Cursul 2

- Definiție (Precizia mașinii)

Cea mai mare eroare relativă posibilă a rotunjirii unui număr la formatul de virgulă mobilă cu t cifre binare se numește precizia mașinii:

$$\varepsilon_M := \max_x |\varepsilon_r(fl(x))| = \max_x \left| \frac{x - fl(x)}{x} \right| = 2^{-t}$$

- Exemple:

Precizia unei mașini cu 8 biți este de $2^{-4} = 0.0625$

I) Rezolvarea sistemelor de ecuații liniare pătratice

$$\begin{cases} a_{00}x_0 + a_{01}x_1 + \dots + a_{0\,n-1}x_{n-1} = b_0 \\ a_{10}x_0 + a_{11}x_1 + \dots + a_{1\,n-1}x_{n-1} = b_1 \\ \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \\ a_{n-1\,0}x_0 + a_{n-1\,1}x_1 + \dots + a_{n-1\,n-1}x_{n-1} = b_{n-1} \end{cases} \quad (*)$$

$x_0, x_1 \dots x_{n-1} \in \mathbb{R}$ sunt necunoscute

- Rescriem sistemul de ecuații liniare pătratice într-un mod mai compact

$$x = (x_0 \;\; x_1 \dots x_{n-1})^T \in \mathbb{R}^n \text{ (vector)}$$

$$b = (b_0 \;\; b_1 \dots b_{n-1})^T \in \mathbb{R}^n \text{ (vector)}$$

$$A = \begin{bmatrix} a_{00} & a_{01} & \dots & a_{0\,n-1} \\ \cdot & \cdot & \cdot & \cdot \\ a_{n-1\,0} & a_{n-1\,1} & \dots & a_{n-1\,n-1} \end{bmatrix} \in \mathcal{M}_n(\mathbb{R}) \text{ (matrice)}$$

$$(*) \iff \boxed{A \cdot x = b}$$

- Când are $A \cdot x = b$ o unică soluție?
Când A este inversabilă, i.e. $\det(A) \neq 0$
- Cum îl rezolvăm?

Metoda Cramer (1750):

$$x_i = \frac{\det(X_i)}{\det(A)}, \forall i = \overline{0, n-1},$$

unde X_i ← matricea A cu b pe coloana i

- De câte operații avem nevoie pentru a rezolva un sistem cu Metoda Cramer?

Calculul determinantului: $O(n!)$

Metoda Cramer: $O(n \cdot n!)$

$n = 20$: Aproximativ $5 \cdot 10^{19}$ operații, pe care un procesor de 2.5 GHz le face în $2 \cdot 10^{10}$ secunde = 633 de ani.

Q: Cum rezolvăm <u>rapid</u> sisteme de ecuații liniare pătratice?

- Am observat la seminar că nu avem nevoie de Cramer pentru a rezolva un sistem de tipul

$$\begin{cases} a_{00}x_0 + a_{01}x_1 + a_{02}x_2 = b_0 \\ \quad a_{11}x_1 + a_{12}x_2 = b_1 \\ \quad\quad a_{22}x_2 = b_2 \end{cases}$$

- Definiție (Matrici triunghiulare)

Matricea $A = (a_{ij})_{i,j=\overline{0,n-1}} \in \mathcal{M}_n(\mathbb{R})$ s.n.

i) Inferior triunghiulară dacă

$$a_{ij} = 0 \quad \forall\ 0 \le i < j \le n-1$$

ii) Superior triunghiulară dacă

$$a_{ij} = 0 \quad \forall\ 0 \le j < i \le n-1$$

- Observație

Fie $A \in \mathcal{M}_n(\mathbb{R})$ o matrice triunghiulară

A inversabilă $\Leftrightarrow \det A = a_{00} \cdot a_{11} \cdot \ldots \cdot a_{n-1\,n-1} \neq 0$

$\Leftrightarrow a_{ii} \neq 0 \quad \forall i = \overline{0, n-1}$

- Algoritm de rezolvare a sistemelor cu matrici superior triunghiulare

$U x = b$, unde $u_{ii} \neq 0$, $\forall i = \overline{0, n-1}$

$\Updownarrow$

$$\begin{cases} u_{00} x_0 + u_{01} x_1 + \dots + u_{0\,n-2} x_{n-2} + u_{0\,n-1} x_{n-1} = b_0 \\ \qquad u_{11} x_1 + \dots + u_{1\,n-2} x_{n-2} + u_{1\,n-1} x_{n-1} = b_1 \\ \qquad \cdots\cdots\cdots \\ \qquad u_{n-2\,n-2} x_{n-2} + u_{n-2\,n-1} x_{n-1} = b_{n-2} \\ \qquad u_{n-1\,n-1} x_{n-1} = b_{n-1} \end{cases}$$

(E_{n-1}) : $x_{n-1} = \dfrac{b_{n-1}}{u_{n-1\,n-1}}$

(E_{n-2}) : $u_{n-2\,n-2}\, x_{n-2} = b_{n-2} - u_{n-2\,n-1}\, x_{n-1}$

$\Rightarrow x_{n-2} = \dfrac{b_{n-2} - u_{n-2\,n-1}\, x_{n-1}}{u_{n-2\,n-2}}$

$\cdots\cdots\cdots$

(E_k) : $\boxed{x_k = \dfrac{b_k - \sum_{i=k+1}^{n-1} u_{ki}\, x_i}{u_{kk}}}$ $\forall k = \overline{0, n-1}$

(Metoda Substituției Descendente)

- Observații:

  i) Analog, se deduce metoda substituției ascendente pentru sisteme cu matrici inferior triunghiulare.

  ii) Complexitatea celor două metode este de doar $O(n^2)$!

  iii) În cele ce urmează, vom căuta o modalitate de a transforma orice sistem de ecuații liniare în sisteme triunghiulare.

- Definiție (Transformări elementare)

Numim tranformări elementare următoarele operații cu ecuațiile unui sistem (liniile matricei asociate și termenii liberi) ce nu modifică soluția acestuia.

i) Permutarea a două ecuații / linii

$$E_i \longleftrightarrow E_k$$

ii) Înmulțirea unei linii / ecuații cu un scalar $\alpha \in \mathbb{R}^*$:

$$E_i \longleftarrow \alpha E_i$$

iii) Adunarea unei linii / ecuații cu o altă linie / ecuație scalată cu $\alpha \in \mathbb{R}^*$:

$$E_i \longleftarrow E_i + \alpha E_k$$

• Exemplu: Fie sistemul liniar de ecuații:

$$\begin{cases} x_0 + x_1 + x_2 = 6 \\ 2x_0 + 4x_1 + 2x_2 = 16 \\ -x_0 + 5x_1 - 4x_2 = -3 \end{cases}$$

$$\Updownarrow$$

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 \\ 2 & 4 & 2 \\ -1 & 5 & -4 \end{bmatrix}}_{A} \cdot \underbrace{\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} 6 \\ 16 \\ -3 \end{bmatrix}}_{b}$$

- Pasul $k=0$: Facem 0-uri sub coloana 0 A

$$E_1 \leftarrow E_1 - 2E_0 = E_1 - \frac{a_{10}}{a_{00}} E_0$$

$$E_2 \leftarrow E_2 + E_0 = E_2 - \frac{a_{20}}{a_{00}} E_0$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 6 & -3 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 4 \\ 3 \end{bmatrix}$$

- Pasul $k=1$: Facem 0-uri sub coloana 1 A

$$E_2 \leftarrow E_2 - 3E_1 = E_2 - \frac{a_{21}}{a_{11}} E_1$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & -3 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 4 \\ -9 \end{bmatrix}$$

- Aplicăm metoda substituției descendente

$x_2 = 3$, $x_1 = 2$, $x_0 = 1$

- Ideea apare prima dată în timpul dinastiei Han (sec. I, Fang cheng) și e generalizată de Gauss (1809)

- Algoritm de transformare a unui sistem de ecuații liniare pătratic oarecare într-un sistem de ecuații superior triunghiular: Metoda de eliminare Gauss (MEG)

Input: $A=(a_{ij})_{0\le i,j\le n-1}$ cu minorii principali nenuli

$$b=(b_i)_{0\le i\le n-1}$$

- Pentru k de la 0 la $n-2$:
  - Pentru i de la $k+1$ la $n-1$:
    - $b_i = b_i - \dfrac{a_{ik}}{a_{kk}}\, b_k$
    - Pentru j de la k la $n-1$:
      - $a_{ij} = a_{ij} - \dfrac{a_{ik}}{a_{kk}}\, a_{kj}$

Output: A superior triunghiulară și b modificat

- Complexitate: $O(n^3)$! Pentru $n=20$ și un procesor de $2.5\ GHz$: $\dfrac{8\cdot 10^3}{2.5\cdot 10^9} = 3.2\times 10^{-6}\,s$!

- Exemple de sisteme ce nu pot fi transformate cu MEG

i) $\begin{bmatrix} 0 & 1 \\ -2 & 4 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} -3 \\ 6 \end{bmatrix}$

ii) $\begin{bmatrix} 1 & 1 & 2 \\ 2 & 2 & -1 \\ -2 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ -1 \end{bmatrix}$

iii) $\begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $0 < \varepsilon << 1$ (ex.: $\varepsilon = 10^{-18}$)

$$E_1 \leftarrow E_1 - \frac{1}{\varepsilon} E_0$$

$$\begin{bmatrix} \varepsilon & 1 \\ 0 & 1 - \frac{1}{\varepsilon} \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 - \frac{1}{\varepsilon} \end{bmatrix}$$

$$x_1 = \frac{2 - \frac{1}{\varepsilon}}{1 - \frac{1}{\varepsilon}} = \frac{2 - 10^{18}}{1 - 10^{18}} \approx 1$$

$$\varepsilon x_0 = 1 - 1 = 0 \Rightarrow x_0 = 0$$

Verificare: $\begin{cases} \varepsilon \cdot 0 + 1 \cdot 1 = 1 \checkmark \\ 1 \cdot 0 + 1 \cdot 1 = 2 \times \end{cases}$