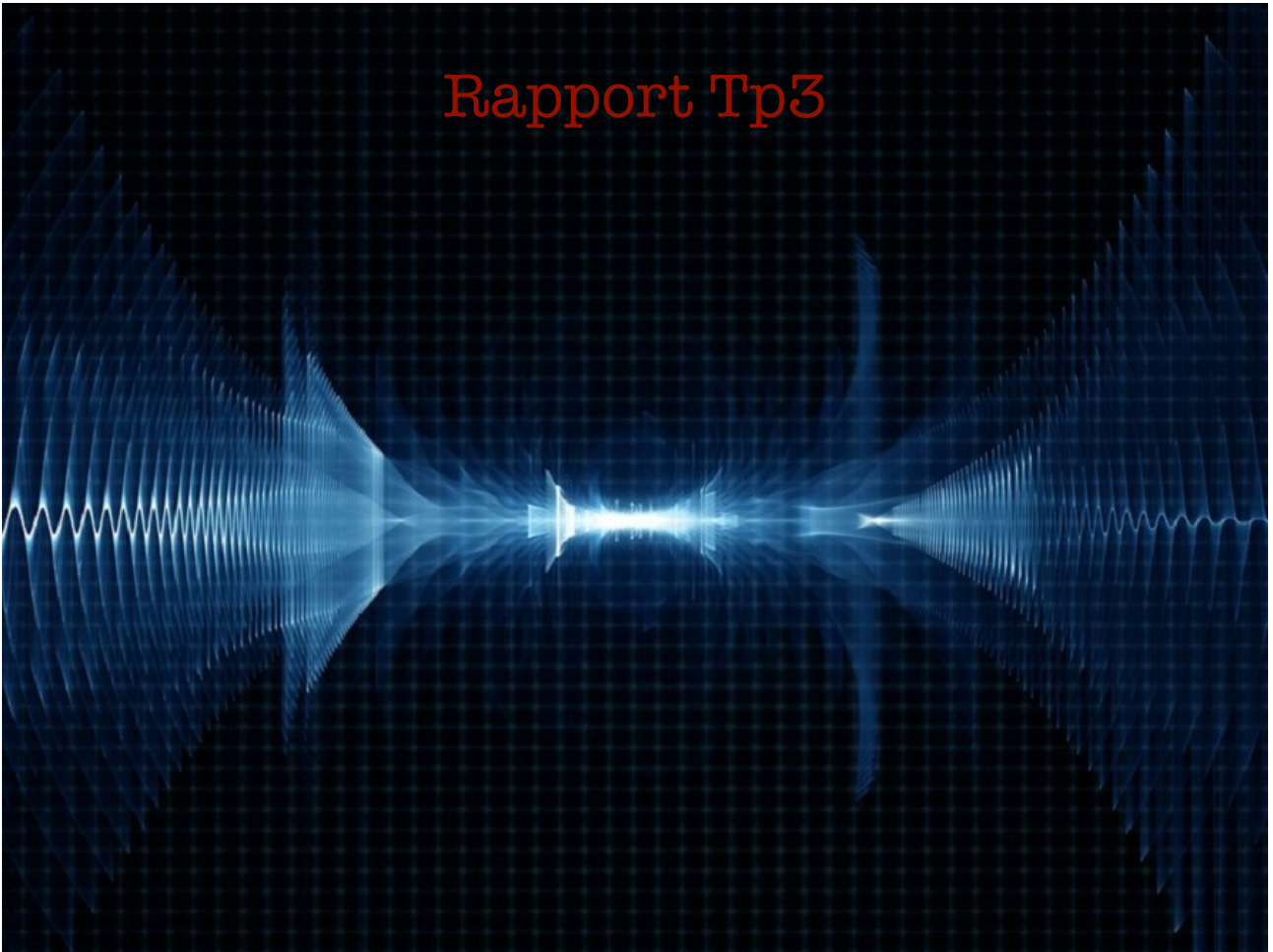


Rapport Tp3



Suppression du bruit provoqué par les mouvements du corps

Realiser par : Taha Najah
Encadré par : M Alae Ammour

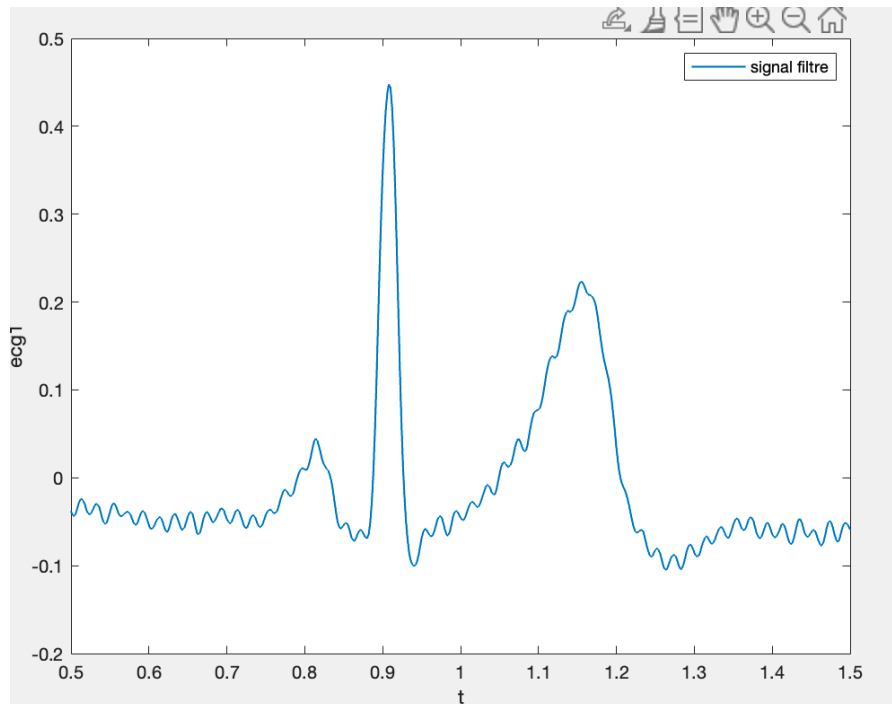
Objectifs

Suppression du bruit autour du signal produit par un électrocardiographe.
Recherche de la fréquence cardiaque.

1) Sauvegarder le signal ECG sur votre répertoire de travail, puis charger-le dans Matlab à l'aide la commande load.

```
load('ecg.mat');
```

2) Ce signal a été échantillonné avec une fréquence de 500Hz. Tracer-le en fonction du temps, puis faire un zoom sur une période du signal.



```
fe = 500;  
N = length(ecg);  
t = (0:N-1)*1/fe;  
f=(0:N-1)*(fe/N);  
fc = 0.5;  
fc0 = 50;  
fc1=40;  
plot(t,ecg,'linewidth',1)  
xlim([0.5 1.5])  
legend(" signal d'origine")  
xlabel("t");  
ylabel(« ecg »);
```

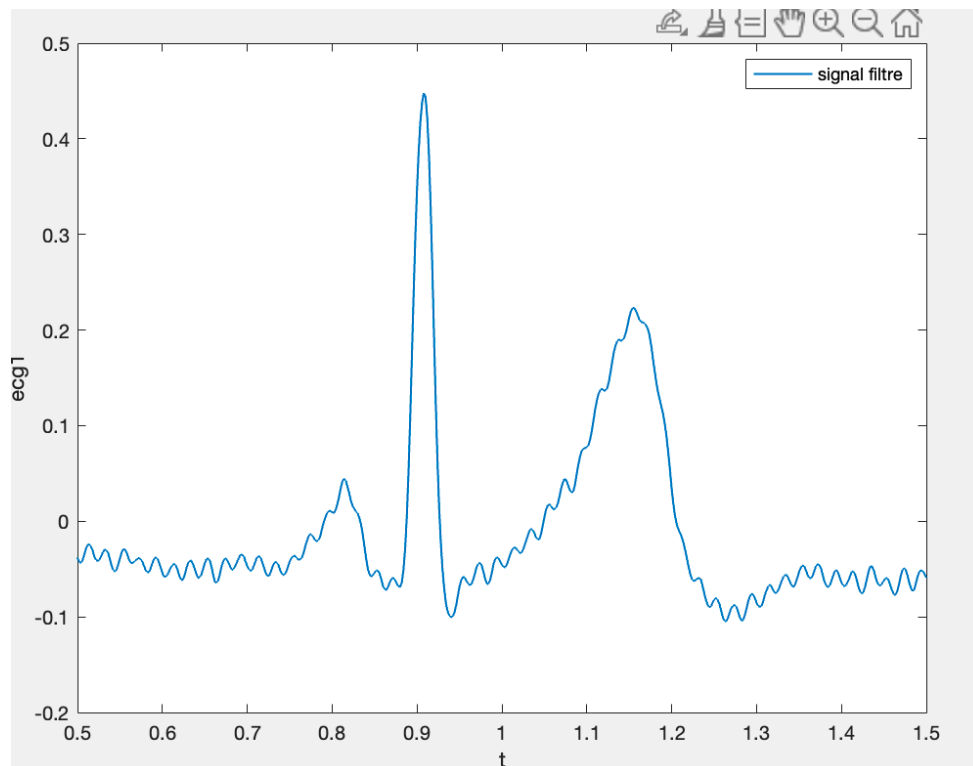
3) Pour supprimer les bruits à très basse fréquence dues aux mouvements du corps, on utilisera un filtre idéal passe-haut.

```

%transformer de fourier rapide
y = fft(ecg);
fshift = (-N/2:N/2-1)*(fe/N);
%filtrage
%creation du filtre pass haut
filtre_pass_Haut = ones(size(ecg));
index_fc = ceil((fc*N)/fe);
filtre_pass_Haut(1:index_fc) = 0;
filtre_pass_Haut(N-index_fc+1:N) = 0;
%filtrage
ecg_filtre_freq = filtre_pass_Haut .* y;
%restitution du signal filtrer
ecg_filtre_temp = ifft(ecg_filtre_freq,"symmetric");
%le bruit de bass frequence
bruit= ecg-ecg_filtre_temp;

```

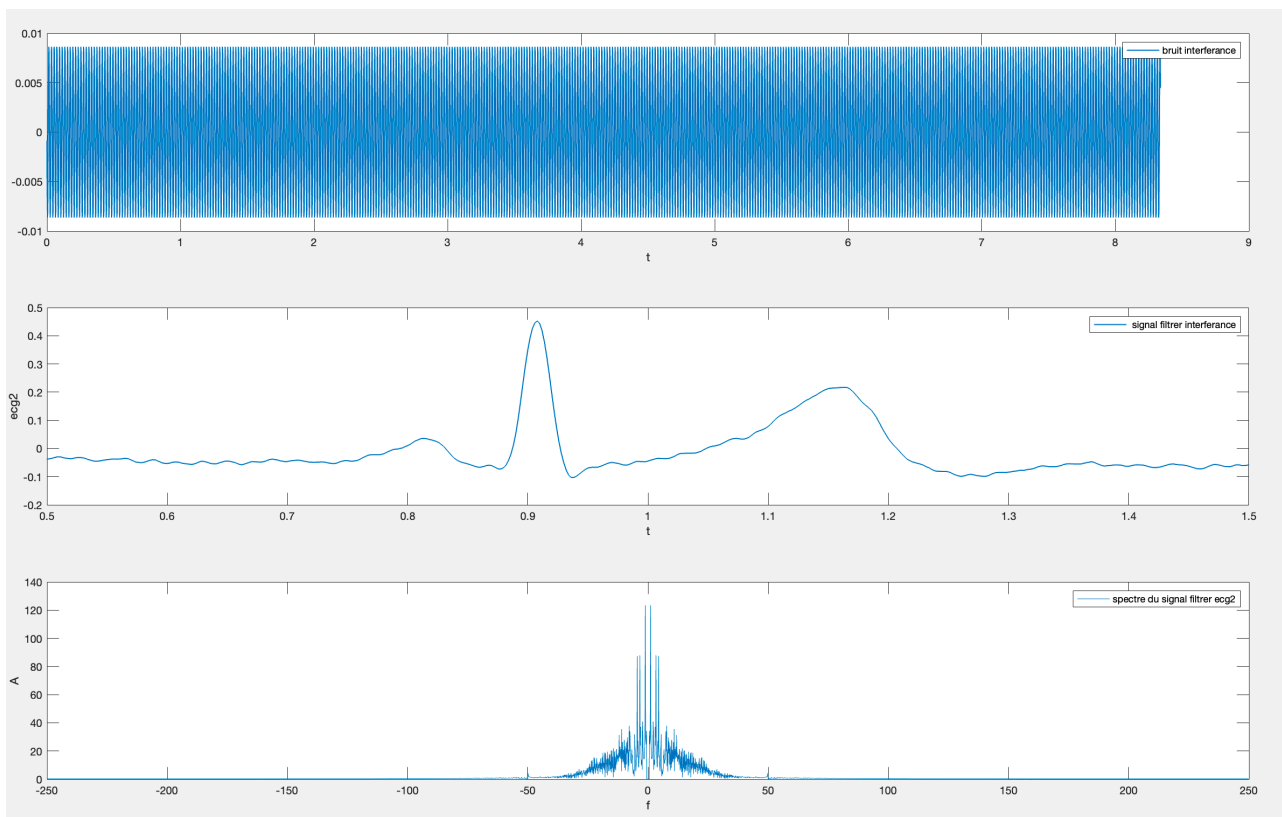
4) Tracer le nouveau signal ecg1:



5) Appliquer un filtre Notch idéal pour supprimer cette composante.

```
%filtrage du bruit d'interference
%creation du filtre
filtre_interference = ones(size(ecg));
index_fc0 = ceil((fc0*N)/fe)+1;
filtre_interference(index_fc0)=0;
filtre_interference(N-index_fc0+1)=0;
%application du filtre
ecg_filtre_int_freq = filtre_interference .*fft(ecg_filtre_temp);
%restitution du signal filtrer
ecg_filtre_int_temp = ifft(ecg_filtre_int_freq,"symmetric");
bruit_inter = ecg_filtre_temp-ecg_filtre_int_temp;
```

6) Visualiser le signal ecg2 après filtrage.



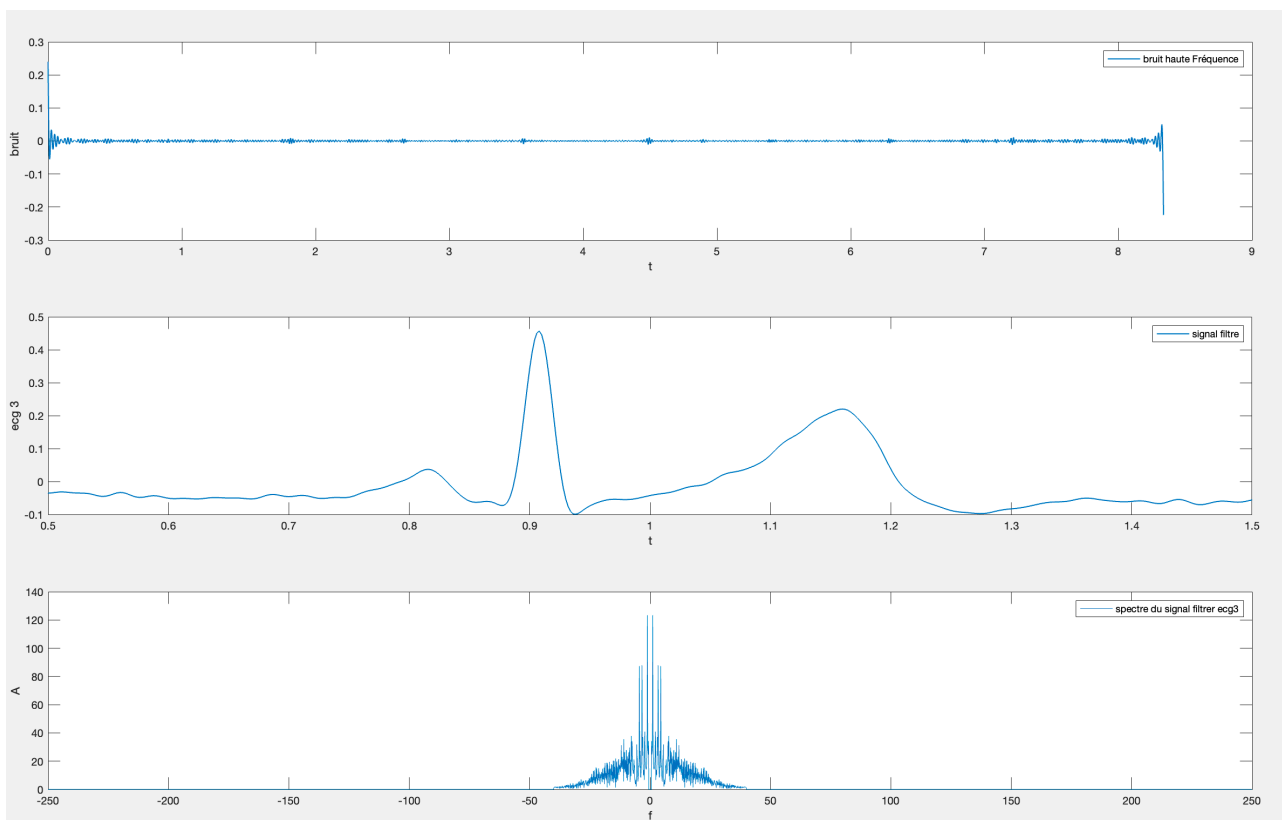
Amélioration du rapport signal sur bruit

7) Chercher un compromis sur la fréquence de coupure :

```
%creation du filtre pass bas
filtre_pass_bas = zeros(size(ecg));
index_fc1 = ceil((fc1*N)/fe);
filtre_pass_bas(1:index_fc1)=1;
filtre_pass_bas(N-index_fc1+1:N)=1;

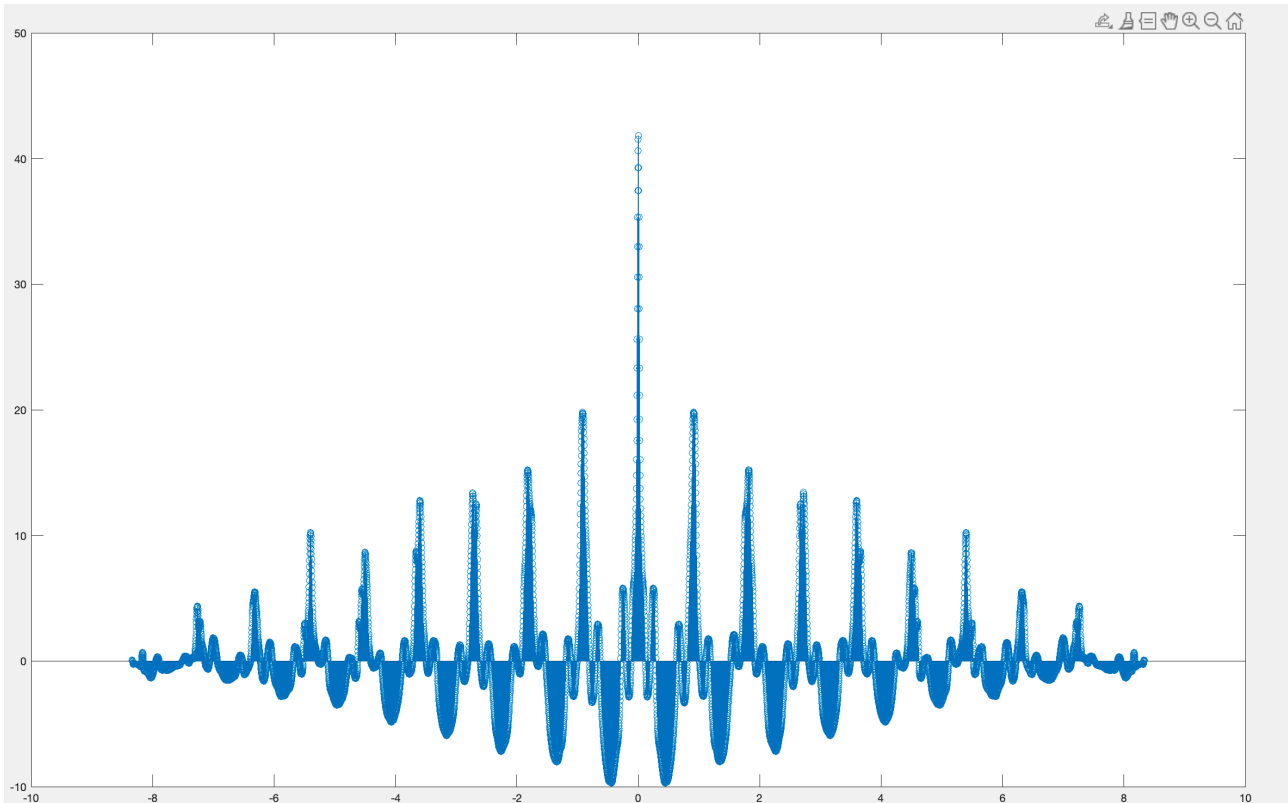
%application du filtre
ecg_filtre_bas_freq = filtre_pass_bas .*fft(ecg_filtre_temp);
%restitution du signal filtrer
ecg_filtre_bas_temp = ifft(ecg_filtre_bas_freq,"symmetric");
%bruit haute frequence
bruit_haut = ecg_filtre_int_temp-ecg_filtre_bas_temp;
```

8) Visualiser une période du nouveau signal filtré ecg3 :



Identification de la fréquence cardiaque avec la fonction d'autocorrélation

9)Ecrire un programme permettant de calculer l'autocorrélation du signal ECG, puis de chercher cette fréquence cardiaque de façon automatique:



% Identification de la fréquence cardiaque avec la fonction d'autocorrélation

```
subplot(4,3,12)
```

```
[c,lags] = xcorr(ecg_filtre_bas_temp,ecg_filtre_bas_temp);
```

```
stem(lags/fe,c)
```

10)Votre programme trouve-t-il le bon pouls ?

Grâce au resultat de la commande xcorr on a pu détecter une corrélation dans les maximum a taux = 2

