

# Jupyter notebook como auxiliar en la enseñanza de ecuaciones diferenciales con experimentos

## Jupyter notebook as an aid in teaching differential equations with experiments

Luis G. Trujillo-Franco <sup>a,\*</sup>, Julio A. Guízar-García <sup>a</sup>, Jaime González-Sierra <sup>a</sup>, Josué D. Rivera-Fernández <sup>a</sup>,  
Hugo F. Abundis-Fong <sup>b</sup>, Erik López-Vázquez <sup>c</sup>

<sup>a</sup>Unidad Profesional Interdisciplinaria de Ingeniería Campus Hidalgo, Instituto Politécnico Nacional, 42162, San Agustín Tlaxiaca, Hidalgo, México.

<sup>b</sup>División de Estudios de Posgrado e Investigación, Tecnológico Nacional de México/I.T. Pachuca, 42080, Pachuca, Hidalgo, México.

<sup>c</sup>Área de ingeniería mecánica automotriz, Universidad Politécnica de Pachuca, 43830, Zempoala, Hidalgo, México.

### Resumen

Las herramientas didácticas y la tecnología de la información actuales, al ser aplicadas a la enseñanza de las ciencias, permiten demostrar y divulgar de manera atractiva y cautivante las aplicaciones de las ecuaciones diferenciales por medio de experimentos básicos que motivan a los estudiantes a profundizar en el tema al encontrar un sentido práctico a las formalidades que se les transmiten en los salones de clase y que es común que perciban como un obstáculo que casi nunca contribuye a mejorar su desempeño profesional. Las ecuaciones diferenciales ordinarias son herramientas fundamentales para las diversas ramas de las ciencias y la ingeniería, dado que son ampliamente utilizadas en la descripción formal de los fenómenos y procesos involucrados en el desarrollo tecnológico. Dada su importancia, en los programas de estudio de educación superior en ingeniería, son de suma importancia los cursos dedicados al tema de las ecuaciones diferenciales, donde se abordan los métodos analíticos y numéricos utilizados para su solución. En este trabajo se proponen dos ejemplos de aplicación de las ecuaciones diferenciales al estudio de sistemas físicos, utilizando una tarjeta Arduino Due y el ambiente de programación JupyterLab.

**Palabras Clave:** Modelado de sistemas, Solución numérica de ecuaciones diferenciales, Herramientas computacionales para la enseñanza.

### Abstract

The teaching tools and current information technology, when applied to science teaching, allow the applications of differential equations to be demonstrated and disclosed in an attractive and captivating way through basic experiments that motivate students to delve deeper into the subject by finding a practical meaning to the formalities that are transmitted to them in the classrooms and that they commonly perceive as an obstacle that seldom contributes to improving their professional performance. Ordinary differential equations are fundamental tools for the various branches of science and engineering since they are widely used in the formal description of the phenomena and processes involved in technological development. Given their importance, in higher education study programs in engineering, courses dedicated to differential equations are of utmost importance, where the analytical and numerical methods used for their solution are addressed. In this work, two examples of the application of differential equations to the study of physical systems are proposed, using an Arduino Due card and the JupyterLab programming environment.

**Keywords:** System modeling, Numerical solution of differential equations, Computational tools for teaching.

## 1. Introducción

En la enseñanza de la ingeniería en disciplinas como mecánica, eléctrica, electrónica, mecatrónica entre otras, existe

la necesidad de ofrecer a los estudiantes cursos de formación en ciencias básicas que incluyan al menos uno de ecuaciones diferenciales. En este contexto, han sido muchos los esfuerzos a nivel internacional por crear soluciones creativas e innovado-

\*Autor para correspondencia: ltrujillof@ipn.mx

**Correo electrónico:** ltrujillof@ipn.mx (Luis Gerardo Trujillo-Franco), jguizarg2100@alumno.ipn.mx (Julio Alexis Guízar-García), jagonzalezsi@ipn.mx (Jaime González-Sierra), jdrieveraf@ipn.mx (Josué Daniel Rivera-Fernández), hugo.af@pachuca.tecnm.mx (Hugo Francisco Abundis-Fong), erick\_lopez@upp.edu.mx (Erik López-Vázquez).

**Historial del manuscrito:** recibido el 27/06/2024, última versión-revisada recibida el 26/08/2024, aceptado el 22/08/2024, publicado el 30/11/2024. **DOI:** <https://doi.org/10.29057/icbi.v12iEspecial4.13281>



ras que logren atrapar la atención de los estudiantes, para que den el seguimiento necesario a sus cursos teóricos en el ámbito de las matemáticas y su utilidad con el proceso de solución de problemas y desarrollo de productos, aspectos en los que está involucrado el modelado matemático de sistemas dinámicos y de control. La incorporación de interfaces web no es un tema nuevo en el ámbito educativo en el área de ingeniería. En trabajos como (Castilla y Peña, 2023; Zúñiga López y Avilés-Cruz, 2020; Panagiotopoulos y Manolis, 2016; Laky *et al.*, 2023) se presentan aplicaciones basadas en navegador web como complemento y refuerzo a cursos teóricos como análisis de señales, mecánica de fluidos y procesos en la industria farmacéutica.

Los últimos avances en el ámbito de la programación orientada a objetos han permitido la proliferación de recursos para la difusión de código en diversos lenguajes de programación en forma de foros en los que muchos usuarios comparten valiosas piezas de código que coadyuvan a la integración de soluciones para diversos proyectos, ya sea para los entusiastas de la programación y la electrónica, como para los docentes que buscan captar la atención de sus estudiantes, involucrándose en el ambiente en el que suelen consultar información y se ha vuelto una parte fundamental de su técnica de estudio; el navegador de Internet. El ambiente Jupyter Notebook permite la creación de contenido interactivo, usando un navegador web, para compartir fundamentos teóricos de la más estricta formalidad que se complementa con código que permite desarrollar prácticas con resultados numéricos para su validación e incluso incluir experimentos con un nivel de detalle que antes era inimaginable, incorporando poderosas herramientas computacionales como el software MATLAB™ (Kamen, 2007).

El presente trabajo tiene por objetivo presentar los avances de un esfuerzo hecho por los autores para mejorar la experiencia de aprendizaje en el tema de ecuaciones diferenciales, utilizando un cuadernillo de prácticas en un formato innovador que involucra un par de experimentos clásicos de modelado, contenidos en un texto de narrativa sencilla y el uso del hardware abierto Arduino para la generación de una tarjeta de adquisición de datos de bajo costo. Las principales aportaciones del presente trabajo son las siguientes:

1. La evaluación de un cuadernillo de prácticas del método de Runge-Kutta para la solución numérica de ecuaciones diferenciales con experimentos, basado en navegador web.
2. La aplicación de un sistema de adquisición de datos de arquitectura abierta, basado en Arduino, con una precisión de 12 bits y una frecuencia de muestreo máxima de 100 kHz.
3. La validación numérica y experimental del modelo matemático de dos sistemas dinámicos, clásicos en la literatura en un contexto didáctico e interactivo.
4. El uso del lenguaje Python como alternativa al software de matemáticas de alto desempeño y alto costo, reduciendo el consumo de recursos computacionales y costos de licencias.

El presente trabajo está organizado de la siguiente manera: En la Sección 2 se presentan los fundamentos teóricos del método numérico evaluado para la simulación de los dos sistemas que se proponen. En la Sección 3 se describen los aspectos

principales de los dos sistemas físicos propuestos como experimentos. En la Sección 4 se describe la integración del ambiente Jupyter con los experimentos, en la Sección 5 se presentan los resultados experimentales más relevantes y finalmente en la Sección 6 se presentan algunas conclusiones.

## 2. Solución numérica de ecuaciones diferenciales

Los métodos de Runge-Kutta se utilizan para encontrar la solución de ecuaciones diferenciales ordinarias (EDO), que en forma genérica se expresan como:

$$\frac{dy}{dx} = f(x, y), \quad (1)$$

donde  $x$  es la variable independiente. Los métodos de Runge-Kutta (RK) se expresan en forma generalizada usando la siguiente expresión (Chapra *et al.*, 2006):

$$y_{(i+1)} = y_i + \varphi(x_i, y_i, h)h, \quad (2)$$

donde  $h$  es el paso o incremento, relacionada en la parte experimental con el periodo de muestreo, por otra parte,  $\varphi(x_i, y_i, h)$  se denomina función de incremento, misma que puede interpretarse como una pendiente que es representativa en el intervalo de análisis, la función  $\varphi$  adquiere la forma:

$$\varphi = a_1k_1 + a_2k_2 + \dots + a_nk_n. \quad (3)$$

Los coeficientes  $a_i$  son constantes y los términos  $k_i$  se determinan como:

$$k_1 = f(x_i, y_i), \quad (4)$$

$$k_2 = f(x_i + p_1h, y_i + q_{11}k_1h), \quad (5)$$

$$k_3 = f(x_i + p_2h, y_i + q_{21}k_1h + q_{22}k_2h), \quad (6)$$

⋮

$$k_n = f(x_i + p_{n-1}h, y_i + q_{n-1,1}k_1h + q_{n-1,2}k_2h + \dots \quad (7)$$

$$+ q_{n-1,n}k_{n-1}h); \quad (8)$$

donde los términos  $p_i$  y  $q_i$  son constantes. Además, los términos  $k$  son relaciones de recurrencia. Es decir,  $k_i$  aparece en la ecuación  $k_{i+1}$ , la cual aparece en la ecuación  $k_{i+2}$ , etc. Esta característica vuelve eficientes a los métodos RK para ser aplicados utilizando una computadora, dado que la recurrencia permite un uso más eficiente de la memoria y los recursos de la computadora, al evitar el congelamiento del sistema operativo, dado que limita el uso excesivo de la memoria destinada a la ejecución del programa.

El más popular de los métodos RK es el de cuarto orden. Como en el caso de los procedimientos de segundo orden, hay un número infinito de versiones. La siguiente, es la forma comúnmente usada y, por lo tanto, se conoce como el método clásico RK de cuarto orden (Chapra *et al.*, 2006; Zill, 1997)

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)h, \quad (9)$$

con:

$$k_1 = f(x_i, y_i), \quad (10)$$

$$k_2 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1h\right), \quad (11)$$

$$k_3 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_2h\right), \quad (12)$$

$$k_4 = f(x_i + h, y_i + k_3h). \quad (13)$$

El método de Runge-Kutta se encuentra en las librerías básicas de software de matemáticas como MATLAB™ y Octave como un importante auxiliar en las simulaciones de sistemas dinámicos, en el formato de simulación gráfica por medio de diagramas de bloque y la simulación numérica por medio de la línea de comandos. En el presente trabajo se reporta la evaluación del método de Runge-Kutta de hasta cuarto orden en dos ejemplos prácticos, incluido en un cuadernillo de prácticas en formato de página web.

### 3. Dos ejemplos de aplicación.

En este trabajo, se propone el uso de mediciones de las variables significativas de dos sistemas físicos diferentes. Se expresan las ecuaciones diferenciales que describen a los sistemas propuestos como ejemplos y se aborda la solución numérica de los modelos planteados utilizando el método de Runge-Kutta. Se presenta un contraste de los resultados numéricos con los obtenidos con las mediciones físicas. En la presente propuesta, la tarjeta de adquisición de datos es una tarjeta Arduino Due, mientras que las plantas experimentales son un circuito RLC, en el que la variable de salida es el voltaje en el capacitor y una figura de madera anclada a un resorte para simular un sistema masa-resorte-amortiguador, en el que se mide la aceleración.

#### 3.1. Primer ejemplo: Circuito resistencia, bobina y capacitor en serie.

Un ejemplo clásico de aplicación consiste en el análisis de corriente y voltaje de un circuito que consta de una resistencia, un capacitor y un inductor en serie como se muestra en la Figura 1 donde se presenta el diagrama esquemático del circuito propuesto, en el que la variable a analizar es el voltaje  $v_c$ , medido en el capacitor C.

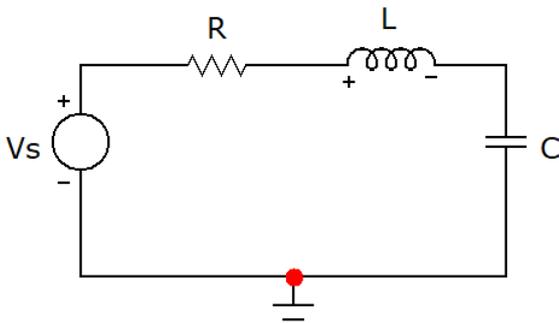


Figura 1: Circuito RLC serie propuesto; diagrama esquemático.

La planta experimental consiste en un circuito resistencia-inductancia-capacitancia (RLC), conectado a un sistema de adquisición de datos que consiste en una tarjeta Arduino Due, ver Figura 2, usando esta configuración se consigue una frecuencia de muestreo precisa de 5 kHz. Los datos se transfieren a una computadora por medio del protocolo VSP (puerto serie virtual), usando un puerto USB.

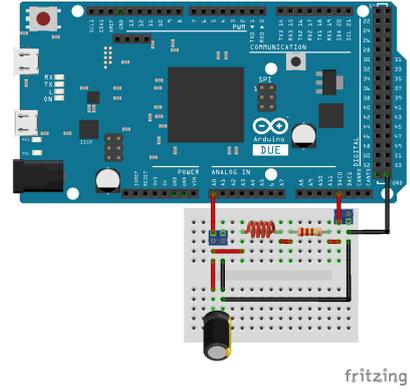


Figura 2: Plataforma experimental basada en Arduino.

La tarjeta Arduino Due cuenta con un convertidor digital-analógico con una resolución de 10 bits, el cual se utiliza para generar la señal de voltaje tipo escalón para el experimento propuesto. La parte numérica se realiza usando las librerías de matemáticas y tratamiento de arreglos y matrices incluidas en Python. Finalmente se comparan los resultados obtenidos por medio del método numérico con los datos experimentales. El modelo matemático que modela el comportamiento del circuito RLC serie es la siguiente ecuación integro-diferencial.

$$L \frac{di_L}{dt} + Ri_L + \frac{1}{C} \int i_L dt = V_s, \quad (14)$$

donde  $i_L$  es la corriente que circula a través del inductor L. Derivando (14) una vez respecto del tiempo, se cancela la integral y se obtiene:

$$L \frac{d^2 i_L}{dt^2} + R \frac{di_L}{dt} + \frac{1}{C} i_L = \frac{dV_s}{dt}. \quad (15)$$

Dado que se tiene una ecuación diferencial de segundo orden, es de gran utilidad recurrir a la representación en espacio de estados para reducir el orden de las ecuaciones diferenciales que modelan al sistema para obtener la representación del sistema lineal:

$$\dot{x}_1 = -\frac{R}{L}x_1 - \frac{1}{L}x_2 + \frac{1}{L}V_s, \quad (16)$$

$$\dot{x}_2 = \frac{1}{C}x_1; \quad (17)$$

con  $x_1 = i_L$ ,  $x_2 = \dot{v}_C$  y  $v_C = \frac{1}{C} \int i_L dt$  la corriente en el inductor y el voltaje en el capacitor respectivamente. La representación de ecuaciones diferenciales por medio del espacio de estados es especialmente útil cuando se trata con sistemas dinámicos a los que se les aplican leyes de control o esquemas de estimación de parámetros. Una forma para el método de Runge-Kutta de segundo orden, para el circuito RLC es:

$$i_{L_{i+1}} = i_{L_i} + h \left( \frac{1}{2}k_{11} + \frac{1}{2}k_{12} \right), \quad (18)$$

$$v_{C_{i+1}} = v_{C_i} + h \left( \frac{1}{2}k_{21} + \frac{1}{2}k_{22} \right); \quad (19)$$

donde:

$$k_{11} = -\frac{R}{L}i_{L_i} - \frac{1}{L}v_{C_i} + \frac{1}{L}V_{s_i}, \quad (20)$$

$$k_{12} = \frac{1}{C}v_{C_i}, \quad (21)$$

$$k_{21} = -\frac{R}{L}(i_{L_i} + hk_{11}) - \frac{1}{L}(v_{C_i} + hk_{12}) + \frac{1}{L}V_{s_i}, \quad (22)$$

$$k_{22} = \frac{1}{C}(v_{C_i} + hk_{11}). \quad (23)$$

La estructura para el cálculo de los coeficientes  $k_i$  para el método Runge-Kutta de cuarto orden es la siguiente:

$$k_{11} = -\frac{R}{L}i_{L_i} - \frac{1}{L}v_{C_i} + \frac{1}{L}V_{s_i}, \quad (24)$$

$$k_{12} = \frac{1}{C}i_{L_i}, \quad (25)$$

$$k_{21} = -\frac{R}{L}(i_{L_i} + 0.5h) - \frac{1}{L}(v_{C_i} + 0.5hk_{12}) + \frac{1}{L}V_{s_i}, \quad (26)$$

$$k_{22} = \frac{1}{C}(i_{L_i} + 0.5h), \quad (27)$$

$$k_{31} = -\frac{R}{L}(i_{L_i} + 0.5h) - \frac{1}{L}(v_{C_i} + 0.5hk_{22}h) + \frac{1}{L}V_{s_i}, \quad (28)$$

$$k_{32} = \frac{1}{C}(i_{L_i} + 0.5h), \quad (29)$$

$$k_{41} = -\frac{R}{L}(i_{L_i} + h) - \frac{1}{L}(v_{C_i} + hk_{32}) + \frac{1}{L}V_{s_i}, \quad (30)$$

$$k_{42} = \frac{1}{C}(i_{L_i} + h). \quad (31)$$

### 3.2. Segundo ejemplo: Sistema masa-resorte-amortiguador.

El segundo ejemplo experimental es un sistema masa-resorte-amortiguador, el cual se muestra en la Figura 3. Este ejemplo académico es un juguete de madera que tiene unido un resorte de acero con una constante de rigidez lo suficientemente pequeña para permitir un análisis visual de la frecuencia natural y el movimiento, al mismo tiempo, sus características reducen la exigencia a la tarjeta de adquisición de datos y al acelerómetro, al reducir considerablemente la frecuencia de muestreo requerida.

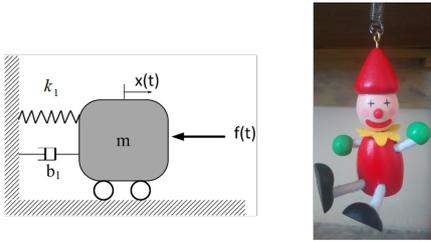


Figura 3: Sistema masa-resorte-amortiguador.

La ecuación diferencial ordinaria que modela al sistema vibratorio de naturaleza mecánica es la siguiente:

$$m \frac{d^2x}{dt^2} + b \frac{dx}{dt} + kx = 0, \quad (32)$$

$$accel = \frac{d^2x}{dt^2}; \quad (33)$$

con  $accel$  la aceleración de la masa y  $x(t)$  la posición. En forma compacta se puede escribir:

$$m\ddot{x} + b\dot{x} + kx = 0, \quad (34)$$

$$accel = \ddot{x}; \quad (35)$$

con  $\dot{\}$  el operador derivada  $\frac{dx}{dt} = \dot{x}$

Como en el caso anterior, se recurre a la representación en espacio de estados de (33) para obtener:

$$\dot{x}_1 = x_2, \quad (36)$$

$$\dot{x}_2 = -\frac{b}{m}x_2 - \frac{k}{m}x_1; \quad (37)$$

con  $x_1 = x(t)$  la posición y  $x_2 = \dot{x}_1$  la velocidad instantáneas de la masa o del juguete de madera que funge como planta experimental o ejemplo práctico. Una forma para el método de Runge-Kutta de segundo orden, para el sistema masa resorte amortiguador es:

$$x_{1,i+1} = x_{1,i} + h\left(\frac{1}{2}k_{11} + \frac{1}{2}k_{12}\right), \quad (38)$$

$$x_{2,i+1} = x_{2,i} + h\left(\frac{1}{2}k_{21} + \frac{1}{2}k_{22}\right); \quad (39)$$

con:

$$k_{12} = -\frac{b}{m}x_2 - \frac{k}{m}x_1, \quad (40)$$

$$k_{11} = x_2, \quad (41)$$

$$k_{22} = -\frac{b}{m}(x_2 + hk_{11}) - \frac{k}{m}(x_1 + hk_{12}), \quad (42)$$

$$k_{21} = x_2 + hk_{11}. \quad (43)$$

La conexión del sensor de aceleración y la tarjeta Arduino se muestra en la Figura 4. El sensor de aceleración se une al juguete de madera usando un tornillo y un imán de neodimio.

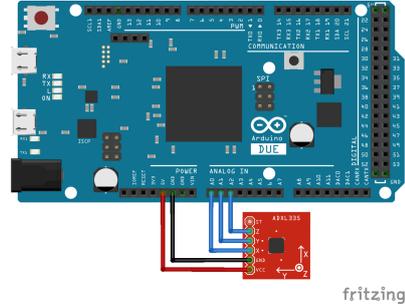


Figura 4: Conexiones físicas del acelerómetro.

## 4. Integración del Jupyter Notebook con los experimentos

La parte gráfica de la plataforma educativa se desarrolló usando el ambiente de desarrollo JupyterLab, que permite la combinación del lenguaje de creación de texto científico LaTeX con bloques de código interactivo usando el kernel de Python. Actualmente, existen opciones para la generación de código interactivo sin la necesidad de instalar compiladores o intérpretes de lenguajes como C++, C# y por supuesto Python. En la presente propuesta se utiliza el lenguaje de programación Phyton para la ejecución de las líneas de código desde el navegador, para su versión online. La portada del cuadernillo de Python se muestra en la Figura 5.

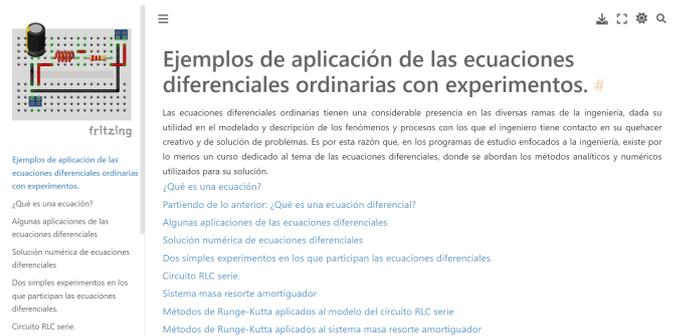


Figura 5: Portada del cuadernillo de prácticas.

El cuadernillo de prácticas con experimentos guiados se ofrece en dos formatos:

- Versión on-line: En esta versión, el cuadernillo de prácticas se encuentra en forma de una página web, contenida en un repositorio de GitHub, con la limitante de que los experimentos no se realizan en tiempo real, sino que se usa un archivo con los datos almacenados de una prueba previa para su descarga y visualización.

- Versión en red local: En esta versión, el cuadernillo de prácticas se ejecuta en la computadora del instructor, con el hardware conectado a un puerto USB. En este formato, los usuarios usan el navegador web de su preferencia para la visualización del cuadernillo y la ejecución en tiempo real de los experimentos. Este formato tiene la limitante de depender de las capacidades de la computadora que funge como servidor.

Para el caso de la ejecución del cuadernillo de prácticas en el formato Jupyter Notebook, se ejecutan las instrucciones descritas en el Código 1, que debe guardarse en un archivo dentro de la carpeta en la que se encuentra el notebook a compartir. Se omiten las instrucciones de graficación, pero se encuentran incluidas en el cuadernillo embebido en el ambiente JupyterLab.

Código 1: Ejecución de un Jupyter Notebook en una red local.

```
import os
import socket

hostname=socket.gethostname()
IPAddr=socket.gethostbyname(hostname)
print("El nombre de esta computadora es:" +hostname)
print("La dirección IP de esta computadora es:" +IPAddr)

os.system("jupyter notebook --port 8889 --allow-root --no-browser --ip="+IPAddr)
```

La presentación como caso de estudio del circuito RLC serie en el cuadernillo de prácticas se muestra en la Figura 6, donde se describen los fundamentos teóricos y las instrucciones para el desarrollo del experimento.



Figura 6: Presentación de fundamentos teóricos.

La comunicación entre la tarjeta de adquisición de datos, a la cual se conectan tanto el acelerómetro como el circuito RLC y el código contenido en el ambiente Jupyter se lleva a cabo mediante el uso de la librería de comunicación serial disponible en Python, llamada PySerial, la comunicación serie se realiza en configuración Full-Speed de 1.2 Mb/s por segundo, usando las especificaciones que se muestran en el Código 2.

Código 2: Código para la comunicación USB.

```
from __future__ import division, absolute_import, print_function
import serial # Necesaria para la comunicación con la tarjeta
import array # Necesaria para almacenar los datos
f_muestreo = 5000 # Frecuencia de muestreo

muestras = 2**10 # Número de muestras.
a = array.array('H') # Arreglo de datos.
# Inicialización del puerto serie
ser = serial.Serial(port='COM5', # Número de puerto asignado a la tarjeta
baudrate=2000000,
parity=serial.PARITY_NONE,
stopbits=serial.STOPBITS_ONE,
bytesize=serial.EIGHTBITS)
```

La comunicación serie establecida es efectiva para ambos casos de estudio, el circuito RLC y el sistema masa-resorte-amortiguador.

El método de RK de cuarto orden para el sistema mecánico, se presenta en un bloque de código en el lenguaje Python. En el formato de presentación del cuadernillo de prácticas en su modalidad de red local, la ejecución de las líneas de código del método numérico reportadas en el Código 3 es completamente interactiva, dado que es posible editar el valor numérico de los parámetros del sistema para fines comparativos y de validación. En la Figura 7 se muestra el aspecto de la página web en la que se ejecuta el código del método Runge-Kutta y la parte experimental.



Figura 7: Caso de estudio sistema mecánico.

Código 3: Método Runge-Kutta de cuarto orden, sistema mecánico.

```
p0=0.045;
v0=0.0; # Condiciones iniciales
A=0;
U0=A;

m = 43/1000;
b = 0.0115;
k = 4.3;

h=1/10000; # Ancho del paso
N=int(15/h); # Número total de pasos

n=0
for n in range(N):
    tr[n+1]=n*h;
    U[n]=0;
    U[n+1]=0

    # Método Runge-Kutta 4to orden
    k11=v4[n]
    k12=-b/m*v4[n]-k/m*p4[n]+1/m*U[n];

    k21=v4[n]+0.5*h;
    k22=-b/m*(v4[n]+0.5*h)-k/m*(p4[n]+0.5*h*k12)+1/m*U[n];

    k31=v4[n]+0.5*h;
    k32=-b/m*(v4[n]+0.5*h)-k/m*(p4[n]+0.5*h*k22)+1/m*U[n];

    k41=v4[n]+h;
    k42=-b/m*(v4[n]+h)-k/m*(p4[n]+h*k32)+1/m*U[n];

    p4[n+1]=p4[n]+(1/6)*h*(k11+2*k21+2*k31+k41);
    v4[n+1]=v4[n]+(1/6)*h*(k12+2*k22+2*k32+k42);
    acc4[n]=-(b/m)*v4[n]-(k/m)*p4[n]+1/m*U[n];
    n=n+1;
```

La estructura general de la plataforma educativa propuesta se muestra en la Figura 8, en donde se muestra el acceso al ambiente JupyterLab por medio del uso de un navegador web, sin importar el sistema operativo o la instalación del software Python en los dispositivos de los usuarios. Existen alternativas para la creación de entradas interactivas para la determinación de los parámetros teóricos por medio de controles digitales tipo deslizante y generación de gráficas animadas con el uso de la librería interact, desarrollada especialmente para el kernel de Python, la presente propuesta las incluye como una prueba piloto con la restricción de que funcionan únicamente en la opción de red local de la ejecución del cuadernillo de prácticas Jupyter.

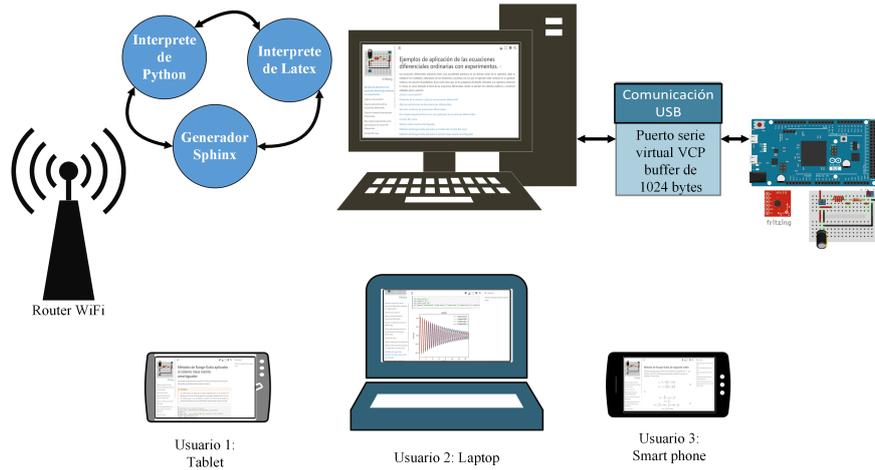


Figura 8: Diagrama a bloques de la plataforma didáctica.

En el Código 4 se muestra el algoritmo para la programación del método RK de segundo orden en el lenguaje Python, para el caso de estudio del circuito RLC.

Código 4: Método Runge-Kutta de cuarto orden.

```
#Método Runge-Kutta de segundo orden
i0=0;
v0=0; # Condiciones iniciales

R = 1570;
L = 15.02; #Valores de resistencia R, capacitancia C e inductancia L
C = 10.0e-6;
h=1/2000; # Ancho del paso
N=int(0.4/h); # Número total de pasos

n=0
for n in range(N):
    tr[n+1]=n+h;
    U[n]=1.638;
    U[n+1]=1.638;
    # Cálculo de los coeficientes.
    k11=-R/L*i2[n]-1/L*v2[n]+1/L*U[n];
    k12=1/C*i2[n];

    k21=-R/L*(i2[n]+h*k11)-1/L*(v2[n]+h*k12)+1/L*U[n];
    k22=1/C*(i2[n]+h*k11);

    i2[n+1]=i2[n]+h*(0.5*k11+0.5*k21);
    v2[n+1]=v2[n]+h*(0.5*k12+0.5*k22);

    n=n+1;
```

de 10 bits. Las mediciones del voltaje en el capacitor se sincronizan a una tasa fija de 5000 muestras por segundo, usando las capacidades de acceso directo a memoria (DMA) del microcontrolador de 32 bits incluido en la tarjeta Arduino. El uso del esquema DMA permite la transmisión al puerto USB de la computadora que ejecuta JupyterLab. Los parámetros del circuito RLC serie se reportan en la Tabla 1. La transmisión de las muestras se realiza en un esquema de post-procesamiento, es decir, no se muestran gráficas en tiempo real para mejorar el desempeño de la tarjeta y el cuadernillo de prácticas digital.

Tabla 1: Parámetros del circuito RLC serie.

| Parámetro | Valor             |
|-----------|-------------------|
| $V_s$     | 1.63 [V]          |
| $R$       | 1600 [ $\Omega$ ] |
| $L$       | 15 [H]            |
| $C$       | 1 [ $\mu F$ ]     |

## 5. Resultados experimentales

Los casos de estudio descritos en la Sección 3 ofrecen la oportunidad de validar dos modelos matemáticos diferentes, usando una misma tarjeta de adquisición de datos y dos sensores diferentes. Se realiza la medición directa del voltaje en el capacitor del circuito RLC descrito en la Figura 1 y la aceleración del juguete de madera mostrado en la Figura 3, por medio de un acelerómetro con salida analógica. En esta sección se muestran los resultados más relevantes del uso del cuadernillo de prácticas para la validación experimental de los modelos matemáticos (15) y (33).

### 5.1. Circuito RLC

La respuesta experimental del circuito RLC se presenta en esta sección, junto con la comparación de los resultados de la solución numérica del modelo matemático del sistema con el voltaje medido en el capacitor. Para la validación del modelo matemático del circuito se usa la respuesta al escalón. La excitación del circuito se genera utilizando el convertidor digital-analógico de la tarjeta Arduino Due, la cual tiene una resolución

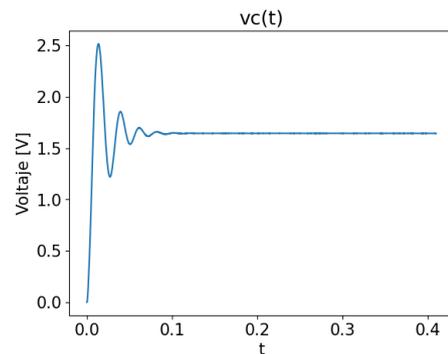


Figura 9: Resultados experimentales, circuito RLC.

El valor alto de inductancia reportado en la Tabla 1 se consigue utilizando el devanado primario de un transformador comercial de núcleo de hierro dulce. La resistencia es el único parámetro variable y consiste en un potenciómetro de 5 k $\Omega$  a medio Watt de potencia, la variación de la resistencia permite demostrar las respuestas sub-amortiguadas de los sistemas dinámicos, con la posibilidad de mostrar de forma práctica este

tema adicional. Los valores seleccionados de los parámetros de capacitancia y resistencia permiten que el circuito tenga una frecuencia natural menor a 50 Hz, lo que significa que la dinámica del circuito RLC es lo suficientemente lenta para ser adquirida con precisión por el convertidor analógico-digital de la tarjeta Arduino Due. En la Figura 9 se muestra la respuesta al escalón del circuito RLC. El desempeño del convertidor digital-analógico, en sincronía con el convertidor analógico-digital es muy aceptable para el experimento propuesto.

Los resultados numéricos se comparan con los datos experimentales, de manera gráfica en la Figura 10, en donde se observa un muy buen desempeño del algoritmo del método de Runge-Kutta programado con base en las ecuaciones (19). Es posible notar una coincidencia considerable entre los datos experimentales y los resultados de la solución numérica del modelo para los métodos de Runge-Kutta de segundo a cuarto orden.

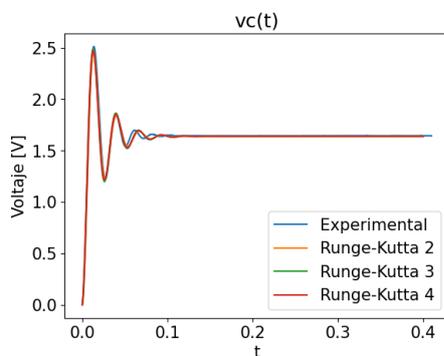


Figura 10: Comparación de resultados, circuito RLC.

## 5.2. Sistema masa-resorte-amortiguador.

La solución numérica del modelo matemático del juguete de madera mostrado en la Figura 3 y considerado un sistema masa-resorte-amortiguador de un solo grado de libertad, se presenta en esta sección. El modelo matemático a simular, usando el método de Runge-Kutta es el conjunto de ecuaciones en forma de espacio de estados definido en las ecuaciones (37). Los parámetros del sistema mecánico se reportan en la Tabla 2, donde el parámetro del amortiguador se considera constante y se obtiene de forma empírica, usando el método de prueba y error, dado que el amortiguamiento en sistema vibratorios es un tema abierto de investigación (Brandt, 2011).

Tabla 2: Parámetros del sistema masa-resorte-amortiguador.

| Parámetro | Valor         |
|-----------|---------------|
| $m$       | 0.043 [kg]    |
| $b$       | 0.0015 [Ns/m] |
| $k$       | 4.3 [N/m]     |

Los resultados de las mediciones de aceleración, en condiciones de vibraciones libres se muestran en la gráfica en el dominio del tiempo de la Figura 11, donde las condiciones iniciales para la prueba se establecieron en una posición inicial de 4.5 cm y una velocidad inicial de 0 m/s.

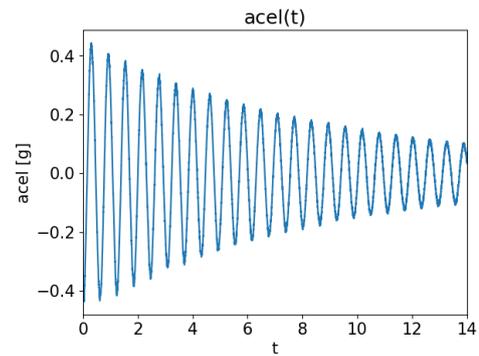


Figura 11: Resultados experimentales, sistema mecánico.

La comparación de las mediciones de aceleración y los resultados numéricos se muestra de forma gráfica en la Figura 12, donde se presenta la evolución en el dominio del tiempo de la dinámica del sistema de un solo grado de libertad.

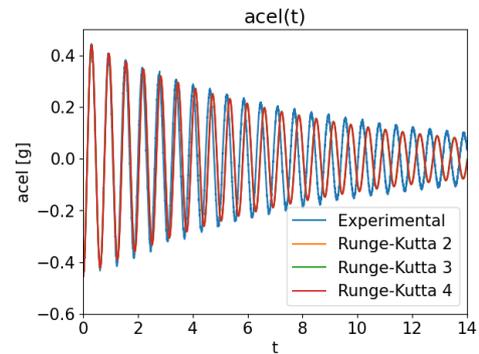


Figura 12: Comparación de resultados, sistema mecánico.

Se observa un correcto desempeño de los métodos de Runge-Kutta de segundo a cuarto orden en la aproximación de la aceleración. Las discrepancias que se presentan entre los datos numéricos y los experimentales se deben principalmente a dinámicas no modeladas y a los efectos no lineales, dado que el modelo aproxima al sistema a uno de naturaleza lineal e invariante en el tiempo.

Las plantas experimentales descritas en la Sección 3 fueron mostradas a un total de 27 estudiantes como materiales complementarios a dos asignaturas relacionadas con el modelado de sistemas dinámicos. Las plataformas experimentales permitieron la demostración de dos tópicos principales; adquisición de datos y validación experimental de modelos matemáticos. Algunos comentarios de los estudiantes respecto de los experimentos son:

- “Fue una práctica bastante interesante que me ayudó a comprender de una mejor manera el tema.”
- “El experimento realizado resultó provechoso para el análisis del modelo dinámico del circuito presentado, además de generar un interés por su realización y las probables aplicaciones que este experimento puede tener a largo plazo en diferentes unidades de aprendizaje, además de ser una demostración para nosotros los estudiantes de como se podrían poner nuestro conocimientos ya adquiridos a la práctica.”

- “Son prácticas que motivan e impulsan a seguir estudiando y comprender mejor la teoría y los ejercicios de clase, además de que vamos conociendo más componentes de los circuitos.”
- “Es interesante como los comportamientos dinámicos de los circuitos se pueden modelar matemáticamente.”

En cuanto a las propuestas de mejora, áreas de oportunidad de la plataforma didáctica conformada por el cuadernillo de prácticas en el formato JupyterLab y las plantas experimentales, algunos comentarios y sugerencias importantes son:

- “Sería de gran utilidad contar con más aparatos, para la manipulación individual de cada alumno o por lo menos de cada equipo.”
- “Tener una mejor explicación sobre el funcionamiento del código y las placas.”
- “Mi sugerencia sería poner un potenciómetro de mayor capacidad para que podamos observar los 3 casos de amortiguamiento en los circuitos de segundo orden.”
- “Me gustaría que hubiera habido más prácticas como esa, para poder tener distintos datos y obtener distintos valores de resistencias por equipos.”

En la Figura 13 se muestra el resultado de la evaluación de la plataforma, en una escala de 0 a 5 de los aspectos más importantes de los experimentos y el uso del cuadernillo en forma general. Se observa una buena aceptación del método de enseñanza experimental aplicado y al mismo tiempo se evidencia que las evaluaciones más bajas se obtienen en los aspectos relacionados con el lenguaje de programación.

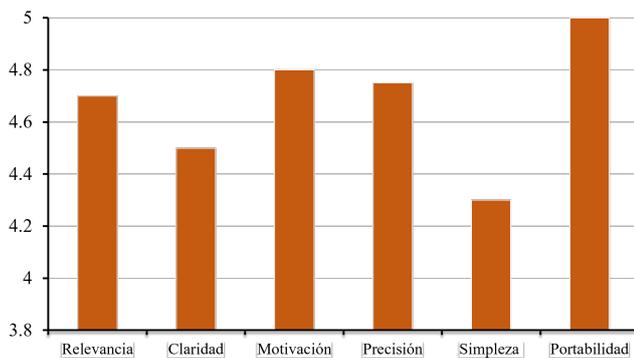


Figura 13: Evaluación de la propuesta por parte de los estudiantes.

Como trabajo futuro se incluirá una evaluación interactiva elaborada con herramientas de código abierto como Jupyter-Quiz, kahoot! o alguna plataforma similar para elaboración de

cuestionarios. El objetivo principal de la evaluación será la determinación del nivel de aprovechamiento del material didáctico de forma individual por parte de los estudiantes, dado que la propuesta presentada en este trabajo se centra en la puesta en marcha de los experimentos y está diseñada para su presentación de manera colectiva. Se pretende construir una cantidad suficiente de tarjetas de adquisición para que pueda realizarse la práctica en equipos de máximo cuatro integrantes. De manera adicional, se pretende incluir un contexto de análisis de sistemas en el dominio de la frecuencia y aprovechar las capacidades de la tarjeta de adquisición de datos para analizar sistemas mecánicos con un comportamiento dinámico diferente, como las vigas empotradas lo cual coadyuvará también a complementar con validaciones experimentales a los cursos de elemento finito en el tema de análisis modal de estructuras.

## 6. Conclusiones

El uso del ambiente de generación de contenido didáctico JupyterLab permite introducir a los cursos teóricos, un componente práctico muy importante y enriquecedor que coadyuva de manera importante a la comprensión de las ecuaciones diferenciales como herramientas de modelado de sistemas dinámicos. El uso de experimentos sencillos aporta una motivación a los estudiantes dado el análisis de información sobre fenómenos tangibles en forma de datos obtenidos de sensores. La validación experimental de los modelos matemáticos es altamente portátil gracias a las herramientas de hardware y software libres que permiten la integración de sistemas de adquisición de datos a los programas de simulación numérica con una precisión muy aceptable, que es adecuada para el aula destinada a la enseñanza de la ingeniería.

## Referencias

- Brandt, A. (2011). *Dynamic Signals and Systems*, capítulo 2, pp. 7–33. John Wiley & Sons, Ltd.
- Castilla, R. y Peña, M. (2023). Jupyter notebooks for the study of advanced topics in fluid mechanics. *Computer Applications in Engineering Education*, pp. 1–13.
- Chapra, S., Canale, R., Brito, J., y Hano, M. (2006). *Métodos numéricos para ingenieros*. Elibro Catedra. McGraw-Hill.
- Kamen, E. W. (2007). *Fundamentals of Signals and Systems: Using the Web and MATLAB*, third edition. Prentice Hall.
- Laky, D. J., Casas-Orozco, D., Abdi, M., Feng, X., Wood, E., Reklaitis, G. V., y Nagy, Z. K. (2023). Using pharmacy with jupyter notebook to teach digital design in pharmaceutical manufacturing. *Computer Applications in Engineering Education*, 31(6):1662–1677.
- Panagiotopoulos, C. G. y Manolis, G. D. (2016). A web-based educational software for structural dynamics. *Computer Applications in Engineering Education*, 24(4):599–614.
- Zill, D. G. (1997). *Ecuaciones Diferenciales con Aplicaciones*. Grupo Editorial Iberoamérica, México, tercera edición.
- Zúñiga López, A. y Avilés-Cruz, C. (2020). Digital signal processing course on jupyter python notebook for electronics undergraduates. *Computer Applications in Engineering Education*, 28(5):1045–1057.