

MATH50003 Numerical Analysis (2022–23)

Problem Sheet 2

This problem sheet explores the bounding of floating point arithmetic errors, and shows how these can be used to bound errors in algorithms.

Please complete the problems using pen-and-paper, though some can be verified using Julia.

Problem 1 Suppose $0 \leq x < \min F_{\sigma, Q, S}^{\text{normal}}$ (the *sub-normal range*). Show that rounding has guaranteed *absolute error*:

$$\begin{aligned}\text{fl}^{\text{up}}(x) &= x + \delta_x^{\text{up}} \\ \text{fl}^{\text{down}}(x) &= x + \delta_x^{\text{down}} \\ \text{fl}^{\text{near}}(x) &= x + \delta_x^{\text{near}}\end{aligned}$$

where

$$\begin{aligned}|\delta_x^{\text{up/down}}| &\leq 2^{1-\sigma-S} \\ |\delta_x^{\text{near}}| &\leq 2^{-\sigma-S}\end{aligned}$$

Problem 2.1 Suppose $|\epsilon_i| \leq \epsilon$ and $n\epsilon < 1$. Show that

$$\prod_{k=1}^n (1 + \epsilon_k) = 1 + \theta_n$$

for some constant θ_n satisfying

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

Hint: use induction.

Problem 2.2 Show if $x_1, \dots, x_n \in F$ then

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

where $|\theta_n| \leq E_{n, \epsilon_m/2}$, assuming $n\epsilon_m < 2$. You may assume all operations are within the normalised range.

Problem 2.3 Show if $x_1, \dots, x_n \in F$ then

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

where, for $M = \sum_{k=1}^n |x_k|$, $|\sigma_n| \leq M E_{n-1, \epsilon_m/2}$, assuming $n\epsilon_m < 2$. You may assume all operations are within the normalised range. Hint: use Problem 2.1 to first write

$$x_1 \oplus \cdots \oplus x_n = x_1(1 + \theta_{n-1}) + \sum_{j=2}^n x_j(1 + \theta_{n-j+1}).$$

Problem 3.1 Consider the algorithm `exp_taylor_fast` from lectures:

```
In [ ]: function exp_taylor_fast(x, n)
    ret = zero(x) # 0 of same type as x
    summand = one(x)
    for k = 0:n
        ret += summand
        summand *= x/(k+1)
    end
    ret
end
```

```
Out[ ]: exp_taylor_fast (generic function with 1 method)
```

Write this algorithm as a one-line mathematical function $\exp_n^t(x)$ involving \oplus , \otimes , and \otimes . You may find it convenient to use the notation:

$$\bigoplus_{k=1}^n x_k := x_1 \oplus \cdots \oplus x_n = (\cdots ((x_1 \oplus x_2) \oplus x_3) \cdots \oplus x_{n-1}) \oplus x_n$$

$$\bigotimes_{k=1}^n x_k := x_1 \otimes \cdots \otimes x_n = (\cdots ((x_1 \otimes x_2) \otimes x_3) \cdots \otimes x_{n-1}) \otimes x_n$$

Problem 3.2 Show that

$$\exp_n^t(x) = \sum_{k=0}^{n-1} \frac{x^k}{k!} + \varepsilon_n$$

where

$$|\varepsilon_n| \leq \exp(|x|)(2E_{n,\epsilon_m/2} + E_{n,\epsilon_m/2}^2),$$

assuming $n\epsilon_m < 1$. You may assume all operations are within the normalised range. Hint: combine Problem 2.2 and 2.3 and note that $E_{k,\epsilon_m/2} \leq E_{j,\epsilon_m/2}$ when $k \leq j$.

Problem 3.3 For $x > 0$, find a bound on the relative error $|\rho_n|$ where

$$\exp_n^t(x) = (1 + \rho_n) \exp x.$$

Why does the bound break down when $x < 0$?

Problem 3.4 Give two reasons why the above error bound is not valid as $n \rightarrow \infty$ if $F_{\sigma,Q,S}$ is fixed. If S and Q are allowed to depend on n can we guarantee convergence to $\exp x$?