I. Digital Filters Design

Introduction

RIF or RII?

- RII

➢Desired frequency response with very few poles and zeros

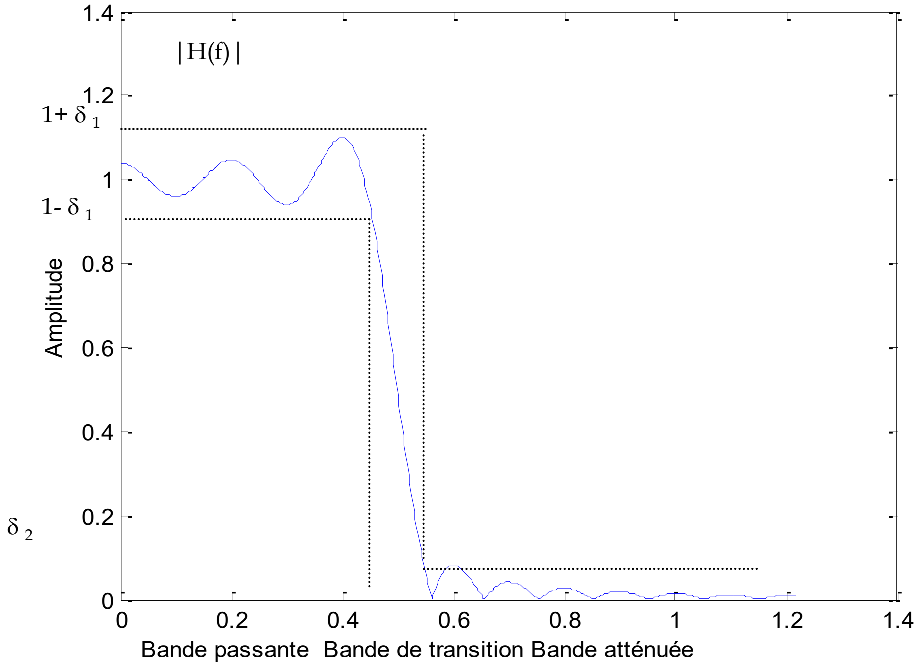
☻Risk of instability (recursion)

- RIF

✓Still stable

✓Possibility of having a constant group delay

☻No poles outside zeros (long impulse response)

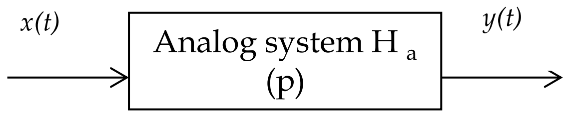


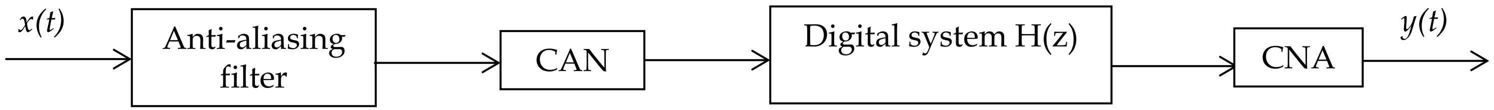
Introduction

1. Definition of Filter Characteristics (Desired frequency response or impulse response).
2. Determination of its coefficients (Best approximation respecting constraints of stability, speed, precision, linear phase shift, etc.).
3. Computer and/or electronic realization of the filter.

How ???

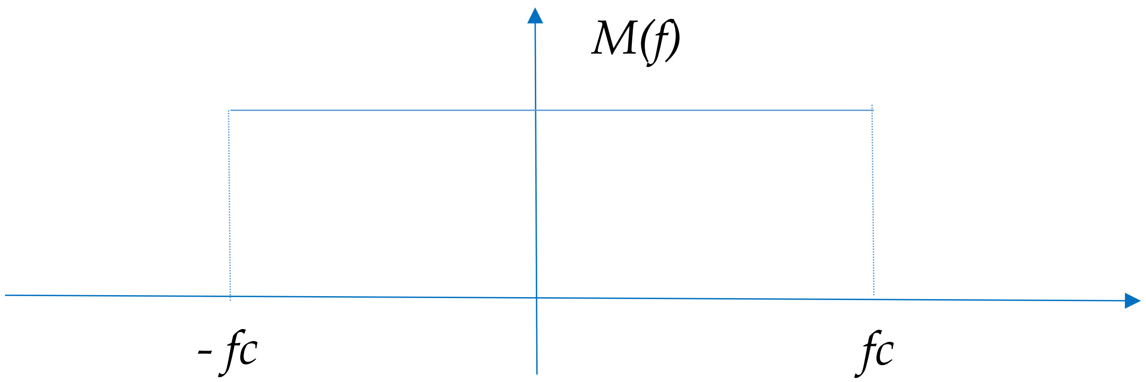
- ✓ Direct approach
- ✓ Indirect approach

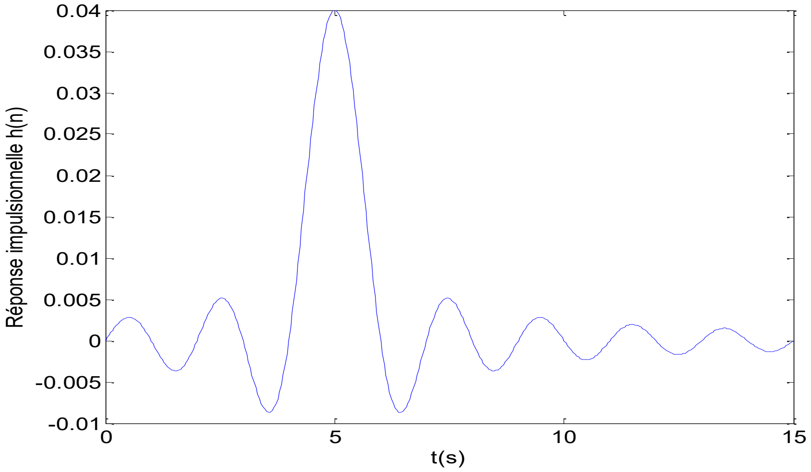




Introduction

✓Ideal Filter





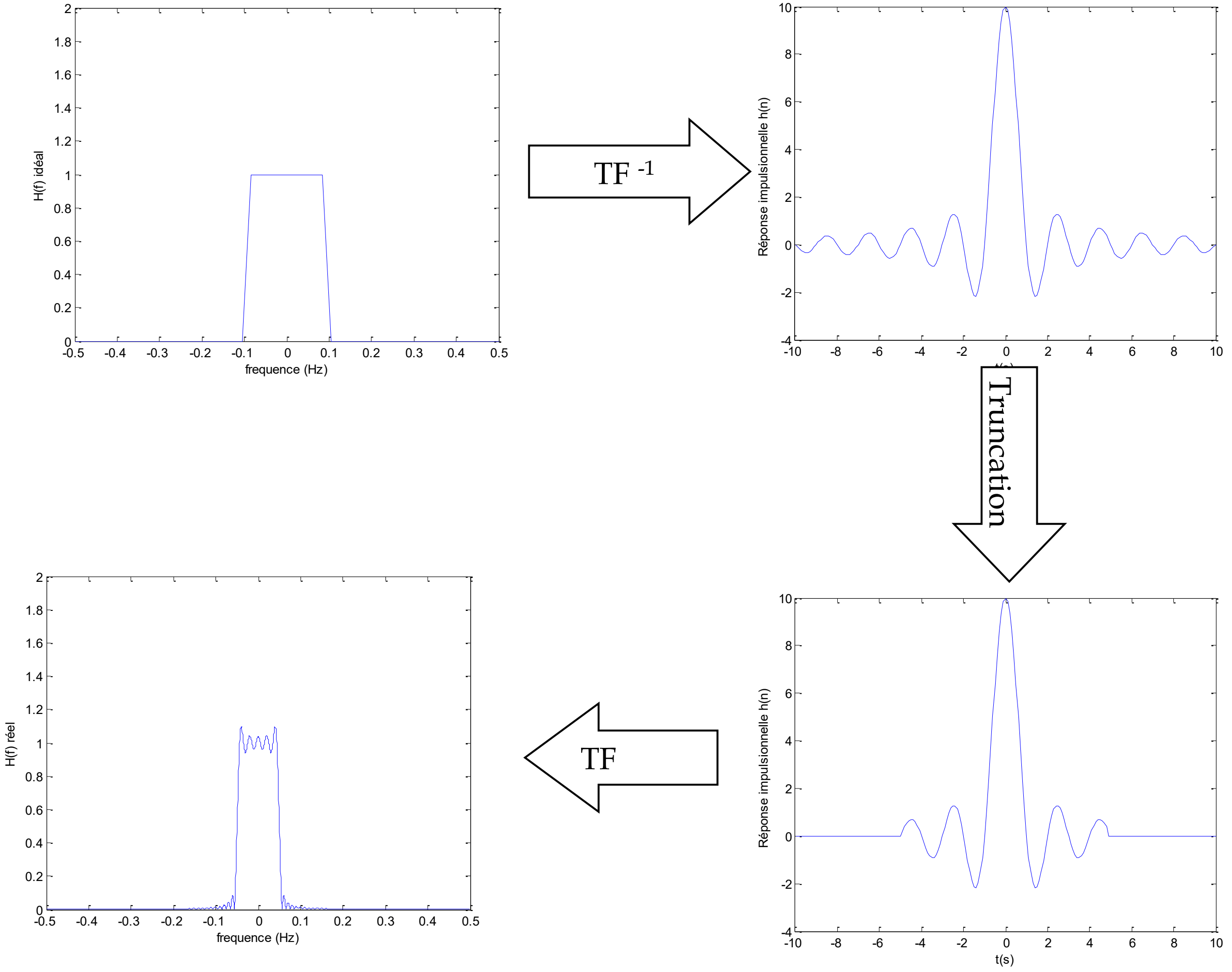
✓Real Causal Filter

$$\prod_{2f_c}(f)e^{-2\pi jfT_0} \qquad TF^{-1} \quad \Rightarrow \quad sinc(2f_c(t-T_0))$$

-∞ to +∞ Infinite ⇒ Limit

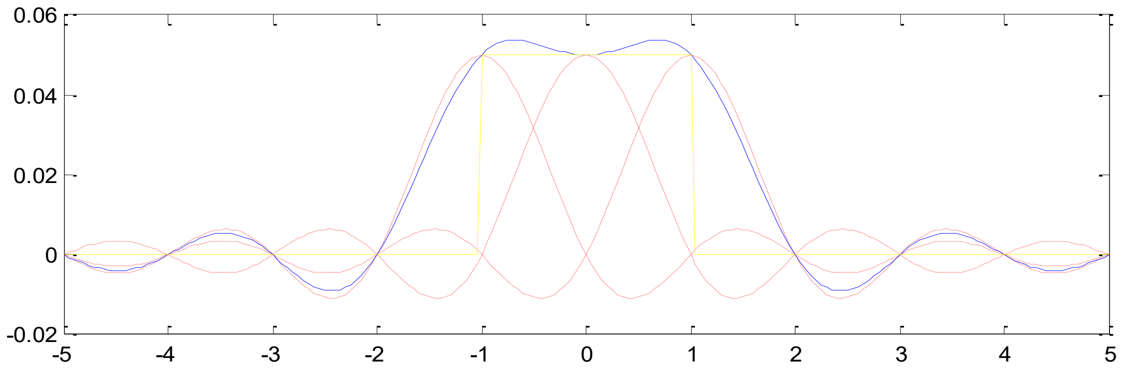
Introduction

Real Filter



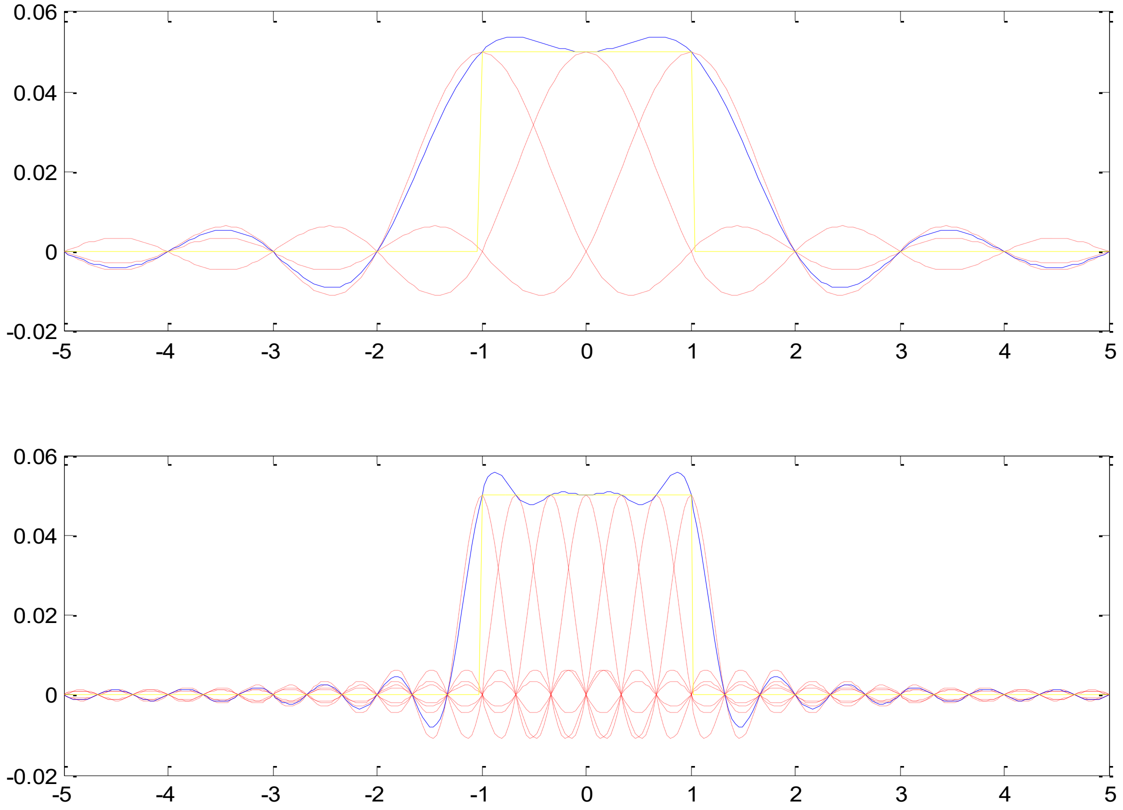
Introduction

Real Filter: Truncation on NTe points in the time domain $\Rightarrow$ Convolution with sinc which vanishes every 1/ NTe = fe /N in the frequency domain



Introduction

Real Filter: Truncation on NTe points in the time domain ⇒ Convolution with sinc which vanishes every 1/ NTe = fe /N in the frequency domain



Introduction

Filter specifications

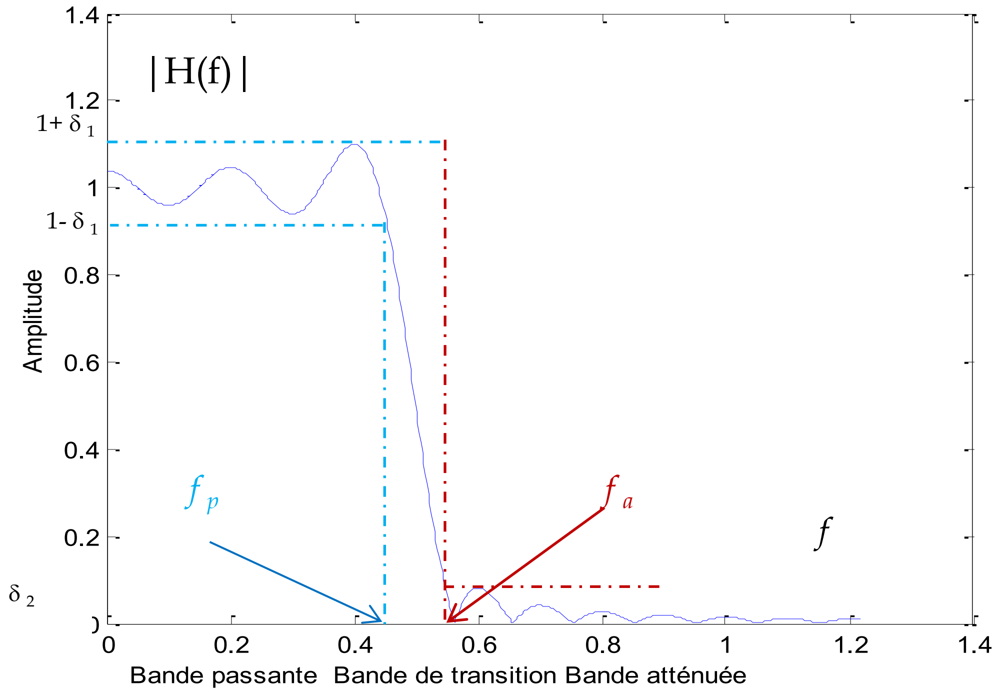
- The bandwidth from 0 to f_p
- The attenuated band (or cut BA) of f_{has} up to $f_e/2$
- The gain of the filter in the passband.
- The width $\Delta f = f_a - f_p$ from the transition zone

$$f_c = f_p + \Delta f/2 = (f_a + f_p)/2$$

- The amplitude of the bandwidth oscillations

$$\delta_1 \Rightarrow A_p = 20\log(1+\delta_1)$$

- The amplitude of the attenuated band ripples $\delta_2 \Rightarrow A_a = -20\log(\delta_2)$



1. FIR Filters Design

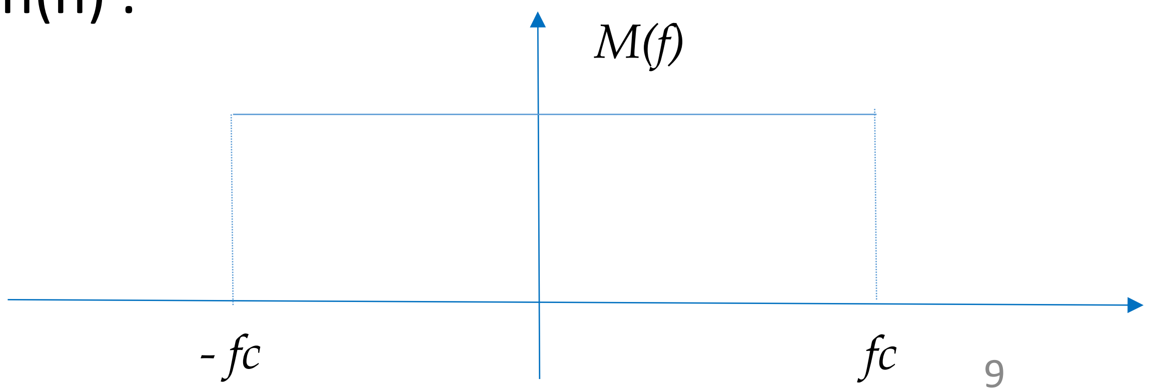
RIF: $y(n) = \sum_{i=0}^{N-1} b_i x(n-i) \quad \Rightarrow \quad H(z) = \sum_{i=0}^{N-1} b_i z^{-i} = \frac{b_0 \prod_{i=1}^{N}(z-z_i)}{z^N}$

A RIF filter has a polynomial (non-rational) transfer function, it cannot be obtained by transposition of an analog filter.

✓H(f) → TFTD $^{-1}$ → h(n) (window method)

✓H(f) → Frequency sampling H(k) → TFD $^{-1}$ → h(n) .

✓Other methods (iterative, optimization)

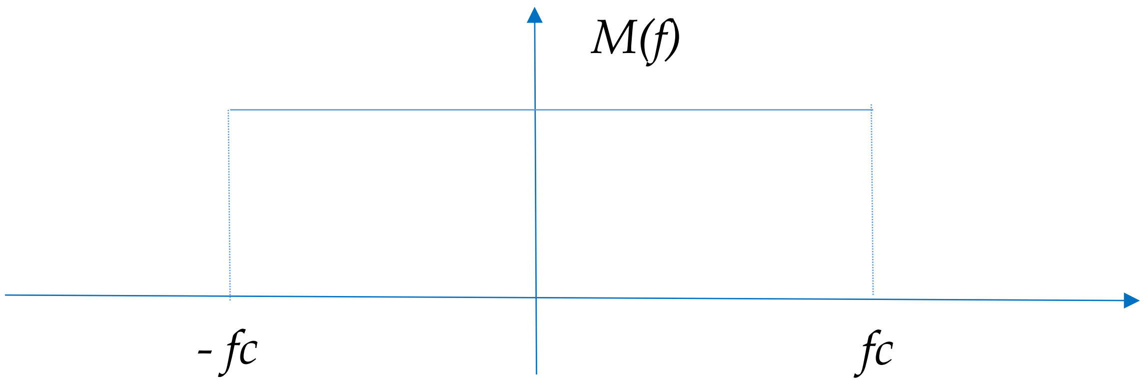


Faculty of Electrical Engineering, USTHB [akourgli@usthb.dz
http://perso.usthb.dz/~akourgli/

1. FIR Filters Design: Window Method

RIF: $y(n) = \sum_{i=0}^{N-1} b_i x(n-i)$

✓H(f) → TFTD $^{-1}$ → h(n)



$$h(n) = \frac{1}{f_e} \int_{-f_e/2}^{f_e/2} H(f)\, e^{2\pi j f n T_e} df$$

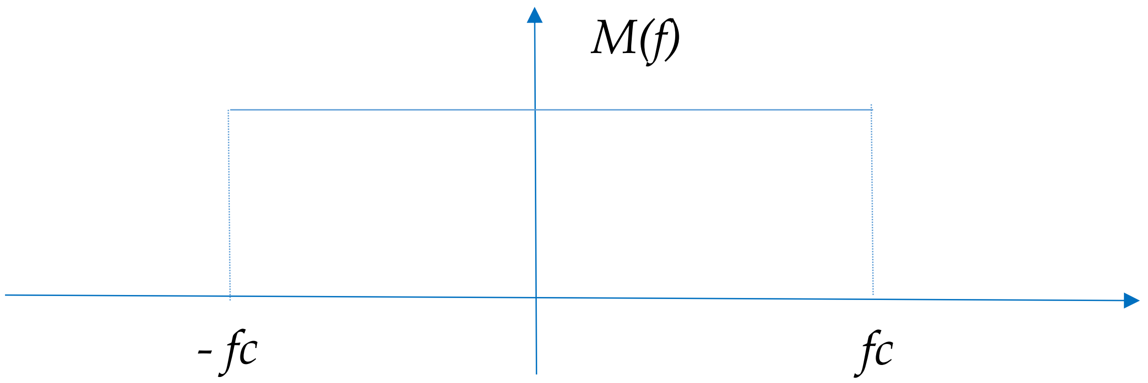
$$h(n) = \frac{1}{f_e} \int_{-f_c}^{f_c} e^{2\pi j f n T_e} df = \frac{1}{2\pi j n T_e f_e} \left[e^{2\pi j f n T_e}\right]_{-f_c}^{f_c}$$

$$h(n) = \frac{1}{2\pi j n} \left[e^{2\pi j f_c n T_e} - e^{-2\pi j f_c n T_e}\right] = \frac{1}{\pi n} \sin(2\pi f_c n T_e) = 2 f_c T_e \,\mathrm{sinc}(2 f_c n T_e)$$

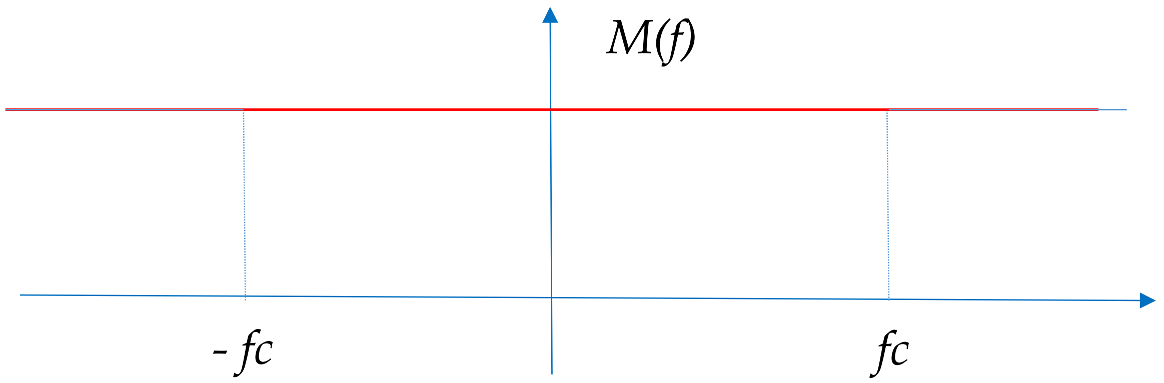
1. FIR Filters Design: Window method

We pose $f_c = \frac{f_c}{f_e/2} = 2f_cT_e$

$h(n) = 2f_cT_e sinc\,(2f_cnT_e) = f_c sinc(nf_c)$

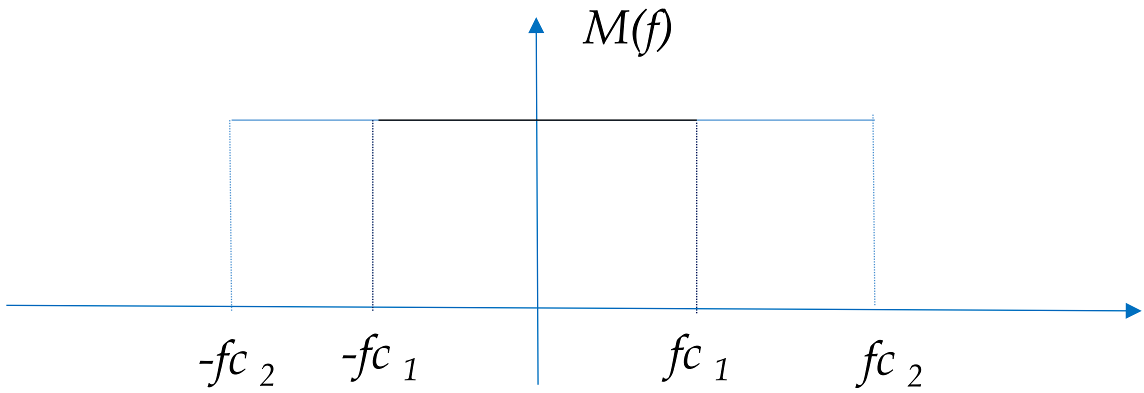


- High pass → $H_h(f) = 1 - H_b(f)$ → $h(n) = \delta(n) - f_c sinc(nf_c)$



Faculty of Electrical Engineering, USTHB [akourgli@usthb.dz
http://perso.usthb.dz/~akourgli/

1. FIR Filters Design: Window method



We pose $f_c = \frac{f_c}{f_e/2} = 2f_cT_e$

$$h(n) = 2f_cT_e sinc\,(2f_cnT_e) = f_c sinc(nf_c)$$

- High pass → $H_h(f) = 1 - H_b(f)$ → $h(n) = \delta(n) - f_c sinc(nf_c)$
- Band pass → $H_{bd}(f) = H_{b2}(f) - H_{b1}(f)$ → $h(n) = f_{c2} sinc(nf_{c2}) - f_{c1} sinc(nf_{c1})$
- Stop band → $H_h(f) = 1 - H_{bd}(f)$ → $h(n) = \delta(n) - f_{c2} sinc(nf_{c2}) + f_{c1} sinc(nf_{c1})$

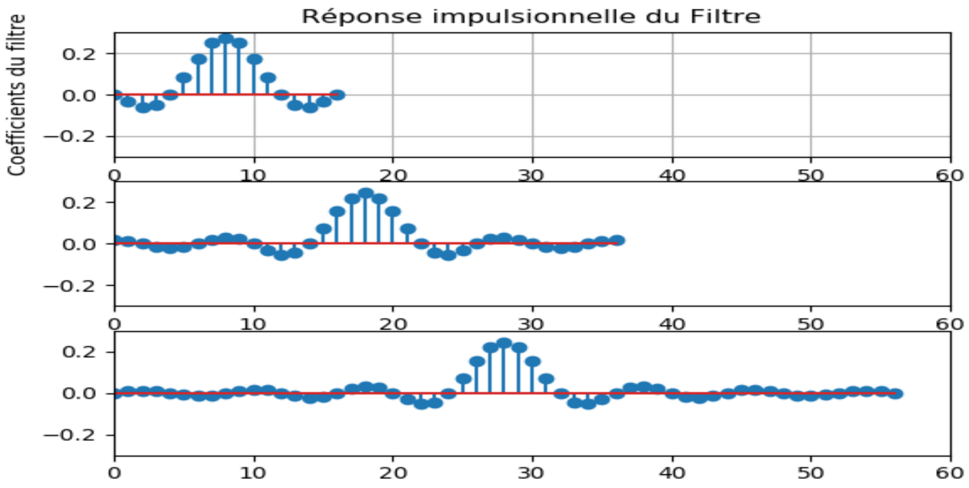
1. FIR Filters Design: Window method

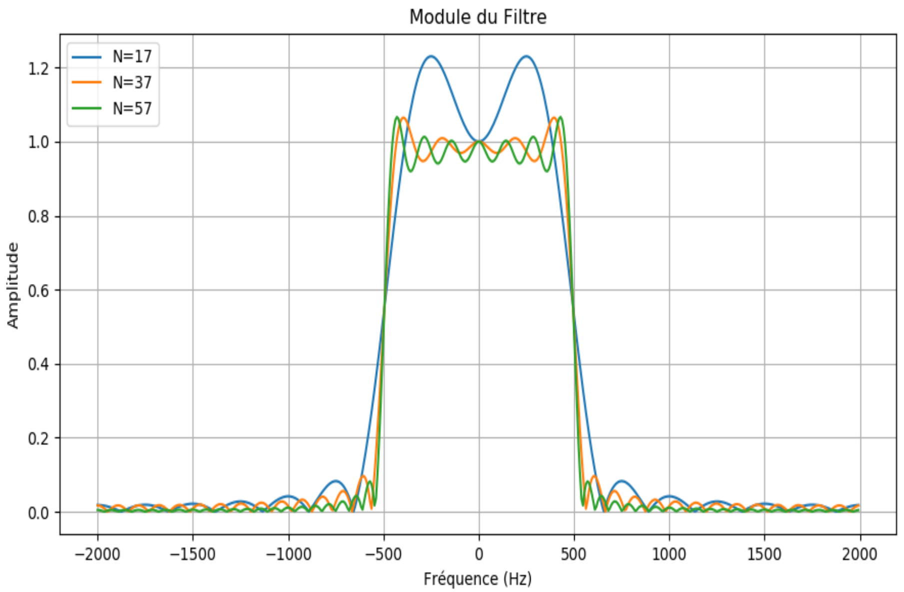
$$h(n) = 2f_c T_e sinc\,(2f_c n T_e) = f_c sinc(nf_c)$$

-∞ to +∞ Infinite ⇒ Limit

✓ Limit the number of samples of $h(n)$ to N

$h'_N(n) = h(n).w(n)$



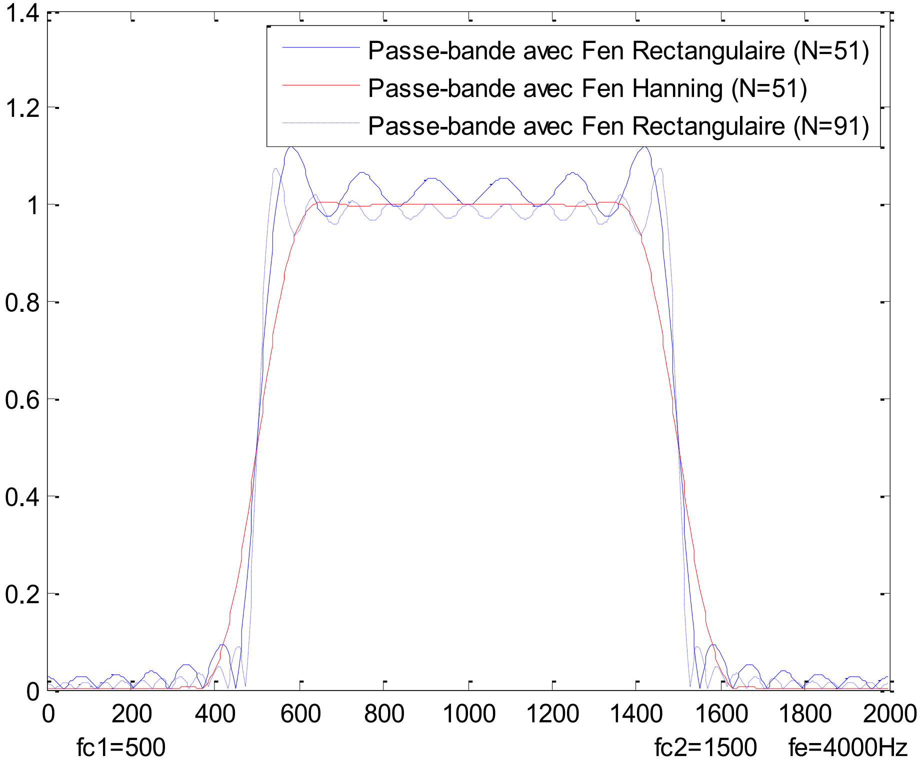


✓ Shift the response by N/2 to make it causal

1. FIR Filters Design: Window method

✓Using other windows

$$h'_N(n)=h(n).w(n)$$



2	3	1
Windows w $_N$ (n)	Width of Transition Δ f = (2 Δ f / fe)	Attenuation in attenuated band Aa
$w_{Rect}(n)=\begin{cases}1 & pour\ \lvert n\rvert \le \frac{N-1}{2}\\ 0 & ailleurs\end{cases}$	1.8/N	21
$w_{Han}(n)=\begin{cases}0.5+0.5\cos(\frac{2\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2}\\ 0 & ailleurs\end{cases}$	6.2/N	44
$w_{Ham}(n)=\begin{cases}0.54+0.46\cos(\frac{2\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2}\\ 0 & ailleurs\end{cases}$	6.6/N	53
$w_{Black}(n)=\begin{cases}0.42+0.5\cos(\frac{2\pi n}{N-1})+0.08\cos(\frac{4\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2}\\ 0 & ailleurs\end{cases}$	11/N	74

$\delta_1 \Rightarrow A_p = 20\log(1+\delta_1$

1. FIR Filters Design: Window method

Example: We want to synthesize a low-pass filter with cutoff frequency $fc = fe/10$ with $\Delta f = fe/5$ and an attenuated band ripple > 50 db

1. Normalize frequencies

- $fc/(fe/2) \Rightarrow fc = 0.2$
- $\Delta f/(fe/2) \Rightarrow \Delta f = 0.4$

$$h(n) = f_c sinc(nf_c) = 0.2 sinc(0.2n)$$

Windows $w_N(n)$	Width of Transition $\Delta f = (2\Delta f/fe)$	Attenuation in attenuated band Aa
$w_{Rect}(n) = \begin{cases} 1 & pour\ \lvert n\rvert \le \frac{N-1}{2} \\ 0 & ailleurs \end{cases}$	1.8/N	21
$w_{Han}(n) = \begin{cases} 0.5 + 0.5\cos(\frac{2\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2} \\ 0 & ailleurs \end{cases}$	6.2/N	44
$w_{Ham}(n) = \begin{cases} 0.54 + 0.46\cos(\frac{2\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2} \\ 0 & ailleurs \end{cases}$	6.6/N	53
$w_{Black}(n) = \begin{cases} 0.42 + 0.5\cos(\frac{2\pi n}{N-1}) + 0.08\cos(\frac{4\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2} \\ 0 & ailleurs \end{cases}$	11/N	74

2. Window selection based on Permissible ripple in attenuated band

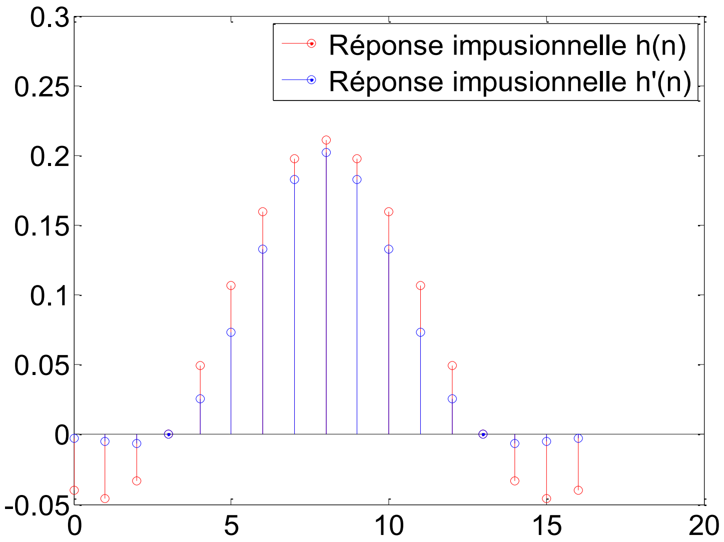
Hamming → Determination of N: N=6.6/ Δf =16.5 we take N=17

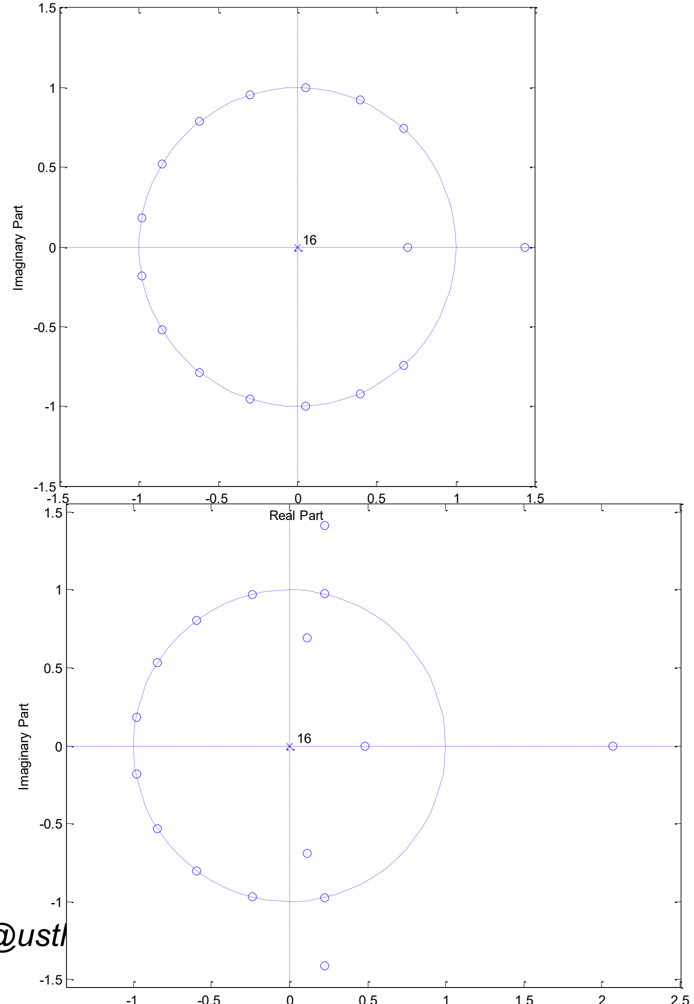
3. Calculation of $h'_N(n) = 0.2 sinc(0.2n)\left[0.54 + 0.46\cos(\frac{2\pi n}{16})\right]$ for-8≤n≤8

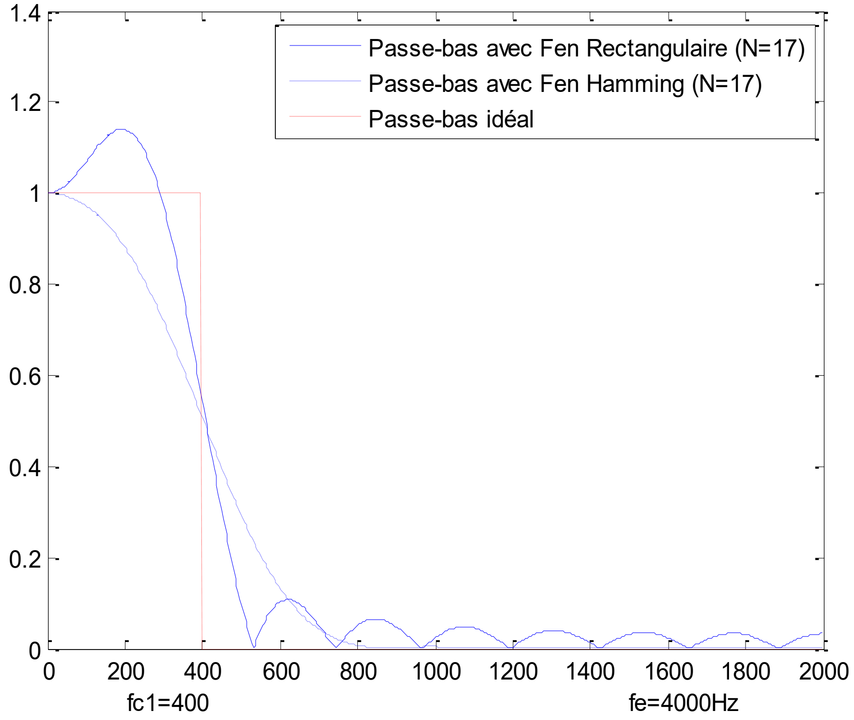
4. Shift the indices n by 8 to have a causal impulse response

1. FIR Filters Design: Window method

Example: We want to synthesize a low-pass filter with cutoff frequency $fc = fe/10$ with $\Delta f = fe/5$ and an attenuated band ripple > 50 db

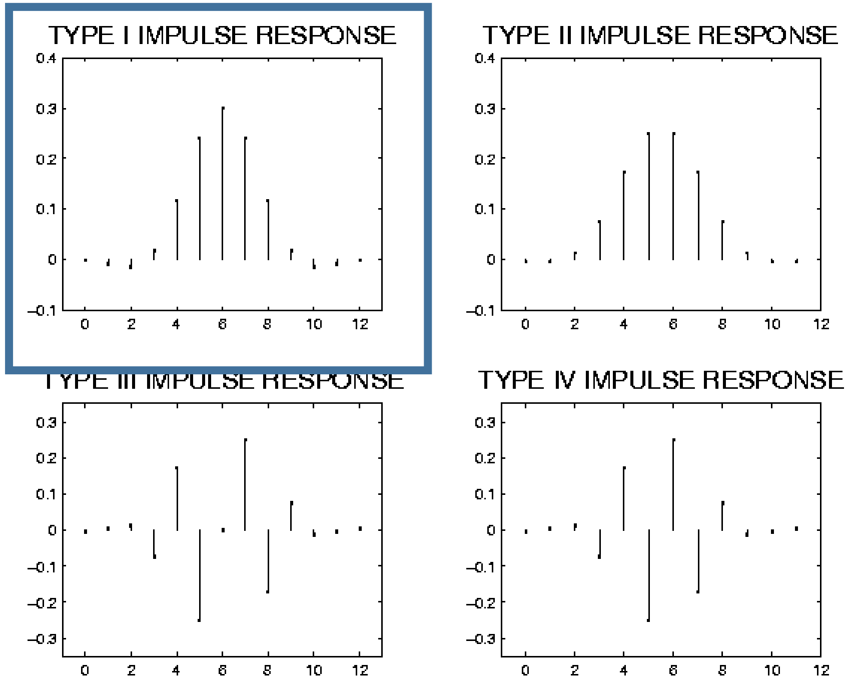






1. FIR Filters Design: Window method

Types of Filters



https://cnx.org/contents/1prPUN_Y@4.38:ScWn0u94@3/Zero-Locations-of-Linear-Phase-FIR-Filters

Type I (odd N) $\Rightarrow \quad H(f) = \sum_{n=0}^{N-1} b_n e^{-2\pi j f n T_e}$

$$H(f) = b_0 + b_1 e^{-2\pi j f T_e} + \cdots b_{\frac{N-1}{2}} e^{-2\pi j f \frac{N-1}{2} T_e} + \ldots + b_1 e^{-2\pi j f (N-2) T_e} + b_0 e^{-2\pi j f (N-1) T_e}$$

$$H(f) = 2e^{-2\pi j f \frac{N-1}{2} T_e}\left(b_0 \cos\left(2\pi f \frac{N-1}{2} T_e\right) + b_1 \cos(\ldots) + \cdots \ldots\ldots + b_{(N-1)/2}\right)$$

$$H(f) = 2e^{-\pi j f (N-1)\, T_e}\left(b_0 \cos(\pi f (N-1) T_e) + b_1 \cos(\ldots) + \cdots \ldots . + 0.5 b_{(N-1)/2}\right)$$

Example

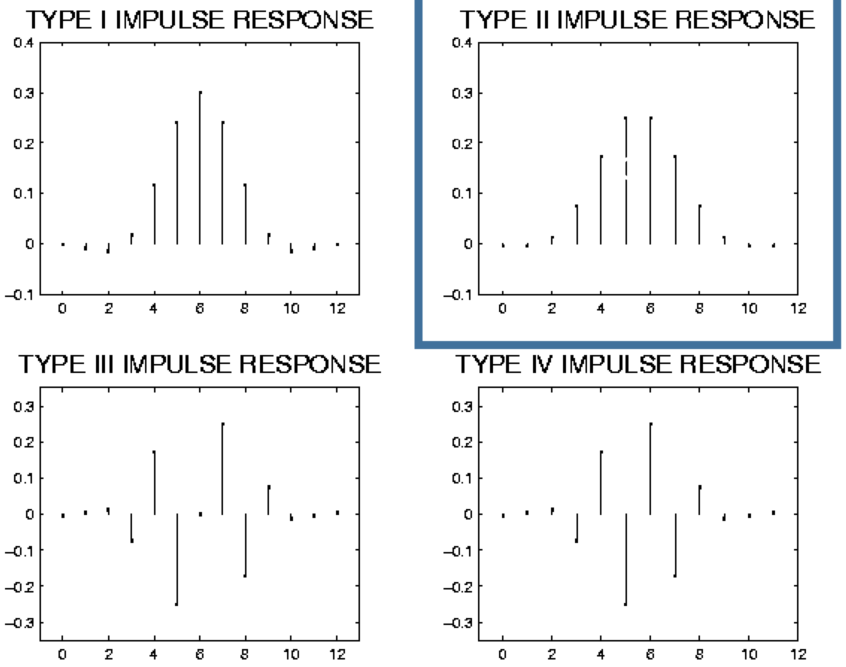
$$H(f) = b_0 + b_1 e^{-2\pi j f T_e} + b_2 e^{-4\pi j f T_e} + b_1 e^{-6\pi j f T_e} + b_0 e^{-8\pi j f T_e}$$

$$H(f) = e^{-4\pi j f T_e}\left(b_0 e^{4\pi j f T_e} + b_1 e^{2\pi j f T_e} + b_2 + b_1 e^{-2\pi j f T_e} + b_0 e^{-4\pi j f T_e}\right)$$

$$H(f) = 2e^{-4\pi j f T_e}\left(b_0 \cos 4\pi f T_e + b_1 \cos 2\pi f T_e + 0.5\, b_2\right)$$

1. FIR Filters Design: Window method

Types of Filters



https://cnx.org/contents/1prPUN_Y@4.38:ScWn0u94@3/Zero-Locations-of-Linear-Phase-FIR-Filters

Type II (N even) $\Rightarrow \quad H(f) = \sum_{n=0}^{N-1} b_n e^{-2\pi j f n T_e}$

$$H(f) = b_0 + b_1 e^{-2\pi j f T_e} + \cdots + \ldots + b_1 e^{-2\pi j f (N-2) T_e} + b_0 e^{-2\pi j f (N-1) T_e}$$

$$H(f) = 2e^{-2\pi j f \frac{N-1}{2} T_e}(b_0 \cos(\pi f N T_e) + b_1 \cos(\ldots) + .. + b_{\frac{N}{2}-1} \cos(\ldots))$$

$$H(f) = 2e^{-\pi j f (N-1) T_e}(b_0 \cos(\pi f (N-1) T_e) + b_1 \cos(\ldots) + .. + b_{\frac{N}{2}-1} \cos(\ldots))$$

Example

$$H(f) = b_0 + b_1 e^{-2\pi j f T_e} + b_1 e^{-4\pi j f T_e} + b_0 e^{-6\pi j f T_e}$$

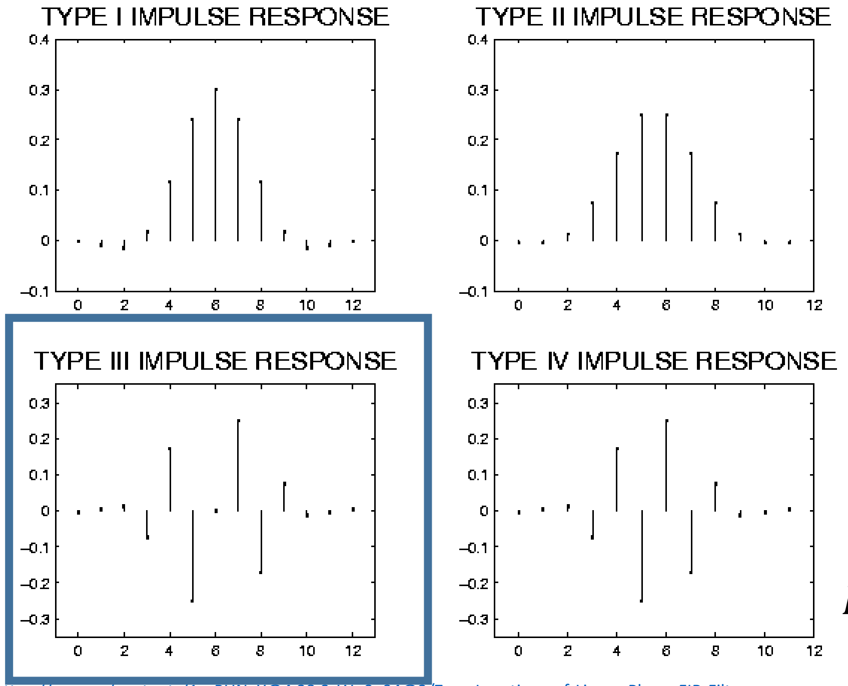
$$H(f) = e^{-3\pi j f T_e}(b_0 e^{3\pi j f T_e} + b_1 e^{\pi j f T_e} + b_1 e^{-\pi j f T_e} + b_0 e^{-3\pi j f T_e})$$

$$H(f) = 2e^{-3\pi j f T_e}(b_0 \cos 3\pi f T_e + b_1 \cos \pi f T_e)$$

$$H(\tfrac{f_e}{2}) = 2e^{-\frac{3\pi}{2} j}(b_0 \cos 3\pi/2 + b_1 \cos \pi/2)$$

1. FIR Filters Design: Window method

Types of Filters



https://cnx.org/contents/1prPUN_Y@4.38:ScWn0u94@3/Zero-Locations-of-Linear-Phase-FIR-Filters

Type III (odd N)$\Rightarrow$ $H(f) = \sum_{n=0}^{N-1} b_n e^{-2\pi j f n T_e}$

$$H(f) = b_0 - b_1 e^{-2\pi j f T_e} + \cdots - \cdots + 0 - \cdots + b_1 e^{-2\pi j f (N-2) T_e} - b_0 e^{-2\pi j f (N-1) T_e}$$

$$H(f) = 2j e^{-2\pi j f \frac{N-1}{2} T_e} \left(b_0 \sin\left(2\pi f \frac{N-1}{2} T_e\right) - b_1 \sin(\dots) + \cdots - \cdots + \cdots - \cdots\right)$$

$$H(0) = 0$$

Example

$$H(f) = b_0 - b_1 e^{-2\pi j f T_e} + 0 + b_1 e^{-6\pi j f T_e} - b_0 e^{-8\pi j f T_e}$$

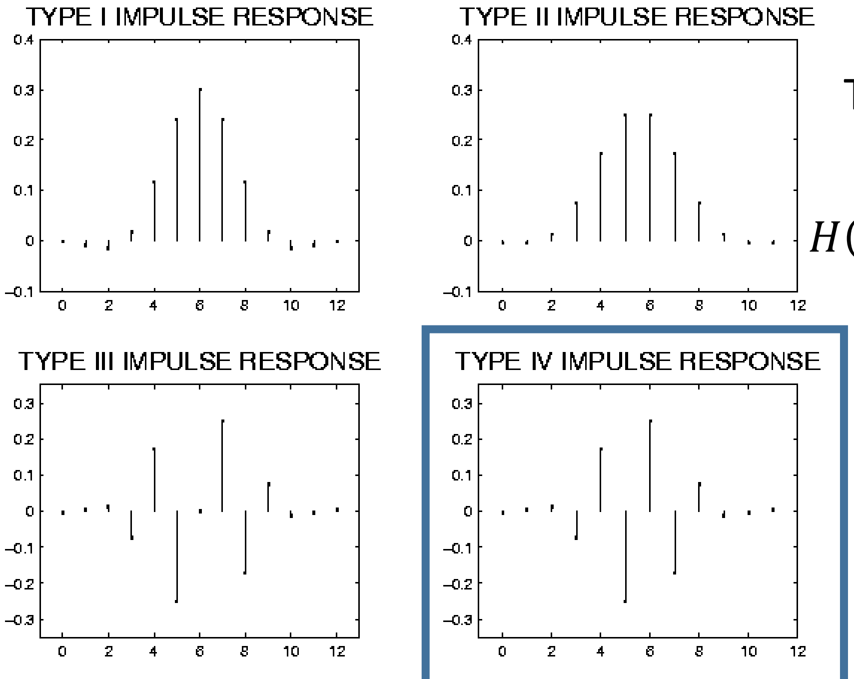
$$H(f) = e^{-4\pi j f T_e} \left(b_0 e^{4\pi j f T_e} - b_1 e^{2\pi j f T_e} + b_1 e^{-2\pi j f T_e} - b_0 e^{-4\pi j f T_e}\right)$$

$$H(f) = 2j\, e^{-4\pi j f T_e} (b_0 \sin 4\pi f T_e \quad - b_1 \sin 2\pi f T_e)$$

$$H\left(\frac{f_e}{2}\right) = 2j e^{-2\pi j} (b_0 \sin 2\pi + b_1 \sin \pi) = 0$$

1. FIR Filters Design: Window method

Types of Filters



https://cnx.org/contents/1prPUN_Y@4.38:ScWn0u94@3/Zero-Locations-of-Linear-Phase-FIR-Filters

Type VI (N even)⇒ $H(f) = \sum_{n=0}^{N-1} b_n e^{-2\pi j f n T_e}$

$$H(f) = 2je^{-2\pi jf\frac{N-1}{2}T_e}(b_0 \sin(\pi f(N-1)T_e) - b_1 \sin(\pi f(N-2)T_e) + \cdots - \cdots + \cdots - \cdots)$$

$$H(0) = 0$$

Example

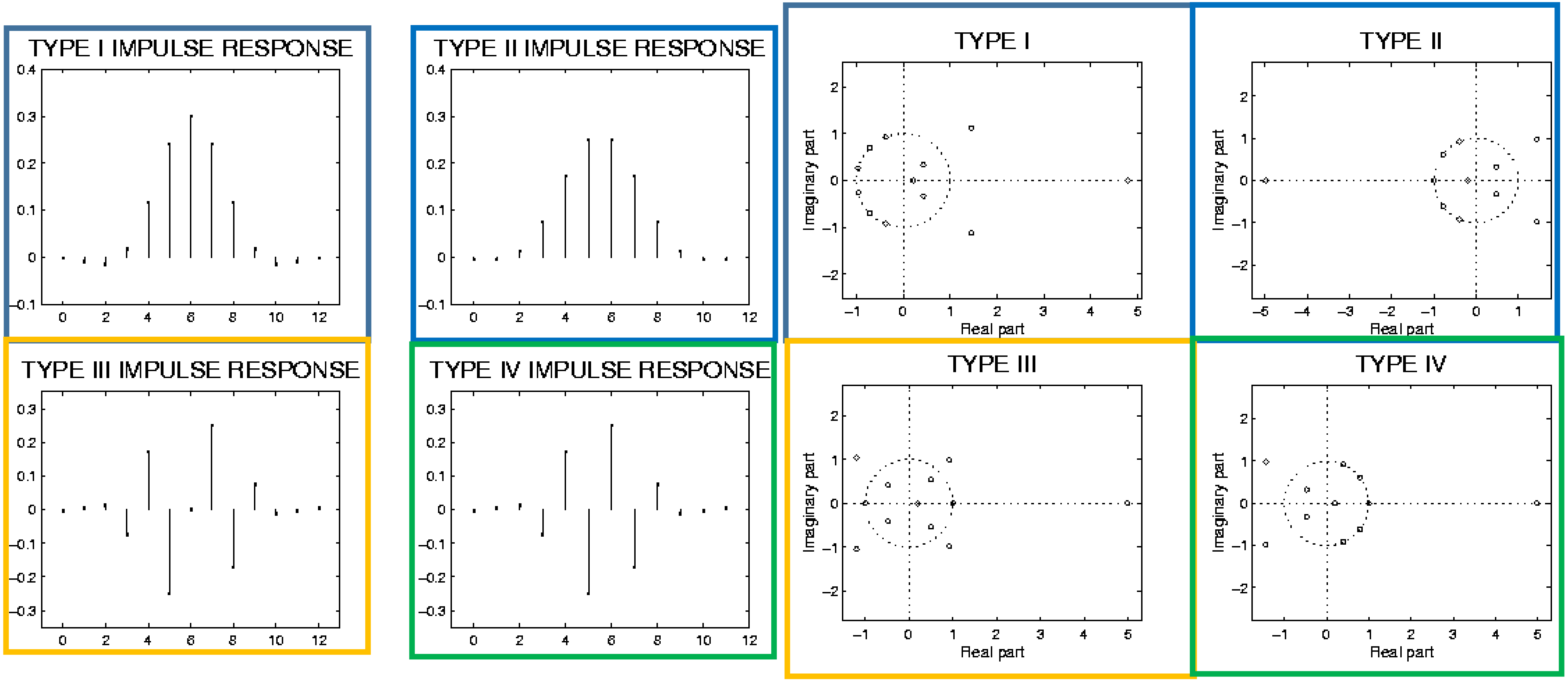
$$H(f) = b_0 - b_1 e^{-2\pi jfT_e} + b_1 e^{-4\pi jfT_e} - b_0 e^{-6\pi jfT_e}$$

$$H(f) = e^{-3\pi jfT_e}(b_0 e^{3\pi jfT_e} - b_1 e^{\pi jfT_e} + b_1 e^{-\pi jfT_e} - b_0 e^{-3\pi jfT_e})$$

$$H(f) = 2j\, e^{-4\pi jfT_e}(b_0 \sin 3\pi f T_e \quad - b_1 \sin \pi f T_e)$$

1. FIR Filters Design: Window method

Types of Filters



Faculty of Electrical Engineering, USTHB [akourgli@usthb.dz
http://perso.usthb.dz/~akourgli/

1. FIR Filters Design: Window method

Types of Filters

✓Type I filter is the most commonly used

✓The type II filter has a zero at -1 (number of zeros is odd so distribution of zeros in conjugate pairs + a zero at fe /2 since a sum of cosines), it cannot therefore be used for a high-pass or a band-stop.

✓Type III (odd N) and IV (even N) filters allow to obtain an anti-symmetric impulse response with linear phase shift as well (sum of sines instead of cosines).

They both have zeros at 1 (sin at f=0 is zero), so their use is not suitable for low-pass filters. In addition, the type III filter also has a 0 at -1, so it can only be used for a bandpass.

2. RII Filters Design

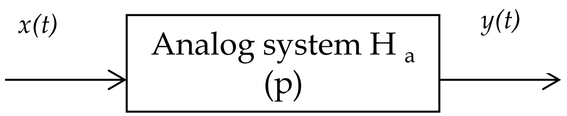
- ✓Possibility to obtain a narrow transition band for a reasonable order,
- ✓Risk of instability due to high numerical sensitivity of the coefficients,
- ✓Strongly nonlinear phase variation.

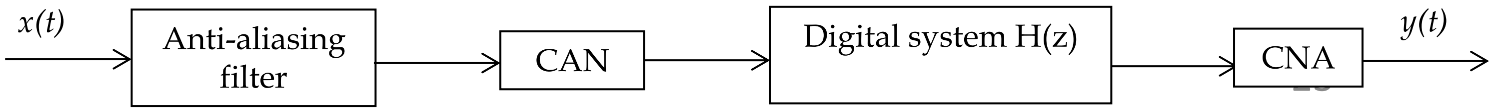
$$y(n)=\sum_{i=0}^{N} b_i x(n-i)-\sum_{i=1}^{M} a_i y(n-i) \Longrightarrow H(z)=\frac{\sum_{i=0}^{N} b_i \, .z^{-i}}{1+\sum_{i=1}^{M} a_i \, .z^{-i}}$$

How ???

- ✓ Direct approach (placement of poles and zeros)
- Indirect approach (Synthesis of an analog filter $H_a(p) \rightarrow p=$ fct $(z) \rightarrow H(z)$

Impulse Invariance and Bilinear Transformation)





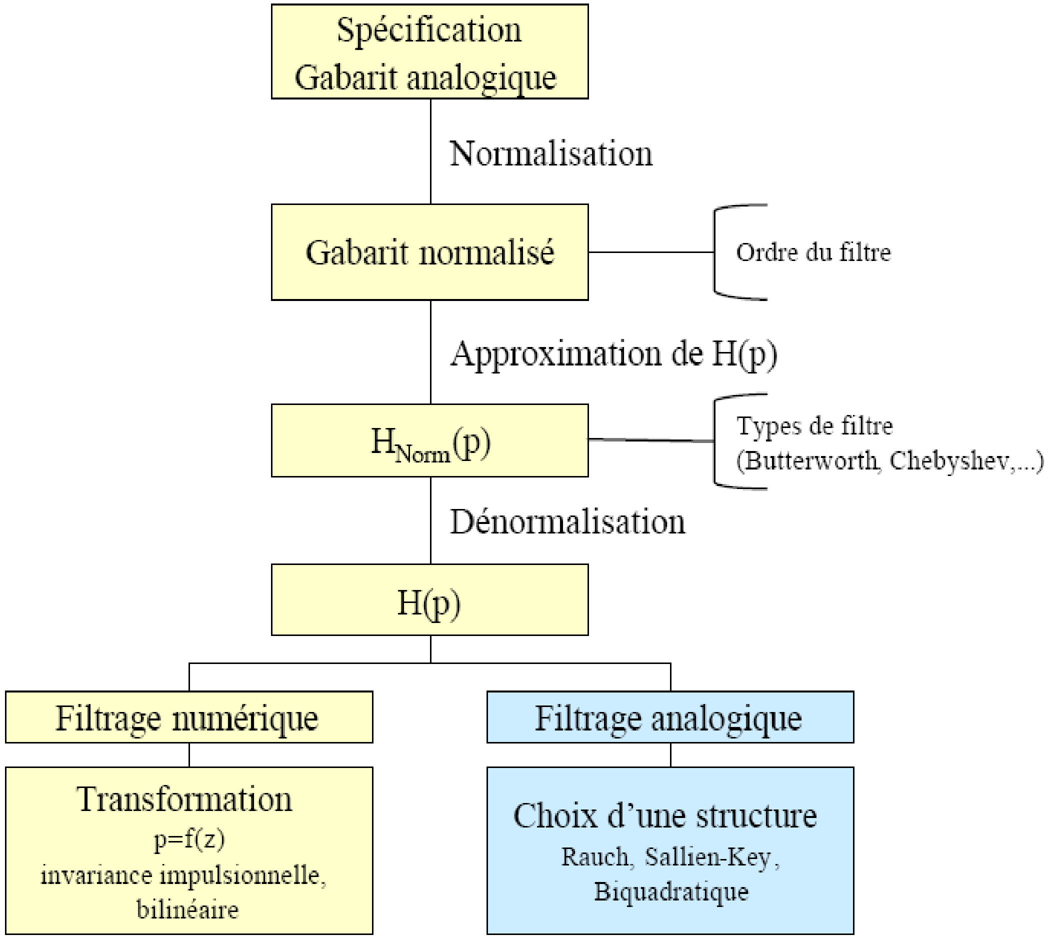
2. RII Filters Design : Indirect Methods

Synthesize a recursive digital filter from an analog filter taken as a model:

$H_a(p) \rightarrow p = \text{fct}(z) \rightarrow H(z)$

✓ the filter must have an imposed impulse or step response: these are the methods of impulse invariance and step invariance.

✓ the filter must have a frequency response falling within a given template: this is the bilinear transformation .

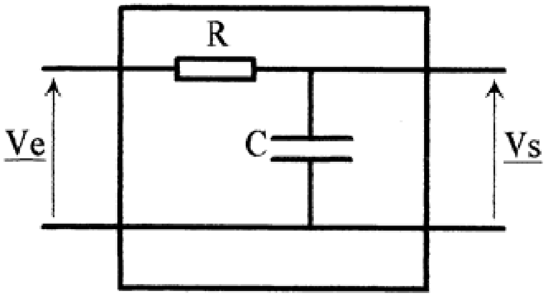


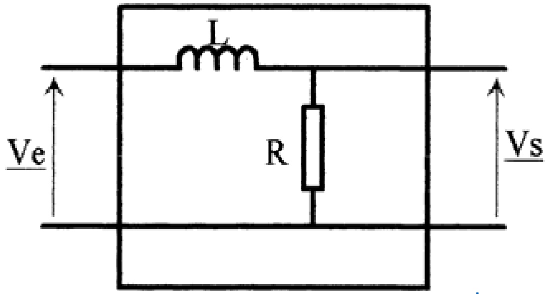
2. RII Filters Design : Indirect Methods

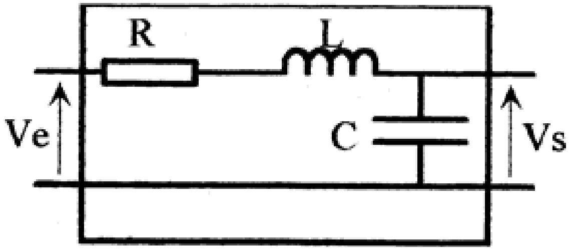
Reminders on analog filters

To design an analog filter:

✓Use passive filters by combining resistors, capacitors and/or coils.

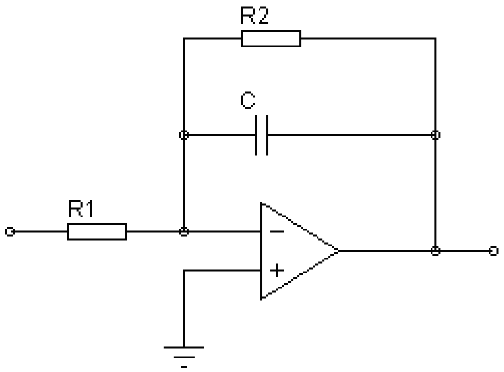


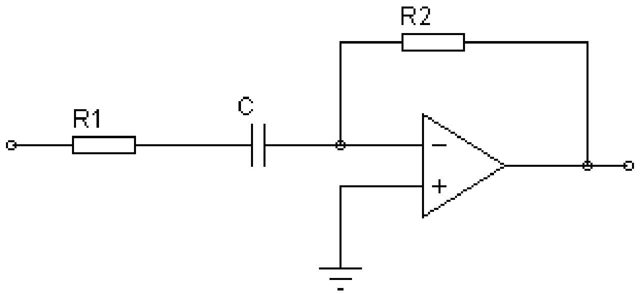




https://les-electroniciens.com/sites/default/files/cours/filtrage_analogue.pdf

✓Use active filters with an amplifying element (transistor, AIL, etc.)





://ressource.electron.free.fr/bts/cours/le-filtrage-actif-passif.pdf

2. RII Filters Design : Indirect Methods

Reminders on analog filters

Notes B

From a few hundred MHz, all components (including wires or printed circuits) have non-negligible inductances and capacitances at these frequencies and are part of the filter → low-pass .

- ✓Passive filters (L,C, quartz, etc.) are used for high frequencies
- ✓integrated linear amplifiers (ALI) limited to 1Mhz or switched capacitor filters (integrated R and C, ALI, MOS controlled switch) have programmable frequencies (<10 Mhz)
- ✓digital filters are limited to frequencies < 100MHz

2. RII Filters Design : Indirect Methods

Reminders on analog filters

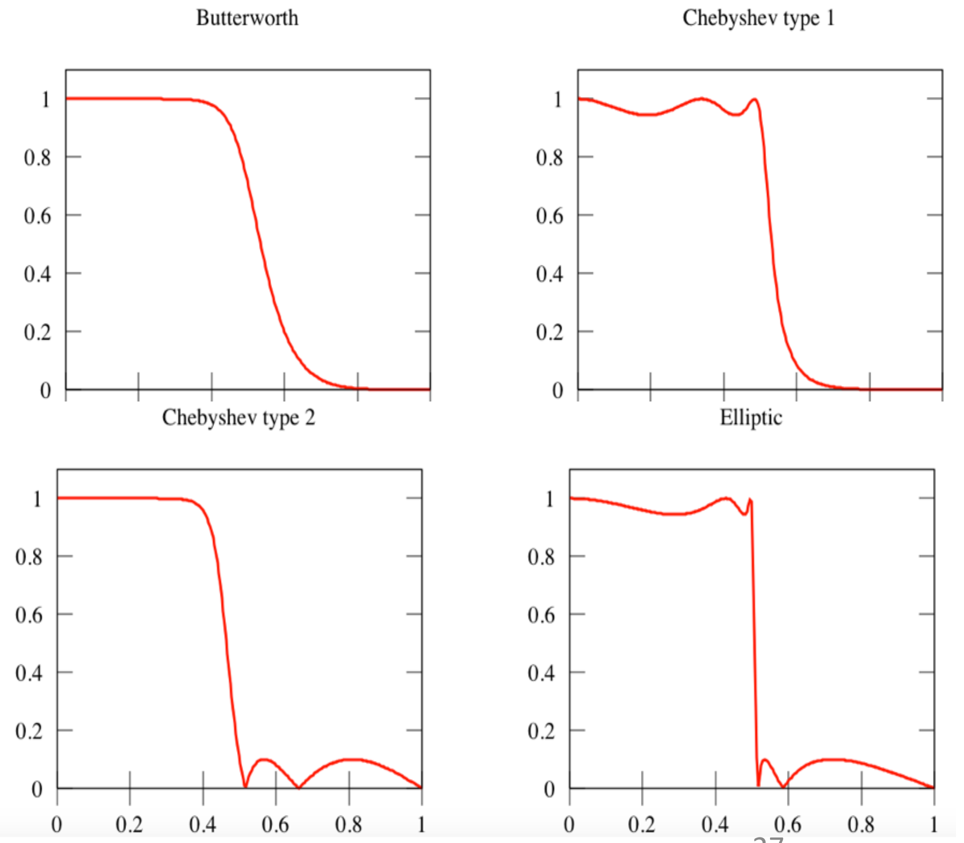
A set of analog filters tested and proven for their properties in terms of attenuation slope and ripple in the passband and attenuated

Butterworth : *Not very steep* cutoff but *constant gain in band* Passing, quasi-linear phase shift

Chebyshev : Significant *cutting* stiffness but
N ripples in the passband (Cheb 1) or attenuated (Cheb II)

Cauer (also called elliptical): *Extremely steep cut* but *ripples* in the bandwidth and Attenuated.

Bessel : *Almost constant group delay* but *bad selectivity even for high order*



2. RII Filters Design : Indirect Methods

Reminders on analog filters

Butterworth : the amplitude response is maximally flat in the bandwidth and monotonically decreasing from a certain frequency (cutoff frequency)

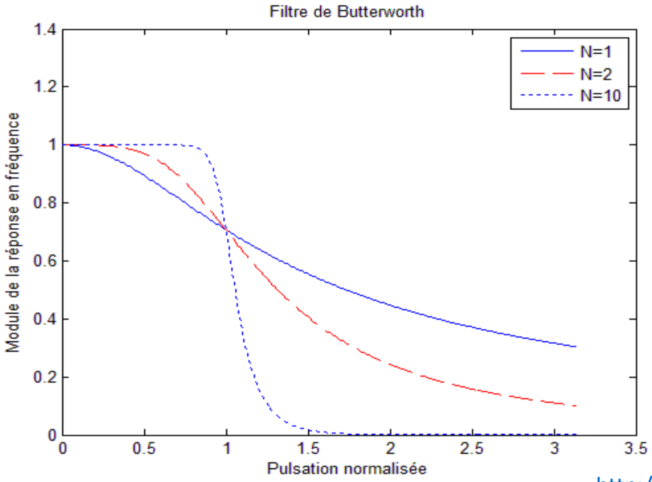
$$|H(\omega)| = \frac{1}{\sqrt{1+\left(\frac{\omega}{\omega_c}\right)^{2N}}} \qquad |H(\omega)| = \frac{1}{\sqrt{1+\varepsilon^2 . \left(\frac{\omega}{\omega_c}\right)^{2N}}}$$

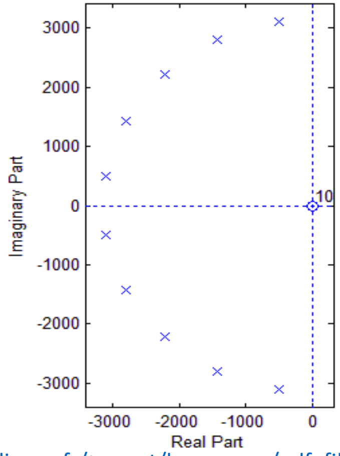
$$N \geq \frac{\log\left(10^{\frac{Aa}{10}} - 1\right)}{2\log(\omega_a/\omega_p)} \qquad N \geq= \frac{\log\left((10^{\frac{Aa}{10}} - 1)/(10^{\frac{Ap}{10}} - 1)\right)}{2\log(\omega_a/\omega_p)}$$

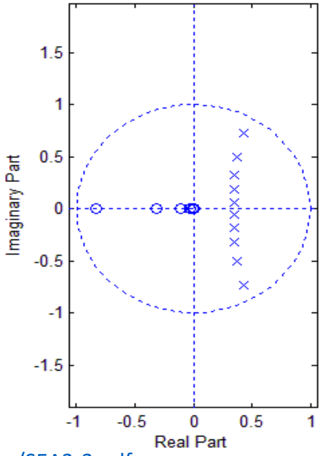
Ordre du filtre	Dénominateur (le numérateur est à 1)
2	$p^2+\sqrt{2}\,p+1$
3	$(p^2+p+1)(p+1)$
4	$(p^2+1.8477p+1)(p^2+0.7653p+1)$
5	$(p^2+1.6180p+1)(p^2+0.6180\,p+1)(p+1)$
6	$(p^2+1.9318p+1)(p^2+\sqrt{2}\,p+1)(p^2+0.5176p+1)$

$\omega_c = \sqrt{\omega_p \omega_a}$ cut-off pulse and

$\varepsilon = \sqrt{10^{\frac{Ap}{10}} - 1}$ Design parameter setting the tolerance in bandwidth $\delta_1 = 1 - \frac{1}{\sqrt{1+\varepsilon^3}}$







http://www.lirmm.fr/~nouet/homepage/pdf_files/SEA3-2.pdf

2. RII Filters Design : Indirect Methods

Reminders on analog filters

Chebyshev : Significant cutoff stiffness but ripples in the bandwidth (Chebyshev 1) or attenuated (Chebyshev **2**)

$$|H(\omega)| = \frac{1}{\sqrt{1+\varepsilon^2 . T_N^2\left(\frac{\omega}{\omega_c}\right)}}$$

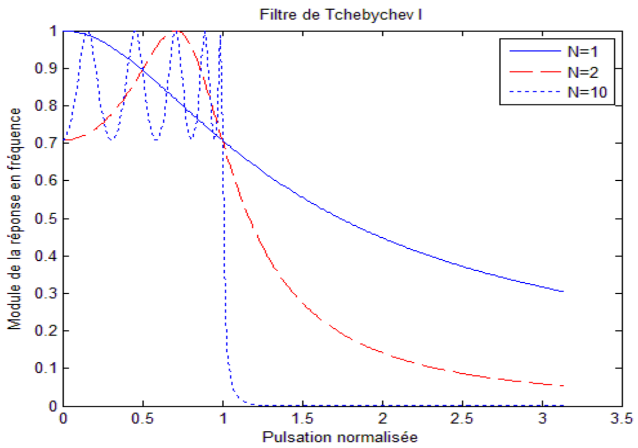
$$T_N(x) = \begin{cases} \cos[N.ar\cos(x)] & pour \quad 0 \le |x| \le 1 \\ \cosh[N.ar\cosh(x)] & pour \quad |x| \ge 1 \end{cases}$$

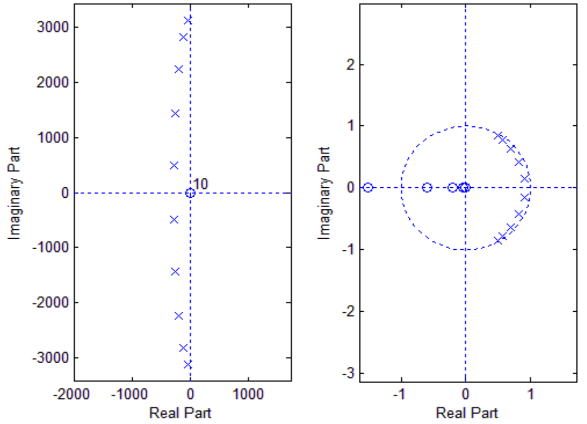
Ordre du filtre	Dénominateur (le numérateur est à 1) ε=1
2	$0.9070\ p^2+0.9956\ p+1$
3	$(1.0058\ p^2+0.497\ p+1)(2.023\ p+1)$
4	$(1.0136\ p^2+0.2828\ p+1)(3.5791\ p^2+2.4113\ p+1)$
5	$(2.3293\ p^2+1.0911\ p+1)(1.0118\ p^2+0.1610\ p+1)(3.454p+1)$

$$N \ge \frac{\log\left(\sqrt{\left(10^{\frac{Aa}{10}}-1\right)\Big/\left(10^{\frac{Ap}{10}}-1\right)} + \sqrt{\left(10^{\frac{Aa}{10}}-1\right)\Big/\left(10^{\frac{Ap}{10}}-1\right)-1}\right)}{\log\left(\omega_a/\omega_p + \sqrt{(\omega_a/\omega_p)^2-1}\right)} \approx \frac{\log\left(2.10^{\frac{Aa}{20}}\Big/\sqrt{10^{\frac{Ap}{10}}-1}\right)}{\log\left(\omega_a/\omega_p + \sqrt{(\omega_a/\omega_p)^2-1}\right)}$$

$$arcosh(x) = \ln\left(x+\sqrt{x^2-1}\right) \qquad x \ge 1$$

$$N \ge = \frac{\arg\cosh\left(\sqrt{(10^{\frac{Aa}{10}}-1)/(10^{\frac{Ap}{10}}-1)}\right)}{\arg\cosh(\omega_a/\omega_p)}$$





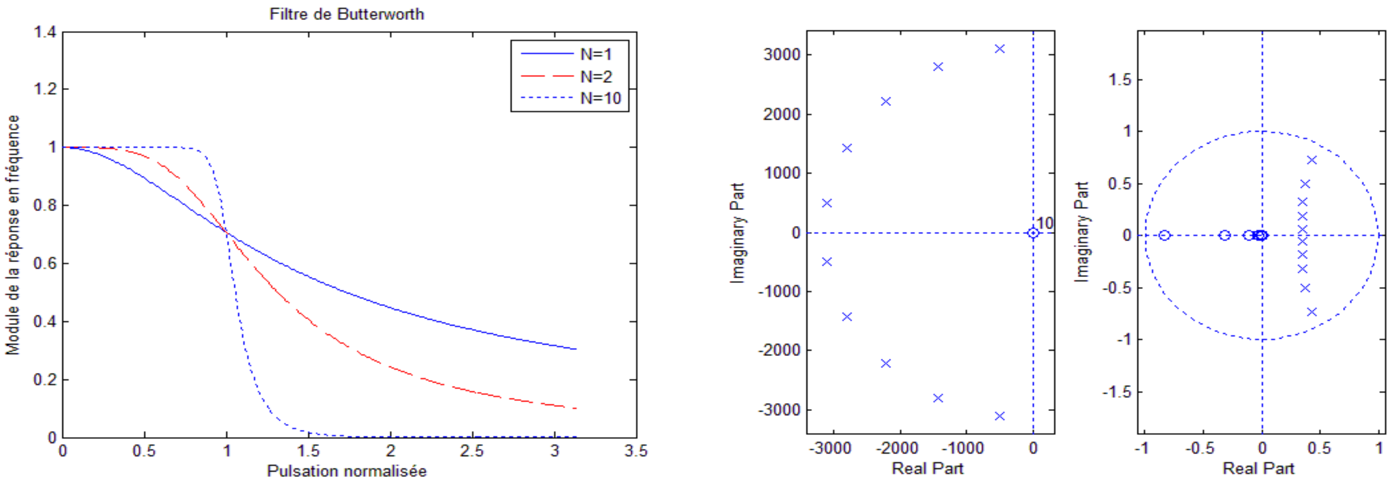
2. RII Filters Design : Indirect Methods

Reminders on analog filters

Butterworth :

$$|H(\omega)| = \frac{1}{\sqrt{1+\left(\dfrac{\omega}{\omega_c}\right)^{2N}}}$$

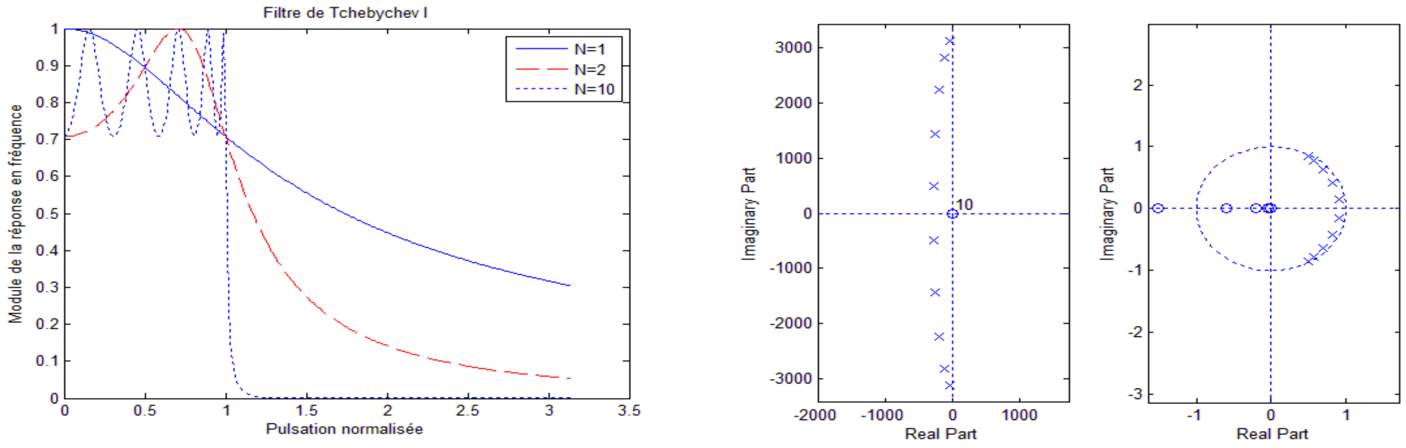
$$|H(\omega)| = \frac{1}{\sqrt{1+\varepsilon^2.\left(\dfrac{\omega}{\omega_c}\right)^{2N}}}$$



Chebyshev :

$$|H(\omega)| = \frac{1}{\sqrt{1+\varepsilon^2.T_N^2\left(\dfrac{\omega}{\omega_c}\right)}}$$

$$T_N(x) = \begin{cases} \cos[N.ar\cos(x)] & pour \quad 0 \le |x| \le 1 \\ \cosh[N.ar\cosh(x)] & pour \quad |x| \ge 1 \end{cases}$$



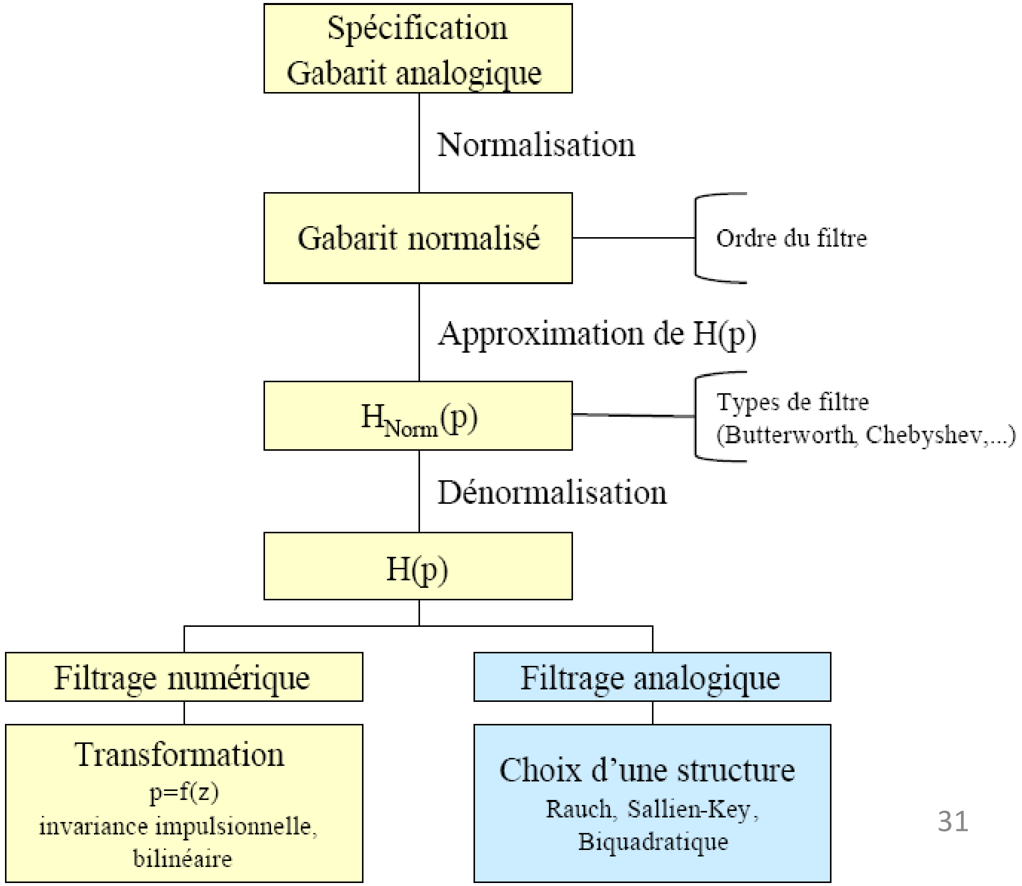
2. RII Filters Design : Indirect Methods

Denormalization

$H_N(p)$ of polynomial analog filters (Butterworth , Chebyshev, Bessel , Cauer , etc.) are given for *normalized cutoff frequencies* and only for *low pass* filters

Low pass $p = p/\omega_A$	Bandpass $p = \frac{1}{B}\left(\frac{p}{\omega_A} + \frac{\omega_A}{p}\right)$
High pass $p = \omega_A / p$	Tape cutter $p = \left[\frac{1}{B}\left(\frac{p}{\omega_A} + \frac{\omega_A}{p}\right)\right]^{-1}$

$$B = (\omega_{A2} - \omega_{A1}) \qquad \omega_A = \sqrt{\omega_{A1}\omega_{A2}}$$



2. RII Filters Design : Bilinear transformation method

Objective : To best match the *analog and digital domains*

Analog filter H(p) → Digital filter *H(z)* with $z = e^{2\pi jfTe} = e^{pTe} \Rightarrow p = Ln(z)/Te$.

Nonlinear relationship between analog frequencies f_A and digital frequencies f_N

→ Approximation of ln(z) to preserve the polynomial character of H(z)

an approximation of ln(z) by Laurent series:

$$\ln(z) \approx 2\left(\left(\frac{1-z^{-1}}{1+z^{-1}}\right) + \frac{1}{3}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)^3 + \frac{1}{5}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)^5 + \dots\dots\dots\right) \Rightarrow \boxed{p = \frac{\ln(z)}{T_e} \approx \frac{2}{T_e}\frac{1-z^{-1}}{1+z^{-1}} = \frac{2}{T_e}\frac{z-1}{z+1}}$$

$$p = j\omega_A = \frac{\ln(z)}{T_e} = \frac{2}{T_e}\frac{e^{j\omega_N T_e}-1}{e^{j\omega_N T_e}+1} = \frac{2}{T_e}\frac{e^{j\omega_N T_e/2}(e^{j\omega_N T_e/2}-e^{-j\omega_N T_e/2})}{e^{j\omega_N T_e/2}(e^{j\omega_N T_e/2}+e^{-j\omega_N T_e/2})} = \frac{2}{T_e}\frac{j\sin(\omega_N T_e/2)}{\cos(\omega_N T_e/2)}$$

$$\Rightarrow \boxed{\omega_A = \frac{2}{T_e}tg\left(\frac{\omega_N T_e}{2}\right) = \frac{2}{T_e}tg\left(\frac{\pi f_N}{f_e}\right)}$$

2. RII Filters Design : Bilinear transformation method

- ✓From the digital filter specifications, calculate the analog pulsation

$$\boxed{\omega_A = \frac{2}{T_e} tg\left(\frac{\omega_N T_e}{2}\right) = \frac{2}{T_e} tg\left(\frac{\pi f_N}{f_e}\right)}$$

- ✓Determine the template of the analog filter H_N (p) normalized to order *n.*

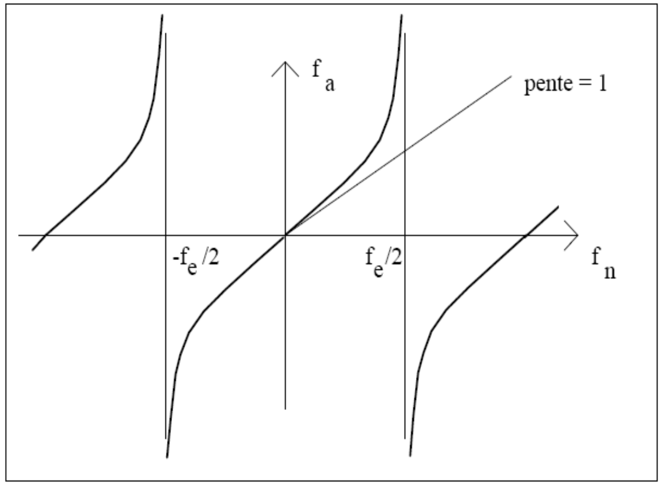
- ✓ Denormalize the filter according to ω_A → H(p)

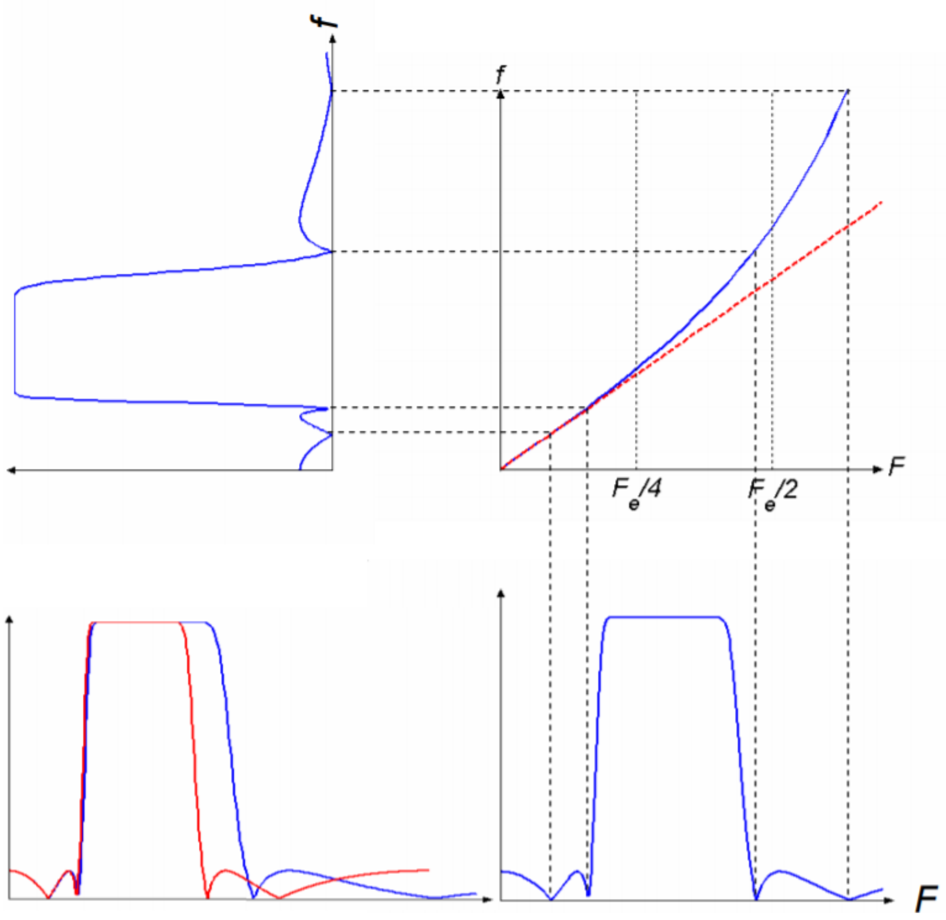
- ✓H(p)→H(z)=H(p) where $\boxed{p = \frac{\ln(z)}{T_e} \approx \frac{2}{T_e} \frac{1 - z^{-1}}{1 + z^{-1}} = \frac{2}{T_e} \frac{z - 1}{z + 1}}$

2. RII Filters Design : Bilinear transformation method

***Note** :* Distortion on the frequency axis

$$\omega_A = \frac{2}{T_e} tg\left(\frac{\omega_N T_e}{2}\right) = \frac{2}{T_e} tg\left(\frac{\pi f_N}{f_e}\right)$$





https://perso.esiee.fr/~venardo/doc/FiltresNumeriques.pdf

2. RII Filters Design : Bilinear transformation method

Example 1 : We want to design a first-order digital low-pass filter from a transfer function of an RC filter in the continuous domain ($H_N(p) = 1/(1 + p)$). The desired cutoff frequency is f_N = 30 Hz and the sampling frequency is f_e = 150 Hz

✓ Calculation of $\boxed{\omega_A = \frac{2}{T_e} tg\left(\frac{\omega_N T_e}{2}\right) = \frac{2}{T_e} tg\left(\frac{\pi f_N}{f_e}\right)}$ ➡ $\omega_A = \frac{2}{T_e} tg\left(\frac{\pi}{5}\right) = \frac{2}{T_e} 0.7265$

✓ Denormalize the filter according to ω_A →

$$H(p) = \left.\frac{1}{(1+p)}\right|_{p=p/\omega_A} = \frac{\omega_A}{p+\omega_A} = \frac{\frac{2}{T_e}0.7265}{p+\frac{2}{T_e}0.7265}$$

✓ Determine

$$H(z) = H(p)\Big|_{p=\frac{2}{T_e}\frac{z-1}{z+1}} = \frac{\frac{2}{T_e}0.7265}{\frac{2}{T_e}\frac{z-1}{z+1}+\frac{2}{T_e}0.7265} = \frac{0.7265(z+1)}{(z-1)+0.7265(z+1)} = \frac{0.7265(z+1)}{1.7265z-0.2735}$$

Noticed

$$f_A = \frac{2}{T_e(2\pi)} tg\left(\frac{\pi}{5}\right) = \frac{f_e}{\pi} 0.7265 = 34.69 \neq f_N = 30$$

2. RII Filters Design : Bilinear transformation method

Example 2 : We want to create a second-order low-pass digital filter with the following characteristics: cut-off frequency f_N = 500 Hz (gain of 1), sampling frequency f_e = 5 kHz,

- ✓Calculation of

$$\omega_A = \frac{2}{T_e} tg\left(\frac{\omega_N T_e}{2}\right) = \frac{2}{T_e} tg\left(\frac{\pi}{10}\right) = \frac{2}{T_e} 0.325$$

- ✓ Denormalize

$$H_N(p) = \frac{0.1075}{p^2 + 0.1075p + 1} \Rightarrow H(p) = H_N(p = p/\omega_A) = \frac{0.1075\omega_A{}^2}{p^2 + 0.1075p\omega_A + \omega_A{}^2} = \frac{\frac{4}{T_e^2}0.0114}{p^2 + \frac{2}{T_e}0.035p + \frac{4}{T_e^2}0.1056}$$

- ✓Determine

$$H(z) = H(p)\Big|_{p=\frac{2}{T_e}\frac{z-1}{z+1}} = \frac{\frac{4}{T_e^2}0.0114}{\left(\frac{2}{T_e}\frac{z-1}{z+1}\right)^2 + \frac{2}{T_e}0.035\frac{2}{T_e}\frac{z-1}{z+1} + \frac{4}{T_e^2}0.1056} = \frac{0.0114}{\left(\frac{z-1}{z+1}\right)^2 + 0.035\frac{z-1}{z+1} + 0.1056}$$

$$H(z) = \frac{0.0114(z+1)^2}{(z-1)^2 + 0.035(z-1)(z+1) + 0.1056(z+1)^2} = \frac{0.0114(z+1)^2}{1.14z^2 - 1.7888z + 1.0706}$$

3. Multi-rate filtering

Example : Synthesize a low-pass filter with the following specifications: *fp* =100Hz *fa* =300Hz *Aa* > 50 db *fe* =20kHz.

Hamming $\Rightarrow$ **N=331**

Windows w $_N$ (n)	Width of Transition Δ f = (2 Δ f / fe)	Attenuation in attenuated band Aa
$w_{Rect}(n)=\begin{cases}1 & pour\ \lvert n\rvert \le \frac{N-1}{2}\\ 0 & ailleurs\end{cases}$	1.8/N	21
$w_{Han}(n)=\begin{cases}0.5+0.5\cos(\frac{2\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2}\\ 0 & ailleurs\end{cases}$	6.2/N	44
$w_{Ham}(n)=\begin{cases}0.54+0.46\cos(\frac{2\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2}\\ 0 & ailleurs\end{cases}$	6.6/N	53
$w_{Black}(n)=\begin{cases}0.42+0.5\cos(\frac{2\pi n}{N-1})+0.08\cos(\frac{4\pi n}{N-1}) & pour\ \lvert n\rvert \le \frac{N-1}{2}\\ 0 & ailleurs\end{cases}$	11/N	74

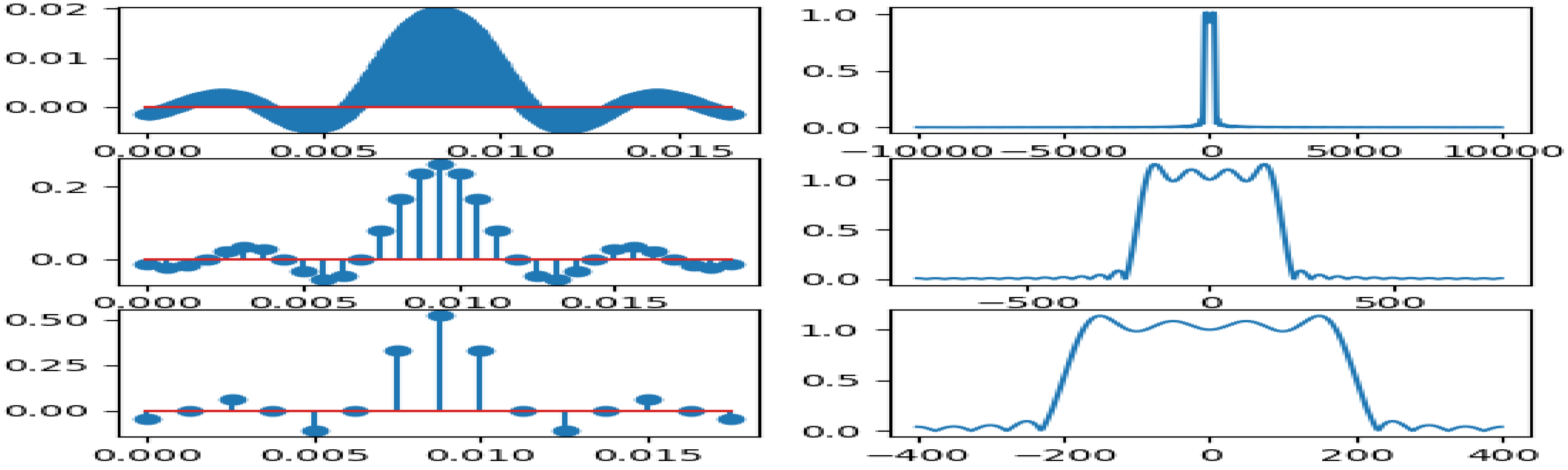
Filter occupies a small portion of the bandwidth (low pass with *fc* = 1% *(fe)* .

Downsample and then upsample without loss of information.

$\Rightarrow$ Take *fe '= fe /25* =800Hz $\Rightarrow$ N=15 << 331 $\Rightarrow$ Faster filter.

3. Multi-rate filtering

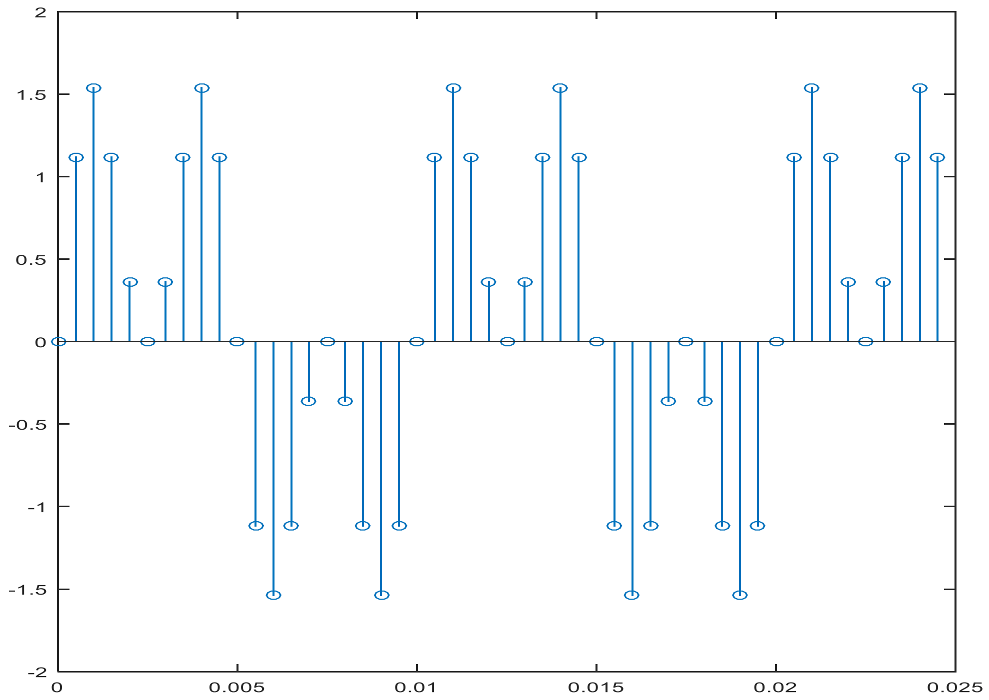
Example : Synthesize a low-pass filter with the following specifications: *fp* =100Hz *fa* =300Hz *Aa* $>$ 50 db *fe* =20kHz.

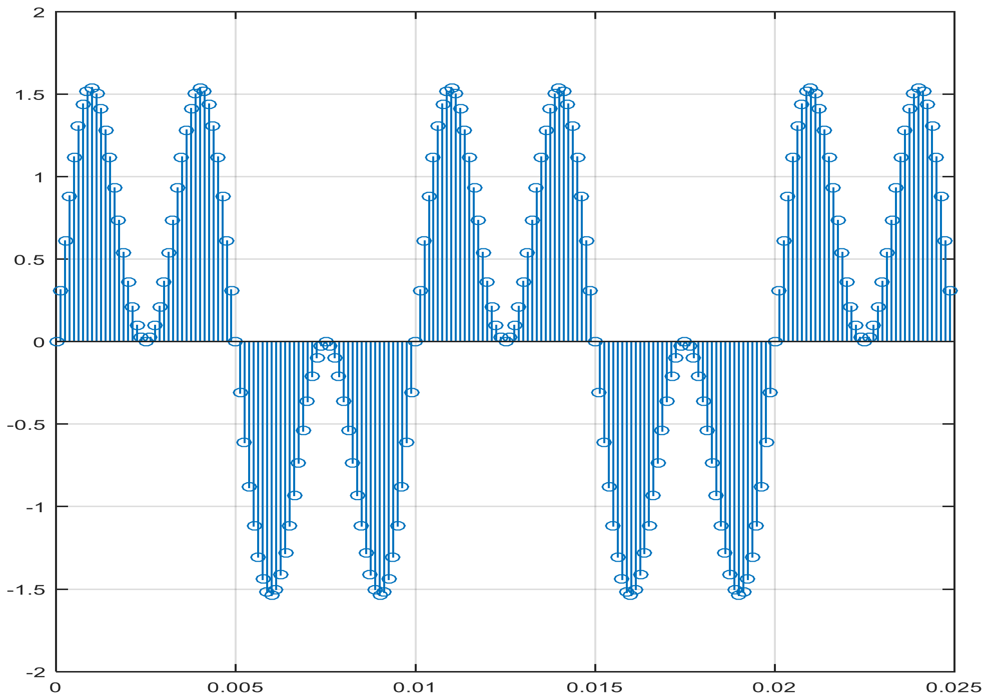


- Principle of filter banks: Divide the bandwidth into several narrow-band filters, therefore filters that can be easily decimated $\Rightarrow$ a time saving.

3. Multi-rate filtering

Example : Interpolation before digital-to-analog conversion





3. Multi-rate filtering

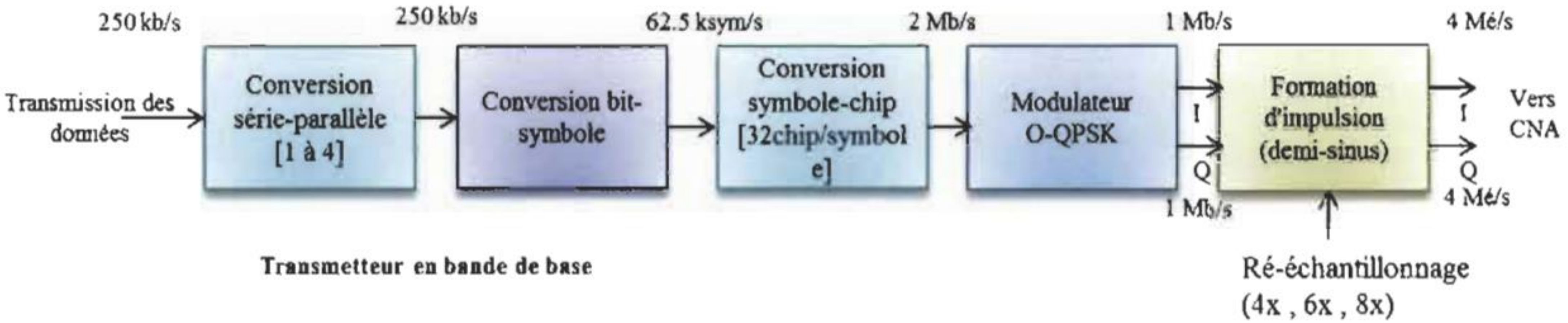
Examples of multi-cadence systems

- Audio applications where multiple sampling frequencies coexist: 32kHz (broadcast); 44.1kHz (digital compact disc); 48kHz (digital soundtrack); 96kHz
- Processing chain corresponding to the baseband transmitter of a QPSK modulator where there are three processing frequencies: the bit frequency, the symbol frequency and the sampling frequency

3. Multi-rate filtering

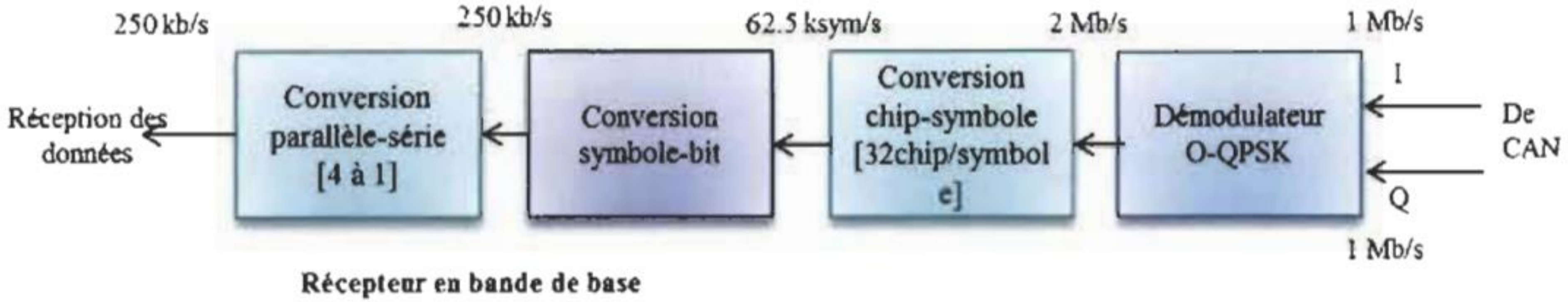
Examples of multi-cadence systems

-



-

:y



3. Multi-rate filtering

- Low-pass filtering $\Rightarrow f_{max}$ decreases
- Modulation $\Rightarrow f_{max}$ increases

Interest in changing frequency

✓Adjust the sampling frequency to minimize computation time.

✓Increase SNR before CNA

✓Changing frequencies

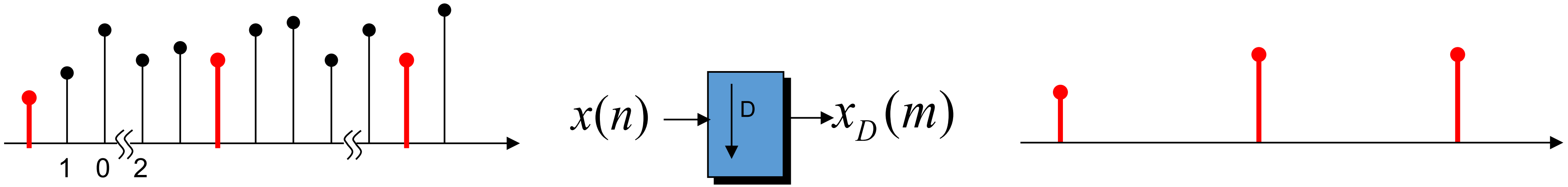
✓Filter Banks

How ??

1. Decimation or downsampling (decrease of f_{max})
2. Interpolation or oversampling (increasing f_{max}).

3.1. **Downsampling** and Upsampling

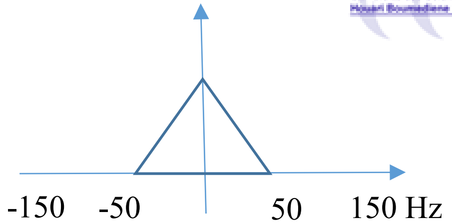
- Make from the original signal x a signal x_D having fewer samples than the original signal $\Rightarrow$ “removing” samples $\Rightarrow$ **decimation** (down sampling)



- $T_e' = DT_e \Rightarrow f_e' = f_e/D$
- Downsampling D consists of getting rid of D - 1 samples out of D
- Sample directly at f_e/D, but there is no guarantee that f_e/D >2 f_{max} . So there is a risk of spectral aliasing.

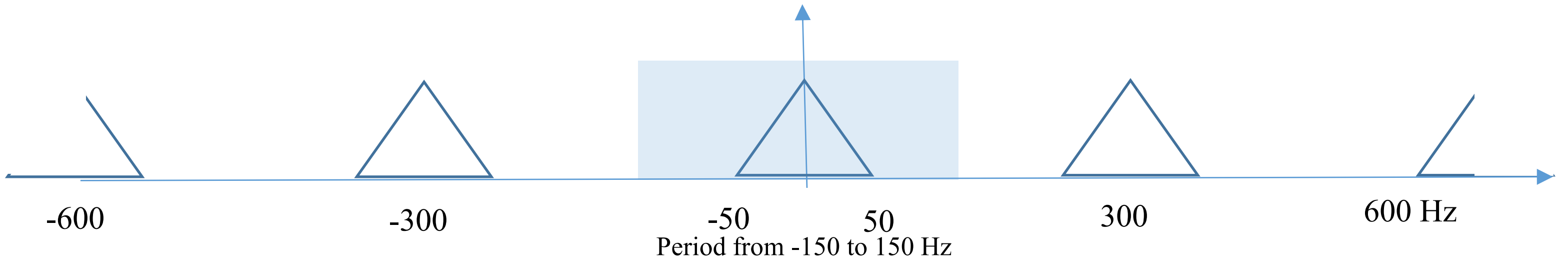
$$X_D(z) = \sum_{-\infty}^{+\infty} x_D(m) z^{-m} = \sum_{-\infty}^{+\infty} x(n/D) z^{-n/D}$$

3.1. **Downsampling** and Upsampling

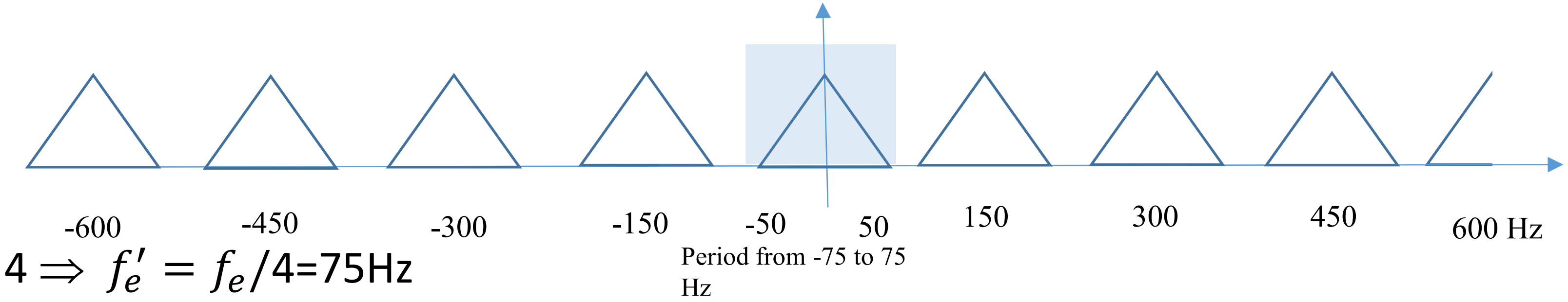


Reminders: Example of a frequency response of the signal decimated by 2 then by 4

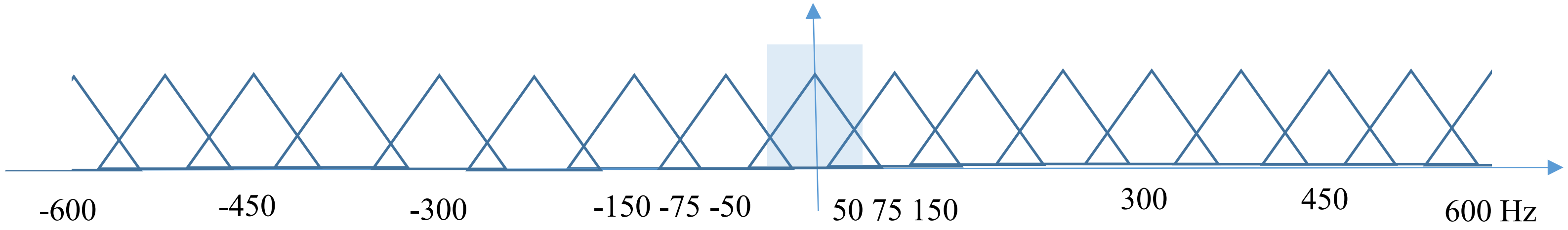
- Original signal



- Decimation by 2 $\Rightarrow$ $f_e' = f_e/2$=150Hz



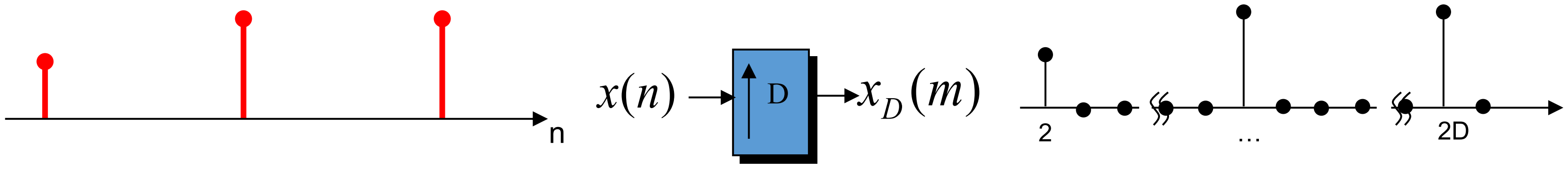
- Decimation by 4 $\Rightarrow$ $f_e' = f_e/4$=75Hz

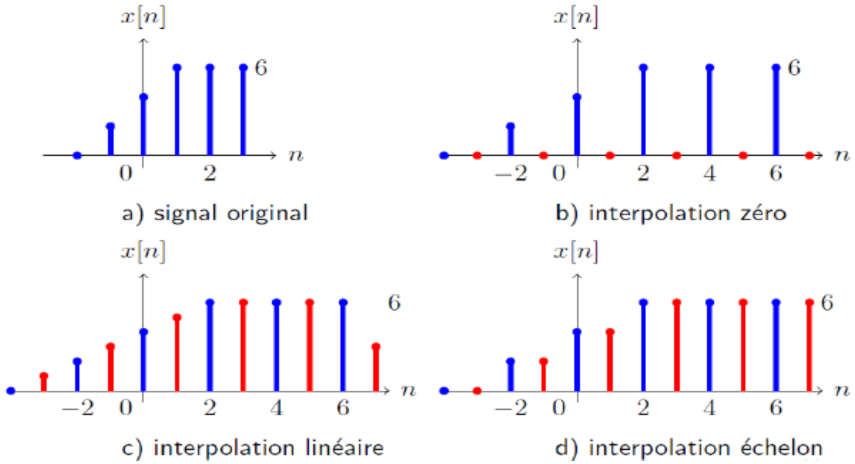


Solution: Before decimating, remove the frequency content of x(n) after $f_e/2D$ and f_e/D since the spectrum becomes periodic with period f_e/D

3.1. Downsampling and **Upsampling**

- Make from the original signal x a signal x_M a signal of the same duration but containing more samples than the original signal $\Rightarrow$ "adding" samples $\Rightarrow$ **interpolation** (upsampling).

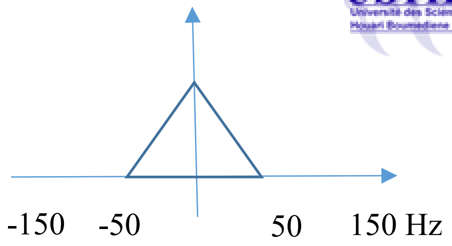




- $T_e' = T_e/D \Rightarrow f_e' = D\,f_e$
- Oversampling of D $\Rightarrow$ Add D - 1 points
- Original signal has no frequency content beyond $f_e/2$

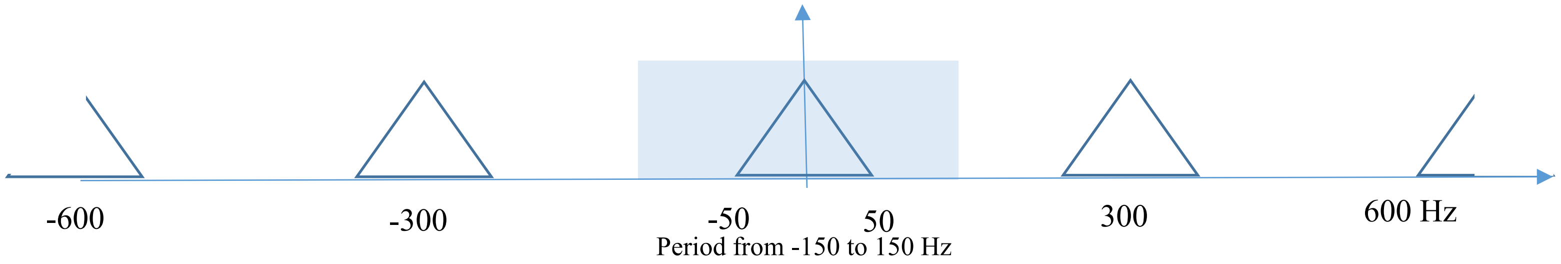
$$X_D(z) = \sum_{-\infty}^{+\infty} x_D(m) z^{-m} = \sum_{-\infty}^{+\infty} x_D(nD) z^{-nD} = \sum_{-\infty}^{+\infty} x(n) (z^{-n})^D = X(z^D)$$

3.1. Downsampling and **Upsampling**

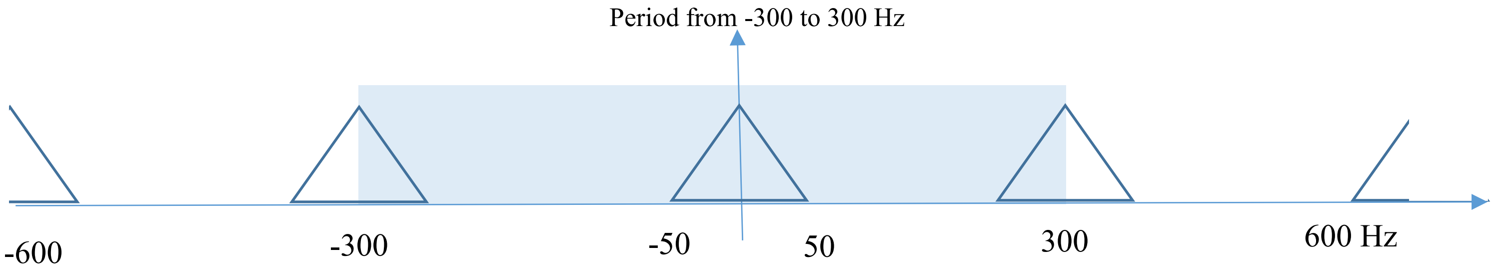


Reminders: Example of a frequency response of the signal interpolated by 2 then by 4

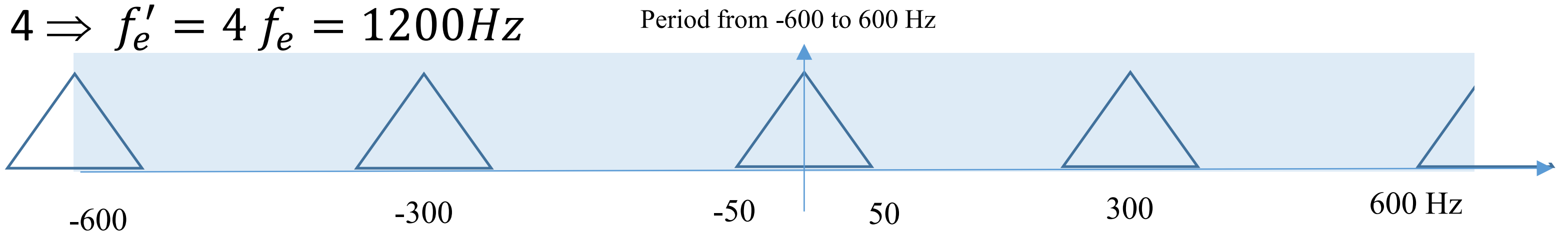
- Original signal



- Interpolation by 2 $\Rightarrow\ f_e' = 2\ f_e$=600Hz



- Interpolation by 4 $\Rightarrow\ f_e' = 4\ f_e = 1200Hz$



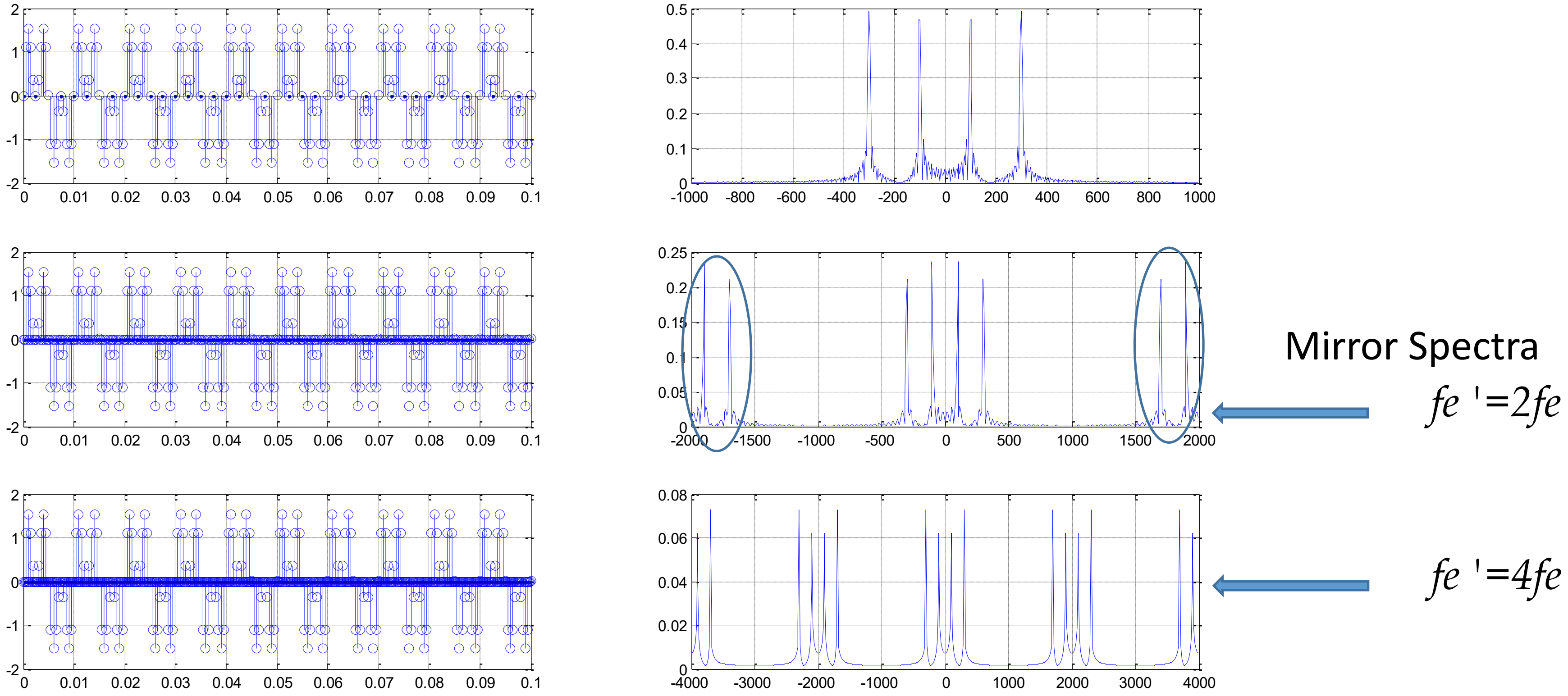
Same spectrum shape but sampling frequency of $\boldsymbol{D\, f_e}$ $\Rightarrow$

Solution

Remove mirror spectra by low-pass filtering from $f_e/2$

3.1. Downsampling and **Upsampling**

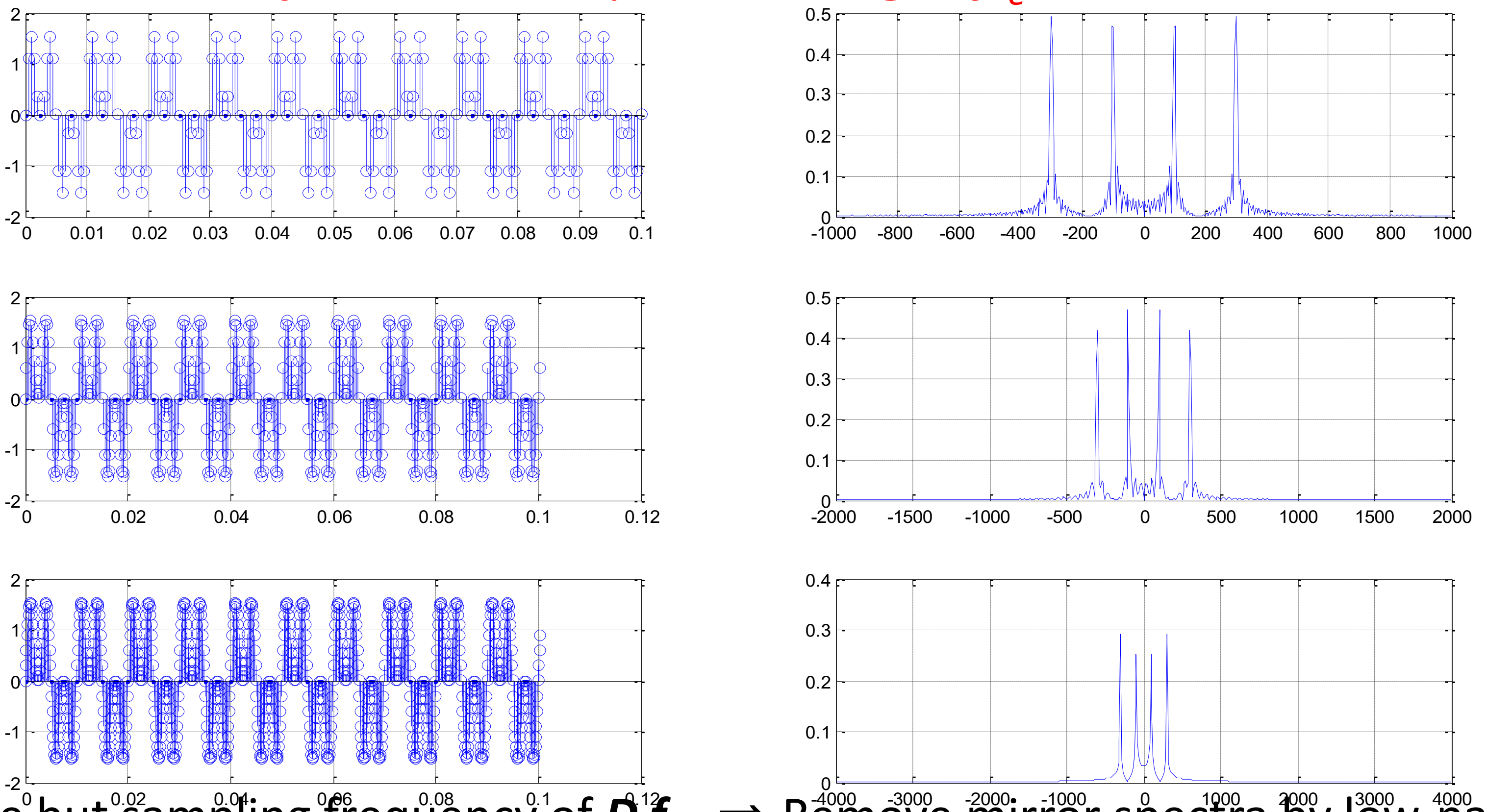
Python example: Signal = sum of 2 sinusoids of frequencies 100 and 300Hz. We interpolate it by 2 then by 4.



Same spectrum shape but sampling frequency of $\boldsymbol{D f_e}$ $\Rightarrow$ Remove mirror spectra by low-pass filtering from $f_e/2$

3.1. Downsampling and **Upsampling**

Python example: Signal = sum of 2 sinusoids of frequencies 100 and 300Hz. We interpolate it by 2 then by 4. **Solution: After interpolation,** low-pass filtering at $f_e/2$



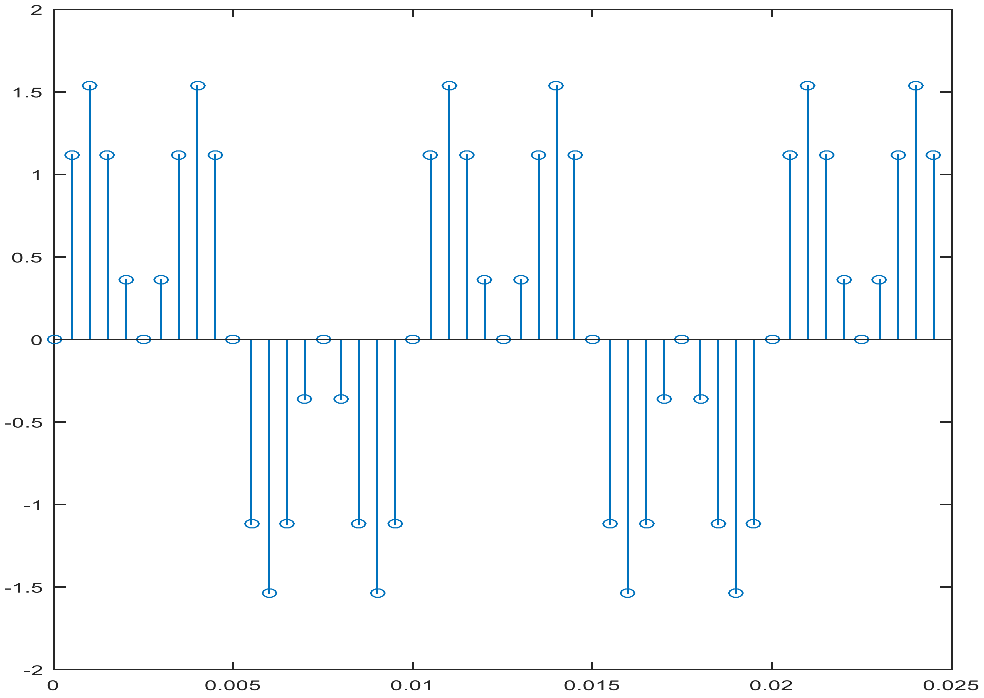
$fe' = 2fe$

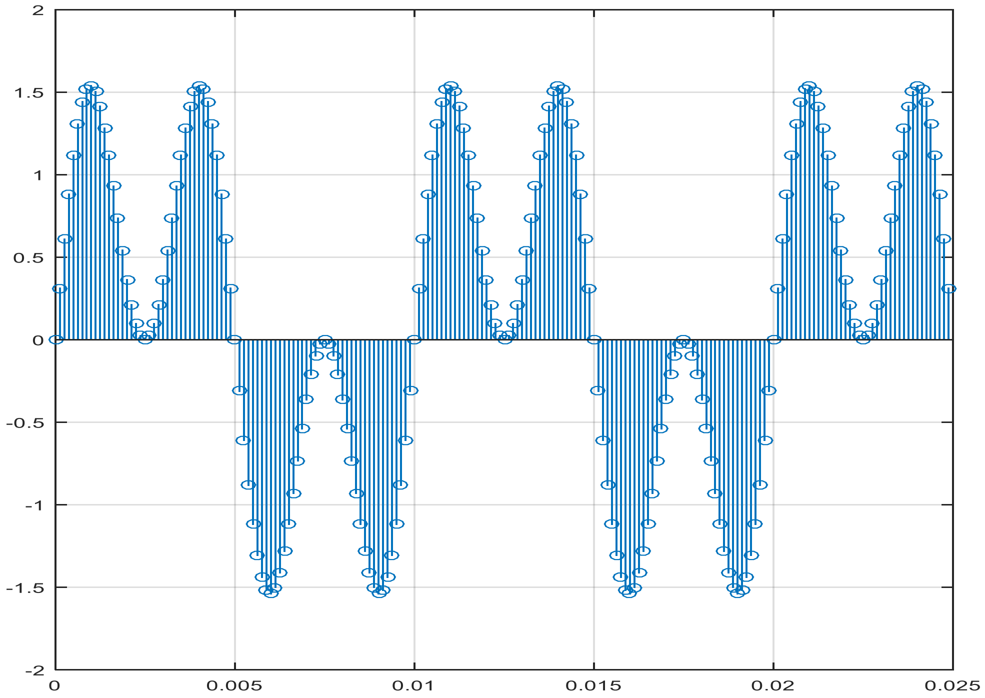
$fe' = 4fe$

Same spectrum shape but sampling frequency of $D f_e$ $\Rightarrow$ Remove mirror spectra by low-pass filtering from $f_e/2$

3.1. Downsampling and **Upsampling**

Python example: Signal = sum of 2 sinusoids of frequencies 100 and 300Hz. We interpolate it by 2 then by 4. **Solution: After interpolation,** low-pass filtering at $f_e/2$

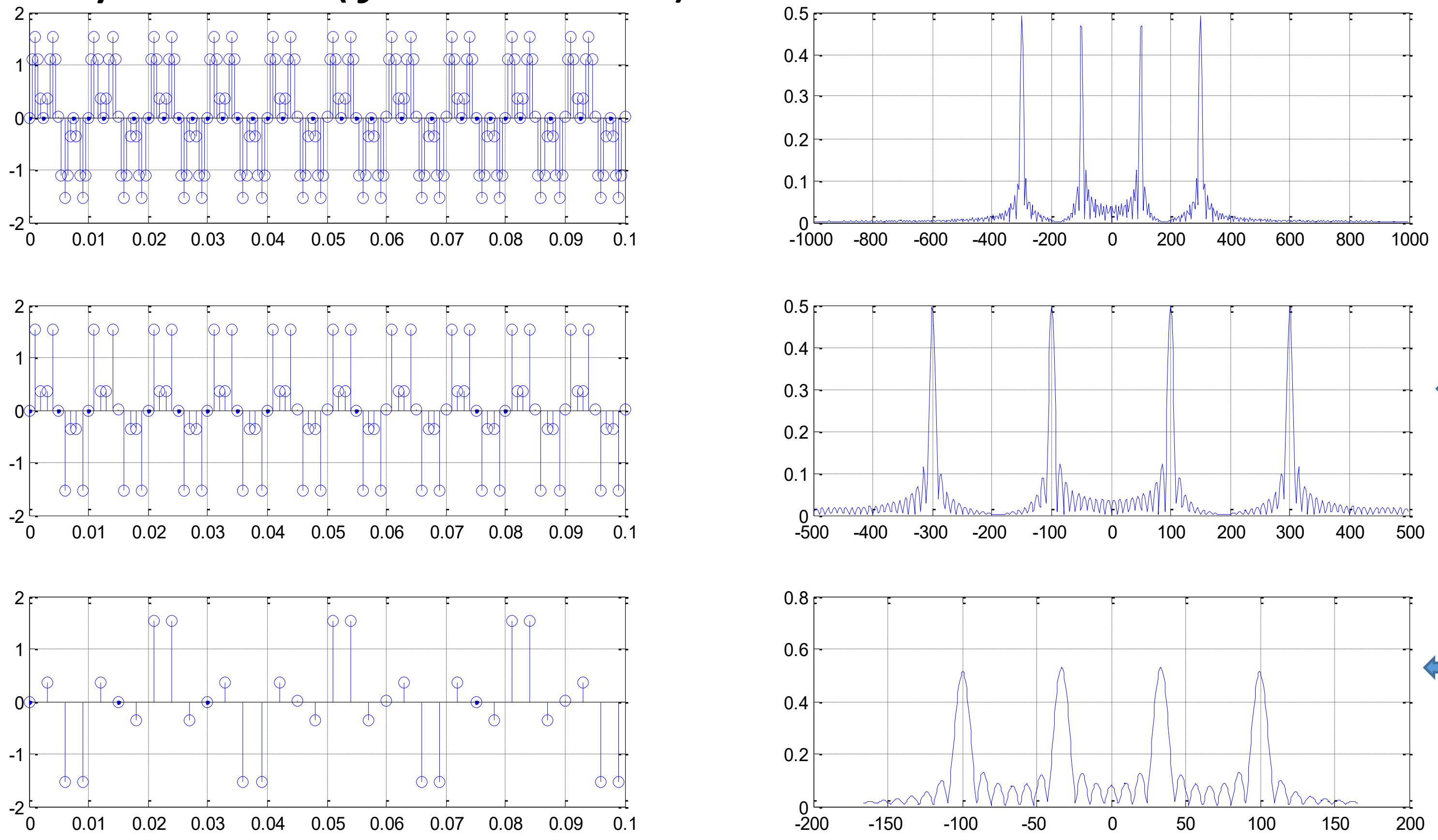




3.1. **Downsampling** and Upsampling

Python example: Signal = sum of 2 sinusoids of frequencies 100 and 300Hz.

It is decimated by 2 and 6 (*fe* = 2000 Hz)



$fe\,' = fe/2$

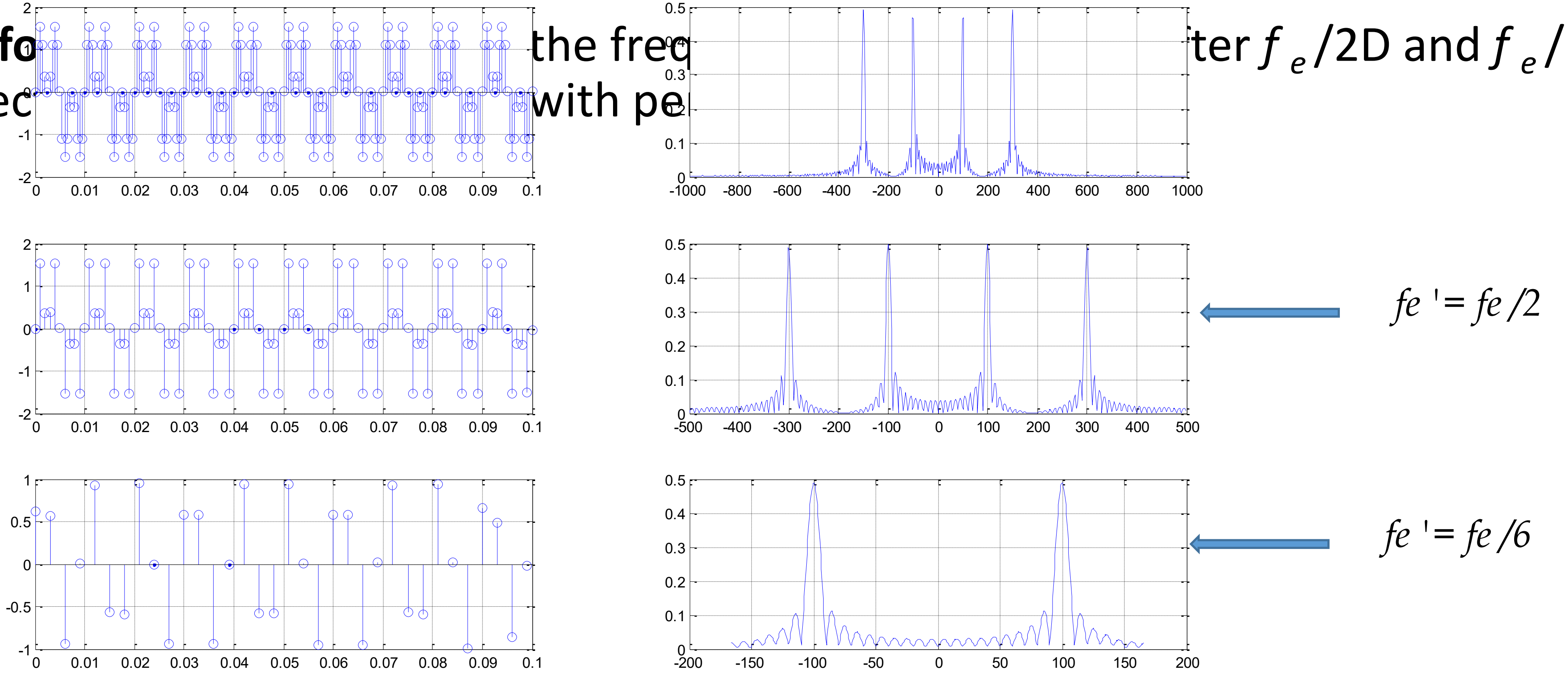
$fe\,' = fe/6$

3.1. **Downsampling** and Upsampling

Python example: Signal = sum of 2 sinusoids of frequencies 100 and 300Hz.

It is decimated by 2 and 6 (*fe* = 2000 Hz).

Solution: Befo … the freq … fter $f_e/2D$ and f_e/D
since the spec … with pe

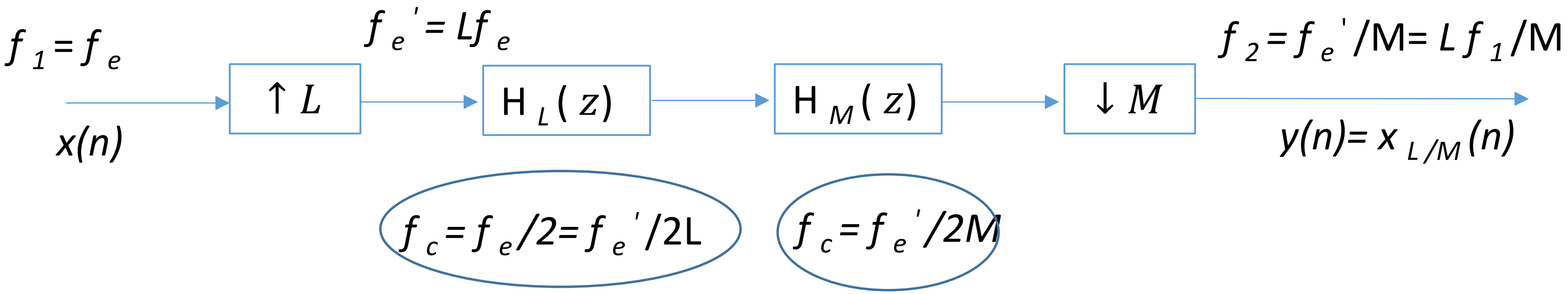


3.2. Rational Frequency Change

Digitally change the sampling rate of a signal x(n) from a sampling rate f_1 to a sampling rate f_2 where $f_2/f_1 = L/M$.

Oversample the signal x(n) by a factor L and then decimate it by a factor M.

- Oversampling by L must be followed by a low-pass filter cutting at $f'_e/2L$ ($f_e/2$) to remove spectra due to oversampling where f_e is the sampling frequency of the input signal.
- The decimation by M must be preceded by an anti-aliasing filtering having a cutoff frequency of $f_e'/2M$, if f_e' is the sampling frequency of the input signal.



- Keep only one of the two low-pass filters: smaller f_c or (max(L,M))

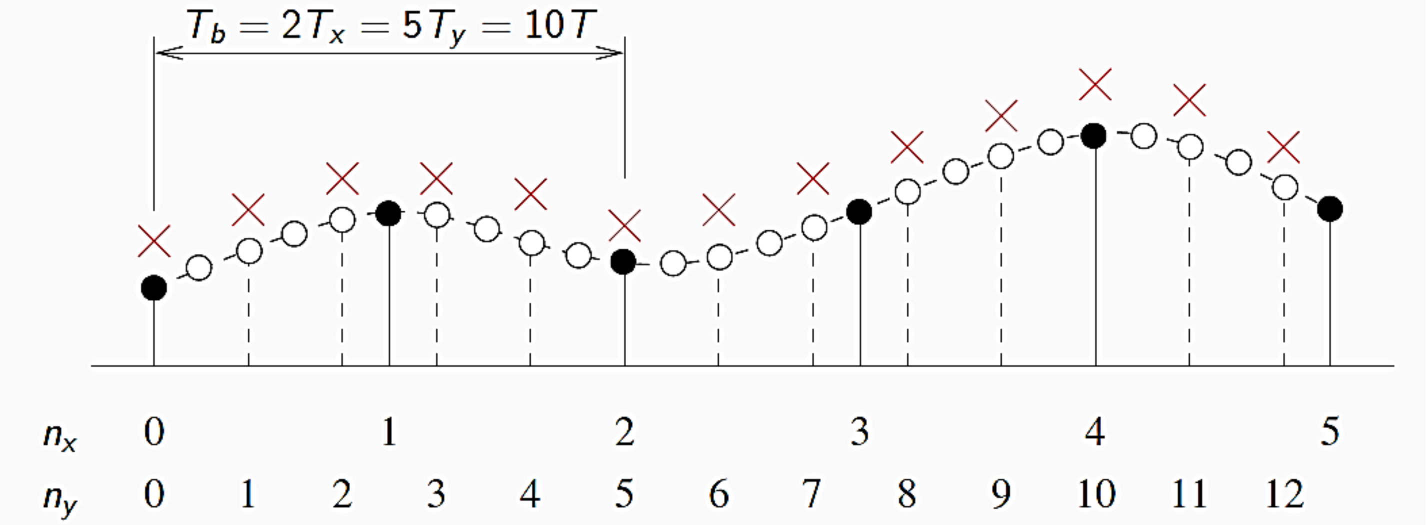
3.2. Rational Frequency Change

Digitally change the sampling rate of a signal x(n) by one frequency sampling rate f_1 at a sampling frequency f_2 where $f_2/f_1 = L/M$.

Oversample the signal x(n) by a factor L and then decimate it by a factor M.

- If Let M are coprime, decimation and interpolation are commutative

Example: Conversion where L = 5 and M = 2



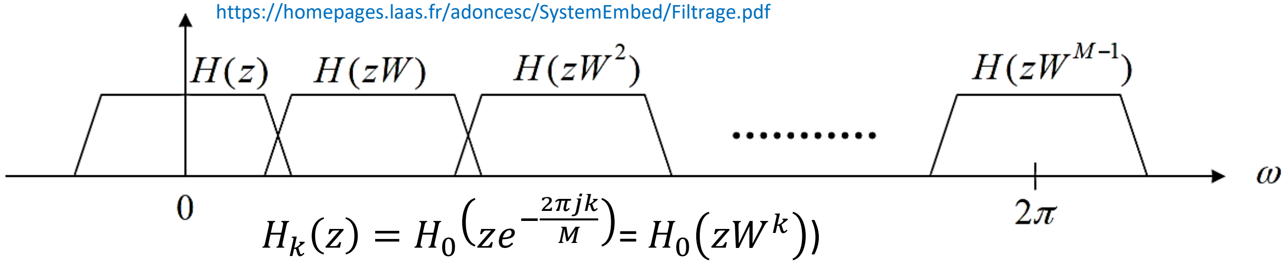
https://www.fsg.ulaval.ca/departements/professeurs/paul-fortier-111/

3.3. Filter Banks

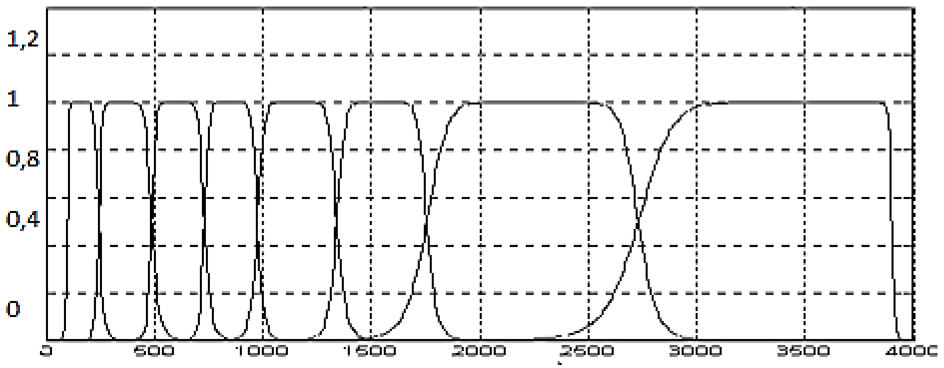
A filter bank is a set of filters, with a common input or output.

A filter bank is a set of filters ($H_0(z)$;….;$H_{M-1}(z)$) such that the filter bandwidths form (+/-) a partition of the set of frequencies and their frequency responses are deduced from each other by certain relations

https://homepages.laas.fr/adoncesc/SystemEmbed/Filtrage.pdf



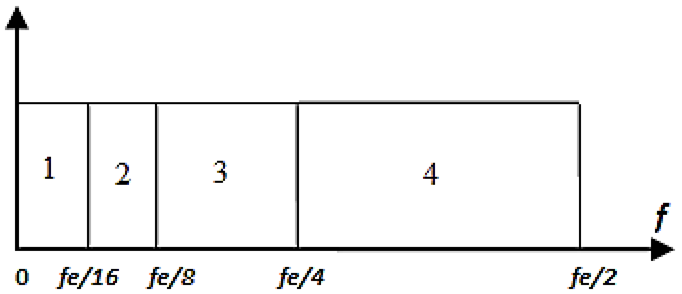
$$H_k(z) = H_0\left(ze^{-\frac{2\pi jk}{M}}\right) = H_0\left(zW^k\right)$$



Other decompositions: Mel scale (psychoacoustics), Wavelets etc.

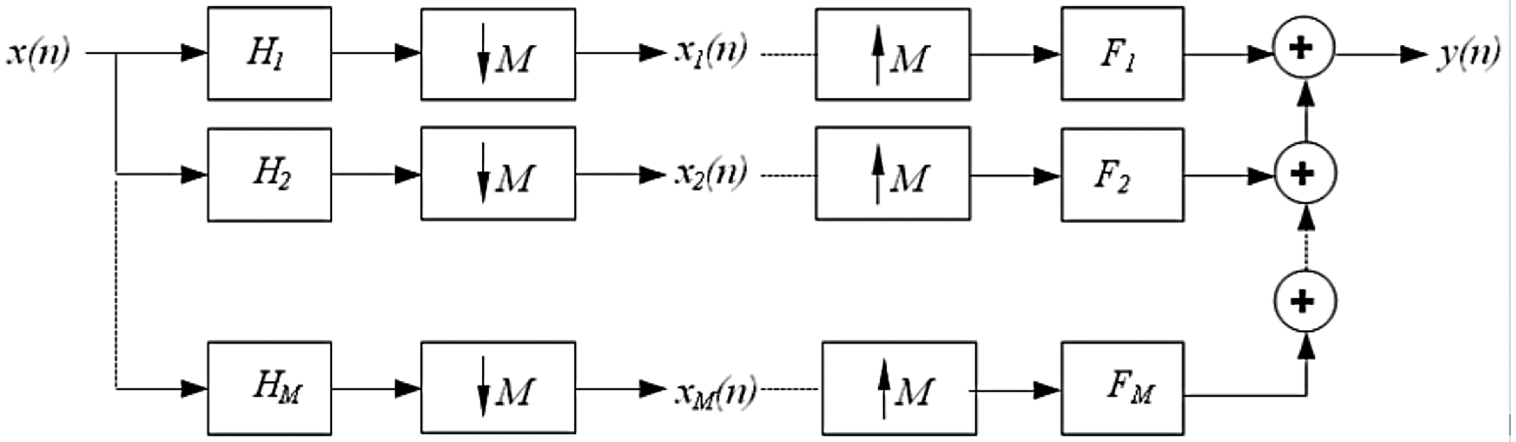
Each output of a bank's filter is narrowband ,

$\Rightarrow$ we can therefore decimate it to treat it more quickly.



3.3. Filter Banks

The bench decomposes the signal x(n) into M subband signals (number of channels M = M decimation → critical sampling)



- ✓ $H_k(z)$ are the analysis (decomposition) filters
- ✓ $F_k(z)$ are the synthesis (reconstruction) filters

Perfect reconstruction when in the absence of any processing in the sub-bands the output signal y(n)=x(nk).

But the analysis and synthesis filters are imperfect ⇒ broadening of the sub-bands ⇒ aliasing at the outputs of the decimators.

Find the filters to compensate for the effects of aliasing to have perfect reconstruction.

3.3. Filter Banks

Reminders:

- Dirac comb: $\sum_{n=-\infty}^{\infty}\delta(t-nT_e)$periodic with period $T_e \Rightarrow$ decomposable into Fourier series such that

$C_n=\frac{1}{T_e} \Rightarrow \sum_{n=-\infty}^{\infty}\delta(t-nT_e)=\frac{1}{T_e}\sum_{n=-\infty}^{\infty}e^{2\pi jn\frac{t}{T_e}}$ **In discrete** $\rightarrow \sum\delta(n-mM)=\frac{1}{M}\sum_{k=0}^{M-1}e^{2\pi jk\frac{n}{M}}$

Decimation of M equivalent to sampling of M $\Rightarrow y(m)=y(n/M)=\sum h(n)\delta(n-kM)$

- Transfer function$Y(z)=\sum_{n=-\infty}^{\infty}\left(h(n)\frac{1}{M}\sum_{k=0}^{M-1}e^{\frac{2\pi jkn}{M}}\right)z^{-n/M}=\frac{1}{M}\sum_{k=0}^{M-1}\left(\sum_{n=-\infty}^{\infty}h(n)\left(e^{-\frac{2\pi jk}{M}}z^{\frac{1}{M}}\right)^{-n}\right)$

$Y(z)=\frac{1}{M}\sum_{k=0}^{M-1}H\left(W_M^k z^{\frac{1}{M}}\right)$ with$W_M=e^{-\frac{2\pi j}{M}}$

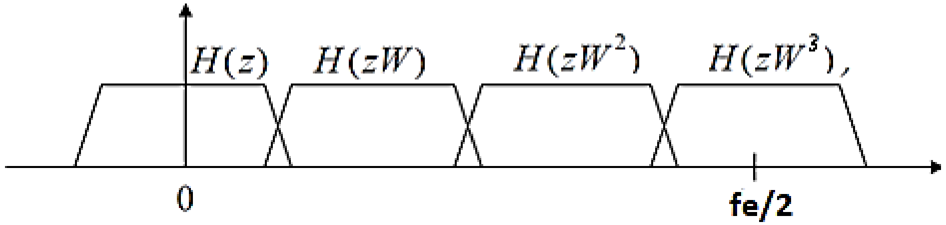
- Frequency response $\Rightarrow Y(f)=\frac{1}{M}\sum_{k=0}^{M-1}H\left(f-\frac{kf_e}{M}\right)\Rightarrow$ ***Periodizationde*** $\boldsymbol{\frac{f_e}{M}}$

Interpolation M $\Rightarrow Y(z)=\sum_{-\infty}^{+\infty}y(k)z^{-k}=\sum_{-\infty}^{+\infty}h(nM)z^{-nM}=\sum_{-\infty}^{+\infty}h(n)(z^{-n})^M=H(z^M)$

3.3. Filter Banks

Examples:

Decimation of 4

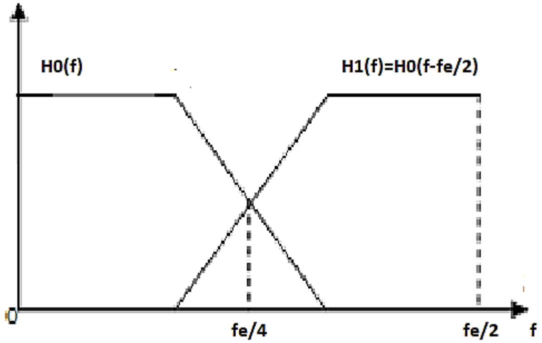


$Y(z) = \frac{1}{4}\sum_{k=0}^{3} H\left(W_4^k z^{\frac{1}{4}}\right) \text{with} W_4 = e^{-\frac{2\pi j}{4}}$

$Y(z) = \frac{1}{4}\left(H\left(W_4^0 z^{\frac{1}{4}}\right) + H\left(W_4^1 z^{\frac{1}{4}}\right) + H\left(W_4^2 z^{\frac{1}{4}}\right) + H\left(W_4^3 z^{\frac{1}{4}}\right)\right)$

$Y(z) = \frac{1}{4}\left(H\left(z^{\frac{1}{4}}\right) + H\left(-jz^{\frac{1}{4}}\right) + H\left(-z^{\frac{1}{4}}\right) + H\left(jz^{\frac{1}{4}}\right)\right) \Rightarrow \mathrm{Y(f)} = \frac{1}{4}\sum_{k=0}^{3} H_0\left(f - \frac{kf_e}{4}\right)$

Interpolation ofe 4 $\Rightarrow Y(z) = X(z^4)$



Decimation of 2

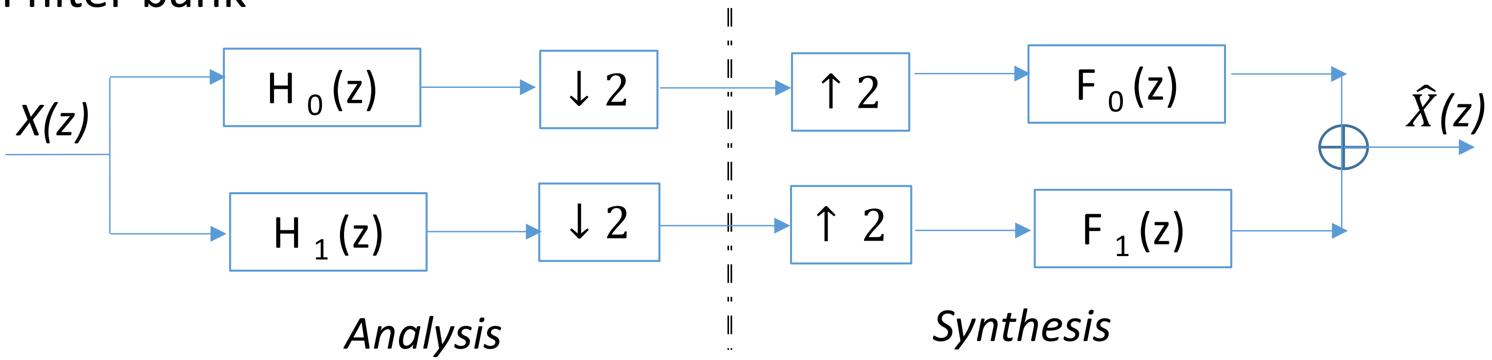
$Y(z) = \frac{1}{2}\sum_{k=0}^{1} H\left(W_2^k z^{\frac{1}{2}}\right) \text{with} W_2 = e^{-\frac{2\pi j}{2}}$

$Y(z) = \frac{1}{2}\left(H\left(z^{\frac{1}{2}}\right) + H\left(-z^{\frac{1}{2}}\right)\right) = \frac{1}{2}\left(H_0(z) + H_1(z)\right) = \frac{1}{2}\left(H_0(z) + H_0((-z)\right) \Rightarrow \mathrm{Y(f)} = \frac{1}{2}\left[H_0(f) + H_0\left(f - \frac{f_e}{2}\right)\right]$

Interpolation ofe 2 $\Rightarrow Y(z) = H(z^2)$

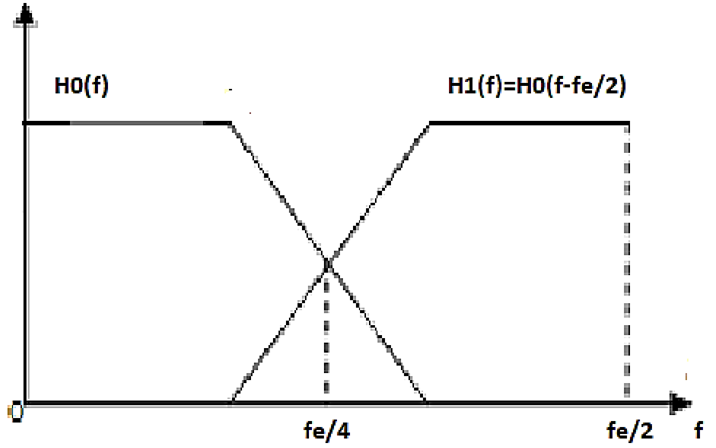
3.3. Filter Banks

Example : 2-channel filter bank



✓ $H_0(z)$ and $H_1(z)$ the low-pass and high-pass filters of the **analysis bench**

✓ $F_0(z)$ and $F_1(z)$ the low-pass and high-pass filters of the **synthesis bench**

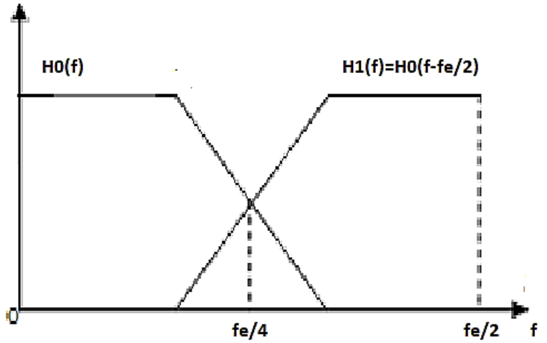


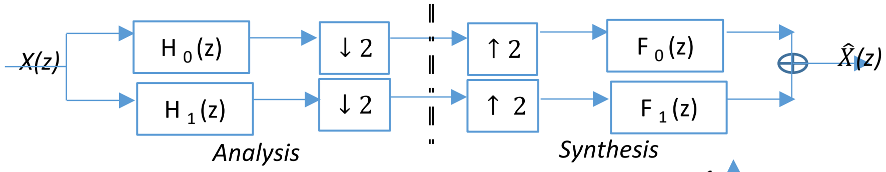
Critical sampling

$H_1(z) = H_0(-z) \Rightarrow H_1(f) = H_0(f - \frac{f_e}{2})$

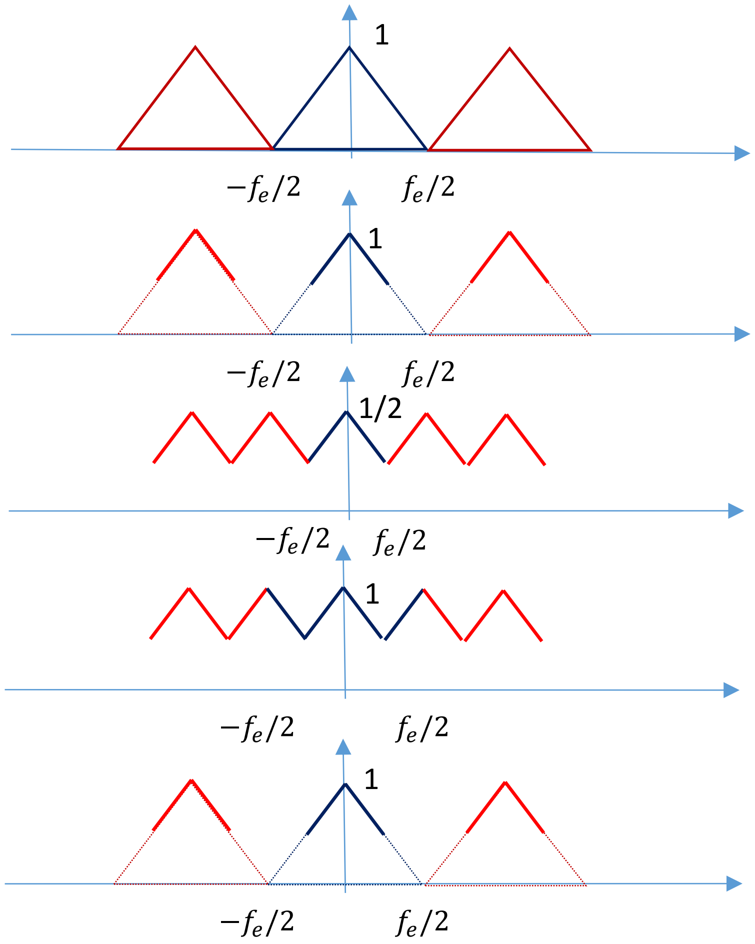
$H_1(z) = H_0(-z^{-1}) \Rightarrow H_1(f) = H_0(\frac{f_e}{2} - f)$

3.3. Filter Banks





Example : 2-channel filter bank

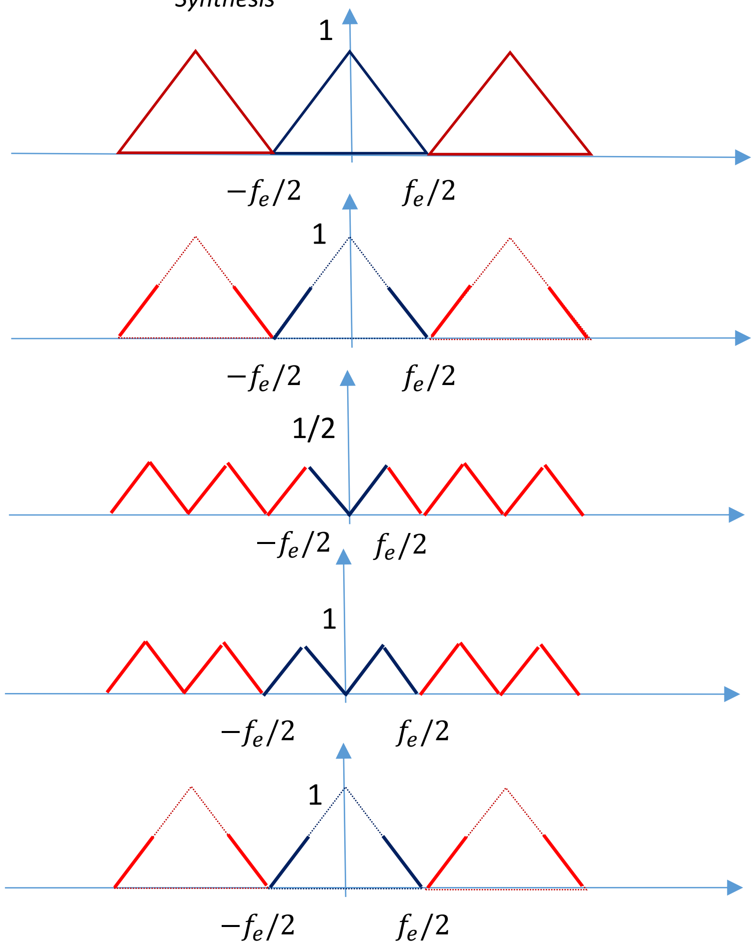


P-low$H_0(z)$

↓ 2

↑ 2

P-low$F_0(z)$



P-high$H_1(z)$

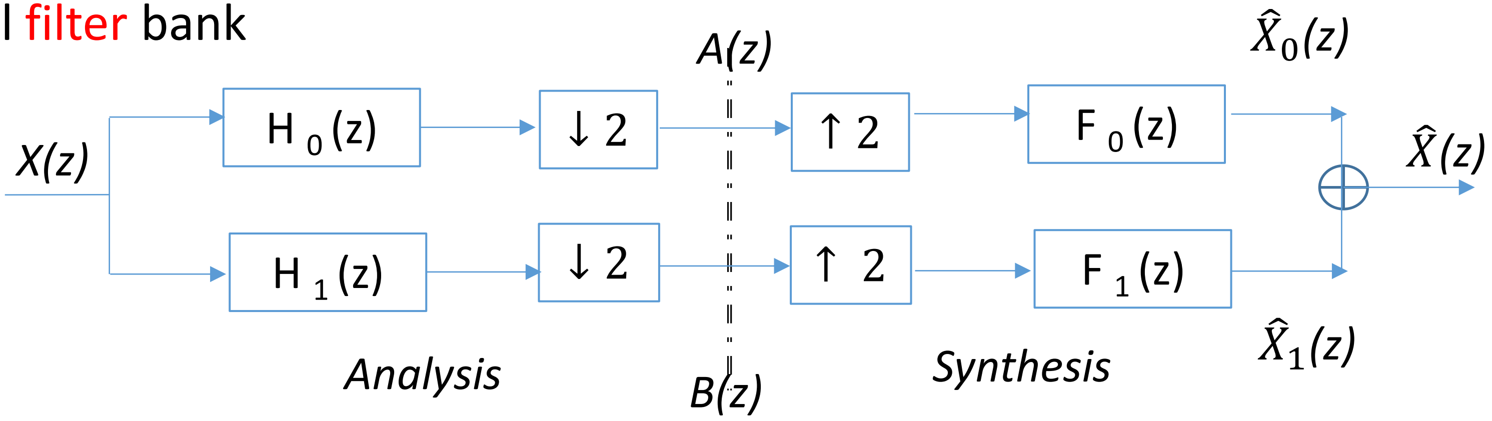
↓ 2

↑ 2

P-high$F_1(z)$

3.3. Filter Banks

Example : 2-channel filter bank



- $A(z) = \left(\frac{1}{2}\sum_{k=0}^{1} H_0\left(W_2^k z^{\frac{1}{2}}\right) X\left(W_2^k z^{\frac{1}{2}}\right)\right)$ with $W_2^1 = e^{-\frac{2\pi j}{2}} = -1$

$\Rightarrow A(z) = \frac{1}{2}\left(H_0\left(z^{\frac{1}{2}}\right)X\left(z^{\frac{1}{2}}\right) + H_0\left(-z^{\frac{1}{2}}\right)X\left(-z^{\frac{1}{2}}\right)\right)$

- $\hat{X}_0(z) = A(z^2)F_0(z) \Rightarrow \hat{X}_0(z) = F_0(z)\frac{1}{2}\left(H_0(z)X(z) + H_0(-z)X(-z)\right)$
- Likewise $\hat{X}_1(z) = F_1(z)\frac{1}{2}\left(H_1(z)X(z) + H_1(-z)X(-z)\right)$

$\Rightarrow \boldsymbol{\hat{X}(z) = \frac{1}{2}\left(F_0(z)H_0(z) + F_1(z)H_1(z)\right)X(z) + \frac{1}{2}\left(F_0(z)H_0(-z) + F_1(z)H_1(-z)\right)X(-z)}$

3.3. Filter Banks

Example : 2-channel filter bank

$$\hat{X}(z) = \frac{1}{2}\left(F_0(z)H_0(z) + F_1(z)H_1(z)\right)X(z) + \frac{1}{2}\left(F_0(z)H_0(-z) + F_1(z)H_1(-z)\right)X(-z)$$

Reconstruction if $\hat{x}(n) = x(n)$within a delay, i.e.:

- $F_0(z)H_0(z) + F_1(z)H_1(z) = 2\, z^{-k}$
- $F_0(z)H_0(-z) + F_1(z)H_1(-z) = 0 \quad \Rightarrow \begin{cases} F_0(z) = -H_1(-z) et\ F_1(z) = H_0(-z) \\ F_0(z) = H_1(-z)\ et\ F_1(z) = -H_0(-z) \end{cases} \quad ou$

Eliminate the term in X(-z) $\Rightarrow$ QMF filters (Quadrature Mirror Filter)

If we impose $H_1(z) = H_0(-z)$ or *$h_1(n) = (-1)^n h_0(n)$*

- $F_0(z) = H_0(z)$ or $f_0(n) = h_0(n)$
- $F_1(z) = -H_0(-z)$ or $f_1(n) = -(-1)^n h_0(n)$

compensation for the effects of folding to have perfect reconstruction

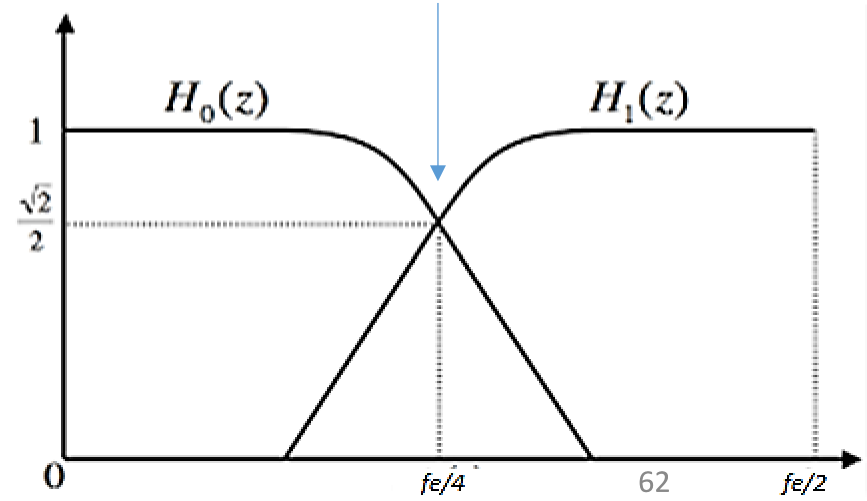
3.3. Filter Banks

Example : 2-channel filter bank

$$\hat{X}(z) = \frac{1}{2}\big(F_0(z)H_0(z) + F_1(z)H_1(z)\big)X(z) + \frac{1}{2}\big(F_0(z)H_0(-z) + F_1(z)H_1(-z)\big)X(-z)$$

□ $H_1(z) = H_0(-z)$ □ $F_0(z) = H_0(z)$ □ $F_1(z) = -H_0(-z)$

Mirror effect around $f_e/4$



$H_1(z) = H_0(-z) \Rightarrow H_1(e^{2\pi j f T_e}) = H_0(-e^{2\pi j f T_e}) = H_0(e^{2\pi j (f-\frac{f_e}{2}) T_e})$

$\Rightarrow H_1(f) = H_0\left(f - \frac{f_e}{2}\right)$

- $F_0(z)H_0(z) + F_1(z)H_1(z) = 2\, z^{-k}$

$\rightarrow H_0^2(z) - H_1^2(z) = 2\, z^{-k}$

Faculty of Electrical Engineering, USTHB [akourgli@usthb.dz
http://perso.usthb.dz/~akourgli/

3.3. Filter Banks

Example : 2-channel filter bank

Eliminate the term in X(-z) $\Rightarrow$ CQF filters (Conjugate Quadrature Filter)

We impose $H_1(z) = \pm H_0(-z^{-1})\, z^{-(L-1)}$ or $h_1(n) = \mp(-1)^n h_0(L-1-n)$

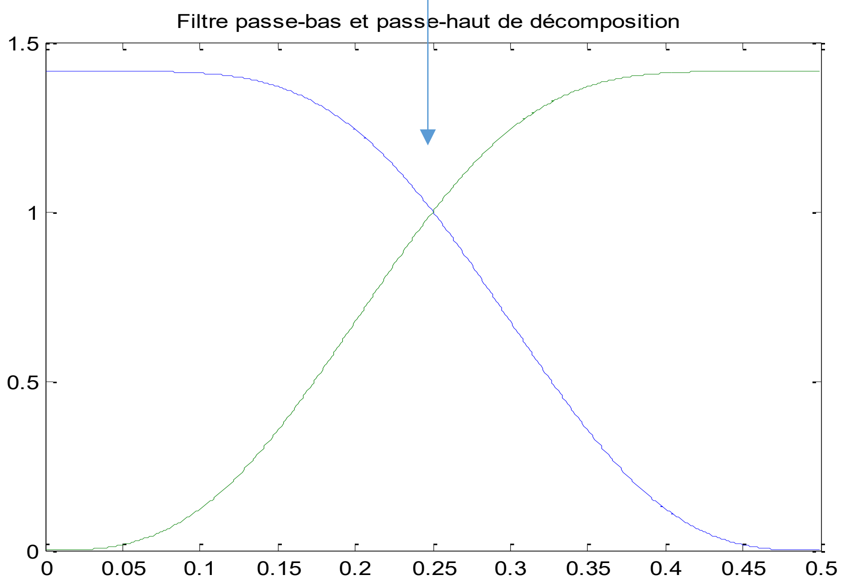
- $F_0(z) = H_1(-z) = H_0(z^{-1})\, z^{-(L-1)}$ or $f_0(n) = h_0(L-1-n)$
- $F_1(z) = -H_0(-z)$ or $f_1(n) = -(-1)^n h_0(n)$

Rigorously accurate reconstruction of the input signal unlike to QMF filters but greater computational complexity

$$H_1(z) = H_0(-z^{-1}) \Rightarrow H_1(e^{2\pi j f T_e}) = H_0(e^{-2\pi j (f-\frac{f_e}{2})T_e})$$

$$\Rightarrow \; H_1(f) = H_0\left(\frac{f_e}{2} - f\right)$$

Mirror effect around $f_e/4$



○Quadrature: The 2 filters intersect at $f_e/4$. The bandwidth used by each of the filters is half of the initial band. The sum their squared modulus is 2