

Master IMA M1  
UE Signal et Image  
Partie 1 : Introduction

P. Courtellemont

[pcourtrel@univ-lr.fr](mailto:pcourtrel@univ-lr.fr)

<http://perso.univ-lr.fr/pcourtrel/>

# Signal et Image

## Plan du cours

### I. Introduction

#### Approche fréquentielle

- Signaux monodimensionnels périodiques
- Signaux quelconques
- Signaux numériques, discrétisation, échantillonnage
- Observation spectrale, TFD et TFD 2D

#### Systèmes numériques

- Filtres numériques
- Produits de convolution. Cas 2D
- Sur et sous échantillonnage. Bancs de filtres
- Traitement des Images.
- Approches multi résolution

### II. Signaux aléatoires

- variables aléatoires
- Processus aléatoires. Stationnarité
- Processus MA, AR et ARMA
- Estimation des paramètres d'un AR

### III. Traitement de l'information

#### Application à la compression

- codage de source, entropie,
- compression sans perte : codages entropiques

- par dictionnaire, par prédiction...

- Codages par transformée

- La DCT et la compression JPEG

- Quantifications scalaire et vectorielle

### IV. Communications numériques

- Modulations numériques
- modulation par impulsions codées
- transmission du signal numérique
- Applications : RDS, NICAM...
- Détection et correction d'erreurs



# Signal et Image

## Plan de la partie 1 (ce document)

### Repères :

Introduction

[page 4](#)

Signaux monodimensionnels périodiques - Séries de Fourier

[page 11](#)

Signaux quelconques – Transformée de Fourier

[page 18](#)

Signaux numériques, discrétisation, échantillonnage

[page 21](#)

Observation spectrale, TFTD et TFCT

[page 27](#)

La TFD

[page 31](#)

Algorithmes de FFT

[page 40](#)

Produits de convolution. Convolution circulaire et filtrage par FFT

[page 49](#)

Sur et sous échantillonnage.

[page 58](#)

Codages en sous bandes. Approches multi résolution, filtres QMF

[page 65](#)



# Signal et Image

## Introduction

Que signifie *signal* ? toute entité véhiculant une *information* (Picinbono 1989)  
C'est une représentation obtenue à partir des variations d'une grandeur physique.

exemples :

grandeur physique

information



Onde acoustique

musique, parole...

Courant électrique

mesure physique...

Traitement du signal :

extraire l'information (filtrage, détection, estimation, analyse spectrale...)

mettre en forme le signal (modulation, échantillonnage....)

-> forme adaptée à la transmission ou au stockage

analyser l'information

-> reconnaissance de formes



# Signal et Image

Quelques applications en vrac :

- téléphonie mobile,
- cartes son, synthèse musicale, ...
- codage vidéo et audio en visiophonie, formats d'échange audio et vidéo, ...
- reconnaissance de la parole,
- vision industrielle,
- suivi de cibles, radars, ...
- analyse du sous sol (ondes sismiques, recherche pétrolière...)
- aide au diagnostic médical (EEG, ECG, ...)
- GPS,
- ... !!

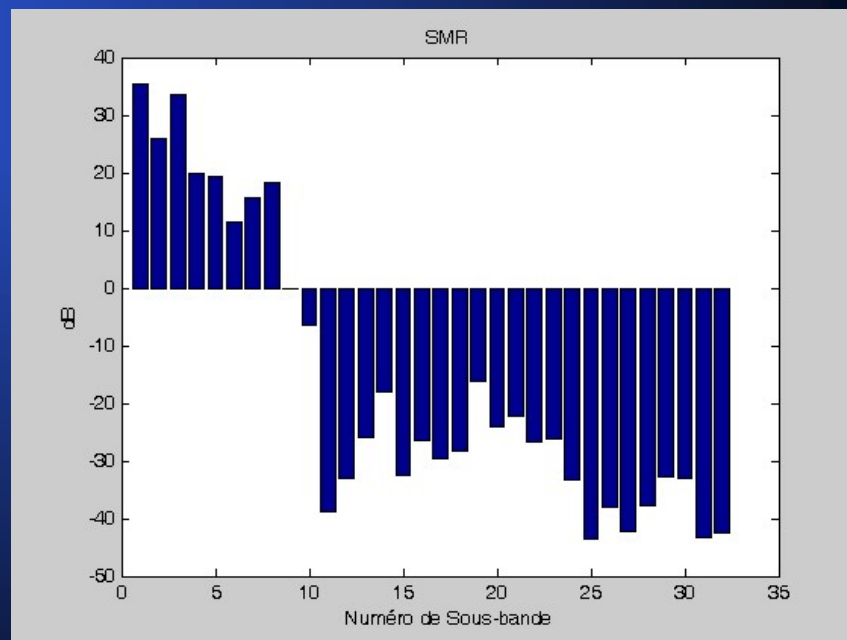


# Signal et Image

Exemples d'applications traitées dans le cadre de ce cours (et réalisées sous matlab) :



- détection du numéro appelant (DTMF)
- codeur MPEG audio layer I
- suppression du bruit moteur en téléphonie mobile
- annulation d'écho en téléphonie
- codage de canal
- codeur JPEG
- codeur MP3



# Signal et Image

## Classification des signaux :

- selon leur dimension (**classification dimensionnelle**)

Tension électrique  $v(t)$  = signal *unidimensionnel*

Image statique niveaux de gris  $\Leftrightarrow$  luminance  $I(x,y)$  = signal *bidimensionnel*

Séquence d'images  $\Leftrightarrow I(x,y,t)$  = signal *tridimensionnel*

La théorie du signal est *indépendante* de la *nature physique* du signal

- selon le caractère déterministe ou aléatoire (**classification phénoménologique**)

Signal déterministe : l'évolution peut être parfaitement prédite par un modèle mathématique approprié

Signal aléatoire : comportement *imprévisible*  $\Rightarrow$  description *statistique*

tout signal physique comporte une *composante* aléatoire  
(perturbation externe, phénomène quantique ...)

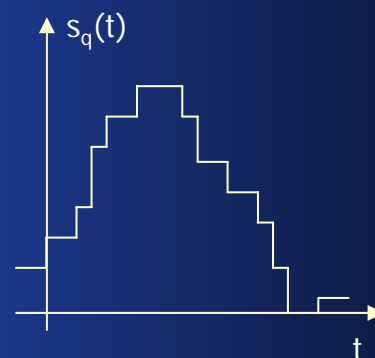
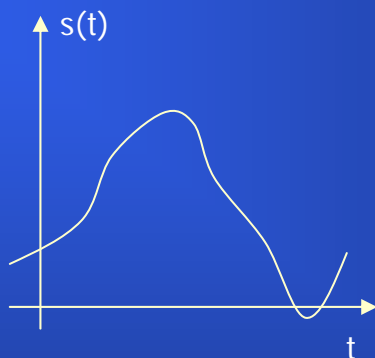
- selon leur nature discrète ou continue (**classification morphologique**)

# Signal et Image

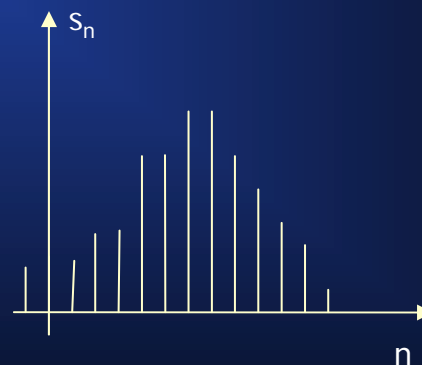
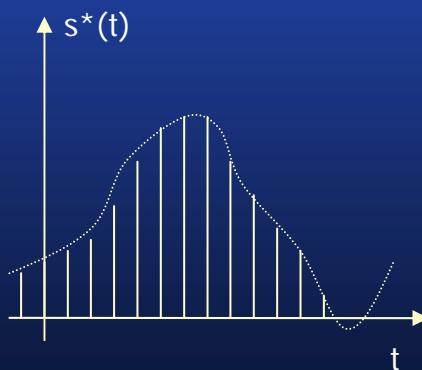
continu

discret

continu



discret







# Signal et Image

Signaux continus et signaux discrets :

Signal « à temps continu » = signal susceptible d'être observé et mesuré à chaque instant.

La plupart des phénomènes physiques (en tous cas les phénomènes naturels) sont de nature continue.

Le signal est dit à temps discret lorsqu'il est défini et susceptible d'être mesuré à des certains instants seulement.

Les signaux continus et échantillonnés *à intervalles réguliers* font partie de cette catégorie.

Quelques fréquences d'échantillonnage repères :

Signal	Fréquence d'échantillonnage
parole	8 kHz (téléphonie) 16 kHz (conférence audio)
Audio	32 kHz 44,1 kHz 48 kHz
Vidéo	Environ 10 MHz (définition PAL)





# Signal et Image

Moyenne, énergie et puissance :

La *moyenne* d'un signal déterministe se définit par, avec  $T_1$  temps d'intégration :

$$m = \lim_{T_1 \rightarrow +\infty} \frac{1}{T_1} \int_{T_1} x(t) dt$$

Cette valeur représente la composante continue du signal. Pour un signal périodique de période  $T_0$ , cette valeur se ramène à :

$$m = \frac{1}{T_0} \int_{T_0} x(t) dt$$

Pour un signal à temps continu  $x(t)$ , et un signal à temps discret  $x(k)$ , leur *énergie* s'exprime par :

$$E = \int_{-\infty}^{+\infty} |x(t)|^2 dt \quad \text{et} \quad E = \sum_k |x(k)|^2$$

et leur *puissance moyenne* par :

$$P = \lim_{T \rightarrow +\infty} \frac{1}{T} \int_T |x(t)|^2 dt \quad \text{et} \quad P = \lim_{N \rightarrow +\infty} \frac{1}{N} \sum_{k=1}^N |x(k)|^2$$





# Signal et Image

## Modélisation des signaux

En signal, une approche naturelle est **l'approche fréquentielle** : elle a un sens physique immédiat.

En image, elle est une approche parmi d'autres modèles, qui sert de référence aux techniques utilisées en traitement ou en analyse d'images.

On considère alors l'image comme un signal, au départ analogique, en 2 dimensions.

### Un principe général :

Un signal peut se décomposer en une somme de signaux particuliers, plus simples car pouvant être décrits par un nombre réduit de paramètres. (on parle de fréquences pures en signal).

Cette modélisation repose sur le principe de linéarité : les outils agissent sur chaque partie d'un tout comme si chacune des parties était seule. La **linéarité** est associée à un grand nombre d'outils mathématiques qui exploitent cette propriété (très forte mais finalement suivie par la nature). Ce principe s'oppose aux approches système ou systémiques (le tout l'emporte sur les parties).

### Signaux monodimensionnels périodiques

Le signal variable le plus simple est le signal sinusoïdal  $s(t) = a \sin(2\pi f_0 t)$  avec  $f_0 = 1/T_0$ . 2 paramètres le caractérisent : son amplitude  $a$  et sa fréquence  $f_0$ .

En changeant l'origine des temps en  $t_0$ , on obtient :  $s(t) = a \sin(2\pi f_0 t - \phi_0)$  avec  $\phi_0 = 2\pi f_0 t_0$ .

$\phi_0$  désigne la phase, troisième paramètre permettant de décrire complètement la sinusoïde.





# Signal et Image

La représentation fréquentielle des signaux cherchent à représenter les 3 paramètres précédents qui caractérisent la sinusoïde,  $a$ ,  $f_0$  et  $\phi_0$ . A la place d'un point dans un espace 3D, on préfère décomposer en 2 graphiques 2D où les 2 points  $(a, f_0)$  et  $(\phi_0, f_0)$  ont une même abscisse sur un axe des fréquences.

## Exponentielles complexes

L'exponentielle complexe est plus simple à manipuler mathématiquement. La fonction exponentielle complexe est une fonction à valeur complexe dont la partie réelle est un cosinus et la partie imaginaire un sinus :

$$e^{j\theta} = \cos \theta + j \sin \theta$$

Pour les signaux précédents, avec  $\theta = 2 \pi f_0 t$ , nous écrivons :

$$\cos (2 \pi f_0 t) = (e^{j (2 \pi f_0 t)} + e^{-j (2 \pi f_0 t)}) / 2 \text{ et } \sin (2 \pi f_0 t) = (e^{j (2 \pi f_0 t)} - e^{-j (2 \pi f_0 t)}) / 2j$$

Même si cette écriture permet d'étendre aux signaux à valeurs complexes, nous nous limiterons à ceux à valeurs réelles.

Représenter un sinus dans une représentation fréquentielle oblige à introduire la partie négative de l'axe des fréquences puisque le terme  $-j (2 \pi f_0 t)$  intervenant dans l'exponentielle est à considérer comme :  $j (2 \pi (-f_0) t)$



# Signal et Image

En adoptant de représenter un signal sinusoïdal par une exponentielle complexe, plusieurs représentations sont possibles : les couples partie réelle / partie imaginaire ou **module / argument**. C'est cette dernière représentation la plus utilisée.

Remarques :

- \* un cosinus ne se distingue d'un sinus que par la phase puisque  $\cos(\alpha) = \sin(\alpha + \pi/2)$ . On adopte souvent la fonction cos pour représenter un signal sinusoïdal.
- \* lorsque  $T_0$  augmente, la fonction  $a \cos(2\pi f_0 t)$  tend vers un signal constant de valeur  $a$ . Observons la représentation fréquentielle de l'exponentielle complexe correspondante :



La représentation d'un signal sinusoïdal fait intervenir un couple de fréquence  $f_0$  et  $-f_0$ . C'est bien l'association des 2 fréquences, l'une positive (une exponentielle complexe  $e^{j2\pi f_0 t}$ ) et l'autre négative (issue de  $e^{-j2\pi f_0 t}$ ), qui permet de reconstruire une fréquence réelle sinusoïdale par leur somme.

Il est incorrect de dire que seule la positive a une existence réelle et l'autre est artificielle.



# Signal et Image

Somme de signaux sinusoïdaux et signaux périodiques :

La représentation d'un signal sinusoïdal ne fait intervenir qu'une fréquence  $f_0$  (et  $-f_0$ ) et une somme de signaux sinusoïdaux ne fera intervenir que des valeurs particulières pour former un *spectre de raies*.

On peut se poser la question suivante : quels sont les signaux qui peuvent se construire grâce à une somme de sinusoïdes ?

La réponse est due à Joseph Fourier : tous les signaux périodiques physiquement réalisables, c'est-à-dire dont la période est physiquement réalisable (les autres périodes étant obtenues par répétition).

Ses conclusions permettent d'écrire qu'un signal de période  $T_0$  peut se décomposer de façon unique sous la forme d'une somme infinie de signaux sinusoïdaux de fréquences multiples de  $f_0 = 1/T_0$ .

$$s_{T_0}(t) = \sum_{n=0}^{\infty} \alpha_n \cdot \cos(2\pi n f_0 t - \varphi_n)$$

En utilisant une propriété des fonctions trigonométriques, on peut ré-écrire sous la forme :

$$s_{T_0}(t) = \sum_{n=0}^{\infty} a_n \cdot \cos(2\pi n f_0 t) + b_n \cdot \sin(2\pi n f_0 t)$$

avec  $a_n = \alpha_n \cos(\phi_n)$  et  $b_n = \alpha_n \sin(\phi_n)$





# Signal et Image

Cette expression correspond à la *décomposition en série de Fourier*.

Les fonctions sin et cos utilisées constituent une base au sens des espaces vectoriels du sous-ensemble des fonctions périodiques de période  $T_0$ .

Une fonction périodique de période  $T_0$  s'écrit donc sous la forme d'une combinaison linéaire des fonctions de cette base. Les coefficients de la décomposition sont les coefficients de pondération de cette combinaison linéaire.

Les fonctions exponentielles  $e^{j2\pi n f_0 t}$  forment une autre base du même sous ensemble.

Ainsi, on peut aussi écrire :

$$s_{T_0}(t) = \sum_{n=-\infty}^{\infty} c_n \cdot e^{j2\pi n f_0 t}$$

Les coefficients  $c_n$  sont appelés *coefficients de Fourier*.

Dans les espaces vectoriels, nous avons coutume d'utiliser des bases de vecteurs orthonormés.

Ces notions sont liées à la définition d'un produit scalaire. Or, il est possible de définir un produit scalaire entre 2 fonctions dans l'ensemble des fonctions périodiques de période  $T_0$  par :

$$\langle s_n(t), s_m(t) \rangle = \frac{1}{T_0} \int_a^{a+T_0} s_n(t) \cdot \overline{s_m(t)} dt$$



# Signal et Image

Si ce produit scalaire est nul, alors  $s_n$  et  $s_m$  sont orthogonales sur l'intervalle  $]a, a+T_0[$  de largeur  $T_0$ .

Il est possible de démontrer que l'ensemble E des fonctions exponentielles de fréquences multiples d'une même fréquence  $f_0$  forme un ensemble de fonctions orthogonales entre elles sur tout intervalle de largeur  $T_0$ . Il suffit de voir que :

$$\langle s_n(t), s_m(t) \rangle = \frac{1}{T_0} \int_{-T_0/2}^{+T_0/2} \exp\left(j(n-m)\frac{2\pi}{T_0}t\right) dt = \frac{1}{T_0} T_0 \frac{\sin((n-m)\pi)}{(n-m)\pi}$$

$$\text{si } n \neq m, \langle s_n(t), s_m(t) \rangle = 0, \text{ sinon } \langle s_n(t), s_m(t) \rangle = 1$$

Ainsi les fonctions exponentielles complexes de E forme une base orthogonale du sous-ensemble des fonctions périodiques de période  $T_0$ .

Pour calculer les coefficients  $c_n$ , il suffit de reprendre l'expression

$$s_{T_0}(t) = \sum_{n=0}^{\infty} c_n \cdot e^{2\pi j n f_0 t}$$

Puis de multiplier chaque membre de l'équation par  $\overline{e^{2\pi j m f_0 t}} = e^{-2\pi j m f_0 t}$

On intègre alors les 2 membres sur un intervalle de largeur  $T_0$  et on échange l'ordre de l'intégrale et de la somme. Le seul terme non nul du membre de droite s'obtient pour  $n=m$ . On obtient :

$$c_n = (1/T_0) \int_a^{a+T_0} s_{T_0}(t) \cdot e^{-j2\pi m f_0 t} dt$$





# Signal et Image

Il existe un certain nombre de conditions suffisantes qui garantissent l'existence des coefficients  $C_n$ .

Par exemple, il suffit que les fonctions soient *de carré-intégrable*, c'est-à-dire :

$$\int_a^{a+T_0} |s_{T_0}(t)|^2 dt < +\infty$$

Une période d'un signal de carré-intégrable contient une énergie finie. C'est le cas de tous les signaux périodiques physiquement réalisables.

## Signaux monodimensionnels quelconques et Transformée de Fourier

Pour généraliser les résultats précédents aux signaux quelconques, il suffit de considérer qu'un signal non périodique est un signal périodique de période infinie. Si nous réécrivons l'équation précédente par :

$$T_0 c_n = \int_a^{a+T_0} s_{T_0}(t) \cdot e^{-j2\pi n f_0 t} dt$$





# Signal et Image

Quand  $T_0$  tend vers l'infini,  $f_0$  tend vers 0. Intéressons nous au second membre de l'équation précédente :

$$\int_a^{a+T_0} s_{T_0}(t) \cdot e^{-j2\pi f_0 t} dt$$

Quand  $f_0$  tend vers 0, le spectre de raies d'abscisses  $nf_0$  devient un spectre continu. Les fréquences discrétisées  $nf_0$  constituent l'axe des fréquences  $f$ .

On définit :

$$S(f) = \int_{-\infty}^{+\infty} s(t) \cdot e^{-2\pi j f t} dt$$

comme la **transformée de Fourier** du signal  $s(t)$ .

La transformée de Fourier inverse s'écrit :

$$s(t) = \int_{-\infty}^{+\infty} S(f) \cdot e^{+2\pi j f t} df$$

Il faut bien voir que  $s(t)$  et  $S(f)$  sont 2 représentations du même signal, dans 2 espaces différents, temporel et fréquentiel.





# Signal et Image

Comme pour les coefficients de Fourier, notons que la condition de carré-intégrabilité est une condition suffisante pour l'existence de la transformée.

Tous les signaux physiquement réalisables possèdent une transformée de Fourier.

Résultats importants :

Un **signal pair** à valeurs **réelles** a un **spectre réel et pair**.

Un signal **impair** à valeurs **réelles** a un spectre **impair et imaginaire pur**.

Pour tous les signaux, on a plusieurs représentations possibles :

$$S(f) = \operatorname{Re}(S(f)) + j \operatorname{Im}(S(f)) = |S(f)| e^{j\phi(f)}$$

$$\phi(f) = \arg(S(f))$$

On préfère les représentations du spectre d'amplitude  $|S(f)|$  et du spectre de phase  $\phi(f)$ .

Une représentation très utilisée aussi est celle du spectre d'énergie  $|S(f)|^2$ .





# Signal et Image

Quelques propriétés intéressantes :

$$y(t) = \overline{x(t)} \Rightarrow Y(f) = \overline{X(-f)}$$

$$y(t) = x(t - t_0) \Rightarrow Y(f) = e^{-j2\pi f t_0} X(f)$$

$$y(t) = x(t) * z(t) = \int_{-\infty}^{+\infty} x(t - \tau) z(\tau) d\tau \Rightarrow Y(f) = X(f) Z(f)$$

$$y(t) = x(t) z(t) \Rightarrow Y(f) = X(f) * Z(f)$$

$$y(t) = \frac{d^n x(t)}{dt^n} \Rightarrow Y(f) = (j2\pi f)^n X(f)$$



# Signal et Image

## Discrétisation des signaux par échantillonnage

La manipulation informatique des signaux impose leur transformation en valeurs numériques. Cette discrétisation se fait généralement par prélèvement d'échantillons régulièrement espacés d'une durée  $T_e$ , période d'échantillonnage.

Nous pouvons modéliser cette opération en multipliant le signal  $s(t)$  par un *peigne de Dirac* :

$$s_e(t) = s(t) \cdot \sum_{n=-\infty}^{+\infty} \delta(t - nT_e) = \sum_{n=-\infty}^{+\infty} s(nT_e) \cdot \delta(t - nT_e)$$

La transformée de Fourier du signal échantillonné s'écrit :

$$S(f) = \sum_{n=-\infty}^{+\infty} s(nT_e) e^{-j2\pi f n T_e}$$

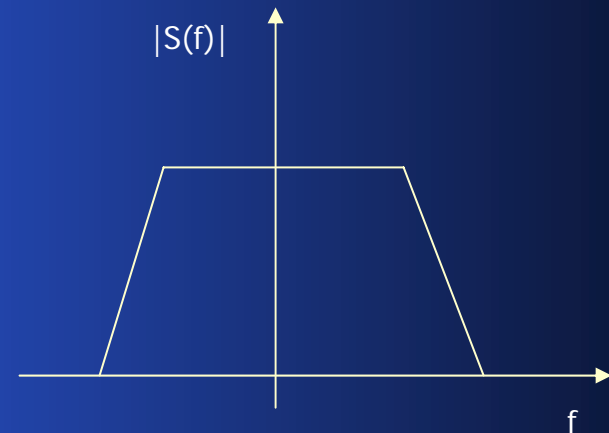
De manière évidente,

$$S(f + f_e) = \sum_{n=-\infty}^{+\infty} s(nT_e) e^{-j2\pi (f + f_e) n T_e} = \sum_{n=-\infty}^{+\infty} s(nT_e) e^{-j2\pi f n T_e} e^{-j2\pi f_e n T_e} = S(f)$$

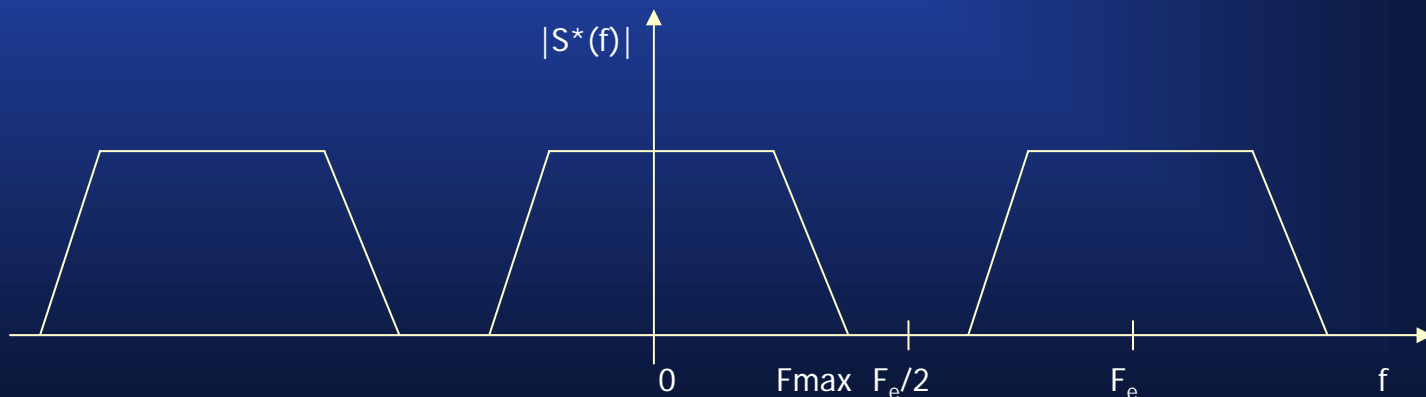
Ainsi, la transformée de Fourier d'un signal échantillonné est périodique, de période  $f_e = 1/T_e$

# Signal et Image

Spectre  $S(f)$  du signal  $s(t)$  :  
représentation du spectre d'amplitude  
Supposons également que  $f_e$  soit  
supérieure à  $2 f_{\max}$ .



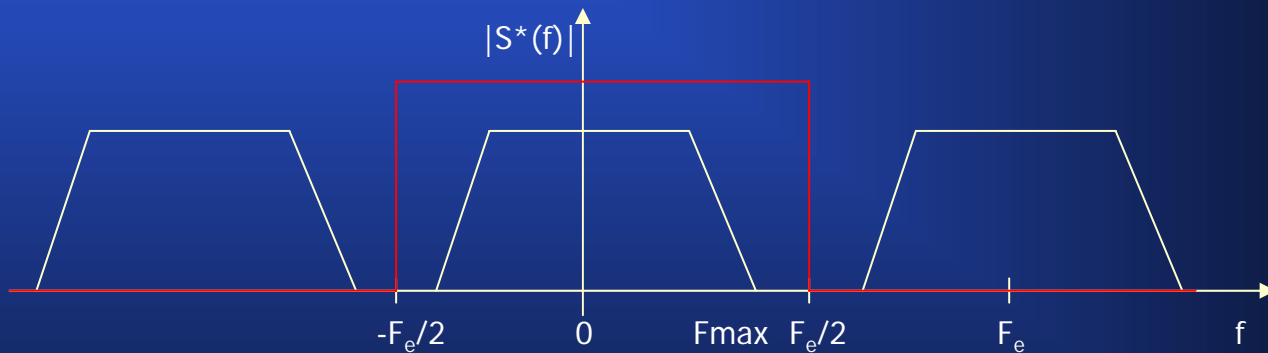
On sait que le spectre de  $s_{Te}(t)$  est périodique, avec une période égale à  $f_e$  : chaque période reproduit exactement le spectre de  $s(t)$ .



# Signal et Image

## Aliasing et recouvrement de spectre

Pour reconstituer le signal original, il faut interpoler, c'est-à-dire déterminer les valeurs manquantes entre les échantillons. Les spectres de  $s(t)$  et  $s_{Te}(t)$  sont identiques dans la période principale, donc, si on élimine les autres périodes, on obtient le spectre de  $s(t)$ . Cette opération peut se réaliser à l'aide d'un filtre passe-bas de fréquence de coupure égale à  $f_e/2$ , appelé *filtre cardinal*. Le signal obtenu en sortie du filtre est bien  $s(t)$ , puisqu'il a le même spectre que lui.

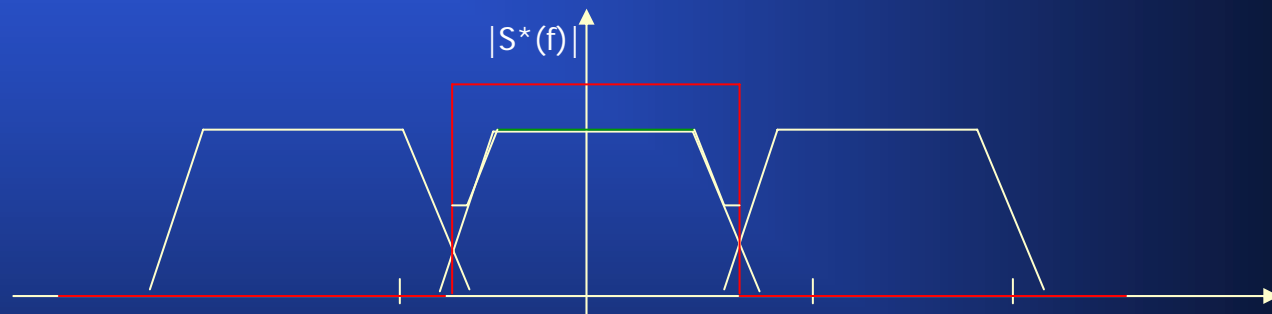


# Signal et Image

## Théorème de Shannon

On ne retrouve  $s(t)$  que si le spectre de  $s(t)$  est borné et si  $f_e > 2 f_{\max}$ . Si cette condition n'est pas respectée, dans le spectre de  $s_{Te}(t)$ , les différentes « périodes » se chevauchent et dans la période principale, le spectre n'est pas celui de  $s(t)$ . Après passage dans le filtre cardinal, le signal obtenu n'est donc pas  $s(t)$ .

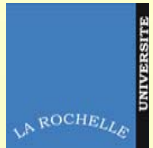
La condition  $f_e > 2 f_{\max}$  constitue le théorème de Shannon ou théorème d'échantillonnage:  
*La fréquence d'échantillonnage doit être supérieure au double de la plus haute fréquence contenue dans le signal à échantillonner.*



Si cette condition n'est pas respectée, la partie du spectre de  $s(t)$  de fréquence supérieure à  $f_e/2$  se retrouve, après échantillonnage, au-dessous de  $f_e/2$ , comme le montre la figure précédente. Ce phénomène s'appelle *le repliement du spectre*. C'est comme si on avait replié la partie supérieure du spectre de  $s(t)$  autour de la verticale  $f = f_e/2$

(en anglais *aliasing* du fait des *alias* entre les périodes)





# Signal et Image

## Filtre anti-repliement :

Le respect du théorème de Shannon impose d'utiliser une fréquence  $f_e$  au moins égale à  $2 f_{max}$ . Il est donc nécessaire de connaître le contenu fréquentiel du signal d'entrée. La fréquence d'échantillonnage résultante est souvent trop importante et incompatible avec les gabarits des traitements fréquentiels envisagés.

Il faut alors envisager un filtrage passe-bas en début de la chaîne de traitement. Ce filtre est appelé filtre *anti-repliement* ou *anti-aliasing*.

Il est souvent de nature analogique.

Réalisé de manière numérique, il doit respecter la fréquence de Shannon (même pour un passe-bas !) conduisant à un « sur-échantillonnage ».

## Echantillonnage fréquentiel

Si on échantillonne cette fois un spectre  $S(f)$  (continu) tous les  $\Delta f$  en fréquence, les conséquences seront semblables : une périodisation temporelle du signal d'origine (avec une période  $1/\Delta f$ ) et un repliement éventuel si le signal possède des valeurs non nulles sur un intervalle de longueur supérieure à  $\Delta T = 1/\Delta f$

On peut exprimer cela par le théorème de Shannon *pour l'échantillonnage fréquentiel* :

*Soit  $s(t)$  un signal limité dans le temps à  $|t| < T_{max}$ .*

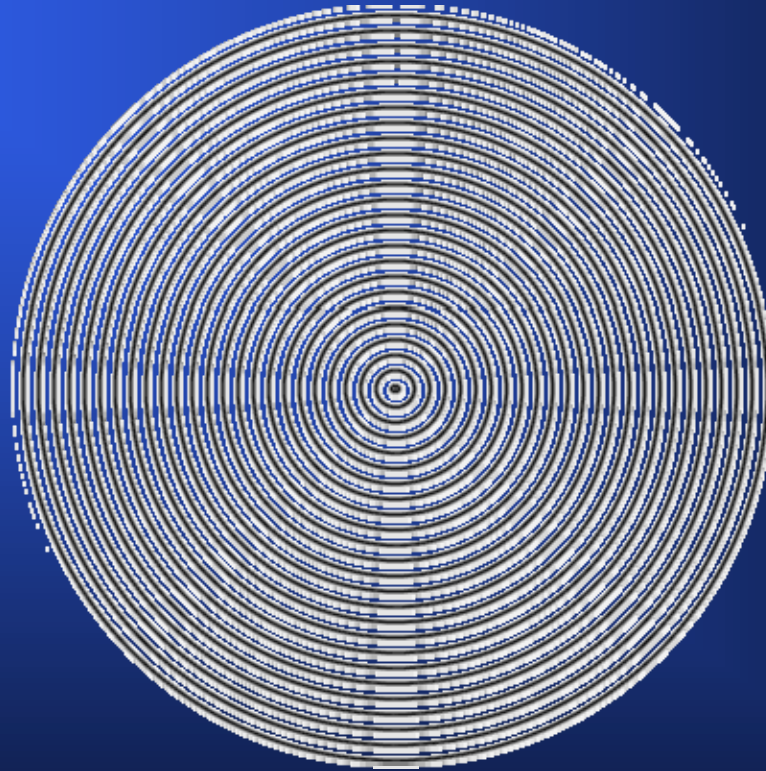
*La transformée de Fourier inverse d'un spectre  $S(f)$  échantillonné tous les  $f_0$  sera une répétition périodique de  $s(t)$  avec une période  $T_0 = 1/f_0$ . Il n'y a pas de recouvrement temporel (aliasing temporel) si et seulement si :*

$$T_0 > 2 T_{max}$$



# Signal et Image

Exemple d'aliasing en dimension 2 :





# Signal et Image

Transformée de Fourier à temps discret (TFTD) :

On pose traditionnellement  $T_e=1$ .

Il s'agit du produit scalaire entre le signal  $x(n)$  et l'exponentielle complexe  $e^{j2\pi fn}$

Cette transformée de Fourier appliquée aux signaux discrets est une fonction **à fréquence continue**, périodique de période 1.

Il est d'usage de la représenter sur un intervalle de longueur 1, de  $-1/2$  à  $+1/2$

$$S(f) = \sum_{n=-\infty}^{+\infty} s(n) e^{-j2\pi fn}$$

$$s(n) = \int_{-1/2}^{+1/2} S(f) e^{+j2\pi fn} df$$

L'impulsion a pour TFTD, 1. L'échelon unité n'a pas de TFTD au sens des fonctions (mais au sens des distributions). La porte rectangulaire de largeur  $N$  et débutant à  $n=0$  a pour TFTD :

$$G_N(f) = \sum_{n=0}^{N-1} e^{-j2\pi fn} = \frac{\sin(\pi f N)}{(\pi f)} e^{-j\pi f}$$





# Signal et Image

Ce résultat permet d'expliquer les ondulations parasites qui apparaissent lors de la mesure pratique des spectres des signaux échantillonnés, qui revient à multiplier le signal par une fenêtre rectangulaire. Le spectre obtenu est le résultat de la convolution du spectre du signal par  $G_N(f)$ .

La **résolution en fréquence** (appelée *limite de Fourier*) est de l'ordre de la largeur des lobes de cette fonction :  $2/N$ .

Une « fréquence » d'amplitude faible au voisinage d'une d'amplitude plus élevée sera masquée par le premier lobe secondaire.

La séparation dans ce cas peut être améliorée par l'emploi de **fenêtres de pondération** (Hamming...).

Mais c'est au détriment de la séparation de « fréquences » très voisines mais d'amplitude semblables car les 2 raies seront confondues dans un lobe principal élargi par la fenêtre (la fenêtre rectangulaire a le lobe principal le plus étroit de toutes les fenêtres).

La **précision** est la capacité à trouver la bonne fréquence d'un signal. Elle est limitée par le nombre de points utilisés dans le calcul de la TFTD qui se fera par une Transformée de Fourier Discrète (voir plus loin).



# Signal et Image

## Transformée de Fourier à Court Terme (TFCT) :

Considérons un signal simple mais qui serait composé de plusieurs fréquences *successives*. La TFTD contient effectivement l'information sur l'ordre chronologique dans lequel apparaissent ces fréquences, mais cette information est cachée dans le spectre de phase, inexploitable.

Si on divise la durée totale du signal en sous-intervalles de même durée, on peut calculer plusieurs spectres qui affichés côte à côte, donne la représentation appelée Transformée de Fourier à Court Terme.

Le module au carré de la TFCT est aussi appelé *spectrogramme*.

En réduisant la durée des intervalles, pour une meilleure précision temporelle, les spectres sont calculés avec moins de points, et donc la précision fréquentielle diminue puisque les lobes des ondulations s'élargissent.

Le défaut de la TFCT est d'utiliser des fenêtres qui masquent les périodicités longues, et donc les fréquences basses.

L'idée de *l'analyse multi résolution* (ou *multi-échelle*) est d'échantillonner à des cadences différentes le signal de manière à effectuer l'étude à plusieurs niveaux de résolution.



# Signal et Image

La transformée de Fourier fenêtrée ou TFTC a une **résolution temps-fréquence fixe**. Cette résolution peut être modifiée par un changement d'échelle sur la fenêtre  $g$ . C'est une représentation complète, stable et redondante du signal. Elle est donc inversible.

Nous verrons plus tard une alternative :

## La Transformée en ondelettes

La transformée en ondelettes remplace la sinusoïde de la transformée de Fourier par une famille de translations et dilatations d'une même fonction, l'ondelette.

Les paramètres de translation et de dilatation sont les deux arguments de la transformée en ondelettes.

La transformée en ondelettes a donc **une résolution temps-fréquence qui dépend de l'échelle  $s$** .

Un **scalogramme** remplace le spectrogramme.

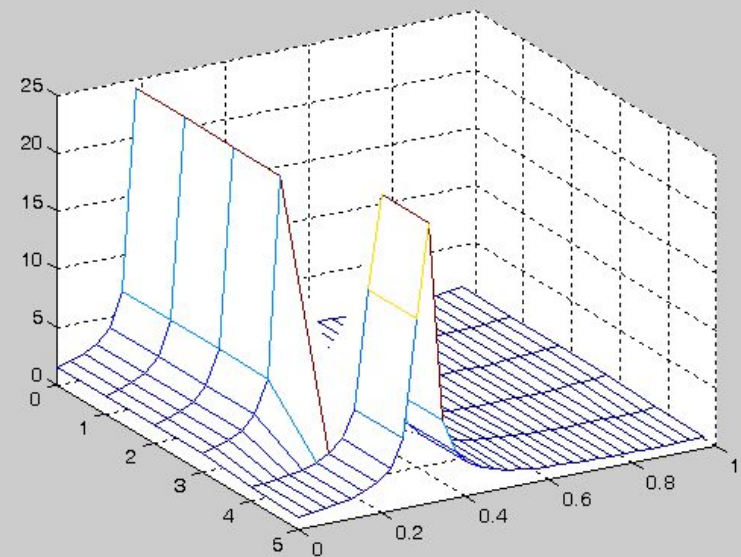
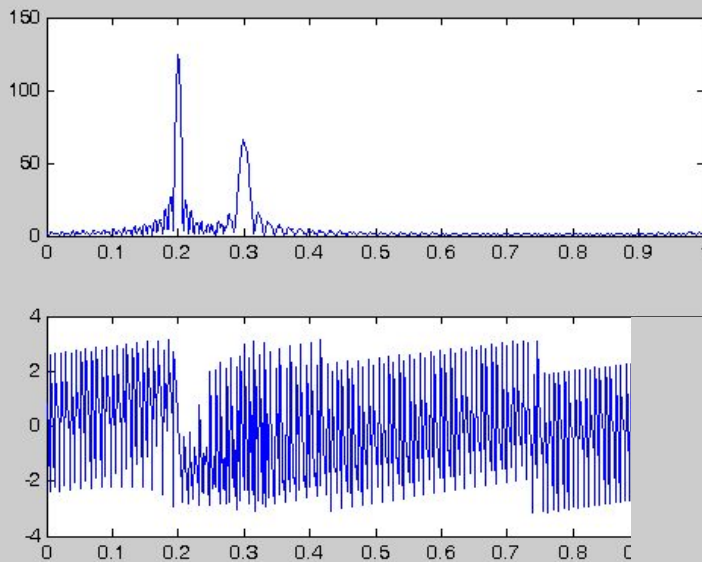
La transformée en ondelettes se calcule par une transformée en ondelettes rapide. Celle-ci effectue une transformée discrète par des *convolutions circulaires*, elles-mêmes calculées par *FFT*. Toutes ces notions vont être abordées un peu plus loin.

Les *ondelettes dyadiques* sont des ondelettes dont la dilatation vérifie une propriété spécifique permettant d'implémenter les transformées par des **bancs de filtres**.

Un **film** illustrant l'approche multirésolution en image...



# Signal et Image







# Signal et Image

## La Transformée de Fourier Discrète monodimensionnelle (TFD ou TFD-1D)

Cette transformée, popularisée par son calcul rapide (TFR ou FFT : Fast Fourier Transform), fait correspondre une suite de  $N$  valeurs à une autre de suite de  $N$  valeurs numériques également.

On considère un signal numérique défini par  $N$  échantillons temporels, obtenus par échantillonnage avec la période  $T_e$ .

Sa transformée de Fourier  $S(f)$  est donc périodique de période  $f_e = 1/T_e$ .

Choisissons  $N$  échantillons de  $S(f)$  dans sa période, ainsi  $S(f)$  est échantillonnée à la cadence  $f_0 = f_e/N = 1/NT_e$ .

Ce dernier résultat entraîne une périodicité du signal temporel de  $NT_e$ .

Le seuls signaux à être interprétés sans erreur, seront les signaux périodiques (période  $NT_e$ ), échantillonnés suffisamment vite ( $T_e$ ) pour respecter le théorème d'échantillonnage temporel, et observés sur un nombre entiers de périodes (1 période =  $NT_e$ ). Dans ce cas la TFD s'assimile à la Transformée de Fourier du signal.

Un signal de durée limité pourra être étudié correctement : il suffit de le transformer en signal périodique de période  $NT_e$ .







# Signal et Image

## Expression de la TFD-1D

Comme précédemment, on utilise une décomposition sur une base de fonctions orthogonales.

Les fonctions utilisées sont ici les exponentielles complexes discrètes d'ordre  $N$  (encore appelées  $N$ èmes racines de l'unité).

$$W_N^n = e^{-j\frac{2\pi n}{N}}$$

On peut d'ailleurs facilement montrer leur orthogonalité :

Soit  $u$  et  $v$ , deux entiers tels que :

$$0 \leq u \leq N - 1 \quad \text{et} \quad 0 \leq v \leq N - 1$$

On montre facilement :

$$\frac{1}{N} \sum_{n=0}^{N-1} e^{-j\frac{2\pi un}{N}} \overline{e^{-j\frac{2\pi vn}{N}}} = \begin{cases} 0 & \text{pour } u \neq v \\ 1 & \text{pour } u = v \end{cases}$$

Ceci définit un produit scalaire. On peut ainsi décomposer les signaux numériques sur le même principe que la décomposition en série de Fourier.





# Signal et Image

## Expression de la TFD 1D (suite)

Soit  $s$  un signal numérique défini par  $N$  échantillons  $s(n)$ , notés  $s_n$ , avec  $0 \leq n \leq N-1$

La TFD de la suite  $s_n$  s'écrit :

$$S_k = \sum_{n=0}^{N-1} s_n e^{-j \frac{2\pi nk}{N}}$$

La TFD inverse (TFDI) se calcule à partir de :

$$s_n = \frac{1}{N} \sum_{k=0}^{N-1} S_k e^{j \frac{2\pi nk}{N}}$$

Il faut bien remarquer que l'on perd toute référence aux valeurs des instants correspondant aux échantillons. Nous avons une relation entre une suite indexée par une variable entière  $n$  et une suite indexée par  $k$ .

$s$  est périodique de période  $N$  et  $S$  est périodique de période  $N$ . Vérifions le :

$$s_{n+N} = \frac{1}{N} \sum_{k=0}^{N-1} S_k e^{j \frac{2\pi(n+N)k}{N}} = \frac{1}{N} \sum_{k=0}^{N-1} S_k e^{j \frac{2\pi nk}{N}} e^{j \frac{2\pi Nk}{N}} = s_n$$





# Signal et Image

## Expression usuelle

On préfère une représentation fréquentielle centrée autour de la fréquence nulle. Dans ce cas les expressions deviennent :

La **TFD** de la suite  $s_n$  s'écrit :

$$S_k = \sum_{n=0}^{N-1} s_n e^{-j \frac{2\pi nk}{N}}$$

La **TFD inverse** (TFDI) se calcule à partir de :

$$s_n = \frac{1}{N} \sum_{k=-(N-1)/2}^{+(N-1)/2} S_k e^{j \frac{2\pi nk}{N}} \quad \text{symétrique /0 quand N est impair.}$$

La précision fréquentielle dépend du nombre de points adoptés pour le calcul. Les points en fréquences, sont espacés de  $1/N$ .

Cette précision est améliorée par la technique du **zero-padding** : on calcule la TFD sur un nombre  $N$  pouvant être largement supérieur au nombre de points disponible du signal.



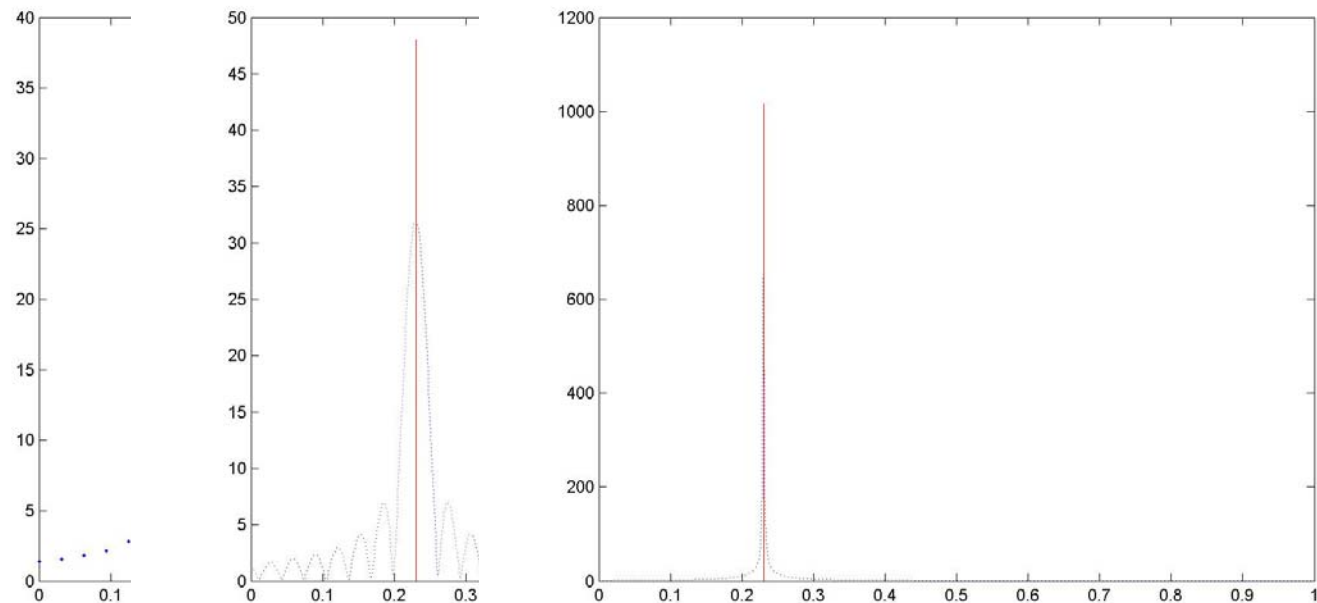
# Signal et Image

## Précision et résolution

$N=32$  points d'un signal sinusoïdal, TFD sur  $N$  points

$N=32$  points du même signal, TFD sur 1024 points (zero-padding)

$N=1024$  points du même signal, TFD sur 1024 points



# Signal et Image

## Expression d'une TFD-2D

Les images sont des signaux numériques à 2 dimensions, mais ces dimensions ne font plus références au temps : ce sont des dimensions spatiales, en hauteur et largeur.

Considérons une image de dimension  $N \times M$ .

Les expressions précédentes peuvent être étendues à ce cas 2D :

$$S_{u,v} = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} s_{n,m} e^{-j2\pi(\frac{nu}{N} + \frac{mv}{M})} \quad \text{pour } -\frac{N}{2} \leq \{u \text{ et } v\} \leq \frac{N}{2} - 1$$

Et la transformée de Fourier discrète bidimensionnelle inverse :

$$s_{n,m} = \frac{1}{NM} \sum_{u=-N/2}^{+N/2-1} \sum_{v=-N/2}^{+N/2-1} S_{u,v} e^{j2\pi(\frac{un}{N} + \frac{vm}{M})} \quad \text{quand } N \text{ est pair}$$

pour  $0 \leq n \leq N-1$  et  $0 \leq m \leq M-1$

Le spectre est à valeurs complexes. Il faut donc 2 graphiques pour le représenter : le spectre d'amplitude et le spectre de phase sont représentés de manière séparée.

Le spectre de phase est souvent négligé car difficilement interprétable.

Le spectre d'amplitude est généralement représentée avec la fréquence nulle au centre. Sa valeur au centre ( $S_{0,0}$ ) représente la moyenne des valeurs des échantillons de l'image.

# Signal et Image

## Remarques

- \* Le spectre de phase est souvent négligé alors qu'il contient l'information sur le contenu de l'image;

En effet, le spectre d'amplitude suit une loi décroissante quand  $u$  et  $v$  augmentent (c'est-à-dire quand  $(u^2 + v^2)^{1/2}$  augmente), et cela quelque soit l'image.

Une expérience intéressante consiste à remplacer le spectre d'amplitude par tout autre loi décroissante.

L'image est préservée dans son ensemble !!

- \* Le spectre d'amplitude a une dynamique très importante. Représentée sur une échelle linéaire, il est peu visible (point blanc au centre, noir sur le reste de l'image). On lui préfère des représentations utilisant des échelles logarithmiques du type  $\log(1 + |S(u,v)|)$ .

- \* La transformée d'une image de dimension  $N \times M$  est une image de dimension  $N \times M$ .

Son centrage (origine des fréquences au centre de l'image) peut se faire par permutation des cadrans une fois la transformée effectuée, ou bien au préalable avant son calcul.

Dans ce dernier cas, pour obtenir une translation dans le domaine des fréquences, il faut multiplier par une exponentielle complexe dans le domaine espace. En pratique, on multiplie chaque pixel  $(i,j)$  par  $(-1)^{i+j}$ .

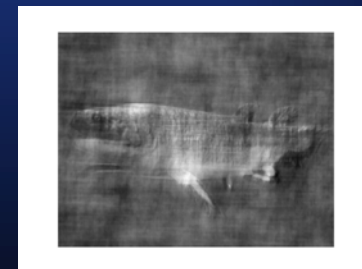
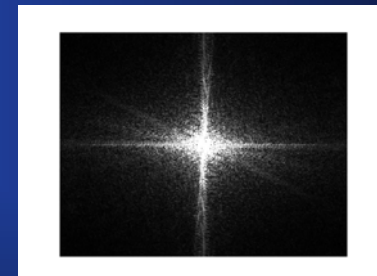
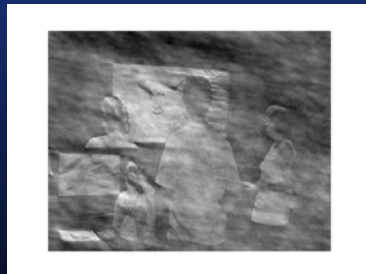
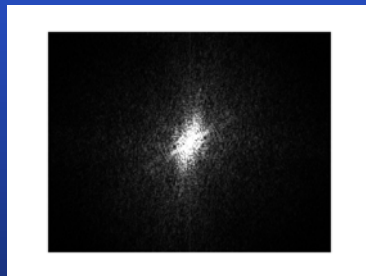
- \* Les images étant à valeurs réelles, le spectre vérifie :  $S_{u,v} = \overline{S_{-u,-v}}$

Le spectre d'amplitude vérifie :  $|S_{u,v}| = |S_{-u,-v}|$

et le spectre de phase vérifie :  $\phi_{u,v} = -\phi_{-u,-v}$

# Signal et Image

Expérience : remplaçons le spectre de phase d'une image par celui d'une autre image



# Signal et Image

## Calcul de la TFD 2D

Par analogie avec les signaux temporels, la transformée de Fourier d'une image numérique peut être notée TFTD-2D, et de la même façon, la Transformée de Fourier Discrète est notée TFD-2D, et se calcule par :

$$S_{u,v} = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} s_{n,m} e^{-j2\pi(\frac{nu}{N} + \frac{mv}{M})}$$

On voit, qu'en récrivant cette expression sous la forme suivante,

$$S_{u,v} = \sum_{n=0}^{N-1} \left( e^{-j2\pi \frac{nu}{N}} \sum_{m=0}^{M-1} s_{n,m} e^{-j2\pi \frac{mv}{M}} \right)$$

on fait apparaître dans la parenthèse, pour chaque valeur de  $n$ , la TFD 1D de la suite  $s_{n,m}$  le long de la variable  $m$ .

Remarque : sous MATLAB, la TFD-2D s'obtiendrait, en faisant appel à la fonction FFT par `fft(fft(s).')`. C'est ce que fait la fonction `fft2(x,M,N)`.

Lorsque  $s_{n,m}$  est séparable, c'est-à-dire  $s_{n,m} = s1_n \cdot s2_m$ , la TFD-2D devient un produit de 2 TFD-1D :

$$S_{u,v} = S1_u \cdot S2_v$$

Le calcul se simplifie donc largement.





# Signal et Image

## le calcul de la FFT ou TFR

La TFD est restée un outil peu utilisée jusqu'à l'apparition d'algorithmes « rapides » permettant son calcul.

Le plus connu est dû à Cooley et Tuckey et date de 1965.

Le calcul direct de la TFD sur N points nécessite  $2N^2$  multiplications et  $2N(N-1)$  additions.

L'algorithme proposé réduit à  $N \log_2(N)$  le nombre d'opérations.

Sans nuire à la généralité, prenons le cas de  $N=8$ .

Il faut calculer :

$$S_k = \sum_{n=0}^{N-1} s_n e^{-j \frac{2\pi nk}{N}} = \sum_{n=0}^7 s_n e^{-j \frac{2\pi nk}{N}}$$

Introduisons  $W_N$ , la racine n-ième de l'unité (appelée « twiddle factor » en anglais) :

$$W_N = \exp\left(-j \frac{2\pi}{N}\right)$$

$$S_k = \sum_{n=0}^7 s_n W_N^{nk}$$



# Signal et Image

Pour  $N=8$ , explicitons la relation précédente :

$$\begin{bmatrix} S_0 \\ S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \\ S_7 \end{bmatrix} = \begin{bmatrix} W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 \\ W_8^0 & W_8^1 & W_8^2 & W_8^3 & W_8^4 & W_8^5 & W_8^6 & W_8^7 \\ W_8^0 & W_8^2 & W_8^4 & W_8^6 & W_8^8 & W_8^{10} & W_8^{12} & W_8^{14} \\ W_8^0 & W_8^3 & W_8^6 & W_8^9 & W_8^{12} & W_8^{15} & W_8^{18} & W_8^{21} \\ W_8^0 & W_8^4 & W_8^8 & W_8^{12} & W_8^{16} & W_8^{20} & W_8^{24} & W_8^{28} \\ W_8^0 & W_8^5 & W_8^{10} & W_8^{15} & W_8^{20} & W_8^{25} & W_8^{30} & W_8^{35} \\ W_8^0 & W_8^6 & W_8^{12} & W_8^{18} & W_8^{24} & W_8^{30} & W_8^{36} & W_8^{42} \\ W_8^0 & W_8^7 & W_8^{14} & W_8^{21} & W_8^{28} & W_8^{35} & W_8^{42} & W_8^{49} \end{bmatrix} \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{bmatrix}$$

Les facteurs  $W_n$  présentent un certain nombre de propriétés dont certaines sont mises à profit dans l'algorithme :

$$W_N^{nN} = 1, W_N^{N/2} = -1, W_N^{n+N} = W_N^n$$

Ainsi, l'équation précédente se réduit à :

# Signal et Image

$$\begin{bmatrix} S_0 \\ S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \\ S_7 \end{bmatrix} = \begin{bmatrix} W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 \\ W_8^0 & W_8^1 & W_8^2 & W_8^3 & W_8^4 & W_8^5 & W_8^6 & W_8^7 \\ W_8^0 & W_8^2 & W_8^4 & W_8^6 & W_8^0 & W_8^2 & W_8^4 & W_8^6 \\ W_8^0 & W_8^3 & W_8^6 & W_8^1 & W_8^4 & W_8^7 & W_8^2 & W_8^5 \\ W_8^0 & W_8^4 & W_8^0 & W_8^4 & W_8^0 & W_8^4 & W_8^0 & W_8^4 \\ W_8^0 & W_8^5 & W_8^2 & W_8^7 & W_8^4 & W_8^1 & W_8^6 & W_8^3 \\ W_8^0 & W_8^6 & W_8^4 & W_8^2 & W_8^0 & W_8^6 & W_8^4 & W_8^2 \\ W_8^0 & W_8^7 & W_8^6 & W_8^5 & W_8^4 & W_8^3 & W_8^2 & W_8^1 \end{bmatrix} \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{bmatrix}$$

L'algorithme suppose que N est pair : posons  $N=2P$ . Introduisons les 2 sous-suites de  $s_n$  en fonction de la parité de n.

$$u_n = \{s_{2n}\}_{n=0,\dots,P-1}$$

$$v_n = \{s_{2n+1}\}_{n=0,\dots,P-1}$$

On obtient ainsi :

$$S_k = \sum_{n=0}^{N-1} s_n W_N^{nk} = \sum_{i=0}^{P-1} u_i W_{2P}^{2ik} + \sum_{i=0}^{P-1} v_i W_{2P}^{(2i+1)k}$$

$$S_k = \sum_{i=0}^{P-1} u_i W_P^{ik} + W_N^k \sum_{i=0}^{P-1} v_i W_P^{ik}$$

$$S_k = U_k + W_N^k V_k$$

# Signal et Image

Par ailleurs :

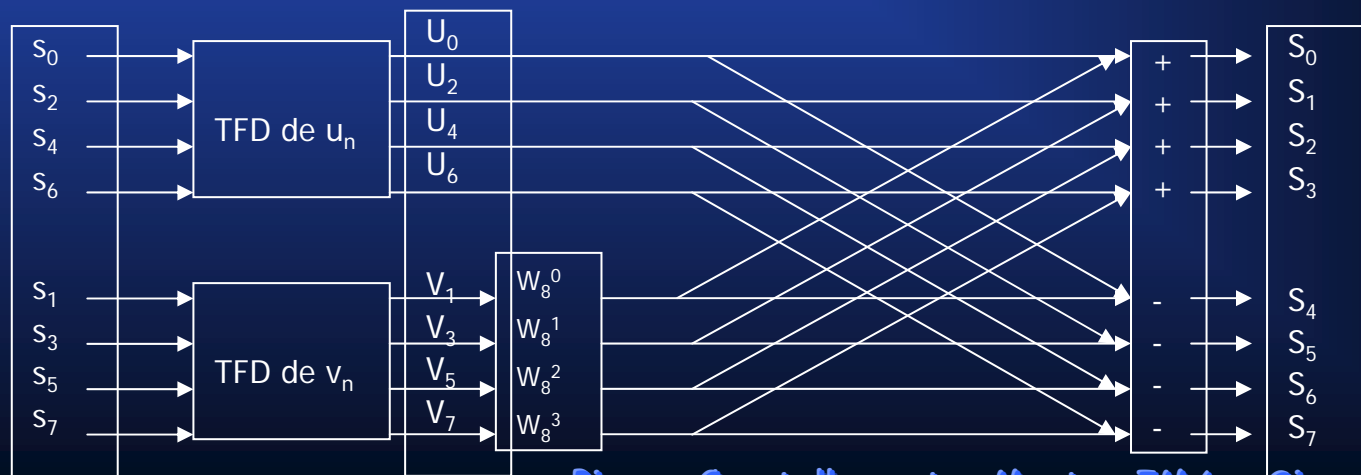
$$S_{(k+P)} = \sum_{n=0}^{N-1} s_n W_N^{(k+P)n} = \sum_{i=0}^{P-1} u_i W_{2P}^{2i(k+P)} + \sum_{i=0}^{P-1} v_i W_{2P}^{(2i+1)(k+P)}$$

$$S_{(k+P)} = \sum_{i=0}^{P-1} u_i W_P^{ik} W_P^{iP} + W_N^k W_N^P \sum_{i=0}^{P-1} v_i W_P^{ik} W_P^{iP}$$

$$S_{(k+P)} = \sum_{i=0}^{P-1} u_i W_P^{ik} \cdot 1 + W_N^k (-1) \sum_{i=0}^{P-1} v_i W_P^{ik} \cdot 1$$

$$S_{(k+P)} = U_k - W_N^k V_k$$

Le calcul de la FFT revient donc à calculer  $U_k$  et  $V_k$  qui sont les TFD sur P points des suites de termes de rang pair et impair. Le calcul revient au schéma suivant :



# Signal et Image

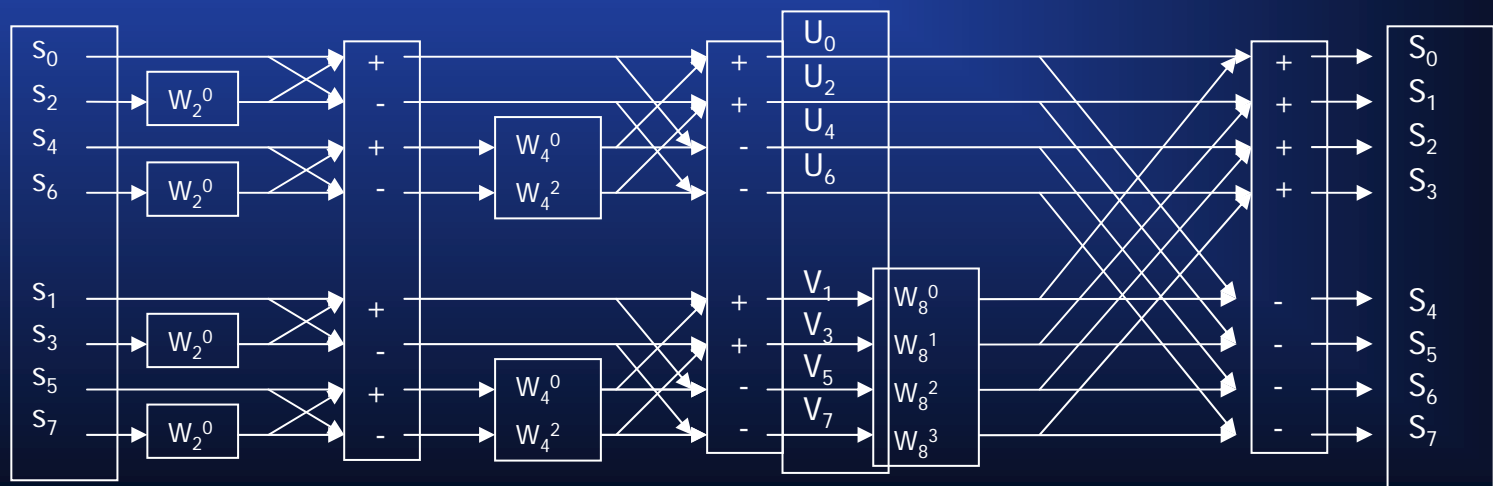
On s'aperçoit sur le schéma précédent qu'il ne reste qu'à exprimer les  $U_k$  et  $V_k$ . Or, ce sont des TFD sur  $P$  points, qui peuvent reprendre le même schéma que précédemment.

Cela est faisable si  $P$  est pair. On peut réitérer le processus à chaque sous-étage, si cette condition est à chaque fois vérifiée, donc si  $N$  au départ est une puissance de 2.

En pratique, lorsque cela n'est pas le cas, les suites d'échantillons sont complétées par des zéros jusqu'à la puissance de 2 immédiatement supérieure. (zero padding).

Il y a d'autres algorithmes qui ne présentent pas cette contrainte (mais en présentent d'autres).

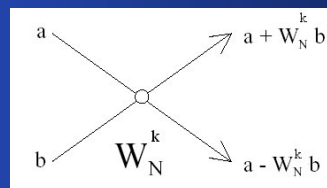
L'algorithme ainsi mis en œuvre présente des motifs à croisement appelé « algorithme papillon ». Notons l'entrelacement temporel qui « bouscule » l'ordre d'apparition des échantillons.



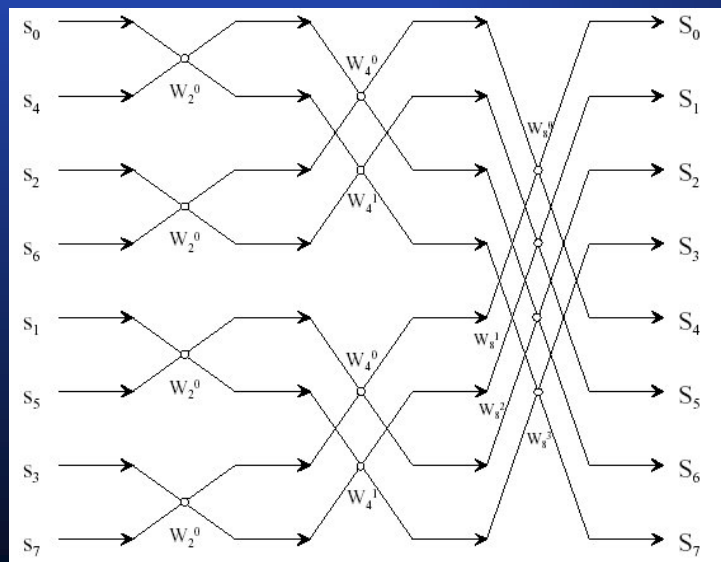
# Signal et Image

On s'aperçoit sur le schéma précédent que le nombre « d'étages » est bien en  $\log_2(N)$ , la complexité opératoire est de l'ordre de  $N \log_2(N)$ . (ou  $N/2 \log_2(N)$  multiplications complexes). Le rapport entre le nombre d'opérations nécessaires par cet algorithme et le calcul direct est :  $2N/\log_2(N)$ . Pour  $N=8$ , ce rapport est de 5.3, pour 512, il est de 113.7 et pour 4096, il est de 682.7. Nous reviendrons sur les conséquences de ce calcul.

L'algorithme repose sur l'utilisation d'un opérateur appelé papillon ou *butterfly* que nous retrouverons dans d'autres transformées :



Avec cette représentation, l'algorithme FFT se représente par :



# Signal et Image

L'algorithme de FFT peut s'écrire sous forme matricielle. On obtient :

$$\begin{bmatrix} X(0) \\ X(4) \\ X(2) \\ X(6) \\ X(1) \\ X(5) \\ X(3) \\ X(7) \end{bmatrix} = \begin{bmatrix} 1 & w^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -w^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & w^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -w^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & w^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -w^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & w^3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -w^3 \end{bmatrix} \mathbf{x}$$

$$\begin{bmatrix} 1 & 0 & w^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & w^0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -w^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -w^0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & w^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & w^2 \\ 0 & 0 & 0 & 0 & 1 & 0 & -w^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -w^2 \end{bmatrix} \mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & 0 & w^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & w^0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & w^0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & w^0 \\ 1 & 0 & 0 & 0 & -w^0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -w^0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -w^0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -w^0 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$

Notons aussi que l'algorithme présenté est dit à entrelacement temporel. Une version tout à fait symétrique et au même coût, opère les permutation sur les S et non les s : l'algorithme est dit à entrelacement fréquentiel.

# Signal et Image

La technique « Reverse Carry »

Comme on peut le voir sur la figure précédente, les couples d'échantillons doivent être choisis au départ selon un ordre particulier :  $s_0$ - $s_4$ ,  $s_2$ - $s_6$ , etc.

Cette incrémentation particulière est appelée "reverse carry" (retenue inverse). Elle consiste à additionner  $N/2$  à l'indice, mais à reporter la retenue à droite plutôt qu'à gauche.

Exemple avec  $N=8$  :

Indice (base 2)	Indice (base 10)
000	0
$000 + 100 = 100$	4
$100 + 100 = 010$	2
$010 + 100 = 110$	6
$110 + 100 = 001$	1
$001 + 100 = 101$	5
$101 + 100 = 011$	3
$011 + 100 = 111$	7

Les DSP proposent tous un mécanisme d'incrémentement de ce type pour leurs pointeurs.





# Signal et Image

## Variantes de l'algorithme FFT:

1) A partir des règles suivantes :

$$W_N^{2n} = W_{N/2}^n, \quad W_N^{n+N/2} = -W_N^n$$

on peut écrire un algorithme semblable pour  $N=k \cdot l$  avec  $k$  et  $l$  entiers, mais l'efficacité est moindre que lorsque  $N$  est une puissance de 2.

2) On peut travailler dans une base différente de 2, voire faire varier la base au cours du calcul (algorithme split-radix). En base 4 par exemple,  $N$  doit être une puissance de 4. La complexité est moindre.

3) Contre toute attente, quand  $N$  est premier, le calcul, loin d'être irréductible, est très performant. Il s'agit de l'algorithme de Rader.

Remarque : quand le facteur d'échelle  $1/N$  est utilisé, il peut être effectué par une division par 2 à chaque étage de l'algorithme.



# Signal et Image

## TFD et convolution discrète : peut-on effectuer un filtrage linéaire par FFT ?

Position du problème : considérons un signal *de durée infinie* devant être filtré continûment par un filtre RIF.

On rappelle qu'un filtre RIF s'écrit :

$$y_n = \sum_{i=0}^{N-1} h_i x_{n-i}$$

Dans le domaine fréquentiel, cette convolution conduit à la multiplication de  $X(f)$  par la réponse fréquentielle du filtre  $H(f)$ .

D'où l'idée suivante : compte tenu de la performance de l'algorithme FFT, n'est-il pas plus avantageux de calculer  $X(f)$  par FFT,  $H(f)$  par FFT, faire le produit et obtenir par FFT inverse le résultat ?

Ce n'est pas si simple.

On peut montrer facilement le résultat suivant, pour une TFD sur  $N$  points :

$$x_n \otimes y_n = \sum_{p=0}^{N-1} x_p y_{(n-p) \bmod N} \xrightarrow{\text{TFD}} X_k Y_k$$

Autrement dit, le produit des TFD des suites  $x$  et  $y$  a pour TFD inverse, le produit de convolution circulaire des suites  $x$  et  $y$  (et non le produit de convolution *linéaire*).

Ce n'est pas perdu !

Prenons là encore un exemple avec  $N=8$ .

2 algorithmes (*Overlap-Save* et *Overlap-Add*) sont à l'origine d'algorithmes de filtrage par FFT. Etudions le deuxième appelé en français Recouvrement-Addition.

# Signal et Image

Considérons un filtre à 5 coefficients  $h_0, \dots, h_4$ .

On veut obtenir :

$$y_n = h_0 x_n + h_1 x_{n-1} + h_2 x_{n-2} + h_3 x_{n-3} + h_4 x_{n-4}$$

Complétons les suites  $h_i$  par des zéros jusqu'à la prochaine puissance de 2 et découpons le signal par blocs successifs  $\mathbf{X}_p$  complétés par des zéros de la manière suivante :

m	0	1	2	3	4	5	6	7
$h_m$	$h_0$	$h_1$	$h_2$	$h_3$	$h_4$	0	0	0
$\mathbf{X}_p$	$x_{n-3}$	$x_{n-2}$	$x_{n-1}$	$x_n$	0	0	0	0
$\mathbf{X}_{p+1}$	$x_{n+1}$	$x_{n+2}$	$x_{n+3}$	$x_{n+4}$	0	0	0	0
$h_{-m \bmod 8}$	$h_0$	0	0	0	$h_4$	$h_3$	$h_2$	$h_1$
$h_{1-m \bmod 8}$	$h_1$	$h_0$	0	0	0	$h_4$	$h_3$	$h_2$
...								
$h_{7-m \bmod 8}$	0	0	0	$h_4$	$h_3$	$h_2$	$h_1$	$h_0$

Les valeurs obtenues à partir du premier bloc sont donc :

# Signal et Image

Bloc P :

$$y_n = h_0 x_{n-3}$$

$$y_1 = h_1 x_{n-3} + h_0 x_{n-2}$$

$$y_2 = h_2 x_{n-3} + h_1 x_{n-2} + h_0 x_{n-1}$$

$$y_3 = h_3 x_{n-3} + h_2 x_{n-2} + h_1 x_{n-1} + h_0 x_n$$

$$y_4 = h_4 x_{n-3} + h_3 x_{n-2} + h_2 x_{n-1} + h_1 x_n$$

$$y_5 = h_4 x_{n-2} + h_3 x_{n-1} + h_2 x_n$$

$$y_6 = h_4 x_{n-1} + h_3 x_n$$

$$y_7 = h_4 x_n$$

Bloc P+1 :

$$y_n = h_0 x_{n+1}$$

$$y_1 = h_1 x_{n+1} + h_0 x_{n+2}$$

$$y_2 = h_2 x_{n+1} + h_1 x_{n+2} + h_0 x_{n+3}$$

$$y_3 = h_3 x_{n+1} + h_2 x_{n+2} + h_1 x_{n+3} + h_0 x_{n+4}$$

$$y_4 = h_4 x_{n+1} + h_3 x_{n+2} + h_2 x_{n+3} + h_1 x_{n+4}$$

$$y_5 = h_4 x_{n+2} + h_3 x_{n+3} + h_2 x_{n+4}$$

$$y_6 = h_4 x_{n+3} + h_3 x_{n+4}$$

$$y_7 = h_4 x_{n+4}$$

On voit que :

$$Y_{n+1} \text{ (linéaire)} = y_4 \text{ (P)} + y_0 \text{ (P+1)}$$

$$Y_{n+2} \text{ (linéaire)} = y_5 \text{ (P)} + y_1 \text{ (P+1)}$$

$$Y_{n+3} \text{ (linéaire)} = y_6 \text{ (P)} + y_2 \text{ (P+1)}$$

$$Y_{n+4} \text{ (linéaire)} = y_7 \text{ (P)} + y_3 \text{ (P+1)}$$

Ainsi, par addition de résultats obtenus sur 2 blocs successifs, la convolution circulaire peut donner les résultats de la convolution linéaire attendue.

Pour réaliser les convolutions circulaires, on utilise une FFT et le filtrage se résume aux opérations suivantes :



# Signal et Image

1. Calcul par FFT de la TFD de la suite  $h_n$  de longueur  $L$  complétée par  $N-L$  zéros, avec  $N$ , une puissance de 2. Ce calcul peut être fait avant.
2. Calcul par FFT de la TFD d'un bloc de longueur  $N-L+1$  de données, complétées par  $L-1$  zéros
3. Multiplication des 2 TFD terme à terme, suivie d'une FFT inverse.
4. Addition des blocs avec recouvrement.

En conclusion :

la FFT de  $h_n$  est calculée auparavant.

A chaque pas de l'algorithme, il faut calculer 1 FFT sur  $N$  points et 1 FFT inverse et les  $N$  multiplications, pour obtenir simplement  $N-L+1$  points de convolution.

Comparé à la convolution directe, on montre que ce calcul devient valable pour  $L=N/2$  et  $L>32$ .

Compte tenu qu'il y a des manipulations de pointeurs et plus de mémoire nécessaire, cette solution est employée à chaque fois que les filtres demandent un « grand nombre » de coefficients.





# Signal et Image

## Cas 2D

Rappelons le contexte des signaux bidimensionnels et en particulier celui des images numériques. La représentation usuelle est celle d'un tableau bidimensionnel contenant les valeurs de l'image échantillonnée. Dans le cas général, les fréquences d'échantillonnage sont  $u_e$  et  $v_e$  en  $x$  et  $y$  et peuvent être différentes. Les pas d'échantillonnage sont respectivement  $\Delta x = 1/u_e$  et  $\Delta y = 1/v_e$ .

Le spectre du signal échantillonné est donc périodisé, avec les périodes  $u_e$  et  $v_e$ .

Nous avons vu également que le théorème d'échantillonnage s'applique et que le « signal » analogique peut être reconstitué à partir de l'image numérique par une fonction porte (le filtre cardinal de reconstruction) d'amplitude  $1/(u_e v_e)$  avec comme support le rectangle  $[-u_e/2, u_e/2] \times [-v_e/2, v_e/2]$ . Sa réponse impulsionnelle s'écrit

$$h(x, y) = \text{sinc}(u_e x) \text{sinc}(v_e y)$$

Sous ces conditions, l'image peut être soumise à des traitements de type filtrage, le plus souvent réalisé dans le domaine spatial.

Une caractéristique spécifique du traitement numérique des images est l'utilisation de filtres non causals.

Si  $s$  est le signal d'entrée et  $w$  le signal de sortie filtré,  $w$  s'écrit à l'aide d'une équation de convolution discrète :

$$w(k, l) = \sum_i \sum_j h(i, j) s(k - i, l - j)$$





# Signal et Image

Le filtrage est le plus souvent implanté directement lorsque le nombre de coefficients non nuls du filtre est fini. Il s'agit alors de calculer, en chaque pixel, le produit scalaire entre coefficients et valeurs de l'image. Le nombre de multiplications par pixel est  $N \times M$  pour un nombre de coefficients de  $N$  par  $M$ . Les dimensions sont choisies impaires pour symétriser le traitement autour du pixel courant. Dans ce cas :

$$w(k, l) = \sum_{i=-m}^{+m} \sum_{j=-n}^{+n} h(i, j) s(k-i, l-j) = h^t s$$

Remarque : on cherche à implanter des filtres séparables à chaque fois que cela est possible. Un filtre est séparable si :

$$h(i, j) = h_1(i) h_2(j)$$

Dans ce cas,

$$\begin{aligned} w(k, l) &= \sum_j h_2(j) \sum_i h_1(i) s(k-i, l-j) \\ &= \sum_j h_2(j) v(k, l-j) \end{aligned}$$

Le traitement est donc équivalent à la mise en cascade de 2 filtres linéaires 1D, le premier,  $h_1$  est appliqué sur toutes les lignes, et le second,  $h_2$ , est appliqué à toutes les colonnes. Le nombre de multiplications devient  $N + M$  à la place de  $NM$ .





# Signal et Image

Effets de bords.

Le problème qui se pose est celui des bords de l'image, pour lequel il « manque » des pixels pour calculer la convolution.

2 techniques principales permettent de prolonger l'image :

- On fait l'hypothèse que les « pixels » hors de l'image sont nuls. Cela revient à limiter le nombre de pixels intervenant dans l'équation de convolution dans la partie transitoire.
- On étend artificiellement le support de l'image en lui adjoignant des lignes et colonnes supplémentaires identiques aux lignes et colonnes des extrémités. Dans ce cas, le filtre est appliqué directement à l'image, sans modification.

On peut aussi ne pas appliquer le filtre aux bords de l'image, mais sa dimension diminue à chaque filtrage...

Filtrage dans le domaine fréquentiel

Comme dans le cas 1D, on peut se poser la question d'un traitement direct dans le domaine fréquentiel, en profitant de l'algorithme FFT, ce qui devient intéressant pour des filtres dont la réponse impulsionnelle est importante. La TFD-2D a déjà été présentée. Comme dans le cas 1D, en réalisant le filtrage par multiplication des TFD, et en revenant dans le domaine spatial, on réalise en fait la convolution circulaire de l'image d'entrée par la TFD inverse de la réponse fréquentielle du filtre.





# Signal et Image

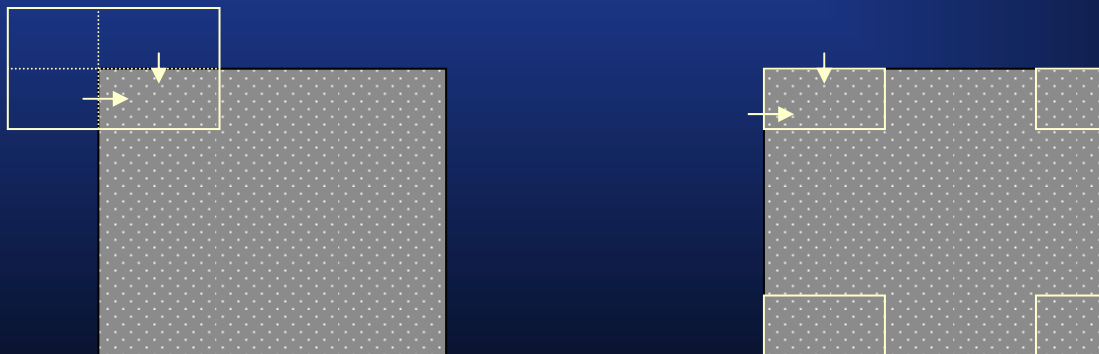
Rappelons que la convolution circulaire (1D) s'exprime par :

$$x_n \otimes y_n = \sum_{p=0}^{N-1} x_p y_{(n-p) \bmod N}$$

Dans le cas 2D, on peut écrire une expression analogue :

$$x_{n,m} \otimes h_{p,q} = \sum_{p=0}^{K-1} \sum_{q=0}^{L-1} h_{p,q} x_{(n-p) \bmod N, (m-q) \bmod M}$$

On montre que le filtrage dans le domaine fréquentiel va se traduire par un autre effet de bord, tout se passe comme si l'image et la réponse impulsionnelle étaient périodiques :





# Signal et Image

La solution consiste à prendre une image de dimension  $2N \times 2N$  au lieu de  $N \times N$  en la complétant par *zero-padding*, (dans le cas d'une image et d'un filtre carrés), ou en complétant l'image à filtrer avec  $K$  zéros dans la direction horizontale et  $L$  dans la direction verticale (pour un filtre de taille  $KL$ );

Pour les images de très grande taille, il est possible de les décomposer en blocs carrés de taille  $N \times N$ . On filtre chaque bloc séparément dans le domaine fréquentiel. Pour lever le problème de circularité, chaque bloc est étendu à une taille de  $2N \times 2N$  en ajoutant des zéros.

Comme le traitement d'un bloc d'indices  $P, Q$  affecte ses voisins, on peut filtrer l'image en additionnant pour chaque pixel du bloc  $B_{P,Q}$  les résultats du filtrage des blocs  $B_{P-1,Q}$ ,  $B_{P,Q-1}$ ,  $B_{P,Q}$ ,  $B_{P+1,Q}$ .

On rejoint ainsi les techniques de recouvrement-addition vues en 1D.





# Signal et Image

## Traitements multi-cadences

Un système multi-cadence est caractérisé par des « cadences » de traitement différentes en divers points de la chaîne de calcul. Par cadence, il faut entendre période d'échantillonnage.

Cette technique liée aux notions de *décimation* et *insertion*, est à la base des bancs de filtres et des méthodes de **codage en sous-bandes** (compression audio), des techniques de modulation multi-porteuses, et de **l'analyse multi-résolution** (traitement d'images – approches pyramidales). Elle conduit à l'analyse en **ondelettes**. (analyse, compression,...).

L'ensemble de ces techniques exigent d'appliquer des sous-échantillonnage et sur-échantillonnage. Au-delà des applications visées ici, le sur et sous échantillonnage permet des changements de fréquence d'échantillonnage, ou de changement de quantification (le sur-échantillonnage compense\* le rapport S/B perdu par une diminution de la quantification). C'est une technique largement utilisée dans les supports audio (CD, DVD,...) pour contourner des opérations plus coûteuses algorithmiquement comme les filtrages passe-bas après bloqueur d'ordre zéro (codage  $\Sigma$ - $\Delta$  par exemple).

\* : Exemple, si la perte de 2 bits diminue de 12dB le rapport S/B, 6dB peuvent être « récupérés » par un sur échantillonnage d'un facteur 4 et autant par mise en forme spectrale du bruit de quantification.



# Signal et Image

## Sur-échantillonnage

Considérons comme exemple une suite  $s(n)$ . On considère la suite  $y$  définie par :

$y(n) = s(n/4)$  pour  $n = 0 \bmod 4$  et  $y(n) = 0$  sinon.

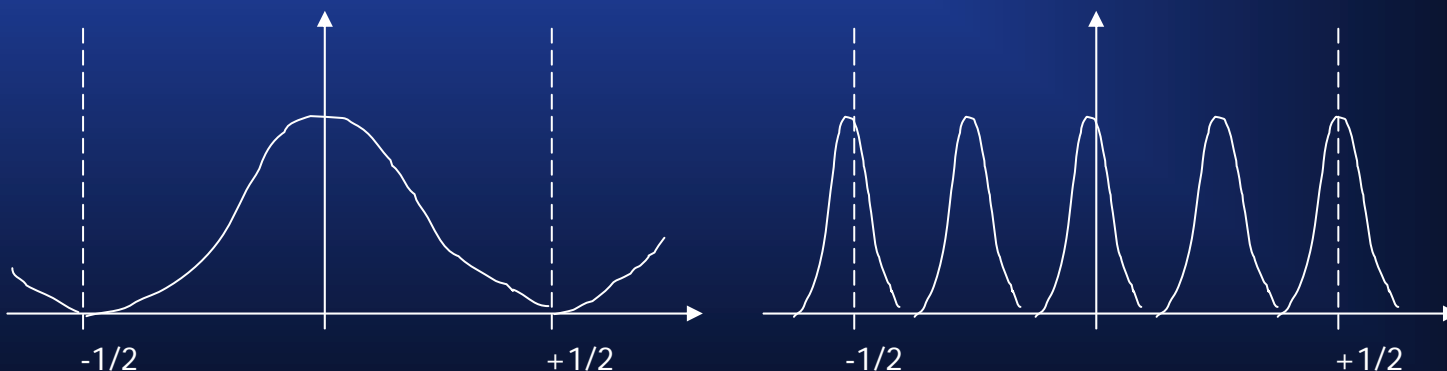
On a donc inséré 3 zéros entre chaque élément de  $s(n)$ . Il s'agit de l'opération d'**insertion** (en anglais *expansion*) de facteur  $M=4$ .

La transformée en  $z$  de la suite  $y(n)$  s'écrit :

$$Y(z) = \sum_k s(k) z^{-4k} = S(z^4)$$

En posant  $z = \exp(2\pi jf)$ , la transformée de Fourier (TFTD) s'écrit :  $Y(f) = S(4f)$

Cela revient à retrouver dans l'intervalle  $(-1/2, +1/2)$  le spectre de  $s(n)$  répliqué 4 fois.



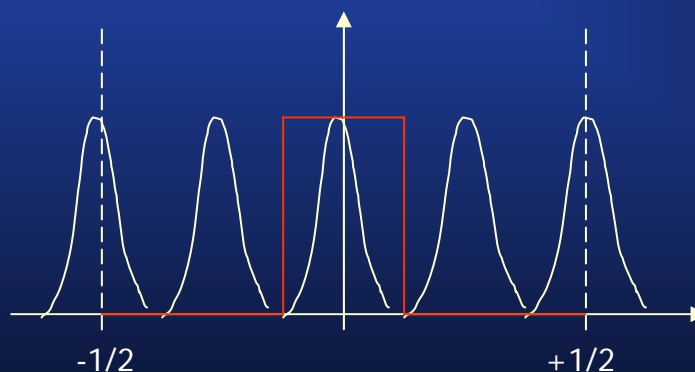
# Signal et Image

On dit qu'il y a apparition *d'images* ou *fréquences-images*. L'insertion de zéros entre les échantillons de  $s(n)$  a introduit des fréquences élevées correspondant aux transitions brutales.

Si on veut que  $y(n)$  soit conforme à  $s(n)$  mais simplement échantillonné plus rapidement, il faut faire suivre l'insertion par un filtrage passe bas de fréquence de coupure  $1/8 = 1/2M$ .

Dans la figure précédente, on voit en effet qu'il faut conserver le contenu de la bande de fréquences comprises dans l'intervalle  $[-1/2M, 1/2M]$ .

Le filtre qui doit être utilisé est un filtre passe bas idéal de fréquence de coupure  $1/2M$  et de gain  $M$ .



# Signal et Image

## Sous-échantillonnage

Considérons maintenant la suite  $y(n)$  obtenue en prélevant un échantillon sur  $M$  de la suite  $s(n)$ . On peut écrire  $y(n) = s(Mn)$ .

Cette opération s'appelle **décimation** d'un facteur  $M$ .

Prenons la transformée en  $z$  de  $y(n)$  :

$$Y(z) = \sum_{n=-\infty}^{+\infty} s(4n)z^{-n}$$

$$Y(z) = \sum_{p=-\infty}^{+\infty} s(p) \left( \frac{1}{4} \sum_{k=0}^3 e^{2j\pi kp/4} \right) z^{-p/4}$$

$$Y(z) = \frac{1}{4} \sum_{k=0}^3 \left( \sum_{p=-\infty}^{+\infty} s(p) \left( z^{1/4} e^{-2j\pi k/4} \right)^{-p} \right)$$

$$Y(z) = \frac{1}{4} \sum_{k=0}^3 S(z^{1/4} e^{-2j\pi k/4})$$

Pour cela, on a utilisé la propriété suivante :

$$\frac{1}{N} \sum_{k=1}^{N-1} e^{2j\pi kn/N} = \begin{cases} 1 & \text{si } n = 0 \bmod N \\ 0 & \text{sinon} \end{cases}$$



# Signal et Image

D'une manière générale, avec  $W_M = \exp(-2\pi j/M)$ , la racine n-ième de l'unité :

$$Y(z) = \frac{1}{M} \sum_{k=0}^{M-1} S(z^{1/M} W_M^k)$$

En passant à la transformée de Fourier (TFTD), on obtient :

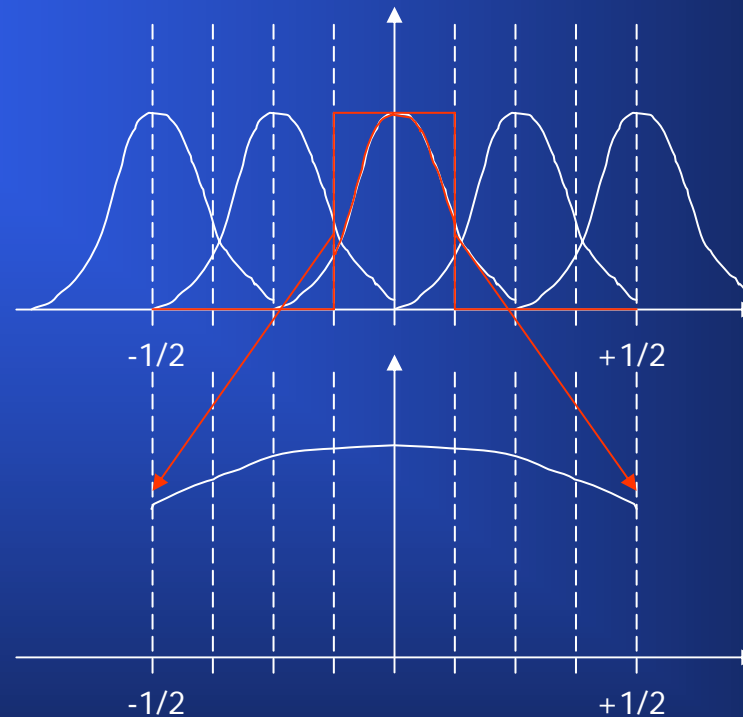
$$Y(f) = \frac{1}{M} \sum_{k=0}^{M-1} S\left(\frac{f-k}{M}\right)$$

Ainsi,  $Y(f)$  est la somme algébrique de  $M$  contributions de  $S(f)$ , mais décalées de  $1/M$  ce qui se traduit par un repliement de spectre.

Là encore, il faudra envisager un filtrage passe-bas de fréquence de coupure  $1/2M$  et de gain  $M$ , avant la décimation.



# Signal et Image



Conclusion :

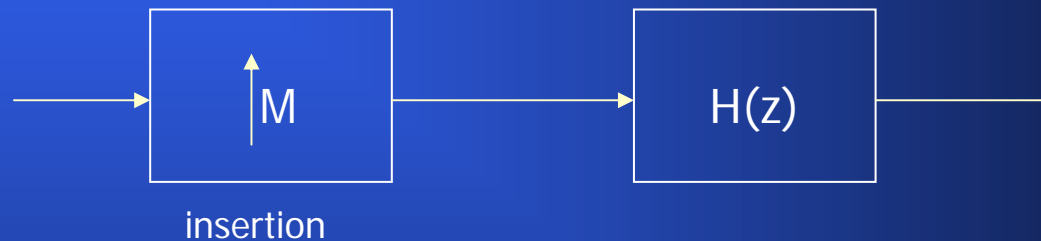
Pour sur-échantillonner d'un facteur  $M$ , il faut :  
 effectuer une insertion de facteur  $M$ , puis filtrer par un passe-bas de gain  $M$  et de fréquence  $1/2M$ .  
 Pour sous-échantillonner d'un facteur  $M$ , il faut :  
 filtrer par un passe-bas de gain  $M$  et de fréquence  $1/2M$  puis effectuer une décimation de facteur  $M$ .



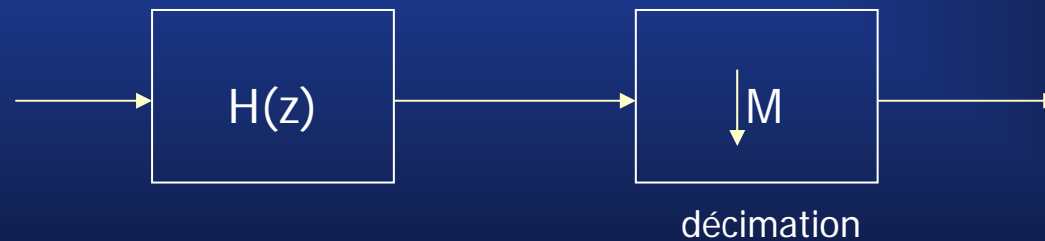
# Signal et Image

## Notation :

Sur-échantillonnage :



Sous-échantillonnage :



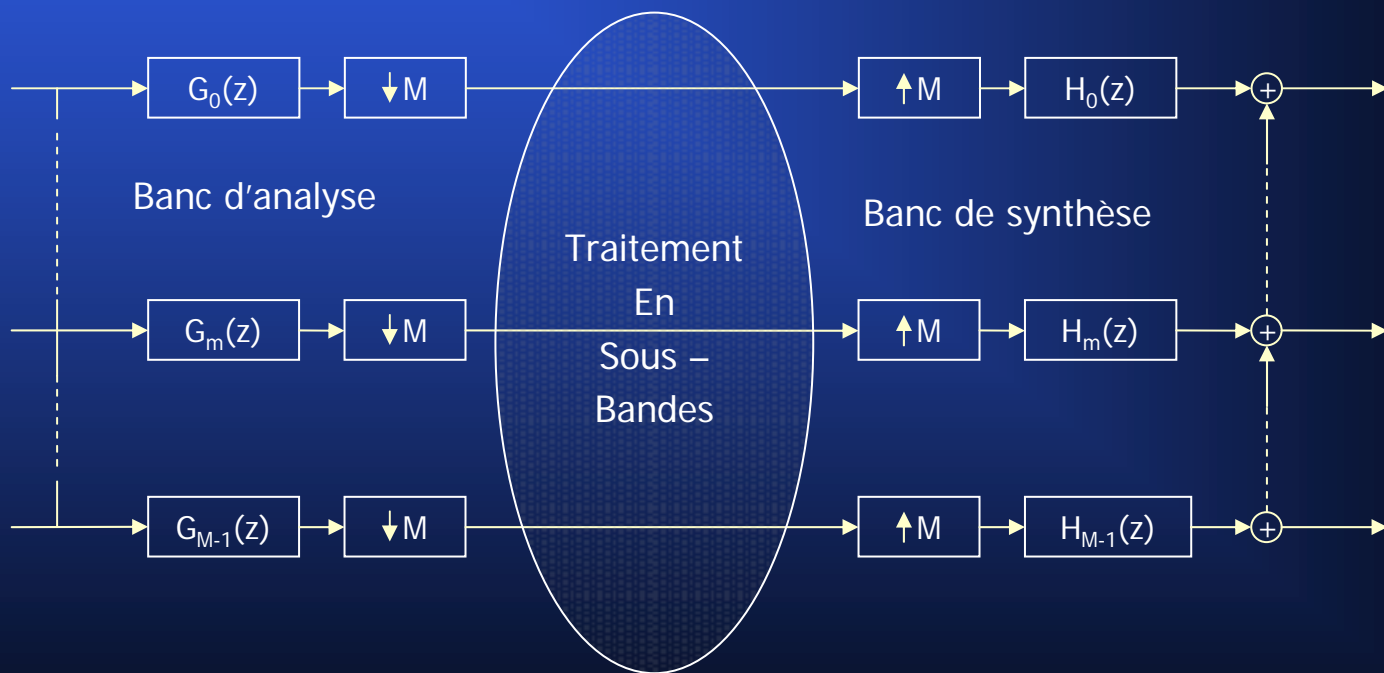
# Signal et Image

## Bancs de filtres

Un banc de filtres est un ensemble de filtres agissant simultanément en découpant la bande de fréquence en  $M$  sous-bandes.

Supposons que les filtres soient des filtres passe-bande idéaux, de largeur  $1/M$ , et disjoints.

Les  $M$  signaux de sortie sont à bande limitée et peuvent être échantillonnés à une cadence  $M$  fois plus faible sans aliasing.



# Signal et Image

## Problème de la reconstruction parfaite

Reprenons le schéma du banc de filtre précédent (en le limitant ici à 2).



Se pose le problème dit de la reconstruction parfaite : quels sont les filtres qui permettent de retrouver en sortie le signal d'entrée sans aucune modification ?

Solution évidente :

$G_0$  et  $H_0$  sont 2 passe-bas idéaux dans la bande  $(0, 1/4)$

$G_1$  et  $H_1$  sont 2 passe-haut idéaux dans la bande  $(1/4, 1/2)$

Cette solution présente l'inconvénient d'utiliser des filtres dont la réponse impulsionnelle est infinie...

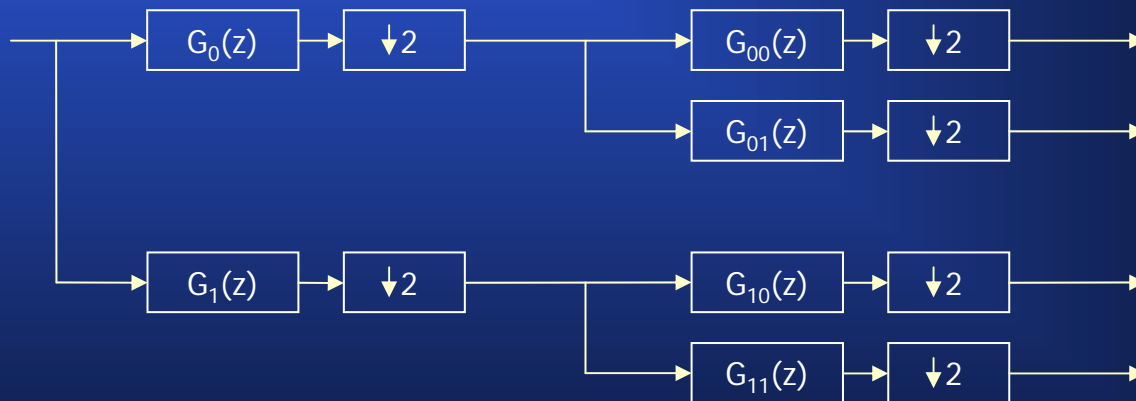
Existe-t-il d'autres solutions utilisant des filtres RIF réalisables mais assurant malgré tout une reconstruction parfaite ?

# Signal et Image

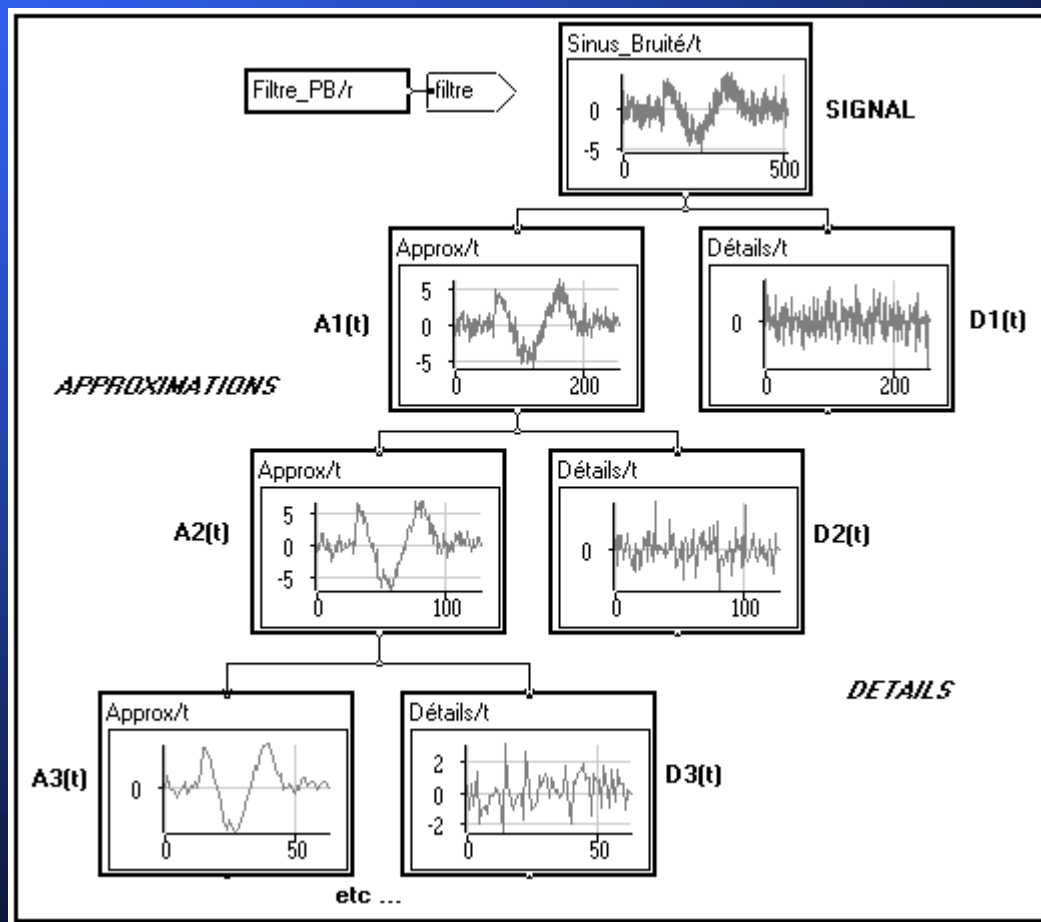
Quelques systèmes répondent à cette question, comme les filtres en quadrature ou les filtres orthogonaux.

Une parenthèse : nous nous limitons au cas du filtre à 2 voies car ils permettent des structures très générales, par une décomposition en octaves.

Ils peuvent être associés à l'analyse en ondelettes...

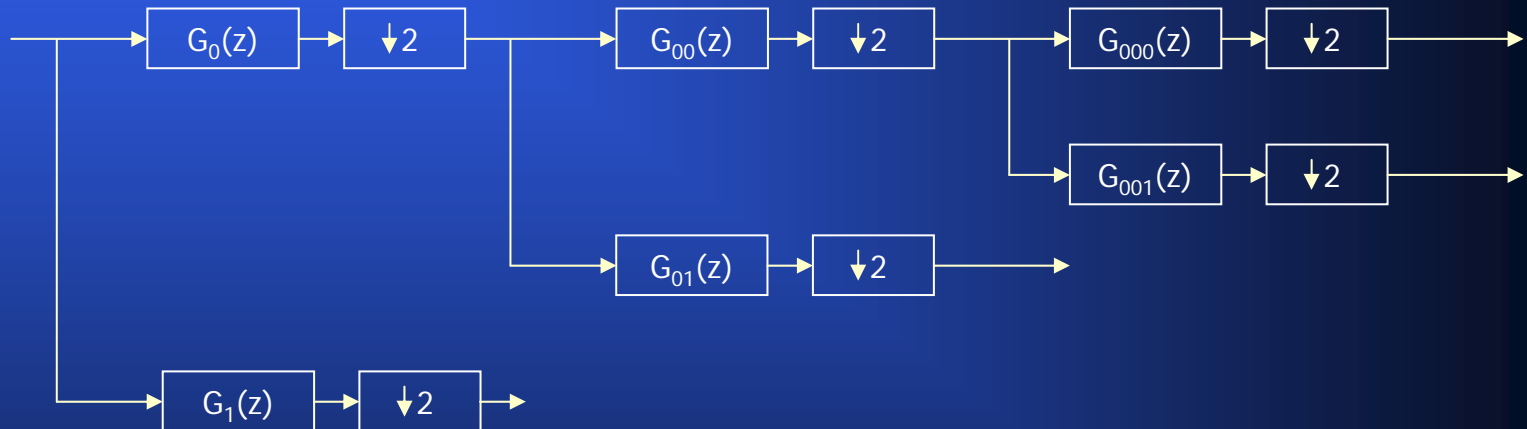


# Signal et Image



# Signal et Image

Décomposition pyramidale d'un signal



# Signal et Image

Revenons au problème de reconstruction parfaite...



Avec les notations ci-dessus et les fonctions de transfert en  $z$  écrites précédemment, on peut montrer les résultats suivants :

$$Z_0(z) = [G_0(z)X(z) + G_0(-z)X(-z)]/2$$

$$Z_1(z) = [G_1(z)X(z) + G_1(-z)X(-z)]/2$$

$$Y_0(z) = H_0(z)Z_0(z)$$

$$Y_1(z) = H_1(z)Z_1(z)$$

Or :

$$X^\wedge(z) = Y_0(z) + Y_1(z)$$

D'où:

$$\begin{aligned} 2X^\wedge(z) &= [G_0(z)H_0(z) + G_1(z)H_1(z)]X(z) + [G_0(-z)H_0(z) + G_1(-z)H_1(z)]X(-z) \\ &= T(z)X(z) + A(z)X(-z) \end{aligned}$$

La reconstruction parfaite est assurée si  $X^\wedge(z) = z^{-r}X(z)$

On en déduit :

$$T(z) = z^{-r} \quad (1)$$

$$A(z) = 0 \quad (2)$$



# Signal et Image

## Filtres en quadrature QMF

Beaucoup de solutions au problème précédent existent.

Les filtres en quadrature imposent  $H_0(z) = G_1(-z)$  et  $H_1(z) = -G_0(-z)$  pour assurer (2). On remplace alors dans (1).

On peut imposer alors  $G_0(z) = G_1(-z)$ . En passant aux transformées de Fourier, cela donne :  $G_0(\exp(2j\pi f)) = G_1(\exp(2j\pi(f-1/2)))$ . Les réponses fréquentielles des filtres ont une symétrie miroir par rapport à la fréquence  $1/4$ .

Le banc de filtres est un banc de filtres QMF (Quadrature Mirror Filters).

En reportant ces conditions, on montre successivement que :

$$G_0(z) = H_0(z) \quad \text{et} \quad H_0^2(z) - H_0^2(-z) = 2z^{-r}$$

$H_0(z)$  ne peut avoir plus de 2 coefficients :  $H_0(z) = h_0z^{-k_0} + h_1z^{-k_1}$

Par identification :  $4h_0h_1z^{-(k_0+k_1)} = 2z^{-r}$  avec  $k_0+k_1$  impair, par ex 0 et 1, et  $h_0h_1 = 1/2$ . Il faut  $h_0 = h_1$  pour une phase linéaire (condition (1)).

$$\text{D'où } H_0^2(f) = \cos^2(\pi f)/2.$$

Ces filtres n'ont pas une grande sélectivité.

D'autres conditions permettent d'obtenir des filtres CQF (Conjugate Quadrature Filters) ou en encore des filtres « orthogonaux ».

La décomposition en sous-bandes utilise aussi largement des blocs FFT pour le filtrage.

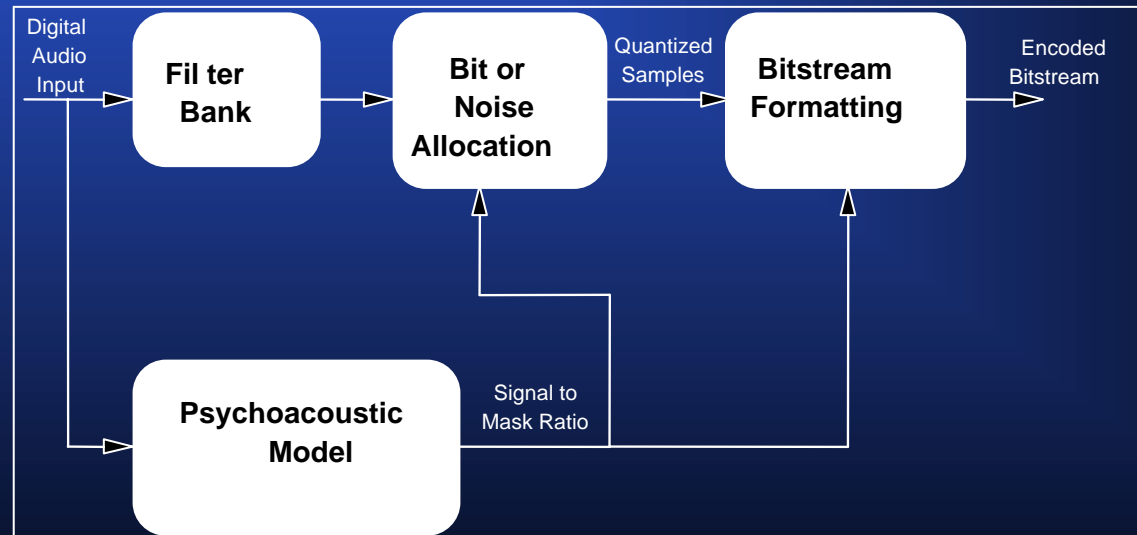




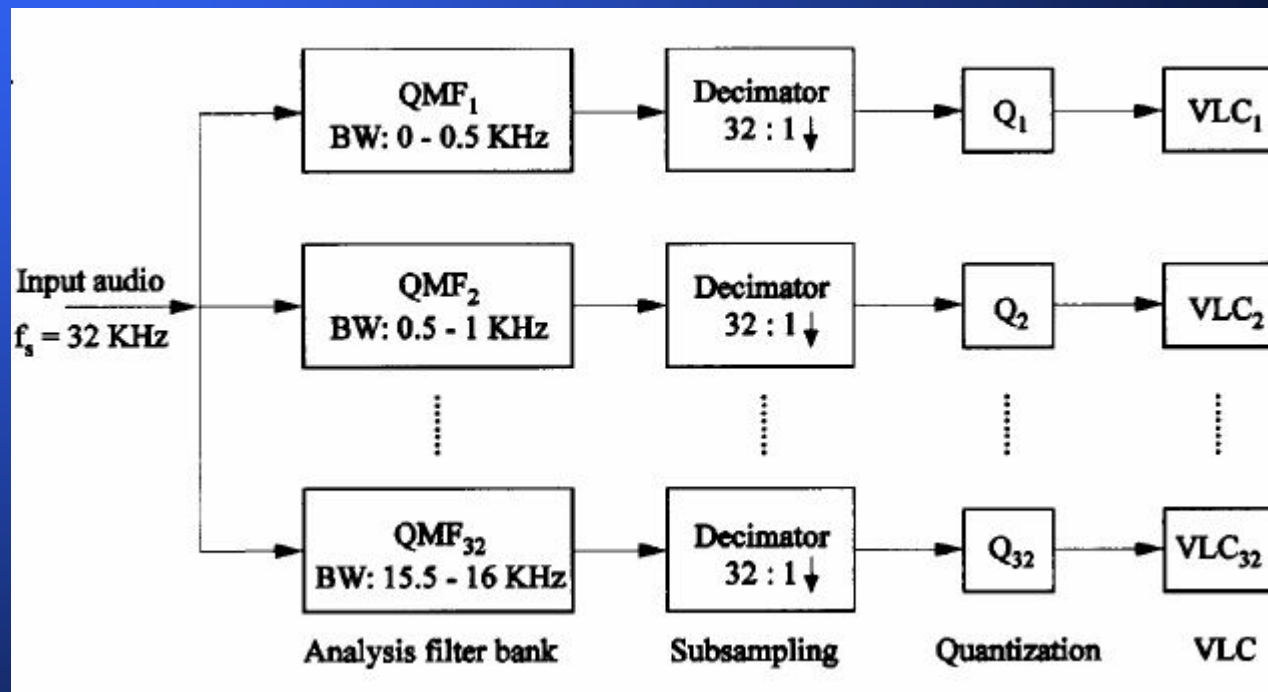
# Signal et Image

Application : le codage des signaux audio en MPEG

Supposons que l'on filtre en  $M$  sous bandes un bloc d'échantillons sonores d'une longueur  $L$ . Chaque bande contient  $L$  échantillons après les filtres et  $L/M$  échantillons après décimation. Soit un total de  $M \cdot L/M = L$  échantillons. Il n'y a aucune compression dans cette technique. Mais si par des considérations d'ordre psycho-acoustique, on alloue moins de bits (voire, on supprime) aux échantillons correspondant à des bandes de fréquences données, on peut compresser le signal sans perte de qualité si les changements ne sont pas audibles.



# Signal et Image



# Signal et Image

