

Autor: Marcin Sitko

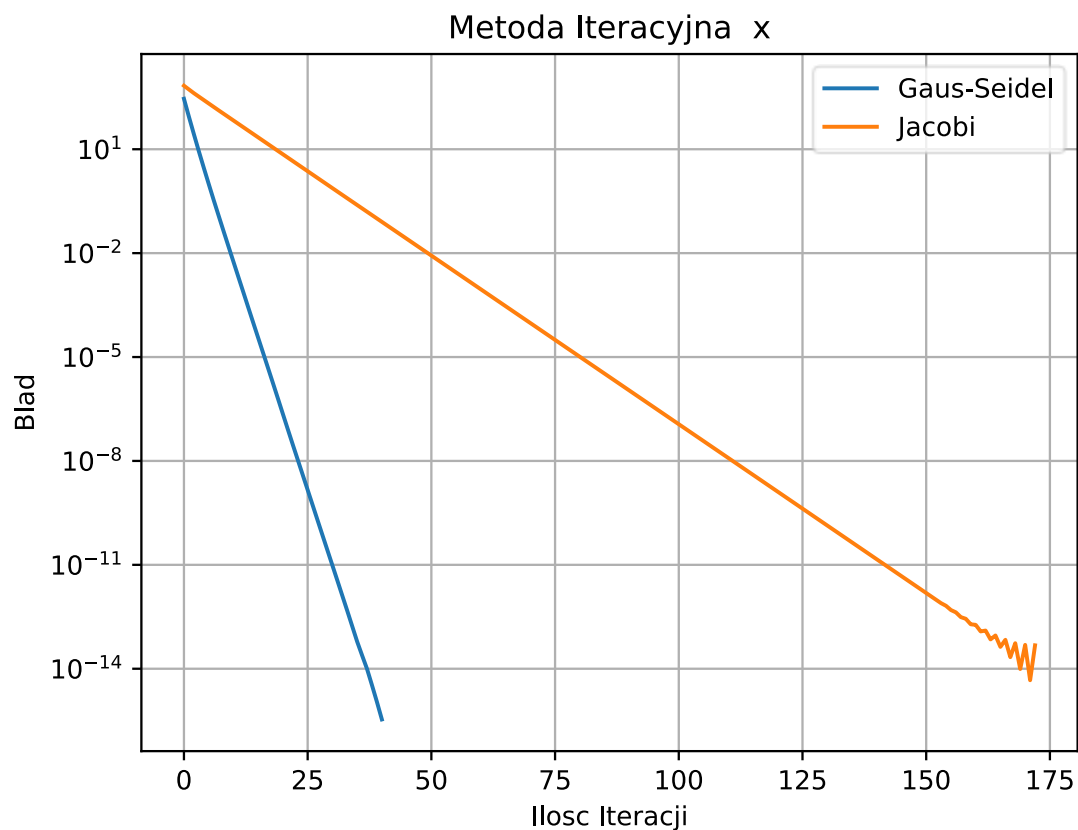
Sprawozdanie – NUM5

W zadaniu należy rozwiązać zadane równanie macierzowe dla $N = 100$ za pomocą metody *Jacobiego* oraz *Gausa – Seidela*

$$\begin{pmatrix} 3 & 1 & 0.2 & & & \\ 1 & 3 & 1 & 0.2 & & \\ 0.2 & 1 & 3 & 1 & 0.2 & \\ \dots & \dots & \dots & \dots & \dots & \dots \\ & & & 0.2 & 1 & 3 & 1 \\ & & & 0.2 & 1 & 3 & \end{pmatrix} \mathbf{x} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ \dots \\ N-1 \\ N \end{pmatrix}$$

Rozwiązując to zadania wybrałem, iż ilość iteracji potrzebnych do uzyskania wyniku z dokładnością równą niemalże epsilonowi maszynowemu będzie równa 170, po takiej ilości iteracji zaczyna się „szum” czyli powstają błędy numeryczne

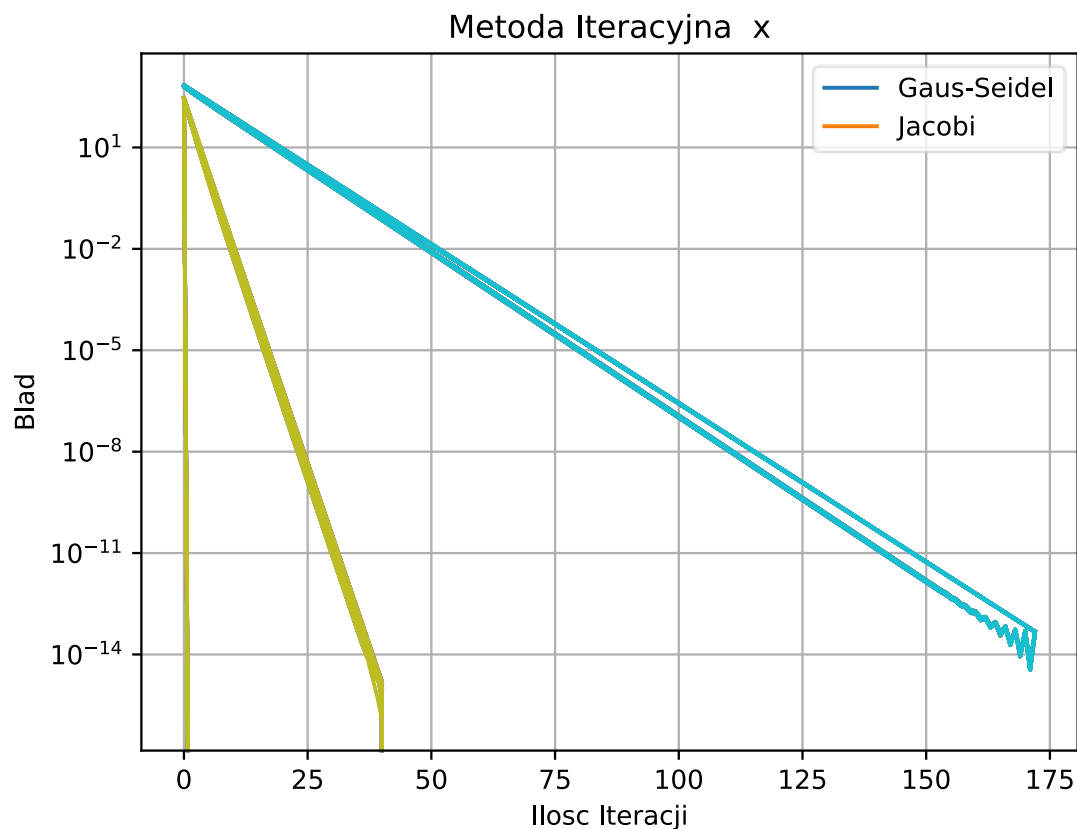
Wykres Przedstawiający Przybliżenie rozwiązania wraz ze wzrastającą ilością iteracji



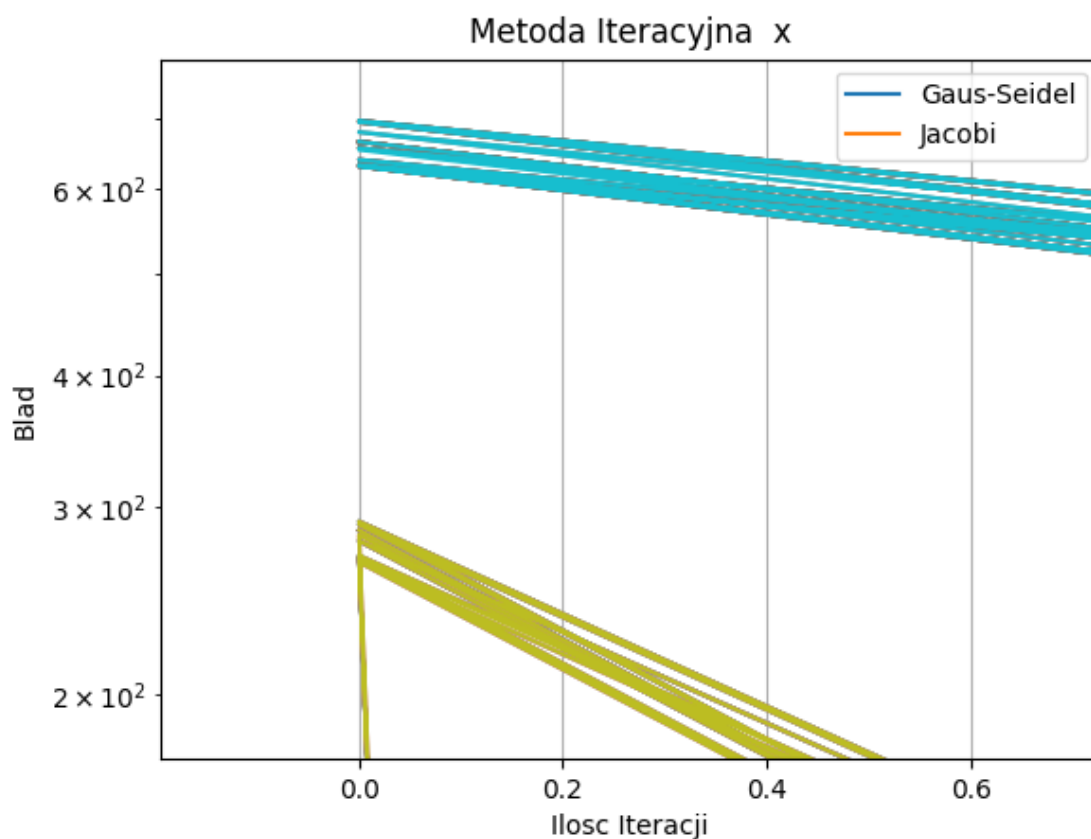
Patrząc na wykres można zauważyć, że metoda *Gaussa – Seidel* zbiega dużo szybciej i wręcz jest dokładniejsza w maksymalnym punkcie osiąga błąd rzędu $1e - 17$, Metoda Gaussa-Seidela daje nam wynik z dokładnością do epsilon maszynowego zaledwie po ok. 40 iteracjach natomiast metoda Jacobiego daje nam wynik trochę mniej dokładny po 170 iteracjach błąd w maksymalnym punkcie wynosi ok. $1e - 15$, Można powiedzieć że metoda Gaussa- Seidela jest znacznie szybsza, zbiega szybciej

Dodatkowo wybierając losowy punkt metody te dadzą ten sam, lub bardzo podobny wynik, można to zobaczyć wybierając *opcje numer 3* z mojego menu oraz podając dowolną liczbę punktów rozpoczęcia jedyna różnica jaka nastąpi będzie na samym początku wykresu następnie wykres będzie praktycznie identyczny

Przykładowy wykres dla 10 punktów startowych wybranych losowo



Po dosyć dużym przybliżeniu w miejscu rozpoczęcia wykresu widzimy następującą rzecz



Znaczna różnica istnieje tylko na początku wykresy następnie wykres zaczyna się „scalać” oznacza to że bez względu na wybranie punktu startowego metoda Jacobiego oraz Metoda Gaussa-Seidel będzie zbieżna, Oczywiście dalej Metoda Gaussa-Seidela będzie szybsza.

W moim Programie można także zobaczyć wyniki poszczególnych implementacji algorytmów wybierając opcję numer 1 z mojego menu.

```

[1.1726069204515493 | 0.375237973774515706 | 0.5548999253689075 | 0.740663834889241576 | 0.9268238956061977 | 1.111807426365375 | 1.2962972717646517 | 1.481482921365333 | 1.6666666889136743 | 1.8518519452948397 | 2.037837047783699 | 2.22222211483657 | 2.407847018368299 | 2.5925925923669672 | 2.7777777348249 | 2.9629629638191952 | 3.148148148135404 | 3.33333333327199 | 3.51851851819688 | 3.703783703763342 | 3.8888888888929 | 4.0784704784087 | 4.25925925925925 | 4.439999999999999 | 4.629629629629629 | 4.818481481481846 | 5.000000000000001 | 5.18518518519688 | 5.37037837037037 | 5.555555555555556 | 5.740784708740784 | 5.925925925925926 | 6.111111111111111 | 6.296296296296297 | 6.481481481481846 | 6.666666666666667 | 6.85185185185185 | 7.03783703787038 | 7.222222222222222 | 7.407847047840786 | 7.592592592592594 | 7.777777777777778 | 7.962962962962969 | 8.148148148148147 | 8.333333333333334 | 8.518518518518519 | 8.703783703783704 | 8.8888888888929 | 9.07847047840784 | 9.25925925925926 | 9.439999999999999 | 9.629629629629629 | 9.818481481481847 | 10.000000000000001 | 10.185185185185187 | 10.37837837837038 | 10.555555555555557 | 10.740784708740784 | 10.925925925925924 | 11.111111111111112 | 11.296296296296296 | 11.481481481481846 | 11.666666666666667 | 11.85185185185185 | 12.03783703787038 | 12.222222222222222 | 12.407847047840786 | 12.592592592592593 | 12.777777777777778 | 12.962962962962964 | 13.148148148148152 | 13.333333333333337 | 13.518518518518519 | 13.703783703783703 | 13.8888888888892 | 14.07847047840784 | 14.259259259259258 | 14.439999999999999 | 14.629629629629629 | 14.818481481481847 | 15.000000000000001 | 15.185185185185187 | 15.37837837037837 | 15.555555555555557 | 15.740784708740784 | 15.925925925925924 | 16.111111111111112 | 16.296296296296297 | 16.481481481481846 | 16.666666666666667 | 16.85185185185185 | 17.03783703787038 | 17.222222222222222 | 17.407847047840786 | 17.592592592592594 | 17.777777777777778 | 17.962962962962969 | 18.148148148148147 | 18.333333333333334 | 18.518518518518519 | 18.703783703783704 | 18.8888888888929 | 19.07847047840784 | 19.25925925925926 | 19.439999999999999 | 19.629629629629629 | 19.818481481481847 | 20.000000000000001 | 20.185185185185187 | 20.37837837837038 | 20.555555555555557 | 20.740784708740784 | 20.925925925925924 | 21.111111111111112 | 21.296296296296296 | 21.481481481481846 | 21.666666666666667 | 21.85185185185185 | 22.03783703787038 | 22.222222222222222 | 22.407847047840786 | 22.592592592592593 | 22.777777777777778 | 22.962962962962964 | 23.148148148148152 | 23.333333333333337 | 23.518518518518519 | 23.703783703783703 | 23.8888888888892 | 24.07847047840784 | 24.259259259259258 | 24.439999999999999 | 24.629629629629629 | 24.818481481481847 | 25.000000000000001 | 25.185185185185187 | 25.37837837037837 | 25.555555555555557 | 25.740784708740784 | 25.925925925925924 | 26.111111111111112 | 26.296296296296297 | 26.481481481481846 | 26.666666666666667 | 26.85185185185185 | 27.03783703787038 | 27.222222222222222 | 27.407847047840786 | 27.592592592592594 | 27.777777777777778 | 27.962962962962969 | 28.148148148148147 | 28.333333333333334 | 28.518518518518519 | 28.703783703783704 | 28.8888888888929 | 29.07847047840784 | 29.25925925925926 | 29.439999999999999 | 29.629629629629629 | 29.818481481481847 | 30.000000000000001 | 30.185185185185187 | 30.37837837837038 | 30.555555555555557 | 30.740784708740784 | 30.925925925925924 | 31.111111111111112 | 31.296296296296296 | 31.481481481481846 | 31.666666666666667 | 31.85185185185185 | 32.03783703787038 | 32.222222222222222 | 32.407847047840786 | 32.592592592592593 | 32.777777777777778 | 32.962962962962964 | 33.148148148148152 | 33.333333333333337 | 33.518518518518519 | 33.703783703783703 | 33.8888888888892 | 34.07847047840784 | 34.259259259259258 | 34.439999999999999 | 34.629629629629629 | 34.818481481481847 | 35.000000000000001 | 35.185185185185187 | 35.37837837037837 | 35.555555555555557 | 35.740784708740784 | 35.925925925925924 | 36.111111111111112 | 36.296296296296297 | 36.481481481481846 | 36.666666666666667 | 36.85185185185185 | 37.03783703787038 | 37.222222222222222 | 37.407847047840786 | 37.592592592592594 | 37.777777777777778 | 37.962962962962969 | 38.148148148148147 | 38.333333333333334 | 38.518518518518519 | 38.703783703783704 | 38.8
```

Widacze we wyniki się pokrywają w obu przypadkach implementacji algorytmów oraz z wynikami biblioteki numerycznej dokładane różnice pomiędzy wynikami można zobaczyć wybierając opcje numer 4 z mojego menu

[illegible]

Widać że różnice są bardzo małe , rzędu $1e-15$ tak więc można stwierdzić ze przybliżenie które generuje moja implementacja algorytmu jest rozsądne

Sugeruje to że algorytmy działają poprawnie