Universidade Federal do Ceará
Centro de Tecnologia
Departamento de Engenharia de Teleinformática
Engenharia de Telecomunicações

Sinais e Representação por Série de Fourier

Apresentando o Fenômeno de Gibbs

Aluno	Lucas de Souza Abdalah - 385472
Professores	João César Moura Mota e André Lima Ferrer de Almeida
Disciplina	Sinais e Sistemas - TI0116

Fortaleza, 15 de dezembro de 2017

Conteúdo

1 Introdução

As senóides são as ondas mais simples e puras existentes, pois se originam da projeção sobre uma reta de um ponto girando em circulo, têm uma única frequência e, para finalizar a descrição, basta apontarmos amplitude das ondas.

Tendo em vista que utilizando a série de Fourier podemos representar diversos tipos de sinais, será apresentado como deduzir, implementar em Python 3.X uma série que aproxima uma Onda Quadrada e os efeitos associados.

2 Desenvolvimento

2.1 Séries de Fourier e Embasamento Teórico

Em 1822 foi publicado um trabalho que revolucionaria diversas áreas do conhecimento. Tendo pesquisado a fundo a relação das séries trigonométricas com fenômenos de propagação do calor, podemos dizer que Jean-Baptiste Joseph Fourier fez sua contribuição mais relevante a ciência ao estudar a condução térmica em uma placa metálica. Deduziu desse problema, primeiramente, a chamada Série de Fourier[1]. Logo abaixo, são apresentados os chamados coeficientes ${c_k}^2$ da série.

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{(jk\omega_0 t)} \tag{1}$$

$$c_k = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{(-jk\omega_0 t)} dt \tag{2}$$

A função $x(t)$ possui um período fundamental T e sua frequência fundamental é definida como $\omega_0 = \frac{2\pi}{T}$.

Apresentada a série de Fourier, as ferramentas e os parâmetros aplicados serão detalhados no decorrer do trabalho.

2.2 Problema Proposto

1. Faça um truncamento de M termos de uma resposta em frequência (onda quadrada) de um sistema discreto. Dê seus próprios parâmetros.

2.3 Dedução da Série de Fourier

De antemão, utilizaremos nesse trabalho um sinal no domínio do tempo, tendo em vista a propriedade da dualidade da transformada de Fourier.

Tome as seguintes transformadas:

$$\mathcal{F}\{x_1(t)\} = X_1(j\omega) \tag{3}$$
$$\mathcal{F}\{x_2(t)\} = X_2(j\omega) \tag{4}$$

Mostra-se que: se

$$x_2(t) = X_1(j\omega)|_{\omega=t} \tag{5}$$

Então

$$X_2(j\omega) = 2\pi x_1(t)|_{t=\omega} \tag{6}$$

No caso estudado, teremos que estimar a série de Fourier que modela o sinal abaixo.

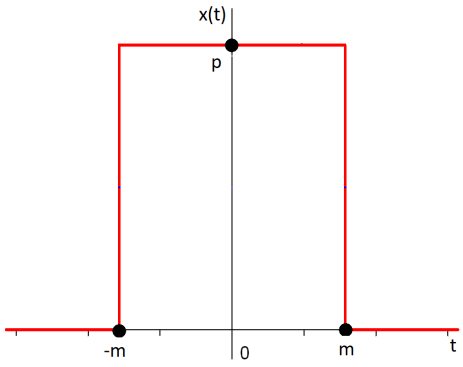

Figura 1: Onda quadrada de amplitude p e período fundamental $T = 4m$.

Observando o gráfico teremos: $x(t) = p$, quando $-m < t < m$, caso contrário, $x(t) = 0$.

Definiremos por conveniência que o período é $2a$, então $a = 2m$. Feitas as primeiras ponderações e definições necessárias podemos calcular os coeficientes utilizando a equação (2):

$$c_k = \frac{1}{2a} \int_{-a}^{a} p e^{(-jk\omega_0 t)} dt \tag{7}$$

Sendo p constante, sairá da integral.

$$c_k = \frac{p}{2a} \int_{-a}^{a} e^{(-jk\omega_0 t)} dt \tag{8}$$

Agora, teremos $c_0 = p$ e o termo geral para $k \neq 0$:

$$c_k = \frac{pj}{2k\omega_0}(e^{(-jk\omega_0 a)} - e^{(jk\omega_0 a)}) \tag{9}$$

Aplicando os coeficientes na série:

$$x(t) = \frac{c_0}{2} + \sum_{k=-\infty}^{-1} c_k e^{(jk\omega_0 t)} + \sum_{k=1}^{\infty} c_k e^{(jk\omega_0 t)} \tag{10}$$

$$x(t) = \frac{p}{2} + \frac{pj}{2\omega_0}\left\{ \sum_{k=-\infty}^{-1} \frac{1}{k}(e^{(-jk\omega_0 a)} - e^{(jk\omega_0 a)})e^{(jk\omega_0 t)} + \sum_{k=1}^{\infty} \frac{1}{k}(e^{(-jk\omega_0 a)} - e^{(jk\omega_0 a)})e^{(jk\omega_0 t)} \right\} \tag{11}$$

Obtendo, finalmente, a série de Fourier do sinal.

2.4 Implementação em Python 3.X

Primeiramente, importamos as bibliotecas que iremos trabalhar. Uma que define diversas funções matemáticas elementares como exponencial, funções trigonométricas, etc. Já a segunda, utiliza-se para plotagem dos gráficos.

```
import numpy as np
import matplotlib.pyplot as plt
```

Definimos nossa função *fsSW* que tem como parãmtetro o vetor t tempo, o real T período, *p_amp* é amplitude e *k_upperbound* será o limite supeior e o simétrico será o limite inferior.

```
def fsSW(t, T, p_amp, k_upperbound):
    #Define-se o periodo "T" e o intervalo "a" de que a onda
        permanece em amplitude determinada
    a = T/4

    #A relacao da frequencia fundamental com o periodo
    w_zero =2*np.pi/T

    #A constante que multiplica o somatorio (Tendo em vista
        economia de processamento)
    Q = (p_amp*1j)/(2*a*w_zero)

    #Iniciando o acumulador da serie de fourier
    fsSW = 0

    #Aproximacao do somatorio da funcao calculada
    for k in range((-1)*k_upperbound, k_upperbound, 1):
        #Calcula o termo de k=0
        if k == 0:

            fsSW = fsSW + 0.5*p_amp
        #Calcula os termos restantes
        else:
            fsSW = fsSW + 0.5*(Q/k)*((np.exp(1j*k*w_zero*(-a)))
                - np.exp(1j*k*w_zero*(a)))*np.exp(1j*k*w_zero*t)

    #Retorna a soma final
    return(fsSW)
```

Criando vetores x para iterar e plotar a função:

```
#Gerando variaveis no eixo do tempo
x_1 = np.linspace(-3*np.pi, 3*np.pi, 50)
x_2 = np.linspace(-3*np.pi, 3*np.pi, 100)
x_3 = np.linspace(-3*np.pi, 3*np.pi, 500)
x_4 = np.linspace(-3*np.pi, 3*np.pi, 1000)
x_5 = np.linspace(-3*np.pi, 3*np.pi, 10000)
```

Finalmente, a função sendo chamada:

```
#Atribuindo as variaveis do a ao d o retorno da funcao com os
    parametros indicados
a = fsSW(x_1, 2*np.pi, 1, 2)
b = fsSW(x_2, 2*np.pi, 1, 10)
c = fsSW(x_3, 2*np.pi, 1, 50)
d = fsSW(x_4, 2*np.pi, 1, 1000)
e = fsSW(x_5, 2*np.pi, 1, 10000)
```

E agora plotando os gráficos, comparados com x_4.

```
plt.figure('1'), plt.grid('on'), plt.plot(x_1, a , 'r')
plt.plot(x_4, d, 'b')

plt.figure('2'), plt.grid('on'), plt.plot(x_2, b, 'r')
plt.plot(x_4, d, 'b')

plt.figure('3'), plt.grid('on'), plt.plot(x_3, c, 'r')
plt.plot(x_4, d, 'b')

plt.figure('4'), plt.grid('on'), plt.plot(x_5, e, 'r')
plt.plot(x_4, d, 'b')
```

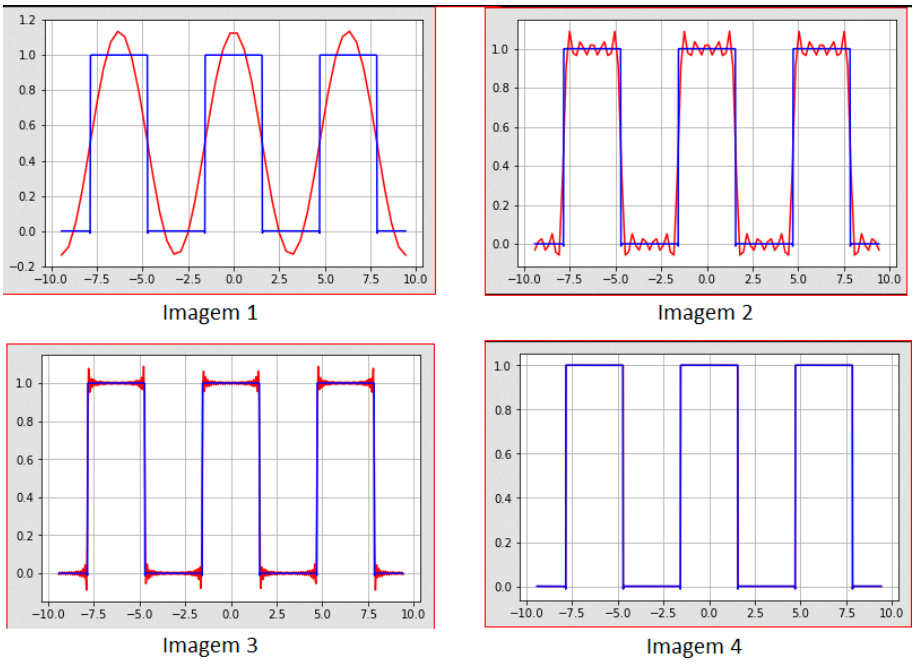

Figura 2: Em vermelho, as somas parciais de n parcelas exponeciais complexas, com n = 2, n = 10, n = 50, n = 1.000. Já as azuis mantém-se constante em n = 1.000, exceto na imagem 4, com n = 10.000.

3 Análise dos Resultados

3.1 Modelagem

Quando modelamos um sinal por série de Fourier teoricamente, teremos uma estimação de como será o sinal com uma soma de infinitos termos.

Entretanto, os recursos de memória e processamento dos sistemas são finitos, por consequência não podemos efetuar essa "soma infinita"de forma a obter o sinal exato.

Mas, se não podemos fazer tal operação, como estimar esse sinal computacionalmente? Os sistemas e os *softwares* atuais, têm um "absurdo"poder de processamento e capacidade de efetuar milhares de cálculos em poucos segundos. Então, em vez de tentarmos uma soma infinita, somamos um número muito grande de parcelas, para nos aproximarmos cada vez mais do sinal calculado teoricamenteo, evidentemente respeitando o limite de dados que o computador consegue processar.

Fazendo o número de somas das parcelas, vemos o sinal convergindo para o estimado, porém com algumas pequenas variações.

3.2 Fenômeno de Gibbs

Sabendo que a série de Fourier aproxima um função f, ela converge pontualmente onde não há descontinuidade, porém em regiões de descontinuidade existe uma pertubação, motivada pela própria descontinuidade. O resultado são alguns saltos e oscilações "inesperadas"e são essas variações que chamamos de Fenômeno de Gibbs. Quando aumentamos o n, teremos um oscilação mais contida na região da descontinuidade e um sinal "mais comportado", porém o fenômeno ocorrerá quão grande for n, sendo $n < \infty$, veremos o efeito, por menor que seja.

4 Referências

[1] Oppenheim, Alan V. – Signals and systems, 2nd Edition;

[2] Santos, Fabiano J. – Introdução às Séries de Fourier;
matematica.pucminas.br/profs/web_fabiano/calculo4/sf.pdf

[3] Teorema de Fourier – Forma de Onda, Espectro e Espectrograma;
qsl.net/py4zbz/teoria/espectro.htm

[4] Wikipedia – Fourier Series;
https://en.wikipedia.org/wiki/Fourier_series

[5] Souza, Felippe – Transformadas de Fourier;
webx.ubi.pt/ felippe/texts2/an_sinais_cap8.pdf

[6] Nazari, Luiz F. – Trabalho de Conclusão de Curso: Séries de Fourier e Fenômeno de Gibbs;