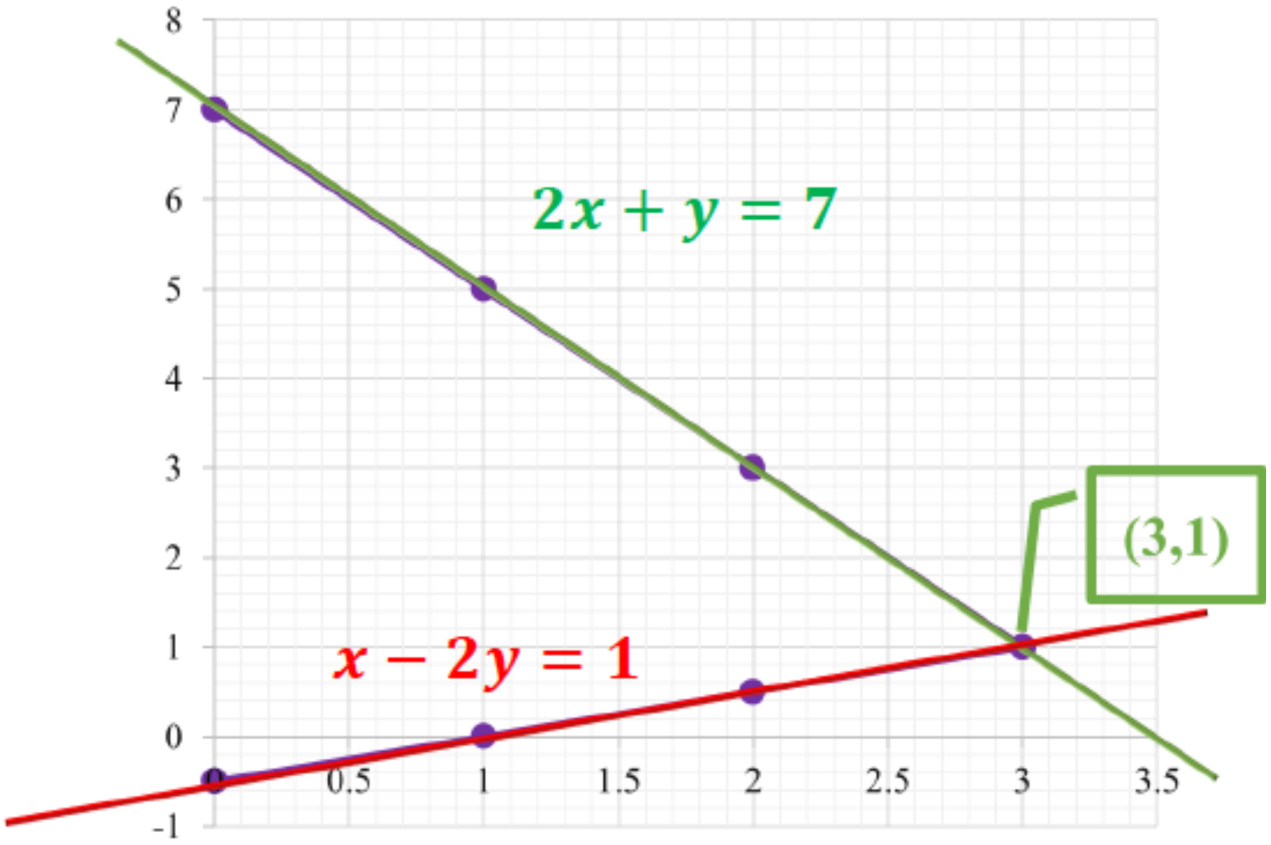
Total Number of FLOPs involved in Thomas Algorithm~6×(N-1)+1+2×(N-1)~8N<<<N^3

Iterative Solvers

$$2x + y = 7 \qquad x - 2y = 1$$

$$x = \frac{7 - y}{2} \qquad\qquad x = 1 + 2y$$

$$y = 7 - 2x \qquad\qquad y = \frac{-(1 - x)}{2}$$



$$x - 2y = 1$$
$$2x + y = 7$$

Start with initial guess

$$(x^0, y^0) = (0,0)$$

$$x^{i+1} = 1 + 2y^i \qquad y^{i+1} = 7 - 2x^{i+1}$$

i	x^{i+1}	y^{i+1}
initial guess	0	0
0	1	5
1	11	-15
2	-29	65
3	131	-255

Diverging

Start with initial guess

$$(x^0, y^0) = (0,0)$$

$$2x + y = 7$$

$$x - 2y = 1$$

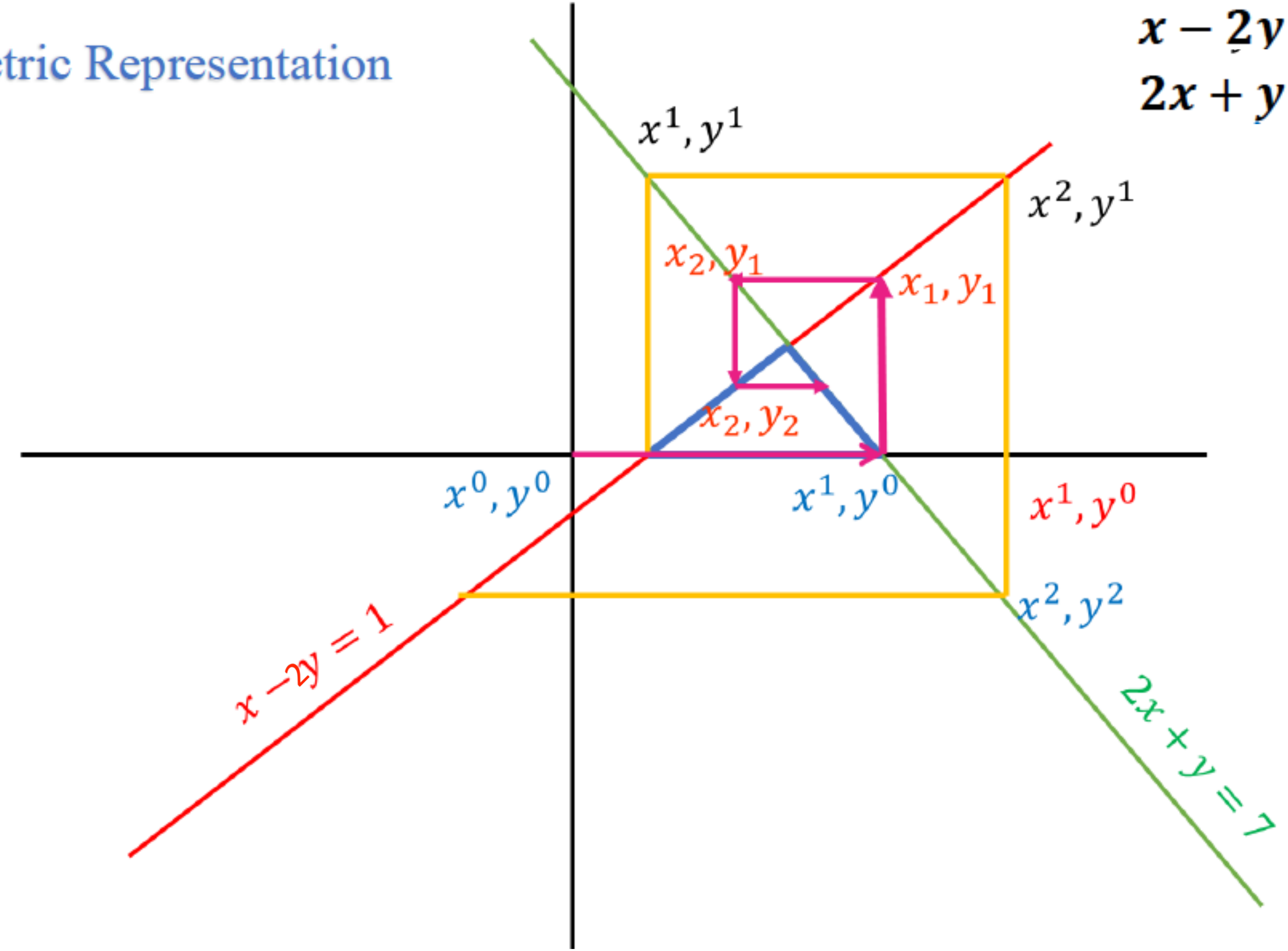
$$\left| \frac{x_j^{(N)} - x_j^{(N-1)}}{x_j^{(N)}} \right| < \varepsilon_{tolerance} \quad for \quad \forall j$$

$$x^{i+1} = 3.5 - 0.5y^i \qquad y^{i+1} = 0.5x^{i+1} - 0.5$$

i	$x^{i+1}=x_1$	$y^{i+1}=x_2$
initial guess	0	0
0	3.5	1.25
1	2.875	0.9375
2	3.03125	1.015625
3	2.992	0.996
4	3.001	1.009

Geometric Representation

$$x - 2y = 1$$
$$2x + y = 7$$



Diagonal dominance

The path which will succeed depends on diagonal dominance.

$$A = \begin{bmatrix} 1 & -2 \\ 2 & 1 \end{bmatrix} \quad A = \begin{bmatrix} 2 & 1 \\ 1 & -2 \end{bmatrix}$$

Diagonal dominance

$$|a_{11}| \geq \sum_{\substack{i=2 \\ i \neq 1}}^{n} |a_{1i}|$$

$$|a_{22}| \geq \sum_{\substack{i=1 \\ i \neq 2}}^{n} |a_{2i}|$$

$$|a_{ii}| \geq \sum_{\substack{i=1 \\ i \neq j}}^{n} |a_{ij}|$$

$$Ax = b$$

- $$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n = b_1$$
  $$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \ldots + a_{2n}x_n = b_2$$
  $$.$$
  $$.$$
  $$a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \ldots + a_{nn}x_n = b_n$$
- $$x_1 = \frac{b_1 - (a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n)}{a_{11}}$$
- $$x_2 = \frac{b_2 - (a_{21}x_1 + a_{23}x_3 + \ldots + a_{2n}x_n)}{a_{22}}$$

$$Generalization \quad x_i = \frac{b_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij}\, x_j}{a_{ii}}$$

Sub Diagonal element

Super Diagonal element

- $$\color{red}{Gauss\ Siedel} \quad x_i^k = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}\, x_j^k - \sum_{j=i+1}^{n} a_{ij}\, x_j^{k-1}}{a_{ii}}$$
- $$\color{red}{Jacobi\ iteration} \quad x_i^k = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}\, x_j^{k-1} - \sum_{j=i+1}^{n} a_{ij}\, x_j^{k-1}}{a_{ii}} = \frac{b_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij}\, x_j^{k-1}}{a_{ii}}$$

Improvement of convergence under relaxation

- Slight enhancement of *Gauss Siedel* method to increase convergence.
- After each step:
  - $x_i^{new} = \lambda x_i^{new} + (1-\lambda)x_i^{old}$
  - λ is a weighting factor between 0 and 2.
  - If λ=0, old value is kept
  - λ=1, new value is accepted

$0 \leq \lambda \leq 1$
weighted average b/w old and new value
UNDER RELAXATION

- It makes a nonconvergent system convergence by damping out oscillations.
- For $1 < \lambda \leq 2$ Extra weight is present on the present value.
  - There is an implicit assumption that the way in which we are moving is true.
  - Improve the estimate by pushing it closed to the truth.
  - OVER RELAXTATION
  - Help converge faster in an already convergent system.
  - Successive /simultaneous over relaxation.

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Solution of non-Linear Equations

Bracketing Methods

Prof. Jayati Sarkar

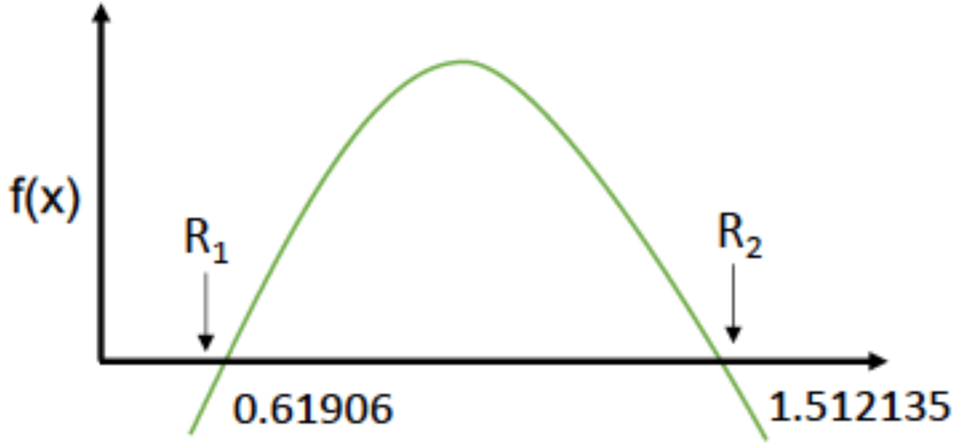
Solution of Non-linear Equation

- f(x) = 0; f(x) is not a polynomial equation
- It is often termed as transcedental function having an analytical form.

Example: f(x) = $Cos(x) - xe^x$

- It cannot be expressed as a finite sequence of algebraic operations.

$$f(x) = x - \frac{e^x}{3} = 0$$



- Aim is to find the points of intersection of the curve with the x-axis.
- Solutions/ Numerically solving for such equations
- Graphical methods fails for variables > 2
- Cannot be solved analytically for complex cases
- Unlike linear case → Possible to have finite number of multiple solutions.
  - Rank |A| ~ n : Unique solution
  - Rank (A) ≠ n : Infinite no of solutions.

General setup: Let x be a variable of interest. The objective is to find the value of x, which satisfies the non-linear equation f(x) = 0

Examples:

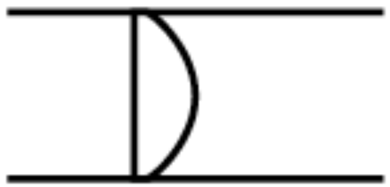
1. van der Waals equation of state for gases

$$\boxed{\left(P + \frac{a}{v^2}\right)(v - b) = RT}$$

- Interested to find out molar volume occupied by particular gas for a particular P/T combination so that appropriate container vessel can be selected.
- Can be tested how good it conforms to ideal gas law.

2. Friction/resistance of flow in pipes: Friction factor(Non – dimensional number)

$$\boxed{0 = \frac{1}{\sqrt{f}} + 2\log\left(\frac{\varepsilon}{3.7D} + \frac{2.51}{Re\sqrt{f}}\right)}$$



ε = roughness of pipe

D = diameter of pipe

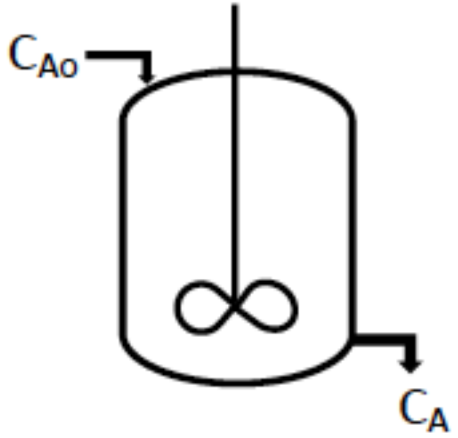
$Re = \frac{Dv\rho}{\mu}$; Where,

ρ = density
D = diameter
v = average velocity
μ = viscosity

3. CSTR catalytic reaction : $\boxed{\dfrac{C_{Ao}}{\tau} - \dfrac{C_A}{\tau} - \dfrac{KC_A}{(1+KC_A)^2} = 0}$

τ = residence time

Non – linear kinetics



Extension to multivariable cases

$X_1, X_2, X_3 \ldots\ldots\ldots X_n$

$f_1(X_1, X_2, \ldots\ldots\ldots X_n) = 0$

$f_2(X_1, X_2, \ldots\ldots\ldots X_n) = 0$

.

.

.

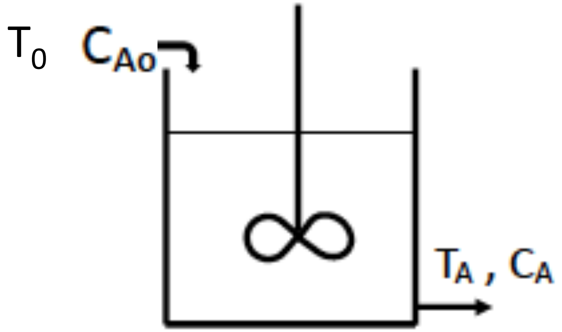
$f_3(X_1, x_2, x_3, \ldots\ldots X_n) = 0$

$$\underline{x} = \begin{Bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{Bmatrix} \qquad \underline{f}(\underline{x}) = \begin{Bmatrix} f_1(\underline{x}) \\ f_2(\underline{x}) \\ \cdot \\ \cdot \\ \cdot \\ f_n(\underline{x}) \end{Bmatrix}$$

$$\boxed{\underline{f}(\underline{x}) = 0}$$

Example:

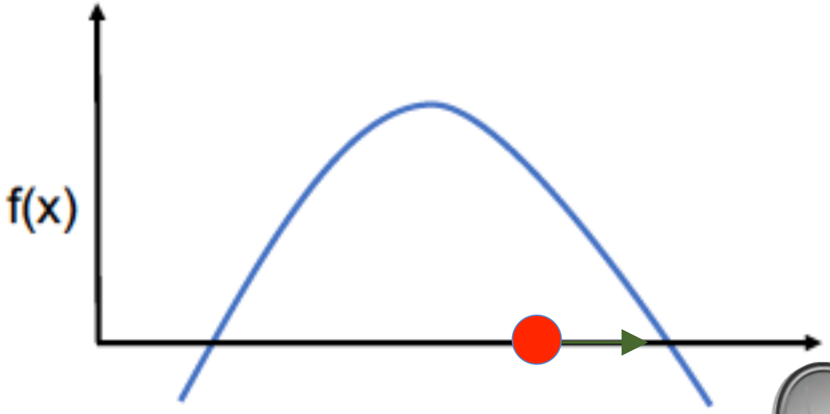
Adiabatic CSTR



- multivariable problem in C_A and T
- Propane combustion in excess O_2
- Mass balance: $\frac{C_{Ao}}{\tau} - \frac{C_A}{\tau} - Koe^{-\left(\frac{E}{RT}\right)} C_A{}^{2.00} = 0$ $\Big\}$ $f_1(C_A,T) = 0$
- Energy balance: $\frac{T_o}{\tau} - \frac{T}{\tau} - \frac{\Delta H}{\rho Cp}\left[K_o e^{-\left(\frac{E}{RT}\right)} C_A{}^{2.00}\right] = 0$ $\Big\}$ $f_2(C_A,T) = 0$

General strategy:

- Start with one or two initial guess.
- Move the initial guess towards the solution with a strategy.
- Verify if stopping criteria is satisfies.
- If satisfies ⇒ that's your solution.



Bracketing methods / Open methods

Bracketing methods:

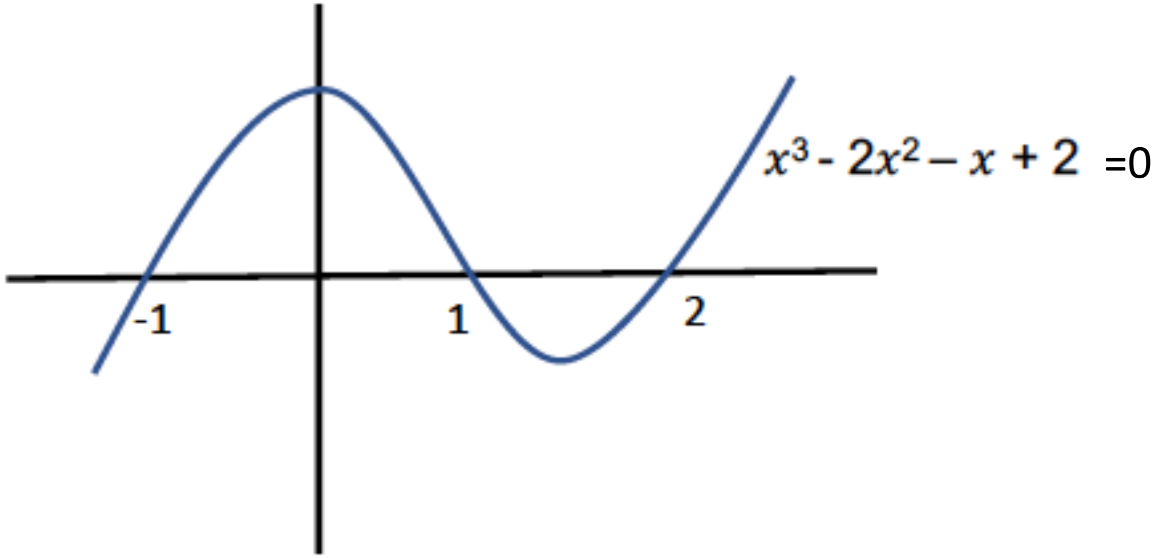
- Bisection method
- Regula Falsi

Open methods:

- Secant
- Fixed point iteration
- Newton Raphson method

Root finding:

All root of a polynomial



Bisection method

1. Two guesses one towards the left and one towards the right
    - f(x^l) > 0, f(x^r) < 0
    - Initial guess such that $[f^l f^r < 0]$

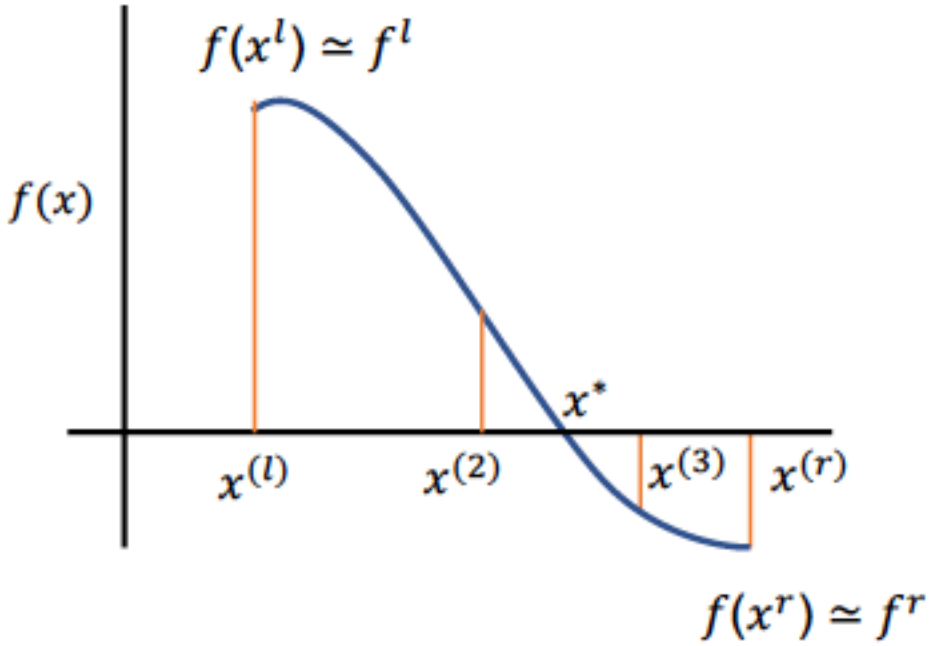
2. Move towards $x^{\overset{(i+1)}{(2)}} = \dfrac{x^l + x^r}{2}$

3. If the solution has converged
    - $\left|x^{(2)} - x^{(l)}\right| < \varepsilon \qquad \varepsilon \sim 10^{-4}$
    - $|f(x^2)| < \delta \;\; \Rightarrow$ if yes solution has reached

$\Big\downarrow$ If not

4. Verify bracketing
    - If $f^l f^{(2)} < 0$ then $x^{(r)} = x^{(2)}$
    - if not $x^{(l)} = x^{(2)}$

5. Go back to 2



Regula Falsi Method

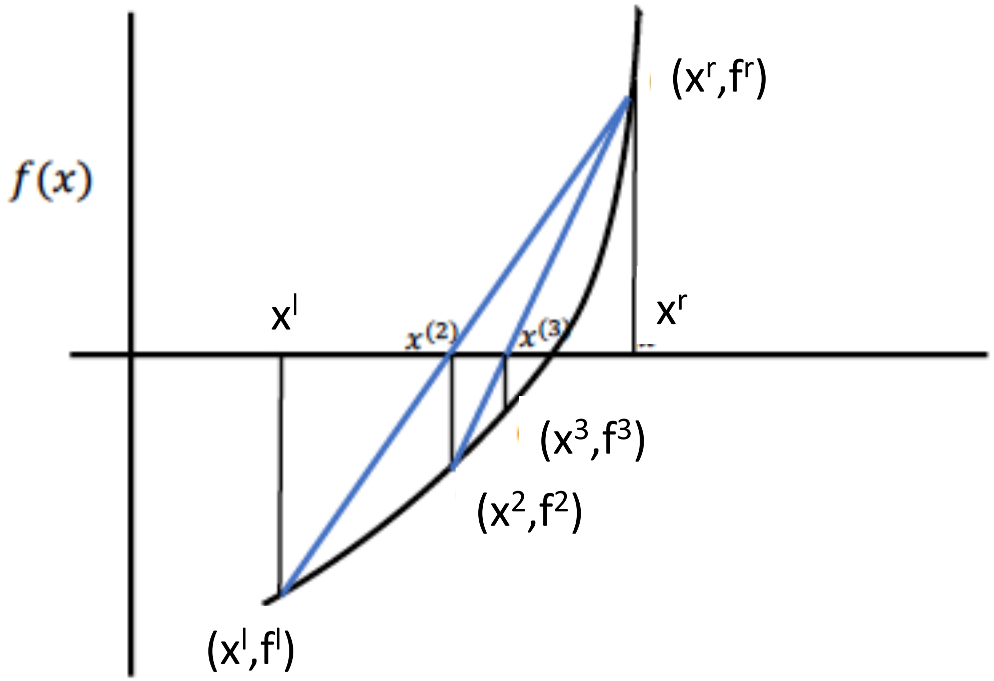
Straight line connecting f^l & f^r

$$\frac{y-f^l}{x-x^l}=\frac{f^r-f^l}{x^r-x^l}$$

This intersects the x axis

$$\frac{0-f^l}{x^{i+1}-x^l}=\frac{f^r-f^l}{x^r-x^l}$$

$$x^{i+1}=x^l-f^l\frac{\left(x^r-x^l\right)}{\left(f^r-f^l\right)}$$



1. Initial guess x^l & x^r such tha $f^l \times f^r < 0$.
2. Evaluate

$$x^{i+1} = x^l - f^l \frac{\left(x^r - x^l\right)}{\left(f^r - f^l\right)}$$

3. Check if solution has converges

or $\left|x^{i+1} - x^l\right| < \varepsilon^{tol}$

if yes $x^{sol} = x^{i+1}$

4. If not verify bracketing if $f^l f^{i+1} > 0$

$$x^l = x^{i+1}$$

if not : $x^r = x^{i+1}$

5. Go back to 2.

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Solution of non-Linear Equations

Open Methods

Prof. Jayati Sarkar

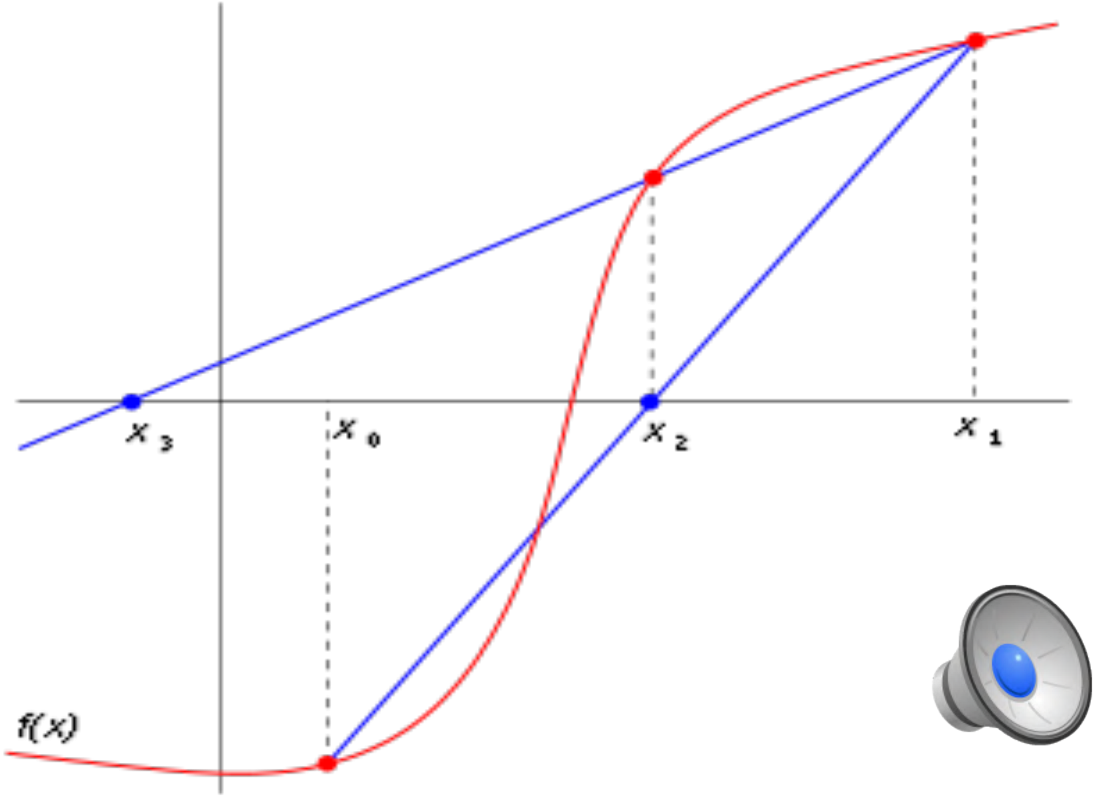
Secant Method

$$x^{i+1} = x_l - f_l \frac{(x_r - x_l)}{(f_r - f_l)}$$

Check

$$f(x^{i+1})f(x^i) < 0$$

In secant method we are not interested whether new solution brackets the solution or not.



1. Initial guesses x^0& x^1

2. $x^{i+1}= x^i - f^i \frac{x^i - x^{i-1}}{f^i - f^{i-1}}$

3. Check if $|x^{i+1} - x^l| < \varepsilon^{tol}$

if yes $x^{sol} = x^{i+1}$

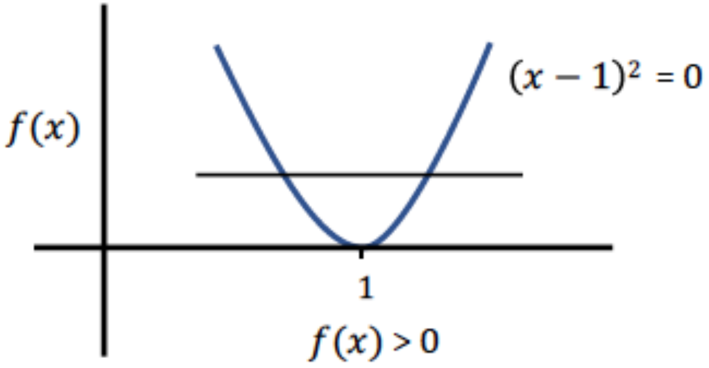
if no $\Rightarrow$ go back to step 2

Secant Method			f(x)=x-e^x/3		x^0=1	x^1=2	
					tol=1*10-4		
Itn	x^i-1	x^i	f^i-1	f^i	x^(i+1)=x^i-f^i*(x^i-x^i-1)/(f^i-f^i-1)	f^(i+1)	|x(i-1)-x(i+1)|
1	1	2	0.093906057	-0.4630187	1.16861534	0.096103886	0.16861534
2	2	1.16861534	-0.4630187	0.096103886	1.311516555	0.074250358	0.688483445
3	1.16861534	1.311516555	0.096103886	0.074250358	1.79704301	-0.213552036	0.62842767
4	1.311516555	1.79704301	0.074250358	-0.213552036	1.436777893	0.03440517	0.125261338
5	1.79704301	1.436777893	-0.213552036	0.03440517	1.486766287	0.012509487	0.310276723
6	1.436777893	1.486766287	0.03440517	0.012509487	1.515325761	-0.001642036	0.078547868
7	1.486766287	1.515325761	0.012509487	-0.001642036	1.512011934	6.27852E-05	0.025245647
8	1.515325761	1.512011934	-0.001642036	6.27852E-05	1.512133976	2.94813E-07	0.003191785
9	1.512011934	1.512133976	6.27852E-05	2.94813E-07	1.512134552	-5.33749E-11	0.000122617
10	1.512133976	1.512134552	2.94813E-07	-5.33749E-11	1.512134552	0	5.75656E-07
11	1.512134552	1.512134552	-5.33749E-11	0	1.512134552	0	1.0422E-1[illegible]

In bisection method the two initial guesses will be impossible to be found as $f^l f^r > 0$

- SECANT METHOD doesn't have a problem as it doesn't go for bracketing
- Problem with secant method is however if the line joining x_l & x_r is parallel to

  x –axis $\rightarrow f^i - f^{i-1} = 0$



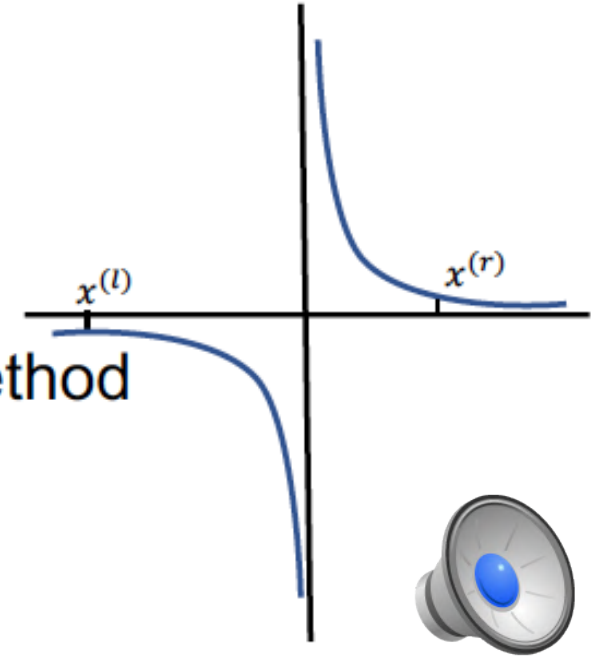
$\boxed{f(x) = \frac{1}{x}}$

$$x \rightarrow 0^+ \ldots +\propto$$
$$\rightarrow 0^- \ldots -\propto$$

$\downarrow$

Doesn't have solution, Bisection method will work

as x→0, $f(x^l)f(x^r) \rightarrow$ will find out discontinuity by bisection method



Fixed Point Iteration

$$x = g(x)$$

$$Recasted\ form: x - g(x) = f(x) = 0$$

$$x - \frac{e^x}{3} = 0$$

$$g(x) = \frac{e^x}{3} \qquad g(x) = ln(3x)$$

1. Start with 1 initial *guess*
2. Strategy $x^{i+1} = g\left(x^i\right)$
3. Verify if $\left|x^{i+1} - x^i\right| < \varepsilon^{tol}$

$$\updownarrow$$

yes $\Rightarrow$ solution

Fixed Point Iteration Method

f(x)=x-e^x/3

x^0=1

tol=1*10-4

g(x)=Exp(x)/3

iteration Number	x^i	g(x^i)	Error=|x^i1-x^i-1|
0	1	0.906093943	
1	0.906093943	0.824879185	0.093906057
2	0.824879185	0.760535032	0.081214758
3	0.760535032	0.713140191	0.064344153
4	0.713140191	0.680129473	0.047394841
5	0.680129473	0.658044437	0.033010718
6	0.658044437	0.643670808	0.022085035
7	0.643670808	0.634485097	0.014373629
8	0.634485097	0.628683586	0.009185712
9	0.628683586	0.625046831	0.005801511
10	0.625046831	0.622777817	0.003636755
11	0.622777817	0.621366327	0.002269014
12	0.621366327	0.620489894	0.00141149
13	0.620489894	0.619946314	0.000876433
14	0.619946314	0.619609416	0.00054358
15	0.619609416	0.619400705	0.000336899
16	0.619400705	0.619271443	0.00020871
17	0.619271443	0.6191914	0.000129262
18	0.6191914	0.61914184	8.0043E-05
19	0.61914184	0.619111156	4.956E-05
20	0.619111156	0.61909216	3.06839E-05
21	0.61909216	0.619080399	1.89965E-05

g(x)=LN(3x)

iteration Number	x^i	g(x^i)	Error=|x^i1-x^i-1|
0	1	1.098612289	
1	1.098612289	1.192660116	0.098612289
2	1.192660116	1.274798493	0.094047828
3	1.274798493	1.34140041	0.082138377
4	1.34140041	1.392326439	0.066601917
5	1.392326439	1.429588334	0.050926029
6	1.429588334	1.455998813	0.037261895
7	1.455998813	1.474304423	0.026410479
8	1.474304423	1.48679859	0.01830561
9	1.48679859	1.4952375	0.012494167
10	1.4952375	1.500897346	0.00843891
11	1.500897346	1.504675448	0.005659846
12	1.504675448	1.507189515	0.003778103
13	1.507189515	1.508858957	0.002514066
14	1.508858957	1.509965996	0.001669442
15	1.509965996	1.51069942	0.001107039
16	1.51069942	1.511185024	0.000733424
17	1.511185024	1.511506416	0.000485604
18	1.511506416	1.511719069	0.000321392
19	1.511719069	1.511859748	0.000212653
20	1.511859748	1.511952803	0.000140679
21	1.511952803	1.512014351	9.30548E-0

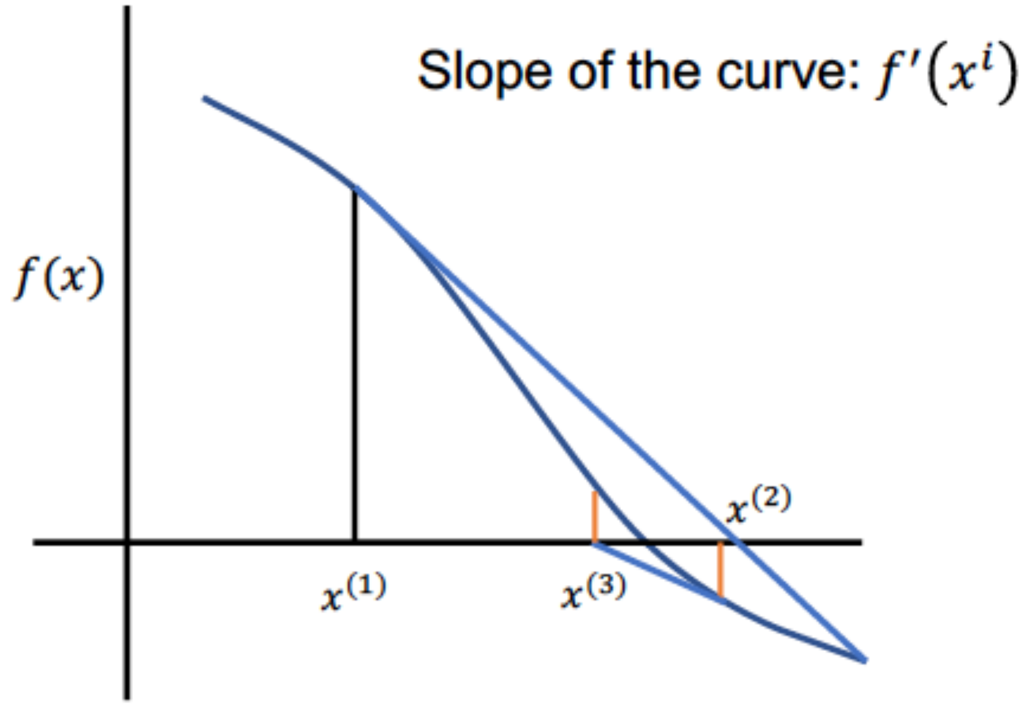
Newton Raphson Method

$$y - f(x^i) = f'(x^i).[x - x^i]$$

Intersects x-axis at $(x^{i+1}, 0)$

$$0 - f(x^i) = f'(x^i).[x^{i+1} - x^i]$$

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



1. Start with one initial guess.

2. Strategy $x^{i+1} = x^i - \frac{f(x^i)}{f'(x^i)}$

3. Check if $\left|x^{i+1} - x^i\right| < \varepsilon^{tol}$ $\rightarrow$ if yes $x^{i+1} = x_{sol}$

Speed of iteration: $NR > Secant\ or\ Regula\ Falsi > Fixed\ point + Bisection$

Newton Raphson Iteration Method			f(x)=x-e^x/3	f`(x)=1-e^x/3	tol 1*10^-4
iteration Number	x^i	f(x^i)	f`(x^i)	x^i+1	Error=|x^i1-x^i-1|
0	2	-0.4630187	-1.4630187	1.683518263	0.316481737
1	1.683518263	-0.111303955	-0.794822218	1.543481972	0.140036291
2	1.543481972	-0.016804879	-0.560286851	1.513488623	0.029993349
3	1.513488623	-0.000694853	-0.514183476	1.51213725	0.001351373
4	1.51213725	-1.38198E-06	-0.512138632	1.512134552	2.69846E-06
5	1.512134552	-5.5056E-12	-0.512134552	1.512134552	1.07503E-11
6	1.512134552	0	-0.512134552	1.512134552	0

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Solution of non-Linear Equations

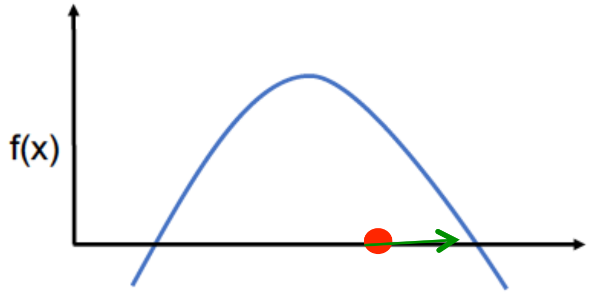
Error Estimation/Convergence

Prof. Jayati Sarkar

1. Start with initial guess /guesses

2. $$x^{i+1} = \frac{x^l + x^r}{2} \rightarrow \text{Bisection method}$$

$$x^{i+1} = g(x^i) \rightarrow \text{Fixed point Iteration}$$



$$x^{i+1} = x^i - \frac{f(x^i)}{f'(x^i)} \rightarrow \text{NR}$$

$$x^{i+1} = x^i - f^i \frac{x_i - x_{i-1}}{f_i - f_{i-1}} \rightarrow \text{RF/ Secant}$$

3. Stopping criteria $\boxed{\left|x^{i+1} - x^i\right| < \varepsilon_{tol} \ \ \textbf{or,} \ \left|x^{i+1} - x^l\right| < \varepsilon_{tol}}$

Convergence

$$E_t^i = \left|x^i - x^t\right|$$

As i increases $x^i \rightarrow$ should go to true solution

$$E^i = \left|x^i - x^{i-1}\right|$$

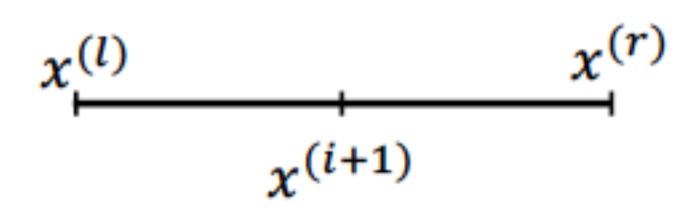
$$\boxed{E^{i+1} \rightarrow E^i \rightarrow E^{i-1} \ldots\ldots \rightarrow E^1 \rightarrow E^0}$$

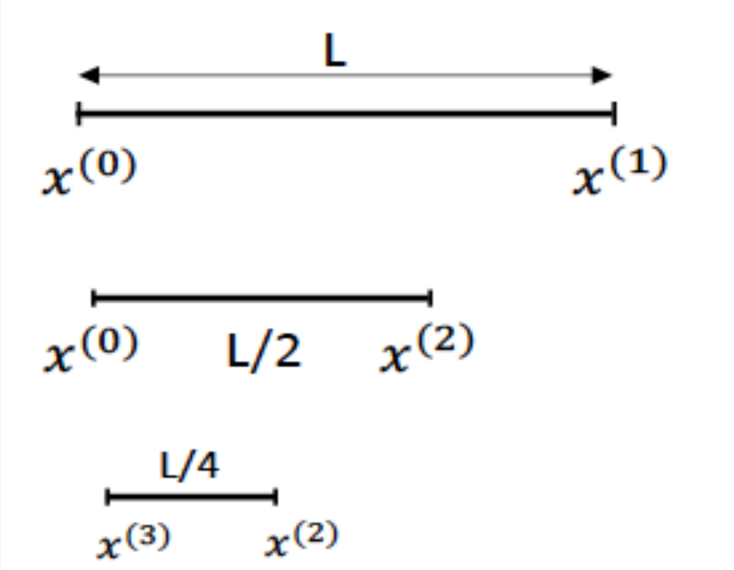
Bisection Method

$$x^{i+1} = \frac{x^l + x^r}{2}$$

$$E^{i+1} = \left|x^{i+1} - x^i\right|$$

$$= \left|\frac{x^l + x^r}{2} - x^l\right| = \left|\frac{x^r - x^l}{2}\right|$$





iteration:0

iteration:1

iteration:2

$$E^0 = L$$

$$E^1 = \frac{L}{2}$$

$$E^2 = \frac{L}{2^2}$$

$$E^i = \frac{L}{2^i}$$

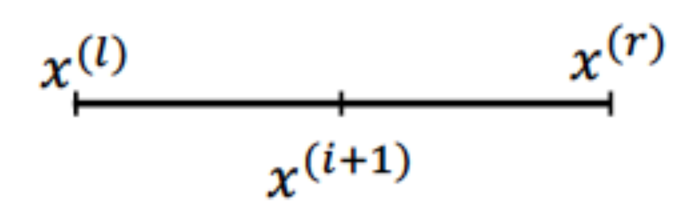
$$E^{i+1} = \frac{E^i}{2}$$

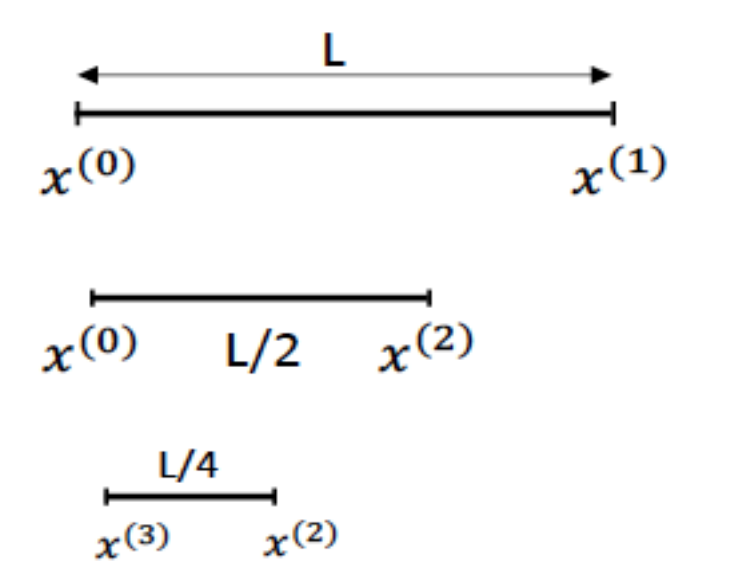
Bisection Method

$$x^{i+1} = \frac{x^l + x^r}{2}$$

$$E^{i+1} = \left|x^{i+1} - x^i\right|$$

$$= \left|\frac{x^l + x^r}{2} - x^l\right| = \left|\frac{x^r - x^l}{2}\right|$$





iteration:0

iteration:1

iteration:2

$$E^0 = L$$

$$E^1 = \frac{L}{2}$$

$$E^2 = \frac{L}{2^2}$$

$$E^i = \frac{L}{2^i}$$

$$E^{i+1} = \frac{E^i}{2}$$

Rate of Convergence

$E^{i+1} = \propto (E^i)^\eta$ η →determines the order of convergence

Bisection Method: $E^{i+1} = \frac{E^i}{2}$ $\alpha = \frac{1}{2}, \eta = 1$ *Linear Rate of Convergence*

Newton Raphson Method $\eta = 2$ *Quadratic Rate of Convergence*

Regula Falsi/Secant Method $1 < \eta < 2$ *Non-Linear Rate of Convergence*

Bisection Method:

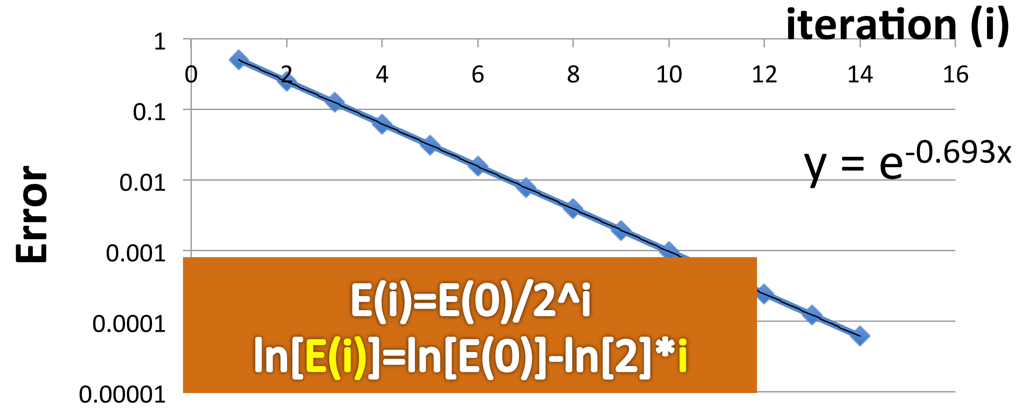
Given ε_{tol} → how many iterations are required for convergence can also be found out

$$E^i < \varepsilon_{tol}$$

$$E^i = \frac{L}{2^i} < \varepsilon_{tol}$$

$$i = \frac{\ln\left(\frac{L}{\varepsilon_{tol}}\right)}{\ln 2}$$

Bisection Method			f(x)=x-e^x/3		x^l=1	x^r=2	tol=1*10-4	
Itn	x^l	x^r	f^l	f^r	x^(i+1)=0.5(x^l +x^r)	f^(i+1)	f^l f^(i+1)	Error
1	1	2	0.093906057	-0.4630187	1.5	0.006103643	0.000573169	0.5
2	1.5	2	0.006103643	-0.4630187	1.75	-0.168200892	-0.001026638	0.25
3	1.5	1.75	0.006103643	-0.168200892	1.625	-0.067806346	-0.000413866	0.125
4	1.5	1.625	0.006103643	-0.067806346	1.5625	-0.027744394	-0.000169342	0.0625
5	1.5	1.5625	0.006103643	-0.027744394	1.53125	-0.010067718	-6.14498E-05	0.03125
6	1.5	1.53125	0.006103643	-0.010067718	1.515625	-0.001796801	-1.0967E-05	0.015625
7	1.5	1.515625	0.006103643	-0.001796801	1.5078125	0.002199369	1.34242E-05	0.0078125
8	1.5078125	1.515625	0.002199369	-0.001796801	1.51171875	0.000212816	4.6806E-07	0.00390625
9	1.51171875	1.515625	0.000212816	-0.001796801	1.513671875	-0.000789104	-1.67934E-07	0.001953125
10	1.51171875	1.513671875	0.000212816	-0.000789104	1.512695313	-0.000287423	-6.11681E-08	0.000976563
11	1.51171875	1.512695313	0.000212816	-0.000287423	1.512207031	-3.71233E-05	-7.90042E-09	0.000488281
12	1.51171875	1.512207031	0.000212816	-3.71233E-05	1.511962891	8.78913E-05	1.87046E-08	0.000244141
13	1.511962891	1.512207031	8.78913E-05	-3.71233E-05	1.512084961	2.53953E-05	2.23202E-09	0.00012207
14	1.512084961	1.512207031	2.53953E-05	-3.71233E-05	1.512145996	-5.86119E-06	-1.48846E-10	6.10352E-05



Number of iterations required to reach desired accuracy

$$i = \frac{ln\left(\frac{L}{\varepsilon_{tol}}\right)}{ln2}$$

$$i = \frac{\ln\left(\frac{1}{1\times10^{-4}}\right)}{\ln(2)} = 13.2877 \approx 14$$

Fixed Point Iteration (FPI)

$x = g(x)$

If $\bar{x}$ is the solution

$$\bar{x} = g(\bar{x}) \quad \ldots\ldots(1)$$

$$g'(\xi) = \frac{g(\bar{x}) - g(x^i)}{(\bar{x} - x^i)}$$

$$x^{i+1} = g\left(x^i\right) \ldots(2) \quad \text{FPI}$$

$$\bar{x} - x^{i+1} = g(\bar{x}) - g\left(x^i\right) = g'(\xi)[\bar{x} - x^i] \rightarrow \text{Mean value theorem} \quad (\bar{x} < \xi < x_i)$$

$$E^{(i+1)} = g'(\xi)E^{(i)} \underbrace{= \alpha E^{(i)} = \alpha^2 E^{i-1} = \ldots\ldots = \alpha^i E^1}_{\text{Linear rate of convergence}}$$

$\propto = g'(\xi)$

Guaranteed to convergence,

if $|\propto| < 1$

$-1 < \propto < 1$

$$x - \frac{e^x}{3} = 0$$

$$g(x) = \frac{e^x}{3}$$

$$g'(x) = \frac{e^x}{3}$$

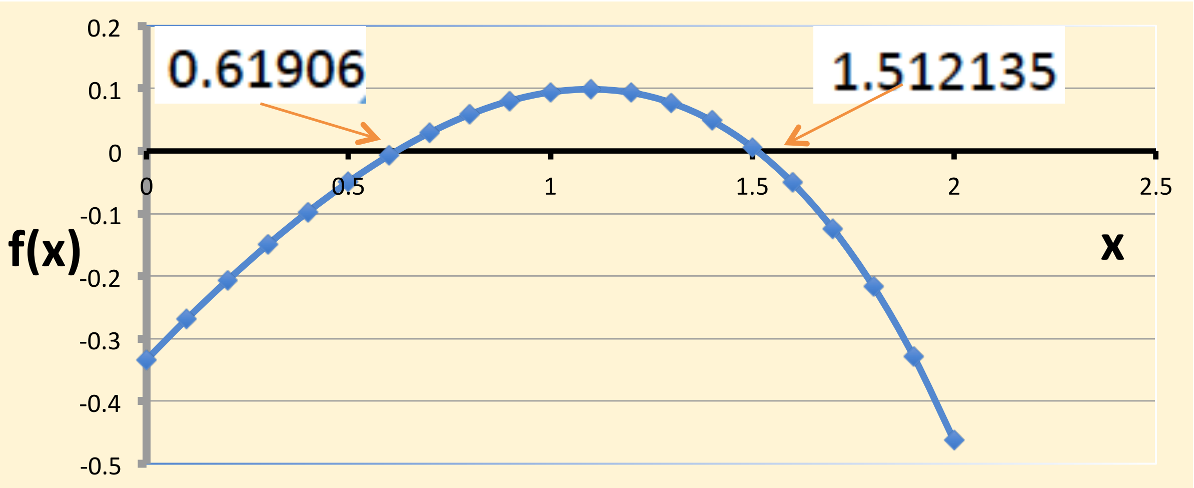
If $e^x > 3$

$x > ln3$

$x > 1.0986$

In reality starts diverging at $x = 1.52$

$g'(x) > 1$ is sufficient, not necessary condition



iteration Number	x^i	g(x^i)	Error=|x^i1-x^i-1|
0	1.0986	0.999987711	
1	0.999987711	0.906082808	0.098612289
2	0.906082808	0.82487	0.093904903
3	0.82487	0.760528047	0.081212808
4	0.760528047	0.71313521	0.064341953
5	0.71313521	0.680126085	0.047392837
6	0.680126085	0.658042208	0.033009125
7	0.658042208	0.643669373	0.022083877
8	0.643669373	0.634484186	0.014372835
9	0.634484186	0.628683013	0.009185187
10	0.628683013	0.625046473	0.005801173
11	0.625046473	0.622777594	0.00363654
12	0.622777594	0.621366189	0.002268879
13	0.621366189	0.620489808	0.001411405
14	0.620489808	0.619946261	0.000876381
15	0.619946261	0.619609383	0.000543547
16	0.619609383	0.619400685	0.000336878
17	0.619400685	0.61927143	0.000208698
18	0.61927143	0.619191392	0.000129254
19	0.619191392	0.619141835	8.00382E-05

iteration Number	x^i	g(x^i)	Error=|x^i1-x^i-1|
0	1.52	1.524075065	
1	1.524075065	1.530298442	0.004075065
2	1.530298442	1.539851762	0.006223377
3	1.539851762	1.55463295	0.00955332
4	1.55463295	1.577782944	0.014781189
5	1.577782944	1.614734675	0.023149994
6	1.614734675	1.675518026	0.036951731
7	1.675518026	1.7805205	0.060783351
8	1.7805205	1.977647904	0.105002474
9	1.977647904	2.408575792	0.197127404
10	2.408575792	3.706038451	0.43[illegible]7888
11	3.706038451	13.5640941	1.2[illegible]59
12	13.5640941	259232.8096	9.[illegible]4
13	259232.8096	#NUM!	259219.2455

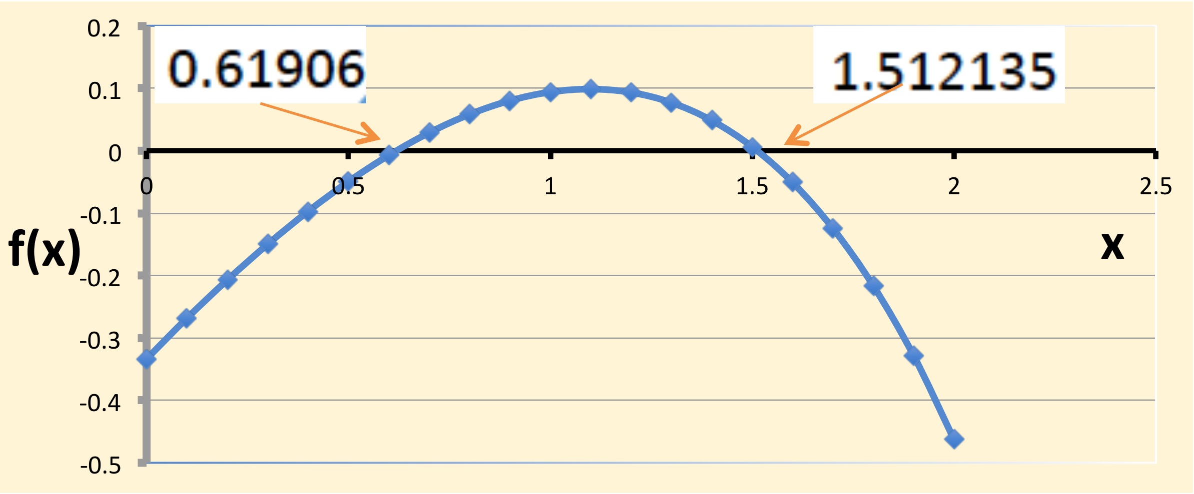
$$g(x) = ln(3x)$$

$$g'(x) = \frac{1}{x}$$

For $x < 1$, $g'(x) > 1$ | Method doesn't work

In reality starts diverging at

$x < 0.62$

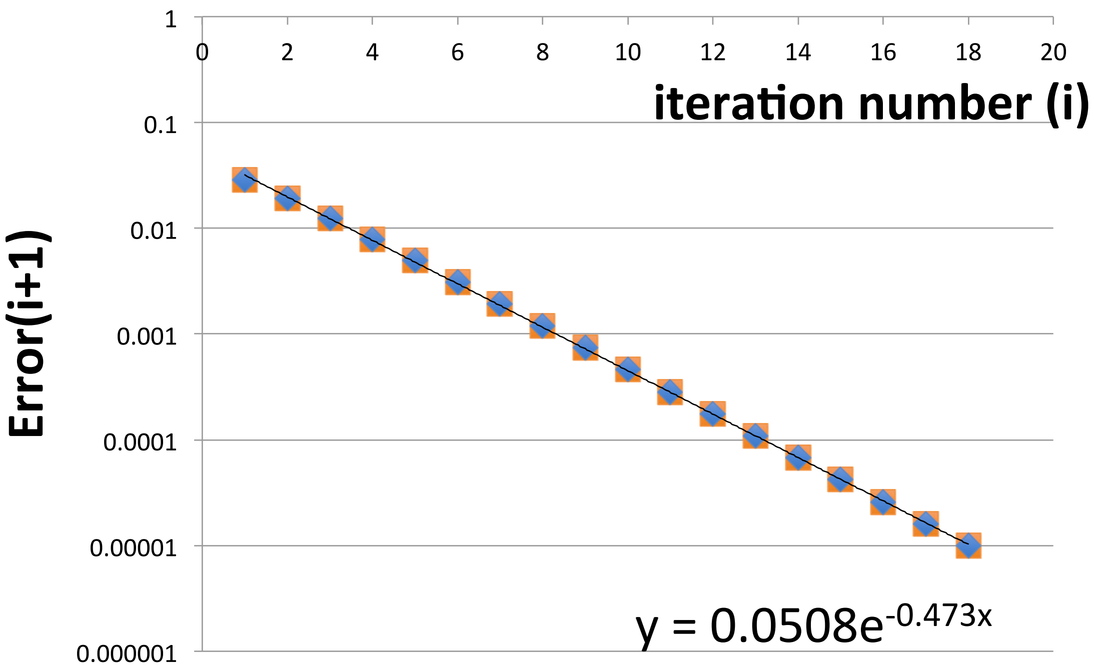


iteration Number	x^i	g(x^i)	Error=|x^i1-x^i-1|
0	1	1.098612289	
1	1.098612289	1.192660116	0.098612289
2	1.192660116	1.274798493	0.094047828
3	1.274798493	1.34140041	0.082138377
4	1.34140041	1.392326439	0.066601917
5	1.392326439	1.429588334	0.050926029
6	1.429588334	1.455998813	0.037261895
7	1.455998813	1.474304423	0.026410479
8	1.474304423	1.48679859	0.01830561
9	1.48679859	1.4952375	0.012494167
10	1.4952375	1.500897346	0.00843891
11	1.500897346	1.504675448	0.005659846
12	1.504675448	1.507189515	0.003778103
13	1.507189515	1.508858957	0.002514066
14	1.508858957	1.509965996	0.001669442
15	1.509965996	1.51069942	0.001107039
16	1.51069942	1.511185024	0.000733424
17	1.511185024	1.511506416	0.000485604
18	1.511506416	1.511719069	0.000321392
19	1.511719069	1.511859748	0.000212653
20	1.511859748	1.511952803	0.000140679
21	1.511952803	1.512014351	9.30548E-05

iteration Number	x^i	g(x^i)	Error=|x^i1-x^i-1|
0	0.619	0.618962282	
1	0.618962282	0.618901347	3.77176E-05
2	0.618901347	0.618802895	6.0935E-05
3	0.618802895	0.618643807	9.84519E-05
4	0.618643807	0.618386685	0.000159088
5	0.618386685	0.617970975	0.000257123
6	0.617970975	0.617298499	0.00041571
7	0.617298499	0.616209708	0.000672475
8	0.616209708	0.614444351	0.001088791
9	0.614444351	0.611575375	0.001765357
10	0.611575375	0.606895219	0.002868976
11	0.606895219	0.599213165	0.004680156
12	0.599213165	0.586474412	0.007682054
13	0.586474412	0.564986049	0.012738753
14	0.564986049	0.527658048	0.021488363
15	0.527658048	0.459305447	0.037328001
16	0.459305447	0.32057246	0.0683526
17	0.32057246	-0.039034656	0.13873298
18	-0.039034656	#NUM!	0.359607116

Fixed Point Iteration: Rate of Convergence

g(x)=Exp(x)/3			
iteration Number	x^i	g(x^i)	Error=|x^i1-x^i-1|
0	0.7	0.671250902	
1	0.671250902	0.652227804	0.028749098
2	0.652227804	0.639937678	0.019023099
3	0.639937678	0.632120897	0.012290125
4	0.632120897	0.627199008	0.007816781
5	0.627199008	0.624119588	0.004921889
6	0.624119588	0.622200619	0.003079419
7	0.622200619	0.621007779	0.00191897
8	0.621007779	0.620267458	0.001192839
9	0.620267458	0.619808431	0.000740321
10	0.619808431	0.619523988	0.000459027
11	0.619523988	0.619347793	0.000284444
12	0.619347793	0.619238677	0.000176195
13	0.619238677	0.619171112	0.000109116
14	0.619171112	0.619129279	6.75652E-05



$$\ln[E^{(i+1)}] = \ln[E^1] + i \times \ln\alpha$$

$$\begin{aligned} \alpha &= g'(\xi) \\ &\approx g'(\bar{x}) \\ &\approx 0.6192 \\ \ln(0.6192) &\approx -0.4793 \end{aligned}$$

Derivation of Newton Raphson Method

Current guess: x^i

Taylor Series Expansion around x^i

$$f(x) = f(x^i) + (x - x^i) f'(x^i) + \frac{(x - x^i)^2}{2!} f''(x^i) + \ldots$$

Assumptions: $f(x^{i+1}) = 0,\ x^{i+1} \approx x^i$

$$0 = f(x^i) + (x^{i+1} - x^i) f'(x^i)$$

$$\boxed{x^{i+1} = x^i - \frac{f(x^i)}{f'(x^i)}}$$

Error Estimation: Newton Raphson Method

Taylor Series of true value $\bar{x}$

$$0 = f\left(x^i\right) + \left(\bar{x} - x^i\right) f'\left(x^i\right) + \frac{\left(\bar{x} - x^i\right)^2}{2!} f''\left(x^i\right) + \ldots$$

$$0 = f\left(x^i\right) + E\left(x^i\right) f'\left(x^i\right) + \frac{E\left(x^i\right)^2}{2!} f''\left(x^i\right) + \ldots$$

$$0 = f\left(x^i\right) + E\left(x^i\right) f'\left(x^i\right) + \frac{E\left(x^i\right)^2}{2!} f''\left(\xi\right) \quad (1)$$

Mean Value Theorem

$$x^i < \xi < \bar{x}$$

Newton Raphson Method

$$0 = f\left(x^i\right) + \left(x^{i+1} - x^i\right) f'\left(x^i\right)$$

$$0 = f\left(x^i\right) + \left(x^{i+1} - \bar{x} + \bar{x} - x^i\right) f'\left(x^i\right)$$

$$0 = f\left(x^i\right) + \left(-E(x^{i+1}) + E(x^i)\right) f'\left(x^i\right) \quad \ldots\ldots(2)$$

Error Estimation: Newton Raphson Method

Taylor Series of true value $\bar{x}$

$$0 = f\left(x^i\right) + \left(\bar{x} - x^i\right) f'\left(x^i\right) + \frac{\left(\bar{x} - x^i\right)^2}{2!} f''\left(x^i\right) + \ldots$$

$$0 = f\left(x^i\right) + E\left(x^i\right) f'\left(x^i\right) + \frac{E\left(x^i\right)^2}{2!} f''\left(x^i\right) + \ldots$$

Mean Value Theorem

$$x^i < \xi < \bar{x}$$

$$0 = f\left(x^i\right) + E\left(x^i\right) f'\left(x^i\right) + \frac{E\left(x^i\right)^2}{2!} f''\left(\xi\right) \quad (1)$$

Newton Raphson Method

$$0 = f\left(x^i\right) + \left(x^{i+1} - x^i\right) f'\left(x^i\right)$$

$$0 = f\left(x^i\right) + \left(x^{i+1} - \bar{x} + \bar{x} - x^i\right) f'\left(x^i\right)$$

$$0 = f\left(x^i\right) + \left(-E(x^{i+1}) + E(x^i)\right) f'\left(x^i\right) \quad \ldots\ldots(2)$$

Error Estimation: Newton Raphson Method

Subtracting Eqn. 1 and Eqn. 2 we get:

$$E\left(x^{i+1}\right) = -\frac{E\left(x^{i}\right)^{2}}{2!}\,\frac{f''(\xi)}{f'\left(x^{i}\right)} \qquad x^{i} < \xi < \bar{x}$$

$$E\left(x^{i+1}\right) \approx -\frac{E\left(x^{i}\right)^{2}}{2!}\,\frac{f''\left(x^{i}\right)}{f'\left(x^{i}\right)}$$

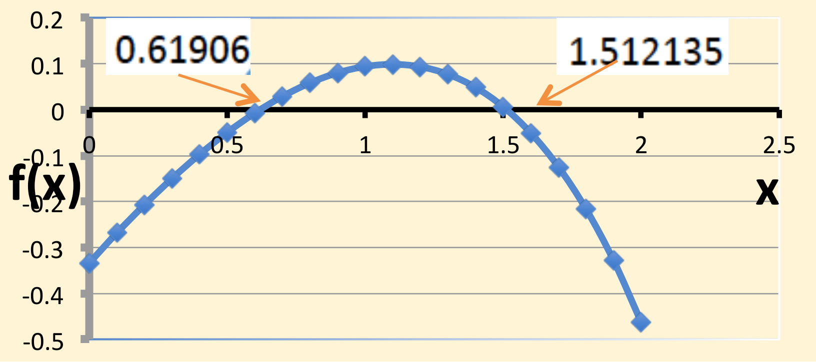
$$E\left(x^{i+1}\right) \approx \left[-\frac{f''\left(x^{i}\right)}{2!\,f'\left(x^{i}\right)}\right] E\left(x^{i}\right)^{2}$$

η

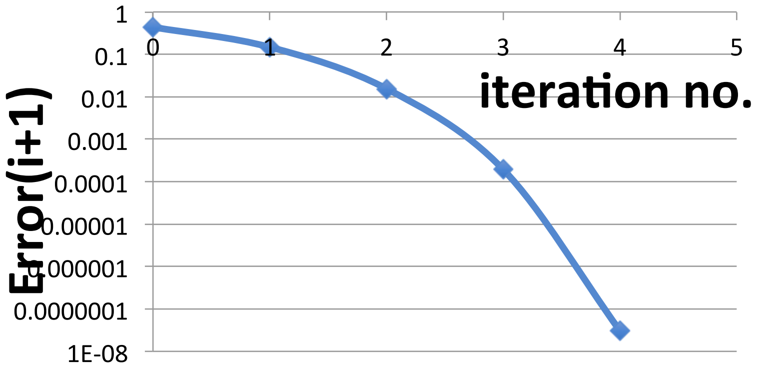
α

Rate of convergence is **QUADRATIC**

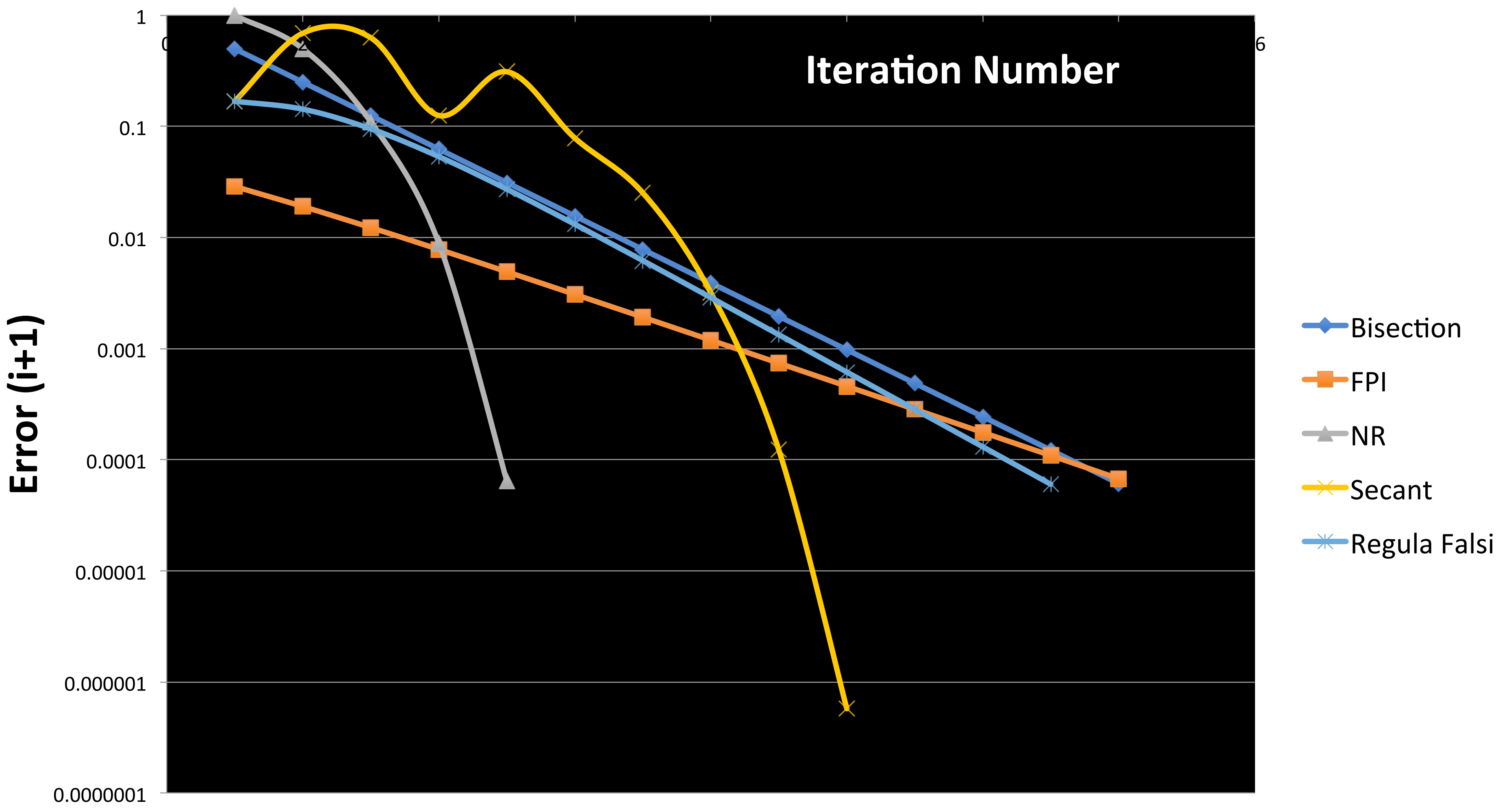
iteration Number	x^i	f(x^i)	f`(x^i)	x^i+1	Error=|x^i+1-x^i|
0	0.9	0.080132296	0.180132296	0.455147534	0.444852466
1	0.455147534	-0.070321113	0.474531354	0.60333819	0.148190656
2	0.60333819	-0.006065658	0.390596153	0.61886742	0.015529231
3	0.61886742	-7.38629E-05	0.381058717	0.619061256	0.000193836
4	0.619061256	-1.16283E-08	0.380938732	0.619061287	3.05254E-08



Newton Raphson Method			f(x)=x-e^x/3	f`(x)=1-e^x/3	tol 1*10^-4
Itn. No	x^i	f(x^i)	f`(x^i)	x^i+1	Error=|x^i+1-x^i|
0	1.2	0.093294359	-0.106705641	2.074315156	0.874315156
1	2.074315156	-0.578716129	-1.653031285	1.724221281	0.350093876
2	1.724221281	-0.14516277	-0.869384051	1.557249308	0.166971973
3	1.557249308	-0.024667085	-0.581916393	1.51485991	0.042389398
4	1.51485991	-0.001401371	-0.516261281	1.512145449	0.002714461
5	1.512145449	-5.58108E-06	-0.51215103	1.512134552	1.08973E-05



Convergence Rate of different Methods



NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

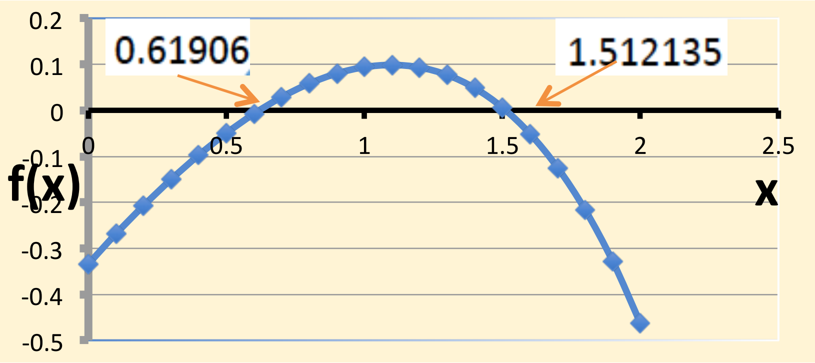
Solution of non-Linear Equations

NR/Multivariable case

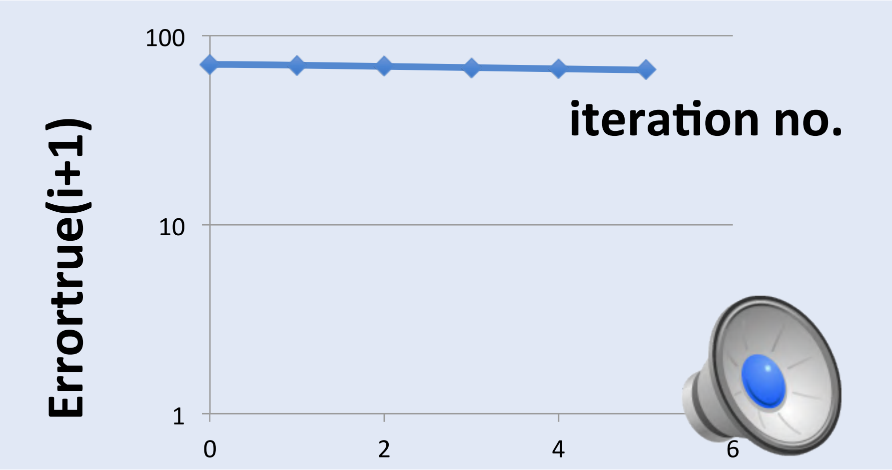
Prof. Jayati Sarkar

Problem Areas: Newton Raphson Method

$$x^{i+1} = x^{i} - \frac{f\left(x^{i}\right)}{f'\left(x^{i}\right)}$$



Newton Raphson Iteration Method			f(x)=x-e^x/3	f`(x)=1-e^x/3	tol 1*10^-4
iteration Number	x^i	f(x^i)	f`(x^i)	x^i+1	Error=|x^i+1-x^t|
0	1.1	0.098611325	-0.001388675	72.11110792	70.59897337
1	72.11110792	-6.92365E+30	-6.92365E+30	71.11110792	69.59897337
2	71.11110792	-2.54707E+30	-2.54707E+30	70.11110792	68.59897337
3	70.11110792	-9.37014E+29	-9.37014E+29	69.11110792	67.59897337
4	69.11110792	-3.44708E+29	-3.44708E+29	68.11110792	66.59897337
5	68.11110792	-1.26811E+29	-1.26811E+29	67.11110792	65.59897337



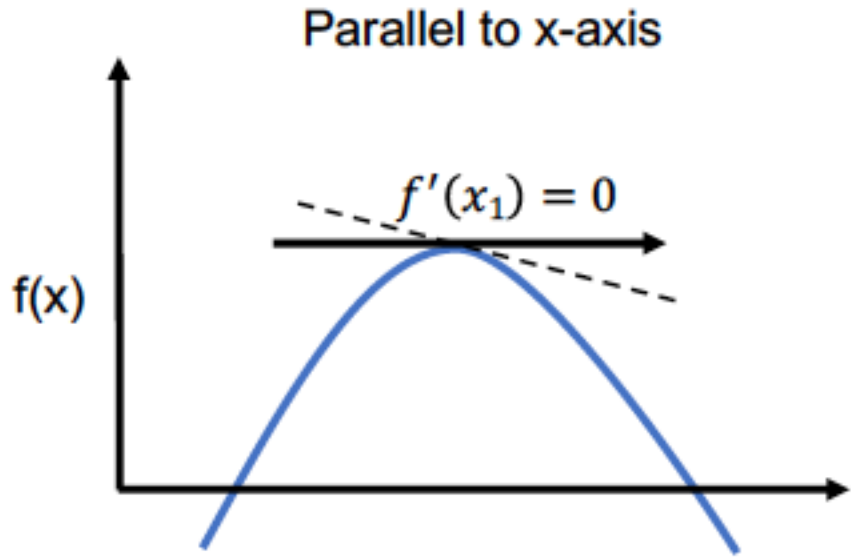
Problem Areas: Newton Raphson Method

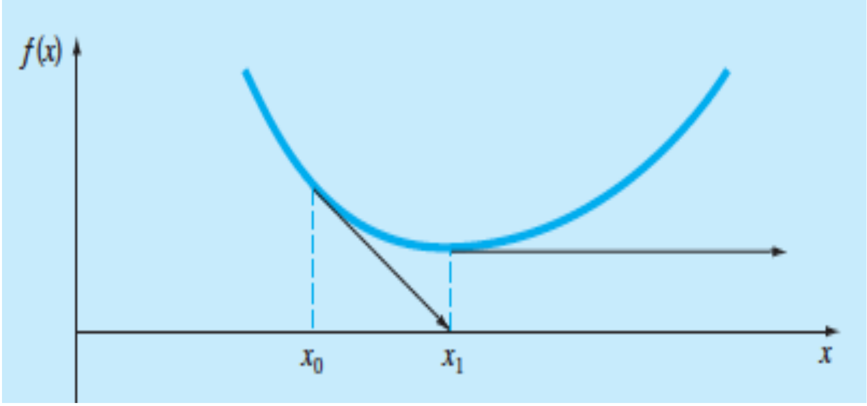
$$\boxed{f(x)=0}$$

$$\boxed{x^{(i+1)} = x^{(i)} - \frac{f\left(x^{(i)}\right)}{f'\left(x^{(i)}\right)}}$$

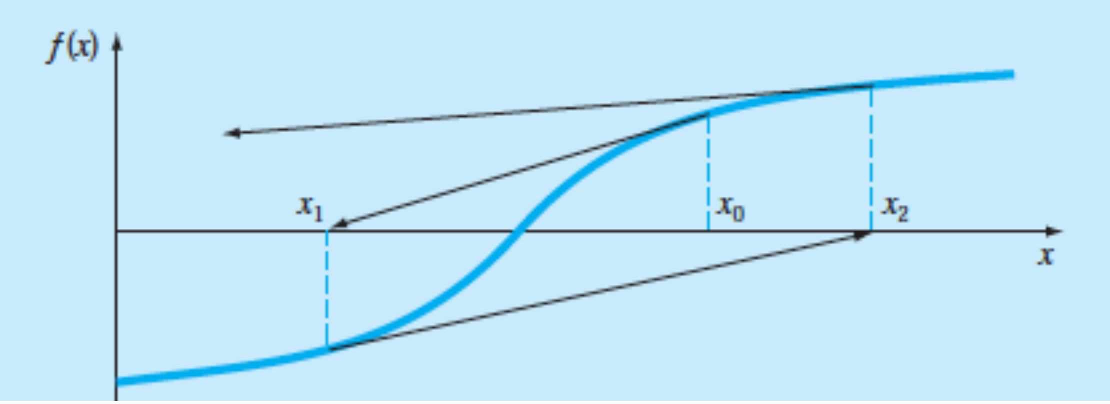
$$if \quad f'\left(x^{(i)}\right)=0$$

not possible to move forward

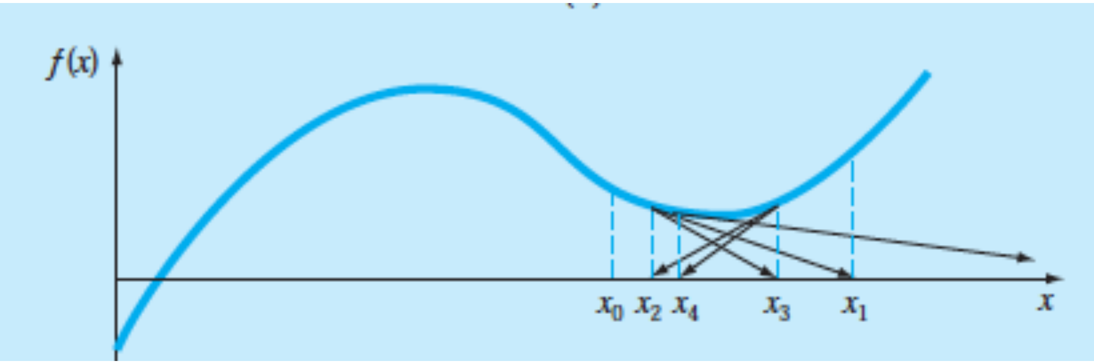




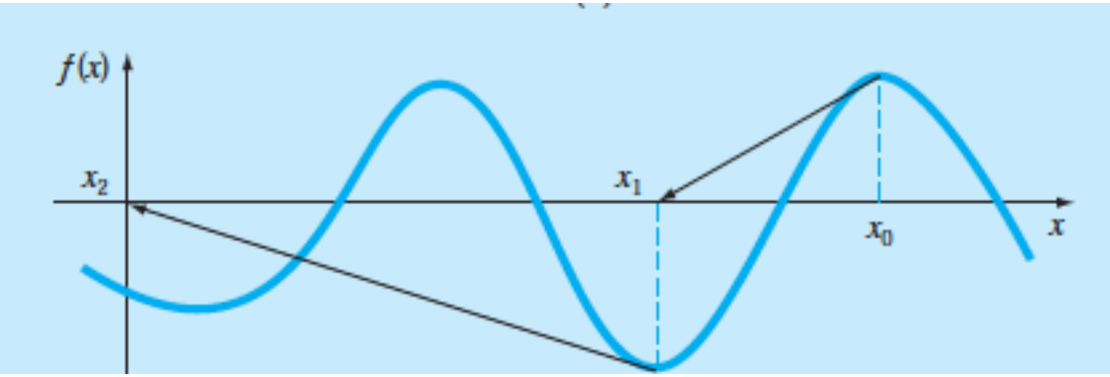
Newton Raphson Method: Poor Convergence



Loop and Divergence



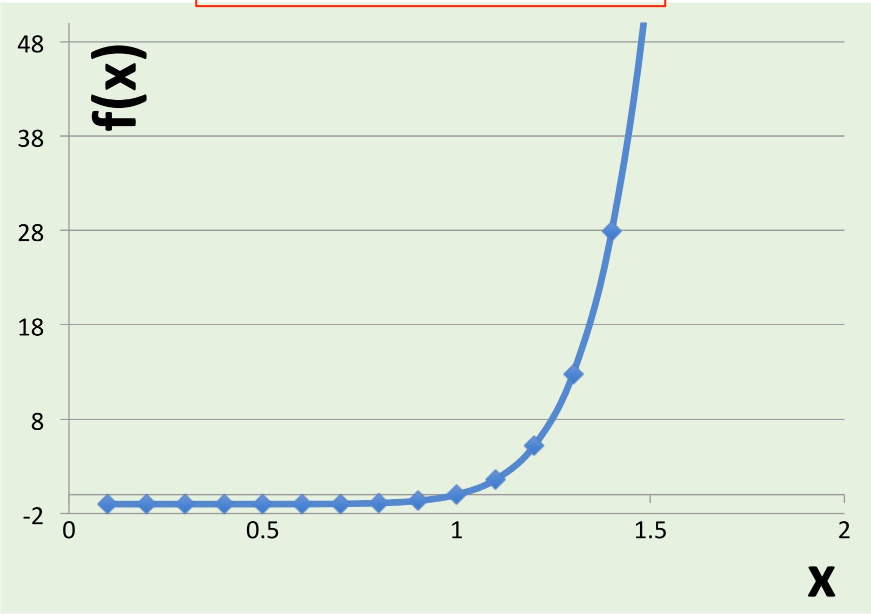
Oscillation around Minima or Maxima

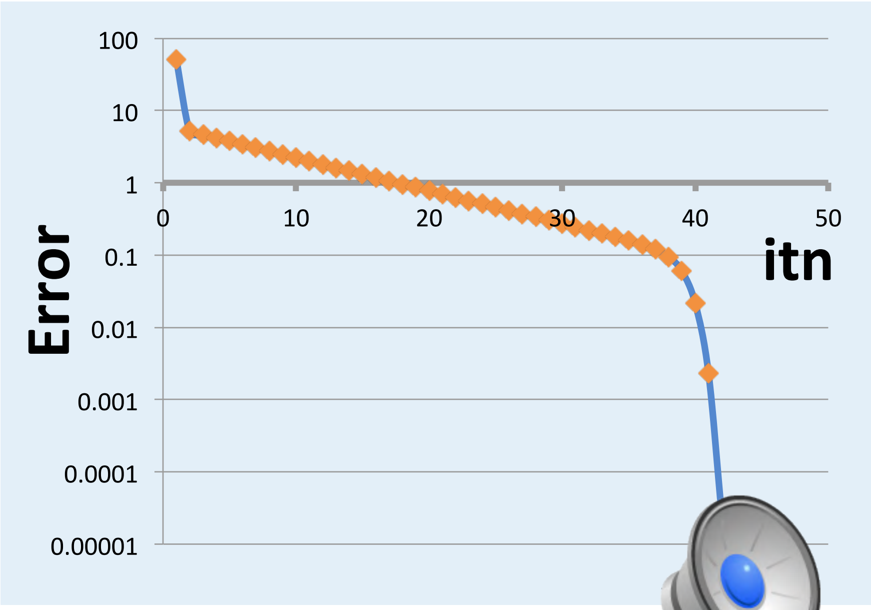


An initial guess near one root lands near a location several roots away

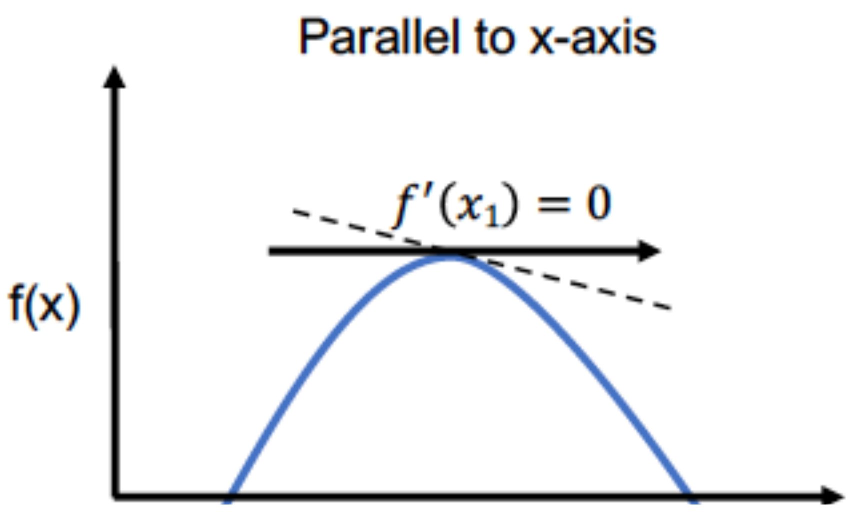
$$f(x) = x^{10} - 1 = 0$$

itn no	x(i)	f(i)	f`(i)	x(i+1)=	Error
1	0.5	-0.999023438	0.01953125	51.65	51.15
2	51.65	1.35115E+17	2.61597E+16	46.485	5.165
3	46.485	4.71117E+16	1.01348E+16	41.8365	4.6485
4	41.8365	1.64268E+16	3.92643E+15	37.65285	4.18365
5	37.65285	5.72768E+15	1.52118E+15	33.887565	3.765285
6	33.887565	1.99712E+15	5.89336E+14	30.4988085	3.3887565
7	30.4988085	6.96352E+14	2.28321E+14	27.44892765	3.04988085
8	27.44892765	2.42803E+14	8.84562E+13	24.70403489	2.744892765
9	24.70403489	8.46601E+13	3.42698E+13	22.2336314	2.470403488
10	22.2336314	2.95192E+13	1.32768E+13	20.01026826	2.22336314
11	20.01026826	1.02927E+13	5.14371E+12	18.00924143	2.001026826
12	18.00924143	3.58884E+12	1.99278E+12	16.20831729	1.800924143
13	16.20831729	1.25135E+12	7.72043E+11	14.58748556	1.620831729
14	14.58748556	4.36319E+11	2.99105E+11	13.128737	1.458748556
15	13.128737	1.52135E+11	1.15879E+11	11.8158633	1.3128737
16	11.8158633	53046236849	44894084748	10.63427697	1.18158633
17	10.63427697	18496079117	17392888267	9.570849276	1.063427697
18	9.570849276	6449184014	6738361278	8.613764348	0.957084927
19	8.613764348	2248691422	2610579222	7.752387914	0.861376434
20	7.752387914	784070216.9	1011391879	6.977149123	0.77523879
21	6.977149123	273388379.9	391833936.9	6.279434214	0.69771491
22	6.279434214	95324633.58	151804496	5.651490799	0.627943415
23	5.651490799	33237644.28	58812172.68	5.086341736	0.565149063
24	5.086341736	11589249.7	22785041.39	4.577707606	0.50863413
25	4.577707606	4040921.242	8827392.637	4.119936959	0.457770647
26	4.119936959	1408981.851	3419913.618	3.707943555	0.411993403
27	3.707943555	491281.3301	1324945.547	3.337149955	0.370793601
28	3.337149955	171298.9439	513312.0965	3.003436907	0.333713047
29	3.003436907	59727.98466	198868.7844	2.703098245	0.300338662
30	2.703098245	20825.59662	77047.13161	2.4328014	0.270296845
31	2.4328014	7261.172653	29851.07068	2.189554759	0.24324664
32	2.189554759	2531.55048	11566.50899	1.97068574	0.218869019
33	1.97068574	882.4332477	4482.872281	1.773840237	0.196845503
34	1.773840237	307.4217666	1738.723478	1.597031348	0.176808889
35	1.597031348	106.9280696	675.8043276	1.438807931	0.158223417
36	1.438807931	37.02141119	264.2563358	1.298711343	0.140096589
37	1.298711343	12.64980151	105.1026588	1.178354716	0.120356627
38	1.178354716	4.161315905	43.80103747	1.083349754	0.095004962
39	1.083349754	1.226829103	20.55503401	1.023664661	0.059685092
40	1.023664661	0.263505419	12.34296217	1.002316024	0.021348637
41	1.002316024	0.023403117	10.21038368	1.000023934	0.00229209
42	1.000023934	0.000239369	10.00215429	1.000000003	2.39317E-05



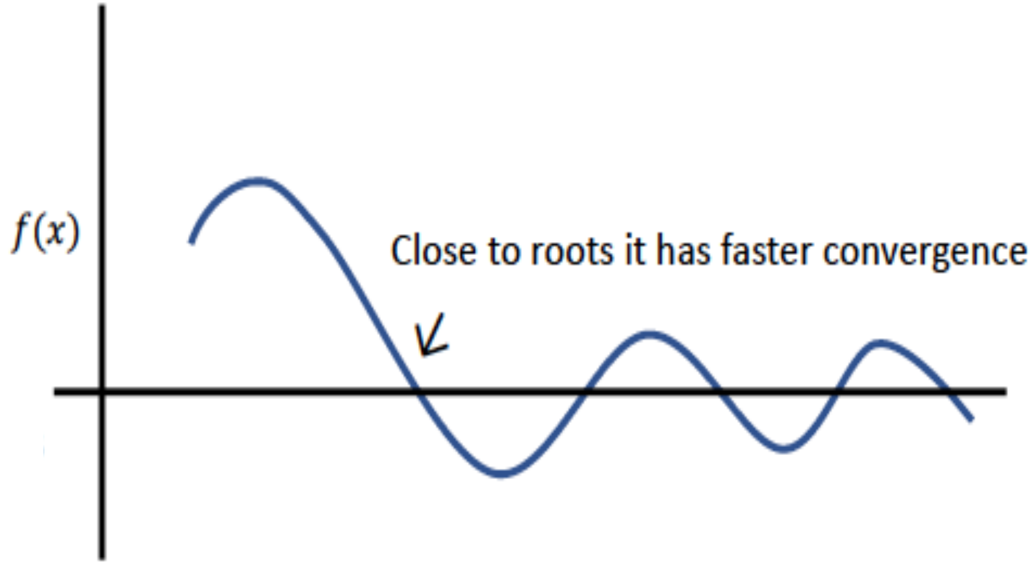


Improvements



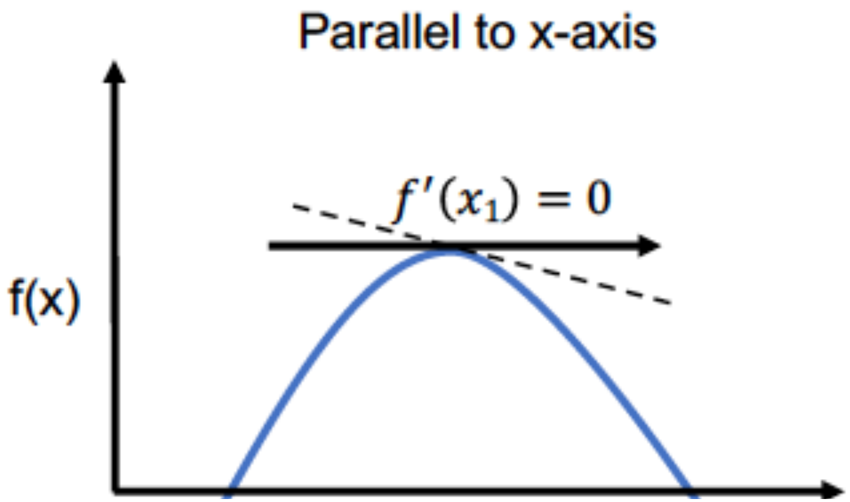
$$x^{(i+1)} = x^{(i)} - \frac{f\left(x^{(i)}\right)}{f'\left(x^{(i)}\right) + \beta}$$

add arbitrarily a small value to give a small slope



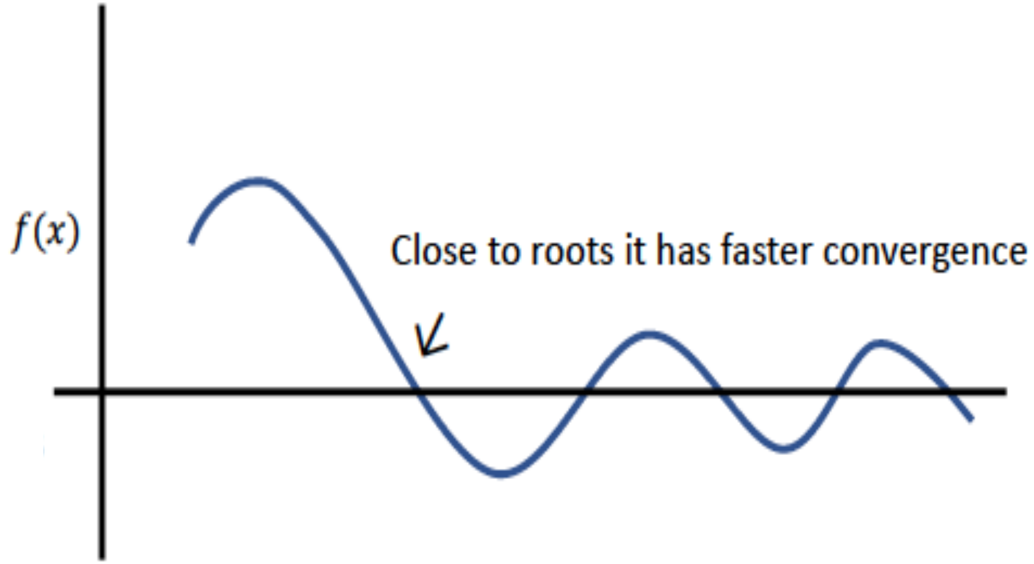
$$x^{(i+1)} = x^{(i)} - m\frac{f\left(x^{(i)}\right)}{f'\left(x^{(i)}\right)}$$

Improvements



$$x^{(i+1)} = x^{(i)} - \frac{f\left(x^{(i)}\right)}{f'\left(x^{(i)}\right) + \beta}$$

add arbitrarily a small value to give a small slope



$$x^{(i+1)} = x^{(i)} - m\frac{f\left(x^{(i)}\right)}{f'\left(x^{(i)}\right)}$$

Improvements

$$f(x)=0 \qquad\qquad \frac{f(x)}{f'(x)}=0$$

Both have the same solution $\boxed{\bar{x}}$

1. Modified Equation to solve:

$$u(x)=\frac{f(x)}{f'(x)}=0$$

Objective is to find the root of u(x)=0

$$\begin{aligned}
x^{(i+1)} &= x^{(i)}-\frac{u\left(x^{(i)}\right)}{u'\left(x^{(i)}\right)}\\
&= x^{(i)}-\frac{f\left(x^{(i)}\right)/f'\left(x^{(i)}\right)}{\left[f'\left(x^{(i)}\right)f'\left(x^{(i)}\right)-f\left(x^{(i)}\right)f''\left(x^{(i)}\right)\right]/\left[f'\left(x^{(i)}\right)\right]^2}\\
&= x^{(i)}-\frac{f\left(x^{(i)}\right)f'\left(x^{(i)}\right)}{\left[f'\left(x^{(i)}\right)f'\left(x^{(i)}\right)-f\left(x^{(i)}\right)f''\left(x^{(i)}\right)\right]}
\end{aligned}$$

2. Modified Equation to solve:

$$F(x)=f^2(x)=0$$

Multivariable case

$X_1, X_2, X_3 \ldots\ldots\ldots X_n$

$f_1(X_1, X_2, \ldots\ldots\ldots\ldots X_n) = 0$

$f_2(X_1, X_2, \ldots\ldots\ldots\ldots X_n) = 0$

.

.

.

$f_n(x_1, x_2, x_3 \ldots\ldots\ldots x_n) = 0$

Short hand case $\underline{f}(\underline{x}) = 0$

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_n \end{bmatrix}$$

Extension of fixed point iteration

$\underline{f}(\underline{x}) = 0$

$\underline{x} = \underline{x} + \underline{f}(\underline{x}) \Rightarrow \text{Let } \underline{x} + \underline{f}(\underline{x}) = g(\underline{x})$

$x_1 = g_1(\underline{x})$

$x_2 = g_2(\underline{x})$

.

.

$x_n = g_n(\underline{x})$

(1) start with initial guesses : $x_1{}^{(0)}, x_2{}^{(0)}, \ldots\ldots x_n{}^{(0)} \Rightarrow \underline{x}^{(0)}$

(2) $\underline{x}^{(i+1)} = \underline{g}(\underline{x}^{i})$

(3) Check for convergence $\left|E_j^{(i+1)}\right| = \left|x_j^{(i+1)} - x_j^{(i)}\right|$

$\max_j \{E_j{}^{i+1}\} \le E_{tol}$

$\left|E_1^{(100)}\right| = 0.01$

$\left|E_2^{(100)}\right| = 0.02$

$\left|E_3^{(100)}\right| = 0.002$

Criteria for convergence (FPI)

$$E^{(i+1)} = g'\,(\xi)E^{(i)} \qquad x^{(i)} < \xi \leq x^{(i+1)}$$

$$g_1\left(\underline{x}\right) = g_1\left(\underline{x^{(i)}}\right) + \frac{\partial g_1}{\partial x_1}\Big|_{\underline{x^{(i)}}}\left(x_1 - x_1^{(i)}\right) + \frac{\partial g_1}{\partial x_2}\Big|_{\underline{x^{(i)}}}\left(x_2 - x_2^{(i)}\right) + \ldots + \frac{\partial g_1}{\partial x_n}\Big|_{\underline{x^{(i)}}}\left(x_n - x_n^{(i)}\right)$$

$$x_1^t = x_1^{(i+1)} + \frac{\partial g_1}{\partial x_1}\Big|_{\underline{x^{(i)}}}\left(x_1^t - x_1^{(i)}\right) + \frac{\partial g_1}{\partial x_2}\Big|_{\underline{x^{(i)}}}\left(x_2^t - x_2^{(i)}\right) + \ldots + \frac{\partial g_1}{\partial x_n}\Big|_{\underline{x^{(i)}}}\left(x_n^t - x_n^{(i)}\right)$$

$$E_1^{(i+1)} = \frac{\partial g_1}{\partial x_1}\Big|_{\underline{x^{(i)}}} E_1^{(i)} + \frac{\partial g_1}{\partial x_2}\Big|_{\underline{x^{(i)}}} E_2^{(i)} + \ldots + \frac{\partial g_1}{\partial x_n}\Big|_{\underline{x^{(i)}}} E_n^{(i)}$$

Criteria for convergence (FPI)

$$E_j^{(i+1)} = \frac{\partial g_j}{\partial x_1}\Big|_{\underline{x^{(i)}}} E_1^{(i)} + \frac{\partial g_j}{\partial x_2}\Big|_{\underline{x^{(i)}}} E_2^{(i)} + \ldots + \frac{\partial g_j}{\partial x_n}\Big|_{\underline{x^{(i)}}} E_n^{(i)}$$

Sufficient condition for stability

$$\left[\left|\frac{\partial g_j}{\partial x_1}\right| + \left|\frac{\partial g_j}{\partial x_2}\right| + \ldots + \left|\frac{\partial g_j}{\partial x_n}\right|\right]\Big|_{\underline{x^{(i)}}} \leq 1$$

for j=1 to n

Very Stringent Condition

Modification

$f(x) = 0$

$\beta f(x) = 0$

$x + \beta f(x) = x$; $let \;\; x + \beta f(x) = g(x)$; β can be changed such that sufficient condition is met.

Newton Raphson Method: Multivariable

$$f_j\left(\underline{x}\right)=0$$

$$\underline{x}\Longrightarrow\underline{x}^{(i)}$$

$$\cancelto{0}{f_j\left(\underline{x}^{(i+1)}\right)}=f_j\left(\underline{x}^{(i)}\right)+\frac{\partial f_j}{\partial x_1}\Big|_{\underline{x}^{(i)}}\left(x_1^{(i+1)}-x_1^{(i)}\right)+\frac{\partial f_j}{\partial x_2}\Big|_{\underline{x}^{(i)}}\left(x_2^{(i+1)}-x_2^{(i)}\right)+\ldots..+\frac{\partial f_j}{\partial x_n}\Big|_{\underline{x}^{(i)}}\left(x_n^{(i+1)}-x_n^{(i)}\right)$$

$$\begin{bmatrix}\Delta x_1^{(i+1)}\\ \Delta x_2^{(i+1)}\\ \vdots\\ \Delta x_n^{(i+1)}\end{bmatrix}$$

In Matrix form:

$$\boxed{-f_j\left(\underline{x}^{(i)}\right)=\begin{bmatrix}\dfrac{\partial f_j}{\partial x_1} & \dfrac{\partial f_j}{\partial x_2}\ldots\ldots. & \dfrac{\partial f_j}{\partial x_n}\end{bmatrix}\Big|_{\underline{x}^{(i)}}\begin{bmatrix}\left(x_1^{(i+1)}-x_1^{(i)}\right)\\ \left(x_2^{(i+1)}-x_2^{(i)}\right)\\ \vdots\\ \left(x_n^{(i+1)}-x_n^{(i)}\right)\end{bmatrix}}$$

for j=1 to n

- If n = 2 $\Rightarrow$ j = 1,2

$$-f_1\left(x_1^{(i)},\ x_2^{(i)}\right) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \end{bmatrix} \begin{bmatrix} x_1^{(i+1)} - x_1^{(i)} \\ x_2^{(i+1)} - x_2^{(i)} \end{bmatrix}$$

$$-f_2\left(x_1^{(i)},\ x_2^{(i)}\right) = \begin{bmatrix} \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix} \begin{bmatrix} x_1^{(i+1)} - x_1^{(i)} \\ x_2^{(i+1)} - x_2^{(i)} \end{bmatrix}$$

$$-\begin{bmatrix} f_1^{(i)} \\ f_2^{(i)} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \begin{bmatrix} \Delta x_1^{(i+1)} \\ \Delta x_2^{(i+1)} \end{bmatrix}$$

Jacobian Matrix (J)

$$-\underline{f}(\underline{x}^i) = J(\underline{x}^i)\,\Delta\underline{x}^{(i+1)}$$

$$\Delta\underline{x}^{(i+1)} = -J^{-1}\underline{f}(\underline{x}^i)$$

$$\underline{x}^{(i+1)} = \underline{x}^{(i)} - J^{-1}\underline{f}\left(\underline{x}^{(i)}\right)$$

if Singular

Improvements

$$\underline{x}^{(i+1)} = \underline{x}^{(i)} - \left[\left(J^T J\right)^{-1} J^T\right]\underline{f}\left(\underline{x}^{(i)}\right)$$

$$\underline{x}^{(i+1)} = \underline{x}^{(i)} - \left[\left(J^T J + \beta I\right)^{-1} J^T\right]\underline{f}\left(\underline{x}^{(i)}\right)$$

leverberg Marqu
Modification

NUMERICAL METHODS IN CHEMICAL ENGINEERING

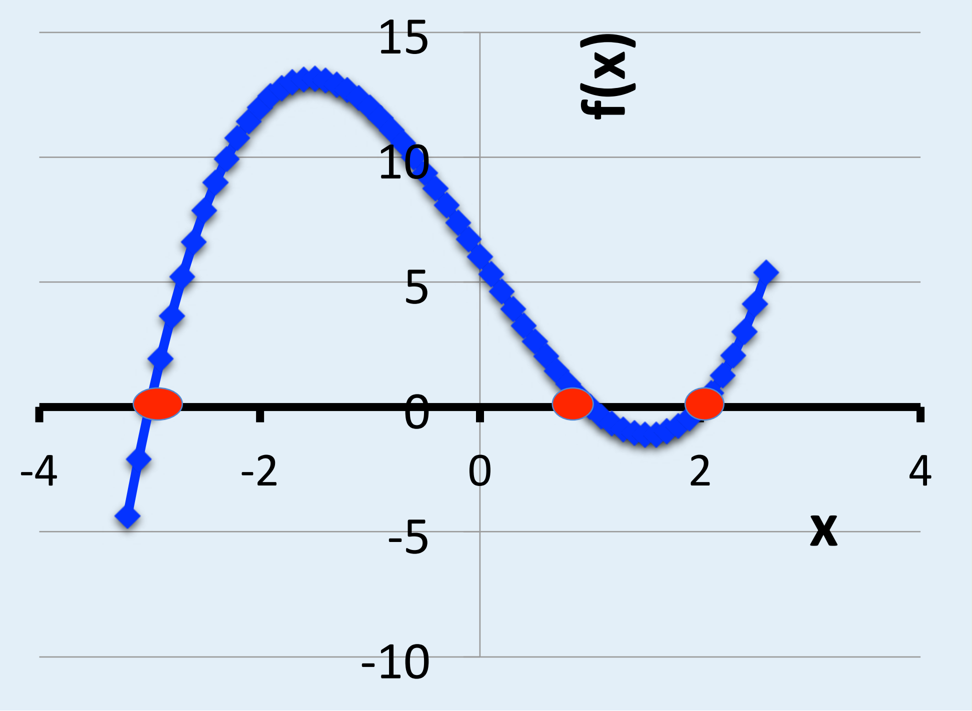
CLL-113

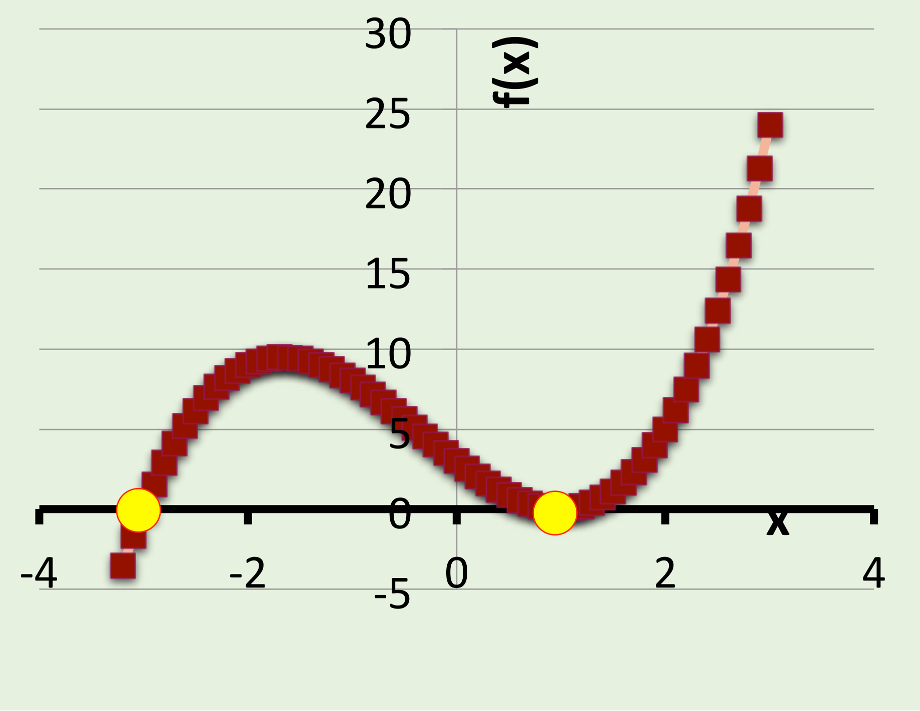
Solution of non-Linear Equations

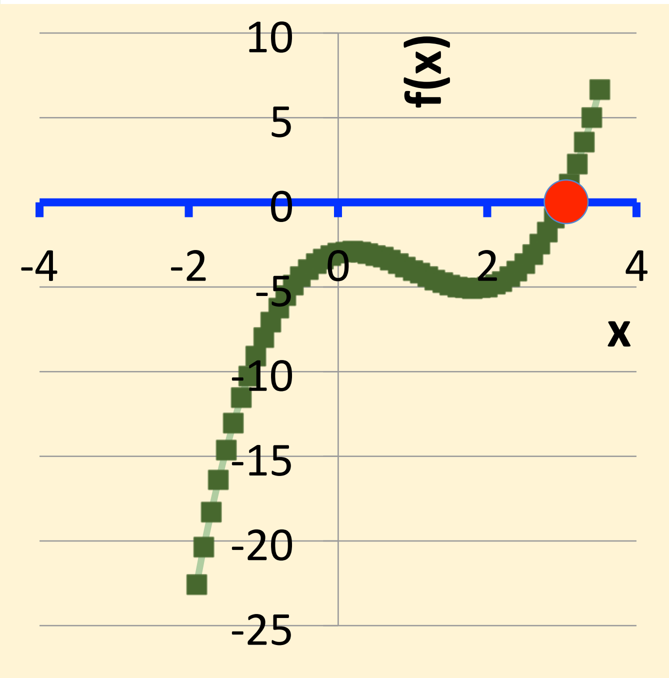
Root Finding of Polynomials

Prof. Jayati Sarkar

Roots of Polynomial







$$f(x) = x^3 - 7x + 6$$
$$= (x-1)(x-2)(x+3) = 0$$

$$f(x) = x^3 + x^2 - 5x + 3$$
$$= (x-1)^2(x+3) = 0$$

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

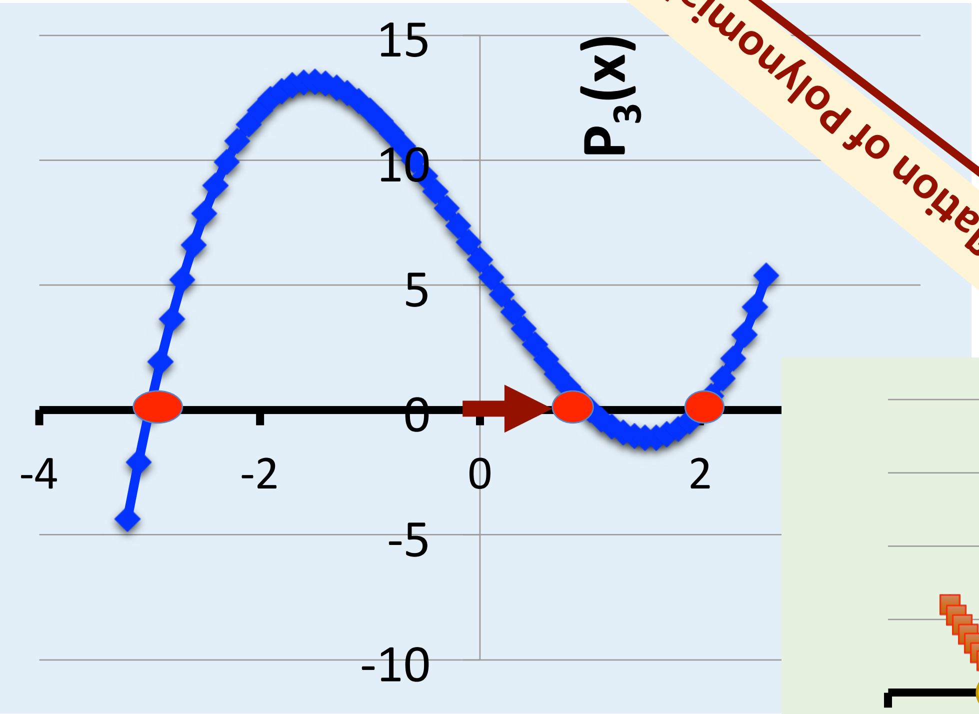
$\bar{x} = 1, 2, -3$ Real Unique Roots

$\bar{x} = 1, 1, -3$ Repeated Roots

$\bar{x} = -i, i, 3$ Complex Roots

Roots of Polynomial: Newton Raphson Method

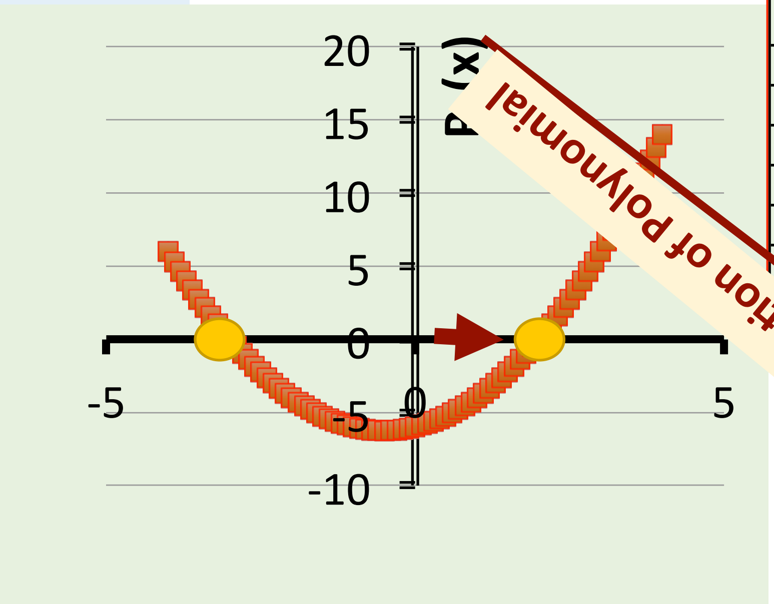
$$P_3(x) = x^3 - 7x + 6$$
$$= (x-1)(x-2)(x+3) = (x-1)P_2(x)$$



Deflation of Polynomial

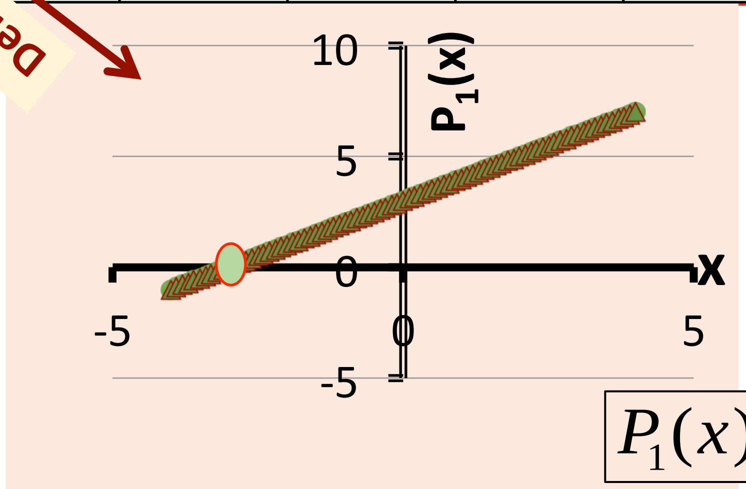
itn	x(i)	f(x)	f`(x)	x(i+1)	Error
1	0	6	-7	0.857142857	0.857142857
2	0.857142857	0.629737609	-4.795918367	0.988449848	0.131306991
3	0.988449848	0.046599285	-4.068900694	0.999902397	0.011452549
4	0.999902397	0.00039044	-4.000585589	0.999999993	9.75957E-05

$$P_2(x) = x^2 + x - 6 = (x-2)P_1(x)$$



Deflation of Polynomial

itn	x(i)	f(x)	f`(x)	x(i+1)	Error
1	1	-4	3	2.33333333	1.33333333
2	2.33333333	1.77777778	7	2.07936508	0.25396825
3	2.07936508	0.40312421	6.23809524	2.01474211	0.06462297
4	2.01474211	0.0739279	6.04422634	2.00251095	0.01223116
5	2.00251095	0.01256107	6.00753286	2.00042007	0.00209089
6	2.00042007	0.00210051	6.0012602	2.00007006	0.00035001
7	2.00007006	0.00035028	6.00021017	2.00001168	5.8378E-05



$$P_1(x) = x + 3$$

$$\bar{x} = 1, 2, -3$$ Real Unique Roots

Roots of Polynomial: Newton Raphson Method

1. Start with P_n
2. Choose initial guess x_1
3. Use N-R to get the first root $\bar{x}_1$
4. Deflate the polynomial to get P_{n-1}
   - $P_n \rightarrow P_{n-1} \rightarrow\ P_{n-2}.\ \ \ldots\ldots\ldots \rightarrow P_1$

$$x^{i+1} = x^i - m\frac{f\left(x^i\right)}{f'\left(x^i\right)}$$

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

- 1. Complex root a+ib
- 2. another complex root a-ib

$$\left(x-(a+ib)\right)\left(x-(a-ib)\right) = x^2 - rx - s$$

Deflate the polynomial by 2

$$P_n(x) \Rightarrow P_{n-2}(x) \Rightarrow P_{n-4}(x) \Rightarrow \ldots\ldots\ldots \Rightarrow P_2(x)$$ if n is Even

$$P_n(x) \Rightarrow P_{n-2}(x) \Rightarrow P_{n-4}(x) \Rightarrow \ldots\ldots\ldots \Rightarrow P_1(x)$$ if n is Odd

Bairstow's Method

$$P_n(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \dots\dots\dots + a_n x^n$$

$$Q(x) = x^2 - rx - s$$

Quadratic

Is this a factor?

$$P_n(x) = P_{n-2}(x)Q(x) + R(x)$$

$$P_{n-2}(x) = b_2 + b_3 x + b_4 x^2 \dots\dots\dots + b_n x^{n-2}$$

$$R(x) = b_0 + b_1(x - r)$$

Bairstow's Method

$$P_n(x) = \underline{P_{n-2}(x)}\,\underline{Q(x)} + \underline{R(x)}$$

$$a_0 + a_1x + a_2x^2 + a_3x^3 + \dots\dots\dots + a_nx^n$$
$$= \underline{\left(b_2 + b_3x + b_4x^2 \dots\dots\dots + b_nx^{n-2}\right)} \times \underline{\left(x^2 - rx - s\right)} + \underline{\left[b_0 + b_1\left(x - r\right)\right]}$$

$$\boxed{\begin{array}{lllllllll}
a_0 & + a_1x & + a_2x^2 & + a_3x^3 & + \dots\dots\dots\dots & + a_{n-2}x^{n-2} & + a_{n-1}x^{n-1} & + a_nx^n \\
= & & b_2x^2 & + b_3x^3 & + b_4x^4 \dots\dots & + b_{n-2}x^{n-2} & + b_{n-1}x^{n-1} & + b_nx^n \\
 & -b_2rx & -b_3rx^2 & -b_4rx^3 & -b_5rx^4 \dots\dots & -b_{n-1}rx^{n-2} & -b_nrx^{n-1} & +0 \\
-b_2s & -b_3sx & -b_4sx^2 & -b_5sx^3 & -b_6sx^4 \dots\dots & -b_nsx^{n-2} & +0 & +0 \\
\multicolumn{8}{l}{+\left(b_0 - b_1r\right) + b_1x}
\end{array}}$$

Bairstow's Method

$$
\begin{aligned}
x^n &: a_n = b_n \\
x^{n-1} &: a_{n-1} = b_{n-1} - b_n r \\
x^{n-2} &: a_{n-2} = b_{n-2} - b_{n-1} r - b_n s \\
&\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\
x^2 &: a_2 = b_2 - b_3 r - b_4 s \\
x^1 &: a_1 = b_1 - b_2 r - b_3 s \\
x^0 &: a_0 = b_0 - b_1 r - b_2 s
\end{aligned}
$$

$$
\begin{aligned}
x^n &: b_n = a_n \\
x^{n-1} &: b_{n-1} = a_{n-1} + b_n r \\
x^{n-2} &: b_{n-2} = a_{n-2} + b_{n-1} r + b_n s \\
&\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\
x^2 &: b_2 = a_2 + b_3 r + b_4 s \\
x^1 &: b_1 = a_1 + b_2 r + b_3 s \\
x^0 &: b_0 = a_0 + b_1 r + b_2 s
\end{aligned}
$$

$$
\begin{aligned}
x^n &: b_n = a_n \\
x^{n-1} &: b_{n-1} = a_{n-1} + b_n r \\
x^i &: b_i = a_i + b_{i+1} r + b_{i+2} s \\
&\qquad ......i = n-2..to..0
\end{aligned}
$$

Bairstow's Method

$$\begin{aligned} x^n &: b_n = a_n \\ x^{n-1} &: b_{n-1} = a_{n-1} + b_n r \\ x^i &: b_i = a_i + b_{i+1} r + b_{i+2} s \\ &\quad ......i = n\text{-}2..to..0 \end{aligned}$$

$$R(x) = b_0 + b_1(x - r)$$

$$b_0(r,s) \Rightarrow 0$$

$$b_1(r,s) \Rightarrow 0$$

0

$$b_1(r+\Delta r, s+\Delta s) = b_1(r,s) + \frac{\partial b_1}{\partial r}\Delta r + \frac{\partial b_1}{\partial s}\Delta s$$

0

$$b_0(r+\Delta r, s+\Delta s) = b_0(r,s) + \frac{\partial b_0}{\partial r}\Delta r + \frac{\partial b_0}{\partial s}\Delta s$$

$$\begin{bmatrix} \dfrac{\partial b_1}{\partial r} & \dfrac{\partial b_1}{\partial s} \\ \dfrac{\partial b_0}{\partial r} & \dfrac{\partial b_0}{\partial s} \end{bmatrix} \begin{bmatrix} \Delta r \\ \Delta s \end{bmatrix} = \begin{bmatrix} -b_1(r,s) \\ -b_0(r,s) \end{bmatrix}$$

$$[M] \begin{bmatrix} \Delta r \\ \Delta s \end{bmatrix} = \begin{bmatrix} -b_1(r,s) \\ -b_0(r,s) \end{bmatrix}$$

$$\begin{bmatrix} \Delta r \\ \Delta s \end{bmatrix} = \boxed{[M]^{-1}} \begin{bmatrix} -b_1(r,s) \\ -b_0(r,s) \end{bmatrix}$$

Bairstow's Method

$$x^n \;:\; c_n = b_n$$

$$x^{n-1} :\; c_{n-1} = b_{n-1} + c_n r$$

$$x^i :\; c_i = b_i + c_{i+1} r + c_{i+2} s$$

$$\ldots\ldots i = n - 2..to..1$$

$$\begin{bmatrix} \dfrac{\partial b_1}{\partial r} & \dfrac{\partial b_1}{\partial s} \\ \dfrac{\partial b_0}{\partial r} & \dfrac{\partial b_0}{\partial s} \end{bmatrix} \begin{bmatrix} \Delta r \\ \Delta s \end{bmatrix} = \begin{bmatrix} -b_1(r,s) \\ -b_0(r,s) \end{bmatrix}$$

$$\begin{bmatrix} c_2 & c_3 \\ c_1 & c_2 \end{bmatrix} \begin{bmatrix} \Delta r \\ \Delta s \end{bmatrix} = \begin{bmatrix} -b_1(r,s) \\ -b_0(r,s) \end{bmatrix}$$

$$\Delta s = \frac{-b_0 + \dfrac{c_1}{c_2} b_1}{c_2 - \dfrac{c_1}{c_2} c_3}, \Delta r = \frac{-b_1 - c_3 \Delta s}{c_2}$$

$$r^{(i+1)} = r^{(i)} + \Delta r$$

$$s^{(i+1)} = s^{(i)} + \Delta s$$

$$\left|\varepsilon_{a,r}\right| = \left|\frac{\Delta r}{r}\right| 100\%$$

$$\left|\varepsilon_{a,s}\right| = \left|\frac{\Delta s}{s}\right| 100\%$$

Roots

$$\boxed{x = \frac{r \pm \sqrt{r^2 + 4s}}{2}}$$

Bairstow's Method

1. P_{n-2} is a third order polynomial or greater. Bairstow's method is repeated for the quotient for finding new values of r and s.

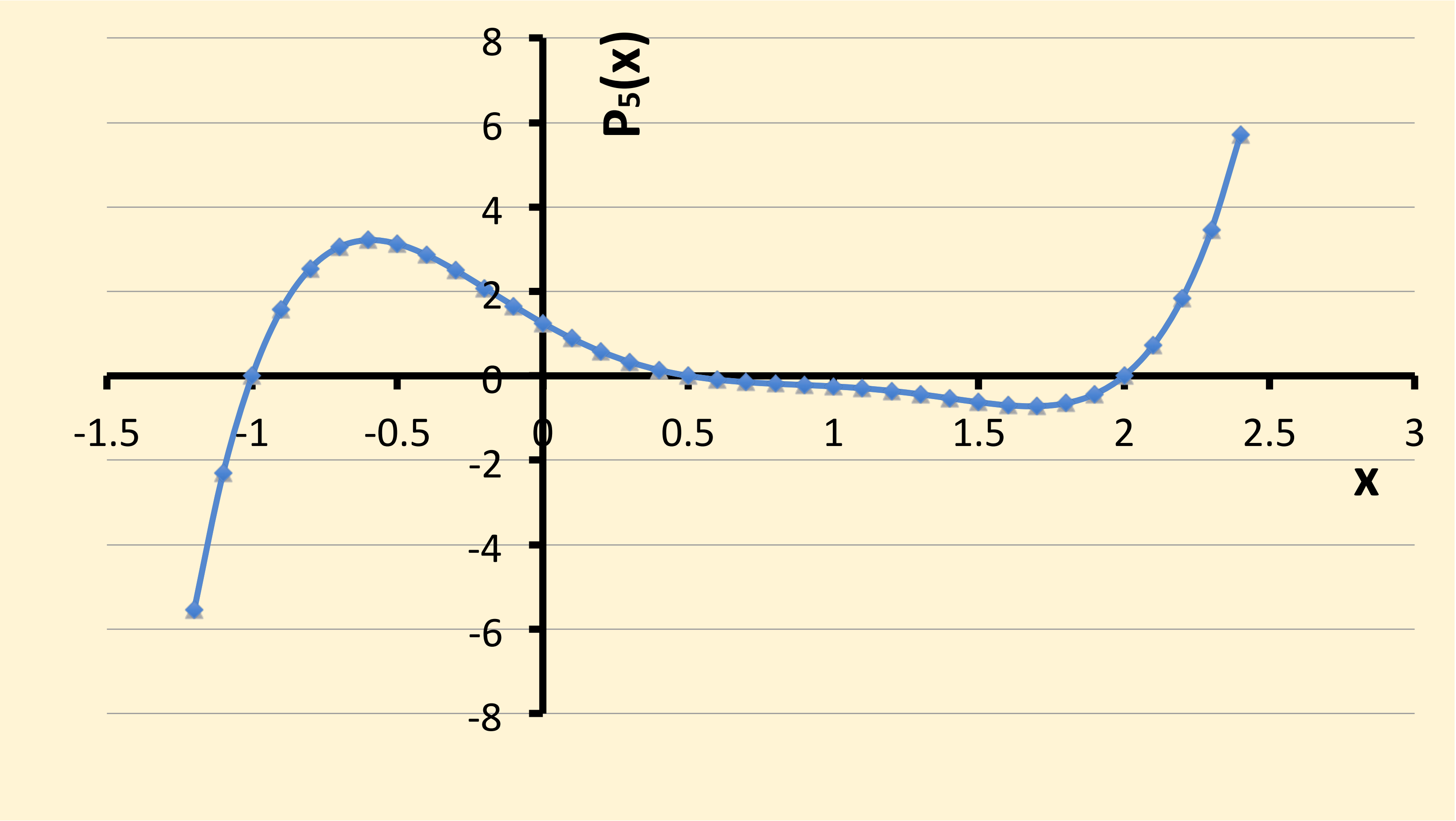
2. The quotient is quadratic. Roots can be evaluated directly.

$$x = \frac{r \pm \sqrt{r^2 + 4s}}{2}$$

3. The quotient is first order polynomial, the remaining single root can be evaluated as $x = -r/s$

$$P_5(x) = x^5 - 3.5x^4 + 2.75x^3 + 2.125x^2 - 3.875x + 1.25$$

P5(x)	a5	a4	a3	a2	a1	a0
	1	-3.5	2.75	2.125	-3.875	1.25



Real Roots:

x_1=-1,

x_2=0.5

x_3=2

$$P_5(x) = x^5 - 3.5x^4 + 2.75x^3 + 2.125x^2 - 3.875x + 1.25$$

P5(x)

a5	a4	a3	a2	a1	a0
1	-3.5	2.75	2.125	-3.875	1.25

itn	r	s	b5	b4	b3	b2	b1	b0
1	-1.00	-1.00	1.00	-4.50	6.25	0.38	-10.50	11.38
2	-0.64	0.14	1.00	-4.14	5.56	-2.03	-1.80	2.13
3	-0.51	0.47	1.00	-4.01	5.27	-2.45	-0.15	0.17
4	-0.50	0.50	1.00	-4.00	5.25	-2.50	0.00	0.00
5	-0.50	0.50	1.00	-4.00	5.25	-2.50	0.00	0.00
6	-0.50	0.50	1.00	-4.00	5.25	-2.50	0.00	0.00

itn	c5	c4	c3	c2	c1
1	1.00	-5.50	10.75	-4.88	-16.38
2	1.00	-4.79	8.78	-8.34	4.79
3	1.00	-4.52	8.05	-8.69	8.08
4	1.00	-4.50	8.00	-8.75	8.37
5	1.00	-4.50	8.00	-8.75	8.38
6	1.00	-4.50	8.00	-8.75	8.38

dels	delr	Errs=	Err_r
1.138109017	0.35583014	824.0656852	55.23855773
0.331624608	0.133056764	70.5984393	26.03274392
0.030468695	0.011426623	6.091274153	2.286758475
-0.00020233	-0.000313592	0.040466059	0.062718311
1.03876E-08	6.52645E-08	2.07753E-06	1.30529E-05
-1.67709E-14	-1.08671E-14	#DIV/0!	#DIV/0!

$$x = \left(-0.5 \pm \sqrt{-0.5^2 + 4 \times 0.5}\right) \Big/ 2 \qquad x_1 = 0.5, x_2 = -1$$

$$b_5 = a_5$$
$$b_4 = a_4 + b_5 r$$
$$b_3 = a_3 + b_4 r + b_5 s$$
$$b_2 = a_2 + b_3 r + b_4 s$$
$$b_1 = a_1 + b_2 r + b_3 s$$
$$b_0 = a_0 + b_1 r + b_2 s$$

$$c_5 = b_5$$
$$c_4 = b_4 + c_5 r$$
$$c_3 = b_3 + c_4 r + c_5 s$$
$$c_2 = b_2 + c_3 r + c_4 s$$
$$c_1 = b_1 + c_2 r + c_3 s$$

$$\Delta s = \frac{-b_0 + \dfrac{c_1}{c_2} b_1}{c_2 - \dfrac{c_1}{c_2} c_3}, \quad \Delta r = \frac{-b_1 - c_3 \Delta s}{c_2}$$

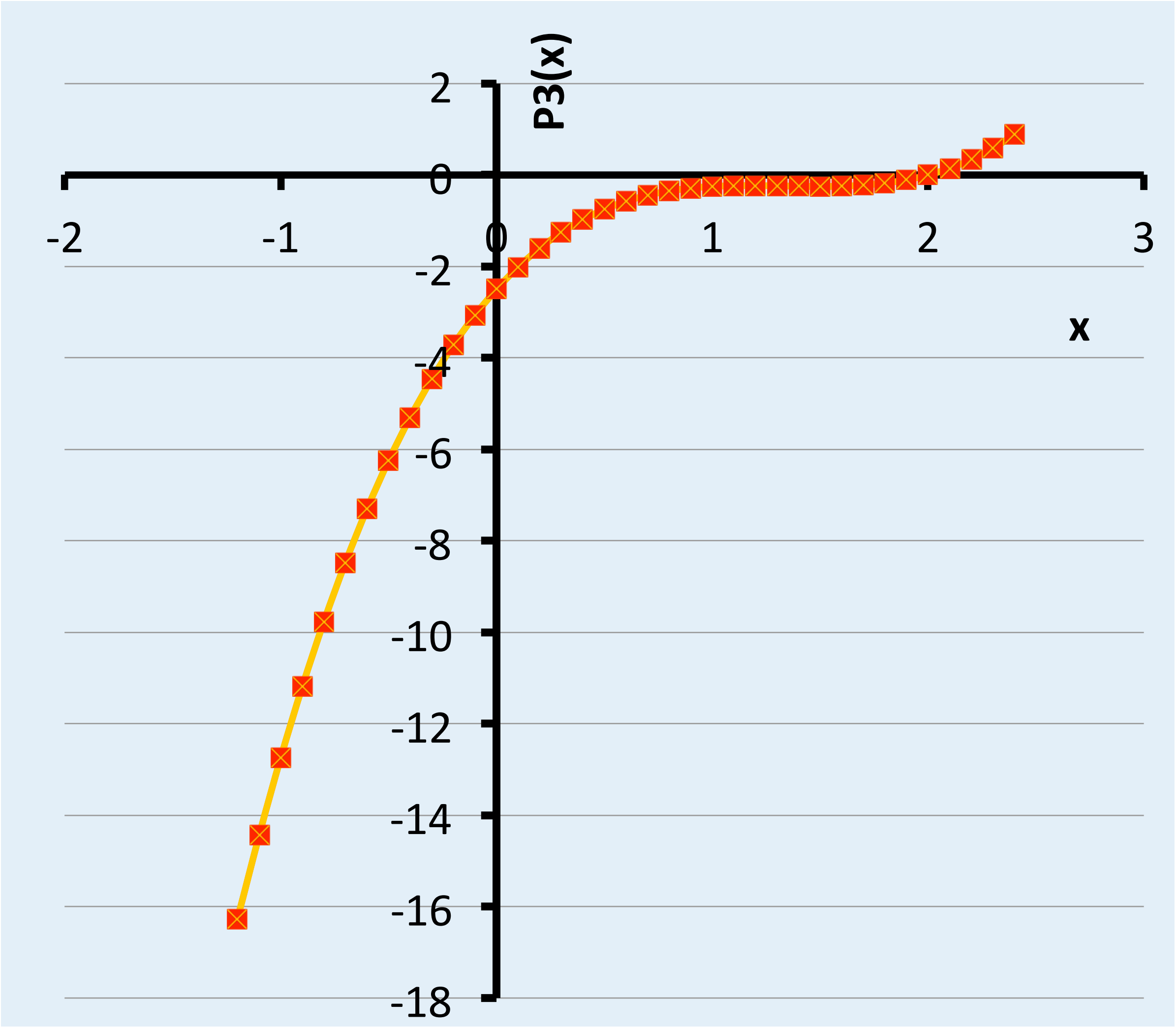
$$x = \left(r \pm \sqrt{r^2 + 4s}\right) \Big/ 2$$

$$P_3(x) = b_2 + b_3 x + b_4 x^2 + b_5 x^3$$

b5	b4	b3	b2
1.00	-4.00	5.25	-2.50

P3(x)

a3	a2	a1	a0
1	-4	5.25	-2.5



Real Roots:

x_1=-1,

x_2=0.5

x_3=2

$$P_3(x) = b_2 + b_3 x + b_4 x^2 + b_5 x^3 \qquad P_3(x) = x^3 - 4x^2 + 5.25x - 2.5$$

P3(x)	a3	a2	a1	a0
	1	-4	5.25	-2.5

itn	r	s	b3	b2	b1	b0	c3	c2	c1	dels	delr	Errs=	Err_r
1	0.00	5.00	1.00	-4.00	10.25	-22.50	1.00	-4.00	15.25	88.42	24.67	3.584459459	0.264049955
2	24.67	93.42	1.00	20.67	608.44	16936.41	1.00	45.33	1820.08	1445.09	-45.30	-70.04163723	-0.029443207
3	-20.63	1538.50	1.00	-24.63	2051.95	-80234.30	1.00	-45.26	4524.33	-2283.16	-5.11	88.70166669	0.006859424
4	-25.74	-744.65	1.00	-29.74	26.09	21471.67	1.00	-55.48	709.46	510.76	9.68	-31.79737754	-0.041372997
5	-16.06	-233.89	1.00	-20.06	93.64	3185.91	1.00	-36.13	440.05	180.68	7.59	-21.33269308	-0.142716826
6	-8.47	-53.21	1.00	-12.47	57.66	172.60	1.00	-20.94	181.80	54.93	5.38	-17.75781197	3.126486167
7	-3.09	1.72	1.00	-7.09	28.91	-104.11	1.00	-10.19	62.13	17.68	4.57	11.94047118	0.235791492
8	1.48	19.40	1.00	-2.52	20.92	-20.41	1.00	-1.04	38.77	-20.95	-0.04	-14.49929367	0.022664987
9	1.45	-1.56	1.00	-2.55	0.00	1.48	1.00	-1.11	-3.16	0.37	0.34	0.209251221	-0.284808763
10	1.78	-1.18	1.00	-2.22	0.11	0.33	1.00	-0.44	-1.85	-0.03	0.19	-0.016737978	-0.152346768
11	1.97	-1.22	1.00	-2.03	0.03	0.04	1.00	-0.06	-1.31	-0.03	0.03	-0.01612642	-0.025990154
12	2.00	-1.25	1.00	-2.00	0.00	0.00	1.00	0.00	-1.25	0.00	0.00	-0.000526834	3.52232E-05
13	2.00	-1.25	1.00	-2.00	0.00	0.00	1.00	0.00	-1.25	0.00	0.00	-9.69272E-10	-6.18632E-08

$$b_3 = a_3$$
$$b_2 = a_2 + b_3 r$$
$$b_1 = a_1 + b_2 r + b_3 s$$
$$b_0 = a_0 + b_1 r + b_2 s$$

$$c_3 = b_3$$
$$c_2 = b_2 + c_3 r$$
$$c_1 = b_1 + c_2 r + c_3 s$$

$$\Delta s = \frac{-b_0 + \dfrac{c_1}{c_2} b_1}{c_2 - \dfrac{c_1}{c_2} c_3}, \quad \Delta r = \frac{-b_1 - c_3 \Delta s}{c_2}$$

$$x = \left(2 \pm \sqrt{2^2 - 4 \times 1.25}\right) / 2 \qquad x_3 = (1 + 0.5i), x_4 = (1 - 0.5i)$$

$$P_1(x) = b_2 + b_3 x \qquad \boxed{x_5 = -b_2 / b_3 = 2}$$

$$x = \left(r \pm \sqrt{r^2 + 4s}\right) /$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Regression Analysis and Curve Fitting

Least Square Method

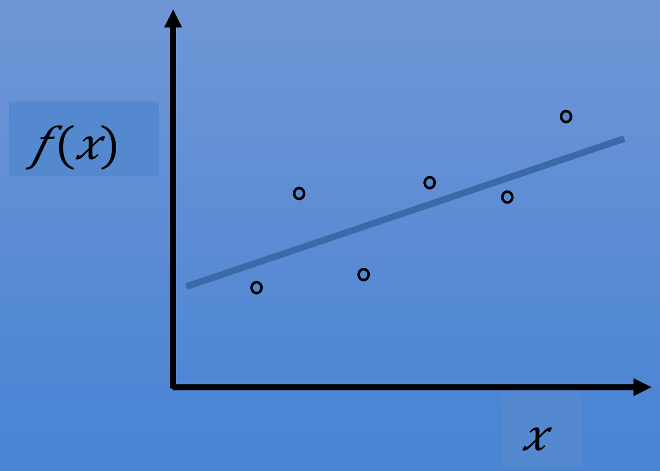
Prof. Jayati Sarkar

Curve Fitting

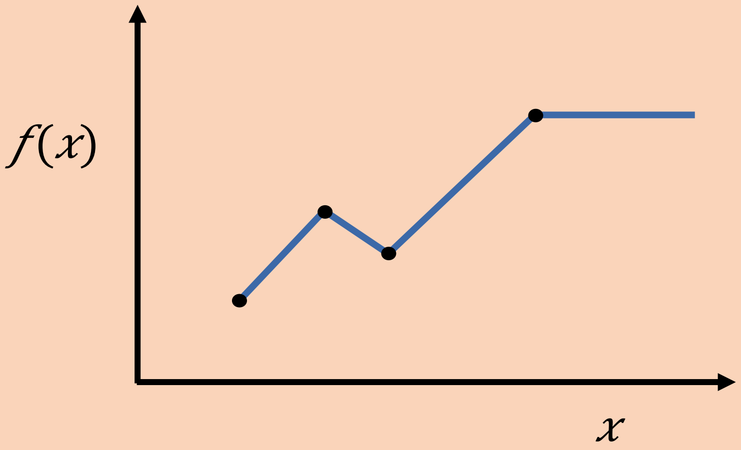
✓ Data is given at discrete values along a continuum

- Growth rate as a function of concentration:

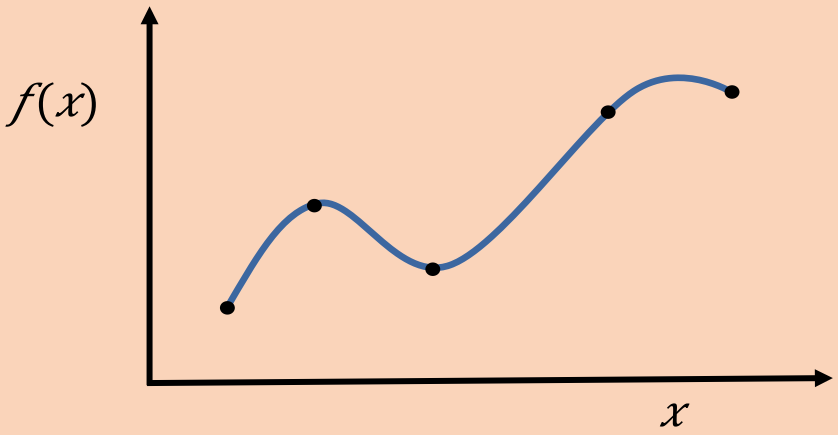
C	0.5	0.8	1.5	2.5	4
X	1.1	2.4	5.3	7.6	8.9



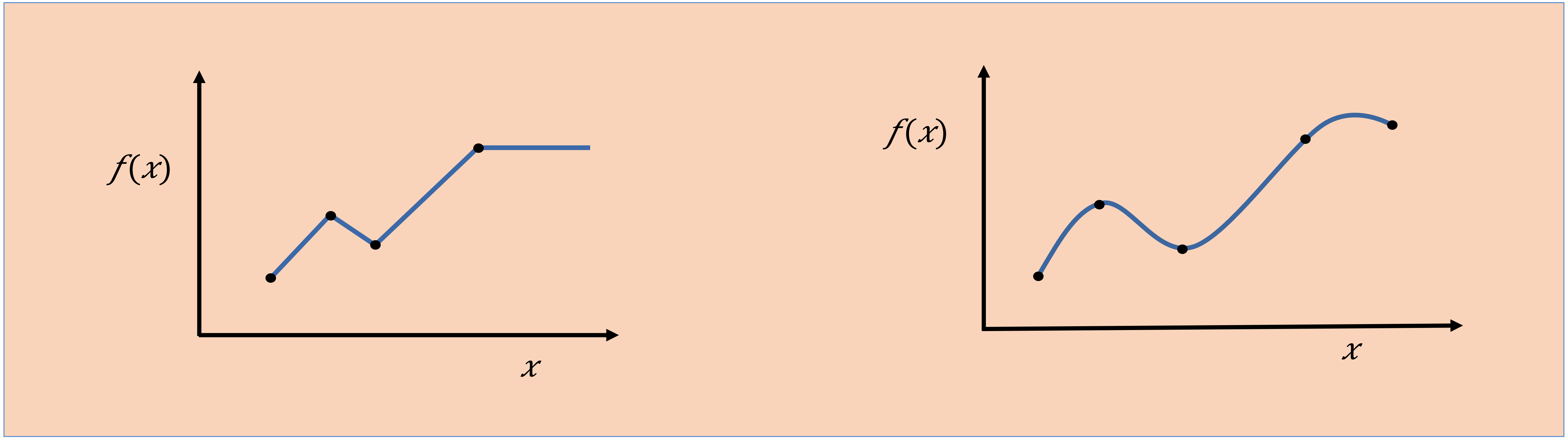
Did not attempt to connect the points but characterizes general upward trend . Curve fitting which does this is called ‘**Regression**’



Used straight line segments or linear interpolation to connect the points. Fails when the underlying curve is highly curve linear or data is widely placed



Captures meandering suggested by data



- When data is precise as obtained from steam tables.
- Density ρ , viscosity η as a function of temperature.
- Heat capacity C_p as a function of Temperature and Pressure

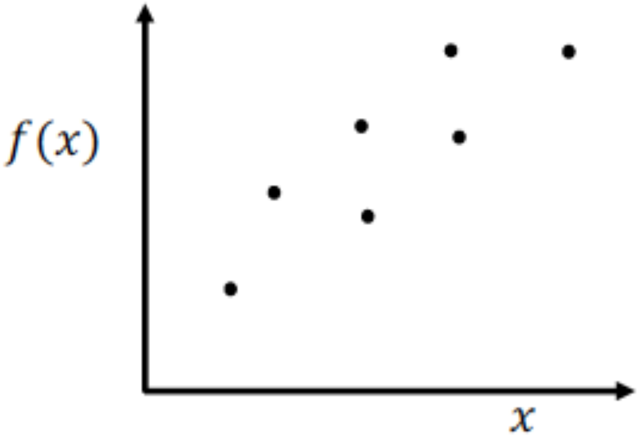
  ↓

  The approach is to fit a curve that passes directly through each point.

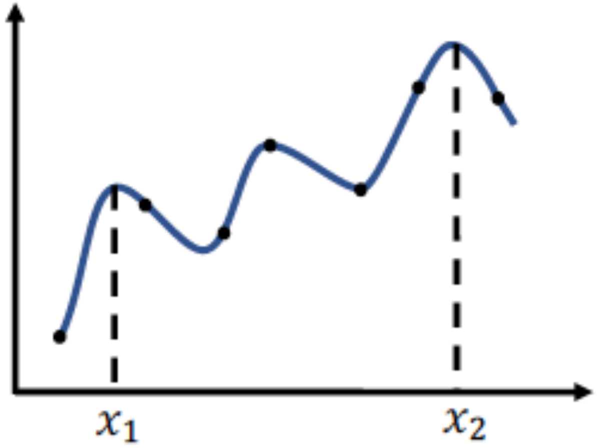
- Converts discrete data into analytical form such that any in between points within the range

  be interpolated.

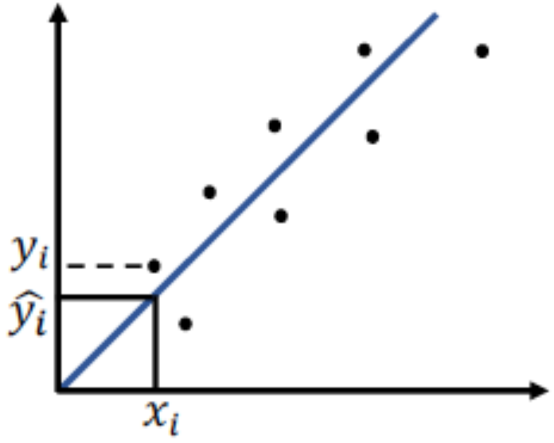
Linear Regression



Data with significant errors



Try to fit polynomial predicted values at x_1 & x_2 will be way out of range of the data



More exact result by using least square fit

Criteria for best fit:

$(x_i, y_i) \quad i = 1, N$

Let us fit a straight line

$$\hat{y}_i = a_o + a_1 x_i$$

$\downarrow$

Predicted value at a particular x_i

Linear Regression

Data: (x_1,y_1) (x_2,y_2) (x_3,y_3) (x_4,y_4) ……(x_N,y_N)

Model: $y=a_o+a_1 x$

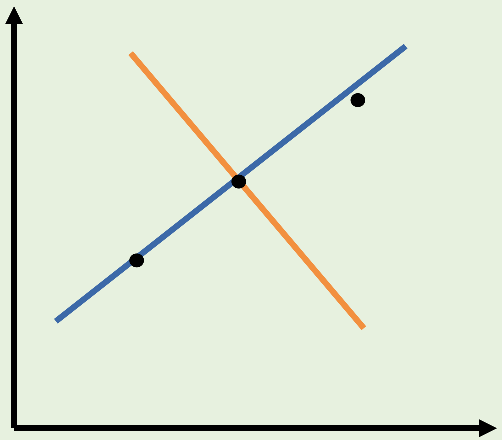
Predicted: $\hat{y}_i = a_o + a_1 x_i$

$$\sum_{i=1}^{N} e_i = \sum_{i=1}^{N} (y_i - \hat{y}_i) = \sum_{i=1}^{N} (y_i - a_0 - a_1 x_i)$$

where N = total number of points

- Minimize the er

Inadequate Criteria



Best fit line passes through two points

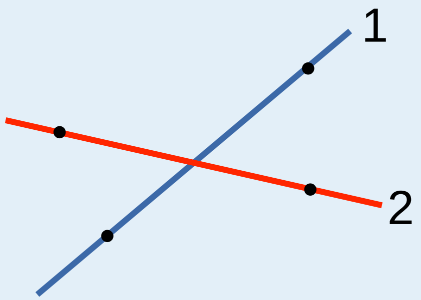
$$\sum_{i=1}^{N} e_i = \sum_{i=1}^{N} (y_i - \hat{y}_i) = \sum_{i=1}^{N} (y_i - a_0 - a_1 x_i)$$

where N = total number of points

Any straight line passing through the mid point also minimizes the error so it gets cancelled.

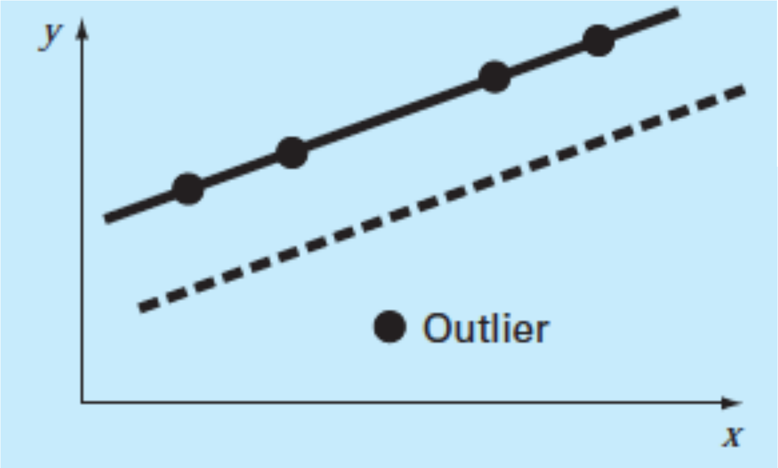
Another criteria can be to minimize the sum of absolute values of the discrepancies

$$\sum_{i=1}^{N} |e_i| = \sum_{i=1}^{N} |y_i - a_0 - a_1 x_i|$$



Any of the straight line lying between straight lines 1 & 2 passing through the 4 points will satisfy this, we will not end up with a UNIQUE solution.

Error Minimization Criteria



Third Criteria → minimize the maximum error

This gives undue advantage to a point which is an outlier with a large error.

The strategy that circumvents all the problem is to minimize the sum of the squares of the residuals.

$$S_r = \sum_{i=1}^{N} e_i^2 = \sum_{i=1}^{N} \left(y_{i,measured} - y_{i,modeled}\right)^2 = \sum_{i=1}^{N} (y_i - a_0 - a_1 x_i)^2$$

↓

This gives a unique line that satisfies the data

$$S_r = \sum_{i=1}^{N} (y_i - a_0 - a_1 x_i)^2$$

$$\downarrow$$

Strategy is to minimize the square of the residual

$$S_r = S_r(a_o, a_1) \quad \Rightarrow \quad S_r]_{min}, \quad \Rightarrow \quad \frac{\partial S_r}{\partial a_0} = 0, \frac{\partial S_r}{\partial a_1} = 0,$$

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

$$0 = \sum y_i - \sum a_o - \sum a_1 x_i$$

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

$$0 = \sum y_i x_i - \sum a_o x_i - \sum a_1 {x_i}^2$$

$$\boxed{0 = \sum y_i - \sum a_o - \sum a_1 x_i}$$

$$\boxed{0 = \sum y_i x_i - \sum a_o x_i - \sum a_1 {x_i}^2}$$

$$N a_o + (\sum x_i) a_1 = \sum y_i \qquad \rightarrow \quad (1)$$

$$(\sum x_i) a_0 + (\sum {x_i}^2) a_1 = \sum x_i y_i \quad \rightarrow \quad (2)$$

$$eqn(1) \times \sum x_i \; - eqn(2) \times N$$

$$\Rightarrow \; a_1 = \frac{N \sum x_i y_i - \sum x_i y_i}{N \sum x_i^2 - (x_i)^2}$$

From 1st equation

$$\boxed{\begin{aligned} a_o &= \frac{\sum y_i}{N} - \left(\frac{\sum x_i}{N}\right) a_1 \\ &= Y_{avg} - x_{avg} a_1 \end{aligned}}$$

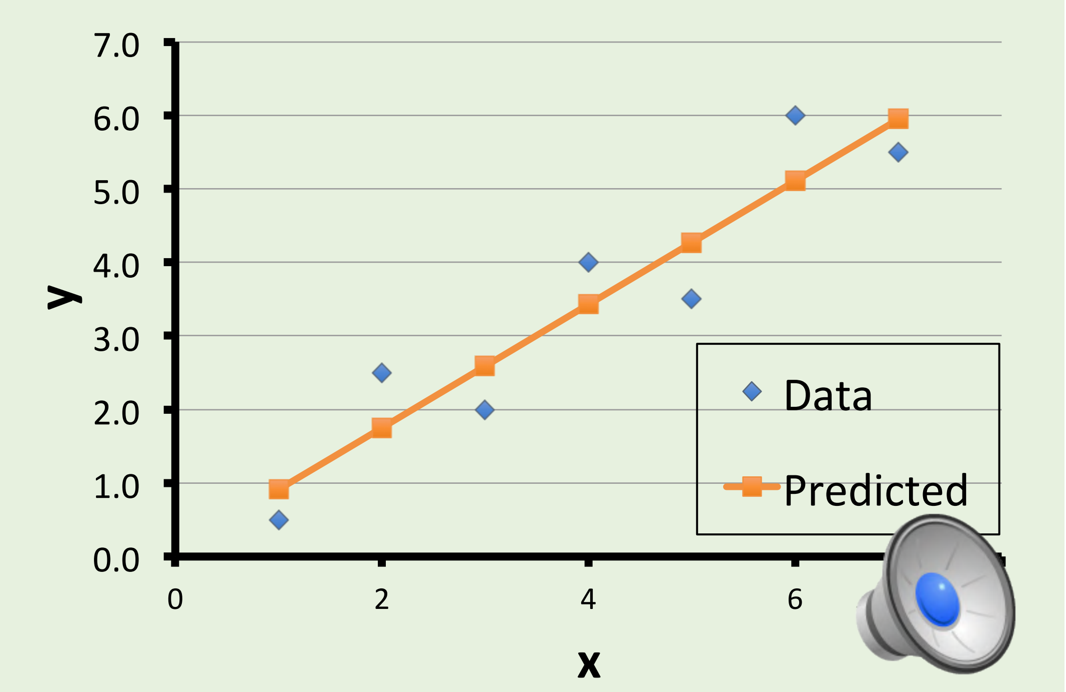
No of Data Point	x_i	y_i	x^2_i	$x_i y_i$	$y_{predicted}$
1	1	0.5	1.00	0.50	0.91
2	2	2.5	4.00	5.00	1.75
3	3	2.0	9.00	6.00	2.59
4	4	4.0	16.00	16.00	3.43
5	5	3.5	25.00	17.50	4.27
6	6	6.0	36.00	36.00	5.11
7	7	5.5	49.00	38.50	5.95
	28	24	140	119.5	

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

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

$$y = a_0 + a_1 x$$

a_1	a_0
0.8393	0.0714



Quantification of Error of Linear Regression

$$S_r = \sum_{i=1}^{N} e_i^2 = \sum_{i=1}^{N} (y_i - a_0 - a_1 x_i)^2$$

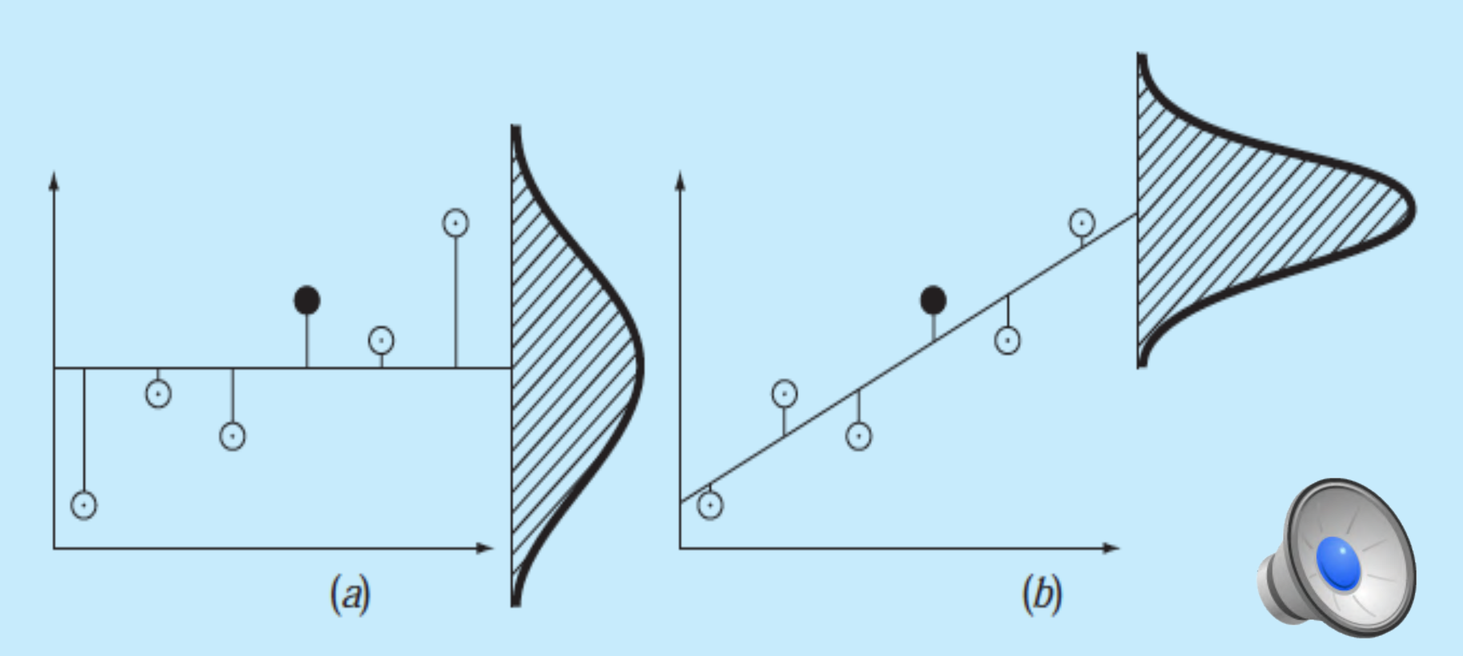
Standard Variation for the Regression Line

$$S_{y/x} = \sqrt{\frac{S_r}{N-2}}$$

Standard Error of the Estimate

Standard Derivation for the data

$$S_y = \sqrt{\frac{S_t}{N-1}} = \sqrt{\frac{\left(y_i - \bar{y}\right)^2}{N-1}}$$



$$r^2 = \frac{S_t - S_r}{S_t}$$

r: Correlation Coefficient

Quatification of Errors

No of Data Point	x_i	y_i	$y_{predicted}$	$(y_i - y_{bar})^2$	$(y_i - y_{predicted})^2$
1	1	0.5	0.91	8.58	0.17
2	2	2.5	1.75	0.86	0.56
3	3	2.0	2.59	2.04	0.35
4	4	4.0	3.43	0.33	0.33
5	5	3.5	4.27	0.01	0.59
6	6	6.0	5.11	6.61	0.80
7	7	5.5	5.95	4.29	0.20
	28	24	24	22.71	2.99

St	Sr	r2
22.71	2.99	0.87
Sy/x	**Sy**	**r**
0.77	1.95	0.93

$$S_r = \sum_{i=1}^{N} e_i^2$$

Standard Error of the Estimate

$$S_{y/x} = \sqrt{\frac{S_r}{N-2}}$$

Standard Derivation for the data

$$S_y = \sqrt{\frac{S_t}{N-1}} = \sqrt{\frac{\left(y_i - \bar{y}\right)^2}{N-1}}$$

$$r^2 = \frac{S_t - S_r}{S_t}$$

r^2: Coefficient of Determination

Least Square Method

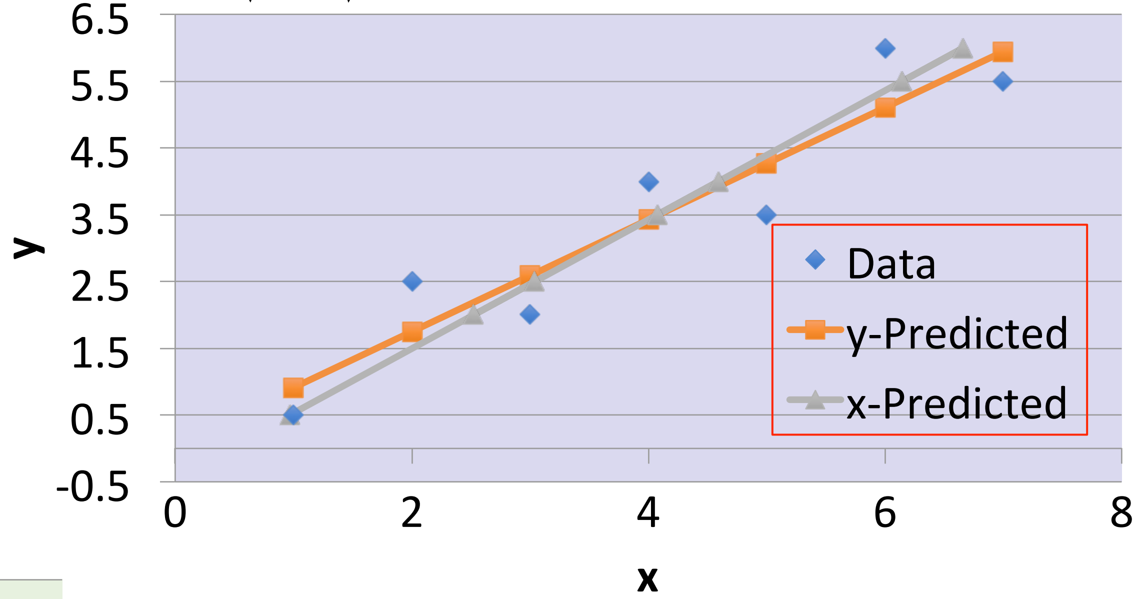
$$x = b_0 + b_1 y$$

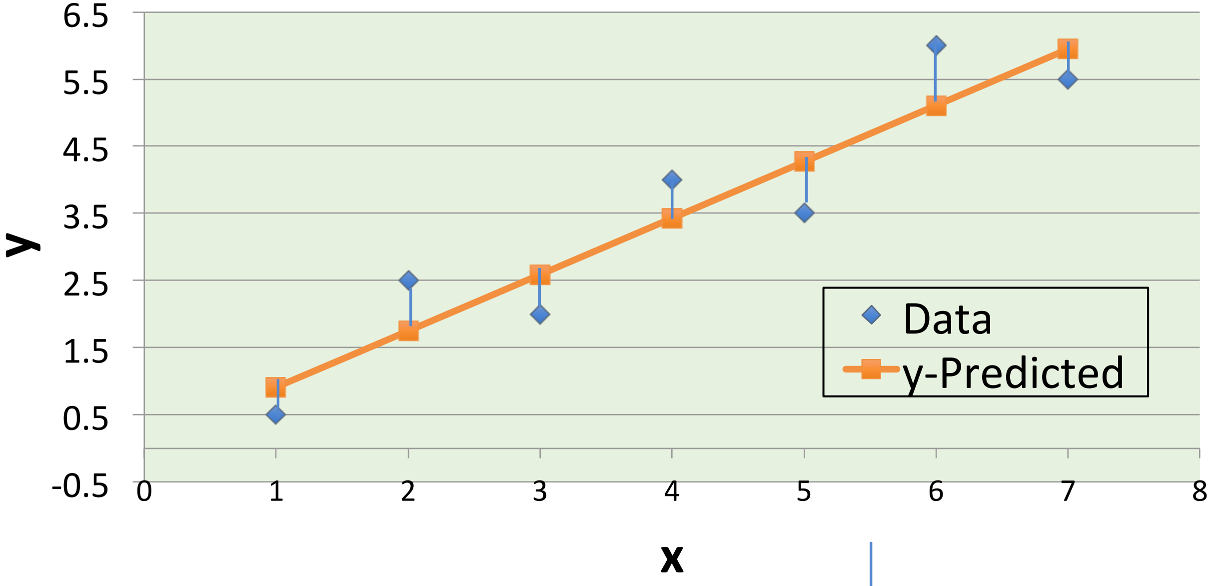
$$b_1 = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum y_i^2 - \left(\sum y_i\right)^2}$$

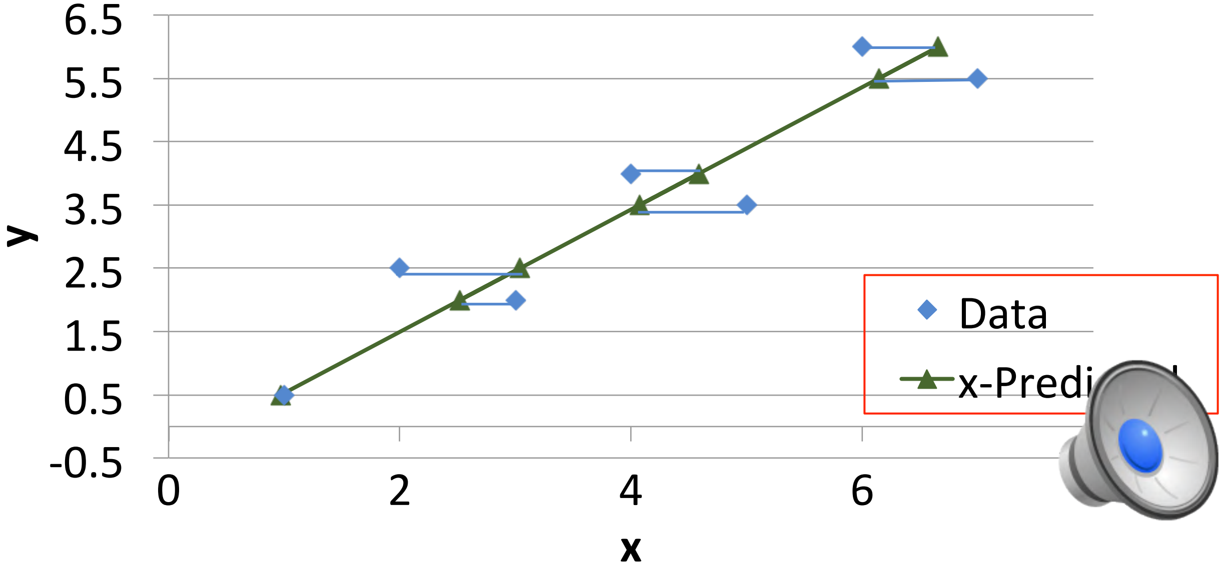
$$b_0 = \bar{x} - a_1 \bar{y}$$

b_1	b_0
1.0346	0.4528

No of Data Point	x_i	y_i	y^2_i	$x_i y_i$	$x_{predicted}$
1	1	0.5	0.25	0.50	0.97
2	2	2.5	6.25	5.00	3.04
3	3	2.0	4.00	6.00	2.52
4	4	4.0	16.00	16.00	4.59
5	5	3.5	12.25	17.50	4.07
6	6	6.0	36.00	36.00	6.66
7	7	5.5	30.25	38.50	6.14
	28	24	105	119.5	







NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Regression Analysis and Curve Fitting

Multi-linear Regression

Prof. Jayati Sarkar

$$(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$$

$$y = a_o + a_1 x$$

$$x = b_o + b_1 y$$

$$\frac{x}{b_1} = \frac{b_o}{b_1} + y$$

$$y = \left(\frac{1}{b_1}\right) x + \left(-\frac{b_o}{b_1}\right)$$

No of Data Point	x_i	y_i	x^2_i	y^2_i	$x_i y_i$	$y_{predicted}$	$x_{predicted}$
1	1	0.5	1.00	0.25	0.50	0.91	0.97
2	2	2.5	4.00	6.25	5.00	1.75	3.04
3	3	2.0	9.00	4.00	6.00	2.59	2.52
4	4	4.0	16.00	16.00	16.00	3.43	4.59
5	5	3.5	25.00	12.25	17.50	4.27	4.07
6	6	6.0	36.00	36.00	36.00	5.11	6.66
7	7	5.5	49.00	30.25	38.50	5.95	6.14
	28	24	140	105	119.5		

$$y = a_0 + a_1 x$$

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

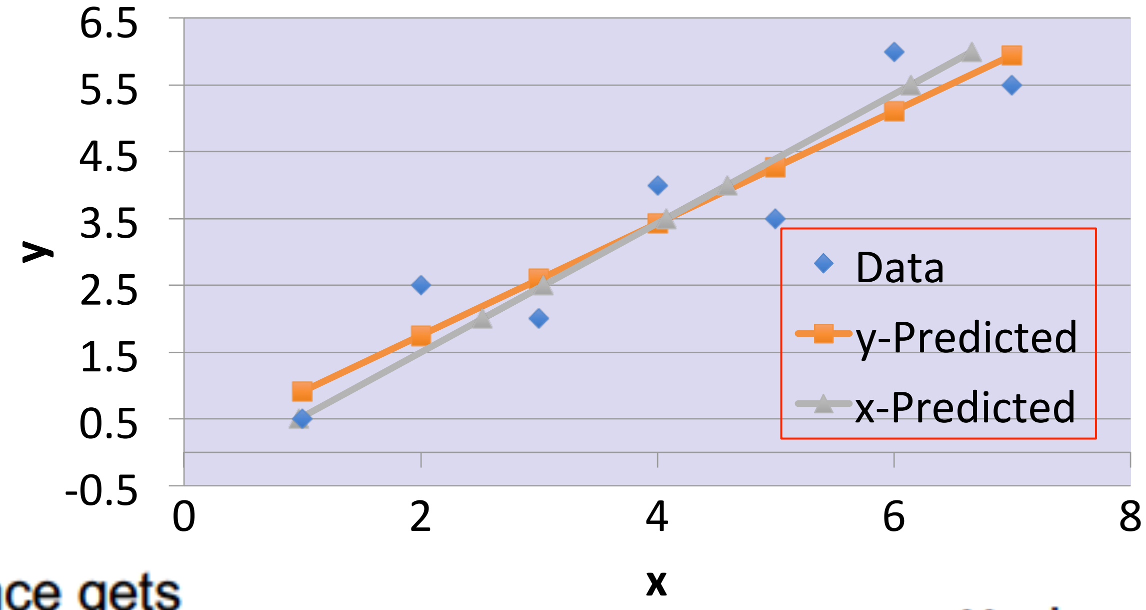
$$a_0 = \bar{y} - a_1 \bar{x}$$

$$x = b_0 + b_1 y$$

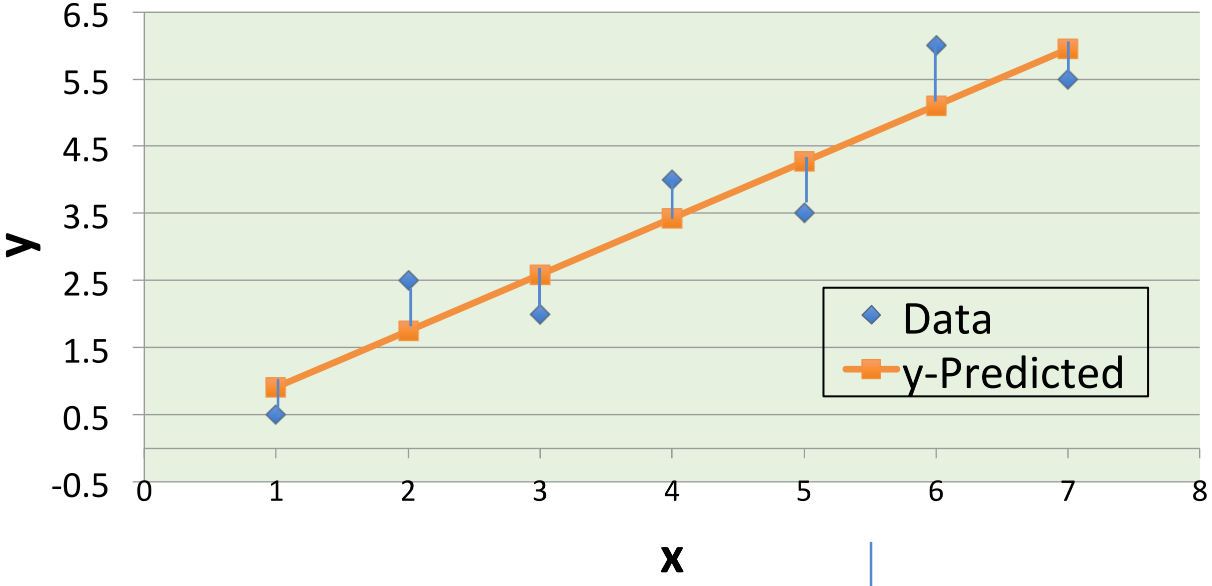
$$b_1 = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum y_i^2 - \left(\sum y_i\right)^2}$$

$$b_0 = \bar{x} - a_1 \bar{y}$$

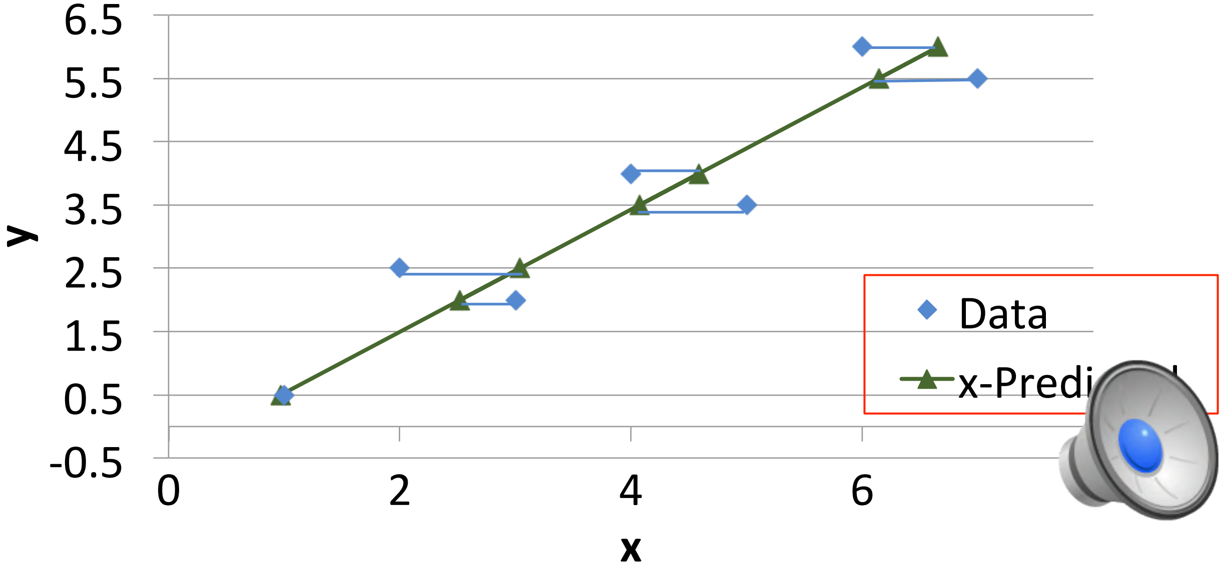
a_1	a_0	b_1	b_0	$a_0(P)=-b_0/b_1$	$a_1(P)=$
0.8393	0.0714	1.0346	0.4528	-0.4377	0.96



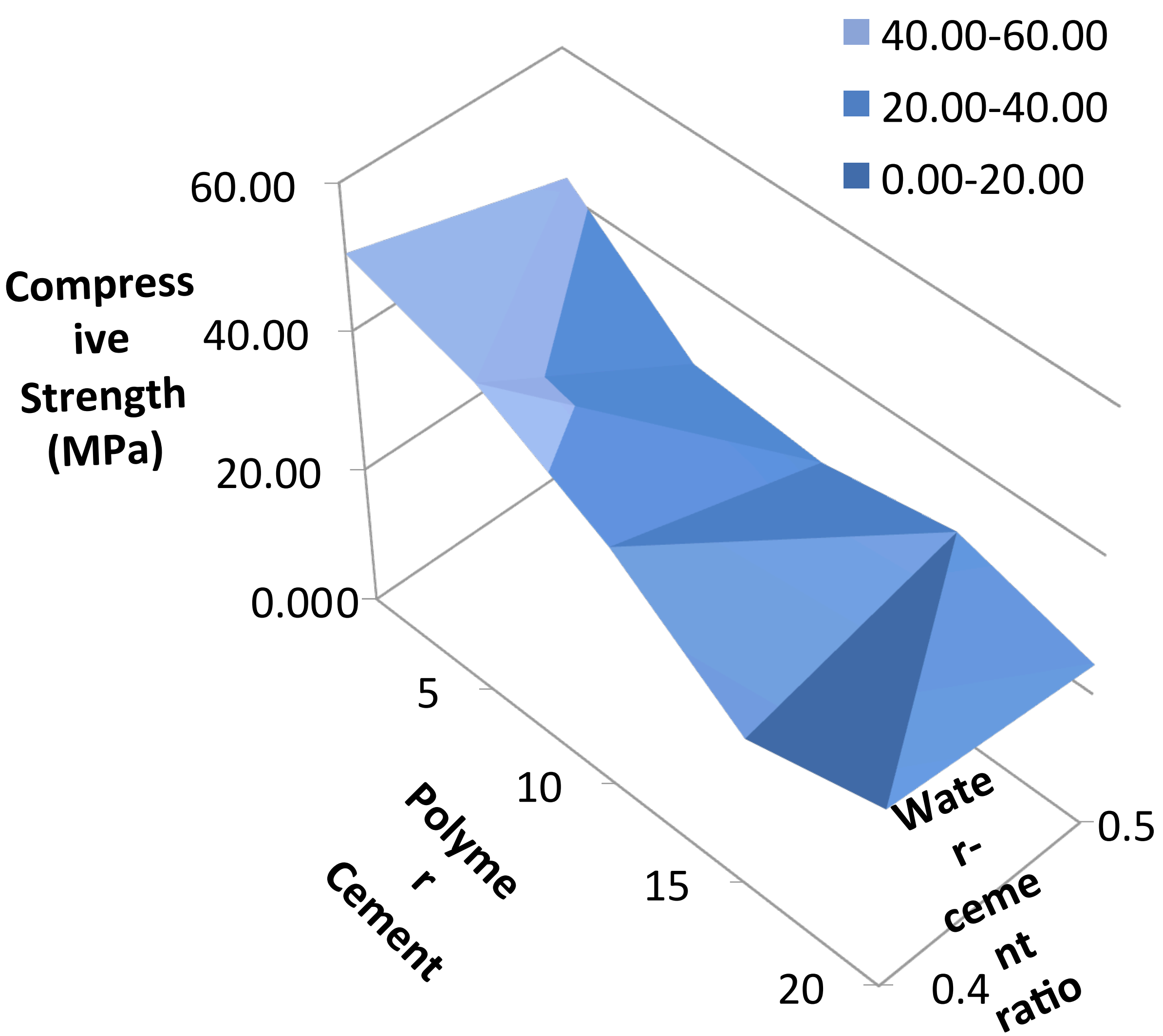
Vertical distance gets minimizes

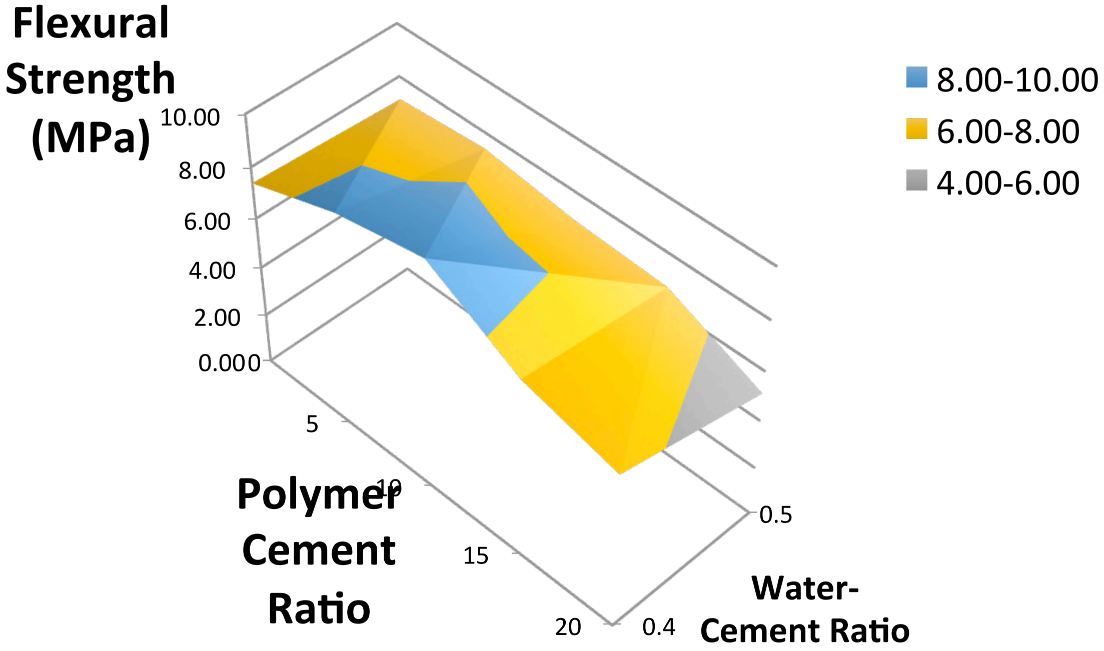


Horizontal distances
Get minimized



	Polymer-cement ratio (x_1)	Water-cement ratio (x_2)	Compressive Strength (MPa) (y_1)	Flexural Strength (MPa)(y_2)
Pure cemet mortar	0	0.4	50.63	7.39
50/50-5	5	0.4	45.31	8.59
50/50-10	10	0.4	35.63	9.30
50/50-15	15	0.4	22.21	7.30
50/50-20	20	0.4	27.18	6.40
Pure cemet mortar	0	0.5	42.40	7.11
50/50-5	5	0.5	27.87	7.37
50/50-10	10	0.5	26.30	6.90
50/50-15	15	0.5	29.60	6.70
50/50-20	20	0.5	24.27	5.10





Multilinear Regression

Data: (x_i, u_i, w_i, y_i) i = 1 to N

Model: $\hat{y} = a_o + a_1 x + a_2 u + a_3 w$

Error: $e_i = y_i - \hat{y}_i = y_i - [a_o + a_1 x_i + a_2 u_i + a_3 w_i]$

Objective: To find a_o, a_1, a_2, a_3 such that

$$S_r = \sum_{min} e_i^2$$

$$min \sum_{i=1}^{N} (y_i - [a_o + a_1 x_i + a_2 u_i + a_3 w_i])^2$$

$$\frac{\partial S_r}{\partial a_o} = \sum -2(y_i - a_o - a_1 x_i - a_2 u_i - a_3 w_i) = 0$$

$$\frac{\partial S_r}{\partial a_1} = \sum -2x_i(y_i - a_o - a_1 x_i - a_2 u_i - a_3 w_i) = 0$$

$$\frac{\partial S_r}{\partial a_2} = \sum -2u_i e_i = 0$$

$$\frac{\partial S_r}{\partial a_3} = \sum -2w_i e_i = 0$$

$$\sum y_i = a_o N + a_1 \sum x_i + a_2 \sum u_i + a_3 \sum w_i = 0$$

$$\sum y_i x_i = a_o \sum x_i + a_1 \sum x_i x_i + a_2 \sum u_i x_i + a_3 \sum w_i x_i = 0$$

$$\sum y_i u_i = a_o \sum u_i + a_1 \sum x_i u_i + a_2 \sum u_i u_i + a_3 \sum w_i u_i = 0$$

$$\sum y_i w_i = a_o \sum w_i + a_1 \sum x_i w_i + a_2 \sum u_i w_i + a_3 \sum w_i w_i = 0$$

Solve by Gauss Elimination/Gauss Seidel Method

$$\begin{bmatrix} N & \sum x_i & \sum u_i & \sum w_i \\ \sum x_i & \sum x_i x_i & \sum u_i x_i & \sum w_i x_i \\ \sum u_i & \sum x_i u_i & \sum u_i u_i & \sum w u_i \\ \sum w_i & \sum x_i w_i & \sum u_i w_i & \sum w_i w_i \end{bmatrix} \begin{bmatrix} a_o \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum y_i x_i \\ \sum y_i u_i \\ \sum y_i w_i \end{bmatrix}$$

	Polymer-cement ratio (x)	Water-cement ratio (u)	Compressive Strength (MPa) (y)	x^2	u^2	xu	yx	yu
Pure cemet mortar	0	0.4	50.63	0	0.16	0	0	20.252
50/50-5	5	0.4	45.31	25	0.16	2	226.55	18.124
50/50-10	10	0.4	35.63	100	0.16	4	356.3	14.252
50/50-15	15	0.4	22.21	225	0.16	6	333.15	8.884
50/50-20	20	0.4	27.18	400	0.16	8	543.6	10.872
Pure cemet mortar	0	0.5	42.40	0	0.25	0	0	21.2
50/50-5	5	0.5	27.87	25	0.25	2.5	139.35	13.935
50/50-10	10	0.5	26.30	100	0.25	5	263	13.15
50/50-15	15	0.5	29.60	225	0.25	7.5	444	14.8
50/50-20	20	0.5	24.27	400	0.25	10	485.4	12.135
N=10	100	4.5	331.4	1500	2.05	45.00	2791.35	147.60

$$\begin{bmatrix} N & \sum x_i & \sum u_i \\ \sum x_i & \sum x_i x_i & \sum u_i x_i \\ \sum u_i & \sum x_i u_i & \sum u_i u_i \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum y_i x_i \\ \sum y_i u_i \end{bmatrix}$$

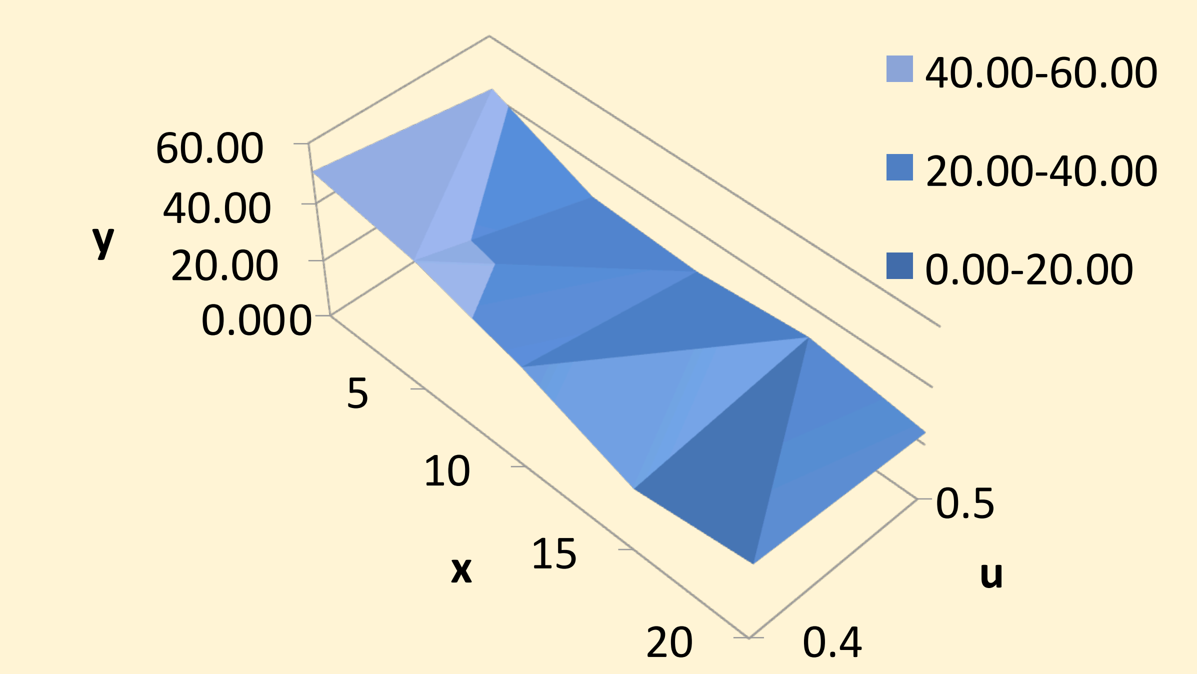
$$\begin{bmatrix} 10 & 100 & 4.5 \\ 100 & 1500 & 45 \\ 4.5 & 45 & 2.05 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 331.4 \\ 2791.35 \\ 147.6 \end{bmatrix}$$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 71.1333 \\ -1.0453 \\ -61.200 \end{bmatrix}$$

$$\boxed{y = 71.1333 - 1.0453x - 61.200u}$$

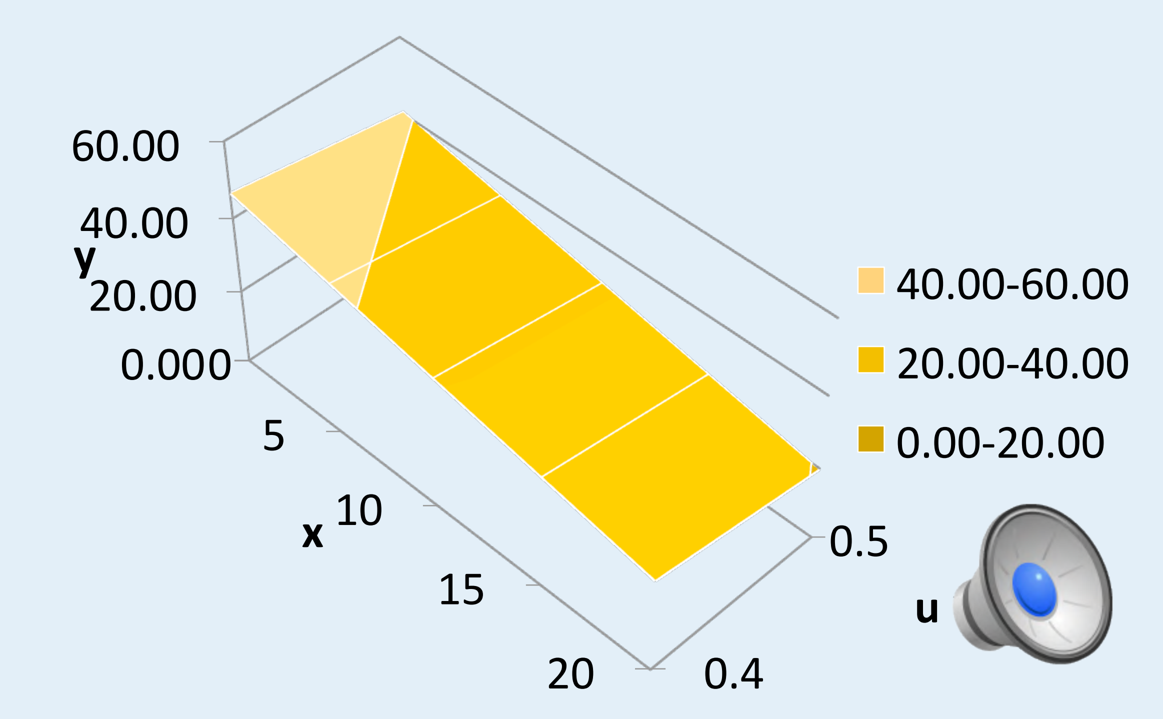
Actual

u\x	0	5	10	15	20
0.4	50.63	45.31	35.63	22.21	27.18
0.5	42.40	27.87	26.30	29.60	24.27



Predicted

u\x	0	5	10	15	20
0.4	46.65	41.43	36.20	30.97	25.75
0.5	40.53	35.31	30.08	24.85	19.63



$$\boxed{y = 71.1333 - 1.0453x - 61.200u}$$

Matrix Least Squares Method

$$(x_1, y_1) \quad (x_2, y_2) \ldots \ldots . \quad \Rightarrow \quad \hat{y} = a_o + a_1 x$$

$$y_1 = a_o + a_1 x_i + e_i$$

$$\hat{y}_1 = 1.\, a_o + x_1.\, a_1$$

$$\hat{y}_2 = 1.\, a_o + x_2.\, a_1$$

$$\hat{y}_3 = 1.\, a_o + x_3.\, a_1$$

$$\hat{y}_4 = 1.\, a_o + x_4.\, a_1$$

$$\begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix} \begin{bmatrix} a_o \\ a_1 \end{bmatrix}$$

$$\underset{\hat{y}}{\underset{\Downarrow}{\begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix}}} = \underset{X}{\underset{\Downarrow}{\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix}}} \underset{\phi}{\underset{\Downarrow}{\begin{bmatrix} a_o \\ a_1 \end{bmatrix}}}$$

Matrix Least Squares Method

$$(x_1, y_1) \quad (x_2, y_2) \ldots \ldots\ldots \quad \Rightarrow \quad \hat{y} = a_o + a_1 x$$

$$y_1 = a_o + a_1 x_i + e_i$$

$$\hat{y}_1 = 1.a_o + x_1.a_1$$

$$\hat{y}_2 = 1.a_o + x_2.a_1$$

$$\hat{y}_3 = 1.a_o + x_3.a_1$$

$$\hat{y}_4 = 1.a_o + x_4.a_1$$

$$\begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix} \begin{bmatrix} a_o \\ a_1 \end{bmatrix}$$

$$\begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_N \end{bmatrix} \begin{bmatrix} a_o \\ a_1 \end{bmatrix}$$

$$\begin{array}{ccc} \Downarrow & \Downarrow & \Downarrow \\ \hat{y} & X & \phi \end{array}$$

Least square solution that minimizes $\sum e_i^2$

$$\underline{Y} = X\,\underline{\emptyset} + \underline{e}$$

$$\underline{e} = \underline{Y} - X\,\underline{\emptyset}$$

$$S(\underline{\emptyset}) = \sum e_i^2$$

$$= (\underline{Y} - X\,\underline{\emptyset})^T(\underline{Y} - X\,\underline{\emptyset})$$

$$= \underline{Y}^T\underline{Y} - \underline{Y}^T X\underline{\emptyset} - (X\underline{\emptyset})^T\underline{Y} + (X\underline{\emptyset})^T(X\underline{\emptyset})$$

$$= \underline{Y}^T\underline{Y} - \underline{Y}^T X\underline{\emptyset} - \underline{\emptyset}^T X^T\underline{Y} + \underline{\emptyset}^T X^T X\underline{\emptyset}$$

$$\frac{\partial s}{\partial \emptyset} = -2X^T\underline{y} + 2X^T X\underline{\emptyset} = 0$$

$$\Rightarrow (X^T X)\underline{\emptyset} = X^T\underline{Y}$$

$$\Rightarrow \underline{\emptyset} = (X^T X)^{-1} X^T\underline{Y}$$

No of Data Point	x_i	y_i
1	1	0.5
2	2	2.5
3	3	2.0
4	4	4.0
5	5	3.5
6	6	6.0
7	7	5.5
	28	24

a_1	a_0
0.8393	0.0714

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 1 & 5 \\ 1 & 6 \\ 1 & 7 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 2.5 \\ 2.0 \\ 4.0 \\ 3.5 \\ 6.0 \\ 5.5 \end{bmatrix}$$

$$\boxed{e + X\Phi = Y}$$

$$\emptyset = (X^T X)^{-1} X^T Y$$

Model: $\hat{y} = a_o + a_1 x + a_2 u + a_3 w$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} 1 & x_1 & u_1 & w_1 \\ 1 & x_2 & u_2 & w_2 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_N & u_N & w_N \end{bmatrix} \begin{bmatrix} a_o \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_N \end{bmatrix}$$

$$\boxed{Y = X\Phi + e}$$

Least Square = $(X^T X)^{-1} X^T Y$

Polynomial Fitting

$$C_p = a_o + a_1 T + a_2 T^2 + a_3 T^3$$

$\rightarrow$ Dependent variable are non linear functions of independent variable

Data : $(T_1, C_{p_1}), (T_2, C_{p_2}), \ldots, (T_3, C_{p_3})(T_N, C_{pN})$

This gets converted to

$$y_1 = a_o + a_1 x_1 + a_2 u_1 + a_3 w_1$$

$$\underline{y} = \begin{bmatrix} C_{p1} \\ C_{p2} \\ \vdots \\ C_{pN} \end{bmatrix} \quad X = \begin{bmatrix} 1 & T_1 & T_1^2 & T_1^3 \\ 1 & T_2 & T_2^2 & T_2^3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & T_N & T_N^2 & T_N^3 \end{bmatrix} \begin{matrix} \cdots & T_1^m \\ \cdots & T_2^m \\ & \\ \cdots & T_N^m \end{matrix}$$

$\downarrow$

Extend to m[th] order polynomial

$$\begin{bmatrix} a_o \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = (X^T X)^{-1} X^T Y \rightarrow left\ inverse\ of\ matrix\ X$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Regression Analysis and Curve Fitting

Functional and non-linear Regression

Prof. Jayati Sarkar

Multilinear Regression

Data: (x_i, u_i, w_i, y_i) i = 1 to N

Model: $\hat{y} = a_o + a_1 x + a_2 u + a_3 w$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} 1 & x_1 & u_1 & w_1 \\ 1 & x_2 & u_2 & w_2 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_N & u_N & w_N \end{bmatrix} \begin{bmatrix} a_o \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_N \end{bmatrix}$$

$$\boxed{Y = X\Phi + e}$$

$$\boxed{\Phi} = (X^T X)^{-1} X^T Y$$

$$s_{y/x} = \sqrt{\frac{S_r}{N - (m+1)}}$$

Standard Error

Polynomial Fitting

$$C_p = a_o + a_1 T + a_2 T^2 + a_3 T^3$$

→ Dependent variable are non linear functions of independent variable

Data : $(T_1, C_{p_1}), (T_2, C_{p_2}), \ldots, (T_3, C_{p_3})(T_N, C_{pN})$

This gets converted to

$$y_1 = a_o + a_1 x_1 + a_2 u_1 + a_3 w_1$$

$$\underline{y} = \begin{bmatrix} C_{p1} \\ C_{p2} \\ \vdots \\ C_{pN} \end{bmatrix} \quad X = \begin{bmatrix} 1 & T_1 & T_1^2 & T_1^3 \\ 1 & T_2 & T_2^2 & T_2^3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & T_N & T_N^2 & T_N^3 \end{bmatrix} \begin{matrix} \cdots & T_1^m \\ \cdots & T_2^m \\ & \\ \cdots & T_N^m \end{matrix}$$

↓

Extend to m^th^ order polynomial

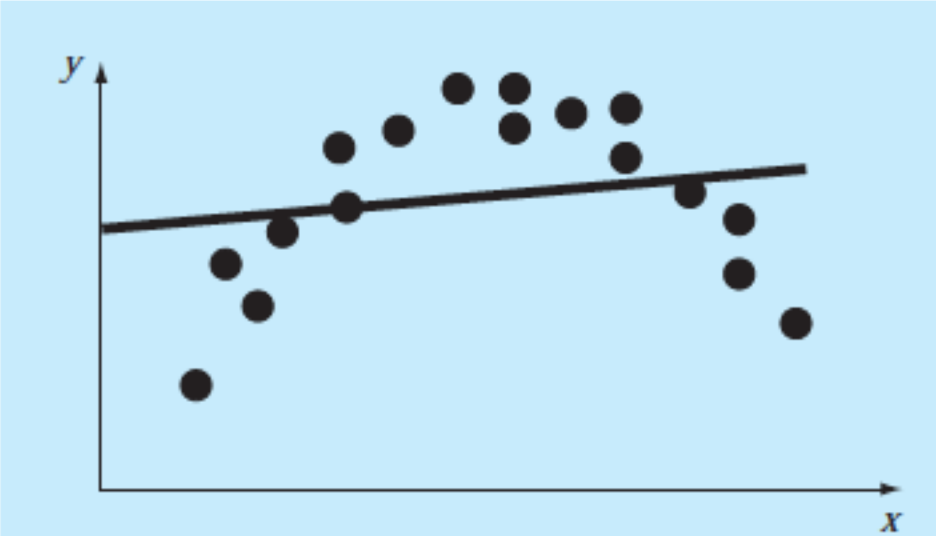
$$\begin{bmatrix} a_o \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = (X^T X)^{-1} X^T Y \ \rightarrow left\ inverse\ of\ matrix\ X$$

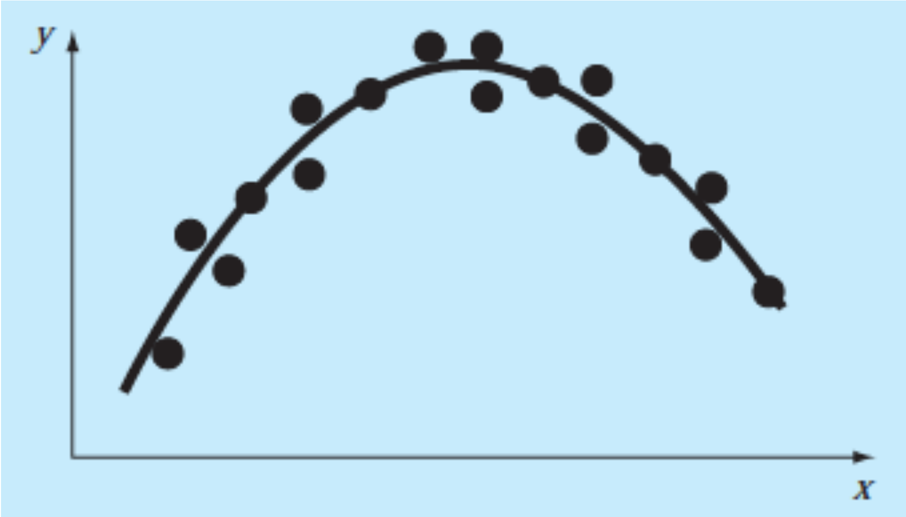
Polynomial Fitting

Note:

1. m << N
2. If m = N-1 overfitting of polynomial
3. As m increases matrix X becomes ill condition.
4. If $m \geq 6 \;\Rightarrow\; \left|\frac{\lambda_{max}}{\lambda_{min}}\right| \sim 10^{10} \;\rightarrow$ should not be used

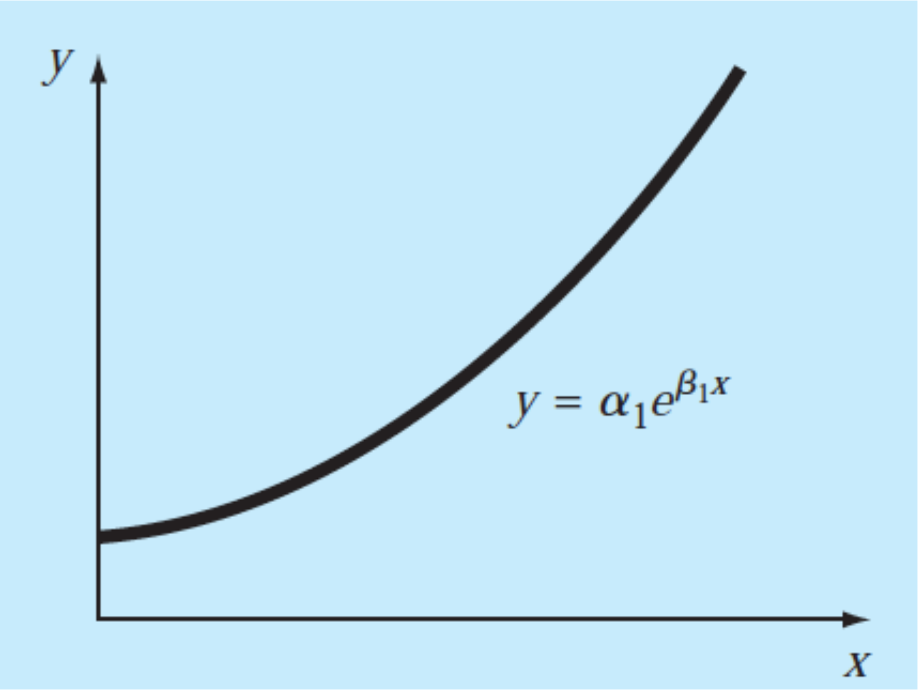
Functional Regression





Exponential Model

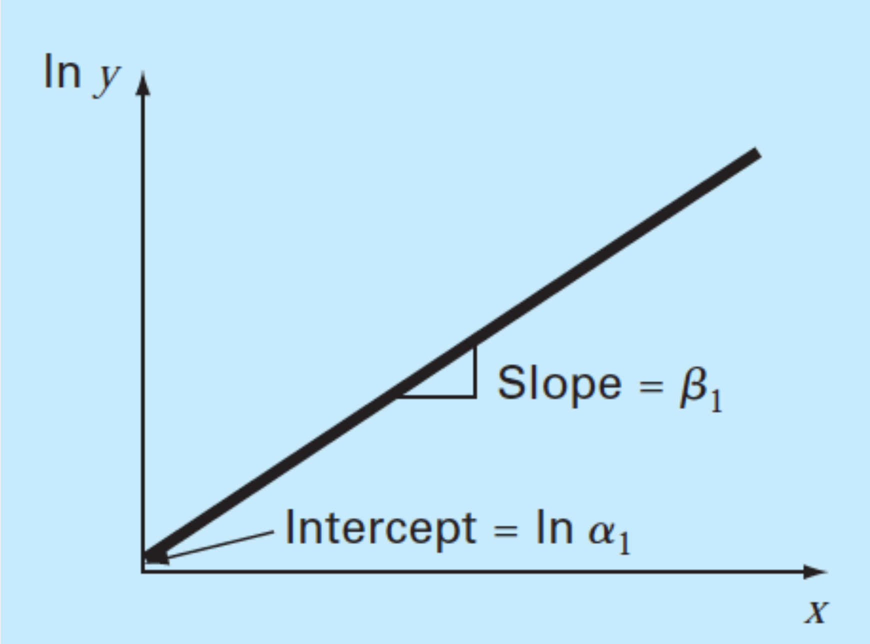
$$y = \alpha_1 e^{\beta_1 x}$$



Linearization

$$\ln(y) = \ln(\alpha_1) + \beta_1 x$$

$$Y = a_0 + a_1 x$$

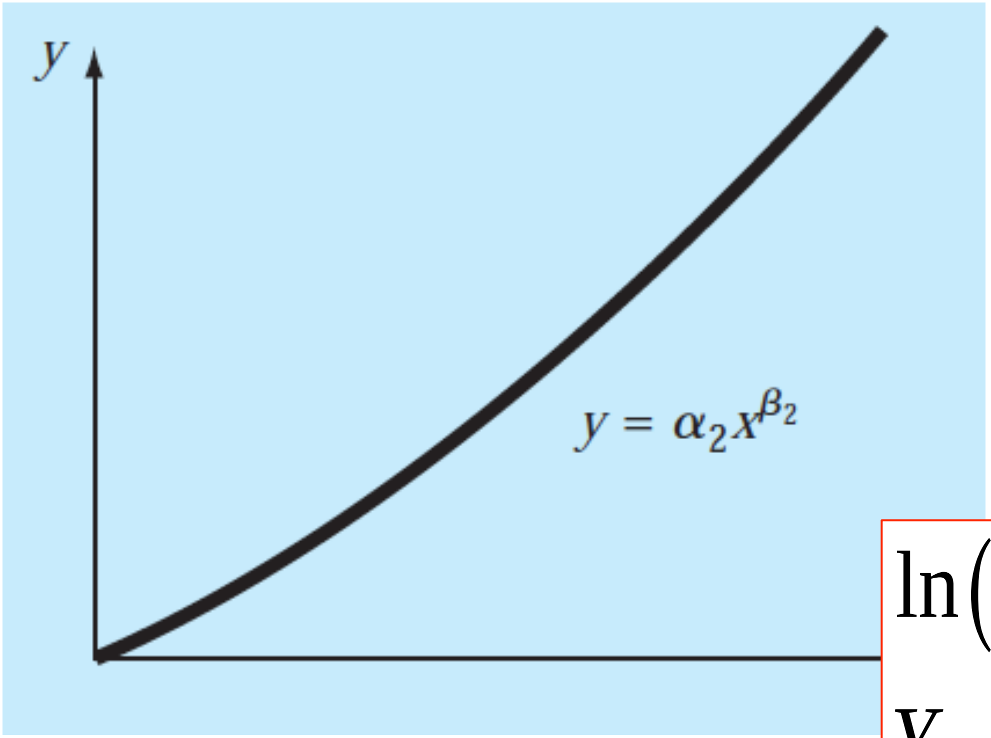


Example: Population growth, Radioactivity Decrease, Arrhenius Law

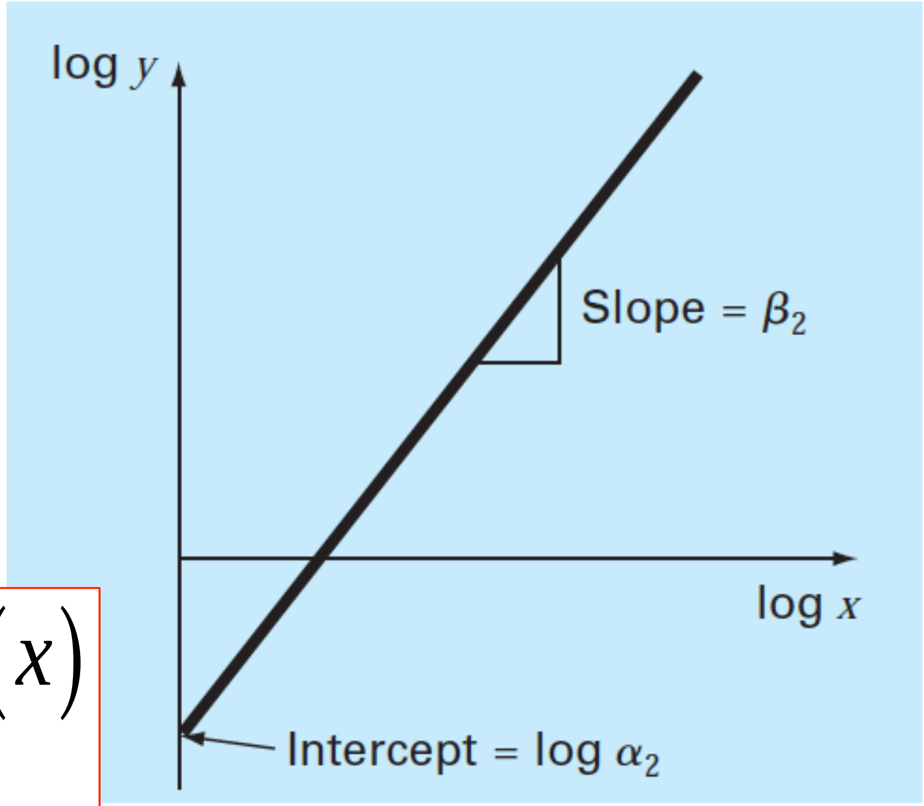
$$k = k_o e^{-E/RT} \Rightarrow \underbrace{lnk}_{\text{rate constant}} = lnk_o + \underbrace{\left(\frac{E}{R}\right)}_{\text{activation energy}}\left(-\frac{1}{T}\right)$$

Power Law Model

$$y = \alpha_2 x^{\beta_2}$$



Linearization



$$\ln(y) = \ln(\alpha_2) + \beta_2 \ln(x)$$

$$Y = a_0 + a_1 X$$

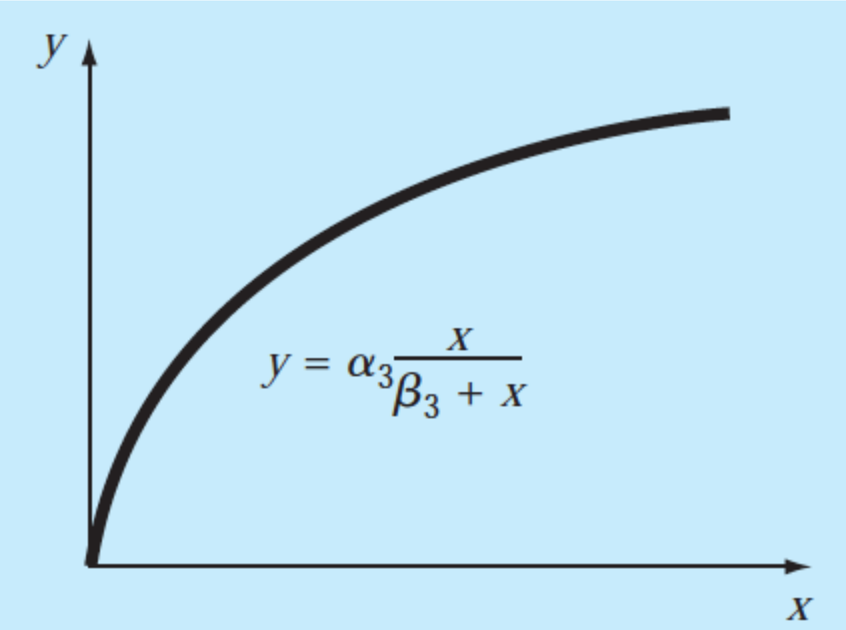
$$\mu = m\gamma^{-n-1}$$

$$Y = a_0 + a_1 X$$

Example: non-Newtonian fluid flow

Saturation Growth Rate

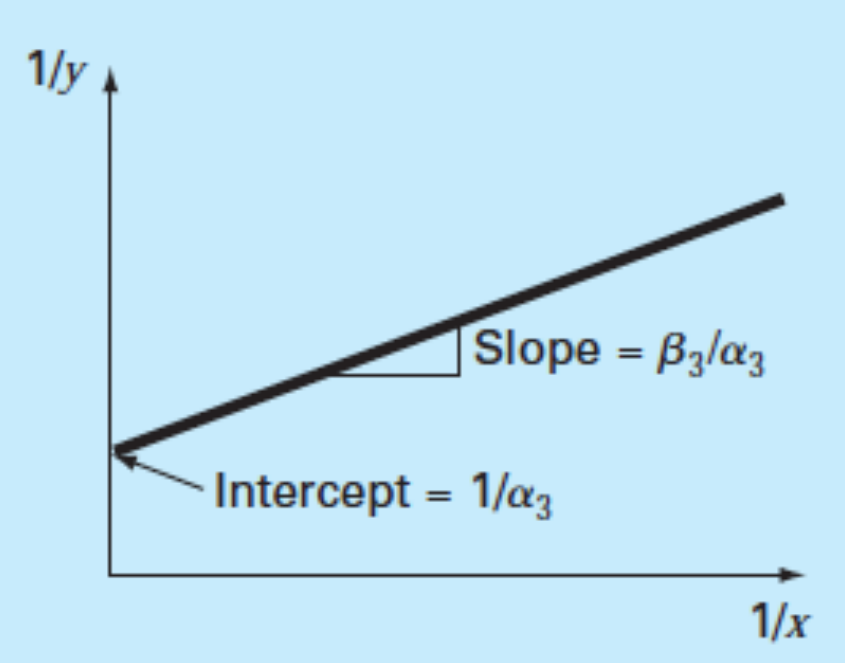
$$y = \alpha_3 \frac{x}{\beta_3 + x}$$



Linearization

$$\frac{1}{y} = \frac{1}{\alpha_3} + \beta_3 \frac{1}{x}$$

$$Y = a_0 + a_1 X$$



Example: Enzymatic kinetics

$$r = \frac{k\,S}{k_m + S}$$

$r = rate, S = concentration\ of\ substrate,$

$k_m = saturation\ constant$

Nonlinear Regression

$$f(x) = a_o(1 - e^{-a_1 x})$$

$$y_i = f(x_i, a_o, a_1, \ldots.., a_m) + e_i$$

y_i ↓ Measured value

x_i ↓ Independent variable

$a_1, \ldots$ ↓ Nonlinear function of parameters $a_o, \ldots \ldots$

e_i ↓ Random error

$$y_i = f(x_i) + e_i \rightarrow shorter\ representation \ldots\ldots (1)$$

Nonlinear model can be expanded in a Taylor series

$$f(x_i)^{j+1} = f(x_i)^j + \left.\frac{\partial f(x_i)}{\partial a_o}\right|^j \Delta a_o + \left.\frac{\partial f(x_i)}{\partial a_1}\right|^j \Delta a_1 \ldots. (2)$$

$j + 1 = current\ iteration$ $\qquad \Delta a_o = a_o^{j+1} - a_o^j$

$j = previous\ iteration$ $\qquad \Delta a_1 = a_1^{j+1} - a_1^j$

$$y_i - f(x_i)^j = \left.\frac{\partial f(x_i)}{\partial a_o}\right|^j \Delta a_o + \left.\frac{\partial f(x_i)}{\partial a_1}\right|^j \Delta a_1 + e_i$$

MATRIX FORM:

$$[D] = [Z^j][\Delta A] + [E]$$

$\downarrow$

Z calculated at previous guess

$$[D] = \begin{bmatrix} y_1 - f(x_1)^j \\ y_2 - f(x_2)^j \\ \vdots \\ y_N - f(x_N)^j \end{bmatrix} \quad [Z^j] = \begin{bmatrix} \frac{\partial f_1}{\partial a_o} & \frac{\partial f_1}{\partial a_1} \\ \frac{\partial f_2}{\partial a_o} & \frac{\partial f_2}{\partial a_1} \\ \vdots & \vdots \\ \frac{\partial f_N}{\partial a_o} & \frac{\partial f_N}{\partial a_N} \end{bmatrix}^j \quad \Delta A = \begin{bmatrix} \Delta a_o \\ \Delta a_1 \\ \vdots \\ \Delta a_N \end{bmatrix}$$

N= no. of data points

For $\frac{\partial f_i}{\partial a_k}$,

$i = iteration\ data\ point$

$k = k^{th} point$

$j = previous\ iteration$

Applying least square theory

$$\left[[Z^j]^T[Z^j]\right][\Delta A] = \left[\{Z^j\}^T\{D\}\right]$$

$$\Downarrow$$

$$a_o^{j+1} = a_o^j + \Delta a_o$$

$$a_1^{j+1} = a_1^j + \Delta a_1$$

$$max\left|\frac{a_k^{j+1} - a_k^j}{a_k^j}\right| < \varepsilon_{tol}$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Curve Fitting

Newtons Divided Difference Formula

Prof. Jayati Sarkar

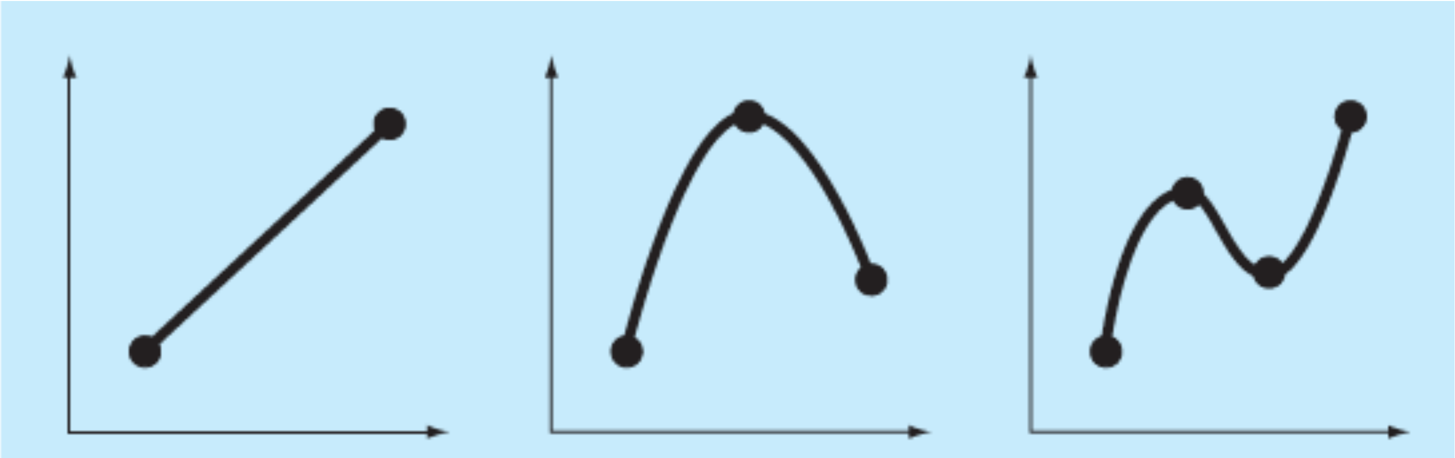
Curve Fitting and Interpolation

- To estimate intermediate values between precise data points.

$$\left(x_1, y_1\right)\left(x_2, y_2\right)........\left(x_n, y_n\right)$$

- The most common method used for this purpose is polynomial interpolation by fitting a (n-1)th order polynomial.

$$Y = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1}$$



Curve Fitting and Interpolation

$$\left(x_1, y_1\right)\left(x_2, y_2\right)\ldots\ldots\left(x_n, y_n\right) \qquad Y = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1} x^{n-1}$$

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ y_n \end{bmatrix} \qquad \boldsymbol{X} = \begin{bmatrix} 1 & x_1 & x_1^2 \ldots\ldots\ldots x_1^{n-1} \\ 1 & x_2 & x_2^2 \ldots\ldots\ldots x_2^{n-1} \\ \vdots & \vdots & \vdots\ldots\ldots\ldots \quad \vdots \\ \vdots & \vdots & \vdots\ldots\ldots\ldots \quad \vdots \\ 1 & x_n & x_n^2 \ldots\ldots\ldots x_n^{n-1} \end{bmatrix}_{n \times n}$$

$$\boxed{\begin{aligned} \Phi &= \left(X^T X\right)^{-1} X^T Y \\ &= X^{-1}\left(X^T\right)^{-1} X^T Y \\ &= X^{-1} Y \end{aligned}}$$

Inverting X becomes difficult as n goes beyond 6

Curve Fitting and Interpolation

S.no	Temp	P
1	0	0.0002
2	20	0.0012
3	40	0.0060
4	60	0.0300
5	80	0.0900
6	100	0.2700

Y	X					
0.0002	1	0	0	0	0	0
0.0012	1	20	400	8000	160000	3200000
0.0060	1	40	1600	64000	2560000	102400000
0.0300	1	60	3600	216000	12960000	777600000
0.0900	1	80	6400	512000	40960000	3276800000
0.2700	1	100	10000	1000000	100000000	10000000000

Phi

a_0=0.0002

a_1=0.000852167

a_2=-8.14375E-05

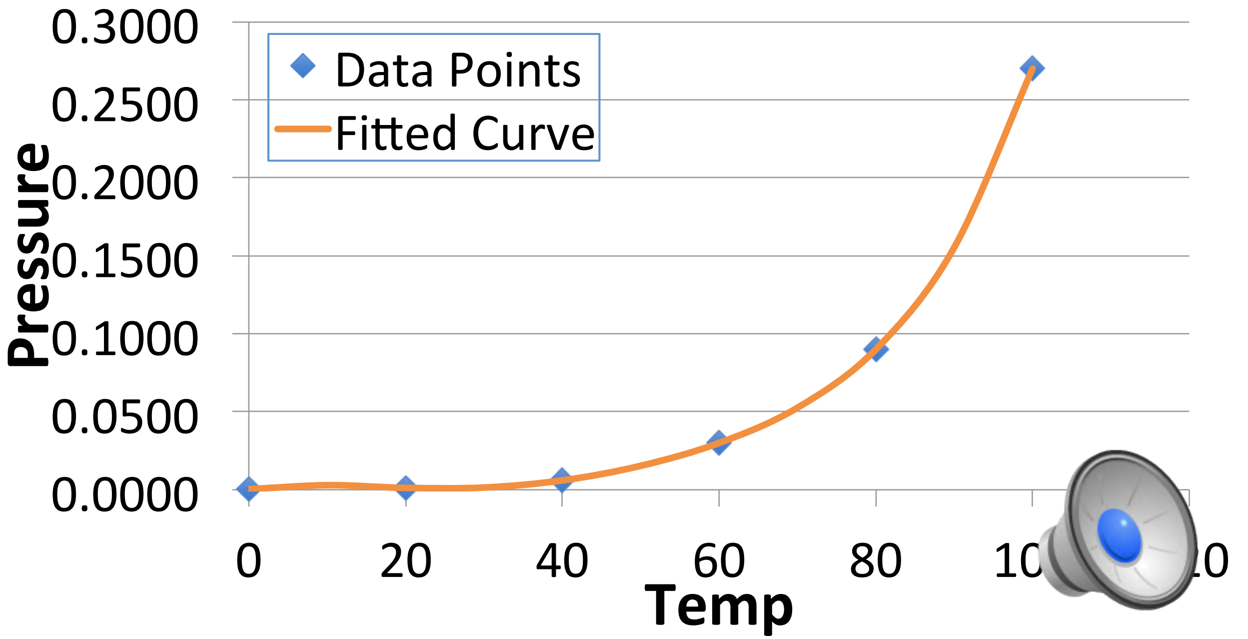
a_3=2.67604E-06

a_4=-3.39063E-08

a_5=1.71354E-10

$$\Phi = X^{-1}Y$$

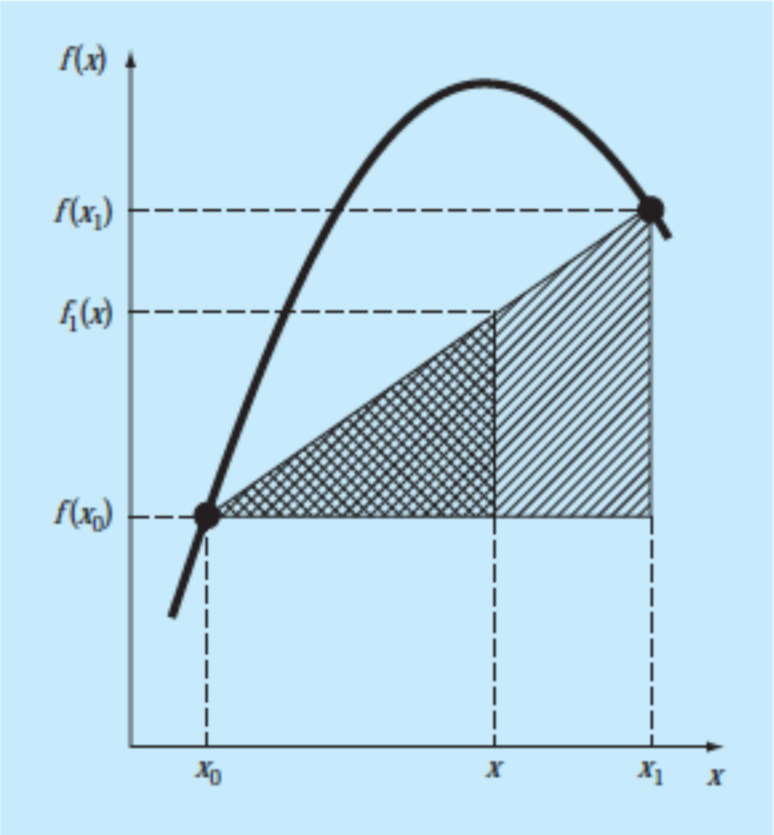
$$Y = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5$$



Newton`s Divided-Difference Interpolating Polynomials

Linear Interpolation

The simplest form of interpolation is to connect two data points with a straight line. Using similar triangles



$$\frac{f_1(x)-f(x_0)}{x-x_0}=\frac{f(x_1)-f(x_0)}{x_1-x_0}$$

finite-divided-difference

$$f_1(x)=f(x_0)+\frac{f(x_1)-f(x_0)}{(x_1-x_0)}(x-x_0)$$

Newton`s Divided-Difference Interpolating Polynomials

Linear Interpolation

$$f_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{(x_1 - x_0)}(x - x_0)$$

X	y=ln(x)
1	0.00
2	0.69
3	1.10
4	1.39
5	1.61
6	1.79

Series1

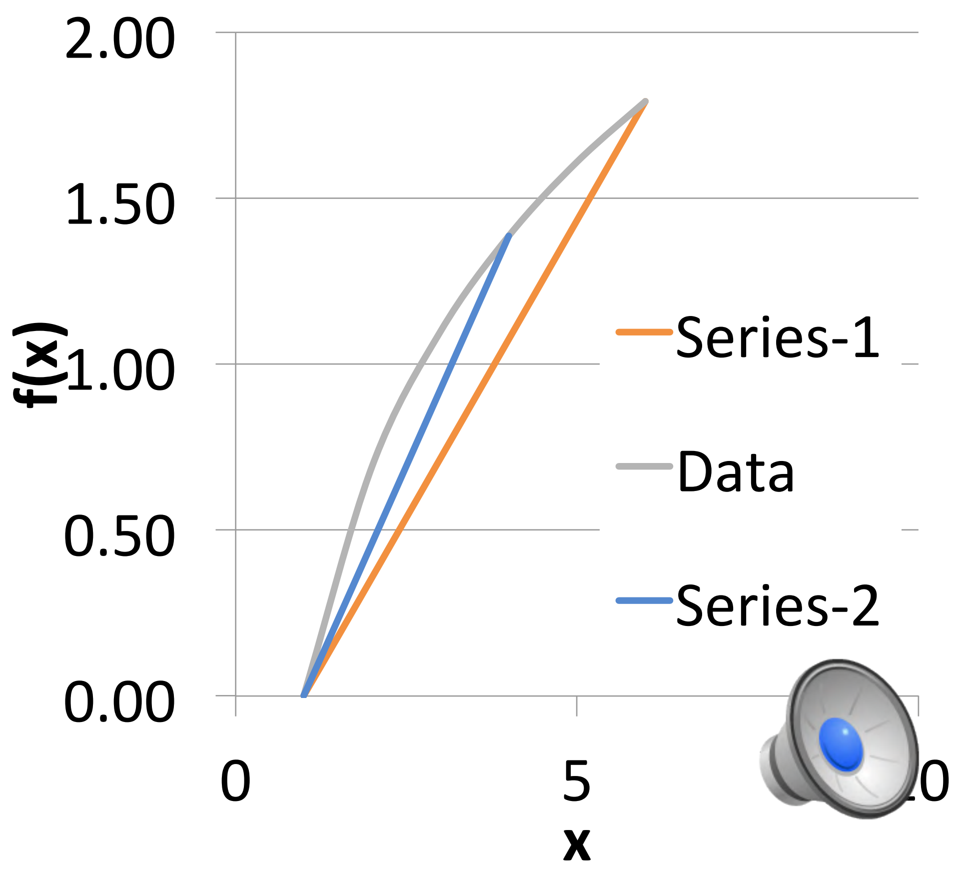
x_0	x_1
1	6.00
$f(x_0)$	$f(x_1)$
0	1.79

Series2

x_0	x_1
1	4.00
$f(x_0)$	$f(x_1)$
0	1.39

X	Series1: $f_1(x)$	Series2: $f_1(x)$
1	0.00	0.00
2	0.36	0.46
3	0.72	0.92
4	1.08	1.39
5	1.43	1.85
6	1.79	2.31

	Error(2)
Series1	48.30
Series2	33.33



Newton`s Divided-Difference Interpolating Polynomials

Quadratic Interpolation

If three data points are available, this can be accomplished with a second-order polynomial (also called a quadratic polynomial or a parabola).

$$f_2(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1)$$

$$= b_0 + b_1x - b_1x_0 + b_2x^2 + b_2x_0x_1 - b_2xx_0 - b_2xx_1$$

$$\boxed{\begin{aligned} a_0 &= b_0 - b_1x_0 + b_2x_0x_1 \\ a_1 &= b_1 - b_2x_0 - b_2x_1 \\ a_2 &= b_2 \end{aligned}}$$

$$f_2(x) = a_0 + a_1x + a_2x^2$$

Newton`s Divided-Difference Interpolating Polynomials

Quadratic Interpolation

$$f_2(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1)$$

$$b_0 = f(x_0)$$

$$b_1 = \frac{f(x_1) - f(x_0)}{(x_1 - x_0)}$$

Linear Interpolation

$$b_2 = \frac{\dfrac{f(x_2) - f(x_1)}{(x_2 - x_1)} - \dfrac{f(x_1) - f(x_0)}{(x_1 - x_0)}}{(x_2 - x_0)}$$

Second Order Curvature

Newton`s Divided-Difference Interpolating Polynomials

Quadratic Interpolation

$$f_2(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1)$$

$$b_0 = f(x_0)$$

$$b_1 = \frac{f(x_1) - f(x_0)}{(x_1 - x_0)}$$

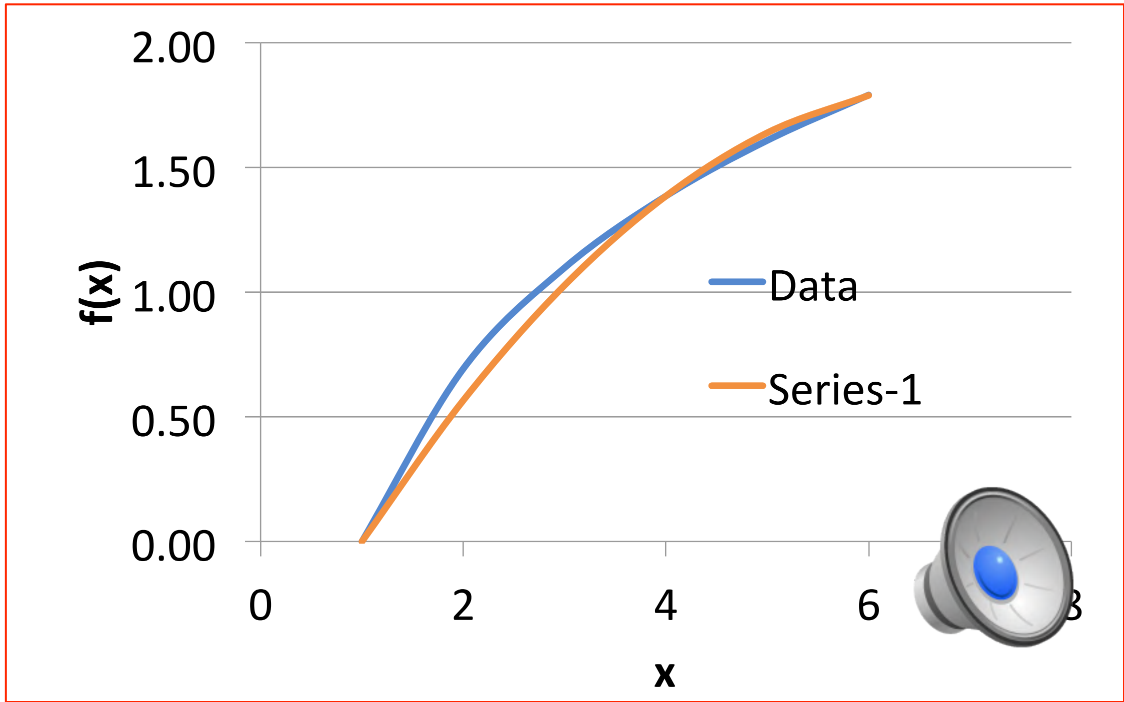
$$b_2 = \frac{\dfrac{f(x_2) - f(x_1)}{(x_2 - x_1)} - \dfrac{f(x_1) - f(x_0)}{(x_1 - x_0)}}{(x_2 - x_0)}$$

x_0	x_1	x_2
1	4	6
$f(x_0)$	$f(x_1)$	$f(x_2)$
0	1.39	1.79

b_0	0.00
b_1	0.46
b_2	-0.05

	Error(2)
Series1	18.32

X	y=ln(x)	Series1: f2(x)
1	0.00	0.00
2	0.69	0.57
3	1.10	1.03
4	1.39	1.39
5	1.61	1.64
6	1.79	1.79



Newton`s Divided-Difference Interpolating Polynomials

General Interpolating Polynomial

$$f_n(x) = b_0 + b_1(x - x_0) + \ldots + b_n(x - x_0)(x - x_1)\ldots.(x - x_{n-1})$$

$$b_0 = f(x_0)$$

$$b_1 = f[x_1, x_0]$$

$$b_2 = f[x_2, x_1, x_0]$$

$$\vdots$$

$$b_n = f[x_n, x_{n-1}, \ldots., x_1, x_0]$$

First finite divided difference

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

Second finite divided difference

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

nth finite divided difference

$$f[x_n, x_{n-1}, \ldots., x_1, x_0] = \frac{f[x_n, x_{n-1}, \ldots., x_1] - f[x_{n-1}, \ldots., x_1, x_0]}{(x_n - x_0)}$$

Newton`s Divided-Difference Interpolating Polynomials

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + b_3(x - x_0)(x - x_1)(x - x_2)$$

Cubic Interpolating Polynomial

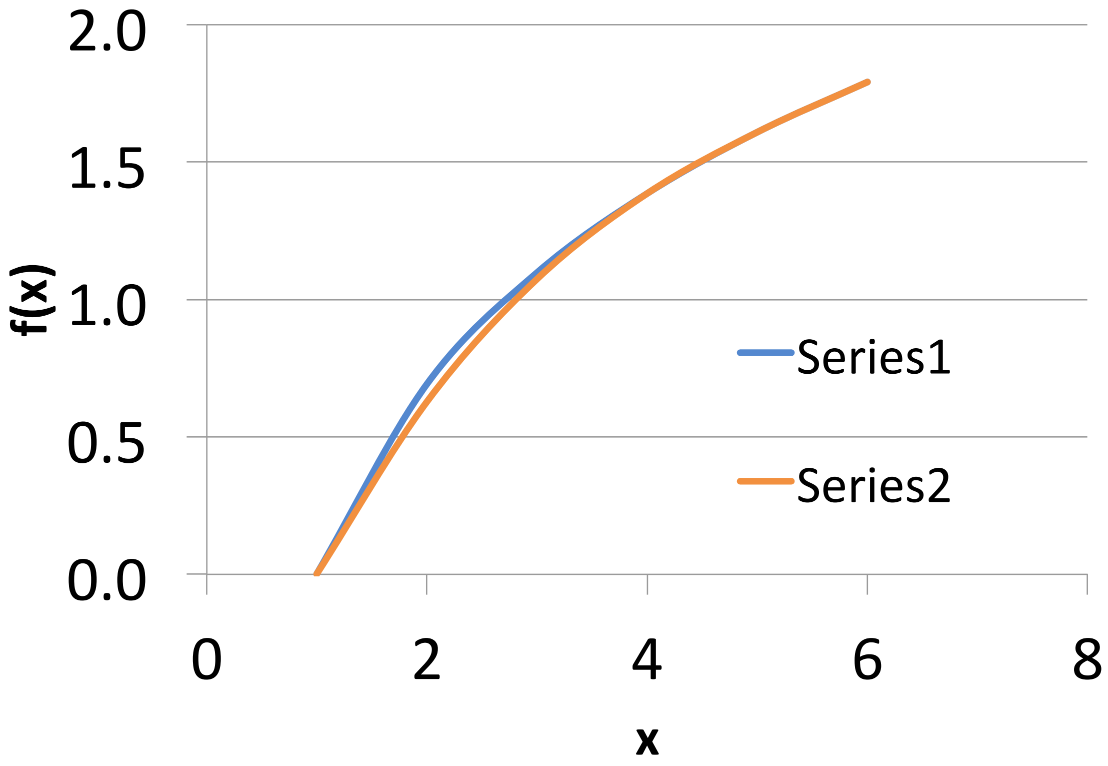
i	x_i	$f(x_i)$	First	Second	Third
0	x_0	$f(x_0)$	$f[x_1,x_0]$	$f[x_2,x_1,x_0]$	$f[x_3,x_2,x_1,x_0]$
1	x_1	$f(x_1)$	$f[x_2,x_1]$	$f[x_3,x_2,x_1]$	
2	x_2	$f(x_2)$	$f[x_3,x_2]$		
3	x_3	$f(x_3)$			

i	x_i	$f(x_i)$	First	Second	Third
0	1	0.00	0.46	-0.06	0.01
1	4	1.39	0.22	-0.02	
2	5	1.61	0.18		
3	6	1.79			

b_0	0.00
b_1	0.46
b_2	-0.06
b_3	0.01

X	y=ln(x)	Series1:$f_3(x)$
1	0.00	0.00
2	0.69	0.63
3	1.10	1.08
4	1.39	1.39
5	1.61	1.61
6	1.79	1.79

	Error(2)
Series1	9.29



NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Curve Fitting

Lagrange Polynomial and Newtons Divided Difference Formula

Prof. Jayati Sarkar

Newton`s Divided-Difference Error Analysis

$$\boxed{f_n(x) = b_0 + b_1(x - x_0) + \ldots + b_n(x - x_0)(x - x_1)\ldots.(x - x_{n-1})}$$

Also, as was the case with the Taylor series, a formulation for the truncation error can be obtained.

$$\begin{aligned}
R_n &= \frac{f^{(n+1)}(\xi)}{(n+1)!}(x_{i+1} - x_i)^{n+1} \\
&= \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)(x - x_1)\ldots.(x - x_n) \\
&= f[x, x_n, x_{n-1}, \ldots., x_1, x_0](x - x_0)(x - x_1)\ldots.(x - x_n) \\
&\cong f[x_{n+1}, x_n, x_{n-1}, \ldots., x_1, x_0](x - x_0)(x - x_1)\ldots.(x - x_n)
\end{aligned}$$

Newton`s Divided-Difference Error Analysis

$$\boxed{f_n(x) = b_0 + b_1(x - x_0) + \ldots + b_n(x - x_0)(x - x_1)\ldots.(x - x_{n-1})}$$

$$\boxed{R_n \cong f[x_{n+1}, x_n, x_{n-1}, \ldots., x_1, x_0](x - x_0)(x - x_1)\ldots.(x - x_n)}$$

$$f_{n+1}(x) = b_0 + b_1(x - x_0) + \ldots + b_n(x - x_0)(x - x_1)\ldots.(x - x_{n-1}) + b_{n+1}(x - x_0)(x - x_1)\ldots.(x - x_n)$$

$$f_{n+1}(x) = f_n(x) + b_{n+1}(x - x_0)(x - x_1)\ldots.(x - x_n)$$

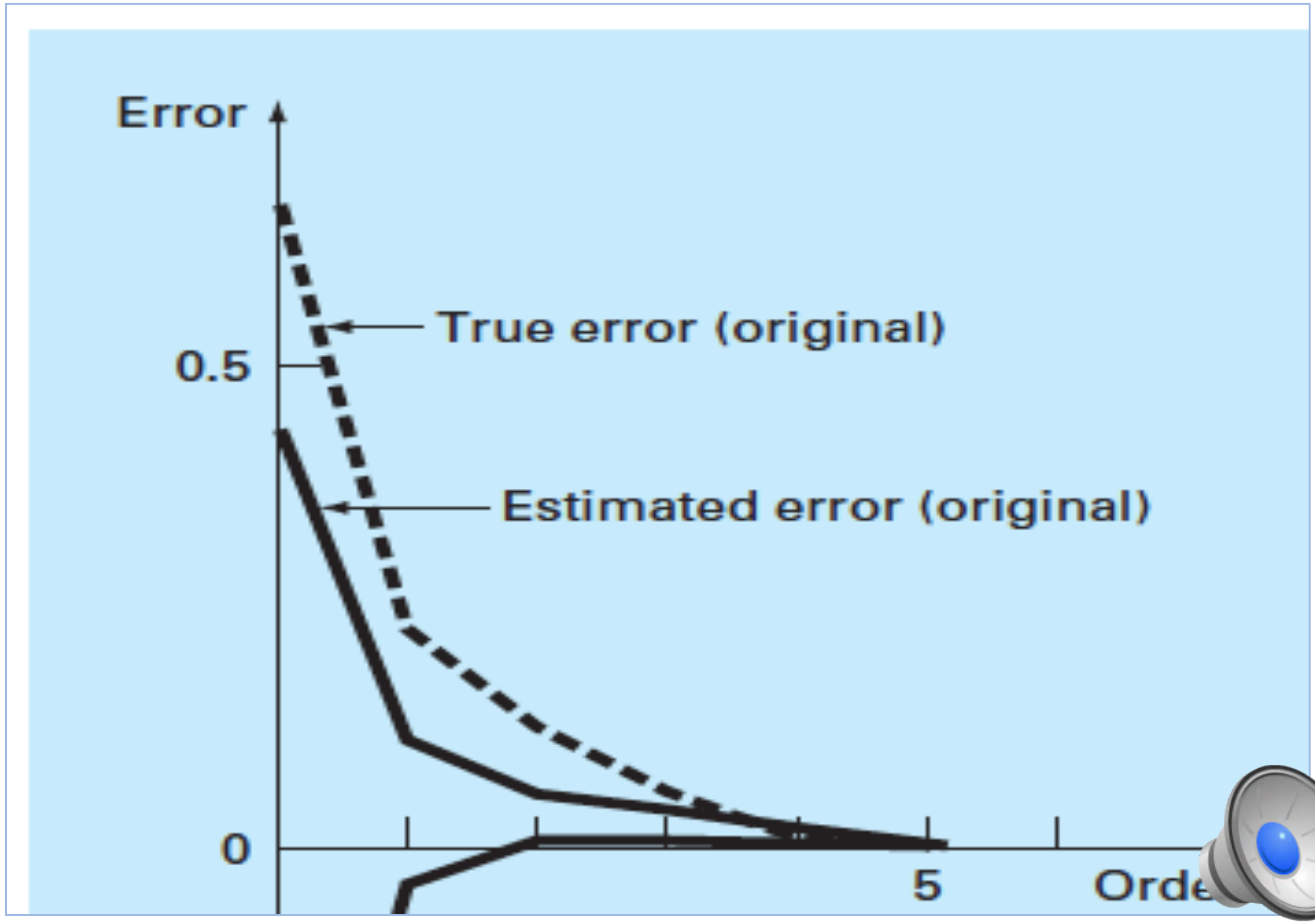
$$f_{n+1}(x) = f_n(x) + R_n$$

X	y=ln(x)	$f_2(x)$
1	0.00	0.00
2	0.69	0.57
3	1.10	1.03
4	1.39	1.39
5	1.61	1.64
6	1.79	1.79

$$E_a(2) = |0.69 - 0.57| = 0.12$$

$$R(2) = |0.63 - 0.57| = 0.06$$

X	y=ln(x)	$f_3(x)$
1	0.00	0.00
2	0.69	0.63
3	1.10	1.08
4	1.39	1.39
5	1.61	1.61
6	1.79	1.79



Lagrange Interpolating Polynomial

$$(x_1, y_1) \quad (x_2, y_2) \; . \; . \; \ldots\ldots \; (x_n, y_n)$$

$$P_1(x) = \frac{(x-x_2)(x-x_3)\ldots.(x-x_n)}{(x_1-x_2)(x_1-x_3)\ldots.(x_1-x_n)}$$

$$P_1(x_1) = 1 \qquad P_1(x_2) = 0 \qquad P_1(x_3) = 0 \;\ldots\ldots\ldots\ldots P_1(x_n) = 0$$

$$y_1 P_1(x) \xrightarrow{@\, x \,=\, x_1} = y_1$$

$$y_1 P_1(x) \xrightarrow[\text{@ } \forall\, x \text{ other than } x_1]{} = 0$$

$$P_2(x) = \frac{(x-x_1)(x-x_3)\ldots.(x-x_n)}{(x_2-x_1)(x_2-x_3)\ldots.(x_2-x_n)}$$

$$P_2(x_1) = 0 \qquad P_2(x_2) = 1 \qquad P_2(x_3) = 0 \;\ldots\ldots\ldots\ldots P_2(x_n) = 0$$

Generalizing:

$$P_i(x) = \frac{(x-x_1)(x-x_2)\ldots.(x-x_{i-1})(x-x_{i+1})\ldots..(x-x_n)}{(x_i-x_1)(x_i-x_2)\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..(x_i-x_n)}$$

$$\boxed{P_i(x) = \prod_{\substack{j=1 \\ i \neq j}}^{n} \frac{(x - x_j)}{(x_i - x_j)}}$$

$$P_i(x_j) = \delta_{ij} \begin{cases} = 1 & \text{for } i = j \\ = 0 & \text{for } i \neq j \end{cases}$$

$$f_{n-1}(x) = y_1P_1(x) + y_2P_2(x) \;.\;.\;.\;.\;.\;.\;. + y_n P_n(x)$$

if we substitute $x = x_1$

$$f(x_1) = y_1.\underbrace{1}_{P_1(x_1)=1} + y_2.\underbrace{0}_{P_1(x_2)=0} + \;.\;.\;.\;.\;.\; y_n \underbrace{P_n(x_1)}_{0} \quad = y_1$$

$$f(x_1) = y_1$$

$$@\, x = x_2$$

$$f(x_2) = y_1 \underset{\substack{\Downarrow \\ =0}}{P_1(x_2)} + y_2 \underset{\substack{\Downarrow \\ =1}}{P_2(x_2)} + y_3 \underset{\substack{\Downarrow \\ =0}}{P_3(x_2)} \; . \; . \; . \; .$$

$$= y_2$$

Lagrangian interpolating function is

$$f(x) = \sum_{i=1}^{n} y_i P_i(x)$$

n-1

$$P_i(x) = \prod_{\substack{j=1 \\ i \neq j}}^{n} \frac{(x - x_j)}{(x_i - x_j)}$$

Lagrange Interpolation

$$(1,0)(4,1.386294)$$

$$f_1(x)=y_1P_1(x)+y_2P_2(x) \qquad P_1(x)=\frac{(x-x_2)}{(x_1-x_2)},P_2(x)=\frac{(x-x_1)}{(x_2-x_1)}$$

$$f_1(2)=y_1P_1(2)+y_2P_2(2) \qquad P_1(2)=\frac{(2-4)}{(1-4)}=,P_2(2)=\frac{(2-1)}{(4-1)}$$

$$\boxed{f_1(2)=0\times 0.667+1.386295\times 0.333=0.462098}$$

Lagrange Interpolation

$$(1,0)(4,1.386294)(6,1.791760)$$

$$f_2(x) = y_1 P_1(x) + y_2 P_2(x) + y_3 P_3(x)$$

$$P_1(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)}, P_2(x) = \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)}, P_3(x) = \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)}$$

$$f_2(2) = y_1 P_1(2) + y_2 P_2(2) + y_3 P_3(2)$$

$$P_1(2) = \frac{(2-4)(2-6)}{(1-4)(1-6)}, P_2(2) = \frac{(2-1)(2-6)}{(4-1)(4-6)}, P_3(2) = \frac{(2-1)(2-4)}{(6-1)(6-4)}$$

$$\boxed{f_2(2) = 0 \times 0.533 + 1.386295 \times 0.666 + 1.79176 \times (-0.2) = 0.5}$$

Derivation of Lagrange Polynomial from Newton Divided Difference Formula

$$(x_1, y_1)(x_2, y_2)$$

$$f[x_1, x_2] = \frac{f(x_1) - f(x_2)}{(x_1 - x_2)} = \frac{f(x_1)}{(x_1 - x_2)} + \frac{f(x_2)}{(x_2 - x_1)}$$

$$f_1(x) = f(x_1) + f[x_1, x_2](x - x_1)$$

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

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

$$= y_1 \times P_1(x) + y_2 \times P_2(x)$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Curve Fitting

Numerical Differentiation

Prof. Jayati Sarkar

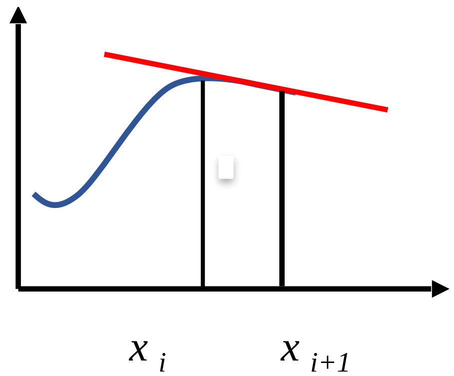
Differentiation

Given data (x_i, y_i) or a function

$y = f(x)$ ➞ Obtain $\frac{dy}{dx}$

Differentiation find slope of tangent of the curve at any point x

Diff: $\frac{dy}{dx} = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x} \approx \frac{y_{i+1} - y_i}{x_{i+1} - x_i}$

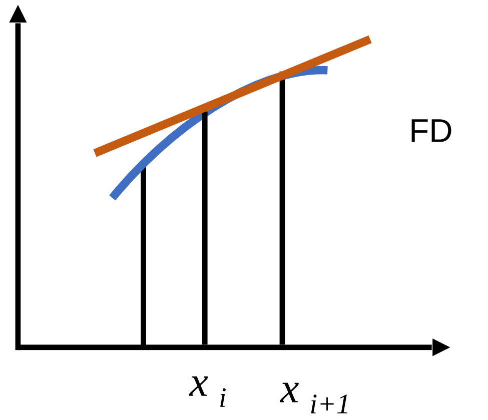


Application of Differentiation: $x^{i+1} = x^i - \frac{f(x^i)}{f'(x^i)}$

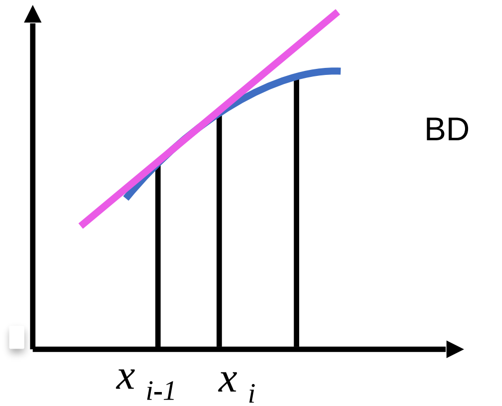
if derivative $f'(x)$ is not available analytically $\Rightarrow x^{i+1} = x^i - \frac{\delta f(x^i)}{f(x^i+\delta) - f(x^i)}$

Differentiation

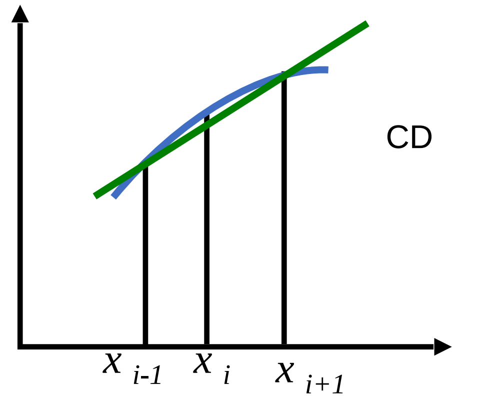
$$\frac{dy}{dx} = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} \rightarrow \text{Forward Difference}$$



$$\frac{dy}{dx} = \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \rightarrow \text{Backward Difference}$$



$$\frac{dy}{dx} = \frac{y_{i+1} - y_{i-1}}{x_{i+1} - x_{i-1}} \rightarrow \text{Central Difference}$$



Truncation Error

$$y_{i+1} = f(x_{i+1}) \approx f(x_i) + \Delta x f'(x_i) + \frac{\Delta x^2}{2!} f''(x_i) + \dots\dots\dots$$

$$y_{i+1} = y_i + \Delta x y_i' + \frac{\Delta x^2}{2!} y_i'' + \frac{\Delta x^3}{3!} f'''(\xi) \rightarrow (1)$$

$$y_{i-1} = y_i - \Delta x y_i' + \frac{\Delta x^2}{2!} y_i'' - \frac{\Delta x^3}{3!} f'''(\xi) \rightarrow (2)$$

From (1) $\rightarrow y_i' = \frac{y_{i+1} - y_i}{\Delta x} - \boxed{\frac{\Delta x}{2!} f_i''(\xi)}$ $\rightarrow$ Truncation error

$\theta(\Delta x)$ 1^{st} order method (Forward difference)

From (2) $\rightarrow y_i' = \frac{y_i - y_{i-1}}{\Delta x} + \boxed{\frac{\Delta x}{2!} f_i''(\xi)}$ $\rightarrow$ Truncation error

$\theta(\Delta x)$ 1^{st} order method (Backward Difference)

Truncation Error

$$y_{i+1} = y_i + \Delta x y_i' + \frac{\Delta x^2}{2!} y_i'' + \frac{\Delta x^3}{3!} f'''(\xi) \rightarrow (1)$$

$$y_{i-1} = y_i - \Delta x y_i' + \frac{\Delta x^2}{2!} y_i'' - \frac{\Delta x^3}{3!} f'''(\xi) \rightarrow (2)$$

Subtract (2) from (1)

$$y_{i+1} - y_{i-1} = 2\Delta x y_i' + \frac{2\Delta x^3}{3!} f'''(\xi)$$

$$y_i' = \underbrace{\frac{y_{i+1} - y_{i-1}}{2\Delta x}}_{\text{CDS}} - \underbrace{\frac{\Delta x^2}{3!} f'''(\xi)}_{\text{2nd order error (Truncation error)}}$$

Truncation Error

$$y_{i+1} = y_i + \Delta x y_i' + \frac{\Delta x^2}{2!} y_i'' + \frac{\Delta x^3}{3!} f'''(\xi) \rightarrow (1)$$

$$y_{i-1} = y_i - \Delta x y_i' + \frac{\Delta x^2}{2!} y_i'' - \frac{\Delta x^3}{3!} f'''(\xi) \rightarrow (2)$$

By adding (1) & (2) we get

$$y_{i+1} - 2y_i + y_{i-1} = \frac{2\Delta x^2}{2!} y_i'' + \frac{2\Delta x^4}{4!} f''''(\xi)$$

$$y_i'' = \underbrace{\frac{y_{i+1} - 2y_i + y_{i-1}}{\Delta x^2}}_{\text{CDS}} - \underbrace{\frac{\Delta x^2}{12} f''''(\xi)}_{\text{Error}}$$

Method of undetermined coefficients

$$f'(x_i) = a_1 f(x_{i+1}) + a_2 f(x_i) + a_3 f(x_{i-1})$$

undetermined coefficients: a_1, a_2, a_3

3 point difference formula

$$f'(x_i) = a_1\left[f(x_i) + \Delta x f'(x_i) + \frac{\Delta x^2}{2!} f''(x_i) + \frac{\Delta x^3}{3!} f'''(x_i)\right] + [a_2 f(x_i)]$$
$$+ a_3\left[f(x_i) - \Delta x f'(x_i) + \frac{\Delta x^2}{2!} f''(x_i) - \frac{\Delta x^3}{3!} f'''(x_i)\right]$$

$$\Rightarrow f'(x_i) = f(x_i)[a_1 + a_2 + a_3] + f'(x_i)(a_1 - a_3)\Delta x + \frac{f''(x_i)\Delta x^2}{2!}(a_1 + a_3) + \ldots\ldots\ldots$$

$$\Rightarrow f'(x_i) = f(x_i)[a_1 + a_2 + a_3] + f'(x_i)(a_1 - a_3)\Delta x + \frac{f''(x_i)\Delta x^2}{2!}(a_1 + a_3) + \ldots\ldots\ldots$$

$$\Rightarrow a_1 + a_2 + a_3 = 0$$

$$\Delta x(a_1 - a_3) = 1$$

$$a_1 + a_3 = 0 \rightarrow a_1 = -a_3$$

$$\boxed{a_2 = 0}$$

$$-2a_3 = \frac{1}{\Delta x} \Rightarrow \boxed{a_3 = -\frac{1}{2\Delta x}}$$

$$\Rightarrow \boxed{a_1 = \frac{1}{2\Delta x}}$$

$$f'(x_i) = a_1 f(x_{i+1}) + a_2 f(x_i) + a_3 f(x_{i-1})$$

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

Truncation Error

$$f'(x_i) = f(x_i)[a_1 + a_2 + a_3] + \Delta x f'(x_i)[\underbrace{a_1 - a_3}_{\frac{1}{\Delta x}}] + \frac{(\Delta x)^2}{2!} f''(x_i)[\underbrace{a_1 + a_3}_{\mathbf{1}}] + \boxed{\frac{(\Delta x)^3}{3!} f'''(x_i)[a_1 - a_3]}$$

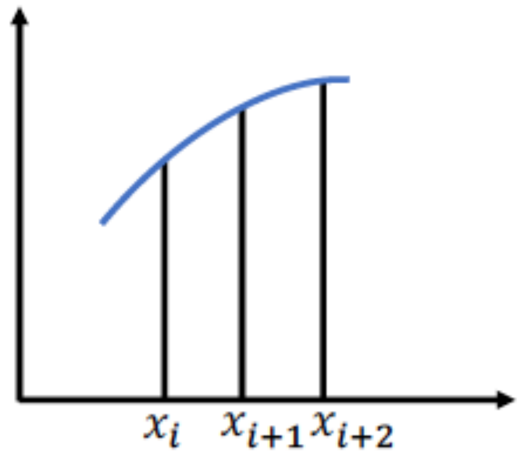
Error

$$= \frac{(\Delta x)^3}{3!} f'''(x_i)[a_1 - a_3]$$

$$\approx \frac{(\Delta x)^2}{3!} f'''(x_i)$$

3 point forward difference

$$f'(x_i) = a_1 f(x_i) + a_2 f(x_{i+1}) + a_3 f(x_{i+2})$$



$$= a_1 f_i + a_2\left[f_i + \Delta x f_i' + \frac{\Delta x^2}{2!} f_i'' + \frac{\Delta x^3}{3!} f_i''' + \cdots\right] + a_3\left[f_i + 2\Delta x f_i' + \frac{4\Delta x^2}{2!} f_i'' + \frac{8\Delta x^3}{3!} f_i''' + \cdots\right]$$

$$f'(x_i) = [a_1 + a_2 + a_3] f_i + \Delta x f_i'[a_2 + 2a_3] + \frac{\Delta x^2}{2!} f_i''[a_2 + 4a_3] + \boxed{\Delta x^3 (a_2 + 8a_3)\frac{f_i'''}{3!}}$$

$$\boxed{\begin{aligned} &a_1 + a_2 + a_3 = 0 \;\Rightarrow a_1 = -a_2 - a_3 \\ &a_2 + 2a_3 = \frac{1}{\Delta x} \\ &a_2 + 4a_3 = 0 \end{aligned}}$$

Error

3 point forward difference

$$\boxed{\begin{aligned} &a_1 + a_2 + a_3 = 0 \;\Rightarrow a_1 = -a_2 - a_3 \\ &a_2 + 2a_3 = \frac{1}{\Delta x} \\ &a_2 + 4a_3 = 0 \end{aligned}}$$

$$a_3 = \frac{-1}{2\Delta x} \qquad a_2 = \frac{2}{\Delta x} \qquad a_1 = -\frac{3}{2\Delta x}$$

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

Error

$$\Delta x^3 (a_2 + 8a_3) \frac{f_i'''}{3!}$$

$$= \frac{-\Delta x^2}{3} f_i'''$$

Comparisons

Forward Difference:

$$f'(x_i) = \frac{f_{i+1} - f_i}{\Delta x} - \frac{\Delta x}{2!} f_i''(\xi)$$

Central Difference:

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

3 Pt FWD Difference:

$$f'(x_i) = \frac{-3f_i + 4f_{i+1} - f_{i+2}}{2\Delta x} - \frac{(\Delta x)^2}{3} f_i'''(\xi)$$

$$f(x) = -0.1x^4 - 0.15x^3 - 0.5x^2 - 0.25x + 1.2$$

$$\boxed{f'(x) = -0.4x^3 - 0.45x^2 - x - 0.25 = -0.9125 @ x = 0.5}$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Numerical Differentiation

Prof. Jayati Sarkar

Comparisons

Forward Difference:

$$f'(x_i)=\frac{f_{i+1}-f_i}{\Delta x}-\frac{\Delta x}{2!}f_i''(\xi)$$

Central Difference:

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

3 Pt FWD Difference:

$$f'(x_i)=\frac{-3f_i+4f_{i+1}-f_{i+2}}{2\Delta x}-\frac{(\Delta x)^2}{3}f_i'''(\xi)$$

$$f(x)=-0.1x^4-0.15x^3-0.5x^2-0.25x+1.2$$

$$\boxed{f'(x)=-0.4x^3-0.45x^2-x-0.25=-0.9125@x=0.5}$$

delx	x(i-1)	x(i)	x(i+1)	x(i+2)	f(i-1)	f(i)	f(i+1)	f(i+2)
1	-5.000000000000E-01	5.00E-01	1.500000000000E+00	2.500000000000E+00	1.212500000000E+00	9.25E-01	-1.312500000000E+00	-8.800000000000E+00
0.1	4.000000000000E-01	5.00E-01	6.000000000000E-01	7.000000000000E-01	1.007840000000E+00	9.25E-01	8.246400000000E-01	7.045400000000E-01
0.01	4.900000000000E-01	5.00E-01	5.100000000000E-01	5.200000000000E-01	9.340378490000E-01	9.25E-01	9.157871490000E-01	9.063971840000E-01
0.001	4.990000000000E-01	5.00E-01	5.010000000000E-01	5.020000000000E-01	9.259116253499E-01	9.25E-01	9.240866246499E-01	9.231714971984E-01
0.0001	4.999000000000E-01	5.00E-01	5.001000000000E-01	5.002000000000E-01	9.250912412504E-01	9.25E-01	9.249087412497E-01	9.248174649972E-01
0.0000	4.999900000000E-01	5.00E-01	5.000100000000E-01	5.000200000000E-01	9.250091249125E-01	9.25E-01	9.249908749125E-01	9.249817496500E-01
0.000001	4.999990000000E-01	5.00E-01	5.000010000000E-01	5.000020000000E-01	9.250009124991E-01	9.25E-01	9.249990874991E-01	9.249981749965E-01
0.0000001	4.999999000000E-01	5.00E-01	5.000001000000E-01	5.000002000000E-01	9.250000912500E-01	9.25E-01	9.249999087500E-01	9.249998175000E-01
0.00000001	4.999999900000E-01	5.00E-01	5.000000100000E-01	5.000000200000E-01	9.250000091250E-01	9.25E-01	9.249999908750E-01	9.249999817500E-01
0.000000001	4.999999990000E-01	5.00						
1E-10	4.999999999000E-01	5.00						
1E-11	4.999999999900E-01	5.00						
1E-12	4.999999999990E-01	5.00						
1E-13	4.999999999999E-01	5.00						
1E-14	5.000000000000E-01	5.00						
1E-15	5.000000000000E-01	5.00						

$$f(x) = -0.1x^4 - 0.15x^3 - 0.5x^2 - 0.25x + 1.2$$

$$f'(x) = -0.4x^3 - 0.45x^2 - x - 0.25$$
$$= -0.9125 @ x = 0.5$$

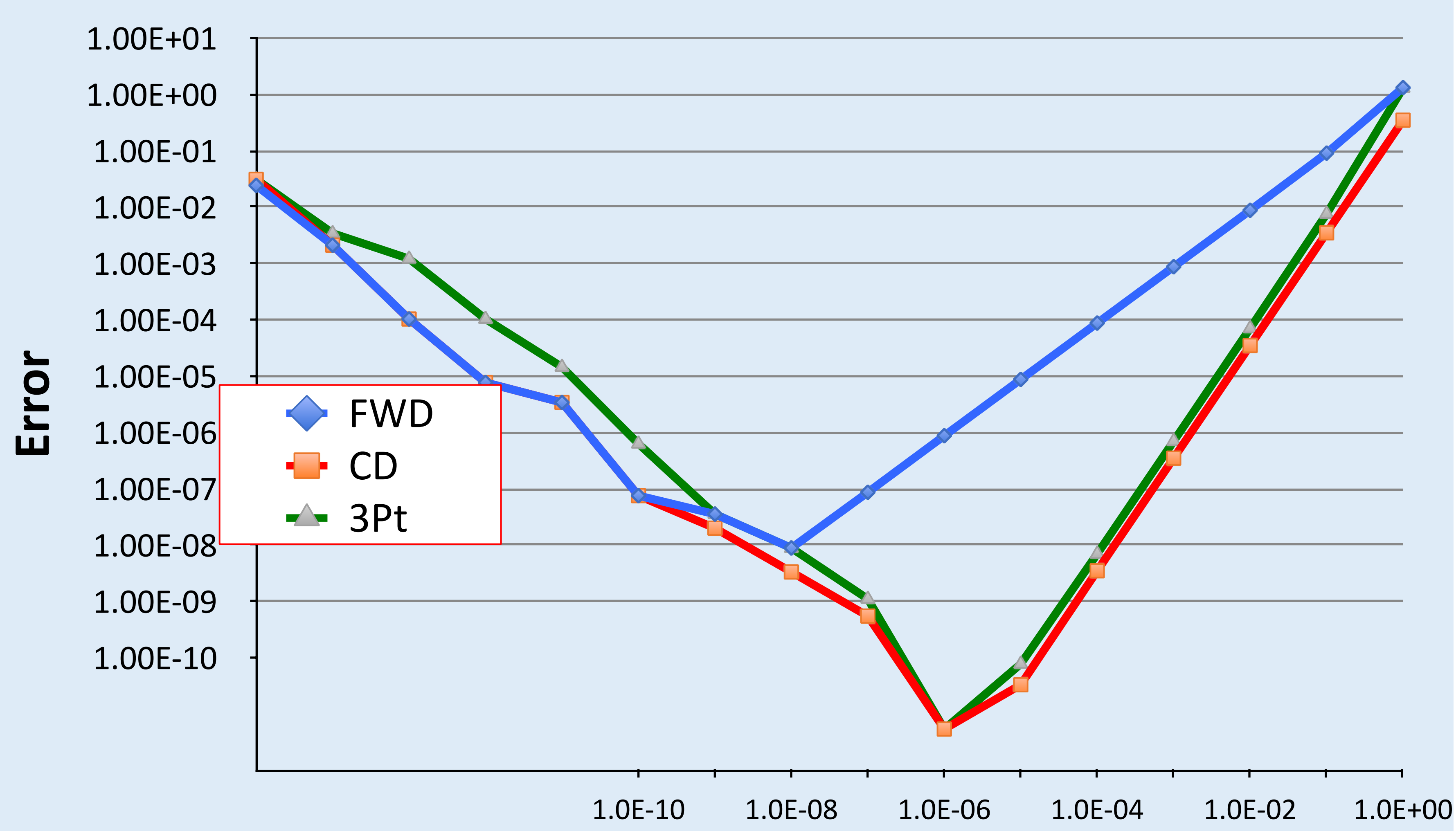
Forward Difference	Central Difference	3PtFWD Difference	Error		
f`(i)	f`(i)	f`(i)	E_FD	E_CD	E_3Pt
-2.237500000000E+00	-1.262500000000E+00	3.875000000000E-01	1.325000000000	0.350000000000	1.300000000000
-1.003600000000E+00	-9.160000000000E-01	-9.049000000000E-01	0.091100000000	0.003500000000	0.007600000000
-9.212851000000E-01	-9.125350000000E-01	-9.124294000000E-01	0.008785100000	0.000035000000	0.000070600000
-9.133753500999E-01	-9.125003500000E-01	-9.124992994000E-01	0.000875350100	0.000000350000	0.000000700600
-9.125875034999E-01	-9.125000034998E-01	-9.124999929999E-01	0.000087503500	0.000000003500	0.000000007000
-9.125087500284E-01	-9.125000000332E-01	-9.124999999222E-01	0.000008750028	0.000000000033	0.000000000078
-9.125008749722E-01	-9.125000000054E-01	-9.125000000054E-01	0.000000874972	0.000000000005	0.000000000005
-9.125000866028E-01	-9.124999994503E-01	-9.124999988952E-01	0.000000086603	0.000000000550	0.000000001105
-9.125000088872E-01	-9.125000033361E-01	-9.125000088872E-01	0.000000008887	0.000000003336	0.000000008887
-9.124999644783E-01	-9.125000199894E-01	-9.124999644783E-01	0.000000035522	0.000000019989	0.000000035522
-9.125000755006E-01	-9.125000755006E-01	-9.125006306121E-01	0.000000075501	0.000000075501	0.000000630612
-9.125034061697E-01	-9.125034061697E-01	-9.12514508399E-01	0.000003406170	0.000003406170	0.000014508400
-9.124923039394E-01	-9.124923039394E-01	-9.126033262419E-01	0.000007696061	0.000007696061	0.000103326242
-9.126033262419E-01	-9.126033262419E-01	-9.137135492665E-01	0.000103326242	0.000103326242	0.001213549267
-9.103828801926E-01	-9.103828801926E-01	-9.159339953158E-01	0.002117119807	0.002117119807	0.003433995316
-8.881784197001E-01	-9.436895709314E-01	-9.436895709314E-01	0.024321580300	0.031189570931	0.031189570931

Error Comparisons

Forward Difference: $$f'(x_i) = \frac{f_{i+1} - f_i}{\Delta x} - \frac{\Delta x}{2!} f_i''(\xi)$$

Central Difference: $$f'(x_i) = \frac{f_{i+1} - f_{i-1}}{2\Delta x} - \frac{(\Delta x)^2}{3!} f_i'''(\xi)$$

3 Pt FWD Diff: $$f'(x_i) = \frac{-3f_i + 4f_{i+1} - f_{i+2}}{2\Delta x} - \frac{(\Delta x)^2}{3} f_i'''(\xi)$$



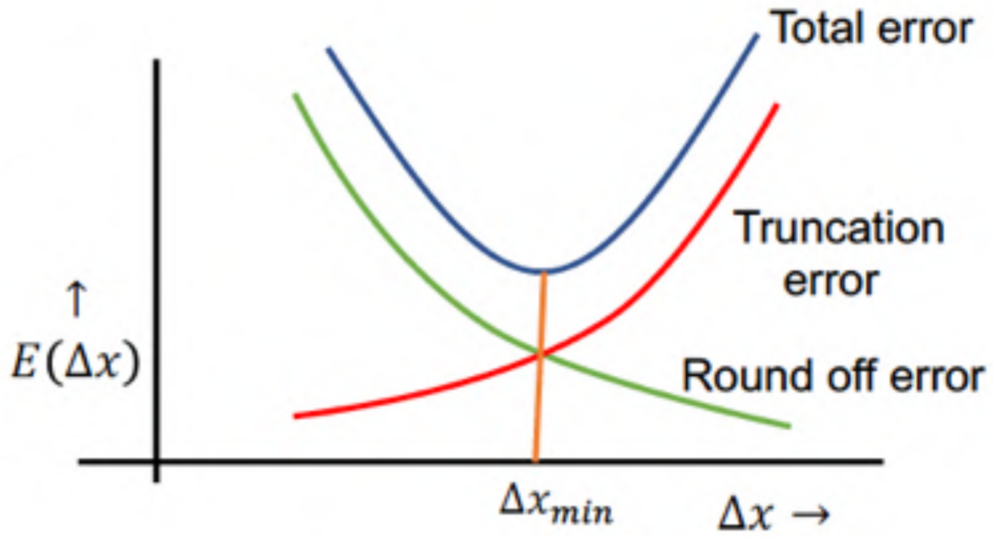
Error analysis

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

Truncation error: $e_{trunc} = \left|\frac{f''(\xi)}{4}\Delta x\right|$

Round off error: $e_{round\ off} = \frac{e_{i+1} - e_i}{\Delta x} = \left|\frac{2\varepsilon_{precision}}{\Delta x}\right|$

Total error: $E(\Delta x) = \frac{f''(\xi)}{4}\Delta x + \frac{2\varepsilon_{precision}}{\Delta x}$



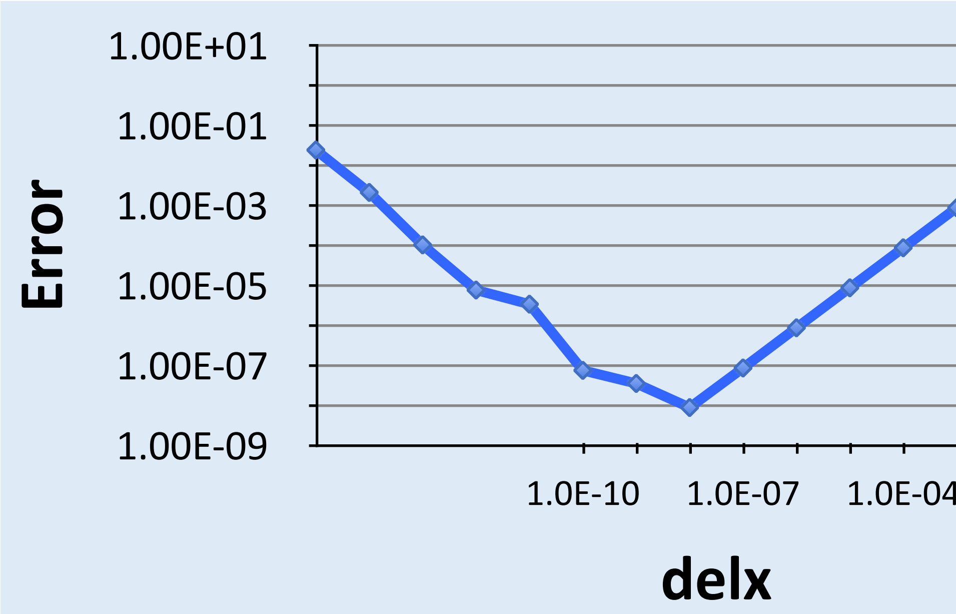
Error analysis

Forward Difference:

Total error: $E(\Delta x) = \frac{f''(\xi)}{4}\Delta x + \frac{2\varepsilon_{precision}}{\Delta x}$

$$E'(\Delta x) = \frac{f''(\xi)}{4} - \frac{2\varepsilon_{precision}}{\Delta x^2{}_{min}} = 0$$

$$\boxed{\begin{aligned} \Delta x^2{}_{min} &\propto \varepsilon_{pre} \\ \Delta x_{min} &\propto (\varepsilon_{pre})^{\frac{1}{2}} \end{aligned}}$$



For suppose $\varepsilon_{pre} = 2x10^{-16}$ (double precision for Excel)

$$\Delta x_{min} \propto 10^{-8}$$

Central difference:

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

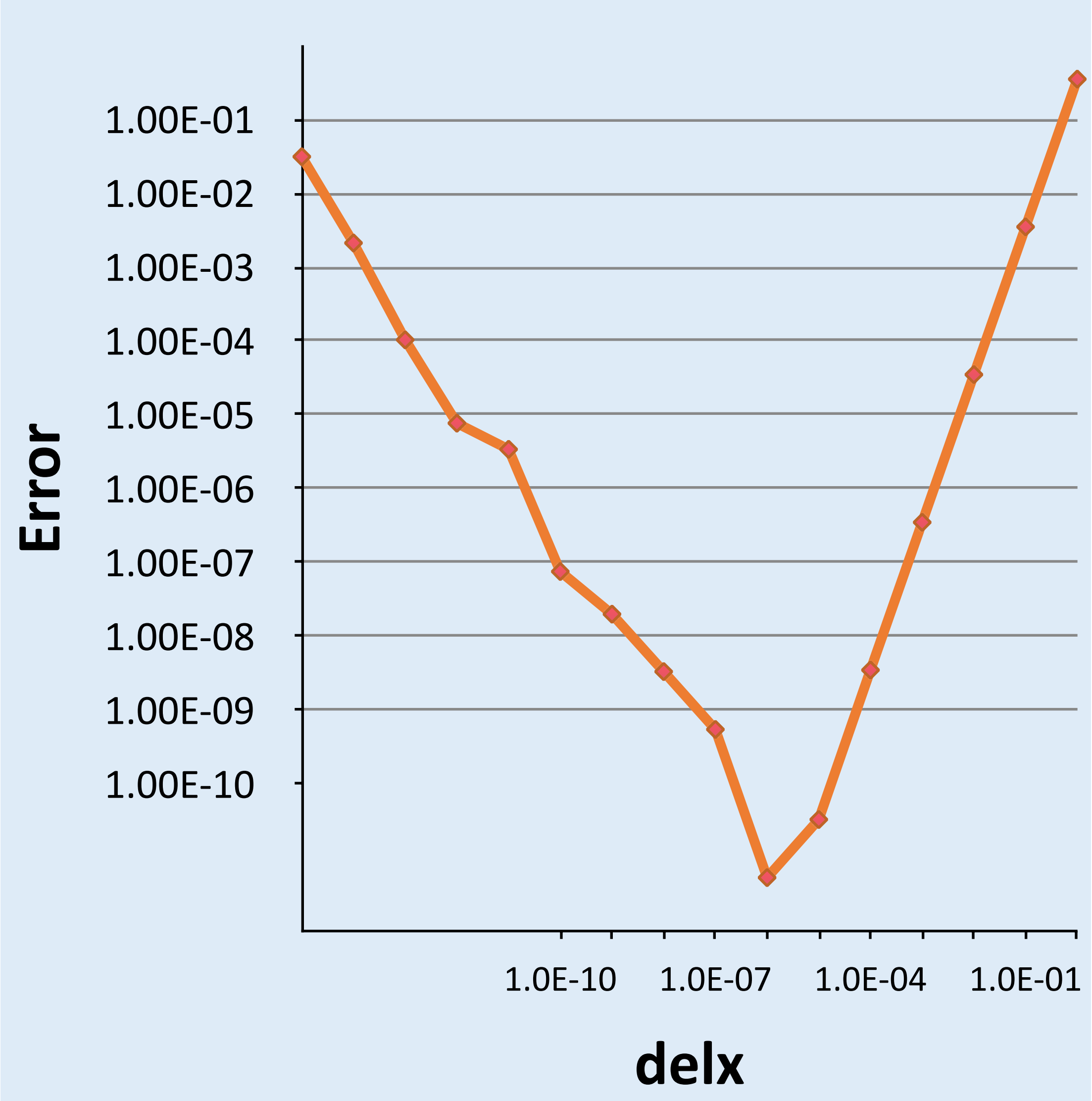
Truncation error = $\left|\frac{\Delta x^2}{6} f'''(x_i)\right|$

Round off error = $\frac{\varepsilon_p}{\Delta x}$

Total error(E) = $\frac{f'''(\xi)}{6}\Delta x^2 + \frac{\varepsilon_p}{\Delta x}$

$$\frac{dE(\Delta x)}{d\Delta x} = -\frac{\varepsilon_p}{\Delta x^2} + \frac{2}{6} f'''(\xi)\Delta x = 0$$

$$\Delta x \propto \varepsilon_p^{\frac{1}{3}} \sim 10^{-5}$$



Central difference:

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

$$E = \frac{4\varepsilon_p}{\Delta x^2} + \frac{\Delta x^2}{12} f^{IV}(\xi) = 0$$

$$\frac{dE}{d\Delta x} = -\frac{8\varepsilon_p}{\Delta x^3} + \frac{2}{12} f^{IV}(\xi)\Delta x = 0$$

$$\Delta x \sim \varepsilon_p^{\frac{1}{4}}$$

Summary

- As Δx decreases truncation error decreases round off error increases. There exists a Δx_{opt} for which total error is *minimum*

$$\boxed{\begin{aligned} f'_{FWD} &\rightarrow \Delta x_{opt} \propto \varepsilon^{\frac{1}{2}} \\ f'_{CD\ 3pt} &\rightarrow \Delta x_{opt} \propto \varepsilon^{\frac{1}{3}} \\ f''_{CD} &\rightarrow \Delta x_{opt} \propto \varepsilon^{\frac{1}{4}} \end{aligned}}$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Numerical Integration

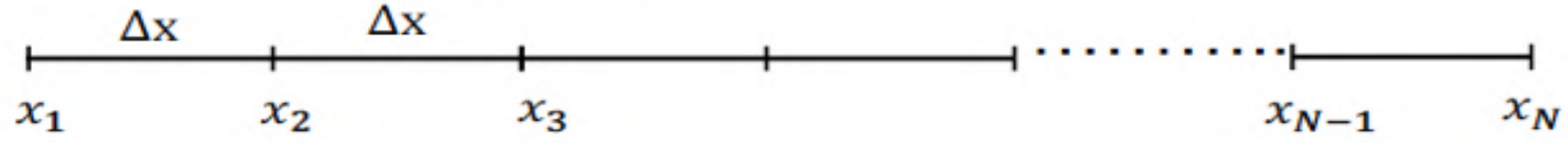
Newton Cotes Formula

Prof. Jayati Sarkar

Newton's Forward Difference Method
(Curve Fitting)

DATA: (x_1, y_1) (x_2, y_2) . . .. (x_N, y_N)

Equidistant:



$$\boxed{\alpha = \frac{x - x_1}{\Delta x}}$$

x_1 $\alpha = 0$ x_2 $\alpha = 1$ x_3 $\alpha = 2$

x_N $\alpha =$ N-1

$$\boxed{\Rightarrow \alpha = i - 1 \; for \; x = x_i}$$

DATA: $(0, y_1)$ $(1, y_2)$ ....$((N-1), y_N)$

Newton's Forward Difference Method (Curve Fitting)

$$\boxed{\text{DATA: } (0, y_1)\ (1, y_2)\ \ldots\ldots\ldots\ldots\ldots ((N\text{-}1), y_N)}$$

$$P(\alpha) = b_o + b_1\alpha + b_2\alpha(\alpha-1)\ldots\ldots\ldots + b_{N-1}\alpha(\alpha-1)\ldots\ldots(\alpha-(N-2))$$

1. $\alpha = 0 \qquad y = y_1 \qquad \boxed{y_1 = b_o}$

2. $\alpha = 1 \qquad y = y_2 \qquad y_2 = b_o + b_1 \qquad b_1 = y_2 - y_1 = \Delta y_1 \qquad \boxed{b_1 = \Delta y_1}$

3. $\alpha = 2$

$$y_3 = b_o + 2b_1 + 2b_2$$

$$b_2 = \frac{1}{2!}[Y_3 - Y_1 - 2\Delta Y_1] = \frac{1}{2!}[\underbrace{Y_3 - Y_2}_{\Delta y_2} + \underbrace{Y_2 - Y_1}_{\Delta y_1} - 2\Delta Y_1]$$

$$= \frac{1}{2!}[\Delta Y_2 - \Delta Y_1] = \frac{\Delta^2 Y_1}{2!}$$

$$\{\because \Delta^2 = \Delta(\Delta Y_1)\}$$

$$\boxed{b_2 = \frac{\Delta^2 Y_1}{2!}}$$

Newton's Forward Difference Method (Curve Fitting)

DATA: $(0, y_1)\ (1, y_2)\ \ldots\ldots\ldots\ldots\ldots((N\text{-}1), y_N)$

$$P(\alpha) = b_o + b_1\alpha + b_2\alpha(\alpha-1)\ldots\ldots\ldots + b_{N-1}\alpha(\alpha-1)\ldots\ldots(\alpha-(N-2))$$

4. $\alpha = 3$

$$y_4 = b_0 + 3b_1 + 3.2b_2 + 3.2.1b_3$$

$$b_3 = \frac{1}{6}(y_4 - b_0 - 3b_1 - 6b_2)$$

$$= \frac{1}{6}\left(y_4 - y_1 - 3(\Delta y_1) - 6\times\frac{1}{2!}(\Delta^2 y_1)\right)$$

$$= \frac{1}{3!}\left(y_4 - y_1 - 3(y_2 - y_1) - 6\times\frac{1}{2!}(y_3 - 2y_2 + y_1)\right)$$

$$= \frac{1}{3!}(y_4 - 3y_3 + 3y_2 - y_1)$$

$$\boxed{b_3 = \frac{\Delta^3 y_1}{3!}}$$

$$\Delta^3 y_1 = \Delta(\Delta^2 y_1)$$
$$= \Delta(\Delta(\Delta y_1))$$
$$= \Delta(\Delta(y_2 - y_1))$$
$$= \Delta(\Delta y_2 - \Delta y_1)$$
$$= \Delta((y_3 - y_2) - (y_2 - y_1))$$
$$= \Delta y_3 - 2\Delta y_2 + \Delta y_1$$
$$= (y_4 - y_3) - 2(y_3 - y_2) + (y_2 - y_1)$$
$$= y_4 - 3y_3 + 3y_2 - y_1$$

Newton's Forward Difference Method (Curve Fitting)

DATA: $(0, y_1)\ (1, y_2)\ \ldots\ldots\ldots\ldots\ldots((N\text{-}1), y_N)$

$$P(\alpha) = b_o + b_1\alpha + b_2\alpha(\alpha-1)\ldots\ldots\ldots + b_{N-1}\alpha(\alpha-1)\ldots\ldots(\alpha-(N-2))$$

$$\boxed{b_1 = y_1} \qquad \boxed{b_2 = \frac{\Delta^2 y_1}{2!}} \qquad \boxed{b_3 = \frac{\Delta^3 y_1}{3!}} \qquad \boxed{b_i = \frac{\Delta^i y_1}{i!}}$$

$$\boxed{P(\alpha) = y_1 + \Delta y_1\alpha + \frac{\Delta^2 y_1}{2!}\alpha(\alpha-1) + \ldots.. + \frac{\Delta^{N-1} y_1}{(N-1)!}\alpha(\alpha-1)..(\alpha-(N-2))}$$

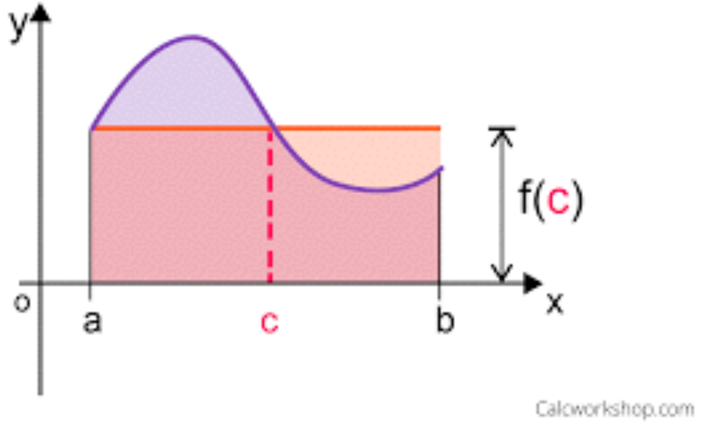
cross check $\quad @\ \alpha = 0 \qquad Y = Y_1$

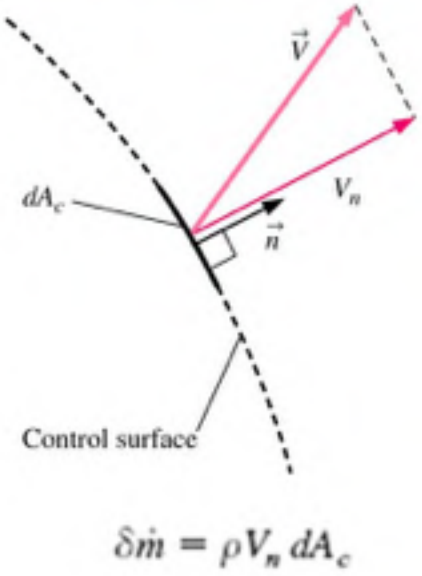
$@\ \alpha = 1 \qquad Y = Y_1 + (Y_2 - Y_1).1 = Y_2$

Integration-Applications

- To calculate the mean:

$$= \frac{1}{b-a}\int_a^b f(x)dx$$



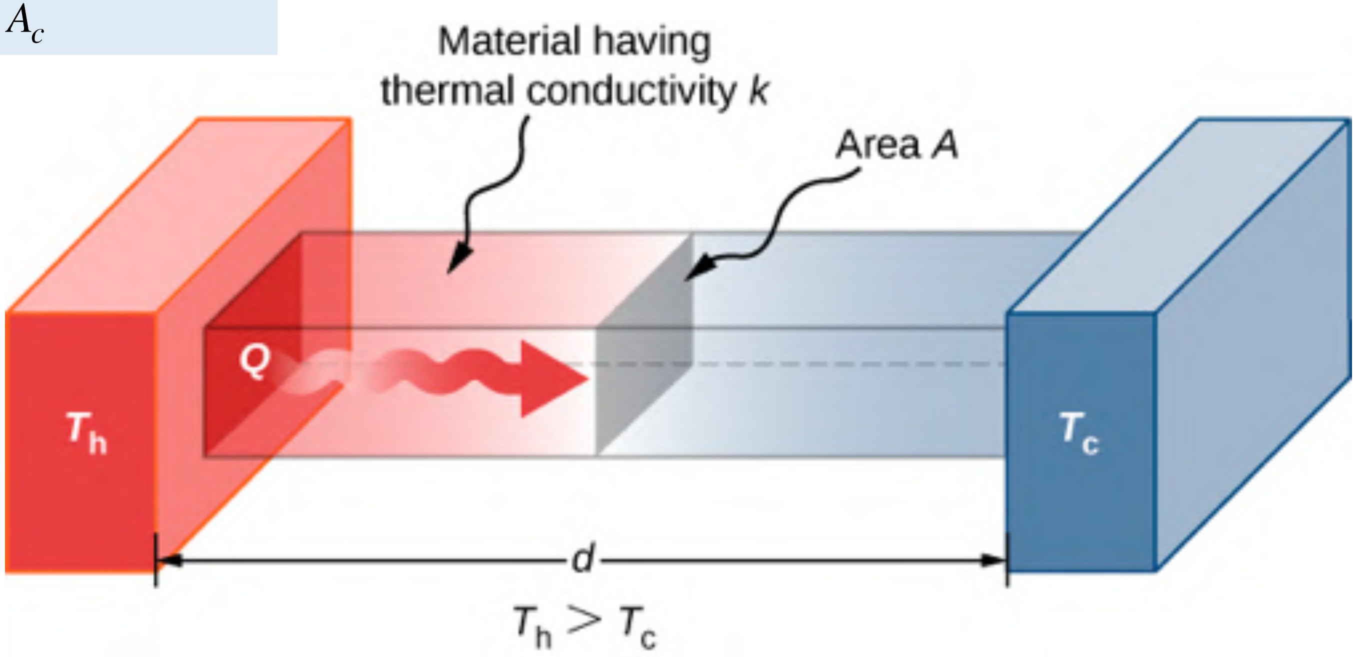


- Mass flow:

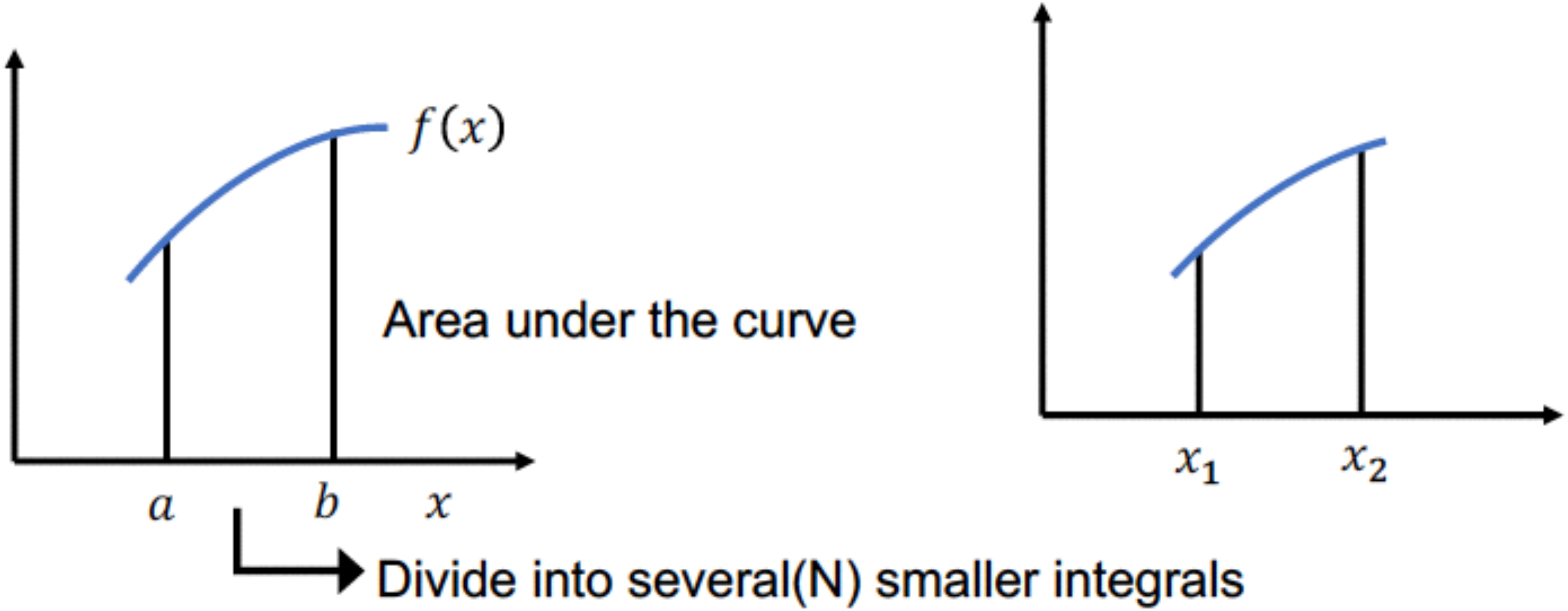
$$= \int_{A_c} \rho \vec{V} \bullet d\vec{A}_c = \int_{A_c} \rho V_n dA_c$$

- Net heat loss:

$$= \int_S \overrightarrow{flux} \bullet d\vec{S}$$



Numerical Integration



Geometric Interpretation of Integration

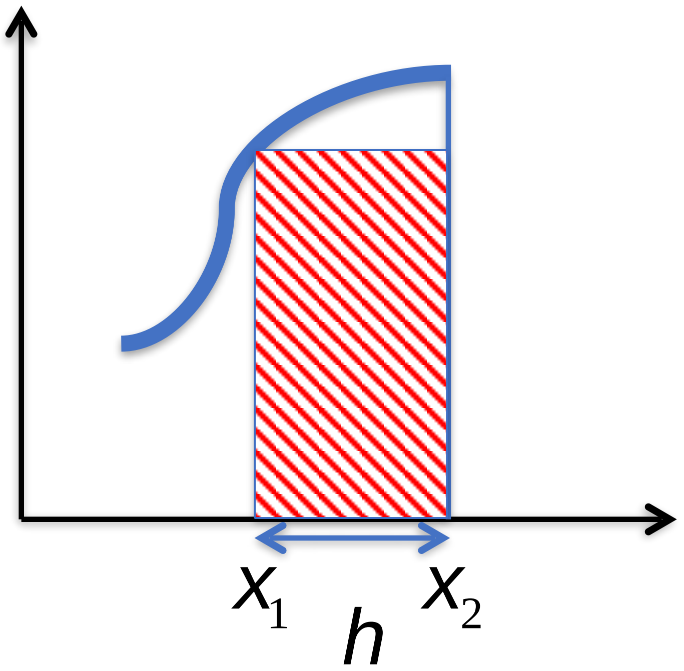
$$I = \int_a^b f(x)dx \;\cong\; \int_a^b f_n(x)dx$$

$$f_n(x) = a_o + a_1x + a_2x^2 + \dots\dots\dots + a_nx^n$$

- Newton Cotes Integration Formula:

Euler Backward: $\boxed{I = f(x_1)(x_2 - x_1)} + \theta(h)$

truncation error

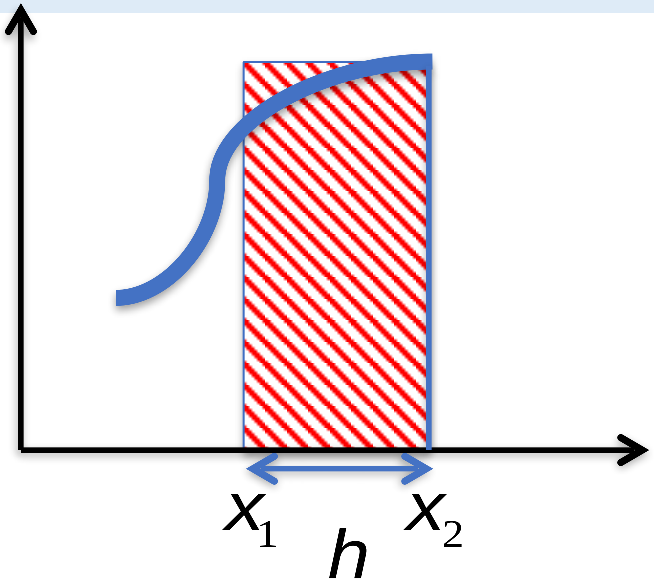


Geometric Interpretation of Integration

- Newton Cotes Integration Formula:

Euler Forward: $\boxed{I = f(x_2)(x_2 - x_1)}$ $+ \theta(h)$

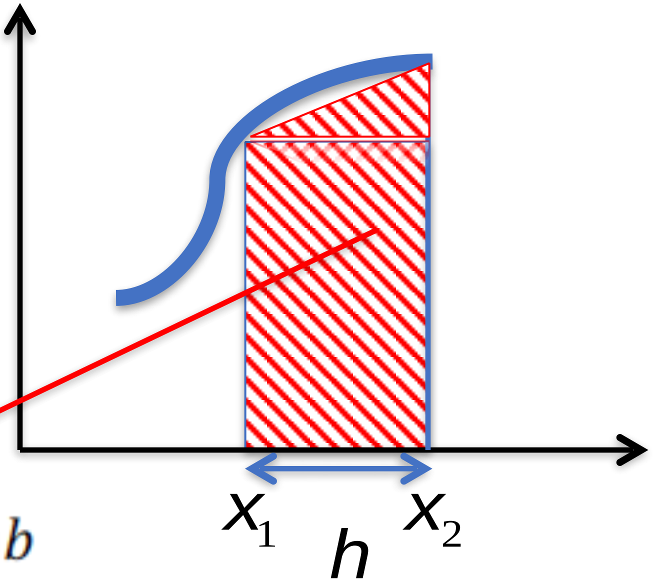
truncation error



Trapezoidal Rule:

$$\boxed{I = \frac{1}{2}\left(f(x_1) + f(x_2)\right)(x_2 - x_1)} + \theta(h^2)$$

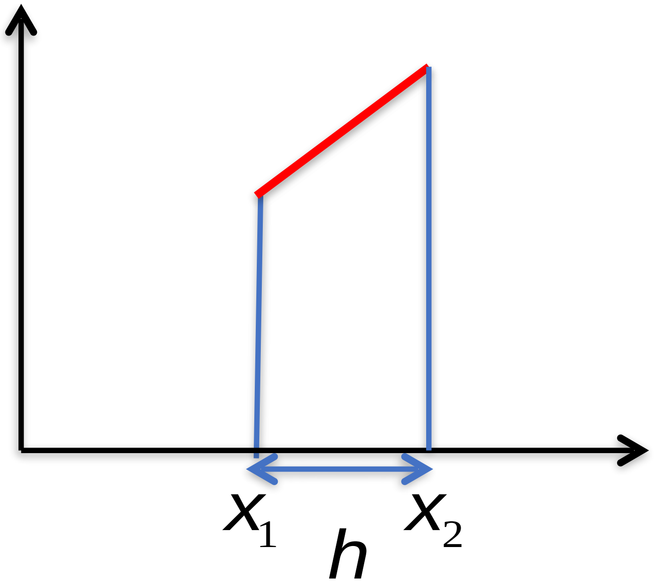
truncation error



Area under the trapezoid $= \frac{1}{2}(l_1 + l_2)b$

Trapezoidal Rule:

Two points (x_1, y_1) (x_2, y_2)



$$I = \int_{x_1}^{x_2} y\,dx$$

$$\approx \int_{x_1}^{x_2} \hat{y}\,dx$$

$$= \int_{x_1}^{x_2} [y_1 + (x - x_1)m]\,dx$$

$$= (y_1 - mx_1)x \Big|_{x_1}^{x_2} + m\frac{x^2}{2}\Big|_{x_1}^{x_2}$$

$$= (y_1 - mx_1)(x_2 - x_1) + \frac{m(x_2{}^2 - x_1{}^2)}{2}$$

$$= y_1 h - x_1(y_2 - y_1) + \frac{(y_2 - y_1)}{2}(x_2 + x_1)$$

$$= y_1\left[h + x_1 - \frac{x_2}{2} - \frac{x_1}{2}\right] + y_2\left[\frac{x_2}{2} - \frac{x_1}{2}\right]$$

$$= \frac{h}{2}(y_1 + y_2) \quad \rightarrow \text{ Trapezoidal rule}$$

$$\frac{\hat{y} - y_1}{x - x_1} = \frac{y_2 - y_1}{x_2 - x_1}$$

$$\hat{y} = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x - x_1) + y_1$$

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

Newton's Forward Difference Method

$$y(\alpha) = y_1 + \Delta y_1 \alpha + \frac{\Delta^2 y_1}{2!}\alpha(\alpha - 1) + \ldots\ldots + \frac{\Delta^{N-1} y_1}{(N-1)!}\alpha(\alpha - 1)..(\alpha - (N-2))$$

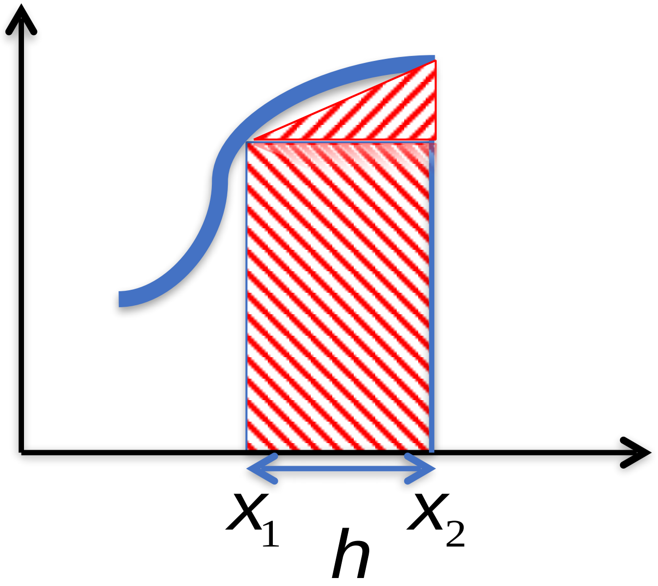
$$= y_1 + \alpha \Delta y_1 + \frac{h^2 \alpha(\alpha-1)}{2!} f''(\xi)$$

$$\begin{aligned}
\Delta^2 y_1 &= \Delta(\Delta y_1) \\
&= \Delta(y_2 - y_1) \\
&= (\Delta y_2 - \Delta y_1) \\
&= ((y_3 - y_2) - (y_2 - y_1)) \\
&= y_3 - 2y_2 + y_1
\end{aligned}$$

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

Two points (x_1, y_1) (x_2, y_2)

$$I = \int_{x_1}^{x_2} y dx$$



$$\alpha = \frac{x - x_1}{h} \Rightarrow \boxed{dx = h d\alpha}$$

$$\begin{aligned} x&: \ x_1 \to x_2 \\ \alpha&: \ 0 \to 1 \end{aligned}$$

$$\int_{x_1}^{x_2} y dx \to \int_0^1 yh\, d\alpha$$

$$= \int_0^1 h[y_1 + \alpha \Delta y_1]\, d\alpha + \int_0^1 \frac{h^3(\alpha(\alpha-1))}{2!} f''(\xi) d\alpha$$

$$= \int_0^1 h[y_1 + \alpha\Delta y_1]\, d\alpha + \int_0^1 \frac{h^3(\alpha(\alpha-1))}{2!} f''(\xi)\, d\alpha$$

$$= hy_1\alpha \Big|_0^1 + h\Delta y_1 \frac{\alpha^2}{2} \Big|_0^1 + \frac{h^3 f''(\xi)}{2!} \left(\frac{\alpha^3}{3} - \frac{\alpha^2}{2}\right) \Big|_0^1$$

$$= hy_1 + \frac{h\Delta y_1}{2} + \frac{h^3 f''(\xi)}{2} \left(\frac{1}{3} - \frac{1}{2}\right)$$

$$= hy_1 + \frac{h(y_2 - y_1)}{2} - \frac{h^3 f''(\xi)}{12}$$

$$= \frac{h(y_1 + y_2)}{2} - \frac{h^3 f''(\xi)}{12}$$

Trapezoidal Rule

Error

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Numerical Integration

Newton Cotes Formula

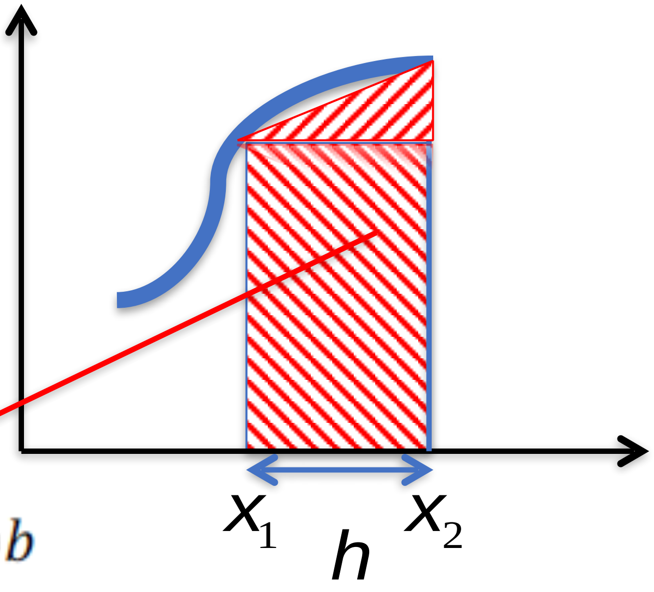
Prof. Jayati Sarkar

Trapezoidal Rule:

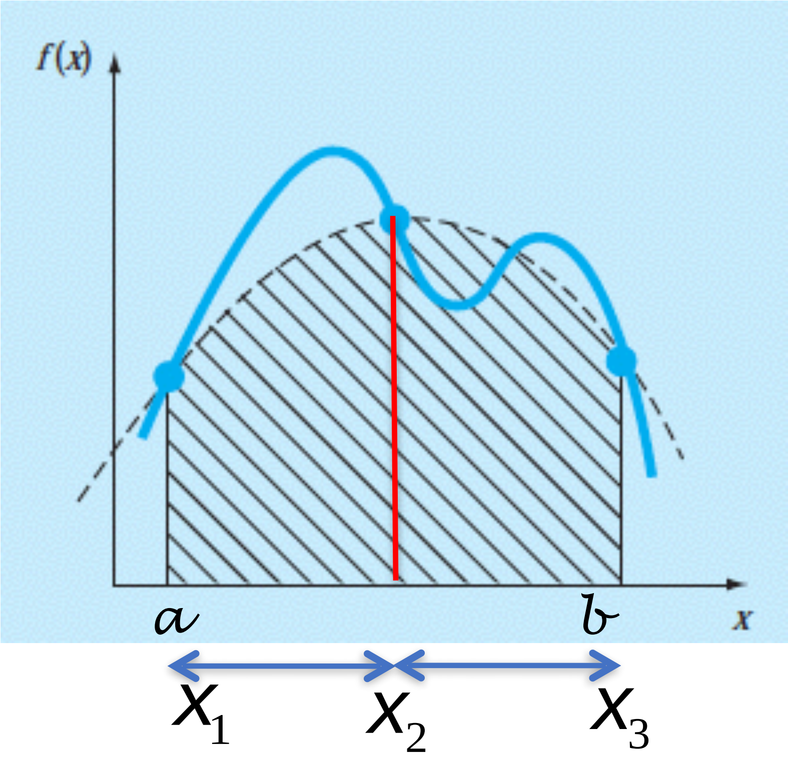
$$\boxed{I = \frac{1}{2}\left(f(x_1) + f(x_2)\right)\left(x_2 - x_1\right)} \quad + \ \theta(h^2)$$

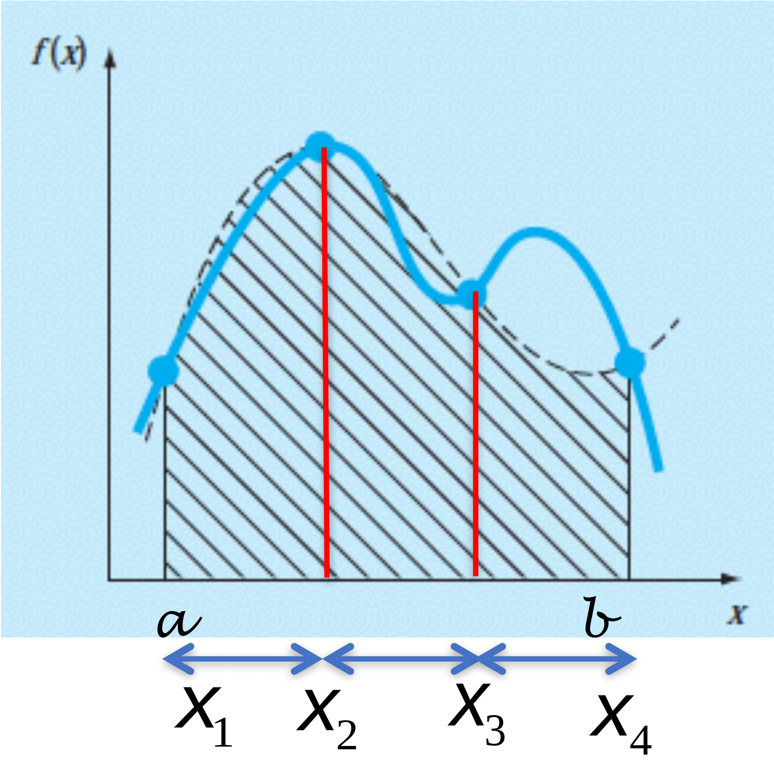
truncation error

Area under the trapezoid = $\frac{1}{2}(l_1 + l_2)b$



Simpsons 1/3rd and 3/8th Rule





$$h=(b-a)/2$$

$$y(\alpha)=y_1+\Delta y_1\alpha+\frac{\Delta^2 y_1}{2!}\alpha(\alpha-1)$$

$$E(\alpha)=\frac{h^3 f'''(\xi)}{3!}\alpha(\alpha-1)(\alpha-2)$$

$$h=(b-a)/3$$

$$y(\alpha)=y_1+\Delta y_1\alpha+\frac{\Delta^2 y_1}{2!}\alpha(\alpha-1)+\frac{\Delta^3 y_1}{3!}\alpha(\alpha-1)(\alpha-2)$$

$$E(\alpha)=\frac{h^4 f^{iv}(\xi)}{4!}\alpha(\alpha-1)(\alpha-2)(\alpha-3)$$

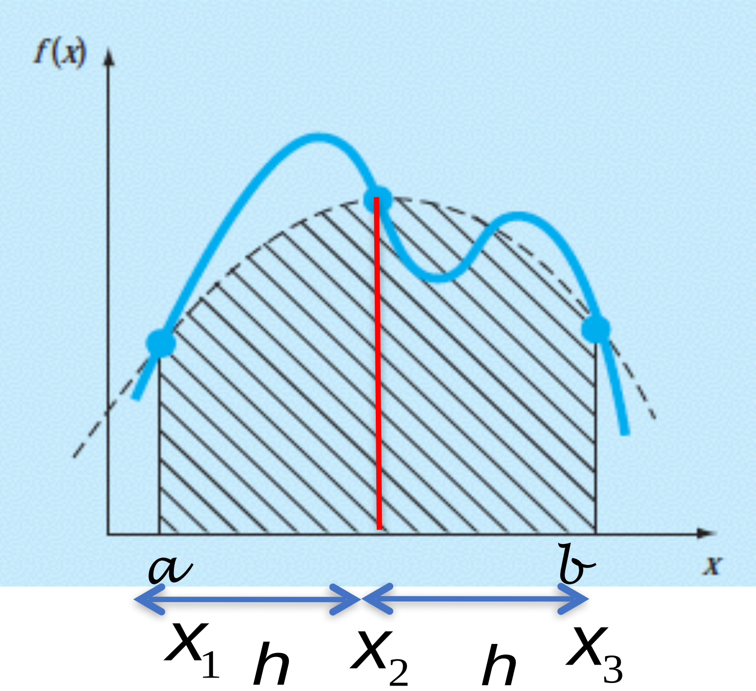
Simpsons 1/3rd Rule

$$\alpha = \frac{x - x_1}{h} \Rightarrow \boxed{dx = h d\alpha} \qquad h = \frac{(b-a)}{2}$$

$$x : x_1 \rightarrow x_3$$

$$\alpha : 0 \rightarrow 2$$

$$\int_{x_1}^{x_3} y\,dx \rightarrow \int_0^2 y h\,d\alpha$$



$$I = \int_0^2 [y_1 + \Delta y_1 \alpha + \frac{\Delta^2 y_1}{2!}\alpha(\alpha - 1)] h d\alpha$$

$$= h y_1 \alpha + \frac{h \Delta y_1 \alpha^2}{2!} + \frac{h \Delta^2 y_1}{2!}\left[\frac{\alpha^3}{3} - \frac{\alpha^2}{2}\right]\Big|_0^2$$

$$= h y_1 (2 - 0) + \frac{h \Delta y_1}{2!}(2^2 - 0^2) + \frac{h \Delta^2 y_1}{2!}\left[\frac{2^3}{3} - \frac{2^2}{2}\right]$$

$$= h\left[2y_1 + 2\Delta y_1 + \Delta^2 y_1\left[\frac{4}{3} - 1\right]\right]$$

$$= h\left[2y_1 + 2(y_2 - y_1) + \frac{1}{3}(y_3 - 2y_2 + y_1)\right]$$

$$= \frac{h}{3}[y_1(6 - 6 + 1) + y_2(6 - 2) + y_3 . 1]$$

$$I = \frac{h(y_1 + 4y_2 + y_3)}{3}$$

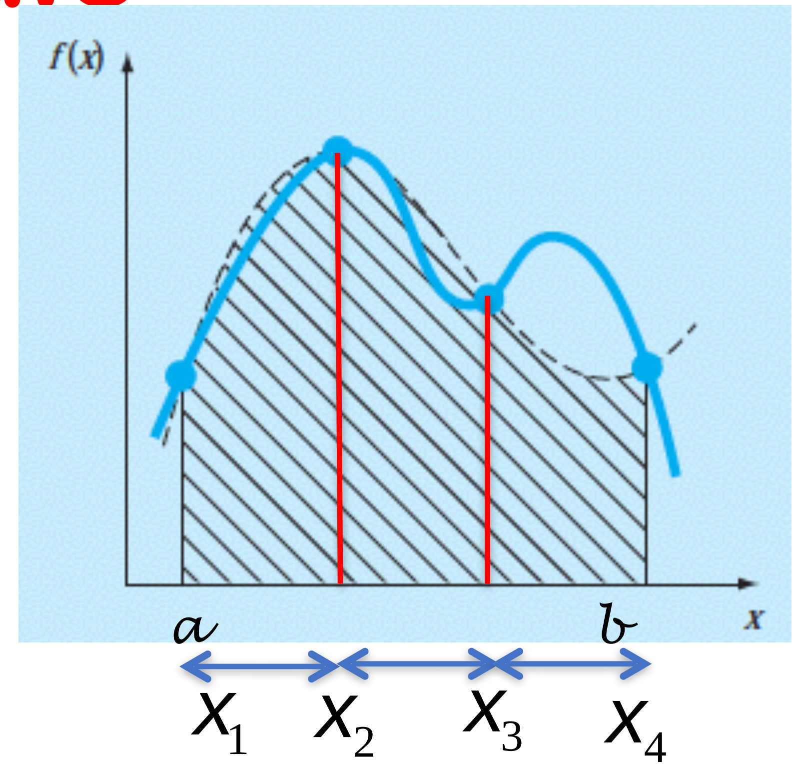
Simpson's $\frac{1}{3^{rd}}$ rule

Simpsons 3/8th Rule

$$\alpha = \frac{x - x_1}{h} \Rightarrow \boxed{dx = h d\alpha} \qquad h = \frac{(b-a)}{3}$$

$$x : x_1 \to x_4 \qquad \int_{x_1}^{x_4} y\,dx \to \int_0^3 y h\, d\alpha$$

$$\alpha : 0 \to 3$$



$$I = \int_0^3 \left[y_1 + \Delta y_1 \alpha + \frac{\alpha(\alpha-1)\Delta^2 y_1}{2!} + \frac{\alpha(\alpha-1)(\alpha-2)\Delta^3 y_1}{3!} \right] h d\alpha$$

$$= \left\{ h y_1 \alpha + \frac{h \Delta y_1 \alpha^2}{2!} + \frac{h\Delta^2 y_1}{2!}\left[\frac{\alpha^3}{3} - \frac{\alpha^2}{2}\right] + \frac{h\Delta^3 y_1}{6}\left[\frac{\alpha^4}{4} - \frac{3\alpha^3}{3} + \frac{2\alpha^2}{3}\right] \right\} \Big|_0^3$$

$$= 3hy_1 + \frac{h\Delta y_1}{2} \times 9 + \frac{h\Delta^2 y_1}{2}\left(9 - \frac{9}{2}\right) + \frac{h\Delta^3 y_1}{6}\left[\frac{81}{4} - 27 + 9\right]$$

$$= 3hy_1 + \frac{9}{2} h\Delta y_1 + \frac{9}{4} h\Delta^2 y_1 + \frac{9}{24} h\Delta^3 y_1$$

$$= 3hy_1 + \frac{9}{2} h(y_2 - y_1) + \frac{9h}{4}(y_3 - 2y_2 + y_1) + \frac{9h}{24}(y_4 - 3y_3 + 3y_2 - y_1)$$

$$= h\left[\frac{9}{24} y_1 + \frac{27}{24} y_2 + \frac{27}{24} y_3 + \frac{9}{24} y_4\right]$$

$$I = \frac{3h}{8}\left[y_1 + 3y_2 + 3y_3 + y_4\right]$$

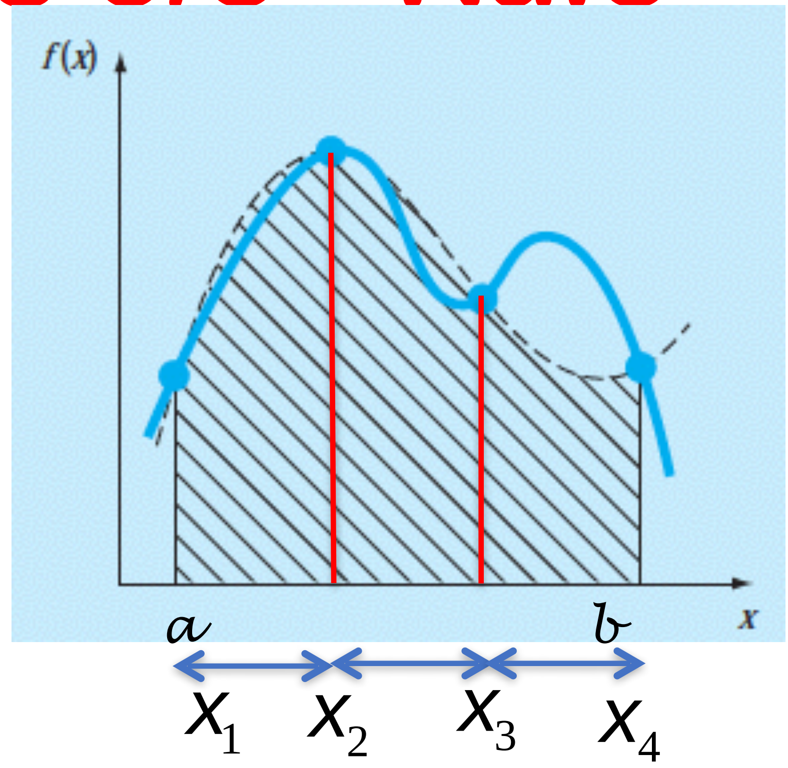
Simpsons 3/8th Rule

Error Analysis of Simpsons 3/8th Rule

$$\alpha = \frac{x - x_1}{h} \Rightarrow \boxed{dx = h\,d\alpha} \qquad h = \frac{(b-a)}{3}$$

$$x : x_1 \to x_4 \qquad \int_{x_1}^{x_4} y\,dx \to \int_0^3 y h\,d\alpha$$

$$\alpha : 0 \to 3$$



$$E(\alpha) = \int_0^3 \frac{h^5 f^{iv}(\xi)}{4!}\alpha(\alpha-1)(\alpha-2)(\alpha-3)d\alpha$$

$$= \frac{h^5 f^{iv}(\xi)}{4!}\int_0^3 \left(\alpha^4 - 6\alpha^3 + 11\alpha^2 - 6\alpha\right)d\alpha$$

$$= \frac{h^5 f^{iv}(\xi)}{4!}\left[\frac{\alpha^5}{5} - \frac{3\alpha^4}{2} + \frac{11\alpha^3}{3} - 3\alpha^2\right]_0^3$$

$$= \frac{h^5 f^{iv}(\xi)}{4!}\left[\frac{3^5}{5} - \frac{3^5}{2} + \frac{11\times 3^3}{3} - 3^3\right]$$

$$= \frac{27h^5 f^{iv}(\xi)}{24}\left[\frac{9}{5} - \frac{9}{2} + \frac{11}{3} - 1\right]$$

$$= \frac{27h^5 f^{iv}(\xi)}{24\times 30}\left[54 - 135 + 110 - 30\right]$$

$$= -\frac{3h^5 f^{iv}(\xi)}{80}$$

$$E = -\frac{3h^5 f^{iv}(\xi)}{80}$$

$$-\frac{(b-a)^5 f^{iv}(\xi)}{6480}$$

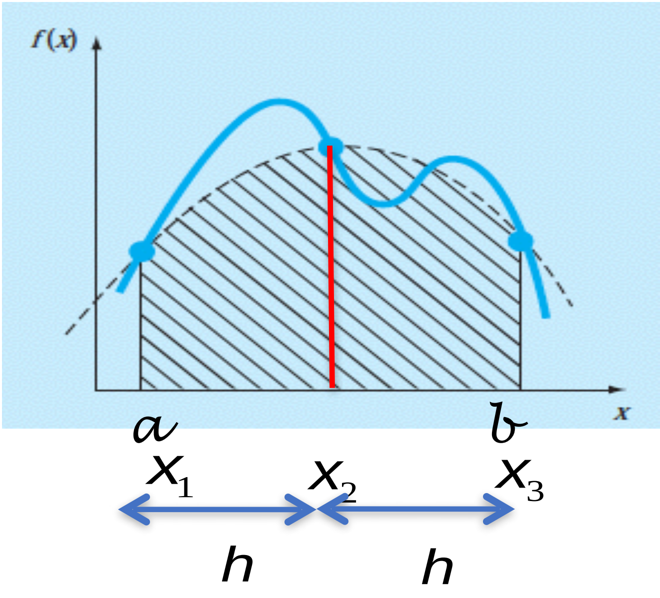
Simpsons 3/8th Rule

Error in Simpsons 1/3rd Rule

$$\alpha = \frac{x - x_1}{h} \Rightarrow \boxed{dx = h\,d\alpha} \qquad h = \frac{(b-a)}{2}$$

$$x : x_1 \to x_3 \qquad \int_{x_1}^{x_3} y\,dx \to \int_0^2 y h\,d\alpha$$

$$\alpha : 0 \to 2$$



$$E(\alpha) = \int_0^2 \frac{h^4 f'''(\xi)}{3!}\alpha(\alpha-1)(\alpha-2)\,d\alpha + \int_0^2 \frac{h^5 f^{iv}(\xi)}{4!}\alpha(\alpha-1)(\alpha-2)(\alpha-3)\,d\alpha$$

$$= \frac{h^4 f'''(\xi)}{3!}\int_0^2 \left(\alpha^3 - 3\alpha^2 + 2\alpha\right)d\alpha + \frac{h^5 f^{iv}(\xi)}{4!}\int_0^2 \left(\alpha^4 - 6\alpha^3 + 11\alpha^2 - 6\alpha\right)d\alpha$$

$$= \frac{h^4 f'''(\xi)}{3!}\left(\frac{\alpha^4}{4} - \alpha^3 + \alpha^2\right)\Bigg|_0^2 + \frac{h^5 f^{iv}(\xi)}{4!}\left(\frac{\alpha^5}{5} - \frac{3}{2}\alpha^4 + \frac{11}{3}\alpha^3 - 3\alpha^2\right)\Bigg|_0^2$$

$$= \frac{h^4 f'''(\xi)}{3!}\left(\frac{2^4}{4} - 2^3 + 2^2\right) + \frac{h^5 f^{iv}(\xi)}{4!}\left(\frac{2^5}{5} - \frac{3}{2}\times 2^4 + \frac{11}{3}\times 2^3 - 3\times 2^2\right)$$

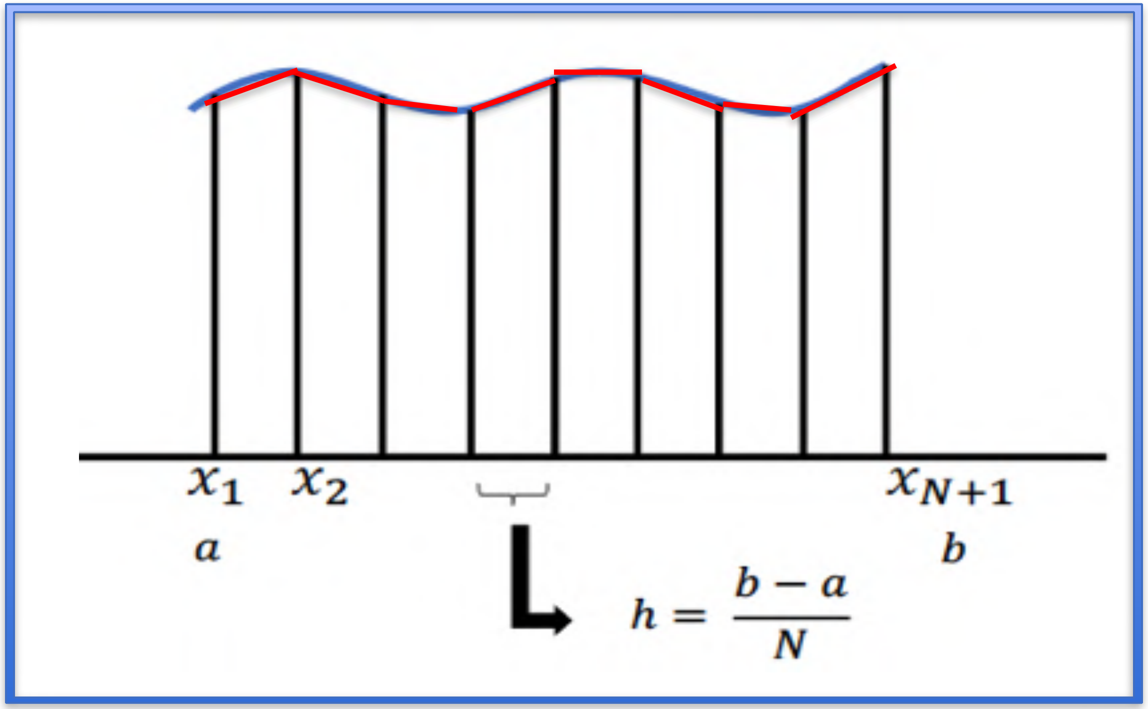
$$= \boxed{0} + \frac{h^5 f^{iv}(\xi)}{4!}\left(\frac{32}{5} - 24 + \frac{88}{3} - 12\right) \quad = -\frac{h^5 f^{iv}(\xi)}{90}$$

$$\boxed{\begin{aligned} E &= -\frac{h^5 f^{iv}(\xi)}{90} \\ &= -\frac{(b-a)^5 f^{iv}(\xi)}{2880} \\ &\text{Simpson's } \tfrac{1}{3rd} \text{ rule} \end{aligned}}$$

Repeated use of Trapezoidal Rule

$$I = \int_{x_1}^{x_{N+1}} y\,dx$$

$$I = \int_{x_1}^{x_2} f(x)dx + \int_{x_2}^{x_3} f(x)dx + \dots\dots\dots + \int_{x_n}^{x_{n+1}} f(x)dx$$



$$= \frac{h\left[f(x_1) + \boxed{f(x_2)}\right]}{2} + \frac{h\left[\boxed{f(x_2)} + \boxed{f(x_3)}\right]}{2} + \frac{h\left[\boxed{f(x_3)} + f(x_4)\right]}{2} + \dots$$

$$+ \frac{h\left[f(x_{N-1}) + \boxed{f(x_N)}\right]}{2} + \frac{h\left[\boxed{f(x_N)} + f(x_{N+1})\right]}{2}$$

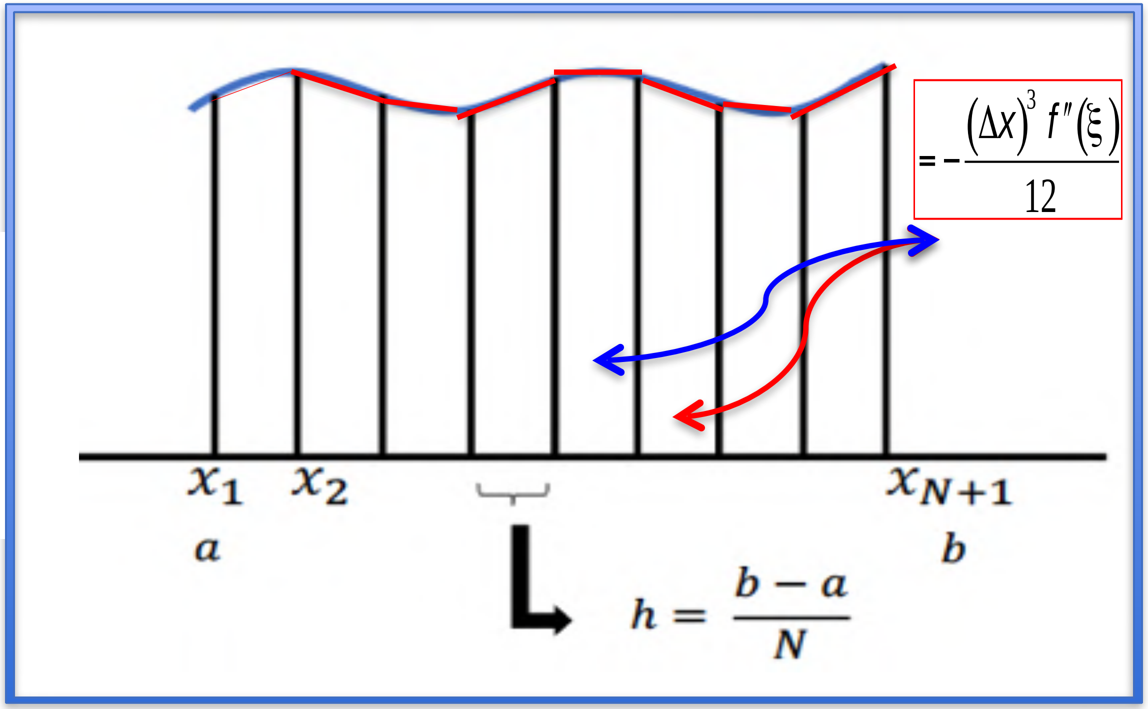
Repeated use of Trapezoidal Rule

$$I = \int_{x_1}^{x_{N+1}} y\,dx$$

$$= \frac{h\left[f(x_1)+f(x_2)\right]}{2} + \frac{h\left[f(x_2)+f(x_3)\right]}{2} + \frac{h\left[f(x_3)+f(x_4)\right]}{2} + \dots$$

$$+ \frac{h\left[f(x_{N-1})+f(x_N)\right]}{2} + \frac{h\left[f(x_N)+f(x_{N+1})\right]}{2}$$

$$= h\frac{\left\{f(x_1)+2\sum_{i=2}^{N} f(x_i)+f(x_{N+1})\right\}}{2}$$



$$= (b-a)\frac{\left\{f(x_1)+2\sum_{i=2}^{N} f(x_i)+f(x_{N+1})\right\}}{2N}$$

WIDTH

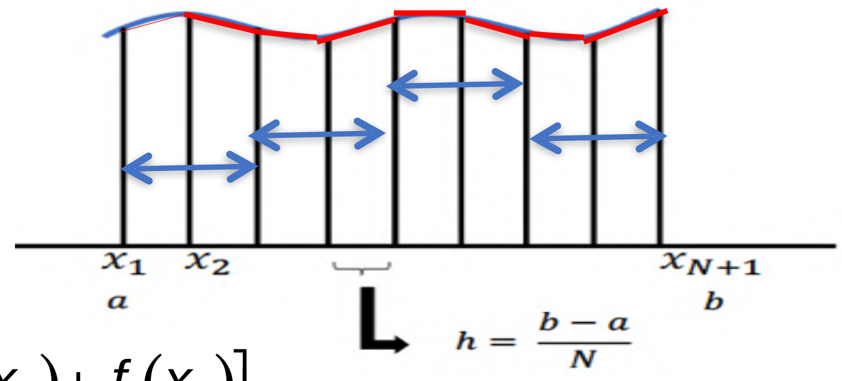
AVERAGE HEIGHT

$$\text{Error} = \frac{-(b-a)^3}{12N^3}\sum_{i=1}^{N} f''(\xi_i)$$

$$= -\frac{(b-a)^3}{12N^3} N \times \overline{f''} \propto \frac{1}{N^2}$$

Repeated use of Simpson`s 1/3rd Rule

$$I=\int_{x_1}^{x_{N+1}} y\,dx=\int_{x_1}^{x_3} y\,dx+\int_{x_3}^{x_5} y\,dx+\ldots+\int_{x_{N-1}}^{x_{N+1}} y\,dx$$



$$=\frac{h\left[f(x_1)+4f(x_2)+f(x_3)\right]}{3}+\frac{h\left[f(x_3)+4f(x_4)+f(x_5)\right]}{3}+\frac{h\left[f(x_5)+4f(x_6)+f(x_7)\right]}{3}+\ldots$$

$$+\frac{h\left[f(x_{N-3})+4f(x_{N-2})+f(x_{N-1})\right]}{3}+\frac{h\left[f(x_{N-1})+4f(x_N)+f(x_{N+1})\right]}{3}$$

$$\boxed{=-\frac{(\Delta x)^5 f^{iv}(\xi)}{90}}$$

$$=\frac{h}{3}\left[f(x_1)+4\sum_{i=2,4,6}^{N} f(x_i)+2\sum_{i=1,3,5}^{N-1} f(x_i)+f(x_{N+1})\right]$$

$$=(b-a)\frac{\left[f(x_1)+4\sum_{i=2,4,6}^{N} f(x_i)+2\sum_{i=1,3,5}^{N-1} f(x_i)+f(x_{N+1})\right]}{3N}$$

WIDTH

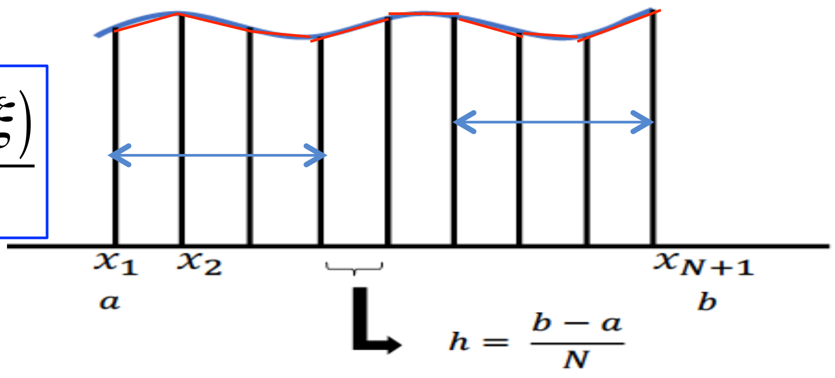
AVERAGE HEIGHT

$$Error=-\frac{(b-a)^5}{90N^5}\sum_{i=1}^{N/2} f^{iV}(\xi_i)$$

$$=-\frac{(b-a)^5}{180N^5}N\times\overline{f^{iV}}\propto\frac{1}{N^4}$$

Repeated use of Simpson`s 3/8th Rule

$$I=\int_{x_1}^{x_{N+1}} y\,dx=\int_{x_1}^{x_4} y\,dx+\int_{x_4}^{x_7} y\,dx+\ldots+\int_{x_{N-2}}^{x_{N+1}} y\,dx \quad \boxed{=-\frac{3(\Delta x)^5 f^{iv}(\xi)}{80}}$$



$$=\frac{3h\left[f(x_1)+3f(x_2)+3f(x_3)+f(x_4)\right]}{8}+\frac{3h\left[f(x_4)+3f(x_5)+3f(x_6)+f(x_7)\right]}{8}+\frac{3h\left[f(x_7)+3f(x_8)+3f(x_9)+f(x_{10})\right]}{8}+\ldots$$

$$+\frac{3h\left[f(x_{10})+3f(x_{11})+3f(x_{12})+f(x_{13})\right]}{8}+\ldots+\frac{3h\left[f(x_{N-2})+3f(x_{N-1})+3f(x_N)+f(x_{N+1})\right]}{8}$$

$$=\frac{3h}{8}\left[f(x_1)+3\sum_{i=2,5,8,..}^{N-1} f(x_i)+3\sum_{i=3,6,9..}^{N} f(x_i)+2\sum_{i=4,7,10...}^{N-2} f(x_i)+f(x_{N+1})\right]$$

$$=(b-a)\times\frac{3}{8N}\left[f(x_1)+3\sum_{i=2,5,8,..}^{N-1} f(x_i)+3\sum_{i=3,6,9..}^{N} f(x_i)+2\sum_{i=4,7,10...}^{N-2} f(x_i)+f(x_{N+1})\right]$$

WIDTH

AVERAGE HEIGHT

$$\boxed{\begin{aligned} Error &=-\frac{3(b-a)^5}{80N^5}\sum_{i=1}^{N/3} f^{iV}(\xi_i) \\ &=-\frac{(b-a)^5}{80N^5}N\times\overline{f^{iV}}\propto\frac{1}{N^4}\end{aligned}}$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Numerical Integration

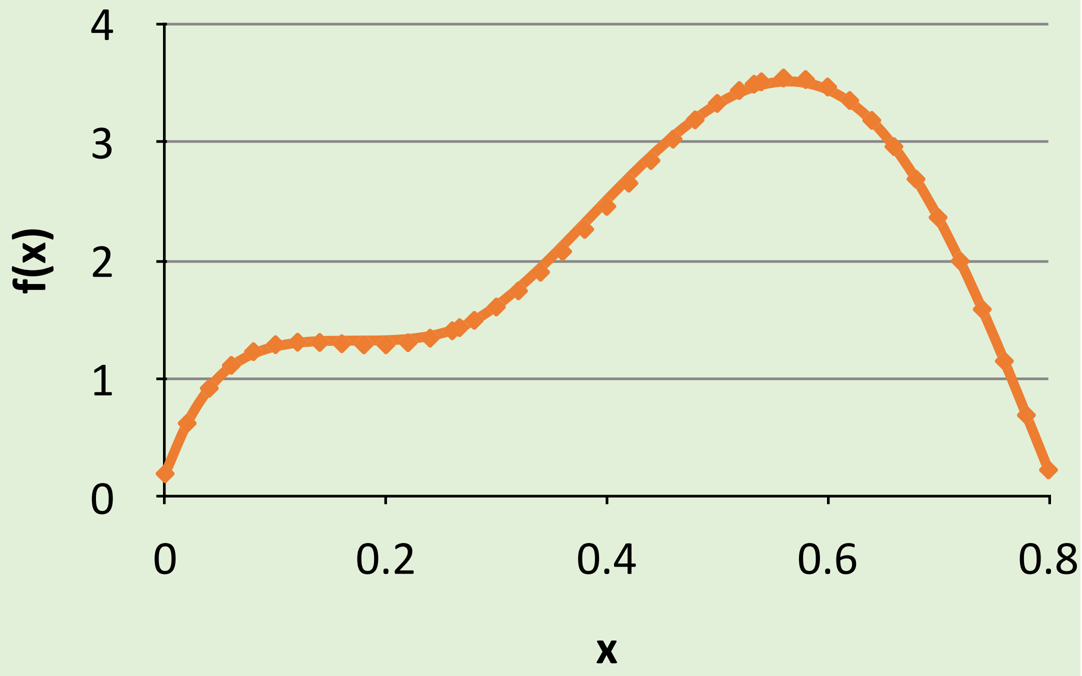
Newton Cotes Formula: Worked Out Examples

Prof. Jayati Sarkar

Integration-Example

$$f(x) = 0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$$

$$\boxed{I = \int_0^{0.8} f(x)\,dx}$$



Analytical Solution:

$$I = 0.2x + \frac{25}{2}x^2 - \frac{200}{3}x^3 + \frac{675}{4}x^4 - \frac{900}{5}x^5 + \frac{400}{6}x^6 \Big|_0^{0.8}$$

$$\boxed{I = 1.6405333}$$

Trapezoidal Rule

Analytical Solution: 1.6405333

Single Domain, N=1

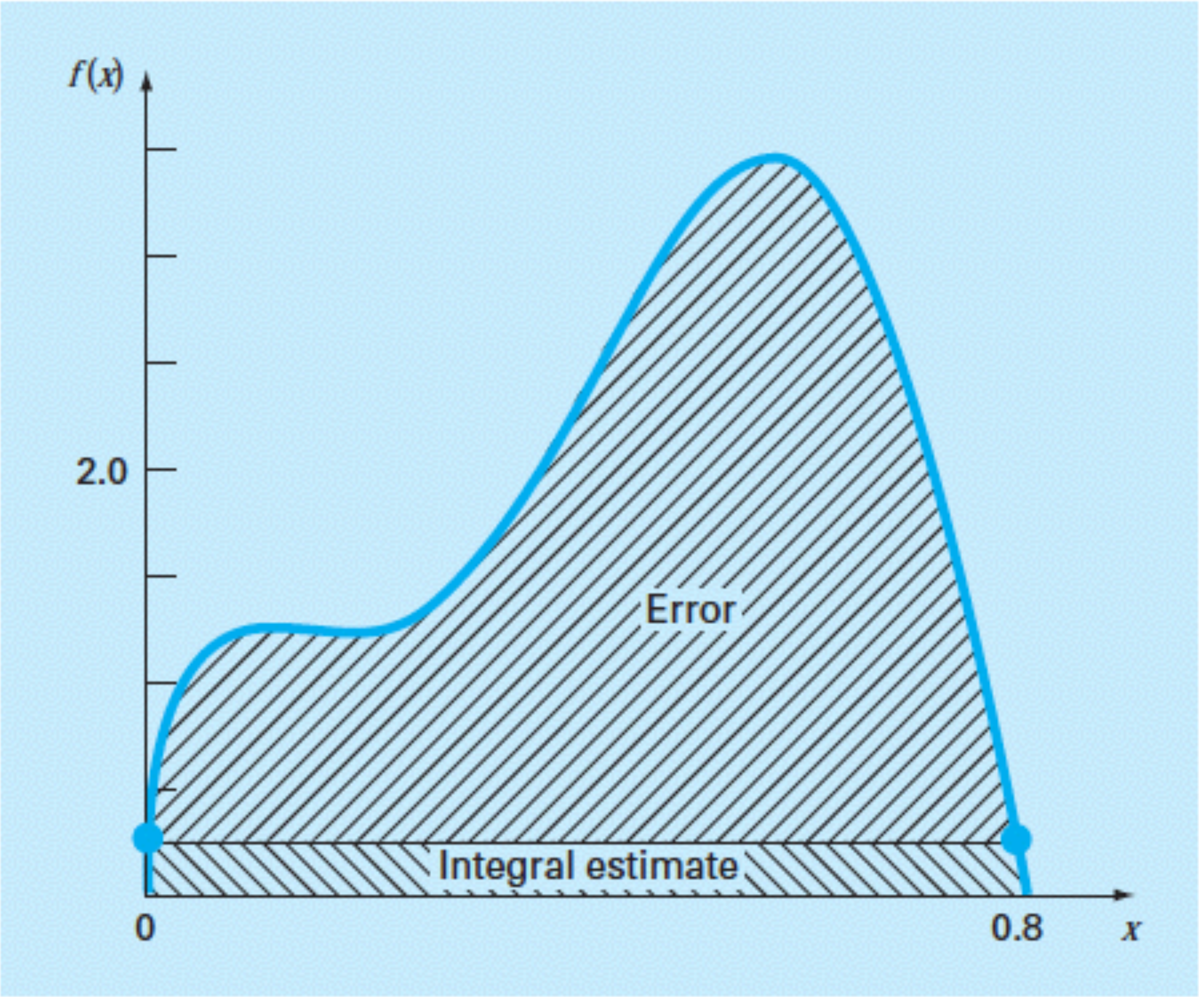
$$\boxed{h = 0.8}$$

$$I = \frac{h}{2}\left(f\left(x_1\right) + f\left(x_2\right)\right)$$

$$= \frac{h}{2}\left(f\left(0\right) + f\left(0.8\right)\right)$$

$$= \frac{0.8}{2}\left(0.2 + 0.232\right)$$

$$= 0.1728$$



Trapezoidal Rule

Analytical Solution: 1.6405333

2 Domains, N=2:

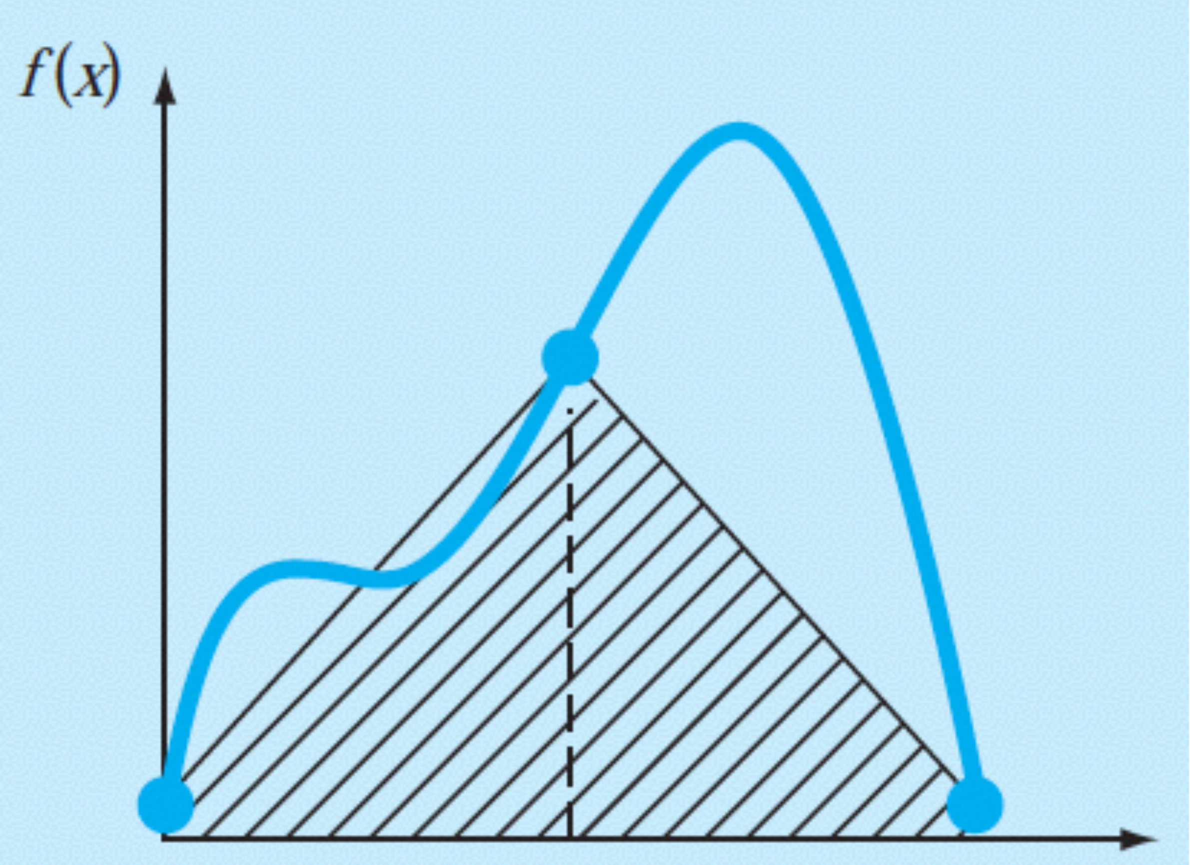
$$\boxed{h = 0.4}$$

$$I = \frac{h}{2}\left(f(x_1) + 2\times f(x_2) + f(x_3)\right)$$

$$= \frac{h}{2}\left(f(0) + 2f(0.4) + f(0.8)\right)$$

$$= \frac{0.4}{2}\left(0.2 + 2\times 2.456 + 0.232\right)$$

$$= 1.0688$$



x_1 x_2 x_3

Trapezoidal Rule

Analytical Solution: 1.6405333

3 Domains, N=3:

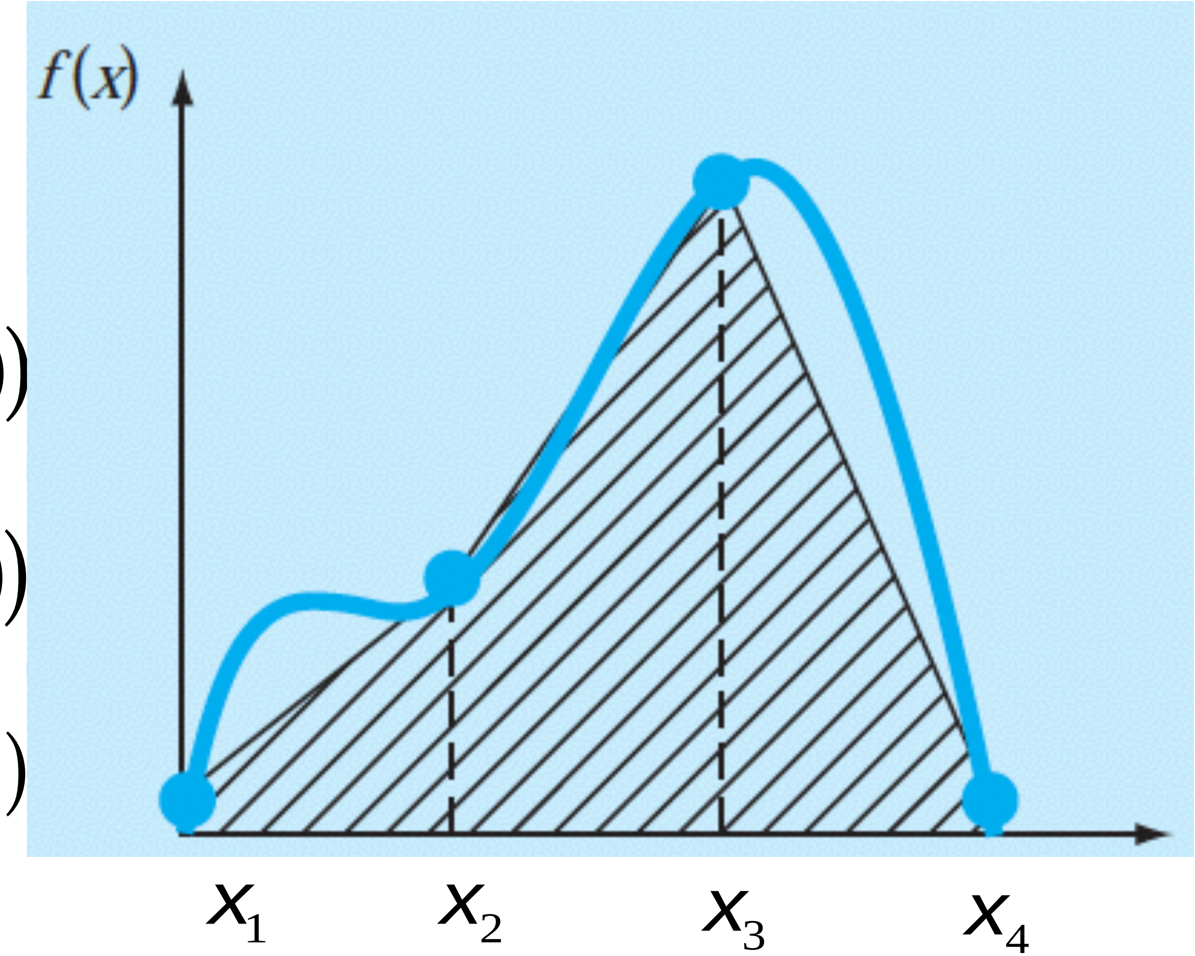
$$\boxed{h = 0.8/3}$$

$$I = \frac{h}{2}\left(f(x_1) + 2\times f(x_2) + 2\times f(x_3) + f(x_4)\right)$$

$$= \frac{h}{2}\left(f(0) + 2f(0.2667) + 2f(0.5334) + f(0.8)\right)$$

$$= \frac{0.2667}{2}\left(0.2 + 2\times 1.4329 + 2\times 3.4872 + 0.232\right)$$

$$= 1.3698$$



Trapezoidal Rule

Analytical Solution: 1.6405333

4 Domains, N=4:

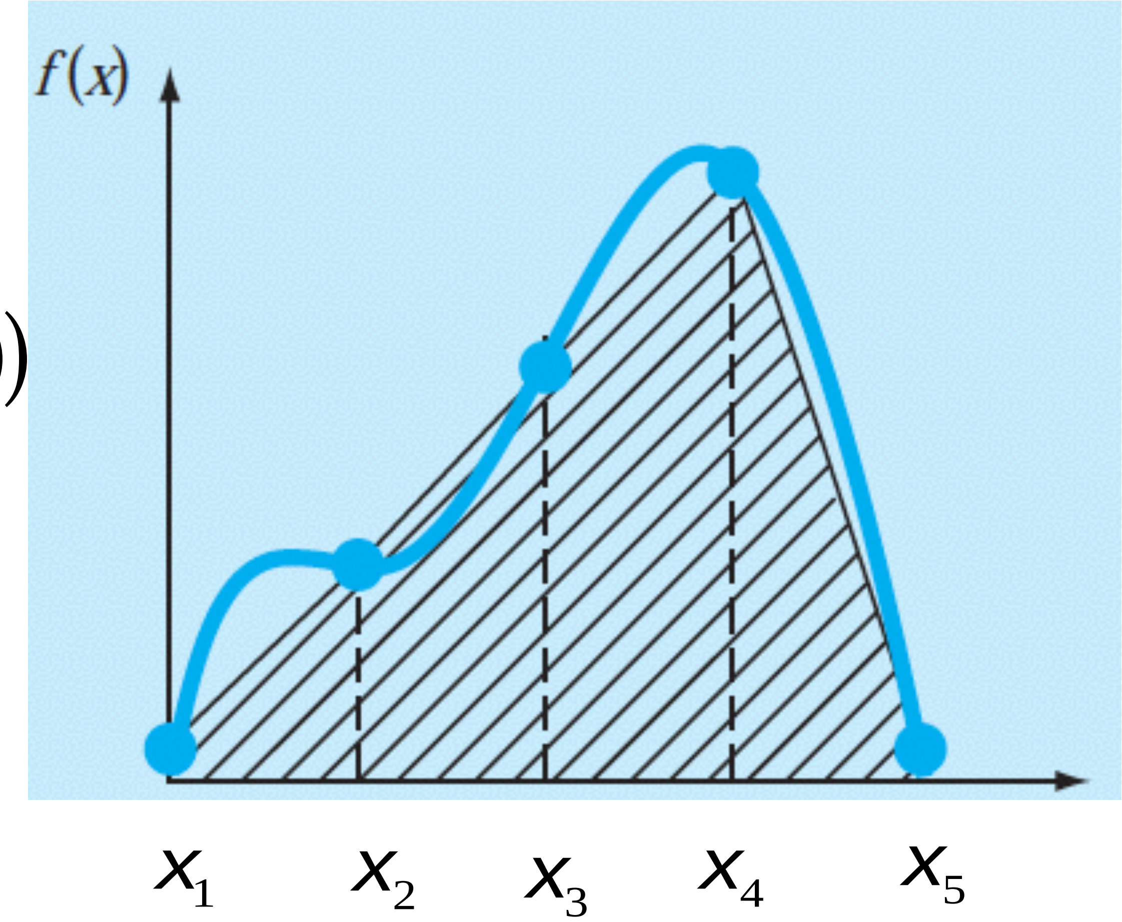
$$\boxed{h = 0.8/4}$$

$$I = \frac{h}{2}\left(f\left(x_1\right) + 2\times f\left(x_2\right) + 2\times f\left(x_3\right) + 2\times f\left(x_4\right) + f\left(x_5\right)\right)$$

$$= \frac{h}{2}\left(f\left(0\right) + 2f\left(0.2\right) + 2f\left(0.4\right) + 2f\left(0.6\right) + f\left(0.8\right)\right)$$

$$= \frac{0.2}{2}\left(0.2 + 2\times 1.288 + 2\times 2.456 + +2\times 3.464 + 0.232\right)$$

$$= 1.4848$$



Error Estimate in Trapezoidal Rule

$$= -\frac{(b-a)^3}{12N^2} \times \overline{f''}$$

$$f(x) = 0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$$

$$f'(x) = 25 - 400x + 2025x^2 - 3600x^3 + 2000x^4$$

$$f''(x) = -400 + 4050x - 10800x^2 + 8000x^3$$

$$\overline{f''(x)} = \frac{\int_0^{0.8} f''(x)dx}{(0.8-0)} = \frac{1}{0.8}\int_0^{0.8}\left(-400 + 4050x - 10800x^2 + 8000x^3\right)dx$$

$$= \frac{1}{0.8}\left(-400x + 4050\frac{x^2}{2} - \frac{10800}{3}x^3 + \frac{8000}{4}x^4\right)\Bigg|_0^{0.8}$$

$$= -60$$

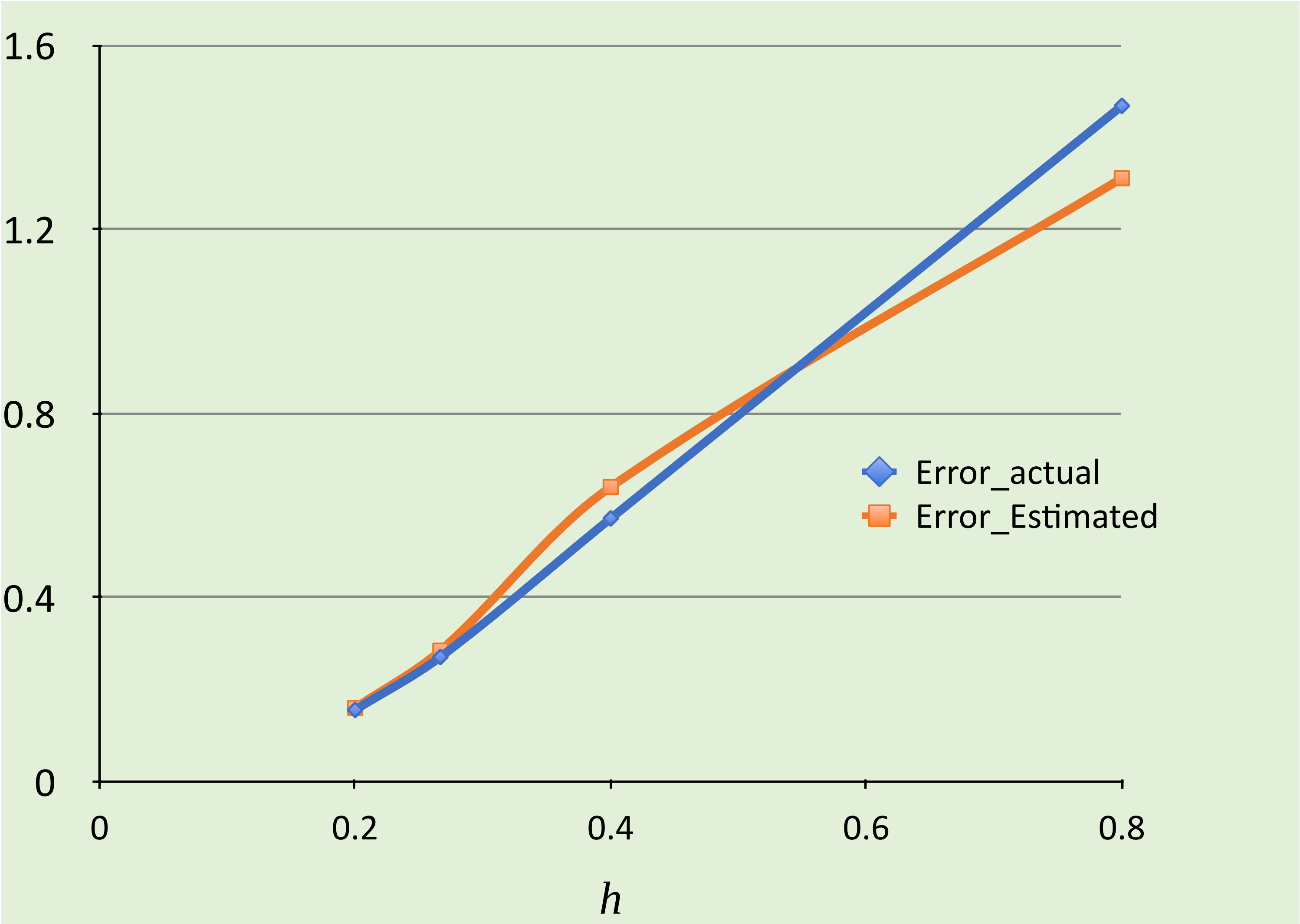
Error in Trapezoidal Rule

Analytical Solution: 1.6405333

$$= -\frac{(b-a)^3}{12N^2} \times \overline{f''}$$

Number of Domains	Actual	Estimated
1	*=1.6405333-0.1728* *=1.4677333* *Error $_{actual}$=1.4677333/1.6405333=89.47%*	*=-(0.8)^3×(-60)/12× (1)^2* *=1.31072* *Error$_{estimatel}$=1.31072 /1.6405333=79.90%*
2	*=1.6405333-1.0688* *=0.5717333* *Error $_{actual}$=0.5717333/1.6405333=34.85%*	*=-(0.8)^3×(-60)/12× (2)^2* *=0.64* *Error$_{estimatel}$=0.64 /1.6405333=39.01%*
3	*=1.6405333-1.3698* *=0.2707333* *Error $_{actual}$=0.2707333/1.6405333=16.50%*	*=-(0.8)^3×(-60)/12× (3)^2* *=0.28444* *Error$_{estimatel}$=0.28444 /1.6405333=17.34%*
4	*=1.6405333-1.4848* *=0.1557333* *Error $_{actual}$=0.1557333/1.6405333=9.49%*	*=-(0.8)^3×(-60)/12× (4)^2* *=0.16* *Error$_{estimatel}$=0.16 /1.6405333=9.75%*

Error in Trapezoidal Rule



Simpson`s 1/3rd Rule

$$\boxed{h = 0.4}$$

$$I = \frac{h}{3}\left(f(x_1) + 4 \times f(x_2) + f(x_3)\right)$$

$$= \frac{h}{3}\left(f(0) + 4 \times f(0.4) + f(0.8)\right)$$

$$= \frac{0.4}{3}\left(0.2 + 4 \times 2.456 + 0.232\right)$$

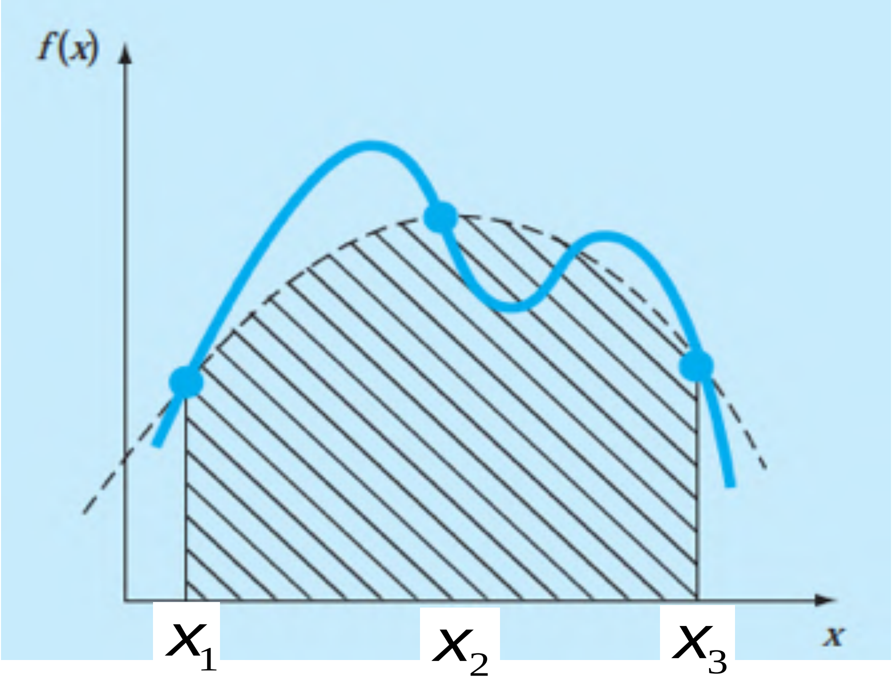
$$= 1.367$$

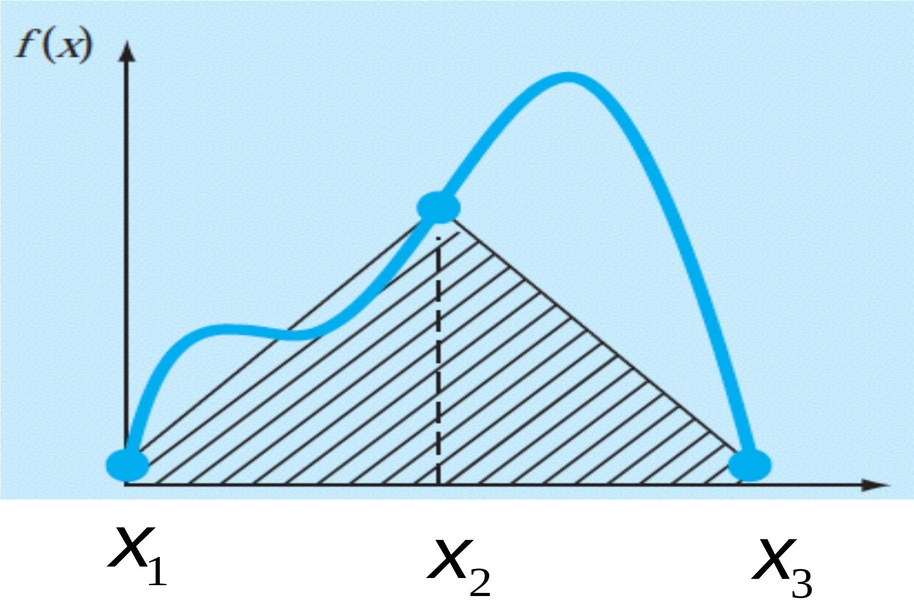
$$E_t = 1.6405333 - 1.367 = 0.2735333$$

$$\boxed{\varepsilon_t = 16.67\%}$$

$$E_{t,2T} = 1.6405333 - 1.0688 = 0.5717333$$

$$\varepsilon_{t,2T} = 34.85\%$$





Error Estimate in Simpsons 1/3rd Rule

$$f(x)=0.2+25x-200x^2+675x^3-900x^4+400x^5$$

$$f'(x)=25-400x+2025x^2-3600x^3+2000x^4$$

$$=-\frac{(b-a)^5}{180N^4}\times\overline{f^{iv}}$$

$$f''(x)=-400+4050x-10800x^2+8000x^3$$

$$f^{iii}(x)=4050-21600x+24000x^2$$

$$f^{iv}(x)=-21600+48000x$$

$$\overline{f^{iv}(x)}=\frac{\int_0^{0.8} f^{iv}(x)dx}{(0.8-0)}=\frac{1}{0.8}\int_0^{0.8}(-21600+48000x)dx$$

$$=\frac{1}{0.8}\left(-21600x+48000\frac{x^2}{2}\right)\Bigg|_0^{0.8}$$

$$=-2400$$

$$=-\frac{(b-a)^5}{180N^4}\times\overline{f^{iv}}$$

$$=-\frac{0.8^5}{180\times 2^4}\times(-2400)$$

$$=0.27307$$

$$E_t=0.2735333$$

Simpson`s 3/8th Rule

$$\boxed{h = 0.2667}$$

$$I = \frac{3h}{8}\left(f(x_1) + 3\times f(x_2) + 3\times f(x_3) + f(x_4)\right)$$

$$= \frac{3}{8}\times\frac{0.8}{3}\left(f(0) + 3\times f(0.2667) + 3\times f(0.5334) + f(0.8)\right)$$

$$= 0.1\left(0.2 + 3\times 1.43287 + 3\times 3.472 + 0.232\right)$$

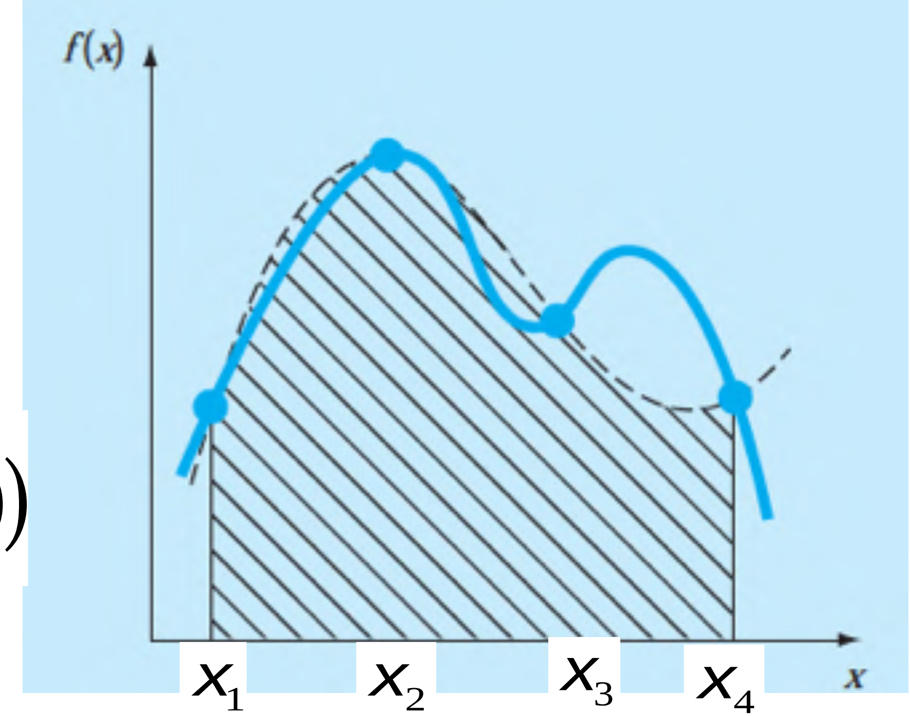
$$= 1.519221$$

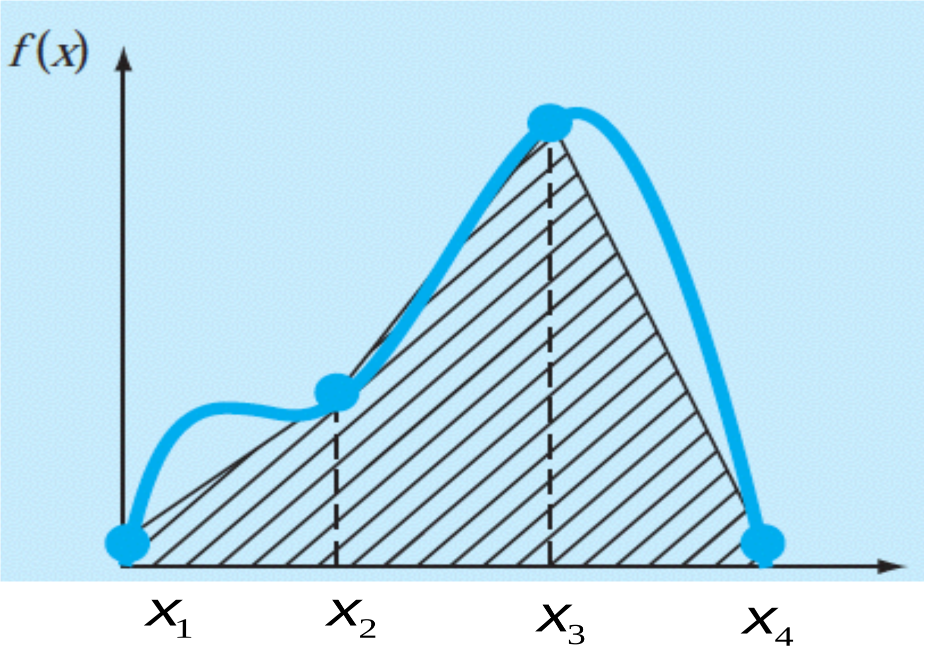
$$E_t = 1.6405333 - 1.519221 = 0.1213123$$

$$\boxed{\varepsilon_t = 7.395\%}$$

$$E_{t,3T} = 1.6405333 - 1.3698 = 0.2707333$$

$$\varepsilon_{t,3T} = 16.5\%$$





Error Estimate in Simpsons 3/8th Rule

$$f(x) = 0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$$

$$f'(x) = 25 - 400x + 2025x^2 - 3600x^3 + 2000x^4$$

$$= -\frac{(b-a)^5}{80N^4} \times \overline{f^{iv}}$$

$$f''(x) = -400 + 4050x - 10800x^2 + 8000x^3$$

$$f^{iii}(x) = 4050 - 21600x + 24000x^2$$

$$f^{iv}(x) = -21600 + 48000x$$

$$\overline{f^{iv}(x)} = \frac{\int_0^{0.8} f^{iv}(x)dx}{(0.8-0)} = \frac{1}{0.8}\int_0^{0.8}(-21600 + 48000x)dx$$

$$= \frac{1}{0.8}\left(-21600x + 48000\frac{x^2}{2}\right)\Bigg|_0^{0.8}$$

$$= -2400$$

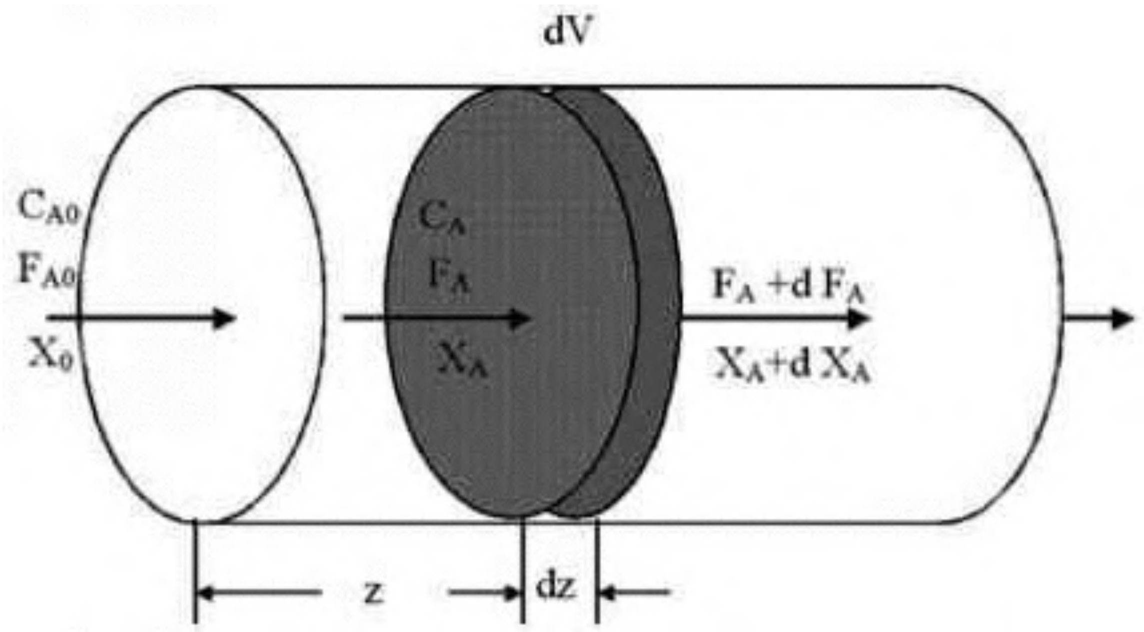
$$= -\frac{(b-a)^5}{80N^4} \times \overline{f^{iv}}$$

$$= -\frac{0.8^5}{80 \times 3^4} \times (-2400)$$

$$= 0.121363$$

$$E_t = 0.1213123$$

Plug Flow Reactor



$$X_A = \frac{F_{A0} - F_A}{F_{A0}}$$

$$dF_A = -F_{A0} dX_A$$

$$-r_A = kC_A^n$$

$$-r_A = kC_{A0}^n (1 - x_A)^n$$

$$F_A(z) - F_A(z + dz) + r_A dV = 0$$

$$V = \int \frac{dF_A}{r_A}$$

$$-dF_A + r_A dV = 0$$

$$V = F_{A0} \int \frac{dx_A}{-r_A}$$

$$V = \frac{F_{A0}}{kC_{A0}^n} \int_0^{X_{A\_EXIT}} \frac{dx_A}{(1 - x_A)^n}$$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Numerical Integration

Newton Cotes Formula: Worked Out Examples

Prof. Jayati Sarkar

Richardson Extrapolation

$$I_{true} = I_h + C_1 h^n + C_2 h^{n+1} \quad \dots\dots\dots\dots (1)$$

$$I_{true} = I_{h/2} + 2C_1 \left(\frac{h}{2}\right)^n + 2C_2 \left(\frac{h}{2}\right)^{n+1} + \dots\dots\dots. (2)$$

$$Eqn\ (2) \times 2^{n-1} + Eqn\ (1) \times -1$$

$$I_{true} = \frac{2^{n-1} I\left(\frac{h}{2}\right) - I(h)}{2^{n-1} - 1} + \frac{-\frac{C_2}{2} h^{n+1}}{2^{n-1} - 1}$$

After Richardson extrapolation:

	n		Order of Error
Trapezoidal Rule	3	$\dfrac{2^2 I\left(\dfrac{h}{2}\right) - I(h)}{2^2 - 1}$	4
1/3rd Simpsons Rule	5	$\dfrac{2^4 I\left(\dfrac{h}{2}\right) - I(h)}{2^4 - 1}$	6
3/8th Simpsons Rule	5	$\dfrac{2^4 I\left(\dfrac{h}{2}\right) - I(h)}{2^4 - 1}$	6

Richardson Extrapolation

$$f(x) = 0.2 + 25x - 250x^2 + 675x^3 - 900x^4 + 400x^5$$

from a = 0   b = 0.8

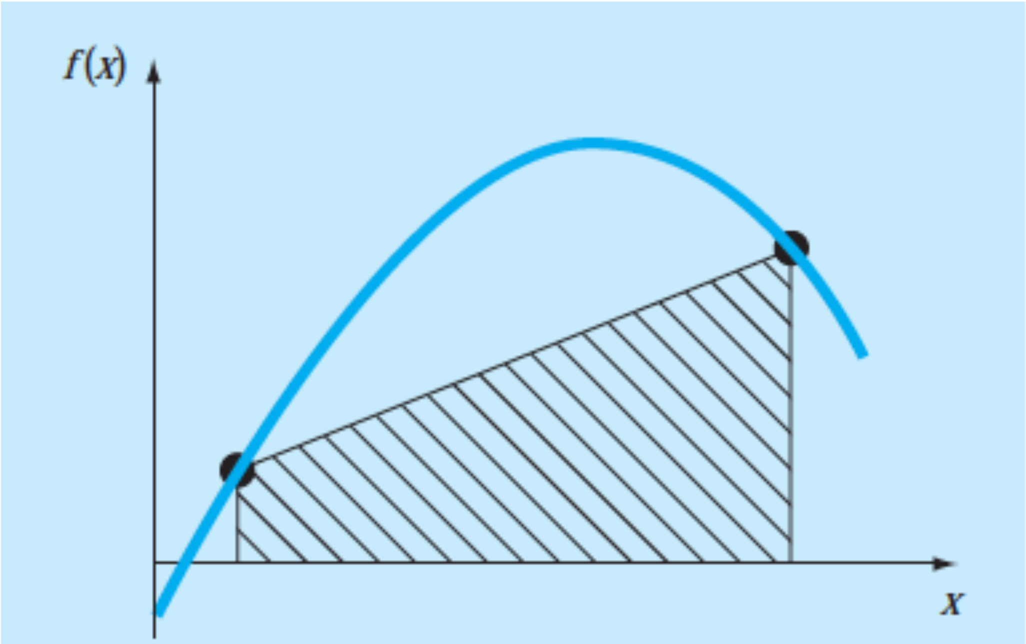
True = 1.640533

Segments	h	Integral	ε_t %
1	0.8	0.1728	89.5
2	0.4	1.0688	34.9
4	0.2	1.4848	9.5

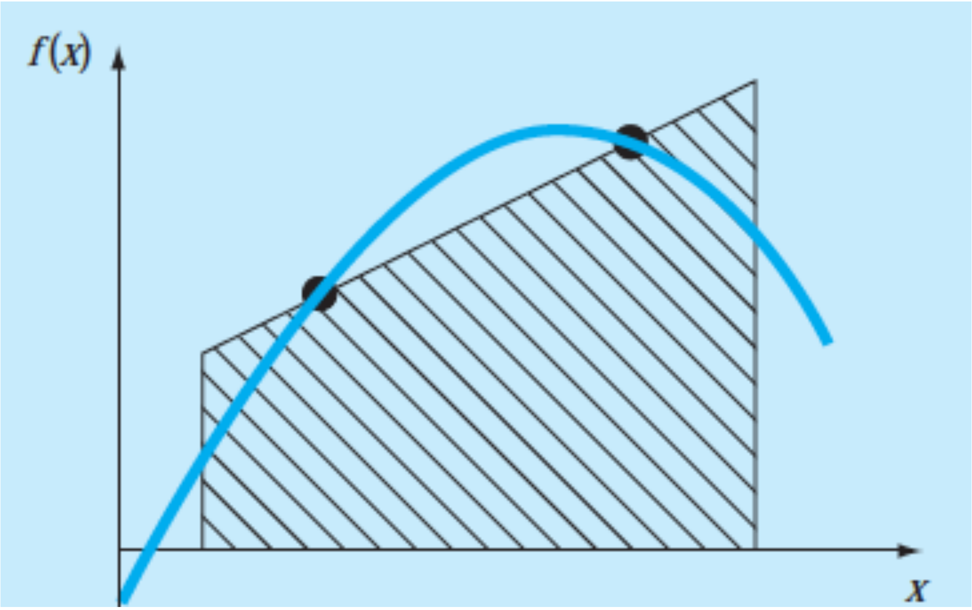
True value: 1.640533

Segments	h	Integral	Richardson Extrapolation (4/3)I(h/2)-I(h)/3	Error(%)
1	0.8	0.1728	1.3675	16.65%
2	0.4	1.0688	1.6235	1.038%
4	0.2	1.4848		

Open Integration



Trapezoidal Rule



Better estimate by passing the straight line through two intermediate points rather than the two extreme points

Trapezoidal Rule

Method of undetermined coefficients:

$$I_{trapezoidal} = a_1 f(a) + a_2 f(b)$$

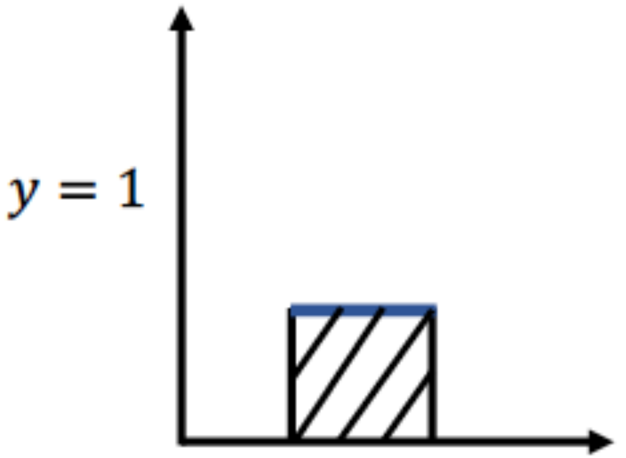
1. $f(x) = 1$

$$a_1 + a_2 = \int_a^b 1\, dx = [x] = b - a$$

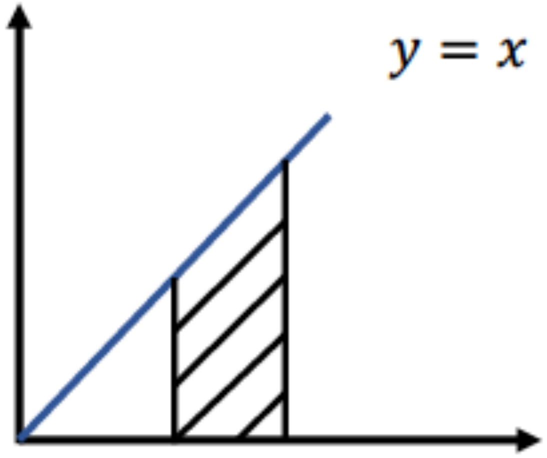
2. $f(x) = x$

$$a_1 . a + a_2 . b = \int_a^b x dx = \frac{x^2}{2}\bigg|_a^b = \frac{b^2 - a^2}{2}$$

$$a_2[b - a] = \frac{b^2 - a^2}{2} - (ba - a^2) = \frac{b^2 + a^2 - 2ab}{2} = \frac{(b-a)^2}{2}$$

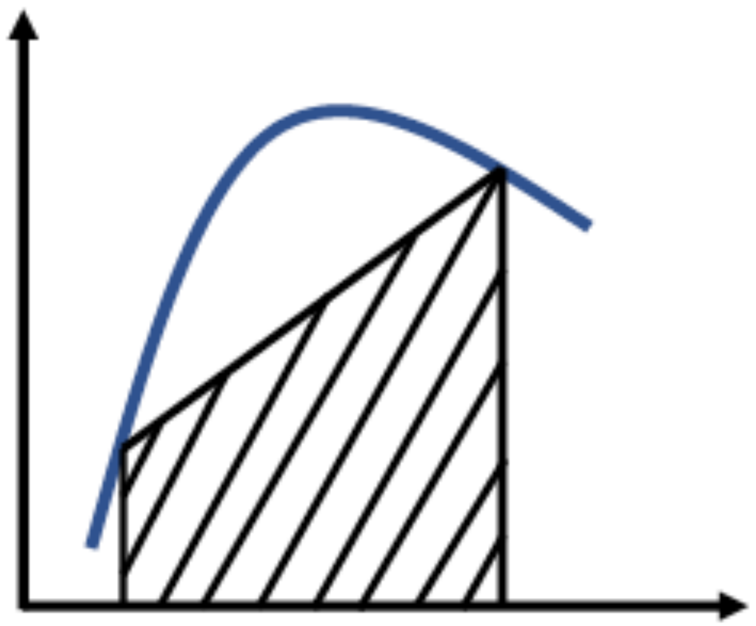


$$\boxed{\begin{aligned} a_2 &= \frac{b-a}{2} \\ a_1 &= \frac{b-a}{2} \end{aligned}}$$

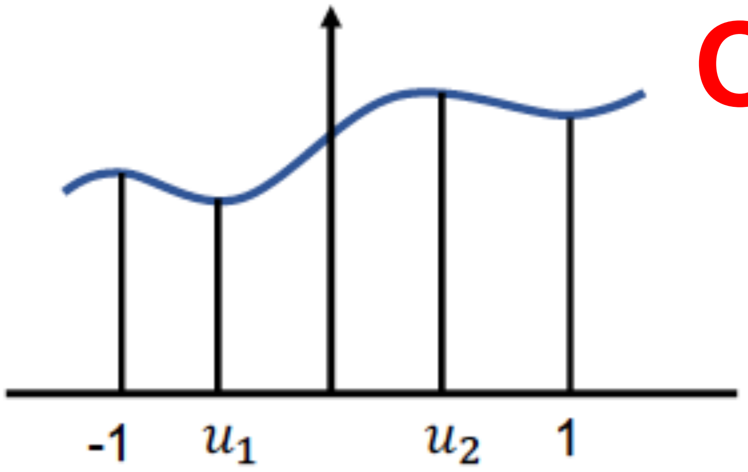


Trapezoidal Rule

$$I_{trapezoidal} = \frac{(b-a)}{2} f(a) + \frac{(b-a)}{2} f(b)$$



Open Integration



$$I_{quad} = a_1 f(u_1) + a_2 f(u_2) \qquad u_1, u_2 \in (-1,\ 1)$$

$$\boxed{\int_{-1}^{1} f(z)\,dz}$$

- Select N points with in this interval such that the integral is a weighted average of functions computed at this N points

$$x = [a, b] \qquad z = [-1, 1]$$

$$x = mz + c$$
$$a = m(-1) + c$$
$$b = m(1) + c$$

$$\Rightarrow c = \frac{a+b}{2}$$

$$\Rightarrow m = \frac{b-a}{2}$$

$$\Rightarrow x = \left(\frac{b-a}{2}\right) Z + \left(\frac{a+b}{2}\right)$$

$$\boxed{\therefore Z = \frac{2}{b-a}\left[x - \left(\frac{a+b}{2}\right)\right]}$$

Gauss Legendre Quadrature Formula

True: $\int_{-1}^{1} f(x)dx$

Numerical: $a_1 f(u_1) + a_2 f(u_2)$

$f(x) = 1 \quad \int_{-1}^{1} 1\, dx = x\Big|_{-1}^{1} = 2$

$a_1.1 + a_2.1$

$f(x) = x \quad \int_{-1}^{1} x dx = \frac{x^2}{2}\Big|_{-1}^{1} = \frac{1^2}{2} - \frac{(-1)^2}{2} = 0$

$a_1.u_1 + a_2.u_2$

$f(x) = x^2 \quad \int_{-1}^{1} x^2 dx = \frac{x^3}{3}\Big|_{-1}^{1} = \frac{1^3}{3} - \frac{(-1)^3}{3} = \frac{2}{3}$

$a_1.{u_1}^2 + a_2.{u_2}^2$

$f(x) = x^3 \quad \int_{-1}^{1} x^3 dx = \frac{x^4}{4}\Big|_{-1}^{1} = \frac{1^4}{4} - \frac{(-1)^4}{4} = 0$

$a_1.{u_1}^3 + a_2.{u_2}^3$

Gauss Legendre Quadrature Formula

$$a_1 + a_2 = 2 \quad \rightarrow (1)$$
$$a_1.u_1 + a_2.u_2 = 0 \quad \rightarrow (2)$$
$$a_1.u_1{}^2 + a_2.u_2{}^2 = \frac{2}{3} \quad \rightarrow (3)$$
$$a_1.u_1{}^3 + a_2.u_2{}^3 = 0 \quad \rightarrow (4)$$

$$u_1 \times eqn(1) - eqn(2)$$
$$\Rightarrow a_2 = \frac{2u_1}{(u_1 - u_2)}$$
$$\Rightarrow a_1 = 2 - \frac{2u_1}{(u_1 - u_2)} = -\frac{2u_2}{(u_1 - u_2)}$$

$$eqn(3) \Rightarrow -\frac{2u_1^2 u_2}{(u_1 - u_2)} + \frac{2u_1 u_2^2}{(u_1 - u_2)} = \frac{2}{3}$$
$$\Rightarrow \frac{2u_1 u_2}{(u_1 - u_2)}(u_2 - u_1) = \frac{2}{3}$$
$$\Rightarrow u_1 u_2 = -\frac{1}{3}$$

$$eqn(4) \Rightarrow -\frac{2u_2}{(u_1 - u_2)} u_1^3 + \frac{2u_1}{(u_1 - u_2)} u_2^3 = 0$$
$$\Rightarrow \frac{u_1 u_2}{(u_1 - u_2)}(u_2^2 - u_1^2) = 0$$
$$\Rightarrow u_1 u_2 (u_1 + u_2) = 0$$
$$\Rightarrow u_1 + u_2 = 0$$
$$\Rightarrow u_1 = -u_2$$

Gauss Legendre Quadrature Formula

$$u_1 = \frac{1}{\sqrt{3}},\ u_2 = -\frac{1}{\sqrt{3}}$$

$$a_1 = \frac{-2u_2}{(u_1 - u_2)} = \frac{-2\left(\frac{-1}{\sqrt{3}}\right)}{\left(\frac{1}{\sqrt{3}} - \left(\frac{-1}{\sqrt{3}}\right)\right)} = 1$$

$$a_2 = \frac{2u_1}{(u_1 - u_2)} = \frac{2\left(\frac{1}{\sqrt{3}}\right)}{\left(\frac{1}{\sqrt{3}} - \left(\frac{-1}{\sqrt{3}}\right)\right)} = 1$$

$$I \cong f\left(\frac{-1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

Gauss Legendre Quadrature Formula

Points (n+1)	no of unknowns	Exact for polynomial of degree	Weighting Function	Function Arguments	Truncation error
2	2 × 2 = 4	2 × 2 -1 = 3	$C_o = 1$ $C_1 = 1$	$x_o = -0.57735$ $x_1 = 0.57735$	$\cong f^4(\xi)$
3	2 × 3 = 6	2 × 3 -1 = 5	$C_o = 0.556$ $C_1 = 0.889$ $C_2 = 0.556$	$x_o = -0.775$ $x_1 = 0$ $x_2 = 0.775$	$\cong f^6(\xi)$
n+1	2 × (n+1) = 2n+2	(2n+2) – 1 = 2n+1			

- ✓ Order is same as the no of unknowns

Gauss Legendre Quadrature -Example

$$f(x) = 0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$$

$$\int_0^{0.8} f(x)\,dx$$

$$x = \frac{(0.8-0)}{2} Z + \left(\frac{0.8+0}{2}\right) \qquad x = 0.4Z + 0.4$$

$$I = \int_0^{0.8} f(x)\,dx = \int_{-1}^{1} 0.4 \times f(z)\,dz = \int_{-1}^{1} g(z)\,dz$$

$$f(z) = 0.2 + 25(0.4+0.4Z) - 200(0.4+0.4Z)^2$$
$$+675(0.4+0.4Z)^3 - 900(0.4+0.4Z)^4 + 400(0.4+0.4Z)^5$$

$$g(-0.57735) = \mathbf{0.516740523} \qquad g(0.57735) = \mathbf{1.3058376}$$

$$I_{GQ} = 1 \times g\left(-\frac{1}{\sqrt{3}}\right) + 1 \times g\left(\frac{1}{\sqrt{3}}\right) = \mathbf{1.822578123}$$

=1.6405333-1.822578123=-0.182044823

Error $_{actual}$=-0.182044823/1.6405333=-11.1%

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Ordinary Differential Equations, Initial Value Problem

Runge Kutta Methods, Euler Method

Prof. Jayati Sarkar

Modelling of Chemical kinetics in a batch reactor

Reactions in Series:

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$

Model Equations

$$\boxed{\begin{gathered} \frac{dC_A}{dt} = -k_1 C_A^n \\ \frac{dC_B}{dt} = k_1 C_A^n - k_2 C_B^m \\ \frac{dC_c}{dt} = k_2 C_B^m \end{gathered}}$$

- Initial conditions
  - At t=0,
  - $C_A = C_{Ao}$, $C_B = C_{Bo}$, $C_C = C_{Co}$

These Set of Model Equations can be written in standard form for 1st order ODE

$$\frac{dy_i}{dt} = f_i\left(y_1, y_2, \ldots. y_N\right)$$

for i=1,2,…N

The initial conditions of the above equations are:

$$t = 0, y_i\left(0\right) = \alpha_i \ldots\ldots.(known)$$

- If RHS of coupled ODE contains t, such that
  - $\frac{dy_i}{dt} = f_i(y_1, y_2, \dots \dots \dots \dots . . y_N, t)$ for for $i = $ 1, 2,..........N
  - t=0, $y_i(0) = \alpha_i$ ....(known)

How to recast in standard form?

We introduce a new dependent variable

$$\frac{dy_{N+1}}{dt} = 1$$

and

t=0; $y_{N+1}(0)$=0

Or in general case, t=t_0 ; $y_{N+1}(t_0)$= t_0

- With the introduction of new dependent variable y_{N+1}

$$\frac{dy_i}{dt} = f_i(y_1, y_2, \dots \dots \dots \dots . . y_N, y_{N+1})$$

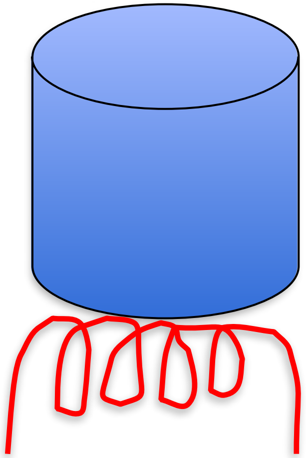
For $i = 1,2, \dots \dots . . N$+1

The new set now has (N+1) coupled ODE

Example

Heating of an hot oil bath with time dependent heat source.

Let T and Q(t) be bath temperature and heat rate respectively.



- Heat balance on bath gives

ambient air temperature

$$V\rho C_p \frac{dT}{dt} = Ah(T - T_a) - VQ(t)$$

Bath volume

area for heat loss

Initial temp. of bath is T_0, before the heater is turned on i.e. t=0; $T = T_0$

Convert to Standard Form

$$y_1 = T\,;\; y_2 = t$$

- The Differential equation for y_2 is

$$\frac{dy_2}{dt} = 1$$

And equation of y_1 is simply heat transfer equation with t replaced by y_2

$$\frac{dy_1}{dt} = \frac{Ah}{V\rho C_p}(y_1 - T_0) - \frac{Q(y_2)}{\rho C_p}$$

- The initial condition for y_1 and y_2 are
  - t = 0; $y_1 = T_0\,;\, y_2 = 0$
- Thus standard form is recovered.

Ordinary DE with higher order derivative

$$y'' + F(y', y) = 0$$

$\boxed{\text{How to recast in standard form?}}$ $\quad y_1 = y\, ;\ y_2 = \frac{dy}{dt}$

$$\frac{dy_2}{dt} = -F(y_2, y_1)$$

Thus new set of equations

$$\frac{dy_1}{dt} = y_2 = f_1(y_1, y_2)$$

$$\frac{dy_2}{dt} = -F\,(y_2, y_1) = f_2(y_1, y_2)$$

If the initial conditions are

$$H_i\big(y'(0), y(0)\big) = 0; \qquad i = 1,2$$

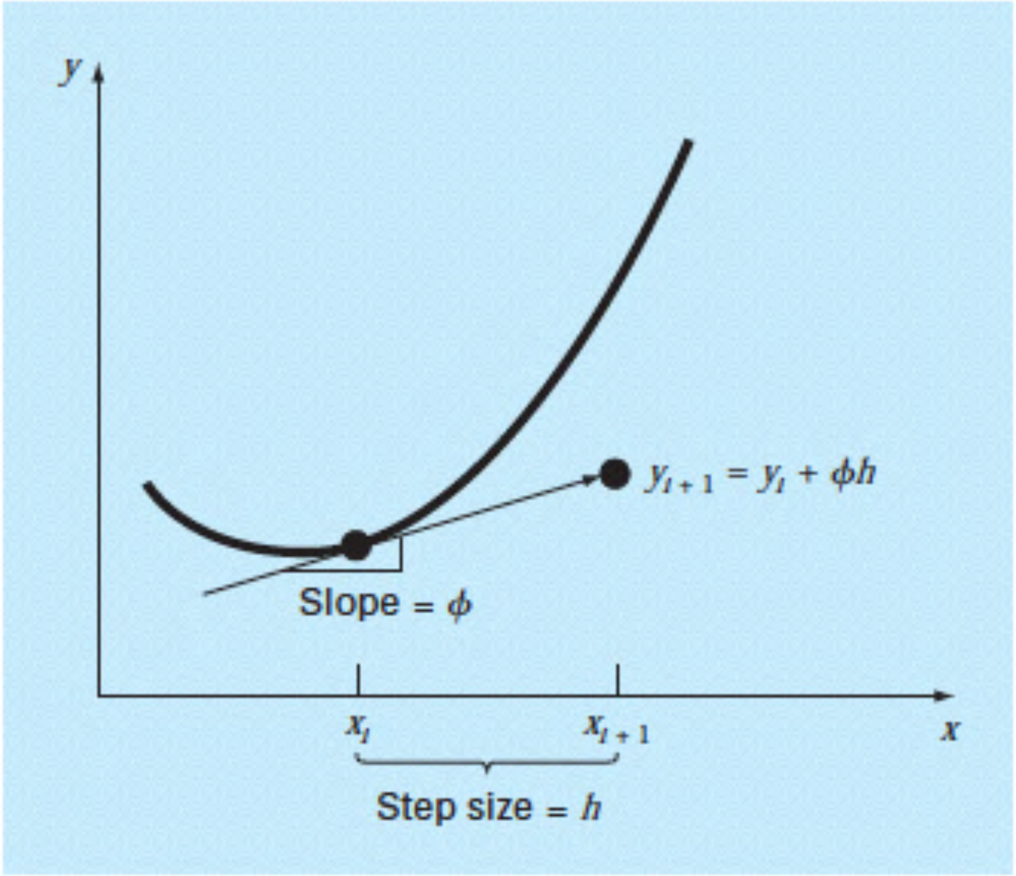
They will be recast in the form $\quad H_i\big(y_2(0), y_1(0)\big) = 0; \qquad i = 1,2$

ORDINARY DIFFERENTIAL METHOD – INITIAL VALUE PROBLEM

$$\frac{dy}{dx} = f(x, y)$$

$$y(0) = \alpha_0$$

$$\boxed{y_{i+1} = y_i + \phi h}$$



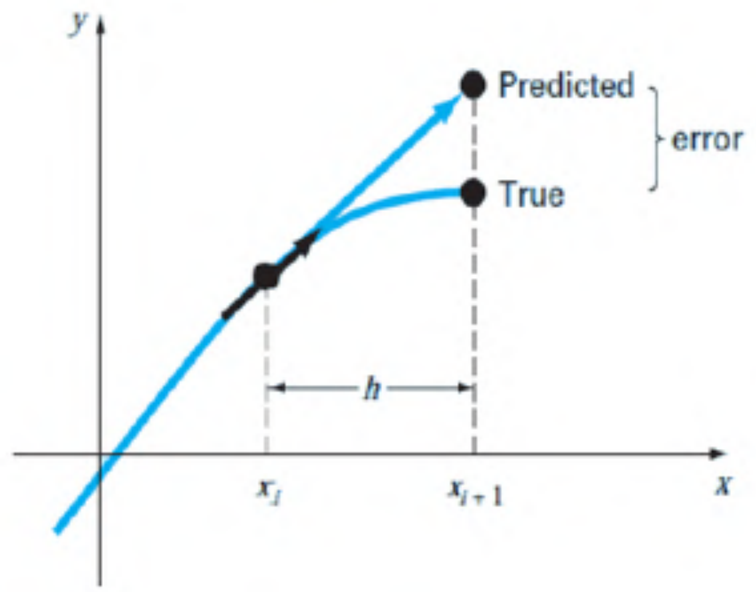
Euler Explicit Method

$$\frac{dy}{dx} = f(x, y)$$

$$y(0) = \alpha_0$$

The first derivative provides a direct estimate of the slope at x_i

- $\phi = f(x_i, y_i)$
- where $f(x_i, y_i)$ is the differential equation evaluated at x_i *and* y_i .



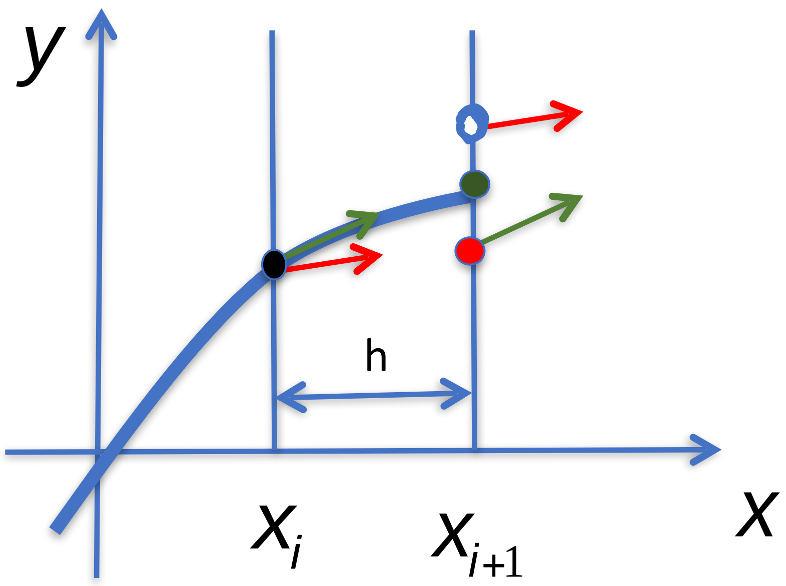
$$\boxed{y_{i+1} = y_i + hf(x_i, y_i)}$$

Euler Implicit Method

$$\frac{dy}{dx} = f(x, y)$$

$$y(0) = \alpha_0$$

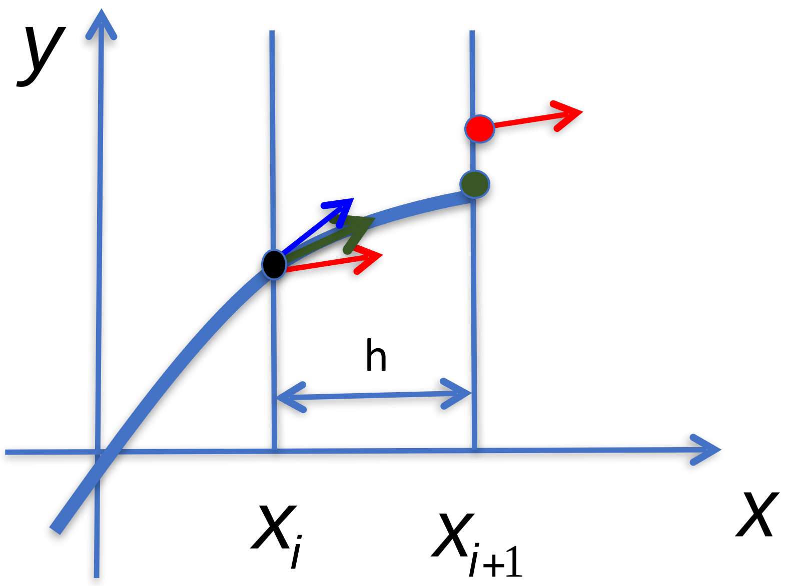
$$\boxed{y_{i+1} = y_i + hf\left(x_{i+1}, y_{i+1}\right)}$$



Semi Implicit (Crank Nicholson) Method

$$\frac{dy}{dx} = f(x, y)$$
$$y(0) = \alpha_0$$

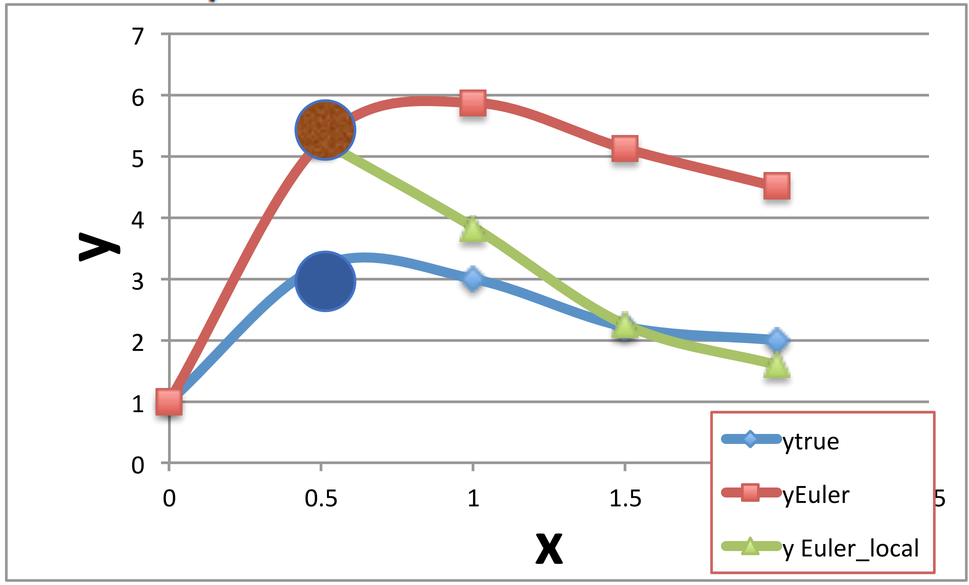
$$\boxed{y_{i+1} = y_i + \frac{h}{2}\left[f\left(x_i, y_i\right) + f\left(x_{i+1}, y_{i+1}\right)\right]}$$



Example: Use Euler's method to numerically integrate

$$\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$$

from x = 0 to x = 2 with a step size of 0.5. The initial condition at x = 0 is y = 1.



Solution

Exact solution: $y = -0.5x^4 + 4x^3 - 10x^2 + 8.5x + 1;$

Explicit Euler:

- $y_{i+1} = y_i + f(x_i, y_i)h$

$$y(0.5) = y(0) + f(0, 1)0.5$$

$$y(0.5) = 1.0 + 8.5(0.5) = 5.25$$

True Solution @ x=0.5

$$y = -0.5(0.5)^4 + 4(0.5)^3 - 10(0.5)^2 + 8.5(0.5) + 1 = 3.21875$$

Error@ x=0.5 $\quad E_t = \text{true} - \text{approximate} = 3.21875 - 5.25 = -2.03125 \qquad \varepsilon_t = -63.1\%$

Example: Use Euler's method to numerically integrate

Second Step: True Solution @ x=1.0 is 3.0

Explicit Euler:

- $y_{i+1} = y_i + f(x_i, y_i)h$

$$y(1) = y(0.5) + f(0.5, 5.25)0.5$$
$$= 5.25 + [-2(0.5)^3 + 12(0.5)^2 - 20(0.5) + 8.5]0.5$$
$$= 5.875$$

$$y(1) = y_{true}(0.5) + hf(0.5, y_{true}(0.5))$$
$$= 3.21875 + 0.5 \times f(0.5, 3.211875)$$
$$= 3.21875 + 0.5 \times (1.25)$$
$$= 3.84375$$

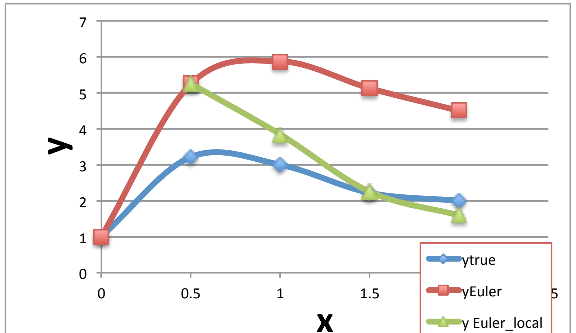
Global Error@ x=1.0

Local Error@ x=1.0

$E_t = \text{true} - \text{approximate} =$ 5.875-3=2.875

$E_t = \text{true} - \text{approximate} =$ 3.84375-3=0.84375

$$\boxed{\varepsilon_t = 2.875 \times 100\% / 3 = 95.8\%}$$



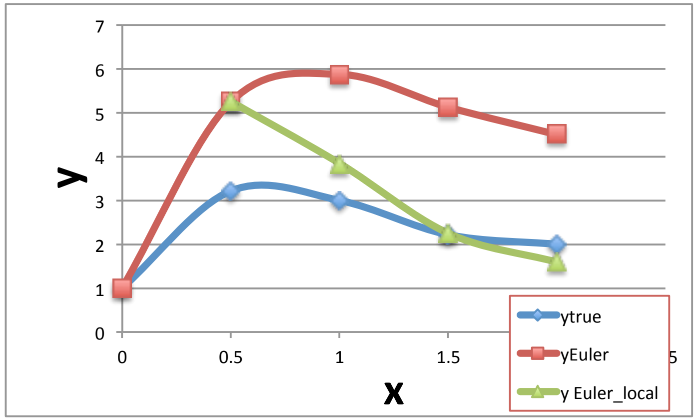
$$\boxed{\varepsilon_t = 0.84375 \times 100\% / 3 = 95.8\% = 28.125\%}$$

$$\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$$

Solution

Exact solution: $y = -0.5x^4 + 4x^3 - 10x^2 + 8.5x + 1;$

$$\boxed{y_{i+1} = y_i + hf\left(x_i, y_i\right)}$$



h	0.5					
x	f(x) =dy/dx	ytrue	yEuler	y Euler_local	Global_Error	Local_error
0	8.5	1	1			
0.5	1.25	3.21875	5.25	5.25	-63.10679612	-63.10679612
1	-1.5	3	5.87	3.84375	-95.83333333	-28.125
1.5	-1.25	2.21875	5.125	2.25	-130.9859155	-1.408450704
2	0.5	2	4.5	1.59375	-125	20.3125

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Ordinary Differential Equations

Runge Kutta Methods, Euler Method

Prof. Jayati Sarkar

Error Analysis of Euler Method

Round off Errors:

caused by the limited numbers of significant digits that can be retained by a computer.

Truncation /Discretization Errors:

caused by the nature of the techniques employed to approximate values of y

- ❖ Local Truncation Error
- ❖ Propagated Truncation Error

The sum of the two is the **Global Truncation Error**

$\boxed{\text{➢ } y' = f(x, y)}$

Taylor series expansion about a starting value (x_i, y_i)

$$\boxed{y_{i+1} = y_i + y_i' h + \frac{y_i''}{2!} h^2 + \cdots + \frac{y_i^n}{n!} h^n + R_n}$$

$h = x_{i+1} - x_i$ and R_n = the remainder term

Error Analysis of Euler Method

$$R_n = \frac{y^{(n+1)}(\xi)}{(n+1)!} h^{n+1}$$ where ξ lies in the interval from x_i to x_{i+1}

$$y_{i+1} = y_i + f(x_i, y_i)h + \boxed{\frac{f'(x_i, y_i)}{2!} h^2 + \cdots + \frac{f^{n-1}(x_i, y_i)}{n!} h^n + O(h^{n+1})}$$

True local truncation error $E_t = \frac{f'(x_i, y_i)}{2!} h^2 + \cdots + O(h^{n+1})$

Approximate local truncation error $E_a = \frac{f'(x_i, y_i)}{2!} h^2$, or $E_a = O(h^2)$

Global Truncation Error ~ *O(h)*

Error Analysis of Euler Method

$$\frac{dy}{dx} = f(x_i, y_i) = -2x^3 + 12x^2 - 20x + 8.5$$

$$y_{i+1} = y_i + f(x_i, y_i)h$$

Error@ x=0.5

$$E_t = \text{true} - \text{approximate} = 3.21875 - 5.25 = -2.03125$$

- $E_t = \frac{f'(x_i,y_i)}{2!}h^2 + \frac{f''(x_i,y_i)}{3!}h^3 + \frac{f^3(x_i,y_i)}{4!}h^4$

$$f'(x_i, y_i) = -6x^2 + 24x - 20$$

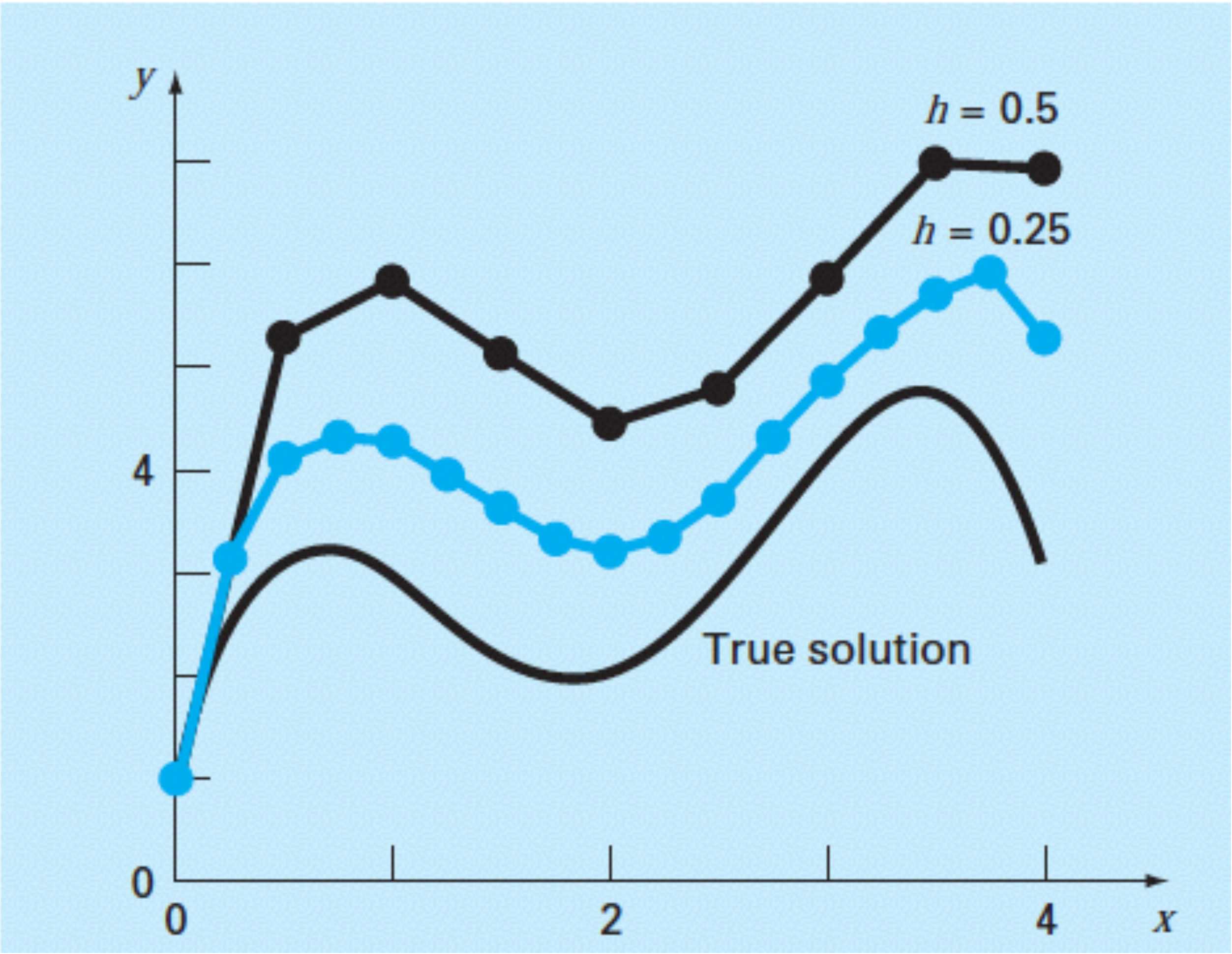
$$f''(x_i, y_i) = -12x + 24$$

$$f^3(x_i, y_i) = -12$$

- $E_{t,2} = \frac{-6(0.0)^2 + 24(0.0) - 20}{2}(0.5)^2 = -2.5$
- $E_{t,3} = \frac{-12(0.0) + 24(0.0)}{6}0.5^3 = 0.5$
- $E_{t,4} = \frac{-12}{24}0.5^4 = -0.03125$

- Total truncation error: $E_t = E_{t,2} + E_{t,3} + E_{t,4} = -2.5 + 0.5 - 0.03125 = -2.03125$

- $E_{t,2} > E_{t,3} > E_{t,4}$



- Observations:
  - The error decreases as h decreases.
  - The method will provide error-free predictions if the function is linear, because for a straight line the second derivative would be zero.

Higher-Order Taylor Series Methods

$$y_{i+1} = y_i + f(x_i, y_i)h + \frac{f'(x_i, y_i)}{2!}h^2$$

$$E_a = \frac{f''(x_i, y_i)}{3!}h^3$$

The incorporation of higher-order terms is simple enough to implement for polynomials

not so trivial when the ODE is more complicated.

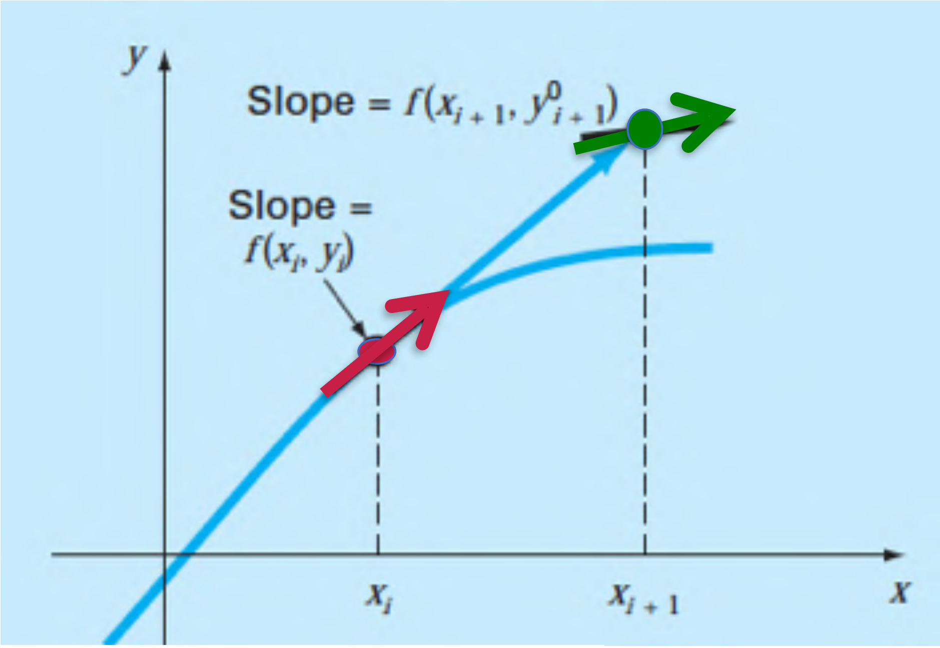
$$f'(x_i, y_i) = \frac{\partial f(x, y)}{\partial x} + \frac{\partial f(x, y)}{\partial y}\frac{dy}{dx}$$

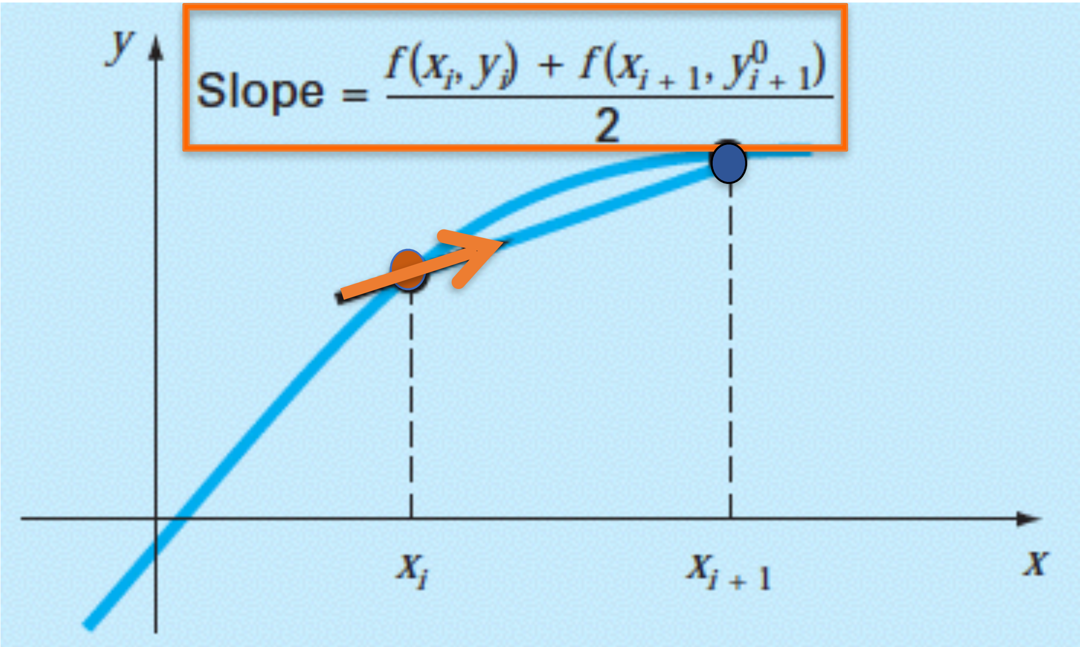
$$f''(x_i, y_i) = \frac{\partial[\frac{\partial f}{\partial x} + (\frac{\partial f}{\partial y})(\frac{dy}{dx})]}{\partial x} + \frac{\partial[\frac{\partial f}{\partial x} + (\frac{\partial f}{\partial y})(\frac{dy}{dx})]}{\partial y}\frac{dy}{dx}$$

Higher-order derivatives become increasingly more complicated.

Alternative one-step methods have been developed. These schemes are comparable in performance to the higher-order Taylor-series approaches but require only the calculation of first derivatives.

Heun's Method





- $\dfrac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$
- h=0.5

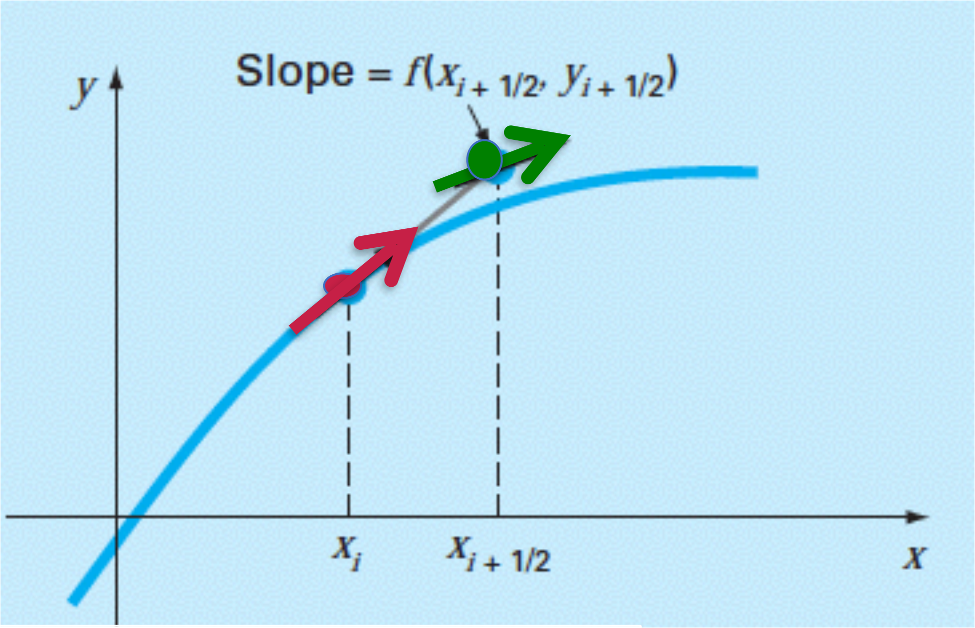
$$y_{Heun}(i+1) = y(i) + hs[y(i), x(i)]$$
$$y_{Heun}(i+1) = y(i) + h[w_1 k_1 + w_2 k_2]$$

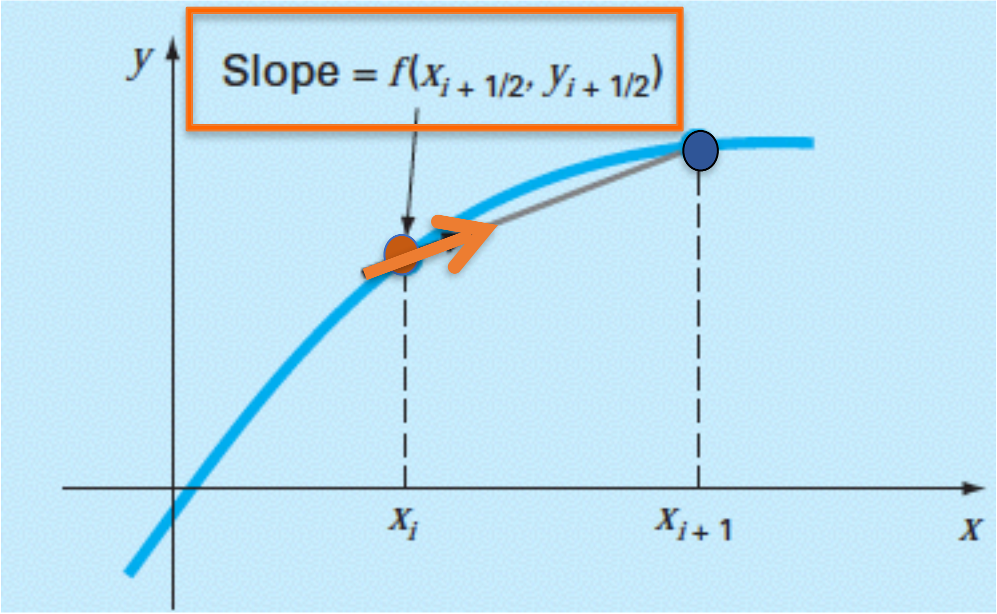
- w_1=0.5, $w_2 = 0.5$

$$k_1 = f[y(i), x(i)]$$
$$k_2 = f[y(i) + hk_1, x(i) + h]$$

x	f(x)	ytrue	k1	k2	S	yH	Error
0	8.50	1.00				1.00	
0.5	1.25	3.22	8.50	1.25	4.88	3.44	-6.80
1	-1.50	3.00	1.25	-1.50	-0.13	3.38	-12.50
1.5	-1.25	2.22	-1.50	-1.25	-1.38	2.69	-21.13
2	0.50	2.00	-1.25	0.50	-0.38	2.50	-25.00

Mid Point Method





- $\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$
- h=0.5

$$y_{Mid}(i+1) = y(i) + hs[y(i), x(i)]$$

$$y_{Mid}(i+1) = y(i) + h[w_1k_1 + w_2k_2]$$

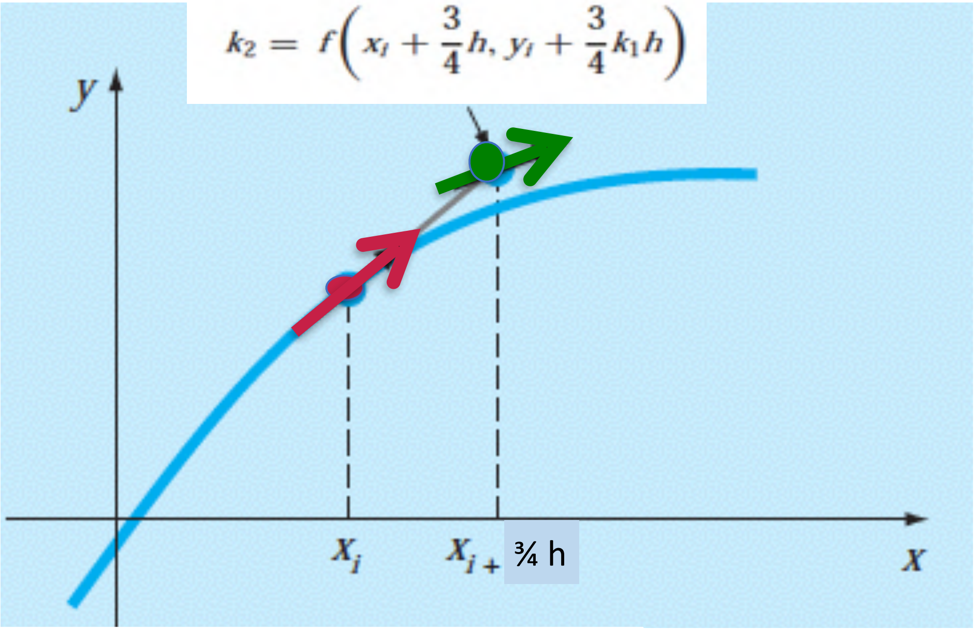
- $w_1=0, w_2 = 1$

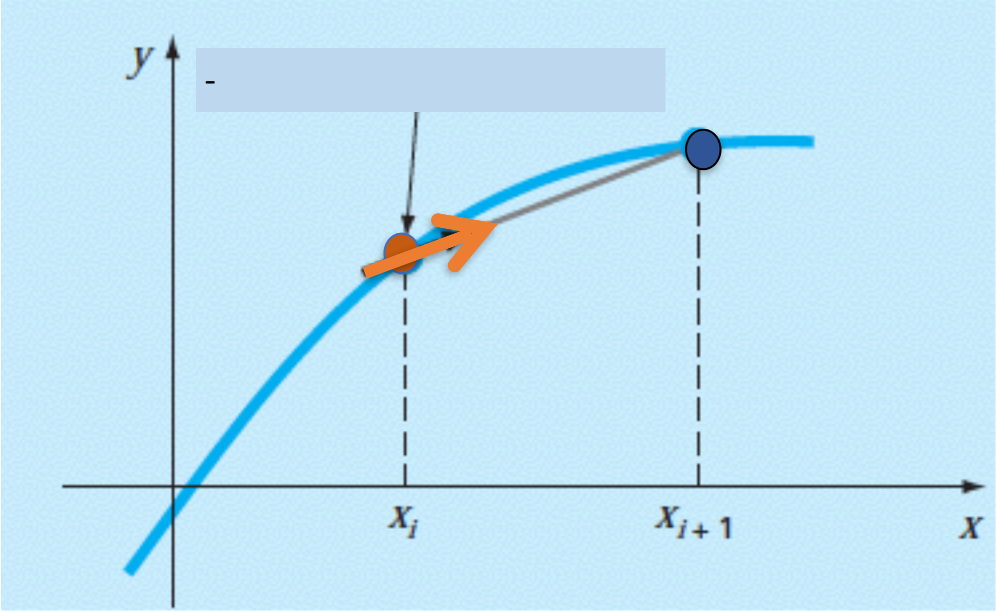
$$k_1 = f[y(i), x(i)]$$

$$k_2 = f[y(i) + 0.5hk_1, x(i) + 0.5h]$$

x	f(x	yt	k1	k2	S	yMi	Error
0	8.50	1.00				1.00	
0.5	1.25	3.22	8.50	4.22	4.22	3.11	3.40
1	-1.50	3.00	1.25	-0.59	-0.59	2.81	6.25
1.5	-1.25	2.22	-1.50	-1.66	-1.66	1.98	10.56
2	0.50	2.00	-1.25	-0.47	-0.47	1.75	12.50

Ralston Method





- $\dfrac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$
- h=0.5

$$y_R(i+1) = y(i) + hs[y(i), x(i)]$$

$$y_R(i+1) = y(i) + h[w_1 k_1 + w_2 k_2]$$

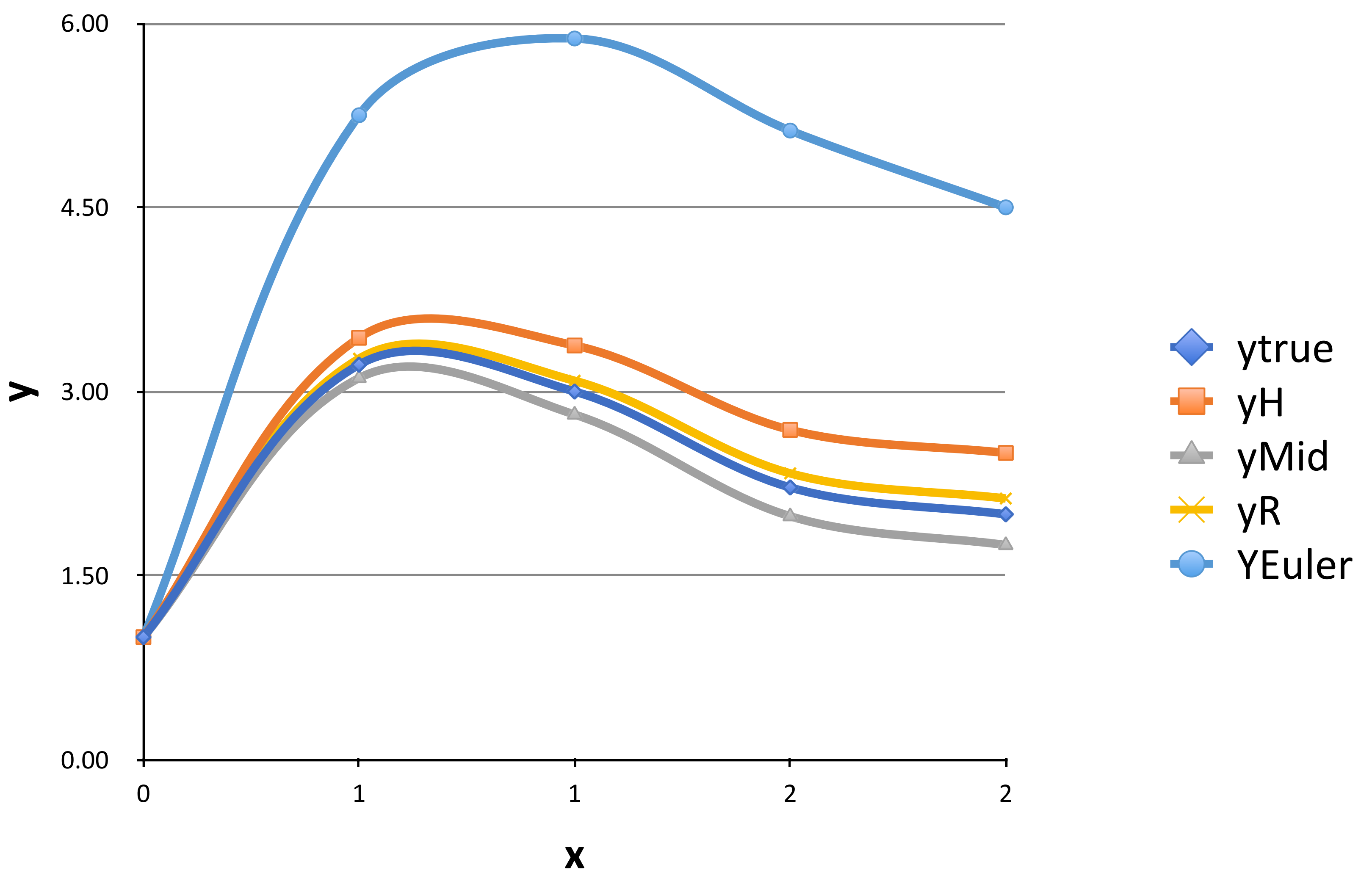
$$w_1 = 0.33, w_2 = 0.67$$

$$k_1 = f[y(i), x(i)]$$

$$k_2 = f[y(i) + 0.75hk_1, x(i) + 0.75h]$$

x	f(x)	yt	k	k2	S	yR	Err
0	8.50	1.00				1.00	
0.5	1.25	3.22	8.50	2.58	4.53	3.27	-1.51
1	-1.50	3.00	1.25	-1.15	-0.36	3.09	-2.92
1.5	-1.25	2.22	-1.50	-1.51	-1.51	2.33	-5.18
2	0.50	2.00	-1.25	0.00	-0.41	2.13	-6.44

Comparison



Taylor series expansion

- $y_{i+1} = y_i + hs(y_i, t_i)$
- $y_{i+1} = y_i + h(\frac{dy}{dt})_i + \frac{h^2}{2!}\left(\frac{d^2y}{dt^2}\right)_i + \frac{h^3}{3!}\left(\frac{d^3y}{dt^3}\right)_i + \cdots$
- $(\frac{dy}{dt})_i = f_i;$
- $\left(\frac{d^2y}{dt^2}\right)_i = \frac{d}{dt}\left(\frac{dy}{dt}\right) = \frac{d}{dt}[f(y,t)] = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial t} = f_t + f_y f$
- $\boxed{y_{i+1} = y_i + hf_i + \frac{h^2}{2!}[f_t + f_y f]_i + \theta(h^3)}$

For RK 2 Method

- $y_{i+1} = y_i + h[w_1 f(y_i, t_i)] + w_2 f(y_i + qhk_1, t_i + ph)]$
- $f(y_i + qhk_1, t_i + ph) = f(y_i, t_i) + qhk_1\frac{\partial f}{\partial y_i} + ph\frac{\partial f}{\partial t_i} + \theta(h^2)$

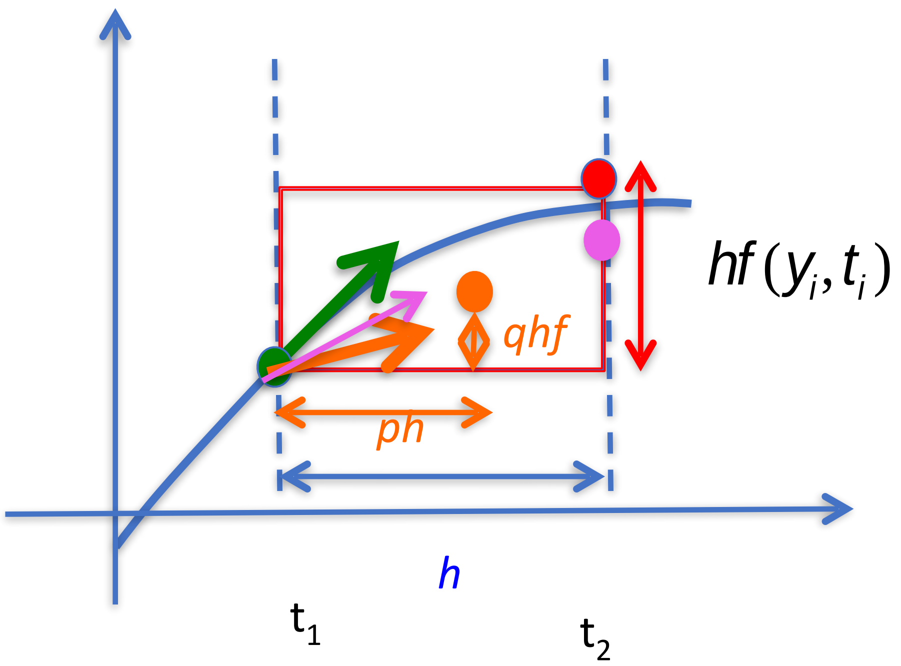
$$= f_i + qhk_1 f_{y_i} + phf_{t_i} + \theta(h^2)$$

$$y_{i+1} = y_i + h\left[w_1 f_i\right] + h\left[w_2 f_i + w_2 qhk_1 f_{y_i} + w_2 phf_{t_i} + O\left(h^2\right)\right]$$

- $\boxed{y_{i+1} = y_i + h(w_1 f_i + w_2 f_i) + h^2[w_2 p f_t + w_2 q f f_y]_i + \theta}$

RK 2 Method

- $w_1 + w_2 = 1$
- $w_2 p = \frac{1}{2}$
- $w_2 q = \frac{1}{2}$



HEUN METHOD

- $\boldsymbol{q} = \boldsymbol{p} = 1; w_2 = \frac{1}{2}; w_1 = \frac{1}{2}$

MIDPOINT METHOD

- $\boldsymbol{q} = \boldsymbol{p} = \frac{1}{2}; w_2 = 1; w_1 = 0$

RALSTON METHOD

- $\boldsymbol{q} = \boldsymbol{p} = \frac{3}{4}; w_2 = \frac{2}{3}; w_1 = \frac{1}{3}$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Ordinary Differential Equations

Runge Kutta Methods, Euler Method

Prof. Jayati Sarkar

RK-n Methods

$$y_{i+1} = y_i + h\sum_{m=1}^{n} w_m k_m$$

$$k_1 = f\left(y_i, t_i\right)$$

$$k_m = f\left(y_i + h\left[q_{m,1}k_1 + q_{m,2}k_2 + \ldots\ldots + q_{m,m-1}k_{m-1}\right], \left(t_i + p_m h\right)\right)$$

Accuracies

Eulers	RK-2	RK-3	RK-4	RK-5
O(h²)	***O(h³)***	***O(h⁴)***	***O(h⁵)***	***O(h⁵)***

RK-4 Methods

$$y_{i+1} = y_i + hS(y_i, t_i)$$

$$S(y_i, t_i) = w_1 k_1 + w_2 k_2 + w_3 k_3 + w_4 k_4$$

p_2	q_{21}			
p_3	q_{31}	q_{32}		
p_4	q_{41}	q_{42}	q_{43}	
	w_1	w_2	w_3	w_4

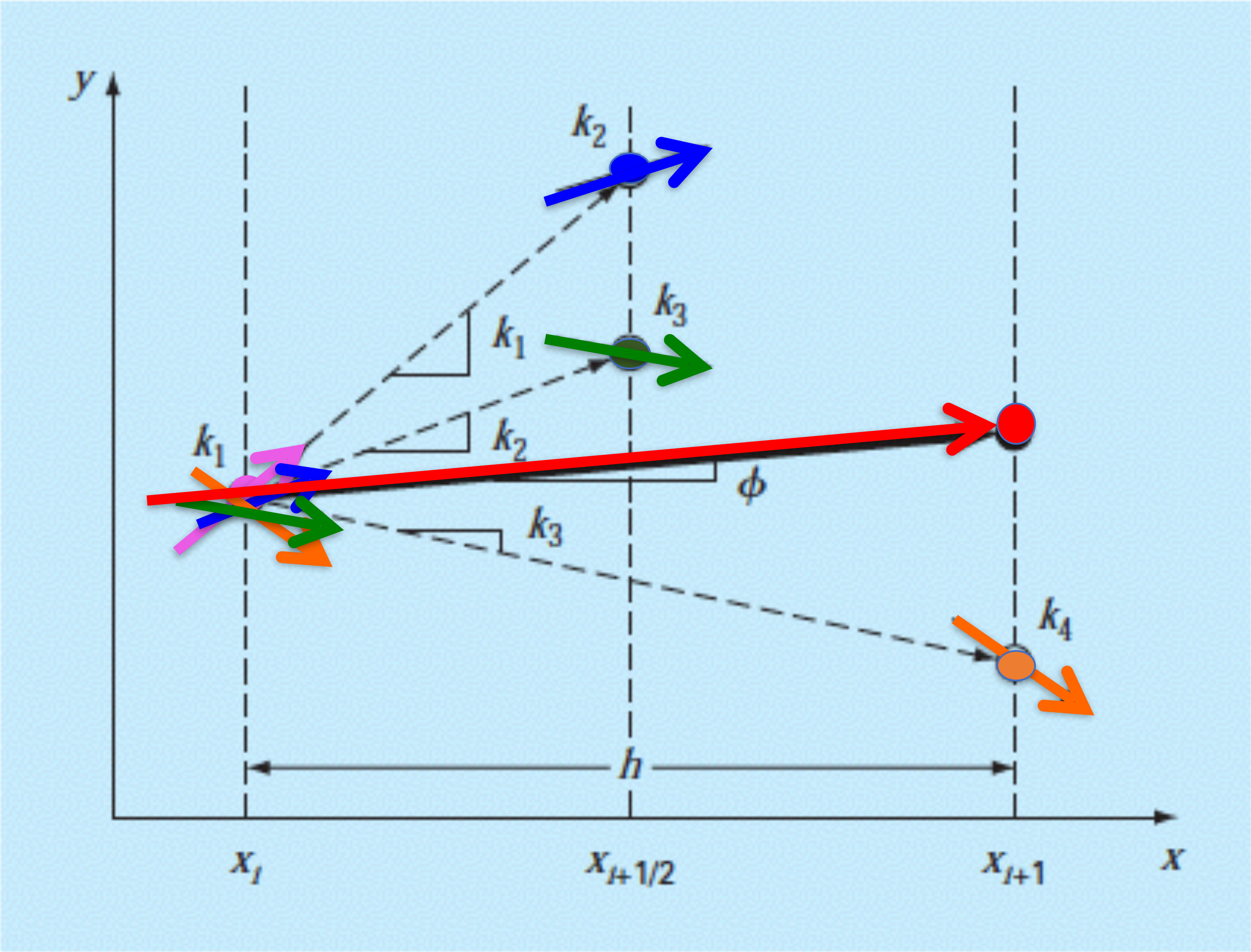
$$k_1 = f(y_i, t_i)$$

$$k_2 = f(y_i + hq_{2,1}k_1, (t_i + p_2 h))$$

$$k_3 = f(y_i + hq_{3,1}k_1 + hq_{3,2}k_2, (t_i + p_3 h))$$

$$k_4 = f(y_i + hq_{4,1}k_1 + hq_{4,2}k_2 + hq_{4,3}k_3, (t_i + p_4 h))$$

Classical RK-4 Methods



Popular RK-4 Methods

p_2	q_{21}			
p_3	q_{31}	q_{32}		
p_4	q_{41}	q_{42}	q_{43}	
	w_1	w_2	w_3	w_4

Classic RK-4

0.5	0.5			
0.5	0	0.5		
1	0	0	1	
	1/6	1/3	1/3	1/6

RK-Fehlberg Method

1/4	1/4			
3/8	3/32	9/32		
12/13	1932/2197	-7200/2197	7296/219	
	25/216	1408/2565		-1/5

RK-Gill

0.5	0.5			
0.5	(√2-1)/2	(2-√2)/2		
1	0	-1/√2	(2+√2)/2	
	1/6	(2-√2)/6	(2+√2)/6	1/6

Multi Step Method

Heun's Method

Predictor: Euler Method

$$k_1 = f\left(y_i, x_i\right)$$

$$y_{i+1}^{0} = y_i + hk_1 \qquad \boxed{+O(h^2)}$$

Corrector: Trapezoidal Rule

$$k_2^{0} = f\left(y_{i+1}^{0}, x_{i+1}\right)$$

$$\bar{y}_{i+1}^{1} = y_i + \frac{h}{2}\left(k_1 + k_2^{0}\right)$$

Corrector repeated multiple times

$$k_2^{m} = f\left(\bar{y}_{i+1}^{m}, x_{i+1}\right)$$

$$\bar{y}_{i+1}^{m+1} = y_i + \frac{h}{2}\left(k_1 + k_2^{m}\right) \quad m = 1 \text{ to } M$$

$$\boxed{\bar{y}_{i+1}^{m+1} = \bar{y}_{i+1}^{m} = \bar{y}_i^{m}}$$

$$\bar{y}_{i+1} = y_i + \frac{h}{2}\left[f\left(y_i, t_i\right) + f\left(\bar{y}_{i+1}, t_{i+1}\right)\right]$$

CRANK NICHOLSON (SEMI-IMPLICIT METHOD) $\boxed{+O(h^3)}$

Improved Multi Step Method

non-starting Heun's Method

Predictor: $$y_{i+1}^{0} = y_{i-1}^{m} + f(x_i, y_i^m)\,2h \quad \boxed{+O(h^3)}$$

Corrector: $$y_{i+1}^{j} = y_i^m + \frac{f(x_i, y_i^m) + f(x_{i+1}, y_{i+1}^{j-1})}{2}h$$

$$\boxed{+O(h^3)}$$

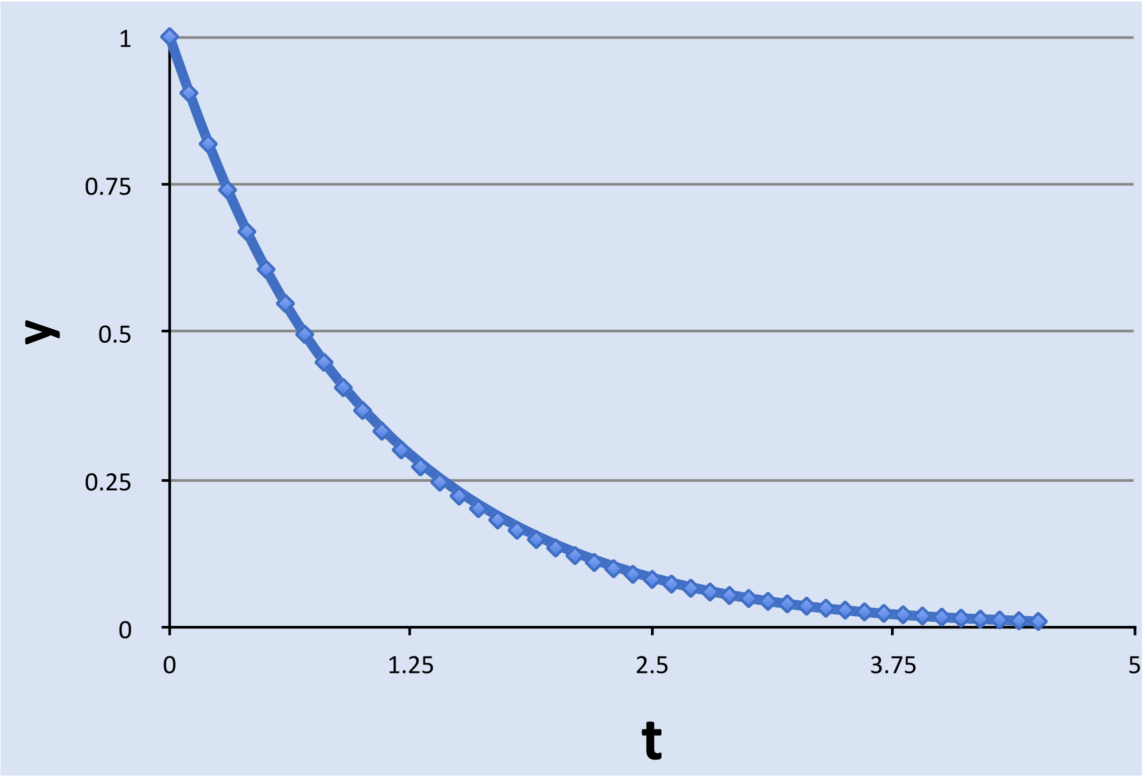
$$(\text{for } j = 1, 2, \ldots, m)$$

use original Heun's Method at i=0

Stability of Euler Method

$$\frac{dy}{dt} = -\lambda y \; with \; t = 0, y = 1 \qquad (\lambda \; is + ve)$$

Analytical solution $y = e^{-\lambda t}$



Stability of Euler Method

$$\frac{dy}{dt} = -\lambda y \; with \; t = 0, y = 1 \quad (\lambda \; is + ve)$$

Analytical solution $y = e^{-\lambda t}$

$$y = y_e + \varepsilon \qquad y_e = exact\ solution$$

$$\frac{d(y_e+\varepsilon)}{dt} = -\lambda(y_e+\varepsilon)$$

$$\frac{d(y_e)}{dt} + \lambda y_e = -[\frac{d\varepsilon}{dt} + \lambda\varepsilon]$$

$$\boxed{\frac{d\varepsilon}{dt} = -\lambda\varepsilon}$$

Error also satisfies original equation

$$\varepsilon_{N+1} - \varepsilon_N = -h\lambda\varepsilon_N$$

Euler Method

A= $\frac{\varepsilon_{N+1}}{\varepsilon_N} = (1 - h\lambda)$

To enforce the stability we want the error at t_{n+1} to be smaller than t_n

$$\left|\frac{\varepsilon_{N+1}}{\varepsilon_N}\right| \leq 1$$

$$|(1 - h\lambda)| \leq 1$$

$0 \leq h\lambda \leq 2$ i.e step size h should be smaller than $2/\lambda$ to ensure stability

Stability Envelope (A)

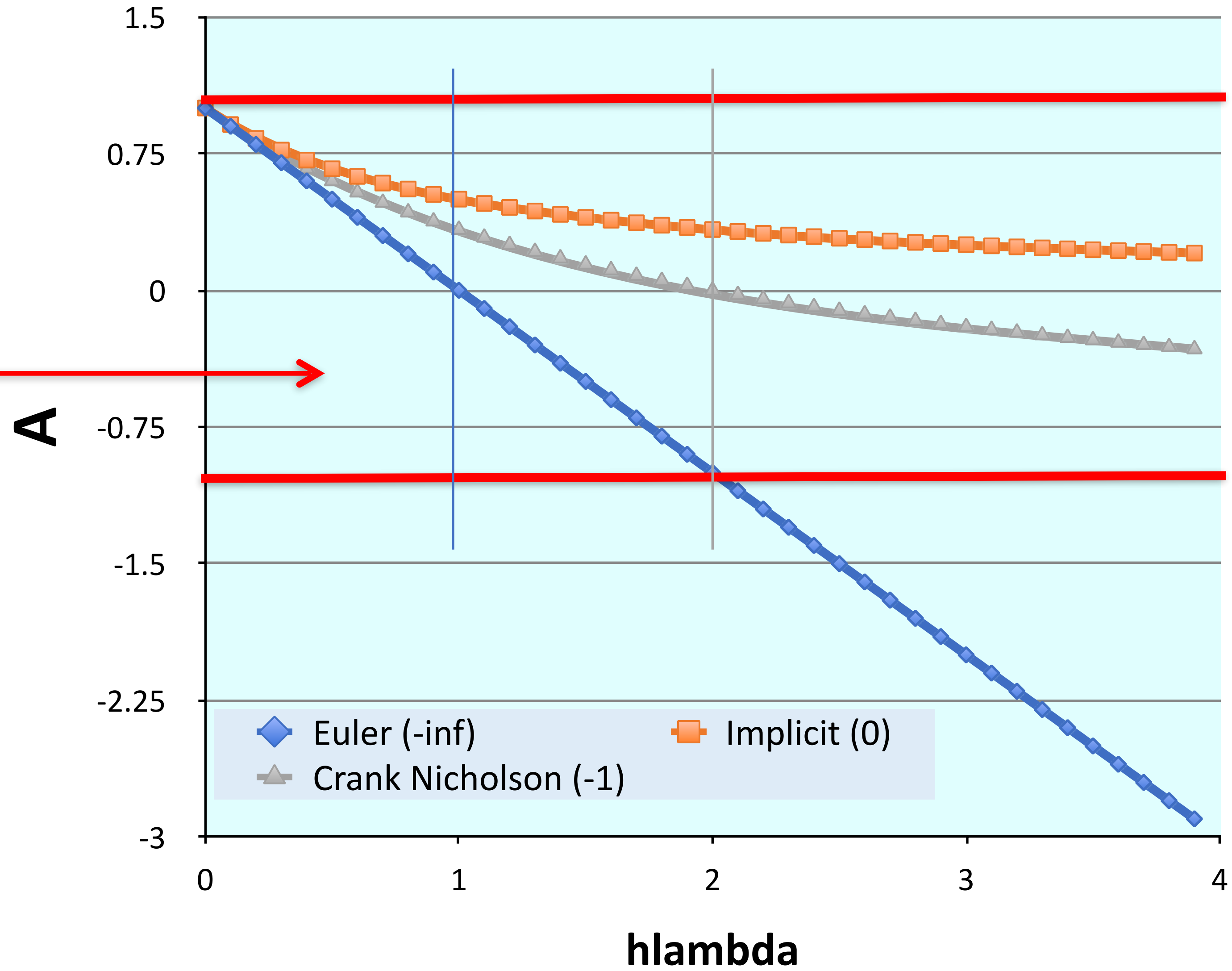
- i) $A_{Euler} = 1 - h\lambda$

  ii) $A_{Imp} = \frac{1}{1+h\lambda}$

  iii) $A_{CN} = \frac{1-\frac{h\lambda}{2}}{1+\frac{h\lambda}{2}}$

Stability Envelope

A= $\dfrac{\varepsilon_{N+1}}{\varepsilon_N}$ =-ve



Globally Stable

Converges but oscillatory after this

Converges before this Oscillatory and diverges

- i) $A_{Euler} = 1 - h\lambda$
- ii) $A_{Imp} = \dfrac{1}{1+h\lambda}$
- iii) $A_{CN} = \dfrac{1-\frac{h\lambda}{2}}{1+\frac{h\lambda}{2}}$

Richardson Extrapolation

$$\bar{y}_{i+1} = y_{i+1} + c_1 \boxed{h^n} + c_2 h^{n+1} + \ldots\ldots\ldots\ldots (1)$$

same

$$\bar{y}_{i+\frac{1}{2}} = y_{i+\frac{1}{2}} + c_1 \left(\frac{h}{2}\right)^n + c_2 \left(\frac{h}{2}\right)^{n+1}$$

$$\bar{y}_{i+1} = y_{i+1}\left(\frac{h}{2}\right) + 2\, c_1 \left(\frac{h}{2}\right)^n + 2c_2 \left(\frac{h}{2}\right)^{n+1} \ldots\ldots\ldots\ldots (2)$$

$$\boxed{\Delta = y_{i+1} - y_{i+1}\left(\frac{h}{2}\right)}$$

$$2^{n-1} eq(2) - eq(1); gives$$

$$\bar{y}_{i+1} = y_{i+1}\left(\frac{h}{2}\right) + \frac{\Delta}{2^{n-1}-1} - \frac{c_2 h^{n+1}}{2(2^{n-1}-1)}$$

$$\boxed{\bar{y}_{i+1} = y_{i+1}\left(\frac{h}{2}\right) + \frac{\Delta}{2^{n-1}-1}} + \boxed{\theta(h^{n+1})}$$

For explicit Euler

- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{2^1 - 1} + \theta(h^2)$
- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{1} + \theta(h^2)$
- $\bar{y}_{i+1} = 2y_{i+1}(\frac{h}{2}) - y_{i+1}(h) + \theta(h^2)$

For RK-2 method

- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{2^2 - 1} + \theta(h^4)$
- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{3} + \theta(h^4)$

For RK-4 method

- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{2^4 - 1} + \theta(h^6)$
- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{15} + \theta(h^6)$

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Ordinary Differential Equations

Runge Kutta Methods, Euler Method

Prof. Jayati Sarkar

Richardson Extrapolation

$$\bar{y}_{i+1} = y_{i+1} + c_1 \boxed{h^n} + c_2 h^{n+1} + \dots\dots\dots\dots \ (1)$$

same

$$\bar{y}_{i+\frac{1}{2}} = y_{i+\frac{1}{2}} + c_1 \left(\frac{h}{2}\right)^n + c_2 \left(\frac{h}{2}\right)^{n+1}$$

$$\Delta = y_{i+1} - y_{i+1}\left(\frac{h}{2}\right)$$

$$\bar{y}_{i+1} = y_{i+1}\left(\frac{h}{2}\right) + 2\, c_1 \left(\frac{h}{2}\right)^n + 2 c_2 \left(\frac{h}{2}\right)^{n+1} \ \dots\dots\dots\dots\dots (2)$$

$$2^{n-1} eq(2) - eq(1); gives$$

$$\bar{y}_{i+1} = y_{i+1}\left(\frac{h}{2}\right) + \frac{\Delta}{2^{n-1}-1} - \frac{c_2 h^{n+1}}{2(2^{n-1}-1)}$$

$$\bar{y}_{i+1} = y_{i+1}\left(\frac{h}{2}\right) + \frac{\Delta}{2^{n-1}-1} + \boxed{\theta(h^{n+1})}$$

For explicit Euler

- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{2^1 - 1} + \theta(h^3)$
- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{1} + \theta(h^3)$
- $\bar{y}_{i+1} = 2y_{i+1}(\frac{h}{2}) - y_{i+1}(h) + \theta(h^3)$

For RK-2 method

- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{2^2 - 1} + \theta(h^4)$
- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{3} + \theta(h^4)$

For RK-4 method

- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{2^4 - 1} + \theta(h^6)$
- $\bar{y}_{i+1} = y_{i+1}(\frac{h}{2}) + \frac{\Delta}{15} + \theta(h^6)$

Concept of Adaptive Step-Size

$$\frac{dy}{dt} = -y^2$$

Step-Size? Value of λ?

$$f(y) = y^2$$

Linearize the Equation

$$f(y) = f(y_0) + \frac{\partial f}{\partial y}_{y_0} (y - y_0)$$

$$f(y) = {y_0}^2 + 2y_0(y - y_0)$$

$$f(y) = -{y_0}^2 + 2y_0 y$$

- $\frac{dy}{dt} = -2y_0 y + {y_0}^2$

$$\lambda = 2y_0$$

$$\lambda = 2y_i$$

Euler Stability Envelope

$$0 \leq h\,\lambda \leq 2$$

- $i = 0, y_i = 1$

$$\boxed{\lambda = 2; h < \frac{2}{\lambda} \text{ i.e. } h < 1}$$

$$h = 0.2$$

$$y_i = y_0 + h\left(-{y_0}^2\right)$$

$$y_1 = 1 + 0.2(-1) = 0.8$$

$$\lambda = 2y_1 = 1.6$$

$$\boxed{h < \frac{2}{1.6} < 1.25}$$

Maximum allowable step size increases as iteration increases.

Adaptive Step-Size

$$\bar{y}_{i+1} = y_{i+1}(h) + c_1 h^n$$

$$\bar{y}_{i+1} = y_{i+1}\left(\frac{h}{2}\right) + 2c_1\left(\frac{h}{2}\right)^n$$

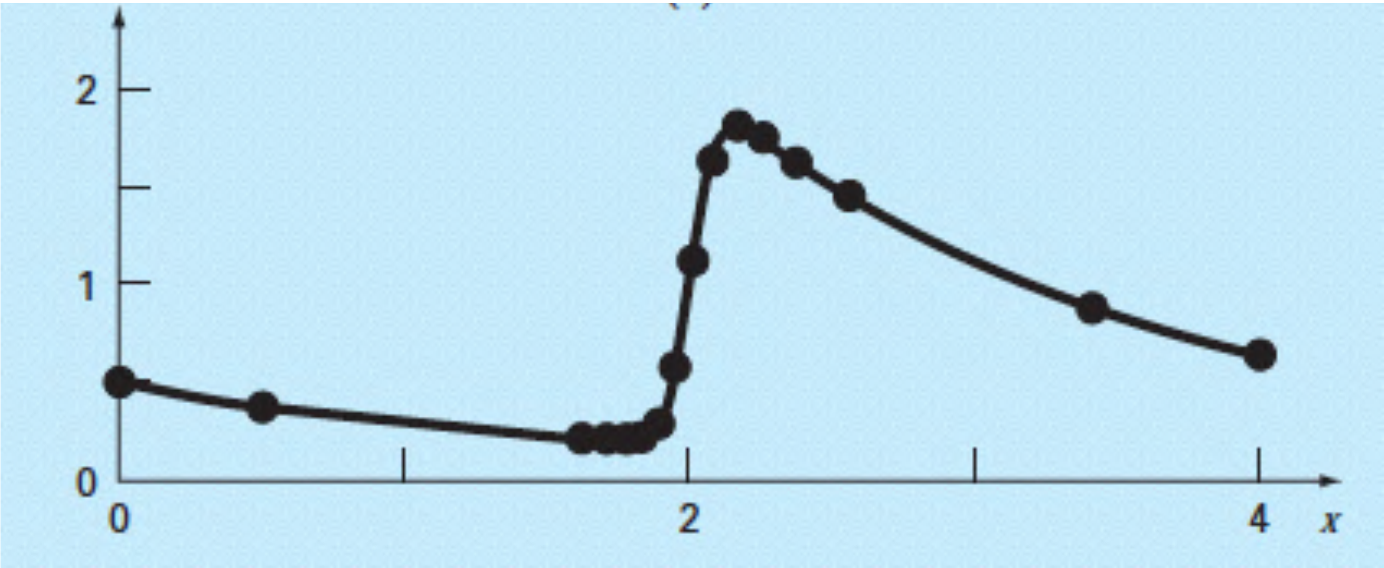
Subtracting both equations

$$0 = -\Delta + h^n[c_1 - 2^{1-n}c_1]$$

$$\Delta = ch^n$$

$$\Delta_{new} = ch_{new}^n$$

$$h_{new} = \left(\frac{\Delta_{new}}{\Delta}\right)^{\frac{1}{n}} h$$



If Δnew~εtol

$$\boxed{h_{new} = \left(\frac{\varepsilon_{tol}}{\Delta}\right)^{1/n} h}$$

If εtol~10^{-2}, n=2

Δ1=0.0625

$$\boxed{h_{new} = \left(\frac{0.01}{0.0625}\right)^{0.5} h = 0.4h}$$

Δ2=0.00625

$$\boxed{h_{new} = \left(\frac{0.01}{0.00625}\right)^{0.5} h = 1.265h}$$

MULTI-STEP METHODS

- 1. ADAM'S FAMILY OF METHODS
- 2. BACKWARD DIFFERENCE FORMULA

$$[y_0, y_1 --- y_{i-1}, y_i]$$

$$[f_0, f_1 --- f_{i-1}, f_i]$$

- $\frac{dy}{dx} = f(y, t) \qquad y(t_0) = y_0$

$$y_{i+1} = a_0 f_{i+1} + a_1 f_i + \cdots a_n f_{i-n+1} + b_1 y_i + \cdots b_n y_{i-n+1}$$

$$\boxed{y_{i+1} = a_0 f_{i+1} + \sum_{j=1}^{n} a_j f_{i-j+1} + \sum_{j=1}^{n} b_j f_{i-j+1}}$$

ADAM –BASHFORTH METHOD

nth order

$$y_{i+1} = a_1 f_i + \cdots a_n f_{i-n+1} + b_1 y_i$$

$\boxed{a_0 = 0}$ Explicit

ADAM MOULTON METHOD

nth order

$$y_{i+1} = a_0 f_{i+1} + a_1 f_i + \cdots a_n f_{i-n+1} + b_1 y_i$$

$\boxed{a_0 \neq 0}$ Implicit

Backward Difference Formula

$$y_{i+1} = a_0 f_{i+1} + b_1 y_i + \cdots b_n y_{i-n+1}$$

$$\frac{dy}{dt}_{i+1} = f(y_{i+1}, t_{i+1})$$

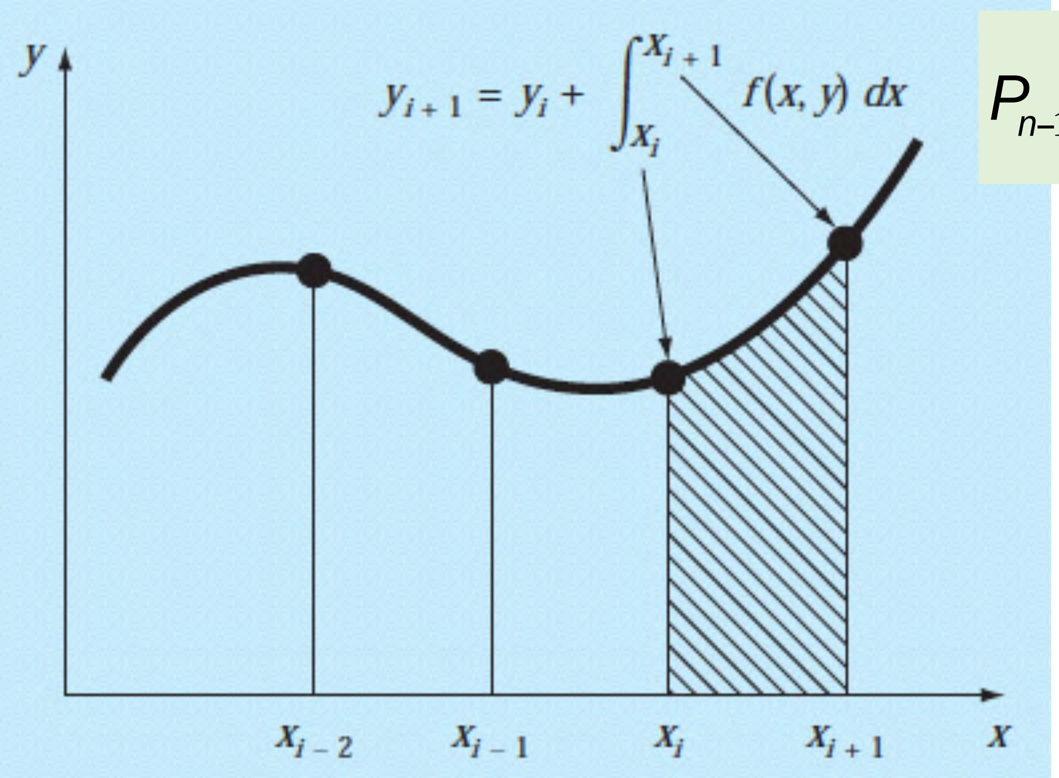
Replace $\frac{dy}{dt}_{i+1}$ with a approximate numerical derivative

$$\frac{y_{i+1} - y_i}{h} = f_{i+1} \text{ i.e } y_{i+1} = y_i + h f_{i+1}$$

Implicit Euler $\boxed{\theta(h)}$

$$\frac{3y_{i+1} - 4y_i + y_{i-1}}{2h} = f_{i+1}$$

$\boxed{\text{2nd order BDF}}$ $\boxed{\theta(h^2)}$



$$P_{n-1}(i) = f_i + \alpha\nabla f_i + \frac{\alpha(\alpha+1)}{2!}\nabla^2 f_i + \dots + \frac{\alpha(\alpha+1)..(\alpha+n-2)}{2!}\nabla^{n-1} f$$

$$\alpha = \frac{t - t_i}{h} \qquad dt = h\,d\alpha$$

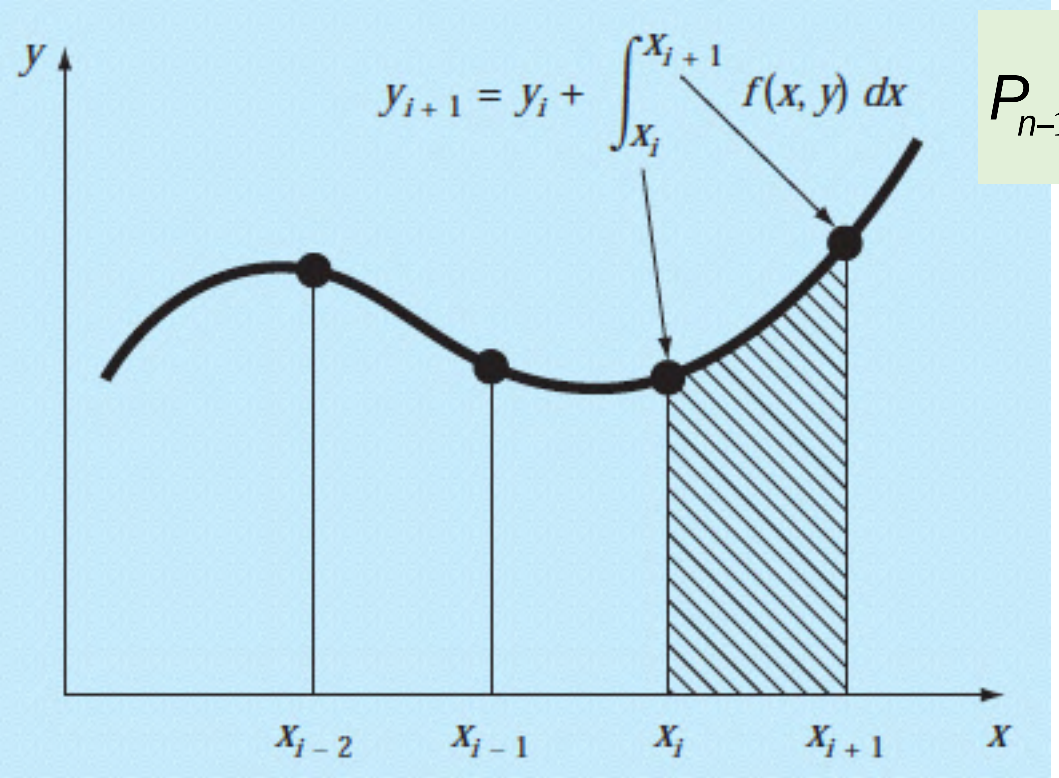
$$\boxed{y_{i+1} - y_i = \int_{t_i}^{t_{i+1}} P_{n-1}(i)\,dt = \int_0^1 P_{n-1}(i)\,h\,d\alpha = h\int_0^1 P_{n-1}(i)\,d\alpha}$$

$$y_{i+1} = y_i + h\int_0^1 P_{n-1}(i)\,d\alpha$$

$O(h^n)$

LTE:::$O(h^{n+1})$

GTE:::$O(h^n)$



$$P_{n-1}(i) = f_i + \alpha\nabla f_i + \frac{\alpha(\alpha+1)}{2!}\nabla^2 f_i + \ldots + \frac{\alpha(\alpha+1)..(\alpha+n-2)}{2!}\nabla^{n-1} f$$

$$\alpha = \frac{t - t_i}{h} \qquad dt = h d\alpha$$

$$\boxed{y_{i+1} - y_i = \int_{t_i}^{t_{i+1}} P_{n-1}(i)dt = \int_0^1 P_{n-1}(i)hd\alpha = h\int_0^1 P_{n-1}(i)d\alpha}$$

❖ n=1

$$\boxed{y_{i+1} = y_i + h\int_0^1 P_0(i)d\alpha = y_i + h\int_0^1 f_i d\alpha = y_i + hf_i}$$

LTE:::O(h^2) GTE:::O(h^1)

Adams Bashforth 1st Order Method

❖ n=2

$$y_{i+1} = y_i + h\int_0^1 P_1(i)d\alpha = y_i + h\int_0^1 \left[f(i) + \alpha\nabla f_i\right]d\alpha = y_i + h\left[\alpha f_i + \frac{\alpha^2}{2}(f_i - f_{i-1})\right]\Bigg|_0^1 = h\left[\frac{3}{2}f_i - \frac{1}{2}f_{i-1}\right]$$

LTE:::O(h^3) GTE:::O(h^2) Adams Bashforth 2ndOrder Method

❖ n=3

$$y_{i+1} = y_i + h\int_0^1 P_2(i)d\alpha = y_i + h\int_0^1 \left[f(i) + \alpha\nabla f_i + \frac{\alpha(\alpha+1)}{2!}\nabla^2 f\right]d\alpha = y_i + h\left[\frac{23}{12}f_i - \frac{4}{3}f_{i-1} + \frac{5}{12}f_{i-2}\right]$$

LTE:::O(h^4) GTE:::O(h^3) Adams Bashforth 3rd Order Method

➢ STARTING A PROBLEM

- $\frac{dy}{dt} = f \qquad\qquad y_0 @ t_0$
- $y_{i+1} = y_i + a_1 f_i + a_2 f_{i-1} + a_3 f_{i-2}$

$$@i = 0 \quad y_1 = y_0 + a_1 f_0 + \boxed{a_2 f_{-1} + a_3 f_{-2}}$$

$$@i = 1 \quad y_2 = y_1 + a_1 f_1 + a_2 f_0 + \boxed{a_3 f_{-1}}$$

$$@i = 2 \quad y_3 = y_2 + a_1 f_2 + a_2 f_1 + a_3 f_0$$

➢ NON SELF STARTING PROBLEMS

1. Use Euler to compute $y_1 \rightarrow \theta(h)$
2. Use AB-2 to compute $y_2 \rightarrow \theta(h^2)$
3. Use AB-3 and so on....................

➢ OR USE RK METHOD

1. RK-3 for y_1 and y_2
2. AB-3 for y_3 and y_4

and so on....................

➢ADAM-BASHFORTH-MOULTON METHOD

It's a predictor corrector method

1. AB n method to obtain $y_{i+1}^{0} \rightarrow Predictor$
2. AM n method to obtain $\tilde{y}_{i+1}^{1}, \tilde{y}_{i+1}^{2}, \tilde{y}_{i+1}^{3} \ldots \ldots \tilde{y}_{i+1}^{m}$ →Corrector

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Ordinary Differential Equations

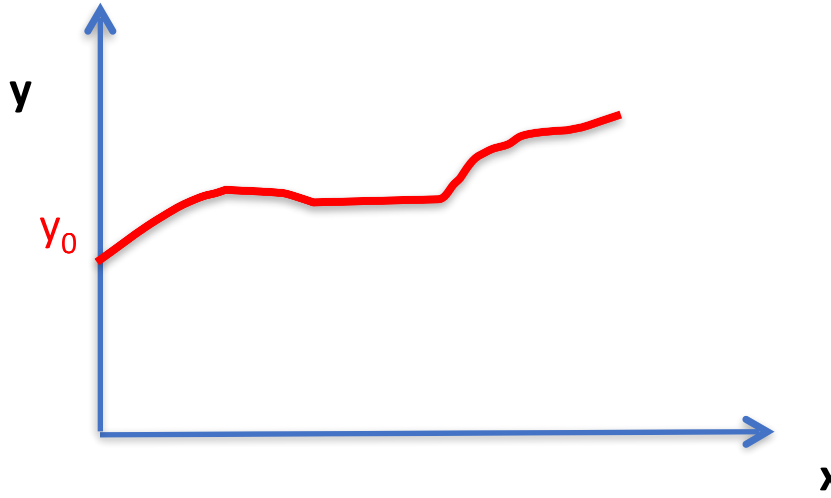
Boundary Value Problem

Shooting method, Finite Difference Method

Prof. Jayati Sarkar

Ordinary Differential Equation- Boundary value problem

ODE-IVP : Auxiliary condition at 1 point



ODE-BVP : Auxiliary condition at 2 points

Plug flow reactor

Model: $\left(\frac{F}{A}\right)\frac{dC_A}{dx} = -r(C_A)$

Initial condition: $C_A(x = 0) = C_0$

Axial Dispersion

Model $\left(\frac{F}{A}\right)\frac{dC_A}{dx} = D\frac{d^2C_A}{dx^2} - r(C_A)$

Boundary condition

$C_A(x = 0) = C_0$; $\boxed{C_A(x = L) = 0}$

2^{nd} order ODE requires two boundary conditions.

General boundary value problems

$\boxed{\text{Model } f(y'', y', y, x) = 0}$

BC's	$T_1(y', y, x)(at\ x_1) = 0$
	$T_2(y', y, x)(at\ x_2) = 0$

Model Forms of ODE in Chemical Engg.

$$\frac{d^2y}{dx^2} + p(x,y)\frac{dy}{dx} + q(x,y) = 0$$

Linear if $p(x,y)$= $p(x)$; $q(x,y) = r(x)y + s(x)$

if $s(x) = 0$ Homogeneous ODE

Boundary conditions:

1. Dirichlet: $y(x = x_1) = \alpha; y(x = x_2) = \beta$
2. Neumann $y'(x = x_1) = \alpha; y'(x = x_2) = \beta$
3. Mixed $y + c_1 y' = \alpha$

Shooting Method

$$\frac{d^2y}{dx^2} + p\frac{dy}{dx} + q = 0$$

$$\frac{dy}{dx} + c_1 y = \alpha \qquad @x = a$$

$$\frac{dy}{dx} + c_2 y = \beta \qquad @x = b$$

Convert ODE-BVP to ODE-IVP

Same model equation

- BC1. Remain same $@x = a$
- BC2. Gets replaced with another bc at the same location of x=a
- Then solve ODE-IVP

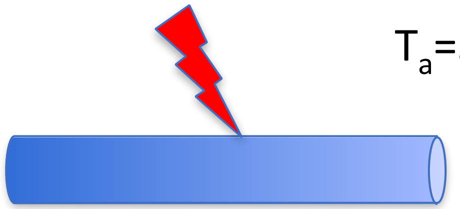
Assume:

$$\frac{dy_1}{dx} = y_2$$

$$\frac{dy_2}{dx} + py_2 + q = 0$$

$$\frac{dy_2}{dx} = -py_2 - q$$

$$\boxed{\frac{d}{dx}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} y_2 \\ -py_2 - q \end{bmatrix}}$$



$$\frac{d^2T}{dx^2} = 0.01\left(T - T_a\right)$$

$$y_1 = T$$

$$\frac{d}{dx}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} y_2 \\ 0.01(y_1 - 30) \end{bmatrix}$$

Actual BC:

$$y_1 = 30 @ x = 0$$

$$\boxed{y_1 = 80 @ x = L}$$

Modified BC:

$$y_1 = 30 @ x = 0$$

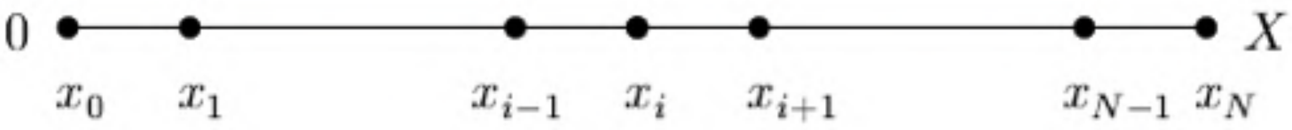
$$y_2 = 0 @ x = 0$$

ODE-BVP

$$-\frac{d^2u}{dx^2} = f \ in \ \Omega(0,1) \qquad \text{BC } u(0) = u(1) = 0$$

1D: $\Omega = (0, X)$, $\quad u_i \approx u(x_i)$, $\quad i = 0, 1, \ldots, N$

grid points $\quad x_i = i\Delta x \qquad$ mesh size $\quad \Delta x = \frac{X}{N}$



Central Difference Approximation

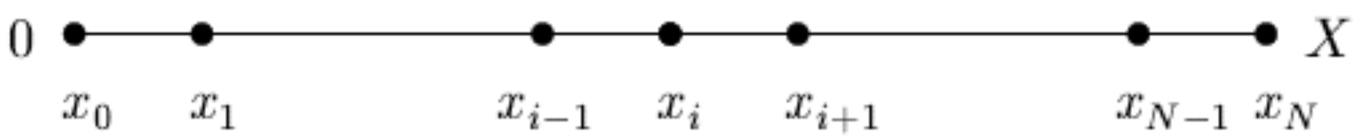
$$-\left(\frac{u_{i-1} - 2u_i + u_{i+1}}{\Delta x^2}\right) = f_i \qquad i = 1,2, \ldots \ldots \ldots \ldots, N-1$$

$u_0 = 0$; $u_N = 0$ *Homogeneous Dirichlet B.C* $\qquad \boxed{Au = F}$

$$A = \frac{1}{\Delta x^2}\begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & 1 & & \\ & & & & & \\ & & & & -1 & 2 \end{bmatrix}; \qquad u = \begin{bmatrix} u_1 \\ u_2 \\ \\ u_{N-1} \end{bmatrix}; \qquad \mathrm{F} = \begin{bmatrix} f_1 \\ f_2 \\ \\ f_{N-1} \end{bmatrix}$$

Neumann BC

$$\frac{\partial u}{\partial x}(1) = 0 \qquad u(0) = 0$$



$$-\left(\frac{u_{i-1} - 2u_i + u_{i+1}}{\Delta x^2}\right) = f_i$$

CDS

$$\frac{u_{N+1} - u_{N-1}}{\Delta x} = 0$$

$$-\left(\frac{u_{N-1} - 2u_N + u_{N+1}}{\Delta x^2}\right) = f_N$$

u_{N+1} gets eliminated

$$\frac{-u_{N-1} + u_N}{\Delta x^2} = \frac{f_N}{2}$$

$$A = \frac{1}{\Delta x^2}\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & 1 & \\ & & & & \\ & & & -1 & 1 \end{bmatrix} \qquad u = \begin{bmatrix} u_1 \\ u_2 \\ \\ u_N \end{bmatrix}; \qquad \mathrm{F} = \begin{bmatrix} f_1 \\ f_2 \\ \\ f_{\frac{N}{2}} \end{bmatrix}$$

One sided scheme

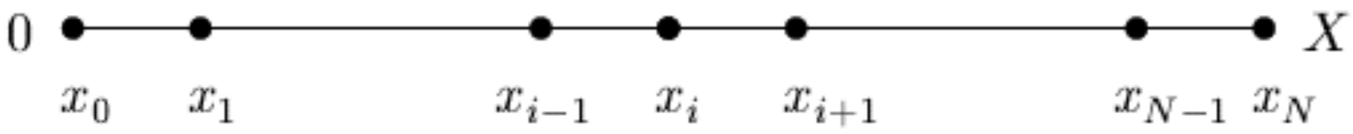
$$\frac{-3u_{N-2} + 4u_{N-1} - u_N}{2\Delta x} = 0$$

$$Au = F$$

$$A = \frac{1}{\Delta x^2}\begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & 1 & & \\ & & & & & \\ & & & -1 & 2 & 1 \\ & & & -3 & 4 & -1 \end{bmatrix}$$

$$u = \begin{bmatrix} u_1 \\ u_2 \\ \\ u_N \end{bmatrix}; \quad \mathrm{F} = \begin{bmatrix} f_1 \\ f_2 \\ \\ f_{N-1} \\ 0 \end{bmatrix}$$

Other types of Boundary conditions



$$-\left(\frac{u_{i-1} - 2u_i + u_{i+1}}{\Delta x^2}\right) = f_i$$

1. Non homogeneous Dirichlet BC

$$u_0 = g_0$$

$$\frac{2u_1 - u_2}{(\Delta x)^2} = f_1 + \frac{g_0}{(\Delta x)^2}$$ 1st eqn change

2. Non homogeneous Nuemann BC

$$\frac{\partial u}{\partial x}(1) = g_1$$

$$u_{N+1} = u_{N-1} + 2\Delta x g_1$$

$$-\left(\frac{u_{N-1} - 2u_N + u_{N+1}}{\Delta x^2}\right) = f_N$$

$$-\left(\frac{2u_{N-1} - 2u_N}{\Delta x^2}\right) = f_N + \frac{2\Delta x g_1}{\Delta x^2}$$

$$\frac{-u_{N-1} + u_N}{\Delta x^2} = \frac{f_N}{2} + \frac{g_1}{\Delta x}$$ Only f changes

3. Non homogeneous Robin B.C

$$\frac{\partial u}{\partial x}(1) + \alpha u(1) = g_2$$

$$\frac{u_{N+1} - u_{N-1}}{2\Delta x} + \alpha u_N = g_2$$

$$u_{N+1} = 2g_2\Delta x - 2\alpha u_N \Delta x + u_{N-1}$$

$$\frac{-u_{N-1} + (1 + \alpha\Delta x)u_N}{\Delta x^2} = \frac{f_N}{2} + \frac{g_2}{\Delta x}$$ Both A and f changes

NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Ordinary Differential Equations

PDE

Prof. Jayati Sarkar

Classification of Partial Differential Equations

- A general second order PDE

$$a\phi_{xx} + b\phi_{xy} + c\phi_{yy} + d\phi_x + e\phi_y + f\phi + g = 0$$

where

$$a,b,c,d,e,f,g = F(x,y) \neq F(\phi)$$

- The behavior of the PDE depends on the sign of the discriminant

$$D = b^2 - 4ac$$

$if\ D<0$ → **PDE is called *elliptic***

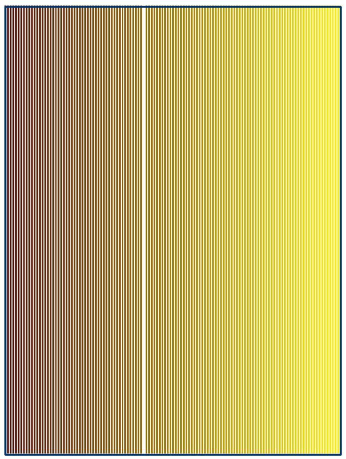
$if\ D=0$ → **PDE is called *parabolic***

$if\ D>0$ → **PDE is called *hyperbolic***

Steady state Heat Conduction in a 1D slab

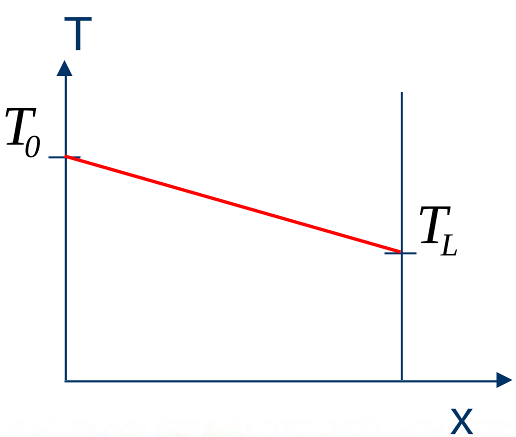
$$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) = 0$$

with $T(0) = T_0 \;, T(L) = T_L$



For const k, the soln is given by

$$T(x) = T_0 + \frac{(T_L - T_0)}{L}x$$

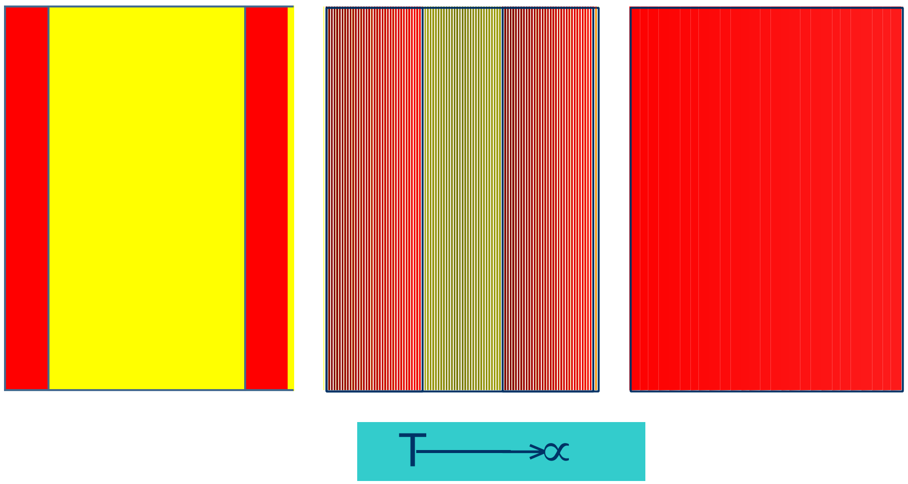


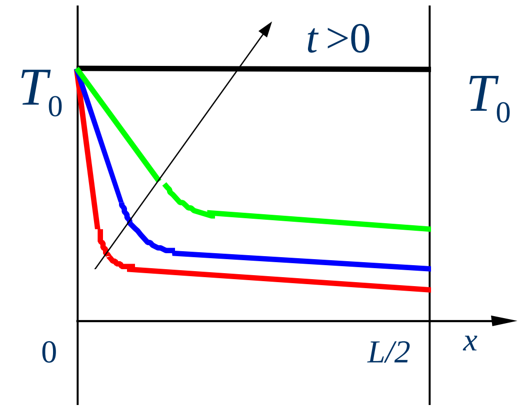
Important lessons to learn!

1. The temperature at any point x is *influenced* by temperatures at *both boundaries*.

2. In the absence of source terms, $T(x)$ is *bounded* by the *boundary* temperatures.

Parabolic Partial Differential Equation





Unsteady state Heat Conduction in a 1D slab

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \qquad \alpha = k/\rho C_p$$

with

$$T(x,0) = T(x)$$

$$T(0,t) = T_0 \quad T^{i}$$

$$(L,t) = T_0$$

Parabolic Partial Differential Equation (Contd.)

Solution is given by

$$T(x,t) = T_0 + \sum_{n=1}^{\infty} B_n \sin\left(\frac{n\pi x}{L}\right) e^{\frac{-\alpha n^2 \pi^2}{L^2} t}$$

where

$$B_n = \frac{2}{L}\int_0^L (T_i(x) - T_o)\sin\left(\frac{n\pi x}{L}\right) dx$$

Important lessons to learn!

The *boundary temperature* T_0 *influences* the temperature T(x,t) at every point in the domain, just as with elliptic PDE's

The *initial conditions* only affect *future* temperatures, not *past* temperatures.

Parabolic Partial Differential Equation (Contd.)

Solution is given by

$$T(x,t) = T_0 + \sum_{n=1}^{\infty} B_n \sin\left(\frac{n\pi x}{L}\right) e^{\frac{-\alpha n^2 \pi^2}{L^2} t}$$

where

$$B_n = \frac{2}{L}\int_0^L (T_i(x) - T_o)\sin\left(\frac{n\pi x}{L}\right) dx$$

Important lessons to learn!

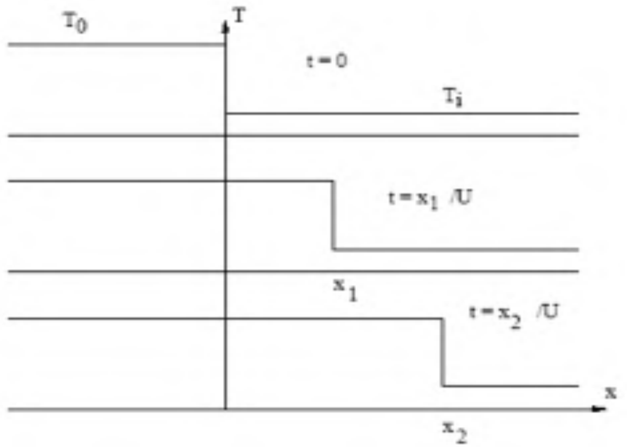
The initial conditions influence the temperature at every point in the domain for all future times. The *amount* of influence decreases with time, and may affect different spatial points to different degrees.

A *steady state* is reached for $t \;\; \infty$. Here, the solution becomes independent of T_i

$(x\ ,0)$. *It also recovers its elliptic spatial behavior.*

The temperature is *bounded* by its initial and boundary conditions in the absence of source term

Hyperbolic Partial Differential Equations



Temperature Variation with time

- 1D flow of fluid in a channel

$$\frac{\partial\left(\rho C_P T\right)}{\partial t}+\frac{\partial\left(\rho C_P U T\right)}{\partial t}=0$$

with

$$T(x,0)=T_i(x)$$
$$T(x\leq 0,t)=T_0$$

Solution is given by

$$T(x,t)=T((x-Ut),0)$$

or

$$T(x,t)=T_i \quad for \quad t<\frac{x}{U}$$
$$=T_0 \quad for \quad t\geq\frac{x}{U}$$

Important lessons to learn!

1.The *upstream* boundary conditions and not the downstream condition *affects* the solution of the domain.

2. The *inlet* boundary condition *propagates* with a finite speed U.

3. The inlet boundary condition is not felt at point x until t=x/U.

Laplace Equation (Elliptic)

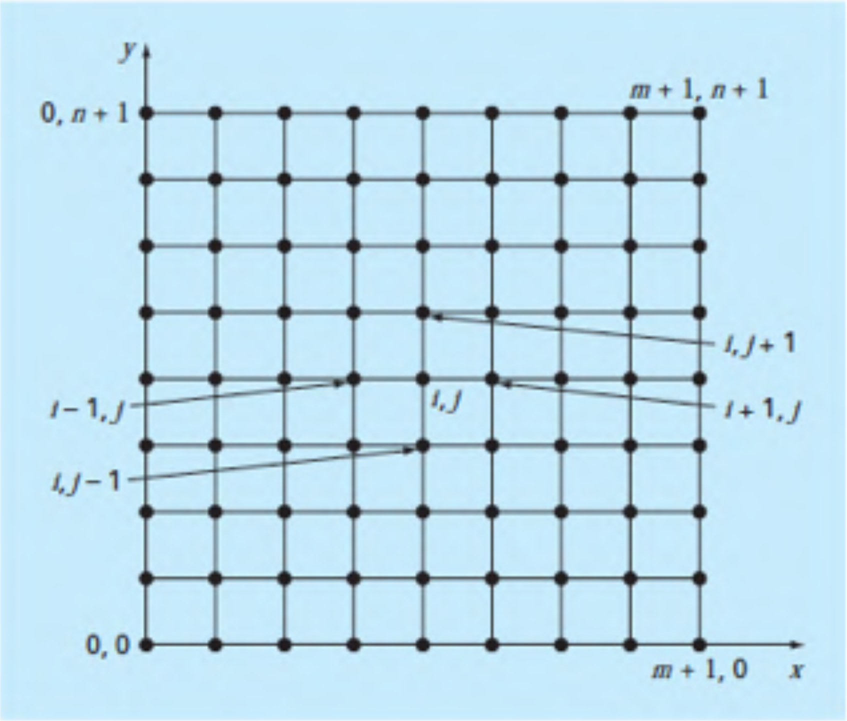
- Heat Transfer in 2D

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2}$$

$$\frac{\partial^2 T}{\partial y^2} = \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2}$$

$$\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} = 0$$

$$T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - 4T_{i,j} = 0$$



Laplace Equation (Elliptic)-Dirichlet Boundary Condition

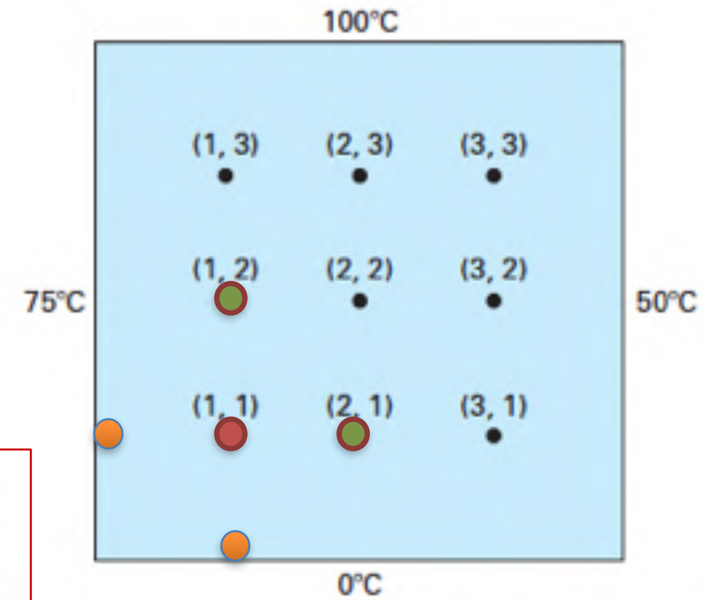
- Heat Transfer in 2D

$$T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - 4T_{i,j} = 0$$

$$T_{21} + T_{01} + T_{12} + T_{10} - 4T_{11} = 0$$

$$T_{01} = 75 \text{ and } T_{10} = 0,$$

$$-4T_{11} + T_{12} + T_{21} = -75$$



$$
\begin{array}{rrrrrrrrrl}
4T_{11} & -T_{21} & & -T_{12} & & & & & & = 75 \\
-T_{11} & +4T_{21} & -T_{31} & & -T_{22} & & & & & = 0 \\
& -T_{21} & +4T_{31} & & & -T_{32} & & & & = 50 \\
-T_{11} & & & +4T_{12} & -T_{22} & & -T_{13} & & & = 75 \\
& -T_{21} & & -T_{12} & +4T_{22} & -T_{32} & & -T_{23} & & = 0 \\
& & -T_{31} & & -T_{22} & +4T_{32} & & & -T_{33} & = 50 \\
& & & -T_{12} & & & +4T_{13} & -T_{23} & & = 175 \\
& & & & -T_{22} & & -T_{13} & +4T_{23} & -T_{33} & = 100 \\
& & & & & -T_{32} & & -T_{23} & +4T_{33} & = 150
\end{array}
$$

- Gauss Seidel

$$T_{i,j} = \frac{T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1}}{4}$$

$$T_{i,j}^{\text{new}} = \lambda T_{i,j}^{\text{new}} + (1-\lambda) T_{i,j}^{\text{old}}$$

$$|(\varepsilon_a)_{i,j}| = \left| \frac{T_{i,j}^{\text{new}} - T_{i,j}^{\text{old}}}{T_{i,j}^{\text{new}}} \right| 100\%$$

$$i=1, j=1 \qquad T_{11} = \frac{0+75+0+0}{4} = 18.75$$

applying overrelaxation yields

$$T_{11} = 1.5(18.75) + (1-1.5)0 = 28.125$$

For $i=2, j=1,$

$$T_{21} = \frac{0+28.125+0+0}{4} = 7.03125$$

$$T_{21} = 1.5(7.03125) + (1-1.5)0 = 10.54688$$

For $i=3, j=1,$

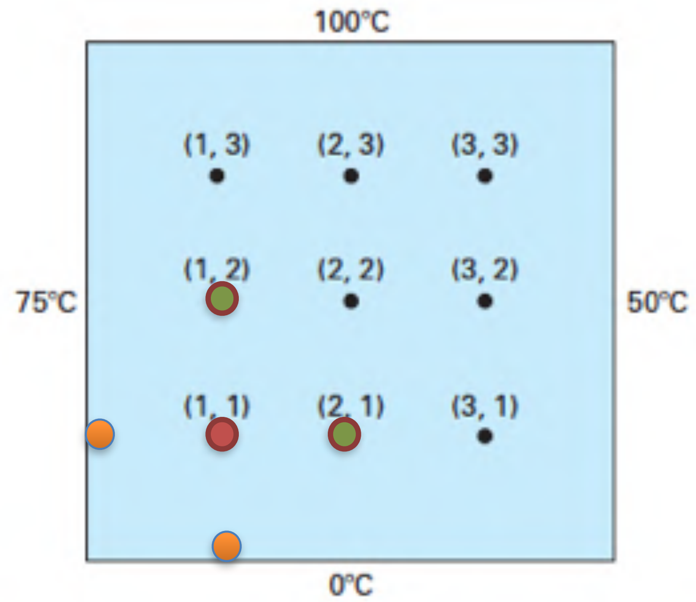
$$T_{31} = \frac{50+10.54688+0+0}{4} = 15.13672$$

$$T_{31} = 1.5(15.13672) + (1-1.5)0 = 22.70508$$

The computation is repeated for the other rows to give

$$T_{12} = 38.67188 \quad T_{22} = 18.45703 \quad T_{32} = 34.18579$$
$$T_{13} = 80.12696 \quad T_{23} = 74.46900 \quad T_{33} = 96.99554$$

Because all the $T_{i,j}$'s are initially zero, all ε_a's for the first iteration will be 100%.



For the second iteration the results are

$$T_{11} = 32.51953 \quad T_{21} = 22.35718 \quad T_{31} = 28.60108$$
$$T_{12} = 57.95288 \quad T_{22} = 61.63333 \quad T_{32} = 71.86833$$
$$T_{13} = 75.21973 \quad T_{23} = 87.95872 \quad T_{33} = 67.68736$$

The error for $T_{1,1}$ can be estimated as

$$|(\varepsilon_a)_{1,1}| = \left|\frac{32.51953 - 28.12500}{32.51953}\right| 100\% = 13.5\%$$

Laplace Equation (Elliptic)

- Solution after 9th iteration

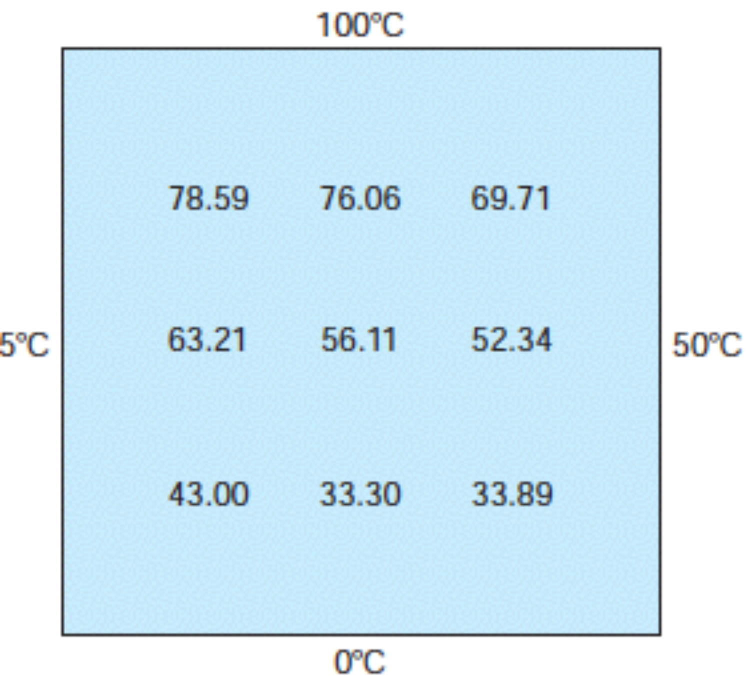


plate is 40×40 cm

$k' = 0.49 \text{ cal/(s} \cdot \text{cm} \cdot {}^\circ\text{C)}$

- Post Processing(Heat Fluxes)

$$q_x = -k' \frac{T_{i+1,j} - T_{i-1,j}}{2\,\Delta x} \qquad q_x = -k' \frac{T_{i,j+1} - T_{i,j-1}}{2\,\Delta y}$$

The resultant heat flux can be computed from these two quantities

$$q_n = \sqrt{q_x^2 + q_y^2}$$

where the direction of q_n is given by

$$\theta = \tan^{-1}\left(\frac{q_y}{q_x}\right)$$

for $q_x > 0$ and

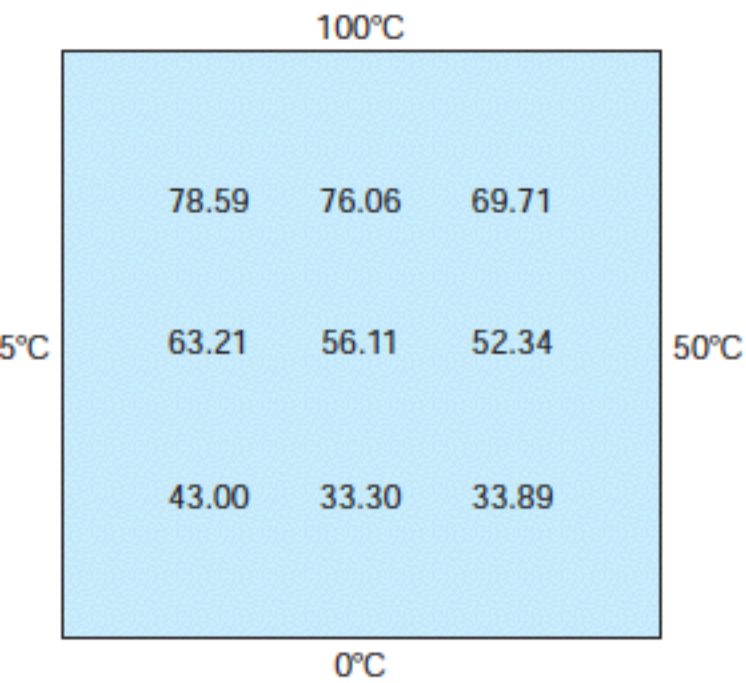
$$\theta = \tan^{-1}\left(\frac{q_y}{q_x}\right) + \pi \qquad \text{for } q_x < 0$$

$$q_x = -0.49 \frac{\text{cal}}{\text{s} \cdot \text{cm} \cdot {}^\circ\text{C}} \frac{(33.29755 - 75)^\circ\text{C}}{2(10\text{ cm})} = 1.022 \text{ cal/(cm}^2 \cdot \text{s)}$$

$$q_y = -0.49 \frac{\text{cal}}{\text{s} \cdot \text{cm} \cdot {}^\circ\text{C}} \frac{(63.21152 - 0)^\circ\text{C}}{2(10\text{ cm})} = -1.549 \text{ cal/(cm}^2 \cdot \text{s)}$$

Laplace Equation (Elliptic)

- Solution after 9th iteration



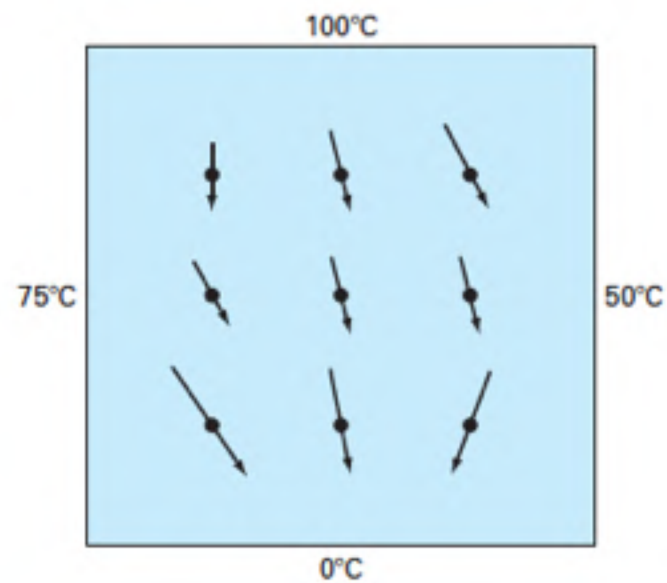
- Post Processing(Heat Fluxes)

The resultant flux can be computed with

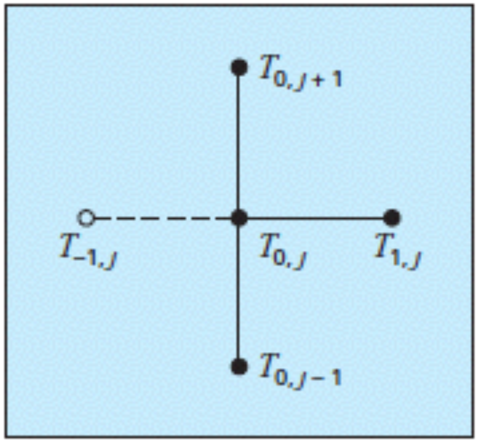
$$q_n = \sqrt{(1.022)^2 + (-1.549)^2} = 1.856 \text{ cal/(cm}^2\cdot\text{s)}$$

and the angle of its trajectory by

$$\theta = \tan^{-1}\left(\frac{-1.549}{1.022}\right) = -0.98758 \times \frac{180^\circ}{\pi} = -56.584^\circ$$



Laplace Equation (Elliptic)-Neumann Boundary Condition



$$T_{1,j} + T_{-1,j} + T_{0,j+1} + T_{0,j-1} - 4T_{0,j} = 0$$

$$\frac{\partial T}{\partial x} \cong \frac{T_{1,j} - T_{-1,j}}{2\,\Delta x} \qquad T_{-1,j} = T_{1,j} - 2\,\Delta x \frac{\partial T}{\partial x}$$

$$2T_{1,j} - 2\,\Delta x \frac{\partial T}{\partial x} + T_{0,j+1} + T_{0,j-1} - 4T_{0,j} = 0$$

Heated Plate with an Insulated Edge

Solution. The general equation to characterize a derivative at the lower edge (that is, at $j = 0$) of a heated plate is

$$T_{i+1,0} + T_{i-1,0} + 2T_{i,1} - 2\,\Delta y \frac{\partial T}{\partial y} - 4T_{i,0} = 0$$

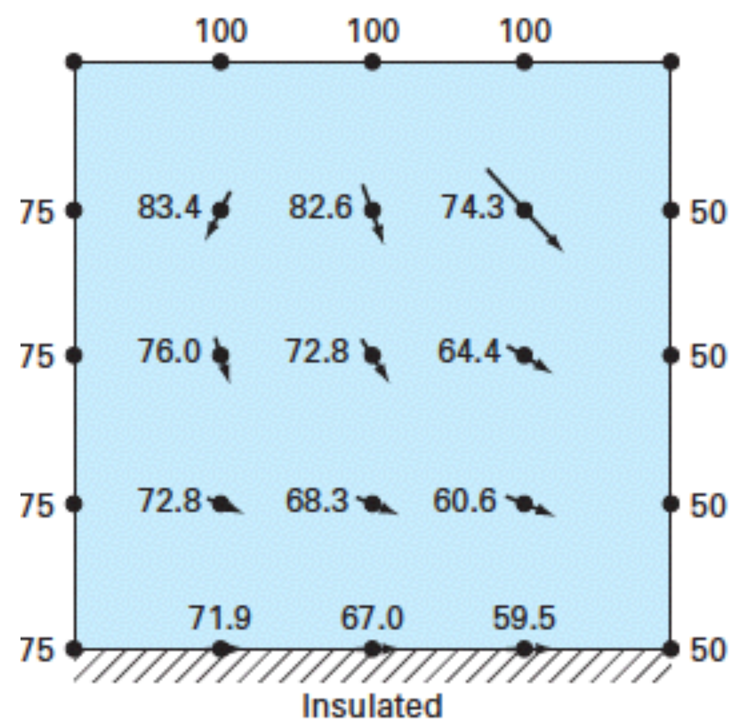
For an insulated edge, the derivative is zero and the equation becomes

$$T_{i+1,0} + T_{i-1,0} + 2T_{i,1} - 4T_{i,0} = 0$$

Laplace Equation (Elliptic)-Neumann Boundary Condition

$$
\begin{bmatrix}
4 & -1 & & -2 & & & & & & & & \\
-1 & 4 & -1 & & -2 & & & & & & & \\
 & -1 & 4 & & & -2 & & & & & & \\
-1 & & & 4 & -1 & & -1 & & & & & \\
 & -1 & & -1 & 4 & -1 & & -1 & & & & \\
 & & -1 & & -1 & 4 & & & -1 & & & \\
 & & & -1 & & & 4 & -1 & & -1 & & \\
 & & & & -1 & & -1 & 4 & -1 & & -1 & \\
 & & & & & -1 & & -1 & 4 & & & -1 \\
 & & & & & & -1 & & & 4 & -1 & \\
 & & & & & & & -1 & & -1 & 4 & -1 \\
 & & & & & & & & -1 & & -1 & 4
\end{bmatrix}
\begin{Bmatrix} T_{10} \\ T_{20} \\ T_{30} \\ T_{11} \\ T_{21} \\ T_{31} \\ T_{12} \\ T_{22} \\ T_{32} \\ T_{13} \\ T_{23} \\ T_{33} \end{Bmatrix}
=
\begin{Bmatrix} 75 \\ 0 \\ 50 \\ 75 \\ 0 \\ 50 \\ 75 \\ 0 \\ 50 \\ 175 \\ 100 \\ 150 \end{Bmatrix}
$$

$$
\begin{array}{lll}
T_{10} = 71.91 & T_{20} = 67.01 & T_{30} = 59.54 \\
T_{11} = 72.81 & T_{21} = 68.31 & T_{31} = 60.57 \\
T_{12} = 76.01 & T_{22} = 72.84 & T_{32} = 64.42 \\
T_{13} = 83.41 & T_{23} = 82.63 & T_{33} = 74.26
\end{array}
$$



Unsteady Heat Conduction Equation (Parabolic)

$$k\frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t}$$

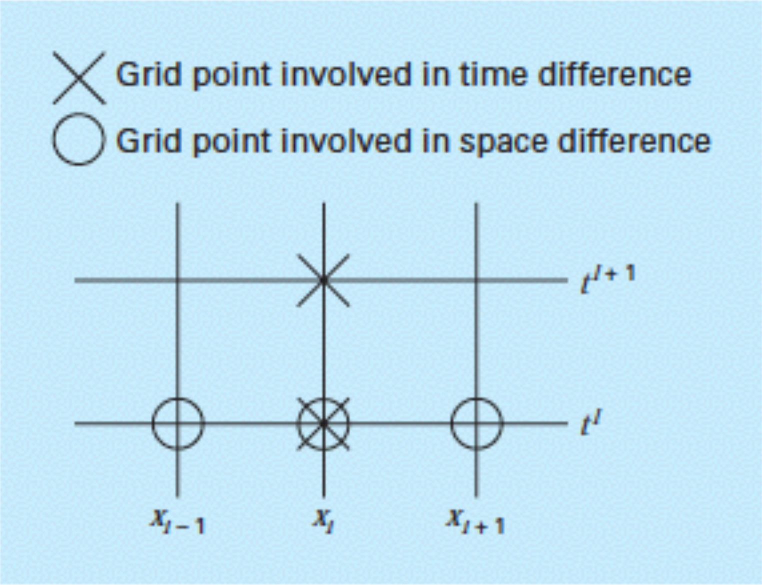
$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1}^l - 2T_i^l + T_{i-1}^l}{\Delta x^2}$$

$$\frac{\partial T}{\partial t} = \frac{T_i^{l+1} - T_i^l}{\Delta t}$$

$$k\frac{T_{i+1}^l - 2T_i^l + T_{i-1}^l}{(\Delta x)^2} = \frac{T_i^{l+1} - T_i^l}{\Delta t}$$

$$T_i^{l+1} = T_i^l + \lambda\left(T_{i+1}^l - 2T_i^l + T_{i-1}^l\right)$$

where $\lambda = k\,\Delta t/(\Delta x)^2$.



Unsteady Heat Conduction Equation (Parabolic)

Explicit Solution of the One-Dimensional Heat-Conduction Equation

thin rod with a length of 10 cm

$k' = 0.49$ cal/(s · cm · °C)

$\Delta x = 2$ cm, and $\Delta t = 0.1$ s.

$T(0) = 100$°C and $T(10) = 50$°C

$C = 0.2174$ cal/(g · °C) and $\rho = 2.7$ g/cm^3.

$\boxed{\lambda = 0.835(0.1)/(2)^2 = 0.020875.}$

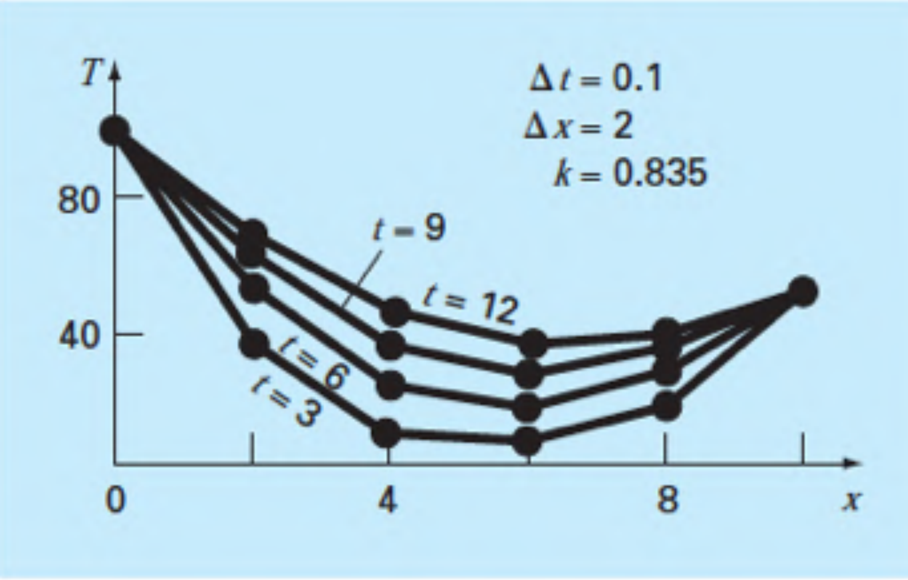
$$T_1^1 = 0 + 0.020875[0 - 2(0) + 100] = 2.0875$$

At the other interior points, $x = 4$, 6, and 8 cm, the results are

$$T_2^1 = 0 + 0.020875[0 - 2(0) + 0] = 0$$
$$T_3^1 = 0 + 0.020875[0 - 2(0) + 0] = 0$$
$$T_4^1 = 0 + 0.020875[50 - 2(0) + 0] = 1.0438$$

At $t = 0.2$ s, the values at the four interior nodes are computed

$$T_1^2 = 2.0875 + 0.020875[0 - 2(2.0875) + 100] = 4.0878$$
$$T_2^2 = 0 + 0.020875[0 - 2(0) + 2.0875] = 0.043577$$
$$T_3^2 = 0 + 0.020875[1.0438 - 2(0) + 0] = 0.021788$$
$$T_4^2 = 1.0438 + 0.020875[50 - 2(1.0438) + 0] = 2.0439$$



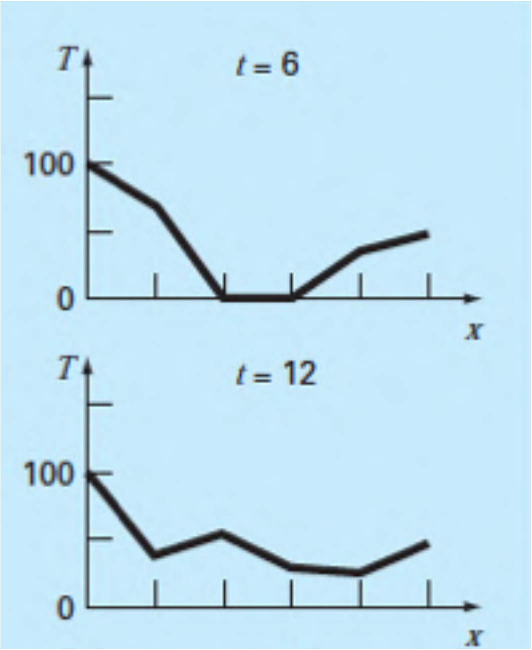
Unsteady Heat Conduction Equation (Parabolic)

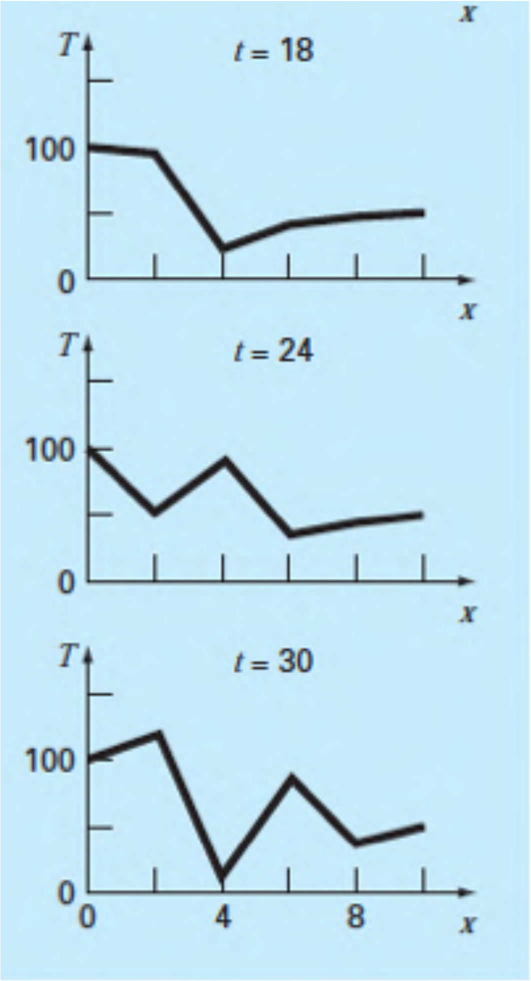
Explicit Solution of the One-Dimensional Heat-Conduction Equation

stable if $\lambda \le 1/2$

$$\boxed{\Delta t \le \frac{1}{2}\frac{\Delta x^2}{k}}$$

$\lambda = 0.735.$





NUMERICAL METHODS IN CHEMICAL ENGINEERING

CLL-113

Ordinary Differential Equations

PDE

Prof. Jayati Sarkar

भारतीय प्रौद्योगिकी संस्थान दिल्ली
Indian Institute of Technology Delhi


Thank you