Raport - Zadanie numeryczne 3

Grzegorz Janysek

22 listopada 2021

1 Wstęp teoretyczny

1.1

W ogólnym przypadku złożoność numerycznego rozwiązywania układów równań z liniowych to $O(n^3)$. Daje się ją jednak znacząco zmniejszyć stosując algorytmy wykożystujące strukturę macierzy. Zadanie skupia się na metodzie dla szczególego rodzaju kwadratowej macjeży żadkiej: macieży wstęgowej (in. pasmowej). Charakteryzuje się ona tym, że poza główną diagonalą i wstęgą wokół niej, wszystkie elementy są zerowe.

$$\begin{bmatrix}
\bullet & \circ & \circ & & & & & \\
\circ & \bullet & \circ & \circ & & & & \\
 & \circ & \bullet & \circ & \circ & & & \\
 & & \ddots & \ddots & \ddots & \ddots & & \\
 & & & \circ & \bullet & \circ & \circ & \\
 & & & & \circ & \bullet & \circ & \circ \\
 & & & & & \circ & \bullet & \circ \\
 & & & & & & \circ & \bullet
\end{bmatrix} \tag{1}$$

1.2

Przechowywanie macieży w pamięci w ogólnym przypadku zajmuje $x * n^2$, gdzie x to rozmiar typu danych. Do zapisania macierzy dla typu danych *float64* konieczne by było:

$$\text{dla } n = 100: \qquad 8\text{B} * 100^2 = 80\text{kB} \tag{2}$$

$$\text{dla } n = 1{,}000: \qquad 8\text{B} * 1{,}000^2 = 8\text{MB} \tag{3}$$

$$\text{dla } n = 10{,}000: \qquad 8\text{B} * 10{,}000^2 = 800\text{MB} \tag{4}$$

Dla ogólnej macierzy wstęgowej większość elementów to zera, na znanych pozycjach poza wstęgą. Pozwala to na optymalizację polegającą na przechowywaniu jedynie wstęgi w postaci np. tablicy dwuwymiarowej $a \times n$, której rozmiar to $x * a * n$, gdzie x to rozmiar typu danych, a to szerokość wstęgi. Na zapis w ten sposób potrzeba znacząco mniej pamięci. Zakładając szerokość wstęgi $a = 4$ (jak dla macierzy w zadaniu) mamy:

$$\text{dla } n = 100: \qquad 8\text{B} * 4 * 100 = 3.2\text{kB} \tag{5}$$

$$\text{dla } n = 1{,}000: \qquad 8\text{B} * 4 * 1{,}000 = 32\text{kB} \tag{6}$$

$$\text{dla } n = 10{,}000: \qquad 8\text{B} * 4 * 10{,}000 = 320\text{kB} \tag{7}$$

1.3

Faktoryzacja LU zachowuje strukturę macieży pasmowej. Wiedząc to można pominąć obliczanie elementów macieży L oraz U poza wstęgą, ponieważ z powyższego wynika, że będą one zerowe. Ogranicza to konieczność obliczeń tylko dla elemetów wstęgi, których ilosć jest rzędu n. Dodatkowo ilość elementów sumy koniecznych do obliczania wartości elementu w L lub w U pozostaje nie większa niż szerokość wstęgi. Wykożystując to można zmniejszyć złożoność rozkładu dla macieży wstęgowej do $O(n)$.

Otrzymujemy przekształcenia (8, 9) jawnych wzorów na wartości elementów L i U, gdzie W to zbiór elementów należących do wstęgi

$$u_{ij} = a_{ij} - \sum_{k \in P} l_{ik} u_{kj} \qquad P = \{k : k < i \wedge l_{ik} \in W \wedge u_{kj} \in W\} \tag{8}$$

$$l_{ij} = \frac{1}{u_{jj}} \left(a_{ij} - \sum_{k \in Q} l_{ij} u_{kj} \right) \qquad Q = \{k : k < j \wedge l_{ik} \in W \wedge u_{kj} \in W\} \tag{9}$$

W połączeniu z *backward-substitution* i *forward-substitution* których złożoność dla macierzy wstęgowej wynosi $O(n)$, otrzymujemy całkowitą złożoność rozwiązywania układów równań liniowych z macieżą wstęgową wynoszącą $O(n)$

1.4

W ćwiczeniu należy znaleźć $y = A^{-1}x$ gdzie

$$A = \begin{bmatrix} 1.2 & \frac{0.1}{1} & \frac{0.4}{1^2} & & & & \\ 0.2 & 1.2 & \frac{0.1}{2} & \frac{0.4}{2^2} & & & \\ & 0.2 & 1.2 & \frac{0.1}{3} & \frac{0.4}{3^2} & & \\ & & \ddots & \ddots & \ddots & \ddots & \\ & & & 0.2 & 1.2 & \frac{0.1}{N-2} & \frac{0.4}{(N-2)^2} \\ & & & & 0.2 & 1.2 & \frac{0.1}{N-1} \\ & & & & & 0.2 & 1.2 \end{bmatrix} \qquad x = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ \vdots \\ N \end{bmatrix} \tag{10}$$

2 Wyniki

$$det(A) = 78240161.00959387 \tag{11}$$

$$y = \begin{bmatrix} 0.03287133486041399 \\ 1.339622798096375 \\ 2.066480295894664 \\ 2.825543605175336 \\ 3.557571715528883 \\ 4.284492868897645 \\ 5.00721018451999 \\ 5.727664002754518 \\ \vdots \\ 67.19108363147355 \\ 67.9053991828134 \\ 68.61971410401004 \\ 69.33402833257784 \\ 70.0483379441879 \\ 70.7650588638003 \\ 71.53915685603329 \end{bmatrix} \tag{12}$$

3 Podsumowanie

Podczas rozwiązywania układów równań liniowych kluczowa jest znajomość struktury macieży. Pozwala to na dobranie odpowiedniego sposobu przechowywania w pamięci, oraz algorytmu faktoryzacji, potencjalnie dramatycznie zmniejszając zużycie pamięci oraz czas procesora.