Taller 2
Respuesta de sistema a señales arbitrarias - MATLAB

Jaime Patricio Chiqui Chiqui
Ingenieria en Telecomunicaciones
Universidad de Cuenca
Cuenca, Ecuador
jpatricio.chiquic@ucuenca.edu.ec

***Abstract*—This report presents the analysis of the dynamic response of a second-order system with delay to various excitation signals using MATLAB. Through the implementation of transfer functions, the system's response was evaluated under impulse, step, and arbitrary input signals. Additionally, a mixed PRBS-type signal combining sinusoidal, ramp, and step components was generated to study the transient and steady-state behavior. The use of MATLAB functions such as `tf`, `impulse`, `step`, and `lsim` allowed for a deeper understanding of how a linear time-invariant (LTI) system reacts to diverse inputs. The obtained results highlight the importance of signal type and delay in shaping the system's temporal response.**

***Index Terms*—Transfer function, MATLAB, LTI systems, arbitrary signals, PRBS, impulse response, step response.**

I. Introducción

El análisis de la respuesta de un sistema ante diferentes señales de entrada es uno de los pilares fundamentales en el estudio de la teoría de control. Este taller se centra en la simulación y comprensión del comportamiento dinámico de sistemas lineales invariantes en el tiempo (LTI) mediante herramientas computacionales en MATLAB.

A través de este trabajo, se estudió la respuesta de un sistema de segundo orden con retardo ante señales de tipo impulso, escalón y señales arbitrarias. Asimismo, se implementó una señal combinada tipo PRBS (Pseudo Random Binary Signal), la cual permite observar la respuesta del sistema ante excitaciones más complejas.

El uso de funciones como `tf`, `impulse`, `step` y `lsim` en MATLAB proporciona un entorno ideal para modelar, analizar y visualizar la respuesta temporal de sistemas, facilitando la interpretación de parámetros dinámicos como la ganancia, el retardo y la estabilidad [1].

El objetivo principal es comprender cómo el tipo de señal de entrada influye directamente en la forma de la respuesta del sistema y cómo los parámetros dinámicos, como la ganancia y el retardo, modifican su comportamiento.

II. Marco teórico

A. Función de transferencia

La función de transferencia es una herramienta matemática que describe la relación entre la entrada y la salida de un sistema LTI en el dominio de Laplace. Su definición general se expresa como:

$$G(s) = \frac{Y(s)}{X(s)} \tag{1}$$

donde $Y(s)$ representa la transformada de Laplace de la salida y $X(s)$ la de la entrada, bajo condiciones iniciales nulas.

Para este taller, el sistema de estudio corresponde a un sistema de segundo orden con retardo puro, definido por la ecuación:

$$G(s) = \frac{3}{s^2+2s+3}e^{-2s} \tag{2}$$

El término exponencial e^{-2s} representa un tiempo muerto o retardo de 2 s, el cual provoca un desplazamiento temporal en la respuesta del sistema. Este tipo de sistemas son comunes en procesos térmicos, eléctricos e industriales donde la acción de control no se refleja de forma instantánea.

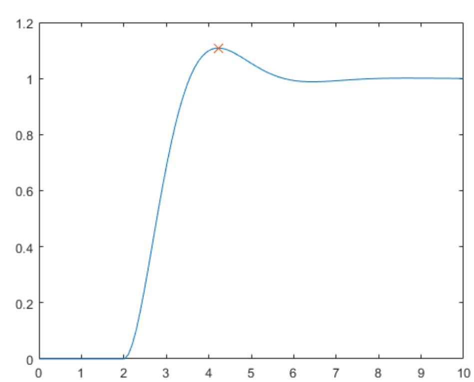

Fig. 1: Modelo de un sistema de segundo orden con retardo.

En la Figura 1 se muestra el bloque funcional del sistema de segundo orden con retardo. El término e^{-2s} genera un desplazamiento temporal sin alterar la amplitud ni la forma de la respuesta dinámica.

B. Tipos de señales de entrada

Para analizar un sistema, se utilizan señales de prueba estándar, tales como el impulso, el escalón, la rampa y la senoidal. Cada una permite observar un aspecto específico del comportamiento dinámico.

La Tabla I resume las principales señales y sus transformadas de Laplace.

TABLE I: Principales señales de entrada y su transformada de Laplace.

Señal	Expresión	Transformada de Laplace
Impulso	$\delta(t)$	1
Escalón unitario	$u(t) = 1$	$\frac{1}{s}$
Senoidal	$A\sin(\omega t)$	$\frac{A\omega}{s^2+\omega^2}$
Rampa	$r(t) = t$	$\frac{1}{s^2}$
Exponencial	$e^{-\alpha t}$	$\frac{1}{s+\alpha}$
Arbitrarias	Combinaciones o formas no estándar, útiles para representar condiciones reales.	Depende de la señal.

Estas señales permiten evaluar la estabilidad, el régimen permanente y la respuesta en frecuencia de los sistemas de control [2].

C. Principales comandos de MATLAB

- **tf(num, den)** – Crea la función de transferencia de un sistema continuo en MATLAB a partir de los vectores del numerador y denominador.
- **tf(num, den, ts)** – Define una función de transferencia en tiempo discreto, donde `ts` es el tiempo de muestreo.
- **tf(num, den, 'InputDelay', delay)** – Agrega un retardo o tiempo muerto al sistema (representado como e^{-sT_d}).
- **impulse(sys)** – Grafica y calcula la respuesta al impulso de un sistema.
- **step(sys)** – Obtiene la respuesta al escalón unitario, mostrando la evolución temporal hasta el régimen permanente.
- **lsim(sys, u, t)** – Simula la respuesta del sistema ante una señal de entrada arbitraria `u(t)` en el intervalo de tiempo `t`.

D. Señales PRBS (Pseudo Random Binary Sequence)

Una señal PRBS (*Pseudo Random Binary Sequence*) es una secuencia binaria pseudoaleatoria utilizada para excitar un sistema en un amplio rango de frecuencias de forma controlada. Estas señales son ampliamente empleadas en experimentos de identificación de sistemas, ya que permiten evaluar la respuesta del modelo ante múltiples componentes frecuenciales sin requerir una excitación sinusoidal continua [4].

Su estructura se basa en una secuencia de bits generada por registros de desplazamiento lineal (*Linear Feedback Shift Registers*, LFSR), que producen una secuencia periódica determinista con propiedades estadísticas similares a una señal aleatoria. Aunque la señal alterna entre dos niveles fijos, por ejemplo $-A$ y $+A$, su distribución temporal parece aleatoria.

Matemáticamente, una PRBS puede expresarse como:

$$u(t) = A \cdot (-1)^{b(t)} \tag{3}$$

donde $b(t)$ representa la secuencia binaria pseudoaleatoria generada por el LFSR y A corresponde a la amplitud máxima del estímulo.

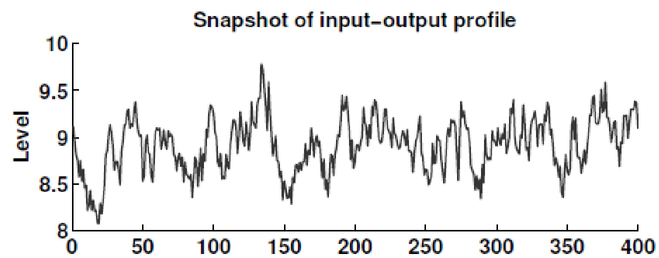

Fig. 2: Ejemplo conceptual de una señal PRBS utilizada como excitación de un sistema dinámico.

En la Figura 2 se aprecia que la señal PRBS alterna entre valores altos y bajos en instantes no periódicos, excitando el sistema con un amplio espectro de frecuencias. Esto permite estimar la dinámica completa del sistema de manera eficiente sin necesidad de múltiples pruebas con diferentes señales de entrada.

Para efectos de simulación, también pueden emplearse señales *PRBS extendidas*, en las que se combinan secciones senoidales, rampas y escalones. Estas variaciones permiten analizar el comportamiento no lineal o retardado de los sistemas bajo condiciones de excitación más realistas, como se muestra en la Sección 14.

III. Desarrollo

En esta sección se describe paso a paso el procedimiento realizado para analizar la respuesta de un sistema de segundo orden con retardo ante diferentes tipos de señales, tanto típicas como arbitrarias. Se emplearon las funciones `tf`, `impulse`, `step` y `lsim` de MATLAB, siguiendo las indicaciones de la guía [1].

A. Codifique en MATLAB la función de transferencia del ejemplo

De acuerdo con la ecuación (2.3) del taller, la función de transferencia está dada por:

$$G(s) = \frac{3}{s^2+2s+3} \tag{4}$$

El siguiente código define la función de transferencia y grafica las respuestas al impulso y al escalón:

```
num = [3];          % Numerador de la funcion de TF
den = [1 2 3];      % Denominador de la funcion de TF
Gs = tf(num, den);  % Tiempo continuo

impulse(Gs); grid on;
step(Gs); grid on;
```

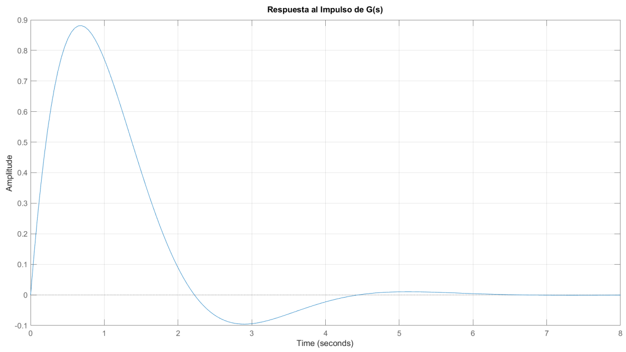

Fig. 3: Respuesta al impulso del sistema continuo $G(s)$.

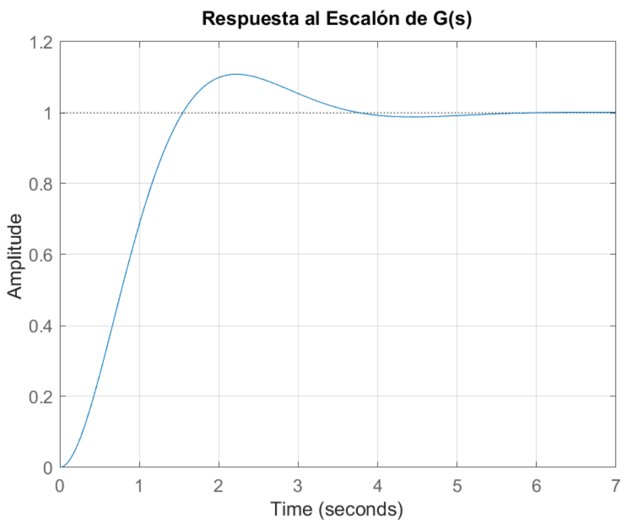

Fig. 4: Respuesta al escalón del sistema continuo $G(s)$.

Las Figuras 3 y 4 muestran que el sistema presenta una respuesta subamortiguada típica de un sistema de segundo orden con oscilaciones amortiguadas antes de alcanzar el régimen permanente.

B. Almacene los valores en vectores para facilitar el análisis

El objetivo de esta parte es guardar los valores de salida y tiempo de la función `step()` en vectores, para realizar análisis posteriores.

```
[y, t] = step(Gs);
figure();
plot(t, y)
```

Al almacenar los valores en vectores, se pueden calcular fácilmente parámetros como el valor máximo, el tiempo de establecimiento y la sobreoscilación.

C. 3. Respuesta del sistema con retardo

El sistema con retardo se representa como:

$$G(s) = \frac{3}{s^2+2s+3}e^{-2s} \tag{5}$$

```
delay = 2;
Gs_delay = tf(num, den, 'InputDelay', delay);

[y1, t1] = impulse(Gs_delay);
[y2, t2] = step(Gs_delay);

figure;
plot(t1, y1, 'b', 'LineWidth', 1.5);
title('Respuesta al Impulso de G(s) con Retardo');
xlabel('Tiempo [s]'); ylabel('Amplitud');
grid on; drawnow;

figure;
plot(t2, y2, 'b', 'LineWidth', 1.5); hold on;
title('Respuesta al Escal n de G(s) con Retardo');
xlabel('Tiempo [s]'); ylabel('Amplitud');
grid on; drawnow;
```

Fig. 5: Respuesta al impulso del sistema con retardo.

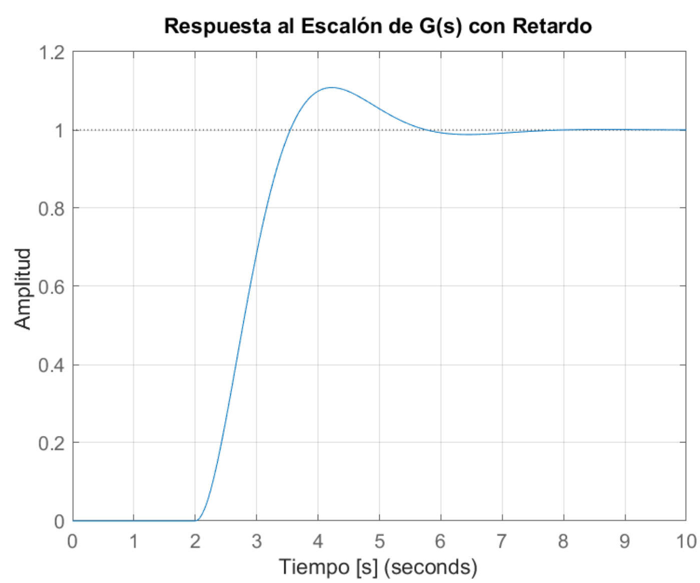

Fig. 6: Respuesta al escalón del sistema con retardo.

Como se observa en la Figura 5 y 6, la presencia del término e^{-2s} provoca un desplazamiento temporal en la respuesta del sistema equivalente al tiempo muerto $T_d = 2s$.

1) Diferencias entre Ambas Respuestas: La principal diferencia radica en la relación fundamental entre las entradas (escalón vs. impulso) en un Sistema Lineal e Invariante en el Tiempo (LTI): la respuesta al escalón es la integral de la respuesta al impulso.

- **Entrada**
  - Respuesta al Escalón: Cambio permanente (Entrada constante).
  - Respuesta al Impulso: Perturbación momentánea (Área unitaria).
- **Valor Final**
  - Respuesta al Escalón: **Constante y no nulo** (1.0).
  - Respuesta al Impulso: **Cero** (Regresa al reposo).
- **Forma de Onda**
  - Respuesta al Escalón: Sigmoidal convergente a 1.0.
  - Respuesta al Impulso: Pulso oscilatorio convergente a 0.
- **Información Clave**
  - Respuesta al Escalón: Ganancia DC (K) y Error en estado estacionario.
  - Respuesta al Impulso: Dinámica interna (Oscilación y Amortiguamiento).

2) Sistema con retardo con marcadores máximo: El siguiente codigo grafica los puntos donde la grafica alcanza los puntos maximos para la respuesta al impulso del sistema con retardo (Fig. 7) y para la respuesta al escalon del sistema con retardo (Fig. 8).

```
% Encontrar m ximos
[max_val, idx] = max(y1);
t_max = t1(idx);    % Tiempo del m ximo en impulso

[max_val2, idx2] = max(y2);
t_max2 = t2(idx2); % Tiempo del m ximo en escal n

figure(); % -- Impulso con retardo + punto m ximo
plot(t1, y1, 'b', 'LineWidth', 1.5); hold on;
plot(t_max, max_val, 'ro', 'MarkerSize', 8, '
    LineWidth', 2);
title('Respuesta al Impulso con Retardo (p. max)');
xlabel('Tiempo [s]'); ylabel('Amplitud'); grid on;

figure(); % --Escal n con retardo + punto m ximo
plot(t2, y2, 'b', 'LineWidth', 1.5); hold on;
plot(t_max2, max_val2, 'ro', 'MarkerSize', 8, '
    LineWidth', 2);
title('Respuesta al Escal n con Retardo (p. max)');
xlabel('Tiempo [s]'); ylabel('Amplitud'); grid on;
```

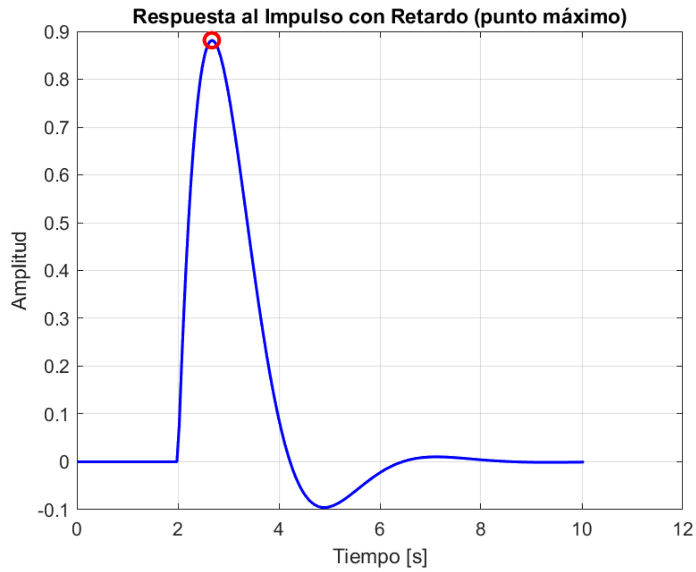

Fig. 7: Respuesta al impulso del sistema con retardo y su valor máximo.

Fig. 8: Respuesta al escalón del sistema con retardo y su valor máximo.

3) Análisis y Verificación del Retardo de Tiempo (τ): El comportamiento inicial del sistema es un aspecto crucial para la caracterización del modelo. A partir de la inspección de los datos (vectores de tiempo y amplitud) y las gráficas de respuesta transitoria (escalón e impulso), se procedió a la verificación del retardo de tiempo.

Determinación del Tiempo Muerto: Se determinó que el sistema presenta un **retardo de tiempo puro** (τ) de 2 segundos.

- **Observación en los Vectores:** Al analizar los vectores de salida, se constata que la amplitud del sistema permanece rigurosamente en cero para todo $t < 2$ s.

- **Significado Físico:** Este fenómeno, también conocido como **tiempo muerto** (dead time), indica el tiempo que tarda la perturbación de entrada en viajar a través del proceso antes de que se inicie cualquier cambio medible en la salida.

4) Describa un ejemplo práctico donde un sistema presente este comportamiento con retardo: **Ejemplo Práctico: Control de Temperatura en un Horno Industrial**

Un ejemplo clásico de un sistema que presenta retardo (τ) y dinámica oscilatoria (segundo orden) es el control de la temperatura de un horno industrial.

- **Retardo de Tiempo (τ):** El 2 s de retardo se debe al tiempo de transporte que tarda el calor generado por el elemento calefactor en llegar al sensor (termopar) y a la propia inercia térmica del sensor. La salida no reacciona hasta que este tiempo ha transcurrido.
- **Dinámica de Segundo Orden (Oscilación):** La capacidad térmica de las paredes del horno y de las piezas que contiene actúa como un sistema de acumulación de energía. Esto, combinado con el retardo puro, provoca que el sistema sea propenso al **sobreimpulso** y la oscilación antes de estabilizarse en el punto de ajuste.

D. Respuesta a señales arbitrarias

1) Replique en MATLAB la señal de excitación arbitraria mostrada en la guia [1]: Para replicar la misma señal de la guia se ha generado el siguiente codigo:

```
t_g = 0:0.1:30;    % Dimension de la grafica
signal = zeros(size(t_g));
%Condiciones por tramos
signal(t_g >= 10 & t_g < 20) = 5;
signal(t_g >= 20) = 10;

figure(); % Grafica
plot(t_g, signal, 'b--', 'LineWidth', 2);
title(' Se al arbitraria dada'); grid on;
xlabel('Tiempo [s]'); ylabel('Amplitud');
```

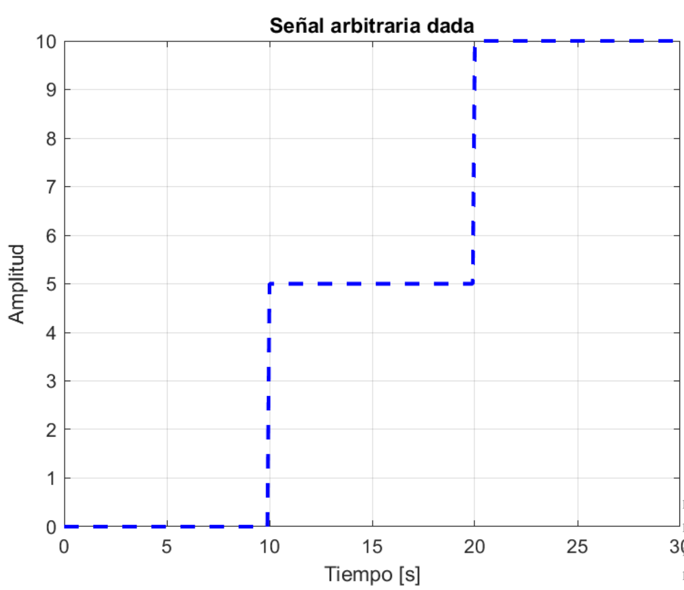

Fig. 9: Señal arbitraria replicada de la guia

2) Analice la respuesta del sistema descrito en el paso anterior (función de transferencia de un sistema de segundo orden con retardo) cuando es excitado por una señal arbitraria, como se muestra en la Figura 2.2 (a). La Figura 2.2 (b) presenta un ejemplo del resultado esperado en la guia [1]: Para replicar la misma señal de la guia se ha generado el siguiente codigo:

```
% --- Respuesta del sistema a la se al arbitraria
[y_arb, tt_arb] = lsim(Gs, signal, t_g);

figure;
plot(tt_arb, y_arb, ...
    t_g, signal, 'LineWidth', 1.8);
xlabel('Tiempo [s]');
ylabel('Amplitud');
title('Respuesta del Sistema a Se al excitaci n
    Arbitraria');
grid on; drawnow;
```

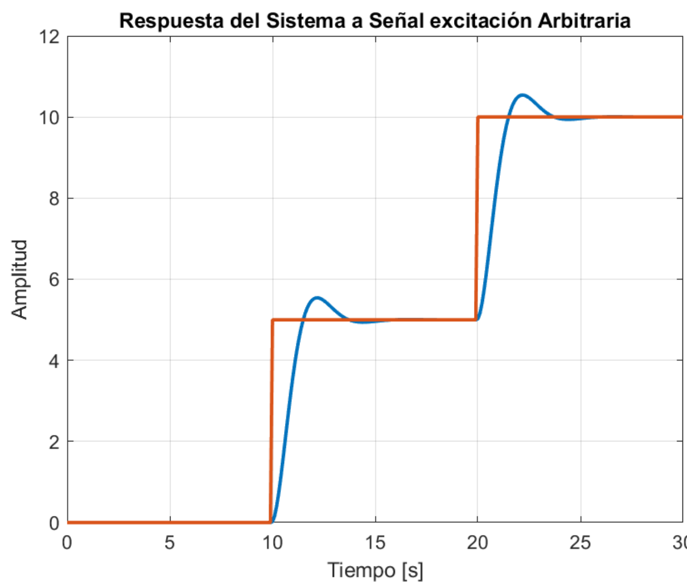

Fig. 10: Respuesta de un sistema de segundo orden con retardo ante una señal de excitación arbitraria.

3) Repita el paso 1 y 2, para la señal arbitraria de la Figura 2.3 de la guia [1]: Para analizar una señal de entrada personalizada, se define una señal $u(t)$ por tramos que incluye valores constantes y una rampa lineal. Luego, se emplea `lsim()` para simular la salida del sistema.

```
t_arb = 0:0.1:40;
u_arb = zeros(size(t_arb));

u_arb(t_arb >= 0 & t_arb < 10) = 0;
u_arb(t_arb >= 10 & t_arb < 20) = 5;
idx = (t_arb >= 20 & t_arb < 30);
u_arb(idx) = linspace(15, 25, sum(idx));
u_arb(t_arb >= 30) = 25;

[y_arb, tt_arb] = lsim(Gs, u_arb, t_arb);

figure;
plot(tt_arb, y_arb, 'LineWidth', 1.8); grid on;
title('Respuesta del sistema a se al arbitraria');
xlabel('Tiempo [s]'); ylabel('Amplitud');
```

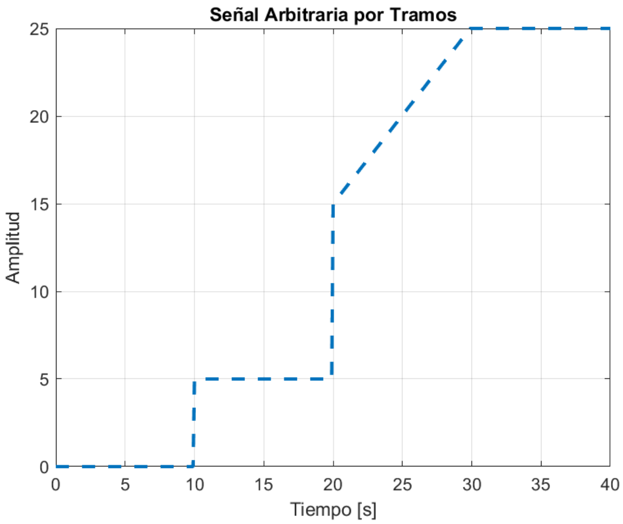

Fig. 11: Señal arbitraria dada por tramos.

Para replicar el paso 2 para esta nueva señal con difererentes tramos se usa la siguiente lineas de codigo (parecido al anterior)

```
[y_arb2, tt_arb2] = lsim(Gs, u_arb, t_arb);

figure;
plot(t_arb, u_arb, '--', ...
    tt_arb2, y_arb2, 'LineWidth', 2);
xlabel('Tiempo [s]'); ylabel('Amplitud');
title('Respuesta una Se al Arbitraria por Tramos');
grid on; drawnow;
```

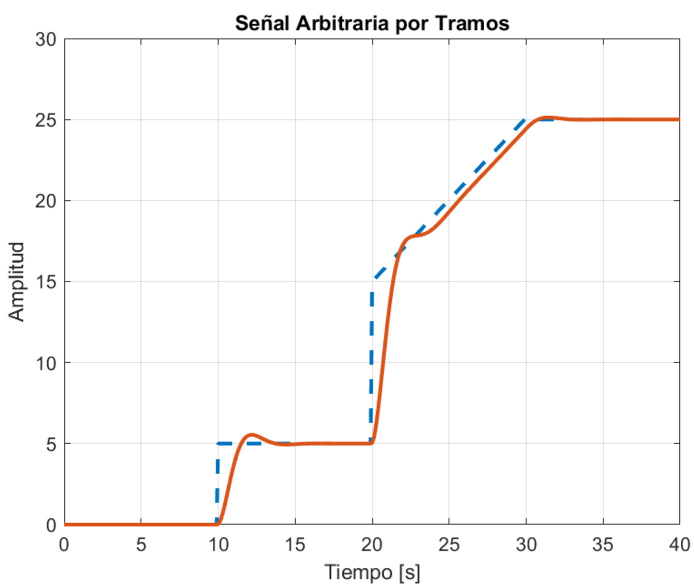

Fig. 12: Respuesta de un sistema de segundo orden con retardo ante una señal de excitación arbitraria que incluye rampas.

E. Actividad Reto: Señal PRBS y respuesta del sistema

1) Generar una señal aleatoria: Finalmente, se genera una señal PRBS combinada con componentes senoidales, rampas y escalones, tanto como de subida y de bajada.

Listing 1: Generación de la señal PRBS

```
t_prbs = 0:0.1:40;
u_prbs = zeros(size(t_prbs));

idx1 = (t_prbs >= 0 & t_prbs < 10);
u_prbs(idx1) = sin(2*pi*0.25*t_prbs(idx1));

idx2 = (t_prbs >= 10 & t_prbs < 15);
u_prbs(idx2) = 3;

idx3 = (t_prbs >= 15 & t_prbs < 20);
u_prbs(idx3) = linspace(3,5,sum(idx3));

idx4 = (t_prbs >= 20 & t_prbs < 25);
u_prbs(idx4) = 5;

idx5 = (t_prbs >= 25 & t_prbs < 30);
u_prbs(idx5) = linspace(5,-3,sum(idx5));

idx6 = (t_prbs >= 30);
u_prbs(idx6) = -3;

% --- Gr fica de se al PRBS
figure;
plot(t_prbs, u_prbs, 'LineWidth', 1.5); hold on;
xlabel('Tiempo [s]');
ylabel('Amplitud');
title('Se al PRBS Combinada Aleatoria');
grid on;
drawnow;
```

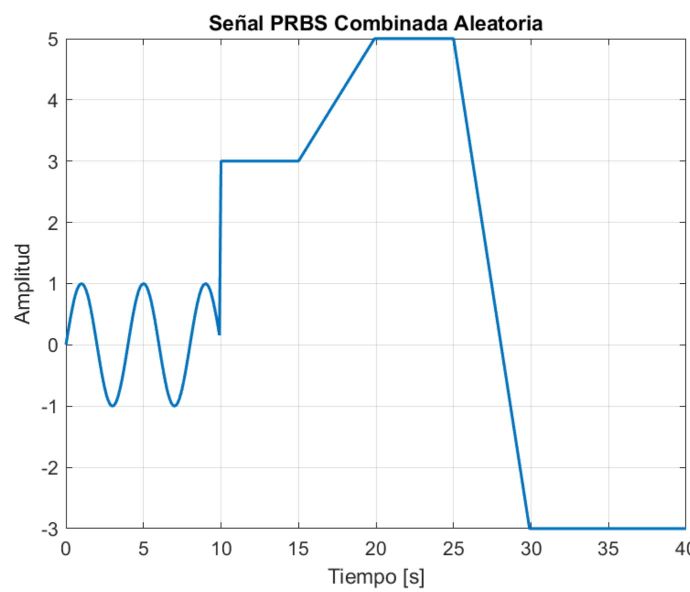

Fig. 13: Señal PRBS combinando Rampas, senoidales, escalones, etc.

2) Utilizar la función de transferencia definida en la ecuación 2.2 para obtener la respuesta a la señal de excitación arbitraria generada en el punto anterior: Para generar la respuesta con retardo de 2 [s] usamos la ec. 5, la cual era:

$$G(s) = \frac{3}{s^2+2s+3}e^{-2s}$$

El codigo que hace para graficar la señal PRBS con su respuesta del sistema con un retardo de 2 [s] es:

```
% --- Respuesta del sistema PRBS con retardo
y_prbs = lsim(Gs_delay, u_prbs, t_prbs);

% --- Gr fica de se al PRBS y respuesta
figure;
plot(t_prbs, u_prbs, 'LineWidth', 1.5); hold on;
plot(t_prbs, y_prbs, 'LineWidth', 2);
xlabel('Tiempo [s]'); ylabel('Amplitud');
title('Se al PRBS Combinada y Respuesta del Sistema con Retardo');
legend('Entrada PRBS', 'Salida del Sistema', 'Location', 'best');
grid on; drawnow;
```

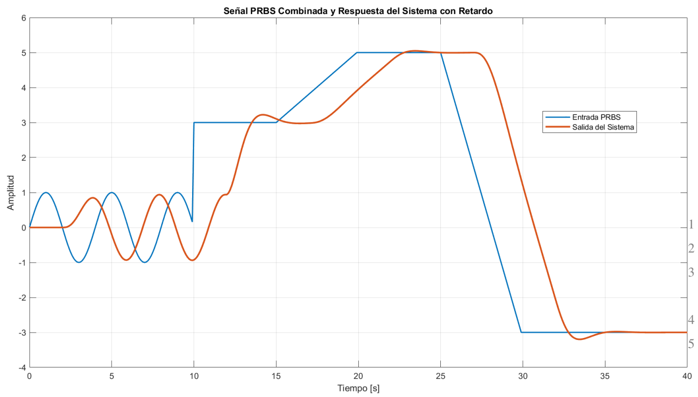

Fig. 14: Señal PRBS y respuesta del sistema con retardo.

> En la Figura 14 se aprecia cómo la señal PRBS, compuesta por diferentes segmentos (senoidal, rampas y escalones), permite evaluar el comportamiento dinámico del sistema bajo excitaciones variadas, reproduciendo condiciones similares a señales de prueba reales usadas en identificación de sistemas.

F. Reto 2:

1) La segunda parte del desafío consiste en replicar el experimento (tareas [1-3]) en Python utilizando la biblioteca control. Los conocimientos básicos necesarios para completar el desafío se impartirán durante la práctica: La señal PRBS en python es la misma que la de matlab, en mi caso use google colab, lo que necesitamos adicional es las librerias:

Listing 2: Generación de la señal PRBS en python

```
# ============
# Instalar la librer a "control"
# =================
!pip install control

# ==============
# Importar librer as necesarias
# ================
import numpy as np
import matplotlib.pyplot as plt
import control as ctrl
```

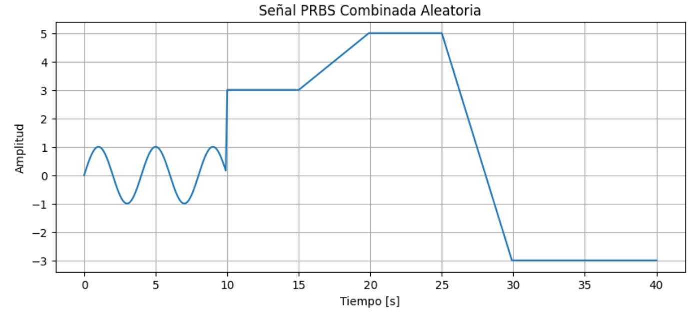

Fig. 15: Señal PRBS combinando Rampas, senoidales, escalones, etc. En python

El codigo que hace para graficar la señal PRBS con su respuesta del sistema con un retardo de 2 [s] usando la biblioteca **control** es:

Definimos la funcion de transferencia a la respuesta con retardo de 2 [s] usamos la ec. 5:

```
# Crear sistema con retardo (usando aprox de Pad )
Gs = ctrl.tf(num, den)
num_delay, den_delay = ctrl.pade(delay, 4)   # Aproximaci n de Pad  orden 4
delay_tf = ctrl.tf(num_delay, den_delay)
Gs_delay = ctrl.series(delay_tf, Gs)
```

Adicionalme para graficar respuesta del sistema con retardo hacemos:

```
t_out, y_prbs = ctrl.forced_response(Gs_delay, T=t_prbs, U=u_prbs)

# Gr fica de la se al PRBS y la respuesta del sistema
plt.figure(figsize=(10, 4))
plt.plot(t_prbs, u_prbs, label='Entrada PRBS', linewidth=1.5)
plt.plot(t_out, y_prbs, label='Salida del Sistema', linewidth=2)
plt.xlabel('Tiempo [s]')
plt.ylabel('Amplitud')
plt.title('Se al PRBS Combinada y Respuesta del Sistema con Retardo')
plt.legend(loc='best')
plt.grid(True)
plt.show()
```

En la Fig. 16 se puede apreciar la grafica:

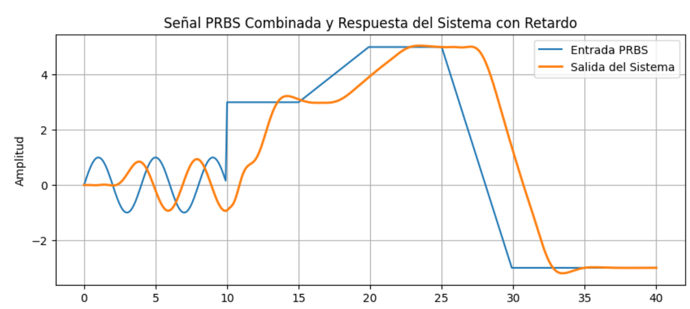

Fig. 16: Señal PRBS y respuesta del sistema con retardo usando biblioteca Control.

IV. Código de MATLAB

El código que se usó para desarrollar este taller en MATLAB está cargado en **GitHub** como se pidió en diapositiva, en el siguiente enlace se puede obtener el código: Obtener código.

Y el codigo para python se encuentra en el siguiente enalce: Google colab - Control

V. Conclusiones

El desarrollo del presente taller permitió comprender de manera integral el comportamiento dinámico de un sistema lineal invariante en el tiempo (LTI) de segundo orden con retardo, a través del análisis de su respuesta frente a diferentes tipos de señales de excitación.

Las simulaciones realizadas en MATLAB evidenciaron que el retardo temporal (e^{-2s}) desplaza la respuesta del sistema sin alterar su forma ni su estabilidad, destacando su efecto directo sobre la fase y la velocidad de respuesta. Asimismo, se confirmó que la respuesta al impulso revela la dinámica interna del sistema, mientras que la respuesta al escalón permite evaluar su comportamiento en régimen permanente.

Por otra parte, el uso de señales arbitrarias y combinadas (tipo PRBS) demostró ser una herramienta eficiente para excitar al sistema en un rango amplio de frecuencias, facilitando la observación simultánea de fenómenos transitorios y estacionarios. Este enfoque es especialmente útil en tareas de identificación de sistemas, donde se requiere obtener un modelo representativo a partir de datos experimentales.

Finalmente, la implementación complementaria en Python, utilizando la librería `control`, corroboró la validez de los resultados obtenidos en MATLAB, mostrando la versatilidad de ambas plataformas para el análisis y simulación de sistemas dinámicos con retardo.

References

[1] G. Guzmán and P. A. Barbecho Bautista, *Guía de Laboratorio - Taller 2: Respuesta de un sistema a señales arbitrarias (MATLAB)*, Universidad de Cuenca, Facultad de Ingeniería, 2024.

[2] K. Ogata, *Ingeniería de Control Moderna*, 5th ed., Pearson Educación, 2010.

[3] MathWorks Documentation, *Control System Toolbox - Analyze and Design Control Systems*, Available: https://www.mathworks.com/help/control/, Accessed: Oct. 2025.

[4] L. Ljung, *System Identification: Theory for the User*, 2nd ed., Prentice Hall, 1999.