

Задание к семинару №8

Решить квазилинейное уравнение переноса

$$\begin{cases} \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0 \\ u(x, 0) = \frac{1}{1 + \left(\frac{x-50}{10}\right)^4} \\ u(0, t) = u(0, 0) \end{cases} \quad (1)$$

Задачу следует решать на отрезке $[0; 100]$ по пространству. Шаг равномерной сетки по пространству $h = 1$, шаг по времени $\tau = 0.05$. Число шагов по времени $N = 1000$.

Расчет вести с помощью консервативной чисто неявной схемы для квазилинейного уравнения переноса

$$\frac{\hat{u}_n - u_n}{\tau} + \frac{\hat{u}_n^2 - \hat{u}_{n-1}^2}{2h} = 0. \quad (2)$$

Приложение

Для рисования графиков и создания анимации в Python можно использовать библиотеку `matplotlib`. Для каждого временного шага необходимо вызывать функцию `plot` для построения графика решения на текущем временном слое. Для сохранения анимации можно использовать функцию `ani.save('animation.mp4', writer='ffmpeg', fps=30)`

Пример кода на Python для построения анимации:

```
%matplotlib notebook
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.animation as animation

fig, ax = plt.subplots()

ax.set_xlim(-1.5, 1.5)
ax.set_ylim(-1.5, 1.5)

line, = ax.plot([], [], lw=2)

def init():
    line.set_data([], [])
    return line,

def update(frame):
    theta = np.radians(frame)
    x = np.cos(theta)
```

```

y = np.sin(theta)
line.set_data([0, x], [0, y])
return line,

ani = animation.FuncAnimation(fig, update,
                              frames=np.arange(0, 361, 6),
                              init_func=init, blit=True)

plt.show()

```