

Московский авиационный институт
(национальный исследовательский университет)
Институт № 8 «Компьютерные науки и прикладная математика»

**Лабораторная работа № 2
по курсу "Теоретическая механика
и компьютерное моделирование"**

Кинематика системы

Выполнил студент группы М8О-206Б-21
Синюков А.С.
Преподаватель: Чекина Евгения Алексеевна

Оценка:
Дата: 21.12.2022

Вариант № 24

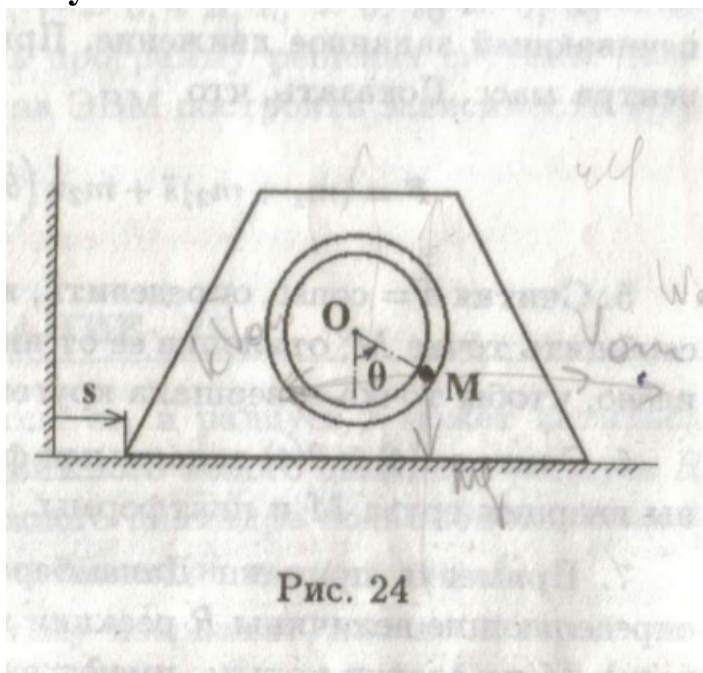
Задание:

Невесомая трубка, выгнутая в форме кругового кольца радиуса r , закреплена на платформе, которая имеет массу m_1 и может скользить без трения по горизонтальной плоскости.

В трубке движется без трения точечный груз M массы m_2 .

Ввести подвижную систему координат, связанную с платформой. Считая $s(t)$ и $\theta(t)$ заданными функциями времени, вычислить абсолютную скорость и абсолютное ускорение точки M . Изобразить на чертеже составляющие векторов V_{abs} и W_{abs}

Рисунок:



Текст программы:

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.animation import FuncAnimation
import sympy as sp
import math

def Platform(x0, y0): #линия в форме трапеции из 5 точек
    PX = [x0 - 10, x0 - 5, x0 + 5, x0 + 10, x0 - 10]
    PY = [y0 - 7.5, y0 + 10, y0 + 10, y0 - 7.5, y0 - 7.5]
    return PX, PY

# Составление с помощью sympy законов движения, скоростей и ускорений
t = sp.Symbol('t')
s = 4 * sp.cos(3 * t)
phi = 4 * sp.sin(t - 10)

x = s * sp.cos(math.pi) + 0.8
y = -s * sp.sin(math.pi) + 7.5
```

```

vx = sp.diff(x, t)
vy = sp.diff(y, t)

vmod = sp.sqrt(vx * vx + vy* vy)

wx = sp.diff(vx, t)
wy = sp.diff(vy, t)

wmod = sp.sqrt(wx * wx + wy* wy)

xa = x - 5 * sp.sin(phi)
ya = y + 5 * sp.cos(phi)

# Рассчеты движений и графиков скоростей и ускорений
T = np.linspace(0, 20, 1000)

X_def = sp.lambdify(t, x)
Y_def = sp.lambdify(t, y)
V_def = sp.lambdify(t, vmod)
W_def = sp.lambdify(t, wmod)
XA_def = sp.lambdify(t, xa)
YA_def = sp.lambdify(t, ya)
Cord_def = sp.lambdify(t, t)

X = X_def(T)
Y = Y_def(T)
V = V_def(T)
W = W_def(T)
XA = XA_def(T)
YA = YA_def(T)
Cord = Cord_def(T)

fig = plt.figure(figsize = (20, 10))

# Строим саму платформу с призмой и точкой
ax1 = fig.add_subplot(1, 2, 1)
ax1.axis('equal')
ax1.set(xlim=[-20, 20], ylim=[-20, 30])
ax1.plot([X.min() - 10, X.max() + 10], [0, 0], 'black')
ax1.plot([X.min() - 10, X.min() - 10], [0, Y.max() + 15], 'black')

PrX, PrY = Platform(X[0], Y[0])
Prism = ax1.plot(PrX, PrY, 'blue')[0]

radius, = ax1.plot([X[0], XA[0]], [Y[0], YA[0]], 'black')

Phi = np.linspace(0, 6.28, 20)
r = 0.2
Point = ax1.plot(XA[0] + r * np.cos(Phi), YA[0] + r * np.sin(Phi), 'blue')[0]

```

```

# Строим графики
ax2 = fig.add_subplot(4, 2, 2)
ax2.set(xlim=[0, 15], ylim=[0, 15])
Vgx = [Cord[0]]
Vgy = [V[0]]
V_graph, = ax2.plot(Vgx, Vgy, 'blue')
ax2.set_ylabel('V')

ax3 = fig.add_subplot(4, 2, 4)
ax3.set(xlim=[0, 15], ylim=[0, 40])
Wgx = [Cord[0]]
Wgy = [W[0]]
W_graph, = ax3.plot(Wgx, Wgy, 'orange')
ax3.set_ylabel('W')

plt.subplots_adjust(wspace = 0.2, hspace = 0.2)

# Анимлируем
def anima(i):
    PrX, PrY = Platform(X[i], Y[i])
    Prism.set_data(PrX, PrY)
    radius.set_data([X[i], XA[i]], [Y[i], YA[i]])
    Point.set_data(XA[i] + r * np.cos(Phi), YA[i] + r * np.sin(Phi))
    Vgx.append(Cord[i])
    Vgy.append(V[i])
    Wgx.append(Cord[i])
    Wgy.append(W[i])
    V_graph.set_data(Vgx, Vgy)
    W_graph.set_data(Wgx, Wgy)
    return Prism, radius, Point, V_graph, W_graph

anim = FuncAnimation(fig, anima, frames = 1000, interval = 1, blit = True)

plt.show()

```

Результат работы программы:

