ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

**ОТЧЕТ**

**О ВЫПЛОНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

**«ДИНАМИКА СИСТЕМЫ»**

**ПО ДИСЦИПЛИНЕ «ТЕОРЕТИЧЕСКАЯ МЕХАНИКА И ОСНОВЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ»**

**ВАРИАНТ ЗАДАНИЯ №27**

Выполнил(а) студент группы М8О-201Б-22

Чибугаев И.А.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Проверил и принял

Волков В.Е.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Москва, 2024

*Задание:* проинтегрировать систему дифференциальных уравнений движения системы с двумя степенями свободы с помощью средств Python. Построить анимацию движения системы, а также графики законов движения системы и указанных в задании реакций для разных случаев системы.

*Код:*

import numpy as np

import sympy as sp

import matplotlib.pyplot as plt

from scipy.integrate import solve\_ivp

*# Константы системы*

R = 5.0  *# радиус большой шестерни*

r = 1.5  *# радиус маленькой шестерни*

g = 9.81  *# ускорение свободного падения*

m1, m2, m3 = 10.0, 5.0, 2.0  *# массы*

c = 50.0  *# жесткость пружины*

M1, M2 = 0.0, 0.0  *# моменты*

*# Переменные*

phi, phi\_dot, phi\_ddot = sp.symbols('phi phi\_dot phi\_ddot')

theta, theta\_dot, theta\_ddot = sp.symbols('theta theta\_dot theta\_ddot')

*# Лагранжиан*

T\_rot1 = (1 / 2) \* (m1 + m2) \* (R\*\*2) \* theta\_dot\*\*2

T\_rot2 = (1 / 2) \* m2 \* (R - r)\*\*2 \* phi\_dot\*\*2

T\_crank = (1 / 6) \* m3 \* (R - r)\*\*2 \* phi\_dot\*\*2

T = T\_rot1 + T\_rot2 + T\_crank

V\_potential = m2 \* g \* (R - r) \* sp.sin(phi) + (1 / 2) \* c \* phi\*\*2

V = V\_potential

L = T - V

*# Уравнения Лагранжа*

L\_theta = sp.diff(sp.diff(L, theta\_dot), 't') - sp.diff(L, theta)

L\_phi = sp.diff(sp.diff(L, phi\_dot), 't') - sp.diff(L, phi)

*# Упрощение уравнений*

eq\_theta = sp.simplify(L\_theta).subs('t', 0)  *# Убираем временные производные*

eq\_phi = sp.simplify(L\_phi).subs('t', 0)

*# Решение системы уравнений*

a11 = eq\_phi.coeff(phi\_ddot, 1)

a12 = eq\_phi.coeff(theta\_ddot, 1)

a21 = eq\_theta.coeff(phi\_ddot, 1)

a22 = eq\_theta.coeff(theta\_ddot, 1)

b1 = -eq\_phi.subs([(phi\_ddot, 0), (theta\_ddot, 0)])

b2 = -eq\_theta.subs([(phi\_ddot, 0), (theta\_ddot, 0)])

*# Числовые функции*

phi\_ddot\_func = sp.lambdify((phi, phi\_dot, theta, theta\_dot), a11 \* a22 - a12 \* a21)

theta\_ddot\_func = sp.lambdify((phi, phi\_dot, theta, theta\_dot), b2 - b1)

*# Численное решение системы*

def equations(t, y):

    phi, phi\_dot, theta, theta\_dot = y

    phi\_ddot\_val = phi\_ddot\_func(phi, phi\_dot, theta, theta\_dot)

    theta\_ddot\_val = theta\_ddot\_func(phi, phi\_dot, theta, theta\_dot)

    return [phi\_dot, phi\_ddot\_val, theta\_dot, theta\_ddot\_val]

*# Начальные условия*

y0 = [0, 0, 0, 0]  *# phi, phi\_dot, theta, theta\_dot*

t\_span = (0, 10)

t\_eval = np.linspace(\*t\_span, 1000)

solution = solve\_ivp(equations, t\_span, y0, t\_eval=t\_eval)

*# Построение графиков*

phi\_values = solution.y[0]

theta\_values = solution.y[2]

plt.figure(figsize=(10, 6))

plt.plot(t\_eval, phi\_values, label=r'$\phi(t)$')

plt.plot(t\_eval, theta\_values, label=r'$\theta(t)$')

plt.xlabel('Время (с)')

plt.ylabel('Углы (рад)')

plt.title('Движение системы')

plt.legend()

plt.grid()

plt.show()

*Скриншот:*

