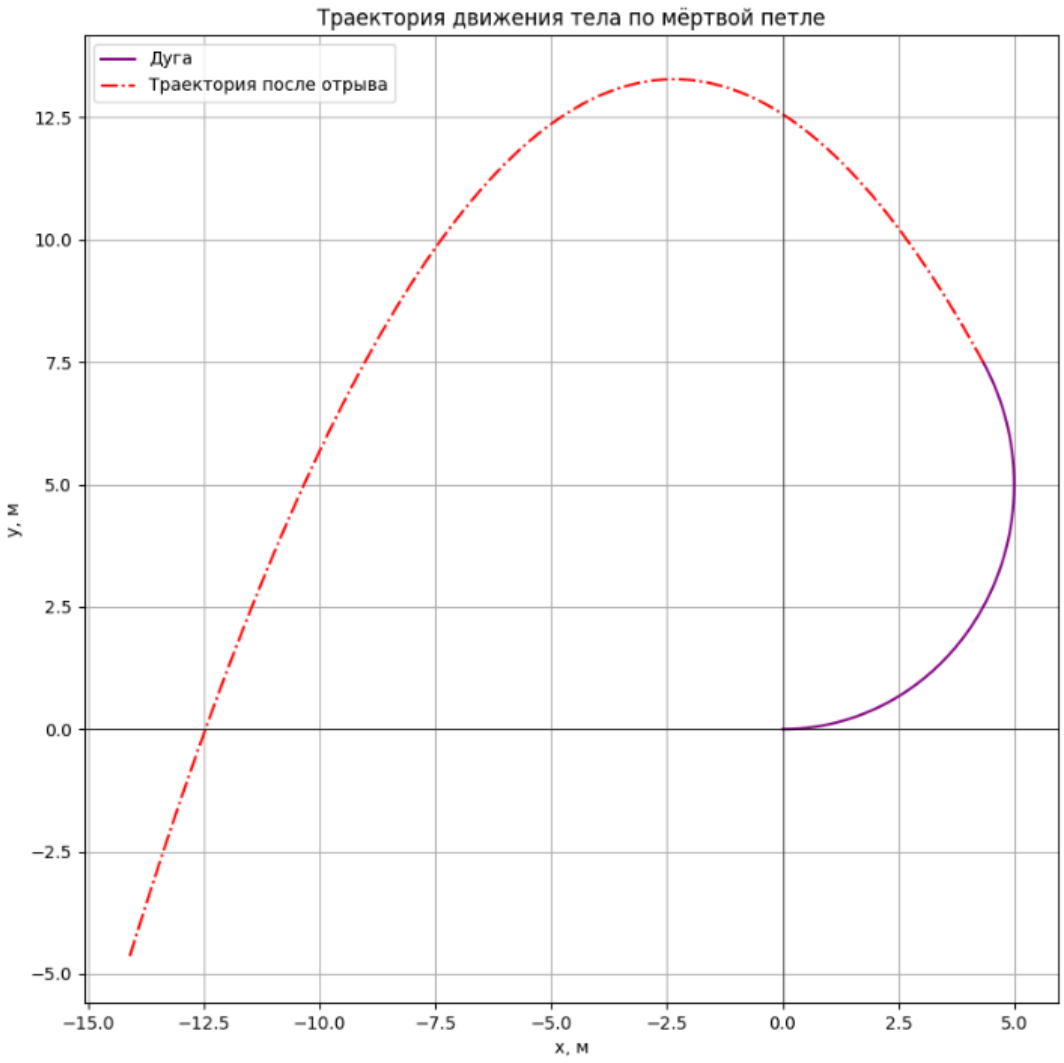
Отчет по физике
Моделирование_Практика_1
Тимофеев В[redacted] М3212

Код:

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

# Параметры
m = 2.0 # сокращается
R = 5.0
a = np.pi/2 + np.pi/6
g = 9.81
k_fr = 0.02

# Минимальная начальная скорость для прохождения всей дуги
E_pot = g * R * (1 - np.cos(a))
F_fr = k_fr * g * R * a
v0 = np.sqrt(2 * E_pot + 2 * F_fr)

# Визуализация дуги
theta_values = np.linspace( start: 0, a,  num: 1000)
Arc_x = R * np.sin(theta_values)
Arc_y = -R * np.cos(theta_values) + R

# Визуализазция тракетории после отрыва от дуги
vx_AfterSeparation = v0 * np.cos(a)
vy_AfterSeparation = v0 * np.sin(a)

FlyTime_values = np.linspace( start: 0,  stop: 3,  num: 1000)
Flying_x = Arc_x[-1] + vx_AfterSeparation * FlyTime_values
Flying_y = Arc_y[-1] + vy_AfterSeparation * FlyTime_values - 0.5 * g * FlyTime_values**2

# График
plt.figure(figsize=(10, 10))
plt.plot( *args: Arc_x, Arc_y, label="Дуга", color="purple")
plt.plot( *args: Flying_x, Flying_y, label="Траектория после отрыва", color="red", linestyle="-.")

# Оформление графика
plt.xlabel("x, м")
plt.ylabel("y, м")
plt.axhline( y: 0, color="black", linewidth=0.5)
plt.axvline( x: 0, color="black", linewidth=0.5)
plt.legend()
plt.title("Траектория движения тела по мёртвой петле")
plt.grid()
plt.show()
```

Визуализация:



Введение

В данном отчёте представлен код для моделирования движения тела по мёртвой петле с учетом трения. Код включает расчёт минимальной начальной скорости, чтобы тело могло пройти всю дугу, вычисление координат на дуге и баллистическую траекторию после отрыва.

Параметры задачи

- m = 2.0 кг — масса тела
- R = 5.0 м — радиус дуги
- a = π/2 + π/6 — угловой размер дуги (в радианах)
- g = 9.81 м/с^2 — ускорение свободного падения
- k_fr = 0.02 — коэффициент трения

Анализ изменения энергии/учет силы трения

В нижней точке:

- Кинетическая энергия: $E_{k0} = \frac{1}{2}mv_0^2$
- Потенциальная энергия: $E_{p0} = 0$ (принимаем начальный уровень как y=0)

В точке на дуге под углом *α*:

- Кинетическая энергия: $E_k = \frac{1}{2}mv_0^2 - mgR(1-\cos(a)) - \mu mgRa$ (в точке отрыва или подъема)
- Потенциальная энергия: $E_p = m \cdot g \cdot h = m \cdot g \cdot R \cdot (1-\cos(a))$, где высота h над нижней точкой дуги будет равна R(1−cos(a)), так как h — это вертикальное смещение от нижней точки дуги, равна разнице между радиусом и вертикальной проекцией.

Сила трения:

Вдоль дуги $F_{\text{тр}} = \mu mg$. Работа, которую совершает сила трения на длине дуги S=Ra равняется $F_{\text{тр}} \cdot s = \mu mgRa$

Расчёт минимальной начальной скорости

Чтобы тело смогло подняться до угла *α* на дуге, его E_k0 должна компенсировать как изменение E_p, так и работу F_fr. Таким образом: $\frac{1}{2}mv_0^2 = mgR(1-\cos(a)) + \mu mgRa$

$v_0 = \sqrt{2gR(1-\cos(a)) + 2\mu gRa}$ что в коде эквивалентно

```
E_pot = g * R * (1 - np.cos(a))
F_fr = k_fr * g * R * a
v0 = np.sqrt(2 * E_pot + 2 * F_fr)
```

Вычисление координат дуги

Координаты тела на дуге рассчитываются с помощью углового значения θ. Так как дуга является частью окружности, то координаты можно определить через радиус и угол θ:

```
Arc_x = R * np.sin(theta_values)
Arc_y = -R * np.cos(theta_values) + R
```

Скорости на момент отрыва

Когда тело достигает конца дуги, его скорость разлагается на горизонтальную и вертикальную компоненты:

```
vx_AfterSeparation = v0 * np.cos(a)
vy_AfterSeparation = v0 * np.sin(a)
```

Траектория после отрыва (баллистическое движение)

После отрыва от дуги тело движется по параболической траектории под углом. Для моделирования такого движения используются следующие формулы:

```
25    FlyTime_values = np.linspace( start: 0,  stop: 3,  num: 1000)
26    Flying_x = Arc_x[-1] + vx_AfterSeparation * FlyTime_values
27    Flying_y = Arc_y[-1] + vy_AfterSeparation * FlyTime_values - 0.5 * g * FlyTime_values**2
```

Здесь g учитывает ускорение свободного падения, и формула описывает траекторию тела при баллистическом движении, где Flying_x и Flying_y являются текущими координатами в зависимости от времени взятого из FlyTime_values. Для y-координаты учитывается ускорение свободного падения (0.5 * g * FlyTime_values**2).

Заключение

В отчёте рассмотрен код для визуализации движения тела по мёртвой петле с учетом трения. Используемые формулы позволяют высчитать начальную скорость, траекторию на дуге и баллистическое движение после отрыва.