Metody Numeryczne Projekt 2
"Układy równań liniowych"
Sprawozdanie

Grzegorz Choiński, 165105

09.05.2018

1 Wstęp

Celem projektu było zaimplementowanie w języku C++ trzech metod rozwiązywania układów równań liniowych, dwóch iteracyjnych (Jacobiego i Gaussa-Seidla) oraz bezpośredniej (faktoryzacja LU). Układ równań liniowych wykorzystywany w obliczeniach ma następującą postać:

$$\mathbf{Ax} = \mathbf{b}$$

gdzie $\mathbf{A}$ jest daną macierzą pasmową, $\mathbf{b}$ danym wektorem wyrazów wolnych, a $\mathbf{x}$ jest wektorem rozwiązań.

2 Zadanie A

Celem tego zadania było wypełnienie macierzy $\mathbf{A}$ o rozmarze NxN, dla N = 905 oraz wektora wyrazów wolnych $\mathbf{b}$ o długości N, którego n-ty element ma wartość sin(n ·6), a nastepnie stworzenie z nich układu równań.

$$\begin{bmatrix} 6 & -1 & -1 & 0 & 0 & 0 & 0 & \dots & 0 \\ -1 & 6 & -1 & -1 & 0 & 0 & 0 & \dots & 0 \\ -1 & -1 & 6 & -1 & -1 & 0 & 0 & \dots & 0 \\ 0 & -1 & -1 & 6 & -1 & -1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & -1 & -1 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \\ x_{905} \end{bmatrix} = \begin{bmatrix} sin(6) \\ sin(12) \\ sin(18) \\ sin(24) \\ \vdots \\ sin(5430) \end{bmatrix} \quad \textbf{(1)}$$

Równanie liniowe postaci $\mathbf{Ax} = \mathbf{b}$

3 Zadanie B

Celem tego zadania było zaimplementowanie metod iteracyjnych - Jacobiego oraz Gaussa-Seidla oraz za pomocą ich rozwiązać równanie **(1)**, a następnie porównać czasy oraz ilość iteracji, po których norma z residuum jest mniejsza od $\epsilon = 10^{-9}$.

Metoda	Liczba Iteracji	Czas [s]
Jacobi	62	0.466
Gauss-Seidl	36	0.201

Tabela 1. Porównanie czasów poszczególnych metod iteracyjnych.

4 Zadanie C

W tym zadaniu należało stworzyć nową macierz A_2 i rozwiązać metodami iteracyjnymi nowe równanie **(2)**.

$$\begin{bmatrix} 3 & -1 & -1 & 0 & 0 & 0 & 0 & \dots & 0 \\ -1 & 3 & -1 & -1 & 0 & 0 & 0 & \dots & 0 \\ -1 & -1 & 3 & -1 & -1 & 0 & 0 & \dots & 0 \\ 0 & -1 & -1 & 3 & -1 & -1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & -1 & -1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \\ x_{905} \end{bmatrix} = \begin{bmatrix} sin(6) \\ sin(12) \\ sin(18) \\ sin(24) \\ \vdots \\ sin(5430) \end{bmatrix} \quad \textbf{(2)}$$

Równanie $A_2\mathbf{x} = \mathbf{b}$

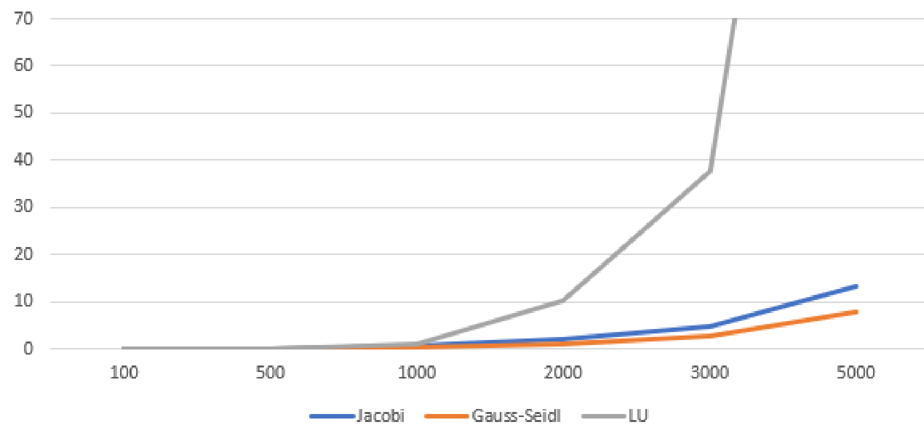
Okazuje się, że podczas próby wykorzystania obu metod, norma z residuum zaczyna wzrastać do nieskończoności zamiast maleć. Następstwem jest to, że algorytmy nigdy się nie zatrzymają, przez co ani metoda Jacobiego, ani Gaussa-Seidla nigdy się nie zbiegnie.

5 Zadanie D

W tym zadaniu należało rozwiązać równianie **(2)** za pomocą bezpośredniej metody LU. Norma z residuum z tak rozwiązanego równania wynosi $8.49723 \cdot 10^{-13}$. Norma rzędu 10^{-13} jest bardzo mała i niemalże równa zero, co świadczy o wysokiej dokładności rozwiązania.

6 Zadanie E

W tym zadaniu należało stworzyć wykres zależności czasu trwania poszczególnych algorytmów od niewiadomych N = {100, 500, 1000, 2000, 3000, 5000 } dla równania **(1)**.



Rysunek 1: Wykres zależności czasu algorytmu od wielkości macierzy

7 Wnioski

Z tabeli (Tabela 1.) wynika, że metoda Gaussa-Seidla jest około dwa razy szybsza od Jacobiego. Jednakże, w projekcie została wykorzystana nieoptymalna wersja metody Jacobiego. Można ją zrównoleglić ze względu na fakt, że poszczególne wartości wektora x^k są wyliczane z wektora x^{k-1} z poprzedniej iteracji, w przeciwieństwie do Gaussa-Seidla, gdzie wykorzystywane są wartości wyliczone w bierzącej iteracji.

Z wykresu (Rysunek 1.) wynika, że metody iteracyjne przeważają czasowo na metodą bezpośrednią. Wynika to ze złożoności algorytmów, gdzie metoda Jacobiego i Gaussa jest złożoności O(n^2), gdzie LU to O(n^3). Można wywnioskować, że dla macierzy o dużych rozmiarach lepiej wykorzystywać metody iteracyjne. Należy jednak mieć na uwadze, że dokładność tych metod jest mniejsza od LU, co może mieć ogromne znaczenie w szczególnych przypadkach.

Ponadto, dane mogą się tak dobrać, że metoda iteracyjna nie zbiegnie się do poprawnego rozwiązania, tak jak to się zdarzyło w równaniu **(2)** (Zadanie C), gdzie tylko metoda LU dała rozwiązanie (Zadanie D).