MA633L-Numerical Analysis

Lecture 6 : Machine Epsilon and Floating Point System

Panchatcharam Mariappan[1]

[1]Associate Professor
Department of Mathematics and Statistics
IIT Tirupati, Tirupati

January 16, 2025

भारतीय प्रौद्योगिकी संस्थान तिरुपति
IIT
TIRUPATI

Recap

Error Definitions

Definition 1 (True Error, Absolute Error and Relative Error)

Suppose that $\tilde{a}$ is an approximation to the true value a.

$$E_t = a - \tilde{a},$$

$$E_{tabs} = |a - \tilde{a}|$$

$$\varepsilon_t = \frac{|a - \tilde{a}|}{|a|}, a \neq 0$$

If $\tilde{a}$ is the present approximation to the previous approximation a. The relative error is

$$\varepsilon_a = \frac{|a - \tilde{a}|}{|a|}, a \neq 0$$

$$\varepsilon_t \text{ or } \varepsilon_a < \varepsilon_s = (0.5 \times 10^{2-n})\%$$

We consider $n = 8$

Floating Point

Fixed Point System

- Every real number is represented by a finite or infinite sequence of decimal digits in a decimal notation.
- Most of the computers have two ways of representing numbers, called fixed point and floating point.
- In a fixed point system, the position of the decimal of the point is fixed. Also, all numbers are given with a fixed number of decimals after the decimal point. For example, number with 4 decimals, are 1.6250, 45.3903, 0.3393.
- However, fixed-point representations are not applicable in most scientific computations due to their limited range.

Some Constants

For example, let us consider the following: Avogadro number, Planck's constant, distance between earth and sun, light year and so on. Representing these constants are complicated in fixed point, however, it can be written in floating point as follows:

$$\text{Avogadro Constant} = 6.02214076 \times 10^{23}$$
$$\text{Planck's Constant} = 6.62607004 \times 10^{-34} m^2 kg/s$$
$$\text{Distance between Sun and Earth} = 1.4711 \times 10^{11} m$$
$$\text{Light Year} = 9.4605284 \times 10^{15} m$$

Floating Point

- In floating point system, the number of significant digits is kept fixed, whereas the decimal point is **floating**.
- Here, a significant digit of a number c is any given digit of c, except possibly for zeros to the left of the first nonzero digit;
- These zeros serve only to fix the position of the decimal point. Note that, any other zero is a significant digit of c.

$$12300, 1.2300, 0.0012300$$

all have 5 significant digits.
In the bunch of constants, we have used exponents to represent very large and very small numbers. That is a number 156.78 could be represented as 0.15678×10^3 in a floating point base-10 system.

Floating Point

In fact, theoretically, any nonzero number a can be written as

$$a = \pm m \times 10^n, \quad 0.1 \leq |m| < 1, \quad n \text{ integer} \tag{1}$$

in the floating point base-10 system. Here

$$a = \pm 0.d_1 d_2 \cdots d_k \times 10^n, \quad 0 \leq d_i \leq 9 \tag{2}$$

This is usually referred as normalized floating point.

Floating Point in Computers

Computers use only binary number, therefore, m is limited to k binary digits and $\overline{n}$ is limited, which gives the following representations, called as floating point base-2 system.

$$c = \pm b \times 2^{\overline{n}}, \quad b = 0.b_1 b_2 \cdots b_k, \quad b_1 > 0 \tag{3}$$

These numbers c is called k-digit binary machine numbers. Fraction parts m or b is called the mantissa. n or $\overline{n}$ are called exponent of a or c.
Note that, one can represent only finitely many numbers using this notation and they become less and less dense with increasing a. That is, there are as many numbers between 1 and 3 as there are between 1000 and 2020.

Floating Point

Example 2

List all floating-point numbers that can be represented in the form

$$c = \pm(0.b_1b_2b_3)_2 \times 2^{\pm k}$$

where $b_1, b_2, b_3, k \in \{0, 1\}$

Solution: There are two choices for $\pm$, two for b_1, two for b_2 and two for b_3. There are three choices for the exponent. Therefore, we have $2 \times 2 \times 2 \times 2 \times 3 = 48$ possible numbers.

Positive distinct numbers are $0, 1; \frac{3}{2}, \frac{1}{2}; \frac{7}{4}, \frac{5}{4}, \frac{3}{4}, \frac{1}{4}; \frac{7}{8}, \frac{5}{8}, \frac{3}{8}, \frac{1}{8}; \frac{7}{16}, \frac{5}{16}, \frac{3}{16}, \frac{1}{16};$

or we can write them as

$$0, \frac{1}{16}, \frac{2}{16}\frac{3}{16}, \frac{4}{16}, \frac{5}{16}, \frac{6}{16}, \frac{7}{16}, \frac{8}{16}, \frac{10}{16}, \frac{12}{16}, \frac{14}{16}; \frac{16}{16}, \frac{20}{16}, \frac{24}{16}, \frac{28}{16}$$

Floating Point

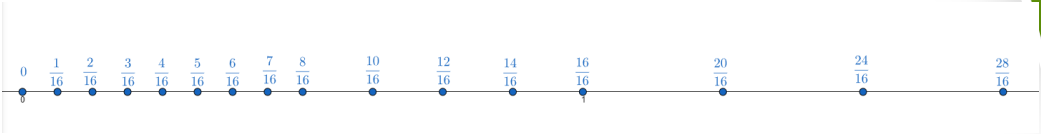

Figure 1: Example

Noramlized Floating Point

Example 3

List all floating-point numbers that can be represented in the form

$$c = \pm(0.b_1b_2b_3)_2 \times 2^{\pm k}$$

where $b_1 = 1, b_2, b_3, k \in \{0, 1\}$

Solution: There are two choices for $\pm$, two for b_1, two for b_2 and two for b_3. There are three choices for the exponent. Therefore, we have $2 \times 2 \times 2 \times 2 \times 3 = 48$ possible numbers.

Positive distinct numbers

$$0, \frac{4}{16}, \frac{5}{16}, \frac{7}{16}\frac{8}{16}, \frac{10}{16}, \frac{12}{16}, \frac{14}{16}; \frac{16}{16}, \frac{20}{16}, \frac{24}{16}, \frac{28}{16}$$

Floating Point

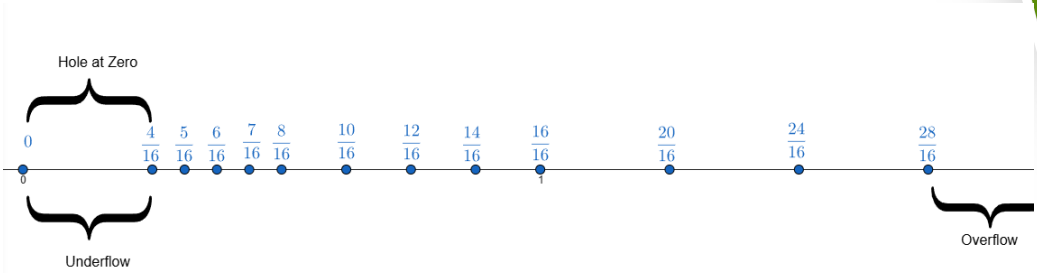

Figure 2: Example

Machine Epsilon

Definition 4 (Machine Epsilon)
Let ε be the smallest positive machine number with $1 + \varepsilon > 1$, also known as machine accuracy.

Then there are no numbers in intervals,

$$[1, 1 + \epsilon], [2, 2 + 2\epsilon], \cdots, [2025, 2025 + 2025\varepsilon]$$

So, when a computation produces an output $2025 + 2025\frac{\varepsilon}{2}$, then the computer will store it as either 2025 or 2025ε.
It is also called as interval machine epsilon.
What is the machine epsilon for the above examples?

Machine Epsilon

- It indicates the precision of the floating-point representation
- It measures how accurately numbers can be represented in a computer's floating-point format
- It is not possible to obtain greater accuracy

$$\varepsilon = 2^{-\text{bits used for magnitude of mantissa}}$$

If p is the precision, then $\varepsilon = 2^{-(p-1)}$

Rounding Machine Epsilon

Definition 5 (Rounding Machine Epsilon)
Let ϵ_R be the smallest positive machine number with $1 + \varepsilon_R > 1$, also known as machine accuracy.

However, it measures the rounding precision in the system. It is closely related to ϵ but focuses more on the error due to rounding during arithmetic operations. If p is the precision, then $\varepsilon_R = 2^{-p}$

Smallest Number

Definition 6 (Smallest (Normalized) Number)
The smallest number refers to the smallest positive normalized number that can be represented in a given floating-point system.

However, this is not necessarily the smallest number that can be expressed, as subnormal (denormal) numbers can also exist which are smaller than the smallest normalized number.

- It defines the limit below which underflow occurs
- It causes loss of precision or treated as zero
- It represents the lower bound of the range of representable valued

Largest Number

Definition 7 (Largest Number)
The largest number refers to the largest representable finite number that can be represented in a given floating-point system.

The upper limit beyond which the system cannot represent numbers. Values exceeding this treated as infinity or lead to overflow.

- It defines the limit above which overflow occurs
- Numbers larger than this will result in infinity (according to system)
- It represents the upper bound of the range of representable values

float

In 1985, the IEEE (Institute for Electrical and Electronic Engineers) published a report called Binary Floating Point Arithmetic Standard 754-1985. There are three formats for handling the binary and decimal floating point numbers as per IEEE754-2008 report, namely single, double and extended precision.
Float or Single Precision floating point: A 32-bit (binary digit) is used for a real number. The first is a sign indicator, denoted by s. This is followed by an 8-bit exponent, e, called exponents, and a 23-bit binary fraction, f, called mantissa. The base for the exponent is 2. The name is single precision as it occupies a single 32 bit register.

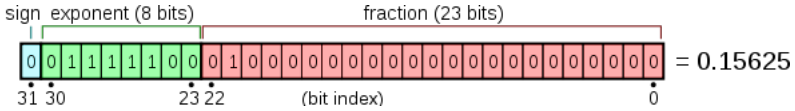

Figure 3: Float, Source: Wikipedia

float

The real value assumed by given float is

$$(-1)^{b_{31}} \times (1.b_{22}b_{21}\cdots b_0)_2 \times 2^{(b_{30}b_{29}b_{28}\cdots b_{23})_2 - 127}$$

which gives the value as

$$\text{value} = (-1)^{\text{sign}} \times \left(1 + \sum_{i=1}^{23} b_{23-i}2^{-i}\right) \times 2^{(e-127)}$$

float

Here

$$\text{sign} \in \{0,1\}, e = (b_{30}b_{29}b_{28}\cdots b_{23})_2 = \sum_{i=0}^{7} b_{23+i}2^i$$

$$e \in \{1,2,\cdots,2^8-2\} = \{1,2,\cdots,254\}$$

$$2^{e-127} \in \{2^{-126},2^{-125},\cdots,2^{127}\} = \{10^{-38},\cdots,10^{38}\}$$

$$(1.b_{22}b_{21}\cdots b_0)_2 = 1 + \sum_{i=1}^{23} b_{23-i}2^{-i}$$

$$(1.b_{22}b_{21}\cdots b_0)_2 \in \{1, 1+2^{-23},\cdots,2-2^{-23}\} \subset [1,2)$$

$$SP_\varepsilon = 2^{-23} = 1.1920929 \times 10^{-7}$$

$$SP_{\min} = 1.17549435 \times 10^{-38}$$

$$SP_{\max} = 1.70141183 \times 10^{38}$$

double

Double: A 64-bit (binary digit) is used for a real number. The first is a sign indicator, denoted by s. This is followed by an 11-bit exponent, e, called exponents, and a 52-bit binary fraction, f, called mantissa. The base for the exponent is 2. It requires two register and hence the name double.

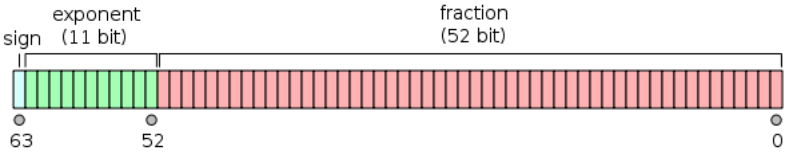

Figure 4: Double,Source: Wikipedia

double

The real value assumed by given double is

$$(-1)^{b_{63}} \times (1.b_{51}b_{50} \cdots b_0)_2 \times 2^{(b_{62}b_{61}b_{60} \cdots b_{52})_2 - 1023}$$

which gives the value as

$$\text{value} = (-1)^{\text{sign}} \times \left(1 + \sum_{i=1}^{52} b_{52-i} 2^{-i}\right) \times 2^{(e-1023)}$$

double

Here

$$\text{sign} \in \{0,1\}, e = (b_{62}b_{61}b_{60}\cdots b_{52})_2 = \sum_{i=0}^{10} b_{52+i}2^i$$

$$e \in \{1,2,\cdots,2^{11}-2\} = \{1,2,\cdots,2046\}$$

$$2^{e-1023} \in \{2^{-1022}, 2^{-1021}, \cdots, 2^{1023}\} = \{10^{-308},\cdots,10^{308}\}$$

$$(1.b_{51}b_{50}\cdots b_0)_2 = 1 + \sum_{i=1}^{52} b_{52-i}2^{-i}$$

$$(1.b_{51}b_{50}\cdots b_0)_2 \in \{1, 1+2^{-52}, \cdots, 2-2^{-52}\} \subset [1,2)$$

$$DP_\varepsilon = 2^{-52} = 2.220446 \times 10^{-16}$$

$$DP_{\min} = 2.2250 \times 10^{-308}$$

$$DP_{max} = 1.7977 \times 10^{308}$$

Quadruple

Quadruple Precision: A 128-bit (binary digit) is used for a real number. The first is a sign indicator, denoted s. This is followed by an 15-bit exponent, e, called exponents, and a 112-bit binary fraction, f, called mantissa. The base for the exponent is 2. It requires 4 registers.

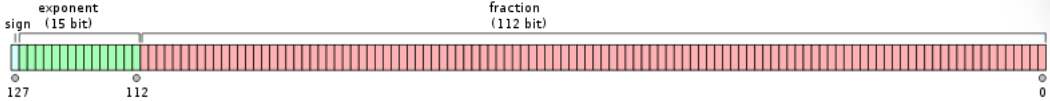

Figure 5: Quadruple, Source: Wikipedia

Quadruple

The real value assumed by given quadruple is

$$(-1)^{b_{127}} \times (1.b_{112}b_{111}\cdots b_0)_2 \times 2^{(b_{126}b_{125}\cdots b_{112})_2 - 16383}$$

which gives the value as

$$\text{value} = (-1)^{sign} \times \left(1 + \sum_{i=1}^{112} b_{112-i}2^{-i}\right) \times 2^{(e-16383)}$$

Quadruple

Here

$$\text{sign} \in \{0,1\}, e = (b_{126}b_{125}\cdots b_{112})_2 = \sum_{i=0}^{14} b_{112+i}2^i$$

$$e \in \{1,2,\cdots,2^{15}-2\} = \{1,2,\cdots,32766\}$$

$$2^{e-16383} \in \{2^{-16382}, 2^{-16381}, \cdots, 2^{16383}\} = \{10^{-4932}, \cdots, 10^{4932}\}$$

$$(1.b_{112}b_{111}\cdots b_0)_2 = 1 + \sum_{i=1}^{112} b_{112-i}2^{-i}$$

$$(1.b_{112}b_{111}\cdots b_0)_2 \in \{1, 1+2^{-112}, \cdots, 2-2^{-112}\} \subset [1,2)$$

$$QP_\varepsilon = 2^{-112} = 2.938736 \times 10^{-34}$$

$$QP_{\min} = 3.3621 \times 10^{-4932}$$

$$QP_{\max} = 1.1897 \times 10^{4932}$$

long double

Long Double: A 80-bit (binary digit) is used for a real number. The first is a sign indicator, denoted by s. This is followed by an 15-bit exponent, e, called exponents, and a 64-bit binary fraction, f, called mantissa. The base for the exponent is 2.

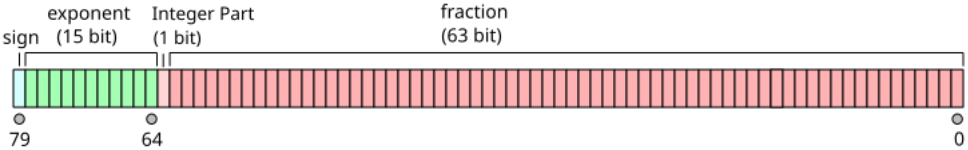

Figure 6: Long Double,Source: Wikipedia

Compute $LDP_{\varepsilon}, LDP_{\min}$ and $LDP_{\max}$

Thanks

Doubts and Suggestions

panch.m@iittp.ac.in

भारतीय प्रौद्योगिकी संस्थान तिरुपति
IIT
TIRUPATI

MA633L-Numerical Analysis

Lecture 6 : Machine Epsilon and Floating Point System

Panchatcharam Mariappan[1]

[1]Associate Professor
Department of Mathematics and Statistics
IIT Tirupati, Tirupati

January 16, 2025

भारतीय प्रौद्योगिकी संस्थान तिरुपति
IIT
TIRUPATI