

## Numerical Analysis MATH50003 (2024–25) Problem Sheet 3

In this problem sheet we explore floating point numbers and bounding errors in arithmetic with rounding. We begin with some simple examples of expressing real numbers in binary format and representing them as floating point numbers:

---

**Problem 1** What is  $\pi$  to 5 binary places? Hint: recall that  $\pi \approx 3.14$ .

**Problem 2** What are the single precision  $F_{32} = F_{127,8,23}$  floating point representations for the following:

$$2, \quad 31, \quad 32, \quad 23/4, \quad (23/4) \times 2^{100}$$

---

Floating point numbers cannot represent every real number. the next question explores the spacing between consecutive floating point numbers:

---

**Problem 3** Let  $m(y) = \min\{x \in F_{32} : x > y\}$  be the smallest single precision number greater than  $y$ . What is  $m(2) - 2$  and  $m(1024) - 1024$ ?

---

Not every calculation involving floating point numbers has errors: sometimes they are exact. The next questions explore cases where they are exact, and where we have to round the number to represent them as a floating point number:

---

**Problem 4** Suppose  $x = 1.25$  and consider 16-bit floating point arithmetic ( $F_{16}$ ). What is the error in approximating  $x$  by the nearest float point number  $\text{fl}(x)$ ? What is the error in approximating  $2x$ ,  $x/2$ ,  $x + 2$  and  $x - 2$  by  $2 \otimes x$ ,  $x \oslash 2$ ,  $x \oplus 2$  and  $x \ominus 2$ ?

**Problem 5** Show that  $1/5 = 2^{-3}(1.1001100110011\dots)_2$ . What are the exact bits for  $1 \oslash 5$ ,  $1 \oslash 5 \oplus 1$  computed using half-precision arithmetic ( $F_{16} := F_{15,5,10}$ ) (using default rounding)?

---

Arithmetic with floating point numbers is exact up to rounding and the *round bound* gives a way of precisely bounding the errors involved. To simplify the discussion we use *idealised floating point numbers*  $F_{\infty,S}$ , which avoids technicalities of subnormal numbers. We first see how error bounds can be deduced for two very simple expressions:

---

**Problem 6** Prove the following bounds on the *absolute error* of a floating point calculation in idealised floating-point arithmetic  $F_{\infty,S}$  (i.e., you may assume all operations involve normal floating point numbers):

$$\begin{aligned} (\text{fl}(1.1) \otimes \text{fl}(1.2)) \oplus \text{fl}(1.3) &= 2.62 + \varepsilon_1 \\ (\text{fl}(1.1) \ominus 1) \oslash \text{fl}(0.1) &= 1 + \varepsilon_2 \end{aligned}$$

such that  $|\varepsilon_1| \leq 11\epsilon_m$  and  $|\varepsilon_2| \leq 40\epsilon_m$ , where  $\epsilon_m$  is machine epsilon.

---

In the lectures/notes we saw how the effect of rounding errors in right-sided divided differences could be bounded. Here we extend this to central differences which yields a different heuristic for the optimal choice of  $h$ :

---

**Problem 7(a)** Assume that  $f^{\text{FP}} : F_{\infty,S} \rightarrow F_{\infty,S}$  satisfies  $f^{\text{FP}}(x) = f(x) + \delta_x$  where  $|\delta_x| \leq c\epsilon_m$

for all  $x \in F_{\infty, S}$ . Show that

$$\frac{f^{\text{FP}}(x+h) \ominus f^{\text{FP}}(x-h)}{2h} = f'(x) + \varepsilon$$

where the (absolute) error is bounded by

$$|\varepsilon| \leq \frac{|f'(x)|}{2} \epsilon_{\text{m}} + \frac{M}{3} h^2 + \frac{2c\epsilon_{\text{m}}}{h}.$$

**Problem 7(b)** Use the previous result to deduce, heuristically, an  $\alpha$  such that choosing  $h = C\epsilon_{\text{m}}^\alpha$  will result in, roughly, optimal accuracy.

We can also bound the errors in expressions involving many floating point numbers, a property that is necessary for bounding the error in more complicated algorithms. We first deduce a convenient function  $E_{n,\epsilon}$  which will be used in the resulting error bounds:

**Problem 8(a)** Suppose  $|\epsilon_k| \leq \epsilon$  and  $n\epsilon < 1$ . Show that  $\prod_{k=1}^n (1 + \epsilon_k) = 1 + \theta_n$  for some constant  $\theta_n$  satisfying

$$|\theta_n| \leq \underbrace{\frac{n\epsilon}{1 - n\epsilon}}_{E_{n,\epsilon}}.$$

We finally are in a place to bound products and additions of  $n$  floating point numbers. This lays the ground-work for understanding errors in simple algorithms such as in linear algebra.

**Problem 8(b)** Show if  $x_1, \dots, x_n \in F_{\infty, S}$  then

$$x_1 \otimes \dots \otimes x_n = x_1 \cdots x_n (1 + \theta_{n-1})$$

where  $|\theta_n| \leq E_{n, \epsilon_{\text{m}}/2}$ , assuming  $n\epsilon_{\text{m}} < 2$ .

**Problem 8(c)** Show if  $x_1, \dots, x_n \in F_{\infty, S}$  then

$$x_1 \oplus \dots \oplus x_n = x_1 + \dots + x_n + \sigma_n$$

where, for  $M = \sum_{k=1}^n |x_k|$ ,  $|\sigma_n| \leq ME_{n-1, \epsilon_{\text{m}}/2}$ , assuming  $n\epsilon_{\text{m}} < 2$ .