ESO208A: Computational Methods in Engineering

Richa Ojha

Department of Civil Engineering

IIT Kanpur

Acknowledgements: Profs. Abhas Singh and Shivam Tripathi (CE)

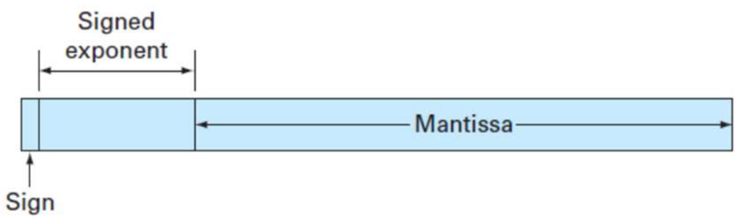
Round-off error

Number representation in computers

- Integer
- Fixed point
- Floating point

Round-off error

Mantisaa



- Mantisaa is usually normalized if it has leading zero digits

  For example, $1/34 = 0.0294117$ (in a base 10 system)
- If this has to be stored in a computer, that allows 4 decimal places.

  $1/34$ would be stored as 0.0294×10^{0}

- The number is normalized to remove leading zero, 0.2941×10^{-1}
- The consequence of normalization is that absolute value of m is
- limited. That is $1/b \leq m < 1$, where b is the base.

Round-off error

3- important properties

1. Maximum positive value

x_{max} 0.999×10^{9}

x_{min} -0.999×10^{9}

Over flow error

2. Hole near zero

0.000×10^{-9}

0.100×10^{-9} 10^{-9}

0.101×10^{-9} $- 10^{-12}$

0.102×10^{-9}

3. Interval between numbers increase

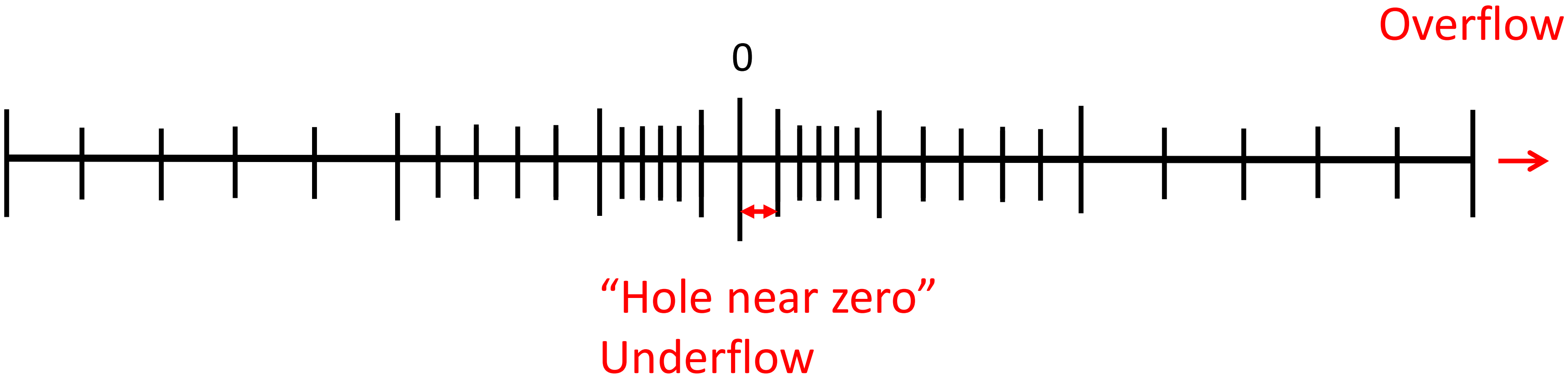
0.998×10^{3}

0.999×10^{3} 1

0.100×10^{4} 1

0.101×10^{4} 10

Floating point number representation



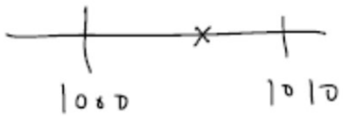
Consider a hypothetical system

System 3 places for mantissa
1 place for exponent

1000 $\quad 0.100 \times 10^4$

1010 $\quad 0.101 \times 10^4$

1007



Round-off error

Chopping

$1007 \rightarrow 0.100\cancel{7} \times 10^4$

$|\Delta x| \le 10$

Rounding

$|\Delta x| \le 5$

Relative error

$$\left|\frac{\Delta x}{x}\right| = \frac{7}{1007}$$

Chopping $\left|\frac{\Delta x}{x}\right| \le \frac{10}{1000} = 10^{-2}$

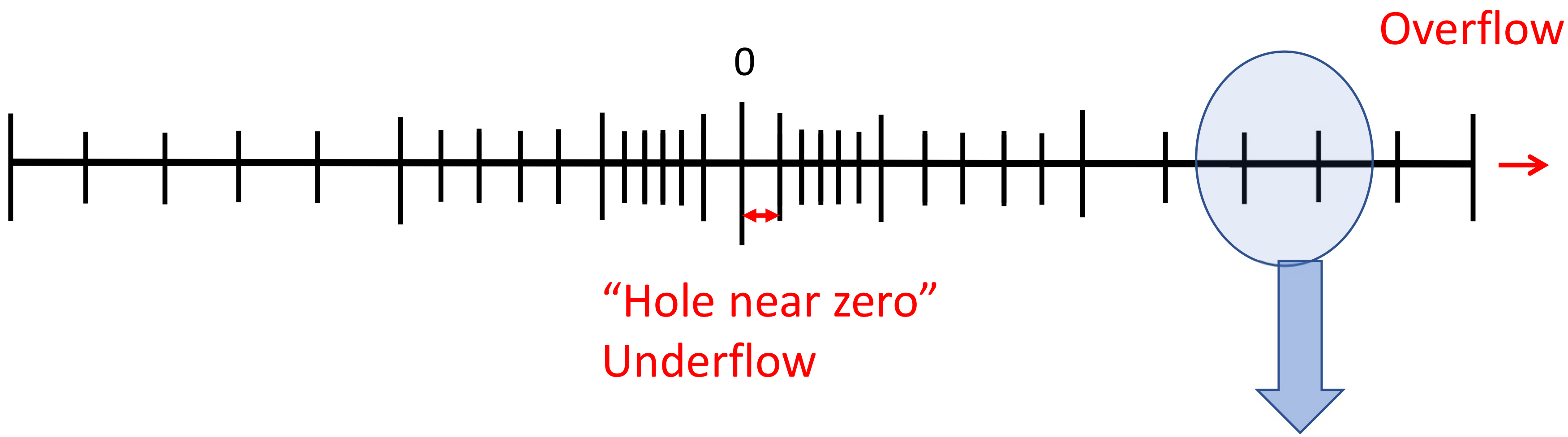
Rounding $\le \frac{1}{2} 10^{-2}$

General Round off $\left|\frac{\Delta x}{x}\right| \le \frac{1}{2} 10^{1-t}$

$t \rightarrow$ number of significant digits in mantissa

$\le \frac{1}{2} 2^{1-t}$

Floating point number representation

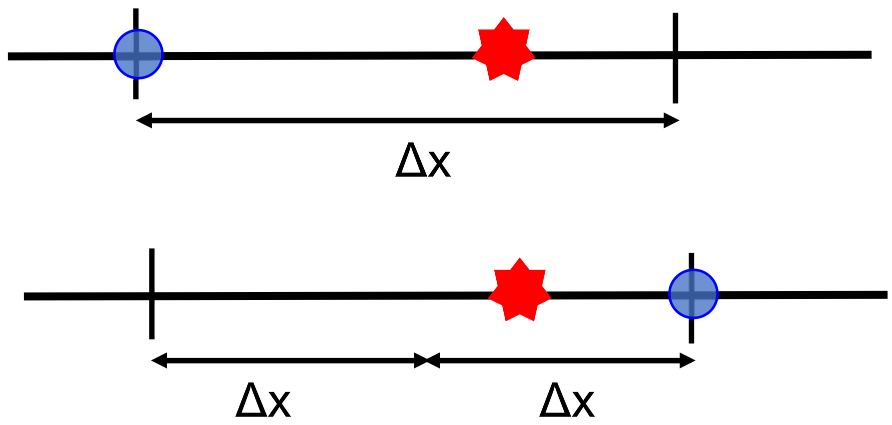


Chopping

Round-off Errors

$$\left|\frac{\Delta x}{x}\right| \leq u = 0.5 \times b^{1-t}$$

Rounding



Real number in Maths and Computer are not the same

Round-off errors can be avoided subtraction of nearly equal nos.

Round-off error

Why round-off error is important?

Let us say you want to add two numbers, $208.00 + 0.25 = 208.25$

In computer, the numbers would be represented as:

0.208×10^3

0.25×10^0

In floating point, we can change the number such that it has the highest power

0.208×10^3

$+ 0.00025 \times 10^3 = 0.20825 \times 10^3$

Computer will round off and will return 0.208×10^3

Round-off error

Why round-off error is important?

Another example

$a+1-a = 1$

Let us take a= 10^20

Output from computer = 0 (because for this large number there is a hole)

If, I write, a-a+1 then output from computer =1

Round-off error

Why round-off error is important?

Most important effect is in subtractions

Subtraction of two nearly equal numbers:

0.246×10^3

0.245×10^3

In floating point, we can change the number such that it has the highest power

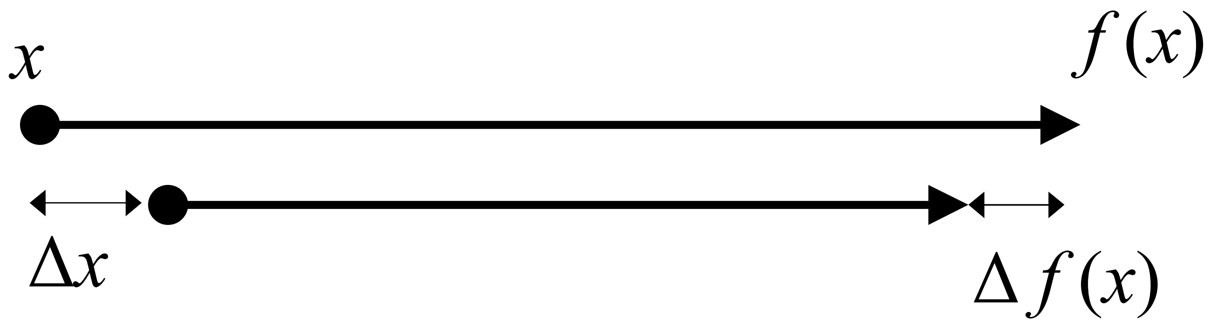
0.246×10^3

$- 0.245 \times 10^3 = 0.001 \times 10^3$

Matissa normalize $= 0.100 \times 10^1$ (3-significant digits)

But we actually have 1 significant digit. This is called loss of significance

Forward error analysis



Condition number of the problem

$$C_p = \frac{\text{Relative error in } f(x)}{\text{Relative error in } x} = \frac{\Delta f(x) / f(x)}{\Delta x / x} = \left| \frac{x f'(x)}{f(x)} \right|$$

$C_p \leq 1$ - well-conditioned problem

$C_p > 1$ - ill-conditioned problem

Characteristic of the problem

Forward Error Analysis:

Single Variable Function: $y = f(x)$. If an error is introduced in x, what is the error in y?

$$\Delta x = x - \tilde{x} \qquad \Delta y = y - \tilde{y} = f(x) - f(\tilde{x})$$

$$f(x) = f(\tilde{x} + \Delta x) = f(\tilde{x}) + \Delta x f'(\tilde{x}) + \frac{\Delta x^2}{2!} f''(\tilde{x}) + ..$$

Assuming the error to be small, the 2nd and higher order terms are neglected. (a first order approximation!)

$$\Delta y = f(x) - f(\tilde{x}) \approx \Delta x f'(\tilde{x})$$

Condition Number of the Problem (C_p):

$$C_p = \frac{Relative\ Error\ in\ y}{Relative\ Error\ in\ x} = \left|\frac{\Delta y / y}{\Delta x / x}\right| \approx \left|\frac{\Delta x f'(\tilde{x}) / f(x)}{\Delta x / x}\right| = \left|\frac{x f'(\tilde{x})}{f(x)}\right|$$

Also: $C_p = \left|\frac{\Delta y / y}{\Delta x / x}\right| = \left|\frac{\left(f(x) - f(\tilde{x})\right) / f(x)}{\Delta x / x}\right| = \left|\frac{x}{f(x)} \frac{f(\tilde{x} + \Delta x) - f(\tilde{x})}{\Delta x}\right|$

As $\Delta x \longrightarrow 0$,

$$C_p = \left|\frac{x f'(\tilde{x})}{f(x)}\right|$$

$$C_p = \left|\frac{\tilde{x} f'(\tilde{x})}{f(\tilde{x})}\right|$$

$C_p < 1$: problem is well-conditioned, error is attenuated

$C_p > 1$: problem is ill-conditioned, error is amplified

$C_p = 1$: neutral, error is translated

Examples of Forward Error Analysis and C_p:

✓ Problem 1: $y = e^x$;

$$\Delta y = \Delta x e^x;\ C_p = \left| \frac{\Delta y / y}{\Delta x / x} \right| = x.$$

The problem is well-conditioned for $0 \leq |x| < 1$; neutral at $|x| = 1$ and ill-conditioned for $|x| > 1$.

Examples of Forward Error Analysis and C_p:

✓ Problem 2: Solve the following system of equations:

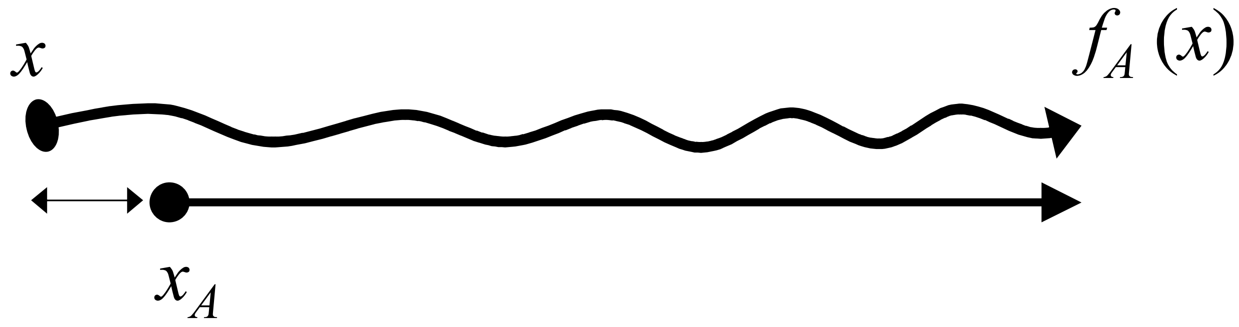
$$x + \alpha y = 1; \qquad \alpha x + y = 0$$

Solving: $x = \frac{1}{1-\alpha^2} = x(\alpha)$; $\quad x'(\alpha) = \frac{2\alpha}{(1-\alpha^2)^2}$

$$C_p = \left|\frac{\alpha x'(\alpha)}{x}\right| = \left|\frac{\alpha \dfrac{2\alpha}{(1-\alpha^2)^2}}{\dfrac{1}{1-\alpha^2}}\right| = \left|\frac{2\alpha^2}{1-\alpha^2}\right|$$

well-conditioned for $|\alpha| << 1$ and ill-conditioned for $\alpha \approx 1$.

Backward error analysis



Condition number of the algorithm

$$\left|\frac{x-x_A}{x}\right| \le C_A u \qquad u \text{ is machine precision}$$

Characteristic of the numerical stability of the algorithm

small C_A - stable algorithm

large C_A - instable algorithm

Backward error analysis- Example

Example

4 digit decimal

$$u = \frac{1}{2} \times 10^{1-4}$$

$$= 0.5 \times 10^{-3}$$

$$u = \frac{1}{2} 10^{1-t}$$

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

$f(x)$
$= 0.8688 \times 10^{-2}$.

$$x = 1^{\circ}$$

$$fl(\pi/180) = 0.1745 \times 10^{-1}$$

$$fl(\sin x) = 0.1745 \times 10^{-1}$$

$$fl(1+\sin x) = 0.1017 \times 10^{1}$$

$$fl(\sqrt{1+\sin x}) = 0.1008 \times 10^{1}$$

$$fl(\sqrt{1+\sin x} - 1) = 0.8000 \times 10^{-2}$$

Backward error analysis- Example

$$\sqrt{1+\sin x} - 1 = 0.8000 \times 10^{-2}$$

$$x_A = 0.9204 \times 10^{1}$$

$$\left|\frac{x - x_A}{x}\right| = 0.0796 \leq C_A u$$

$$\boxed{C_A \approx 160}$$

$$f(x) = \left(\sqrt{1+\sin x} - 1\right)\frac{(\sqrt{1+\sin x}+1)}{(\sqrt{1+\sin x}+1)}$$

$$= \frac{\sin x}{\sqrt{1+\sin x}+1}$$

$$C_A \approx 0.4$$

Condition Number of an algorithm can be changed

Summary

- How mantissa is represented in computers?
- What are chopping and rounding?
- What is Condition Number of Problem and Algorithm?