

TP 7 MASTER 1 Filtres Numériques en précision finie

1. INTRODUCTION

Dans ce TP nous allons nous intéresser à l'effet de la précision finie des coefficients d'un filtre numérique lorsqu'on l'implémente dans un DSP (Digital Signal Processor). En effet, lors de l'analyse des filtres numériques RIF ou RII (Etude spectrale : fonction de transfert en Z, Réponse fréquentielle et position des zéros et pôles dans le plan z) on suppose en fait que les coefficients du filtre et les échantillons du signal ont une précision infinie (nombre de bits illimités). Alors que dans le cas pratique réel, le nombre de bits utilisé pour chaque coefficient chaque échantillon est fini (on parle de précision finie). Ceci va donc engendrer des erreurs dites de quantification ou d'arrondi. Quelle est donc l'effet de ces erreurs sur les caractéristiques réelles du filtre implémenté ?

2. FILTRE RII PASSE-BAS DU DEUXIEME ORDRE EN PRECISION INFINIE

Le filtre numérique est spécifié par le gabarit en dB de la figure 1 avec comme paramètres :
 $F_p = 1\text{kHz}$; $F_s = 3\text{kHz}$; $\Delta 1 = 1\text{dB}$; $\Delta 2 = -20\text{dB}$; $F_e = 10\text{kHz}$

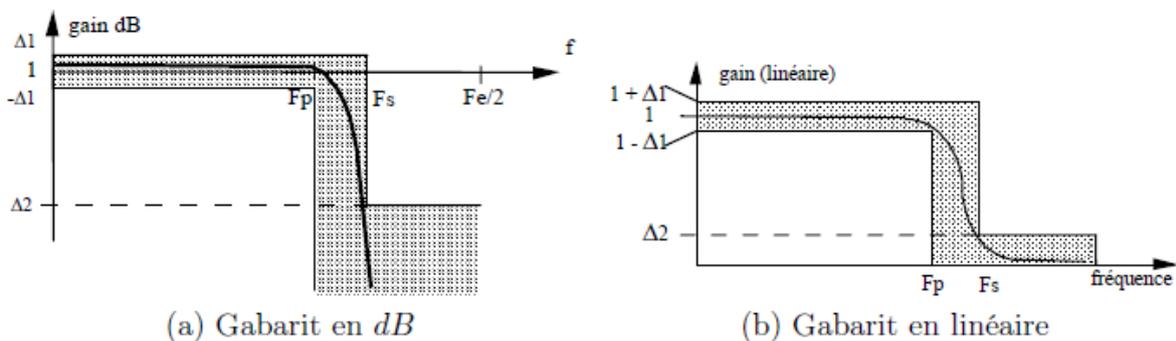


FIG. 1 – Gabarit des filtres passe-bas à étudier

La synthèse par la méthode bilinéaire du filtre de Chebyshev $H(z)$ aboutit à la fonction :

$$H(z) = \frac{0.079(z + 1)^2}{z^2 - 1.2z + 0.516}$$

Questions :

- Écrire le programme Matlab permettant de visualiser la réponse fréquentielle en module et en phase du filtre numérique (voir l'annexe A décrivant ce programme Matlab). Faites la relation entre l'échelle des abscisses de la commande plot de Matlab et les fréquences. Quelles sont donc les relations entre fréquence d'échantillonnage et fréquence de coupure ? Le filtre respecte-t-il le gabarit spécifié ?
- Visualisez la réponse impulsionnelle.
- Tracez dans le plan complexe les pôles et les zéros du filtre. Le filtre est-il stable ?

On désire réaliser un filtre numérique $H(z)$ équivalent à un filtre analogique de Butterworth passe-bas $H(j\omega)$ du deuxième ordre qui présente une fréquence de coupure F_c de 1KHz. La fréquence d'échantillonnage F_e sera de 8kHz. Son filtre de Butterworth normalisé est le suivant :

$$H_{Normalise}(j\omega) = \frac{1}{1 + \sqrt{2}j\omega - \omega^2}$$

$$H_{Normalise}(p) = \frac{1}{1 + \sqrt{2}p + p^2}$$

- Le gabarit d'un filtre passe-bas est défini comme indiqué sur la figure 1.(b). Calculer les paramètres $\Delta 1$ et $\Delta 2$ pour que le filtre passe bas de fréquence de coupure $F_c = 1KHz$ entre dans le gabarit, lorsque $F_p = 0.7F_c$ et $F_s = 3F_c$.
- Faire la synthèse par la méthode bilinéaire du filtre $H(j\omega)$ afin d'obtenir $H_b(z)$. On étudiera l'influence de la distorsion en fréquence impliquée par la méthode.
- Tracez dans le plan complexe les pôles et les zéros des filtres. Les filtres sont-ils stables ?
- Faire la même chose en utilisant la méthode de synthèse basée sur l'invariance impulsionnelle afin d'obtenir $H_i(z)$.
- Écrire le programme Matlab permettant de faire la synthèse des filtres $H_b(z)$ et $H_i(z)$. L'annexe C est à compléter pour répondre à cette question.

3. FILTRE RII PASSE-BAS DU DEUXIEME ORDRE EN PRECISION FINIE

L'étude suivante reprend l'exemple du filtre RII précédent.

- Les coefficients sont arrondis sur 7 puis 3 bits après la virgule. À l'aide du programme Matlab fourni en annexe E, étudier les modifications de la réponse fréquentielle du filtre quantifié, en particulier par rapport au gabarit. Quelles sont les problèmes apportés par la quantification sur la stabilité du filtre ? On placera un nombre suffisant de bits avant la virgule pour éviter les débordements.
- On se basera sur le diagramme des pôles et des zéros du filtre en vérifiant l'influence de la quantification sur les valeurs des coefficients du filtre puis sur les valeurs des pôles. Tracez le diagramme des pôles et des zéros avant après quantifications sur 8 et 4 bits.
- Étudiez les problèmes de débordements du filtre. A quelles conditions le filtre peut-il diverger ?

4. FILTRE RIF PASSE-BAS DU DEUXIEME ORDRE EN PRECISION INFINIE

Le gabarit du filtre passe-bas étant défini comme à la figure 1, la synthèse du filtre donne deux fonctions de transfert $H(z)$ selon le nombre de points considéré ($N = 11$ puis $N = 7$) :

$$H_{11}(z) = -0,0309396.(z^{-1}+z^{-9}) - 0,0390182.(z^{-2}+z^{-8}) + 0,0766059.(z^{-3}+z^{-7}) + 0,288307.(z^{-4}+z^{-6}) + 0,4.z^{-5}$$

$$H_7(z) = -0,0409365.(1 + z^{-6}) + 0,078369.(z^{-1} + z^{-5}) + 0,289996.(z^{-2} + z^{-4}) + 0,4.z^{-3}$$

- Écrire le programme Matlab permettant de visualiser la réponse fréquentielle en module et en phase du filtre numérique (voir l'annexe B décrivant ce programme Matlab). Faites la relation entre l'échelle des abscisses de la commande plot de Matlab et les fréquences. Quelles sont donc les relations entre fréquence d'échantillonnage et fréquence de coupure ? Le filtre respecte-t-il le gabarit spécifié ?
- Visualisez la réponse impulsionnelle.

- Tracez dans le plan complexe les pôles et les zéros du filtre. Le filtre est-il stable ?
- Soit le filtre passe bas idéal de fréquence de coupure Ω_c . Déterminez la réponse impulsionnelle $h(n)$ de ce filtre. On prendra comme application numérique une fréquence de coupure $F_c = 1kHz$ pour une fréquence d'échantillonnage $F_e = 8kHz$ et un Δf de $2kHz$.
- Quel est la longueur théorique du filtre respectant ce gabarit ?

Le gabarit du filtre passe-bas est défini comme indiqué sur la figure 1.

- Synthèse selon la méthode de fenêtrage. Le gabarit idéal $Hi(\Omega)$ est échantillonné, et la réponse impulsionnelle correspondante $hi(n)$ est calculée puis tronquée sur un nombre fini d'échantillons M . Cette réponse est ensuite multipliée par la fenêtre choisie. On obtient $h(n) = hi(n) \cdot w(n)$.
- Écrire le programme Matlab permettant de faire la synthèse du filtre RIF $H(z)$ par la méthode de fenêtrage pour des fenêtres rectangulaire et de Hamming. Le programme de l'annexe D est à compléter.
- Visualisez la réponse fréquentielle en module et en phase, observez la réponse impulsionnelle.
- Étudiez l'influence de la fenêtre sur le filtre numérique. Peut-on mettre en évidence le phénomène de Gibbs.

5. FILTRE RIF PASSE-BAS DU DEUXIEME ORDRE EN PRECISION FINIE

Reprendre les questions précédentes du filtre RIF sur le filtre :

$$H_{11}(z) = -0.0309396(z^{-1}+z^{-9})-0.0390182(z^{-2}+z^{-8})+0.0766059(z^{-3}+z^{-7})+0.288307(z^{-4}+z^{-6})+0.4z^{-5}$$

- On vérifiera particulièrement les points suivants :
 - caractéristique en phase du filtre ;
 - problèmes de débordement de l'accumulateur.
- programme de l'annexe E

Écrire le programme Matlab réalisant la fonction de convolution dans le domaine temporelle à l'aide du programme de l'annexe F. La fonction conv(x, h) prend 2 vecteurs de longueur N et M et retourne un vecteur de longueur $N + M + 1$.

- Écrire le programme Matlab réalisant la fonction de convolution dans le domaine fréquentiel par l'utilisation de la fonction FFT.
- Vérifier que le résultat est identique dans les deux cas.
- Calculer précisément les complexités des deux approches. Comparer les complexités des méthodes fréquentielle et temporelle pour différentes valeurs de N et M . On pourra par exemple regarder les cas $N = 11, M = 7$; $N = M = 1024$; $N = 1024, M = 11$.
- Étudiez les réponses temporelles et fréquentielles des fenêtres Rectangulaire, Bartlett, Hamming, Hanning, Blackman à l'aide du programme de l'annexe H.
- Retrouvez les caractéristiques du cours (largeur du lobe principal, atténuation du lobe secondaire) et étudiez l'influence du nombre de points N de la fenêtre.

Annexe

A Analyse d'un filtre numérique RII passe-bas du second ordre

```
close all;

% Premiere partie
% 1 Analyse de filtres numériques passe-bas du second ordre
% 1.1 Analyse d'un filtre numérique RII passe-bas du second ordre

b = [0.079 2*0.079 0.079];          %Numérateur
a = [1 -1.2 0.516];                %Dénominateur

N = 32; n=0:N-1;
delta = [1; zeros(N-1,1)]; %Impulsion
step = ones(N,1);                  %Echelon unité
h = filter(b, a, delta);           %Réponse du filtre (ou impz(b,a);)
figure(1); stem(n,h);

L = 256;
[h,w] = freqz(b,a,L); %Retourne le resultat de la TF sur L points entre 0 et pi
m = abs(h); p = angle(h);

figure(2); plot(w(1:L-1),20*log10(m(1:L-1))); title('Log Magnitude Filtre RII 2nd ordre');
axis([0 pi -60 2]); grid
figure(5); plot(w,m); title('Magnitude'); figure(3); plot(w,p); title('Phase');

figure(4); zplane(b,a);
zero = roots(b); pole = roots(a);

Delta1 = 1; Delta2 = -20;
Fe = 10000;
Fp = 1000; NFp = round(L*Fp/(Fe/2));
Fs = 3000; NFs = round(L*Fs/(Fe/2));
gabh = [Delta1*ones(NFs,1); Delta2*ones(L-NFs,1)];
gabl = [-Delta1*ones(NFp,1); -5000*ones(L-NFp,1)];
figure(2); hold on; plot(w,gabh,'r'); plot(w,gabl,'r');
```

B Analyse d'un filtre numérique RIF passe-bas du second ordre

% 1.2 Analyse d'un filtre numérique RIF passe-bas du second ordre

```
b11 = [0 -0.0309396 -0.0390182 0.0766059 0.288307 0.4 0.288307 0.0766059 -0.0390182 -0.0309396 0];
b7 = [-0.0409365 0.078369 0.289996 0.4 0.289996 0.078369 -0.0409365];
[h11,w] = freqz(b11,1,L);
[h7,w] = freqz(b7,1,L);
figure; plot(w,20*log10(abs(h11)), 'g'); title('Log Magnitude Filtre RIF 2nd ordre');
axis([0 pi -60 2]); hold on;
plot(w,20*log10(abs(h7)), 'b'); grid
plot(w,gabh, 'r'); plot(w,gabl, 'r');
figure; plot(w,unwrap(angle(h11)), 'g'); title('Phase Filtre RIF 2nd ordre');
grid hold on; plot(w,unwrap(angle(h7)), 'b');

h11 = filter(b11, 1, delta);
h7 = filter(b7, 1, delta);
figure; subplot(2,1,1); stem(n,h11, 'g'); subplot(2,1,2); stem(n,h7, 'b')
figure; zplane(b11,1); figure; zplane(b7,1);
```

C Synthèse d'un filtre RII passe-bas

% 2 Synthèse d'un filtre passe-bas du second ordre

% 2.1 Synthèse d'un filtre RII

Fe = 8000; Fc = 1000; Fp = 700; Fs = 3000;

```
[N, Wn] = buttord(?,?,?,?);
disp(sprintf('Ordre du filtre de butterworth : %d \n',N));
[b a] = butter(?, ??, 's'); % Filtre analogique de Butterworth
[bb ab] = bilinear(b, a, Fe); % Transformation bilinéaire sans prédistorsion
[bbp abp] = bilinear(b, a, Fe, Fc); % Transformation bilinéaire avec prédistorsion en Fc

[bi ai] =impinvar(b, a, Fe); % Invariance impulsionnelle

L = 256; [hb,w] = freqz(bb,ab,L); figure;
plot(w,20*log10(abs(hb)), 'g'); title('Log Magnitude Filtre RII Butterworth');
axis([0 pi -40 2]); grid

[hbp,w] = freqz(bbp,abp,L);
hold on; plot(w,20*log10(abs(hbp)), 'b');

[hi,w] = freqz(bi,ai,L);
plot(w,20*log10(abs(hi)), 'c');
```

D Synthèse d'un filtre RIF passe-bas

% 2.2 Synthèse d'un filtre RIF

% Réponse impulsionnelle théorique : $h(n) = Wc/\pi \text{ sinc}(nWc/\pi)$
Fe = 8000; Fc = 1000; Wc = 2*pi*Fc/Fe;

% Méthode du fenetrage

N = 33; % Longueur du filtre RIF, doit etre impaire pour la suite
alpha = (N-1)/2; n=0:N-1;
b = ?????; %RI du filtre idéal
bh = b.*hamming(N)';

[h,w] = freqz(b,1,L); figure; plot(w,20*log10(abs(h)), 'b');
title('Log Magnitude Filtre RIF fenetrage'); axis([0 pi -60 2]); grid
[hh,w] = freqz(bh,1,L);
hold on; plot(w,20*log10(abs(hh)), 'g');

figure; stem(n,b, 'b'); hold on; stem(n,bh, 'g'); grid

% Methode de l'echantillonnage frequentiel avec une TFD sur N points

N = 33;
Hef = [ones(???,1); zeros(???,1); ones(???,1)] .* expression_de_la_phase;
%Mettre eventuellement un terme de phase

figure; subplot(2,1,1); stem(n,abs(Hef));
subplot(2,1,2); stem(n,angle(Hef));

bef = ifft(Hef,N);
[hef,w] = freqz(bef,1,L);

figure; subplot(2,1,1);
plot(w,20*log10(abs(hef)), 'b'); title('Log Magnitude Filtre RIF echantillonnage frequentiel');
axis([0 pi -40 2]); subplot(2,1,2); plot(w,unwrap(angle(hef)), 'b');
title('Phase Filtre RIF echantillonnage frequentiel');

figure; stem(n,abs(bef), 'b'); grid

E Filtrage numérique en précision finie

```
b = [0.079 2*0.079 0.079];           %Numérateur
a = [1 -1.2 0.516];                 %Dénominateur

L = 256;

[h,w] = freqz(b,a,L); figure; plot(w,20*log10(abs(h))); title('Log Magnitude Filtre RII 2nd ordre')
axis([0 pi -40 6]); grid
gabH = [Delta1*ones(NFs,1); Delta2*ones(L-NFs,1)];
gabL = [-Delta1*ones(NFp,1); -5000*ones(L-NFp,1)]; hold on;

% Arrondi sur b bits en Ca2 (1 bit de signe)
bit=8;                               %Nombre de bits
q=2^(1-bit);
bQ8= ???; %Quantification des coefficients
aQ8= ???; %Quantification des coefficients
[hQ8,w] = freqz(bQ8,aQ8,L); plot(w,20*log10(abs(hQ8)),'g');

bit=4;                               %Nombre de bits
q=2^(1-bit); bQ4=???; aQ4= ???; [hQ4,w] = freqz(bQ4,aQ4,L);
plot(w,20*log10(abs(hQ4)),'c');

plot(w,gabH,'r'); plot(w,gabL,'r'); legend('Précision infinie','8 bits','4 bits');

figure; zplane(b,a); zero = roots(b); pole = roots(a); figure;
zplane(bQ8,aQ8); figure; zplane(bQ4,aQ4);

% 3.2 Étude d'un filtre IIR du quatrième ordre

% Forme directe
% a(1)*y(n) = b(1)*x(n) + b(2)*x(n-1) + ... + b(nb+1)*x(n-nb)
%           - a(2)*y(n-1) - ... - a(na+1)*y(n-na)
b = 0.0599*[1 -2.1832 2.9976 -2.1832 1];
a = [1 -3.1914 4.17 -2.5854 0.6443];
..... A COMPLETER

% Forme cascade
b1 = sqrt(0.0599)*[1 -0.6511 1];      b2 = sqrt(0.0599)*[1 -1.5321 1];
a1 = [1 -1.5684 0.6879];             a2 = [1 -1.623 0.9366 ];
..... A COMPLETER
```

F Convolution Discrete

```
close all; %Ferme les figures en cours

% II.4 Convolution

N=11; M=7;
L=N+M-1; n=0:L-1;

h = [0; 0.1; 0.2; 0.3; 0.2; 0.1; 0; zeros(L-M,1)];
x = [ones(N,1); zeros(L-N,1)];
y1 = conv(x, h);
figure; stem(n,h,'r'); hold on; stem(n,x,'b'); stem(n,y1(1:L),'k');

..... A COMPLETER
```

G Analyse spectrale

```
% II.5 Analyse spectrale

N=64;
hr=boxcar(N);
figure; plot(hr,'y. '); hold on; grid on
ht=bartlett(N); plot(ht,'r:')
hn=hanning(N); plot(hn,'g--')
hm=hamming(N); plot(hm,'m-.')
hb=blackman(N); plot(hb,'b-')

xlabel('n');
ylabel('w(n)'); title('Reponses temporelles de différentes fenetres');
legend('Rectangulaire','Bartlett','Hanning','Hamming','Blackman')
AXIS([0 N 0 1.1])

figure
subplot(2,3,1)
grid on
HR = fft(hr,1024);
plot((0:510)/1024,20*log10(abs(HR(1:511)/HR(1))))
AXIS([0 0.5 -80 0]);
ylabel('W dB')
Title('Fenêtre rectangulaire')

subplot(2,3,2)
grid on
HT = fft(ht,1024);
plot((0:510)/1024,20*log10(abs(HT(1:511)/HT(1))))
AXIS([0 0.5 -80 0]);
ylabel('W dB')
Title('Fenêtre de Bartlett')

subplot(2,3,3)
grid on
HN = fft(hn,1024);
plot((0:510)/1024,20*log10(abs(HN(1:511)/HN(1))))
```

```

AXIS([0 0.5 -80 0]);
ylabel('W dB')
Title('Fenêtre de Hanning')

subplot(2,3,4)
grid on
HM = fft(hm,1024);
plot((0:510)/1024,20*log10(abs(HM(1:511)/HM(1))))
AXIS([0 0.5 -80 0]);
ylabel('W dB')
Title('Fenêtre de Hamming')

subplot(2,3,5)
grid on
HB = fft(hb,1024);
plot((0:510)/1024,20*log10(abs(HB(1:511)/HB(1))))
AXIS([0 0.5 -80 0]);
ylabel('W dB')
Title('Fenêtre de Blackman')

subplot(2,3,6)
grid on
HR = fft(hr,1024);
plot((0:510)/1024,abs(HR(1:511)/HR(1)))
AXIS([0 0.25 0 1]);
ylabel('|W| lineaire')
Title('Fenêtre rectangulaire')

% Analyse d'un sinus

N = 32;           % nombre d'échantillons
n = 0 : N-1;     % vecteur des indices (nombres entiers)

a = 12;          % amplitude de la sinusoïde
nu_o = 50;       % fréquence de la sinusoïde

nu_e = 300;      % fréquence d'échantillonnage
f_o = nu_o/nu_e; % fréquence normalisée du signal
T_e = 1/nu_e;    % période d'échantillonnage
nT_e = n*T_e;   % vecteur des instants (en seconde)

y_n = cos(2*pi*f_o*n);

figure; subplot(3,1,1) ; stem(0:N-1+5,[y_n, zeros(1,5)]); axis([0 N-1+5 -1 1]);
title('Fonction cos(2pi f_o n) échantillonnée tronquée sur N=32 points'); ylabel('x_N(n)'); grid

fmin = -1/2; fmax = 1/2;
pas_f = (fmax-fmin)/1024 ; f=(fmin:pas_f:fmax-pas_f) ;
Y = fft(y_n,1024);
subplot(3,1,2) ; plot(f,abs(fftshift(Y)))
title('X_N(e^{-j2pif})=TF\{x_N(n)\}') ; grid
Y = fft(y_n.*hamming(N)',1024);
hold on; plot(f,abs(fftshift(Y)), 'r');

```

```
pas_f = (fmax-fmin)/N ; f=(fmin:pas_f:fmax-pas_f) ;  
Y = fft(y_n);  
subplot(3,1,3) ; stem(f,abs(fftshift(Y)))  
title('X_N(k)=TFD\{x_N(n)\}') ; grid  
Y = fft(y_n.*hamming(N));  
hold on; stem(f,abs(fftshift(Y)),'r')
```

..... A COMPLETER
