Krzysztof Czarnowus	zadanie NUM4	grupa 3

1. Wstęp

Obliczenie rozwiązań układu równań:

$$A\,x = b \tag{1}$$

dla macierzy w postaci:

$$\mathbf{A} = \begin{pmatrix} 12 & 8 & 1 & 1 & \dots & 1 & 1 & 1 & 1 \\ 1 & 12 & 8 & 1 & \dots & 1 & 1 & 1 & 1 \\ 1 & 1 & 12 & 8 & \dots & 1 & 1 & 1 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & 1 & \dots & 1 & 12 & 8 & 1 \\ 1 & 1 & 1 & 1 & \dots & 1 & 1 & 12 & 8 \\ 1 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & 12 \end{pmatrix}$$

oraz wektora $b = (5, 5, \dots, 5)^T$ możliwe jest do wykonania w liniowym czasie dzięki zastosowaniu wzoru Shermana-Morrisona:

$$(B+uv^T)^{-1} = B^{-1} - \frac{B^{-1}uv^TB^{-1}}{1+v^TB^{-1}u} \tag{2}$$

Można zauważyć, że rozkładając wyjściową macierz $A = B + uv^T$ w odpowiedni sposób, to znaczy przyjmując $u = v = (1, 1, \dots, 1)^T$ otrzymana macierz B jest macierzą rzadką, której jedyne niezerowe elementy znajdują się na diagonali oraz jeden rząd nad nią. Łącząc odpowiednio wzory 1. i 2. oraz przekształcając je, otrzymuje się wyrażenie na wektor rozwiązań układu:

$$x = (B+uv^T)^{-1}b = B^{-1}b - \frac{B^{-1}uv^Tb}{1+v^TB^{-1}u} \tag{3}$$

Aby uniknąć odwracania macierzy, można zdefiniować pomocnicze wektory y oraz z jako rozwiązania układów:

$$B\,y = b \tag{4}$$

$$B\,z = u \tag{5}$$

Po ich wstawieniu do wzoru 3. jako y = B^{-1} b oraz z = B^{-1} u otrzymuje się ostateczny wzór:

$$x = y - \frac{(v^T y) z}{1 + v^T z} \quad (6)$$

2. Opracowanie wyników

Napisano program w języku C++ rozwiązujący wyjściowy układ równań przy wykorzystaniu wzoru Shermana-Morrisona. Wymiar badanej macierzy podawany jest jako argument wejściowy programu. Ze względu na prosty kształt macierzy wstęgowej B nie było konieczne przechowywanie jej w pamięci – została ona wykorzystana jedynie do szybkiego wyliczenia wektorów y oraz z ze wzorów 4. oraz 5. metodą back substitution. Otrzymane tak wektory wstawiono do wzoru 6., otrzymując gotowy wektor rozwiązań układu równań. Dla wymiaru N = 80 ma on postać:

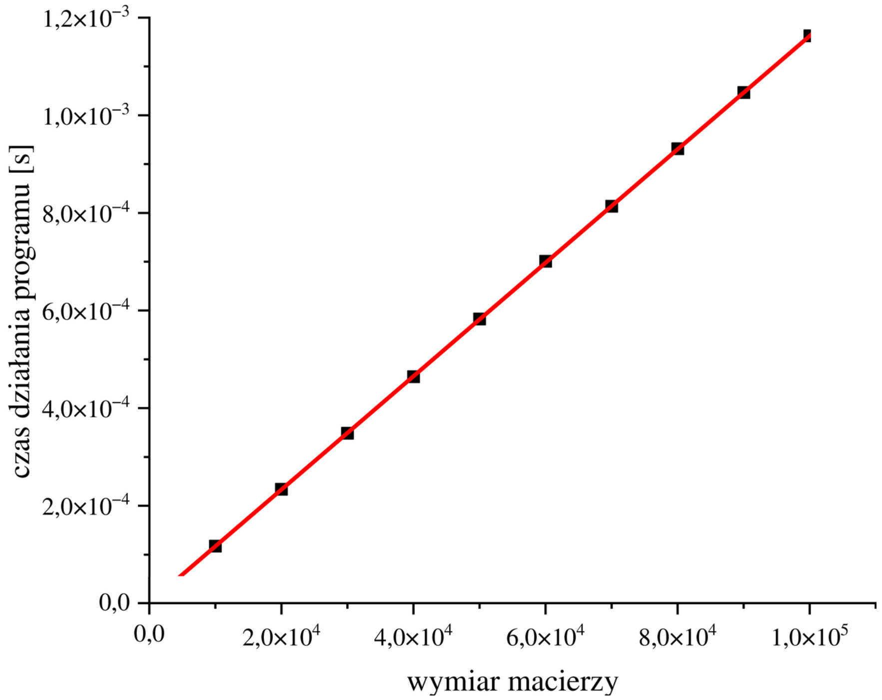
$$x = \begin{bmatrix} 0.0508187 \\ 0.0508187 \\ 0.0508187 \\ . \\ . \\ . \\ 0.0302393 \\ 0.0831579 \end{bmatrix}$$

Otrzymany wynik sprawdzono pisząc program w języku Python, rozwiązujący zadany układ równań z użyciem biblioteki NumPy; uzyskano identyczne rozwiązanie. Jest to metoda prostsza programistycznie, ale bardzo kosztowna obliczeniowo, ponieważ posiadająca złożoność rzędu $\mathcal{O}(n^3)$, co pokazano rozwiązując zadanie NUM3.

Tabela 1. Czas obliczania rozwiązania równania dla różnych wymiarów macierzy A.

wymiar macierzy	czas działania [ms]	wymiar macierzy	czas działania [s]
10 000	0.1172	60 000	0.7007
20 000	0.2338	70 000	0.8136
30 000	0.3484	80 000	0.9313
40 000	0.4644	90 000	1.0464
50 000	0.5826	100 000	1.1628

Dla dedykowanego do zadanego równania programu stworzonego w języku C++ zmierzono czas działania dla wymiarów macierzy z zakresu 10 000 – 100 000 z krokiem równym 10 000. Każde obliczenia powtórzono 5000 razy, aby uzyskać odpowiednio wysoką wartość, po czym podzielono go przez ilość powtórzeń, aby znać dokładny czas jednokrotnego przeprowadzenia obliczeń. Otrzymane wartości przedstawiono w tabeli 1. oraz na rysunku 1.



Rysunek 1. Zależność czasu działania programu od różnych wymiarów macierzy wraz z dopasowaną w programie Origin prostą o współczynniku $R^2 = 0{,}99$.

Zaobserwowana zależność czasu działania od rozmiaru danych wejściowych jest liniowa, co wskazuje na to, że napisany program ma złożoność obliczeniową $\mathcal{O}(n)$, zgodnie z oczekiwaniami.

3. Podsumowanie

Chociaż macierz występująca w badanym układzie równań nie była macierzą rzadką, dało się ją w prosty sprowadzić do postaci macierzy wstęgowej za pomocą zastosowania wzoru Shermana-Morrisona. Pokazało to użyteczność zastosowanego narzędzia, które w wyraźny sposób wpłynęło na znaczną poprawę złożoności obliczeniowej rozwiązania rozważanego problemu.