

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Институт №8 «Компьютерные науки и прикладная математика»
Кафедра 806 «Вычислительная математика и программирование»

Лабораторная работа №6
по курсу «Численные методы»

Численное решение уравнений гиперболического типа.

Выполнил: *К. Д. Каширин*
Группа: *М8О-408Б-20*
Преподаватель: *Д. Е. Пивоваров*

Москва, 2023

Условие

1. Используя явную схему крест и неявную схему, решить начально-краевую задачу для дифференциального уравнения гиперболического типа. Аппроксимацию второго начального условия произвести с первым и со вторым порядком. Осуществить реализацию трех вариантов аппроксимации граничных условий, содержащих производные: двухточечная аппроксимация с первым порядком, трехточечная аппроксимация со вторым порядком, двухточечная аппроксимация со вторым порядком. В различные моменты времени вычислить погрешность численного решения путем сравнения результатов с приведенным в задании аналитическим решением $U(x, t)$. Исследовать зависимость погрешности от сеточных параметров τ, h .

2. Вариант 10:

$$\frac{\partial^2 u}{\partial t^2} + 3 \frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial u}{\partial x} - u - \cos x \exp(-t),$$

$$u_x(0, t) = \exp(-t),$$

$$u_x(\pi, t) = -\exp(-t),$$

$$u(x, 0) = \sin x,$$

$$u_t(x, 0) = -\sin x.$$

Аналитическое решение: $U(x, t) = \exp(-t) \sin x$..

Метод решения

Программа позволяет пользователю с помощью консольного ввода выбрать режим ввода параметров и метод решения гиперболического уравнения.

Сеточная функция представлена матрицей U размерности $K \times N$, где K — число временных слоёв, N — число пространственных шагов.

Явная конечно-разностная схема была записана в форме:

$$u_j^{k+1} = \{(\sigma + \mu) * u_{j+1}^k + (-2 * \sigma + 2 + c * \tau^2) * u_j^k + (\sigma - \mu) * u_{j-1}^k + (-1 + d * \frac{\tau}{2}) * u_j^{k-1} + \tau^2 * f(j * h, k * \tau)\} / \{1 + d * \frac{\tau}{2}\}.$$

Неявная конечно-разностная схема была записана в форме:

$$a_j u_{j-1}^{k+1} + b_j u_j^{k+1} + c_j u_{j+1}^{k+1} = d_j, j = 1 \dots N - 2, k = 0, 1, 2, \dots$$

где (для двухточечной аппроксимации с первым порядком)

$$a_j = \frac{b}{2h} - \frac{a}{h^2},$$

$$b_j = \frac{1}{\tau^2} + \frac{d}{2\tau} + 2 \frac{a}{h^2},$$

$$c_j = -\frac{b}{2h} - \frac{a}{h^2}, j = 1, \dots, N - 2.$$

$$a_0 = 0,$$

$$b_0 = \beta_0 - \frac{\alpha_0}{h},$$

$$c_0 = \frac{\alpha_0}{h},$$

$$a_{N-1} = -\frac{\alpha_I}{h},$$

$$b_{N-1} = \beta_I + \frac{\alpha_I}{h},$$

$$c_{N-1} = 0.$$

Аппроксимация второго начального условия первым порядком была записана следующей формулой:

$$u_j^I = \psi_0(j * h) + \psi_I(j * h) * \tau.$$

Аппроксимация второго начального условия вторым порядком была записана следующей формулой:

$$u_j^I = \psi_0(j * h) + \psi_I(j * h) * \left(\tau - d \frac{\tau^2}{2} \right) +$$

$$(a * \text{deriv2}(\psi_0, j * h) + b * \text{deriv}(\psi_0(j * h)) + c * \psi_0(j * h) + f(j * h, \tau)) * \frac{\tau}{2}$$

где

$f(x, t) = -\cos(x) * e^{-t}$ — свободный член уравнения.

Описание программы

Программа состоит из одного файла main.cpp, включающего функции граничных и начальных условий; функция свободного члена уравнения; функция для запуска явной конечно-разностной схемы, принимающая вид аппроксимации граничных условий; функция, осуществляющая двухточечную/трехточечную аппроксимацию с первым/вторым порядком; функция, осуществляющая аппроксимацию второго начального условия с первым/вторым порядком; функция метода прогонки для решения СЛАУ с трёхдиагональной матрицей; функция, осуществляющая двухточечную/трехточечную аппроксимацию с первым/вторым порядком и принимающая три массива, соответствующие диагоналям матрицы СЛАУ, а также вектор свободных коэффициентов; функция, вычисляющая погрешность; функция для расчёта первой/второй производной

Результаты

Для построения графиков функций (аналитического решения и численного) была написана программа на языке Python, использующая библиотеки numpy и matplotlib. Графики были построены для временного слоя timeSlice = 200, оранжевый цвет использовался для аналитического решения, чёрный — для численного.

```
k.d.kashirin@macbook-C02F11GJMD6M lab6 % g++ -std=c++11 main.cpp
k.d.kashirin@macbook-C02F11GJMD6M lab6 % ./a.out
Data written to file: explicit2Points1Bound1Init.txt
Data written to file: explicit2Points1Bound2Init.txt
Data written to file: explicit3Points2Bound1Init.txt
Data written to file: explicit3Points2Bound2Init.txt
Data written to file: explicit2Points2Bound1Init.txt
Data written to file: explicit2Points2Bound2Init.txt
Data written to file: implicit2Points1Bound1Init.txt
Data written to file: implicit2Points1Bound2Init.txt
Data written to file: implicit3Points2Bound1Init.txt
Data written to file: implicit3Points2Bound2Init.txt
Data written to file: implicit2Points2Bound1Init.txt
Data written to file: implicit2Points2Bound2Init.txt
```

Рис. 1. Консольное взаимодействие программы с пользователем.

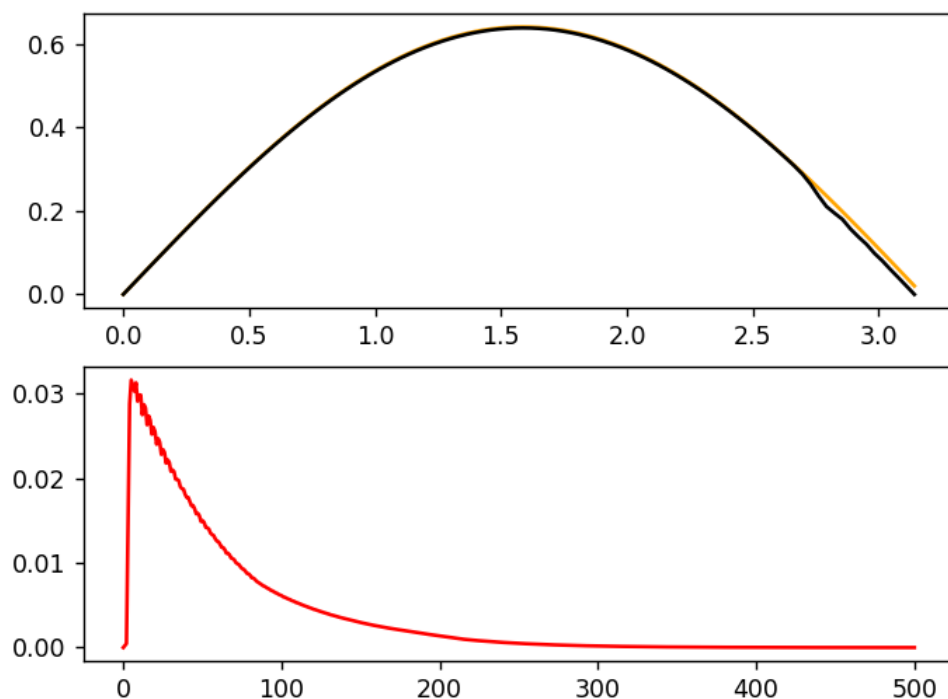


Рис. 2. График численного решения явной конечно-разностной схемой с двухточечной аппроксимацией второго порядка для граничных условий и аппроксимацией начального условия со вторым порядком.

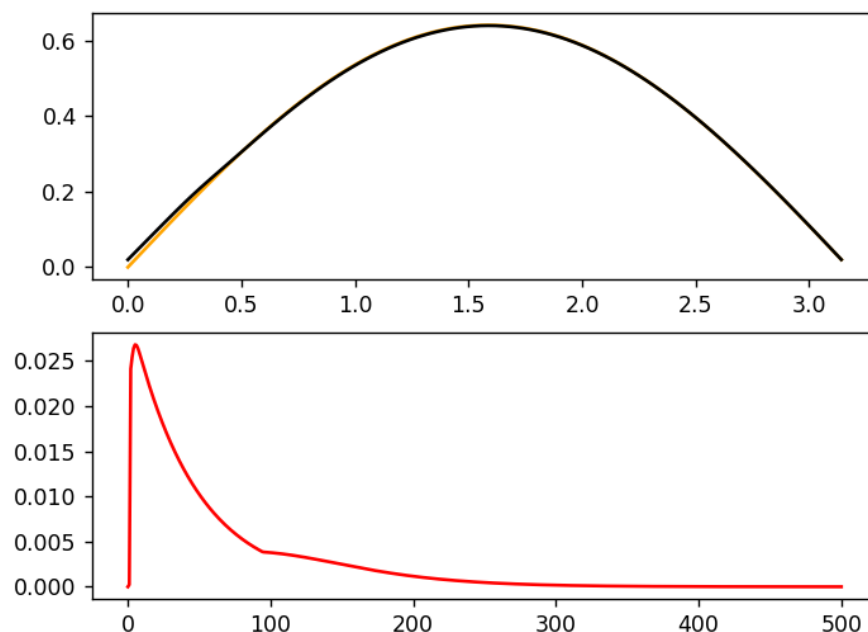


Рис. 3. График численного решения неявной конечно-разностной схемой с трёхточечной аппроксимацией второго порядка для граничных условий и аппроксимацией начального условия с первым порядком.

Вывод

В результате лабораторной работы по численному решению уравнений гиперболического типа, я освоил разнообразные методы, включая явные и неявные конечно-разностные схемы. Этот опыт позволил мне углубить знания в области моделирования физических процессов. Полученные умения могут оказаться востребованными не только для научных исследований, но и в разработке специализированного программного обеспечения, направленного на анализ и моделирование динамических систем.