Universidad de Sonora

División de Ciencias Exactas y Naturales

Física Computacional I

Práctica #7: Espacios fase.

Integrante:
Alejandro Guirado García

18 de marzo de 2016

1. Introducción

El espacio fásico, espacio de fases o diagrama de fases es una construcción matemática que permite representar el conjunto de posiciones y momentos conjugados de un sistema de partículas. Más técnicamente, el espacio de fases es una variedad diferenciable de dimensión par, tal que las coordenadas de cada punto representan tanto las posiciones generalizadas como sus momentos conjugados correspondientes. Es decir, cada punto del espacio fásico representa un estado del sistema físico. Ese estado físico vendrá caracterizado por la posición de cada una de las partículas y sus respectivos momentos.

$$\frac{\partial^2\theta}{dt^2} + \frac{g}{l}\sin\theta = 0 \tag{1}$$

Donde g es la aceleración debido a la gravedad, l es la longitud del pendulo y θ es el desplazamiento angular.

El espació fase que representa la familia de soluciones para el péndulo simple, es la siguiente:

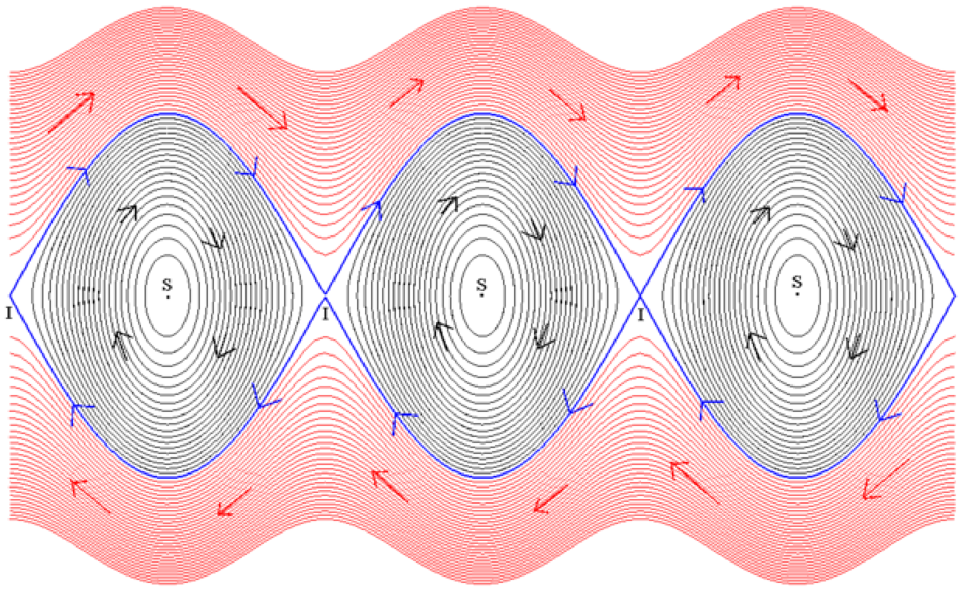

Figura 1: Representación de familia de soluciones para el péndulo simple.

2. Código

El código ejemplo es sobre el comportamiento entre zorros y conejos como iban disminuyendo o aumentando dependiendo de la cantidad de alimento y cantidad de depredadores. Dicho comportamiento estaba modelado con una ecuación diferencial, el código fue cambiado para modelar el comportamiento del ángulo.

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

b = 0.0

g = 9.8
l = 1.
c = g/l

X_f0 = array([ -4*np.pi , 4*np.pi])
X_f1 = array([ -2*np.pi, np.pi*0])
t = np.linspace(0, 40, 500) #rango de tiempo

def pend(y, t, b, c):
     theta, omega = y
     dydt = [omega, -b*omega - c*np.sin(theta)]
     return dydt

values  = linspace(-1, 1, 50)

vcolors = p.cm.Blues(linspace(0.5, 1., len(values)))
# colors for each trajectory
vcolors2 = p.cm.Greens(linspace(0.5, 1., len(values)))
f2 = p.figure()

for v, col in zip(values, vcolors):
    X0 = v * X_f0
    # starting point
    X = integrate.odeint( pend, X0, t, (b, c) )
    p.plot( X[:,0], X[:,1], lw=2*v, color=col,
```

```
    label='X0=(%.f, %.f)' % ( X0[0], X0[1]) )

for v, col in zip(values, vcolors):
    X0 = v * X_f1
    # starting point
    X = integrate.odeint( pend, X0, t, (b, c) )
    p.plot( X[:,0], X[:,1], lw=1*v, color=col,
    label='X0=(%.f, %.f)' % ( X0[0], X0[1]) )

p.title('Espacio fase del pendulo simple.')
p.grid()
p.xlim((-4*np.pi)/2, (4*np.pi)/2)
p.ylim(-4*np.pi, 4*np.pi)
f2.savefig('rabbits_and_foxes_2.png')
```

3. Resultados

Para construir la figura, se utilizó la biblioteca Matplotlib de Python para graficación, y nos se adaptó el código del ejemplo resuelto en el tutorial de Lotka-Volterra que aparece en el SciPy CookBook. El código nos arrojo la siguiente representación:

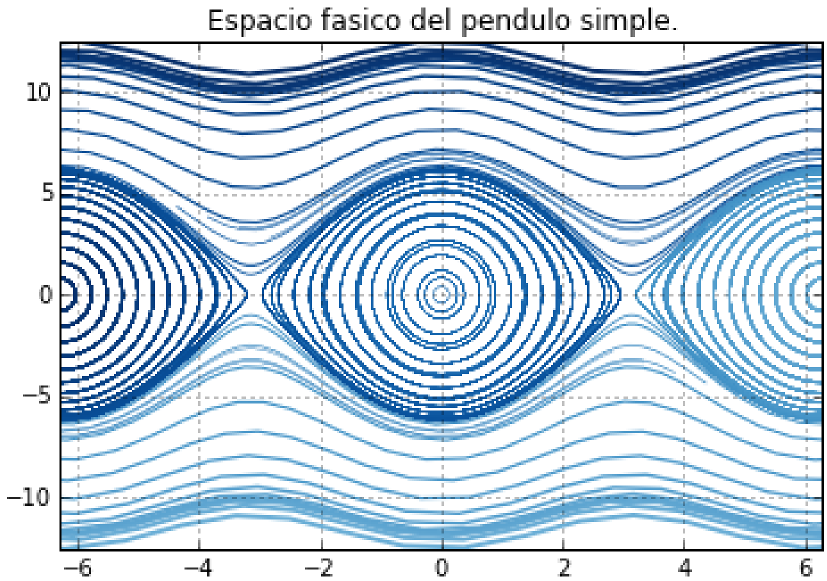

Figura 2: Representación de familia de soluciones para el péndulo simple.

4. Bilbiografía

1. Wikipedia, *Pendulum*. Recuperado el 16 de febrero de 2016 de https://en.wikipedia.org/wiki/Pe
2. Chevtorno,*"Simple gravity pendulum"model assumes no friction or air resistance*. Recuperado el 16 de febrero de 2016 de `https://en.wikipedia.org/wiki/Pendulum#/media/File:Simple_gravity_pendulum.svg`
3. Scipy, *odeint*. http://scipy.github.io/devdocs/generated/scipy.integrate.odeint.html
4. Lizárraga, C. *Actividad 7 (2016-1)*. Recuperado el 16 de febrero de 2016 de http://computacional1.pbworks.com/w/page/105233358/Actividad %205 %20(2016-1