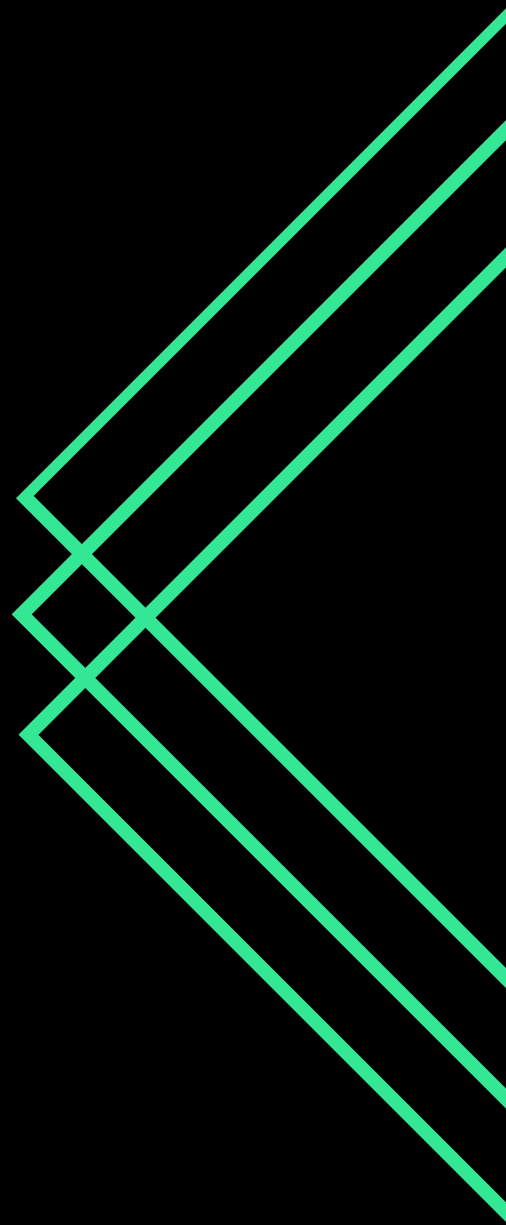
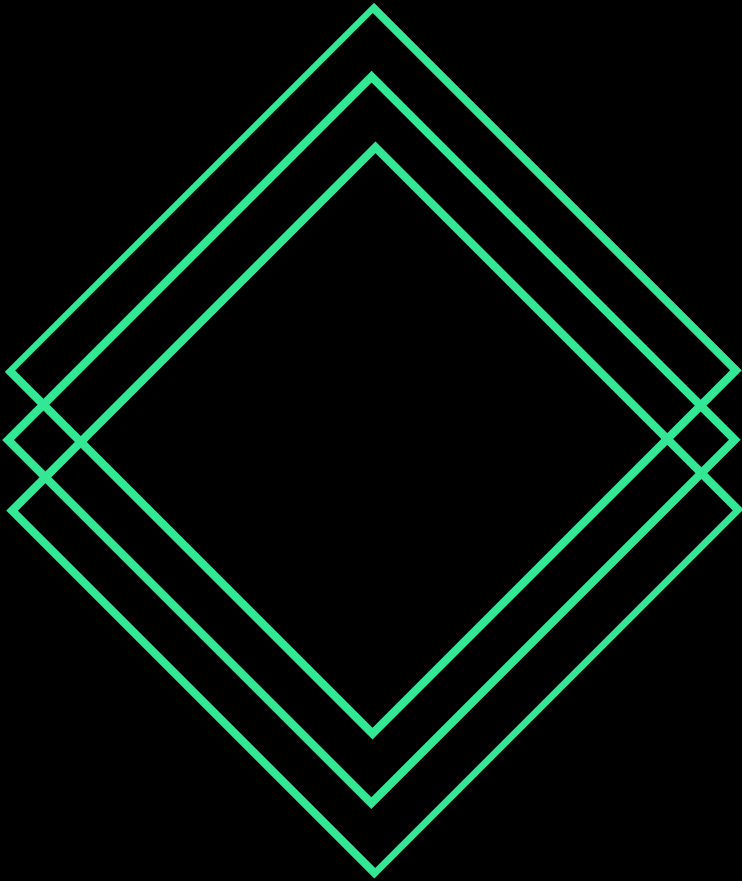


RAPPORT TP4



Manal Haggoud

TP4 FILTRAGE ANALOGIQUE

OBJECTIFS :

APPLIQUER UN FILTRE RÉEL POUR SUPPRIMER LES COMPOSANTES INDÉSIRABLES D'UN SIGNAL.

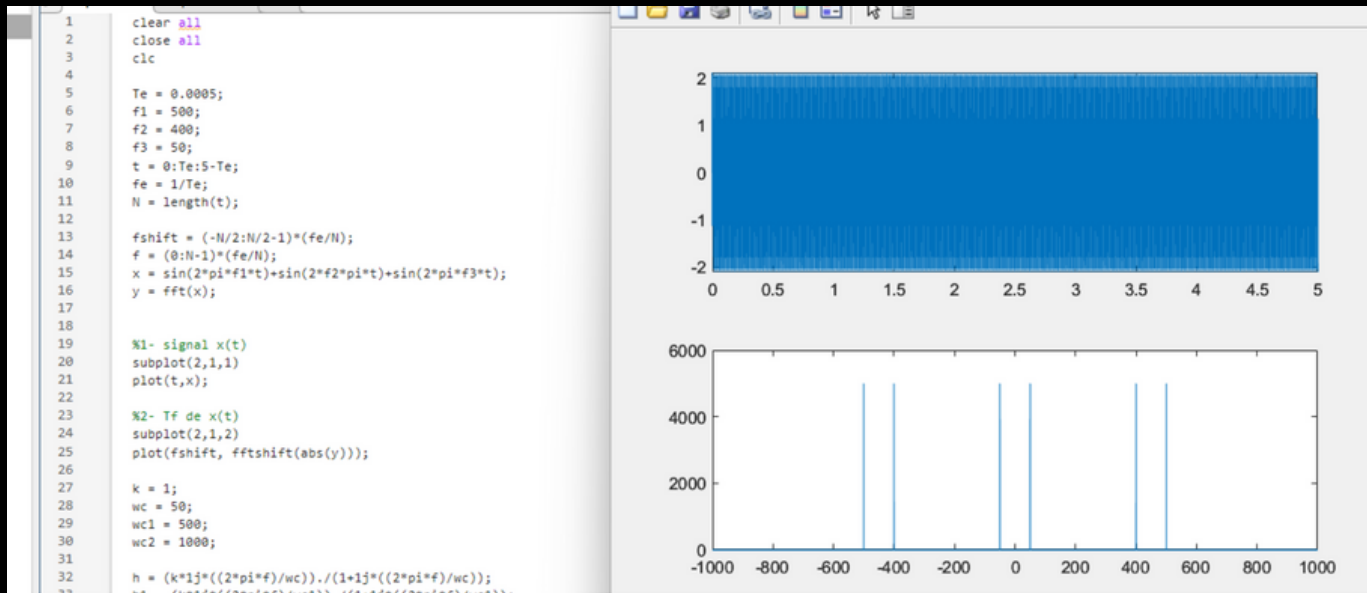
AMÉLIORER LA QUALITÉ DE FILTRAGE EN AUGMENTANT L'ORDRE DU FILTRE.

INTRODUCTION :

LE FILTRAGE ANALOGIQUE EST UNE TECHNIQUE UTILISÉE POUR SÉPARER LES DIFFÉRENTES FRÉQUENCES D'UN SIGNAL ANALOGIQUE EN UTILISANT DES COMPOSANTS ÉLECTRONIQUES TELS QUE DES RÉSISTANCES, DES CONDENSATEURS ET DES INDUCTEURS. IL EXISTE DIFFÉRENTS TYPES DE FILTRES ANALOGIQUES, NOTAMMENT LES FILTRES PASSE-BAS, PASSE-HAUT, PASSE-BANDE ET COUPURE DE BANDE, CHACUN AYANT DES CARACTÉRISTIQUES DE FRÉQUENCE DE COUPURE ET DE PENTE DIFFÉRENTES. LES FILTRES ANALOGIQUES SONT UTILISÉS DANS UNE VARIÉTÉ D'APPLICATIONS, TELLES QUE LA RADIO, LA TÉLÉVISION, LA COMMUNICATION, LES ÉQUIPEMENTS AUDIO ET LES INSTRUMENTS DE MESURE.

1 - FILTRAGE ET DIAGRAMME DE BODE : IMAGE

ON TRACE LE SIGNAL $x(t)$ ET SA TRANSFORMÉ DE FOURRIER



LA TRANSMITTANCE COMPLEXE EST UN CONCEPT UTILISÉ POUR DÉCRIRE LA PERFORMANCE D'UN FILTRE ANALOGIQUE OU D'UN AUTRE SYSTÈME DE TRAITEMENT DE SIGNAL. ELLE EST DÉFINIE COMME LE RAPPORT ENTRE LA TENSION DE SORTIE D'UN SYSTÈME ET LA TENSION D'ENTRÉE, EXPRIMÉ EN TANT QUE NOMBRE COMPLEXE. LA TRANSMITTANCE COMPLEXE EST GÉNÉRALEMENT EXPRIMÉE EN FONCTION DE LA FRÉQUENCE, ET ELLE PEUT ÊTRE REPRÉSENTÉE GRAPHIQUEMENT SUR UN DIAGRAMME DE BODE, QUI MONTRE LA FRÉQUENCE EN ABSCISSE ET LA TRANSMITTANCE COMPLEXE EN ORDONNÉE. LA TRANSMITTANCE COMPLEXE PEUT ÊTRE UTILISÉE POUR DÉTERMINER LES CARACTÉRISTIQUES DE FRÉQUENCE D'UN SYSTÈME, TELLES QUE LES FRÉQUENCES DE COUPURE ET LA PENTE, AINSI QUE POUR CONCEVOIR ET OPTIMISER LES FILTRES ET LES AUTRES SYSTÈMES DE TRAITEMENT DE SIGNAL.

ON TRACE LE MODULE DE H, SON GAIN ET SON DÉPHASAGE AVEC $T=0.0005$

```
subplot(2,1,2)
plot(fshift, fftshift(abs(y)));
```

```
k = 1;
wc = 50;
wc1 = 500;
wc2 = 1000;
```

```
h = (k*1j*((2*pi*f)/wc))./(1+1j*((2*pi*f)/wc));
h1 = (k*1j*((2*pi*f)/wc1))./(1+1j*((2*pi*f)/wc1));
h2 = (k*1j*((2*pi*f)/wc2))./(1+1j*((2*pi*f)/wc2));
```

```
G = 20*log(abs(h));
G1 = 20*log(abs(h1));
G2 = 20*log(abs(h2));
```

```
P = angle(h);
P1 = angle(h1);
P2 = angle(h2);
```

```
figure(2)
```

```
%1- Fonction de Gain
subplot(3,1,1)
semilogx(abs(h));
```

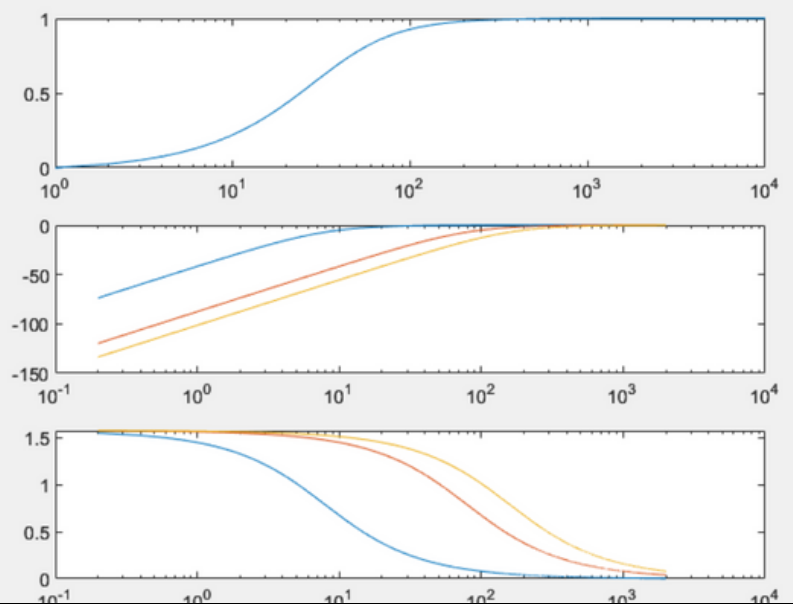
```
%2- 20 log(H) gain / phase
subplot(3,1,2)
semilogx(f,G,f,G1,f,G2);
subplot(3,1,3)
semilogx(f,P,f,P1,f,P2);
```

```
%
```

```
%
```

```
%
```

File Edit View Insert Tools Desktop Window Help



ON APPLIQUE SUR LE SIGNAL LA TRANSMITTANCE COMPLEXE DE FREQUENCE 1000

```
h2 = (k*1j*((2*pi*f)/wc2))./(1+1j*((2*pi*f)/wc2));
```

```
G = 20*log(abs(h));
G1 = 20*log(abs(h1));
G2 = 20*log(abs(h2));
```

```
P = angle(h);
P1 = angle(h1);
P2 = angle(h2);
```

```
figure(2)
```

```
%1- Fonction de Gain
subplot(3,1,1)
semilogx(abs(h));
```

```
%2- 20 log(H) gain / phase
subplot(3,1,2)
semilogx(f,G,f,G1,f,G2);
subplot(3,1,3)
semilogx(f,P,f,P1,f,P2);
```

```
%
```

```
%
```

```
%
```

```
figure(3)
```

```
%
```

```
%N3- filtrage
```

```
subplot(2,1,1)
```

```
filterh2 = [h2(1:floor(N/2)), flip(h2(1:floor(N/2)))];
```

```
s = y .* filterh2;
```

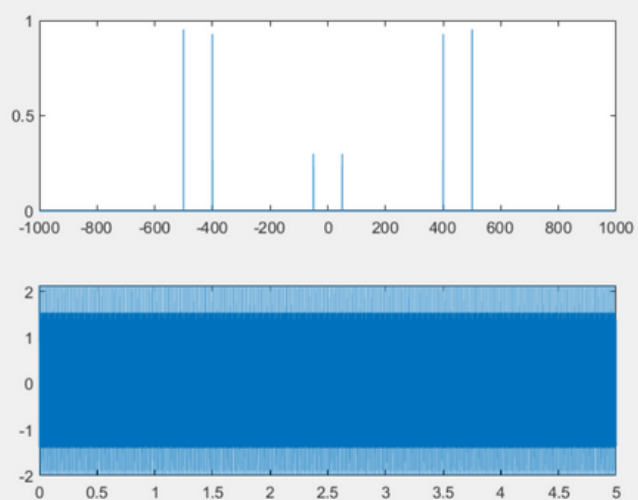
```
S = ifft(s, "symmetric");
```

```
plot(fshift, fftshift(2*abs(fft(S))/N));
```

```
%N4- signal filtré
```

```
subplot(2,1,2)
```

```
plot(t, S);
```



WCZ
x
y

ON REMARQUE QUE LES FILTRES ANALOGIQUES SONT MOINS EFFICACES QUE LES FILTRES RÉELS, ON NE PEUT PAS ELIMINER LES BRUITS DE BASSES FREQUANCES COMPLÈTEMENT, ON POURRA JUSTE LES DIMINUER.

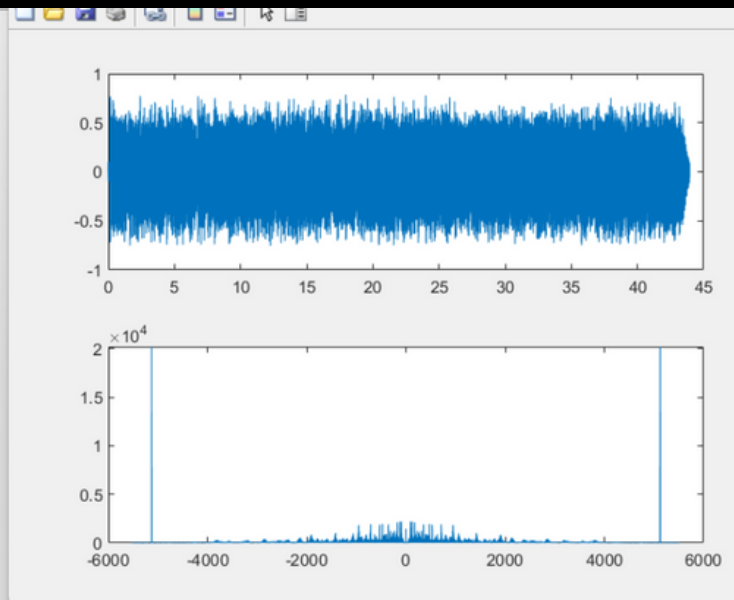
2 - DÉ-BRUITAGE D'UN SIGNAL SONORE

DANS SON PETIT STUDIO DU CROUS, UN MAUVAIS FUTUR INGÉNIEUR A ENREGISTRÉ UNE MUSIQUE EN « .WAV » AVEC UN TRÈS VIEUX MICRO. LE RÉSULTAT EST PEU CONCLUANT, UN BRUIT STRIDENT S'EST AJOUTÉ À SA MUSIQUE. HEUREUSEMENT SON VOISIN, EXPERT EN TRAITEMENT DU SIGNAL EST LÀ POUR LE SECOURIR :

« C'EST UN BRUIT TRÈS HAUTE FRÉQUENCE, IL SUFFIT DE LE SUPPRIMER. » DIT-IL SÛR DE LUI.

ON TRACE LE SIGNAL ET SA TRANSFORMÉE DE FOURRIER :

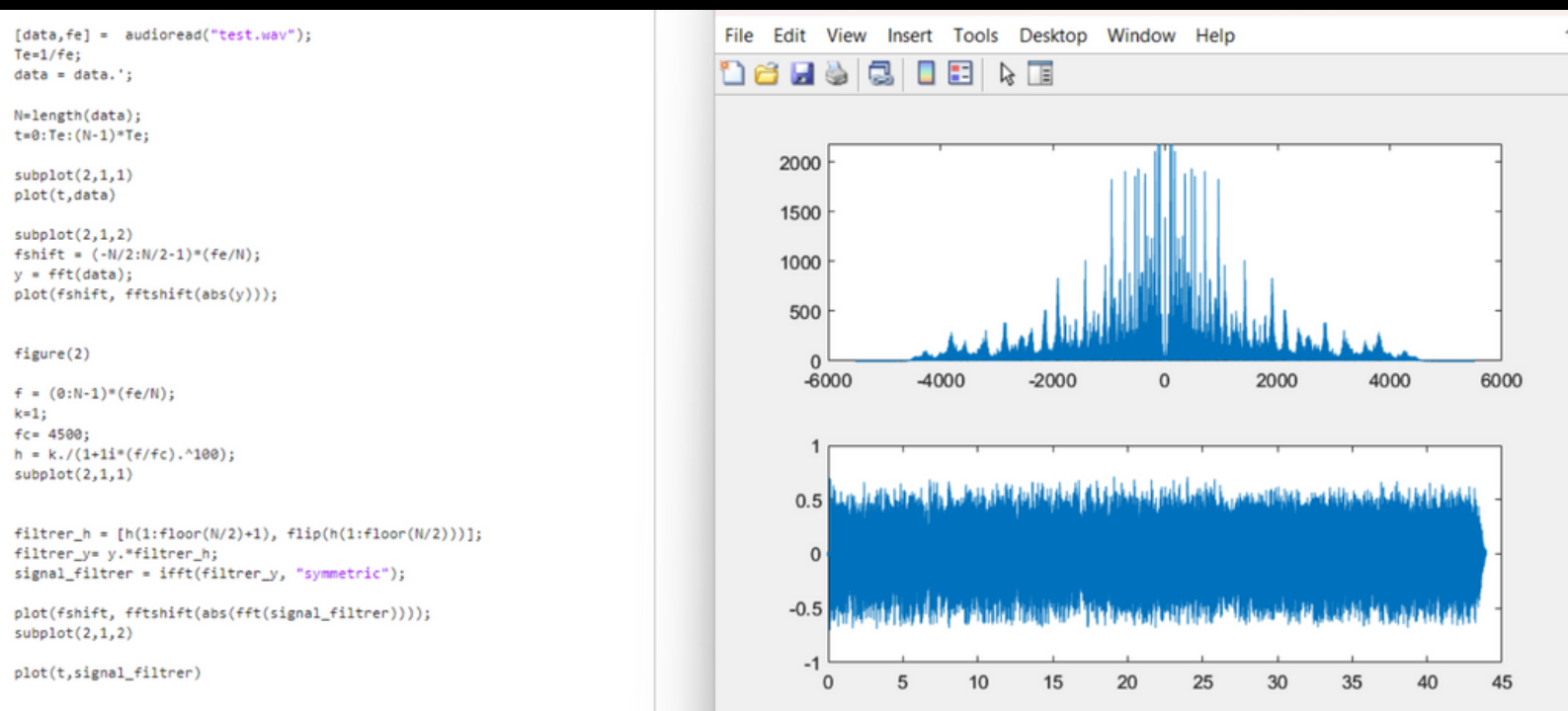
```
1 clear all
2 close all
3 clc
4
5 [data,fe] = audioread("test.wav");
6 Te=1/fe;
7 data = data.';
8
9 N=length(data);
10 t=0:Te:(N-1)*Te;
11
12 subplot(2,1,1)
13 plot(t,data)
14
15 subplot(2,1,2)
16 fshift = (-N/2:N/2-1)*(fe/N);
17 y = fft(data);
18 plot(fshift, fftshift(abs(y)));
19
20
21 figure(2)
22
23 f = (0:N-1)*(fe/N);
24 k=1;
25 fc= 4500;
26 h = k./(1+1i*(f/fc).^100);
27 subplot(2,1,1)
28
29
30 filter_h = [h(1:floor(N/2)+1), flip(h(1:floor(N/2)))];
31 filter_y= y.*filter_h;
32 signal_filter = ifft(filter_y, "symmetric");
33
34 plot(fshift, fftshift(abs(fft(signal_filter))));
35 subplot(2,1,2)
```



ON REMARQUE QUE DANS LE SPECTRE SUIVANT IL Y A UNE FRÉQUENCE ENTRE 4000 ET 6000 D'AMPLITUDE 2 QUI CONSTITUE LE BRUIT, POUR L'ÉIMINER ON DOIT APPLIQUER UN FILTRE PASSE BAS QUI NOUS PERMETTRA DE DIMINUER L'INTENSITÉ DU BRUIT.

ON UTILISE UN FILTRE BUTTERWORTH D'ORDRE 100 :

SPECTRE DU SIGNAL APRES L'APPLICATION DU FILTRE :



ON REMARQUE QU'ON A PU ÉLIMINER LE BRUIT.