

# **FILTROS DIGITAIS FIR e IIR**

André Heidemann Iarozinski

**CURITIBA**

**2015**

## 1. Introdução

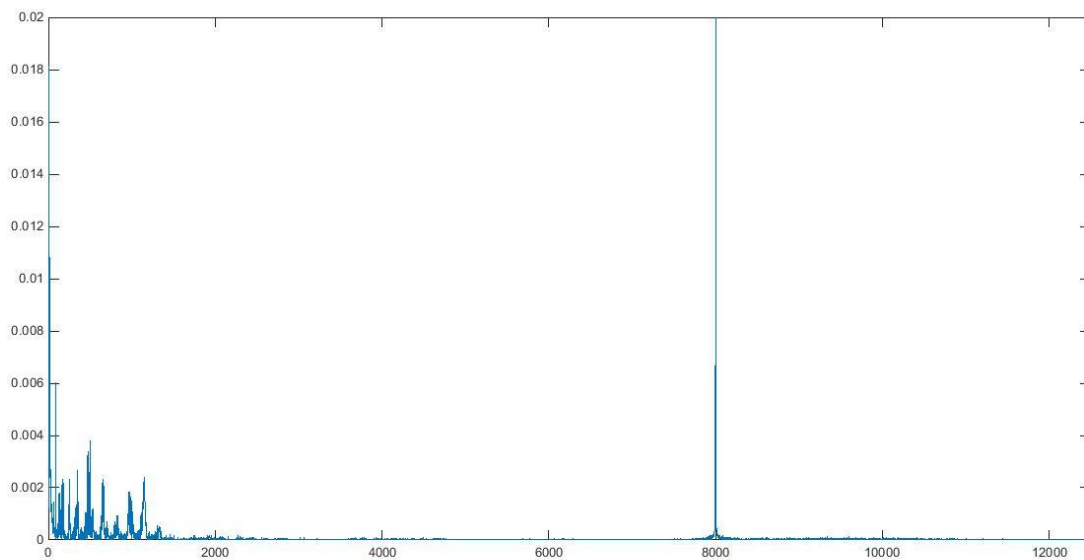
O objetivo do trabalho consiste na captura de um áudio com uma fonte poluidora para posteriormente ser filtrado através de um filtro digital desenvolvido no MATLAB.

## 2. Desenvolvimento

De início foi utilizado um software em um smartphone para gerar a fonte poluidora, um sinal de 8kHz utilizado para treinamento de animais de estimação.

Foi gravado então no MATLAB a locução das palavras “um, dois, três, quatro, cinco” durante 5 segundos com o sinal de 8KHz simultaneamente. A taxa de amostragem utilizada foi de 44100Hz para a captação do áudio.

O espectro do sinal pode ser visto abaixo através da FFT obtida.



O código utilizado para o procedimento foi o seguinte:

```
fs=44100; % freq de amostragem
recc = audiorecorder(44100, 16, 1);
disp('inicio da gravacao')
recordblocking(recc, 5); % duração da gravação: 5 seg
disp('fim da gravacao');

u = getaudiodata(recc);

S = fft(u);

L = length(S); % normalizando S
P2 = abs(S/L);
P = P2(1:L/2+1);
P(2:end-1) = 2*P(2:end-1);

f = fs*(0:(L/2))/L;

plot(f,P) % plotando a FFT de 0Hz até fs/2
```

**O áudio foi salvo na variável “u”.**

**Em seguida foi projetado o filtro FIR cujos parâmetros foram os seguintes:**

**-Atenuação mínima de 40 dB na banda de rejeição**

**-Atenuação máxima de 2 dB na banda passante**

**-fs (freq. de borda da banda de passagem) foi definida como 7kHz**

**-fs (freq. de borda da banda de rejeição) foi definida como 7.5kHz**

**Segue abaixo o algoritmo desenvolvido para o filtro FIR:**

```
% Filtro passa baixas FIR

fa = 44100; % frequencia de amost
s1 = u;    %% sinal gravado

fp = 7000; % frequência de passagem
fs = 7500; % frequência de corte

% normalização das frequências
wp = (fp/(fa/2))*pi
ws = (fs/(fa/2))*pi

bt = ws - wp; %banda de transição

M = ceil((6.6*pi/bt)) + 1; % M de acordo com a tabela das janelas
wc = (ws + wp)/2; %frequência de corte intermediária

alfa = (M-1)/2; % filtro passa baixas ideal
n=0:M-1;
m = n- alfa + eps;

hd = sin(wc*m) ./ (pi*m) ; % resposta impulsiva do fpb ideal
jan = hamming(M)'; %calcula a janela de hamming
h = hd.*jan; % multiplicação entre os vetores

sinal_filtrado = conv(h,s1); %convolução entre os sinais

sound(sinal_filtrado,fa);

S = fft(sinal_filtrado);

L = length(S); % normalizando S
P2 = abs(S/L);
P = P2(1:L/2+1);
P(2:end-1) = 2*P(2:end-1);

f = fa*(0:(L/2))/L;

plot(f,P) % plotando a FFT do sinal filtrado de 0Hz até fs/2
```

Em seguida foi projetado o filtro Butterworth IIR cujos parâmetros foram os seguintes:

-Atenuação mínima de 40 dB na banda de rejeição

-Atenuação máxima de 2 dB na banda passante

-wp foi definido como  $0.3\pi$  pois:

$f_s/2 \cdot 0.3 = 6615\text{Hz}$ , a voz humana é considerada uma faixa de frequência de 400 até 3.4KHz mas optei por escolher uma frequência acima para preservar a qualidade da voz e remover somente a fonte poluidora.

-ws foi definido com  $0.35\pi = 7717\text{Hz}$  (garante fonte poluidora na faixa de rejeição)

Abaixo o algoritmo desenvolvido para o filtro IIR:

```
%%% filtro passa-baixa IIR  butterworth

rp=2;
as=30;

wp=0.3*pi;      %%% freq de borda da banda de passagem
ws=0.35*pi;     %%% freq de borda da banda de rejeição

T=1;
wap = wp/T;
was = ws/T;
%% prototipo do filtro analogico ( isso é o qe muda p/ butter, cheby e elip)

N = ceil( log10 ( (10^(rp/10) -1) / (10^(as/10) -1) ) / (2*log10(wp/ws)) );

wc = (wp/(((10^(rp/10)-1))^(1/(2*N)))));

[z,p,k] = buttap(N); %%% retorna os zeros, os polos e o ganho

num = real(poly(z));          % numerador da func de transf Ha(s)
num = num*(wc^N)*k;
den= real(poly(p*wc));        %%% denominador da funcao de transf
[numd,dend] =impinvar(num,den,T); %%% discretizando
sys =tf(numd,dend);

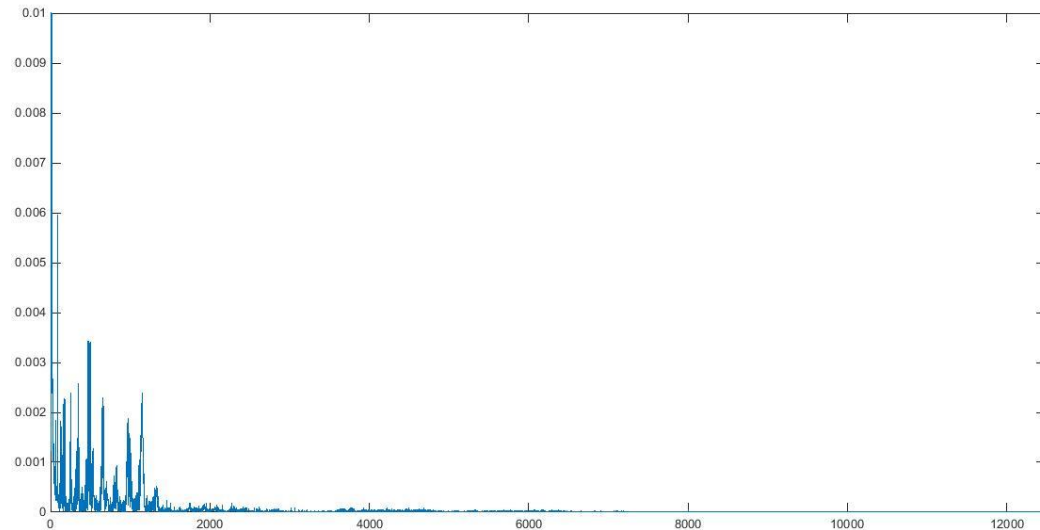
x=filter(numd,dend,u);        % filtragem do sinal
sound(x,44100);

w=0:pi/100:pi;
H = freqz(numd,dend,w);          %%% Ha(s)
Hma = abs(H);
Hfase = angle(H);
Hmdb = 20*log10((Hma+eps)/(max(Hma)));

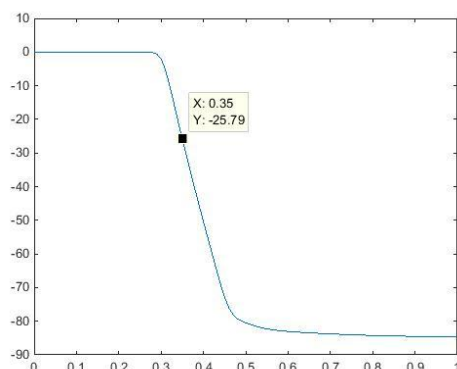
plot(w/pi, Hmdb);              % filtro em dB
```

### 3.Resultados

O filtro FIR funcionou conforme o esperado, filtrou completamente a fonte poluidora e atingiu os requisitos do projeto. A resposta espectral do sinal filtrado foi a seguinte:

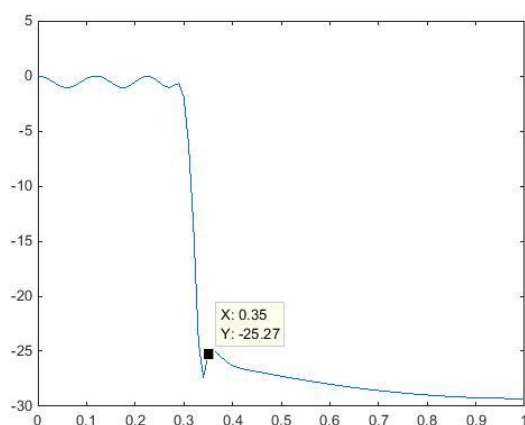


Já o filtro Butterworth apresentou alguns problemas para as frequências escolhidas. Com atenuação de 25 dB na banda de rejeição o filtro respondeu da seguinte forma:



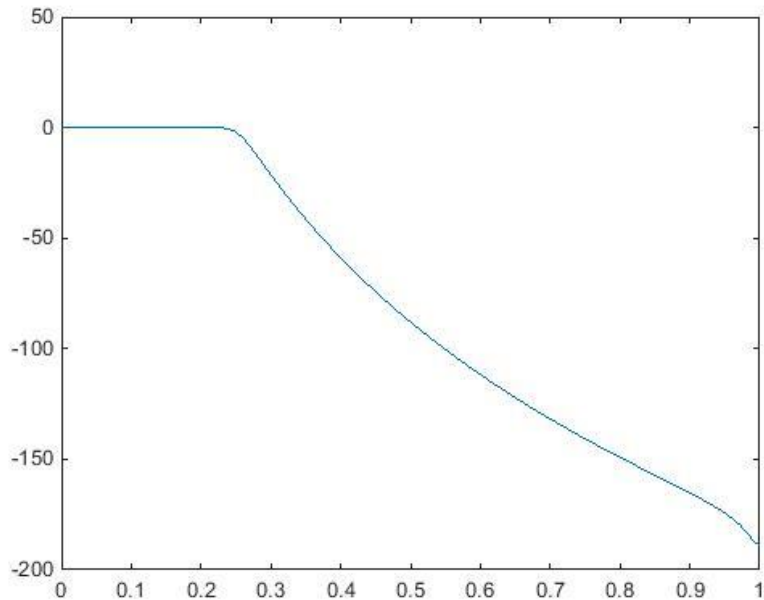
Na figura acima, está indicado  $\omega_s$ .

Com atenuação de 30 dB o filtro ficou no limite da estabilidade:



Acima de 30 dB o filtro ficava instável e tive que alterar a frequência de borda da banda de passagem de  $wp=0.3\pi$  para  $wp=0.25\pi$  o que resultou em uma perda de espectro da voz, que foi de 6615Hz para 5512Hz de frequência de borda. Após esta alteração o filtro funcionou corretamente e cumpriu os requisitos do projeto.

Abaixo o filtro com  $wp=0.25\pi$ :



Abaixo o sinal filtrado com o filtro Butterworth corrigido:

