Numerical Computations

Hamid Sarbazi-Azad &
Samira Hossein Ghorban
Department of Computer Engineering
Sharif University of Technology (SUT)
Tehran, Iran

What are numerical methods/computations?

***Definition**:* Study of approximation techniques for solving mathematical problems numerically.

Not all the (in fact, very few) mathematical problems have closed-form solutions…

➔ So, numerical methods are necessary.

Examples:

- Extracting the roots of a polynomial with a degree > 4
- Computing the eigenvalues of a matrix ➔PageRank
- Solving most ODEs and PDEs

Why do you need to learn numerical methods?

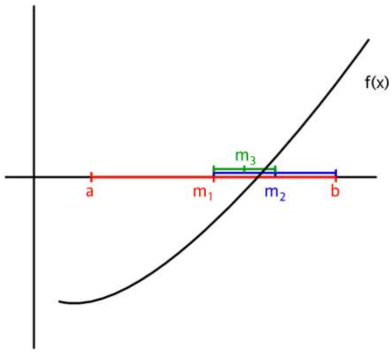
- Most (> 99.9% *) of the real-world problems in science and engineering are complicated and can only be solved numerically.
- Analytic solutions (the math you learned all the way) are in fact very rare. Nevertheless, they are the core truth that helps us understand the structure of the math problems and test the integrity of our numerical schemes.

You need numerical methods in your life as an engineer & as a scientist

Example: Finding the root of a function f(x) in [a, b]

Bisection Method

1. Pick **two** points, a and b, such that f(a) and f(b) have opposite signs.
2. Bisect interval (a, b) into (a, m) and (m, b), where m is the **mid-point** of the original interval. Keep the half interval for which f(x) retains opposite signs at the two end points.
1. **Repeat step 2** until the refined interval is short enough (depending on how accurate you want the solution to be). The mid-point of this interval is the numerical solution.



Consider $ax^3 + bx^2 + cx + d = 0$

Analytic solution*

$$q=\frac{1}{3}a_1-\frac{1}{9}a_2^2;\ r=\frac{1}{6}(a_1a_2-3a_0)-\frac{1}{27}a_2^3.$$

If $q^3+r^2>0$, one real root and a pair of complex conjugate roots,

$q^3+r^2=0$, all roots real and at least two are equal,

$q^3+r^2<0$, all roots real (irreducible case).

Let

$$s_1=[r+(q^3+r^2)^{\frac{1}{2}}]^{\frac{1}{3}},\ s_2=[r-(q^3+r^2)^{\frac{1}{2}}]^{\frac{1}{3}}$$

then

$$z_1=(s_1+s_2)-\frac{a_2}{3}$$

$$z_2=-\frac{1}{2}(s_1+s_2)-\frac{a_2}{3}+\frac{i\sqrt{3}}{2}(s_1-s_2)$$

$$z_3=-\frac{1}{2}(s_1+s_2)-\frac{a_2}{3}-\frac{i\sqrt{3}}{2}(s_1-s_2).$$

If z_1, z_2, z_3 are the roots of the cubic equation

$$z_1+z_2+z_3=-a_2$$

$$z_1z_2+z_1z_3+z_2z_3=a_1$$

$$z_1z_2z_3=-a_0$$

Numerical solution

Bisection method

applies ☺

* From Abramowitz & Stegun's *Handbook*

Now consider $ax^4 + bx^3 + cx^2 + dx + e = 0$

Analytic solution*

$$u^3-a_2u^2+(a_1a_3-4a_0)u-(a_1^2+a_0a_3^2-4a_0a_2)=0$$

and determine the four roots of the quartic as solutions of the two quadratic equations

$$v^2+\left[\frac{a_3}{2}\mp\left(\frac{a_3^2}{4}+u_1-a_2\right)^{\frac{1}{2}}\right]v+\frac{u_1}{2}\mp\left[\left(\frac{u_1}{2}\right)^2-a_0\right]^{\frac{1}{2}}=0$$

If all roots of the cubic equation are real, use the value of u_1 which gives real coefficients in the quadratic equation and select signs so that if

$$z^4+a_3z^3+a_2z^2+a_1z+a_0=(z^2+p_1z+q_1)(z^2+p_2z+q_2),$$

then

$$p_1+p_2=a_3,\ p_1p_2+q_1+q_2=a_2,\ p_1q_2+p_2q_1=a_1,\ q_1q_2=a_0.$$

If z_1, z_2, z_3, z_4 are the roots,

$$\Sigma z_i=-a_3,\ \Sigma z_iz_jz_k=-a_1,$$

$$\Sigma z_iz_j=a_2,\ z_1z_2z_3z_4=a_0.$$

Numerical solution

Bisection method

applies ☺

No need to change anything ☺

* From Abramowitz & Stegun's *Handbook*

Now consider $ax^5 + bx^4 + cx^3 + dx^2 + ex + f = 0$

Analytic solution*	Numerical solution
Does not exist ☹ except for a few specific special cases.	Bisection method applies and works as usual ☺ **At this point, numerical methods become the only option.**

Numerical methods in AI (artificial intelligence)

Local search in *continuous* spaces*

- Suppose we want to place 3 new airports anywhere in the map, such that the sum of squared distances from each city on the map to its nearest airport is minimized.
- The state space is defined by the locations of the three airports: $(x_1, y_1), (x_2, y_2), (x_3, y_3)$
- We can model the problem by this function:

$$f(x_1, x_2, x_3, y_1, y_2, y_3) = \sum_{i=1}^{3} \sum_{c \in C_i} (x_i - x_c)^2 + (y_i - y_c)^2$$

We should minimize *f*.

* from Artificial Intelligence: A Modern Approach, 3rd ed.

Numerical methods in AI (cont'd)

In order to minimize f we have to use the gradient method:

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial x_3}, \frac{\partial f}{\partial y_1}, \frac{\partial f}{\partial y_2}, \frac{\partial f}{\partial y_3}\right)$$

and solve equation $\nabla f = 0$.

The most effective algorithm is **Newton-Raphson** method, which is a venerable numerical method.

$$\boldsymbol{x} \leftarrow \boldsymbol{x} - H_f^{-1}(\boldsymbol{x})\nabla f(\boldsymbol{x})$$

We will see **Newton-Raphson** method to solve such problems!

Numerical methods in image processing

Eigenvalues play an important role in image processing applications, such as:

- Measurement of image sharpness
- Human face segmentation
- Whitening

You can understand how they work only after you study the math behind them.

We will learn about eigenvalue through this course.

(c) H Sarbazi-Azad & S Hossein Ghorban Numerical Computations - Lecture#1: Introduction 11

More real-world computational problems

The world's largest matrix computation:

Google's PageRank is an eigenvector of a matrix of order 3 billion*

Airlines use optimization algorithms to decide ticket prices, airplane, fuel needs, and crew assignments**

* http://www.mathworks.com/moler/exm/chapters/pagerank.pdf

** The Global Airline Industry, Peter Belobaba, Amedeo Odoni and Cynthian Barnhart , WILEY

(c) H Sarbazi-Azad & S Hossein Ghorban Numerical Computations - Lecture#1: Introduction 12

More real-world computational problems (cont'd)

Car companies can improve the crash safety by using computer simulations. These simulations are essentially solving partial differential equations numerically*

Hedge funds (private investment funds) use tools from different fields of numerical analysis to calculate the value of stocks*

* Computer Science An Overview, Wikipedians, Edited by Paul Muljadi

Supercomputers: The most powerful computation machines

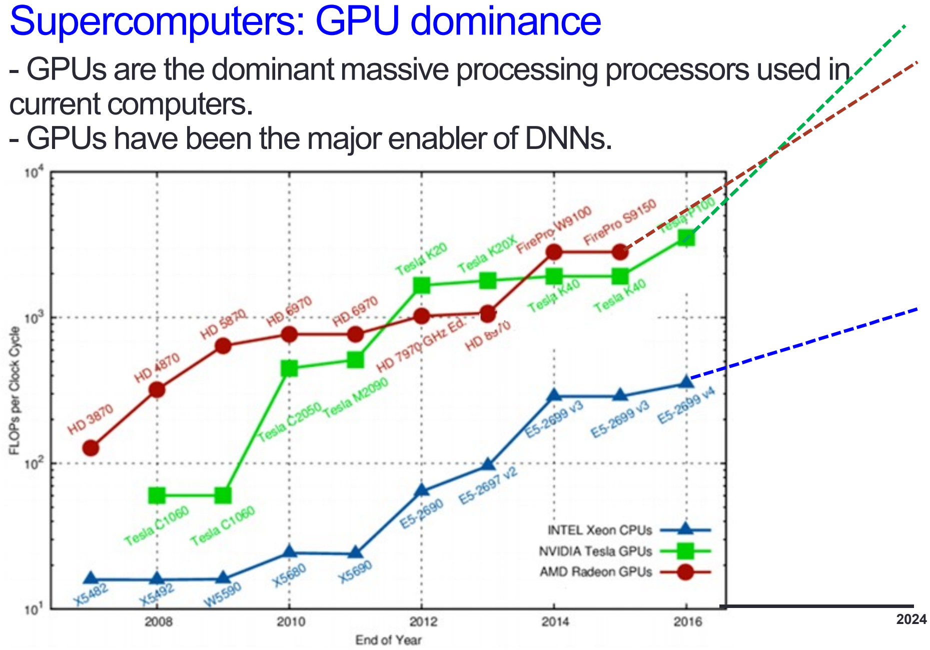
Advances in integrated circuits technology (i.e. Moore's law) and parallel computer architectures have fuelled the thirst for building powerful computers, aka supercomputers, during the last seven decades.

Supercomputers are evaluated every six months and listed in www.top500.org based on computing power in FLOPS (Floating-point* Operations Per Second).

* Remember IEEE 754 Floating-point format! Adding/multiplying two IEEE 754 floating-point numbers is considered a FLOP.

Supercomputers: GPU dominance

- GPUs are the dominant massive processing processors used in current computers.
- GPUs have been the major enabler of DNNs.



Supercomputers: current top 10 machines

Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	**Frontier** - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,730,112	1,102.00	1,685.65	21,100
2	**Supercomputer Fugaku** - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,630,848	442.01	537.21	29,899
3	**LUMI** - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE EuroHPC/CSC Finland	1,110,144	151.90	214.35	2,942
4	**Summit** - EDR Infin DOE/SC/ United St			200.79	10,096
5	**Sierra** - I EDR Infin DOE/NNS United St			25.71	7,438
6	**Sunway** National S China			25.44	15,371
7	**Perlmutt** 10, HPE DOE/SC/ United States			93.75	2,589
8	**Selene** - NVIDIA DGX A100, AMD EPYC 7742 64C 2.25GHz, NVIDIA A100, Mellanox HDR Infiniband, Nvidia NVIDIA Corporation United States	555,520	63.46	79.22	2,646
9	**Tianhe-2A** - TH-IVB-FEP Cluster, Intel Xeon E5-2692v2 12C 2.2GHz, TH Express-2, Matrix-2000, NUDT National Super Computer Center in Guangzhou China	4,981,760	61.44	100.68	18,482
10	**Adastra** - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE Grand Equipement National de Calcul Intensif - Centre Informatique National de l'Enseignement Suprieur (France	319,072	46.10	61.61	921

Today, computer engineers must know parallel programming, especially **GPU programming**.

References

1. جزوه محاسبات عددی، ترجمه حمید سربازی آزاد.

ترجمه کتاب:

N. V. Kopchenova, I. A. Maron, Computational Mathematics, Mir Publisher.

2. NVIDIA, Nvidia CUDA C Programming Guide, Version 6.5, 2014.

3. J. Sanders, E. Kandrot, CUDA by Example: an introduction to general-purpose GPU programming, NVIDIA, 2010.

Course outline

Section 0: Introduction to NC & Parallel programming with GPUs

Section 1: Approximate computations and error estimation

Section 2+: Computing the values of functions **+ solving equations**

Section 3: Numerical solution of systems of linear equations

Section 4: Numerical solution of systems of nonlinear equations

Section 5: Interpolation of functions

Section 6: Numerical differentiation

Section 7: Numerical integration

Section 8: Approximate solution of ordinary differential equations

Section 9: Boundary value problems for ordinary differential equations

Section 10: Numerical solution of equations with partial derivatives and of integral equations.

NC course team

Lecturers: Hamid Sarbazi-Azad & Samira Hossein Ghorban
Emails: azad@{sharif.edu / ipm.ir} & s.hosseinghorban@ipm.ir
Phones: 66166622, 66166650
Rooms: 621, 817

Head TA: Armin Ahmadzadeh
Email: a.ahmadzadeh@ipm.ir
Phone: 66166672 & 24509409
Room: HPCAN Laboratory at Sharif CE (703) and IPM HPC (Level 3)

Grading policy

Assignments* 50%

* All communicated/handled on CW.

- Homework assignments 30%
- Programming exercises** 20%

** Extra scores for GPU implementation and a prize for the best!

➔ **NO EMAILS, NO EXCUSES, PLEASE !!!**

Midterm Exams (Sections 1-4) 25%

Thursday 3rd Azar

Final Exam (Sections 5-8) 25%

ANY QUESTIONS?