

Trabalho DTFT

August 16, 2017

1 Universidade Federal do Rio Grande do Norte

1.1 Processamento Digital de Sinais

1.1.1 Transformada de Fourier de Tempo Discreto – MATLAB

Aluno: Lucas Bezerra Dantas Saraiva

Professor: Luiz Felipe de Queiroz Silveira

Data de entrega : 16/08/2017

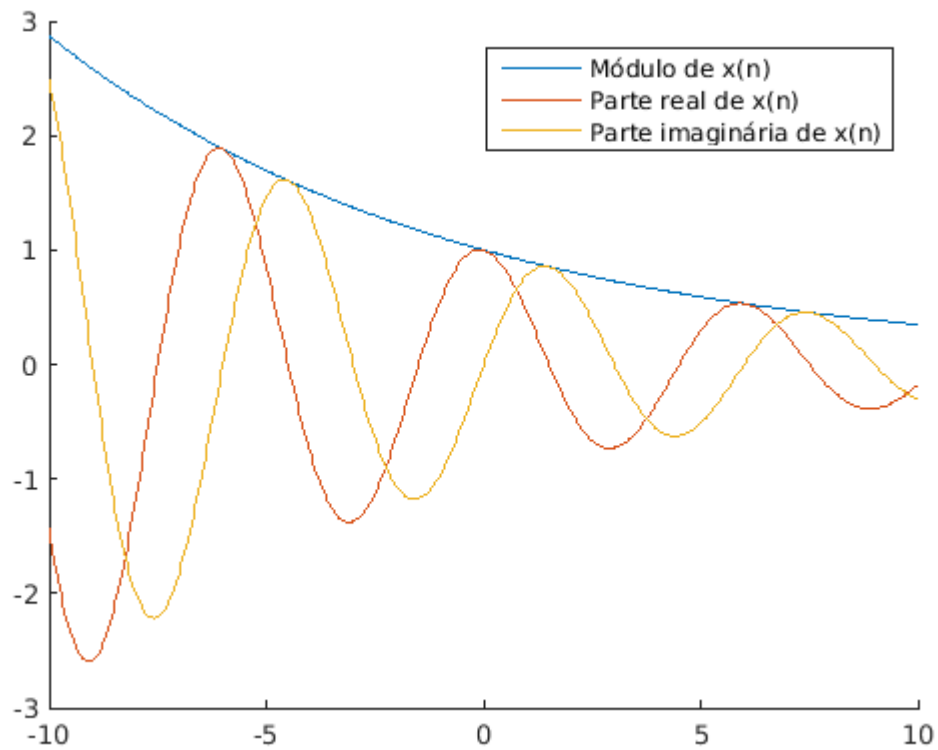
Nesse experimento, avaliaremos a resposta do sinal $x(n)$ em frequência com a DTFT. A equação que o caracteriza é dada por

$$x(n) = \left(0.9 \cdot e^{\frac{j\pi}{3}}\right)^n$$

Para $(0 \leq n \leq 10)$.

A proposta era montar a DTFT deste sinal no Matlab, montando as equações analiticamente e as projetando via `plot()`. O sinal $x(n)$ não possui quaisquer simetria (Par ou Ímpar) a primeira vista:

```
In [11]: clc;clear all;close all;
          %% Parâmetros dado pelo professor:
          n=-10:0.04:10;                      % Montagem do eixo n
          x=(0.9*exp(1j*pi/3)).^n;             % Montagem do sinal
          hold on
          plot(n,abs(x))
          plot(n,real(x))
          plot(n,imag(x))
          legend('Módulo de x(n)', 'Parte real de x(n)', 'Parte imaginária de x(n)')
```



Portanto, por ser um sinal complexo e sem simetria, não sua transformada não terá simetria. Abaixo algumas propriedades:

$f(n)$	$F(e^{j\omega})$
Par	Par
Ímpar	Ímpar
Real e par	Real e par
Real e ímpar	Imaginária e ímpar
Imaginária e par	Imaginária e par
Complexa e par	Complexa e par
Complexa e ímpar	Complexa e ímpar
Complexa e ímpar	Complexa e ímpar
Complexa e ímpar	Complexa e ímpar
Complexa e ímpar	Complexa e ímpar

A equação da transformada consiste em:

$$X(e^{j\omega}) = \sum_{-\infty}^{\infty} x[n]e^{-j\omega n}$$

Com $\omega = \frac{k\pi}{M}$ e M amostras.

Como o sinal de entrada não é simétrico, sua transformada não deve ter simetria. Faremos então o teste:

```
In [ ]: clc;clear all;close all;
        %% Parâmetros dado pelo professor:
        n=0:0.04:10;                % Montagem do eixo n
        x=(0.9*exp(1j*pi/3)).^n;      % Montagem do sinal
        M=500;                       % Número de amostras
        k=-M:M-1;                    % Montagem do eixo k
        w=pi*k/M;                    % Ômega em função de k
        %% Função
        X=dtft(x,n,w)                % Chama função DTFT()

In [4]: function [X] = dtft(x,n,w)
        % X = valores da TFTD calculados em w pontos de frequências
        % x = sequência de duração finita sobre n
        % n = vetor de índices de tempo
        % w = vetor de índices de frequência
        M=floor(length(w));          % Número de amostras do vetor ômega
        k=w*(M/pi);                  % Reconstrução do vetor k
        X=x*exp((-1j*pi/M).*(n'*k)); % Equação da DTFT

        %% Plotting
        figure ()
        hold on
        title('Espectro de X(e^j^w)')
        plot(w/pi,abs(X));
        plot(w/pi,real(X));
        plot(w/pi,imag(X));
        grid;
        legend('Magnitude de X(e^j^w)','Parte Real de X(e^j^w)','Parte Imaginária de X(e^j^w)');
        legend('Location','Northwest')
        ylabel('Espectro de x(n)')
        xlabel('\pi')
```

