Уравнение теплопроводности

Вайцуль А. Н.

1 Математическое описание задачи

1. Уравнение теплопроводности

Задача решает уравнение теплопроводности в двумерной области:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$$

с заданными граничными условиями:

- $u(x, 0) = 0, u(x, 1) = 1;$
- $u(0, y) = 0, u(1, y) = 1;$
- Угловые точки $u(0, 0)$ и $u(1, 1)$ задаются вручную как 0.5.

2. Дискретизация

Уравнение решается с помощью метода сеток:

- Пространственная область $[0, 1] \times [0, 1]$ дискретизируется в $n \times n$ узлов равномерной сетки. Шаг $\Delta x = \Delta y = \frac{1}{n-1}$;
- Лапласиан $\nabla^2 u$ аппроксимируется конечными разностями:

$$\nabla^2 u \approx \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{\Delta x^2} + \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{\Delta y^2}$$

- Шаг по времени Δt выбирается в соответствии с критерием устойчивости:

$$\Delta t \leq \frac{\Delta x^2}{4}.$$

3. Обновление значений

Значения $u_{i,j}$ обновляются на каждом временном слое $n + 1$ согласно схеме:

$$u_{i,j}^{(n+1)} = u_{i,j}^{(n)} + \Delta t \cdot \nabla^2 u_{i,j}^{(n)}.$$

4. Итеративный процесс Итерации повторяются k раз. С каждом шагом решение приближается к стационарному состоянию ($\frac{\partial u}{\partial t} \to 0$).

5. Вывод разностного уравнения

Аппроксимация первой производной $\frac{\partial u}{\partial t}$:

Для производной по времени используем разложение Тейлора для функции $u(x, y, t)$ в окрестности временного слоя n:

$$u_{i,j}^{n+1} = u_{i,j}^{n} + \Delta t \frac{\partial u}{\partial t}\bigg|_{t^n} + \frac{\Delta t^2}{2} \frac{\partial^2 u}{\partial t^2}\bigg|_{t^n} + \dots$$

Пренебрегая членами порядка ($O(\Delta t^2)$) и выше, можем выразить первую производную:

$$\frac{\partial u}{\partial t}\bigg|_{t^n} \approx \frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t}.$$

Аппроксимация второй производной $\frac{\partial^2 u}{\partial x^2}$:

Рассмотрим разложение Тейлора для $u(x, y, t)$ по пространству x около узла i:

$$u_{i+1,j}^{n} = u_{i,j}^{n} + \Delta x \frac{\partial u}{\partial x}\bigg|_{x_i} + \frac{\Delta x^2}{2} \frac{\partial^2 u}{\partial x^2}\bigg|_{x_i} + ...,$$

$$u_{i-1,j}^{n} = u_{i,j}^{n} - \Delta x \frac{\partial u}{\partial x}\bigg|_{x_i} + \frac{\Delta x^2}{2} \frac{\partial^2 u}{\partial x^2}\bigg|_{x_i} + ...,$$

Сложим эти два разложения:

$$u_{i+1,j}^{n} + u_{i-1,j}^{n} = 2u_{i,j}^{n} + \Delta x^2 \frac{\partial^2 u}{\partial x^2}\bigg|_{x_i} + O(\Delta x^4).$$

Отсюда выражаем вторую производную:

$$\frac{\partial^2 u}{\partial x^2}\bigg|_{x_i} \approx \frac{u_{i+1,j}^{n} - 2u_{i,j}^{n} + u_{i-1,j}^{n}}{\Delta x^2}.$$

2 Алгоритм с точки зрения программирования

1. Инициализация данных

1. Ввод размерности матрицы n и количества итераций k.
2. Рассчитываем шаг по пространству $\Delta x = \Delta y = \frac{1}{(n-1)}$.
3. Шаг по времени Δt определяется как $\Delta t = \frac{\Delta x^2}{4}$ для выполнения критерия устойчивости.

4. Создается матрица $n \times n$, заполняется начальными значениями 0.5.

5. Устанавливаются граничные условия:

   - $u(0, x) = 1, u(1, x) = 0,$

   - $u(y, 0) = 0, u(y, 1) = 1,$

   - углы $u(0, 0)$ и $u(1, 1) = 0.5$.

2. Основной итеративный цикл

1. На каждой итерации создается копия массива arr для хранения нового временного слоя.

2. Для каждой внутренней точки (i, j) вычисляется лапласиан:

$$laplacian = \frac{arr[i-1, j] - 2 \cdot arr[i, j] + arr[i+1, j]}{\Delta x^2} + \frac{arr[i, j-1] - 2 \cdot arr[i, j] + arr[i, j+1]}{\Delta y^2}.$$

3. Вычисляется новое значение:

$$new_arr[i, j] = arr[i, j] + \Delta t \cdot laplacian.$$

4. Обновленный массив $new - arr$ заменяет текущий массив arr.

3. Визуализация

1. Используется библиотека $matplotlib$ для построения графика с изолиниями.

2. Значения матрицы визуализируются в виде цветных контуров с использованием карты $viridis$.

3. На графике отображаются: изолинии, числовые значения в каждой ячейке, сетка.

3 Результаты в виде изолиний

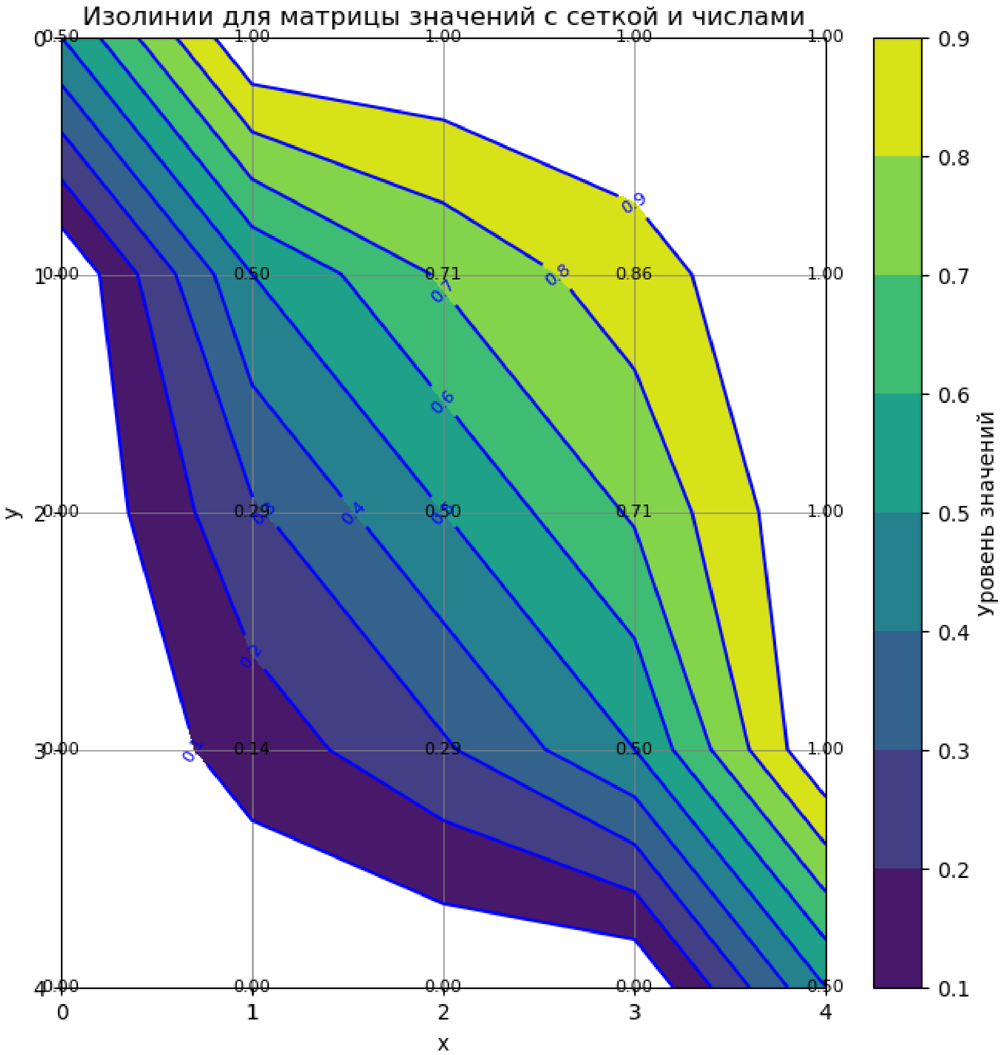

Рис. 1: Изолинии. Матрица(5×5). Итерации $= 50$

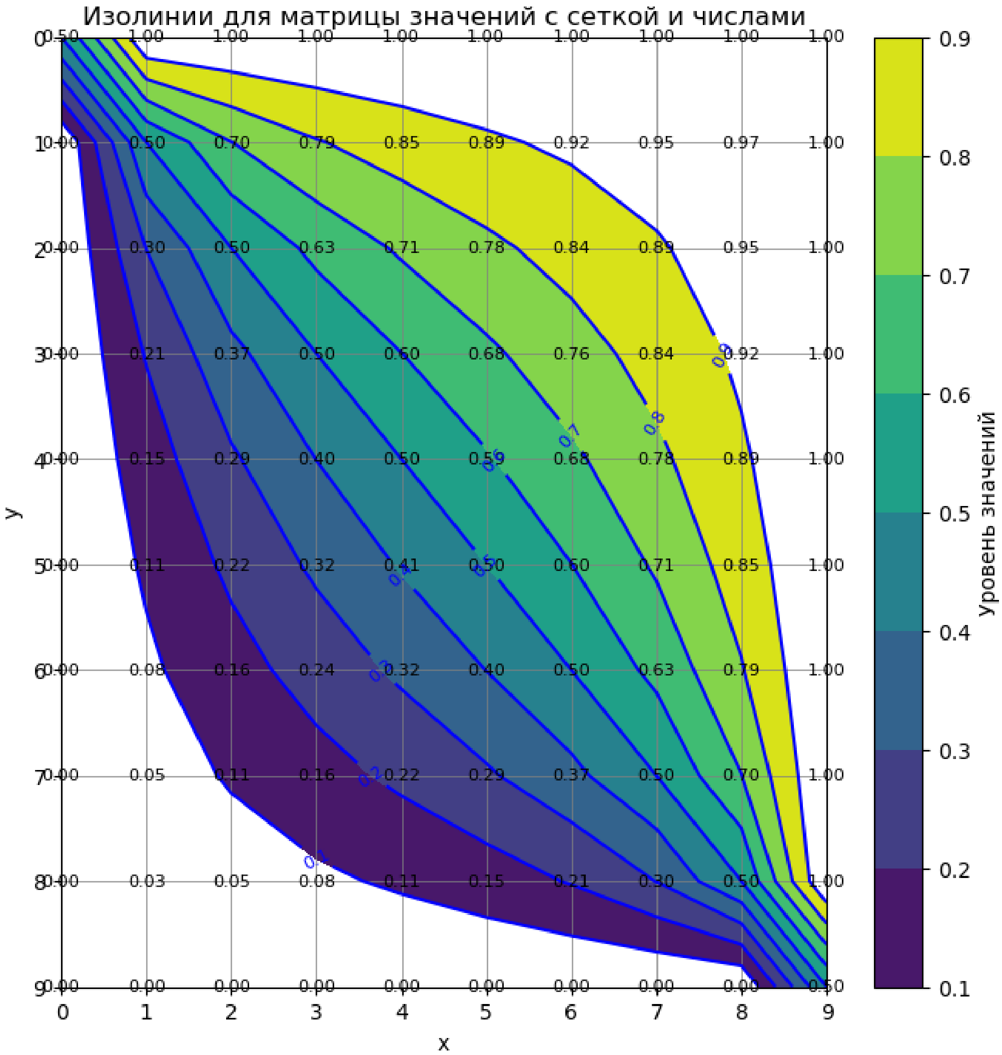

Рис. 2: Изолинии. Матрица(10×10). Итерации $= 200$

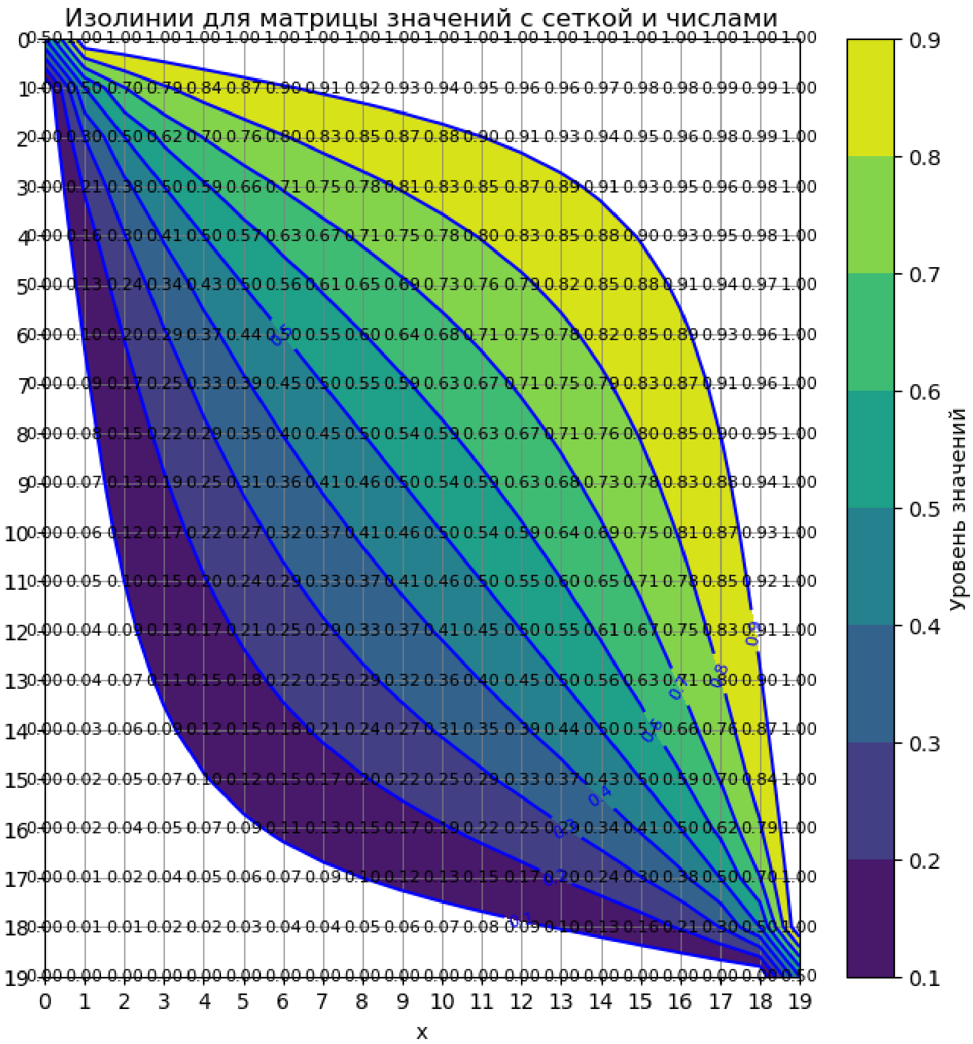

Рис. 3: Изолинии. Матрица(20×20). Итерации = 1000