

Traitement numérique du signal

Gabriel CHARDIN

Traitement numérique du signal

*Troisième école d'automne sur la détection de
rayonnement à très basse température*

Gabriel CHARDIN
DAPNIA/SPP, CE-Saclay
chardin@hep.saclay.cea.fr

Plan

Introduction et avertissement

I - Quelques définitions

- signaux analogiques et digitaux
- ADC, DAC et al.
- Quelques performances typiques: numérisation et transfert

II - Transformée de Fourier

- Historique rapide
- Définition et propriétés fondamentales
- Convolution et FFT
- Aspects numériques: FFT, ex. : 10^3 , 10^6 échantillons
- Et si je prends un nombre d'échantillons $\neq 2^n$?
- Identité de Parseval
- Théorème de Wiener-Khinchin
- Approximation d'une fonction par les n premiers termes de Fourier
- Le phénomène de Gibbs
- Fenêtrage: les principales fenêtres utilisées

III - Théorèmes fondamentaux

- Echantillonnage et perte d'information
 - Théorème d'échantillonnage
 - Fréquence de Nyquist
 - Filtre anti-repliement
- Inégalité de Schwartz
- Relation « d'incertitude » ; conséquences

IV - Les transformées de Laplace et la z-transformation

- Expression et propriétés
- Transformations inverses
- Passer du plan s au plan z
- Relation avec la transformée de Fourier

VI - Filtrage numérique

- Définitions
- Filtrer dans le domaine des fréquences
- Filtres FIR et IIR
- Une recette simple pour construire un filtre de réponse spectrale (à peu près) quelconque
- Approximation au filtre passe-bas idéal
- Les filtres les plus courants; comportement en phase et en amplitude (garder ou pas la forme du signal d'entrée)

- Comment passer d'un filtre passe-bas à un filtre passe-haut, à un filtre passe-bande, à un filtre stop-bande ?
- Une recette simple afin de construire un filtre de réponse spectrale quelconque
- Considérations sur la stabilité des filtres digitaux
- Aspects numériques ; influence de la précision sur le filtrage ; erreurs d'arrondi. Oscillations : dépassement, cycles limites.

VIII - Ajustement dans l'espace des temps et des fréquences

Filtrage optimal (Wiener)

- Quel est le problème ?
- Hypothèses de calcul
- Ajustement linéaire au sens des moindres carrés dans l'espace des temps par un ensemble de fonctions
- Ajustement dans l'espace des fréquences
- Optimal filtering template
- Remarques
- Et si les conditions changent au cours du temps ?

IX - Quelques mots sur les transformées temps-fréquence

- Transformations en ondelettes
- Spectrogramme de Gabor
- Logiciels temps-fréquence commerciaux et publics

Quelques références bibliographiques

Appendice : LabVIEW, logiciel d'instrumentation multistandard ; le groupe d'utilisateurs CEA, CERN; le CD-ROM CEA/CERN/NI du groupe d'utilisateurs.

Introduction et avertissement

Résumer en quelques dizaines de pages les principes ou ne serait-ce que les principaux résultats d'un domaine aussi vaste que le Traitement numérique du signal tient de la gageure. Il ne faut donc pas s'attendre à trouver ici un cours complet mais plutôt quelques pistes sur un sujet qui nécessiterait de nombreux ouvrages afin d'être relativement exhaustif.

En outre, de nombreuses voies de recherche, comme les traitements en temps-fréquence, et particulièrement les traitements en ondelettes, ne sont ici qu'effleurés. J'ai par contre tenté d'expliquer plus en détail le filtrage optimal, sujet sur lequel plusieurs questions m'avaient été posées au cours de l'Ecole d'Aussois.

Les techniques de construction des filtres numériques, qui nécessitent une réelle expertise si l'on veut obtenir des performances optimales pour un faible nombre de coefficients du filtre, ce qui est parfois fondamental pour les applications en temps réel à grande vitesse d'acquisition, sont également traitées ici de façon assez cavalière, en insistant plutôt sur les grandes catégories de filtres, ainsi que sur certaines techniques qui permettent d'aborder la construction d'un filtre sans commettre de grossières erreurs.

On le voit, un très grand chemin sépare ces notes d'un véritable cours de traitement du signal. Espérons néanmoins que ces notes permettront une présentation de quelques-uns des théorèmes les plus fondamentaux du traitement du signal. On trouvera également dans la liste des références quelques ouvrages qui permettront au lecteur intéressé de poursuivre plus avant son étude.

Une grande partie de ce cours avait été basé sur des illustrations au moyen d'instruments virtuels (VIs) écrits au moyen du logiciel LabVIEW, bien adapté au traitement et à l'analyse des données. Le lecteur intéressé pourra se procurer, en m'envoyant un email, les instruments LabVIEW sur lesquels était basé le cours, s'il désire poursuivre son exploration de cette façon.

I - Quelques définitions élémentaires

• Signaux analogiques et digitaux

Un signal analogique, c'est comme la prose de Monsieur Jourdain, on le rencontre en permanence dans les signaux « continus » du monde courant. La plupart des phénomènes physiques, si on n'y regarde pas de trop près, dépendent à la fois du temps t comme d'une variable continue, la grandeur physique elle-même objet de la mesure, comme la tension aux bornes d'une résistance ou le courant traversant un circuit, possède une amplitude que l'on peut également considérer comme une variable continue.

Par exemple, la tension aux bornes d'une résistance alimentée par un courant sinusoïdal répond à une loi de la forme: $f(t) = A \sin(f_0 t)$

Discrétisation du temps: pour des questions technologiques et de quantité d'information à manipuler, on choisit en général d'utiliser un système qui va enregistrer la valeur de la grandeur f à des temps discrets t_i . Dans la suite, on supposera en outre que l'échantillonnage se fait à des temps t_i équidistants et, en choisissant de façon adéquate l'origine des temps :

$t_i = i * \Delta t$, où i appartient à \mathbb{N} ou à \mathbb{Z} . Il s'agit alors de l'échantillonnage d'un signal analogique.

Si en outre, et c'est le cas dans le traitement numérique du signal, les amplitudes elles-mêmes (et non plus seulement les valeurs des temps t_i) sont à valeurs discrètes, on parle de signal digital.

(Remarque : il existe des systèmes où la grandeur échantillonnée est de type analogique, même si la grandeur elle-même, ici la charge, est échantillonnée à temps discrets (par exemple, par commutation de capacité, les CCD sont une bonne approximation d'un tel système analogique à canaux discrets).

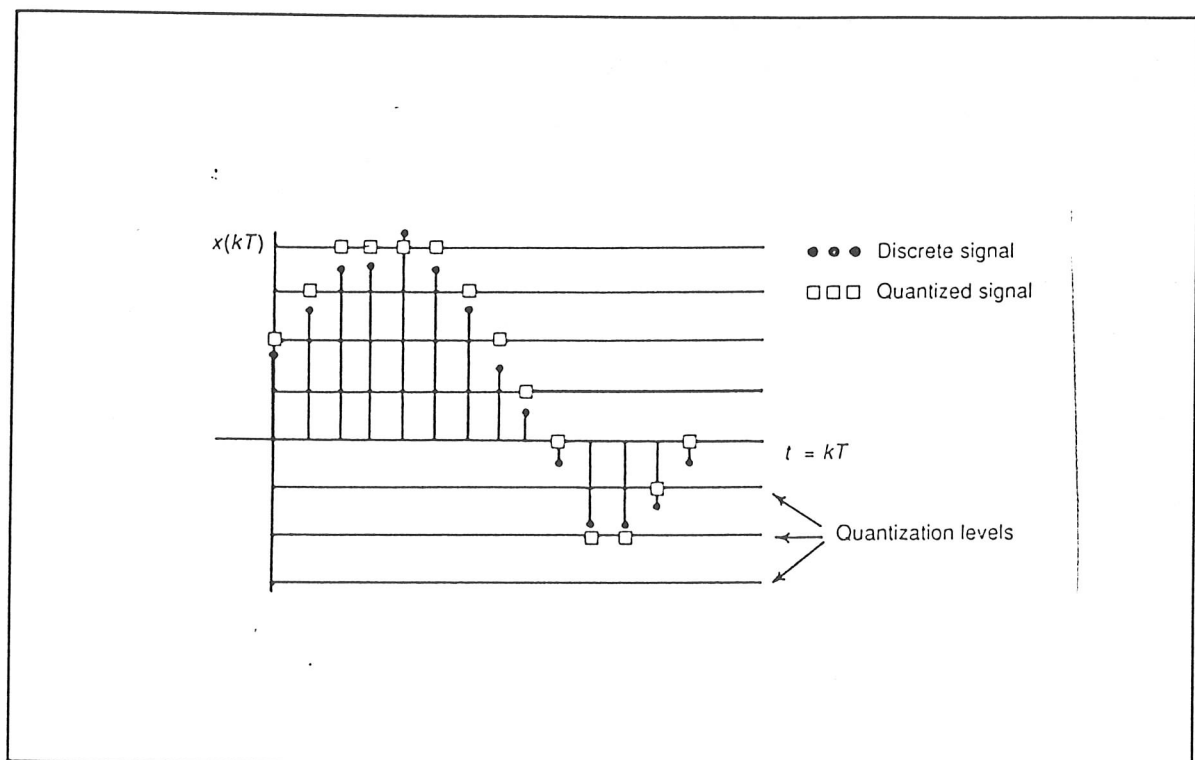


Fig.1 : L'échantillonnage d'un signal continu (« analogique ») met en général en jeu à la fois la discrétisation du signal dans son échantillonnage (échantillonnage à temps discret) ainsi que la quantification de l'amplitude, déterminée par comparaison à des niveaux discrets de référence d'amplitude.

Système linéaire, causal et t -invariant

On dira qu'un système de traitement d'un signal, qui reçoit en entrée un signal $f(t)$ et fournit en sortie un signal $g(t)$, est linéaire si, dans le cas où la réponse à un signal quelconque $f_1(t)$ est la fonction $g_1(t)$, et de même pour $f_2(t)$ avec pour

réponse $g_2(t)$, la réponse au signal d'entrée $a f_1(t) + b f_2(t)$ est le signal de sortie $a g_1(t) + b g_2(t)$.

On parlera de système causal si dans le cas d'un signal d'entrée $f(t)$ tel que $f(t) = 0$ pour $t < 0$, le signal de sortie $g(t)$ est tel que $g(t) = 0$ pour $t < 0$.

On supposera enfin que la réponse des systèmes que nous considérerons dans la suite est invariante dans le temps (c'est-à-dire que le fonctionnement du système ne dépend pas explicitement du temps). Dans la suite, on considérera principalement des systèmes linéaires causaux et invariants dans le temps.

• ADC, DAC et al.

Pour qu'un signal puisse être manipulé par un ordinateur, qui ne sait pas gérer les aspects continus d'un signal analogique, il faut qu'il soit numérisé.

Pour réaliser la numérisation d'un signal, on utilise un convertisseur analogique-digital ou ADC (Analog-to-Digital Converter). Une fois le traitement numérique effectué, il peut être utile de fournir à nouveau à un système externe un signal analogique et l'on parle alors, dans la conversion digital vers analogique, de DAC (Digital-to-Analog Converter).

En ce qui concerne les ADC, de nombreuses techniques existent pour numériser un signal. Signalons le cas particulier des Flash-ADC, dont la structure interne permet d'accélérer l'opération de numérisation en effectuant des opérations de comparaison parallèles: un flash-ADC (au moins au sens originel) sur 8 bits met ainsi en jeu 256 comparateurs, ce qui permet de parvenir à une numérisation très rapide sans réaliser d'intégration du signal, mais est pratiquement trop coûteux à mettre en œuvre pour des numériseurs 12 bits ou, pire encore, 16 bits.

Agissant sur les signaux numérisés, on trouve des processeurs spécialisés ou DSP (Digital Signal Processors) dont les performances progressent rapidement au fil des années, à tel point que beaucoup d'appareils qui mettaient jusqu'à récemment en jeu des composants discrets analogiques sont maintenant remplacés avantageusement, à la fois en termes de coût et de flexibilité, par des processeurs de signaux numériques. Du fait du rapide développement de la puissance des processeurs, on utilise de plus en plus des solutions numériques dans les problèmes de traitement de signal, notamment le filtrage, en raison de son faible coût, sa flexibilité et sa relative fiabilité.

S'il y a une quinzaine d'années, une puissance d'un mégaflops (soit un million d'opérations flottantes par seconde) nécessitait un investissement conséquent, Texas Instruments a sorti vers la fin des années 80 des versions d'une puissance d'une cinquantaine de mégaflops de son processeur vedette, et l'on voit maintenant offerts sur le marché et couplés à de simples PCs des systèmes à base de DSP dont la puissance atteint et dépasse même le gigaflops (soit un milliard d'opérations flottantes par seconde). L'évolution de la puissance des processeurs

et DSP en fonction du temps suit d'ailleurs approximativement une courbe exponentielle.

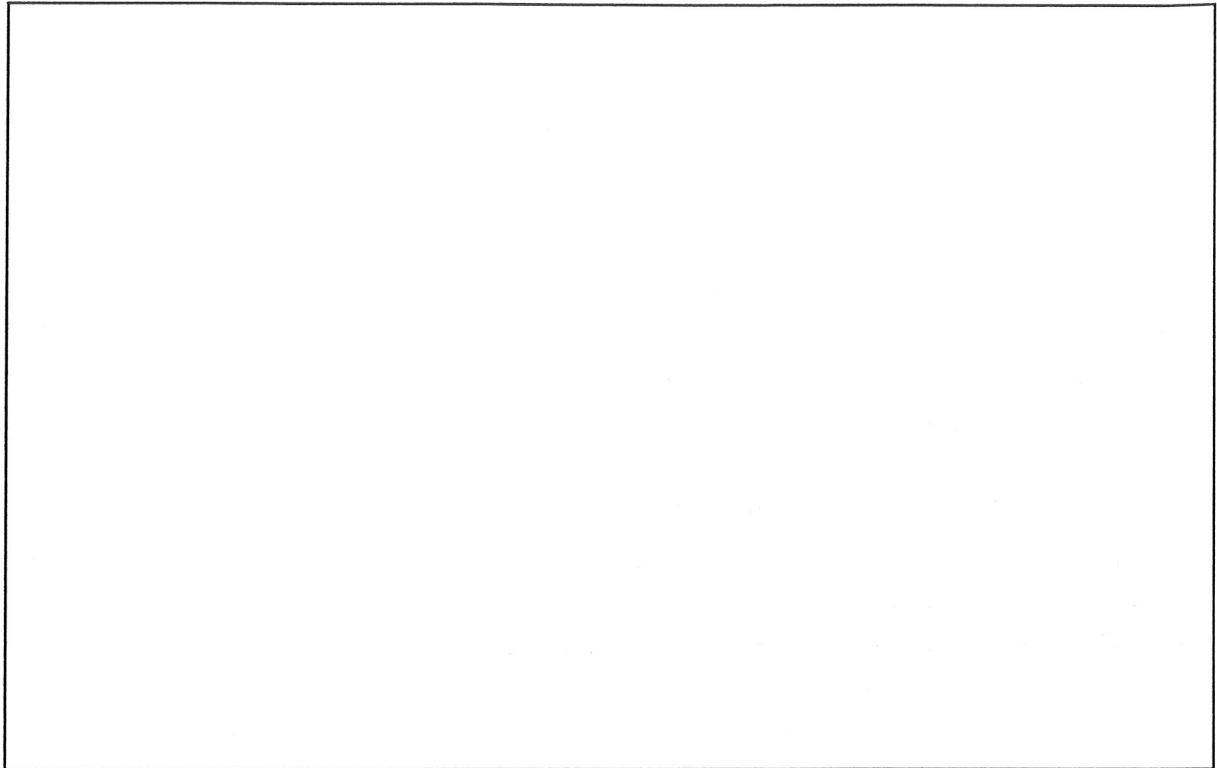


Fig.2 : Le développement de l'intégration de plus en plus poussée et des progrès concernant la dissipation dans les processeurs ont conduit au cours des trente dernières années à une croissance pratiquement exponentielle de la puissance de calcul pour un budget donné. Si cette croissance se poursuivait de cette façon, des systèmes possédant une puissance supérieure à celle du cerveau humain devraient apparaître dans quelques dizaines d'années au plus.

Le développement de la puissance des processeurs « généralistes » comme le Pentium d'Intel, le PowerPC d'IBM et de Motorola ou encore le processeur de la série Alpha de Digital, qui atteignent maintenant des puissances de plusieurs centaines de millions d'instructions par seconde pour leurs versions les plus puissantes, permettent eux-mêmes de réaliser des traitements inenvisageables il y a seulement quelques années. L'utilisation des processeurs spécialisés du type DSP permet néanmoins d'alléger considérablement la charge sur le processeur gérant une unité centrale de PC ainsi que de répondre à des impératifs de temps réel qui ne sont en général pas satisfaits par les systèmes d'exploitation généralistes présents sur les Macintosh ou autres PCs, qui peuvent abandonner leurs tâches pendant plusieurs dizaines de millisecondes.

De façon générale, les fonctions de base qui peuvent agir sur un (ou plusieurs) signaux sont :

- multiplication
- différentiation
- intégration

- addition

-...

(ces opérations peuvent d'ailleurs être effectuées de façon purement numérique ou sur le signal analogique en utilisant des composants discrets).

Comme on le verra dans la suite, la définition d'une stratégie adaptée en traitement du signal permet, grâce aux techniques de Transformée Rapide de Fourier (FFT) ou encore aux algorithmes « pyramidaux » de décomposition en ondelettes, d'analyser en fréquence ou de réaliser un filtrage, en temps réel, d'un signal échantillonné à quelques dizaines de mégahertz, et plus encore dans les applications de traitement d'image, ce qui était réservé à des installations extrêmement coûteuses et des objectifs essentiellement militaires il y a seulement une dizaine d'années.

• Quelques performances typiques: numérisation et transfert

On trouve couramment aujourd'hui des numériseurs (ADCs) 16 bits opérant à une centaine ou quelques centaines de kHz, des numériseurs 12 bits opérant à une centaine de MHz, des numériseurs 8 bits opérant à plusieurs GHz (ou même au-delà).

De plus en plus souvent, les constructeurs d'oscilloscopes numériques proposent des solutions numériques qui permettent d'augmenter la précision de numérisation. La technique utilisée met en jeu un suréchantillonnage à très grande vitesse (par rapport à la vitesse à laquelle on veut réaliser l'échantillonnage) du signal. De façon interne, l'oscilloscope réalise ensuite un traitement *mathématique* effectuant la moyenne des échantillons regroupés par paquets. Pour un suréchantillonnage d'un facteur 4, on peut ainsi gagner, dans l'hypothèse où le bruit du numériseur est blanc dans cette bande de fréquence (ce qui est généralement le cas pour les grandes vitesses de numérisation) un bit, soit un facteur 2, de précision dans la numérisation. Si l'on désire gagner deux bits en précision, il faut réaliser un suréchantillonnage d'un facteur 16 en vitesse (par exemple 160 Mhz pour une fréquence de numérisation désirée de 10 Mhz) afin de gagner le facteur $4 = \sqrt{16}$ équivalent aux 2 bits gagnés.

La fidélité et la précision des numériseurs sont généralement un ou deux bits en deçà du nombre de bits de numérisation. Par exemple, pour un numériseur 12 bits, il est assez rare que la fidélité de la numérisation dépasse 10 ou 11 bits (fidélité : on envoie un signal parfaitement calibré dans un numériseur, et on observe la dispersion du résultat des numérisations).

Un autre paramètre important concerne la linéarité de la numérisation: il est en effet essentiel, afin de ne pas déformer un signal, qu'une rampe de potentiel soit effectivement transformée par le numériseur en un escalier le plus linéaire possible. Afin de calibrer les numériseurs, certains constructeurs comme LeCroy ou Ortec proposent des générateurs de signaux calibrés qui permettent de vérifier la linéarité de l'ADC. La méthode qui fournit le plus d'information sur la fidélité,

le bruit et les caractéristiques de (non-)linéarité d'un numériseur met en jeu des mesures de distortion harmonique: un synthétiseur de signaux de grande précision fournit un signal d'entrée dont on varie la fréquence. La sortie numérique de l'ADC est analysée en fréquence, ce qui permet de définir notamment les paramètres de distortion harmonique et de puissance de bruit.

Signalons que le CERN a mis en œuvre, dans un de ses projets de Recherche et Développement, RD13, un banc complet de mesures des caractéristiques des numériseurs, et notamment de leur linéarité et de leur fidélité, dont la partie logiciel, le package LabADC écrit sous le logiciel LabVIEW, est une référence dans le domaine.

Notons par ailleurs que l'on a vu apparaître pour certaines applications des numériseurs 24 bits. Il ne s'agit pas de numériseurs capable d'une précision relative 10^{-7} , ce qui serait une précision impressionnante, mais de numériseurs adaptés à de très fortes dynamiques où l'on a en fait utilisé plusieurs numériseurs pour les différentes gammes dynamiques. La précision réelle pour chaque gamme étant de l'ordre de 8 à 10 bits, les bits restants reflètent seulement l'étendue de la dynamique des signaux que l'on veut étudier.

II - Transformée de Fourier

• Historique rapide

Il est pratiquement impossible de passer sous silence les principales propriétés de la transformée de Fourier, et de son implémentation rapide lorsque le nombre d'échantillons est égal à une puissance de 2, la FFT (Fast Fourier Transform).

Rappelons que tout signal périodique peut être décomposé en série de Fourier, comme Fourier l'a montré en 1807. Historiquement, les travaux de Fourier, qui avait montré que même des "non-fonctions" comme un créneau, ou des signaux en dents de scie, qui présentent des discontinuités et n'étaient pas considérées comme des fonctions par les mathématiciens de l'époque, ont joué un rôle considérable dans la redéfinition de la notion de fonction. L'un des problèmes les plus cruciaux venait du fait que dans l'expression :

$$f(x) = a_0/2 + \sum_{i=1}^{\infty} a_k \cos(k\pi x/l) + b_k \sin(k\pi x/l)$$

le membre de droite est une fonction analytique alors que Fourier montrait que sa décomposition pouvait être effectuée pour des fonctions possédant des discontinuités. Le travail de Fourier a eu une influence non seulement sur la notion de fonction, mais aussi sur la notion d'infini, de réalisation « presque partout », de l'intégrale de Lebesgue et sur la notion d'ensemble.

• Définition et propriétés fondamentale

Définissons, dans la limite continue, la transformée de Fourier d'une fonction h par

$$H(f) = \int_{-\infty}^{\infty} h(t)e^{2\pi jft} dt$$

On démontre que la transformée de Fourier inverse s'écrit simplement

$$h(t) = \int_{-\infty}^{\infty} H(f)e^{-2\pi jft} df$$

Notez qu'en utilisant comme variable la fréquence plutôt que la pulsation ω , on évite les facteurs 2π que l'on ne sait jamais très bien rétablir et qui font que la puissance d'un signal, par exemple, est toujours définie à une puissance de (2π) près...

Avec cette définition, il est facile de vérifier que si $h(t)$ et $H(f)$ sont considérées comme des fonctions conjuguées au sens de la transformée de Fourier, on a également les associations suivantes:

$h(at) \Leftrightarrow \frac{1}{ a } H\left(\frac{f}{a}\right)$	changement d'échelle de temps
$\frac{1}{ a } h\left(\frac{t}{a}\right) \Leftrightarrow H(af)$	changement d'échelle de fréquence
$h(t - t_0) \Leftrightarrow H(f)e^{2\pi jft_0}$	décalage temporel
$h(t) e^{-2\pi jft_0} \Leftrightarrow H(f - f_0)$	décalage en fréquence

Notons également l'importante relation, qui sera utile dans la suite:

$$\sum_{n=-\infty}^{\infty} \delta(t - n\Delta) \Leftrightarrow \sum_{n=-\infty}^{\infty} \delta\left(\omega - \frac{2n\pi}{\Delta}\right)$$

indiquant que la transformée de Fourier d'un « peigne de Dirac » est également un « peigne de Dirac ».

Passons maintenant à quelques propriétés un peu moins évidentes de la transformée de Fourier.

Convolution et transformée de Fourier

Une propriété extrêmement importante concerne également la convolution de deux fonctions. Nous verrons en effet que la convolution intervient dès que l'on veut exprimer le signal de sortie d'un système en fonction de la valeur du signal d'entrée et de ce que l'on appellera la fonction de réponse du système supposé ici linéaire. Sans encore entrer plus en détail dans ces questions, définissons le produit de convolution de deux fonctions g et h comme

$$g*h \equiv \int_{-\infty}^{\infty} g(\tau)h(t - \tau) d\tau$$

En faisant une transformation de coordonnée entre t et $(t-\tau)$, on voit facilement que

$$g*h = h*g$$

Il est également facile de démontrer que la transformée de Fourier de la fonction $g*h$, considérée comme une fonction de t , est

$$g*h(t) \Leftrightarrow G(f)H(f)$$

La transformée de Fourier du produit de convolution de deux fonctions est donc égale au produit des transformées de Fourier.

Fonction de corrélation - Théorème de Wiener-Khinchin

Une autre propriété très importante qui reviendra souvent dans les problèmes d'estimation de la puissance d'un signal, et qui exprime le théorème de Wiener-Khinchin, fait intervenir la fonction de corrélation de deux fonctions g et h

$$\text{Corr}(g,h) \equiv \int_{-\infty}^{\infty} g(t+\tau)h(\tau) d\tau$$

On démontre de façon tout-à-fait identique au cas du produit de convolution (seule le signe de τ a changé dans l'expression, et l'on sait que changer le signe de l'argument revient pour la fonction transformée à prendre le complexe conjugué) que l'on a la propriété

$$\text{Corr}(g,h) \Leftrightarrow G(f)H^*(f)$$

Dans le cas particulier où la fonction h est égale à la fonction g , on a

$$\text{Corr}(g,g) \Leftrightarrow G(f)G^*(f) = |G(f)|^2 \text{ (théorème de Wiener-Khinchin).}$$

En d'autres termes, la fonction d'autocorrélation d'une fonction fournit par transformée de Fourier la puissance du spectre de cette fonction.

Identité de Parseval

Une autre identité qui se révèle également de grande importance est l'identité de Parseval

$$\int_{-\infty}^{\infty} g(\tau)g^*(t) d\tau \equiv \int_{-\infty}^{\infty} |g(\tau)|^2 d\tau = \int_{-\infty}^{\infty} |G(f)|^2 df$$

qui indique que le produit scalaire (ici la norme, mais on démontre facilement que c'est également vrai pour le produit scalaire) possède la même valeur qu'il soit exprimé dans l'espace des fréquences ou dans l'espace des temps.

L'une des intégrales ci-dessus (carré de la fonction ou de sa transformée de Fourier) définit la puissance totale associée à un signal. Si en général on ne s'intéresse qu'assez peu à la puissance en fonction du temps, d'intérêt limité, on utilise par contre beaucoup la notion de puissance sur un certain intervalle de fréquence, notamment pour comparer la puissance du signal à celle du bruit dans une zone de fréquence déterminée. Ceci conduit par exemple à la notion de rapport signal/bruit à travers une ligne de transmission déterminée.

Le lecteur doit être averti qu'il existe plusieurs définitions différentes de la puissance d'un signal. En effet, la puissance $|G(f)|^2$ considérée comme une fonction de la fréquence est définie dans l'approximation continue précédente comme une fonction de la fréquence qui va de $-\infty$ à $+\infty$, alors qu'en général on cherche plutôt à définir la fonction pour les seules valeurs *positives* de la fréquence. Pour un signal réel, les composantes de fréquence positive et négative fournissent les mêmes valeurs de puissance et l'on peut donc définir sans problème une fonction définie seulement sur les valeurs positives de la fréquence *en multipliant par 2* toutes les valeurs autres que le continu (le terme d'indice 0). Cette opération permet de conserver la puissance totale associée à un signal.

En général, si on veut visualiser le spectre en fréquence d'un signal, on représentera la fonction pour les seules valeurs positives de la fréquence, alors que pour les calculs, il est en général beaucoup plus commode d'utiliser la fonction comportant à la fois les fréquences négatives et positives.

Nous avons énoncé ces propriétés de la transformée dans le cadre continu. Que deviennent-elles dans le cas, beaucoup plus fréquent en pratique et qui nous intéresse ici, d'un signal échantillonné à des valeurs discrètes du temps ?

Transformée de Fourier de fonctions échantillonnées de façon discrète dans le temps

Dans un grand nombre de cas, le signal est échantillonné à des intervalles de temps réguliers, séparés par un intervalle de temps Δ . Idéalement, la fonction d'échantillonnage consiste à prendre la valeur instantanée de la fonction, ce qui revient à approximer l'opération d'échantillonnage par la multiplication par une fonction δ de Dirac. Dans une option plus réaliste, on pourra approximer l'échantillonnage par la modulation par une porte de largeur finie (échantillonnage « naturel »).

Notons que très souvent, dans la pratique, on va effectuer la transformée de Fourier d'un signal qui n'est pas périodique, mais unique et limité dans le temps. On réalise alors en fait une extension de la fonction représentant le signal en la rendant périodique à partir de la reproduction du signal d'origine suivant le schéma :

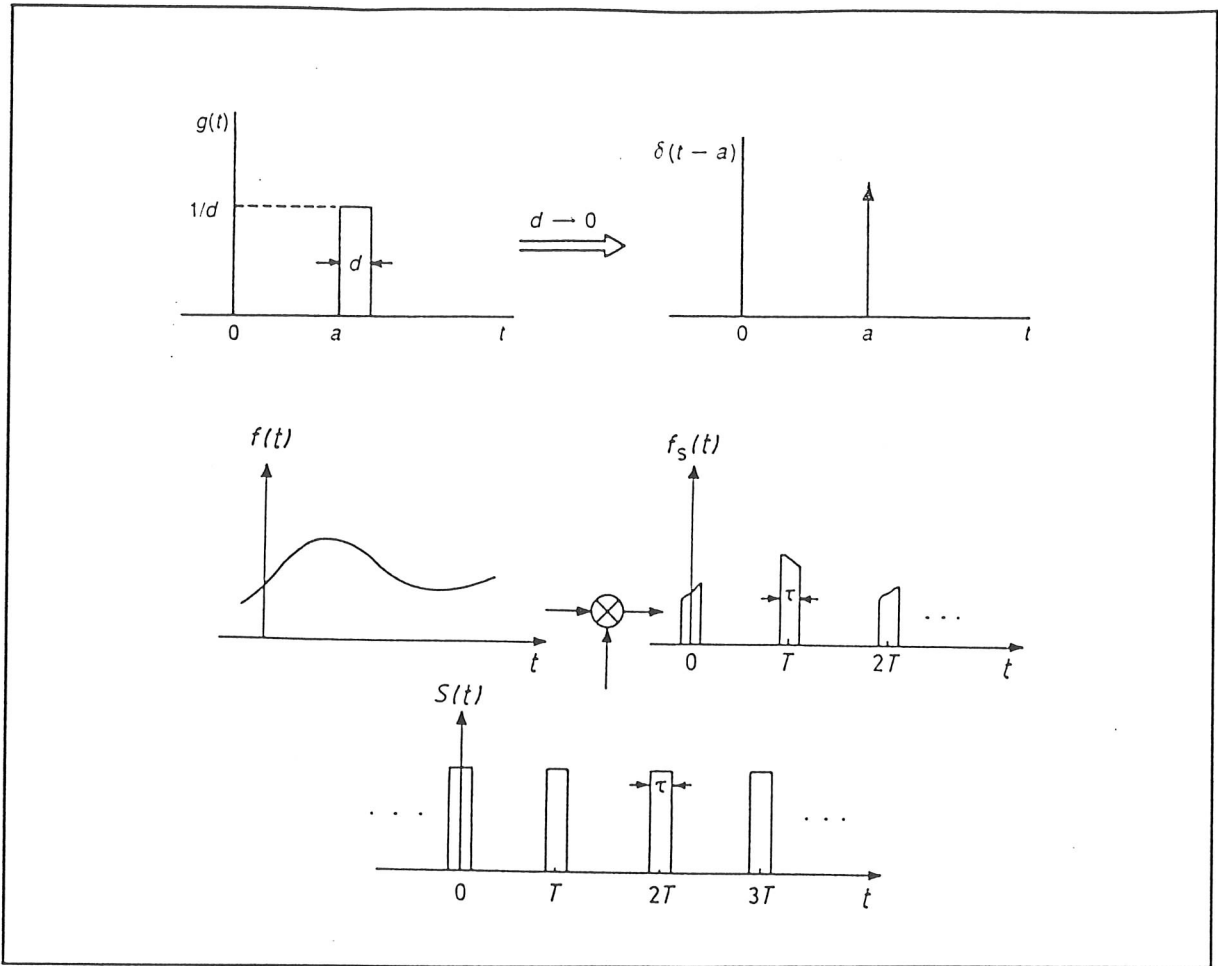


Fig. 3 : Si le « peigne de Dirac » (infiniment fin) peut être considéré comme la limite de largeur nulle de l'échantillonnage par une porte de largeur d , on appelle « échantillonnage naturel » l'échantillonnage par une porte de largeur d non nulle.

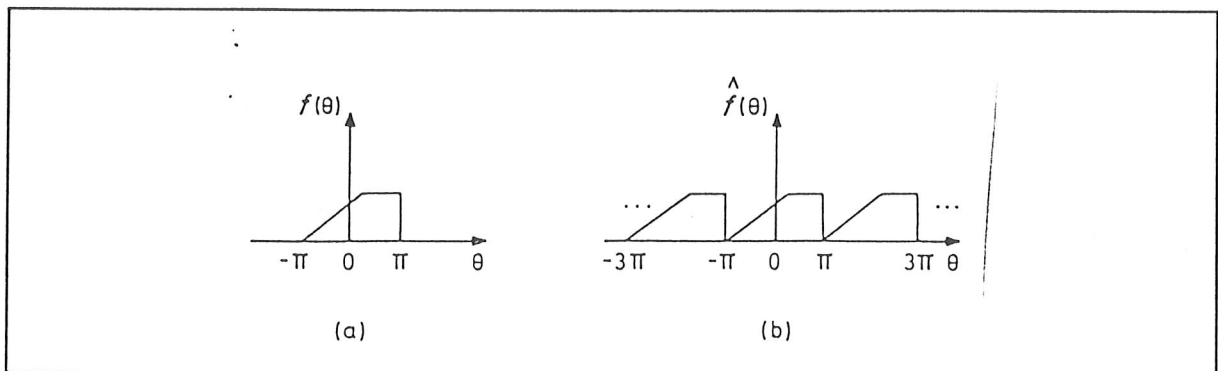


Fig. 4 : Si l'on analyse en termes d'une série de Fourier un phénomène unique, il ne faut pas oublier que tout se passe mathématiquement comme si l'on avait affaire à l'extension périodique de la fonction définie sur l'intervalle $[-\pi, \pi]$ de la variable angulaire de la transformée de Fourier.

Pour simplifier, on supposera que l'intervalle de définition de la fonction s'étend de $-T$ à $+T$. Afin de déterminer les coefficients de Fourier d'une fonction,

on utilise les propriétés d'orthogonalité des fonctions de base sinus et cosinus, afin d'obtenir :

$$a_k = 2/T \int_{-T/2}^{T/2} f(\vartheta) \cos k\vartheta d\vartheta$$

et une expression similaire (l'écrire et le vérifier explicitement à partir des relations d'orthogonalité) pour les coefficients b_k

$$b_k = 2/T \int_{-T/2}^{T/2} f(\vartheta) \sin k\vartheta d\vartheta$$

Exemple : soit $f(\theta) = -A$ si θ appartient à $[-\pi, 0]$ et $+A$ si θ appartient à $[0, \pi]$.

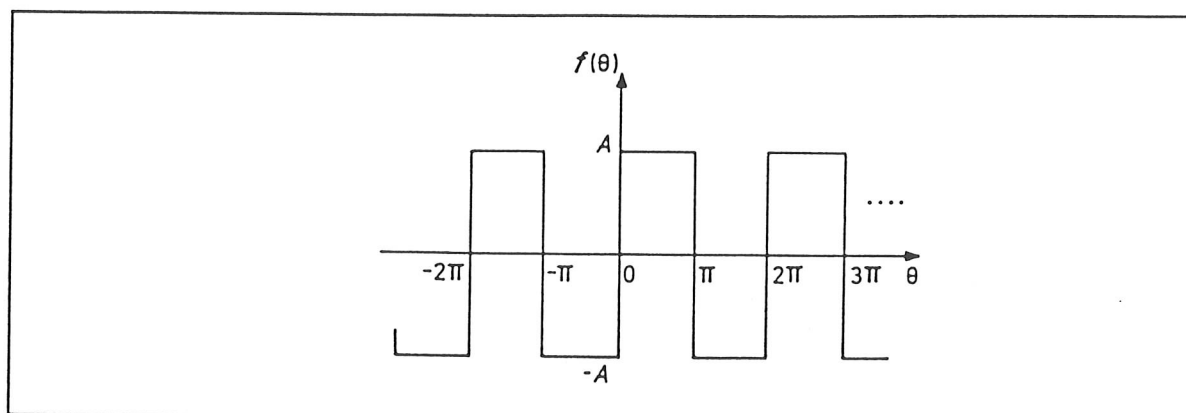


Fig.5 : La fonction créneau, bien que comportant des discontinuités, admet une décomposition en série de Fourier.

Sa transformée de Fourier vaut :

$$f(\vartheta) = \frac{4A}{\pi} \sum_0^{\infty} \frac{\sin(2k+1)\vartheta}{(2k+1)}$$

De façon générale, on parlera d'analyse spectrale pour la décomposition d'un signal en ses composantes de Fourier.

Aspects numériques: Transformée Rapide de Fourier (FFT)

Si l'on s'en réfère brutalement à l'apparence des formules décrivant les relations entre les coefficients dans l'espace des temps et dans l'espace des fréquences d'un signal, il semble à première vue que le calcul de la transformée de Fourier nécessite de l'ordre de $N \times N = N^2$ additions et multiplications afin de passer d'une représentation à une autre (temps à fréquence ou inversement).

Partons en effet de l'expression de la transformée discrète de Fourier notée F d'une fonction f , à partir de ses échantillons f_k pris à des intervalles réguliers

$$F_n = \sum_{k=0}^{N-1} e^{2i\pi nk} f_k \equiv \sum_{k=0}^{N-1} W^{nk} f_k$$

où l'on a noté $W = e^{2i\pi/N}$

Si l'on calcule brutalement tous les termes, le nombre d'opérations flottantes serait donc, pour un nombre de points égal à 1000, de l'ordre de quelques

millions, ce qui ne demande aujourd'hui qu'une fraction de seconde à un PC de bonne facture. Par contre, cette méthode demande des temps très conséquents (plusieurs jours à plusieurs semaines) si l'on cherche à calculer de cette façon la transformée de Fourier d'un ensemble de données comprenant de l'ordre d'un million d'échantillons.

C'est à J.W. Cooley et à J.W. Tukey que l'on doit, dans les années 60, d'avoir « découvert » puis popularisé l'un des algorithmes les plus efficaces et les plus utiles de l'analyse en fréquence d'un signal. Les guillemets de la phrase précédente viennent de ce qu'en fait un grand nombre d'individus avaient déjà découverts eux aussi des techniques similaires, mais n'avaient apparemment pas jugé utile de faire connaître leur découverte.

La technique de Cooley et Tukey consiste à réduire la difficulté du problème grâce à des dichotomies successives qui permettent de se ramener finalement à des transformées de Fourier à un seul élément, c'est-à-dire à l'identité (voyez-vous pourquoi ?).

Comme le nombre d'étapes nécessaires avant d'arriver à l'étape finale de la dichotomie est égal au logarithme en base 2 du nombre d'échantillons, le nombre total d'opérations flottantes impliquées dans la transformation est maintenant de l'ordre de $N \log_2(N)$. La différence de ce nombre d'opérations avec le nombre précédent permet maintenant d'envisager, même sur des PCs de puissance standard, la transformée de Fourier de notre signal échantillonné sur de l'ordre d'un million de points en quelques secondes, au lieu des jours ou même des semaines du calcul précédent.

Le principe de la dichotomie, reprenant un algorithme développé par Danielson et Lanczos en 1942 (!), est le suivant. On décompose la transformée de Fourier discrète en deux sommes sur les termes pairs et impairs, notant $W \equiv e^{2\pi i / N}$

$$\begin{aligned} F_k &= \sum_{j=0}^{N-1} e^{2i\pi jk / N} f_j \\ &= \sum_{j=0}^{N/2-1} e^{2i\pi(2j)k / N} f_{2j} + \sum_{j=0}^{N/2-1} e^{2i\pi(2j+1)k / N} f_{2j+1} \\ &= \sum_{j=0}^{N/2-1} e^{2i\pi jk / (N/2)} f_{2j} + W^k \sum_{j=0}^{N/2-1} e^{2i\pi jk / (N/2)} f_{2j+1} \\ &= F_k^{pair} + W^k F_k^{impair} \end{aligned}$$

On s'est donc ramené à une somme comportant maintenant chacune seulement $N/2$ termes, dont l'un des termes est multiplié par un facteur constant ($W \equiv e^{2\pi i / N}$). Notez que si, dans l'expression précédente, l'indice k défile toujours sur l'ensemble des valeurs de 0 à $N-1$, les deux termes pair et impair sont périodiques en k de période $N/2$ et non pas N , ce qui fait que l'on a effectivement réduit la complexité du problème par, approximativement, un facteur 2.

Rien ne nous interdit maintenant de continuer en décomposant à nouveau chacun des termes en ses termes pairs et impairs, notés i et p afin de simplifier les

notations dans la suite. Nous pouvons donc par exemple définir maintenant, dans une deuxième étape de dichotomie, les sommes partielles

$$F_k^{pp} = \sum_{j=0}^{N/4-1} e^{2i\pi jk/(N/4)} f_{4j}$$

$$F_k^{pi} = \sum_{j=0}^{N/4-1} e^{2i\pi jk/(N/4)} f_{4j+2}$$

$$F_k^{ip} = \sum_{j=0}^{N/4-1} e^{2i\pi jk/(N/4)} f_{4j+3}$$

$$F_k^{ii} = \sum_{j=0}^{N/4-1} e^{2i\pi jk/(N/4)} f_{4j+1}$$

où l'on a regroupé, respectivement, les indices pair-pair, pair-impair, impair-pair et impair-impair suivant la valeur du reste modulo 4 maintenant.

Rien ne nous interdit de continuer maintenant la procédure en classant les termes en fonction des restes successifs modulo 2^n que l'on peut classer de la même façon comme une suite d'indices pairs et impairs. Pour que la technique de dichotomie puisse être menée jusqu'à son terme, il faut (évidemment) que le nombre d'échantillons soit égal à une puissance de 2.

Lorsque chaque somme ne contient plus qu'un terme, la transformée discrète à un seul élément se réduit à l'identité puisque

$$\sum_{j=0}^0 e^{2i\pi jk} f_k = f_k$$

Le dernier problème à résoudre consiste maintenant à déterminer quel indice n associer à l'étape finale des dichotomies successives. Supposons par exemple que l'on veuille déterminer quel indice n correspond à la dichotomie

$$F_k^{ppiiipip\dots pppip} = f_n$$

En revenant à la deuxième dichotomie avec les termes

$$F_k^{pp}, F_k^{pi}, F_k^{ip} \text{ et } F_k^{ii}$$

on voit facilement qu'au terme $F_k^{ppiiipip\dots pppip}$ ci-dessus correspond l'indice n dont l'écriture en base 2 s'écrit $pippp\dots pipiiip$, c'est-à-dire le nombre dont l'écriture en base deux s'écrit en renversant l'ordre des indices et en remplaçant chaque p (pair) par 0 et chaque i (impair) par 1 dans l'expression ci-dessus (pour vous en convaincre, reprenez l'exemple des 4 premiers termes et regardez comment sont associés à leurs indices les termes avec reste 0, 1, 2, 3 modulo 4). Cette opération de renversement des bits, du fait qu'elle est une involution (elle est elle-même son inverse) permet de réaliser les associations de termes de la transformée rapide de Fourier sur place, c'est-à-dire sans créer de nouveau tableau intermédiaire en mémoire.

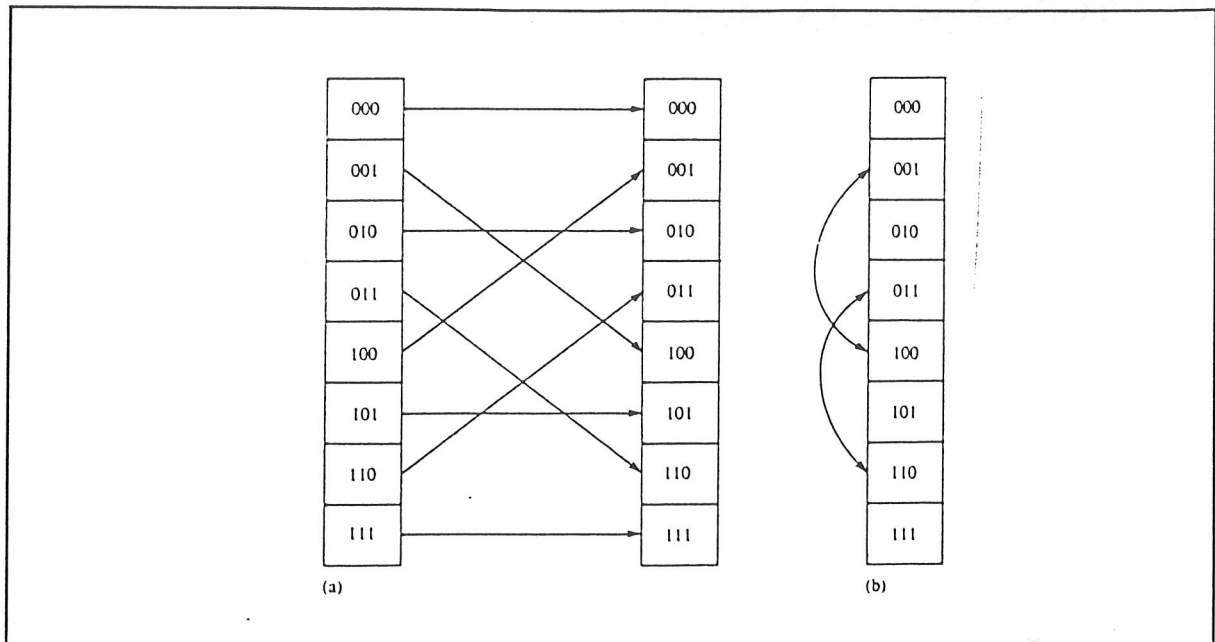


Fig. 6 : Au cours de la Transformée de Fourier Rapide (FFT), il est nécessaire de réorganiser les éléments en mémoire (par renversement des bits de l'écriture binaire de leur indice) avant de les multiplier par les puissances adéquates du facteur $W = e^{2i\pi/N}$. Comme on le voit sur la figure, avec seulement une mémoire intermédiaire, il est possible de réaliser cette transformation sur place, sans faire appel à la création (coûteuse en mémoire) d'un deuxième tableau pour la Transformée de Fourier.

Notons que même s'il existe des techniques qui permettent de limiter le nombre des opérations dans le cas d'un nombre d'échantillons qui n'est pas égal à une puissance de 2, il est *très déconseillé* de choisir un tel nombre, cela pour deux raisons. En effet, les techniques les plus efficaces mettent en jeu un nombre au moins égal à trois fois le nombre d'opérations mises en jeu dans le cas de la puissance de 2 immédiatement supérieure. Plus grave encore pour la majorité des applications, la transformation ne peut pas se faire sans des manipulations mémoire importantes, alors que l'algorithme de dichotomie peut se faire « sur place », sans créer de tableau intermédiaire en mémoire.

En conclusion, utilisez *toujours* un nombre d'échantillons égal à une puissance de 2 dans tous vos calculs de transformée de Fourier. Dans le cas contraire, si vous avez de la chance, votre machine ne mettra « que » trois fois plus de temps à effectuer votre calcul, mais à un coût en mémoire qui nécessite au moins ce nombre d'échantillons.

Faute de cette précaution, dans le cas où votre logiciel n'est pas sophistiqué, la différence de temps calcul, notamment pour un nombre d'échantillons ayant des gros facteurs dans sa décomposition en nombres premiers, peut devenir parfaitement rédhibitoire pour de grands nombres d'échantillons ($> 10^4$ typiquement).

III - Théorèmes fondamentaux

Dans le chapitre consacré aux propriétés de la transformée de Fourier, nous avons vu que nous pouvions considérer le processus d'échantillonnage comme la multiplication de la fonction continue originelle par un « peigne de Dirac » dans le cas d'un échantillonnage idéal, ou par des fenêtres carrée de largeur finie dans le cas de l'échantillonnage « naturel » (figures 3 et 4). Intuitivement, on a la sensation que l'on « perd de l'information » au cours de cette procédure d'échantillonnage. Le but du théorème suivant, le théorème d'échantillonnage, est de préciser dans quelles conditions on perd ou non de l'information dans cette opération.

Théorème d'échantillonnage :

Si une fonction continue $h(t)$ de transformée de Fourier $H(f)$ est échantillonnée à un intervalle Δ (donc à une fréquence $1/\Delta$), et possède un spectre limité aux fréquences inférieures en valeur absolue à $1/2\Delta$, c'est-à-dire si

$$H(f) = 0 \text{ quelle que soit } |f| > f_c = \frac{1}{2\Delta}$$

il est alors possible de la reconstruire *exactement* à partir de ses échantillons pris à fréquence $1/\Delta$.

De façon plus précise, l'expression qui permet de retrouver la fonction *continue* h à partir de ses valeurs échantillonnées h_n est

$$h(t) = \sum_{n=-\infty}^{n=\infty} h_n \frac{\sin\left[\frac{\pi}{\Delta}(t-n\Delta)\right]}{\frac{\pi}{\Delta}(t-n\Delta)}$$

Ce théorème remarquable, dû à Shannon, indique que l'on ne perd *aucune information* sur un signal s'il est limité dans son spectre de fréquence à une bande de fréquence finie, et si l'on échantillonne suffisamment vite, c'est-à-dire à au moins deux fois la fréquence la plus élevée contenue dans le signal.

Afin de démontrer ce théorème, revenons à l'expression de la fonction échantillonnée $h_s(t)$ en fonction de la fonction analogique $h(t)$ originelle:

$$h_s(t) = \sum_{n=-\infty}^{\infty} h(n\Delta)\delta(t-n\Delta)$$

Regardons maintenant l'influence du processus d'échantillonnage sur la transformée de Fourier de la fonction h

Nous avons vu que la transformée de Fourier du produit de deux fonctions est égal au produit de convolution des transformées de Fourier de ces fonctions. En utilisant la propriété citée plus haut sur la transformée de Fourier d'un « peigne de Dirac », soit:

$$\sum_{n=-\infty}^{\infty} \delta(t - n\Delta) \leftrightarrow \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{\Delta}\right)$$

nous pouvons donc écrire que la transformée de Fourier de $h_s(t)$, la fonction échantillonnée, s'écrit:

$$\frac{1}{\Delta} H(\omega) * \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{\Delta}\right)$$

où le caractère $*$ représente le produit de convolution. Mais la distribution de Dirac δ est l'élément neutre pour la convolution, ce qui implique que l'expression précédente se simplifie en:

$$\frac{1}{\Delta} \sum_{n=-\infty}^{\infty} H\left(f - \frac{n}{\Delta}\right)$$

En d'autres termes, le processus d'échantillonnage crée une infinité de clones décalés en fréquence de la quantité $1/\Delta$.

On voit sur la figure 7 que l'opération d'échantillonnage est réversible, et donc que l'on n'a pas perdu d'information au cours de la transformation, à condition que les clones ne se recouvrent pas. Faute de quoi, à l'évidence, il devient impossible de faire la différence entre le recouvrement des spectres individuels et le spectre composite considéré comme original. Cela nécessite que le signal étudié ne contienne pas de fréquences supérieures à $f_m = 1/2\Delta$, également appelée la *fréquence de Nyquist*.

Dans le cas contraire, il est donc essentiel, avant d'analyser en fréquence un signal, de préfiltrer la zone de fréquence supérieure à la fréquence de Nyquist $f_m = 1/2\Delta$. En pratique, on prend un peu de marge car un filtre infiniment raide est non-causal et pas réalisable physiquement (voir le chapitre sur le filtrage). Typiquement, on utilisera un filtre avec une fréquence de coupure égale à environ la moitié de la fréquence de Nyquist, par exemple, afin de tenir compte de la raideur finie de tout filtre réaliste.

Si l'on désirait reconstruire le signal initial dans sa version continue, il conviendrait, en raison de la présence des « clones » dans le spectre du signal échantillonné, d'appliquer une coupure en fréquence afin de ne garder que le clone centré sur la fréquence nulle (figure 7), qui n'est autre que le spectre original du signal continu. Dans l'espace des fréquences, cela revient à multiplier le spectre par la fonction

$$G(f) = 1 \quad f \in [-1/2\Delta, 1/2\Delta]$$

$$G(f) = 0 \quad \text{autrement.}$$

Sa transformée de Fourier inverse est

$$g(t) = \frac{\sin \frac{\pi}{\Delta} t}{\frac{\pi}{\Delta} t}$$

ce qui conduit simplement (exercice) à la formule de Shannon pour la fonction reconstruite à partir de ses échantillons

$$h(t) = \Delta \sum_{n=-\infty}^{n=\infty} h_n \frac{\sin\left[\frac{\pi}{\Delta}(t - n\Delta)\right]}{\pi(t - n\Delta)}$$

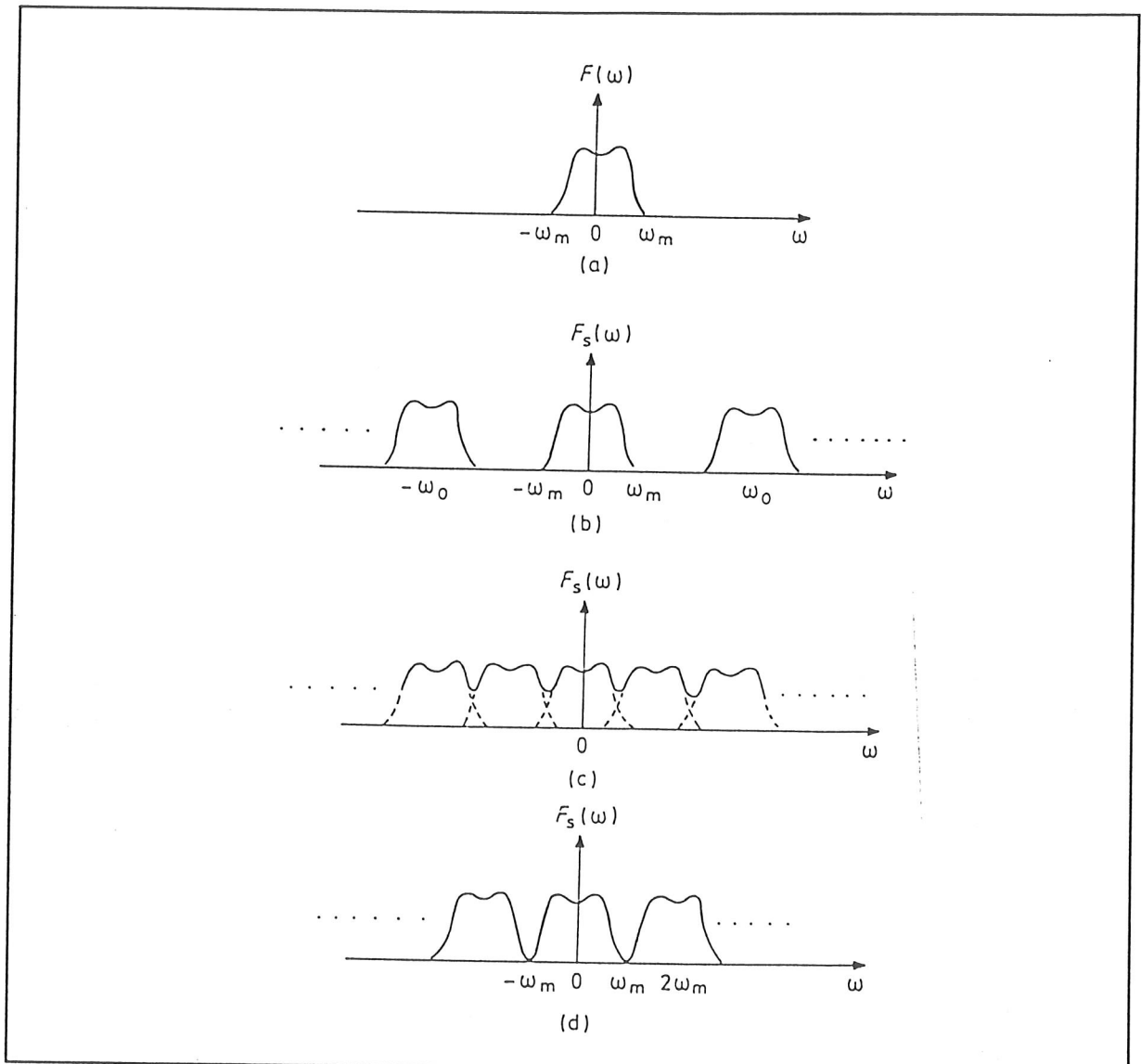


Fig. 7 : L'échantillonnage à temps discret à pour conséquence le phénomène de repliement des fréquences. Si la distribution de fréquence du signal original est représentée sur la partie (a) de la figure, le processus d'échantillonnage conduit dans l'espace des fréquences à la duplication du spectre translaté à chaque fois par la fréquence d'échantillonnage (b).

Faute de réaliser ces conditions, ou d'effectuer un préfiltrage adéquat, on a le phénomène de repliement de fréquence. Le phénomène d'aliasing s'observe sur un signal répétitif lorsque l'échantillonnage d'un signal trop rapide fournit l'image d'un signal à fréquence plus faible. On a donc deux signaux différents qui ont mêmes valeurs d'échantillonnage, ce qui montre que l'on n'est plus dans les

conditions du théorème de Shannon sur la reconstruction fidèle du signal initial à partir de sa numérisation. C'est pour contrer ce phénomène d'aliasing et de repliement de fréquence que l'on commence en général par préfiltrer le signal, lui appliquant ce que l'on appelle un *filtre anti-repliement*.

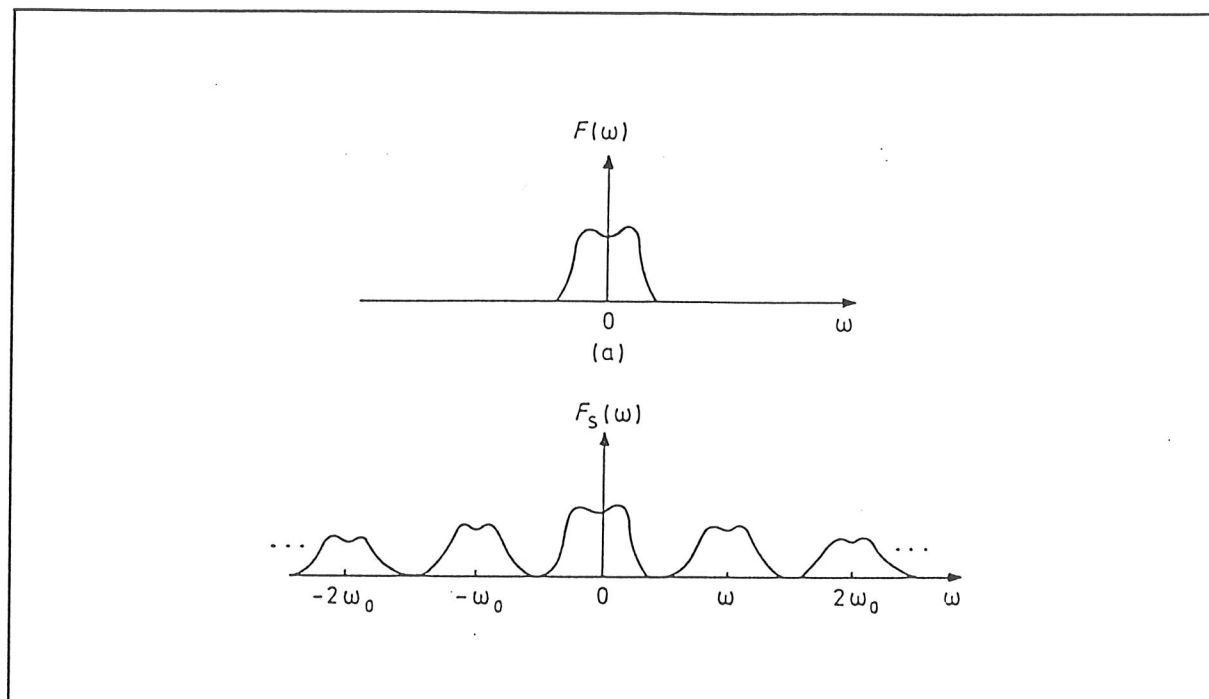


Fig. 8 : L'échantillonnage « naturel », par une largeur de porte finie, conduit aux mêmes conclusions qualitatives concernant le phénomène de repliement en fréquence. Pour une largeur finie de la fonction d'échantillonnage, on remarquera cependant que les amplitudes des spectres aliasés dépendent de l'ordre, à la différence du peigne de Dirac où tous les ordres ont le même poids.

A l'inverse, lorsque l'on commence à échantillonner un signal, et que l'on ne sait pas très bien jusqu'à quelle fréquence s'étend son spectre, on peut « faire comme si » le signal satisfaisait au critère de Nyquist, calculer son spectre en fréquence, et regarder sa forme (figure 9). Si le spectre n'atteint pas des valeurs proches de zéro à la fréquence de Nyquist, il y a toutes les chances, par continuité, que le spectre s'étende au delà de cette fréquence et que le spectre obtenu ne reflète donc pas le spectre véritable du signal.

Il faudra donc, *a posteriori*, appliquer des filtres de fréquence de coupure de plus en plus faible jusqu'à ce que l'on voit décroître et tendre vers zéro la puissance au voisinage de la fréquence de Nyquist. On aura alors satisfait aux conditions du théorème d'échantillonnage, et le spectre observé reflètera a priori le spectre du signal *filtré*.

Si l'on veut absolument obtenir le spectre du signal *non filtré*, il faut alors échantillonner à une fréquence plus élevée, jusqu'à ce que l'on observe à nouveau que la puissance du signal dans la partie haute du spectre tende vers zéro.

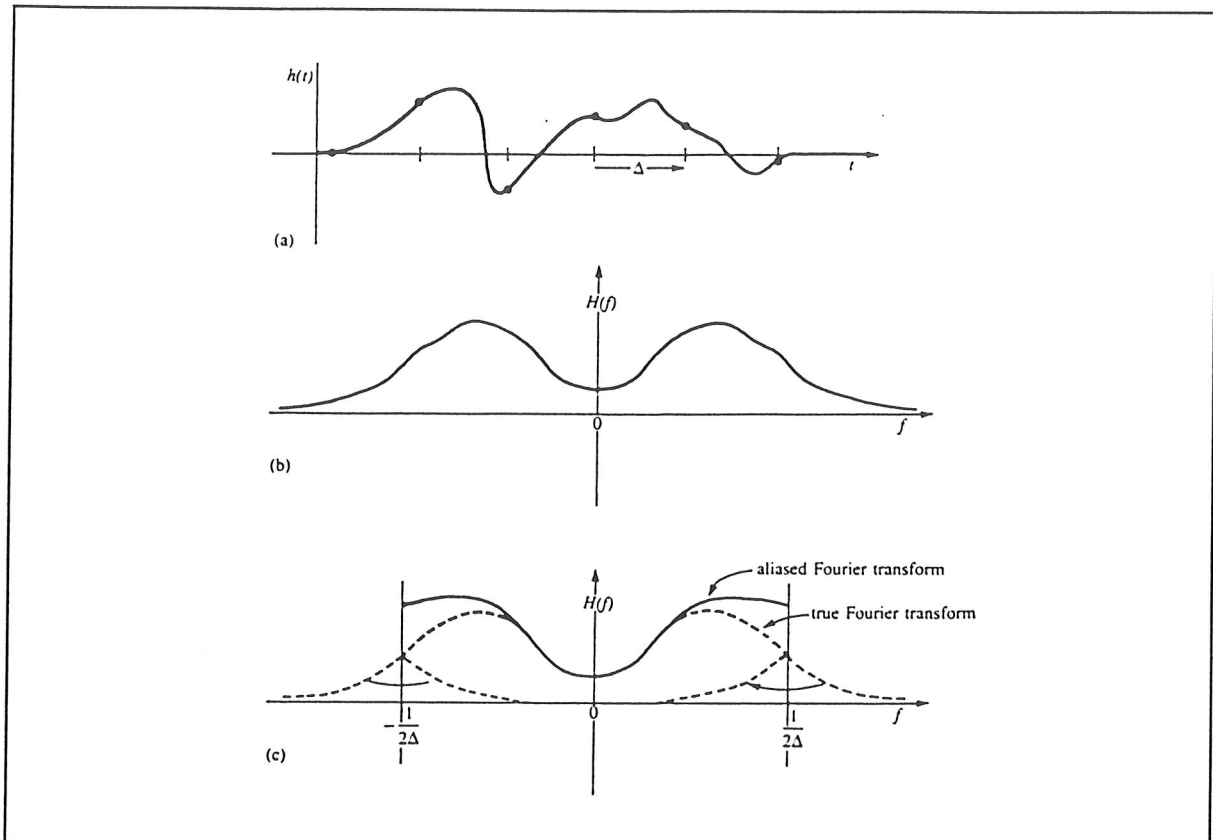


Fig. 9 : Dans le cas d'une fonction à support borné dans le temps (a), nous sommes certains, d'après la relation d'incertitude temps-fréquence, que son support en fréquence ne peut pas être borné et comporte des composantes non nulles à des fréquences aussi grandes que l'on veut. Supposons par exemple que le spectre en fréquence soit approximativement représenté par (b) et échantillonnons maintenant le signal à une fréquence $1/\Delta$. Le théorème d'échantillonnage de Shannon nous avertit que la puissance du signal extérieure à l'intervalle $[-1/2\Delta, 1/2\Delta]$ (\pm la fréquence de Nyquist) va se retrouver repliée sur cet intervalle. Un spectre dont la puissance remonte vers le bord de l'intervalle indique que le filtrage n'a pas été réalisé de façon correcte puisque, par continuité, le bruit « déborde » presque certainement de l'intervalle de Nyquist.

**« Smearing » introduit par la fonction de réponse:
est-ce récupérable ou pas ?**

On peut poser une question similaire au problème de l'échantillonnage et qui concerne la perte ou non d'information subie par un signal passé à travers un système ayant une fonction de réponse $h(t)$ à une impulsion unité envoyée sur la voie d'entrée à $t = 0$. On appelle $h(t)$ la fonction de transfert du système. Là aussi, la réponse s'obtient en considérant le problème dans l'espace des fréquences. Pour un système linéaire en effet, la fonction de sortie d'un système s'exprime très simplement comme le produit de convolution entre le signal d'entrée et la réponse impulsionnelle du système étudié. En effet, en écrivant:

$$g(t) = \int h(\tau) f(t - \tau) d\tau$$

où f est le signal à l'entrée du filtre, et g le signal obtenu en sortie, on vérifie immédiatement, la fonction δ de Dirac étant l'élément neutre vis-à-vis de la

convolution, que l'on obtient bien la fonction h de transfert en sortie si en entrée on fournit une fonction δ .

On peut retrouver l'information « perdue » (et qui ne l'est donc évidemment pas réellement) par le « smearing » du système à condition que la transformée de Fourier de la fonction de réponse impulsionnelle ne s'annule pas et que le signal soit enregistré de $-\infty$ à $+\infty$. La démonstration est élémentaire: en effet, la transformée de Fourier d'un produit de convolution est égal au produit des transformées de Fourier des deux fonctions considérées. Si donc la transformée de Fourier de la réponse impulsionnelle ne s'annule pas, il est possible de calculer son inverse à toute fréquence et de revenir à la fonction initiale, d'où l'affirmation précédente.

Approximation d'une fonction par les n premiers termes de sa série de Fourier

Regardons également comment, pour une fonction comme la fonction créneau, les approximations successives de la décomposition de Fourier (en rajoutant un à un les termes de la décomposition de Fourier), construisent petit à petit la fonction créneau.

Du fait de l'orthogonalité des fonctions sinus et cosinus intervenant dans la décomposition de Fourier, il est facile de montrer que cette décomposition limitée à l'ordre n constitue *la meilleure approximation au sens des moindres carrés* que l'on puisse réaliser sur le sous-espace vectoriel engendré par les n premières fonctions sinus et cosinus de la série de Fourier.

On peut par ailleurs facilement démontrer que la convergence est uniforme en tout point où la fonction que l'on veut représenter par une série de Fourier est continue. On voit néanmoins, au moins dans le cas particulier de la fonction créneau, que les approximations successives par les sommes partielles de Fourier présentent des oscillations (de l'ordre de 10% de la hauteur de discontinuité) au voisinage de ce point. Afin d'étudier ce phénomène, notons d'abord que la fonction « marche d'escalier », ou fonction de Heaviside, et la fonction « dent de scie » constituent des cas génériques d'étude du fait que toute fonction présentant une discontinuité d'amplitude finie peut être représentée comme la somme d'une fonction continue et d'une fonction « marche » ou « dent de scie ».

Historiquement, c'est Michelson qui, en 1898, dénicha ce problème du fait des appareils d'analyse et de synthèse en série de Fourier qu'il avait développés. La possibilité qu'il avait de réaliser des synthèses de séries de Fourier jusqu'à des ordres élevés (plusieurs dizaines de termes) lui avait montré que le phénomène d'oscillation précédent persistait même si l'on augmentait l'ordre d'approximation de façon importante. Il se tourna donc vers le physicien mathématicien J. Gibbs qui proposa une explication au phénomène qui porte aujourd'hui son nom.

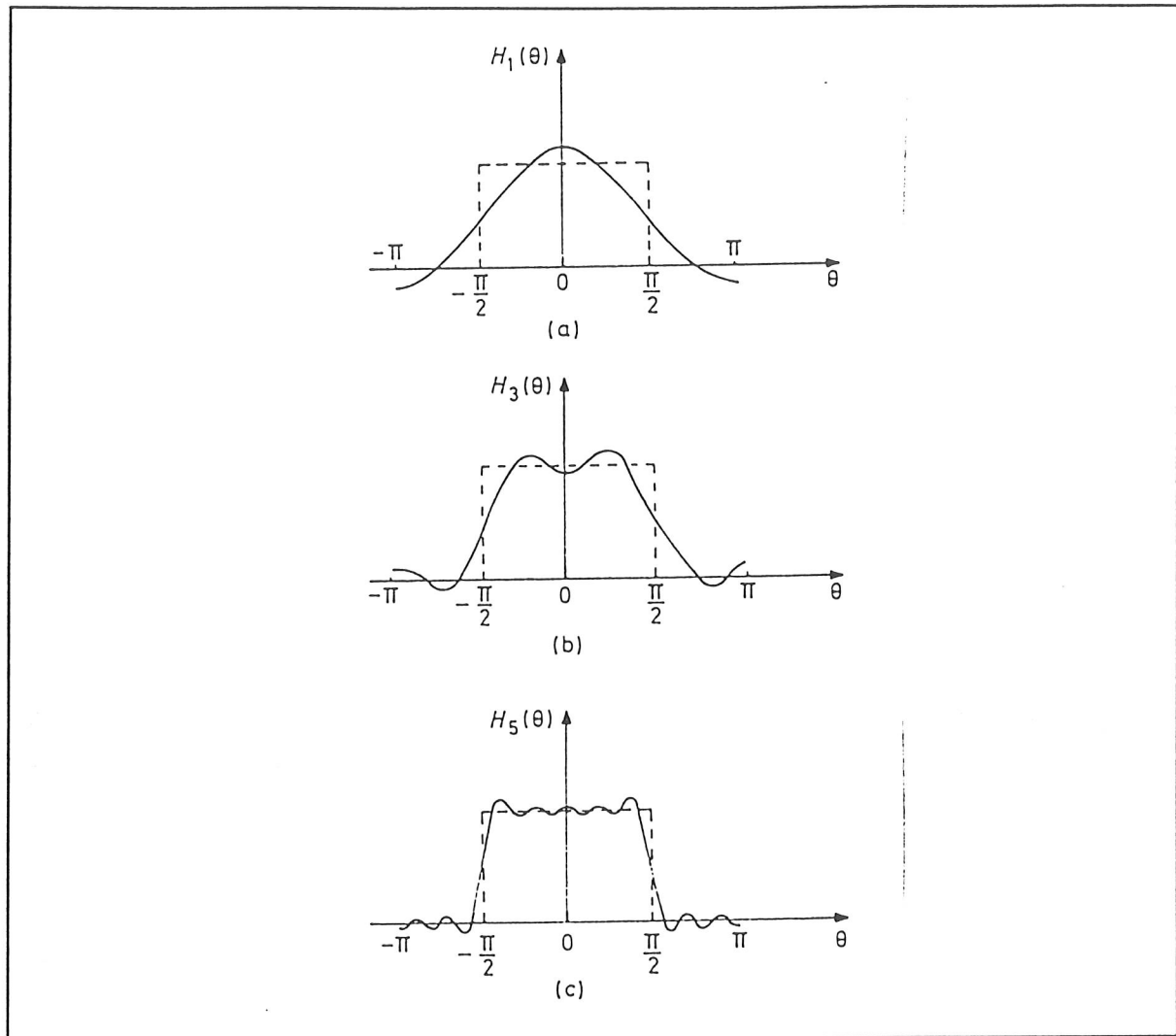


Fig. 10 : On voit sur cette figure les trois premières approximations de Fourier de la fonction porte. Si la première approximation est clairement grossière, on voit que les approximations suivantes présentent un phénomène d'ondulation au voisinage du bord de la porte dont l'amplitude est de l'ordre de 9%.

Le phénomène de Gibbs

Le phénomène de Gibbs illustre le problème de la reconstruction fidèle d'une fonction présentant une (ou plusieurs) discontinuités. Quel que soit l'ordre d'approximation de l'intégrale de Fourier, la forme approchée de la fonction présente toujours des ondulations d'amplitude *fixe* par rapport au saut de la fonction en son point de discontinuité. Lorsqu'on augmente le nombre de termes dans l'approximation, les ondulations qui constituent le phénomène de Gibbs se rapprochent de plus en plus du point de discontinuité, mais sans diminuer d'amplitude (toujours de l'ordre de 9% d'overshoot par rapport à l'amplitude de la discontinuité).

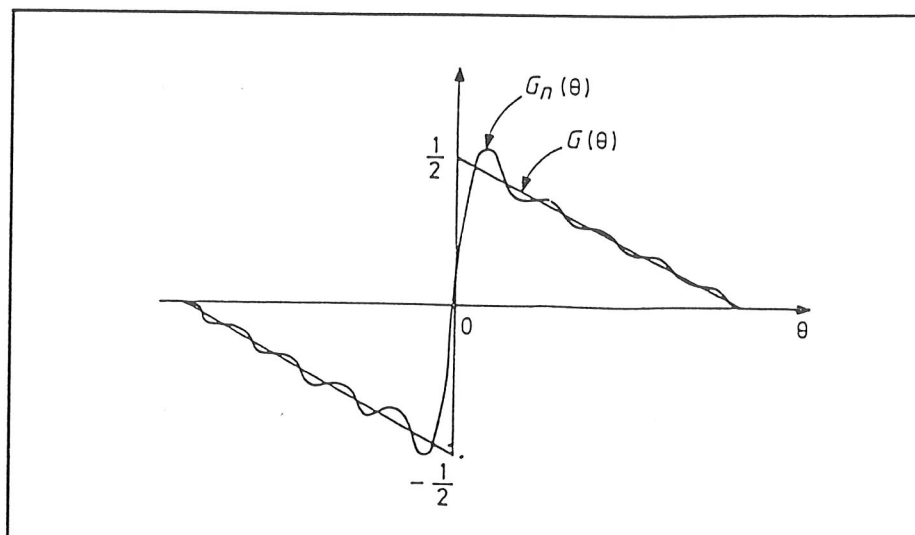


Fig.11 : On visualise ici le phénomène de Gibbs dans le cas de la fonction en dent de scie. Au voisinage du point de discontinuité, les oscillations ont une amplitude qui est toujours de l'ordre de 9% d'overshoot par rapport à l'amplitude de la discontinuité.

Regardons par exemple les approximations successives de la série de Fourier. Le phénomène d'oscillation devient rapidement clairement visible. Ces oscillations convergent-elles : non ! Quel que soit le nombre de termes utilisé dans la synthèse, on trouve toujours des oscillations d'amplitude finie au voisinage de la discontinuité. En fait, l'erreur tend vers une valeur d'oscillation d'environ 9 % (8.949%) au voisinage d'une discontinuité lorsque l'ordre n de l'approximation de la série de Fourier (somme des termes $-n$ à $+n$) tend vers l'infini.

L'interprétation de ce phénomène réside dans l'influence de la troncature. En effet, une somme tronquée de Fourier n'est rien d'autre que la multiplication de la série complète par un fenêtrage rectangulaire dans l'espace des fréquences. En repassant dans l'espace des temps, cette multiplication se traduit par un produit de convolution entre la fonction originale et la transformée de Fourier de la fonction porte qui représente la somme tronquée (Fig. 12). La transformée de Fourier de cette fonction porte, du fait de son ubiquité dans les phénomènes de filtrage, le nom de « kernel de Dirichlet ». Il est facile de démontrer, par sommation directe, que l'expression analytique du « kernel de Dirichlet » est

$$D_n(\theta) = \sum_{k=-n}^n \exp(jk\theta) = \frac{\sin\left(n + \frac{1}{2}\right)\theta}{\sin\frac{\theta}{2}}$$

Cet effet d'oscillation réapparaît dans la construction de tout filtre dès que l'on cherche à maximiser sa raideur en fonction du nombre de termes intervenant dans le filtre. Nous allons voir qu'une façon de s'en débarrasser réside dans le fenêtrage, mais l'on perdra alors en netteté ce que l'on a gagné en suppression d'oscillation.

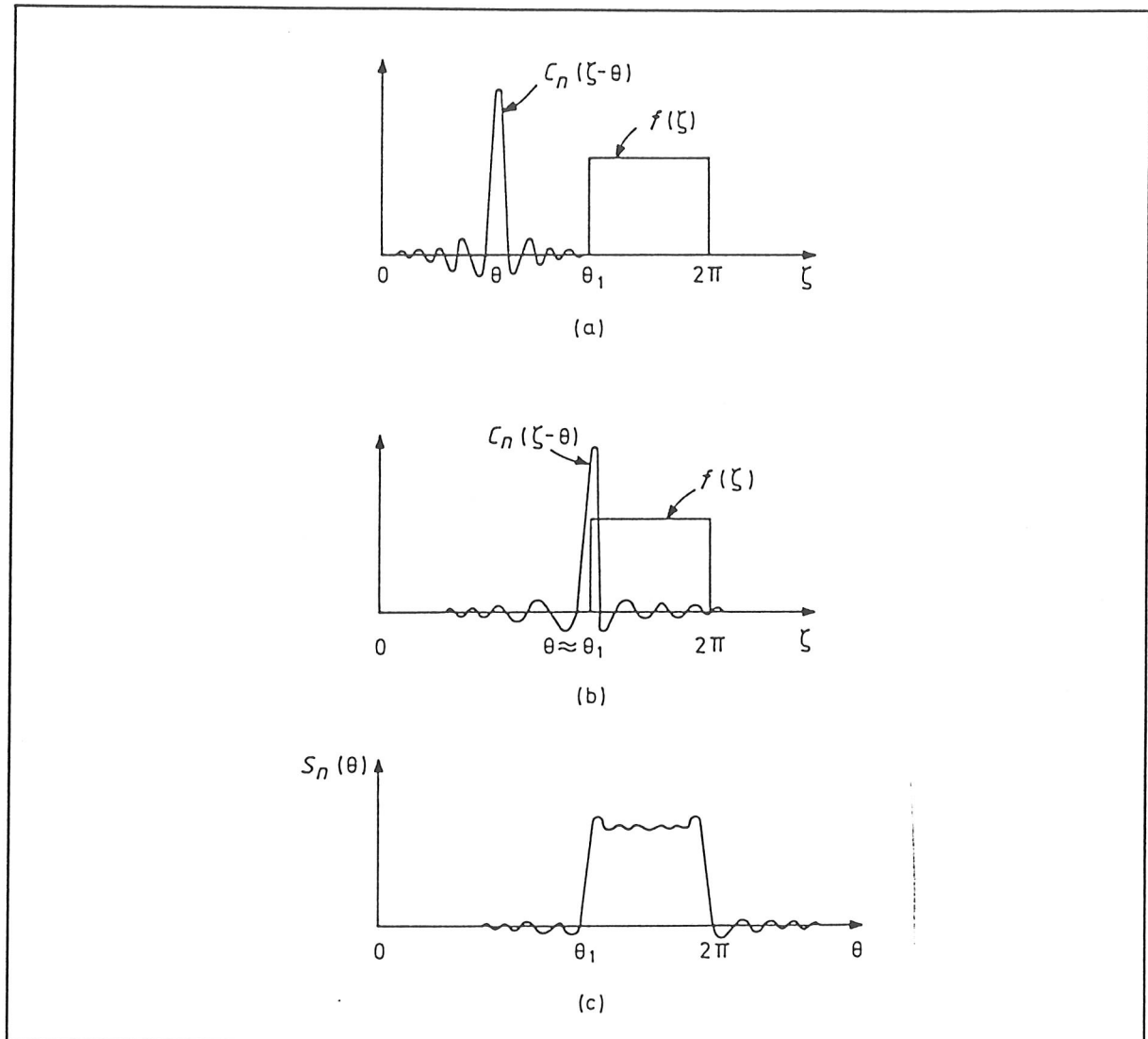


Fig. 12 : Une somme tronquée de Fourier étant équivalente à un fenêtrage rectangulaire dans l'espace des fréquences, on peut comprendre le phénomène de Gibbs dans l'espace des temps au moyen de la convolution par le « kernel de Dirichlet », transformée de Fourier de cette fonction porte.

Notons encore que l'on retrouve là l'outil, auquel on donne ici le nom de kernel de Dirichlet, que l'on avait déjà rencontré dans le cadre du théorème d'échantillonnage où, là aussi, figurait le problème du fenêtrage lorsqu'il s'est agi de supprimer les alias engendrés par l'échantillonnage discret.

V - Fenêtrage

Afin d'atténuer le phénomène de Gibbs, on peut définir des approximations de la fonction non plus en fonction des approximations tronquées de la série de Fourier mais en fonction du « lissage » de ces approximations. Dans le cas de la fenêtre de Fejer, il s'agit de la moyenne arithmétique des n premières approximations, dans le cas de la fenêtre de Lanczos, de leur intégrale

(correctement normalisée). Chacune de ces opérations linéaires correspond à la multiplication de la série de Fourier initiale par une fonction fenêtre.

Le fenêtrage s'utilise donc lorsque l'on veut atténuer les effets d'une discontinuité d'une fonction. Prenons pour exemple le cas d'une sinusoïde enregistrée sur un nombre non-entier de périodes dont on s'attendrait naïvement à ce que le spectre en fréquence soit une fonction δ de Dirac à la fréquence de la sinusoïde. Or la transformée de Fourier de cette fonction, dont il ne faut pas oublier que tout se passe comme si elle se répétait indéfiniment, « bave » sur les fréquences voisines (« spectral leakage ») sauf si la sinusoïde se referme sur elle-même. Afin de lutter contre ce phénomène parasite, on peut réduire le phénomène de « fuite spectrale » en multipliant la fonction par une fonction fenêtre qui va intégrer en un sens ou l'autre ces oscillations. Comme on n'a rien sans rien, la disparition des oscillations va avoir pour corrolaire une dégradation du pouvoir de discrimination en fréquence (Fig. 13).

Les principales fenêtres utilisées

Quelques-unes des principales fonctions de fenêtrage utilisées sont:

- la fonction de Parzen, (les fonctions de Fejer ou Bartlett, également en forme de triangle, sont pratiquement identiques) dont les coefficients sont donnés par l'expression

$$w_j = 1 - \left| \frac{j - 1/2(N-1)}{\frac{1}{2}(N-1)} \right|$$

- la fenêtre de Hanning ou von Hann sont également pratiquement identiques:

$$w_j = \frac{1}{2} \left| 1 - \cos\left(\frac{2\pi j}{N-1}\right) \right|$$

- la fenêtre de Welch (parabolique)

$$w_j = 1 - \left(\frac{j - \frac{1}{2}(N-1)}{\frac{1}{2}(N-1)} \right)^2$$

- la fenêtre de Lanczos

$$w_j = \frac{\sin(2n\pi / N)}{(2n\pi / N)}$$

Il est assez étonnant de constater à quel point un grand nombre de gens ont réussi à attacher leur nom à une fenêtre ou à une autre en théorie du signal. Les différences entre les unes et les autres, sauf peut-être pour des spécialistes ou des applications très spécifiques, sont le plus souvent inobservables. En pratique, on utilisera, pour des raisons de commodité de calcul, l'une ou l'autre des fonctions

triangle ou parabolique (Parzen ou Wlech) qui suffiront amplement pour la plupart des applications. La plupart des « packages » logiciels de traitement de signal proposent d'ailleurs des fonctions permettant de générer automatiquement ces fenêtres.

Window Type	Width of Main Lobe		Approximate Amount by Which the First Side Lobe Is Down Compared with the Main Lobe (dB)
	ω	ωT	
Rectangular	$\frac{2\omega_s}{M}$	$\frac{4\pi}{M}$	13
Bartlett	$\frac{4\omega_s}{M}$	$\frac{8\pi}{M}$	26
von Hann	$\frac{4\omega_s}{M}$	$\frac{8\pi}{M}$	32
Hamming	$\frac{4\omega_s}{M}$	$\frac{8\pi}{M}$	40
Blackman	$\frac{6\omega_s}{M}$	$\frac{12\pi}{M}$	60

Fig. 13 : Comparaison des performances de quelques fonctions de fenêtrage. On remarquera la compétition entre la décroissance des ondulations des lobes secondaires et celle de la largeur de fréquence associée à la fenêtre.

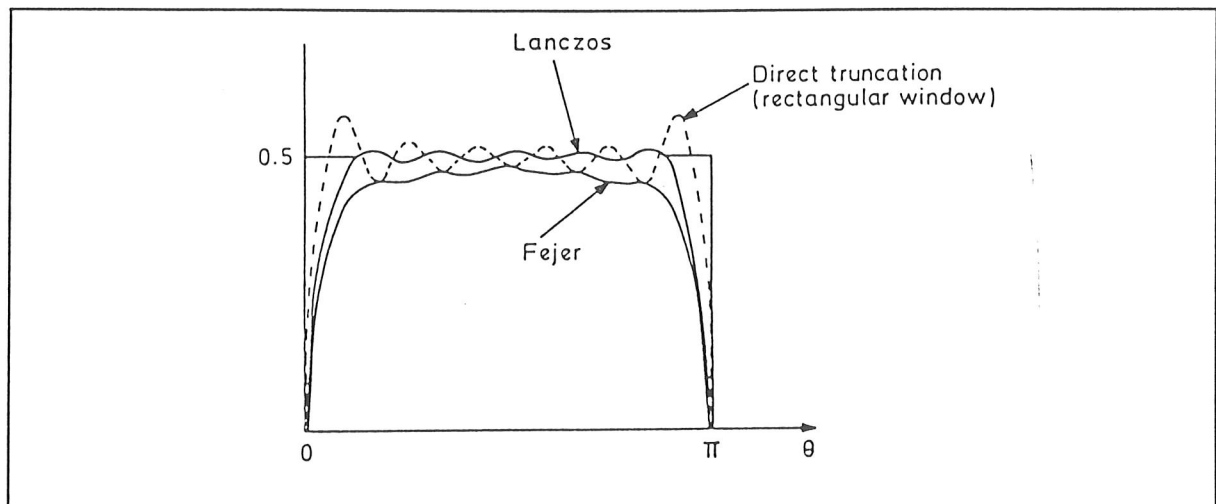


Fig. 14 : Influence des fenêtres de Lanczos et Fejer sur le phénomène de Gibbs.

Inégalité de Schwartz

Le caractère défini positif de toute norme permet de démontrer l'inégalité de Schwartz dans le cas particulier de la norme définie sur l'ensemble des fonctions de carré intégrable

$$\left(\int_{-\infty}^{\infty} f^2(t) dt \right) \left(\int_{-\infty}^{\infty} g^2(t) dt \right) \geq \left(\int_{-\infty}^{\infty} f(t)g(t) dt \right)^2$$

En effet, quelque soit le nombre α , on a

$$\left(\int_{-\infty}^{\infty} [f(t) + \alpha g(t)]^2 dt \right) = \int_{-\infty}^{\infty} f^2(t) dt + 2\alpha \int_{-\infty}^{\infty} f(t)g(t) dt + \alpha^2 \int_{-\infty}^{\infty} g^2(t) dt \geq 0$$

En considérant l'expression précédente comme une équation du second degré en α , le fait que cette expression soit toujours positive impose sur le déterminant de cette équation l'importante inégalité de Schwartz ci-dessus.

Relation « d'incertitude » :

Considérons maintenant l'association d'une fonction exprimée dans l'espace des temps et sa transformée de Fourier dans l'espace des fréquences

$$f(t) \leftrightarrow F(\omega)$$

Nous allons démontrer qu'il existe une relation entre la largeur du support de la fonction $f(t)$ et celle de sa transformée de Fourier $F(\omega)$. Néanmoins, cette relation « d'incertitude », qui est analogue à la relation d'Heisenberg pour un couple de variables conjuguées

$$\Delta t \Delta \omega \geq 1/2$$

est vérifiée dans la mesure où l'on définit la largeur d'une distribution à partir de son moment centré d'ordre 2 ou écart quadratique moyen, soit

$$(\Delta t)^2 = \frac{\int_{-\infty}^{\infty} t^2 f^2(t) dt}{\int_{-\infty}^{\infty} f^2(t) dt}$$

et, de même pour $F(\omega)$

$$(\Delta \omega)^2 = \frac{\int_{-\infty}^{\infty} \omega^2 |F(\omega)|^2 dt}{\int_{-\infty}^{\infty} |F(\omega)|^2 dt}$$

Maintenant, l'expression de $(\Delta \omega)^2$ peut se réécrire

$$(\Delta \omega)^2 = \frac{\int_{-\infty}^{\infty} |j\omega F(\omega)|^2 dt}{\int_{-\infty}^{\infty} |F(\omega)|^2 dt}$$

et l'on reconnaît au numérateur la norme au carré de la transformée de Fourier de la dérivée $f'(t)$ de $f(t)$.

En utilisant maintenant l'identité de Parseval sur l'égalité des normes d'une fonction et de sa transformée de Fourier, on obtient

$$(\Delta t \Delta \omega)^2 = \frac{\left(\int_{-\infty}^{\infty} t^2 f^2(t) dt \right) \left(\int_{-\infty}^{\infty} [f'(t)]^2 dt \right)}{\left[\int_{-\infty}^{\infty} |f(t)|^2 dt \right]^2}$$

En appliquant l'inégalité de Schwartz au numérateur, on obtient

$$\left(\int_{-\infty}^{\infty} t^2 f^2(t) dt \right) \left(\int_{-\infty}^{\infty} [f'(t)]^2 dt \right) \geq \left(\int_{-\infty}^{\infty} t f'(t) f(t) dt \right)^2$$

ou, en intégrant par parties

$$\geq \left(\frac{1}{2} t f^2(t) \Big|_{-\infty}^{\infty} - \frac{1}{2} \int_{-\infty}^{\infty} f^2(t) dt \right)^2$$

Supposant maintenant que $f(t)$ tend « suffisamment vite vers 0 », on obtient

$$(\Delta\omega \Delta t)^2 \geq \frac{1}{4} \text{ ou } \Delta\omega \Delta t \geq \frac{1}{2}$$

Cette relation implique qu'il n'est *pas possible* d'avoir simultanément une fonction et sa transformée de Fourier de supports aussi étroits que l'on veut. Leur largeur, au sens défini plus haut, vérifie nécessairement la relation « d'incertitude » ci-dessus.

On peut démontrer un résultat plus fort en constatant que si une fonction possède un support borné, c'est-à-dire qu'il existe un intervalle en dehors duquel elle est nulle, sa transformée de Fourier est analytique et ne peut donc s'annuler sur un segment faute de quoi, du fait que l'on peut déterminer une fonction analytique à partir de ses valeurs sur un segment, elle serait nécessairement nulle partout. Donc:

Si une fonction a un support fini, sa transformée de Fourier ne peut être à support limité en fréquence.

Du fait de la symétrie de la transformée de Fourier et de sa transformée inverse, la relation inverse est également vérifiée.

IV - Transformée de Laplace et transformation z

Expression et propriétés de la transformation de Laplace

La transformation de Laplace, généralisant la transformée de Fourier $G(f)$ d'une fonction $g(t)$, se définit comme

$$G(s) = \int_{0^-}^{\infty} g(t) \exp(-st) dt$$

définie maintenant sur le plan complexe où s prend ses valeurs.

Notez que cette transformation se définit en général par l'intégrale de 0 à l'infini et non plus de façon symétrique sur temps (et fréquences) négatives. La fonction de Laplace trouve ses applications dans le calcul opérationnel et dans l'étude des réseaux.

Nous n'étudierons pas plus avant les propriétés de la transformée de Laplace mais citons néanmoins quelques propriétés qui rappellent et généralisent celles

de la transformée de Fourier, notant partout $F(s)$ la transformée de Laplace de la fonction $f(t)$

$f'(t)$	$sF(s) - f(0-)$
$f^{(n)}(t) = \frac{d^n f(t)}{dt^n}$	$s^n F(s) - \sum_{k=1}^n s^{n-k} f^{(k-1)}(0-)$
$\int_{0-}^t f(\tau) d\tau$	$\frac{F(s)}{s}$
$(-t)^n f(t)$	$\frac{d^n}{ds^n} F(s)$
$\exp(\alpha t) f(t)$	$F(s - \alpha)$

Expression et propriétés de la transformation-z

A toute suite $\{f(n)\}$, on peut associer sa transformée en z

$$f(n) \rightarrow \sum_{n=-\infty}^{\infty} f(n\Delta) z^{-n}$$

où z est une variable complexe (et comme d'habitude Δ est le temps d'échantillonnage).

Sa relation avec la transformée de Laplace apparaît si l'on effectue le changement de variable

$$z^{-1} \equiv \exp(-Ts)$$

L'intérêt de la transformation z apparaît dans la recherche des solutions propres d'un filtre linéaire à coefficients constants (voir plus loin, le chapitre concernant les filtres FIR et IIR). On cherchera alors des solutions en série géométrique et c'est exactement la forme de la transformation z .

En raison de sa relation avec les transformée de Fourier, on démontre facilement les propriétés suivantes:

- linéarité
- translation

on vérifie facilement qu'une translation d'un facteur constant m de tous les indices conduit à la nouvelle transformation z

$$f(n) \rightarrow F(z) = \sum_{n=-\infty}^{\infty} f(n\Delta) z^{-n} \Leftrightarrow f(n-m) \rightarrow \sum_{n=-\infty}^{\infty} f[(n-m)\Delta] z^{-(n-m)} = z^{-m} F(z)$$

- convolution: comme dans le cas de la série de Fourier, on démontre facilement que la transformée z du produit de convolution de deux fonctions est égale au produit de la transformée de Fourier de chacune des deux fonctions.

- convergence: signalons simplement ici que l'étude du domaine de définition de convergence de la série entière associée à une suite, et des fractions rationnelles qui seront définies dans la suite, est un moyen d'étude essentiel de la stabilité des filtres.

VI - Filtrage numérique

Chacun connaît les opérations de filtrage et leur nécessité: la « ronflette » du 50 Hz sur une installation hi-fi mal branchée, les moirés sur l'image TV du costume à petites rayures de cet homme politique, ce tuner peu sélectif qui vous permet d'écouter simultanément « Radio Notre-Dame » et « Fun Radio », fournissent des exemples de l'importance du filtrage dans les aspects les plus courants de notre vie quotidienne. Là aussi, le numérique apporte des performances, une flexibilité et une approche de plus en plus remarquables.

Définitions

Nous nous plaçons dans le cadre des systèmes linéaires, de réponse invariante par translation dans le temps et, en général, causaux définis plus haut. S'il est bien sûr possible de réaliser avec une très bonne approximation des filtres de réponse spectrale à peu près quelconque au moyen de composants électroniques discrets, les méthodes numériques de filtrage sont devenues de plus en plus populaires au fur et à mesure que s'accroissait la puissance des DSP (Digital Signal Processors) et une grande partie des applications de filtrage, notamment celles qui exigent flexibilité et programmabilité, mettent maintenant très souvent en jeu un filtrage numérique.

Les principaux *filtres idéaux* sont le filtre passe-bas, le filtre passe-haut, le filtre passe-bande et le filtre stoppe-bande (fig. 15). Ces filtres, ayant des fronts infiniment raides, sont cependant des idéalizations et ne sont donc pas réalisables en pratique. On appelle « filtre idéal » un filtre pour lequel la transmission en amplitude est constante (que l'on prend égale à 1 pour simplifier) sur certaines bandes de fréquence et nulle ailleurs. On voit que les filtres ci-dessus, représentés sur la figure 15, répondent à cette définition. Par ailleurs, si l'on écrit la fonction de transfert d'un filtre idéal sous la forme

$$H(\omega) = |H(\omega)| \exp(j\psi(\omega))$$

alors un filtre idéal est un filtre qui vérifie, par définition:

$$\psi(\omega) = -k\omega, \text{ o } k \text{ est une constante positive}$$

Afin de voir pour quelle raison on appelle un tel comportement « idéal » pour la fonction de transfert d'un filtre, remarquons qu'un signal d'entrée, décrit par sa transformée de Fourier $F(\omega)$, va se trouver transformé en sortie en

$$G(\omega) = \exp(-jk\omega) F(\omega)$$

En prenant la transformée de Fourier inverse de $G(\omega)$, la table de propriétés élémentaires de la transformée de Fourier donnée plus haut fournit la relation

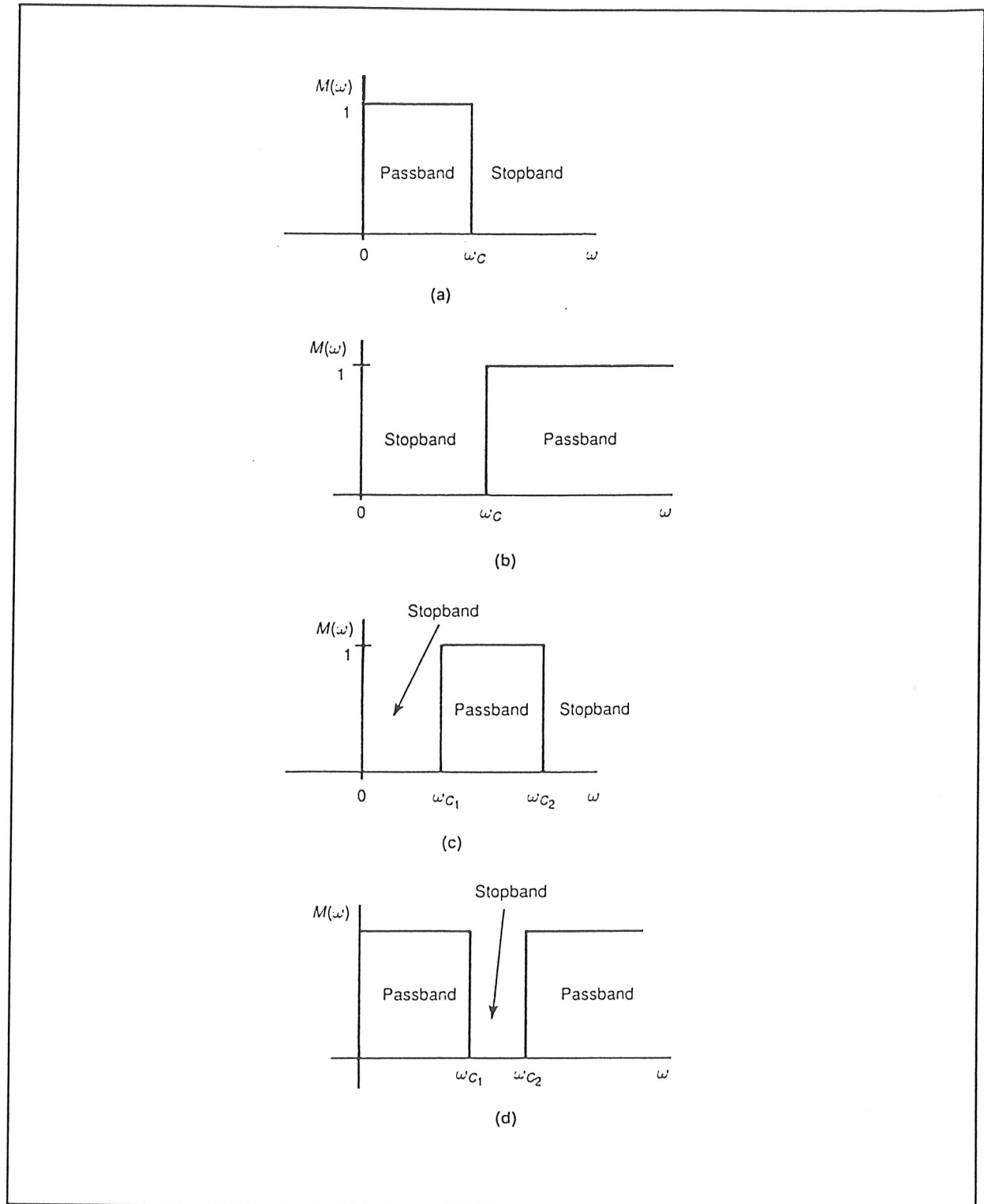


Fig. 15 : Principaux types de filtres (idéalisés et non réalisables sans approximation):

- filtre passe-bas
- filtre passe-haut
- filtre passe-bande
- filtre stoppe-bande

$$g(t) = f(t-k)$$

Si cette condition de phase linéaire est vérifiée, on voit que le signal *n'est pas déformé en sortie*, et cela quelle que soit la forme du signal d'entrée, mais seulement retardé d'un temps k . Pour des raisons de causalité, on a évidemment demandé que la constante k soit positive.

Filtrer dans le domaine des fréquences

L'approche la plus simple en termes de filtrage consiste à aborder la question dans l'espace des fréquences. En effet, c'est bien là en général que les caractéristiques que l'on cherche à imposer à un filtre sont les plus claires. Dans cet espace, il suffit de dessiner le filtre, de trouver une expression numérique pour l'expression du filtre qui soit satisfaisante, de multiplier la transformée de Fourier du signal par cette expression du filtre et de repasser au signal filtré en effectuant une transformée de Fourier inverse.

Cette approche est tellement simple que l'on ne voit pas très bien pourquoi l'on s'ennuierait à écrire des ouvrages entiers sur les techniques de construction de filtres numériques. En fait, elle est évidemment un peu simpliste. Le premier problème tient au fait que cette approche n'est intrinsèquement pas « temps réel ». En effet, afin de passer dans l'espace des fréquences, il faut avoir enregistré le signal et effectué une transformée de Fourier, une opération fort peu locale, les variables temps et fréquence étant conjuguées. En outre, si l'on taille à la serpe dans la forme du filtre, c'est-à-dire si l'on cherche à obtenir un filtre avec des fronts très raides afin de couper très efficacement dans le domaine des fréquences, le filtre résultant demande un très grand nombre de coefficients et possède en outre la propriété, dans l'espace des temps, d'avoir une réponse impulsionnelle qui met en jeu un grand nombre d'oscillations. Les deux aspects sont liés et ces oscillations sont une illustration du phénomène de Gibbs dont nous avons parlé plus haut.

Même en définissant un filtre dans l'espace des fréquences, un certain nombre de traitements prophylactiques doivent être appliqués aux données:

- ne pas oublier les fréquences négatives dans le design du filtre. Respecter la relation de symétrie $H(-f) = H(f)^*$ si l'on traite des signaux à valeurs réels.
- prendre garde aux discontinuités fictives qui sont introduites dans toute transformée de Fourier où l'on voit une fluctuation de la ligne de base. Commencer donc en général par ajuster une droite qui passe par le premier et par le dernier point et soustraire cette dérive aux données de façon à limiter l'influence des basses fréquences et du fait que les données ne « bouclent » pas
- si vous voulez garder un caractère approximativement « temps réel » à vos données, prenez garde aux caractéristiques trop raides. En effet, tout filtre est d'autant plus acausal qu'il opère brutalement en fréquence, s'il est fidèle, c'est-à-dire qu'il met en jeu un grand nombre d'échantillons avant le temps 0. Evidemment, comme il n'est pas question de supposer que vous allez savoir

remonter dans le passé, une façon de contourner le problème est d'accepter que votre filtre ait un retard d'autant plus important qu'il comporte de coefficients à temps négatif, ceci afin de rendre le filtre réalisable. Mais il vous faudra couper vos données en paquets d'autant plus longs que vous demanderez un caractère raide à votre filtre.

- comme d'habitude, commencer par *préfiltrer de façon analogique* votre signal si nécessaire car un filtrage numérique ne pourra jamais compenser les erreurs associées à un préfiltrage qui ne correspond pas au critère de Nyquist.

Filtres FIR et IIR

Si nous définissons comme précédemment un système comme une transformation entre l'entrée $f(n)$ résultant en la sortie $g(n)$, nous pouvons, sur le sous-ensemble des systèmes linéaires, définir deux grandes classes de systèmes

- **FIR (Finite-duration Impulse Response)**

$$g(n) = \sum_0^N a_r f(n-r)$$

La dénomination FIR signifie que si le signal d'entrée devient nul, le signal de sortie finira également par s'annuler après un temps (nombre de pas du filtre) fini. Une condition nécessaire et suffisante est que la somme des valeurs absolues des coefficients du filtre soit bornée

$$\sum_0^N |a_r| < +\infty$$

- **IIR (Infinite-duration Impulse Response)**

Il s'agira en général d'un filtre de la forme:

$$g(n) = \sum_0^M a_r f(n-r) + \sum_1^N b_r g(n-r)$$

Strictement parlant, comme son nom l'indique, un filtre IIR a seulement pour propriété que si le signal d'entrée devient nul, le signal de sortie pourra parfaitement rester non nul pendant un temps (nombre de pas du filtre) infini. Même si un tel comportement peut être obtenu avec un nombre infini de coefficients non nuls a_i dans une expression du type

$$g(n) = \sum_0^N a_r f(n-r), \text{ avec } \sum_0^N |a_r| = +\infty$$

on entend en général par filtre IIR un filtre à nombre *fini* de coefficients et *récurif*, comme dans l'expression ci-dessus, puisqu'il fait intervenir les valeurs des sorties précédentes du filtres et pas seulement les valeurs du signal d'entrée. De ce fait, on n'a pas de garantie que le signal de sortie tende vers zéro même si le signal d'entrée devient nul.

La z-transformation permet d'étudier les propriétés de ces deux catégories de filtres. La raison en est que le caractère linéaire du système amène à rechercher

des solutions en série géométrique, pour lesquelles la z -transformation est à l'évidence adaptée, afin de construire une base pour les solutions.

Dans le cas FIR, la fonction z à étudier est

$$H(z) = \sum_0^N a_r z^{-r}$$

alors que pour le système IIR décrit plus haut, la fonction z devient

$$H(z) = \frac{\sum_0^N a_r z^{-r}}{1 + \sum_0^M b_r z^{-r}}$$

La première remarque concerne le fait que les filtres FIR sont stables de façon inhérente. Une façon un peu mathématique de le démontrer est de remarquer que la fonction z associée a tous ses pôles en $z = 0$, qui se trouve à l'intérieur du cercle unité. Dans le cas d'un filtre IIR, au contraire, on a affaire à une fraction rationnelle dont la position des pôles reste à déterminer.

Pour quelle raison alors aller risquer l'instabilité d'un filtre en utilisant une forme récursive ? D'autant que l'instabilité sur un système discret à nombres de bits fini peut conduire à des comportements oscillatoires ou de saturation qui ne ressemblent pas à la divergence vers l'infini du système continu correspondant et donc qu'il ne sera donc pas toujours évident de détecter. La raison en est qu'il est en général possible de réaliser un filtre IIR, pour des caractéristiques données, en utilisant un nombre de coefficients *plus faible* que si l'on utilise la forme polynomiale du filtre FIR. En particulier, les filtres mettant en jeu des formes de variation rapide en fonction de la fréquence (si l'on se rapproche des pôles) nécessitent pour un filtre IIR un nombre nettement moins important de coefficients qu'une forme FIR. Si la puissance de calcul est limitée, et pour des applications « temps réel », il peut donc être intéressant de faire appel à une forme IIR plutôt qu'à un filtre FIR.

Par ailleurs, les contraintes en termes de (non-)déformation vont également intervenir dans le choix d'un filtre FIR ou IIR. Si l'on veut une très grande linéarité de la phase (associée en général à une conservation de la forme du signal, si la variation de l'amplitude en fonction de la fréquence n'est pas trop rapide), il faudra a priori utiliser une forme FIR. Si au contraire ce sont les considérations sur l'amplitude du signal qui l'emportent, on pourra choisir, à nouveau principalement pour des raisons de puissance de calcul nécessaire autant que pour la possibilité de reproduire des formes tourmentées, une forme IIR.

Une recette simple afin de construire un filtre FIR de réponse spectrale (à peu près) quelconque

Supposons que l'on s'intéresse à un filtre défini par la relation entre la sortie y_i et les signaux d'entrée x_i

$$y_n = \sum_{k=0}^M c_k x_{n-k}$$

mettant en jeu $M+1$ coefficients.

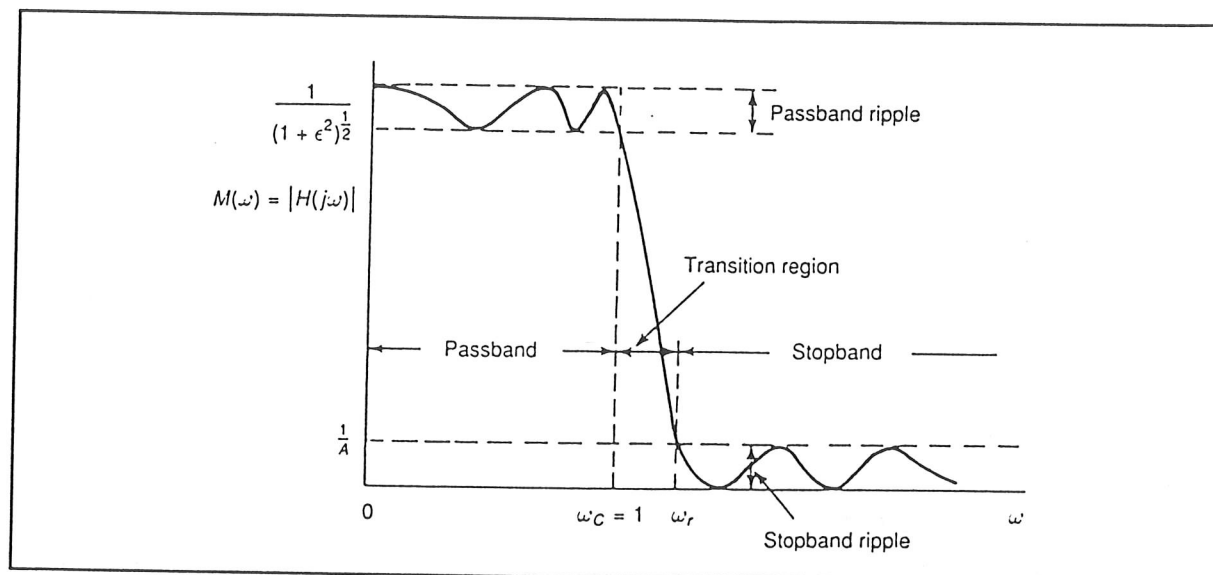


Fig. 16 : Terminologie des caractéristiques des filtres

- ω_c : fréquence de transition du filtre
- ondulation de passe-bande (« passband ripple »): amplitude maximum de l'ondulation du filtre dans la bande passante du filtre
- ondulation de bande atténuée (« stopband ripple »): amplitude maximum de l'ondulation du filtre dans la bande atténuée du filtre
- les paramètres ϵ et A traduisent les deux paramètres précédents grâce aux expressions

$$\frac{1}{(1 + \epsilon^2)^{1/2}} \text{ et } \frac{1}{A}$$

- région de transition du filtre: intervalle de fréquence délimité par la plus haute fréquence de la bande passante et la plus petite fréquence de la bande atténuée où le filtre atteint les valeurs d'ondulation précédentes.

Une recette simple afin de construire un filtre FIR de réponse spectrale (à peu près) quelconque

Supposons que l'on s'intéresse à un filtre défini par la relation entre la sortie y_i et les signaux d'entrée x_i

$$y_n = \sum_{k=0}^M c_k x_{n-k}$$

mettant en jeu $M+1$ coefficients.

Associé à ce filtre, on a la réponse spectrale

$$H(f) = \sum_{k=0}^M c_k e^{-2j\pi k f \Delta}$$

Comme on le sait, le problème est inversible puisqu'il s'agit de l'expression d'une transformée de Fourier discrète à $(M+1)$ coefficients. Ce qui nous intéresse en général est de fixer $H(f)$ afin d'obtenir les coefficients c_k , mais si l'on opère de cette façon, comme pour une fonction polynomiale que l'on contraint à passer par un grand nombre de points, la fonction résultante aura en général tendance à osciller très fortement entre les valeurs sur lesquelles on les a fixés, si bien que notre approche n'est pas d'une grande utilité.

Afin de progresser, on peut suivre l'approche suivante:

- définir avec une grande précision (celle de la fréquence de Nyquist adaptée à votre système, par exemple) la forme du filtre $H(f)$
- calculer la transformée de Fourier de ce filtre, qui aura donc en général un très grand nombre de coefficients et ne sera pas directement utilisable facilement
- maintenant, tronquer l'expression du filtre en ne gardant que les fréquences les plus basses. Mais *attention* ! afin de tronquer intelligemment, il ne faut pas oublier que l'on doit toujours agir de la même façon sur les fréquences négatives et sur les fréquences positives. Dans la troncature, on gardera donc les n premiers et les $n-1$ derniers coefficients (seulement $n - 1$ car le continu est à part et décrit par le premier coefficient) de la transformée de Fourier de la fonction de transfert $H(f)$.
- maintenant, repasser dans l'espace des fréquences afin de voir si votre approximation vous satisfait. Si la forme ne vous convient pas, essayez d'augmenter le nombre de coefficients, ou modifiez-les au moyen de l'une des fonctions de fenêtrage définies plus haut, cela jusqu'à ce que vous arriviez à un filtre satisfaisant.

Approximation du filtre passe-bas idéal

Dans l'approche du dessin d'un filtre le plus idéal possible passe-bas, il nous faut définir quelles caractéristiques (raideur de la variation en amplitude au détriment de la linéarité de la phase, ou au contraire très grande linéarité en phase demandée, absence d'ondulations dans la bande passante, ou dans la bande d'arrêt, ... (Ce sous-chapitre n'est pas achevé. Désolé...))

Les filtres les plus courants; comportement en phase et en amplitude (garder ou pas la forme du signal d'entrée)

- **Butterworth**

(filtre " plat ", pente et $(n-1)$ premières dérivées nulles à l'origine) a pour dépendance en fréquence

$$|F_n(\omega)|^2 = \frac{1}{1 + \left(\frac{\omega}{\omega_c}\right)^{2n}}$$

Comme l'indique sa forme, la fonction est extrêmement plate autour de $w = 0$, et décroît régulièrement sans ondulations. C'est un filtre orienté amplitude. Comme on le voit également, il est de forme IIR (fraction rationnelle).

• **Chebyshev**

$$|H(\omega)|^2 = \frac{1}{1 + \varepsilon^2 T_n^2(\omega)}$$

où T_n est un polynôme de Chebyshev d'ordre n .

Ces filtres présentent des oscillations dans la bande passante puis une décroissance régulière vers la bande stoppante.

• **Elliptiques**

Comme leur nom l'indique, ces filtres font entrer en jeu des fonctions elliptiques. Ils présentent des oscillations à la fois dans la bande passante et dans la bande d'arrêt (dont on peut ajuster le niveau, comme dans le cas précédent).

Plus généralement, il existe un algorithme, celui de maintenant disponible dans un grand nombre de « packages » logiciels de traitement du signal qui propose une procédure d'itération afin de trouver une bonne estimation des meilleurs coefficients permettant de répondre aux contraintes de:

- ω_c : fréquence de transition du filtre
- ondulation de passe-bande (« passband ripple »): amplitude maximum de l'ondulation du filtre dans la bande passante du filtre
- ondulation de bande atténuée (« stopband ripple »): amplitude maximum de l'ondulation du filtre dans la bande atténuée du filtre
- amplitude des paramètres ε et A d'amplitude d'oscillation dans la bande passante et stoppante (voir figure 17).

De ce point de vue, même si cette procédure est coûteuse en temps calcul, elle ne doit être faite qu'une fois dans la construction du filtre. Elle fait office de « boîte noire » mais permet de résoudre l'essentiel des problèmes de recherche des coefficients de filtres.

Comment passer d'un filtre passe-bas à un filtre passe-haut, à un filtre passe-bande, à un filtre stop-bande ?

Une fois que l'on a défini une approximation répondant aux critères demandés d'un filtre passe-bas, il est facile d'obtenir, par un simple changement de variable, les formes associées aux filtres passe-haut, passe-bande et stop-bande.

Les changements de variable nécessaires afin de passer du filtre passe-bas élémentaire de réponse spectrale:

$$H(\omega) = 1 \quad \text{si } \omega \in [-1, 1]$$

$$H(\omega) = 0 \quad \text{autrement}$$

sont ainsi:

- filtre passe-bas de fréquence de coupure ω_0

$$\omega \rightarrow \frac{\omega}{\omega_0}$$

- filtre passe-haut de fréquence de coupure ω_0

$$\omega \rightarrow \frac{\omega_0}{\omega}$$

- filtre passe-bande de fréquences de coupures ω_1 et ω_2

$$\omega \rightarrow \frac{1}{\omega_2 - \omega_1} \left(\omega + \frac{\omega_2 \omega_1}{\omega} \right)$$

- filtre stoppe-bande de fréquences de coupures ω_1 et ω_2

$$\omega \rightarrow \frac{\omega_2 - \omega_1}{\omega + \frac{\omega_2 \omega_1}{\omega}}$$

Considérations sur la stabilité des filtres digitaux

Influence de la précision sur le filtrage ; erreurs d'arrondi. Oscillations : dépassement, cycles limites.

Nombre fini de bits (surtout 8 à 12 bits, à moindre titre pour 16 bits, précisions typiques dans les ADC et DAC) a des conséquences qui peuvent être désastreuses sur les filtres par exemple : déplacement et petite zone de flou sur les pôles dans le plan complexe et instabilité imprévue. Dans le même ordre d'idées, erreurs d'arrondi dans les multiplications et plus grave, additions et soustractions.

Instabilité due aux erreurs ou effets de taille finie dans les filtres résulte d'une erreur dans la position des pôles donnée par l'expression 11.29 (p.412) du Baher. Commentaire sur la qualité des filtres suivant leur implémentation en cascade (plus stable numériquement car chaque étape à un numérateur et un dénominateur de degré au plus 2) ou parallèle (tous les coefficients d'un coup, plus efficace en mémoire, forme canonique, mais peut être plus instable numériquement).

VIII - Ajustement dans l'espace des fréquences et Filtrage optimal (Wiener)

Historiquement, c'est Kolmogorov et Wiener qui, au cours des années 40, ont établi le formalisme de filtrage optimal dans le cas des processus stationnaires (à la fois en ce qui concerne le bruit et le signal). Le critère d'optimalité qu'ils ont utilisés repose sur l'estimation au sens des moindres carrés et définit ce que l'on appelle aujourd'hui le *filtre de Wiener*.

Le formalisme développé par Kolmogorov et Wiener ne s'applique cependant pas aux phénomènes non-stationnaires (pour le bruit et/ou le signal) et il fallut attendre les années 60 pour que Kalman développe un formalisme, lui aussi basé sur le même critère d'optimalité au sens des moindres carrés, traitant le cas des

processus non-stationnaires, à coefficients variables dans le temps, et aujourd'hui appelé *filtre de Kalman*.

Après avoir défini les hypothèses de calcul, nous développons le cas simple de l'ajustement dans l'espace des temps d'un signal par un ensemble de fonctions de référence. Nous passons ensuite au cas de l'ajustement dans l'espace des fréquences, avant de terminer par le cas du filtre de Wiener proprement dit qui traite plus symétriquement bruit et signal. Finalement, on donne quelques indications sur les algorithmes de construction des filtres de Kalman.

Terminologie

En ce qui concerne le vocabulaire, notons ici qu'il existe en fait plusieurs problèmes de « filtrage » optimal suivant que l'on décide (ou qu'il est nécessaire) d'opérer ou pas en temps réel. Ainsi:

- réaliser l'estimation de filtrage à partir des seuls échantillons $D(t')$ tels que $t' \leq t$ s'appelle le *filtrage* (qui peut être réalisé en temps réel si on a une puissance suffisante de calcul).

- estimer $S(t)$ à partir des échantillons $D(t')$ quel que soit t' s'appelle le *lissage* (smoothing) ou *ajustement* (fit). Un tel ajustement est, strictement parlant, évidemment impossible à réaliser totalement en temps réel.

- enfin, dans certaines situations, on pourra également chercher à déterminer la valeur de $S(t+\tau)$ à un instant *futur* en fonction de la valeur des échantillons pris jusqu'à l'instant présent t . On parlera alors de *prédiction*.

Dans la suite, sauf mention explicite, on utilisera indifféremment la dénomination de « filtrage » pour chacun des cas précédents.

Hypothèses de calcul

Supposons que l'on ait affaire à un signal stochastique qui ne puisse être observé qu'en présence de bruit. Ce que l'on mesure, c'est un signal $D(t)$ qui est la somme d'un signal $S(t)$ d'amplitude variable, stochastique, et d'un bruit $N(t)$. Supposons également que le bruit possède une puissance spectrale déterminée (invariante au cours du temps, bien que l'expression de « spectre ne dépendant pas du temps » soit un peu paradoxale) et que les événements ont tous la même forme $S_0(t)$ mais que leur amplitude A diffère. Ce formalisme, qui traite assez symétriquement bruit et signal, supposés tous deux connus dans leurs distributions stationnaires, est celui qui mène au filtrage optimal de Wiener.

Dans beaucoup de situations, notamment celles qui mènent à l'utilisation des deux méthodes d'ajustement (fit) dans l'espace des temps et dans l'espace des fréquences que nous allons développer ici, la situation expérimentale est un peu différente dans le sens où bruit et signal ne sont pas traités sur un pied d'égalité du point de vue temporel. En effet, quand un événement survient, surtout dès qu'il a une amplitude qui dépasse nettement le bruit de ligne de base, on sait repérer son temps d'arrivée. Le but du jeu est alors de trouver le signal idéal de

paramètres d'amplitude et de décalage temporel à déterminer qui, soustrait au signal enregistré, ne laisse plus qu'un signal totalement compatible avec le bruit.

Supposons maintenant que chaque intervalle de fréquence se comporte comme une source de bruit gaussien d'amplitude $N(f)$ et de puissance moyenne (module carré)

$$|N(f)|^2$$

L'idée de l'ajustement optimal va donc être simplement de réaliser un ajustement qui remet chaque intervalle de fréquence à sa juste valeur et qui exprime tout le monde en rapport signal/bruit avant de réaliser un ajustement égalitaire, où tous les intervalles en fréquence se comportent après normalisation comme des variables gaussiennes de dispersion unité vis-à-vis du bruit. Dans cet espace un peu particulier, afin de déterminer la valeur de l'estimateur optimal de l'amplitude de l'événement, on va chercher à minimiser, au sens des moindres carrés, le χ^2 (résidu normalisé dont la fonction de distribution est calculable si les hypothèses concernant le caractère gaussien et d'indépendance des différents degrés de liberté du bruit sont bien vérifiées) en faisant varier les paramètres d'amplitude et de décalage temporel de l'événement idéal.

Ajustement (fit) dans l'espace des temps

Commençons par un cas qui peut sembler l'approche la plus évidente et qui nous permettra de résoudre la majeure partie du problème de minimisation et de ramener à une algèbre linéaire. Elle consiste à réaliser un ajustement (ici sur l'espace des temps) par un ensemble de m fonctions de références (sans référence au bruit et à sa puissance).

Supposons que nous ayons à notre disposition N échantillons du signal et que nous cherchions à représenter par une combinaison linéaire de fonctions \vec{F}_i . L'indice i désigne la fonction, et la flèche dans l'expression précédente désigne le fait que chacune des fonctions s'étend sur N échantillons et est en ce sens vectorielle. Dans ce premier problème de minimisation, on va chercher à minimiser l'écart quadratique entre le signal observé et une combinaison linéaire des fonctions de référence, ce qui revient à minimiser l'expression

$$\left(\vec{S} - \sum_i \lambda_i \vec{F}_i \right)^2$$

En exprimant que la dérivée est nulle par rapport au λ_i s'ils répondent à notre problème (quadratique) de minimisation, on obtient l'ensemble des conditions

$$\left(\vec{S} - \sum_i \lambda_i \vec{F}_i \right) \cdot \vec{F}_j = 0, \quad \forall j$$

Soit, de façon équivalente

$$\vec{S} \cdot \vec{F}_j = \sum_i \lambda_i \vec{F}_i \cdot \vec{F}_j, \quad \forall j$$

Si les fonctions \vec{F}_i sont indépendantes, il existe une solution unique à ce système matriciel. Dans le cas particulier, très simple, où les fonctions \vec{F}_i sont orthogonales entre elles, le système précédent se résout immédiatement

$$\lambda_i = \frac{\vec{S} \cdot \vec{F}_i}{\vec{F}_i \cdot \vec{F}_i}$$

D'un point de vue calculatoire, notons que, dans le cas général où les fonctions \vec{F}_i ne sont pas orthogonales entre elles, les $m(m+1)/2$ produits scalaires $\vec{F}_i \cdot \vec{F}_j$ peuvent être calculés une fois pour toutes et que les seuls calculs à effectuer à chaque nouvelle fournée de N échantillons consistent en les produits scalaires (N multiplications et $(N-1)$ additions pour chaque fonction) entre le signal mesuré (N échantillons) et chacune des fonctions de référence: $\vec{S} \cdot \vec{F}_i$. Excepté le fait que le calcul est matriciel si les fonctions de référence ne sont pas orthogonales entre elles, on voit que le problème possède une solution numérique directe et relativement simple.

Ajustement dans l'espace des fréquences

Puisque l'on a réalisé un ajustement au sens des moindres carrés dans l'espace des temps, et déterminé une solution unique en fonction des fonctions modèles \vec{F}_i , on pourrait penser que l'on a trouvé la solution optimale. Mais notre approche doit nous apparaître comme naïve du fait qu'elle ne fait pas intervenir les caractéristiques du bruit et que l'on doit bien se douter qu'une telle solution n'est sans doute pas optimale.

Revenons donc à nos hypothèses concernant le bruit. Comme nous avons supposé sur chaque intervalle en fréquence se comporte comme une source de bruit gaussien sans corrélation en fréquence vérifiant

$$\langle n(f) \rangle = 0$$

$$\langle n(f), n(f') \rangle = N^2(f) \delta(f - f')$$

où $n(f)$ représente une réalisation particulière d'un échantillonnage de bruit, alors que $N^2(f)$ représente la puissance moyenne associée.

Il devient alors naturel de minimiser l'expression de χ^2 :

$$\chi^2 = \sum \left(\frac{(D(f) - A \times S(f))^2}{|N(f)|^2} \right)$$

Attention, pour calculer cette somme, on calcule la quantité entre parenthèses pour chaque intervalle de fréquence et on somme les contributions sur toutes les fréquences.

Afin de déterminer l'amplitude A , dérivons l'expression précédente par rapport à A , et résolvons en supposant qu'elle vaut 0 (dérivée nulle pour l'optimum).

On trouve :

$$A = \frac{\sum \frac{D(f) \times S^*(f)}{|N(f)|^2}}{\sum \frac{|S(f)|^2}{|N(f)|^2}}$$

Afin de déterminer l'erreur typique que nous commettons dans l'estimation de l'amplitude, on peut déterminer la variance de cet estimateur, qui vaut :

$$\sigma_A^2 = \frac{1}{\sum \frac{|S(f)|^2}{|N(f)|^2}}$$

(comme on le vérifiera immédiatement du fait que $d\chi^2/dA^2$ est un terme constant en raison de l'expression quadratique du χ^2 en fonction de A).

Template du filtrage optimal

En utilisant le théorème de Parseval (invariance de la norme, et donc du produit scalaire) sous la transformée de Fourier, on peut repasser à une expression dans l'espace des temps plutôt que celui des fréquences (ce qui nous évitera la plupart du temps de calculer la transformée de Fourier) :

$$A = \sum D(t) \times F(t)$$

où $F(t)$, le template du filtrage optimal est construit (une fois pour toutes si l'on connaît effectivement la forme du signal attendu) en prenant la transformée de Fourier de

$$\frac{S^*(f)}{|N(f)|^2}$$

Déterminer précisément le spectre de bruit

Comment faire afin de déterminer précisément la puissance de bruit, qui intervient dans l'expression du template de l'ajustement optimal, associé à une ligne de mesure?

Supposons pour simplifier que l'on sache repérer les zones où il n'y a pas de signal ou encore que l'on puisse se tourner vers une région de mesure (un « blanc ») où seule le bruit apparaît dans le signal mesuré. Supposons également que l'on ait réalisé la transformée de Fourier et calculé la puissance en fonction de la fréquence pour un ensemble de N échantillons de bruit pur (en l'absence d'événements) pris à une fréquence $1/\Delta$.

Quelle est alors la précision sur la valeur de la puissance de bruit que l'on obtient par cette méthode pour chaque intervalle de fréquence grâce à la transformée de Fourier? La réponse peut paraître assez surprenante: dans la mesure où le bruit est gaussien pour chaque intervalle de fréquence, la variance de la mesure est (en moyenne) toujours égale au carré de la valeur trouvée pour

l'intervalle de fréquence considéré. Comment alors augmenter la précision de la détermination du spectre de bruit?

Pour cela, il existe principalement deux méthodes: on peut d'abord enregistrer un très grand nombre mN d'échantillons (où m est un entier le plus grand possible) toujours à la même fréquence $1/\Delta$ où l'on veut ensuite opérer ensuite dans la prise de données, puis en effectuer la transformée de Fourier. Comme on échantillonne à une fréquence déterminée $1/\Delta$, la fréquence de Nyquist associée n'a pas changé, mais du fait que l'on a augmenté le nombre mN de points de la prise de données, le pas en fréquence dans la transformée de Fourier est maintenant plus petit ($1/mN\Delta$). Il est alors possible de regrouper les m valeurs adjacentes dans le spectre obtenu afin de les additionner et d'obtenir une estimation plus précise de la puissance pour chaque valeur de la fréquence. Pour chacune de ces valeurs, on gagne ainsi un facteur \sqrt{m} par rapport à la détermination qui aurait été effectuée avec seulement N échantillons.

Une deuxième méthode, un peu plus efficace et surtout moins coûteuse du point de vue de l'utilisation de la mémoire, consiste à prendre un grand nombre m de séquences de N échantillons, à en calculer le spectre de puissance grâce à la transformée de Fourier et à effectuer la moyenne des spectres (évidemment pas l'inverse, qui consisterait à effectuer la moyenne des échantillonnages puis à en calculer le spectre de bruit). Alors que la première méthode requiert la création en mémoire d'un tableau de mN points, la deuxième méthode ne demande que la création d'un tableau de N points (le tableau somme que l'on divise *in fine* par le nombre m de prise de données « bruit »).

Si l'on soupçonne par ailleurs que des événements de petite amplitude peuvent venir se cacher dans le signal de « bruit » enregistré, il peut être utile d'établir un critère de vraisemblance permettant de déterminer la probabilité que la puissance totale, ou encore la puissance dans un certain intervalle de fréquence, dépasse une valeur déterminée (par exemple à 2.5 ou 3 écarts-standard de la valeur moyenne) et de rejeter les prises de données qui ne respectent pas ce critère sur la puissance.

Résumé: recette pratique afin de réaliser un ajustement optimal.

- Échantillonner le signal, évidemment en respectant le critère d'échantillonnage de Shannon.
- Réaliser un événement moyen (template dans l'espace des temps), en sommant une sélection d'événements et en prenant garde à ce qu'ils aient bien le même décalage en temps par rapport au point de déclenchement. Prendre garde au problème de l'alignement en temps (discriminateur à fraction constante).
- Réaliser un spectre de bruit (comment avoir une bonne précision).
- Réaliser le template dans l'espace des fréquences au moyen de la formule ci-dessus.

- Maintenant, lorsque les signaux arrivent, réaliser la convolution du signal enregistré avec le template afin de déterminer le temps d'arrivée et l'amplitude du signal, ou ajuster une parabole sur le résultat de la convolution.

Remarques sur l'ajustement optimal

Si l'on applique sans discernement les formules précédentes, l'ajustement (fit) dans l'espace des fréquences risque fort de donner des résultats décevants, voire plus mauvais qu'un filtrage dans l'espace des temps. D'où les quelques remarques qui suivent, destinées à indiquer les principaux pièges du filtrage optimal.

Remarque 1 : si le bruit est blanc, on voit très simplement en appliquant la formule de définition du « template » du filtrage optimal avec un terme constant pour $N(f)$ que l'ajustement optimal amène à un résultat identique à l'ajustement dans l'espace des temps. Dans ce cas, le « template » du filtrage optimal n'est rien d'autre que la forme de l'événement lui-même, sans autre transformation.

Remarque 2 : dans la mesure où les hypothèses sont vérifiées (chaque intervalle de fréquence se comporte comme une source aléatoire gaussienne indépendante des autres), seule la quantité que nous avons calculée à le droit de s'appeler un χ^2 . En particulier, le résidu de l'ajustement dans l'espace des temps, au chapitre précédent, n'a aucune raison d'être normalisé, et ne l'est en général pas. Ceci est dû au fait que les valeurs des échantillons en temps sont corrélées l'une à l'autre (suivent les fluctuations de la « ligne de base »), à la différence des valeurs des intervalles de fréquence et qu'il n'y a pas de mesure naturelle du bruit d'intervalle à intervalle dans l'espace des temps. Par normalisé, on entend que ce résidu vaut en moyenne 1 par « degré de liberté ». Ceci constitue un test important de la description que l'on a faite du signal et du bruit. Si en effet le χ^2 par degré de liberté dans l'espace des fréquences est nettement plus petit que 1 (aux fluctuations statistiques près), cela signifie en général que l'on a surestimé la puissance de bruit, ou, plus couramment, que l'on a sous-estimé la puissance de bruit dans le cas contraire ($\chi^2/N > 1$). Notons que cette valeur de χ^2/N différente de 1 peut être une indication du fait que le caractère gaussien du bruit n'est pas vérifié (voir également la remarque suivante).

Remarque 3 : Le cas du « bruit » de secteur 50 Hz, ou encore d'une résonance d'un appareil méritent quelques remarques particulières. Dans ce cas, les hypothèses de distribution gaussienne et d'indépendance d'un intervalle de fréquence à l'autre, nécessaires à la dérivation de l'estimation optimale, ne sont clairement pas vérifiées. Tout expérimentateur a vu apparaître sur un spectre de bruit le 50 Hz (60 Hz aux Etats-Unis) et ses harmoniques (ou sous-harmoniques). La phase du 50 Hz est évidemment « parfaitement » déterminée et pas une

variable aléatoire. Comme l'origine de ce bruit vient de l'alimentation électrique, il y a une parfaite corrélation en phase et en amplitude entre les composantes du 50 Hz et ses différents harmoniques. Cela signifie également que le système d'intérêt n'a pas un comportement purement linéaire (puisque à une excitation fréquentiellement pure, on fait correspondre en sortie le signal original *plus* ses harmoniques). Donc, lorsque le χ^2 par degré de liberté est nettement différent de 1, cela peut également signifier qu'il existe une forte corrélation entre les principales composantes du bruit.

C'est également ce qui se passe dans le cas des vibrations d'un système en résonance lorsque son facteur de qualité Q est élevé. Ici aussi, on a en général des harmoniques qui sont très fortement corrélées en phase et en puissance à l'excitation si elle existe. Le point important, comme dans celui du 50 Hz, est que l'on peut en général clairement voir sur le signal la résonance comme un phénomène cohérent, dont on peut repérer la phase.

Dans les deux cas précédents (50 Hz et phénomène de résonance), si un blindage plus efficace ou une plus grande rigidité du montage ne permettent pas de s'affranchir de ces composantes systématiques, il convient de déterminer la phase du 50 Hz (ou de la fréquence de résonance), et non pas d'appliquer brutalement à ces composantes particulières de bruit la procédure de filtrage optimal (qui supprimerait dans le même temps les composantes de même fréquence du signal). On peut par exemple déterminer très précisément la phase du 50 Hz en enregistrant le signal d'une des phases du secteur (attention au triphasé, il faut utiliser *deux* fonctions déphasées).

En d'autres termes, il convient de commencer par *soustraire* la plus grande partie possible de ce signal systématique avant d'appliquer la procédure de filtrage optimal qui va, elle, *supprimer* les composantes en fonction inverse de la puissance de bruit.

Remarque 4 : si l'on se contente de calculer l'amplitude par le produit scalaire de l'expression ci-dessus, on obtiendra en général une résolution qui peut se révéler très médiocre si l'on ne prend pas garde au décalage temporel entre le « template » et le signal réel. C'est le cas notamment lorsque le déclenchement s'effectue au moyen d'un discriminateur à niveau constant, c'est-à-dire d'un module qui fournit un signal de déclenchement dès que le signal dépasse un niveau constant au-dessus de la ligne de base. Il est facile de se rendre compte que ce type de déclenchement a tendance à accepter plus tôt les événements de grande amplitude, et plus tard les événements de faible amplitude. On trouve dans le commerce des modules appelés « discriminateurs à fraction constante » qui fournissent un signal de déclenchement approximativement toujours à la même fraction, paramétrable, par exemple 50%, du signal et cela indépendamment de son amplitude (idéalement). Mais si le signal est très proche du seuil effectif de

déclenchement, il est en général très bruité et les discriminateurs à fraction constante sont dans ce cas inefficaces.

Afin de contourner le problème du décalage temporel, il est également possible de calculer le produit scalaire qui intervient dans l'expression de l'amplitude pour différents décalages temporels du template. Dans une première approximation, l'amplitude est le maximum des produits scalaires réalisés pour les différents décalages en temps. Si l'on veut peaufiner le résultat, on peut ensuite ajuster une fonction (en général une parabole) sur le sommet de la courbe afin de déterminer le maximum d'amplitude. Le décalage temporel associé au sommet de la parabole constitue également l'estimateur du temps d'arrivée de l'événement.

Notons que c'est plus ou moins ce que l'on fait de toute façon dans le cas du véritable filtrage optimal (causal, en temps réel), où ce produit scalaire est calculé en permanence au fur et à mesure que sont acquis les échantillons. Si l'on effectue ce calcul strictement en temps réel, chaque étape de calcul du filtrage, si le template du *filtre* comporte N échantillons, nécessite tout de même N multiplications et $N - 1$ additions. Si les contraintes de temps réel ne sont pas trop fortes et que le nombre d'échantillons du template optimal est important, il peut être plus économique de découper les données en blocs d'échantillons (avec un recouvrement afin d'éviter les effets de bord) et de passer dans l'espace des fréquences où le produit de convolution s'exprime simplement comme une multiplication des transformées de Fourier. On revient ensuite dans l'espace des temps au moyen d'une transformée de Fourier inverse. Dans le cas où l'on a pris garde d'utiliser un nombre d'échantillons égal à une puissance de 2, les algorithmes de FFT (transformée rapide de Fourier) décrits précédemment peuvent être utilisés et le nombre d'opérations par échantillon est alors de l'ordre de $\log_2(N) \times O(1)$, où $O(1)$ est un facteur qui vaut quelques unités.

Remarque 5 : si l'on veut que l'ajustement (fit) optimal fournisse pour A une amplitude calibrée, que l'on peut directement relier au signal enregistré, il ne faut pas oublier d'inclure dans l'expression de l'amplitude le coefficient de normalisation, presque toujours absent de l'expression que l'on trouve dans les manuels, soit :

$$\frac{1}{\sum \frac{|S(f)|^2}{|N(f)|^2}}$$

Remarque 6 : En pratique, il peut être efficace de procéder en deux temps. On peut d'abord réaliser un filtrage optimal assez grossier en temps réel et de déclarer que l'on a un déclenchement à un seuil fixe à 3 ou 4 fois le bruit de la ligne de base (le critère exact dépend de la puissance de calcul que vous avez à votre

disposition, ainsi que de la forme effective du bruit à basse fréquence: plus il y a du bruit en $1/f$, plus la tactique précédente est difficile à appliquer). Cette stratégie permet de ne conserver que les candidats événements les plus probables (ce filtrage de sélection s'effectue en général avec un assez petit nombre de coefficients afin de respecter les contraintes imposées par le « temps réel », les coefficients du filtre pouvant être obtenus par exemple par décimation du template optimal). Une fois obtenu un déclenchement par cette méthode, on peut alors déclencher l'enregistrement du post-trigger et procéder alors à un ajustement optimal au moyen de la procédure décrite plus haut.

Remarque 7 : Il n'est pas évident a priori que notre critère de minimisation au sens des moindres carrés (qui additionne par exemple sans discrimination les résidus de chacun des intervalles de fréquence) soit le plus adapté. Si notre connaissance du signal et du bruit sont correctes et que le comportement de ceux-ci sont invariants au cours du temps, ne dépendent pas d'autres paramètres (comme l'amplitude, si une non-linéarité se fait jour, par exemple), et si l'échantillonnage en fréquence est adapté aux caractéristiques du signal et du bruit, notre critère des moindres carrés est adapté. Il n'est pas interdit par contre de vérifier les hypothèses sur le signal et le bruit en vérifiant que la *fonction* résidu, écart entre le signal réellement observé et le signal de forme idéale reconstruit, n'a effectivement pas de structure particulière.

On pourra trouver un exemple concret d'analyse de signal au moyen du fit optimal dans l'article suivant: A. E. Szymkowiak, R. L. Kelley, S. H. Moseley et C. K. Stahle, J. of Low Temp. Phys. 93 (1993) 281, qui décrit l'analyse suivie par les « recordmen » de la résolution bolométrique.

Filtrage optimal de Wiener

Le problème du filtrage optimal se trouve souvent exprimé dans les ouvrages de traitement du signal sous un formalisme un peu différent, où signal et bruit sont traités de façon plus symétrique.

Supposons ainsi que l'on cherche à retrouver dans un signal mesuré $s(t)$ un « vrai » signal $u(t)$. Le signal observé peut être « corrompu » pour deux raisons principales:

- d'une part, le système qui traite le signal physique peut avoir une fonction de réponse étendue qui émousse la forme du signal initial. Supposons donc que la fonction de transfert du signal soit $r(t)$ dans l'espace des temps et $R(f)$ dans l'espace des fréquences. Dans ce cas, l'influence de la fonction de transfert sur le signal s'écrit simplement:

$$s(t) = \int_{-\infty}^{\infty} r(\tau)u(t - \tau)d\tau$$

ou, dans le langage de l'espace des fréquences:

$$S(f) = R(f)U(f)$$

Il est donc très facile, comme on l'a déjà vu plus haut à propos du « smearing », de prendre en compte une fonction de transfert: il suffit de diviser par elle dans l'espace des fréquences, ce qui est possible dès que la fonction de transfert ne s'annule pas, c'est-à-dire ne met pas à zéro certaines zones de fréquence.

Le cas du bruit constitue la deuxième atteinte au signal, celle-là un peu plus difficile à combattre, d'autant que la solution ne sera cette fois-ci pas exacte. Supposons maintenant que le signal mesuré $c(t)$ (ou $C(f)$ dans le langage des fréquences) soit la somme du signal précédent $s(t)$ et d'un bruit $n(t)$.

Nous allons définir le filtre optimal à coefficients constants sous la condition que si l'on applique le filtre $\phi(t)$ ou $\Phi(f)$ au signal mesuré composite $c(t)$, on produise le signal le plus proche possible, à nouveau au sens des moindres carrés, du « véritable » signal, inaccessible, $u(t)$.

Nous cherchons donc à optimiser le filtre $\Phi(f)$ de façon à ce que:

$$\int_{-\infty}^{\infty} \left| \frac{[S(f) + N(f)]\Phi(f)}{R(f)} - \frac{S(f)}{R(f)} \right|^2 df$$

$$= \int_{-\infty}^{\infty} \frac{1}{|R(f)|^2} \{ |S(f)|^2 |1 - \Phi(f)|^2 + |N(f)|^2 |\Phi(f)|^2 \} df$$

soit minimal, et où nous avons utilisé le fait que le bruit $N(f)$ et le signal $S(f)$ sont *a priori* décorrélés en fréquence, ce qui est équivalent à ce que leur produit intégré soit nul.

Différenciant par rapport à Φ , on trouve une équation linéaire en Φ qui donne

$$\Phi(f) = \frac{|S(f)|^2}{|S(f)|^2 + |N(f)|^2}$$

C'est la forme du filtre optimal au sens de Wiener. On voit que ce filtre prend toujours ses valeurs entre 0 et 1, et que, comme dans le cas précédent, il défavorise les régions où le rapport signal/bruit est mauvais.

Dans la mesure où signal et bruit sont traités de façon symétrique, il reste cependant à définir une procédure afin de déterminer le spectre du signal et le spectre de bruit. Pour cela, si l'on ne dispose pas d'une région de référence (ou « blanc ») permettant de se calibrer en bruit, il faudra là aussi sélectionner une région particulière de signal où l'on pourra isoler le spectre du signal et soustraire ensuite la fonction trouvée au spectre composite, qui est le seul que l'on puisse obtenir sur la région de signal + bruit elle-même, cela afin de déterminer le spectre de bruit.

Et si les conditions changent au cours du temps ?

Dans les paragraphes précédents, nous avons fait l'hypothèse que la forme de signal et que le spectre de bruit étaient bien définis et invariants au cours du

temps. Mais il arrive fréquemment dans la réalité que forme de signal ou spectre de bruit évoluent (dérive des paramètres du détecteur, ou des conditions expérimentales: orientation du détecteur, spectre de source variable...). Dans la mesure où les formes fonctionnelles de ces aspects du signal peuvent être précisés, il est possible de compléter le formalisme développé par Wiener afin de prendre en compte l'évolution des caractéristiques. On décrit ici de façon très rapide les bases du formalisme développé par Kalman et qui donnent lieu au filtrage de Kalman.

Algorithme récursif de Kalman (*Désolé, ce paragraphe n'est pas achevé...*)

IX - Quelques mots sur les transformations temps-fréquence

- Transformations en ondelettes

On a vu fleurir au cours de ces quelques dernières années les études et les programmes destinées au traitement des données (filtrage, analyse, ...) en termes d'ondelettes. En conclusion de ce paragraphe, on trouvera ainsi les références des deux "packages" qui apparaissent proposer des essais de standard en matière de traitement en ondelettes.

Comme dans le cas des traitements temps-fréquence, tel le spectrogramme de Gabor, le traitement en ondelettes tente de contourner l'un des problèmes posés par l'analyse de Fourier. En effet, l'analyse en termes de fréquences pures, c'est-à-dire de fonction sinus et cosinus, fonctions à valeur évidemment non nulle, sauf en des points isolés, sur l'ensemble de l'intervalle d'étude, pose deux problèmes :

- d'une part, elle n'est pas adaptée aux signaux localisés dans le temps et la puissance du signal peut se trouver "diluée" dans celle du bruit si l'intervalle d'étude n'est pas adapté

- d'autre part, l'approximation du signal en termes de ses composantes « principales » de Fourier est rien moins qu'évidente. Il arrive souvent en effet, dans le cas de l'analyse de Fourier, que le nombre de composantes qui interviennent de façon significative dans la décomposition d'un signal localisé dans le temps fasse intervenir un grand nombre de composantes de Fourier. Ce caractère résulte d'ailleurs presque nécessairement de la relation d'incertitude temps-fréquence. Par opposition, le traitement en ondelettes fait intervenir un traitement en approximations successives où l'on peut voir émerger avec plus de précision, à chaque itération, la forme du signal en fonction de ses composantes à chaque niveau de détail de la description. En outre, l'analyse en ondelettes fait intervenir des fonctions à support compact, donc a priori mieux adaptées à l'analyse de phénomènes limités dans le temps et/ou l'espace, comme des impulsions dues aux passages de particules individuelles. Par contre, dans le cas d'un signal modulé, comme c'est souvent le cas dans les mesures infrarouges, le traitement en ondelettes n'apportera sans doute d'amélioration dans le traitement de l'information.

Afin de donner une idée un peu plus précise de l'analyse en ondelettes, et de sa hiérarchie de décomposition, tentons de résumer la philosophie de la décomposition en ondelettes. Le principe est d'utiliser une « ondelette-mère » dont les translations, en générale par pas entiers, constituent une base de l'ordre zéro de la décomposition en ondelettes. Un exemple d'ondelette est donné, par exemple par les ondelettes de Daubechies de la figure 18.

Les approximations successives, ou détails, de la fonction à étudier sont alors révélés par les ordres suivants, obtenus par dilatation et translation à partir de l'ondelette-mère, de la décomposition en ondelettes.

On devine à cette description que le choix de la famille d'ondelettes de base permet d'arriver à des compressions de données plus ou moins efficaces et est donc une étape fondamentale. En particulier, on connaît des familles d'ondelettes de classe C_i (continuité de la fonction et de ses dérivées à l'ordre i) et même C_∞ (une famille a été découverte par Y. Meyer, de l'Ecole Polytechnique).

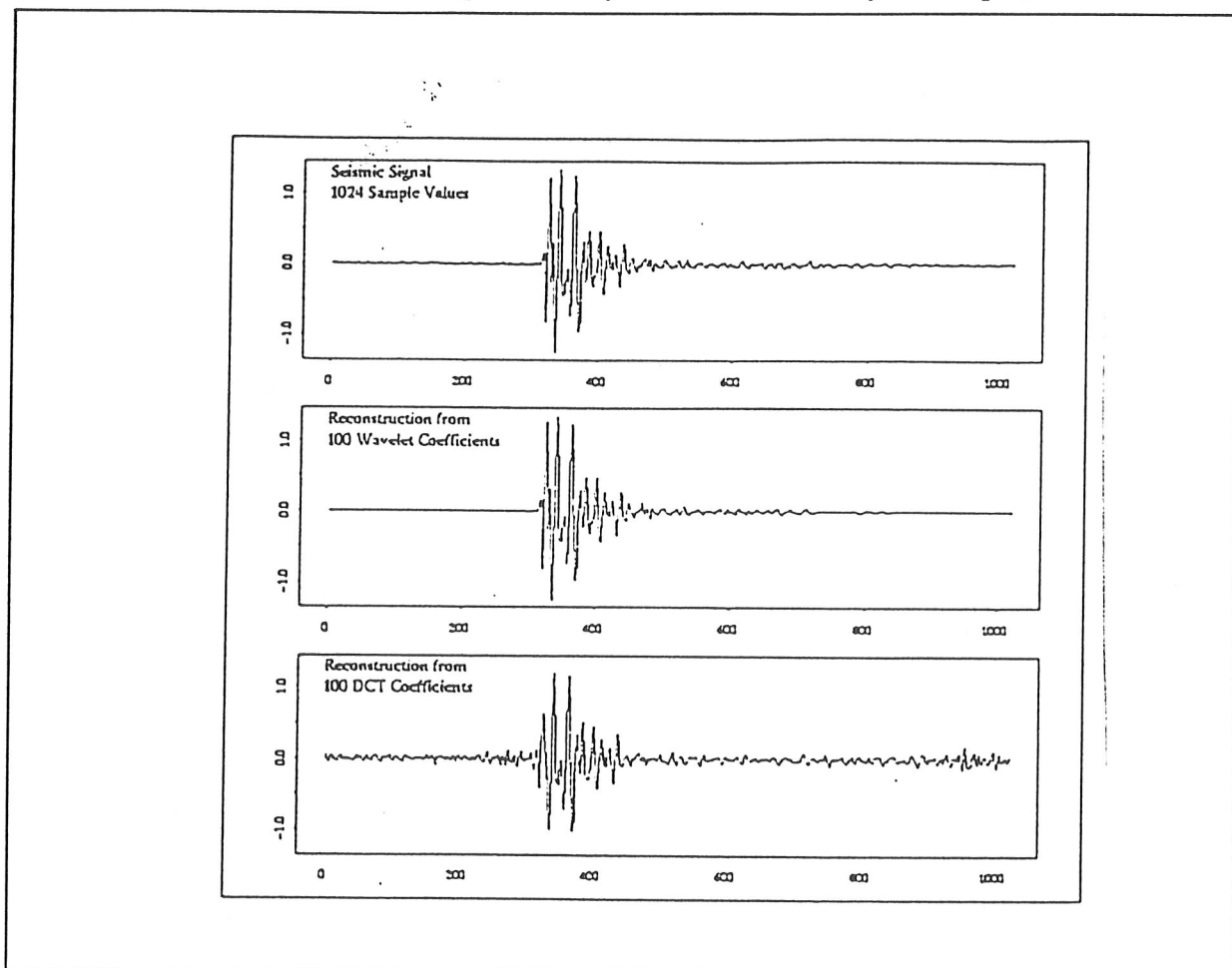


Fig. 17 : Signal sismique de 1024 échantillons (haut), reconstruit au moyen de 100 coefficients d'ondelettes (milieu), et reconstruit à partir de 100 termes de Fourier (bas). La décomposition en ondelettes reproduit clairement plus fidèlement le signal original.

A l'évidence, il existe un très grand nombre de fonctions qui peuvent être utilisées comme ondelette-mère et servir à la construction d'une base pour les

1. Commercial software for wavelet analysis

Package	Contents	Available on:	Supplier
Adapted Waveform Analysis	Subroutine library for wavelet, wavelet packet, and cosine packet analysis for 1- and 2-D data	Any workstation	F M A & H Corp. 1020 Sherman Ave. Hamden, CT 06514 203-248-8212
Adapted Wavelet Analysis	C subroutine library for wavelet, wavelet packet, and cosine packet analysis for 1- and 2-D data; associated with the book <i>Adapted Wavelet Analysis From Theory to Software</i> by Mladen Victor Wickerhauser (A.K. Peters, 1994)	DOS, Macintosh, Unix	A K Peters Ltd. 289 Linden St. Wellesley, MA 02181 617-235-2210 http://www.tiac.net/users/akpeters
JTFA	A stand-alone toolkit for joint time-frequency analysis of 1-D signals	Unix, Windows, Macintosh	National Instruments 6540 Bridge Point Parkway Austin, TX 78730 512-794-0100 http://www.natinst.com
MacWavelets	Wavelet analysis of 1- and 2-D data	Macintosh	ntergalactic Reality Cupertino, CA 95014 408-255-7958 http://www.intergalact.com
Matlab Wavelets Toolbox	A Matlab toolbox for wavelet and wavelet packet analysis of 1- and 2-D data, complete with graphical user-interface	Unix, Windows, Macintosh	The MathWorks Inc. 24 Prime Park Way Natick, MA 01760-1520 508-653-1415 http://www.mathworks.com
Mathematica Wavelet Explorer	Written in the Mathematica language with fully programmable built-in functions and utilities; features include 1- and 2-D wavelet and wavelet packet transforms	Unix, Windows, Macintosh	Wolfram Research Inc. 100 Trade Center Dr. Champaign, IL 61820-7237 800-441-6284 http://www.wolfram.com
Numerical Recipes	Subroutine library associated with the book <i>Numerical Recipes in C and Numerical Recipes in Fortran</i> , second editions by William Press, Saul Teukolsky, William Vetterling, and Brian Flannery (Cambridge University Press, 1992)	Any workstation	Cambridge University Press 40 West 20th St. New York, NY 10011-4211 800-872-7423 http://www.cup.org
PV-WAVE Signal Processing Toolkit	Add-on to PV-WAVE Advantage for signal processing, including tools for wavelet analysis and filter design	Unix, Windows, Macintosh	Visual Numerics Inc. PV-Wave Products Division 6230 Lookout Rd. Boulder, CA 80301 800-447-7147 http://www.vni.com
S+Wavelets	An object-oriented interactive language based on the S-PLUS software system for wavelet, wavelet packet, cosine packet, and matching pursuit analysis for 1- and 2-D data;	Unix, Windows	Data Analysis Products Division, MathSoft, Inc. Seattle, WA 98109 800-569-0123 http://www.mathsoft.com
The SignaScope	An interactive toolkit for time-scale and time-frequency analysis of 1- and 2-D data	Unix	Signition Inc. Box 1020 Los Alamos, NM 87544-1020 505-455-2789
Tsunami Plus	An efficient library of C routines for wavelet and wavelet packet analysis of 1- and 2-D data	DOS	Mac A. Cody Associates 408 Worcester Way Richardson, TX 75080-3433 214-437-1517 http://www.dfw.net/~mcody
WavBox	A Matlab wavelet toolbox for wavelet and wavelet packet analysis of 1- and 2-D data. Associated with the <i>Handbook of Wavelet Transform Algorithms: Wav Box Software User's Guide, Atlas, and Manual</i> by Carl Toswell (L. Birkhauser, 1996)	Unix	WavBox ToolSmiths Box 9925 Stanford, CA 94309-9925 415-323-4336 http://www.wavbox.com
WaveTool	A general wavelet and multirate software toolbox featuring optimal filter design and signal processing tools	Unix	Aware Inc. One Oak Park Bedford, MA 01730 617-276-4000 http://www.aware.com
WpLab	A user-friendly interactive environment for wavelet, wavelet packet, and cosine packet analysis of 1-D data; developed by Digital Diagnostic Corp. and Yale University	Windows	A K Peters Ltd. 289 Linden St. Wellesley, MA 02181 617-235-2210 http://www.tiac.net/users/akpeters

2. Public domain or shareware software for wavelet analysis

Package	Contents	Developed by:	Internet or other address
Wavelet Workbench	IDL library derived from WaveLab	Amara Graps for Research Systems Inc.	http://www.amara.com/wwwbdev/wwwbdev.html
Khoros Wavelet and Compression Toolbox	Khoros toolbox	Computer Research and Applications Group at Los Alamos National Laboratory	http://www.c3.lanl.gov/~cjhamil/Wavelets/main.html
Matching Pursuit Package	Unix tools and C code	Stéphane Mallat and Zhifeng Zhang	ftp://cs.nyu.edu/pub/wave/software
Mathematica Wavelet Programs	Mathematica programs	Jack Cohen, Tong Chen, and Meng Xu	ftp://timna.Mines.edu/wavelets
MegaWave	Unix tools and C code	Jacques Froment	http://www.ceremade.dauph.ft/~mw
Rice Wavelet Toolbox	Matlab toolkit	DSP Group at Rice University	http://jazz.rice.edu/software/RWT
Uvi_Wave	For Matlab and Khoros 2	Signal Theory Group at University of Vigo Bayona-Vigo, Spain	http://www.tsc.uvigo.es/~wavelets/uvi_wave.html
W-Transform	Matlab toolkit	M. K. Kwong	ftp://info.mcs.anl.gov/pub/W-transform
Waterloo Maple Software Wavelet Share Libraries	Maple wavelet library	Jacques Gelinas	ftp://daisy.uwaterloo.ca/pub/maple/5.3/share/daub
Wavelet at Imager	C routines	Bob Lewis	http://www.cs.ubc.ca/nest/imager/contributions/bobl/wvl/top.html
WaveLab	Matlab toolkit	J. Buckheit, S. Chen, D. Donoho, X. Huo, I. Johnstone, J. Scargle, and T. Yu	http://playfair.stanford.edu:80/~wavelab/
WaveLib	C library with Matlab interface	Mikael Bourges-Sevenier	http://www.sim.int-evry.fr/~bourges/WaveLib.html
Wavelet Toolbox	Matlab toolkit	David Newland	(e-mail) den@eng.cam.ac.uk
Wavelet Toolbox in Khoros	Khoros toolbox	Jonio Cavalcanti and Ramiro Jordon	ftp://ftp.khoral.com/pub/khoros/Khoros2/contrib/toolboxes/wavelet
Wavelets for Kids	Mathematica notebook	Peter Mueller and Brani Vidakovic	ftp://ftp.isds.duke.edu/pub/Users/brani/papers
WaveThresh	S-Plus library	Guy Nason	http://www.stats.bris.ac.uk/pub/software/wavetresh/WaveThresh.html
XWPL	X Window system	Fazal Majid	http://pascal.math.yale.edu/pub/wavelets/software/xwpl/html/xwpl.html

approximations successives de la décomposition précédente. Suivant les caractéristiques du signal à étudier, l'une ou l'autre des bases peut apparaître plus adaptée et on se trouve là devant l'un des degrés de liberté de ce type d'analyse.

En termes de compression, on peut voir sur la figure 19 un exemple de décomposition d'un signal de vibrations sismiques qui montre clairement l'intérêt d'une décomposition en ondelettes.

On trouvera ci-après deux tableaux indiquant les principaux « packages », commerciaux et en libre distribution, consacrés à l'étude des ondelettes (et de quelques autres aspects de l'analyse temps-fréquence). Parmi ceux-ci, deux d'entre eux semblent actuellement s'imposer en raison d'une part de leur caractère multi-plateforme (PC, Mac, stations Unix, ...) et de l'étendue de leur implantation sur ces plateformes. Il s'agit de packages de « Mathematica » de Wolfram Research, et de la « Wavelet Toolbox » de MATLAB. La toolbox « Wavelet Explorer » de Wolfram Research, distribuée en France par Ritme, est proposée à un coût de l'ordre de 4000 FF (il faut avoir à sa disposition Mathematica en version 2.2 au moins afin de pouvoir l'utiliser).

Le prix de la « Wavelet Toolbox » de MATLAB est plus conséquent encore puisqu'il se situe aux alentours de 35.000 FF en prix public (avec néanmoins des réductions très importantes pour l'enseignement).

Notons qu'il existe une autre transformée temps-fréquence, maintenant un peu dépassée par l'engouement pour les ondelettes, le spectrogramme de Gabor. Il s'agit d'une décomposition qui utilise les fonctions de Wigner en « paquets d'onde » gaussiens, également utilisées dans le traitement du signal et, notamment la reconnaissance de la parole. Signalons que National Instruments propose un « package » Joint Time-Frequency Analysis (JTFA) qui montre les possibilités de ce type de transformations. (*Ce paragraphe n'est pas achevé. Désolé...*)

Appendice

Comme signalé en introduction, une grande partie de ce cours a été illustrée par des simulations ou des calculs effectués au moyen de programmes (des Instruments Virtuels ou VIs dans la terminologie de *National Instruments*) écrits sous LabVIEW. Ce logiciel, qui comporte de nombreuses possibilités de pilotage et d'échange d'information avec du matériel d'acquisition, ainsi que des bibliothèques très complètes de traitement du signal, est devenu au fil des années l'un des standards en termes d'acquisition des données et de traitement associé « en ligne ». Disponible sur les plate-formes PC, Mac, et les stations HP et Sun Solaris, les codes sources sont transportables d'une plate-forme à l'autre sans modification si l'on n'utilise pas d'extensions au langage.

D'importants groupes d'utilisateurs existent depuis quelques années au CEA et au CERN, et le nombre de ces utilisateurs permet d'envisager de ne pas « reconstruire le monde » à chaque application mais de pouvoir disposer des bibliothèques et du savoir-faire des utilisateurs voisins. Dans cette optique, le CEA a réalisé, en collaboration avec National Instruments et le CERN, un CD-ROM regroupant un grand nombre d'utilitaires et des exemples d'applications réalisés par des utilisateurs du CEA et du CERN, ainsi que la dernière version des quelques centaines de « drivers » (pilotes d'instruments) disponibles dans le domaine public. Les utilisateurs du CEA devraient a priori avoir reçu le CD-ROM en question, mais toute personne intéressée peut envoyer un email soit à moi-même (chardin@hep.saclay cea.fr) soit encore à Patrice Menou de *National Instruments* (patrice.menou@natinst.com) afin de recevoir une copie (gratuite...) du CD-ROM.

Les instruments virtuels (VIs) sur lesquels a été basé une large partie de ce cours sont également disponibles sur support disquette ou site ftp (malheureusement pas permanent pour le moment) pour les personnes intéressées. Notez cependant qu'ils n'ont pas été créés en tant qu'applications indépendantes et qu'ils nécessitent donc que vous ayez à votre disposition le logiciel LabVIEW.

Quelques références bibliographiques

Baher H., *Analog and digital signal processing*, (John Wiley & Sons, Chichester, 1990).

Bellanger M., *Traitement numérique du signal*, (Masson, Paris, 1981).

Bracewell R.N., *Numerical Transforms*, Science vol. 248, numéro du 11 mai 1990.

Bruce A., Donoho D., Hong-Ye G., *Wavelet Analysis*, IEEE Spectrum, Octobre 1996, pp. 26-35.

Chen C.H. et al., *Signal Processing Handbook*, (Marcel Dekker Inc, New York, 1988).

Cunningham E.P., *Digital filtering, an introduction*, (John Wiley & Sons, New York, 1995).

→ Duvaut P., *Traitement du signal, Concepts et applications*, (Hermès, Paris, 1991).

Elliot D.F., *Handbook of Digital Signal Processing Engineering Applications*, (Academic Press, San Diego, 1987).

Gasquet C. et P. Witomski, *Analyse de Fourier et applications*, (Masson, Paris, 1990).

Harris F.J., *On the use of windows for harmonic analysis with the Discrete Fourier Transform*, Proceedings of the IEEE-66, 1978, Vol. 1.

LabVIEW Analysis VI Reference Manual, (National Instruments, Austin, 1996) (l'un des volumes de la documentation du logiciel LabVIEW, langage graphique de programmation adapté à l'acquisition des données et à leur traitement).

McClellan J.H., T.W. Parks et L.R. Rabiner, « *A computer program for designing optimum FIR linear-phase filters* », IEEE Transactions on Electroacoustics, AU-21, pp. 506-526.

Meyer Y., *Ondelettes et opérateurs, Tome I: Ondelettes* (Hermann, Paris, 1990).

Meyer Y., S. Jaffard, O. Rioul, *L'analyse par ondelettes*, Pour la Science, Septembre 1987.

Parks T.W. et C.S. Burrus, *Digital Filter Design*, (John Wiley & Sons, New York, 1987).

Press W., Teukolsky S., W. Vetterling, B. Flannery, *Numerical Recipes in C, 2nd edition*, (Cambridge University Press, Cambridge, 1992).

Rabiner L.R. et B. Gold, *Theory and Application of Digital Signal Processing*, (Prentice Hall, Englewood Cliffs, 1975).

Szymkowiak A.E., R. L. Kelley, S. H. Moseley et C. K. Stahle, J. of Low Temp. Phys. 93 (1993) 281.

