6 Численное решение ОДУ

6.1 Решение нелинейной краевой задачи

Написать программу для численного решения краевой задачи:

$$u''(x) = u(2x^2 - 2\ln u - 2),\ u(0) = 1,\ u(1) = 1/e$$

с помощью конечно-разностного метода 2-го порядка

1. Программа должна содержать функцию, которая принимает на вход количество узлов и порог точности для остановки итераций в методе Ньютона, и возвращает массив значений решения в узлах сетки и координаты узлов сетки.

2. Фукнция должна решать нелинейную систему уравнений, полученную после конечно-разностной аппроксимации исходной задачи, методом Ньютона.

3. Для решения линейной системы на каждой итерации нужно использовать либо готовую либо собственную функцию для решения трехдиагональной системы.

4. Программа должна запускать расчет для сгущающихся равномерных сеток (шаг уменьшается вдвое), и строить в логарифмической шкале график зависимости ошибки от величины шага. За условно точное решение нужно взять численное решение на очень подробной сетке. График должен показывать, что порядок аппроксимации равен 2.

5. Программа должна выводить графики численного и точного решения для тестовой задачи.

6.2 Численное решение задачи Коши

Написать программу для решения задачи Коши для произвольной системы ОДУ:

$$\vec{u}_t = \vec{f}(t, \vec{u})$$
$$\vec{u}(0) = \vec{u}_0$$

методом Рунге-Кутты с таблицей Бутчера:

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}$$

Перед решением задачи нужно подставить коэффициенты метода в условия порядка и найти порядок метода.

Требования к программе:

1. Программа должна содержать функцию, принимающую на вход: отрезок $[0, T]$, вектор начального условия $\vec{u}_0$, ссылку на функцию $\vec{f}(t, \vec{u})$, ссылку на функцию для вычисления Якобиана $\frac{\partial \vec{f}}{\partial \vec{u}}$, количество шагов интегрирования, точность для критерия остановки итераций по нелинейности
2. Функция должна решать систему методом Рунге-Кутты, для решения нелинейного уравнения нужно использовать метод Ньютона.
3. Программа должна выводить графики численного и точного решения для тестовой задачи.
4. Программа должна делать расчёты на последовательности вложенных сеток с уменьшением шага вдвое, вычислять «фактический» порядок аппроксимации и строить график зависимости ошибки от шага сетки (или от числа узлов) в логарифмическом масштабе. *Фактический порядок должен быть близок к теоретическому на гладких решениях.*

6.3 Спектральный метод для линейной краевой задачи

Написать программу для численного решения линейной краевой задачи

$$u''(x) = f(x),\ u(-1) = a,\, u(1) = b$$

спектральным методом.

1. Программа должна содержать функцию, принимающую на вход число узлов и ссылку на функцию f.
2. Функция должна строить чебышёвскую сетку, и заполнять матрицу системы уравнений относительно неизвестных значений в узлах. В каждой внутреннем узле нужно использовать конечно-разностную формулу для 2-й производной, построенную по всем узлам сетки. Для решения линейной системы можно использовать готовую фукнцию.
3. Функция должна возвращать массив узлов и массив значений в узлах.
4. Программа должна строить для тестовой задачи с точным решением график зависимости ошибки от числа узлов в логарифмической шкале (нужно в цикле выполнять вычисления на сгущающихся сетках).
5. *Убедитесь, что в логарифмической шкале ошибка убывает быстрее, чем линейно. В противном случае, в программе есть ошибка.*
6. Программа должна выводить графики численного и точного решения для тестовой задачи.

6.4 Численное решение задачи Коши

Написать программу для решения задачи Коши для произвольной системы ОДУ:

$$\vec{u}_t = \vec{f}(t, \vec{u})$$
$$\vec{u}(0) = \vec{u}_0$$

методом Рунге-Кутты с таблицей Бутчера:

$$\begin{array}{c|cc} 1/4 & 1/4 & 0 \\ 3/4 & 1/2 & 1/4 \\ \hline & 1/2 & 1/2 \end{array}$$

Перед решением задачи нужно подставить коэффициенты метода в условия порядка и найти порядок метода.

Требования к программе:

1. Программа должна содержать функцию, принимающую на вход: отрезок $[0, T]$, вектор начального условия $\vec{u}_0$, ссылку на функцию $\vec{f}(t, \vec{u})$, ссылку на функцию для вычисления Якобиана $\frac{\partial \vec{f}}{\partial \vec{u}}$, количество шагов интегрирования, точность для критерия остановки итераций по нелинейности

2. Функция должна решать систему методом Рунге-Кутты, для решения нелинейного уравнения нужно использовать метод Ньютона.

3. Программа должна выводить графики численного и точного решения для тестовой задачи.

4. Программа должна делать расчёты на последовательности вложенных сеток с уменьшением шага вдвое, вычислять «фактический» порядок аппроксимации и строить график зависимости ошибки от шага сетки (или от числа узлов) в логарифмическом масштабе. *Фактический порядок должен быть близок к теоретическому на гладких решениях.*