RAPPORTDU TP1: ANALYSE SPECTRALE D’UN SIGNAL TRANSFORMÉE DE FOURIER DISCRÈTE

Fait par: Oumayma Bennouna (IA)

Objectif du TP:

- Représentation de signaux et applications de la transformée de Fourier discrète (TFD) sous Matlab.
- Evaluation de l’intérêt du passage du domaine temporel au domaine fréquentiel dans l’analyse et l’interprétation des signaux physiques réels.

Introduction

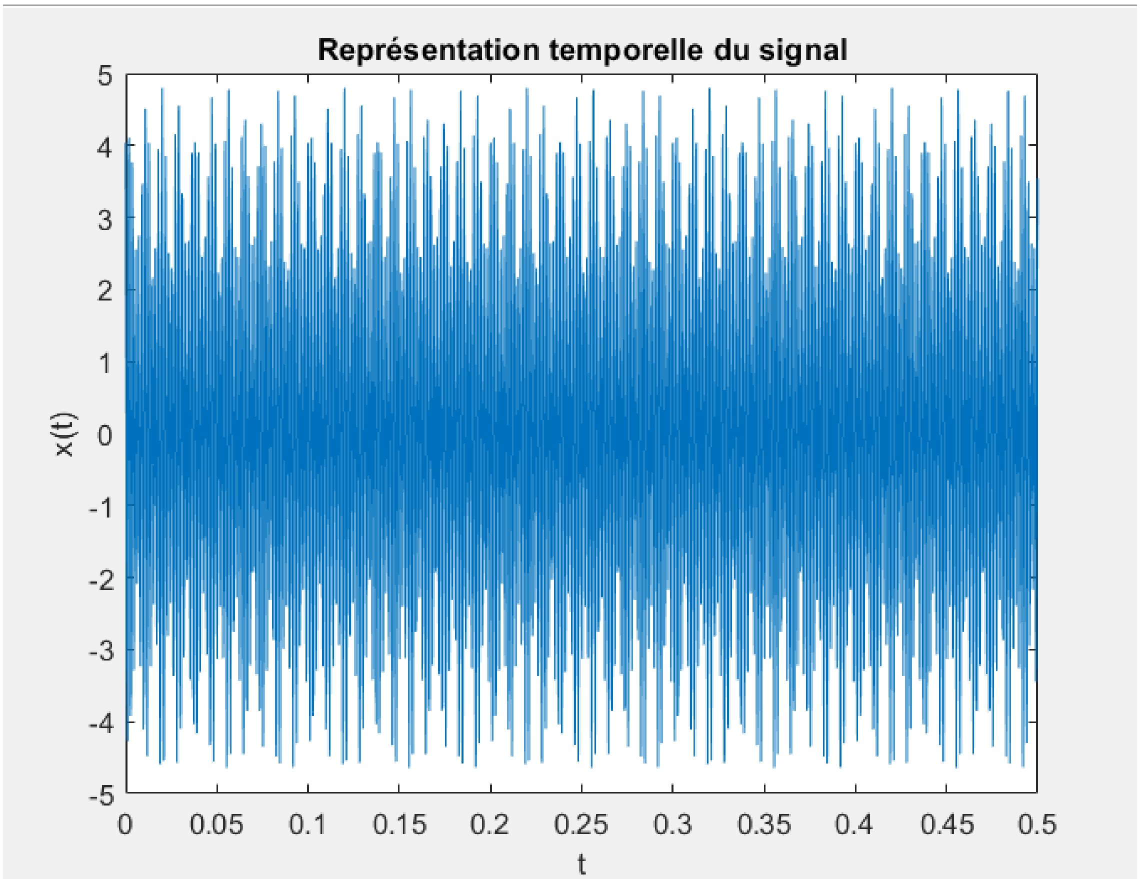
Dans ce rapport de travaux pratiques, nous allons explorer l'utilisation de la transformée de Fourier discrète (TFD) sous Matlab pour l'analyse spectrale d'un signal périodique. Nous allons utiliser la commande fft pour calculer la TFD de ce signal et visualiser son spectre en utilisant la commande abs. Nous allons également utiliser la fonction fftshift pour mieux visualiser le contenu fréquentiel du signal. Enfin, nous allons introduire un bruit blanc gaussien dans le signal original et utiliser la TFD pour révéler les fréquences qui y correspondent. L'objectif de ce TP est de comprendre les avantages de passer du domaine temporel au domaine fréquentiel dans l'analyse et l'interprétation des signaux physiques réels.

REPRÉSENTATION TEMPORELLE ET FRÉQUENTIELLE

1- Représentation temporelle

```
fe=10000;
Te=1/fe;
N=5000;

t = 0:Te:(N-1)*Te;
x = 1.2*cos(2*3.141592*440*t+1.2)+3*cos(2*3.141592*550*t)+0.6*cos(2*3.141592*2500*t);

plot(t,x)
```

Nous utilisons un signal constitués d'une somme de trois sinusoïdes de fréquences : 440 Hz , 550 Hz, 2500 Hz.



2- Représentation fréquentielle

```
%Spectre du signal

f = (0:N-1)*(fe/N);
y=fft(x);
plot(f,abs(y));
```

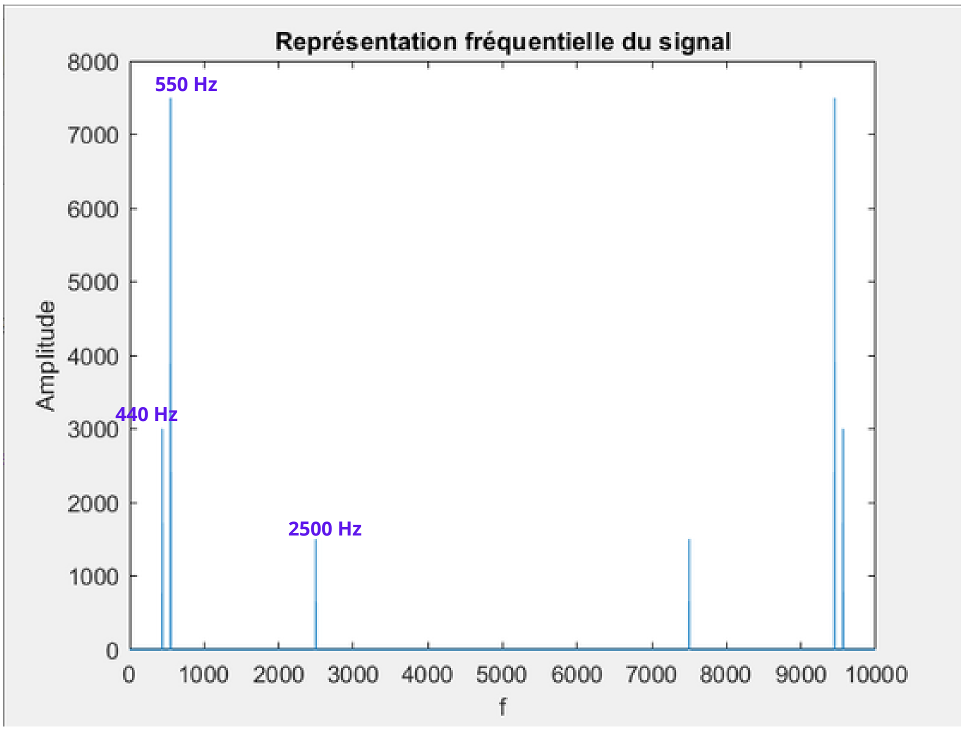
La fonction fft génère un spectre discret qui est un fonction complexe

La fonction fft effectue la transformée de Fourier pour passer du domain temporel au domaine fréquentiel

y

2001x1 complex double

	1
1	-2.1527e+02 + 0.0000e+00i
2	-63.5965 + 98.0971i
3	-56.5698 + 64.8022i

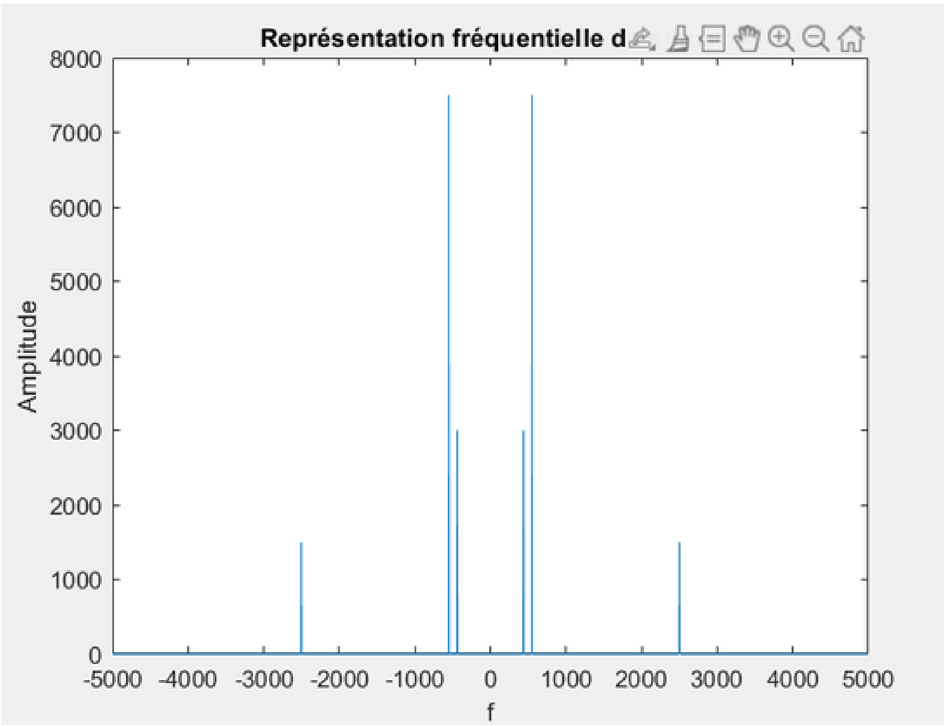


on remarque aussi que les piques sont symétriques par rapport à la fréquence Fe/2=5000Hz et on l'appelle la symétrie conjuguée

3- Représentation fréquentielle avec fftshift

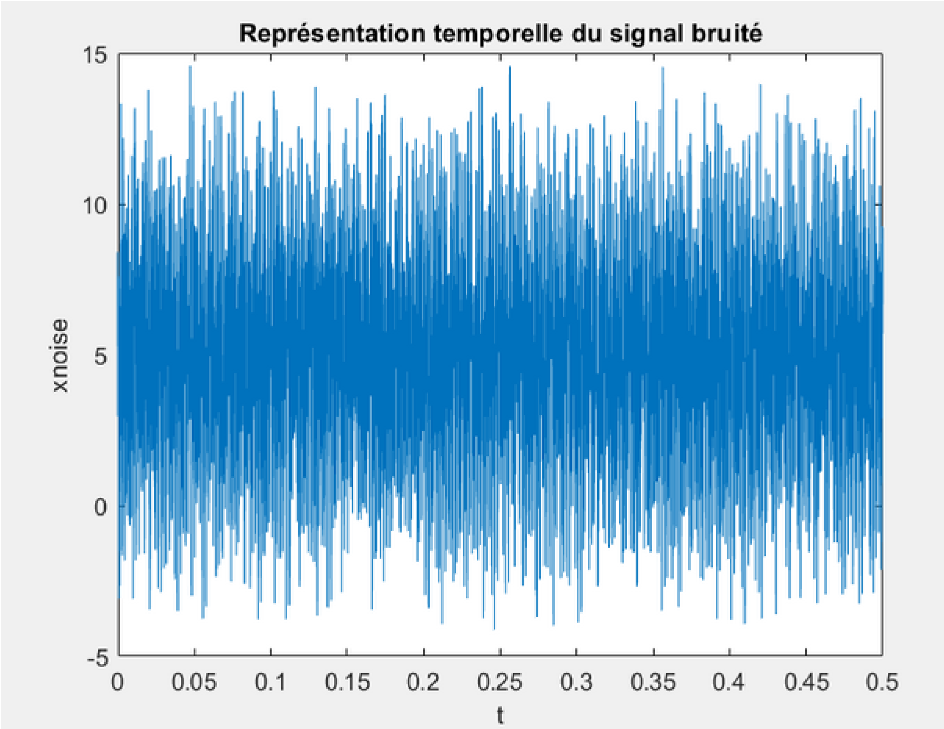
```
fshift = (-N/2:N/2-1)*(fe/N);
y=fft(x);
plot(fshift,fftshift(abs(y)));
```

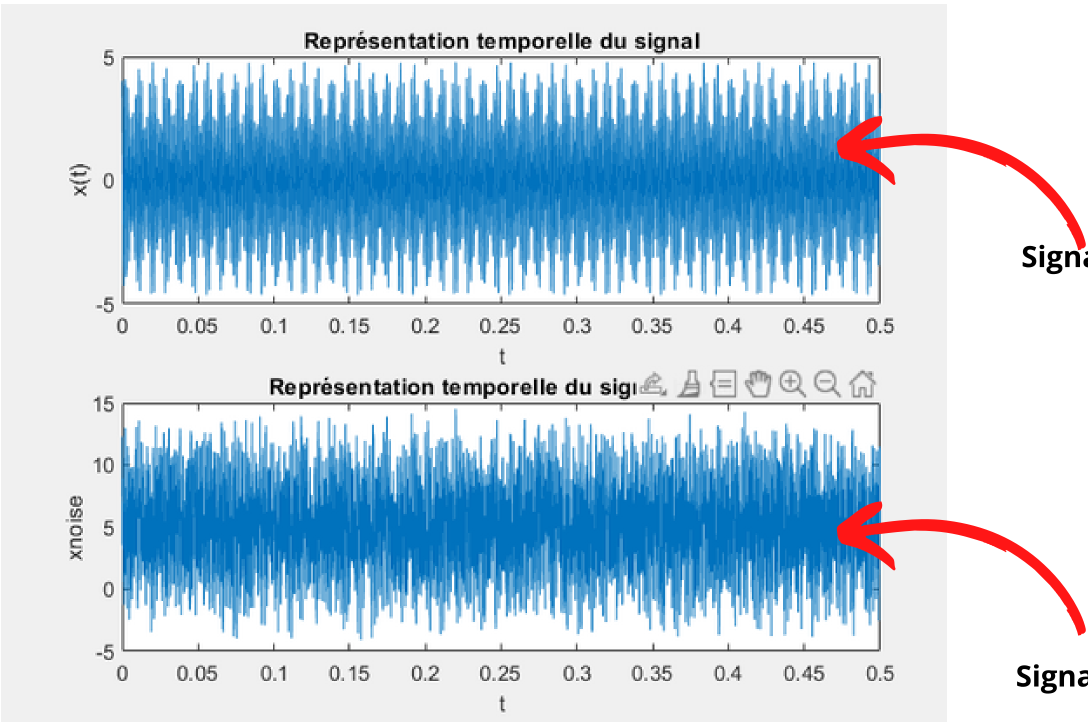
La fonction **fftshift** permet de décaler la fréquence nulle d'un signal dans le domaine fréquentiel. Cela signifie que la moitié inférieure des fréquences (comprises entre 0 et la fréquence de Nyquist) sont décalées vers la moitié supérieure, et vice versa.



4- Génération d'un bruit gaussien

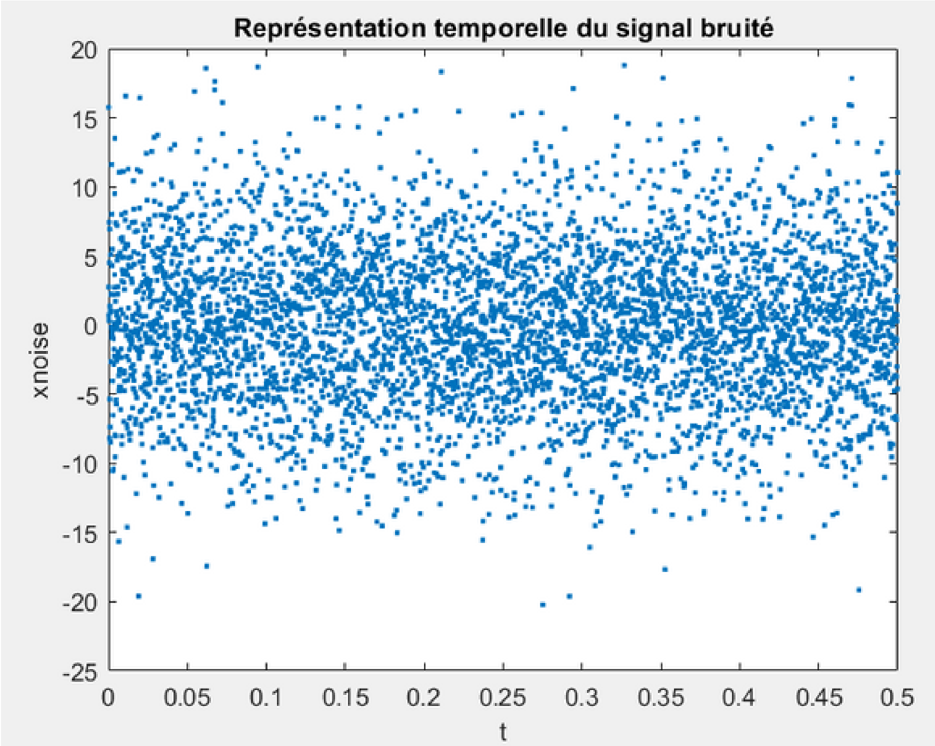
```
% Ajout du bruit

xnoise = x+2*randn(size(x));
plot(t,xnoise)
plot(t,xnoise)
xlabel('t');
ylabel('xnoise');
title('Représentation temporelle du signal bruité');
%sound(xnoise)
```





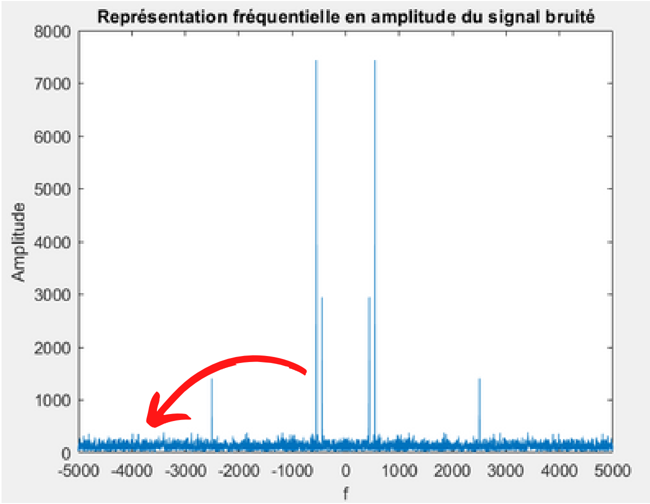
Signal normal

Signal bruité



Nous remarquons que le bruit est un signal stationnaire, car la plupart des valeur sont centrés sur 0, entre -5 et 5, ce qui venut dire que la moyenne sera toujours égale à 0.

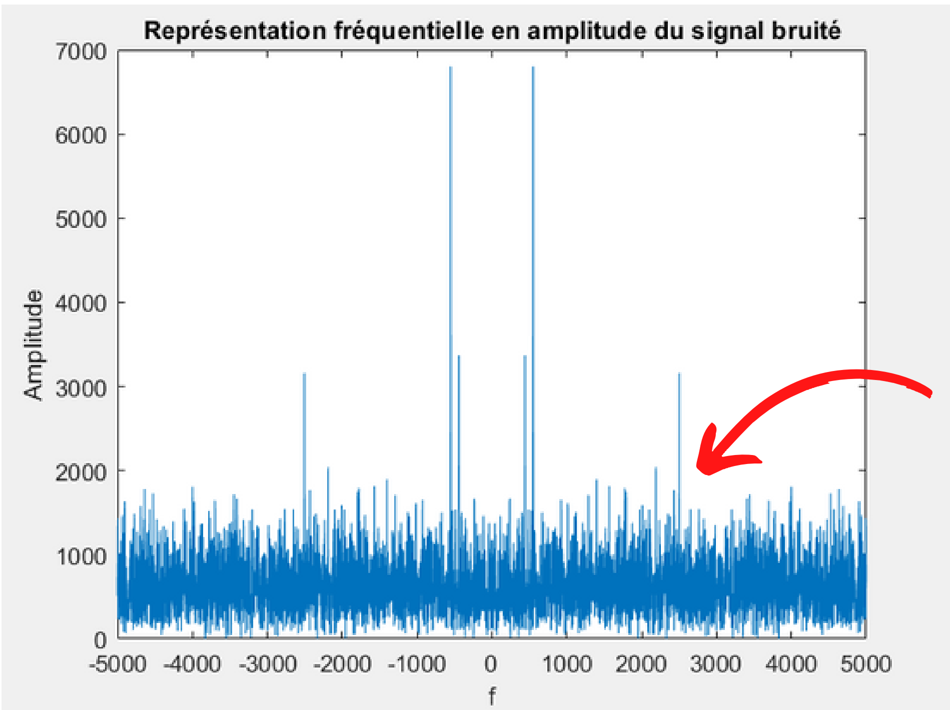
```
%Spectre du signal bruité
f = (0:N-1)*(fe/N);
fshift = (-N/2:N/2-1)*(fe/N);
ynoise=fft(xnoise);
plot(fshift,fftshift(abs(ynoise)));
xlabel('f');
ylabel('Amplitude');
title('Représentation fréquentielle en amplitude du signal');
```

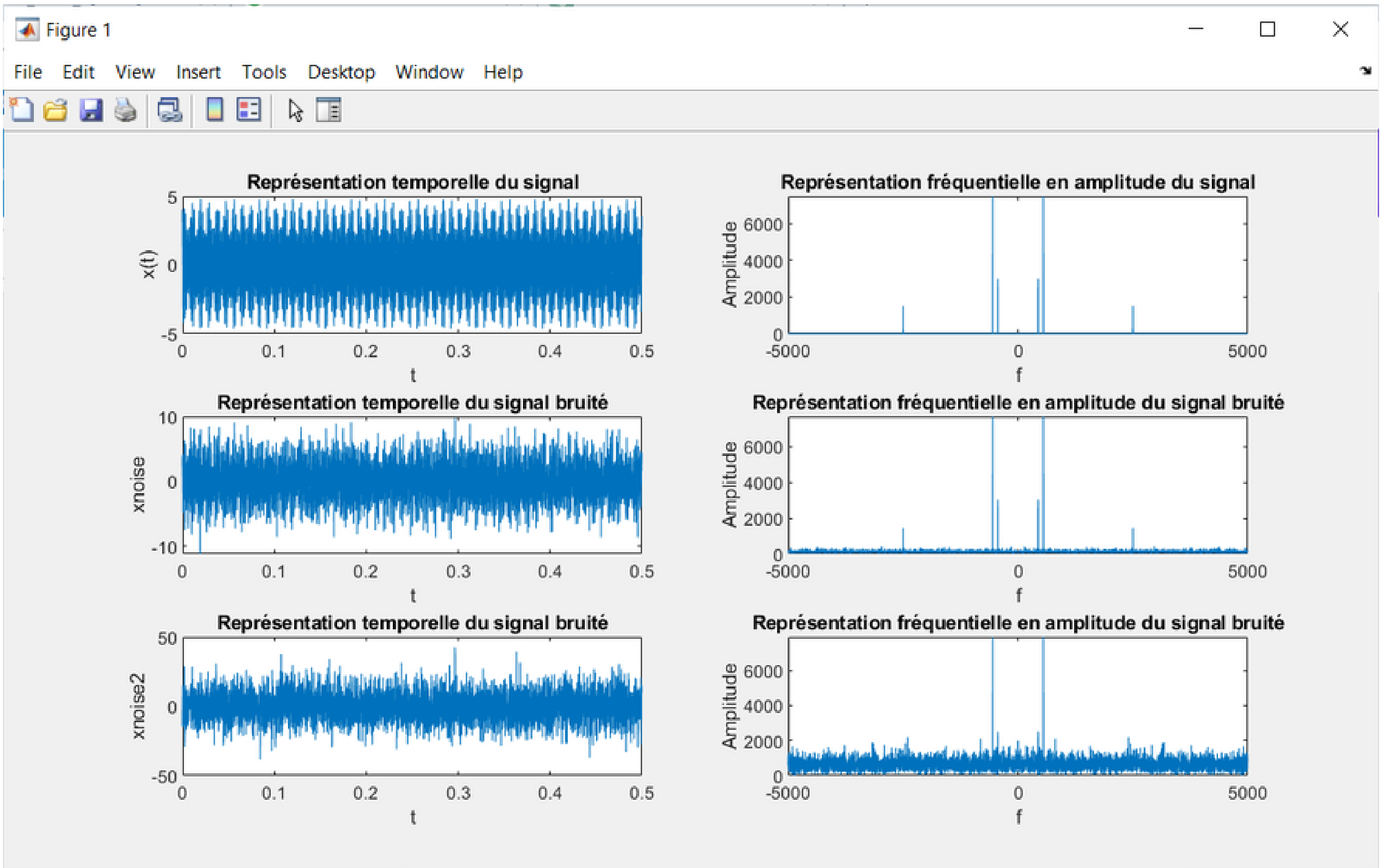


bruité

Nous remarquons l'apparition de nouveaux piques dans le spectre fréquentiels, générés par le bruit, mai sl'information est toujours présentes car les autres piques de fréquences sont toujours là,

Si `xnoise2 = x+10*randn(size(t));`

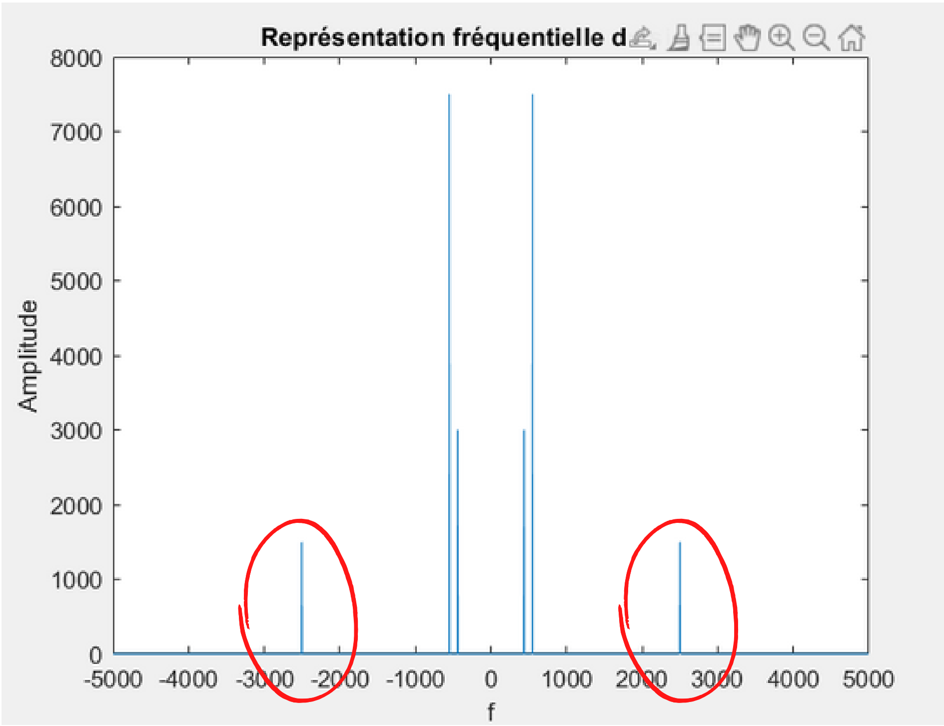




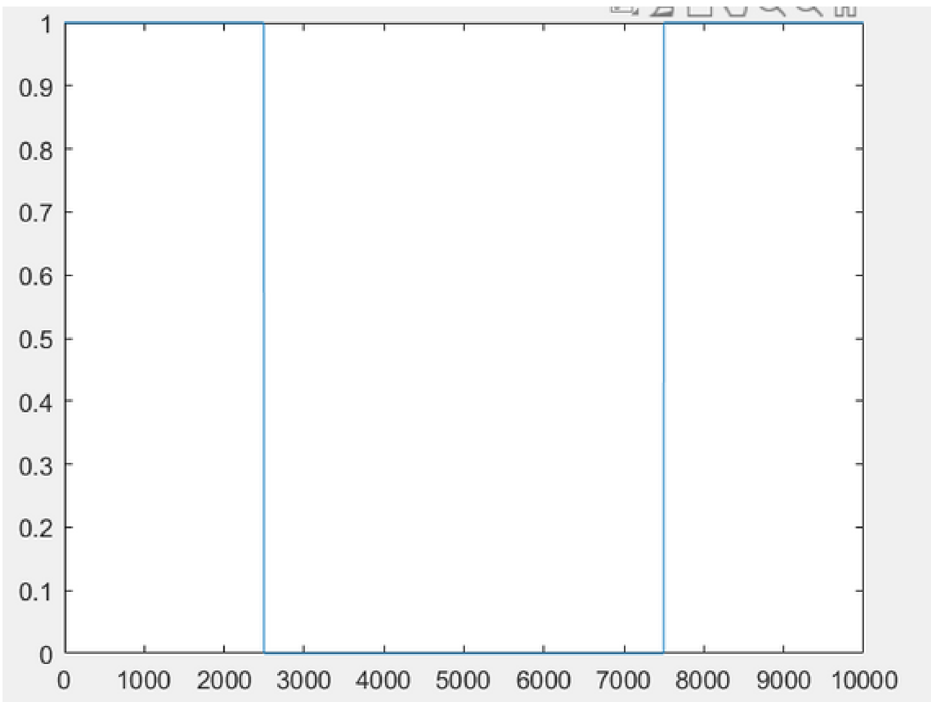
Lorsque nous augmentons l'intensité du bruit, nous remarquons que l'information devient de mois en moins visible, le spectre supposé nous aider à filtrer le signal, devient brouillé par le bruit, et du coup impossible de détecter les piques de fréquence du signal initial.

5- Filtrage du signal

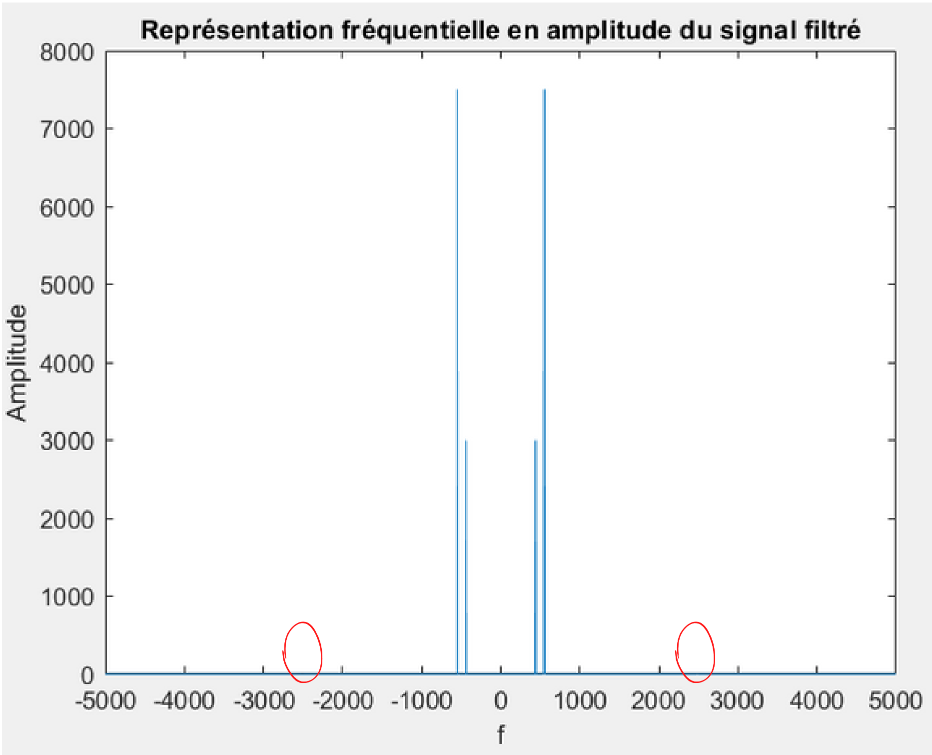
Nous procéderons par la suite à l'élimination des hautes fréquences supérieurs à 2500 Hz, pour cela nous allons concevoir un filtre passe-bas



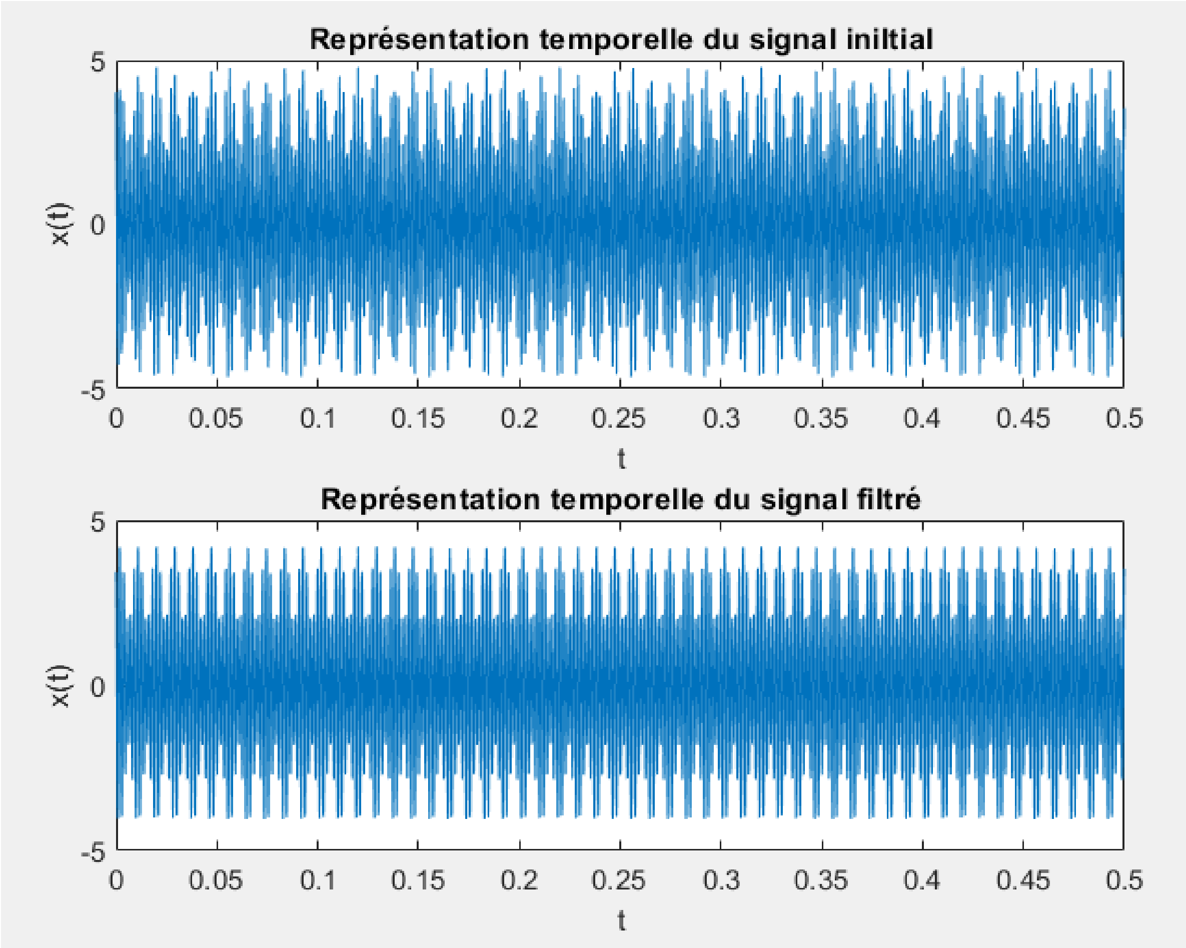

```
%Conception du filtre

fc=2500 % fréquence de coupure
pass_bas=zeros(size(x));
index_fc = ceil((fc*N)/fe);
pass_bas(1:index_fc)= 1;
pass_bas(N-index_fc+1:N) = 1;
xlabel('f');
ylabel('Amplitude')
title('Filtre pass-bas');
%plot(f,pass_bas);
```



```
sign_freq = pass_bas.*y;
%plot(f,sign_freq);
filtered_sign = ifft(sign_freq,"symmetric");
plot(fshift,fftshift(abs(fft(filtered_sign))));
xlabel('f');
ylabel('Amplitude');
title('Représentation fréquentielle en amplitude du signal filtré');
```



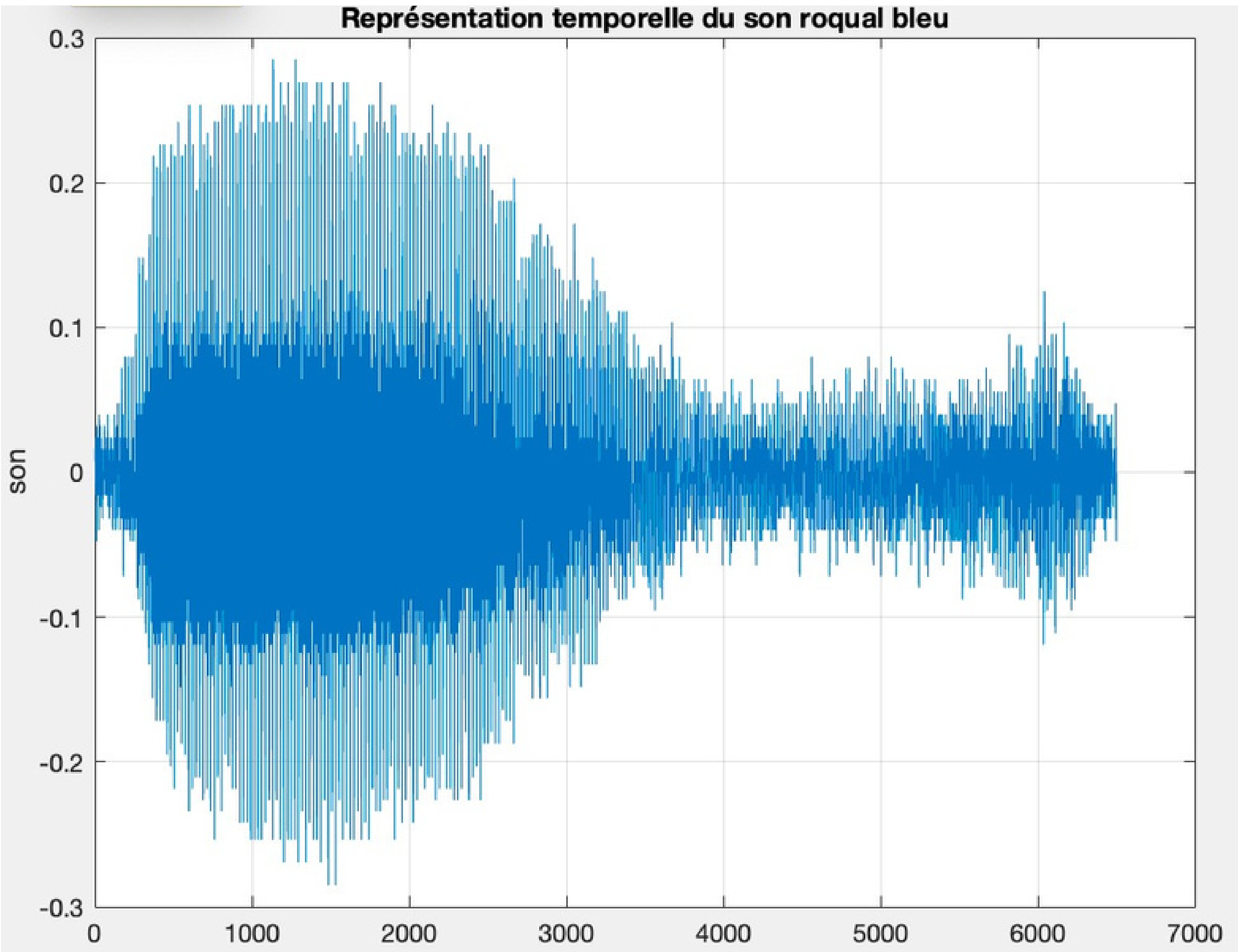
Nous remarquons la disparition du piques de la fréquence 2500 Hz grâce au filtrage que nous avons effectué



Après l'application du filtrage, nous pouvons remarquer que qu'une partie du bruit créé par les hautes fréquences a bel et bien été supprimé, et que l'allure du signal s'est nettement améliorée.

Analyse fréquentielle du chant du rorqual bleu

```
s='bluewhale.wav';
[son,Fe]=audioread(s);
sound(son,Fe);

son1 = son(2.45e4: 3.10e4);
sound(son,Fe);

plot(son1);
xlabel('t');
ylabel('son');
title('Représentation temporelle du son roqual bleu')
grid on
```



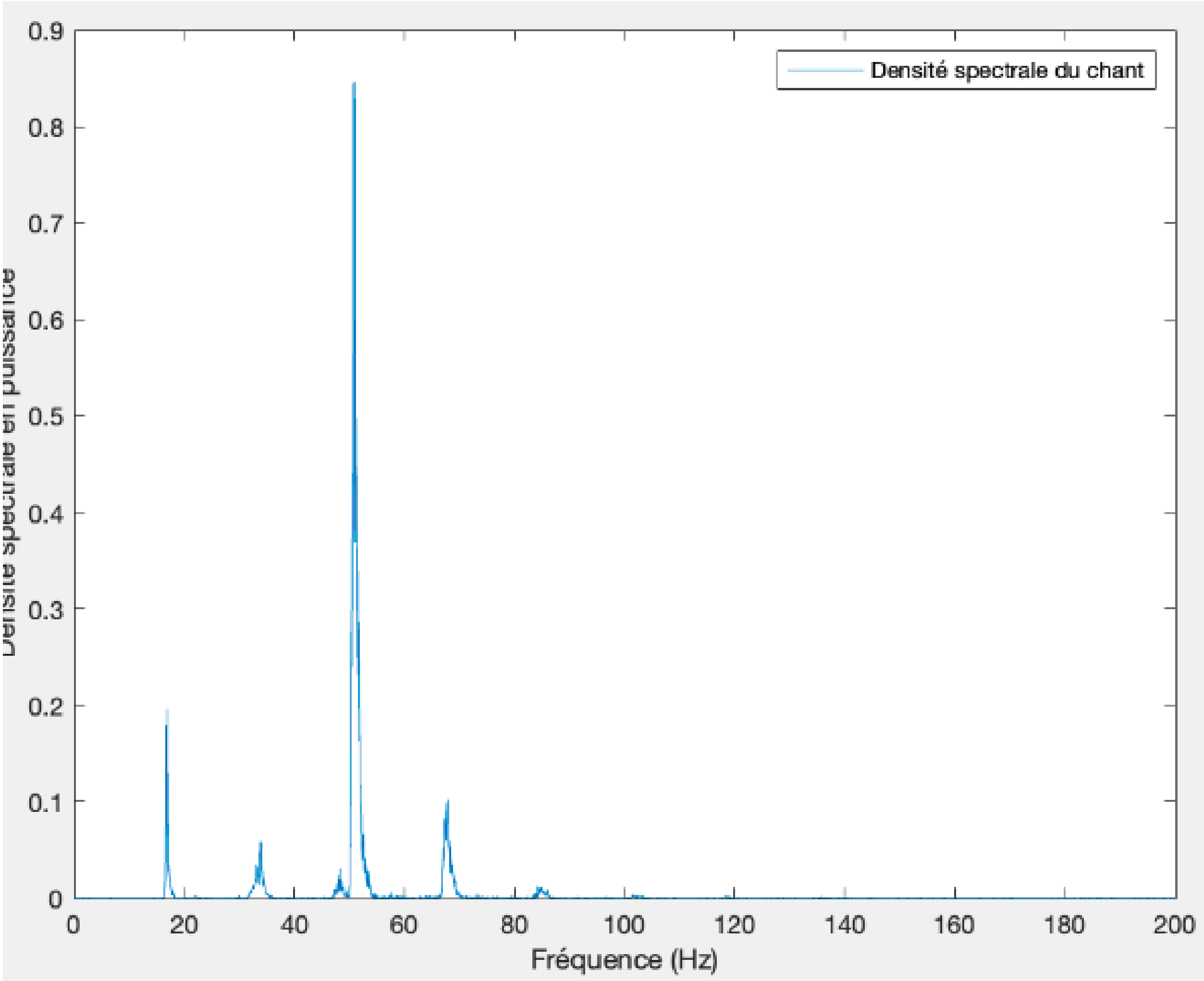
```
fourier = fft(son1); % transformation de fourier rapide sur le chant

Densite_spectrale = abs(fourier).^2/N; %Densité spectrale du Chant

f = (0:floor(N/2))*(Fe/N)/10;
plot(f,Densite_spectrale(1:floor(N/2)+1));
legend("Densité spectrale du chant");
xlabel("Fréquence (Hz)");
ylabel("Densité spectrale en puissance");
```

La première ligne utilise la fonction fft() pour effectuer une transformée de Fourier rapide sur le fichier audio "son1" et stocker le résultat dans la variable "fourier".

La deuxième ligne calcule la densité spectrale de puissance (DSP) en utilisant la formule DSP = abs(fourier)^2 / N. La fonction abs() prend la valeur absolue des éléments de "fourier" et la fonction^2 les éléve au carré, ceci est fait pour obtenir une mesure de l'intensité de chaque fréquence dans le signal. La variable N est le nombre d'échantillons dans le signal.

La troisième ligne crée un vecteur "f" contenant les fréquences en Hertz correspondant aux éléments de "fourier", en utilisant la formule f = (0:floor(N/2))*(Fe/N)/10, où Fe est la fréquence d'échantillonnage du signal.

En gros, ce code utilise la transformée de Fourier pour analyser les fréquences présentes dans un fichier audio, puis trace les résultats sous forme de densité spectrale de puissance en fonction de la fréquence pour une meilleure visualisation.


Densité spectrale du chant
Densité spectrale en puissance
Fréquence (Hz)
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
20
40
60
80
100
120
140
160
180
200

Conclusion

Nous pouvons conclure, grâce à ce TP que le Spectre fréquentiel et d'une très grande importance lorsque nous voulons extraire l'information du signal temporel. Le passage du domaine temporel au domaine fréquentiel nous permet de détecter les fréquence qui constitue l'information d'origine et aussi de détecter les fréquence du bruit pour pouvoir les éliminer par la suite grâce à un filtre. Nous obtiendront par la suite le signal d'origine non-bruité.