Współbieżna eliminacja Gaussa-Jordana – Adrian Glanowski

Dla dowolnej rozszerzonej macierzy M o wymiarach $n \times (n+1)$, powstałej z doklejenia wektora b (dla ułatwienia $M_{i,n+1} = b_i$) do macierzy kwadratowej A,

$$M = \left[\begin{array}{cccc|c} A_{11} & A_{12} & \cdots & A_{1n} & b_1 \\ A_{21} & A_{22} & \cdots & A_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} & b_n \end{array}\right]$$

wprowadzamy niepodzielne operacje prowadzące do przekształcenia A w macierz jednostkową.

Alfabet operacji:

- A_{ij} – obliczenie współczynnika x_{ij}, przez który należy pomnożyć j-ty wiersz, $x_{ij} = \dfrac{M_{ii}}{M_{ji}}$
- B_{ikj} – pomnożenie elementu M_{jk} przez wyliczony wcześniej współczynnik x_{ij}, $M_{jk} *= x_{ij}$
- C_{ikj} – odjęcie k-tego elementu i-tego wiersza od k-tego elementu j-tego wiersza, $M_{jk} -= M_{ik}$
- D_i – obliczenie współczynnika y_i normalizacji i-tego wiersza $y_i = \dfrac{1}{M_{ii}}$
- E_{ik} – pomnożenie elementu M_{ik} przez wyliczony wcześniej współczynnik y_i

Uwaga: wiersz i-ty może dalej być nazywany wierszem źródłowym, a j-ty wierszem docelowym (bo jego element docelowo zerujemy) w kontekście operacji A, B, C.

Eliminacja przy użyciu wporwadzonego alfabetu:

Żeby doprowadzić A do macierzy jednostkowej, należy w każdej kolumnie wyzerować wysztkie elementy poza leżącymi na przekątnej. Zatem przyjmując i jako wiersz źródłowy, a j $(i \neq j)$ jako wiersz docelowy, wykonujemy:

- A_{ij}
- B_{iij} (w celu wyzerowania wspomnianego elementu, ale z oczywistych powodów mnożenie należy zastosować dla całego wiersza)
- C_{iij} (jw.)
- D_i
- E_{ii} (w celu uzyskania 1 na przekątnej, ale z oczywistych powodów mnożenie należy zastosować dla całego wiersza)

Wykonujemy te operacje, aż każdy wiersz będzie wierszem źródłowym (normalizacja występuje raz dla każdego wiersza)

Ostatecznie dla wiersza źrodłowego i otrzymujemy:

$$\Sigma_i = \{A_{i1}, B_{i11}, B_{i21}, \dots, B_{i(n+1)1}, C_{i11}, C_{i21}, \dots, C_{i(n+1)1},$$

$$\dots$$

$$A_{i(i-1)}, B_{i1(i-1)}, B_{i2(i-1)}, \dots, B_{i(n+1)(i-1)}, C_{i1(i-1)}, C_{i2(i-1)}, \dots, C_{i(n+1)(i-1)},$$

$$A_{i(i+1)}, B_{i1(i+1)}, B_{i2(i+1)}, \dots, B_{i(n+1)(i+1)}, C_{i1(i+1)}, C_{i2(i+1)}, \dots, C_{i(n+1)(i+1)},$$

$$\dots,$$

$$A_{in}, B_{i1n}, B_{i2n}, \dots, B_{i(n+1)n}, C_{i1n}, C_{i2n}, \dots, C_{i(n+1)n},$$

$$D_i, E_{i1}, E_{i2}, \dots, E_{i(n+1)}\}$$

Cała eliminacja natomiast to:

$$\Sigma = \Sigma_1 \cup \Sigma_2 \cup \dots \cup \Sigma_n$$

Relacja zależności:

Żeby stworzyć relację zależności należy rozważyć powiązania między operacjami w obrębie eliminacji jednego wiersza oraz pomiędzy wierszami:

- Zależności w obrębie wiersza:
  - Operacje A, B, C w obrębie jednego elementu są bezpośrednio powiązane, więc zależne od siebie.
    $Dep_1 = \{(A_{ij}, B_{ikj}) \mid k \in \{1, \dots, n+1\} \wedge i, j \in \{1, \dots, n\} \wedge i \neq j\}$
    $Dep_2 = \{(B_{ikj}, C_{ikj}) \mid k \in \{1, \dots, n+1\} \wedge i, j \in \{1, \dots, n\} \wedge i \neq j\}$
  - Wiele powyższych operacji jest implementacyjnie redundantnych (mnożenie i odejmowanie wyzerowanych elementów), więc można je pominąć – w tradycyjnej, eliminacji Gaussa są to elementy, których kolumna jest mniejsza niż aktualnie rozważany wiersz, w wariancie Jordana są to elementy, które nie znajdujące się na przękątnej, których kolumna jest mniejsza niż aktualnie rozważany wiersz.

Obrazek pomocniczy, na kolorowo oznaczono istotne elementy (i=3):

$$M = \left[\begin{array}{ccccc|c} A_{11} & 0 & A_{13} & \cdots & A_{1n} & b_1 \\ 0 & A_{22} & A_{23} & \cdots & A_{2n} & b_2 \\ 0 & 0 & A_{33} & \cdots & A_{3n} & b_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & A_{n3} & \cdots & A_{nn} & b_n \end{array}\right]$$

Ostatecznie otrzymujemy zatem:

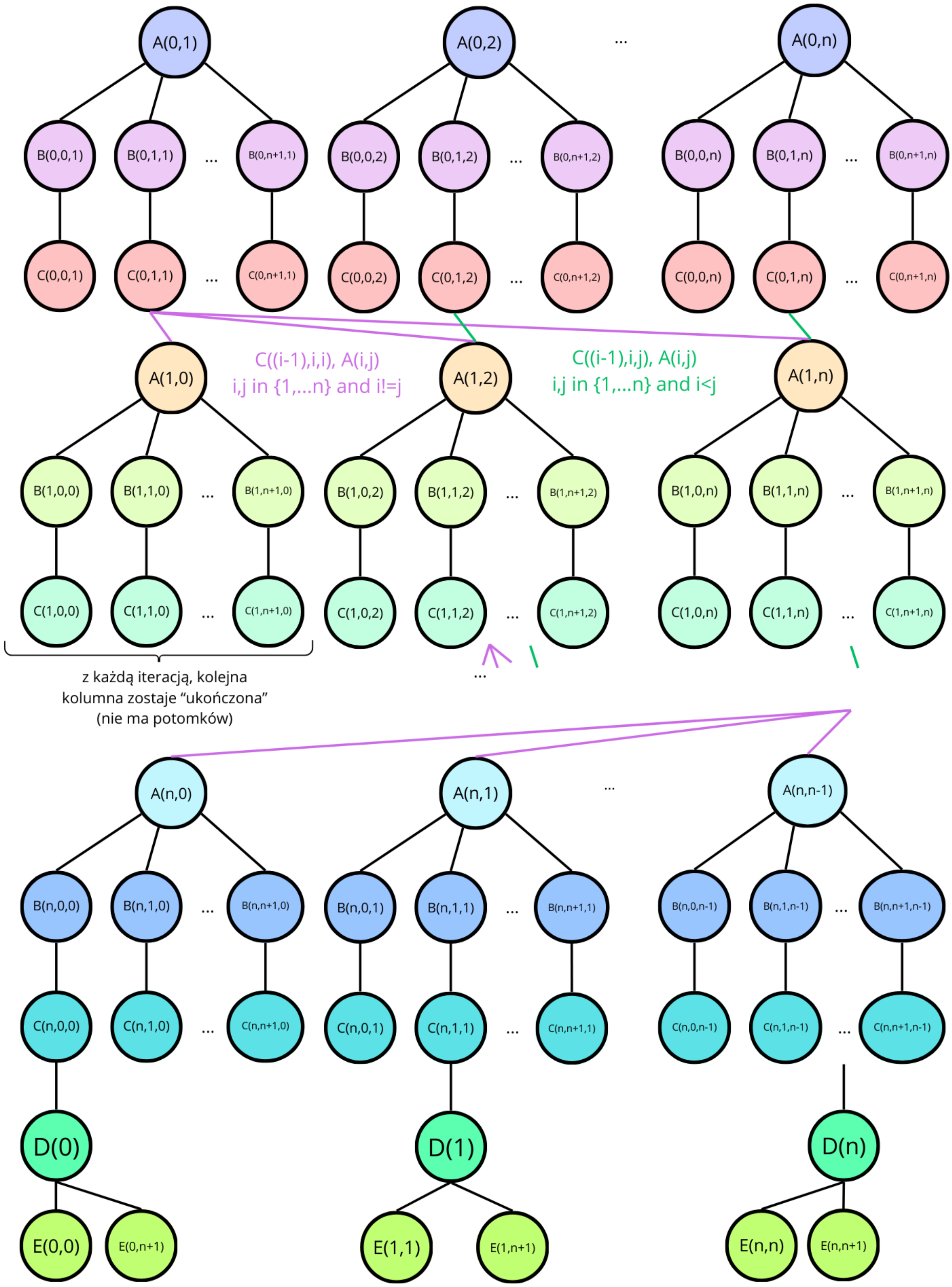
$$Dep_1 = \{(A_{ij}, B_{ikj}) \mid k \in \{i, \dots, n+1\} \wedge i,j \in \{1, \dots, n+1\} \wedge i < j\} \cup \{(A_{ij}, B_{ikj}) \mid (k \in \{i, \dots, n+1\} \vee k = j) \wedge i,j \in \{1, \dots, n\} \wedge i > j\}$$

$$Dep_2 = \{(B_{ikj}, C_{ikj}) \mid k \in \{i, \dots, n+1\} \wedge i,j \in \{1, \dots, n+1\}\}$$

W przypadku Dep_2 (zastosowania operacji C) pomijamy przekątną, gdyż odjęcie 0 od elementu na przekątnej jest zbędne

- Zależności między wierszami:
  - Część Gaussa – nie można wykonać operacji A_{ij}, jeśli nie została wykonana eliminacje, w których i oraz j były wierszem docelowym, a $i-1$ wierszem źródłowym (wystarczy, że eliminacja została policzona dla elementu na przekątnej, gdyż on jest wykorzystywany w operacji A)
    $$Dep_3 = \{(C_{(i-1)ii}, A_{ij}), (C_{(i-1)ij}, A_{ij}) \mid i,j \in \{1, \dots, n\} \wedge i < j\}$$
  - Część Jordana – nie można wykonać operacji A_{ij}, jeśli nie została wykonana eliminacja, w której i był wierszem docelowym, a $i-1$ wierszem źródłowym (wystarczy, że eliminacja została policzona dla elementu na przekątnej, gdyż on jest wykorzystywany w operacji A)
    $$Dep_4 = \{(C_{(i-1)ii}, A_{ij}) \mid i,j \in \{1, \dots, n\} \wedge i > j\}$$
- Normalizacja wierszy:
  - Operacje D oraz E są bezpośrednio powiązane w obrębie jednego wiersza, więc zależne od siebie.
    $$Dep_5 = \{(D_i, E_{ij}) \mid i,j \in \{1, \dots, n\}\}$$
  - Wiele powyższych operacji jest implementacyjnie redundantnych (normalizowanie elementów, które ostatecznie będą zerami, czyli w zasadzie wszystkie, poza przekątną i elementami doklejonego wektora), więc można je pominąć (przy założeniu, że <u>normaliacja zostanie wykonana na samym końcu</u>).
    Ostatecznie otrzymujemy zatem:
    $$Dep_5 = \{(D_i, E_{ii}) \mid i \in \{1, \dots, n\}\} \cup \{(D_i, E_{i(n+1)}) \mid i \in \{1, \dots, n\}\}$$
    Aby zapewnić, że noramlizacja wykona się jako ostatnie działanie dla danego wiersza, dodajemy
    $$Dep_6 = \{(C_{nii}, D_i) \mid i \in \{1, \dots, n\}\} \cup \{(C_{n(n+1)i}, D_i) \mid i \in \{1, \dots, n\}\}$$
    Ostatecznie ze wszystkich powyższych warunków otrzymujemy:
    $$D = sym\{\{Dep_1 \cup Dep_2 \cup Dep_3 \cup Dep_4 \cup Dep_5 \cup Dep_6\}^+\} \cup I_\Sigma$$

Graf Diekerta:



UWAGA: Dla uproszczenia grafu pominięto optymalizację B oraz C

Postać normalna Foaty:

Jak wspomniano wyżej, eliminacja Gaussa-Jordana sprowadza się do ciągu operacji Σ_i dla każdego wiersza źródłowego. Ostatecznie więc otrzymany ciąg operacji to:

$$t = [\langle \Sigma_1, \Sigma_2, \dots, \Sigma_n \rangle]$$

Przekształcenie ciągu operacji t do postaci normalnej foaty wykorzystując wcześniej przedstawione reguły Dep_i okazuje się trywialne.

$$FNF =$$

$$[\{A_{12}, A_{13}, \dots, A_{1n}\}]_{\equiv_I^+}{}^\frown$$

$$\left[\left\{B_{112}, B_{122}, \dots, B_{1(n+1)2}, B_{113}, B_{123}, \dots, B_{1(n+1)3}, \dots, B_{11n}, B_{12n}, \dots, B_{1(n+1)n}\right\}\right]_{\equiv_I^+}{}^\frown$$

$$\left[\left\{C_{112}, C_{122}, \dots, C_{1(n+1)2}, C_{113}, C_{123}, \dots, C_{1(n+1)3}, \dots, C_{11n}, C_{12n}, \dots, C_{1(n+1)n}\right\}\right]_{\equiv_I^+}{}^\frown$$

$$[\{A_{21}, A_{23}, \dots, A_{2n}\}]_{\equiv_I^+}{}^\frown$$

$$\left[\left\{B_{211}, \dots, B_{2(n+1)2}, B_{223}, \dots, B_{2(n+1)3}, \dots, B_{22n}, \dots, B_{2(n+1)n}\right\}\right]_{\equiv_I^+}{}^\frown$$

$$\left[\left\{C_{211}, \dots, C_{2(n+1)2}, C_{223}, \dots, C_{2(n+1)3}, \dots, C_{22n}, \dots, C_{2(n+1)n}\right\}\right]_{\equiv_I^+}{}^\frown$$

$$\dots{}^\frown$$

$$\left[\left\{A_{n1}, A_{n2}, \dots, A_{n(n-1)}\right\}\right]_{\equiv_I^+}{}^\frown$$

$$\left[\left\{B_{n11}, B_{nn1}, B_{n(n+1)1}, B_{n22}, B_{nn2}, B_{n(n+1)2}, \dots, B_{n(n-1)(n-1)}, B_{nn(n-1)}, B_{n(n+1)(n-1)}\right\}\right]_{\equiv_I^+}{}^\frown$$

$$\left[\left\{C_{n11}, C_{nn1}, C_{n(n+1)1}, C_{n22}, C_{nn2}, C_{n(n+1)2}, \dots, C_{n(n-1)(n-1)}, C_{nn(n-1)}, C_{n(n+1)(n-1)}\right\}\right]_{\equiv_I^+}{}^\frown$$

$$[\{D_1, D_2, \dots, D_n\}]_{\equiv_I^+}{}^\frown$$

$$\left[\left\{E_{11}, E_{1(n+1)}, E_{22}, E_{2(n+1)}, \dots, E_{nn}, E_{n(n+1)}\right\}\right]_{\equiv_I^+}$$