Sprawozdanie z zadania numerycznego 5

1. Instrukcja uruchomienia

- Aby uruchomić program należy użyć *python NUM5.py*

2. Cel ćwiczenia

Porównanie metod iteracyjnych Jacobiego i Gaussa-Seidela.

3. Opis ćwiczenia

1) Analiza problemu i wyprowadzenie wzorów
2) Zaimplementowanie algorytmów
3) Określenie warunku zbiegnięcia i warunku stopu
4) Obliczenie różnicy między poszczególnymi iteracjami a ostatnią iteracją
5) Przedstawienie na wykresie szybkości zbiegania oby metod

4. Wstęp teoretyczny

Zacznijmy od omówienia problemu przedstawionego w zadaniu. Rozwiązywanie układów równań za pomocą wcześniej stosowanych technik prowadziło do otrzymania dokładnych rozwiązań równań (pomijając błędu zaokrągleń). Metody te nazywaliśmy metodami dokładnymi.

Natomiast w metodach iteracyjnych rozwiązanie dokładne teoretycznie otrzymujemy po wykonaniu nieskończenie wielu kroków. Oczywiście nie wykonujemy naszych programów w nieskończoność, lecz oczekujemy że po skończonej ilości kroków zbliżymy się do wyniku dokładnego który będzie mieścił się w przyjętych przez nas granicach błędu zaokrąglenia.

Gdy mamy macierz rzadką by metody iteracyjne mogły być wydajne trzeba uwzględnić strukturę macierzy i unikać mnożenia przez zero. Nasza macierz A jest macierzą rzadką więc przy odpowiednim zastosowaniu metod iteracyjnych rozwiązanie otrzymamy w sposób efektywny.

$$A = \begin{pmatrix} 3 & 1 & 0.2 & & & & \\ 1 & 3 & 1 & \cdots & & 0 & \\ 0.2 & 1 & 3 & & & & \\ & \vdots & & \ddots & & \vdots & \\ & & & & 3 & 1 & 0.2 \\ & 0 & & \cdots & 1 & 3 & 1 \\ & & & & 0.2 & 1 & 3 \end{pmatrix}$$

Aby metoda Jacobiego była zbieżna nasza macierz A musi być silnie diagonalnie dominująca. To znaczy że element na głównej diagonali musi być większy od sumy pozostałych elementów w danym wierszu. Formalnie możemy ten warunek zapisać w postaci:

$$\forall_i \; |a_{ii}| > \sum_{k \neq i} |a_{ik}|$$

Uwzględniając strukturę naszej macierzy widzimy że w najgorszym przypadku suma elementów w wierszu jest równa 2.4 co jest mniejsze od elementu na diagonali. Metoda Jacobiego zbiegnie. Natomiast metoda Gaussa-Seidela jest zbieżna gdy mamy macierz symetryczną i dodatnio określoną co też zachodzi dla naszej macierzy.

Naszą macierz możemy zapisać jako: $A = A_1 + A_2$, a równanie przekształcić w następujących krokach:

$$Ax = b$$

$$(A_1 + A_2)x = b$$

$$A_1 x = -A_2 x + b$$

Otrzymujemy więc równanie iteracyjne które możemy zapisać jako:

$$A_1 x^{(n+1)} = -A_2 x^{(n)} + b$$

Przyjmując że macierz A można rozłożyć na L, D, U, gdzie L - macierz poddiagonalna, D – diagonala, U – macierz nad diagonalna, możemy wyprowadzić dwie metody iteracyjne.

Metoda Jacobiego: A = D + (L+U), gdzie D przyjmujemy że jest naszym A_1 natomiast L+U jest A_2. Otrzymujemy wzór:

$$Dx^{(n+1)} = -(L+U)x^{(n)} + b$$

który po rozpisaniu na składowe przyjmuje postać:

$$x_i^{(n+1)} = \frac{b_i - \sum_{k<i} a_{ik} x_k^{(n)} - \sum_{k>i} a_{ik} x_k^{(n)}}{a_{ii}}$$

Jeśli teraz uwzględnimy jak zbudowana jest nasza macierz to wzór dla naszego przypadku będzie miał postać:

$$x_i^{(n+1)} = \frac{b_i - x_{i-1}^{(n)} - 0.2x_{i-2}^{(n)} - x_{i+1}^{(n)} - 0.2x_{i+2}^{(n)}}{3}$$

Metoda Gaussa-Seidela: $A = (L+D) + U$ gdzie $L+D$ przyjmujemy że jest naszym A_1 natomiast U jest A_2. Otrzymujemy wzór:

$$(L+D)x^{(n+1)} = - Ux^{(n)} + b$$

który po rozpisaniu na składowe przyjmuje postać:

$$x_i^{(n+1)} = \frac{b_i - \sum_{k<i} a_{ik} x_k^{(n+1)} - \sum_{k>i} a_{ik} x_k^{(n)}}{a_{ii}}$$

Jeśli teraz uwzględnimy jak zbudowana jest nasza macierz to wzór dla naszego przypadku będzie miał postać:

$$x_i^{(n+1)} = \frac{b_i - x_{i-1}^{(n+1)} - 0.2x_{i-2}^{(n+1)} - x_{i+1}^{(n)} - 0.2x_{i+2}^{(n)}}{3}$$

Dzięki uwzględnianiu elementów naszej macierzy we wzorach nie ma potrzeby przechowywania jej.

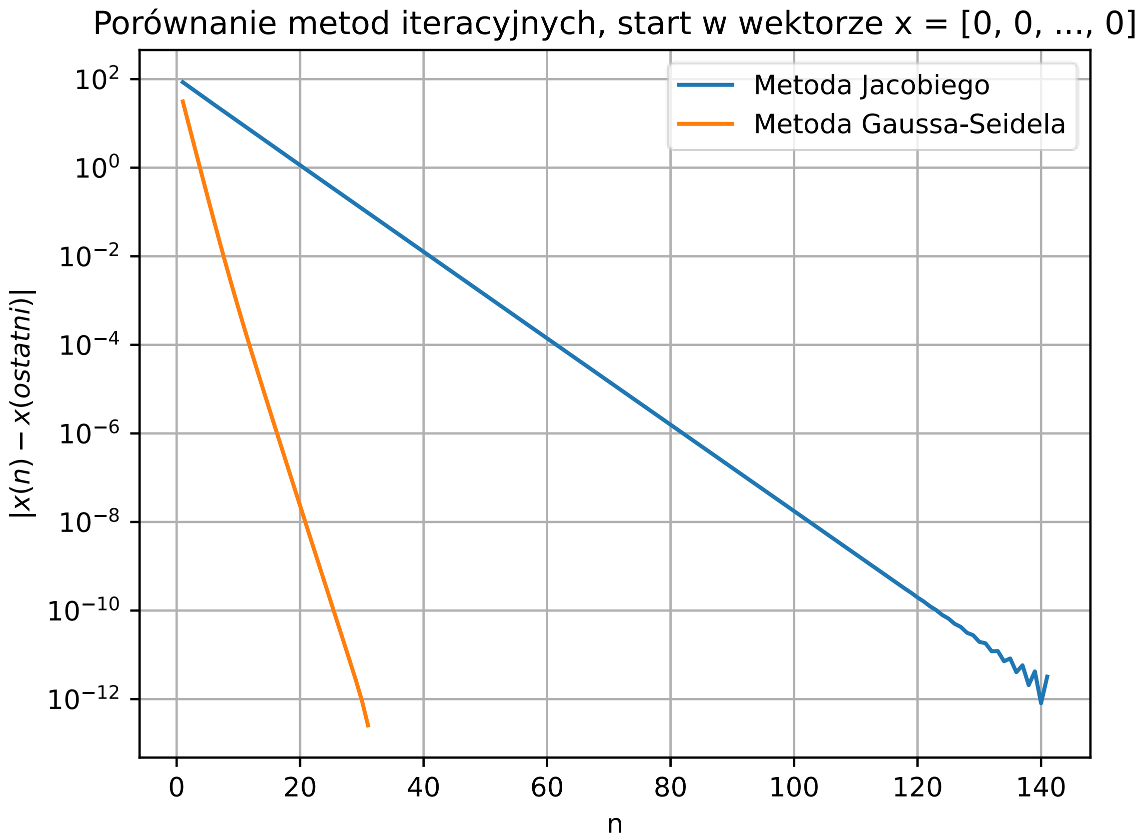
Do określenia kiedy metoda zbiegła i należy ją zakończyć użyłem norm euklidesowych wektorów. Porównuję normę poprzedniej iteracji z normą obecnej iteracji i jeśli ich moduł różnicy jest mniejszy od określonej przeze mnie precyzji to przerywam pętlę. Zastosowałem też zabezpieczenie w przypadku gdyby rozwiązanie nie zbiegało do określonej precyzji, w postaci ustalonej maksymalnej ilości iteracji.

5. Wyniki

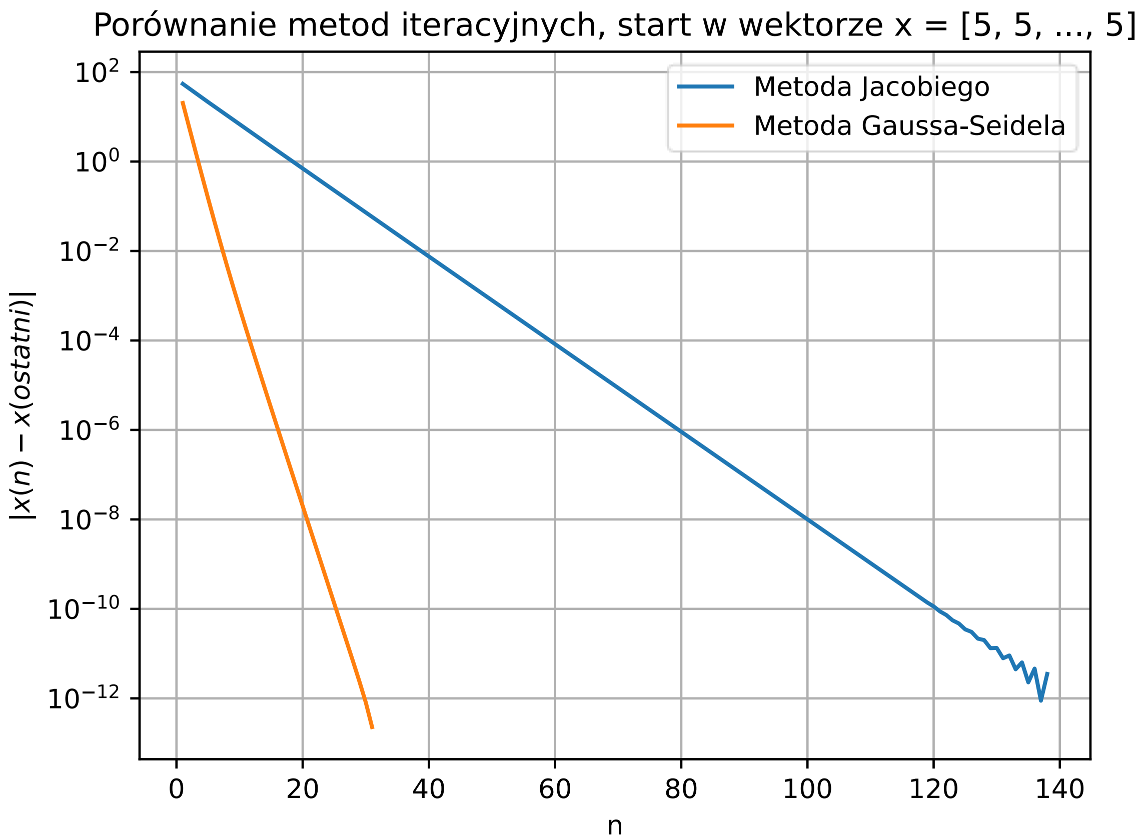
Rozwiązaniem równania o precyzji 10^{-12} są wektory:

Dla metody Jacobiego x = [0.17126009249155258, 0.37523973745157657, 0.5548999253689163, 0.7406038489241693, 0.926023095096212, 1.111087426360555, 1.2962972717646715, 1.481482921363556, 1.6666660898137005, 1.851851945294868, 2.0370370477384023, 2.22222221145369 2, 2.4074074103683376, 2.5925925923670095, 2.7777777776340744, 2.9629629630302445, 3.1481481481354994, 3.3333333333327744, 3.518518518519747, 3.703703703703404, 3.888888888888994, 4.074074074074157, 4.259259259259324, 4.444444444444523, 4.629629629629709, 4.814814814814898, 5.000000000000088, 5.185185185185278, 5.370370370370465, 5.555555555555656, 5.740740740740843 5, 5.92592592592603 3, 6.111111111111222, 6.296296296296411, 6.481481481481601, 6.666666666667895, 6.851851851851979, 7.037037037037169, 7.222222222222357, 7.407407407407547, 7.592592592592734, 7.777777777777927, 7.962962962963114, 8.148148148148303, 8.333333333333494, 8.51851851851868, 8.703703703703868, 8.888888888889058, 9.074074074074247, 9.259259259259437, 9.444444444444624, 9.629629629629816, 9.814814814815005, 10.000000000000194, 10.18518518518538, 10.370370370370575, 10.555555555555761, 10.740740740740952, 10.92592592592614 2, 11.11111111111133, 11.296296296296518, 11.481481481481708, 11.666666666666897, 11.851851851852084, 12.03703703703727, 12.222222222222463, 12.407407407407648, 12.592592592592835, 12.77777777777802, 12.962962962963209, 13.148148148148396, 13.333333333333583, 13.518518518518768, 13.703703703703942, 13.888888888889213, 14.07407407407416, 14.259259259259332, 14.44444444444772, 14.629629629618128, 14.814814814829456, 15.000000000089537, 15.18518518457868, 15.370370372004578, 15.555555556026114, 15.740740716824009, 15.925926030892645, 16.1111109410473, 16.29629569123323, 16.48148655684238, 16.666651111165528, 16.85185669432795, 17.03722138543668, 17.221300545231557, 17.409241909490657, 17.596318652565518, 17.73605398322128, 18.10744020272875, 18.031154065721136, 16.956038061793006, 26.47924370835429]^T

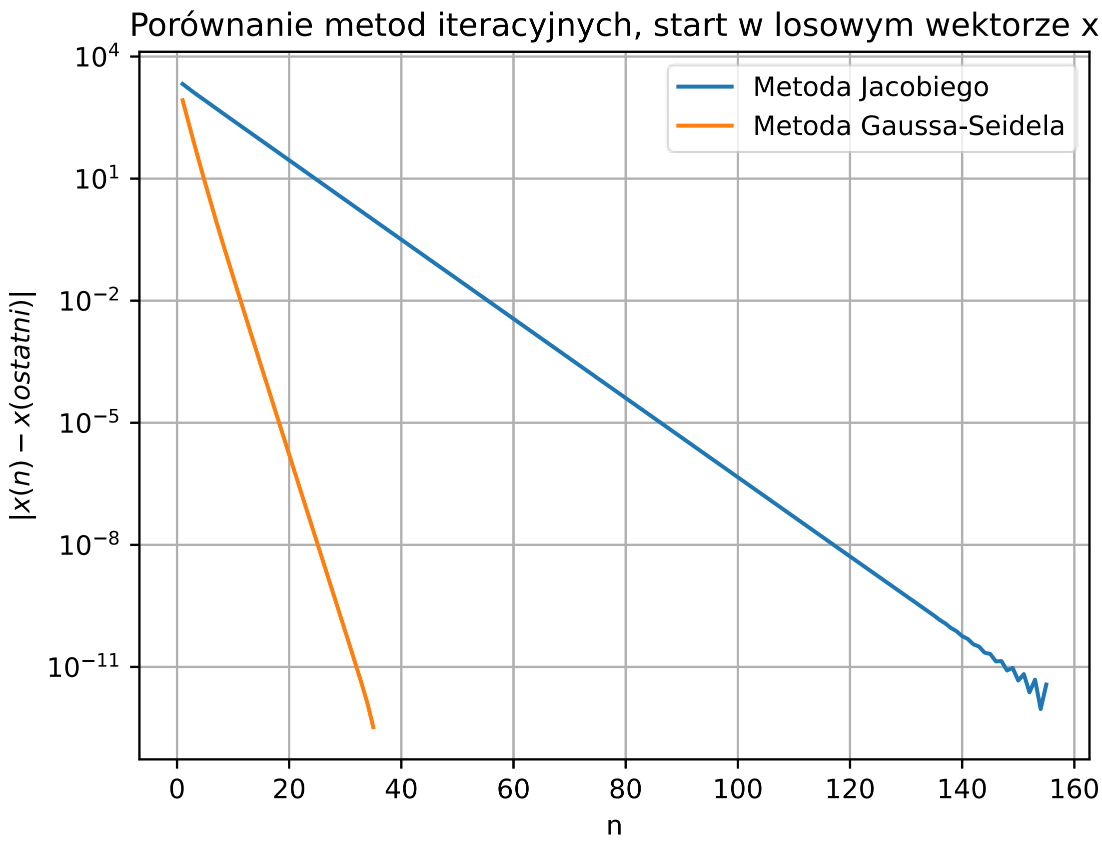
Dla metody Gaussa-Seidela x = [0.1712600924915492, 0.3752397374515706, 0.5548999253689074, 0.7406038489241576, 0.9260230950961977, 1.1110874263605375, 1.2962972717646515, 1.4814829213635328, 1.6666660898136743, 1.8518519452948397, 2.0370370477383695, 2.222222211453657, 2.407407410368299, 2.5925925923669677, 2.777777777634029, 2.9629629630301952, 3.148148148135448, 3.3333333333327193, 3.518518518519688, 3.7037037037033413, 3.8888888888889284, 4.074074074074087, 4.2592592592592515, 4.4444444444444455, 4.62962962962962 9, 4.814814814814814, 4.999999999999999, 5.185185185185186, 5.370370370370369, 5.5555555555555545, 5.740740740740741, 5.9259259259259265, 6.11111111111111, 6.296296296296297, 6.481481481481481, 6.666666666666667, 6.85185185185185, 7.037037037037038, 7.222222222222221, 7.407407407407406, 7.592592592592594, 7.777777777777778, 7.962962962962963, 8.148148148148147, 8.333333333333334, 8.518518518518519, 8.703703703703704, 8.88888888888889, 9.074074074074074, 9.259259259259258, 9.444444444444445, 9.629629629629628, 9.814814814814817, 10.000000000000002, 10.185185185185187, 10.370370370370368, 10.555555555555557, 10.74074074074074, 10.925925925925924, 11.111111111111112, 11.296296296296296, 11.481481481481481, 11.66666666666667, 11.851851851851853, 12.037037037037036, 12.222222222222216, 12.407407407407412, 12.592592592592597, 12.777777777777773, 12.962962962962955, 13.14814814814816, 13.333333333333341, 13.518518518518496, 13.7037037037037, 13.888888888888978, 14.074074074073875, 14.259259259259098, 14.4444444444475, 14.62962962961783, 14.81481481482925, 15.000000000089303, 15.1851851845784, 15.370370372004409, 15.555555556025842, 15.74074071682383, 15.92592603089242, 16.111110941047112, 16.29629569123304, 16.481486556842203, 16.666651111165365, 16.851856694327793, 17.037221385436535, 17.221300545231433, 17.409241909490536, 17.59631865256542, 17.736053983221193, 18.10744020272868, 18.031154065721086, 16.956038061792967, 26.47924370835427]^T



1. Porównanie prędkości zbiegania metod iteracyjnych dla określonego wektora początkowego



2. Porównanie prędkości zbiegania metod iteracyjnych dla określonego wektora początkowego



3. Porównanie prędkości zbiegania metod iteracyjnych dla określonego wektora początkowego

6. Wnioski

Obie metody dają takie same wyniki do określonej precyzji. Na wykresach możemy zauważyć że obie metody zbiegają i kończą swoją pracę przed ustaloną granicą iteracji (500). Metoda Gaussa-Seidela zbiega znacznie szybciej niż metoda Jacobiego. Dzieje się tak ponieważ w metodzie Gaussa-Seidela do obliczań używamy najbardziej aktualnych wartości, gdzie w Jacobim cały czas bazujemy na wektorze z poprzedniej iteracji. Wynik również nie zależy od wyboru wektora początkowego.