

Sprawozdanie

Metody Numeryczne 2, laboratorium 1

Grzegorz Rozdzialik (D4, grupa lab. 2)

13 października 2016

1 Zadanie

Rozwiązywanie układu równań z macierzą trójdagonalną w dziedzinie zespolonej metodą Gaussa-Seidla w tył (Backwards Gauss-Seidel).

Szukamy rozwiązania układu równań $Ax = b$, gdzie $A \in \mathbb{C}^{n \times n}$ (A - macierz trójdagonalna, $\det A \neq 0$), $b \in \mathbb{C}^n$, $n \in \mathbb{N}$.

Układ $Ax = b$ można zapisać w postaci:

$$\begin{bmatrix}
 d_1 & u_1 & 0 & 0 & 0 & \dots & 0 \\
 l_2 & d_2 & u_2 & 0 & 0 & \dots & 0 \\
 0 & l_3 & d_3 & u_3 & 0 & \dots & 0 \\
 0 & 0 & l_4 & d_4 & u_4 & \dots & 0 \\
 & & & \ddots & \ddots & \ddots & \vdots \\
 \vdots & \vdots & & 0 & l_{n-3} & d_{n-3} & u_{n-3} & 0 & 0 \\
 & & & & 0 & l_{n-2} & d_{n-2} & u_{n-2} & 0 \\
 & & & & & 0 & l_{n-1} & d_{n-1} & u_{n-1} \\
 0 & 0 & \dots & \dots & \dots & 0 & l_n & d_n
 \end{bmatrix}
 \begin{bmatrix}
 x_1 \\
 x_2 \\
 x_3 \\
 \vdots \\
 x_{n-1} \\
 x_n
 \end{bmatrix}
 =
 \begin{bmatrix}
 b_1 \\
 b_2 \\
 b_3 \\
 \vdots \\
 b_{n-1} \\
 b_n
 \end{bmatrix} \quad (1)$$

gdzie $l_i, d_i, u_i, b_i, x_i \in \mathbb{C}^n$ dla każdego $i = 1, \dots, n$.

2 Opis metody

Metoda Gaussa-Seidla w tył jest iteracyjną metodą rozwiązywania układów równań liniowych. Jest ona podobna do zwykłej metody Gaussa-Seidla, przy czym w tym przypadku obliczenia rozpoczynamy od ostatnich elementów wektora rozwiązań.

Mając przybliżenie początkowe $x^{(0)} \in \mathbb{C}^n$ tworzymy ciąg kolejnych przybliżeń rozwiązań $\{x^{(k)}\}$ ($k = 1, 2, \dots$) taki, że

$$\lim_{k \rightarrow +\infty} x^{(k)} = x$$

Po wymnożeniu lewej strony układu (1) otrzymujemy następujący układ równań:

$$\left\{ \begin{array}{l} d_1 x_1 + u_1 x_2 = b_1 \\ l_2 x_1 + d_2 x_2 + u_2 x_3 = b_2 \\ l_3 x_2 + d_3 x_3 + u_3 x_4 = b_3 \\ \vdots \\ l_i x_{i-1} + d_i x_i + u_i x_{i+1} = b_i \\ \vdots \\ l_{n-1} x_{n-2} + d_{n-1} x_{n-1} + u_{n-1} x_n = b_{n-1} \\ l_n x_{n-1} + d_n x_n = b_n \end{array} \right.$$

Następnie z i -tego równania wyliczamy:

$$x_i = \frac{b_i - l_i x_{i-1} - u_i x_{i+1}}{d_i} \quad \text{dla } i = 2, \dots, n-1$$

Wyjątkiem są równania pierwsze oraz ostatnie, ponieważ nie istnieją współczynniki odpowiednio l_1 oraz u_n , stąd:

$$x_1 = \frac{b_1 - u_1 x_2}{d_1}$$
$$x_n = \frac{b_n - l_n x_{n-1}}{d_n}$$

W k -tym kroku obliczamy przybliżenie $x^{(k+1)}$ w taki sposób, że najpierw obliczamy $x_n^{(k+1)}$, następnie zmniejszamy indeks dolny, obliczając kolejno $x_{n-1}^{(k+1)}, \dots, x_1^{(k+1)}$.

Przy wyliczaniu $x_i^{(k+1)}$ używamy najbardziej aktualnych przybliżeń (za-
miast $x_{i+1}^{(k)}$ używamy $x_{i+1}^{(k+1)}$, wyjątkiem jest $x_n^{(k)}$).

Należy oddzielnie przybliżać część rzeczywistą oraz część urojoną.

3 Implementacja metody

Wprowadźmy następujące oznaczenia:

$$\begin{array}{ll} u_{r,j} = \operatorname{Re}(u_j) & u_{i,j} = \operatorname{Im}(u_j) \\ d_{r,j} = \operatorname{Re}(d_j) & d_{i,j} = \operatorname{Im}(d_j) \\ l_{r,j} = \operatorname{Re}(l_j) & l_{i,j} = \operatorname{Im}(l_j) \\ x_{r,j}^{(k)} = \operatorname{Re}(x_j^{(k)}) & x_{i,j}^{(k)} = \operatorname{Im}(x_j^{(k)}) \\ b_{r,j} = \operatorname{Re}(b_j) & b_{i,j} = \operatorname{Im}(b_j) \end{array}$$

dla $j = 1, \dots, n$.

Poniższe wzory opisują sposób obliczenia kolejnego przybliżenia $x^{(k+1)}$:

$$x_{r,n}^{(k+1)} = \frac{b_{r,n} + d_{i,n}x_{i,n}^{(k)} - l_{r,n}x_{r,n-1}^{(k)} + l_{i,n}x_{i,n-1}^{(k)}}{d_{r,n}}$$

$$x_{i,n}^{(k+1)} = \frac{b_{i,n} - d_{i,n}x_{r,n}^{(k)} - l_{r,n}x_{i,n-1}^{(k)} - l_{i,n}x_{r,n-1}^{(k)}}{d_{r,n}}$$

dla $j = (n-1), \dots, 2$

$$x_{r,j}^{(k+1)} = \frac{b_{r,j} + d_{i,j}x_{i,j}^{(k)} - l_{r,j}x_{r,j-1}^{(k)} + l_{i,j}x_{i,j-1}^{(k)} - u_{r,j}x_{r,j+1}^{(k+1)} + u_{i,j}x_{i,j+1}^{(k+1)}}{d_{r,j}}$$

$$x_{i,j}^{(k+1)} = \frac{b_{i,j} - d_{r,j}x_{i,j}^{(k)} - l_{r,j}x_{i,j-1}^{(k)} - l_{i,j}x_{r,j-1}^{(k)} - u_{i,j}x_{r,j+1}^{(k+1)} - u_{r,j}x_{i,j+1}^{(k+1)}}{d_{r,j}}$$

$$x_{r,1}^{(k+1)} = \frac{b_{r,1} + d_{i,1}x_{i,1}^{(k)} - u_{r,1}x_{r,2}^{(k+1)} + u_{i,1}x_{i,2}^{(k+1)}}{d_{r,1}}$$

$$x_{i,1}^{(k+1)} = \frac{b_{i,1} - d_{r,1}x_{i,1}^{(k)} - u_{i,1}x_{r,2}^{(k+1)} - u_{r,1}x_{i,2}^{(k+1)}}{d_{r,1}}$$

4 Warunek stopu

Z możliwych warunków stopu wybrałem warunek Gill'a. Obliczenia trwają dopóki spełniona jest nierówność

$$\|x^{(k+1)} - x^{(k)}\| \geq \varepsilon \|x^{(k)}\| + \delta$$

gdzie $\varepsilon, \delta > 0$.

Dodatkowo w implementacji nałożyłem restrykcję na maksymalną ilość iteracji. Po 10000 iteracjach obliczenia nie będą kontynuowane.

5 Poprawność algorytmu

Algorytm działa poprawnie dla macierzy diagonalnie dominujących (kolumnowo lub wierszowo), oraz dla macierzy dodatnio określonych. Wtedy też w ogólności metoda Gaussa-Seidla, a więc i Gaussa-Seidla w tył, jest zbieżna globalnie.

Dla macierzy nie spełniający wyżej wymienionych kryteriów, algorytm nie daje dobrych rozwiązań.

6 Przykłady

Układ 1 TODO

Układ 2 TODO