Universidad de Sonora
División de Ciencias Exactas y Naturales
Departamento de Física

Actividad 7: Sistema de Masas con Oscilaciones no Lineales y Forzadas
Física Computacional I

Rolando Abdel Fimbres Grijalva

14 de abril de 2018

1. Introducción

En la actividad se estudió la modelación de un fenómeno físico. Con la ayuda del lenguaje de programación Python. El caso de estudio fue un sistema de resortes acoplado. El método con el que podremos determinar la solución de éste sistema es mediante la aplicación de métodos numéricos. De esa forma podremos acercarnos a una solución más exacta.
A continuación se escribirá una breve síntesis del artículo de Temple H. Fay y Sarah Duncan Graham. También se añadirán imágenes de los códigos utilizados para resolver las problemáticas.

2. Síntesis

Estudiar las ecuaciones diferenciales es algo muy importante ya que para todo estudiante de física, la gran mayoría de fenómenos pueden modelarse gracias a ellas. Es muy importante contar con la ayuda de una herramienta que nos ayude para su solución. En éste caso sería la programación utilizando métodos numéricos.
En el artículo podemos observar que se desarrolla un problema por así llamar çlásico". Éste es el de dos resortes con dos pesas que se encuentran colgadas del techo. Según el artículo debemos recurrir a la Ley de Hooke, pues las fuerzas restaurativas se comportan de acuerdo a ésta ley. El problema se maneja con la ayuda de dos ecuaciones diferenciales lineales de segundo orden acopladas. Al nosotros diferenciar y sustituir valores, observamos que el movimiento de cada pesa es determinado por una ecuación diferencial de cuarto orden.

2.1. El Resorte Acoplado

El modelo consta de dos resortes y dos pesas que se cuelgan del techo. debemos tomar en cuenta que cada resorte cuenta con una constante en este caso: k_1 y k_2. Y masas: m_1 y m_2.

2.2. Ley de Hooke

Asumiremos que existen en el caso oscilaciones pequeñas. Las fuerzas restaurativas resultarían: $-k_1 l_1$ y $-k_2 l_2$ en donde l_1 y l_2 son las compresiones o elongamientos de los resortes. Debido a que la masa superior lleva ambos resortes, hay dos fuerzas de restauración actuando.
Tenemos presente una fuerza de restauración hacia arriba $k_1 l_1$ ejercida por el elongamiento del primer resorte. Después del segundo una con forma $k_2(x_2 - x_1)$. La segunda masa solo cuenta con la fuerza de restauración del segundo resorte. Asumiendo que no hay fuerzas de amortiguamiento presentes, las Leyes de Newton que representan a los movimientos son:

$$m_1 \ddot{x_1} = -k_1 x_1 - k_2(x_1 - x_2)$$
$$m_2 \ddot{x_2} = -k_2(x_2 - x_1)$$

Para encontrar una ecuación para x_1 y x_2 que no dependen una de otra se resuelve la ecuación de x_2 y se sustituye en la de x_1, obteniendo las ecuaciones:

$$m_1 m_2 x_1^{(4)} + (m_2 k_1 + k_2(m_1 + m_2))\ddot{x_1} + k_1 k_2 x_1 = 0$$
$$m_1 m_2 x_2^{(4)} + (m_2 k_1 + k_2(m_1 + m_2))\ddot{x_2} + k_1 k_2 x_2 = 0$$

2.3. Añadiendo no linealidad

Al asumir que las fuerzas restauradoras son no lineales y tienen la forma $-kx + \mu x^3$, entonces las ecuaciones quedan:

$$m_1\ddot{x_1} = -\delta_1\dot{x_1} - k_1x_1 + \mu_1x_1^3 - k_2(x_1 - x_2) + \mu_2(x_2 - x_1)^3$$
$$m_2\ddot{x_2} = -\delta_2\dot{x_2} - k_2(x_2 - x_1) + \mu_2(x_2 - x_1)^3$$

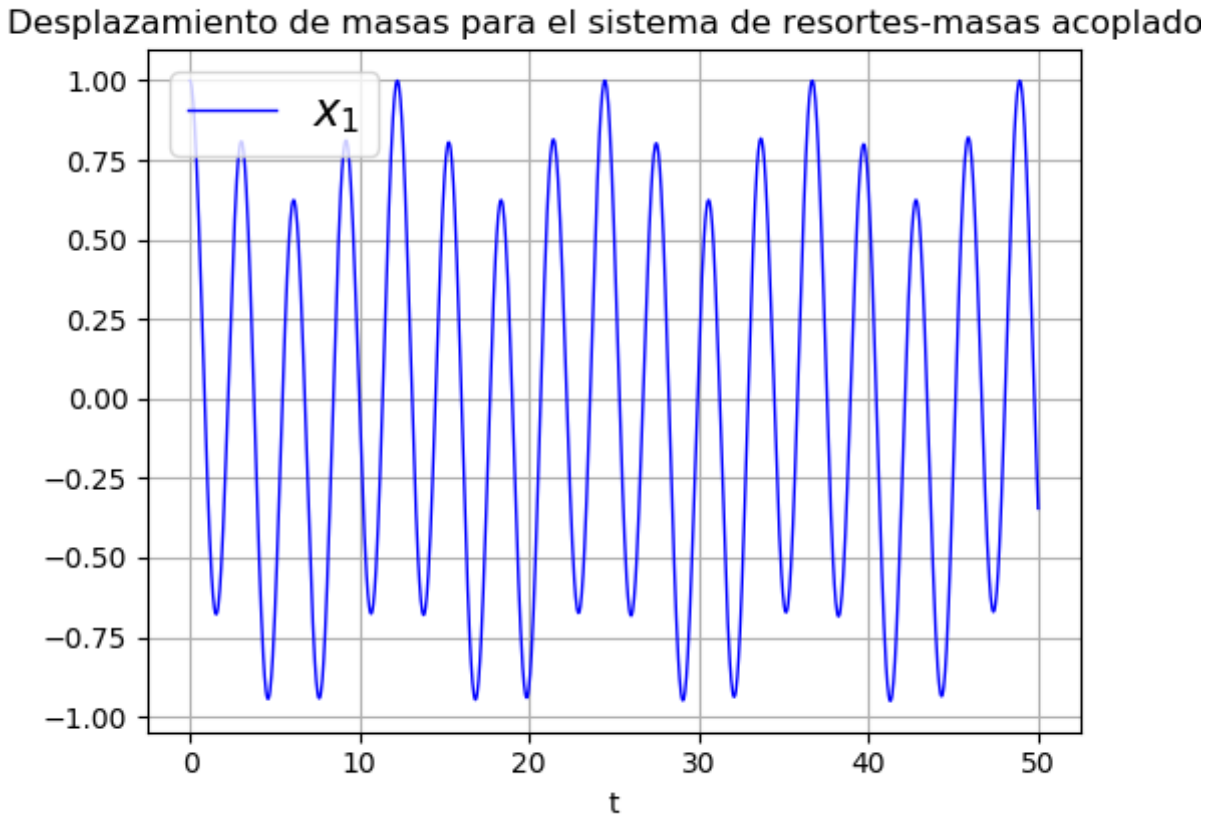
La solución se complica mucho más que en los casos de linealidad.

```
def vectorfield(w, t, p):
    """
    Defines the differential equations for the coupled spring-mass system.

    Arguments:
        w :  vector of the state variables:
                  w = [x1,y1,x2,y2]
        t :  time
        p :  vector of the parameters:
                  p = [m1,m2,k1,k2,L1,L2,b1,b2,d1,d2,u1,u2]
    """
    x1, y1, x2, y2 = w
    m1, m2, k1, k2, L1, L2, b1, b2, u1, u2 = p

    # Create f = (x1',y1',x2',y2'):
    f = [y1,
         (-b1 * y1 - k1 * (x1 - L1) + u1 * x1 * x1 * x1 - k2 * (x1 - x2 - L2) + u2 * (x1 - x2) * (x1
- x2) * (x1 - x2)) / m1,
         y2,
         (-b2 * y2 - k2 * (x2 - x1 - L2) + u2 * (x2 - x1) * (x2 - x1) * (x2 - x1)) / m2]
    return f
```

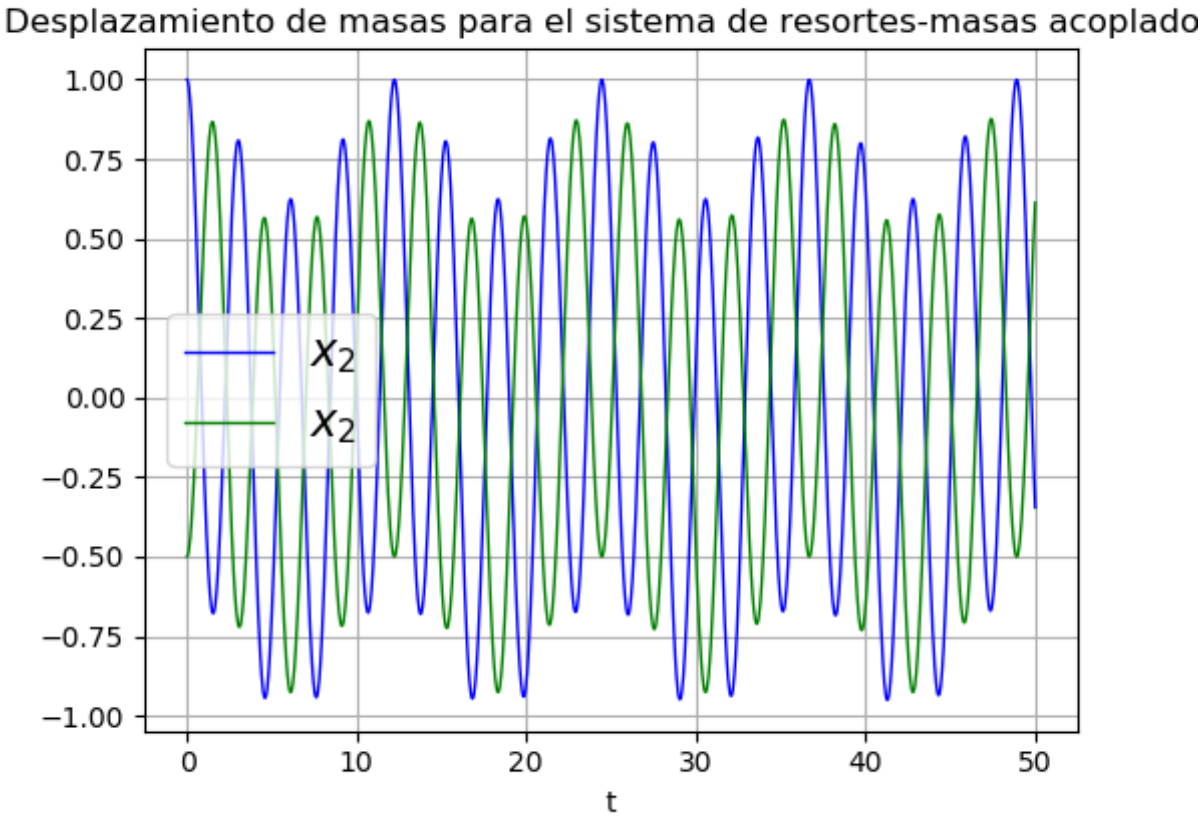
Ejemplo 3.1: Asuma que $m_1 = m_2 = 1$. Describe el movimiento para resortes con constantes $k_1 = 0{,}4$ y $k_2 = 1{,}808$, coeficientes de amortiguamiento $\delta_1 = 0$ y $\delta_2 = 0$, coeficientes de no linealidad $\mu_1 = -1/6$ y $\mu_2 = -1/10$, y con condiciones iniciales $(x_1(0), \dot{x_1}(0), x_2(0), \dot{x_2}(0)) = (1, 0, -1/2, 0)$.
Debido a la falta de amortiguamiento, existe un comportamiento oscilatorio que aparenta periodicidad.

```
# Use ODEINT to solve the differential equations defined by the vector field
from scipy.integrate import odeint

# Parameter values
# Masses:
m1 = 1.0
m2 = 1.0
# Spring constants
k1 = 0.4
k2 = 1.808
# Natural lengths
L1 = 0.0
L2 = 0.0
# Coeficientes de amortiguamiento
b1 = 0.0
b2 = 0.0
# Coeficientes de no-linealidad
u1 = -1.0/6.0
u2 = -0.1

# Initial conditions
# x1 and x2 are the initial displacements; y1 and y2 are the initial velocities
x1 = 1.0
y1 = 0.0
x2 = -0.5
y2 = 0.0

# ODE solver parameters
abserr = 1.0e-8
relerr = 1.0e-6
stoptime = 50
numpoints = 1250


# Create the time samples for the output of the ODE solver.
# I use a large number of points, only because I want to make
# a plot of the solution that looks nice.
t = [stoptime * float(i) / (numpoints - 1) for i in range(numpoints)]

# Pack up the parameters and initial conditions:
p = [m1, m2, k1, k2, L1, L2, b1, b2, u1, u2]
w0 = [x1, y1, x2, y2]

# Call the ODE solver.
wsol = odeint(vectorfield, w0, t, args=(p,),
              atol=abserr, rtol=relerr)

with open('31.dat', 'w') as f:
    # Print & save the solution.
    for t1, w1 in zip(t, wsol):
        print (t1, w1[0], w1[1], w1[2], w1[3], file=f)
```

De este modo obtenemos los resultados siguientes.




Desplazamiento de masas para el sistema de resortes-masas acoplado
x_1
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10
20
30
40
50
t


Desplazamiento de masas para el sistema de resortes-masas acoplado
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10
20
30
40
50
t
x_2
x_2

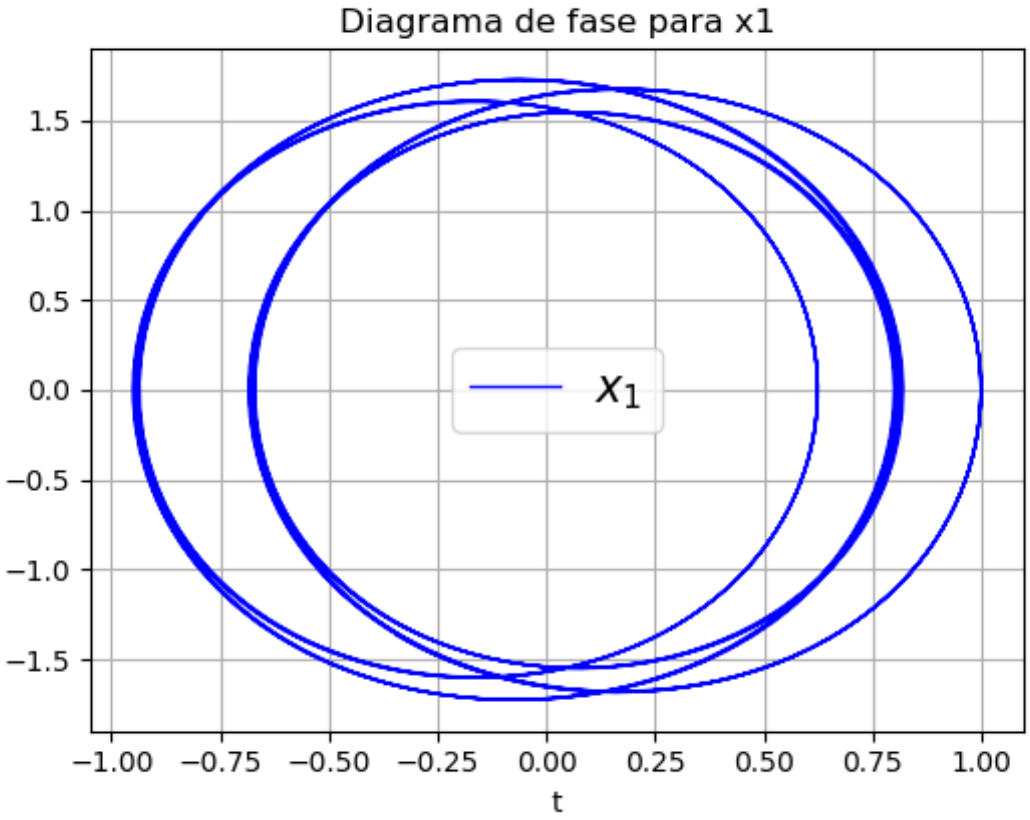


Diagrama de fase para x1
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
x_1
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
t

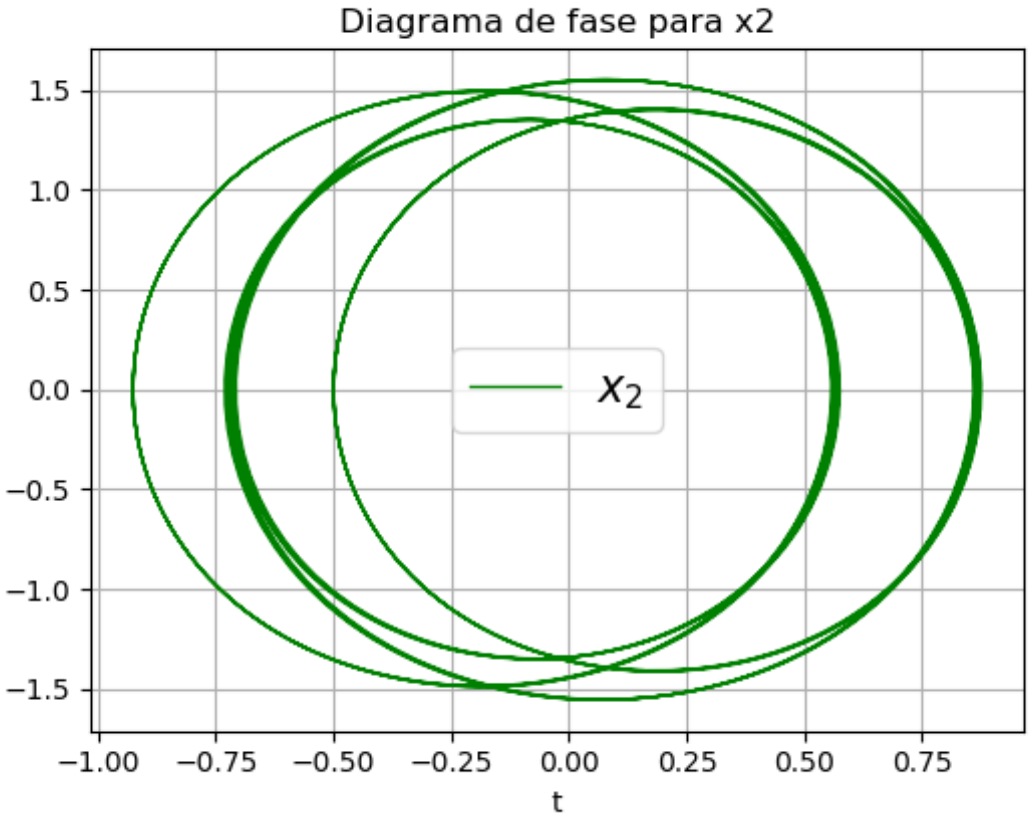
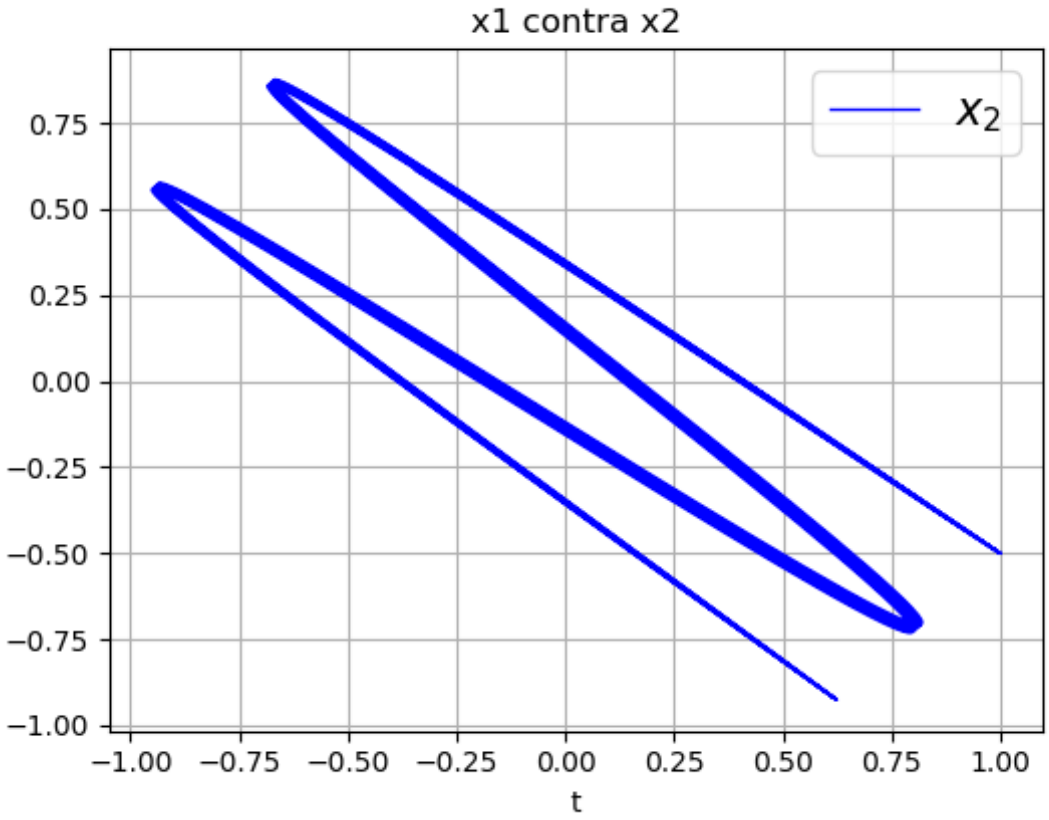


Diagrama de fase para x2
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
t
x_2



Ejemplo 3.2: Asuma que $m_1 = m_2 = 1$. Describe el movimiento para resortes constantes $k_1 = 0{,}4$ y $k_2 = 1{,}808$, coeficientes de amortiguamiento $\delta_1 = 0$ y $\delta_2 = 0$, coeficientes de no linealidad $\mu_1 = -1/6$ y $\mu_2 = -1/10$, y con condiciones iniciales $(x_1(0), \dot{x_1}(0), x_2(0), \dot{x_2}(0)) = (-0{,}5, 1/2, 3{,}001, 5{,}9)$.

```python
from scipy.integrate import odeint

# Parameter values
# Masses:
m1 = 1.0
m2 = 1.0
# Spring constants
k1 = 0.4
k2 = 1.808
# Natural lengths
L1 = 0.0
L2 = 0.0
# Coeficientes de amortiguamiento
b1 = 0.0
b2 = 0.0
# Coeficientes de no-linealidad
u1 = -1.0/6.0
u2 = -0.1

# Initial conditions
# x1 and x2 are the initial displacements; y1 and y2 are the initial velocities
x1 = -0.5
y1 = 1.0/2.0
x2 = 3.001
y2 = 5.9
```

Obteniendo los resultados:

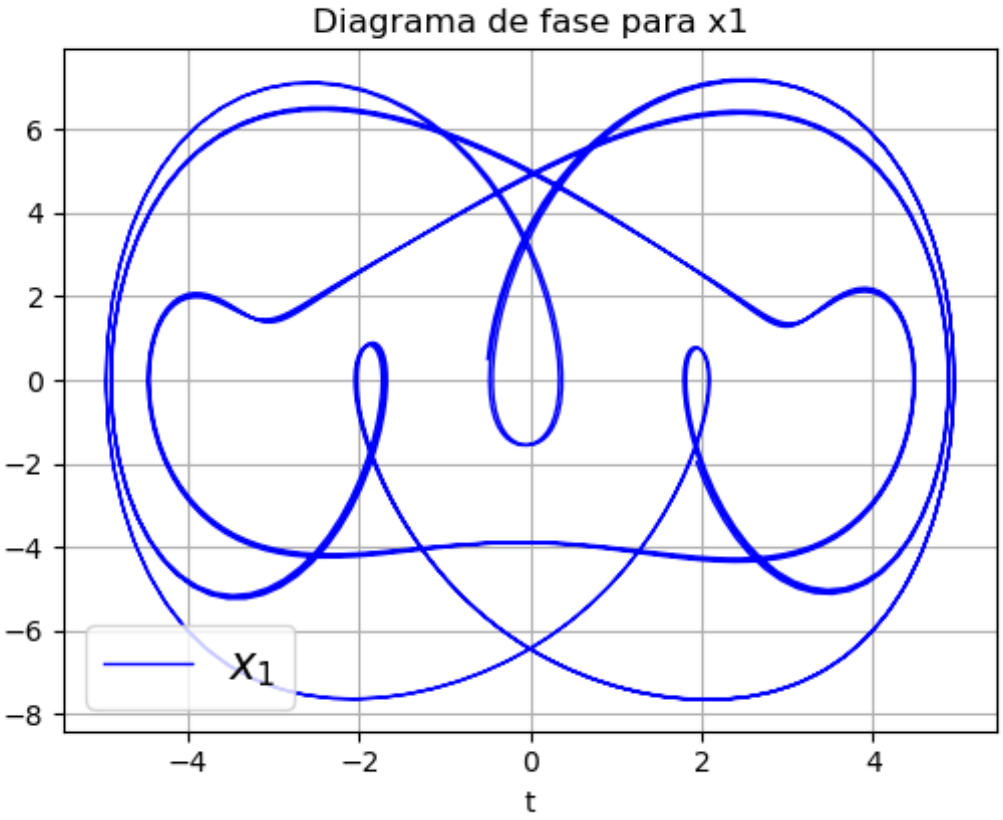


Diagrama de fase para x1
6
4
2
0
−2
−4
−6
−8
−4
−2
0
2
4
t
x_1

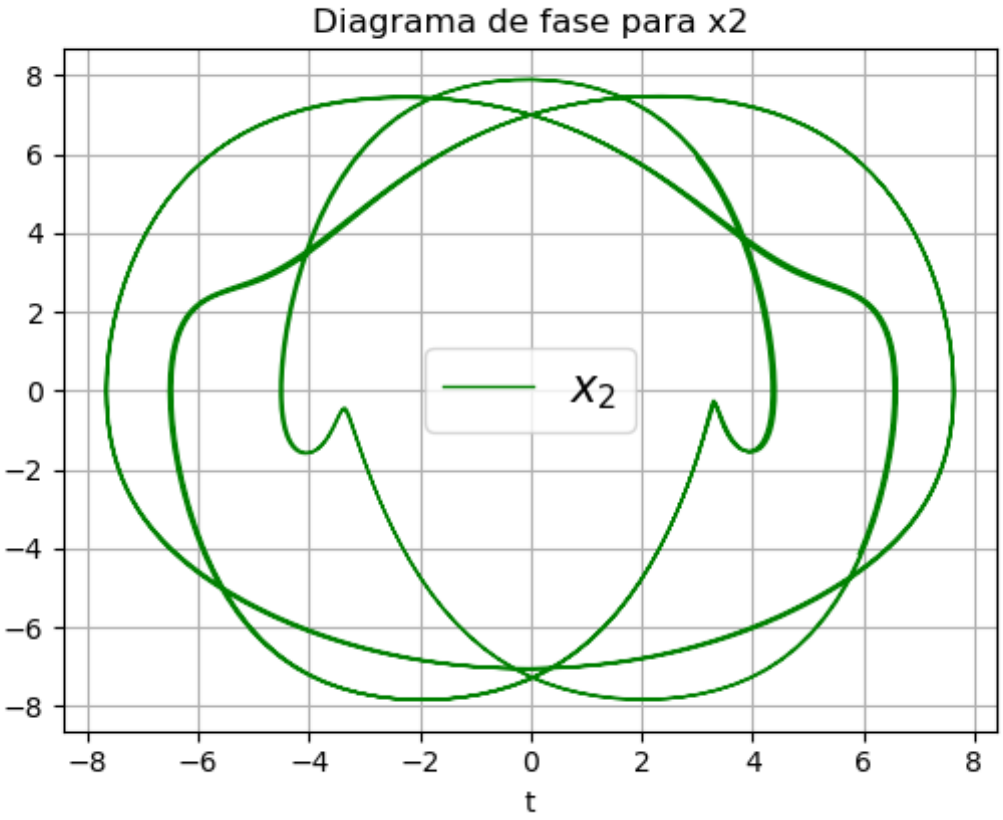
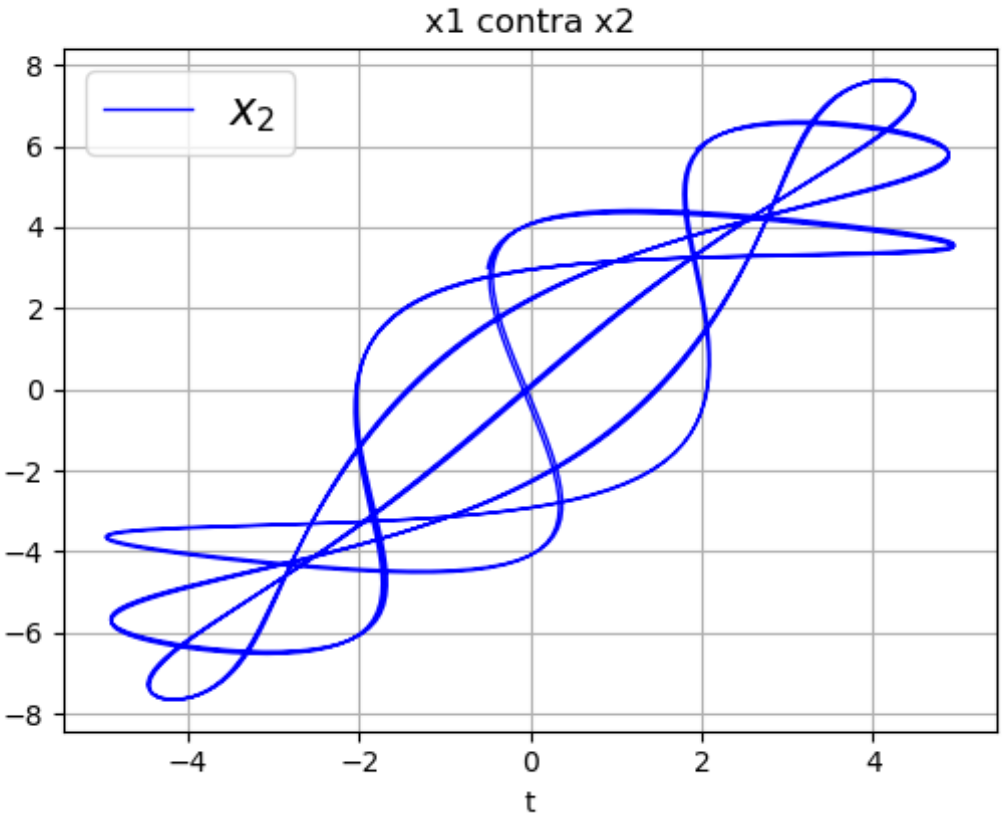


Diagrama de fase para x2
8
6
4
2
0
−2
−4
−6
−8
−8
−6
−4
−2
0
2
4
6
8
t
x_2



Ejemplo 3.3: Asuma que $m_1 = m_2 = 1$. Describe el movimiento para resortes constantes $k_1 = 0{,}4$ y $k_2 = 1{,}808$, coeficientes de amortiguamiento $\delta_1 = 0$ y $\delta_2 = 0$, coeficientes de no linealidad $\mu_1 = -1/6$ y $\mu_2 = -1/10$, y con condiciones iniciales $(x_1(0), \dot{x_1}(0), x_2(0), \dot{x_2}(0)) = (-0{,}6, 1/2, 3{,}001, 5{,}9)$.

```
from scipy.integrate import odeint

# Parameter values
# Masses:
m1 = 1.0
m2 = 1.0
# Spring constants
k1 = 0.4
k2 = 1.808
# Natural lengths
L1 = 0.0
L2 = 0.0
# Friction coefficients
b1 = 0.0
b2 = 0.0
# Nonlinear components
u1 = -1.0/6.0
u2 = -0.1

# Initial conditions
# x1 and x2 are the initial displacements; y1 and y2 are the initial velocities
x1 = -0.6
y1 = 0.5
x2 = 3.001
y2 = 5.9
```

Dando los siguientes resultados:

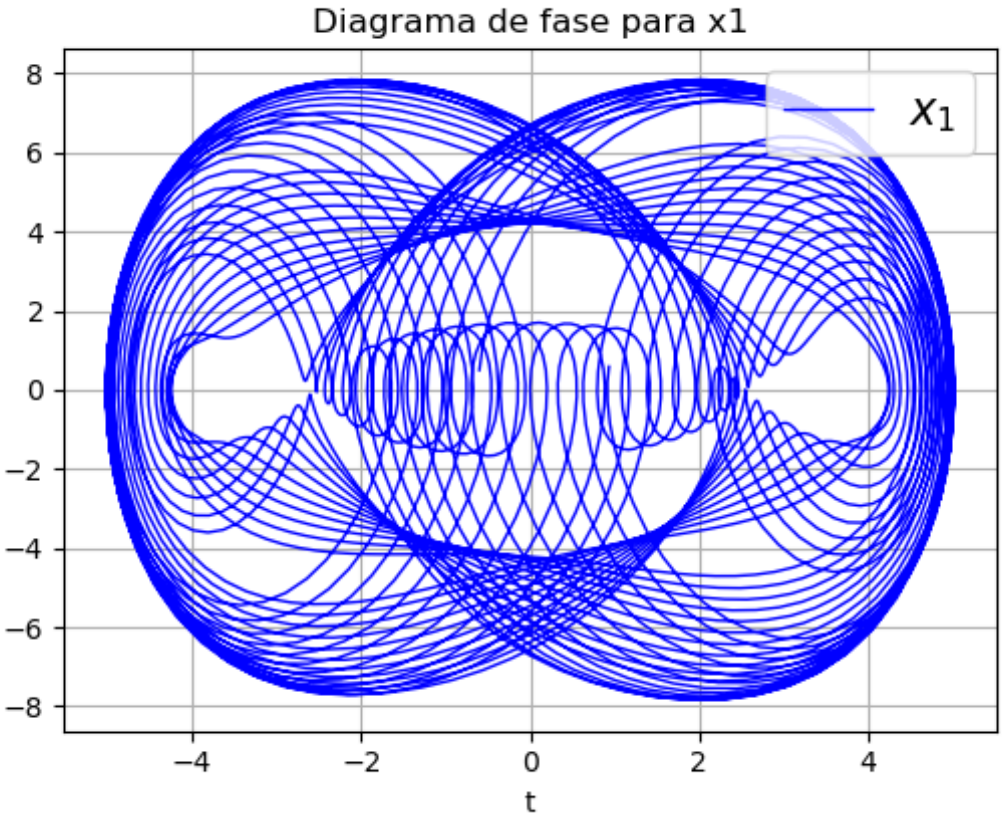


Diagrama de fase para x1
x_1
8
6
4
2
0
−2
−4
−6
−8
−4
−2
0
2
4
t

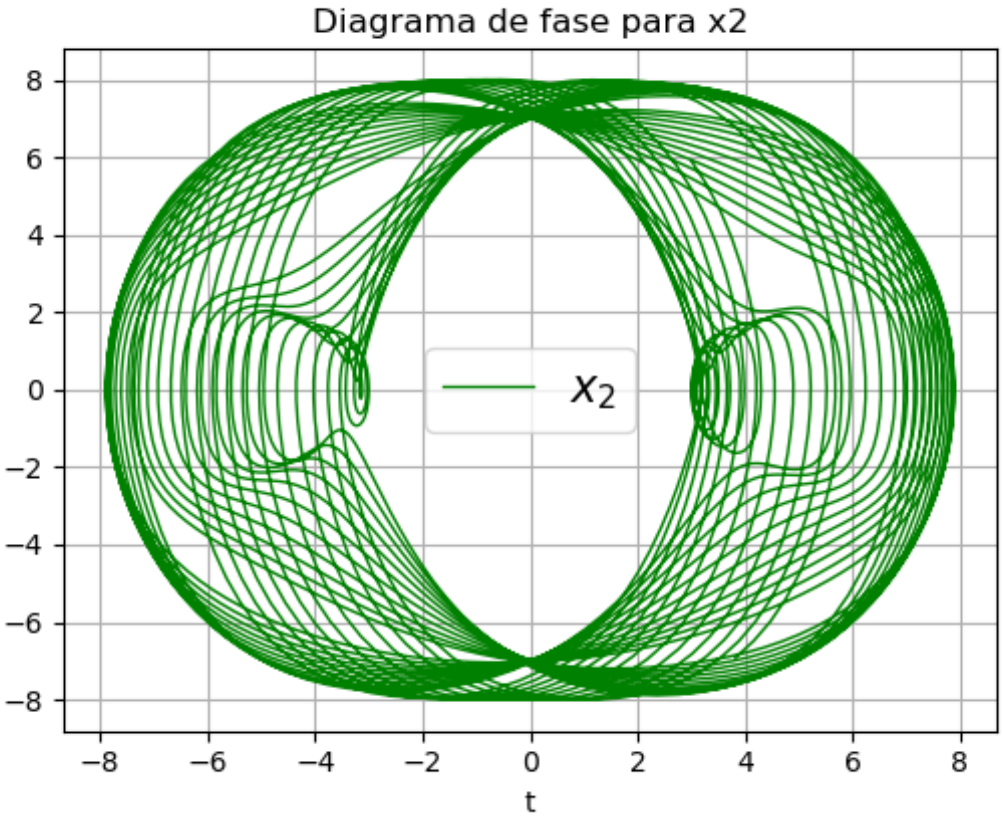
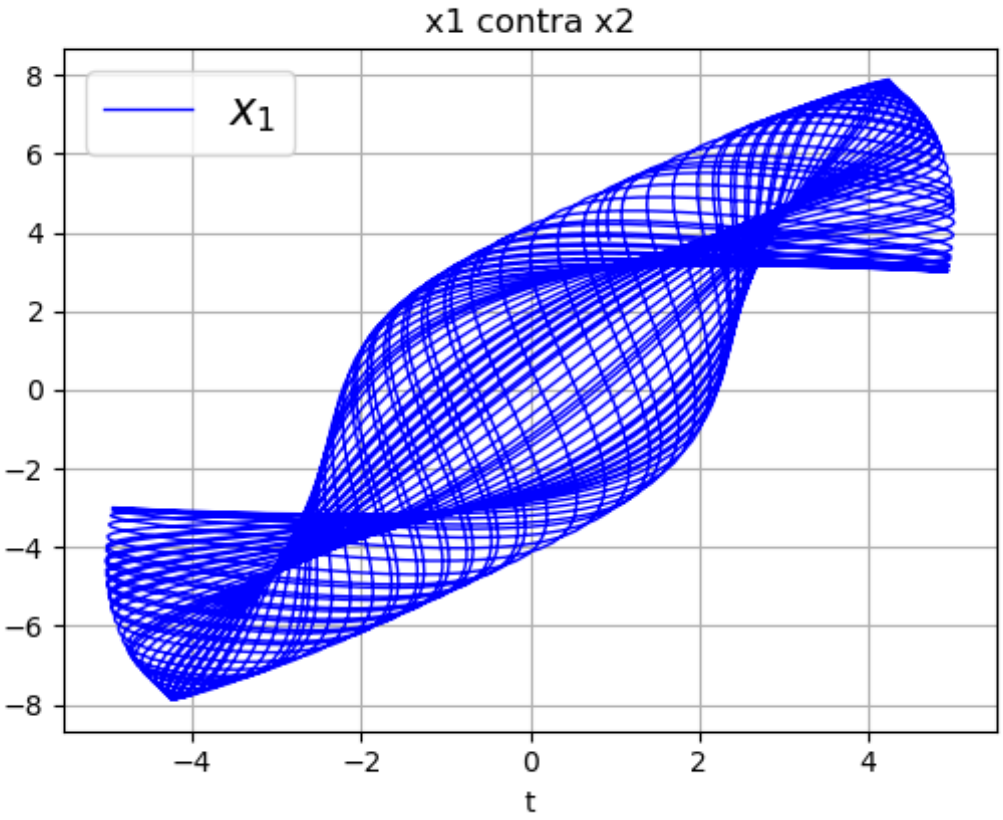


Diagrama de fase para x2
8
6
4
2
0
−2
−4
−6
−8
−8
−6
−4
−2
0
2
4
6
8
t
x_2



2.4. Añadiendo forzamiento

Para agregar una fuerza externa al sistema se aplica una fuerza con forma senoidal: $F\cos(wt)$. Nuestras ecuaciones quedan:

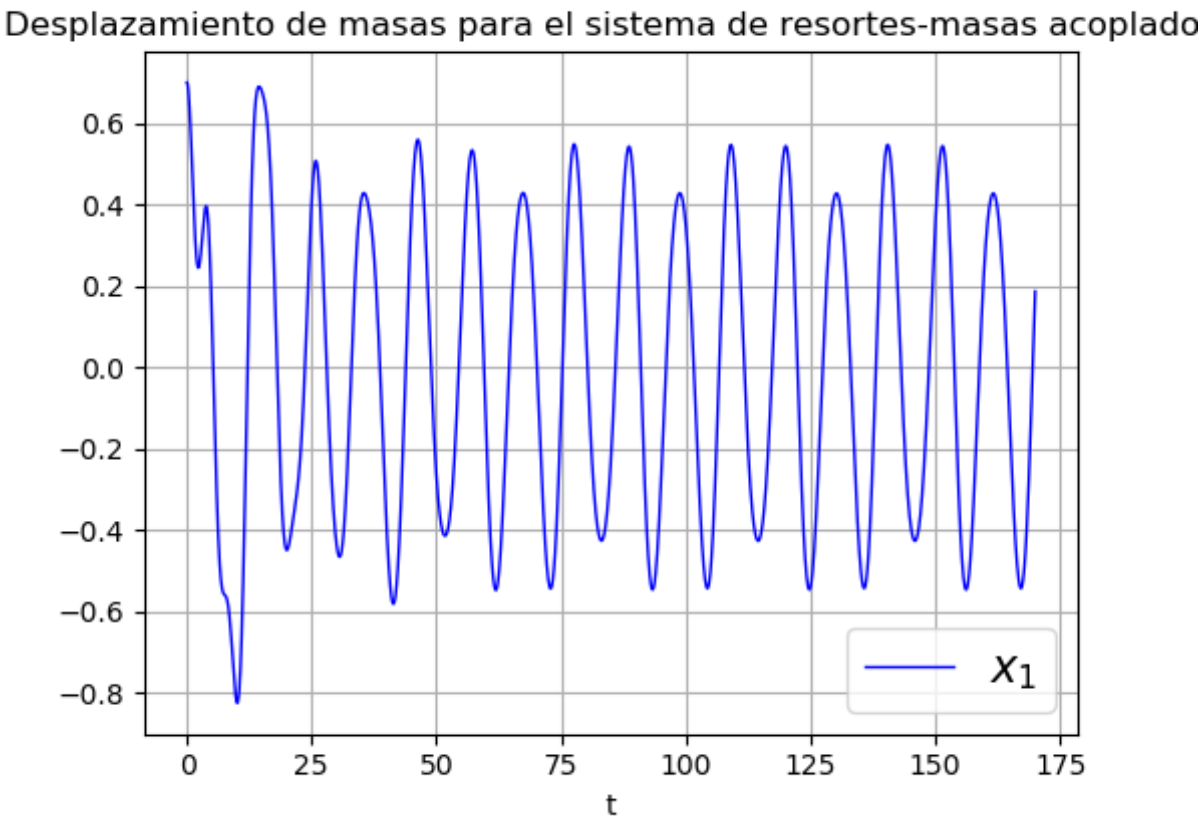
$$m_1\ddot{x_1} = -\delta_1\dot{x_1} - k_1x_1 + \mu_1x_1^3 - k_2(x_1 - x_2) + \mu_2(x_2 - x_1)^3 + F_1\cos(w_1t)$$
$$m_2\ddot{x_2} = -\delta_2\dot{x_2} - k_2(x_2 - x_1) + \mu_2(x_2 - x_1)^3 + F_2\cos(w_2t)$$

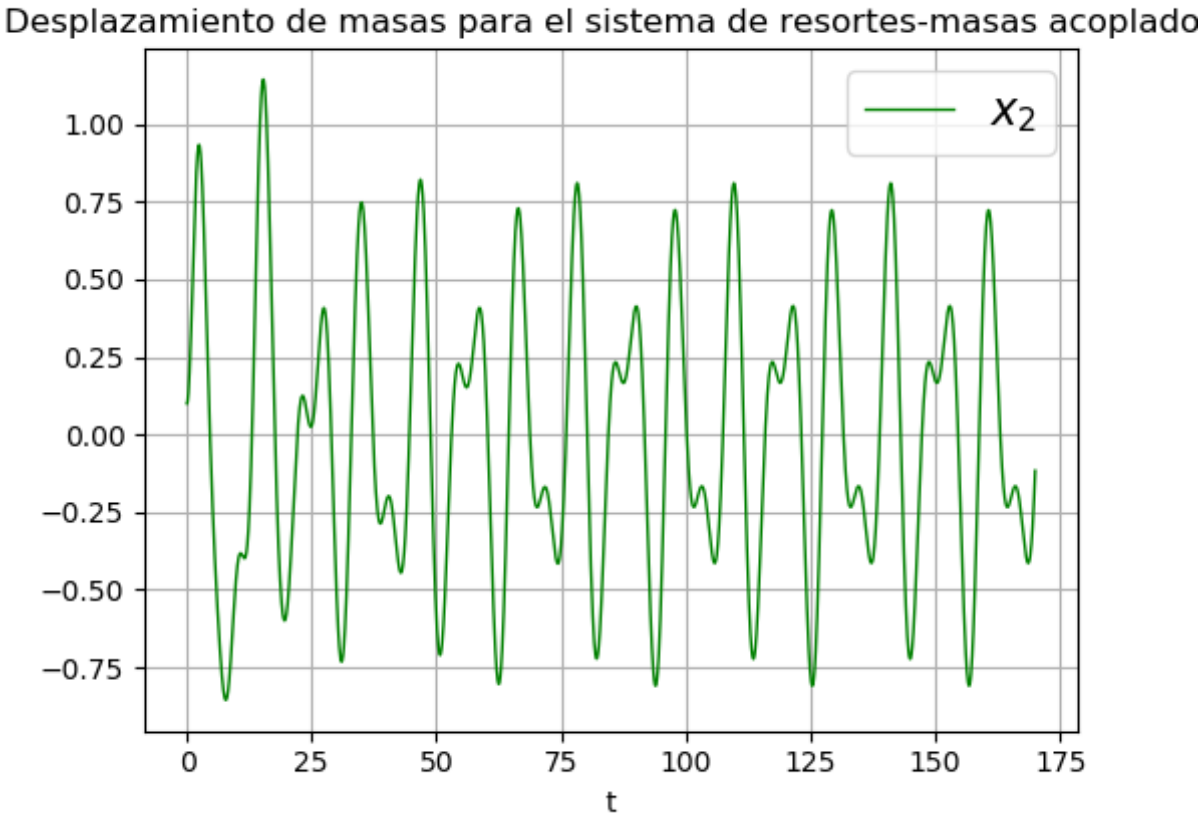
Ejemplo 4.1: Asuma $m_1 = m_2 = 1$. Describe el movimiento con constantes $k_1 = 2/5$ y $k_2 = 1$, coeficientes de amortiguamiento $\delta_1 = 1/10$ y $\delta_2 = 1/5$, coeficientes de no linealidad $\mu_1 = 1/6$ y $\mu_2 = 1/10$, fuerzas de amplitud $F_1 = 1/3$ y $F_2 = 1/5$, frecuencias de forzamiento $w_1 = 1$ y $w_2 = 3/5$ y con condiciones iniciales $(x_1(0), \dot{x_1}(0), x_2(0), \dot{x_2}(0)) = (0{,}7, 0, 0{,}1, 0)$.
Modificamos el campo vectorial:

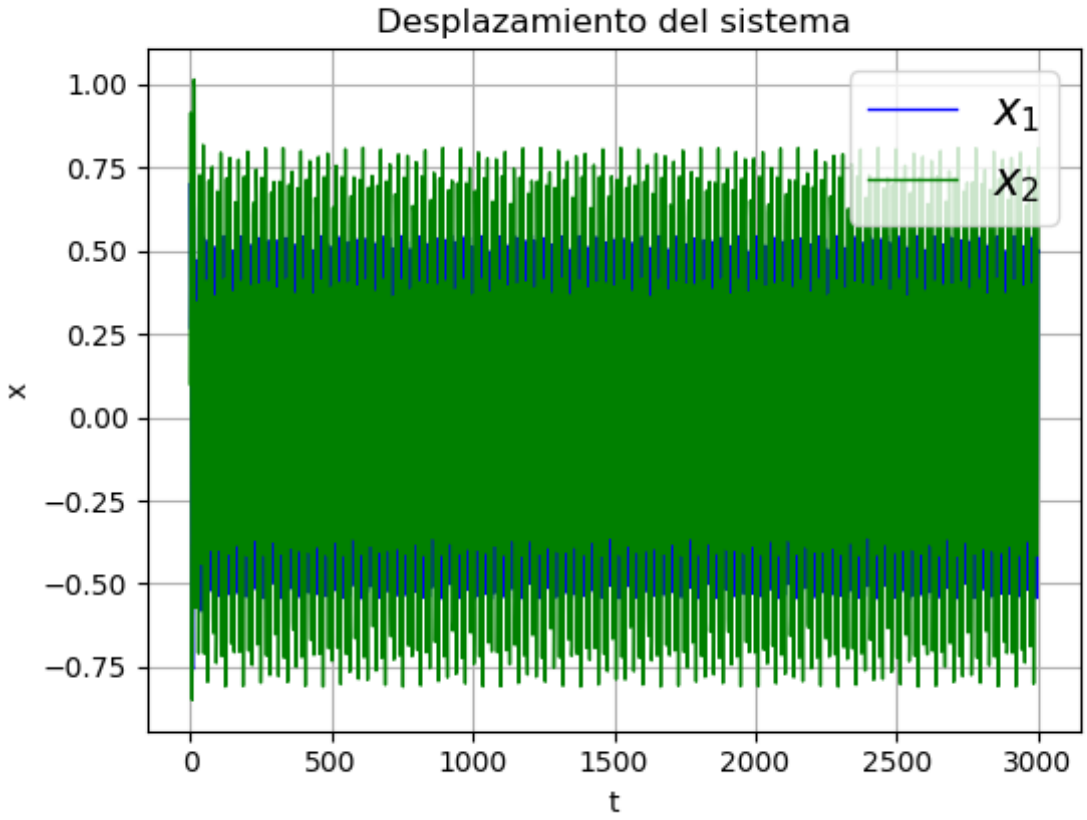
```
import numpy as np
def vectorfield(w, t, p):
    """
    Defines the differential equations for the coupled spring-mass system.

    Arguments:
        w :  vector of the state variables:
                  w = [x1,y1,x2,y2]
        t :  time
        p :  vector of the parameters:
                  p = [m1,m2,k1,k2,L1,L2,b1,b2,d1,d2,u1,u2]
    """
    x1, y1, x2, y2 = w
    m1, m2, k1, k2, L1, L2, b1, b2, u1, u2, f1, f2, q1, q2 = p

    # Create f = (x1',y1',x2',y2'):
    f = [y1,
         (-b1 * y1 - k1 * (x1 - L1) + u1 * x1**3 - k2 * (x1 - x2 - L2) + u2 * (x1 - x2)**3 + f1*(np.c
os(q1*t))) / m1,
         y2,
         (-b2 * y2 - k2 * (x2 - x1 - L2) + u2 * (x2 - x1)**3 + f2*(np.cos(q2*t))) / m2]
    return f
```

```
from scipy.integrate import odeint
import numpy as np

# Parameter values
# Masses:
m1 = 1.0
m2 = 1.0
# Spring constants
k1 = 2.0/5.0
k2 = 1.0
# Natural lengths
L1 = 0.0
L2 = 0.0
# Friction coefficients
b1 = 1.0/10.0
b2 = 1.0/5.0
# Nonlinear components
u1 = 1.0/6.0
u2 = 0.1
# Forcing Amplitudes
f1 = 1.0/3.0
f2 = 1.0/5.0
# Forcing Frequencies
q1 = 1.0
q2 = 3.0/5.0

# Initial conditions
# x1 and x2 are the initial displacements; y1 and y2 are the initial velocities
x1 = 0.7
y1 = 0.0
x2 = 0.1
y2 = 0.0
```

Obteniendo los resultados:


Desplazamiento de masas para el sistema de resortes-masas acoplado
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
0
25
50
75
100
125
150
175
t
x_1


Desplazamiento de masas para el sistema de resortes-masas acoplado
x_2
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0
25
50
75
100
125
150
175
t


Desplazamiento del sistema
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
x
0
500
1000
1500
2000
2500
3000
t
x_1
x_2

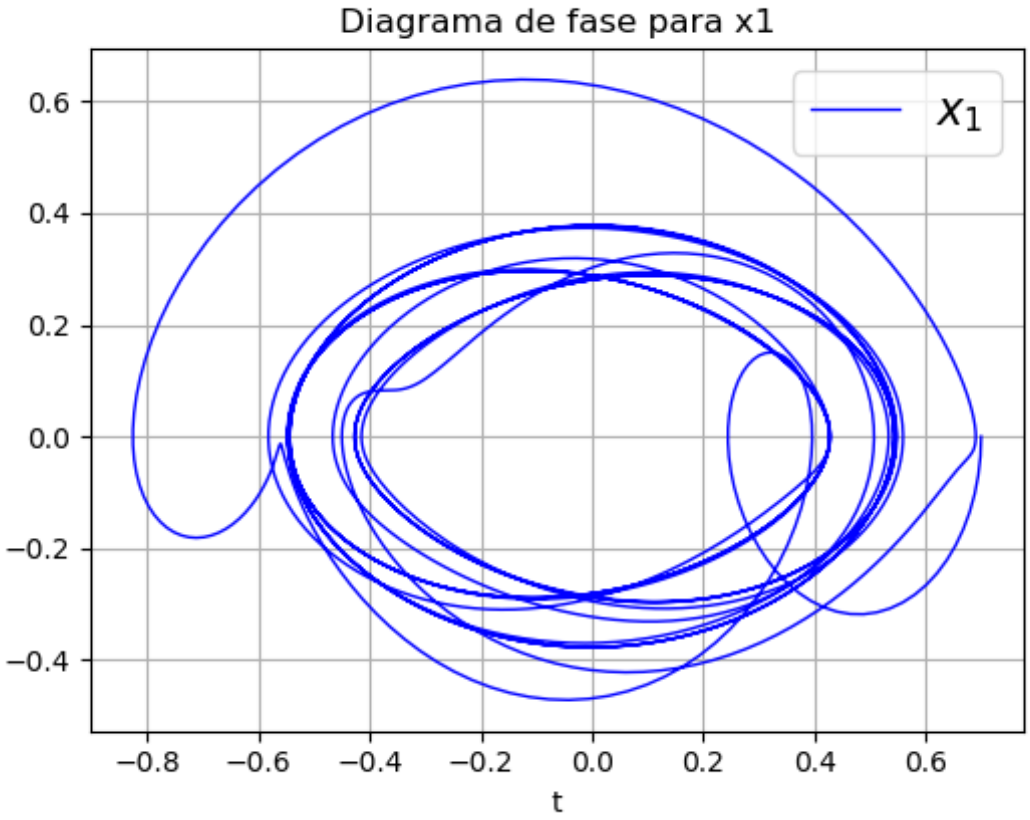


Diagrama de fase para x1
x_1
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.8
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
t

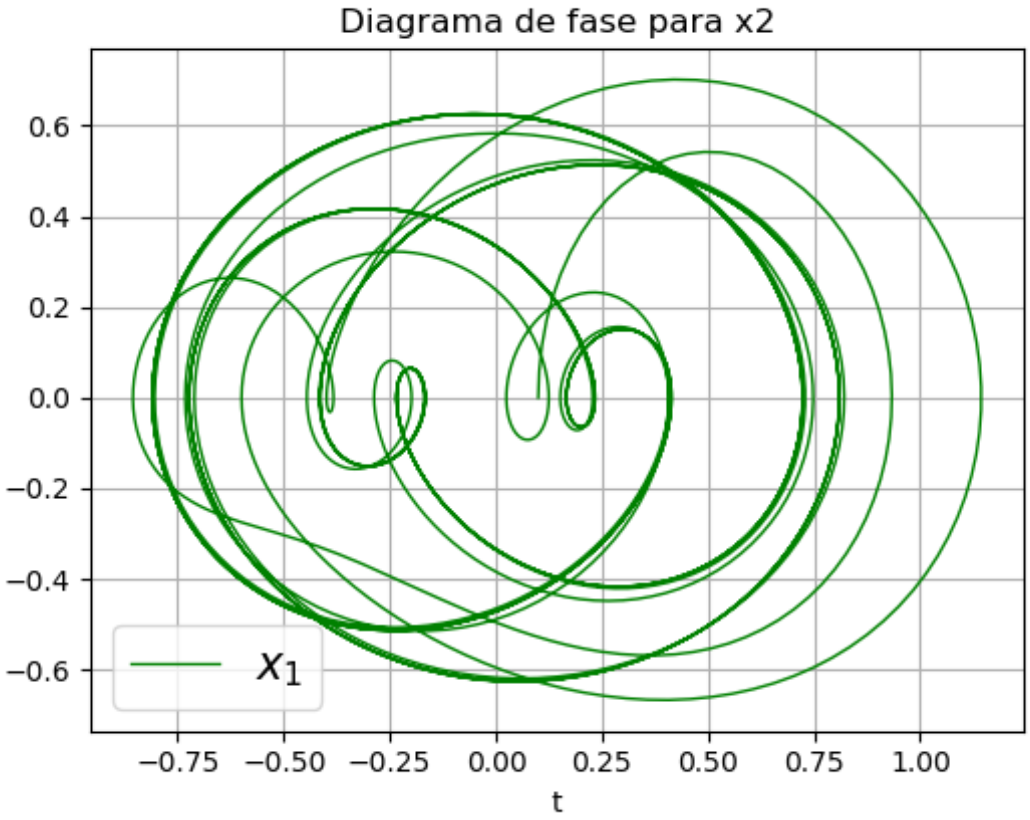
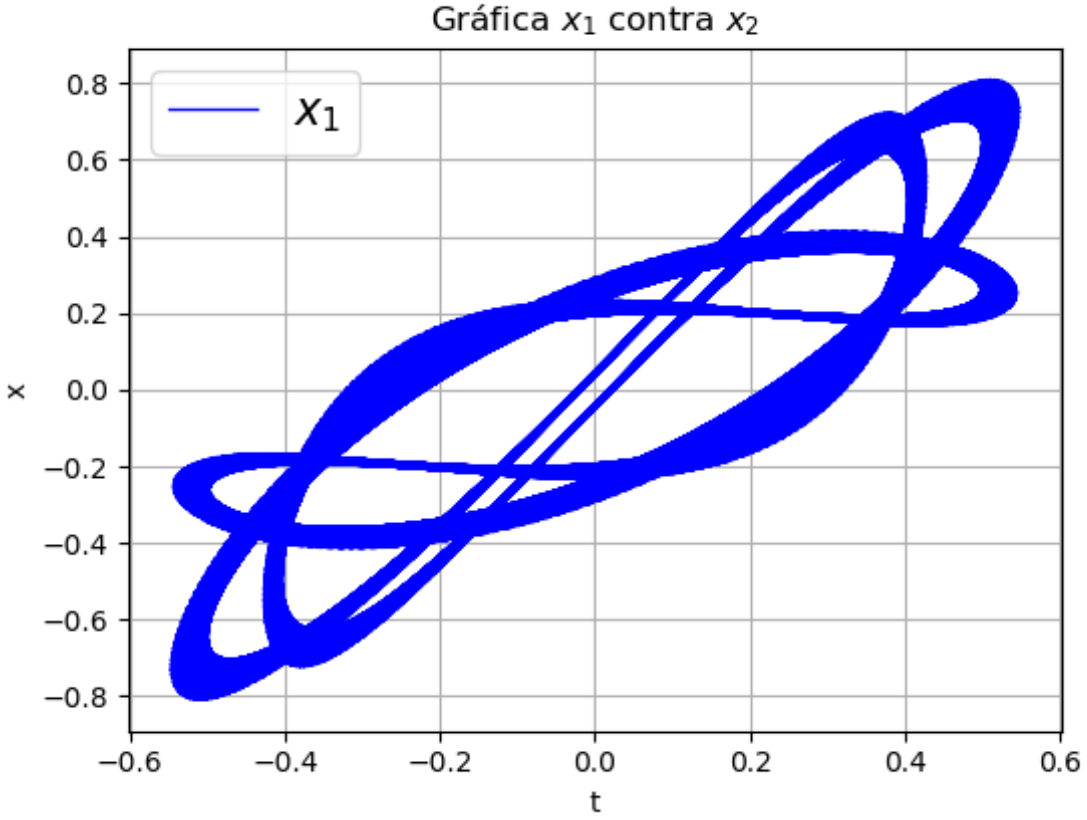


Diagrama de fase para x2
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
t
x_1


Gráfica x_1 contra x_2
x_1
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
x
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
t

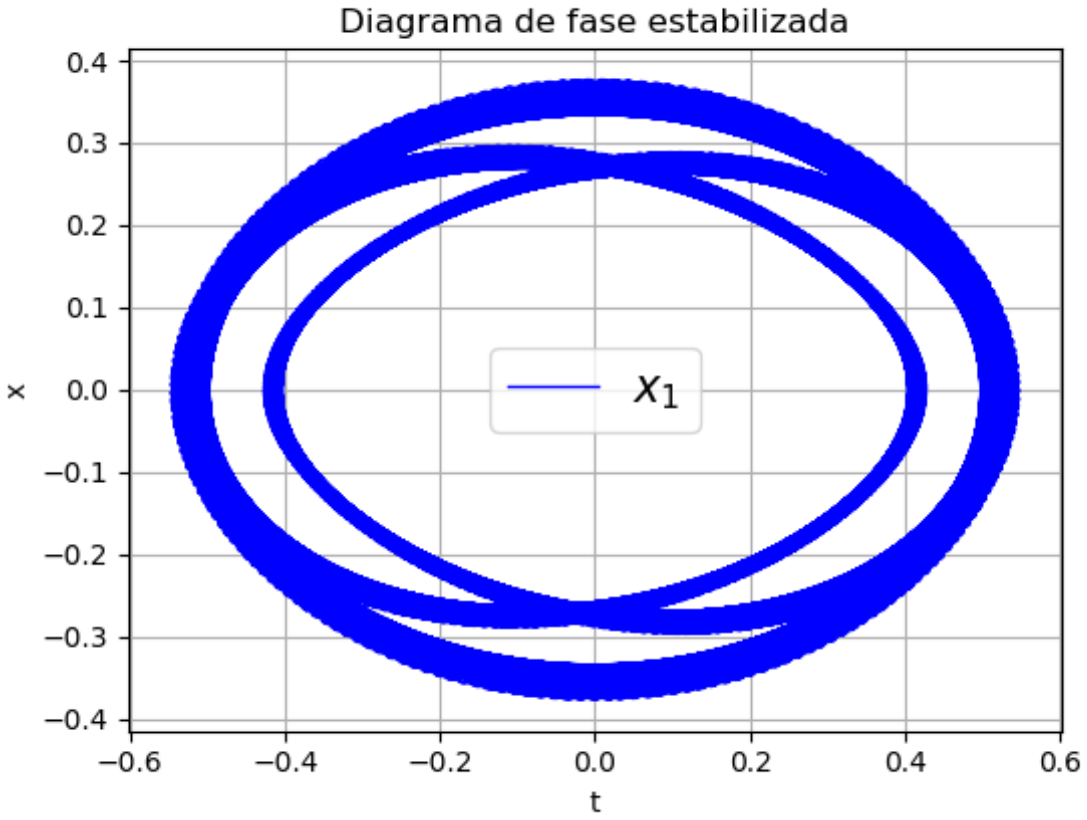
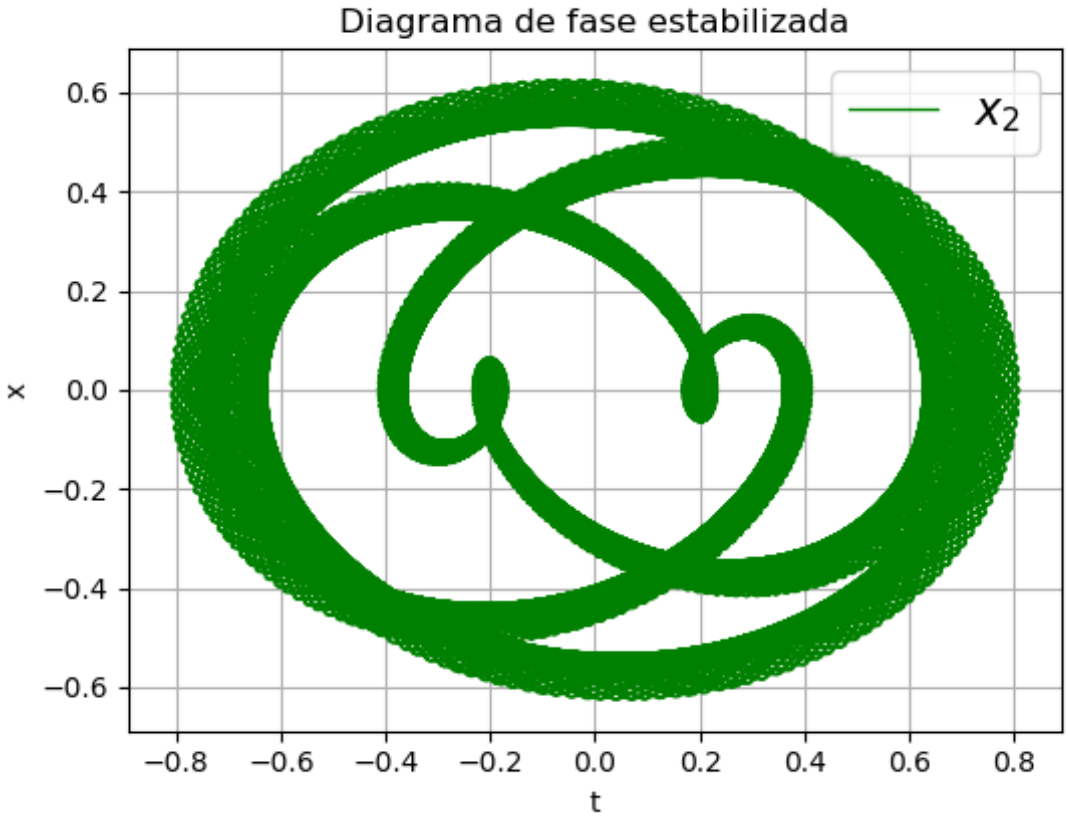


Diagrama de fase estabilizada
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
−0.3
−0.4
x
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
t
x_1



3. Conclusiones

Con esta actividad se aprendio a hacer un mejor uso de las herramientas para integracion numericas que ofrece python. Fue de gran ayuda porque permite tambien una visualizacion grafica de manera rapida.

4. Bibliografia

- Fay H., T., Duncan G. S., 2003., Coupled spring equations pp. 65-79. Int. J. Math. Educ. Sco. Technol.

5. Apéndice

1.¿Qué más te llama la atención de la actividad completa?
Que en el caso de fuerzas externas en el resorte debimos modificar el vector y ya no solo las condiciones iniciales.
2.¿De un sistema de masas acopladas como se trabaja en esta actividad, hubieras pensado que abre toda una nueva área de fenómenos no lineales?
No, no lo había pensado, sabía que habría un cambio, resultados nuevos, mas no una amplia gama de distintos resultados.
3.¿Qué propondrías para mejorar la actividad? ¿Te ha parecido interesante el reto?
Me parece bien, y no se me ocurre que se podría agregar.

4.¿Quisieras estudiar más éste tipo de fenómenos no lineales?
Si, pues considero que la mayoría de fenómenos son de éste tipo. No lineales.