

Задание к семинару №9

Решить однородное уравнение теплопроводности с граничными условиями Дирихле

$$\begin{cases} \frac{\partial u}{\partial t} = k \frac{\partial^2 u}{\partial x^2} \\ u(x, 0) = e^{-(x-5)^4} + 0.01x \\ u(0, t) = u(0, 0) \\ u(a, t) = u(a, 0) \end{cases} \quad (1)$$

на отрезке $x \in [0; a]$ при $t \in [0; T]$. Выбрать $a = 20$ и $T = 20$, а коэффициент теплопроводности $k = 2$. Шаг по пространству $h = 0.01$, шаг по времени $\tau = 0.05$. Расчет проводить с помощью комплексной схемы Розенброка. После каждого временного слоя выводить решение на текущем слое для получения анимационной картинки. Чтобы избежать постоянного изменения масштаба графика, рекомендуется после команды `plot` вставить команду `axis([0 a 0 1])`, не забыв далее поставить команду `pause(1e-6)`. Тот же расчет необходимо повторить с граничными условиями Неймана

$$u_x(0, t) = u_x(a, t) = 0. \quad (2)$$

Приложение

1. Счет будет идти быстрее, если использовать библиотеку SciPy для работы с разреженными матрицами. Например, функция `scipy.sparse.diags` позволяет создать разреженную матрицу с диагоналями, что значительно ускорит вычисления.
2. Граничные условия нужно включить в оператор пространственного дифференцирования Λ_x , видоизменив в нем первую и последнюю строки. Поскольку в данной задаче коэффициент теплопроводности постоянен, оператор Λ_x можно вычислить один раз до начала расчета.
3. Для аппроксимации граничных условий Неймана со 2 порядком следует использовать метод фиктивных точек.

Пример использования разреженных матриц в Python с использованием SciPy:

```
import numpy as np
import scipy.sparse as sp
```

```
N = 5
```

```
main_diag = 3 * np.ones(N)
```

```
off_diag = -1 * np.ones(N-1)
```

```
A = sp.diags([main_diag, off_diag, off_diag], [0, -1, 1], format='csr')
```

В этом примере:

- **main_diag**: массив, содержащий значения на главной диагонали матрицы.
- **off_diag**: массив, содержащий значения на диагоналях ниже и выше главной диагонали.
- **sp.diags([main_diag, off_diag, off_diag], [0, -1, 1], format='csr')**:
 - Первый аргумент: список массивов, каждый из которых содержит значения для соответствующей диагонали.
 - Второй аргумент: список целых чисел, указывающих позиции диагоналей относительно главной диагонали. 0 соответствует главной диагонали, -1 — диагонали ниже главной, 1 — диагонали выше главной.
 - **format='csr'**: указывает формат хранения разреженной матрицы (в данном случае, CSR — Compressed Sparse Row).