Numerical Analysis

Lecture1: Introduction

Instructor: Prof. Guanding Yu
Zhejiang University

Outline

Motivation

- Q1: Find x?

$$\mathbf{Ax} = \mathbf{b}$$

Motivation

- Q1: Find x?

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

- Q2: Find x?

$$x = (x+1)^{\frac{1}{2}} + x^{\frac{1}{3}}$$

Motivation

- Q1: Find x?

$$\mathbf{Ax} = \mathbf{b}$$

- Q2: Find x?

$$x = (x+1)^{\frac{1}{2}} + x^{\frac{1}{3}}$$

- Q3: Wireless Channel Capacity

$$C = \int_0^\infty \log(1+x)\exp(-x)dx$$

Motivation

- Q1: Find x?

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

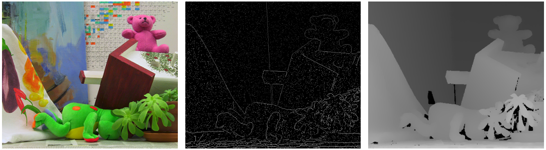
- Q2: Find x?

$$x = (x+1)^{\frac{1}{2}} + x^{\frac{1}{3}}$$

- Q3: Wireless Channel Capacity

$$C = \int_0^{\infty} \log(1+x)\exp(-x)dx$$

- Q4:Guided depth upsampling



$$\frac{\partial D(\mathbf{p};t)}{\partial t} = \mathrm{div}\left(\mathbf{W}(\mathbf{p};t)\nabla D(\mathbf{p};t)\right) \quad s.t.\ \ D(\mathbf{p};0) = D_0(\mathbf{p})$$

Objective

Introduce the applied numerical methods, including:

- Numerical approaches for
  - linear equations
  - nonlinear equations
  - differentiation and integration
  - partial differential equations

Objective

Introduce the applied numerical methods, including:

- Numerical approaches for
  - linear equations
  - nonlinear equations
  - differentiation and integration
  - partial differential equations
- Error analysis

Outline

Binary floating point numbers

- Long real format
  - Default data type in MATLAB, 'double' in C
  - Base: 2

1 bit	11 bits	52 bits
s	c	f

$$(-1)^s 2^{c-1023}(1+f)$$

  - Ex: 0 10000000011 101110 . . . 0

Binary floating point numbers

- Long real format
  - Default data type in MATLAB, 'double' in C
  - Base: 2

1 bit	11 bits	52 bits
s	c	f

$$(-1)^s 2^{c-1023}(1+f)$$

  - Ex: 0 10000000011 101110 . . . 0
  - Maximum / $2^{1023} \cdot (2-2^{-52}) \approx 0.17977 \times 10^{309}$

Binary floating point numbers

- Long real format
  - Default data type in MATLAB, 'double' in C
  - Base: 2

1 bit	11 bits	52 bits
s	c	f

$$(-1)^s 2^{c-1023}(1+f)$$

  - Ex: 0 10000000011 101110 . . . 0
  - Maximum / $2^{1023} \cdot (2-2^{-52}) \approx 0.17977 \times 10^{309}$
  - Minimum / $2^{-1022} \cdot (1+0) \approx 0.22251 \times 10^{-307}$

Binary floating point numbers

- Long real format
  - Default data type in MATLAB, 'double' in C
  - Base: 2

1 bit	11 bits	52 bits
s	c	f

$$(-1)^s 2^{c-1023}(1+f)$$

  - Ex: 0 10000000011 101110 . . . 0
  - Maximum / $2^{1023} \cdot (2 - 2^{-52}) \approx 0.17977 \times 10^{309}$
  - Minimum / $2^{-1022} \cdot (1+0) \approx 0.22251 \times 10^{-307}$
  - Overflow / underflow

Binary floating point numbers

- Long real format
  - Default data type in MATLAB, 'double' in C
  - Base: 2

1 bit	11 bits	52 bits
s	c	f

$$(-1)^s 2^{c-1023}(1+f)$$

  - Ex: 0 10000000011 101110 . . . 0
  - Maximum / $2^{1023} \cdot (2-2^{-52}) \approx 0.17977 \times 10^{309}$
  - Minimum / $2^{-1022} \cdot (1+0) \approx 0.22251 \times 10^{-307}$
  - Overflow / underflow
  - http://en.wikipedia.org/wiki/IEEE_floating_point

Decimal Floating Point Numbers

- Base: 10
- k-digit decimal machine number:

$$\pm 0.d_1 d_2 \dots d_k \times 10^n, \quad 1 \leq d_1 \leq 9, \quad 0 \leq d_i \leq 9$$

- Any positive number within the numerical range can be written:

$$y = 0.d_1 d_2 \dots d_k d_{k+1} d_{k+2} \dots \times 10^n$$

- Two ways to represent y with k digits:
  1. Chopping:
  $$fl(y) = 0.d_1 d_2 \dots d_k \times 10^n$$
  2. Roundoff: Add $5 \times 10^{n-(k+1)}$ and chop:
  $$fl(y) = 0.\delta_1 \delta_2 \dots \delta_k \times 10^n$$

- Roundoff error

Errors and Significant Digits

Errors:

If p^* is an approximation to p, the absolute error is $|p - p^*|$, and the relative error is $|p - p^*|/|p|$, provided that $p \neq 0$.

Ex 2 Determine the absolute and relative errors when approximating p by p^* when

(a) $p = 0.3000 \times 10^1$ and $p^* = 0.3100 \times 10^1$;

(b) $p = 0.3000 \times 10^{-3}$ and $p^* = 0.3100 \times 10^{-3}$;

(c) $p = 0.3000 \times 10^4$ and $p^* = 0.3100 \times 10^4$.

Solution

(a) For $p = 0.3000 \times 10^1$ and $p^* = 0.3100 \times 10^1$ the absolute error is 0.1, and the relative error is $0.333\overline{3} \times 10^{-1}$.

(b) For $p = 0.3000 \times 10^{-3}$ and $p^* = 0.3100 \times 10^{-3}$ the absolute error is 0.1×10^{-4}, and the relative error is $0.333\overline{3} \times 10^{-1}$.

(c) For $p = 0.3000 \times 10^4$ and $p^* = 0.3100 \times 10^4$, the absolute error is 0.1×10^3, and the relative error is again $0.333\overline{3} \times 10^{-1}$.

Errors and Significant Digits

Significant digits:

The number p^* is said to approximate p to t significant digits if t is the largest nonnegative integer for which

$$\frac{|p - p^*|}{|p|} \leq 5 \times 10^{-t}$$

$$\begin{aligned}
\left|\frac{y - fl(y)}{y}\right| &= \left|\frac{0.d_1 d_2 \ldots d_k d_{k+1} \ldots \times 10^n - 0.d_1 d_2 \ldots d_k \times 10^n}{0.d_1 d_2 \ldots \times 10^n}\right| \\
&= \left|\frac{0.d_{k+1} d_{k+2} \ldots \times 10^{n-k}}{0.d_1 d_2 \ldots \times 10^n}\right| = \left|\frac{0.d_{k+1} d_{k+2} \ldots}{0.d_1 d_2 \ldots}\right| \times 10^{-k} \\
&\leq \frac{1}{0.1} \times 10^{-k} = 10^{-k+1}
\end{aligned}$$

Floating Point Operations

- Floating Point Operations
  - Machine addition: $x \oplus y = fl(fl(x) + fl(y))$
  - Subtraction: $x \ominus y = fl(fl(x) - fl(y))$
  - Multiplication: $x \otimes y = fl(fl(x) \times fl(y))$
  - division: $x \oslash y = fl(fl(x) \div fl(y))$
- Cancelation
  - Common problem: Subtraction of nearly equal numbers:

$$fl(x) = 0.d_1 d_2 \ldots d_p \alpha_{p+1} \alpha_{p+2} \ldots \alpha_k \times 10^n$$

$$fl(y) = 0.d_1 d_2 \ldots d_p \beta_{p+1} \beta_{p+2} \ldots \beta_k \times 10^n$$

gives fewer digits for significance:

$$fl(fl(x) - fl(y)) = 0.\sigma_{p+1} \sigma_{p+2} \ldots \sigma_k \times 10^{n-p}$$

Floating Point Operations - Errors

Given $x = 5/7$, $u = 0.714251$, $v = 98765.9$, and $w = 0.111111 \times 10^{-4}$. Determine the five-digit chopping values of

Operation	Result	Actual value	Absolute error	Relative error
$x \ominus u$	0.30000×10^{-4}	0.34714×10^{-4}	0.471×10^{-5}	0.136
$(x \ominus u) \oplus w$	0.27000×10^{1}	0.31242×10^{1}	0.424	0.136
$(x \ominus u) \otimes v$	0.29629×10^{1}	0.34285×10^{1}	0.465	0.136
$u \oplus v$	0.98765×10^{5}	0.98766×10^{5}	0.161×10^{1}	0.163×10^{-4}

Floating Point Operations - Errors

Consider a quadratic equation $x^2 + 62.10x + 1 = 0$, whose roots are approximately

$$x_1 = -0.01610723 \qquad x_2 = -62.08390.$$

We use four digital rounding arithmetic to find the root.

$$\sqrt{b^2 - 4ac} = 62.06.$$

$$fl(x_1) = \frac{-62.10 + 62.06}{2.000} = -0.02000$$

with the large relative error

$$\frac{|-0.01611 + 0.02000|}{|-0.01611|} \approx 2.4 \times 10^{-1}$$

Floating Point Operations - Errors

To obtain a more accurate four-digit rounding approximation for x_1, we change the quadratic formula into

$$x_1 = \frac{-2c}{b + \sqrt{b^2 - 4ac}}$$

Then

$$fl(x_1) = \frac{-2.000}{62.10 + 62.06} = -0.01610$$

which has the small relative error 6.2×10^{-4}.

The lesson: Think before you compute!

Outline

Algorithms

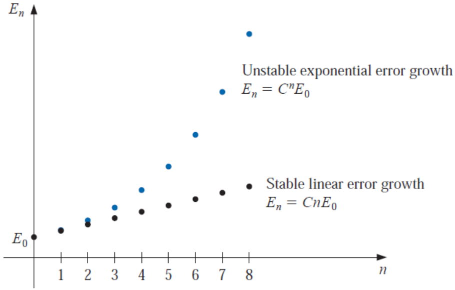
Definition

An algorithm is a set of operations to sovle a problem or approximate a solution to the problem.

- Growth of error
  Suppose $E_0 > 0$ is an initial error, and E_n is the error after n operations.
  - $E_n \approx CnE_0$: linear growth of error
  - $E_n \approx C^n E_0$: exponential growth of error

Stability

- Stability properties of algorithms
  - Stable: small changes in the initial data produce small changes in the final result
  - Unstable or conditionally stable: large errors in final results for all or some initial data with small errors



Rate of convergence (sequences)

Definition

Suppose $\{\beta_n\}_{n=1}^{\infty}$ is a sequence converging to zero, and $\{\alpha_n\}_{n=1}^{\infty}$ converges to a number α. If a positive constant K exists with

$$|\alpha_n - \alpha| \le K|\beta_n|, \quad \text{for large} \;\; n,$$

then we say that $\{\alpha_n\}_{n=1}^{\infty}$ converges to α with rate of convergence $O(\beta_n)$, indicated by $\alpha_n = \alpha + O(\beta_n)$.

Polynomial rate of convergence

Normally we will use

$$|\beta_n| = \frac{1}{n^p}$$

and look for the largest value $p > 0$ such that $\alpha_n = \alpha + O(\frac{1}{n^p})$

Rate of convergence (functions)

Definition

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

$$|F(h) - L| \leq K|G(h)|, \quad \text{for sufficiently small } \ h,$$

then we write $F(h) = L + O(G(h))$.

Polynomial rate of convergence

Normally we will use

$$G(h) = h^p,$$

and look for the largest value $p > 0$ such that $F(h) = L + O(h^p)$.

Reading Assignment

- Numerical Analysis, Ch.1, Ch.2.1