**Damien : code de mse.m corrigé pendant 2h par Perrier le 14/12 :**

clc

clear all

close all

% Programme qui la MSE en fonction du RSB

%% Choix des paramètres

[signal,Fe] = audioread('102.wav');

p = 20; %ordre du prédicteur

N = 1000;

x = signal(27100:27500); %trame

px = var(x); %variance de la trame

% figure(1);

figure(2);

% figure(3);

%% Boucle sur le RSB

K = zeros(80,1);

for RSB = 1:80

pb = px\*10^((-RSB)/10); %variance du bruit

% Définition de v et y

n = size(x);

n = n(1);

v = sqrt(pb)\*randn(n,1); %génération du signal bruit

s = x+v; %génération du signal bruité

% LPC & filtre de Wiener

[A,pe] = lpc(x,p);

G = sqrt(N\*pe);

num\_wiener = G/sqrt(pb\*N);

A\_inv = flip(A);

AA = conv(A,A\_inv);

AA(p+1) = AA(p+1)+num\_wiener^2;

sqrt\_gain\_wiener= sqrt(num\_wiener^2/sum(AA))

AAracine = roots(AA); %%probleme !!! AAracine devrait etre symetrique et il ne l'est pas ! 2 premiers valeurs non sym

poly(AAracine);

% sumAA(RSB) = sum(AA)

% sum\_poly\_AAracine = sum(poly(AAracine))

% K(RSB) = sum(AA)/sum(poly(AAracine));

% sqrt\_gain\_wiener= sqrt(num\_wiener^2/sum(AA))

% sqrt\_gain\_wiener\_bis=sqrt(num\_wiener^2/(K(RSB)\*sum(poly(AAracine))))

size\_AAracine = size(AAracine);

size\_AAracine = size\_AAracine(1);

% sqrt(sum(poly(AAracine)))

AAracine\_caus = zeros(size\_AAracine/2,1);

j=0;

for i=1:size\_AAracine

if abs(AAracine(i))<1

j = j+1;

AAracine\_caus(j) = AAracine(i);

end

end

gain\_wiener=num\_wiener^2/sum(AA);

A1=poly(AAracine\_caus);

gain\_wiener\_caus\_temp=num\_wiener/sum(A1);

K(RSB)=sqrt(gain\_wiener)/gain\_wiener\_caus\_temp;

K(RSB)\*gain\_wiener\_caus\_temp

sqrt(gain\_wiener)

y = filter(num\_wiener\*K(RSB),A1,s);

figure(100)

freqz(num\_wiener\*K(RSB),A1,1024);

pause(0.01)

%n = 512;

% figure;

% subplot(2,1,1)

% phasez(num\_wiener,A1,n);

% subplot(2,1,2)

% freqz(num\_wiener,A1,n);

MSE(RSB) = sum((x-y).^2/N);

% %Graphiques

% title('Racines de AA');

%

% Max\_x = max(x);

% Max\_s = max(s);

% Max\_y = max(y);

% M = max([Max\_x,Max\_s,Max\_y]);

% M = 1.2\*M;

%

% figure(1)

% plot(x);

% ylim([-M M]);

% title('signal non bruité x');

% xlabel('échantillons');

% pause(0.1)

%

% figure(2)

% plot(x,'b')

% ylim([-M M]);

% hold on

% plot(y,'r');

% hold off

% legend('signal original x','signal débruité y');

% title('signaux x et y');

% xlabel('échantillons');

% pause(0.3)

%

% figure(3)

% plot(y,'r')

% ylim([-M M]);

% title('signal débruité y');

% xlabel('échantillons');

end

%%

figure;

rsb = [1:80];

plot(rsb,MSE)

title('MSE');

xlabel('RSB')