Teoria współbieżności

Eliminacja Gaussa

Kamil Koczera

1.1 Niepodzielne operacje:

Zdefiniowane zostały następujące niepodzielne operacje:

- $A_{i,k}$ – wykonanie dzielenia $M_{k,i}/M_{i,i} \rightarrow m_{k,i}$, dzięki czemu zostaje znaleziony mnożnik dla wiersza i, do odejmowania go od k-tego wiersza
- $B_{i,j,k}$ – wykonanie mnożenia $M_{i,j} * m_{k,i} \rightarrow n_{k,j,i}$, co oznacza pomnożenie j-tego elementu wiersza i przez mnożnik - do odejmowania od k-tego wiersza
- $C_{i,j,k}$ – wykonanie odejmowania $M_{k,j} - n_{k,j,i} \rightarrow M_{k,j}$, co oznacza odjęcie j-tego elementu wiersza i od wiersza k

1.2 Alfabet:

Alfabet został zdefiniowany w następujący sposób (n to rozmiar macierzy):

$$\Sigma_A = \{A_{i,k} \mid i = 1, 2, \dots, n-1;\ k = i+1, i+2, \dots, n\}$$

$$\Sigma_B = \{B_{i,j,k} \mid i = 1, 2, \dots, n-1;\ j = i, i+1, \dots, n+1;\ k = i+1, i+2, \dots, n\}$$

$$\Sigma_C = \{C_{i,j,k} \mid i = 1, 2, \dots, n-1;\ j = i, i+1, \dots, n+1;\ k = i+1, i+2, \dots, n\}$$

$$\Sigma = \Sigma_A \cup \Sigma_B \cup \Sigma_C$$

1.3 Relacja (nie)zależności

Relacja zależności została sformułowana w następujący sposób:

$$D_1 = \{(A_{i,k}, B_{i,j,k}) | A_{i,k}, B_{i,j,k} \in \Sigma\}$$

$$D_2 = \{(B_{i,j,k}, C_{i,j,k}) | B_{i,j,k}, C_{i,j,k} \in \Sigma\}$$

$$D_3 = \{(C_{i1,j,k1}, A_{i2,k2}) | A_{i2,k2}, C_{i1,j,k1} \in \Sigma \wedge \mathrm{j} = \mathrm{i2} \wedge (\mathrm{i1} \neq \mathrm{i2} \vee \mathrm{k1} \neq \mathrm{k2}) \wedge (\mathrm{k1} = \mathrm{i2} \vee \mathrm{k1} = \mathrm{k2})\}$$

$$D_4 = \{(C_{i1,j,k1}, B_{i2,j,k2}) | B_{i2,j,k2}, C_{i1,j,k1} \in \Sigma \wedge \mathrm{i2} = \mathrm{k1} \vee (\mathrm{j} = \mathrm{i2} \wedge \mathrm{j} = \mathrm{k2} - 1 \wedge \mathrm{k2} = \mathrm{k1} \wedge \mathrm{i1} \neq \mathrm{i2})\}$$

$$D_5 = \{(C_{i1,j,k}, C_{i2,j,k}) | C_{i1,j,k}, C_{i2,j,k} \in \Sigma\}$$

$$D = D_1 \cup D_2 \cup D_3 \cup D_4 \cup D_5$$

Relacja niezależności została wyprowadzona na podstawie relacji zależności:

$$I = \Sigma^2 - D$$

1.4 Algorytm eliminacji Gaussa w postaci ciągu symboli alfabetu

W celu skonstruowania ciągu operacji, odpowiadającego algorytmowi eliminacji, zostały zdefiniowane następujące podciągi (n to rozmiar macierzy):

$o_{i,k}$ – wyzerowanie elementu $M_{k,i}$ poprzez odjęcie i-tego wiersza od k-tego.

$o_{i,k} = (A_{i,k}, B_{i,i,k}, C_{i,i,k}, B_{i,i+1,k}, C_{i,i+1,k}, \ldots, B_{i,n,k}, C_{i,n,k}, B_{i,n+1,k}, C_{i,n+1,k})$

Zatem cały algorytm eliminacji (bez podstawiania wstecz) można zapisać następująco (z założeniem, że ten zapis nie oznacza ciągu ciągów, a konkatenację ciągów w jeden):

$(o_{1,2}, o_{1,3}, \ldots, o_{1,n}, o_{2,3}, o_{2,4}, \ldots, o_{2,n}, \ldots, o_{n-2,n-1}, o_{n-2,n}, o_{n-1,n})$

1.5 Graf Diekerta

Zbiór G wszystkich bezpośrednich zależności (czyli krawędzi w grafie Diekerta) prezentuje się następująco:

$$G_1 = \{(A_{i,k}, B_{i,j,k}) | A_{i,k}, B_{i,j,k} \in \Sigma\}$$

$$G_2 = \{(B_{i,j,k}, C_{i,j,k}) | B_{i,j,k}, C_{i,j,k} \in \Sigma\}$$

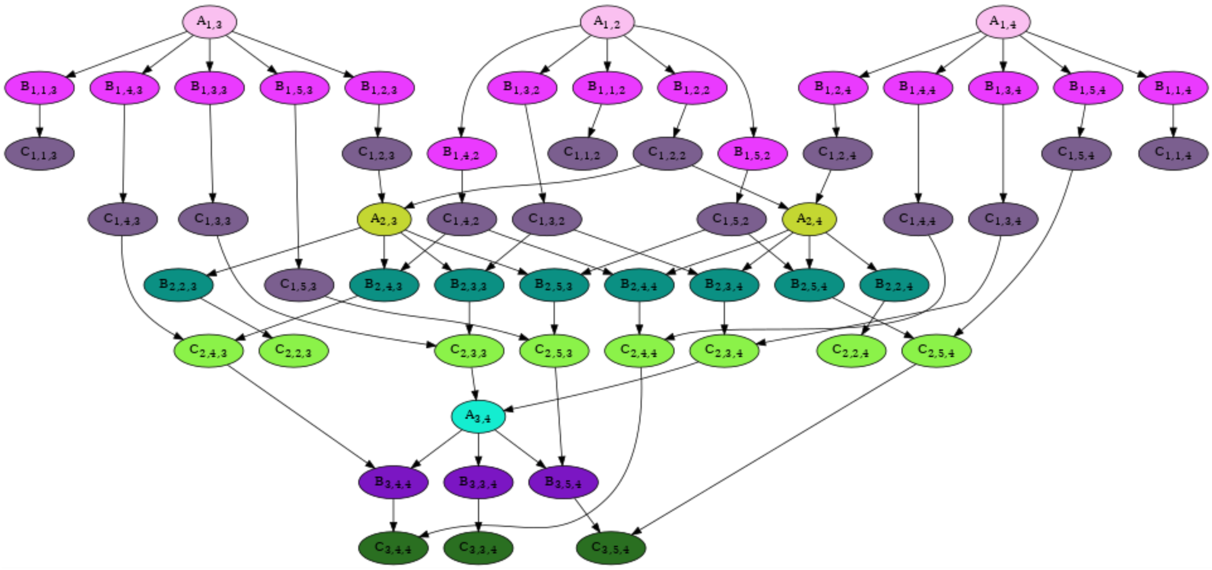
$$G_3 = \{(C_{i1,j,k1}, A_{i2,k2}) | A_{i2,k2}, C_{i1,j,k1} \in \Sigma \wedge \mathrm{j} = \mathrm{i2} \wedge \mathrm{i1} = \mathrm{i2} - 1 \wedge (\mathrm{k1} = \mathrm{i2} \vee \mathrm{k1} = \mathrm{k2})\}$$

$$G_4 = \{(C_{i1,j,k1}, B_{i2,j,k2}) | B_{i2,j,k2}, C_{i1,j,k1} \in \Sigma \wedge \mathrm{i2} = \mathrm{k1} \wedge \mathrm{i2} = \mathrm{i1} + 1 \wedge \mathrm{j} \neq \mathrm{i2}\}$$

$$G_5 = \{(C_{i1,j,k}, C_{i2,j,k}) | C_{i1,j,k}, C_{i2,j,k} \in \Sigma \wedge \mathrm{i2} = \mathrm{i1} + 1 \wedge \mathrm{j} \neq \mathrm{i2}\}$$

$$G = G_1 \cup G_2 \cup G_3 \cup G_4 \cup G_5$$

Graf (z zaznaczonymi klasami Foaty) dla macierzy 4x4 prezentuje się następująco:



1.6 Klasy Foaty

Przyjmując i = 1, 2, 3, …, n-1 klasy Foaty mają następującą postać:

$$F_{Ai} = \{A_{i,k} | k = i + 1, i + 2, \dots, n\}$$

$$F_{Bi} = \{B_{i,j,k} | k = i + 1, i + 2, \dots, n \ \wedge j = i, i + 1, \dots, n + 1\}$$

$$F_{Ci} = \{C_{i,j,k} | k = i + 1, i + 2, \dots, n \ \wedge j = i, i + 1, \dots, n + 1\}$$

Postać normalna Foaty prezentuje się zatem następująco:

$$[F_{A1}][F_{B1}][F_{c1}][F_{A2}][F_{B2}][F_{C2}] \dots [F_{An-1}][F_{Bn-1}][F_{Cn-1}]$$

2. Kilka słów o implementacji:

Programy zostały napisane w języku Java i wymagają do uruchomienia JDK w wersji co najmniej 11. Do wygenerowania grafu w formacie .png wymagany jest zestaw narzędzi „Graphviz”.

W katalogu z zadaniem znajdują się trzy katalogi:

- Diekert – zawiera program do wygenerowania grafu Diekerta w formacie „.dot” (wygenerowany plik znajdzie się w katalogu Diekert pod nazwą „graph.dot”). Domyślnie przyjmuje rozmiar macierzy jako 4 (4x4), ale można podać rozmiar jako argument programu. Można wygenerować graf w formacie „.png” za pomocą komendy (będąc w katalogu, gdzie znajduje się plik „graph.dot”):

  ```
  $ dot graph.dot -Tpng > nazwa_pliku.png
  ```

  Gdzie nazwa_pliku to nazwa wygenerowanego pliku w formacie „.png”
- Gauss_elimination – zawiera implementację współbieżnego algorytmu eliminacji Gaussa. Domyślnie jako nazwę pliku wejściowego przyjmuje „in.txt”, ale można podać nazwę pliku jako argument programu. Wynik zostaje zapisany w pliku „out.txt” w głównym katalogu projektu.

Oba powyższe programy są projektami „maven”.

- Tester – zawiera program do sprawdzania poprawności zaimplementowanej eliminacji Gaussa, udostępniony przez prowadzącego. Znajduje się pod linkiem:

  https://github.com/macwozni/Matrices