APPM 4650 — Floating point arithmetic

Algorithms vs Pseudocode

Definition 0.1 *An* algorithm *is a procedure that describes a finite number of steps to be performed in a specific order. NOTE: There is no mention of problem statement here.*

Definition 0.2 *A* pseudocode *is a way of describing an algorithm in a clear consise form. It includes : Input, output and distinct clear steps.*

Example: Write a pseudocode for the algorithm of summing N values $\{x_i\}_{i=1}^N$. The algorithm is $S = \sum_{i=1}^N x_i$.
Solution:
Input: N, $\{x_i\}_{i=1}^N$
Output: S
Step 1: Set $S = 0$. (initialize)
Step 2: for $i = 1, \ldots, N$ do
$S = S + x_i$ (add terms)
Step 3: Output S.
Our job in this class is to design or choose an algorithm to approximate the solution of a problem. We want this technique to be reliable. What does that mean mathematically?
We would like that if there is a small change in the input data, that there is just a small change in the output data. Such an algorithm is called *stable*. Otherwise the algorithm is called *unstable*.
There are some algoirthms that are stable for a class of input data but not all input data. These are called *conditionally stable*.
Example: Consider the following two algorithms:
Algorithm 1: $p_n = c_1(1/3)^n + c_2 3^n$
Algorithm 2: $p_n = \frac{10}{3}p_{n-1} - p_{n-2}$
These algorithms are equivalent (in exact arithmetic) when $p_0 = 1$ and $p_1 = 1/3$. This corresponds to $c_1 = 1$ and $c_2 = 0$. Thus the first algorithm can be simplified to $p_n = (1/3)^n$ for all n.
Now supposed we used 5 digit arithmetic and errors were made in solving for c_1 and c_2. Let $\hat{c}_1 = 1.0000$ and $\hat{c}_2 = -0.12500 \times 10^{-5}$ denote the approximate coefficients. Then algorithm 1 becomes

$$\hat{p}_n = 1.0000(1/3)^n - 0.12500 \times 10^{-5}(3)^n$$

as $n \to \infty$, $\hat{p}_n \to -\infty$ instead of 0.
The code stability_example illustrates how quickly this error grows.
There is a way to determine if a problem is stable. It is called the condition number. See Section 2.4 of Dahlquist and Björk for more information.

Convergence Rates

Big O vs little o

Definition 0.3 *Let $f(x)$ and $g(x)$ be defined in some neighborhood of $x = 0$.*

- *We say $f(x) = o(g(x))$ as $x \to 0$ (little o) if*

$$\lim_{x\to o}\left|\frac{f(x)}{g(x)}\right| = 0$$

In other words, $f(x) \to 0$ faster than $g(x)$.

- *We say $f(x) = O(g(x))$ as $x \to 0$ (big O) if there exist a positive constant M such that*

$$\left|\frac{f(x)}{g(x)}\right| \leq M$$

for all x in some neighborhood of 0. In other words, $f(x)$ and $g(x)$ have the same behavior up to a constant around 0.

Example: Verify that $\epsilon^2 \log(\epsilon) = o(\epsilon)$ as $\epsilon \to 0^+$.
Solution:

$$\begin{aligned} \lim_{\epsilon \to 0^+} \frac{\epsilon^2 \log(\epsilon)}{\epsilon} &= \lim_{\epsilon \to 0^+} \epsilon \log(\epsilon) \\ & \lim_{\epsilon \to 0^+} \frac{\log(\epsilon)}{1/\epsilon} \\ & \lim_{\epsilon \to 0^+} \frac{1/\epsilon}{-1/\epsilon^2} \text{ by L'Hopital's rule} \\ & \lim_{\epsilon \to 0^+} -\epsilon = 0 \end{aligned}$$

Example: Verify $\sin(\epsilon) = O(\epsilon)$ as $\epsilon \to 0^+$.
Solution: We need to look at $\frac{\sin(\epsilon)}{\epsilon}$. We can add 0 to the top and bottom to get

$$\frac{\sin(\epsilon) - \sin(0)}{\epsilon - 0}$$

Now we can use the mean value theorem.
By the mean value theorem, there exist a number $c \in [0, \epsilon]$ such that

$$\frac{\sin(\epsilon) - \sin(0)}{\epsilon - 0} = \cos(c).$$

This means that

$$|\sin(\epsilon)| \leq |\epsilon \cos(c)| \leq |\epsilon|$$

since $|\cos(c)| \leq 1$.
Therefore $\sin(\epsilon) = O(\epsilon)$.

In this class we will talk about the convergence rates of different algorithms. These are all in terms of big O.

Definition 0.4 *Suppose $\{\beta_n\}_{n=1}^{\infty}$ is a sequence known to convergence to 0 and $\{\alpha_n\}_{n=1}^{\infty}$ converges to α. If there exist and constant K such that*

$$|\alpha_n - \alpha| \leq K|\beta_n|$$

for large n then we say that $\{\alpha_n\}_{n=1}^{\infty}$ converges to α at a rate *$O(\beta_n)$. This can be written as $\alpha_n = \alpha + O(\beta_n)$.*

Example: Find the rate of convergence of $\sin(1/n)$ as $n \to \infty$.
Solution: We know that close to 0, $\sin(x)$ and x behave roughly the same. Thus we choose to compare $\sin(1/n)$ with $1/n$.
Note $\sin(1/n) \leq 1/n$ for large n is a re-statement of above. Thus $\sin(1/n) = O(1/n)$.

Definition 0.5 *Suppose that* $\lim_{h\to 0} G(h) = 0$ *and* $\lim_{h\to 0} F(h) = L$*. If there exists a positive constant* K *such that*

$$|F(h) - L| \leq K|G(h)|$$

for h *sufficiently small, then we write* $F(h) = L + O(G(h))$*.*
We will often use $G(h) = h^p$ *where* $p > 0$*.*

Example: Find the rate of convergence of $\lim_{h\to 0} \frac{\sin(h)}{h} = 1$.
Solution: For this problem we will use the McLaurin expansion of $\sin(h)$.
Recall $\sin(h) = h - \frac{h^3}{3!} + \frac{h^5}{5!} + \cdots$.
Thus

$$\begin{aligned}
\left|\frac{\sin(h)}{h} - 1\right| &= \left|\frac{h - \frac{h^3}{3!} + \frac{h^5}{5!} + \cdots}{h} - 1\right| \\
&= \left|1 - \frac{h^2}{3!} + \frac{h^4}{5!} + \cdots - 1\right| \\
&= O(h^2)
\end{aligned}$$