

Traitement Avancé du Signal

(Support de cours)

Master 1 : Réseaux de Télécommunications

Faculté de Génie Electrique

Université des Sciences et de la Technologie Houari Boumediene

Site perso : <http://perso.usthb.dz/~akourgli/>

e-mails: akourgli@usthb.dz , kourgliassia@gmail.com

Programme

Introduction : Rappels divers

Chapitre 1. Rappels sur les filtres numériques (RIF et RII)

(3 Semaines)

1. Transformée en Z
2. Structures, fonctions de transfert, stabilité et implémentation des filtres numériques (RIF et RII)
3. Filtre numérique à minimum de phase
4. Méthodes de synthèses des filtres RIF et des filtres RII
5. Filtres numériques Multicadences

TP n°1 : Synthèse de filtres RIF et RII

TP n°2 : Filtrage Mult-cadences

Chapitre 2. Signaux aléatoires et processus stochastiques

(4 Semaines)

1. Rappel sur les processus aléatoires
2. Notions de processus stochastiques
3. Stationnarités au sens large et strict et Ergodicité
4. Exemples de processus stochastiques (processus de Poisson, processus gaussien et Markovien)
5. Statistiques d'ordre supérieur (Moments et cumulants, Polyspectres, processus non gaussiens, traitements non linéaires)
6. Densité spectrale de puissance
7. Filtre adapté, filtre de Wiener
8. Périodogramme, corrélogramme, périodogramme moyenné, périodogramme lissé
9. Introduction au filtrage particulière

TP n°3 : Processus Aléatoires, périodogramme et variantes

TP n°4 : Filtrage adapté et Filtrage de Wiener

Chapitre 3. Analyse spectrale paramétrique et filtrage numérique adaptatif (4 Semaines)

1. Méthodes paramétriques
2. Modèle AR (Lévinson, Yulewalker, Burg, Pisarenko, Music ...)- Modèle ARMA
3. Algorithme du gradient stochastique LMS
4. Algorithme des moindres carrés récurrents RLS

TP n°5 : Modélisation AR et Filtrage adaptatif (LMS)

Chapitre 4. Analyse temps-fréquence et temps-échelle (4 Semaines)

1. Dualité temps-fréquence
2. Transformée de Fourier à court terme
3. Ondelettes continues, discrètes et ondelettes dyadiques
4. Analyse multi-résolution et bases d'ondelettes
5. Transformée de Wigner-Ville
6. Analyse Temps-Echelle,

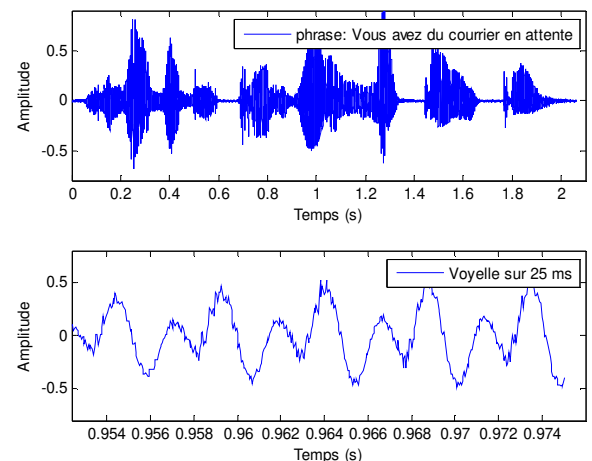
TP n°6 : Transformées temps-fréquence/Echelle

Introduction : Rappels divers

Un **signal** est la représentation physique de l'information qu'il transporte de sa source à son destinataire. Il sert de vecteur à une information. Il constitue la manifestation physique d'une grandeur mesurable (courant, tension, force, température, pression, etc.). Les signaux sont des grandeurs électriques variant en fonction du temps $x(t)$ obtenues à l'aide de capteurs. Sur le plan analytique : Un signal sera une fonction d'une variable réelle, en général le temps.

Exemples :

- ☐ Onde acoustique : délivré par un micro (parole, musique, ...)
- ☐ Signaux biologiques : EEG, ECG
- ☐ Tension aux bornes composant électronique
- ☐ Signaux géophysiques : vibrations sismiques
- ☐ Finances : cours du pétrole
- ☐ Images, Vidéos



Remarque: Tout signal physique comporte une *composante* aléatoire (perturbation externe, bruit, erreur de mesure, etc ...). Le **bruit** est défini comme tout phénomène perturbateur gênant la perception ou l'interprétation d'un signal, par analogie avec les nuisances acoustiques (interférence, bruit de fond, etc.). La différenciation entre le signal et le bruit est artificielle et dépend de l'intérêt de l'utilisateur : les ondes électromagnétiques d'origine galactique sont du bruit pour un ingénieur des télécommunications par satellites et un signal utile pour les radioastronomes.

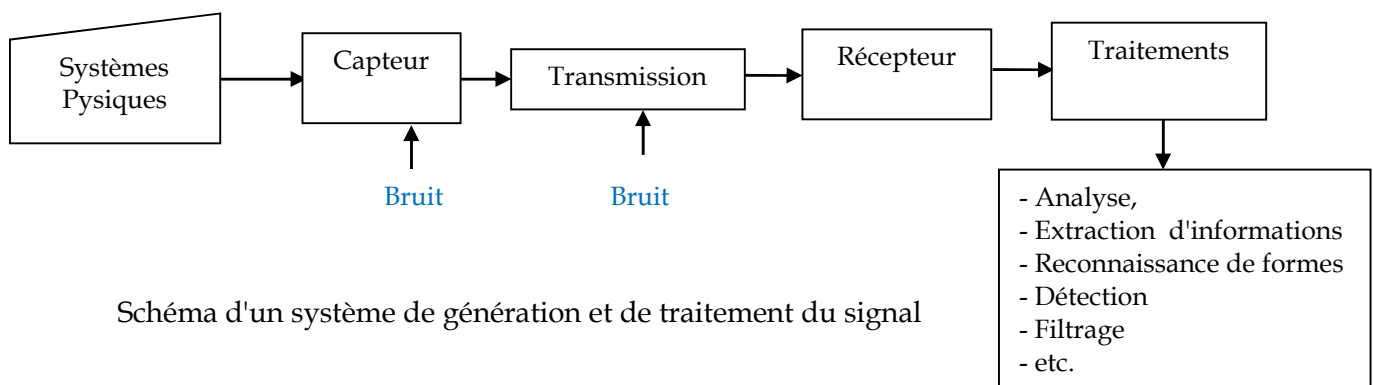


Schéma d'un système de génération et de traitement du signal

Les fonctions du traitement du signal peuvent se diviser en deux catégories : l'élaboration des signaux (incorporation des informations) et l'interprétation des signaux (extraction des informations). Les principales fonctions intégrées dans ces deux parties sont les suivantes [1]:

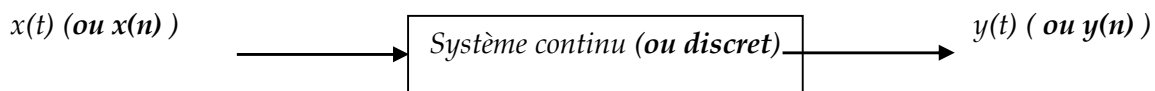
Élaboration des signaux : synthèse, modulation, codage/compression, etc.

Interprétation des signaux : filtrage, détection, identification, analyse, mesure, etc.

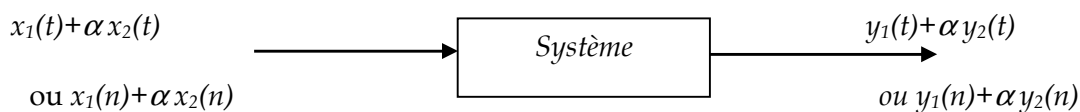
1. Théorie des systèmes linéaires et invariants dans le temps (SLIT) discrets et continus

Un système linéaire est un modèle de système qui applique un opérateur linéaire à un signal d'entrée. C'est une abstraction mathématique très utile en automatique, traitement du signal, mécanique et télécommunications. Les systèmes linéaires sont ainsi fréquemment utilisés pour décrire un système non linéaire en ignorant les petites non-linéarités.

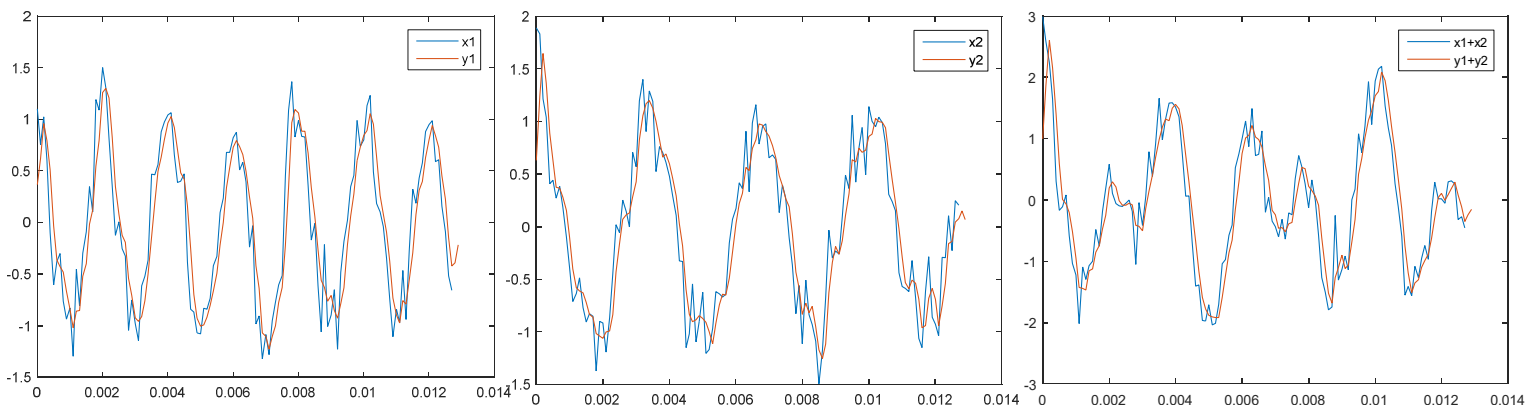
Un système est continu si à une entrée continue $x(t)$, il fournit une sortie continue $y(t)$. Un système discret fera correspondre à une suite d'entrées discrètes $x(n)$ une suite de sorties discrètes $y(n)$.



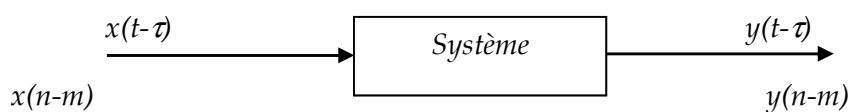
- Le système sera dit linéaire si quand on applique une entrée $k x(t)$ (ou $k x(n)$), la sortie sera $k y(t)$ (ou $k y(n)$). Si deux entrées $x_1(t)$ et $x_2(t)$ engendrent deux sorties $y_1(t)$ et $y_2(t)$ alors $x_1(t) + x_2(t)$ engendrera $y_1(t) + y_2(t)$. (De même, dans le cas discret : si deux entrées $x_1(n)$ et $x_2(n)$ engendrent deux sorties $y_1(n)$ et $y_2(n)$ alors $x_1(n) + x_2(n)$ engendrera $y_1(n) + y_2(n)$)



Exemple



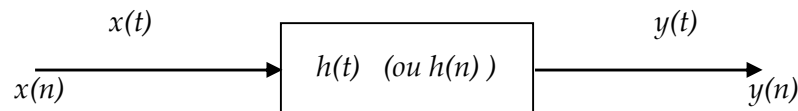
- S'il y a invariance dans le temps, une translation de l'entrée $x(t) \Rightarrow x(t-\tau)$ (ou $x(n) \Rightarrow x(n-m)$) se traduira par une même translation dans le temps de la sortie $y(t) \Rightarrow y(t-\tau)$ (ou $y(n) \Rightarrow y(n-m)$).



Si le système est invariant, cela implique que le système réagit de la même façon quel que soit l'instant auquel nous appliquons ses excitations. Cette propriété exprime que la caractéristique du système ne dépend pas de l'origine du temps, on parle encore de stationnarité.

Convolution

Si les hypothèses de linéarité et d'invariance temporelle sont vérifiées, on peut caractériser le système par sa réponse impulsionnelle $h(t)$ (ou $h(n)$).



On peut en déduire l'effet d'une entrée quelconque sous la forme d'une convolution. Cette dernière est l'opération de traitement de signal la plus fondamentale. Elle indique que la valeur du signal de sortie à l'instant t (ou n) est obtenue par la sommation (intégrale) pondérée des valeurs passées du signal d'excitation $x(t)$ (ou $x(n)$). La fonction de pondération est précisément la réponse impulsionnelle $h(t)$ (ou $h(n)$) :

$$y(t) = x(t) * h(t) = \int x(\tau)h(t - \tau)d\tau = \int x(t - \tau)h(\tau)d\tau$$

$$y(n) = h(n) * x(n) = \sum_{m=-\infty}^{\infty} h(m)x(n - m) = \sum_{m=-\infty}^{\infty} x(m)h(n - m)$$

La réponse impulsionnelle $h(t)$ (ou $h(n)$) est le signal qu'on obtient en sortie $y(t)=h(t)$ (ou $y(n)=h(n)$) si on applique en entrée une impulsion "de Dirac" $x(t)=\delta(t)$ (ou $x(n)=\delta(n)$). Ainsi, le Dirac est l'élément neutre de l'opération de convolution :

$$x(t) * \delta(t) = x(t)$$

$$\delta(n) * x(n) = x(n)$$

Quelques propriétés du Dirac

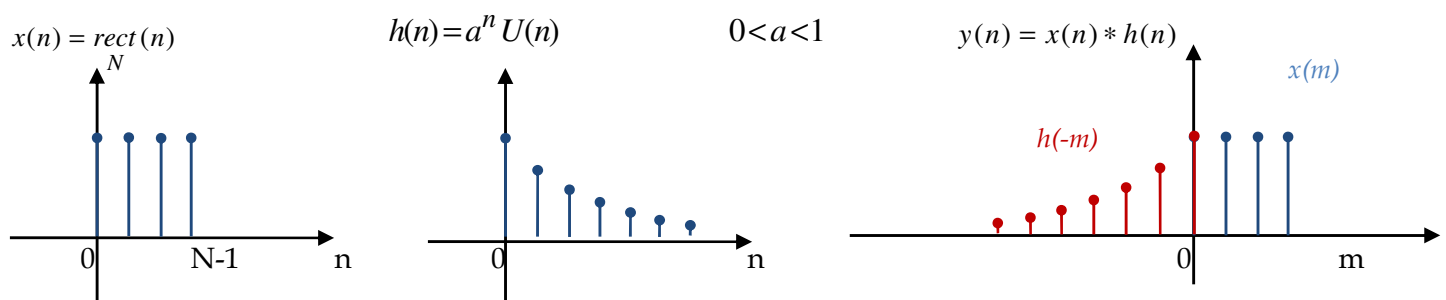
$$\int_{-\infty}^{+\infty} x(t)\delta(t - t_0)dt = x(t_0)$$

$$x(t) * \delta(t - t_0) = x(t - t_0)$$

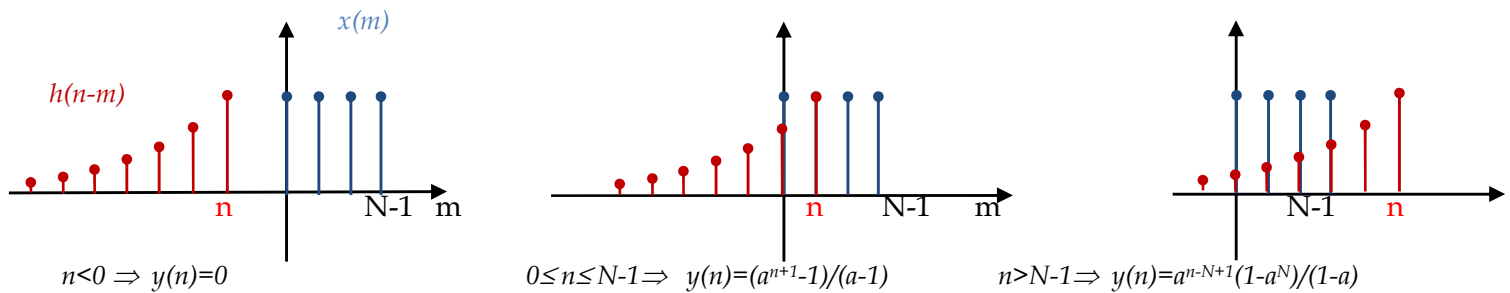
$$x(t)\delta(t - t_0) = x(t_0)\delta(t - t_0)$$

Le calcul de la convolution consiste donc à calculer la somme du produit $x(\tau)h(t - \tau)$ (ou $x(m)h(n - m)$). Le signal $h(t - \tau)$ (ou $h(n - m)$) est simplement le signal initial $h(\tau)$ (ou $h(m)$), retourné dans le temps pour donner $h(-\tau)$ (ou $h(-m)$) puis translaté de t (ou n). En calculant alors l'ensemble des produits obtenus en faisant « glisser » h , c'est-à-dire pour tous les décalages de t (ou n), on obtient le produit de convolution pour tout t (ou n).

Exemple 1 : (en discret)



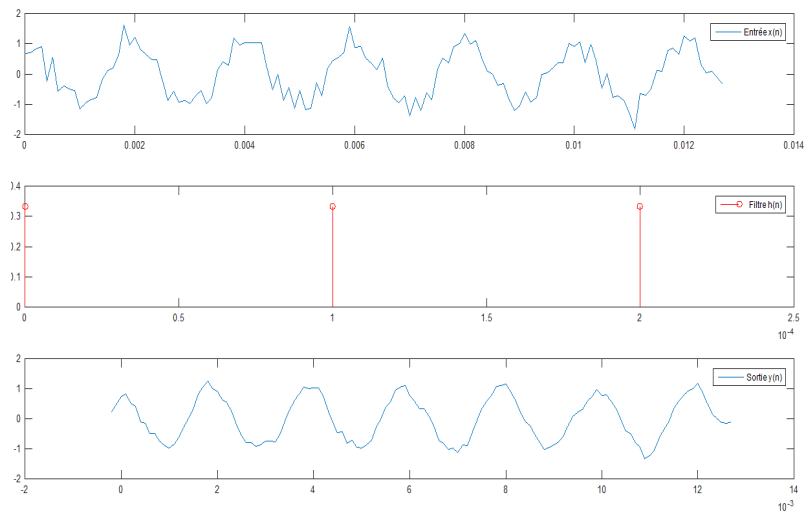
On distingue 3 cas :



Exemple 2 : (Voir TP n°1)

$$h(t) = \frac{1}{T} \Pi_T(t) \Rightarrow y(t) = \frac{1}{T} \int_{t-T/2}^{t+T/2} x(\tau) d\tau$$

$$h(n) = \frac{1}{N+1} \Pi_{N+1}(n) \Rightarrow y(n) = \frac{1}{N+1} \sum_{m=-N/2}^{N/2} x(n+m)$$



Exemple 3: Soit le signal $x(n) = \{2, -1, 3\}$ et $h(n) = \{1, 2, 2, 3\}$

Calculer $y(n)=x(n)*h(n)$

Pour des séquences finies, on peut utiliser la méthode des colonnes

$Y(n)=\{2, 3, 5, 10, 3, 9\}$

$h[n]$	=	1	2	2	3
$x[n]$	=	2	-1	3	
		2	4	4	6
			-1	-2	-2
				3	6
					6
					9
		2	3	5	10
					3
					9

Remarques

Si on applique à un SLIT une entrée sinusoïdale réelle ou complexe de fréquence f_0 , alors, la sortie sera une sinusoïde dont l'amplitude et la phase pourront être modifiées mais qui conservera la même forme (une sinusoïde) et la même fréquence f_0 . On dit que les sinusoïdes sont les fonctions propres des SLIT.

Un système linéaire invariant est un système dont le comportement dans le temps, peut-être décrit par :

- soit par une équation différentielle linéaire à coefficients constants: $\sum_{i=0}^M a_i \frac{d^i y(t)}{dt^i} = \sum_{i=0}^N b_i \frac{d^i x(t)}{dt^i}$

- soit par une équation aux différences, pour le cas discret: $\sum_{i=0}^M a_i y(n-i) = \sum_{i=0}^N b_i x(n-i)$,

2. Stabilité et causalité

Une contrainte importante pour la formalisation de nombreux problèmes est de respecter la notion de *causalité* (les effets ne peuvent pas précéder la cause). Dans le cas des SLIT, cette causalité se traduit par le fait que pour: $h(t) = 0$ pour $t < 0$ (ou $h(n) = 0$ pour $n < 0$).

$$x(t) = 0, t < t_0 \text{ alors } y(t) = 0, t < t_0 \Rightarrow h(t) = 0, t < 0, y(t) = \int_{-\infty}^t x(\tau)h(t-\tau) d\tau = \int_0^{+\infty} x(t-\tau)h(\tau) d\tau$$

Ce qui nous donne en discret :

$$x(n) = 0, n < n_0 \text{ alors } y(n) = 0, n < n_0 \Rightarrow h(n) = 0, n < 0$$

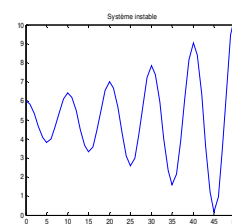
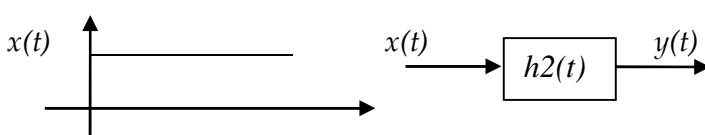
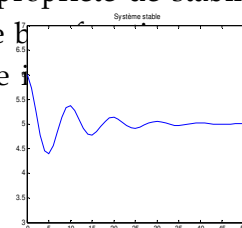
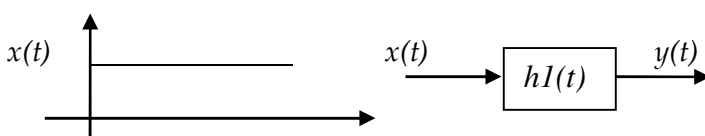
$$y(n) = \sum_{m=-\infty}^n x(m)h(n-m) = \sum_{m=0}^{+\infty} x(n-m)h(m)$$

- si h et x sont causaux $y(n) = \sum_{m=0}^n h(n-m)x(m)$

Remarque:

Nous pouvons envisager mémoriser les signaux d'entrée et faire un traitement de ceux-ci en temps différé, les systèmes utilisés ne sont plus alors nécessairement causaux car pour élaborer la sortie à l'instant t_i (ou n_i), nous disposons en mémoire des entrées aux instants suivants. C'est souvent le cas en traitement d'image, en traitement de parole effectué après mémorisation du signal à traiter.

Une autre notion fondamentale est la *stabilité* des systèmes. La propriété de stabilité des systèmes bouclés est non seulement une performance mais une exigence pour le bon fonctionnement d'une boucle d'asservissement ou de régulation. Une boucle instable est une boucle qui ne respecte pas la définition la plus courante de cette stabilité est la suivante :



On dit qu'un système est stable si, en lui appliquant une entrée bornée quelconque, la sortie reste

bornée, ce qui implique dans le cas des SLIT:

ou

Remarque : Pour le reste, les définitions seront données pour le cas discret.

3. Energie et puissance

Toute transmission d'information s'accompagne de transferts d'énergie. En effet, les signaux continus ou discrets sont essentiellement caractérisés par l'énergie ou la puissance qu'ils véhiculent. Ce sont les seules grandeurs physiques auxquelles sont sensibles les détecteurs. Beaucoup de capteurs physiques mesurent une énergie ou une quantité quadratique. Par exemple, les capteurs optiques mesurent une intensité, les compteurs d'électricité mesurent une énergie, etc. Compte tenu de la définition fondamentale, l'énergie du signal entre les instants t et $t+dt$ est : $|x(t)|^2 dt$ (puissance instantanée multipliée par le temps).

Soit un signal $x(n)$ à temps discret, tel que $\sum_{n=-\infty}^{+\infty} |x(n)|^2$ existe et converge. Alors le signal est dit à énergie finie et la valeur de cette somme est appelée énergie du signal :

$$E_x = \sum_{n=-\infty}^{+\infty} |x(n)|^2$$

Exemples:

$x(n) = \text{Rect}(n/N)$ énergie finie. $x(n) = a$ (constante) et $x(n) = A \sin(2\pi f_0 n)$ ne sont pas à énergie finie

Pour un signal périodique, cette somme ne converge pas. On peut néanmoins définir la puissance d'un signal $x(n)$ périodique de période N par :

$$P_x = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} |x(n)|^2 \quad \text{ou} \quad P_x = \frac{1}{2N} \sum_{n=-N}^{N-1} |x(n)|^2$$

Dans le cas général, on parle de signaux à puissance moyenne finie définie par:

$$P_x = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=-N/2}^{N/2-1} |x(n)|^2 \quad \text{ou} \quad P_x = \lim_{N \rightarrow \infty} \frac{1}{2N} \sum_{n=-N}^{N-1} |x(n)|^2$$

Exemples:

Signal continu $x(t)=a$, $A \sin(2\pi f_0 t)$, signaux périodiques, échelon unité, peigne de Dirac.

Il existe des signaux ni périodiques, ni d'énergie finie, pour lesquels la puissance ne peut être définie, comme par exemple la rampe $x(n)=n$. Il s'agit là de définitions mathématiques, en pratique, un signal mesuré ne l'est jamais sur un intervalle de temps infini. On peut commencer à visualiser un signal à un instant qu'on prendra comme origine des temps, et dans ce cas on arrêtera son examen au bout d'un temps T_{obs} :

$$E_x = \sum_{n=0}^{N_{obs}} |x(n)|^2$$

Remarque

Signal à énergie finie $\Rightarrow$ puissance nulle

Signal à puissance finie $\Rightarrow$ énergie infinie

Le calcul de l'énergie ou la puissance permet d'obtenir une première caractérisation du signal. Par ailleurs, la théorie du signal a largement développé des méthodes d'étude basées sur la corrélation pour caractériser le comportement temporel du signal.

Exercice d'application :

Calculer l'énergie et la puissance des signaux: $\Pi(n)$, $A \cos(2\pi f_0 n)$.

4. Corrélation et auto-corrélation

La fonction de corrélation permet de mesurer le degré de ressemblance entre deux signaux en fonction d'un décalage. Considérons $x(n)$ et $y(n)$ deux signaux d'énergie finie, la fonction d'intercorrélation $R_{xy}(k)$ est

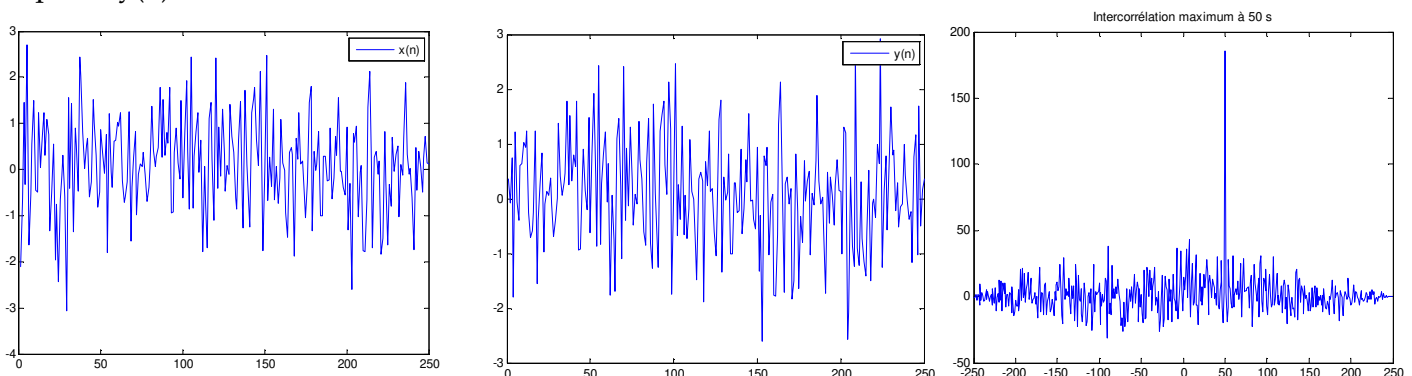
$$\text{définie par: } R_{xy}(k) = \sum_{n=-\infty}^{\infty} x(n) y^*(n-k)$$

L'inter-corrélation entre $x(t)$ et $y(t)$ atteint un maximum pour un retard k si $x(n)=y(n-k)$

$$\text{Pour des signaux à puissance moyenne finie, elle vaut : } R_{xy}(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} x(n) y^*(n-k)$$

Exemples

Soient un signal aléatoire et sa version décalée de 50s. On remarque que les signaux se ressemblent le plus quand $y(n)$ est décalé de 50 secondes.



Pour l'auto-corrélation, on remplace $y(n)$ par $x(n)$ on obtient l'expression de l'auto-corrélation pour les

$$\text{signaux à énergie finie: } R_{xx}(k) = \sum_{n=-\infty}^{\infty} x(n) x^*(n-k)$$

L'auto-corrélation permet de détecter des régularités, des profils répétés dans un signal comme un

signal périodique perturbé par beaucoup de bruit (Voir TP n°1)

Propriétés :

- Pour $k = 0$, on retrouve l'énergie du signal $R_{xx}(0) = E_x$ et $R_{xx}(k)$ est maximale en $k=0$
- Si $x(n)$ est réel, l'auto-corrélation est réelle et paire.
- L'auto-corrélation d'un signal de durée N aura une taille $2*N-1$

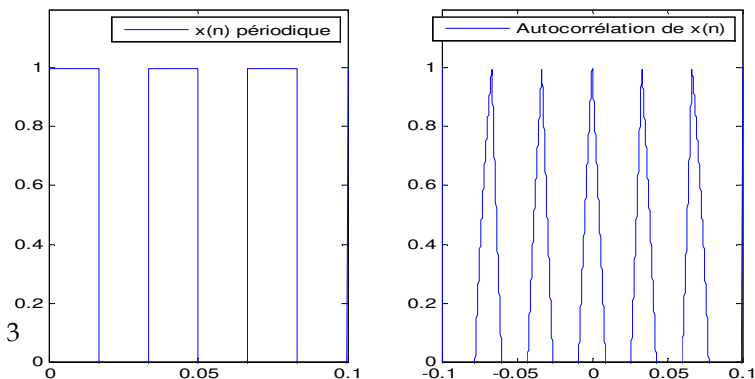
Auto-corrélation des signaux périodiques : Le calcul sur une seule période suffit. L'auto-corrélation d'un signal périodique est-elle même périodique. Par définition, le signal périodique ressemble parfaitement à lui-même, décalé d'une ou plusieurs périodes.

- signaux périodiques

$$R_x(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} x(n) x^*(n-k)$$

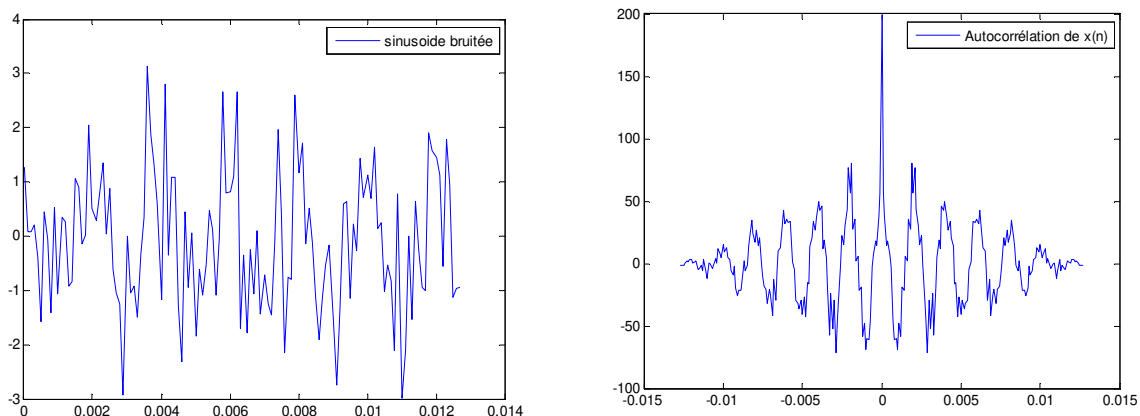
Exercice d'application Soit le signal $x(n)=(n+1)$ pour $n=0$ à 3

Calculer l'auto-corrélation de x et déduire son énergie

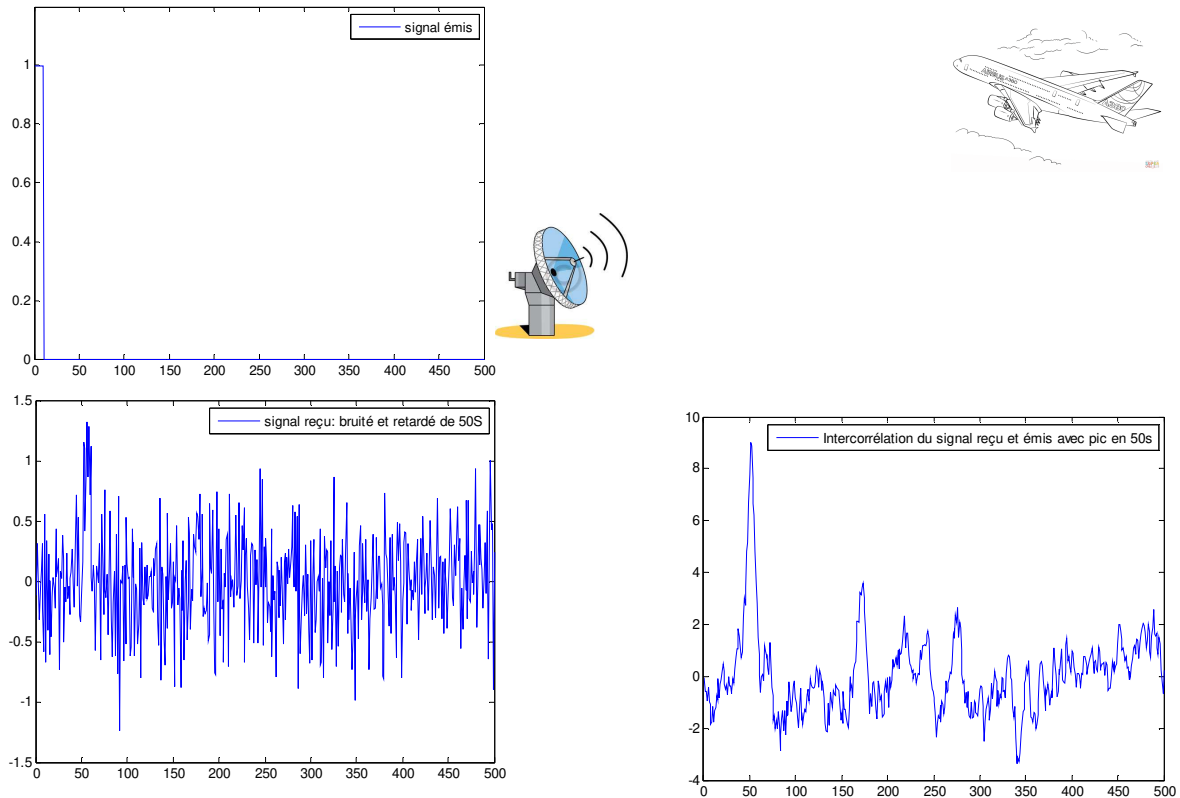


5. Applications fondamentales des fonctions de corrélation

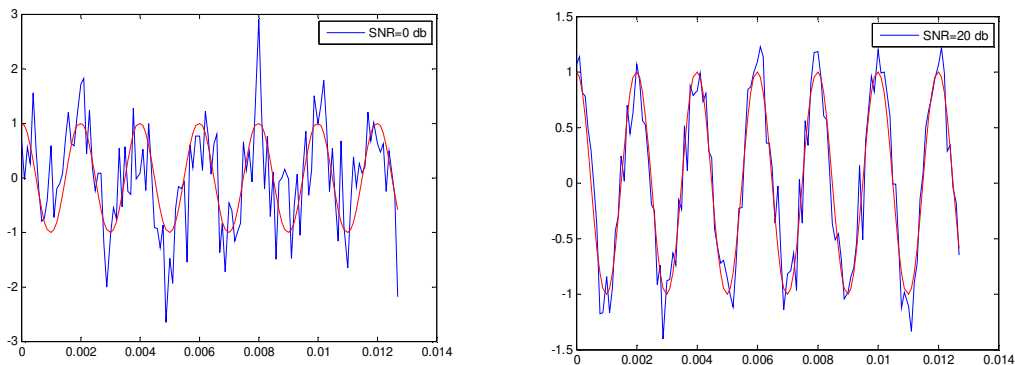
Extraction d'un signal noyé dans du bruit, mesure d'un temps ou retard, détection d'un signal périodique (Voir TP n° 1). L'exemple ci-dessous illustre l'auto-corrélation d'un signal sinusoïdal d'amplitude 1 noyé dans du bruit Gaussien de variance 1.



La corrélation est largement utilisée dans les systèmes radar. Ainsi, pour détecter un avion, on envoie une impulsion, puis on reçoit une version retardée, atténuée et bruitée de cette impulsion. L'intercorrélation du signal reçu et émis présentera un pic à l'instant correspondant au retard.



Remarque: La notion de bruit est relative, elle dépend du contexte. Le rapport signal/bruit désigne la qualité de la transmission d'une information par rapport aux parasites. Il est défini par: $SNR_{db} = 20 \log(P_S/P_B)$



6. Filtres à réponses impulsionnelles finies ou infinies (RIF et RII)

- Si les a_i sont $\neq 0$, le système est dit récursif (RII), il est non récursif s'il ne dépend que des $x(n-i)$ (RIF)

- Si le système est à réponse impulsionnelle de durée finie (RIF), alors : $y(n) = \sum_{m=0}^K h(m)x(n-m)$
 Dans ce cas, le système numérique est une fenêtre centrée sur les K plus récents échantillons.

- Si le système est à réponse impulsionnelle de durée infinie (RII) : $y(n) = \sum_{m=0}^{+\infty} h(m)x(n-m)$

Dans ce cas, il est nécessaire de connaître tous les échantillons présents et passés, le système a une mémoire de longueur infinie.

Exemple 1 $y(n)=x(n)+a_1x(n-1)+a_2x(n-2)+\dots+a_kx(n-k)$ est l'équation aux différences finies d'un filtre RIF

avec comme réponse impulsionnelle $h(n)=\delta(n)+a_1\delta(n-1)+a_2\delta(n-2)+\dots+a_k\delta(n-k)$ qui, on peut le constater, est bien finie.

Exemple 2 $y(n)=x(n)+a_1y(n-1)$ est l'équation aux différences finies d'un filtre RII

avec $y(n-1)=x(n-1)+a_1y(n-2) \Rightarrow y(n)=x(n)+a_1x(n-1)+a_1^2y(n-2)$

de même $y(n-2)=x(n-2)+a_1y(n-3) \Rightarrow y(n)=x(n)+a_1x(n-1)+a_1^2x(n-2)+a_1^3y(n-3)$

$\Rightarrow y(n)=x(n)+a_1x(n-1)+a_1^2x(n-2)+a_1^3x(n-3)+\dots+a_1^m y(n-m)$

En poursuivant le procédé à l'infini $y(n)$ dépend d'une infinité de $x(n-k)$ ce qui en fait un filtre RII.

Exemple d'application

Les séquences $x(n)$ (réel) et $y(n)$ représentent respectivement l'entrée et la sortie d'un système discret. Pour chaque cas, identifiez celles représentant

- a) des systèmes linéaires, b) des systèmes causaux, c) des systèmes invariants aux translations de n ,
d) des systèmes assurément ou possiblement stables (en fonctions des constantes)

1. $y(n) = x(n) + bx(n-1)$
2. $y(n) = x(n) + bx(n+1)$
6. $y(n) = b^{x(n)}$ b : constante réelle
3. $y(n) = nx(n)$
5. $y(n) = x(n)e^n$
7. $y(n) = |x(n)|$
4. $y(n) = x(n) \sin(2\pi f_0 n)$

- a) Tous sauf 6 et 7 b) Tous sauf 2 c) Les systèmes 1, 2, 6 et 7 d) 1 (b finie), 2 (b finie), 4, 6 (b finie) et 7.

I. Rappels sur les filtres numériques (RIF et RII)

La transformée de Fourier est un outil précieux d'analyse et de traitement des signaux. Cependant, dans certains problèmes (comme le filtrage numérique), les limites de la TF sont vite atteintes. La transformée en Z, qui s'applique aux signaux discrets, généralise la TF et permet de dépasser ces limites [10]. Elle est tout-à-fait analogue à la transformée de Laplace, mais plus facile à utiliser. Ce type de transformée permet de décrire aisément les signaux à temps discret et la réponse des systèmes linéaires invariants soumis à des entrées diverses. C'est un outil qui permet de calculer la réponse impulsionnelle d'un système linéaire invariant décrit par une équation aux différences finies. Elle permet l'interprétation directe des caractéristiques des signaux et des filtres dans le domaine des fréquences [9].

1. Transformée en Z et propriétés

- **Définition :** La TZ est la généralisation de la TFTD ($X(f) = \sum_{n=-\infty}^{+\infty} x(nT_e)e^{-2\pi jfnT_e}$).

Soit un signal discret $x(n)$. Sa TZ est définie par:

$$X(z) = \sum_{n=-\infty}^{+\infty} x(n).z^{-n} \quad \text{où } z \text{ est une variable complexe définie partout où cette série converge.}$$

En effet, comme cette transformation est une série infinie, elle n'existera que pour les valeurs de z pour lesquelles cette série converge.

Exemples :

$$- x(n) = \delta(n) \Rightarrow X(z) = 1,$$

$$- x(n) = \delta(n-k) \Rightarrow X(z) = z^{-k}, \quad \text{si } k > 0 \text{ RDC} = \mathbb{C} - \{0\} \quad \text{si } k < 0 \text{ RDC} = \mathbb{C} - \{\infty\}$$

$$- x(n) = (1, 2, 3, 5, 0, 2) \text{ on peut écrire } x(n) = \delta(n) + 2.\delta(n-1) + 3.\delta(n-2) + 5.\delta(n-3) + 2.\delta(n-5)$$

$$X(z) = 1 + 2z^{-1} + 3z^{-2} + 5z^{-3} + 2z^{-5} \text{ RDC} = \mathbb{C} - \{0\}$$

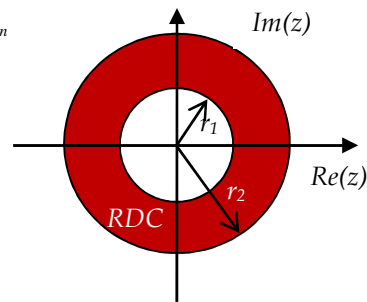
L'ensemble des valeurs de la variable complexe z pour lesquelles la série converge est appelée Région De Convergence (RDC):

$$RDC = \left\{ z \in \mathbb{C} / \sum_{n=-\infty}^{+\infty} |x(n).z^{-n}| < +\infty \right\}$$

De façon générale, on montre que la RDC est un anneau de convergence centré sur l'origine défini par :

$$r_1 < |z| < r_2 \quad \text{avec } r_1 = \lim_{n \rightarrow +\infty} |x(n)|^{1/n} \text{ et } r_2 = \lim_{n \rightarrow +\infty} |x(-n)|^{-1/n}$$

où r_1 peut être réduit à 0 et r_2 peut être égal à $+\infty$.

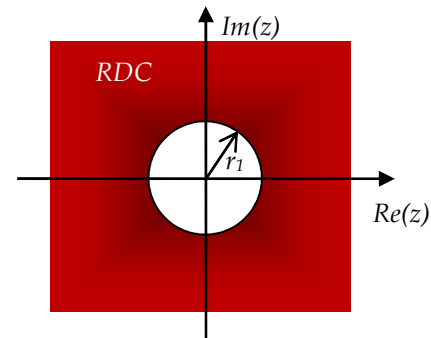
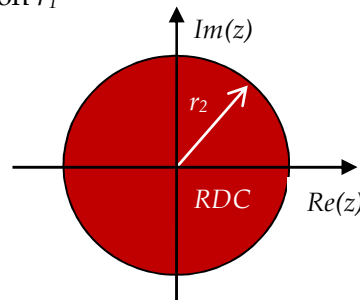


- $x(n)=0$ pour $n < n_0 \Rightarrow r_2 = +\infty$,

RDC = région extérieure au cercle de rayon r_1

- $x(n)=0$ pour $n > n_0 \Rightarrow r_1 = 0$

RDC = disque de rayon r_2



$\Rightarrow$ système anti-causal : RDC cercle. système causal : RDC extérieure au cercle.

Exemples

- Soit $a > 0$, $x(n) = \begin{cases} a^n & \text{si } n \geq 0 \\ 0 & \text{si } n < 0 \end{cases} \Rightarrow X(z) = \sum_{n=0}^{+\infty} a^n z^{-n} = \frac{z}{z-a}$, convergente pour $|z| > a$.

- Soit $b > 0$, $y(n) = \begin{cases} 0 & \text{si } n \geq 0 \\ -b^n & \text{si } n < 0 \end{cases} \Rightarrow Y(z) = \sum_{n=0}^{-\infty} -b^n z^{-n} = \frac{z}{z-b}$, convergente pour $|z| < b$.

- Soient $a > 0, b > 0$, $w(n) = \begin{cases} a^n & \text{si } n \geq 0 \\ b^n & \text{si } n < 0 \end{cases} \Rightarrow W(z) = \frac{z}{z-a} - \frac{z}{z-b}$, convergente pour $b > |z| > a$.

Remarque : La TZ de a^n pour $n \in]-\infty, +\infty[$ n'existe pas.

Les propriétés qui sont les plus utilisées sont résumées comme suit :

Si on définit : $x(n) \xrightarrow{TZ} X(z)$, $x_1(n) \xrightarrow{TZ} X_1(z)$ et $x_2(n) \xrightarrow{TZ} X_2(z)$

- Linéarité : $a x_1(n) + b x_2(n) \xrightarrow{TZ} a X_1(z) + b X_2(z)$ - Convolution : $x_1(n) * x_2(n) \xrightarrow{TZ} X_1(z) X_2(z)$

- Théorème du retard : $x(n-k) \xrightarrow{TZ} z^{-k} X(z)$ - Avance : $x(n+k) \xrightarrow{TZ} z^k X(z) - \sum_{n=0}^{k-1} x(n) z^{k-n}$

Exemples

$$x(n) = \cos(w_0 n) U(n) = \frac{1}{2} (e^{jw_0 n} + e^{-jw_0 n}) U(n) \Rightarrow$$

$$X(z) = \frac{1}{2} \left(\frac{1}{1 - e^{jw_0} z^{-1}} + \frac{1}{1 - e^{-jw_0} z^{-1}} \right) = \frac{1 - \cos(w_0) z^{-1}}{1 - 2 \cos(w_0) z^{-1} + z^{-2}} \quad \text{avec} \quad |z| > 1$$

2. TZ rationnelles (correspondant aux SLID)

Les systèmes linéaires invariants décrits par une équation aux différences finies possèdent une transformée en Z rationnelle c'est ainsi que celles-ci vont s'écrire comme le rapport de deux polynômes en z^{-1} .

$$\sum_{i=0}^M a_i y(n-i) = \sum_{i=0}^N b_i x(n-i) \xrightarrow{TZ} \sum_{i=0}^M a_i z^{-i} Y(z) = \sum_{i=0}^N b_i z^{-i} X(z)$$

On peut caractériser un système LI par $h(n)$ ou par la transformée en Z ($H(z)$) de sa réponse impulsionnelle $h(n)$, encore appelée fonction de transfert du système.

$$\Rightarrow H(z) = \frac{\sum_{i=0}^N b_i z^{-i}}{\sum_{i=0}^M a_i z^{-i}} = \frac{b_0}{a_0} z^{M-N} \frac{N(z)}{D(z)} = \frac{b_0}{a_0} z^{M-N} \frac{\prod_{i=1}^N (z - z_i)}{\prod_{i=1}^M (z - p_i)} = K z^{M-N} \frac{\prod_{i=1}^N (z - z_i)}{\prod_{i=1}^M (z - p_i)}$$

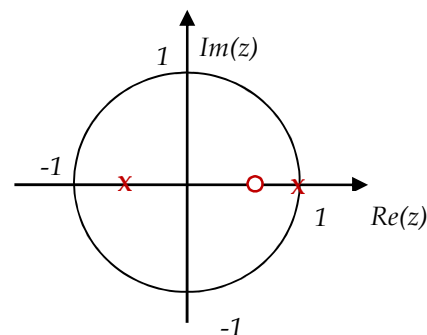
On appelle zéros, les valeurs de z pour lesquelles $H(z)=0$ et on appelle pôles, les valeurs de z pour lesquelles $H(z)$ est infini (annule le dénominateur). C'est ainsi que $H(z)$ possède N zéros (z_i), M pôles (p_i). Si $M > N$, elle possède $(M-N)$ zéros en 0, sinon $(N-M)$ pôles en 0.

Ainsi, la position de ses pôles et de ses zéros (+le facteur d'amplitude $K=b_0/a_0$) va nous fournir une description complète de $H(z)$ (par conséquent de $h(n)$ et $H(f)$) donc du comportement du système. $H(z)$ peut donc être représentée sous la forme d'un cercle modélisant la position des pôles et des zéros dans le plan complexe.

Exemple

$$H(z) = \frac{3z - 2}{(z - 1)(z + 0.5)}$$

Un zéro en $2/3$ et deux pôles $p_1 = -0.5$ et $p_2 = 1$



Remarques

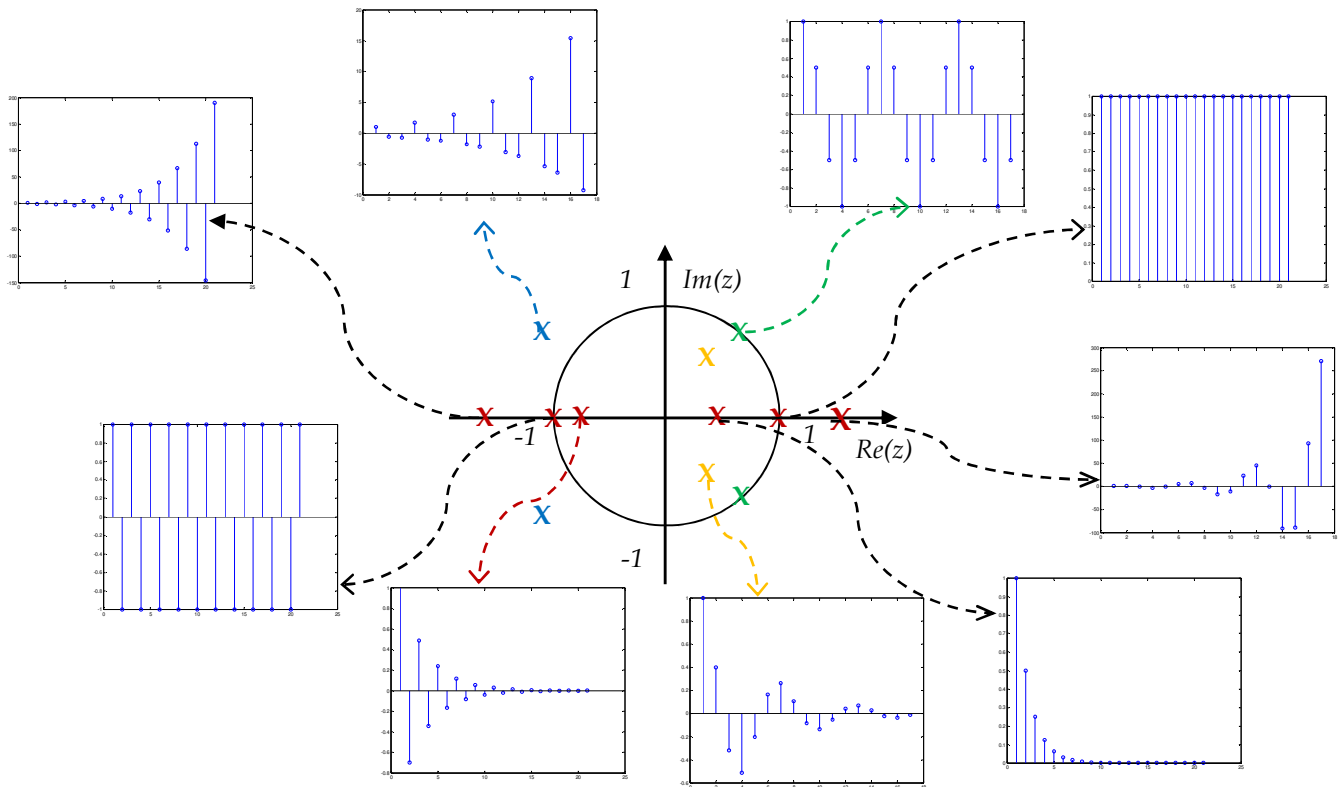
- Dans la plupart des systèmes, les a_i et les b_i sont réels $\Rightarrow$ les pôles et les zéros sont soit réels soit des paires de complexes conjugués.

- Rappelons que le rayon d'un système causal se trouve à l'extérieur d'un cercle. Par ailleurs, s'il est stable :

$$\sum_n |h(n)| < \infty, \text{ puisque } H(z) = \sum_{n=-\infty}^{+\infty} h(n) z^{-n}, \text{ il suffit donc que } z=1 \text{ fasse partie de la RDC.}$$

- Pour un système causal et stable, tous les pôles sont à l'intérieur du cercle unité ($|p_i| < 1, \forall i$). Le domaine de convergence ne peut contenir de pôles puisque la TZ ne converge pas aux pôles. S'il est anti-causal, il sera stable si les pôles sont à l'extérieur du cercle unité.

- Si le filtre est non-récurif $H(z) = \sum_{i=0}^N b_i \cdot z^{-i}$. Un filtre RIF a tous ses pôles à l'origine et sera donc toujours stable.



- A un pôle p_i simple ou multiple va correspondre une réponse impulsionnelle qui converge si $|p_i| < 1$. Elle divergera dans le cas contraire, soit si $|p_i| > 1$.

- Sachant qu'à chaque pôle complexe est associé un pôle conjugué cela donnera une réponse impulsionnelle $h(n)$ oscillante (cosinus ou sinus) amortie si $|p_{i=1,2}| < 1$ ou divergente si $|p_{i=1,2}| > 1$.

- Dans un système à phase minimale, tous les zéros sont à l'intérieur du cercle unité ($|z_i| < 1, \forall i$).

• Détermination de la réponse fréquentielle des filtres numériques

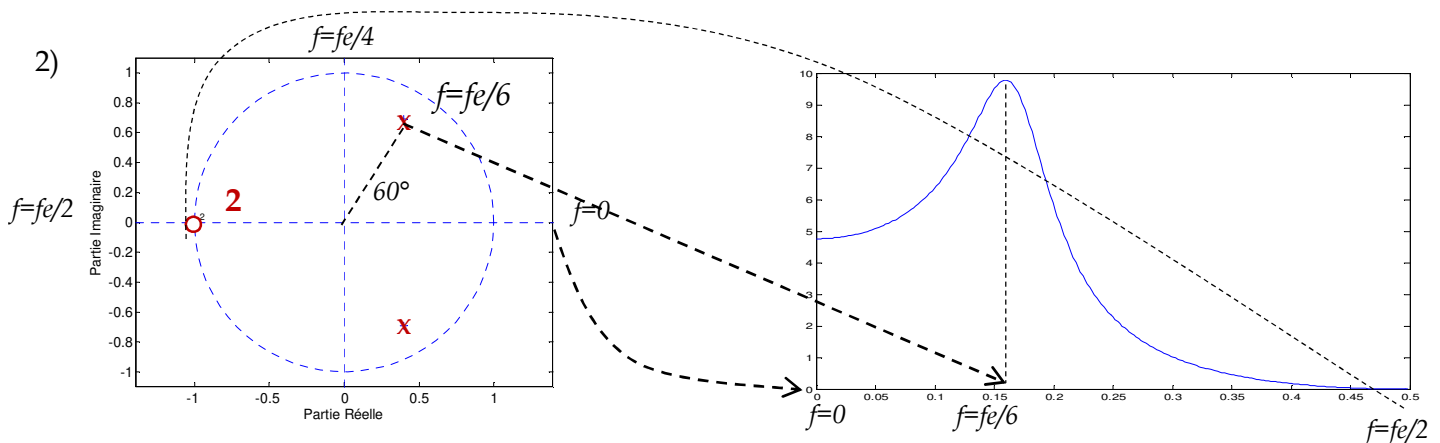
On suppose que le cercle unité ($|z| = 1$) $\in$ RDC de $X(z)$. On restreint le calcul de $X(z)$ au cercle unité en posant $z = e^{2\pi j f T_e}$.

Lorsqu'un zéro est placé sur un point donné du plan en z , la réponse fréquentielle sera de 0 au point considéré. Un pôle quant à lui produira un pic au point correspondant. Plus les pôles ou les zéros sont proches du cercle unité, plus ils influencent la réponse en fréquence [11].

- un zéro ou un pôle à l'origine n'influent pas sur le module de la réponse fréquentielle.

- un zéro sur le cercle unité introduit une annulation du module pour la fréquence correspondant
- Un zéro au voisinage du cercle unité introduit une atténuation dans le module de la réponse en fréquence. Atténuation d'autant plus importante que le zéro est proche du cercle unité.
- Un pôle sur le cercle unité introduit une résonance infinie dans le module de la réponse en fréquence pour la fréquence correspondante.
- Un pôle au voisinage du cercle unité introduit une résonance d'autant plus importante dans le module de la réponse en fréquence que le pôle est proche du cercle unité.

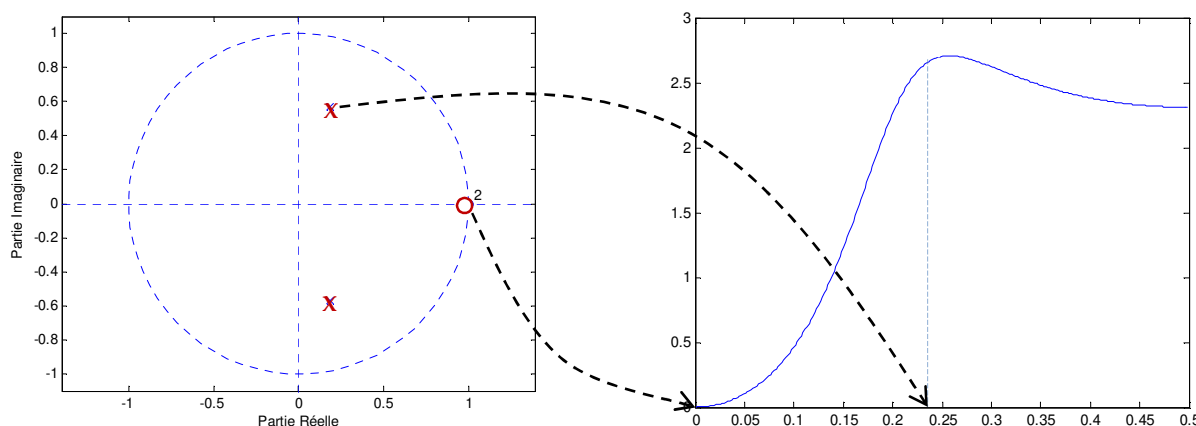
1) Sur la figure ci-dessous le cercle complet correspond à une fréquence d'échantillonnage f_e . Des pôles proches du cercle unité sont à l'origine de larges pics tandis que des zéros proches ou sur le cercle unité produisent des minima. Ce tracé nous permettra d'identifier la nature du filtre. On peut aussi avoir une idée sur son comportement général : passe-bas, passe-haut ou passe-bande, connaître sa ou ses fréquences de coupure.



$$3) H(z) = \frac{z^2 - 2z + 1}{z^2 - 0.371z + 0.36} \Rightarrow \text{Zéros double en } z=-1, \text{ pôles } p_{1,2} = \pm 0.6e^{j72^\circ}$$

- Un zéro double en $z = 1 \Rightarrow |H(f)| = 0$ pour $f = 0$

- Des pôles proches du cercle unité $\Rightarrow$ maxima.



Remarque : Puisque les coefficients du filtre sont réels, les pôles et zéros sont réels (sur l'axe des réels) ou paires de complexes conjugués.

- **Détermination de la réponse impulsionnelle des FN (TZ inverse)**

La transformée en Z présente l'avantage d'être plus facilement inversible que la transformée de Fourier. Le passage de la TZ vers $h(n)$ peut se faire par le biais de la transformées en Z de signaux élémentaires connus à condition qu'il soit possible d'écrire $H(z)$ comme la combinaison de transformées élémentaires. Dans le cas contraire, on peut employer l'intégration sur un contour fermé en utilisant le calcul des résidus, ou le développement en puissance de z et de z^{-1} , ou encore le développement en fractions élémentaires [12].

1. La relation générale de la transformée en z inverse est donnée par l'équation donnée par l'intégrale de Cauchy : $x(n) = \frac{1}{2\pi.j} \oint_C X(z).z^{n-1}.dz$, où C est un contour fermé parcouru dans le sens inverse des aiguilles d'une montre contenant l'origine.

En pratique, on utilise le théorème des résidus: $x(n) = \sum_{p_i \text{ poles de } z^{n-1}X(z)} \text{Re s} \left[z^{n-1}X(z) \right]_{z=p_i}$

$$\text{Re s} [z^{n-1}X(z)]_{z=p_i} = \frac{1}{(m-1)!} \frac{d^{m-1}}{dz^{m-1}} \left[(z-p_i)^m z^{n-1}X(z) \right]_{z=p_i}$$

2. Transformée inverse par division polynômiale : Il est possible de calculer la transformée en Z inverse selon les puissances croissantes de z^{-1} (système causal) ou selon les puissances décroissantes de z (système anti-causal).

$$X(z) = \sum_n C_n z^{-n} \xrightarrow{TZ^{-1}} x(n) = C_n$$

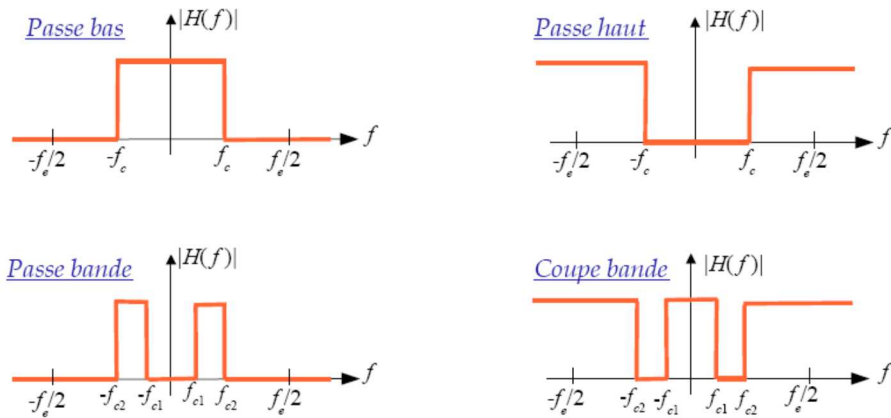
3. L'idée générale de cette approche consiste à trouver pour une fonction $X(z)$ complexe un développement en fonctions en Z plus simples et pour lesquelles une transformée inverse est connue:

$$X(z) = \sum_i X_i(z) \xrightarrow{TZ^{-1}} x(n) = \sum_i x_i(n)$$

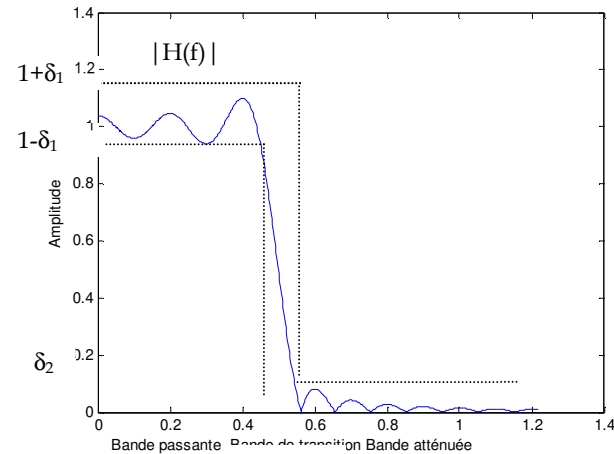
où les $X_i(z)$ sont des fonctions dont les TZ⁻¹ sont connues (Voir page 38).

3. Caractéristiques des filtres numériques

Un filtre numérique est constitué d'un groupement de circuits logiques astreints à un processus de calcul (ou algorithme) qui confère à ce filtre une fonction déterminée (passe-bas, passe-haut, passe-bande, réjecteur de bande, intégrateur [$y(n)=(x(n)+x(n-1))/2$], différentiateur [$y(n)=(x(n)-x(n-1))/2$], ...). Il faut souligner que certains filtres ne sont pas conçus pour arrêter une fréquence, mais pour modifier légèrement le gain à différentes fréquences, comme les égaliseurs. Ce sont tous des systèmes linéaires, discrets, invariants dans le temps et unidimensionnels. De plus, pour qu'ils soient physiquement réalisables, il faut qu'ils soient nécessairement causaux.



Les filtres représentés ci-dessus sont idéaux. Dans un cas réel il n'est pas possible d'obtenir une fréquence de coupure aussi raide. Le passage entre zones passantes et zones atténuées se fait par des zones dites "de transition" f_p-f_a dont la largeur va exprimer la sélectivité du filtre. Par ailleurs, les bandes passantes et atténuées ne sont également pas idéales, elles contiennent des ondulations dont l'amplitude est exprimée par les paramètres d'ondulation en bande passante δ_1 et bande atténuée δ_2 [9].



- Filtre à phase linéaire :**

Idéalement, il est souhaitable qu'un filtre possède une phase linéaire dans la bande passante. Une phase linéaire assurera un même déphasage pour toutes les fréquences (pas de distorsion). Les filtres FIR peuvent générer des filtres à phase linéaire à la condition que la réponse impulsionnelle soit symétrique.

Si un filtre est à phase linéaire, sa réponse fréquentielle est de la forme :

$$H(f) = R(f)e^{-j\phi(f)} \text{ avec } \phi(f) = \phi_0 + 2\pi f\tau$$

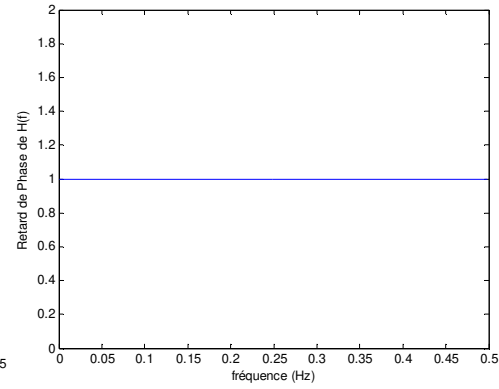
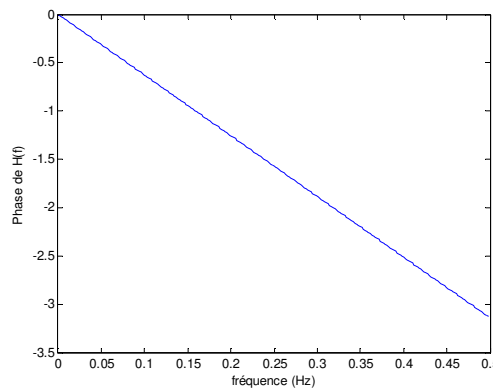
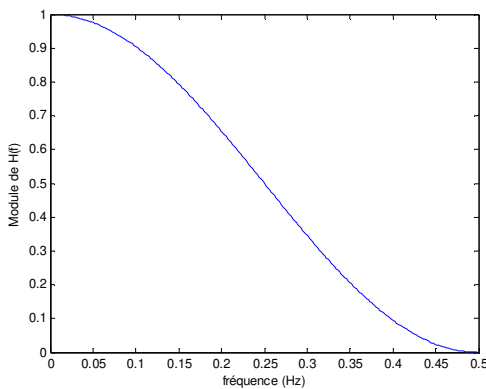
Et la dérivée de cette dernière par rapport à f fournit le 'retard de groupe', défini donc par :

$$\beta = -\frac{d\phi(f)}{df}$$

et qui correspond au retard subi par le signal après être passé par un filtre. Si la phase est linéaire (filtres RIF symétrique), le retard est constant et le signal à la sortie aura donc une distorsion minimale puisque l'effet de la phase sur le signal sera un simple décalage temporel (primordial dans un système audio)

Exemple: $y(n) = \frac{1}{4}(x(n) + 2x(n-1) + x(n-2))$

$$\Rightarrow h(n) = \frac{1}{4}(\delta(n) + 2\delta(n-1) + \delta(n-2)) \text{ et } H(f) = e^{-2\pi jf} \cos^2(\pi f)$$



- **Filtre à minimum de phase :**

Pour un filtre RII, le retard de groupe n'est pas constant. On souhaite avoir un minimum de retard en s'orientant faire des filtres à minimum de phase. Un filtre est dit à minimum de phase si lui et son inverse sont tous deux stables et causaux. Ce qui implique que tous les pôles et zéros de la fonction de transfert $H(z)$ sont à l'intérieur du disque unité. Un filtre à phase minimale présente une réponse impulsionnelle (coefficients de $H(z)$) à tendance décroissante car ses zéros sont en module inférieurs ou égaux à l'unité.

Exemple : $y(n) = 0,75 x(n) + 0,25 x(n-1)$

$y(n)$ est obtenu plutôt à partir des échantillons "récents" du signal d'entrée qu'à partir des échantillons "anciens". Il faut noter que tout filtre causal et stable peut être décomposé en un produit de filtres à phase minimale par une cellule passe-tout.

- **RII ou RIF ?**

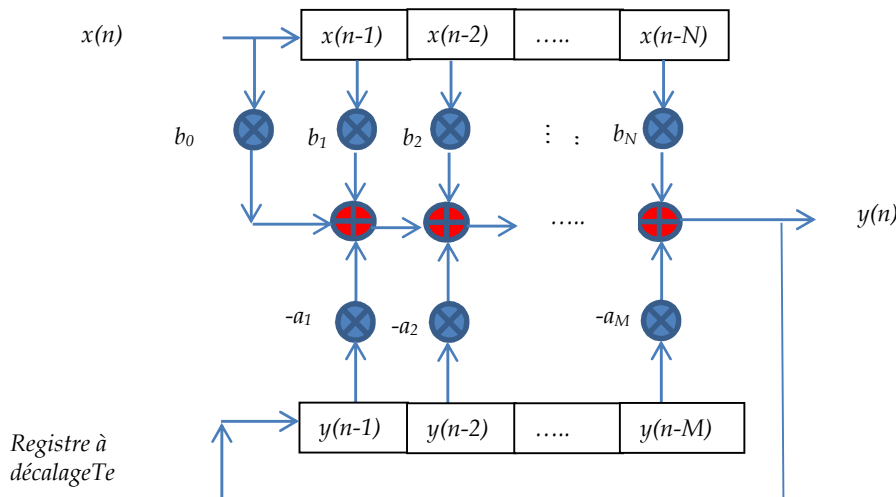
➤ Les filtres RII, on l'avantage qu'ils sont efficaces. Avec très peu de pôles et zéros on peut assurer la plupart des réponses fréquentielles dont on peut avoir besoin dans les applications audio. Cependant, le filtre étant récursif, les erreurs de précision numérique deviennent une question d'importance, car ils peuvent s'amplifier et devenir hors de contrôle, d'abord dans la forme de bruit, mais éventuellement dans la forme d'instabilité. La forme de la réponse impulsionnelle n'est pas facile à déterminer, non plus, car elle est définie indirectement par les pôles et zéros de $H(f)$.

➤ Par contre, les filtres RIF n'ont jamais des problèmes d'instabilité, car la sortie n'est qu'une somme finie d'échantillons de l'entrée. Cependant, quand la réponse impulsionnelle est longue, le numéro d'opérations peut devenir un facteur décisif quand il faut choisir entre RIF ou RII. Un autre avantage des RIF est le retard de groupe constant, qui permet d'avoir une distorsion de phase minimale sur le signal traité [13].

- **Structure des filtres numériques**

L'application d'un filtre numérique implique le calcul de la sortie $y(n)$ à l'instant $t=nT_e$ à partir des sorties et entrées précédentes plus la valeur courante de l'entrée.

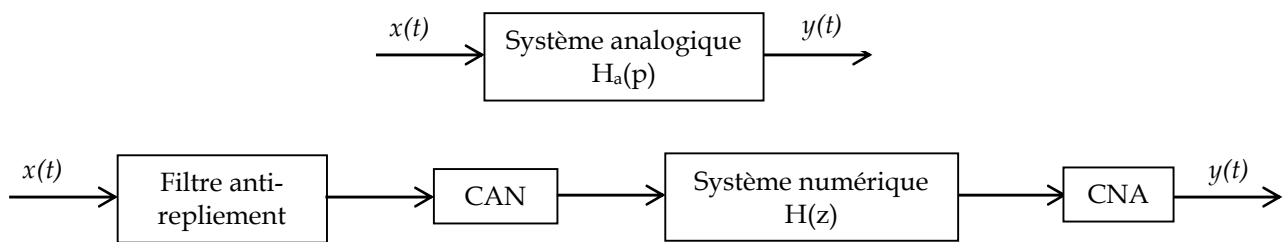
$$\sum_{i=0}^M a_i y(n-i) = \sum_{i=0}^N b_i x(n-i) \quad \text{En prenant } a_0=1, \text{ on obtient}$$



Un filtre numérique est généralement constitué des éléments suivants : un ou plusieurs organes de retard (ce sont des registres à décalage jouant le rôle de mémoires retardées), pilotés par une horloge de période; des opérateurs arithmétiques (additionneurs et multiplieurs); des registres fournissant les coefficients de pondération du filtre [18]. Ainsi, à chaque top d'horloge T_e , les valeurs des registres subissent un décalage permettant de calculer la nouvelle sortie. Cette structure peut être aussi réalisée par logiciel.

4. Synthèse des Filtres numériques et Gabarit analogique

La synthèse d'un filtre est un ensemble de processus qui débute par la définition des caractéristiques du filtre, jusqu'à sa réalisation informatique et/ou électronique, en passant par la détermination de ses coefficients. Pour synthétiser un filtre numérique, on considère connu le gabarit du filtre analogique et on cherche un système numérique caractérisée par une fonction de transfert $H(z)$ à insérer dans le circuit ci-dessus permettant de satisfaire le gabarit analogique.



La détermination de la fonction de transfert d'un filtre numérique, par une méthode directe, n'est pas toujours très simple. Par contre, le problème qui consiste à transformer un filtre analogique en un filtre numérique est relativement simple. De ce fait, de nombreuses méthodes sont proposées pour concevoir un filtre numérique à partir du filtre analogique équivalent. Dans tous les cas, la synthèse d'un filtre numérique est une approximation d'un filtre analogique idéal équivalent. Il est nécessaire de contraindre un certain nombre de paramètres.

La synthèse d'un filtre numérique comprend les étapes suivantes :

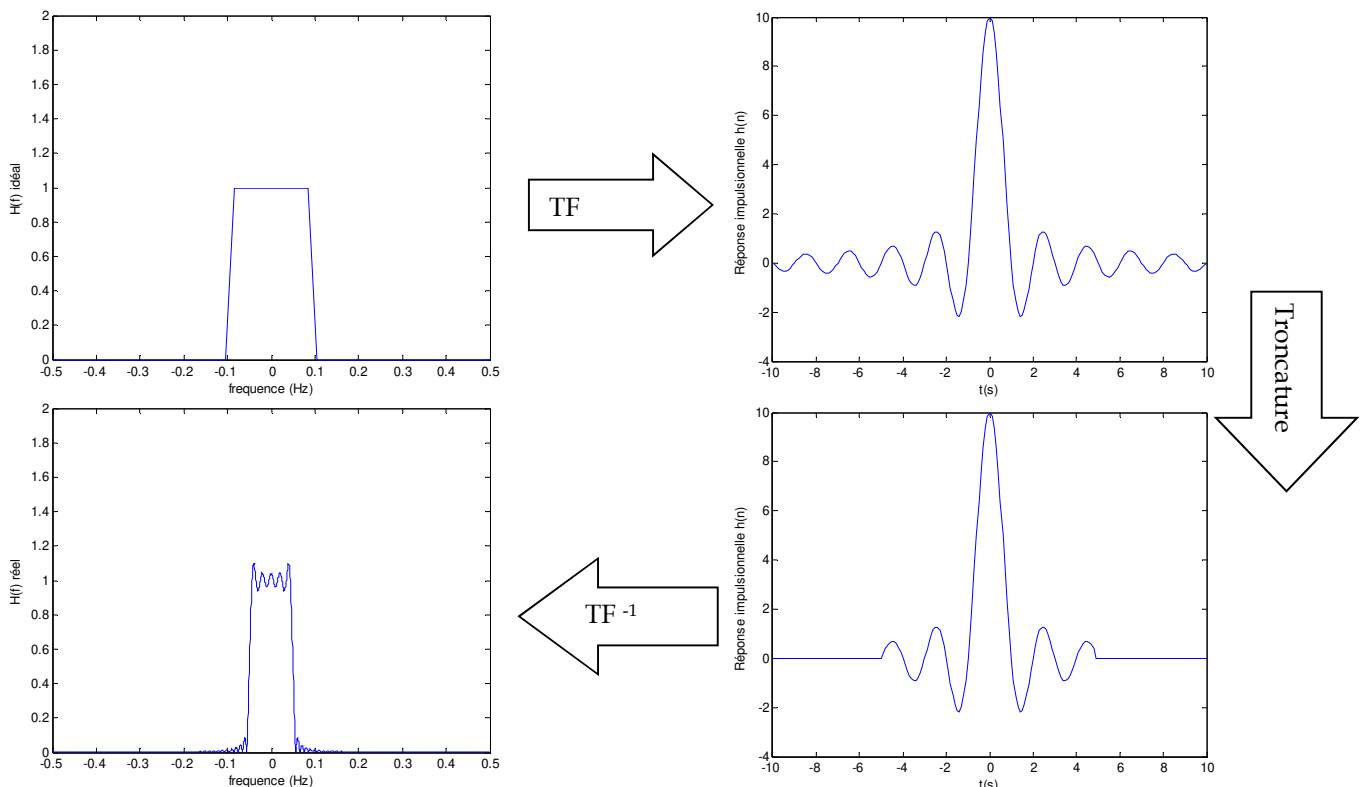
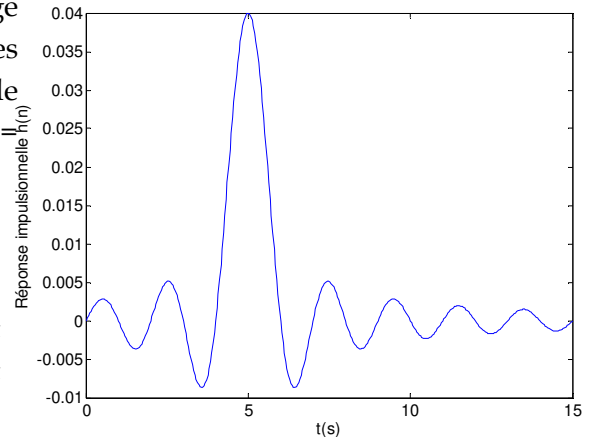
1. la détermination d'une réponse en fréquence idéale souhaitée ;

2. la détermination de la meilleure approximation sous un certain nombre de contraintes (stabilité, rapidité, précision, déphasage linéaire, etc.) ;
3. le choix d'une structure de calcul réalisant le filtre approximé.

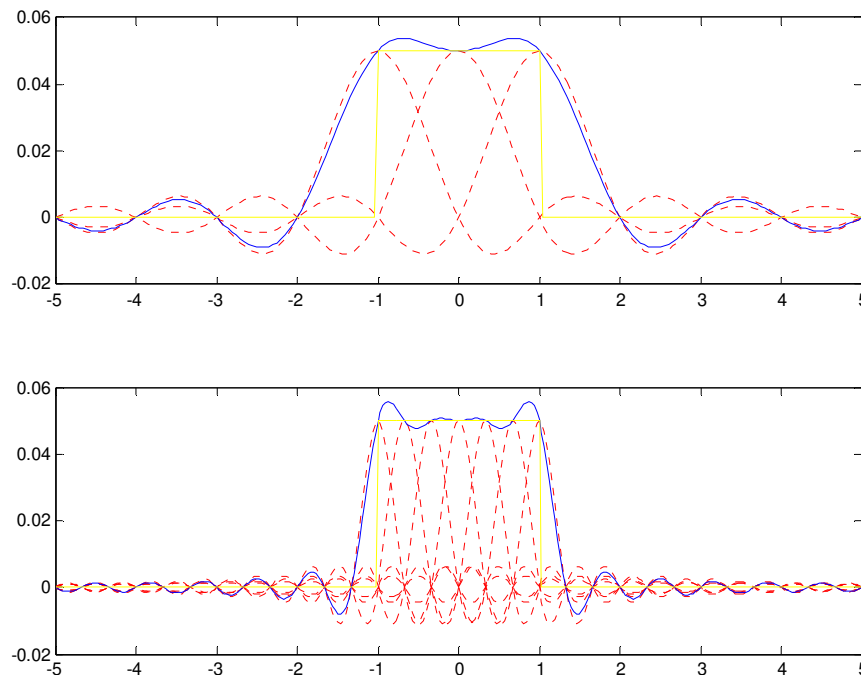
Les fonctions modèles utilisées pour la synthèse des filtres sont soit la réponse impulsionnelle soit la réponse en fréquence (celle-ci est préférée) de filtres analogiques connus. Si l'on emploie la réponse impulsionnelle, les éléments $h(n)$ de la réponse impulsionnelle numérique sont obtenus en calculant $h(t)$, la réponse impulsionnelle du filtre analogique, aux instants $t=nT_e$.

Rappelons que les filtres idéaux présentent un déphasage linéaire et ne sont pas physiquement réalisables, car les réponses fréquentielles idéales correspondent à une réponse temporelle non-causale. Par exemple, en considérant le filtre passe-bas $H(f) = \Pi(f) e^{-2\pi j f T}$, on obtiendra un sinc décalé de T .

On peut observer qu'il faut annuler une partie du signal. De ce fait, il n'est plus possible d'obtenir un filtre passe-bas idéal (droit et avec une droite de transition perpendiculaire). Il s'ensuit que les filtres qui vont pouvoir être réellement synthétisés n'ont pas de réponse fréquentielle correspondant à la fonction porte, mais pourront s'en rapprocher.



Comme l'illustre la figure suivante, la troncature (multiplication par une porte de largeur NT_e) du sinc dans le domaine temporel se traduira par une convolution dans le domaine fréquentiel du filtre idéal avec un sinc s'annulant tous les NT_e . Pour de grandes valeurs de N , les sinc dans la bande passante se compenseront les uns les autres mais autour des points de discontinuité (fréquence de coupure), les ondulations restent apparentes.



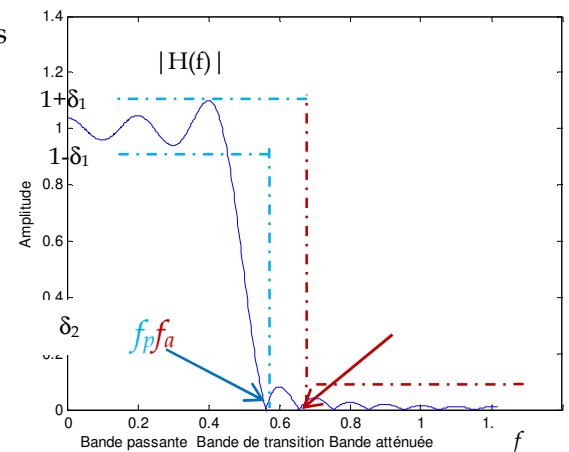
On peut observer que les différences vis-à-vis du filtre idéal (soit la fonction porte) sont principalement les ondulations dans la bande passante et dans la bande atténuée ainsi que la largeur de la transition.

C'est ainsi que les spécifications du filtre vont être définies par un gabarit fréquentiel linéaire ou en dB (décibels). Ce gabarit indique la ou les fréquences de coupure, la largeur de la bande de transition minimale souhaitée, le maximum d'ondulation de la bande passante et de la bande atténuée, la fréquence d'échantillonnage et éventuellement l'ordre maximal permis.

Gabarit d'un filtre

Le gabarit d'un filtre n'est autre que l'ensemble des caractéristiques du filtre, à savoir :

- la bande passante (BP) de 0 jusqu'à f_p
- la bande atténuée (ou coupée BA) de f_a jusqu'à $f_q/2$
- Le gain du filtre dans la bande passante.
- L'atténuation du filtre en bande coupée f_a .
- la largeur $\Delta f = f_a - f_p$ de la zone de transition $\Rightarrow f_c = f_a + \Delta f / 2 = (f_a + f_p) / 2$
- l'amplitude des oscillations en bande passante :
 $\delta_1 \Rightarrow A_p = 20 \log(1 + \delta_1)$ ondulation permise en BP
- l'amplitude des oscillations en bande atténuée :
 $\delta_2 \Rightarrow A_a = -20 \log(\delta_2)$ ondulation permise en BA



En pratique, plus les fréquences f_a et f_p sont proches, plus l'ordre du filtre devra être élevé. Pour un filtre idéal, ces valeurs seraient confondues.

5. Synthèse des filtres RIF par la méthode des fenêtres

L'emploi des filtres RIF peut se révéler attrayant eu égard à ses nombreux avantages : stabilité inconditionnelle (Tous les pôles sont en 0), phase linéaire possible. Néanmoins, ils présentent l'inconvénient de nécessiter un plus grand nombre de coefficients que les filtres RII pour obtenir les mêmes caractéristiques

fréquentielles à cause de l'absence de pôles hors 0. Ainsi, toute fonction de filtrage numérique stable et causale peut être approchée par la fonction de transfert d'un filtre RIF.

Rappelons que la sortie d'un filtre RIF va s'exprimer comme une combinaison linéaire d'un ensemble fini d'éléments d'entrée :

$$y(n) = \sum_{i=0}^{N-1} b_i x(n-i) \quad \text{d'où} \quad H(z) = \sum_{n=0}^{N-1} b_n \cdot z^{-n} \Rightarrow H(f) = \sum_{n=0}^{N-1} b_n \cdot e^{-2\pi j f n T_e}$$

Ainsi, les coefficients de pondération ne sont rien d'autre que les valeurs de la réponse impulsionnelle du filtre. Ces coefficients constituent les coefficients du développement en série de Fourier de la fonction de transfert $H(f)$ (voir TFTD chapitre 1)

Du fait qu'un filtre RIF possède une fonction de transfert polynomiale (non rationnelle), il ne peut être obtenu par transposition d'un filtre continu. Les deux méthodes les plus utilisées pour l'approximation des filtres numériques RIF sont alors:

- Développement par série de Fourier : cette série est ensuite tronquée par des fonctions fenêtres pour limiter la réponse impulsionnelle. Les coefficients de Fourier coïncident avec les échantillons de la réponse impulsionnelle du Filtre.
- Echantillonnage de la réponse fréquentielle : Cette méthode fait appel à la TFD. Celle-ci est appliquée aux coefficients recherchés b_i pour obtenir une suite fréquentielle qui corresponde à la réponse fréquentielle du filtre.

Il existe d'autres méthodes telles les méthodes d'optimisation qui sont basées sur la minimisation d'un critère d'erreur entre la courbe réelle et le filtre idéal [9].

La méthode de la fenêtre consiste, connaissant l'expression analytique $H(f)$ de la réponse fréquentielle continue (dont la formulation mathématique connue) à approcher, à déterminer par utilisation de la transformée de Fourier à temps discret inverse, la réponse impulsionnelle. Cette réponse temporelle non causale obtenue sera retardée pour la rendre causale [18]. Ainsi :

1. A partir du gabarit idéal du filtre, on détermine les coefficients du filtre en limitant le calcul à N valeurs réparties symétriquement autour de $n=0$. Puis, on calcule de la TFTD inverse du filtre idéal qui nous permettra de retrouver les échantillons de la réponse impulsionnelle soient les coefficients du filtre :

$$h(n) = \begin{cases} \frac{1}{f_e} \int_{-f_e/2}^{f_e/2} H(f) e^{2\pi j f n T_e} df & N \text{ impair (Filtre de type I)} \\ \frac{1}{f_e} \int_{-f_e/2}^{f_e/2} H(f) e^{-j\pi f T_e} e^{2\pi j f n T_e} df & N \text{ pair (Filtre de type II)} \end{cases}$$

Remarques :

- Le filtre de type I est le plus employé (C'est celui que nous développons dans ce qui suit) hormis pour les différenciateurs.

- Le filtre de type II possède un zéro en -1 (Nombre de zéros est impair donc répartition de zéros en paires conjuguées + un zéro en $f_e/2$ puisque une somme de cosinus), il ne peut donc être employé pour un passe-haut ou un coupe-bande.
- Il existe des filtres de type III (N impair) et IV (N pair) permettant d'obtenir une réponse impulsionnelle anti-symétrique avec déphasage linéaire également (somme de sinus au lieu de cosinus). Ils possèdent tous deux des zéros en 1 (sin en $f=0$ est nulle), leur utilisation n'est donc pas adaptée aux filtres passe-bas. En outre le filtre de type III possède également un 0 en -1, on ne peut l'employer que pour un passe-bande.

2. Cette méthode produit une série infinie de coefficients, on limite, alors la réponse impulsionnelle à N échantillons (troncature). Sachant que la troncature induit des ondulations, on peut faire appel aux fenêtres de pondération pour les atténuer. Ainsi, la réponse impulsionnelle idéale $h(n)$ sera multipliée par la fenêtre discrète $w_N(n)$ de longueur N:

$$h'_N(n) = h(n) \cdot w_N(n). \text{ (Choix de la fenêtre : voir chapitre 1)}$$

3. Il ne reste plus qu'à décaler la réponse impulsionnelle $h(n)$ pour avoir une solution causale.

Remarque : Pour le choix de f_c il faudra faire attention aux fréquences de coupure à prendre en compte. Afin d'avoir de bons résultats lors de la synthèse, ce ne sont pas les fréquences de coupure du filtre idéal qu'il faut utiliser mais il faut déplacer celles-ci afin de les centrer dans la zone de transition. Pour un passe-bas, l'augmenter de la demi zone de transition (Δf), soit $f_p + \Delta f$ et pour un passe-haut, la diminuer de la demi zone de transition. Pour un passe-bande, diminuer la première fréquence de coupure de la demi zone de transition et augmenter la seconde de la demi zone de transition, pour un rejeteur de bande, on fera l'inverse.

Exemple :

- Calcul de la réponse impulsionnelle idéale (cas N impair)

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

$$\text{On pose } f_c = (f_p + f_a)/2 \Rightarrow h(n) = \frac{1}{f_e} \int_{-f_c}^{f_c} e^{2\pi j f n T_e} df = \frac{1}{2\pi j n f_e T_e} e^{2\pi j f n T_e} \Big|_{-f_c}^{f_c}$$

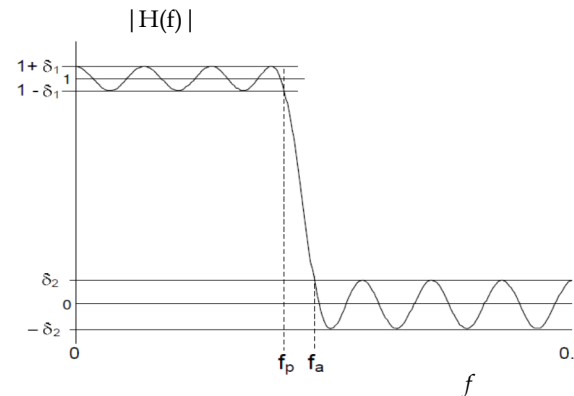
$$\Rightarrow h(n) = \frac{e^{2\pi j f_c n T_e} - e^{-2\pi j f_c n T_e}}{2\pi j n} = \frac{\sin(2\pi f_c n T_e)}{\pi n} = \frac{\sin(\pi n f_c / (f_e/2))}{\pi n}$$

On normalise les fréquences par rapport à $f_e/2$ (ou f_e) c.a.d que l'on remplace partout f_c par $f_c/(f_e/2)$, on obtient alors :

$$h(n) = \frac{\sin(\pi n f_c / (f_e/2))}{\pi n} \frac{f_c / (f_e/2)}{f_c / (f_e/2)} = f_c \frac{\sin(\pi n f_c)}{\pi n f_c}$$

Tout aussi facilement, on peut déterminer les réponses impulsionnelles d'un passe-haut ($1-H(f)$), d'un passe-bande (différence de 2 passe-bas) et d'un coupe-bande donnés.

FGE, USTHB [assiakourgli@gmail.com / <http://perso.usthb.dz/~akourgli/>]



Les valeurs de la réponse impulsionnelle idéale $h(n)$ sont données au tableau suivant. Les fréquences f_c indiquées dans ce tableau (fréquences de coupure désirées) s'expriment également en fréquences normalisées (divisées par $f_e/2$).

	Réponse impulsionnelle $h(n)$	
Type de filtre	$h(n)$ pour $n \neq 0$	$h(n)$ $n=0$
Passe-bas	$f_c \frac{\sin(\pi n f_c)}{\pi n f_c}$	f_c
Passe-haut	$-f_c \frac{\sin(\pi n f_c)}{\pi n f_c}$	$1-f_c$
Passe-bande	$f_{c2} \frac{\sin(\pi n f_{c2})}{\pi n f_{c2}} - f_{c1} \frac{\sin(\pi n f_{c1})}{\pi n f_{c1}}$	$f_{c2} - f_{c1}$
Rejécteur de bande	$f_{c1} \frac{\sin(\pi n f_{c1})}{\pi n f_{c1}} - f_{c2} \frac{\sin(\pi n f_{c2})}{\pi n f_{c2}}$	$1-(f_{c2} - f_{c1})$

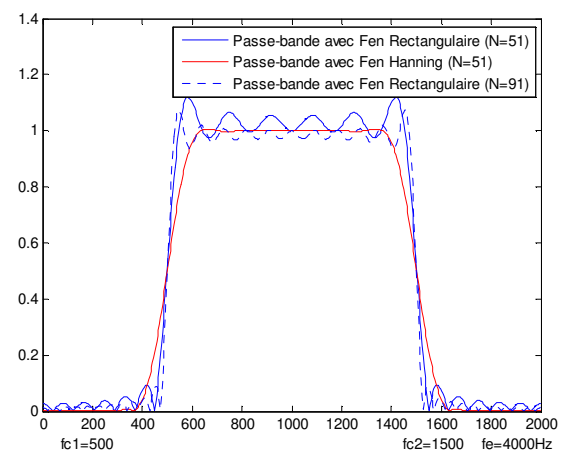
- Limitation du nombre d'échantillons à N

$$h'_N(n) = h(n) \cdot w(n)$$

$$\text{si } w(n) = \begin{cases} 1 & \text{si } |n| \leq \frac{N-1}{2} \\ 0 & \text{ailleurs} \end{cases} \Rightarrow |W(f)| = \frac{\sin(N\pi f)}{\sin(\pi f)} \quad H'_N(f) = H(f) * W(f)$$

La troncature temporelle introduit des ondulations et induit une zone de transition moins rapide déterminée par la largeur du lobe principal. Un compromis est à faire entre la raideur et l'amplitude des ondulations. Notons que cette méthode donne des ondulations de même amplitude dans la bande passante et dans la bande atténuée.

Remarque: Si N augmente, l'étendue des oscillations diminue (la réponse est plus plate) et la largeur de la bande de transition décroît (Δf faible). On peut observer que les oscillations les plus fortes tendent à se concentrer aux discontinuités. Ces dernières ne diminuent pas si N augmente : on a toujours un dépassement à peu près égal à 9% se concentrant aux points de discontinuité : c'est le phénomène de Gibbs (Voir TP n°1).



Pour diminuer les oscillations : on utilise les fenêtres de pondération qui permettent d'obtenir de forte atténuation des oscillations mais cela se fait au détriment de la largeur de la bande de transition qui devient plus grande.

Pour le choix de la fenêtre de pondération, on procédera comme suit : En fonction de l'atténuation δ_2 requise dans la spécification du filtre, on choisira le type de fenêtre $w(n)$ à utiliser. Puis en fonction de la

largeur de la zone de transition Δf (spécifiée aussi au départ) et du type de la fenêtre $w(n)$, on déterminera la longueur de la réponse impulsionnelle N .

Fenêtres $w_N(n)$	Largeur de Transition : $\Delta f (2\Delta f / f_c)$	Atténuation en bande atténuée A
$w_{\text{Rect}}(n) = \begin{cases} 1 & \text{pour } n \leq \frac{N-1}{2} \\ 0 & \text{ailleurs} \end{cases}$	$1.8/N$	21
$w_{\text{Han}}(n) = \begin{cases} 0.5 + 0.5 \cos\left(\frac{2\pi n}{N-1}\right) & \text{pour } n \leq \frac{N-1}{2} \\ 0 & \text{ailleurs} \end{cases}$	$6.2/N$	44
$w_{\text{Ham}}(n) = \begin{cases} 0.54 + 0.46 \cos\left(\frac{2\pi n}{N-1}\right) & \text{pour } n \leq \frac{N-1}{2} \\ 0 & \text{ailleurs} \end{cases}$	$6.6/N$	53
$w_{\text{Black}}(n) = \begin{cases} 0.42 + 0.5 \cos\left(\frac{2\pi n}{N-1}\right) + 0.08 \cos\left(\frac{4\pi n}{N-1}\right) & \text{pour } n \leq \frac{N-1}{2} \\ 0 & \text{ailleurs} \end{cases}$	$11/N$	74

Exemple : On veut synthétiser un filtre passe-bas de fréquence de coupure $f_c = f_e/10$ avec $\Delta f = f_e/5$ et une ondulation en bande atténuée > 50 db (voir TP n°3)

On normalise les fréquence $f_c / (f_e/2) \Rightarrow f_c = 0.2$ $\Delta f / (f_e/2) \Rightarrow \Delta f = 0.4$

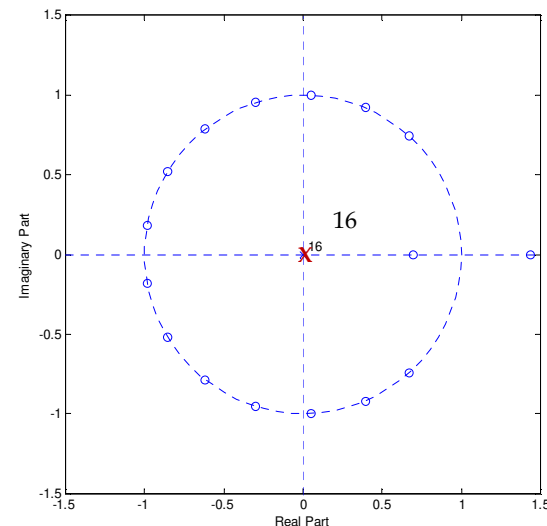
a- $h(n) = 0.2 \frac{\sin(\pi n / 5)}{\pi n / 5}$

b- On choisit $w(n)$ comme étant la fenêtre de Hamming

($A_a = 20 \log(\delta_2) = -53$) $\Rightarrow \Delta f = 6.6/N$

c- On calcule $N = 6.6 / \Delta f = 16,5$ on prend $N = 17$

d- on calcule les valeurs $h(n) = 0.2 \frac{\sin(\pi n / 5)}{\pi n / 5}$ pour $-8 \leq n \leq 8$

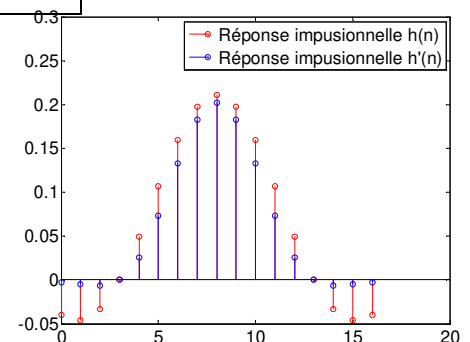


N	-8	-7	-6	-5	-4	-3	-2	-1	0
$h(n)$	-0,0399	-0,0456	-0,0329	0	0,0493	0,1064	0,1597	0,1974	0,2110
N	1	2	3	4	5	6	7	8	
$h(n)$	0,1974	0,1597	0,1064	0,0493	0	-0,0329	-0,0456	-0,0399	

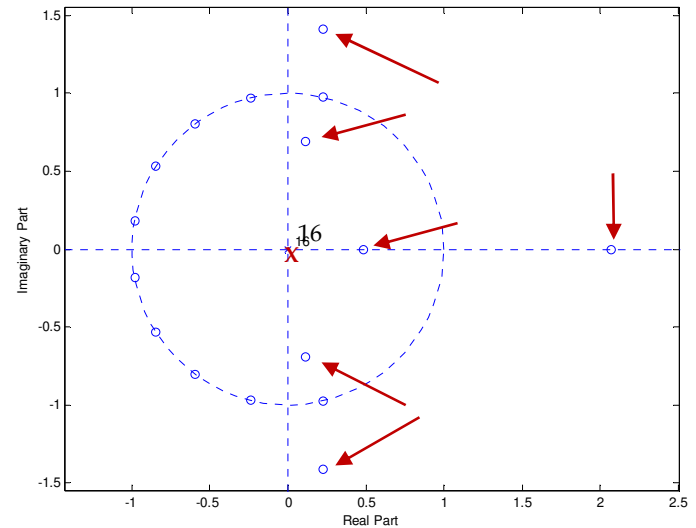
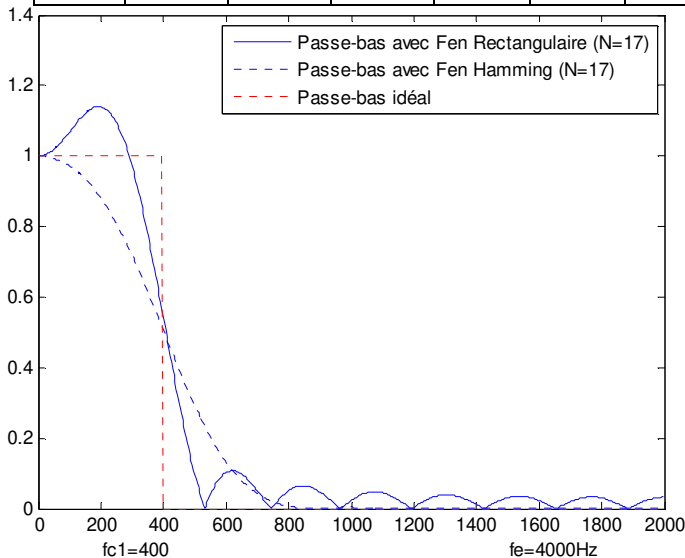
e- On multiplie $h(n)$ par $w(n)$ pour trouver $h'_N(n) = h(n) \cdot w(n)$

$$h'_N(n) = 0.2 \frac{\sin(\pi n / 5)}{\pi n / 5} \left[0.54 + 0.46 \cos\left(\frac{2\pi n}{16}\right) \right]$$

f- On translate le résultat de 8 échantillons.



N	0	1	2	3	4	5	6	7	8
$h(n)$	-0,0399	-0,0456	-0,0329	0	0,0493	0,1064	0,1597	0,1974	0,2110
$h'_N(n)$	-0,0031	-0,0050	-0,0067	0	0,0255	0,0730	0,1325	0,1826	0,2023
N	9	10	11	12	13	14	15	16	
$h(n)$	0,1974	0,1597	0,1064	0,0493	0	-0,0329	-0,0456	-0,0399	
$h'_N(n)$	0,1826	0,1325	0,0730	0,0255	0	-0,006	-0,005	-0,0030	



On peut remarquer l'emplacement de zéros autour du premier 0 en 1 permet d'atténuer les lobes secondaire et de maintenir une réponse cste autour du zéro.

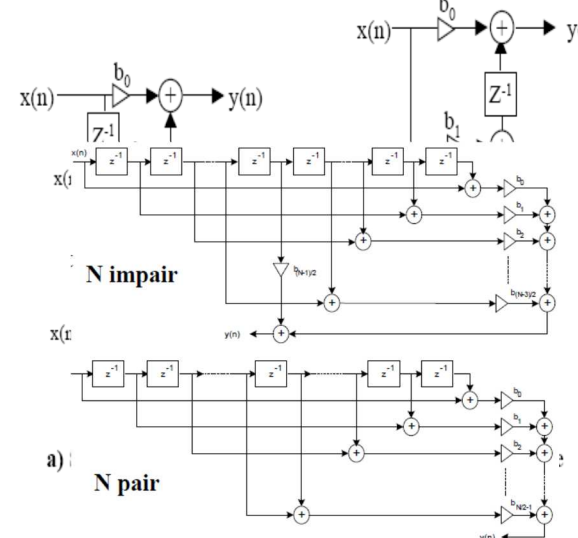
Constitution et réalisation des filtres numériques RIF

La réalisation concrète d'un filtre numérique consistera en fait à matérialiser l'algorithme de calcul pour la structure retenue. On aura la possibilité de travailler : Soit en logique câblée (assemblage d'organes logiques, tels que portes, mémoires, etc ...), soit en logique programmée (organisation autour d'un processeur de traitement du signal (DSP) ou, même, utilisation d'un microprocesseur(micro-ordinateur) standard).

La structure canonique directe (transversale ou non réursive) est donnée ci-contre:

Un filtre RIF nécessite $(N-1)$ opérations de multiplication, N opération d'addition pour chaque nouvel échantillon à filtrer.

On peut également exprimer la complexité en nombre de multiplication-accumulation qui, dans le cas d'un filtrage RIF, vaut N [9]. Le coût mémoire d'un filtrage RIF est de $2(N+1)$ [($N+1$) coefficients b_i et ($N+1$) points mémoire pour le vecteur des entrées $x(i)$]. Si la fréquence d'échantillonnage du signal d'entrée vaut f_e , cela signifie que le filtre devra être réalisé en un temps T_{calcul} inférieur à $T_e = 1/f_e$. Pour un filtre à retard de groupe constant, la structure sera la suivante :



6. Synthèse des filtres récurrents RII

Le principe de définition du cahier des charges d'un filtre récurrent (RII) se déroule de la même manière que pour un filtre non récurrent (RIF) mais la synthèse se fera de différente manière. L'intérêt d'employer un filtre RII réside principalement dans la possibilité d'obtenir une bande de transition étroite pour un ordre raisonnable bien qu'il présente d'une part, un risque d'instabilité du à une grande sensibilité numérique des coefficients mais que l'on peut toutefois contrôler en déterminant une structure mieux adaptée, et d'autre part, une variation de phase fortement non linéaire.

Rappelons que pour un filtre RII, la sortie s'exprime comme une combinaison linéaire d'un ensemble fini d'éléments d'entrées et de sortie:

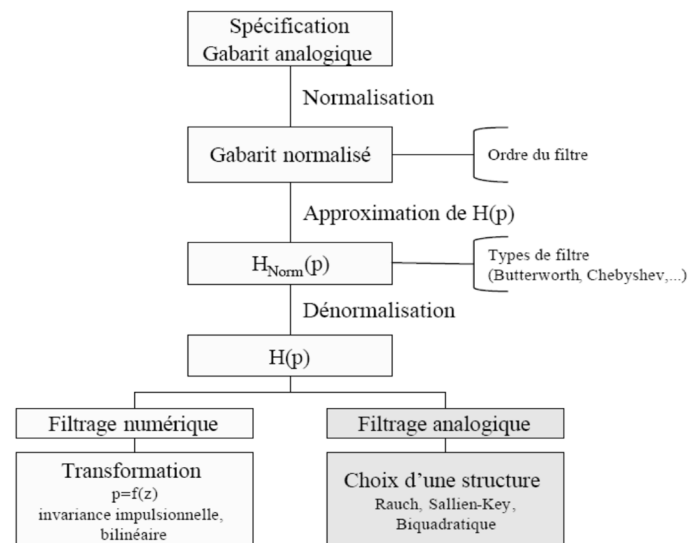
$$y(n) = \sum_{i=0}^N b_i x(n-i) - \sum_{i=1}^M a_i y(n-i) \quad \text{d'où} \quad H(z) = \frac{\sum_{i=0}^N b_i \cdot z^{-i}}{1 + \sum_{i=1}^M a_i \cdot z^{-i}}$$

La méthode directe consiste à placer des pôles et des zéros aux fréquences utiles mais la plus courante (indirecte dite aussi de transposition) est l'utilisation des méthodes de synthèse des filtres analogiques aboutissant à une fonction $H(p)$ correspondant aux spécifications. Une fonction permettant le passage du plan p dans le plan z ($p = f(z)$) est ensuite utilisée pour obtenir $H(z)$. Cette fonction doit maintenir la stabilité du filtre analogique et maintenir, au mieux, les caractéristiques de la réponse fréquentielle $H(f)$ du filtre numérique.

• Rappels sur les filtres analogiques

Il existe de nombreuses méthodes permettant de synthétiser un filtre numérique récurrent à partir d'un filtre analogique pris comme modèle [18]:

- le filtre doit avoir une réponse impulsionnelle ou indicelle imposée : ce sont les méthodes de l'invariance impulsionnelle.
- le filtre doit avoir une réponse fréquentielle entrant dans un gabarit donné : c'est la transformation bilinéaire.

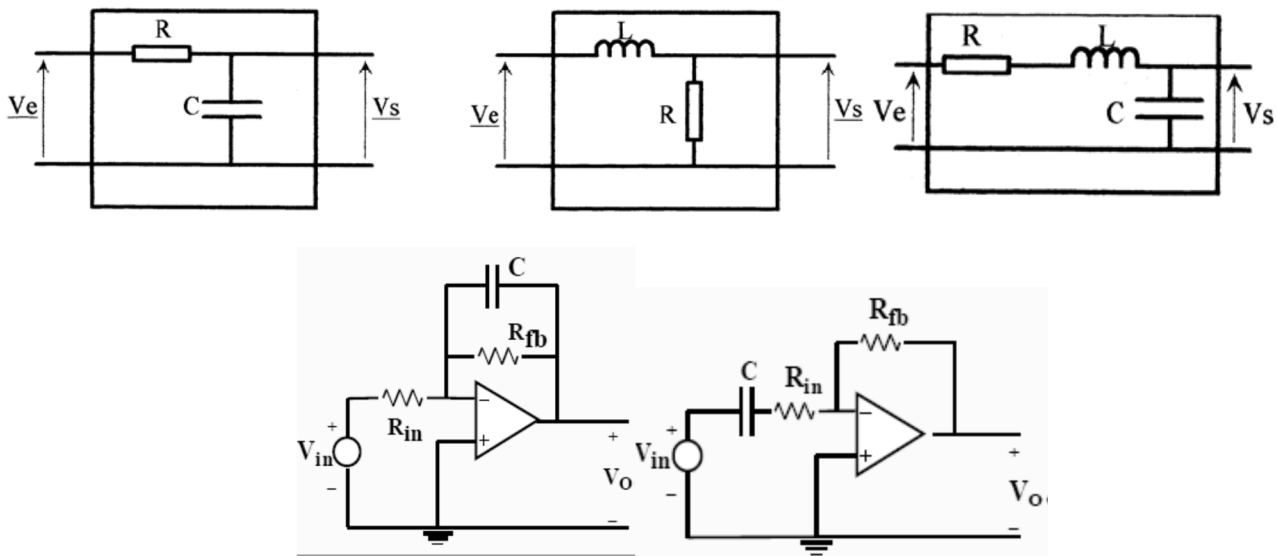


Pour concevoir un filtre analogique, on peut employer des filtres passifs obtenus par combinaison de résistances, de condensateurs et/ou de bobines ou encore utiliser des filtres actifs comportant un élément amplificateur (transistor, AO, etc.) qui permet donc de modifier les amplitudes des signaux. Ainsi, un filtre passif de base (de premier ordre) peut être composé d'une cellule RC ou d'une cellule RL.

Il faut noter que :

- les filtres numériques sont limités à des fréquences < 100MHz
- les filtres passifs (L,C, quartz, etc ;) sont utilisés pour les hautes fréquences
- les filtres actifs (R,C, ALI) utilisent des amplificateurs linéaires intégrés (ALI) limités à 1Mhz
- les filtres à capacité commutés (R et C intégrés, ALI, Interrupteur commandé MOS) ont des fréquences (<10 Mhz) programmables

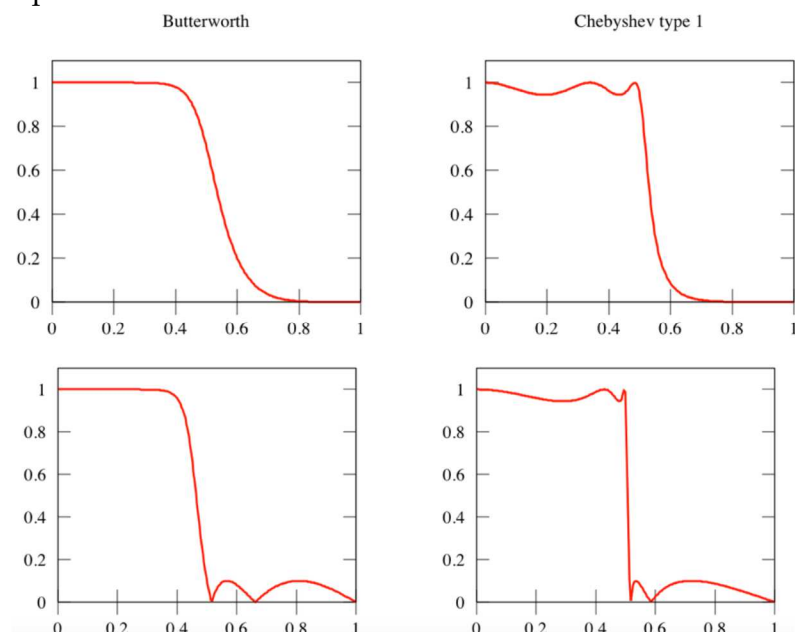
Ci-dessous, sont donnés quelques exemples de structure pour des filtres passifs passe-bas de premier et second ordre suivis de filtres actifs passe-bas de premier et second ordre aussi.



Pour obtenir des filtres d'ordre N plus élevé, on emploiera des filtres du premier et du deuxième ordremis en cascade. Les pentes asymptotiques seront proportionnelles au nombre de cellules ($+N \uparrow$ et $+\Delta f \downarrow$). A noter également que les filtres passe-bas ou passe-haut peuvent avoir un nombre entier d'ordre (1, 2, 3...) tandis que les filtres passe-bande ou coupe-bande ne peuvent qu'avoir un ordre pair (2, 4, 6, ...) car ils sont formés de paires de cellules : 2 cellules RC ou une cellule RC et une cellule RL.

Pour synthétiser des filtres analogiques répondant à un gabarit donné, on choisira parmi un ensemble de filtres analogiques testés et éprouvés donc connus pour leurs propriétés en termes de pente d'atténuation et d'ondulation dans la bande passante et atténuée telles que :

- Filtres de Butterworth : Coupure peu raide mais gain constant en bande passante
- Filtres de Tchebychev (Chebyshev) : Raideur de coupure importante mais ondulations dans la bande passante (Chebyshev 1) ou atténuée (Chebyshev 2) et simple à mettre en oeuvre
- Filtres de Caue (dit aussi elliptiques) : Coupure extrêmement raide mais ondulations dans la bande passante et atténuée mais circuits plus complexes à réaliser.
- Filtres de Bessel : Retard de groupe constant mais mauvaise sélectivité même pour ordre élevé.



Remarque : Le filtre de Bessel est intéressant pour filtrer les signaux large bande (modulations HF haut débit : PSK, 8-PSK, OFDM...) en préservant les phases. Il n'a pas intérêt dans le domaine du filtrage numérique.

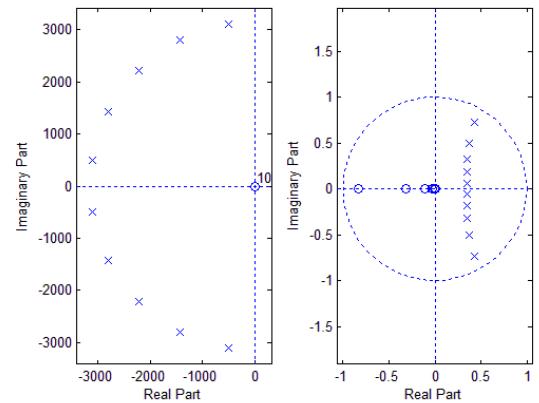
1. Un filtre de Butterworth est caractérisé par le fait que la réponse d'amplitude est maximale plate dans la bande passante et monotonement décroissante à partir d'une certaine fréquence (fréquence de coupure).

L'amplitude d'un filtre analogique Passe-bas de Butterworth d'ordre N est définie par l'expression :

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

ω_c est la pulsations de coupure limites et ε est un paramètre de conception qui fixe la région de tolérance dans la bande passante $\delta_1 = 1/(1+\varepsilon^2)^{1/2}$. Le filtre analogique aura N pôles placés sur le cercle unité (à partie réelle négative).

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	$(p^2 + 1.6180p + 1)(p^2 + 0.6180p + 1)(p + 1)$
6	$(p^2 + 1.9318p + 1)(p^2 + \sqrt{2}p + 1)(p^2 + 0.5176p + 1)$



L'ordre N est déterminé par la zone de transition ($\omega_a - \omega_p$) et les ondulations permises en bande passante $A_p = 20 \log(1 + \delta_1)$ et bande atténuée ($A_a = -\log \delta_2$)

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

$$N \geq \frac{\log\left(10^{\frac{A_a}{10}} - 1\right) / \left(10^{\frac{A_p}{10}} - 1\right)}{2 \log(\omega_a / \omega_p)} = \frac{\log\left(\left(\frac{1}{\delta_2^2} - 1\right) / \left(\frac{1}{\delta_1^2} - 1\right)\right)}{2 \log(\omega_a / \omega_p)}$$

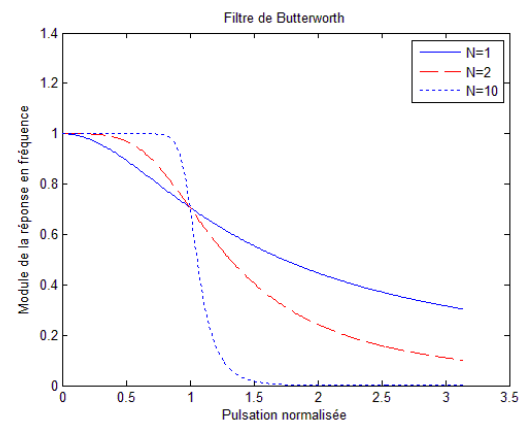
Remarque : Plus N est élevé, plus grande est la sélectivité (zone de transition Δf étroite) mais un filtre RII de fait de sa récursivité a une période d'initialisation avant le régime permanent qui augmente avec l'ordre du filtre. En outre, son déphasage augmente avec l'ordre du filtre.

Exemple : On souhaite réaliser un filtre passe-bas de gain unitaire ne comportant pas d'oscillations ni dans la bande passante, ni dans la bande atténuée tel que : $A_p = 1\text{db}$, $A_a = 40\text{db}$, $f_p = 1\text{kHz}$ et $f_a = 3\text{kHz}$.

Déterminer le filtre et N.

$$N \geq \frac{\log(10^4 / 0.2589)}{2 \log(3/1)} = 4.80 \approx 5$$

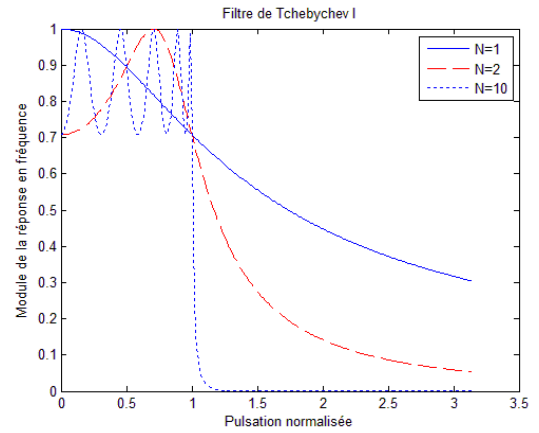
2. Un filtre de Tchebychev est caractérisé généralement par une coupée. La forme analytique du module de réponse fréquentielle d'un filtre analogique de Tchebychev (ou Chebyshev) d'ordre N est donnée par :



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

Où ε est un paramètre réel inférieur à l'unité qui détermine l'amplitude des oscillations dans la bande passante et où $T_N(x)$ est le polynôme de Tchebychev d'ordre N, définie par :

$$T_N(x) = \begin{cases} \cos[N \arccos(x)] & \text{pour } 0 \leq |x| \leq 1 \\ \cosh[N \operatorname{arccosh}(x)] & \text{pour } |x| \geq 1 \end{cases}$$



L'ordre N est déterminé par la zone de transition ($\omega_a - \omega_p$) et les ondulations permises en bande passante $A_p = 20 \log(1 + \delta_1)$ et bande atténuée ($A_a = -\log \delta_2$) *

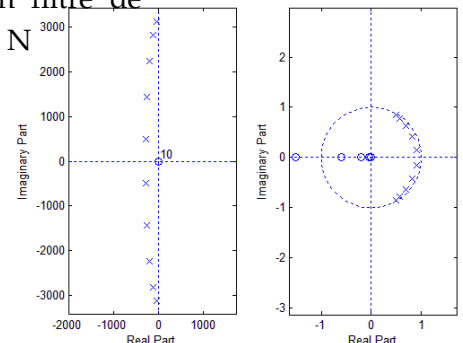
$$N \geq \frac{\log \left(\sqrt{\left(10^{\frac{A_a}{10}} - 1\right)} / \sqrt{\left(10^{\frac{A_p}{10}} - 1\right)} + \sqrt{\left(10^{\frac{A_a}{10}} - 1\right)} / \sqrt{\left(10^{\frac{A_p}{10}} - 1\right)} - 1 \right)}{\log \left(\omega_a / \omega_p + \sqrt{(\omega_a / \omega_p)^2 - 1} \right)}$$

$$N \geq \frac{\log \left(2 \cdot 10^{\frac{A_a}{20}} / \sqrt{\left(10^{\frac{A_p}{10}} - 1\right)} \right)}{\log \left(\omega_a / \omega_p + \sqrt{(\omega_a / \omega_p)^2 - 1} \right)}$$

Exemple

On souhaite réaliser un filtre passe-bas de gain unitaire avec un filtre de Tchebychev tel que : $A_p = 1\text{dB}$, $A_a = 40\text{dB}$, $f_p = 1\text{kHz}$ et $f_a = 3\text{kHz}$. Déterminer N

$$N \geq \frac{\log \left(2 \cdot 10^{\frac{A_a}{20}} / \sqrt{\left(10^{\frac{A_p}{10}} - 1\right)} \right)}{\log \left(\omega_a / \omega_p + \sqrt{(\omega_a / \omega_p)^2 - 1} \right)} = \frac{2.100 / 0.5089}{\sqrt{3} + \sqrt{8}} = 3.39 \approx 4$$



Dans la pratique, on utilise trois valeurs d'ondulation, $\varepsilon = 0.1 \text{ dB}$, 0.5 dB et 1 dB . Pour ces 3 valeurs $H_N(p)$ est donnée :

Ordre du filtre	Dénominateur (le numérateur est à 1)	Ordre du filtre	Dénominateur (le numérateur est à 1)
2	$0.3017 p^2 + 0.7158 p + 1$	2	$0.6595 p^2 + 0.9402 p + 1$
3	$(0.5918 p^2 + 0.5736 p + 1)(1.031 p + 1)$	3	$(0.8753 p^2 + 0.5483 p + 1)(1.596 p + 1)$
4	$(0.7518 p^2 + 0.3972 p + 1)(1.6053 p^2 + 2.0475 p + 1)$	4	$(0.9402 p^2 + 0.3297 p + 1)(2.8057 p^2 + 2.3755 p + 1)$
5	$(0.8368 p^2 + 0.2787 p + 1)(1.5725 p^2 + 1.3712 p + 1)(1.855 p + 1)$	5	$(2.0974 p^2 + 1.2296 p + 1)(0.9654 p^2 + 0.2161 p + 1)(2.759 p + 1)$

Ordre du filtre	Dénominateur (le numérateur est à 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	$(2.3293 p^2 + 1.0911 p + 1)(1.0118 p^2 + 0.1610 p + 1)(3.454 p + 1)$

On rappelle que les fonctions de transfert $H_N(p)$ des filtres analogiques polynômiaux (Butterworth, Tchebychev, Bessel, Caue, etc.) sont données pour des fréquences de coupure normalisées et uniquement pour des filtres passe-bas [2].

Passe-bas	$p = p / \omega_A$
Passe-haut	$p = \omega_A / p$

Passe-bande	$p = \frac{1}{B} \left(\frac{p}{\omega_A} + \frac{\omega_A}{p} \right)$	Pulsation centrale $\omega_A = \sqrt{\omega_{A1} \omega_{A2}}$
Coupe-bande	$p = \left[\frac{1}{B} \left(\frac{p}{\omega_A} + \frac{\omega_A}{p} \right) \right]^{-1}$	Largeur de Bande $B = (\omega_{A2} - \omega_{A1})$

A partir de l'expression de $H_N(p)$ normalisée, on dénormalise en remplaçant p par les valeurs données au tableau précédent permettant d'aboutir à une fonction de transfert dénormalisée $H(p)$.

La Méthode de la transformation bilinéaire a pour objectif de faire coïncider au mieux les domaines analogique et numérique. Les filtres qui en sont dérivés sont plus stables que sont obtenus à travers l'emploi de la méthode de la variance impulsionnelle. Mais, en contrepartie, elle introduit une distorsion sur l'axe des fréquences.

On sait qu'un filtre analogique est caractérisé par sa fonction de transfert $H(p)$ et un filtre numérique est défini par sa fonction de transfert $H(z)$ avec $z = e^{2\pi j f T_e} = e^{p T_e}$. Ce qui implique que $p = \ln(z)/T_e$. Dans les deux cas, cette transformation entraîne une relation non linéaire entre les fréquences f_A du domaine analogique et les fréquences f_N du domaine numérique.

Pour conserver le caractère polynomial des fonctions de transfert, une approximation de $\ln(z)$ par les séries de Laurent est employée :

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

En ne conservant que le terme du premier ordre, on définit la transformation bilinéaire :

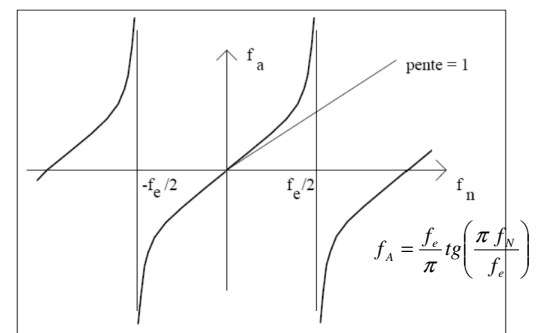
$$p = \frac{\ln(z)}{T_e} \approx \frac{2}{T_e} \frac{1-z^{-1}}{1+z^{-1}} = \frac{2}{T_e} \frac{z-1}{z+1}$$

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

Ainsi, on peut montrer qu'on obtient la correspondance entre fréquence analogique f_A (ou pulsation analogique ω_A) et la fréquence numérique f_N (ou ω_N) par :

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

Cette équation montre qu'il n'y a pas égalité entre pulsation analogique et pulsation discrète et que la relation les liant n'est pas non plus linéaire puisqu'il y a distorsion des fréquences, y compris du retard de groupe.



Pour faire la synthèse d'un filtre numérique par la transformation bilinéaire, on procède comme suit :

- On définit les caractéristiques souhaitées du filtre numérique (fréq d'échantillonnage, de coupure, etc.)

- On calcule les pulsations analogiques ω_A correspondant aux pulsations numériques $\omega_A = \frac{2}{T_e} \operatorname{tg}\left(\frac{\pi f_N}{f_e}\right)$

- On détermine le gabarit du filtre analogique $H_N(p)$ normalisée d'ordre n (Chebyshev, Butterworth, etc.) qui servira de modèle au filtre numérique et on écrit la fonction de transfert dénormalisée $H(p)$ de ce filtre analogique (qu'il faut recalculer en fonction de ω_A).

- On applique la transformation bilinéaire à $H(p)$ en remplaçant $p = \frac{2}{T_e} \frac{1-z^{-1}}{1+z^{-1}}$ ce qui donne la fonction $H(z)$.

Exemple 1: On désire concevoir un filtre passe-bas numérique de premier ordre à partir d'une fonction de transfert d'un filtre RC dans le domaine continu ($H_N(p)=1/(1+p)$). La fréquence de coupure désirée est $f_N=30$ Hz et la fréquence d'échantillonnage est $f_e=150$ Hz.

$$\text{On calcule d'abord } \omega_A = \frac{2}{T_e} \operatorname{tg}\left(\frac{\pi}{5}\right) = \frac{2}{T_e} 0.7265 \text{ puis } H(p) = \frac{1}{(1+p)} \Big|_{p=p/\omega_A} = \frac{\frac{2}{T_e} 0.7265}{p + \frac{2}{T_e} 0.7265}$$

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

$$\text{On peut remarquer que } f_A = \frac{2}{T_e(2\pi)} \operatorname{tg}\left(\frac{\pi}{5}\right) = \frac{f_e}{\pi} 0.7265 = 34.69 \neq f_N = 30$$

Exemple 2: On désire réaliser un filtre numérique passe-bas du second ordre avec les caractéristiques suivantes [18]: fréquence de coupure $f_N= 500$ Hz (gain de 1), fréquence d'échantillonnage $f_e=5$ kHz,

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

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

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

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

Pour un filtre numérique passe-bas ou passe-haut, on peut déterminer $H(z)$ directement à partir de $H_N(p)$ en utilisant la pulsation normalisée $\Omega_c = \text{tg}\left(\frac{\pi f_N}{f_e}\right)$ et remplacer p dans $H_N(p)$ comme suit:

- pour un passe-bas par $p = \frac{1}{\Omega_c} \frac{z-1}{z+1}$ soit $H(z) = H_N(p)|_{p=\frac{1}{\Omega_c} \frac{z-1}{z+1}}$

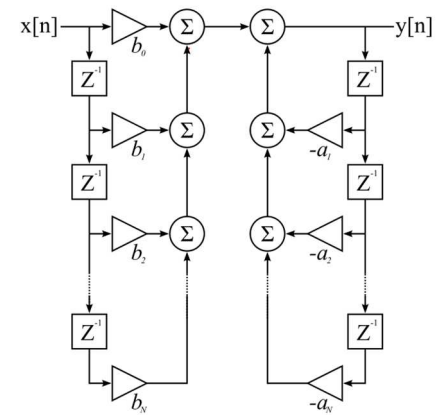
- pour un passe-haut par $p = \Omega_c \frac{z+1}{z-1}$ soit $H(z) = H_N(p)|_{p=\Omega_c \frac{z+1}{z-1}}$

- pour un passe bande ou un rejeteur de bande, la détermination de ω_A s'obtient par la pulsation centrale $\omega_A = \sqrt{\omega_{A1}\omega_{A2}}$

Remarque: Puisqu'il n'est pas possible de faire coïncider précisément la réponse fréquentielle du filtre analogique (sur $[0, +\infty[$ et numérique (sur $[0, f_e/2]$) la transformation bilinéaire aura pour objet de faire coïncider au mieux les deux réponses. Cette méthode donne d'assez bons résultats à condition que les fréquences caractéristiques ne soient pas trop proches de la demi-fréquence d'échantillonnage.

• Réalisation du filtre RII

Un filtre RII nécessite $(2N+1)$ opérations de multiplication, $2N$ opérations d'addition pour chaque nouvel échantillon à filtrer ou $(2N+1)$ MAC (Multiplication-Accumulation). Le coût mémoire d'un filtre RII en structure directe est de $(4N+3)$ mémoire $[(2N+1)$ coefficients b_i et $2(N+1)$ points mémoire ou retard pour les vecteurs des entrées $x(n)$ et des sorties $y(n)$ [9].



7. Les filtres multi-cadences

Le traitement multi-cadences implique plusieurs fréquences d'échantillonnage. Il peut avoir plusieurs motivations. Il peut être le fruit de contraintes liées à une application dans laquelle les flux de signaux numériques à traiter et à générer doivent l'être à des fréquences d'échantillonnage différentes [20]. Cette situation peut se rencontrer par exemple dans les applications audio où co-existent plusieurs fréquences d'échantillonnage : 32kHz (diffusion); 44,1 kHz (disque compact numérique); 48kHz (bande sonore numérique); 96kHz; ...

Cette approche peut aussi permettre d'améliorer les caractéristiques d'implantation d'un algorithme en adoptant pour chacune de ses étapes un échantillonnage que l'on appelle critique dans le sens où la fréquence de traitement est choisie égale à la fréquence de Nyquist. Ainsi la chaîne de traitement correspondant à l'émetteur bande de base d'un modulateur DPQSK où l'on a trois fréquences de traitement : la fréquence bit, la fréquence symbole et la fréquence d'échantillonnage [20]

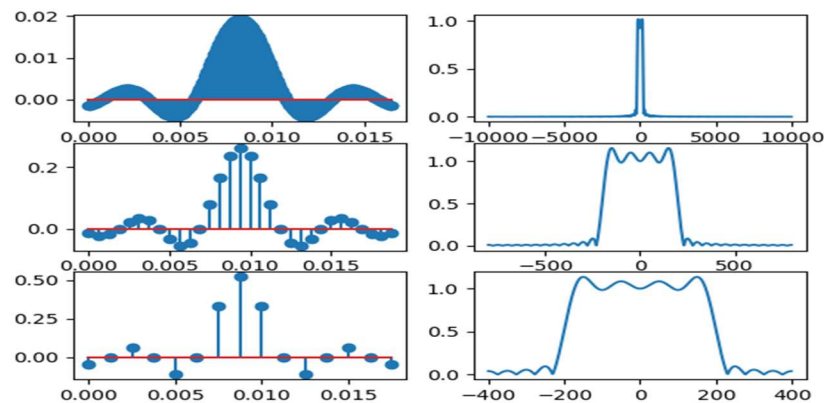
Dans les systèmes considérés aux chapitres précédents, n'était considéré qu'une seule fréquence ou cadence d'échantillonnage $f_e = 1/T_e$. Mais il arrive que la fréquence maximale f_{max} soit diminuée (filtrage passe-bas) ou augmentée (le cas d'une modulation). Dans les deux cas, il est plus judicieux d'adapter la fréquence d'échantillonnage afin de minimiser le temps de calcul. Il s'agira d'une décimation ou sous-échantillonnage

(augmentation de f_{max}) et d'une interpolation ou sur-échantillonnage (diminution de f_{max}). A noter, également que le sur-échantillonnage est souvent utilisé pour augmenter le SNR dans les opérations de CNA. Tous deux interviennent dans le changement de fréquence et dans les techniques des bancs de filtres

Exemple : Supposons que l'on veuille synthétiser un filtre passe-bas avec les spécifications suivantes : $f_p=100\text{Hz}$ $f_a=300\text{Hz}$ $A_a>50\text{ dB}$ $f_e=20\text{kHz}$.

L'emploi de la fenêtre de Hamming nous donne $N=331$, soit un nombre de coefficients important. Or le filtre occupe une petite portion de la bande passante (passe-bas avec $f_c=1\% f_e$). On peut donc sous-échantillonner et puis sur-échantillonner sans perte d'informations.

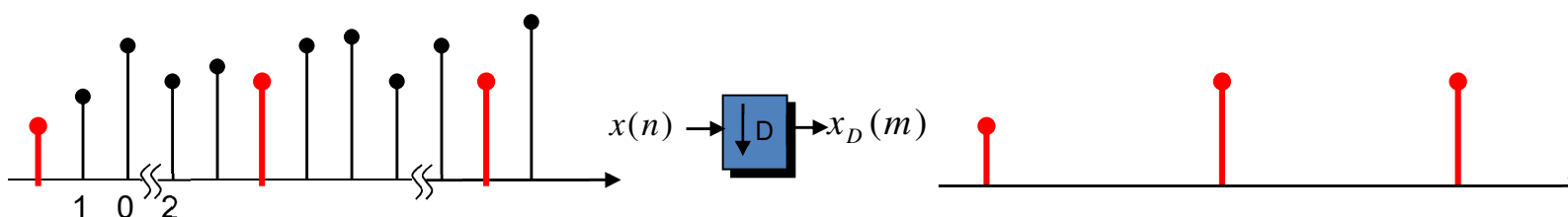
Ainsi en prenant $f_e=f_e/25=800\text{Hz}$ cela nous donnerait $N=15 \ll 331$ et on obtient, ainsi, un filtre plus rapide. C'est le principe des bancs de filtres qui revient à diviser la bande passante en plusieurs filtres à bande étroite donc des filtres que l'on pourra facilement décimer ce qui représente donc un gain de temps.



Sous échantillonnage et sur-échantillonnage

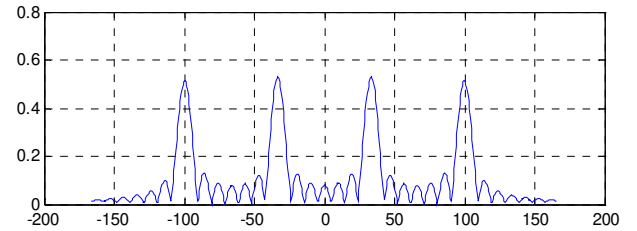
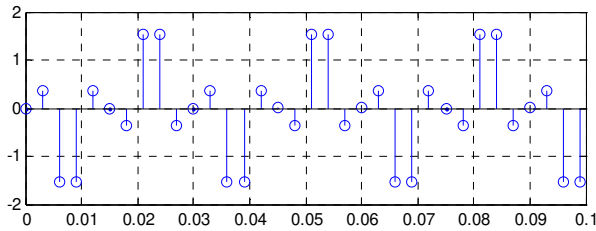
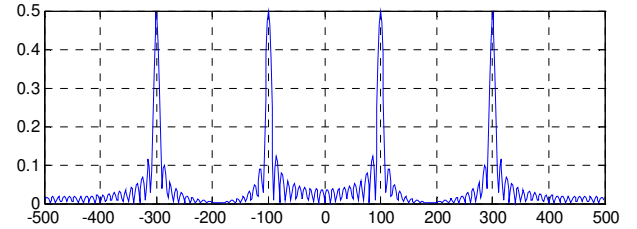
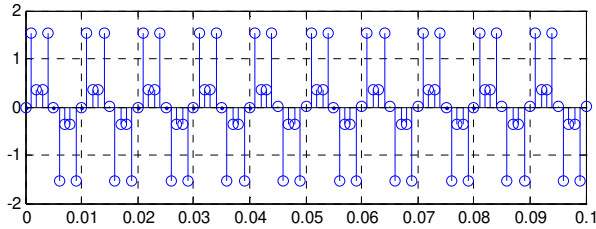
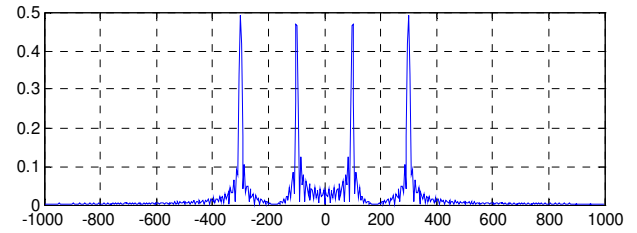
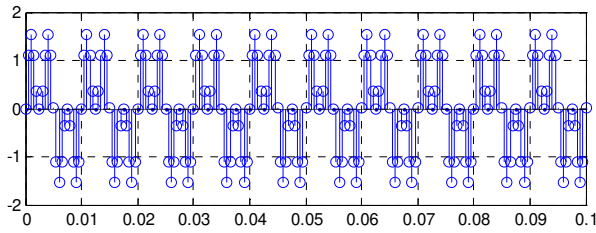
Rappelons que lorsqu'on souhaite échantillonner un signal, nous devons respecter le théorème de Shannon à savoir prendre une fréquence d'échantillonnage $f_e/2 > f_{max}$. Si ce signal a subi des traitements qui ont eu pour résultat, entre autres, la modification de la fréquence maximale. On parlera alors de sous-échantillonnage ou de sur-échantillonnage.

Sous-échantillonnage : Il consiste à fabriquer à partir du signal original x un signal x_D comportant moins d'échantillons que le signal d'origine. Il faut "enlever" des échantillons, d'où le nom décimation (En anglais : down sampling). On parle de sous-échantillonnage d'un facteur D quand le signal d'origine x comporte N échantillons et le signal sur-échantillonné x_D en comporte N/D , avec $x_D(n) = x(n/D)$

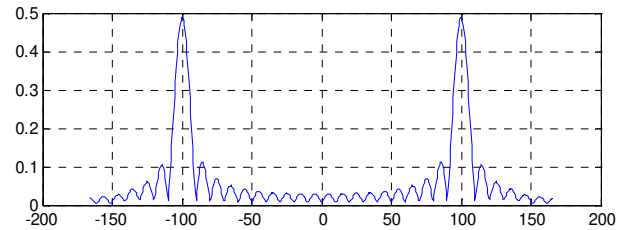
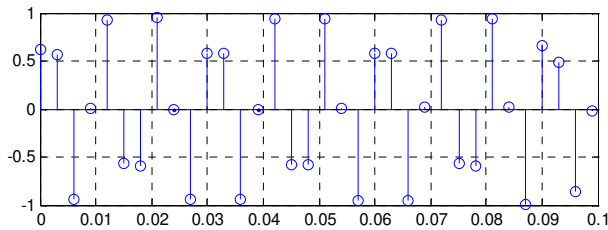
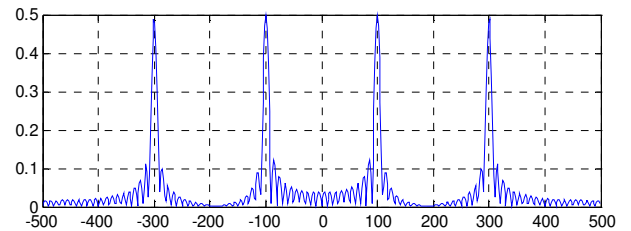
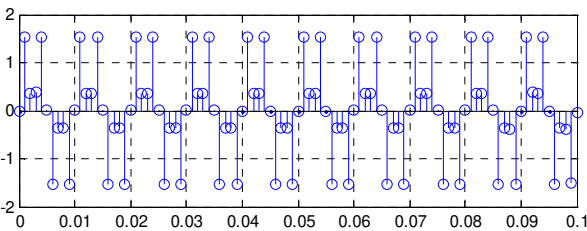
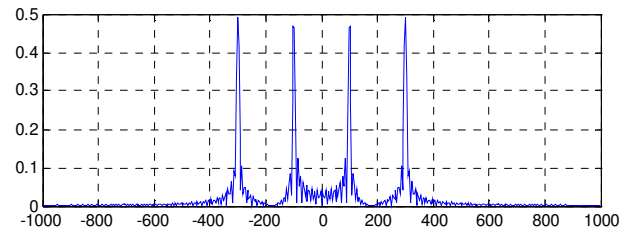
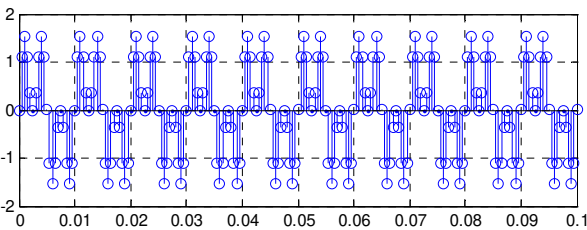


Le sous-échantillonnage naïf consiste à se débarrasser de $D - 1$ échantillons sur D . Cela revient à devoir échantillonner directement à f_e/D , mais rien ne garantit que $f_e/D > 2 f_{max}$. Il y a donc un risque de repliement spectral.

Exemple : Le signal suivant est le résultat de la somme de 2 sinusoides de fréquences 100 et 300Hz. On le décime de 2 et de 6. On remarque que lorsque $f_e=f_e/6$ il y a repliement puisque la condition de Shannon n'est plus respectée.



Pour sous-échantillonner correctement, il va falloir supprimer le contenu fréquentiel de $x(n)$ entre $f_e/2D$ et f_e/D puisque le spectre devient périodique de période f_e/D et ce avant la décimation.



Dans ce cas, nous avons pour la fonction de transfert $X_D(z) = \sum_{-\infty}^{+\infty} x_D(m)z^{-m} = \sum_{-\infty}^{+\infty} x(n/D)z^{-n/D}$

Rappels :

Peigne de Dirac : $\sum_{n=-\infty}^{\infty} \delta(t - nT_e)$ périodique de période $T_e \Rightarrow$ décomposable en série de Fourier telle que $C_n = \frac{1}{T_e} \Rightarrow \sum_{n=-\infty}^{\infty} \delta(t - nT_e) = \frac{1}{T_e} \sum_{n=-\infty}^{\infty} e^{2\pi j n \frac{t}{T_e}}$ En discret $\sum \delta(n - mM) = \frac{1}{M} \sum_{m=0}^{M-1} e^{2\pi j m \frac{n}{M}}$

La décimation de M est équivalente à un échantillonnage de M $\Rightarrow y(m) = y\left(\frac{n}{M}\right) = \sum x(n) \cdot \delta(n - mM)$

► Fonction de transfert $Y(z) = \sum_{n=-\infty}^{\infty} \left(x(n) \frac{1}{M} \sum_{k=0}^{M-1} e^{\frac{2\pi j k n}{M}} \right) z^{-n/M} = \frac{1}{M} \sum_{k=0}^{M-1} \left(\sum_{n=-\infty}^{\infty} x(n) \left(e^{-\frac{2\pi j k}{M}} z^{\frac{1}{M}} \right)^{-n} \right)$

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

► Réponse en fréquence

$y(m) = x(n) \cdot \sum \delta(n - mM) \Rightarrow Y(f) = X(f) * \frac{f_e}{M} \sum_{m=0}^{M-1} \delta\left(f - \frac{mf_e}{M}\right)$

$\Rightarrow Y(f) = \frac{f_e}{M} \sum_{m=0}^{M-1} X\left(f - \frac{mf_e}{M}\right) \Rightarrow$ **Périodisation de $\frac{f_e}{M}$**

Exemples:

- Décimation de 4 $Y(z) = \frac{1}{4} \sum_{k=0}^3 X\left(W_4^k z^{\frac{1}{4}}\right)$ avec $W_4 = e^{-\frac{2\pi j}{4}}$

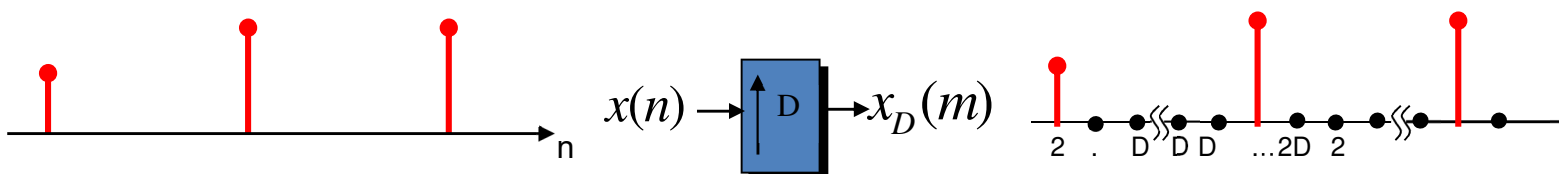
$Y(z) = \frac{1}{4} \left(X\left(W_4^0 z^{\frac{1}{4}}\right) + X\left(W_4^1 z^{\frac{1}{4}}\right) + X\left(W_4^2 z^{\frac{1}{4}}\right) + X\left(W_4^3 z^{\frac{1}{4}}\right) \right)$

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

o Décimation de 2

$Y(z) = \frac{1}{2} \sum_{k=0}^1 X\left(W_2^k z^{\frac{1}{2}}\right)$ avec $W_2 = e^{-\frac{2\pi j}{2}}$ $Y(z) = \frac{1}{2} \left(X\left(z^{\frac{1}{2}}\right) + X\left(-z^{\frac{1}{2}}\right) \right) \Rightarrow Y(f) = \frac{1}{2} \left[X(f) + X\left(f - \frac{f_e}{2}\right) \right]$

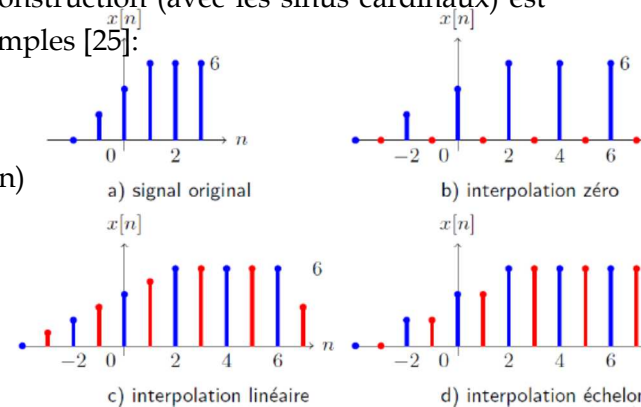
Sur-échantillonnage : Il consiste à fabriquer à partir du signal original x un signal x_M un signal de même durée mais comportant davantage d'échantillons que le signal d'origine. Il faut "rajouter" des échantillons, d'où le nom d'interpolation (en anglais : upsampling). On parle de sur-échantillonnage d'un facteur M quand le signal d'origine x comporte N échantillons et le signal sur-échantillonné en comporte MN, avec $x_M(Mn) = x(n)$.



Théoriquement, si le signal d'origine a été bien échantillonné, la reconstruction parfaite est possible, donc le sur-échantillonnage aussi. Cependant, la formule de reconstruction (avec les sinus cardinaux) est compliquée à mettre en pratique. On fait appel à des idées plus simples [25]:

- Intcaler des zéros (interpolation zéro)
- Recopier la valeur de l'instant précédent (interpolation échelon)
- Tracer une droite entre deux échantillons disponibles

Adjacents (interpolation linéaire)



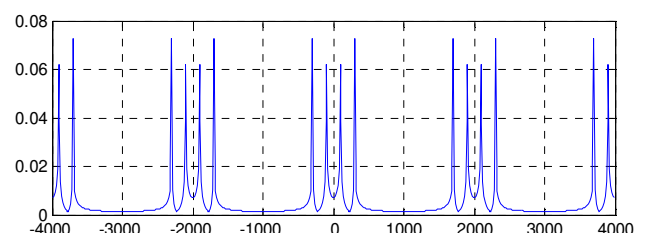
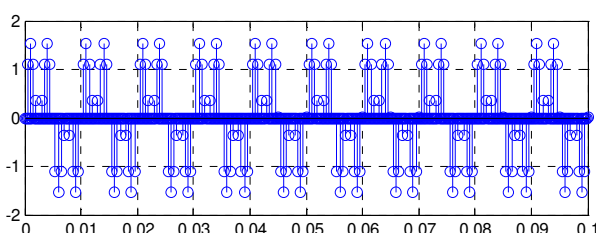
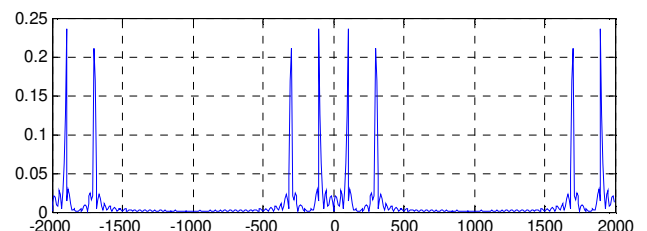
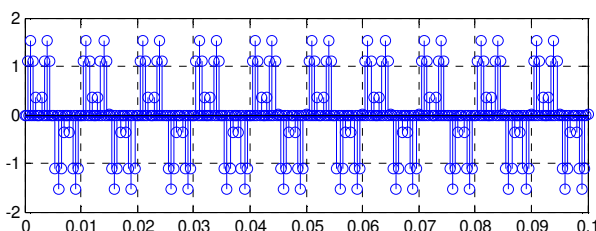
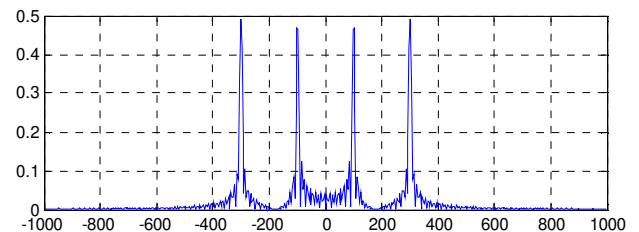
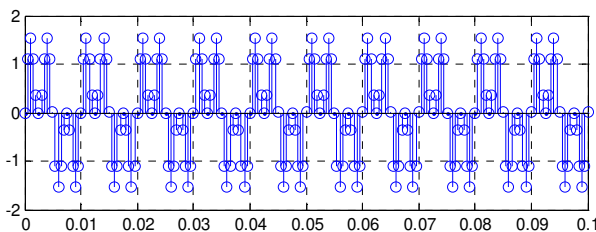
Le signal d'origine n'a pas de contenu fréquentiel au-delà de $f_e/2$. Le processus d'interpolation est susceptible d'en ajouter. On le fait donc suivre d'un filtrage passe-bas à la coupure $f_e/2$ pour supprimer le contenu ajouté. C'est la raison pour laquelle la méthode d'interpolation n'a pas beaucoup d'importance.

Exercice d'application : Vérifier que l'interpolation zéro préserve le spectre du signal.

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

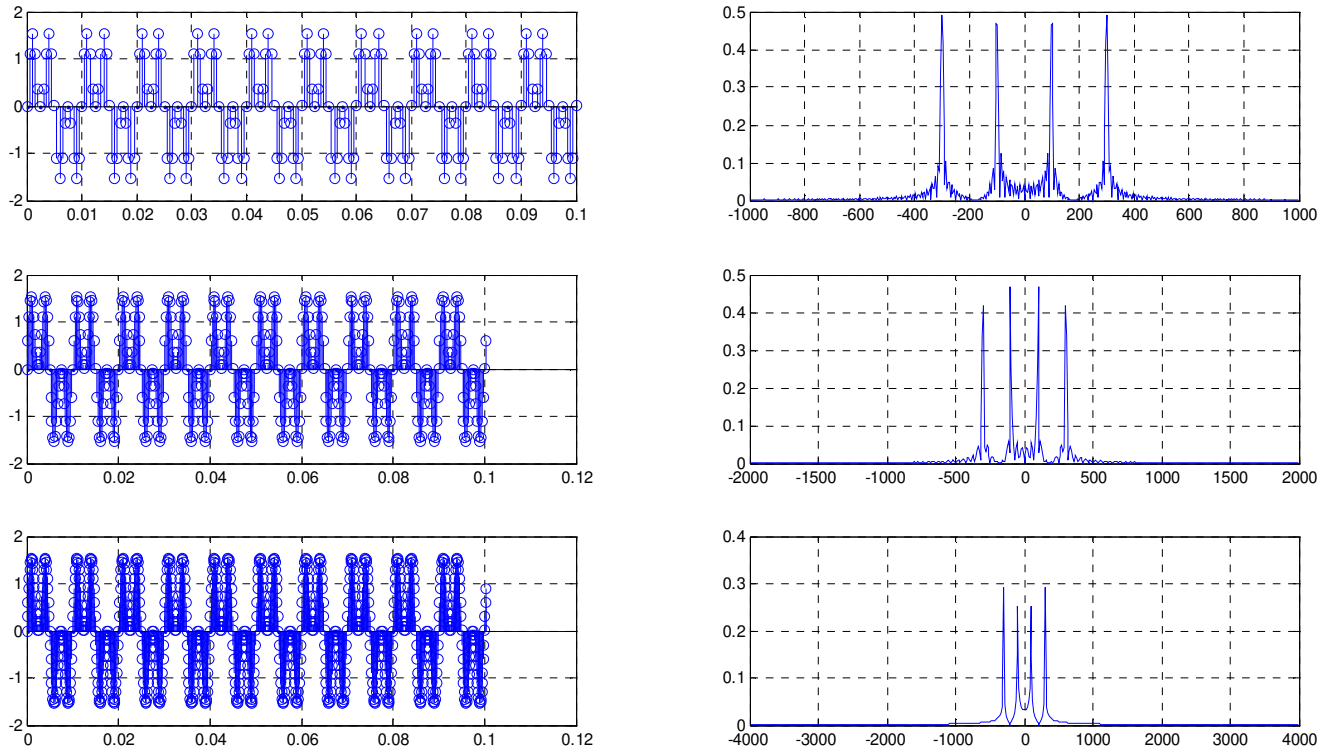
Ce qui en fréquence nous donne : $X_D(f) = X(z = e^{2\pi j f D T_e} = e^{2\pi j f D / f_e})$ ce qui signifie que le signal original et sur-échantillonné ont le même spectre, on ne fait que parcourir le cercle D fois.

Exemple : On reprend le même signal que l'on interpole de 2 puis de 4.

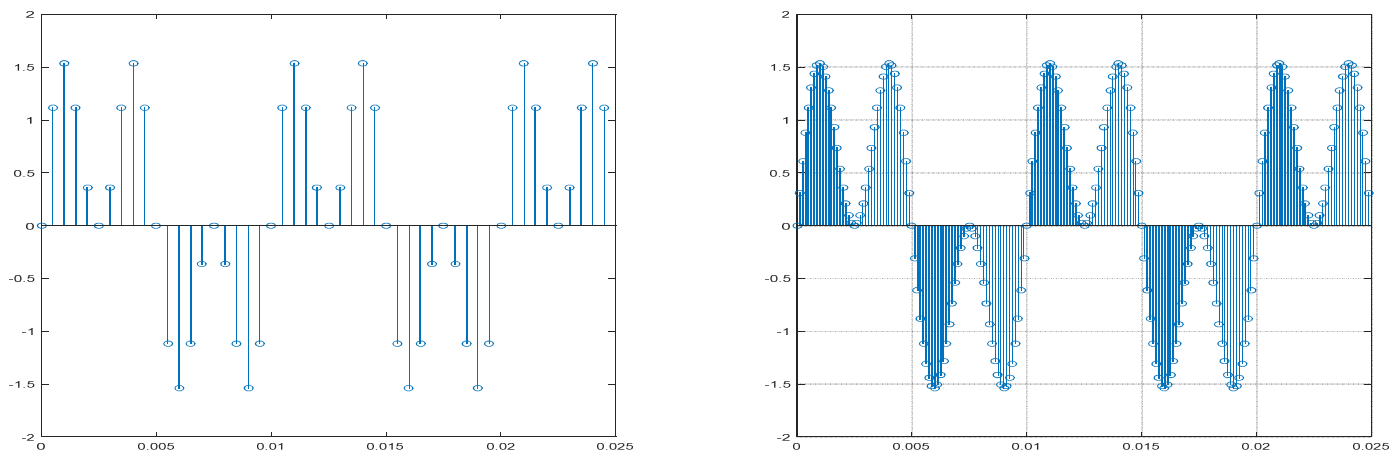


Après sur-échantillonnage, la forme du spectre reste la même aucune énergie n'a été ajoutée au signal), mais la fréquence d'échantillonnage est maintenant de Df_e . Le sur-échantillonnage fait apparaître ce que l'on appelle des spectres miroirs et pour obtenir un signal dont la forme temporelle correspondrait au signal $x(n)$ que l'on aurait échantillonné à une fréquence Df_e .

Il faut supprimer ces spectres miroirs et donc filtrer par un filtre passe-bas le signal sur-échantillonné $y(m)$ avec un filtre ayant une bande atténuée à partir de $f_e/2$



Nous pouvons observer l'effet l'interpolation suivie du filtrage passe-bas anti-miroirs (équivalent à un 0 padding dans le domaine fréquentiel)



Exemples:

- Interpolation de 4 $Y(z) = X(z^4)$

- Interpolation de 2 $Y(z) = X(z^2)$

Remarque :

La décomposition polyphases intervient dans les bancs de filtres, elle permet une représentation simple pour le sur et sous échantillonnage [21]. Quand un signal subit une décimation d'un facteur 2, cela revient à séparer les échantillons avec un indice pair de ceux avec un indice impair, ou encore, à séparer deux phases distinctes ayant des délais (aussi appelé écarts de phase) différents dans le vecteur [22]. Si on considère une décimation d'un facteur M, le principe est exactement le même mais avec M phases distinctes.

Changement de fréquence rationnel

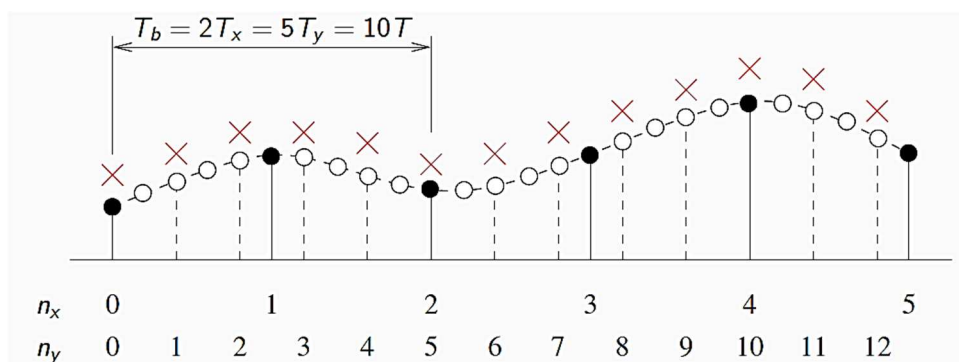
Supposons que l'on veuille changer numériquement la fréquence d'échantillonnage d'un signal $x(n)$ et passer d'une fréquence d'échantillonnage f_1 à une fréquence d'échantillonnage f_2 où $f_2/f_1 = L/M$. Cela conduit donc à sur-échantillonner le signal $x(n)$ d'un facteur L puis à le décimer d'un facteur M.

- Le sur-échantillonnage par L doit être suivie d'un filtre passe-bas coupant à $f_e/2L$ pour supprimer les spectres dus au sur-échantillonnage où f_e est la fréquence d'échantillonnage du signal de sortie.
- La décimation par M doit être précédée d'un filtrage anti-repliement ayant une fréquence de coupure de $f_e/2M$, si f_e est la fréquence d'échantillonnage du signal d'entrée.

On ne conservera alors qu'un seul des deux filtres passe-bas : celui dont la fréquence de coupure est la plus basse



Remarque : Si L et M sont premiers entre eux, décimation et interpolation sont commutatifs, ci-dessous un exemple : de conversion où $L = 5$ et $M = 2$ [31]



En outre, d'après la théorie des nombres, si L et M sont premiers entre eux alors ils existent deux entiers l_0 et m_0 que $l_0L - m_0M = -1 \rightarrow z^{-1} = z^{-(l_0L - m_0M)} = z^{-l_0L} z^{m_0M}$

Exemple : On souhaite sur une réduction de fréquence d'échantillonnage d'un signal audio d'un facteur 2/3 (passage de 48kHz à 32kHz). A noter que décimation et interpolation ne sont pas commutatives sauf si L et M sont premiers entre eux [20].

Remarques

Un filtre de décimation/interpolation à bande passante très étroite est complexe à mettre en œuvre (Ordre élevé)

. Il peut être remplacé par la mise en cascade de filtres plus simples. Les spécifications des filtres individuels sont considérablement assouplis puisque la spécification globale du filtre est partagée entre plusieurs filtres d'ordre inférieur.

Ainsi, pour un facteur de décimation ou interpolation, il est plus judicieux d'opérer par paliers (multi-étages).

Si on doit décimer de 15, cela nécessite un filtre d'ordre élevé. On peut alors procéder en décimant par 3 puis par 5. Ce qui nous permettra d'employer 2 filtres de plus faible ordre.

Les applications du traitement multi-cadences sont nombreuses : Conception de déphaseurs, interfaçage de systèmes numériques à taux différents, implémentation de filtres LP à bande étroite, implémentation de bancs de filtres numériques, codage sous-bande de signaux de parole, filtres miroirs en quadrature, transmultiplexeurs, sur-échantillonnage dans les CA/N et CN/A .

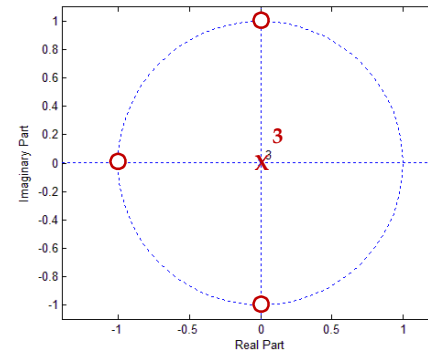
TD n°1 : Synthèses des Filtres et Filtage multicadences

1. Soit $H(z)$ la fonction de transfert d'un SLIT causal avec : $H(z) = \frac{az-1}{z-a}$ avec a réel.

Déterminer les valeurs de a pour lesquelles $H(z)$ correspond à un système stable. Prendre une valeur de $a=0.5$. Représenter alors les pôles et zéros de la fonction, la région de convergence. Donner et tracer $|H(f)|$. Ce filtre est-il à minimum de phase?

2. On suppose que le tracé des pôles et des zéros de ce système est le suivant :

- Est-ce un filtre RIF ou RII ? (Justifier votre réponse)
- Donner l'allure approximative de $H(f)$
- Déterminer $H(z)$ puis déterminer et tracer $h(n)$
- A partir de $h(n)$, étudier la stabilité, la causalité et l'invariance de ce filtre.
- Calculer et tracer $H(f)$ pour au moins 3 valeurs
- Ce filtre possède-t-il un retard de groupe constant (justifier)
- Déterminer sa réponse pour une entrée échelon $x(n)=U(n)$.



3. On considère que l'équation aux récurrences du système suivant est donnée comme suit :

$$y(n) = x(n) - x(n-2) - 0.878 y(n-2)$$

- Etudier la causalité et l'invariance de ce système
- Est-ce un filtre RIF ou RII ?
- Déterminer $H(z)$ et donner le tracé des pôles et zéros, en déduire le rôle de ce filtre :
- Donner les allures approximatives de $h(n)$ et $|H(f)|$
- Déterminer la réponse impulsionnelle $h(n)$ et tracer la pour les 3 premières valeurs
- On suppose que $f_e = 6$ kHz, quelle sera la sortie du filtre si l'on donne en entrée : un signal bruité par une sinusoïde de 500 Hz puis un signal composé de 2 sinusoïdes l'une de 1500 Hz et l'autre de 2500 Hz

4. Obtenir les coefficients d'un filtre à RIF passe-bas par la méthode de fenêtrage pour obtenir les spécifications suivantes :

- Fréquence de coupure idéale : $f_c = 1.75$ kHz
- Largeur de transition : $\Delta f = 0.5$ kHz
- Atténuation en bande atténuée : $A = -20 \log_{10}(\delta) > 51$ dB avec $\delta = \min(\delta_1, \delta_2)$
- Fréquence d'échantillonnage : $f_e = 8$ kHz

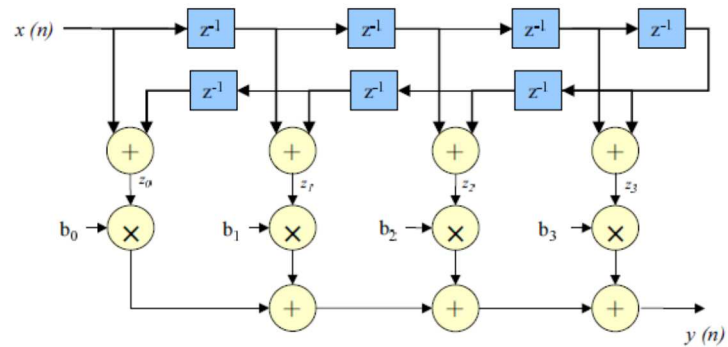
5. On souhaite approcher un filtre idéal passe-haut par un filtre à réponse impulsionnelle finie, synthétisé par la méthode du fenêtrage. Ce filtre doit répondre aux spécifications suivantes :

- ✓ Fréquence de coupure $f_c = 2$ kHz
- ✓ Largeur de transition : $\Delta f = 0.5$ kHz
- ✓ Atténuation en bande atténuée : $A = -20 \log_{10}(\delta) > 40$ dB avec $\delta = \min(\delta_1, \delta_2)$
- ✓ Fréquence d'échantillonnage : $f_e = 8$ kHz
- Déterminer l'expression mathématique exacte de $h'(n)$.
- Calculer $h'(0)$ et tracer approximativement $h'(n)$.
- Quel est l'intérêt du fenêtrage et quel est son inconvénient ?
- Quel est l'inconvénient de cette technique de synthèse des filtres?
- Citer un avantage et un inconvénient de la synthèse par des filtres RIF de même que par les RII.

6. On considère un filtre RIF de longueur $N=8$ dont la Structure est donnée ci-contre :

Avec $b_0=0.2, b_1=0.3, b_2=0.4, b_3=0.5$

- Déterminer l'équation aux différences
- Déterminer la fonction de transfert puis $h(n)$
- Identifier alors le type de filtre RIF (I,II,III ou IV)
- Quel est son inconvénient ?



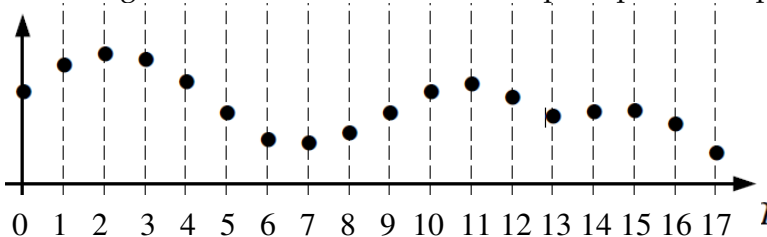
7. On considère le filtre passe-bas analogique normalisé à un pôle unique $H_N(p)=1/(p+1)$. On considère que la fréquence de coupure normalisée à -3db se produit pour $f_c=f_c/10$. Par transformation bilinéaire, trouver le filtre numérique passe-bas équivalent $H(z)$.

8. Soit la fonction de transfert passe bas dénormalisée $H(p)$ suivante : $(p+0.1)/((p+0.1)^2+\omega_a^2)$. Sachant que $\omega_a=4$, utiliser la méthode de la transformée bilinéaire pour transformer ce filtre en numérique. Le filtre numérique aura sa résonance pour $\omega_N = \pi/(2.T_e)$.

9. Soit le signal $x(n)=n U(n)$, donner et tracer le signal résultant si :

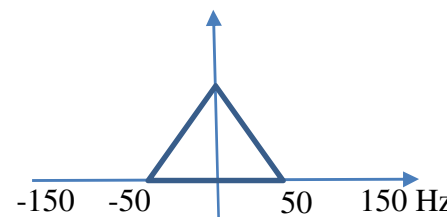
- On le décime par 2, par 3 et par 4
 - On l'interpole de 2, de 4
 - On le décime par 2 puis qu'on l'interpole par 2 et si on inverse les 2 opérations
 - On le décime par 3 et on l'interpole de 2 et si on inverse les 2 opérations
- Vérifier que la décimation n'est pas invariante à la translation contrairement à l'interpolation;

10. Soit le signal suivant, on veut le décimer par 2 puis l'interpoler par 2 également

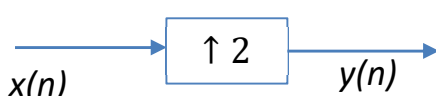
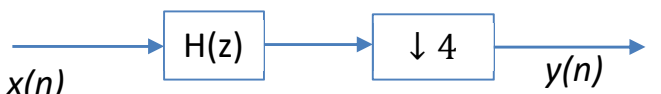
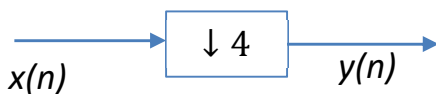
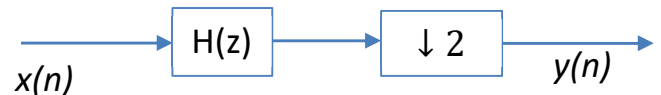
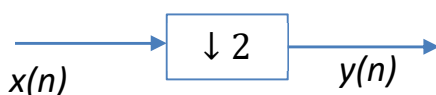


- Donner le tracé du signal décimé puis interpolé
- On considère que la fréquence du signal original est 10kHz, donner la fréquence à la sortie du décimateur puis celle de l'interpolateur.

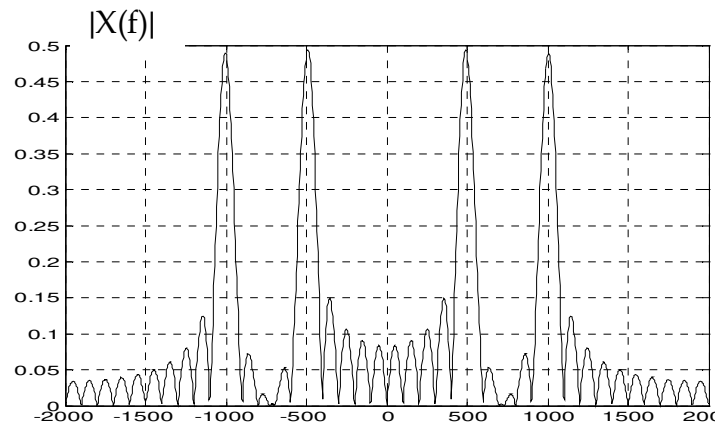
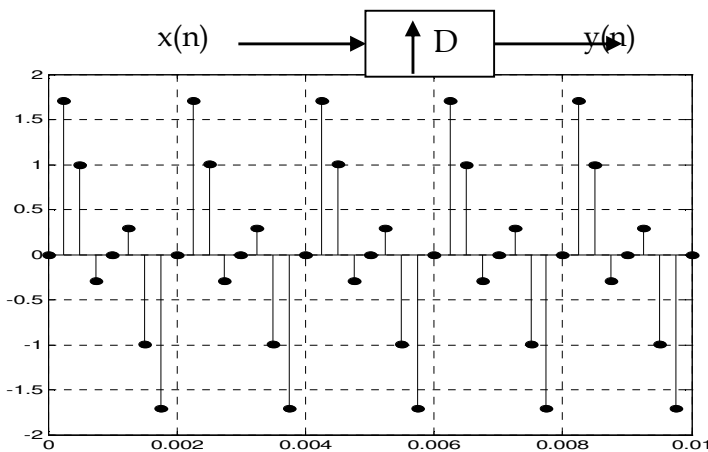
11. On considère le signal $x(n)$ dont la TFTD est donnée comme ci-contre ,



- Tracer la TFTD de la sortie correspondante pour les 8 cas :
- Donner la fréquence de coupure du filtre passe-bas $H(z)$ pour chaque cas
- Préciser le rôle de $H(z)$ dans chaque cas



12. Avant de convertir un signal numérique $x(n)$ (ci-dessous signal et sa TFTD), en signal analogique, on décide de l'interpoler par D.



- Quelle est l'intérêt de l'interpolation dans ce cas ?
- Déterminer l'expression du signal $x(n)$.
- Tracer le signal interpolé pour $D=3$ et sa TFTD.
- On considère que $D=6$, tracer à nouveau le signal interpolé et sa TFTD.
- Donner la décomposition polyphases initiale et finale pour $D=3$
- Tracer le signal et sa TFD dans le cas où l'on fait suivre l'interpolation d'un filtre passe-bas à $f_c/2$

13. Un signal audio est enregistré à une fréquence de 30 kHz, à la réception avant de le convertir en analogique, on opère un changement de fréquence telle que la nouvelle fréquence soit 45 kHz.

- Donner le schéma général de cette opération en employant la décimation et l'interpolation
- Donner la fréquence de coupure de chaque filtre passe-bas, lequel faudra-t-il garder

Solutions

1. Le module vaut toujours 1 (cellule passe-tout), Maximum de phase

2. Tous les pôles en 0 alors RIF, $H(z) = z^{-3} + z^{-2} + z^{-1} + 1$, $h(n) = \delta(n-3) + \delta(n-2) + \delta(n-1) + 1$,

$H(f) = 2(\cos(3\pi f T_e) + \cos(\pi f T_e))e^{-3\pi j f T_e}$, Retard de groupe cst, $x(n) = U(n)$ alors $y(n) = U(n-3) + U(n-2) + U(n-1) + U(n)$

3. Causal, stable, RII, notch filter, $H(z) = \frac{K(z^2 - 1)}{z^2 + 0.878} = \frac{K(1 - z^{-2})}{1 + 0.878z^{-2}}$

4. $\Delta f = 0.125$ $f_c = 0.4375$ $h(n) = f_c \frac{\sin(\pi n f_c)}{\pi n f_c} = \frac{\sin(\pi n 0.4375)}{\pi n}$ pour $n \neq 0$ et $h(0) = f_c$ Hamming $\Rightarrow N = 53$

Les coefficients du filtre sont symétriques, il suffira par conséquent de calculer les valeurs de $h(0)$ à $h(26)$.

$h(0) = f_c = 0.4375$ $w(0) = 0.54 + 0.46\cos(0) = 1$ $h(0) = h(N(0))$ $w(0) = 0.4375$

$$w_{Ham}(n) = \begin{cases} 0.54 + 0.46\cos\left(\frac{2\pi n}{N-1}\right) & \text{pour } |n| \leq \frac{N-1}{2} \\ 0 & \text{ailleurs} \end{cases} = \begin{cases} 0.54 + 0.46\cos\left(\frac{2\pi n}{52}\right) & \text{pour } |n| \leq 26 \\ 0 & \text{ailleurs} \end{cases}$$

$h(0) = 0.4375$, $h(1) = h(-1) = 0.311$, $h(2) = h(-2) = 0.060$, $h(3) = h(-3) = -0.0856$, $h(4) = h(-4) = -0.053$, $h(5) = h(-5) = 0.0325$, $h(6) = h(-6) = 0.0434$, $h(7) = h(-7) = -0.0075$, $h(8) = h(-8) = -0.0319$, ..., $h(26) = h(-26) = -0.0009$.

Pour rendre le filtre causal, on ajoute 26 à chacun des indices.

5. $\Delta f = 0.5$ $f_c = 0.125$ $h(n) = -f_c \frac{\sin(\pi n f_c)}{\pi f_c}$ pour $k \neq 0$ et $h(0) = 1 - f_c$ $A > 40$ Hanning $\Rightarrow N = 51$

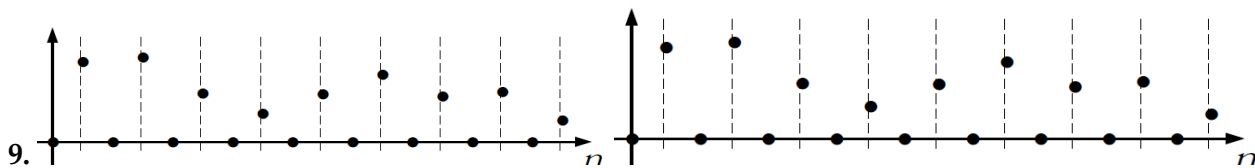
6. $y(n) = b_3(x(n-3) + x(n-4)) + b_2(x(n-2) + x(n-5)) + b_1(x(n-1) + x(n-6)) + b_0(x(n) + x(n-7))$

Type II, pas adéquat pour un passe-haut (un zéro en -1)

7. $\omega_A = \frac{2}{T_e} \operatorname{tg}\left(\frac{\pi}{10}\right) = \frac{0.65}{T_e} \Rightarrow H(p) = H_N(p) \Big|_{p/\omega_A} = \frac{\omega_A}{p + \omega_A} = \frac{0.65/T_e}{p + 0.65/T_e}$

$p = \frac{2}{T_e} \frac{1-z^{-1}}{1+z^{-1}} \Rightarrow H(z) = H(p) \Big|_{p=\frac{2}{T_e} \frac{1-z^{-1}}{1+z^{-1}}} = \frac{0.65/T_e}{\frac{2}{T_e} \frac{1-z^{-1}}{1+z^{-1}} + 0.65/T_e} = \frac{0.245(1+z^{-1})}{1-0.509z^{-1}}$

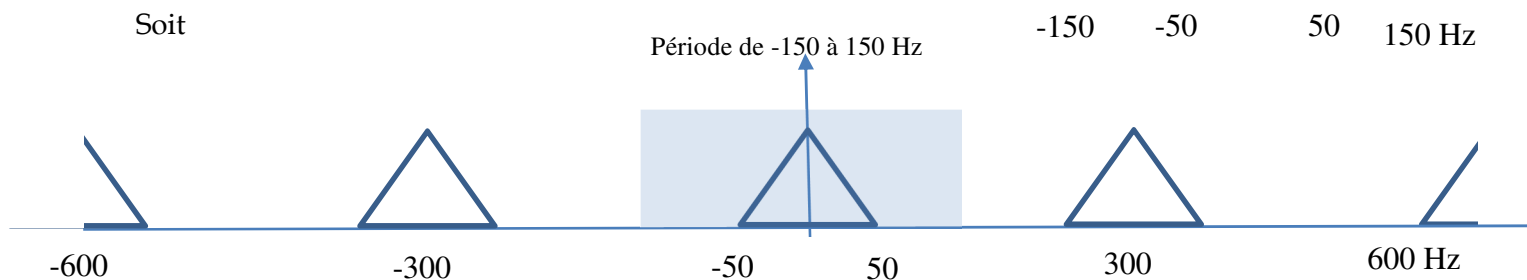
8. $\omega_a = 4 = \frac{2}{T_e} \operatorname{tg}\left(\frac{\pi}{4}\right) \Rightarrow T_e = 0.5 \Rightarrow p = 4 \frac{1-z^{-1}}{1+z^{-1}} \Rightarrow H(z) = \frac{4 \frac{1-z^{-1}}{1+z^{-1}} + 0.1}{\left(4 \frac{1-z^{-1}}{1+z^{-1}} + 0.1\right)^2 + 16} = \frac{0.125 + 0.006z^{-1} - 0.118z^{-2}}{1 + 0.0006z^{-1} - 0.950z^{-2}}$



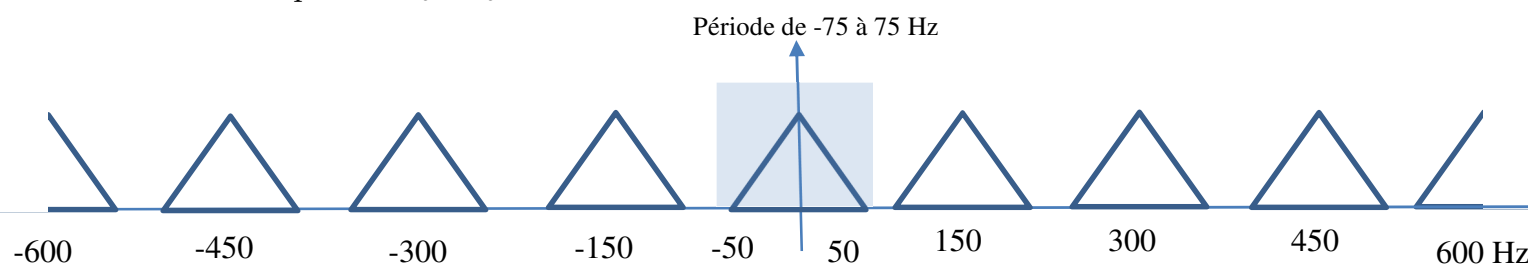
Après décimation $f_e = 5 \text{ kHz}$ Après interpolation $f_e = 10 \text{ kHz}$

10. Le signal original est périodique de période 300 Hz

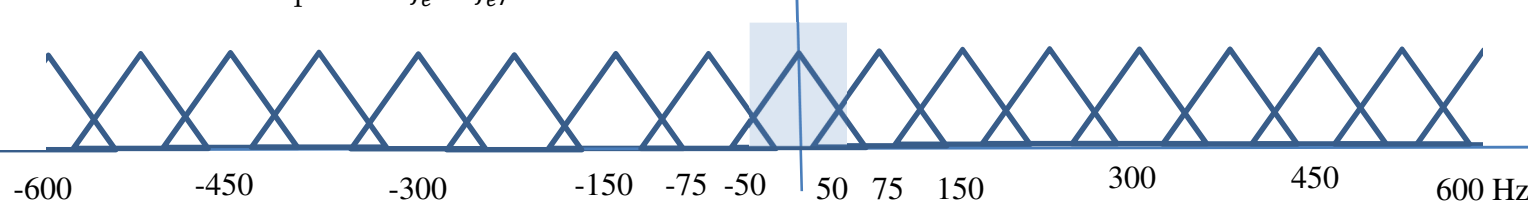
Soit



Décimation par 2 $\Rightarrow f'_e = f_e/2 = 150 \text{ Hz}$



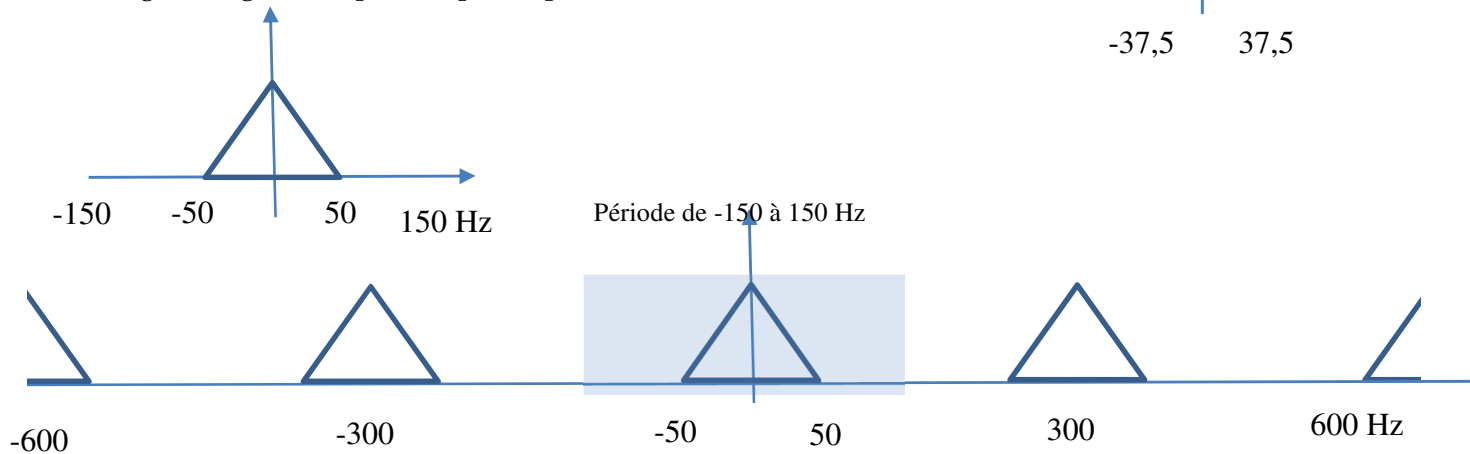
Décimation par 4 $\Rightarrow f'_e = f_e/4 = 75 \text{ Hz}$ Période de 37,5 à 37,5 Hz



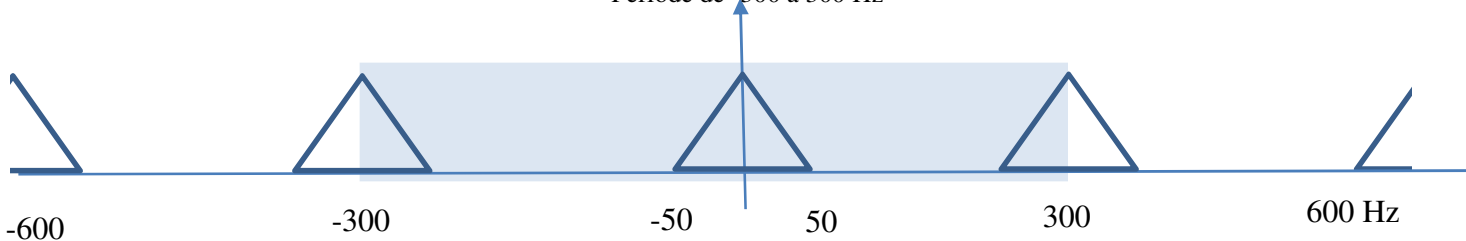
C'est pour cette raison qu'on place un filtre passe-bas à $f_e'/2 = f_e/(2M)$



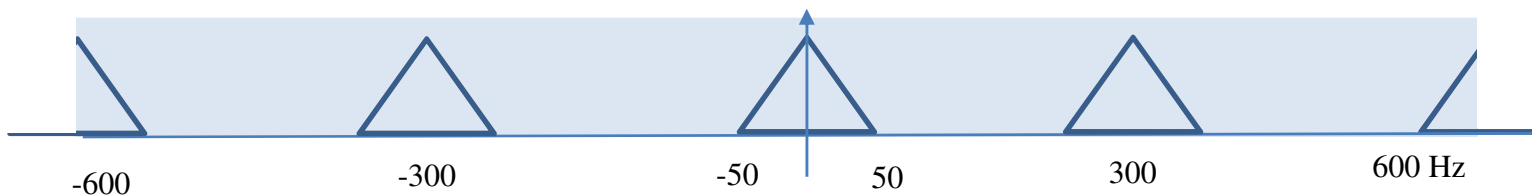
Le signal original est périodique de période 300Hz



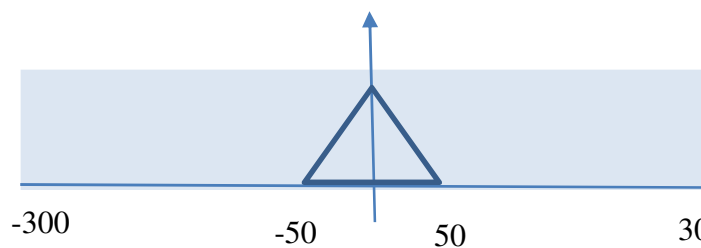
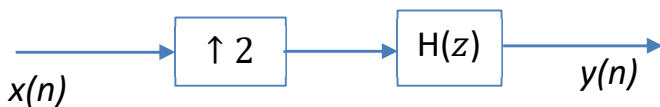
Interpolation par 2 $\Rightarrow f_e' = 2f_e = 600\text{Hz}$

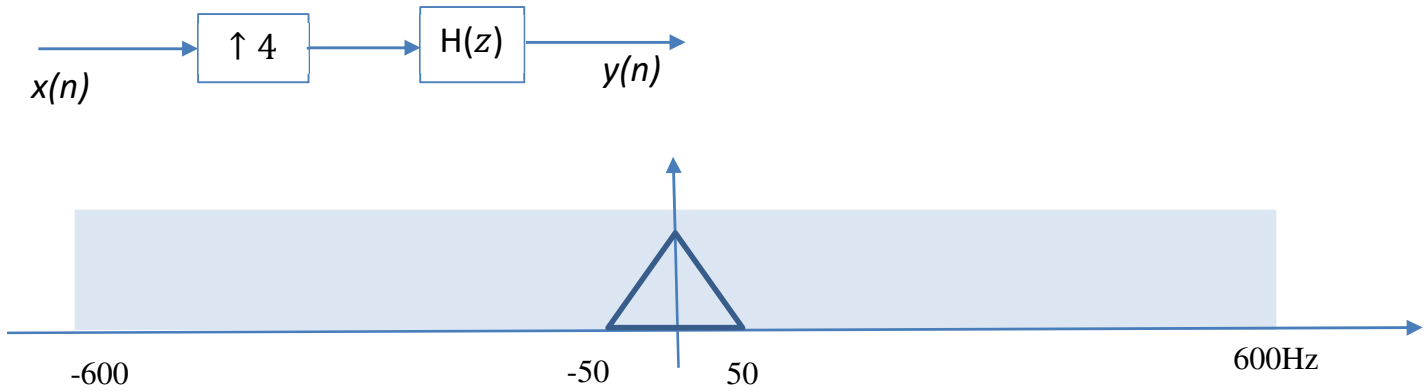


Interpolation par 4 $\Rightarrow f_e' = 4f_e = 1200\text{Hz}$



Apparition de spectres miroirs $\Rightarrow$ Filtre passe-bas à $f_e/2$





$$f_c = \frac{300}{2.2} = 75, f_c = \frac{300}{2.4} = 37,5$$

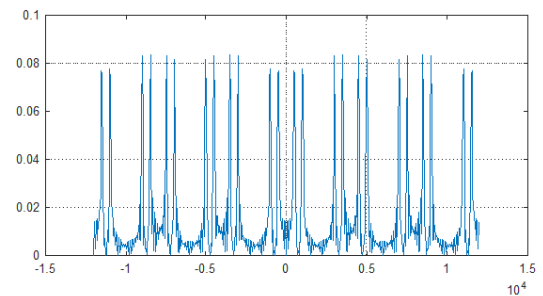
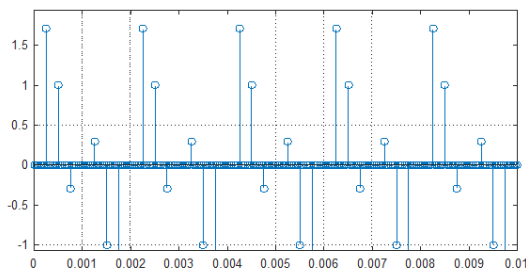
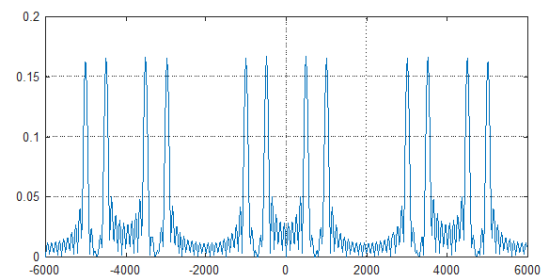
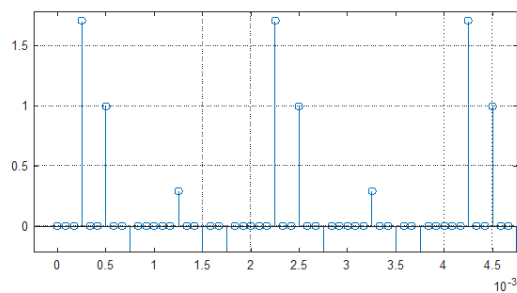
$$f_c = \frac{300}{2} = 150, f_c = \frac{300}{2} = 150$$

Décimation: Anti-repliement

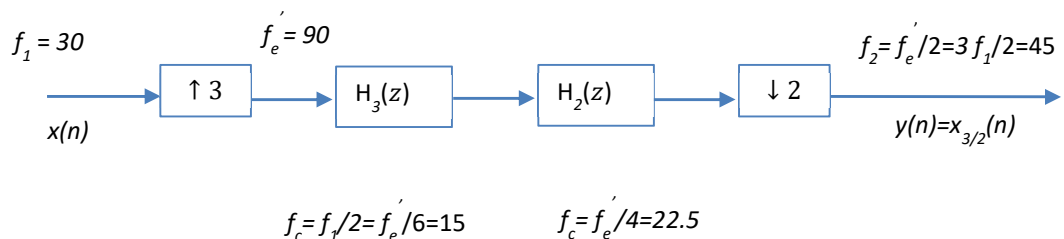
Interpolation: Anti-miroirs (Voir cours)

11. Gagner en précision avant la conversion numérique analogique

- $h(n) = \sin(2\pi \cdot 500 \cdot n) + \sin(2\pi \cdot 1000 \cdot n)$ $f_e = 4\text{kHz}$
- $D=3 \Rightarrow$ Insérer 2 zéros entre 2 échantillons consécutifs. TFTD sur 3 périodes de -6kHz à 6kHz
- $D=6 \Rightarrow$ Insérer 5 zéros entre 2 échantillons consécutifs. TFTD sur 6 périodes de -12kHz à 12kHz



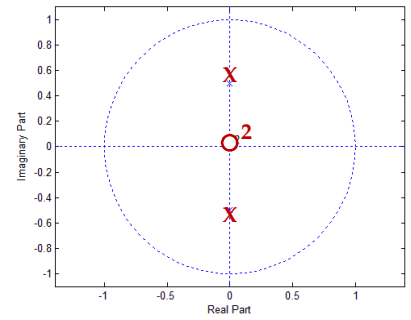
12.



Exercice supplémentaires

1. On suppose le tracé des pôles et des zéros du système suivant :

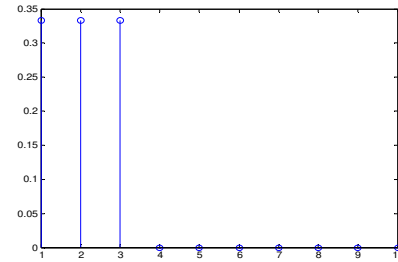
- Est-ce un filtre RIF ou RII ? (Justifier votre réponse)
- Donner l'allure approximative de $H(f)$
- Déterminer $H(z)$ puis déterminer l'équation de récurrence
- Déterminer et tracer $h(n)$
- A partir de $h(n)$, étudier la stabilité, la causalité et l'invariance du filtre



2. On considère le système suivant : $x(n)$ → $h(n)$ → $y(n)$

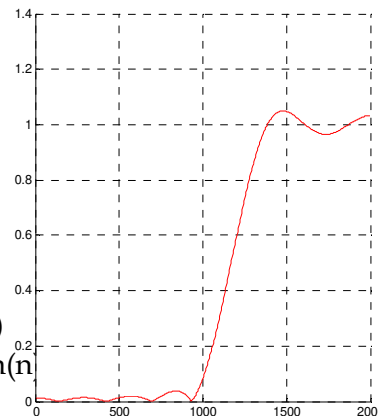
On suppose que $h(n)$ est donné comme ci-contre.

- Etudier la causalité. Est ce un filtre RIF ou RII
- A partir de l'expression de $h(n)$, déduire le rôle de ce filtre :
- Déterminer l'équation aux récurrences du système :
- Déterminer les pôles et zéros de ce filtre puis donner leur tracé. En déduire un tracé approximative de $|H(f)|$
- Calculer et tracer $|H(f)|$ puis en déduire le tracé du module de la TFD pour $N=6$.



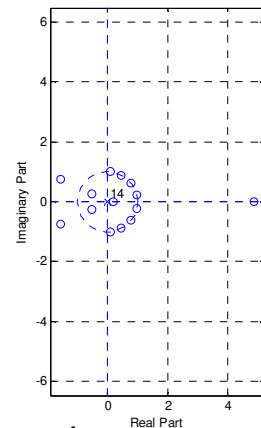
3. Au cours de la transmission d'un signal numérique (échantillonné à une fréquence de 2,5 kHz), il a été affecté d'un bruit localisé entre les bandes de fréquence 350 Hz et 550 Hz. On veut éliminer le bruit par l'emploi d'un filtre RIF possédant une bande de transition $\Delta f=100$ Hz. Concevoir ce filtre par la méthode du fenêtrage. On souhaite une atténuation en bande atténuée $A=-20 \log_{10}(\delta) > 20$ dB.

- Tracer $H(f)$ idéal
- Déterminer $h(n)$ et l'ordre N du filtre
- Calculer $h(0)$, $h(1)=h(-1)$
- Tracer alors approximativement $H(f)$



4. Les tracés du module de la réponse en fréquence de 0 à $f_e/2$ et des pôles et zéros d'un filtre numérique synthétisé par la méthode des fenêtres sont donnés comme suit :

- Le déphasage de ce filtre est-il linéaire ? (Justifier)
- Déterminer $H(k)$ puis la réponse impulsionnelle $h(n)$
- Calculer $h(0)$.
- On veut employer une fenêtre de hanning. Que devient le tracé du module de la réponse en fréquence (le superposer à $H(f)$)



5. On considère le filtre analogique d'ordre 3 associé à la fonction d'approximation de Butterworth. Réaliser le filtre numérique passe-bas correspondant en utilisant la transformation bilinéaire. La fréquence de coupure est $f_c=1$ kHz et la fréquence d'échantillonnage $f_e=10$ kHz.

6. On cherche à réaliser un filtre numérique équivalent au filtre analogique dénormalisé de transmittance

$$H_A(p) = \frac{1}{1+0.2p}$$

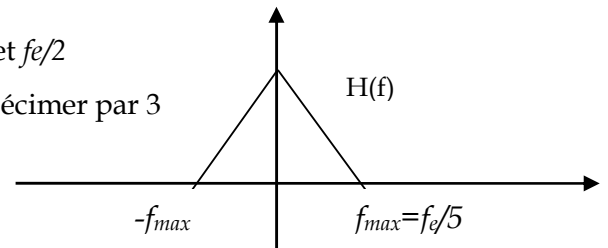
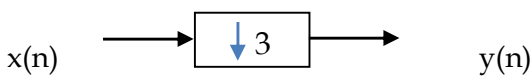
- Déterminer la réponse impulsionnelle $h_a(t)$
- Calculer la réponse en z de ce filtre obtenue par transformation bilinéaire pour une fréquence d'échantillonnage $T_e = 0.2$. Quelle est sa fréquence de coupure ? Calculer la réponse en z d'un filtre similaire de même fréquence de coupure que le filtre analogique.

7. Soit le signal suivant : $h(n) = \sin(2\pi \cdot 150 \cdot n) + \sin(2\pi \cdot 350 \cdot n)$ et le module de la TFD dont les tracés respectifs sont donnés ci-dessus : (voir exo 9)

- Déterminer T_e .
- Quel est l'intérêt de l'interpolation ?
- Tracer le signal (les 20 premières valeurs) et la TF obtenue pour une interpolation de 2 puis de 4 (sans filtrage passe-bas)
- Pourquoi emploie-t-on un filtre passe-bas lors l'interpolation, où le place-t-on ?
- Quel est le type du filtre de Tchebeychev employé dans ce cas et pourquoi ?
- Donner la décomposition polyphases finale pour une interpolation de 2 en donnant l'expression des filtres polyphases

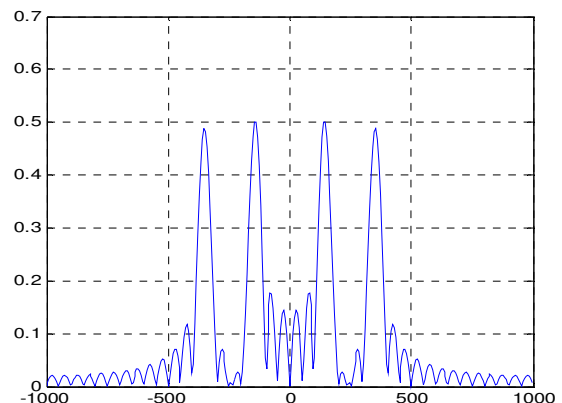
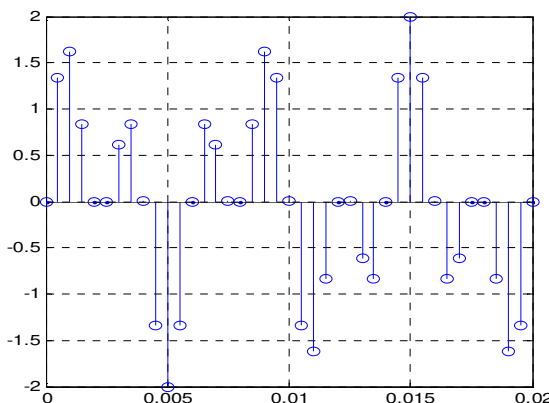
8. Afin de transmettre un signal $x(n)$ (dont la TTFD entre $-f_e/2$ et $f_e/2$

est donné ci-contre) plus rapidement, on décide de le décimer par 3



- Tracer la TTFD obtenue après décimation ($f_e = 30\text{kHz}$)
- On veut placer un filtre anti-repliement, ou doit-on le placer ? (justifier), tracer la TTFD à nouveau
- Donner la décomposition polyphases initiale et finale en donnant l'expression de chaque filtre.

9. Soit le signal suivant : $h(n) = \sin(2\pi \cdot 150 \cdot n) + \sin(2\pi \cdot 350 \cdot n)$ et le module de la TFD dont les tracés respectifs sont donnés ci-après :



- Tracer le signal et la TF obtenue pour une décimation de 2 puis de 4 (avec filtrage passe-bas)
- Donner la décomposition polyphases finale pour une décimation de 4 en donnant l'expression des filtres polyphases
- Quel est l'intérêt de la décomposition polyphases
- Pourquoi emploie-t-on un filtre passe-bas lors de la décimation

Solutions : Interros et examens années précédentes

TP n°1 : Synthèse des filtres RIF et RII

But du TP : Dans ce TP, on teste deux méthodes différentes pour synthétiser un filtre numérique : un filtre RIF par la méthode des fenêtres et un filtre RII par la méthode de la transformation bilinéaire.

1. Rappels

Un filtre de réponse impulsionnelle finie (RIF) possède une fonction de transfert polynomiale. Il ne peut pas être obtenu par transposition d'un filtre continu, comme cela est fait pour les filtres RII. Les filtres RIF présentent l'inconvénient de nécessiter un grand nombre de coefficients pour obtenir les mêmes caractéristiques fréquentielles. Mais par contre, ils sont inconditionnellement stables. On peut synthétiser des filtres RIF à phase linéaire, c'est-à-dire à temps de propagation de groupe constant.

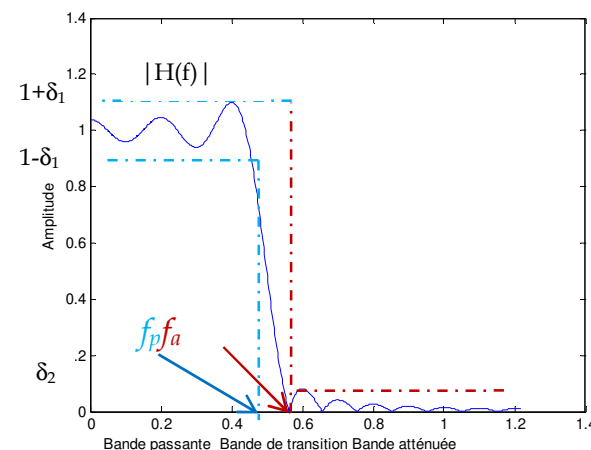
L'intérêt des filtres récursifs (RII) est leur faible coût en calcul. Avec très peu de pôles et zéros on peut assurer la plupart des réponses fréquentielles dont on peut avoir besoin dans les différentes applications. Les inconvénients des filtres récursifs sont : leur non-linéarité en phase linéaire (phase linéaire : temps de propagation constant pour tout fréquence), et leur instabilité numérique. En effet, le filtre étant rétroactif, les erreurs de précision numérique deviennent une question d'importance, car elles peuvent s'amplifier et devenir dehors contrôle, d'abord dans la forme de bruit, mais éventuellement dans la forme d'instabilité. A noter que les filtres RII peuvent être conçus par des méthodes semblables à ceux utilisés pour les filtres analogiques.

Gabarit d'un filtre

Le gabarit d'un filtre n'est autre que l'ensemble des caractéristiques du filtre, à savoir :

- Le gain du filtre dans la bande passante.
- L'atténuation du filtre en bande coupée f_a .
- La fréquence de coupure f_c , on l'exprime souvent sous forme normalisée par rapport à la fréquence d'échantillonnage.
- La largeur de bande de transition Δf souhaitée qui doit être la plus petite possible.
- Les éventuelles oscillations en bande passante et/ou atténuée.

- La détermination des coefficients d'un filtre RIF par la méthode de la fenêtre est réalisée par la fonction Python `firwin` de la bibliothèque `scipy.signal`.



2. Démonstration

-*- coding: utf-8 -*-

```
import numpy as np; import scipy.signal as sp; import matplotlib.pyplot as plt
from plot_zplane import zplane
#Conception d'un filtre RIF passe-bas fc=200, fe =1000, Deltaf = 100, Aa>20
plt.figure(1)
fc=200;fe=1000; Deltaf=100
fcn=2*fc/fe ; Deltafn=Deltaf/(fe/2)
Ncoef = int(np.ceil(1.8/Deltafn))
n = np.arange(-(Ncoef//2),Ncoef//2+1)
b = fcn*np.sinc(n*fcn);
#Visualisation de la réponse impulsionnelle et de la réponse fréquentielle
L=256
f,Hf= sp.freqz(b,1,L,fs=fe);
plt.subplot(211); plt.stem(b);
plt.subplot(212); plt.plot(f, np.abs(Hf))
plt.title('Module du Filtre'); plt.xlabel('Fréquence (Hz)'); plt.ylabel('Amplitude')
```

FGE, USTHB [assiakourgli@gmail.com / <http://perso.usthb.dz/~akourgli/>]

plt.legend()

Détermination et tracé des pôles et zéros

plt.figure(2)

a=np.array([1])

z,p = zplane(b,a)

Effet de l'augmentation du nombre de coefficients

fc=0.2;fe=1; fcn=2*fc/fe

A=np.array([21, 61, 101]);i=1;

a = np.array([1]);

L = 256;

plt.figure(3)

for N in A:

 n = np.arange(-(N//2),N//2+1)

 b = fcn*np.sinc(n*fcn);

 f,Hf= sp.freqz(b,a,L,fs=fe);

 plt.subplot(2,3,i); plt.stem(b);

 plt.subplot(212); plt.plot(f, np.abs(Hf),label='N=%d'%N)

 plt.title('Module du Filtre'); plt.grid(True); plt.xlabel('Fréquence (Hz)');plt.ylabel('Amplitude')

 plt.legend()

 i+=1

#Effet du fenêtrage

from scipy.signal import windows

plt.figure(4)

N=41;

A=np.array([np.ones(N), windows.hann(N),windows.hamming(N)]);

i=1;

for Fen in A:

 n1 = np.arange(-(N//2),N//2+1)

 b1 = fcn*np.sinc(n1*fcn)*Fen;

 f,Hf1= sp.freqz(b1,a,L,fs=fe);

 plt.subplot(2,3,i); plt.stem(b1); plt.title('Filtre * Fen %d'%i);

 plt.subplot(212); plt.plot(f, np.abs(Hf1),label='Fen %d'%i)

 plt.title('Module du Filtre'); plt.grid(True); plt.xlabel('Fréquence (Hz)');plt.ylabel('Amplitude')

 plt.legend()

 i+=1

#Effet du fenêtrage (affichage en db)

plt.figure(5)

fc=0.2;fe=2; fcn=2*fc/fe; N=41;

a = np.array([1]);

L = 256;

A=np.array([np.ones(N), windows.hann(N),windows.hamming(N)]);i=1;

for Fen in A:

 n2 = np.arange(-(N//2),N//2+1)

 b2= fcn*np.sinc(n2*fcn)*Fen;

 f,Hf1= sp.freqz(b2,a,L,fs=fe);

 plt.subplot(2,3,i); plt.stem(b2); plt.title('Filtre * Fen %d'%i);

 plt.subplot(212); plt.plot(f, 20*np.log10(np.abs(Hf1)),label='Fen %d'%i)

 plt.title('Module du Filtre'); plt.grid(True); plt.xlabel('Fréquence (Hz)');plt.ylabel('Amplitude')

 plt.legend()

 i+=1

Transformation bilinéaire

fp=195;fa= 205; fe=1000;

```
att_p=1; att_a=40;
wpm = fp*2*np.pi;
wan = fa*2*np.pi;
wpa = 2*fe*np.tan(np.pi*fp/fe) #Compensation de pulsation

N, WA = sp.buttord(wpm,wan, att_p,att_a, fs=fe)
z,p,k = sp.buttap(N); """Filtre analogique hn(p) sous forme de poles, zéros et gain"""
# # sp.cheb2ap(N,att_a); sp.buttap(N); sp.ellipap(N,att_p,att_a)
Bpn,Apn = sp.zpk2tf(z,p,k); """Filtre analogique hn(p) sous forme num, den """
Bp, Ap = sp.lp2lp (Bpn,Apn,wpa); """Denormalisation """
# # sp.lp2hp sp.lp2bp sp.lp2bs
poles_anal = np.roots(Ap);
Bz1, Az1 = sp.bilinear(Bp,Ap,fe); """Passage de H(p) vers H(z) par Transformation Bilinéaire"""

# Comparaison filtres analogique Ha(f) et numérique Hz(f)
L = 256;
fz,Hf = sp.freqz(Bz1,Az1,L,fs=fe);
wa,Ha = sp.freqs(Bp,Ap, worN=np.arange(1,np.pi*fe)); fa=wa*0.5/np.pi;
plt.figure(6);
plt.plot(fz, np.abs(Hf),label='H(f) du filtre Numérique');
plt.plot(fa, np.abs(Ha), label='H(f) du filtre Analogique'); plt.grid(); plt.legend()

plt.figure(7);
z,p=zplane(Bz1,Az1);
```

3. A faire en TP

```
# import numpy as np; import scipy.signal as sp; import matplotlib.pyplot as plt
# from plot_zplane import zplane
# import sounddevice as snd
# import scipy.io.wavfile as wav
# fe,x = wav.read('vousavezducourrierenattente.wav')
# snd.play(x, fe)

# Te=1/fe; N=len(x); t = np.arange(0,N)*Te;
# plt.figure(1);plt.subplot(211); plt.plot(t,x); plt.grid(True);
# plt.xlabel('temps'); plt.ylabel('Amp'); plt.title('Phrase');

# # t = np.arange(0,N)*Te;
# # f0 = 1500; b = 20*np.cos(2*np.pi*f0*t);
# # Xb = x + b ;
# # snd.play(np.int8(Xb), fe)

"""
-----A faire-----
```

Calculer et Visualiser sa TFD en db dans la figure (1)

Rajouter du bruit (3 sinusoides avec une fréquence de 1500, 2000 et 3000)

Ecouter le signal bruité

Visualiser la TFD du signal bruité en db

Concevoir un filtre par la méthode des fenêtres avec $A_a > 50$ et $\Delta f < f_c/10$, f_c à déterminer en utilisant firwin

Visualiser la réponse impulsionnelle et la fonction de Transfert du filtre

Filtrer le signal bruité X_b ($y = \text{sp.lfilter}(b,a,x)$) avec le filtre crée (b,a)

Ecouter et visualiser la TFD du signal filtré et comparer avec l'originale'

Concevoir Filtre par la méthode de la transformation bilinéaire avec $A_a > 50$ $A_p = 2$ avec N à déterminer, il faut essayer plusieurs filtres analogiques (Butterworth, Cheby 1, Cheby 2), lequel vous semble adéquat?

Visualiser la réponse impulsionnelle et la fonction de Transfert du filtre pour chaque méthode

Filtre le signal bruité X_b avec le filtre créé

Ecouter et visualiser la TFD du signal filtré

Commenter

Comparer entre les 2 méthodes des synthèse des filtres. Pour cet exemple, laquelle vous semble préférable?

''''''

TP n°2 : Filtrage multiscadences

1. Interpolation et Décimation : Démo

```
# -*- coding: utf-8 -*-
import numpy as np; import matplotlib.pyplot as plt; import scipy.signal as sp;

N = 30; f0=100; f1=300; fe=2000; Te=1/fe; NF=1024;
t = np.arange(0.0, N*Te, Te); x = np.sin(2.0*np.pi*f0*t) + np.sin(2.0*np.pi*f1*t) ;
TFx = np.fft.fft(x,NF); TFx = np.fft.fftshift(TFx); freq = np.arange(-NF/2,NF/2)*fe/NF;
plt.figure(1)
plt.subplot(321); plt.stem(t, x); plt.subplot(322); plt.plot(freq, np.abs(TFx));
x1 = np.zeros(2*N); x1[::2]=x[::]; t1 = np.arange(0.0, N*Te, Te/2); TFx1 = np.fft.fft(x1,NF);
TFx1 = np.fft.fftshift(TFx1); freq = np.arange(-NF/2,NF/2)*2*fe/NF;
plt.subplot(323); plt.stem(t1, x1); plt.subplot(324); plt.plot(freq, np.abs(TFx1));
x2 = np.zeros(4*N); x2[::4]=x[::]; t2 = np.arange(0.0, N*Te, Te/4); TFx2 = np.fft.fft(x2,NF);
TFx2 = np.fft.fftshift(TFx2); freq = np.arange(-NF/2,NF/2)*4*fe/NF;
plt.subplot(325); plt.stem(t2, x2); plt.subplot(326); plt.plot(freq, np.abs(TFx2));

plt.figure(2)
freq = np.arange(-NF/2,NF/2)*fe/NF;
plt.subplot(321); plt.stem(t, x); plt.subplot(322); plt.plot(freq, np.abs(TFx));
x1 = sp.resample(x,2*N); t1 = np.arange(0.0, N*Te, Te/2); TFx1 = np.fft.fft(x1,NF);
TFx1 = np.fft.fftshift(TFx1); freq = np.arange(-NF/2,NF/2)*2*fe/NF;
plt.subplot(323); plt.stem(t1, x1); plt.subplot(324); plt.plot(freq, np.abs(TFx1));
x2 = sp.resample(x,4*N); t2 = np.arange(0.0, N*Te, Te/4); TFx2 = np.fft.fft(x2,NF);
TFx2 = np.fft.fftshift(TFx2); freq = np.arange(-NF/2,NF/2)*4*fe/NF;
plt.subplot(325); plt.stem(t2, x2); plt.subplot(326); plt.plot(freq, np.abs(TFx2));

N = 100; f0=100; f1=300; fe=2000; Te=1/fe; NF=1024;
t = np.arange(0.0, N*Te, Te); x = np.sin(2.0*np.pi*f0*t) + np.sin(2.0*np.pi*f1*t) ;
TFx = np.fft.fft(x,NF); TFx = np.fft.fftshift(TFx); freq = np.arange(-NF/2,NF/2)*fe/NF;
plt.figure(3)
plt.subplot(321); plt.stem(t, x); plt.subplot(322); plt.plot(freq, np.abs(TFx));
x1 = x[::2]; t1 = t[::2]; TFx1 = np.fft.fft(x1,NF); TFx1 = np.fft.fftshift(TFx1);
freq = np.arange(-NF/2,NF/2)*fe/(2*Nf);
plt.subplot(323); plt.stem(t1, x1); plt.subplot(324); plt.plot(freq, np.abs(TFx1));
x2 = x[::6]; t2 = t[::6]; TFx2 = np.fft.fft(x2,NF); TFx2 = np.fft.fftshift(TFx2);
freq = np.arange(-NF/2,NF/2)*fe/(6*Nf);
plt.subplot(325); plt.stem(t2, x2); plt.subplot(326); plt.plot(freq, np.abs(TFx2));

plt.figure(4)
freq = np.arange(-NF/2,NF/2)*fe/NF;
plt.subplot(321); plt.stem(t, x); plt.subplot(322); plt.plot(freq, np.abs(TFx));
x1 = sp.decimate(x,2); t1 = t[::2]; TFx1 = np.fft.fft(x1,NF); TFx1 = np.fft.fftshift(TFx1);
freq = np.arange(-NF/2,NF/2)*fe/(2*Nf);
plt.subplot(323); plt.stem(t1, x1); plt.subplot(324); plt.plot(freq, np.abs(TFx1));
x2 = sp.decimate(x,6); t2 = t[::6]; TFx2 = np.fft.fft(x2,NF); TFx2 = np.fft.fftshift(TFx2);
freq = np.arange(-NF/2,NF/2)*fe/(6*Nf);
plt.subplot(325); plt.stem(t2, x2); plt.subplot(326); plt.plot(freq, np.abs(TFx2));
```

2. Interpolation et Décimation : A faire

```
# -*- coding: utf-8 -*-

import numpy as np; import matplotlib.pyplot as plt
#from zp_plot import zplane
import scipy.signal as sp;

from scipy.io import wavfile as wf; import winsound ;
fname = 'vousavezducourrierenattente.wav';
winsound.PlaySound(fname, winsound.SND_FILENAME)
fe, x = wf.read(fname);
Te=1/fe; N=len(x); t = np.arange(0,N)*Te;
```

```
# plt.figure(1);plt.subplot(211); plt.plot(t,x); plt.grid(True);
# plt.xlabel('temps'); plt.ylabel('Amp'); plt.title('Phrase');
# #Ecrire dans un fichier audio
# fnameb='Copie du signal.wav';
# Xb=X
# Xb=np.int8(Xb) ; wf.write(fnameb, fe, Xb);
# winsound.PlaySound(fnameb,winsound.SND_FILENAME) ;

#-----A faire-----
# 1
# Tester le sous échantillonnage avec et sans filtrage passe-bas
# Afficher le signal et sa TFD dans chaque cas et zoomer sur une meme zone d'une durée de 50 ms et
commenter
# Ecouter le signa obtenu à chaque fois
# Perçoit-on une déformation du son ? A quoi est-elle due ?
# Jusqu'à quel facteur de décimation peut-on aller sans déformation
# Que se passet-il si on garde la même fe (originale) à l'écoute?

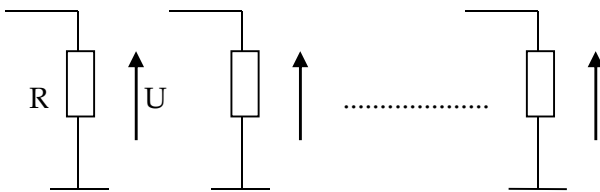
# 2
# Tester le sur échantillonnage avec et sans filtrage passe-bas
# Afficher le signal et sa TFD dans chaque cas et zoomer sur une meme zone d'une durée de 50 ms ou
20ms et commenter
# Ecouter le signa obtenu à chaque fois (wf.wite avec la nouvelle fe en multipliant par facteur d'Interp)
# Perçoit-on une déformation du son ? pourquoi?
# Que se passet-il si vous garde la même fe à l'écoute?

# 3
# On veut passer à un signal dont la fréquence d'échantillonnage est  $1.5 \cdot f_e$ 
# Proposer 2 solutions
# Laquelle vous permet de conserver le plus les valeurs originelles ?
```

II. Processus aléatoires

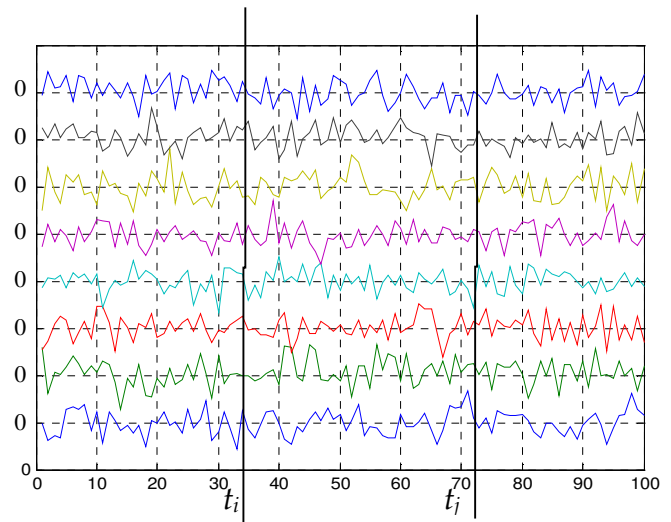
Les processus aléatoires (stochastiques) décrivent l'évolution d'une grandeur aléatoire en fonction du temps (ou de l'espace). On peut définir un processus stochastique comme étant une famille de variables aléatoires indexées par le temps définies sur un même espace de probabilités Ω . Un processus aléatoire ne possède pas de représentations temporelles analytiques. Chaque signal aléatoire observé représente une réalisation particulière de ce processus. Le signal de parole, le signal radar, l'électrocardiogramme, l'électro-encéphalogramme, les signaux sismiques en sont des exemples. Rappelons qu'un signal de réception - constitué d'un signal informatif (aléatoire ou déterministe), d'un signal aléatoire d'interférence et de bruit lié au canal de transmission- est aléatoire par nature.

Exemple 1 : Si l'on prend plusieurs résistances identiques (de même valeur) et que l'on mesure la tension. On trouvera une valeur non nulle due à l'agitation thermique des électrons libres dans la résistance. Les tensions mesurées au cours du temps sur l'ensemble des résistances fourniront plusieurs signaux aléatoires tous différents qui constituent le processus aléatoires.



Signaux susceptibles d'être produits en mesurant la tension aux bornes de plusieurs résistances.

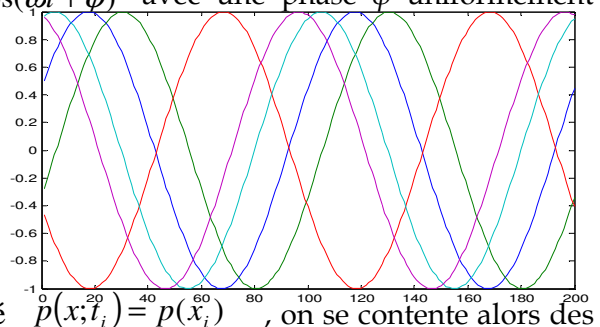
- Chaque tracé fournit un signal aléatoire
- à l'instant t_i , le processus se réduit à une v.a x_i dont la densité est $p(x; t_i) = p(x_i)$



- Deux instants t_i et t_j permettent de définir deux variables aléatoires x_i et x_j on peut définir des densités de probabilités conjointes par: $p(x; t_i, t_j) = p(x_i, x_j)$

Exemple 2 : Signal sinusoïdal à phase aléatoire $X(t, \varphi) = a \cos(\omega t + \varphi)$ avec une phase φ uniformément répartie entre 0 et 2π

- φ est à valeur réelle continue
- $X(t, \varphi)$ est à valeur continue et réelle
- Un signal particulier $X(t, \varphi_i)$ est déterministe



En pratique, il n'est pas aisé d'obtenir la densité de probabilité $p(x; t_i) = p(x_i)$, on se contente alors des moments d'ordre 1 et 2

1. Espérances mathématiques

Statistiques de 1^{er} ordre: Elles permettent de décrire le signal à un instant donné. Le processus aléatoire devient une simple variable aléatoire que l'on peut décrire à l'aide de moments à condition que sa probabilité soit connue $\forall t_i$.

- moyenne statistique: $\mu_x(t_i) = E\{x(t_i)\} = E\{x_i\} = \int_{-\infty}^{+\infty} x(t_i) \cdot p(x, t_i) dx$

- variance: $\sigma_x^2(t_i) = E\{x(t_i) - \mu_x(t_i)\}^2 = E\{x_i^2\} - \mu_x(t_i)^2$

Où $E\{x_i^2\} = \int_{-\infty}^{+\infty} x(t_i)^2 \cdot p(x, t_i) dx$ est la valeur quadratique moyenne (on parle aussi de puissance) et $\sigma_{st}(t_i)$ dit écart-type

Exemple 1: Soit le processus stochastique $x(t)$ défini par $x(t) = a + b \cdot t$ où a et b sont des v.a. dont les probabilités sont connues, alors :

$$\mu_x(t) = E[a + b t_i] = \mu_a + \mu_b t$$

Reprenons l'exemple 2 : Pour un instant donné t_k , on peut calculer des moments statistiques de la variable aléatoire $X(t_k, u)$

- Moyenne :

$$\mu_x(t) = E[X(t)] = \int_0^{2\pi} f_\varphi(\varphi) a \cos(\omega t + \varphi) d\varphi = \frac{1}{2\pi} a \int_0^{2\pi} \cos(\omega t + \varphi) d\varphi = 0$$

-Variance :

$$\sigma_x^2(t) = E[X^2(t)] - \mu_x^2(t) = E[X^2(t)] = \int_0^{2\pi} f_\varphi(\varphi) a^2 \cos^2(\omega t + \varphi) d\varphi = \frac{a^2}{2}$$

Supposons, maintenant, que φ est déterministe et que a est aléatoire de densité $p(a)$, de moyenne μ_a et variance σ_a^2 , alors :

$$\mu_x(t) = E[a \cos(\omega t + \varphi)] = \mu_a \cdot \cos(\omega t + \varphi) \quad \sigma_x^2(t) = \sigma_a^2 \cdot \cos^2(\omega t + \varphi)$$

Statistiques de 2^{ème} ordre:

On dit que le processus est connu à 2 instants t_1 et t_2 , si $\forall t_1, t_2$, la probabilité conjointe est connue. Dans la mesure où l'espérance fait intervenir le produit de deux variables aléatoires (2 instants), on parlera de statistiques d'ordre 2. on peut estimer les statistiques suivantes:

- Fonction d'autocorrélation statistique :

$$R_x(t_1, t_2) = E\{x(t_1) \cdot x^*(t_2)\} = E\{x_1 \cdot x_2^*\} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x_1 \cdot x_2^* \cdot p(x_1, x_2) \cdot dx_1 \cdot dx_2$$

- Fonction d'autocovariance statistique :

$$C_x(t_1, t_2) = E\{[x(t_1) - \mu_x(t_1)] \cdot [x(t_2) - \mu_x(t_2)]^*\} = R_x(t_1, t_2) - \mu_x(t_1) \cdot \mu_x(t_2)^*$$

Elles illustrent la relation entre les statistiques prises à **deux** instants t_1 et t_2 différents. La fonction d'autocorrélation mesure la corrélation entre les signaux émis par un même processus à deux instants différents. Lorsqu'il y a corrélation on peut parler d'un "effet mémoire" du processus. On définit la mémoire du processus comme le temps t_c au delà duquel la corrélation est négligeable, t_c est aussi appelé temps de corrélation. Par contre, si elle est nulle, le signal est complètement aléatoire et le signal $x(t)$ à l'instant t est complètement décorrélé de ce même signal à l'instant passé $t - \tau$.

Notons que pour des processus centrés

$$(\mu_x(t)=0): C_x(t_1, t_2) = R_x(t_1, t_2)$$

On peut étendre ces notions pour mesurer ce lien entre deux processus aléatoires par :

- Fonction d'intercorrélation statistique: $R_{xy}(t_1, t_2) = E\{x(t_1) \cdot y^*(t_2)\} = E\{x_1 \cdot y_2^*\}$
- Fonction d'intercovariance statistique: $C_{xy}(t_1, t_2) = R_{xy}(t_1, t_2) - \mu_x(t_1) \cdot \mu_y^*(t_2)$
- Le coefficient de corrélation est définie par: $\rho_{xy}(t_1, t_2) = \frac{C_{xy}(t_1, t_2)}{\sigma_x(t_1)\sigma_y(t_2)} \quad -1 \leq \rho_{xy}(t_1, t_2) \leq 1$

Reprenons l'exemple 2 et calculons l'auto-corrélation statistique :

$$R_{xx}(t_1, t_2) = E[a \cos(\omega t_1 + \varphi) a \cos(\omega t_2 + \varphi)] = E\left[\frac{a^2}{2} (\cos(\omega(t_1 + t_2) + 2\varphi) + \cos(\omega(t_1 - t_2)))\right] = \frac{a^2}{2} \cos(\omega(t_1 - t_2))$$

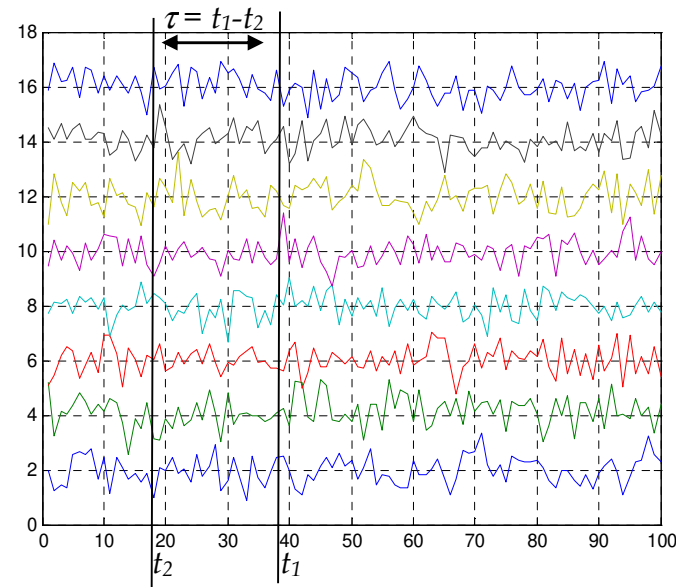
On remarque que c'est une fonction périodique, ne dépendant que de l'écart $t_1 - t_2$. Pour $t_1 = t_2$, on retrouve la variance du processus aléatoire soit $a^2/2$.

3. Processus stationnaires

Beaucoup de processus aléatoires observés en pratique ont des propriétés statistiques qui ne dépendent pas du temps ou l'observation est faite. On dit que ce sont des processus stationnaires.

On désigne par *stationnaire* les processus dont les caractéristiques statistiques sont indépendantes de l'origine du temps. Un processus aléatoire est dit stationnaire au sens strict lorsque toutes ces caractéristiques statistiques c'est à dire tous ses moments à tout ordre sont indépendants de l'origine du temps.

$$\begin{aligned} \Leftrightarrow p(x, t_i) &= p(x); \\ \mu_x(t_i) &= \mu_x \\ R_{xy}(t_1, t_2) &= R_{xy}(\tau) \text{ avec } \tau = t_2 - t_1 \\ &\dots \end{aligned}$$



La stationnarité ne signifie pas pour autant que le processus est indépendant du temps mais plutôt que ses propriétés statistiques ne dépendent pas du moment auquel on commence à les estimer. Ainsi, la température est un processus non stationnaire alors que le lancer de dé est un processus stationnaire.

On peut considérer que de nombreux phénomènes sont approximativement stationnaires sur des durées d'observation finies. Comme nous n'avons pas toujours accès à $p(x, t_i)$, on se contentera de la *stationnarité au sens large (SSL)* lorsque seul μ_x et R_x sont indépendants de t .

Ainsi :

- Un processus stationnaire est dit stationnaire d'ordre 1 si sa moyenne et sa variance sont constantes et donc indépendantes de tout décalage temporelle :

$$\text{et} \quad \mu_x(t) = \mu_x = \text{cste} \quad \sigma_x^2(t_i) = \sigma_x^2$$

- Un processus stationnaire est dit stationnaire d'ordre 2 si ses statistiques d'ordre 2 ne dépendent que de l'écart temporel entre les deux instants t_1 et t_2 :

$$R_x(t_1, t_2) = R_x(\tau) \text{ avec } \tau = t_2 - t_1$$

Reprenons l'exemple précédent $X(t, u) = a \cos(\omega t + u)$ étudions sa stationnarité au sens large lorsque u est uniforme sur $[0, 2\pi]$: la moyenne, la variance sont indépendantes de t et l'auto-corrélation statistique ne dépend que de $t_1 - t_2$. Le processus est donc stationnaire au sens large.

Exemple d'application:

Montrer que le processus $x(t) = a \cos(2\pi t) + b \sin(2\pi t)$ où a, b sont des v.a décorréliées de moyenne nulle et de variance unité est stationnaire au sens large.

Propriétés de la fonction d'autocorrélation pour x aléatoire réel SSL

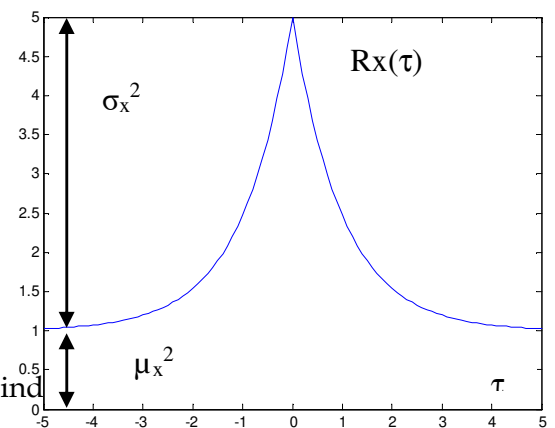
- La fonction de corrélation est paire : $R_x(\tau) = R_x(-\tau)$ avec $\tau = t_2 - t_1$

- La fonction est maximum à l'origine $R_x(0) \geq |R_x(\tau)|$

- Valeurs à l'origine et à l'infini : $R_x(0) = E(x(t)^2) = \mu_x^2 + \sigma_x^2$

$$\text{et} \quad \lim_{\tau \rightarrow \infty} R_x(\tau) = \lim_{\tau \rightarrow \infty} E\{x(t)x(t-\tau)\} = \mu_x^2$$

En effet, si $x(t)$ ne contient ni composante périodique ni composante ind les variables $x(t)$ et $x(t - \tau)$ deviennent statistiquement décorréliées.



Cas discret :

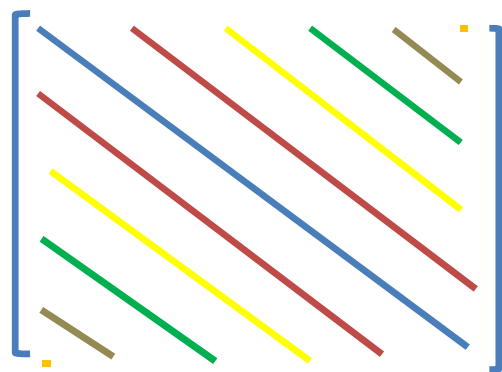
$$\text{On remplacera } t \text{ par } n \rightarrow \quad \mu_x(n) = \mu_x = E\{x(n)\} \quad R_x(n_1, n_2) = E\{x(n_1)x^*(n_2)\}$$

Un processus aléatoire X discret SSL possédera une moyenne constante et une autocorrélation statistique ne dépendant que de $k=n_1-n_2$ soient :

$$\mu_x(n) = \mu_x = \text{cste} \quad \text{et} \quad R_x(n_1, n_2) = R_x(k) \quad \text{avec} \quad k = n_1 - n_2$$

$$R_x(n_1, n_2) = \begin{bmatrix} R_x(0) & R_x(1) & R_x(2) & \dots & R_x(N-1) \\ R_x(1) & R_x(0) & R_x(1) & \dots & R_x(N-2) \\ R_x(2) & R_x(1) & R_x(0) & \dots & R_x(N-3) \\ \dots & \dots & \dots & \dots & \dots \\ R_x(N-1) & R_x(N-2) & R_x(N-3) & \dots & R_x(0) \end{bmatrix}$$

$$\text{SSL} \Rightarrow R_x(n_1, n_2) = R_x(n_1 - n_2) = \begin{bmatrix} R_x(1,1) & R_x(1,2) & R_x(1,3) & \dots & R_x(1,N) \\ R_x(2,1) & R_x(2,2) & R_x(2,3) & \dots & R_x(2,N) \\ R_x(3,1) & R_x(3,2) & R_x(3,3) & \dots & R_x(3,N) \\ \dots & \dots & \dots & \dots & \dots \\ R_x(N,1) & R_x(N,2) & R_x(N,3) & \dots & R_x(N,N) \end{bmatrix}$$



$$R_x(n_1 - n_2) = R_x(k)$$

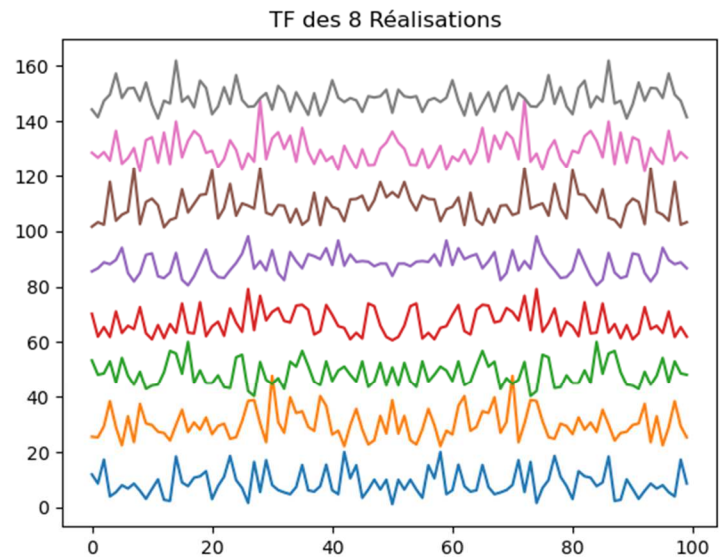
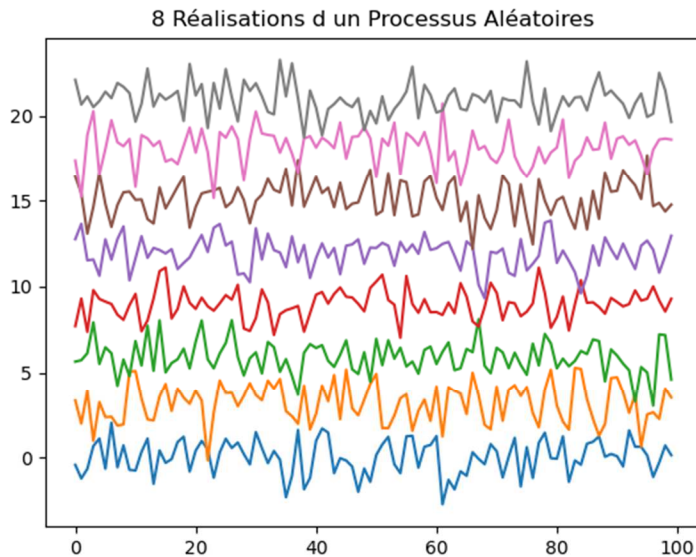
Ainsi, la matrice de corrélation d'un processus stationnaire discret est une matrice Toeplitz carrée. Une matrice carrée est dite Toeplitz si tous les éléments d'une même diagonale ou sous-diagonale sont égaux. On voit directement que c'est le cas ici. Cette propriété est directement liée à la propriété de stationnarité (au sens large) du processus.

Puissance et DSP : Les personnes qui conversent (dans un café ou en cours) génèrent un signal aléatoire qui selon son volume global possèdera une puissance. Si $X(t)$ est SSL, on peut calculer l'espérance de la puissance instantanée par:

$$P_X = E(x(t)^2) = R_x(0) = \mu_x^2 + \sigma_x^2$$

$$\text{L'énergie est alors donnée par : } E_X = \int P_X dt = \int E(x(t)^2) dt$$

Quant au spectre pour un signal aléatoire, il faut noter que chaque réalisation fournira un spectre différent (Voir exemple ci-dessous)



Or, nous avons déjà un outil statistique qui contient une information unique lorsque le processus est considéré SSL: c'est la fonction de corrélation statistique dont la TF fournira une information sur la distribution fréquentielle de la puissance moyenne du signal.

On définit, alors, la densité spectrale de puissance d'un signal aléatoire $X(t)$ stationnaire au sens large comme la fonction de la fréquence f donnée par la TF de la fonction de corrélation statistique du signal:

$$S_x(f) = \int_{-\infty}^{+\infty} R_x(\tau) e^{-j2\pi f\tau} d\tau$$

$$\text{La puissance moyenne totale du processus est : } P_x = \int_{-\infty}^{+\infty} S_x(f) df = R_x(0)$$

Comme le spectre représente une moyenne sur l'ensemble des réalisations possibles du processus, une réalisation particulière peut toujours avoir un spectre de puissance différent de $S_x(f)$.

Bruit blanc : Un bruit blanc est un signal aléatoire stationnaire au second ordre dont la densité spectrale de puissance est constante sur tout l'axe des fréquences.

Il est défini par : $R_x(\tau) = \sigma_x^2 \delta(\tau)$ ce qui implique que sa moyenne sera nulle ($\mu_x=0$) et qu'il est décorrélé.

La TF de la corrélation statistique est, alors, donnée par $S_x(f) = \sigma_x^2$

Dans le cas discret, on aura $\Rightarrow R_x(k) = \sigma_x^2 \delta(k)$ et $R_x(k) = 0$ pour $k \neq 0$ $\mu_x(k) = 0$

On dit alors que le bruit est décorrélé.

Par analogie avec la lumière blanche, on dénomme un tel signal bruit blanc. Il existe donc différents type de bruit blanc en fonction de la variable aléatoire décrivant le bruit. Les plus répandus sont le bruit blanc gaussien où la variable aléatoire suit une loi gaussienne et le bruit blanc uniforme où la variable aléatoire suit une loi uniforme.

4. Processus ergodiques

Il n'est pas toujours possible de réaliser un nombre suffisant de mesures pour établir les propriétés statistiques d'un processus aléatoire. Il est plus facile de lancer un dé 1000 fois que de réquisitionner 1000 personnes pour lancer 1000 dés. Tout comme pour caractériser le bruit thermique, on réaliserait plusieurs mesures en usant de la même résistance au lieu d'en employer 1000. Cela signifie que l'on assimilera les résultats obtenus sur une réalisation à ceux obtenus pour un instant donné t_i pour différentes réalisations.

Un processus aléatoire stationnaire est dit *ergodique* lorsque les valeurs moyennes statistiques et temporelles sont **identiques**.

$$\mu_x = \overline{\mu_x}; \quad R_x(\tau) = \overline{R_x(\tau)} \dots$$

$$\text{où } \bar{x} = \lim_{T \rightarrow +\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) \cdot dt \quad \text{et} \quad \overline{R_{xy}(\tau)} = \overline{x(t) \cdot y^*(t - \tau)}$$

D'un point de vue pratique lorsqu'un phénomène aléatoire est ergodique et stationnaire, on peut mesurer avec un seul appareil fiable à partir de n'importe quel instant.

$$\overline{\mu_x} = \lim_{T \rightarrow +\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) \cdot dt \Rightarrow \mu_x = \overline{\mu_x} \quad \overline{R_x(\tau)} = \lim_{T \rightarrow +\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t)x(t - \tau) \cdot dt \Rightarrow R_x(\tau) = \overline{R_x(\tau)}$$

Reprenons, une énième fois l'exemple 2 et calculons les moyennes temporelles :

$$E[X(t, u_i)] = \overline{X}(u_i) = \frac{1}{T} \int_{-T/2}^{+T/2} a \cos(\omega t + u_i) dt = 0 \quad E[X^2(t, u_i)] = \overline{X^2}(u_i) = \frac{1}{T} \int_{-T/2}^{+T/2} a^2 \cos^2(\omega t + u_i) dt = \frac{a^2}{2}$$

Il est donc ergodique d'ordre 1

Si le processus est ergodique $S_x(f)$ représente le spectre de puissance de n'importe quelle réalisation $x_i(t)$:

$$P_x = \int_{-\infty}^{+\infty} S_x(f) df = \lim_{T \rightarrow +\infty} \frac{1}{T} \int_{-T/2}^{T/2} x_i(t)^2 dt \quad \text{Où } x_i(t) \text{ est une réalisation de } x(t).$$

Dans le cas où le signal $x(t)$ est ergodique, stationnaire, on détermine $S_x(f)$ par le calcul de la DSP temporel sur T :

$$S_x(f) = TF\{R_x(\tau)\} = \lim_{T \rightarrow \infty} \frac{1}{T} |X_T(f)|^2 \quad \text{Où } X_T(f) \text{ est la TF de } x(t) \text{ limité à l'intervalle } [0, T]$$

5. Exemples de Processus

Il existe de nombreuses classes de processus particuliers : les processus de Markov (et notamment les chaînes de Markov quand t est discret), les martingales, les processus Gaussiens, les processus de Poisson.

- **Poisson**

La distribution de Poisson est destinée à modéliser la fréquence des événements au cours d'un intervalle de temps fixe. Étant donné un intervalle de temps de longueur, t , et le taux d'arrivée des événements, λ , le nombre attendu d'événements au cours de cet intervalle de temps est :

$$\mu = \lambda \cdot t$$

Pour que le processus de Poisson soit réalisable, les hypothèses suivantes doivent être remplies :

1. La probabilité de succès sur une très courte période de temps est le paramètre λ multiplié par cette période de temps.
2. La probabilité que plus d'un événement de succès se produise dans l'intervalle de temps défini n'est pas significative. En d'autres termes, la probabilité que plus d'une expérience réussisse dans un intervalle de temps fixe est très faible, et donc non importante ou non significative.
3. La probabilité qu'un événement réussisse se produise pendant un intervalle de temps défini ne dépend pas de ce qui s'est passé précédemment. Autrement dit, chaque expérience réussie est indépendante de l'expérience précédente.

Le processus de Poisson est connu en statistique comme un processus stochastique qui essaie d'enregistrer des événements très improbables en temps continu.

Exemples: panne, accident nucléaire, défaut de fabrication dans une chaîne. Il est également utilisé en Télécommunications pour estimer le nombre d'appels dans un laps de temps donné.

- **Markovien**

L'axiome de Markov traduit que la probabilité de n'importe quel comportement futur, le présent étant connu, n'est pas modifié par toute connaissance supplémentaire du passé.

Un processus est markovien, si pour tout instant u , pour toute valeur donnée $X_u = x$, la probabilité pour que le processus prenne la valeur y à un instant quelconque t ultérieur ($t > u$), ne dépend pas des valeurs prises par le processus avant l'instant u . Le processus est dit sans mémoire.

Les chaînes de Markov sont des processus markoviens qui évoluent dans le temps discret selon des transitions probabilistes. Elles vont donc décrire un ensemble de phénomènes aléatoires selon le principe des processus stochastiques.

Exemples : la hauteur du niveau d'un barrage à un instant t , le cours du pétrole à la fin de la journée

- **Gaussien**

Les processus gaussiens jouent un rôle crucial dans la théorie des processus stochastiques du fait que :

- Certains processus stochastiques peuvent être approchés par des processus gaussiens,
- Les calculs sont facilités dans le cadre des processus gaussiens.
- Théorème central limite et approximations : Chaque fois qu'un phénomène peut être considéré comme la résultante d'un grand nombre de causes aléatoires indépendantes (ex: notes obtenues à l'examen, poids d'un enfant), on peut résumer que ce phénomène suit une loi de distribution normale: c'est le théorème central limite. Ainsi, la distribution statistique de la somme de n V.A. indépendantes et de même loi tend vers la loi Normale quand n tend vers l'infini.

Un processus (X_t) réel est gaussien si : $\forall n \forall, t_1, \dots, t_n$, le vecteur $(X_{t_1}=x_1, \dots, X_{t_n}=x_n)$ est gaussien.

$$Y = \sum_{k=1}^n \alpha_k X_k = \alpha^T \underline{X}$$

Rappelons que toutes les probabilités marginales d'un processus gaussien sont gaussiennes et que toute combinaison linéaire de marginales d'un processus gaussien est gaussienne.

$$p(x_1, x_2, \dots, x_n) = \frac{1}{\sqrt{(2\pi)^n \det(C_X)}} e^{-\frac{1}{2}(\underline{X}-\underline{\mu_X})^T C_X^{-1}(\underline{X}-\underline{\mu_X})}$$

Où

$$\underline{X} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}, \quad \underline{\mu_X} = \begin{bmatrix} \mu_{X1} \\ \vdots \\ \mu_{Xn} \end{bmatrix}, \quad \underline{C_X} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} - \begin{bmatrix} \mu_{X1} \\ \vdots \\ \mu_{Xn} \end{bmatrix}$$

Exemple : Un mouvement brownien (standard) réel est un processus gaussien centré à trajectoires continues. C'est un processus à accroissements indépendants stationnaires et gaussiens.

6. Estimation de la densité spectrale

Les quantités $E\{x\}$, $E\{x^2\}$, $S_{xx}(f)$,... sont impossibles à calculer sur un ordinateur car elles nécessiteraient un nombre de points infini. Sur calculateur, on ne dispose que d'une séquence discrète et finie de N points. En réalité, on calcule des estimées de ces grandeurs en supposant généralement le signal stationnaire et ergodique. On remplace alors le calcul des moyennes statistiques par des moyennes temporelles.

Nous calculerons la DSP sur un nombre de point finis. Ainsi, la transformée de Fourier calculée sur la séquence finie sera donc convoluée avec le spectre de la fenêtre rectangulaire c'est-à-dire un sinus cardinal. Rappelons que les propriétés spectrales du sinus cardinal ne sont pas bien adaptées à l'analyse spectrale du signal (lobes secondaires importants). Pour y remédier, on emploie des fenêtres de pondérations (Hamming, Hanning, Kaiser, etc.)

La méthode du périodogramme à prendre la TF d'une réalisation : $\hat{S}_{xx}(f) = \frac{1}{N} |Y(f)|^2$ avec $Y(f) = \sum_{k=0}^{N-1} y_k e^{-2\pi jfk}$

$$\begin{aligned} \hat{S}_{xx}(f) &= \frac{1}{N} \sum_{l=0}^{N-1} y_l e^{-2\pi jfl} \sum_{m=0}^{N-1} y_m e^{2\pi jfm} = \frac{1}{N} \sum_{l=0}^{N-1} \sum_{m=0}^{N-1} y_l y_m e^{-2\pi jf(l-m)} \\ \text{D'où} \quad &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=k}^{N-1} y_l y_{l-k} e^{-2\pi jfk} = \sum_{k=0}^{N-1} \hat{R}_{yy}(k) e^{-2\pi jfk} = TF(\hat{R}_{yy}(k)) \end{aligned}$$

Le calcul de l'espérance de $\hat{S}_{xx}(f)$ donne :

$$E\{\hat{S}_{xx}(f)\} = E\left\{ \sum_{k=0}^{N-1} \hat{R}_{yy}(k) e^{-2\pi jfk} \right\} = E\left\{ \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=k}^{N-1} y_l y_{l-k} e^{-2\pi jfk} \right\}$$

$$E\{\hat{S}_{xx}(f)\} = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=k}^{N-1} E\{y_l y_{l-k}\} e^{-2\pi jfk} = \sum_{k=0}^{N-1} \frac{N-k}{N} R_{yy}(k) e^{-2\pi jfk} = S_{yy}(f) * N \text{sinc}(fN)^2$$

Le périodogramme est un estimateur biaisé. Le périodogramme est donc en moyenne la convolution du véritable spectre avec la transformée de Fourier de la fenêtre triangulaire. Néanmoins, lorsque $N \rightarrow \infty$, le biais devient nul. La variance est pratiquement indépendante de N et proportionnelle au spectre $\approx S_x(f)^2$. Remarquons que la qualité de l'estimation par périodogramme sera d'autant meilleure que le signal est observé sur une longue plage de stationnarité.

Afin de diminuer la variance de cet estimateur, on peut utiliser un périodogramme moyenné. Cela consiste à séparer le signal en K tranches (de longueur N/K), à calculer le périodogramme sur chaque tranche et à faire la moyenne. Du fait des K moyennages, la variance est presque divisée par K : néanmoins, les tranches étant plus courtes, la résolution diminue ($\Delta f = f_e/N \rightarrow f_e/K$). En pratique, un taux de recouvrement de 40% donne de bons résultats. Au delà, l'indépendance entre les tranches n'est plus respectée.

b. La méthode du corrélogramme consiste à calculer d'abord l'estimée de $R_{xx}(k)$ de la fonction d'autocorrélation puis à prendre pour estimée de la densité spectrale la TF de cette estimée.

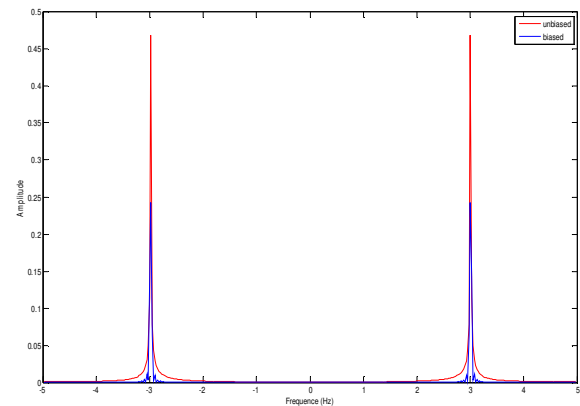
$$\hat{S}_{xx}(f) = TF[\hat{R}_{xx}(k)]$$

A partir d'un échantillon ($x_0, x_1, \dots, x_{N-1}$) indépendants et identiquement distribués, plusieurs estimateurs sont alors possible:

$$\hat{R}_{xx}(k) = \frac{1}{N} \sum_{n=k}^{N-1} x(n) x^*(n-k) \quad \hat{R}_{xx}(k) = \frac{1}{N-k} \sum_{n=k}^{N-1} x(n) x^*(n-k)$$

Le premier estimateur est biaisé car on ne tient pas compte du nombre d'échantillons disponibles qui varie avec le pas k . Le deuxième en tient compte, il est non biaisé.

Les corrélogrammes présentent chacun un extrema pour la fréquence normalisée 3 qui est bien la fréquence de la sinusoïde étudiée. Il ressort également de ces courbes que le corrélogramme non-biaisé permet de mieux faire ressortir la fréquence de la sinusoïde mais au dépend d'une plus grande variance sur les bords.



c) Dans cette méthode, on recherche un modèle AR, MA ou ARMA pour la séquence x_k . Dans le cas d'un modèle auto-régressif $x_k = a_1 x_{k-1} + \dots + a_r x_{k-r} + u_k$

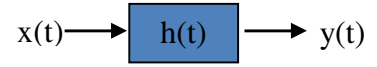
$$\hat{S}_{xx}(f) = \left| \frac{1}{1 + \sum_{k=1}^r a_k e^{-i2\pi f k}} \right|^2 u_0$$

7. Filtrage des Processus aléatoires

Un signal aléatoire est amené à être transmis, analysé, transformé, etc. Conserve-t-il son caractère aléatoire? sa stationnarité? Que deviennent sa moyenne et son auto-corrélation statistiques lors d'un filtrage linéaire. On examinera la transformation des caractéristiques du signal dans le domaine fréquentiel qui

permettra d'aborder la notion de filtre formeur. Ces connaissances nous permettront d'envisager une application directe qui est le filtrage optimal et adapté.

Soit un système linéaire et invariant dans le temps (SLIT) défini par sa réponse impulsionnelle $h(t)$ ou sa fonction de transfert $H(f)$:



Rappelons que la réponse du système linéaire et invariant à un signal quelconque déterministe est donnée par :

$$y(t) = x(t) * h(t) = \int x(\tau)h(t - \tau)d\tau$$

Cette expression implique que si $x(t)$ est un signal aléatoire, le signal en sortie $y(t)$ est forcément un signal aléatoire puisque la sortie est une somme pondérée de l'entrée. Il faudra donc caractériser $y(t)$ de manière statistique, de même que pour $x(t)$, en étudiant les grandeurs statistiques déjà vues au chapitre précédent.

Moyenne et autocorrélation statistiques de $y(t)$

Supposons que $x(t)$ est un processus aléatoire, la moyenne $\mu_y(t)$ du signal de sortie (aléatoire aussi) peut se calculer par :

$$\begin{aligned}\mu_y(t) &= E\{y(t)\} = E\{x(t) * h(t)\} = E\left\{\int h(\tau)x(t - \tau)d\tau\right\} \\ &= \int h(\tau)E\{x(t - \tau)\}d\tau = \int h(\tau)\mu_x(t - \tau)d\tau = h(t) * \mu_x(t)\end{aligned}$$

De la même façon, on peut calculer l'auto-corrélation statistique de $y(t)$ soit $R_y(t_1, t_2)$:

$$\begin{aligned}R_y(t_1, t_2) &= E\{y(t_1)y^*(t_2)\} = E\{x(t_1) * h(t_1).x^*(t_2) * h^*(t_2)\} = E\left\{\int h(\tau_1)x(t_1 - \tau_1)d\tau_1 \int h^*(\tau_2)x^*(t_2 - \tau_2)d\tau_2\right\} \\ &= \int \int h(\tau_1)h^*(\tau_2)E\{x(t_1 - \tau_1)x^*(t_2 - \tau_2)\}d\tau_1d\tau_2\end{aligned}$$

Si $x(t)$ est stationnaire au sens large (SSL) alors:

- La moyenne statistique de $y(t)$ devient :

$$\mu_y(t) = \int h(\tau)\mu_x(t - \tau)d\tau = \int h(\tau)\mu_x d\tau = \mu_x \int h(\tau)d\tau = \mu_x H(0) \quad (f = 0)$$

- La corrélation statistique de y devient :

$$\begin{aligned}R_y(t_1, t_2) &= \int \int h(\tau_1)h^*(\tau_2)E\{x(t_1 - \tau_1)x^*(t_2 - \tau_2)\}d\tau_1d\tau_2 \\ &= \int \int h(\tau_1)h^*(\tau_2)R_x(\tau - \tau_1 + \tau_2)d\tau_1d\tau_2 = \int h(\tau)h^*(\tau)R_x(\tau)d\tau = R_y(\tau)\end{aligned}$$

Ainsi, si le signal d'entrée est stationnaire au sens large (SSL) alors le signal de sortie sera aussi SSL

$$\text{Sachant que } f(\tau) * h(\tau) = \int f(\tau_2)h(\tau - \tau_2)d\tau_2 = \int f(\tau - \tau_2)h(\tau_2)d\tau_2$$

$$\text{Alors } f(\tau) * h(-\tau) = \int f(\tau_2)h(\tau_2 + \tau)d\tau_2 = \int f(\tau + \tau_2)h(\tau_2)d\tau_2$$

$$\text{C'est ainsi que } R_y(\tau) = R_x(\tau) * h(\tau) * h^*(-\tau)$$

Densité spectrale

En appliquant la transformée de Fourier aux deux membres de l'équation précédente, la densité spectrale de puissance est obtenue :

$$S_y(f) = TF(R_y(\tau)) = TF(R_x(\tau) * h(\tau) * h^*(-\tau)) = S_x(f) \cdot H(f) \cdot H^*(f) = |H(f)|^2 S_x(f)$$

Ainsi, la densité spectrale de puissance du signal de sortie est égale à la densité spectrale de puissance du signal d'entrée multipliée par le carré du module de la réponse en fréquences du système (la phase de $H(f)$ n'intervient pas). C'est une propriété très importante à la base de nombreuses applications dont notamment la notion de filtre formeur et le filtrage optimal.

Par ailleurs, si l'on souhaite calculer la corrélation du signal de sortie en évitant la lourdeur de calcul inhérente au produit de convolution, on calculera la densité spectrale de puissance du signal de sortie et on prendra la transformée de Fourier inverse.

$$R_y(\tau) = TF^{-1}(|H(f)|^2 S_x(f)) = \int |H(f)|^2 S_x(f) e^{2\pi j f \tau} df$$

La puissance moyenne du signal de sortie est alors obtenue par : $E\{y(t)^2\} = R_y(0) = \int |H(f)|^2 S_x(f) df$

Formule des interférences

Soient $y_1(t)$ la sortie d'un système $h_1(t)$ dont l'entrée est un signal aléatoire $x_1(t)$ et $y_2(t)$ la sortie d'un système $h_2(t)$ dont l'entrée est un signal aléatoire $x_2(t)$. La formule de l'interférence permet de relier l'intercorrélation entre les sorties de deux filtres à celles des entrées de ces filtres:

$$S_{y_1 y_2}(f) = H_1(f) S_{x_1 x_2}(f) H_2^*(f)$$

On a aussi : $R_{xy}(\tau) = R_x(\tau) * h^*(-\tau) \Rightarrow S_{xy}(f) = S_x(f) H^*(f)$

et $R_{yx}(\tau) = R_x(\tau) * h(\tau) \Rightarrow S_{yx}(f) = H(f) S_x(f)$

Exemple 1 : Soit le processus stochastique $x(t)$ SSL de corrélation statistique: $R_x(\tau) = \sigma^2 e^{-\frac{|\tau|}{\theta}}$ et le système linéaire et invariant de réponse impulsionnelle $h(t) = 5e^{-2t}U(t)$. On obtient alors:

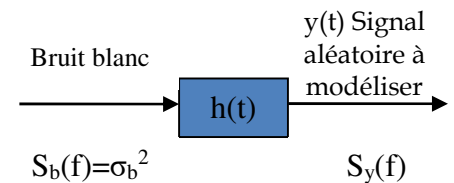
$$-\mu_y(t) = \mu_x H(0) = 0 \times \frac{5}{2 + 2\pi j f} \Big|_{f=0} = 0 \times 5/2 = 0$$

$$-S_y(f) = |H(f)|^2 S_x(f) = \left| \frac{5}{2 + 2\pi j f} \right|^2 \cdot TF\left(\sigma^2 e^{-\frac{|\tau|}{\theta}}\right) = \frac{25}{4 + (4\pi^2 f^2)} \cdot \frac{2\sigma^2 \theta}{1 + 4\pi^2 f^2 \theta^2}$$

Exemple 2 : Considérons par exemple le cas où le signal en entrée est un bruit blanc $b(t)$. Sa DSP est donc une fonction constante. Alors, en sortie, on aura un signal tel que : $S_y(f) = \text{constante} |H(f)|^2$

Notion de filtre formeur

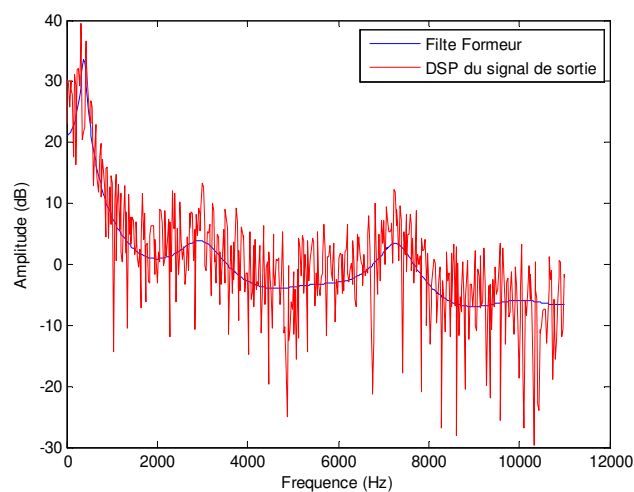
Supposons donné un signal aléatoire $y(t)$: On appelle filtre formeur de $y(t)$; le filtre de fonction de transfert $H(f)$; tel que $y(t)$ est généré par passage d'un bruit blanc $b(t)$ dans $H(f)$.



La détermination du filtre formeur s'effectue en inversant la formule précédente:

$$|H(f)|^2 = S_y(f) / \sigma_b^2$$

On pourra donc décrire le signal aléatoire par les paramètres du filtre et la variance du bruit blanc. En effet, si on fait passer le bruit blanc dans un filtre linéaire et stationnaire à paramètres ajustables et si on obtient le signal désiré (à modéliser) à la sortie du filtre, alors on peut dire que toute l'information spectrale est contenue dans le filtre représenté par ses coefficients (Voir exo 1 du TP n°3)



Remarque: Ainsi le bruit blanc joue pour l'aléatoire l'équivalent de la distribution de Dirac pour le déterministe.

Rappelons que le bruit blanc n'a pas d'existence physique car il serait de puissance infinie. Une approximation du bruit blanc à bande limitée appelé aussi bruit blanc coloré est défini par :

$$S_b(f) = \begin{cases} \sigma_b^2 & |f| < B \\ 0 & \text{ailleurs} \end{cases}$$

- **Application 1: Filtrage adapté et filtrage optimal**

La transmission d'un signal s'accompagne de distorsions (dus aux milieux de transmission) qu'il serait souhaitable d'éliminer ou au moins d'atténuer avant tout traitement ultérieur. Si l'on connaît le signal (déterministe) d'origine, on parlera de détection. Dans le cas contraire ou si le signal est aléatoire on emploiera le terme estimation. Il existe de nombreuses approches de détection, on peut, entre autres procéder par filtrage qui permettra de rehausser le signal noyé dans le bruit.

Considérons un signal déterministe $x(t)$ supposé connu, dont on souhaite tester la présence possible dans une observation $s(t)$. Le bruit d'observation est supposé quant à lui stationnaire SSL de densité spectrale $S_b(f)$. On cherche un filtre $H(f)$ qui maximise le SNR à un instant précis T_0 .

On suppose donc que le signal utile $x(t)$ est noyé dans un bruit $b(t)$ stationnaire SSL additif, d'où :

$$s(t) = x(t) + b(t)$$

On filtre le signal par un filtre linéaire dont la réponse impulsionnelle est $h(t)$. A la sortie du filtre, on obtient un signal :

$$y(t) = s(t) * h(t) = x(t) * h(t) + b(t) * h(t) = x_2(t) + b_2(t)$$

A la sortie du filtre et à l'instant T_0 , le SNR s'écrit : $SNR(T_0) = \frac{Puis(x_2(T_0))}{Puis(b_2(T_0))}$

Le but étant de trouver $h(t)$, on va alors exprimer le SNR en fonction de $h(t)$ (ou $H(f)$), ainsi :

- Le signal au numérateur $x_2(t)$ est déterministe donc sa puissance s'exprime comme suit:

$$Puis(x_2(T_0)) = |x_2(T_0)|^2 = |TF^{-1}(X_2(f))|^2 = \left| \int X(f)H(f)e^{2\pi j f T_0} df \right|^2$$

- Le signal $b_2(t)$ est issu du filtrage d'un signal aléatoire $b(t)$ SSL donc il est aussi SSL, c'est ainsi que sa puissance pourra se formuler comme suit:

$$Puis(b_2(T_0)) = E\{b_2(T_0)^2\} = R_{b_2}(\tau=0) = TF^{-1}(S_{b_2}(f))\Big|_{\tau=0} = \int S_b(f)|H(f)|^2 df$$

Ainsi, on arrive à exprimer le SNR comme suit :

$$SNR(T_0) = \frac{Puis(x_2(T_0))}{Puis(b_2(T_0))} = \frac{\left| \int X(f)H(f)e^{2\pi j f T_0} df \right|^2}{\int S_b(f)|H(f)|^2 df} \quad SNR(T_0) = \frac{\left| \int a(f)b^*(f)df \right|^2}{\int a(f)a^*(f)df}$$

Que l'on peut écrire aussi, comme étant :

$$\text{Avec } a(f) = \sqrt{S_b(f)}H(f) \quad \text{et } b(f) = X^*(f)e^{-2\pi j f T_0} / \sqrt{S_b(f)}$$

Pour trouver $H(f)$ qui maximise ce rapport, on fait appel à l'inégalité de cauchy-schwartz:

$$\left| \int a(f)b^*(f)df \right|^2 \leq \int a(f)a^*(f)df \quad \int b(f)b^*(f)df$$

$$\text{Ce qui nous donne : } SNR(T_0) = \frac{\left| \int a(f)b^*(f)df \right|^2}{\int a(f)a^*(f)df} \leq \int b(f)b^*(f)df$$

avec égalité (soit le maximum) lorsque $a(f)=kb(f) \Rightarrow$ Cette dernière nous fournit l'expression suivante du filtre optimal $H(f)$:

$$H(f)_{optimal} = k.X^*(f)e^{-2\pi jfT_0} / S_b(f)$$

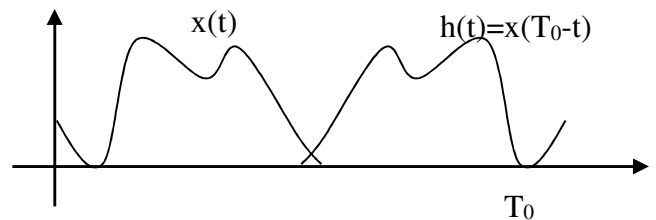
L'expression du SNR devient : $SNR(T_0)_{max} = \int \frac{|X(f)|^2}{S_b(f)} df$ (Ce maximum est indépendant de T_0)

Cas particulier : Par ailleurs, si le bruit $b(n)$ est blanc, on parle de filtre adapté:

$$H(f)_{adapté} = k / \sigma_b^2 X^*(f) e^{-2\pi jfT_0} \quad \text{donnant} \quad h(t)_{adapté} = k / \sigma_b^2 x^*(T_0 - t)$$

$$SNR(T_0)_{max} = \frac{E_x}{\sigma_b^2}$$

La réponse impulsionnelle du filtre représente le signal utile $x(t)$ renversé et décalé de T_0 . Le filtrage adapté revient à effectuer l'inter-corrélation entre l'observation et le signal à détecter.



Cette réponse n'est pas causale ce qui ne permet pas de l'appliquer en temps réel. Cependant, ce filtre peut-être appliqué à un signal sauvegardé.

Exemple1 : Détection d'une impulsion

Considérons un système émettant une impulsion $x(t)$ rectangulaire de durée T_0 et d'amplitude A . Le bruit additif est blanc. Le filtre aura la même expression, soit $h(t)=x(t)$.

Exemple 2 : En sonar ou en radar, on cherche à localiser une « cible ». Cette cible peut être par exemple un bateau ou un avion. Pour cela, on procède de la façon suivante : on émet un signal $x(t)$, qui parcourt la distance d jusqu'à la cible, sur laquelle il sera réfléchi en direction d'un récepteur. Le récepteur reçoit alors le signal $y(t)$ bruité, atténué (de a) et retardé de T_{AR} (voir TP n°3)

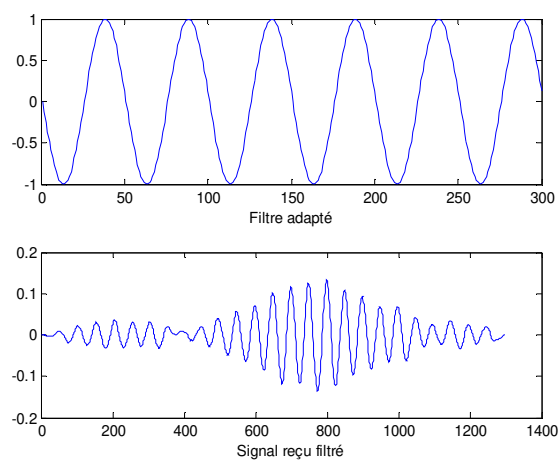
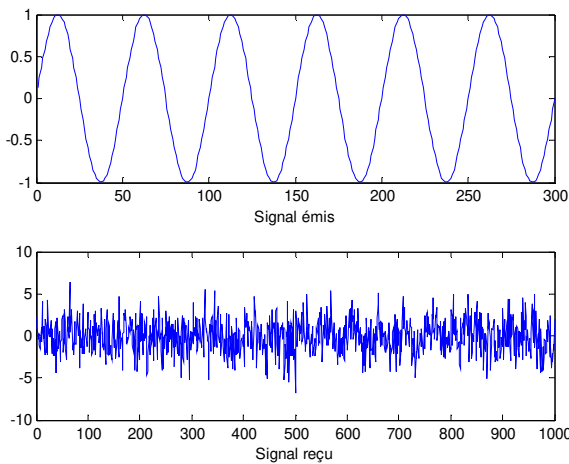
$$y(t) = a.x(t - T_{AR}) + b(t). \text{ où } T_{AR} \text{ correspondant au temps d'aller-retour } (T_{AR} = 2d/v)$$

On utilisera alors comme filtre adapté le filtre adapté à $x(t)$, soit $h(t) = x^*(T_0 - t)$

La sortie de ce filtre sera l'intercorrélation des signaux $y(t)$ et $x(t)$. On obtient alors :

$$R_{yx}(t - T_0) = a.R_{xx}(t - T_0 - T_{AR}) + R_{bx}(t - T_0).$$

Sachant que l'autocorrélation est maximale en 0, l'intercorrélacion sera maximale pour $t = T_0 + T_{AR}$ ce qui nous permettra de déterminer T_{AR} .



Remarques

- Le choix du signal 'x' est important : Il est souhaitable que le max de sa fonction d'autocorrélation R_{xx} soit facile à identifier (pic bien visible comme pour la fonction triangle)

- Dans le cas où le signal est sinusoïdal, le filtre adapté est un passe bande idéal centré sur la fréquence du signal (détection synchrone).

Exercice d'Application (Examen 2014/2015)

Soit $x(t) = 1 - t/T$ avec $t \in [0, T]$

On utilise ce signal pour déterminer la distance d'un objet. Le signal reçu par le récepteur après réflexion sur l'objet est : $y(t) = \alpha x(t - T_D) + b(t)$ où α est une constante réelle positive et $b(t)$ est un bruit blanc de densité spectrale de puissance σ^2 . On veut maximiser le rapport signal sur bruit

1. Déterminer la réponse impulsionnelle du filtre $h(t)$ telle que $\int h(t)^2 dt = 1$ (prendre $T_0 = T$).
2. Donner le rapport signal sur bruit après filtrage.
3. Exprimer l'inter-corrélation temporelle $R_{yx}(t - T_D)$ en fonction de l'auto-corrélation de $x(t)$ et de l'inter-corrélation des signaux $b(t)$ et $x(t)$
4. Comment déterminer le temps T_D ?
5. Pourquoi ne peut-on pas utiliser le filtrage optimal pour un signal déterministe dont on ignore l'expression?

• Application 2: Filtrage de Wiener

Il fait partie de la famille des estimateurs linéaires à variance minimale qui consistent à chercher une estimée

$\hat{y}$ de y qui soit une fonction linéaire des observations, c'est-à-dire: $\hat{y} = \sum_{i=1}^n h_i x_i = h^T x$

Dans le cas de l'estimation en moyenne quadratique, le filtre h doit alors être déterminé de telle sorte que la variance de l'erreur d'estimation $E\{(y - \hat{y})^2\}$ soit minimale :

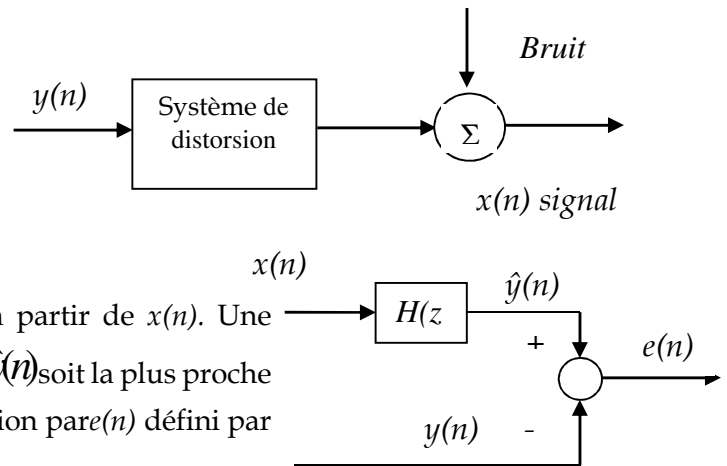
$$E\{(y - \hat{y})^2\} = E\{(y - h^T x)^T (y - h^T x)\} = E\{y y^T\} - E\{y^T h^T x\} - E\{h x^T y\} + E\{h x^T h^T x\}$$

On commence par dériver par rapport à h puis on annule la dérivée, ce qui nous donne

$$-E\{y^T x\} - E\{x^T y\} + 2h E\{x^T x\} = 0 \Rightarrow h = E\{y^T x\} / E\{x^T x\} = R_{yx} / R_{xx}$$

Dans de nombreuses applications, les signaux temporels sont entachés d'un bruit que l'on souhaite supprimer ou du moins réduire. Comme le signal utile aléatoire occupe les mêmes bandes de fréquences que le signal parasite, on ne peut recourir au filtrage classique. Le filtre de Wiener apporte une solution à ce problème lorsque le processus est stationnaire. Les filtres de Wiener sont dits optimum au sens du critère de l'erreur quadratique moyenne entre leur sortie et une sortie désirée. .

La figure suivante illustre un problème courant d'estimation linéaire. $y(n)$ correspond au signal aléatoire stationnaire qui nous intéresse mais n'est pas directement accessible. Seul $x(n)$ l'est, il est obtenu après passage de $y(n)$ dans un système linéaire suivi de l'addition d'un bruit aléatoire stationnaire $bb(n)$.



Le problème qui se pose est comment retrouver $y(n)$ à partir de $x(n)$. Une solution consiste à filtrer $x(n)$ de telle sorte que la sortie $\hat{y}(n)$ soit la plus proche possible de $y(n)$. On peut mesurer la qualité de l'estimation par $e(n)$ défini par :

$$e(n) = y(n) - \hat{y}(n)$$

On cherche donc un filtre qui minimise l'erreur. Il est pratique de chercher à minimiser $e^2(n)$ car c'est une fonction quadratique facilement dérivable. Par ailleurs, étant donné que les signaux sont aléatoires, la fonction coût à minimiser est l'erreur quadratique moyenne (MSE) définie par : $\xi(n) = E(e^2(n))$

Si on suppose que le filtre recherché H est un filtre RIF de longueur N , on peut en calculer les coefficients par résolution d'un système linéaire d'équations. $h = [b_0 \ b_1 \ \dots \ b_{N-1}]^T$

Le signal estimé $\hat{y}(n)$ peut alors s'écrire : $\hat{y}(n) = \sum_{i=0}^{N-1} b_i x(n-i)$

Rappelons que l'on cherche à minimiser $\xi(n) = E\{e^2(n)\} = E\left\{\left(y(n) - \sum_{i=0}^{N-1} b_i x(n-i)\right)^2\right\}$

Pour en obtenir le minimum, il suffit de chercher de dériver et d'annuler la fonction coût par rapport aux variables b_i de la réponse impulsionnelle du filtre. La dérivée de la fonction coût par rapport au $j^{\text{ème}}$ point de la réponse impulsionnelle est donnée par :

$$\frac{\partial \xi}{\partial b_j} = E \left\{ \frac{\partial}{\partial b_j} \{e^2(n)\} \right\} = E \left\{ 2e(n) \frac{\partial e(n)}{\partial b_j} \right\} = E \left\{ 2e(n) \frac{\partial}{\partial b_j} \{-b_j x(n-j)\} \right\}$$

$$\frac{\partial \xi}{\partial b_j} = -E \{ 2e(n) x(n-j) \} = -E \left\{ 2 \left(y(n) - \sum_{i=0}^{N-1} b_i x(n-i) \right) x(n-j) \right\}$$

En faisant l'hypothèse que les signaux $x(n)$ et $y(n)$ sont stationnaires, et en annulant la dérivée, on trouve :

$$R_{yx}(j) = \sum_{i=0}^{N-1} b_i R_{xx}(j-i)$$

Ce qui pour les différentes valeurs de j , nous donne le système d'équations suivantes à résoudre :

$$\begin{bmatrix} R_{xx}(0) & R_{xx}(1) & \dots & R_{xx}(N-1) \\ R_{xx}(1) & R_{xx}(0) & \dots & R_{xx}(N-2) \\ & & \dots & \\ R_{xx}(N-1) & R_{xx}(N-2) & \dots & R_{xx}(0) \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{N-1} \end{bmatrix} = \begin{bmatrix} R_{yx}(0) \\ R_{yx}(1) \\ \vdots \\ R_{yx}(N-1) \end{bmatrix}$$

Il faut noter que l'obtention des coefficients du filtre repose sur la connaissance de la fonction d'autocorrélation du signal d'entrée et de l'intercorrrelation entre les signaux d'entrée et de sortie désirée.

Exemple 1 : On suppose que l'observation $x(n)=y(n)+bb(n)$ et que le bruit additif $bb(n)$ est centré et non corrélé au signal. Simplifions les équations de Wiener-Hopf en conséquences:

$$R_{yx}(k)=E\{y(n) [y(n-k)+bb(n-k)] \} = R_{yy}(k)$$

$$R_{xx}(k)=E\{(y(n)+bb(n))(y(n-k)+bb(n-k))\} = R_{yy}(k)+R_{bb}(k)$$

Soit le système à résoudre suivant :

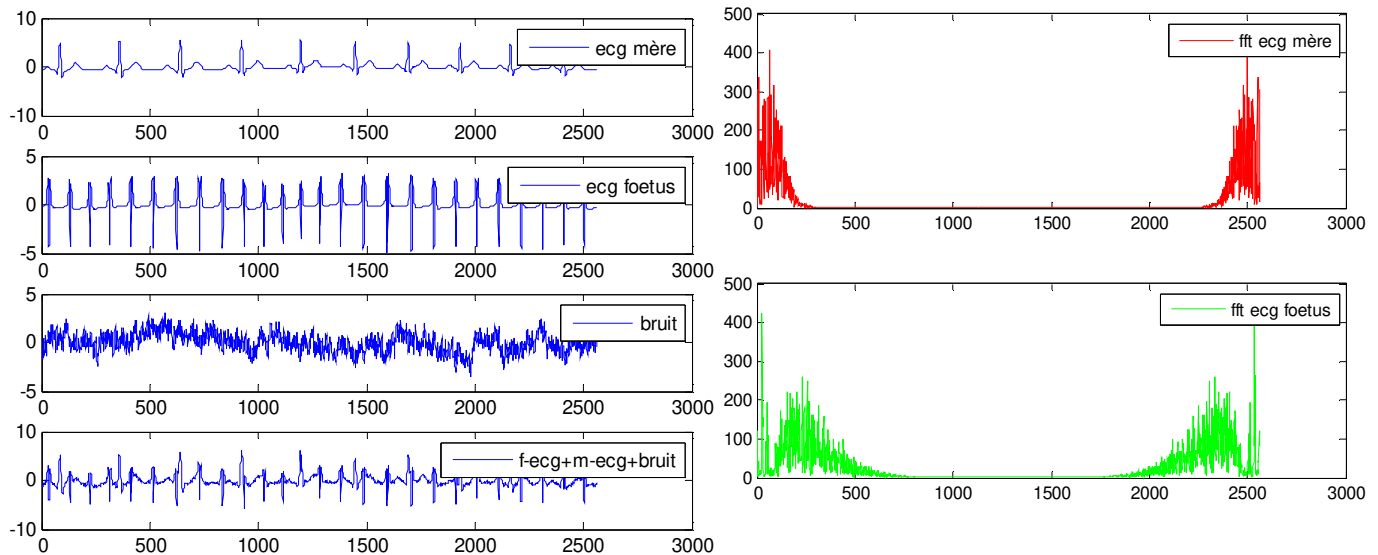
$$\begin{bmatrix} R_{xx}(0) & R_{xx}(1) & \dots & R_{xx}(N-1) \\ R_{xx}(1) & R_{xx}(0) & \dots & R_{xx}(N-2) \\ & & \dots & \\ R_{xx}(N-1) & R_{xx}(N-2) & \dots & R_{xx}(0) \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{N-1} \end{bmatrix} = \begin{bmatrix} R_{xx}(0) - R_{bb}(0) \\ R_{xx}(1) - R_{bb}(1) \\ \vdots \\ R_{xx}(N-1) - R_{bb}(N-1) \end{bmatrix}$$

Exemple 2: On suppose que l'observation $x(n) = y(n) + bb(n)$

Le signal à estimer $y(n)$ a pour la fonction d'autocorrélation $R_y(k)=\alpha^{|k|}$ $0 < \alpha < 1$. Il est décorrélé du bruit blanc $bb(n)$ de variance σ_b^2 . Cherchons $h(n)$ tel que $H(z) = b_0 + b_1 z^{-1}$

$$\begin{bmatrix} 1 + \sigma_b^2 & \alpha \\ \alpha & 1 + \sigma_b^2 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 \\ \alpha \end{bmatrix} \quad H(z) = \frac{1}{(1 + \sigma_b^2) - \alpha^2} [(1 + \sigma_b^2 - \alpha^2) + \alpha \sigma_b^2 z^{-1}]$$

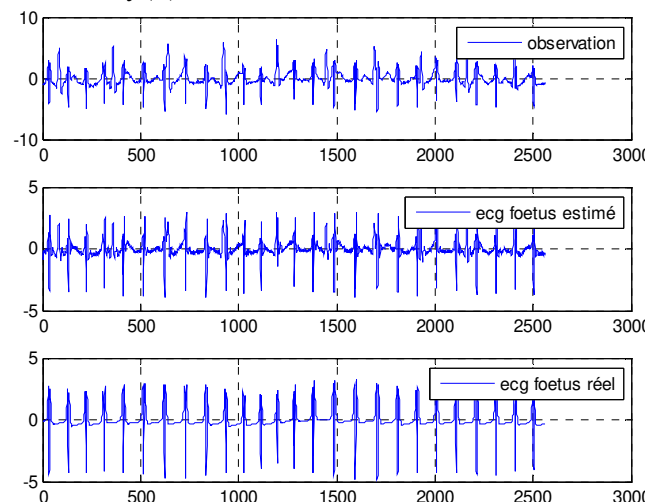
Exemple 3: Comme illustration du filtrage de Wiener, on prend la mesure de l'activité cardiaque d'un fœtus à l'aide d'un électrocardiogramme (ECG) pris au niveau de l'abdomen de la mère (le signal $x(n)$) et qui va naturellement être perturbé par l'ECG de celle-ci auquel se rajoute le bruit thermique des électrodes et des équipements électroniques. Pour retrouver l'ecg du fœtus, on réalise une deuxième mesure fournissant l'ECG de la mère (le signal bb). On peut employer alors le filtre de Wiener pour estimer le signal $y(n)$ représentant l'ECG du fœtus.



On peut observer que les TF des deux signaux ecg (mère et fœtus) occupent la même plage de fréquences. A partir des auto-corrélations de l'ECG mesuré sur l'abdomen $x(n)$ et celui de la mère $bb(n)$, on retrouve les paramètres b_i du filtre qu'on appliquera à $x(n)$ pour obtenir une estimée de $y(n)$ soit l'ECG foetal.

Remarque

Quand les fonctions d'auto et d'intercorrélation ne sont pas connues (cas le plus courant), alors on va approcher le filtre optimal de Wiener en utilisant une boucle de retour et un algorithme de minimisation: c'est ce que l'on appelle **le filtrage adaptatif**. Dans ce cas, on remplacera la connaissance des fonctions de corrélation par une phase d'apprentissage permettant de modifier itérativement la réponse impulsionnelle du filtre.



TD n°2 : Processus aléatoires

1. Soient $X(t)$ et $Y(t)$ deux signaux aléatoires indépendants et stationnaires au sens large (SSL) tels que

$$R_X(\tau) = 9e^{-a|\tau|} \text{ et } R_Y(\tau) = b\delta(\tau) + c. \text{ On considère } Z(t) = X(t) + Y(t),$$

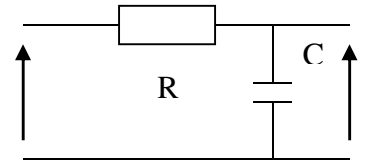
- Quelles conditions doit-on imposer sur les constantes a , b et sur c (Justifier)
- Calculer les statistiques de premier ordre de $Z(t)$
- Calculer $R_Z(t, \tau)$. $Z(t)$ est-il SSL?
- Peut-on calculer sa DSP $S_Z(f)$? si oui calculer la.

2. Soit le processus aléatoire $Z(t) = X\cos(2\pi f_0 t) - Y\sin(2\pi f_0 t)$, où X, Y sont des variables aléatoires gaussiennes indépendantes à moyennes nulles et de variances $\sigma_x^2 = \sigma_y^2 = \sigma^2$; f_0 est une constante.

- Calculer la valeur moyenne $E\{Z(t)\}$ et la variance σ_z^2 .
- Calculer la fonction d'autocorrélation $R_Z(t_1, t_2)$ et examiner si le processus est SSL.

3. L'entrée $x(t)$ du circuit est un bruit blanc de fonction d'autocorrélation $R_x(\tau) = \sigma_x^2 \delta(\tau)$

- Déterminer la densité spectrale de la sortie $y(t)$, notée $S_y(f)$
- Déterminer la fonction d'autocorrélation de la sortie $y(t)$, notée $R_y(\tau)$ ainsi que sa puissance
- Remplacer R par une self L et C par une résistance et reprendre les questions



4. Les signaux $x(n)$ et $y(n)$ ont été obtenus en filtrant, au moyen d'un filtre à réponse impulsionnelle finie, un bruit blanc $b(n)$ gaussien centré de variance σ^2 .

A] L'équation de filtrage est la suivante : $x(n) = 2.b(n) + 0.5.b(n-1) - 0.2.b(n-2) + 0.1.b(n-3)$

- Calculez, en fonction de σ^2 , les coefficients d'autocorrélation d'ordre 0,1,2,3 du signal $x(n)$.
- On notera $R_{xx}(0), R_{xx}(1), R_{xx}(2), R_{xx}(3)$ ces coefficients.
- La répartition des niveaux d'amplitude du signal $x(n)$ est elle gaussienne (sans justifier)?

B] $y(n) = b(n) - b(n-2)$

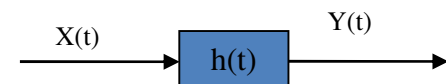
- Donnez la TZ de la réponse impulsionnelle du filtre qui a permis d'obtenir $y(n)$ à partir de $b(n)$.
- Placez les zéros de ce filtre sur un cercle unité, quelles sont la ou les fréquence(s) coupées par ce filtre?
- Tracez approximativement sa réponse en fréquence.

C] On considère maintenant le coefficient d'intercorrélation $R_{xy}(k)$, entre les signaux $x(n)$ et $y(n)$ donné par $R_{xy}(k) = E[x(n)y(n-k)^*]$.

Calculez $R_{xy}(0), R_{xy}(1), R_{xy}(2), R_{xy}(-1), R_{xy}(-2)$

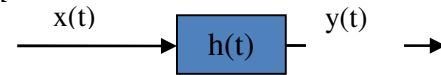
5. Soit $X(t) = A + b(t)$ un signal réel aléatoire, où A est une constante réelle et $b(t)$ est un bruit blanc de densité spectrale de puissance σ_b^2 , et soit un filtre moyennneur de réponse impulsionnelle:

$$h(t) = \frac{1}{T} \Pi(t - T/2)$$



- Exprimer l'autocorrélation statistique du signal d'entrée $R_{xx}(t, \tau)$. $X(t)$ est-il SSL?
- Déterminer la moyenne statistique de $y(t)$.
- Montrer que $R_{yy}(\tau) = \frac{\sigma^2}{T} \Lambda_T(\tau) + A^2$ et en déduire σ_y et μ_y .
- Prendre $A=2$ et tracer $X(t)$ et $Y(t)$.

6. On considère le schéma ci-contre où $x(t)=s(t)+b(t)$ avec $s(t) = \Pi_1(t)$ et $b(t)$ est un bruit blanc Gaussien centré de variance 1



- Déterminer la densité de probabilité de $x(t)$. $x(t)$ est-il SSL ?
- On veut maximiser le rapport signal sur bruit (SNR)
- Déterminer et tracer $h(t)$ pour $k=2$ et $T_0=2$. Calculer alors le SNR.
- Quelle loi de probabilité suit $y(t)$ (justifier).
- Donner un exemple concret de l'utilisation de ce type de filtre
- Si $s(t)$ est déterministe et inconnu, quel filtre utilise-t-on ?

7. On considère la transmission de deux symboles :

$$s_0(t)=A \quad t \in [0, T] \quad , \quad s_1(t)=-A \quad t \in [0, T]$$

à travers un canal à bruit blanc additif Gaussien. Le signal reçu s'écrit : $x(t)=s_i(t)+b(t)$ où $b(t)$ est un bruit blanc centré de DSP σ_b^2

- Déterminer la réponse impulsionnelle du filtre adapté $h(t)$ correspondant à $s_0(t)$ telle que $\int h(t)^2 dt = 1$
- Même question pour $s_1(t)$. Puis, donner le rapport signal sur bruit, dans chaque cas.

On note $y_{si}(t)$, la sortie du filtre correspondant à $s_i(t)$,

- Montrer que pour $b(t)$ elle s'écrit : $y_b(T) = \frac{1}{\sqrt{T}} \int_0^T b(t) dt$

- Déterminer la moyenne et la variance de $y_b(T)$
- Dans l'hypothèse où le bruit blanc est Gaussien, déterminer la loi de probabilité de la variable aléatoire

$$Y = y_{si}(t) + y_b(t)$$

8. On considère un processus de la forme $X(n) = \theta + W(n)$ où $W(n)$ est un processus gaussien de moyenne 0 et de variance 1 tel que $W(n)$ et $W(j)$ sont indépendants si $n \neq j$. On suppose que θ suit une loi $N(0, \sigma^2)$ indépendante de $W(n)$ $n \in \mathbb{Z}$.

- Caractériser le filtre de Wiener permettant d'estimer θ à partir de $X_n, X_{n-1}, \dots, X_{n-N+1}$.
- Calculer les coefficients du filtre.

9. On désire dé-bruiter un signal de parole $z(n)$ corrompu par du bruit additif $b(n)$ indépendant du signal sonore et ce, par filtrage de Wiener. On suppose connues quelques valeurs d'autocorrélation pour les deux signaux telles que :

$$R_{ZZ}[0] = 1.5; R_{ZZ}[1] = 0.5; R_{ZZ}[2] = 0.25; R_{ZZ}[3] = 0.125; R_{ZZ}[4] = 0.0625$$

$$R_{bb}[0] = 1; R_{bb}[1] = 0.25; R_{bb}[2] = 0.0625; R_{bb}[3] = 0.015625$$

- Pourquoi ne peut-on pas employer un filtre adapté ou un filtre moyenneur ?
- Donner les équations de Wiener-Hopf permettant d'estimer $z(n)$.
- Déterminer le filtre de Wiener d'ordre 2 permettant de retrouver le signal utile $\hat{x}(n)$ puis exprimer $H(z)$.

10. On considère un problème d'estimation de bruit $b(n)$.

$$\text{Le signal observé est } x(n) = s(n) + b(n) - b(n-1).$$

On suppose que la corrélation statistique du signal $s(n)$ est $R_{ss}(n) = 0.8^{|n|}$ et qu'il est décorrélé du bruit dont l'autocorrélation est $R_{bb}(n) = 0.8\delta(n)$.

- Déterminer les moyennes statistiques de $s(n)$ et $b(n)$.
- Quand à-t-on recours au filtre de Wiener ?
- Donner les équations de Wiener-Hopf permettant d'estimer $b(n)$
- Déterminer le filtre de Wiener d'ordre 2 permettant de retrouver le signal utile $\hat{b}(n)$.
- Exprimer $\hat{b}(n)$

Solutions

1. Voir interro 1 2016/2017

$$2. \mu_z(t) = 0 \quad \sigma_z^2(t) = \sigma^2 \quad R_z(\tau) = \sigma^2 \cos(2\pi f_0 \tau) \text{ SSL}$$

$$3. R_{xx}(t, \tau) = A^2 + \sigma_b^2 \delta(\tau) = \text{fct}(\tau) \quad \mu_x(t) = A \Rightarrow \text{SSL} \quad H(f) = \text{sinc}(fT) e^{-j\pi f T} \quad \mu_y(t) = A \quad \text{filtre moyennneur}$$

$$4. H(f) = 1/(1 + 2\pi j f RC), \quad S_y(f) = \sigma_x^2 / (1 + 4(\pi f RC)^2) \quad R_y(\tau) = \sigma_x^2 / 2RC e^{-|\tau|/RC}, \quad P = R_y(0) = \sigma_x^2 / 2RC, \text{ interro1 16/17}$$

$$5. R_x(0) = (b_0^2 + b_1^2 + b_2^2 + b_3^2) \cdot \sigma^2 \quad R_x(1) = (b_0 b_1 + b_1 b_2 + b_2 b_3) \cdot \sigma^2 \quad R_x(2) = (b_0 b_2 + b_1 b_3) \cdot \sigma^2 \quad R_x(3) = (b_0 b_3) \cdot \sigma^2 \quad R_x(k \geq 4) = 0$$

$$H(z) = (z^2 - 1)/z^2 \quad R_{xy}(0) = 2.2 \quad R_{xy}(1) = 0.4 \quad R_{xy}(2) = -0.2 \quad R_{xy}(3) = 0.1 \quad R_{xy}(-1) = -0.5 \quad R_{xy}(-2) = -2$$

$$6. x(t) \text{ Gaussien de moyenne } s(t) \text{ et de variance } 1. \quad x(t) \text{ non stationnaire} \quad S_b(f) = |H(f)|^2 \quad S_s(f) = |H(f)|^2 \text{sinc}^2(f) \quad h(t) = \frac{1}{2} \Pi\left(\frac{t}{2} - t\right) \quad \text{SNR} = 1 \quad y(t) \text{ Gaussien} \quad \text{Radar ou Sonar} \quad \text{Passe-bas (moyennneur)}$$

$$7. \text{ Interro1 14/15} \quad 8. R_{xx}(0) = 1 + \sigma^2 \quad R_{xx}(k > 0) = \sigma^2 \quad R_{\theta x}(k) = \sigma^2 \quad b_i = \sigma^2 / (N\sigma^2 + 1) \quad 9. \mu_z = 0, \mu_b = 0, \text{ signal (parole) aléatoire,}$$

$$10. \mu_s = 0, \mu_b = 0, S_b(f) = 0.8. \text{ Quand signal utile et bruit occupent même plage de fréquences. } b_0 = 0.8/2.6 \text{ et } b_1 = 0.$$

Exercices supplémentaires

1. Soit le signal aléatoire défini par $x(t) = A_1 e^{j2\pi f_1 t} + A_2 e^{j2\pi f_2 t}$ où A_1 et A_2 sont deux v.a gaussiennes, décorrelées, centrées et de variance σ^2

- Donner la d.d.p conjointe $f_{A_1, A_2}(A_1, A_2)$
- Déterminer la moyenne statistique du signal $x(t)$ ainsi que sa fonction d'autocorrélation et la D.S.P
- Déterminer la d.d.p de $x(t)$
- Le processus $x(t)$ est-il stationnaire au 2^{ème} ordre ?

2. Le signal aléatoire $x(t)$ réel SSL possède une fonction d'autocorrélation de la forme $R_x(\tau) = \sigma_x^2 e^{-\beta|\tau|}$. Un autre signal est lié à $x(t)$ par l'équation déterministe suivante : $y(t) = ax(t) + b$, où a et b sont des constantes données.

- Quelle est la fonction d'autocorrélation de $y(t)$? En déduire μ_y et σ_y^2
- Quelle est la fonction d'intercorrélacion et de covariance de $x(t)$ et $y(t)$?
- Calculer le coefficient de corrélation. Ce résultat était-il prévisible, pourquoi ?

3. Parmi les fonctions d'auto-corrélation statistiques suivantes lesquelles peuvent être celles d'un processus aléatoire réel?

$R_{x1}(\tau) = \Lambda_2(\tau) - 2$	$R_{x2}(\tau) = -\Lambda_2(\tau) + 2$	$R_{x3}(\tau) = e^{-2 \tau }$	$R_{x4}(\tau) = e^{-2\tau} U(\tau)$	$R_{x5}(\tau) = \delta(\tau-1) + \delta(\tau+1)$
--------------------------------------	---------------------------------------	-------------------------------	-------------------------------------	--

4. On considère le signal $x(n) = g(n) \cos(2\pi k_0 n / N + \phi)$ défini pour $n \in [0, N-1]$, et où $g(n)$ est une fonction aléatoire SSL, indépendante de ϕ v.a uniforme ente 0 et 2π .

- Calculez l'autocorrélation de $x(n)$
- Déduisez en sa densité spectrale de puissance, en fonction de la DSP de g , $S_g(f)$.

5. Soit le signal aléatoire $x(t)$ SSL dont la DSP est donnée par $S_x(f) = \sigma^2 B \Pi_B(f)$

- Déterminer l'autocorrélation statistique de $x(t)$
- En déduire la moyenne et la variance statistique de $x(t)$.

Ce signal est transmis à travers un SLIT dont la fonction de transfert $H(f) = \Pi_A(f)$ avec $A < B$

- Le signal de sortie est-il aléatoire? (Justifier) SSL? (Justifier)
- Déterminer $S_y(f)$ et en déduire $R_y(\tau)$ puis les moments statistiques d'ordre 1 de $y(t)$.

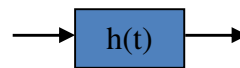
6. On dispose d'un signal reçu qui est la version bruitée, retardée et atténuée d'un signal d'intérêt $s(n)$. Le bruit $b(n)$ est supposé blanc gaussien de variance σ^2 . Le problème est de déterminer l'amplitude A et le retard n_0 dans le signal reçu $x(n) = A.s(n - n_0) + b(n)$. Sachant que le rapport signal-à-bruit est maximum en sortie du filtre adapté de réponse $h(n) = s(-n)$.

- Vérifier que la sortie $y(n)$ de ce filtre s'exprime comme la somme de deux fonctions de corrélation.
- Calculer la variance σ_b^2 du bruit de sortie.
- On prend pour $s(n)$ une impulsion rectangulaire de largeur L , tracer alors un exemple de signal reçu et de sortie du filtre adapté.

7. Soit $x(t) = t$ avec $0 \leq t \leq T$

On utilise ce signal pour déterminer la distance d'un objet. Sachant que le signal reçu $y(t)$ par le récepteur est retardé de T' et bruité par un bruit blanc Gaussien de densité spectrale de puissance σ^2 :

1. Tracer approximativement $y(t)$.
2. Déterminer l'expression générale de la réponse impulsionnelle du filtre $h(t)$ (dont l'énergie vaut 1) permettant de maximiser le rapport signal sur bruit.
3. Tracer $h(t)$ en fonction de $T_0 = 2T$ et $k = \sigma^2$, puis $z(t)$ la sortie du filtre (prendre $T' = 10$).
4. Donner le rapport signal sur bruit après filtrage.
5. Pourquoi ce filtrage est dit, en général, optimal et, en particulier, adapté?



8. On considère un processus de la forme $X(n) = b(n) + \alpha b(n-1) + W(n)$ où $W(n)$ est un processus gaussien de moyenne 0 et de variance σ^2 tel que $W(n)$ et $W(j)$ sont indépendants si $n \neq j$. On suppose que $b(n)$ est une variable aléatoire uniforme à valeurs dans $\{-1, 1\}$, indépendante de $W(n)$ $n \in \mathbb{Z}$ et que de même, $b(n)$ et $b(j)$ sont indépendants si $n \neq j$ ($P(b(n) = 1) = P(b(n) = -1) = 1/2$).

- Construire un filtre d'ordre 3 qui estime $b(n)$.

9. On considère un problème d'estimation d'un signal $s(n)$ bruité..

Le signal observé est $x(n) = \alpha s(n) + b(n)$.

On suppose que $s(n)$ et $b(n)$ sont SSL et décorréles et que le bruit possède une DSP $S_{bb}(f) = 0.25$.

- Rappeler les conditions d'applications du filtrage de Wiener
- Déterminer le filtre de Wiener d'ordre 2 permettant de retrouver le signal utile $\hat{s}(n)$. On supposera que $R_{ss}(k) = 2 \cdot 0.5^{|k|}$
- Exprimer $H(z)$ puis $\hat{s}(n)$

10. On considère un problème d'estimation d'un signal $s(n)$ bruité et ayant subi un écho.

Le signal observé est $x(n) = s(n) + 0.5s(n-1) + b(n)$.

On suppose connue l'autocorrélation du signal utile et qu'elle a pour expression $0.5^{|k|}$ et l'on suppose que le signal utile est décorrélé du bruit dont l'autocorrélation est $R_{bb}(k) = 0.25^{|k|}$.

1. Déterminer les moyennes statistiques de $x(n)$ et $b(n)$
2. Donner les équations de Wiener-Hopf permettant d'estimer $s(n)$
3. Déterminer le filtre de Wiener d'ordre 2 permettant de retrouver le signal utile $\hat{s}(n)$.
4. Exprimer $\hat{s}(n)$ et commenter

Solutions

1. Voir interro 1 2014/2015 2. $R_y(\tau) = \sigma_x^2 e^{-\beta|\tau|} + b^2$ $S_y(f) = 2\beta\sigma_x^2 / (\beta^2 + 4(\pi f)^2) + b^2\delta(f)$ $R_{xy}(\tau) = a R_x(\tau)$
3. Voir interro 1 2015/2016 4. $R_x(m) = R_g(m) \cos(2\pi k_0 m / N) / 2$ $S_x(f) = [S_g(f - k_0) + S_g(f + k_0)] / 4$
5. $R_x(\tau) = \sigma^2 B^2 \text{sinc}(B\tau)$, $\mu_x = 0$, $\sigma_x^2 = \sigma^2 B^2$, $x(t) \text{ SSL} + h(t) \text{ SLIT} \Rightarrow y(t) \text{ SSL}$, $S_y(f) = \sigma^2 B \Pi_A(f)$, $R_x(\tau) = \sigma^2 A B \text{sinc}(A\tau)$
6. $y(n) = A R_s(n - n_0) + R_{bs}(n)$ $\sigma_b^2 = R_b(0)$ $S_{b'}(f) = \sigma_b^2 |H(f)|^2 \Rightarrow R_{b'}(0) = \sigma_b^2 \int_{-1/2}^{1/2} |H(f)|^2 df$
7. Examen 15/16 8. $R_{xx}(1:3) = (1 + \alpha^2 + \sigma^2, \alpha, 0)$ $R_{bx}(1:3) = (1, 0, 0)$
9. Signaux : utile et observation stationnaires et conjointement stationnaires connus
 $R_{xx}(k) = \alpha^2 R_{ss}(k) + R_{bb}(k)$ $R_{sx}(k) = \alpha R_{ss}(k)$

TP n° 3 : Processus aléatoires et estimation spectrale

Buts : Manipuler des signaux aléatoires, les caractériser grâce à leurs moments d'ordre 1 (moyenne, variance) et d'ordre 2 (autocorrélation, covariance). Aborder et acquérir les notions de stationnarité et d'ergodicité. Estimation spectrale. Aborder la notion de Filtre Formeur.

1. Démo

```
# -*- coding: utf-8 -*-
import numpy as np; import matplotlib.pyplot as plt;
import scipy.signal as sp
N_R=500; N_va=1000; A=1; f0=.002; X=[];

t = np.linspace(0,N_va-1,N_va)
phi=np.random.uniform(0,2*np.pi,N_R);

for i in range (N_R):
    X=np.append(X,A*np.cos(2*np.pi*f0*t+phi[i]))
X= np.resize(X,new_shape=(N_R,N_va))

plt.figure(1);plt.plot(t,X[0,:]);plt.plot(t,X[1,:]);plt.plot(t,X[2,:]);plt.plot(t,X[3,:]);
"""Stationnarité d'ordre 1"""
moy=np.mean(X,0); var=np.var(X,0);
plt.figure(2); plt.subplot(211); plt.plot(moy); plt.subplot(212); plt.plot(var);
"""Stationnarité d'ordre 2"""
Cx=np.cov(X, rowvar=False);
plt.figure(3); plt.imshow(Cx);
plt.figure(4); plt.plot(Cx[0,:]); plt.plot(Cx[20,:]);plt.plot(Cx[40,:]);
"""Ergodisme d'ordre 1"""
X1=X[1,:];
moy_t=np.mean(X1); var_t=np.var(X1);
"""Ergodisme d'ordre 2"""
Cx_t=np.correlate(X1-moy_t,X1-moy_t,'full')/N_va
Rx_t=np.correlate(X1,X1,'full')/N_va
plt.figure(5)
plt.plot(Cx_t,label='Cov temp'); plt.plot(Rx_t,label='Cor temp');
plt.legend()

# A = np.random.uniform(-1,1,N_R);
# for i in range (N_R):
#     X=np.append(X,A[i]*np.cos(2*np.pi*f0*t))

#X=np.random.randn(N_R,N_va)

# mux=0; varx=1;
# X=mux+np.sqrt(varx)*np.random.randn(N_R,N_va)

""" 1 réalisation d'un bruit blanc Gaussien """
""" Hypothèse ergodique + stationnaire """
mux=0; varx=1; N=1000;
bb=mux+np.sqrt(varx)*np.random.randn(N)
fe=22050; Te=1/fe;
t = np.linspace(0, N-1, N)*Te;
Moy_s=Moy_t=np.mean(bb);
Var_s=Var_t=np.var(bb);
Rb_s=Rb_t = np.correlate(bb,bb,mode='same')/N;
tt = np.linspace(1-N/2, N/2-1, N)*Te;
plt.figure(6);plt.subplot(311); plt.plot(t,bb); plt.grid(True);
plt.subplot(312); plt.plot(tt,Rb_s); plt.grid(True);
plt.xlabel('temps'); plt.ylabel('Amp'); plt.title('Autocor statistique du bruit blanc');
```

```
NF=2048;
TFb = np.fft.fft(bb,NF); TFb=np.fft.fftshift(TFb);
Sbf = np.abs(TFb)**2/N; ff = np.linspace(-NF/2, NF/2-1, NF)*fe/NF;
s=np.mean(Sbf)
plt.subplot(313); plt.plot(ff,Sbf); plt.grid(True);plt.title("DSP du bruit blanc");

""" Calcul de la DSP : Estimation spectrale"""
f0=5000; X = bb + 0.5*np.cos(2*np.pi*f0*t)
#f1=5050; X = X + 0.5*np.cos(2*np.pi*f1*t)
NF=2048;
TFx1 = np.fft.fft(X,NF)/N; TFx1=np.fft.fftshift(TFx1);
Sxf1 = np.abs(TFx1)**2; ff = np.linspace(-NF/2, NF/2-1, NF)*fe/NF;
Rx_s=Rx_t = np.correlate(X,X,mode='same')/N; Sxf2 = np.fft.fft(Rx_s,NF)/N;
Sxf2=np.fft.fftshift(Sxf2); Sxf2 = np.abs(Sxf2)
ff3,Sxf3 =
sp.periodogram(X,fs=fe>window='boxcar',nfft=NF,return_onesided=False,scaling='spectrum')
ff4,Sxf4 =
sp.periodogram(X,fs=fe>window='hann',nfft=NF,return_onesided=False,scaling='spectrum')
ff5, Sxf5 = sp.welch(X,fs=fe>window='hann',nperseg=N//4,
nfft=NF,return_onesided=False,scaling='spectrum')
var_ss=np.mean(Sxf1)
ff = np.linspace(-NF/2, NF/2-1, NF)*fe/NF;
plt.figure(7)
plt.subplot(221);plt.plot(ff,Sxf1,ff3,Sxf3); plt.title('DSP=Périodogramme Fenêtre
Rectangulaire');
plt.subplot(222);plt.plot(ff,Sxf2); plt.title('Correlogramme');
plt.subplot(223);plt.plot(ff4,Sxf4); plt.title('Périodogramme Fenêtre Hanning');
plt.subplot(224);plt.plot(ff5,Sxf5); plt.title('Périodogramme Moyennée Fenêtre Hanning');

""" Notion de filtre Formeur """
a = np.array([1,-0.839,-0.015,-0.320,0.197,0.055,-0.285,0.067,0.044,0.003,0.178])
b = np.array([1,0]);
L = 500; f,H= sp.freqz(b,a,L,fs=fe);
Y = sp.lfilter(b,a,bb) ;
TFb = TFb[0:L];
plt.figure(8); plt.subplot(211);plt.plot(f, 20*np.log10(abs(TFb/TFb.max())));
plt.title('Signal d entree : Bruit Blanc'); plt.xlabel('Fréq (Hz)');plt.ylabel('Amp');
plt.subplot(212);plt.plot(f, 20*np.log10(abs(H/H.max())));
plt.title('Module du Filtre (bleu)+ Sortie du filtre (orange)'); plt.xlabel('Fréq
(Hz)');plt.ylabel('Amp');
TFy = np.fft.fft(Y); TFy = TFy[0:L];
plt.plot(f, 20*np.log10(abs(TFy/TFy.max())));
```

2. A faire

Télécharger le fichier 'vous avez du courrier en attente.wav' et le placer dans le même répertoire que votre programme commençant comme suit :

```
import numpy as np; import matplotlib.pyplot as plt;
from scipy.io import wavfile as wf; import winsound ;
fname = 'vousavezducourrierenattente.wav';
winsound.PlaySound(fname, winsound.SND_FILENAME)
fe, X = wf.read(fname);
Te=1/fe; N=len(X); t = np.linspace(0, N-1, N)*Te;
plt.figure(1);plt.subplot(211); plt.plot(t,X); plt.grid(True);
plt.xlabel('temps'); plt.ylabel('Amp'); plt.title('Phrase');
N1=21000;N2=21500; Y=X[N1:N2] ; ty=np.linspace(N1, N2-1, N2-N1)*Te;
plt.subplot(212); plt.plot(ty,Y); plt.grid(True);
plt.xlabel('temps'); plt.ylabel('Amp'); plt.title('morceau de la phrase');
#zz=np.int8(Y) ; wf.write("z.wav", fe, zz);
```

```
#winsound.PlaySound("z.wav",winsound.SND_FILENAME) ;
```

1. Le signal étudié est un processus aléatoire ou une réalisation d'un signal aléatoire?
2. Calculer la DSP. Est-elle d'origine statistique ou temporelle?
3. Calculer et visualiser la moyenne et la variance temporelles pour différentes parties du signal qui se chevauchent pour des durées de $N=1000, 500, 250, 125$. Pour quelle valeur de N peut-on considérer le processus stationnaire?
1. Comparer les 4 techniques en faisant varier $A1, A2$ et $A3$ et commenter.
2. Rapprocher les fréquences et observer.
3. Commenter les quatre techniques d'estimation de la DSP en faisant varier N .
4. Quelle est l'avantage et l'inconvénient du moyennage?

TP n° 4 : Filtrage adapté et Filtrage de Wiener

Buts : Filtrage optimal

- Détection d'un signal déterministe connu noyé dans du bruit blanc → Filtrage adapté
- Filtrage d'un signal aléatoire SSL noyé dans un bruit SSL → Filtrage de Wiener

1. Démo

```
# -*- coding: utf-8 -*-
import numpy as np; import matplotlib.pyplot as plt
" Filtrage adapté "
N = 500; var = 0.16 ; mu = 0;
x=np.ones(10); Retard = 100
y = np.sqrt(var) * np.random.randn(N) + mu
y[Retard:Retard+len(x)]= y[Retard:Retard+len(x)] + x
#Retard=300; y[Retard:Retard+len(x)]= y[Retard:Retard+len(x)] + x
Ryx = np.correlate(y,x,"full"); Ryx=Ryx[len(x)-1:]
plt.figure(1)
plt.subplot(311); plt.plot(x); plt.title('signal émis')
plt.subplot(312); plt.plot(y); plt.title('signal reçu')
plt.subplot(313); plt.plot(Ryx); plt.title('Intercorrélation entre signal émis et signal reçu')
import scipy.linalg as la; import scipy.signal as sp;
fecg=[]; mecg=[];

""" Lecture des fichiers """
with open('fecg.txt') as f:
    for i in f:
        fecg.append(float(i))
fecg = np.array(fecg)
with open('mecg.txt') as f:
    for i in f :
        mecg.append(float(i))
mecg = np.array(mecg)
""" Rajout du bruit """
N=len(mecg)
var = 0.01; bb=(var**0.5)*np.random.randn(N); obs=fecg+mecg+bb;
"""Détermination des coefficients du Filtre de Wiener"""
P=10;
obs=obs-np.mean(obs); mecg=mecg-np.mean(mecg);
Rxx = np.correlate(obs,obs,mode='full')[len(obs)-1:];
Rmecg = np.correlate(mecg,mecg,mode='full')[len(mecg)-1:];
Ryx = Rxx-Rmecg;
C = la.toeplitz(Rxx[0:P]);
B = Ryx[0:P];
b = la.inv(C).dot(B)
"""Filtrage"""
a = np.array([1,0]);
y = sp.lfilter(b,a,obs) ;
plt.figure(2);
plt.subplot(311); plt.plot(fecg); plt.grid(True); plt.title('ecg bébé');
plt.subplot(312); plt.plot(obs); plt.grid(True);plt.title('Observation');
plt.subplot(313); plt.plot(fecg); plt.plot(y); plt.grid(True);plt.title('ecg bébé estimé');
```

2. A faire

FGE, USTHB [assiakourgli@gmail.com / <http://perso.usthb.dz/~akourgli/>

"A faire filtrage adapté"

1. Créer un signal composé d'une suite aléatoire de symboles A et -A d'une durée de 10 chacun séparé de 20s

```
import numpy as np
import matplotlib.pyplot as plt
W = np.random.randint(0,2,10); A = 5;
W = 2*W-1
W = A*W
x = []; pause = np.zeros(20)
for i in W:
```

```
    symb = i*np.ones(10)
    x = np.append(x,symb)
    x = np.append (x, pause)
```

2. Rajouter bruit blanc

3. Utiliser filtre adapté

4. Visualiser les 3 signaux

"A faire Filtrage de Wiener"

```
import numpy as np; import scipy.signal as sp; import matplotlib.pyplot as plt
import sounddevice as snd
import scipy.io.wavfile as wav
import scipy.linalg as la; import scipy.signal as sp;
fe,x = wav.read('vousavezducourrierenattente.wav')
```

1. Créer un signal composé du signal audio + du bruit blanc

2. Ecouter pour vérifier la présence de l'écho

3. Déterminer les coefficients du filtre de Wiener

4. Filtrer

5. Ecouter signal filtré

6. Visualiser les 3 signaux: signal audio original, avec écho et celui filtré

7. Visualiser les 3 périodogramme des signaux: signal audio original, bruité et celui filtré sur une petite portion

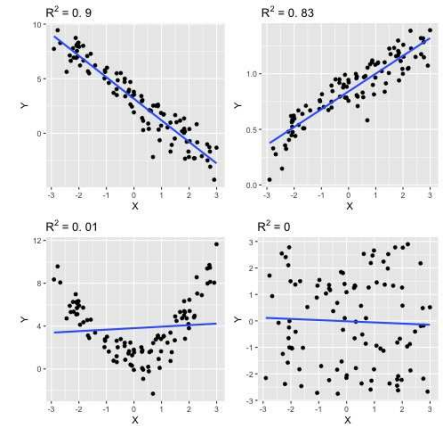
III. Analyse spectrale paramétrique et filtrage numérique adaptatif

Introduction

La modélisation paramétrique consiste à associer à un signal un modèle, représenté par un vecteur dit paramètre $\theta = [\theta_1, \theta_2, \dots, \theta_p]$ censé représenter au mieux le signal considéré. Elle nécessite le choix d'un modèle qui ne peut se faire que si l'on possède des informations sur le signal.

Elle permet entre autres : de représenter les données par un vecteur de plus petite dimension. Ce qui pourra servir lors de la compression et transmission, comme signature pour une identification ou classification.

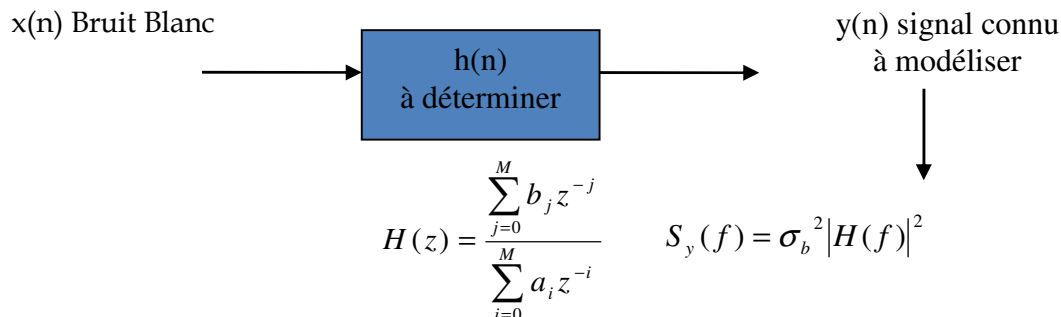
Nous aborderons dans ce cours le cas de la modélisation spectrale qui nous permettra d'estimer le spectre du signal.



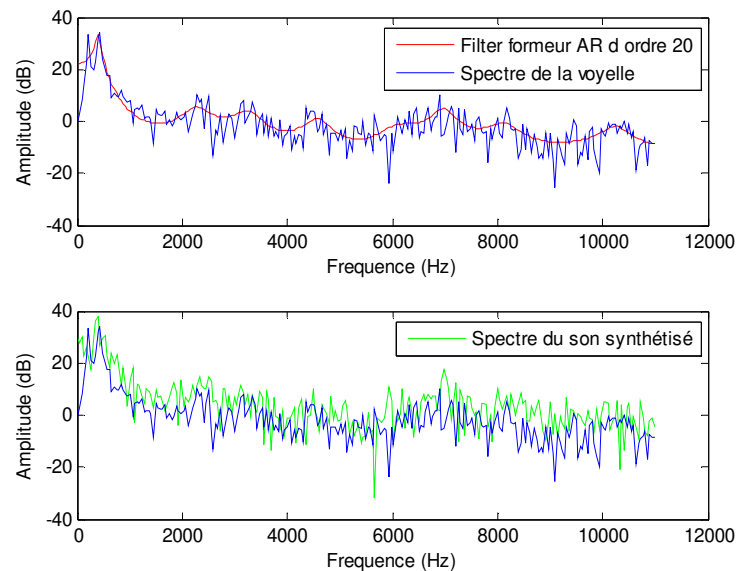
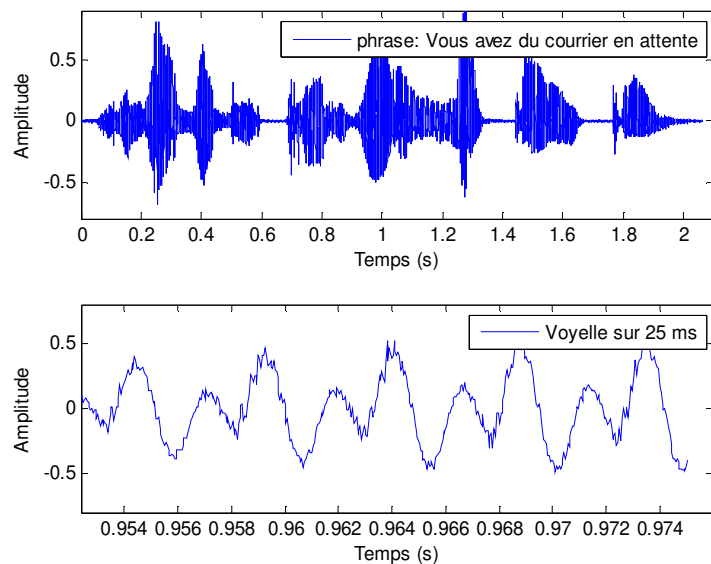
1. Modélisation spectrale paramétrique

Nous avons vu précédemment qu'un signal aléatoire peut être modélisé (synthétisé) comme la réponse d'un filtre linéaire à une excitation sous forme de bruit blanc tel que $|H(f)|^2 = S_x(f) / \sigma_b^2$. Ce filtre formeur $H(f)$ est aussi dit processus générateur. Ces paramètres associés à la variance du bruit σ_b^2 constituent le modèle mathématique correspondant au signal aléatoire. Le concept de processus générateur de signal a été particulièrement développé et appliqué avec des filtres numériques. Ce sont les modèles de signaux les plus utilisés en traitement statistique du signal (estimation, prédiction...). Selon la nature du filtre, on peut obtenir différents modèles de signaux (AR, MA, ARMA, etc.)

Exemple : Lors d'un appel par GSM (téléphone portable), le portable qui fait office d'un micro-ordinateur regroupant différentes fonctionnalités dont l'analyse, la synthèse, le codage, etc. va nous permettre de modéliser la parole (aléatoire) en opérant un codage LPC par tranches de dizaines de ms. Elle consiste à retrouver les paramètres du filtre formeur $h(n)$ pour chaque tranche $y(n)$ enregistrée et analysée. Ce sont ces paramètres (a_i et b_j) qui seront transmis pour produire un signal de synthèse approchant le signal original $y(n)$.



Ci-dessous la phrase "vous avez du courrier en attente" échantillonnée à une fréquence $f_e = 22050$ (45531 échantillons) et un zoom sur une voyelle de durée 25 ms (500 échantillons).



En comparant l'allure du filtre formeur $H(f)$ à celle de $S_y(f)$, on note que l'on retrouve l'allure générale du spectre de la voyelle, notamment les fréquences dont la puissance est maximale. Sachant que le spectre de la voyelle comporte 500 valeurs et que le filtre $H(f)$ équivalent est obtenu à partir de 20 coefficients, il vaut mieux transmettre 21 coefficients ($20 a_i$ + variance du bruit) que 500 valeurs.

C'est ainsi que les modèles autorégressifs sont d'un emploi de plus en plus répandu en traitement du signal : codage et transmission par prédiction linéaire, synthèse de parole, reconnaissance, etc.

Remarque : Le signal de parole est un processus aléatoire non-stationnaire à long terme, mais il est considéré comme stationnaire dans des fenêtres temporelles d'analyse de l'ordre de 20 à 30ms. Cette propriété de stationnarité à court terme permet donc une analyse et modélisation progressive du signal de parole. Pour éviter toutes pertes d'information, on veillera à prendre des fenêtres chevauchantes.

2. Modèle auto-régressif (AR)

Les signaux autorégressifs sont obtenus par passage d'un bruit blanc dans un filtre purement récursif. Ce filtre est donc de réponse impulsionnelle infinie.

$$H(z) = 1 / \left(1 + \sum_{i=1}^N a_i z^{-i} \right)$$

A partir de $H(z)$, on peut déterminer l'équation aux différences : $y(n) = x(n) - \sum_{i=1}^N a_i y(n-i)$

Cela signifie que le signal $y(n)$ est supposé être prédictible en fonction d'un certain nombre de ses valeurs antérieures.

Sachant que $x(n)$ est un bruit blanc alors d'où $R_{xx}(k) = \sigma^2 \delta(k)$

$$- \mu_x = E\{x(n)\} = 0 \quad - R_{xx}(0) = E\{x(n)^2\} = \sigma^2 \quad - R_{xx}(k) = E\{x(n)x(n-k)\} = 0 \text{ pour } k \neq 0$$

Calculons alors $R_{yy}(k)$

$$\begin{aligned} R_{yy}(k) &= E\{y(n)y(n-k)\} = E\{x(n)y(n-k)\} - \sum_{i=1}^N a_i E\{y(n-i)y(n-k)\} \\ &= E\left\{x(n) \sum_{k'} x(n-k-k') h(k')\right\} - \sum_{i=1}^N a_i R_{yy}(k-i) = \sum_{k'} E\{x(n)x(n-k-k')\} h(k') - \sum_{i=1}^N a_i R_{yy}(k-i) \\ &= \sum_{k'} R_{xx}(k+k') h(k') - \sum_{i=1}^N a_i R_{yy}(k-i) = R_{xx}(k)h(0) + R_{xx}(k+1)h(1) + \dots - \sum_{i=1}^N a_i R_{yy}(k-i) \end{aligned}$$

- Si $k=0$, $R_{yy}(0) = R_{xx}(0)h(0) - \sum_{i=1}^N a_i R_{yy}(-i) = \sigma^2 \cdot 1 - \sum_{i=1}^N a_i R_{yy}(i)$

- Pour $k=1$ à N , $R_{yy}(k) = 0 - \sum_{i=1}^N a_i R_{yy}(k-i) = -\sum_{i=1}^N a_i R_{yy}(k-i)$

On peut utiliser une forme matricielle : $R \cdot \underline{a} = \underline{S}$ (Rappelons que pour un signal réel $R_{yy}(k) = R_{yy}(-k)$)

$$\begin{bmatrix} R_{yy}(0) & R_{yy}(1) & \dots & R_{yy}(N) \\ R_{yy}(1) & R_{yy}(0) & \dots & R_{yy}(N-1) \\ & & \dots & \\ R_{yy}(N) & R_{yy}(N-1) & \dots & R_{yy}(0) \end{bmatrix} \cdot \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

R est la matrice d'autocorrélation dont le terme général r_{ij} ne dépend que de la différence $i-j$ (Matrice de Toeplitz). La résolution de ces équations dites de Yule-Walker permet de connaître les paramètres du filtre et la variance du bruit blanc.

On peut reformuler cette matrice sous la forme suivante (On l'emploiera en TP) :

$$\overbrace{\begin{bmatrix} R_{yy}(0) & R_{yy}(1) & \dots & R_{yy}(P-1) \\ R_{yy}(1) & R_{yy}(0) & \dots & R_{yy}(P-2) \\ & & \dots & \\ R_{yy}(P-1) & R_{yy}(P-2) & \dots & R_{yy}(0) \end{bmatrix}}^C \cdot \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_P \end{bmatrix} = \overbrace{\begin{bmatrix} -R_{yy}(1) \\ -R_{yy}(2) \\ \vdots \\ -R_{yy}(P) \end{bmatrix}}^B$$

$$\sigma^2 = R_{yy}(0) + \sum_{i=1}^P a_i R_{yy}(i)$$

Remarques : Nous ne disposons pas d'un processus aléatoire mais d'une seule réalisation soit $y(n)$, il n'est pas donc pas possible de calculer l'auto-corrélation statistique $R_{yy}(k)$. Cette dernière sera remplacée par l'autocorrélation temporelle en faisant l'hypothèse que le processus est ergodique (voir exo 1 du TP n°4).

Il existe divers algorithmes (Burg, Levinson) qui permettent d'estimer assez rapidement les a_i et σ sans passer par l'inversion matricielle. Tout comme il est possible de déterminer l'ordre N adéquat (critère AIC).

Exemple d'application

1. On considère le modèle auto-régressif (AR) d'ordre 1 tel que : $x(n) = -a_1 x(n-1) + b(n)$

- Déterminer les équations de Yule-Walker pour ce modèle
- En supposant que $x(n)$ est connu, déterminer les paramètres du modèle.
- Déterminer les $R_x(k)$ (les a_i sont supposés connus)

Réponses :

- $(a_1 = -R_x(1)/R_x(0) \quad \sigma^2 = R_x(0)(1-a_1^2))$
- $(R_x(0) = \sigma^2/(1-a_1^2) \quad R_x(1) = -a_1 \sigma^2/(1-a_1^2) \quad R_x(k) = (-a_1)^k \sigma^2/(1-a_1^2))$

2. Refaire le même travail pour un modèle d'ordre 2 tel que : $x(n) = -a_1 x(n-1) - a_2 x(n-2) + b(n)$

Réponses :

- $a_1 = R_x(1)[R_x(2) - R_x(0)]/[R_x(0)^2 - R_x(1)^2] \quad a_2 = [R_x(1)^2 - R_x(0)R_x(2)]/[R_x(0)^2 - R_x(1)^2]$
- $\sigma^2 = R_x(0) + R_x(1)^2[R_x(2) - R_x(0)]/[R_x(0)^2 - R_x(1)^2] + R_x(2)[R_x(1)^2 - R_x(0)R_x(2)]/[R_x(0)^2 - R_x(1)^2]$
- $R_x(1) = -a_1 R_x(0)/(a_2 + 1) \quad R_x(2) = (-a_2 + a_1^2/(1+a_2))R_x(0) \quad R_x(0) = (1+a_2)\sigma^2/(1+a_2 - a_1^2 - a_1^2 - a_2^3 + a_2 a_1^2)$

-Algorithme de Levinson

C'est un algorithme qui permet de résoudre tout système du type $Ax = b$ avec A Toeplitz, donc en particulier les équations normales de Yule-Walker :

$$\begin{bmatrix} R_{yy}(0) & R_{yy}(1) & \dots & R_{yy}(N) \\ R_{yy}(1) & R_{yy}(0) & \dots & R_{yy}(N-1) \\ & & \dots & \\ R_{yy}(N) & R_{yy}(N-1) & \dots & R_{yy}(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Le principe de l'algorithme de Levinson est de trouver de façon récursive les paramètres d'ordre k en fonction des paramètres d'ordre $k-1$.

- Initialisation : $a_1(1) = k(1) = -\frac{R_{yy}(1)}{R_{yy}(0)}$

$$\sigma_1^2 = (1 - |a_1|^2)R_{yy}(0)$$

- Récursion : pour k allant de 2 à P (ordre du modèle AR)

$$a_k(k) = -\frac{R_{yy}(k) + \sum_{i=1}^{k-1} a_{k-1}(i)R_{yy}(k-i)}{\sigma_{k-1}^2} = k(k)$$

$$a_k(i) = a_{k-1}(i) + a_k(k)a_{k-1}(k-i) \quad \text{pour } i = 1, \dots, k-1$$

$$\sigma_k^2 = (1 - |a_k(k)|^2)\sigma_{k-1}^2$$

3. Modèle à moyenne ajustée (MA)

Les signaux à moyenne mobile sont obtenus par passage d'un bruit blanc dans un filtre purement transverse.

Ce filtre est aussi appelé filtre à réponse impulsionnelle finie : $H(z) = \sum_{i=0}^M b_i z^{-i}$

Le signal $y(n)$ est supposé pouvoir s'écrire comme une combinaison linéaire d'échantillons décorrélés entre eux, ce qui peut se formaliser comme une combinaison linéaire d'échantillons d'un bruit blanc $x(n)$.

$$\text{On a donc : } y(n) = \sum_{i=0}^M b_i x(n-i)$$

$$\text{et } \mu_y = E\{y(n)\} = \sum_{i=0}^M b_i \mu_x = \mu_x \sum_{i=0}^M b_i = \mu_x \cdot H_f(0)$$

On cherche les paramètres du filtre qui génèrent $y(t)$ à partir de $x(t)$, bruit blanc centré :

$$R_{yy}(k) = E\{y(n)y(n-k)\} = E\left\{\sum_{i=0}^M b_i x(n-i) \cdot \sum_{j=0}^M b_j x(n-j-k)\right\}.$$

$$R_{yy}(k) = \sum_{i=0}^M b_i \cdot \sum_{j=0}^M b_j E\{x(n-i) \cdot x(n-j-k)\} = \sum_{i=0}^M b_i \cdot \sum_{j=0}^M b_j R_{xx}(j+k-i)$$

- Si $j+k \neq i \Rightarrow R_{yy}(k) = 0$
- Sinon $\Rightarrow R_{yy}(k) = \sigma^2 \sum_{j=0}^{M-k} b_{j+k} \cdot b_j$

Le problème est non linéaire en fonction des coefficients, il faut un algorithme de programmation non linéaire pour obtenir b_i à partir des $R_{yy}(k)$. Cependant, l'algorithme de Durbin permet d'approcher la solution optimale avec de bons résultats. Le principe de cet algorithme consiste à identifier le modèle MA d'ordre M avec un modèle AR d'ordre $N \gg M$ (TP n°4). En effet, tout modèle MA peut être identifié à un modèle AR

d'ordre infini: $\sum_{i=0}^M b_i z^{-i} = 1 / \sum_{i=0}^{\infty} a_i z^{-i}$

Exemple

On considère le modèle à moyenne ajustée (MA) :

A. d'ordre 1 tel que $x(n) = e(n) + b_1 \cdot e(n-1)$

- Calculer μ_x .
- En supposant que $x(n)$ est connu, déterminer les paramètres du modèle.
- Connaissant les paramètres du modèle, déterminer les $R_x(k)$

B. d'ordre 2 tel que : $x(n)=e(n)+b_1 \cdot e(n-1)+b_2 \cdot e(n-2)$

- Calculer μ_x .
- Connaissant les paramètres du modèle, déterminer les $R_x(k)$ Réponses

$$\mu_x = 0 \quad R_x(0) = (1 + b_1^2) \cdot \sigma^2 \quad R_x(1) = b_1 \cdot \sigma^2 \quad R_x(k) = 0 \text{ pour } k \geq 2$$

$$b_1 = (R_x(0) \pm \sqrt{R_x(0)^2 - 4R_x(1)^2}) / 2R_x(1) \quad \sigma^2 = 2 / (R_x(0) \pm \sqrt{R_x(0)^2 - 4R_x(1)^2})$$

$$R_x(0) = (1 + b_1^2 + b_2^2) \cdot \sigma^2 \quad R_x(1) = (b_1 + b_1 b_2) \cdot \sigma^2 \quad R_x(2) = b_2 \cdot \sigma^2 \quad R_x(k) = 0 \text{ pour } k \geq 3$$

Remarque : Il est très important de remarquer que nous ne disposons que d'une seule réalisation du signal aléatoire à modéliser $y(n)$, de ce fait l'auto-corrélation statistique $R_{yy}(k)$ est obtenu de l'auto-corrélation temporelle en considérant le processus ergodique.

4. Modèle ARMA

Les signaux ARMA sont obtenus par passage d'un bruit blanc dans un filtre récursif appelé aussi filtre à réponse impulsionnelle infinie (R.I.I.). Ces signaux sont une combinaison des signaux AR et MA. La fonction de transfert du filtre présente un numérateur et un dénominateur:

$$\text{Soit } y(n) = \sum_{i=0}^M b_i x(n-i) - \sum_{i=1}^N a_i y(n-i) \quad H(z) = \sum_{i=0}^M b_i z^{-i} / \left(1 + \sum_{i=1}^N a_i z^{-i} \right)$$

$$\text{La corrélation statistique de } y(n) \text{ s'écrit alors : } R_{yy}(k) = - \sum_{i=1}^N a_i R_{yy}(k-i) + \sigma^2 \sum_{j=0}^{M-k} b_{j+k} \cdot b_j$$

C'est une équation non linéaire en a_i et b_j .

La modélisation ARMA peut se décomposer en une modélisation AR suivie d'une modélisation MA. Le modèle AR présente une simplicité de calcul par rapport aux modèles MA et ARMA du fait où les coefficients AR sont solutions d'un système linéaire d'équations. Alors que la détermination des coefficients MA et ARMA requiert la résolution d'équations non linéaires. Cependant, le modèle ARMA permet de modéliser aussi bien les minima que les maxima de la DSP et est donc moins restrictif que le modèle AR.

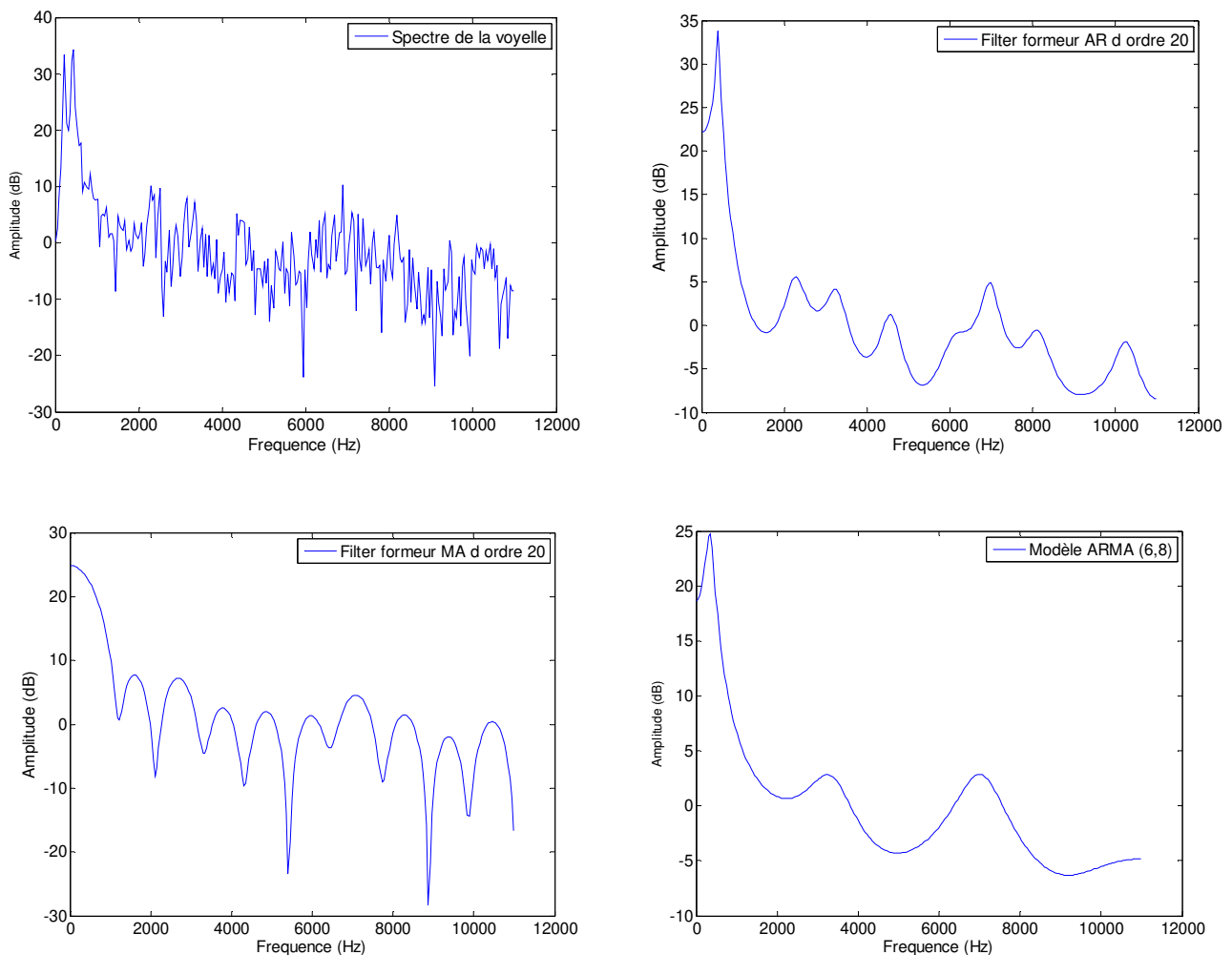
Les applications des modèles AR, MA, ARMA sont nombreuses, entre autres :

- la modélisation et la prédiction de série temporelle dite séries chronologiques Une série chronologique est une suite formée d'observations au cours du temps que l'on cherche à modéliser pour la prédiction de données futures. Ainsi, en finance, cela permet de modéliser le cours des devises ou du pétrole. Alors qu'en météorologie, les cela permet de faire des prévisions sur la température ou les précipitations. Dans chacun des cas, on essaiera à partir d'un échantillon de données de construire le meilleur modèle qui s'ajuste ces données.

- l'estimation du spectre d'un signal aléatoire, etc. Cette dernière application est basée sur l'identification des paramètres du modèle considéré : Le modèle AR est bien adapté aux signaux composés

de raies pures dans du bruit blanc. Alors que le modèle MA est bien adapté aux signaux dont la puissance est nulle dans certaines bandes de fréquences.

Exemple : Reprenons à nouveau l'exemple de parole, pour lequel, nous avons testé les trois variantes



Rappelons que pour un ordre du filtre faible, le modèle ne sera pas en mesure de capturer toutes les informations pertinentes du signal, à contrario s'il est trop élevé, il modélisera le bruit. Rappelons qu'avec le périodogramme (Algorithmes rapides : FFT) et ses variantes :

- Résolution en $f_e/N \Rightarrow$ Difficile de séparer 2 fréquences proches,
- Lobes secondaires occultant les fréquences à faibles amplitudes

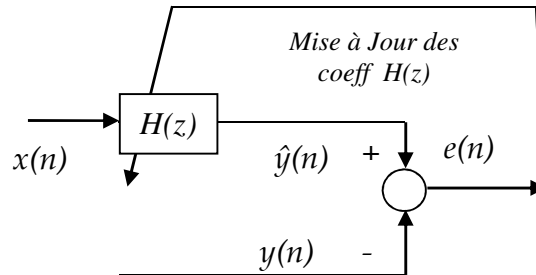
Nous n'irons pas plus loin, dans le cadre de ce cours, le but n'étant que d'introduire les notions de modélisation et de prédiction linéaire.

6. Filtrage Numérique adaptatif

Dans le filtrage de Wiener, le critère d'optimalité est stochastique : on désire minimiser la moyenne de l'erreur au carré. Cela requiert des statistiques de second ordre des processus (moyennes et corrélations). En

pratique, la matrice de corrélation $R_{xx}(k)$ et l'inter-corrélation $R_{yx}(k)$ ne sont pas connues et doivent donc être estimés à partir des données observées (temporelles) en supposant les processus ergodiques.

Le second problème est lié à l'hypothèse même de stationnarité qui n'est pas vérifiée à long terme. Le filtrage de Wiener devient inadéquat pour les situations dans lesquelles le signal ou le bruit sont non stationnaires. Dans de telles situations le filtre optimal doit être variable dans le temps. La solution à ce problème est fournie par le filtrage adaptatif.



Le filtrage adaptatif comporte une mise à jour récursive des paramètres (coefficients) du filtre. L'algorithme part de conditions initiales prédéterminées et modifie de façon récursive les coefficients du filtre pour s'adapter au processus. Pour cela, les coefficients de la réponse impulsionnelle du filtre sont adaptés en fonction de l'erreur par une boucle de retour.

- L'algorithme du gradient fournit un algorithme récursif de calcul des coefficients du filtre de Wiener (sans passer par l'inversion de R_{xx}).
- L'algorithme du gradient stochastique (LMS) est une version déterministe dans laquelle les grandeurs statistiques impliquées sont remplacées par des valeurs instantanées.
- L'algorithme des moindres carrés (RLS) qui est un algorithme récursif du LMS basé sur la minimisation de la somme des erreurs quadratiques sur un laps de temps donné.

6.1. Algorithme du Gradient (de descente)

On cherche à calculer les paramètres du filtre linéaire optimal.

- A l'instant n , ces paramètres sont notés $h(n) = [b_0(n) \ b_1(n) \ \dots \ b_{N-1}(n)]^T$ et
- les entrées du filtre forment le vecteur $X(n) = [x(n), x(n-1), \dots, x(n-N+1)]^T$.
- A chaque instant n , la sortie du filtre est un signal $\hat{y}(n)$ que l'on souhaite aussi proche que possible du signal original $y(n)$.

On rappelle que la solution optimale donnée par la solution des équations de Wiener-Hopf est : $R_{xx}h = R_{yx}$

avec $\xi(n) = E\{e^2(n)\} = E\left\{\left(y(n) - \sum_{i=0}^{N-1} b_i x(n-i)\right)^2\right\}$

L'algorithme procède en trois étapes, une étape d'initialisation et deux étapes qui sont itérées :

1. Initialisation : au temps $n = 0$, on attribue une valeur initiale $h(0) = [0, 0, \dots, 0]$
2. A chaque instant n (l'instant courant), on calcule le gradient de l'erreur (la dérivée) $\nabla_{h(n)} \xi(n)$ par rapport au vecteur $h(n)$: $\nabla_{h(n)} \xi(n) = -2R_{yx} + 2R_{xx}h(n)$
3. On ajuste les paramètres du filtre selon : $h(n+1) = h(n) - \frac{\mu}{2} \nabla_{h(n)} \xi(n) = h(n) + \mu (R_{yx} - R_{xx}h(n))$

$\frac{\mu}{2}$: un petit scalaire positif qui permet d'ajuster la vitesse et la précision des ajustements. A noter qu'il existe des versions à pas d'adaptation μ_n décroissant.

L'idée de cet algorithme est que les ajustements en sens opposé au gradient (descente du gradient) vont conduire $h(n)$ vers la valeur h_{opt} qui est associée à l'erreur minimale ξ_{min} . Si la dérivée est positive alors on va diminuer $h(n)$ de manière à se déplacer vers le minimum de la fonction de coût. Le terme correctif sera donc choisi négatif et vice-versa

L'algorithme est stable si $0 < \mu < \frac{2}{\lambda_{max}}$ où λ_{max} est le max des valeurs propres de R_{xx}

Cet algorithme est aussi appelé algorithme du gradient déterministe puisque les quantités R_{yx} et R_{xx} sont parfaitement connues et déterminées. On parle aussi de filtrage de Wiener adaptatif.

6.1. Algorithme du Gradient stochastique (LMS)

La méthode de descente (gradient) est un algorithme qui estime, par itérations successives, une solution qui tend vers la solution optimale de l'équation de Wiener-Hopf, sans inverser la matrice R_{xx} . Néanmoins, à chaque itération, il faut calculer les corrélations statistiques R_{xx} et R_{yx} que nous avons remplacé par des corrélations temporelles sous hypothèse d'ergodicité.

L'algorithme adaptatif connu sous le nom de Least-Mean-Square (LMS) repose sur la simplification du gradient de $\xi(n)$ soit $\nabla_{h(n)} \xi(n) = -2R_{yx} + 2R_{xx}h$ en remplaçant les moyennes statistiques (ergodiques : temporelles) R_{xx} et R_{yx} par des estimations instantanées :

$$\hat{R}_{xx} = X(n)X^T(n)$$

$$\hat{R}_{yx} = y(n)X(n)$$

Ainsi le gradient devient $\hat{\nabla}_{h(n)} \xi(n) = -2y(n)X(n) + 2X(n)X^T(n)h(n)$

A noter que minimiser l'erreur quadratique instantanée est moins efficace et moins précis que corriger l'erreur quadratique moyenne.

Ainsi,

$$\hat{h}(n+1) = \hat{h}(n) + \mu X(n) \overbrace{[y(n) - X^T(n)\hat{h}(n)]}^{e(n)}$$

Soit

$$\hat{h}(n+1) = \hat{h}(n) + \mu X(n)e(n)$$

1. Initialisation : au temps $n = 0$, $\hat{h}(0) = [0, 0, \dots, 0]$

2. Estimation du signal d'erreur : $e(n) = y(n) - X^T(n)\hat{h}(n)$

3. Actualisation des coefficients : $\hat{h}(n+1) = \hat{h}(n) + \mu X(n)e(n)$

L'algorithme LMS converge en moyenne, c'est-à-dire que, pour $n \rightarrow \infty$, $\hat{h}(n) \rightarrow h_{opt}$ des équations de Wiener-Hopf. A noter que l'algorithme de descente du gradient tend vers l'erreur minimale ξ_{min} lorsque n tend vers l'infini et ceci de façon exponentielle régulière, en raison de l'estimation exacte des R_{xx} et R_{yx} . Au contraire, l'algorithme LMS, qui repose sur des approximations très grossières de ses quantités converge avec des oscillations chaotiques autour ξ_{min} , et ce en dépit d'un choix correct de μ .

6.3. Algorithme des moindres carrés

Les équations de Wiener-Hopf sont obtenues en minimisant l'erreur quadratique moyenne $\xi(n) = E\{e^2(n)\} = E\{(y(n) - \hat{y}(n))^2\}$. L'algorithme des moindres carrés est basé sur la minimisation de la somme des erreurs quadratiques : $\xi(n) = \sum_{k=1}^n \lambda^{n-k} e^2(k)$ sur un laps de temps donné.

Si $\lambda = 1$, $\xi(n)$ est somme des erreurs quadratiques; Si $\lambda < 1$, les erreurs passées sont pondérées avec un facteur (d'oubli) qui décroît exponentiellement.

Version récursive du RMS,

$$\xi(n) = \sum_{k=1}^n \lambda^{n-k} e^2(k) \quad \text{où} \quad e(k) = y(k) - \hat{y}(k) = y(k) - h^T(n)X(k)$$

$$R_{xx\lambda}(n)h(n) = R_{yx\lambda}(n)$$

$$\text{Avec} \quad R_{xx\lambda}(n) = \sum_{k=1}^n \lambda^{n-k} X(k)X^T(k) = \lambda \sum_{k=1}^{n-1} \lambda^{n-k-1} X(k)X^T(k) = \lambda R_{xx\lambda}(n-1) + X(n)X^T(n)$$

$$\text{et} \quad R_{yx\lambda}(n) = \sum_{k=1}^n \lambda^{n-k} y(k)X(k) = \lambda \sum_{k=1}^{n-1} \lambda^{n-k-1} y(k)X(k) + y(n)X(n) = \lambda R_{yx\lambda}(n-1) + y(n)X(n)$$

Pour $\lambda = 1$, la matrice $R_{xx\lambda}(n)$ est une approximation de la matrice d'auto-corrélation. Elle est symétrique telle que : $R_{xx\lambda}(n)_{ij} = \frac{1}{n} \sum_{k=1}^n x(k-i+1)x(k-j+1)$

L'algorithme se présente comme suit :

$$\text{- Initialisation } R_{xx\lambda}^{-1}(0) = \left(\sum_{k=-n_0}^0 X(k)X^T(k) \right)^{-1} \quad \text{et} \quad R_{yx\lambda}(0) = \sum_{k=-n_0}^0 y(k)X(k)$$

$$\text{- } \hat{h}(n) = R_{xx\lambda}^{-1}(n)R_{yx\lambda}(n)$$

$$\text{avec} \quad R_{xx\lambda}(n) = \lambda R_{xx\lambda}(n-1) + X(n)X^T(n) \quad \text{et} \quad R_{yx\lambda}(n) = \lambda R_{yx\lambda}(n-1) + y(n)X(n)$$

Remarque : Le filtrage des moindres carrés se place en déterministe. Alors que le filtre de Wiener utilise la méthode des moindres carrés en stochastique.

Exemples d'application

Le filtrage adaptatif (au sens de Wiener, au sens des moindres carrés, le filtrage de Kalman, le filtrage particulaire) est utilisé pour divers objectifs entre autres : la modélisation, la prédiction, l'identification des systèmes, l'annulation d'interférence (écho, bruit), etc. permettent la reconstitution d'un signal en milieu bruit

- L'annulation d'écho va nous fournir un premier exemple d'utilisation de l'algorithme LMS. Téléphone (micro et haut-parleur proches).
- Annulation d'un brouilleur tel un signal perturbé par un brouilleur sinusoïdal de fréquence connue mais dont on ignore l'amplitude A et la phase φ .

6.3. Autres filtres adaptatifs

- Kalman : Le filtre de Kalman est employé pour estimer des paramètres d'un système évoluant dans le temps à partir de mesures bruitées. Il est particulièrement adapté pour le tracking du fait qu'il permet non seulement de prédire les paramètres mais aussi de rectifier les erreurs de mesure et du modèle. Il opère de façon similaire aux filtres adaptatifs. La prédiction se fait en employant l'estimation précédente. Le calcul d'erreur permet, ensuite, de faire la mise à jour de cette prédiction grâce aux nouvelles mesures.

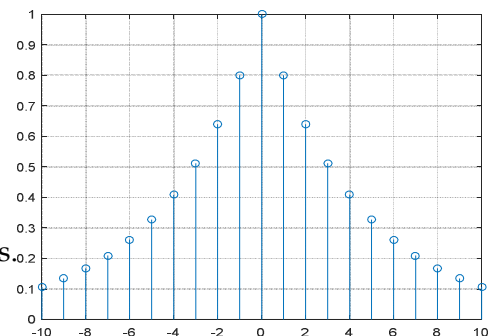
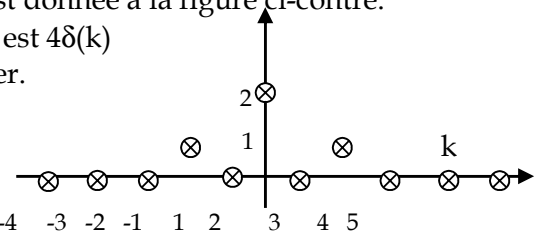
A noter que la version étendue (Filtre de Kalman étendu) permet de prendre en compte une modélisation non linéaire.

- Particulaire (méthode de Monté-Carlo séquentielle) : Une alternative aux filtres de Kalman étendu. Ce filtre a pour but d'estimer la densité postérieure des variables qu'on cherche à estimer (dites variables d'état) en fonction des observations (variables d'observation) en supposant que les variables d'état constituent une chaîne de Markov de premier ordre.

Il est utilisé particulièrement pour des applications de navigation : suivi de cible, guidage de missile, robotique, etc.

TD n° 4 : Modélisation AR et Filtrage adaptatif

- Soit un signal aléatoire $y(n)$ SSL et ergodique dont l'autocorrélation temporelle $\overline{R}_y(k) = \alpha^{|k|}$ avec $0 < \alpha < 1$
 - Identifier le modèle linéaire adéquat (AR ou MA) pour $y(n)$.
 - En supposant que le système $h(n)$ est filtre purement récursif, donner le schéma du modèle en définissant l'entrée, le système, et la sortie.
 - Rappeler les hypothèses nécessaires liées à l'emploi de ce modèle.
 - On considère que le modèle est d'ordre 1, déterminer ses paramètres.
- On considère le filtre linéaire à temps discret défini par $y(n) = x(n) + b_1 x(n-1) + b_2 x(n-2)$.
où $X(n)$ et $Y(n)$ désignent respectivement les processus aléatoires réels d'entrée et de sortie du filtre où b_1 et b_2 sont 2 coefficients réels. On suppose que $x(n)$ est une suite de variables aléatoires centrées, indépendantes et de variance σ^2 .
 - Donner l'expression de $R_x(k)$ et $S_x(f)$ puis donner l'expression de $R_y(k)$ et tracer la pour $b_1=1$ et $b_2=-1$.
 - Connaissant la DSP du signal $y(n)$, sur quoi se base-t-on pour le choix du modèle ?
- Soit un processus AR défini par : $y(n) = -a_1 y(n-1) - a_2 y(n-2) + x(n)$ où $x(n)$ est un processus blanc décorrélé de variance 1
 - Calculer $\mu_y(n)$ puis sans calcul, expliquer pourquoi $y(n)$ est SSL.
 - Montrer que pour $k > 0$, $R_y(k) = -a_1 R_y(k-1) - a_2 R_y(k-2)$
 - Déterminer a_1 et a_2
- On veut modéliser le signal $y(n)$ dont l'autocorrélation statistique est donnée à la figure ci-contre. $R_y(k)$. On suppose que l'autocorrélation statistique du signal d'entrée est $4\delta(k)$
 - Déterminer les moments statistiques d'ordre 1 du signal à modéliser.
 - S'agit-il d'un modèle AR ou MA? Justifier
 - Déterminer l'ordre du modèle
 - En supposant que 2 coefficients sont égaux, à partir des valeurs de $R_y(k)$, déduire que l'un des coefficients est nul puis déterminer les coefficients de ce modèle puis donner l'équation aux différences liant $y(n)$ et $x(n)$
- On veut modéliser le signal $y(n)$ SSL dont la corrélation statistique n'a été tracée que de -10 à 10. Elle est donnée à la figure ci-contre.
 - S'agit-il d'un modèle AR ou MA? Justifier
 - Donner les propriétés statistiques du signal d'entrée.
 - On suppose que le modèle est d'ordre 1, déterminer ses paramètres.
 - Expliquer la notion de filtre formeur et son utilité en téléphonie



Solutions

- Modèle AR ($R_y(k) \neq 0$) $x(n)$ entrée bb, $y(n)$ signal aléatoire à modéliser $h(n)$ filtre formeur (modèle math) entrée bb + ergodisme $a_1 = -\alpha$ $\sigma_x^2 = 1 - \alpha^2$
- $R_x(k) = \sigma_x^2 \delta(k)$ $S_x(f) = \sigma_x^2$ MA d'ordre 2 $R_y(0) = 1 + b_1^2 + b_2^2$ $R_y(1) = b_1(1 + b_2)$ $R_y(2) = b_2$ $R_y(k) = 0$ pour $k \geq 3$
- $\mu_y(n) = 0$ entrée bb SSL $\Rightarrow$ sortie SSL Interro 1 14/15 4. Voir examen 17/18 5. Examen 16/17

Exercices supplémentaires

1. On considère un signal aléatoire stationnaire $x(n)$ et l'on suppose connu ses coefficients d'autocorrélation : $R(0) = 3\sigma^2$, $R(1) = 2\sigma^2$, $R(2) = \sigma^2$, $R(3) = 0$

- On cherche le filtre MA d'ordre 3 de ce signal. Identifiez les paramètres du filtre.

- Si le signal $x(n)$ a été obtenu par filtrage d'un bruit blanc gaussien de variance σ^2 par un filtre à réponse impulsionnelle finie de fonction de transfert $H(z) = 1 + a.z^{-1} + b.z^{-2}$, en déduire les valeurs de a et b .

2. Soit un filtre formeur dont l'équation aux différences est $y(n) = 0.5(x(n) + x(n-1) + x(n-2) + x(n-3))$

- Expliquer la notion de filtre formeur puis identifier ce modèle linéaire AR ou MA
- Donner la moyenne, l'autocorrélation et la DSP de son entrée $x(n)$.
- Calculer et tracer $R_{yy}(k)$ puis déterminer $S_{yy}(f)$

3. Les signaux $x(n)$ et $y(n)$ ont été obtenus en filtrant, au moyen d'un filtre à réponse impulsionnelle finie, un bruit blanc $b(n)$ gaussien centré de variance σ^2 .

$$x(n) = 2.b(n) + 0.5.b(n-1) - 0.2.b(n-2) + 0.1.b(n-3) \quad y(n) = b(n) - b(n-2)$$

- Identifier les 2 modèles puis pour chacun, calculer et tracer les coefficients d'autocorrélation
- $x(n)$ et $y(n)$ sont-ils Gaussiens (justifier)
- Calculer les intercorrélations $R_{xy}(k)$, et $R_{yx}(k)$ et commenter
- Donner quelques applications des modèles AR, MA, ARMA

4. Soit un filtre formeur dont l'équation aux différences est $y(n) = -\alpha y(n-1) - \beta y(n-2) + x(n)$

- Identifier l'ordre du modèle linéaire AR.
- Déterminer les paramètres du modèles, on suppose que $R_{yy}(k) = 2 \cdot 0.5^{|k|}$

5. Soit un filtre formeur dont l'équation aux différences est $y(n) = \alpha y(n-1) + x(n)$

- Identifier l'ordre du modèle linéaire AR.
- Déterminer la moyenne de $y(n)$ et montrer que $R_{yy}(k) = \alpha^k R_{yy}(0)$, déduire une condition sur α .
- Montrer que $R_{yy}(0) = R_{xx}(0) / (1 - \alpha^2)$
- Tracer $R_{yy}(k)$ (Prendre $R_{xx}(0) = \sigma^2 = 1$).
- Citer 2 applications concrètes des modèles AR.

6. Soit un filtre formeur dont l'équation aux différences est $y(n) = 0.25 y(n-1) - 0.25 y(n-2) + x(n)$

- Identifier ce modèle linéaire AR ou MA (Justifier)
- Déterminer la moyenne de $y(n)$ et donner l'expression de $R_{yy}(k)$.
- Calculer et tracer $R_{yy}(k)$ (Prendre $R_{xx}(0) = \sigma^2 = 1$).

Solutions

1. $b_0=1$ $b_1=1$ et $b_2=1$ et par identification, on trouve $a=1$ et $b=1$

2. MA, $\mu_x=0$ et $S_x(f) = \sigma^2 R_y(0) = \sigma^2$, $R_y(1) = R_y(-1) = 0.75 \sigma^2$, $R_y(-2) = R_y(2) = 0.5 \sigma^2$, $R_y(-3) = R_y(3) = 0.25 \sigma^2$, $R_y(k > 3) = 0$. $S_y(f) = \sigma^2 (1 + 1.5 \cos(4\pi f) + \cos(6\pi f) + 0.5 \cos(8\pi f))$

3. Interro 2 15/16 4. Ratt 15/16 5. Examen 15/16 6. Ratt 14/15

TP n° 5 : Modélisation paramétrique et Filtrage Numérique adaptatif

Ce TP a pour objectif :

- Modéliser un signal aléatoire par les modèles AR et MA (par approximation AR) et d'en extraire les informations utiles.
- Tester une méthode de filtrage adaptatif (LMS)

Exercice 1: Télécharger le fichier 'vous avez du courrier en attente.wav' et le placer dans le même répertoire que votre programme puis sélectionner une partie entre 21000 et 21500

I. Démo

-*- coding: utf-8 -*-

```

""" Modélisation AR """
# import numpy as np; import matplotlib.pyplot as plt;
# from scipy.io import wavfile as wf;
# import scipy.linalg as la; import scipy.signal as sp;

# fname = 'vousavezducourrierenattente.wav';
# #winsound.PlaySound(fname, winsound.SND_FILENAME)
# fe, S = wf.read(fname);
# Te=1/fe; N=len(S); t = np.linspace(0, N-1, N)*Te;
# plt.figure(1);plt.subplot(211); plt.plot(t,S); plt.grid(True);
# plt.xlabel('temps'); plt.ylabel('Amp'); plt.title('Phrase');
# L=500;N1=21000;N2=N1+L; y=S[N1:N2] ; ty=np.linspace(N1, N2-1, N2-N1)*Te;
# plt.subplot(212); plt.plot(ty,y); plt.grid(True);
# plt.xlabel('temps'); plt.ylabel('Amp'); plt.title('morceau de la phrase');

# """ Modélisation AR """
# P=20;      "P Nbre de coefficients du modèle AR>2"
# y=y-np.mean(y)
# R = np.correlate(y,y,mode='full')[len(y)-1:];
# #R=R/R.max(); "Autocor"
# C = la.toeplitz(R[0:P]);
# B = -R[1:P+1];
# a = la.inv(C).dot(B)
# a = np.append(1,a)
# sigma_carre = sum(a*R[0:P+1]);
# """ Visualisation de la réponse fréquentielle du filtre trouvé et du signal y de départ """
# b = np.array([1,0]);
# f,H= sp.freqz(b,a,L//2,fs=fe);
# TFy = np.fft.fft(y); TFy = TFy[1:L//2+1];
# plt.figure(2); plt.subplot(121);plt.plot(f, 20*np.log10(abs(TFy/TFy.max())));
# plt.plot(f, 20*np.log10(abs(H/H.max())));
# plt.title('Module du Filtre trouvé (orange)+ Signal original modélisé(bleu)');
# plt.xlabel('Fréq (Hz)');plt.ylabel('Amp');
# # """ Synthèse à partir du filtre trouvé """
# bb = (sigma_carre**0.5)*np.random.randn(L);
# #bb=bb-np.mean(bb);
# y_synt = sp.lfilter(b,a,bb) ;

```

```
# TFys = np.fft.fft(y_synt); TFys = TFys[0:L//2];
# plt.figure(2); plt.subplot(122); plt.plot(f, 20*np.log10(abs(TFys/TFys.max())));
# plt.plot(f, 20*np.log10(abs(TFy/TFy.max())));
# plt.title('signal synthétisé (orange)+ Signal synthétisé(bleu)');
# plt.xlabel('Fréq (Hz)'); plt.ylabel('Amp');
"""-----"""

""" LMS : Algorithme du Gradient stochastique : Restauration """
import numpy as np
import matplotlib.pyplot as plt
import scipy.signal as sp

fecg=[]; mecg=[];
""" Lecture des fichiers """
with open('fecg.txt') as f:
    for i in f: fecg.append(float(i))
fecg = np.array(fecg)
with open('mecg.txt') as f:
    for i in f: mecg.append(float(i))
mecg = np.array(mecg)
""" Rajout du bruit """
N=len(mecg); var = 0.01;
bb=(var**0.5)*np.random.randn(N);
x=fecg+bb+mecg; #x=x-np.mean(x)
y=fecg; #y=y-np.mean(y)

""" Détermination du filtre par LMS """
Ncoef=7; mu=0.1;
h = np.zeros(Ncoef) # Initialisation des coefficients du filtre
L = len(x) # Longueur du signal
yest = np.zeros(N) # y(n) estimé
En = np.zeros(N) # Erreur
for i in range(Ncoef,L//10):
    X = np.flipud(x[i-Ncoef+1:i+1]); #Prendre X=[x(n),x(n-1),...,x(n-Ncoef+1)]
    yest[i] = np.dot(h, X)
    En[i] = x[i] - yest[i]
    h = h + mu * En[i] * X
h = h/sum(abs(h))

""" Visualisation des résultats """
plt.figure(1)
y_est = sp.lfilter(h,1,x)
plt.subplot(311); plt.plot(x,'r', label= 'observation'); plt.plot(y,'b', label= 'fecg'); plt.grid(); plt.legend()
plt.subplot(313); plt.plot(En,'g',label='erreur'); plt.grid(); plt.legend()
plt.subplot(312); plt.plot(y,'b', label= 'fecg'); plt.plot(y_est,'r', label= 'ecg restauré'); plt.legend()
"""-----"""

# """ Prédiction par LMS """
# mux=0; varx=0.01; N=1000
# bb=mux+np.sqrt(varx)*np.random.randn(N)

# fe=10000 ;Te=1/fe;
# f0= 100;
```

```
# t = np.arange(0,N)*Te
# y = 0.5*np.cos(2*np.pi*f0*t)
# x = y + bb

# Ncoef=5; mu=0.2;
# hp = np.zeros(Ncoef) # Initialisation des coefficients du filtre
# L = len(x) # Longueur du signal
# yest = np.zeros(N) # y(n) estimé
# En = np.zeros(N) # Erreur
# for i in range(Ncoef,L-1):
#     X= np.flipud(x[i-Ncoef+1:i+1]); #Prendre X=[x(n),x(n-1),...,x(n-Ncoef+1)]]
#     yest[i] = np.dot(hp, X)
#     En[i] = x[i] - yest[i]
#     hp = hp + mu * En[i] * X

# plt.figure(2); plt.plot(x,'b', label= 'observation'); plt.plot(yest,'r', label= 'estimé')
# plt.plot(En,'g', label= 'Erreur instantanée'); plt.grid(); plt.legend()
"""-----"""

""" Identification de système par LMS """
# mux=0; varx=0.1; N=500
# x = mux+np.sqrt(varx)*np.random.randn(N)
# b = np.array([0.65, -0.35, 0.1])
# y = sp.lfilter(b,[1,0],x)

# Ncoef=3; mu=0.6;
# hs = np.zeros(Ncoef) # Initialisation des coefficients du filtre
# yest = np.zeros(N) # y(n) estimé
# En = np.zeros(N) # Erreur
# for i in range(Ncoef,N-1):
#     X= np.flipud(x[i-Ncoef+1:i+1]); #Prendre X=[x(n),x(n-1),...,x(n-Ncoef+1)]]
#     yest[i] = np.dot(hs, X)
#     En[i] = y[i] - yest[i]
#     hs = hs + mu * En[i] * X

# print(hs)
# plt.figure(3); plt.plot(y,'b', label= 'Sortie'); plt.plot(yest,'r', label= 'Sortie estimée')
# plt.plot(En,'g', label= 'Erreur instantanée'); plt.grid(); plt.legend()

"""A faire AR """

# import numpy as np; import scipy.signal as sp; import matplotlib.pyplot as plt
# import sounddevice as snd
# import scipy.io.wavfile as wav
# import scipy.linalg as la;

# def ModeleAR(y,P):
```

```
# y=y-np.mean(y)
# R = np.correlate(y,y,mode='full')[len(y)-1:]
# C = la.toeplitz(R[0:P])
# B = -R[1:P+1]
# a = la.inv(C).dot(B)
# a = np.append(1,a)
# sigma_carre = sum(a*R[0:P+1]);
# return a,sigma_carre

# fe,x = wav.read('voyelles.wav')
# snd.play(x, fe)

# 1. Lire, afficher et écouter le fichier voyelles.wav
# 2. Repérer les 3 parties correspondants aux 3 "A"
# NA=7000
# A1 = x[162400:162400+NA];
# A2 = x[253200:253200+NA]
# A3 = x[340000:340000+NA]
# 3. Afficher les 3 "A"
# 4. Calculer et afficher leur periodogramme en db
# 5. Déterminer le filtre formeur de chaque "A" en utilisant la fonction ModeleAR
# 6. Calculer l'erreur quadratique moyenne (MSE) entre les coefficients de chaque A
# from sklearn.metrics import mean_squared_error
# mseA12=mean_squared_error(a1,a2)
# 7. Refaire les étapes de 2 à 6 pour la lettre "E"
# 8. Calculer le MSE entre les 3 "A" et les 3 "E"

"""A faire Restauration par LMS """
"""
# 1. Lire l'audio vousavezducourrierenattente.wav dans une variable y
# 2. Créer un signal x composé du signal audio (y) + du bruit blanc
# 3. Écouter pour vérifier la présence du bruit
# 4. Centrer les variables y et x
# 5. Déterminer par LMS le filtre h en prenant la partie correspondant au silence (mu=0.01 et for i in
range(Ncoef,100) )
# 6. Inverser et normaliser le filtre h
# 7. Filtrer x par le filtre
# 8. Visualiser les résultats
# 9. Écouter l'audio restauré
# ss=input()
# snd.play(np.int8(y_est), fe)
"""
```

IV. Analyse temps-fréquence et temps-échelle

La plupart des signaux du monde réel ne sont pas stationnaires, et c'est justement dans l'évolution de leurs caractéristiques (statistiques, fréquentielles, temporelles, spatiales) que réside l'essentiel de l'information qu'ils contiennent. Les signaux vocaux et les images en sont des exemples courants [27]. Rappelons que l'analyse de Fourier permet une caractérisation globale du signal (on intègre de $-\infty$ à $+\infty$), on perd toute localisation temporelle ou spatiale, l'idéal est de faire appel à une transformation qui nous apporte l'information sur le contenu fréquentiel tout en préservant la localisation (temporelle ou spatiale) afin d'obtenir une représentation temps/fréquence ou espace/échelle du signal. C'est ainsi que deux théories ont été élaborées, la transformée de Fourier à fenêtre puis la transformée continue par ondelettes.

Les ondelettes, famille de fonctions déduites d'une même fonction, appelée ondelette mère, par opérations de translations et dilatations, ont trouvé, de par la puissance de leur théorie, des applications dans de nombreux domaines aussi variés que les mathématiques (analyse, probabilités, fractales), le traitement du signal (compression, astronomie, sismique), la physique (mécanique quantique, turbulence).

1. Dualité temps-fréquence, transformée de Fourier à court terme

La transformée de Fourier à court terme (TFCT) et ses dérivées (notamment le spectrogramme) sont les méthodes temps-fréquence les plus utilisées dans les applications pratiques. Ainsi, cette classe de méthodes représente la solution la plus répandue pour éliminer les limitations de la transformée de Fourier. L'idée de base est très simple et efficace : on décompose le signal en petits segments et on applique, sur chacune des sections, la transformée de Fourier en obtenant ainsi le spectre « local ». La totalité des spectres « locaux » indique alors comment le spectre varie au cours du temps.

$$TFCT(t, f) = \int_{-\infty}^{+\infty} x(\tau) h^*(\tau - t) e^{-j2\pi f\tau} d\tau$$

Où $h(n)$ est une fenêtre de pondération (Rectangulaire, Bartlett, Hanning, Hamming, Gaussienne etc.)

Le spectrogramme est le module au carré de la TFCT

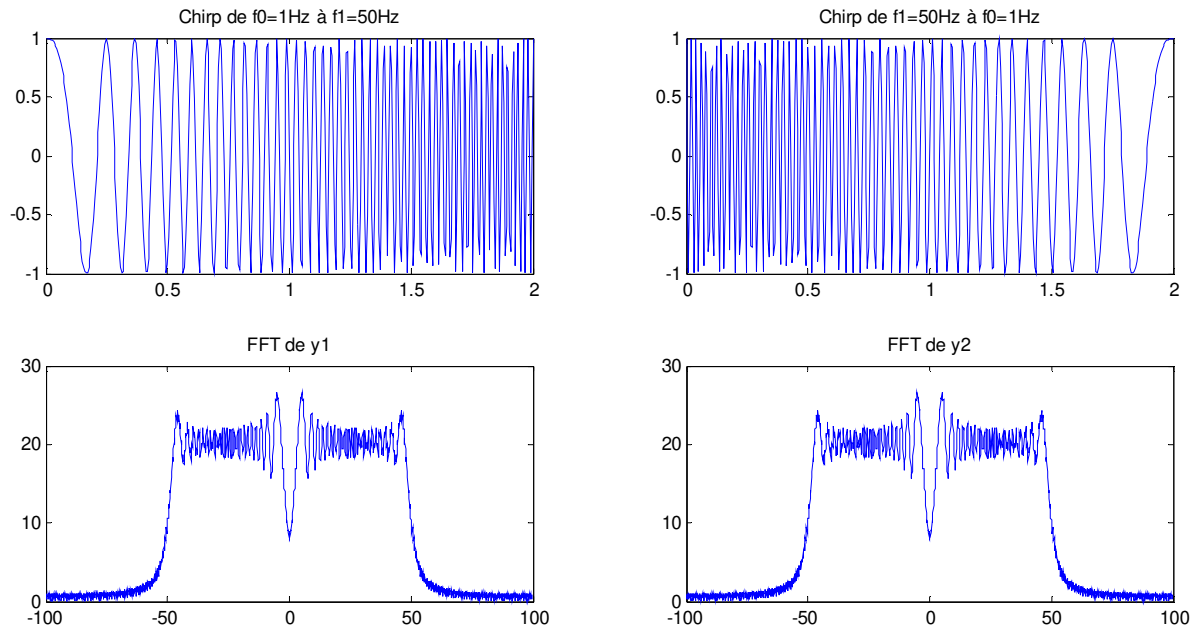
$$S_x(t, f) = \left| \int_{-\infty}^{+\infty} x(\tau) h^*(\tau - t) e^{-j2\pi f\tau} d\tau \right|^2$$

Il faut relever n'existe pas qu'une seule TFCT puisqu'elle dépend de : La durée de la fenêtre (choisie pour que le signal soit supposé stationnaire sur cette durée), la forme de la fenêtre (compromis largeur-hauteur des lobes), le taux de recouvrement entre les fenêtres.

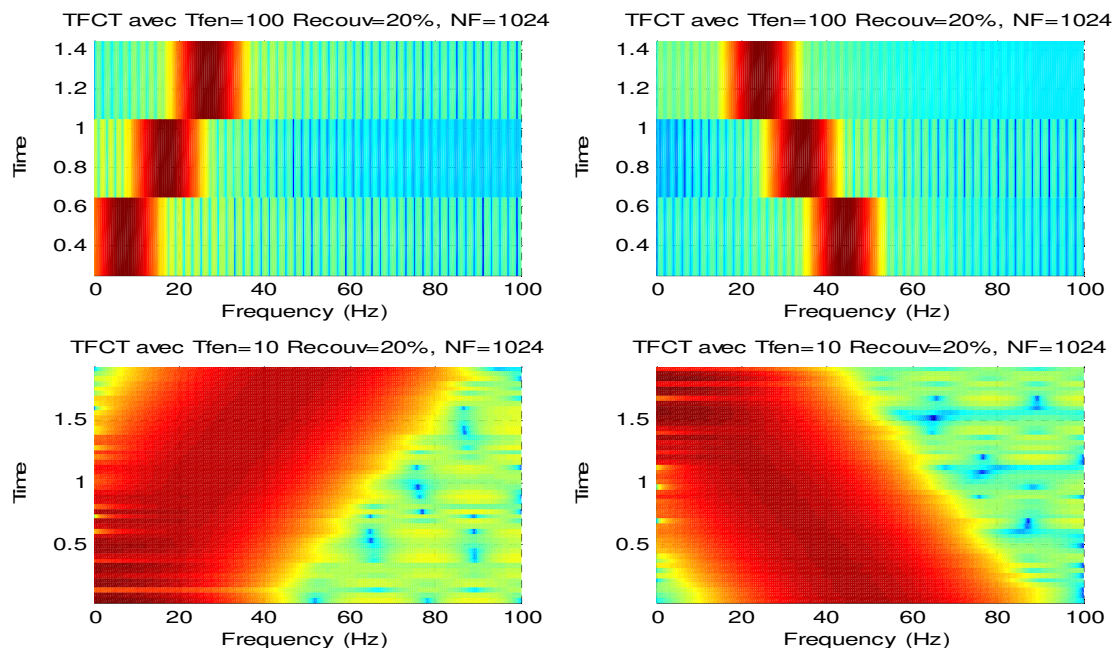
Le rôle de la fenêtre $h(t)$ (dont l'énergie doit valoir 1) est de découper un voisinage de longueur L du point t , dans lequel le contenu fréquentiel est analysé. On conçoit qu'il y a un compromis entre la longueur L de

$h(t)$ qui représente la résolution temporelle, qui induit une résolution fréquentielle en f_0/L , et la capacité de la TFCT à suivre des modulations plus ou moins rapides. Ces 2 résolutions évoluent en inverse l'une de l'autre. Il a été montré (principe d'incertitude d'Heisenberg) que la fenêtre qui réalise le meilleur compromis temps-fréquence est la fenêtre gaussienne.

Exemple : Analyse d'un signal chirp allant de f_1 puis f_2 et inversement (voir TP n°6)



Comme illustré ci-haut, on perd toute localisation temporelle puisqu'on obtient la même TF pour les deux signaux. Pour y remédier on calcule la TFCT en effectuant la TFD pour différents intervalles et recouvrements



Les TFDs sur chaque fenêtre glissante fournissent le spectrogramme qui permet d'adapter la TF à la caractérisation des signaux non stationnaires. On obtient, alors, une représentation temps-fréquence permettant de localiser la distribution de l'énergie simultanément en temps et fréquence. Rappelons que la

longueur de la fenêtre choisie va conditionner le nombre de points fréquentiels (résolution fréquentielle) et la résolution temporelle (spectre moyen sur la fenêtre temporelle).

Il est clair que pour ce cas, la TFCT opérant avec une taille de fenêtre unique ne permet pas de localiser précisément chaque fréquence. Celle-ci devrait s'adapter en fonction de l'évolution du signal. L'idéal serait de pouvoir choisir une fenêtre et une forme d'onde (signal oscillant dans une fenêtre temporelle donnée) que l'on pourrait dilater (pour les basses fréquences) et contracter (fréquences élevées) à volonté.

2. Transformée de Wigner-Ville (temps-fréquence)

A l'instar de la transformée de Fourier à court terme, elle permet d'illustrer la distribution de l'énergie simultanément en temps et en fréquence.

Rappelons que la densité spectrale spectrale est : $S_x(f) = |X(f)|^2 = TF\{R_x(\tau)\}$

$$\text{Où } R_x(\tau) = \int_{-\infty}^{\infty} x(t)x^*(t-\tau)dt = \int_{-\infty}^{\infty} x(t+\tau/2)x^*(t-\tau/2)dt$$

L'expression $x(t+\tau/2)x^*(t-\tau/2)$ peut-être alors considérée comme une corrélation instantanée, pour laquelle on peut définir une densité spectrale instantanée plus communément appelée Transformée temps-fréquence de Wigner-Ville :

$$WV_x(t, f) = \int_{-\infty}^{\infty} x(t+\tau/2)x^*(t-\tau/2)e^{-2\pi jf\tau}d\tau$$

Cette transformée conserve bon nombre des propriétés de la TF sauf qu'elle ne conserve pas la propriété de linéarité :

$$x(t) = \sum_{k=1}^N x_k(t) \rightarrow WV_x(t, f) = \sum_{k=1}^N WV_{x_k}(t, f) + \sum_{k,m=1(k \neq m)}^N WV_{x_k x_m}(t, f)$$

Le second terme montre que cette transformée crée des fréquences (interférences) qui n'ont pas d'existence réelle. En outre, il fait apparaître des énergies négatives engendrant des difficultés d'interprétation. Une version pseudo lissée permet de résoudre partiellement quelques uns de ces problèmes.

Exemples :

$$- x(t) = e^{2\pi j f_0 t} \rightarrow WV_x(t, f) = \delta(f - f_0)$$

$$- y(t) = \cos(2\pi f_0 t) = \frac{e^{2\pi j f_0 t} + e^{-2\pi j f_0 t}}{2}$$

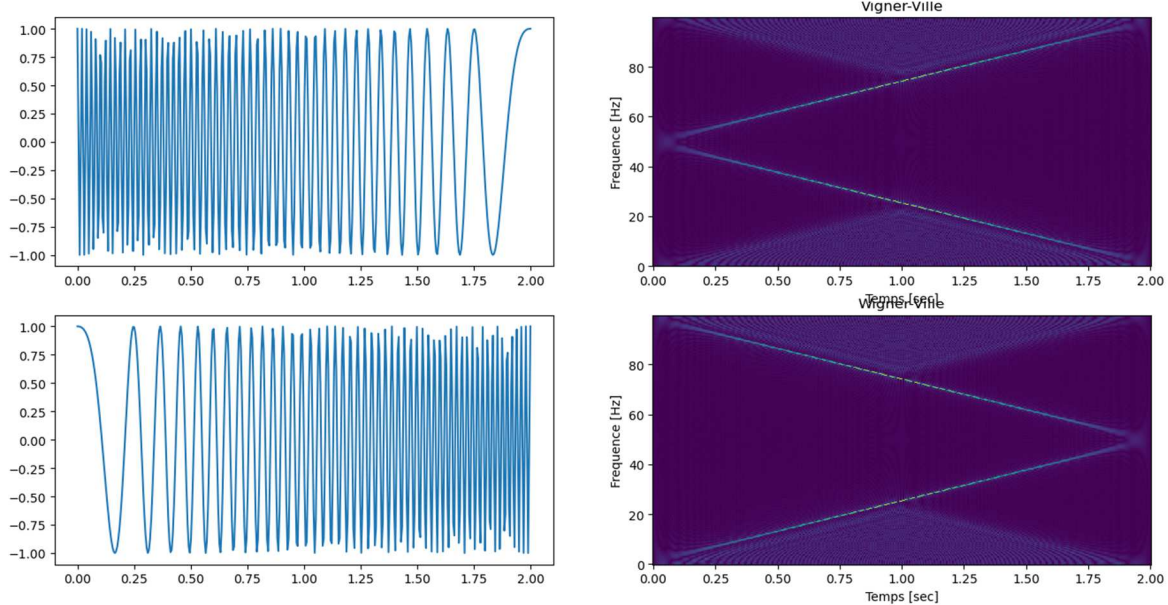
$$WV_y(t, f) = \frac{1}{4} \left(\int_{-\infty}^{\infty} [e^{2\pi j f_0 (t+\frac{\tau}{2})} + e^{-2\pi j f_0 (t+\frac{\tau}{2})}] [e^{-2\pi j f_0 (t-\frac{\tau}{2})} + e^{2\pi j f_0 (t-\frac{\tau}{2})}] e^{-2\pi j f \tau} d\tau \right)$$

$$WV_y(t, f) = \frac{1}{4} \left(\int_{-\infty}^{\infty} e^{2\pi j f_0 \tau} e^{-2\pi j f \tau} d\tau + \int_{-\infty}^{\infty} e^{4\pi j f_0 t} e^{-2\pi j f \tau} d\tau + \int_{-\infty}^{\infty} e^{-4\pi j f_0 t} e^{-2\pi j f \tau} d\tau + \int_{-\infty}^{\infty} e^{-2\pi j f_0 \tau} e^{-2\pi j f \tau} d\tau \right)$$

$$WV_y(t, f) = \frac{1}{4} (\delta(f - f_0) + e^{4\pi j f_0 t} \delta(f) + e^{-4\pi j f_0 t} \delta(f) + \delta(f - f_0))$$

$$\rightarrow WV_y(t, f) = \frac{1}{4}(\delta(f - f_0) + \delta(f + f_0)) + 2\cos(4\pi f_0 t)\delta(f)$$

Exemple : Analyse d'un signal chirp allant de f1 puis f2 et inversement



En dépit des interférences, on observe une bonne localisation des fréquences.

3. Ondelettes continues (temps-échelle)

Elle a été introduite par Jean Morlet en 1981 pour résoudre des problèmes de signaux sismiques en recherche pétrolière.

Partant d'une fenêtre h (dite fonction mère) ayant pour symbole ψ dépendant de t , on peut générer un ensemble de fonctions de base similaire par dilatation (indice a) et translation (indice b) d'un seul prototype $\psi_{a,b}(t)$:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad a > 0$$

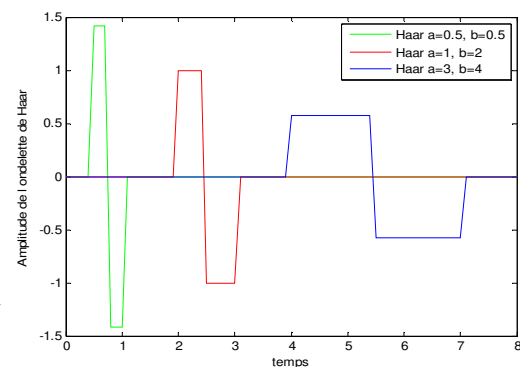
Où $a > 0$ est un paramètre d'échelle de contraction ($a < 1$) ou de dilatation ($a > 1$) de la fenêtre et b une translation de la fenêtre. On notera que la norme de $\psi_{a,b}$ est conservée lors du changement de facteur d'échelle (voir figure ci-dessous):

$$\|\psi_{a,b}\| = \int_{-\infty}^{+\infty} \frac{1}{a} \left| \psi\left(\frac{t-b}{a}\right) \right|^2 dt = \|\psi\|^2$$

Exemple :

$$\text{La fonction de Haar } \psi(t) = \begin{cases} 1 & \text{si } 0 \leq t \leq \frac{1}{2} \\ -1 & \text{si } \frac{1}{2} \leq t \leq 1 \\ 0 & \text{ailleurs} \end{cases}$$

$$\text{Ainsi } \psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) = \begin{cases} \frac{1}{\sqrt{a}} & \text{si } b \leq t \leq b + \frac{a}{2} \\ -\frac{1}{\sqrt{a}} & \text{si } b + \frac{a}{2} \leq t \leq b + a \\ 0 & \text{ailleurs} \end{cases}$$



Exemples d'ondelettes

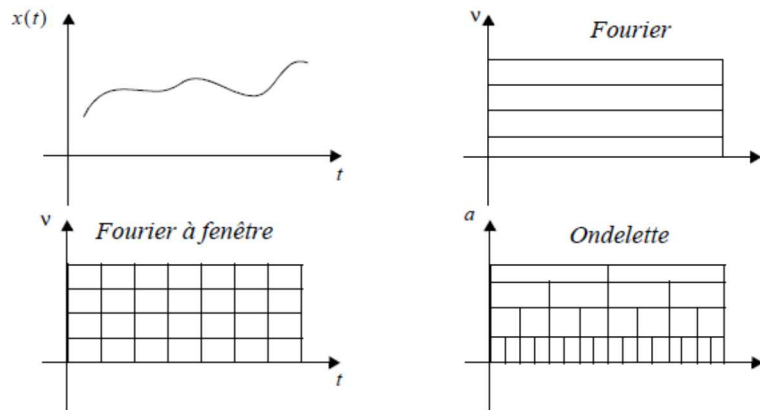
- L'ondelette chapeau mexicain (Paul à l'ordre 2) $\psi(t) = \frac{1}{\sigma\sqrt{2\pi}} \left(1 - \frac{t^2}{\sigma^2}\right) e^{-\frac{t^2}{2\sigma^2}}$
- L'ondelette de Morlet $\psi(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}} e^{-2\pi j f t}$

L'ondelette de Morlet est une sinusoïde complexe modulée par une gaussienne. L'ondelette de Paul décroît plus vite que celle de Morlet et autorise des localisations en temps plus précises. L'ondelette est obtenue par dérivée d'une gaussienne et permet des localisations temporelles d'une qualité légèrement inférieure à celle de Paul [25]

En décomposant le signal $x(t)$ sur cette famille, on obtient ainsi les coefficients d'ondelettes $WT_{x,\psi}(a, b)$ qui caractérisent le coefficient de la décomposition du signal $x(t)$ dans cette base, soit :

$$WT_{x,\psi}(a, b) = \int_{-\infty}^{\infty} x(t) \psi_{a,b}^*(t) dt = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t) \psi\left(\frac{t-b}{a}\right)^* dt$$

L'analyse par ondelettes commence avec une fenêtre d'analyse a de largeur très fine, la translate sur tout le signal puis recommence en augmentant l'échelle [25]. Ses coefficients mesurent, en un certain sens, les fluctuations du signal $x(t)$ autour du point $t = b$, à l'échelle fournie par a . En diminuant a , le support de $\psi_{a,b}$ réduit en temps et donc couvre une plage en fréquence plus grande et vice versa. Donc $1/a$ est proportionnel à une fréquence.



Pour un facteur d'échelle assez grand, la représentation des coefficients d'ondelettes en fonction de b , la position, donne une représentation de "la forme générale de la fonction". Par contre un facteur d'échelle faible correspond à une représentation des singularités.

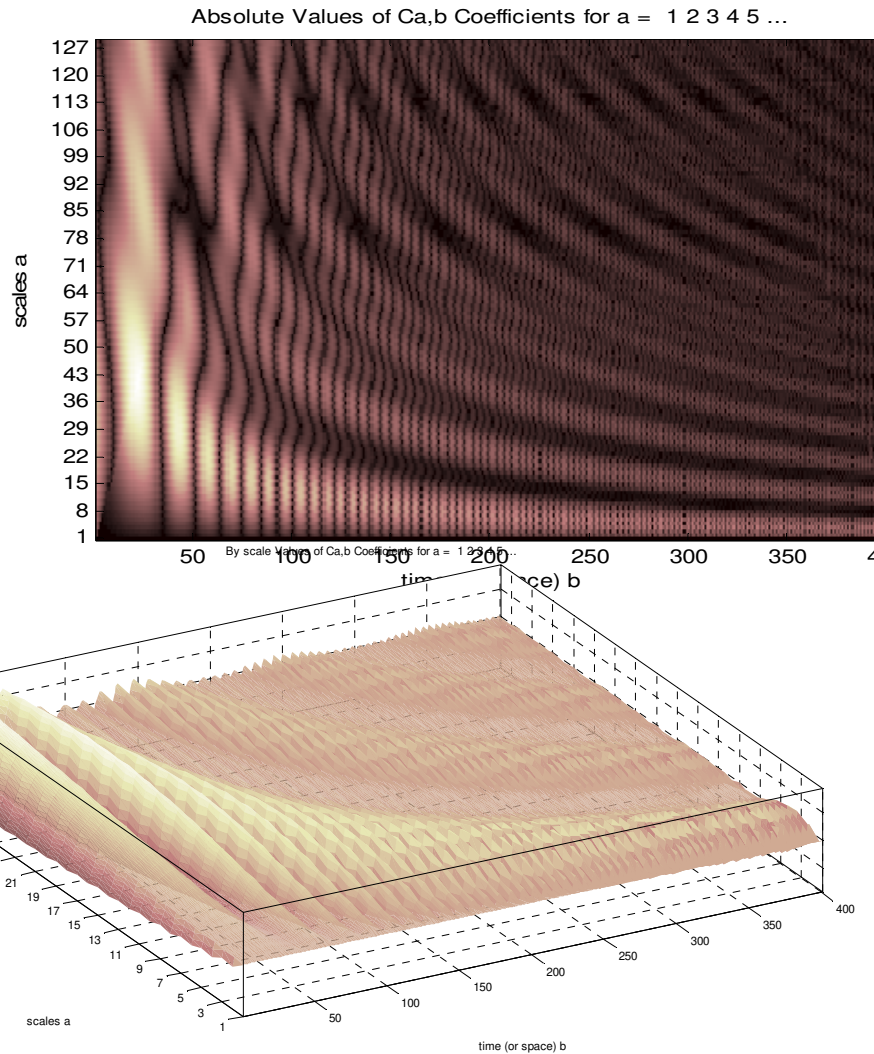
La transformée en ondelettes est un opérateur linéaire, invariant par translation, et par dilatation. Quelle que soit l'échelle et quel que soit l'endroit, l'analyse du signal se fait avec la même fonction. La transformée en ondelettes d'un signal n'est pas unique, elle dépend de l'ondelette mère utilisée. En effet, l'ondelette mère $\psi(t)$ devra avoir une bonne localisation (nulle en dehors d'un certain intervalle), et devra être oscillante le nombre de moments nuls correspond au nombre d'oscillations).

On peut montrer que si la fonction analysante (l'ondelette) est correctement choisie, la transformation en ondelettes est inversible. Le signal $x(t)$ peut être reconstruit après double intégration suivant le facteur d'échelle a et le paramètre de translation b :

$$x(t) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \frac{1}{a^2} WT_{x,\psi}(a, b) \psi_{a,b}^*(t) da db$$

Exemple d'application : Analyse du signal chirp avec l'ondelette de Haar (Voir TP n°6)

Les coefficients de l'ondelette sont représentés en fonction du temps où les valeurs les plus élevées sont de couleur claire. On note qu'un coefficient à une amplitude d'autant plus grande que l'ondelette ressemble au signal sur la portion analysée. Lorsque la fenêtre est étroite (onde étroite), on observe les hautes fréquences et lorsqu'elle est large, ce sont les coefficients des basses fréquences qui sont élevés.



Toutefois bien que la transformation continue en ondelettes permette d'obtenir une bonne localisation temps-fréquence, toutefois, l'information engendrée par la transformation est infiniment redondante (a et b sont continues engendrant une sur analyse). Pour y remédier, on a recours à l'emploi d'ondelettes discrètes dyadiques.

4. Ondelettes discrètes (DWT), ondelettes dyadiques et analyse multi-résolution

Dans les applications, comme le traitement des images, la quantité d'information à traiter peut être importante, il est nécessaire d'optimiser les calculs et la taille des données. La multi-résolution apporte une solution à ce type de problème. Celle-ci est basée sur une transformation discrète qui diffère de la précédente d'un facteur 2 (transformation dyadique). Pour limiter la quantité d'information générée, l'ondelette encode uniquement la différence d'information entre deux résolutions successives. Cette technique, se prête bien à une décomposition/composition par banc de filtre [25]

Dans la plupart des formulations de la DWT (transformée en ondelettes discrètes), les bases respectives sont donc dérivées par une échelle dyadique des bases avec $a=2^{-j}$ et un changement d'unité $b=k 2^{-j}$:

Les valeurs discrètes du facteur d'échelle a et du paramètre de translation b seront considérées sous la forme :

$$a = 2^{-j} \text{ et } b = k 2^{-j}$$

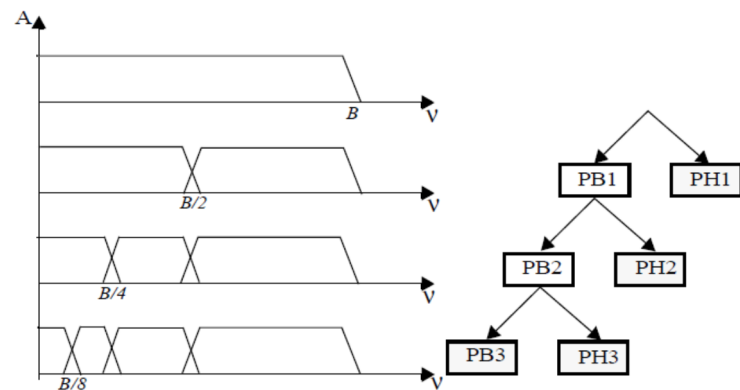
Avec ces valeurs de a et b , l'équation devient : $C_x(2^{-j}, k 2^{-j}) = 2^{\frac{j}{2}} \int_{-\infty}^{+\infty} x(t) \psi(2^j t - k) dt$

Si la fonction $x(t)$ est discrétisée, en supposant une période d'échantillonnage égale à 1, pour des raisons de simplicité, l'équation s'écrit alors : $C_x(2^{-j}, k 2^{-j}) = 2^{\frac{j}{2}} \sum_n x(n) \psi(2^j n - k)$

La construction de telles ondelettes peut s'aborder comme un problème de choix de base de décomposition de signal, mais aussi comme un problème de décomposition en sous-bandes (voir chapitre précédent).

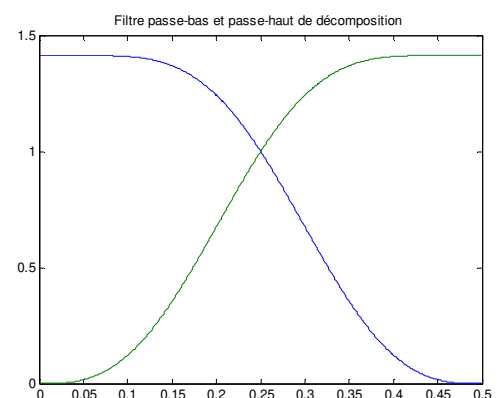
L'idée de l'analyse multirésolution développée par Meyer et Mallat permet d'analyser un signal à différentes échelles à travers des opérateurs linéaires à des niveaux de résolutions correspondant à différentes bandes de fréquences spatiales.

Le signal est décomposé de manière itérative par un banc de filtres. A chaque niveau, le spectre du signal est partagé en deux bandes par un filtre passe-bas et un par filtre passe-haut. Le filtre passe-bas (fonction d'échelle) donne les informations grossières tandis que le filtre passe-haut (ondelettes) encode les détails. Le processus est répété jusqu'à obtenir la résolution souhaitée. L'avantage de cette méthode est de n'utiliser que deux filtres. Quand le processus est arrêté, le signal peut être défini par le jeu de coefficients correspondant [25]: $\{PB3, PH3, PH2, PH1\}$.

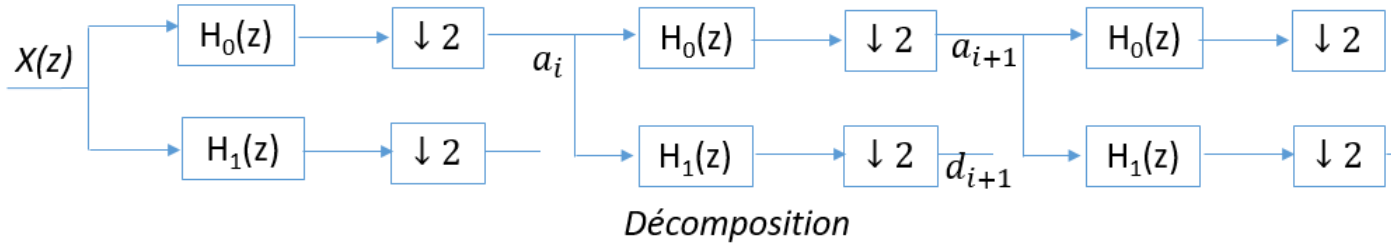


L'ondelette doit être très compacte ainsi, ses coefficients doivent être, pour la plupart, proche de zéro. Cette condition dépend de quelques paramètres comme la régularité de la fonction, le nombre de moments nuls et la taille de son support. La compacité du support impose que les filtres (h_0) qui vont engendrer la fonction d'échelle (donnant une approximation grossière du signal) et l'ondelette (h_1 fournissant les détails) soient à réponse impulsionnelle finie.

Cette condition impose d'utiliser des filtres miroirs conjugués, Les deux filtres sont symétriques conjugués autour de la fréquence centrale. (Passe-bas et passe haut ont la même fréquence de coupure). Les plus connus sont les filtres de Daubechies à support compact. Les filtres en quadrature ont une réponse impulsionnelle finie. Ils utilisent un filtre passe-bas et un filtre passe-haut. Le terme quadrature vient du fait que la somme des modules au carré des deux filtres est constante (2 ou 1) dite condition d'orthogonalité. La bande passante utilisée, par chacun des filtres, est la moitié de la bande initiale.). Ils se coupent en $f_c/4$. On peut dans ce cas, utiliser un décimateur pour réduire la quantité d'information d'un facteur 2 [25].



La transformée en ondelettes discrète sur des bases orthonormées se ramène à des opérations de filtrage numérique suivies de sous échantillonnage. La reconstruction est parfaite et s'effectue également par des filtrages numériques précédés de sur-échantillonnage [27]



La décomposition suivant l'algorithme de Mallat est obtenue [27]:

$$a_{j+1}[n] = \sum_k h_0[2n - k] a_j[k]$$

$$d_{j+1}[n] = \sum_k h_1[2n - k] a_j[k]$$

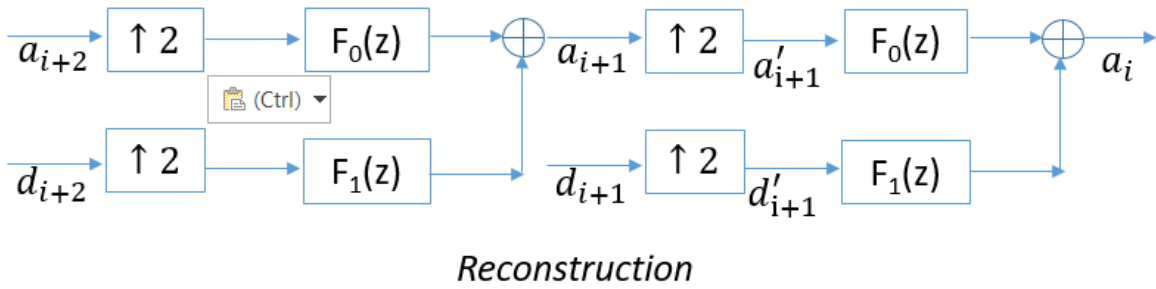
Et la reconstruction suivant l'algorithme de Mallat est obtenue par:

$$a_j[n] = \sum_k \{f_0[n - 2k] a_{j+1}[k] + f_1[n - 2k] d_{j+1}[k]\}$$

Remarque

Si on appelle a'_{j+1} le signal a_{j+1} après interpolation (insertion d'un 0 entre les 2 échantillons successifs) et d'_{j+1} le signal d_{j+1} après interpolation alors [27]:

$$a_j[n] = \sum_k \{f_0[n - k] a'_{j+1}[k] + f_1[n - k] d'_{j+1}[k]\}$$



Nous avons vu au chapitre précédent qu'un banc de filtres à reconstruction parfaite pouvait s'obtenir via les filtres CQF. Pour connaître les filtres $h_0(n)$, $h_1(n)$, $f_0(n)$, et $f_1(n)$, dans le cas de bases orthogonales, il suffit de connaître seulement le filtre $h_0(n)$ puisque les trois autres filtres se calculent à partir de ce dernier (filtres de reconstruction identiques aux filtres d'analyse mais retournés dans le temps).

$$H_1(z) = H_0(-z^{-1})z^{-(L-1)} \quad \text{soit} \quad h_1(n) = (-1)^n h_0(L-1-n)$$

$$F_0(z) = H_1(-z) = H_0(z^{-1})z^{-(L-1)} \quad \text{soit} \quad f_0(n) = h_0(L-1-n)$$

$$F_1(z) = -H_0(-z) \quad \text{soit} \quad f_1(n) = (-1)^n h_0(n)$$

Exemple 1

$$H_0(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}$$

$$\Rightarrow H_1(z) = (b_0 - b_1 z^1 + b_2 z^2 - b_3 z^3) z^{-3} = -b_3 + b_2 z^{-1} - b_1 z^{-2} + b_0 z^{-3}$$

$$\Rightarrow F_0(z) = (b_0 + b_1 z^1 + b_2 z^2 + b_3 z^3) z^{-3} = b_3 + b_2 z^{-1} + b_1 z^{-2} + b_0 z^{-3}$$

$$\Rightarrow F_1(z) = -(b_0 - b_1 z^{-1} + b_2 z^{-2} - b_3 z^{-3}) = b_0 - b_1 z^{-1} + b_2 z^{-2} - b_3 z^{-3}$$

Ainsi, considérons par exemple l'ondelette de Daubechies à 6 coefficients telle que la fonction d'échelle h

soit :

$$\checkmark \quad h_0 = [0.2352 \ 0.5706 \ 0.3252 \ -0.0955 \ -0.0604 \ 0.0249],$$

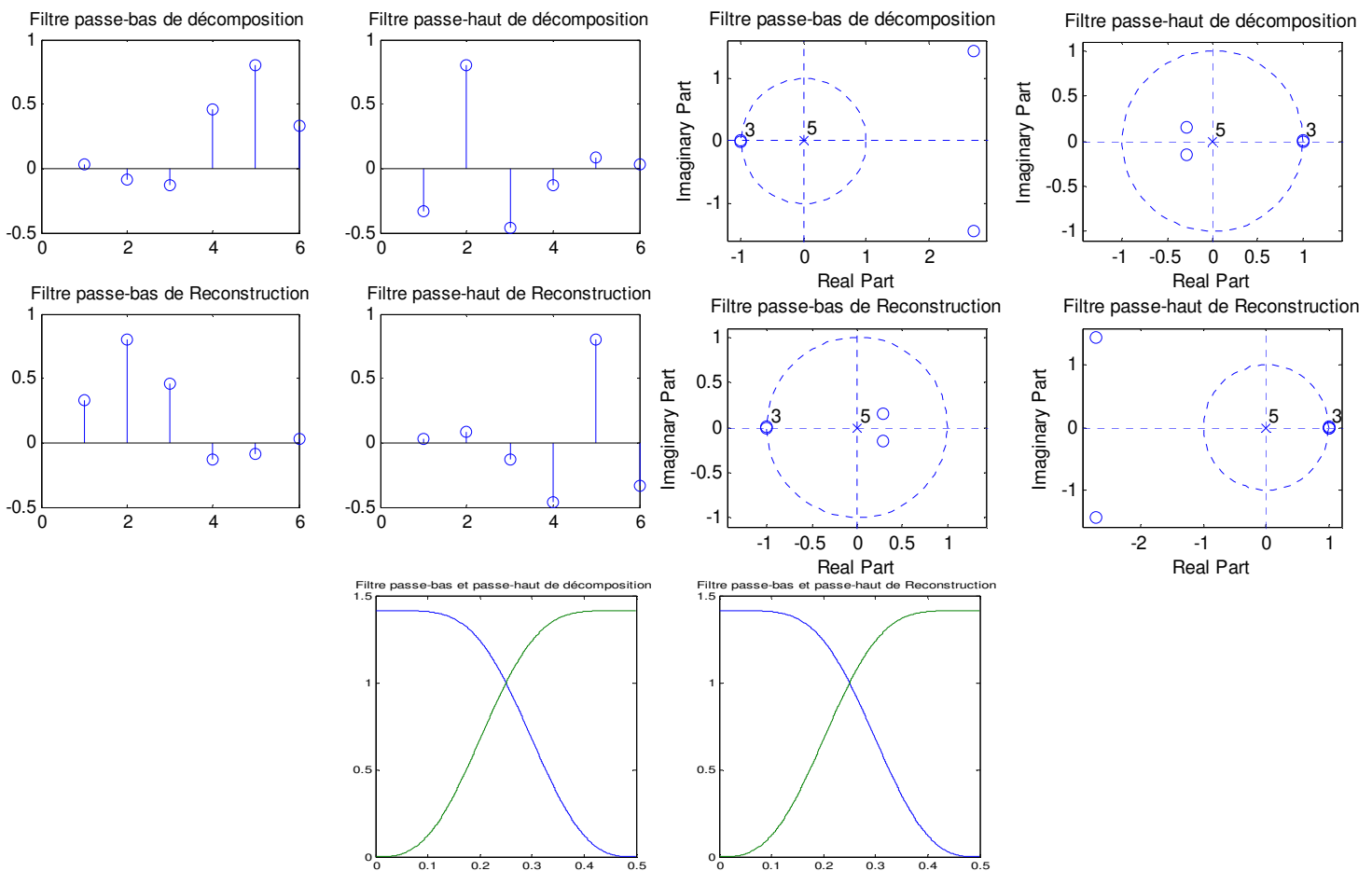
h_1 est obtenu en retournant h dans le temps et en inversant le signe des coefficients impairs, ce qui nous donne :

$$\checkmark \quad h_1 = [-0.0249 \ -0.0604 \ 0.0955 \ 0.3252 \ -0.5706 \ 0.2352]$$

Filtres de reconstruction identiques aux filtres d'analyse mais retournés dans le temps):

$$\checkmark \quad f_0 = [0.0249 \ -0.0604 \ -0.0955 \ 0.3252 \ 0.5706 \ 0.2352], \quad f_1 = [0.2352 \ -0.5706 \ 0.3252 \ 0.0955 \ -0.0604 \ -0.0249]$$

Ci - dessous sont donnés les coefficients d'ondelettes de Daubechies 3 avec le tracé des pôles et zéros ainsi que des réponses en fréquence (Voir TP n°6)



Remarquons que ce sont filtre CQF (vor chapitre précédent) et donc sont orthogonaux [28] :

$$|H_0(f)|^2 + |H_0(f \pm f_e/2)|^2 = 2, \quad |H_1(f)|^2 + |H_1(f \pm f_e/2)|^2 = 2$$

$$|H_0(f)|^2 + |H_1(f)|^2 = 2$$

$$H_0(f)H_1^*(f) + H_0(f \pm f_e/2)H_1^*(f \pm f_e/2) = 0$$

Comme les coefficients ne sont pas symétriques, le déphasage ne sera pas linéaire. Rappelons que la seule ondelette orthogonale qui soit symétrique est l'ondelette de Haar.

Exemple d'application 1 : On suppose que le signal à analyser est une rampe et que l'ondelette est celle de Haar.

Filtres de **Décomposition** $h_0 = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$, $h_1 = \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$, Filtres de **Reconstruction** $f_0 = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$, $f_1 = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}$

Décomposition

$$a_{j+1}[n] = \sum_k h_0[2n - k]a_j[k] = \sum_k f_0[k - 2n]a_j[k] \quad d_{j+1}[n] \\ = \sum_k h_1[2n - k]a_j[k] = \sum_k f_1[k - 2n]a_j[k]$$

	Approximation $h_0[-k] = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} = f_0[k]$	Détail $h_1[-k] = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} = f_1[k]$
Niveau 0	0,1,2,3,4,5,6,7	
Niveau 1	$\frac{1}{\sqrt{2}}, \frac{5}{\sqrt{2}}, \frac{9}{\sqrt{2}}, \frac{13}{\sqrt{2}}$	$-\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}$

Reconstruction:

$$a_j[n] = \sum_k \{f_0[n - k]a_{j+1}[k] + f_1[n - k]d_{j+1}[k]\} = \sum_k \{h_0[k - n]a_{j+1}[k] + h_1[k - n]d_{j+1}[k]\}$$

	Approximation $f_0[-k] = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} = h_0[k]$	Détail $f_1[-k] = \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} = h_1[k]$
Niveau 1	$\frac{1}{\sqrt{2}}, \frac{5}{\sqrt{2}}, \frac{9}{\sqrt{2}}, \frac{13}{\sqrt{2}}$ $0, \frac{1}{\sqrt{2}}, 0, \frac{5}{\sqrt{2}}, 0, \frac{9}{\sqrt{2}}, 0, \frac{13}{\sqrt{2}}$ $\frac{1}{2}, \frac{1}{2}, \frac{5}{2}, \frac{5}{2}, \frac{9}{2}, \frac{9}{2}, \frac{13}{2}, \frac{13}{2}$	$-\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}$ $0, -\frac{1}{\sqrt{2}}, 0, -\frac{1}{\sqrt{2}}, 0, -\frac{1}{\sqrt{2}}, 0, -\frac{1}{\sqrt{2}}$ $-\frac{1}{2}, \frac{1}{2}, -\frac{1}{2}, \frac{1}{2}, -\frac{1}{2}, \frac{1}{2}, -\frac{1}{2}, \frac{1}{2}$
Niveau 0	0,1,2,3,4,5,6,7	

Exemple d'application 2 :

On suppose que le filtrage passe-bas h_0 est un opérateur de moyennage entre 2 coefficients et que le passe-haut h_1 est un opérateur qui encode la demi-différence entre 2 coefficients [cours 2013] tels que $h_0 = \{0.5, 0.5\}$, $h_1 = \{-0.5, 0.5\}$, $f_0 = 2 * \{0.5, 0.5\}$, $f_1 = 2 * \{0.5, -0.5\}$

Signal original : 11, 9, 5, 7

Décomposition

	Approximation $h_0[-k]=\{0.5, 0.5\}$	Détail $h_1[-k]=\{0.5, -0.5\}$
Niveau 1	10, 6	1, -1
Niveau 2	8	2
Signal décomposé	8, 2, 1, -1	

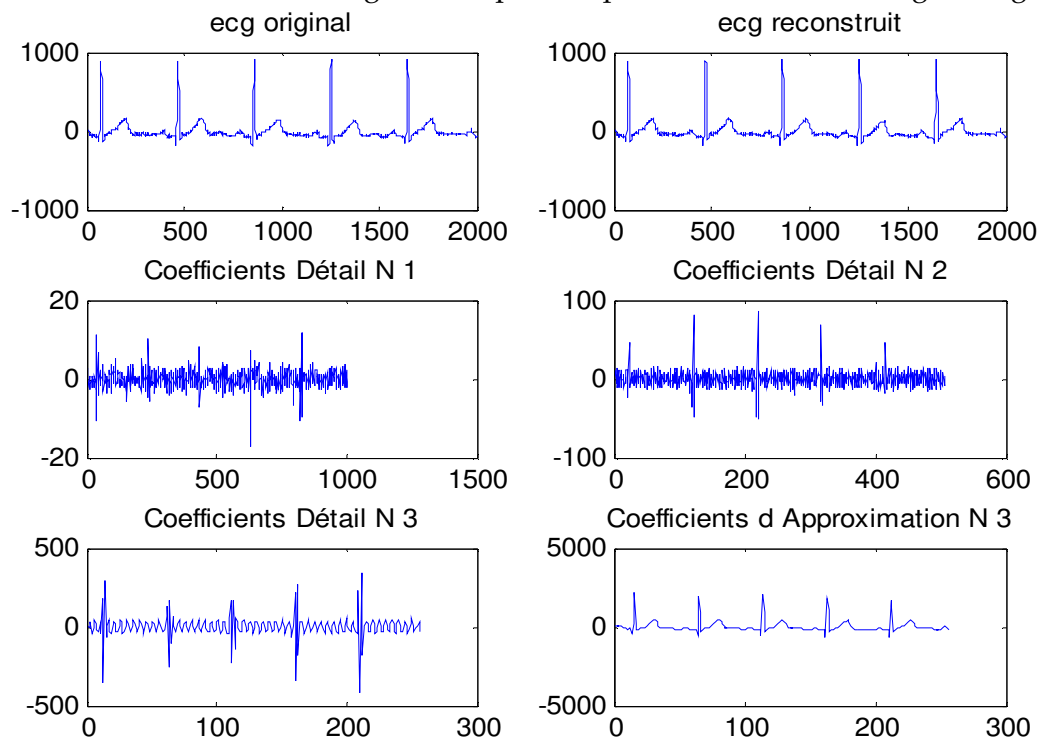
Reconstruction

	Approximation $f_0[-k]=\{1, 1\}$	Détail $f_1[-k]=\{-1, 1\}$
Niveau 2	8 0, 8, 0 [8, 8]	2 0, 2, 0 [2, -2]
Niveau 1	10, 6 0, 10, 0, 6, 0 [10, 10, 6, 6]	1, -1 0, 1, 0, -1 [1, -1, -1, 1]
Signal reconstruit	11, 9, 5, 7	

L'opérateur h_0 donne une représentation approximative alors que h_1 donne une représentation plus précise (détail). A la fin du processus, on garde les détails et la dernière approximation, on commençant par le dernier niveau, on obtient : $S_D = \{8, 2, 1, -1\}$.

Il est possible de reconstruire le signal à partir des éléments de la décomposition en sommant le résultat de l'application des filtres f_0 et f_1 .

Exemple 3 Voici le résultat de la décomposition du signal ecg, ne sont gardés à la fin que la dernière approximation et tous les niveaux de détails grâce auxquels on pourra reconstruire le signal original (TP n°6)



Remarques :

- Les filtres de Daubechies conduisent à des filtres à réponse impulsionnelle finie (RIF) mais ne sont pas linéaires en phase.
- On peut construire des bases d'ondelettes conduisant à des filtres RIF linéaires en phase telles les B splines. Il s'agit des bases biorthogonales où familles de décomposition et reconstruction sont différentes et orthogonales uniquement entre elles [29]:

$$H_0(f)F_0^*(f) + H_0(f + f_e/2)F_0^*(f + f_e/2) = 2 \quad \text{et} \quad H_1(f)F_1^*(f) + H_1(f + f_e/2)F_1^*(f + f_e/2) = 2$$

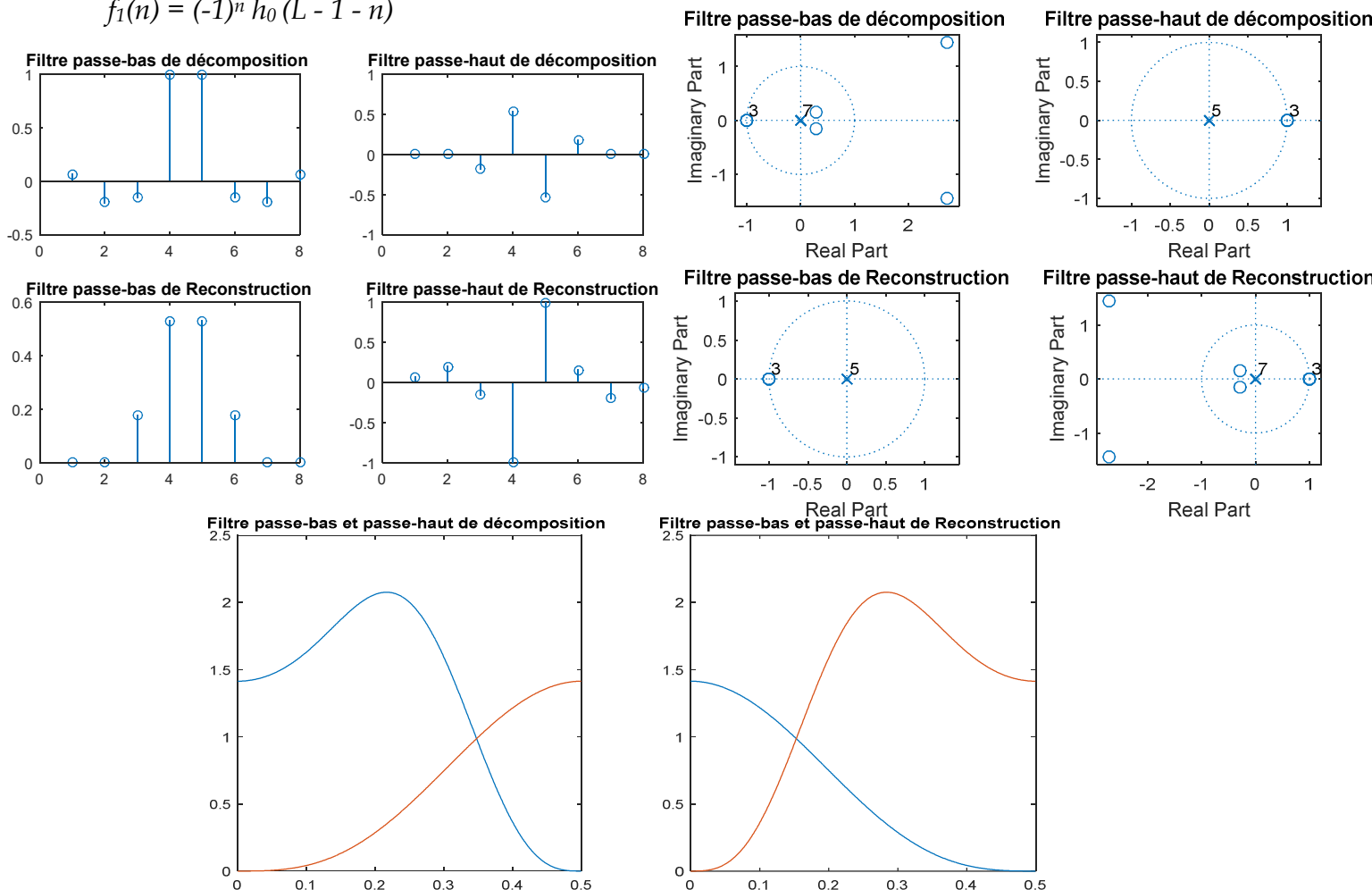
$$H_0(f)F_1^*(f) + H_0(f + f_e/2)F_1^*(f + f_e/2) = 0 \quad \text{et} \quad H_1(f)F_0^*(f) + H_1(f + f_e/2)F_0^*(f + f_e/2) = 0$$
- En traitement d'images, on préférera les filtres bi-orthogonaux (JPEG2000).

Exemple d'ondelettes biorthogonales

Les filtres passe-bas et passe haut de décomposition sont différents tout comme les filtres de reconstruction. Cependant les filtres de décomposition et reconstruction sont duaux. En effet, on obtient le filtre Passe-haut de reconstruction $F_1(z)$ en retournant dans le temps le passe-bas de décomposition $H_0(z)$ et en inversant le signe d'un coefficient sur deux. De même pour le filtre passe-bas de reconstruction $F_0(z)$ en retournant dans le temps le passe-bas de décomposition $H_1(z)$ et en inversant le signe d'un coefficient sur deux, soient :

$$h_1(n) = (-1)^n f_0(L - 1 - n)$$

$$f_1(n) = (-1)^n h_0(L - 1 - n)$$



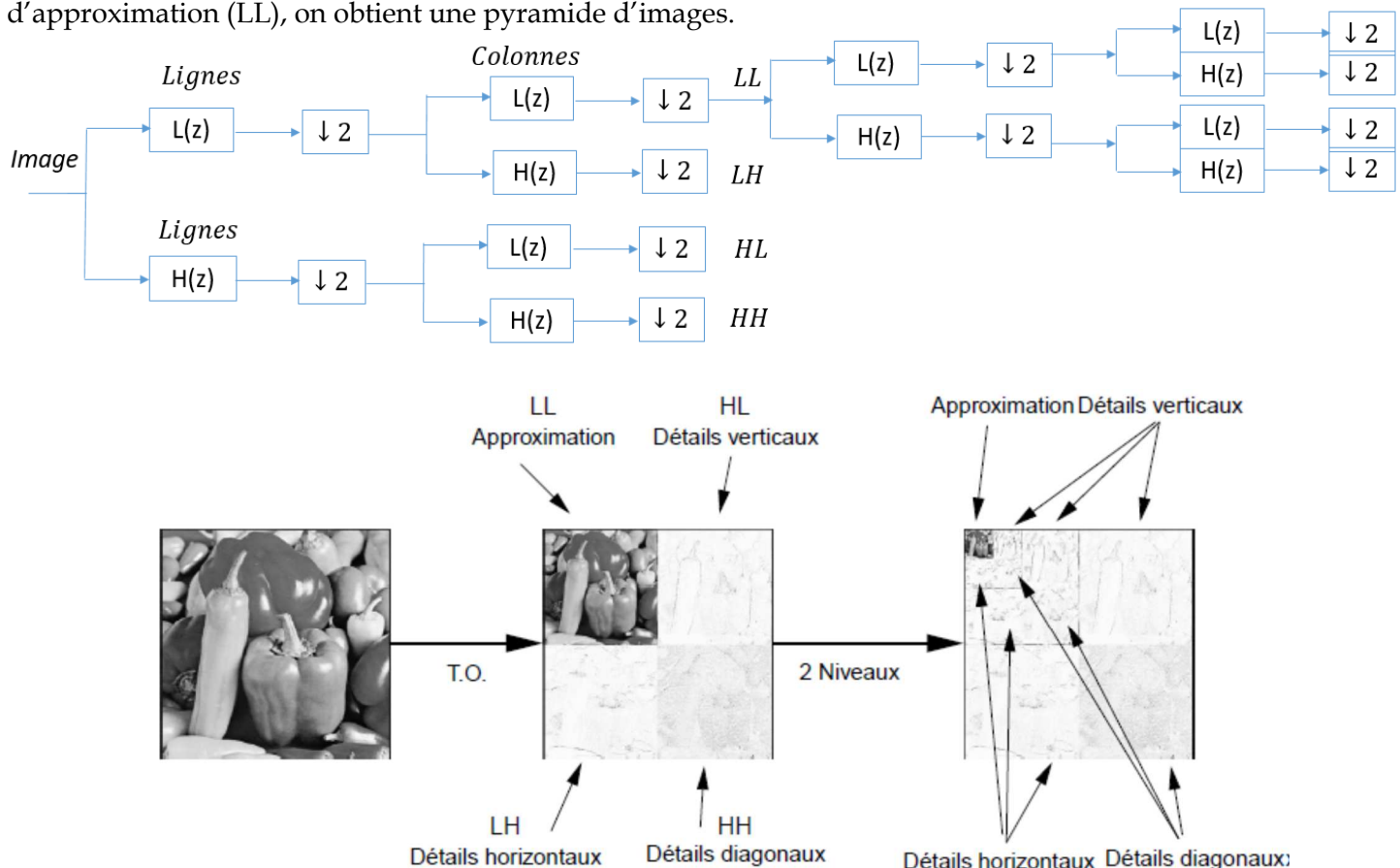
On notera la symétrie des réponses impulsionnelles des filtres passe-bas et l'anti-symétrie des filtres passe-haut d'analyse et de reconstruction conduisant donc à un déphasage linéaire. On notera également la dualité entre les filtres d'analyse et de reconstruction.

Remarque : Le lifting scheme permet une implantation très simple des décompositions en ondelettes et de leurs opérations inverses en employant la factorisation polyphases [21]

5. Exemples d'application de la DWT sur une image

Les ondelettes sont bien adaptées au débruitage des signaux. Le principe est simple, on décompose le signal bruité et l'on force à zéro les coefficients qui ne passent pas un seuil. Puis l'on reconstruit le signal (Voir TP n°6). De la même façon on peut compresser le signal.

Pour une image, on appliquera deux filtres le long des lignes l'un passe-haut (H), l'autre passe-bas (L), ensuite, on ne considère qu'une colonne sur deux et on applique à nouveau les 2 filtres. En ne considérant qu'une ligne sur deux, on obtient 4 images de taille réduite de moitié. En réitérant ce processus sur l'image d'approximation (LL), on obtient une pyramide d'images.



Cet exemple donne le résultat d'une compression jpeg et jpeg2000 (Ondelettes) avec un bpp de 0.2



TD n°4 : Transformées Temps-Fréquence et Temps-Echelle

- Déterminer la TFCT du signal suivant (h est une fenêtre porte de largeur T):

$$x(t) = \begin{cases} e^{2\pi j f_1 t} & \text{pour } t < t_0 \\ e^{2\pi j f_2 t} & \text{pour } t > t_0 \end{cases}$$

- Calculer la transformée de Wigner-Ville du signal $x(t) = A_1 e^{2\pi j f_1 t} + A_2 e^{2\pi j f_2 t}$

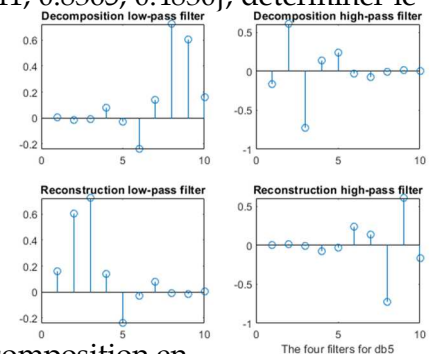
- Soient les ondelettes suivantes :

- La fonction de Haar $\psi(t) = \begin{cases} 1 & \text{si } 0 \leq t \leq \frac{1}{2} \\ -1 & \text{si } \frac{1}{2} \leq t \leq 1 \\ 0 & \text{ailleurs} \end{cases}$

- L'ondelette chapeau mexicain $\psi(t) = \alpha(1 - t^2)e^{-\frac{t^2}{2}}$

Calculer leur TF et tracer la

- Soit le filtre passe bas de décomposition suivant : $h_0 = \{-0.1294, 0.2241, 0.8365, 0.4830\}$, déterminer le filtre passe-haut de décomposition et les filtres de reconstruction



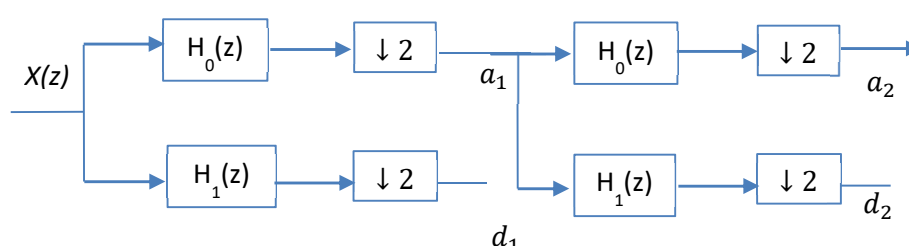
- Soient les filtres de décomposition et reconstruction suivants, vérifier les propriétés liant les 4 filtres

- Soit le signal suivant $x = \{2, 4, 8, 12, 14, 0, 2, 1\}$ donner et tracer sa décomposition en ondelettes si la fonction d'échelle est $h_0 = [0.5, 0.5]$

- Soit le signal suivant : $f = \{4, 6, 10, 12, 8, 6, 5, 5\}$

- Donner et tracer sa décomposition au niveau 1 en ondelettes en employant l'ondelette de Haar.
- Reconstruire le signal à partir du niveau 1
- Calculer son énergie et puis celle après décomposition.
- Vérifier qu'elle se conserve et que 98% de cette énergie se trouve dans le signal approximé.
- Donner la décomposition aux niveaux 2 puis 3.

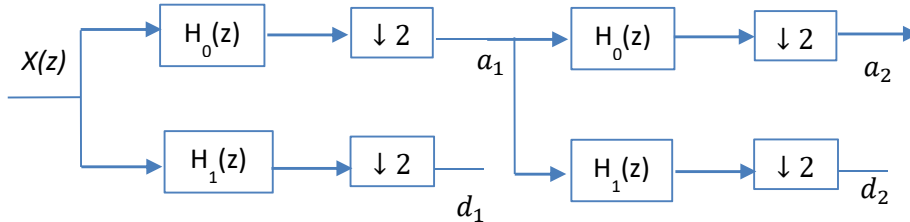
- On décompose un signal $x(n)$ par l'ondelette de Haar $h_0 = \{\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\}$ suivant le schéma ci-dessous. Les 8 premières valeurs du signal $x(n)$ sont $=\{4, 4, 6, 8, 8, 10, 9, 11\}$



- Déterminer a_1, d_1, a_2, d_2 , (**Calcul détaillé**) puis en déduire le signal décomposé.
- En supposant que $f_c = 8\text{kHz}$, donner les bandes de fréquences occupées par a_1, d_1, a_2, d_2 ,
- Donner le schéma de reconstruction et l'expression de filtres $F_0(z)$ et $F_1(z)$.

9. On a décomposé un signal $x(n)$ par l'ondelette de Haar $h_0 = \{\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\}$ suivant le schéma ci-dessous.

Les 8 premières valeurs du signal **décomposé** sont $=\{4, 4, -1, 1, 2/\sqrt{2}, -1/\sqrt{2}, 0, 1/\sqrt{2}\}$



- Donner le schéma de reconstruction et l'expression des filtres $F_0(z)$ et $F_1(z)$.
- En supposant que $f_c = 18\text{kHz}$, donner les bandes de fréquences occupées par a_1, d_1, a_2, d_2 ,
- Identifier a_1, d_1, a_2, d_2 puis **reconstruire** le signal original

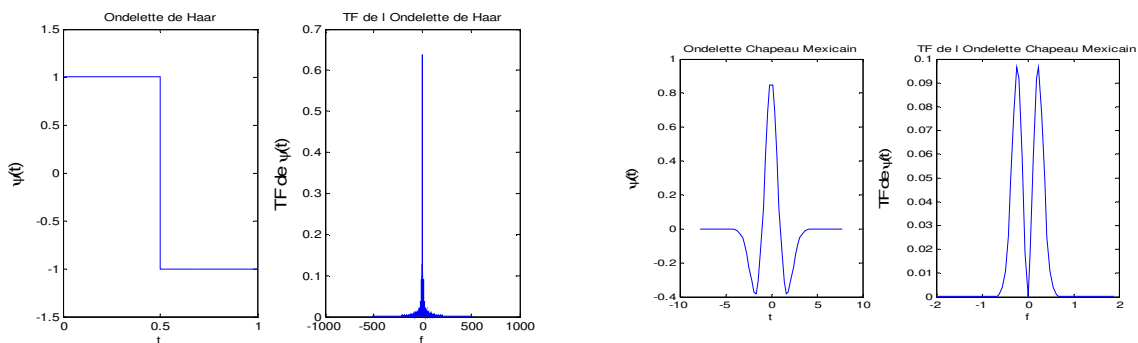
Solutions

1.

$$x(t) = \begin{cases} Tsinc((f-f_1)T)e^{-2\pi j(f-f_1)t} & \text{pour } t + \frac{T}{2} < t_0 \\ Tsinc((f-f_2)T)e^{-2\pi j(f-f_2)t} & \text{pour } t - \frac{T}{2} > t_0 \\ \left(t_0 - t + \frac{T}{2}\right) sinc\left(\left(t_0 - t + \frac{T}{2}\right)(f-f_1)T\right)e^{-2\pi j(f-f_1)\left(\frac{t_0+t}{2} - \frac{T}{4}\right)} + \\ \left(\frac{T}{2} - t_0 + t\right) sinc\left(\left(\frac{T}{2} - t_0 + t\right)(f-f_2)T\right)e^{-2\pi j(f-f_2)\left(\frac{t_0+t}{2} + \frac{T}{4}\right)} & \text{pour } -\frac{T}{2} < t - t_0 < \frac{T}{2} \end{cases}$$

2. $WV_x(t, f) = |A_1|^2 \delta(f - f_1) + |A_2|^2 \delta(f - f_2) + 2A_1 A_2 \cos(2\pi(f_1 - f_2)) \delta\left(f - \frac{(f_1 + f_2)}{2}\right)$

3. Haar $\hat{\psi}(f) = j \frac{\sin^2(\frac{\pi f}{2})}{\frac{\pi f}{2}} e^{-\pi j f}$ Chapeau mexicain $\hat{\psi}(f) = \beta 4\pi^2 f^2 e^{-\frac{f^2}{2}}$



4. $h_1 = \{-0.4830, 0.8365, -0.2241, -0.1294\}$, $f_0 = \{0.4830, 0.8365, 0.2241, -0.1294\}$,
 $f_1 = \{-0.1294, -0.2241, 0.8365, -0.4830\}$

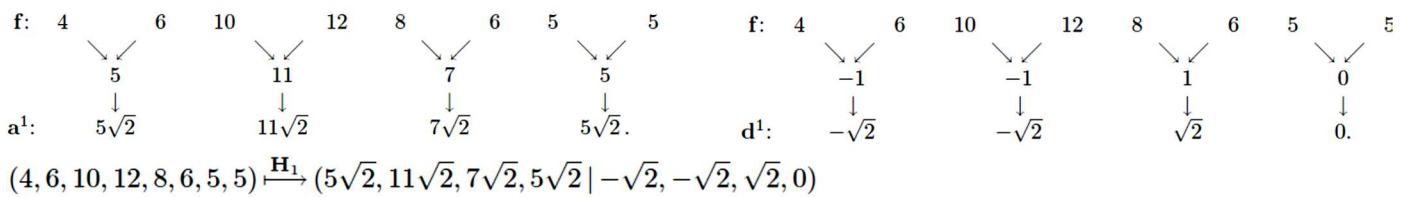
5.

Niveau	Approximation	Détails
0	2 4 8 12 14 0 2 1	
1	3 10 7 1.5	-1 -2 7 0.5
2	6.5 4.25	-3.5 2.75
3	5.375	1.125

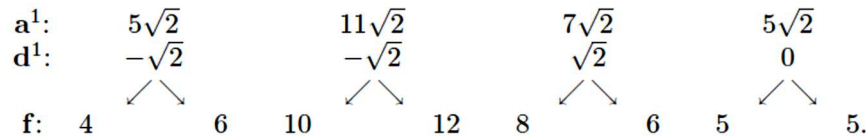
Signal décomposé : $\{5.375, 1.125, -3.5, 2.75, -1, -2, 7, 0.5\}$

7. Filtres de **Décomposition** $h_0 = \left[\frac{1}{\sqrt{2}} \quad \frac{1}{\sqrt{2}} \right]$, $h_1 = \left[-\frac{1}{\sqrt{2}} \quad \frac{1}{\sqrt{2}} \right]$, Filtres de **Reconstruction** $f_0 = \left[\frac{1}{\sqrt{2}} \quad \frac{1}{\sqrt{2}} \right]$, $f_1 = \left[\frac{1}{\sqrt{2}} \quad -\frac{1}{\sqrt{2}} \right]$

Décomposition



Reconstruction

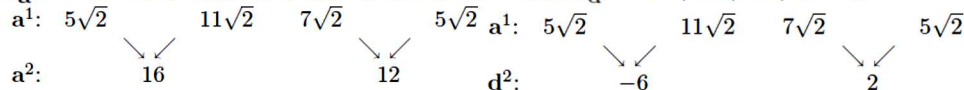


Energie

$$\mathcal{E}_f = 4^2 + 6^2 + \dots + 5^2 = 446.$$

$$\mathcal{E}_{(a^1 | d^1)} = 25 \cdot 2 + 121 \cdot 2 + \dots + 2 + 0 = 446$$

$$\mathcal{E}_{a^1} = 25 \cdot 2 + 121 \cdot 2 + 49 \cdot 2 + 25 \cdot 2 = 440 \quad \mathcal{E}_{d^1} = 2 + 2 + 2 + 0 = 6$$



$$(a^2 | d^2 | d^1) = (16, 12 | -6, 2 | -\sqrt{2}, -\sqrt{2}, \sqrt{2}, 0)$$

$$(a^3 | d^3 | d^2 | d^1) = (14\sqrt{2} | 2\sqrt{2} | -6, 2 | -\sqrt{2}, -\sqrt{2}, \sqrt{2}, 0)$$

88% de l'énergie est contenue dans a³ représentant pourtant le 8^{ème} du signal

8. Examen 19/20 9. Ratt 19/20

Exercices Supplémentaires

1. Wigner-Ville – Signal gaussien

$$z(t) = \frac{1}{\sqrt{\sigma}} e^{-\pi t^2 / \sigma^2} \longrightarrow W_z(t, f) = \sqrt{2} e^{-2\pi(t^2 / \sigma^2 + \sigma^2 f^2)}$$

– Signal monochromatique d'enveloppe gaussienne

$$z(t) = \frac{1}{\sqrt{\sigma}} e^{-\pi(t-t_0)^2 / \sigma^2} e^{2i\pi f_0 t} \longrightarrow W_z(t, f) = \sqrt{2} e^{-2\pi\alpha(t-t_0)^2 / \sigma^2 - 2\pi\sigma^2(f-f_0)^2}$$

– Chirp idéal

$$z(t) = e^{i\pi\beta t^2 + 2i\pi f_0 t} \longrightarrow W_z(t, f) = \delta(f - \beta t - f_0)$$

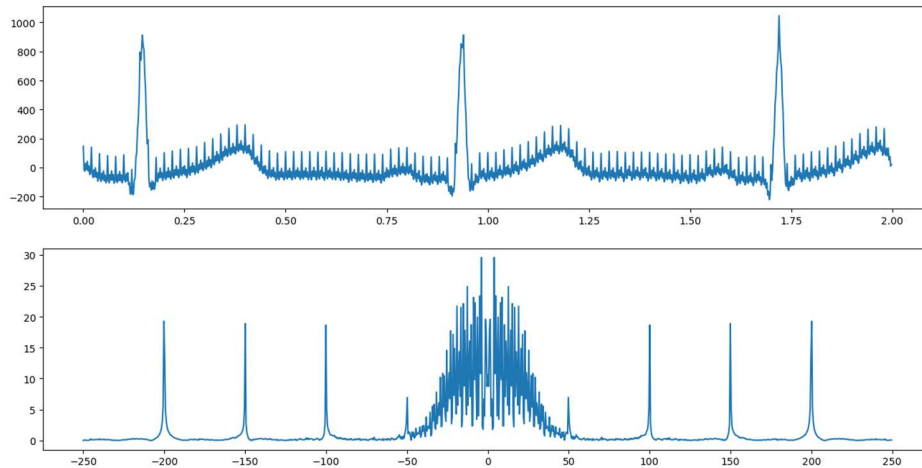
2. Soit le signal suivant : x(n)={0, 1, -2, 3, -4, 5, -6, 7 }

- Donner sa décomposition en ondelettes au niveau 1 en employant l'ondelette de Haar tel que le filtre passe-bas de décomposition soit $h_0 = \{\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\}$
- Donner le schéma de reconstruction permettant de passer d'un niveau à un autre.
- Reconstruire le signal à partir du niveau 1.
- Donner le signal décomposé aux niveaux 2 puis 3.
- Vérifier que l'énergie se conserve à tous les niveaux (1, 2 et 3)
- Proposer un moyen de compresser ce signal.

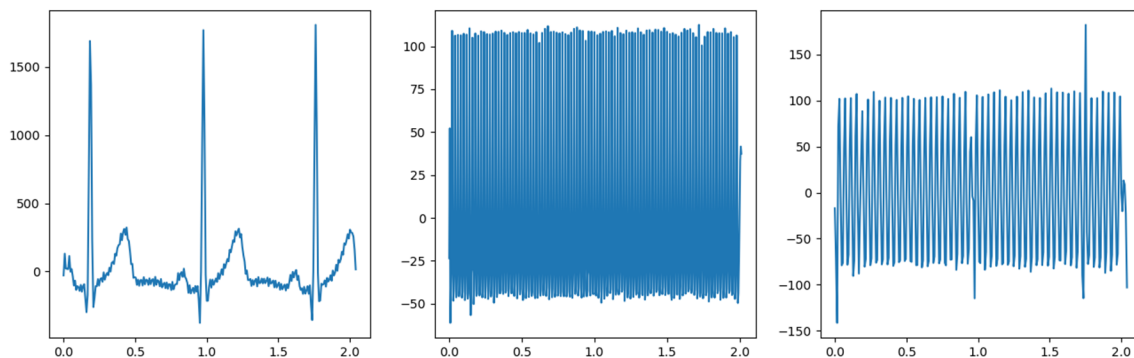
3. Soit le signal suivant : x(n)={11, 9, 5, 7, 5, 11, 7, 9}

- Citer un inconvénient de l'ondelette continue.
- Donner le schéma de décomposition en ondelettes discrètes pour 2 niveaux.

- Donner le schéma correspondant pour la répartition en fréquences.
 - Donner sa décomposition en ondelettes au niveau 2 en employant les filtres suivants :
 $h_0 = \{0.5, 0.5\}$, $h_1 = \{-0.5, 0.5\}$, $f_0 = 2 * \{0.5, 0.5\}$, $f_1 = 2 * \{0.5, -0.5\}$
 - L'énergie se conserve-t-elle ?
 - Reconstruire le signal à partir du niveau 2.
4. A la transmission d'un signal ecg, il a été affecté d'un bruit issu de la combinaison de 3 sinusoïdes (voir figure ci-dessous : signal bruité + Module de sa TFD).



En analysant le signal par les ondelettes discrètes dyadiques, nous avons obtenu les graphes suivant:



- Identifier pour chacun des 3 signaux : le signal (approximé ou détail), le niveau de décomposition
- Pour chacun des 3 signaux, esquisser le module de la TFD
- Proposer une solution pour supprimer le bruit (explication détaillée)

Solutions :

2. examen 17/18

3. Ratt 17/18

4. Examen 20/21

TP n°6 : De la TFCT, Wigner-Ville aux ondelettes

Ce TP a pour objectif de:

- Se familiariser avec les représentations temps-fréquence et temps-échelle
- Aborder la DWT, les notions de bancs de filtre à travers les filtres de décomposition et synthèse
- Appliquer la DWT au débruitage

1. Démo

```
# -*- coding: utf-8 -*-
import numpy as np; import matplotlib.pyplot as plt; import scipy.signal as sp;
NF = 1024; Te=0.005; fe=1/Te; N=round(2/Te);
t = np.linspace(0, 2, N);
x1= sp.chirp(t, f0=50, t1=2, f1=1, method='linear');
x2= sp.chirp(t, f0=1, t1=2, f1=50, method='linear');

""" TFD """
TFx1 = np.fft.fft(x1,NF); TFx1 = np.fft.fftshift(TFx1);
TFx2 = np.fft.fft(x2,NF); TFx2 = np.fft.fftshift(TFx2); freq = np.arange(-NF/2,NF/2)*fe/NF;
plt.figure(1)
plt.subplot(221); plt.plot(t, x1); plt.title('Chirp de 50 à 1 Hz'); plt.subplot(222); plt.plot(freq, np.abs(TFx1));
plt.subplot(223); plt.plot(t, x2); plt.title('Chirp de 1 à 50 Hz'); plt.subplot(224); plt.plot(freq, np.abs(TFx2));

"""Spectrogramme : TFCT"""
f, tt, Sxx1 = sp.spectrogram(x1, fe, nperseg=100, noverlap=20);
f, tt, Sxx2 = sp.spectrogram(x2, fe, nperseg=100, noverlap=20);
plt.figure(2)
plt.subplot(221); plt.plot(t, x1); plt.subplot(222); plt.pcolormesh(tt, f, Sxx1, shading='auto')
plt.ylabel('Frequence [Hz]'); plt.xlabel('Temps [sec]'); plt.title('Spectrogramme Tfen=100, TRec=20');
plt.subplot(223); plt.plot(t, x2); plt.subplot(224); plt.pcolormesh(tt, f, Sxx2, shading='auto')
plt.ylabel('Frequence [Hz]'); plt.xlabel('Temps [sec]'); plt.title('Spectrogramme Tfen=100, TRec=20');
f, tt, Sxx1 = sp.spectrogram(x1, fe, nperseg=20, noverlap=4);
f, tt, Sxx2 = sp.spectrogram(x2, fe, nperseg=20, noverlap=4);
plt.figure(3)
plt.subplot(221); plt.plot(t, x1); plt.subplot(222); plt.pcolormesh(tt, f, Sxx1, shading='auto')
plt.ylabel('Frequence [Hz]'); plt.xlabel('Temps [sec]'); plt.title('Spectrogramme Tfen=20, TRec=4');
plt.subplot(223); plt.plot(t, x2); plt.subplot(224); plt.pcolormesh(tt, f, Sxx2, shading='auto')
plt.ylabel('Frequence [Hz]'); plt.xlabel('Temps [sec]'); plt.title('Spectrogramme Tfen=20, TRec=4');

"""Wigner-Ville """
from tftb.processing import WignerVilleDistribution
WVx1=WignerVilleDistribution(x1,timestamps=t)
WVx2=WignerVilleDistribution(x2,timestamps=t)
tfr_x1, t_x1, f_x1 = WVx1.run()
tfr_x2, t_x2, f_x2 = WVx2.run()
plt.figure(4)
plt.subplot(221); plt.plot(t, x1); plt.subplot(222); plt.pcolormesh(t_x1, f_x1*fe, np.abs(tfr_x1), shading='auto')
plt.ylabel('Frequence [Hz]'); plt.xlabel('Temps [sec]'); plt.title('Wigner-Ville');
plt.subplot(223); plt.plot(t, x2); plt.subplot(224); plt.pcolormesh(t_x2, f_x2*fe, np.abs(tfr_x2), shading='auto')
plt.ylabel('Frequence [Hz]'); plt.xlabel('Temps [sec]'); plt.title('Wigner-Ville');

"""Ondelettes continues """
import pywt
Larg = np.arange(1, 31);
cwtmatr,fs = pywt.cwt(x2, Larg, 'mexh')
plt.figure(5)
plt.imshow(cwtmatr, aspect='auto', vmax=abs(cwtmatr).max(), vmin=-abs(cwtmatr).max())

""" Ondelettes Discrètes : Filtres d'analyse et Synthèse """
ondelettes = pywt.Wavelet('haar')
plt.figure(6)
plt.subplot(221); plt.stem(ondelettes.dec_lo); plt.title('Passe-bas de Décomposition');
FGE, USTHB [ assiakourgli@gmail.com / http://perso.usthb.dz/~akourgli/
```

```
plt.subplot(222); plt.stem(ondelettes.dec_hi); plt.title('Passe-Haut de Décomposition');
plt.subplot(223); plt.stem(ondelettes.rec_lo); plt.title('Passe-bas de Reconstruction');
plt.subplot(224); plt.stem(ondelettes.rec_hi); plt.title('Passe-bas de Reconstruction');
L=512;
```

```
""" Application de la DWT : Décomposition et Reconstruction """
x = np.genfromtxt('ecg.dat');
coeffs = pywt.wavedec(x, 'db5', level=3)
cA3, cD3, cD2, cD1 = coeffs
plt.figure(7)
plt.subplot(321); plt.plot(x); plt.title('Signal original');
plt.subplot(322); plt.plot(cD1); plt.title('Détail niveau 1');
plt.subplot(323); plt.plot(cD2); plt.title('Détail niveau 2');
plt.subplot(324); plt.plot(cD3); plt.title('Détail niveau 3');
plt.subplot(325); plt.plot(cA3); plt.title('Approximation niveau 3');

x_reconst=pywt.waverec([cA3, cD3, cD2,], 'db5');
plt.subplot(326); plt.plot(x_reconst); plt.title('Signal Reconstruit');
```

2. A faire