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 №2

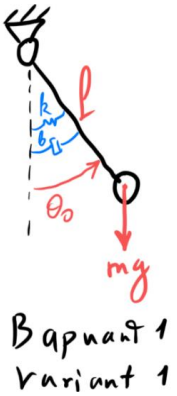
Оглавление

ОСНОВНАЯ ЧАСТЬ

$$m = 0.2\ kg, \qquad k = 3.6\ \frac{N}{m}, \qquad b = 0.3\frac{N*s}{m}, \qquad l = 0.96\ m$$

$$\theta_0 = 0.2827521206\ rad$$

1. Дифференциальное уравнение



Кинетическая энергия

$$K = \frac{1}{2}ml^2\dot{\theta}^2$$

Потенциальная энергия при малых углах

$$P_{mg} = mgl(1 - cos\theta) = \frac{1}{2}mgl\theta^2$$

$$P_k = \frac{1}{2}k\theta^2$$

$$P = P_{mg} + P_k = mgl\theta^2 + \frac{1}{2}k\theta^2 = \frac{1}{2}\theta^2(mgl + k)$$

Демпфирующий момент

$$Q = -b\dot{\theta}$$

Лагранжиан

$$L = K - P = \frac{1}{2}ml^2\dot{\theta}^2 - \frac{1}{2}\theta^2(mgl + k)$$

Уравнение Лагранжа

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = Q$$

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(ml^2\dot{\theta}\right) + (k + mgl)\theta = -b\dot{\theta}$$

$$ml^2\ddot{\theta} + b\dot{\theta} + (k + mgl)\theta = 0$$

$$\ddot{\theta} + \frac{b}{ml^2}\dot{\theta} + \frac{(k + mgl)}{ml^2}\theta = 0$$

$$\omega = \frac{(k + mgl)}{ml^2}, \qquad \mathrm{c} = \frac{b}{ml^2}, \qquad \zeta = \frac{\mathrm{c}}{2\omega} = \frac{b}{2ml^2\omega}$$

Аналитическое решение при начальных условиях $\theta(0) = \theta_0,\ \dot{\theta}(0) = \dot{\theta}_0$

1. Колебание

$$\zeta = \frac{b}{2ml^2\omega} < 1$$

Корни характеристического уравнения

$$\mathrm{C}_1 = \theta_0$$

$$\mathrm{C}_2 = \frac{\dot{\theta}_0 + \zeta\omega\theta_0}{\omega\sqrt{1-\zeta^2}}$$

$$\theta(t) = e^{-\omega t}\left(\theta_0 + t\left(\dot{\theta}_0 + \omega\theta_0\right)\right)$$

$$\theta(t) = e^{-\zeta\omega t}\left(\theta_0 \cos\left(\omega t\sqrt{1-\zeta^2}\right) + \frac{\dot{\theta}_0 + \zeta\omega\theta_0}{\omega\sqrt{1-\zeta^2}} \sin\left(\omega t\sqrt{1-\zeta^2}\right)\right)$$

2. Критическое затухание

$$\zeta = \frac{b}{2ml^2\omega} = 1$$

Корни характеристического уравнения

$$\lambda_{1,2} = -\omega$$

$$\mathrm{C}_1 = \theta_0$$

$$\mathrm{C}_2 = \dot{\theta}_0 + \omega\theta_0$$

$$\theta(t) = e^{-\omega t}\left(\theta_0 + t\left(\dot{\theta}_0 + \omega\theta_0\right)\right)$$

3. Перезатухание

$$\zeta = \frac{b}{2ml^2\omega} > 1$$

Корни характеристического уравнения

$$\lambda_1 = -\omega\left(\zeta + \sqrt{\zeta^2 - 1}\right)$$

$$\lambda_1 = -\omega\left(\zeta - \sqrt{\zeta^2 - 1}\right)$$

$$C_1 = \theta_0 - \frac{\lambda_1\theta_0 - \dot{\theta}_0}{\lambda_1 - \lambda_2}$$

$$C_2 = \frac{\lambda_1\theta_0 - \dot{\theta}_0}{\lambda_1 - \lambda_2}$$

$$\theta(t) = e^{-\lambda_1 t}\left(\theta_0 - \frac{\lambda_1\theta_0 - \dot{\theta}_0}{\lambda_1 - \lambda_2}\right) + e^{-\lambda_2 t}\frac{\lambda_1\theta_0 - \dot{\theta}_0}{\lambda_1 - \lambda_2}$$

2. Метод Эйлера и Рунге-Кутты

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

m = 0.2
k = 3.6
b = 0.3
l = 0.96
g = 9.81

def tetta_sol(tetta, omega1, t):
    omega = np.sqrt((k + m * g *l) / (m * l**2))
    dzetta = b / (m * l**2) / (2 * omega)
    if dzetta  < 1:
        C1 = tetta
        C2 = (omega1 + dzetta  * omega * tetta) / omega * np.sqrt(1 - dzetta **2)
        return np.exp(-dzetta  * omega * t) * (C1 * np.cos(omega * np.sqrt(1 - dzetta **2) * t)
                                               + C2 * np.sin(omega * np.sqrt(1 - dzetta **2) * t))
    elif dzetta == 1:
        C1 = tetta
        C2 = omega1 + omega * tetta
        return (C1 + C2 * t) * np.exp(-omega * t)
    else:
        l1 = -omega * (dzetta  + np.sqrt(dzetta **2 - 1))
        l2 = -omega * (dzetta  - np.sqrt(dzetta **2 - 1))
        C2 = (l2 * tetta - omega1) / (l1 - l2)
        C1 = tetta - C2
        return C1 * np.exp(l1 * t) + C2 * np.exp(l2 * t)

def pendulum_system(state):
    tetta = state[0]
    omega = state[1]

    dtetta = omega
    domega = -b / (m * l**2) * omega - ((k + m * g * l) / (m * l**2)) * tetta

    return np.array([dtetta, domega])

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

def equation(t):
    return tetta_sol(tetta0, omega0, t_fe)

tetta0 = 0.2827521206
omega0 = 0
x0 = np.array([tetta0, omega0])

Tf = 5
h = 0.01

```

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

x_an = equation(t_fe)

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

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('tetta(t)')
plt.title('tetta(t) vs Time')
plt.legend()
plt.grid(True)

plt.tight_layout()
plt.show()
```

График

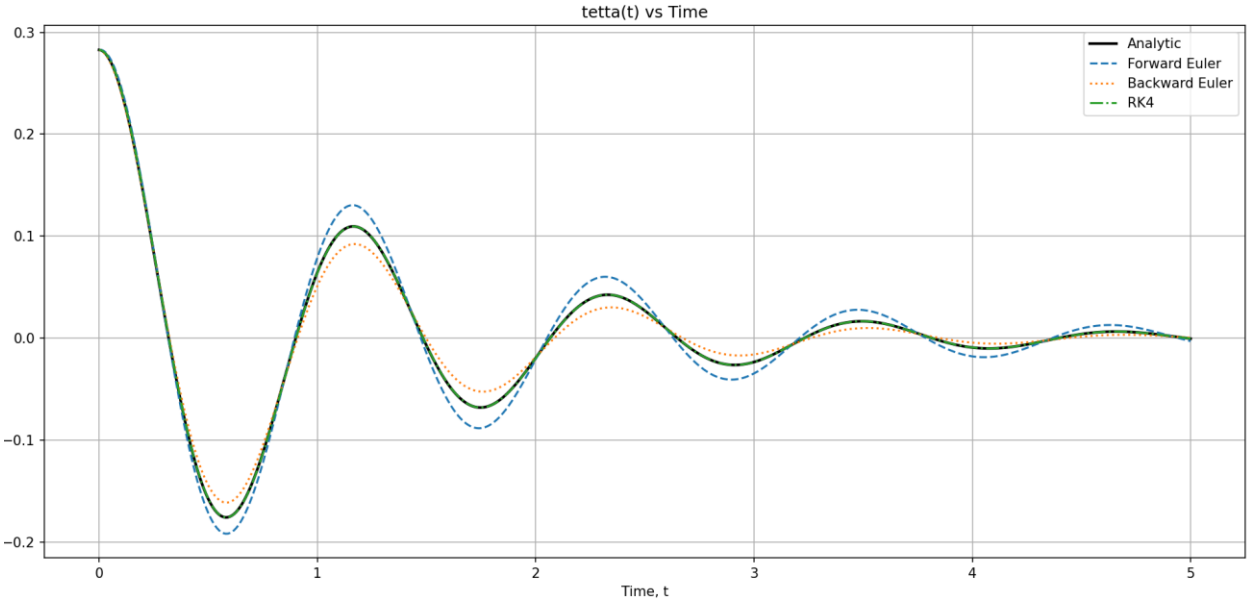


Рисунок 1. График $\theta(t)$ для аналитического решения и численных методов: Forward Euler, Backward Euler, метода Рунге-Кутта 4-го порядка

ЗАКЛЮЧЕНИЕ

В ходе выполнения практической работы было выведено уравнение системы, состоящей из маятника, пружины и демпфера. Произведено сравнение численных методов интегрирования линейного дифференциального уравнения - Forward Euler, Backward Euler, метода Рунге-Кутта 4-го порядка. Также построено аналитическое решение.

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.