



Universidad de Sonora

Departamento de Física

Licenciatura en Física

Computacional I

2017-1

Reporte

Actividad #8: Atractores Extraños. El Efecto Mariposa.

Leonardo Coronado Arvayo

Profesor: Carlos Lizárraga Celaya

12 de mayo de 2017

Índice

1. Introducción	2
2. Sistema de Lorenz	2
2.1. Análisis del Sistema de Lorenz	2
3. Conclusiones	7

Resumen

La presente practica trata sobre el planteamiento y revolvimiento de sistemas de Lorenz o atractores de Lorenz usando códigos de Python que por una lado resuelven el sistema de ecuaciones diferenciales y por otro grafican las soluciones.

1. Introducción

Los atractores de Lorenz son la solución caótica de un sistema de ecuaciones diferenciales donde se tienen las variables, parámetros del sistema y valores iniciales.

En la presente actividad se plantean de manera general en que consisten los sistemas de Lorenz, seguido por características para casos en específico de los sistemas.

Luego se grafican 5 casos de los atractores de Lorenz usando un código elaborado por [3] que permite primeramente resolver el sistema de ecuación de Lorenz y graficar los resultados (se grafican para diferentes valores de los parámetros ρ , σ y β).

Por ultimo se elabora un gif que muestra la evolución de un atractor de Lorenz usando el código de [4] y Python 3.6. Además, se presenta un mp4 con el código de [5] y Python 2.7. Ambos archivos se adjuntan al Github.

2. Sistema de Lorenz

El sistema de Lorenz se refiere a un sistema de ecuaciones diferenciales estudiado por Edward Lorenz [1]. El sistema brinda soluciones caóticas para sistemas dinámicos que están condicionados por ciertos valores iniciales y parámetros constantes [1]. De forma específica, las soluciones del sistema dinámico (también llamado atractor de Lorenz), es un conjunto de soluciones caóticas que cuando se grafican se asemejan a una mariposa o el número ocho [1].

Un ejemplo del sistema de ecuaciones de Lorenz, en este caso de un modelo simplificado para la convección atmosférica es [1]:

$$\frac{dx}{dt} = \sigma(y - x) \quad (1)$$

$$\frac{dy}{dt} = x(\rho - z) - y \quad (2)$$

$$\frac{dz}{dt} = xy - \beta z \quad (3)$$

Los valores de x , y y z son las propiedades/estado del sistema; t es el tiempo y σ , ρ y β son parámetros del sistema [1].

2.1. Análisis del Sistema de Lorenz

Normalmente se asumen que los parámetros σ , ρ y β son positivos [1]. Dependiendo de los valores que se les de a los parámetros el sistema puede exhibir comportamiento caótico en distintos grados. Por ejemplo:

- Lorenz estudio los valores de $\sigma=10$, $\rho=28$ y $\beta = \frac{8}{3}$ donde el sistema tiene comportamiento caótico para estos valores (y cercanos a ellos) [1].
- Si $\rho < 1$, entonces solo existe un punto de equilibrio (el origen) que corresponde a la no convección [1]. De esta forma se dice que todas las órbitas convergen al origen (siendo este el atractor global).
- Para $\rho = 1$ y $\rho > 1$ se tienen dos puntos críticos que son respectivamente:

$$\left(\sqrt{\beta(\rho - 1)}, \sqrt{\beta(\rho - 1)}, \rho - 1 \right)$$

$$\left(-\sqrt{\beta(\rho - 1)}, -\sqrt{\beta(\rho - 1)}, \rho - 1 \right)$$

Estos corresponden a una convección suave, y dichos puntos son estables solo si se cumple:

$$\rho < \sigma \frac{\sigma + \beta + 3}{\sigma - \beta - 1}$$

Que solo puede tener valores de ρ positivos si $\sigma > (\beta+1)$; en el valor crítico los dos puntos de equilibrio pierden estabilidad por la Bifurcación de Hopf que sucede cuando la solución pierde estabilidad y se requiere una solución periódica en su lugar [2].

Ejemplos de las soluciones del sistema de Lorenz para distintos valores de ρ se pueden ver en la siguiente imagen:

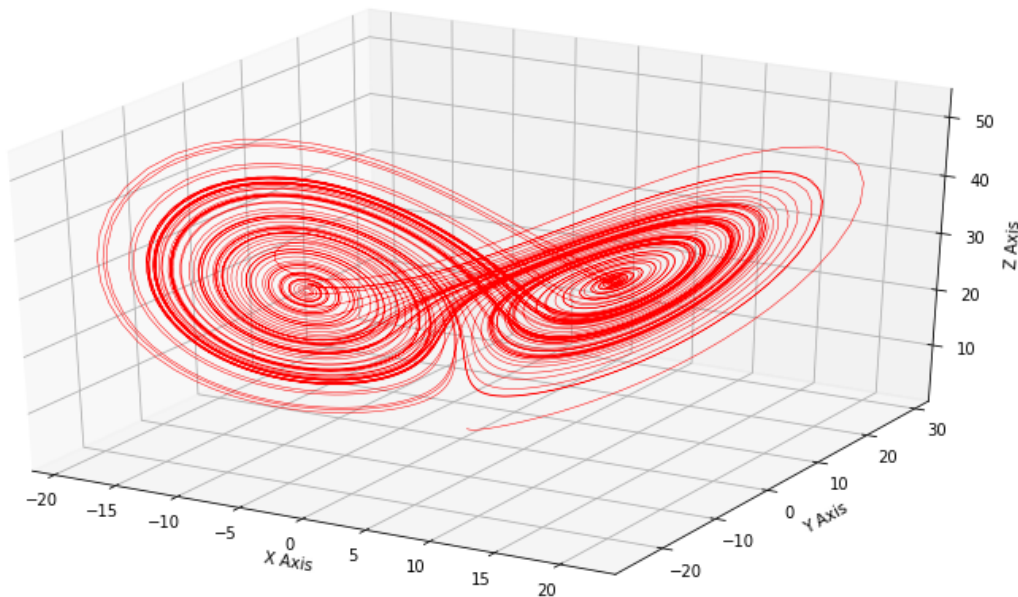


Figura 1: Atractor de Lorenz para $\sigma=10$, $\rho=28$, $\beta=2.667$

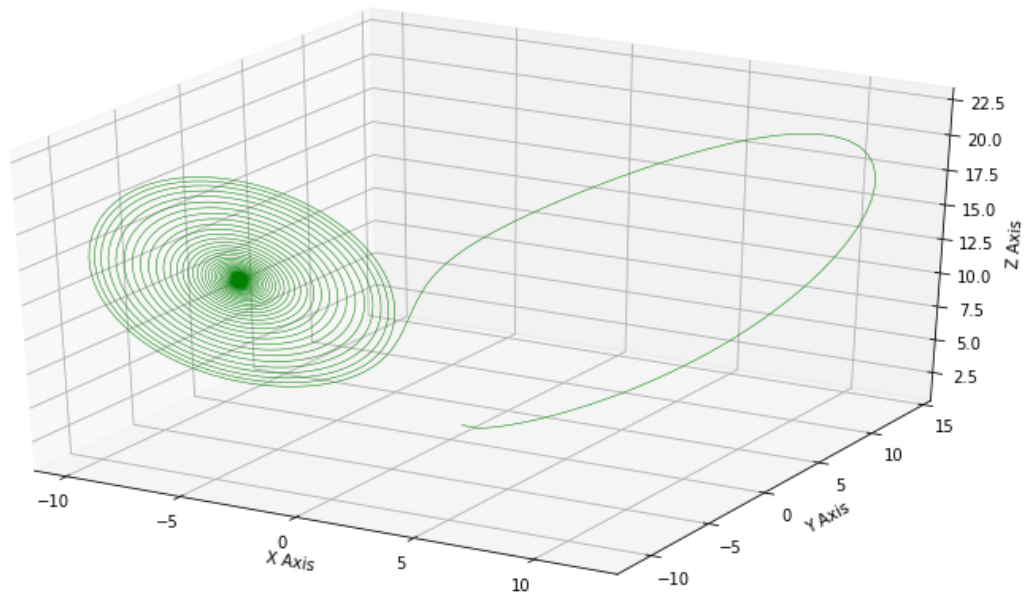


Figura 2: Atractor de Lorez para $\sigma=10$, $\rho=14$, $\beta=2.667$

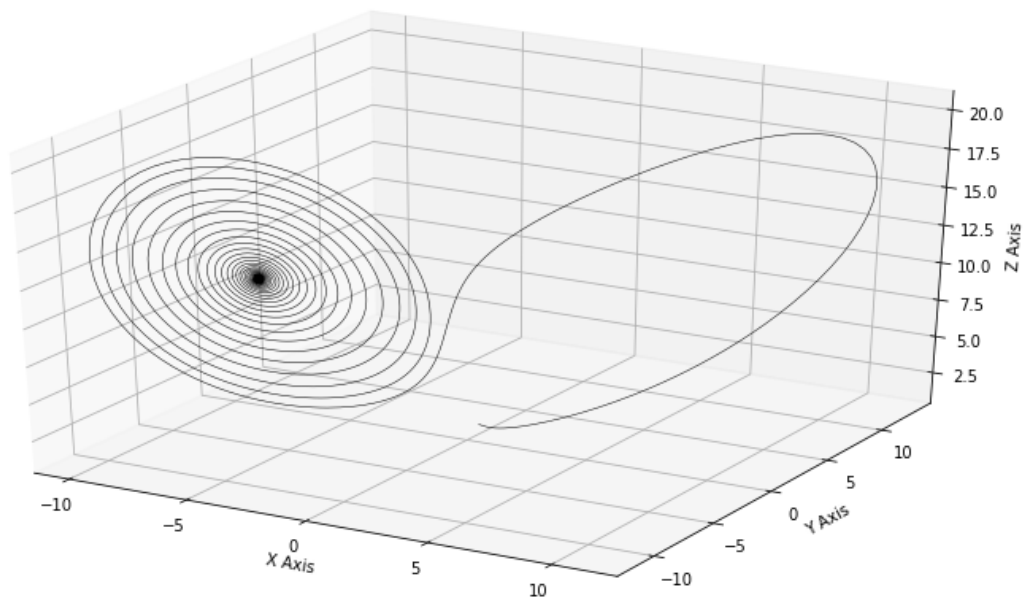


Figura 3: Atractor de Lorez para $\sigma=10$, $\rho=13$, $\beta=2.667$

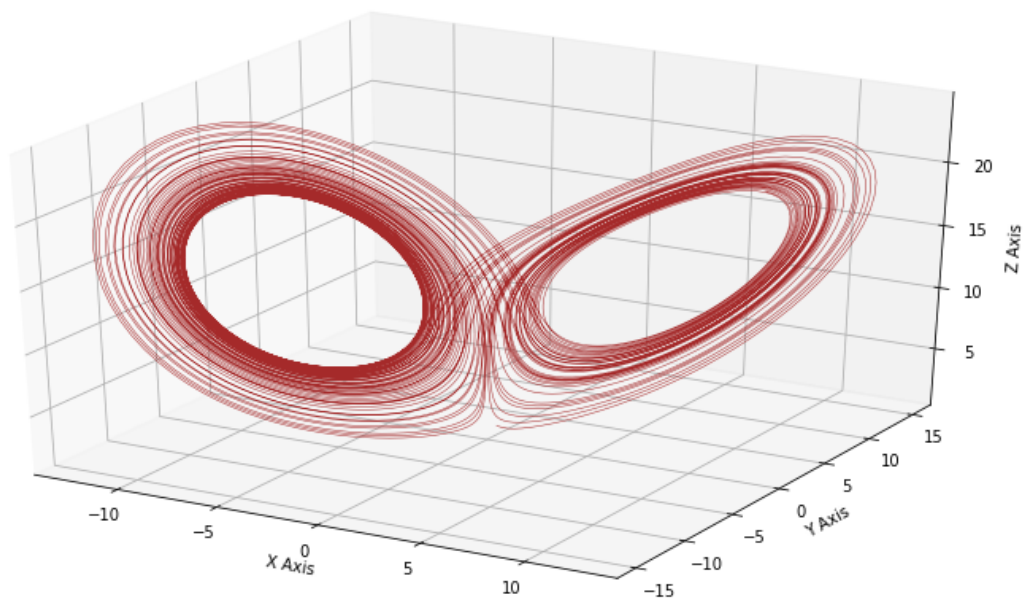


Figura 4: Atractor de Lorez para $\sigma=10$, $\rho=15$, $\beta=2.667$

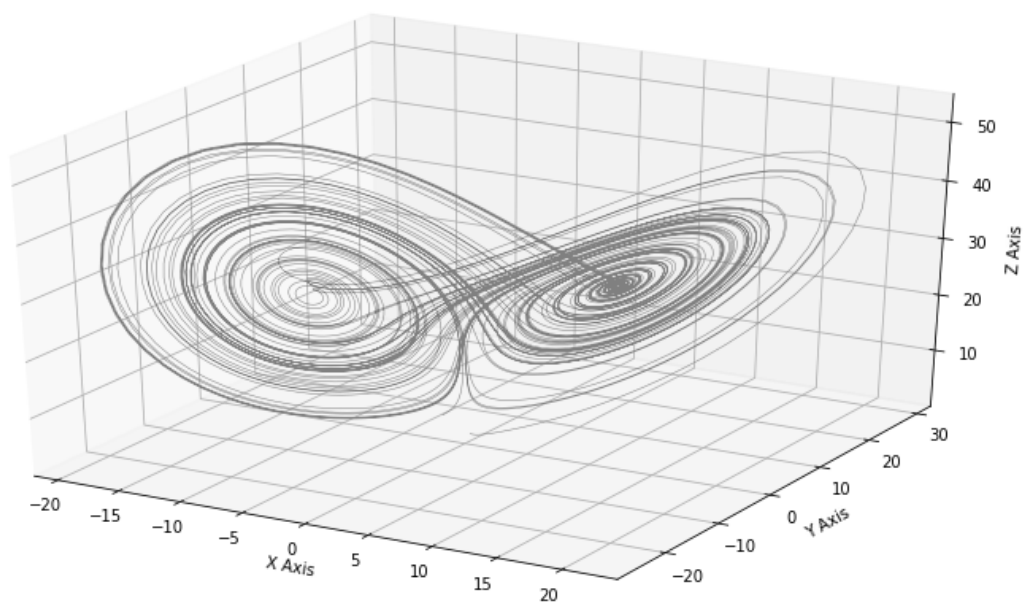


Figura 5: Atractor de Lorez para $\sigma=10$, $\rho=28$, $\beta=2.667$

Las imágenes anteriores se hicieron usando el código:

```
# Plot of the Lorenz Attractor based on Edward Lorenz's 1963 "Deterministic
# Nonperiodic Flow" publication.
# http://journals.ametsoc.org/doi/abs/10.1175/1520-0469%281963%29020%3C0130%3ADNF%3E2.0.CO%3E2
```

```
#
# Note: Because this is a simple non-linear ODE, it would be more easily
#       done using SciPy's ode solver, but this approach depends only
#       upon NumPy.

import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

def lorenz(x, y, z, s=10, r=28, b=2.667):
    x_dot = s*(y - x)
    y_dot = r*x - y - x*z
    z_dot = x*y - b*z
    return x_dot, y_dot, z_dot

dt = 0.01
stepCnt = 10000

# Need one more for the initial values
xs = np.empty((stepCnt + 1,))
ys = np.empty((stepCnt + 1,))
zs = np.empty((stepCnt + 1,))

# Setting initial values
xs[0], ys[0], zs[0] = (0., 1., 1.05)

# Stepping through "time".
for i in range(stepCnt):
    # Derivatives of the X, Y, Z state
    x_dot, y_dot, z_dot = lorenz(xs[i], ys[i], zs[i])
    xs[i + 1] = xs[i] + (x_dot * dt)
    ys[i + 1] = ys[i] + (y_dot * dt)
    zs[i + 1] = zs[i] + (z_dot * dt)

fig = plt.figure()
ax = fig.gca(projection='3d')

ax.plot(xs, ys, zs, lw=0.5)
ax.set_xlabel("X Axis")
ax.set_ylabel("Y Axis")
ax.set_zlabel("Z Axis")
ax.set_title("Lorenz Attractor")

plt.show()
```

Ademas, se usaron los codigos proporcionados por [4] y [5] para realizar un gif y un mp4 del comportamiento del atractor de Lorenz en el tiempo. Para el primero se uso Python 2.7 mientras para el segundo Python 3.6. El vídeo y el gif están adjuntos en la carpeta de Github.

3. Conclusiones

La actividad resulto muy interesante primeramente por que se tiene en mano códigos para resolver sistemas de ecuaciones diferenciales y segundo por que los sistemas caóticos me parecen interesantes, aparte de que son apreciables de observar.

Ademas, el poder crear un mp4 y gif que muestre la evolución de las soluciones del sistema en el tiempo no solo es bonito de ver sino que puede resultar útil para simular toda clase de problemas (que me parece una habilidad importante).

Referencias

- [1] Lorenz system. Wikipedia: https://en.wikipedia.org/wiki/Lorenz_system
- [2] Hopf bifurcation. Wikipedia: https://en.wikipedia.org/wiki/Hopf_bifurcation
- [3] Matplotlib. mplot3d example code, Lorenz attractor: https://matplotlib.org/2.0.0/examples/mplot3d/lorenz_attractor.html
- [4] Gboeing. Lorenz System: <https://github.com/gboeing/lorenz-system>
- [5] Jake VanderPlas. Animating the Lorenz System in 3D: <https://jakevdp.github.io/blog/2013/02/16/animating-the-lorentz-system-in-3d/>