Петербургский национальный исследовательский университет

информационных технологий, механики и оптики

Студент Гребцов Александр Андреевич Преподаватель Ракшин Егор Александрович

Группа R4150 Отчет принят

Отчет по практической работе №1

Оглавление

ОСНОВНАЯ ЧАСТЬ 3

1. Дифференциальное уравнение 3

2. Метод Эйлера и Рунге-Кутты 4

ЗАКЛЮЧЕНИЕ 6

ОСНОВНАЯ ЧАСТЬ

$$a = 6.26, \qquad b = 9.7, \qquad 2c = 5.4, \qquad d = 4.17$$

1. Дифференциальное уравнение

$$a \cdot \ddot{x} + b \cdot \dot{x} + c \cdot x = d$$

$$6.26\ddot{x} + 9.72\dot{x} + 5.4x = 4.17$$

Найдем корни характеристического уравнения

$$6.26\ddot{x} + 9.72\dot{x} + 5.4x = 0$$

$$\ddot{x} + 1.552\dot{x} + 0.863x = 0$$

$$\lambda^2 + 1.552\lambda + 0.863 = 0$$

$$\lambda = \frac{-1.552 \pm \sqrt{-1.043}}{2} = -0.776 \pm 0.511i$$

Система является устойчивой, так как отрицательная действительная часть комплексных корней приводит к затухающим колебаниям.

Общее решение однородного уравнения

$$x_0(t) = e^{-0.776}(C_1 \cos(0.511t) + C_2 \sin(0.511t))$$

Частное решение

$$cx_c = d$$

$$x_c = \frac{d}{c} = \frac{4.17}{5.4} = 0.772$$

Общее решение

$$x(t) = e^{-0.776t}(C_1 \cos(0.511t) + C_2 \sin(0.511t)) + 0.772$$

Начальные условия

$$x(0) = 0, \qquad \dot{x}(0) = 0, \qquad t = 0$$

$$0 = C_1 + 0.772$$

$$C_1 = -0.772$$

$$0 = -0.776C_1 + 0.511C_2$$

$$C_2 = -1.171$$

Аналитическое решение

$$x(t) = e^{-0.776t}(-0.772 \cos(0.511t) - 1.171 \sin(0.511t)) + 0.772$$

2. Метод Эйлера и Рунге-Кутты

```python
import numpy as np
import matplotlib.pyplot as plt

a = 6.26
b = 9.72
c = 5.4
d = 4.17

def linear(x):
    x1, x2 = x
    x1_dot = x2
    x2_dot = (d - b*x2 - c*x1) / a
    return np.array([x1_dot, x2_dot])

def forward_euler(fun, x0, Tf, h):
    t = np.arange(0, Tf + h, h)
    x_hist = np.zeros((len(x0), len(t)))
    x_hist[:, 0] = x0
    for k in range(len(t) - 1):
        x_hist[:, k + 1] = x_hist[:, k] + h * fun(x_hist[:, k])
    return x_hist, t

def backward_euler(fun, x0, Tf, h, tol=1e-8, max_iter=100):
    t = np.arange(0, Tf + h, h)
    x_hist = np.zeros((len(x0), len(t)))
    x_hist[:, 0] = x0
    for k in range(len(t) - 1):
        x_hist[:, k + 1] = x_hist[:, k]
        for i in range(max_iter):
            x_next = x_hist[:, k] + h * fun(x_hist[:, k + 1])
            error = np.linalg.norm(x_next - x_hist[:, k + 1])
            x_hist[:, k + 1] = x_next
            if error < tol:
                break
    return x_hist, t

def runge_kutta4(fun, x0, Tf, h):
    t = np.arange(0, Tf + h, h)
    x_hist = np.zeros((len(x0), len(t)))
    x_hist[:, 0] = x0
    for k in range(len(t) - 1):
        k1 = fun(x_hist[:, k])
        k2 = fun(x_hist[:, k] + 0.5 * h * k1)
        k3 = fun(x_hist[:, k] + 0.5 * h * k2)
        k4 = fun(x_hist[:, k] + h * k3)
        x_hist[:, k + 1] = x_hist[:, k] + (h / 6.0) * (k1 + 2*k2 + 2*k3 + k4)
    return x_hist, t

C1 = -0.772
C2 = -1.171
l1 = -0.776
l2 = 0.511
xp = 0.772

def equation(t):
    return np.exp(l1 * t) * (C1 * np.cos(l2 * t) + C2 * np.sin(l2 * t)) + xp

x0 = np.array([0.0, 0.0])
Tf = 10
h = 0.1

x_fe, t_fe = forward_euler(linear, x0, Tf, h)
x_be, t_be = backward_euler(linear, x0, Tf, h)
x_rk4, t_rk4 = runge_kutta4(linear, x0, Tf, h)
```

```python
x_an = equation(t_fe)

plt.figure(figsize=(24, 8))

plt.subplot(1, 3, 1)
plt.plot(t_fe, x_an, 'k', lw=2, label='Analytic')
plt.plot(t_fe, x_fe[0, :], '--', label='Forward Euler')
plt.plot(t_be, x_be[0, :], ':', label='Backward Euler')
plt.plot(t_rk4, x_rk4[0, :], '-.', label='RK4')
plt.xlabel('Time, t')
plt.ylabel('x(t)')
plt.title('x(t) vs Time')
plt.legend()
plt.grid(True)

plt.tight_layout()
plt.show()

```

График

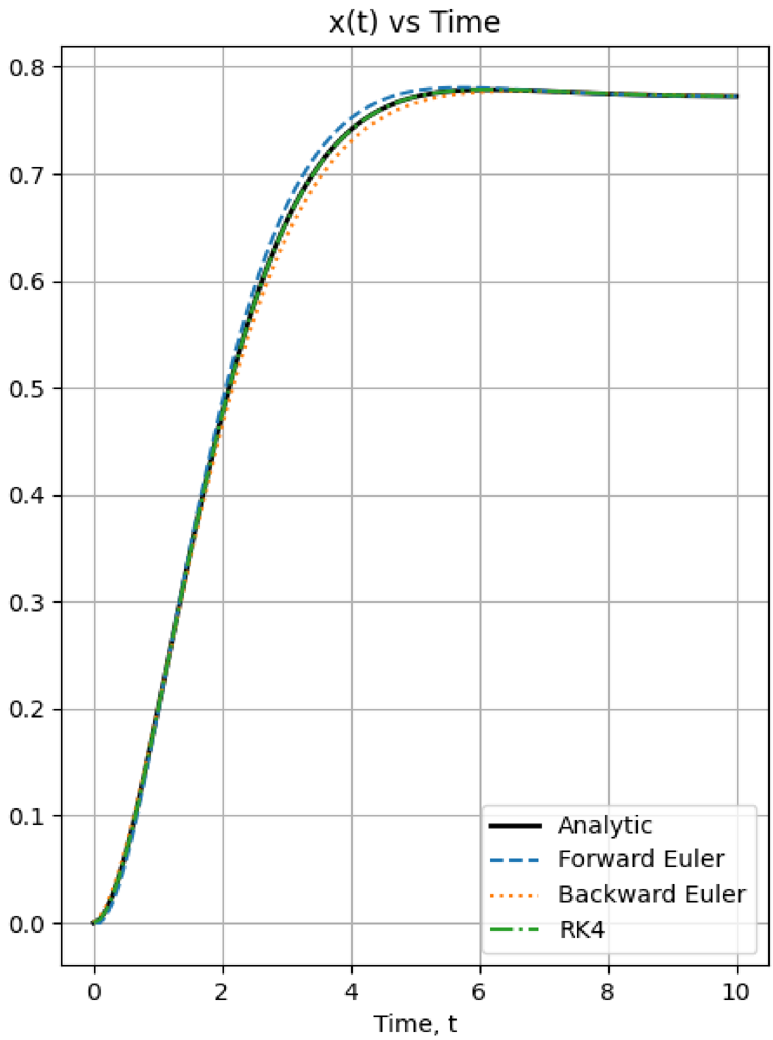

Рисунок 1. График $x(t)$ для аналитического решения и численных методов: Forward Euler, Backward Euler, метода Рунге-Кутта 4-го порядка

ЗАКЛЮЧЕНИЕ

В ходе выполнения практического работы было произведено сравнение численных методов интегрирования линейного дифференциального уравнения второго порядка - Forward Euler, Backward Euler, метода Рунге-Кутта 4-го порядка. Также построено аналитическое решение.

Был построен график, на основе которого была подтверждена устойчивость системы. Было определено, что метод Рунге-Кутта достаточно точно совпадает с аналитическим решением, в то время как метод Forward Euler имеет расхождение с аналитическим решением, также чуть в меньшей степени расхождение имеет Backward Euler.