

TP N 02 : Synthèse et application d'un filtre RII pass-bas par transformation bilinéaire.

But de TP: Dans ce TP, on synthétisera un filtre RII par la méthode des pôles et zéros puis la méthode de l'invariance impulsionnelle et enfin par transformation bilinéaire en utilisant des filtres analogiques (chebychev et Butterworth).

Rappels sur les filtres RII :

Les filtres RII n'auront pas une phase linéaire (phase linéaire : temps de propagation constant pour tout fréquence). L'intérêt des filtres récurrents (RII) est leur faible coût en calcul. Les inconvénients des filtres récurrents sont : leur non-linéarité en phase ; et leur instabilité numérique. Avec très peu de pôles et zéros on peut assurer la plupart des réponses fréquentielles, dont on peut avoir besoin dans les applications audio. Cependant, le filtre étant rétroactif, les erreurs de précision numériques deviennent une question d'importance, car ils peuvent s'amplifier et devenir hors contrôle, d'abord dans la forme de bruit, mais éventuellement dans la forme d'instabilité. Mais, les filtres RII peuvent être conçus par des méthodes semblables à ceux utilisés pour les filtres analogiques.

• Analyse d'un filtre RII :

Soit le filtre $h(n)$ décrit par l'équation aux différences suivantes :

$$y(n) = 1.2y(n-1) - 0.516y(n-2) + 0.079x(n) + 2 * 0.079x(n-1) + 0.079x(n-2)$$

- La première étape consiste à déterminer les vecteurs b et a . On calcule $H(z)$ (coefficient en z^{-1}) et on trouve : Numérateur : $[0.079 \ 2*0.079 \ 0.079]$ et Dénominateur : $[1 \ -1.2 \ 0.516]$.
- Puis, par programme on peut : déterminer et tracer la réponse impulsionnelle, la réponse fréquentielle (module et phase), le retard de groupe, les pôles et les zéros étudier la stabilité, la nature du filtre, etc.

```
clear all; close all; clc ;% Initialisation des vecteurs de coefficient de N et D
b = [0.079 2*0.079 0.079] ; % Numérateur
a = [1 -1.2 0.516] ; % Dénominateur
figure(1) ; zplane(b,a) % Les pôles et les zéros
N = 32 ; n=0:N-1 ;
delta = [1;zeros(N-1,1)] ; % Impulsion de Dirac
h = filter(b,a,delta) ; % En filtrant une impulsion par la commande « filter ». La réponse impulsionnelle
figure(2) ;
stem(n,h) ;
echelon = ones(1,N) ; % Echelon unitaire
h_ind = filter(b,a,echelon) ; % La réponse indicielle
```

figure(3) ; stem(n,h_ind) ;

```
L = 256 ; fe = 1 ;
[H,f]=freqz(b,a,L,fe) ; % Calcule de H(f) pour Npts valeur de f entre 0 et fe/2 en normalisant fe à 1.
module = abs(H) ;
figure(4) ; plot(f,module) ; % Tracé de module de H en linéaire
phase = angle(H) ;
figure(5) ; plot(f,phase) ; % Tracé de module de la phase
figure(6) ; plot(f,20*log10(H)) ; % Tracé de module de H en dB
[tau,f]=grpdelay(b,a,L,fe) % Calcule du temps de propagation de groupe
figure(7) ; plot(f,tau) ; % Tracé du temps de propagation de groupe
[num,den]=freqz(H,h,2,2)
```

- 1- Calculer les pôles de ce filtre (à préparer), correspondent-ils à ceux de la figure1 ?
- 2- A partir du tracé des pôles et des zéros, esquisser l'allure de $h(n)$ et $H(f)$ en justifiant vos réponses (à préparer). Confirmer avec les figures (2) et (4).
- 3- Etudier la stabilité du filtre (à partir du tracé des pôles et de $h(n)$). Quel est le rôle de ce filtre ?
- 4- Quelle valeur de b faut-il changer pour faire de ce filtre un passe-haut ?
- 5- Modifier les valeurs de a pour avoir une réponse impulsionnelle divergente. Le filtre obtenu est-il stable ?
- 6- Enlever les commentaires et comparer les figures 5 et 6. Quel lien les relie ?
- 7- Quel retard de groupe souhaite-t-on avoir dans la bande passante du filtre ?

I. Synthèse d'un filtre numérique par placement des pôles et zéros :

Il s'agit de déterminer, grâce à la position des pôles et des zéros, la fonction de transfert, les coefficients d'un filtre et l'équation de récurrence d'un filtre numérique coupe-bande du second ordre qui a les caractéristiques suivantes :

- Fréquence à rejeter: 125 Hz
- Largeur de bande à 3 dB: ± 10 Hz
- Fréquence d'échantillonnage: 500 Hz

```
clear all; close all; clc;
fe = 500 ; df = 10 ; fc = 125 ;
teta = 360*fc/fe ; tet = 2*pi*fc/fe ;
R = 1-df*pi/fe;
K = (exp(2*j*teta)-2*R*cos(teta)*exp(j*teta)+R*R)/(exp(2*j*teta)-1);
K =abs(K) ;
a = [1 -2*R*cos(tet)+ R*R] ;
b = K*[1 0 -1] ;
dirac = [1;zeros(99,1)] ;
h = filter(b,a,dirac) ;
```

```
[H F] = freqz(b,a,512,fe) ;
[tau, f] = grpdelay(b,a,512,fe) ;
figure;
subplot(2,2,1) ; hold on ; plot(h,'r') ;
subplot(2,2,2) ; hold on ; plot(F,abs(H),'r') ;
subplot(2,2,3) ; hold on ; zplane(b,a) ;
subplot(2,2,4) ; hold on ; plot(f,tau,'r') ;
```

- 1- Quelle est la nature de la bande passante créée.
- 2- Modifier ce programme pour en faire un coup-bande et commenter les graphes obtenus.
- 3- Reprendre ce programme pour filtrer le signal ecg (fe=500Hz, et bruit à 50Hz).
 - Visualiser les TFD du signal avant et après filtrage.
 - Comparer avec la méthode des fenêtres, laquelle vous semble préférable (justifier).

II. Synthèse d'un filtre numérique par transformation d'un filtre analogique :

Pour déterminer les coefficients du filtre RII, il suffit de synthétiser chaque filtre en continu $H(p)$ puis de passer à $H(z)$, soit par transformation bilinéaire, ou par invariance impulsionnelle.

On veut, par exemple, synthétiser un filtre pass-bas avec les spécifications suivantes :

$F_e = 3000\text{Hz}$; $f_p = 500\text{Hz}$, et une oscillation en bande passante $\delta_1 < 3\text{ db}$ et une atténuation en bande atténuée $\delta_2 < 40\text{db}$.

On commence par choisir un filtre analogique normalisé $H_n(p)$ (en continu) d'ordre N, puis on le dénormalise pour créer $H(p)$ continu. Ensuite une méthode de transformation permettant trouver les paramètres du filtre numérique.

```
Fe = 3000; fp = 500; att_p=3 ; att_a = 40 ; N = 10 ;
wp = fp*2*pi ;
[z,p,k] = cheb1ap(N,att_p) ;
[Bpn,Apn] = zp2tf(z,p,k) ;
[Bp, Ap] = lp2lp(Bpn,Apn,wp) ;
[Bn, An] =impinvar(Bp,Ap,Fe) ;
figure ;
subplot(1,2,1) ; zplane(1,Ap) ;
subplot(1,2,2) ; zplane(1,An) ;
[r,p,k] = residue(Bp,Ap) ;
t= 0:1/(5*Fe):0.02 ;
ha = exp(t*(p.'))*r ;
hn = filter(Bn,An,[1;zeros(49,1)]) ;
figure ;
subplot(1,2,1); plot(t,ha) ; hold on ;
stem(0:1/Fe:49/Fe*hn,'r')
[Ha,w]=freqz(Bp,Ap,2*pi*(1:20:Fe/2)) ;
subplot(1,2,2); plot(w/(2*pi),abs(Ha)) ;
```

hold on ; stem (f,abs(H),'r.')

- 1) Que contiennent z, p et $k, B_{pn}, A_{pn}, B_p, A_p$?
- 2) A partir du tracé des pôles, étudier la stabilité des 2 filtres.
- 3) Aidez vous du help pour expliquer les instructions :
`zp2tf (z,p,k); lp2lp(Bpn,Apn,wp) ; [r,p,k] = residue(Bp,Ap); t= 0:1/(5*Fe):0.02 ; ha = exp(t*(p.))*r ;
hn = filter(Bn,An,[1;zeros(49,1)]) ; freqz(N,D,2*pi*(1:20:Fe/2)) ; plot(w/(2*pi),abs(Ha));`
- 4) Commenter les réponses impulsionnelle et fréquentielles.
- 5) Quelles sont les caractéristiques du filtre de chebychev (avantages et inconvénient).
- 6) Que se passe-t-il si on augmente N ?
- 7) Testez les autres filtres analogiques et commenter les différences (oscillations, Δf , etc).
- 8) Que faut-il modifier pour en faire un pass-haut ?
- 9) Quel est l'inconvénient principal de l'approche par invariance impulsionnelle ?

III. On veut refaire le même travail en employant la transformation bilinéaire
`[Bn, An] = bilinear(Bp,Ap,Fe,fp) ;`

- Comparer les 2 méthodes lorsque f_p est proche de la demi-fréquence d'échantillonnage $f_e/2$.
- Tester le fonction **cheb1ord**.