Traitement Numérique du Signal

Nathalie Thomas

IRIT/ENSEEIHT
Nathalie.Thomas@enseeiht.fr

Traitement Numérique du signal

Filtrage numérique

- **Rappel : la transformée de Laplace**

→ Définition :

$$X(p) = TL\left[x(t)\right] = \int_0^{+\infty} x(p)e^{-pt}dt,\ p \in \mathbb{C}$$

→ Outil d'étude des systèmes linéaires invariants dans le temps ANALOGIQUES

$$x(t) \longrightarrow \boxed{h(t) \underset{TL^{-1}}{\overset{TL}{\rightleftarrows}} H(p)} \longrightarrow y(t) = x(t) * h(t) \xrightarrow{TL} Y(p) = H(p)X(p)$$

Fonction de transfert

- Étude temporelle (réponse indicielle, réponse à une rampe): stabilité (pôles de H(p) à partie réelle négative), rapidité (temps de montée, temps de réponse à x%), précision (erreur statique, erreur de trainage)

- Etude fréquentielle (réponse à une entrée sinusoïdale) : diagrammes de Bode => fréquence de coupure, bande passante, atténuation en bande coupée, résonnance …

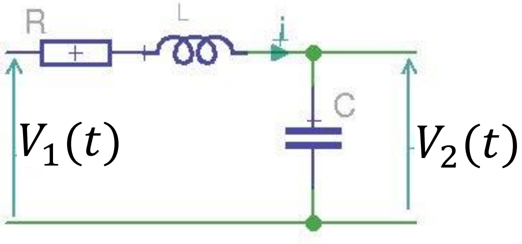
Réponse en fréquence

$$H(f) = \left[H(p)\right]_{p=j\omega=j2\pi f} \quad (TL = TF)$$

Filtrage numérique

- **Rappel : la transformée de Laplace**

→ Exemple :

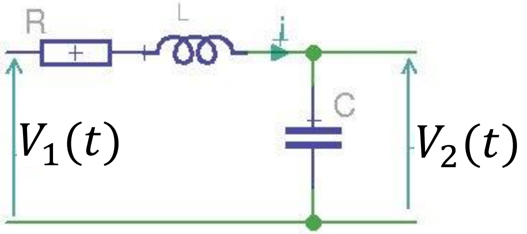


$$V_1(t) = LC\frac{d^2V_2(t)}{dt^2} + RC\frac{dV_2(t)}{dt} + V_2(t)$$

Filtrage numérique

- **Rappel : la transformée de Laplace**

→ <u>Exemple :</u>

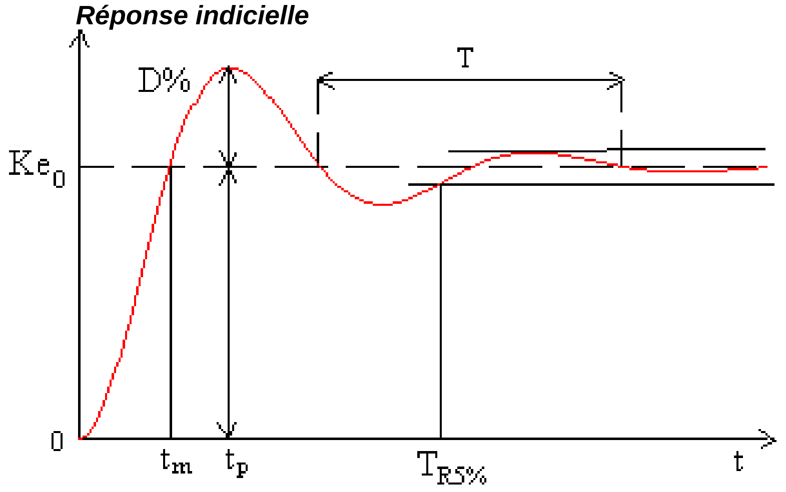


$$V_1(t) = LC\frac{d^2V_2(t)}{dt^2} + RC\frac{dV_2(t)}{dt} + V_2(t)$$

TL

$$H(p) = \frac{V_2(p)}{V_1(p)} = \frac{1}{LCp^2 + RCp + 1}$$

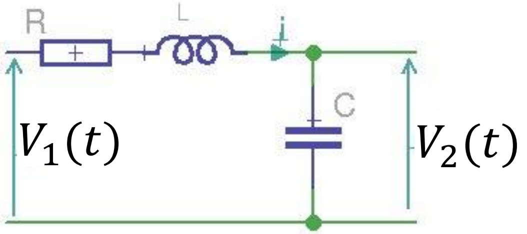
Étude temporelle



Filtrage numérique

- **Rappel : la transformée de Laplace**

→ Exemple :



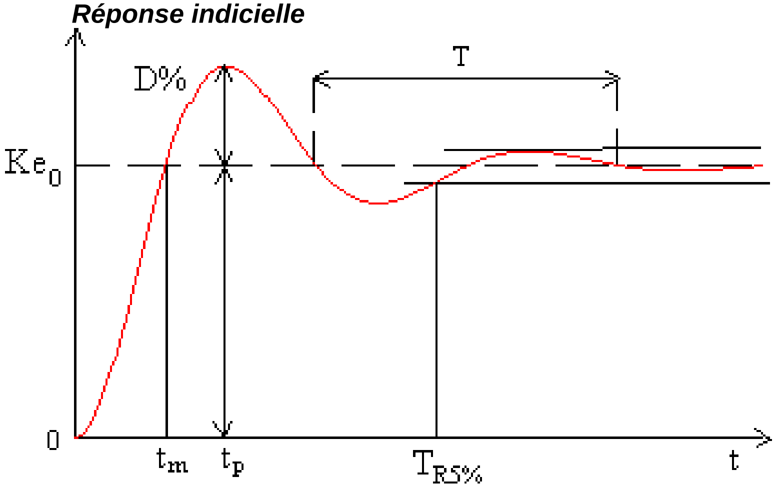
$$V_1(t) = LC\frac{d^2V_2(t)}{dt^2} + RC\frac{dV_2(t)}{dt} + V_2(t)$$

TL

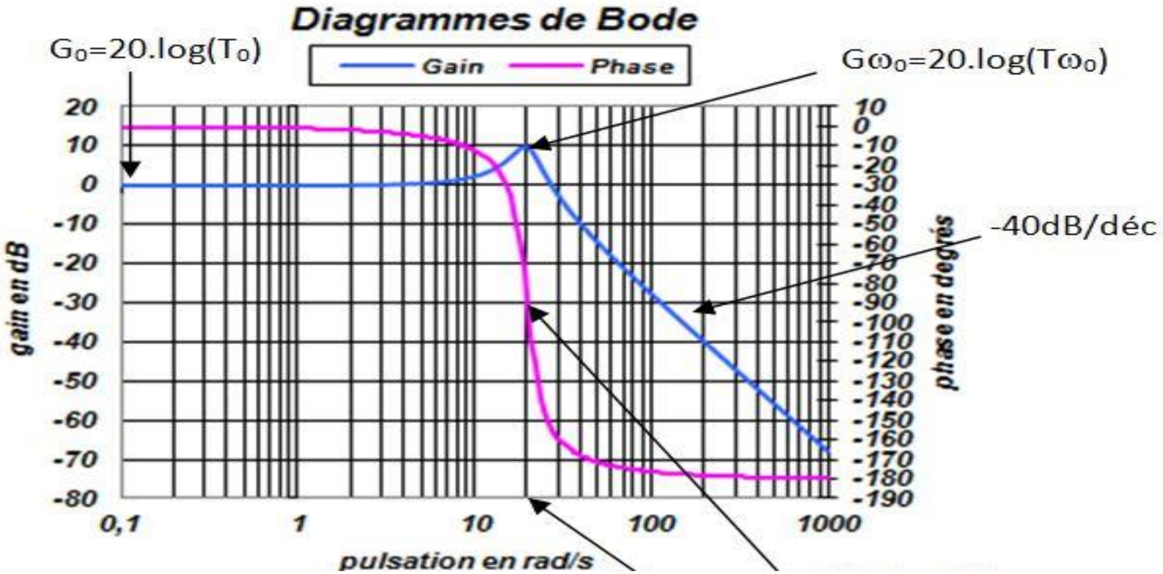
$$H(p) = \frac{V_2(p)}{V_1(p)} = \frac{1}{LCp^2 + RCp + 1}$$

$p = j\omega$

Étude temporelle



Étude fréquentielle



Filtrage numérique

- **La transformée en z**

→ <u>Définition :</u> $$X(z) = TZ\left[x(n)\right] = \sum_{n=-\infty}^{+\infty} x(n) z^{-n}, z \in \mathbb{C}$$

Filtrage numérique

- **La transformée en z**

→ Définition : $$X(z) = TZ\left[x(n)\right] = \sum_{n=-\infty}^{+\infty} x(n) z^{-n},\ z \in \mathbb{C}$$

→ Outil d'étude des systèmes linéaires invariants dans le temps NUMERIQUES

$$x(n) \longrightarrow \boxed{h(n) \underset{TZ^{-1}}{\overset{TZ}{\rightleftarrows}} H(z)} \longrightarrow \quad y(n) = x(n) * h(n) \xrightarrow{TZ} Y(z) = \underline{H(z)} X(z)$$

Fonction de transfert

- Étude temporelle : stabilité, rapidité, précision
- Etude fréquentielle :

$$\underline{H(\tilde{f})} = \left[H(z)\right]_{z=e^{j2\pi\tilde{f}}} \ (TZ = TFD) \qquad \tilde{f} = \frac{f}{F_e}$$

(fréquence normalisée)

Réponse en fréquence

Filtrage numérique

- **La transformée en z**

→ <u>Définition :</u> $X(z) = TZ\left[x(n)\right] = \sum_{n=-\infty}^{+\infty} x(n) z^{-n}, z \in \mathbb{C}$

→ <u>Outil d'étude des systèmes linéaires invariants dans le temps NUMERIQUES</u>

$$x(n) \longrightarrow \boxed{h(n) \underset{TZ^{-1}}{\overset{TZ}{\rightleftarrows}} H(z)} \longrightarrow \quad y(n) = x(n) * h(n) \xrightarrow{TZ} Y(z) = \underline{H(z)}X(z)$$

Fonction de transfert

- <u>Étude temporelle</u> : stabilité, rapidité, précision
- <u>Etude fréquentielle</u> :

$$\underline{H(\tilde{f})} = \left[H(z)\right]_{z=e^{j2\pi\tilde{f}}} \; (TZ = TFD) \qquad \tilde{f} = \frac{f}{F_e}$$

(fréquence normalisée)

Réponse en fréquence

→ <u>Principales propriétés :</u>

- <u>Linéarité :</u> $TZ\left[ax(n) + by(n)\right] = aTZ\left[x(n\right] + bTZ\left[y(n\right]$
- <u>Décalage temporel :</u> $TZ\left[x(n - n_0)\right] = z^{-n_0} TZ\left[x(n)\right]$
- <u>Produit de convolution :</u> $TZ\left[x(n) * y(n)\right] = X(z)Y(z)$

Filtrage numérique

- **La transformée en z**

→ <u>Définition :</u> $$X(z) = TZ\left[x(n)\right] = \sum_{n=-\infty}^{+\infty} x(n) z^{-n}, z \in \mathbb{C}$$

→ <u>Outil d'étude des systèmes linéaires invariants dans le temps NUMERIQUES</u>

$$x(n) \longrightarrow \boxed{h(n) \underset{TZ^{-1}}{\overset{TZ}{\rightleftarrows}} H(z)} \longrightarrow \quad y(n) = x(n) * h(n) \xrightarrow{TZ} Y(z) = \underline{H(z)}X(z)$$

Fonction de transfert

- <u>Étude temporelle</u> : stabilité, rapidité, précision
- <u>Etude fréquentielle</u> :

$$\underline{H(\tilde{f})} = \left[H(z)\right]_{z=e^{j2\pi\tilde{f}}} \; (TZ = TFD)$$

Réponse en fréquence

→ <u>Principales propriétés :</u>

- <u>Linéarité :</u> $TZ\left[ax(n) + by(n)\right] = aTZ\left[x(n\right] + bTZ\left[y(n\right]$
- <u>Décalage temporel :</u> $TZ\left[x(n - n_0)\right] = z^{-n_0} TZ\left[x(n)\right]$
- <u>Produit de convolution :</u> $TZ\left[x(n) * y(n)\right] = X(z)Y(z)$

Voir Poly pour plus de détails (domaine d'existence, TZ inverse …)

Filtrage numérique

- **Filtres linéaires invariants dans le temps**

→ Linéarité :

$$\left.\begin{array}{l} x_1(n) \rightarrow \boxed{\textit{Filtre}} \rightarrow y_1(n) \\ x_2(n) \rightarrow \boxed{\textit{Filtre}} \rightarrow y_2(n) \end{array}\right\} \quad x_1(n) + x_2(n) \rightarrow \boxed{\textit{Filtre}} \rightarrow y_1(n) + y_2(n)$$

→ Invariance dans le temps :

$$x(n) \rightarrow \boxed{\textit{Filtre}} \rightarrow y(n) \qquad x(n-n_0) \rightarrow \boxed{\textit{Filtre}} \rightarrow y(n-n_0)$$

Filtrage numérique

- **Réponse impulsionnelle et fonction de transfert**

Impulsion unité
(Impulsion de Kronecker)

$$\delta(n) = 1 \; pour \; n = 0$$
$$= 0 \; pour \; n \neq 0$$

$$\longrightarrow \boxed{\textit{Filtre}} \longrightarrow h(n)$$

Réponse
impulsionnelle

Filtrage numérique

- **Réponse impulsionnelle et fonction de transfert**

Impulsion unité
(Impulsion de Kronecker)

$$\begin{aligned}\delta(n) &= 1 \; pour \; n = 0 \\ &= 0 \; pour \; n \neq 0\end{aligned} \;\longrightarrow\; \boxed{\textit{Filtre}} \;\longrightarrow\; h(n)$$

Réponse impulsionnelle

$$h(n) \xrightarrow{TZ} H(z)$$

Fonction de transfert

Filtrage numérique

- **Réponse impulsionnelle et fonction de transfert**

Impulsion unité
(Impulsion de Kronecker)

$$\delta(n) = 1 \ pour\ n = 0$$
$$= 0 \ pour\ n \neq 0$$

$$\rightarrow \boxed{\textit{Filtre}} \rightarrow h(n)$$

Réponse impulsionnelle

$$h(n) \xrightarrow{TZ} H(z)$$

Fonction de transfert

$$x(n) \rightarrow \boxed{h(n) \underset{TZ^{-1}}{\overset{TZ}{\rightleftarrows}} H(z)} \rightarrow y(n) = x(n) * h(n) = \sum_{n=-\infty}^{+\infty} h(k)x(n-k) \xrightarrow{TZ} Y(z) = H(z)X(z) \quad \Rightarrow H(z) = \frac{Y(z)}{X(z)}$$

Filtrage numérique

- **Réponse impulsionnelle et fonction de transfert**

Impulsion unité
(Impulsion de Kronecker)

$$\delta(n) = 1 \; pour \; n = 0$$
$$= 0 \; pour \; n \neq 0$$

$\longrightarrow \boxed{\textit{Filtre}} \longrightarrow h(n)$ Réponse impulsionnelle

$h(n) \xrightarrow{TZ} H(z)$ Fonction de transfert

$$x(n) \longrightarrow \boxed{h(n) \underset{TZ^{-1}}{\overset{TZ}{\rightleftarrows}} H(z)} \longrightarrow y(n) = x(n) * h(n) = \sum_{n=-\infty}^{+\infty} h(k)x(n-k) \xrightarrow{TZ} Y(z) = H(z)X(z) \quad \Rightarrow H(z) = \frac{Y(z)}{X(z)}$$

- **Conditions de réalisabilité (données sur la réponse impulsionnelle)**

$\rightarrow$ <u>Causalité</u> $\quad h(n) = 0 \text{ pour } n < 0$

$\rightarrow$ <u>Stabilité (entrée bornée => sortie bornée)</u> $\quad \sum_{n=-\infty}^{+\infty} |h(n)| < \infty$

Filtrage numérique

$$x(n) \longrightarrow \boxed{h(n) \underset{TZ^{-1}}{\overset{TZ}{\rightleftarrows}} H(z)} \longrightarrow y(n) = x(n) * h(n) = \sum_{n=-\infty}^{+\infty} h(k)x(n-k) \xrightarrow{TZ} Y(z) = H(z)X(z) \quad \Rightarrow H(z) = \frac{Y(z)}{X(z)}$$

- **Réponse en fréquence et temps de propagation de groupe**

→ <u>Réponse en fréquence (ou réponse harmonique) :</u>

$$H(\tilde{f}) = [H(z)]_{z=e^{j2\pi\tilde{f}}} \; (TZ = TFD)$$

$$\tilde{f} = \frac{f}{F_e}$$ (fréquence normalisée)

Remarque : $\left|H(\tilde{f})\right|^2 = \left[H(z)\,H\left(z^{-1}\right)\right]_{z=e^{j2\pi\tilde{f}}}$

→ <u>Temps de propagation de groupe (TPG) :</u>

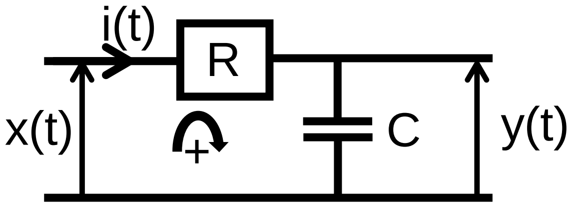
$$TPG(\tilde{f}) = -\frac{1}{2\pi}\frac{d\Phi_H(\tilde{f})}{d\tilde{f}}$$

Filtrage numérique

- **Filtres numériques RATIONNELS**

→ <u>Définis par analogie avec les filtres analogiques :</u>

Exemple :



TL

$$y(t) = x(t) - RC\frac{dy(t)}{dt}.$$

Équation différentielle en temporel

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

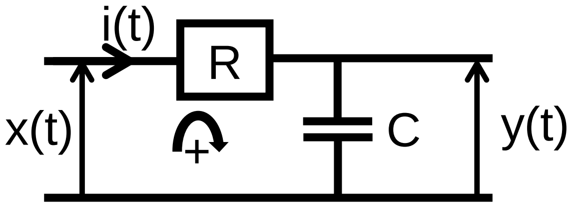
Fonction de transfert rationnelle en p

Filtrage numérique

- **Filtres numériques RATIONNELS**

→ <u>Définis par analogie avec les filtres analogiques :</u>

Exemple :



$$y(t) = x(t) - RC\frac{dy(t)}{dt}.$$

Équation différentielle en temporel

TL

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

Fonction de transfert rationnelle en p

$$\boxed{H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}}}$$

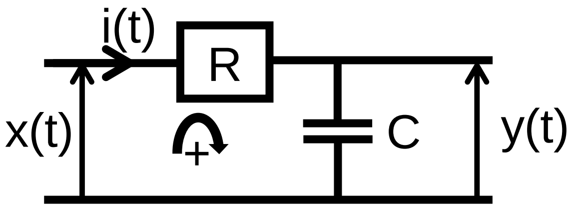
Fonction de transfert RATIONNELLE en z

Filtrage numérique

- **Filtres numériques RATIONNELS**

→ <u>Définis par analogie avec les filtres analogiques :</u>

Exemple :



TL

$$y(t) = x(t) - RC\frac{dy(t)}{dt}$$ Équation différentielle en temporel

$$H(p) = \frac{Y(p)}{X(p)} = \frac{1}{1 + RCp}$$ Fonction de transfert rationnelle en p

$$\boxed{H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}}}$$ Fonction de transfert RATIONNELLE en z

TZ^{-1}

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

(Remarque : on impose $a_0 = 1$)

0. Slide de connexion

1 **Allez sur wooclap.com**

2 **Entrez le code d'événement dans le bandeau supérieur**

Code d'événement

SIGSEQ3

https://app.wooclap.com/SIGSEQ3

Copier le lien de participation

QUESTION 1

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Soit un filtre numérique défini par la relation suivante entre son entrée à l'instant nTe, notée x(n), et sa sortie à l'instant nTe, notée y(n) :

y(n)=0.1y(n-1)+0.4y(n-2)+x(n)-0.5x(n-1)+0.2x(n-2).

Sa fonction de transfert H(z) est donnée par :

1. $$H(z) = \frac{1 - 0{,}1z^{-1} - 0{,}4z^{-2}}{1 - 0{,}5z^{-1} + 0{,}2z^{-2}}$$

2. $$H(z) = \frac{1 - 0{,}5z^{-1} + 0{,}2z^{-2}}{1 - 0{,}1z^{-1} - 0{,}4z^{-2}}$$

QUESTION 2

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Soit un filtre numérique défini par la fonction de transfert suivante :

$$H(z) = \frac{Y(z)}{X(z)} = 1 - 0.5z^{-1} + 0.2z^{-2}$$

L'équation récurrente qui lie son entrée $x(n) = TZ^{-1}[X(z)]$ à sa sortie $y(n) = TZ^{-1}[Y(z)]$ est donnée par :

1. $y(n) = 1 - 0.5x(n-1) + 0.2x(n-2)$

2. $y(n) = x(n) - 0.5x(n-1) + 0.2x(n-2)$

3. $y(n) = x(n) + 0.5y(n-1) - 0.2y(n-2)$

Filtrage numérique

- **Filtres numériques RATIONNELS**

→ <u>Définition :</u>

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}}$$ Fonction de transfert RATIONNELLE en z

TZ^{-1}

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

(Remarque : on impose $a_0 = 1$)

Filtrage numérique

- **Filtres numériques RATIONNELS**

→ <u>Définition :</u>

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}}$$ Fonction de transfert RATIONNELLE en z

TZ^{-1}

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

(Remarque : on impose $a_0 = 1$)

Définis par deux jeux de coefficients

$$(a_0 = 1)y(n) = -\sum_{k=1}^{M-1} a_k y(n-k) + \sum_{k=0}^{N-1} b_k x(n-k)$$

Filtrage numérique

- **Filtres numériques RATIONNELS**

→ Définition :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}}$$ Fonction de transfert RATIONNELLE en z

TZ^{-1}

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

(Remarque : on impose $a_0 = 1$)

Définis par deux jeux de coefficients

$$(a_0 = 1) y(n) = -\sum_{k=1}^{M-1} a_k y(n-k) + \sum_{k=0}^{N-1} b_k x(n-k)$$

```
%filtrage sous Matlab
y=filter(B,A,x);
```

y : Signal filtré

x : Signal à filtrer

Définition du filtre à utiliser (deux tableaux de coefficients)

Filtrage numérique

- **Filtres numériques RATIONNELS**

→ Définition :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}}$$ Fonction de transfert RATIONNELLE en z

TZ^{-1}

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

(Remarque : on impose $a_0 = 1$)

Définis par deux jeux de coefficients

$$(a_0 = 1) y(n) = -\sum_{k=1}^{M-1} a_k y(n-k) + \sum_{k=0}^{N-1} b_k x(n-k)$$

Système bouclé

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

```
%filtrage sous Matlab
y=filter(B,A,x);
```

y : Signal filtré

x : Signal à filtrer

Définition du filtre à utiliser (deux tableaux de coefficients)

Filtrage numérique

- **Filtres numériques RATIONNELS**

→ Définition :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}}$$

Fonction de transfert RATIONNELLE en z

TZ^{-1}

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

Équation récurrente en temporel

(Remarque : on impose $a_0 = 1$)

Définis par deux jeux de coefficients

$$(a_0 = 1)\, y(n) = -\sum_{k=1}^{M-1} a_k y(n-k) + \sum_{k=0}^{N-1} b_k x(n-k)$$

%filtrage sous Matlab

y=filter(B,A,x);

y : Signal filtré

x : Signal à filtrer

Définition du filtre à utiliser (deux tableaux de coefficients)

Système bouclé

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

$$\Rightarrow h(n) = -\sum_{k=1}^{M-1} a_k h(n-k), \text{ pour } n \geq N$$

Filtres de type RII = à Réponse Impulsionnelle Infinie

Filtrage numérique

- **Stabilité des filtres numériques RATIONNELS**

$\rightarrow$ Condition sur les pôles de la fonction de transfert (démo dans le poly) :

$$H(z) = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}} = \sum_{k=0}^{M-1} \frac{A_k}{1 - p_k z^{-1}}$$

(hypothèse : N<M)

TZ^{-1}

$$h(n) = \sum_{k=0}^{M-1} A_k p_k^n u(n)$$

(solution causale[(1)])

Condition de stabilité des filtres numériques rationnels

$$\boxed{\sum_{n=-\infty}^{+\infty} |h(n)| < \infty \quad \text{si } |p_k| < 1 \ \forall k}$$

Filtrage numérique

• Stabilité des filtres numériques RATIONNELS

→ Condition sur les pôles de la fonction de transfert (démo dans le poly) :

$$H(z) = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{\sum_{k=0}^{M-1} a_k z^{-k}} = \sum_{k=0}^{M-1} \frac{A_k}{1 - p_k z^{-1}}$$

TZ^{-1}

(hypothèse : N<M)

$$h(n) = \sum_{k=0}^{M-1} A_k p_k^n u(n)$$ (solution causale[1])

Condition de stabilité des filtres numériques rationnels

$$\boxed{\sum_{n=-\infty}^{+\infty} |h(n)| < \infty \quad \text{si } |p_k| < 1 \ \forall k}$$

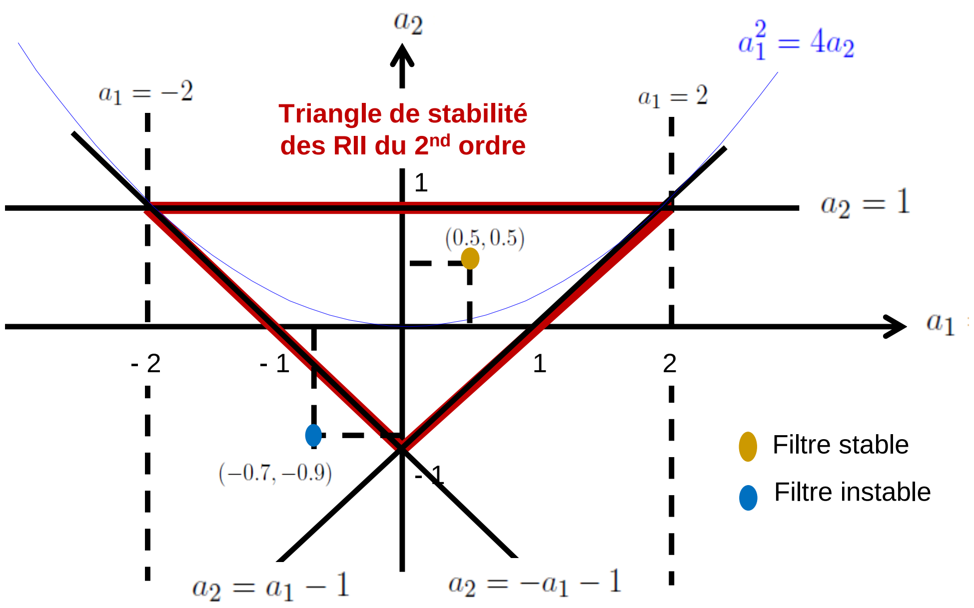
→ Condition sur les coefficients (a_k, b_k) (voir exercice 4 dans le poly) :

- Filtres du 1ier ordre $H(z) = \dfrac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}}$

$\boxed{|a_1| < 1}$ **Condition de stabilité des RII du 1ier ordre**

- Filtres du 2nd ordre $H(z) = \dfrac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}$

$\boxed{(a_1, a_2) \in \text{ triangle de stabilité}}$ **Condition de stabilité des RII du 2nd ordre**



[1]) La TZ inverse n'est pas unique. Elle sera différente selon le contour choisi dans le domaine de convergence pour la calculer : voir exercices poly sur la TZ.

Filtrage numérique

- **Filtres numériques RATIONNELS à réponse impulsionnelle finie (type RIF)**

→ <u>Définition :</u>

$$y(n) = \cancel{\boxed{-\sum_{k=1}^{M-1} a_k y(n-k)}} + \sum_{k=0}^{N-1} b_k x(n-k) \quad \xrightarrow{TZ} \quad H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{N-1} b_k z^{-k}$$

Équation récurrente en temporel <u>sans boucle de réaction</u>

Filtrage numérique

- **Filtres numériques RATIONNELS à réponse impulsionnelle finie (type RIF)**

→ <u>Définition :</u>

$$y(n) = \cancel{\boxed{-\sum_{k=1}^{M-1} a_k y(n-k)}} + \sum_{k=0}^{N-1} b_k x(n-k) \quad \xrightarrow{TZ} \quad H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{N-1} b_k z^{-k}$$

Équation récurrente en temporel <u>sans boucle de réaction</u>

Définis par un seul jeu de coefficients

```
%filtrage sous Matlab
y=filter(B,1,x);
```

Définition du filtre (RIF) à utiliser

y : Signal filtré

x : Signal à filtrer

```
y=conv(x,B,'same');
```

'same' returns the central part of the convolution that is the same size as x.

Filtrage numérique

- **Filtres numériques RATIONNELS à réponse impulsionnelle finie (type RIF)**

→ <u>Définition :</u>

$$y(n) = \cancel{\boxed{-\sum_{k=1}^{M-1} a_k y(n-k)}} + \sum_{k=0}^{N-1} b_k x(n-k) \quad \xrightarrow{TZ} \quad H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{N-1} b_k z^{-k}$$

Équation récurrente en temporel <u>sans boucle de réaction</u>

Définis par un seul jeu de coefficients

```
%filtrage sous Matlab
y=filter(B,1,x);
```

y : Signal filtré

Définition du filtre (RIF) à utiliser

x : Signal à filtrer

```
y=conv(x,B,'same');
```

'same' returns the central part of the convolution that is the same size as x.

Système non bouclé

$$y(n) = \sum_{k=0}^{N-1} b_k x(n-k) = \sum_{k=0}^{N-1} h(k) x(n-k)$$

- Filtres non récursifs
- Inconditionnellement stables
- Remarque : $b_k = h(k)$ mais pour un filtre rendu causal (voir plus loin)

QUESTION 3

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Soit un filtre numérique défini par la relation suivante entre son entrée à l'instant nTe, notée x(n), et sa sortie à l'instant nTe, notée y(n) :
$y(n)=0.3y(n-1)+0.5x(n)-0.4x(n-1)$.
Ce filtre est de type :

1. RIF
2. RII
3. Pas assez d'éléments pour répondre à la question

QUESTION 4

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Soit un filtre numérique défini par la relation suivante entre son entrée à l'instant nTe, notée x(n), et sa sortie à l'instant nTe, notée y(n) :

$y(n)=x(n)-0.5x(n-1)+0.2x(n-2)$

Ce filtre est réalisable :

1. Vrai
2. Faux

QUESTION 5

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Soit un filtre numérique défini par la relation suivante entre son entrée à l'instant nTe, notée x(n), et sa sortie à l'instant nTe, notée y(n) :
$y(n)=0.3y(n-1)+0.5x(n)-0.4x(n-1)$.
On pourra implanter ce filtre sous Matlab en utilisant la ligne de code :

1. `y=filter(0.3,[0.5 -0.4],x)`

2. `x=filter(0.3,[0.5 -0.4],y)`

3. `y=filter([0.5 -0.4],[1 -0.3],x)`

4. `y=filter([1 -0.3],[0.5 -0.4],x)`

QUESTION 6

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Soit un filtre numérique défini par la relation suivante entre son entrée à l'instant nTe, notée x(n), et sa sortie à l'instant nTe, notée y(n) :
$y(n)=0.1x(n)-0.4x(n-1)+0.6x(n-2)$.
On pourra implanter ce filtre sous Matlab en utilisant la ligne de code :

1. `y=filter(1,[0.1 -0.4 0.6],x)`

2. `y=filter(0,[0.1 -0.4 0.6],x)`

3. `y=filter([0.1 -0.4 0.6],1,x)`

4. `y=filter([0.1 -0.4 0.6],0,x)`

Filtrage numérique

- **Synthèse et implantation des filtres numériques rationnels**

Spécifications à respecter :

GABARIT

Filtrage numérique

- **Synthèse et implantation des filtres numériques rationnels**

Spécifications à respecter :

GABARIT

? COEFFICIENTS ?

$$\{a_k\}_{k=0,\ldots,M-1}$$

$$\{b_k\}_{k=0,\ldots,N-1}$$

Définissant un filtre respectant le gabarit fixé

Filtrage numérique

- **Synthèse et implantation des filtres numériques rationnels**

Spécifications à respecter :

GABARIT

↓

SYNTHESE

↓

COEFFICIENTS

$$\{a_k\}_{k=0,\ldots,M-1}$$

$$\{b_k\}_{k=0,\ldots,N-1}$$

Définissant un filtre respectant le gabarit fixé

Filtrage numérique

- **Synthèse et implantation des filtres numériques rationnels**

Spécifications à respecter :

GABARIT

↓

SYNTHESE

↓

COEFFICIENTS

$$\{a_k\}_{k=0,\ldots,M-1}$$

$$\{b_k\}_{k=0,\ldots,N-1}$$

Définissant un filtre respectant le gabarit fixé

IMPLANTATION sous Matlab

Tableaux de coefficients définissant le filtre à utiliser

y=filter(B,A,x);

Signal filtré

Signal à filtrer

Remarque : A=[1] pour les filtres RIF

IMPLANTATION en temps réel :

Filtre RII

$$x(n) \rightarrow \boxed{\begin{array}{c}\{a_k\}_{k=0,\ldots,M-1}\\ \{b_k\}_{k=0,\ldots,N-1}\end{array}} \rightarrow y(n) = -\sum_{k=1}^{M-1} a_k y(n-k) + \sum_{k=0}^{N-1} b_k x(n-k)$$

Filtre RIF

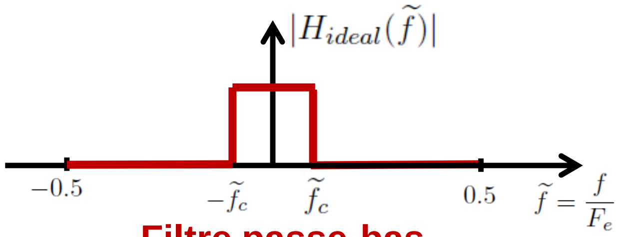
$$x(n) \rightarrow \boxed{\{b_k\}_{k=0,\ldots,N-1}} \rightarrow y(n) = \sum_{k=0}^{N-1} b_k x(n-k)$$

À réaliser en T_e secondes (période d'échantillonnage)

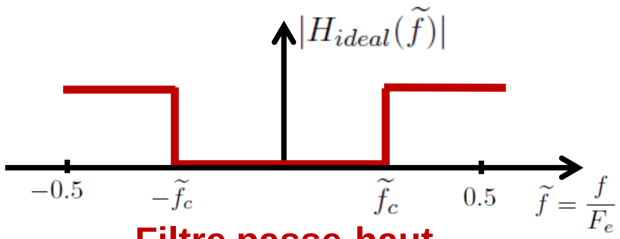
Filtrage numérique

- **Synthèse et implantation des filtres numériques rationnels**

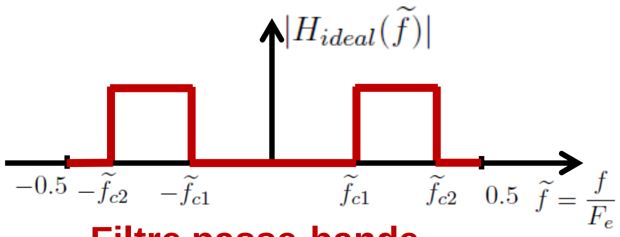
→ Réponses en fréquence cibles idéales



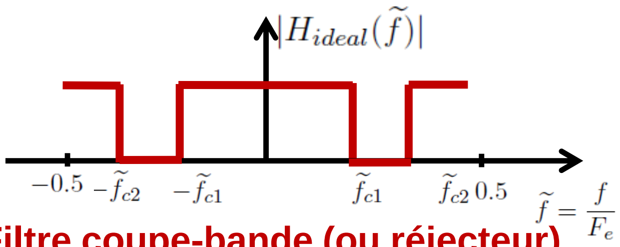
Filtre passe-bas



Filtre passe-haut



Filtre passe-bande



Filtre coupe-bande (ou réjecteur)

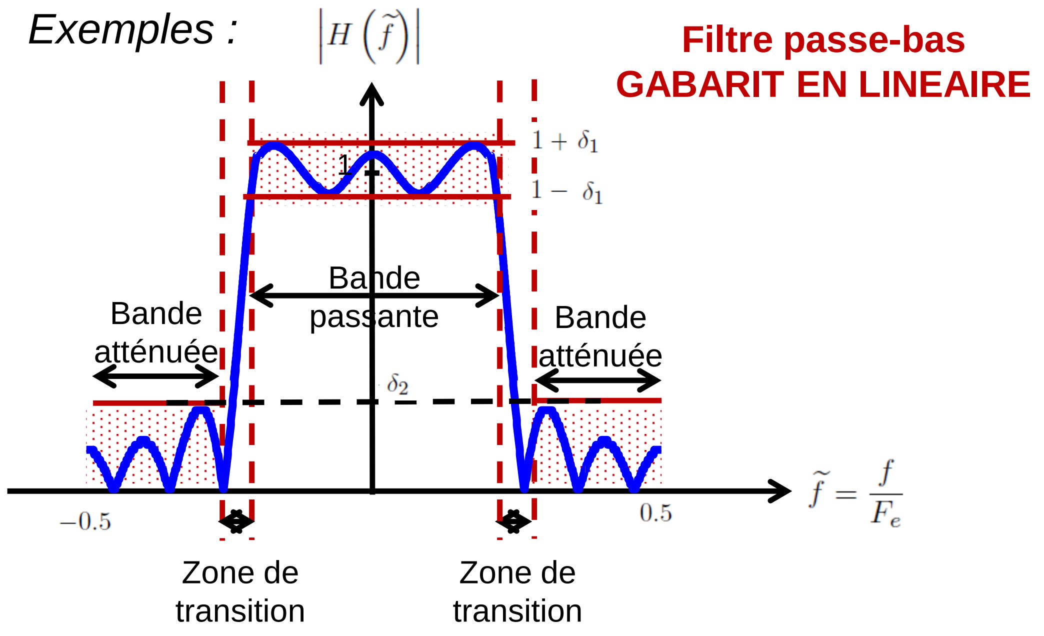
Filtrage numérique

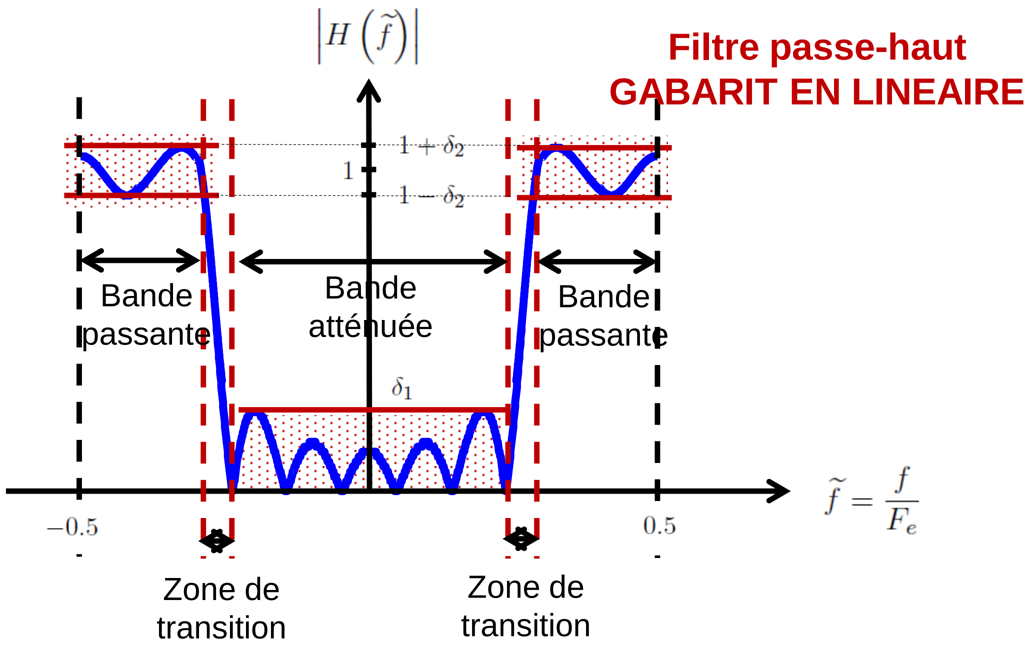
- **Synthèse et implantation des filtres numériques rationnels**

→ Notion de gabarit à respecter

- Sur le module de la réponse en fréquence

Exemples :





- Sur la phase de la réponse en fréquence : filtres de phase ou passe-tout

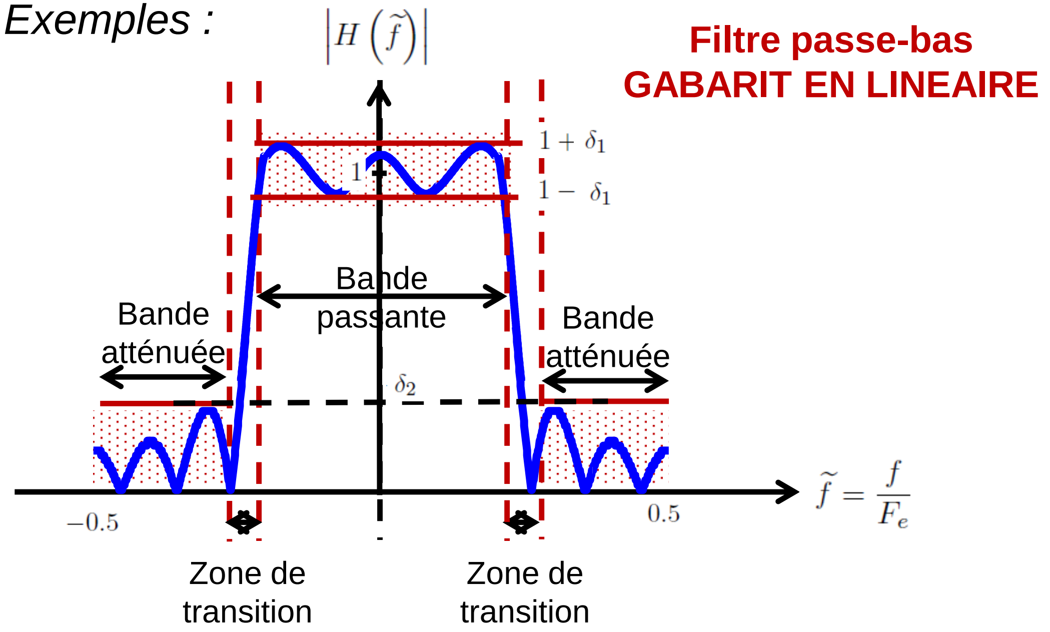
Filtrage numérique

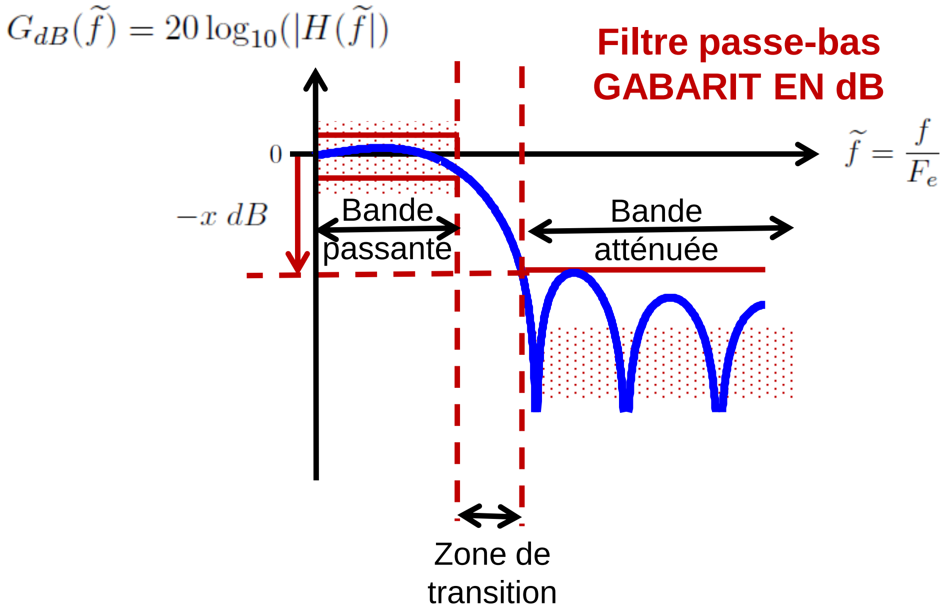
- **Synthèse et implantation des filtres numériques rationnels**

→ Notion de gabarit à respecter

- Sur le module de la réponse en fréquence

Exemples :



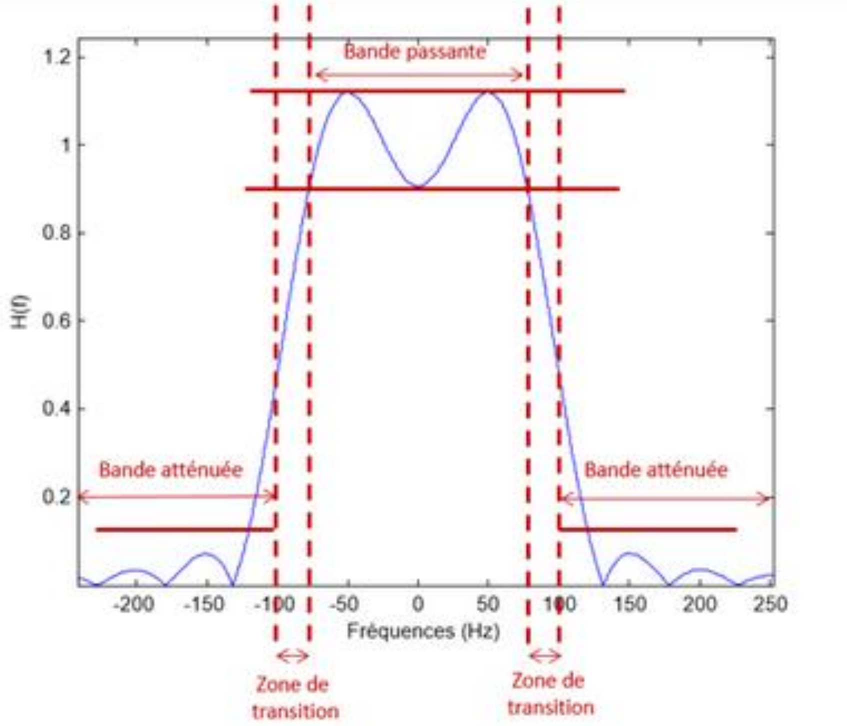


- Sur la phase de la réponse en fréquence : filtres de phase ou passe-tout

QUESTION 7

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

La figure ci-dessous trace, en bleu, la réponse en fréquence d'un filtre numérique et en rouge le gabarit à respecter. Le filtre numérique respecte le gabarit fixé.



1 Vrai

2 Faux

Filtrage numérique

- **Synthèse et implantation des filtres numériques rationnels**

Spécifications à respecter :

GABARIT

↓

SYNTHESE

Synthèse de **RIF**

Synthèse de **RII**

Deux méthodes différentes

↓

COEFFICIENTS

$\{a_k\}_{k=0,\dots,M-1}$

$\{b_k\}_{k=0,\dots,N-1}$

Le filtre qui respecte le gabarit n'est pas unique

← **d'un filtre** respectant le gabarit fixé →

IMPLANTATION sous Matlab

Tableaux de coefficients définissant le filtre à utiliser

y=filter(B,A,x);

Signal filtré

Signal à filtrer

Remarque : A=[1] pour les filtres RIF

IMPLANTATION en temps réel :

Filtre RII

$$x(n) \rightarrow \boxed{\begin{array}{c}\{a_k\}_{k=0,\dots,M-1}\\ \{b_k\}_{k=0,\dots,N-1}\end{array}} \rightarrow y(n) = -\sum_{k=1}^{M-1} a_k y(n-k) + \sum_{k=0}^{N-1} b_k x(n-k)$$

Filtre RIF

$$x(n) \rightarrow \boxed{\{b_k\}_{k=0,\dots,N-1}} \rightarrow y(n) = \sum_{k=0}^{N-1} b_k x(n-k)$$

À réaliser en T_e secondes (période d'échantillonnage)

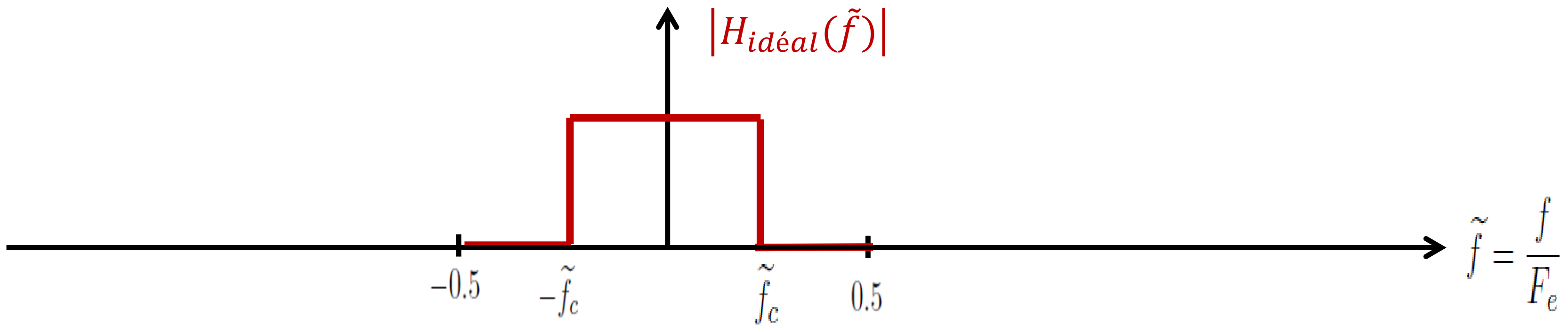
Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

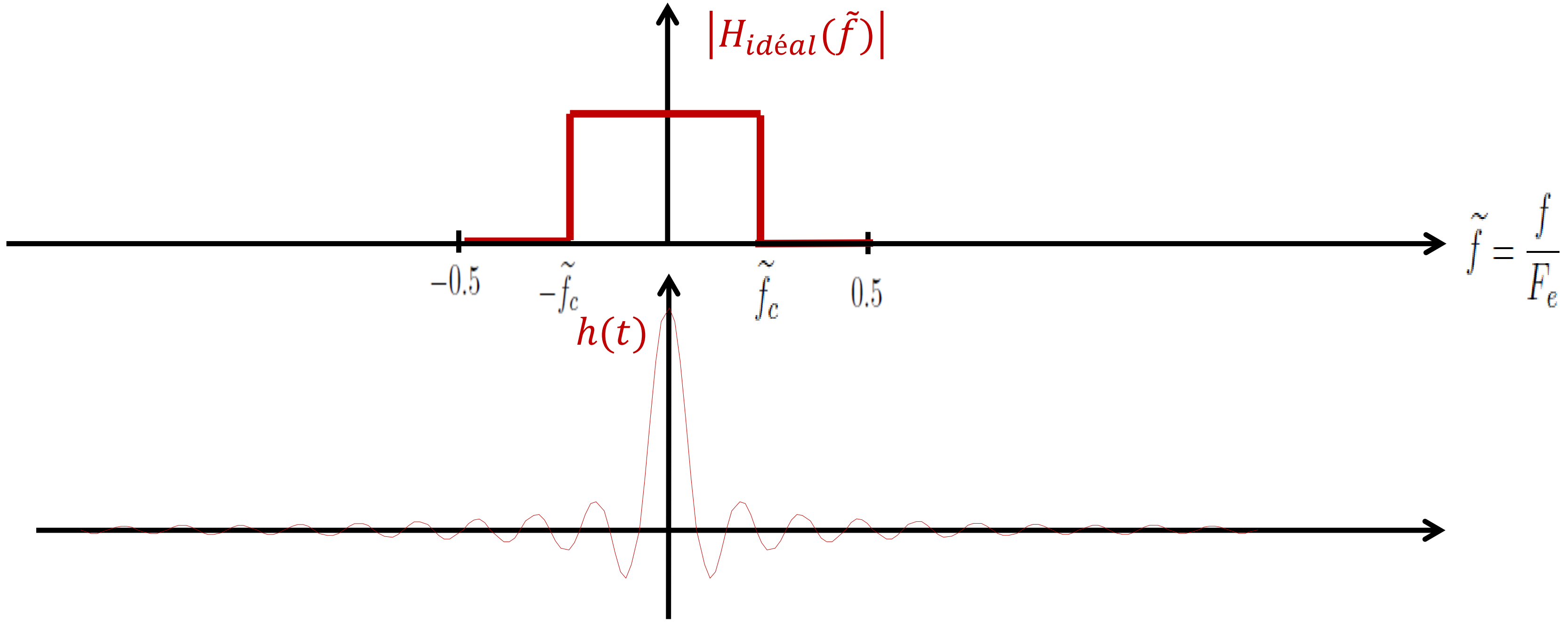
→ Exemple : synthèse d'un filtre passe-bas numérique



Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

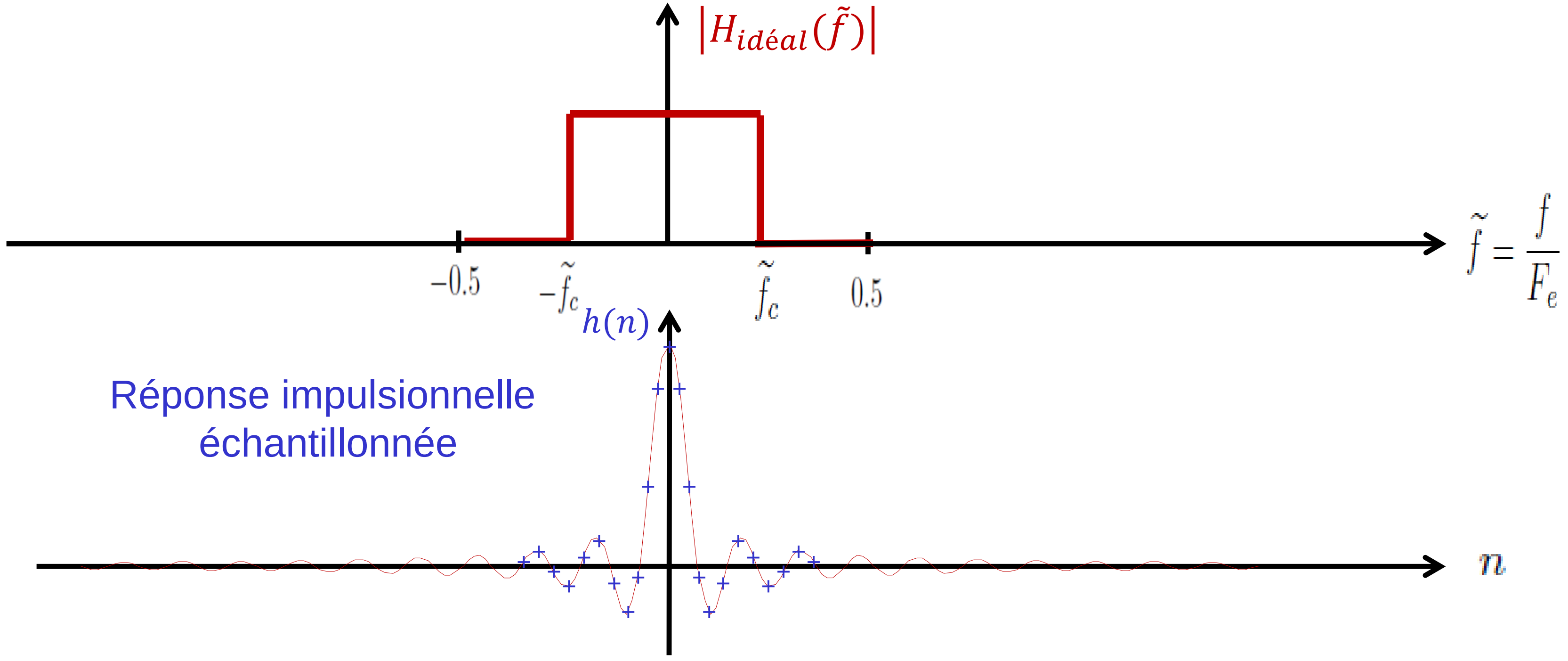
  → Exemple : synthèse d'un filtre passe-bas numérique



Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

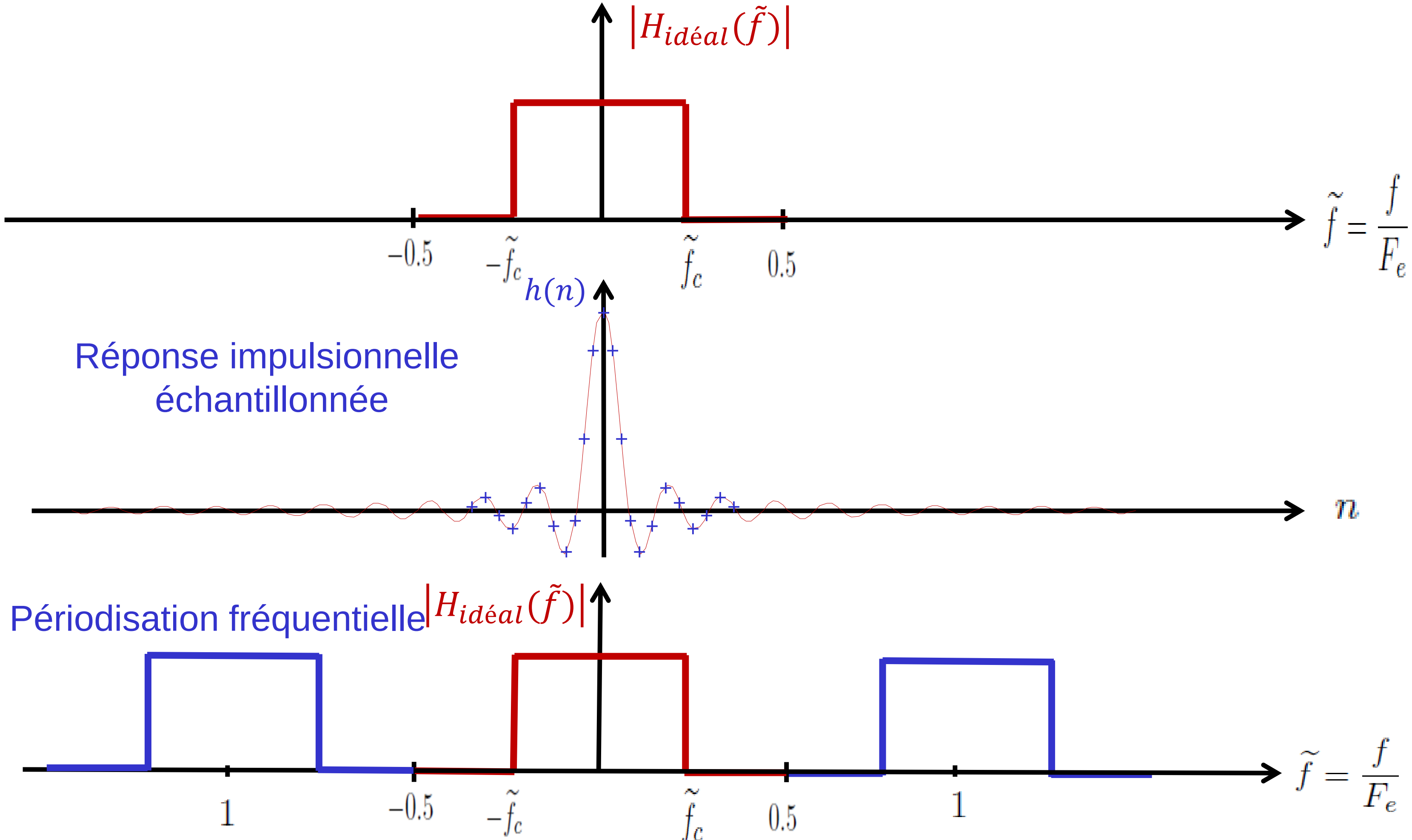
→ Exemple : synthèse d'un filtre passe-bas numérique



Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

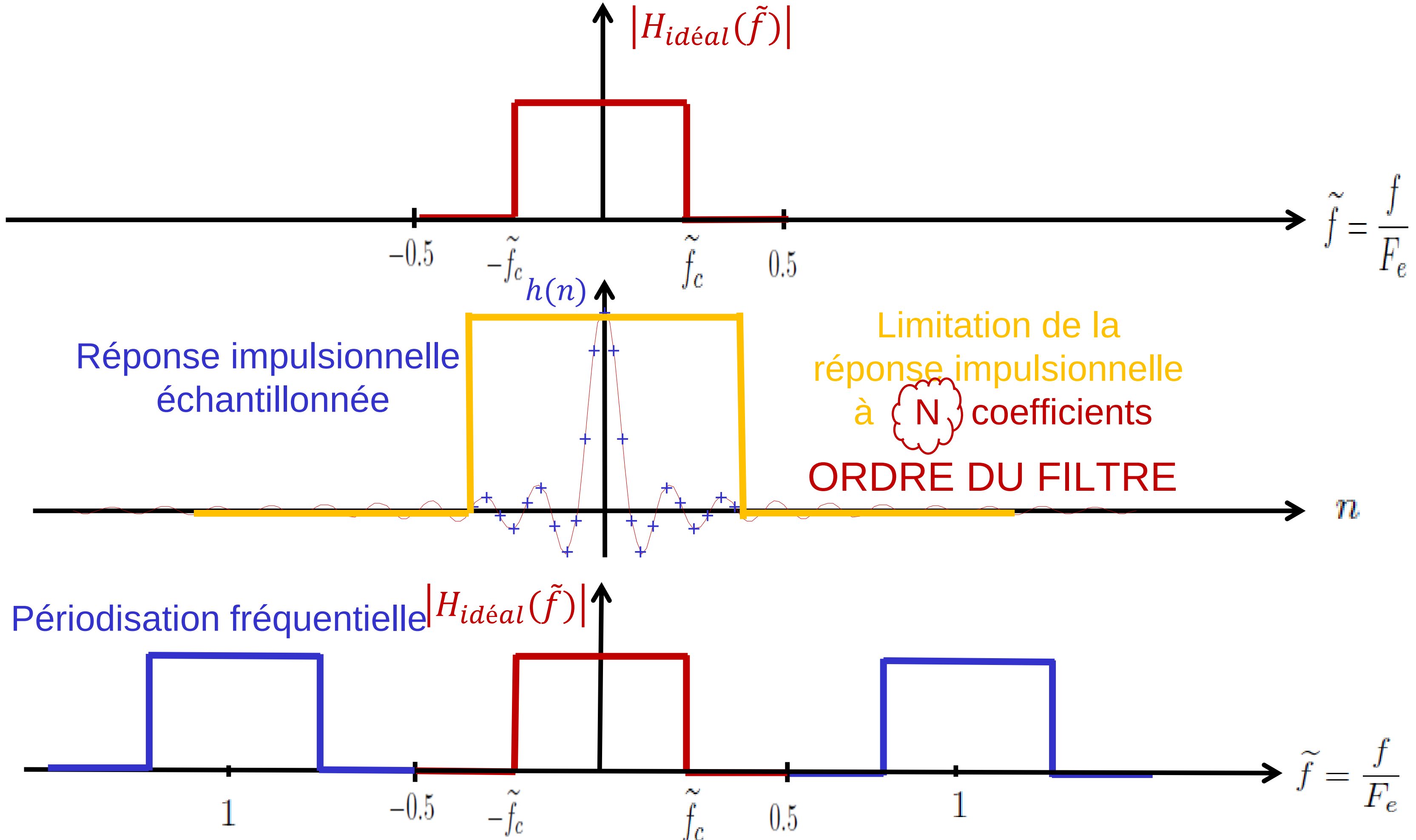
→ Exemple : synthèse d'un filtre passe-bas numérique



Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

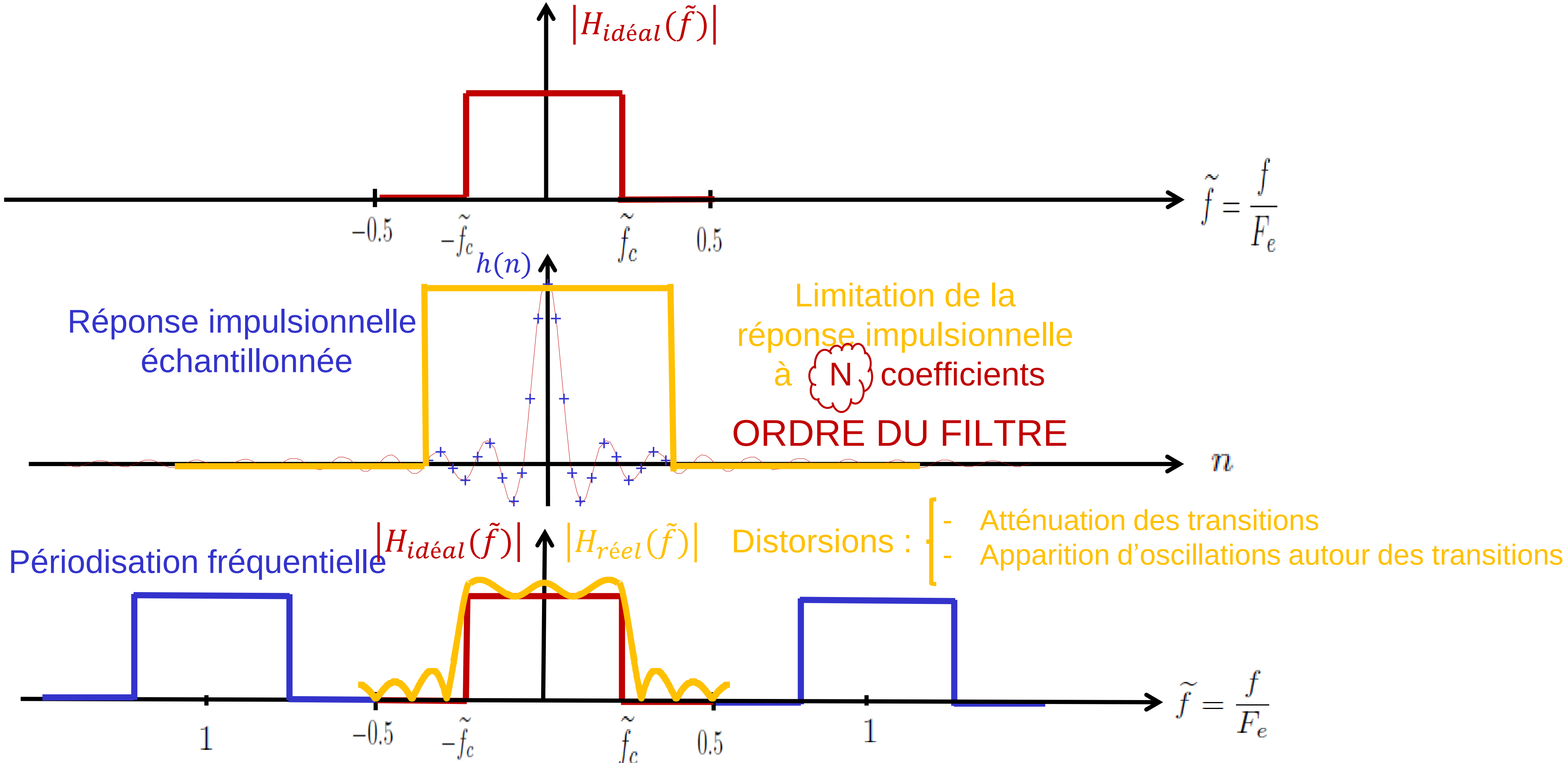
  → Exemple : synthèse d'un filtre passe-bas numérique



Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

→ Exemple : synthèse d'un filtre passe-bas numérique

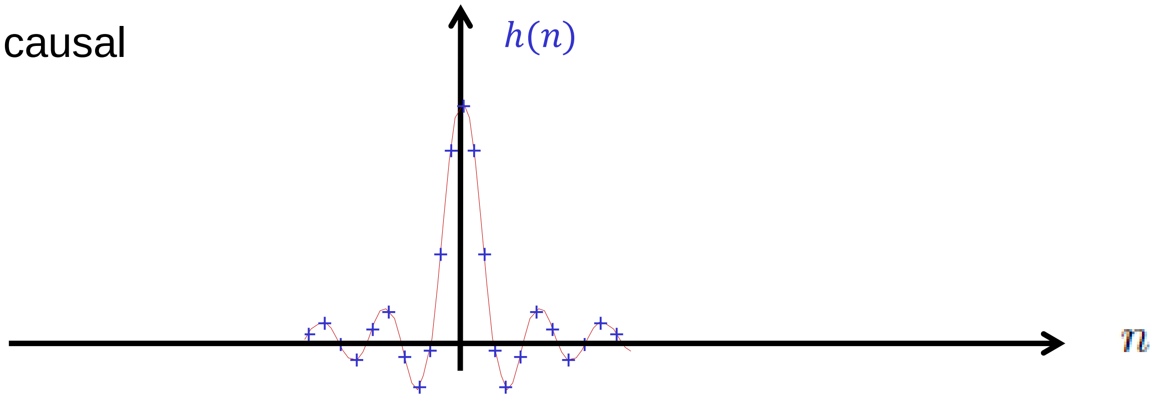


Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

  → Exemple : synthèse d'un filtre passe-bas numérique

  - Filtres non causal

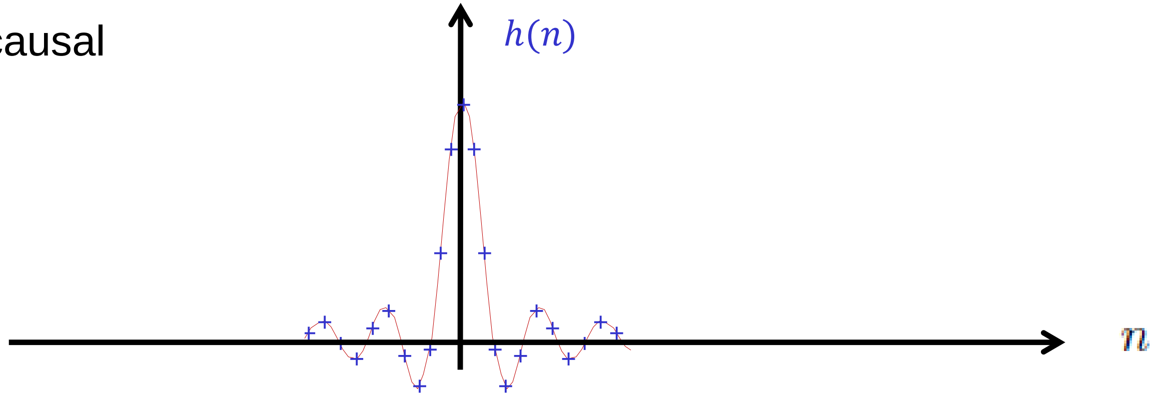


Filtrage numérique

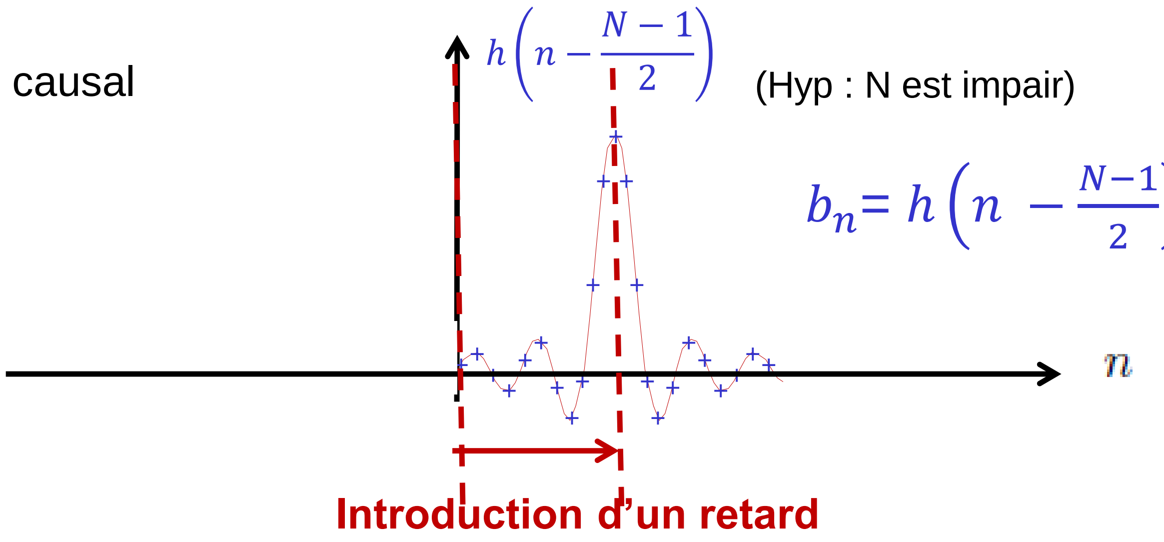
- **Synthèse des filtres numériques de type RIF**

  → Exemple : synthèse d'un filtre passe-bas numérique

  - Filtre non causal



  - Filtre rendu causal



(Hyp : N est impair)

$$b_n = h\left(n - \frac{N-1}{2}\right)$$

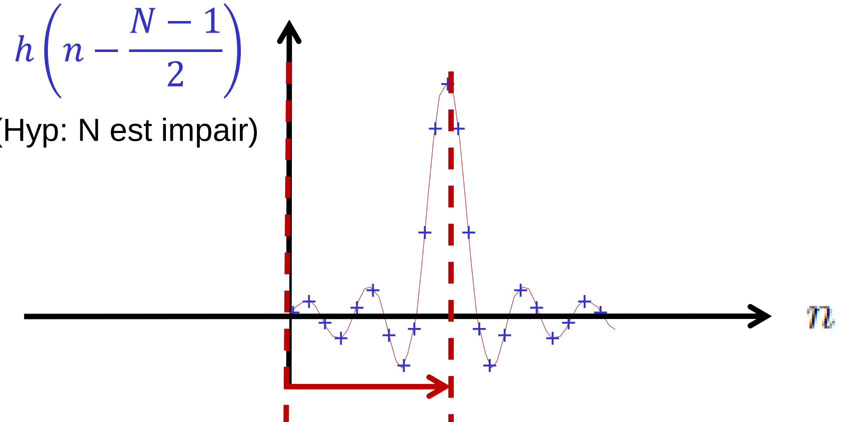
Filtrage numérique

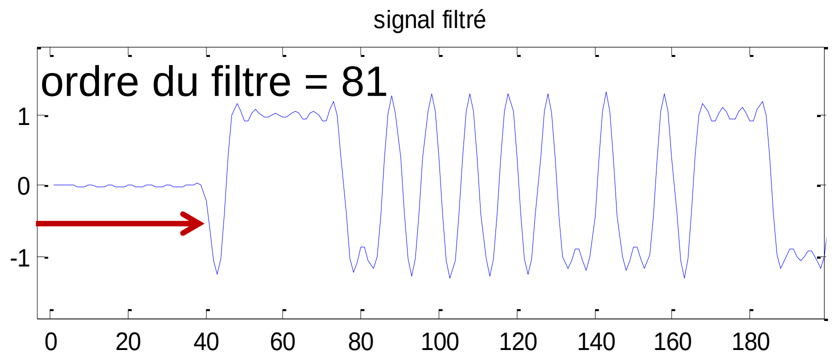
- **Synthèse des filtres numériques de type RIF**

  → Exemple : synthèse d’un filtre passe-bas numérique

$h\left(n-\frac{N-1}{2}\right)$

(Hyp: N est impair)



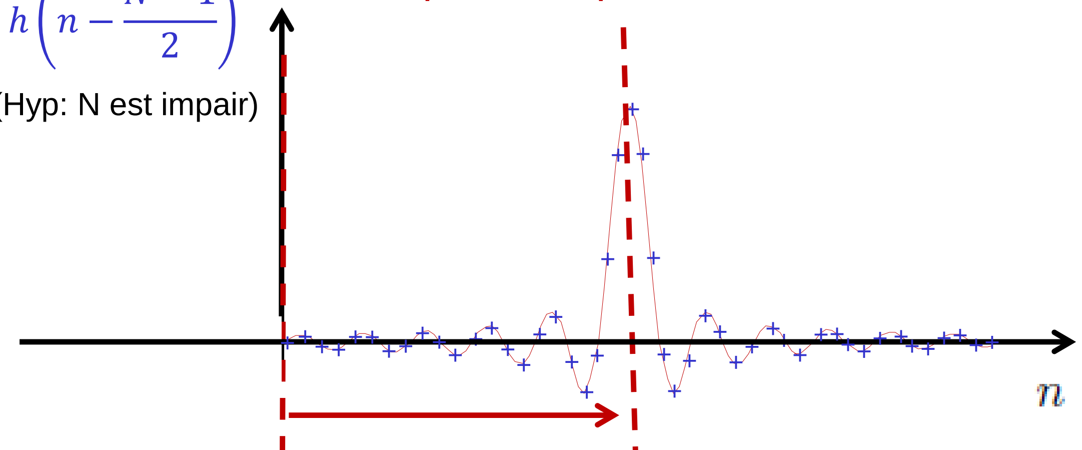


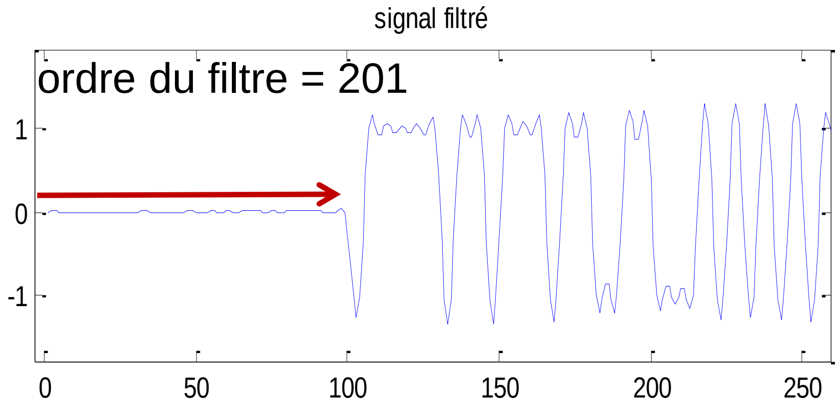
Introduction d’un retard sur la réponse impulsionnelle

Introduction d’un retard sur le signal filtré

$h\left(n-\frac{N-1}{2}\right)$

(Hyp: N est impair)





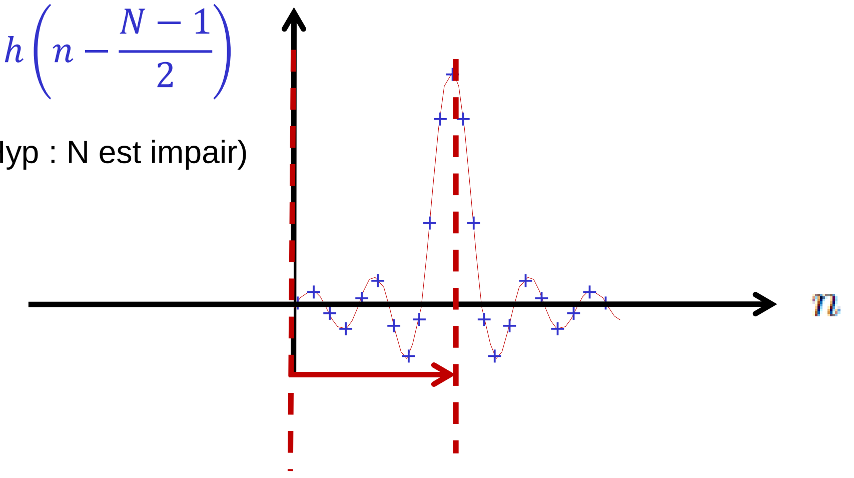
Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

  → Exemple : synthèse d'un filtre passe-bas numérique

$h\left(n - \frac{N-1}{2}\right)$

(Hyp : N est impair)



$$Retard = \frac{Ordre - 1}{2} \times T_e$$

Retard ⇒ Constante ajoutée au TPG :

$$\left|H_{causal}(\tilde{f})\right| = \left|H_{non\ causal}(\tilde{f})\right|$$

$$\underbrace{TPG_{filtre\ causal}\left(\tilde{f}\right) = \underbrace{TPG_{filtre\ non\ causal}\left(\tilde{f}\right)}_{\text{Constant si } h(n) \text{ pair ou impair (voir poly)}} + \frac{N-1}{2}}_{}$$

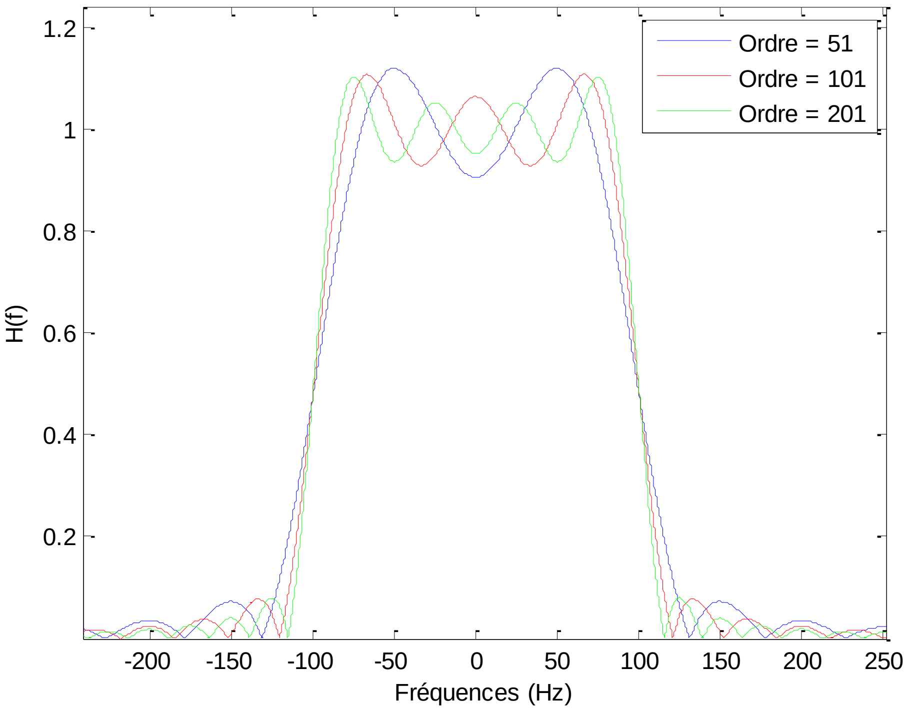
Le TPG d'un filtre RIF est constant si sa réponse impulsionnelle est paire ou impaire

Filtrage numérique

- **Synthèse des filtres numériques de type RIF**

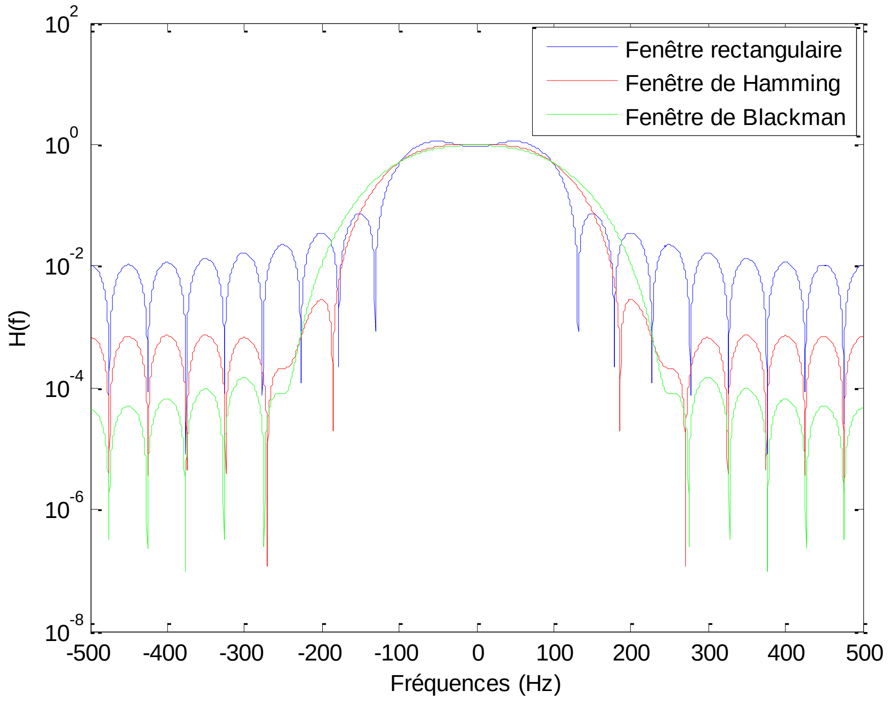
→ Paramètres permettant de respecter le gabarit : ORDRE et FENETRE DE TRONCATURE

→ Influence de l'ordre (fenêtre rectangulaire)



Atténuation plus ou moins importantes des transitions

→ Influence de la fenêtre de troncature (ordre fixé à 21)

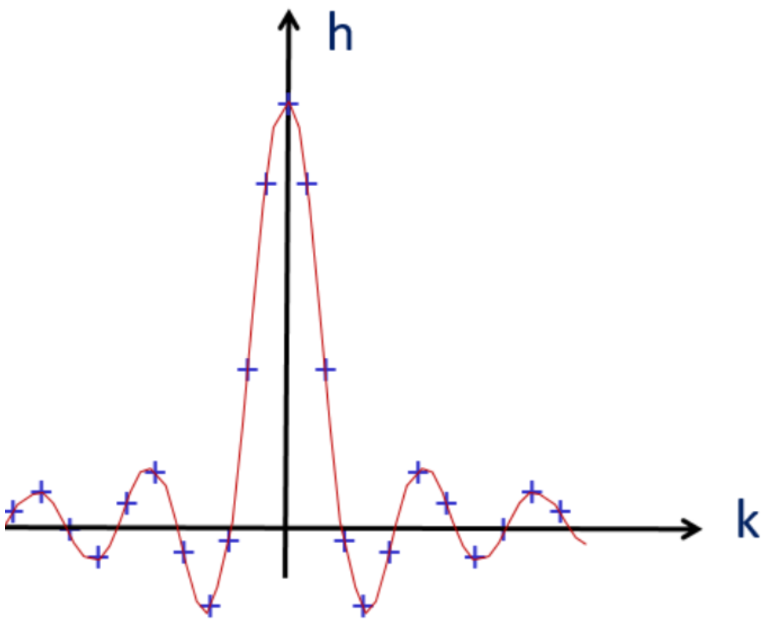


Ondulations plus ou moins importantes autour des transitions

QUESTION 8

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Soit le filtre défini par la réponse impulsionnelle échantillonnée h donnée par la figure. Ce filtre est d'ordre :

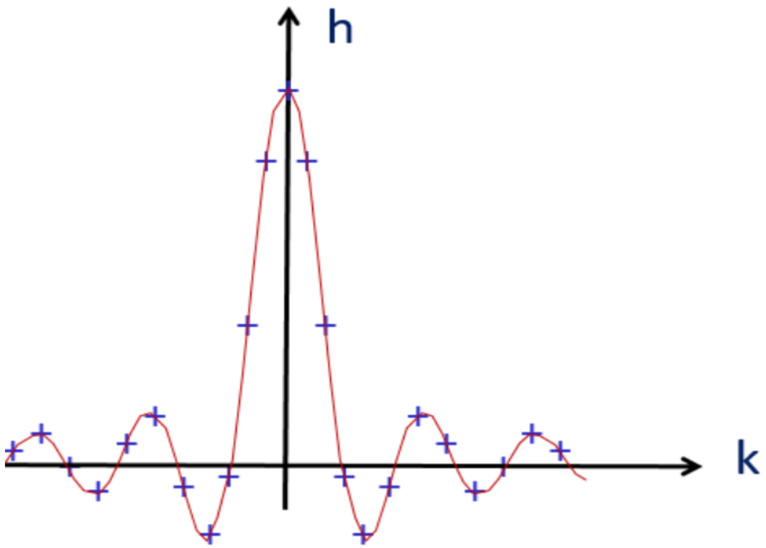


1. 5
2. 12
3. 23

QUESTION 8

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Le filtre de réponse impulsionnelle h donnée par la figure est réalisable :



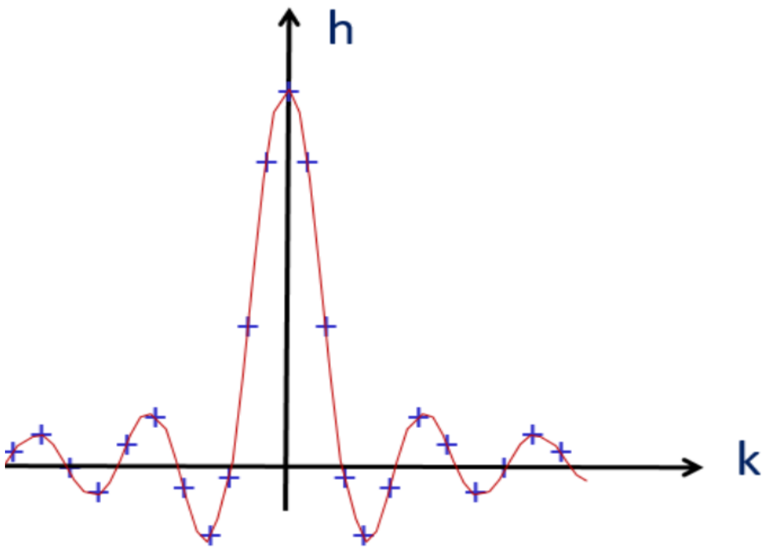
1. Vrai
2. Faux

QUESTIONs 9 et 10

Allez sur **wooclap.com** et utilisez le code **SIGSEQ3**

Soit un filtre dont la réponse impulsionnelle, échantillonnée à Te=1/Fe (points bleu), est donnée par la figure. Donner, en nombre d'échantillons, le retard que va introduire ce filtre.

Soit une fréquence d'échantillonnage Fe de 500 Hz et un filtre dont la réponse impulsionnelle, échantillonnée à Te=1/Fe (points bleu), est donnée par la figure. Donner, en ms, le retard que va introduire ce filtre.



Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Utilisation des bibliothèques
de modèles analogiques :
Butterworth, Tchebycheff, Cauer, Bessel…

Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Spécifications à respecter :

$\mathbf{H(\tilde{f})}$

Utilisation des bibliothèques
de modèles analogiques :
Butterworth, Tchebycheff, Cauer, Bessel…

Fonction de transfert :

$\mathbf{H(z)}$

Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Spécifications à respecter :
$H(\tilde{f})$

Spécifications à respecter :
$H(f)$

↓

Utilisation des bibliothèques
de modèles analogiques :
Butterworth, Tchebycheff, Cauer, Bessel…

↓

Fonction de transfert :
$H(p)$

Fonction de transfert :
$H(z)$

Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Spécifications à respecter :
$H(\tilde{f})$

↓ ?

Spécifications à respecter :
$H(f)$

↓

Utilisation des bibliothèques
de modèles analogiques :
Butterworth, Tchebycheff, Cauer, Bessel…

↓

Fonction de transfert :
$H(p)$

↓ ?

Fonction de transfert :
$H(z)$

Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Spécifications à respecter :
H ($\tilde{f}$)

? $f = \tilde{f} F_e$

Spécifications à respecter :
H (f)

Utilisation des bibliothèques
de modèles analogiques :
Butterworth, Tchebycheff, Cauer, Bessel…

Fonction de transfert :
H (p)

?

TRANSFORMEE BILINAIRE :
$$H(z) = [H(p)]_{p=\frac{2}{T_e}\frac{1-z^{-1}}{1+z^{-1}}}$$
Stabilité et réponse en fréquence conservées

Fonction de transfert :
H (z)

Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Spécifications à respecter :
H ($\tilde{f}$)

↓ **?** $f = \tilde{f} F_e$

Spécifications à respecter :
H (f)

↓

Utilisation des bibliothèques
de modèles analogiques :
Butterworth, Tchebycheff, Cauer, Bessel…

↓

Fonction de transfert :
H (p)

↓ **?**

Fonction de transfert :
H (z)

TRANSFORMEE BILINAIRE :

$$H(z) = [H(p)]_{p=\frac{2}{T_e}\frac{1-z^{-1}}{1+z^{-1}}}$$

Stabilité et réponse en fréquence conservées
MAIS Distorsion de l'axe des fréquences:

$$\tilde{f} = \frac{1}{\pi}\arctan(\pi f T_e)$$

Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Spécifications à respecter :
H ($\tilde{f}$)

? $\cancel{f = \tilde{f} F_e}$ $f = \frac{1}{\pi T_e}\tan(\pi \tilde{f})$

Prédistorsion

Spécifications à respecter :
H (f)

Utilisation des bibliothèques
de modèles analogiques :
Butterworth, Tchebycheff, Cauer, Bessel…

Fonction de transfert :
H (p)

TRANSFORMEE BILINAIRE :
$$H(z) = [H(p)]_{p=\frac{2}{T_e}\frac{1-z^{-1}}{1+z^{-1}}}$$

?

Stabilité et réponse en fréquence conservées
MAIS Distorsion de l'axe des fréquences:
$$\tilde{f} = \frac{1}{\pi}\arctan(\pi f T_e)$$

Fonction de transfert :
H (z)

Filtrage numérique

- **Synthèse des filtres numériques de type RII**

Spécifications à respecter :

H ($\tilde{f}$)

?

$\cancel{f = \tilde{f} F_e}$

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

Prédistorsion

Spécifications à respecter :

H (f)

Utilisation des bibliothèques
de modèles analogiques :
Butterworth, Tchebycheff, Cauer, Bessel…

Fonction de transfert :

H (p)

TRANSFORMEE BILINAIRE :

$$H(z) = [H(p)]_{p=\frac{2}{T_e}\frac{1-z^{-1}}{1+z^{-1}}}$$

?

Fonction de transfert :

H (z)

Stabilité et réponse en fréquence conservées
MAIS Distorsion de l'axe des fréquences:

$$\tilde{f} = \frac{1}{\pi} \arctan(\pi f T_e)$$

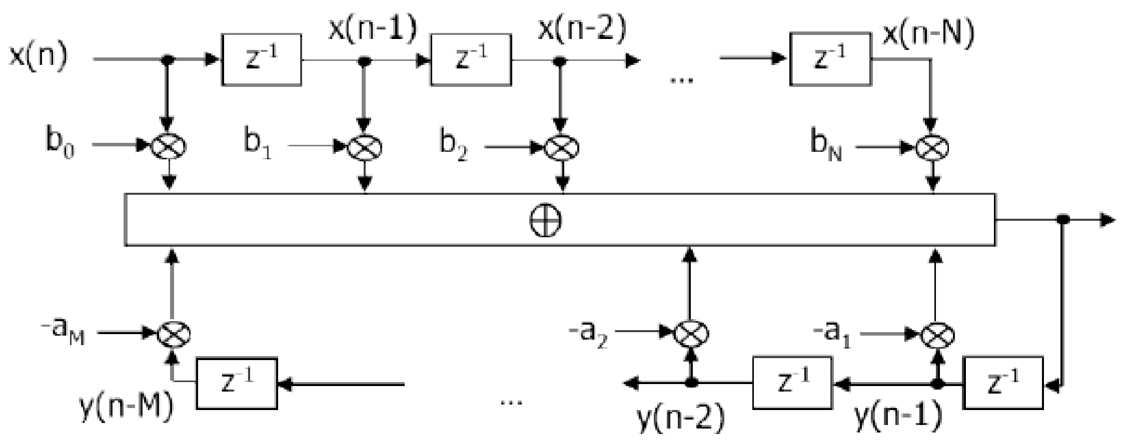
Voir poly d'exercices pour un exemple de synthèse de filtre passe-bas RII

Filtrage numérique

- **Implantation des filtres numériques**

→ <u>Structure directe :</u>

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



M+N+1 opération +/x, 2 files d’attente

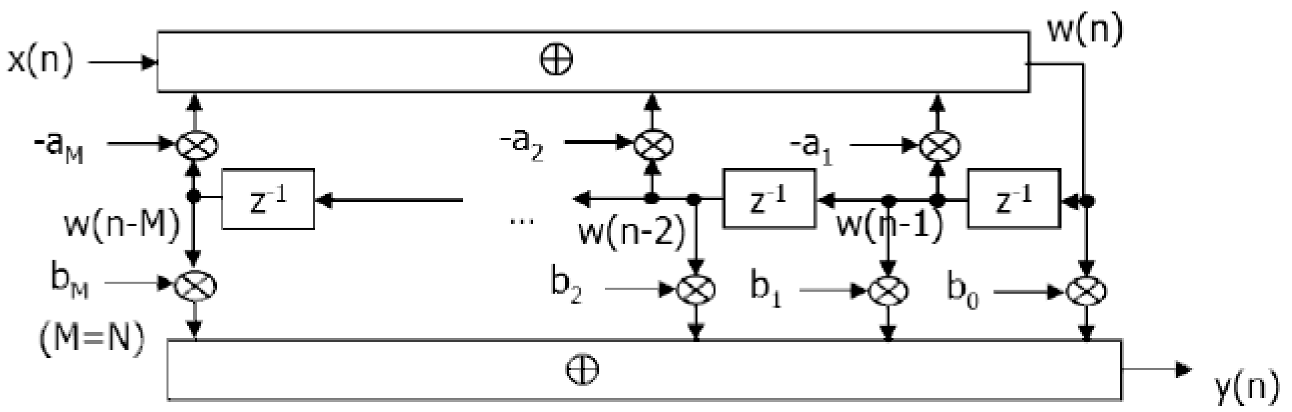
Filtrage numérique

- **Implantation des filtres numériques**

→ <u>Structure canonique :</u>

$$W(z) = \frac{X(z)}{\sum_{k=0}^{M-1} a_k z^{-k}}$$

$$\begin{cases} w(n) = -\displaystyle\sum_{k=1}^{M-1} a_k w(n-k) + x(n) \\ y(n) = \displaystyle\sum_{k=0}^{N-1} b_k w(n-k) \end{cases}$$



M+N+1 opération +/x, **1** file d'attente

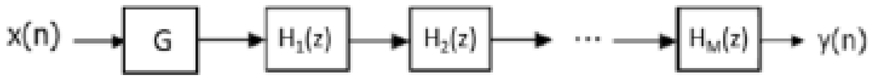
Filtrage numérique

- **Implantation des filtres numériques**

→ <u>Structures décomposées :</u>

→ Série (ou cascade) :

$$H(z) = G \prod_{i=0}^{M-1} H_i(z)$$

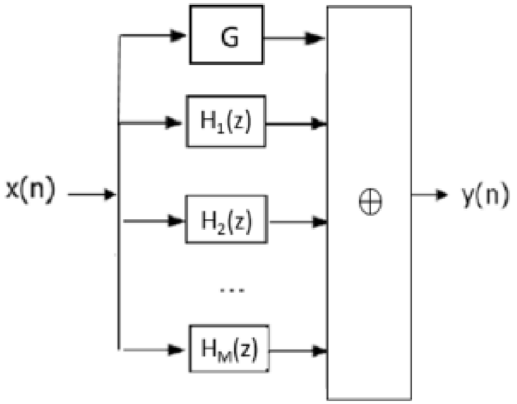


$H_i(z)$ Cellules du premier ou du deuxième ordre :

$$H_i(z) = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}} \qquad H_i(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

→ Parallèle :

$$H(z) = G + \sum_{i=0}^{M-1} H_i(z)$$



$H_i(z)$ Cellules du premier ou du deuxième ordre :

$$H_i(z) = \frac{b_0}{1 + a_1 z^{-1}}$$

$$H_i(z) = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

Filtrage numérique

- **Implantation des filtres numériques**

→ Structure non récursive :

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