Laborator 5

Condensarea pivotală pentru calculul determinanţilor (Metoda lui Chio)

Prezentarea Problemei: Considerăm matricea $A = (a_{ij})_{1 \le i,j \le n} \in \mathbb{R}^{n \times n}$ şi ne propunem să calculăm $\det(A)$.

Prezentarea Metodei: Aplicăm formula

$$\det(A) = \frac{1}{a_{11}^{n-2}} \begin{vmatrix} \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} & \begin{vmatrix} a_{11} & a_{13} \\ a_{21} & a_{23} \end{vmatrix} & \dots & \begin{vmatrix} a_{11} & a_{1n} \\ a_{21} & a_{2n} \end{vmatrix} \\ \begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} & \begin{vmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{vmatrix} & \dots & \begin{vmatrix} a_{11} & a_{1n} \\ a_{31} & a_{3n} \end{vmatrix} \\ \vdots & \vdots & \dots & \vdots \\ \begin{vmatrix} a_{11} & a_{12} \\ a_{n1} & a_{n2} \end{vmatrix} & \begin{vmatrix} a_{11} & a_{13} \\ a_{n1} & a_{n3} \end{vmatrix} & \dots & \begin{vmatrix} a_{11} & a_{1n} \\ a_{n1} & a_{nn} \end{vmatrix} \end{vmatrix}, \tag{1}$$

unde $a_{11} \neq 0$, şi în continuare se reia formula (1) pentru $n-1, n-2, \dots$ până când se obţine un determinant de ordin 2.

Observatii:

1.Dacă $a_{11} = 0$ şi există $2 \le i \le n$ pentru care $a_{i1} \neq 0$, atunci se permută în A liniile a şi i, iar $\det(A)$ îsi schimbă semnul.

2. Dacă $a_{11} = 0$, $\forall 2 \le i \le n$, avem $a_{i1} = 0$, atunci $\det(A) = 0$.

Algoritmul Pseudocod

1. citeşte n, a_{ij}, $1 \le i, j \le n$
2. $det \leftarrow 1$
3. repetă
   - 3.1. dacă $a_{11} = 0$ atunci
     - 3.1.1. $i \leftarrow 2$
     - 3.1.2. cât timp $(i \le n)$ şi $(a_{i1} = 0)$ execută
       - 3.1.2.1. $i \leftarrow i + 1$
     - 3.1.3. dacă $i > n$ atunci
       - 3.1.3.1. scrie $'det(A) = 0'$
       - 3.1.3.2. *ieşire*
     - 3.1.4. pentru $j = 1, 2, ..., n$ execută
       - 3.1.4.1. $aux \leftarrow a_{1j}$
       - 3.1.4.2. $a_{1j} \leftarrow a_{ij}$
       - 3.1.4.3. $a_{ij} \leftarrow aux$
     - 3.1.5. $det \leftarrow -det$
   - 3.2. pentru $i = 1, 2, ..., n-2$ execută // calculăm produsul valorilor a_{11}^{n-2}
     - 3.2.1. $det \leftarrow det \cdot a_{11}$
   - 3.3. pentru $i = 2, 3, ..., n$ execută // aplicăm $a_{ij} \leftarrow \begin{vmatrix} a_{11} & a_{1j} \\ a_{i1} & a_{ij} \end{vmatrix}$, $2 \le i, j \le n$

3.3.1. pentru $j = 2, 3, ..., n$ execută

3.3.1.1. $a_{ij} \leftarrow a_{ij} \cdot a_{11} - a_{i1} \cdot a_{1j}$

3.4. $n \leftarrow n - 1$

3.5. pentru $i = 1, 2, ..., n$ execută

3.5.1. pentru $j = 1, 2, ..., n$ execută

3.5.1.1. $a_{ij} \leftarrow a_{i+1,j+1}$

până când $(n = 1)$

4. $det \leftarrow a_{11}/det$

5. scrie $'det(A) =', det$.

Exemple: Calculaţi determinanţii următoarelor matrice

$$A = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 2 & 0 & 1 \\ 0 & 1 & 1 & 3 \\ 2 & 1 & -2 & -1 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 1 & 0 & 1 & 2 \\ 6 & 3 & 2 & -1 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 1 & 1 & -2 & 3 & 0 \\ 0 & 0 & 0 & -1 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 4 & 1 & 3 & 1 \\ 1 & 5 & 4 & 1 \\ 1 & 1 & 10 & 6 \\ 0 & 1 & 1 & 10 \end{pmatrix}.$$

Metoda Jacobi pentru rezolvarea iterativă a sistemelor de ecuaţii liniare

Prezentarea Problemei: Considerăm sistemul liniar

$$A \cdot x = b, \tag{2}$$

unde $A = (a_{ij})_{1 \le i,j \le n} \in \mathbb{R}^{n \times n}$ este matricea sistemului (2) şi $b = (b_i)_{1 \le i \le n} \in \mathbb{R}^n$ este termenul liber al sistemului (2).
Ne propunem să aproximăm $x \in \mathbb{R}^n$ soluţia unică a sistemului (2) cu o precizie dată ε.

Prezentarea Metodei: Pentru $x^{(0)} \in \mathbb{R}^n$ aproximaţia iniţială a soluţiei sitemului (2), ales arbitrar (de exemplu vectorul nul), calculăm

$$x_i^{(k+1)} = \left(b_i - \sum_{\substack{j=1 \\ j \ne i}}^{n} a_{ij} x_j^{(k)} \right) \Big/ a_{ii}, 1 \le i \le n,\ k \ge 0,$$

până când este îndeplinită condiţia

$$\max_{1 \le i \le n} \left| x_i^{(k+1)} - x_i^{(k)} \right| \le \varepsilon,$$

unde ε reprezintă precizia cu care dorim să obţinem soluţia sistemului. Atunci $x \simeq x^{(k+1)}$.
Observatie: O condiţie suficientă pentru obţinerea soluţiei sitemului (2), cu precizia ε, este ca matricea A să fie dominant diagonală pe linii sau coloane şi strict dominant diagonală pe cel puţin una din linii sau coloane.

Algoritmul Pseudocod

1. citeşte n, a_{ij}, $1 \le i, j \le n$, b_i, $1 \le i \le n$, ε, $itmax$, x_i, $1 \le i \le n$
2. $it \leftarrow 0$

3. repetă

  3.1. $max \leftarrow 0$

  3.2. pentru $i = 1, 2, ..., n$ execută

    3.2.1. $S \leftarrow 0$

    3.2.2. pentru $j = 1, 2, ..., n$ execută

      3.2.2.1. dacă $j \neq i$ atunci

        3.2.2.1.1. $S \leftarrow S + a_{ij} \cdot x_j$

    3.2.3. $y_i \leftarrow (b_i - S)/a_{ii}$

    3.2.4. dacă $max < |y_i - x_i|$ atunci

      3.2.4.1. $max \leftarrow |y_i - x_i|$

  3.3. pentru $i = 1, 2, ..., n$ execută

    3.3.1. $x_i \leftarrow y_i$

  3.4. $it \leftarrow it + 1$

până când $(max \leq \varepsilon)$ sau $(it > itmax)$

4. dacă $it > itmax$ atunci

  4.1. scrie *'Nu se poate obţine soluţia în',* itmax, *'iteraţii, cu precizia',* ε

  4.2. *ieşire*

5. scrie (*'Soluţia obţinută în* ',it, *'iterţii cu precizia',*ε, 'este', x_i, $1 \leq i \leq n$)

Exemple: Să se determine soluţiile următoarelor sisteme cu preciziile $\varepsilon = 10^{-4}, \varepsilon = 10^{-7}, \varepsilon = 10^{-10}$

$$a) \begin{cases} 4x_1 - x_2 + x_3 = 8 \\ 2x_1 + 5x_2 + 2x_3 = 3 \\ x_1 + 3x_2 + 5x_3 = 13 \end{cases} \quad b) \begin{cases} 2x_1 + x_2 = -30 \\ x_1 + 4x_2 + x_3 = -132 \\ x_2 + 4x_3 + x_4 = -132 \\ x_3 + 2x_4 = -30 \end{cases} \quad c) \begin{cases} 4x_1 - x_2 - x_3 = 10 \\ -2x_1 + 4x_2 - x_4 = 10 \\ -2x_1 + 4x_3 - x_4 = 20 \\ -2x_2 - 2x_3 + 4x_4 = 0. \end{cases}$$

Solution: a) Pentru $\varepsilon = 10^{-4}, x^{(0)} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \Rightarrow$ it $= 17$ şi $x = \begin{pmatrix} 1.000002 \\ -0.999978 \\ 3.000026 \end{pmatrix}$,

b) Pentru $\varepsilon = 10^{-4}, x^{(0)} = 0_4 \Rightarrow$ it $= 20$ şi $x = \begin{pmatrix} -1.99998283 \\ -25.99998283 \\ -25.99998283 \\ -1.99998283 \end{pmatrix}$.