Conception de filtres numériques

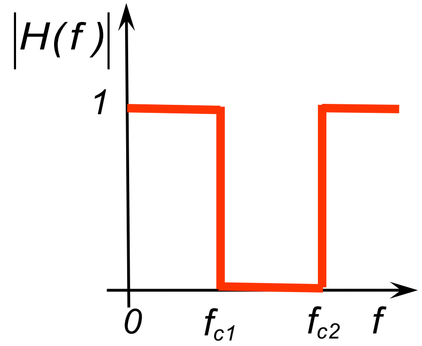
Hugues GARNIER

hugues.garnier@univ-lorraine.fr

Objectifs de la conception de filtres numériques

On veut réaliser un filtrage donné

défini à partir de sa réponse fréquentielle



Déterminer l'équation aux différences qui permet de réaliser le filtrage

$$s(k) = -\sum_{i=1}^{N} a_i \, s(k-i) + \sum_{i=0}^{M} b_i e(k-i)$$

Déterminer N et M ainsi que la valeur des coefficients a_i et b_i

Caractérisation d'un filtre

Un filtre est défini par sa ***réponse fréquentielle en amplitude et en phase***

$$H(f) = \frac{S(f)}{E(f)} = \left|H(f)\right| e^{j\varphi(f)}$$

$$\left|H(f)\right| = \frac{\left|S(f)\right|}{\left|E(f)\right|}$$

$$\varphi(f) = Arg\left(H(f)\right)$$

En régime harmonique permanent

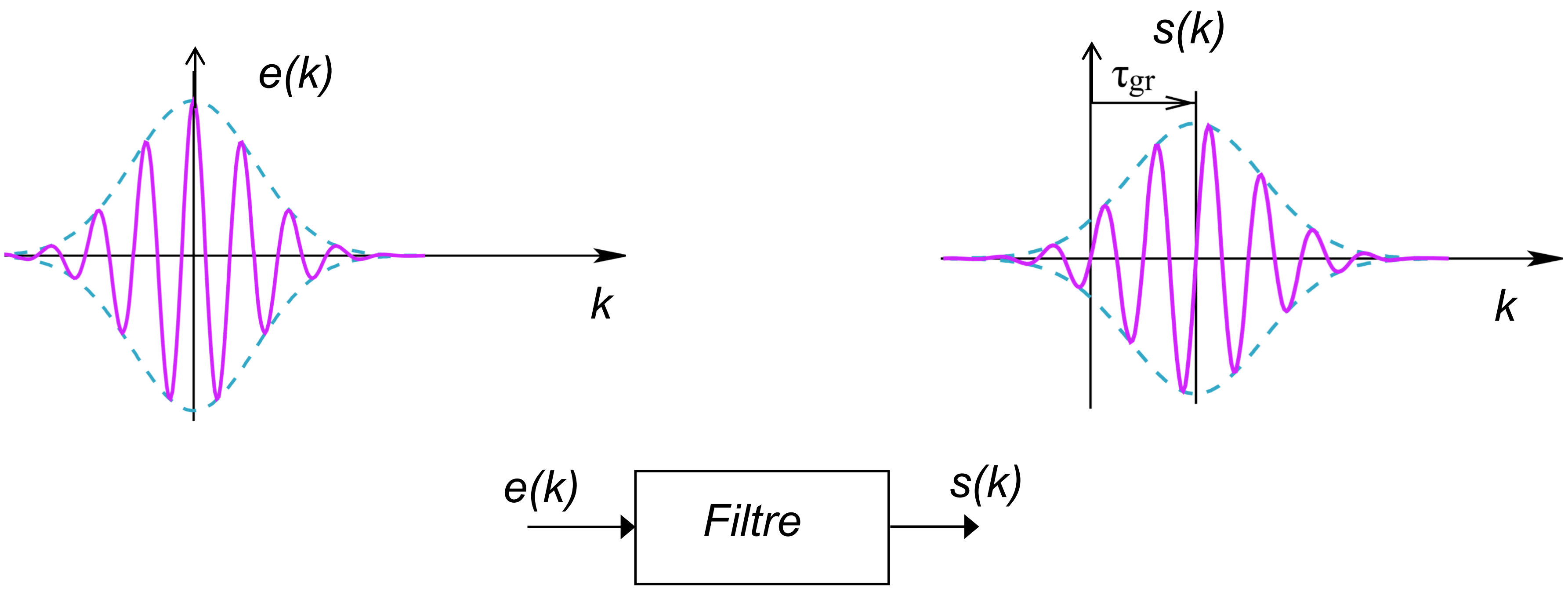
$$s(k) = \left|H(f)\right| \sin\left(2\pi f k T_e + \varphi(f)\right) = \left|H(f)\right| \sin\left(2\pi f \left(kT_e + \tau_\varphi(f)\right)\right)$$

On définit :

Retard de phase $\tau_\varphi(f) = -\dfrac{\varphi(f)}{2\pi f}$

Retard de groupe $\tau_{gr}(f) = -\dfrac{d\varphi(f)}{2\pi f}$

Illustration du concept de retour de groupe



Le retard de groupe τ_{gr} mesure le temps mis par l'énergie pour atteindre la sortie du filtre lorsque le signal n'est pas purement sinusoïdal (propagation de paquets d'onde par exemple comme ci-dessus)

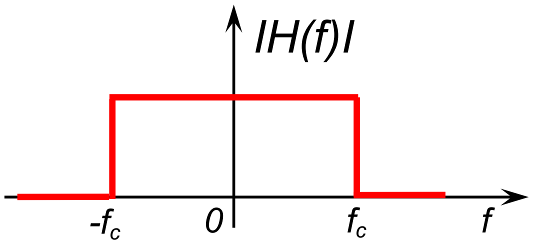
Spécifications idéales en amplitude et phase

Un traitement n'apporte pas de distorsions s'il restitue en sortie un signal de même forme qu'en entrée. *s(k)* peut subir une amplification et un retard

$$s(k) = K\, e(k - k_0)$$

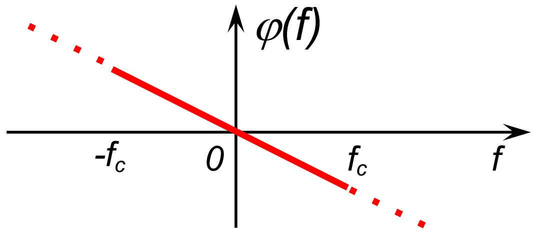
$$H(f) = K e^{-j2\pi f k_0 T_e}$$

$$S(f) = K e^{-j2\pi f k_0 T_e} E(f)$$



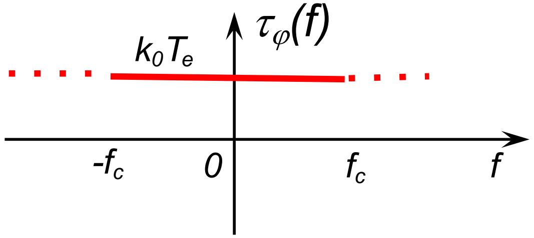
Spécifications en amplitude : $|H(f)| = |K| \quad |f| < f_c$

Le filtre idéal doit avoir une réponse en ***<u>amplitude</u>*** ***<u>constante</u>*** *dans la bande passante*



Spécifications en phase : $\varphi(f) = -2\pi k_0 T_e f \quad |f| < f_c$

$$\tau_\varphi(f) = -\frac{\varphi(f)}{2\pi f} = k_0 T_e \qquad \tau_g(f) = -\frac{d\varphi(f)}{2\pi f} = 0$$

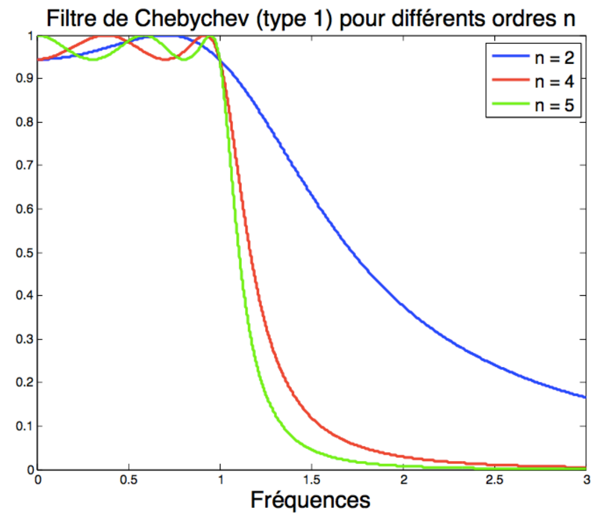


Le filtre idéal doit avoir une ***<u>phase linéaire</u>*** *dans la bande passante ce qui se traduit par un <u>retard de phase</u>* <u>constant</u> *(retard de groupe nul)*

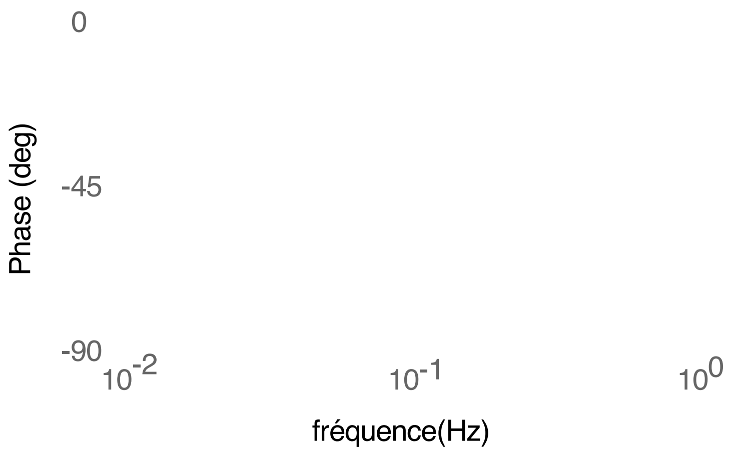
Types de distorsion en pratique

Distorsion d'amplitude

- Si la réponse fréquentielle en amplitude n'est pas constante dans la bande passante, les composantes fréquentielles ne sont pas toutes amplifiées ou atténuées de la même manière



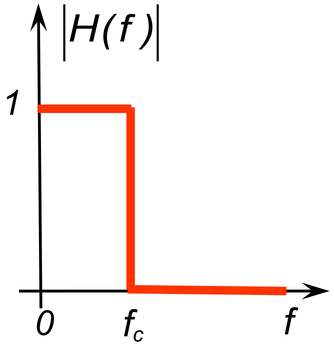
- ***Distorsion de phase***
  - Si la réponse fréquentielle en phase n'est pas linéaire dans la bande passante, les composantes fréquentielles ne sont pas toutes retardées de la même manière



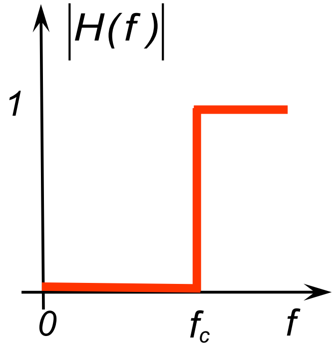
Réponse fréquentielle des 4 filtres idéaux

Selon la bande à supprimer, on distingue 4 types de filtres

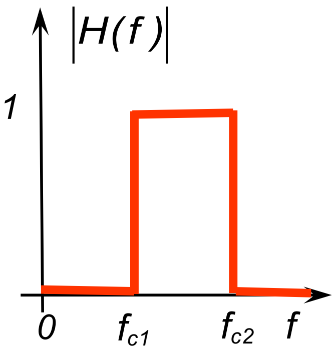
Passe-bas



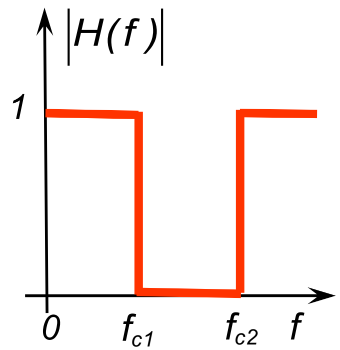
Passe-haut



Passe-bande



Coupe-bande

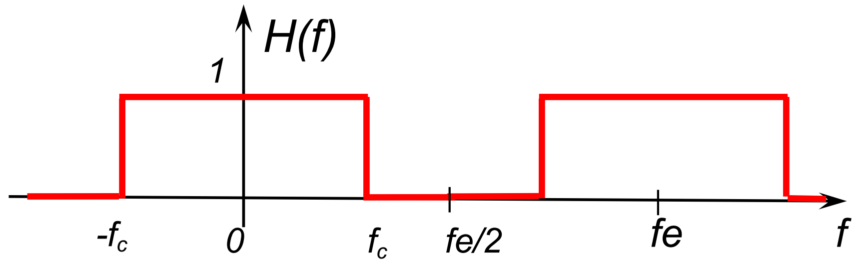


Réponse impulsionnelle d'un filtre passe-bas idéal

Réponse fréquentielle du filtre numérique passe-bas idéal

$$H(f) = \begin{cases} 1 \quad si \quad |f| < f_c \\ 0 \quad si \quad |f| \geq f_c \end{cases}$$

$$H\left(f + nf_e\right) = H(f), \quad n \in Z$$



Réponse impulsionnelle calculée par TFtd inverse

$$\boxed{h(k) = TFtd^{-1}\left(H(f)\right)}$$

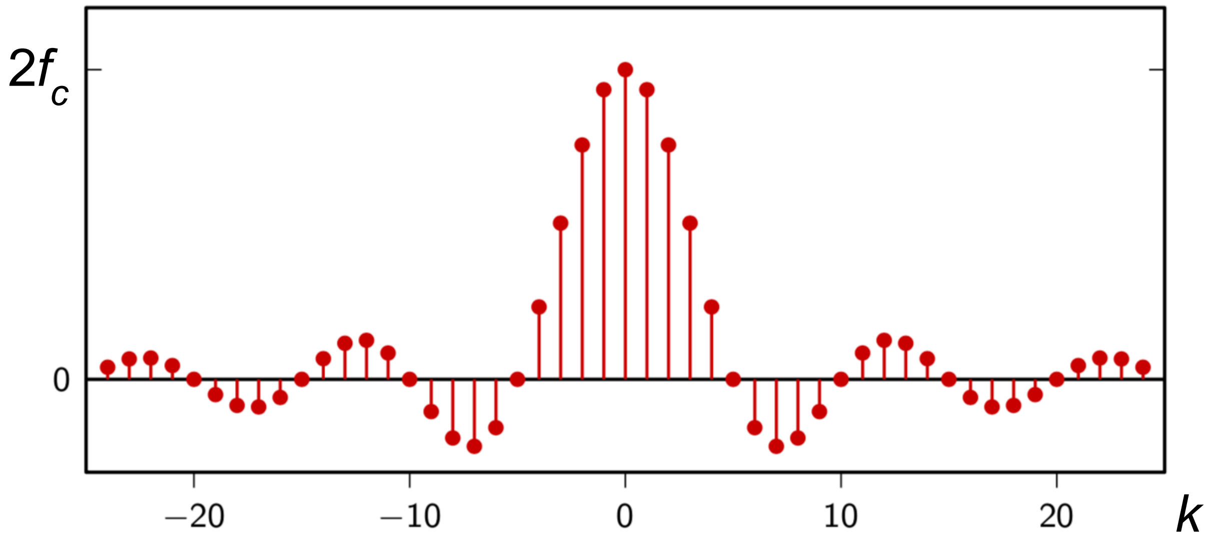
$$h(k) = \int_{-f_e/2}^{f_e/2} H(f)\, e^{j2\pi fkT_e}\, df = \int_{-f_c}^{f_c} 1\, e^{j2\pi fkT_e}\, df$$

$$h(k) = \frac{1}{\pi kT_e}\left(\frac{e^{j2\pi kT_e f_c} - e^{j2\pi kT_e f_c}}{2j}\right) = \frac{\sin\left(2\pi kT_e f_c\right)}{\pi kT_e}$$

$$\boxed{h(k) = 2f_c \operatorname{sinc}\left(2f_c kT_e\right)}$$

Réponse impulsionnelle d'un filtre numérique passe-bas idéal

$$h(k) = 2f_c \mathrm{sinc}\left(2f_c k T_e\right)$$



h(k)≠ 0   k < 0,   ***réponse non causale***

h(±∞)≠ 0   ***réponse infinie !***

Filtre idéal non réalisable

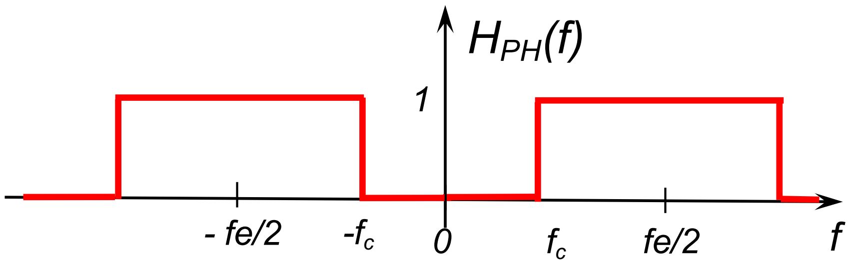
En pratique, on va approcher le filtre idéal !

Réponse impulsionnelle d'un filtre passe-haut idéal d'après celle du passe-bas idéal

Réponse fréquentielle du filtre numérique passe-haut idéal

$$H_{PH}(f) = \begin{cases} 1 & si \ |f| \geq f_c \\ 0 & si \ |f| < f_c \end{cases}$$

$$H_{PH}\left(f + nf_e\right) = H_{PH}(f), \quad n \in Z$$



Lien entre $H_{PH}(f)$ et $H_{PB}(f)$

$$H_{PH}(f) = 1 - H_{PB}(f)$$

Réponse impulsionnelle calculée par TFtd inverse

$$h_{PH}(k) = TFtd^{-1}\left(H_{PH}(f)\right)$$

$$h_{PH}(k) = TFtd^{-1}\left(1\right) - TFtd^{-1}\left(H_{PB}(f)\right)$$

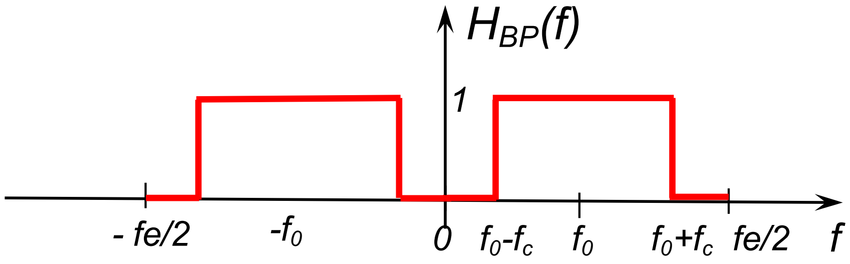
$$h_{PH}(k) = \delta(k) - 2f_c \mathrm{sinc}\left(2f_c k T_e\right)$$

Réponse impulsionnelle d'un filtre passe-bande idéal d'après celle du passe-bas idéal

Réponse fréquentielle du filtre passe-bande (Band-Pass - BP) idéal

$$H_{BP}(f) = \begin{cases} 1 & si \quad |f \pm f_0| \le f_c \\ 0 & si \quad |f \pm f_0| > f_c \end{cases}$$

$$H_{BP}(f + nf_e) = H_{BP}(f), \quad n \in Z$$



Lien entre $H_{BP}(f)$ et $H_{PB}(f)$

$$H_{BP}(f) = H_{PB}(f + f_0) + H_{PB}(f - f_0) \qquad \text{modulation}$$

Réponse impulsionnelle calculée par TFtd inverse

$$h_{BP}(k) = TFtd^{-1}(H_{PB}(f))$$

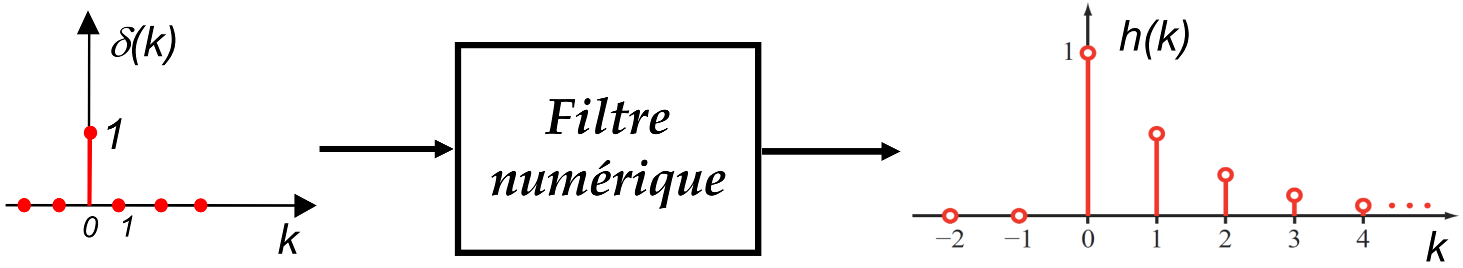
$$h_{BP}(k) = TFtd^{-1}(H_{PB}(f + f_0)) + TFtd^{-1}\left(H_{PB}(f - f_0)\right)$$

$$h_{BP}(k) = 4f_c \cos\left(2\pi f_0 k T_e\right) \mathrm{sinc}\left(2f_c k T_e\right)$$

$$TFtd^{-1}(H_{PB}(f \pm f_0)) = e^{\mp j2\pi fkT_e} h_{PB}(k)$$

Contraintes pratiques sur la réponse impulsionnelle pour la conception d'un filtre numérique

- Rappel
  - ***réponse impulsionnelle*** $h(k)$ = réponse du filtre lorsque l'entrée est une impulsion de Kronecker $\delta(k)$

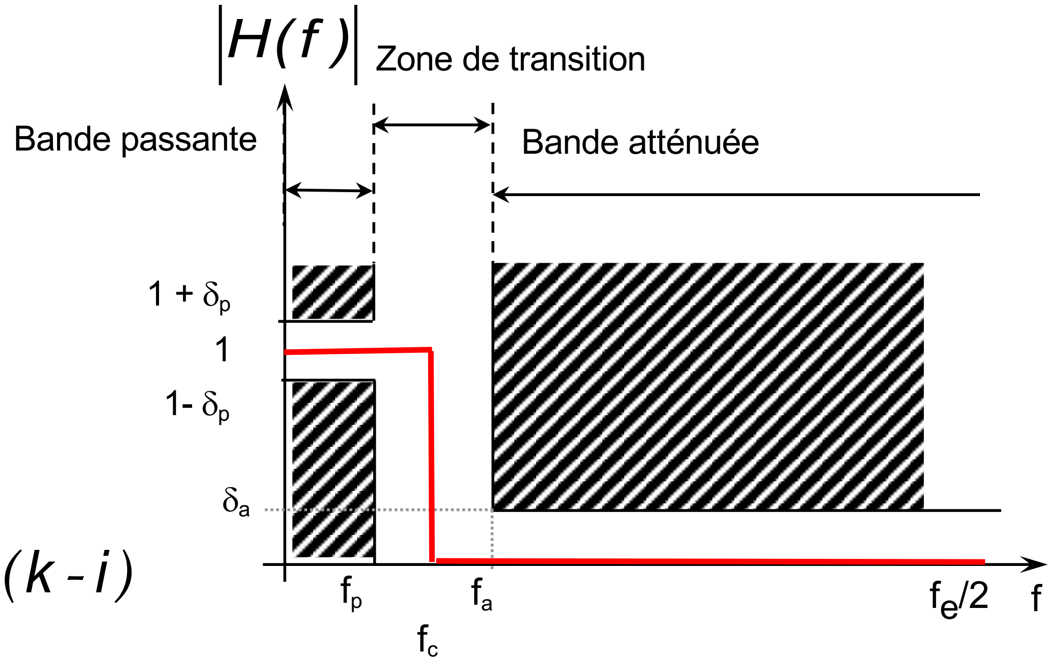


- *Contraintes sur* $h(k)$
  - **Causalité** $\quad h(k) = 0 \quad pour \quad k < 0$
  - **Stabilité** $\quad \sum_{k=-\infty}^{+\infty} |h(k)| < \infty$
  - **Réponse finie à valeurs réelles** $\quad h(k) \in \Re \quad pour \quad 0 \leq k < L$

Conception de filtres numériques

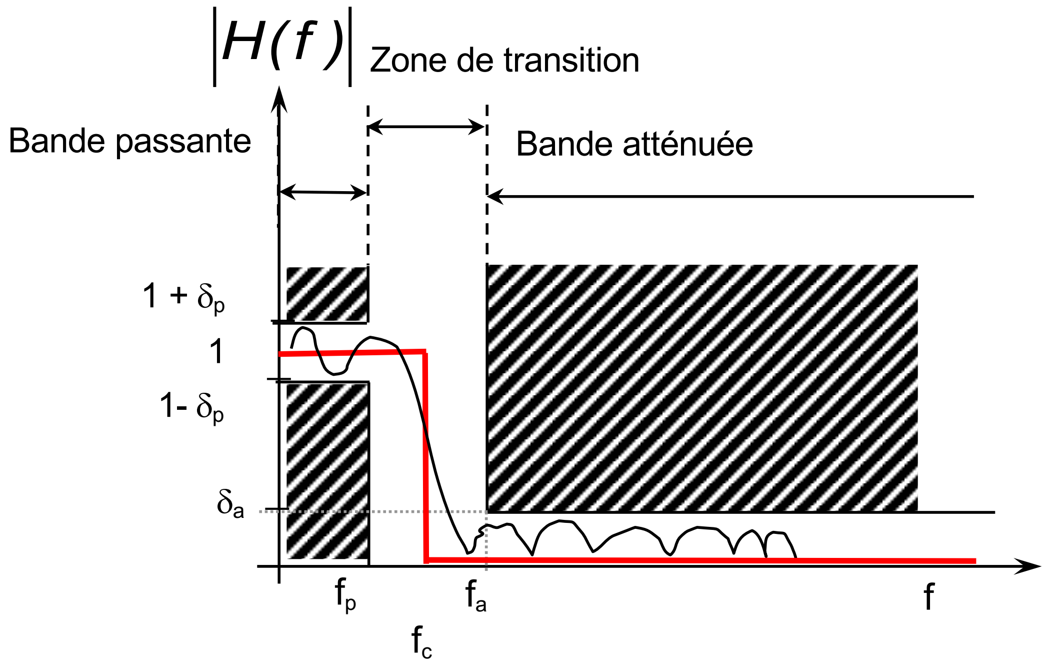
- ***Concevoir un filtre numérique*** consiste à déterminer la fonction de transfert en *z* du filtre (et donc son équation aux différences) qui va approcher au mieux les spécifications sur la réponse fréquentielle en amplitude (*celles de la phase sont complexes à satisfaire simultanément*)

$$H(z) = \frac{\sum_{i=0}^{M} b_i z^{-i}}{1 + \sum_{i=1}^{N} a_i z^{-i}}$$



$$s(k) = -\sum_{i=1}^{N} a_i\, s(k-i) + \sum_{i=0}^{M} b_i e(k-i)$$

Réponse fréquentielle désirée : gabarit



Principe **:** Il faut respecter *(ou définir)* le cahier des charges et donc préciser :

- la (ou les) fréquence(s) de coupure f_c désirée (s)
- les valeurs limites de la bande passante f_p et bande atténuée f_a
- les valeurs permises des ondulations en bande passante δ_p et atténuée δ_a
- la largeur de la zone de transition Δf permise
- éventuellement l'ordre maximal permis.

Exemple de gabarit : filtre passe-bas

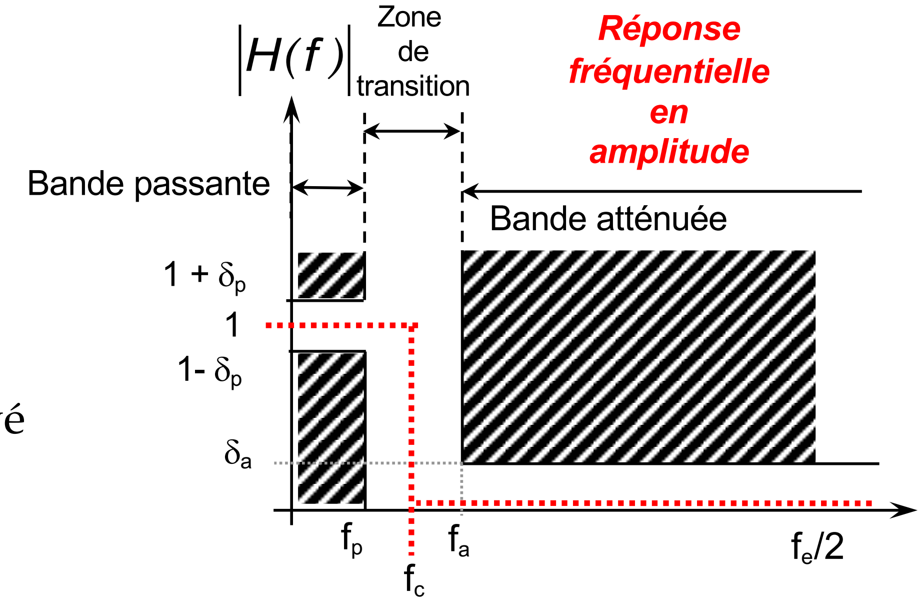
La réponse fréquentielle est définie par :

$$H(f) = \begin{cases} 1 & si \ |f| < f_p \\ 0 & si \ |f| > f_a \end{cases}$$

avec $f_c = 0.5(f_p + f_a)$

Plus f_p et f_a sont proches, plus l'ordre sera élevé

Pour un filtre idéal, $f_p = f_a = f_c$



On spécifie le ***gabarit du filtre*** en donnant :

- l'ondulation en bande passante (BP)

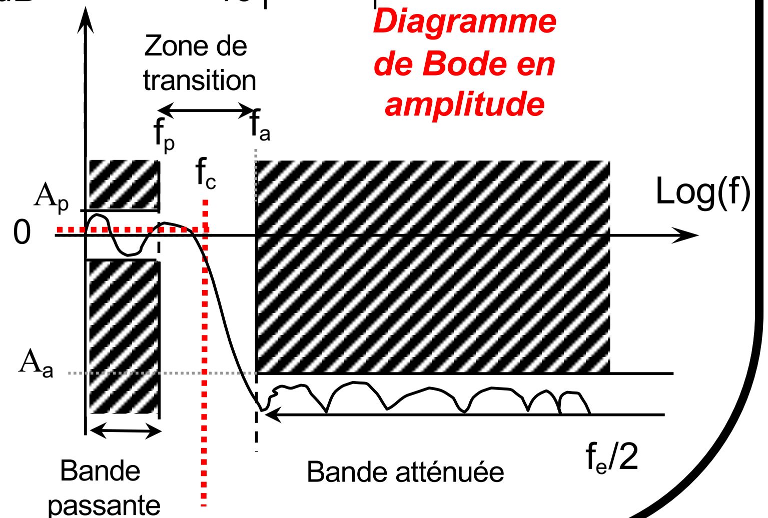
$$A_p(\mathrm{dB}) = 20\log_{10}(1+\delta_p) \text{ ou } \quad \delta_p = 10^{0.05A_p} - 1$$

- l'ondulation en bande atténuée (BA)

$$A_a(\mathrm{dB}) = -20\log_{10}\delta_a \quad \text{ou} \quad \delta_a = 10^{-0.05A_a}$$

- les fréquences f_p et f_a

$$H_{dB} = 20\log_{10}|H(f)|$$



Le déciBel (dB) - *Rappels*

Le diagramme de Bode en amplitude consiste à tracer le module de $H(f)$, exprimé en décibels (dB)

$$H_{dB} = 20\log_{10}\left(\left|H(f)\right|\right) = 20\log_{10}\left(\frac{\left|Y(f)\right|}{\left|E(f)\right|}\right)$$

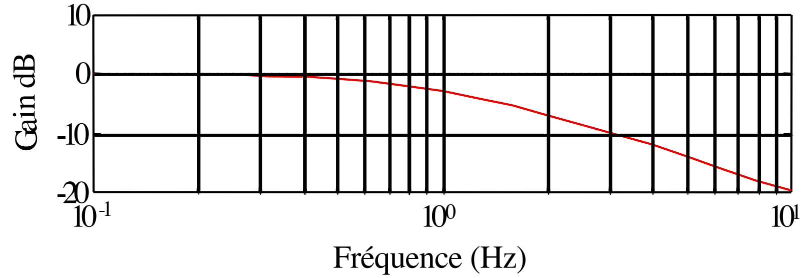


Tableau de correspondance entre rapport d'amplitude des signaux sinusoïdaux d'entrée/sortie et le gain en dB

$\lvert Y(f)/E(f)\rvert$	Gain (dB)
10^{-n}	$-20 \times n$
0.01	-40
0.1	-20
0.316	-10
0.5	-6
0.707	-3

$\lvert Y(f)/E(f)\rvert$	Gain (dB)
1	0
1.414	3
2	6
3.162	10
10	20
10^{n}	$20 \times n$

Rappel des 2 types de filtres numériques

Filtres à Réponse Impulsionnelle Finie (ou filtres RIF)

$$H(z) = \frac{S(z)}{E(z)} = \frac{\sum_{i=0}^{M} b_i z^{M-i}}{z^M} = \frac{\sum_{i=0}^{M} b_i z^{-i}}{1}$$

$$s(k) = \sum_{i=0}^{M} b_i e(k-i)$$

Filtres à Réponse Impulsionnelle Infinie (ou filtres RII)

$$H(z) = \frac{S(z)}{E(z)} = \frac{\sum_{i=0}^{M} b_i z^{-i}}{1 + \sum_{i=1}^{N} a_i z^{-i}}$$

$$s(k) = -\sum_{i=1}^{N} a_i s(k-i) + \sum_{i=0}^{M} b_i e(k-i)$$

Méthodes de conception de filtres RIF

Il en existe de nombreuses dont :

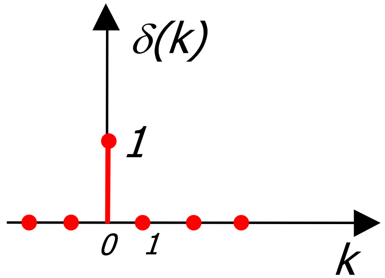
- ***la méthode de la fenêtre***

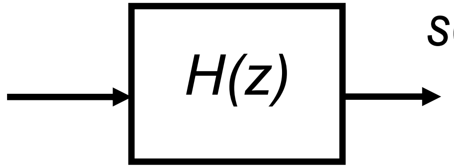
- la méthode d'échantillonnage fréquentiel

- ...

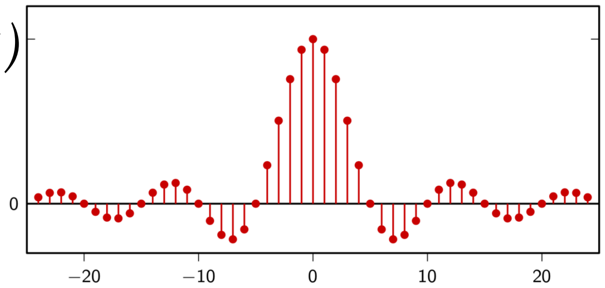
Principe de la méthode de la fenêtre

- Si *H(z)* est connue, la ***réponse impulsionnelle*** *h(k)* est la réponse obtenue lorsqu'on envoie en entrée une impulsion de Kronecker $\delta(k)$

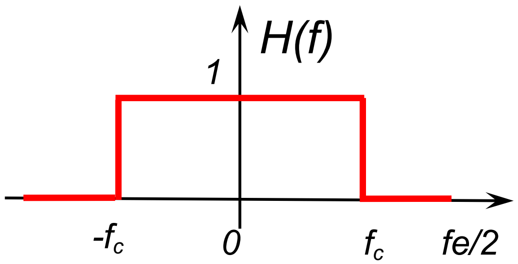




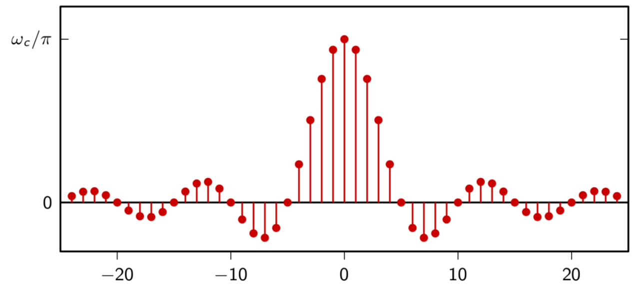
$$s(k) = h(k)$$



- Si *H(z)* est inconnue, mais que *H(f)* est spécifiée, *h(k)* est *la TFtd inverse de H(f)*



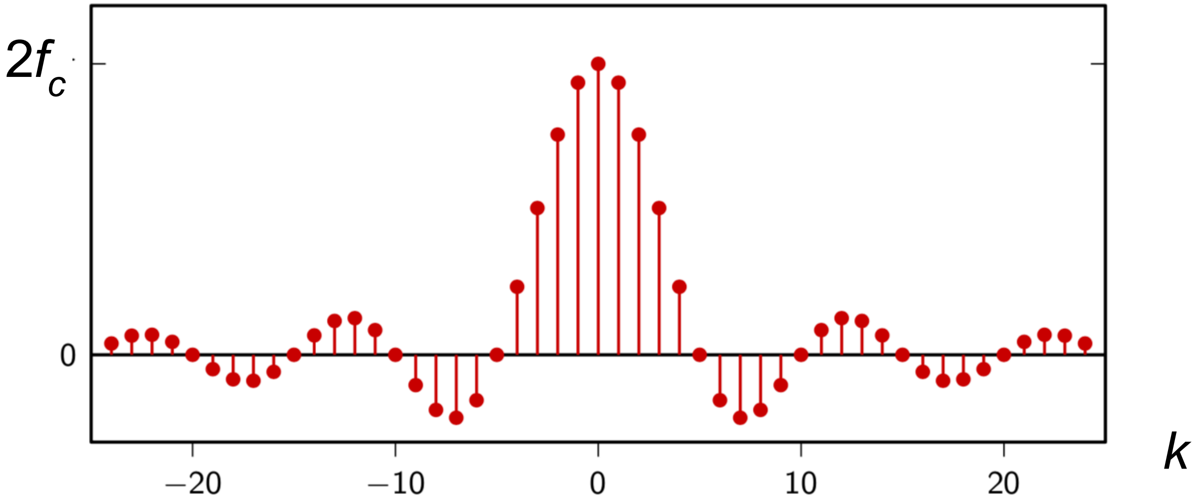
$$h(k) = TFtd^{-1}\left(H(f)\right)$$



$$H(z) = TZ(h(k)) = \sum_{k=-\infty}^{+\infty} h(k)\, z^{-k}$$

Prise en compte des contraintes pratiques sur la réponse impulsionnelle d'un filtre passe-bas idéal

$$h(k) = 2f_c \mathrm{sinc}\left(2f_c k T_e\right)$$



$h(\pm\infty) \neq 0$ ***réponse infinie !***

En pratique : on va tronquer (fenêtrer) la réponse impulsionnelle idéale

$h(k) \neq 0$ $k < 0$, ***réponse non causale !***

En pratique : on va retarder la réponse impulsionnelle

Conception de filtres RIF par la méthode de la fenêtre

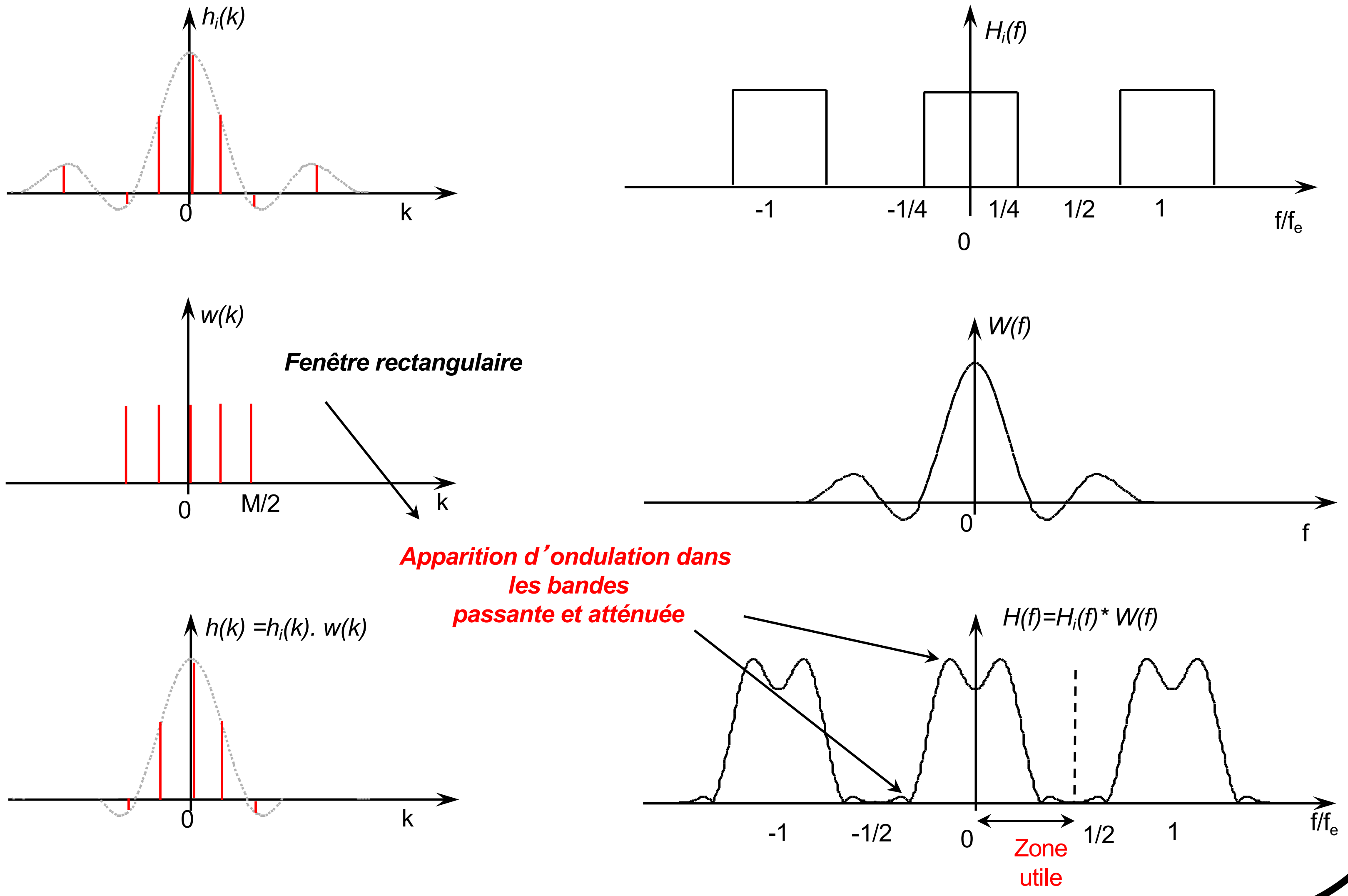
Principe

1. On détermine par TFtd inverse la réponse impulsionnelle du filtre idéal
2. On tronque la réponse impulsionnelle (revient à multiplier par une fenêtre)
3. On retarde cette réponse pour la rendre causale
4. Les valeurs successives de la réponse impulsionnelle sont les coefficients du filtre recherché $b(k)=h(k)$

$$H(z)=\frac{S(z)}{E(z)}=\sum_{i=0}^{M}h_i z^{-i}$$

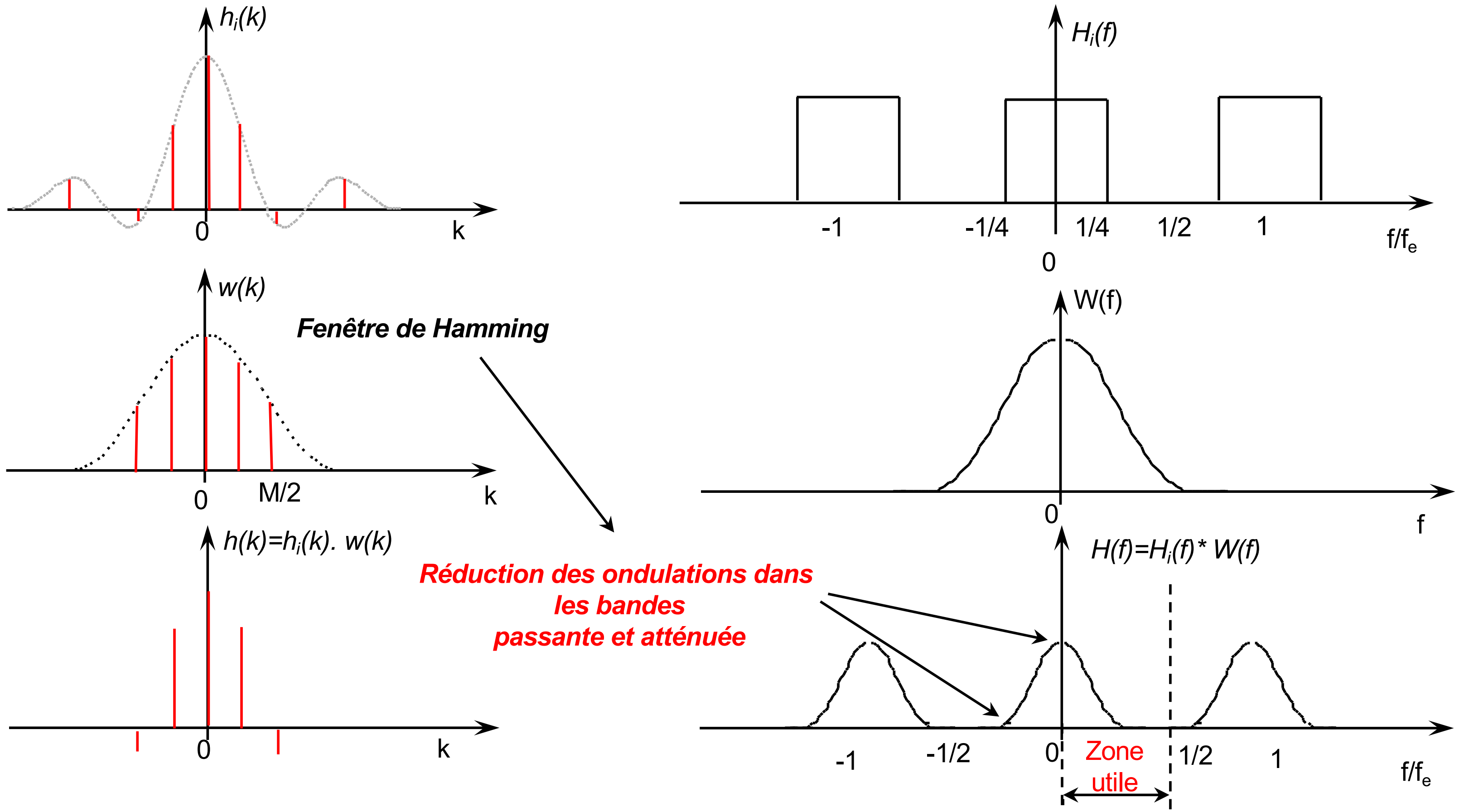
$$s(k)=\sum_{i=0}^{M}h_i e(k-i)$$

Synthèse de filtres RIF avec la fenêtre naturelle



Réduction des ondulations - Fenêtrage

Pour diminuer les ondulations en bandes passante et atténuée, on peut choisir une autre fenêtre : Hamming, Hanning, Blackmann, Kaiser, …



Exemple de synthèse sous Matlab :

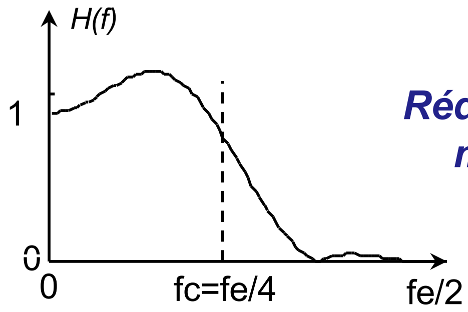
fir1(M,fcn,window(M+1))

ordre M=5	ordre M=5
Fenêtre **rectangulaire**	Fenêtre de **hamming**
h=fir1(5,0.5,boxcar(6))	***h=fir1(5,0.5,hamming(6))***

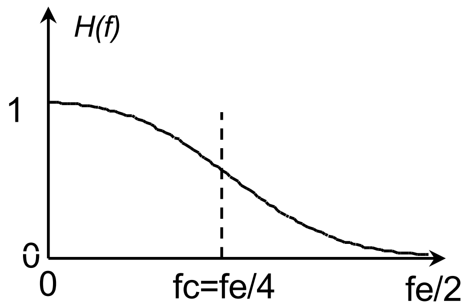
Sous Matlab, les fréquences de coupure sont normalisées par rapport à f_e / 2

fc=fe/4 d'où fn=fc/(fe/2)=0.5

Fenêtre rectangulaire	Fenêtre de hamming
$h(0) = h(6) = -0.0882$	$h(0) = h(6) = -0.0078$
$h(1) = h(5) = 0.1471$	$h(1) = h(5) = 0.0645$
$h(2) = h(3) = 0.4412$	$h(2) = h(3) = 0.4433$



Réduction des ondulations mais diminution de la raideur de coupure

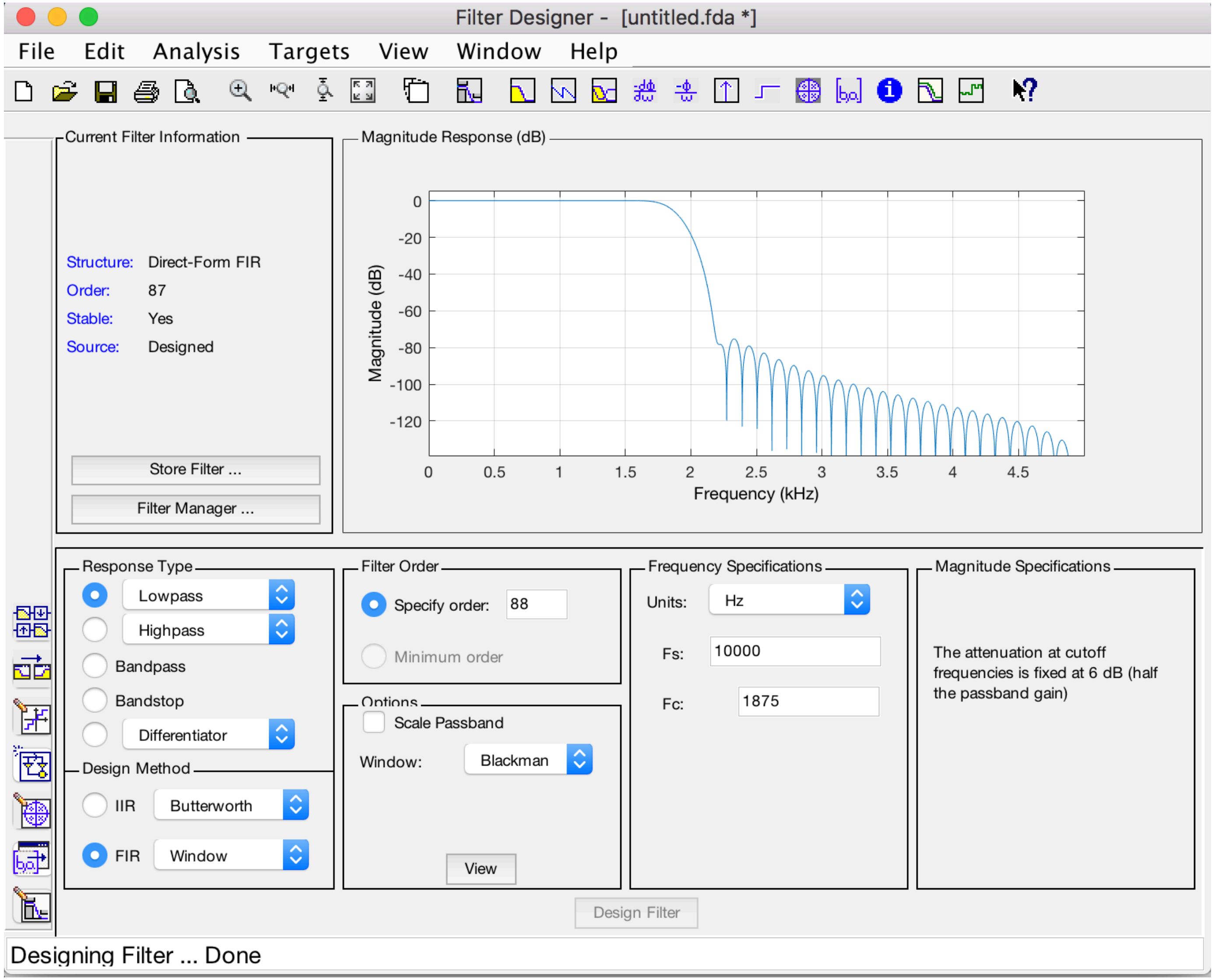


Exemple : synthèse d'un filtre passe-bas avec l'interface graphique ***filterDesigner*** de Matlab

Synthétiser un filtre numérique RIF passe-bas ayant les caractéristiques suivantes :

- ordre M=88
- fréquence d'échantillonnage f_e = 10 kHz
- fréquence centrale : f_0 = 1875 Hz
- fenêtre de Blackman

Synthèse de filtres RIF avec l'interface graphique *filterDesigner* de Matlab



Méthodes de conception de filtres RII

Il en existe de nombreuses dont :

- la méthode de l'invariance impulsionnelle

- ***la méthode qui exploite le lien entre réponse fréquentielle et la localisation des pôles et des zéros***

- ***la méthode de la transformation bilinéaire***

- **...**

Lien entre pôles et zéros de *H(z)* et réponse fréquentielle

$$H(z)=\frac{\left(z-e^{j\Omega_0}\right)\left(z-e^{-j\Omega_0}\right)}{\left(z-ae^{j\Omega_0}\right)\left(z-ae^{-j\Omega_0}\right)}$$

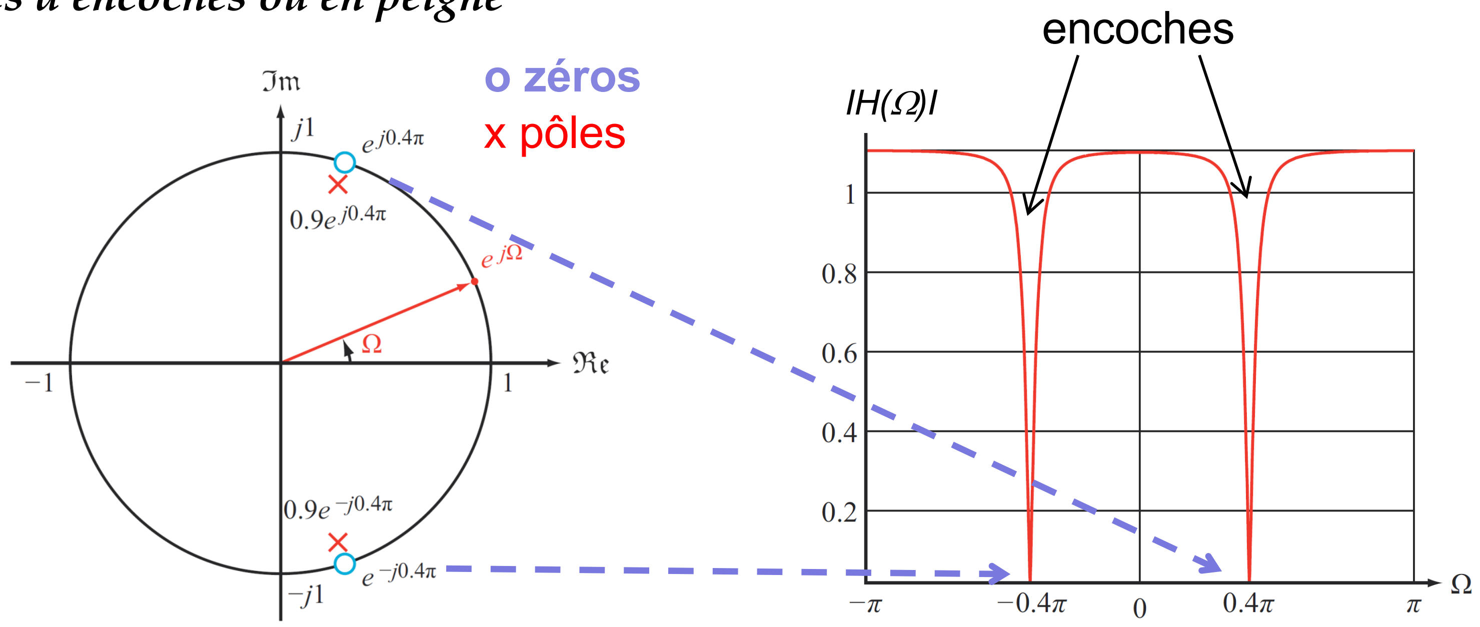
$$z=e^{j\Omega T_e}$$

$$H(\Omega)=\left|H(\Omega)\right|e^{j\varphi(\Omega)}$$

Des ***zéros*** proches ou sur le cercle unité produisent des minima au niveau de la réponse fréquentielle en amplitude.

Des ***pôles*** proches du cercle unité produisent de larges pics sur la réponse fréquentielle en amplitude

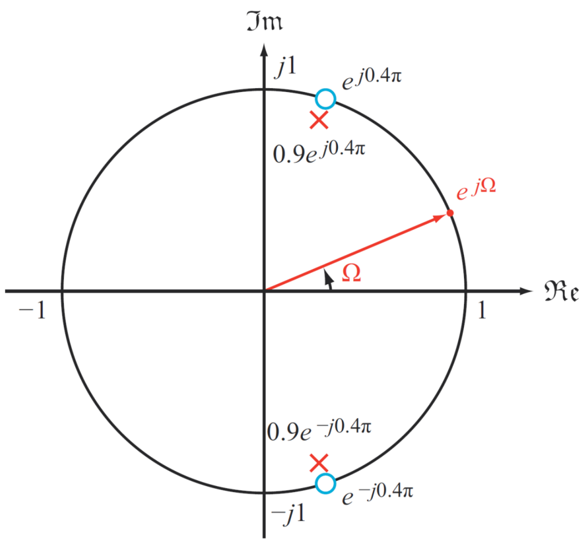
Il est donc possible, par un placement judicieux des pôles et des zéros, d'obtenir des filtres à encoches ou en peigne

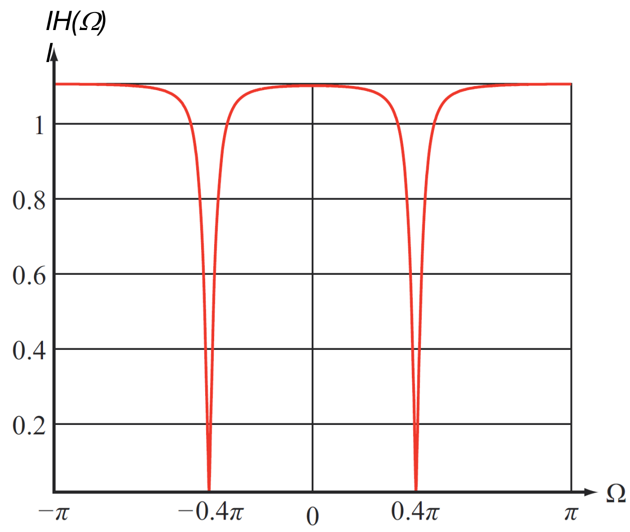


Filtres à encoches (*notch filter*) pour supprimer une composante sinusoïdale parasite

- ***But :*** concevoir un filtre qui supprime ***une et une seule*** composante sinusoïdale f_0 sans affecter les autres composantes fréquentielles
- ***Solution*** : placer deux zéros sur le cercle unité pour supprimer la composante sinusoïdale f_0 et placer deux pôles proches des zéros à l'intérieur du cercle pour ***compenser*** l'effet des zéros aux autres fréquences
- ***Remarque*** : si $f_0=0$, suppression de la composante continue

$$H(z)=\frac{\left(z-e^{j\Omega_0}\right)\left(z-e^{-j\Omega_0}\right)}{\left(z-ae^{j\Omega_0}\right)\left(z-ae^{-j\Omega_0}\right)}=\frac{z^2-2\cos(\Omega_0)z+1}{z^2-2a\cos(\Omega_0)z+a^2}\qquad \Omega_0=2\pi\frac{f_0}{f_e}\qquad a<1$$





a: permet d'ajuster la sélectivité du filtre
Plus *a* est proche de 1, plus le filtre est sélectif

Filtre à encoches *(notch filter)* pour supprimer une composante sinusoïdale parasite

Exemple : concevoir un filtre numérique qui supprime la composante sinusoïdale $f_0 = 200\ Hz$ lorsque le signal est échantillonné à $f_e = 2000\ Hz$

•*Fonction de transfert*

$$H(z) = \frac{z^2 - 2\cos(\Omega_0)z + 1}{z^2 - 2a\cos(\Omega_0)z + a^2} \qquad \Omega_0 = 2\pi\frac{f_0}{f_e} \qquad a < 1$$

$$\Omega_0 = 2\pi\frac{f_0}{f_e} = 0,2\pi \qquad a = 0,9$$

$$H(z) = \frac{z^2 - 2\cos(0,2\pi)z + 1}{z^2 - 1.8\cos(0,2\pi)z + 0,9^2} = \frac{z^2 - 0,809z + 1}{z^2 - 1,45z + 0.81}$$

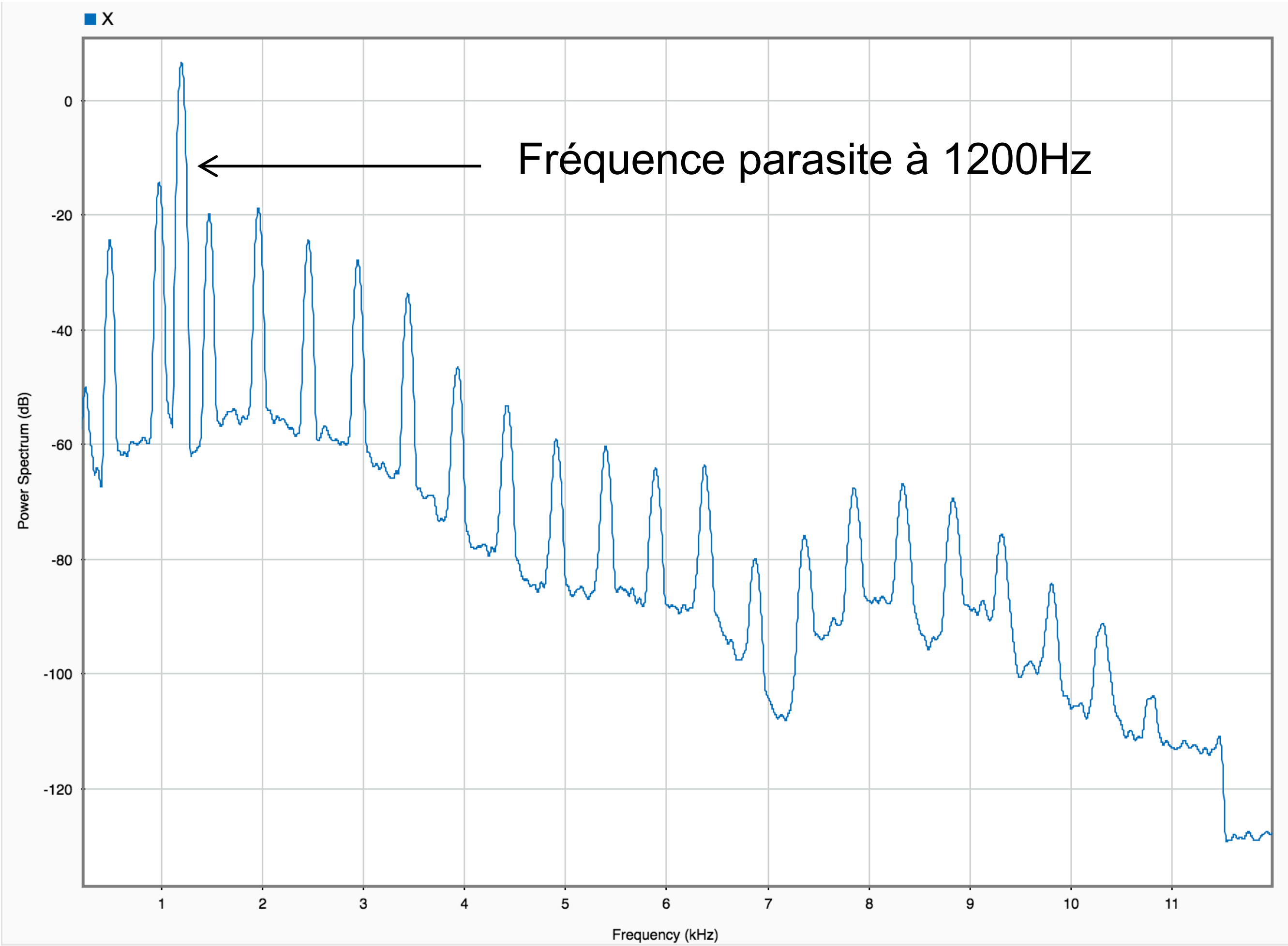
•*Equation aux différences*

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 - 0,809z^{-1} + z^{-2}}{1 - 1,45z^{-1} + 0.81z^{-2}}$$

$$y(k) = 1,45y(k-1) - 0,81y(k-2) + x(k) - 0,809x(k-1) + x(k-2)$$

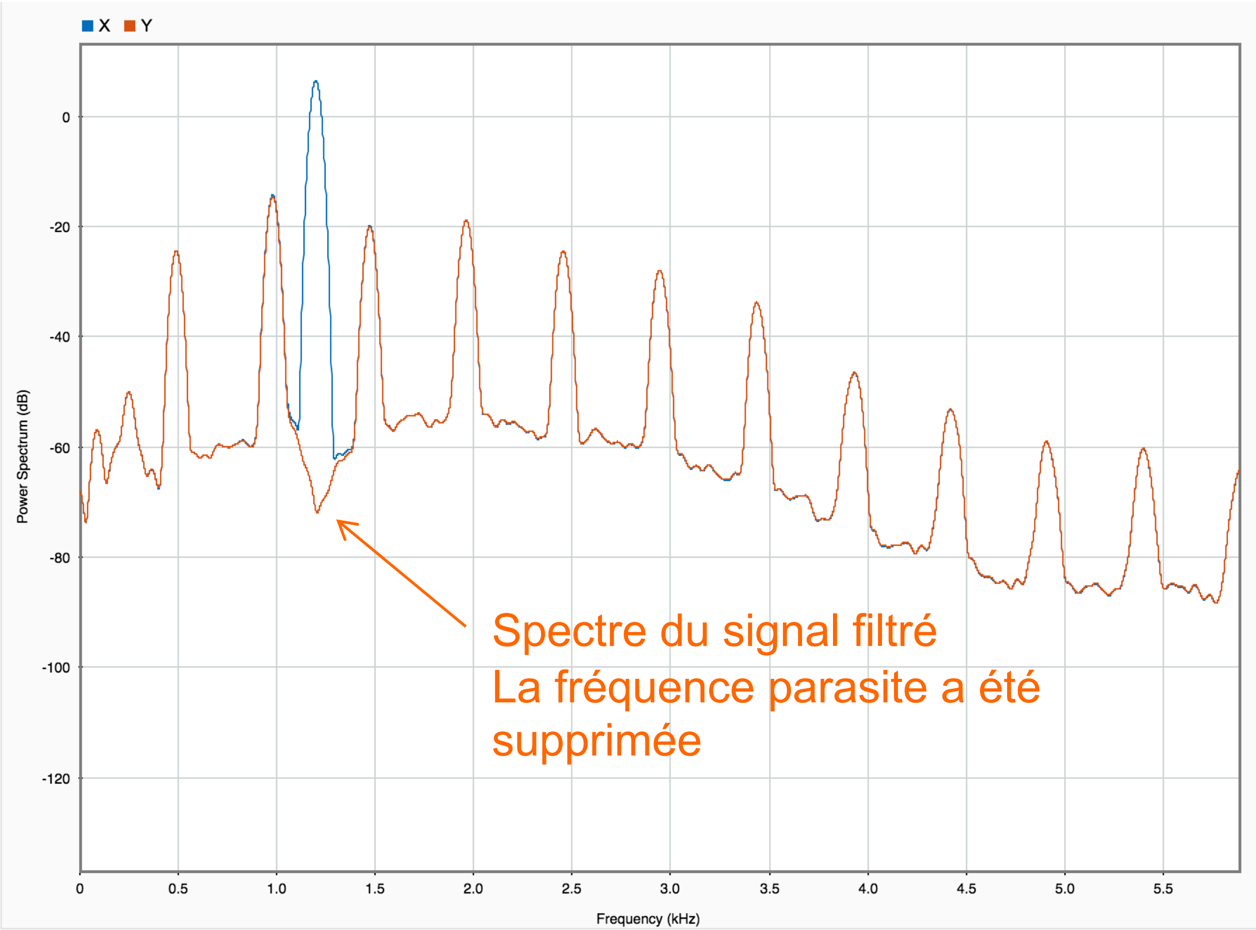
Exemple : note de trompette contaminée par une fréquence parasite

1. Analyse spectrale pour identifier la fréquence parasite



Autre exemple : note de trompette contaminée par une fréquence parasite

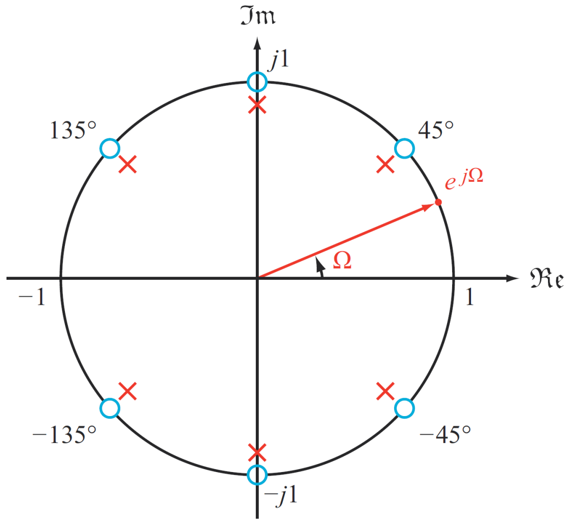
2. Concevoir et appliquer un filtre à encoche pour supprimer la fréquence parasite
3. Analyse spectrale du signal filtré

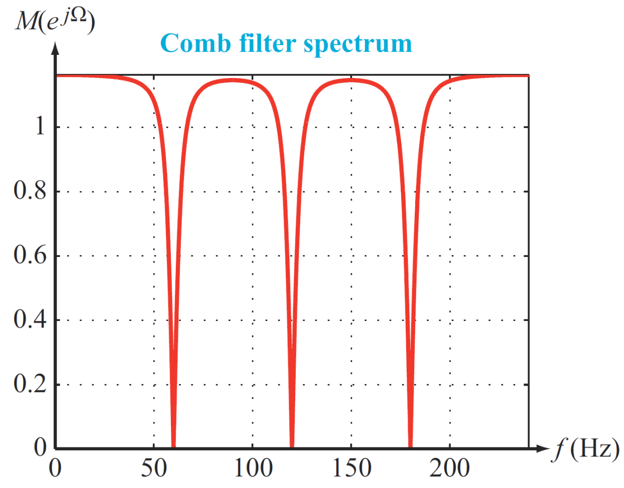


Filtres en peigne *(comb filter)* pour supprimer des signaux périodiques

- ***But :*** concevoir un filtre qui supprime les harmoniques d'un signal périodique sans affecter les autres composantes fréquentielles

- ***Solution*** : mise en cascade de 2 *(ou plus)* filtres à encoches





Exemple d'application : séparation du son produit par de 2 trompettes jouant simultanément 2 notes différentes !

Synthèse de filtres RII par transformation bilinéaire

- ***Principe***

On dispose d'un **filtre analogique** ayant un gabarit fréquentiel répondant au cahier des charges demandées *(filtre de type Butterworth par exemple)*

Objectif : trouver le filtre numérique RII ayant une réponse fréquentielle équivalente à celle du filtre analogique

Remarque : la coïncidence des réponses est limitée à la zone utile du filtre numérique, soit pour des fréquences comprises entre 0 et $f_e/2$.

Approximation avancée et transformation bilinéaire

On connaît *H(s)* Comment en déduire *H(z)* ???

Relation liant la variable de la Trans. en Z à la variable de la Trans. de Laplace :

$$z = e^{sT_e}$$

$$s = \frac{1}{T_e} \ln(z)$$

Relation non linéaire !

$$z = e^{sT_e} \approx 1 + T_e s + \dots \quad \Rightarrow \quad s = \frac{z-1}{Te}$$

Approximation avancée
=approximation de l' intégrale par la méthode des rectangles
Ne conserve pas la stabilité !

$$z = \frac{e^{\frac{sT_e}{2}}}{e^{\frac{-sT_e}{2}}} \approx \frac{1 + \frac{T_e}{2} s + \dots}{1 - \frac{T_e}{2} s + \dots} \quad \Rightarrow \quad \boxed{s = \frac{2}{T_e} \frac{z-1}{z+1} = \frac{2}{T_e} \frac{1-z^{-1}}{1+z^{-1}}}$$

Transformation bilinéaire
=approximation de l'intégrale par la méthode des trapèzes
Conserve la stabilité

Synthèse de filtre RII par la transformation bilinéaire
Exemple

Concevoir le filtre numérique équivalent à un filtre analogique ci-dessous pour une période d'échantillonnage $T_e = 1$ s.

$$H(s) = \frac{2s}{s^2 + 6s + 8}$$

a) Déterminer par la méthode bilinéaire la fonction de transfert en z du filtre RII équivalent.

b) Vérifier avec la commande Matlab

```
>>[Numd,Dend] = bilinear(Numc,Denc,Fs)
```

c) Exprimer la fonction de transfert en puissance négative de z

d) En déduire l'équation aux différences du filtre.

Synthèse de filtre RII
par la transformation bilinéaire - Exemple

$T_e = 1$ s

$$H(s)=\frac{2s}{s^2+6s+8} \xrightarrow{s=\frac{2}{T_e}\frac{1-z^{-1}}{1+z^{-1}}=2\frac{1-z^{-1}}{1+z^{-1}}} H(z)=\frac{2\times 2\frac{1-z^{-1}}{1+z^{-1}}}{\left(2\frac{1-z^{-1}}{1+z^{-1}}\right)^2+6\times 2\frac{1-z^{-1}}{1+z^{-1}}+8}$$

$$H(z)=\frac{2\times 2\left(1-z^{-1}\right)\left(1+z^{-1}\right)}{4\left(1-z^{-1}\right)^2+12\left(1-z^{-1}\right)\left(1+z^{-1}\right)+8\left(1+z^{-1}\right)^2}=\frac{4-4z^{-2}}{24+8z^{-1}}=\frac{1/6z^2-1/6}{z^2+1/3z}$$

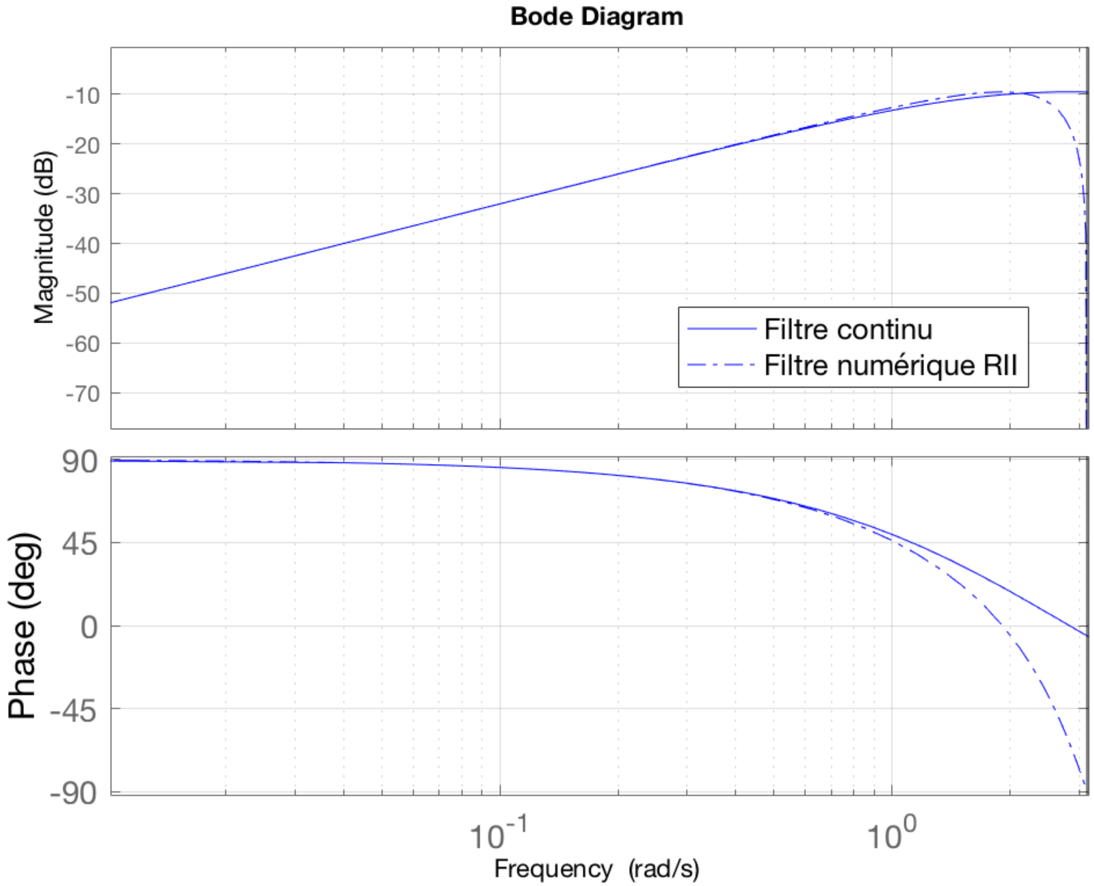
Sous Matlab :

```
>> [Numd,Dend] = bilinear([0 2 0],[1 6 8],1)
    Numd =   0.1667   0.0000   -0.1667

    Dend =    1.0000    0.3333   0.0000
```

Synthèse de filtres RII par la transformation bilinéaire - Exemple

Comparaison des réponses fréquentielles des filtres analogique et numérique



Synthèse de filtres RII
par la transformation bilinéaire - Exemple

Déduction de l'équation aux différences du filtre RII

$$H(z)=\frac{1/6z^2-1/6}{z^2+1/3z}$$

$$\frac{Y(z)}{E(z)}=\frac{1/6-1/6z^{-2}}{1+1/3z^{-1}}$$

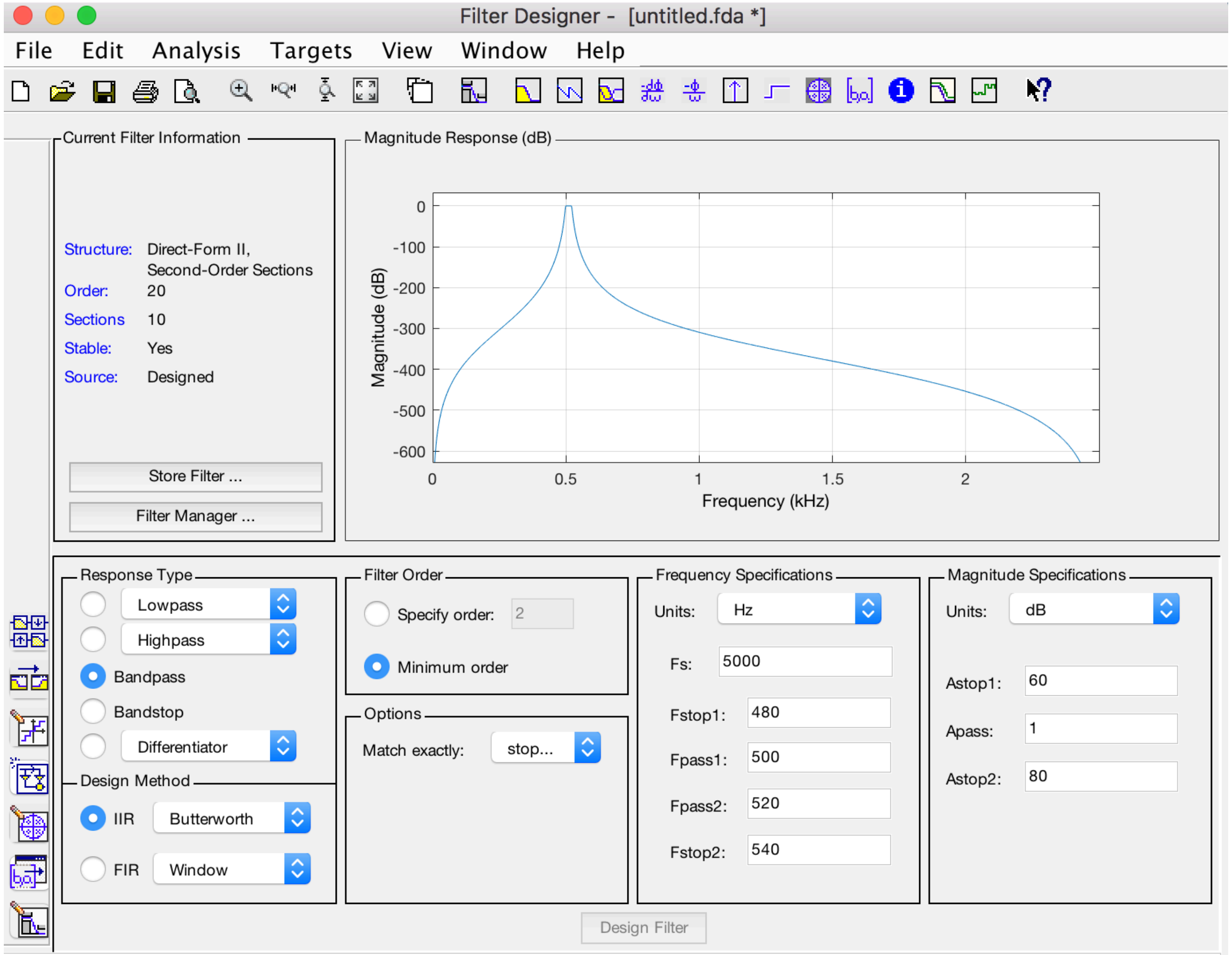
$$y(k)=-\frac{1}{3}y(k-1)+\frac{1}{6}e(k)-\frac{1}{6}e(k-2)$$

Exemple : synthèse d'un filtre passe-bande avec l'interface graphique ***filterDesigner*** de Matlab

Concevoir un filtre numérique RII passe-bande ayant les caractéristiques suivantes :

- ordre : 20
- fréquence d'échantillonnage f_e = 5 kHz
- fréquence centrale : f_0 = 500 Hz
- gain unitaire à la fréquence centrale

Synthèse de filtres RII avec l’interface graphique *filterDesigner* de Matlab



Comparatif filtres RIF / RII

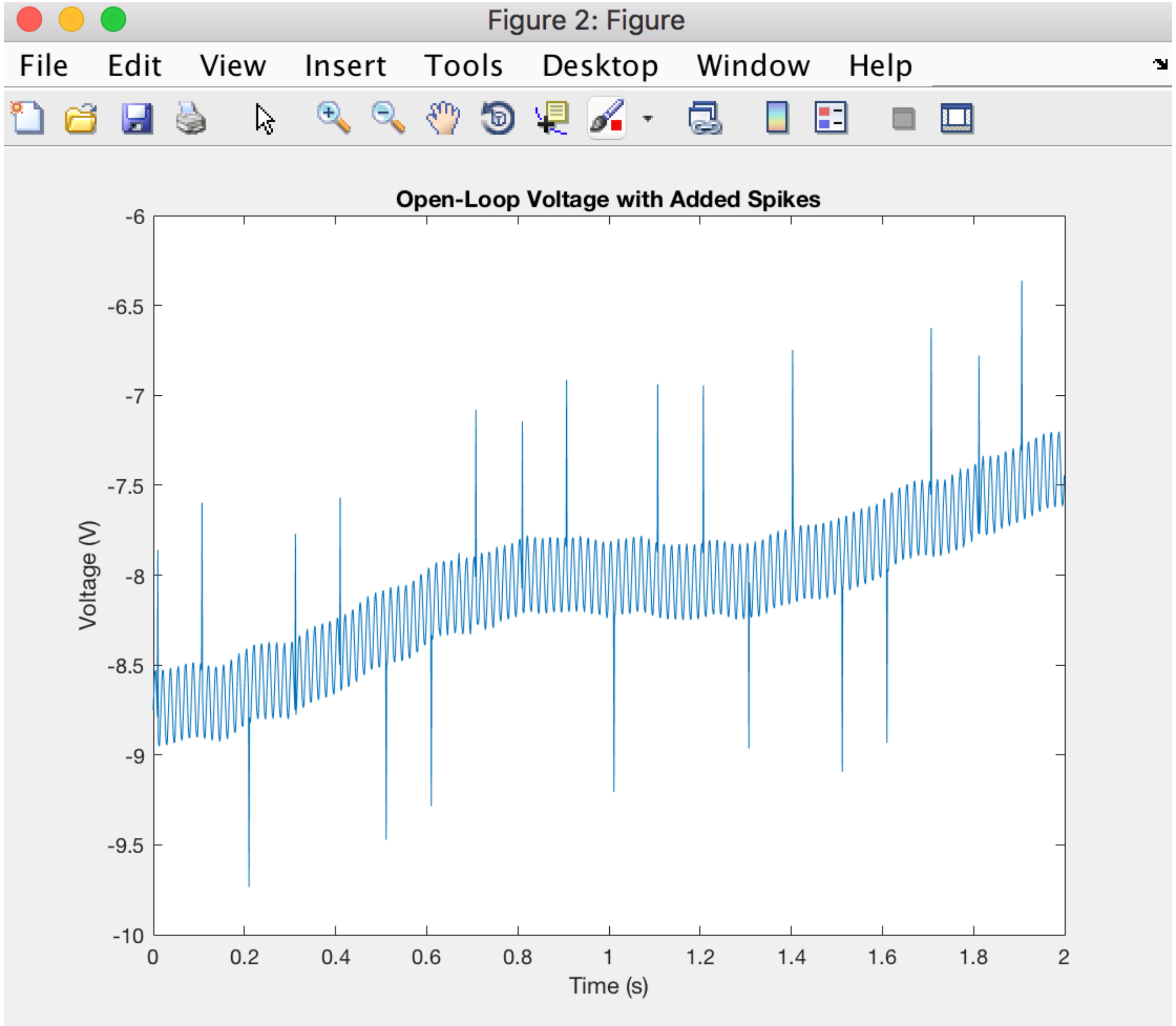
Filtres RIF

- Toujours stables
- Phase linéaire
- Faciles à concevoir
- Pas d'équivalent en analogique

Filtres RII

- Peuvent être instables
- Phase non linéaire
- Ordre souvent plus faible que RIF
  - Nécessitent moins d'opérations et de places mémoires

Comment supprimer des pics parasites ?



Un autre filtre numérique utile à connaître : le filtre médian

C'est un filtre numérique ***non linéaire*** souvent utilisé pour

- supprimer les valeurs aberrantes (*outliers*) ou pics parasites sur un signal

Principe : remplacer chaque valeur du signal par la valeur médiane calculée sur une fenêtre glissante de largeur N

Exemple : extrait d'un signal comprenant 9 valeurs : **5 7 7 8 100 7 8 6 5**

Le filtre médian va ordonner les valeurs : **5 5 6 7 7 7 8 8 100**

La valeur aberrante 100 sera remplacée par la médiane (7 ici)

La sortie du filtre médian donnera donc : **5 7 7 8 7 7 8 6 5**

La valeur aberrante a été remplacée par une valeur "de consensus" entre les valeurs voisines

Sous Matlab : y = medfilt1(x,N);

N : Largeur de fenêtre glissante pour le calcul de la médiane