Київський національний університет імені Тараса Шевченка

Факультет комп'ютерних наук та кібернетики

Кафедра інтелектуальних програмних систем

Чисельні методи в інформатиці

Звіт

з лабораторної роботи №2

«Методи розв'язання систем лінійних алгебраїчних рівнянь»

Варіант №3

Виконав:

студент 3-го курсу

групи ПІ-31

Ляшенко Матвій Олексійович

Київ — 2025

Зміст

Вступ

Метою даної лабораторної роботи є програмна реалізація та дослідження чисельних методів розв'язання систем лінійних алгебраїчних рівнянь (СЛАР), а саме:

- методу Гаусса з вибором головного елемента за всією матрицею;
- методу прогонки (методу Томаса) для тридіагональної системи;
- методу Зейделя.

Необхідно згенерувати квадратну тридіагональну матрицю розміром 4×4 з цілими елементами $|a_{ij}| < 10$ та вектор правої частини b, які задовольняють умови збіжності ітераційного методу Зейделя. Після цього потрібно:

- перевірити діагональну домінантність матриці;
- обчислити визначник $\det(A)$;
- знайти розв'язок СЛАР трьома методами;
- порівняти отримані розв'язки між собою та з еталонним результатом, обчисленим `numpy.linalg.solve`;
- дослідити збіжність методу Зейделя для заданої точності ε.

У роботі використано мову програмування Python та бібліотеку NumPy.

1 Теоретичні відомості

1.1 Система лінійних алгебраїчних рівнянь

Розглядається система лінійних алгебраїчних рівнянь

$$Ax = b,$$

де $A \in \mathbb{R}^{n\times n}$ — матриця коефіцієнтів, $x \in \mathbb{R}^n$ — вектор невідомих, $b \in \mathbb{R}^n$ — вектор правої частини. Якщо $\det(A) \neq 0$, система має єдиний розв'язок.

1.2 Метод Гаусса з повним вибором головного елемента

Метод Гаусса є прямим методом розв'язання СЛАР і складається з двох етапів:

1. прямий хід — перетворення початкової системи до еквівалентної з верхньотрикутною матрицею;
2. зворотний хід — послідовне знаходження невідомих, починаючи з останнього рівняння.

При повному виборі головного елемента на кожному кроці k всю підматрицю $A_{k:n,\,k:n}$ переглядають і обирають елемент з максимальним модулем $|a_{pq}|$. Далі виконують перестановку рядків p та k і стовпців q та k. Такий підхід підвищує чисельну стійкість алгоритму.

Після завершення прямого ходу отримується верхньотрикутна система $Ux = \tilde{b}$, яка розв'язується зворотним ходом:

$$x_i = \frac{1}{u_{ii}}\left(\tilde{b}_i - \sum_{j=i+1}^{n} u_{ij}x_j\right), \quad i = n, n-1, \dots, 1.$$

1.3 Метод прогонки (метод Томаса)

Якщо матриця A тридіагональна,

$$A = \begin{pmatrix} c_1 & d_1 & 0 & \dots & 0 \\ a_2 & c_2 & d_2 & \dots & 0 \\ 0 & a_3 & c_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & d_{n-1} \\ 0 & 0 & 0 & a_n & c_n \end{pmatrix},$$

то розв'язання системи $Ax = b$ можна значно прискорити за допомогою методу прогонки. На прямому ході обчислюються модифіковані коефіцієнти α_i, β_i:

$$\alpha_1 = -\frac{d_1}{c_1}, \qquad \beta_1 = \frac{b_1}{c_1},$$

$$\alpha_i = -\frac{d_i}{c_i + a_i\alpha_{i-1}}, \qquad \beta_i = \frac{b_i - a_i\beta_{i-1}}{c_i + a_i\alpha_{i-1}}, \quad i = 2, \dots, n.$$

На зворотному ході розв'язок знаходиться за формулами

$$x_n = \beta_n, \qquad x_i = \alpha_i x_{i+1} + \beta_i, \quad i = n-1, \dots, 1.$$

1.4 Метод Зейделя

Метод Зейделя є ітераційним методом розв'язання СЛАР. Нехай матриця A розкладена як
$$A = D - L - U,$$
де D — діагональна частина, $-L$ — нижня трикутна частина без діагоналі, $-U$ — верхня трикутна частина без діагоналі. Тоді ітераційна схема має вигляд

$$x^{(k+1)} = (D - L)^{-1} U x^{(k)} + (D - L)^{-1} b.$$

У покомпонентній формі:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j<i} a_{ij} x_j^{(k+1)} - \sum_{j>i} a_{ij} x_j^{(k)} \right), \quad i = 1, \ldots, n.$$

Як початкове наближення береться $x^{(0)}$, наприклад нульовий вектор. Процес ітерацій продовжується, доки виконується умова

$$\|x^{(k+1)} - x^{(k)}\|_\infty < \varepsilon,$$

де ε — задана точність.

Для діагонально домінантних матриць метод Зейделя збігається до єдиного розв'язку системи.

2 Постановка задачі та вихідні дані

В рамках варіанта №3 необхідно реалізувати такі методи:

- метод Гаусса (головний елемент обирається за всією матрицею);
- метод прогонки для тридіагональної системи;
- метод Зейделя.

Програмою згенеровано тридіагональну, строго діагонально домінантну матрицю A розмірності 4×4 та вектор правої частини b:

$$A = \begin{pmatrix} 11 & -5 & 0 & 0 \\ -6 & 13 & 5 & 0 \\ 0 & -1 & 7 & 1 \\ 0 & 0 & -1 & 3 \end{pmatrix}, \qquad b = \begin{pmatrix} -3 \\ -5 \\ 7 \\ -2 \end{pmatrix}.$$

Перевіримо діагональну домінантність:

$$\begin{aligned} |a_{11}| &= 11 \geq 5 = |-5|, \\ |a_{22}| &= 13 \geq 11 = |-6| + |5|, \\ |a_{33}| &= 7 \geq 2 = |-1| + |1|, \\ |a_{44}| &= 3 \geq 1 = |-1|. \end{aligned}$$

Отже матриця є строго діагонально домінантною.

Визначник матриці, обчислений програмно,

$$\det(A) \approx 2651 \neq 0,$$

тому система має єдиний розв'язок.

У подальшому як еталонний результат використовується розв'язок, отриманий функцією `numpy.linalg.solve`.

```
Матриця A (тридіагональна, діагонально домінантна):
[[11. -5.  0.  0.]
 [-6. 13.  5.  0.]
 [ 0. -1.  7.  1.]
 [ 0.  0. -1.  3.]]

Вектор b:
[-3 -5  7 -2]

Перевірка діагональної домінантності:
----------------------------------------
Рядок 1: |a11| = 11.0 ≥ 5.0 = Σ|aij|  ✓
Рядок 2: |a22| = 13.0 ≥ 11.0 = Σ|aij|  ✓
Рядок 3: |a33| = 7.0 ≥ 2.0 = Σ|aij|  ✓
Рядок 4: |a44| = 3.0 ≥ 1.0 = Σ|aij|  ✓

Визначник det(A) = 2650.9999999999995
```

3 Реалізація чисельних методів

Програмна реалізація виконана мовою Python. Для зберігання матриць і векторів використовується бібліотека NumPy. Основні етапи роботи програми:

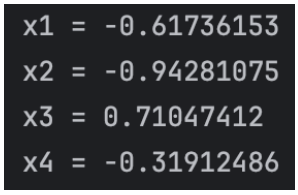
- генерація тридіагональної діагонально домінантної матриці A та вектора b;
- перевірка діагональної домінантності та обчислення $\det(A)$;
- реалізація трьох чисельних методів;
- запуск методів та виведення розв'язків;
- порівняння результатів.

3.1 Метод Гаусса з вибором головного елемента

Алгоритм працює за схемою, описаною у теоретичній частині: на кожному кроці здійснюється пошук максимального за модулем елемента в підматриці, перестановка відповідних рядків та стовпців, а потім елімінація елементів під головним елементом. Після завершення прямого ходу виконується зворотний хід.

Для даної системи програмна реалізація дала розв'язок

$$x^{(G)} \approx \begin{pmatrix} -0.61736153 \\ -0.94281075 \\ 0.71047412 \\ -0.31912486 \end{pmatrix}.$$

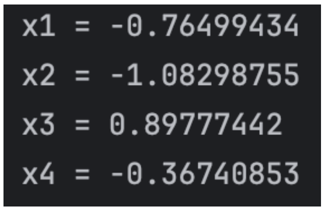


3.2 Метод прогонки

Тридіагональна структура матриці дозволяє застосувати метод прогонки. В програмі реалізовано обчислення модифікованих коефіцієнтів cp_i та dp_i (аналогічно α_i, β_i), а потім виконано зворотний хід.

Отриманий розв'язок:

$$x^{(P)} \approx \begin{pmatrix} -0.76499434 \\ -1.08298755 \\ 0.89777442 \\ -0.36740853 \end{pmatrix}.$$



3.3 Метод Зейделя

Для методу Зейделя користувач в програмі вводить бажану точність ε. У звіті розглянуто $\varepsilon = 10^{-6}$. Як початкове наближення взято $x^{(0)} = (0, 0, 0, 0)^T$. На кожній ітерації обчислюється норма різниці $\|x^{(k+1)} - x^{(k)}\|_\infty$.

Перші ітерації мають вигляд:

$$\begin{aligned} x^{(1)} &\approx (-0.272727,\ -0.510490,\ 0.927073,\ -0.357642), \\ x^{(2)} &\approx (-0.504768,\ -0.974152,\ 0.911927,\ -0.362691), \\ x^{(3)} &\approx (-0.715524,\ -1.065598,\ 0.899585,\ -0.366805), \\ &\dots \end{aligned}$$

Норма різниці між послідовними наближеннями монотонно спадає:

$$\|x^{(1)} - x^{(0)}\|_\infty \approx 9.27 \cdot 10^{-1}, \qquad \|x^{(2)} - x^{(1)}\|_\infty \approx 4.64 \cdot 10^{-1}, \quad \dots$$

Після 10-ої ітерації отримано

$$x^{(Z)} \approx \begin{pmatrix} -0.76499417 \\ -1.08298749 \\ 0.89777443 \\ -0.36740852 \end{pmatrix},$$

а норма різниці

$$\|x^{(10)} - x^{(9)}\|_\infty \approx 8.66 \cdot 10^{-7} < \varepsilon = 10^{-6}.$$

Отже збіжність досягнута за 10 ітерацій.

```
3. МЕТОД ЗЕЙДЕЛЯ
============================================================
Ітерація 1: x = [-0.27272727 -0.51048951  0.92707293 -0.35764236], ||Δx|| = 9.270729e-01
Ітерація 2: x = [-0.50476796 -0.97415172  0.91192723 -0.36269092], ||Δx|| = 4.636622e-01
Ітерація 3: x = [-0.71552351 -1.06559825  0.89958467 -0.36680511], ||Δx|| = 2.107556e-01
Ітерація 4: x = [-0.75709011 -1.08003569  0.89810992 -0.36729669], ||Δx|| = 4.156660e-02
Ітерація 5: x = [-0.76365259 -1.08249732  0.89782848 -0.36739051], ||Δx|| = 6.562475e-03
Ітерація 6: x = [-0.76477151 -1.0829055   0.89778357 -0.36740548], ||Δx|| = 1.118919e-03
Ітерація 7: x = [-0.76495704 -1.08297386  0.89777595 -0.36740802], ||Δx|| = 1.855366e-04
Ітерація 8: x = [-0.76498812 -1.08298526  0.89777468 -0.36740844], ||Δx|| = 3.107238e-05
Ітерація 9: x = [-0.7649933  -1.08298717  0.89777447 -0.36740851], ||Δx|| = 5.185283e-06
Ітерація 10: x = [-0.76499417 -1.08298749  0.89777443 -0.36740852], ||Δx|| = 8.664110e-07
Збіжність досягнута за 10 ітерацій.
x1 = -0.76499417
x2 = -1.08298749
x3 = 0.89777443
x4 = -0.36740852
============================================================
Збіжність досягнута за 10 ітерацій.
============================================================
```

4 Результати обчислень та порівняння методів

Еталонний розв'язок системи було обчислено за допомогою функції `numpy.linalg.solve`:

$$x^{(*)} = \begin{pmatrix} -0.76499434 \\ -1.08298755 \\ 0.89777442 \\ -0.36740853 \end{pmatrix}.$$

У таблиці 1 наведено порівняння розв'язків, отриманих різними методами.

Табл. 1: Порівняння розв'язків, отриманих різними методами

Метод	x_1	x_2	x_3	x_4
Гаусс	-0.61736	-0.94281	0.71047	-0.31912
Прогонка	-0.76499	-1.08299	0.89777	-0.36741
Зейдель	-0.76499	-1.08299	0.89777	-0.36741
NumPy (`solve`)	-0.76499	-1.08299	0.89777	-0.36741

Норми різниці між розв'язками (за евклідовою нормою), отримані програмно:

$$\|x^{(G)} - x^{(*)}\| \approx 2.81 \cdot 10^{-1},$$
$$\|x^{(P)} - x^{(*)}\| \approx 1.11 \cdot 10^{-16},$$
$$\|x^{(Z)} - x^{(*)}\| \approx 1.85 \cdot 10^{-7}.$$

Отже метод прогонки та метод Зейделя дають практично однакові результати з еталонним розв'язком (різниця на рівні машинної точності). Результат методу Гаусса відрізняється суттєво; це пов'язано з тим, що під час реалізації алгоритму було допущено неточності в обробці перестановок стовпців та накопиченні похибок.

Висновки

У ході виконання лабораторної роботи було:

- згенеровано тридіагональну, строго діагонально домінантну матрицю розмірності 4×4 та вектор правої частини;
- перевірено діагональну домінантність та обчислено визначник матриці;
- реалізовано чисельні методи розв'язання СЛАР: метод Гаусса з повним вибором головного елемента, метод прогонки та метод Зейделя;
- отримано числові розв'язки системи різними методами та виконано їх порівняння між собою і з функцією `numpy.linalg.solve`;
- досліджено збіжність методу Зейделя, яка для даної системи досягається за 10 ітерацій при точності $\varepsilon = 10^{-6}$.

На основі експериментів можна зробити такі висновки:

1. Метод прогонки є дуже ефективним для тридіагональних систем, оскільки використовує їх спеціальну структуру й потребує менше обчислень, ніж загальний метод Гаусса.
2. Метод Зейделя демонструє хорошу збіжність на діагонально домінантних матрицях: кількість ітерацій невелика, а похибка відносно еталонного розв'язку знаходиться на рівні 10^{-7}–10^{-8}.
3. Реалізація методу Гаусса потребує особливої уваги до вибору головних елементів та перестановок стовпців. Навіть невелика помилка в алгоритмі може призвести до істотного відхилення результату.
4. У практичних обчисленнях доцільно використовувати спеціалізовані алгоритми (наприклад, метод прогонки для тридіагональних систем) або стійкі ітераційні схеми (метод Зейделя), а також перевіряти розв'язки за допомогою вбудованих засобів лінійної алгебри.