

Лабораторна робота № 3

Розробка програми для розв'язання системи лінійних рівнянь методом Ітерації.

Мета роботи: практично засвоїти принципи ітераційних методів (методів послідовного наближення) при роботі з матричними структурами систем лінійних рівнянь та керування точністю обчислень.

Теоретичні відомості

Історично ітераційні методи розв'язання систем лінійних рівнянь розвивались як альтернативні точним методам (наприклад Гаусса) щодо розв'язання систем з великою кількістю рівнянь та невідомих за відсутності комп'ютерів, або при їх низькій продуктивності. На сьогоднішній день метод ітерації розглядається як загальний приклад ітераційного алгоритму.

Нехай дано систему лінійних рівнянь:

$$\left. \begin{array}{l} a_{11} \cdot x_1 + a_{12} \cdot x_2 + \dots + a_{1n} \cdot x_n = b_1 \\ a_{21} \cdot x_1 + a_{22} \cdot x_2 + \dots + a_{2n} \cdot x_n = b_2 \\ \dots \quad \dots \quad \dots \quad \dots \quad \dots \\ a_{n1} \cdot x_1 + a_{n2} \cdot x_2 + \dots + a_{nn} \cdot x_n = b_n \end{array} \right\} \quad (3.1)$$

Використавши матричні позначення:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix},$$

систему (3.1) коротко можна записати у вигляді матричного рівняння:

$$\mathbf{Ax} = \mathbf{b} \quad (3.1')$$

Вважаючи, що діагональні елементи матриці \mathbf{A} не дорівнюють нулю:

$$a_{ii} \neq 0 \quad (i = 1, 2, \dots, n)$$

розв'яжемо перше рівняння системи (3.1) відносно x_1 , друге рівняння – відносно x_2 , тощо. В результаті таких дій одержимо альтернативну систему:

$$\left. \begin{aligned} x_1 &= \beta_1 + \alpha_{12}x_2 + \alpha_{13}x_3 + \dots + \alpha_{1n}x_n \\ x_2 &= \beta_2 + \alpha_{21}x_1 + \alpha_{23}x_3 + \dots + \alpha_{2n}x_n \\ \dots &\quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\ x_n &= \beta_n + \alpha_{n1}x_1 + \alpha_{n2}x_2 + \dots + \alpha_{n,n-1}x_{n-1} \end{aligned} \right\} \quad (3.2)$$

де

$$\beta_i = \frac{b_i}{a_{ii}}; \quad \alpha_{ij} = -\frac{a_{ij}}{a_{ii}} \quad \text{при } i \neq j$$

і $\alpha_{ij} = 0$ при $i = j$, ($i, j = 1, 2, \dots, n$).

Позначивши матрицями:

$$\mathbf{\alpha} = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \dots & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & \dots & \alpha_{2n} \\ \dots & \dots & \dots & \dots \\ \alpha_{n1} & \alpha_{n2} & \dots & \alpha_{nn} \end{pmatrix} \quad \text{та} \quad \mathbf{\beta} = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \dots \\ \beta_n \end{pmatrix},$$

систему (3.2) можна записати в матричній формі:

$$\mathbf{x} = \mathbf{\beta} + \mathbf{\alpha} \cdot \mathbf{x} \quad (3.2')$$

Систему (3.2) будемо розв'язувати методом послідовних наближень. За перше наближення можна прийняти, наприклад, вектор вільних членів $\mathbf{x}^{(0)} = \mathbf{\beta}$. Далі, послідовно знаходимо вектори \mathbf{x} :

$$\mathbf{x}^{(1)} = \mathbf{\beta} + \mathbf{\alpha} \cdot \mathbf{x}^{(0)} \quad \text{- перше наближення};$$

$$\mathbf{x}^{(2)} = \mathbf{\beta} + \mathbf{\alpha} \cdot \mathbf{x}^{(1)} \quad \text{- друге наближення, тощо, використовуючи загальну}$$

формулу:

$$\mathbf{x}^{(k+1)} = \mathbf{\beta} + \mathbf{\alpha} \cdot \mathbf{x}^{(k)} \quad (k = 0, 1, 2, \dots) \quad (3.3)$$

Якщо послідовність наближень $\mathbf{x}^{(0)}$, $\mathbf{x}^{(1)}$, ..., $\mathbf{x}^{(k)}$, ... має границю

$$\mathbf{x} = \lim_{k \rightarrow \infty} \mathbf{x}^{(k)}$$

то ця границя є розв'язком системи (3.2), а отже і (3.1).

Метод послідовних наближень, записаний формулою (3.3) називають методом ітерацій. Процес ітерацій добре збігається (кількість ітерацій для

одержання заданої точності незначна) у тому випадку, коли елементи матриці α досить малі за абсолютним значенням. Щодо системи рівнянь (3.1) це означає, що діагональні елементи a_{ii} за абсолютною величиною повинні бути більше недіагональних, а в ідеальному випадку мають перевищувати суму недіагональних елементів.

Для виконання останньої умови, систему лінійних рівнянь перетворюють, переставляючи рівняння місцями та використовуючи лінійні перетворення.

При практичних розрахунках процес ітерацій зупиняють у випадках:

- досягнення заданої точності коренів;
- досягнення заданої максимальної кількості ітерацій;
- досить малій різниці між декількома послідовно знайденими значеннями коренів.

В комп'ютерних розрахунках використання останнього способу зупинки ускладнене, тому використовують комбінацію першого та другого.

Принциповим є вибір способу оцінки точності. Логічно оцінювати точність за різницею (розбігом) r лівої та правої частини системи (3.1).

Однак, при заданій точності, наприклад $\varepsilon = 0,01$, досягнення випадку $|r_i| \leq \varepsilon$ для усіх рівнянь, може не забезпечити відповідний рівень точності для кожного кореня. Доцільніше використовувати значення сумарного розбігу,

що за умови $\sum_{i=1}^n |r_i| \leq \varepsilon$, повинно забезпечити достатню точність знаходження коренів системи.

Розробка програми

Програма розв'язку системи лінійних рівнянь методом ітерації повинна містити наступні послідовні частини:

1. Блок введення елементів матриці початкової системи;
2. Вибір початкових значень коренів;

3. Ітераційну процедуру, що об'єднує оцінку поточного значення розбігу, обчислення нових значень коренів та логіку переходу до наступної ітерації чи припинення розв'язку;
4. Вивід результатів обчислень.

Виконання першої частини нічим принципово не відрізняється від аналогічної, запропонованої в роботі 2 для методу Гаусса. Однак, кількість масивів може відрізнятися: доцільно вводити окремий одновимірний масив для правих частин рівнянь **B[N]** і одновимірний масив для поточних значень розбігів **R[N]**, який можна описати і локально – в описі функції.

Частина 2 – вибір початкових значень коренів – може виконуватись за простою процедурою, через реалізацію формули **X[i]:=B[i]/A[i,i]**. Інший варіант частини 2 може включати перетворення початкової системи до вигляду, придатного для швидкого збігання процесу.

Частина 3 є ключовою, і може бути реалізована у вигляді циклу з післяумовою: **Repeat ... Until**, в тілі якого реалізується цикл по знаходженню нових наближень коренів. Умовою виходу з циклу можна вважати досягнення однієї з двох подій: достатньої точності коренів або граничного числа ітерації. Сумарний розбіг може бути реалізований у вигляді зовнішньої функції, як показано нижче:

```
Function RS(X:myarrN):real;  
Var I,J : integer;  
R: myarrN;  
Begin  
For I:=1 to N do  
  Begin  
    R[I]=-B[I];  
    For J:=1 to N do  
      R[I]:=R[I]+A[I,J]*X[J];  
    End;  
  RS:=0;  
  For I:=1 to N do  
    RS:=RS+Abs(R[i]);  
  End;
```

Якщо лічильник ітерацій k знаходиться усередині циклу, умова виходу з циклу матиме вигляд:

Until (RS(X) <= eps) or (k > N)

eps – задана точність;

N – гранична кількість ітерацій.

Вивід результатів не відрізняється від реалізованого в методі Гаусса, за виключенням того, що може бути доцільно додатково вивести кількість ітерацій та кінцеве значення сумарного розбігу.

Завдання

Розробити, скласти та відлагодити програму розв'язання системи лінійних рівнянь методом ітерацій. Як варіант – з процедурою перетворення системи

Зробити висновки.

Рекомендована література: [6], [7], [9].