Московский авиационный институт

(национальный исследовательский университет)

Институт № 8 «Информационные технологии и прикладная математика»

**Лабораторная работа №3**

**по курсу «Теоретическая механика»**

**Динамика системы**

Выполнил студент группы М8О-207Б-21

Павлов Иван Дмитриевич

Преподаватель: Беличенко Михаил Валериевич

Оценка:

Дата:

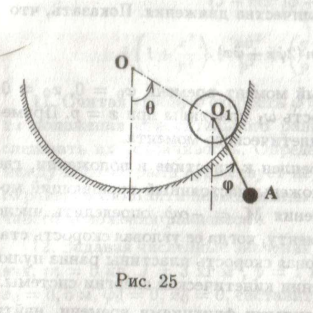
Москва, 2022

**Вариант №25**

**Задание:**

Численно решить дифференциальные уравнения движения механической системы в среде Octave (или Matlab), сделать задание №12 курсовой и построить анимацию движения системы.

**Механическая система:**

****

**Текст программы**

import math

import numpy as np

from scipy.integrate import odeint

from matplotlib.animation import FuncAnimation

import matplotlib.pyplot as plt

def Lagrange(y, t, m1, m2, l, g, r, R):

dy = np.zeros(4)

dy[0] = y[2]

dy[1] = y[3]

a11 = m2 \* l \* np.cos(y[0] - y[1])

a12 = ((3 / 2) \* m1 + m2) \* (R - r)

a21 = l

a22 = (R - r) \* np.cos(y[0] - y[1])

b1 = -(m1 + m2) \* g \* np.sin(y[1]) + m2 \* l \* y[2] \*\* 2 \* np.sin(y[0] - y[1])

b2 = -g \* np.sin(y[0]) - (R-r) \* y[3] \*\* 2 \* np.sin(y[0] - y[1])

dy[2] = (b1 \* a22 - b2 \* a12) / (a11 \* a22 - a12 \* a21)

dy[3] = (b2 \* a11 - b1 \* a21) / (a11 \* a22 - a12 \* a21)

return dy

l = 0.65

m1 = 5

m2 = 3.5

r = 0.2

R = 1

g = 9.80665

T = np.linspace(0, 50, 1000)

phi0 = 0

theta0 = math.pi / 4

dphi0 = 0

dtheta0 = 0

y0 = [phi0, theta0, dphi0, dtheta0]

Y = odeint(Lagrange, y0, T, (m1, m2, l, g, r, R))

phi = Y[:, 0]

theta = Y[:, 1]

def sector(x, y, r):

cx = [x + r \* np.cos(i / 100) for i in range(314, 628)]

cy = [y + r \* np.sin(i / 100) for i in range(314, 628)]

return (cx, cy)

def circle(x, y, r):

cx = [x + r \* np.cos(i / 100) for i in range(0, 628)]

cy = [y + r \* np.sin(i / 100) for i in range(0, 628)]

return (cx, cy)

t = np.linspace(0, 10, 1001)

X0 = 0

Y0 = 0

X1 = (R - r) \* np.sin(theta)

Y1 = -(R - r) \* np.cos(theta)

X2 = X1 + l \* np.sin(phi)

Y2 = Y1 - l \* np.cos(phi)

Vx2 = np.diff(X2)

Vy2 = np.diff(Y2)

Wx2 = np.diff(Vx2)

Wy2 = np.diff(Vy2)

fig = plt.figure()

ax = fig.add\_subplot(1, 1, 1)

ax.axis('equal')

ax.set(xlim=[-1.5, 1.5], ylim=[-3, 1.5])

sector1, = ax.plot(\*sector(X0, Y0, R), 'red')

circle1, = ax.plot(\*circle(X1[0], Y1[0], r), 'blue')

circle2, = ax.plot(\*circle(X2[0], Y2[0], 0.1), 'green')

point = ax.plot(X0, Y0, marker='o', color='black')[0]

point1 = ax.plot(X1, Y1, marker='o', color='black')[0]

point2 = ax.plot(X2, Y2, marker='o', color='black')[0]

line1 = ax.plot([X1[0], X2[0]], [Y1[0], Y2[0]], color='black')[0]

line2 = ax.plot([X0, X1[0]], [Y0, Y1[0]], linestyle = '--', linewidth = 1, color='black')[0]

line3 = ax.plot([X1[0], X1[0]], [Y1[0], Y1[0] - r - 0.01], linestyle = '--', linewidth = 1, color='black')[0]

def kadr(i):

circle1.set\_data(\*circle(X1[i], Y1[i], r))

circle2.set\_data(\*circle(X2[i], Y2[i], 0.1))

sector1.set\_data(\*sector(X0, Y0, R))

point.set\_data(X0, Y0)

point1.set\_data(X1[i], Y1[i])

point2.set\_data(X2[i], Y2[i])

line1.set\_data([X1[i], X2[i]], [Y1[i], Y2[i]])

line2.set\_data([X0, X1[i]], [Y0, Y1[i]])

line3.set\_data([X1[i], X1[i]], [Y1[i], Y1[i] - r - 0.01])

return [circle1, point, point2, line1, sector1, point1]

model = FuncAnimation(fig,

kadr,

interval=(T[1] - T[0]) \* 1000,

frames=len(T))

plt.show()

**Результат работы:**

