A partir de esta misma idea, podemos pensar que una imagen puede ser descrita por los puntos que N epiciclos van dibujando a medida que rotan en el tiempo. Podemos determinar su posicion espacial en diferentes instantes de tiempo a traves de la siguiente expresion matematica: pk[t] = ∑Cke^(iωkt) donde pk[t] es la coordenada del punto en el plano complejo para el instante de tiempo t, ωk = ω0k es la frecuencia del sistema y Ck es una costante compleja. 1. Realice el desarrollo matematico correspondiente para poder obtener, de forma simplificada, las coordenadas x e y correspondientes a los valores de pk

Para obtener las coordenadas x e y del punto en el plano complejo para el instante de tiempo t, podemos realizar el desarrollo matemático separando las partes real e imaginaria de pk[t].

La expresión general dada es:

pk[t] = ∑Cke^(iωkt)

Primero, vamos a separar la parte real (x) e imaginaria (y) del punto pk[t]. Para ello, necesitamos la fórmula de Euler:

e^(iθ) = cos(θ) + i \* sin(θ)

Ahora, podemos expresar pk[t] como:

pk[t] = ∑Cke^(iωkt) = ∑Ck \* (cos(ωkt) + i \* sin(ωkt))

Separando en partes reales (x) e imaginarias (y):

x = ∑Ck \* cos(ωkt)

y = ∑Ck \* sin(ωkt)

Estas son las expresiones simplificadas para las coordenadas x e y del punto pk[t] en el plano complejo para el instante de tiempo t, en términos de las constantes complejas Ck, las frecuencias ωk y el tiempo t. Con estas ecuaciones, podemos calcular las coordenadas espaciales del punto en diferentes momentos de tiempo a medida que los epiciclos rotan.

import numpy as np

import matplotlib.pyplot as plt

# Listado de radios R (longitudes de los epiciclos)

radios = [1.5, 2.0, 0.8]

# Listado de constantes complejas Ck

constantes\_cplx = [1 + 0.5j, # C1

0.7 - 0.3j, # C2

0.4 + 0.9j] # C3

# Frecuencia angular ω0

frecuencia\_omega = 2 \* np.pi

# Número de puntos en el tiempo para el gráfico

num\_puntos\_tiempo = 1000

# Calcula las coordenadas x e y en función del tiempo t

t = np.linspace(0, 2 \* np.pi, num\_puntos\_tiempo)

x = np.zeros(num\_puntos\_tiempo)

y = np.zeros(num\_puntos\_tiempo)

for k, R, C in zip(range(len(radios)), radios, constantes\_cplx):

x += np.real(C) \* np.cos(k \* frecuencia\_omega \* t)

y += np.real(C) \* np.sin(k \* frecuencia\_omega \* t)

# Grafica los epiciclos

plt.figure(figsize=(8, 6))

plt.plot(x, y)

plt.xlabel('x')

plt.ylabel('y')

plt.title('Gráfico de los epiciclos')

plt.grid()

plt.show()