Algoritmi paraleli și distribuiți

Sisteme de ecuații liniare

Mitică Craus

Universitatea Tehnică "Gheorghe Asachi" din Iași

Cuprins

Introducere

- Sistemele de ecuții liniare pot fi rezolvate prin tehnici de calcul paralel datorită multiplelor operații independente una de alta.
- Vom studia în această lecție metoda lui Gauss de rezolvare a sistemelor de ecuații liniare.

Rezolvarea sistemelor de ecuații liniare prin metoda lui Gauss

- Se consideră un sistem format din n ecuații liniare cu n necunoscute:

$$\begin{aligned}
a_{0,0}x_0 + a_{0,1}x_1 + \cdots + a_{0,n-1}x_{n-1} &= b_0 \\
a_{1,0}x_0 + a_{1,1}x_1 + \cdots + a_{1,n-1}x_{n-1} &= b_1 \\
\cdots\cdots\cdots & \\
a_{n-1,0}x_0 + a_{n-1,1}x_1 + \cdots + a_{n-1,n-1}x_{n-1} &= b_{n-1}
\end{aligned}$$

- Metoda lui Gauss constă în reducerea sistemului la forma triunghiulară

$$\begin{aligned}
x_0 + a'_{0,1}x_1 + \cdots + a'_{0,n-1}x_{n-1} &= b'_0 \\
x_1 + \cdots + a'_{1,n-1}x_{n-1} &= b'_1 \\
\cdots\cdots\cdots & \\
x_{n-1} &= b'_{n-1}
\end{aligned}$$

- și rezolvarea ecuațiilor în ordine inversă (întâi ultima, apoi penultima și la sfârșit prima ecuație din sistem)

Algoritmul eliminării Gaussiene - varianta secvențială

- *Notații*:
  - $A[0..n-1,0..n-1]$ este un tablou bidimensional, de dimensiune nxn.
  - $B[0..n-1]$ este un tablou unidimensional, de dimensiune n.
- *Premise*:
  - Coeficienții inițiali $a_{i,j}, i,j = 0,1,\dots,n-1$ sunt memorați în tabloul A.
  - Coeficienții $a'_{i,j}, i,j = 0,1,\dots,n-1$ vor fi memorați tot în tabloul A.
  - Tabloul B va memora coeficienții inițiali $b_i, i = 0,1,\dots,n-1$ și soluția sistemului.

ELIMINARE_GAUSSIANA(A,B,n)

for $k \leftarrow 0$ **to** $n-1$ /* bucla exterioară */
do
    for $j \leftarrow k+1$ **to** $n-1$
    do $A[k,j] \leftarrow \frac{A[k,j]}{A[k,k]}$ /* pasul de impărțire cu $A[k,k]$ */
    $B[k] \leftarrow \frac{B[k]}{A[k,k]}$
    $A[k,k] \leftarrow 1$
    for $i \leftarrow k+1$ **to** $n-1$
    do for $j \leftarrow k+1$ **to** $n-1$
        do $A[i,j] \leftarrow A[i,j] - A[i,k]xA[k,j]$ /* pasul de eliminare */
        $B[i] \leftarrow B[i] - A[i,k]xB[k]$
        $A[i,k] \leftarrow 0$

Algoritmul eliminării Gaussiene - varianta paralelă

- *Notații*:
  - $A[0..n-1, 0..n-1]$ este un tablou bidimensional, de dimensiune nxn.
  - $B[0..n-1]$ este un tablou unidimensional, de dimensiune n.
- *Premise*:
  - Coeficienții inițiali $a_{i,j}, i,j = 0,1,\ldots,n-1$ sunt memorați în tabloul A.
  - Coeficienții $a'_{i,j}, i,j = 0,1,\ldots,n-1$ vor fi memorați tot în tabloul A.
  - Tabloul B va memora coeficienții inițiali $b_i, i = 0,1,\ldots,n-1$ și soluția sistemului.

```
ELIMINARE_GAUSSIANA_PARALELA(A, B, n)
for k ← 0 to n−1 /* bucla exterioară */
do
    for j ← k+1 to n−1
    do A[k,j] ← A[k,j] / A[k,k]   /* pasul de impărțire cu A[k,k] */
    B[k] ← B[k] / A[k,k]
    A[k,k] ← 1
    for i ← k+1 to n−1
    do  in parallel
        for j ← k+1 to n−1
        do A[i,j] ← A[i,j] − A[i,k]xA[k,j]   /* pasul de eliminare */
        B[i] ← B[i] − A[i,k]xB[k]
        A[i,k] ← 0
```

Implementare pe un sistem cu n unități de procesare

- Se consideră un sistem format din n unități de procesare, $p_0, p_1, \ldots, p_{n-1}$.
- Fiecare unitate de procesare $p_i, i \in \{0, 1, \ldots, n-1\}$, dispune, în memoria locală, de coeficienții ecuatiei a-i-a (linia i a tabloului A plus $B[i]$).
- În prima fază, unitatea de procesare p_0 execută pasul de imparțirea cu $A[0,0]$: $A[0,j] = \frac{A[0,j]}{A[0,0]}, j = 0, \ldots, n-1$. Apoi trimite $A[0,j], j = 0, 1, \ldots, n-1$ celorlalte unități de procesare, după care, fiecare unitate de procesare $p_i, i = 1 \ldots, n-1$ execută pasul de eliminare: $A[i,j] = A[i,j] - A[i,0]xA[0,j], i, j = 1, \ldots, n-1$ și $A[i,0] = 0, i = 1 \ldots, n-1$.
- În faza a doua, p_1 preia rolul lui p_0 și execută imparțirea cu $A[1,1]$: $A[1,j] = \frac{A[1,j]}{A[1,1]}, j = 1 \ldots, n-1$. Apoi trimite $A[1,j], j = 1 \ldots, n-1$ unităților de procesare $p_2, \ldots, p_{n-1}$, după care, fiecare unitate de procesare $p_i, i = 2 \ldots, n-1$ execută pasul de eliminare: $A[i,j] = A[i,j] - A[i,1]xA[1,j], i, j = 2, \ldots, n-1$ și $A[i,1] = 0, i = 2, \ldots, n-1$.
- În faza k, p_k execută imparțirea cu $A[k,k]$: $A[k,j] = \frac{A[k,j]}{A[k,k]}, j = k \ldots, n-1$. Apoi trimite $A[k,j], j = k \ldots, n-1$ unităților de procesare $p_{k+1}, \ldots, p_{n-1}$, după care, fiecare unitate de procesare $p_i, i = k+1 \ldots, n-1$ execută pasul de eliminare: $A[i,j] = A[i,j] - A[i,k]xA[k,j], i, j = k+1, \ldots, n-1$ și $A[i,k] = 0, i = k+1, \ldots, n-1$.

Exemplu de implementare pe un sistem cu 8 unități de procesare

P_0	1	(0,1)	(0,2)	(0,3)	(0,4)	(0,5)	(0,6)	(0,7)
P_1	0	1	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)
P_2	0	0	1	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)
P_3	0	0	0	(3,3)	(3,4)	(3,5)	(3,6)	(3,7)
P_4	0	0	0	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)
P_5	0	0	0	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)
P_6	0	0	0	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)
P_7	0	0	0	(7,3)	(7,4)	(7,5)	(7,6)	(7,7)

Figura 1 : Pasul de imparțire cu $A[3,3]$:

$$A[3,j] = \frac{A[3,j]}{A[3,3]}, j = 4, \ldots 7, \text{ și } A[3,3] = 1$$

Exemplu de implementare pe un sistem cu 8 unități de procesare - continuare

P_0	1	(0,1)	(0,2)	(0,3)	(0,4)	(0,5)	(0,6)	(0,7)
P_1	0	1	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)
P_2	0	0	1	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)
P_3	0	0	0	1	(3,4)	(3,5)	(3,6)	(3,7)
P_4	0	0	0	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)
P_5	0	0	0	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)
P_6	0	0	0	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)
P_7	0	0	0	(7,3)	(7,4)	(7,5)	(7,6)	(7,7)

Figura 2 : Broadcast linia 3: $A[3,j], j = 4, \ldots, 7$

Exemplu de implementare pe un sistem cu 8 unități de procesare - continuare

P_0	1	(0,1)	(0,2)	(0,3)	(0,4)	(0,5)	(0,6)	(0,7)
P_1	0	1	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)
P_2	0	0	1	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)
P_3	0	0	0	1	(3,4)	(3,5)	(3,6)	(3,7)
P_4	0	0	0	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)
P_5	0	0	0	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)
P_6	0	0	0	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)
P_7	0	0	0	(7,3)	(7,4)	(7,5)	(7,6)	(7,7)

Figura 3 : Pasul de eliminare:

$A[i,j] = A[i,j] - A[i,3]xA[3,j], i,j = 4,\ldots,7$, și
$A[i,3] = 0, i = 4,\ldots,7$

Complexitatea timp a implementării algoritmului eliminării Gaussiene pe un lanț de n unități de procesare

Teorema (1)

Complexitatea timp a implementării algoritmului eliminării Gaussiene pe un lanț de n unități de procesare este $\Theta(n^2)$.

Demonstrație.

Timpul consumat în iterația k este $\Theta(n-k-1)$. Timpul total este $\sum_{k=0}^{k=n-1}\Theta(n-k-1)$ $=\Theta(n-1)+\Theta(n-1)+\cdots+\Theta(0)=\Theta(n^2)$. □

Comentarii bibliografice

- Capitolul *Sisteme de ecuații liniare* are la bază cartea
  V. Kumar, A. Grama A. Gupta & G Karypis, Introduction to Parallel Computing: Design and Analysis of Algorithms, Addison Wesley, 2003
  și ediția mai veche
  V. Kumar, A. Grama A. Gupta & G Karypis, Introduction to Parallel Computing: Design and Analysis of Algorithms, Benjamin-Cummings, 1994