

SPRAWOZDANIE - LABORATORIUM 3

Metoda największego spadku dla macierzy wstęgowej

Zuzanna Grzesik

18 marca 2020

1 Wstęp teoretyczny

W trakcie laboratoriów zajęliśmy się tematem metody największego spadku dla macierzy wstęgowej.

1.1 Macierz wstęgowa

Jest to macierz rzadka, której elementy są zerowe poza diagonalą i wstęgą wokół niej. Poniżej przykładowa macierz wstęgowa, trójdzielna:

$$\begin{pmatrix} a_{11} & a_{12} & 0 & \cdots & \cdots & 0 \\ a_{21} & a_{22} & a_{23} & \ddots & \ddots & \vdots \\ 0 & a_{32} & a_{33} & a_{34} & \ddots & \vdots \\ \vdots & \ddots & a_{43} & a_{44} & a_{45} & 0 \\ \vdots & \ddots & \ddots & a_{54} & a_{55} & a_{56} \\ 0 & \cdots & \cdots & 0 & a_{65} & a_{66} \end{pmatrix} \quad (1)$$

Dla macierzy $n \times n$ jej element $a_{i,j}$ są różne od zera gdy

$$i - k_1 \geq j \geq i + k_2 \quad (2)$$

Gdzie $k_1, k_2 \geq 0$ określają szerokość wstęgi.

Ze względu na charakterystyczną postać tego typu macierzy, możemy je zapisać w pamięci w $n \cdot (k_1 + k_2 + 1)$ komórkach, zamiast w n^2 .

1.2 Metoda największego spadku

Jest to jedna z metod iteracyjnych rozwiązywania układów równań liniowych. Każda iteracja polega na wykonaniu kroku w lokalnie najlepszym kierunku czyli w kierunku wyznaczonym przez ujemny gradient [1].

Przyjmijmy następujące oznaczenia: \mathbf{A} - macierz współczynników, \mathbf{b} - wektor wyrazów wolnych, \mathbf{x} - wektor niewiadomych oraz ∇Q - gradient Q , gdzie Q to forma kwadratowa układu równań. W metodzie tej przybliżone rozwiązanie w $i + 1$ iteracji będzie miało postać:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{v}_i \quad (3)$$

Jako \mathbf{v}_i wybieramy kierunek gradientu Q :

$$\nabla Q = \mathbf{A}\mathbf{x}_i - \mathbf{b} = -\mathbf{r}_i, \quad (4)$$

$$\mathbf{v}_i = -\mathbf{r}_i \quad (5)$$

By znaleźć współczynnik α_i musimy obliczyć $Q(\mathbf{x}_{i+1})$:

$$Q(\mathbf{x}_{i+1}) = Q(\mathbf{x}_i - \alpha_i \mathbf{r}_i) = -\frac{1}{2} \mathbf{x}_i^T \mathbf{r}_i - \frac{1}{2} \mathbf{x}_i^T \mathbf{b} + \frac{1}{2} \alpha_i^2 \mathbf{r}_i^T \mathbf{A} \mathbf{r}_i + \alpha_i \mathbf{r}_i^T \mathbf{r}_i. \quad (6)$$

Następnie różniczkujemy po parametrze α :

$$\frac{\partial Q}{\partial \alpha_i} = \mathbf{r}_i^T \mathbf{r}_i + \alpha_i \mathbf{r}_i^T \mathbf{A} \mathbf{r}_i. \quad (7)$$

Wiedząc, że $\frac{\partial Q}{\partial \alpha_i} = 0$ otrzymujemy [2]:

$$\alpha_i = -\frac{\mathbf{r}_i^T \mathbf{r}_i}{\mathbf{r}_i^T \mathbf{A} \mathbf{r}_i}. \quad (8)$$

Dzięki obliczonym w ten sposób wartościom poszczególnych macierzy i wektorów, możemy dokonywać kolejnych przybliżeń rozwiązania układu równań.

2 Zadanie do wykonania

2.1 Opis problemu

Zadanie w trakcie laboaratoriów polegało na rozwiązaniu układu równań liniowych $\mathbf{A}\mathbf{x} = \mathbf{b}$ metodą największego spadku. Na początku należało utworzyć macierz układu równań o wymiarze $n = 1000$ i zainicjować jej elementy zgodnie ze wzorem:

$$A[i][j] = \begin{cases} \frac{1}{1+|i-j|} & \text{gdy } |i-j| \leq m \\ 0 & \text{gdy } |i-j| > m \end{cases}, \quad (9)$$

dla $m = 5$ oraz $i, j = 0, \dots, n-1$. Należało również zainicjować wektor wyrazów wolnych \mathbf{b} i uzupełnić go wg wzoru:

$$b[i] = i + 1, i = 0, \dots, n-1. \quad (10)$$

Następnym zadaniem było zaprogramowanie metody największego spadku wg instrukcji:

$$\begin{aligned} &do \{ \mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k \\ &\quad \alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k} \\ &\quad \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{r}_k \\ &\} while (\|\mathbf{r}_k\|_2 > 10^{-6}) \end{aligned}$$

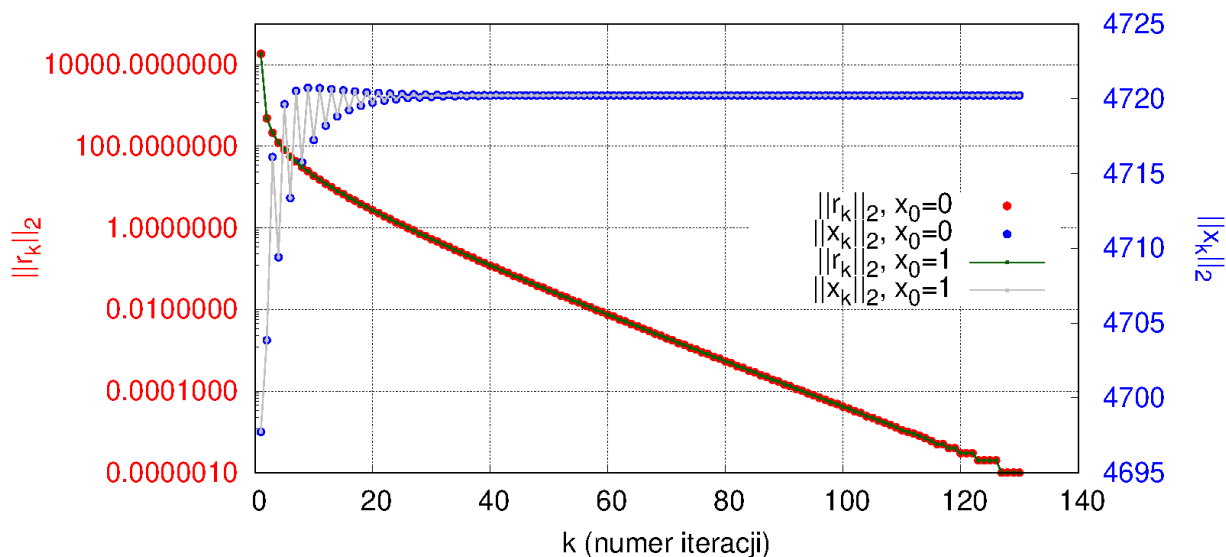
gdzie: k - numer iteracji, \mathbf{x}_k - aktualne przybliżenie wektora rozwiązań, \mathbf{r}_k jest wektorem reszt, a $\|\mathbf{r}_k\|_2$ normą euklidesową wektora reszt ($\|\mathbf{r}_k\|_2 = \sqrt{\mathbf{r}_k^T \mathbf{r}_k}$). Dla tak zaprogramowanej metody należało rozwiązać układ równań w dwóch przypadkach: dla $x_0 = 0$ oraz dla $x_0 = 1$ i zaobserwować czy dla różnych wektorów startowych zachodzą zmiany w liczbach iteracji.

Przygotowałam program z procedurą utworzoną na podstawie powyższego algorytmu. Utworzyłam odpowiednie tablice dynamiczne reprezentujące macierze i wektory, typu double - podwójnej precyzji. Stworzyłam również funkcje pomocnicze:

- mnożącą macierz i wektor,
- zwracającą iloczyn skalarny dwóch wektorów,
- dodającą dwa wektory, w tym jeden przemnożony przez jakiś współczynnik.

Wykonałam obliczenia w programie dla dwóch wariantów wektorów startowych z przyjętym warunkiem zakończenia iteracji $\|\mathbf{r}_k\|_2 > 10^{-6}$ (ponieważ prowadziłam obliczenia dla podwójnej precyzji), a wyniki (numery iteracji, wartości normy euklidesowej wektora reszt, wartości α_k , wartości normy euklidesowej wektora rozwiązań) zapisałam do pliku. Dla obu wariantów obliczeń zmierzyłam czas wykonywanych operacji korzystając z funkcji clock i wypisałam je na ekran.

Rozwiązałam również ten sam układ równań metodą eliminacji zupełnej, korzystając z gotowej funkcji gaussj i procedur do alokowania pamięci dla macierzy z bibliotek Numerical Recipes. Dla tej metody również zmierzyłam czas wykonywania obliczeń.



Rysunek 1: Wykresy funkcji $f(k) = \|r_k\|_2$ oraz $g(k) = \|x_k\|_2$, dla k - numeru iteracji

2.2 Wyniki

Uzyskane wyniki przedstawiłam na wykresie (Rysunek 1).

Na podstawie danych wyjściowych oraz wykresów można również zauważyć, że liczba wykonanych iteracji w procedurze jest taka sama dla obu wariantów wektorów startowych.

Dla wykonywanych obliczeń metodą największego spadku oraz metodą eliminacji zupełnej zmierzyłam czterokrotnie czas wykonywania operacji.

$x_0 = 0$	$x_0 = 1$	Metoda eliminacji zupełnej
0.027068	0.025454	6.612864
0.017762	0.017222	6.358071
0.018719	0.17019	6.3981231
0.016438	0.016278	6.456073

Jak widać w powyższej tabeli, czasy wykonania obliczeń dla $x_0 = 0$ i dla $x_0 = 1$ nie różnią się od siebie w dużym stopniu (co więcej wiemy też, że liczba wykonanych iteracji dla obu przypadków jest taka sama). Natomiast czas wykonania obliczeń korzystając z procedury gaussj jest znacznie dłuższy. Co więcej - gdy wykonamy obliczenia dla macierzy o wymiarach $10^4 \times 10^4$ czas wykonania obliczeń dla metody największego spadku wynosi ok. 0.4 s, a dla procedury gaussj ponad kilkadziesiąt minut (nie udało mi się zmierzyć czasu ponieważ trwało to tak długo).

3 Wnioski

Korzystając z metody największego spadku możemy w dość szybki sposób rozwiązać układ równań liniowych. W przypadku tych laboratoriów, korzystaliśmy z "pełnowymiarowej" macierzy, ale gdy odpowiednio zmodyfikujemy nasz program, będziemy mogli zapisać macierz w formie mniejszej macierzy lub kilku wektorów (ponieważ jest to macierz rzadka), dzięki czemu oszczędzimy pamięć. Metoda Gaussa-Jordana wypada w porównaniu do zaimplementowanej przez nas metody dużo wolniej, a dodatkowo wymaga od nas operowania na całej macierzy.

Literatura

- [1] Metoda najszybszego spadku <http://optymalizacja.w8.pl/NajszybszegoSpadku.html>
- [2] Tomasz Chwiej *Rozwiązywanie algebraicznych układów równań liniowych metodami iteracyjnymi*. http://galaxy.agh.edu.pl/~chwiej/mn/uarl_liter_1920.pdf