Metody numeryczne-Układy równań liniowych

Adam Białek 193677

1. **Wstęp**

Celem projektu jest implementacja, analiza oraz porównanie trzech metod rozwiązywania układów równań liniowych: Jacobiego, Gaussa-Seidla oraz faktoryzacji LU.

Równanie rozwiązywane w projekcie:

Ax = b

Zadanie zostało zrealizowane w językach C++ oraz Matlab (Matlab wykorzystany został tylko do sporządzenia wykresów). W większości realizowanych zadań macierz A oraz wektor b będą przyjmowały następującą postać:

A:

11	-1	-1	0	0	0
-1	11	-1	-1	0	0
-1	-1	11	-1	-1	0
0	-1	-1	11	-1	-1
0	0	-1	-1	11	-1
0	0	0	-1	-1	11

Macierz A na głównej diagonali ma same wartości 11, a na diagonalach sąsiednich i skrajnych wartości -1. Macierz przedstawiona ma wymiary 6x6, a macierz używana w zadaniach ma wymiary 977x977.

b:

0
-0.76
0.99
-0.54
-0.29
0.91

Kolejne elementy wektora pionowego b są wyliczane zgodnie ze wzorem: **sin(4*n)** gdzie n jest n-tym elementem elementem wektora (liczonym od 0). Przedstawiony wektor ma 6 elementów, a wektor używany w zadaniach ma 977 elementów.

W obliczeniach dla metod iteracyjnych maksymalna liczba iteracji wynosi 1000.
Wektor x jest wypełniany początkowo wartościami 1.

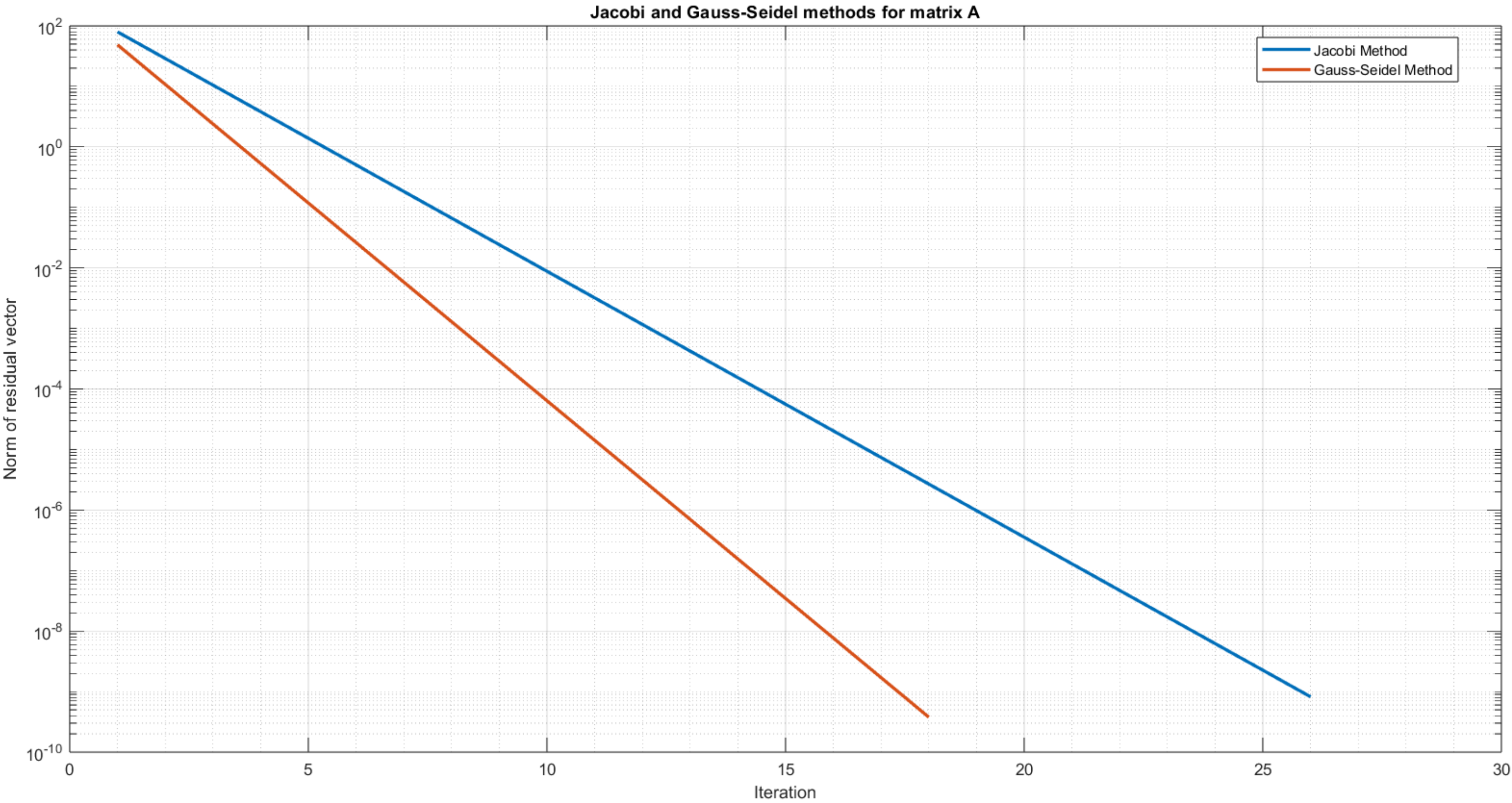
2. Analiza

Zadanie B

W zadaniu B dla macierzy opisanej we wprowadzeniu sprawozdania rozwiązywane jest równanie macierzowe:

$$\mathbf{Ax = b}$$

Do rozwiązywania tego równania wykorzystywane są metody iteracyjne Jacobiego i Gaussa-Seidla. Norma wektora residuum dla poszczególnych metod zmieniała się z każdą iteracją w przedstawiony sposób:



Jak widać na wykresie obie metody są zbieżne. Metoda Gaussa-Seidla potrzebowała mniej iteracji aby otrzymać wynik wystarczająco dokładny dla przeprowadzanych testów.

```
##################################
########## TASK A and B ##########
##################################

Jacobi method
        Iterations: 26
        Residual norm: 8.19771e-10
        Duration: 0.181439s

Gauss-Seidel method
        Iterations: 18
        Residual norm: 3.77495e-10
        Duration: 0.127969s
```

Metoda Gaussa-Seidla również zajęła mniej czasu, co zostało przedstawione na załączonym obrazie.

Zadanie C

W zadaniu C równanie macierzowe **Ax = b** korzystało z innych wartości niż w przypadku poprzedniego zadania. Wymiary macierzy A oraz wektora b pozostały takie same. Wartości w wektorze b, również pozostały niezmienne, ale wartości którymi uzupełniono macierz A zostały zmodyfikowane. Zamiast wartości 11 na głównej diagonali umieszczono 3.

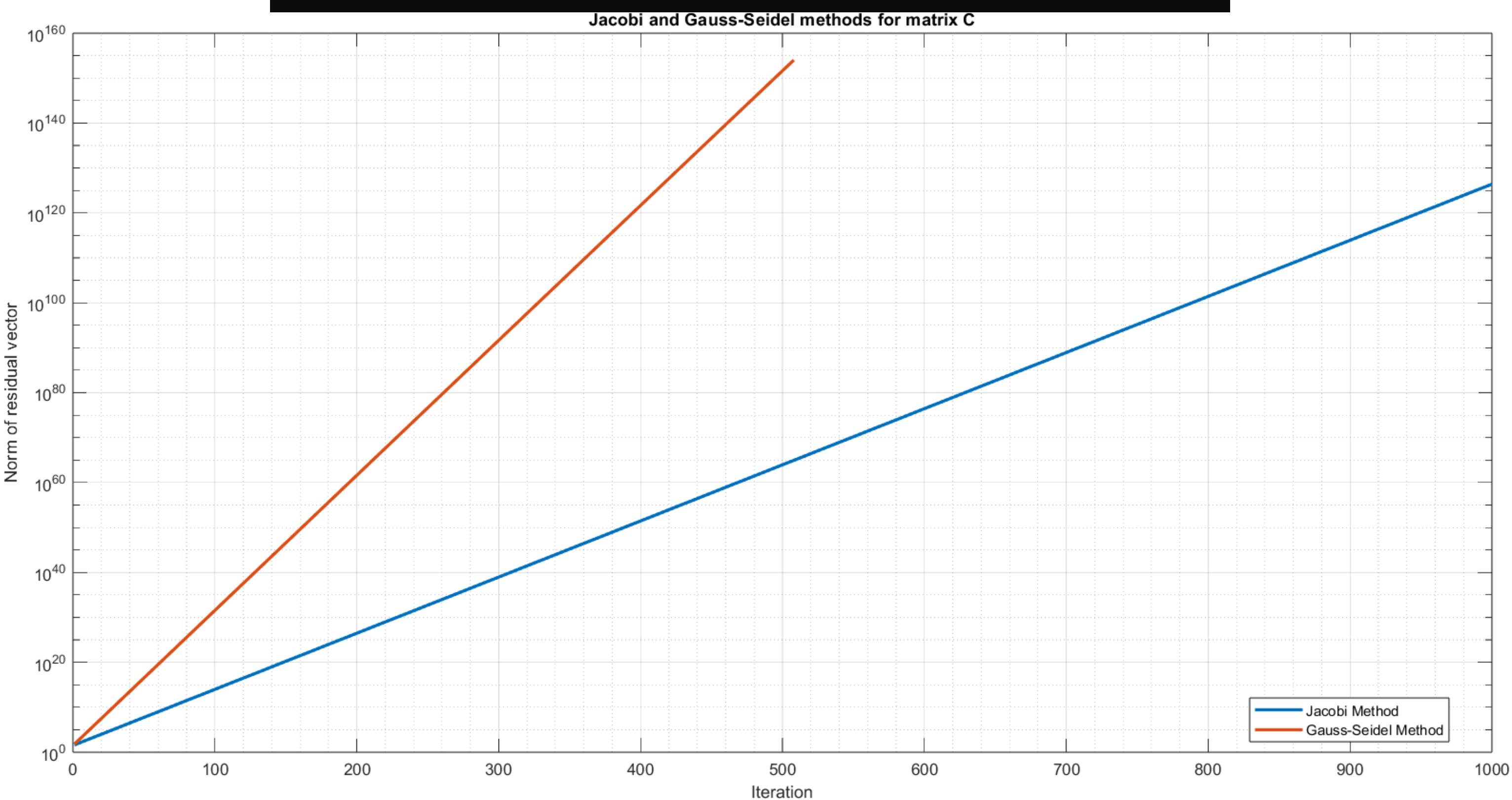
Wygląd macierzy 6x6 z danymi wartościami:

3	-1	-1	0	0	0
-1	3	-1	-1	0	0
-1	-1	3	-1	-1	0
0	-1	-1	3	-1	-1
0	0	-1	-1	3	-1
0	0	0	-1	-1	3

Próba znalezienia przybliżonego rozwiązania przy użyciu metod Jacobiego i Gaussa-Seidla kończy się niepowodzeniem. Dla macierzy A oraz wektora b w opisanych postaciach metody nie są zbieżne.

```
Jacobi method
        Iterations: 1000
        Residual norm: 2.55484e+126
        Duration: 5.84282s

Gauss-Seidel method
        Iterations: 1000
        Residual norm: inf
        Duration: 4.89132s
```



Obie metody wykonują max iteracji równy 1000 ponieważ błąd z każdą iteracją się zwiększa zatem nie jest osiągana pożądana precyzja rozwiązania. Tak jak w zadaniu B metoda Gaussa-Seidla się szybciej zbiegał, to w przypadku C jego błąd rośnie szybciej niż w przypadku metody Jacobiego i przy około 510 iteracji błąd osiąga tak duże wartości że przestaje być poprawnie obsługiwany przez język C++.

Zadanie D

W zadaniu D wymagana jest implementacja i zastosowanie metody faktoryzacji LU na równaniu opisanym w zadaniu C. Wyniki dla tej metody prezentują się w następujący sposób:

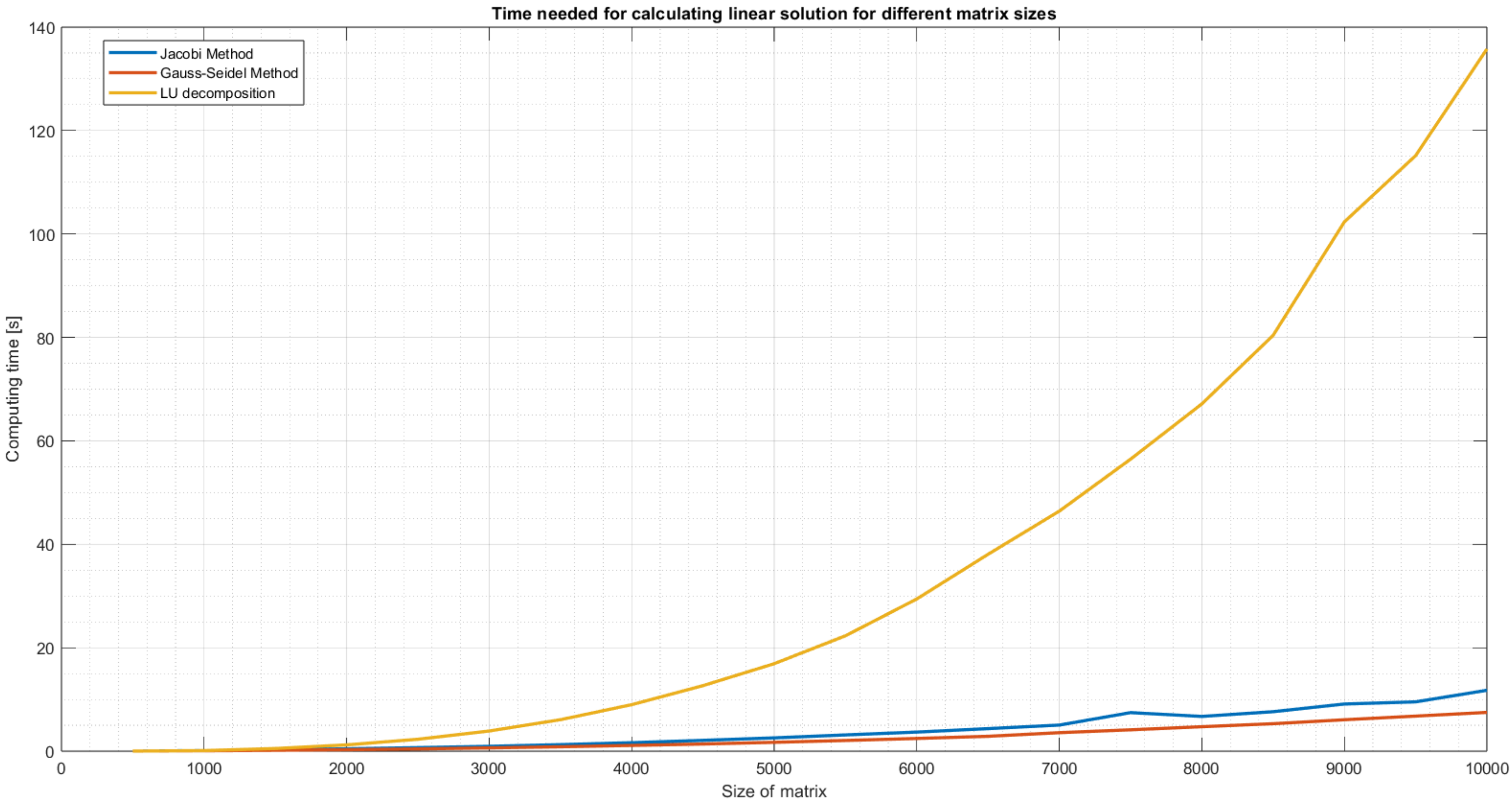
```
LU decomposition
        Residual norm: 1.68787e-13
        Duration: 0.100943s
```

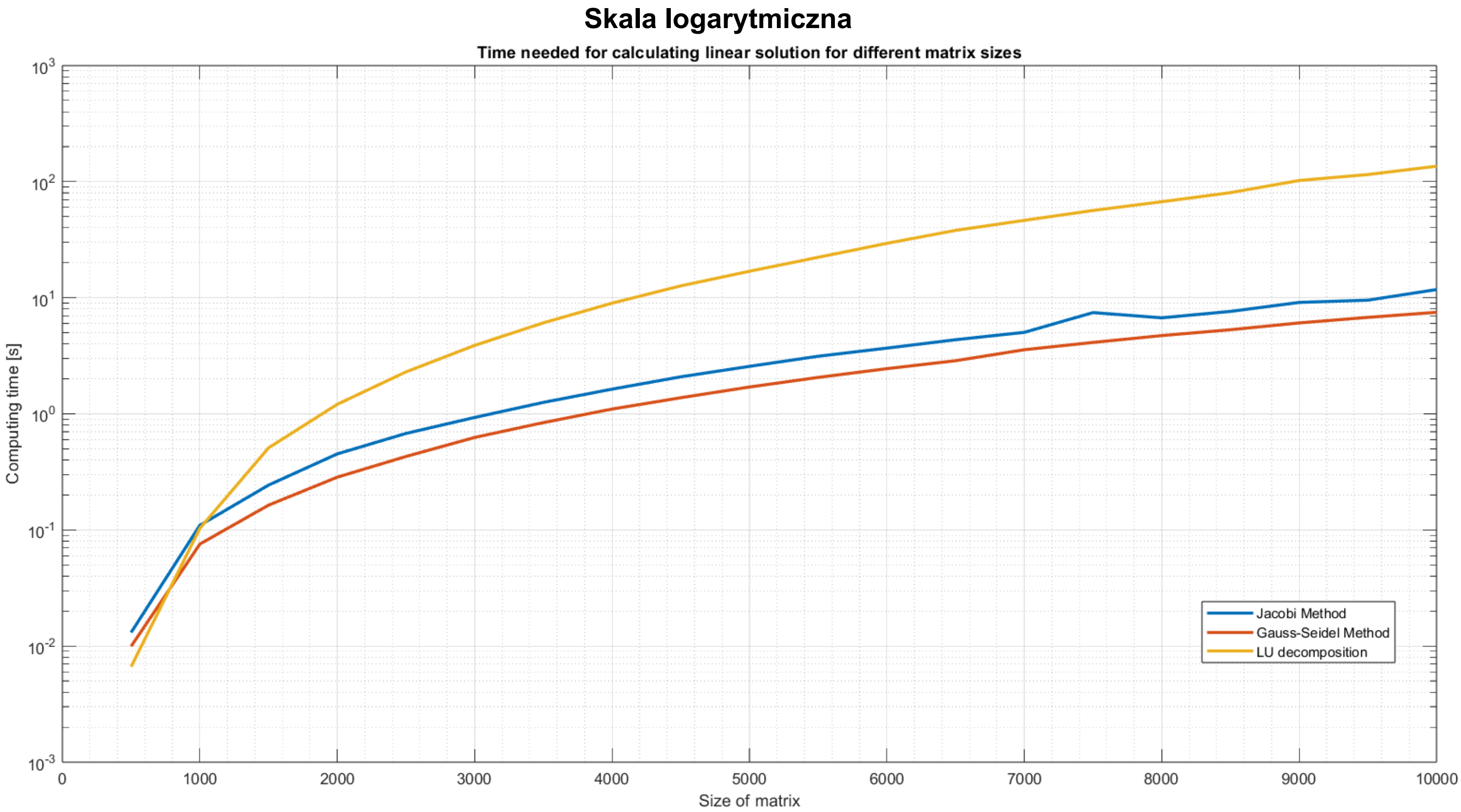
Metoda osiągnęła precyzję rzędu 10^{-13} co jest wynikiem o precyzji wyższej niż określone w zadaniach 10^{9}. Metoda poprawnie odnalazła rozwiązania równania macierzowego. Co pokazuje, że istnieją sytuacje w których przedstawione metody iteracyjne zawodzą, a metoda faktoryzacji LU odpowiednio odnajduje rozwiązania.

Zadanie E

Celem zadania E jest przeanalizowanie prędkości działania poszczególnych metod dla różnych rodzajów macierzy. W realizacji zadania obliczane są rozwiązania 20 równań macierzowych dla rozmiaru macierzy N={500, 1000, 1500, 2000, 2500, … , 9500, 10000}. Macierz A oraz wektor b są wypełnione wartościami zgodnie z opisem ze wstępu. Równania rozwiązywane są przy użyciu trzech opracowanych metod: Jacobiego, Gaussa-Seidla oraz faktoryzacji LU. Wyniki czasowe prezentują się w następujący sposób:

Skala liniowa





Wraz ze wzrostem rozmiaru macierzy czas obliczania rozwiązań równania macierzowego wydłuża się. Wzrost czasu wykonywania obliczeń w zależności od rozmiaru macierzy jest największy dla metody faktoryzacji LU, ale dla macierzy rozmiaru 500x500 ta metoda uzyskała najniższy czas wykonywania.

Niezależnie od zastosowanej metody poprawnie odnajdywały rozwiązania równania macierzowego.

```
#######  Results for N= 1000  #######
Jacobi method
        Iterations: 26
        Residual norm: 8.29529e-10
        Duration: 0.109s

Gauss-Seidel method
        Iterations: 18
        Residual norm: 3.82023e-10
        Duration: 0.076s

LU decomposition
        Residual norm: 2.65744e-15
        Duration: 0.103s
```

```
#######  Results for N= 5000  #######
Jacobi method
        Iterations: 27
        Residual norm: 6.78995e-10
        Duration: 2.565s

Gauss-Seidel method
        Iterations: 18
        Residual norm: 6.62447e-10
        Duration: 1.703s

LU decomposition
        Residual norm: 6.02612e-15
        Duration: 16.91s
```

```
#######  Results for N= 10000  #######
Jacobi method
        Iterations: 27
        Residual norm: 9.61057e-10
        Duration: 11.76s

Gauss-Seidel method
        Iterations: 19
        Residual norm: 2.71345e-10
        Duration: 7.490s

LU decomposition
        Residual norm: 8.46048e-15
        Duration: 135.8s
```

Tabela czasów

Rozmiar macierzy	Jacobi [s]	Gauss-Seidel [s]	Faktoryzacja LU [s]
500x500	0.013	0.010	0.007
1000x1000	0.109	0.076	0.103
1500x1500	0.243	0.163	0.509
2000x2000	0.451	0.285	1.208
2500x2500	0.677	0.429	2.297
3000x3000	0.930	0.626	3.887
3500x3500	1.254	0.840	6.079
4000x4000	1.631	1.099	8.984
4500x4500	2.084	1.377	12.65
5000x5000	2.565	1.703	16.91
5500x5500	3.134	2.060	22.31
6000x6000	3.683	2.450	29.42
6500x6500	4.349	2.868	38.06
7000x7000	5.034	3.566	46.41
7500x7500	7.448	4.121	56.46
8000x8000	6.713	4.720	67.15
8500x8500	7.627	5.302	80.40
9000x9000	9.111	6.068	102.3
9500x9500	9.528	6.775	115.2
10000x10000	11.76	7.490	135.8

3. Podsumowanie

Podczas realizacji projektu udało się stwierdzić, że metody iteracyjne Jacobiego i Gaussa-Seidla mogą nie znaleźć rozwiązań równania macierzowego. Udało się to metodzie faktoryzacji LU, ale nie oznacza to że jest ona niezawodnym sposobem rozwiązywania tego typu równań.

Jeżeli metody analizowane metody iteracyjne są w stanie znaleźć miejsce zerowe, to nadają się one do rozwiązywania tego typu jeżeli zależy nam na precyzji np. takiej jak w zadaniu czyli 10^{-9}. Jeżeli zależy nam na większej precyzji to okażą się one wolniejsze od metody faktoryzacji LU, ponieważ muszą one wykonać dużo więcej iteracji, zatem jeżeli zależy nam na większej precyzji wyniku to lepsza może okazać się metoda faktoryzacji LU.

```
#######  Results for N= 1500  #######
Jacobi method
        Iterations: 1000
        Residual norm: 2.42855e-15
        Duration: 9.14818s

Gauss-Seidel method
        Iterations: 1000
        Residual norm: 2.42855e-15
        Duration: 9.86897s

LU decomposition
        Residual norm: 3.2871e-15
        Duration: 0.58993s
```

```
#######  Results for N= 2500  #######
Jacobi method
        Iterations: 1000
        Residual norm: 3.12302e-15
        Duration: 29.63s

Gauss-Seidel method
        Iterations: 1000
        Residual norm: 3.12302e-15
        Duration: 29.255s

LU decomposition
        Residual norm: 4.18399e-15
        Duration: 3.26181s
```

Podczas realizacji zadania związanego z czasem wykonywania poszczególnych metod zauważyłem, że w metodzie faktoryzacji LU najwięcej czasu zajmowało wyznaczenie macierzy L oraz U, a metody podstawienia w przód i wstecz były niewielkim odsetkiem całego czasu. Można z tego wywnioskować, że metoda faktoryzacji LU dobrze się sprawdzi jeżeli chcemy rozwiązać wiele równań gdzie tylko wektor b jest zmienny, a macierz A, czyli macierze L i U również, pozostaje niezmienna.