

# Guía Laboratorio 5

## Procesamiento Digital de Señales

### Transformada Z

Paula Pérez, Alejandro Escobar y Cristian Ríos

2024-1

#### NOTAS:

- Enviar el informe del laboratorio con el siguiente nombre: *Lab5\_PDS\_Apellido\_Nombre.ipynb*
- Enviar junto con el informe los archivos adicionales generados y descargados. Todo esto debe ir en un archivo comprimido con el siguiente nombre: *Lab5\_PDS\_Apellido\_Nombre.zip*
- OJO! Recuerde tener cuidado con la indentación y caracteres como el guión bajo y las llaves cuando copie y pegue el código entregado en esta guía.
- Las preguntas deberán ser resueltas en el notebook indicando sus respectivos numerales.

## 1. Transformada Z

Considere los siguientes sistemas representados con ecuaciones en diferencias. Elija una ecuación de acuerdo a su último número de cédula, aplique la transformada Z y halle la función de transferencia del sistema.

0.  $y[n] = x[n] + 8x[n-1] - 2y[n-2] + 5y[n-1]$
1.  $y[n] = y[n-1] - 0,5y[n-2] + x[n] + x[n-1]$
2.  $y[n] = 6y[n-1] - 10y[n-2] + 3x[n-2] - 7x[n-2] + x[n]$
3.  $y[n] + 0,3y[n-2] = x[n] + 15x[n-1] - 2y[n-1]$
4.  $y[n] - 0,7y[n-1] + 0,2y[n-2] = x[n] - x[n-1] + x[n-2]$
5.  $y[n] = 5,4y[n-2] - 2y[n-2] + x[n-1] + 5x[n-1]$
6.  $y[n] = 4y[n-1] - 3y[n-2] + y[n-3] + x[n-1] + x[n-2]$
7.  $y[n] = -2y[n-1] - 15y[n-3] + 10x[n-2] + 10x[n-2] - 0,2x[n]$
8.  $y[n] - 2,83y[n-1] + 4y[n-3] = 2x[n] - 3x[n-1] + x[n-2]$
9.  $y[n] = y[n-1] + 0,5y[n-1] + x[n-2] - 10x[n-3]$

**Nota:** Para desarrollar el laboratorio es necesario representar el numerador y denominador de la función de transferencia usando listas en Python. Recuerde que las listas solo contienen los coeficientes de los polinomios de Z ordenados de mayor a menor orden.

1. Importe las funciones del archivo adjunto *ztrans.py* y úselas para graficar la respuesta en frecuencia y fase, la respuesta al impulso, la respuesta al escalón, y el diagrama de polos y ceros.
2. ¿Dónde están ubicados los polos y los ceros?
3. ¿Qué se puede decir de la respuesta en frecuencia y fase de la función de transferencia?
4. ¿Qué se puede decir acerca de la respuesta al impulso y al escalón?
5. ¿Qué se puede decir acerca de la estabilidad del sistema?

## 2. Transformada Z: Introducción a los filtros digitales

1. Genere una señal sinusoidal cuya frecuencia se incremente en el tiempo de forma cuadrática. La señal debe tener una duración de 40 segundos y una frecuencia de muestreo de 5000 Hz.
2. Grafique y escuche la señal ¿Qué se puede decir acerca del audio?
3. Genere la función de transferencia de un filtro de respuesta finita al impulso (FIR) usando las siguientes instrucciones.

```
import scipy.signal as signal
n = 9
num = signal.firwin(n, [0.2, 0.4], pass_zero=False)
den=np.zeros(n)
den[0] = 1
```

4. Grafique la respuesta en frecuencia y fase, la respuesta al impulso, la respuesta al escalón, y el diagrama de polos y ceros del filtro FIR.
5. ¿Qué puede concluir acerca de las gráficas anteriores? ¿Qué tipo de filtro es? ¿Cuáles son las frecuencias de corte? ¿Dónde están ubicados los polos y ceros? ¿Qué puede decir de la estabilidad del filtro?
6. Calcule la respuesta del filtro ante la señal generada previamente. Grafique la señal antes y después de pasar por el filtro. ¿Qué le ocurrió a la señal?

Tip: Recuerde que para calcular la respuesta del sistema ante una entrada debe usar la convolución.

```
data2=np.convolve(data, num, mode='same')
```

7. Aumente progresivamente el orden  $n$  del filtro (mínimo 3 veces), repita el procedimiento anterior (items 3, 4, 5, y 6), y concluya. Que puede decir de la respuesta en frecuencia, y de los polos y ceros.
8. Genere una señal sinusoidal cuya frecuencia se incremente en el tiempo de forma cúbica, la señal debe tener la misma duración (40 seg) y una frecuencia de muestreo de 3000 Hz. Repita los pasos 3, 4, 5 y 6. Concluya de acuerdo a las gráficas obtenidas.

## 3. Agregando polos y ceros al sistema

1. Retorne al sistema anterior con  $n = 10$ .
2. Use la función `computeZ(num, den, zeros, poles, data)` que se encuentra en el archivo adjunto (ztrans.py) para agregar polos y ceros a una función de transferencia. Observe un ejemplo a continuación:

```
n = 10
num = signal.firwin(n, [0.1, 0.4], pass_zero=False)
den=np.zeros(n)
den[0]=1

# Agregue polos conjugados en los puntos z1=-sqrt(2)/4+sqrt(2)/2j y z2=-sqrt(2)/4-sqrt(2)/2j y ...
# un cero en z=0.5
computeZ(num, den, [0.5], [complex(-np.sqrt(2)/4,np.sqrt(2)/2), ...
                           complex(-np.sqrt(2)/4,-np.sqrt(2)/2)], data)
```

3. Explique el comportamiento del sistema si se agrega de forma independiente:
  - Dos ceros conjugados de magnitud 1.3
  - Un polo real de magnitud 1.4

## 4. Conclusiones

Realice conclusiones generales sobre la práctica. Recuerde que las conclusiones son parte fundamental de su evaluación en el laboratorio, tómese el tiempo de pensar las conclusiones.