

Actividad 6

ANDRÉS IGNACIO RODRÍGUEZ MENDOZA



Introducción

La ecuación diferencial se facilita considerando ángulos más pequeños que un radián, es decir, $\theta \ll 1$, por lo cual, $\sin \theta \approx \theta$. Reemplazando esto en la ecuación diferencial, queda,

$$\frac{d^2\theta}{dt^2} + \frac{g}{l}\theta = 0.$$

Dadas las condiciones iniciales $\theta(0) = \theta_0$ y $d\theta/dt(0) = 0$, la solución es,

$$\theta(t) = \theta_0 \cos\left(\sqrt{\frac{g}{l}}t\right)$$

El movimiento es armónico simple, donde θ_0 es la semiamplitud de oscilación. El periodo del movimiento es,

$$T_0 = 2\pi\sqrt{\frac{l}{g}}.$$

Para amplitudes más grandes que la aproximación a ángulos pequeños, se puede calcular el periodo invirtiendo la ecuación de la velocidad angular,

$$\frac{dt}{d\theta} = \frac{l}{2g} \frac{1}{\sqrt{\cos \theta - \cos \theta_0}}$$

e integrando sobre un ciclo completo,

$$T = t(\theta_0 \rightarrow 0 \rightarrow -\theta_0 \rightarrow 0 \rightarrow \theta_0),$$

o dos veces un medio ciclo,

$$T = 2t(\theta_0 \rightarrow 0 \rightarrow -\theta_0),$$

o cuatro veces un cuarto de ciclo,

$$T = 2t(\theta_0 \rightarrow 0),$$

que nos lleva a

$$T = 4\sqrt{\frac{l}{2g}} \int_0^{\theta_0} \frac{1}{\sqrt{\cos \theta - \cos \theta_0}} d\theta.$$

Nótese que esta integral diverge como θ_0 se aproxime a la vertical. Así, con la energía adecuada, un péndulo en su máximo podría tomar un tiempo arbitrariamente largo para caer.

Para resolver numericamente esta integral se utiliza *Scipy.integrate.quad* para integrales definidas de la biblioteca de funciones *Scipy.integrate* en Python. Para mostrar que la integral diverge cuando el ángulo inicial tiene a π se resuelve la integral desde cero hasta π . Los resultados se comparan con el periodo de la aproximación a ángulos pequeños.

Código

```
import numpy as np
from scipy import integrate
import matplotlib.pyplot as plt

n=1000

# Arreglos
x=[]
TT0=[]
x_0=np.linspace(0.001,np.pi + 0.001, n)

# el integrando
I = lambda x,a: 1/np.sqrt(np.cos(x) - np.cos(a))

for i in range(n):
# la integral
    theta_0=x_0[i]
    T , err= integrate.quad(I, 0, theta_0, args=(theta_0,))

# el error
    TT0.append(np.sqrt(2)/np.pi * T)

# Gráfica desviación
plt.figure(1)
plt.plot(x_0 * 180 / np.pi, TT0, "r" )
plt.title('Desviación periodo real - aproximación')
plt.xlabel(r'$\theta_0$(deg)$')
plt.ylabel("T/To")
plt.xlim(0,90)
plt.ylim(1,1.2)
plt.grid()

plt.show()

# Gráfica divergencia
plt.figure(1)
```

```
plt.plot(x_0 * 180 / np.pi, TT0, "b" )
plt.title('Divergencia en  $\theta_0 = \pi$ ')
plt.xlabel(r' $\theta_0$  (deg)')
plt.ylabel("T/To")
plt.xlim(0,180)
plt.ylim(1,5)
plt.grid()

plt.show()
```

Gráficas



