Introductory Computational Mathematics

Semester 2, 2022

Dr Elliot Carr

Tarang Janawalkar

Contents

1 Preliminaries

1.1 Errors

Errors in calculations are a common problem in numerical analysis. We can quantify the magnitude of such an error by two measures.

Definition 1.1 (Absolute and relative error)**.** Let $\tilde{x}$ be an approximatiotn of x. Then the **absolute error** is given by

$$\text{absolute error} = |\tilde{x} - x|.$$

The **relative error** is given by

$$\text{relative error} = \frac{|\tilde{x} - x|}{|x|}.$$

It is important to realise that the absolute error can be misleading when comparing different sizes of errors, i.e., it is always small for small values of x and $\tilde{x}$.

1.2 Floating Point Arithmetic

The set of real numbers $\mathbb{R}$ contains uncountably many elements. Computers have a limited number of bits, and can therefore only represent a small subset of these elements.
The most common approximation of real arithmetic used in computers is known as **floating point arithmetic**.

Definition 1.2 (Floating point number system)**.** A floating point number system $\mathbb{F}(\beta, k, m, M)$ is a *finite subset* of the real number system characterised by the parameters:

- $\beta \in \mathbb{N}$: the base
- $k \in \mathbb{N}$: the number of digits in the significand
- $m \in \mathbb{Z}$: the minimum exponent
- $M \in \mathbb{Z}$: the maximum exponent

Definition 1.3 (Floating point numbers)**.** The floating point numbers $f \in \mathbb{F}(\beta, k, m, M)$ are real numbers expressible in the form

$$f = \pm (d_1.d_2d_3 \ldots d_k)_\beta \times \beta^e$$

where $e \in \mathbb{Z}$ is the **exponent** satisfying $m \leq e \leq M$. The quantity $d_1.d_2d_3 \ldots d_k$ is known as the **significand**, where d_i are base-β digits, with $d_1 \neq 0$ unless $f = 0$, to ensure a unique representation of f.

Computers primarily use floating point number systems with base $\beta = 2$ (binary), other common bases include $\beta = 10$ (decimal[1]) and $\beta = 16$ (hexadecimal).

[1]Note that for base-10, we do not need to include the subscript in the significand.

To illustrate the finiteness of the floating point number system, consider the following example:

$$
\begin{aligned}
\mathbb{F}(10,\ 3,\ -1,\ 1) = \{&0, \\
&\pm 1.00 \times 10^{-1}, \quad \pm 1.01 \times 10^{-1}, \quad \dots, \quad \pm 9.99 \times 10^{-1}, \\
&\pm 1.00 \times 10^{0}, \quad \pm 1.01 \times 10^{0}, \quad \dots, \quad \pm 9.99 \times 10^{0}, \\
&\pm 1.00 \times 10^{1}, \quad \pm 1.01 \times 10^{1}, \quad \dots, \quad \pm 9.99 \times 10^{1} \} \\
= \{&0, \\
&\pm 0.100, \quad \pm 0.101, \quad \dots, \quad \pm 0.999, \\
&\pm 1.00, \quad \pm 1.01, \quad \dots, \quad \pm 9.99, \\
&\pm 10.0, \quad \pm 10.1, \quad \dots, \quad \pm 99.9 \}
\end{aligned}
$$

Note that the numbers in this set are not equally spaced, (smaller spacing for smaller exponents).

Definition 1.4 (Overflow and underflow)**.** Consider the value $x \in \mathbb{R}$, if x is too small in magnitude to be represented in $\mathbb{F}$, an **underflow** occurs which typically causes the number to be replaced by zero.
Similarly, if x is too large in magnitude to be represented in $\mathbb{F}$, an **overflow** occurs which typically causes the number to be replaced by infinity.

Corollary 1.2.0.1. *The smallest and largest values (in magnitude) of* $\mathbb{F}$ *are given by*

$$
\begin{aligned}
\textit{smallest} &= \beta^{m} \\
\textit{largest} &= \left(1 - \beta^{-k}\right) \beta^{M+1}.
\end{aligned}
$$

The cardinality of the positive elements in $\mathbb{F}$*, denoted* $\mathbb{F}^{+}$*, is given by*

$$|\mathbb{F}^{+}| = (M - m + 1)(\beta - 1)\beta^{k-1}$$

so that by including negative numbers and zero, the cardinality of $\mathbb{F}$ *is given by*

$$|\mathbb{F}| = 2|\mathbb{F}^{+}| + 1.$$

1.2.1 Representing Real Numbers as Floating Point Numbers

If we wish to represent a real number[2] x that is not exactly representable in $\mathbb{F}$, we can **round** the number to the nearest *representable* number.
The error committed by this process is known as the **roundoff error**.

1.2.2 Converting between Floating Point Number Systems

Consider $fl : \mathbb{R} \to \mathbb{F}(\beta,\ k,\ m,\ M)$, defined as function which maps real numbers x to the nearest element in $\mathbb{F}$. To determine $fl(x)$:

1. Express x in base β.
2. Express x in scientific form.

[2] x must satisfy $\min(\mathbb{F}) \le x \le \max(\mathbb{F})$.

3. Verify that $m \leq e \leq M$:
   - If $e > M$, then $x = \infty$.
   - If $e < m$, then $x = 0$.
   - Otherwise, round the significand to k digits.

The relative error produced by rounding x to $fl\,(x)$ is bounded according to

$$\frac{|x - fl\,(x)|}{|x|} \leq \frac{1}{2}\beta^{1-k}.$$

Definition 1.5 (Unit roundoff)**.** The **unit roundoff** or **machine precision** u of a floating point number system $\mathbb{F}\,(\beta,\ k,\ m,\ M)$ is given by

$$u = \frac{1}{2}\beta^{1-k}.$$

1.2.3 IEEE Floating Point Standard

IEEE 754 is the standard for floating point arithmetic used by most modern computers.
It is a binary format, with several variants. The most common variant is **IEEE double precision**, which is based on $\mathbb{F}\,(2,\ 53,\ -1022,\ 1023)$.
The basic properties of this format are summarised in the following table.

Unit roundoff	$u = 1.11 \times 10^{-16}$
Largest representable positive number	1.80×10^{308}
Smallest representable positive number	2.23×10^{-308}
Special values	± 0, $\pm\infty$, `NaN`