

# Traitement du signal avec Scilab : analyse et synthèse des filtres numériques

Lors de séances précédentes, nous avons déjà mis en œuvre des filtres numériques (voir tp sur la convolution ainsi que sur la transmission numérique en bande de base). A chaque fois, à partir du filtre analogique équivalent, nous avons fenêtré et discrétisé la réponse impulsionnelle, le filtrage proprement dit étant réalisé grâce à une opération de convolution.

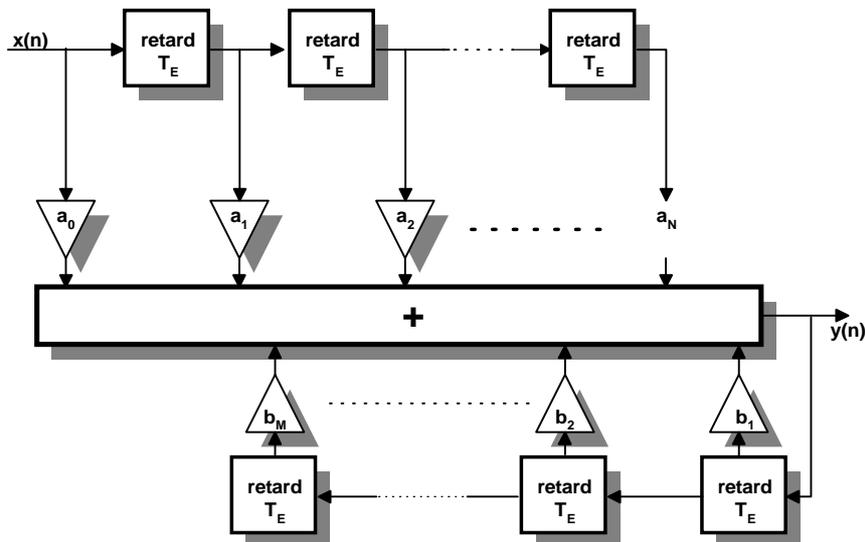
Notre objectif est maintenant de passer en revue d'autres méthodes permettant la synthèse de tels filtres. Nous aurons besoins pour cela dans un premier temps d'un bref rappel des propriétés de la transformation en « z ».

## 1. Description et classification des filtres numériques

Un filtre numérique est généralement implanté dans un ordinateur par l'algorithme correspondant à une équation récurrente (ou équation aux différences) du type :

$$\begin{aligned} y(n) &= a_0 \cdot x(n) + a_1 \cdot x(n-1) + a_2 \cdot x(n-2) + \dots + b_1 \cdot y(n-1) + b_2 \cdot y(n-2) + \dots \\ &= \sum_{k=0}^{k=N} a_k \cdot x(n-k) + \sum_{k=1}^{k=M} b_k \cdot y(n-k) \end{aligned}$$

où  $x(n)$  et  $y(n)$  représentent respectivement l'entrée et la sortie du filtre à l'instant  $n.T_E$  ( $T_E$  étant la période d'échantillonnage), tandis que les  $N$  termes  $a_k$  et les  $M$  termes  $b_k$  sont les coefficients du filtre. L'algorithme est alors équivalent à la structure ci-après :



On peut également envisager de calculer la transformée de Fourier du signal incident, de la multiplier par la réponse fréquentielle du filtre, puis de calculer la transformée inverse. Nous limiterons cependant notre étude à l'équation récurrente.

On peut alors classer les filtres de différentes manières :

**Filtre récursif :**

- C'est la forme générale définie plus haut, ou la sortie dépend des échantillons d'entrée mais aussi des échantillons de sortie des instants précédents.

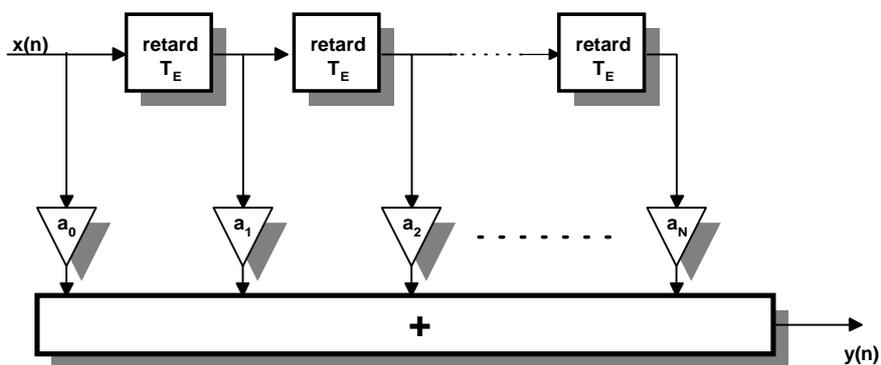
**Filtres non récursifs :**

- Dans ce cas, la sortie dépend uniquement des échantillons d'entrée et les termes  $b_k$  sont nuls. L'équation précédente devient alors :

$$y(n) = a_0 \cdot x(n) + a_1 \cdot x(n-1) + a_2 \cdot x(n-2) + \dots$$

$$= \sum_{k=0}^{k=N} a_k \cdot x(n-k)$$

- La structure équivalente est alors la suivante :



- N'étant pas contre réactionnés, ces filtres sont naturellement stables (ce qui n'est pas le cas des précédents).

Dans le cas d'un filtre non récursif, on peut remarquer que les termes  $a_k$  représentent les coefficients résultant de l'échantillonnage de la réponse impulsionnelle du filtre, et l'opération décrite ci-dessus représente la convolution de l'entrée par cette réponse.

Le nombre  $N$  de ces coefficients étant fini (pour des raisons matérielles), la réponse impulsionnelle du filtre sera donc finie également. Comme on le voit, la réponse impulsionnelle joue un rôle très important dans le filtrage numérique, et on peut envisager un second type de classement :

**Filtres à réponse impulsionnelle infinie (FRII ou IIRF) :**

- Leur réponse impulsionnelle s'étendant sur un nombre infini d'échantillons, ce sont forcément des filtres récursifs ;

**Filtre à réponse impulsionnelle finie (FRIF ou FIRF) :**

- Leur définition en fait des filtres naturellement stables. Ils correspondent généralement à un filtre non récursif, mais certaines structures de filtres récursifs ont une réponse impulsionnelle finie (voir exemples ci-après).

**2. Transformée en « z »**

**2.1. Introduction**

Comme on peut le voir, l'équation récurrente n'est pas toujours très facile à manipuler, aussi est-il souvent préférable d'introduire une transformation, dite en « z », dont l'opérateur  $z^{-1}$  correspond à un retard d'une période  $T_E$  de l'horloge d'échantillonnage ;  $z^{-n}$  correspondra alors à un retard de  $n$  période d'horloge.

L'équation récurrente d'un filtre récursif quelconque :

$$\begin{aligned}
 y(n) &= a_0 \cdot x(n) + a_1 \cdot x(n-1) + a_2 \cdot x(n-2) + \dots + b_1 \cdot y(n-1) + b_2 \cdot y(n-2) + \dots \\
 &= \sum_{k=0}^{k=N} a_k \cdot x(n-k) + \sum_{k=1}^{k=M} b_k \cdot y(n-k)
 \end{aligned}$$

devient alors :

$$\begin{aligned}
 Y(z) &= a_0 \cdot X(z) + z^{-1} \cdot a_1 \cdot X(z) + z^{-2} \cdot a_2 \cdot X(z) + \dots + z^{-1} \cdot b_1 \cdot Y(z) + z^{-2} \cdot b_2 \cdot Y(z) + \dots \\
 &= \sum_{k=0}^{k=N} z^{-k} \cdot a_k \cdot X(z) + \sum_{k=1}^{k=M} z^{-k} \cdot b_k \cdot Y(z)
 \end{aligned}$$

ou encore :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{a_0 + a_1 \cdot z^{-1} + a_2 \cdot z^{-2} + \dots}{1 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2} + \dots} = \frac{\sum_{k=0}^{k=N} z^{-k} \cdot a_k}{1 + \sum_{k=1}^{k=M} z^{-k} \cdot b_k}$$

H(z) représentant la transformée en z du filtre numérique.

## 2.2. Principales propriétés

En supposant les différentes suites convergentes, la transformation en z admet les propriétés suivantes :

### valeur initiale

$$y(0) = \lim_{z \rightarrow \infty} Y(z)$$

### valeur finale

$$y(\infty) = \lim_{z \rightarrow 1} (z-1) \cdot Y(z)$$

### sommation et multiplication par une constante

$$Z\{a \cdot y(n) + b \cdot x(n)\} = a \cdot Z\{y(n)\} + b \cdot Z\{x(n)\}$$

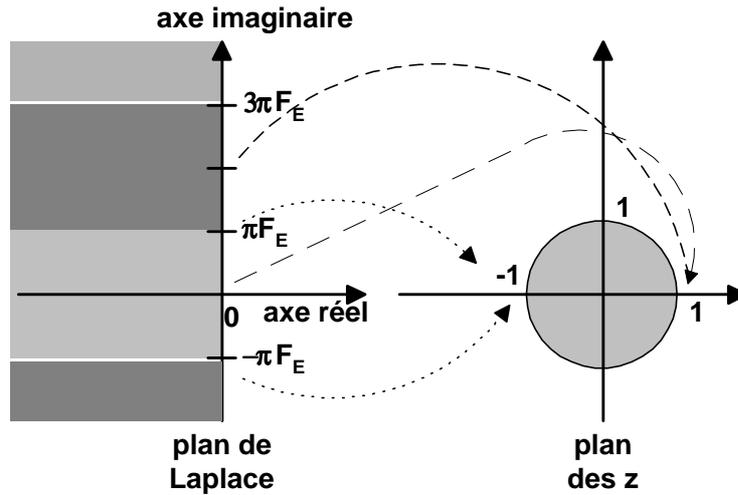
### convolution

Un produit de convolution dans le domaine temporelle devient un simple produit pour la transformation en z.

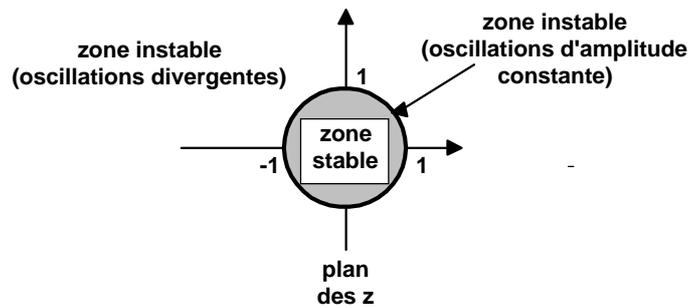
### relation avec la transformation de Laplace

L'opérateur retard  $z^{-1}$  de la transformation en z devient l'opérateur retard  $e^{-T_E p}$  pour la transformation de Laplace. A la variable  $p = \sigma + j\omega$  du plan de Laplace, correspond donc la variable z de module  $e^{\sigma T_E}$  et d'argument  $\omega T_E$ . Il en résulte que :

- les points de l'axe imaginaire du plan de Laplace ( $\sigma=0$ ) correspondront donc au cercle unité dans le plan z ;
- le parcours d'une distance  $2\pi F_E$  ( $F_E$  étant la fréquence d'échantillonnage) sur cet axe imaginaire, est équivalent à un tour complet sur le cercle ;
- les points de partie réelle positive ( $\sigma > 0$ ) du plan de Laplace seront à l'extérieur de ce cercle ;
- les points de partie réelle négative ( $\sigma < 0$ ) seront à l'intérieur du cercle.

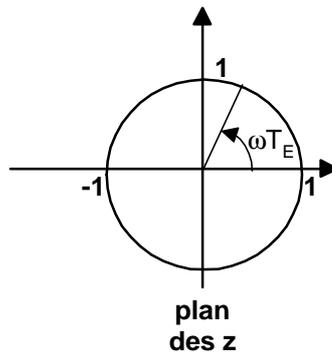


Sachant que pour qu'un système soit stable, ses pôles dans le plan de Laplace doivent être à partie réelle négative, on peut donc en déduire que :  
 Pour qu'un système soit stable, les pôles de sa transformée en z doivent être à l'intérieur du cercle unité.



**relation avec la transformation de Fourier**

En régime harmonique la variable de Laplace devient  $p=j\omega$  et l'opérateur retard  $e^{-j\omega T_E}$ ; on peut donc en déduire que la transformée de Fourier d'une séquence est la transformée en z évaluée sur le cercle unité, gradué en  $\omega T_E$ .



On retrouve la périodicité mise en évidence lors de l'étude du spectre d'un signal échantillonné ; une grandeur fréquentielle  $X(f)$  est parfois notée  $X(e^{-j\omega T_E})$ , ou encore  $X(\theta)$  ( $\theta$  variant de 0 à  $2\pi$ ) pour mettre en évidence cette périodicité si la grandeur temporelle est échantillonnée.

**2.3. Analyse de quelques filtres**

Afin de se familiariser avec ces différentes propriétés, étudions les quelques exemples ci-après.

### Filtre passe bas

Considérons le filtre passe bas suivant dont la transformée en z a pour expression :

$$H(z) = \frac{1+z^{-1}}{1-az^{-1}}$$

Ce filtre est stable si son unique pôle, de valeur « a » se trouve à l'intérieur du cercle unité. On doit donc vérifier  $|a| < 1$ .

La fréquence du signal incident n'a un sens que dans la gamme allant du continu à  $F_E/2$  (théorème de Shannon). D'après ce que nous avons vu sur la correspondance entre le plan de Fourier et le plan des z, on peut donc déduire que :

- en continu,  $f=0$  et  $z=1$ , le gain statique vaut donc  $2/(1-a)$  ;
- en haute fréquence, à  $F_E/2$ , z vaut  $-1$  et le gain est alors nul ; on obtient bien un filtre passe bas.

Pour tracer l'évolution du gain et de la phase en régime harmonique, il suffit de remplacer z par  $e^{-j\omega T_E}$  soit encore  $\cos(\omega T_E) + j \sin(\omega T_E)$ .

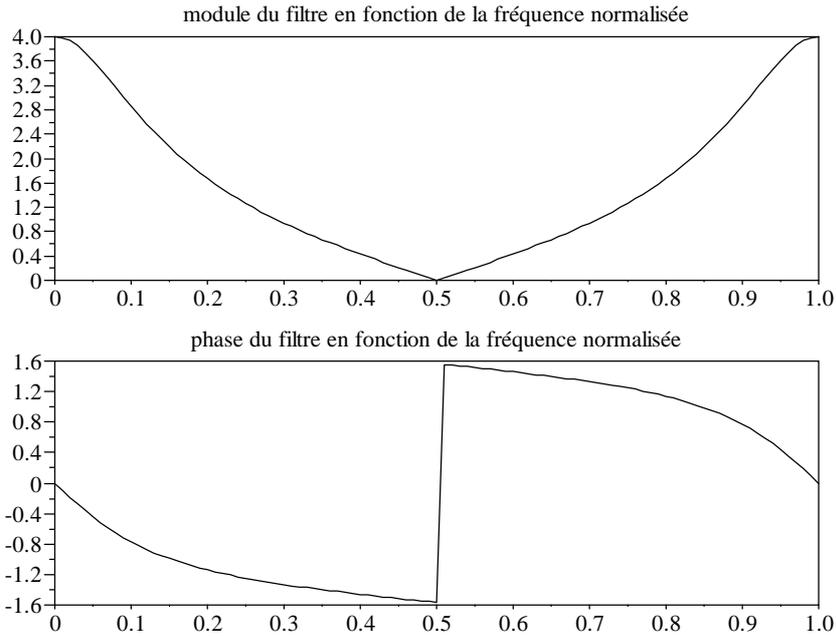
On obtient alors la valeur suivante pour H(f) :

$$H(f) = \frac{2}{1-a} \frac{1}{1+j \frac{1+a}{1-a} \operatorname{tg} \frac{\pi f}{F_E}}$$

La fréquence de coupure vaut alors :  $F_C = \frac{F_E}{\pi} \operatorname{arctg} \frac{1+a}{1-a}$

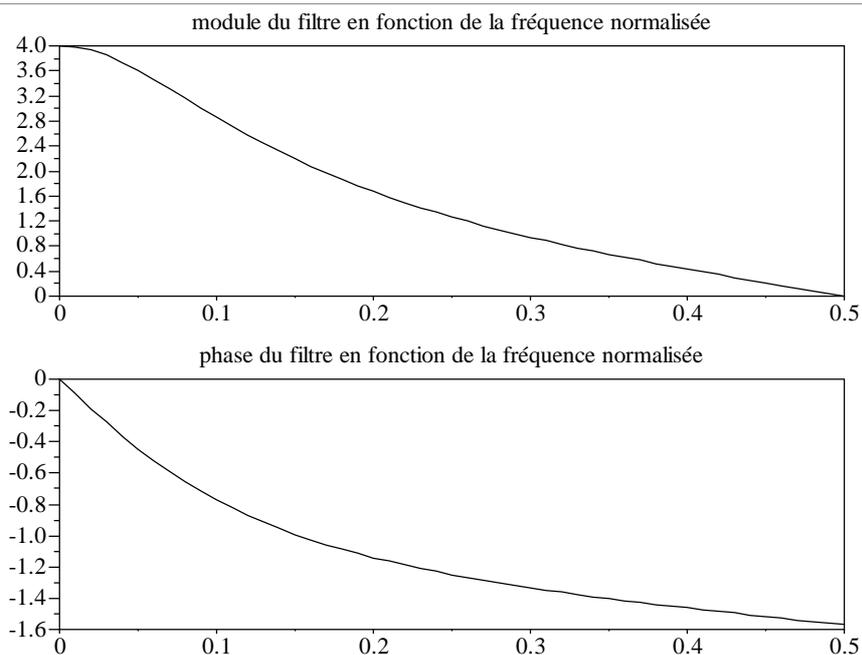
Vérifions ces calculs avec Scilab. Pour cela nous définirons le polynôme en z par ses coefficients avec l'instruction « poly » (les coefficients de poids faible en tête). Puis nous calculerons les différents points souhaités, au moyen de l'instruction « freq ».

```
clear
//
// constante
a=.5;
//
// définition du numérateur
h1=poly([1 1], 'z', 'c');
//
// définition du dénominateur
h2=poly([-a 1], 'z', 'c');
//
// gamme de fréquence normalisée par rapport à la fréquence d'échantillonnage
f=(0:.01:1);
//
// calcul des différents points de la fonction de transfert
hf=freq(h1,h2,exp(2*pi*i*f));
//
// affichage
xbasc(); xset("font size",4);
xsetech([0,0,1,.5]); plot2d(f,abs(hf))
xtitle("module du filtre en fonction de la fréquence normalisée")
xsetech([0,.5,1,.5]); plot2d(f,atan(imag(hf),real(hf)));
xtitle("phase du filtre en fonction de la fréquence normalisée")
```



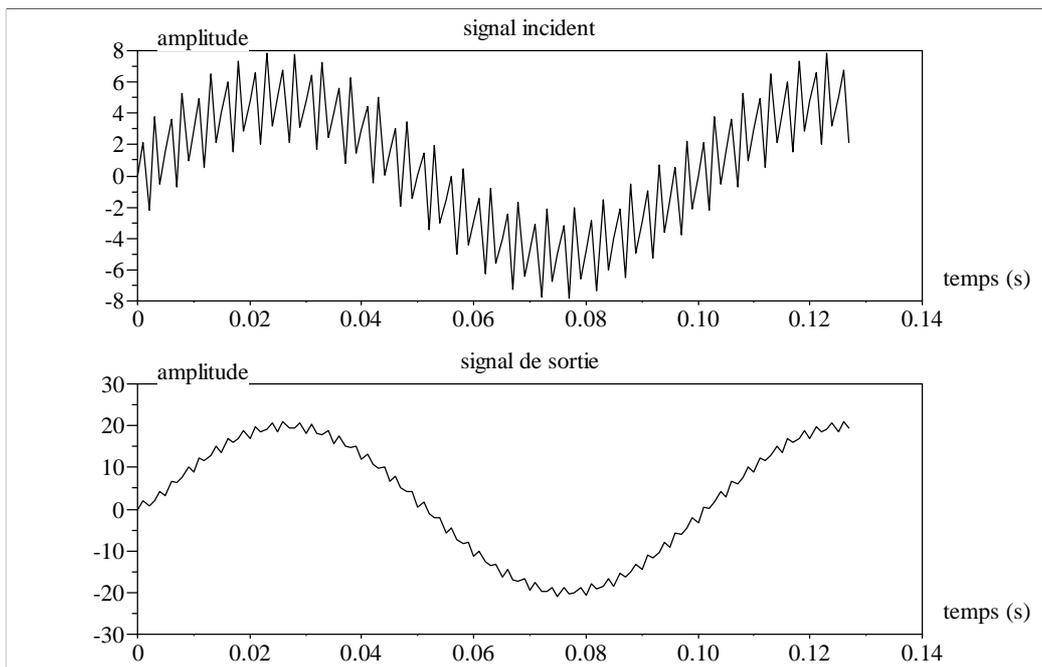
Comme on peut le constater, le spectre est périodique et seule la partie allant de 0 à 0,5 (soit de 0 à  $F_E/2$ ) nous intéresse. Modifions le programme en conséquence :

```
// gamme de fréquence normalisée par rapport à la fréquence d'échantillonnage
f=(0:.01:.5);
//
// calcul des différents points de la fonction de transfert
hf=freq(h1,h2,exp(2*%pi*i*f));
//
// affichage
xbasec(); xset("font size",4);
xsetech([0,0,1,.5]); plot2d(f,abs(hf));xtitle("module du filtre en fonction de la fréquence normalisée")
xsetech([0,.5,1,.5]); plot2d(f,atan(imag(hf),real(hf)));
xtitle("phase du filtre en fonction de la fréquence normalisée")
```



Observons maintenant la réponse à un signal temporel au moyen de l'instruction « flts ». L'utilisation de cette dernière nécessite la création d'un système linéaire par « syslin ».

```
clear
//
// constantes et vecteur temps
a=.5; TE=1e-3;
t=TE*(0:127);
//
// définition d'une entrée
x=5*sin(2*%pi*0.01/TE*t)+3*sin(2*%pi*0.4/TE*t);
//
// définition de la fonction en z
h=poly([1 1], 'z', 'c')./poly([-a 1], 'z', 'c');
//
// création d'un système linéaire
hz=syslin('d', h);
//
// filtrage
y=flts(x, hz);
// affichage
xbasc(); xset("font size", 4);
xsetech([0,0,1,.5]); plot2d(t,x); xtitle("signal incident", "temps (s)", "amplitude")
xsetech([0,.5,1,.5]); plot2d(t,y); xtitle("signal de sortie", "temps (s)", "amplitude")
```

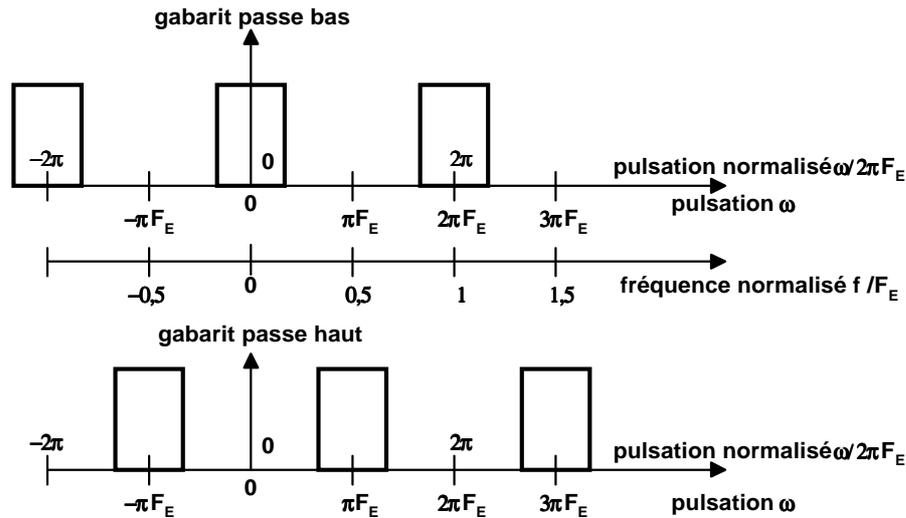


On vérifie bien que le signal haute fréquence est atténué par rapport au signal basse fréquence. Le filtrage n'est cependant pas d'une grande efficacité, il ne s'agit que d'un filtre du premier ordre.

### Filtre passe haut

A partir d'un filtre analogique de fréquence de coupure  $\omega_0$ , on peut déterminer le filtre passe haut correspondant en faisant la transformation  $p/\omega_0$  vers  $\omega_0/p$ .

Sur les gabarits fréquentiels ci-après, ont été représentés un filtre passe bas idéal puis un filtre passe haut idéal en fonction de la fréquence du signal, ainsi que de cette même fréquence normalisée par rapport à  $F_E$ .



A partir de cette figure, déduire la transformation à faire dans le plan en z pour passer d'un filtre passe bas à un filtre passe haut, et montrer que le filtre précédent devient alors :

$$H(z) = \frac{z^{-1} - 1}{z^{-1} + a}$$

Refaire alors la même étude que précédemment.

### Filtere récurisif à réponse impulsionnelle finie

Voici maintenant un exemple de filtre récurisif, qui contrairement à la majorité des filtres du même type, est à réponse impulsionnelle finie :

$$H(z) = \frac{z^{-1} - a^4 z^{-5}}{a - z^{-1}}$$

Donner la condition de stabilité sur a.

Décomposer en éléments simples H(z) et calculer la réponse impulsionnelle h(n) du filtre. Montrer qu'elle vaut :

$$\begin{aligned} h(n) &= 0 && \text{pour } n \leq 0 \\ h(n) &= a^{n-1} && \text{pour } 1 \leq n \leq 4 \\ h(n) &= 0 && \text{pour } n \geq 5 \end{aligned}$$

(la transformée inverse de  $1/(1-az^{-1})$  vaut  $a^n \cdot u(n)$  où u(n) est l'échelon unité)

Vérifier avec Scilab la validité de ce résultat.

### 3. Synthèse des filtres non récurisifs (à réponse impulsionnelle finie)

Comme nous l'avons vu, l'équation récurrente de tels filtres est de la forme :

$$\begin{aligned} y(n) &= a_0 \cdot x(n) + a_1 \cdot x(n-1) + a_2 \cdot x(n-3) + \dots \\ &= \sum_{k=0}^{k=N} a_k \cdot x(n-k) \end{aligned}$$

ce qui conduit à une transformée en z de la forme :

$$H(z) = \frac{Y(z)}{X(z)} = a_0 + a_1.z^{-1} + a_2.z^{-2} + \dots = \sum_{k=0}^{k=N} z^{-k} . a_k = \frac{\sum_{k=0}^{k=N} z^{N-k} . a_k}{z^N}$$

La réponse impulsionnelle de ces filtres est finie, comme le montre l'équation précédente ; ils sont donc inconditionnellement stables. Le dernier membre de l'équation précédente met d'ailleurs en évidence la présence de N pôles situés à l'origine du plan z, donc à l'intérieur du cercle unité.

Les coefficients du filtre représentent les coefficients de la réponse impulsionnelle discrétisée ; on conçoit aisément alors que cette dernière va jouer un rôle important dans les méthodes de synthèse. Nous allons aborder les deux principales que sont la méthode dite de la fenêtre et la méthode de l'échantillonnage en fréquence.

Avant d'aborder les méthodes de synthèse, mettons en évidence une propriété importante de ces filtres, qui est la possibilité d'avoir une phase linéaire

### 3.1. Filtres à phase linéaire

Tous les filtres à réponse impulsionnelle finie, et donc les filtres non récursifs, ont comme caractéristique de pouvoir présenter une phase linéaire. Le temps de propagation de groupe, qui est au signe près la dérivée de la phase par rapport à la pulsation, est alors constant :

$$t_p = - \frac{d\varphi(\omega)}{d\omega}$$

Cette propriété est très intéressante dans de nombreuses applications, car les composantes du signal aux diverses fréquences subissent un retard constant. La distorsion du signal est alors extrêmement faible.

Afin de simplifier l'étude nous utiliserons la pulsation réduite (normalisé par rapport à  $2\pi F_E$ ) qui évolue de 0 à  $2\pi$  lorsque la fréquence évolue de 0 à  $F_E$ .

Une phase linéaire se caractérise donc par une pente constante de la phase en fonction de la pulsation réduite, ou éventuellement un saut de  $\pi$ . Ces sauts se produisent uniquement si la fonction possède un ou plusieurs zéros sur le cercle unité. ; ils ont lieu au moment où le gain est nul, et sont donc sans conséquences pour la réponse du filtre.

Pour obtenir une phase linéaire, on montre qu'il suffit que la réponse impulsionnelle du filtre présente une symétrie ou une antisymétrie. Le tableau ci-après résume les caractéristiques de ces réponses.

| Discontinuité de phase de valeur $\pi$ pour les valeurs suivantes de la pulsation réduite |                 | Réponse impulsionnelle de N échantillons  |
|---|-----------------|---|
| $\theta=0$  | $\theta=+/-\pi$ |   |
| non   | non             | symétrique , N impair<br>symétrique, N pair<br>antisymétrique, N impair<br>antisymétrique, N pair |
| non   | oui             |   |
| oui   | oui             |   |
| oui   | non             |   |

A l'aide de Scilab, tester ces propriétés sur des réponses impulsionnelles simples (triangulaires par exemple).

### 3.2. Synthèse par la méthode des fenêtres

#### Principe de la méthode

La méthode consiste à partir du gabarit fréquentielle souhaité, par exemple un filtre passe bas idéal, à déterminer une réponse impulsionnelle réalisable. Les principaux points sont :

- choix d'un gabarit idéal ;
- périodisation de ce gabarit (périodisation due à l'échantillonnage) ;
- décomposition en série de Fourier du gabarit périodisé pour obtenir les coefficients de la réponse impulsionnelle ;
- troncature symétrique de la réponse impulsionnelle de manière à limiter le nombre d'échantillons ;

suyvant le placement de la fenêtre de troncature le filtre sera à nombre d'échantillons pair ou impair, avec le propriétés qui en découlent ;

- décalage de la réponse pour obtenir un filtre causal.

Illustrons cette méthode par la synthèse d'un filtre passe bas idéal, de pulsation réduite  $\theta=0,25$  (c'est à dire une fréquence de coupure de  $F_C=0,25.F_E$ ) et de gain basse fréquence unitaire.

Ce gabarit périodisé à la fréquence  $F_E$  devient un signal carré de rapport cyclique  $2.F_C/F_E$  de d'amplitude unitaire. La décomposition en série de Fourier (spectre bilatéral) d'un tel signal a pour expression :

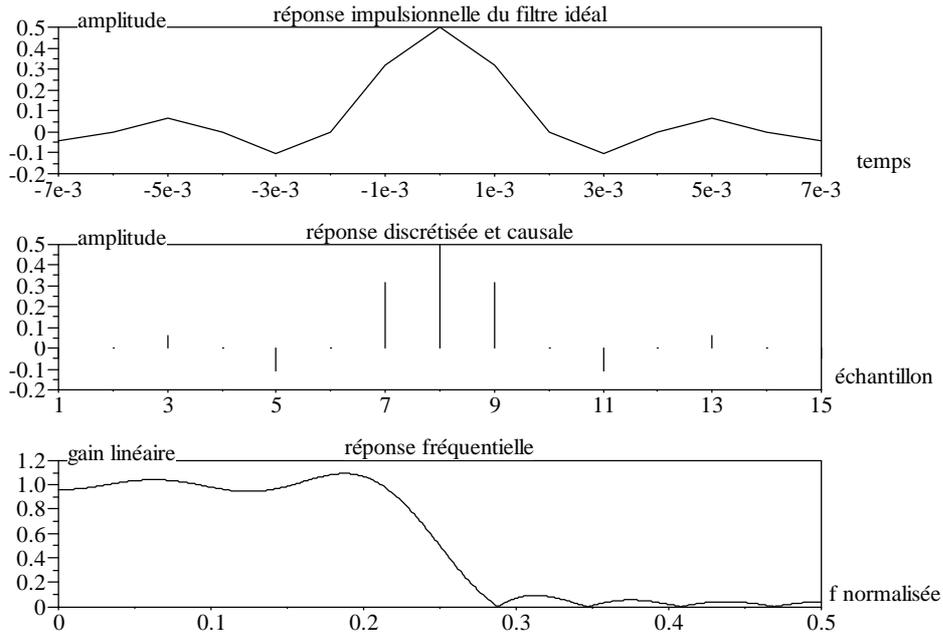
$$g(n) = \frac{1}{F_E} \int_{-F_C}^{F_C} 1.e^{2j\pi n \frac{f}{F_E}} df = \frac{2 F_C}{F_E} \frac{\sin(2\pi n \frac{F_C}{F_E})}{2\pi n \frac{F_C}{F_E}}$$

Remarque : on obtiendrait le même résultat en calculant la réponse impulsionnelle du filtre idéal (par transformation de Fourier inverse) et en considérant que l'échantillonnage introduit la multiplication de cette réponse par un coefficient  $1/T_E$ .

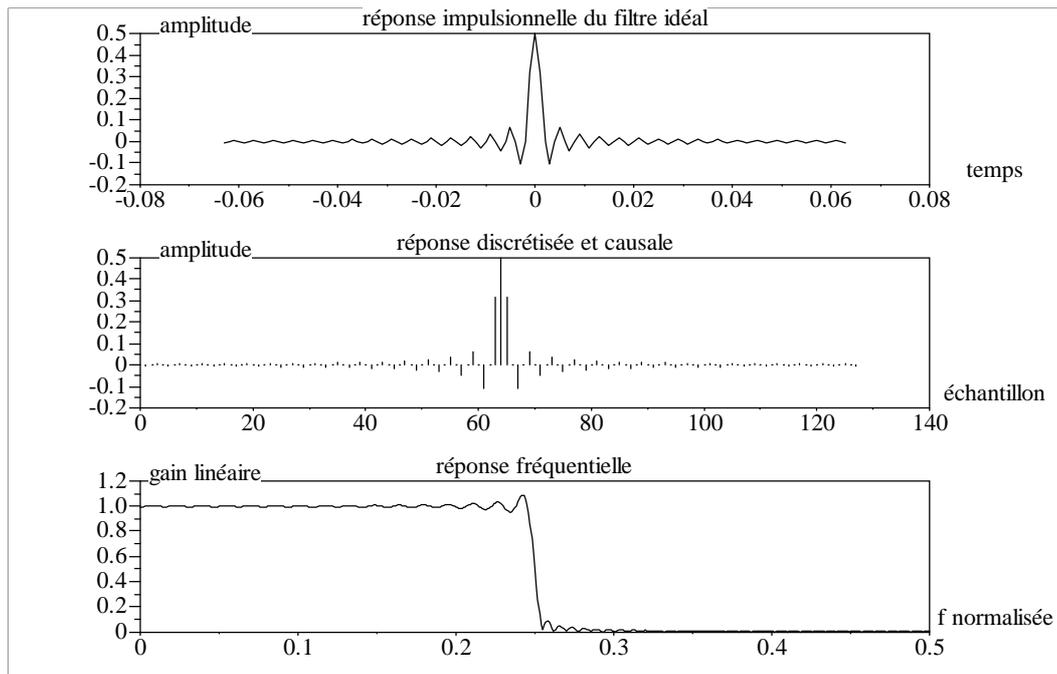
Le programme ci-après permet l'influence du nombre d'échantillons sur la réponse fréquentielle obtenue :

```
clear ;
//
// définition des constantes
Te=1e-3 ; Nb=15; Fc=.25/Te;
//
// paramètres "temps"
t=Te*(-(Nb-1)/2 : (Nb-1)/2);
//
// réponse impulsionnelle
g=2*Fc*Te*(sin(2*pi*t*Fc)/(2*pi*t*Fc+(t==0)))+(t==0));
//
// calcul de la réponse fréquentielle à partir le transformée en z
f=(0:.001:.5);
h=poly(g,'z','c');
H=freq(h,1,exp(2*pi*i*f));
//
// affichage
xbasc(); xset("font size",4);
xsetech([0,0,1,.3]); plot2d(t,g);
xtitle("réponse impulsionnelle du filtre idéal","temps","amplitude") ;
//
xsetech([0,1/3,1,.3]); plot2d3((1:length(g)),g);
xtitle("réponse discrétisée et causale","échantillon","amplitude") ;
//
xsetech([0,2/3,1,.3]); plot2d(f,abs(H));
xtitle("réponse fréquentielle","f normalisée","gain linéaire") ;
```

Résultats obtenus pour 15 échantillons :



Résultats obtenus pour 127 échantillons.



Comme nous pouvons le remarquer, la réponse fréquentielle obtenue n'est pas tout à fait celle attendue et présente de nombreuses ondulations, dans la bande passante comme dans la bande atténuée ; d'autre part la coupure n'est pas aussi raide que celle du filtre idéal.

Ces caractéristiques sont les conséquences de la troncature de la réponse impulsionnelle qui se trouve multipliée alors par une fonction porte. Dans le domaine fréquentiel, cela se traduit par une convolution avec une fonction en sinus cardinal (la transformée de Fourier d'une porte).

Ce phénomène est connu sous le nom de phénomène de Gibbs.

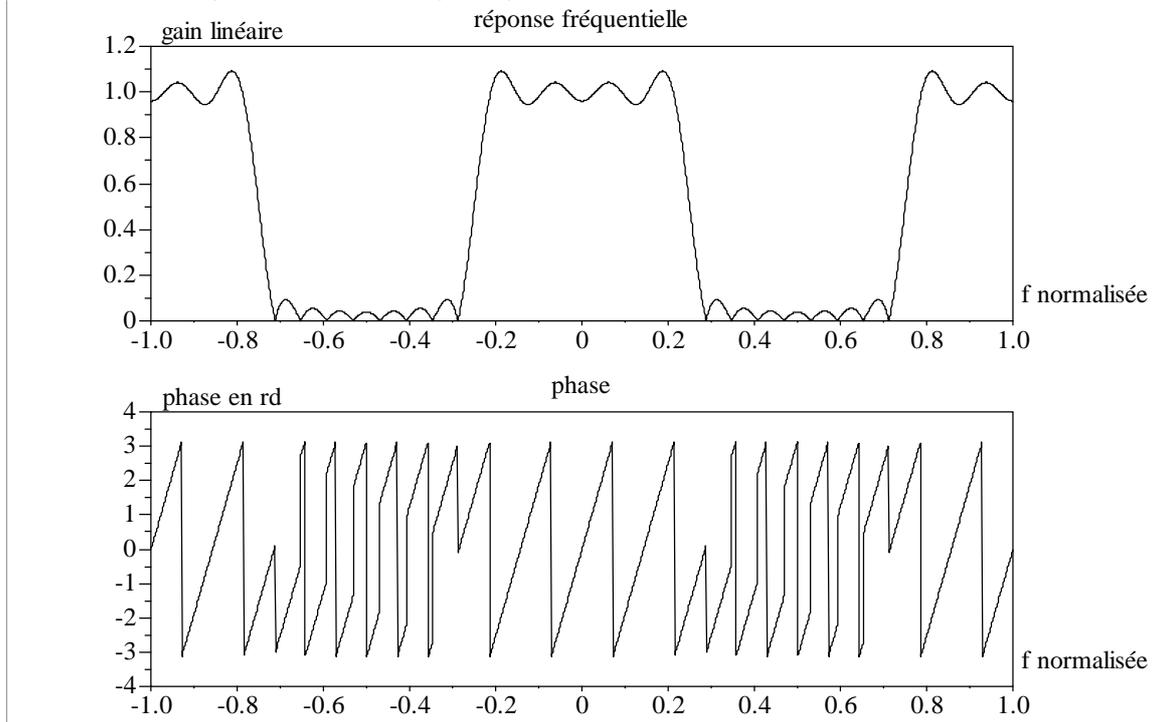
On peut remarquer que le choix d'un nombre d'échantillons important pour synthétiser le filtre conduit à une coupure plus raide. Cependant, quel que soit ce nombre, l'amplitude des oscillations reste identique.

### Linéarité de la phase

Observons maintenant la phase de notre filtre.

```
clear ;
//
// définition des constantes
Te=1e-3 ; Nb=15; Fc=.25/Te;
//
// paramètres "temps"
t=Te*(-(Nb-1)/2 : (Nb-1)/2);
//
// réponse impulsionnelle
g=2*Fc*Te*(sin(2*pi*t*Fc)/(2*pi*t*Fc+(t==0))+ (t==0));
//
// calcul de la réponse fréquentielle à partir le transformée en z
f=(-1:.001:1);
h=poly(g,'z','c');
H=freq(h,1,exp(2*pi*i*f));
//
// affichage
xbasc(); xset("font size",4);
xsetech([0,0,1,.5]); plot2d(f,abs(H));xtitle("réponse fréquentielle","f normalisée","gain linéaire")
xsetech([0,.5,1,.5]); plot2d(f,atan(imag(H), real(H)));
xtitle("phase","f normalisée","phase en rd")
```

Cas d'un nombre impair d'échantillons (N=15) :

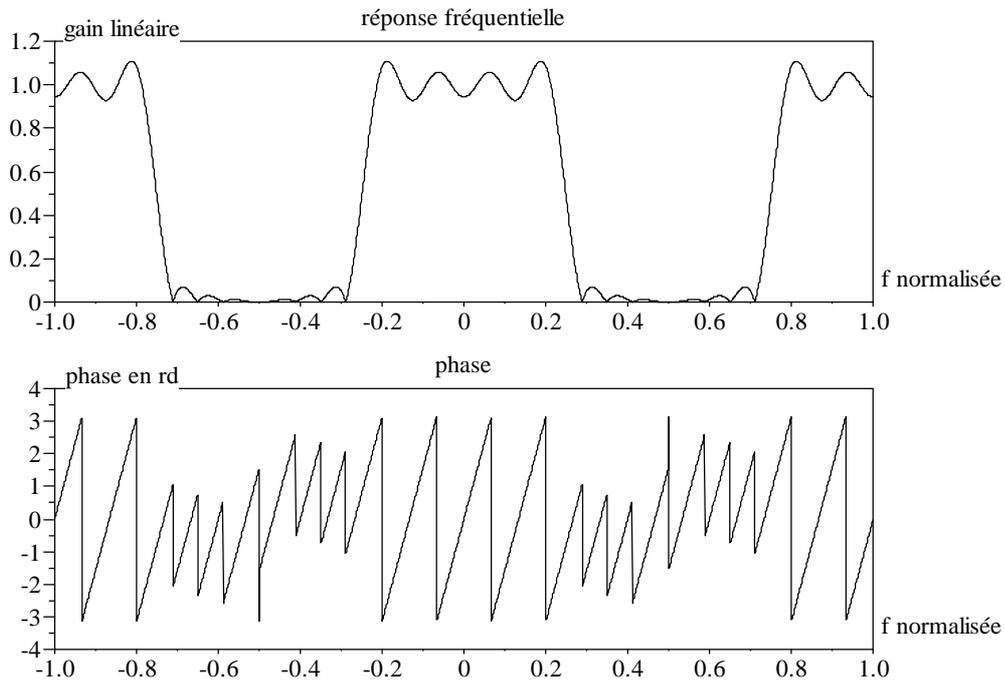


On obtient bien une phase évoluant linéairement en fonction de la fréquence ; on peut noter deux types de discontinuités sur la courbe :

- des discontinuités de  $2\pi$  dues au fait que l'affichage ne se fait qu'entre  $-\pi$  et  $+\pi$  ;
- des discontinuités de  $\pi$  lorsque le gain est nul, comme le prévoyait le tableau précédent (voir paragraphe sur la phase linéaire), discontinuités dues à la présence, sur la fonction en z, de zéros sur le cercle unité.

Cas d'un nombre pair d'échantillons (N=16) :

```
clear ;
//
// définition des constantes
Te=1e-3 ; Nb=16; Fc=.25/Te;
//
// paramètres "temps"
t=Te*(-(Nb-1)/2 : (Nb-1)/2);
//
// réponse impulsionnelle
g=2*Fc*Te*(sin(2*pi*t*Fc)/(2*pi*t*Fc+(t==0))+t==0);
//
// calcul de la réponse fréquentielle à partir le transformée en z
f=(-1:.001:1);
h=poly(g,'z','c');
H=freqz(h,1,exp(2*pi*i*f));
//
// affichage
xbasec(); xset("font size",4);
xsetech([0,0,1,.5]); plot2d(f,abs(H));xtitle("réponse fréquentielle","f normalisée","gain linéaire")
xsetech([0,.5,1,.5]); plot2d(f,atan(imag(H), real(H)));
xtitle("phase","f normalisée","phase en rd")
```



On retrouve bien le même phénomène, avec comme nuance, la présence d'une discontinuité de  $\pi$  particulière lorsque la fréquence vaut  $\pm F_E/2$  (c'est à dire  $\pm 0,5$  pour la fréquence normalisée et  $\pm \pi$  pour la pulsation normalisée), le nombre d'échantillons étant pair.

### Choix d'une fenêtre

Nous avons pu observer le phénomène de Gibbs, résultat du fenêtrage de la réponse impulsionnelle. Comme pour la transformée de Fourier discrète, par un choix de fenêtre approprié, il est possible de « modeler » les ondulations de la réponse fréquentielle.

On cherchera pour ces fenêtres, une transformée de Fourier ayant un lobe principal le plus étroit possible, ainsi que des lobes secondaires contenant aussi peu d'énergie que possible. La forme répondant le mieux à cette définition est l'impulsion de Dirac, dont la transformée de Fourier inverse correspond à un signal continu (donc pas de fenêtrage du tout).

La réduction des oscillations s'obtient au prix d'une diminution de la pente du gain du filtre dans la zone de transition.

Citons quelques noms de fenêtres classiques : rectangulaire, Barlett, Hanning, Hamming, Blackman, Kaiser, Chebyshev.

Le logiciel Scilab propose pour le calcul des filtres par cette méthode, l'instruction « wfir » (pour windows FIR) dont la syntaxe est :

```
[wft, wfm, fr]= wfir (ftype, forder, cfreq, wfype, fpar)
```

Cette fonction renvoie :

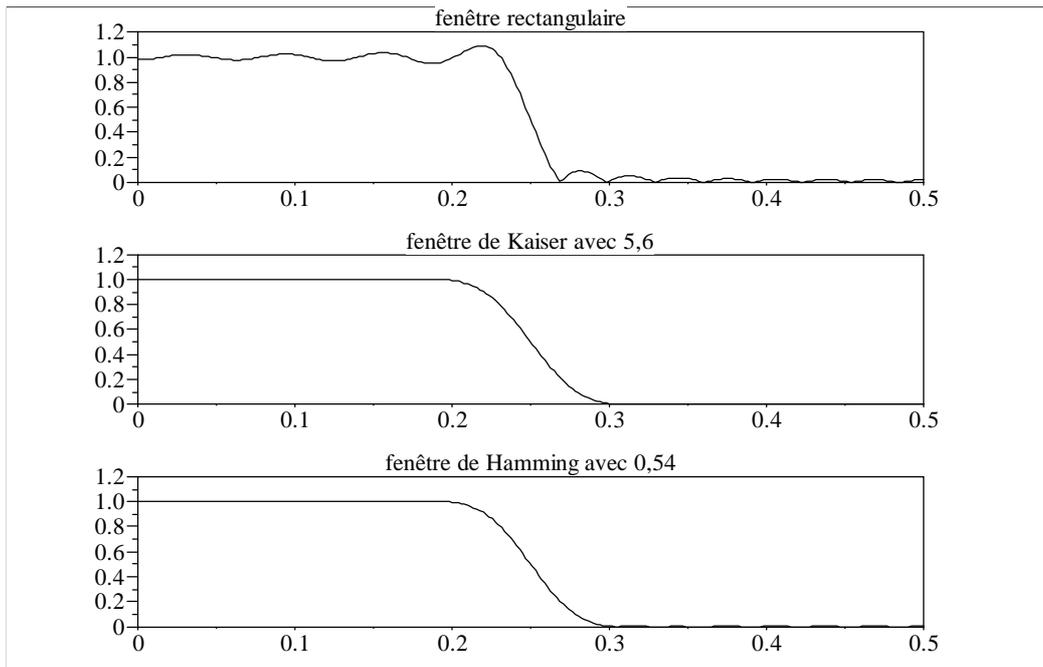
- un vecteur wft contenant les coefficients du filtre ;
- un vecteur wfm sur 256 points contenant la réponse fréquentielle ;
- un vecteur fr contenant sur 256 points contenant les fréquences associées à wfm, de 0 à 0,5 (valeurs réduites).

Il faut lui fournir :

- ftype le type de filtre, lp (passe bas), hp (passe haut), bp (passe bande) ou sb (coupe bande) ;
- le nombre de points « foder » du filtre ;
- la ou les fréquences de coupures normalisées cfreq ;
- le type de fenêtre choisi re (rectangulaire), tr (triangulaire), hm (Hamming), hn (Hanning), kr (Kaiser), ou ch (Chebychev) ;
- certains arguments supplémentaires fonction des fenêtres choisies.

Voici un exemple d'utilisation avec le même filtre que précédemment, pour 33 échantillons et différents types de fenêtres :

```
clear ;
//
// fenêtre rectangulaire
[hrc,hrm,fr]=wfir('lp',33,[.25 .4],'re',[0 0]);
//
// fenêtre de Kaiser avec  $\beta=5,6$ 
[hkc,hkm,fk]=wfir('lp',33,[.25 .4],'kr',[5.6 0]) ;
//
// fenêtre de Hamming avec  $\alpha=0,54$ 
[hhc,hhm,fh]=wfir('lp',33,[.25 .4],'hm',[0.54 0]) ;
//
// affichage
xbasc() ; xset ("font size",4);
xsetech([0,0,1,.3]); plot2d(fr,hrm);xtitle("fenêtre rectangulaire");
xsetech([0,1/3,1,.3]); plot2d(fk,hkm);xtitle("fenêtre de Kaiser avec 5,6");
xsetech([0,2/3,1,.3]); plot2d(fh,hhm);xtitle("fenêtre de Hamming avec 0,54");
```



Reprendre les études précédentes pour un filtre passe bande avec différentes fenêtres.

### 3.3. Méthode de l'échantillonnage en fréquence

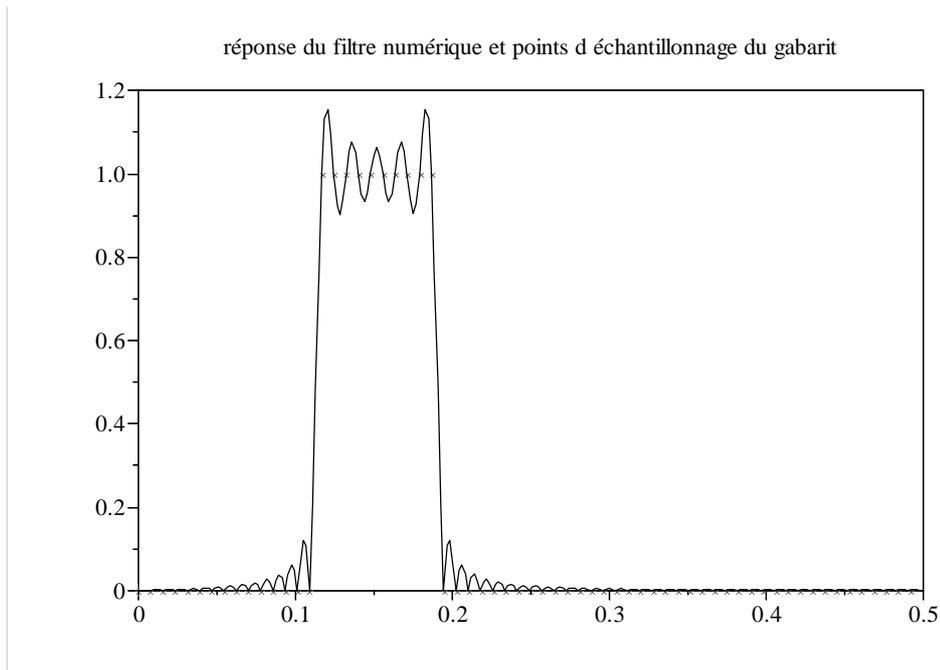
Cette méthode présente la démarche inverse de la précédente : c'est cette fois la réponse fréquentielle souhaitée qui va être échantillonnée.

Si on souhaite réaliser le filtrage sur le calculateur par transformation de Fourier, on obtient directement les coefficients ; si on souhaite réaliser le filtrage au moyen de l'équation récurrente, il faut prendre la transformée de Fourier inverse des coefficients obtenus.

Le programme ci-après illustre cette méthode pour un filtre passe bande.

Le gabarit analogique est échantillonné et donne  $H_a$ . La fonction « fsfirin » de Scilab permet de calculer les points intermédiaires de la réponse fréquentielle du filtre numérique résultant.

```
clear
//
// gabarit analogique synthétisé du filtre
Ha=[0*ones(1,15) ones(1,10) 0*ones(1,39)];
//
// calcul de la réponse fréquentielle réelle
Hd=fsfirin(Ha,1);
//
// paramètres d'affichage
fd=0.5/length(Hd)*(0:length(Hd)-1);
fa=0.5/length(Ha)*(0:length(Ha)-1);
//
// affichage
xbasc();xset("font size",4);
plot2d(fa,abs(Ha),style=-2);
plot2d(fd,abs(Hd));
xtitle("réponse du filtre numérique et points d'échantillonnage du gabarit");
```



On peut noter que le filtre numérique calculé passe bien par les points d'échantillonnage, mais présente une ondulation importante en dehors de ces points. Cette ondulation peut être réduite en échantillonnant des points dans la zone de transition.

#### 4. Synthèse des filtres récursifs à réponse impulsionnelle infinie

L'équation récurrente prend pour ces filtres la forme la plus générale possible :

$$y(n) = a_0 \cdot x(n) + a_1 \cdot x(n-1) + a_2 \cdot x(n-2) + \dots + b_1 \cdot y(n-1) + b_2 \cdot y(n-2) + \dots$$

$$= \sum_{k=0}^{k=N} a_k \cdot x(n-k) + \sum_{k=1}^{k=M} b_k \cdot y(n-k)$$

et l'expression de la transformation en z est alors :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{a_0 + a_1 \cdot z^{-1} + a_2 \cdot z^{-2} + \dots}{1 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2} + \dots} = \frac{\sum_{k=0}^{k=N} z^{-k} \cdot a_k}{1 + \sum_{k=1}^{k=M} z^{-k} \cdot b_k}$$

Contrairement aux précédents, ces filtres possèdent des pôles autres que ceux l'origine du plan z, et présente donc une forte ressemblance avec les filtres analogiques. De nombreuses méthodes de synthèse partent alors de la réponse fréquentielle classique d'un filtre analogique (type Butterworth Chebychev etc..) et opèrent une transformation de la variable p.

Nous examinerons les deux méthodes de synthèse les plus courantes, à savoir l'invariance impulsionnelle et la transformation bilinéaire.

##### 4.1. Méthode de l'invariance impulsionnelle

Le principe consiste à partir de la réponse impulsionnelle connue d'un filtre analogique, mais contrairement à la méthode de la fenêtre pour les FRIF, on n'opère cette fois aucune troncature ; tous les termes de la réponse sont intégrés dans une sommation en z, à partir de laquelle on détermine la fonction de transfert générale.

Prenons l'exemple classique d'un filtre passe bas du premier ordre. Sa réponse fréquentielle est ;

$$H(j\omega) = \frac{1}{1 + j \frac{\omega}{\omega_0}}$$

$\omega_0$  étant la pulsation de coupure.

La réponse impulsionnelle d'un tel filtre a pour expression :

$$h(t) = \omega_0 e^{-\omega_0 t}$$

La réponse impulsionnelle du filtre numérique sera alors :

$$h_d(t) = A \omega_0 \sum_0^{\infty} e^{-n \omega_0 T_E} \cdot \delta(t - nT_E)$$

$T_E$  étant la période d'échantillonnage et A un coefficient permettant éventuellement d'ajuster le gain basse fréquence.

La transformation en z d'une telle suite a alors pour expression :

$$h_d(z) = A \omega_0 \sum_0^{\infty} z^{-n} e^{-n \omega_0 T_E}$$

On rappelle d'autre part l'expression de convergence de la série ci-après :

$$\sum_0^{\infty} a^n = \frac{1}{1-a} \quad \text{si } |a| < 1$$

d'où :

$$h_d(z) = A \omega_0 \sum_0^{\infty} z^{-n} e^{-n \omega_0 T_E} = A \omega_0 \frac{1}{1 - e^{-\omega_0 T_E} z^{-1}}$$

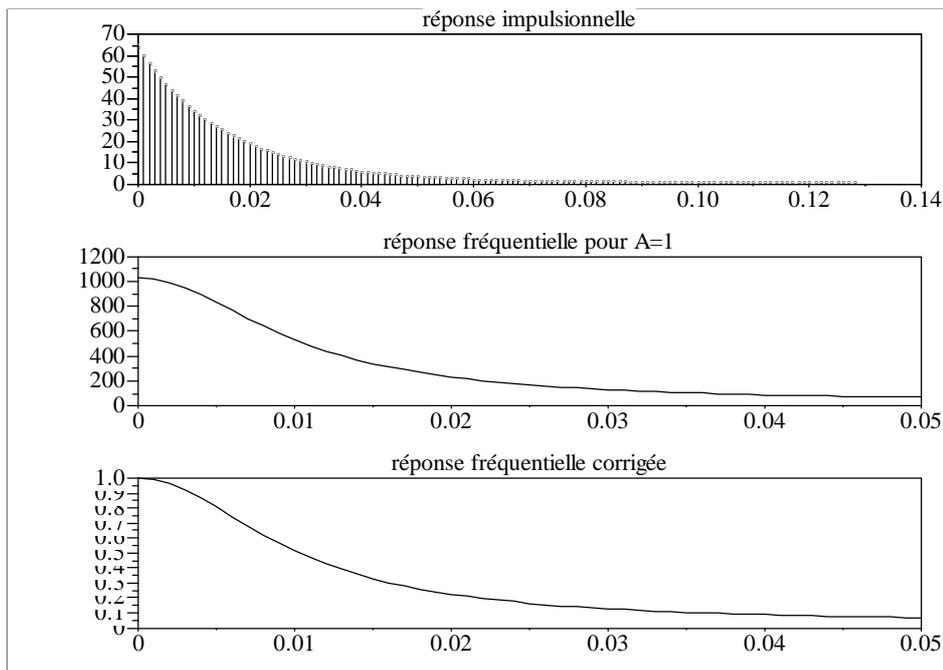
On obtient ainsi directement l'expression de la transformée en z.

Le passage de la réponse impulsionnelle  $h_d(t)$  échantillonnée à la réponse fréquentielle introduit un coefficient multiplicateur ; en effet, le  $h_d(t)$  est le produit de  $h(t)$  par un peigne de Dirac de période  $T_E$ . La transformée de Fourier de  $h_d(t)$  sera donc la transformée de  $h(t)$  convoluée par la transformée du peigne de Dirac, ce qui équivaut à périodiser la transformée de  $h(t)$  et à la multiplier par  $T_E$ . Si on souhaite malgré tout obtenir un gain statique unitaire pour le filtre numérique, on peut remplacer A dans notre cas par  $(1 - e^{-\omega_0 T_E}) / \omega_0$ .

Le programme suivant illustre la méthode décrite précédemment.

```
//
clear
//
// définition des constante
Te=1e-3; w0=.01*2*%pi/Te; A=1;
//
// variable temps
t=Te*(0:128);
//
// réponse impulsionnelle
h=w0*exp(-w0*t);
//
// description du filtre par la transformée en z, numérateur puis dénominateur
H1n=A*w0*poly([0 1], 'z', 'c'); H1d= poly([-exp(-w0*Te) 1], 'z', 'c');
//
// description d'un second filtre en ajustant le gain basse fréquence
A=(1-exp(-Te*w0))/w0;
//
H2n=A*w0*poly([0 1], 'z', 'c'); H2d= poly([-exp(-w0*Te) 1], 'z', 'c');
//
//
```

```
// calcul de la réponse fréquentielle des deux filtres précédents
f=(0:.001:.05);
hdf1=freq(H1n,H1d,exp(2*pi*i*f));
hdf2=freq(H2n,H2d,exp(2*pi*i*f));
//
// affichage
xbasec(); xset("font size",4);
xsetech([0,0,1,.3]);plot2d(t,h, style=-9);plot2d3(t,h);xtitle("réponse impulsionnelle");
xsetech([0,1/3,1,.3]);plot2d(f,hdf1);xtitle("réponse fréquentielle pour A=1");
xsetech([0,2/3,1,.3]);plot2d(f,hdf2);xtitle("réponse fréquentielle corrigée");
```



#### 4.2. Transformation bilinéaire

L'idée de cette méthode est de partir d'un gabarit analogique fréquentiel classique (filtre de Butterworth, Chebychev etc...), puis d'effectuer une transformation sur la variable « p » de manière à obtenir la variable « z ».

La formule de passage peut être obtenue en considérant la correspondance entre une opération d'intégration dans le domaine analogique de Laplace, où elle se traduit par une fonction de transfert :

$$\frac{Y(p)}{X(p)} = \frac{1}{p}$$

et la même opération dans le domaine numérique, où la méthode dite des trapèzes permet de déterminer l'équation récurrente d'un intégrateur par :

$$y(n) = \frac{x(n) + x(n-1)}{2} \cdot T_E + y(n-1)$$

d'où une fonction de transfert en z qui a pour expression :

$$\frac{Y(z)}{X(z)} = \frac{T_E}{2} \frac{1+z}{1-z}$$

On obtient donc une correspondance entre p et z :

$$p \mapsto \frac{2}{T_E} \cdot \frac{1-z^{-1}}{1+z^{-1}}$$

En régime harmonique, on peut ensuite exprimer la variable  $z = e^{-j\omega T_E} = e^{-j\theta}$  ( $\theta$  étant la pulsation réduite) en fonction de la variable  $p = j\Omega$  et trouver une relation entre la pulsation analogique et la

pulsation numérique. On obtient alors la relation :

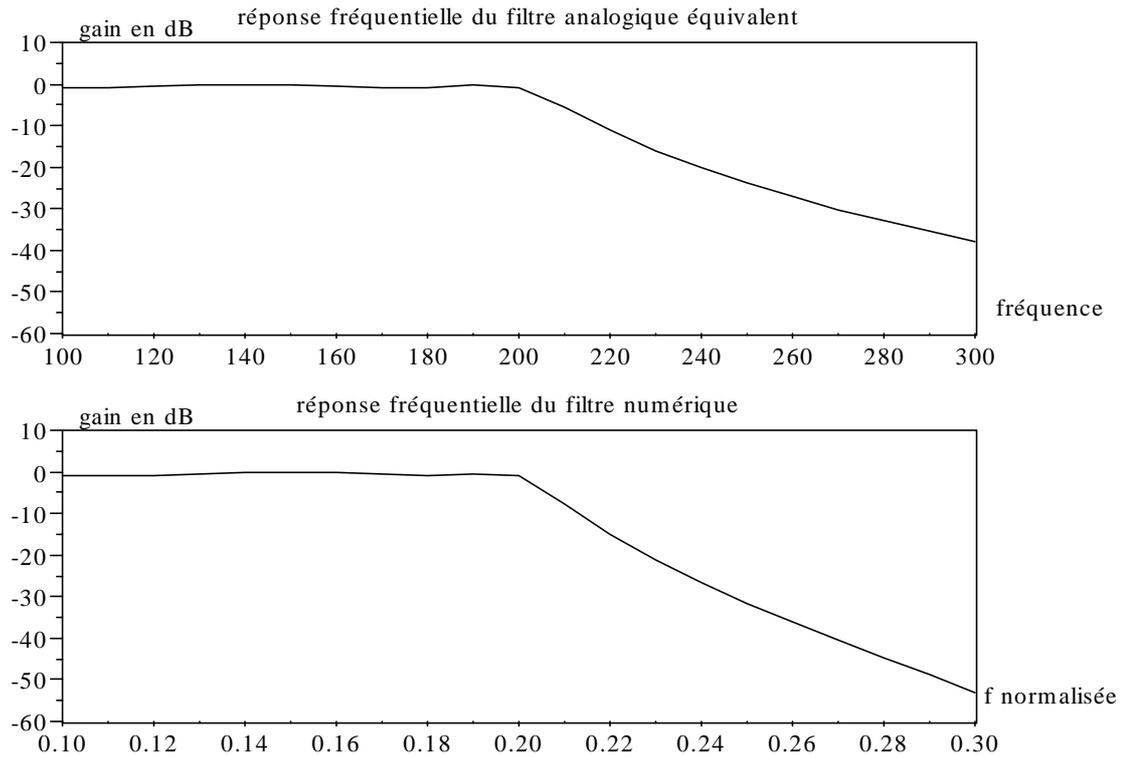
$$\Omega = \frac{2}{T_E} \operatorname{tg} \frac{\theta}{2}$$

Cette relation montre que la transformation produit une déformation de l'échelle des fréquence du spectre. On calculera donc le filtre numérique à partir d'un filtre numérique dont les fréquences remarquables auront été adaptées au moyen de la relation précédente.

Le programme suivant donne un exemple de calcul, pour un filtre passe bas de fréquence de coupure de 200 Hz, le système étant échantillonné à 1 kHz. A partir d'un filtre passe bas analogique de Chebychev, du 6<sup>ème</sup> ordre, dont la fréquence de coupure à été adaptée, on effectue la changement de variable avec l'instruction « horner », à laquelle il faut préciser la fonction d'origine et la transformation souhaitée.

L'affichage permet de comparer la réponse fréquentielle du filtre obtenu et de son équivalent analogique (aux fréquences non modifiées).

```
clear
//
// définition des constantes
Te=1e-3; Fcd=200;
//
// filtre analogique équivalent
Ga=analpf(6,'cheb1',[.1,0],2*%pi*Fcd);
//
// filtre analogique pour le calcul
teta=Fcd*(2*Te)*%pi;
Fcad=(2/Te*tan(teta/2))/(2*%pi);
Gad=analpf(6,'cheb1',[.1,0],2*%pi*Fcad);
//
// définition de la variable z
z=poly(0,'z');
//
// transformation bilinéaire
Gd=horner(Gad,2/Te*(z-1)/(z+1));
//
// calcul des points du filtre analogique équivalent
fa=(100:10:300);
Gain_a=freq(Ga(2),Ga(3),%i*2*%pi*fa);
//
// calcul des points du filtre numérique
fd=(.1:.01:.3);
Gain_d=freq(Gd(2),Gd(3),exp(%i*2*%pi*fd));
//
// affichage
xbasc(); xset("font size",4);
//
xsetech([0,0,1,.5]);plot2d(fa,20*log10(abs(Gain_a)), rect=[100, -60 300, 10]);
xtitle("réponse fréquentielle du filtre analogique équivalent","fréquence", "gain en dB");
//
xsetech([0,1/2,1,.5]);plot2d(fd,20*log10(abs(Gain_d)), rect=[0.1, -60 0.3, 10]);
xtitle("réponse fréquentielle du filtre numérique","f normalisée","gain en dB");
```



## **Bibliographie**

Une introduction au traitement du signal par A.W.M. Van Oen Eden et N.A.M. Vertoeckx chez Masson  
Traitement numérique du signal par G. Blanchet et M. Charbit chez Hermes