Numerical Analysis MATH50003 (2025–26) 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 π to 5 binary places? Hint: recall that $\pi \approx 3.14$.

SOLUTION We subtract off powers of two until we get 5 places. Eg we have

$$\pi = 3.14\ldots = 2+1.14\ldots = 2+1+0.14\ldots = 2+1+1/8+0.016\ldots = 2+1+1/8+1/64+0.000\ldots$$

Thus we have $\pi = (11.001001\ldots)_2$. The question is slightly ambiguous whether we want to round to 5 digits so either `11.00100` or `11.00101` would be acceptable. **END**

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

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

SOLUTION Recall that we have σ`,Q,S = 127,8,23`. Thus we write

$$2 = 2^{128-127} * (1.00000000000000000000000)_2$$

The exponent bits are those of

$$128 = 2^7 = (10000000)_2$$

Hence we get the bits

```
0 10000000 00000000000000000000000
```

We write

$$31 = (11111)_2 = 2^{131-127} * (1.1111)_2$$

And note that $131 = (10000011)_2$ Hence we have the bits

```
0 10000011 11110000000000000000000
```

On the other hand,

$$32 = (100000)_2 = 2^{132-127}$$

and $132 = (10000100)_2$ hence we have the bits

```
0 10000100 00000000000000000000000
```

Note that

$$23/4 = 2^{-2} * (10111)_2 = 2^{129-127} * (1.0111)_2$$

and $129 = (10000001)_2$ hence we get:

```
0 10000001 01110000000000000000000
```

Finally,

$$23/4 * 2^{100} = 2^{229-127} * (1.0111)_2$$

and $229 = (11100101)_2$ giving us:

```
0 11100101 01110000000000000000000
```

END

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 $m(1024) - 1024$?

SOLUTION The next float after 2 is $2 * (1 + 2^{-23})$ hence we get $m(2) - 2 = 2^{-22}$:

```
nextfloat(2f0) - 2, 2^(-22)
```

```
(2.3841858f-7, 2.384185791015625e-7)
```

similarly, for $1024 = 2^{10}$ we find that the difference $m(1024) - 1024$ is $2^{10-23} = 2^{-13}$:

```
nextfloat(1024f0) - 1024, 2^(-13)
```

```
(0.00012207031f0, 0.0001220703125)
```

END

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 $\mathrm{fl}(x)$? 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 $x \ominus 2$?

SOLUTION None of these computations have errors since they are all exactly representable as floating point numbers. **END**

Problem 5 Show that $1/5 = 2^{-3}(1.1001100110011\ldots)_2$. 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)?

SOLUTION

For the first part we use Geometric series:

$$\begin{aligned}
2^{-3}(1.1001100110{\color{magenta}011\ldots})_2 &= 2^{-3}\left(\sum_{k=0}^{\infty} \frac{1}{2^{4k}} + \frac{1}{2}\sum_{k=0}^{\infty} \frac{1}{2^{4k}}\right) \\
&= \frac{3}{2^4}\frac{1}{1 - 1/2^4} = \frac{3}{2^4 - 1} = \frac{1}{5}
\end{aligned}$$

Write $-3 = 12 - 15$ hence we have $q = 12 = (01100)_2$. Since $1/5$ is below the midpoint (the midpoint would have been the first magenta bit was 1 and all other bits are 0) we round down and hence have the bits:

$$0\ 01100\ 1001100110$$

Adding 1 we get:

$$1 + 2^{-3} * (1.1001100110)_2 = (1.00110011011)_2 \approx (1.0011001101)_2$$

Here we write the exponent as $0 = 15 - 15$ where $q = 15 = (01111)_2$. Thus we have the bits:

$$0\ 01111\ 0011001101$$

END

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):

$$(\mathrm{fl}(1.1) \otimes \mathrm{fl}(1.2)) \oplus \mathrm{fl}(1.3) = 2.62 + \varepsilon_1$$
$$(\mathrm{fl}(1.1) \ominus 1) \oslash \mathrm{fl}(0.1) = 1 + \varepsilon_2$$

such that $|\varepsilon_1| \leq 11\epsilon_{\mathrm{m}}$ and $|\varepsilon_2| \leq 40\epsilon_{\mathrm{m}}$, where ϵ_{m} is machine epsilon.

SOLUTION

The first problem is very similar to what we saw in lecture. Write

$$(\mathrm{fl}(1.1) \otimes \mathrm{fl}(1.2)) \oplus \mathrm{fl}(1.3) = (1.1(1+\delta_1)1.2(1+\delta_2)(1+\delta_3) + 1.3(1+\delta_4))(1+\delta_5)$$

where we have $|\delta_1|, \ldots, |\delta_5| \leq \epsilon_{\mathrm{m}}/2$. We first write

$$1.1(1+\delta_1)1.2(1+\delta_2)(1+\delta_3) = 1.32(1+\varepsilon_1)$$

where, using the bounds:

$$|\delta_1\delta_2|, |\delta_1\delta_3|, |\delta_2\delta_3| \leq \epsilon_{\mathrm{m}}/4, |\delta_1\delta_2\delta_3| \leq \epsilon_{\mathrm{m}}/8$$

we find that

$$|\varepsilon_1| \leq |\delta_1| + |\delta_2| + |\delta_3| + |\delta_1\delta_2| + |\delta_1\delta_3| + |\delta_2\delta_3| + |\delta_1\delta_2\delta_3| \leq (3/2 + 3/4 + 1/8) \leq 5/2\epsilon_{\mathrm{m}}$$

Then we have

$$1.32(1+\varepsilon_1) + 1.3(1+\delta_4) = 2.62 + \underbrace{1.32\varepsilon_1 + 1.3\delta_4}_{\varepsilon_2}$$

where

$$|\varepsilon_2| \leq (15/4 + 3/4)\epsilon_{\mathrm{m}} \leq 5\epsilon_{\mathrm{m}}.$$

Finally,

$$(2.62 + \varepsilon_2)(1 + \delta_5) = 2.62 + \underbrace{\varepsilon_2 + 2.62\delta_5 + \varepsilon_2\delta_5}_{\varepsilon_3}$$

where, using $|\varepsilon_2\delta_5| \leq 3\epsilon_{\rm m}$ we get,

$$|\varepsilon_3| \leq (5 + 3/2 + 3)\epsilon_{\rm m} \leq 10\epsilon_{\rm m}.$$

For the second part, we do:

$$(\mathrm{fl}(1.1) \ominus 1) \oslash \mathrm{fl}(0.1) = \frac{(1.1(1+\delta_1) - 1)(1+\delta_2)}{0.1(1+\delta_3)}(1+\delta_4)$$

where we have $|\delta_1|, \ldots, |\delta_4| \leq \epsilon_{\rm m}/2$. Write

$$\frac{1}{1+\delta_3} = 1 + \varepsilon_1$$

where, using that $|\delta_3| \leq \epsilon_{\rm m}/2 \leq 1/2$, we have

$$|\varepsilon_1| \leq \left|\frac{\delta_3}{1+\delta_3}\right| \leq \frac{\epsilon_{\rm m}}{2}\frac{1}{1-1/2} \leq \epsilon_{\rm m}.$$

Further write

$$(1+\varepsilon_1)(1+\delta_4) = 1 + \varepsilon_2$$

where

$$|\varepsilon_2| \leq |\varepsilon_1| + |\delta_4| + |\varepsilon_1||\delta_4| \leq (1 + 1/2 + 1/2)\epsilon_{\rm m} = 2\epsilon_{\rm m}.$$

We also write

$$\frac{(1.1(1+\delta_1) - 1)(1+\delta_2)}{0.1} = 1 + \underbrace{11\delta_1 + \delta_2 + 11\delta_1\delta_2}_{\varepsilon_3}$$

where

$$|\varepsilon_3| \leq (11/2 + 1/2 + 11/4) \leq 9\epsilon_{\rm m}$$

Then we get

$$(\mathrm{fl}(1.1) \ominus 1) \oslash \mathrm{fl}(0.1) = (1+\varepsilon_3)(1+\varepsilon_2) = 1 + \underbrace{\varepsilon_3 + \varepsilon_2 + \varepsilon_2\varepsilon_3}_{\varepsilon_4}$$

and the error is bounded by:

$$|\varepsilon_4| \leq (9 + 2 + 18)\epsilon_{\rm m} \leq 29\epsilon_{\rm m}.$$

END

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 $f^{\rm FP} : F_{\infty,S} \to F_{\infty,S}$ satisfies $f^{\rm FP}(x) = f(x) + \delta_x$ where $|\delta_x| \leq c\epsilon_{\rm m}$ for all $x \in F_{\infty,S}$. Show that

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

where the (absolute) error is bounded by

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

SOLUTION

In floating point we have

$$\begin{aligned}\frac{f^{\mathrm{FP}}(x+h) \ominus f^{\mathrm{FP}}(x-h)}{2h} &= \frac{f(x+h) + \delta_{x+h} - f(x-h) - \delta_{x-h}}{2h}(1+\delta_1) \\ &= \frac{f(x+h) - f(x-h)}{2h}(1+\delta_1) + \frac{\delta_{x+h} - \delta_{x-h}}{2h}(1+\delta_1)\end{aligned}$$

From PS1 Q4 we get the error term

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + \delta^{\mathrm{T}}$$

where

$$|\delta^{\mathrm{T}}| \leq Mh^2/6.$$

Thus

$$(f^{\mathrm{FP}}(x+h) \ominus f^{\mathrm{FP}}(x-h))/(2h) = f'(x) + \underbrace{f'(x)\delta_1 - \delta^{\mathrm{T}}(1+\delta_1) + \frac{\delta_{x+h} - \delta_{x-h}}{2h}(1+\delta_1)}_{\varepsilon}$$

where

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

END

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

SOLUTION

We want to balance the errors

$$\frac{M}{3}h^2 \approx \frac{2c\epsilon_{\mathrm{m}}}{h}$$

I.e. we want $h^3 = C\epsilon_{\mathrm{m}}$, that is, $\alpha = 1/3$.

END

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 θ_n satisfying

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

SOLUTION

$$\prod_{k=1}^{n+1}(1+\epsilon_k) = \prod_{k=1}^{n}(1+\epsilon_k)(1+\epsilon_{n+1}) = (1+\theta_n)(1+\epsilon_{n+1}) = 1 + \underbrace{\theta_n + \epsilon_{n+1} + \theta_n\epsilon_{n+1}}_{\theta_{n+1}}$$

where

$$
\begin{aligned}
|\theta_{n+1}| &\leq \frac{n\epsilon}{1-n\epsilon}(1+\epsilon)+\epsilon \\
&= \frac{n\epsilon+n\epsilon^2}{1-(n+1)\epsilon} \underbrace{\frac{1-(n+1)\epsilon}{1-n\epsilon}}_{\leq 1} + \frac{\epsilon-(n+1)\epsilon^2}{1-(n+1)\epsilon} \\
&\leq \frac{(n+1)-\epsilon}{1-(n+1)\epsilon}\epsilon \leq \frac{(n+1)\epsilon}{1-(n+1)\epsilon} = E_{n+1,\epsilon}.
\end{aligned}
$$

END

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 $x_1, \ldots, x_n \in F_{\infty,S}$ then

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

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

SOLUTION

We can expand out:

$$x_1 \otimes \cdots \otimes x_n = (\cdots((x_1x_2)(1+\delta_1)x_3(1+\delta_2)\cdots x_n(1+\delta_{n-1})) = x_1 \cdots x_n(1+\delta_1)\cdots(1+\delta_{n-1})$$

where $|\delta_k| \leq \epsilon_{\mathrm{m}}/2$. The result then follows from the previous result.

END

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

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

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

SOLUTION

Using Problem 8(a) we write:

$$
\begin{aligned}
(\cdots((x_1+x_2)(1+\delta_1)+x_3)(1+\delta_2)\cdots+x_n)(1+\delta_{n-1}) &= x_1\prod_{k=1}^{n-1}(1+\delta_k)+\sum_{j=2}^n x_j \prod_{k=j-1}^{n-1}(1+\delta_j) \\
&= x_1(1+\theta_{n-1})+\sum_{j=2}^n x_j(1+\theta_{n-j+1})
\end{aligned}
$$

where we have for $j = 2, \ldots, n$

$$|\theta_{n-j+1}| \leq E_{n-j+1,\epsilon_{\mathrm{m}}/2} \leq E_{n-1,\epsilon_{\mathrm{m}}/2}.$$

Thus we have

$$\sum_{j=1}^n x_j(1+\theta_{n-j+1}) = \sum_{j=1}^n x_j + \underbrace{\sum_{j=1}^n x_j\theta_{n-j+1}}_{\sigma_n}$$

where

$$|\sigma_n| \le \sum_{j=1}^{n} |x_j \theta_{n-j+1}| \le \sup_j |\theta_{n-j+1}| \sum_{j=1}^{n} |x_j| \le \|\boldsymbol{x}\|_1 E_{n-1,\epsilon_{\mathrm{m}}/2}.$$

END
