Indian Institute of Science Education and Research, Mohali

IDC402 Term Paper
Rössler Attractor

Name by	-	Athira Sreejith
Roll no:	-	MS18033

Course	-	IDC402 - Nonlinear Dynamics
Instructor	-	Prof Sudeshna Sinha
Date of submission	-	29-04-2022

Contents

1 Introduction

Rossler attractor is a strange attractor of the Rossler system. n attractor is a set of states toward which a system tends to evolve, for a wide variety of initial conditions of the system. Being a strange chaotic attractor, Rossler attractor is charecterised by a frcatal structure as well as sensitive dependence on initial conditions.

1.1 Rössler system

Rössler system is a system of three non-linear ordinary differential equations studied initially by Otto Rössler. This system has only one non-linear term, and Rössler attractor is one of the simplest possible strange attractor.

The defining equations of Rössler systems are:

$$\begin{aligned} \dot{x} &= -y - z \\ \dot{y} &= x + ay \\ \dot{z} &= b + z(x - c) \end{aligned}$$

The chaotic attractor studied by Rössler used the parameter values $a = 0.2$, $b = 0.2$ and $c = 5.7$.

2 Fixed points

The fixed points can be obtained by setting the three Rössler equations to zero and the (x, y, z) coordinates of each fixed point can be obtained by solving the resulting equations.

$$\begin{cases} x = \frac{c \pm \sqrt{c^2 - 4ab}}{2} \\ y = -\left(\frac{c \pm \sqrt{c^2 - 4ab}}{2a}\right) \\ z = \frac{c \pm \sqrt{c^2 - 4ab}}{2a} \end{cases}$$

For a given set of parameter values, the fixed points are given by

$$\left(\frac{c + \sqrt{c^2 - 4ab}}{2}, \frac{-c - \sqrt{c^2 - 4ab}}{2a}, \frac{c + \sqrt{c^2 - 4ab}}{2a}\right)$$
$$\left(\frac{c - \sqrt{c^2 - 4ab}}{2}, \frac{-c + \sqrt{c^2 - 4ab}}{2a}, \frac{c - \sqrt{c^2 - 4ab}}{2a}\right)$$

Among these fixed points, one can be found in the center of the attractor loop while the other one is relatively far from the attractor.

3 Eigenvalues and Eigenvectors

Jacobian of this system is given by:

$$\begin{pmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ z & 0 & x - c \end{pmatrix}$$

The eigenvalues are therefore given by

$$-\lambda^3 + \lambda^2(a + x - c) + \lambda(ac - ax - 1 - z) + x - c + az = 0$$

For the fixed point located at the center of the attractor for the parameter values used by Rössler, we get,

$$\lambda_1 = 0.0971028 + 0.995786i$$
$$\lambda_2 = 0.0971028 - 0.995786i$$
$$\lambda_3 = -5.68718$$

4 Plotting the Rössler Attractor

The 3-d plot of Rössler attractor, has been plotted for $a = 0.2$, $b = 0.2$ and $c = 5.7$, the parameter values studied by Rössler.

4.1 Code

Initially the variables are defined, and a function called Rössler is defined.

```
from mpl_toolkits.mplot3d import axes3d
import matplotlib.pyplot as pl
import numpy as np

#Defining the function Rossler which returns the new values of x,y,z after
    time dt
def rossler(x,y,z):
    #updating the lists containing the values of x,y,z
    x_t.append(x)
    y_t.append(y)
    z_t.append(z)

    #Finding the new values of x,y,z after time dt
    x_i = x + (-y - z) * dt
    y_i = y + (x + a*y) * dt
    z_i = z + (b + z*(x-c)) * dt

    return(x_i,y_i,z_i)

x_t = []
y_t = []
z_t = []
t_t = []

t_initial = 0.0
dt = 0.02
t_final = 60
T = np.arange(t_initial, t_final, dt)
a, b, c = 0.2, 0.2, 5.7

x0,y0,z0 = (0.0,-5.0,0.0)
```

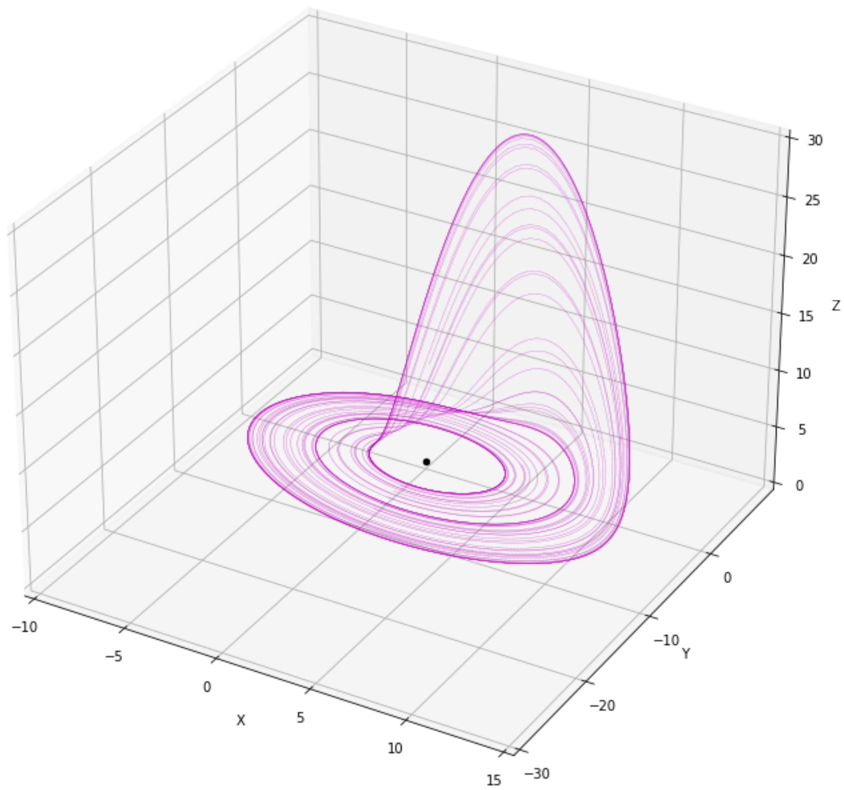
The Rössler function is now used to find the evolution of the system, to obtain a plot as given below.

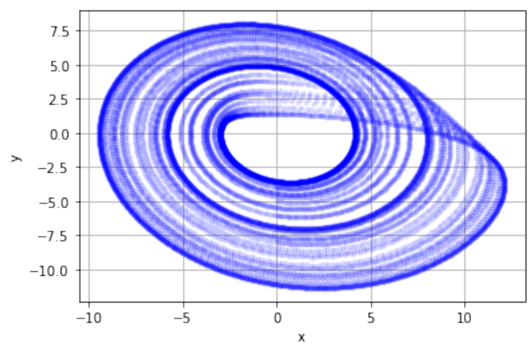
```
x_t = []
y_t = []
z_t = []
t_t = []

ax = pl.figure(figsize = (15,12)).add_subplot(111, projection='3d')
ax.set_xlim([-10, 15])
ax.set_ylim([-30, 10])
ax.set_zlim([0, 30])
ax.set_yticks(np.arange(-30, 10, 10))
```

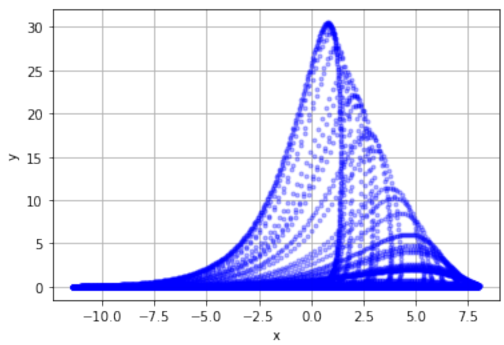
```
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')

for y0 in np.arange(-8,-3,1.0): #setting different values of y0 in the initial condition
    x0, z0 = 0.0, 0.0

    for t in T:

        new_x,new_y,new_z = rossler(x0,y0,z0) #calculating new x,y,z
        t_t.append(t)

        ax.plot([x0, new_x], [y0, new_y], [z0, new_z], 'm-',alpha = 0.6, linewidth=0.3) #plotting the x,y,z values

        x0, y0, z0 = new_x, new_y, new_z #Updating the new values of x,y,z


ax.scatter(0,0,0,color='black') #plotting  the origin
pl.show()


pl.plot(x_t,y_t,'b.',alpha = 0.1)
pl.ylabel("y")
pl.xlabel("x")
pl.grid(True)
pl.show()

pl.plot(y_t,z_t,'b.',alpha = 0.3)
pl.ylabel("z")
pl.xlabel("y")
pl.grid(True)
pl.show()

pl.plot(x_t,z_t,'b.',alpha = 0.3)
pl.ylabel("x")
pl.xlabel("z")
pl.grid(True)
pl.show()
```

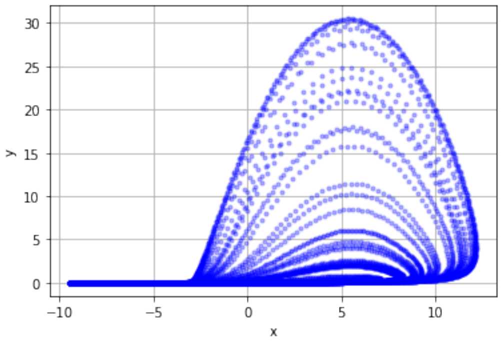
4.2 Plots

The 3D plot as well as the projections in xy,yz and xz planes are given below.


X
Y
Z
−10
−5
0
5
10
15
−30
−20
−10
0
0
5
10
15
20
25
30


x
y
−10
−5
0
5
10
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0


x
y
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
30
25
20
15
10
5
0



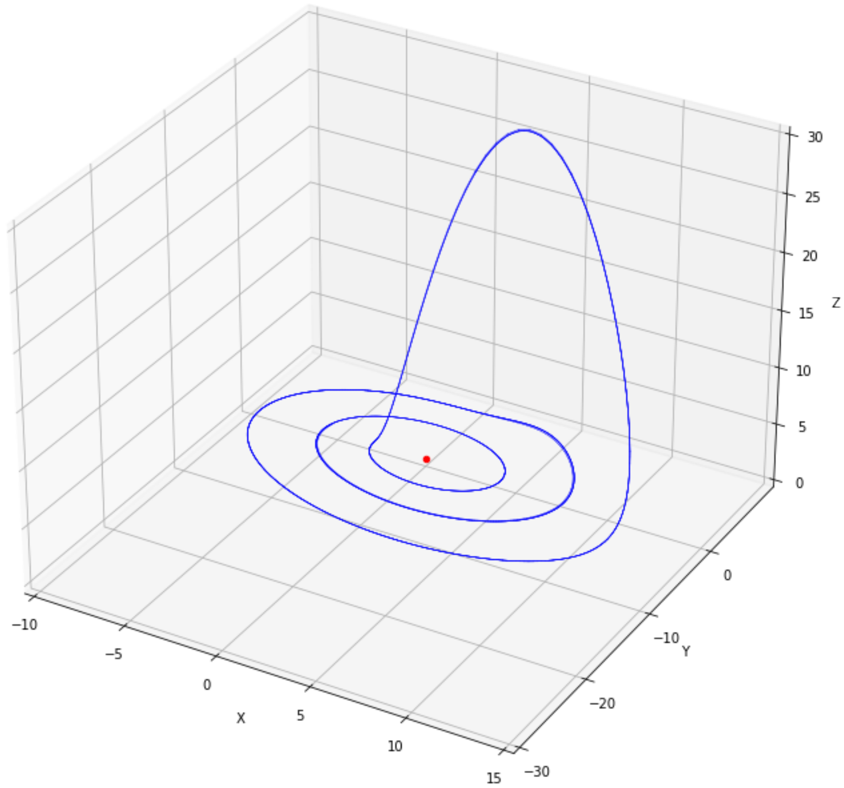
5 Time series

The time series of the Rössler attractor was studied at the paramter values $a = b = 0.2$ and $c = 5.7$.

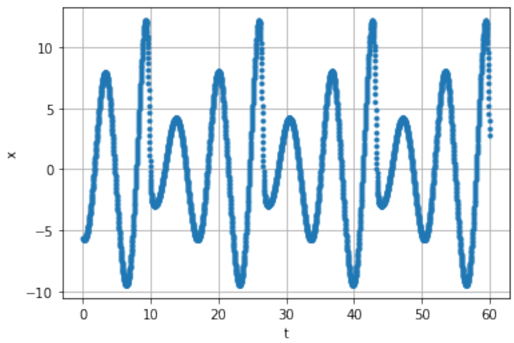
5.1 Code

```
ax = pl.figure(figsize = (15,12)).add_subplot(111, projection='3d')
ax.set_xlim([-10, 15])
ax.set_ylim([-30, 10])
ax.set_zlim([0, 30])
ax.set_yticks(np.arange(-30, 10, 10))
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')

for t in T:
    new_x,new_y,new_z = rossler(x0,y0,z0)
    t_t.append(t)

    ax.plot([x0, new_x], [y0, new_y], [z0, new_z], 'b-', linewidth=0.5)

    x0, y0, z0 = new_x, new_y, new_z


ax.scatter(0,0,0,color='red')

pl.show()


pl.plot(t_t,x_t,'.')
pl.ylabel("x")
pl.xlabel("t")
pl.grid(True)
pl.show()


pl.plot(t_t,y_t,'.')
pl.ylabel("y")
pl.xlabel("t")
pl.grid(True)
pl.show()

pl.plot(t_t,z_t,'.')
pl.ylabel("z")
pl.xlabel("t")
pl.grid(True)
pl.show()
```

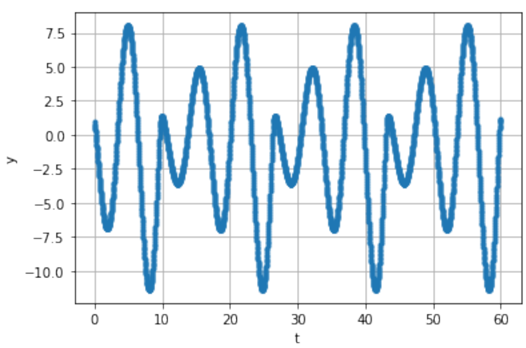
5.2 Plots



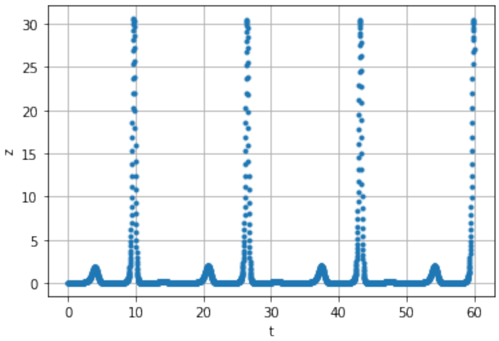
30
25
20
15
10
5
0
Z
−10
−5
0
X
5
10
15
0
−10
Y
−20
−30



10
5
0
−5
−10
x
0
10
20
30
40
50
60
t



7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
y
0
10
20
30
40
50
60
t



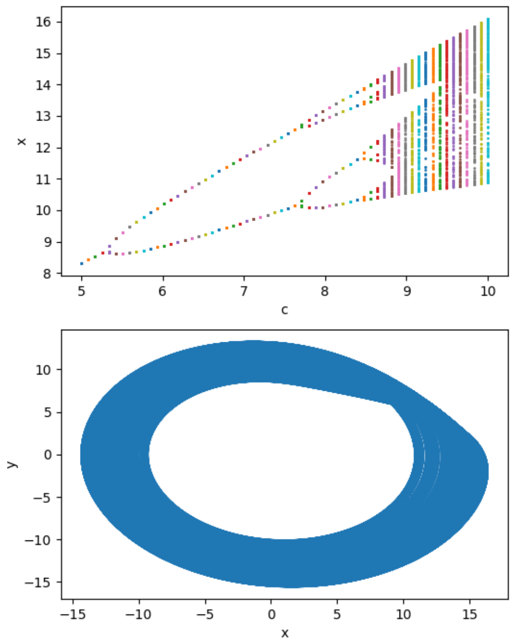
6 Bifurcation Diagram

The bifurcation diagram was obtained by varying c and plotting x vs c.

6.1 Code

```
ax = pl.figure(figsize = (15,12)).add_subplot(111, projection='3d')
ax.set_xlim([-10, 15])
ax.set_ylim([-30, 10])
ax.set_zlim([0, 30])
ax.set_yticks(np.arange(-30, 10, 10))
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')

for t in T:
    new_x,new_y,new_z = rossler(x0,y0,z0)
    t_t.append(t)

    ax.plot([x0, new_x], [y0, new_y], [z0, new_z], 'b-', linewidth=0.5)

    x0, y0, z0 = new_x, new_y, new_z


ax.scatter(0,0,0,color='red')

pl.show()


pl.plot(t_t,x_t,'.')
pl.ylabel("x")
pl.xlabel("t")
pl.grid(True)
pl.show()


pl.plot(t_t,y_t,'.')
pl.ylabel("y")
pl.xlabel("t")
pl.grid(True)
pl.show()

pl.plot(t_t,z_t,'.')
pl.ylabel("z")
pl.xlabel("t")
pl.grid(True)
pl.show()
```

6.2 Plots



Period doubling bifurcations can be clearly seen from the plots.

7 Conclusion

Rossler attractor is one of the simplest strange attractor. Its 3d plots, time series and bifurcation diagrams have been created and analysed.

References

`https://en.wikipedia.org/wiki/R%C%B6ssler_attractor`
`https://en.wikipedia.org/wiki/Attractor`
`NonlinearDynamicsandChaos-StevenH.Strogatz`