Sprawozdanie

Układy równań liniowych

Jan Krupiniewicz

s193320

Wstęp

Celem projektu było zaimplementowanie oraz analiza dwóch metod iteracyjnych (Jacobiego i Gaussa-Seidla) oraz jednej metody bezpośredniej (faktoryzacja LU) do rozwiązywania układów równań liniowych. Testy przeprowadzono na układach równań wynikających z dyskretyzacji równań różniczkowych, powszechnie stosowanych w wielu dziedzinach, takich jak elektronika, mechanika czy termodynamika. Badano również wpływ różnych parametrów na efektywność metod.

Konstrukcja układu równań [Z.A]

Układ równań liniowych przyjął postać $Ax = b$, gdzie A jest macierzą systemową, b wektorem pobudzenia, a x wektorem rozwiązań. Ponda to jako kryterium stopu dla algorytmów iteracyjnych zostało wykorzystane residuum. Obliczając normę euklidesową residuum, możliwe było określenie błędu w każdej iteracji algorytmu. Jako warunek stopu została przyjęta norma 10^{-6}.

Macierz A została zdefiniowana jako macierz pasmowa o rozmiarze $N \times N$ dla wartości $N = 18$. Macierz zawierająca pięć diagonalnych pasm określonych przez wartości $a_2 = a_3 = -1$ oraz $a_1 = 8$. Poniżej przedstawiona jest uzyskana macierz:

```
[[8, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [-1, 8, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [-1, -1, 8, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, -1, -1, 8, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, -1, -1, 8, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, -1, -1, 8, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, -1, -1, 8, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, -1, -1, 8, -1, -1, 0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, -1, -1, 8, -1, -1, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, -1, -1, 8, -1, -1, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 8, -1, -1, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 8, -1, -1, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 8, -1, -1, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 8, -1, -1, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 8, -1, -1, 0],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 8, -1, -1],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 8, -1],
 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 8]]
```

Uzyskana macierz o wymiarach 18×18 *.*

Wektor b został wygenerowany zgodnie z instrukcjami, a N analogicznie jak dla macierzy A zostało określone na podstawie numeru indeksu. Ostateczna postać wektora zaprezentowana jest poniżej.

```
[0j,
 (-0.7568024953079282-0j),
 (0.9893582466233818-0j),
 (-0.5365729180004349+0j),
 (-0.2879033166650653-0j),
 (0.9129452507276277+0j),
 (-0.9055783620066238+0j),
 (0.27090578830786904-0j),
 (0.5514266812416906+0j),
 (-0.9917788534431158-0j),
 (0.7451131604793488-0j),
 (0.017701925105413577+0j),
 (-0.7682546613236668-0j),
 (0.9866275920404853-0j),
 (-0.5215510020869119+0j),
 (-0.3048106211022167-0j),
 (0.9200260381967907+0j),
 (-0.8979276806892913+0j)]
```

Uzyskany wektor wektorem o długości N *, którego* n *–ty element ma wartość* $\sin\big(n(f\ +\ 1)\big)$*.*

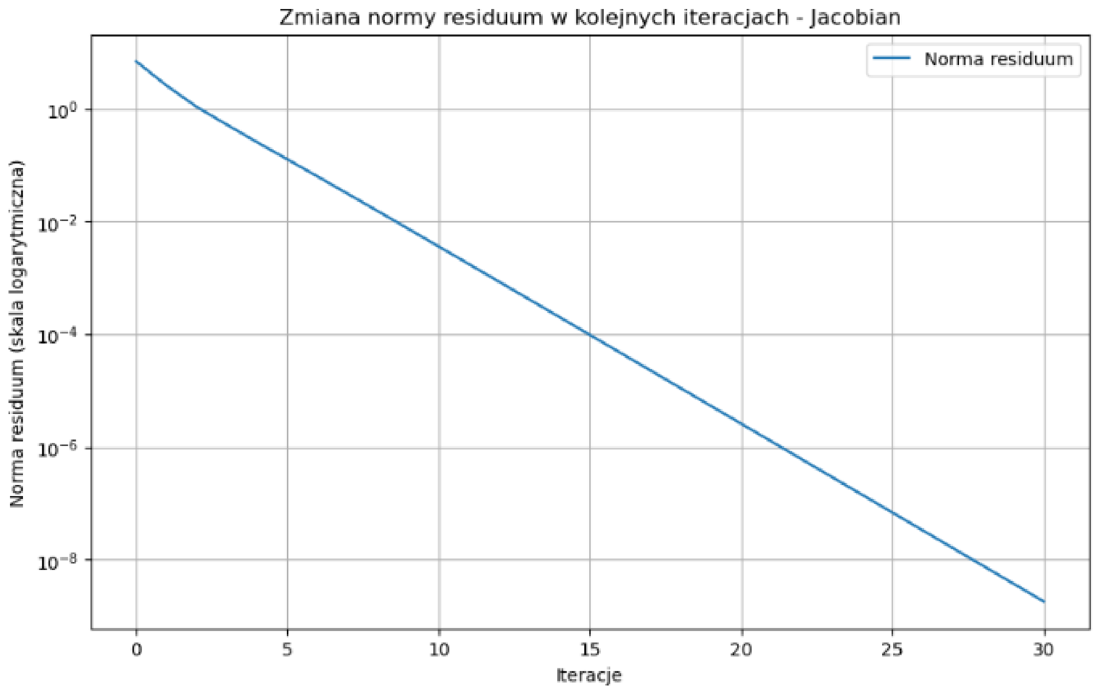
Implementacja metod iteracyjnych [Z. B]

W przypadku metody Jacobiego wykonano 31 iteracji, zaś dla metody Gaussa-Seidla zaledwie 19. Porównując czas trwania obu algorytmów można dojść do tych samych wniosków, tutaj również metoda Gaussa osiąga lepsze rezultaty. Wynika to z faktu, że Gauss-Seidel uwzględnia już obliczone nowe wartości w trakcie iteracji, co przyspiesza proces zbliżania się do rozwiązania.

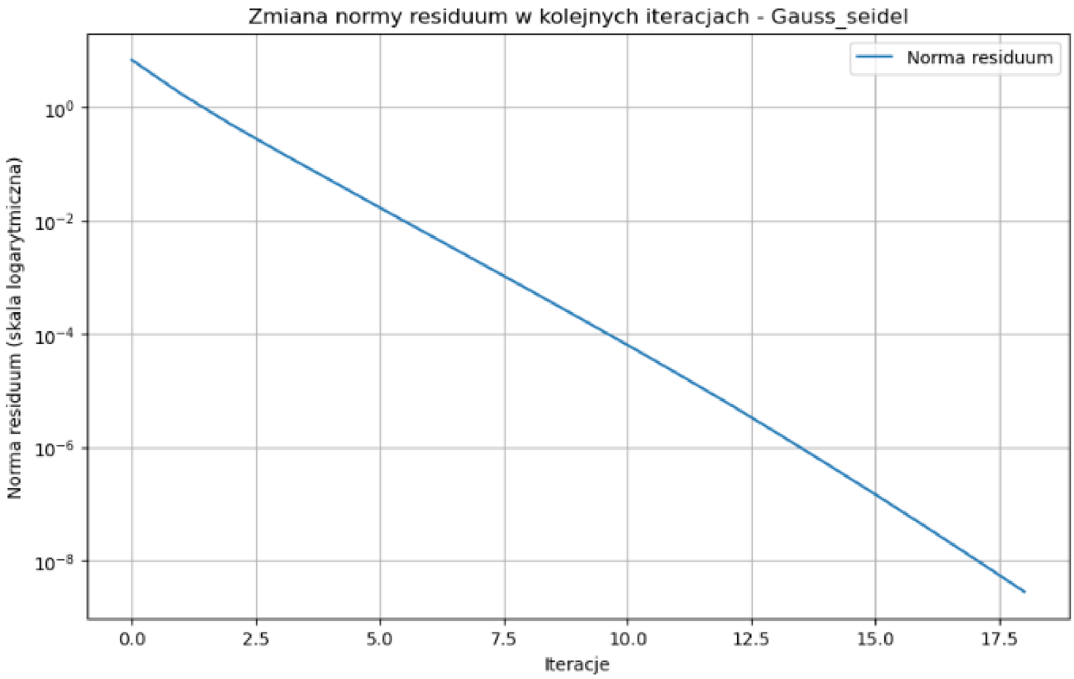
```
Jacobian Iterations: 31
Jacobian Final Residuum:  1.831518159356449e-09
Jacobian Time taken:  0.002032756805419922
```

```
Gauss-Seidel Iterations: 19
Gauss-Seidel Final Residuum:  2.8287718972785925e-09
Gauss-Seidel Time taken:  0.0009915828704833984
```

Analizując zmianę normy residuum w kolejnych iteracjach, zauważono, że obie metody wykazywały szybkie zmniejszanie się błędu, a następnie spowolnienie tego procesu w miarę zbliżania się do rozwiązania. Na wykresie z ośmieleniem skali logarytmicznej na osi Y można zaobserwować, że początkowo norma residuum maleje gwałtownie, a następnie zmiany stają się mniej znaczące.



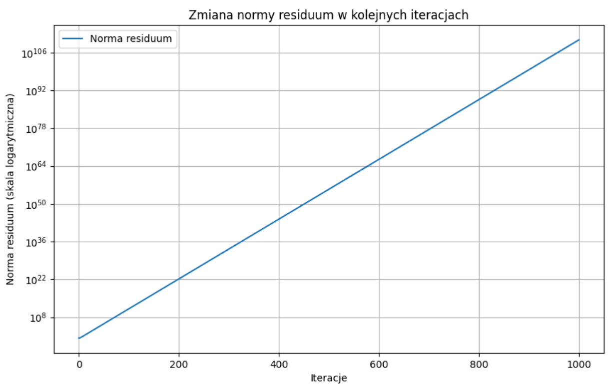
Jak widać na powyższym wykresie norma residuum maleje z kolejnymi iteracjami aż do osiągnięcia oczekiwanej wartości bliskiej zeru.

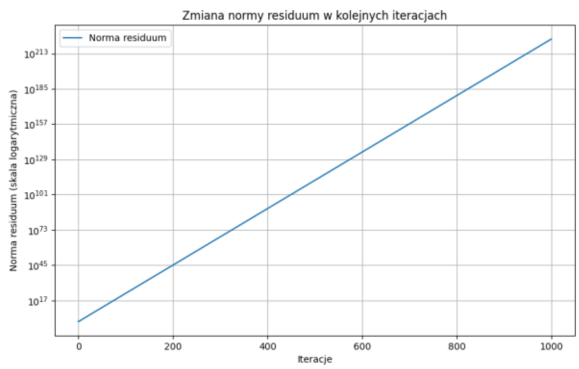


Dla metody Gaussa-Seidla obserwujemy szybszy spadek niż w przypadku metody Jacobiego. Metoda ta osiąga nieznacznie szybciej oczekiwaną dokładność.

Analiza braku zbieżności metod iteracyjnych [Z. C]

Po sporządzeniu wykresów metod iteracyjnych dla nowych parametrów macierzy A , $a_1 = 3$ i $a_2 = a_3 = -1$, stwierdziłem że obie metody nie wykazują zbieżności.





Na powyższych diagramach widać że zarówno dla metody Jacobiego jak i Gaussa-Seidla osiągamy podobny rezultat – brak uzyskania prawidłowego rozwiązania.

Przyczyny braku zbieżności mogą wynikać z charakterystyki nowej macierzy A. . Mogą występować takie czynniki jak dominacja jednej z diagonalnych pasm macierzy, co może prowadzić do niestabilności metod iteracyjnych.

Wniosek z tej analizy jest taki, że zastosowanie metod iteracyjnych do rozwiązywania układów równań liniowych wymaga uwagi przy doborze parametrów macierzy. Nawet drobne zmiany w wartościach elementów mogą mieć znaczący wpływ na zbieżność tych metod. W przypadku braku zbieżności metod iteracyjnych, konieczne może być zastosowanie innych metod, takich jak metoda faktoryzacji LU, którą opiszę poniżej.

Metoda faktoryzacji LU [Z. D]

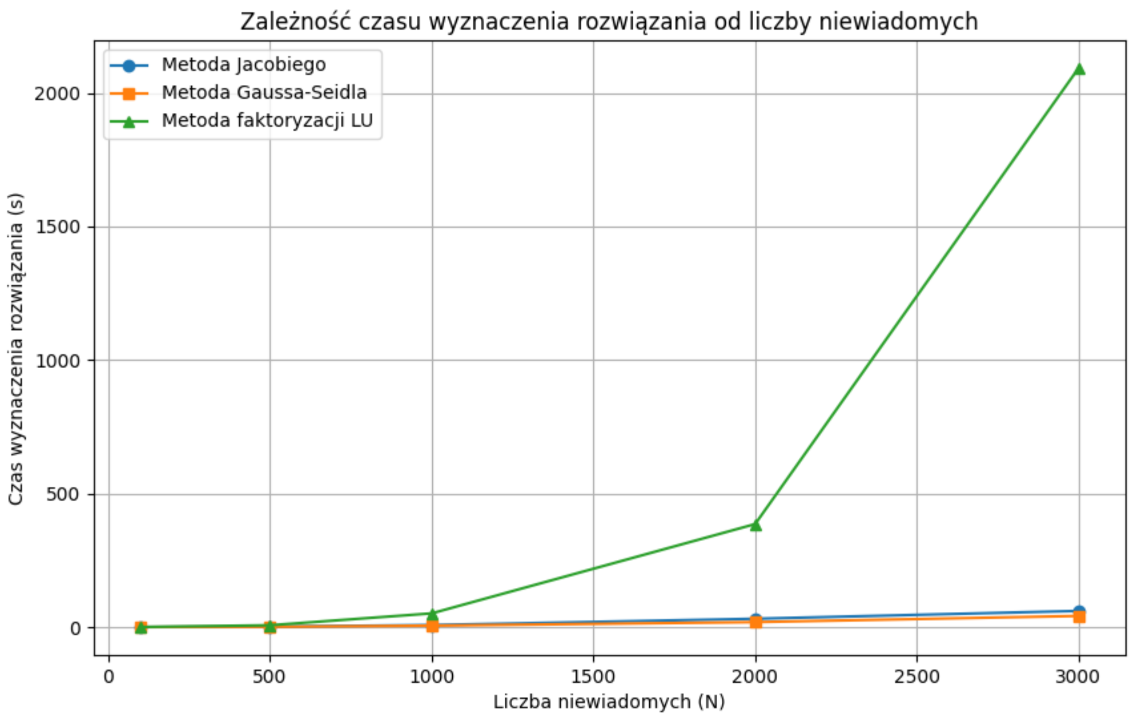
Po zastosowaniu metody faktoryzacji LU do rozwiązania układu równań liniowych opisanego w zadaniu C, uzyskano wartość normy residuum bliską zeru, co wskazuje na bardzo dobrą jakość uzyskanego rozwiązania.

Implementacja metody faktoryzacji LU pozwoliła na bezpośrednie rozwiązanie układu równań liniowych, omijając proces iteracyjny, który może być podatny na braki zbieżności dla problematycznych danych. W przypadku metody LU, uzyskane rozwiązanie jest bardzo dokładne, co potwierdza wysoką efektywność tej metody dla dowolnych parametrów macierzy.

```
Final LU Residuum:  9.992007221626409e-16
```

Przedstawiona powyżej ostateczna wartości residuum jest bardzo bliska zeru – sugeruje to wysoką dokładność metody bezpośredniej.

Wyznaczanie rozwiązań dla różnych wartości N [Z. E]



Metoda faktoryzacji LU zajmuje najwięcej czasu wykonania dla każdej z badanych wartości N. Jest to spodziewane, ponieważ faktoryzacja LU jest metodą bezpośrednią, która wymaga większej ilości obliczeń w porównaniu do metod iteracyjnych.

Metoda Jacobiego zajmuje mniej czasu niż faktoryzacja LU, ale więcej niż metoda Gaussa-Seidla. Jest to zgodne z oczekiwaniami, ponieważ metoda Jacobiego jest mniej efektywna niż metoda Gaussa-Seidla, co wynika z jej bardziej klasycznej natury.

Metoda Gaussa-Seidla zajmuje najmniej czasu wykonania dla każdej z badanych wartości NN. Jest to efekt tego, że metoda Gaussa-Seidla wykorzystuje informacje z aktualnych iteracji, co może przyspieszyć proces zbieżności w porównaniu do metody Jacobiego.

Wnioskiem z analizy wykresu jest to, że w zależności od wymagań czasowych i dokładności rozwiązania należy wybrać odpowiednią metodę do rozwiązania układów równań liniowych. Metoda faktoryzacji LU jest najbardziej kosztowna obliczeniowo, ale może być preferowana, jeśli wymagane jest bardzo dokładne rozwiązanie. Metody iteracyjne, takie jak Jacobiego i Gaussa-Seidla, mogą być bardziej efektywne czasowo, zwłaszcza dla większych macierzy, ale mogą wymagać większej liczby iteracji w celu uzyskania wystarczającej dokładności.

Podsumowanie

Wykonanie projektu pozwoliło na lepsze zrozumienie działania oraz porównanie efektywności różnych metod rozwiązywania układów równań liniowych. Metody iteracyjne okazały się być bardziej skuteczne dla pewnych parametrów macierzy, jednakże metoda faktoryzacji LU nadal jest użyteczna w przypadku mniejszych rozmiarów problemów. Ważne jest również uwzględnienie czasu wykonania, który może być kluczowy w praktycznych zastosowaniach.