



Filtres Numériques

Généralités



Transformée en Z

Cas continu :

$$e(t) = s(t) + RC \frac{ds(t)}{dt} \quad H(p) = \frac{S(p)}{E(p)} \quad H(f) = H(p) \Big|_{p=j2\pi f}$$

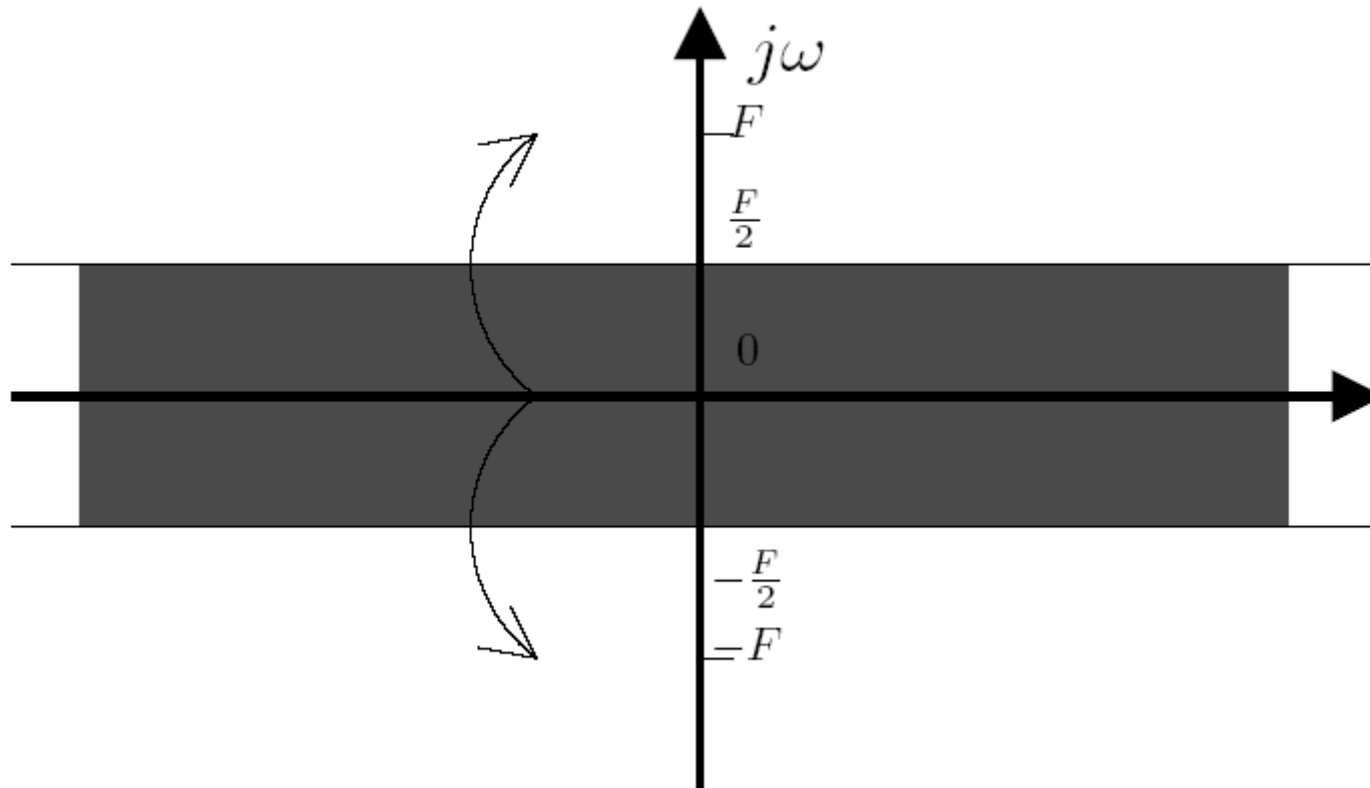
discrétisation :

$$e[nT_e] = s[nT_e] + RC \frac{s[nT_e] - s[(n-1)T_e]}{T_e}$$
$$H(z) = \frac{S(z)}{E(z)} \quad H("f") = H(z) \Big|_{z=e^{j2\pi f}}$$

Transformée de Laplace d'un signal échantillonné :

$$X_e(p) = \int_{-\infty}^{+\infty} x_e(t) e^{-pt} dt \quad \text{avec} \quad x_e(t) = x(nT_e)$$

$$X_e(p) = \sum_{n=-\infty}^{+\infty} x_e[n] e^{-npT_e}$$

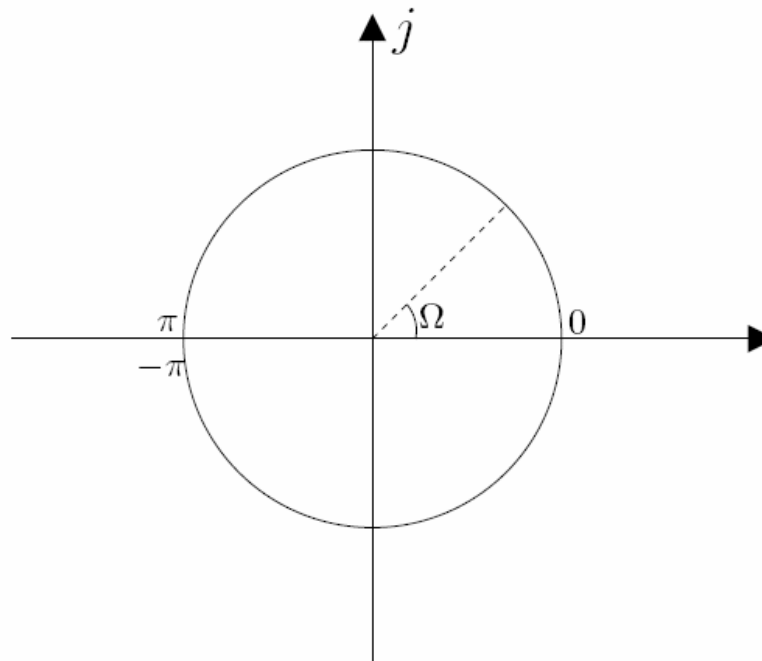




Transformée en Z

$$X_e(p) = \sum_{n=-\infty}^{+\infty} x_e[n] e^{-npT_e} = \sum_{n=-\infty}^{+\infty} x_e[n] z^{-n} \quad \text{avec } z = e^{pT_e}$$

$$X(z) = \sum_{n=-\infty}^{+\infty} x[n] z^{-n}$$





Propriétés

linéarité

$$w[n] = a x[n] + b y[n] \xrightarrow{TZ} W(z) = a X(z) + b Y(z)$$

Décalage temporel (théorème du retard)

$$w[n] = x[n - m] \xrightarrow{TZ} W(z) = z^{-m} X(z)$$

Convolution (théorème de)

$$w[n] = x[n] * y[n] \xrightarrow{TZ} W(z) = X(z) Y(z)$$



Systemes LIT

Fonction de transfert discrète :

$$H(z) = \sum_{n=-\infty}^{+\infty} h[n] z^{-n}$$

Causalité

$$H(z) = \sum_{n=0}^{+\infty} h[n] z^{-n}$$

$$y[n] = \sum_{m=0}^{+\infty} h[m] x[n-m]$$

Stabilité

$$\sum_{n=0}^{+\infty} |h[n]| < \infty$$



Filtres numériques

Équations aux différences

$$y[n] = - \sum_{i=1}^p a_i y[n-i] + \sum_{i=0}^q b_i x[n-i]$$

Fonctions de transfert rationnelle

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^q b_i z^{-i}}{1 + \sum_{i=1}^p a_i z^{-i}}$$



...(suite)

Pôles et zéros

$$H(z) = \frac{N(z)}{D(z)} = \frac{\sum_{i=0}^q b_i z^{-i}}{1 + \sum_{i=1}^p a_i z^{-i}} = \frac{\prod_{i=1}^q (1 - \beta_i z^{-1})}{\prod_{i=1}^p (1 - \alpha_i z^{-1})}$$

Zéros de transmission

$$z = \beta_i \Rightarrow N(z) = 0 \Rightarrow H(z) = 0$$

Pôles

$$z = \alpha_i \Rightarrow D(z) = 0 \Rightarrow H(z) \rightarrow \infty$$

[stable si $|\alpha_i| < 1$]



Fonction de transfert :

amplitude, phase et temps de groupe

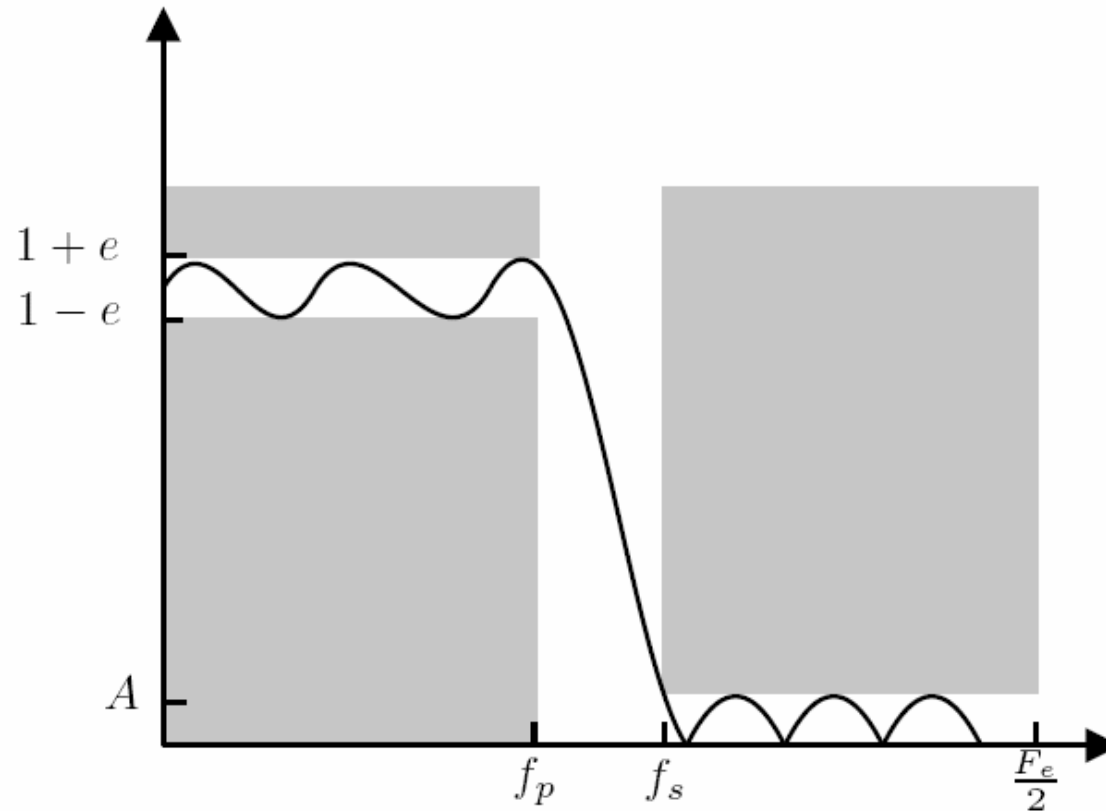
$$H(z) = \frac{N(z)}{D(z)} = \frac{\prod_{i=1}^q (1 - \beta_i z^{-1})}{\prod_{i=1}^p (1 - \alpha_i z^{-1})} \Bigg|_{z=e^{j2\pi f}} = \frac{N(f)}{D(f)}$$

$$\frac{N(f)}{D(f)} = \frac{|N(f)| e^{j\theta(f)}}{|D(f)| e^{j\varphi(f)}} = \frac{|N(f)|}{|D(f)|} e^{j(\theta(f) - \varphi(f))}$$

$$T(f) = -\frac{1}{2\pi} \frac{d(\theta(f) - \varphi(f))}{df}$$

Spécifications en amplitude

Gabarit (linéaire)



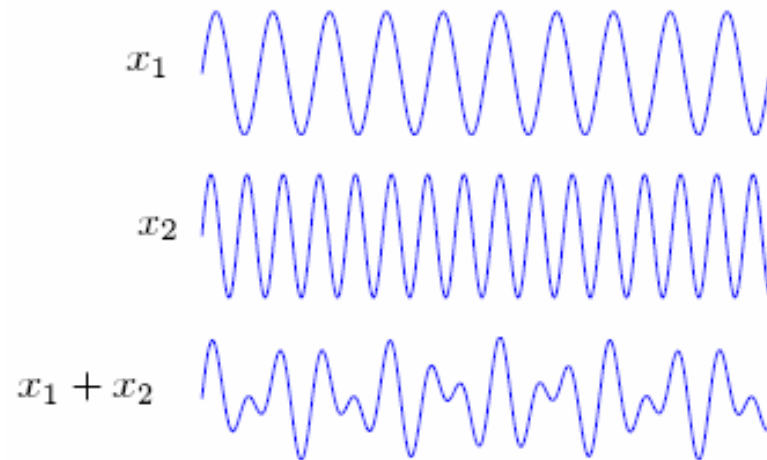
$$R_p (dB) = 20 \log_{10} \left(\frac{1+e}{1-e} \right)$$

$$R_s (dB) = 20 \log_{10} (A)$$

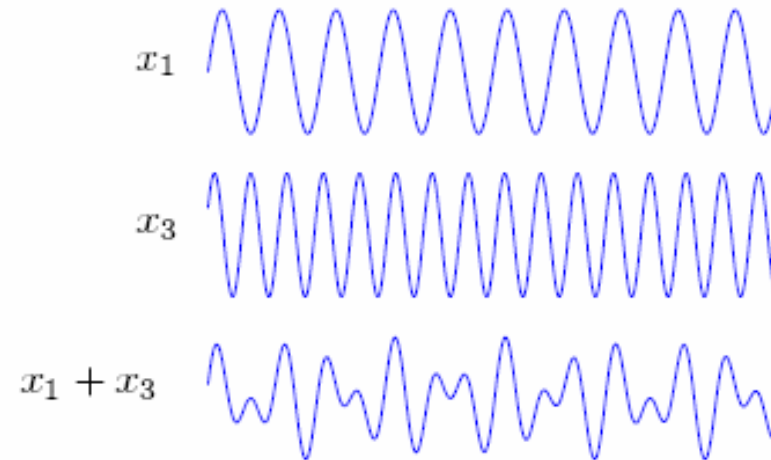


Spécifications en phase

Phase linéaire

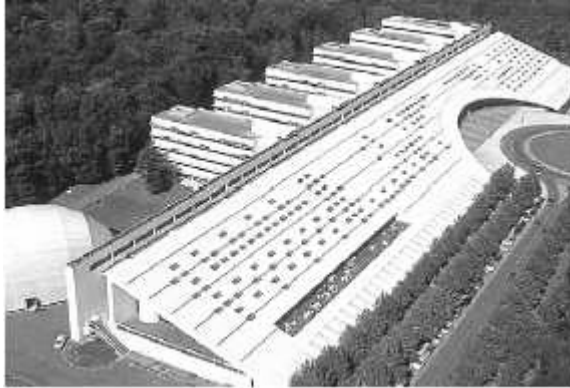


(a) Entrée



(b) Sortie

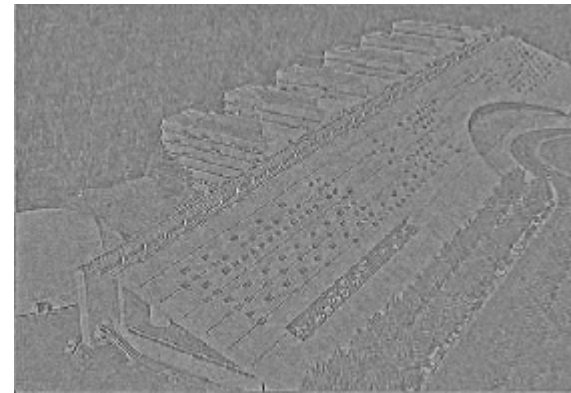
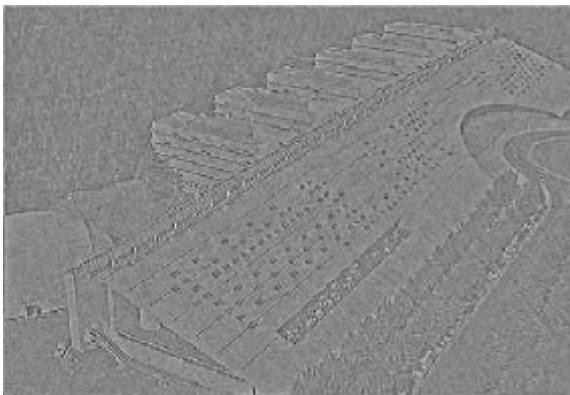
...(suite)



$$X(f_h, f_v) = |X(f_h, f_v)| e^{j(2\pi\phi_h(f_h) + 2\pi\phi_v(f_v))}$$

$$X_\phi(f_h, f_v) = e^{j(2\pi\phi_h(f_h) + 2\pi\phi_v(f_v))}$$

$$X_{\text{mod}}(f_h, f_v) = |X(f_h, f_v)|$$





Filtres numériques

Filtres RIF (FIR)

Phase linéaire

$$x(t) = A_1 \sin(2\pi f_1 t) + A_2 \sin(2\pi f_2 t)$$



$$y(t) = k_1 A_1 \sin(2\pi f_1 t + \varphi_1) + k_2 A_2 \sin(2\pi f_2 t + \varphi_2)$$

$$y(t) = kx(t - \tau)$$

$$y(t) = k_1 A_1 \sin 2\pi f_1 \left(t + \frac{\varphi_1}{2\pi f_1} \right) + k_2 A_2 \sin 2\pi f_2 \left(t + \frac{\varphi_2}{2\pi f_2} \right)$$

$$\frac{\varphi_1}{2\pi f_1} = \frac{\varphi_2}{2\pi f_2} \quad \text{donc} \quad \boxed{\varphi(f) = -2\pi f\tau + \varphi_0}$$



...(suite)

Temps de groupe

$$T(f) = -\frac{1}{2\pi} \frac{d(-2\pi f\tau + \varphi_0)}{df} = \tau$$

Condition

- $h(t)$ doit être symétrique ou antisymétrique
 - *Non réalisable avec les filtres analogiques*

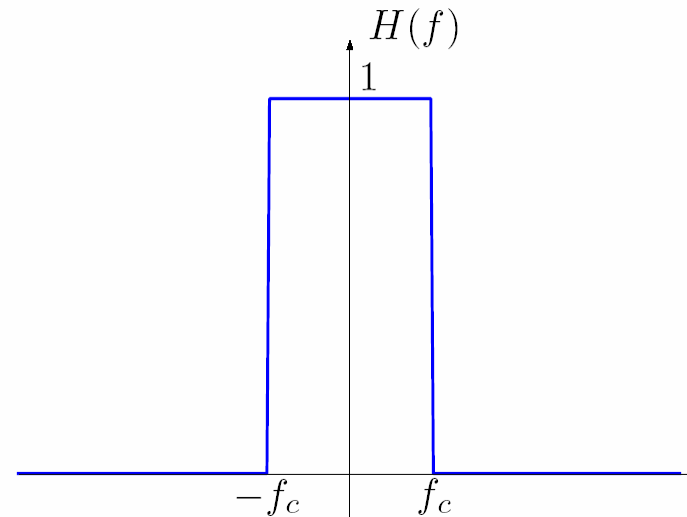


Synthèse

méthode de la fenêtre

- Spécification en fréquence sous la contrainte d'une réponse impulsionnelle symétrique ou anti-symétrique.
- Filtre idéal :

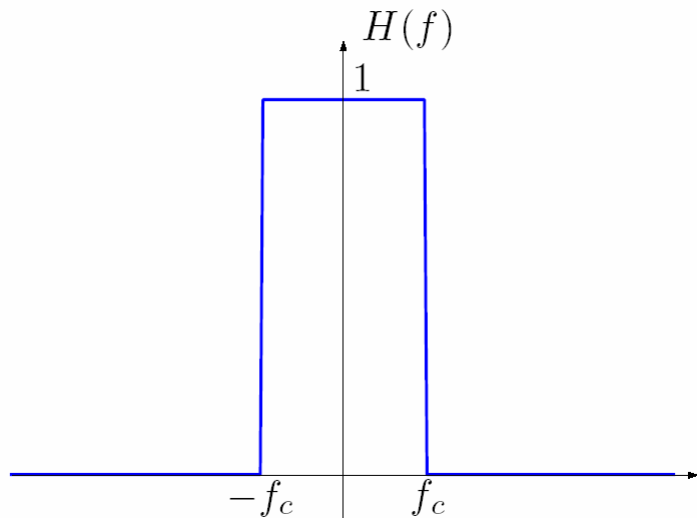
$$H(f) = \begin{cases} 1 & \text{si } |f| \leq f_c, \\ 0 & \text{sinon} \end{cases}$$



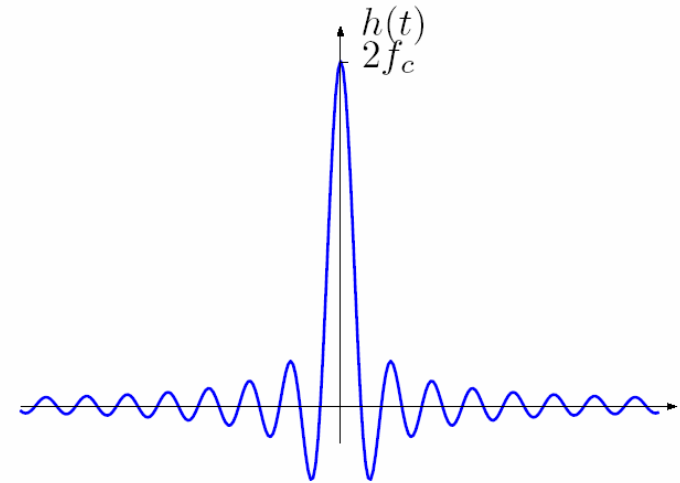


Aparté 1

(Fonctions rectangle et sinus cardinal)



TF^{-1} →

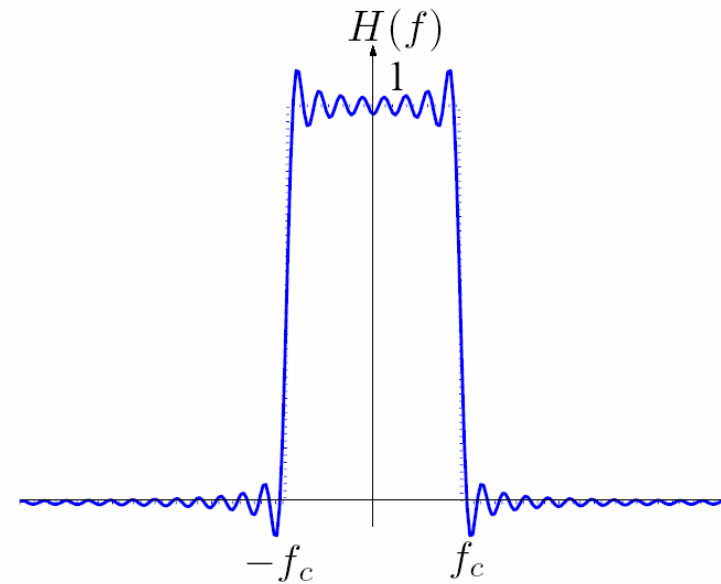
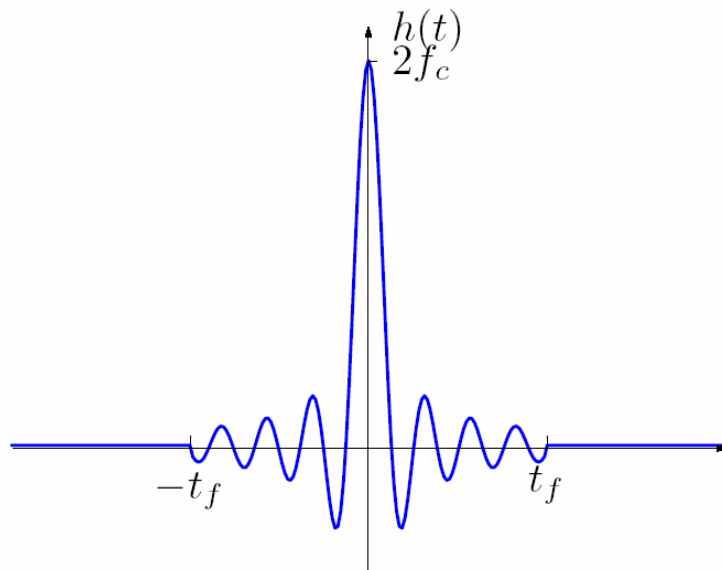


$$\text{rect}(f/2f_c) = \begin{cases} 1 & \text{if } |f| < f_c \\ 0 & \text{else.} \end{cases}$$

$$2f_c \text{sinc}(2f_c t) = 2f_c \frac{\sin(\pi 2f_c t)}{\pi 2f_c t}$$

Réponse impulsionnelle finie

$$h(t) = \begin{cases} 2f_c \operatorname{sinc}(2f_c t) & \text{si } |t| \leq t_f, \\ 0 & \text{sinon} \end{cases}$$



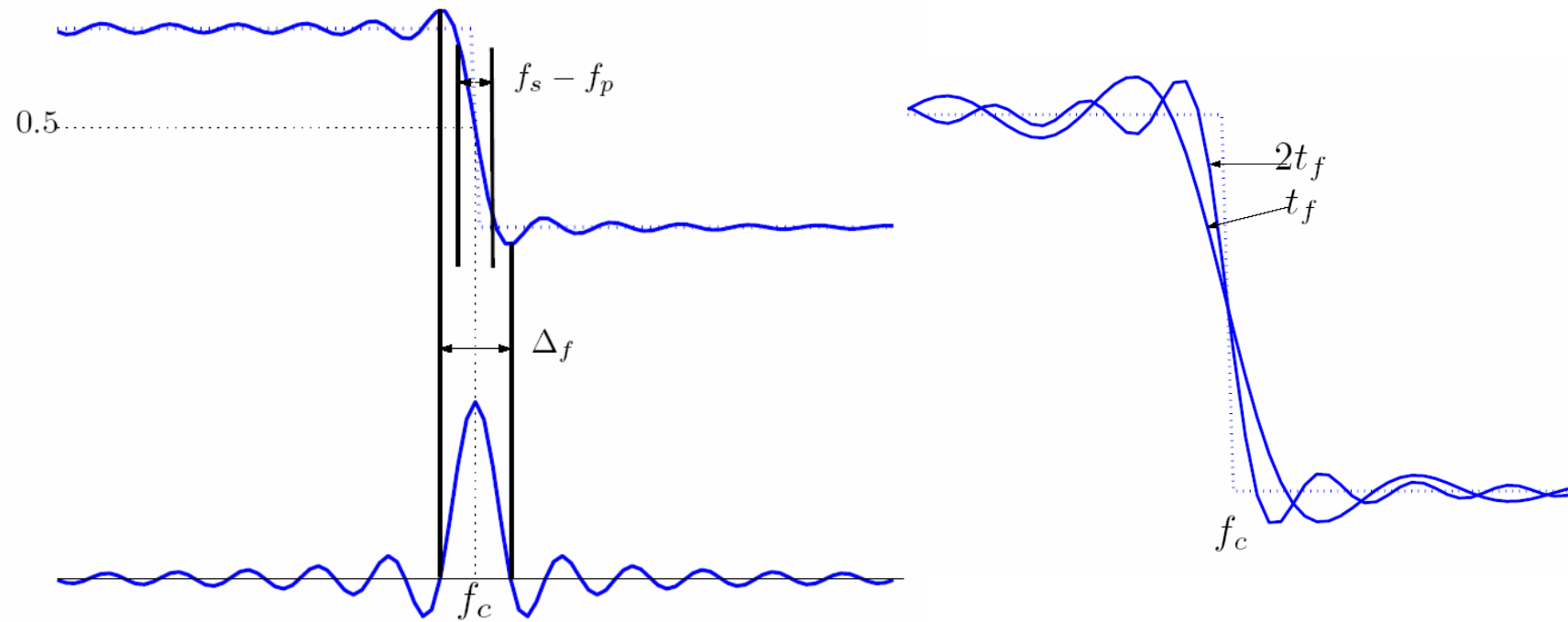


Aparté 2

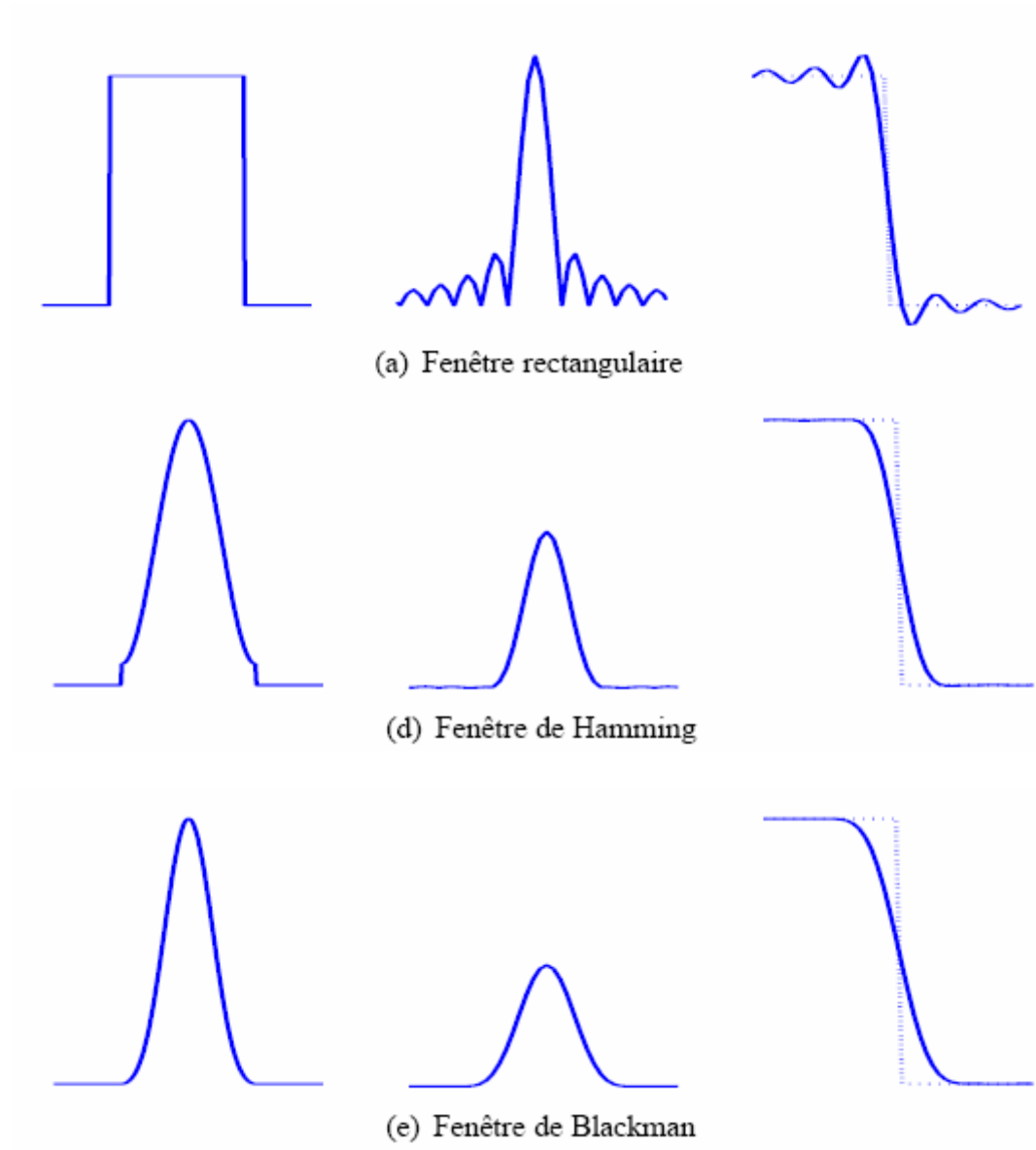
(théorème du fenêtrage)

$$x(t) \cdot y(t) \begin{array}{c} \xrightarrow{TF} \\ \xleftarrow{TF^{-1}} \end{array} \int_{-\infty}^{+\infty} X(\nu) Y(f - \nu) d\nu$$

Longueur de la fenêtre



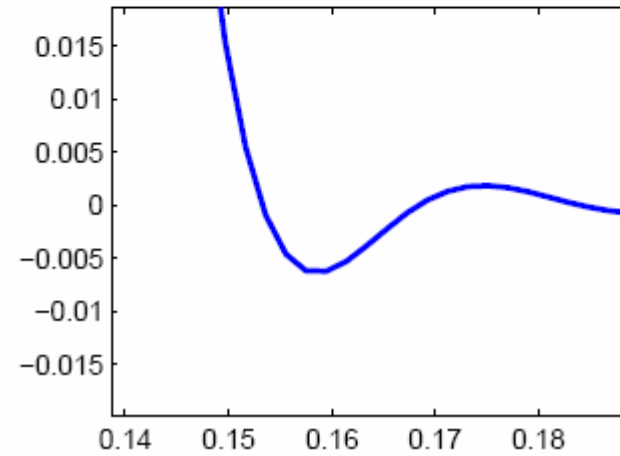
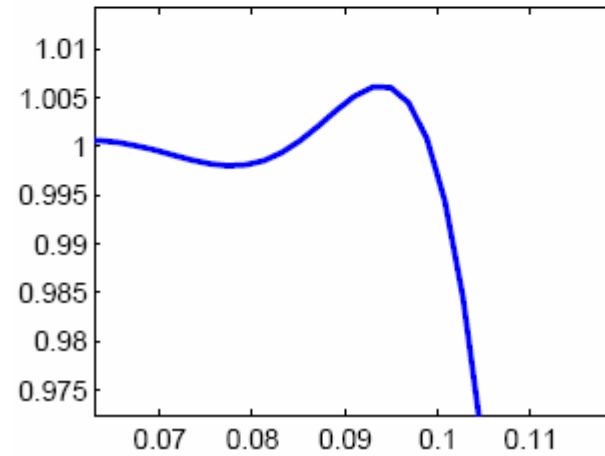
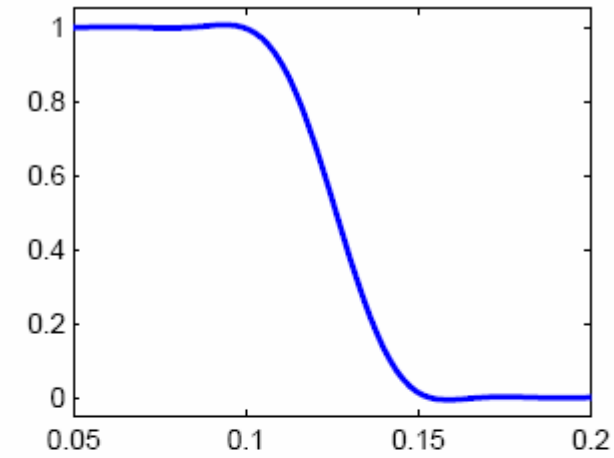
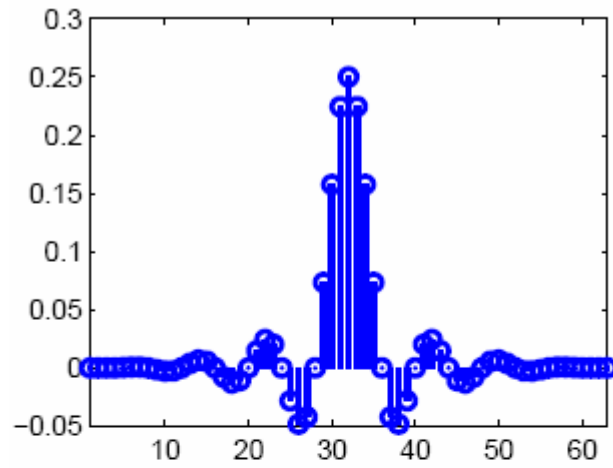
Fenêtres



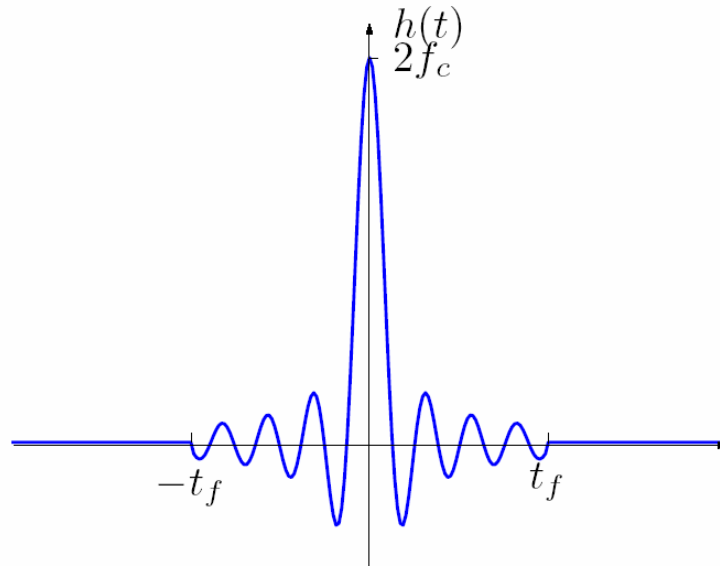
Fenêtres

Fenêtre	largeur du lobe principal	amplitude lobe secondaire	largeur bande de transition ($f_s - f_p$)	atténuation (A)
<u>rectangulaire</u>	$\frac{2}{N}$	$-13.3dB$	$\frac{0.92}{N}$	$20.9dB$
Bartlett	$\frac{4}{N}$	$-26.5dB$	$\frac{4}{N}$	$26dB$
Hanning	$\frac{4}{N}$	$-31.5dB$	$\frac{3.11}{N}$	$43.9dB$
<u>Hamming</u>	$\frac{4}{N}$	$-42.7dB$	$\frac{3.37}{N}$	$54.5dB$
<u>Blackman</u>	$\frac{6}{N}$	$-58.1dB$	$\frac{5.56}{N}$	$75.3dB$

Filtre obtenu



Phase



Systeme causal

$$h(t) * \delta(t - t_f) \xrightarrow{TF} H(f) \exp(-j2\pi f t_f)$$

$$T(f) = t_f$$



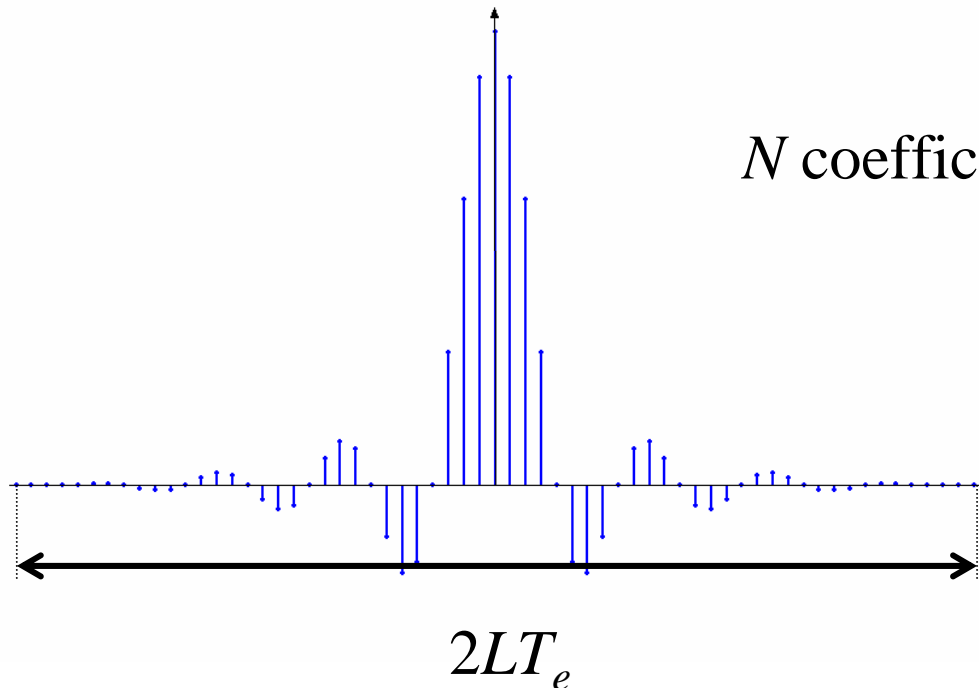
Phase (cas discret)

Soit $h[n] = h(nT_e)$ et $LT_e = t_f$, alors,

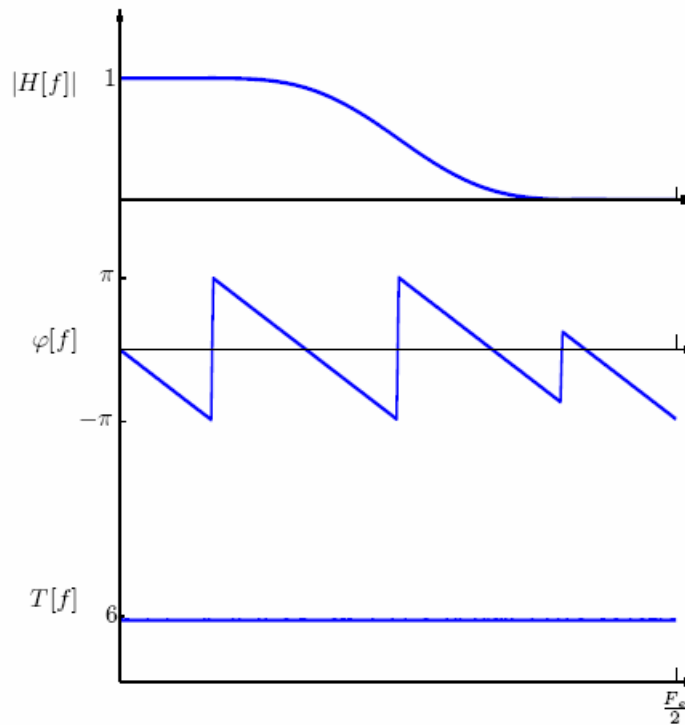
Causalité :

$$h[n] * \delta[n - L] \xrightarrow{TFD} H[k] \exp\left(-j2\pi \frac{k}{F_e} L\right)$$

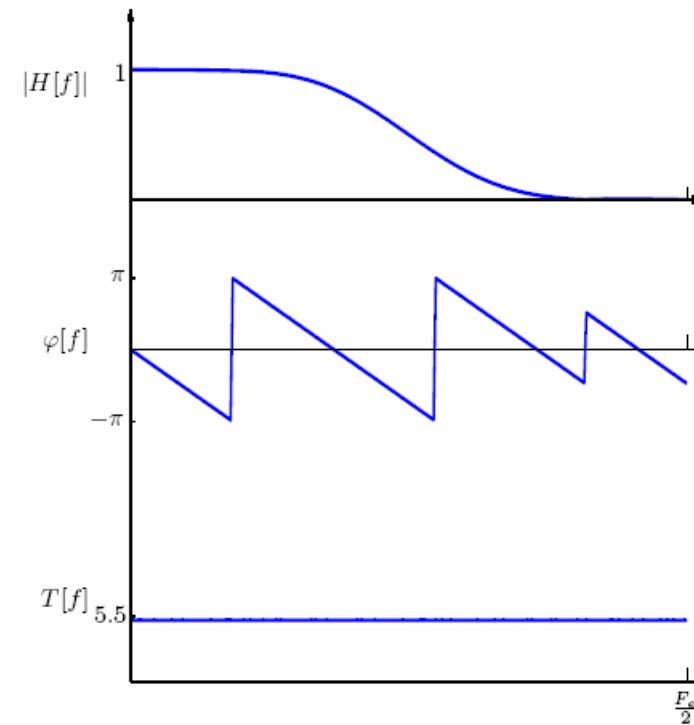
N coefficients : $L = \frac{N-1}{2}$



...(suite)

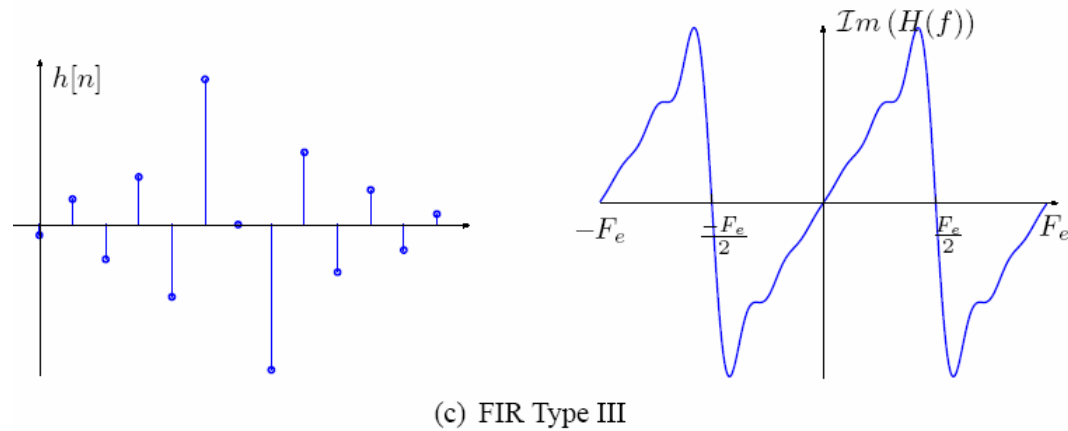
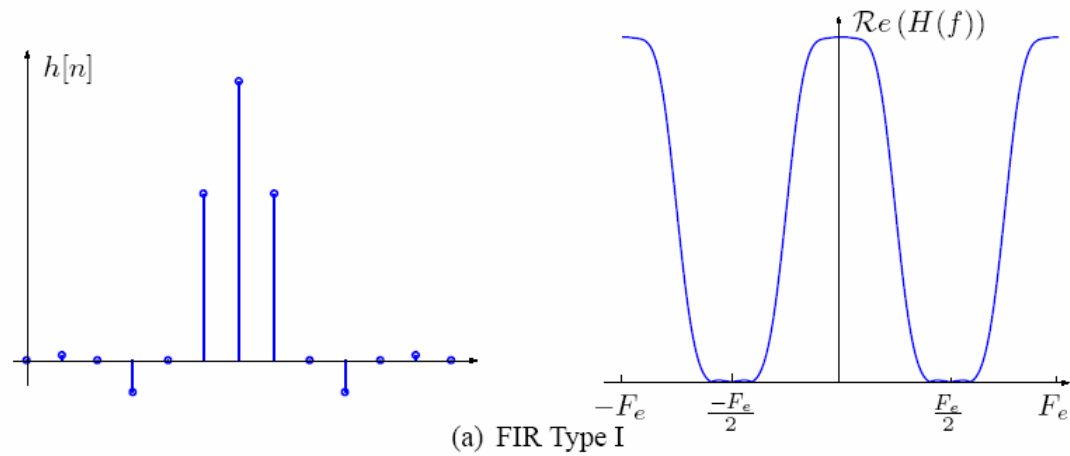


(a) Filtre ayant un nombre impair de coefficients (13)



(b) Filtre ayant un nombre pair de coefficients (12)

Types de filtre (exemple)





Réponse impulsionnelle, fonction de transfert, stabilité et équation aux différences

Réponse impulsionnelle : $h[n]$

Fonction de transfert :

$$H(z) = \sum_{k=-\infty}^{+\infty} h[k] z^{-k} = \sum_{k=0}^{N-1} h[k] z^{-k} = \frac{N(z)}{D(z)} = \frac{N(z)}{1}$$

Equation aux différences :

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{N-1} h[k] z^{-k}$$

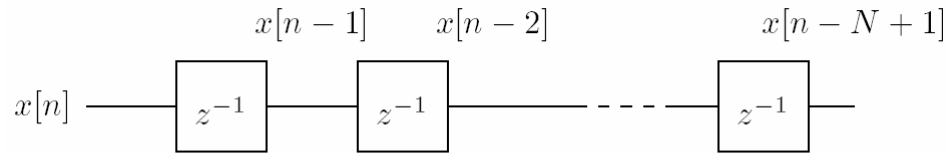
$$Y(z) = \sum_{k=0}^{N-1} h[k] X(z) z^{-k} = \sum_{k=0}^{N-1} b_k X(z) z^{-k}$$

$$y[n] = \sum_{k=0}^{N-1} b_k x[n-k]$$

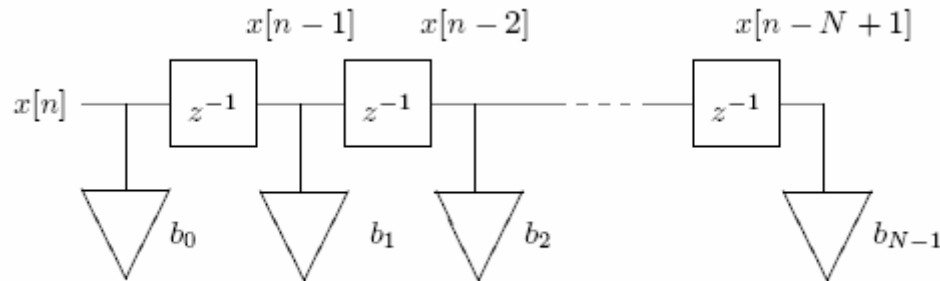


Implantation

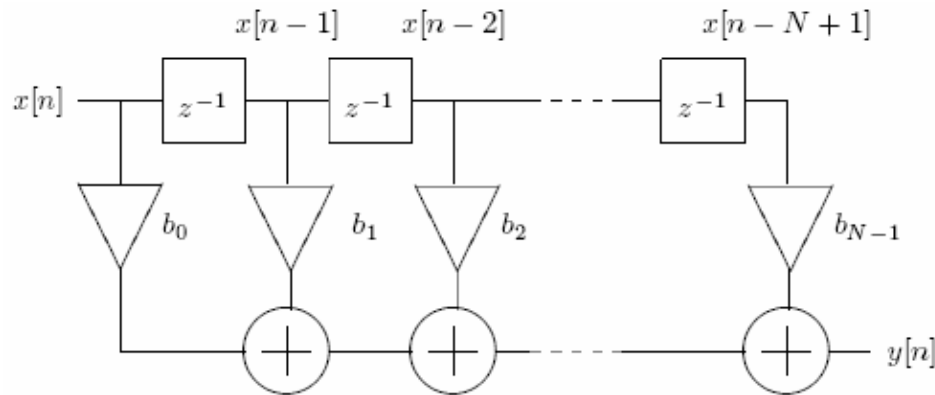
$$y[n] = \sum_{k=0}^{N-1} b_k x[n-k]$$



(a) Chaîne de retards



(c) Produit terme à terme



(e) Somme des produit terme à terme

$x[n]$
 $x[n-1]$
 \dots
 $x[n-N+1]$

(b) Données fournies par le traitement

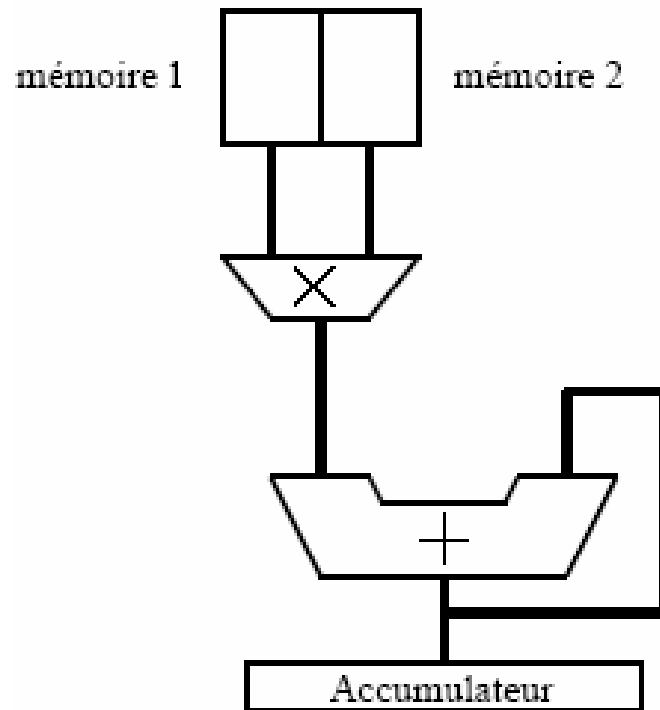
$x[n] \cdot b_0$
 $x[n-1] \cdot b_1$
 \dots
 $x[n-N+1] \cdot b_{N-1}$

(d) Données fournies par le traitement

$$y[n] = \sum_{k=0}^{N-1} b[k]x[n-k]$$

(f) Données fournies par le traitement

...(suite)



$$y[n] = \sum_{k=0}^{N-1} b_k x[n-k]$$

FIG. IV.15 – Multiplieur-accumulateur



Algorithme

1. Le nouvel échantillon $x[]$ est rangé au début du tableau de donnée et l'accumulateur est mis à 0.
2. Boucle de traitement sur le nombre de coefficients (N), compteur de coefficients est mis à 0.
 - (a) Multiplication de la donnée et du coefficient courants et accumulation au résultat précédent dans l'accumulateur.
 - (b) Si encore des coefficients à traiter, incrémentation du compteur de coefficients et retour à l'étape 2a ; sinon aller à l'étape 3.
3. Le contenu de l'accumulateur est transféré vers la case mémoire devant contenir le résultat du filtrage (y par exemple).
4. Le tableau de données est préparé pour le prochain échantillon $x[]$. C'est l'étape de vieillissement des données (figure IV.16) : on recopie le contenu de chaque case mémoire dans la case mémoire immédiatement supérieure en partant de l'adresse $N - 1$, l'adresse 1 se trouve ainsi disponible pour recevoir le prochain échantillon en entrée du filtre.
5. on retourne à l'étape 1 en attente du prochain échantillon $x[]$.

...(suite)

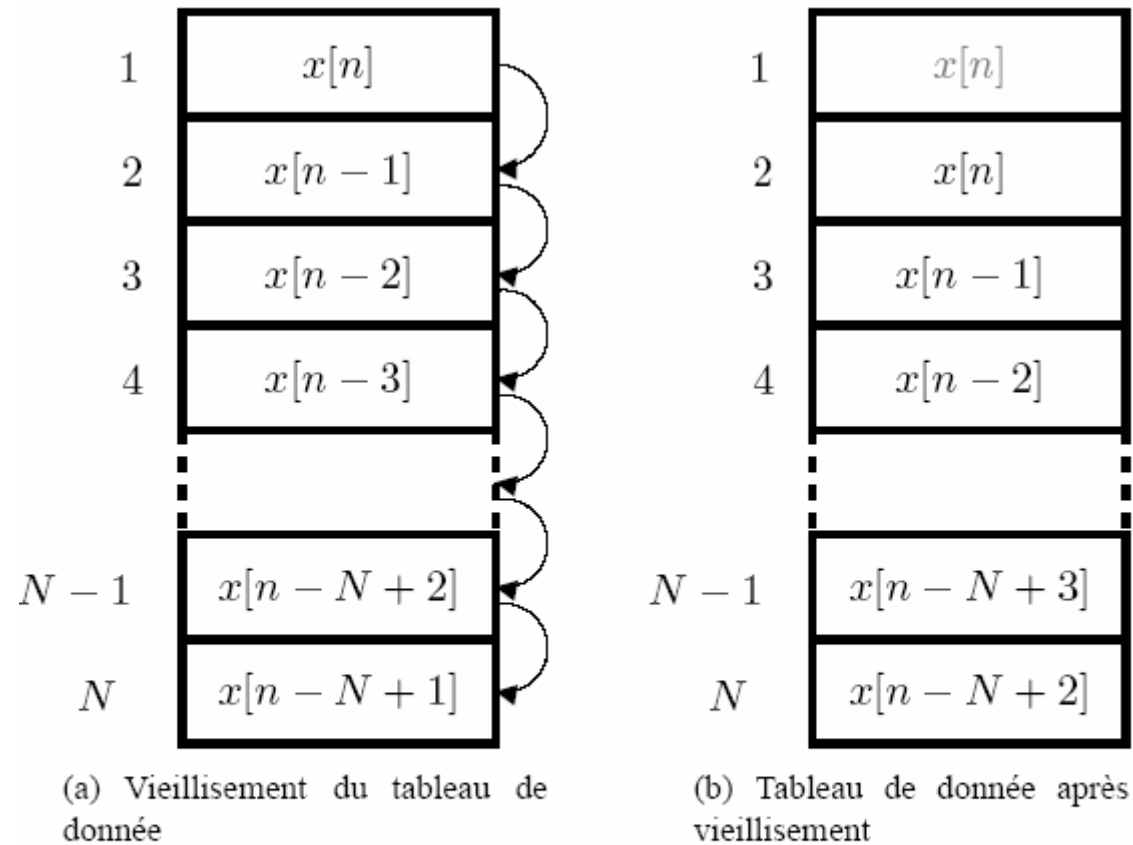


FIG. IV.16 – Gestion du tableau de données



Filtres numériques

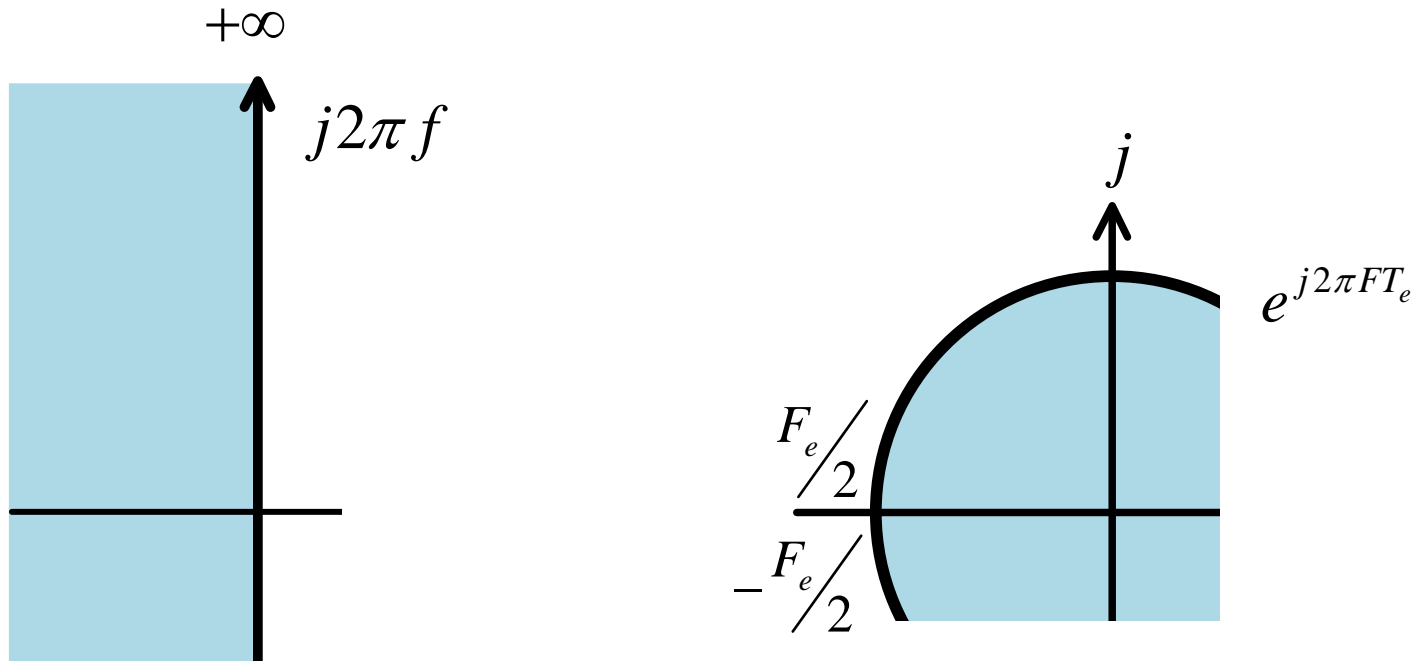
Filtres RII (IIR)



Étapes de conception

- Prototype analogique
 - Fonctions d'approximation
 - Transposition numérique
 - Transposition $p \rightarrow z$
 - Structure de réalisation
-
- Methode numérique directe : Yule Walker
 - Méthode par optimisation

Transformation $p \rightarrow z$



$-\infty$

f : fréquence analogique
 F : fréquence numérique

...(suite)

- Transformation bilinéaire :
 - Réalise une relation bijective entre les fréquences analogiques et les fréquences numériques.
 - Repose sur l'approximation de l'intégrale continue par la méthode des trapèzes.

Transformée de Laplace

$$y(t) = \int_{-\infty}^t x(u) du$$

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

$$y_n = \sum_{k=-\infty}^n T_e \frac{(x_k + x_{k-1})}{2}$$

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

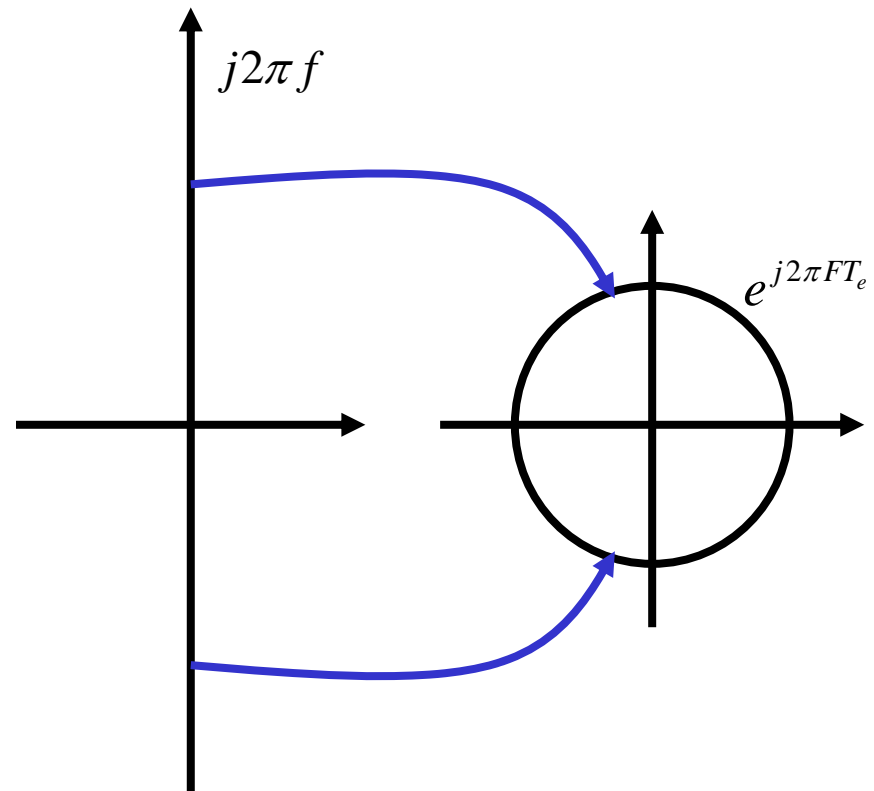
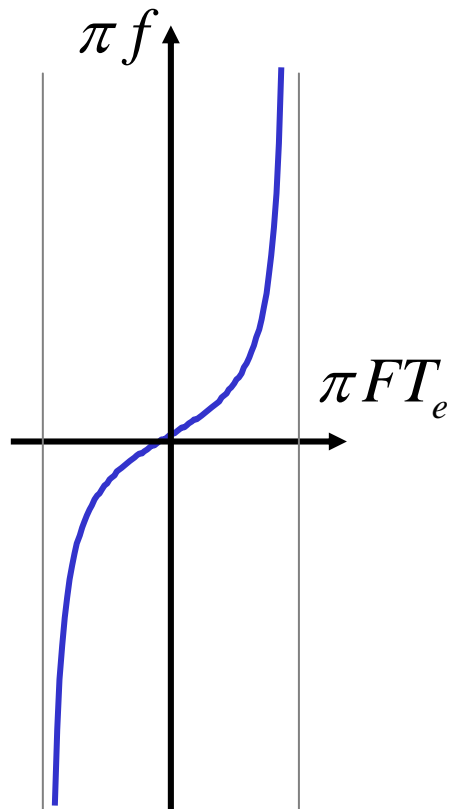
Transformée en z

$$p = \frac{2}{T_e} \frac{1 - z^{-1}}{1 + z^{-1}}$$

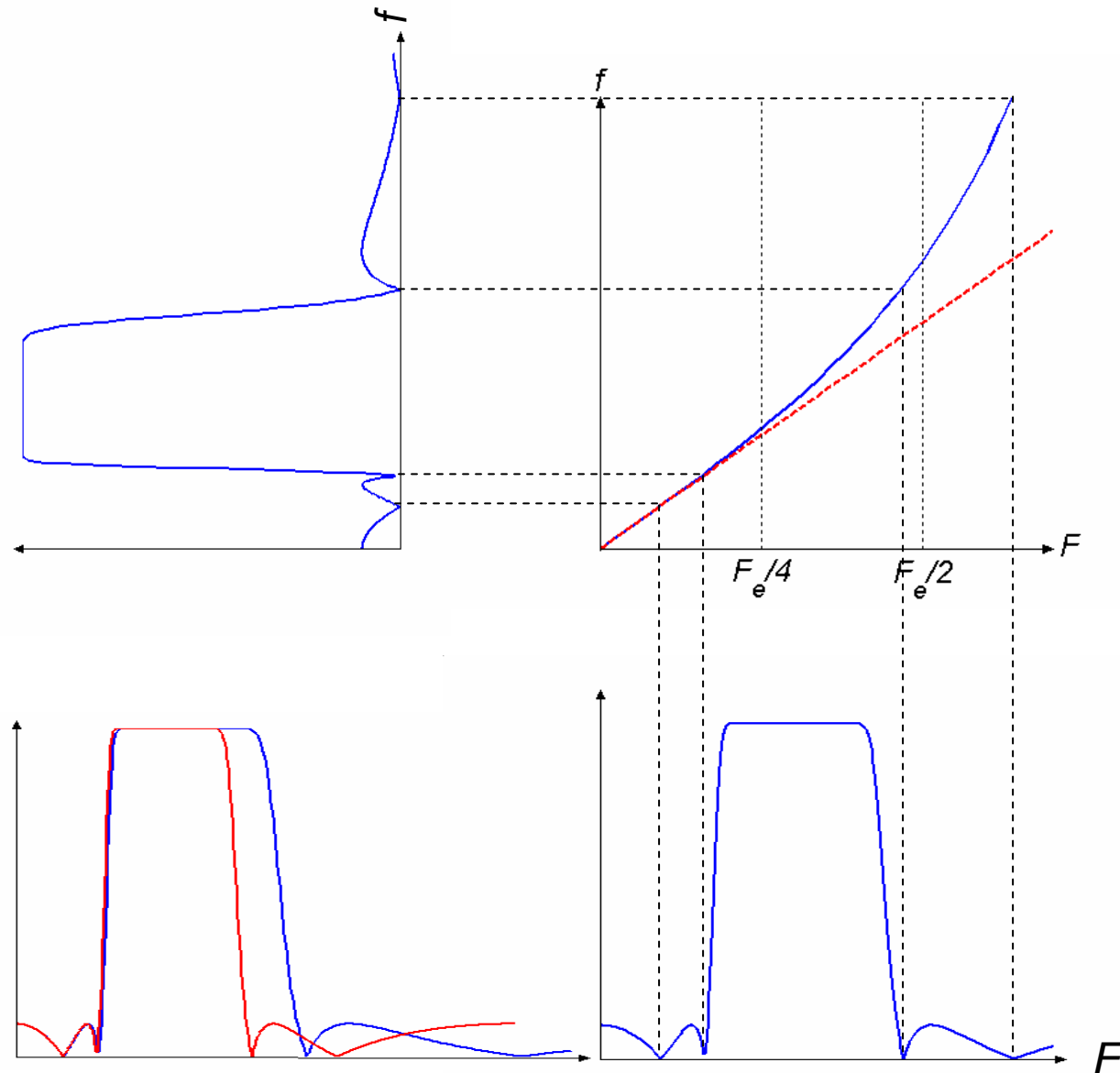
...(suite)

$$p \Big|_{p=j2\pi f} = \frac{2}{T_e} \frac{1-z^{-1}}{1+z^{-1}} \Big|_{\exp(j2\pi FT_e)}$$

$$\pi f = \frac{1}{T_e} \tan(\pi FT_e)$$



Prédistorsion





Synthèse

- Définition du gabarit numérique en fonction de l'application.
- Prédistorsion des fréquences caractéristiques du gabarit (f_p, f_s).
- Ce gabarit prédistorsion est utilisé pour calculer un filtre prototype analogique en utilisant les fonctions d'approximations :
 - Butterworth
 - Tchebycheff I
 - Tchebycheff II
 - Elliptic (Cauer)
 - ...
- Le filtre prototype analogique est transformé en filtre numérique avec la transformation bilinéaire qui supprime la prédistorsion introduite à l'étape précédente. Ce filtre respecte alors les contraintes du gabarit numérique.



Filtres numériques

Équations aux différences

$$y[n] = - \sum_{i=1}^p a_i y[n-i] + \sum_{i=0}^q b_i x[n-i]$$

Fonctions de transfert rationnelle

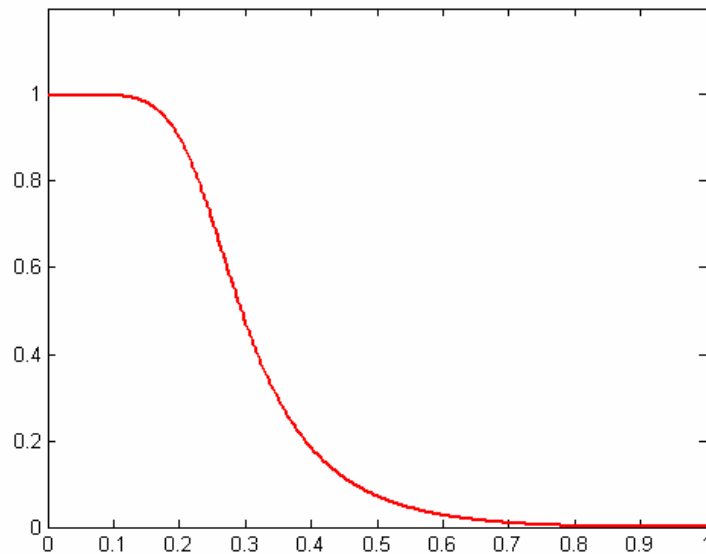
$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^q b_i z^{-i}}{1 + \sum_{i=1}^p a_i z^{-i}}$$

Caractéristiques des filtres RII

- Filtres de Butterworth

$$|H(f)|^2 = \frac{1}{1 + \left(\frac{f}{f_c}\right)^{2N}}$$

- Défini par son ordre N et sa fréquence de coupure f_c .
- Fonction de transfert en module monotone.



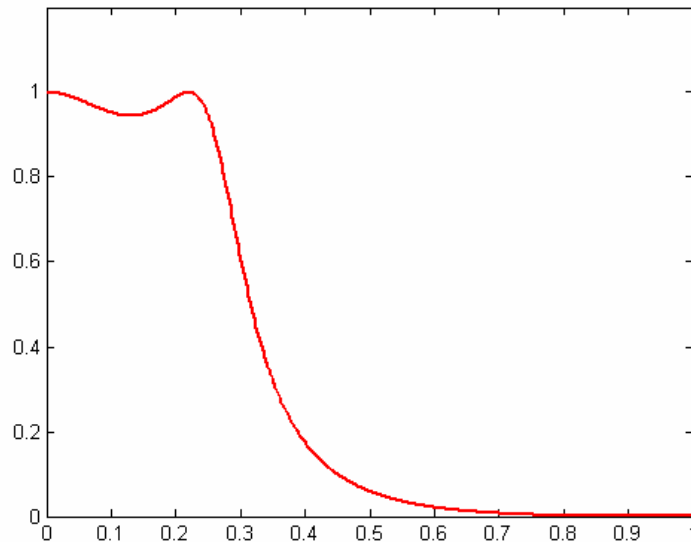
```
% Butterworth filter
% sampling frequency : 16 kHz
% Fp : 1800 Hz
% Fa : 4000 Hz
% Rp : 0.01 dB % Ra : 50 dB
[N, Wn]= buttord(1800/8000,4000/8000,0.01,50);
[B,A] = butter(N,Wn);
H=freqz(B,A);
plot(linspace(0,1,512),abs(H))
```

...(suite)

• Filtres de Tchebycheff I

$$|H(f)|^2 = \frac{1}{1 + \varepsilon^2 T_N^2 \left(\frac{f}{f_p} \right)}$$

- Défini par son ordre N et la fréquence délimitant la bande passante f_p et ε l'ondulation en bande passante.
- Ondulation en bande passante et monotone en bande atténuée.



```
% Chebyshev I filter
% sampling frequency : 16 kHz
% Fp : 1800 Hz
% Fa : 4000 Hz
% Rp : 0.01 dB
% Ra : 50 dB
[N, Wn] =
cheb1ord(1800/8000,4000/8000,0.01,50); [B,A] =
cheby1(N,0.01,Wn);
H=freqz(B,A);
plot(linspace(0,1,512),abs(H))
```

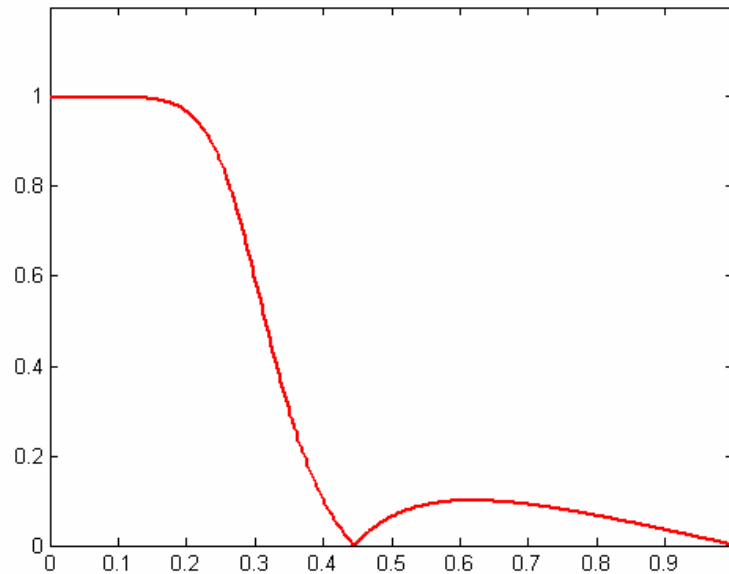
$T_N(\)$ est un polynome de Tchebycheff d'ordre N

...(suite)

• Filtres de Tchebycheff II

$$|H(f)|^2 = 1 - \frac{1}{1 + \varepsilon^2 T_N^2\left(\frac{f_s}{f}\right)}$$

- Défini par son ordre N , la fréquence délimitant la bande atténuée f_s et l'atténuation ε .
- Monotone en bande passante et ondulation en bande atténuée.



$T_N(\)$ est un polynome de Tchebycheff d'ordre N

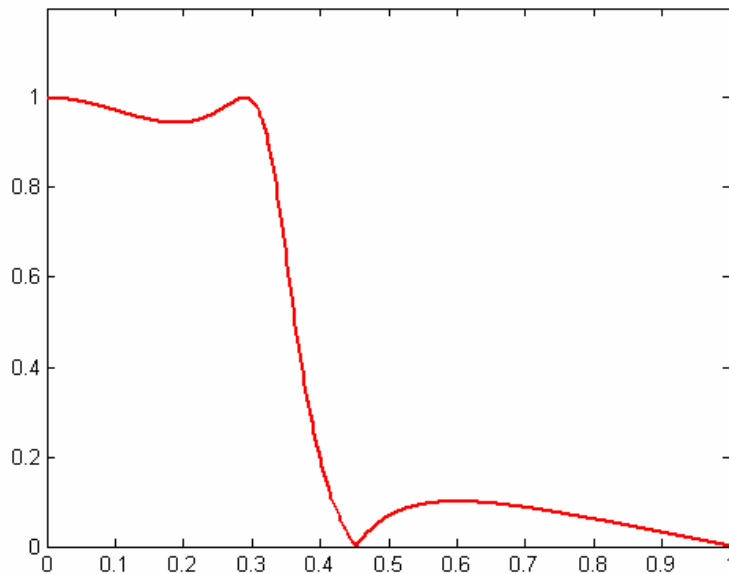
```
% Chebyshev II filter
% sampling frequency : 16 kHz
% Fp : 1800 Hz
% Fa : 4000 Hz
% Rp : 0.01 dB
% Ra : 50 dB
[N, Wn] =
cheb2ord(1800/8000,4000/8000,0.01,50); [B,A] =
cheby2(N,50,Wn);
H=freqz(B,A);
plot(linspace(0,1,512),abs(H))
```

...(suite)

- Filtres elliptique

$$|H(f)|^2 = \frac{1}{1 + \varepsilon^2 R_N^2 \left(\frac{f}{\sqrt{f_p f_s}} \right)}$$

- Défini par son ordre N et la fréquence délimitant la bande atténuée f_s , la fréquence délimitant la bande passante f_p et ε l'atténuation et l'ondulation en bande passante.
- Ondulation en bande passante et ondulation en bande atténuée.



$R_N(\cdot)$ est un polynôme rationnel de Tchebycheff d'ordre N

```
% Elliptic filter
% sampling frequency : 16 kHz
% Fp : 1800 Hz
% Fa : 4000 Hz
% Rp : 0.01 dB % Ra : 50 dB
[N, Wn] =
ellipord(1800/8000,4000/8000,0.01,50);
[B,A] = ellip(N,0.01,50,Wn);
H=freqz(B,A);
plot(linspace(0,1,512),abs(H))
```



Pôles et zéros

$$H(z) = \frac{N(z)}{D(z)} = \frac{\sum_{i=0}^q b_i z^{-i}}{1 + \sum_{i=1}^p a_i z^{-i}} = \frac{\prod_{i=1}^q (1 - \beta_i z^{-1})}{\prod_{i=1}^p (1 - \alpha_i z^{-1})}$$

Zéros de transmission

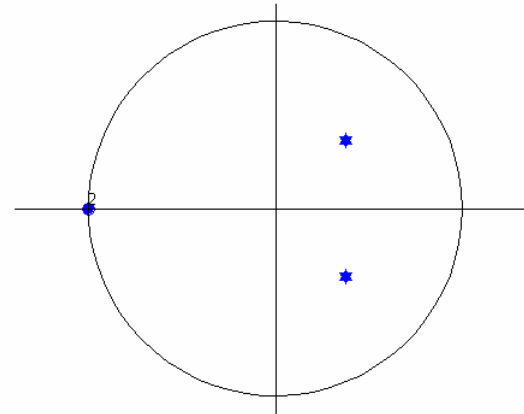
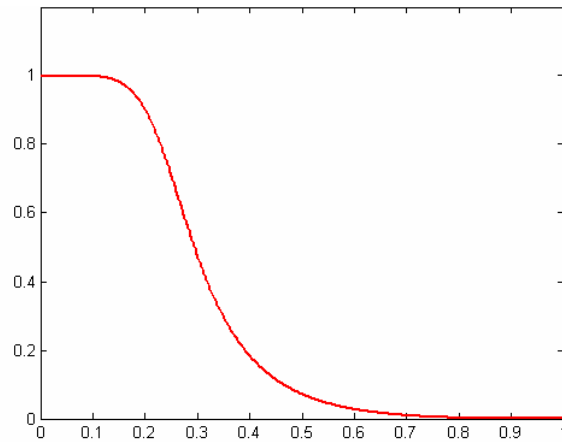
$$z = \beta_i \Rightarrow N(z) = 0 \Rightarrow H(z) = 0$$

Pôles

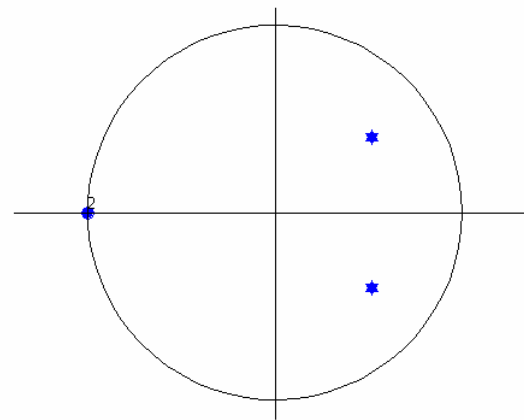
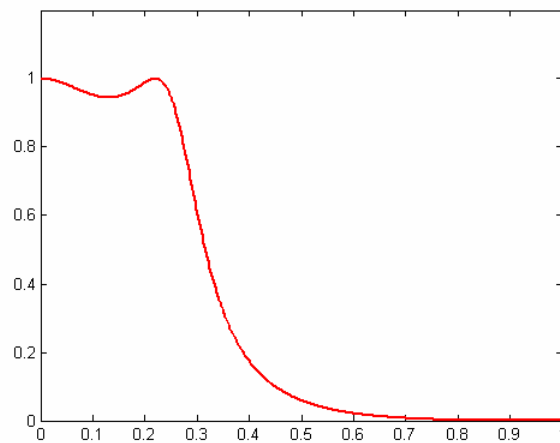
$$z = \alpha_i \Rightarrow D(z) = 0 \Rightarrow H(z) \rightarrow \infty$$

...(suite)

Butterworth

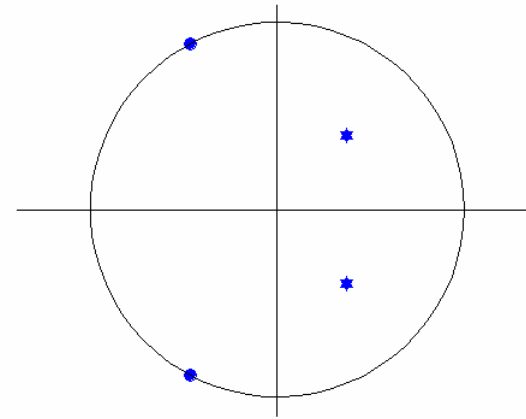
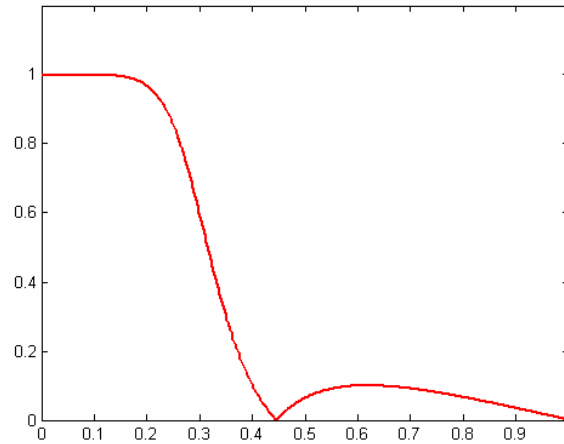


Tchebycheff I

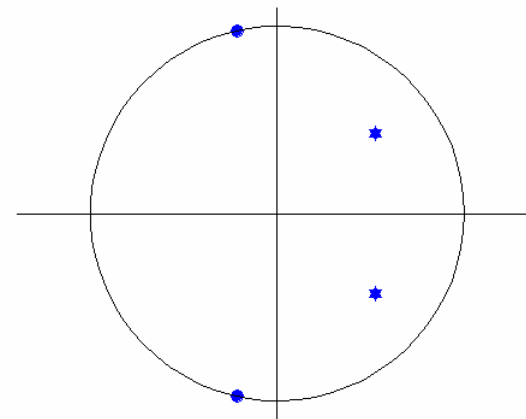
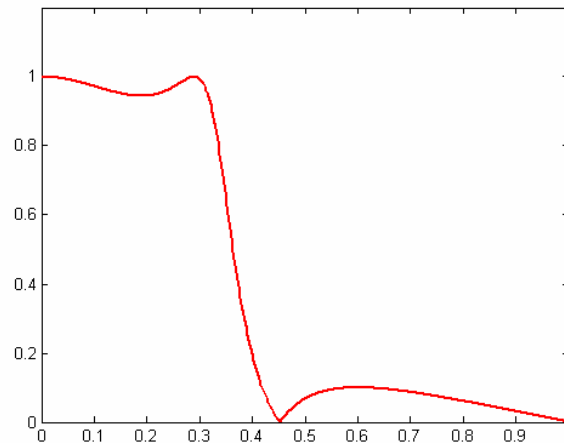


...(suite)

Tchebycheff II

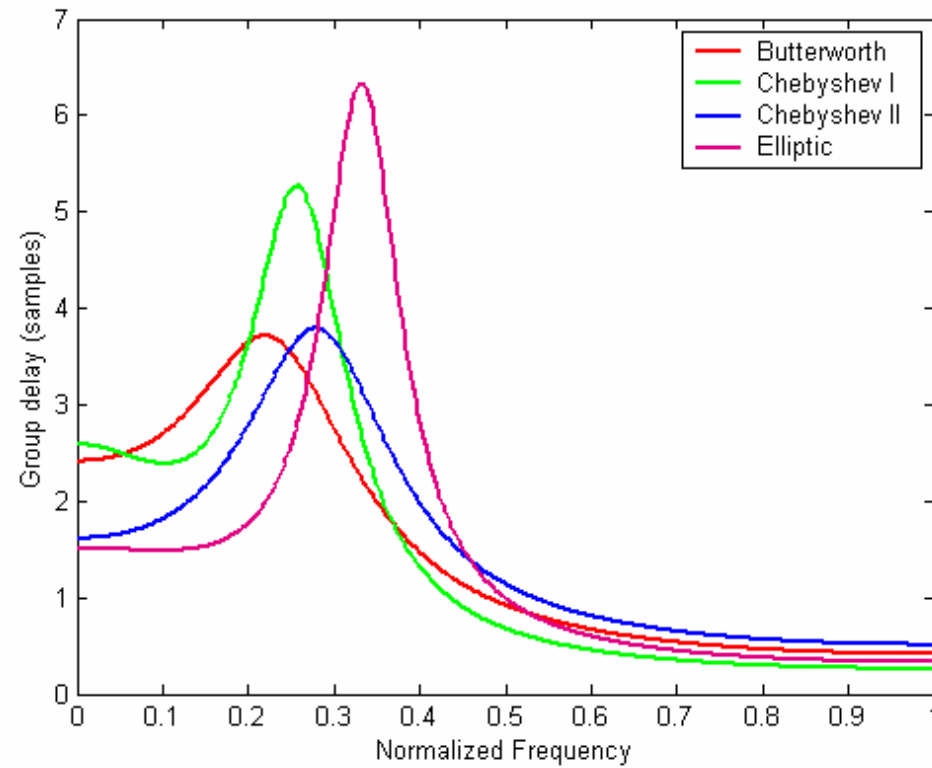


Elliptique



Temps de groupe

$$T(f) = -\frac{1}{2\pi} \frac{d\phi(f)}{df}$$

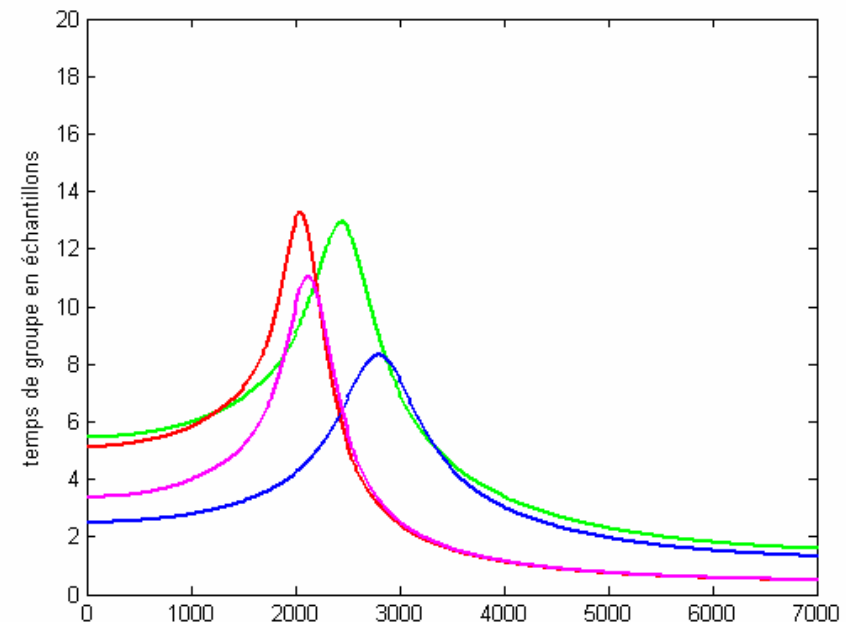
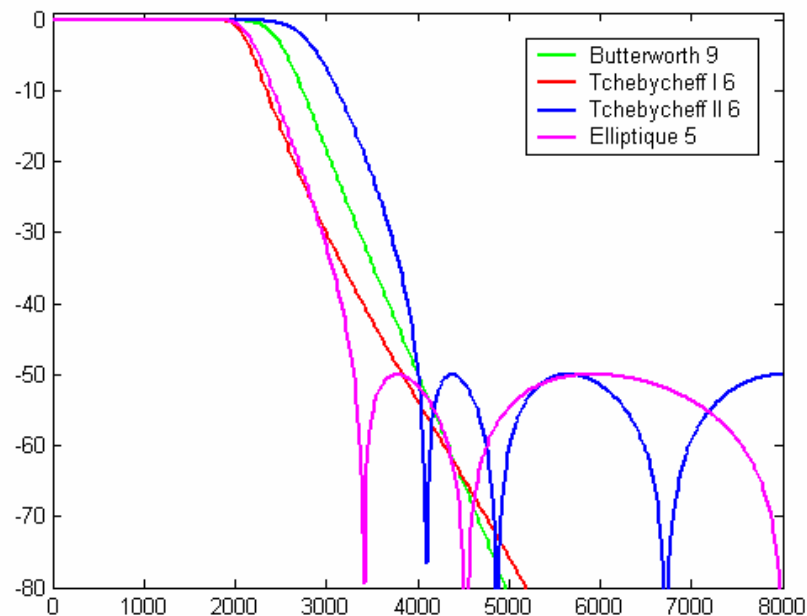


...(suite)

Fréquence d'échantillonnage : 16 kHz

F_p : 1800 Hz, F_s : 4000 Hz

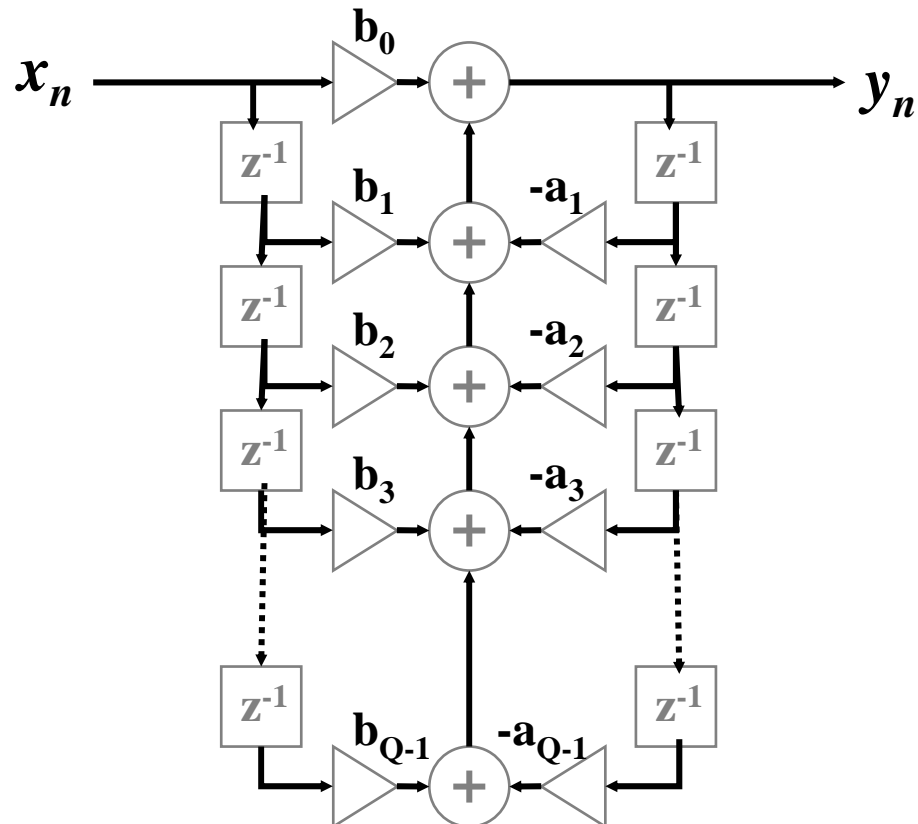
R_p : 0.01 dB, R_s : 50 dB



Structure d'implantation

$$y[n] = - \sum_{i=1}^p a_i y[n-i] + \sum_{i=0}^q b_i x[n-i]$$

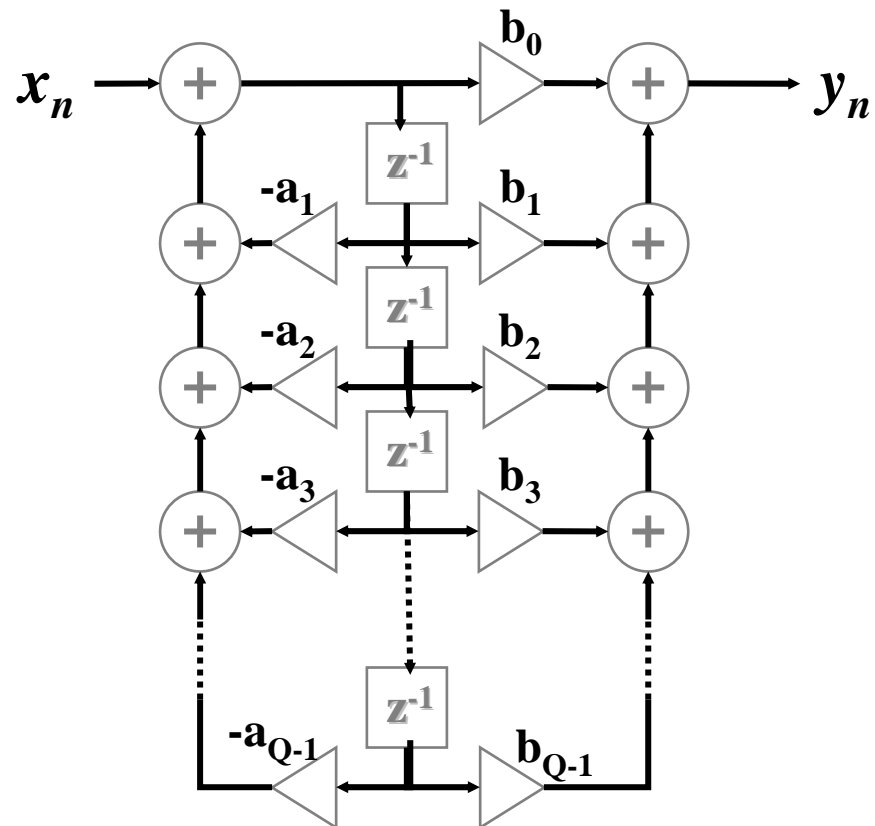
Forme directe I



$$H(z) = B(z) \frac{1}{A(z)}$$

...(suite)

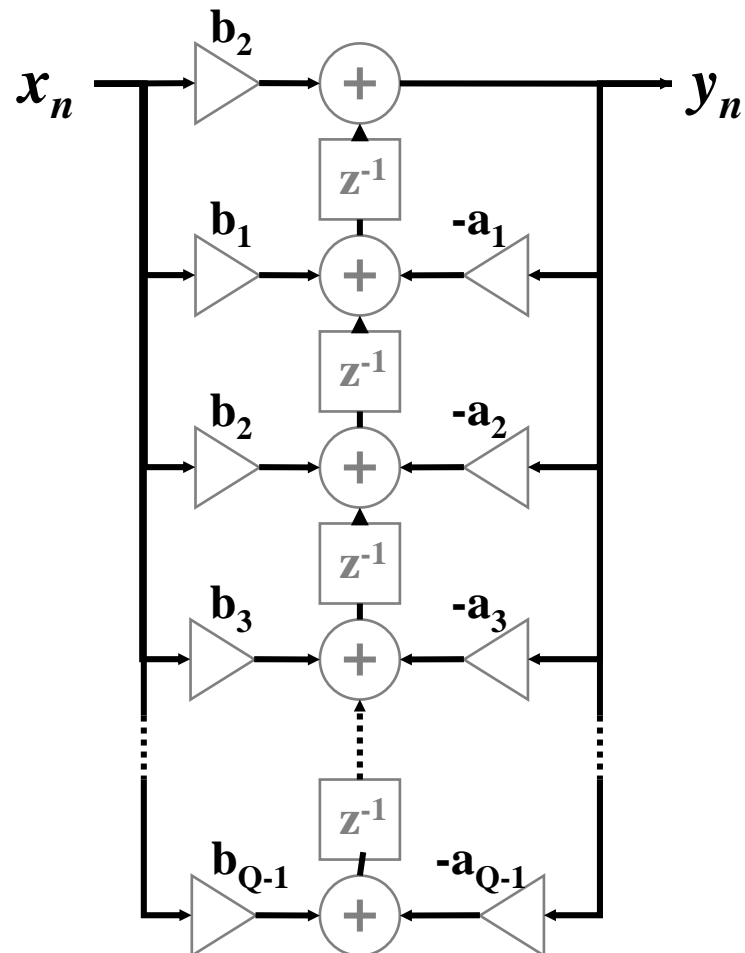
Forme directe II



$$H(z) = \frac{1}{A(z)} B(z)$$

...(suite)

Forme directe transposée II



$$H(z) = \frac{1}{A(z)} B(z)$$



Quantification des coefficients

- La précision finie des processeurs et des cases mémoires implique la quantification des coefficients.

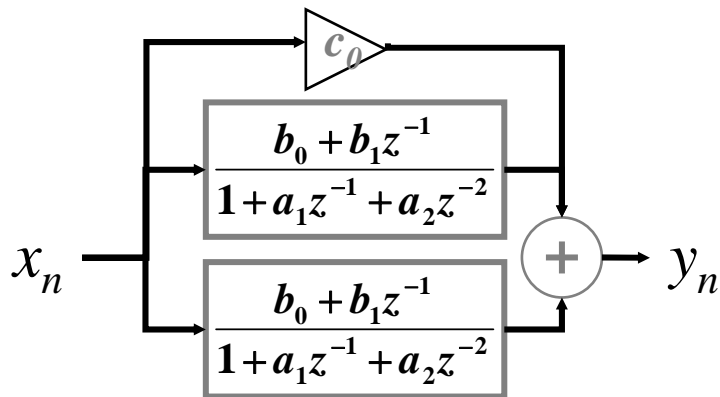
$$\bar{a}_k = a_k + \Delta a_k \quad \text{et} \quad \bar{b}_k = b_k + \Delta b_k$$

- La modification des coefficients modifie la fonction de transfert.
- Plus le filtre est d'ordre élevé, plus la perturbation introduite par la quantification est importante

...(suite)

- sensibilité des coefficients à la quantification → cellules d'ordre 2

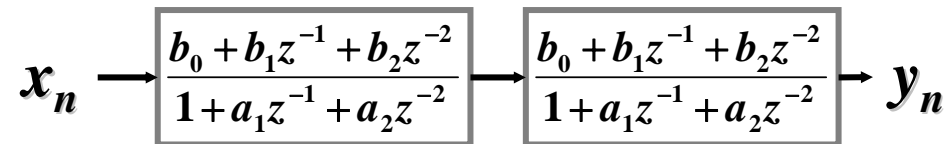
Structure parallèle



Décomposition en éléments simples

$$\frac{B(z)}{A(z)} = c_0 + \sum_{i=0}^{N-1} \frac{b_{i0} + b_{i1}z^{-1}}{1 + a_{i1}z^{-1} + a_{i2}z^{-2}}$$

Structure cascade

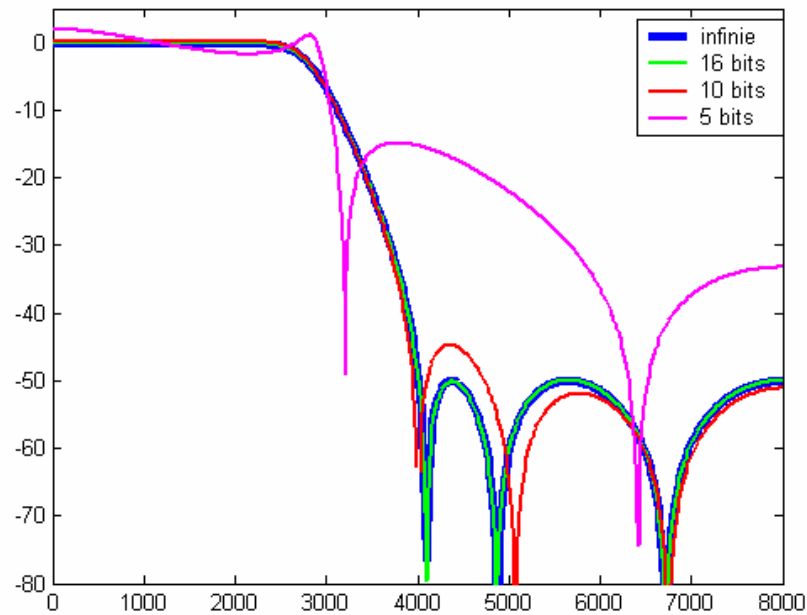


Factorisation spectrale

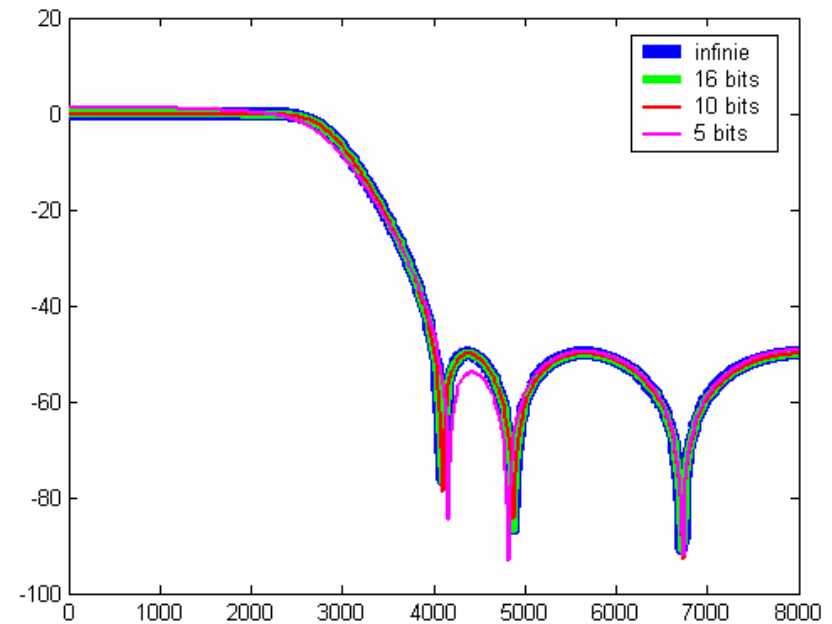
$$\frac{B(z)}{A(z)} = b_0 \prod_{i=0}^{N-1} \frac{1 + b_{i1}z^{-1} + b_{i2}z^{-2}}{1 + a_{i1}z^{-1} + a_{i2}z^{-2}}$$

Quantification

Structure directe

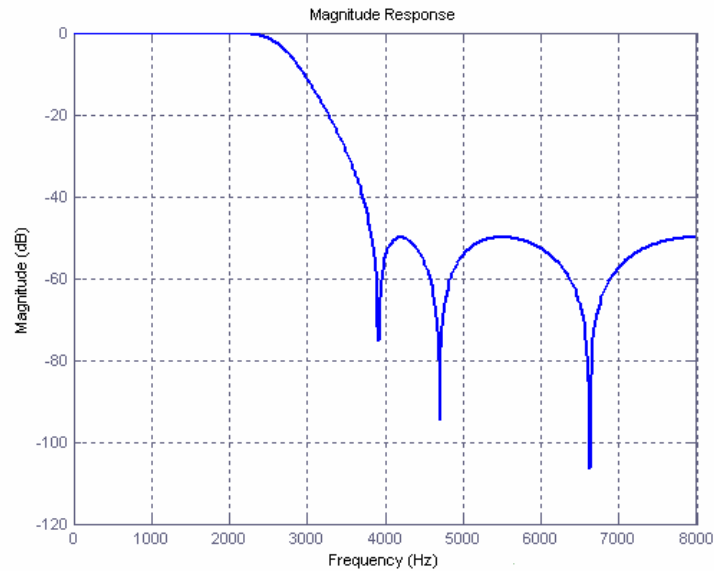


Structure cascade

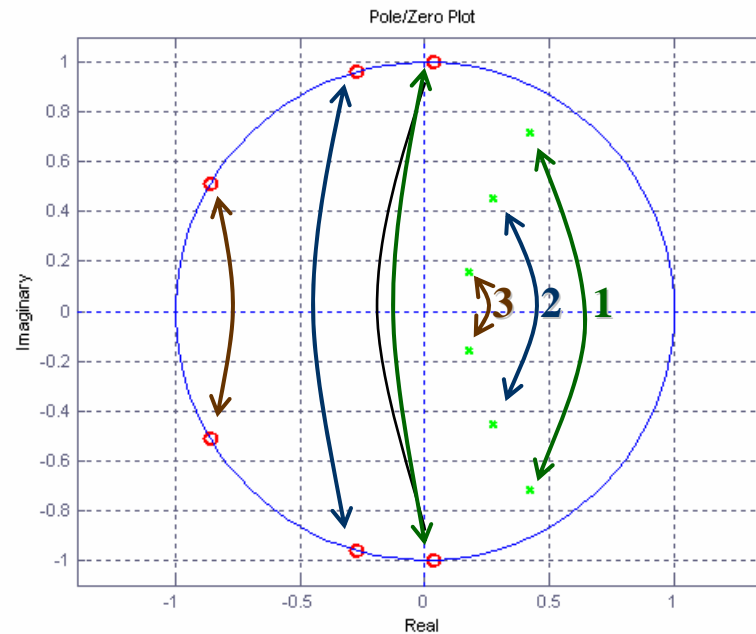




Factorisation spectrale



```
%calcul des racines
P=roots(A)
Z=roots(B)
% fabrication des polynomes d'ordre 2
% à partir des racines complexes conjuguées
A1=poly(P(1:2))
B1=poly(Z(3:4))
A2=poly(P(3:4))
B2=poly(Z(5:6))
A3=poly(P(5:6))
B3=poly(Z(1:2))
% Cela peut être fait automatiquement par la fonction matlab
% [sos,g]=tf2sos(B,A);
```



Calcul

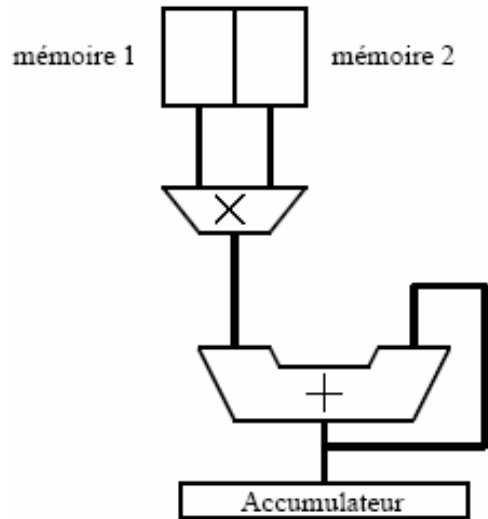
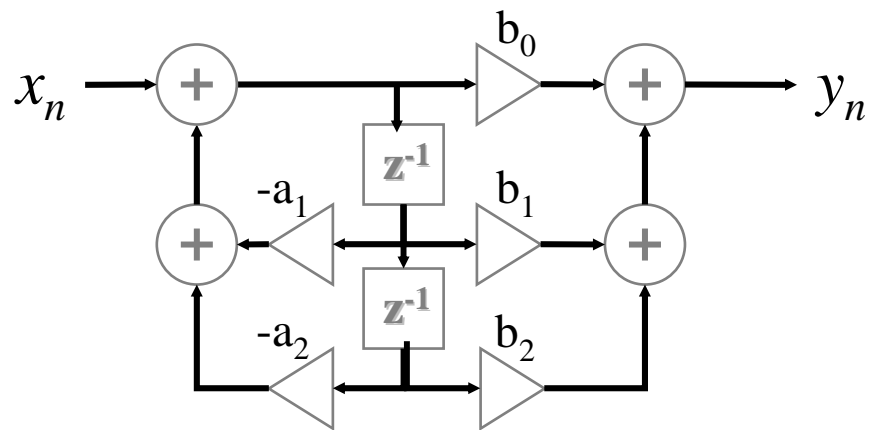


FIG. IV.15 – Multiplieur-accumulateur



$$ACC=x(n)$$

$$ACC=ACC - a_1 \times w(n-1)$$

$$ACC=ACC - a_2 \times w(n-2)$$

$$w(n)=ACC$$

$$ACC=w(n) \times b_0$$

$$ACC=ACC + b_2 \times w(n-2)$$

$$ACC=ACC + b_1 \times w(n-1)$$

$$ACC$$

$$y(n)=ACC$$

$$w(n-2)=w(n-1)$$

$$w(n-1)=w(n)$$