FACULTAD DE INGENIERÍA

BIOINGENIERÍA

Bioseñales y sistemas

Señales y Sistemas discretos

1. OBJETIVOS

General

Desarrollar habilidades en el manejo del lenguaje Python aplicado a las señales discretas elementales.

Específicos

- Utilizar Python para implementar funciones que generen señales discretas básicas, como impulsos unitarios, escalones y señales sinusoidales discretas.
- Aplicar Python para realizar operaciones básicas en señales discretas, como la adición, sustracción y multiplicación.
- Emplear bibliotecas como Matplotlib para visualizar gráficamente estas señales y entender sus características fundamentales.

2. PROCEDIMIENTO

Realice los enunciados en un Notebook o un script entregable.

Representación de señales discretas

Para la representación de una secuencia discreta en Python, se utiliza un vector fila el cual contiene los valores correspondientes. Sin embargo, dicho vector fila no incluye ninguna información acerca de cuál es la muestra correspondiente a cada valor del vector. Es por esto que para una correcta representación de una señal discreta *x(n)*, es necesario crear un vector para cada *x* y otro para cada n. Por ejemplo, la secuencia $x(n) = \{5, 6, \hat{7}, 5, 4, 3\}$ se representaría como:

n = [-2, -1, 0, 1, 2, 3]
x = [5, 6, 7, 5, 4, 3]

Donde el sombrero indica cuál es la muestra correspondiente al índice $n = 0$. En caso de que se trate de una señal discreta en la que no se requiera información de la posición, o en la que la información de la posición sea trivial (es decir que la señal inicie en $n = 0$) se puede omitir el vector correspondiente a las muestras.

Implemente las siguientes funciones, tanto las que se presentan como las que se propone implementar: intente usar la lógica empleada en la función impulso para la generación de las de más funciones.

Secuencias Discretas Elementales

Impulso unitario

El impulso unitario se define como:

$$\delta(n-n_0) = \begin{cases} 1 & si\ n = n_0 \\ 0 & si\ n \neq n_0 \end{cases}$$

En Python, implementamos el impulso unitario mediante la siguiente función:

```
def impseq(n0, n1, n2):
#   Genera x(n) = delta(n-n0); n1 <= n <= n2
# ----------------------------------------------
    n = np.arange(n1,n2+1) # Se crea el vector de muestras
    x = (n-n0) == 0
    return [x,n]
```

Escalón unitario

El escalón unitario se define como:

$$u(n-n_0) = \begin{cases} 1 & si\ n \geq n_0 \\ 0 & si\ n < n_0 \end{cases}$$

1. Implemente una función en Python que permita generar una función escalón unitario, definida en un intervalo $n_1 \leq n_0 \leq n_2$

Rampa

La función rampa se define como:

$$r(n-n_0) = \begin{cases} 0 & si\ n < n_0 \\ n-n_0 & si\ \ n \geq n_0 \end{cases}$$

2. Implemente una función en Python que permita generar una función rampa, definida en un intervalo $n_1 \leq n_0 \leq n_2$ con una pendiente m.

Exponenciales complejas

Las exponenciales complejas son de la forma:

$$x(n) = e^{(\sigma + j\omega 0)n}, \forall\, n$$

En la que σ produce una atenuación si es negativo o una amplificación si es positivo. En Python se hace uso del comando *np.exp*.

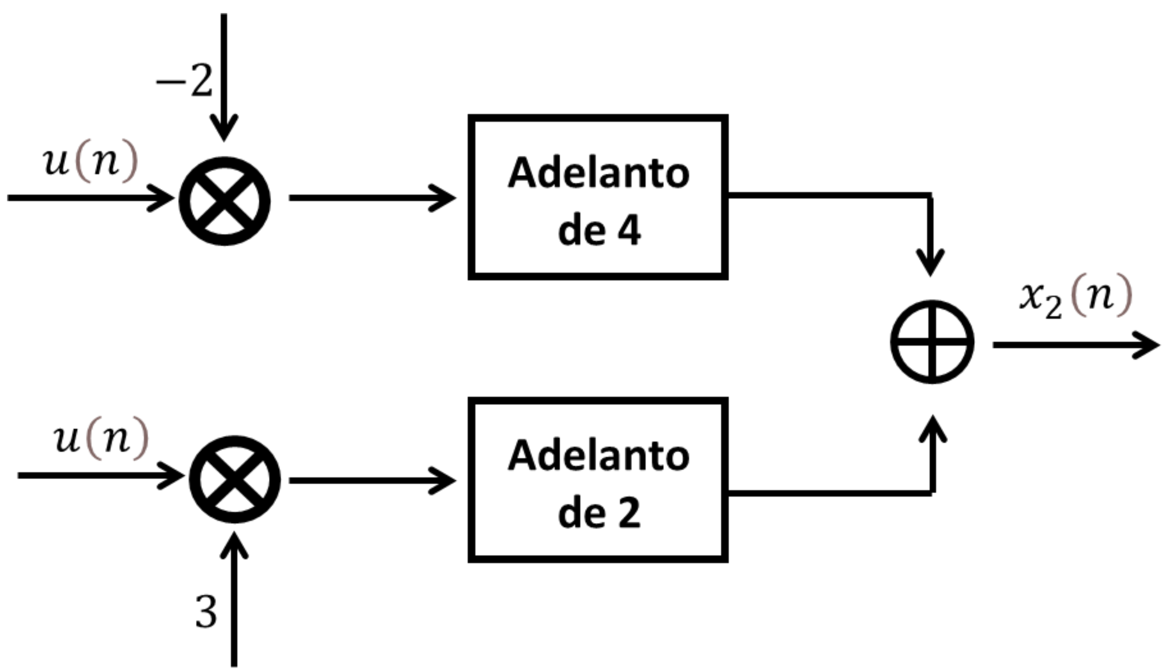
Sinusoides

Estas secuencias son de la forma:

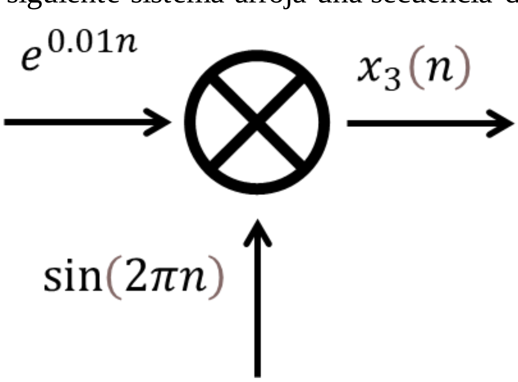
$$x(n) = \sin\,(\omega_0 + \theta_0)$$

3. Genere las siguientes secuencias usando las funciones básicas de Python que se han presentado. Grafique los resultados.
   a. $x_1(n) = \; 3\delta(n+1) + 5\delta(n+3) + 3\delta(n+2) + 3\delta(n) + \; \delta(n)$. ¿Cómo debe ser el vector de muestras?
   b. El siguiente sistema arroja una secuencia definida entre $-6 \le n \le 6$

   Nota: el símbolo ⊗ corresponde a un multiplicador y el ⊕ a un sumador



   c. El siguiente sistema arroja una secuencia definida entre $0 \le n \le 100$



d. $x_4(n) = 2r(n+3) - r(n-2) - 5u(n-3), -10 \leq n \leq 10$

4. Genere la siguiente secuencia

$$x[n] = \begin{cases} r[n] & 0 \leq n \leq 5 \\ r[n-5] & 6 \leq n \leq 11 \\ r[n-10] & 12 \leq n \leq 17 \end{cases}$$

   a. Derive la secuencia
   b. Realice un subplot con las dos secuencias, la original, y la derivada. Describa lo que observa y el porqué no es exactamente la derivada del caso continuo.

5. Sea $x(n) = \{0,1,2,3,4,\hat{5},4,3,2,1,0,1,2,3,4,5,5,5,5,5,10,10,10,10\}$. Genere la secuencia anterior y grafique los resultados. Use las funciones que generó antes para generar la secuencia concatenando secuencias más simples. Además, encuentre las siguientes secuencias:
   a. $x_5(n) = 2x(n-4) + x(n)$
   b. $x_6(n) = 0.001e^{0.5n}x(n) + 10sin(0.05\pi n)\, x(n+2), -20 \leq n \leq 20$

3. Anexos

Convenciones de operadores de sistemas