compte rendu du :

TP1- Analyse spectrale d'un signal Transformée de Fourier discrète

Réalisé par : Hiba QOUIQA

filière : IA

Encadré par : Mr. Alae AMMOUR

2022-2023

Sommaire :

1. Buts 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.

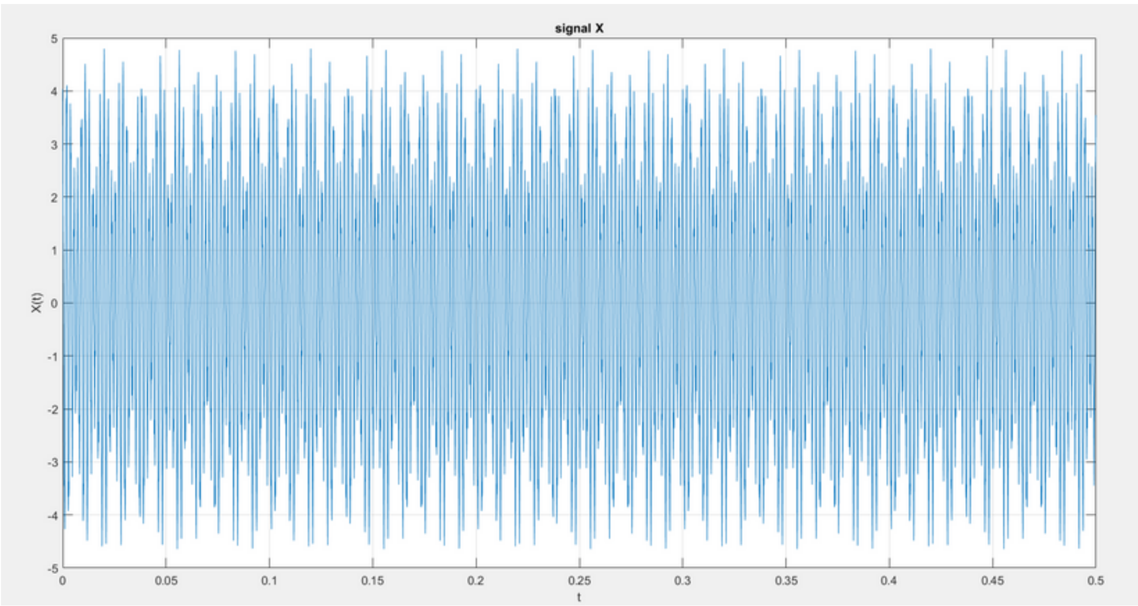
2. Représentation temporelle et fréquentielle:

Considérons un signal périodique x(t) constitué d'une somme de trois sinusoïdes de fréquences 440Hz, 550Hz, 2500Hz:

$$\mathbf{x}(\mathbf{t}) = \mathbf{1.2cos}(\mathbf{2pi440t} + \mathbf{1.2}) + \mathbf{3cos}(\mathbf{2pi550t}) + \mathbf{0.6cos}(\mathbf{2pi2500t})$$

1- la représentation temporelle du signal X :

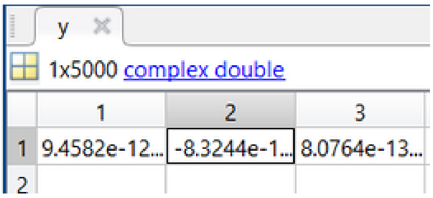
```
%%
fe=10000; %la fréquence d'echantillonnage
Te=1/fe; % la période d'echantillonnage
N=5000; % nbr d'echantillons


t=0:Te:(N-1)*Te; % le pas c'est une période d'échantillonnage

x=1.2*cos(2*pi*440*t+1.2)+3*cos(2*pi*550*t)+0.6*cos(2*pi*2500*t);


plot(t,x)
grid on
title('signal x')
xlabel('t')
ylabel('X(t)')
```



2- la Tf du signal X(t):

on remarque que la fonction fft génère un spectre discret qui est une fonction complexe qui a des valeurs dans la partie réelle et la partie imaginaire

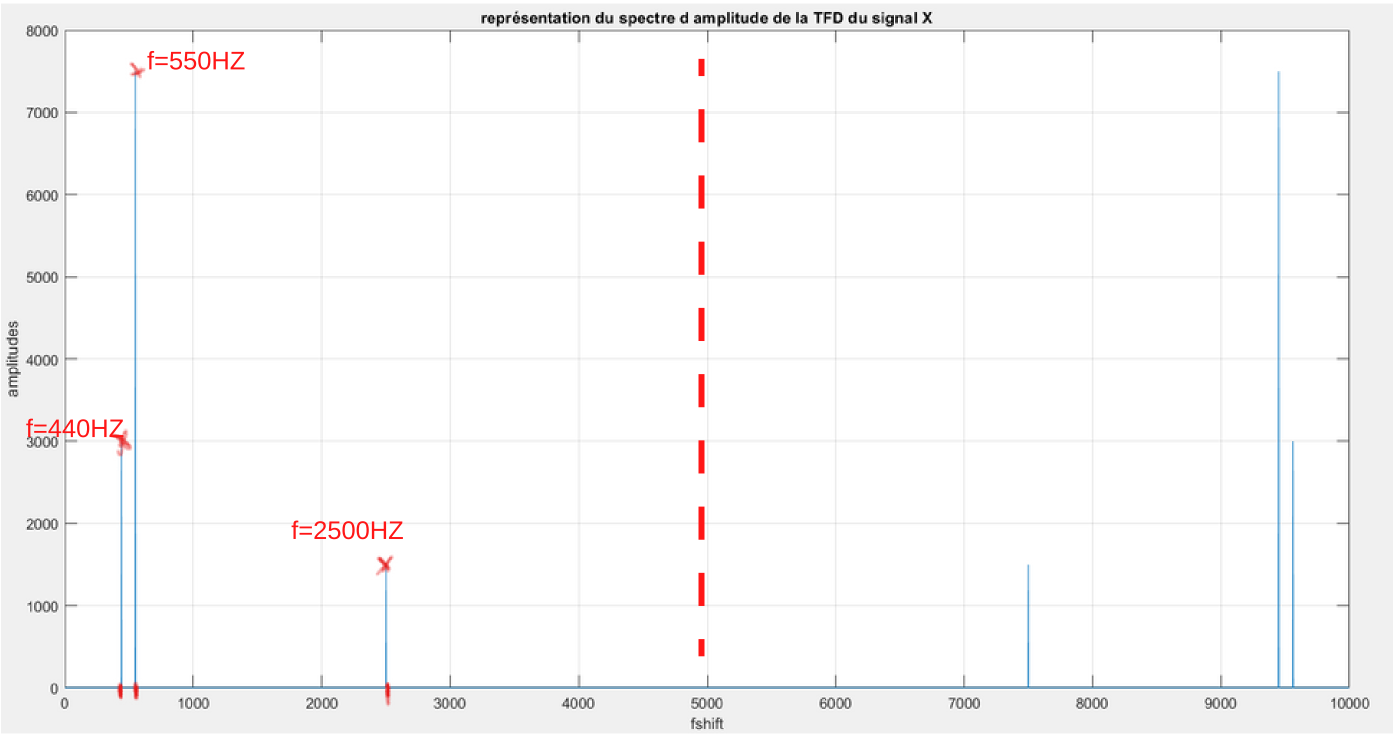
```
y=fft(x);
```



- le script Matlab:

```
% Representation fréquentielle

fshift = 0: fe/N: (N-1)*(fe/N); % le pas de discrétisation est fe/N
% est ça génère aussi 5000 echantillons

 y=fft(x)

 plot(fshift,abs(y)); % la fct est pour calculer les amplitudes
 grid on
 title('représentation du spectre d amplitude de la TFD du signal X')
 xlabel('fshift')
 ylabel('amplitudes')
```

- la figure:



- Remarques:

 - on remarque que dans la figure apparait 6 piques , les trois premiers piques reprèsentent les trois fréquences qui constituent le signal X(t) : 440Hz, 550Hz, 2500Hz.

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

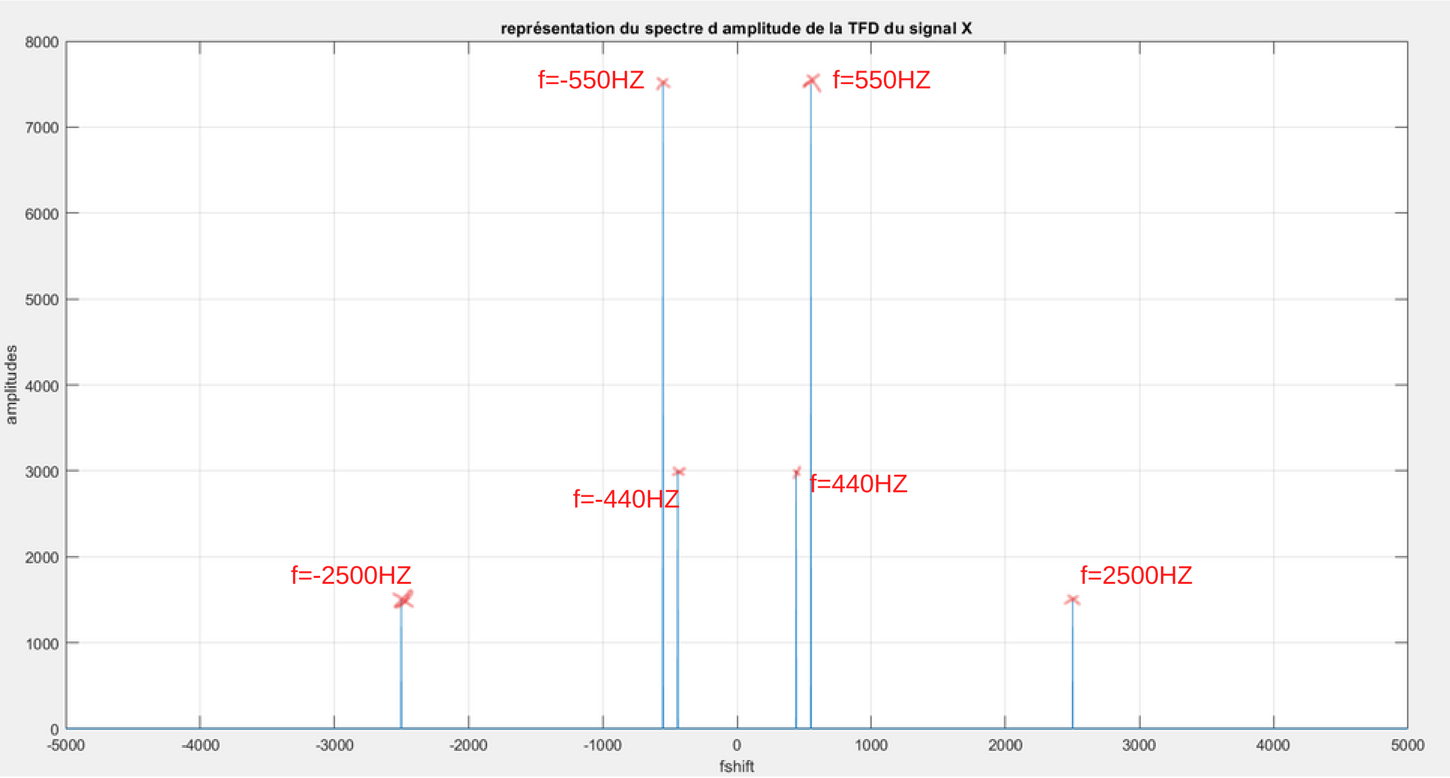
2- la Tf du signal X(t):

on remarque que la fonction fftshift effectue un décalage circulaire centré sur zéro du spectre en amplitude obtenu par la commande fft. c-à-d un spectre bilateral.

- le script Matlab:

```
% Representation fréquentielle

%%fshift = 0: fe/N: (N-1)*(fe/N); % le pas de discrétisation est fe/N
fshift2 = (-N/2:(N/2)-1)*(fe/N);
% est ça génère aussi 5000 echantillons

 y=fft(x)

 %plot(fshift,abs(y)); % la fct abs est pour calculer les amplitudes
 plot(fshift2,fftshift(abs(y))); % la fct abs est pour calculer les amplitudes
 grid on
 title('représentation du spectre d amplitude de la TFD du signal X')
 xlabel('fshift')
 ylabel('amplitudes')
```

- la figure:



Un bruit correspond à tout phénomène perturbateur gênant la transmission ou l'interprétation d'un signal.

4. on génère un bruit gaussien dans le signal x(t) :

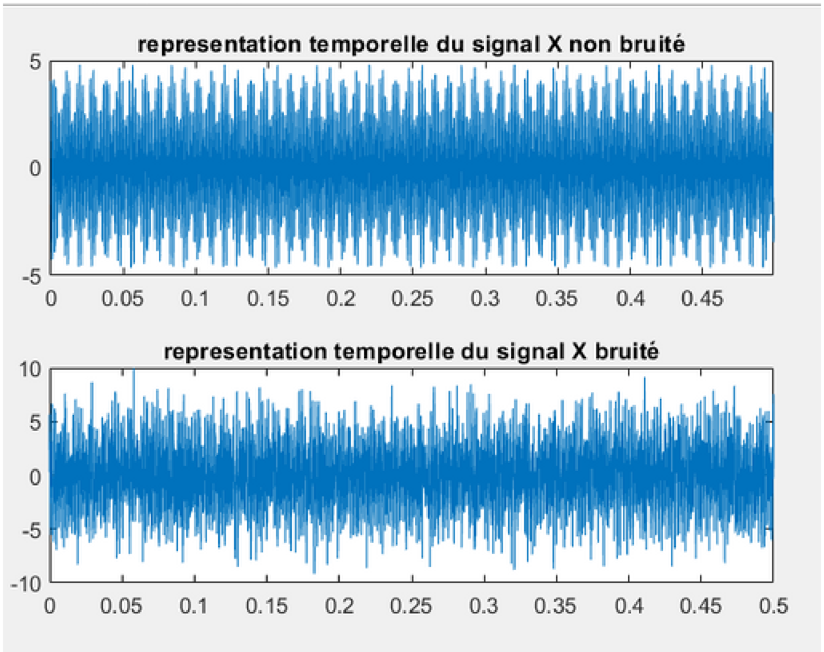
un bruit gaussien est un bruit aléatoire qui suit la loi de Gauss.

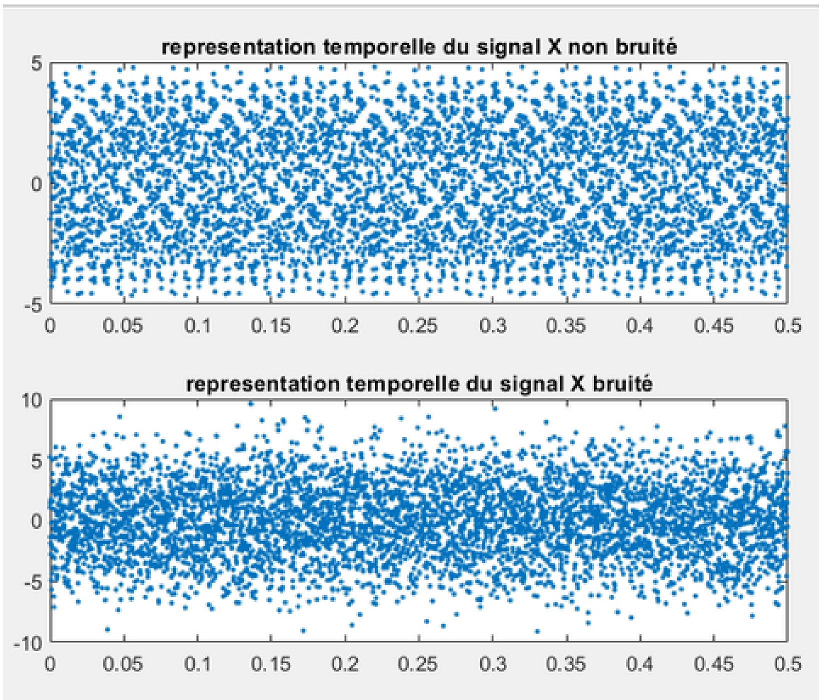
- le script Matlab:

```
% Signal bruité

xbruit =  x + 2*randn(size(x));
subplot(2,1,1)
plot(t,x)
title('representation temporelle du signal X non bruité');
subplot(2,1,2)
plot(t,xbruit)
title('representation temporelle du signal X bruité');
```

la fct size() est pour générer un vecteur de meme taille que x

- la figure:





- Remarque:

- la majorité des valeurs du bruit sont centrées sur 0 entre -3 et 3.

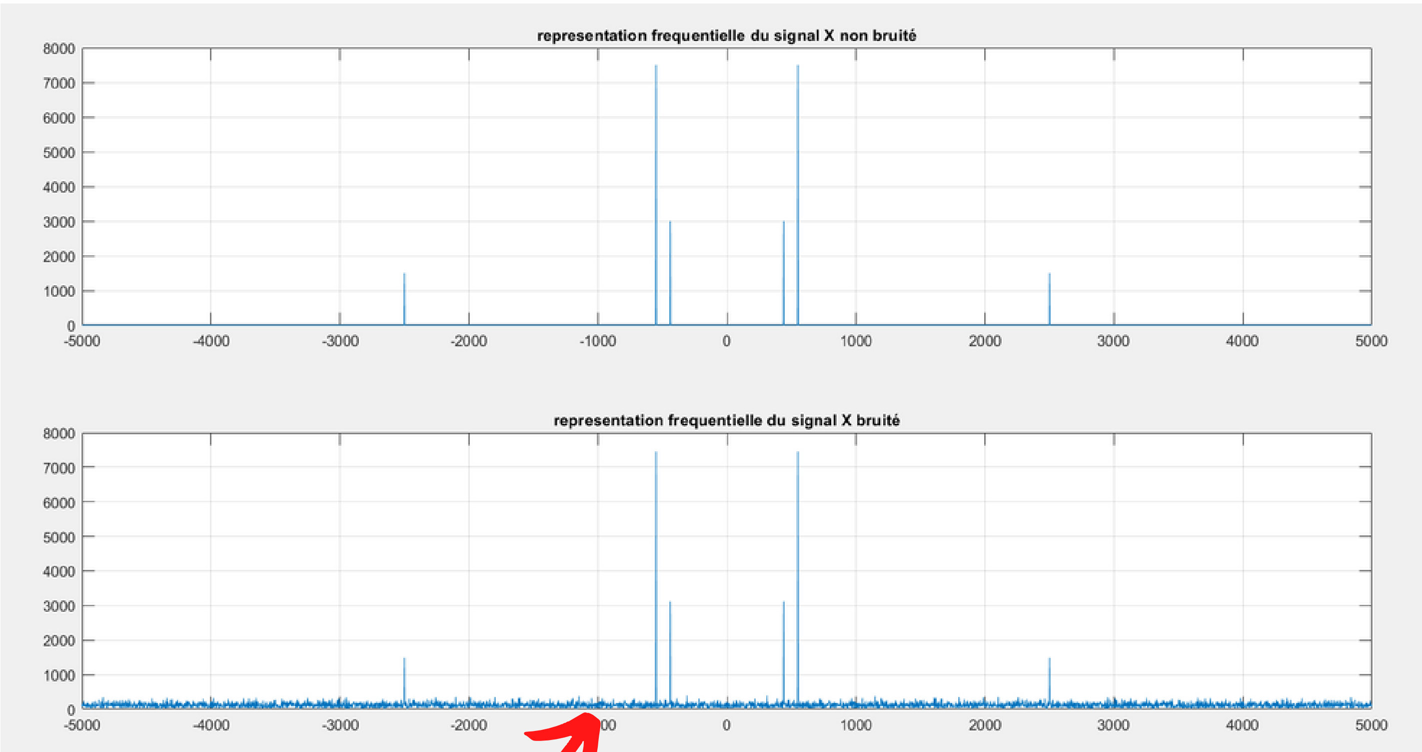
5.quand on utilise la fct sound on entend le bruit dans le signal x .

6.

- le script Matlab:

```
% Signal bruité

xbruit =  x + 2*randn(size(x));
ybruit = abs(fft(xbruit));
subplot(2,1,1)
plot(fshift2,fftshift(abs(y)));
title('representation frequentielle du signal X non bruité');
grid on
subplot(2,1,2)
plot(fshift2,fftshift(ybruit));
title('representation frequentielle du signal X bruité');
grid on
```

- la figure:



- Remarque:

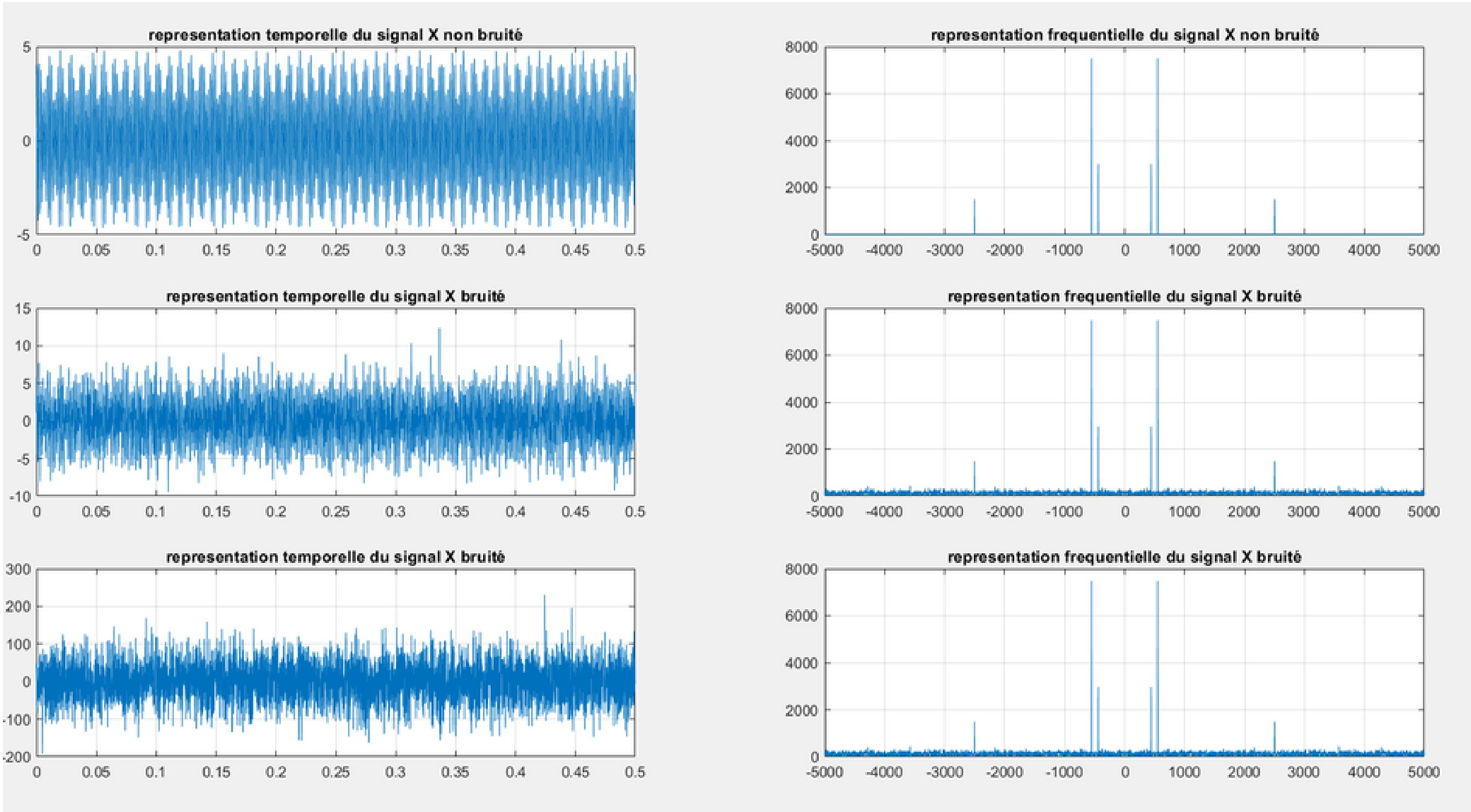
- on remarque dans le signal bruité la présence d'autre petits piques.
- ça ne signifie pas qu'on a perdu complétement l'information car les autres piques des fréquences constituant le signal d'origine X(t) sont présents mais cette fois avec d'autres petits piques.

7.

- le script Matlab:

```
xbruit =  x + 2*randn(size(x));
ybruit = abs(fft(xbruit));

xbruit2 = x + 50*randn(size(x));
ybruit2 = abs(fft(xbruit));

subplot(3,2,1)
plot(t,x)
title('representation temporelle du signal X non bruité');
grid on

subplot(3,2,2)
plot(fshift2,fftshift(abs(y)));
title('representation frequentielle du signal X non bruité');
grid on

subplot(3,2,3)
plot(t,xbruit)
title('representation temporelle du signal X bruité');
grid on

subplot(3,2,4)
plot(fshift2,fftshift(abs(ybruit)));
title('representation frequentielle du signal X bruité');
grid on

subplot(3,2,5)
plot(t,xbruit2)
title('representation temporelle du signal X bruité');
grid on

subplot(3,2,6)
plot(fshift2,fftshift(abs(ybruit2)));
title('representation frequentielle du signal X bruité');
grid on
```

- les figures:



- Remarque:

 - on remarque que plus que l'intensité du bruit augmente plus qu'il devient difficile d'extraire l'information de la représentation temporelle par contre dans la representation fréquentielle c'est façile d'extraire l'information comme on observe dans les trois figures à gauche on peut façilement distinguer les piques des fréquences 440Hz , 550 Hz et 2500Hz des piques qu'a généré le bruit . D'ou il devient façile de réstituer le signal d'origine.

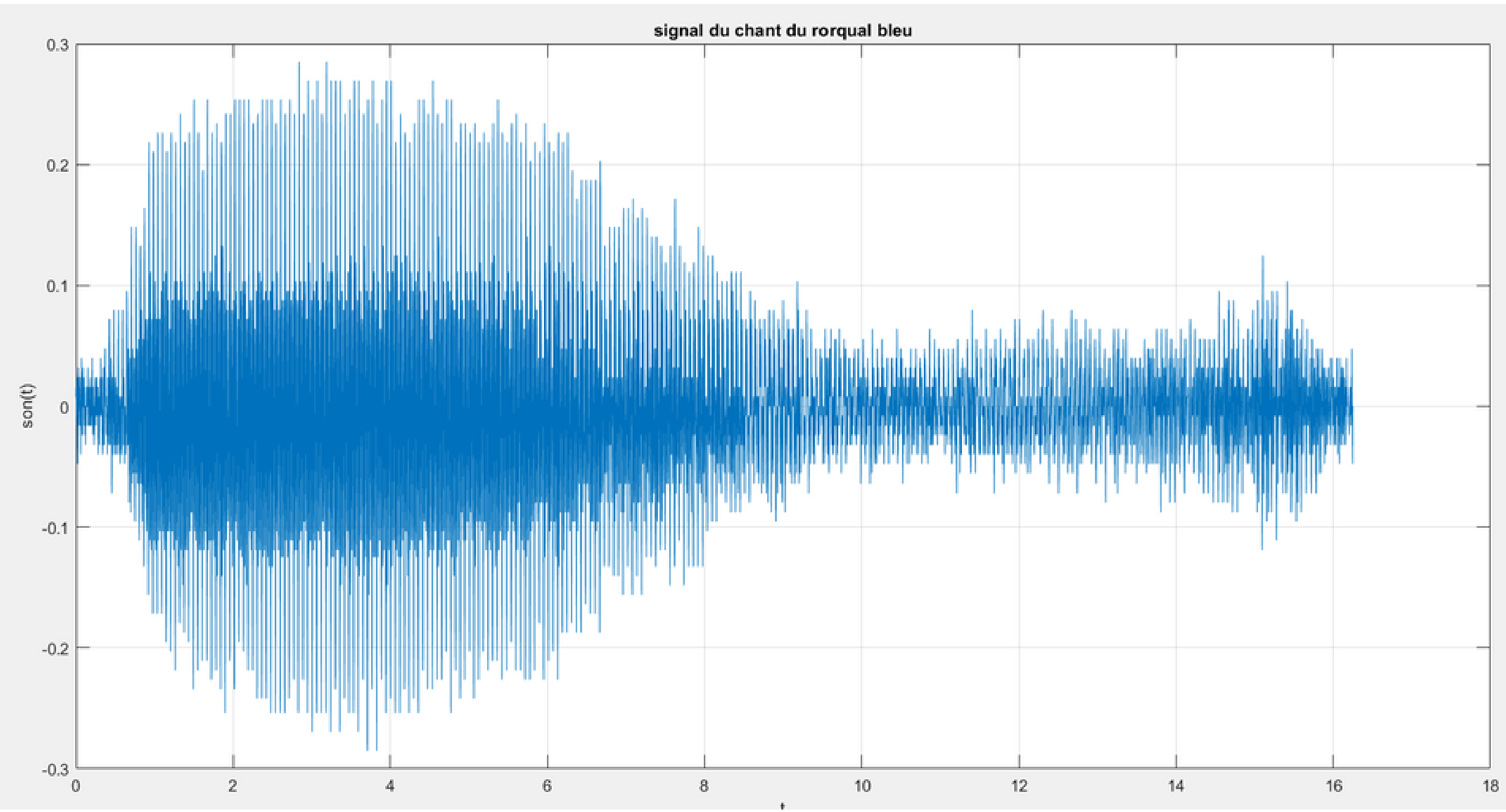
3.Analyse fréquentielle du chant du rorqual bleu

Il existe plusieurs signaux dont l'information est encodée dans des sinusoïdes. Les ondes sonores est un bon exemple. Considérons maintenant des données audios collectées à partir de microphones sous - marins au large de la Californie. On cherche à détecter à travers une analyse de Fourier le contenu fréquentiel d'une onde sonore émise pas un rorqual bleu.

```
%Question 1:
[whale,fe]=audioread("bluewhale.au");
son=whale(2.45e4:3.10e4);

%question 2:
sound(son,fe)
```

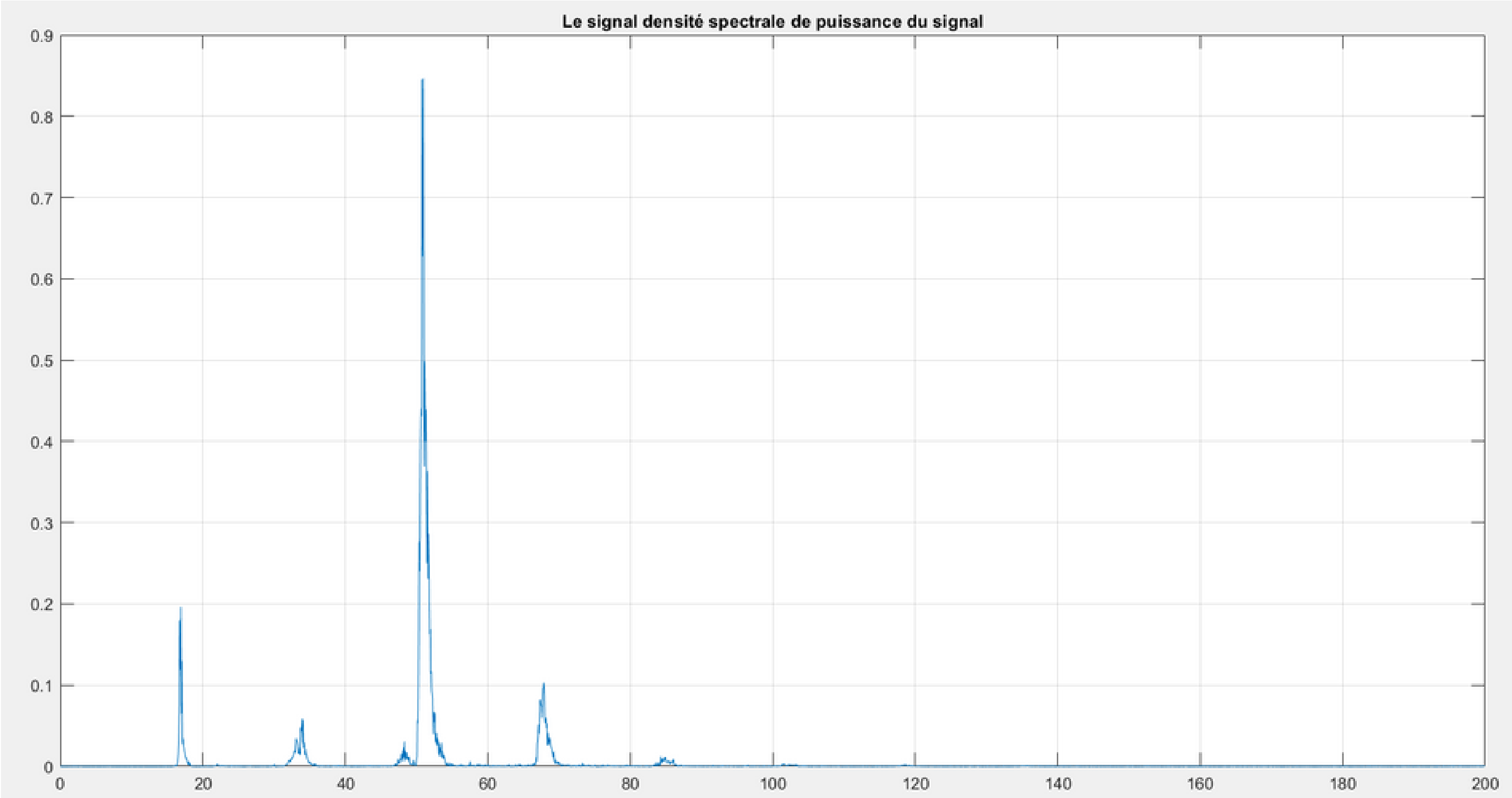
3.on trace le signal du son , suivant le programme ci-dessous :

```
%visualisation du signal
te=1/fe;%periode d'echantillonnage
N=length(son);%le nombre d'echantillons
t = (0:N-1)*(10*te);

plot(t,son)
title("signal du chant du rorqual bleu")
xlabel('t')
ylabel('son(t)')
grid on
```



Pour identifier les composantes fréquentielles de ce signal audio, on utilise la TDF .Dans certaines applications qui traitent de grandes quantités de données avec fft, il est courant de redimensionner l'entrée de sorte que le nombre d'échantillons soit une puissance de 2.
Pour cela on spécifie une nouvelle longueur de signal qui sera une puissance de 2 , comme il'est montré dans le programme suivant :

```
%%
%question 3:
x = abs(fft(son)).^2/N;
f = (0:floor(N/2))*(fe/N)/10;
plot(f,x(1:floor(N/2)+1));
title('Le signal densité spectrale de puissance du signal')
grid on
```



CONCLUSION :

D'après ce tp , on peut bien comprendre l'importance du passage du domaine temporel au domaine fréquentiel ; on appliquant la TFD sur le signal étudié on peut observer le spectre de phase ou d'amplitude de ce signal ce qui permet de filtrer toutes les fréquences qui le constituent et donc rendre façile d'éliminer le bruit et enfin réstituer le signal d'origine non-bruité ce qui a été impossible de le faire dans le domaine temporelle.