Actividad 9

José Daniel Gaytán Villarreal
Grupo 3

8 de mayo de 2019

Resumen

Se planteó una ecuación diferencial de segundo orden para un sistema masa-resorte acoplado, la cual se resolvió a través de un código en el lenguaje de programación Python.

1. Introducción

El presente trabajo busca modelar un sistema masa-resorte acoplado, para simular así el movimiento de los cuerpos a través del lenguaje Python.

El objetivo de la actividad fue el resolver un sistema de ecuaciones diferenciales de segundo orden, manejándolo como un sistema de primer orden, a través de un código en el cookbook de SciPy.

En la sección 2, se habla brevemente sobre cómo se modificó la ecuación diferencial para introducirla en el código; a su vez, se mostrará la gráfica obtenida dadas ciertas condiciones iniciales al sistema. Por último, en la sección 3, se emite una breve conclusión referente a la metodología utilizada y sus subsecuentes resultados.

2. Desarrollo

2.1. Metodología

El sistema a modelar, obtenido de las notas del profesor Fitzpatrick[1], era el siguiente:

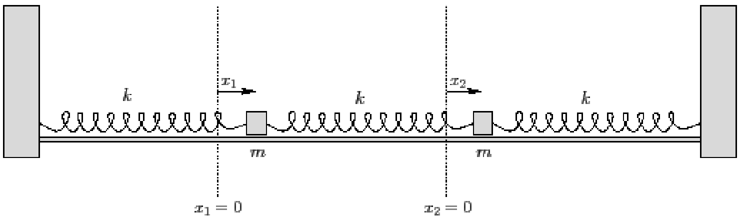

Figura 1: Sistema masa-resorte acoplado.

Para resolverlo, fue necesario identificar las fuerzas que actuán sobre cada cuerpo.

Sabemos, de la ley de Hooke, que, para un sistema masa-resorte, .[el] alargamiento unitario que experimenta un material elástico es directamente proporcional a la fuerza aplicada sobre el mismo F."Podemos escribir lo anterior como una ecuación de la siguiente manera:

$$F = -kx \tag{1}$$

o en su manera diferencial

$$\frac{d^2x}{dt^2} = -kx \tag{2}$$

Dado que nuestro código solo es capaz de resolver ecuaciones diferenciales de primer orden, fue necesario replantear nuestra ecuación a una equivalente de orden menor. Hacemos un cambio de variable

$$y = \frac{dx}{dt}$$

y reescribimos así nuestra ecuación 2

$$\frac{dy}{dt} = -kx \tag{3}$$

Para el primer cuerpo, de nuestra ecuación 1, vemos que las fuerzas que actúan sobre él son:

$$F_1 = -k_1(x_1 - L_1) - k_2(x_1 - x_2 + L_2 - L_1)$$

La segunda ley de Newton nos dice que

$$F = ma$$

Reescribiendo:

$$m_1 a_1 = -k_1(x_1 - L_1) - k_2(x_1 - x_2 + L_2 - L_1)$$

Tomando en cuenta la fricción sobre el sistema y diviendo sobre la masa:

$$a_1 = (-b_1 v_1 - k_1(x_1 - L_1) - k_2(x_1 - x_2 + L_2 - L_1))/m_1$$

En su forma diferencial:

$$\frac{d^2 x_1}{dt^2} = (-b_1 \frac{dx_1}{dt} - k_1(x_1 - L_1) - k_2(x_1 - x_2 + L_2 - L_1))/m_1$$

Recordando nuestro cambio de variable por la ecuación 3:

$$\frac{dy_1}{dt} = (-b_1 y_1 - k_1(x_1 - L_1) - k_2(x_1 - x_2 + L_2 - L_1))/m_1 \quad (4)$$

Realizando un proceso similar para el segundo cuerpo, obtenemos su ecuación diferencial:

$$\frac{dy_2}{dt} = (-b_2 y_2 - k_2(x_2 - L_2) - k_2(x_2 - x_1 + L_1 - L_2))/m_2 \quad (5)$$

Una vez obtenidas las ecuaciones, se procedió a adaptar un código preexistente del cookbook de SciPy[2], cuyos resultados pueden verse en la siguiente sección.

2.2. Resultados

Del modelo anterior se obtuvo la siguiente gráfica:

Figura 2: Parámetros: $m_1 = m_2 =$1.0, $k_1 = k_2 = k_3 =$1.0, $b_1 = b_2 =$0.0, $L_1 = L_2 =$1.0, C.I: $x_1 =$1.0, $x_2 = y_1 = y_2 =$0.0.

3. Conclusión

De la gráfica podemos visualizar cómo es que oscilarian las dos masas: una oscilación desfasada pero al mismo tiempo simétrica. Visualizar el movimiento es clave para poder comprender un concepto físico como este.

Hemos visto la eficacia del lenguaje Python para resolver ecuaciones diferenciales. Es necesario, pues, seguir estudiado los diferentes mètodos que este lenguaje nos presta para el modelaje de sistemas físicos, un punto clave en el día a día de un físico.

Referencias

[1] Fitzpatrick, R. (s.f) *Two Spring-Coupled Masses*. Universidad de Texas. Recuperado el 1 de mayo de 2017 de: https://farside.ph.utexas.edu/teaching/315/Waves/node18.html

[2] Coupled spring-mass system. (2017) *SciPy Cookbook*. Recuperado el 1 de mayo de 2017 de: https://scipy-cookbook.readthedocs.io/items/CoupledSpringMassSystem.html