

# Analogia entre vetores e sinais

V. C. Parro e-mail: [vparro@ieee.org](mailto:vparro@ieee.org)

17 de fevereiro de 2021

## Objetivos Analogia entre vetores e sinais - Série de fourier

Esta atividade tem como objetivo permitir ao aluno que visualize a formação da base "ortogonal de" de Fourier e como esta base permite a descrição de um sinal em um espaço N-dimensional.

Na aula de teoria definimos como referência o sinal  $g_n(t)$  e que a respectiva projeção de um sinal  $g(t)$  pode ser calculada pela equação 1.

$$c_n = \frac{\int_{T_o} g(t)g_n(t)dt}{\int_{T_o} g_n(t)^2 dt} \quad \} = \quad \forall o. \quad (1)$$

### Proposta de análise.

1. Um sinal periódico  $g(t)$  é definido pela equação 2 no intervalo  $0 \leq t \leq 1$  que representa exatamente a equação de um período deste sinal que equivale a  $T_0 = 1s$ .

$$g(t) = e^{-t} \quad (2)$$

2. Utilizando como base a função descrita pela Equação 2 e a função a ser decomposta pela Equação 3.

$$g_n(t) = e^{-jn\omega_o t} \quad (3)$$

3. Determine o sinal resultante da primeira projeção -  $p_1(t)$  determinando os coeficientes  $c_1$  e  $c_{-1}$ . Verifique o resíduo -  $r_1(t)$  que pode ser calculado pela Equação 4. Compare a projeção  $p_1(t)$  e o resíduo  $r_1(t)$  com o sinal original -  $g(t)$ .

$$\begin{aligned} \bullet c_1 &= \frac{\int_{T_o} g(t)g_1(t)dt}{\int_{T_o} g_1(t)^2 dt} \\ \bullet c_{-1} &= \frac{\int_{T_o} g(t)g_{-1}(t)dt}{\int_{T_o} g_{-1}(t)^2 dt} \\ p_1(t) &= c_1 e^{-j\omega_o t} + c_{-1} e^{j\omega_o t} \\ r_1(t) &= g(t) - p_1(t) \end{aligned} \quad (4)$$

4. Determine o sinal resultante da segunda projeção -  $p_2(t)$  determinando os coeficientes  $c_2$  e  $c_{-2}$ . Verifique o resíduo -  $r_2(t)$  que pode ser calculado pela Equação 5. Compare a projeção  $p_2(t)$  e o resíduo  $r_2(t)$  com o sinal original -  $g(t)$ . Determine os coeficientes  $c_2$  e  $c_{-2}$  pela Equação 6 e compare com os valores previamente calculados, o que você conclui?

$$c_2 = \frac{\int_{T_o} g(t)g_2(t)dt}{\int_{T_o} g_2(t)^2dt}$$

$$c_{-2} = \frac{\int_{T_o} g(t)g_{-2}(t)dt}{\int_{T_o} g_{-2}(t)^2dt} \quad (5)$$

$$p_2(t) = c_2 e^{-2j\omega_o t} + c_{-2} e^{2j\omega_o t}$$

$$r_2(t) = g(t) - (p_1(t) + p_2(t))$$

$$c_2 = \frac{\int_{T_o} r_1(t)g_2(t)dt}{\int_{T_o} g_2(t)^2dt}$$

$$c_{-2} = \frac{\int_{T_o} r_1(t)g_{-2}(t)dt}{\int_{T_o} g_{-2}(t)^2dt} \quad (6)$$

5. Crie uma estrutura recorrente que permita determinar a projeção -  $p_n(t)$ , determinando os coeficientes  $c_n$  e  $c_{-n}$ . Verifique o resíduo -  $r_n(t)$  que pode ser calculado pela Equação 7. Compare a projeção  $p_n(t)$  e o resíduo  $r_n(t)$  com o sinal original -  $g(t)$ .

$$c_n = \frac{\int_{T_o} g(t)g_n(t)dt}{\int_{T_o} g_n(t)^2dt}$$

$$c_{-n} = \frac{\int_{T_o} g(t)g_{-n}(t)dt}{\int_{T_o} g_{-n}(t)^2dt} \quad (7)$$

$$p_n(t) = c_n e^{-nj\omega_o t} + c_{-n} e^{nj\omega_o t}$$

$$r_n(t) = g(t) - (p_1(t) + p_2(t) + \dots + p_n(t))$$

## 1 Estudo de um caso

Vamos eleger como base para o estudo inicial o sinal pulsado indicado na figura 1. menor

Para medirmos o grau de semelhança entre o sinal  $g(t)$  e um sinal harmônico de mesma frequência angular  $\omega_0 = \frac{\pi}{2} \frac{rd}{s}$  utilizaremos como variável de ajuste a amplitude deste sinal  $c$ . Para estabelecer um critério sobre o grau de semelhança elaboramos a equação 21 que determina o erro quadrático entre o sinal  $g(t)$  e o sinal harmônico.

$$Erro = \int_{T_o} (g(t) - c \sin(\frac{\pi}{2}t))^2 dt \quad (8)$$

PARA A

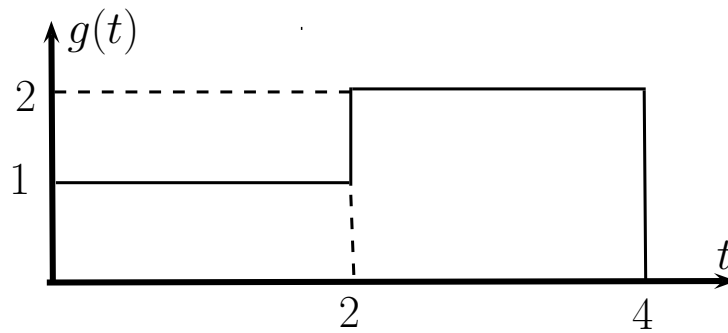


Figura 1: Sinal  $g(t)$

### 1.1 Qual o valor de $c$ que minimiza o erro?

Vamos utilizar a abordagem de determinação de máximos e mínimos estudado em cálculo 1.

$$\frac{d \text{Erro}}{dc} = \frac{d \int_{T_o} (g(t) - c \sin(\frac{\pi}{2}t)^2 dt}{dc} = 0 \quad (9)$$

### 1.2 Como resolver este problema usando Matlab?

Primeiro vamos definir a função que desejamos minimizar:

Matlab - função a ser minimizada

```
function [J] = Erro(c,n)
% função mérito

syms t % indica que a variável t será tratada como simbólica

% determina o erro instantâneo
erro1 = @(c,n,t) (1-(-c*sin(t*n*pi/2))).^2;
erro2 = @(c,n,t) (2-(-c*sin(t*n*pi/2))).^2;

% faz a integral do erro
J = int(erro1(c,n,t),t,0,2)+int(erro2(c,n,t),t,2,4);

% converte a expressão obtida em valor numérico
J = eval(J);

end
```

Após definida a função mérito que determina o valor do **Erro** de acordo com a equação 21, vamos construir o código que implementa a solução proposta ppela equação 22.

## Matlab

```
%% Análise de otimização
```

```
for k=1:Total  
    n=k;  
    x(k) = fminsearch(@(c) Erro(c,n),0)  
end
```

```
figure(11)
```

```
stem(n,abs(x),'LineWidth',2)  
grid minor; title('Entrada'); grid minor;xlabel('Frequência')
```

O resultado quando escolhemos  $Total = 1$  corresponde a  $c = 0.6366$ . Podemos comparar ambas as funções no mesmo gráfico 1. Indica também a potência obtida na frequência de  $0.25Hz$  é de:

$$P_g = \frac{0.6366^2}{2} = 0.2026W \quad (10)$$

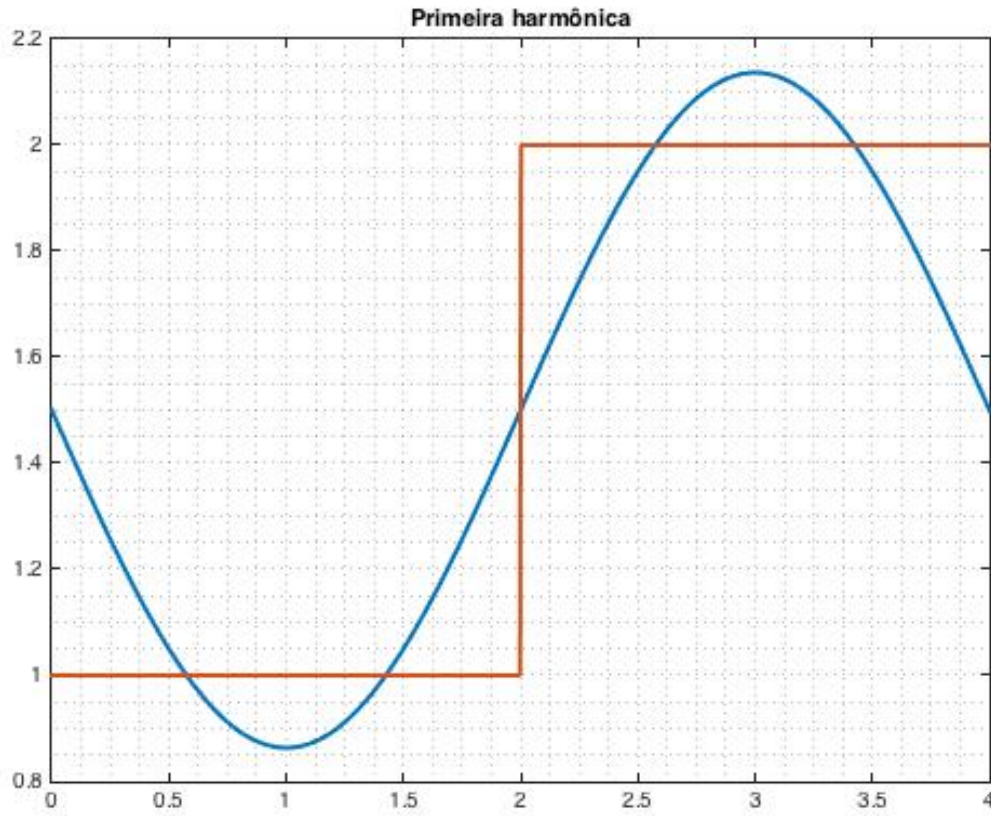


Figura 2: Comparação entre  $g(t)$  e a primeira harmônica resultante.

Quando optamos por  $Total = 10$  podemos observar quais outras harmônicas fazem parte da composição do sinal  $g(t)$ . Há um aspecto interessante que as harmônicas que correspondem aos valores pares de  $n$  são todas nulas, indicando que o sinal  $g(t)$  não tem "projeções não nulas" nestas frequências.

**O que acontece se verificamos as "projeções" em sinais harmônicos do tipo cossenos?** Observamos que todos os valores de  $c(n)$  são nulos indicando que o sinal  $g(t)$  não tem projeções em harmônicas do tipo cosseno.

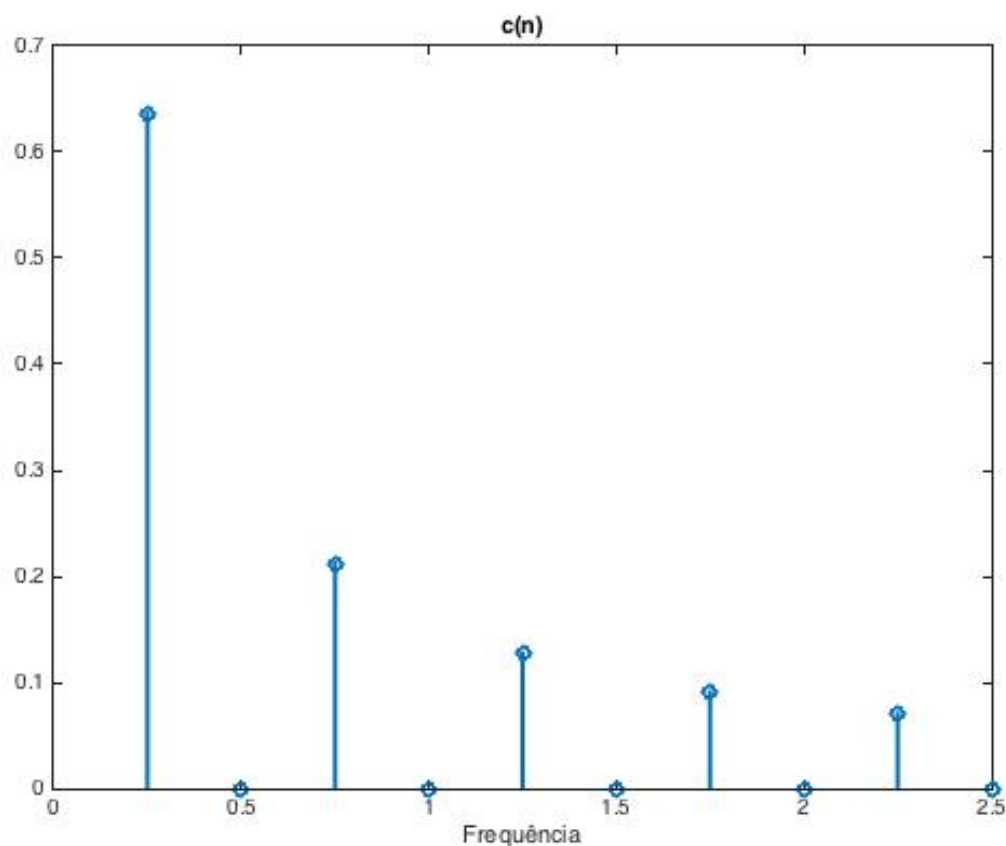


Figura 3: Coeficientes considerando  $Total = 10$ .

## 2 Formalizando o problema

Em aula estudamos que um sinal periódico  $g(t)$  pode ser decomposto em uma soma ponderada de sinais como indicado na equação 11 e chamamos este processo de **Síntese**.

$$g(t) = c_1x_1(t) + c_2x_2(t) + c_3x_3(t) + c_4x_4(t) + c_5x_5(t) + \dots \quad (11)$$

em sua forma compacta temos 24.

$$g(t) = \sum_{n=0}^N c_n x_n(t) \quad (12)$$

Uma questão importante é como escolher o conjunto de sinais que compõem a base sobre a qual o sinal será analisado. Usando uma analogia com vetores, estes sinais tem de ser ortogonais e respeitar a condição imposta em 13.

$$\int_{T_o} x_m(t)x_n(t)dt = 0 \quad (13)$$

Para o caso onde a base atenda a equação 13 os coeficientes  $c_n$  podem ser calculados utilizando 14 que é a solução analítica da equação 22 que chamamos de **Análise**.

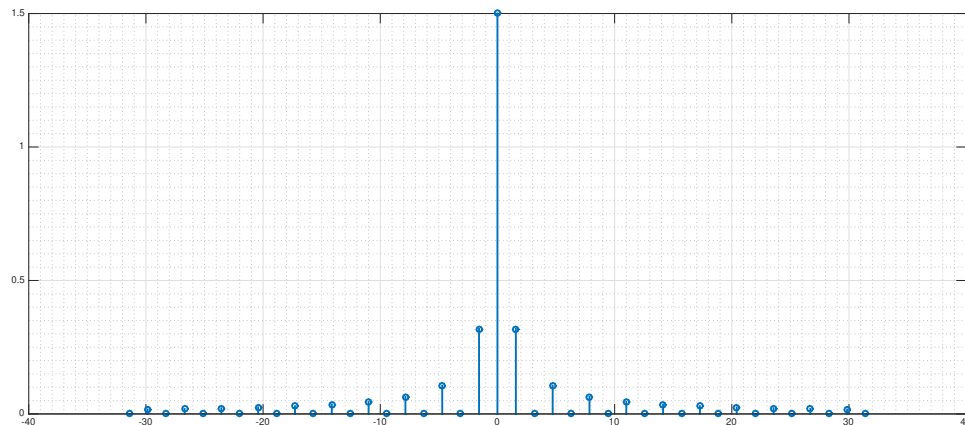


Figura 4: Distribuição dos coeficientes em função da frequência para o problema 1.

$$c_n = \frac{\int_{T_o} g(t)x_n(t)dt}{\int_{T_o} x_n^2(t)dt} \quad (14)$$

### Análise de Fourier

Considerando o sinal  $g(t)$  ilustrado na figura 1 representando um período completo, determinar as projeções do sinal considerando uma base exponencial 17.

$$x_n(t) = e^{-jn\omega_o t} \quad (15)$$

Resolvendo o numerador da equação 14 considerando  $\omega_o = \frac{2\pi}{4} = \frac{\pi}{2}$ :

$$N = \int_0^2 e^{-jn\frac{\pi}{2}t} dt + \int_2^4 2e^{-jn\frac{\pi}{2}t} dt \quad (16)$$

Resolvendo o denominador:

$$D = \int_0^4 e^{-jn\frac{\pi}{2}t} e^{jn\frac{\pi}{2}t} dt \quad (17)$$

## Matlab

```
%% solução do problema 01

clc; % limpa a tela
clear; % limpa as variáveis
close all; % fecha as figuras ativas
syms t; % configura modo de simulação literal

%% Configura variáveis de entrada

To=4; % período do sinal
wo=2*pi/To; % frequência fundamental
N=20;
n=linspace(-20,20,41); % n indexador de harmônica
w=n.*wo; % vetor de frequência

% A amplitude do sinal

%% Configura o numerador da equação (4)

xn=@(wo,A,n,t) A*exp(-j*n*wo*t);
N= int(xn(wo,1,n,t),t,0,2)+int(xn(wo,2,n,t),t,2,4);
resN=eval(N); % coloca N na forma numérica

%% Configura o denominador da equação (4)

D= int(xn(wo,1,n,t).*xn(-wo,1,n,t),t,0,4);
resD=eval(D); % coloca D na forma numérica

% determinando Dn

Dn=resN./resD;

%% Visualizando os resultados

stem(w,abs(Dn),'LineWidth',2)
```

## 2.1 O que significam os coeficientes determinados? $D_n$

Calculando o termo para  $n = 0$  temos  $D_0 = 1.5$  que indica que o sinal  $g(t)$  tem valor médio de 1.5 ( a unidade neste caso está relacionada a grandeza em análise). Repetindo processo para  $n = 1$  temos  $D_1 = j0.3183$ , qual seu significado? Na verdade podemos dizer que o sinal  $g(t)$  pode ser aproximado por 18.

$$g(t) = 1.5 + j0.3183e^{j\frac{\pi}{2}t} \quad (18)$$



Obviamente não é possível representar neste caso o sinal  $g(t)$  com uma função complexa em  $t$ . Se formos atentos, quando estamos no domínio  $\omega$  devemos considerar a condição quando  $n = -1$  que representa a mesma frequência. Desta forma aproximamos por 19.

$$g_1(t) = 1.5 + j0.3183e^{j\frac{\pi}{2}t} - j0.3183e^{-j\frac{\pi}{2}t} \quad (19)$$

$$g_1(t) = 1.5 + 0.3183[-2\sin(\frac{\pi}{2}t)] \quad (20)$$

Manipulando a equação 19 podemos obter 23. Neste ponto vamos voltar a proposta inicial: o valor encontrado 0.6366 é o que mais aproxima o sinal  $g(t)$  de um sinal harmônico  $x_1(t) = 0.6366\sin(\frac{\pi}{2}t)$ . Como podemos verificar suando o matlab? A medida do erro foi estabelecida como:

$$Erro = \int_{T_o} (g(t) - c \sin(\frac{\pi}{2}t)^2) dt \quad (21)$$

Neste ponto o Matlab pode ajudar bastante na simulação de um número elevado de  $n$  para uma análise mais detalhada.

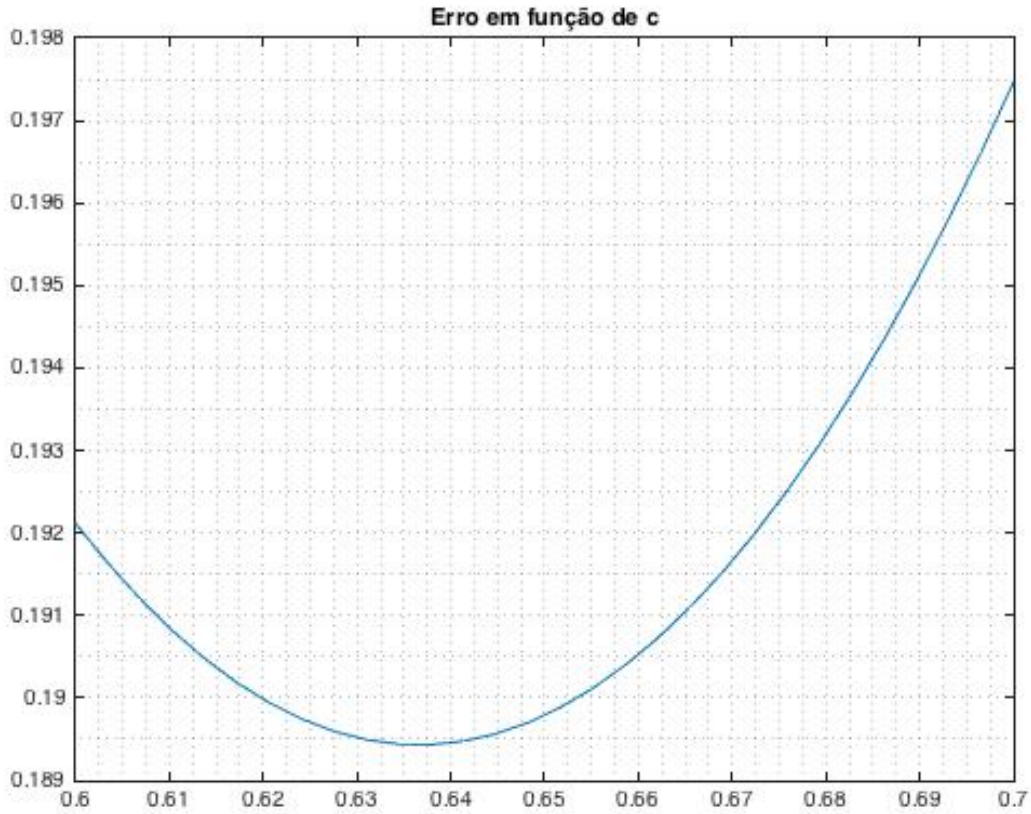


Figura 5: Erro em função do coeficiente  $c$ .

### Análise do erro

```
%% Análise de erro conforme aumenta-se a ordem de Fourier

syms t

erro1 = @(c,n) (1-(1.5-c*sin(t*n*pi/2))).^2 % erro no nível baixo
erro2 = @(c,n) (2-(1.5-c*sin(t*n*pi/2))).^2 % erro no nível alto

c=linspace(0.6,0.7,30); % coeficiente c

for k=1:30
    E(k)= eval(int(erro1(c(k),1),t,0,2)+int(erro2(c(k),1),t,2,4));
end

figure(4)

set(findall(gcf,'Type','line'),'LineWidth',2);
set(gca,'FontSize',14,'LineWidth',1.5);

plot(c,E); grid minor; title('Erro em função de c')
```

Concluimos que o coeficiente encontrado minimiza o erro definido por 21 quando a frequência do sinal em análise  $g(t)$  é a mesma do sinal comparativo  $x_1(t)$ . Vamos formular o problema para  $n = 3$ . Temos que  $D_3 = j0.1061$  e  $D_{-3} = -j0.1061$  que resulta em um sinal do tipo 22.

$$g_3(t) = g_1(t) + j0.1061e^{j3\frac{\pi}{2}t} - j0.1061e^{-j3\frac{\pi}{2}t} \quad (22)$$

$$g_3(t) = g_1(t) + 0.1061[-2\sin(3\frac{\pi}{2}t)] \quad (23)$$

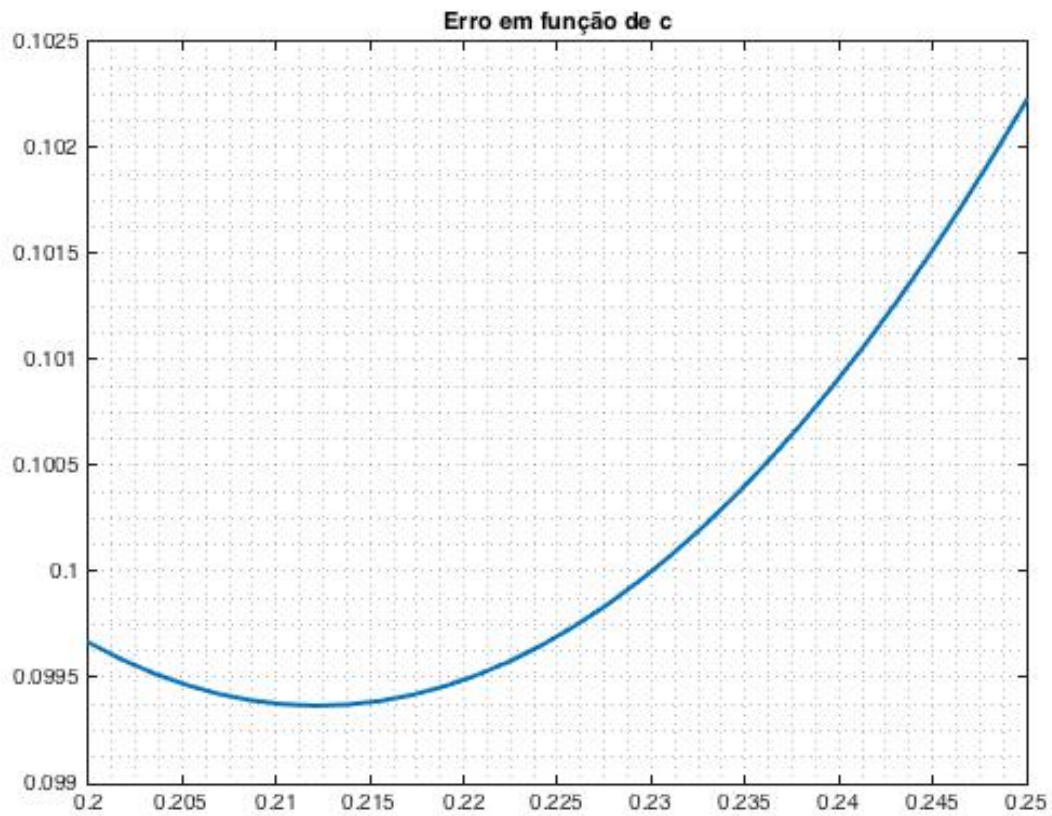


Figura 6: Erro em função do coeficiente  $c$ .

#### Análise do erro

```
%% Analisando para n=3

syms t

erro1 = @(c,n) (1-(1.5-0.6366*sin(t*pi/2)-c*sin(3*t*pi/2))).^2
erro2 = @(c,n) (2-(1.5-0.6366*sin(t*pi/2)-c*sin(3*t*pi/2))).^2

c=linspace(0.2,0.25,30);

for k=1:30
    E(k)= eval(int(erro1(c(k),1),t,0,2)+int(erro2(c(k),1),t,2,4));
end

figure(6)

set(findall(gcf,'Type','line'),'LineWidth',2);
set(gca,'FontSize',14,'LineWidth',1.5);

plot(c,E,'LineWidth',2); grid minor; title('Erro em função de c')
```

## Síntese

Neste ponto o Matlab pode ajudar bastante na simulação de um número elevado de  $n$  para uma análise mais detalhada.

$$g(t) = \sum_{n=0}^N c_n x_n(t) \quad (24)$$

## Matlab - complementando o código anterior

```
%% Fazendo a síntese dos sinais
% Análise de primeira harmônica

M=1000;
gt=Dn(Total+1);          % inciai sempre com o valor médio do sinal
t= linspace(0,To,M);     % cria o vetor tempo com 1000 pontos em 1
                          % período

for k=1:1
    gt = gt + Dn(Total+1+k)*exp(k*j*wo*t) + Dn(Total+1-k)*exp(-k*j*wo*t);
end

figure(2)
plot(t,gt,'LineWidth',2), title('Primeira harmônica'); grid minor

%% Criando o sinal g(t) para compararmos

gtr=[ones(1,M/2) 2*ones(1,M/2)];
hold; plot(t,gtr,'LineWidth',2)

%% Análise para a enésima harmônica

Nx=10;                   % Nx representa qual harmônica
                          % estamos interessados em analisar

for k=2:Nx
    gt = gt + Dn(Total+1+k)*exp(k*j*wo*t) + Dn(Total+1-k)*exp(-k*j*wo*t);
end

figure(3)

plot(t,gt,'LineWidth',2), title('Nx=10'); grid minor
hold; plot(t,gtr,'LineWidth',2)
```

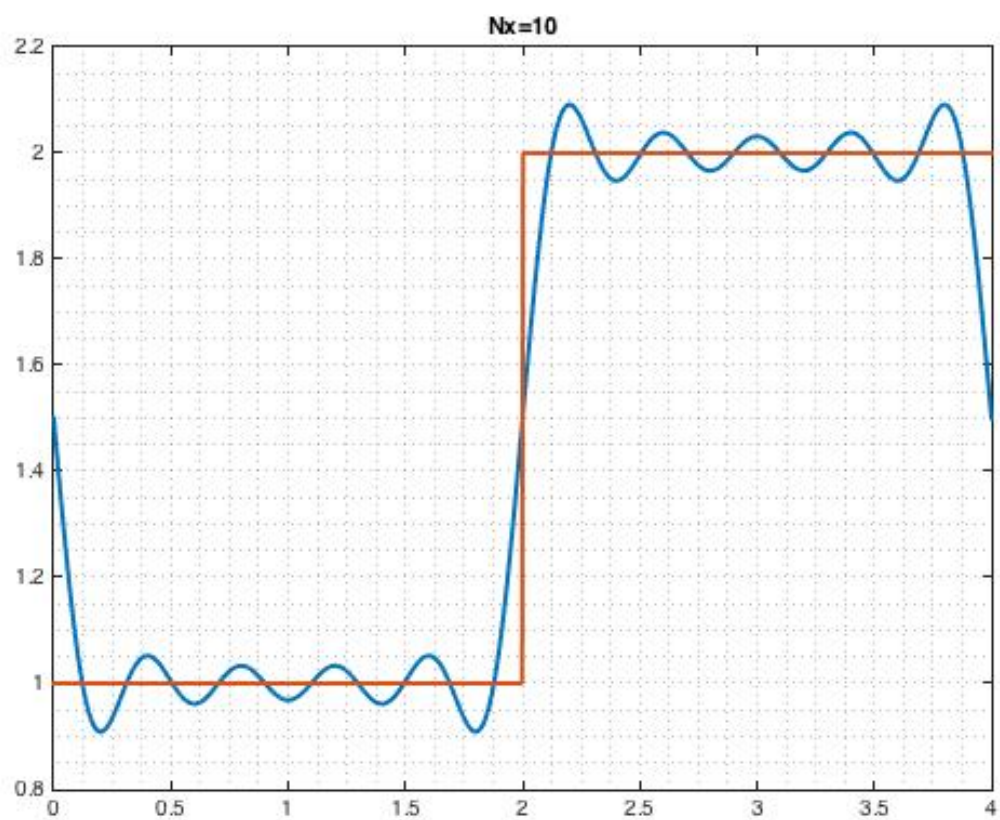


Figura 7: Comparação entre  $g(t)$  e a somatória referente a  $N_x = 10$ .