

Processamento de Sinais em Tempo Discreto

- Prof. Dr. Samuel Lourenço Nogueira



Proposta opcional para NF

- Média: **E**xercícios e atividades **P**ara **C**asa (EPC)
- Média: **P**rovinhas no término de cada módulo (P)
- ~~Projeto Final da disciplina (PF)~~ (opcional)

$$\text{Nota Final} = 0.2 * \text{EPC} + 0.5 * \text{P} + 0.3 * \text{PF}$$

Caso opte por não fazer:

$$\text{Nota Final} = 0.35 * \text{EPC} + 0.65 * \text{P}$$

Projeto final, entrega:

- 10/12
 - Short Paper 6 páginas, com até 3 integrantes
 - Vídeo único de até 10 minutos apresentando desenvolvimento e resultados

5 e 10

- Aulas passadas (Sobre Fourier)
 - Filtro de Resposta Finita ao Impulso (Filtro FIR)
 - Resposta em Frequência de Filtro FIR

Conteúdo Programático

Projeto de um filtro FIR

- Fundamentos
 - Exemplo completo
 - Resumo dos passos
- } basicamente os passos

Projeto de Filtro FIR

Fundamentos

Lembrando que a saída de um filtro FIR é dada pela convolução entre um sinal e os coeficientes do filtro:

$$y[n] = \sum_{k=0}^K b[k]x[n-k]$$

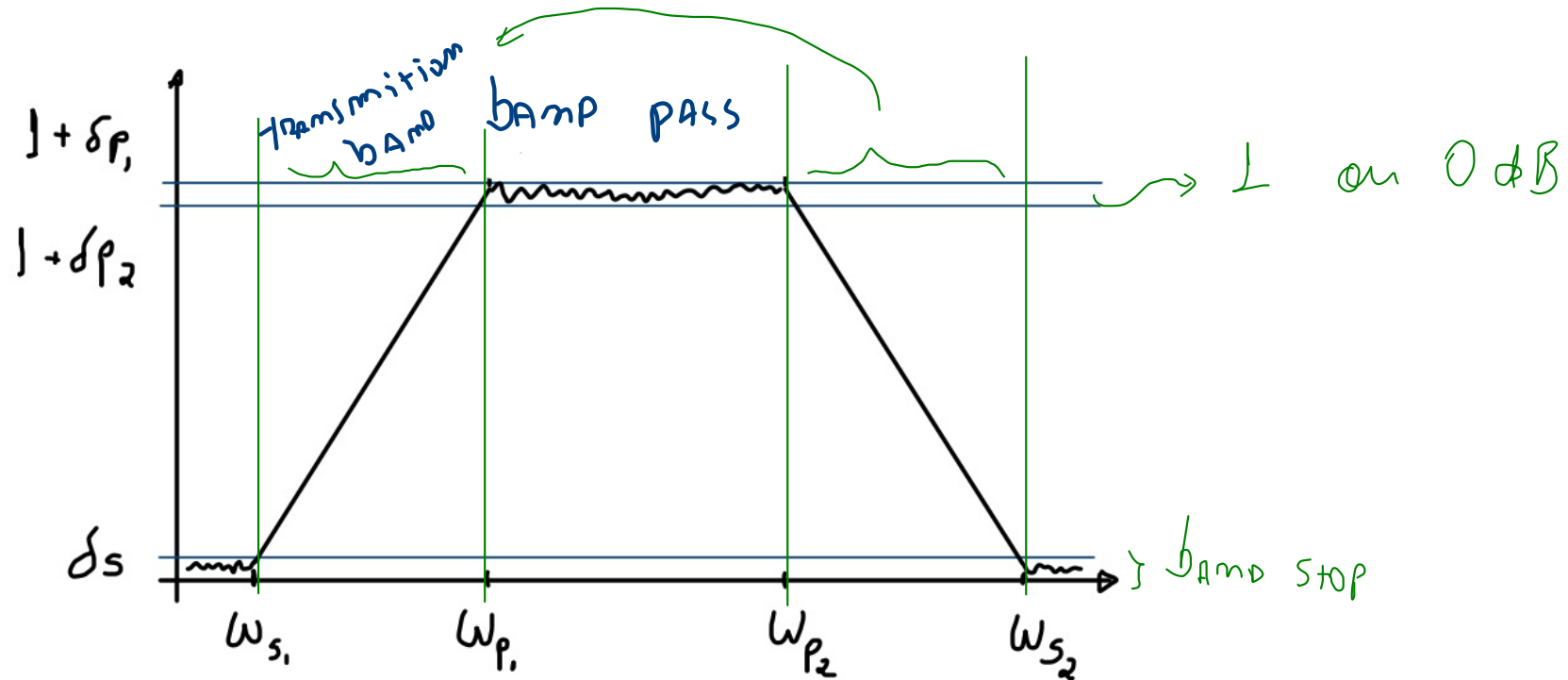
↳ preciso saber os $b[k]$

- O Projeto de um filtro FIR consiste na determinação do conjunto de coeficientes $b[k]$ que compõe o mesmo. Neste curso, utilizaremos o método de janelamento, no entanto, existem outras metodologias tais como: amostragem em frequência, equi-ripple ótimo, entre outras.

Projeto de Filtro FIR

Fundamentos

As especificações de um filtro podem ser vistas na ilustração abaixo:



Projeto de Filtro FIR

Fundamentos

- Método do Janelamento:

Como as diferentes janelas respondem no domínio da frequência, quando são sujeitas ao Impulso Unitário

Neste tipo utilizaremos três tipos de janela:

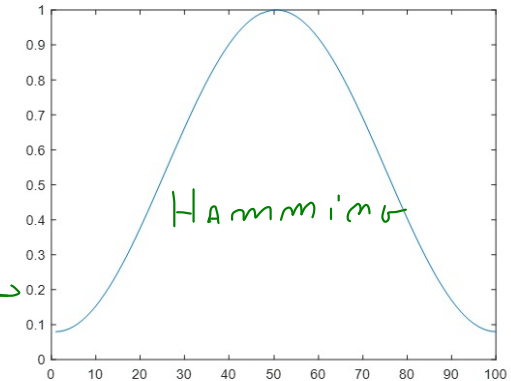
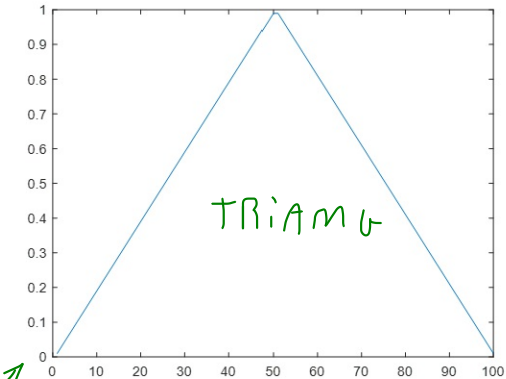
- Unitária = [1 1 1 1 1]
- Triangular = [1/3 2/3 1 2/3 1/3]
- Hamming = [0.08 0.54 1 0.54 0.08]

$$W[m] = 1 - \left| \frac{m - N/2}{N} \right|$$
$$W[m] = \sin^2(\pi m / N)$$

Utilizando a função "windows" no Matlab/Octave você encontrará diversos outros tipos.
>> doc window

Verifique:

```
>> plot(window(@triang,100));  
>> plot(window(@hamming,100));
```



Projeto de Filtro FIR

Fundamentos

Neste exemplo utilizaremos o script abaixo, para apresentar o comportamento na frequência de um sinal filtrado por coeficientes em formato de janela.

>> Topico7Exemplo1.m

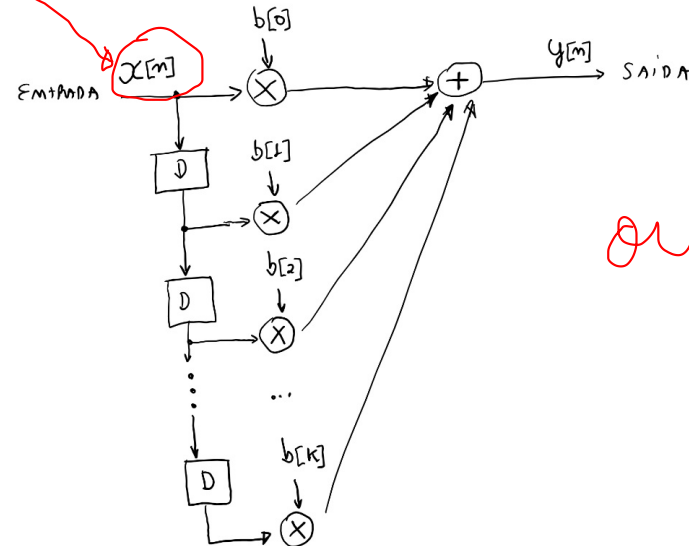
Verifique:

- Foi definida uma função impulso unitário
- Sendo a mesma filtrada com as três janelas apresentadas no início, através da convolução com os coeficientes:

- Unitária = [1 1 1 1 1]
- Triangular = [1/3 2/3 1 2/3 1/3]
- Hamming = [0.08 0.54 1 0.54 0.08]

$b[0]$

$b[4]$



ou

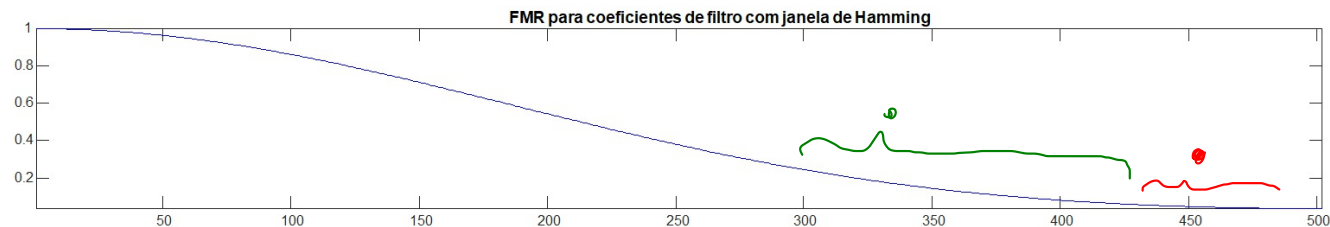
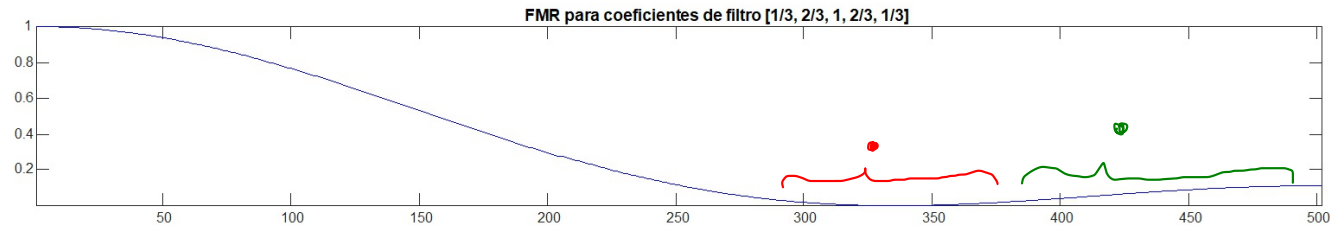
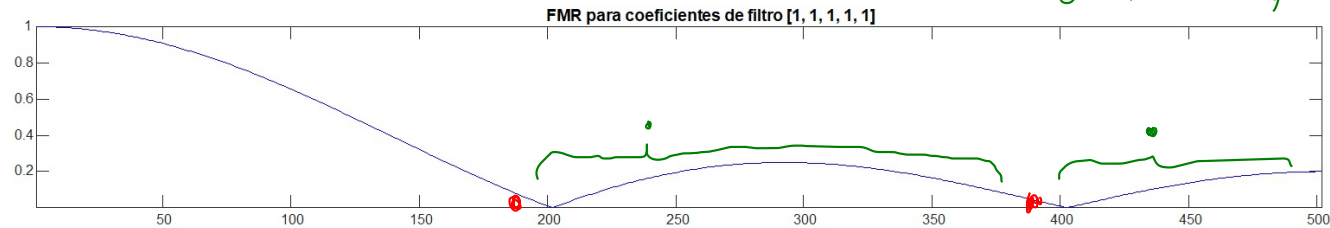
$$y[n] = \sum_{k=0}^K b[k]x[n-k]$$

Projeto de Filtro FIR

Fundamentos

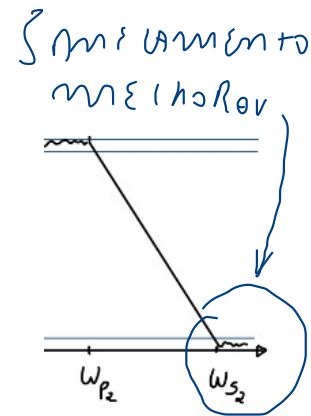
Em cada um dos casos, foram plotadas as magnitudes da DFT sobre o sinal filtrado. Sendo esta denominada: Resposta de Magnitude em Frequência (FMR) do Filtro ao Impulso Unitário Finito (FIR).

• boa Atenuação
• Atenuação Ruim



Triâng

Hamming



Projeto de Filtro FIR

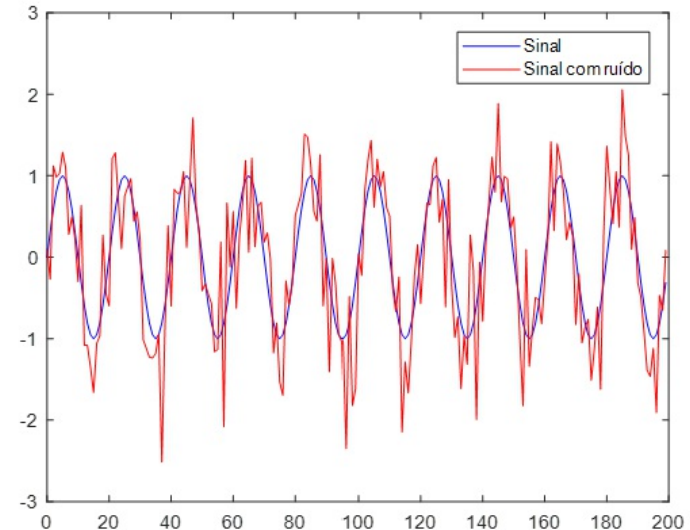
Como visto anteriormente, o uso de janelamento (triangular, Hamming, etc), melhora a resposta de um filtro FIR.

>> Topico7Exemplo2.m

- Definições iniciais: (%% Parte A - Inicio)
 - F_s : 100Hz
 - N : 200 amostras
 - Criamos um sinal senoidal de 5 Hz e adicionamos um ruído com relação sinal ruído de $S/R=5$
 - `x_noise = awgn(x_d,5);`

Definimos a quantidade de coeficientes do filtro (empiricamente),

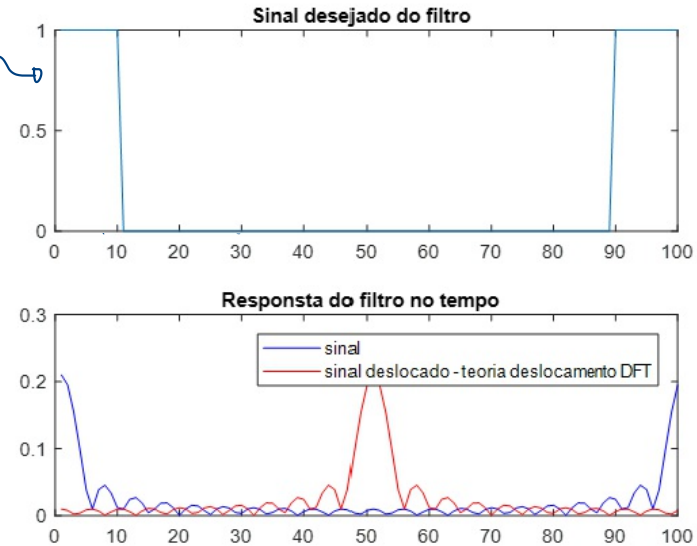
Criamos a função impulso unitário com quantidade de coeficientes superior ao de coeficientes do filtro



Projeto de Filtro FIR

Como visto anteriormente, o uso de janelamento (triangular, Hamming, etc), melhora a resposta de um filtro FIR.

- >> Topico7Exemplo2.m
 - fir1 Matlab : (%% Parte B - fir1 Matlab - Lowpass)
 - Para motivos de comparação e desempenho utilizamos a função "fir1" do Matlab/Octave, e criamos um filtro FIR lowpass, antes de iniciarmos o projeto.
 - Projeto filtro (%% Parte C - Projeto Filtro FIR Lowpass)
 - Criou a resposta de magnitude em frequência (FMR) ideal do nosso filtro, e fizemos IFFT da mesma.



Projeto de Filtro FIR

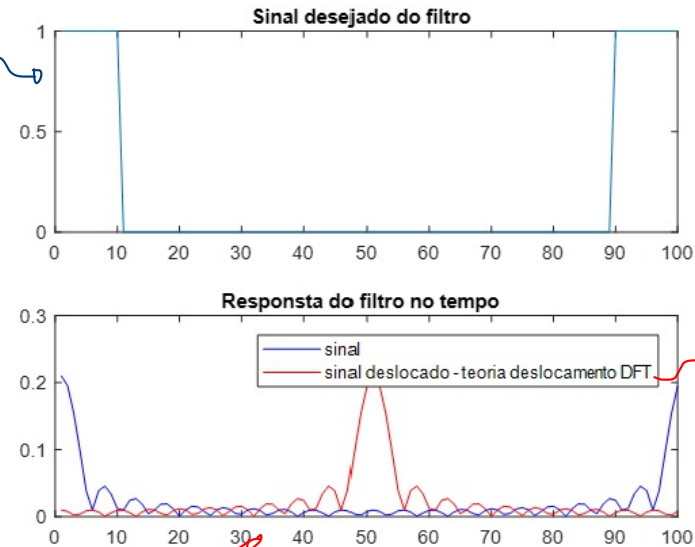
Como visto anteriormente, o uso de janelamento (triangular, Hamming, etc), melhora a resposta de um filtro FIR.

- >> Topico7Exemplo2.m - fir1 Matlab : (%% Parte B - fir1 Matlab - Lowpass)
- Para motivos de comparação e desempenho utilizamos a função "fir1" do Matlab/Octave, e criamos um filtro FIR lowpass, antes de iniciarmos o projeto.

- Projeto filtro (%% Parte C - Projeto Filtro FIR Lowpass)
 - Criou a resposta de magnitude em frequência (FMR) ideal do nosso filtro, e fizemos IFFT da mesma.

Utilizando o sinal deslocado no tempo (x2_), extrai-se a quantidade a quantidade de pontos centrais do sinal, igual ao número de coeficientes (taps).

No entanto, como vimos anteriormente seria interessante que tais coeficientes tivessem distribuição próxima à uma janela (triangular, Hamming, etc)



Diminuí a
parte oscilante!

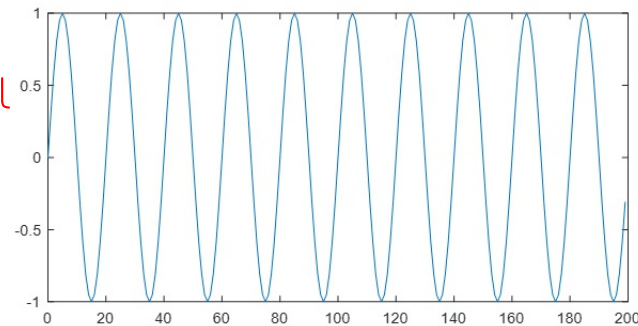
MA + LB
(x2 -)

Projeto de Filtro FIR

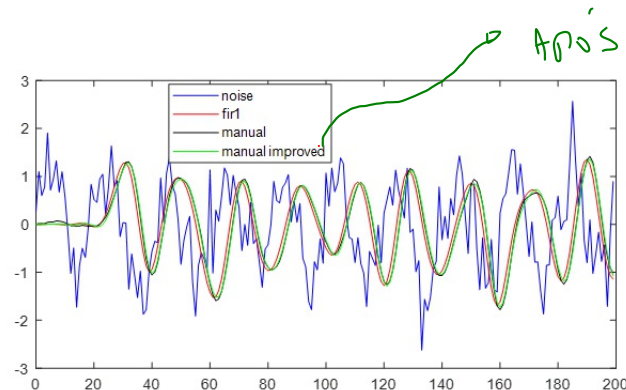
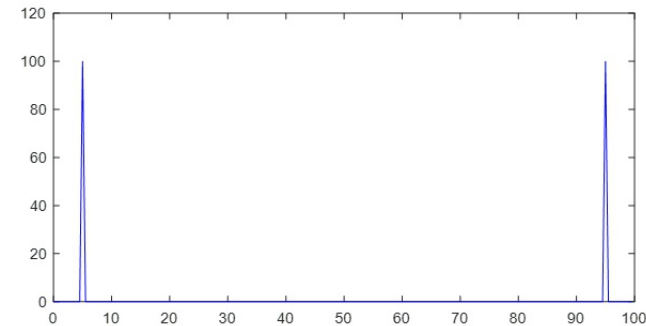
Como visto anteriormente, o uso de janelamento (triangular, Hamming, etc), melhora a resposta de um filtro FIR.

>> Topico7Exemplo2.m

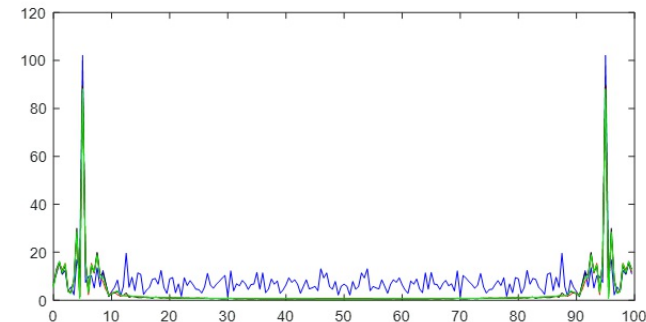
siml
original



FFT



Após janelamento DE Hamming!



Projeto de Filtro FIR

Resumo de Projeto de Filtros

- 1) Definir o formato do filtro no domínio da frequência, com as devidas frequências de corte:
 - a. Lowpass
 - b. Bandpass
 - c. Highpass
 - d. Etc
- 2) Extrair os coeficientes do filtro
 - a. Realizar a IFFT sobre o sinal no domínio do tempo
 - b. Deslocar o sinal (teoria do deslocamento da FFT)
 - c. Extrair a quantidade de pontos centrais do sinal no domínio do tempo, gerando o $b[k]$
- 3) Avaliar a qualidade do filtro:
 - a. Definir uma função impulso unitário, algumas vezes superior ao vetor de coeficientes do filtro ($b[k]$)
 - b. Aplicar o filtro sobre o função impulso unitário, e verificar a resposta de magnitude em frequência

↳ no MATLAB é automático
(gratuito de b')

