SPRAWOZDANIE NUMERYCZNE

NUM5

Zadanie

Rozwiąż układ równań

$$\begin{pmatrix} 3 & 1 & 0.15 & & & & \\ 1 & 3 & 1 & 0.15 & & & \\ 0.15 & 1 & 3 & 1 & 0.15 & & \\ \dots & \dots & \dots & \dots & \dots & \dots & \\ & & & 0.15 & 1 & 3 & 1 \\ & & & & 0.15 & 1 & 3 \end{pmatrix} \mathbf{x} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ \dots \\ N-1 \\ N \end{pmatrix}$$

dla N = 124 za pomocą metod Jacobiego i Gaussa-Seidela. Przedstaw graficznie różnicę pomiędzy pomiędzy dokładnym rozwiązaniem a jego przybliżeniami w kolejnych iteracjach wybierając kilka zestawów punktów startowych. Na tej podstawie porównaj dwie metody.

Cel badania

Celem badania było sprawdzenie efektywności rozwiązywania układu równań metodą jacobiego oraz metodą Gaussa-Seidela dla macierzy rzadkiej, symetrycznej oraz silnie diagonalno dominującej(jest to potrzebne dla zbierzności metdy Jacobiego) o wymiarach NxN, N[124, 1240].

Metoda badawcza

W przedstawionym kodzie zaimplementowano dwie iteracyjne metody rozwiązania układu równań liniowych o postaci **Ax = b**, gdzie **A** to macierz współczynników, **x** to wektor niewiadomych, a **b** to wektor wyrazów wolnych. Skorzystano z metod Jacobiego i Gaussa-Seidela, a następnie porównano ich efektywność.

1. Metoda Jacobiego

Metoda Jacobiego polega na iteracyjnym poprawianiu przybliżonego rozwiązania według wzoru:

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1, j\neq i}^{N} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{N} a_{ij} x_j^{(k)}}{a_{ii}}$$

Dla korzystania ze wzoru musimy zapisać macierz A jako sumę A = L + D + U, gdzię L – macierz trójkątna dolna, D – macierz diagonalna, U – macierz diagonalna górna.

Dla tworzenia równania iteracyjnego możemy przekształcić równanie

Ax = b

Niech A = A + B

(A + B)x = b

Ax + Bx = b

Ax1 = -Bx2 + b

Z czego otrzymujey równanie rekurencyjne

$$A * x^{n+1} = -Bx^n + b$$

Metoda Jacobiego polega ma podstawieniu na miejsce macierzy A macierz diagonalną D, a na miejsce B pomnożoną przez skalar *(-1) sumę macierzy L oraz U. Więc

A = D B = -(L + U)

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

Co po podstawieniu sum daje nam ogólny wzór

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1, j\neq i}^{N} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{N} a_{ij} x_j^{(k)}}{a_{ii}}$$

Dla naszej macierzy wzór wygląda w następny sposób:

$$x_i^{(k+1)} = \frac{x_{i-1}^k - 0.15x_{i-2}^k - x_{i+1}^k - 0.15 * x_{i+2}^k}{3}$$

Korzystając z tego wzoru będziemy rozwiązywali układ równań.

Algorytm

Algorytm programu rozpoczyna się od losowego wektora początkowego **x** o długości **n**, gdzie **n** to liczba niewiadomych. W każdej iteracji, dla każdej niewiadomej, obliczana jest nowa wartość na podstawie powyższego wzoru. Obliczenia te są powtarzane przez określoną liczbę iteracji lub do momentu osiągnięcia zadanego poziomu dokładności. Algorytm monitoruje normę Euklidesową różnicy między kolejnymi wektorami rozwiązań.

2. Metoda Gaussa-Seidela

Metoda Gaussa-Seidela różni się od metody Jacobiego swoim podstawiemniem A oraz B, w danym przypadku wygląda w taki sposób:

A = L+D, B = -U

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

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

Podstawiając macierzy i doprowadzając do ogólnego wzoru:

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{N} a_{ij} x_j^{(k)}}{a_{ii}}$$

W przypadku naszej macierzy wygląda tak:

$$x_i^{k+1} = \frac{b_i - x_{i-1}^{k+1} - 0.15 * x_{i-2}^{n+1} - x_{i+1}^n - 0.15 * x_{i+2}^n}{3}$$

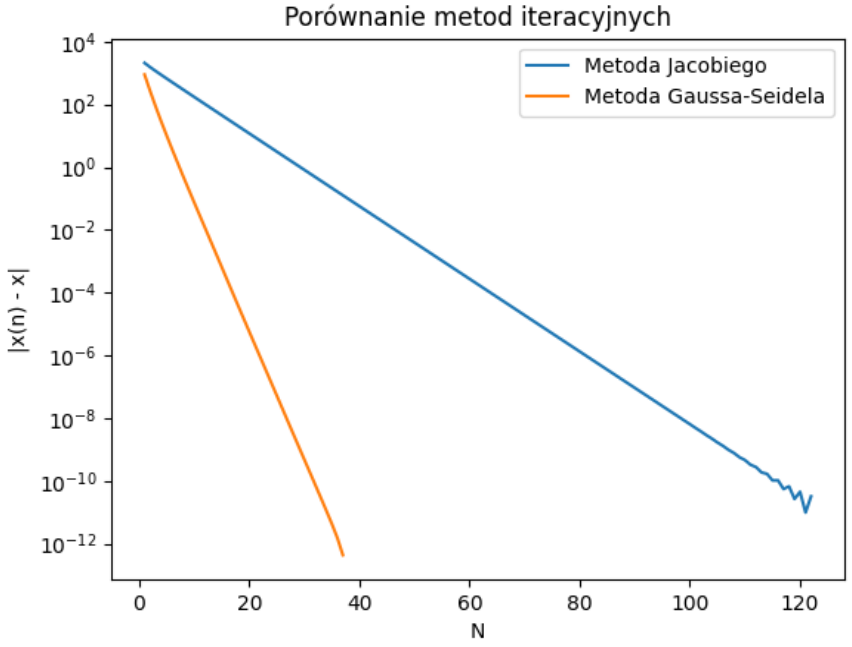
Metoda Gaussa-Seidela na odmiane od metody Jacobiego aktualizuje każdą niewiadomą natychmiast po jej obliczeniu.

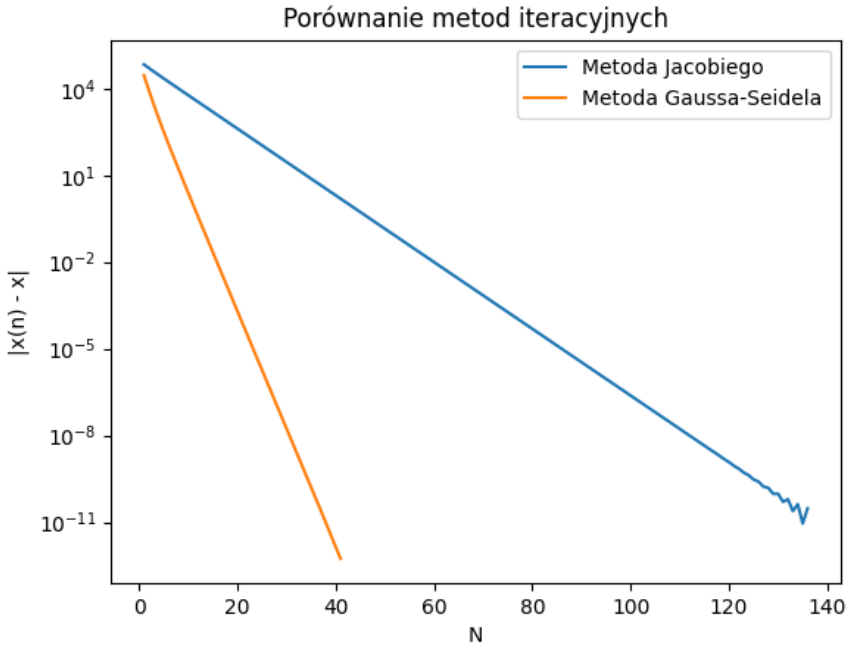
Algorytm

Podobnie jak w metodzie Jacobiego, algorytm monitoruje normę Euklidesową różnicy między kolejnymi wektorami rozwiązań i iteruje do momentu osiągnięcia zadanego poziomu dokładności lub maksymalnej liczby iteracji.

Wyniki

W celu porównania skuteczności obu metod, dla każdej z nich uzyskano listę błędów, reprezentujących różnicę między kolejnymi przybliżeniami a rozwiązaniem dokładnym. Następnie obliczono normę Euklidesową różnicy między ostatnim przybliżeniem a pozostałymi w kolejnych iteracjach. Na koniec wygenerowano wykres, na którym porównano zbieżność obu metod dla różnych punktów startowych. Pierwszy wykres podany jest dla N=124, drugi dla sprawdzenia, jak się zmieni wynik dla innego N =1240. x startowy jest wektowem zerowym, wyliczenie normy jest roznica |x(n) - **x**|, gdzie drugi **x** jest ostatnim. Dane na wykresach podane w log10.





Wynik otrzymany metodą Jacobiego:

[0.1780152340635857, 0.3811616836876326, 0.5652840941471752, 0.7547708048481034, 0.9434196260965555, 1.1320640844032979, 1.3207574555928163, 1.5094336291730517, 1.6981131715567483, 1.886792484492782, 2.0754716888563207, 2.2641509449091566, 2.4528301886640502, 2.641509433883411, 2.8301886792755524, 3.0188679245232968, 3.207547169812768, 3.3962264150955144, 3.584905660378295, 3.773584905661439, 3.962264150944448, 4.150943396227482, 4.339622641510508, 4.528301886793536, 4.716981132076561, 4.905660377359584, 5.094339622642608, 5.283018867925629, 5.471698113208652, 5.660377358491673, 5.849056603774695, 6.037735849057717, 6.226415094340738, 6.41509433962376, 6.603773584906784, 6.792452830189807, 6.981132075472831, 7.169811320755855, 7.358490566038878, 7.547169811321903, 7.735849056604924, 7.924528301887951, 8.113207547170974, 8.301886792453997, 8.490566037737022, 8.679245283020045, 8.86792452830307, 9.056603773586094, 9.245283018869115, 9.433962264152138, 9.622641509435157, 9.811320754718183, 10.0000000000012, 10.188679245284225, 10.377358490567243, 10.566037735850264, 10.75471698113328, 10.943396226416302, 11.132075471699322, 11.320754716982337, 11.509433962265357, 11.698113207548376, 11.886792452831395, 12.075471698114413, 12.264150943397434, 12.452830188680453, 12.641509433963478, 12.830188679246495, 13.018867924529516, 13.207547169812537, 13.39622641509556, 13.584905660378583, 13.773584905661606, 13.962264150944634, 14.150943396227653, 14.33962264151068, 14.528301886793704, 14.716981132076727, 14.905660377359752, 15.094339622642783, 15.283018867925804, 15.471698113208832, 15.660377358491857, 15.849056603774882, 16.03773584905791, 16.226415094340936, 16.415094339623955, 16.60377358490698, 16.792452830189998, 16.981132075473017, 17.169811320756043, 17.358490566039052, 17.54716981132207, 17.735849056605087, 17.924528301888095, 18.11320754717111, 18.30188679245412, 18.49056603773713, 18.679245283020126, 18.867924528303128, 19.05660377358617, 19.245283018868943, 19.433962264152687, 19.62264150943493, 19.81132075470939, 20.00000000005952, 20.188679245069896, 20.37735849088514, 20.566037737632744, 20.754716964256563, 20.94339630136565, 21.132075297361308, 21.320754490967264, 21.509438472994873, 21.698088743609663, 21.88686701225498, 22.07543406853168, 22.263069200985687, 22.460321797505866, 22.613376527798945, 22.875158456135736, 23.232865218771277, 21.061564102655357, 33.15116870484314]

Wynik otrzymany metodą Gaussa-Seidela:

[0.17801523406351125, 0.38116168368743697, 0.5652840941469756, 0.7547708048477387, 0.9434196260962063, 1.132064084402806, 1.320757455592308, 1.5094336291724486, 1.6981131715560995, 1.8867924844920785, 2.0754716888555467, 2.264150944908373, 2.45283018866317, 2.6415094338825553, 2.8301886792745976, 3.01886792452237, 3.2075471698117664, 3.3962264150945263, 3.5849056603772578, 3.7735849056604116, 3.962264150943381, 4.150943396226427, 4.339622641509425, 4.528301886792456, 4.7169811320754755, 4.905660377358479, 5.094339622641528, 5.283018867924504, 5.47169811320757, 5.660377358490547, 5.849056603773597, 6.0377358490566, 6.226415094339617, 6.415094339622656, 6.603773584905638, 6.792452830188704, 6.9811320754716775, 7.169811320754728, 7.358490566037737, 7.547169811320743, 7.73584905660379, 7.924528301886777, 8.11320754716982, 8.301886792452821, 8.490566037735856, 8.679245283018856, 8.867924528301902, 9.056603773584886, 9.245283018867953, 9.433962264150914, 9.622641509433993, 9.811320754716954, 10.00000000000002, 10.188679245283007, 10.37735849056604, 10.566037735849063, 10.754716981132065, 10.943396226415103, 11.132075471698116, 11.32075471698112, 11.509433962264175, 11.698113207547136, 11.88679245283023, 12.075471698113164, 12.264150943396267, 12.45283018867922, 12.641509433962272, 12.830188679245287, 13.018867924528289, 13.207547169811335, 13.39622641509433, 13.584905660377364, 13.773584905660378, 13.962264150943392, 14.150943396226422, 14.33962264150943, 14.528301886792448, 14.716981132075484, 14.905660377358465, 15.094339622641547, 15.283018867924495, 15.47169811320758, 15.66037735849053, 15.849056603773617, 16.03773584905657, 16.22641509433966, 16.415094339622602, 16.603773584905706, 16.792452830188633, 16.981132075471745, 17.169811320754665, 17.358490566037794, 17.547169811320696, 17.735849056603826, 17.92452830188675, 18.11320754716984, 18.301886792452816, 18.490566037735853, 18.67924528301887, 18.867924528301888, 19.056603773584932, 19.245283018867752, 19.433962264151482, 19.62264150943379, 19.811320754708234, 20.000000000058424, 20.18867924506882, 20.377358490884117, 20.566037737631742, 20.754716964255614, 20.94339630136474, 21.132075297360444, 21.32075449096646, 21.509438472994116, 21.698088743608967, 21.886867012254346, 22.07543406853111, 22.263069200985175, 22.46032179750543, 22.613376527798568, 22.875158456135434, 23.232865218771042, 21.0615641026552, 33.15116870484305]

Wartości własne:

[5.29899284 5.29597272 5.29094366 5.2839124 5.27488832 5.26388346
 5.25091249 5.23599268 5.21914387 5.20038841 5.17975115 5.15725936
 5.13294271 5.10683316 5.07896498 5.04937458 5.01810054 4.98518347
 4.95066594 4.91459243 4.87700921 4.83796429 4.79750726 4.75568929
 4.71256294 4.66818214 4.62260202 4.57587886 4.52806996 4.47923354
 4.42942862 4.37871493 4.32715279 4.27480302 4.22172678 4.16798553
 4.11364086 4.05875442 4.0033878 3.9476024 3.89145938 3.83501949
 3.77834302 3.72148969 3.6645185 3.60748772 3.55045472 3.49347595
 3.43660679 3.37990148 3.32341309 3.26719337 3.21129273 3.15576015
 3.10064311 3.04598753 2.99183774 2.93823638 2.88522439 2.83284096
 2.78112347 2.73010749 2.67982669 2.6303129 2.58159603 2.53370404
 2.48666301 2.44049704 2.39522831 2.35087706 2.30746161 2.26499835
 2.22350179 2.18298456 2.14345743 2.10492936 2.06740753 2.03089736
 1.99540259 1.96092529 1.92746591 1.89502336 1.86359506 1.83317698
 1.80376371 1.77534855 1.74792353 1.72147954 1.69600634 1.6714927
 1.64792639 1.62529436 1.60358273 1.58277692 1.56286172 1.54382136
 1.5256396 1.50829981 1.49178506 1.47607819 1.46116188 1.44701876
 1.43363147 1.42098273 1.40905544 1.39783274 1.38729807 1.35966292
 1.36822862 1.37743527 1.34439659 1.33766873 1.35172357 1.32596105
 1.30025507 1.30102085 1.32095868 1.31260714 1.33152745 1.30409231
 1.3092409 1.30640462 1.31651039 1.30229901]

Analiza wyników

Dany eksperyment pokazuje, że dla zadanej macierzy metoda Gaussa-Seidela efektywniej wykorzystuję jej własności, co powoduje szybszą zbieżność błędu dla tej metody. Takie czynniki jak rzadkość macierzy, silna diagonalna dominacja oraz symetria wpływa na szybkość zbieżności obu metod, jednak w tym przypadku wpływ na zbiezność także okazują wartości własne macierzy, które są skoncentrowane wokół jedności. Sprawa polega na tym, że metoda Gaussa_seidela na bieżąco aktualizuje wartości niewiadomych, co pozwala na bardziej skuteczne uwzględnianie informacje z poprzednich iteracji.

Obie metody są skuteczne i ich skuteczność zalezy od warunków początkowych oraz właśności macierzy, w tym jednak przypadku bardziej efektywniejszą jest metoda Gaussa-Seidela.

Autor

Maksym Yankovenko