EJERCICIO 2.1

Semejanzas:

1. **Propósito común**: Todas buscan representar funciones periódicas mediante una combinación de senos y cosenos (o exponenciales complejas).
2. **Convergencia**: En todos los casos, la serie de Fourier converge a la función original, en el sentido de que la serie puede aproximar cualquier función periódica en un intervalo.
3. **Coeficientes**: En todas las formas se obtienen coeficientes que dependen de la función a aproximar. Estos coeficientes son los mismos, solo que la forma en que se escriben varía.
4. **Dependencia del período**: El período de la función original TTT es fundamental en todas las representaciones, ya que las frecuencias de los términos de la serie estarán relacionadas con 1/T1/T1/T.

Diferencias:

1. **Serie exponencial**: La función se representa usando exponenciales complejas. Es decir:

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i\frac{2\pi n}{T}t}$$

   Donde Cn son los coeficientes complejos.

2. **Serie trigonométrica**: Aquí la representación es explícita en términos de senos y cosenos.

$$f(t) = a_0 + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{2\pi n}{T}t\right) + b_n \sin\left(\frac{2\pi n}{T}t\right) \right]$$

   donde a0, an, y bn son los coeficientes obtenidos a partir de la función original.
3. **Serie compacta**: Es una versión más compacta y moderna que combina la forma exponencial y trigonométrica, expresando la serie de Fourier en una forma eficiente, pero no difiere demasiado en términos de los coeficientes. En algunos contextos, puede referirse a una forma simplificada de las series anteriores.

Diferencias y semejanzas entre transformada de Fourier, transformada de Fourier en tiempo discreto, y transformada discreta de Fourier.

Semejanzas:

1. **Objetivo común**: Las tres transformadas buscan descomponer una señal en sus componentes frecuenciales, permitiendo analizar la frecuencia de la señal, lo que es útil en áreas como procesamiento de señales, comunicaciones, análisis de imágenes, etc.

2. **Representación en frecuencia**: Las tres transformadas proporcionan una representación de la señal original en el dominio de la frecuencia, aunque en diferentes contextos (tiempo continuo o discreto).
3. **Base matemática**: Todas ellas se basan en las funciones sinusoidales (senos y cosenos) o exponentes complejos (como e^jwt para la transformada continua y e^j2pik/N para las transformadas discretas).

Diferencias:

1. Transformada de Fourier (TF):

- **Dominio**: Se aplica a **señales continuas en el tiempo**.
- **Fórmula**:

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t} dt$$

• Salida: La salida es una **función continua** que describe cómo están distribuidas las frecuencias a lo largo del espectro de la señal.
• Aplicación: Es útil para analizar señales que tienen componentes de frecuencia continua, como señales analógicas.
• Condiciones **de existencia**: Para que la TF exista, la señal debe cumplir ciertas condiciones de integralidad (como ser de energía finita o tener un decaimiento adecuado en el infinito).

2. Transformada de Fourier en Tiempo Discreto (TFTD):

- **Dominio**: Se aplica a **señales discretas en el tiempo**, pero en **tiempo continuo en frecuencia**.
- **Fórmula**:

$$X(f) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi f n}$$

• Salida: El resultado es una **función continua en frecuencia**, similar a la TF, pero la señal de entrada es discreta.
• Aplicación: Esta transformada se utiliza para analizar señales discretas en el tiempo, pero con un espectro de frecuencias continuo.

• Condiciones **de existencia**: La señal debe ser discreta y, generalmente, su espectro es analizado en un intervalo continuo de frecuencias, aunque la señal original es discreta.

3. Transformada Discreta de Fourier (DFT):

- **Dominio**: Se aplica a **señales discretas en el tiempo** y produce un **espectro discreto**.
- **Fórmula**

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi kn/N}, \quad k = 0, 1, 2, \ldots, N-1$$

• Salida: La salida es una **secuencia discreta de frecuencias** que corresponde a las frecuencias que componen la señal. El espectro es finito y tiene NNN puntos.
• Aplicación: Se usa principalmente en el procesamiento digital de señales, especialmente en la implementación de algoritmos de computación como la FFT (Fast Fourier Transform), donde las señales son procesadas como secuencias de datos finitos.

• Condiciones **de existencia**: La señal debe ser una secuencia de longitud finita (generalmente periódica o representada de forma periódica).

CONSULTE EN QUÉ CONSISTE EL ALGORITMO FAST FOURIER TRANSFORM - (FFT) Y SU UTILIDAD PARA EL CÁLCULO DE LA TRANSFORMADA DISCRETA DE FOURIER. EXPLIQUE EN DETALLE EL ALGORITMO FFT Y SU COSTO COMPUTACIONAL VS EL DE LA TRANSFORMADA DISCRETA.

• La **DFT directa** es computacionalmente costosa, con un tiempo de cálculo de O(N^2)
• La **FFT**, usando el algoritmo de Cooley-Tukey, reduce el tiempo de cálculo a O(NlogN), lo que permite calcular la DFT de manera mucho más eficiente.
• El uso de la FFT es crucial en la práctica, especialmente cuando se trabaja con señales grandes o en tiempo real, ya que ofrece una mejora drástica en la eficiencia computacional sin sacrificar precisión.

EJERCICIO 2.4

Distorsión Total de Armónicos (THD) y Factor de Potencia

Distorsión Total de Armónicos (THD) es una medida de la distorsión armónica en una señal, expresada como la relación entre la suma de las amplitudes de los armónicos (frecuencias múltiples de la frecuencia fundamental) y la amplitud de la frecuencia fundamental:

$$THD = \frac{\sqrt{\sum_{n=2}^{\infty} |A_n|^2}}{|A_1|}$$

donde A es la amplitud de la fundamental, y An son las amplitudes de los armónicos.

Factor de Potencia (PF) es la relación entre la potencia real P y la potencia aparente S:

$$PF = \frac{P}{S} = \cos(\phi)$$

El factor de potencia se ve afectado por la distorsión armónica, ya que los armónicos pueden reducir la potencia real y aumentar la potencia reactiva.

Cálculo del THD usando la FFT:

La FFT descompone una señal en sus componentes frecuenciales. Para calcular el THD a partir de la FFT:

1. Realiza la FFT de la señal de corriente.
2. Calcula la amplitud de la frecuencia fundamental (A1) y las amplitudes de los armónicos (An).
3. Aplica la fórmula del THD.

Cálculo de la distorsión del factor de potencia con el THD:

Un alto valor de THD implica una mayor distorsión armónica, lo que disminuye el factor de potencia. El **factor de potencia distorsionado (DPF)** se puede aproximar usando la relación entre el THD y el ángulo de fase resultante de la distorsión armónica.

Ejemplo: Rectificador de onda completa con carga

i) Carga resistiva neta:

- La corriente tiene solo la frecuencia fundamental, por lo que el THD es bajo y el factor de potencia es cercano a 1.

ii) Carga RC en serie:

- Los armónicos aumentan debido a la carga reactiva. El THD aumentará, y el factor de potencia disminuirá.

Condiciones para simulaciones:

- **Resistiva (R)**: Prueba con diferentes valores de R (e.g., 10Ω, 100Ω).
- **RC**: Varía R y C (e.g., R=100Ω, C=10µF).

Discusión:

- Para la carga resistiva, el THD es bajo, por lo que el factor de potencia se mantiene cerca de 1.
- Con la carga RC, los armónicos aumentan, lo que incrementa el THD y reduce el factor de potencia debido a la mayor presencia de potencia reactiva.

Modulación de Amplitud (AM) y Detección Coherente

La **modulación por amplitud (AM)** es un tipo de modulación en la que la amplitud de una señal portadora se modula de acuerdo con la señal mensaje. La **detección coherente** en AM implica que el receptor utiliza una señal portadora de la misma frecuencia y fase que la transmitida para demodular correctamente la señal. Esta técnica es ampliamente usada en comunicaciones analógicas, como en la transmisión de radio AM.

Aplicaciones de AM y Detección Coherente:

- **Radiodifusión AM**: Usada en estaciones de radio AM para transmitir música, voz y otros contenidos.
- **Transmisiones de datos**: AM se utiliza en algunas formas de transmisión de datos por su simplicidad y efectividad en entornos con poca interferencia.
- **Comunicación en satélites**: Algunos sistemas de comunicación satelital utilizan AM para transmitir información.

Ejemplo Ilustrativo en Python:

A continuación, se presenta un ejemplo de cómo se puede graficar una señal AM con dos tipos de señales mensaje (pulso rectangular y coseno) en el dominio del tiempo y de la frecuencia usando **rfft** de `numpy`.

```
import numpy as np

import matplotlib.pyplot as plt



# Definición de parámetros

Fs = 10000  # Frecuencia de muestreo en Hz

T = 1       # Duración de la señal en segundos

t = np.linspace(0, T, int(Fs*T), endpoint=False)  # Vector de tiempo



# Definir señales mensaje
```

```
def mensaje_rectangular(t, frecuencia):
    return 0.5 * (1 + np.sign(np.sin(2 * np.pi * frecuencia * t)))


def mensaje_coseno(t, frecuencia):
    return np.cos(2 * np.pi * frecuencia * t)


# Función para la modulación AM
def modulacion_am(senal_mensaje, fc, indice_modulacion):
    portadora = np.cos(2 * np.pi * fc * t)  # Señal portadora
    return (1 + indice_modulacion * senal_mensaje) * portadora  # AM


# Parámetros de modulación
fc = 1000  # Frecuencia de la portadora en Hz
indice_modulacion = 0.5  # Índice de modulación
frecuencia_mensaje = 50  # Frecuencia de la señal mensaje


# Generar señales de mensaje
senal_mensaje_rectangular = mensaje_rectangular(t, frecuencia_mensaje)
senal_mensaje_coseno = mensaje_coseno(t, frecuencia_mensaje)


# Generar señales AM
senal_am_rectangular = modulacion_am(senal_mensaje_rectangular, fc, indice_modulacion)
senal_am_coseno = modulacion_am(senal_mensaje_coseno, fc, indice_modulacion)
```

```
# Graficar las señales en el dominio del tiempo
plt.figure(figsize=(10, 6))

plt.subplot(2, 2, 1)
plt.plot(t, senal_mensaje_rectangular)
plt.title('Mensaje Rectangular en el Tiempo')
plt.xlabel('Tiempo (s)')
plt.ylabel('Amplitud')

plt.subplot(2, 2, 2)
plt.plot(t, senal_am_rectangular)
plt.title('Señal AM con Mensaje Rectangular')
plt.xlabel('Tiempo (s)')
plt.ylabel('Amplitud')

plt.subplot(2, 2, 3)
plt.plot(t, senal_mensaje_coseno)
plt.title('Mensaje Coseno en el Tiempo')
plt.xlabel('Tiempo (s)')
plt.ylabel('Amplitud')

plt.subplot(2, 2, 4)
```

```
plt.plot(t, senal_am_coseno)
plt.title('Señal AM con Mensaje Coseno')
plt.xlabel('Tiempo (s)')
plt.ylabel('Amplitud')

plt.tight_layout()
plt.show()

# Graficar las señales en el dominio de la frecuencia usando rfft
frequencies = np.fft.rfftfreq(len(t), 1/Fs)

# FFT de las señales
fft_rectangular = np.fft.rfft(senal_am_rectangular)
fft_coseno = np.fft.rfft(senal_am_coseno)

# Graficar en el dominio de la frecuencia
plt.figure(figsize=(10, 6))

plt.subplot(2, 1, 1)
plt.plot(frequencies, np.abs(fft_rectangular))
plt.title('Espectro de la Señal AM con Mensaje Rectangular')
plt.xlabel('Frecuencia (Hz)')
plt.ylabel('Amplitud')
```

plt.subplot(2, 1, 2)

plt.plot(frequencies, np.abs(fft_coseno))

plt.title('Espectro de la Señal AM con Mensaje Coseno')

plt.xlabel('Frecuencia (Hz)')

plt.ylabel('Amplitud')

plt.tight_layout()

plt.show()

Explicación del Código:

1. **Señales de Mensaje**: Se definen dos tipos de señales de mensaje:
    - **Mensaje Rectangular**: Usamos una función de señal cuadrada o pulso rectangular.
    - **Mensaje Coseno**: Usamos una función coseno.
2. **Modulación AM**: La función `modulacion_am` modula la amplitud de la portadora según la señal de mensaje, usando el índice de modulación especificado.
3. **Graficado en el Tiempo**: Las señales de mensaje y sus correspondientes señales AM se grafican en el dominio del tiempo.
4. **Transformada Rápida de Fourier (FFT)**: Se utiliza `np.fft.rfft` para calcular la transformada de Fourier de las señales AM y graficarlas en el dominio de la frecuencia. La función `rfft` es adecuada para señales reales.

Salida del Código:

- **Dominio del Tiempo**: Se visualizarán las señales de mensaje (rectangular y coseno) y las señales AM correspondientes.
- **Dominio de la Frecuencia**: Se observarán los espectros de las señales AM, donde se pueden identificar los componentes de la frecuencia portadora y las posibles componentes de los armónicos dependiendo del índice de modulación.

Conclusión:

Este ejemplo permite explorar cómo la modulación AM afecta a diferentes señales de mensaje (rectangular y coseno) y cómo se pueden analizar tanto en el dominio del tiempo como en el de la frecuencia utilizando la FFT. Los usuarios pueden ajustar el índice de modulación y observar cómo varían los resultados.

Referencias Generales:

1. **Libros sobre Modulación y Comunicaciones**:
   - **Proakis, J.G. (2001).** *Digital Communications*. McGraw-Hill.
     Este libro es un texto clásico que cubre en detalle la teoría de las comunicaciones digitales, incluyendo la modulación AM y otros tipos de modulación. Es una excelente referencia para la modulación y análisis de señales en comunicaciones.
   - **Haykin, S. (2001).** *Communication Systems*. Wiley.
     Otro texto fundamental sobre comunicaciones que explica diversos esquemas de modulación, incluyendo AM, y su implementación.
2. **Teoría de la Transformada Rápida de Fourier (FFT)**:
   - **Brigham, E.O. (1988).** *The Fast Fourier Transform and Its Applications*. Prentice Hall.
     Este libro es una excelente referencia para aprender sobre la FFT, su implementación y sus aplicaciones en procesamiento de señales.
   - **Oppenheim, A.V., & Schafer, R.W. (2009).** *Discrete-Time Signal Processing*. Pearson.
     Este texto aborda la FFT en el contexto del procesamiento de señales discretas y su uso para análisis espectral.
3. **Artículos y recursos en línea**:
   - **Wikipedia - Modulación por Amplitud (AM)**:
     https://es.wikipedia.org/wiki/Modulaci%C3%B3n_por_amplitud
     Proporciona una descripción general y detallada de cómo funciona la modulación AM, su teoría básica y sus aplicaciones.
   - **Wikipedia - Transformada Rápida de Fourier (FFT)**:
     https://es.wikipedia.org/wiki/Transformada_rapida_de_Fourier
     Esta página es útil para comprender los principios matemáticos de la FFT y cómo se utiliza para descomponer señales en el dominio de la frecuencia.

Documentación de Python:

1. **Documentación oficial de NumPy** (para FFT y cálculo numérico):
   - https://numpy.org/doc/stable/reference/routines.fft.html
     Aquí se puede aprender sobre las funciones relacionadas con FFT en NumPy, como `rfft` y otras herramientas útiles para el análisis espectral.
2. **Documentación de Matplotlib** (para gráficos y visualización):
   - https://matplotlib.org/stable/users/pyplot_tutorial.html
     Esta es la documentación oficial de `matplotlib`, una librería para crear gráficos en Python, que se utiliza en el código para generar las representaciones gráficas en el dominio del tiempo y de la frecuencia.

Referencias Adicionales sobre Modulación AM y Detección Coherente:

1. **Sklar, B. (2001).** *Digital Communications: Fundamentals and Applications*. Prentice Hall.

Este texto cubre la modulación y demodulación de señales, incluyendo AM y la detección coherente, además de tratar temas de transmisión y recepción en sistemas de comunicación.

2. **Ziemer, R.E., & Tranter, W.H. (2002).** *Principles of Communications: Systems, Modulation, and Noise*. Pearson.
   Un excelente libro que aborda tanto la teoría como las aplicaciones de la modulación AM y otros métodos de modulación en sistemas de comunicaciones.