Lecture Note-Numerical Analysis (2): Numerical Error

1. Floating point handling : Round off error

(1) Floating point handling : Round off error

$$\pi = 3.14159265358979310 8xxxxxxxxxx \cdots$$
$$100\pi = 314.1592653589793108xxxxxxxxxx \cdots$$
$$0.01\pi = 0.03141592653589793108xxxxxxxxxx \cdots$$

Single precision:

$$\pi \approx 0.31415926E+01$$
$$100\pi \approx 0.31415926E+03$$
$$0.01\pi \approx 0.31415926E-01$$

Double precision:

$$\pi \approx 0.3141592653589793E+01$$
$$100\pi \approx 0.3141592653589793E+03$$
$$0.01\pi \approx 0.3141592653589793E-01$$

(2) IEEE 754 format as a standard computer representation of real numbers

S	exponent	fraction

S: sign bit

Real number (floating point number): f

$$\boxed{\begin{aligned} f &= \text{sign} \times 2^{\text{exponent}} \times \text{fraction} \\ \text{fraction} &= 1.\text{xxxxxxxxx} \ \cdots \quad \text{, where 1 is not explicitly stored} \end{aligned}}$$

(3) IEEE 754 format for 1 32-bit float real data type 3

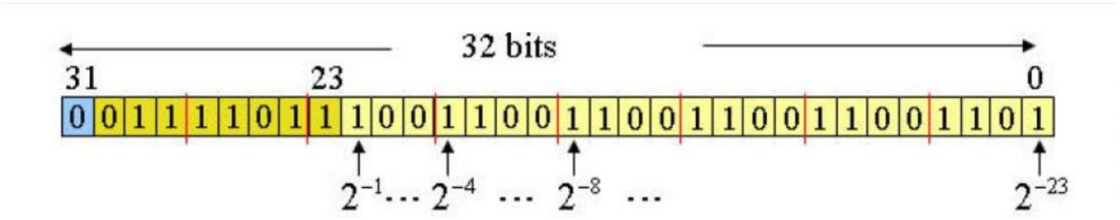
2-bit (sign 1bit, exponent: 8bits, fraction: 23 bits)

31								23																							0
0	0	1	1	1	1	0	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1
S	SE	E6	E5	E4	E3	E2	E1	E0	F22	F21	F20	F19	F18	F17	F16	F15	F14	F13	F12	F11	F10	F9	F8	F7	F6	F5	F4	F3	F2	F1	F0

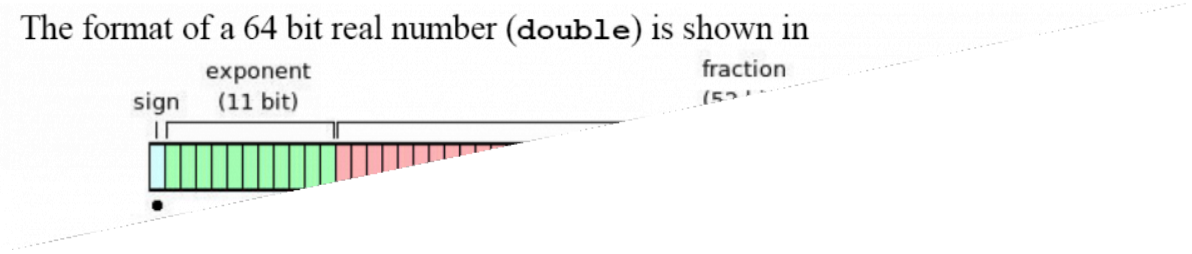
S : sign bit

SE : sign bit for exponent

F22 = 2^{-1} , F21 = 2^{-1}, F20 = 2^{-2}, F19 = 2^{-3}, F18 = 2^{-4}, F17 = 2^{-5}, ………………………, F0 = 2^{-23}



(Note) 64-bit (sign 1 bit, exponent: 11bits, fraction: 52 bits)

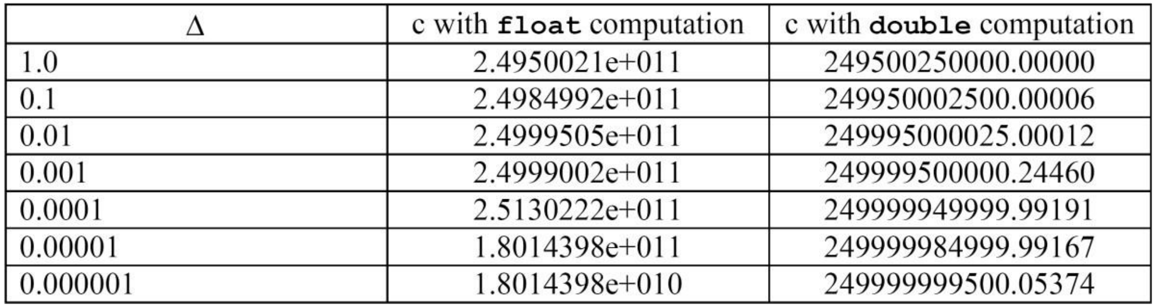

The format of a 64 bit real number (**double**) is shown in

(4) Scientific notion for Real numbers such as $\pi \approx 3.1415926E+00$

(Example 1) $$f = \int_0^{1000} x^3 dx = \frac{1}{4} x^4 \Big|_0^{1000} = 2.5E+11$$

Numerical approximation by

$$f = \Delta \sum_0^N x^3, \quad \text{with } N = \frac{1000}{\Delta}$$



Δ	c with **float** computation	c with **double** computation
1.0	2.4950021e+011	249500250000.00000
0.1	2.4984992e+011	249950002500.00006
0.01	2.4999505e+011	249995000025.00012
0.001	2.4999002e+011	249999500000.24460
0.0001	2.5130222e+011	249999949999.99191
0.00001	1.8014398e+011	249999984999.99167
0.000001	1.8014398e+010	249999999500.05374

See the last two rows with the single precision (float) → incorrect answers

2. Taylor Series Expansion of a smooth function at a given point: Truncation Error

- Non-elementary functions such as trigonometric, exponential and others are expressed in an approximate fashion using Taylor series when their values, derivatives, and integrals are computed.

- Any smooth function can be approximated as a polynomial. Taylor series provides a means to predict the value of a function at one point in terms of the function value and its derivatives at another point.

$$f(x) = f(a) + f'(a)(x-a) + \frac{1}{2!} f''(a)(x-a)^2 + \frac{1}{3!} f^{(3)}(a)(x-a)^3 + \cdots + \frac{1}{n!} f^{(n)}(a)(x-a)^n + R_n$$

Where the remainder term $R_n = \frac{1}{(n+1)!} f^{(n+1)}(\xi)(x-a)^{n+1}$ for some $\xi \in [x, a], \text{if } x \le a$ or $\xi \in [a, x], \text{if } a \le x$

Generally, we use this formula in the numerical analysis with the following condition

$|x-a| << 1$ to meet $\lim_{n\to\infty} (x-a)^{n+1} = 0$

- Order of the Taylor series approximation

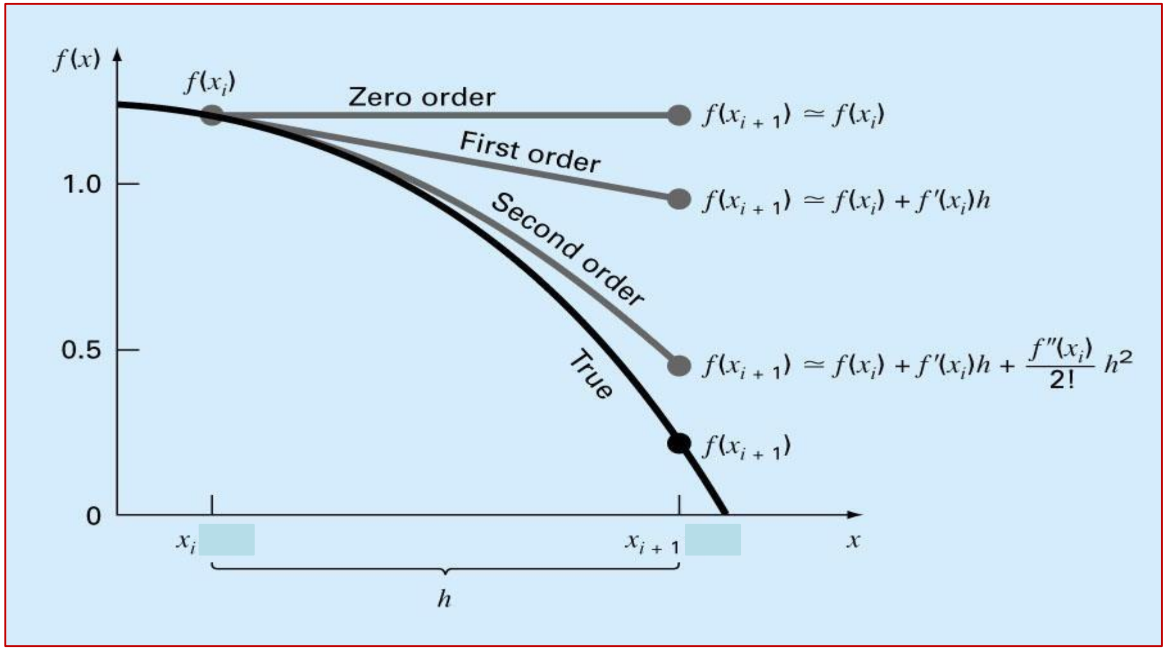
-zero order: $f(x) \approx f(a)$

- 1st order: $f(x) \approx f(a) + f'(a)(x-a)$

-2nd order: $f(x) \approx f(a) + f'(a)(x-a) + \frac{1}{2!} f''(a)(x-a)^2$

-3rd order: $f(x) \approx f(a) + f'(a)(x-a) + \frac{1}{2!} f''(a)(x-a)^2 + \frac{1}{3!} f^{(3)}(a)(x-a)^3$

(Example 1) for $a = x_i$, $x = x_{i+1} = x_i + h$


$f(x)$
$f(x_i)$
Zero order
$f(x_{i+1}) \simeq f(x_i)$
First order
$f(x_{i+1}) \simeq f(x_i) + f'(x_i)h$
Second order
$f(x_{i+1}) \simeq f(x_i) + f'(x_i)h + \frac{f''(x_i)}{2!}h^2$
True
$f(x_{i+1})$
1.0
0.5
0
x_i
x_{i+1}
x
h

○ Truncation Error: The numerical error caused by ignoring the remainder terms during the Taylor series approximation

(Example 2) $f(x) = x^4$ with $x_i = 1,\quad x_{i+1} = x_i + h$

$$\boxed{\begin{array}{c} f(x)=x^4 \\ f'(x)=4x^3 \\ f'(x)=12x^2 \\ f^{(3)}(x)=24x \\ f^{(4)}(x)=24 \\ f^{(5)}(x)=0 \\ \vdots \end{array}} \rightarrow \boxed{\begin{array}{c} f(1)=1 \\ f'(1)=4 \\ f'(1)=12 \\ f^{(3)}(1)=24 \\ f^{(4)}(1)=24 \\ f^{(5)}(1)=0 \\ \vdots \end{array}} \rightarrow \boxed{\begin{array}{l} f(1+h) \approx f(1) + f'(1)h + \dfrac{1}{2!} f'(1)h^2 + \dfrac{1}{3!} f^{(3)}(1)h^3 + \cdots \\ \qquad = 1 + 4h + 6h^2 + 4h^3 + h^4 \end{array}}$$

In case of $h = 0.5$, the true value becomes $\boxed{f(1+h) = f(1.5) = 1.5^4 = 5.0625}$

Order	Approximation	Truncation Error
Zero order	$f(1+h) \approx 1$	4.0625
1st order	$f(1+h) \approx 1 + 4 \times 0.5 = 1 + 2.0 = 3.0$	3.0625
2nd order	$f(1+h) \approx 1 + 4 \times 0.5 + 6 \times 0.5^2$ $= 3.0 + 1.5 = 4.5$	0.5625
3rd order	$f(1+h) = 1 + 4 \times 0.5 + 6 \times 0.5^2 + 4 \times 0.5^3$ $= 4.5 + 0.5 = 5.0$	0.0625
4th order	$f(1+h) = 1 + 4 \times 0.5 + 6 \times 0.5^2 + 4 \times 0.5^3 + 0.5^4$ $= 5.0625$	0.0

h	TRUE	0th	1st	2nd	3rd	4th
2	81	1	9	33	65	81
1	16	1	5	11	15	16
0.5	5.0625	1	3	4.5	5	5.0625
0.25	2.44140625	1	2	2.375	2.4375	2.44140625
0.125	1.601806641	1	1.5	1.59375	1.6015625	1.601806641
0.0625	1.274429321	1	1.25	1.2734375	1.274414063	1.274429321
0.03125	1.130982399	1	1.125	1.130859375	1.130981445	1.130982399
0.015625	1.063980162	1	1.0625	1.063964844	1.063980103	1.063980162

3. Error Propagation associated with Taylor Series Approximation

○ **Assume the following recursive (or iterative) approximation formula using the Taylor Series**

$$\begin{aligned} x_{j+1} &= x_j + h \\ f(x_{j+1}) &= f(x_j + h) \approx f(x_j) + f'(x_j) \times h \end{aligned} \qquad \text{for} \quad j = 0,1,2,3,\cdots$$

(Example 2) $f(x) = x^2$ **with** $x_0 = 0,\quad x_{j+1} = x_j + h$

$f'(x) = 2x$

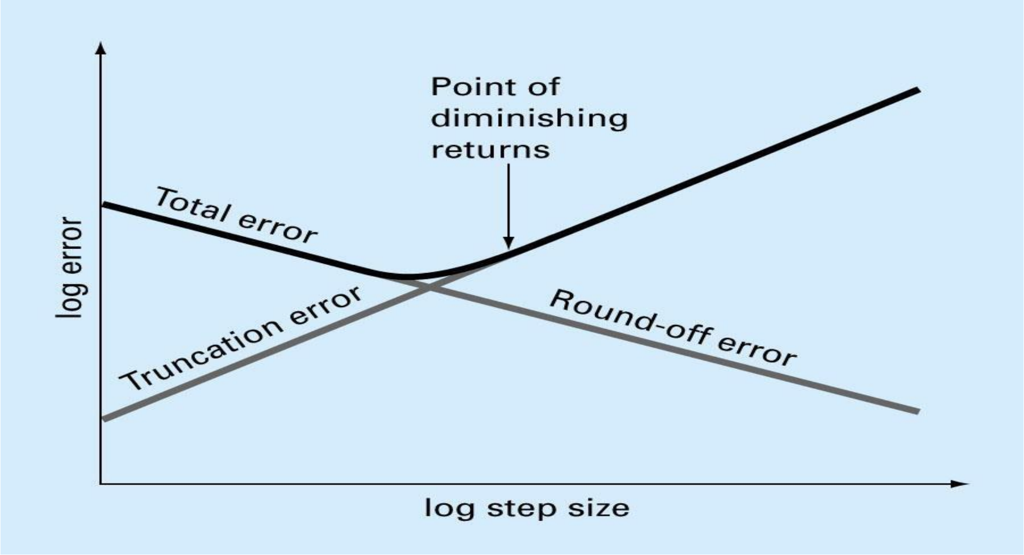
for h=0.1

j	x_j	$f'(x_j)$	$f(x_{j+1})$	**TRUE**	**Error**
0	**0**	**0**	**0**	**0**	**0**
1	**0.1**	**0.2**	**0.02**	**0.01**	**-0.01**
2	**0.2**	**0.4**	**0.06**	**0.04**	**-0.02**
3	**0.3**	**0.6**	**0.12**	**0.09**	**-0.03**
4	**0.4**	**0.8**	**0.2**	**0.16**	**-0.04**
5	**0.5**	**1**	**0.3**	**0.25**	**-0.05**
6	**0.6**	**1.2**	**0.42**	**0.36**	**-0.06**
7	**0.7**	**1.4**	**0.56**	**0.49**	**-0.07**
8	**0.8**	**1.6**	**0.72**	**0.64**	**-0.08**
9	**0.9**	**1.8**	**0.9**	**0.81**	**-0.09**
44	**4.4**	**8.8**	**19.8**	**19.36**	**-0.44**
45	**4.5**	**9**	**20.7**	**20.25**	**-0.45**

for h=0.5

j	x_j	$f'(x_j)$	$f(x_{j+1})$	TRUE	Error
0	0	0	0	0	0
1	0.5	1	0.5	0.25	-0.25
2	1	2	1.5	1	-0.5
3	1.5	3	3	2.25	-0.75
4	2	4	5	4	-1
5	2.5	5	7.5	6.25	-1.25
6	3	6	10.5	9	-1.5
7	3.5	7	14	12.25	-1.75
8	4	8	18	16	-2
9	4.5	9	22.5	20.25	-2.25

4. Total Numerical Error = Round-off error + Truncation Error



○ **How to reduce the total numerical error**

- Round-off error: by using double precision using IEEE data format

→ **You should declare the type of data (single or double) regardless of the computing machine (32-bit or 64-bit machines)**

- Truncation error: By using higher order approximation

By using smaller step size

Etc.

→ **limited by computer speed**

○ **Efficient Numerical methods can relieve many of these errors.**

○ **Some Terminology**

- **Overflow: Any number larger than the largest number that can be expressed on a computer will result in an overflow.**

  (Example): $\lim_{\varepsilon \to 0} \pm \frac{1}{\varepsilon} = \pm\infty$

- **Underflow (Hole) : Any positive number smaller than the smallest number that can be represented on a computer will result in an underflow.**

- **Stable Algorithm: In extended calculations, it is likely that many round-offs will be made. Each of these plays the role of an input error for the remainder of the computation, impacting the eventual output. Algorithms, with which the cumula tive effect of all such error is limited, so that a useful result is generated, are called "stable" algorithms. When accumula tion is devastating and the solution is overwhelmed by the error, such algorithms are called unstable.**

End of Lecture