

**CSE/Math 455**  
**Lecture #5**

**Announcement**  
**No office**  
**hours today.**

**Floating Point Arithmetic.**

Today I'll discuss some important issues about simple computations.

**Example.**  $f(x) = \tan x - \sin x$ ,  $x$  near zero  $x = 10^{-10}$

$$\gg tx = \tan(x)$$

$$\gg sx = \sin(x)$$

$$\gg fx = tx - sx$$

$fx = 0$       Loss of precision as we get closer to 0.

Now we can do this one of two ways

**Use Trigonometric Identities.**

$$\begin{aligned} f(x) &= \tan x - \sin x = \left( \frac{\sin x}{\cos x} - \sin x \right) \\ &= \sin x \left( \frac{1}{\cos x} - 1 \right) = \sin x (\sec x - 1) \\ f(x) &= \sin x \left( \sqrt{1 + \tan^2 x} - 1 \right) \\ &= \sin x \left( \sqrt{1 + \tan^2 x} - 1 \right) \frac{\sqrt{1 + \tan^2 x} + 1}{\sqrt{1 + \tan^2 x} + 1} \\ &= \frac{\sin x \tan^2 x}{\sqrt{1 + \tan^2 x} + 1} = \frac{\sin x \tan^2 x}{\sec x + 1} \end{aligned}$$

$f(x) 5 \times 10^{-31}$

**Or use Taylor Series** – Only need a few terms

$$\tan x = x + \frac{x^3}{3} + \frac{2x^5}{15} + O(x^7)$$

$$\sin x = x - \frac{x^3}{6} + \frac{x^5}{120} + O(x^7)$$

$$f(x) = \tan x - \sin x = \frac{x^3}{2} + \frac{7x^5}{120} + O(x^7)$$

This is often much easier.

Inf, - Inf, NaN special numbers

Operations that generate NaN (not a number)

$$\text{Inf} + (-\text{Inf})$$

$$0 * \text{Inf}$$

$$\text{Inf}/\text{Inf}$$

$$0/0$$

$$\sqrt{-3}$$

$$1/0 = \text{Inf}$$

$$-1/0 = -\text{Inf}$$

$$1/\text{Inf} = 0$$

$$1/(-\text{Inf}) = -0$$

IEEE arithmetic has two zeroes, +0 and -0.

Closed System

{floating point numbers, Inf, -Inf, NaN}

→ { floating point numbers, Inf, -Inf, NaN}

operations

Taking Some Advantage

$$f(x) = \sec x - \tan x \quad x \text{ near } \pi/2$$

$$f\left(\frac{\pi}{2}\right) = \infty - \infty = \text{NaN}$$

MATLAB is smart enough not to give you Inf here.

$$\begin{aligned} f(x) &= (\sec x - \tan x) \frac{(\sec x + \tan x)}{\sec x + \tan x} \\ &= \frac{\sec^2 x - \tan^2 x}{\sec x + \tan x} = \frac{1}{\sec x + \tan x} = \frac{1}{\infty} = 0 \end{aligned}$$

It's rare for floating point software to do this, but it can.

Now we consider a different type of computation

Linear Systems of Equations

$$\frac{\epsilon_m}{10} x_1 + x_2 = 1$$

Good approximate answer is  $x_1 = x_2 = 1$ .

$$\begin{aligned} & \left( \begin{array}{cc|c} \epsilon_m/10 & 1 & 1 \\ 1 & 1 & 2 \end{array} \right) \quad \epsilon_m \text{ machine unit} \\ \rightarrow & \left( \begin{array}{cc|c} \epsilon_m/10 & 1 & 1 \\ 0 & 1 - 10/\epsilon_m & 2 - 10/\epsilon_m \end{array} \right) \quad \epsilon_m \underline{\text{round}} \\ & \approx \left( \begin{array}{cc|c} \epsilon_m/10 & 1 & 1 \\ 0 & -10/\epsilon_m & -10/\epsilon_m \end{array} \right) \end{aligned}$$

Back solve  $x_1 = 0, x_2 = 1$

Again there is a “fix”. Partial Pivoting —

put largest uneliminated entry in the column in pivot or diagonal position

$$\begin{aligned} & \left( \begin{array}{cc|c} 1 & 1 & 2 \\ \epsilon_m/10 & 1 & 1 \end{array} \right) \\ \rightarrow & \left( \begin{array}{cc|c} 1 & 1 & 2 \\ 0 & \epsilon_m/10 & 1 - 2 \cdot \epsilon_m/10 \end{array} \right) \quad \text{round} \\ & \approx \left( \begin{array}{cc|c} 1 & 1 & 2 \\ 0 & 1 & 1 \end{array} \right) \quad \text{Yield } \underline{x_1 = x_2 = 1} \end{aligned}$$

Sometimes, changing the algorithm does no good at all.

$$(1 + 2\epsilon_m)x_1 + (1 + w\epsilon_m)x_2 = 2$$

$$(1 + \epsilon_m)x_1 + x_2 = 2$$

Use augmented matrix approach

$$\left( \begin{array}{cc|c} 1 + 2\epsilon_m & 1 + 2\epsilon_m & 2 \\ 1 + \epsilon_m & 1 & 2 \end{array} \right)$$

$$\alpha = \frac{1 + \epsilon_m}{1 + 2\epsilon_m} \times 1 - \epsilon_m$$

To machine precision, I get

$$\left( \begin{array}{cc|c} 1 + 2\epsilon_m & 1 + 2\epsilon_m & 2 \\ 0 & -\epsilon_m & 2\epsilon_m \end{array} \right)$$

$$\underline{x_1 = 4, \quad x_2 = -2} \quad x = \begin{pmatrix} -2 \\ 4 \end{pmatrix}$$

Correct to  
machine  
precision

Small change in right hand side

$$\left( \begin{array}{cc|c} 1 + 2\epsilon_m & 1 + 2\epsilon_m & 2 + 4\epsilon_m \\ 1 + \epsilon_m & 1 & 2 + \epsilon_m \end{array} \right)$$

Correct answer is  $x_1 = x_2 = 1$

With rounding, I get,  $x_1 = 0, x_2 = 2$  Why?

Every trick I know except increasing the precision, yields similar wrong answers.

**Reason.**

Coefficient matrix

$$A = \begin{pmatrix} 1 + 2\epsilon_m & 1 + 2\epsilon_m \\ 1 + \epsilon_m & 1 \end{pmatrix}$$

is very close to singular.  $A$  is called ill-conditioned. Thus any algorithm could have difficulty solving a linear system with coefficient matrix  $A$ .

More on this next time.