Non-linear Physics Coursework

URN: 6667703
MSc Physics, University of Surrey, United Kingdom

Contents

1 The Duffing Equation **2**

1.1 Question 1 2

1.2 Question 2 3

1.3 Question 3 6

2 Cascade route to chaos and folding of strange attractors **8**

2.1 Question 1 8

2.2 Question 2 8

3 Chaos Recognition **9**

3.1 Question 1 9

References **13**

1 The Duffing Equation

1.1 Question 1

The Duffing equation is given as:

$$\ddot{x} + 2\gamma\dot{x} + \alpha x + \beta x^3 = F_e \cos(\omega t) \tag{1}$$

whereby $\alpha = -1$, $\beta = +1$ and $\omega = +1$. In this scenario, x is the displacement, 2γ is the damping in the system, $F_e \cos(\omega t)$ is the external forcing of the system and βx^3 is the cubic restoring force. It is easier to analyse the solutions to such equation by converting the non linear equation to a system of first order differential equations.

Finding a suitable variable transformation

After replacing the values of α, β and ω in the Duffing equation, we have:

$$\ddot{x} + 2\gamma\dot{x} - x + x^3 = F_e \cos(t) \tag{2}$$

Let $\dot{x} = u$. Therefore, we can rewrite the Duffing equation as follows:

$$\dot{u} + 2\gamma u - x + x^3 = F_e \cos(t)$$

A system of differential equations

$$\begin{aligned} \dot{x} &= u & (3) \\ \dot{u} &= F_e \cos(t) - 2\gamma u - x^3 + x & (4) \end{aligned}$$

We are given $F_{cons} = -\frac{\partial V(x)}{\partial x}$. The potential $V(x)$ is given as follows:

$$V(x) = -\int F_{cons}\, dx$$

In this case, $F_{cons} = x^3 - x$. Therefore,

$$V(x) = \frac{x^4}{4} - \frac{x^2}{2} \tag{5}$$

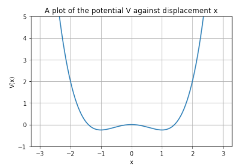


Figure 1: The plot of potential V against displacement x

1.2 Question 2

This time, F_e is equal to zero in equation(4). We will find the nullclines, fixed points and then sketch the phase portraits for $\gamma < 0, \gamma = 0$ and $\gamma > 0$ using GeoGebra online plotting tool.

Vertical Nullclines

$$\begin{aligned} \dot{x} &= 0 \\ u &= 0 \end{aligned}$$

Horizontal Nullclines

$$\begin{aligned} \dot{u} &= 0 \\ 0 &= -2\gamma u + x(-x^2 + 1) \end{aligned}$$

For $u = 0$,

$$-x^3 + x = 0$$

Therefore, $x = 0$, $x = 1$ and $x = -1$. Hence, the fixed points are $(0,0), (1,0)$ and $(-1,0)$.

Using our system of equations (3) and (4), we can construct its Jacobian matrix. The Jacobian matrix is given by

$$J(x,u) = \begin{pmatrix} 0 & 1 \\ -3x^2 + 1 & -2\gamma \end{pmatrix}$$

(i) For fixed point $(0,0)$

$$J(0,0) = \begin{pmatrix} 0 & 1 \\ 1 & -2\gamma \end{pmatrix}$$

(ii) For fixed point $(1,0)$

$$J(1,0) = \begin{pmatrix} 0 & 1 \\ -2 & -2\gamma \end{pmatrix}$$

(iii) For fixed point $(-1, 0)$

$$J(-1,0) = \begin{pmatrix} 0 & 1 \\ -2 & -2\gamma \end{pmatrix}$$

Hence for fixed points (ii) and (iii), they will share similar features as they have the same Jacobian matrix.

For case (i),

when $\gamma = 0$

$$J = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

The eigenvalues are 1 and -1. Their corresponding eigenvectors are $(1\ 1)$ and $(-1\ 1)^T$ respectively.

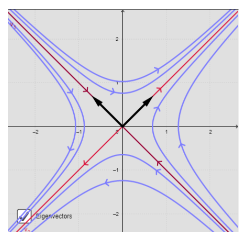


Figure 2: Phase portrait for eigenvectors $(1\ 1)$ and $(-1\ 1)^T$.

when $\gamma > 0$

In particular, we choose $\gamma = 2$

$$J = \begin{pmatrix} 0 & 1 \\ 1 & -4 \end{pmatrix}$$

The eigenvalues are $\sqrt{5} - 2$ and $-\sqrt{5} - 2$. Their corresponding eigenvectors are $(\sqrt{5} + 2\ \ 1)$ and $(-\sqrt{5} + 2\ \ 1)^T$ respectively.

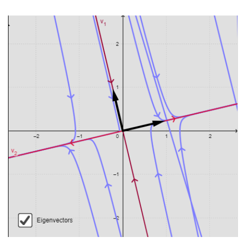


Figure 3: Phase portrait for eigenvectors $(\sqrt{5} + 2\ \ 1)$ and $(-\sqrt{5} + 2\ \ 1)^T$.

when $\gamma < 0$

In particular, we choose $\gamma = -2$

$$J = \begin{pmatrix} 0 & 1 \\ 1 & 4 \end{pmatrix}$$

The eigenvalues are $\sqrt{5}+2$ and $-\sqrt{5}+2$. Their corresponding eigenvectors are $(\sqrt{5}-2 \;\; 1)$ and $(-\sqrt{5}-2 \;\; 1)^T$ respectively.

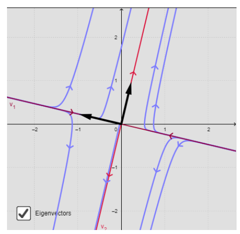


Figure 4: Phase portrait for eigenvectors $(\sqrt{5}-2 \;\; 1)$ and $(-\sqrt{5}-2 \;\; 1)^T$.

For case (ii) or (iii),

when $\gamma = 0$

$$J = \begin{pmatrix} 0 & 1 \\ -2 & 0 \end{pmatrix}$$

The eigenvalues are $\sqrt{2}i$ and $-\sqrt{2}i$. Their corresponding eigenvectors are $(-\sqrt{2}i \;\; 2)$ and $(\sqrt{2}i \;\; 2)^T$ respectively.

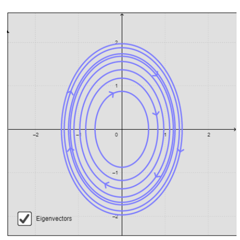


Figure 5: Phase portrait for eigenvectors $(-\sqrt{2}i \;\; 2)$ and $(\sqrt{2}i \;\; 2)^T$.

when $\gamma < 0$

In particular, we choose $\gamma = -2$

$$J = \begin{pmatrix} 0 & 1 \\ -2 & 4 \end{pmatrix}$$

The eigenvalues are $2+\sqrt{2}$ and $2-\sqrt{2}$. Their corresponding eigenvectors are $(\frac{\sqrt{2}-1}{\sqrt{2}} \;\; 1)$ and $(\frac{\sqrt{2}+1}{\sqrt{2}} \;\; 1)^T$ respectively.

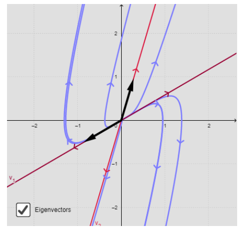


Figure 6: Phase portrait for eigenvectors $(\frac{\sqrt{2}-1}{\sqrt{2}} \quad 1)$ and $(\frac{\sqrt{2}+1}{\sqrt{2}} \quad 1)^T$.

when $\gamma > 0$

In particular, we choose $\gamma = 2$

$$J = \begin{pmatrix} 0 & 1 \\ -2 & -4 \end{pmatrix}$$

The eigenvalues are $-2+\sqrt{2}$ and $-2-\sqrt{2}$. Their corresponding eigenvectors are $(\frac{-\sqrt{2}-1}{\sqrt{2}} \quad 1)$ and $(\frac{-\sqrt{2}+1}{\sqrt{2}} \quad 1)^T$ respectively.

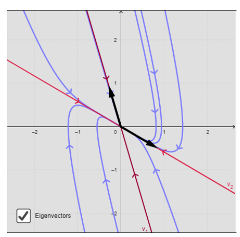


Figure 7: Phase portrait for eigenvectors $(\frac{-\sqrt{2}-1}{\sqrt{2}} \quad 1)$ and $(\frac{-\sqrt{2}+1}{\sqrt{2}} \quad 1)^T$.

1.3 Question 3

The new system of equations is now given as:

$$\begin{aligned} \dot{x} &= v && (6) \\ \dot{v} &= F_e \cos(\theta) - 2\gamma v - x^3 + x && (7) \end{aligned}$$

Consider $\gamma = 1, F_e = 1$ and $\theta = \pi$. The phase portrait is found in figure 8.
Consider $\gamma = 0, F_e = 1$ and $\theta = 2\pi$. The phase portrait is found in figure 9.
Consider $\gamma = 0, F_e = 0$ and $\theta = 0$. The phase portrait is found in figure 10.
Consider $\gamma = -1, F_e = -1$ and $\theta = \pi$. The phase portrait is found in figure 11.
Consider $\gamma = 1, F_e = 20$ and $\theta = 2\pi$. The phase portrait is found in figure 12.

All the phase portraits have been plotted using GeoGebra online plotting tool.

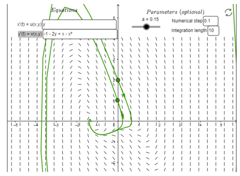


Figure 8

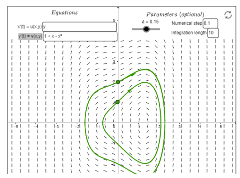


Figure 9

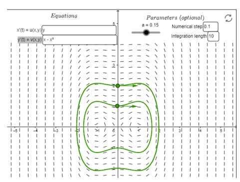


Figure 10

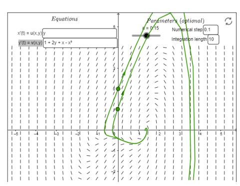


Figure 11

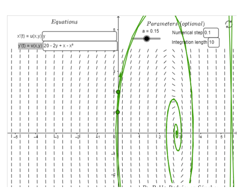


Figure 12

2 Cascade route to chaos and folding of strange attractors

2.1 Question 1

Consider $\gamma = 0.25, F_e = 0.43, \alpha = -1, \beta = 1, \omega = 1$ and $\theta = 0$. The phase portrait is found in figure 13.

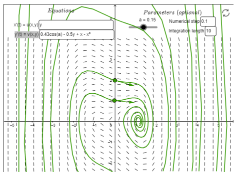


Figure 13

2.2 Question 2

Consider $\gamma, \alpha, \beta, \omega$ to be constant and $0.1 < F_e < 0.43$. The phase portraits are found in figure 14 and 15 for two different F_e values in the specified range.

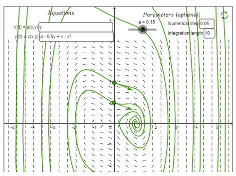


Figure 14

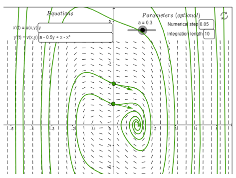


Figure 15

3 Chaos Recognition

3.1 Question 1

We use the pandas library to read the sample files in python. The peak values from each sample files are extracted. Hence, a sequence of maxima Z_n is formed. In order to determine whether the samples files are chaotic motion or not, we need to plot Z_{n+1} versus Z_n. Whereby, n ranges from 1 to the total number of peaks in each data file. Below are the plots for sample 1, 2 and 3 respectively.

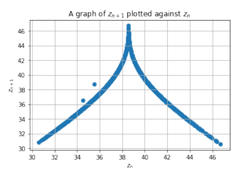


Figure 16: The plot of Z_{n+1} versus Z_n for sample 1

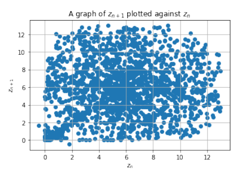


Figure 17: The plot of Z_{n+1} versus Z_n for sample 2

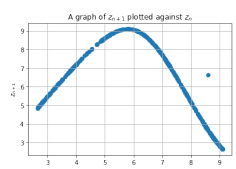


Figure 18: The plot of Z_{n+1} versus Z_n for sample 3

A very important feature of chaos is that it is not random. Therefore, it is deterministic. A Poincaré plot was made by Lorenz, whereby he plotted Z_{n+1} versus Z_n.[1] It was found out that the map is very similar to a tent map. There is a maximum Z value for which a cut-off happens, the system will then switch to the next lobe. Hence, it can be shown that the system switches between the 2 lobes chaotically. From the samples we have obtained, we can then deduce that only sample 1 seems to satisfy the chaotic motion described. The code for these plots can be found in appendix 3.

Appendices

Appendix 1

```
[1]: import matplotlib.pyplot as plt
     import numpy as np

[21]: x = np.arange(-3,3,0.01)
      v = (x**4)/4 - (x**2)/2

      plt.plot(x,v)
      plt.ylim(-1, 5)
      plt.xlabel("x")
      plt.ylabel("V(x)")
      plt.title("A plot of the potential V against displacement x")
      plt.grid()

      plt.show()
```

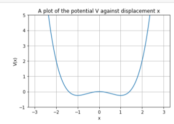


Figure 19: code for the plot of potential V against displacement x

Appendix 3

[96]:
```python
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```

[97]:
```python
#naming the sample file
sample1file = "sample1_NLP_CW2_sp2018.dat"
```

[98]:
```python
#read the sample file using pandas library
df1= pd.read_csv(sample1file,sep="  ", header= None)
```

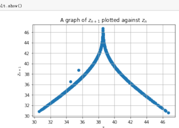
```
<ipython-input-98-575c0374d1b7>:2: ParserWarning: Falling back to the 'python'
engine because the 'c' engine does not support regex separators (separators > 1
char and different from '\s+' are interpreted as regex); you can avoid this
warning by specifying engine='python'.
  df1= pd.read_csv(sample1file,sep="  ", header= None)
```

[99]:
```python
#taking values from first column and second column of the sample files
x = df1[0].values
y = df1[1].values
```

[100]:
```python
#a "for loop" is used to index the values whereby the first and last values are
 ↪being skipped.
z= []
for index, i in enumerate(y):
    if index == 0 or index == len(y)-1:
        continue
    else:

#by skipping the first and last values, it is possible to find the peak values
 ↪for any other number inside the sequence.
#there is a peak if the current value is greater than the value before and after
 ↪it. Slicing method is used below.
        if y[index-1] < y[index] and y[index+1] < y[index]:

#peak values are added to the list z
            z.append(y[index])
```

[101]:
```python
#the list is converted to an array
z = np.array(z)
```

[102]:
```python
#a scatter plot is made whereby the values for Z_n+1 is plotted against Z_n.
plt.scatter(z[0:len(z)-1],z[1:])
plt.xlabel("$z_n$")
plt.ylabel("$z_{n+1}$")
plt.title("A graph of $z_{n+1}$ plotted against $z_n$")
plt.grid()

plt.show()
```

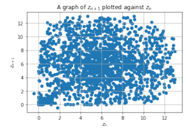


[103]:
```python
#repeat method for sample 2 and sample 3
sample2file = "sample2_NLP_CW2_sp2018.dat"
sample3file = "sample3_NLP_CW2_sp2018.dat"

df2= pd.read_csv(sample2file,sep="  ", header= None)
df3= pd.read_csv(sample3file,sep="  ", header= None)

a = df2[0].values
b = df2[1].values
c = df3[0].values
d = df3[1].values
```

```python
e= []
for index, i in enumerate(b):
    if index == 0 or index == len(b)-1:
        continue
    else:

        if b[index-1] < b[index] and  b[index+1] < b[index]:

            e.append(b[index])

e = np.array(e)
plt.scatter(e[0:len(e)-1],e[1:])
plt.xlabel("$z_n$")
plt.ylabel("$z_{n+1}$")
plt.title("A graph of $z_{n+1}$ plotted against $z_n$")
plt.grid()

plt.show()


f= []
for index, i in enumerate(d):
    if index == 0 or index == len(d)-1:
        continue
    else:

        if d[index-1] < d[index] and d[index+1] < d[index]:

            f.append(d[index])

f = np.array(f)
plt.scatter(f[0:len(f)-1],f[1:])
plt.xlabel("$z_n$")
plt.ylabel("$z_{n+1}$")
plt.title("A graph of $z_{n+1}$ plotted against $z_n$")
plt.grid()

plt.show()
```

```
<ipython-input-103-52eeae7b3906>:5: ParserWarning: Falling back to the 'python'
engine because the 'c' engine does not support regex separators (separators > 1
char and different from '\s+' are interpreted as regex); you can avoid this
warning by specifying engine='python'.
  df2= pd.read_csv(sample2file,sep="  ", header= None)
<ipython-input-103-52eeae7b3906>:6: ParserWarning: Falling back to the 'python'
engine because the 'c' engine does not support regex separators (separators > 1
```

```
char and different from '\s+' are interpreted as regex); you can avoid this
warning by specifying engine='python'.
  df3= pd.read_csv(sample3file,sep="  ", header= None)
```



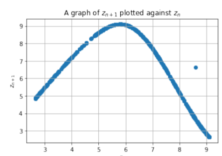


Figure 20: code for the 3 plots of sample 1, 2 and 3 respectively to determine chaotic motion.

References

[1] Edward N Lorenz. Deterministic nonperiodic flow. *Journal of atmospheric sciences*, 20(2):130–141, 1963.