CHAPTER 1 (4 LECTURES)
FLOATING POINT ARITHMETIC AND ERRORS

1. Numerical analysis

Numerical analysis, area of mathematics and computer science that creates, analyzes, and implements algorithms for obtaining numerical solutions to problems involving continuous variables. Such problems arise throughout the natural sciences, social sciences, engineering, medicine, and business. Since the mid 20th century, the growth in power and availability of digital computers has led to an increasing use of realistic mathematical models in science and engineering, and numerical analysis of increasing sophistication is needed to solve these more detailed models of the world. The formal academic area of numerical analysis ranges from quite theoretical mathematical studies to computer science issues. A major advantage for numerical technique is that a numerical answer can be obtained even when a problem has no analytical solution. However, result from numerical analysis is an approximation, in general, which can be made as accurate as desired. For example to find the approximate values of $\sqrt{2}$, π etc.

With the increasing availability of computers, the new discipline of scientific computing, or computational science, emerged during the 1980s and 1990s. The discipline combines numerical analysis, symbolic mathematical computations, computer graphics, and other areas of computer science to make it easier to set up, solve, and interpret complicated mathematical models of the real world.

1.1. **Common perspectives in numerical analysis.** Numerical analysis is concerned with all aspects of the numerical solution of a problem, from the theoretical development and understanding of numerical methods to their practical implementation as reliable and efficient computer programs. Most numerical analysts specialize in small subfields, but they share some common concerns, perspectives, and mathematical methods of analysis. These include the following:

- When presented with a problem that cannot be solved directly, they try to replace it with a "nearby problem" that can be solved more easily. Examples are the use of interpolation in developing numerical integration methods and root-finding methods.
- There is widespread use of the language and results of linear algebra, real analysis, and functional analysis (with its simplifying notation of norms, vector spaces, and operators).
- There is a fundamental concern with error, its size, and its analytic form. When approximating a problem, it is prudent to understand the nature of the error in the computed solution. Moreover, understanding the form of the error allows creation of extrapolation processes to improve the convergence behaviour of the numerical method.
- Numerical analysts are concerned with stability, a concept referring to the sensitivity of the solution of a problem to small changes in the data or the parameters of the problem. Numerical methods for solving problems should be no more sensitive to changes in the data than the original problem to be solved. Moreover, the formulation of the original problem should be stable or well-conditioned.

In this chapter, we introduce and discuss some basic concepts of scientific computing. We begin with discussion of floating-point representation and then we discuss the most fundamental source of imperfection in numerical computing namely roundoff errors. We also discuss source of errors and then stability of numerical algorithms.

2. Floating-point representation of numbers

Any real number is represented by an infinite sequence of digits. For example

$$\frac{8}{3} = 2.66666\cdots = \left(\frac{2}{10^1} + \frac{6}{10^2} + \frac{6}{10^3} + \dots\right) \times 10^1.$$

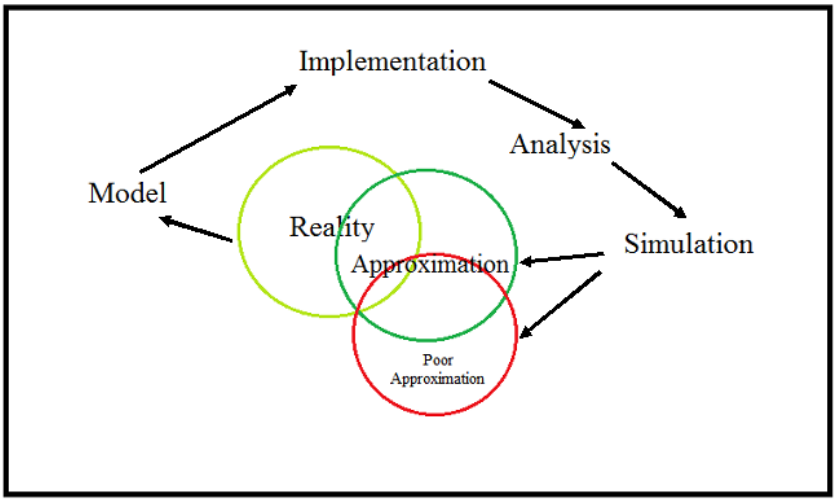

FIGURE 1. Numerical Approximations

This is an infinite series, but computer use an finite amount of memory to represent numbers. Thus only a finite number of digits may be used to represent any number, no matter by what representation method.
For example, we can chop the infinite decimal representation of $\frac{8}{3}$ after 4 digits,

$$\frac{8}{3} = \left(\frac{2}{10^1} + \frac{6}{10^2} + \frac{6}{10^3} + \frac{6}{10^4}\right) \times 10^1 = 0.2666 \times 10^1.$$

Generalizing this, we say that number has n decimal digits and call this n as precision.
For each real number x, we associate a floating point representation denoted by $fl(x)$, given by

$$fl(x) = \pm(0.a_1a_2 \ldots a_n)_\beta \times \beta^e,$$

here β based fraction is called mantissa with all a_i integers and e is known as exponent. This representation is called $\beta-$based floating point representation of x and we take base $\beta = 10$ in this course. For example,

$$\begin{aligned} 42.965 &= 4 \times 10^1 + 2 \times 10^0 + 9 \times 10^{-1} + 6 \times 10^{-2} + 5 \times 10^{-3} \\ &= 0.42965 \times 10^2. \\ -0.00234 &= -0.234 \times 10^{-2}. \end{aligned}$$

Number 0 is written as $0.00 \ldots 0 \times 10^e$. Likewise, we can use for binary number system and any real x can be written

$$x = \pm q \times 2^m$$

with $\frac{1}{2} \leq q \leq 1$ and some integer m. Both q and m will be expressed in terms of binary numbers. For example,

$$\begin{aligned} 1001.1101 &= 1 \times 2^3 + 2 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-4} \\ &= (9.8125)_{10}. \end{aligned}$$

Remark 2.1. *The above representation is not unique.*
For example, $0.2666 \times 10^1 = 0.02666 \times 10^2$ *etc.*

Definition 2.1 (Normal form)**.** *A non-zero floating-point number is in normal form if the values of mantissa lies in* $(-1, -0.1]$ *or* $[0.1, 1)$.

Therefore, we normalize the representation by $a_1 \neq 0$. Not only the precision is limited to a finite number of digits, but also the range of exponent is also restricted. Thus there are integers m and M such that $-m \leq e \leq M$.

Definition 2.2 (Overflow and underflow)**.** *An overflow is obtained when a number is too large to fit into the floating point system in use, i.e $e > M$. An underflow is obtained when a number is too small, i.e $e < -m$. When overflow occurs in the course of a calculation, this is generally fatal. But underflow is non-fatal: the system usually sets the number to* 0 *and continues. (Matlab does this, quietly.)*

2.1. **Rounding and chopping.** Let x be any real number and $fl(x)$ be its machine approximation. There are two ways to do the "cutting" to store a real number

$$x = \pm(0.a_1a_2\dots a_na_{n+1}\dots) \times 10^e, \quad a_1 \neq 0.$$

(1) Chopping: We ignore digits after a_n and write the number as following in chopping

$$fl(x) = (.a_1a_2\dots a_n) \times 10^e.$$

(2) Rounding: Rounding is defined as following

$$fl(x) = \begin{cases} \pm(0.a_1a_2\dots a_n) \times 10^e, \ 0 \leq a_{n+1} < 5 & \text{(rounding down)} \\ \pm[(0.a_1a_2\dots a_n) + (0.00\dots01)] \times 10^e, \ 5 \leq a_{n+1} < 10 & \text{(rounding up).} \end{cases}$$

Example 1.

$$fl\left(\frac{6}{7}\right) = \begin{cases} 0.86 \times 10^0 & \text{(rounding)} \\ 0.85 \times 10^0 & \text{(chopping).} \end{cases}$$

3. Errors in numerical approximations

Definition 3.1 (Absolute and relative error)**.** *If $fl(x)$ is the approximation to the exact value x, then the absolute error is $|x - fl(x)|$, and relative error is $\dfrac{|x - fl(x)|}{|x|}$.*

Remark: As a measure of accuracy, the absolute error may be misleading and the relative error is more meaningful.

Example 2. *Find the largest interval in which $fl(x)$ must lie to approximate $\sqrt{2}$ with relative error at most 10^{-5} for each value of x.*

Sol. We have

$$\left|\frac{\sqrt{2} - fl(x)}{\sqrt{2}}\right| \leq \sqrt{2} \cdot 10^{-5}.$$

Therefore

$$\begin{aligned} |\sqrt{2} - fl(x)| &\leq \sqrt{2} \cdot 10^{-5}, \\ -\sqrt{2} \cdot 10^{-5} &\leq \sqrt{2} - fl(x) \leq \sqrt{2} \cdot 10^{-5} \\ -\sqrt{2} - \sqrt{2} \cdot 10^{-5} &\leq -fl(x) \leq -\sqrt{2} + \sqrt{2} \cdot 10^{-5} \\ \sqrt{2} + \sqrt{2} \cdot 10^{-5} &\geq fl(x) \geq \sqrt{2} - \sqrt{2} \cdot 10^{-5}. \end{aligned}$$

Hence interval (in decimals) is $[1.4141994\cdots,\ 1.4142277\cdots]$.

3.1. **Chopping and Rounding Errors.** Let x be any real number we want to represent in a computer. Let $fl(x)$ be the representation of x in the computer then what is largest possible values of $\dfrac{|x - fl(x)|}{|x|}$? In the worst case, how much data we are losing due to round-off errors or chopping errors?

Chopping errors: Let

$$\begin{aligned} x &= (0.a_1a_2\dots a_na_{n+1}\dots) \times 10^e \\ &= \left(\frac{a_1}{10} + \frac{a_2}{10^2} + \cdots + \frac{a_n}{10^n} + \frac{a_{n+1}}{10^{n+1}} + \cdots\right) \\ &= \left(\sum_{i=1}^{\infty} \frac{a_i}{10^i}\right) \times 10^e, \ a_1 \neq 0, \\ fl(x) &= (0.a_1a_2\dots a_n) \times 10^e = \left(\sum_{i=1}^{n} \frac{a_i}{10^i}\right) \times 10^e. \end{aligned}$$

Therefore

$$|x - fl(x)| = \left(\sum_{i=n+1}^{\infty} \frac{a_i}{10^i} \right) \times 10^e$$

Now since each $a_i \leq 9 = 10 - 1$, therefore,

$$\begin{aligned} |x - fl(x)| &\leq \sum_{i=n+1}^{\infty} \frac{10-1}{10^i} \times 10^e \\ &= (10-1) \left[\frac{1}{10^{n+1}} + \frac{1}{10^{n+2}} + \dots \right] \times 10^e \\ &= (10-1)) \left[\frac{\frac{1}{10^{n+1}}}{1 - \frac{1}{10}} \right] \times 10^e \\ &= 10^{e-n}. \end{aligned}$$

Therefore absolute error bound is

$$E_a = |x - fl(x)| \leq 10^{e-n}.$$

Now

$$|x| = (0.a_1 a_2 \dots a_n)_{10} \times 10^e \geq 0.1 \times 10^e = \frac{1}{10} \times 10^e.$$

Therefore relative error bound is

$$E_r = \frac{|x - fl(x)|}{|x|} \leq \frac{10^{-n} \times 10^e}{10^{-1} \times 10^e} = 10^{1-n}.$$

Rounding errors: For rounding

$$fl(x) = \begin{cases} (0.a_1 a_2 \dots a_n)_{10} \times 10^e = \left(\sum_{i=1}^{n} \frac{a_i}{10^i} \right) \times 10^e, \ 0 \leq a_{n+1} < 5 \\ (0.a_1 a_2 \dots a_{n-1}[a_n + 1])_{10} \ 10^e = \left(\frac{1}{10^n} + \sum_{i=1}^{n} \frac{a_i}{10^i} \right) \times 10^e, \ 5 \leq a_{n+1} < 10. \end{cases}$$

For $0 < a_{n+1} < 5 = 10/2$,

$$\begin{aligned} |x - fl(x)| &= \sum_{i=n+1}^{\infty} \frac{a_i}{10^i} \times 10^e \\ &= \left[\frac{a_{n+1}}{10^{n+1}} + \sum_{i=n+2}^{\infty} \frac{a_i}{10^i} \right] \times 10^e \\ &\leq \left[\frac{10/2 - 1}{10^{n+1}} + \sum_{i=n+2}^{\infty} \frac{(10-1)}{10^i} \right] \times 10^e \\ &= \left[\frac{10/2 - 1}{10^{n+1}} + \frac{1}{10^{n+1}} \right] \times 10^e \\ &= \frac{1}{2} 10^{e-n}. \end{aligned}$$

For $5 \leq a_{n+1} < 10$,

$$\begin{aligned} |x - fl(x)| &= \left| \sum_{i=n+1}^{\infty} \frac{a_i}{10^i} - \frac{1}{10^n} \right| \times 10^e \\ &= \left| \frac{1}{10^n} - \sum_{i=n+1}^{\infty} \frac{a_i}{10^i} \right| \times 10^e \\ &= \left| \frac{1}{10^n} - \frac{a_{n+1}}{10^{n+1}} - \sum_{i=n+2}^{\infty} \frac{a_i}{10^i} \right| \times 10^e \\ &\leq \left| \frac{1}{10^n} - \frac{a_{n+1}}{10^{n+1}} \right| \times 10^e \end{aligned}$$

Since $-a_{n+1} \leq -10/2$, therefore

$$\begin{aligned} |x - fl(x)| &\leq \left| \frac{1}{10^n} - \frac{10/2}{10^{n+1}} \right| \times 10^e \\ &= \frac{1}{2} 10^{e-n}. \end{aligned}$$

Therefore, for both cases absolute error bound is

$$E_a = |x - fl(x)| \leq \frac{1}{2}\ 10^{e-n}.$$

Also relative error bound is

$$E_r = \frac{|x - fl(x)|}{|x|} \leq \frac{1}{2}\ \frac{10^{-n} \times 10^e}{10^{-1} \times 10^e} = \frac{1}{2} 10^{1-n} = 5 \times 10^{-n}.$$

4. Significant Figures

The term significant digits is often used to loosely describe the number of decimal digits that appear to be accurate. The definition is more precise, and provides a continuous concept.
Looking at an approximation 2.75303 to an actual value of 2.75194, we note that the three most significant digits are equal, and therefore one may state that the approximation has three significant digits of accuracy. One problem with simply looking at the digits is given by the following two examples:

(1) 1.9 as an approximation to 1.1 may appear to have one significant digit, but with a relative error of 0.73, this seems unreasonable.
(2) 1.9999 as an approximation to 2.0001 may appear to have no significant digits, but the relative error is 0.00010 which is almost the same relative error as the approximation 1.9239 is to 1.9237.

Thus, we need a more mathematical definition of the number of significant digits. Let the number x and approximation x^* be written in decimal form. The number of significant digits tells us to about how many positions x and x^* agree. More precisely, we say that x^* has m significant digits of x if the absolute error $|x - x^*|$ has zeros in the first m decimal places, counting from the leftmost nonzero (leading) position of x, followed by a digit from 0 to 5.
Examples:
5.1 has 1 significant digit of 5: $|5 - 5.1| = \mathbf{0}.1$.
0.51 has 1 significant digits of 0.5: $|0.5 - 0.51| = 0.\mathbf{0}1$.
4.995 has 3 significant digits of 5: $5 - 4.995 = \mathbf{0.00}5$.
4.994 has 2 significant digits of 5: $5 - 4.994 = \mathbf{0.0}06$.
0.57 has all significant digits of 0.57.
1.4 has 0 significant digits of 2: $2 - 1.4 = 0.6$.
In the terms of relative errors, the number x^* is said to approximate x to m significant digits (or figures) if m is the largest nonnegative integer for which

$$\frac{|x - x^*|}{|x|} \leq 0.5 \times 10^{-m}.$$

If the relative error is greater than 0.5, then we will simply state that the approximation has zero significant digits.

For example, if we approximate π with 3.14 then relative errors is

$$E_r = \frac{|\pi - 3.14|}{\pi} \approx 0.00051 \leq 0.005 = 0.5 \times 10^{-2},$$

and therefore it is correct to two significant digits.
Also 4.994 has 2 significant digits of 5 as relative errors is $(5 - 4.994)/5 = 0.0012 = 0.12 \times 10^{-2} \leq 0.5 \times 10^{-2}$.
Some numbers are exact because they are known with complete certainty. Most exact numbers are integers: exactly 12 inches are in a foot, there might be exactly 23 students in a class. Exact numbers can be considered to have an infinite number of significant figures.

5. Rules for mathematical operations

In carrying out calculations, the general rule is that the accuracy of a calculated result is limited by the least accurate measurement involved in the calculation. In addition and subtraction, the result is rounded off so that it has the same number of digits as the measurement having the fewest decimal places (counting from left to right). For example,
100 (assume 3 significant figures) +23.643 (5 significant figures) = 123.643,
which should be rounded to 124 (3 significant figures).
In addition to inaccurate representation of numbers, the arithmetic performed in a computer is not exact. The arithmetic involves manipulating binary digits by various shifting, or logical, operations.

Let the floating-point representations $fl(x)$ and $fl(y)$ are given for the real numbers x and y and that the symbols $\oplus$, $\ominus$, $\otimes$ and $\oslash$ represent machine addition, subtraction, multiplication, and division operations, respectively. We will assume a finite-digit arithmetic given by

$$x \oplus y = fl(fl(x) + fl(y)), \qquad x \ominus y = fl(fl(x) - fl(y)),$$
$$x \otimes y = fl(fl(x) \times fl(y)), \qquad x \oslash y = fl(fl(x) \div fl(y)).$$

This arithmetic corresponds to performing exact arithmetic on the floating-point representations of x and y and then converting the exact result to its finite-digit floating-point representation.

Example 3. *Suppose that $x = \frac{5}{7}$ and $y = \frac{1}{3}$. Use five-digit chopping for calculating $x + y$, $x - y$, $x \times y$, and $x \div y$.*

Sol. Here $x = \frac{5}{7} = 0.714285\cdots$ and $y = \frac{1}{3} = 0.33333\cdots$.
Using the five-digit chopping values of x and y are

$$fl(x) = 0.71428 \times 10^0 \quad \text{and} \quad fl(y) = 0.33333 \times 10^0.$$

Thus,

$$x \oplus y = fl(fl(x) + fl(y)) = fl(0.71428 \times 10^0 + 0.33333 \times 10^0) = fl(1.04761 \times 10^0) = 0.10476 \times 10^1.$$

The true value is $x + y = \frac{5}{7} + \frac{1}{3} = \frac{22}{21}$, so we have
Absolute Error $E_a = |\frac{22}{21} - 0.10476 \times 10^1| = 0.190 \times 10^{-4}$.
Relative Error $E_r = \dfrac{0.190 \times 10^{-4}}{\frac{22}{21}} = 0.182 \times 10^{-4}$.
Similarly we can perform other calculations.
Now we find some rules for absolute and relative errors while we do addition/subtraction or multiplication/division.
Further we show some examples of arithmetic with different exponents.

Example 4. *Add the following floating-point numbers* $0.4546e3$ *and* $0.5433e7$.

Sol. This problem contains unequal exponent. To add these floating-point numbers, take operands with the largest exponent as,

$$0.5433e7 + 0.0000e7 = 0.5433e7.$$

(Because $0.4546e3$ changes in the same operand as $0.0000e7$).

Example 5. *Subtract the following floating-point numbers:*
1. $0.5424e - 99$ *from* $0.5452e - 99$
2. $0.3862e - 7$ *from* $0.9682e - 7$

Sol. On subtracting we get $0.0028e-99$. Again this is a floating-point number but not in the normalized form. To convert it in normalized form, shift the mantissa to the left. Therefore we get $0.28e-101$. This condition is called an underflow condition.
Similarly, after subtraction we get $0.5820e-7$.

Example 6. *Multiply the following floating point numbers:* $0.1111e74$ *and* $0.2000e80$.

Sol. On multiplying we obtain $0.1111e74 \times 0.2000e80 = 0.2222e153$. This shows overflow condition of normalized floating-point numbers.

Example 7. *The error in the measurement of area of a circle is not allowed to exceed* 0.5%. *How accurately the radius should be measured.*

Sol. Area of the circle is $A = \pi r^2$ (say).

$$\therefore \frac{\partial A}{\partial r} = 2\pi r.$$

Percentage error in $A = \dfrac{\delta A}{A} \times 100 = 0.5$

Therefore $\delta A = \dfrac{0.5}{100} \times A = 1/200\pi r^2$

Percentage error in $r = \dfrac{\delta r}{r} \times 100 = \dfrac{100}{r}\dfrac{\delta A}{\frac{\partial A}{\partial r}} = 0.25.$

6. Loss of significance

Roundoff errors are inevitable and difficult to control. Other types of errors which occur in computation may be under our control. The subject of numerical analysis is largely preoccupied with understanding and controlling errors of various kinds.
One of the most common error-producing calculations involves the cancellation of significant digits due to the subtractions nearly equal numbers (or the addition of one very large number and one very small number or multiplication of a small number with a quite large number).
The phenomenon can be illustrated with the following examples.

Example 8. *If* $x = 0.3721478693$ *and* $y = 0.3720230572$. *What is the relative error in the computation of* $x - y$ *using five decimal digits of accuracy?*

Sol. We can compute with ten decimal digits of accuracy and can take it as 'exact'.

$$x - y = 0.0001248121.$$

Both x and y will be rounded to five digits before subtraction. Thus

$$fl(x) = 0.37215$$
$$fl(y) = 0.37202.$$
$$fl(x) - fl(y) = 0.13000 \times 10^{-3}.$$

Relative error, therefore is

$$E_r = \frac{(x-y) - (fl(x) - fl(y))}{x - y} \approx .04\% = 4\%.$$

Example 9. *Use four-digit rounding arithmetic and the formula for the roots of a quadratic equation, to find the most accurate approximations to the roots of the following quadratic equation. Compute the absolute and relative errors.*

$$1.002x^2 + 11.01x + 0.01265 = 0.$$

Sol. The quadratic formula states that the roots of $ax^2 + bx + c = 0$ are

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

Using the above formula, the roots of given eq. $1.002x^2 + 11.01x + 0.01265 = 0$ are approximately (using long format)

$$x_1 = -0.00114907565991, \quad x_2 = -10.98687487643590.$$

We use four-digit rounding arithmetic to find approximations to the roots. We write the approximations of root as x_1^* and x_2^*. These approximations are given by

$$\begin{aligned} x_{1,2}^* &= \frac{-11.01 \pm \sqrt{(-11.01)^2 - 4 \cdot 1.002 \cdot 0.01265}}{2 \cdot 1.002} \\ &= \frac{-11.01 \pm \sqrt{121.2 - 0.05070}}{2.004} \\ &= \frac{-11.01 \pm 11.00}{2.004} \end{aligned}$$

Therefore we find the first root:

$$x_1^* = -0.004990,$$

which has the absolute error $|x_1 - x_1^*| = 0.00384095$ and relative error $|x_1 - x_1^*|/|x_1| = 3.34265968$ (very high).

We find the second root

$$x_2^* = \frac{-11.01 - 11.00}{2.004} = -10.98,$$

which has the following absolute error

$$|x_2 - x_2^*| = 0.006874876,$$

and relative error

$$\frac{|x_2 - x_2^*|}{|x_2|} = 0.000626127.$$

This quadratic formula for the calculation of first root, encounter the subtraction of nearly equal numbers and cause loss of significance. Therefore, we use the alternate quadratic formula by rationalize the expression to calculate x_1 and approximation is given by

$$x_1^* = \frac{-2c}{b + \sqrt{b^2 - 4ac}} = -0.001149,$$

which has the following relative error

$$\frac{|x_1 - x_1^*|}{|x_1|} = 6.584 \times 10^{-5}.$$

Example 10. *The quadratic formula is used for computing the roots of equation* $ax^2+bx+c = 0,\ a \neq 0$ *and roots are given by*

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

Consider the equation $x^2 + 62.10x + 1 = 0$ *and discuss the numerical results.*

Sol. Using quadratic formula and 8-digit rounding arithmetic, we obtain two roots

$$x_1 = -.01610723$$

$$x_2 = -62.08390.$$

We use these values as "exact values". Now we perform calculations with 4-digit rounding arithmetic. We have $\sqrt{b^2 - 4ac} = \sqrt{62.10^2 - 4.000} = \sqrt{3856 - 4.000} = 62.06$ and

$$fl(x_1) = \frac{-62.10 + 62.06}{2.000} = -0.02000.$$

The relative error in computing x_1 is

$$\frac{|fl(x_1) - x_1|}{|x_1|} = \frac{|-0.02000 + .01610723|}{|-0.01610723|} = 0.2417.$$

In calculating x_2,

$$fl(x_2) = \frac{-62.10 - 62.06}{2.000} = -62.10.$$

The relative error in computing x_2 is

$$\frac{|fl(x_2) - x_2|}{|x_2|} = \frac{|-62.10 + 62.08390|}{|-62.08390|} = 0.259 \times 10^{-3}.$$

In this equation since $b^2 = 62.10^2$ is much larger than $4ac = 4$. Hence b and $\sqrt{b^2 - 4ac}$ become two equal numbers. Calculation of x_1 involves the subtraction of nearly two equal numbers but x_2 involves the addition of the nearly equal numbers which will not cause serious loss of significant figures.
To obtain a more accurate 4-digit rounding approximation for x_1, we change the formulation by rationalizing the numerator, that is,

$$x_1 = \frac{-2c}{b + \sqrt{b^2 - 4ac}}.$$

Then

$$fl(x_1) = \frac{-2.000}{62.10 + 62.06} = -2.000/124.2 = -0.01610.$$

The relative error in computing x_1 is now reduced to 0.62×10^{-3}.

Note: However, if rationalize the numerator in x_2 to get

$$x_2 = \frac{-2c}{b - \sqrt{b^2 - 4ac}}.$$

The use of this formula results not only involve the subtraction of two nearly equal numbers but also division by the small number. This would cause degrade in accuracy.

$$fl(x_2) = \frac{-2.000}{62.10 - 62.06} = -2.000/.04000 = -50.00$$

The relative error in x_2 becomes 0.19.
Nested Arithmetic: Accuracy loss due to round-off error can also be reduced by rearranging calculations, as shown in the next example. Polynomials should always be expressed in nested form before performing an evaluation, because this form minimizes the number of arithmetic calculations. One way to reduce round-off error is to reduce the number of computations.

Example 11. *Evaluate* $f(x) = 1.5 + 3.2x - 6.1x^2 + x^3$ *at* $x = 4.71$ *using three-digit arithmetic directly and with nesting.*

Sol. The exact result of the evaluation is (by taking more digits):
Exact: $f(4.71) = 1.5 + 3.2 \times 4.71 - 6.1 \times 4.71^2 + 4.71^3 = -14.263899$.
Now using three-digit rounding arithmetic, we obtain

$$\begin{aligned} f(4.71) &= 1.5 + 3.2 \times 4.71 - 6.1 \times 4.71^2 + 4.71^3 \\ &= 1.5 + 15.1 - 6.1 \times 22.2 + 22.2 \times 4.71 \\ &= 1.5 + 15.1 - 135 + 105 = -13.4. \end{aligned}$$

Similarly if we use three-digit chopping then

$$\begin{aligned} f(4.71) &= 1.5 + 3.2 \times 4.71 - 6.1 \times 4.71^2 + 4.71^3 \\ &= 1.5 + 15.0 - 6.1 \times 22.1 + 22.1 \times 4.71 \\ &= 1.5 + 15.0 - 134 + 104 = -13.5. \end{aligned}$$

The relative error in case of three-digit (rounding) is

$$\left| \frac{-14.263899 + 13.4}{-14.263899} \right| \approx 0.06,$$

and for three-digit (chopping) is

$$\left| \frac{-14.263899 + 13.5}{-14.263899} \right| \approx 0.05.$$

As an alternative approach, we write the polynomial $f(x)$ in a nested manner as

$$f(x) = 1.5 + x(3.2 + x(-6.1 + x)).$$

Using three-digit chopping arithmetic now produces

$$\begin{aligned} f(4.71) &= 1.5 + 4.71(3.2 + 4.71(-6.1 + 4.71)) \\ &= 1.5 + 4.71(3.2 + 4.71(-1.39)) = 1.5 + 4.71(3.2 - 6.54) \\ &= 1.5 + 4.71(-3.34) = 1.5 - 15.7 = -14.2. \end{aligned}$$

In a similar manner, we can obtain a three-digit rounding and answer is -14.3.
The relative error in case of three-digit (chopping) is

$$\left|\frac{-14.263899 + 14.2}{-14.263899}\right| \approx 0.0045,$$

and for three-digit (rounding) is

$$\left|\frac{-14.263899 + 14.3}{-14.263899}\right| \approx 0.0025.$$

Nesting has reduced the relative errors for both the approximations.

Example 12. *How to evaluate $y \approx x - \sin x$, when x is small.*

Sol. Since $x \approx \sin x$, x is small. This will cause loss of significant figures. Alternatively, if we use Taylor series for $\sin x$, we obtain

$$\begin{aligned} y &= x - (x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots) \\ &= \frac{x^3}{6} - \frac{x^5}{6 \times 20} + \frac{x^7}{6 \times 20 \times 42} - \dots \\ &= \frac{x^3}{6}\left[1 - \frac{x^2}{20}(1 - \frac{x^2}{42}(1 - \frac{x^2}{72})(...))\right]. \end{aligned}$$

7. Algorithms and Stability

An algorithm is a procedure that describes, in an unambiguous manner, a finite sequence of steps to be performed in a specified order. The object of the algorithm is to implement a procedure to solve a problem or approximate a solution to the problem. One criterion we will impose on an algorithm whenever possible is that small changes in the initial data produce correspondingly small changes in the final results. An algorithm that satisfies this property is called stable; otherwise it is unstable. Some algorithms are stable only for certain choices of initial data, and are called conditionally stable. The words condition and conditioning are used to indicate how sensitive the solution of a problem may be to small changes in the input data. A problem is well-conditioned if small changes in the input data can produce only small changes in the results. On the other hand, a problem is ill-conditioned if small changes in the input data can produce large changes in the output.
For a certain types of problems, a condition number can be defined. If that number is large, it indicates an ill-conditioned problem. In contrast, if the number is modest, the problem is recognized as a well-conditioned problem.
The condition number can be calculated in the following manner:

$$\begin{aligned} \kappa &= \frac{\text{relative change in output}}{\text{relative change in input}} \\ &= \frac{\left|\dfrac{f(x) - f(x^*)}{f(x)}\right|}{\left|\dfrac{x - x^*}{x}\right|} \\ &\approx \left|\frac{x f'(x)}{f(x)}\right|. \end{aligned}$$

For example, if $f(x) = \dfrac{10}{1-x^2}$, then the condition number can be calculated as

$$\kappa = \left|\frac{xf'(x)}{f(x)}\right| = \frac{2x^2}{|1-x^2|}.$$

Condition number can be quite large for $|x| \approx 1$. Therefore, the function is ill-conditioned.

Example 13. *Compute and interpret the condition number for*
(a) $f(x) = \sin x$ *for* $x = 0.51\pi$.
(b) $f(x) = \tan x$ *for* $x = 1.7$.

Sol. (a) The condition number is given by

$$\kappa = \left|\frac{xf'(x)}{f(x)}\right|.$$

For $x = 0.51\pi$, $f'(x) = \cos(0.51\pi) = -0.03141$, $f(x) = \sin(0.51\pi) = 0.99951$.

$$\therefore \kappa = 0.05035.$$

Since, the condition number is < 1, we conclude that the relative error is attenuated.
(b) $f(x) = \tan x$, $f(1.7) = -7.6966$, $f'(x) = 1/\cos^2 x$, $f'(a) = 1/\cos^2(1.7) = 60.2377$.

$$\kappa = -13.305.$$

Thus, the function is ill-conditioned.

In the following we study an example to create a stable algorithm.

7.1. **Creating Algorithms.** Another theme that occurs repeatedly in numerical analysis is the distinction between numerical algorithms are stable and those that are not. Informally speaking, a numerical process is unstable if small errors made at one stage of the process are magnified and propagated in subsequent stages and seriously degrade the accuracy of the overall calculation.
An algorithm can be thought of as a sequence of problems, i.e. a sequence of function evaluations. In this case we consider the algorithm for evaluating $f(x)$ to consist of the evaluation of the sequence $x_1, x_2, \cdots, x_n$. We are concerned with the condition of each of the functions $f_1(x_1), f_2(x_2), \cdots, f_{n-1}(x_{n-1})$ where $f(x) = f_i(x_i)$ for all i. An algorithm is unstable if any f_i is ill-conditioned, i.e. if any $f_i(x_i)$ has condition much worse than $f(x)$.

Example 14. *Write an algorithm to calculate the expression* $f(x) = \sqrt{x+1} - \sqrt{x}$, *when* x *is quite large. By considering the condition number* κ *of the subproblem of evaluating the function, show that such a function evaluation is not stable. Suggest a modification which makes it stable.*

Sol. Consider

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

so that there is potential loss of significance when x is large. Taking $x = 12345$ as an example, one possible algorithm is

$$\begin{aligned} x_0 : &= x = 12345 \\ x_1 : &= x_0 + 1 \\ x_2 : &= \sqrt{x}_1 \\ x_3 : &= \sqrt{x}_0 \\ f(x) := x_4 : &= x_2 - x_3. \end{aligned}$$

The loss of significance occurs with the final subtraction. We can rewrite the last step in the form $f_3(x_3) = x_2 - x_3$ to show how the final answer depends on x_3. As $f_3'(x_3) = -1$, we have the condition

$$\kappa(x_3) = \left|\frac{x_3 f_3'(x_3)}{f_3(x_3)}\right| = \left|\frac{x_3}{x_2 - x_3}\right|$$

from which we find $\kappa(x_3) \approx 2.2 \times 10^4$ when $x = 12345$. Note that this is the condition of a subproblem arrived at during the algorithm. To find an alternative algorithm we write

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

This suggests the algorithm

$$\begin{aligned} x_0 : &= x = 12345 \\ x_1 : &= x_0 + 1 \\ x_2 : &= \sqrt{x_1} \\ x_3 : &= \sqrt{x_0} \\ x_4 : &= x_2 + x_3 \\ f(x) := x_5 : &= 1/x_4. \end{aligned}$$

In this case $f_3(x_3) = 1/(x_2 + x_3)$ giving a condition for the subproblem of

$$\kappa(x_3) = \left|\frac{x_3 f_3'(x_3)}{f_3(x_3)}\right| = \left|\frac{x_3}{x_2 + x_3}\right|,$$

which is approximately 0.5 when $x = 12345$, and indeed in any case where x is much larger than 1. Thus first algorithm is unstable and second is stable for large values of x. In general such analyses are not usually so straightforward but, in principle, stability can be analysed by examining the condition of a sequence of subproblems.

Example 15. *Write an algorithm to calculate the expression* $f(x) = \sin(a + x) - \sin a$, *when* $x = 0.0001$. *By considering the condition number* κ *of the subproblem of evaluating the function, show that such a function evaluation is not stable. Suggest a modification which makes it stable.*

Sol. Let $x = 0.0001$

$$\begin{aligned} x_0 &= 0.0001 \\ x_1 &= a + x_0 \\ x_2 &= \sin x_1 \\ x_3 &= \sin a \\ x_4 &= x_2 - x_3. \end{aligned}$$

Now to check the effect of x_3 on x_2, we consider the function $f_3(x_3) = x_2 - x_3$

$$\kappa(x_3) = \left|\frac{x_3 f'(x_3)}{f(x_3)}\right| = \left|\frac{x_3}{x_2 - x_3}\right|$$

We obtain a very larger condition number, which shows that the last step is not stable.
Now we modify the above algorithm. We write the equivalent form

$$f(x) = \sin(a + x) - \sin a = 2\sin(x/2)\cos(a + x/2).$$

The modified algorithm is the following

$$\begin{aligned} x_0 &= 0.0001 \\ x_1 &= x_0/2 \\ x_2 &= \sin x_1 \\ x_3 &= \cos(a + x_1) \\ x_4 &= 2x_2x_3. \end{aligned}$$

Now we consider the function $f_3(x_3) = 2x_2x_3$,

$$\kappa(x_3) = \left|\frac{x_3}{x_2 - x_3}\right| = 1.$$

Thus the condition number is quite good, so this form is acceptable.

Remarks

(1) Accuracy tells us the closeness of computed solution to true solution of problem. Accuracy depends on conditioning of problem as well as stability of algorithm.
(2) Stability alone does not guarantee accurate results. Applying stable algorithm to well-conditioned problem yields accurate solution. Inaccuracy can result from applying stable algorithm to ill-conditioned problem or unstable algorithm to well-conditioned problem.

Exercises

(1) Compute the absolute error and relative error in approximations of x by x^*.
a. $x = \pi,\ x^* = 22/7$ b. $x = \sqrt{2},\ x^* = 1.414$ c. $x = 8!,\ x^* = 39900$.
(2) Find the largest interval in which x^* must lie to approximate x with relative error at most 10^{-4} for each value of x.
a. π b. e c. $\sqrt{3}$ d. $\sqrt[3]{7}$.
(3) A rectangular parallelepiped has sides of length 3 cm, 4 cm, and 5 cm, measured to the nearest centimeter. What are the best upper and lower bounds for the volume of this parallelepiped? What are the best upper and lower bounds for the surface area?
(4) Use three-digit rounding arithmetic to perform the following calculations. Compute the absolute error and relative error with the exact value determined to at least five digits.
a. $\sqrt{3} + (\sqrt{5} + \sqrt{7})$ b. $(121 - 0.327) - 119$ c. $-10\pi + 6e - \dfrac{3}{62}$ d. $\dfrac{\pi - 22/7}{1/17}$.
(5) Find the relative error in taking the difference of numbers $\sqrt{5.5} = 2.345$ and $\sqrt{6.1} = 2.470$. Numbers should be correct to four significant figures.
(6) Associative and distributive laws are not always valid in case of normalized floating-point representation.
i. Let $a = 0.5555e1$, $b = 0.4545e1$, c $= 0.4535e1$. Show that

$$a(b - c) \neq ab - ac.$$

ii. Further let $a = 0.5665e1$, $b = 0.5556e - 1$, $c = 0.5644e1$. Show that

$$(a + b) - c \neq (a - c) + b.$$

(7) Calculate the value of $x^2 + 2x - 2$ and $(2x - 2) + x^2$ where $x = 0.7320e0$, using normalized point arithmetic and proves that they are not the same. Compare with the value of $(x^2 - 2) + 2x$.
(8) Use four-digit rounding arithmetic and the formula to find the most accurate approximations to the roots of the following quadratic equations. Compute the absolute errors and relative errors.

$$\frac{1}{3}x^2 + \frac{123}{4}x - \frac{1}{6} = 0.$$

(9) Find the root of smallest magnitude of the equation $x^2 - 1000x + 25 = 0$ using quadratic formula. Work in floating-point arithmetic using a four-decimal place mantissa.
(10) Suppose two points (x_0, y_0) and (x_1, y_1) are on a straight line with $y_1 \neq y_0$. Two formulas are available to find the x-intercept of the line:

$$x = \frac{x_0 y_1 - x_1 y_0}{y_1 - y_0}, \text{ and } x = x_0 - \frac{(x_1 - x_0) y_0}{y_1 - y_0}.$$

Use the data $(x_0, y_0) = (1.31, 3.24)$ and $(x_1, y_1) = (1.93, 4.76)$ and three-digit rounding arithmetic to compute the x-intercept both ways. Which method is better and why?
(11) Consider the identity

$$\int_0^x \sin(xt)dt = \frac{1 - \cos(x^2)}{x}.$$

Explain the difficulty in using the right-hand fraction to evaluate this expression when x is close to zero. Give a way to avoid this problem and be as precise as possible.
(12) a. Consider the stability (by calculating the condition number) of $\sqrt{1 + x} - 1$ when x is near 0. Rewrite the expression to rid it of subtractive cancellation.
b. Rewrite $e^x - \cos x$ to be stable when x is near 0.

(13) Suppose that a function $f(x) = \ln(x+1) - \ln(x)$, is computed by the following algorithm for large values of x using six digit rounding arithmetic

$$\begin{aligned} x_0 : &= x = 12345 \\ x_1 : &= x_0 + 1 \\ x_2 : &= \ln x_1 \\ x_3 : &= \ln x_0 \\ f(x) := x_4 : &= x_2 - x_3. \end{aligned}$$

By considering the condition $\kappa(x_3)$ of the subproblem of evaluating the function, show that such a function evaluation is not stable. Also propose the modification of function evaluation so that algorithm will become stable.

(14) Assume 3-digit mantissa with rounding
a. Evaluate $y = x^3 - 3x^2 + 4x + 0.21 \quad$ for $x = 2.73$.
b. Evaluate $y = [(x-3)x + 4]x + 0.21 \quad$ for $x = 2.73$.
Compare and discuss the errors obtained in part (a) and (b).

(15) a. How many multiplications and additions are required to determine a sum of the form

$$\sum_{i=1}^{n} \sum_{j=1}^{i} a_i \, b_j \; ?$$

b. Modify the sum in part (a) to an equivalent form that reduces the number of computations.

(16) Let $P(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$ be a polynomial, and let x_0 be given. Construct an algorithm to evaluate $P(x_0)$ using nested multiplication.

(17) Construct an algorithm that has as input an integer $n \geq 1$, numbers $x_0, x_1, \cdots, x_n$, and a number x and that produces as output the product $(x - x_0)(x - x_1) \cdots (x_0 - x_n)$.

Bibliography

[Burden] Richard L. Burden, J. Douglas Faires and Annette Burden, "Numerical Analysis," Cengage Learning, 10th edition, 2015.

[Atkinson] K. Atkinson and W. Han, "Elementary Numerical Analysis," John Willey and Sons, 3rd edition, 2004.