Metody rozwiązywania ukłdów równań liniowych

Iwo Czartowski 193066 gr.4 Informatyka sem. 4

26 kwietnia 2024

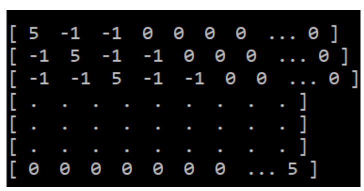
1 Wstęp

Celem niniejszego projektu była implementacja oraz analiza działania metod iteracyjnych Gauss - Seidel i Jacobi oraz metody bezpośredniej faktoryzacji LU. Postanowiłem zrealizować omawiane metody w języku C++ wykorzystującego jego szybkość działania, a wykresy sporządziłem przy użyciu języka Python oraz biblioteki matplotlib.

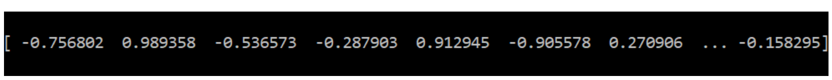
2 Analiza zadań

2.1 Zadanie A

Zważając na mój indeks 193066 otrzymujemy wartości a1 = 5, a2 = -1, a3 = -1 tworzące macierz wstęgową współczynników A o rozmiarze 966 na 966 oraz wektor wyrazów wolnych b o rozmiarze 966 którego n-ty elementy ma wartość $\sin(n * (3 + 1))$.


```
[ 5  -1  -1  0  0  0  0  ... 0 ]
[ -1  5  -1  -1  0  0  0  ... 0 ]
[ -1  -1  5  -1  -1  0  0  ... 0 ]
[ .  .  .  .  .  .  .  .  .  . ]
[ .  .  .  .  .  .  .  .  .  . ]
[ .  .  .  .  .  .  .  .  .  . ]
[ 0  0  0  0  0  0  0  0  ... 5 ]
```

Macierz A


```
[ -0.756802  0.989358  -0.536573  -0.287903  0.912945  -0.905578  0.270906  ... -0.158295]
```

Wektor b

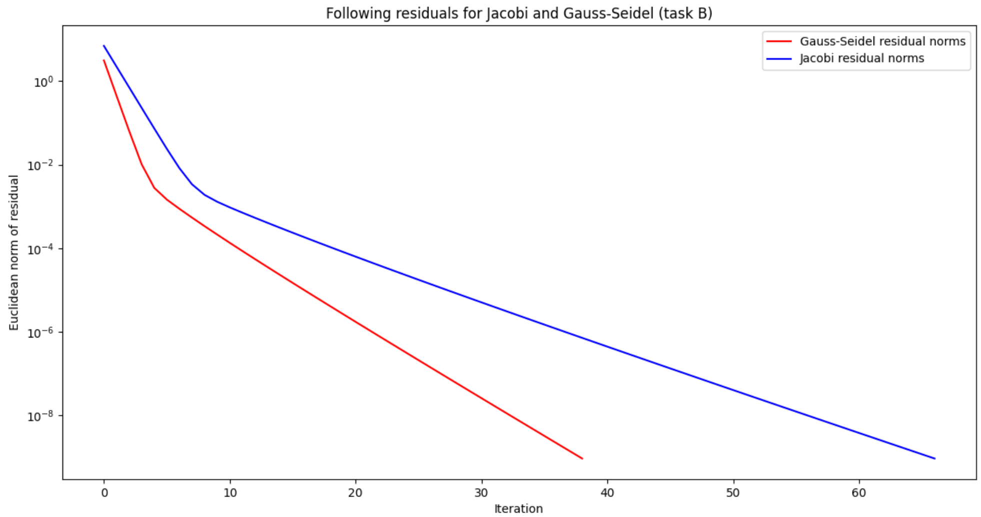
2.2 Zadanie B

Dla układu równań z zadania A wyniki przedstawiają się następująco:

Metoda Jacobi
czas - 0.14s
ilość iteracji - 67

Metoda Gauss-Seidel
czas - 0.092s
ilość iteracji - 39

Oba algorytmy iterował do momentu gdy norma z wektora residuum nie wyniosła zadowalającej wartości wynoszącej 10^{-9}.
Poniżej przedstawiłem jak zmieniała się norma z wektora residuum w kolejnych iteracjach:

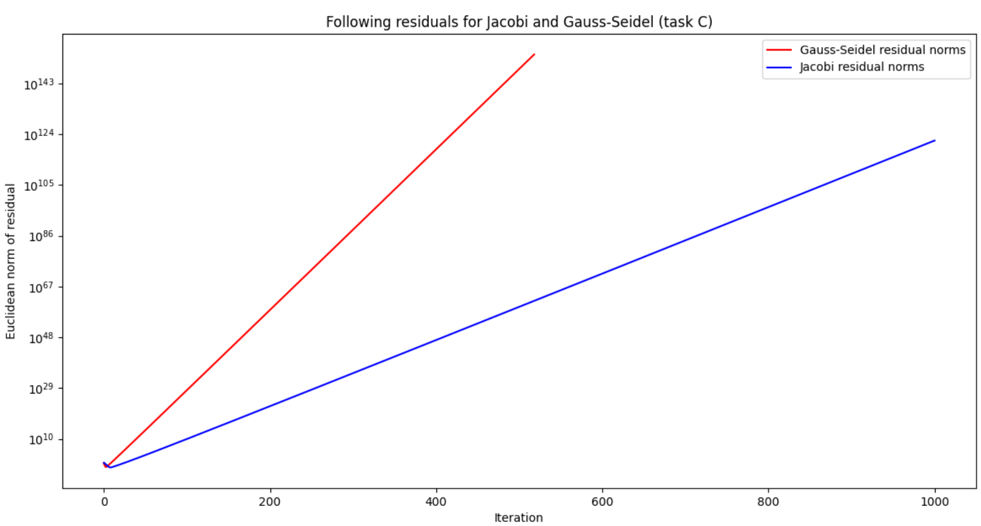


Normy z wektora residuum w kolejnych iteracjach

Można zauważyć, że metoda Gauss-Seidel potrzebowała mniej iteracji oraz około 1.5 razy mniej czasu by wykonać obliczenia.

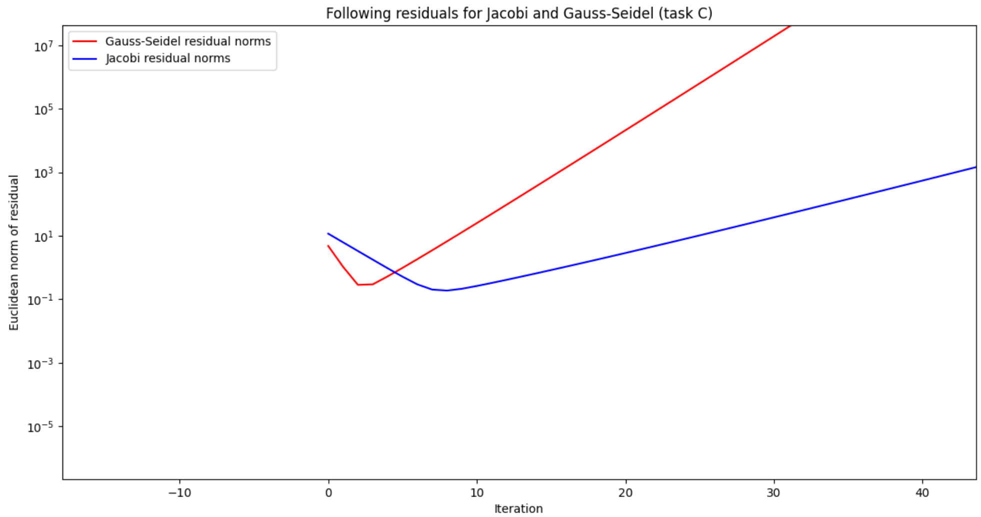
2.3 Zadanie C

W tym przypadku tworzymy układ równań z macierzą wstęgową współczynnuików A jednak z wartościami a1 = 3, a2 = -1, a3 = -1 natomiast wektor b oraz rozmiary pozostają bez zmian. Okazuje się, że dla takich wartości macierzy A obie badane metody nie zbiegają się, gdyż macierz A nie jest diagonalnie dominująca.
Normy z wektorów residuum w następujących iteracjach prezentują się następująco:



Rozbiegające się normy z wektora residuum w kolejnych iteracjach

Możemy zauważyć, że wartości normy residuum zaczynają nie pożądanie rosnąć nie osiągając nawet wartości 10^{-1} w okoliach 5 iteracji.



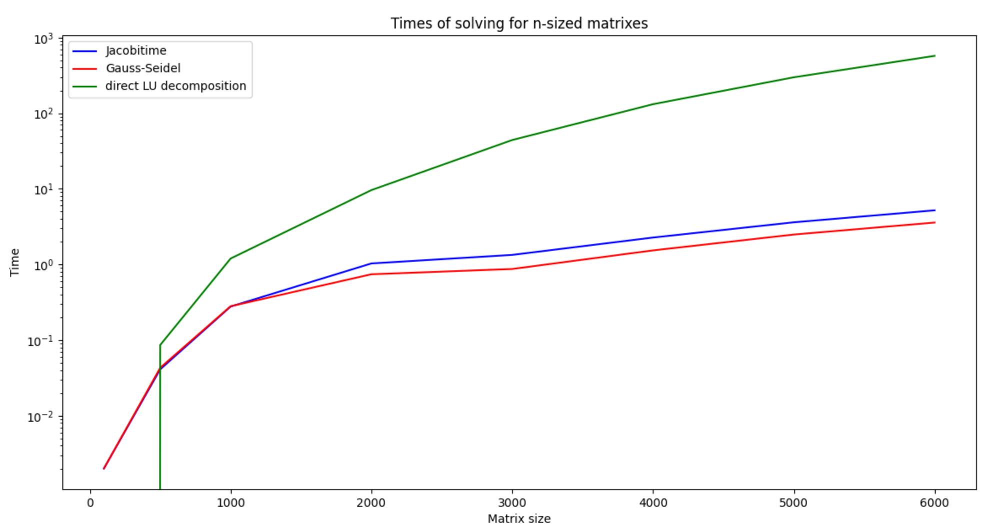
Moment rozbiegu wartości

2.4 Zadanie D

Przy użyciu metody bezpośredniego rozwiązywania układów równań liniowych faktoryzacją LU dla układu z zadania C norma wektora residuum wyniosła 3.67466e-13 co mówi nam o wysokie dokładności tego algorytmu. Omawiana metoda okazuje się w takim razie, być rozwiązaniem problemu rozwiązywania równań liniowych o macierzy współczynników nie będącej diagonalnie dominującej.

2.5 Zadanie E

Poniżej przedstawiony został wykres zależności czasu trwania od ilości niewiadomych w układzie poszczególnych algorytmów. Postanowiłem przedstawić go w skali logarytmicznej aby lepiej zoobrazować jego przebieg.



Wykres zależności czasu trwania od ilości niewiadomych w układzie

Jak można przypuszczać dla każdej z metod czas trwania rośnie wraz z liczbą niewiadomych. Metoda Gauss-Seidl jest szybsza od Jacobi jednak oby dwa te algorytmy są rzędy bardziej optymalne czasowo niż metoda faktoryzacji LU.

2.6 Zadanie F

Wnioskując z wcześniejszej analizy możemy zcharakteryzować metody iteracyjne jako znacznie szybsze niż bezpośrednie, a metode Gauss-Seidel jak najszybszą z omawianych. Okazuje się jednak iż algorytm iteracyjne nie nadają sie do wszystkich układów równań i właśnie w tak owych pomocna przydaje się metoda faktoryzacja LU cechująca się niezawodnością oraz dokładnością.
Podsumowując, jeżeli macierz jest sproych rozmiarów i spełnia warunki zbieżności metod iteracyjnych, powinnyśmy wybrać jako narzędzie metode Gauss-Seidel bądź Jacobi, a w przeciwnym wypadku pozostaje nam bezpośrednia faktoryzacja LU.