CS371 Introduction to Computational Mathematics

Austin Xia

May 21, 2020

Contents

List of Figures

List of Tables

1 Course Information

1.1 Contact

Instructor: Eugene Zima
Email: ezima@uwaterloo.ca

1.2 Grade

4 assignments 32%
6 on-line quizzes 18%
Midterm 20%
Final 30%

2 Floating point

How are numbers stored on a computer?
How does that affect Numerical algorithum and solution
goal of computational mathematics is defined as

- Finding and developing algorithms that solve mathematical problems computaionally (with a computer)
- Desired properties of our algorithum:
  - Accuray: result is numerically close to the actually solution
  - Efficiency: quickly solve the problem with resonable resources
  - Robustness: algorithum works well for a variety of inputs

2.1 Source of Error

- Error in input
  - Measurement Error
  - Rounding Error(due to finite digit of computer)
- Error as a result of calculation, approximation and algorithum
  - Truncation error: Talor series
  - Rounding error in elementary steps of algorithum

2.2 Types of error

- Absolute Error $= |x - \hat{x}$
- Relative Error $= \frac{|x-\hat{x}|}{|x|}$

2.3 Catastrophic Cancellation

cancellation of significant digits makes unknown round-off digits rel-evant to the final result becasue we don't know the lost digit

2.4 Floating point Representation

Allow the decimal point to "float"

Definition 2.4.1
A Floating point number system is defined by three componentsL

- base: the base of number system, b_f
- the mantissa: contains the normalized value of the number, m_f
- exponent: which defines the offset from normalization, e_f

$$F[b = b_f, m = m_f, e_f] = \pm m * b^e$$

Definition 2.4.2 (relative error in converting real number x to a floating number fl(x))

$$\delta_x = \frac{x - fl(x)}{x}$$

we want to find the upper bound of δ

Definition 2.4.3 (machine epsilon)
ϵ_{mach} is the smallest number $\epsilon > 0$ such that $fl(1 + \epsilon) > 1$

Lemma 2.4.1

- $\epsilon_{mach} = b^{1-m}$ if chopping is used
- $\epsilon_{mach} = \frac{1}{2}b^{1-m}$ if rounding is used

Theorem 2.4.2
For any floating point system F, under chopping

$$|\delta_x| = |\frac{x - fl(x)}{x}| \leq \epsilon_{mach}$$

2.5 Floating point operation

$$a \oplus b = fl(fl(a) + fl(b)) = (a(1 + n_1) + b(1 + n_2))(1 + n)$$

$|n| < \epsilon_{mach}$

Definition 2.5.1
A problem is well conditioned with respect to the absolute error if small changes in input reulst in small changes in output
A problem is ill-conditioned if small change in input reulst in large change in output

Definition 2.5.2
Condition number with respect to absolute error is: $\kappa_A = \frac{|\delta z|}{|\delta x|}$ and it is called absolute condition number

Definition 2.5.3
Condition number with respect to relative error is:

$$\kappa_R = \frac{\frac{|\delta_z|}{|z|}}{\frac{|\delta_x|}{|x|}}$$

For $0.1 < K_A, K_R < 10$, a problem is well-conditioned and for $K_A, K_R \to \infty$ problem is ill-formed

2.6 vector norms

There are three standard vector norms 2-norm, 1-norm and ∞ norm

Definition 2.6.1

$$||\vec{x}||_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$$

$$||\vec{x}||_\infty = max(x_i)$$

$$||\vec{x}||_1 = \sum_{i=1}^{n} |X_i|$$

Theorem 2.6.1 (Cauchy-Schwartz Inequality)
Let —— . —— be a vector over a vector space V induced by inner product. Then

$$|\vec{x} * \vec{y}| \leq ||\vec{x}|| * ||\vec{y}||$$

2.7 Stability of a numerical algorithum

If an algorithm propagates error and produce larger error, it's unstable algorithum
small change in initial value produce small change in final result, then algorithum is stable
An algorithm that satisfy this property is called Stable:

- If $E_n \approx C * n * E_0$ then growth of error is linear.
- If $E_n \approx C^n * E_0$ then growth of error is exponential

2.8 Big Oh

Definition 2.8.1
$f(x) = O(x^n) as x \to 0$ is equivelant to: f(x) is bounded from above by $|x|^n$, up to a constant c

Example: $g(x) = 3x^2 + 7x^3$. we say $g(x) = O(X^2)$ as $x \to 0$, $g(x) = O(X^3)$ as $x \to \infty$

3 Root Finding

For a fucntion f, find x* satisfying $f(x*) = 0$
Lots of applications in physics, chem, etc.

Definition 3.0.1 (root finding)
Given f(x) and some error tolerence $\epsilon > 0$, find x^* such that $|f(x^*)| < \epsilon$

3.1 Four algorithms for root finding

3.1.1 Bisection Method

- A simple method to find root
- Convergence is guaranteed
- f(x) should be continuous on [a, b] and $f(a) * f(b) \leq 0$

Theorem 3.1.1
If f(x) is continous on the interval $[a_0, b_0]$ such that $f(a_0)f(b_0) \leq 0$ Then the interval $[a_k, b_k]$ is defined by

$$a_k = \begin{cases} a_{k-1} \;\; if \;\; f(a_{k-1})f((a_{k-1}+b_{k-1})/2) \leq 0 \\ (a_{k-1}+b_{k-1})/2 \;\; otherwise \end{cases}$$

$$b_k = \begin{cases} b_{k-1} \;\; if \;\; f(a_{k-1})f((a_{k-1}+b_{k-1})/2) > 0 \\ (a_{k-1}+b_{k-1})/2 \;\; otherwise \end{cases}$$

How many steps does this algorithum require?

$$2^{-n}|b-a| \leq t \to n \geq \frac{1}{log2} log(\frac{|b-a|}{t})$$

3.1.2 fixed point method

Definition 3.1.1
x^* is a fixed point of $g(x)$ if $g(x^*) = x^*$

4 Numerical Linear Algebra

For matrix A, vector c, find x* satisfying $Ax^* = c$.
Applications in google page rank

5 Polynomial Interpolation

For a set of points, find a polynomial that fits the points

6 Numerical Integration

Find a numerical approximation to Integration

7 Discrete Fourier Method

Fourier transforms time (or space) to/from frequency. One of the most important numerical algorithum