МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М.В.Ломоносова

Механико-математический факультет

Кафедра Теоретической механики и мехатроники

Решение систем линейных уравнений с помощью метода Вращения

Преподаватель:
Самохин Александр Сергеевич
Работу выполнил:
Студент 422 группы
Дергунов Максим Олегович

Москва, 2022 г.

Содержание

1 Математическое описание

Требуется решить линейную систему $Ax = b, A \in \mathbf{M}_n(\mathbf{R}^n)$ вида. Обозначим $a_1 = (a_{11}, \ldots, a_{n1})^t$ – первый столбец матрицы A. Согласно теории метода, существуют $n-1$ матриц $T_{12} = T_{12}(\varphi_{12}), T_{13} = T_{13}(\varphi_{13}), \ldots, T_{1n} = T_{1n}(\varphi_{1n})$, таких, что $T_{1n} \ldots T_{13} T_{12} a_1 = \|a_1\| e_1$ (причем значения углов $\varphi_{1k}, k = 2, \ldots, n$ определяются коэффициентами матрицы). Умножим систему $Ax = b$ на $T_{1n} \ldots T_{13} T_{12}$ слева, получим

$$A^{(1)} x = b^{(1)}$$

где

$$A^{(1)} = T_{1n} \ldots T_{13} T_{12} A = \begin{pmatrix} \|a_1\| & c_{12} & \cdots & c_{1n} \\ 0 & a_{22}^{(1)} & \ldots & a_{2n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & \ldots & a_{nn}^{(1)} \end{pmatrix}, b^{(1)} = T_{1n} \ldots T_{13} T_{12} b$$

Далее процесс применяется к подматрице $\left(a_{ij}^{(1)}\right)_{i,j=2,\ldots,n}$ Пусть сделаны $k-1, k = 1, \ldots, n-1$ шагов этого процесса, т.е. система преобразована к виду

$$A^{(k-1)} x = b^{(k-1)},$$

где $A^{(k-1)} = \prod_{i=k-1}^{1} \prod_{j=n}^{i+1} T_{ij} A, \quad b^{(k-1)} = \prod_{i=k-1}^{1} \prod_{j=n}^{i+1} T_{ij} b, \quad A^{(k-1)} =$

$$\begin{pmatrix} \|a_1\| & c_{12} & c_{13} & \ldots & c_{1,k-1} & c_{1k} & \cdots & c_{1n} \\ & \left\|a_1^{(1)}\right\| & c_{23} & \ldots & c_{2,k-1} & c_{2k} & \ldots & c_{2n} \\ & & \left\|a_1^{(2)}\right\| & \ldots & c_{3,k-1} & c_{3k} & \ldots & c_{3n} \\ & & & \ddots & \vdots & \vdots & \ddots & \vdots \\ & & & & \left\|a_1^{(k-2)}\right\| & c_{k-1,k} & \ldots & c_{k-1,n} \\ & & & & & a_{kk}^{(k-1)} & \ldots & a_{kn}^{(k-1)} \\ & & & & & \vdots & \ddots & \vdots \\ & & & & a_{nk}^{(k-1)} & \ldots & a_{nn}^{(k-1)} & \end{pmatrix}.$$

(здесь $\prod_{j=n}^{i+1}$ означает, что сомножители берутся в порядке $n, \ldots, i+1$). Обозначим

$$a_1^{(k-1)} = \left(a_{kk}^{(k-1)}, \ldots, a_{nk}^{(k-1)}\right)^t$$

- первый столбец подматрицы $\left(a_{ij}^{(k-1)}\right)_{i,j=k,\ldots,n}$. Согласно теории существуют $n-k$ матриц

$$T_{k,k+1} = T_{k,k+1}(\varphi_{k,k+1}), T_{k,k+2} = T_{k,k+2}(\varphi_{k,k+2}), \ldots, T_{k,n} = T_{kn}(\varphi_{kn})$$

таких, что

$$T_{kn}\ldots T_{k,k+2}T_{k,k+1}a_1^{(k-1)} = \left\|a_1^{(k-1)}\right\| e_1^{(n-k+1)}$$

(значения углов $\varphi_{kj}, j = k+1,\ldots,n$ известны), здесь $e_1^{(m)} = (1,0,\ldots,0)^t \in \mathbf{R}^m$. Умножим систему приведенную на $T_{kn}\ldots T_{k,k+2}T_{k,k+1}$ слева, получим

$$A^{(k)}x = b^{(k)}$$

После $n-1$ шагов этого процесса система примет вид:

$$Rx = y$$

r, де

$$R = A^{(n-1)} = \prod_{i=n-1}^{1}\prod_{j=n}^{i+1}T_{ij}A, \quad y = b^{(n-1)} = \prod_{i=n-1}^{1}\prod_{j=n}^{i+1}T_{ij}b$$

Далее система решается обратным ходом метода Гаусса.

2 Описание реализации

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1m}x_m = f_1, \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2m}x_m = f_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mm}x_m = f_m \end{cases}$$

Допустим c_1 и s_1 - ненулевые числа. Умножаем 1-е уравнение системы на c_1, 2-е на s_1 и складываем их; уравнением, которое мы получили, заменяем 1-е уравнение системы. Далее 1-е уравнение начальной системы нужно умножить на - s_1, 2-е - на c_1 и итогом этого заменяем 2-е уравнение. Т.о., первые 2 уравнения заменяем уравнениями:

$$\left(c_1a_{11} + s_1a_{21}\right)x_1 + \left(c_1a_{12} + s_1a_{22}\right)x_2 + \ldots + \left(c_1a_{1m} + s_1a_{2m}\right)x_m = c_1f_1 + s_1f_2$$
$$\left(-s_1a_{11} + c_1a_{21}\right)x_1 + \left(-s_1a_{12} + c_1a_{22}\right)x_2 + \ldots + \left(-s_1a_{1m} + c_1a_{2m}\right)x_m = -s_1f_1 + c_1f_2$$

На параметры c_1 и s_1 наложим 2 условия: - условие исключения x_1 из второго уравнения и

$$c_1^2 + s_1^2 = 1$$

- условие нормировки Получаем:

$$c_1 = \frac{a_{11}}{\sqrt{a_{11}^2 + a_{21}^2}}, \quad s_1 = \frac{a_{21}}{\sqrt{a_{11}^2 + a_{21}^2}}$$

После преобразований получаем систему:

$$\begin{cases} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \ldots + a_{1m}^{(1)}x_m = f_1^{(1)} \\ a_{22}^{(1)}x_2 + \ldots + a_{2m}^{(1)}x_m = f_2^{(1)} \\ a_{31}x_1 + a_{32}x_2 + \ldots + a_{3m}x_m = f_3 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mm}x_m = f_m \end{cases}$$

где

$$a_{1j}^{(1)} = c_1 a_{1j} + s_1 a_{2j} \quad (j = 1, 2, \ldots, m), \quad f_1^{(1)} = c_1 f_1 + s_1 f_2$$
$$a_{2j}^{(1)} = -s_1 a_{1j} + c_1 a_{2j} \quad (j = 1, 2, \ldots, m), \quad f_2^{(1)} = -s_1 f_1 + c_1 f_2$$

Продолжаем процесс пока матрица не станет треугольной, далее применим обратный ход метода Гаусса.

3 Результат работы

В программе так же вычисляется ошибка метода вращения, которая будет суммой элементов, которые должны строго равнятся нулю, но не могут в виду компьютерной неточности.На сайте https://planetcalc.ru/9083/ гененировалась случайная матрица различных размеров. Проверка работы на малых погрешностях:

```
Исходная матрица

  1.000000  2.000000  3.000000  4.000000  5.000000
  2.000000  3.000000  1.000000  5.000000  1.000000
  6.000000  3.000000  1.000000 13.000000  5.000000
  5.000000 12.000000421.000000 15.000000  1.000000
Треугольная матрица полученная методом вращения

  8.124038 10.585868260.461595 20.556279  5.169843
  0.000000  7.344344314.090376  3.321459 -2.005254
  0.000000 -0.000000103.725567 -0.504789 -2.648475
 -0.000000 -0.000000  0.000000  1.073541  3.773230
```

Ответ выводится в файл Решение.txt, для этой матрицы он такой:

x_1	-6.029500
x_2	-1.502107
x_3	-0.008429
x_4	3.514750

Ошибки арифметических операций

Size	Err
4x4	5.036799e-16
10x10	-1.119239e-14
100x100	-2.664989e-14
500x500	-4.363140e-14
1000x1000	-4.363140e-14
10000x10000	-5.822940e-13

4 Работа с программой

Инструкция по работе с программой:
Ввести матрицу в файл input.txt не оставляя пробелов после последних элементов и не оставляя пустую строку. Далее просто запустить. Пример:

```
1 1 1 3
1 1 1 2
3 4 1 1
```

Важно чтобы число строк было на 1 меньше, чем число столбцов, последний столбец это вектор b в системе.

5 Список литературы

[1]Богачев К.Ю. 1999 Практикум на ЭВМ. Методы решения линейных систем и нахождения собственных значений