Чисмодфизпроц.

1 Решение ОДУ

1.1 Разностны схемы для ОДУ и особенности аппроксимации

Рассмотрим задачу Коши для ОДУ, с заданными a, T, b и $f(t)$:

$$\dot{u} + au = f(t), \quad t \in [0, T], \quad u(0) = b. \tag{1.1}$$

Рассмотрим также две схемы ($\tau = T/N, t^n = n\tau, n = 0, ..., N$):

$$\frac{u^{n+1} - u^n}{\tau} + au^n = f(t^n), \quad n = 0, ..., N-1, \quad u^0 = b. \tag{1.2}$$

$$\frac{u^{n+1} - u^{n-1}}{2\tau} + au^n = f(t^n), \quad n = 1, ..., N-1, \quad u^0 = b, \quad \frac{u^1 - u^0}{\tau} + a\frac{u^1 + u^0}{2} = f(t^{1/2}). \tag{1.3}$$

Задача 1. *Решить уравнение* (1.1) *и схемы* (1.2)-(1.3) *аналитически, проанализировать результат. Посчитать порядок аппроксимации схем* (1.2)-(1.3).

Задача 2. *Запрограммировать схемы для разных* τ *и разных* a*, нарисовать графики и сравнить численные решения с точным.*

Задача 3. *Показать с помощью графиков, что схема* (1.3) *лучше, чем* (1.2) *(см. [**?**, гл.4, пар.8]). Показать, что схема* (1.2) *лучше, чем* (1.3) *(см. [**?**, гл.8, пар.7, стр.390-...]). Объяснить.*

1.2 Нахождение характеристик

Рассмотрим (заданную) функцию $c(x) > 0$ – скорость распространения линейных одномерных волн. Уравнение характеристик

$$\dot{x} = c(x), \quad x(0) = a \tag{1.4}$$

определяет функцию $X(t, a)$ (ее физический смысл: $X(t, a)$ – точка до которой дойдет возмущение из точки a за время t). Уравнение (1.4) разрешается в квадратурах и его решение $X(t, a)$ определяется неявно из уравнения:

$$t = T(X(t, a), a) := \int_a^{X(t,a)} \frac{dx}{c(x)}. \tag{1.5}$$

Задача 4. *Задать* $c(x)$*, для которой интеграл* $T(x, a)$ *в* (1.5) *берется в элементарных функциях и вычислить его (можно использовать* **Integrate[...]**[1]*). После этого разрешить уравнение* (1.5)

[1]Здесь и далее используются команды в Wolfram Mathematica

относительно X (аналитически, возможно с помощью **Solve[...]***). В результате получится аналитическое выражение для решения уравнения характеристик* (1.4)*. Построить графики $c(x)$, полученной функции $X(t,a)$ и ее производной по параметру $X_a(t,a)$ (см.* **Plot[...], Manipulate[...], D[...]***).*

Задача 5. *Написать программу для вычисления интеграла $T(x,a)$, обратной функции $X(t,a)$ с помощью базовых команд (***For[...], If[...], Sum[...]** *и т.п). Производную по параметру можно получить с помощью конечной разности, а для построения графика по точкам можно использовать* **ListLinePlot[...]**. *Сравнить полученную функцию с конечно-разностным решением задачи* (1.4) *напрямую, с помощью схем из п. 1.1 (на графике, ессно).*

Задача 6. *Повторить две предыдущие задачи, используя высокоуровневые процедуры пакета* **NIntegrate[...], NSolve[...], FindRoot[...], NDSolve[...]**. *Для получения гладкой функции (а также для вычисления обратной) можно использовать* **Interpolation[...]**. *Строить графики часто удобно в параметрическом виде:* **ParametricPlot[...]**.