Московский авиационный институт (национальный исследовательский университет)
Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

Курсовая работа по курсу “Численные методы”
“Численное решение жестких систем ОДУ с использованием неявных методов Рунге-Кутты”

Студент: Дондоков В. И.
Группа: М8О-409Б-20
Преподаватель: Пивоваров Д. Е.
Дата:
Оценка:

Москва, 2023

Теоретические сведения

Методы Рунге — Кутты большой класс численных методов решения задачи Коши для обыкновенных дифференциальных уравнений и их систем.

К классу методов Рунге — Кутты относятся явный метод Эйлера и модифицированный метод Эйлера с пересчётом, которые представляют собой соответственно методы первого и второго порядка точности. Существуют стандартные явные методы третьего порядка точности, не получившие широкого распространения. Наиболее часто используется и реализован в различных математических пакетах *классический метод Рунге — Кутты*, имеющий четвёртый порядок точности. При выполнении расчётов с повышенной точностью всё чаще применяются методы пятого и шестого порядков точности. Построение схем более высокого порядка сопряжено с большими вычислительными трудностями.

Рассмотрим задачу Коши для системы обыкновенных дифференциальных уравнений первого порядка.

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(x_0) = \mathbf{y}_0.$$

Тогда приближенное значение в последующих точках вычисляется по итерационной формуле:

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)$$

Вычисление нового значения проходит в четыре стадии:

$$\begin{aligned}
\mathbf{k}_1 &= \mathbf{f}\left(x_n, \mathbf{y}_n\right), \\
\mathbf{k}_2 &= \mathbf{f}\left(x_n + \frac{h}{2}, \mathbf{y}_n + \frac{h}{2}\mathbf{k}_1\right), \\
\mathbf{k}_3 &= \mathbf{f}\left(x_n + \frac{h}{2}, \mathbf{y}_n + \frac{h}{2}\mathbf{k}_2\right), \\
\mathbf{k}_4 &= \mathbf{f}\left(x_n + h, \mathbf{y}_n + h\,\mathbf{k}_3\right).
\end{aligned}$$

где h – величина шага сетки по x

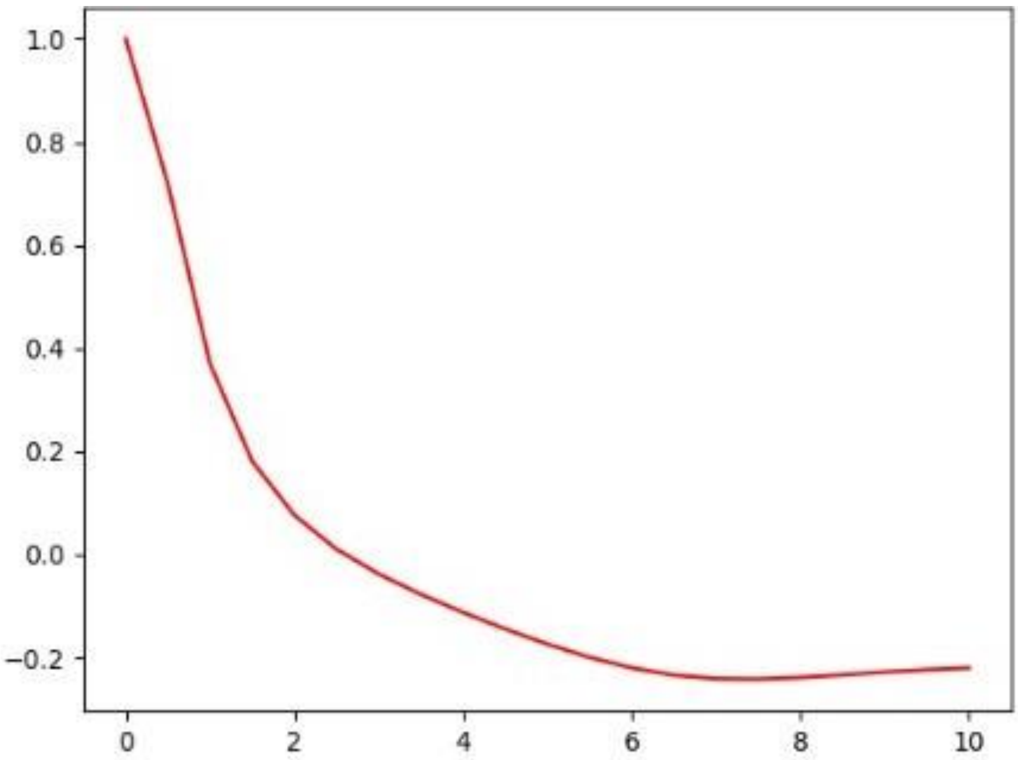
Исходный код

```
import matplotlib.pyplot as plt
from numpy import *



def runge_kutta(points, start_cond_0, start_cond_1, step):
    values = [start_cond_0]
    values_der = [start_cond_1]

    for i in range(0, len(points) - 1):
        x_k = points[i]
        y_k = values[i]
        z_k = values_der[i]

        k1_1 = step * f(x_k, y_k, z_k)
        k1_2 = step * g(x_k, y_k, z_k)

        k2_1 = step * f(x_k + 0.5 * step, y_k + 0.5 * k1_1, z_k + 0.5 * k1_2)
        k2_2 = step * g(x_k + 0.5 * step, y_k + 0.5 * k1_1, z_k + 0.5 * k1_2)

        k3_1 = step * f(x_k + 0.5 * step, y_k + 0.5 * k2_1, z_k + 0.5 * k2_2)
        k3_2 = step * g(x_k + 0.5 * step, y_k + 0.5 * k2_1, z_k + 0.5 * k2_2)

        k4_1 = step * f(x_k + step, y_k + k3_1, z_k + k3_2)
        k4_2 = step * g(x_k + step, y_k + k3_1, z_k + k3_2)

        dy_k = 1 / 6 * (k1_1 + 2 * k2_1 + 2 * k3_1 + k4_1)
        dz_k = 1 / 6 * (k1_2 + 2 * k2_2 + 2 * k3_2 + k4_2)

        values.append(y_k + dy_k)
        values_der.append(z_k + dz_k)

    print(values)
    plt.plot(points, values, color='r', label='aaaaa')
    plt.show()

    return values



def get_points(begin, end, step):
    return arange(begin, end, step).tolist() + [end]



def get_values(points, func):
    return [func(points[i]) for i in range(len(points))]



def f(x, y, z):
    return -0.04 * x + 1.4 * y * z
```

```
def g(x, y, z):
    return 0.04 * x - 1.4 * y * z - 3 * y**2



def main():
    interval = [0, 10]
    step = 0.5

    points = get_points(*interval, step)
    runge_kutta(points, 1, 0.1, step)



if __name__== '_main_':
    main()
```

Результат работы

[1, 0.7179010619471828, 0.3693744694120309, 0.18066372858053567, 0.07622471863691621, 0.010162120729374308, -0.03797358320650591, -0.07742328013051175, -0.11239783991615648, -0.14453503151595054, -0.17391202870854539, -0.19957106455102636, -0.22002694808671028, -0.23395470752892344, -0.2409351847590655, -0.24185385570922469, -0.238633599069128, -0.23347044513538145, -0.22810786865280172, -0.22353186620057713, -0.22006678706634097]



Вывод

Во время выполнения данной работы я изучил неявные методы Рунге-Кутты, а также жёсткие системы ОДУ, для решения которых они применяются. Используя полученные знания реализовал программу по нахождению численного решения жестких систем ОДУ с использованием неявных методов Рунге-Кутты.