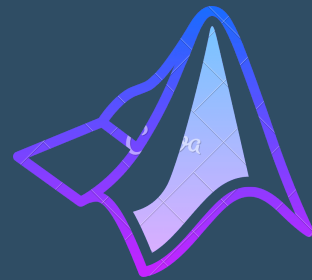


Compte rendu

Traitement de signal

TP 1



Préparé par:
Ilias Manadir

Encadré par :
Mr Alaa

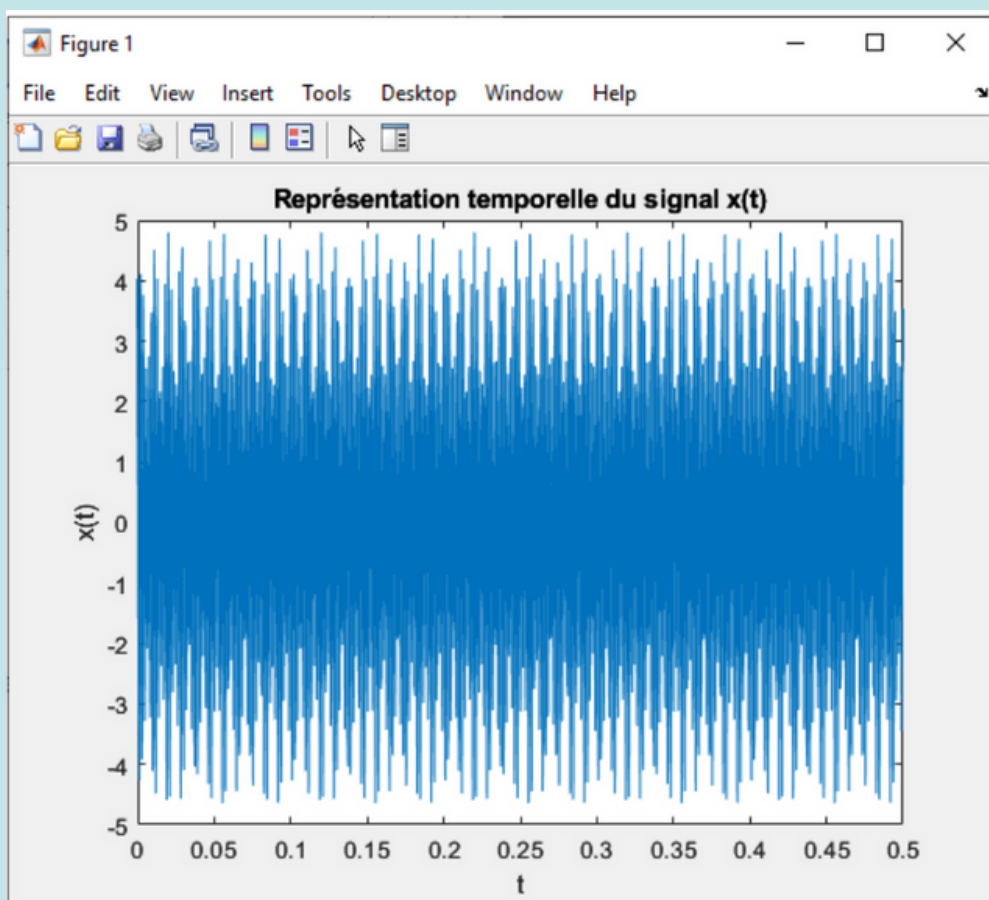
REPRÉSENTATION TEMPORELLE ET FRÉQUENTIELLE :

REPRÉSENTATION TEMPORELLE :

Dans la première partie, on essaie de tracer le signal dans le domaine temporelle. On utilise un signal qui se compose d'une somme de trois sinusôïdes de fréquences différentes : 440 Hz, 550 Hz et 2500 Hz.

```
fe = 10000; %Fréquence d'échantillonnage
te = 1/fe; %Période d'échantillonnage
N = 5000; %Nombre d'échantillons

t = 0:te:(N-1)*te; %Intervalle
x = 1.2*cos(2*pi*440*t+1.2) + 3*cos(2*pi*550*t) + 0.6*cos(2*pi*2500*t); %signal périodique x(t)
```

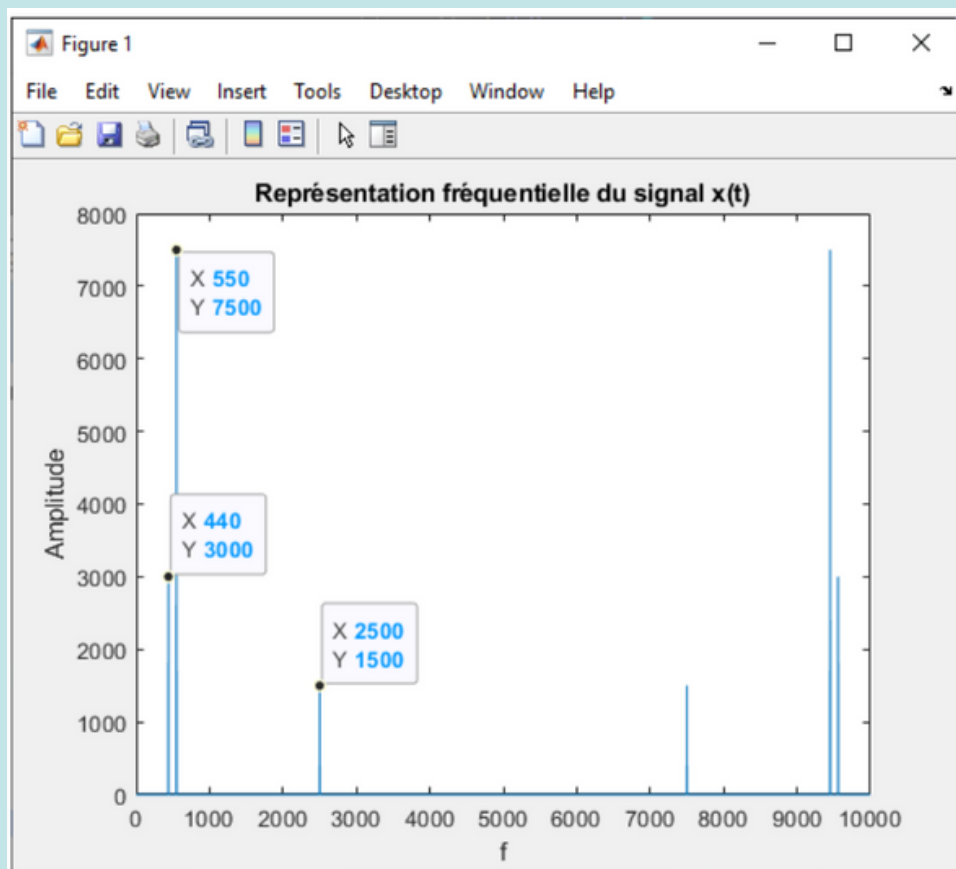


Représentation temporelle du signal

REPRÉSENTATION FRÉQUENTIELLE :

La fonction `fft` génère une fonction complexe (spectre discret). La fonction a la possibilité d'effectuer une transformée de Fourier pour passer du domaine temporel au domaine fréquentiel ce qui est l'objectif de cette partie. Dans le graphe affiché ci-dessous, on remarque que les pics sont symétriques par rapport à la fréquence $f_e/2$ qui est égale à 5000Hz. C'est la symétrie conjuguée

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

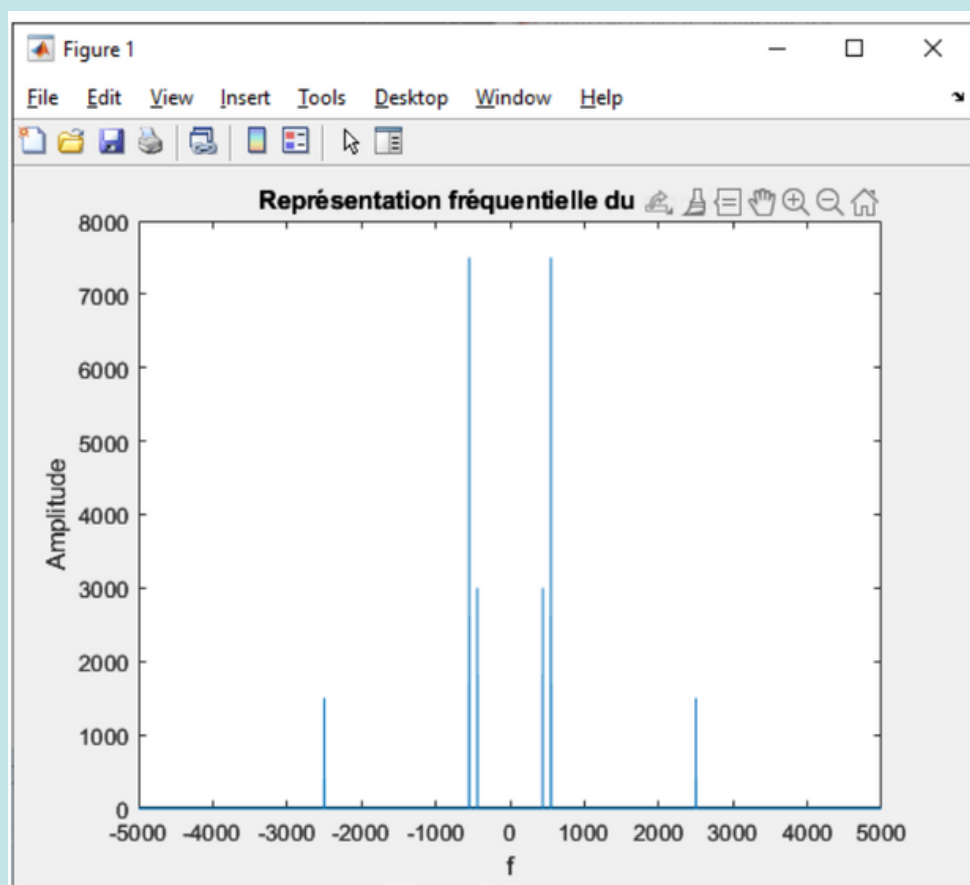


Représentation fréquentielle du signal

REPRÉSENTATION FRÉQUENTIELLE AVEC FFTSHIFT :

La commande `fftshift` dans Matlab permet de décaler la fréquence nulle au centre de l'échantillon de fréquences pour une transformée de Fourier rapide (FFT) ou une transformation inverse de Fourier rapide (IFFT). Cela permet de faciliter l'interprétation des résultats de la FFT pour certains types d'analyses, comme les spectres d'amplitude.

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

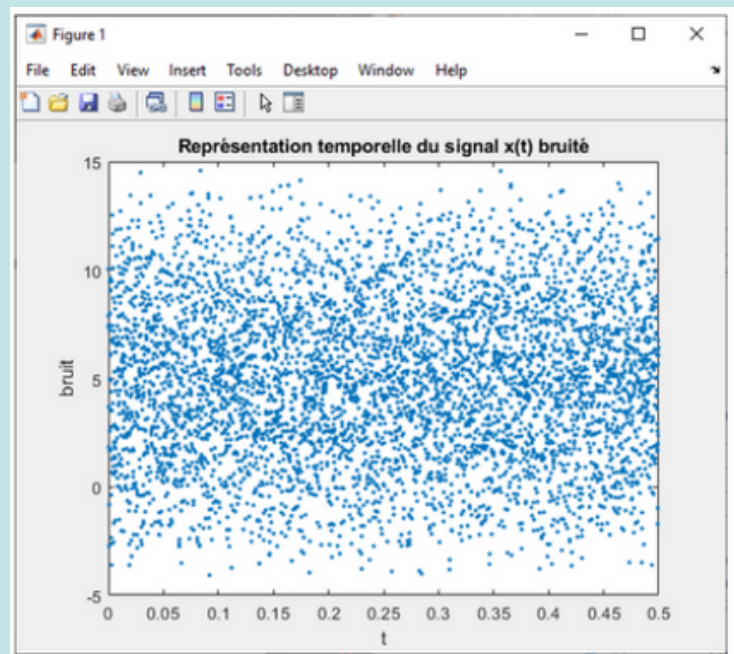
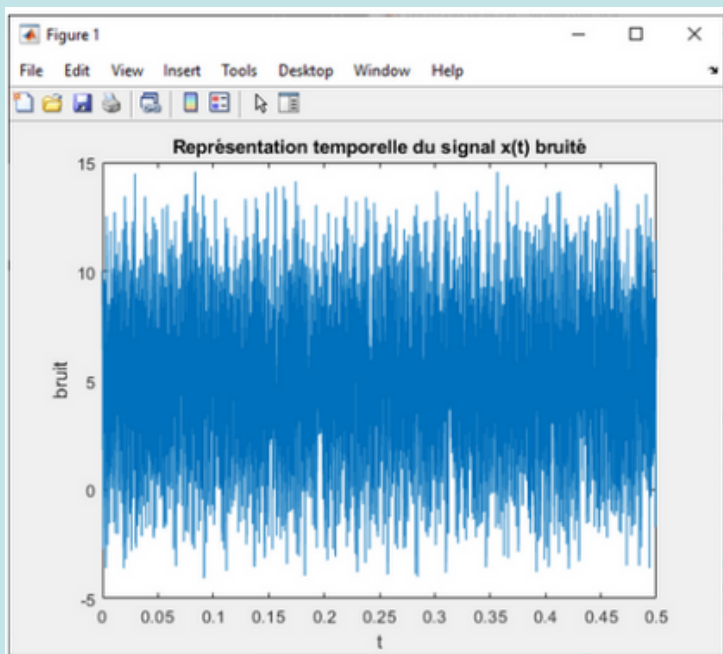


Représentation fréquentielle du signal (fftshift)

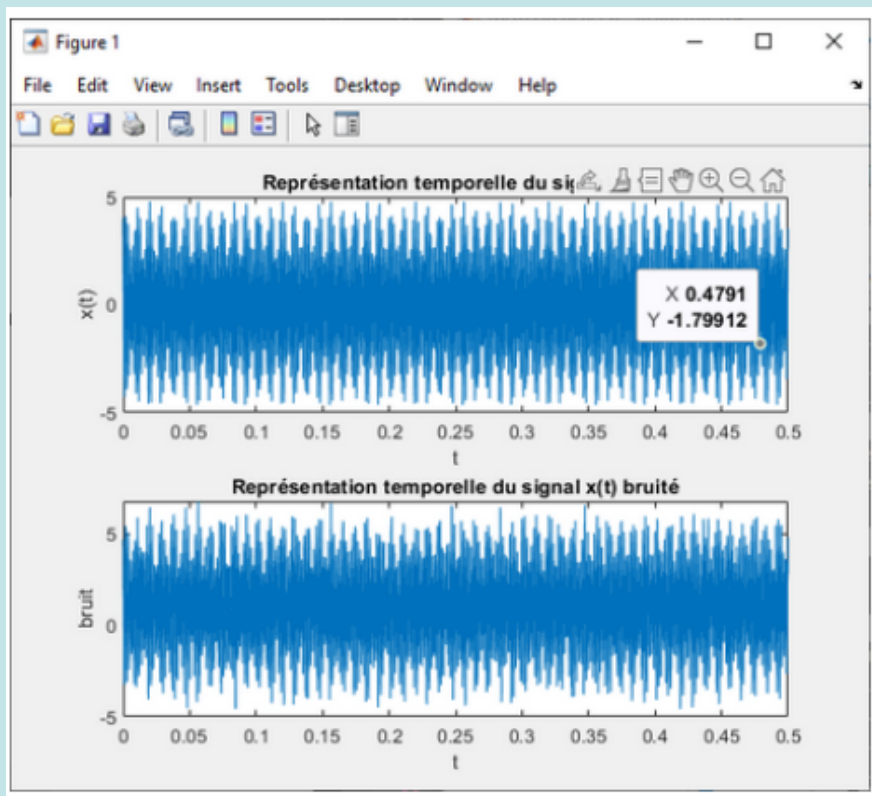
REPRÉSENTATION FRÉQUENTIELLE AVEC FFTSHIFT :

La commande `fftshift` dans Matlab permet de décaler la fréquence nulle au centre de l'échantillon de fréquences pour une transformée de Fourier rapide (FFT) ou une transformation inverse de Fourier rapide (IFFT). Cela permet de faciliter l'interprétation des résultats de la FFT pour certains types d'analyses, comme les spectres d'amplitude.

```
xnoise = x+2*rand(size(t));  
sound(xnoise)  
  
plot(t,xnoise)
```



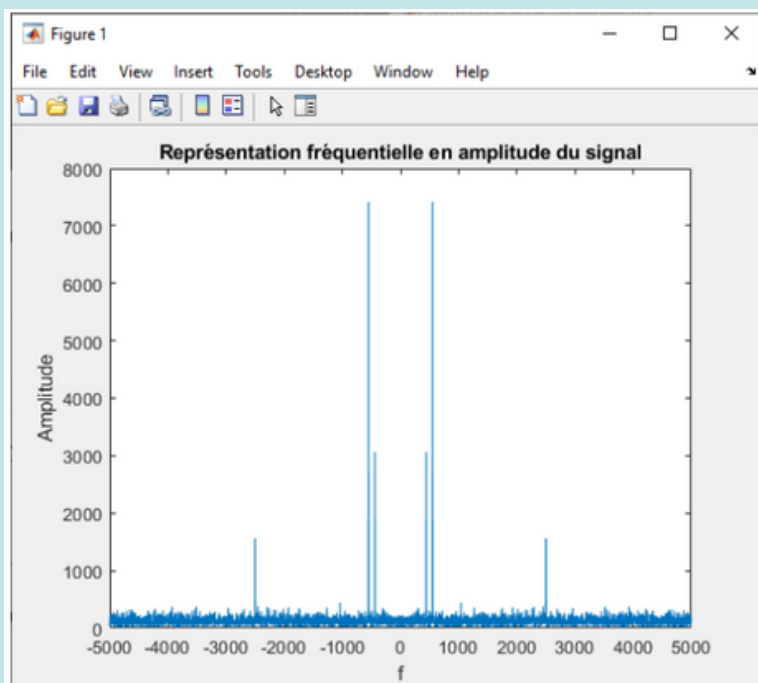
Représentation temporelle du signal bruité



Voici la différence entre les deux signaux

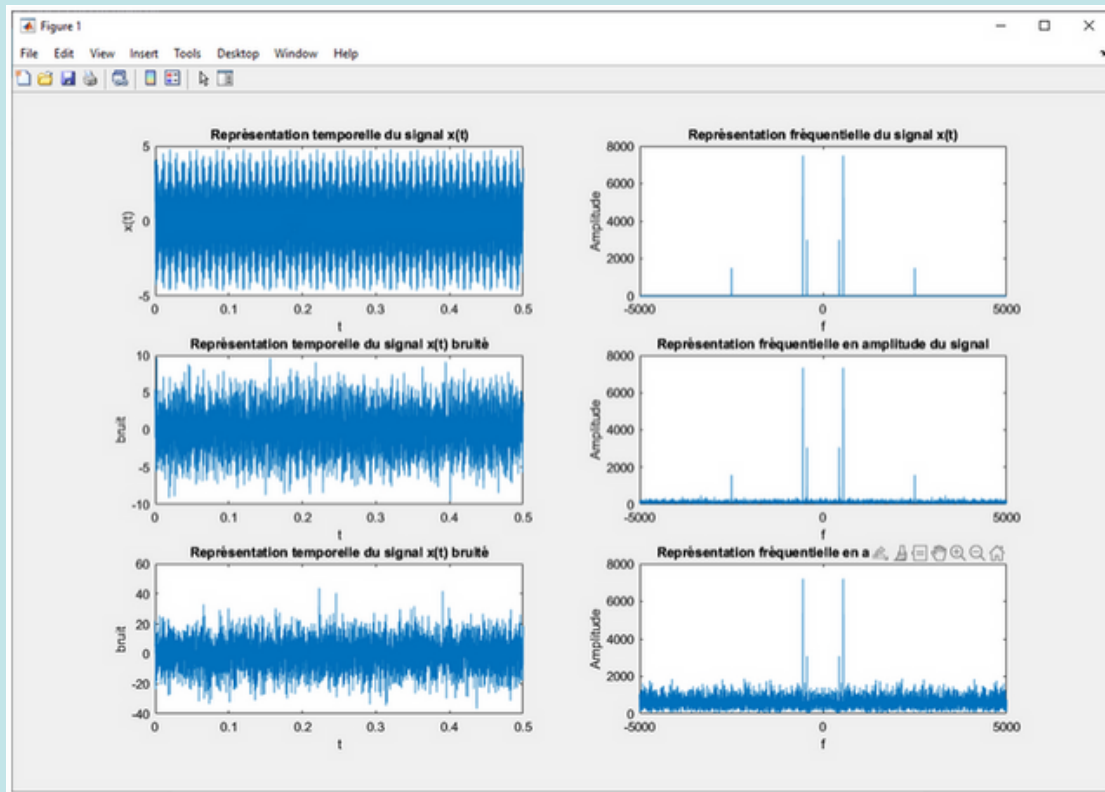
On remarque que le bruit présente des propriétés statiques, car la plupart des valeurs sont tendent vers 0 et sont comprises entre -5 et 5, ce qui laisse la moyenne constante.

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



**Représentation
fréquentielle en
amplitude du signal**

On remarque l'apparition des nouveaux piques dans le spectre fréquentielle générés par le bruit. L'information n'a pas disparu car on peut visualiser les piques d'origine. Dans le cas d'augmentation de l'intensité du bruit, nous constatons que l'information devient de moins en moins visible, alors il sera impossible de détecter les piques de fréquence du signal initial.

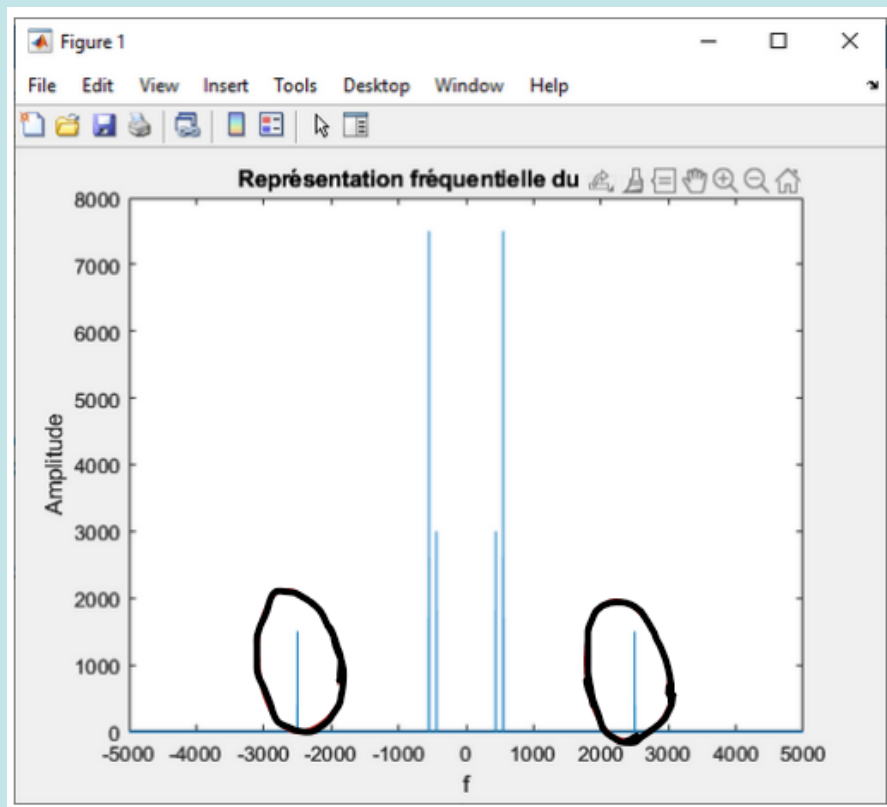


Représentation temporelle et fréquentielle des trois signaux traitées

FILTRAGE DU SIGNAL :

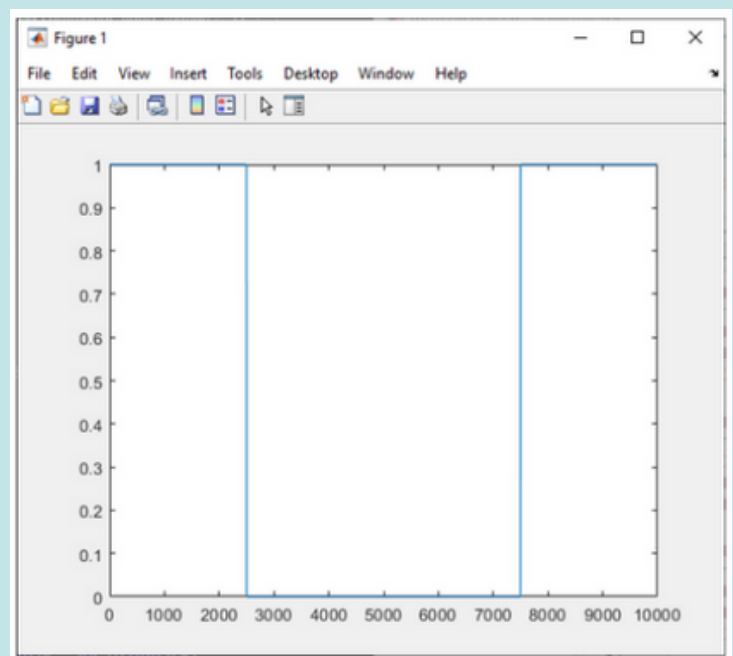
1. CONCEPTION DU FILTRE:

Dans cette étape, on essaiera de filtrer le signal $x(t)$ des hautes fréquences supérieure à 2500Hz, donc on utilise un filtre passebas.



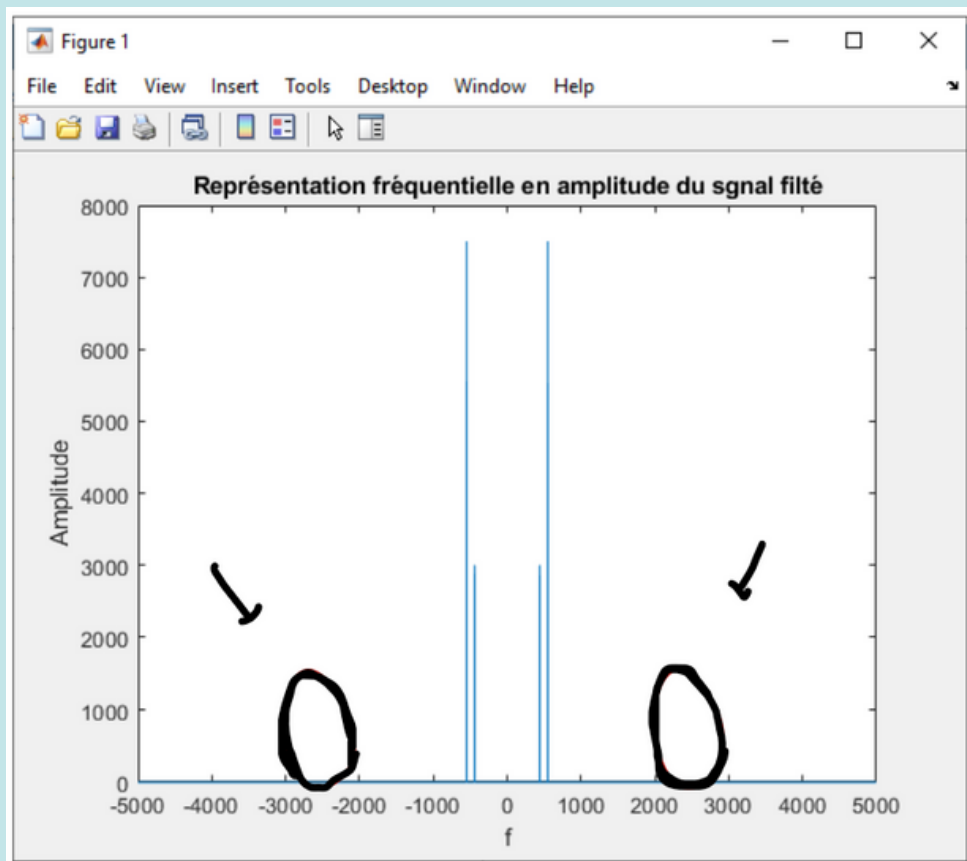
```
fc = 2500; %fréquence de coupure
pass_bas = zeros(size(x));
index_fc = ceil((fc*N)/fe);
pass_bas(1:index_fc) = 1;
pass_bas(N-index_fc+1:N) = 1;
xlabel('f')
ylabel('Amplitude')
title('Filtre pass-bas')

plot(f,pass_bas)
```



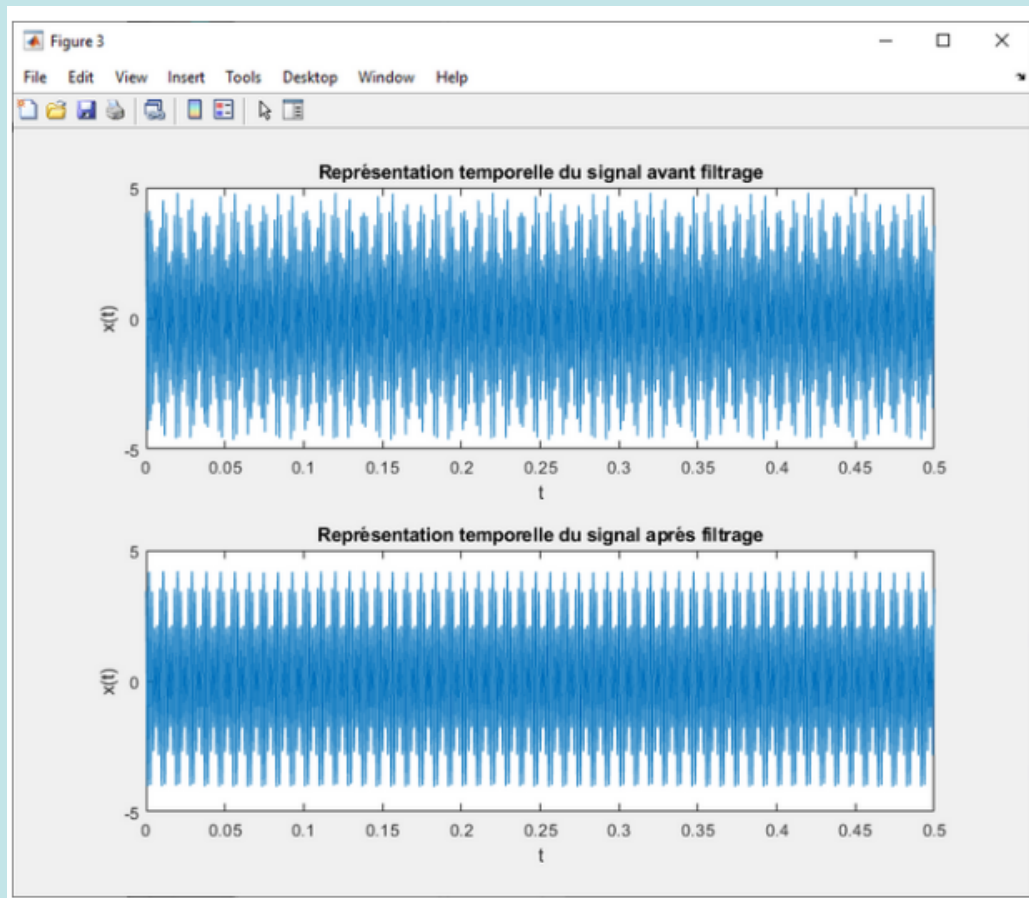
2. APPLICATION DU FILTRAGE:

```
z = pass_bas.*y;%filtre en fct de fréquence  
sign_filtre = ifft(z,"symmetric");  
  
plot(fshift,fftshift(abs(fft(sign_filtre))))
```



Après l'application du filtrage, on remarque que les piques de fréquence 2500 Hz on disparu.

3. COMPARAISON DU SIGNAL AVANT ET APRÈS LE FILTRAGE:



On peut remarquer la différence entre les deux signaux; le signal après filtrage est plus clair grâce au filtrage appliqué qui a supprimé le bruit causé par les hautes fréquences.

CONCLUSION :

On peut conclure de ce TP que le spectre fréquentiel joue un rôle très important dans la détection de bruit. Le passage du domaine temporel au domaine fréquentiel nous a permis de visualiser l'information essentielle plus les fréquences du bruit qu'on éliminer grâce à un filtre. Après l'application du filtre on peut bien voir la différence entre un signal bruité et un signal non-bruité.