Actividad 6: Periodo del Péndulo

Valenzuela Carrillo María Inés

2 de Marzo del 2016

1. Introducción

Al tiempo que le toma al péndulo regresar a la posición inicial se le llama periodo. Para ángulos pequeños la aproximación al periodo del péndulo está dada por:

$$T = 2\pi\sqrt{\frac{l}{g}} \tag{1}$$

Para amplitudes mayores que la aproximación para ángulos pequeños, se puede calcular el periodo exacto por la inversa de la ecuación para la velocidad angular:

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

Al invertirla se obtiene:

$$\frac{dt}{d\theta} = \sqrt{\frac{l}{2g}}\frac{1}{\sqrt{\cos\theta - \cos\theta_0}} \tag{3}$$

Y luego integrando sobre cuatro veces un cuarto de ciclo:

$$T = 4t(\theta_0 \rightarrow 0)$$

Se puede escribir como

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

En esta actividad se pide encontrar una gráfica de error relativo T/T_0, donde T es el periodo para amplitudes arbitrarias y T_0 la aproximación para ángulos pequeños. También se pide demostrar que la integral definida del periodo diverge a medida que el angulo inicial tiende a π, tomando un tiempo infinito en llegar a π o moverse de π.

Para realizar la actividad se utilizara la biblioteca de funciones para integrar que trae Scipy: scipy.integrate. Específicamente la función scipy.integrate.quad para integrales definidas.

2. Periodo del Péndulo

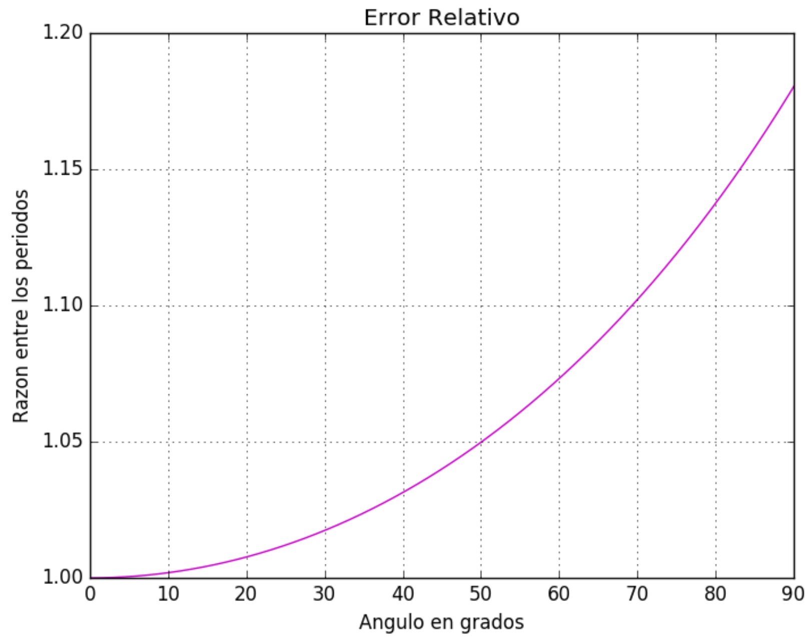
Se realizó un código en Python para obtener el periodo y realizar las graficas.

El código realizado es:

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

#CONSTANTES

g=9.8 #aceleracion gravitacional
l= 2  #longitud del pendulo en metros
n= 500 #numero de angulos
ep=0.001 # error para que no divida entre 0

#variacion del angulo
theta_0 =np.linspace(ep, (np.pi)-ep, n)

#RESULTADOS DE LA INTEGRAL
```

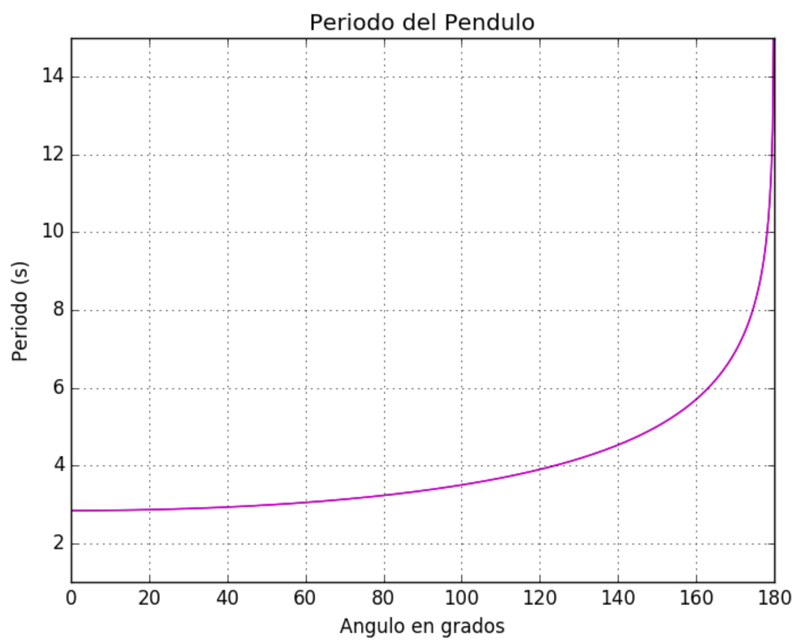
```
I = [0 for i in range(n)]
err = [0 for i in range(n)]
Tr = [0 for i in range(n)]

#PARA ANGULOS PEQUENOS
T0 = 2.0 * np.pi*np.sqrt(l/g)

#RESOLVER LA INTEGRAL
inte = lambda x, t : 1.0 /(np.sqrt(np.cos(x)-np.cos(t)))

for i in range(n):
    theta_00 = theta_0[i]
    I[i] , err [i] = quad(inte, 0, theta_00, args=(
       theta_00))
    Tr[i] = 4*np.sqrt(l/(2*g)) * I[i]


#Periodo real entre el periodo para angulos pequenos
raz=Tr/T0

theta_g = (theta_0*180.0)/np.pi #Transformacion a Grados


#PARA GRAFICAR
plt.plot(theta_g, raz, color="m")
plt.title("Error Relativo")
plt.xlabel("Angulo en grados")
plt.ylabel("Razon entre los periodos")
plt.axis([0,90,1,1.20])
plt.show()
```

Enseguida se muestra la gráfica resultante, la cual muestra la razón entre el periodo para un péndulo de amplitud arbitraria y la aroximación para ángulos pequeños.



En la gráfica de observa como para ángulos mayores a 50 grados el error es mayor al 5 %. Para ángulos menores de 30 fgrados la aproximación es buena.

En la siguiente gráfica se muestra como aumenta en periodo al aumentar el angulo inicial.



En la gráfica anterior se puede observar como al acercarse el angulo a 180 grados (π rad) el periodo tiende a infinito.

Referencias

[1] WIKIPEDIA, THE FREE ENCYCLOPEDIA, *Pendulum (mathematics)*.

[2] Actividad 6 (2016-1), *http://computacional1.pbworks.com/w/page/105454998/Actividad 6 (2016-1)*.

[3] SCIPY, scipy.integrate.quad, *http://docs.scipy.org/doc/scipy/reference /generated/scipy.integrate.quad.html*.