1. Metody rozwiązywania układów równań liniowych

1.1. Metoda eliminacji Gaussa

Układ n równań liniowych można zapisać w postaci:

$$\begin{aligned} a_{00}x_0 + a_{01}x_1 + \cdots + a_{0n}x_n &= b_0 \\ a_{10}x_0 + a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ a_{n0}x_0 + a_{n1}x_1 + \cdots + a_{nn}x_n &= b_n \end{aligned} \tag{1}$$

W metodzie tej wyróżnia się dwa etapy:

- postępowanie proste (etap eliminacji),
- postępowanie odwrotne.

Postępowanie proste polega na sprowadzeniu układu do postaci górnie trójkątnej. W tym celu odejmujemy od i-tego wiersza, wiersz zerowy pomnożony przez mnożnik:

$$m_{i0} = \frac{a_{i0}}{a_{00}} \tag{2}$$

Po wykonaniu tego kroku otrzymujemy:

$$\begin{aligned} a_{00}x_0 + a_{01}x_1 + \cdots + a_{0n}x_n &= b_0 \\ a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\ \dots\dots\dots\dots\dots\dots\dots \\ a_{n1}x_1 + \cdots + a_{nn}x_n &= b_n \end{aligned} \tag{3}$$

W kolejnym kroku odejmujemy od i-tego wiersza, wiersz pierwszy pomnożony przez mnożnik:

$$m_{i1} = \frac{a_{i1}}{a_{11}} \tag{4}$$

Postępujemy tak, aż do uzyskania macierzy w postaci:

$$\begin{aligned} a_{00}x_0 + a_{01}x_1 + \cdots + a_{0n}x_n &= b_0 \\ a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\ \dots\dots\dots\dots\dots \\ a_{nn}x_n &= b_n \end{aligned} \tag{5}$$

W drugim etapie (postępowanie odwrotne) w celu znalezienia rozwiązania układu równań, korzysta się z uzyskanej macierzy trójkątnej górnej i wzorów:

$$x_n = \frac{b_n}{a_{nn}} \tag{6}$$

$$x_i = \frac{b_i - \sum_{k=i+1}^{n} a_{ik}\, x_k}{a_{ii}} \quad dla\ i = n-1, \dots 0$$

Przykład

$$\begin{cases} 2x_0 + 4x_1 + 2x_2 + 1x_3 = 10 \\ 2x_0 + 2x_1 + 3x_2 + 3x_3 = 6 \\ 4x_0 + 2x_1 + 2x_2 + 1x_3 = 6 \\ 0x_0 + 2x_1 + 1x_2 + 1x_3 = 4 \end{cases} \tag{7}$$

Układ równań dany wzorem (7) przedstawiamy w formie macierzy rozszerzonej (na czerwono oznaczono numerację wierszy i kolumn, ostatnia kolumna reprezentuje wektor prawej strony układu):

$$\begin{array}{c c} & \begin{matrix} 0 & 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} 2 & 4 & 2 & 1 & 10 \\ 2 & 2 & 3 & 3 & 6 \\ 4 & 2 & 2 & 1 & 6 \\ 0 & 2 & 1 & 1 & 4 \end{bmatrix} \end{array} \tag{8}$$

W pierwszym etapie obliczamy mnożnik:

$$m_{10} = \frac{a_{10}}{a_{00}} = \frac{2}{2} = 1 \tag{9}$$

Następnie odejmujemy od pierwszego wiersza wiersz zerowy pomnożony przez m_{10}. Analogiczne postępujemy dla drugiego wiersza: obliczamy $m_{20} = 2$ i odejmujemy od wiersza drugiego wiersz zerowy pomnożony przez mnożnik m_{20}. To samo liczymy dla trzeciego wiersza. Ostatecznie otrzymujemy same zera pod przekątną dla zerowej kolumny:

$$\begin{bmatrix} 2 & 4 & 2 & 1 & 10 \\ 0 & -2 & 1 & 2 & -4 \\ 0 & -6 & -2 & -1 & -14 \\ 0 & 2 & 1 & 1 & 4 \end{bmatrix} \tag{10}$$

W kolejnym kroku zerujemy pod przekątną kolejną kolumnę (pierwszą). Obliczamy:

$$m_{21} = \frac{a_{21}}{a_{11}} = \frac{-6}{-2} = 3 \tag{11}$$

Odejmujemy od drugiego wiersza, wiersz pierwszy pomnożony przez m_{21}. Powtarzamy kroki dla ostatniego wiersza (trzeciego). Otrzymujemy:

$$\begin{bmatrix} 2 & 4 & 2 & 1 & 10 \\ 0 & -2 & 1 & 2 & -4 \\ 0 & 0 & -5 & -7 & -2 \\ 0 & 0 & 2 & 3 & 0 \end{bmatrix} \tag{12}$$

Analogicznie kroki wykonujemy dla ostatniego wiersza w celu wyzerowania drugiej kolumny. Ostatecznie otrzymujemy:

$$\begin{bmatrix} 2 & 4 & 2 & 1 & 10 \\ 0 & -2 & 1 & 2 & -4 \\ 0 & 0 & -5 & -7 & -2 \\ 0 & 0 & 0 & 0.2 & -0.8 \end{bmatrix} \tag{13}$$

W drugim etapie (postępowanie odwrotne) w celu znalezienia rozwiązania układu równań, korzystamy ze wzorów:

$$\begin{gathered} x_n = \frac{b_n}{a_{nn}} \\ x_3 = \frac{b_3}{a_{33}} = \frac{-0.8}{0.2} = -4 \\ x_i = \frac{b_i - \sum_{k=i+1}^{n} a_{ik}\, x_k}{a_{ii}} \quad dla\ i = n-1, \dots 0 \\ x_2 = \frac{b_2 - \sum_{k=2+1}^{3} a_{23}\, x_3}{a_{22}} = \frac{-2-(-7)*(-4)}{-5} = 6 \\ x_1 = \frac{-4-[1*6+2*(-4)]}{-2} = 1 \\ x_0 = \frac{10-[4*1+2*6+1*(-4)]}{2} = -1 \end{gathered} \tag{14}$$

Przedstawiony wyżej algorytm działa dopóki na przekątnej macierzy nie ma wartości zera. W celu rozwiązania problemu „0” na przekątnej można zastosować wybór częściowy (z ang. partial pivoting). Polega on na modyfikacji algorytmu w taki sposób, że zamieniamy kolejność wierszy w macierzy tak, by element na diagonali przez który dzielimy wiersz, był największym elementem w danej kolumnie w sensie wartości bezwzględnej.

Przykład rozwiązywania układu równań z wyborem częściowym dla kilku pierwszych iteracji:

```
Uklad rownan:
2       4       2       1       10
2       2       3       3       6
4       2       2       1       6
0       2       1       1       4


4       2       2       1       6
2       2       3       3       6
2       4       2       1       10
0       2       1       1       4

4       2       2       1       6
0       1       2       2.5     3
2       4       2       1       10
0       2       1       1       4

4       2       2       1       6
0       1       2       2.5     3
0       3       1       0.5     7
0       2       1       1       4

4       2       2       1       6
0       3       1       0.5     7
0       1       2       2.5     3
0       2       1       1       4
```

1.2. Metoda Jacobiego

Metoda Jacobiego jest to iteracyjną metodą rozwiązywania układu równań liniowych. Metody iteracyjne polegają na konstruowaniu ciągu przybliżeń wektora rozwiązań $x^{(0)}, x^{(1)} \dots x^{(i)}$ określonego wzorem:

$$x^{(i+1)} = Mx^{(i)} + w \tag{15}$$

gdzie: $i = 0,1 \dots$, M – macierz kwadratowa, w – wektor.

Rozważmy układ równań:

$$Ax = b \tag{16}$$

Macierz A rozkładamy na 3 macierze:

$$A = L + D + U \tag{17}$$

gdzie: L – macierz trójkątna dolna, D – macierz diagonalna, U – macierz trójkątna górna.

Wstawiając równanie (17) do (16) możemy przekształcić kolejno:

$$(L + D + U)x = b \tag{18}$$

$$Dx = -(L + U)x + b \tag{19}$$

$$x = -D^{-1}(L + U)x + D^{-1}b \tag{20}$$

Ciąg przybliżeń rozwiązania przyjmuje następującą postać:

$$x^{(i+1)} = -D^{-1}(L + U)x^{(i)} + D^{-1}b \tag{21}$$

Metoda Jacobiego jest zbieżna dla macierzy nieredukowalnych i diagonalnie słabo dominujących.

Macierz $A = (a_{ij})$ nazywamy diagonalnie słabo dominującą jeśli dla $i = 0,1 \dots n$ spełniane są warunki:

$$|a_{ii}| \geq \sum_{\substack{j=0 \\ j \neq i}}^{n} |a_{ij}| \tag{22}$$

oraz spełniony jest co najmniej jeden warunek dla dowolnego i:

$$|a_{ii}| > \sum_{\substack{j=0 \\ j \neq i}}^{n} |a_{ij}| \tag{23}$$

Przykład 2

```
Uklad rownan:
8       2       2       4       5
2       5       1       1       -4
0       3       4       1       2
-1      -2      1       5       7

Macierz L+U:
0       2       2       4
2       0       1       1
0       3       0       1
-1      -2      1       0

Macierz diagonalna odwrotna (Dodw):
0.125   0       0       0
0       0.2     0       0
0       0       0.25    0
0       0       0       0.2

Rozwiazanie po 5 iteracjach:
x[0]: 0.28535
x[1]: -1.2878
x[2]: 1.28868
x[3]: 0.692447
```

Sprawozdanie zawierające wyniki (zrzuty ekranu z konsoli) przesyłamy do odpowiednio zdefiniowanego zadania na platformie UPEL (np. MN-1 - gr1).

Plik z kodem *.cpp przesyłamy do wirtualnego laboratorium (WU-1). Plik *.cpp powinien zawierać rozwiązanie wszystkich zadań z możliwością wyboru rozwiązania.

Zad 1. Napisz program, który będzie rozwiązywał układ n równań liniowych o n niewiadomych metodą Gaussa (10p). Wymagania:

a) Dane pobierane są z pliku.
b) W przypadku wystąpienia 0 na przekątnej macierzy, program wypisze stosowny komunikat.
c) W wyniku działania program wypisuje:
   - Macierz rozszerzoną (przed obliczeniami)
   - Macierz rozszerzoną (po pierwszym etapie obliczeń – postępowanie proste)
   - Rozwiązanie układu równań ($x_0 - x_n$)

Poprawność działania programu zweryfikować danymi, które podano w przykładzie w rozdziale 1.1.

Rozwiązać układ równań podany w pliku tekstowym: RURL_dane2.txt

To zadanie należy oddać na bieżących zajęciach.

Zad 2. Napisz program, który będzie rozwiązywał układ n równań liniowych o n niewiadomych metodą Jacobiego. Wymagania (5p):

- Dane pobierane są z pliku.
- Program wypisze układ równań (macierz rozszerzoną), sprawdzi czy macierz jest diagonalnie słabo dominująca i wyświetli stosowny komunikat.
- Warunkiem zatrzymania algorytmu jest podana przez użytkownika ilość iteracji.
- Program wypisze macierze: $L + U$ oraz D^{-1}.
- Program wypisze zadaną ilość iteracji i rozwiązanie układu równań.

W sprawozdaniu należy zamieścić rozwiązanie układu równań przedstawionego w przykładzie dla 5 iteracji. Za początkowe wartości wektora x przyjąć 0. Porównaj wyniki uzyskane metodą Jacobiego i Gaussa. Oblicz błąd bezwzględny dla każdego x.

To zadanie można oddać na kolejnych zajęciach.

Zad 3. Zmodyfikuj program (napisać dodatkową funkcję), tak aby rozwiązywał układ n równań liniowych o n niewiadomych z pivoting'iem. Rozwiąż następujący układ równań podany w postaci macierzy rozszerzonej (5p):

$$\begin{bmatrix} 2 & 4 & 2 & 1 & 10 \\ 1 & 2 & 3 & 3 & 6 \\ 4 & 5 & 2 & 1 & 6 \\ 0 & 1 & 2 & 9 & 1 \end{bmatrix}$$

To zadanie można oddać na kolejnych zajęciach.