Akademia Górniczo-Hutnicza
Wydział Informatyki, Elektroniki i Telekomunikacji
Kierunek Informatyka

Teoria Współbieżności

Sprawozdanie

Współbieżna eliminacja Gaussa

Jan Nowakowski

Kraków, 7 grudnia 2020

Spis treści

1 Wprowadzenie oraz zadania do wykoniania.

Teoria śladów służy głównie do modelowania zachowania systemów współbieżnych. Opisuje zależności pomiędzy elementami predefiniowanego zbioru niepodzielnych zadań obliczeniowych. Podstawowym pojęciem teorii śladów jest alfabet będący skończonym zbiorem niepodzielnych zadań obliczeniowych (tasków) - to znaczy niepodzielnych i składających się jedynie z instrukcji wykonywanych sekwencyjnie. Teoria śladów jest szeroko stosowanym formalizmem pozwalającym wyprowadzić optymalne zrównoleglenie dla zadanej grupy tasków.

1.1 Zadania do wykonania.

W zadaniu należy rozważyć dowolną macierz kwadratową o długości boku N, reprezentującą współczynnki układu równań liniowych, oraz wektor długości N, będący wektorem rozwiązań.

1. Zdefiniować podstawowe, niepodzielne zadania obliczeniowe.
2. Zidentyfikować ciąg zadań obliczeniowych wykonywanych przez sekwencyjny algorytm.
3. Zidentyfikować alfabet w sensie teorii śladów.
4. Określić relację zależności.
5. Wyprowadzić graf Dickerta.
6. Przedstawić postać normalną Foaty.
7. Zaimplementować algorytm współbieżnej eliminacji Gaussa w oparciu o postać normalną Foaty.

2 Podstawowe zadania obliczeniowe.

Problem elinacji Gaussa polega na znalezieniu wektora X spełniającego równianie:

$$N * X = Y$$

Algortym zakłada, że do macierzy M ma postać:

$$M = N|Y$$

gdzie operacja | oznacza dopisanie wektora Y jako nową kolumnę w macierzy N. Oznacza to, że macierz M ma N+1 kolumn.

W problemie współbieżnej eliminacji Gaussa można wyróżnić trzy podstawowe, niepodzielne zadania obliczeniowe. Są to odpowiednio:

- $A_{i,k}$ - znalezienie mnożnika dla wiersza i do odjęcia go od wiersza k ($m_{i,k} = M_{k,i}/M_{i,i}$)
- $B_{i,j,k}$ - pomnożenie elementu $M_{i,j}$ przez odpowiedni mnożnik, aby można było wykonać odejmowanie ($n_{i,j,k} = M_{i,j} * m_{i,k}$)
- $C_{i,j,k}$ - odjęcie j-tego elementu w wierszu i przemnożonego przez mnożnik od j-tego elementu w wierszy k ($M_{k,j} = M_{k,j} - n_{i,j,k}$)

Tak zdefiniowane zadania reprezentują pojedyncze operacje arytmetyczne, które należy wykonać w procesie eliminacji Gaussa.

3 Ciąg zadań wykonywany przez algorytm sekwencyjny i alfabet teorii śladów.

W algorytmie sekwencyjnym przetwarzane są całe rzędy na raz, to znaczy, że najpierw jest wyznaczany mnożnik, następnie dla całego rzędu wykonywane są mnożenia poszczególnych elementów i odejmowanie. Ciąg zadań wykonywany przez algorytm dla dowolnego rozmiaru macierzy N można przedstawić następująco:

$$A_{i,i+1}, B_{i,i,i+1}, C_{i,i,i+1}, B_{i,i+1,i+1}, C_{i,i+1,i+1}, B_{i,i+2,i+1}, C_{i,i+2,i+1}, ..., B_{i,N+1,i+1}, C_{i,N+1,i+1},$$
$$A_{i,i+2}, B_{i,i,i+2}, C_{i,i,i+2}, B_{i,i+1,i+2}, C_{i,i+1,i+2}, B_{i,i+2,i+2}, C_{i,i+2,i+2}, ..., B_{i,N+1,i+2}, C_{i,N+1,i+2}, ...,$$
$$A_{i,N}, B_{i,i,N}, C_{i,i,N}, B_{i,i+1,N}, C_{i,i+1,N}, B_{i,i+2,N}, C_{i,i+2,N}, ..., B_{i,N+1,N}, C_{i,N+1,N}$$

dla i zmieniającego się od 1 do N-1 (iteracja po rzędach).

Można to przedstawić za pomocą psedokodu jak na listingu 1.

```
for i <- 1 to N-1:
    for k <- i+1 to N:
        A_ik
        for j <- i to N+1:
            B_ijk
            C_ijk
```

Listing 1: Pseudokod sekwencyjnego algorytmu.

Sekwencja ta jest również śladem, dla którego należy znaleźć optymalne zrównoleglenie.

W alfabecie w sensie teorii śladów występują wszystkie zadania, które wykonał by algorytm sekwencyjny.

$$\sum = (\bigcup_{i=1}^{N-1} \bigcup_{k=i+1}^{N} \{A_{i,k}\}) \bigcup (\bigcup_{i=1}^{N-1} \bigcup_{k=i+1}^{N} \bigcup_{j=i}^{N+1} \{B_{i,j,k}, C_{i,j,k}\})$$

4 Relacja zależności.

W przypadku tego problemu występuje 5 rodzajów zależności między zadaniami.

Zadania B w oczywisty sposób zależą od zadań A, a zadania C do zadań B, ponieważ najpierw należy wyznaczyć mnożnik, następnie dokonać mnożenia i dopiero wtedy można wykonać odejmowane. Te zależności towrzą zbiór:

$$D_1 = \bigcup_{i=1}^{N-1} \bigcup_{k=i+1}^{N} \bigcup_{j=i}^{N+1} \{(A_{i,k}, B_{i,j,k}), (B_{i,j,k}, C_{i,j,k})\}$$

Kolejnym rodzajem zależności jest zleżność zadań A od zadań C. Aby wyznaczyć mnożnik dla dwóch wierszy należy najpierw wykonać odejmowanie na elementach w tych wierszach w kolumnie od przekątnej w dół. Dla przykładu, aby wyznaczyć $A_{2,4}$ należy zakończyć zadanie $C_{1,2,2}$ oraz $C_{1,2,4}$. Ten rodzaj zależności generuje dwa zbiory:

$$D_2 = \bigcup_{i=1}^{N-2} \bigcup_{k=i+2}^{N} \{(C_{i,i+1,i+1}, A_{i+1,k})\}$$

$$D_3 = \bigcup_{i=1}^{N-2} \bigcup_{k=i+2}^{N} \{(C_{i,i+1,k}, A_{i+1,k})\}$$

D_2 pochodzi od zadania C dla elementu z przekątnej, natomiast D_3 pochodzi od zadań C dla elementów poniżej przekątnej.

Część zadań B zależy od zadań C, ponieważ aby wykonać mnożenie danego elementu, należy najpierw od niego odjąć poprzedni element. Generuje to zbiór:

$$D_4 = \bigcup_{i=1}^{N-2} \bigcup_{j=i+2}^{N+1} \bigcup_{k=i+2}^{N} \{(C_{i,j,i+1}, B_{i+1,j,k})\}$$

Również część zadań C zależy od innych zadań typu C, ponieważ aby wykonać odejmowanie należy skończyć poprzednie odejmowania. To generuje zbiór:

$$D_5 = \bigcup_{i=1}^{N-2} \bigcup_{j=i+2}^{N+1} \bigcup_{k=i+2}^{N} \{(C_{i,j,k}, C_{i+1,j,k})\}$$

Zbiory D_1-D_5 zdefiniowano tak, aby nie uwzględniały przechodniości ani symetrii. Ostatecznie zbiór relacji zależności ma postać:

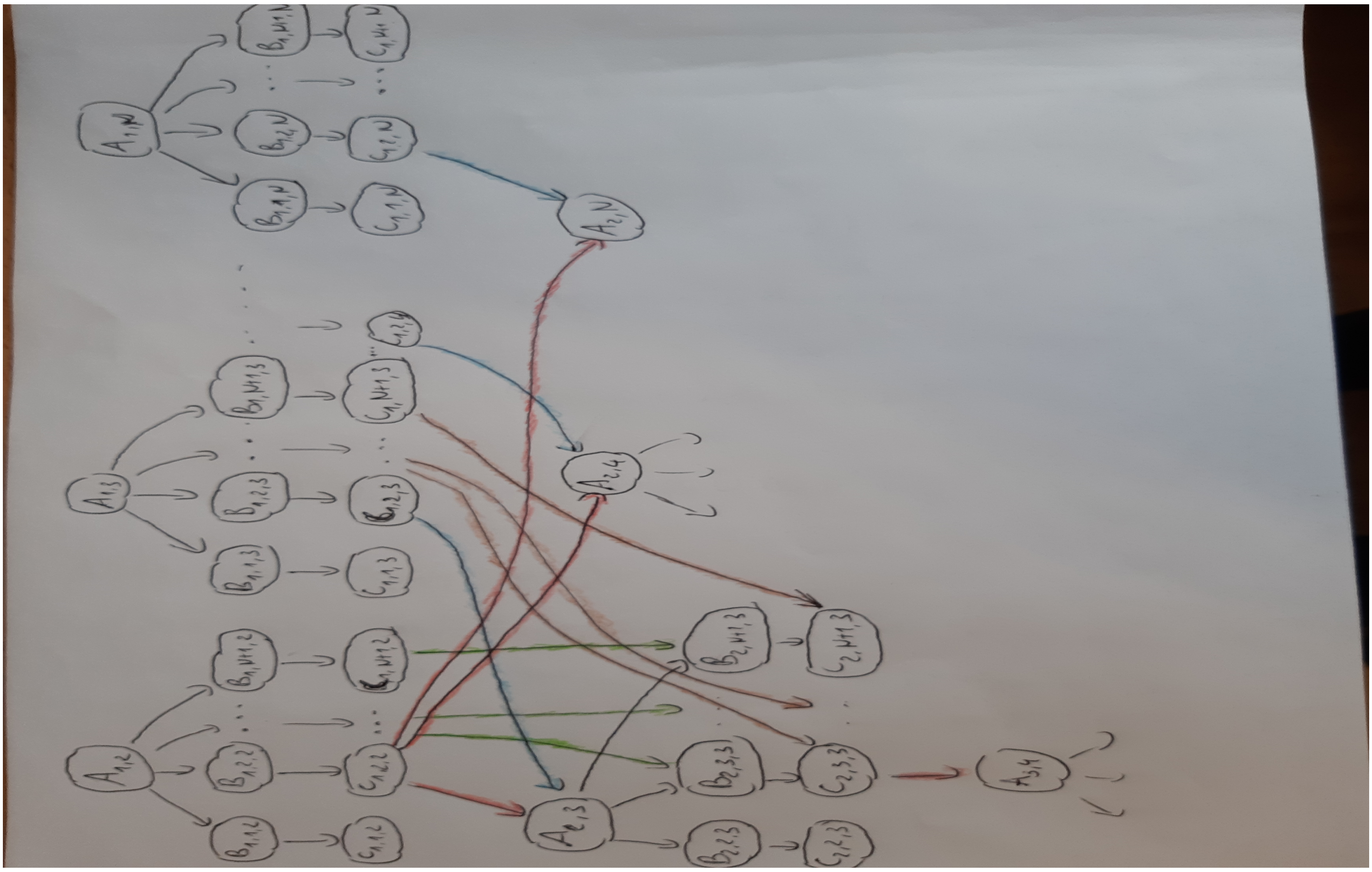
$$D = sym\{(D_1 \bigcup D_2 \bigcup D_3 \bigcup D_4 \bigcup D_5)^+\} \bigcup I_{\sum}$$

5 Graf Dickerta.

Na podstawie kolejności w której zadania wykonuje algorytm sekwencyjny, będącej śladem oraz relacji zależności można wygenerować graf Dickerta. Można zauważyć, że zbiorem wierzchołków będą wszystkie elementu alfabetu w sensie teorii śladów. Są to wszystkie zadania, które wykonałby element sekwencyjny.

Rysunek 1 przedstawia poglądowy szkic grafu Dickerta. Można zauważyć, że zdefiniowane w poprzednim punkcie zbiory D1-D5 wyznaczają zbiór krawędzi w grafie zależności. Na rysunku zaznaczono kolorami 5 rodzajów krawędzi. Kolory krawędzi w grafie na rysunku odpowiadają zbiorą D1-D5, tak że:

- D1 -> kolor czarny
- D2 -> kolor czerwony
- D3 -> kolor niebieski
- D4 -> kolor zielony
- D5 -> kolor brązowy



Rysunek 1: Początek szkicu grafu Dickerat

Zatem graf Dickerta można opisać jako:

$$G = (\sum, D_1 \bigcup D_2 \bigcup D_3 \bigcup D_4 \bigcup D_5)$$

gdzie pierwszy element krotki to zbiór wierzchołków, a drugi element to zbiór krawędzi.

6 Postać normalna Foaty.

Na podstawie grafu Dickerta, relacji zależności oraz śladu można wyznaczyć klasy Foaty. Można zauważyć, że klasy będą zawierały zadania jednego typu. Dla danego i będą występowały 3 klasy Foaty, najpierw ta zawierająca zadania typu A, następnie ta zawierająca wszystkie zadania B i na koniec C. Takie trójki klas Foaty będą występować w ciągu dla i rosnącego od 1 do N-1. Otrzymany wyniki można zaprezetować w ten sposób:

$$[A_{1,2}, A_{1,3}, A_{1,4}, ..., A_{1,N}]$$

$$[B_{1,1,2}, B_{1,2,2}, B_{1,3,2}, ..., B_{1,N+1,2}, B_{1,1,3}, B_{1,2,3}, B_{1,3,3}, ..., B_{1,N+1,3}$$

$$..., B_{1,1,N}, B_{1,2,N}, B_{1,3,N}, ..., B_{1,N+1,N}]$$

$$[C_{1,1,2}, C_{1,2,2}, C_{1,3,2}, ..., C_{1,N+1,2}, C_{1,1,3}, C_{1,2,3}, C_{1,3,3}, ..., C_{1,N+1,3}$$

$$..., C_{1,1,N}, C_{1,2,N}, C_{1,3,N}, ..., C_{1,N+1,N}]$$

$$[A_{2,3}, A_{2,4}, A_{2,5}, ..., A_{2,N}]$$

$$[B_{2,2,3}, B_{2,3,3}, B_{2,4,3}, ..., B_{2,N+1,3}, B_{2,2,4}, B_{2,3,4}, B_{2,4,4}, ..., B_{2,N+1,4}$$

$$..., B_{2,2,N}, B_{2,3,N}, B_{2,4,N}, ..., B_{2,N+1,N}]$$

$$[C_{2,2,3}, C_{2,3,3}, C_{2,4,3}, ..., C_{2,N+1,3}, C_{2,2,4}, C_{2,3,4}, C_{2,4,5}, ..., C_{2,N+1,4}$$

$$..., C_{2,2,N}, C_{2,3,N}, C_{2,4,N}, ..., C_{2,N+1,N}]$$

$$[A_{3,4}, ...][B_{3,3,4}, ...][C_{3,3,4}, ...]$$

$$...$$

$$[A_{N-1,N}][B_{N-1,N-1,N}, B_{N-1,N,N}, B_{N-1,N+1,N}][C_{N-1,N-1,N}, C_{N-1,N,N}, C_{N-1,N+1,N}]$$

7 Wnioski.

Zastosowanie teorii śladów pozwala uzyskać optymalne zrównoleglenie problemu eliminacji Gaussa. Dzięki zdefiniowaniu niepodzielnych zadań obliczeniowych współbieżnie mogą się wykonywać nie tylko zadania dotyczące wierszy macierzy, lecz pojedynczych elementów.