Sprawozdanie - Projekt 2
Układy równań liniowych
Implementacja metod Jacobiego, Gaussa-Seidla i LU

Damian Jankowski s188597

27 kwietnia 2023

Spis treści

1 Wstęp

Celem projektu było zaimplementowanie metod Jacobiego, Gaussa-Seidla i LU oraz porównanie wydajności, dokładności, jak również czasu ich wykonania.

Każda z metod w różny sposób rozwiązuje pewien układ równań liniowych:

$$\boldsymbol{Ax} = \boldsymbol{b} \tag{1}$$

gdzie:

- $\boldsymbol{A}$ – macierz kwadratowa zawierająca współczynniki układu równań,
- $\boldsymbol{b}$ – wektor wyrazów wolnych,
- $\boldsymbol{x}$ – wektor rozwiązań układu.

Macierz A została zdefiniowana jako macierz pasmowa o rozmiarze 997×997:

$$A = \begin{bmatrix} a_1 & a_2 & a_3 & 0 & 0 & 0 & 0 & \dots & 0 \\ a_2 & a_1 & a_2 & a_3 & 0 & 0 & 0 & \dots & 0 \\ a_3 & a_2 & a_1 & a_2 & a_3 & 0 & 0 & \dots & 0 \\ 0 & a_3 & a_2 & a_1 & a_2 & a_3 & 0 & \dots & 0 \\ 0 & 0 & a_3 & a_2 & a_1 & a_2 & a_3 & \dots & 0 \\ 0 & 0 & 0 & a_3 & a_2 & a_1 & a_2 & \dots & 0 \\ 0 & 0 & 0 & 0 & a_3 & a_2 & a_1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & a_3 & a_2 & a_1 \end{bmatrix} \tag{2}$$

gdzie:

$$a_1 = 10, \quad a_2 = -1, \quad a_3 = -1$$

Wektor b długości 997 został zdefiniowany jako:

$$b = \begin{bmatrix} sin(0 \cdot (f+1)) \\ sin(1 \cdot (f+1)) \\ sin(2 \cdot (f+1)) \\ \vdots \\ sin(996 \cdot (f+1)) \end{bmatrix} \quad f = 8 \tag{3}$$

2 Metody rozwiązywania układów równań liniowych

Istnieje wiele metod rozwiązywania układów równań liniowych. W projekcie zostały zaimplementowane trzy z nich: metoda Jacobiego, Gaussa-Seidla i LU.

Pierwsze dwa należą do grupy metod iteracyjnych, natomiast ostatnia jest metodą bezpośrednią.

2.1 Metody iteracyjne

Metody iteracyjne polegają na wyznaczeniu kolejnych przybliżeń rozwiązania układu równań liniowych.

Korzystają one z tzw. macierzy: trójkątnej dolnej (*Lower*) L, górnej (*Upper*) U oraz diagonalnej D, które spełniają warunek:

$$\boldsymbol{A} = \boldsymbol{L} + \boldsymbol{U} + \boldsymbol{D} \tag{4}$$

Warunkiem zakończenia iteracji jest osiągnięcie zadanej dokładności lub maksymalnej liczby iteracji.

2.1.1 Metoda Jacobiego

Metoda Jacobiego polega na wyznaczeniu kolejnych przybliżeń rozwiązania układu równań liniowych zgodnie ze wzorem:

$$\boldsymbol{x^{(k+1)}} = -\boldsymbol{D^{-1}}(\boldsymbol{L} + \boldsymbol{U})\boldsymbol{x^{(k)}} + \boldsymbol{D^{-1}b} \tag{5}$$

gdzie:

- $\boldsymbol{x^{(k)}}$ - wektor przybliżenia rozwiązania w k-tej iteracji

2.1.2 Metoda Gaussa-Seidla

Metoda Gaussa-Seidla podobnie jak metoda Jacobiego polega na wyznaczeniu kolejnych przybliżeń, jednakże zgodnie z tym wzorem:

$$\boldsymbol{x^{(k+1)}} = -(\boldsymbol{D} + \boldsymbol{L})^{-1}\boldsymbol{U}\boldsymbol{x^{(k)}} + (\boldsymbol{D} + \boldsymbol{L})^{-1}\boldsymbol{b} \tag{6}$$

Problemem tej metody jest konieczność wyznaczenia macierzy $(D+L)^{-1}$, czego powinno się unikać z racji dużej złożoności obliczeniowej.

Z tego powodu zamiast wyznaczać odwrotność macierzy $D+L$, stosuje się tzw. podstawienie w przód (ang. *forward substitution*).

2.1.3 Podstawienie w przód

Metoda podstawienia w przód polega na wyznaczeniu kolejnych wartości wektora rozwiązań $\boldsymbol{x}$ układu równań $\boldsymbol{Lx} = \boldsymbol{b}$, w następujący sposób:

$$\boldsymbol{x_1} = \frac{\boldsymbol{b_1}}{\boldsymbol{l_{11}}}$$

$$\boldsymbol{x_2} = \frac{\boldsymbol{b_2} - \boldsymbol{l_{21}x_1}}{\boldsymbol{l_{22}}}$$

$$\boldsymbol{x_i} = \frac{\boldsymbol{b_i} - \sum_{j=1}^{i-1} \boldsymbol{l_{ij}x_j}}{\boldsymbol{l_{ii}}}$$

$$\vdots$$

$$\boldsymbol{x_n = \frac{b_n - \sum_{j=1}^{n-1} l_{nj} x_j}{l_{nn}}}$$

Koniecznym jest by macierz $\boldsymbol{L}$ była macierzą trójkątną dolną, np. w przypadku sumy macierzy $D+L$.

2.1.4 Podstawienie w tył

Metoda podstawienia w tył podobnie jak metoda podstawienia w przód polega na wyznaczeniu kolejnych wartości wektora rozwiązań $\boldsymbol{x}$ tym razem układu równań $\boldsymbol{Ux = b}$. Jednakże w tym przypadku koniecznym jest by macierz $\boldsymbol{U}$ była macierzą trójkątną górną. Kolejne kroki wyglądają następująco:

$$\boldsymbol{x_n = \frac{b_n}{u_{nn}}}$$

$$\boldsymbol{x_{n-1} = \frac{b_{n-1} - u_{n-1,n} x_n}{u_{n-1,n-1}}}$$

$$\boldsymbol{x_i = \frac{b_i - \sum_{j=i+1}^{n} u_{ij} x_j}{u_{ii}}}$$

$$\vdots$$

$$\boldsymbol{x_1 = \frac{b_1 - \sum_{j=2}^{n} u_{1j} x_j}{u_{11}}}$$

2.1.5 Warunek zakończenia

By sprawdzić czy osiągnięto zadaną dokładność należy przy każdej iteracji sprawdzać czy norma tzw. wektora residuum $\boldsymbol{res}$ jest mniejsza od zadanej wartości, np. 10^{-9}.

Wektor residuum jest zdefiniowany następująco:

$$\boldsymbol{res^{(k)} = Ax^{(k)} - b} \tag{7}$$

W idealnej sytuacji powinien być równy wektorowi zerowemu.

Natomiast w większości przypadków, by sprawdzić czy osiągnięto zadaną dokładność wyznacza się normę wektora residuum $\boldsymbol{res}$:

$$||\boldsymbol{res^{(k)}}|| = \sqrt{\sum_{i=1}^{n} (\boldsymbol{res_i^{(k)}})^2} \tag{8}$$

Tym sposobem możemy z góry określić dokładność rozwiązania.

2.2 Metody bezpośrednie

Metody bezpośrednie polegają na wyznaczeniu rozwiązania układu równań bezpośrednio z macierzy współczynników $\boldsymbol{A}$. Odznaczają się one dużą dokładnością, jednakże są czasochłonne i zasobożerne.

2.2.1 Metoda faktoryzacji LU

Metoda faktoryzacji LU polega na rozkładzie macierzy współczynników $\boldsymbol{A}$ na iloczyn macierzy $\boldsymbol{L}$ i $\boldsymbol{U}$:

$$\boldsymbol{A = LU} \tag{9}$$

gdzie:

- $\boldsymbol{L}$ - macierz trójkątna dolna

- $\boldsymbol{U}$ - macierz trójkątna górna

Wtedy układ równań $\boldsymbol{Ax} = \boldsymbol{b}$ można zapisać jako:

$$\boldsymbol{LUx} = \boldsymbol{b}$$

By przejść dalej konieczne jest wyznaczenie potrzebnych macierzy pomocniczych.

1. Na początku tworzy się macierz $\boldsymbol{L}$, która jest macierzą jednostkową, czyli taką, której elementy na głównej przekątnej są równe 1. Natomiast macierz $\boldsymbol{U}$ to kopia macierzy $\boldsymbol{A}$.

   Faktoryzację LU opisać można tym kodem w języku C:

```c
for (int k = 0; k < n - 1; k++) {
    for (int j = k + 1; j < n; j++) {
        L[j][k] = U[j][k] / U[k][k];
        for (int i = k; i < n; i++) {
            U[j][i] = U[j][i] - L[j][k] * U[k][i];
        }
    }
}
```

2. Następnie metodą podstawiania w przód wyznacza się wektor $\boldsymbol{y}$, który jest rozwiązaniem układu równań $\boldsymbol{Ly} = \boldsymbol{b}$.

3. Ostatnim krokiem jest wyznaczenie wektora rozwiązań $\boldsymbol{x}$, który jest rozwiązaniem układu równań $\boldsymbol{Ux} = \boldsymbol{y}$. Z racji, że macierz $\boldsymbol{U}$ jest macierzą trójkątną górną, to wyznaczenie wektora $\boldsymbol{x}$ jest możliwe metodą podstawienia w tył.

3 Implementacja i analiza wyników

3.1 Zadanie A

Zadanie A polegało na zaimplementowaniu układu równań przedstawionego we wstępie.

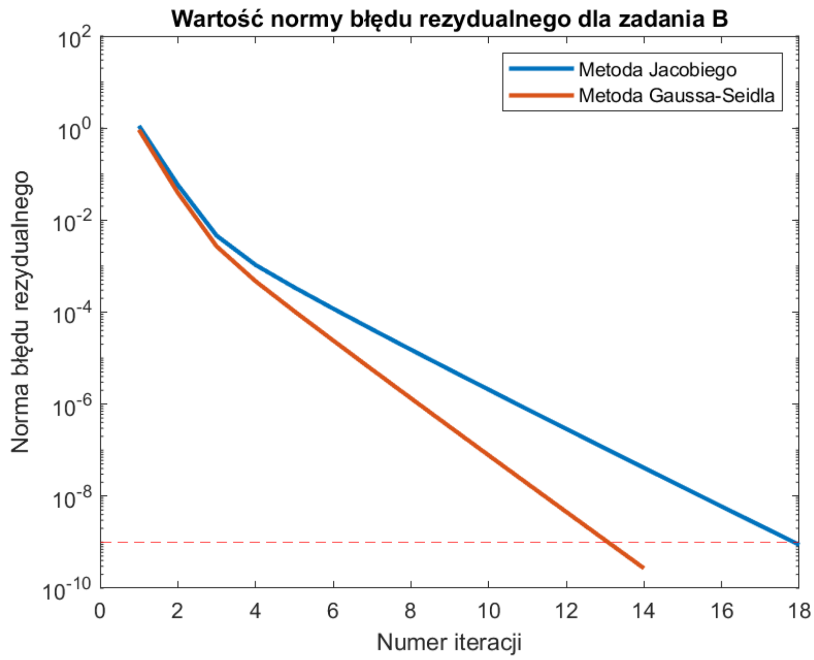
3.2 Zadanie B

Zadanie B polegało na zaimplementowaniu metod iteracyjnych Jasobiego i Gaussa-Seidla, sprawdzeniu ilość iteracji potrzebnych do zakończenia.

Wyniki zostały przedstawione poniżej.

Tabela 1. Porównanie czasu i ilości iteracji dla zadania A

Metoda	Czas	Ilość iteracji
Jacobi	0.12s	18
Gauss-Seidel	0.094s	14



Rysunek 1: Wykres przedstawiający wartość normy wektora residuum od iteracji