



УНИВЕРСИТЕТ ИТМО

Факультет программной
инженерии и компьютерной техники
Вычислительная математика

Лабораторная работа № 1
Вариант 8 (Метод Гаусса-Зейделя)

Преподаватель: Перл Ольга Вячеславовна
Выполнил: Геллер Л. А.
Группа: Р3230

Санкт-Петербург, 2021

Описание метода, расчетные формулы

- 1) Задана СЛАУ $Ax = B$, где:
 A – матрица коэффициентов,
 x – столбец искомых неизвестных,
 B – столбец ответов

Или:

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2$$

...

$$a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n$$

- 2) Выразим неизвестные из каждого уравнения СЛАУ:

$$x_1 = \frac{1}{a_{11}} (b_1 - a_{12}x_2 - a_{13}x_3 - \dots - a_{1n}x_n)$$

$$x_2 = \frac{1}{a_{22}} (b_2 - a_{21}x_1 - a_{23}x_3 - \dots - a_{2n}x_n)$$

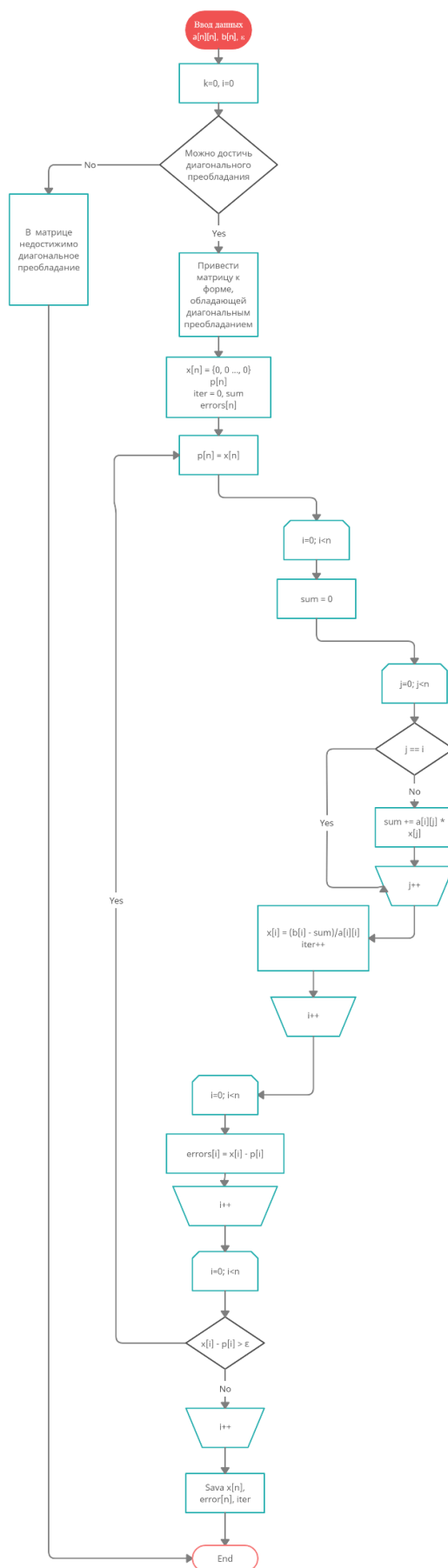
...

$$x_n = \frac{1}{a_{nn}} (b_n - a_{n1}x_1 - a_{n2}x_2 - \dots - a_{nn}x_n)$$

- 3) Зададим начальные значения столбцу неизвестных (нулевые)
- 4) Поочередно будем вычислять очередной x из пункта 2, используя при вычислении уже полученные x . Вычисление таким образом всех x от 1 до n составляет одну итерацию
- 5) Производим итерации до момента, когда новые значения x не станут с заданной погрешностью близкими к предыдущим
- 6) Для сходимости метода достаточным является условие диагонального преобладания:

$$|a_{ij}| \geq \sum_{j \neq i} |a_{ij}|, \text{ где в хотя бы одной строке неравенство выполняется строго}$$

Блок-схема численного метода



Листинг реализованного численного метода программы

```
public static double[] solve(double[][] matrix, double[] answerMatrix, double accuracy) throws MatrixCreateException {  
    if(is_diagonally_dominant(matrix)){  
        double[] x = new double[matrix.length];  
        double[] previousX;  
        iterations = 0;  
        do {  
            previousX = Arrays.copyOf(x, x.length);  
            for (int i=0; i<matrix.length; i++){  
                double sum = 0;  
                for(int j=0; j<matrix.length; j++) {  
                    if (j != i) sum += matrix[i][j] * x[j];  
                }  
                x[i] = (answerMatrix[i] - sum) / matrix[i][i];  
                iterations++;  
            }  
        }while (!converge(previousX, x, accuracy));  
        return x;  
    }else {  
        throw new MatrixCreateException("Sorry, the matrix is not diagonally dominant");  
    }  
}
```

Примеры и результаты работы программы на разных данных

1) Рандомно сгенерированная матрица:

- Run: java Main -r 3
- Enter accuracy: 0.01
- Matrix:
[5.0, 1.0, 3.0]
[3.0, 16.0, 9.0]
[9.0, 2.0, 13.0]
- Answer Matrix:
[1.0, 0.0, 7.0]
- x0: -0.1422435021869653 +- -0.005631609015879779
x1: -0.3596119161080785 +- -0.010655705612854227
x2: 0.6922627193768343 +- 0.005538145566817421
- Solution found in 15 iterations

2) Матрица, загруженная из файла:

- Содержимое файла matrix1:
4 - размер матрицы
3 2 1 6 - матрица
0 2 0 4
0 0 3 7
5 6 7 7
4 4 4 6 - столбец ответов
0,001 - точность вычисления

- Run: java Main -f matrix1
- Sorry, the matrix is not diagonally dominant (К сожалению, в такой матрице невозможно диагональное преобладание)

3) Пользовательский ввод

- Run: java Main -u
- Enter matrix size: 6
- Enter 6 space separated elements of matrix - line 0: 20, 6, 1, 5, 0, 6
- Enter 6 space separated elements of matrix - line 1: 6, 15, 1, 4, 2, 1
- Enter 6 space separated elements of matrix - line 2: 9 9 35 3 9 2
- Enter 6 space separated elements of matrix - line 3: 1 8 1 22 0 7
- Enter 6 space separated elements of matrix - line 4: 2 4 6 3 27 9
- Enter 6 space separated elements of matrix - line 5: 4 2 0 0 0 13
- Enter 6 space separated elements of answer matrix: 0 2 3 2 4 6
- Enter accuracy: 0.001
- x0: -0.17750430146679136 +- -7.1250964489733E-4
x1: 0.207096989772536 +- 1.1911208990744715E-4
x2: 0.06988937981071873 +- 9.443732444366915E-6
x3: -0.1335369679659826 +- -2.9291075935558575E-4
x4: -0.031442348182048156 +- -2.2938289627613337E-4
x5: 0.4842940943324688 +- 2.0090879998274636E-4
- Solution found in 36 iterations

Вывод

В ходе работы была реализована программа, решающая СЛАУ методом Гаусса-Зейделя. Программа проверяет матрицы на достижимость диагонального преобладания и приводит их к нему, после чего находит столбец решений с указанной точностью.

Метод Гаусса-Зейделя является модификацией метода Простых итераций. Отличие заключается в том, что на каждой итерации вычислений используются уже полученные на этой итерации X . Это позволяет значительно ускорить сходимость к решению системы. Недостатком метода является высокая трудоемкость (в том числе из-за необходимости проверки/приведения к диагональному преобладанию)