Aluno: Jedson Jhones Barbosa Alves

Exercício 1 – Filtro de média móvel

Existem reportes[1] que sugerem que pacientes que adquiriram COVID-19 apresentam um incremento da Frequência Respiratória (FR). O valor da FR pode ser obtido mediante análise do sinal respiratório. Dada a importância do sinal respiratório, o DCA pretende implementar métodos de processamento que permitam melhorar a qualidade de sinais de respiração coletados em pacientes com Covid-19. Neste sentido, implemente filtros de média móvel que possibilitem suavizar o sinal de respiração (use uma janela de $M = 20\ amostras$).

a) Método 1: filtro de média móvel usando a fórmula:

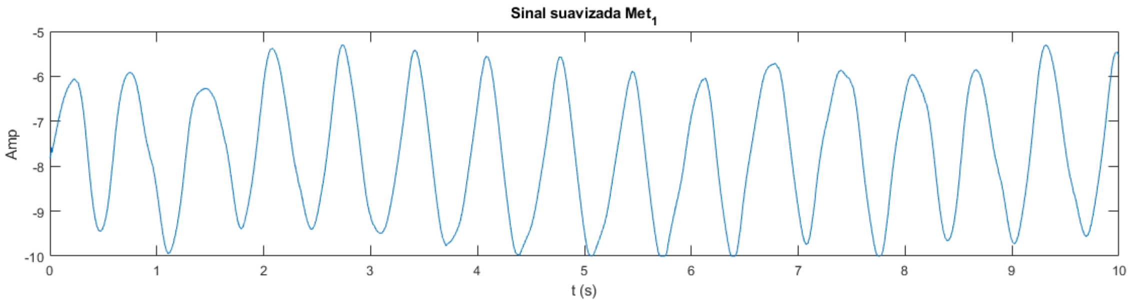
$$y[n] = \frac{1}{M}\sum_{i=0}^{M-1} x[n-1]$$

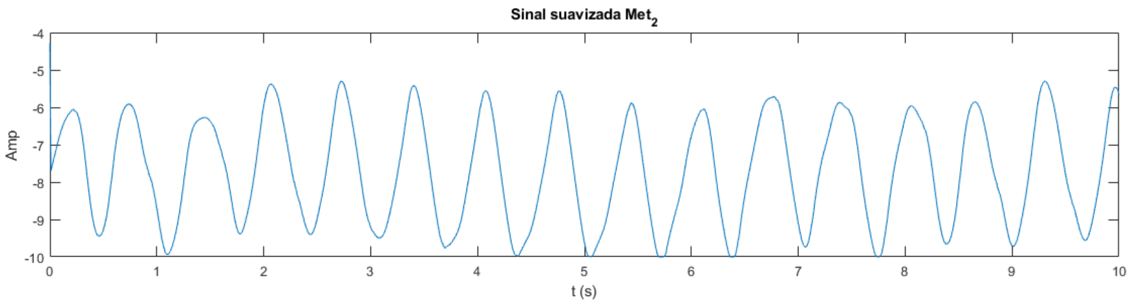
```
xSmooth = x;
M = 20;
for iSample = M:NSamples
   IndWindow = (iSample-M+1):iSample;
   xSmooth(iSample) = mean(x(IndWindow));
end
```

b) Método 2: filtro de média móvel usando a função de convolução (***conv***) usada em pacotes de análise como Matlab/Octave

```
h = ones(1,M)/M;
xSmooth1 = conv(x,h,'same');
```

c) Plote num gráfico o sinal original (sem a aplicação do filtro de média móvel) e o sinal suavizado pelos métodos 1 e 2. Nos gráficos deverá ser representado o tempo (em segundos) no eixo das abscissas. Apresente os 10 primeiros segundos para uma melhor visualizazão.





d) Discuta os resultados obtidos: 1) explique com suas palavras se a aplicação do filtro de média móvel melhorou de alguma forma a qualidade do sinal analisado; 2) Justifique sua resposta a partir da análise dos gráficos anteriormente obtidos.

A Partir da análise do gráfico podemos ver que os filtros média móvel realmente melhorou a qualidade do gráfico, eliminando as oscilações de baixa amplitude. Também podemos visualizar que os dois gráficos da média móvel são semelhantes.

Códigos Questão 1 (MATLAB):

```
% Abrindo o sinal respiration
MatFile = matfile('respiration.mat');
x = MatFile.respiration;
fs = 1000; % sample rate in Hz
ts = 1/fs; % sample period in sec
NSamples = length(x);

t = (0:NSamples-1)/fs;
fig = figure('Position',[10 10 900 300],'color','w');
lin1 = plot(t,x);
xlabel('t (s)')
ylabel('Amp')
xlim([0 10])

%LETRA A
xSmooth = x;
M = 20;
for iSample = M:NSamples
   IndWindow = (iSample-M+1):iSample;
   xSmooth(iSample) = mean(x(IndWindow));
end

%LETRA B

h = ones(1,M)/M;
xSmooth1 = conv(x,h,'same');

%LETRA C

fig = figure('Position',[10 10 900 600],'color','w');
subplot(2,1,1)
lin1 = plot(t,xSmooth);
xlabel('t (s)')
ylabel('Amp')
title('Sinal suavizada Met_1')
xlim([0 10])
subplot(2,1,2)
lin1 = plot(t,xSmooth1);
```

```
xlabel('t (s)')
ylabel('Amp')
title('Sinal suavizada Met_2')
xlim([0 10])
```

Exercício 2 – Relação entre a TFTD e a TZ

A Transformada de Fourier de Tempo Discreto (TFTD) de uma sequência $x[n]$ coincide com os valores da Transformada Z (TZ) de dita sequência, avaliada na circunferência unitária do plano z (verdadeiro sempre que a circunferência unitária esteja dentro da RDC da TZ).

a) Determine a expressão matemática de $X(z)$ e a sua região de convergência (RDC).

Exercicio 2: Relação entre a TFTD e a TZ.

a)

Dado que x[n]= $\frac{1}{3}^n u[n]$ que é um sinal que representa uma sequência exponencial lateral direita. Logo a expressão matemática de X(z) e a sua região de convergência(RDC) para uma sequência exponencial lateral direita:

$$x[n] = a^n u[n] = \begin{cases} a^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

$$X(z) = \sum_{n=-\infty}^{\infty} a^n u[n] z^{-n}$$

$$X(z) = \sum_{n=0}^{\infty} a^n z^{-n} = \sum_{n=0}^{\infty} (a\, z^{-1})^n$$

$$X(z) = \frac{1}{1-az^{-1}} = \frac{z}{z-a} \quad \text{(Expressão matemática de X(z))}$$

$$\left|ax^{-1}\right| < 1 \Rightarrow \frac{|a|}{|z|} < 1 \Rightarrow |z| > |a| \quad \text{(Região de convergência)}$$

Já para o caso de uma exponencial lateral esquerda teríamos:

$$x[n] = -a^n u[-n-1] = \begin{cases} -a^n, & n < 0 \\ 0, & n \geq 0 \end{cases}$$

$$X(z) = \sum_{n=-\infty}^{-1} a^{n} z^{-n}$$

$$X(z) = \sum_{n=1}^{\infty} a^{-n} z^{n} = \; - \sum_{n=1}^{\infty} (a^{-1} z\,)^{n}$$

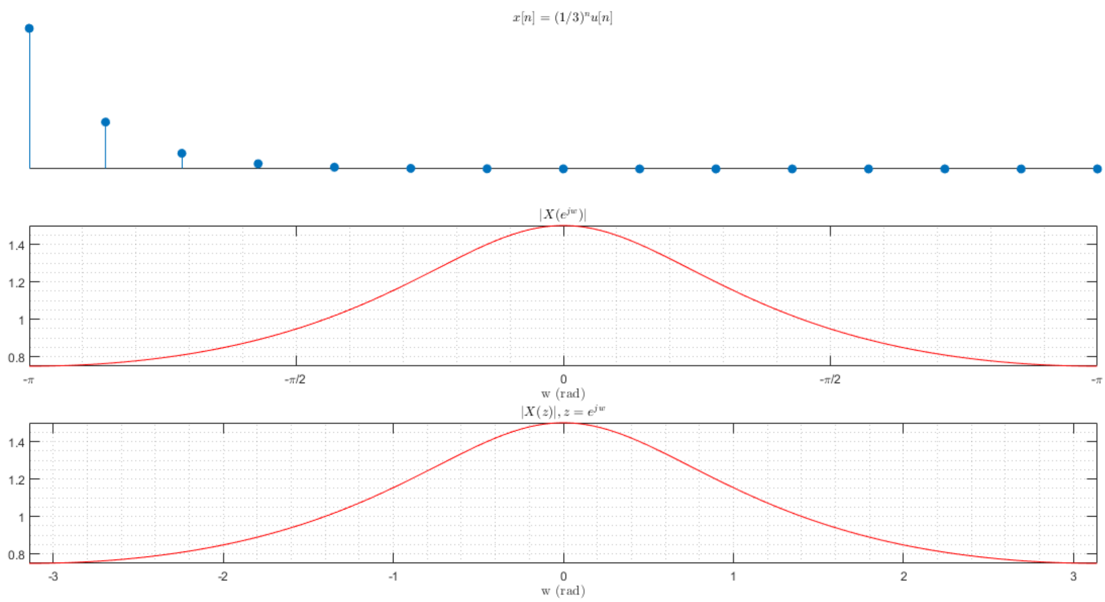
$$1 - \sum_{n=0}^{\infty} (a^{-1} z)^{n}, \sum_{n=1}^{\infty} (a^{-1} z\,)^{n} = \frac{a}{a-z} = \frac{z}{z-a}$$

$1 - \frac{a}{a-z} = \frac{-z}{a-z} =$ $\frac{z}{z-a}$ =, $|z| < |a|$ (Expressão matemática de X(z) e Região de convergência)

Sendo assim a única diferença entre elas, sendo a sua região de convergência.

b) Para efeitos da simulação use as primeiras 15 amostras do sinal $x[n]$ para calcular a TFTD usando a implementação apresentada na Figura 2 b). Para o cálculo da TZ use a expressão matemática de $X(z)$.

c) Represente graficamente: 1) a sequência $x[n]$; 2) magnitude de $X(z)$ avaliada para $z = e^{j\omega}$ e 3) a magnitude a TFTD: $|X(e^{j\omega})|$



Códigos questão 2 (MATLAB)

```
NSamples = 15;
Samples = 0:(NSamples-1);

x = (1/3).^Samples;
%criando o vetor de frequencias de w
w = -pi:0.01:pi;
ejw = exp(-1i*w);
```

```
X = 0;
k = 0;
for n = 0:(NSamples-1)
   k = k + 1;
   X = X + x(k)*ejw.^n;
end
% calculando a transformada z
Xz_fcn = @(z,a) (z)./(z-a);
z1 = exp(1i*w);
Xz_ejw = Xz_fcn(z1,1/3);

%plotagems

fig = figure('Position',[10 10 1200 600],'color','w');

% Plot x[n]
ax = subplot(3,1,1);
lin1 = stem(Samples,x,'filled');
ax.Visible = 'off';
title('$x[n] = (1/3)^nu[n]$','Interpreter','latex','Visible','on');

str = {'-\pi','-\pi/2','0','-\pi/2','-\pi'};
% Plot results for DTFT
ax = subplot(3,1,2);
lin2 = plot(w,abs(X),'-r','LineW',1);
set(ax,'XTick',-pi:(pi/2):pi,'XTickLabel',str);
% ax.TickLabelInterpreter = 'latex';
tl = title('$|X(e^{jw})|$','Interpreter','Latex');
grid minor
% ylim([0 12])
xlim([-pi,pi])
xlabel('w (rad)','Interpreter','latex')

% Plot results form Z transform
ax = subplot(3,1,3);
lin2 = plot(w,abs(Xz_ejw),'-r','LineW',1);
% ax.TickLabelInterpreter = 'latex';
tl = title('$|X(z)|,z = e^{jw}$','Interpreter','Latex');
grid minor
% ylim([0 12])
xlim([-pi,pi])
xlabel('w (rad)','Interpreter','latex')
```

Exercício 3 – Transformada de Fourier Discreta (TFD)

Determine a TFD de um segmento de sinal duração de $N = 2^{11}$ amostras, obtido a partir do arquivo "***respiration.mat***";

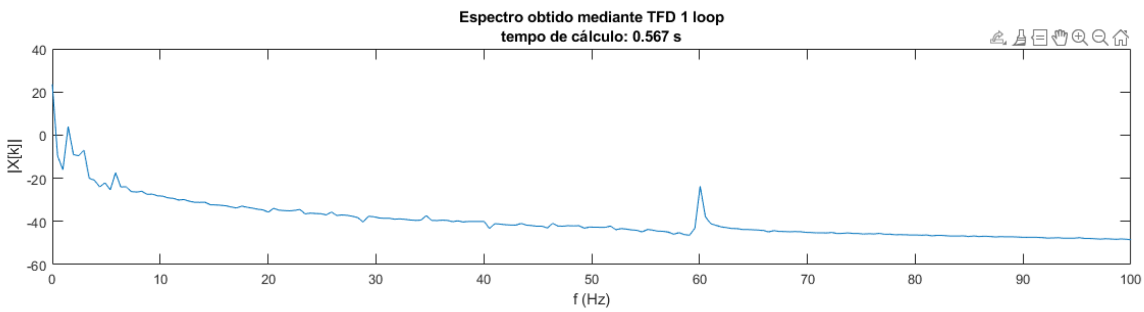
a) Usando alguma implementação diretamente da fórmula da TFD

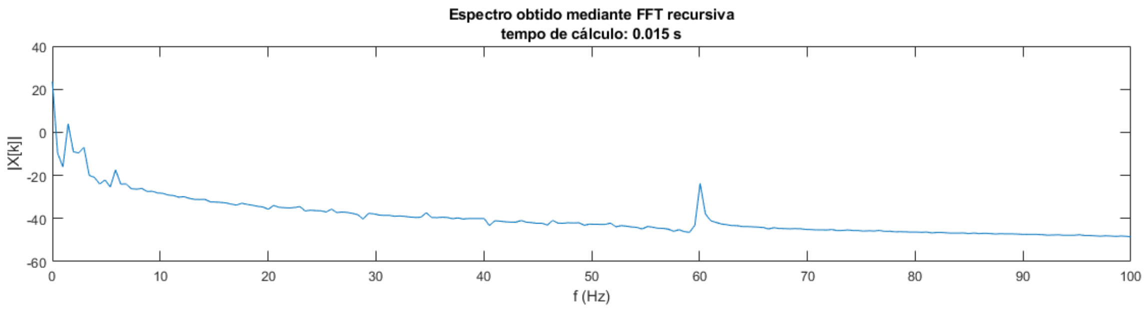
```
nfft = 2^11;
x1 = x(1:nfft);
tic
YY1 = DFT_1loop(x1);
eTime(1) = toc;
YY1 = abs(YY1)/nfft;
YY1 = 2*YY1(1:nfft/2+1);
f = linspace(0,fs/2,nfft/2+1);
```

b) Usando uma implementação da FFT recursiva

```
tic;
Y1 = fft_rec(x1);%fft_rec01(S_1024pts);
eTime(2) = toc;
Y1 = abs(Y1)/nfft;
Y1 = 2*Y1(1:nfft/2+1);
```

c) Baseado nos resultados dos itens a) e b) obtenha a magnitude do espectro do sinal analisado (chamaremos a magnitude do espetro simplesmente de: espectro do sinal). Plote em 2 gráficos separados os espectros obtidos em função da frequência (em Hz).





d) Compare os resultados obtidos em a) e b) no que se refere ao tempo de cálculo e a semelhança nos resultados apresentados nos gráficos do item c).

A FFT recursiva ocorreu mais rápido que pela utilização da TFD 1 loop

Códigos questão 3 (MATLAB)

```
% Read respiration signal
MatFile = matfile('respiration.mat');

x = MatFile.respiration;
fs = 1000; % sample rate in Hz
ts = 1/fs; % sample period in sec

NSamples = length(x);

% Create time vector
t = (0:NSamples-1)/fs;

fig = figure('Position',[10 10 900 300],'color','w');
lin1 = plot(t,x);
xlabel('t (s)')
ylabel('Amp')
xlim([0 10]) % shows from 0 - 10 sec
% ylim([-0.1 1.1])

nfft = 2^11;
x1 = x(1:nfft);
%% LETRA A
tic
YY1 = DFT_1loop(x1);
eTime(1) = toc;
YY1 = abs(YY1)/nfft;
YY1 = 2*YY1(1:nfft/2+1);
f = linspace(0,fs/2,nfft/2+1);
%% LETRA B
tic;
Y1 = fft_rec(x1);%fft_rec01(S_1024pts);
eTime(2) = toc;
Y1 = abs(Y1)/nfft;
Y1 = 2*Y1(1:nfft/2+1);
%% LETRA C
subplot(2,1,1)
plot(f,20*log10(YY1))
xlabel('f (Hz)')
ylabel('|X[k]|')
str = sprintf('tempo de cálculo: %.3f s',eTime(1))
title({'Espectro obtido mediante TFD 1 loop',str})
```

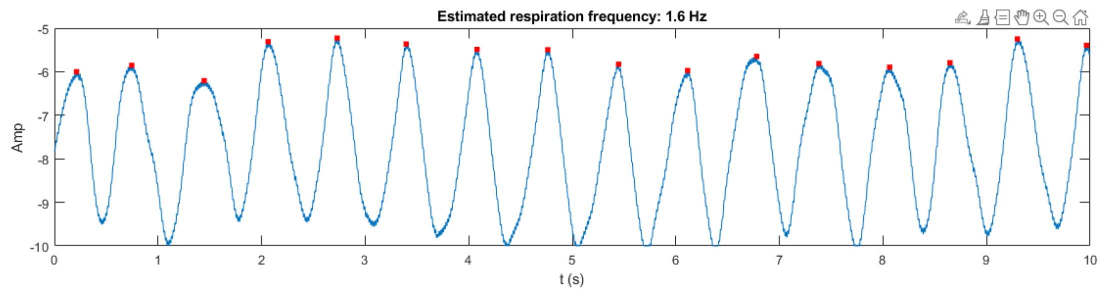
```
xlim([0 100])

subplot(2,1,2)
plot(f,20*log10(Y1))
xlabel('f (Hz)')
ylabel('|X[k]|')
str = sprintf('tempo de cálculo: %.3f s',eTime(2))
title({'Espectro obtido mediante FFT recursiva',str})
xlim([0 100])
```

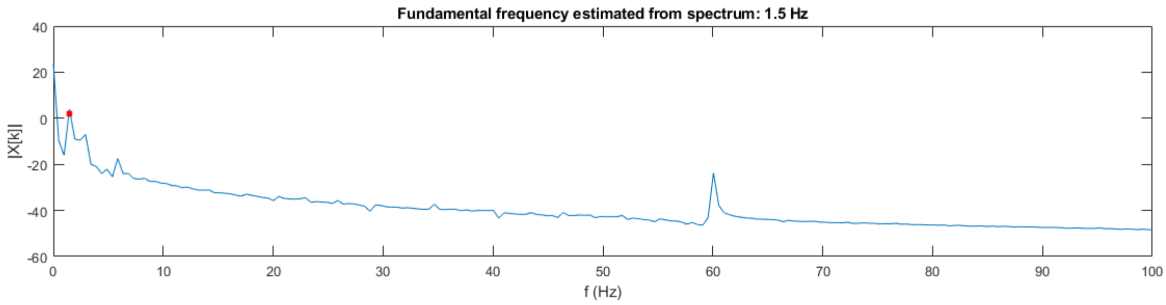
Exercício 4 – Estimação de frequência respiratória

Tomando em consideração importância do parâmetro de frequência respiratória (FR), determine o valor da FR do sinal contido no arquivo "***respiration.mat***". Use para a análise um segmento de sinal de duração correspondente a 10 segundos (lembrar que $fs = 1\ kHz$). Para determinar o valor da FR:

a) Plote o sinal no domínio tempo (amplitude *vs* tempo). Calcule o número de ciclos do sinal (no domínio tempo), divida pela duração (em segundos) do sinal analisado.



b) Retire a média do sinal, calcule o espectro do sinal resultante. Determine o valor de frequência correspondente ao maior pico espectral. Compare os resultados com os obtidos no item a).



c) Explique com as suas palavras os resultados obtidos, e se existe alguma relação entre as 2 formas de análise. Comente qual você acredita seja a solução mais robusta e justifique.

Os resultados são bastante parecidos, mas a análise no domínio espectral acaba sendo mais robusto, por permitir determinar qual seria a frequência fundamental.

Códigos questão 4 (MATLAB)

```
% Read respiration signal
MatFile = matfile('respiration.mat');

x = MatFile.respiration;
fs = 1000; % sample rate in Hz
ts = 1/fs; % sample period in sec

NSamples = length(x);

% Create time vector
t = (0:NSamples-1)/fs;

fig = figure('Position',[10 10 900 300],'color','w');
lin1 = plot(t,x);
xlabel('t (s)')
ylabel('Amp')
xlim([0 10]) % shows from 0 - 10 sec
% ylim([-0.1 1.1])

%% LETRA A
TSim = 10; % tempo de simulação
N = TSim*fs;
x1 = x(1:N);
t1 = t(1:N);
[PeakVal,LocPeak] =findpeaks(x1,'MinPeakDistance',0.5*fs);

subplot(2,1,1)
lin1 = plot(t1,x1);
xlabel('t (s)')
ylabel('Amp')

LinPeak = line(t(LocPeak),PeakVal,'Marker','.','MarkerSize',12,...,
        'LineStyle','none','Color','r');

NPeaks = length(PeakVal);
FreqEstimated =  NPeaks/TSim; % In Hz
title(sprintf('Estimated respiration frequency: %.1f Hz',FreqEstimated));

%% LETRA B
```

```
[Val, loc] = findpeaks(10*log10(Y1),'NPeaks',1,'SortStr','descend');

subplot(2,1,2)
LinFPeak = line(f(loc),Val,'Marker','.','MarkerSize',12,...,
        'LineStyle','none','Color','r');
title(sprintf('Fundamental frequency estimated from spectrum: %.1f Hz',f(loc)));
```