

ISBS	Unité : Traitement du signal TP n°1 - Filtrage élémentaire	ISBS1
------	---	-------

Sujet de J.-F. BERCHER – Professeur ESIEE

## ÉNONCÉ

### TP SOUS PYTHON

Le TP sera réalisé sous Python. Sous Python, on dispose d'un certain nombre de bibliothèques scientifiques qui nous seront utiles, notamment *numpy*, *scipy* et *matplotlib*. Le module *pylab* permet de charger des commandes simplifiées analogues à celles que l'on trouve dans Matlab.

#### *Spyder*

Pour ces TP, vous utiliserez l'éditeur Spyder, qui précharge les modules scientifiques et le mode Pylab. Dans *Outils>Préférences>Console IPython>Graphiques*, vous vérifierez que le mode graphique (Matplotlib) est activé, que le chargement automatique de Numpy et Pylab est coché, et enfin que la sortie graphique est positionnée sur « automatique ». Dans *Démarrage*, vous cocherez « ouvrir une console IPython au démarrage ».



Le but de ce TP est de concevoir et appliquer différents filtres numériques sur un signal périodique de fréquence fondamentale  $f_0=200$  Hz, échantillonné à la fréquence  $F_e=8000$  Hz.

Ce signal est stocké dans le vecteur  $x$  sauvegardé dans le fichier `sig1.npz`. On peut le récupérer sous Python par l'instruction : `f=numpy.load('sig1.npz')`

### I – ANALYSE DES DES DONNÉES :

On peut afficher le signal, en temps et en fréquence, par les instructions suivantes :

```
# Pour charger le truc :
f=numpy.load('sig1.npz')
m=f['m']
x=f['x']

Fe=8000
Te=1/Fe

# Affichage temps
figure(1)
t=arange(len(x))*Te
plot(t,x,t,m)
title('Signal avec dérive lente')
show()

et

# Affichage fréquence
figure(2)
N=len(x)
f=freq(N)
plot(f,abs(fftshift(fft(x))))
xlim([-0.5, 0.5])
```

```
title('TF du signal (module)')
```

Commentez et interprétez.

On souhaite à présent modifier le contenu spectral du signal  $x$  par différents filtres numériques de fonction de transfert  $H(z) = B(z)/A(z)$ . Une fonction standard Python vous sera très utile :

- `lfilter` qui implémente l'équation aux différences~ : cette fonction calcule le vecteur  $y$  des sorties d'un filtre numérique spécifié par le vecteur  $B$  des coefficients du numérateur  $B(z)$ , le vecteur  $A$  des coefficients du dénominateur  $A(z)$ , pour un vecteur d'entrées  $x$ , suivant l'instruction~ :  
`y=filter(B,A,x)`

## II – CALCUL DES FILTRES MOYENNEUR ET SOUSTRACTEUR

Le signal est affecté par une dérive lente de son niveau moyen. On cherche alors à extraire cette valeur moyenne à variations lentes, d'une part, et à calculer le signal débarrassé de cette dérive d'autre part. On notera  $M(n)$  la dérive, et  $x_c(n)$  le signal centré.

### II.1 Partie théorique :

- 1) Quelle expression permet de calculer la moyenne du signal  $x$  sur une période ?
- 2) En déduire un filtre, de réponse impulsionnelle  $g(n)$ , qui calcule cette moyenne  $M(n)$ .
- 3) En déduire un autre filtre, de réponse impulsionnelle  $h(n)$ , qui élimine cette moyenne :  $x_c(n) = x(n) - M(n) = x(n) * h(n)$ . Donner l'expression de  $h(n)$ .

Donner les expressions de  $G(z)$  et de  $H(z)$ .

### II.2 Partie pratique

Pour le filtre moyennneur, puis pour le filtre soustracteur :

- Créer et tracer les deux réponses impulsionnelles (on pourra se servir de l'instruction `ones(L)` qui crée un vecteur ligne de  $L$  uns.
- Tracer les réponses en fréquence de ces filtres. Vous pourrez utiliser la fonction `fft` qui calcule la TF, et tracer le module (`abs`) du résultat.
- Filtrer le signal  $x$  par ces filtres. Tracer les signaux de sortie ainsi que leurs spectres en fréquence. Conclure.

## III – ACCENTUATION D'UNE BANDE DE FRÉQUENCE

On souhaite maintenant accentuer (très nettement) la zone de fréquence autour de 1000 Hz sur le signal initial.

### III.1 Partie théorique

Après un rappel éventuel de l'enseignant sur les filtres rationnels, déterminez les pôles  $p_1$  et  $p_2$  d'un filtre permettant de réaliser cette accentuation. Calculer la fonction de transfert,  $H(z)$ , correspondante ainsi que la réponse impulsionnelle  $h(n)$ .

### III.2 Partie pratique

- Le vecteur de coefficients du dénominateur  $A(z)$  sera calculé par : `A=poly([p1,p2])` et vous vérifierez que vous obtenez les coefficients déterminés "à la main".
- Tracer la réponse fréquentielle du filtre
- Calculer sa réponse impulsionnelle par :  

```
# calcul de la RI
d=zeros(300)
d[1]=1
h_accentue=lfilter([1],a,d)
```

(réponse à une impulsion de dirac calculée sur 300 points). La tracer.
- Calculer et tracer la réponse impulsionnelle obtenue avec la formule théorique. La comparer à la précédente.
- Calculer et tracer la sortie du filtre soumis à l'entrée  $x_c$  ainsi que son spectre. Conclure.
- Comment peut-on simultanément amplifier autour de 1000 Hz et supprimer la dérive ? Proposer un filtre qui réalise les deux opérations.

#### IV – FILTRAGE PASSE-BAS [0- 250 Hz] PAR LA MÉTHODE DE LA FENÊTRE

On cherche maintenant à ne conserver que les basses fréquences (0 à 250 Hz) du signal  $x_c$  en filtrant celui-ci par un filtre passe-bas FIR à  $N=101$  coefficients.

##### IV.1 Partie théorique

On considère un filtre passe-bas idéal dont le module de la fonction de transfert,  $H(f)$ , est une fonction rectangulaire. Calculer la réponse impulsionnelle (infinie) du filtre numérique qui réaliserait ce passe-bas de façon idéale.

##### IV.2 Partie pratique

- a. On veut limiter le nombre de coefficients à  $L$  (RIF). Calculer le vecteur  $h$  à  $L$  coefficients représentant cette réponse pondérée (tronquée) par une fenêtre rectangulaire  $\text{rect}_T(t)$  où  $T = L * T_e$ .
- b. Tracer la réponse fréquentielle de ce filtre.
- c. Calculer et tracer la sortie de ce filtre soumis à l'entrée  $x_c$  ainsi que son spectre.
- d. Observer le temps de propagation de groupe de la réponse fréquentielle : `plot(f, grpdelay(B,A,N))`. Commenter.

