

Traitement Numérique du Signal

Michel Terré

michel.terre@cnam.fr

Table des Matières

1. ECHANTILLONNAGE ET QUANTIFICATION	7
1.1 FORMALISATION DE L'ECHANTILLONNAGE.....	7
1.2 PERIODISATION DU SPECTRE	9
1.3 LE THEOREME DE RECONSTRUCTION.....	16
1.4 QUANTIFICATION	19
2. LES TRANSFORMEES.....	21
2.1 TRANSFORMEE DE FOURIER DISCRETE	21
2.1.1 <i>Définition</i>	21
2.1.2 <i>Egalité de Parseval</i>	22
2.1.3 <i>Interprétation au sens des moindres carrés</i>	22
2.2 TRANSFORMEE DE FOURIER RAPIDE (FFT).....	23
2.3 TRANSFORMEE EN Z.....	28
3. FILTRAGE NUMERIQUE.....	29
3.1 LES SYSTEMES LINEAIRES DISCRETS INVARIANTS DANS LE TEMPS	29
3.1.1 <i>Définition</i>	29
3.1.2 <i>Conditions de stabilité</i>	30
3.2 FONCTIONS PROPRES DES SYSTEMES LINEAIRES INVARIANTS DANS LE TEMPS	31
3.3 SYSTEMES LINEAIRES INVARIANTS DANS LE TEMPS REGIS PAR UNE EQUATION AUX DIFFERENCES.....	31
3.4 FILTRES A REPONSE IMPULSIONNELLE FINIE (RIF)	33
3.4.1 <i>Propriété de phase linéaire</i>	36
3.4.2 <i>Synthèse des filtres à réponse impulsionnelle finie</i>	38
3.4.3 <i>Relations entre le nombre de coefficients et le gabarit</i>	40
3.5 FILTRES A REPONSE IMPULSIONNELLE INFINIE (RII)	41
3.5.1 <i>Cellule purement récursive</i>	41
3.5.2 <i>Cellule du deuxième ordre</i>	42
3.5.3 <i>Cellule générale du second ordre</i>	43
3.5.4 <i>Synthèse des filtres à réponse impulsionnelle infinie</i>	44

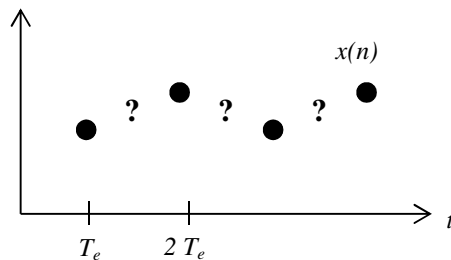
3.6	RELATIONS ENTRE LE NOMBRE DE COEFFICIENTS ET LE GABARIT	47
3.7	APPLICATIONS DU FILTRAGE NUMERIQUE	49
3.7.1	<i>Décimation</i>	49
3.7.2	<i>Interpolation</i>	49
3.7.3	<i>Filtres de Nyquist</i>	51
4.	LES SIGNAUX ALEATOIRES	55
4.1	INTRODUCTION	55
4.2	DENSITE SPECTRALE DE PUISSANCE	59
4.2.1	<i>Problématique de l'Analyse spectrale</i>	59
4.2.2	<i>Estimation Spectrale non paramétrique</i>	60
4.3	LE BRUIT BLANC	66
4.4	EXERCICES CORRIGES	66
5.	PREDICTION LINEAIRE	71
5.1	INTRODUCTION	71
5.2	RESOLUTION A L'ORDRE 1	72
5.3	RESOLUTION A L'ORDRE 2	73
5.4	RESOLUTION A L'ORDRE N.....	74
6.	FILTRAGE ADAPTATIF	77
6.1	INTRODUCTION GENERALE	77
6.2	SOLUTION DE WIENER	78
6.3	INTRODUCTION DU FACTEUR D'OUBLI	82
6.4	ALGORITHME DU GRADIENT (LMS)	84
6.5	ALGORITHME RLS	90
6.6	EXERCICES CORRIGES	92
7.	BIBLIOGRAPHIE	95
8.	ANNEXES	96
8.1	ANNEXE DERIVATION FORME BILINEAIRE.....	96
8.2	ANNEXE, METHODE DE CAPON	97

8.3	ANNEXE, METHODE DE PISARENKO:.....	99
8.4	ANNEXE, METHODE DE PRONY:.....	102
8.5	ANNEXE, ALGORITHME DE LEVINSON :.....	104
8.6	ANNEXE, NEURAL NETWORK AND BACKPROPAGATION	110
9.	EXERCICES	116
9.1	ECHANTILLONNAGE.....	116
9.2	TRANSFORMEE DE FOURIER	118
9.3	FILTRAGE NUMERIQUE RIF	120
9.4	FILTRAGE NUMERIQUE RII.....	122
9.5	SIGNAL ALEATOIRE.....	124
9.6	PREDICTION LINEAIRE.....	126
9.7	ANALYSE SPECTRALE	127
9.8	ESTIMATION.....	128

1. Échantillonnage et quantification

Si on note $x_a(t)$ les valeurs prises au cours du temps t par un signal analogique, l'échantillonnage de ce dernier au rythme d'une période d'échantillonnage T_e , revient à ne disposer, finalement, des valeurs de ce signal, qu'aux instants multiples de T_e . Le signal (ou suite) numérique se note alors $x(n) = x_a(nT_e)$.

La première question qui se pose naturellement est de savoir si on n'a pas perdu de l'information en ne disposant plus des valeurs du signal entre deux instants d'échantillonnage.



Une autre façon de formuler cette question serait : "est-il possible de reconstruire $x_a(t)$ à partir des échantillons $x(n)$?" C'est là l'objet du théorème de reconstruction. Très intuitivement on peut se dire que si on était sûr que le signal "varie très lentement", alors entre deux instants d'échantillonnage, il ne pourrait pas faire grand-chose d'autre que d'aller "tranquillement" d'un point à un autre. Après formalisation on arrivera à écrire cette "variation lente" du signal par une contrainte sur son spectre, ce qui va conduire au théorème de l'échantillonnage en général appelé théorème de Shannon.

1.1 Formalisation de l'échantillonnage

La formalisation de l'opération d'échantillonnage est malheureusement assez délicate avec la notion mathématique habituelle de fonction. Elle s'effectue par contre de manière simple et concise par l'intermédiaire de la théorie des distributions, développée par le mathématicien Laurent Schwartz¹. Le processus d'échantillonnage est ainsi représenté mathématiquement par "l'action" de la distribution de Dirac $\delta(t)$ décalée de nT_e sur le signal analogique $x_a(t)$, ce qui se note : $\langle \delta(t - nT_e), x_a(t) \rangle = x_a(nT_e)$. Et on notera souvent $x_a(t)$ directement sous la forme $x(n)$.

¹ L. Schwartz, "Méthodes mathématiques pour les Sciences Physiques," Ed Hermann, 1961.

Remarque 1 : Dans la suite de ce cours on différenciera la distribution de Dirac $\delta(t)$ (parfois appelée "impulsion de Dirac"), du symbole de Kronecker $\delta(n) = \begin{cases} 1 & \text{si } n = 0 \\ 0 & \text{si } n \neq 0 \end{cases}$, par le fait que la distribution est un opérateur qui s'applique

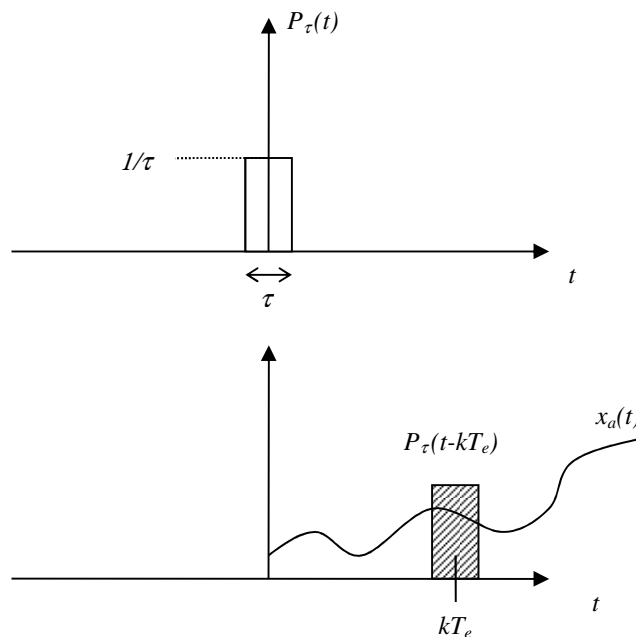
sur un signal et que cet opérateur dépendra d'une variable continue, ici le temps t , alors que le symbole de Kronecker représente plutôt une suite numérique et aura pour argument un nombre entier n .

Remarque 2 :

En électronique, l'utilisation du peigne de Dirac pour "formaliser" l'opération d'échantillonnage d'un signal "continu" (analogique) est assez intuitive physiquement. En effet, on peut considérer qu'un convertisseur analogique numérique estime la moyenne du signal pendant un temps très court. L'échantillonnage à l'instant $t = kT_e$ peut ainsi être formalisé par l'intégrale du signal par une impulsion $p_\tau(t - kT_e)$ définie par :

$$p_\tau(t) = 1, \text{ si } t \in \left[-\frac{\tau}{2}, +\frac{\tau}{2}\right]$$

$$p_\tau(t) = 0, \text{ si } t \notin \left[-\frac{\tau}{2}, +\frac{\tau}{2}\right]$$



Echantillonner le signal à l'instant $t = kT_e$ revient alors à calculer :

$$\int_{-\infty}^{+\infty} p_{\tau}(t - kT_e) x_a(t) dt = \frac{1}{\tau} \int_{kT_e - \frac{\tau}{2}}^{kT_e + \frac{\tau}{2}} x_a(t) dt$$

d'après le théorème de la moyenne, on peut dire que :

$$\frac{1}{\tau} \int_{kT_e - \frac{\tau}{2}}^{kT_e + \frac{\tau}{2}} x_a(t) dt = x_a(kT_e + \varepsilon) \text{ avec } \varepsilon \in \left[-\frac{\tau}{2}, \frac{\tau}{2} \right]$$

dès lors, en faisant tendre τ vers 0, il vient :

$$\lim_{\tau \rightarrow 0} \int_{-\infty}^{+\infty} p_{\tau}(t - kT_e) x_a(t) dt = x_a(kT_e) = x(k) = \int_{-\infty}^{+\infty} x(t) \delta(t - kT_e) dt$$

On retrouve ainsi la définition de la distribution de Dirac.

Finalement, l'opération globale d'échantillonnage peut être formalisée en introduisant un signal "analogique" fictif $x_e(t)$ qui est nul presque partout et égal à $x_a(t)$ pour $t = nT_e$. En introduisant alors le peigne de Dirac :

$$w_{T_e}(t) = \sum_{n=-\infty}^{+\infty} \delta(t - nT_e) \quad (1.)$$

Le signal échantillonné s'écrit comme l'application de ce peigne $w_{T_e}(t)$ au signal $x_a(t)$:

$$x_e(t) = x_a(t) \cdot w_{T_e}(t) \quad (2.)$$

ou encore :

$$x_e(t) = \sum_{n=-\infty}^{+\infty} x_a(t) \delta(t - nT_e) \quad (3.)$$

Note : on peut aussi rencontrer le forme suivante :

$$x_e(t) = \sum_{n=-\infty}^{+\infty} x(n) \delta\left(\frac{t}{T_e} - n\right) \quad (4.)$$

1.2 Périodisation du spectre

L'analyse du spectre du signal échantillonné fait appel aux propriétés de la Transformée de Fourier (\mathcal{F}) des distributions. On doit alors partir de l'écriture du signal échantillonné :

$$x_e(t) = x_a(t) \cdot w_{T_e}(t) \quad (5.)$$

dont la Transformée de Fourier conduit à :

$$X_e(f) = \mathcal{F}(x_e(t)) = \mathcal{F}(x_a(t) \cdot w_{T_e}(t)) = \mathcal{F}(x_a(t)) * \mathcal{F}(w_{T_e}(t)) = X_a(f) * \mathcal{F}(w_{T_e}(t)) \quad (6.)$$

Il apparaît donc que le spectre du signal échantillonné est égal au spectre du signal analogique convolué par la Transformée de Fourier du peigne de Dirac : $\mathcal{F}(w_{T_e}(t))$.

Or on peut montrer² que cette Transformée de Fourier est, elle-même, un peigne de Dirac : $\mathcal{F}(w_{T_e}(t)) = W_{T_e}(f)$. La démonstration (non présentée dans ce cours) procède en deux étapes, on démontre d'abord que :

$$\mathcal{F}(w_{T_e}(t)) = \sum_{n=-\infty}^{+\infty} e^{-j2\pi f n T_e} \quad (7.)$$

puis que :

$$\sum_{n=-\infty}^{+\infty} e^{-j2\pi f n T_e} = \frac{1}{T_e} \sum_{n=-\infty}^{+\infty} \delta\left(f - \frac{n}{T_e}\right) \quad (8.)$$

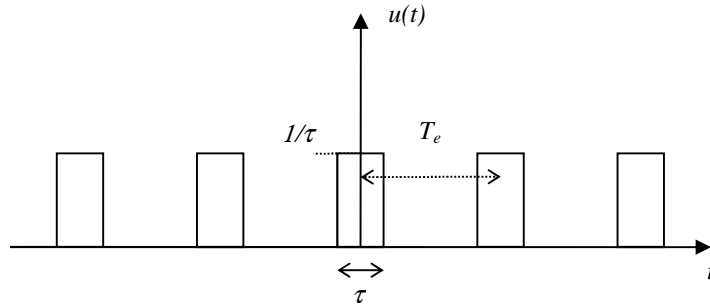
Remarque 3 :

Les résultats sur la Transformée de Fourier du peigne de Dirac peuvent être approximativement "retrouvés" par l'approche suivante :

On considère la fonction $u(t)$ constituée par la suite d'impulsions $p_\tau(t)$ de largeur τ et d'amplitude $\frac{1}{\tau}$, séparées par la durée T_e :

$$u(t) = \sum_{k=-\infty}^{+\infty} p_\tau(t - kT_e)$$

² E. Roubine, "Introduction à la théorie de la communication, " Ed. Masson, 2^{ème} ed., 1979.



La Transformée de Fourier de l'impulsion élémentaire $p_\tau(t)$ donne :

$$\mathcal{F}\{p_\tau(t)\} = \int_{-\infty}^{+\infty} p_\tau(t) e^{-j2\pi f t} dt = \frac{\sin(\pi f \tau)}{\pi f \tau}$$

La Transformée de Fourier de la suite d'impulsions $u(t)$ conduit alors à :

$$\mathcal{F}\{u(t)\} = \int_{-\infty}^{+\infty} \sum_{k=-\infty}^{+\infty} i_p(t - kT_e) e^{-j2\pi f t} dt = \sum_{k=-\infty}^{+\infty} \int_{-\infty}^{+\infty} i_p(t - kT_e) e^{-j2\pi f t} dt = \sum_{k=-\infty}^{+\infty} e^{-j2\pi f kT_e} \frac{\sin(\pi f \tau)}{\pi f \tau}$$

En faisant alors tendre τ vers 0, il vient :

$$\lim_{\tau \rightarrow 0} \mathcal{F}\{u(t)\} = \sum_{k=-\infty}^{+\infty} e^{-j2\pi f kT_e}$$

Pour la deuxième propriété, on peut introduire la fonction $U(f)$ définie par :

$$U(f) = \sum_{k=-\infty}^{+\infty} e^{-j2\pi f kT_e}$$

Grâce aux bornes infinies de cette sommation on peut écrire :

$$e^{-j2\pi f T_e} U(f) = U(f)$$

d'où :

$$(1 - e^{-j2\pi f T_e}) U(f) = 0$$

$U(f)$ ne peut donc être non nulle qu'aux abscisses du type $f = \frac{k}{T_e}$, on peut donc écrire :

$$U(f) = \sum_{k=-\infty}^{+\infty} \alpha_k \delta\left(f - \frac{k}{T_e}\right)$$

Or toujours grâce aux bornes infinies, on peut remarquer que :

$$U(f) = U\left(f + \frac{k}{T_e}\right)$$

Dès lors en appliquant $U(f)$ à une fonction $\varphi(f)$ élémentaire on pourra montrer tous les termes α_k sont égaux entre eux et valent $\frac{1}{T_e}$.

En utilisant l'équation (6), il vient que le spectre du signal échantillonné $X_e(f)$ s'écrit donc de la manière suivante.

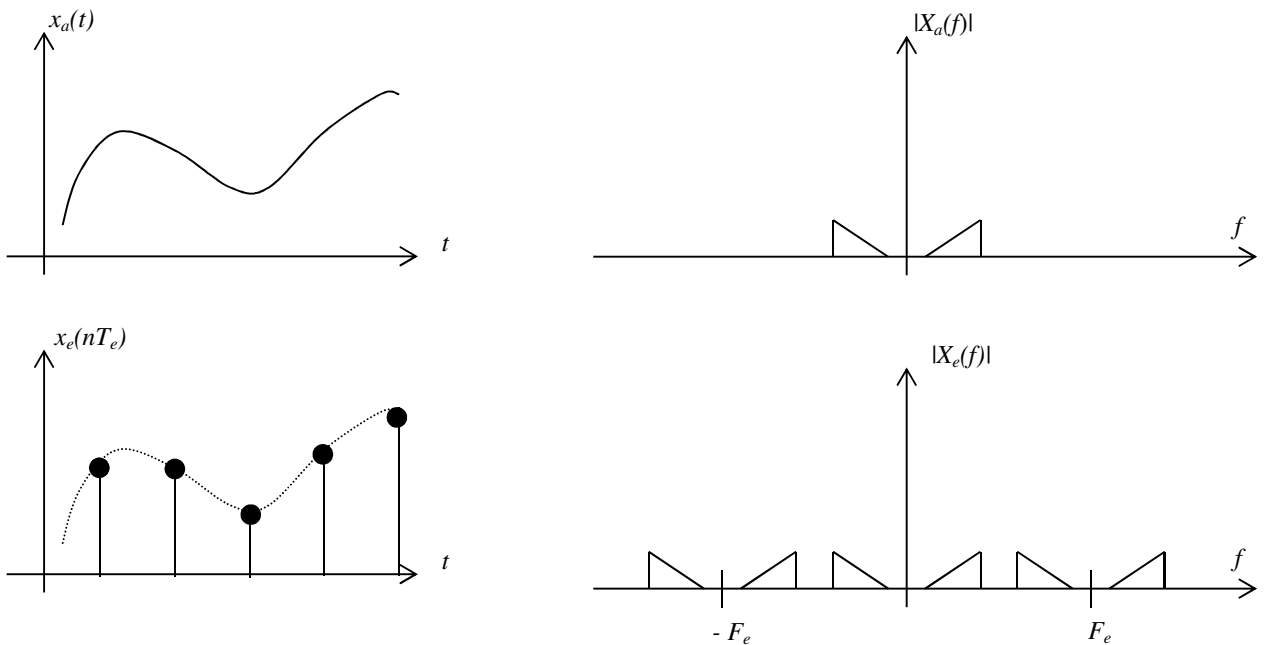
Signal en temporel : $x_e(t) = x_a(t) \cdot w_{T_e}(t)$

Signal en fréquentiel : $X_e(f) = X_a(f) * W_{T_e}(f)$

Donc : $X_e(f) = X_a(f) * \frac{1}{T_e} \sum_{n=-\infty}^{+\infty} \delta\left(f - \frac{n}{T_e}\right)$

D'où : $X_e(f) = \frac{1}{T_e} \sum_{n=-\infty}^{+\infty} X_a\left(f - \frac{n}{T_e}\right)$

$$X_e(f) = \frac{1}{T_e} \sum_{n=-\infty}^{+\infty} X_a(f - nF_e) \quad (9.)$$



Il apparaît donc que l'échantillonnage temporel d'un signal analogique conduit à un signal numérique dont le spectre est la périodisation du spectre d'origine du signal analogique. Cette propriété est très importante et va être à l'origine de la démonstration du théorème de reconstruction. Cependant elle est assez abstraite, car elle a demandé un passage par les distributions pour être établie. On peut en proposer une version "imagée" suivante.

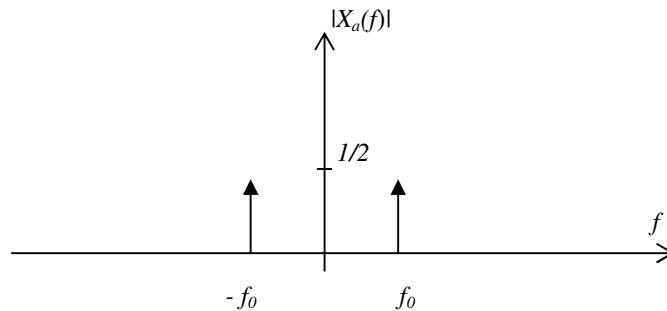
Considérons ainsi le cas très simple d'un signal :

$$x_a(t) = \cos(2\pi f_0 t) \quad (10.)$$

Le spectre de ce signal est alors égal à :

$$X_a(f) = \frac{1}{2} [\delta(f - f_0) + \delta(f + f_0)] \quad (11.)$$

Ce qui peut s'interpréter comme le fait que le signal est en fait constitué de la somme de deux fréquences à f_0 et $-f_0$, $\cos(2\pi f_0 t) = \frac{1}{2} (e^{j2\pi f_0 t} + e^{-j2\pi f_0 t})$. Ce qui se représente graphiquement par le spectre suivant :



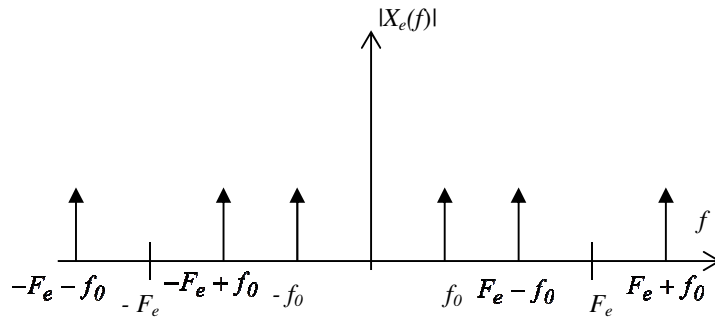
Le signal échantillonné va s'écrire :

$$x_e(nT_e) = \cos(2\pi f_0 nT_e) \quad (12.)$$

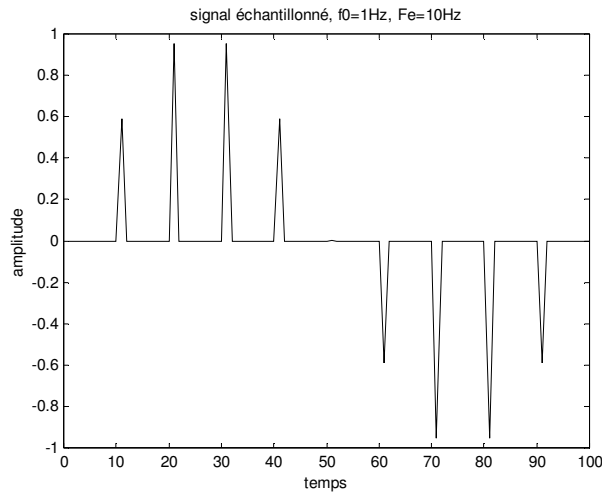
et son spectre :

$$X_e(f) = \frac{1}{2T_e} \sum_{n=-\infty}^{+\infty} \left[\delta\left(f - f_0 - \frac{n}{T_e}\right) + \delta\left(f + f_0 - \frac{n}{T_e}\right) \right] \quad (13.)$$

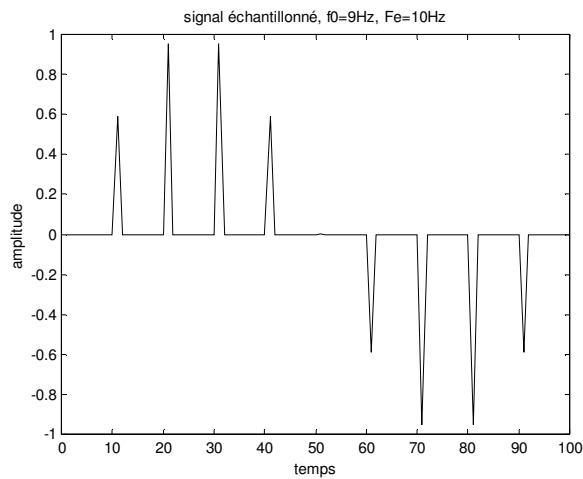
Ce qui sera représenté par :



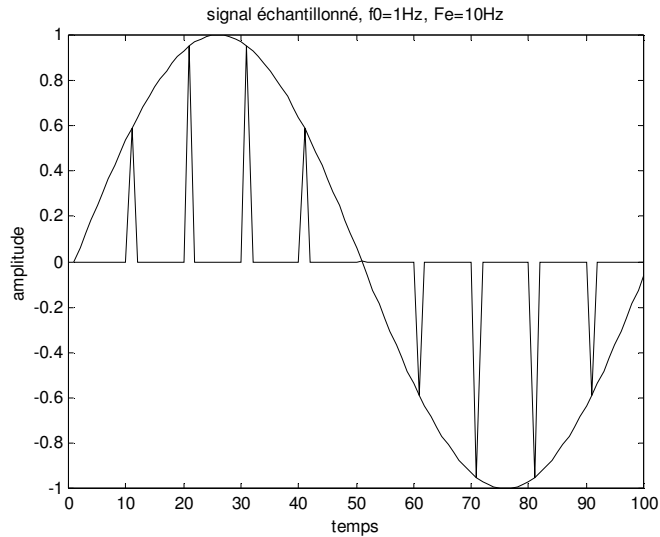
Considérons le cas numérique suivant : $f_0 = 1 \text{ Hz}$ et $F_e = 10 \text{ Hz}$, le signal échantillonné a alors l'allure suivante (attention c'est un sinus qui est représenté ici et non un cosinus) :



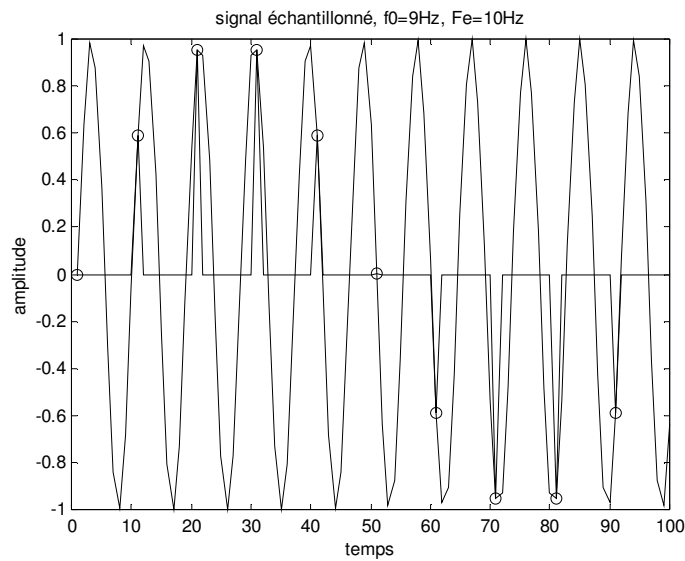
La périodisation nous dit que le spectre de ce signal numérique possède une raie à $f' = F_e - f_0 = 10 - 1 = 9 \text{ Hz}$ et une raie à $f'' = F_e + f_0 = 10 + 1 = 11 \text{ Hz}$. Si on trace le signal temporel $x_a(t) = \sin(2\pi f' nT_e)$ toujours avec $F_e = 10 \text{ Hz}$, on obtient :



On remarque donc que l'on obtient exactement les mêmes échantillons. Si l'on trace les signaux analogiques sur ces points numériques, il vient :



Et pour le signal à 9 Hz :



On peut donc interpréter la périodisation dans le domaine fréquentiel par le fait que tous les signaux analogiques se trouvant à des fréquences du type $f = f_0 \pm kF_e$, donneraient, s'ils étaient échantillonnés à F_e , les mêmes échantillons temporels. On conçoit donc qu'à partir d'un signal numérique, il faudra une condition supplémentaire sur le signal analogique d'origine, pour pouvoir le reconstruire et lever cette ambiguïté.

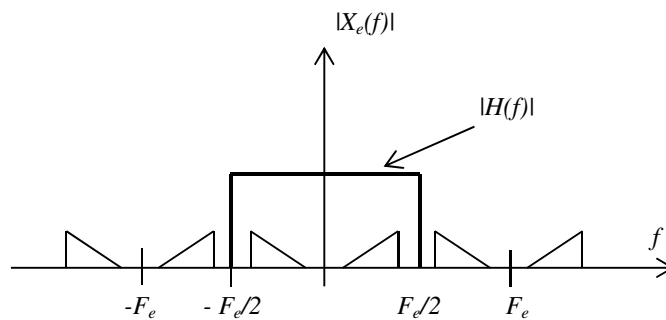
1.3 Le théorème de reconstruction

L'échantillonnage a introduit une périodicité du spectre. Pour reconstituer le signal d'origine on peut "travailler" dans le domaine spectral pour retrouver le spectre du signal analogique. Il ne restera plus alors qu'à effectuer une transformation de Fourier inverse pour reconstituer le signal analogique temporel.

Dans le domaine spectral, il suffit simplement de supprimer les bandes images du signal numérique. En introduisant un filtre idéal $H(f)$, dont la fonction de transfert est définie par :

$$H(f) = \frac{1}{F_e}, \text{ pour } f \in \left[-\frac{F_e}{2}, \frac{F_e}{2} \right]$$

$$H(f) = 0, \text{ pour } f \notin \left[-\frac{F_e}{2}, \frac{F_e}{2} \right]$$



Le signal $\hat{x}_a(t)$ en sortie d'un tel filtre sera défini par le produit de convolution du signal $x_e(t)$ par la réponse impulsionnelle $h(t)$ du filtre $H(f)$. Développons cette réponse impulsionnelle $h(t)$:

$$h(t) = \frac{1}{F_e} \int_{-\infty}^{+\infty} H(f) e^{j2\pi ft} df = \frac{1}{F_e} \int_{-F_e/2}^{F_e/2} e^{j2\pi ft} df = \frac{\sin(\pi F_e t)}{\pi F_e t}$$

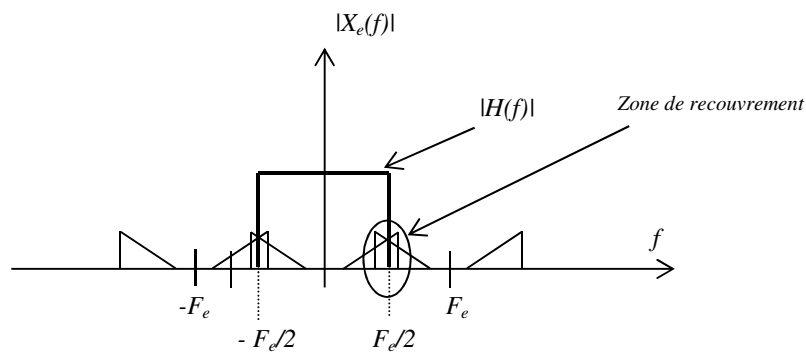
On a donc :

$$\hat{x}_a(t) = \int_{-\infty}^{+\infty} \left[\sum_{n=-\infty}^{+\infty} x_e(\tau) \delta(\tau - nT_e) \right] \frac{\sin \pi F_e (t - \tau)}{\pi F_e (t - \tau)} d\tau \quad (14.)$$

ce qui peut encore s'écrire :

$$\hat{x}_a(t) = \sum_{n=-\infty}^{+\infty} x_e(nT_e) \frac{\sin \pi F_e (t - nT_e)}{\pi F_e (t - nT_e)} \quad (15.)$$

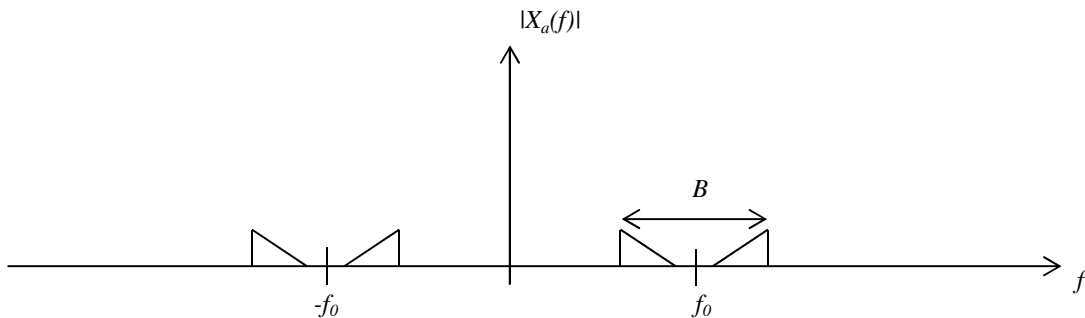
On constate donc que la valeur $\hat{x}_a(t)$ du signal analogique, pour un instant quelconque t n'appartenant pas à la "grille d'échantillonnage temporel" $(kT_e)_{k \text{ entier}}$, peut être obtenue par interpolation des valeurs du signal sur la grille d'échantillonnage. Mais ceci à condition que le raisonnement qui a été proposé dans le domaine spectral soit possible. Pour cela il faut donc s'assurer que l'on peut reconstituer le spectre du signal analogique en filtrant le spectre du signal numérique. Cette condition est vérifiée si et seulement si le spectre d'origine ne contient pas de composantes aux fréquences supérieures ou égales à $\frac{F_e}{2}$. Si ce n'est pas le cas, les bandes images se chevauchent, on dit alors qu'il y a repliement de spectre et le signal reconstitué $\hat{x}_a(t)$ est différent du signal d'origine.



On aboutit finalement au théorème de l'échantillonnage ou théorème de Shannon :

Théorème de l'échantillonnage en bande de base : Un signal qui ne comporte pas de composantes à des fréquences supérieures ou égales à une valeur f_{max} est entièrement déterminé par la suite de ses valeurs à des instants régulièrement espacés d'une durée $T_e = \frac{1}{F_e}$ à condition d'avoir $F_e > 2 f_{max}$

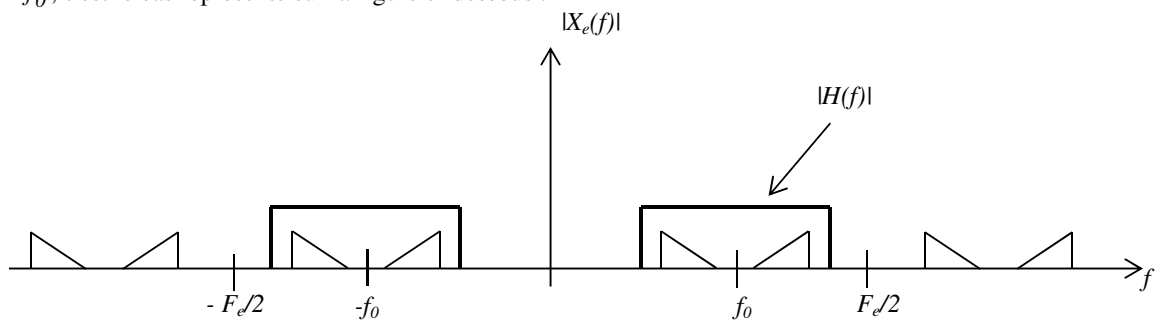
Le raisonnement qui a été mené pour un signal en bande de base, peut être conduit pour un signal dont le spectre se trouverait localisé autour d'une fréquence haute f_0 .



On peut alors énoncer le théorème suivant :

Théorème de l'échantillonnage en bande transposée : Un signal qui occupe une bande de fréquence de largeur B peut-être entièrement déterminé par la suite de ses valeurs à des instants régulièrement espacés d'une durée $T_e = \frac{1}{F_e}$ à condition d'avoir $F_e > 2B$

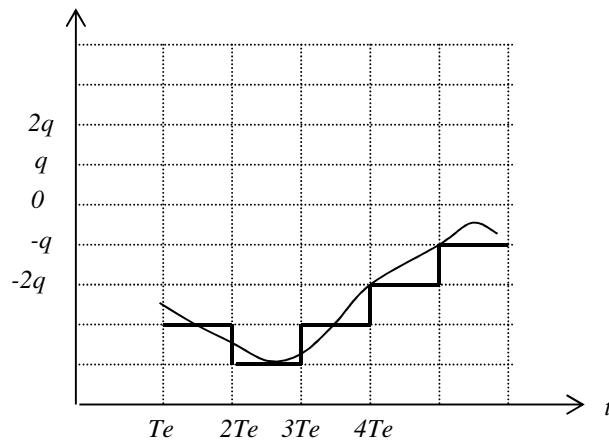
On distingue alors deux cas possibles, le premier appelé suréchantillonnage qui correspond au cas où $F_e > 2B$ et $F_e > f_0$, c'est le cas représenté sur la figure ci-dessous :



Le deuxième cas correspond au sous-échantillonnage pour lequel on $F_e > 2B$ et $F_e < f_0$. Ce cas plus difficile à analyser sera étudié en exercices dirigés.

1.4 Quantification

Dans une chaîne de traitement numérique du signal, l'échantillonnage est en général suivi par une opération de quantification. La quantification est l'approximation de chaque valeur du signal $x_a(t)$ par un multiple entier d'une quantité notée q et appelée "pas de quantification". Si q est constant quelle que soit l'amplitude du signal, la quantification est dite uniforme.



Le signal quantifié $x_q(t)$ diffère du signal d'origine $x_a(t)$ par un terme d'erreur $e(t)$ qui va s'exprimer par :

$$x_a(t) = x_q(t) + e(t) \tag{16.}$$

Ce terme d'erreur $e(t)$ est appelé bruit de quantification.

Si l'on fait abstraction de l'échantillonnage temporel, on peut admettre que ce signal d'erreur est en fait une variable aléatoire uniformément répartie entre $-\frac{q}{2}$ et $\frac{q}{2}$. La puissance P_{Bq} de ce bruit de quantification est alors égale à :

$$P_{Bq} = \int_{-\frac{q}{2}}^{+\frac{q}{2}} \frac{1}{q} x^2 dx \tag{17.}$$

L'intégrale donne alors :

$$P_{Bq} = \frac{q^2}{12} \tag{18.}$$

En général on considère que ce bruit de quantification est un signal aléatoire blanc (voir chapitre sur les signaux aléatoires). On calcule alors le rapport signal sur bruit de quantification. Il s'agit du ratio entre la puissance du signal utile sur la puissance du bruit de quantifications. En notant σ_x^2 la puissance du signal utile et σ_e^2 la puissance du bruit de quantification ($\sigma_e^2 = P_{Bq}$ des équations précédentes), alors le rapport s'écrit $\Gamma = \frac{\sigma_x^2}{\sigma_e^2}$. Ce rapport est souvent exprimé en décibel à travers l'expression $\Gamma_{dB} = 10 \log(\Gamma)$

L'optimisation d'une étape d'échantillonnage réside alors dans la capacité, à être capable de pouvoir quantifier les valeurs maximales de l'amplitude d'un signal, tout en conservant une "finesse" de quantification pour les faibles valeurs du signal.

Pour un convertisseur analogique numérique CAN (analog to digital converter: ADC) de b bits "travaillant" entre $+A/2$ et $-A/2$, le pas de quantification q est égal à $q = \frac{A}{2^b}$. (la plage de variation A , divisée par le nombre maximal représentable avec b bits, soit donc 2^b)

On rappelle que la puissance du bruit de quantification est égale à $\frac{q^2}{12}$. En remplaçant q par sa valeur, le rapport signal sur bruit de quantification Γ est alors donné par :

$$\Gamma_{dB} = 10 \log_{10} \left(\frac{12 \sigma_x^2 2^{2b}}{A^2} \right)$$

En développant cette équation on obtient :

$$\Gamma = 6.02b + 20 \log_{10} \left(\frac{\sqrt{\sigma_x^2}}{A} \right) + 10.8$$

Exemple : Pour un signal gaussien dont la valeur crête serait limitée à $4\sqrt{\sigma_x^2}$, on obtiendrait : $4\sqrt{\sigma_x^2} \leq \frac{A}{2}$, ce qui

donnerait au mieux $4 \frac{\sqrt{\sigma_x^2}}{A} \leq \frac{1}{8}$ d'où $\Gamma_{dB} \leq 6.02b - 7.27$.

(exemple : 16 bits \rightarrow 89 dB, **14 bits \rightarrow 77 dB**, 12 bits \rightarrow 65 dB)

2. Les Transformées

2.1 Transformée de Fourier Discrète

2.1.1 Définition

A partir d'un échantillon de N valeurs du signal numérique: $\{x(nT_e)\}_{n \in \{0,1,2,\dots,N-1\}}$, on peut définir une

correspondance entre la Transformée de Fourier "analogique" $X(f) = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft} dt$ et son expression discrète qui

pourrait s'écrire $X(f) = T_e \sum_{n=-\infty}^{+\infty} x(nT_e)e^{-j2\pi fnT_e}$. Il s'en suit alors immédiatement une discussion sur la convergence

de cette sommation. On peut alors diviser cette transformée discrète par la durée sur laquelle elle est calculée, on passe ainsi d'une notion "d'énergie" à une notion "de puissance". On arrive ainsi à une écriture du type

$X(f) = \lim_{N \rightarrow \infty} \frac{1}{NT_e} T_e \sum_{n=-N/2}^{N/2} x(nT_e)e^{-j2\pi fnT_e}$. Cependant en pratique on ne dispose en général que d'un nombre fini

d'échantillons, la Transformée de Fourier Discrète (TFD) du signal numérique est donc définie par :

$$X(f) = \frac{1}{N} \sum_{n=0}^{N-1} x(nT_e)e^{-j2\pi fnT_e} \quad (19.)$$

On notera que l'on a aussi "recentré" les N échantillons entre les indices 0 et $N-1$ pour éviter d'utiliser la notion de temps négatif. Ces questions de normalisation de la sommation n'ont en général pas une grande importance à moins que l'on ne souhaite absolument faire une correspondance rigoureuse entre le temps continu et le temps discret.

Le calcul de la TFD peut être réalisé pour n'importe quelle valeur de la variable de fréquence f . On peut donc obtenir un spectre $X(f)$ défini pour f variant de manière continue. Ce spectre $X(f)$ peut alors être "échantillonné" au rythme $\frac{F_e}{N}$. On obtient ainsi N valeurs équiréparties de 0 à F_e .

$$X\left(\frac{kF_e}{N}\right) = \frac{1}{N} \sum_{n=0}^{N-1} x(nT_e)e^{-j2\pi \frac{nk}{N}} \quad (20.)$$

D'après le théorème de reconstruction évoqué précédemment, on sait que les valeurs de $X(f)$ aux fréquences f se déduisent de ces N valeurs par interpolation. L'équation d'interpolation est donnée par :

$$\hat{X}(f) = \sum_{k=-\infty}^{+\infty} X\left(\frac{kF_e}{N}\right) \frac{\sin\left(\pi\left(\frac{f}{F_e} - k\right)\right)}{\pi\left(\frac{f}{F_e} - k\right)} \quad (21.)$$

On définit aussi la Transformée de Fourier Inverse (TFI) :

$$x(nT_e) = \sum_{k=0}^{N-1} X\left(\frac{kF_e}{N}\right) e^{+j2\pi\frac{kn}{N}} \quad (22.)$$

2.1.2 Egalité de Parseval

On montre que :

$$\frac{1}{N} \sum_{n=0}^{N-1} |x_e(nT_e)|^2 = \sum_{n=0}^{N-1} \left| X\left(\frac{kF_e}{N}\right) \right|^2 \quad (23.)$$

Pour cela, il suffit d'écrire :

$$\frac{1}{N} \sum_{n=0}^{N-1} |x_e(nT_e)|^2 = \frac{1}{N} \sum_{n=0}^{N-1} x_e(nT_e) x_e^*(nT_e) \quad (24.)$$

$$\frac{1}{N} \sum_{n=0}^{N-1} |x_e(nT_e)|^2 = \frac{1}{N} \sum_{n=0}^{N-1} x_e(nT_e) \sum_{k=0}^{N-1} X^*\left(\frac{kF_e}{N}\right) e^{-j2\pi\frac{kn}{N}} \quad (25.)$$

$$\frac{1}{N} \sum_{n=0}^{N-1} |x_e(nT_e)|^2 = \sum_{k=0}^{N-1} X^*\left(\frac{kF_e}{N}\right) \frac{1}{N} \sum_{n=0}^{N-1} x_e(nT_e) e^{-j2\pi\frac{kn}{N}} \quad (26.)$$

$$\frac{1}{N} \sum_{n=0}^{N-1} |x_e(nT_e)|^2 = \sum_{k=0}^{N-1} \left| X\left(\frac{kF_e}{N}\right) \right|^2 \quad (27.)$$

2.1.3 Interprétation au sens des moindres carrés

Si on considère un signal $x_e(nT_e)$, on peut essayer de le prédire au mieux par une exponentielle complexe du type $A e^{j(2\pi f nT_e + \varphi)}$. Pour identifier les trois paramètres A, f, φ de l'exponentielle, on peut chercher à minimiser l'erreur quadratique entre le signal et l'exponentielle. On doit donc minimiser l'expression suivante :

$$\text{Min}_{A, f, \varphi} \sum_{n=0}^{N-1} \left| A e^{j(2\pi f nT_e + \varphi)} - x(nT_e) \right|^2 \quad (28.)$$

$$\text{Min}_{A, f, \varphi} \sum_{n=0}^{N-1} \left(A^2 + x^2(nT_e) - 2A \text{Re} \left\{ x(nT_e) e^{-j(2\pi f nT_e + \varphi)} \right\} \right) \quad (29.)$$

$$\text{Min}_{A, f, \varphi} \left(N A^2 + \sum_{n=0}^{N-1} x^2(nT_e) - 2NA \text{Re} \left\{ e^{-j\varphi} \frac{1}{N} \sum_{n=0}^{N-1} x(nT_e) e^{-j(2\pi f nT_e + \varphi)} \right\} \right) \quad (30.)$$

On voit donc apparaître la Transformée de Fourier discrète au niveau du troisième terme de cette somme. Si on note cette dernière sous la forme :

$$X(f) = \frac{1}{N} \sum_{n=0}^{N-1} x_e(nT_e) e^{-j2\pi f n T_e} = \rho e^{j\phi} \quad (31.)$$

l'équation à minimiser devient :

$$\text{Min}_{A, f, \phi} \left(N.A^2 + \sum_{n=0}^{N-1} x_e^2(nT_e) - 2NA\rho \cos(\phi - \varphi) \right) \quad (32.)$$

Ce terme est positif et sera donc minimal lorsque $2NA\rho \cos(\phi - \varphi)$ sera maximal. Il faut donc choisir f qui maximise le module ρ de la Transformée de Fourier et $\phi = \varphi$.

Il ne reste plus alors qu'à maximiser en fonction de A :

$$\text{Min}_A \left(N.A^2 + \sum_{n=0}^{N-1} x_e^2(nT_e) - 2NA\rho_{\max} \right) \quad (33.)$$

En annulant alors la dérivée en fonction de A , il vient :

$$2.N.A - 2N\rho_{\max} = 0 \quad (34.)$$

D'où $A = \rho_{\max}$

Il apparaît en définitive que le triplet A, f, ϕ s'obtient simplement en considérant la maximisation sur f du module de la Transformée de Fourier discrète du signal $x_e(nT_e)$. Le module de la Transformée à cette fréquence f donne la valeur de A et la phase de la Transformée cette fréquence f donne ϕ .

2.2 Transformée de Fourier Rapide (FFT)

Si on choisit une formalisation matricielle de la Transformée de Fourier discrète et en se plaçant dans le cas où elle est calculée pour N valeurs aux fréquence $f = \frac{kF_e}{N}$, l'opération peut être formalisée par le passage d'un vecteur d'échantillons temporels :

$\begin{pmatrix} x(0.T_e) \\ x(1.T_e) \\ \vdots \\ x((N-1).T_e) \end{pmatrix}$ que l'on notera plus simplement $\begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix}$, à un vecteur fréquentiel $\begin{pmatrix} X\left(\frac{0.F_e}{N}\right) \\ X\left(\frac{1.F_e}{N}\right) \\ \vdots \\ X\left(\frac{(N-1).F_e}{N}\right) \end{pmatrix}$ que l'on notera plus simplement $\begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{pmatrix}$ au moyen d'une matrice de passage $P = \frac{1}{N} \begin{pmatrix} p_{n,m} \end{pmatrix}$ avec $p_{n,m} = e^{-j2\pi\frac{n.m}{N}}$.

$$\begin{pmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{pmatrix} = P \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix} \quad (35.)$$

En introduisant la racine $n^{\text{ième}}$ de l'unité :

$$W = e^{-j\frac{2\pi}{N}} \quad (36.)$$

L'équation matricielle précédente s'écrit :

$$\begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ \dots \\ X_{N-1} \end{pmatrix} = \frac{1}{N} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W & W^2 & \dots & W^{N-1} \\ 1 & W^2 & W^4 & \dots & W^{2(N-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & W^{N-1} & W^{2(N-1)} & \dots & W^{(N-1)(N-1)} \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ \dots \\ x_{N-1} \end{pmatrix} \quad (37.)$$

Lorsque le nombre N est une puissance de 2, ($N = 2^k$), alors il est possible de mettre à profit certaines particularités de la matrice de passage pour éviter de devoir effectuer les N^2 multiplications que demanderait le produit matriciel direct.

Dans un tel cas de figure, on décompose le vecteur d'échantillons temporels en un vecteur comportant les échantillons d'indice pairs et un vecteur comportant les échantillons d'indice impairs afin d'obtenir les $\frac{N}{2}$ premières valeurs du vecteur fréquentiel.

$$\begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ \vdots \\ X_{N/2-1} \end{pmatrix} = \frac{1}{N} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W^2 & W^4 & \dots & W^{2(N/2-1)} \\ 1 & W^4 & W^8 & \dots & W^{4(N/2-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & W^{2(N/2-1)} & W^{4(N/2-1)} & \dots & W^{2(N/2-1)(N/2-1)} \end{pmatrix} \begin{pmatrix} x_0 \\ x_2 \\ x_4 \\ \dots \\ x_{2(N/2-1)} \end{pmatrix} \\
 + \frac{1}{N} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ W & W^3 & W^5 & \dots & W^{N-1} \\ W^2 & W^6 & W^{10} & \dots & W^{2(N-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ W^{N/2-1} & W^{3(N/2-1)} & W^{5(N/2-1)} & \dots & W^{(N-1)(N/2-1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_3 \\ x_5 \\ \dots \\ x_{N-1} \end{pmatrix} \quad (38.)$$

En désignant par $T_{N/2}$ la matrice qui vient en facteur du vecteur colonne des éléments d'indice pair et en décomposant la matrice facteur du vecteur des éléments d'indice impair en un produit d'une matrice diagonale par la matrice $T_{N/2}$, on obtient :

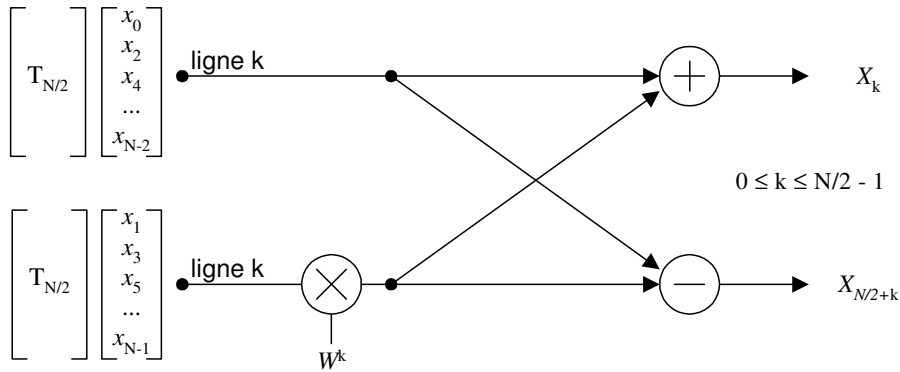
$$\begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ \vdots \\ X_{N/2-1} \end{pmatrix} = \frac{1}{N} T_{N/2} \begin{pmatrix} x_0 \\ x_2 \\ x_4 \\ \vdots \\ x_{2(N/2-1)} \end{pmatrix} + \frac{1}{N} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & W & 0 & 0 & 0 \\ 0 & 0 & W^2 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & W^{(N/2-1)} \end{pmatrix} T_{N/2} \begin{pmatrix} x_1 \\ x_3 \\ x_5 \\ \vdots \\ x_{N-1} \end{pmatrix} \quad (39.)$$

Pour les $N/2$ derniers termes du vecteur fréquentiel, on obtient en utilisant la propriété $W^N = 1$:

$$\begin{pmatrix} X_{N/2} \\ X_{N/2+1} \\ X_{N/2+2} \\ \vdots \\ X_{N-1} \end{pmatrix} = \frac{1}{N} T_{N/2} \begin{pmatrix} x_0 \\ x_2 \\ x_4 \\ \vdots \\ x_{2(N/2-1)} \end{pmatrix} - \frac{1}{N} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & W & 0 & 0 & 0 \\ 0 & 0 & W^2 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & W^{(N/2-1)} \end{pmatrix} T_{N/2} \begin{pmatrix} x_1 \\ x_3 \\ x_5 \\ \vdots \\ x_{N-1} \end{pmatrix} \quad (40.)$$

Il apparaît ainsi que le calcul de X_k et de $X_{k+N/2}$ met en œuvre les mêmes calculs à un changement de signe près.

Ce calcul peut être représenté par le diagramme suivant :



Il apparaît alors que le calcul d'une Transformée de Fourier d'ordre N revient au calcul de deux Transformées d'ordre $N/2$ auquel s'ajoutent $N/2$ multiplications complexes. En itérant ce principe on "descend" jusqu'aux Transformées de Fourier sur deux valeurs, qui s'effectuent au moyen de la matrice :

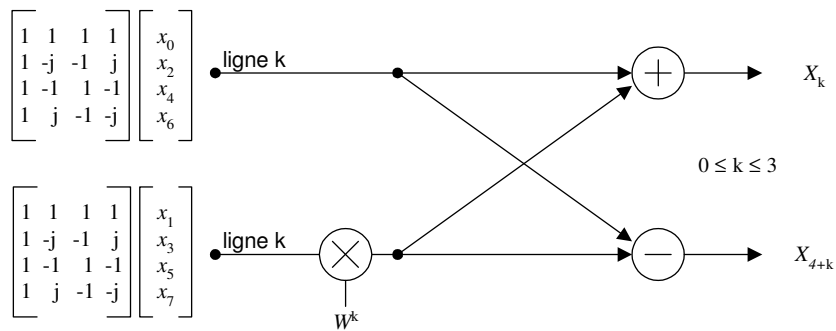
$$T_2 = \begin{bmatrix} I & I \\ I & -I \end{bmatrix} \quad (41.)$$

En énumérant toutes les multiplications à effectuer on constate finalement que l'algorithme de la Transformée de Fourier Rapide (TFR ou FFT en anglais pour Fast Fourier Transform (Cooley and Tuckey 1965)) qui vient d'être développé va demander $N \log_2(N)$ multiplications au lieu des N^2 multiplications du calcul direct.

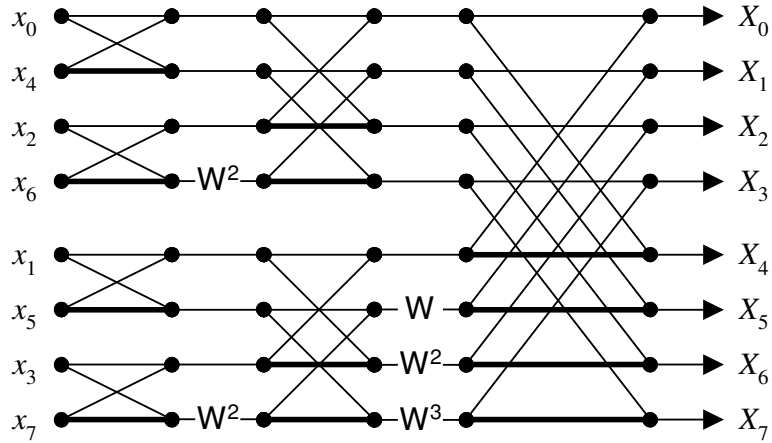
Note : L'introduction du facteur $1/N$ peut se faire à chaque étape en divisant les résultats des additions et soustractions effectuées dans chaque croisillon par $1/2$.

Exemple : FFT d'ordre 8

Une FFT d'ordre 8 aura pour diagramme :

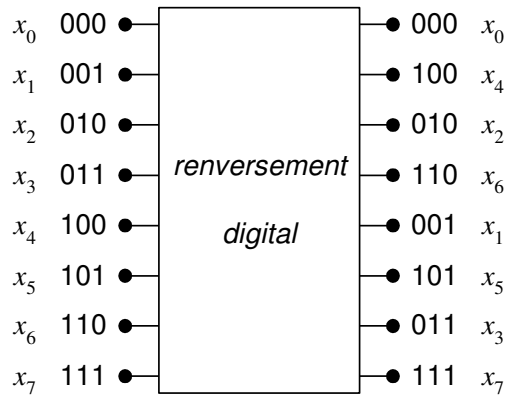


On voit d'après le contenu de T_4 que l'on a intérêt à regrouper x_0 avec x_4 , x_2 avec x_6 , et de même x_1 avec x_5 , et x_3 avec x_7 . Le diagramme a alors la forme suivante (les ronds noirs représentent l'opération d'addition, les traits épais identifient le terme soustrait) :



On constate dans ce diagramme la présence de 4 “papillons FFT” d’ordre 2, 2 “papillons FFT” d’ordre 4 et 1 “papillon FFT” d’ordre 8.

De plus, alors que les termes X_i apparaissent “dans l’ordre”, l’ordre des x_i a été modifié. Cette modification appelée renversement digital, consiste à inverser la représentation binaire des indices des x_i avant chaque calcul de FFT.



Transformée 2 D : $X(f_1, f_2) = \iint x(t_1, t_2) e^{-j2\pi f_1 t_1} e^{-j2\pi f_2 t_2} dt_1 dt_2$

2.3 Transformée en Z

De la même manière que la transformée de Laplace est l'outil fondamental pour l'analyse des systèmes continus, la transformée en Z est l'outil d'analyse pour les systèmes discrets.

On rappelle que la transformée en Z d'une suite $x(n)$ est définie pour $R_1 < Z < R_2$ par l'expression:

$$x(n) \xrightarrow{T(Z)} X(Z) = \sum_{n=-\infty}^{n=+\infty} x(n)Z^{-n} \quad (42.)$$

En considérant l'expression de la Transformée de Fourier discrète :

$$x(n) \xrightarrow{TFD} X(f) = T_e \sum_{n=-\infty}^{n=+\infty} x(nT_e) e^{-j2\pi f n T_e} \quad (43.)$$

Le passage de la transformée en Z à la Transformée de Fourier est immédiat:

$$X(Z) \Big|_{Z=e^{j2\pi f T_e}} = X(f) \quad (44.)$$

(Ceci en faisant abstraction du terme de normalisation T_e que l'on considère égal à 1)

Ainsi l'analyse d'un système discret se fera en général au moyen de la transformée en Z, le passage en Fourier étant immédiat si nécessaire.

3. Filtrage Numérique

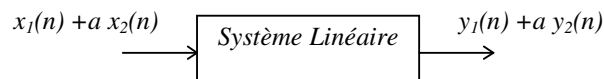
3.1 Les systèmes linéaires discrets invariants dans le temps

3.1.1 Définition

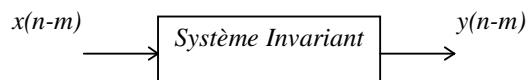
Un système est *discret*, si à la suite d'entrée discrète $x(n)$ correspond une suite de sortie discrète $y(n)$.



Un système est *linéaire*, si à la suite $x_1(n) + a x_2(n)$ correspond la suite $y_1(n) + a y_2(n)$.



Un système est *invariant dans le temps*, si à la suite $x(n-m)$ correspond la suite $y(n-m)$.



Dès lors si $\delta(n)$ est la suite unitaire $\begin{cases} \delta(0) = 1 \\ \delta(n) = 0 \quad \forall n \neq 0 \end{cases}$, alors toute suite $x(n)$ peut s'écrire:

$$x(n) = \sum_{m=-\infty}^{m=+\infty} x(m) \delta(n-m) \quad (45.)$$

si $h(n)$ est la réponse d'un système discret linéaire et invariant dans le temps à la suite $\delta(n)$ alors :

$$x(n) \rightarrow y(n) = \sum_{m=-\infty}^{m=+\infty} x(m) h(n-m) = \sum_{m=-\infty}^{m=+\infty} h(m) x(n-m) \quad (46.)$$

on reconnaît alors une équation de convolution:

$$y(n) = h(n) * x(n) \quad (47.)$$

Ainsi dès qu'un système peut être considéré comme linéaire, discret et invariant dans le temps, il en découle qu'il est :

1) régi par une équation de convolution

2) entièrement déterminé par la réponse $h(n)$ qu'il fournit lorsqu'il est excité par la suite impulsionnelle $\delta(n)$. Cette suite $h(n)$ constituant la réponse impulsionnelle du système.

3.1.2 Conditions de stabilité

Un système discret, linéaire et invariant dans le temps (SLIT) est stable si à toute suite d'entrée bornée correspond une suite de sortie bornée.

Une condition nécessaire et suffisante pour qu'un système soit stable est que la somme des valeurs absolues de sa réponse impulsionnelle soit bornée.

$$\sum_{m=-\infty}^{m=+\infty} |h(m)| < +\infty \quad (48.)$$

Preuve de la condition nécessaire :

Soit $x(n)$ la suite d'entrée bornée définie par : $x(-n) = \text{sgn}(h(n))$ alors, par définition de l'équation de convolution

régissant le système, $y(0) = \sum_{m=-\infty}^{m=+\infty} |h(m)|$. Donc si $y(0) = \sum_{m=-\infty}^{m=+\infty} |h(m)|$ n'est pas $< \infty$ la suite de sortie n'est pas bornée

et la condition de stabilité n'est pas respectée.

Preuve de la condition suffisante :

Soit $x(n)$ une suite d'entrée bornée, c'est à dire:

$$\forall n, \exists M / |x(n)| < M$$

alors:

$$|y(n)| \leq \sum_{m=-\infty}^{m=+\infty} |h(m)| |x(n-m)| \leq \sum_{m=-\infty}^{m=+\infty} |h(m)| M \quad (49.)$$

et si $\sum_{m=-\infty}^{m=+\infty} |h(m)| < \infty$, la suite $y(n)$ est alors bornée.

3.2 Fonctions propres des systèmes linéaires invariants dans le temps

On applique à l'entrée d'un système linéaire discret invariant dans le temps de réponse impulsionnelle $h(n)$ le signal numérique complexe de fréquence f :

$x(n) = e^{j2\pi fnT_e}$. On cherche la réponse temporelle $y(n)$ du système :

$$y(n) = h(n) * x(n) = \sum_{k=-\infty}^{+\infty} h(m) e^{j2\pi f(n-m)T_e} \quad (50.)$$

on peut alors écrire :

$$y(n) = \left[\sum_{k=-\infty}^{+\infty} h(m) e^{-j2\pi fmT_e} \right] e^{j2\pi fnT_e} \quad (51.)$$

ou encore :

$$y(n) = H(f) e^{j2\pi fnT_e} = H(f) x(n) \quad (52.)$$

avec

$$H(f) = \sum_{k=-\infty}^{+\infty} h(m) e^{-j2\pi fmT_e} \quad (53.)$$

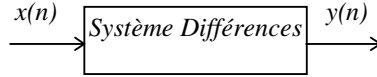
$H(f)$ est un coefficient scalaire complexe indépendant de n (c'est-à-dire du temps) mais qui dépend de la fréquence f . $H(f)$ représente **la réponse en fréquence du filtre**. (C'est la Transformée de Fourier de la réponse impulsionnelle).

Les signaux d'entrée $x(n)$ qui donnent en sortie des signaux $y(n) = H(f)x(n)$ sont appelés **les fonctions propres du système**. Seules les exponentielles complexes $e^{j2\pi fnT_e}$ jouissent de cette propriété. Pour un signal $x(n)$ quelconque, la réponse temporelle $y(n)$ ne peut s'obtenir que par convolution avec $h(n)$ à moins de pouvoir décomposer $x(n)$ en une somme de fonctions propres, ce qui revient à l'exprimer par son spectre.

3.3 Systèmes linéaires invariants dans le temps régis par une équation aux différences

Parmi les systèmes linéaires discrets invariants dans le temps, les systèmes définis par une équation aux différences sont les plus intéressants car ils modélisent un grand nombre de systèmes naturels. Un système de ce type, ou filtre numérique, est défini par la relation suivante:

$$y(n) = \sum_{i=0}^N a_i x(n-i) + \sum_{j=1}^M b_j y(n-j) \quad (54.)$$



La transformée en Z de cette équation donne :

$$\sum_{n=-\infty}^{n=+\infty} y(n)Z^{-n} = \sum_{i=0}^N a_i Z^{-i} \sum_{n=-\infty}^{n=+\infty} x(n-i)Z^{-(n-i)} + \sum_{j=1}^M b_j Z^{-j} \sum_{n=-\infty}^{n=+\infty} y(n-j)Z^{-(n-j)} \quad (55.)$$

d'où :

$$Y(Z) = \sum_{i=0}^N a_i Z^{-i} X(Z) + \sum_{j=1}^M b_j Z^{-j} Y(Z) \quad (56.)$$

ce qui donne la fonction de transfert du système :

$$H(Z) = \frac{Y(Z)}{X(Z)} = \frac{\sum_{i=0}^N a_i Z^{-i}}{1 - \sum_{j=1}^M b_j Z^{-j}} \quad (57.)$$

La fonction de transfert est donc constituée d'un polynôme en Z^{-i} au numérateur sur un autre polynôme en Z^{-j} au dénominateur. Ils peuvent tous les deux être exprimés en fonction de leurs racines :

$$H(Z) = \frac{N(Z)}{D(Z)} = \frac{a_0 \prod_{i=0}^{N-1} (1 - Z_i Z^{-1})(1 - Z_i^* Z^{-1})}{\prod_{j=0}^{M-1} (1 - P_j Z^{-1})(1 - P_j^* Z^{-1})} \quad (58.)$$

A partir de cette expression de la fonction de transfert, une représentation des pôles et des zéros sur le cercle unité s'avère très utile pour caractériser le comportement spectral du système. Comme il a été rappelé brièvement au paragraphe précédent, il est possible d'obtenir la fonction de transfert spectrale de ce système en remplaçant Z par $e^{j2\pi f T_e}$, Z peut donc être vue comme la coordonnée d'un point sur le cercle unité. Z_i sera la coordonnée d'un zéro de transmission dans le plan complexe et P_j d'un pôle dans le plan complexe.

$$|H(f)| = \frac{\prod_{i=0}^{N-1} |MZ_i| |MZ_i^*|}{\prod_{j=0}^{M-1} |MP_j| |MP_j^*|}$$

Si on prend l'exemple d'un filtre d'ordre 2 ($N = M = 2$), la fonction de transfert s'exprime alors simplement comme un ratio de produits de distances.

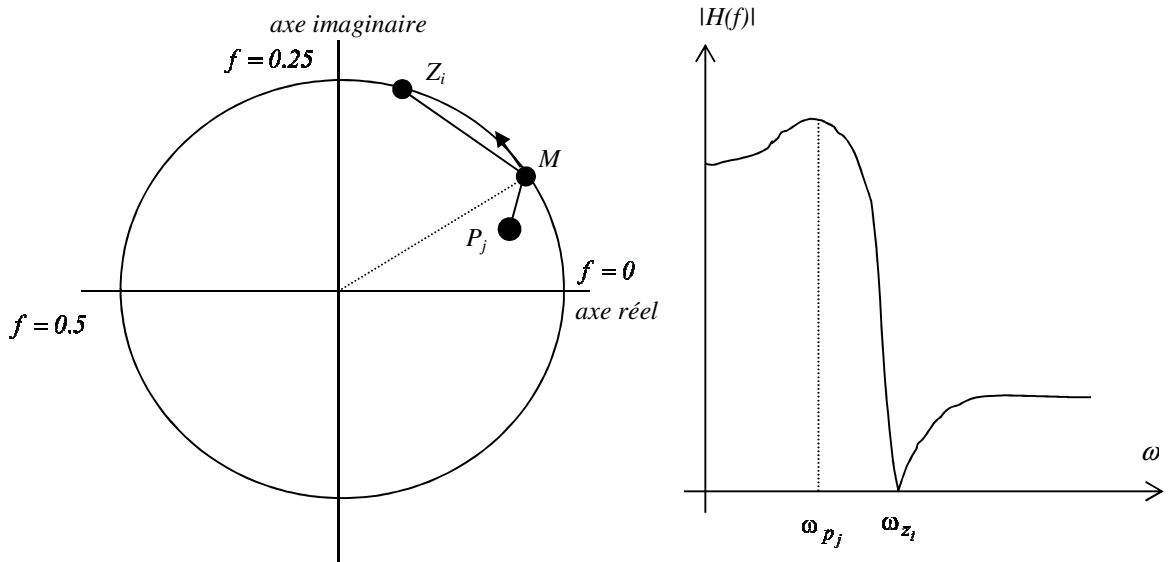
$$|H(Z)| = \frac{|1 - Z_0 Z^{-1}| |1 - Z_0^* Z^{-1}|}{|1 - P_0 Z^{-1}| |1 - P_0^* Z^{-1}|}$$

Ou encore :

$$|H(Z)| = \frac{|Z - Z_0||Z - Z_0^*|}{|Z - P_0||Z - P_0^*|}$$

En introduisant le point M situé à la coordonnée Z et en confondant des points Z₀ et P₀ avec leurs coordonnées, on obtient :

$$|H(f)| = \frac{MZ_0 \overline{MZ_0^*}}{MP_0 \overline{MP_0^*}} \quad (59.)$$



3.4 Filtres à réponse impulsionnelle finie (RIF)

Un filtre à réponse impulsionnelle finie est un système linéaire discret invariant dans le temps, régi par une équation aux différences, pour lequel l'échantillon de sortie $y(n)$ ne dépend que d'un certain nombre d'échantillons d'entrée $x(n)$.

$$y(n) = \sum_{i=0}^N a_i x(n-i) \quad (60.)$$

Exemple 1 :

Soit le filtre défini par l'équation suivante:

$$y(n) = \frac{1}{2} x(n) + \frac{1}{2} x(n-1) \quad (61.)$$

L'étude de ce filtre peut être réalisée au moyen de la transformée en Z:

$$Y(Z) = \frac{1}{2} (X(Z) + Z^{-1} X(Z)) \quad (62.)$$

d'où:

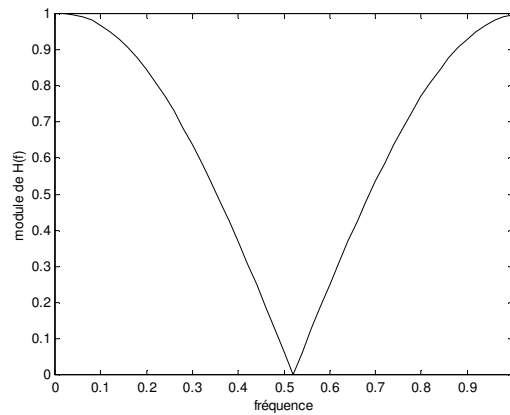
$$H(Z) = \frac{1}{2} + \frac{1}{2} Z^{-1} \quad (63.)$$

Le comportement fréquentiel du filtre s'obtient alors en remplaçant Z par $e^{j2\pi f T_e}$:

$$\begin{aligned} H(f) &= \frac{1}{2} (1 + e^{-j2\pi f T_e}) \\ &= \frac{1}{2} e^{-j\pi f T_e} (e^{j\pi f T_e} + e^{-j\pi f T_e}) \\ &= e^{-j\pi f T_e} \cos(\pi f T_e) \end{aligned}$$

Il s'agit finalement d'un filtre passe bas dont le module de la fonction de transfert suit une courbe en cosinus et dont

le déphasage est linéaire en fonction de la fréquence. Ce déphasage est équivalent à retard $\tau = \frac{T_e}{2}$.



Exemple 2 :

Soit le filtre défini par l'équation suivante:

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

L'étude de ce filtre peut être réalisée au moyen de la transformée en Z:

$$Y(Z) = \frac{1}{4} X(Z) + \frac{1}{2} Z^{-1} X(Z) + \frac{1}{4} Z^{-2} X(Z)$$

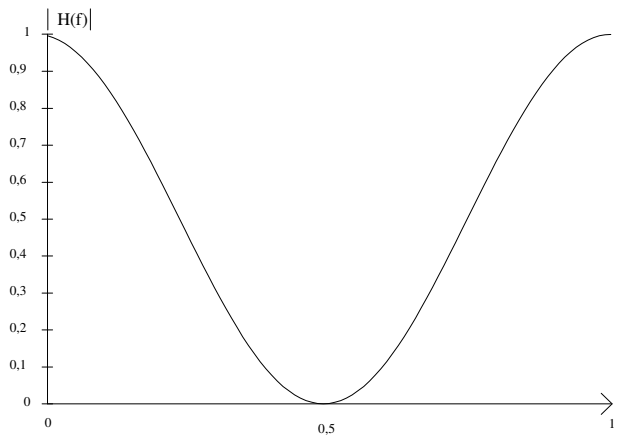
d'où:

$$H(Z) = \frac{1}{4} + \frac{1}{2} Z^{-1} + \frac{1}{4} Z^{-2}$$

Le comportement fréquentiel du filtre s'obtient en remplaçant Z par $e^{j2\pi f T_e}$:

$$\begin{aligned} H(f) &= \frac{1}{2} \left(\frac{1}{2} + e^{-j2\pi f T_e} + \frac{1}{2} e^{-j4\pi f T_e} \right) \\ &= \frac{1}{2} e^{-j2\pi f T_e} \left(1 + \frac{1}{2} e^{j2\pi f T_e} + \frac{1}{2} e^{-j2\pi f T_e} \right) \\ &= \frac{1}{2} e^{-j2\pi f T_e} (1 + \cos 2\pi f T_e) \end{aligned}$$

Il s'agit encore d'un filtre passe bas dont le module de la fonction de transfert suit une courbe en cosinus surélevé et dont le déphasage est linéaire en fonction de la fréquence. Ce déphasage est équivalent à retard $\tau = T_e$.



Dans ces deux exemples il est apparu qu'il était facile d'obtenir la fonction de transfert spectrale d'un filtre numérique à partir de l'équation temporelle régissant ce filtre. Bien entendu, c'est essentiellement l'approche inverse, consistant à trouver l'équation de filtrage qui satisfait à un gabarit fréquentiel donné, qui est la plus importante. On parle alors de synthèse de filtre numérique. Avant de présenter ces techniques de synthèse, il est important de remarquer, que dans

les deux exemples présentés, les filtres décrits avaient un déphasage linéaire en fonction de la fréquence. Il s'agit d'une propriété particulière des filtres numériques, qui a une importance capitale dans les applications où la phase du signal traité est porteuse d'informations.

3.4.1 Propriété de phase linéaire

Si l'on se place dans le cas où l'on recherche un filtre dont la fonction de transfert est du type :

$$H(f) = R(f)e^{-j\varphi(f)}$$

expression dans laquelle $R(f)$ est une fonction réelle de f qui représente le gain du filtre en fonction de la fréquence et où $\varphi(f) = 2\pi f \tau$ est une fonction réelle de f qui représente un terme de déphasage linéaire en fonction de la fréquence, alors on peut exprimer la réponse impulsionnelle d'un tel filtre au moyen de la Transformée de Fourier inverse de $H(f)$ et on obtient :

$$h(t) = \int_{-\infty}^{+\infty} R(f)e^{j2\pi f(t-\tau)} df$$

en décomposant $R(f)$ en une partie paire et une partie impaire:

$$R(f) = P(f) + I(f)$$

il vient:

$$h(t + \tau) = 2 \int_0^{+\infty} P(f) \cos(2\pi f t) df + 2j \int_0^{+\infty} I(f) \sin(2\pi f t) df$$

si on se restreint à des filtres $h(t)$ réels, il vient:

$$h(t + \tau) = 2 \int_0^{+\infty} P(f) \cos(2\pi f t) df = h(\tau - t)$$

$h(t)$ est donc symétrique par rapport à $t = \tau$.

Ainsi tout filtre à réponse impulsionnelle symétrique réel est à phase linéaire.

Le retard équivalent au déphasage est fonction de l'ordre du filtre.

Un filtre RIF symétrique réel à $2p+1$ coefficients entraîne un retard $\tau = pT_e$

Un filtre RIF symétrique réel à $2p$ coefficients entraîne un retard $\tau = (p-1/2)T_e$

preuve :

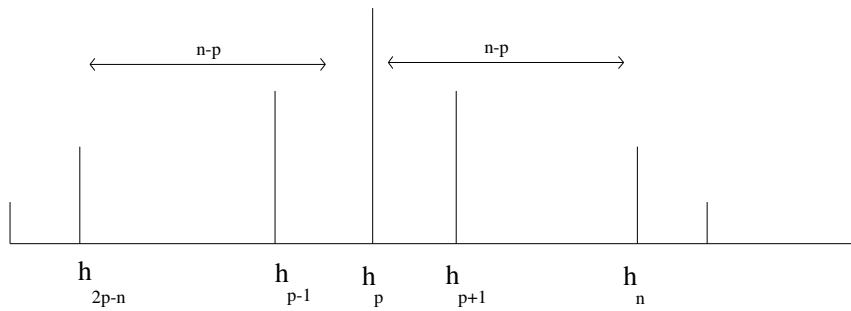
Un filtre RIF symétrique réel à $2p+1$ coefficients entraîne un retard $\tau = pT_e$

La fonction de transfert d'un tel filtre s'écrit:

$$H(f) = \sum_{n=0}^{2p} h(n)e^{-j2\pi f nT_e}$$

Le filtre étant symétrique, on a :

$$\forall n > p, h(n) = h(p - (n - p)) = h(2p - n)$$



en factorisant $e^{-j2\pi f pT_e}$ dans l'expression de $H(f)$, il vient :

$$H(f) = \left(h(p) + \sum_{n=0}^{p-1} h(n)e^{-j2\pi f (n-p)T_e} + \sum_{n=p+1}^{2p} h(n)e^{-j2\pi f (n-p)T_e} \right) e^{-j2\pi f pT_e}$$

or du fait de la symétrie du filtre : $h(n) = h(2p - n)$, d'où:

$$\sum_{n=p+1}^{2p} h(n)e^{-j2\pi f (n-p)T_e} = \sum_{n=p+1}^{2p} h(2p-n)e^{-j2\pi f (n-p)T_e}$$

en effectuant le changement de variable: $i = 2p - n$ ce terme devient égal à:

$$\sum_{i=p-1}^0 h(i) e^{-j2\pi f (p-i)T_e}$$

d'où :

$$H(f) = \left(h(p) + \sum_{n=0}^{p-1} h(n) e^{-j2\pi f (n-p)T_e} + \sum_{n=0}^{p-1} h(n) e^{+j2\pi f (n-p)T_e} \right) e^{-j2\pi f pT_e}$$

ou encore :

$$H(f) = \left(h(p) + 2 \sum_{n=0}^{p-1} h(n) \cos(2\pi f (n-p)T_e) \right) e^{-j2\pi f pT_e}$$

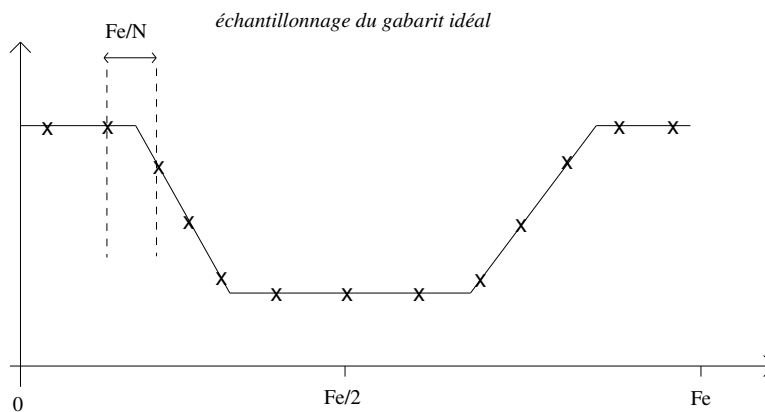
$$= R(f) e^{-j2\pi f \tau} \text{ avec } \tau = pT_e$$

La démonstration pour le cas du filtre à $2p$ coefficients est identique.

3.4.2 Synthèse des filtres à réponse impulsionnelle finie

La synthèse des filtres numériques est un domaine qui a donné lieu à de nombreuses publications et recherches. Seules la base des principales méthodes est exposée dans ce chapitre. La bibliographie fournie en annexe comporte les développements complets sur ce sujet.

La méthode la plus directe pour synthétiser le filtre numérique qui correspond à un gabarit fréquentiel donné consiste simplement à échantillonner ce gabarit dans le plan des fréquences et à calculer la Transformée de Fourier inverse de ce gabarit échantillonné. En l'absence de gabarit de phase la méthode la plus simple consiste à considérer que cette dernière est linéaire en fonction de la fréquence.



on a alors les valeurs de:

$H\left(\frac{kF_e}{N}\right)$ pour $k = 0 \rightarrow N-1$, d'où par Transformée de Fourier inverse:

$$h_i = \sum_{k=0}^{N-1} H\left(\frac{k}{N} F_e\right) e^{j2\pi \frac{k}{N} F_e i T_e} \text{ pour } i = 0 \text{ à } N-1, (N: \text{ordre du filtre})$$

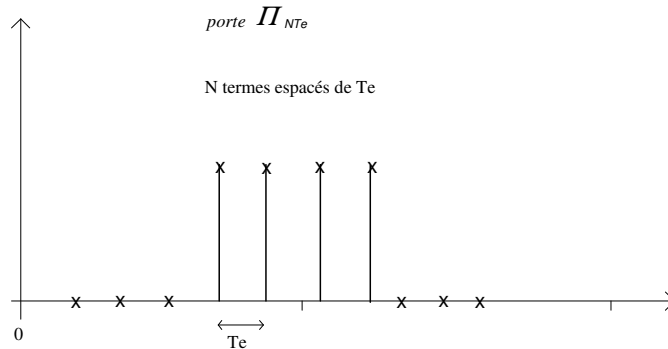
les h_i ainsi obtenus vont bien donner la fonction de transfert idéale aux points $H\left(\frac{kF_e}{N}\right)$, mais ils donneront un spectre avec des ondulations entre ces valeurs.

Pour une fréquence f quelconque, on peut recalculer:

$$H(f) = \frac{1}{N} \sum_{i=0}^{N-1} h_i e^{-j2\pi f i T_e}$$

mais $H(f)$ peut aussi s'obtenir par interpolation des termes $H\left(\frac{kF_e}{N}\right)$. En effet la Transformée de Fourier inverse idéale devrait donner une infinité de termes $(h_i)_{i=-\infty, +\infty}$, or on a considéré uniquement N termes $(h_i)_{i=0, N-1}$. Cela revient à tronquer cette réponse impulsionnelle en la multipliant par une porte $\Pi_{(NT_e)}(NT_e)$. La fonction de transfert obtenu est donc égale au produit de convolution du gabarit idéal par la Transformée de Fourier de la porte $\Pi_{(NT_e)}(NT_e)$.

Transformée de Fourier de la porte $\Pi_{(NT_e)}(NT_e)$:



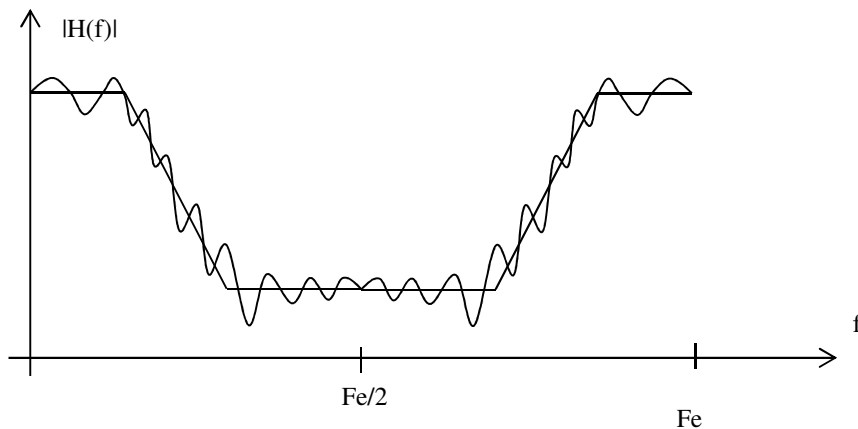
$$\begin{aligned} TF(\pi_{NT_e}) &= \frac{1}{N} \sum_{i=0}^{N-1} e^{-j2\pi f i T_e} \\ &= \frac{1}{N} \frac{1 - e^{-j2\pi f N T_e}}{1 - e^{-j2\pi f T_e}} \\ &= \frac{1}{N} \frac{e^{-j\pi f N T_e} (e^{j\pi f N T_e} - e^{-j\pi f N T_e})}{e^{-j\pi f T_e} (e^{j\pi f T_e} - e^{-j\pi f T_e})} \end{aligned}$$

$$TF(f) = \frac{1}{N} e^{-j\pi f (N-1)T_e} \frac{\sin \pi f N T_e}{\sin \pi f T_e}$$

L'écriture du produit de convolution conduit alors à l'expression de $H(f)$ pour une fréquence f quelconque, en fonction des $H\left(\frac{kF_e}{N}\right)$ (sans avoir besoin de repasser par les $(h_i)_{i=0, N-1}$).

$$H(f) = \sum_{k=0}^{N-1} H\left(\frac{k}{N} F_e\right) \frac{\sin \pi N \left(\frac{f}{F_e} - \frac{k}{N}\right)}{N \sin \pi \left(\frac{f}{F_e} - \frac{k}{N}\right)}$$

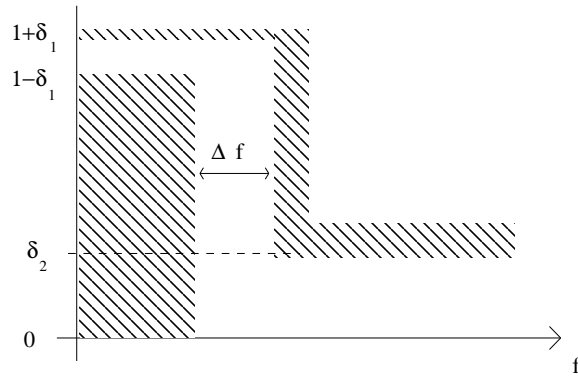
Cette équation constitue une formule d'interpolation pour obtenir $H(f)$ à partir des $H\left(\frac{kF_e}{N}\right)$.



Ainsi la méthode par synthèse conduit à une fonction de transfert qui ondule entre les valeurs idéales. L'amplitude de ces ondulations n'est pas contrôlable et n'est pas constante. C'est contre cette inconvénient que vont tenter de lutter d'autres méthodes de synthèse (cf Remez) qui ne seront pas présentées dans ce cours (voir référence M. Bellanger, pour plus de précisions sur ce sujet).

3.4.3 Relations entre le nombre de coefficients et le gabarit

Considérons un gabarit de filtrage du type de celui présenté ci-dessous:



Il est alors possible d'estimer le nombre de coefficients dont aura besoin un filtre RIF symétrique réel, au moyen de la formule d'approximation suivante:

$$N \approx \frac{2 F_e}{3 \Delta f} \log_{10} \left(\frac{1}{10 \delta_1 \delta_2} \right)$$

3.5 Filtres à réponse impulsionnelle infinie (RII)

3.5.1 Cellule purement réursive

3.5.1.1 Cellule du premier ordre

Soit le filtre défini par l'équation aux différences suivante :

$$y(n) = x(n) + b y(n-1)$$

Ce filtre est identique à un filtre RIF d'ordre infini. Si l'entrée est la suite unitaire:

$$u_0(n) \begin{cases} u_0(0) = 1 \\ u_0(n) = 0 \quad \forall n \neq 0 \end{cases}, \text{ alors la sortie } y(n) \text{ est telle que:}$$

$$\begin{aligned} y(0) &= 1 \\ y(1) &= b \\ y(2) &= b^2 \\ &\vdots \\ y(n) &= b^n \end{aligned}$$

Ce filtre est stable si :

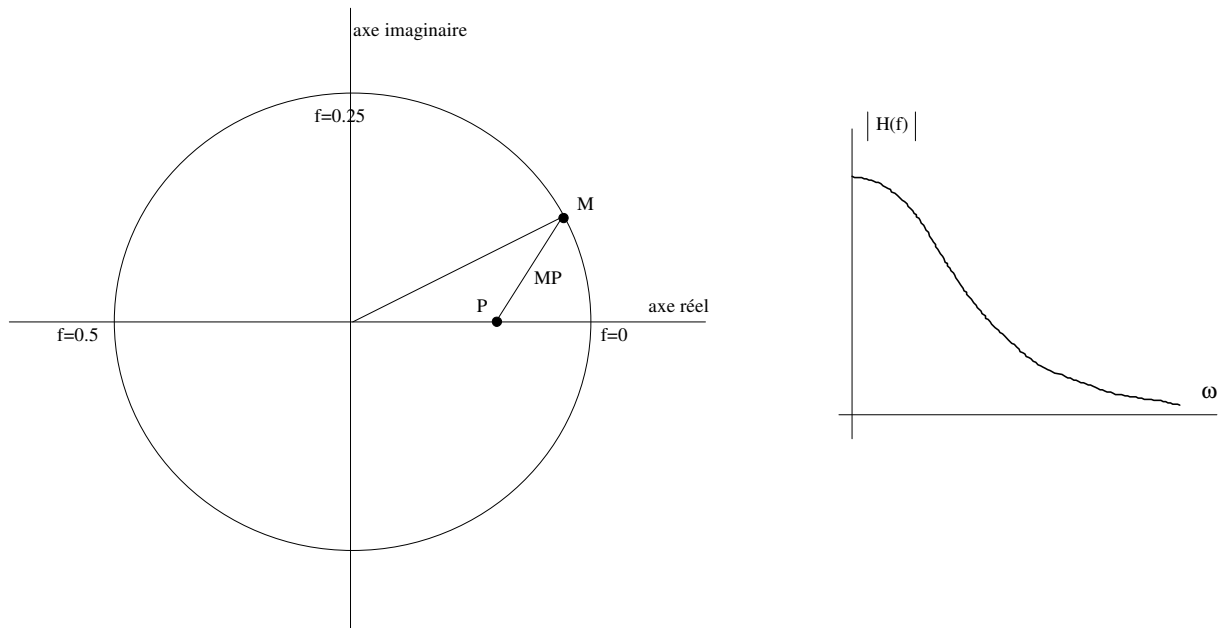
$$\sum_{n=-\infty}^{n=\infty} |b|^n < \infty, \text{ c'est à dire si } |b| < 1$$

Sa fonction de transfert $H(Z)$ s'écrit:

$$H(Z) = \frac{Y(Z)}{X(Z)} = \frac{1}{1 - bZ^{-1}}$$

ou encore avec une représentation sur le cercle unité:

$$|H(Z)| = \frac{1}{MP}$$



3.5.2 Cellule du deuxième ordre

Soit le filtre défini par l'équation aux différences suivante:

$$y(n) = x(n) - b_1 y(n-1) - b_2 y(n-2)$$

La transformée en Z donne:

$$Y(Z)(1 + b_1 Z^{-1} + b_2 Z^{-2}) = X(Z)$$

d'où:

$$H(Z) = \frac{1}{1 + b_1 Z^{-1} + b_2 Z^{-2}} = \frac{Z^2}{Z^2 + b_1 Z + b_2}$$

Deux cas sont alors possibles:

$\Delta = b_1^2 - 4b_2 \geq 0$, la fonction de transfert possède alors deux pôles réels et elle est identique à la mise en cascade de deux cellules du premier ordre. La fonction de transfert globale est donc monotone.

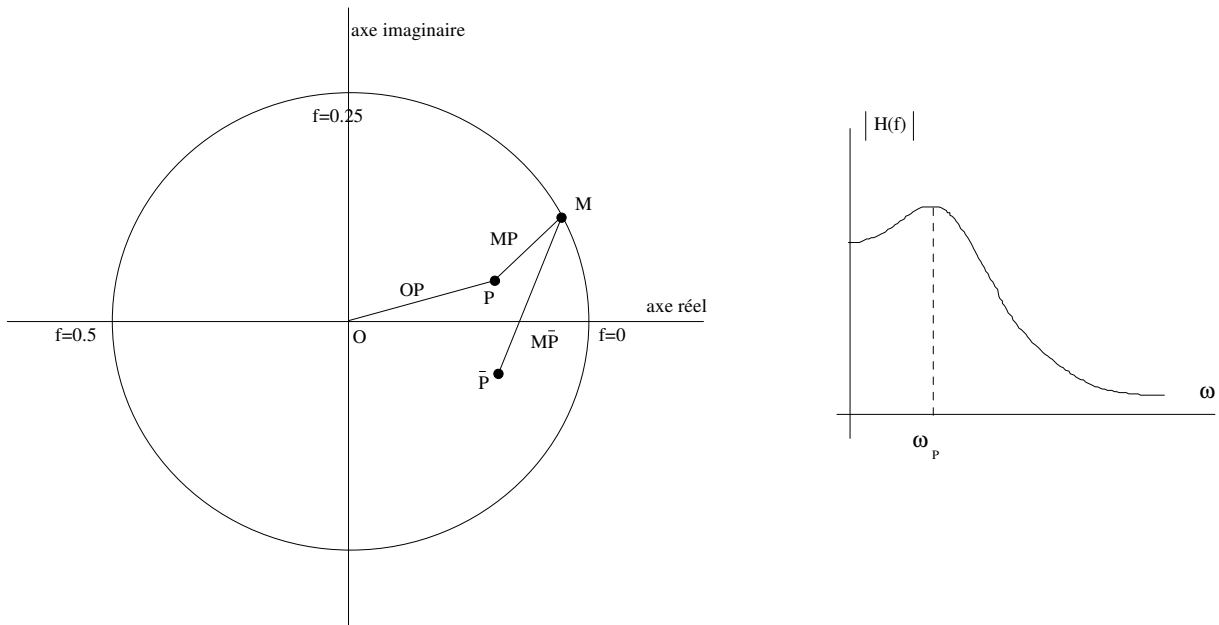
$\Delta = b_1^2 - 4b_2 < 0$, la fonction de transfert possède alors deux pôles complexes conjugués:

$$P = \frac{-b_1}{2} \pm j \frac{\sqrt{-\Delta}}{2}$$

d'où:

$$b_1 = -2 \operatorname{Re}(P) \text{ et } b_2 = |OP|^2$$

$$H(Z) = \frac{1}{MP \cdot \overline{MP}}$$



La fonction de transfert possède une fréquence de résonance et n'est plus strictement monotone.

3.5.3 Cellule générale du second ordre

$$y(n) = a_0 x(n) + a_1 x(n-1) + a_2 x(n-2) - b_1 y(n-1) - b_2 y(n-2)$$

$$H(Z) = \frac{a_0 + a_1 Z^{-1} + a_2 Z^{-2}}{1 + b_1 Z^{-1} + b_2 Z^{-2}}$$

Deux cas particuliers méritent d'être détaillés:

- Le filtre fréquentiel:

Si l'objectif de la fonction de transfert est de filtrer certaines fréquences présentes dans un signal, les zéros du numérateur vont se trouver sur le cercle unité. La fonction de transfert s'écrit alors:

$$H(Z) = a_0 \frac{(Z - Z_0)(Z - \bar{Z}_0)}{(Z - P)(Z - \bar{P})}$$

- Le déphaseur pur:

Une cellule du second ordre peut aussi être utilisée pour répondre, non pas à des objectifs de filtrage fréquentiel, mais à des objectifs de déphasage du signal. Ainsi il est possible de réaliser un déphaseur pur avec une cellule de ce type. Pour cela il suffit d'utiliser un numérateur et un dénominateur image l'un de l'autre.

$$H(Z) = \frac{a_0 + a_1 Z^{-1} + a_2 Z^{-2}}{a_2 + a_1 Z^{-1} + a_0 Z^{-2}}$$

Il est facile de vérifier que:

$$H(Z) = \frac{Z^{-2} D(Z^{-1})}{D(Z)}$$

d'où:

$$|H(Z)|^2 = H(Z)H(\bar{Z}) = \frac{Z^{-2} D(Z^{-1})}{D(Z)} \frac{Z^2 D(Z)}{D(Z^{-1})} = 1$$

et:

$$\varphi(\omega) = 2\varphi_D(\omega) - 2\omega$$

avec $\varphi_D(\omega)$ égal au déphasage du dénominateur $D(Z^{-1})$.

3.5.4 Synthèse des filtres à réponse impulsionnelle infinie

La synthèse des filtres à réponse impulsionnelle infinie utilise des fonctions modèle définie en p et procède par transformation bilinéaire de ces dernières.

3.5.4.1 Rappel de la transformation bilinéaire:

Soit le filtre analogique défini par l'équation suivante:



$$y'(t) = by(t) + x(t)$$

En appliquant la transformée de Laplace: $\int_0^{+\infty} y'(t)e^{-pt} dt = pY(p)$

il vient:

$$pY(p) = bY(p) + X(p)$$

d'où:

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

En exprimant la fonction de transfert du même filtre en numérique, il vient:

$$y(nT_e) = y(n-1)T_e + \int_0^{T_e} y'((n-1)T_e + \tau) d\tau$$

Le calcul de l'intégrale par la formule du trapèze conduit alors à:

$$y(nT_e) - y(n-1)T_e = \frac{T_e}{2} (y'(nT_e) + y'(n-1)T_e)$$

d'où:

$$y(nT_e) - y(n-1)T_e = \frac{T_e}{2} (by(nT_e) + x(nT_e) + by(n-1)T_e + x(n-1)T_e)$$

et par transformée en Z:

$$Y(Z) \left((1 - Z^{-1}) - \frac{bT_e}{2} (1 + Z^{-1}) \right) = \frac{T_e}{2} (1 + Z^{-1}) X(Z)$$

d'où la fonction de transfert en Z:

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

$$H(Z) = \frac{1}{\frac{2}{T_e} \frac{(1 - Z^{-1})}{(1 + Z^{-1})} - b}$$

En identifiant avec la fonction de transfert en p:

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

il est possible de faire l'approximation suivante:

$$p \rightarrow Z \qquad Z \rightarrow p$$

$$p = \frac{2}{T_e} \frac{1-Z^{-1}}{1+Z^{-1}} \quad \text{et} \quad Z = \frac{\frac{2}{T_e} + p}{\frac{2}{T_e} - p}$$

Cette approximation constitue la transformation bilinéaire.

Propriétés de la transformation bilinéaire:

- Transformation du cercle unité:

$$\text{si } Z = e^{j\omega T_e} \Rightarrow p = \frac{2}{T_e} \frac{1 - e^{-j\omega T_e}}{1 + e^{-j\omega T_e}} = \frac{2}{T_e} \frac{e^{+j\omega \frac{T_e}{2}} - e^{-j\omega \frac{T_e}{2}}}{e^{+j\omega \frac{T_e}{2}} + e^{-j\omega \frac{T_e}{2}}} = j \frac{2}{T_e} \operatorname{tg}\left(\frac{\omega T_e}{2}\right)$$

Le cercle unité est donc transformée en axe imaginaire.

- Déformation fréquentielle:

Au lieu d'obtenir :

$$p = j\omega$$

la transformée bilinéaire a conduit à :

$$p = j \frac{2}{T_e} \operatorname{tg}\left(\frac{\omega T_e}{2}\right)$$

en posant f la fréquence vraie et f_d la fréquence déformée par la transformation bilinéaire il vient:

$$f = \frac{1}{\pi T_e} \operatorname{tg}(\pi T_e f_d)$$

La transformation bilinéaire entraîne donc une déformation en fréquence qui est d'autant plus importante que la fréquence est élevée.

Grâce à la transformation bilinéaire la synthèse des filtres numériques de type RII se résume à approcher le gabarit désiré par des fonctions modèles définies en p puis à transformer ces dernières pour obtenir directement les coefficients du filtre.

3.5.4.2 Les fonctions modèle

Plusieurs fonctions modèles permettent d'approcher au mieux un gabarit demandé. Les plus célèbres d'entre elles sont des fonctions de Butterworth, elliptiques ou des polynômes de Tchebycheff.

3.6 Relations entre le nombre de coefficients et le gabarit

Il est possible d'estimer le nombre de coefficients dont aura besoin un filtre RII en fonction du gabarit demandé au moyen de la formule suivante:

$$N \approx 1.08 \log_{10} \left[\frac{2}{\delta_2 \sqrt{\delta_1}} \right] \log_{10} \left[\frac{F_e}{\Delta f} \frac{4}{\pi} \sin \left(2\pi \frac{f_l}{F_e} \right) \right]$$

Remarque

Un certain nombre d'écritures en Z vont être présentées dans ce polycopié, on insistera donc sur le point suivant :

Lorsqu'un filtre de type RII est défini par ses pôles et zéros, il faut être prudent au moment de reconstruire les coefficients avec lesquels on va filtrer le signal.

Deux solutions sont possibles :

- soit on reconstruit le polynôme en Z , on développe puis on repasse en Z^{-1} pour bien identifier les coefficients.
- soit on reconstruit directement le polynôme en Z^{-1} .

Exemple :

Soit le filtre d'ordre 2 défini par deux zéros Z_0 et Z_0^* et par ses deux pôles P_0 et P_0^* .

Reconstruction en Z :

$$H(Z) = \frac{(Z - Z_0)(Z - Z_0^*)}{(Z - P_0)(Z - P_0^*)} = \frac{Z^2 - 2\text{Re}(Z_0)Z + Z_0Z_0^*}{Z^2 - 2\text{Re}(P_0)Z + P_0P_0^*} = \frac{1 - 2\text{Re}(Z_0)Z^{-1} + Z_0Z_0^*Z^{-2}}{1 - 2\text{Re}(P_0)Z^{-1} + P_0P_0^*Z^{-2}}$$

D'où l'équation de filtrage :

$$y(n) = x(n) - 2\text{Re}(Z_0)x(n-1) + Z_0Z_0^*x(n-2) + 2\text{Re}(P_0)y(n-1) - P_0P_0^*y(n-2)$$

Ou reconstruction en Z^{-1} :

$$H(Z) = \frac{(1 - Z_0Z^{-1})(1 - Z_0^*Z^{-1})}{(1 - P_0Z^{-1})(1 - P_0^*Z^{-1})} = \frac{1 - 2\text{Re}(Z_0)Z^{-1} + Z_0Z_0^*Z^{-2}}{1 - 2\text{Re}(P_0)Z^{-1} + P_0P_0^*Z^{-2}}$$

D'où l'équation de filtrage :

$$y(n) = x(n) - 2\text{Re}(Z_0)x(n-1) + Z_0Z_0^*x(n-2) + 2\text{Re}(P_0)y(n-1) - P_0P_0^*y(n-2)$$

NE MELANGER LES DEUX EN AUCUN CAS

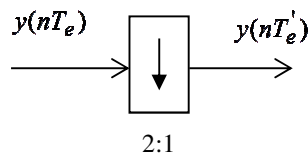
Faux

~~$$H(Z) = \frac{(Z^{-1} - Z_0)(Z^{-1} - Z_0^*)}{(Z^{-1} - P_0)(Z^{-1} - P_0^*)}$$~~

3.7 Applications du filtrage numérique

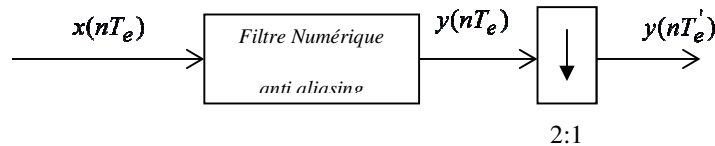
3.7.1 Décimation

Il arrive souvent qu'une chaîne de traitement numérique d'un signal fonctionne avec différents rythmes d'échantillonnage. Lorsque la fréquence d'échantillonnage décroît on parle alors de décimation. L'opération de décimation est triviale, il suffit de supprimer un certain nombre d'échantillons. Elle est en général symbolisée par une flèche orientée de haut en bas.



Le schéma ci-dessus représente une décimation par 2, pour laquelle il suffira de supprimer un échantillon sur deux.

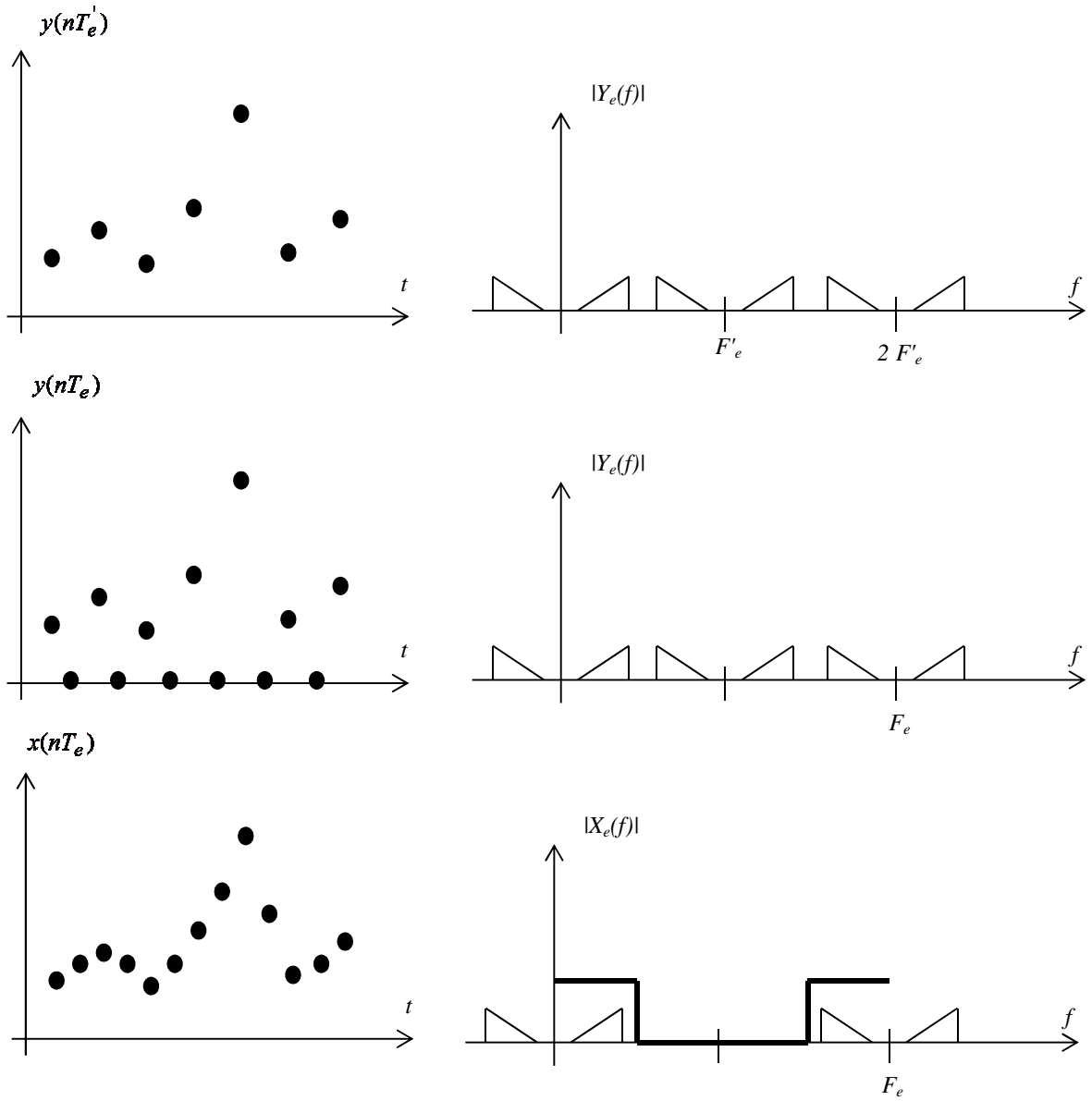
La fréquence d'échantillonnage passera ainsi de F_e à $F'_e = \frac{F_e}{2}$. Avant d'effectuer une telle opération il faut s'assurer que le théorème de Shannon reste vérifié. Il est donc nécessaire de restreindre la bande B du signal afin qu'elle ne dépasse pas $B \leq \frac{F'_e}{2} = \frac{F_e}{4}$. Ce filtrage "anti aliasing" est cette fois réalisé en numérique au cœur des traitements, c'est la différence essentielle avec le filtrage anti aliasing "traditionnel" réalisé en analogique avant l'opération d'échantillonnage.



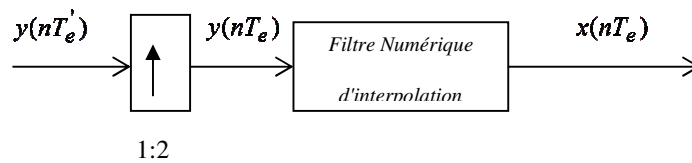
3.7.2 Interpolation

L'opération duale de la décimation est l'opération d'interpolation. Pour l'effectuer on utilise un filtre numérique et une insertion de zéros au milieu du signal d'origine. Considérons, pour l'exposer, le cas d'une interpolation par un facteur 2 d'un signal $y(nT'_e)$. On commence par insérer une valeur nulle entre chaque valeur du signal $y(nT'_e)$. La fréquence d'échantillonnage est alors doublée, on a maintenant $F_e = 2F'_e$. La forme du spectre du signal est inchangée, les valeurs insérées étant des zéros. Cependant ce spectre ne correspond pas à celui que l'on aurait obtenu en échantillonnant réellement le signal analogique avec F_e . Il y a en effet trop de répétitions du motif au niveau du spectre. Il suffit alors simplement de supprimer ces motifs au moyen d'un filtre numérique pour obtenir le spectre du

signal numérique, comme si il avait été échantillonné d'entrée à la fréquence F_e . Ce filtrage numérique correspond à une opération d'interpolation des valeurs temporelles du signal.



L'opération d'interpolation va se symboliser par :



En combinant la décimation et l'interpolation il est possible d'effectuer des modification fractionnaires de la fréquence d'échantillonnage.

3.7.3 Filtres de Nyquist

Dans le domaine des communications numériques, le filtrage numérique joue un rôle important. C'est en particulier le cas lors d'une transmission numérique où les symboles à transmettre sont mis en forme au moyen d'un filtre numérique qui ne doit pas créer d'interférence entre symboles. Les conditions que doit satisfaire ce filtre pour que l'interférence intersymbole soit nulle, ont été énoncées pour la première fois par Nyquist et constituent le "premier critère de Nyquist". Ces conditions peuvent s'énoncer dans le domaine temporel ou dans le domaine fréquentiel.

Dans le domaine temporel, elles s'appliquent à la réponse impulsionnelle $h(t)$ du filtre :

$$\begin{cases} h(0) = 1, \\ h(nT_s) = 0, n \text{ entier } \neq 0 \end{cases}$$

On peut formaliser cette propriété en écrivant : $h(t) \sum_n \delta(t - nT_s) = \delta(t)$

Par Transformée de Fourier de cette équation, on obtient alors : $H(f) * \frac{1}{T_s} \sum_k \delta(f - \frac{k}{T_s}) = 1$

ou encore : $\sum_k H(f - \frac{k}{T_s}) = T_s$

On considère alors que le filtre $H(f)$ a une fréquence de coupure $\frac{1}{T_s}$ telle que : $H(f) = 0$ pour $|f| \geq \frac{1}{T_s}$

L'expression précédente devient alors : $H(f) + H(f - \frac{1}{T_s}) = T_s$

En considérant que le filtre est à phase linéaire, l'égalité devient : $|H(f)| + |H(f - \frac{1}{T_s})| = T_s$

Le **premier critère de Nyquist** s'exprime donc , dans le domaine temporel de la manière suivante :

Un filtre passe bas H de fréquence de coupure $\leq \frac{1}{T_s}$ n'introduit pas d'interférence intersymbole lors de la transmission d'un signal $\sum_k a_k \delta(t - kT_s)$ si sa fonction de transfert $H(f)$ satisfait deux conditions :

1° la phase de $H(f)$ est une fonction linéaire de la fréquence;

2° le module de $H(f)$, c'est à dire le gain en amplitude du filtre est symétrique par rapport au point $\left(\frac{1}{2T_s}, \frac{T_s}{2}\right)$

pour $0 \leq f \leq \frac{1}{T_s}$

Le filtre de fréquence de coupure la plus basse satisfaisant le 1^{er} critère de Nyquist est le filtre passe bas rectangulaire

de fréquence de coupure $f_c = \frac{1}{2T_s}$. La réponse impulsionnelle correspondante est de la forme : $h(t) = \frac{\sin \pi \frac{t}{T_s}}{\pi \frac{t}{T_s}}$

3.7.3.1 Filtre en cosinus surélevé

Une famille de filtres, très importante en communications numériques, est la famille des filtres en cosinus surélevés. Ils ont la particularité de ne pas produire d'interférence entre symboles mais aussi d'être à support spectral borné. Ils permettent ainsi de mettre en forme des signaux de communications pour des canaux à bande limitée, tout en préservant les signaux de l'interférence entre symboles. D'un point de vue théorique ils demandent cependant une infinité de coefficients. Heureusement on peut tronquer leur réponse impulsionnelle sans provoquer de pertes de performances trop importantes (en ce qui concerne les lobes secondaires du spectre). L'analyse fine de ces filtres particuliers sort du cadre de ce cours. Seules la forme analytique de la réponse impulsionnelles de ce filtre sera donnée ci-dessous.

Forme de la réponse impulsionnelle à temps continu du filtre en cosinus surélevé : $h(t) = \frac{\sin\left(\frac{\pi t}{T_s}\right)}{\frac{\pi t}{T_s}} \left[\frac{\sin\left(\beta \frac{\pi t}{T_s}\right)}{1 - 4\left(\frac{\beta t}{T_s}\right)^2} \right]$

Dans le cas d'un modem de communications numériques avec N échantillons par symboles et une période d'échantillonnage T_e , la réponse impulsionnelle échantillonnée (donc les coefficients du filtre) est donnée par :

$$h(k T_e) = \frac{\sin\left(\frac{\pi k}{N}\right)}{\frac{\pi k}{N}} \left[\frac{\sin\left(\beta \frac{\pi k}{N}\right)}{1 - 4\left(\frac{\beta k}{N}\right)^2} \right]$$

Le coefficient β , appelé roll-off du filtre est un coefficient réel positif qui varie entre 0 et 1.

ex: $T_s = 3T_e$

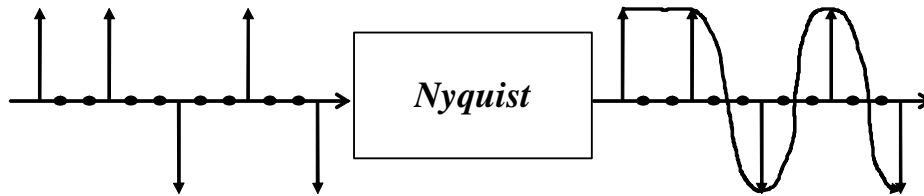


Illustration de l'effet temporel du filtre en racine de cosinus surélevé (absence d'interférences entre symboles aux instants d'échantillonnage)

4. Les signaux aléatoires

4.1 Introduction

La démarche la plus intuitive et naturelle pour représenter un signal consiste à exprimer la valeur de ce dernier au cours du temps. La variable choisie est alors le temps t et le signal s'exprime sous la forme d'une fonction $x(t)$.

Lorsque la fonction $x(t)$ est parfaitement déterminée, on parle de signal déterministe. De tels signaux ont été rencontrés à de nombreuses occasions lors des cours de traitement du signal précédents (en première année du cycle ingénieur en particulier).

On s'est ainsi intéressé aux signaux sinusoïdaux : $x(t) = \sin(\omega t)$ ou à des signaux échelons : $x(t) = U(t)$ ou encore à d'autres types de signaux.

Cependant, lorsque l'on s'intéresse à des signaux de la vie réelle, comme la valeur d'une tension aux bornes d'une résistance ou à la valeur du courant parcourant une antenne ou à d'autres exemples, alors on conçoit bien que les valeurs du signal observé vont être fonctions d'une multitude de phénomènes. Si l'on reprend l'exemple de la résistance, la valeur de la tension va varier continuellement autour d'une valeur moyenne en fonction de l'agitation des électrons, pour une antenne l'environnement électromagnétique va être responsable de nombreuses variations et l'on conçoit intuitivement qu'il est impossible de déterminer de manière tout à fait exacte la valeur du signal $x(t)$.

On formalise alors le signal comme étant une variable aléatoire qui évolue dans le temps et l'on parle de processus aléatoire. En formalisant on pourrait introduire la variable aléatoire $X(t)$ et la distinguer de $x(t)$ qui représente la valeur prise (réalisation) par cette variable aléatoire à l'instant t . Dès lors l'ensemble des valeurs du signal $x(t)$ est considéré comme étant une réalisation particulière d'un processus aléatoire (on parle aussi de trajectoire pour l'ensemble des valeurs prises par un signal au cours du temps).

Dès que l'on introduit cette notion de variable aléatoire on conçoit que l'on va s'intéresser très rapidement aux probabilités liées à cette variable aléatoire, à sa densité de probabilité si la

variable est continue et enfin à ses moments statistiques (moyenne, covariance, et éventuellement des moments d'ordres plus élevés).

Un exemple typique, qui sera repris plus en détail dans la suite du cours, est le cas du bruit blanc gaussien que l'on rencontre très souvent dans les problèmes de traitement du signal et de communications numérique. Or ce signal particulier est justement défini par les adjectifs *blanc* et *gaussien* qui caractérisent parfaitement sa densité de probabilité (gaussien) et ses moments d'ordre deux (blanc).

Partant dorénavant de signaux aléatoires, ce qui est le cas le plus général que l'on puisse envisager, nous allons dès maintenant considérer un sous ensemble de ces signaux qui seront les signaux stationnaires.

Définition de la stationnarité :

Un signal aléatoire est défini à chaque instant t par la loi de probabilité de son amplitude $X(t)$. Cette loi de probabilité peut s'exprimer par une densité de probabilité $p_X(x,t)$ définie de la manière suivante :

$$p_X(x,t) = \lim_{\Delta x \rightarrow 0} \frac{\text{Prob}[x \leq X(t) \leq x + \Delta x]}{\Delta x} \quad (64.)$$

Le signal est stationnaire si ses propriétés statistiques sont indépendantes du temps, c'est à dire, si sa densité de probabilité est indépendante du temps :

$$p_X(x,t) = p_X(x) \quad (65.)$$

Définition d'un signal du second ordre :

Le signal sera dit du second ordre s'il possède un moment d'ordre 1 appelé valeur moyenne, qui est l'espérance mathématique de $X(t)$, notée $E[X(t)]$ et définie par :

$$m_1(t) = E[X(t)] = \int_{-\infty}^{+\infty} xp_X(x,t)dx \quad (66.)$$

et un moment d'ordre 2, appelé fonction de covariance :

$$m_2(t_1, t_2) = E[X(t_1), X(t_2)] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x_1 \cdot x_2 p_X(x_1, x_2; t_1, t_2) dx_1 dx_2 \quad (67.)$$

où $p_X(x_1, x_2; t_1, t_2)$ est la densité de probabilité du couple de variables aléatoires $[X(t_1), X(t_2)]$.

Définition de la stationnarité à l'ordre 2 :

Le caractère de stationnarité peut être limité aux moments du premier et du second ordre, on dit alors que le signal est stationnaire à l'ordre 2. On a alors :

$$m_1 = E[X(t)] = \int_{-\infty}^{+\infty} x \cdot p_X(x) dx, \quad m_1 \text{ constante indépendante du temps}$$

Pour l'ordre 2, l'indépendance du temps s'écrit :

$$p_X(x_1, x_2; t_1, t_2) = p_X(x_1, x_2; 0, t_2 - t_1) = p_X(x_1, x_2; \tau) \text{ avec } \tau = t_2 - t_1 \quad (68.)$$

Seul intervient l'écart entre les deux instants d'observation. On introduit alors la fonction d'autocorrélation $r_{XX}(\tau)$ du signal aléatoire :

$$r_{XX}(\tau) = E[X(t)X(t - \tau)] \quad (69.)$$

La réalisation $x(t)$ du signal aléatoire $X(t)$ possède aussi une moyenne temporelle m_T définie par :

$$m_T = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^{+T} x(t) dt \quad (70.)$$

Définition de l'ergodicité d'un signal stationnaire à l'ordre 2 :

On dira que le signal est ergodique lorsque que l'on peut confondre la moyenne temporelle m_T avec la moyenne m_1 :

$$m_1 = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^{+T} x(t) dt \quad (71.)$$

et lorsque que l'on peut calculer la fonction d'autocorrélation de la même manière, c'est à dire :

$$r_{xx}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^{+T} x(t)x(t-\tau)dt \quad (72.)$$

Ce résultat a des conséquences pratiques très importantes car il va permettre d'accéder aux propriétés statistiques du signal à un instant donné à partir de l'observation de ce signal au cours du temps.

Les développements précédents ont essentiellement concerné des signaux à valeurs réelles. Ces signaux correspondent intuitivement et physiquement à la plupart des signaux rencontrés. Lorsque l'on considère une valeur de tension ou de courant et que l'on se place derrière un convertisseur analogique numérique, les signaux sont bien entendu réels. Cependant, dans bien des cas, en particulier en communication numériques, on sera amené à traiter des enveloppes complexes de signaux modulés (voir aussi la définition du signal analytique). Il est donc nécessaire, afin de ne pas restreindre la généralité de la suite des algorithmes et méthodes présentées dans ce cours, de considérer dorénavant des signaux à valeurs complexes.

De plus on a jusqu'alors bien séparé le processus $X(t)$ et sa réalisation $x(t)$. Cependant afin d'éviter de confondre le processus et la Transformée de Fourier $X(f)$ de sa réalisation, on abandonnera à partir de maintenant la notation $X(t)$. Le lecteur avisé sera à même de comprendre les écritures du type $E[x(t)]$ comme l'espérance de la variable aléatoire.

Note :

On notera souvent la puissance du signal $x(t)$ sous la forme σ_x^2 . On écrira donc $\sigma_x^2 = E[|x(t)|^2]$, ou encore $\sigma_x^2 = r_{xx}(0)$.

Pour des signaux de moyenne nulle (dits signaux centrés) on justifie cette notation qui est en général retenue pour écrire la variance du processus. (rappel $\sigma^2 = E[X^2] - (E[X])^2$).

Résumé du paragraphe :

Le signal aléatoire $x(t)$ est une réalisation d'un processus aléatoire.

Dans ce cours le signal sera considéré comme ergodique et stationnaire à l'ordre 2, c'est à dire :

$$E[x(t)] = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^{+T} x(t) dt = m$$

$$E[x(t)x^*(t - \tau)] = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^{+T} x(t)x^*(t - \tau) dt = r_{xx}(\tau)$$

4.2 Densité Spectrale de Puissance

4.2.1 Problématique de l'Analyse spectrale

La représentation du signal sous la forme $x(t)$ est une démarche naturelle mais elle ne correspond pas forcément à la meilleure représentation physique des signaux rencontrés. En effet, l'individu ou les systèmes électroniques sont souvent plus sensibles à la puissance et à la fréquence des signaux et la représentation du signal sous la forme de sa répartition de puissance en fonction de la fréquence permet, dans bien des cas, d'extraire de manière plus immédiate l'information qui réside dans ce dernier. Le signal est alors représenté par une fonction $P(f)$ appelée densité spectrale de puissance. Le passage de $x(t)$ à $P(f)$ constitue l'*Analyse Spectrale*.

Il existe deux grandes classes de méthodes pour estimer la densité spectrale de puissance d'un signal $x(t)$. La première, l'*estimation spectrale non paramétrique*, n'utilise aucune connaissance a priori sur le signal et part uniquement de l'observation de ce dernier. La deuxième, l'*estimation spectrale paramétrique*, utilise un modèle paramétrique décrivant le signal, modèle à partir duquel il est aisé d'obtenir la densité spectrale de puissance. Les paramètres du modèle sont adaptés en fonction du signal observé. Entre ces deux méthodes il existe une troisième

classe d'approches qui suppose que le signal est composé d'un certain nombre de raies spectrales dont il convient de trouver les fréquences et les puissances. Ce type de méthodes sera classé dans ce cours sous l'appellation d'*estimation spectrale par décomposition harmonique*.

4.2.2 Estimation Spectrale non paramétrique

La Transformée de Fourier de la fonction d'autocorrélation $r_{xx}(\tau)$ du processus aléatoire $x(t)$ stationnaire à l'ordre 2 s'écrit:

$$P(f) = \int_{-\infty}^{+\infty} r_{xx}(\tau) e^{-j2\pi f\tau} d\tau \quad (73.)$$

elle est définie comme étant la densité spectrale de puissance $P(f)$ du processus $x(t)$.

En effet, pour $\tau = 0$, il vient:

$$r(0) = E[x(t)^2] = \int_{-\infty}^{+\infty} P(f) e^{j2\pi f \cdot 0} df = \int_{-\infty}^{+\infty} P(f) df \quad (74.)$$

En supposant l'hypothèse ergodique vérifiée, l'espérance mathématique s'écrit:

$$E[x(t)^2] = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^{+T} x(t)^2 dt = \int_{-\infty}^{+\infty} P(f) df \quad (75.)$$

$P(f)$ représente donc bien la densité spectrale de puissance du processus $x(t)$. Il s'agit là du théorème de Wiener Kintchine.

L'hypothèse ergodique permettant de confondre l'espérance mathématique avec une moyenne temporelle, le développement de $r_{xx}(\tau)$ donne :

$$P(f) = \int_{-\infty}^{+\infty} \left[\lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^{+T} x(t) x^*(t - \tau) \right] e^{-j2\pi f\tau} d\tau \quad (76.)$$

On peut aussi introduire le spectre complexe de la réalisation tronquée du processus $x(t)$:

$$X_{2T}(f) = \int_{-T}^{+T} x(t) e^{-j2\pi ft} dt \quad (77.)$$

En prenant le carré de cette expression, il vient :

$$\frac{1}{2T} |X_{2T}(f)|^2 = \frac{1}{2T} \int_{-T}^{+T} \int_{-T}^{+T} x(u) x^*(v) e^{-j2\pi fu} e^{+j2\pi fv} dudv \quad (78.)$$

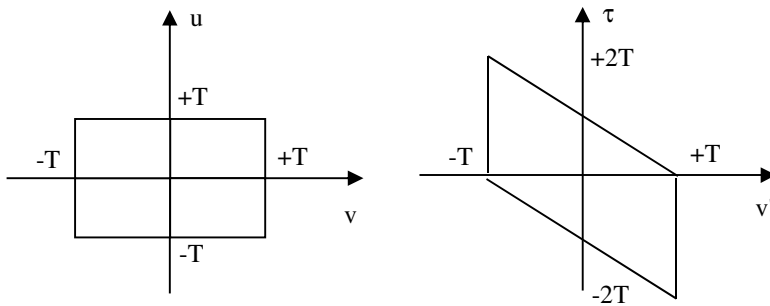
et en prenant l'espérance de cette expression, on obtient :

$$E \left[\frac{1}{2T} |X_{2T}(f)|^2 \right] = \frac{1}{2T} \int_{-T}^{+T} \int_{-T}^{+T} r_{xx}(u-v) e^{-j2\pi f(u-v)} dudv \quad (79.)$$

En effectuant alors le changement de variable suivant

$$\begin{cases} \tau = u - v \\ v' = v \end{cases}$$

et en prenant garde aux intervalles de variation des nouvelles variables, il vient :



Pour $\tau \geq 0$, v' varie de $-T$ à $T - \tau$ et pour $\tau \leq 0$, v' varie de T à $-T - \tau$

$$E \left[\frac{1}{2T} |X_{2T}(f)|^2 \right] = \frac{1}{2T} \int_{\tau=0}^{2T} \int_{v'=-T}^{T-\tau} r_{xx}(\tau) e^{-j2\pi f\tau} d\tau dv' + \int_{\tau=0}^{-2T} \int_{v'=T}^{-T-\tau} r_{xx}(\tau) e^{-j2\pi f\tau} d\tau dv' \quad (80.)$$

$$E\left[\frac{1}{2T}|X_{2T}(f)|^2\right] = \frac{1}{2T} \int_{\tau=0}^{2T} r_{xx}(\tau)(2T-|\tau|)e^{-j2\pi f\tau} d\tau + \int_{\tau=0}^{-2T} r_{xx}(\tau)(-2T+|\tau|)e^{-j2\pi f\tau} d\tau \quad (81.)$$

d'où finalement :

$$E\left[\frac{1}{2T}|X_{2T}(f)|^2\right] = \int_{\tau=-2T}^{2T} r_{xx}(\tau)\left(1-\frac{|\tau|}{2T}\right)e^{-j2\pi f\tau} d\tau \quad (82.)$$

En introduisant alors la fonction I_{4T} indicatrice de l'intervalle $[-2T, +2T]$, il vient :

$$E\left[\frac{1}{2T}|X_{2T}(f)|^2\right] = TF \left[r_{xx}(\tau) \left(1 - \frac{|\tau|}{2T}\right) I_{4T} \right] \quad (83.)$$

On obtient donc :

$$E\left[\frac{1}{2T}|X_{2T}(f)|^2\right] = P(f) * TF \left[\left(1 - \frac{|\tau|}{2T}\right) I_{4T} \right] \quad (84.)$$

Lorsque T tend vers l'infini, le deuxième terme du produit de convolution tend vers $\delta(f)$, d'où :

$$\lim_{T \rightarrow \infty} E\left[\frac{1}{2T}|X_{2T}(f)|^2\right] = P(f) \quad (85.)$$

En considérant alors un cas numérique, où l'observation du signal $x(t)$ se résume à N valeurs échantillonnées à la période T_e , la densité spectrale peut être **estimée** en limitant l'intégrale précédente et en oubliant l'espérance mathématique.

d'où :

$$P_{per}(f) = \frac{1}{NT_e} \left| \sum_{n=0}^{N-1} x(nT_e) e^{-j2\pi nT_e f} \right|^2 \quad (86.)$$

Cet estimateur de la densité spectrale de la densité spectrale de puissance du signal $x(nT_e)$ est appelé *périodogramme*.

Il demande, pour être calculé, la mise au carré de la Transformée de Fourier du signal numérique $x(nT_e)$ sur N points. Il est, depuis la mise au point de l'algorithme de Transformation

de Fourier Rapide (TFR ou FFT en anglais) par J. Cooley et J. Tuckey en 1965, l'estimateur le plus employé.

Jusqu'à la mise au point de l'algorithme de la FFT, la méthode la plus utilisée consistait à estimer N' valeurs de la fonction d'autocorrélation $r(p)$ avec $N' < N$ et à calculer la Transformée de Fourier Discrète (TFD ou DFT en anglais) sur les N' points obtenus. Cet estimateur de la densité spectrale de puissance, du aux travaux de Blackman et Tuckey, porte le nom de *corrélogramme*.

$$P_{cor}(f) = \sum_{p=-(N'-1)}^{N'-1} r_{xx}(p) e^{-j2\pi pf} \quad (87.)$$

avec :

$$r(p) = \frac{1}{N} \sum_{n=p}^{N-1} x(n)x(n-p)^* \quad (88.)$$

$$\text{pour } p = 0 \rightarrow N'-1$$

Pour $N' = N$ les estimateurs du *périodogramme* et du *corrélogramme* sont identiques.

En effet,

$$P_{cor}(f) = \frac{1}{N} \sum_{p=-(N-1)}^{N-1} \sum_{n=p}^{N-1} x(n)x(n-p)^* e^{-j2\pi pf} \quad (89.)$$

d'où:

$$P_{cor}(f) = \frac{1}{N} \sum_{p=-(N-1)}^{N-1} \sum_{n=p}^{N-1} x(n)e^{-j2\pi pf} x(n-p)^* e^{j2\pi(n-p)f} \quad (90.)$$

expression qui peut encore s'écrire:

$$P_{cor}(f) = \frac{1}{N} \left| \sum_{n=0}^{N-1} x(n)e^{-j2\pi nf} \right|^2 = P_{per}(f) \quad (91.)$$

Les deux estimateurs sont donc bien identiques pour $N' = N$.

Propriétés du périodogramme:

Le périodogramme constituant un estimateur de la densité spectrale de puissance du processus $x(nT_e)$, il est nécessaire d'étudier son biais et sa variance.

- Le calcul de l'espérance de $P_{per}(f)$ donne:

$$E[P_{per}(f)] = E \left[\sum_{p=-(N-1)}^{N-1} r_{xx}(p) e^{-j2\pi p f} \right] \quad (92.)$$

avec : $r(p) = \frac{1}{N} \sum_{n=p}^{N-1} x(n)x(n-p)^*$

d'où:

$$E[P_{per}(f)] = \sum_{p=-(N-1)}^{N-1} r_{xx}(p) \frac{N-|p|}{N} e^{-j2\pi p f} \quad (93.)$$

encore grâce aux propriétés de la Transformée de Fourier:

$$E[P_{per}(f)] = P(f) * N \left(\frac{\sin \pi f N}{N \sin \pi f} \right)^2 \quad (94.)$$

L'estimateur du périodogramme est donc une estimation biaisé de la densité spectrale de puissance $P(f)$ du processus $x(nT_e)$. L'estimateur est en fait le résultat du filtrage, dans le

domaine fréquentiel de $P(f)$ par le filtre $N \left(\frac{\sin \pi f N}{N \sin \pi f} \right)^2$. Lorsque N tend vers l'infini ce filtre

tend vers $\delta(f)$, le périodogramme est donc asymptotiquement sans biais.

- Le calcul de la variance est délicat et conduit, pour des signaux gaussiens à :

$$\text{Var}(P_{per}(f)) = P(f) \left[1 + N \left(\frac{\sin(2\pi f N)}{2\pi f N} \right)^2 \right] \quad (95.)$$

Cette variance peut être diminuée en séparant l'ensemble des N valeurs de $x(nT_e)$ en $\frac{N}{K}$ sous-ensembles de $K < N$ valeurs. Il est alors possible de calculer K estimateurs et de faire la moyenne des K estimateurs obtenus. La variance est alors divisée par K^2 . Cette amélioration de la variance de l'estimateur se paye par une diminution de résolution de ce dernier. En effet la résolution spectrale est en $\frac{1}{NT_e}$ dans le cas de N échantillons et en $\frac{1}{KT_e}$ dans le cas de K échantillons.

Enfin, les lobes secondaires de $N \left(\frac{\sin \pi f n}{\pi f N} \right)^2$ peuvent être atténués en introduisant des fenêtres de pondérations qui vont être appliquées directement sur le signal observé $x(nT_e)$.

En conclusion, le périodogramme est un estimateur de la densité spectrale de puissance qui est d'autant meilleur que le signal est observé sur une longue plage de stationnarité. L'algorithme de la Transformée de Fourier Rapide est bien connu et la plupart des processeurs de signaux sont vendus avec des routines de TFR optimisées. Cet estimateur est donc aisé à utiliser et c'est la raison pour laquelle c'est l'estimateur le plus employé aujourd'hui.

Résumé du paragraphe :

La densité spectrale de puissance est par définition la Transformée de Fourier de la fonction d'autocorrélation du signal.

Pour calculer facilement un estimateur de la densité spectrale de puissance, appelé périodogramme, il suffit :

- d'observer N valeurs du signal $x(nT_e)$,
- de calculer une Transformée de Fourier sur N points avec mise au carré.

4.3 Le bruit blanc

Parmi tous les signaux que l'on rencontrera dans ce cours, l'un d'eux, appelé bruit blanc revêt un intérêt particulier.

Ce signal aléatoire est défini comme une suite d'échantillons aléatoires indépendants $b(n)$ tels que :

$$E[b(n)] = 0$$

$$E[b(n)b(n-p)] = 0, \forall p \neq 0$$

$$E[b(n)^2] = \sigma^2$$

La fonction d'autocorrélation de ce signal est donc nulle partout sauf en $p = 0$. Sa Transformée de Fourier est donc une constante. Par analogie avec un spectre lumineux qui comporterait toutes les longueurs d'onde de la lumière et qui donnerait une lumière blanche, on parle alors de signal blanc.

4.4 Exercices corrigés

Exercice 1

On considère un bruit blanc réel $b(n)$ de variance σ^2 . Ce bruit traverse un filtre numérique $A(Z)$ défini par $A(Z) = 1 - \sqrt{2}Z^{-1} + Z^{-2}$. On note $y(n)$ le signal en sortie du filtre.

1. Calculez la fonction d'autocorrélation $r_{yy}(p)$ du signal $y(n)$ pour $p = 0, p = 1, p = 2$
2. Donnez la forme générale de $r_{yy}(p)$
3. Donnez l'expression de la densité spectrale de puissance $P_y(f)$ du signal $y(n)$. (on donnera l'expression la plus factorisée possible).
4. Calculez la valeur de la densité spectrale de puissance pour la fréquence 0 et pour la fréquence $\frac{3F_e}{8}$

Corrigé

Question 1

$$r_{yy}(0T_e) = 4\sigma^2, r_{yy}(1T_e) = -2\sqrt{2}\sigma^2, r_{yy}(2T_e) = \sigma^2$$

Question 2

$$\forall |p| > 2, r_{xx}(p) = 0 \text{ et } r_{xx}(pT_e) = r_{xx}(-pT_e)$$

Question 3

$$P_y(f) = \sum_{p=-\infty}^{p=+\infty} r_{yy}(pT_e) e^{-j2\pi f p T_e}$$

$$P_y(f) = r_{yy}(0T_e) + 2 \sum_{p=1}^{p=+\infty} r_{yy}(pT_e) \cos(2\pi f p T_e)$$

$$P_y(f) = \left(4 - 4\sqrt{2} \cos(2\pi f T_e) + 2 \cos(4\pi f T_e) \right) \sigma^2$$

Question 4

$$P_y(0) = 6 - 4\sqrt{2}$$

$$P_y\left(\frac{3F_e}{8}\right) = 8$$

Exercice 2

On considère un bruit blanc réel $b(n)$ de variance σ^2 . Ce bruit traverse un filtre numérique $H(Z)$ défini par $H(Z) = \frac{1}{1-\alpha Z^{-1}}$. On note $y(n)$ le signal en sortie du filtre.

1. Calculez la fonction d'autocorrélation $r_{yy}(p)$ du signal $y(n)$ pour $p=0$, $p=1$, $p=2$
2. Donnez la forme générale de $r_{yy}(p)$
3. Donnez l'expression de la densité spectrale de puissance $P_y(f)$ du signal $y(n)$. (on donnera l'expression la plus factorisée possible).
4. Calculez la valeur de la densité spectrale de puissance pour la fréquence 0

Corrigé

Question 1 :

$$y(n) = b(n) + \alpha y(n-1)$$

Terme $r_{yy}(0)$:

$$r_{yy}(0) = E[(b(n) + \alpha y(n-1))^2]$$

Or :

$$E[b(n)y(n-1)] = E[b(n)]E[y(n-1)]$$

et :

$$E[b(n)] = 0$$

Donc :

$$E[x(n)y(n-1)] = 0$$

Dans le carré il reste donc :

$$r_{yy}(0) = E[(b(n))^2] + \alpha^2 E[(y(n-1))^2]$$

Or, comme seul "compte" l'écart entre les termes y dans le calcul de l'espérance, on a :

$$E[(y(n-1))^2] = r_{yy}(0)$$

D'où finalement :

$$r_{yy}(0) = \sigma^2 + \alpha^2 r_{yy}(0)$$

$$r_{yy}(0) = \frac{\sigma^2}{1 - \alpha^2}$$

Terme $r_{yy}(1)$:

$$r_{yy}(1) = E[y(n)y(n-1)]$$

$$r_{yy}(1) = E[(b(n) + \alpha y(n-1))y(n-1)]$$

$$r_{yy}(1) = \alpha E[(y(n-1))^2]$$

$$r_{yy}(1) = \alpha r_{yy}(0)$$

$$r_{yy}(1) = \frac{\alpha \sigma^2}{1 - \alpha^2}$$

Terme $r_{yy}(2)$:

$$r_{yy}(2) = E[y(n)y(n-2)]$$

$$r_{yy}(2) = E[(b(n) + \alpha y(n-1))y(n-2)]$$

$$r_{yy}(2) = \alpha E[y(n-1)y(n-2)]$$

$$r_{yy}(2) = \alpha r_{yy}(1)$$

$$r_{yy}(2) = \alpha^2 r_{yy}(0)$$

$$r_{yy}(2) = \frac{\alpha^2 \sigma^2}{1 - \alpha^2}$$

Question 2

$$r_{yy}(p) = \frac{\alpha^{|p|} \sigma^2}{1 - \alpha^2}$$

Question 3

$$P_y(f) = \sum_{p=-\infty}^{p=+\infty} r_{yy}(pT_e) e^{-j2\pi f p T_e}$$

$$P_y(f) = r_{yy}(0) + 2 \sum_{p=1}^{p=+\infty} r_{yy}(p) \cos(2\pi f p T_e)$$

$$P_y(f) = \frac{\sigma^2}{1 - \alpha^2} \left(1 + 2 \sum_{p=1}^{p=+\infty} \alpha^p \cos(2\pi f p T_e) \right)$$

Question 4

$$P_y(0) = \frac{\sigma^2}{1 - \alpha^2} \left(1 + 2 \sum_{p=1}^{p=+\infty} \alpha^p \right)$$

$$P_y(0) = \frac{\sigma^2}{1 - \alpha^2} \left(2 \sum_{p=0}^{p=+\infty} \alpha^p - 1 \right)$$

$$P_y(0) = \frac{\sigma^2}{1 - \alpha^2} \left(\frac{2}{1 - \alpha} - 1 \right)$$

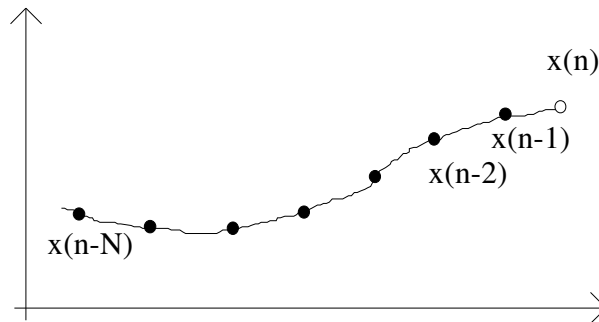
$$P_y(0) = \frac{\sigma^2}{1 - \alpha^2} \left(\frac{1 + \alpha}{1 - \alpha} \right)$$

$$P_y(0) = \frac{\sigma^2}{(1 - \alpha)^2}$$

5. Prédiction linéaire

5.1 Introduction

Dans ce chapitre nous allons considérer que le signal $x(n)$ est supposé être prédictible en fonction d'un certain nombre de ses valeurs antérieures.



Il peut donc s'écrire de la manière suivante :

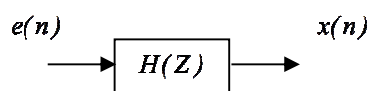
$$x(n) = \sum_{i=1}^N a_i x(n-i) + e(n)$$

équation où les coefficients $(a_i)_{i=1,N}$ constituent les paramètres du modèle et où $e(n)$ est un bruit blanc décorrélé de $x(n)$ de variance σ^2 et qui représente l'erreur de prédiction.

La transformée en Z de cette équation donne alors:

$$X(Z) \left(1 - \sum_{i=1}^N a_i Z^{-i} \right) = E(Z)$$

Le signal $x(n)$ peut donc être vu comme le résultat du passage d'un bruit blanc $e(n)$ de variance σ^2 à travers un filtre de fonction de transfert $H(Z)$.



avec :
$$H(Z) = \frac{1}{1 - \sum_{i=1}^N a_i Z^{-i}}$$

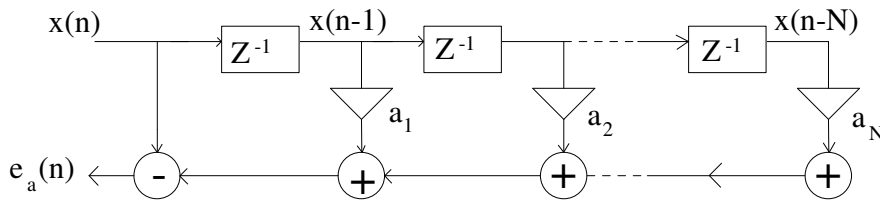
La connaissance des paramètres a_i permet donc de calculer la densité spectrale de puissance $P_{ar}(f)$:

$$P_{ar}(f) = \frac{1}{|1 - \sum_{i=1}^N a_i e^{-j2\pi f T_e}|^2}$$

Il faut donc identifier les paramètres a_i du modèle:

$$x(n) = \sum_{i=1}^N a_i x(n-i) + e_a(n)$$

Ce modèle correspond à la structure RIF suivante:



(l'erreur $e(n)$ est ici notée $e_a(n)$ pour traduire qu'il s'agit d'une erreur avant correspondant à une prédiction de $x(n)$ à partir de $\{x(n-1) \ x(n-2) \ \dots \ x(n-N)\}$)

Il est alors possible d'optimiser un critère de minimisation d'erreur quadratique, c'est à dire, de chercher le jeu de paramètres a_i qui minimisent $E[e_a(n)^2]$.

5.2 Résolution à l'ordre 1

On introduit la fonction $J(a_1)$ à minimiser :

$$J(a_1) = \sum_{m=1}^n (x(m) - a_1 x(m-1))^2$$

Cette fonction est positive convexe, il suffit donc, pour trouver son minimum, de chercher le point d'annulation de sa dérivée en fonction du paramètre a_1 .

$$\frac{\partial J(a_1)}{\partial a_1} = -2 \sum_{m=1}^n x(m-1)(x(m) - a_1 x(m-1))$$

L'annulation de cette dérivée $\frac{\partial J(a_1)}{\partial a_1} = 0$ conduit à :

$$\sum_{m=1}^n x(m-1)(x(m) - a_1 x(m-1)) = 0$$

Ou encore :

$$a_1 = \frac{\sum_{m=1}^n x(m)x(m-1)}{\sum_{m=1}^n x(m-1)^2}$$

On reconnaît alors le ratio de l'autocorrélation d'ordre 1 sur l'autocorrélation d'ordre 0 :

$$a_1 = \frac{r_{xx}(1)}{r_{xx}(0)}$$

5.3 Résolution à l'ordre 2

On introduit la fonction $J(a_1, a_2)$ à minimiser :

$$J(a_1, a_2) = \sum_{m=2}^n (x(m) - a_1 x(m-1) - a_2 x(m-2))^2$$

Cette fonction est positive convexe, il suffit donc, pour trouver son minimum, de chercher le point d'annulation de ses dérivées partielles en fonction des paramètres a_1 et a_2 .

$$\frac{\partial J(a_1, a_2)}{\partial a_1} = -2 \sum_{m=2}^n x(m-1)(x(m) - a_1 x(m-1) - a_2 x(m-2))$$

$$\frac{\partial J(a_1, a_2)}{\partial a_2} = -2 \sum_{m=2}^n x(m-2)(x(m) - a_1 x(m-1) - a_2 x(m-2))$$

L'annulation de ces dérivées partielles conduit à :

$$\sum_{m=2}^n x(m-1)(x(m) - a_1x(m-1) - a_2x(m-2)) = 0$$

$$\sum_{m=2}^n x(m-2)(x(m) - a_1x(m-1) - a_2x(m-2)) = 0$$

Ce système d'équation peut être réorganisé de la manière suivante :

$$\sum_{m=2}^N x(m-1)^2 a_1 + \sum_{m=2}^N x(m-1)x(m-2) a_2 = \sum_{m=2}^N x(m)x(m-1)$$

$$\sum_{m=2}^N x(m-1)x(m-2) a_1 + \sum_{m=2}^N x(m-2)^2 a_2 = \sum_{m=2}^N x(m)x(m-2)$$

On reconnaît alors différents termes de la fonction d'autocorrélation du signal. Le système peut donc être réécrit de la manière suivante :

$$r_{xx}(0)a_1 + r_{xx}(1)a_2 = r_{xx}(1)$$

$$r_{xx}(1)a_1 + r_{xx}(0)a_2 = r_{xx}(2)$$

En introduisant la matrice d'autocorrélation d'ordre 2, définie par :

$$R_2 = \begin{pmatrix} r_{xx}(0) & r_{xx}(1) \\ r_{xx}(1) & r_{xx}(0) \end{pmatrix}$$

Et le vecteur $r_a^2 = \begin{pmatrix} r_{xx}(1) \\ r_{xx}(2) \end{pmatrix}$, le système devient :

$$R_2 \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = r_a^2$$

En introduisant le vecteur $A_2 = [a_1, a_2]^T$ la solution est alors donnée par :

$$A_2 = R_2^{-1} r_a^2$$

5.4 Résolution à l'ordre N

On propose d'écrire directement les équations de manière matricielle. On cherche donc un vecteur de coefficients A_N tel qu'il minimise:

$$E[|e_a(n)|^2] = E[|x(n) - X_{N-1}^T(n-1)A_N|^2]$$

avec :

$$X_N(n-1) = [x(n-1), x(n-2), \dots, x(n-N)]^T$$

et :

$$A_N = [a_1, a_2, \dots, a_N]^T$$

En développant $E[|e_a(n)|^2]$ il vient:

$$E[e_a(n)^2] = E\left[\left(x(n) - X_N^T(n-1)A_N\right)^T \left(x(n) - X_N^T(n-1)A_N\right)\right]$$

$$E[e_a(n)^2] = E\left[x(n)^2 - A_N^T X_N(n-1)x(n) - x(n)X_N^T(n-1)A_N + A_N^T X_N(n-1)X_N^T(n-1)A_N\right]$$

En annulant la dérivée de cette expression par rapport à A_N il vient :

$$A_N = E\left[X_N(n-1)X_N^T(n-1)\right]^{-1} E\left[X_N(n-1)x^*(n)\right]$$

ou encore:

$$A_N = R_N^{-1}r_N^a$$

avec:

$$R_N = \begin{bmatrix} r(0) & r(1) & \cdot & r(N-1) \\ r^*(1) & r(0) & \cdot & \cdot \\ \cdot & \cdot & \cdot & r(1) \\ r^*(N-1) & \cdot & r^*(1) & r(0) \end{bmatrix} \text{ matrice d'autocorrélation du signal et } r_N^a = \begin{bmatrix} r(1) \\ r(2) \\ \cdot \\ r(N) \end{bmatrix}$$

En posant $E_{aN} = E[e_a(n)^2]$ et en utilisant $A_N = R_N^{-1}r_N^a$, il vient:

$$E_{aN} = E[x(n)^2] - 2E\left[\left(X_N^T(n-1)x(n)\right)A_N\right] + A_N^T E\left[X_N(n-1)X_N^T(n-1)\right] A_N$$

$$E_{aN} = E[e_a(n)^2] = E\left[\left(x(n) - X_N^T(n-1)A_N\right)^2\right]$$

$$E_{aN} = r(0) - 2r_N^{aT} A_N + r_N^{aT} A_N$$

$$E_{aN} = r(0) - r_N^{aT} A_N$$

En écrivant A_N et E_{aN} sous une forme matricielle unique, il vient:

$$\begin{bmatrix} r(0) & r_N^{aT} \\ r_N^a & R_N \end{bmatrix} \begin{bmatrix} 1 \\ -A_N \end{bmatrix} = \begin{bmatrix} E_{aN} \\ 0 \end{bmatrix}$$

Cette équation est appelée équation de Yule Walker avant (ou directe ou forward) dans le cas stationnaire

Finalement il est donc possible de trouver le vecteur A_N en inversant la matrice R_N puis de calculer l'énergie d'erreur de prédiction avant. Cependant, le coût de calcul d'une telle approche est alors en $o(N^3)$ ce qui peut s'avérer gênant lorsque N est important. Il est possible de résoudre cette équation avec un coût de calcul proportionnel à $o(N^2)$ en utilisant l'algorithme de Levinson (voir en annexe pour les détails sur cet algorithme).

Résumé du paragraphe :

Pour calculer les coefficients de prédiction linéaire d'un signal, il suffit :

- de calculer la fonction d'autocorrélation du signal
- de former la matrice d'autocorrélation
- d'inverser un système linéaire en utilisant l'équation de Yule Walker

Note : en matlab la fonction s'appelle "yule" et sa syntaxe est : [A,E]=yule(x,N)

6. Filtrage adaptatif

6.1 Introduction Générale

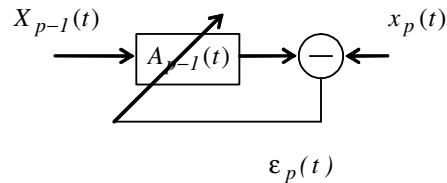
L'attrait que suscite chez les électroniciens l'utilisation du filtrage adaptatif ne tient pas qu'aux intérêts applicatifs de cette technique, pourtant considérables; il s'explique aussi, pour une part non négligeable, par la séduction intellectuelle qu'exerce sur les esprits la mise au point de systèmes artificiels capables de s'adapter de manière indépendante aux diverses situations qu'ils sont appelés à rencontrer. Ce sujet est de nouveau au cœur de l'actualité avec la notion de "deep learning" que l'on rencontre actuellement dans plusieurs secteurs.

Selon les disciplines où elle se rencontre, cette capacité d'adaptation est définie de manière différente. En informatique, par exemple, elle se résume souvent à une aptitude à mettre en œuvre des raisonnements logiques selon un certain nombre de règles prédéfinies ou, éventuellement évolutives; c'est là le domaine de l'intelligence artificielle. En électronique, elle se réduit plutôt à l'adaptation, selon un critère mathématique, des paramètres d'un filtre, le plus souvent numérique; il s'agit alors du domaine du filtrage adaptatif.

Les solutions techniques basées sur l'optimisation de filtres adaptatifs se rencontrent dans une variété de problèmes de détection, d'estimation et d'identification. C'est dans les années 60 et dans le domaine de l'égalisation des voies de transmission de données que des solutions adaptatives ont, pour la première fois, été utilisées à grande échelle. Il existe plusieurs structures possibles, plusieurs types de paramètres et plusieurs critères d'adaptation pour réaliser un filtre adaptatif. Ce cours ne considère que le cas des filtres transverses adaptés selon le critère des moindres carrés.

6.2 Solution de Wiener

Un filtre adaptatif transverse peut être représenté par la structure suivante :



- Schéma général d'un filtre adaptatif -

A partir d'un vecteur de $p - 1$ termes :

$$X_{p-1}(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{p-1}(t) \end{pmatrix} \quad (96.)$$

Le filtre de $p-1$ coefficients :

$$A_{p-1}(t) = \begin{pmatrix} a_1(t) \\ a_2(t) \\ \vdots \\ a_{p-1}(t) \end{pmatrix} \quad (97.)$$

va calculer l'estimée $\hat{x}_p(t)$ du signal $x_p(t)$.

Ce terme $\hat{x}_p(t)$ représente donc la prédiction de $x_p(t)$ à partir de $\{x_1(t) \ x_2(t) \ \dots \ x_{p-1}(t)\}$.

La différence entre $x_p(t)$ et $\hat{x}_p(t)$ est égale à l'erreur de prédiction de $x_p(t)$ à partir de

$\{x_1(t) \ x_2(t) \ \dots \ x_{p-1}(t)\}$. Elle est ici notée $\varepsilon_p(t)$.

L'erreur de prédiction s'écrit finalement :

$$\varepsilon_p(t) = x_p(t) - A_{p-1}^T(t)X_{p-1}(t) \quad (98.)$$

Dans ce chapitre l'exposant T indique que le vecteur $A_{p-1}(t)$ est transposé et conjugué. En général on note plutôt H .

Le critère des moindres carrés consiste à minimiser la fonction de coût $J_p(t)$ suivante :

$$J_p(t) = \sum_{i=0}^t \lambda^{t-i} \varepsilon_p(i) \varepsilon_p^T(i) \quad (99.)$$

Expression dans laquelle le scalaire λ représente un facteur d'oubli réel strictement positif et inférieur à un.

La solution optimale, obtenue au sens de ce critère, est alors donnée par l'expression suivante :

$$A_{p-1}(t) = R_{p-1}^{-1}(t) r_p(t) \quad (100.)$$

expression dans laquelle $R_{p-1}(t)$ représente la matrice d'autocorrélation estimée de $X_{p-1}(t)$ et $r_p(t)$ le vecteur d'intercorrélations estimées de $X_{p-1}(t)$ avec $x_p(t)$:

$$R_{p-1}(t) = \sum_{i=0}^t \lambda^{t-i} X_{p-1}(i) X_{p-1}^T(i) \quad (101.)$$

$$r_p(t) = \sum_{i=0}^t \lambda^{t-i} X_{p-1}(i) x_p^T(i) \quad (102.)$$

la prédiction du signal $x_p(t)$ s'écrit alors :

$$\hat{x}(t) = r_p^T(t) R_{p-1}^{-1}(t) X_{p-1}(t) \quad (103.)$$

La solution optimale requiert finalement un coût de calcul en $O(p-1)^3$ du fait de l'inversion de la matrice $R_{p-1}(t)$.

En développant l'espérance carrée de l'erreur de prédiction, il vient :

$$E[\varepsilon_p(t) \varepsilon_p^T(t)] = E\left[\left(x_p(t) - A_{p-1}^T(t) X_{p-1}(t) \right) \left(x_p(t) - A_{p-1}^T(t) X_{p-1}(t) \right)^T \right]$$

$$E[\mathcal{E}_p(t)\mathcal{E}_p^T(t)] = E\left[\left(x_p(t)x_p^T(t) - A_{p-1}^T(t)X_{p-1}(t)x_p^T(t) - x_p(t)X_{p-1}^T(t)A_{p-1}(t) + A_{p-1}^T(t)X_{p-1}(t)X_{p-1}^T(t)A_{p-1}(t)\right)\right]$$

$$E[\mathcal{E}_p(t)\mathcal{E}_p^T(t)] = E[x_p(t)x_p^T(t)] - 2\operatorname{Re}\left(A_{p-1}^T(t)E[X_{p-1}(t)x_p^T(t)]\right) + A_{p-1}^T(t)E[X_{p-1}(t)X_{p-1}^T(t)]A_{p-1}(t)$$

On introduit alors la matrice $R_{p-1}(t) = E[X_{p-1}(t)X_{p-1}^T(t)]$ d'autocorrélation du vecteur d'entrée $X_{p-1}(t)$ et le vecteur $S_{p-1}(t) = E[X_{p-1}(t)x_p^T(t)]$ d'intercorrélacion entre le vecteur d'entrée $X_{p-1}(t)$ et le signal désiré $x_p(t)$.

L'équation précédente devient alors :

$$E[\mathcal{E}_p(t)\mathcal{E}_p^T(t)] = E[x_p(t)x_p^*(t)] - 2\operatorname{Re}\left(A_{p-1}^T(t)S_{p-1}(t)\right) + A_{p-1}^T(t)R_{p-1}(t)A_{p-1}(t)$$

On cherche le vecteur $A_{p-1}(t)$ des coefficients du filtre qui minimise l'espérance carrée de l'erreur. En considérant que la matrice $R_{p-1}(t)$ est de rang plein, il existe une solution unique qui s'obtient en annulant les dérivées de la forme quadratique $E[\mathcal{E}_p(t)\mathcal{E}_p^T(t)]$ par rapport aux coefficients du filtre $A_{p-1}(t)$.

On décompose alors le filtre $A_{p-1}(t)$ en parties réelle et imaginaire :

$$A_{p-1}(t) = A_{r_{p-1}}(t) + jA_{i_{p-1}}(t)$$

ainsi que le vecteur d'intercorrélacion $S_{p-1}(t)$

$$S_{p-1}(t) = S_{r_{p-1}}(t) + jS_{i_{p-1}}(t)$$

D'où :

$$E[\mathcal{E}_{p-1}(t)\mathcal{E}_{p-1}^T(t)] = E[x_p(t)x_p^T(t)] - 2A_{r_{p-1}}^T S_{r_{p-1}}(t) - 2A_{i_{p-1}}^T S_{i_{p-1}} + A_{p-1}^T(n)R_{p-1}(n)A_{p-1}(n)$$

On considère alors la dérivation de cette expression par rapport à la $k^{\text{ième}}$ composante $a_k(t)$ du vecteur $A_{p-1}(t)$. On décompose la composante $a_k(t)$ en parties réelle et imaginaire :

$$a_k(t) = a_{k_r}(t) + ja_{k_i}(t)$$

et l'on distingue les dérivations par rapport à ces parties réelle et imaginaire.

Remarque :

Si x est une variable complexe $x = a + jb$ avec a et b réels. On définit alors la dérivée par rapport à x par :

$$\frac{\partial}{\partial x} = \frac{1}{2} \left(\frac{\partial}{\partial a} - j \frac{\partial}{\partial b} \right)$$

On a alors $\frac{\partial x}{\partial x} = 1$ et $\frac{\partial x^*}{\partial x} = 0$

On démontre que la dérivation de la forme bilinéaire $A_{p-1}^T(t)R_{p-1}(t)A_{p-1}(t)$ par rapport aux composantes du vecteur $A_{p-1}(n)$ donne les résultats suivants :

$$\frac{\partial A_{p-1}^T(t)R_{p-1}(t)A_{p-1}(t)}{\partial a_{k_r}(t)} = 2 \operatorname{Re} \left(R_{p-1}(t)A_{p-1}(t) \right)_k, \text{ l'indice } k \text{ signifiant que l'on s'intéresse à la } k^{\text{ième}}$$

composante du vecteur $R_{p-1}(t)A_{p-1}(t)$.

et :

$$\frac{\partial A_{p-1}^T(t)R_{p-1}(t)A_{p-1}(t)}{\partial a_{k_i}(t)} = 2 \operatorname{Im} \left(R_{p-1}(t)A_{p-1}(t) \right)_k$$

D'où finalement :

$$\frac{\partial E \left[\varepsilon_{p-1}(t) \varepsilon_{p-1}^T(t) \right]}{\partial a_{k_r}(t)} = -2 \left(S_{r_{p-1}}(t) \right)_k + 2 \operatorname{Re} \left(R_{p-1}(t)A_{p-1}(t) \right)_k$$

et

$$\frac{\partial E \left[\varepsilon_{p-1}(t) \varepsilon_{p-1}^T(t) \right]}{\partial a_{k_i}(t)} = -2 \left(S_{i_{p-1}}(t) \right)_k + 2 \operatorname{Im} \left(R_{p-1}(t)A_{p-1}(t) \right)_k$$

D'où en annulant la dérivée par rapport à $a_k(t)$:

$$-2\left(S_{r_{p-1}}(t)\right)_k + 2 \operatorname{Re}\left(R_{p-1}(t)A_{p-1}(t)\right)_k = 0$$

$$-2\left(S_{i_{p-1}}(t)\right)_k + 2 \operatorname{Im}\left(R_{p-1}(t)A_{p-1}(t)\right)_k = 0$$

En annulant toutes les dérivées partielles il vient donc :

$$-2S_{r_{p-1}}(t) + 2 \operatorname{Re}\left(R_{p-1}(t)A_{p-1}(t)\right) = 0$$

$$-2S_{i_{p-1}}(t) + 2 \operatorname{Im}\left(R_{p-1}(t)A_{p-1}(t)\right) = 0$$

D'où :

$$R_{p-1}(t)A_{p-1}(t) = S_{p-1}(t)$$

et :

$$\boxed{A_{p-1}(t) = R_{p-1}^{-1}(t)S_{p-1}(t)}$$

Cette solution portant le nom de solution de **solution de Wiener-Hopf**.

6.3 Introduction du facteur d'oubli

En considérant alors l'hypothèse ergodique vérifiée, c'est à dire en confondant les espérances mathématiques et les moyennes temporelles, et en introduisant un facteur d'oubli λ dans la moyenne temporelle, afin de munir l'algorithme de propriétés de poursuites, l'espérance mathématique s'écrit :

$$E[x(t)] = \sum_{i=0}^t \lambda^i x(t-i) \text{ avec } \lambda \text{ facteur d'oubli inférieur ou égal à un.}$$

Il est alors immédiat que:

$$R_{p-1}(t+1) = \lambda R_{p-1}(t) + X_{p-1}(t+1)X_{p-1}^T(t+1)$$

$$S_{p-1}(t+1) = \lambda S_{p-1}(t) + X_{p-1}(t+1)x_p^T(t+1)$$

Ainsi, à chaque instant $t+1$, il est possible, à partir du nouveau vecteur $X_{p-1}(t+1)$ et du nouveau scalaire $x_p(t+1)$, de calculer la matrice $R_{p-1}(t+1)$, de l'inverser et de l'appliquer au

vecteur $S_{p-1}(t+1)$. Le vecteur $A_{p-1}(t+1)$ optimal au sens du minimum de l'erreur quadratique pondérée par un facteur d'oubli λ est alors mis à jour.

Le principal inconvénient de cette méthode réside dans l'inversion en temps réel de la matrice $R_{p-1}(t+1)$. Cette inversion demande un coût de calcul proportionnel à p^3 qui peut rapidement s'avérer prohibitif lorsque la dimension p devient importante. Le but des algorithmes des moindres carrés qui vont suivre va être de faire diminuer ce coût de calcul. Pour cela, ils vont utiliser des propriétés récursives de la matrice $R_{p-1}(t)$ et du vecteur $S_{p-1}(t)$. Ces récursions agiront, soit en fonction du temps n , soit en fonction de l'ordre p . Elles vont conduire aux algorithmes des moindres carrés récursifs (MCR) dont le coût de calcul est proportionnel à p^2 . Enfin, dans les cas où les composantes $x_i(t)$ du vecteur $X_{p-1}(t)$ sont en fait les valeurs décalées dans le temps d'un même processus monodimensionnel $x(t)$ l'utilisation de ces propriétés de décalage conduira aux algorithmes rapides dont le coût de calcul est proportionnel à p .

La résolution, à chaque instant $t+1$, de l'équation :

$$A_{p-1}(t+1) = R_{p-1}^{-1}(t+1)S_{p-1}(t+1)$$

n'utilise pas la résolution effectuée à l'instant précédent qui a permis de mettre à jour le vecteur $A_{p-1}(t)$.

Or la matrice $R_{p-1}(t+1)$ a été modifiée, entre l'instant t et l'instant $t+1$, à travers l'équation :

$$R_{p-1}(t+1) = \lambda R_{p-1}(t) + X_{p-1}(t+1)X_{p-1}^T(t+1)$$

Il est alors légitime de chercher à exprimer $A_{p-1}(t+1)$ en fonction de $A_{p-1}(t)$.

Ainsi, en écrivant le filtre optimal au temps $t+1$:

$$A_{p-1}(t+1) = R_{p-1}^{-1}(t+1)S_{p-1}(t+1)$$

et en remplaçant $S_{p-1}(t+1)$ par son expression en fonction de $S_{p-1}(t)$, il vient:

$$A_{p-1}(t+1) = R_{p-1}^{-1}(t+1) \left[\lambda S_{p-1}(t) + X_{p-1}(t+1)x_p^T(t+1) \right]$$

or $S_{p-1}(t) = R_{p-1}(t)A_{p-1}(t)$ d'où:

$$A_{p-1}(t+1) = R_{p-1}^{-1}(t+1) [\lambda R_{p-1}(t)A_{p-1}(t) + X_{p-1}(t+1)x_p^T(t+1)]$$

$$A_{p-1}(t+1) = R_{p-1}^{-1}(t+1) [(R_{p-1}(t+1) - X_{p-1}(t+1)X_{p-1}^T(t+1))A_{p-1}(t) + X_{p-1}(t+1)x_p^T(t+1)]$$

d'où finalement:

$$A_{p-1}(t+1) = A_{p-1}(t) + R_{p-1}^{-1}(t+1)X_{p-1}(t+1)(x_p^T(t+1) - X_{p-1}^T(t+1)A_{p-1}(t))$$

l'expression $x_p^T(t+1) - X_{p-1}^T(t+1)A_{p-1}(t)$ représente la conjuguée de l'erreur de prédiction a priori, faite en utilisant le vecteur $A_{p-1}(t)$, optimal au temps t , pour filtrer le vecteur $X_{p-1}(t+1)$ disponible au temps $t+1$.

En écrivant: $e_p(t+1) = x_p(t+1) - A_{p-1}^T(t)X_{p-1}(t+1)$ cette erreur, l'équation précédente devient:

$$A_{p-1}(t+1) = A_{p-1}(t) + R_{p-1}^{-1}(t+1)X_{p-1}(t+1)e_p^T(t+1)$$

Cette expression, qui porte le nom d'équation normale, permet de mettre à jour le vecteur $A_{p-1}(t+1)$ à partir du vecteur $A_{p-1}(t)$.

6.4 Algorithme du gradient (LMS)

Une première solution consiste à supposer que la matrice $R_{p-1}^{-1}(t+1)$ est constante au cours du temps et égale à δI_{p-1} où I_{p-1} représente la matrice identité et δ un scalaire.

Il s'agit là d'une approximation importante et le critère ne va pas être rigoureusement optimisé. Cependant l'algorithme qui va en découler est sans doute l'algorithme le plus simple à mettre en œuvre et celui dont le coût de calcul est le plus faible. C'est de loin l'algorithme qui a été jusqu'à aujourd'hui le plus utilisé dans les équipements industriels.

Avec cette hypothèse l'équation normale se simplifie et devient :

$$A_{p-1}(t+1) = A_{p-1}(t) + \delta X_{p-1}(t+1)e_p^T(t+1)$$

Le vecteur $X_{p-1}(t+1)$ et le scalaire $x_p(t+1)$ étant disponibles à l'instant $t+1$, il est aisé de calculer: $e_p(t+1) = x_p(t+1) - A_{p-1}^T(t)X_{p-1}(t+1)$ et de mettre à jour le vecteur $A_{p-1}(t+1)$.

Cette mise à jour demande finalement un coût de calcul proportionnel à $O(p-1)$. Cet algorithme est appelé algorithme du gradient stochastique (LMS).

Quantités obtenues au temps n et disponibles au temps $t+1$:

vecteur des coefficients du filtre : $A_{p-1}(t)$

Nouvelles données au temps $n+1$:

vecteur de données reçues : $X_{p-1}(t+1)$

signal de référence : $x_p(t+1)$

Calcul du nouveau filtre au temps $n+1$:

erreur de prédiction avant : $e_p(t+1) = x_p(t+1) - A_{p-1}^T(t)X_{p-1}(t+1)$

mise à jour du filtre au temps $t+1$: $A_{p-1}(t+1) = A_{p-1}(t) + \delta e_p^T(t+1)X_{p-1}(t+1)$

- Algorithme LMS -

Il est à rapprocher de l'algorithme classique du gradient déterministe où un vecteur $A_{p-1}(t+1)$ est adapté en descendant le long de la dérivée d'un critère au moyen d'une formule du type :

$$A_{p-1}(t+1) = A_{p-1}(t) - \frac{\delta}{2} \nabla(\text{critère})$$

en considérant comme critère la minimisation de l'erreur quadratique a priori, cette équation devient :

$$A_{p-1}(t+1) = A_{p-1}(t) - \frac{\delta}{2} \nabla \left(E \left[\left(x_p(t+1) - A_{p-1}^T(t) X_{p-1}(t+1) \right) \left(x_p(t+1) - A_{p-1}^T(t) X_{p-1}(t+1) \right)^T \right] \right)$$

d'où en développant la dérivée :

$$A_{p-1}(t+1) = A_{p-1}(t) + \frac{\delta}{2} \left(2 \left(E \left[X_{p-1}(t+1) x_p^T(t+1) \right] - E \left[X_{p-1}(t+1) X_{p-1}^T(t+1) \right] \right) A_{p-1}(t) \right)$$

Le remplacement des espérances par les valeurs instantanées donne alors :

$$A_{p-1}(t+1) = A_{p-1}(t) + \delta X_{p-1}(t+1) e_p^T(t+1)$$

l'algorithme est alors appelé algorithme du gradient stochastique et le scalaire δ représente le pas de descente de l'algorithme.

Conditions de convergence:

Dans une première approche on peut considérer que l'algorithme est stable dès lors que l'espérance de la valeur absolue de l'erreur a posteriori:

$$|\varepsilon_p(t+1)| = |x_p(t+1) - X_{p-1}^T(t+1) A_{p-1}(t+1)|$$

est inférieure à la valeur absolue a priori:

$$\varepsilon_p(t+1) = x_p(t+1) - X_{p-1}^T(t+1) \left[A_{p-1}(t) + \delta X_{p-1}(t+1) e_p(t+1) \right]$$

Or sachant que:

$$\varepsilon_p(t+1) = x_p(t+1) - X_{p-1}^T(t+1) \left[A_{p-1}(t) + \delta X_{p-1}(t+1) e_p(t+1) \right]$$

il vient:

$$\varepsilon_p(t+1) = e_p(t+1) - \delta X_{p-1}^T(t+1) X_{p-1}(t+1) e_p(t+1)$$

ou encore :

$$\varepsilon_p(t+1) = e_p(t+1) \left(1 - \delta X_{p-1}^T(t+1) X_{p-1}(t+1) \right)$$

La condition de convergence s'écrit alors :

$$E\left\|\varepsilon_p(t+1)\right\| < E\left\|e_p(t+1)\right\|$$

$$\Rightarrow \left|1 - \delta E\left[X_{p-1}^T(t+1)X_{p-1}(t+1)\right]\right| < 1$$

d'où:

$$0 < \delta < \frac{2}{NP_x}$$

où P_x représente la puissance de la composante $x_i(t)$ de puissance maximale du vecteur $X_{p-1}(t)$

(Dans le cas où les composantes de ce vecteur sont les valeurs décalées dans le temps d'un processus monodimensionnel $x(t)$, P_x représente la puissance du processus $x(t)$.)

Une autre approche pour démontrer la stabilité de l'algorithme consiste à introduire le filtre optimal $A_0 = R_{p-1}^{-1}(t+1)S_{p-1}(t+1)$

et la variable $\varepsilon(t+1) = A_{p-1}(t+1) - A_0$

$$\varepsilon(t+1) = A_{p-1}(t) - A_0 + \delta X_{p-1}(t+1)e_p^T(t+1)$$

On introduit l'erreur de filtrage qui aurait été commise avec le filtre optimal :

$$e_0(t+1) = x_p(t+1) - A_0^T X_{p-1}(t+1)$$

D'où :

$$\varepsilon(t+1) = A_{p-1}(t) - A_0 + \delta X_{p-1}(t+1)\left(x_p(t+1) - A_0^T X_{p-1}(t+1)\right)^T$$

$$\varepsilon(t+1) = \varepsilon(t) + \delta X_{p-1}(t+1)\left(e_0(t+1) + A_0^T X_{p-1}(t+1) - A_0^T X_{p-1}(t+1)\right)^T$$

$$\varepsilon(t+1) = \varepsilon(t) + \delta X_{p-1}(t+1)\left(e_0(t+1) - \varepsilon(t)^T X_{p-1}(t+1)\right)^T$$

$$\varepsilon(t+1) = \left(I_{p-1} - \delta X_{p-1}(t+1)X_{p-1}^T(t+1)\right)\varepsilon(t) + \delta X_{p-1}(t+1)e_0^T(t+1)$$

Par définition de $e_0(t+1)$, on a $E[X_{p-1}(t+1)e_0^T(t+1)] = 0$ car $e_0(t+1)$ représente l'innovation de $x_p(t)$ par rapport à la base de prédiction $X_{p-1}(t+1)$.

En prenant alors l'espérance mathématique de l'équation précédente, il vient :

$$E[\varepsilon(t+1)] = (I_{p-1} - \delta R_{p-1}(t+1))E[\varepsilon(t)]$$

L'algorithme LMS convergera en moyenne si et seulement si $E[A_{p-1}(t)] \xrightarrow{t \rightarrow \infty} A_0$

donc si $E[\varepsilon(t)] \xrightarrow{t \rightarrow \infty} 0$

Donc si $0 < \delta < \frac{2}{\lambda_{\max}}$ où λ_{\max} représente la valeur propre maximale de la matrice $R_{p-1}(t)$.

Critère optimisé:

Le fait d'avoir approché $R_{p-1}(t)$ par δI_{p-1} a conduit à optimiser un critère différent de celui de l'erreur quadratique moyenne pondérée.

En considérant la solution obtenue:

$$A_{p-1}(t+1) = A_{p-1}(t) + \delta X_{p-1}(t+1)(x_p(t+1) - X_{p-1}^T(t+1)A_{p-1}(t))^T$$

il vient:

$$A_{p-1}(t+1) = (I_{p-1} - \delta X_{p-1}(t+1)X_{p-1}^T(t+1))A_{p-1}(t) + \delta X_{p-1}(t+1)x_p(t+1)$$

en se plaçant dans le cas où l'algorithme a convergé, il vient pour $t \rightarrow \infty$:

$$E[A_{p-1}(t+1)] = E[A_{p-1}(t)]$$

or : $E[X_{p-1}(t+1)X_{p-1}^T(t+1)] = R_{p-1}(t+1)$ et $E[X_{p-1}(t+1)x_p(t+1)] = S_{p-1}(t+1)$

d'où en prenant l'espérance de l'équation précédente :

$$E[A_{p-1}(t+1)] = E[A_{p-1}(t)] = E[A_{p-1}(t)] - \delta (R_{p-1}(t+1)E[A_{p-1}(t)] - S_{p-1}(t+1))$$

d'où:

$$\text{pour } t \rightarrow \infty \quad E[A_{p-1}(t)] = R_{p-1}^{-1}(t+1)S_{p-1}(t+1)$$

La solution obtenue grâce à l'algorithme du gradient stochastique converge donc *en moyenne* vers la solution optimale. L'algorithme n'optimise donc pas un critère des moindres carrés exacts mais un critère des moindres carrés moyens.

Variantes de l'algorithme

L'algorithme du gradient stochastique possède un grand nombre de variantes dont l'emploi dépend essentiellement de l'application envisagée. On peut citer par exemple:

L'algorithme de gradient normalisé (NLMS):

$$A_{p-1}(t+1) = A_{p-1}(t) + \frac{\delta X_{p-1}(t+1)e_p(t+1)}{X_{p-1}^T(t+1)X_{p-1}(t+1)}$$

L'algorithme du gradient basé sur le signe de l'erreur (error sign):

$$A_{p-1}(t+1) = A_{p-1}(t) + \delta X_{p-1}(t+1) \text{sgn}(e_p^T(t+1))$$

L'algorithme du gradient basé sur le signe de la donnée (data sign):

$$A_{p-1}(t+1) = A_{p-1}(t) + \delta \text{sgn}(X_{p-1}(t+1))e_p^T(t+1)$$

L'algorithme du signe (sign algorithm):

$$A_{p-1}(t+1) = A_{p-1}(t) + \delta \text{sgn}(X_{p-1}(t+1)e_p^T(t+1))$$

Conclusion

L'algorithme du gradient stochastique est un algorithme simple dont le coût de calcul est proportionnel à l'ordre N du filtre à identifier. A condition de respecter un pas d'adaptation suffisamment faible, cet algorithme est stable et optimise un critère des moindres carrés moyens. C'est encore aujourd'hui l'algorithme de filtrage adaptatif le plus employé dans les applications temps réel.

Le problème principal dans son emploi réside dans le choix du pas d'adaptation δ . Un pas faible entraîne une convergence lente souvent incompatible des applications envisagées (égalisation en évocation de fréquence par exemple). Un pas trop fort va conduire, quant à lui, à des résultats imprécis. Il existe un grand nombre de méthodes qui tendent à faire varier ce pas δ au cours de

la phase de convergence. Cela peut avoir pour effet d'accélérer la convergence mais il faut rester circonspect vis à vis de ces méthodes car la convergence de l'algorithme n'est assurée que dans le cas du pas fixe.

6.5 Algorithme RLS

L'algorithme du gradient stochastique partait de l'approximation: $R_{p-1}(t) = \delta I_{p-1}$. Cette simplification a permis de passer d'un coût de calcul proportionnel à $O(p^3)$ à un coût de calcul proportionnel à p . En évitant cette simplification, il est cependant possible de résoudre l'équation:

$$A_{p-1}(t+1) = R_{p-1}^{-1}(t+1)S_{p-1}(t+1)$$

avec un coût de calcul proportionnel à p^2 .

En effet: $R_{p-1}(t+1) = \lambda R_{p-1}(t) + X_{p-1}(t+1)X_{p-1}^T(t+1)$

En utilisant un lemme d'inversion matricielle rappelé ci-dessous :

si: $A = B + CDC^T$

alors: $A^{-1} = B^{-1} - B^{-1}C[D + C^T B^{-1}C]^{-1}C^T B^{-1}$

il vient :

$$R_{p-1}^{-1}(t+1) = \frac{1}{\lambda} R_{p-1}^{-1}(t) - \frac{1}{\lambda} R_{p-1}^{-1}(t) X_{p-1}(t+1) \left[1 + X_{p-1}^T(t+1) \frac{R_{p-1}^{-1}(t)}{\lambda} X_{p-1}(t+1) \right]^{-1} X_{p-1}^T(t+1) \frac{R_{p-1}^{-1}(t)}{\lambda}$$

d'où:

$$R_{p-1}^{-1}(t+1) = \frac{1}{\lambda} \left(R_{p-1}^{-1}(t) - \frac{R_{p-1}^{-1}(t) X_{p-1}(t+1) X_{p-1}^T(t+1) R_{p-1}^{-1}(t)}{\lambda + X_{p-1}^T(t+1) R_{p-1}^{-1}(t) X_{p-1}(t+1)} \right)$$

En multipliant à gauche par $X_{p-1}(t+1)$ et en factorisant, il vient:

$$G_{p-1}(t+1) = R_{p-1}^{-1}(t+1) X_{p-1}(t+1) = \frac{1}{\lambda} R_{p-1}^{-1}(t) X_{p-1}(t+1) \left(1 - \frac{X_{p-1}^T(t+1) R_{p-1}^{-1}(t) X_{p-1}(t+1)}{\lambda + X_{p-1}^T(t+1) R_{p-1}^{-1}(t) X_{p-1}(t+1)} \right)$$

d'où:

$$G_{p-1}(t+1) = \frac{R_{p-1}^{-1}(t)X_{p-1}(t+1)}{\lambda + X_{p-1}^T(t+1)R_{p-1}^{-1}(t)X_{p-1}(t+1)}$$

d'où l'algorithme suivant :

Quantités obtenues au temps t et disponibles au temps $t+1$:

inverse de la matrice d'autocorrélation : $R_{p-1}^{-1}(t)$

vecteur des coefficients du filtre : $A_{p-1}(t)$

Nouvelles données au temps $t+1$:

vecteur de données reçues : $X_{p-1}(t+1)$

signal de référence : $x_p(t+1)$

Calcul du nouveau filtre au temps $n+1$:

erreur de prédiction avant : $e_p(t+1) = x_p(t+1) - A_{p-1}^T(t)X_{p-1}(t+1)$

calcul du gain de Kalman : $G_{p-1}(t+1) = \frac{R_{p-1}^{-1}(t)X_{p-1}(t+1)}{\lambda + X_{p-1}^T(t+1)R_{p-1}^{-1}(t)X_{p-1}(t+1)}$

mise à jour du filtre au temps $t+1$: $A_{p-1}(t+1) = A_{p-1}(t) + G_{p-1}(t+1)e_p(t+1)$

propagation de l'inverse de :

$$R_{p-1}(t+1) : \quad R_{p-1}^{-1}(t+1) = \frac{1}{\lambda} \left(R_{p-1}^{-1}(t) - G_{p-1}(t+1)X_{p-1}^T(t+1)R_{p-1}^{-1}(t) \right)$$

- Algorithme RLS -

Cet algorithme conduit à la mise à jour du vecteur $A_{p-1}(t+1)$ avec un coût de calcul proportionnel à $O(p^2)$. L'inversion de matrice a été évitée en propageant le gain de Kalman $G_{p-1}(t+1)$ récursivement au cours du temps.

Pour $t = 0$, la matrice $R_{p-1}^{-1}(0)$ est choisie égale à $\frac{1}{\delta^2} I_{p-1}$ où δ représente un scalaire petit devant 1.

Cet algorithme, à la différence de l'algorithme du gradient stochastique, optimise rigoureusement le critère des moindres carrés.

L'algorithme RLS s'applique aussi bien au cas d'un signal *multidimensionnel* :

$$X_{p-1}(t) = [x_1(t), x_2(t), \dots, x_{p-1}(t)]^T$$

qu'au cas d'un signal *monodimensionnel* dont les valeurs sont conservés avec un décalage temporel :

$$X_{p-1}(t) = [x(t), x(t-1), \dots, x(t-N+1)]^T$$

Cependant dans ce dernier cas, l'algorithme n'utilise pas cette propriété de décalage des variables. En utilisant cette propriété, il est possible d'optimiser l'algorithme et d'obtenir une version rapide (coût de calcul proportionnel à N).

6.6 Exercices corrigés

Exercice 1

On considère le code Matlab de prédiction linéaire suivant. Ce code utilise l'algorithme du gradient stochastique. Le pas est noté mu, le signal utile est noté x(k), le signal est réel.

.../....

```
For k=1:length(x)

    X(2:N)=X(1:N-1);

    X(1)=x(k);

    e=x(k)-A'*X;

    A=X+mu*e*A;

end
```

1. Des erreurs se sont glissées dans ce code, indiquez les et corrigez les.

Corrigé

```
For k=1:length(x)

    X(2:N)=X(1:N-1);

    X(1)=x(k);

    e=x(k+1)-A'*X;

    A=A+mu*e*X;

end
```

Exercice 2

On considère que l'on reçoit trois signaux réels $x_1(t)$, $x_2(t)$, $x_3(t)$ définis de la manière suivante

$$\begin{aligned}x_1(t) &= s(t) + b_1(t) \\x_2(t) &= 2b_2(t) \\x_3(t) &= 2s(t) + 3b_3(t)\end{aligned}$$

Tous les signaux $s(t)$ et $b_i(t)$ sont réels blancs et indépendants. Le signal $s(t)$ représente le signal utile et les signaux $b_i(t)$ des termes de bruit additifs.

On a : $E[s^2(t)] = 1$ et $\forall i E[b_i^2(t)] = \sigma^2$

1. Calculez les rapports signaux sur bruit Γ_i sur chaque signal $x_i(t)$
2. Déterminez les 3 coefficients a_1, a_2, a_3 qui permettent de prédire au mieux, au sens des moindres carrés, le signal $s(t)$ à partir des signaux $x_1(t), x_2(t), x_3(t)$. C'est-à-dire qui minimisent $E[(e(t))^2]$ avec : $e(t) = s(t) - a_1x_1(t) - a_2x_2(t) - a_3x_3(t)$

Corrigé

Question 1

$$\Gamma_1 = \frac{1}{\sigma^2}, \Gamma_2 = 0, \Gamma_3 = \frac{4}{9\sigma^2}$$

Question 2

Le signal $x_2(t)$ n'apporte aucune information sur le signal utile, ce n'est donc qu'une source de bruit et il est évident que le coefficient $a_2 = 0$. La question se résume donc à :

$$e(t) = s(t) - a_1x_1(t) - a_3x_3(t)$$

La solution de Wiener est donc donnée par :

$$A = \begin{pmatrix} a_1 \\ a_3 \end{pmatrix} = \begin{pmatrix} 1 + \sigma^2 & 2 \\ 2 & 4 + 9\sigma^2 \end{pmatrix}^{-1} \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

$$A = \frac{1}{(1 + \sigma^2)(4 + 9\sigma^2) - 4} \begin{pmatrix} 4 + 9\sigma^2 & -2 \\ -2 & 1 + \sigma^2 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

$$A = \frac{1}{(1 + \sigma^2)(4 + 9\sigma^2) - 4} \begin{pmatrix} 9\sigma^2 \\ 2\sigma^2 \end{pmatrix}$$

7. Bibliographie

M. Bellanger, Traitement numérique du signal : Théorie et pratique, 9^{ème} édition, Dunod, Paris 2010.

D. Ghorbanzadeh, P. Marry, N. Point, D. Vial, Eléments de mathématiques du signal, 3^{ème} édition, Dunod, Paris 2011

A. Quinquis, Le traitement du signal sous Matlab, Hermès Lavoisier, Paris 2007

Murat Kunt, Traitement de l'information, vol 1, Techniques modernes de traitement numérique des signaux, PPUR, Lausanne 1991

J.G. Proakis, D.K. Manolakis, Digital Signal Processing, 4th ed, Prentice Hall, New York 2006.

S. Kay, Statistical Signal processing, Detection theory,prentice Hall, New York, 1993

S. Kay, S. Marple, Spectrum analysis - A modern perspective -, Proceedings of IEEE vol 69 N° 11, November 1981.

F. Cottet, Aide mémoire de traitement du signal, Dunod, 2011

8. Annexes

8.1 Annexe Dérivation Forme Bilineaire

Dérivation de $A^T RA$ en réel

$$u = (a_0 \quad \dots \quad a_{N-1}) \begin{pmatrix} r_{i,j} \\ \vdots \\ a_{N-1} \end{pmatrix}$$

$$u = (a_0 \quad \dots \quad a_{N-1}) \begin{pmatrix} \sum_j r_{k,j} a_j \end{pmatrix}$$

$$u = \sum_k a_k \left(\sum_j r_{k,j} a_j \right)$$

$$u = \sum_k \sum_j a_k a_j r_{k,j}$$

$$u = \sum_{k \neq p} \sum_j a_k a_j r_{k,j} + \sum_j a_p a_j r_{p,j}$$

$$u = \sum_{k \neq p} a_k a_p r_{k,p} + \sum_{k \neq p} \sum_{j \neq p} a_k a_j r_{k,j} + \sum_{j \neq p} a_p a_j r_{p,j} + a_p^2 r_{p,p}$$

$$\frac{\partial u}{\partial p} = \sum_{k \neq p} a_k r_{k,p} + \sum_{j \neq p} a_j r_{p,j} + 2a_p r_{p,p}$$

$R = R^T$ donc $r_{k,p} = r_{p,k}$ d'où :

$$\frac{\partial u}{\partial a_p} = 2a_p r_{p,p} + 2 \sum_{j \neq p} a_j r_{p,j}$$

$$\begin{pmatrix} \vdots \\ \frac{\partial u}{\partial p} \\ \vdots \end{pmatrix} = 2 \begin{pmatrix} \dots & r_{p,j} & \dots \end{pmatrix} \begin{pmatrix} \vdots \\ a_j \\ \vdots \end{pmatrix}$$

Donc :

$$\frac{\partial}{\partial A} A^T RA = 2RA$$

8.2 Annexe, Méthode de Capon

Pour chaque fréquence f , cette méthode cherche un *filtre adapté* dont la réponse vaut 1 pour cette fréquence f et 0 partout ailleurs. Une fois ce filtre obtenu l'estimateur $P_{Capon}(f)$ n'est autre que la puissance de sortie du filtre obtenu pour la fréquence f .

La sortie $y(n)$ de ce filtre s'écrit :

$$y(n) = \sum_{i=0}^N h_i^* x(n-i) \quad (104.)$$

ou plus simplement sous forme vectorielle :

$$y(n) = H^T X(n) \quad (105.)$$

$$\text{avec } H = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_N \end{bmatrix} \text{ et } X(n) = \begin{bmatrix} x(n) \\ x(n-1) \\ \vdots \\ x(n-N+1) \end{bmatrix}$$

Le filtre H doit donc minimiser $E[y(n)^2]$ avec la contrainte $H^T F = 1$

avec :

$$F = \begin{bmatrix} 1 \\ e^{j2\pi f} \\ e^{j4\pi f} \\ \vdots \\ e^{j2N\pi f} \end{bmatrix} \quad (106.)$$

Ce qui s'écrit, en utilisant un multiplicateur de Lagrange:

H optimal est tel que $E[H^T X(n)]^2 + \alpha(I - H^T F)$ est minimal

soit, en annulant la dérivé par rapport à H :

$$H = \frac{\alpha}{2} R_N^{-1} F \quad (107.)$$

avec $R_N = E[X(n)X(n)^T]$ matrice d'autocorrélation du signal $x(n)$.

en introduisant α dans l'expression de la contrainte $H^T F = 1$, il vient :

$$\alpha = \frac{2}{F^T R_N^{-1} F} \quad (108.)$$

d'où finalement :

$$H = \frac{R_N^{-1} F}{F^T R_N^{-1} F} \quad (109.)$$

Pour estimer $P_{Cap}(f)$ il ne reste plus qu'à calculer:

$$P_{Cap}(f) = E[y(n)^2] = E[H^T X(n) X(n)^T H] \quad (110.)$$

$$P_{Cap}(f) = \frac{F^T R_N^{-1} R_N R_N^{-1} F}{F^T R_N^{-1} F F^T R_N^{-1} F} \quad (111.)$$

$$P_{Cap}(f) = \frac{1}{F^T R_N^{-1} F} \quad (112.)$$

or :

$$F^T R_N^{-1} F = [1, e^{-j2\pi f}, e^{-j4\pi f}, \dots, e^{-j2N\pi f}] \begin{bmatrix} \rho_{kl} \end{bmatrix} \begin{bmatrix} 1 \\ e^{j2\pi f} \\ e^{j2N\pi f} \end{bmatrix} \quad (113.)$$

d'où:

$$P_{Cap}(f) = \frac{1}{\sum_{k=0}^N \sum_{l=0}^N \rho_{kl} e^{-j2\pi(k-l)f}} \quad (114.)$$

En conclusion pour obtenir l'estimateur de Capon, il faut:

- observer N échantillons du signal $x(nT_e)$
- estimer la matrice d'autocorrélation et l'inverser pour obtenir les termes ρ_{kl}
- calculer $P_{Cap}(f)$ pour chaque fréquence f .

Cette méthode souffre donc d'un coût de calcul supérieur aux estimateurs précédents. Il est possible de démontrer que cet estimateur à une variance minimale et c'est la raison pour laquelle il est souvent appelé estimateur du minimum de variance.

8.3 Annexe, Méthode de Pisarenko:

Dans cette méthode, le signal $x(n)$ est supposé être constitué d'une somme de N sinusoïdes $s(n)$ et d'un bruit blanc additif $b(n)$.

$$x(n) = s(n) + b(n) \quad (115.)$$

Sachant que tout signal sinusoïdal réel $\sin(n\omega)$ peut s'écrire:

$$\sin(n\omega) = 2 \cos \omega \sin(n-1)\omega - \sin(n-2)\omega$$

il est possible d'écrire:

$$s(n) = - \sum_{m=1}^{2N} a_m s(n-m)$$

d'où:

$$x(n) = - \sum_{m=1}^{2N} a_m s(n-m) + b(n)$$

en remplaçant $s(n-m)$ par $x(n-m) - b(n-m)$, il vient:

$$\sum_{m=0}^{2N} a_m x(n-m) = \sum_{m=0}^{2N} a_m b(n-m)$$

ce qui peut s'écrire matriciellement:

$$X(n)^T A = B(n)^T A$$

avec:

$$X(n)^T = [x(n), x(n-1), \dots, x(n-2N)]^T$$

$$B(n)^T = [b(n), b(n-1), \dots, b(n-2N)]^T$$

et :

$$A = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{2N} \end{bmatrix}$$

d'où:

$$E[X(n)X(n)^T]A = E[X(n)B(n)^T]A = E[(S(n) + B(n))B(n)^T]A$$

$$R_{2N+1}A = E[(S(n) + B(n))B(n)^T]A$$

Or, le bruit est supposé blanc, de variance σ^2 et décorrélé du mélange de sinusoïdes, cette équation devient donc:

$$R_{2N+1}A = \sigma^2 IA = \sigma^2 A$$

Le vecteur A est donc le vecteur propre associé à la valeur propre σ^2 avec la contrainte $a_0 = 1$.

Ayant le vecteur A , on peut écrire la transformée en Z de l'équation:

$$s(n) = -\sum_{m=1}^{2N} a_m s(n-m)$$

$$S(Z) \left(1 - \sum_{m=1}^{2N} a_m Z^{-m} \right) = 0$$

les valeurs de Z pour lesquelles cette équation est vérifiée donnent les valeurs des fréquences présentes dans le mélange. Il faut donc extraire les racines du polynôme $1 - \sum_{m=1}^{2N} a_m Z^{-m}$ pour obtenir les N valeurs de fréquences.

Si N et σ^2 ne sont pas connus a priori, il suffit de surdimensionner la matrice R et d'analyser ses valeurs propres. Théoriquement celles-ci doivent, au bout de N ordres, atteindre une valeur constante égale à σ^2 . Il existe de nombreux critères qui permettent de détecter ce blocage à σ^2 des valeurs propres.

Enfin, il est possible d'obtenir les puissances des sinusoïdes détectées. En effet, il est facile de vérifier que, dans le cas d'un mélange de N sinusoïdes de pulsation ω_i et de puissance P_i , on a :

$$r(1) = P_1 \cos \omega_1 + P_2 \cos \omega_2 + \dots + P_N \cos \omega_N$$

$$r(2) = P_1 \cos 2\omega_1 + P_2 \cos 2\omega_2 + \dots + P_N \cos 2\omega_N$$

$$\dots$$

$$r(N) = P_1 \cos N\omega_1 + P_2 \cos N\omega_2 + \dots + P_N \cos N\omega_N$$

disposant des pulsations ω_i et des coefficients d'autocorrélation $r(i)$, il "suffit" de résoudre ce système pour trouver les puissances respectives P_i des raies spectrales identifiées.

En conclusion pour analyser un signal selon la méthode de Pisarenko, il faut:

- observer N valeurs du signal $x(nT_e)$,

- calculer la matrice d'autocorrélation et en faire la décomposition en éléments propres,
- détecter σ^2 et en déduire le nombre de sinusoïdes,
- extraire les racines d'un polynôme complexe de degré $2N$,
- enfin si l'on veut les puissances, résoudre un système réel de N équations à N inconnues.

La décomposition en éléments propres reste pour l'instant l'étape la plus délicate à réaliser de manière rapide et c'est le frein principal à l'emploi de cette méthode.

8.4 Annexe, Méthode de Prony:

Dans cette méthode le signal est, comme dans la méthode de Pisarenko, supposé être constitué d'une somme de N sinusoides mais le bruit est remplacé par un amortissement sur ces dernières. L'hypothèse de départ s'écrit donc :

$$x(n) = \sum_{m=1}^N b_m Z_m^n$$

avec :

$$Z_m = e^{\alpha_m} e^{j2\pi f_m}$$

On peut alors fabriquer le polynôme :

$$\psi(Z) = \prod_{k=1}^N (Z - Z_k) = \sum_{i=0}^N a_i Z^{N-i} \quad \text{avec } a_0 = 1$$

D'après l'hypothèse de départ sur $x(n)$ on a :

$$x(n-m) = \sum_{l=1}^N b_l Z_l^{n-m}$$

en multipliant cette équation par a_m et en effectuant une sommation sur m , il vient :

$$\sum_{m=0}^N a_m x(n-m) = \sum_{m=0}^N a_m \sum_{l=1}^N b_l Z_l^{n-m}$$

ou encore :

$$\sum_{m=0}^N a_m x(n-m) = \sum_{l=1}^N b_l Z_l^{n-N} \sum_{m=0}^N a_m Z_l^{N-m}$$

or $\sum_{m=0}^N a_m Z_l^{N-m} = 0$ par définition des coefficients a_m du polynôme $\psi(Z)$ dont les racines sont les Z_l , d'où :

$$\sum_{m=0}^N a_m x(n-m) = 0$$

et donc :

$$x(n) = - \sum_{m=1}^N a_m x(n-m)$$

Les coefficients a_m peuvent donc être obtenus par la résolution du système linéaire de dimension N suivant :

$$\begin{bmatrix} x(N+1) \\ \cdot \\ \cdot \\ x(2N) \end{bmatrix} = \begin{bmatrix} x(N) & \cdot & \cdot & x(1) \\ x(N+1) & \cdot & \cdot & x(2) \\ \cdot & \cdot & \cdot & \cdot \\ x(2N) & \cdot & \cdot & x(N) \end{bmatrix} \begin{bmatrix} a_1 \\ \cdot \\ \cdot \\ a_N \end{bmatrix}$$

A partir des N coefficients a_m il est possible de former le polynôme :

$$\psi(Z) = \sum_{m=0}^N a_m Z^{m-i} \text{ avec } a_0 = 1$$

et d'en extraire les N racines complexes. Les modules de ces racines donnent alors les affaiblissements α_m tandis que les phases donnent les fréquences f_m .

Les amplitudes respectives b_m des différentes sinusoïdes peuvent enfin être obtenues en résolvant le système linéaire suivant :

$$\begin{bmatrix} 1 & 1 & \cdot & 1 \\ Z_1 & Z_2 & \cdot & Z_N \\ \cdot & \cdot & \cdot & \cdot \\ Z_1^{N-1} & \cdot & \cdot & Z_N^{N-1} \end{bmatrix} \begin{bmatrix} b_1 \\ \cdot \\ \cdot \\ b_N \end{bmatrix} = \begin{bmatrix} x_0 \\ \cdot \\ \cdot \\ x_{N-1} \end{bmatrix}$$

Pour cette méthode il faut donc :

- observer $2N$ valeurs du signal $x(nT_e)$
- résoudre un système linéaire complexe de dimension N
- extraire les racines d'un polynôme complexe de degré N
- et si l'on veut les amplitudes résoudre un système linéaire complexe de dimension N .

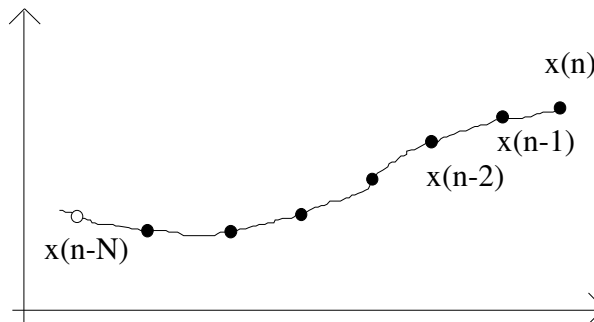
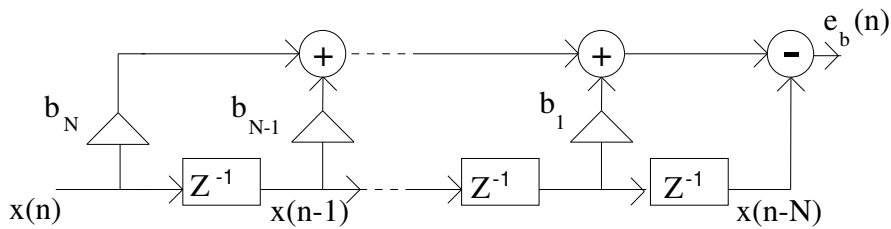
8.5 Annexe, Algorithme de Levinson :

Pour faire passer le coût de calcul de la résolution de l'équation de Yule Walker de $o(N^2)$ à $o(N^3)$, cet algorithme va utiliser une récurrence sur l'ordre N du modèle prédicteur AR. Pour cela, il est nécessaire d'introduire l'erreur de prédiction arrière.

Le modèle de prédiction avant consistait à estimer $x(n)$ à partir de $\{x(n-1) \ x(n-2) \ \dots \ x(n-N)\}$, en inversant l'axe des temps, on peut construire un modèle de prédiction arrière qui va estimer $x(n-N)$ à partir de $\{x(n-N+1) \ \dots \ x(n)\}$. Cela correspond à l'équation de prédiction arrière suivante :

$$x(n-N) = \sum_{i=1}^N b_{N+1-i} x(n+1-i) + e_b(n)$$

modèle, dont la structure RIF est la suivante:



L'équation de prédiction arrière s'écrit de manière matricielle:

$$e_b(n) = x(n-N) - X_N^T(n)B_N$$

avec $X_N(n) = [x(n), x(n-1), \dots, x(n-N+1)]^T$

et $B_N = [b_N, b_{N-1}, \dots, b_1]^T$

Le vecteur B_N optimal est tel qu'il minimise $E_{bN} = E[e_b(n)^2]$

ce qui conduit à: $B_N = E[X_N(n)X_N^T(n)]^{-1} E[X_N(n)x(n-N)]$

d'où:

$$B_N = R_N^{-1} r_N^b \quad (4)$$

avec $r_N^b = [r(N), r(N-1), \dots, r(1)]$

en reportant B_N dans l'expression de $E_{bN} = E[e_b(n)^2]$ il vient:

$$E[e_b(n)^2] = E[x(n-N)^2] - 2E[x(n-N)X_N(n)]B_N + B_N^T E[X_N(n)X_N^T(n)]B_N$$

d'où:

$$E_{bN} = r(0) - r_N^{bT} B_N \quad (5)$$

En réunissant (4) et (5) au sein d'une même équation matricielle, il vient:

$$\begin{bmatrix} R_N & r_N^b \\ r_N^{bT} & r(0) \end{bmatrix} \begin{bmatrix} -B_N \\ I \end{bmatrix} = \begin{bmatrix} 0 \\ E_{bN} \end{bmatrix} \quad (6)$$

Equation de Yule Walker arrière (ou rétrograde ou backward) dans le cas stationnaire

L'algorithme de Levinson est obtenu en réunissant les équations avant et arrière.

- Il est aisé de vérifier $E_{aN} = E_{bN}$

En effet, l'équation (6) peut s'écrire:

$$R_{N+1} \begin{bmatrix} -B_N \\ I \end{bmatrix} = \begin{bmatrix} 0 \\ E_{bN} \end{bmatrix}$$

en multipliant à gauche par la coidentité :

$$J_N = \begin{bmatrix} 0 & 0 & I \\ 0 & I & 0 \\ I & 0 & 0 \end{bmatrix}$$

et en remarquant que, du fait de la symétrie de R_N :

$$J_N R_N = R_N J_N$$

il vient:

$$R_{N+1} \begin{bmatrix} I \\ -JB_N \end{bmatrix} = \begin{bmatrix} E_{bN} \\ 0 \end{bmatrix}$$

en identifiant avec l'équation avant (3):

$$R_{N+1} \begin{bmatrix} I \\ -A_N \end{bmatrix} = \begin{bmatrix} E_{aN} \\ 0 \end{bmatrix}$$

il vient: $A_N = JB_N$ et $E_{bN} = E_{aN} = E_N$

Ce qui revient à dire que prédire $x(n-N)$ à partir de $x(n-N+1), \dots, x(n)$ est identique à prédire $x(n)$ à partir de $x(n-1), \dots, x(n-N)$. La seule différence entre les deux prédictions est le sens de parcours sur la trajectoire de $x(n)$.

- L'algorithme se base sur une récursion sur l'ordre N . Si on connaît A_{N-1} et B_{N-1} à l'ordre $N-1$, il vient pour la prédiction avant à l'ordre N :

$$\begin{bmatrix} R_N & r_N^b \\ r_N^{bT} & r(0) \end{bmatrix} \begin{bmatrix} I \\ -A_{N-1} \\ 0 \end{bmatrix} = \begin{bmatrix} E_{N-1} \\ 0 \\ K_N \end{bmatrix}$$

avec $K_N = r(N) - \sum_{i=1}^{N-1} a_{i,N-1} r(N-i)$

et pour la prédiction arrière:

$$\begin{bmatrix} r(0) & r_N^{aT} \\ r_N^a & R_N \end{bmatrix} \begin{bmatrix} 0 \\ -B_{N-1} \\ 1 \end{bmatrix} = \begin{bmatrix} K_N \\ 0 \\ E_{N-1} \end{bmatrix}$$

d'où:

$$R_{N+1} \begin{bmatrix} 1 \\ -A_{N-1} \\ 0 \end{bmatrix} = \begin{bmatrix} E_{N-1} \\ 0 \\ K_N \end{bmatrix} \quad \text{et} \quad R_{N+1} \begin{bmatrix} 0 \\ -B_{N-1} \\ 1 \end{bmatrix} = \begin{bmatrix} K_N \\ 0 \\ E_{N-1} \end{bmatrix}$$

$$\text{d'où: } R_{N+1} \left[\begin{bmatrix} 1 \\ -A_{N-1} \\ 0 \end{bmatrix} - \frac{K_N}{E_{N-1}} \begin{bmatrix} 0 \\ -B_{N-1} \\ 1 \end{bmatrix} \right] = \begin{bmatrix} E_{N-1} - \frac{K_N^2}{E_{N-1}} \\ 0 \\ 0 \end{bmatrix}$$

d'où:

$$[A_N] = \begin{bmatrix} A_{N-1} \\ 0 \end{bmatrix} - k_N \begin{bmatrix} B_{N-1} \\ -1 \end{bmatrix}$$

et $E_N = E_{N-1} (1 - k_N^2)$ avec $k_N = \frac{K_N}{E_{N-1}}$

- Algorithme de Levinson -

Ce qui correspond aux relations suivantes:

$$E(0) = r(0)$$

$$j: 1 \rightarrow N \begin{cases} k_j = \frac{1}{E_{j-1}} \left[r(j) - \sum_{i=1}^{j-1} a_{i,j-1} r(j-i) \right] \\ a_{j,j} = k_j \\ i: 1 \rightarrow j-1 \left\{ \begin{aligned} a_{i,j} &= a_{i,j-1} - k_j a_{j-i,j-1} \\ E_j &= E_{j-1} (1 - k_j^2) \end{aligned} \right. \end{cases}$$

Remarque:

il est possible d'écrire: $\begin{pmatrix} 1 \\ -A_N \end{pmatrix} = \begin{pmatrix} 1 \\ -A_{N-1} \\ 0 \end{pmatrix} - k_N \begin{pmatrix} 0 \\ -B_{N-1} \\ 1 \end{pmatrix}$

$$\text{et : } \begin{pmatrix} -B_N \\ I \end{pmatrix} = \begin{pmatrix} 0 \\ -B_{N-1} \\ I \end{pmatrix} - k_N \begin{pmatrix} I \\ -A_{N-1} \\ 0 \end{pmatrix}$$

d'où:

$$A_N = \begin{pmatrix} A_{N-1} \\ 0 \end{pmatrix} - k_N \begin{pmatrix} B_{N-1} \\ -I \end{pmatrix}$$

et:

$$B_N = \begin{pmatrix} 0 \\ B_{N-1} \end{pmatrix} - k_N \begin{pmatrix} -I \\ A_{N-1} \end{pmatrix}$$

d'où en reprenant les expressions des erreurs avants et arrières:

$$e_{aN}(n) = x(n) - X_N^T(n-1)A_N$$

$$e_{aN}(n) = x(n) - X_N^T(n-1) \left[\begin{pmatrix} A_{N-1} \\ 0 \end{pmatrix} - k_N \begin{pmatrix} B_{N-1} \\ -I \end{pmatrix} \right]$$

$$e_{aN}(n) = e_{aN-1}(n) - k_N X_N^T(n-1) \begin{pmatrix} -B_{n-1} \\ I \end{pmatrix}$$

$$\text{or: } e_{bN}(n) = x(n-N) - X_N^T(n)B_N$$

$$e_{bN-1}(n) = x(n-N+1) - X_{N-1}^T(n)B_{N-1}$$

$$e_{bN-1}(n-1) = x(n-N) - X_{N-1}^T(n-1)B_{N-1}$$

$$\Rightarrow e_{aN}(n) = e_{aN-1}(n) - k_N e_{bN-1}(n-1)$$

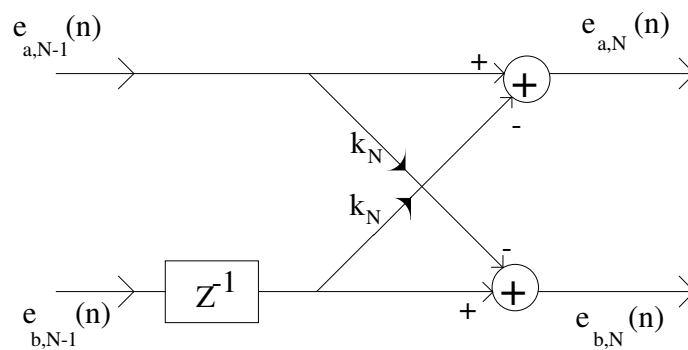
de la même manière :

$$e_{bN}(n) = x(n-N) - X_N^T(n)B_N$$

conduit à :

$$e_{bN}(n) = e_{bN-1}(n-1) - k_N e_{aN-1}(n)$$

Les coefficients k_i correspondent donc à la structure en treillis suivante:



Enfin, on peut démontrer:

$$k_N = \frac{E[e_{a,N-1}(n)e_{b,N-1}(n)]}{E_{N-1}}$$

d'où le nom de coefficients de corrélation partielle (PARCOR) donné souvent aux coefficients k_i .

En conclusion, la modélisation AR demande la mise à jour de N coefficients a_i pour cela il faut:

- observer au minimum N valeurs du signal,
- calculer les coefficients d'autocorrélation,
- appliquer l'algorithme de Levinson (coût de calcul proportionnel à $O(N^2)$).

8.6 Annexe, Neural Network and Backpropagation

The Neural network is composed of a set of layers, each of them composed by a linear filter and non-linear function. We have always an input layer, some hidden layers and an output layer.

In the neuron we will find a non-linear function playing the role of a threshold. As we cannot derivate an Heaviside function it is approached by a sigmoid function. A typical one is given hereafter:

$$f(x) = \frac{e^x}{1 + e^x}$$

And we have :

$$f'(x) = \frac{e^x}{(1 + e^x)^2}$$

$$f'(x) = f(x)(1 - f(x))$$

Let's consider a network with $q + 1$ layers numbered from 0 to q . With the following notations:

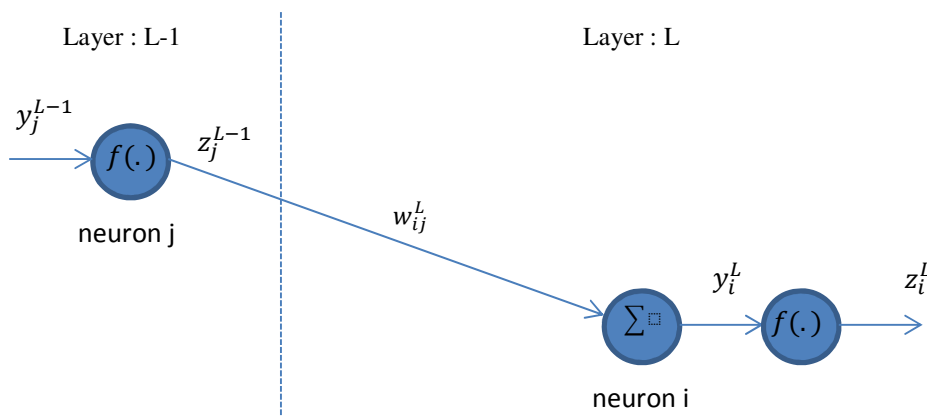
n_L : the number of neurons of the L^{th} layer.

y_i^L : the input of the i^{th} neuron of the L^{th} layer.

z_i^L : the output of the i^{th} neuron of the L^{th} layer.

w_{ij}^L : the coefficient between the j^{th} neuron of the $(L-1)^{\text{th}}$ layer and the i^{th} neuron of the L^{th} layer.

θ_i^L : the constant coefficient of the i^{th} neuron of the L^{th} layer.



We have then :

$$y_i^L = \sum_j w_{ij}^L z_j^{L-1} + \theta_i^L$$

And :

$$z_i^L = f(y_i^L)$$

The network's error, observed at the output is given by:

$$E(s) = \frac{1}{2} \sum_j (z_j^q - t_j^q)^2$$

The first input layer being fed with $s = (y_1^0, y_2^0, \dots, y_n^0)$.

The free parameters are all "synaptic coefficients": w_{ij}^L and the constant parameters θ_i^L , for all layers $L \in [1, q]$.

The main objective is to apply a gradient descent in order to find all w_{ij}^L and θ_i^L , terms. But for that purpose will have then to calculate the derivative of the error with respect to the synaptic coefficients and the constant parameters. But we will process by stages and we will first calculate $\frac{\partial E}{\partial y_i^L}$.

For the output layer (with the sigmoïde function) $L = q$, we have :

$$\frac{\partial E}{\partial y_j^q} = \frac{\partial E}{\partial z_j^q} \frac{\partial z_j^q}{\partial y_j^q} = (z_j^q - t_j^q) z_j^q (1 - z_j^q)$$

For the output layer (with the linear function) $L = q$, we have :

$$\frac{\partial E}{\partial y_j^q} = \frac{\partial E}{\partial z_j^q} \frac{\partial z_j^q}{\partial y_j^q} = (z_j^q - t_j^q)$$

This partial derivation is easy to calculate due to the fact that we are dealing with the output layer. It is more complex for inner layer and we will establish a recursive backward solution.

For an inner layer $< q$, we have :

$$\frac{\partial E}{\partial y_j^L} = \frac{\partial E}{\partial z_j^L} \frac{\partial z_j^L}{\partial y_j^L} = \frac{\partial E}{\partial z_j^L} z_j^L (1 - z_j^L)$$

We can then develop $\frac{\partial E}{\partial z_j^L}$ as follows :

$$\frac{\partial E}{\partial z_j^L} = \sum_k \frac{\partial E}{\partial y_k^{L+1}} \frac{\partial y_k^{L+1}}{\partial z_j^L} = \sum_k \frac{\partial E}{\partial y_k^{L+1}} w_{kj}^{L+1}$$

Merging the two last equations we have then:

$$\frac{\partial E}{\partial y_j^L} = z_j^L (1 - z_j^L) \sum_k \frac{\partial E}{\partial y_k^{L+1}} w_{kj}^{L+1}$$

Finally this equation gives us a recursive way to calculate all $\frac{\partial E}{\partial y_j^L}$ terms from $\frac{\partial E}{\partial y_j^q}$ easily obtained

The gradient descent

As mentioned earlier our goal is to propose a gradient descent algorithm given by the two following equations:

$$w_{ij}^L = w_{ij}^L - \delta \frac{\partial E}{\partial w_{ij}^L}$$

$$\theta_i^L = \theta_i^L - \delta \frac{\partial E}{\partial \theta_i^L}$$

Once more we will use partial derivations:

$$\frac{\partial E}{\partial w_{ij}^L} = \frac{\partial E}{\partial y_i^L} \frac{\partial y_i^L}{\partial w_{ij}^L} = \frac{\partial E}{\partial y_i^L} z_j^{L-1}$$

$$\frac{\partial E}{\partial \theta_i^L} = \frac{\partial E}{\partial y_i^L} \frac{\partial y_i^L}{\partial \theta_i^L} = \frac{\partial E}{\partial y_i^L}$$

The complete algorithm is given hereafter

Initialization with a sigmoïde function for the last layer

$$\frac{\partial E}{\partial y_j^q} = (z_j^q - t_j^q) z_j^q (1 - z_j^q)$$

Initialization with linear function for the last layer

$$\frac{\partial E}{\partial y_j^q} = (z_j^q - t_j^q)$$

Backpropagation, for $L=q-1$ to 1, for all j indexes of the concerned layer:

$$\frac{\partial E}{\partial y_j^L} = z_j^L (1 - z_j^L) \sum_k \frac{\partial E}{\partial y_k^{L+1}} w_{kj}^{L+1}$$

Adaptation, for $L=1$ to q , for all k and j indexes of the concerned layer:

$$w_{ij}^L = w_{ij}^L - \delta \frac{\partial E}{\partial y_i^L} z_j^{L-1}$$

$$\theta_i^L = \theta_i^L - \delta \frac{\partial E}{\partial y_i^L}$$

(for the case of the linear output function the adaptation equation for θ_i^L can potentially be cancelled)

Matlab implementation

You can either code by yourself the previous algorithm, or try to use the neural network matlab toolbox.

Here is a short example of how to use the train function

```
% How to calculate an Inverse Fourier transform with a neural network
clear all;
close all;
clc

K=32;
L=2;
N=1e2;

% Learning base

hc =randn(L,N)+j*randn(L,N); % random complex impulse response
H=zeros(K,N);
for n=1:N
    H(:,n)=fft(hc(:,n),K);
end
T = [real(hc);imag(hc)]; % (desired output)
X= [real(H);imag(H)];    % (input)

% We create the neural network : PMC de type feed-forward

egal_net = newff(minmax(X),[5*2*L 2*L],{'tansig' 'purelin'},'trainlm');

% quelques remarques :

% - egal_net : name of the neural network
% - minmax(X1) : range out the inputs
% - [10 2] : First layer 10 neurons, 2 neurons for the output layerd
% - tansig : non linear function for the first layer
% - purelin : linear function for the output
% (tansig et purelin correspondent à 10 et 2, respectivement)
% - trainlm : Levenberg-Marquardt algorithm used
```

```
egal_net.trainParam.epochs = 100; % max number of iterations

% Learning
[egal_net,tr_egal_net] = train(egal_net,X,T);

save egal_net egal_net; % enregistrement du réseau pour ne pas refaire à
chaque fois l'apprentissage

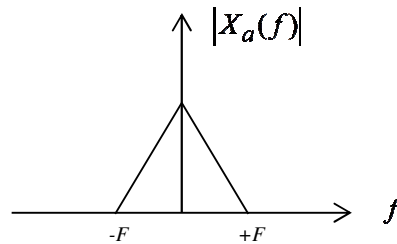
% Test
s1 = sim(egal_net,X);
hest = complex(s1(1,:),s1(2,:));
err = abs(hest-hc);
plot(err)
```

9. Exercices

9.1 Echantillonnage

EXERCICE N°1

Soit un signal analogique à temps continu $x_a(t)$ dont le spectre, représenté par le module de Transformée de Fourier (TF) $X_a(f)$ est à support borné dans l'intervalle $[-F, +F]$. On échantillonne $x_a(t)$ à la fréquence $F_e = \frac{1}{T_e}$ avec $F_e = 5F$. Cet échantillonnage, supposé idéal, s'écrit à l'aide d'un peigne de Dirac : $x_e(t) = x_a(t) \cdot W_{T_e}(t)$. On considère l'allure suivante pour la TF de $x_a(t)$:



1. Représentez graphiquement $|X_e(f)|$ pour $f \in [-20F, +20F]$

On échantillonne maintenant un signal à temps continu de la forme : $x_a(t) = A \cos(2\pi f_0 t) + B$. On obtient ainsi le signal échantillonné $x(nT_e) = x_a(nT_e)$.

2. Déterminez l'expression de $x(nT_e)$, exprimez la Transformée de Fourier de ce signal et représentez graphiquement son spectre

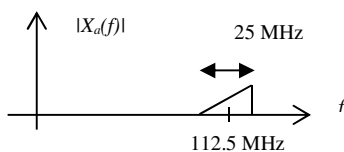
EXERCICE N°2

Le signal $x(t) = \cos(2\pi f_0 t)$ avec $f_0 = 6000 \text{ Hz}$ est échantillonné à $F_e = 10 \text{ kHz}$ avec un convertisseur 8 bits fonctionnant dans la plage $-2\text{V}, +2\text{V}$.

- 1) Calculez les échantillons (non quantifiés) d'une période du signal échantillonné $x(nT_e)$
- 2) Exprimer puis représenter le spectre du signal échantillonné $x(nT_e)$
- 3) Donner une estimation de la puissance du bruit de quantification
- 4) Quelle est la puissance du signal échantillonné $x(nT_e)$
- 5) Le signal numérique $x(nT_e)$ est fourni à un convertisseur numérique-analogique (on supposera le filtre de reconstruction idéal). Quel est la fréquence du signal analogique résultant ?

EXERCICE N°3

Soit un signal analogique réel à temps continu $x_a(t)$ dont le spectre $|X_a(f)|$ est à support borné, de largeur 25 MHz et centrée sur 112.5 MHz. La partie du spectre correspondant aux fréquences "positives" est représentée ci-dessous :



Un ingénieur propose d'échantillonner ce signal à une fréquence $F_e = 90\text{MHz}$. La plupart des ingénieurs auxquels il soumet cette proposition prétendent qu'il ne respecte pas le théorème de l'échantillonnage et ne voient pas l'intérêt de cette solution.

1. Présentez un argumentaire pour défendre la solution de cet ingénieur.
2. Représentez le spectre du signal échantillonné
3. La solution est-elle encore valable si le signal est centré sur 100 MHz et $F_e=50\text{ MHz}$, pouvez-vous exprimer une condition suffisante pour que le "sous échantillonnage" proposé fonctionne toujours convenablement.

EXERCICE N°4

Un ingénieur du son enregistre un concert avec 2 micros. Il supprime, par filtrage, les composantes des signaux analogiques aux fréquences supérieures à 22 kHz, puis il échantillonne les signaux de ses micros et quantifie les valeurs sur 16 bits. Il désire stocker les signaux numérisés sur un CD-ROM. On suppose qu'il n'effectue aucun autre traitement sur ses données (pas de codage contre les éventuelles erreurs par exemple).

Quelle doit être la capacité de son CD pour 70 minutes de concert ?

EXERCICE N°5

En pratique, l'échantillonnage d'un signal à temps continu est suivi d'un codage de chaque échantillon x en une valeur $Q(x)$ représentée sous forme numérique (conversion analogique-numérique). Cette opération constitue une discrétisation du signal avec un pas de quantification Δ que l'on supposera constant. Pour $i\Delta \leq x \leq (i+1)\Delta$, le procédé de quantification retenu dans cet exercice est un arrondi de la valeur de x à $\pm \frac{\Delta}{2}$.

Si on utilise un code binaire sur b bits, la plage de codage vaut $A = 2^b \Delta$.

On définit l'erreur de quantification par $e(x) = Q(x) - x$

1. Déterminez l'expression de l'erreur de quantification pour x situé dans le $i^{\text{ème}}$ intervalle de quantification et tracez l'allure du graphe correspondant.

2. On admet que l'erreur de quantification e , encore appelée bruit de quantification, est une variable aléatoire continue non corrélée à x et dont la densité de probabilité est uniforme. Montrez que e est centrée et exprimez sa variance σ_e^2 en fonction de Δ .

3. On définit le rapport signal à bruit de quantification en dB par : $\Gamma_{dB} = 10 \log_{10} \left(\frac{\sigma_x^2}{\sigma_e^2} \right)$

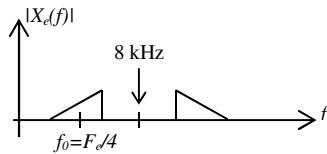
Dans cette expression, σ_x^2 représente la variance de x . La dynamique de x est supposée ne pas dépasser la plage de codage, soit $-\frac{A}{2} \leq x < \frac{A}{2}$. Déterminez l'expression de Γ en fonction de σ_x , du nombre b de bits et de l'amplitude A de la plage de codage. Quel est l'apport en décibel (dB) d'un bit supplémentaire.

4. Calculez la valeur maximum de Γ avant saturation pour un codage sur 16 bits pour un signal sinusoïdal et pour un signal gaussien dont la valeur crête sera saturée à $4\sigma_x$.

9.2 Transformée de Fourier

EXERCICE N°1

Soit un signal numérique **réel** $x(nT_e)$ dont le spectre, pour les fréquences "positives", est représenté ci-dessous :



On multiplie terme à terme les échantillons de ce signal par un signal sinusoïdal réel de fréquence normalisée

$f_{1n} = 0.0125$ pour former un nouveau signal $y(nT_e)$. (Fréquence normalisée : $f_n = \frac{f}{F_e}$)

1. Quelle est la valeur de f_1 ?
2. Exprimez $y(nT_e)$ en fonction de $x(nT_e)$
3. Représentez le spectre de $y(nT_e)$ (on supposera la bande du signal faible devant la valeur de la fréquence f_1)

EXERCICE N°2

On considère un signal analogique $\Pi(t)$ défini de la manière suivante :

$$\Pi(t) = 1, t \in \left[\frac{-\tau}{2}, \frac{\tau}{2} \right]$$

$$\Pi(t) = 0, t \notin \left[\frac{-\tau}{2}, \frac{\tau}{2} \right]$$

On numérise ce signal avec une fréquence d'échantillonnage F_e

1. Quelle condition doit respecter F_e pour respecter le théorème de l'échantillonnage ?

On considère maintenant la porte numérisée de M échantillons :

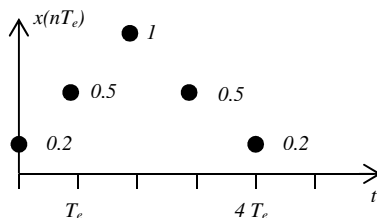
$$\Pi(nT_e) = 1, n \in [0, N-1]$$

$$\Pi(nT_e) = 0, n \in [N, M-1]$$

2. Calculez la Transformée de Fourier discrète de cette porte numérique et tracez le spectre de 0 à F_c en précisant les valeurs des passages par 0.

On introduit la Transformée en cosinus discrète suivante : $C(f) = \sum_{n=-\infty}^{+\infty} x(nT_e) \cos(2\pi f n T_e)$

3. Exprimez $C(f)$ en fonction de la Transformée de Fourier discrète $X(f)$ (le signal temporel est réel)
 4. Quelle condition faut-il avoir sur le signal numérique temporel $x(nT_e)$ pour avoir $C(f) = X(f)$
 5. Soient les N échantillons du signal suivant :



6. La Transformée de Fourier discrète de ce signal est-elle réelle ?
 7. Quel échantillon pourrait-on ajouter pour résoudre le problème ?

EXERCICE N°3

On considère une image que l'on notera I et qui comporte N lignes et M colonnes. On note $I(n, m)$, le pixel de la ligne n colonne m .

- Ecrivez la Transformée de Fourier de l'image en fonction de deux variables p et q

On conjugue cette Transformée et on effectue la Transformée inverse on obtient

- le négatif de l'image ?
- l'image d'origine ?
- l'image d'origine avec une symétrie par rapport à un axe vertical ?
- l'image d'origine avec une double symétrie verticale horizontale ?
- n'importe quoi ?

EXERCICE N°4

(Cas d'étude Radar)

On observe pendant $10 \mu s$ un signal radio que l'on échantillonne à 10 MHz . Ce signal est en fait l'écho d'un pulse radar, c'est à dire un petit morceau de sinussoïde, qui avait été émis un instant auparavant vers un véhicule en mouvement.

On considérera en première approximation dans cet exercice, que le pulse radar émis occupe une bande de 100 kHz (avec une mise en forme) et qu'il a été transposé en fréquence à 5 GHz. En réception il est redescendu en bande de base par transposition de fréquence.

1. Proposez un schéma global de la chaîne émission / réception
2. On effectue une FFT du signal reçu après transposition à la fréquence 0 Hz (dite bande de base), quelle précision peut-on obtenir sur l'estimation de fréquence ?
3. Quelle précision peut-on en déduire sur la vitesse du mobile compte tenu de l'effet Doppler ?
4. Ce principe de mesure de la vitesse peut-il fonctionner ?

9.3 Filtrage numérique RIF

EXERCICE N°1

On considère un filtre RIF symétrique réel pair dont les coefficients sont :

$$h_0 = 0.1, h_1 = 0.3, h_2 = 0.6, h_3 = 0.6, h_4 = 0.3, h_5 = 0.1$$

1. Calculer la réponse de ce filtre à la fréquence 0 Hz
2. Calculer la réponse de ce filtre à la fréquence normalisée 0.5
3. On inverse le signe des coefficients d'indice impair, quelle est la réponse du nouveau filtre obtenu à la fréquence normalisée 0.5
4. Donnez un schéma de réalisation du filtre. Combien faut-il de multiplications par valeur de sortie

EXERCICE N°2

Un filtre numérique à réponse impulsionnelle finie est défini par l'équation suivante :

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

Les coefficients a_k ont été calculés de telle sorte que ce filtre soit de type passe bas avec une fréquence de coupure f_c égale à $F_e / 20$. (F_e représentant la fréquence d'échantillonnage).

1. Que devient ce filtre si ses coefficients a_k sont remplacés par $a_k \cos(2 \pi f_0 k T_e)$ avec $f_0 = F_e / 5$.

EXERCICE N°3

Soit le filtre défini par l'équation : $y(n) = x(n) + 0.7x(n-1) + 0.9x(n-2)$

la fréquence d'échantillonnage est égale à 10 MHz, les signaux sont réels.

1. Quelle puissance de calcul (nombre d'opérations par seconde) minimale faut-il prévoir pour réaliser ce filtre ?

EXERCICE N°4

On considère une chaîne de traitement constituée par la mise en cascade des 4 filtres numériques suivants :

$$u(n) = \sum_{i=0}^7 a_i x(n-i) \quad , \quad v(n) = \sum_{j=0}^4 b_j u(n-j)$$

$$w(n) = \sum_{m=0}^6 c_m v(n-m) \quad , \quad y(n) = \sum_{k=0}^5 d_k w(n-k)$$

Soit $x(n)$ le signal en entrée et $y(n)$ le signal en sortie.

1. Ecrire $y(n)$ en fonction de $x(n)$?
2. Quel est l'ordre du filtre ainsi obtenu ?
3. Quel coefficient affecte $x(n-21)$?

EXERCICE N°5

On considère le filtre numérique RIF $A(z)$ à 64 coefficients tous égaux à 1. Soit $x(n)$ le signal à l'entrée du filtre et $y(n)$ le signal à la sortie du filtre. On se place à l'instant n .

- 1) Serait-il possible d'obtenir la sortie parmi les valeurs délivrées par une FFT du signal d'entrée.
- 2) Tracez la fonction de transfert en fréquence du filtre.

On multiplie terme à terme les coefficients du filtre par $e^{j2\pi \frac{(i \times 1)}{N}}$

- 3) Tracez la fonction de transfert en fréquence du nouveau filtre ainsi obtenu.
- 4) Serait-il possible d'obtenir la sortie à l'instant n au moyen d'une FFT.
- 5) On cherche à décomposer un signal au moyen d'un banc de 64 filtres fréquentiels équidistants. Proposez une solution en utilisant une FFT.

EXERCICE N°6

On considère l'interpolation de facteur 2 la plus simple qui consiste à insérer entre chaque valeur du signal numérique d'origine la demi somme des échantillons encadrant cette nouvelle valeur. Ecrire le filtre interpolateur, exprimez sa réponse en fréquence.

EXERCICE N°7

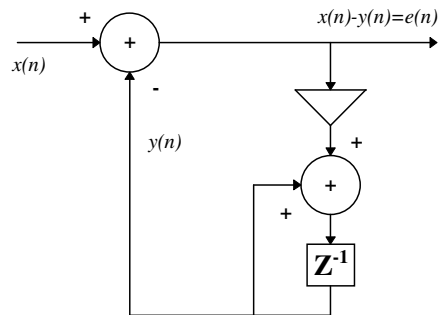
Donnez l'équation d'un filtre RIF d'ordre 2 qui supprime la fréquence $f_0 = 1 \text{ kHz}$, sachant que le signal a été échantillonné à $F_e = 8 \text{ kHz}$.

Comment faudrait-il modifier ce filtre pour qu'il supprime aussi le continu (la fréquence 0 Hz) ?

9.4 Filtrage numérique RII

EXERCICE N°1

Une boucle à verrouillage de phase est modélisée par le circuit suivant :



1. Ecrire la relation entre la suite d'entrée $x(n)$ et la sortie $y(n)$. En déduire la fonction de transfert en Z du système.
2. Quelle est la réponse à l'échelon unité. Vérifier qu'elle correspond bien à un asservissement de $y(n)$ sur $x(n)$, c'est à dire qu'elle tend vers l'unité quand n tend vers l'infini.
3. Le coefficient K représente le gain de l'asservissement. Dans quelle plage doit on choisir ses valeurs pour garantir la stabilité.

EXERCICE N°2

Soit le filtre numérique défini par ses zéros :

$$Z_1 = 0.09 \pm j 0.99$$

$$Z_2 = 0.58 \pm j 0.81$$

et par ses pôles :

$$P_1 = 0.62 \pm j 0.26$$

$$P_2 = 0.70 \pm j 0.58$$

1. De quel type de filtre s'agit-il (RIF, RII, ?)
2. Quel est son ordre
3. Quelle est sa fonction de transfert (passe haut, passe bas, passe bande ?)

EXERCICE N°3

On considère un filtre numérique dont la fonction de transfert en Z s'écrit :

$$H(Z) = \frac{b_2 + b_1 Z^{-1} + Z^{-2}}{1 + b_1 Z^{-1} + b_2 Z^{-2}}$$

1. De quel type de filtre s'agit-il (RII, RIF) et quel est son ordre ?
2. Calculez la réponse en fréquence de ce filtre.
3. On pose $H(Z) = \frac{N(Z)}{D(Z)}$ et on introduit $\varphi_D(\omega)$ comme étant la réponse en phase du dénominateur $D(Z)$, exprimez la réponse en phase de $H(Z)$ en fonction de ω et de $\varphi_D(\omega)$.

EXERCICE N°4

On considère le filtre RII suivant :

$$y(n) = x(n) - b_1 y(n-1) - b_2 y(n-2)$$

1. Quel signal d'entrée $x(n)$ conduit à $y(n) = \delta(n)$, discutez ce résultat.

On rappelle qu'une condition nécessaire et suffisante de stabilité pour un filtre numérique est d'avoir $\sum_{n=-\infty}^{n=+\infty} |h_n|$ bornée.

Expression dans laquelle h_n représente la $n^{ième}$ valeur de la réponse impulsionnelle du filtre.

On se place dans le cas où b_1 et b_2 sont tels que la fraction $\frac{1}{1 + b_1 Z^{-1} + b_2 Z^{-2}}$ a deux pôles complexes conjugués

P et P^*

2. calculez la réponse impulsionnelle du filtre
3. montrez que la condition de stabilité revient à avoir le pôle à l'intérieur du cercle unité
(On pourra décomposer la fraction en éléments simples et effectuer la division. L'écriture du pôle en coordonnées polaires peut être avantageuse pour le calcul)

EXERCICE N°5

On vous demande de réaliser un filtre numérique agissant sur un signal numérisé à 10 MHz. La bande passante s'étend de 0 à 100 Hz et la bande atténuée de 200 Hz à 5 MHz.

1. Quelle puissance de calcul faut-il grossièrement prévoir pour réaliser ce filtre avec un filtre RIF ? un filtre RII ?
2. Que pensez-vous du problème posé et auriez-vous quelques recommandations à faire pour le simplifier.
- 3.

EXERCICE N°6

4. Donnez l'équation d'un filtre RII d'ordre 2 qui supprime la fréquence $f_0 = 1 \text{ kHz}$, sachant que le signal a été échantillonné à $F_e = 8 \text{ kHz}$. On proposera un filtre à encoche avec un pôle de module égal à 0,9.
5. Comment faudrait-il modifier ce filtre pour qu'il supprime aussi le continu (la fréquence 0 Hz) ?

9.5 Signal aléatoire

EXERCICE N°1

On considère un signal aléatoire numérique $x(nT_e)$. Pour simplifier les écritures on suppose $T_e = 1$ et l'on écrit la réalisation du signal à l'instant T_e sous la forme $x(n)$. Le signal $x(n)$ a été obtenu en filtrant, au moyen d'un filtre à réponse impulsionnelle finie, un bruit blanc $b(n)$ gaussien centré de variance σ^2 . L'équation de filtrage est la suivante :

$$x(n) = 2.b(n) + 0.5.b(n-1) - 0.2.b(n-2) + 0.1.b(n-3)$$

1. 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.
2. Le signal $x(n)$ est-il blanc ?
3. La répartition des niveaux d'amplitude du signal $x(n)$ est-elle gaussienne (sans justifier)?
4. Formez la matrice d'autocorrélation d'ordre 3 que l'on notera R_3

On suppose maintenant que l'on dispose d'un autre signal noté $y(n)$, lui aussi obtenu par filtrage du bruit $b(n)$, mais s'écrivant cette fois ci :

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

5. Donnez la Transformée en Z de la réponse impulsionnelle du filtre qui a permis d'obtenir $y(n)$ à partir de $b(n)$.
6. Placez les zéros de ce filtre sur un cercle unité, quelles sont la ou les fréquence(s) coupées par ce filtre ? Tracez à main levée l'allure de sa réponse en fréquence.
7. Le signal $y(n)$ est-il blanc ?

On considère maintenant le coefficient d'intercorrélacion entre les signaux $x(n)$ et $y(n)$. On note $r_{xy}(p)$ ce

coefficient et il est défini par : $r_{xy}(p) = E[x(n)y(n-p)^*]$

8. Calculez $r_{xy}(0), r_{xy}(1), r_{xy}(2)$

EXERCICE N°2

On considère un filtre avec uniquement des zéros :

$$Z_0 = e^{j\frac{\pi}{4}}, Z_0^* = e^{-j\frac{\pi}{4}}$$

On fait passer un bruit $b(n)$ uniformément réparti entre $[0,1]$ à travers ce filtre. On note $x(n)$ la sortie du filtre.

1. Exprimez $x(n)$ en fonction du signal d'entrée.
2. Calculez les coefficients d'autocorrélation d'ordre 0 et 1 de $x(n)$. On notera $r_{xx}(0), r_{xx}(1)$ ces coefficients.

EXERCICE N°3

On considère le cas d'un signal $s(n)$ obtenu au moyen de deux capteurs. Sur le premier capteur le signal est reçu avec une variable de bruit additive $b_1(n)$ et avec une variable $b_2(n)$ sur le deuxième capteur. Ces deux variables de bruit sont supposées indépendantes, gaussiennes, centrées et de même variance égale à σ^2 . Le signal $s(n)$ est supposé être centré et normalisé, c'est à dire $E[s(n)] = 0$ et $E[s^2(n)] = 1$

1. Exprimez le rapport signal sur bruit sur chaque capteur. On notera $x_1(n)$ et $x_2(n)$ les signaux capteurs. Afin d'améliorer ce rapport signal sur bruit on décide de sommer les signaux issus des deux capteurs et de former un nouveau signal $y(n) = x_1(n) + x_2(n)$.

2. Exprimez le nouveau rapport signal sur bruit obtenu sur le signal somme $y(n)$.

On considère maintenant que le deuxième capteur fonctionne moins bien que le premier et que le signal qu'il reçoit s'écrit :

$$x_2(n) = \alpha s(n) + b_2(n)$$

Expression dans laquelle α représente un coefficient d'affaiblissement réel compris entre 0 et 1.

3. Donnez le rapport signal à bruit de combinaison obtenu par simple sommation en tenant compte de ce coefficient α .
4. Tracez ce rapport pour α variant entre 0 et 1 et discutez sur l'efficacité de la sommation.

On propose maintenant d'introduire un coefficient multiplicatif réel a sur la deuxième antenne.

5. Exprimez, en fonction de α et de σ^2 , la valeur optimale que doit prendre ce coefficients a afin de maximiser le rapport signal sur bruit en sortie de combinaison.

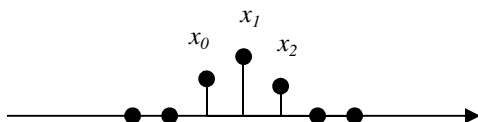
EXERCICE N°4

Soit un bruit blanc $b(n)$ de variance σ^2 qui traverse un filtre RII dont la transformée en Z est $H(Z) = \frac{1}{1-\alpha Z^{-1}}$. On note $x(n)$ le signal à la sortie du filtre.

- Calculez les termes $r_{xx}(p)$ de la fonction d'autocorrélation du signal $x(n)$

EXERCICE N°5

On considère le signal $x(nT_e)$ suivant :



1. Montrez, sur ce signal simple, que
$$\sum_{p=-(N-1)}^{N-1} \sum_{n=p}^{N-1} x(n)x(n-p) = \left(\sum_{n=0}^{N-1} x(n) \right)^2$$
2. Proposez une méthode pour généraliser ce résultat à toute observation d'un signal sur N points

9.6 Prédiction linéaire

EXERCICE N°1

On considère l'observation tronquée de 6 échantillons d'un signal aléatoire $x(n)$:

$$0.23 \quad -0.1 \quad 0.12 \quad -0.4 \quad -0.3 \quad 0.1$$

On suppose le signal nul en dehors de ces valeurs. On choisit d'estimer, à partir d'une observation de N valeurs, le coefficient d'autocorrélation du signal $x(n)$ au moyen de la formule :

$$r_{xx}(p) = \frac{1}{N} \sum_{n=p}^{N-1} x(n)x(n-p)^*$$

1. Calculez les valeurs de $r_{xx}(p)$, pour $p \in [-\infty \quad +\infty]$
2. Donnez la valeur du périodogramme de $x(n)$ pour $f = 0$
3. Donnez la valeur du corrélogramme de $x(n)$ pour $f = 0$

EXERCICE N°2

On considère un signal aléatoire stationnaire $x(n)$ et l'on suppose connus ses coefficients d'autocorrélation :

$$r(0) = 3\sigma^2, \quad r(1) = 2\sigma^2, \quad r(2) = \sigma^2, \quad r(3) = 0$$

On cherche le filtre de prédiction linéaire d'ordre 3 de ce signal.

1. Utilisez l'algorithme de Levinson pour identifier successivement, en fonction des coefficients d'autocorrélation, les filtres prédictifs d'ordre 1, 2 et 3. On notera : $a_{1_1}, a_{1_2}, a_{2_2}, a_{1_3}, a_{2_3}, a_{3_3}$ ces trois filtres.
2. Tracez l'énergie de prédiction en fonction de l'ordre du prédictif pour $\sigma^2 = 1$. Que pouvez-vous dire sur l'évolution de cette énergie de prédiction.
3. On suppose que les autres coefficients d'autocorrélation $r(i)$ sont nuls pour $i \geq 3$. Calculez l'énergie de prédiction à l'ordre 4. Pensez-vous que l'énergie de prédiction va beaucoup évoluer si l'on poursuit la prédiction à des ordres supérieurs. Justifiez votre réponse.
4. 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 en Z $H(Z) = 1 + \alpha Z^{-1} + \beta Z^{-2}$, en déduire les valeurs de α et β

EXERCICE N°3

On considère un problème de transmission dans lequel un émetteur envoie un signal $s(n)$. Le récepteur reçoit ce signal augmenté d'une variable de bruit additive, que l'on notera $b(n)$. Le signal reçu sera noté $x(n) = s(n) + b(n)$. On suppose que le signal émis $s(n)$ est centré et de puissance normalisée et que les échantillons sont indépendants. Le bruit $b(n)$ sera supposé gaussien blanc de variance σ^2 . Le bruit additif et le signal émis peuvent être considérés comme deux variables aléatoires indépendantes.

1. Calculez le rapport de puissance entre le signal utile et le bruit dans le signal reçu (ce rapport dit signal sur bruit se note SSB ou SNR en anglais).

Avant d'être reçu, le signal émis traverse un canal de propagation avec des échos multiples (canal dit multitrajets). On considère que la réponse de ce canal s'écrit $C(Z) = 1 + \alpha Z^{-1}$. Le terme de bruit additif est sommé après le canal multitrajets.

1. Exprimez le nouveau signal reçu $y(n)$ en fonction de $x(n)$ et de $b(n)$
2. Exprimez le nouveau rapport signal à bruit

On choisit de mettre en réception un filtre de prédiction linéaire à réponse impulsionnelle finie afin de "reblanchir le signal reçu". Exprimez le ou les coefficients de ce filtre blanchisseur dans le cas où il aurait un ou deux coefficients.

9.7 Analyse Spectrale

EXERCICE N°1

Le but de cet exercice est de démontrer que lorsqu'un signal aléatoire $x(n)$ de densité spectrale de puissance $P_x(f)$ traverse un filtre numérique de réponse impulsionnelle finie de N termes $h_i, i=0,1,\dots,N-1$ et de fonction de transfert en fréquence $H(f)$, alors cela donne un signal $y(n)$ en sortie dont la densité spectrale de puissance $P_y(f)$ est égale à :

$$P_y(f) = |H(f)|^2 P_x(f).$$

1. démontrez ce résultat (! c'est calculatoire)

EXERCICE N°2

On considère un signal aléatoire $x(n)$ que l'on suppose être blanc et de puissance P_x . Ce signal subit un phénomène d'écho, c'est à dire qu'il est sommé à une version retardée et légèrement atténuée de lui-même. On note $y(n)$ le signal avec l'écho. On note α le coefficient d'affaiblissement et on suppose que le retard τ dû à l'écho peut être assimilé à un nombre exact de périodes d'échantillonnages. On introduit ainsi p tel que $\tau = pT_e$.

1. Ecrivez $y(n)$ en fonction de $x(n)$

2. Quelle est la fonction de transfert en Z du filtre qui permet de passer de $x(n)$ à $y(n)$
3. En supposant que vous observiez un grand nombre de valeurs de $y(n)$ mais que vous ne connaissiez pas $x(n)$, proposez une méthode pour estimer α et τ . (Vous savez juste que $x(n)$ est blanc)
4. Proposez une méthode pour supprimer l'écho

EXERCICE N°3

On échantillonne un signal avec une fréquence d'échantillonnage $F_e = 70 \text{ kHz}$. On effectue ensuite la Transformée de Fourier rapide de 512 valeurs de ce signal et on obtient ainsi 512 valeurs complexes (que l'on numérote de 1 à 512). En analysant les modules de ces valeurs, on identifie deux valeurs très nettement supérieures aux autres. Il s'agit de la 18^{ème} valeur et de la 496^{ème}. On en déduit qu'il y avait sans doute un signal sinusoïdal de forte amplitude dans le signal de départ.

1. Quelle était la fréquence de cette sinusoïde ?

EXERCICE N°4

On considère le signal $x(nT_e) = \alpha_0 \sin(2\pi f_0 nT_e)$

1. Montrez qu'il s'agit d'un signal autorégressif d'ordre 2 dont la variance de l'erreur d'estimation est nulle.
2. Calculez la fonction $r_{xx}(\tau)$ pour $\tau = 0, T_e, 2T_e, \dots, pT_e$
3. Quel est le rang de la matrice d'autocorrélation du signal $x(nT_e)$

On ajoute une deuxième sinusoïde :

$$x(nT_e) = \alpha_0 \sin(2\pi f_0 nT_e) + \alpha_1 \sin(2\pi f_1 nT_e) \text{ avec } f_0 \neq f_1 \text{ et } \alpha_i \neq 0 \text{ et réels.}$$

4. Peut-on trouver un filtre RIF qui annule totalement ce signal ?
5. Quel est le plus petit ordre de ce filtre ?

9.8 Estimation

EXERCICE N°1

On dispose de N observations x_i indépendantes de même moyenne et de même variance σ^2 d'un processus

aléatoire. On estime m par $\hat{m} = \frac{1}{N} \sum_{i=1}^N x_i$

1. \hat{m} est-il biaisé ?
2. Quelle est la variance de \hat{m}
3. Représentez cette estimation par un filtre et une décimation

EXERCICE N°2

On considère le signal $x(nT_e) = \alpha_0 \sin(2\pi f_0 nT_e)$

1. Montrez qu'il s'agit d'un signal autorégressif d'ordre 2 dont la variance de l'erreur d'estimation est nulle.
2. Calculez la fonction $r_{xx}(\tau)$ pour $\tau = 0, T_e, 2T_e, \dots, pT_e$
3. Quel est le rang de la matrice d'autocorrélation du signal $x(nT_e)$

On ajoute une deuxième sinusoïde :

$$x(nT_e) = \alpha_0 \sin(2\pi f_0 nT_e) + \alpha_1 \sin(2\pi f_1 nT_e) \text{ avec } f_0 \neq f_1 \text{ et } \alpha_i \neq 0 \text{ et réels.}$$

4. Peut-on trouver un filtre RIF qui annule totalement ce signal ?
5. Quel est le plus petit ordre de ce filtre ?

On ajoute un terme de bruit :

$$x(nT_e) = \alpha_0 \sin(2\pi f_0 nT_e) + \alpha_1 \sin(2\pi f_1 nT_e) + b(nT_e)$$

$b(nT_e)$ représente un échantillon de bruit blanc de variance σ^2

6. Formez la matrice d'autocorrélation d'ordre 3
7. Quelle sera l'énergie de l'erreur de prédiction dans le cas d'une prédiction linéaire à l'ordre 4 (on répondra uniquement pour les deux cas extrêmes du rapport signal à bruit)

EXERCICE N°3

On se place dans le cas d'un filtre prédictif réel agissant sur un signal réel.

On note $A = (a_1, a_2, \dots, a_N)^T$ les coefficients du filtre et $R = (r_{i,j})$ la matrice d'autocorrélation du signal. Montrez que la dérivée en fonction du vecteur a de la forme $A^T R A$ est égale à $2RA$.

EXERCICE N°4

On reçoit un bloc de données via une interface radio. On dispose ainsi de 500 échantillons qui sont des symboles BPSK auxquels s'est superposé un bruit additif blanc gaussien de variance σ^2 .

Les échantillons ont été échantillonnés à 10 MHz. On dispose de 40 symboles non modulés sur lesquels on effectue une FFT afin d'estimer un écart de fréquence. Quelle est la résolution fréquentielle que l'on obtient en sortie de FFT.

QCM récapitulatif

[1] Le fait d'échantillonner un signal a pour effet :

- a) de symétriser son spectre
- b) d'échantillonner son spectre
- c) de périodiser son spectre
- d) de périodiser le signal

[2] On considère un signal de musique que l'on filtre entre 0 et 24 kHz. Pour respecter le théorème de l'échantillonnage, la fréquence d'échantillonnage doit :

- a) être supérieure à 24 kHz
- b) être supérieure à 48 kHz
- c) être supérieure à 96 kHz
- d) être supérieure à 56 kHz

[3] La Transformée en Z de la réponse impulsionnelle d'un filtre à réponse impulsionnelle infinie possède :

- a) uniquement des zéros
- b) uniquement des pôles
- c) des pôles et des zéros
- d) des pôles ou des zéros mais pas les deux (ou exclusif)

[4] Un filtre numérique à réponse impulsionnelle infinie est :

- a) toujours stable
- b) stable si ses pôles sont à l'intérieur du cercle unité
- c) stable si ses pôles sont à l'extérieur du cercle unité
- d) stable si ses zéros sont sur le cercle unité

[5] On considère le filtre RII dont l'équation temporelle est la suivante : $y(n) = x(n) + 4x(n-1) - 2y(n-1)$, les premiers termes de sa réponse impulsionnelle sont :

- a) 1,0,0,0
- b) 1,5,3,-1
- c) 1,4,-2,0
- d) 1,2,-4,8

[6] On considère la Transformée de Fourier Rapide (TFR en français ou FFT en anglais) sur 256 valeurs d'un signal complexe quelconque échantillonné à une fréquence d'échantillonnage F_e . La FFT fournit :

- a) 256 valeurs réelles
- b) 256 valeurs imaginaires pures
- c) 256 valeurs complexes
- d) 128 valeurs complexes

[7] On dispose de 64 échantillons d'un signal numérique échantillonné à la fréquence $F_e = 230 \text{ kHz}$. Le module de la 15^{ème} valeur de la FFT du signal sur 64 valeurs est très nettement plus important que les autres valeurs. On en déduit qu'il y a dans le signal un signal sinusoïdal réel à la fréquence :

- a) 4.26 kHz

- b) 28.75 kHz
- c) 53.9 kHz
- d) 50.31 kHz

[8] Le périodogramme est un :

- a) estimateur de la Transformée de Fourier
- b) estimateur de la densité de probabilité
- c) estimateur de la densité spectrale de puissance
- d) un prédicteur linéaire

[9] L'algorithme de Levinson est

- a) un algorithme d'identification spectrale
- b) un algorithme de calcul de coefficients de prédiction linéaire
- c) un algorithme de décomposition d'une matrice en éléments propres
- d) un algorithme de calcul de la Transformée de Fourier

[10] Un filtre anti-aliasing (anti repliement spectral) doit être placé :

- a) derrière le CAN
- b) devant le CAN
- c) derrière le CNA
- d) devant le CNA

CAN: Convertisseur Analogique/Numérique

CNA: Convertisseur Numérique/Analogique

[11] Un filtre RIF devant séparer une bande passante d'une bande atténuée avec un bande de transition de 10 kHz et avec un fréquence d'échantillonnage de 1 MHz va demander (ordre de grandeur) environ :

- a) 10 coefficients
- b) 100 coefficients
- c) 1000 coefficients
- d) 10000 coefficients

[12] Si on respecte le théorème de Shannon :

- a) il est impossible de reconstruire les valeurs du signal entre les instants d'échantillonnage
- b) il est possible de reconstruire les valeurs du signal entre les instants d'échantillonnage
- c) il est possible de supprimer les échantillons d'indice pairs
- d) il est possible de calculer la Transformée de Fourier avec N opérations (N représentant le nombre d'échantillons disponibles)