

UNIVERSIDAD DE SONORA

LICENCIATURA EN FÍSICA

FÍSICA COMPUTACIONAL I

Actividad 7 - Sistema de resortes acoplados Completo

Alumno:

José Gabriel Navarro I.

Profesor:

Carlos Lizarraga Celaya

23 de Marzo de 2018



1 Introducción

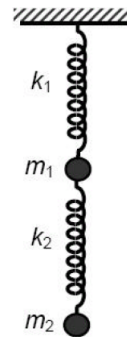
En el presente reporte se complementa la sexta actividad realizada para la clase de Física Computacional I, en donde además abarcar el análisis de las ecuaciones de un sistema de resortes acoplados, se habla también acerca de los sistemas no lineales y forzados.

Se presenta la síntesis realizada la práctica anterior, continuando con la nueva información acerca de sistemas no lineales y forzados del mismo artículo de Fay & Graham, mostrando sus ecuaciones y código para resolver estos problemas numéricamente. Al final se presenta una conclusión de la actividad, así como la bibliografía utilizada para la investigación de los fundamentos y el apéndice.

2 Síntesis - Sistema de resortes acoplados

2.1 Introducción

En este artículo, se investiga uno de los problemas más interesantes de Mecánica, y que ahora normalmente se utiliza para la introducción al estudio de ecuaciones diferenciales. Este problema es el de dos resortes con dos masas puestas en serie, colgando del techo. Si suponemos que las fuerzas restauradoras de los resortes se comportan según la Ley de Hooke, estos dos grados de libertad nos dan un modelo de ecuaciones diferenciales lineales de segundo grado. Al sustituir una ecuación en la otra, el movimiento de las masas puede ser descrito por una ecuación diferencial lineal de cuarto grado.



Con estas ecuaciones, podemos investigar los movimientos de las dos masas, para saber si estas están sincronizadas (es decir en fase), o si son opuestas (en desfase). Además, también se puede observar gráficamente la periodicidad, la amplitud, la fase, y otros conceptos al modificar los parámetros de este modelo.

2.2 El modelo de los resortes acoplados

Como se mencionó anteriormente, el modelo consiste de dos resortes y dos masas. Un resorte, con una constante k_1 , está colgado del techo con una masa m_1 colgando de ella. De aquí, cuelga otro resorte con una constante de k_2 y debajo de ella cuelga una masa m_2 . Al dejarlo en reposo, los resortes se estiran una distancia, a la que llamaremos: x_1 y x_2 .

2.2.1 Asumiendo la Ley de Hooke

Si asumimos que el sistema se mueve con oscilaciones pequeñas, podemos asumir que los resortes tendrán una fuerza restauradora dada por la Ley de Hooke, dada de la forma: $-k_1 l_1$ y $-k_2 l_2$ en donde l_1 y l_2 son las elongaciones o compresiones de los dos resortes. Como m_1 esta atada a los dos resortes, en esta actúan las dos fuerzas restauradoras, mientras que m_2 solamente "siente" la fuerza restauradora del segundo resorte. Sin fricción, la segunda ley de Newton para estas dos masas son de la siguiente forma:

$$\begin{aligned}m_1 \ddot{x}_1 &= -k_1 x_1 - k_2 (x_1 - x_2) \\ m_2 \ddot{x}_2 &= -k_2 (x_2 - x_1)\end{aligned}$$

Para encontrar una ecuación para x_1 que no involucre a x_2 , resolvermos la ecuación de x_2 , y sustituyendo esta ecuación en las ecuaciones anteriores llegamos a la siguiente ecuación diferencial de cuarto grado:

$$m_1 m_2 x_1^{(4)} + (m_2 k_1 + k_2 (m_1 + m_2)) \ddot{x}_1 + k_1 k_2 x_1 = 0$$

Y si se hace ese mismo proceso pero para x_2 , se llega a la misma ecuación anterior. Solamente las posiciones y velocidades iniciales se necesitan para poder determinar la solución.

2.2.2 Algunos ejemplos con masas idénticas

2.1 Describe el movimiento para un sistema de resortes con $k_1 = 6$ y $k_2 = 4$ con condiciones iniciales de $(x_1(0), \dot{x}_1(0), x_2(0), \dot{x}_2(0)) = (1, 0, 2, 0)$.

Resolviendo el problema de una forma analítica, podemos llegar a que las ecuaciones para las posiciones de las masas son:

$$\begin{aligned}x_1(t) &= \cos \sqrt{2}t \\ x_2(t) &= 2 \cos \sqrt{2}t\end{aligned}$$

El movimiento esta sincronizado, y por lo tanto las masas se mueven en fase una con la otra. Solamente tienen amplitud diferentes. El retrato de fase de estas son elipses, y como solamente varían en amplitud, al graficar sus posiciones una contra otra, se obtiene una línea recta. A continuación se presenta el código utilizado para encontrar numéricamente estos resultados y las graficas correspondientes:

Para graficar los datos, solamente se utilizo la línea de código de lectura de datos: `t, x1, xy, x2, y2, er1, er2 = loadtxt('dosresortes2_1.dat', unpack=True)`, y se graficaron los datos como ya se ha echo anteriormente en otras practicas.

```

def vectorfield(w, t, p):
    """
    Definimos las ecuaciones diferenciales para el sistema de doble masa-resorte.
    Arguments:
        w : Vector del estado de las variables
            w = [x1,y1,x2,y2]
        t : Tiempo
        p : Vector de los parametros:
            p = [m1,m2,k1,k2,L1,L2,b1,b2]
    """
    x1, y1, x2, y2 = w
    m1, m2, k1, k2, L1, L2, b1, b2 = p

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

#Usamos la funcion ODEINT para resolver las ecuaciones diferenciales definidas por el vector
from scipy.integrate import odeint
import numpy as np

#Valor de los parametros
# Masas:
m1 = 1.0
m2 = 1.0
# Constante del resorte
k1 = 6.0
k2 = 4.0
# Longitudes naturales
L1 = 0
L2 = 0
# Coeficientes de fricción
b1 = 0.0
b2 = 0.0

# Condiciones iniciales
# x1 and x2 son las pocisiones iniciales(contando la longitud de L), y y1 y y2 son las velocidades
x1 = 1.0
y1 = 0.0
x2 = 2.0
y2 = 0.0

# Parametros de la ED
abserr = 1.0e-8
relerr = 1.0e-6
stoptime = 50.0
numpoints = 250

#Creamos los valores del tiempo, entre mas puntos damos, mejores se veran las graficas.
t = [stoptime * float(i) / (numpoints - 1) for i in range(numpoints)]

# Ponemos a las variables en un vector.
p = [m1, m2, k1, k2, L1, L2, b1, b2]
w0 = [x1, y1, x2, y2]

# Llamamos a la funcion para resolver la ED
wsol = odeint(vectorfield, w0, t, args=(p,),
              atol=abserr, rtol=relerr)

with open('dosresortes2_1.dat', 'w') as f:
    # Imprimimos en el documento la solución
    for t1, w1 in zip(t, wsol):
        print (t1, w1[0], w1[1], w1[2], w1[3], np.abs((w1[0] - (np.cos(np.sqrt(2.0)*t1))) / (np.cos(np.sqrt(2.0)

```

```

#Graficamos la solución

import numpy
from numpy import loadtxt
from pylab import figure, plot, xlabel, grid, hold, legend, title, savefig, ylabel
from matplotlib.font_manager import FontProperties
import matplotlib.pyplot as plt
%matplotlib inline

t, x1, xy, x2, y2, er1, er2 = loadtxt('dosresortes2_1.dat', unpack=True)

figure(1, figsize=(6, 4.5))

xlabel('t')
ylabel('x')
grid(True)
lw = 1.5

plot(t, x1, 'navy', linewidth=lw)
plot(t, x2, 'indianred', linewidth=lw)
plt.axhline(0, color='black', linewidth=0.5)
plt.axvline(0, color='black', linewidth=0.5)

legend((r'$x_1$', r'$x_2$'), prop=FontProperties(size=16))
title('Posición de las Masas para el Sistema Doble Resorte-Masa')
savefig('Ej2_11.png', dpi=100)

```

Las gráficas obtenidas son:

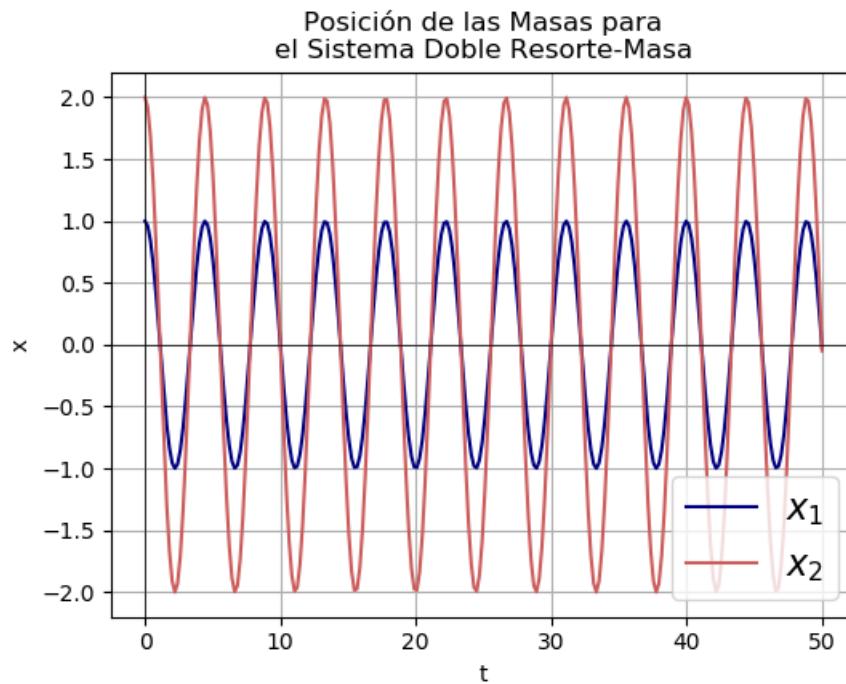
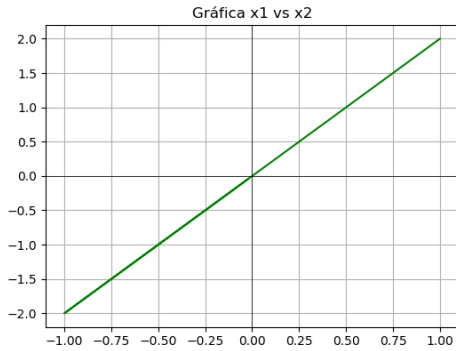
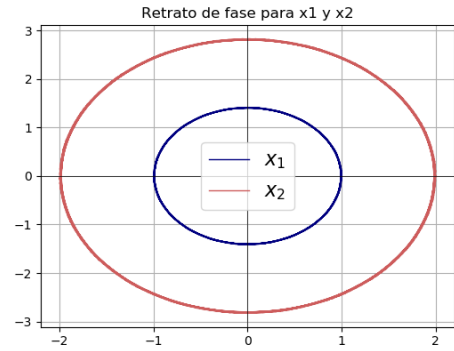


Figure 1: Gráficas de x_1 y x_2



(a) x_1 vs x_2



(b) Retrato de fase de x_1 y x_2

2.2 Describe el movimiento para un sistema de resortes con $k_1 = 6$ y $k_2 = 4$ con condiciones iniciales de $(x_1(0), \dot{x}_1(0), x_2(0), \dot{x}_2(0)) = (-2, 0, 1, 0)$.

Para este caso, la primera masa se mueve hacia abajo mientras que la otra se mueve hacia arriba, tienen el mismo periodo, pero están fuera de fase. La solución analítica es:

$$\begin{aligned} x_1(t) &= -2\cos 2\sqrt{3}t \\ x_2(t) &= \cos 2\sqrt{3}t \end{aligned}$$

El código utilizado para este caso, es el mismo que se usó en el ejemplo anterior, y así fue con todos los ejemplos. Lo único que cambiaba eran los datos a utilizar y las gráficas resultantes:

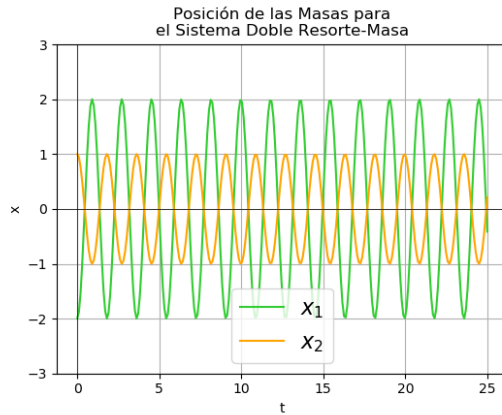
```
from scipy.integrate import odeint
import numpy as np

# Masas:
m1 = 1.0
m2 = 1.0
# Constante del resorte
k1 = 6.0
k2 = 4.0
# Longitudes naturales
L1 = 0
L2 = 0
# Coeficientes de fricción
b1 = 0.0
b2 = 0.0

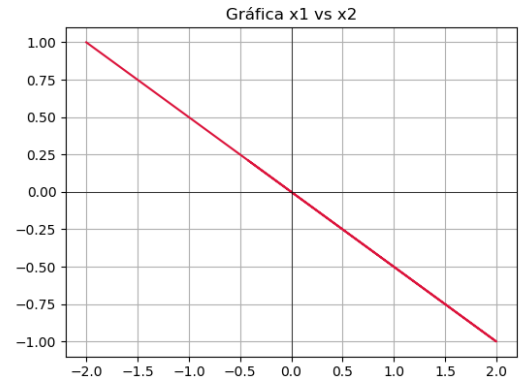
# Condiciones iniciales
x1 = -2.0
y1 = 0.0
x2 = 1.0
y2 = 0.0

# Parametros de la ED
abserr = 1.0e-8
relerr = 1.0e-6
stoptime = 25.0
numpoints = 250
```

Las gráficas son:



(a) Gráfica de x_1 y x_2



(b) Gráfica de x_1 vs x_2

2.3 Describe el movimiento para un sistema de resortes con $k_1 = 0.4$ y $k_2 = 1.808$ con condiciones iniciales de $(x_1(0), \dot{x}_1(0), x_2(0), \dot{x}_2(0)) = (1/2, 0, -1/2, 7/10)$.

Con este ejemplo podemos observar como las condiciones iniciales solo afectan a la amplitud y fase de la soluciones, mientras que las constantes de los resortes determinan el periodo y frecuencia del fenómeno. Los retratos de fase son casi un movimiento periódico entre ellos, y si graficamos x_1 vs x_2 parece una curva de tipo Lissajous. En el código solamente se edito los datos como las constantes y las condiciones iniciales:

```
m1 = 1.0
m2 = 1.0
# Constante del resorte
k1 = 0.4
k2 = 1.808
# Longitudes naturales
L1 = 0
L2 = 0
# Coeficientes de fricción
b1 = 0.0
b2 = 0.0

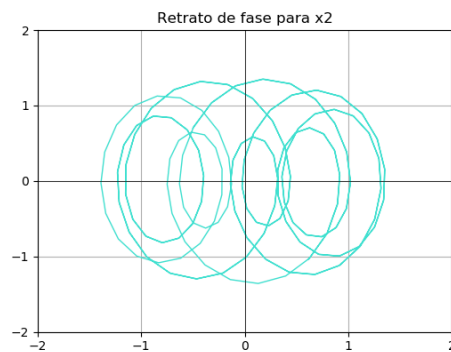
# Condiciones iniciales
x1 = 0.5
y1 = 0.0
x2 = -0.5
y2 = 0.7

# Parametros de la ED
abserr = 1.0e-8
relerr = 1.0e-6
stoptime = 50.0
numpoints = 250
```

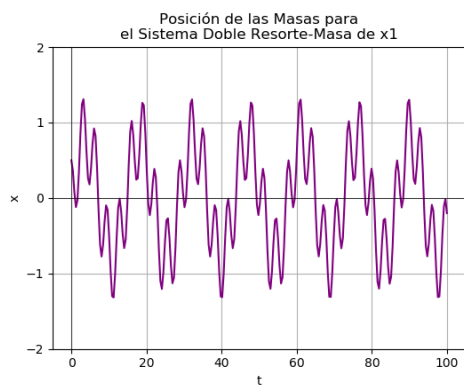
Las graficas resultantes fueron:



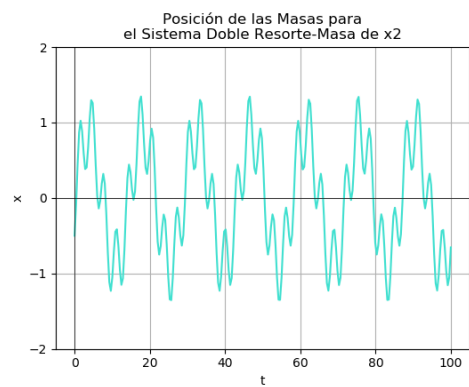
(a) Retrato de fase de x_1



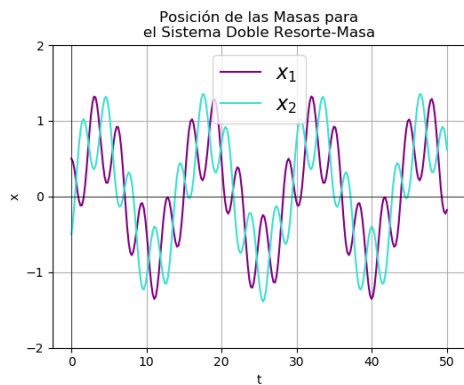
(b) Retrato de fase de x_2



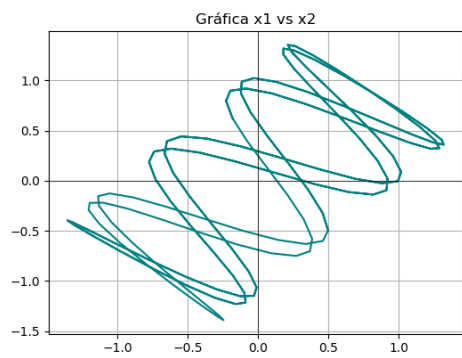
(c) Posición x_1 contra tiempo



(d) Posición x_2 contra tiempo



(e) Gráfica de x_1 y x_2



(f) Gráfica x_1 vs x_2

2.3 Amortiguamiento

Los problemas mas generales acerca de amortiguamiento es el de viscosidad, en donde la fuerza de amortiguamiento es proporcional a la velocidad. El amortiguamiento de la primera masa depende solamente de su velocidad y no en la de la segunda y viceversa. Si añadimos esta fuerza al modelo anteriormente descrito podemos obtener lo siguiente:

$$\begin{aligned}m_1\ddot{x}_1 &= -\delta\dot{x}_1 - k_1x_1 - k_2(x_1 - x_2) \\m_2\ddot{x}_2 &= -\delta\dot{x}_2 - k_2(x_2 - x_1)\end{aligned}$$

Se realiza el mismo proceso de obtener la ecuación de movimiento para una de las variables de posición, ya sea x_1 o x_2 . Sustituimos esta ecuación en la anterior que le corresponde, y obtenemos de nuevo una ecuación diferencial lineal de cuarto grado para ambas posiciones de x .

2.4 Asume $m_1 = m_2 = 1$. Describe el movimiento para un sistema de resortes con $k_1 = 0.4$ y $k_2 = 1.808$, con coeficientes de amortiguamiento $\delta_1 = 0.1$ y $\delta_2 = 0.2$ con condiciones iniciales de $(x_1(0), \dot{x}_1(0), x_2(0), \dot{x}_2(0)) = (1, 1/2, 2, 1/2)$.

Como existe ahora un amortiguamiento, las amplitudes de ambos movimientos disminuyen conforme pasa el tiempo. Al graficar x_1 y x_2 , podemos observar como se mueven casi en sincronía a pesar de que tienen distintas condiciones iniciales. El código para este caso es:

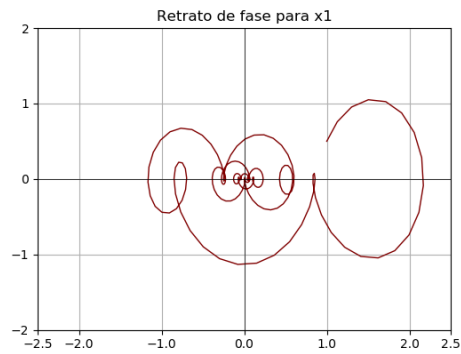
```
from scipy.integrate import odeint

# Masas:
m1 = 1.0
m2 = 1.0
# Constante del resorte
k1 = 0.4
k2 = 1.808
# Longitudes naturales
L1 = 0
L2 = 0
# Coeficientes de fricción
b1 = 0.1
b2 = 0.2

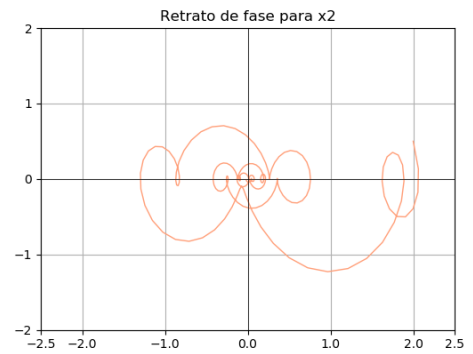
# Condiciones iniciales
x1 = 1.0
y1 = 0.5
x2 = 2.0
y2 = 0.5

# Parametros de la ED
abserr = 1.0e-8
relerr = 1.0e-6
stoptime = 50.0
numpoints = 250
```

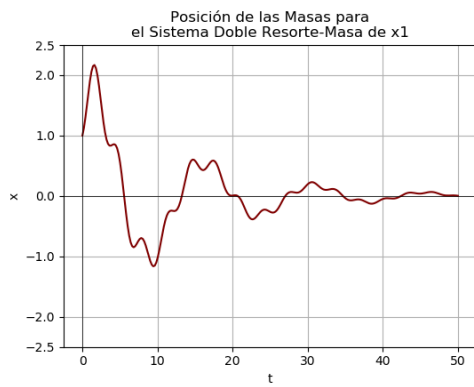
Las graficas resultantes fueron:



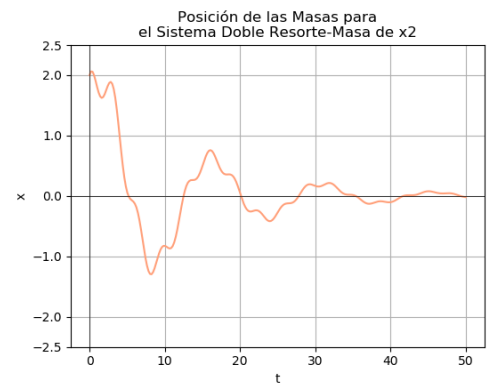
(a) Retrato de fase de x_1



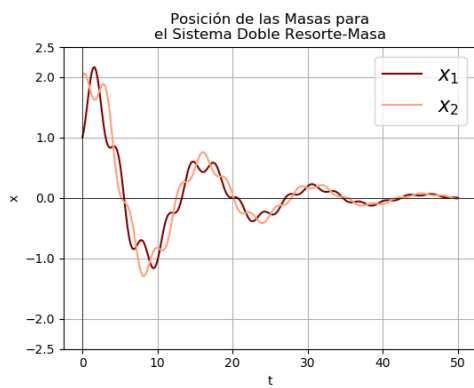
(b) Retrato de fase de x_2



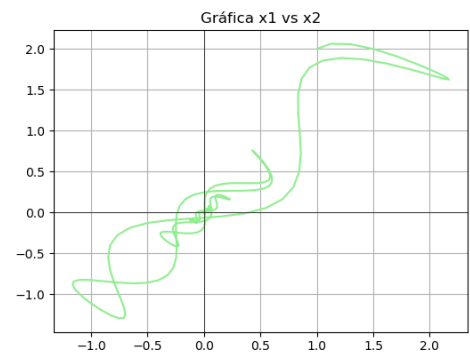
(c) Posición x_1 contra tiempo



(d) Posición x_2 contra tiempo



(e) Gráfica de x_1 y x_2



(f) Gráfica x_1 vs x_2

2.4 Añadiendo no linealidad

Para el caso en el que las fuerzas restauradoras de los resortes ya no obedezcan la Ley de Hooke, se debe modificar el modelo que se tenía. Ahora habrá una nueva constante, la cual será μ , en donde el modelo queda de la siguiente manera:

$$\begin{aligned}m_1\ddot{x}_1 &= -\delta\dot{x}_1 - k_1x_1 - k_2(x_1 - x_2) + \mu_1(x_1 - x_2)^3 \\m_2\ddot{x}_2 &= -\delta\dot{x}_2 - k_2(x_2 - x_1) + \mu_2(x_2 - x_1)^3\end{aligned}$$

La solución para estas ecuaciones son mas complicadas y se comportan de una manera mas extraña que los modelos lineales. Y aunque usemos un método numérico este perderá exactitud conforme pasa el tiempo. A continuación se presentan algunos ejemplos.

3.1 Asume $m_1 = m_2 = 1$. Describe el movimiento para un sistema de resortes con $k_1 = 0.4$ y $k_2 = 1.808$, con coeficientes de amortiguamiento $\delta_1 = 0$ y $\delta_2 = 0$, coeficientes de no lineales de $\mu_1 = -1/6$ y $\mu_2 = -1/10$ con condiciones iniciales de $(x_1(0), \dot{x}_1(0), x_2(0), \dot{x}_2(0)) = (1, 0, -1/2, 0)$.

Como no hay amortiguamiento las oscilaciones parecen ser casi periodicas, y al graficar x_1 con x_2 contra el tiempo, podemos observar como sus movimientos parecen estar fuera de fase. Podemos observar como debido a la no linealidad, el modelo es mas sensible a las condiciones iniciales.

```
x1, y1, x2, y2 = w
m1, m2, k1, k2, L1, L2, b1, b2, n1, n2 = p

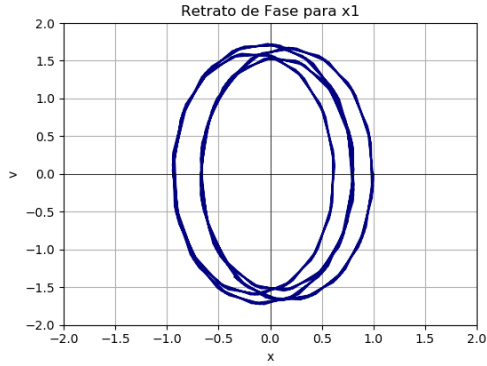
#Creamos f = (x1', y1', x2', y2')
f = [y1,
      (-b1 * y1 - k1 * (x1 - L1) + n1 * (x1 - L1)**3 + k2 * (x2 - x1 - L2) + n2 * (x1 - x2)**3) / m1,

      y2,
      (-b2 * y2 - k2 * (x2 - x1 - L2) + n2 * (x2 - x1)**3) / m2]
return f

m1 = 1.0
m2 = 1.0
# Constante del resorte
k1 = 0.4
k2 = 1.808
# Longitudes naturales
L1 = 0.0
L2 = 0.0
# Coeficientes de fricción
b1 = 0.0
b2 = 0.0
# Coeficiente de no linealidad
n1 = -(1.0/6.0)
n2 = -(1.0/10.0)

# Condiciones iniciales
# x1 and x2 son las posiciones iniciales(contando la longitud de L), y y1 y y2 son las velocidades
x1 = 1.0
y1 = 0.0
x2 = -(1.0/2.0)
y2 = 0.0
```

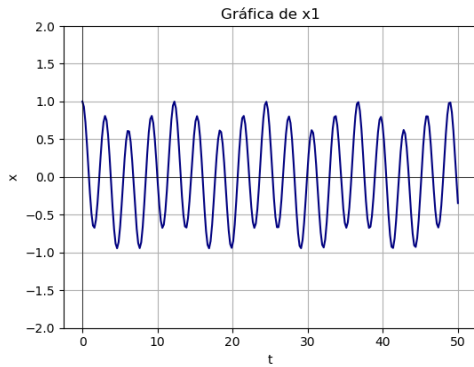
En el código solamente se edito la ecuación principal para añadir los nuevos coeficientes, dándoles los valores correspondientes. El resto se realizo de la misma manera que en los ejemplos anteriores. Las graficas resultantes fueron:



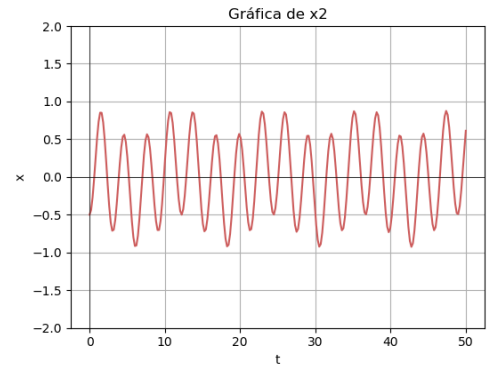
(a) Retrato de fase de x_1



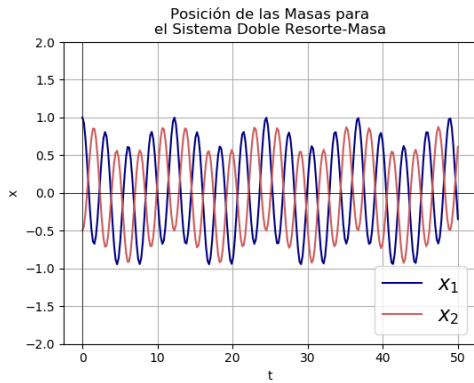
(b) Retrato de fase de x_2



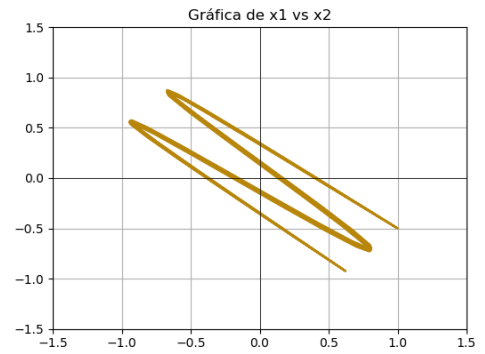
(c) Posición x_1 contra tiempo



(d) Posición x_2 contra tiempo



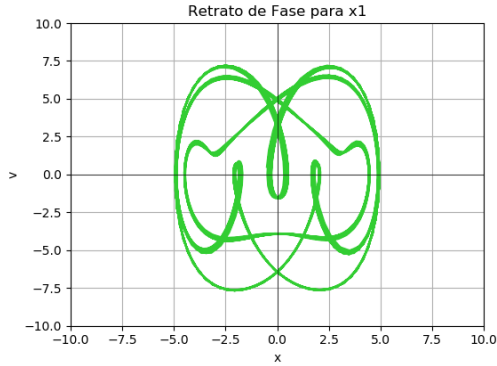
(e) Gráfica de x_1 y x_2



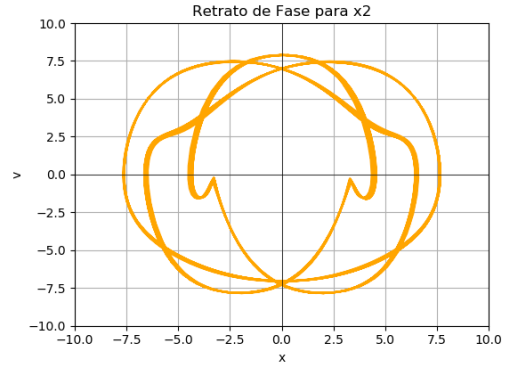
(f) Gráfica x_1 vs x_2

3.2 Asume $m_1 = m_2 = 1$. Describe el movimiento para un sistema de resortes con $k_1 = 0.4$ y $k_2 = 1.808$, con coeficientes de amortiguamiento $\delta_1 = 0$ y $\delta_2 = 0$, coeficientes de no lineales de $\mu_1 = -1/6$ y $\mu_2 = -1/10$ con condiciones iniciales de $(x_1(0), \dot{x}_1(0), x_2(0), \dot{x}_2(0)) = (-0.5, 1/2, 3.001, 5.9)$.

Este ejemplo tiene los mismos datos que el ejemplo anterior, solamente cambian las condiciones iniciales, en donde podemos observar, como un simple cambio hace que las graficas cambien bruscamente.



(a) Retrato de fase de x_1

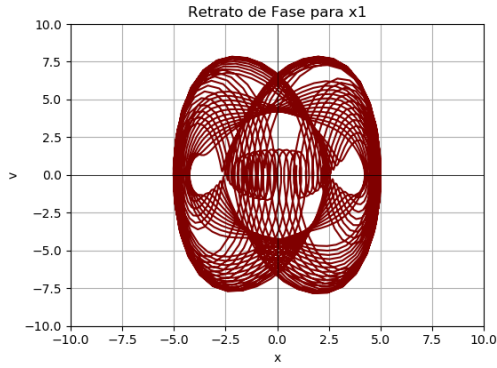


(b) Retrato de fase de x_2

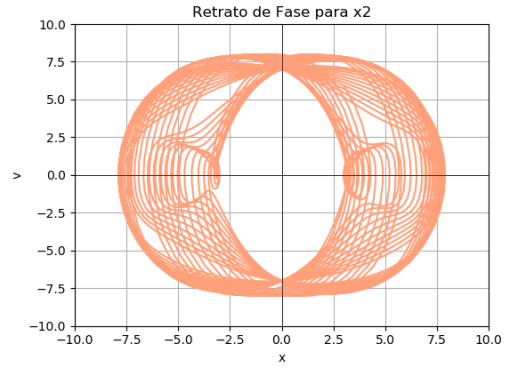


3.3 Asume $m_1 = m_2 = 1$. Describe el movimiento para un sistema de resortes con $k_1 = 0.4$ y $k_2 = 1.808$, con coeficientes de amortiguamiento $\delta_1 = 0$ y $\delta_2 = 0$, coeficientes de no lineales de $\mu_1 = -1/6$ y $\mu_2 = -1/10$ con condiciones iniciales de $(x_1(0), \dot{x}_1(0), x_2(0), \dot{x}_2(0)) = (-0.6, 1/2, 3.001, 5.9)$.

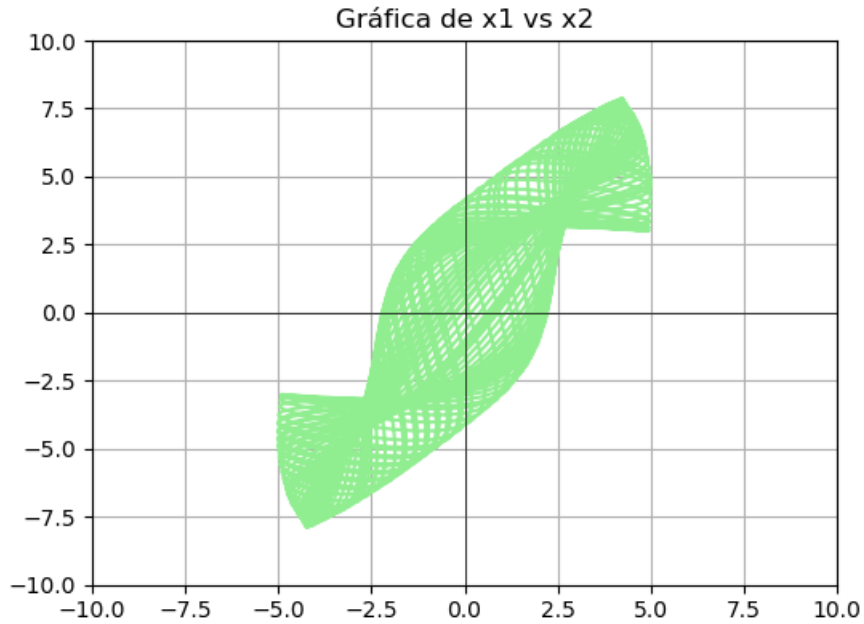
Este ejemplo tiene los mismos datos que el ejemplo anterior, pero ahora el unico cambio que existe es una diferencia de 0.1 en una de las posiciones iniciales, haciendo notar una vez mas que el mas simple cambio a las condiciones iniciales afectan mucho a los sistemas no lineales.



(a) Retrato de fase de x_1



(b) Retrato de fase de x_2



2.5 Añadiendo forzamiento

Es muy fácil añadirle fuerza externa al modelo pasado, en donde incluso puede añadirse una fuerza a cada masa simultáneamente, siendo o no iguales. Suponiendo una fuerza senoidal simple el modelo queda:

$$\begin{aligned}m_1\ddot{x}_1 &= -\delta\dot{x}_1 - k_1x_1 - k_2(x_1 - x_2) + \mu_1(x_1 - x_2)^3 + F_1\cos(w_1t) \\m_2\ddot{x}_2 &= -\delta\dot{x}_2 - k_2(x_2 - x_1) + \mu_2(x_2 - x_1)^3 + F_2\cos(w_2t)\end{aligned}$$

El rango de movimiento para este tipo de modelos tiene muchas posibilidades, resonancia no lineal, el periodo de solución comparte el mismo periodo con el del forzamiento (soluciones armónicas) e incluso soluciones subarmónicas. Las condiciones para que esto pase son muy difíciles de especificar, por lo cual para cerrar, se presenta un ejemplo final:

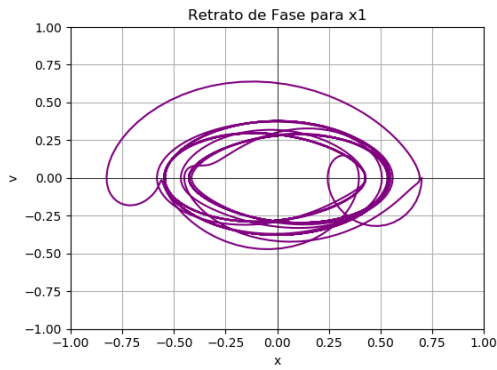
4.1 Asume $m_1 = m_2 = 1$. Describe el movimiento para un sistema de resortes con $k_1 = 2/5$ y $k_2 = 1$, con coeficientes de amortiguamiento $\delta_1 = 1/10$ y $\delta_2 = 1/5$, coeficientes de no lineales de $\mu_1 = 1/6$ y $\mu_2 = 1/10$, fuerzas de amplitud de $F_1 = 1/3$ y $F_2 = 3/5$ y frecuencias de amplitud de $w_1 = 1$ y $w_2 = 3/5$ con condiciones iniciales de $(x_1(0), \dot{x}_1(0), x_2(0), \dot{x}_2(0)) = (0.7, 0, 0.1, 0)$.

Como ahora si existe un coeficiente de amortiguamiento, para valores pequeños de t habrá un movimiento muy variable, pero para t grande se hará constante. Esto se puede observar en las graficas de limite de ciclo. El código solamente cambio de igual manera que la sección anterior, solamente se agrego los nuevos coeficientes y se le asignaron sus nuevos valores:

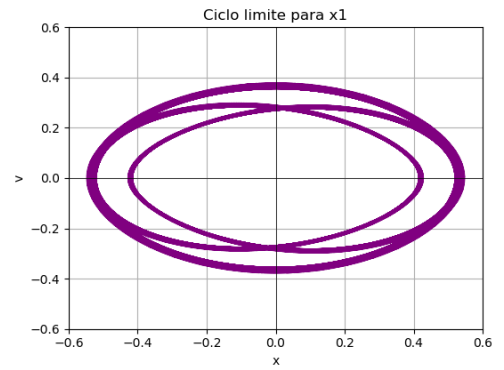
```
#Creamos f = (x1',y1',x2',y2')
f = [y1,
      (-b1 * y1 - k1 * (x1 - L1) + n1 * (x1 - L1)**3 + k2 * (x2 - x1 - L2) + n2 * (x1 - x2)**3 + F1
      * np.cos(w1*t)) / m1,
      y2,
      (-b2 * y2 - k2 * (x2 - x1 - L2) + n2 * (x2 - x1)**3 + F2 * np.cos(w2*t)) / m2]
return f
```

```
#Amplitud
F1 = (1.0/3.0)
F2 = (1.0/5.0)
#Frecuencia
w1 = 1
w2 = (3.0/5.0)
```

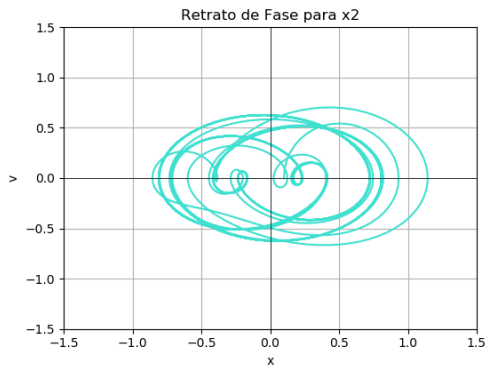
Las graficas obtenidas fueron:



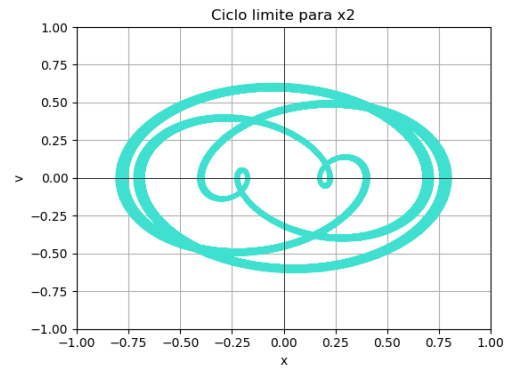
(a) Retrato de fase x_1



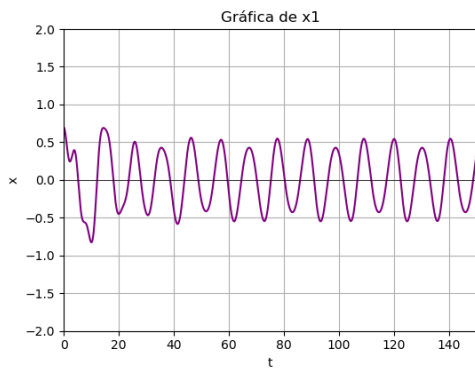
(b) Límite de ciclo para x_1



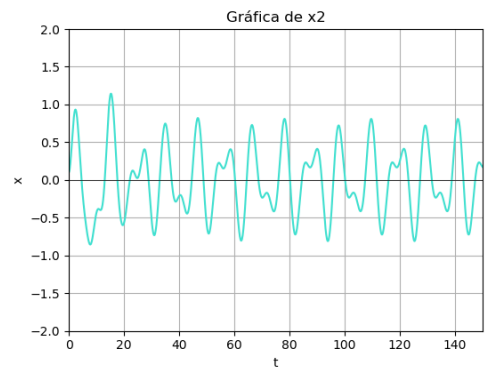
(c) Retrato de fase x_2



(d) Límite de ciclo para x_2



(e) Posición x_1 contra tiempo



(f) Posición x_1 contra tiempo

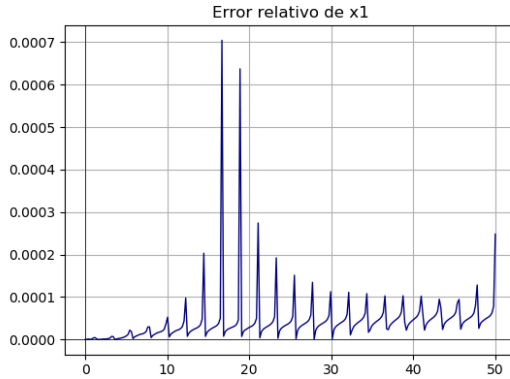
3 Resultado analítico y resultado numérico

Como en Python se realizó una aproximación numérica con los datos disponibles, y en el artículo se proporciona la solución analítica, es posible calcular el error relativo que hay entre nuestra aproximación y el valor real. Para hacer esto, en los primeros dos ejercicios, se imprimió en el archivo la definición del error relativo, restando el valor real al aproximado dividiendo esto entre el valor real.

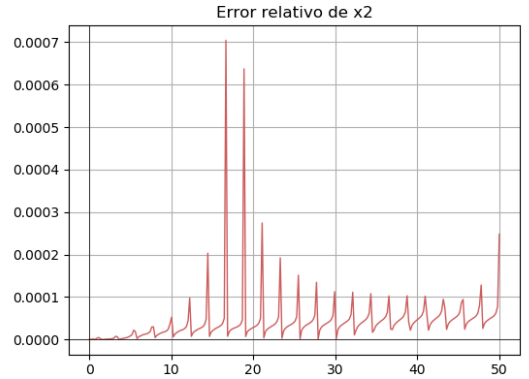
Para el primer caso se obtuvo el error relativo calculando:

```
np.abs((w1[0]-(np.cos(np.sqrt(2.0)*t1)))/(np.cos(np.sqrt(2.0)*t1))) \\  
np.abs((w1[2]-(2.0*np.cos(np.sqrt(2.0)*t1)))/(2.0*np.cos(np.sqrt(2)*t1)))
```

En donde el primer comando es para x_1 y el segundo para x_2 . Las graficas obtenidas del error relativo contra el tiempo, para el ejemplo 2.1 fueron:

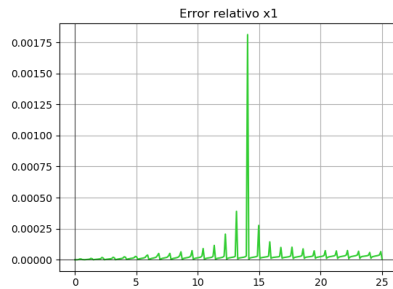


(a) Gráfica del error x_1 contra tiempo

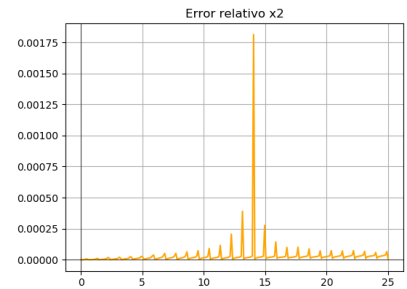


(b) Gráfica del error x_2 contra tiempo

Como se puede notar, los errores son muy pequeños, del orden de 7×10^{-3} , y los errores entre x_1 y x_2 son idénticos. Este es el caso también para el ejemplo 2.2, donde el error es muy pequeño y casi igual para x_1 y x_2 :



(a) Gráfica del error x_1 contra tiempo



(b) Gráfica del error x_2 contra tiempo

4 Conclusiones

Ahora que se realizaron todos los ejemplos presentados por el artículo, podemos observar no solo lo interesante que es este tema de los resortes acoplados, si no también del poder que tiene Python para poder resolver problemas tan complicados como estos.

El mismo artículo menciona como incluso programas numéricos no podrían tomar tan buenas aproximaciones de este tipo de modelos, pero Python logró reproducir todas las gráficas en su totalidad. Esto solo pone en prueba lo poderoso que es Python y todas sus librerías.

5 Bibliografía

- Temple H. Fay, Sarah Duncan Graham (2003) Coupled Spring Equations. Int. J. Educ. Math. Sci. Tech.. Vol. 34, No. 1, pp. 65-79. Recuperado de: http://math.oregonstate.edu/~gibsonn/Teaching/MTH323-010S15/Supplements/coupled_spring.pdf
- Oscilaciones amortiguadas. (2010) Recuperado de: <http://www.sc.ehu.es/sbweb/fisica/oscilaciones/amortiguadas/amortiguadas.htm>

Imágenes utilizadas:

- [http://srv2.fis.puc.cl/mediawiki/index.php/Osciladores_Acoplados_\(Fiz0312\)](http://srv2.fis.puc.cl/mediawiki/index.php/Osciladores_Acoplados_(Fiz0312))

6 Apéndice

1. ¿Qué más te llama la atención de la actividad completa? ¿Qué se te hizo menos interesante?

Me gustaron mucho más estos dos últimos temas que los pasados, ya que esto era algo totalmente nuevo. Este tema, como ya había mencionado anteriormente, es de mis favoritos de Mecánica II, y ver los modelos con sus respectivas soluciones fue muy divertido.

La verdad me gustó mucho esta práctica, no puedo decir que algo no me interese.

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?

La verdad no. Cuando veíamos los casos "bien comportados" en cursos anteriores, nunca pensé en los otros modelos que se podían presentar. Estos son mucho más complicados y el hecho de poder estudiarlos numéricamente con la ayuda de la programación es algo que me parece muy bien.

3. ¿Qué propondrías para mejorar esta actividad? ¿Te ha parecido interesante este reto?

Es muy interesante esta practica, y la verdad me pareció muy completa, sobre todo en la manera con la que se dividió. Me gustaría que hubiera mas referencias para poder entender mejor estos últimos dos temas, ya que el articulo casi no menciona mucho de ellos.

4. ¿Quisieras estudiar mas este tipo de fenómenos no lineales?

Si, es algo muy interesante.