

# **Initiation au traitement du signal et applications**

Cours électif 2A CET 42

## Notes de cours<sup>1</sup>

Frédéric SUR

sur@loria.fr

Département Génie Industriel  
École des Mines de Nancy

Version 0.4, 2009-2012

<sup>1</sup>*Avertissement.* Ce document est constitué de notes de cours dans une version préliminaire. Ces notes doivent être vues comme un complément d'information au cours. Le document contient vraisemblablement des coquilles et erreurs, merci de bien vouloir me les signaler.



# Table des matières

<b>Notations</b>	<b>7</b>
<b>Avant-propos</b>	<b>9</b>
<b>1 Signaux analogiques et filtres associés</b>	<b>11</b>
1.1 Les filtres analogiques . . . . .	11
1.2 Exemple : filtre passe-bas R,C . . . . .	12
1.3 Signaux analogiques . . . . .	14
1.3.1 Rappels et premières propriétés . . . . .	14
1.3.2 Décomposition d'un signal périodique, coefficients de Fourier . . . . .	16
1.3.3 Propriétés des coefficients de Fourier . . . . .	17
1.3.4 Convergence des séries de Fourier . . . . .	19
1.4 Convolution des signaux analogiques périodiques . . . . .	21
<b>2 Signaux numériques et filtres associés</b>	<b>27</b>
2.1 Signaux numériques . . . . .	27
2.1.1 La Transformée de Fourier Discrète . . . . .	29
2.1.2 Transformée de Fourier Rapide . . . . .	31
2.1.3 La transformée de Fourier 2D . . . . .	33
2.2 Filtres numériques . . . . .	34
2.2.1 Définitions . . . . .	34
2.2.2 Transformée en $z$ . . . . .	35
2.2.3 Transformée en $z$ des filtres FIR et IIR . . . . .	35
2.3 Signaux numériques en pratique . . . . .	36
2.3.1 Analyse d'une note de musique . . . . .	36
2.3.2 Interprétation des images dans le domaine fréquentiel . . . . .	37
<b>3 Introduction à la restauration des images</b>	<b>47</b>
3.1 Modèles linéaires de dégradation des images . . . . .	47
3.2 Déconvolution directe . . . . .	50
3.3 Restauration par filtre de Wiener . . . . .	51

3.4	Restauration par l'algorithme de Richardson-Lucy . . . . .	54
<b>4</b>	<b>Compression numérique sans perte</b>	<b>59</b>
4.1	Hypothèse et notations . . . . .	59
4.2	Codes préfixes . . . . .	60
4.3	Théorie de l'information de Shannon . . . . .	61
4.4	Codage de Huffman . . . . .	65
4.5	Exemple . . . . .	66
4.6	Autres algorithmes de compression . . . . .	67
4.6.1	Run-length encoding (RLE) . . . . .	67
4.6.2	Codage arithmétique (1976) . . . . .	67
4.6.3	Codage de Lempel-Ziv-Welch (1984) . . . . .	68
<b>5</b>	<b>Compression numérique avec perte</b>	<b>71</b>
5.1	Décroissance des coefficients de Fourier . . . . .	71
5.2	Effet de Gibbs . . . . .	73
5.3	Transformée discrète en cosinus . . . . .	75
5.4	Quantification . . . . .	77
5.5	Compression MP3 et JPEG . . . . .	77
5.5.1	Compression MP3 . . . . .	77
5.5.2	Compression JPEG . . . . .	77
<b>6</b>	<b>Théorie de l'échantillonnage</b>	<b>79</b>
6.1	Rappels de théorie des distributions . . . . .	79
6.2	Formule de Poisson . . . . .	80
6.3	Théorème d'échantillonnage de Shannon-Nyquist . . . . .	82
6.4	Recouvrement de spectre ou aliasing . . . . .	83
6.5	Retour à la transformée de Fourier discrète . . . . .	83
<b>7</b>	<b>Illustration : sous et sur-échantillonnage</b>	<b>85</b>
7.1	Sur-échantillonnage par <i>zero padding</i> . . . . .	85
7.2	Sous-échantillonnage . . . . .	88
7.2.1	Méthode 1 : décimation brutale . . . . .	88
7.2.2	Méthode 2 : filtrage passe-bas idéal . . . . .	89
7.2.3	Méthode 3 : filtrage passe-bas Butterworth . . . . .	89
<b>8</b>	<b>Analyse temps-fréquence</b>	<b>97</b>
8.1	Principe d'incertitude . . . . .	97
8.2	Transformée de Fourier à fenêtres . . . . .	99
8.3	Illustration du principe d'incertitude . . . . .	100
8.4	Analyse d'un « chirp » . . . . .	100

<b>9 Bestiaire</b>	<b>105</b>
<b>Bibliographie</b>	<b>109</b>



# Notations

$\mathbb{N}, \mathbb{Z}, \mathbb{R}, \mathbb{C}$	ensembles des nombres entiers, relatifs, réels, complexes ;
$\delta_{i,j}$	symbole de Kronecker, $\delta_{i,j} = 1$ si $i = j$ , 0 sinon ;
$\mathbb{E}(X)$	espérance de la variable aléatoire $X$ ;
$E(x)$	partie entière du réel $x$ ;
$\delta_x$	distribution de Dirac en $x \in \mathbb{R}$ ;
$Y$	échelon de Heaviside ;
$x, (x_n), (x_n)_{n \in I}$	suite, suite de terme général $x_n$ , suite à support $I \subset \mathbb{Z}$ ;
$x_n = \mathcal{O}(y_n)$	notation de Landau : $\exists M > 0, \exists N \in \mathbb{N}, \forall n \geq N,  x_n  \leq M y_n $ ;
$i$	unité imaginaire, telle que $i^2 = -1$ ;
$\bar{z}$	conjugué du nombre complexe $z$ ;
$ z $	module du nombre complexe $z \in \mathbb{C}$ ;
$\log(x)$	logarithme népérien du réel $x$ strictement positif ;
$n!$	factorielle du nombre entier $n$ .





# Avant-propos

Ce document est le support du cours électif CET042 *Initiation au traitement du signal et applications* donné en deuxième année à l'École des Mines de Nancy. Il contient les preuves des théorèmes énoncés pendant les séances, ainsi que divers compléments et des éléments de correction pour certains travaux pratiques.

On distingue généralement traitement du **signal analogique** (chapitre 1) et traitement du **signal numérique** (ou *digital* par anglicisme, chapitre 2). Le premier tient du génie électrique et nécessite résistances, bobines, condensateurs, transistors, etc., tandis que le second s'opère par des programmes informatiques sur des ordinateurs ou des puces dédiées (DSP, *Digital Signal Processor*). Comme on le verra, un outil très puissant pour étudier les signaux analogiques est la transformée de Fourier ou les développements en séries de Fourier pour les signaux périodiques, et le pendant numérique est la transformée de Fourier discrète. Si le traitement du signal *numérique* explose depuis quelques décennies, c'est moins grâce à la puissance croissante des puces informatiques que grâce à un algorithme, (re-) découvert dans les années 1960, qui permet de calculer de manière efficace la transformée de Fourier discrète. Il s'agit de l'algorithme très célèbre de la transformée de Fourier rapide (ou FFT, *Fast Fourier Transform*).

Dans ce cours on s'intéressera essentiellement aux signaux numériques, et les différents résultats seront illustrés par des travaux pratiques sous le logiciel MATLAB. Néanmoins, on ne peut pas pour autant passer sous silence la théorie des signaux analogiques, pour au moins deux raisons.

La première est que bon nombre de signaux sont, par essence, analogiques. C'est par exemple le cas des ondes lumineuses (ondes électromagnétiques) ou des ondes sonores (ondes de compressions mécaniques), qui prennent des valeurs évoluant continûment au cours du temps. Pour les représenter sous forme d'un signal numérique, il faut être capable de sélectionner certains instants en lesquels on mesure une grandeur physique associée à l'onde (c'est ce qu'on appelle la discrétisation temporelle) et de représenter la valeur mesurée avec un nombre fini de bits (c'est ce qu'on appelle la quantification). Ce problème de conversion de la représentation analogique vers la représentation numérique (ainsi que le problème inverse) est l'objet de la **théorie de l'échantillonnage** (chapitre 6).

La seconde raison est évoquée dans le livre de Stéphane Mallat (cf bibliographie

page 109) : on ne dispose pas de « bonne théorie » pour estimer la régularité des signaux numériques. Or, on peut par exemple démontrer qu'un signal (analogique) est représenté de manière d'autant plus compacte par sa transformée de Fourier qu'il est régulier (c'est-à-dire de classe  $C^k$ , avec  $k$  augmentant). C'est cet argument qui justifie la **compression avec perte** des signaux numériques par des algorithmes comme JPEG ou MP3 (chapitre 5). Auparavant, nous introduirons la **compression sans perte** toujours associée à la compression avec perte car elle ne coûte (quasiment) rien et, comme son nom l'indique, ne détériore pas le signal original (chapitre 4). Elle est basée sur la *théorie statistique de l'information* initiée par Claude Shannon dans les années 1940-1950.

Nous donnerons également des applications des différents concepts à deux classes de signaux : les sons et les images. Nous présenterons quelques éléments introductifs à la **restauration des images** dégradées (chapitre 3) et illustrerons la théorie de l'échantillonnage par des **problèmes de sous et sur-échantillonnage** (chapitre 7).

Nous traiterons également de l'**analyse temps-fréquence** qui intervient de manière centrale dans les problèmes pratiques d'analyse des signaux « non stationnaires » dont les propriétés changent au cours du temps (chapitre 8).

Des **notes biographiques** figurent au chapitre 9.

*Historique des versions de ces notes de cours :*

- v 0.4 : février 2012 (111 pages).
- v 0.3 : février 2010 (111 pages).
- v 0.2 : septembre 2009 (85 pages).
- v 0.1 : février 2009 (30 pages).

# Chapitre 1

## Signaux analogiques et filtres associés

### 1.1 Les filtres analogiques

On considère dans un premier temps des signaux analogiques à une variable (le temps par exemple).

Soient  $X$  et  $Y$  deux espaces vectoriels normés (respectivement espaces des signaux « en entrée » et espace des signaux « en sortie »). Un signal  $x \in X$  est une application de  $\mathbb{R}$  dans  $\mathbb{C}$  (quitte à prendre la partie réelle pour des signaux réels).

Soit  $A$  une application de  $X$  vers  $Y$ .

On note  $T_a$  l'opérateur retard de  $a$  : pour un signal  $x$ , alors

$$\forall t \in \mathbb{R}, T_a(x)(t) = x(t - a).$$

On suppose les espaces  $X$  et  $Y$  invariants par  $T_a$ , c'est-à-dire que les signaux dans  $X$  (resp. dans  $Y$ ) décalés de  $a$  sont aussi dans  $X$  (resp. dans  $Y$ ).

**Définition 1** L'application  $A : X \rightarrow Y$  est dit *invariante* (ou *stationnaire*), si

$$AT_a = T_aA.$$

Autrement dit,  $A$  est invariant si décaler un signal d'entrée  $x$  de  $a$  décale également le signal de sortie  $y = A(x)$  de  $a$ .

**Définition 2** Une application  $A : X \rightarrow Y$  linéaire, continue, et invariante est un **filtre**.

Il s'agit des bonnes propriétés que l'on peut attendre si on formalise la notion de filtre des électroniciens : un filtre vérifie le principe de superposition (l'image de la

combinaison linéaire de signaux est la même combinaison linéaire des images de ces signaux), est « stable » (une petite perturbation du signal d'entrée entraîne une petite perturbation du signal de sortie grâce à la continuité) et est invariant (utiliser le filtre maintenant ou dans une heure donne le même résultat).

**Définition 3** on note  $e_\lambda$  le signal  $t \mapsto \exp(2i\pi\lambda t)$ . Un tel signal trigonométrique est dit *monochromatique*.

*Remarque :*  $\lambda$  est la fréquence du signal  $e_\lambda$ .

**Théorème 1** Soit  $A$  un filtre. Les signaux  $e_\lambda$  sont fonctions propres de  $A$ , i.e.

$$\forall \lambda \in \mathbb{R}, \exists H(\lambda) \in \mathbb{R}, A(e_\lambda) = H(\lambda)e_\lambda.$$

$H$  est appelée *fonction de transfert du filtre*.

*Démonstration.* Remarquons que  $e_\lambda(t+u) = e_\lambda(t)e_\lambda(u)$ . Soit  $f_\lambda = A(e_\lambda)$ . Soit  $u \in \mathbb{R}$ , alors pour tout  $t \in \mathbb{R}$ , en utilisant l'invariance et la linéarité du filtre  $A$  :

$$f_\lambda(t+u) = T_{-u}f_\lambda(t) = A(T_{-u}e_\lambda)(t) = A(e_\lambda(\cdot+u))(t) = e_\lambda(u)A(e_\lambda)(t).$$

Pour la valeur  $t = 0$  :  $f_\lambda(u) = A(e_\lambda(0)) \cdot e_\lambda(u)$ . D'où le résultat pour  $H(\lambda) = A(e_\lambda)(0)$ .  $\square$

Le théorème 1 nous montre pourquoi la classe des signaux trigonométriques est si importante en traitement du signal.

## 1.2 Exemple : filtre passe-bas R,C

Les filtres analogiques sont réalisés à l'aide de composants électroniques, par exemple des résistances (notées  $R$ ), des bobines ou inductances (notées  $L$ ), et des condensateurs (notés  $C$ ).

Dans cette section nous examinons l'exemple du filtre dit  $RC$ , constitué d'une résistance et d'un condensateur (voir figure 1.1).

Le cours d'« électricité » du lycée nous montre que la tension en sortie  $y$  est régie par l'équation différentielle :

$$RC y'(t) + y(t) = x(t).$$

Au sens des distributions à support sur  $\mathbb{R}^+$ , on peut écrire cette équation sous la forme :

$$(-RC\delta' + \delta) * y = x.$$

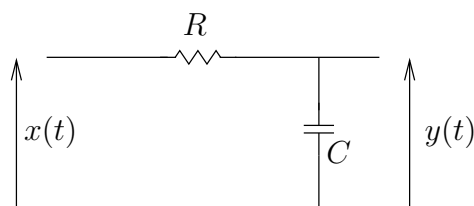


FIG. 1.1 – Le filtre  $RC$ , composé d'une résistance  $R$  et d'un condensateur  $C$ . Le signal analogique  $x(t)$  est la tension en entrée du filtre, le signal  $y(t)$  est la tension en sortie.

Le calcul symbolique de Heaviside nous permet d'écrire les solutions sous la forme :

$$y(t) = (h * x)(t) \quad (1.1)$$

avec (en notant  $Y$  l'échelon d'Heaviside) :

$$h(t) = \frac{1}{RC} e^{-t/RC} Y(t).$$

L'application  $A : x \mapsto y = h * x$  est linéaire, invariante, et continue (à vérifier en exercice facile), il s'agit bien d'un filtre.

**Définition 4 (réponse impulsionnelle)** La fonction

$$h(t) = \frac{1}{RC} e^{-t/RC} Y(t)$$

est appelée **réponse impulsionnelle** du filtre. D'après l'équation (1.1)  $h$  est la réponse  $y$  du filtre lorsque l'entrée  $x$  est une distribution de Dirac  $\delta$  (i.e. une « impulsion »).

Dans ce qui suit, nous allons mettre en évidence une propriété des signaux monochromatiques. En appliquant la transformée de Fourier à l'équation (1.1), on obtient :

$$\widehat{y}(\lambda) = H(\lambda) \cdot \widehat{x}(\lambda)$$

avec :

$$H(\lambda) = \widehat{h}(\lambda) = \frac{1}{1 + i\lambda RC}.$$

En effet :

$$H(\lambda) = \frac{1}{RC} \int_0^{+\infty} e^{-t/RC} e^{-i\lambda t} dt = \frac{1}{1 + i\lambda RC}.$$

Calculons alors le signal en sortie lorsque le signal en entrée est monochromatique. Successivement :

$$\begin{aligned}
 Ae_\lambda(t) &= (h * e_\lambda)(t) \\
 &= \int_{\mathbb{R}} \frac{1}{RC} e^{-s/RC} Y(s) e^{2i\pi\lambda(t-s)} ds \\
 &= e^{2i\pi\lambda t} \int_{\mathbb{R}} \frac{1}{RC} e^{-s/RC} Y(s) e^{-2i\pi\lambda s} ds \\
 &= H(2\pi\lambda) e_\lambda.
 \end{aligned}$$

Ainsi, avec la définition donnée dans le théorème 1, la *fonction de transfert* du filtre  $RC$  est :

$$\mathcal{H}(\lambda) = H(2\pi\lambda) = \frac{1}{1 + 2i\pi\lambda RC}.$$

Le complexe  $\mathcal{H}(\lambda)$  est la valeur par laquelle il faut multiplier le signal trigonométrique  $e_\lambda$  en entrée pour trouver le signal en sortie.

*Remarque* : l'apparition du coefficient  $2\pi$  dépend de la « version » de la transformée de Fourier utilisée.

La figure 1.2 montre l'allure du graphe de  $|\mathcal{H}(\lambda)|^2$ . On voit que les signaux de fréquences basses ( $\lambda$  proche de 0) sont peu modifiés alors que ceux de fréquences élevées ( $\lambda \rightarrow +\infty$ ) sont fortement atténués. On parle donc de filtre *passé-bas*.

*Remarque* : Le lecteur trouvera dans la littérature des exemples de réalisation de filtres passé-haut ou passé-bande à l'aide de montages  $RLC$ .

## 1.3 Signaux analogiques

On considère dans cette section des signaux analogiques (de  $\mathbb{R}$  dans  $\mathbb{C}$ ), périodiques.

### 1.3.1 Rappels et premières propriétés

On considère dans la suite des signaux périodiques, de période  $a > 0$ .

**Définition 5** Les espaces  $L_p^1$  et  $L_p^2$  (et les normes associées) sont définis par :

- $f \in L_p^1(0, a)$  ssi  $\|f\|_1 = \int_0^a |f(x)| dx < +\infty$ .
- $f \in L_p^2(0, a)$  ssi  $\|f\|_2 = \left(\int_0^a |f(x)|^2 dx\right)^{1/2} < +\infty$ .

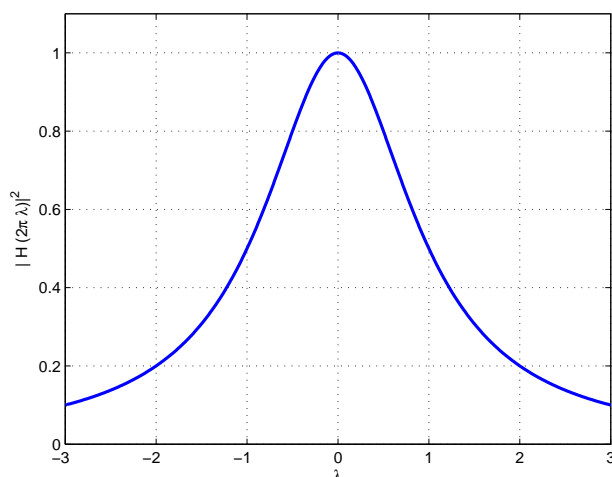


FIG. 1.2 – Carré du module de la fonction de transfert  $\mathcal{H}(\lambda)$  du filtre  $RC$  pour  $2\pi RC = 1$ . On lit sur ce graphe que l'amplitude des signaux monochromatiques de basses fréquences est conservée tandis que les signaux de hautes fréquences sont atténués. Ce sont des propriétés caractéristiques des filtres passe-bas. Pour  $\lambda = 1/(2\pi RC)$ , l'amplitude du signal est multipliée par  $1/\sqrt{2}$ .

De plus,  $L_p^2(0, a)$  muni du produit scalaire :

$$(f, g) = \int_0^a f(t)\bar{g}(t) dt$$

est un espace de Hilbert.

Remarquons que tout intervalle de longueur  $a$  convient pour définir les intégrales intervenant dans  $\|f\|_1$  et  $\|f\|_2$ .

Remarquons également que :  $L_p^2(0, a) \subset L_p^1(0, a)$ .

En effet, d'après l'inégalité de Cauchy-Schwarz :  $\int_0^a |f(x)| dx \leq \sqrt{a} \sqrt{\int_0^a |f(x)|^2 dx}$ .

On dit souvent que les signaux de  $L^2$  sont « à énergie finie ».

**Définition 6** On note  $e_n(t) = e^{2i\pi nt/a}$  (pour  $n \in \mathbb{Z}^*$ ) les **signaux trigonométriques de période**  $a/|n|$ . Le signal monochromatique  $e_n$  a pour fréquence  $|n|/a$ .

Remarque :  $\forall n \in \mathbb{Z}, e_n \in L_p^2(0, a)$ .

**Proposition 1** La famille des  $(e_n)_{n \in \mathbb{Z}}$  est orthogonale, et plus précisément :

$$\int_0^a e_n(t)\bar{e}_m(t) dt = \begin{cases} 0 & \text{si } n \neq m \\ a & \text{si } n = m \end{cases} \quad (1.2)$$

Initiation au traitement du signal et applications

*Démonstration.* Il suffit de faire le calcul en distinguant les deux cas. □

**Définition 7** *Le sous-espace vectoriel de  $L_p^2(0, a)$  engendré par la famille  $(e_n)_{|n| \leq N}$  est l'espace des **polynômes trigonométriques** de degré au plus  $N$ . Il est noté  $\mathcal{T}_N$ .*

*Remarque :* pourquoi « polynômes trigonométriques » ? Tout élément  $p$  de  $\mathcal{T}_N$  s'écrit sous la forme  $p(t) = \sum_{|n| \leq N} c_n (e^{2i\pi t/a})^n$ .

**Théorème 2** *Le sous-espace  $\mathcal{T}_N$  admet la base orthonormée :  $\left( \frac{1}{\sqrt{a}} e_n \right)_{-N \leq n \leq N}$  (il est donc de dimension  $2N + 1$ .)*

*Démonstration.* Il s'agit d'une famille orthonormale d'après la proposition 1, et génératrice par définition de  $\mathcal{T}_N$ . □

**Théorème 3** *Égalité de Parseval pour les polynômes trigonométriques :*

$$\frac{1}{a} \int_0^a |p(t)|^2 dt = \sum_{n=-N}^N |c_n|^2. \tag{1.3}$$

*Démonstration.* Soit  $p = \sum_{|n| \leq N} c_n e_n \in \mathcal{T}_N$ .

D'après le théorème 2,  $(p/\sqrt{a}, p/\sqrt{a}) = \sum_{|n| \leq N} |c_n|^2$ . □

### 1.3.2 Décomposition d'un signal périodique, coefficients de Fourier

Vu l'importance des signaux trigonométriques, il est tentant de voir comment on peut « approcher » un signal de l'espace de Hilbert  $L_p^2(0, a)$  par un signal de  $\mathcal{T}_N$ . Le théorème suivant, illustré par la figure 1.3, nous dit comment « projeter » sur l'espace  $\mathcal{T}_N$ .

**Théorème 4** *Si  $f \in L_p^2(0, a)$ , il existe un unique  $f_N$  dans  $\mathcal{T}_N$  tel que :*

$$\|f - f_N\|_2 = \min\{\|f - p\|_2, p \in \mathcal{T}_N\}.$$

*Ce polynôme trigonométrique  $f_N$  vérifie :  $f_N(t) = \sum_{n=-N}^N c_n e^{2i\pi n t/a}$ ,*

*où :  $c_n = \frac{1}{a} \int_0^a f(t) e^{-2i\pi n t/a} dt$ .*

*F. Sur février 2012*



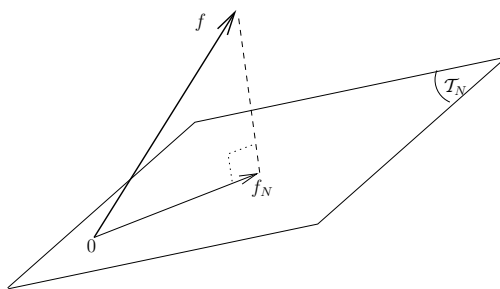


FIG. 1.3 – Projection d'un signal  $f$  de  $L_p^2(0, a)$  sur  $\mathcal{T}_N$ .

*Démonstration.* Soit  $p = \sum_{|n| \leq N} \alpha_n e_n$  un polynôme trigonométrique quelconque de  $\mathcal{T}_N$ . Alors

$$\|f - p\|_2^2 = \|f\|_2^2 - 2\operatorname{Re}(f, p) + \|p\|_2^2.$$

D'après le théorème 3,  $\|p\|_2^2 = a \sum_{|n| \leq N} |\alpha_n|^2$ . D'autre part  $(f, p) = \sum_{|n| \leq N} \overline{\alpha_n} (f, e_n)$ .  
Donc :

$$\|f - p\|_2^2 = \|f\|_2^2 + a \sum_{|n| \leq N} \left( |\alpha_n|^2 - 2\operatorname{Re} \left( \frac{\overline{\alpha_n}}{a} (f, e_n) \right) \right).$$

Or, en notant  $c_n = \frac{1}{a} (f, e_n)$ , on a  $|\alpha_n|^2 - 2\operatorname{Re}(\overline{\alpha_n} c_n) = |c_n - \alpha_n|^2 - |c_n|^2$ .

Donc  $\|f - p\|_2^2$  atteint son minimum lorsque l'on choisit les  $\alpha_n$  égaux aux  $c_n$ .  $\square$

*Remarque :* l'idée de la preuve est valable pour n'importe quelle base orthonormée de l'espace  $\mathcal{T}_N$ .

**Définition 8** Soit  $f \in L_p^2(0, a)$ .

$$c_n = \frac{1}{a} \int_0^a f(t) e^{-2i\pi nt/a} dt$$

(où  $n \in \mathbb{Z}$ ) est le  $n$ -ème **coefficient de Fourier** de  $f$ .

*Remarque :*  $c_n$  et  $c_{-n}$  sont les coefficients des termes de période  $a/|n|$  (ou fréquence  $|n|/a$ ) dans le polynôme trigonométrique "le plus proche" de  $f$  dans  $\mathcal{T}_N$ .

### 1.3.3 Propriétés des coefficients de Fourier

Rappelons tout d'abord le lemme de Riemann-Lebesgue :

*Initiation au traitement du signal et applications*

**Proposition 2 (lemme de Riemann-Lebesgue)** Soit  $f$  une fonction de  $\mathbb{R}$  dans  $\mathbb{C}$  intégrable sur un intervalle  $I \subset \mathbb{R}$ , alors :

$$\int_I f(t)e^{-ixt} dt \rightarrow 0 \text{ lorsque } x \rightarrow \pm\infty$$

En particulier,  $c_n \rightarrow 0$  lorsque  $|n| \rightarrow +\infty$ .

**Proposition 3** Soit  $f \in L_p^2(0, a)$ . Les propriétés suivantes sont vérifiées :

1. Décalage temporel.

Soit  $\tilde{f}(t) = f(t - t_0)$ . Alors  $c_n(\tilde{f}) = e^{-2i\pi n t_0/a} c_n(f)$ .

2. Décalage fréquentiel.

Soit  $\tilde{f}(t) = e^{2i\pi n_0 t/a} f(t)$ . Alors  $c_n(\tilde{f}) = c_{n-n_0}(f)$ .

3. Différentiation :

Si  $f$  est  $C^1$ , alors  $c_n(f') = 2i\pi n/a c_n(f)$ .

4. Signal réel :

Si  $f$  est à valeurs réelles alors  $\overline{c_n(f)} = c_{-n}(f)$  et  $(|c_n(f)|)_n$  est pair.

*Démonstration.* Il suffit de faire le calcul (et une intégration par parties pour le point 3).  $\square$

**Proposition 4 (inégalité de Bessel)** Soit  $f \in L_p^2(0, a)$  et  $(c_n)$  la suite des coefficients de Fourier de  $f$ .

$$\forall N \in \mathbb{N}, \sum_{n=-N}^N |c_n|^2 \leq \frac{1}{a} \int_0^a |f(t)|^2 dt. \quad (1.4)$$

*Démonstration.* D'après la preuve du théorème 4, on a :  $\|f - f_N\|_2^2 + a \sum_{n=-N}^N |c_n|^2 = \|f\|_2^2$ , d'où l'inégalité cherchée.  $\square$

L'inégalité de Bessel permet d'obtenir sans le lemme de Riemann-Lebesgue le résultat suivant.

**Proposition 5** Les coefficients de Fourier « s'éteignent » lorsque  $|n|$  croît :

si  $f \in L_p^2(0, a)$ , alors  $c_n(f) \rightarrow 0$  quand  $|n| \rightarrow +\infty$ .

(en fait ceci est vrai pour  $f \in L_p^1(0, a)$ , cf lemme de Riemann-Lebesgue, proposition 2.)

*Démonstration.* D'après l'inégalité de Bessel, la série de terme général  $|c_n|^2$  converge (les sommes partielles sont croissantes majorées), donc  $c_n \rightarrow 0$  quand  $|n| \rightarrow +\infty$ .  $\square$

### 1.3.4 Convergence des séries de Fourier

**Théorème 5**  $f_N$  converge vers  $f$  dans  $L_p^2$ .

Cela signifie :  $\|f - f_N\|_2^2 = \int_0^a |f(t) - f_N(t)|^2 dt \rightarrow 0$  lorsque  $N \rightarrow +\infty$ .

*Démonstration.* Admis dans cette version du polycopié. □

La convergence des séries de Fourier est illustrée par la figure 1.4.

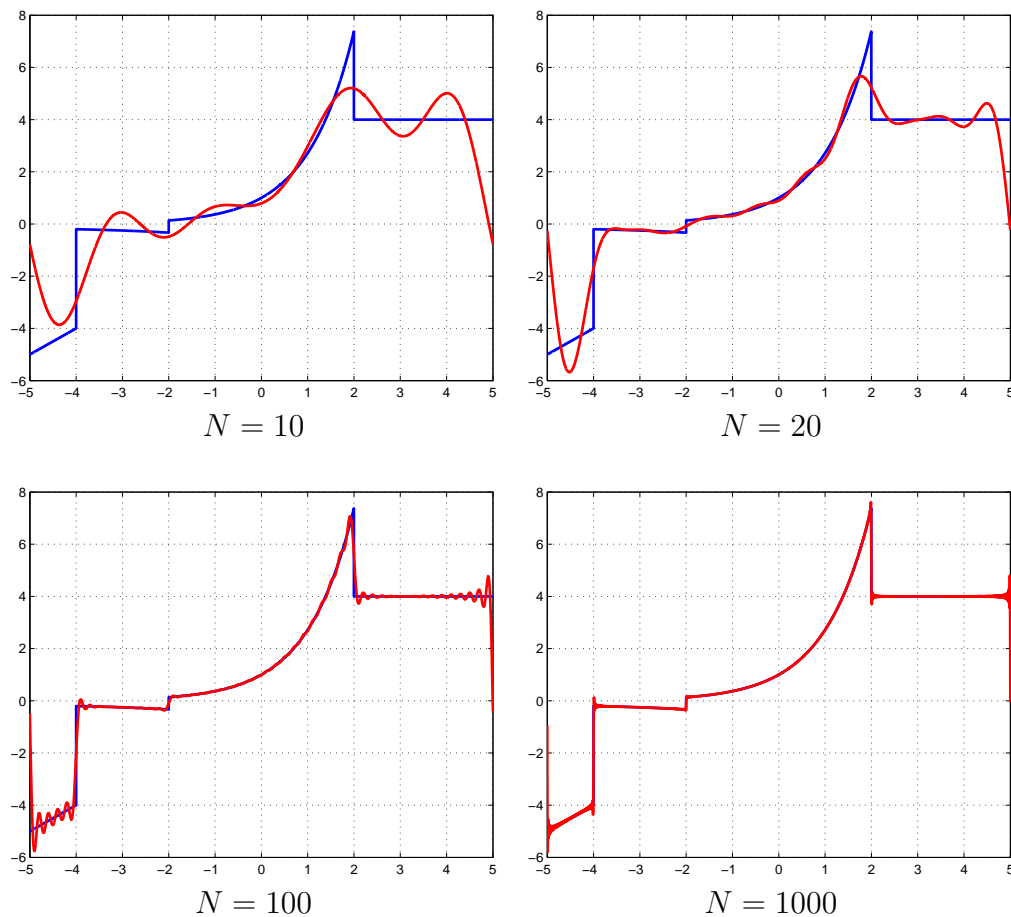


FIG. 1.4 – Convergence d'une série de Fourier dans  $L_p^2(0, a)$ . En bleu, la fonction originale  $f$ , représentée sur une période. En rouge, approximation  $f_N$ , pour différentes valeurs de  $N$ . On voit des oscillations résiduelles autour des discontinuités, qui ne disparaissent pas lorsque  $N$  augmente. Il s'agit du phénomène de Gibbs (voir chapitre 5).

L'égalité de Parseval est valable pour les signaux de  $L_p^2(0, a)$ .

**Proposition 6 (égalité de Parseval)** Soit  $f \in L_p^2(0, a)$  et  $(c_n)$  la suite des coefficients de Fourier de  $f$ .

$$\frac{1}{a} \int_0^a |f(t)|^2 dt = \sum_{n=-\infty}^{+\infty} |c_n|^2. \quad (1.5)$$

*Démonstration.* D'après la preuve du théorème 4 :  $\|f - f_N\|_2^2 + a \sum_{n=-N}^N |c_n|^2 = \|f\|_2^2$ . Il suffit ensuite de passer à la limite avec le théorème 5. □

**Proposition 7 (unicité des coefficients de Fourier)** Deux fonctions de  $L_p^2(0, a)$  ayant les mêmes coefficients de Fourier sont égales (presque partout).

*Démonstration.* Si  $f, g \in L_p^2(0, a)$  sont deux fonctions telles que  $\forall n \in \mathbb{Z}, c_n(f) = c_n(g)$ , alors d'après l'égalité de Parseval  $\|f - g\|_2 = 0$ , ce qui permet de conclure. □

**Théorème 6 (Jordan-Dirichlet)** Soit  $f \in L_p^1(0, a)$  telle que en un point  $t_0$ ,  $f(t_0+)$ ,  $f(t_0-)$ ,  $f'(t_0+)$  et  $f'(t_0-)$  existent.

Alors :

$$f_N(t_0) \rightarrow \frac{1}{2}(f(t_0+) + f(t_0-)) \text{ quand } N \rightarrow +\infty.$$

*Démonstration.* Pour simplifier les notations, voici la preuve pour  $a = 2\pi$ .

Notons  $D_n(x) = \sum_{|k| \leq n} e^{ikx}$ . On montre facilement (à faire en exercice !) que :

$$D_n(x) = \frac{\sin((n + 1/2)x)}{\sin(x/2)}.$$

Alors  $f_N(x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(t)D_n(x - t) dt = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x - t)D_n(t) dt$  (après changement de variable).

D'après ce qui précède :

$$f_N(x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{f(x - t)}{\sin(t/2)} \sin((n + 1/2)t) dt.$$

Comme  $t \mapsto \frac{f(x-t)}{\sin(t/2)}$  n'est a priori pas intégrable en 0, on ne peut pas appliquer directement le lemme de Riemann-Lebesgue.

Après le changement de variable  $t' = -t$  sur  $[-\pi, 0]$ , on peut écrire :

$$f_N(x) = \frac{1}{2\pi} \int_0^{\pi} \frac{f(x - t) + f(x + t)}{\sin(t/2)} \sin((n + 1/2)t) dt.$$

Comme  $\int_0^\pi D_n(t) dt = \pi$ , on arrive à :

$$f_N(x) - \frac{1}{2}(f(t_{0+}) + f(t_{0-})) = \frac{1}{2\pi} \int_0^\pi \frac{f(x-t) - f(t_{0-}) + f(x+t) - f(t_{0+})}{\sin(t/2)} \sin((n+1/2)t) dt.$$

Maintenant :  $f(x-t) - f(t_{0-}) = \mathcal{O}(t)$  et  $f(x+t) - f(t_{0+}) = \mathcal{O}(t)$  quand  $t \rightarrow 0+$  par existence de  $f'(t_{0-})$  et  $f'(t_{0+})$ . Donc la fraction dans l'intégrale précédente est localement intégrable sur  $[0, 2\pi]$ , et le lemme de Riemann-Lebesgue (proposition 2) permet de conclure.  $\square$

*Remarque :* si  $f$  est en plus continue, il y a donc convergence simple de la série de Fourier de  $f$  vers  $f$ .

**Théorème 7** Soit  $f$  de période  $a$ , continue sur  $\mathbb{R}$  et de classe  $C^1$  par morceaux, alors la série de Fourier de  $f$  converge normalement, donc uniformément vers  $f$ .

*Démonstration.* Admis dans cette version du polycopié.  $\square$

*Remarque :* dans ce cas la série de Fourier de  $f'$  s'obtient en dérivant termes à termes la série de Fourier de  $f$ .

## 1.4 Convolution des signaux analogiques périodiques

On a vu dans l'exemple de la section 1.2 que l'action du filtre RC était équivalent à une convolution par la réponse impulsionnelle (dont la transformée de Fourier est la fonction de transfert du filtre). Ceci est en fait une propriété générale des filtres analogiques. Dans cette section figurent des résultats portant sur la convolution (ou, de manière équivalente, sur les filtres analogiques).

**Définition 9 (convolution de fonctions périodiques)** Soient  $f, g \in L_p^1(0, a)$ .

On note pour  $x \in \mathbb{R}$  :  $f * g(x) = \int_0^a f(y)g(x-y) dy$ .

L'application  $f * g$  est appelée le **produit de convolution** de  $f$  et  $g$ .

**Proposition 8** Si  $f, g \in L_p^1(0, a)$ , alors :  $f * g \in L_p^1(0, a)$ .

*Démonstration.* Pour voir que  $f * g$  est périodique, il suffit de faire un changement de variable dans l'intégrale qui définit cette fonction.

D'autre part :  $\|f * g\|_1 \leq \int_0^a \int_0^a |f(y)g(x-y)| dy dx = \|f\|_1 \|g\|_1$  par périodicité de  $f$  et d'après le théorème de Fubini.  $\square$

D'après le cours de mathématiques de 1<sup>ère</sup> année, vous savez déjà que la convolution sur l'espace des « fonctions » n'admet pas d'élément neutre, c'est pour cela que l'on introduit la notion de distribution et le  $\delta$  de Dirac.

L'intérêt de la convolution est sa propriété *régularisante* (voir la figure 1.5).

**Proposition 9 (convolution, régularisation et coefficients de Fourier)** *Si  $f, g \in L^2_p[0, a]$  alors  $f * g$  est continue, bornée et  $2\pi$ -périodique, et  $c_n(f * g) = a c_n(f) \cdot c_n(g)$  pour tout  $n \in \mathbb{Z}$ .*

*Démonstration.* Première partie admise dans cette version 0.3. D'autre part :

$$\begin{aligned} c_n(f * g) &= \frac{1}{a} \int_0^a \left( \int_0^a f(y)g(x-y) \, dy \right) e^{-2i\pi nx/a} \, dx \\ &= \frac{1}{a} \int_0^a \int_0^a f(y)e^{-2i\pi ny/a} g(x-y)e^{-2i\pi n(x-y)/a} \, dx \, dy \end{aligned}$$

et on conclut par le théorème de Fubini et la périodicité.  $\square$

**Proposition 10 (régularisation - bis)** *Si  $f \in L^1$  et  $g \in C^k$  ( $k \in \mathbb{N}$ ), on a même :  $f * g \in C^k$  et  $(f * g)^{(k)} = f * g^{(k)}$ .*

*Démonstration.* Il s'agit d'une application du théorème de convergence dominée de Lebesgue.  $\square$

Les figures 1.6, 1.7, 1.8, et 1.9 illustrent les propriétés de la convolution sur des images. Les images sont vues comme des fonctions périodiques définies sur  $[0, N_x] \times [0, N_y]$  (où  $N_x$  et  $N_y$  sont respectivement la taille horizontale et verticale de l'image), à valeurs dans  $[0, 127]$  (pour une image dont le niveau de gris est codé classiquement sur 8 bits). Remarquons que les images « informatiques » sont bien sûr discrètes, la définition de la convolution et les propriétés énoncées dans ce chapitre doivent être adaptées. C'est l'objet du chapitre suivant.

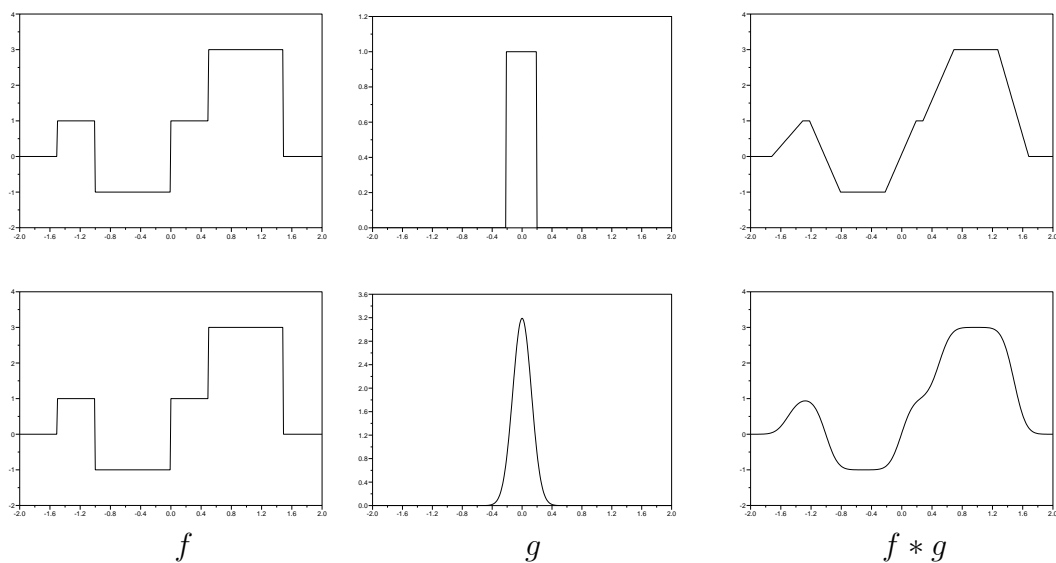


FIG. 1.5 – Convolution. Sur chaque ligne, de gauche à droite :  $f$ ,  $g$ , et  $f * g$ . Le signal  $f$  n'est pas continu. Sur la première ligne  $g$  non plus, mais  $f * g$  est, elle, continue (sans être dérivable). Sur la deuxième ligne  $g$  est en plus dérivable,  $f * g$  le devient aussi. La convolution entraîne un « gain » en régularité. On parle de « régularisation ».

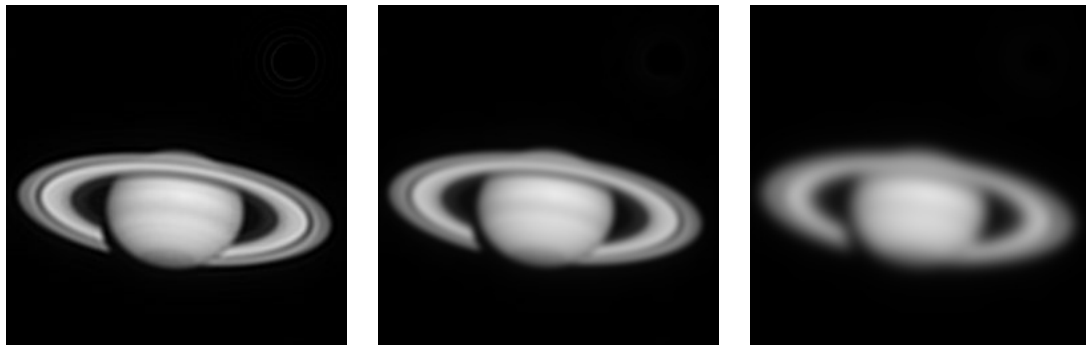


FIG. 1.6 – Convolution d'images. À gauche : image originale. Au milieu : convolution par une Gaussienne d'écart-type 2. À droite : convolution par une Gaussienne d'écart-type 5. Visuellement, plus le noyau de convolution est large (écart-type de la Gaussienne grandissant), plus l'image est floue.

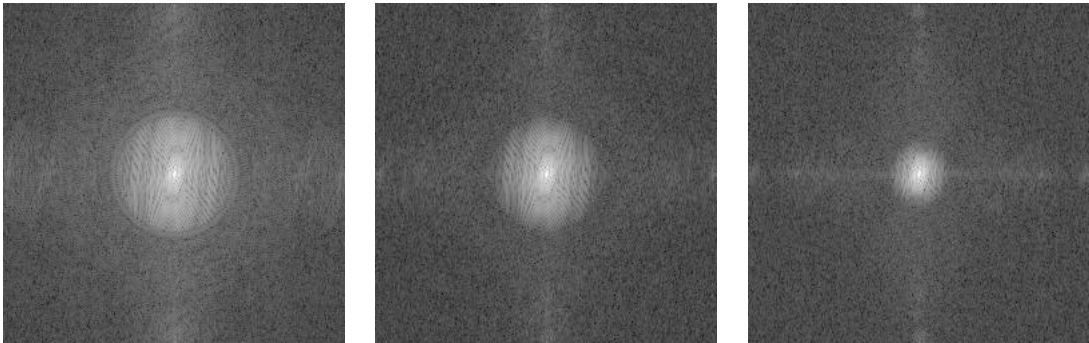


FIG. 1.7 – Convolution d’images et spectre. Les trois images sont les logarithmes des modules des coefficients de Fourier des images de la figure 1.6. Les coefficients sont ordonnés de manière à ce que  $c_{0,0}$  soit au centre de l’image, les hautes (resp. basses) valeurs se traduisent par une luminosité élevée (resp. basse). D’après la proposition 9, le deuxième et le troisième spectre sont obtenus à partir du premier en le multipliant par le spectre de la Gaussienne. Plus la variance de celle-ci est grande, plus son spectre est concentré (voir dans le cours de 1ère année l’expression de la transformée de Fourier d’une Gaussienne), ce qui explique les spectres obtenus. On remarque aussi que plus une image est « régulière », plus son spectre est concentré (et donc plus les coefficients de Fourier décroissent vite lorsque  $|n| \rightarrow +\infty$ ). Cette remarque anticipe sur le chapitre 5.

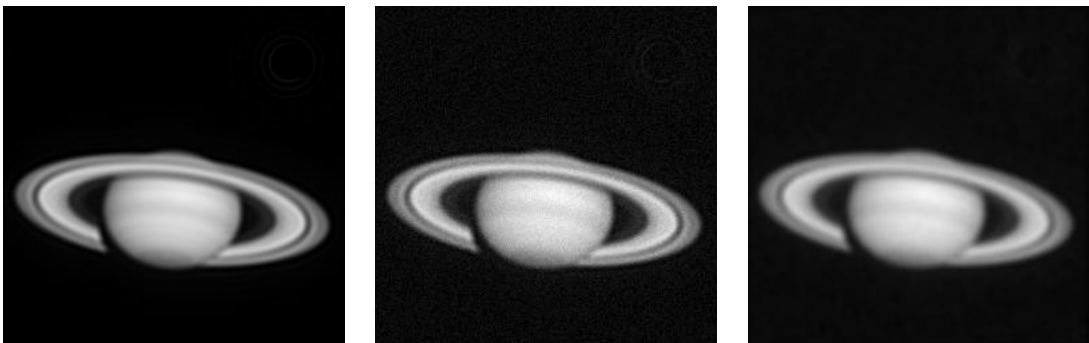


FIG. 1.8 – Application de la régularisation par convolution : « débruitage » naïf. À gauche : image originale. Au milieu : la même image à laquelle on ajoute un bruit blanc Gaussien. La valeur du niveau de gris  $n$  en chaque pixel est transformée en  $n + g$ , où  $g$  est la réalisation d’une variable aléatoire Gaussienne (d’écart-type 20 ici), les valeurs de  $g$  étant indépendantes. À droite : cette image « bruitée » est lissée comme dans l’expérience de la figure 1.6 par convolution par une Gaussienne d’écart-type 2. Le bruit a été atténué, comme bien sûr les détails de l’image originale.



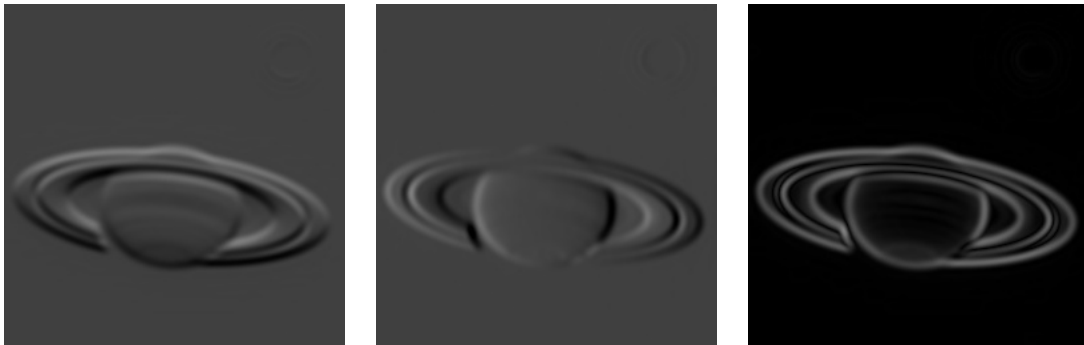


FIG. 1.9 – Illustration de la proposition 10 : détection de contours. L'image de gauche ( $D_x(i, j)$ ) est obtenue à partir de l'image originale par convolution par la dérivée d'une Gaussienne (écart-type 2) selon l'axe des ordonnées. Il revient au même de dériver l'image régularisée. L'image du milieu ( $D_y(i, j)$ ) est obtenue par convolution par la dérivée de la même Gaussienne selon l'axe des abscisses. Un niveau de gris noir correspond à une valeur fortement négative, blanc à une valeur fortement positive, et gris à une valeur proche de 0. À droite : image de la norme du gradient ( $\sqrt{D_x^2(i, j) + D_y^2(i, j)}$ ). Elle correspond aux contours de l'image originale. Le détecteur de contours de Canny est basé sur cette méthode. L'intérêt d'effectuer une convolution par la dérivée d'une Gaussienne est de limiter l'influence du bruit, au prix d'une moins bonne localisation spatiale des contours qui deviennent plus "larges" lorsque l'écart-type est plus élevé. C'est l'idée de la théorie du *scale-space* en traitement d'images / vision par ordinateur.



# Chapitre 2

## Signaux numériques et filtres associés

### 2.1 Signaux numériques

Les signaux (sons, images) traités sur ordinateur sont tous *numériques*. On parle aussi de signal *discret* ou *digital* (anglicisme). La fonction  $f$  à analyser n'est pas connue continûment au cours du temps, on n'en connaît qu'un certains nombres d'échantillons  $y_0 = f(t_0), y_1 = f(t_1), \dots, y_{N-1} = f(t_{N-1})$  à  $N$  instants  $t_0, t_1, \dots, t_{N-1}$ . On supposera dans la suite que les points d'échantillonnage sont régulièrement espacés sur une période  $a$  de  $f : \forall n \in [0, N - 1], t_k = ka/N$ .

On a vu dans la section précédente qu'un certain nombre de résultats intéressants étaient liés à la régularité des signaux. Comme il n'existe pas de « bonne théorie » de la régularité pour les signaux numériques, on est réduit à faire des hypothèses sur la fonction  $f$  originale pour appliquer ces résultats.

Le premier problème est de définir une notion de coefficients de Fourier consistante avec la théorie analogique. Deux points de vue sont possibles : dans le premier on calcule de manière approchée la valeur des coefficients de Fourier, dans la seconde on interpole  $f$  par un polynôme trigonométrique. Ces deux points de vues amènent en fait à la même notion.

**Rappel** : Les coefficients de Fourier de  $f$  de période  $a$  sont donnés par :

$$\forall n \in \mathbb{Z}, c_n = \frac{1}{a} \int_0^a f(t) e^{-2\pi i n t / a} dt.$$

**Premier point de vue.** On calcule la valeur approchée de  $c_n$  ( $n \in \mathbb{Z}$ ) par la méthode des trapèzes (cf figure 2.1, le principe est le même pour une fonction à valeurs complexes).

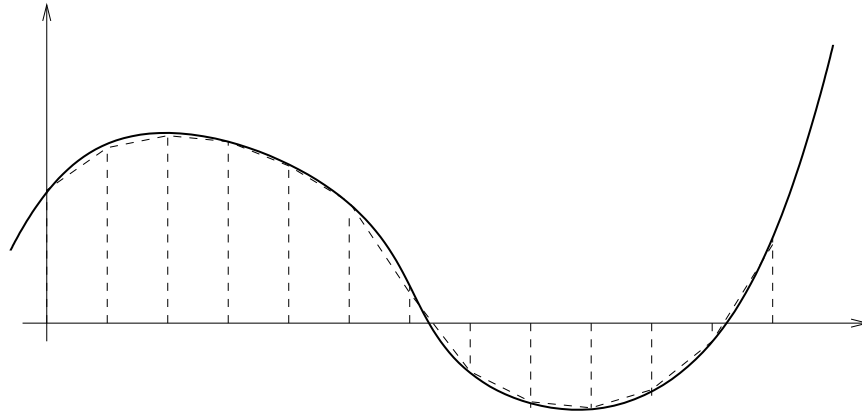


FIG. 2.1 – Méthode des trapèzes pour une fonction réelle. L'intégrale de la fonction réelle représentée est approchée par la somme des aires (signées) des trapèzes s'appuyant sur les intervalles délimités par les points de discrétisation où la valeur de la fonction est connue.

On approche alors  $c_n$  par :

$$c'_n = \frac{1}{a} \sum_{k=1}^N \frac{f((k-1)a/N)e^{-2i\pi(k-1)/N} + f(ka/N)e^{-2i\pi k/N}}{2} \cdot \frac{a}{N}.$$

Après séparation en deux sommes, décalage d'indice, et après avoir remarqué que  $f(0) = f(a)$ , on trouve (à faire en exercice) :

$$\forall n \in \mathbb{Z}, c'_n = \frac{1}{N} \sum_{k=0}^{N-1} f\left(\frac{ka}{N}\right) e^{-2i\pi nk/N}.$$

*Remarque :* la suite  $(c'_n)$  ainsi définie est périodique, de période  $N$ . La suite des coefficients de Fourier tendant vers 0 lorsque  $|n| \rightarrow +\infty$ , seuls les  $N$  coefficients centraux sont des approximations des « vrais » coefficients  $c_n$ .

**Second point de vue.** On cherche les  $N$  coefficients  $c''_n$  du polynôme trigonométrique

$$p(t) = \sum_{n=-N/2}^{N/2-1} c''_n e^{2i\pi nt/a}$$

de manière à ce que ce polynôme interpole  $f$  aux points  $t_k = ka/N, k \in [0, N - 1]$ .

Ce polynôme existe et est unique. Le raisonnement est analogue à celui donnant les coefficients des polynômes interpolateurs de Lagrange. Les coefficients  $c''_n$  vérifient :

$$\sum_{m=-N/2}^{N/2-1} c''_m e^{2i\pi mk/N} = f\left(\frac{ka}{N}\right).$$

D'où pour tout  $n$  :

$$\begin{aligned} \frac{1}{N} \sum_{k=0}^{N-1} f\left(\frac{ka}{N}\right) e^{-2i\pi nk/N} &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{m=-N/2}^{N/2-1} c''_m e^{2i\pi(n-m)k/N} \\ &= \frac{1}{N} \sum_{m=-N/2}^{N/2-1} c''_m \sum_{k=0}^{N-1} e^{2i\pi(n-m)k/N}. \end{aligned}$$

Or  $\sum_{k=0}^{N-1} e^{2i\pi(n-m)k/N} = 0$  si  $n \neq m$  et  $= N$  sinon.

D'où finalement  $c''_n = c'_n$  pour  $n \in [-N/2, N/2 - 1]$ .

*Remarque* : les coefficients de Fourier du polynôme trigonométrique  $p$  sont aussi les  $c''_n$ . (à cause de la propriété d'orthogonalité des  $e_n$ .)

**Conclusion.** On est amené à définir la Transformée de Fourier Discrète d'un signal numérique ( $y_n$ ) par la formule donnant les  $c'_n$  (ou les  $c''_n$ ). Elle peut être vue :

1. soit comme la périodisée de la suite des approximations par la méthode des trapèzes des  $N$  coefficients centraux de la transformée de Fourier de la fonction échantillonnée par les  $y_n$  ;
2. soit comme les coefficients d'un polynôme trigonométrique interpolant les  $y_n$  (et donc comme les coefficients de Fourier de ce polynôme).

Dans le chapitre 6 (paragraphe 6.5) nous reviendrons sur la transformée de Fourier discrète vue comme une approximation de la transformée de Fourier (continue) de la fonction échantillonnée par les  $y_n$ . En particulier, nous verrons dans quelle mesure cette approximation est valide.

### 2.1.1 La Transformée de Fourier Discrète

**Définition 10** On appelle *Transformée de Fourier Discrète* (ou *TFD*) de la suite  $(y_n)_{n \in [0, N-1]}$  la suite des  $(Y_n)_{n \in [0, N-1]}$  définie par

$$Y_n = \frac{1}{N} \sum_{k=0}^{N-1} y_k \omega_N^{-nk} \quad (2.1)$$

où  $\omega_N = e^{2i\pi/N}$ .

*Remarque :* les suites  $(y_n)$  et  $(Y_n)$  peuvent être vues comme des suites sur  $\mathbb{Z}$  de période  $N$ .

**Théorème 8 (Transformée de Fourier Discrète inverse)** *On peut retrouver les  $y_n$  à partir des  $Y_n$  à l'aide de la formule suivante.*

$$y_n = \sum_{k=0}^{N-1} Y_k \omega_N^{nk}. \quad (2.2)$$

*Démonstration.* Par simple calcul :

$$\begin{aligned} \sum_{k=0}^{N-1} Y_k \omega_N^{nk} &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} y_l \omega_N^{k(n-l)} \\ &= \frac{1}{N} \sum_{l=0}^{N-1} y_l \sum_{k=0}^{N-1} \omega_N^{k(n-l)} \\ &= y_n. \end{aligned}$$

car  $\sum_{k=0}^{N-1} \omega_N^{ki}$  vaut 0 si  $i \neq 0$ , et  $N$  sinon.  $\square$

*Remarque : **attention !*** Dans certains logiciels, comme MATLAB ou SCILAB, la transformée de Fourier discrète est donnée par la formule (2.2) (fonction `fft`) et la transformée de Fourier discrète inverse est donnée par la formule (2.1) (fonction `ifft`). Il faut donc adapter toutes les formules données dans la suite (par exemple égalité de Parseval, transformée de Fourier d'une convolution, etc) pour retrouver numériquement les mêmes propriétés.

**Proposition 11 (égalité de Parseval pour les signaux numériques)**

$$\sum_{n=0}^{N-1} |y_n|^2 = N \sum_{n=0}^{N-1} |Y_n|^2. \quad (2.3)$$

*Démonstration.* Encore une fois il s'agit d'un simple calcul :

$$\begin{aligned} \sum_{n=0}^{N-1} |Y_n|^2 &= \sum_{n=0}^{N-1} \frac{1}{N^2} \left( \sum_{k=0}^{N-1} y_k \omega_N^{-nk} \right) \left( \sum_{l=0}^{N-1} \bar{y}_l \omega_N^{nl} \right) \\ &= \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} y_k \bar{y}_l \sum_{n=0}^{N-1} \omega_N^{n(l-k)}. \end{aligned}$$

Or  $\sum_{n=0}^{N-1} \omega_N^{n(l-k)} = N \delta_{l,k}$  où  $\delta$  est le symbole de Kronecker, ce qui permet de conclure.  $\square$

**Définition 11** Soient deux suites  $x$  et  $y$  de période  $N$ , on définit le **produit de convolution discret** :  $z = x * y$  où  $\forall n \in \mathbb{Z}$ ,  $z_n = \sum_{k=0}^{N-1} x_k y_{n-k}$ .

*Remarque* :  $z$  est alors également une suite périodique, de période  $N$ .

**Proposition 12 (TFD d'une convolution)** Soient deux suites  $x$  et  $y$  de période  $N$ ,  $z$  le produit de convolution  $z = x * y$ , et  $X, Y, Z$  les transformées de Fourier discrètes respectives.

Alors :  $\forall n \in \mathbb{Z}$ ,  $Z_n = NX_n Y_n$ .

*Démonstration.* C'est encore un jeu avec les définitions :

$$\begin{aligned} Z_n &= \frac{1}{N} \sum_{k=0}^{N-1} z_k \omega_N^{-nk} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} x_l y_{k-l} \omega_N^{-nk} \\ &= \frac{1}{N} \left( \sum_{l=0}^{N-1} x_l \omega_N^{-nl} \right) \left( \sum_{k'=0}^{N-1} x_{k'} \omega_N^{-nk'} \right) \end{aligned}$$

(après le changement  $k' = l - k$  dans la deuxième somme).

Ceci permet de conclure. □

## 2.1.2 Transformée de Fourier Rapide

Si on évalue la transformée de Fourier d'un signal numérique  $(x_n)$  de longueur  $N$  par la formule (2.1), le nombre d'opérations élémentaires nécessaires est en  $\mathcal{O}(N^2)$  (somme de  $N$  termes à répéter pour les  $N$  coefficients  $X_n$ ). Un algorithme tirant partie des relations trigonométriques sur les puissances de  $\omega_N$  permet d'abaisser ce nombre d'opérations à  $\mathcal{O}(N \log(N))$ . Il s'agit d'une avancée cruciale pour le calcul effectif des TFD. Sans cet algorithme, la « révolution numérique » n'aurait tout simplement pas eu lieu !

En effet, le calcul de la TFD d'une seconde de signal sonore échantillonné à 44.1 kHz (qualité CD) nécessite de l'ordre de  $2 \cdot 10^9$  opérations dans le premier cas (complexité en  $\mathcal{O}(N^2)$ ), et de l'ordre de  $7 \cdot 10^5$  opérations dans le second (complexité en  $\mathcal{O}(N \log(N))$ ). Sur une machine permettant de traiter une opération élémentaire à la fréquence de 1 GHz, il faut respectivement 2 secondes et  $7 \cdot 10^{-4}$  seconde. On voit que

dans le second cas un traitement « temps-réel » est accessible, contrairement au premier cas.

**Hypothèse :** dans la suite,  $N$  est une **puissance de 2**.

Le cas où  $N$  n'est pas une puissance de 2 nécessite des adaptations pour s'y ramener. On ne détaillera pas ces adaptations ici, remarquons seulement qu'une manière simple de faire est de compléter le signal par des 0, de manière à atteindre une longueur qui soit une puissance de 2.

Soit  $(x_n)$  un signal numérique de longueur  $N$ , et  $(X_n)$  sa transformée de Fourier discrète.

*Notation :* on introduit les polynômes :

$$Q(X) = \sum_{k=0}^{N/2-1} x_{2k} X^k \quad \text{et} \quad R(X) = \sum_{k=0}^{N/2-1} x_{2k+1} X^k.$$

Alors :  $X_n = Q((\omega_N^{-n})^2) + \omega_N^{-n} R((\omega_N^{-n})^2)$ , d'où :

$$X_n = Q(\omega_{N/2}^{-n}) + \omega_N^{-n} R(\omega_{N/2}^{-n}). \quad (2.4)$$

*Remarque :*  $\omega_{N/2}^{-n} = \omega_{N/2}^{N/2-n}$  si  $n \geq N/2$ .

*Remarque :*  $(Q(\omega_{N/2}^{-n}))_{0 \leq n < N/2}$  et  $(R(\omega_{N/2}^{-n}))_{0 \leq n < N/2}$  apparaissent alors comme les Transformées de Fourier Discrètes de signaux de taille  $N/2$ .

Nous venons donc de mettre en évidence un algorithme itératif pour le calcul de la transformée de Fourier discrète, appelé **Transformée de Fourier Rapide**, TFR ou **Fast Fourier Transform**, FFT (Cooley-Tuckey 1965). Cf figure 2.2.

**Théorème 9** *Le nombre d'opérations  $T(N)$  nécessaire au calcul de la transformée de Fourier d'un signal discret de longueur  $N$  vérifie  $T(N) = 2T(N/2) + 3N/2$ .*

*Donc :*  $T(N) = \mathcal{O}(N \log(N))$ .

*Démonstration.* Il faut  $T(N/2)$  opérations pour calculer les valeurs prises par  $Q$  et  $R$  d'après la remarque sur la formule (2.4),  $N/2$  opérations pour calculer les  $\omega_N^{-n}$ , et  $2N$  opérations pour calculer les  $X_n$  (à chaque fois une somme et un produit).

Une somme télescopique (exploitant que  $N$  est une puissance de 2) permet de déduire :  $T(N) = \frac{3N}{2} \log_2(N) + NT(1)$  ; d'où finalement  $T(N) = \mathcal{O}(N \log(N))$ .  $\square$



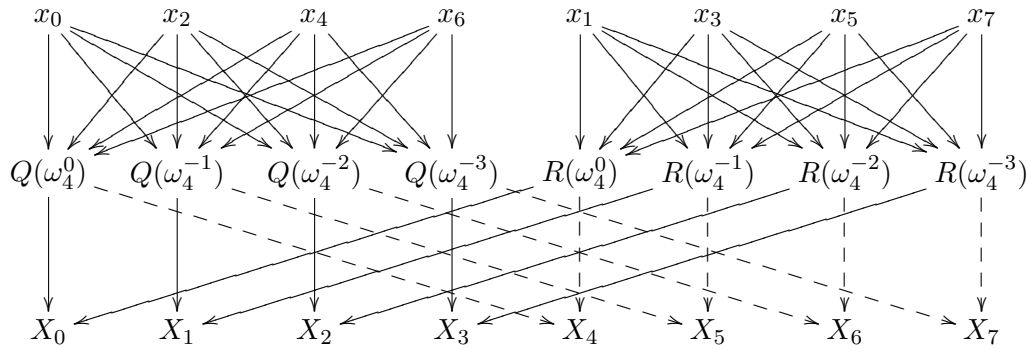


FIG. 2.2 – Schéma de l’algorithme de la Transformée de Fourier Rapide pour un signal de longueur  $N = 8$ . Le calcul intermédiaire est en fait équivalent au calcul de la transformée de Fourier des deux signaux de longueur  $N/2$  ( $x_0, x_2, x_4, x_6$ ) et ( $x_1, x_3, x_5, x_7$ ).

*Remarque :* ceci justifie le nom « Transformée de Fourier Rapide ». L’évaluation du nombre d’opérations n’a de sens que si on suppose que toutes les opérations élémentaires (additions, multiplications) s’exécutent dans le même nombre de cycles processeurs.

### 2.1.3 La transformée de Fourier 2D

Soit  $(x_{n,m})_{\substack{0 \leq n < N \\ 0 \leq m < N}}$  un signal discret 2D (par exemple une image de taille  $N \times N$ ).

**Définition 12 (Transformée de Fourier Discrète 2D)** On appelle **Transformée de Fourier Discrète 2D** (ou **TFD 2D**) de la suite  $(x_{n,m})_{\substack{0 \leq n < N \\ 0 \leq m < N}}$  la suite des  $(X_{n,m})_{\substack{0 \leq n < N \\ 0 \leq m < N}}$  définie par :

$$X_{n,m} = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} x_{k,l} \omega_N^{-mk} \omega_N^{-nl}$$

où  $0 \leq m < N, 0 \leq m < N$ .

Remarquons que l’on peut écrire  $X_{n,m}$  sous la forme :

$$X_{n,m} = \frac{1}{N} \sum_{k=0}^{N-1} \left( \frac{1}{N} \sum_{l=0}^{N-1} x_{k,l} \omega_N^{-nl} \right) \omega_N^{-mk}$$

Pour calculer les  $X_{n,m}$ , il suffit de calculer la TFD-1D sur les colonnes du tableau des  $x_{n,m}$ , puis de calculer la TFD-1D sur les lignes du tableau résultant. Bien sûr, on peut commencer par opérer sur les lignes puis sur les colonnes.

Le calcul des  $X_{n,m}$  revient donc au calcul de  $2N$  TFD-1D, et à l'aide de la FFT on peut calculer la TFD-2D en  $\mathcal{O}(N^2 \log(N))$  opérations.

*Remarque :* on peut montrer que toutes les propriétés précédemment énoncées pour la TFD-1D restent valables.

On reviendra sur la TFD 2D dans la partie 2.3.2.

## 2.2 Filtres numériques

### 2.2.1 Définitions

De même que dans la définition 1, une fonction transformant un signal numérique en un autre signal numérique est **invariante** (ou **stationnaire**) si une translation des indices de l'entrée entraîne la même translation dans les indices de la sortie.

**Définition 13** On appelle **filtre** toute fonction linéaire et invariante transformant un signal numérique en un signal numérique.

**Définition 14** Soit  $\delta$  le signal numérique de Dirac :  $\delta_n = 0$  si  $n \neq 0$  et  $\delta_0 = 1$ .

On appelle **réponse impulsionnelle** du filtre  $f$  le signal numérique  $f(\delta)$ .

**Théorème 10** Un filtre  $f$  est caractérisé par sa réponse impulsionnelle  $h$  et vérifie :  $f(x) = h * x$  (convolution discrète).

*Démonstration.* Soit  $x = (x_n)_{n \in \mathbb{Z}}$  un signal quelconque. Notons  $T_m$  l'opérateur qui décale les indices d'un pas  $m$ . Avec ces notations, le signal  $x$  s'écrit comme la combinaison linéaire :  $x = \sum_{n \in \mathbb{Z}} x_n T_n(\delta)$ .

Par linéarité de  $f$  :  $f(x) = \sum_{n \in \mathbb{Z}} x_n f(T_n(\delta))$ .

$f$  étant invariant :  $f(x) = \sum_{n \in \mathbb{Z}} x_n T_n(f(\delta))$ .

Notons  $h$  le signal  $h = f(\delta)$ . Si  $y = f(x)$ , alors :  $\forall m \in \mathbb{Z}, y_m = \sum_{n \in \mathbb{Z}} x_n h(m - n)$ .

D'où  $f(x) = h * x$ . □

**Définition 15** Un filtre est dit à **réponse impulsionnelle finie** (ou **RIF** ou **FIR**) si le support de sa réponse impulsionnelle est finie, et à **réponse impulsionnelle infinie** (ou **RII** ou **IIR**) dans le cas contraire.

**Définition 16** Si  $\forall n < 0, h_n = 0$ , alors le filtre est dit **causal**. Le calcul du coefficient de la sortie  $y_m$  ne nécessite que des observations  $x_n$  pour  $n \leq m$  (dans le « passé »).

**Définition 17** Un filtre qui transforme les signaux d'entrée bornés en signaux de sortie bornés est dit **stable**. On montre que ceci est équivalent à  $\|h\|_1 = \sum_{n \in \mathbb{Z}} |h_n| < +\infty$ .

### 2.2.2 Transformée en $z$

**Définition 18** La transformée en  $z$  d'un signal numérique  $x = (x_n)_{n \in \mathbb{Z}}$  est la fonction de la variable complexe :  $\mathcal{Z}(x)(z) = \sum_{n \in \mathbb{Z}} x_n z^{-n}$ .

*Remarque :* cette série de Laurent ne converge *a priori* que sur une couronne du plan complexe  $\mathbb{C}$ . On dispose des outils de l'analyse complexe (décomposition en éléments simples, méthode des résidus pour récupérer un signal numérique dont on donne la transformée en  $z$ , par exemple sous la forme d'une fraction rationnelle en  $z$ ).

**Proposition 13** Quelques propriétés vérifiées par la transformée en  $z$  (notée  $\mathcal{Z}$ ) :

- linéarité :  $\mathcal{Z}(\alpha x + \alpha' x') = \alpha \mathcal{Z}(x) + \alpha' \mathcal{Z}(x')$
- décalage temporel :  $\mathcal{Z}(x(\cdot - k)) = z^{-k} \mathcal{Z}(x)$
- convolution :  $\mathcal{Z}(x * x') = \mathcal{Z}(x) \mathcal{Z}(x')$ .

*Démonstration.* Il s'agit d'un simple calcul. □

### 2.2.3 Transformée en $z$ des filtres FIR et IIR

La transformée en  $z$  de la réponse impulsionnelle (appelée fonction de transfert) s'écrit sous la forme  $H(z) = \sum_{k=-\infty}^{\infty} a_k z^{-k}$ . Il s'agit du développement en série de Laurent autour de la singularité  $z = 0$ .

Deux cas sont possibles.

**Premier cas.** On peut écrire  $H$  sous forme de fraction rationnelle (singularité d'ordre fini) :

$$H(z) = \frac{\sum_{k=0}^n \alpha_k z^{-k}}{\sum_{k=0}^m \beta_k z^{-k}} = \frac{A(z)}{B(z)}.$$

On peut donc écrire (cf proposition 13) :  $B(z)Y = A(z)X$ .

Ceci se traduit (par transformée en  $z$  inverse) par une écriture du genre :

$$\forall p \in \mathbb{Z}, \sum_{k=0}^m \beta_k y_{p-k} = \sum_{k=0}^n \alpha_k x_{p-k},$$

appelée *équation aux différences*.

A ce stade, on constate que le filtre FIR est réalisable (i.e. on dispose d'une formule pour calculer les valeurs successives des  $y_n$ ), sous forme non-réursive si  $\beta_k = 0$  pour  $k > 0$  et sous forme réursive sinon.

Dans le cas de la formule non-réursive, la fonction de transfert  $H$  est un polynôme en  $z^{-1}$ , donc les  $a_k$  sont nuls pour  $k < 0$ . Dans ce cas, le filtre est donc causal, et à réponse impulsionnelle finie. On voit facilement que le filtre est alors stable.

Dans le cas de la formule réursive, il s'agit d'un filtre à réponse impulsionnelle infinie. Si  $\beta_0 \neq 0$ , le filtre est causal. Posons-nous alors la question de la stabilité. Remarquons que dans l'équation  $B(z)Y = A(z)X$ , on peut décomposer le polynôme  $B(z)$  en  $\prod_{k=1}^m (x_k - X^{-1})$  où les  $x_k$  sont des complexes, pôles de la fraction rationnelle en  $1/z$  associée à  $H(z)$ .

Comme  $Y = B^{-1}(z)A(z)X$  et :

$$(x_k - X^{-1})^{-1} = \frac{1}{x_k} + \frac{X^{-1}}{x_k^2} + \frac{X^{-2}}{x_k^3} + \frac{X^{-3}}{x_k^4} + \dots$$

on conclut que le filtre n'est stable que si tous les  $x_k$  vérifient :  $|x_k| > 1$ . Comme les pôles de  $H(z)$  sont les inverses des  $x_k$ , on peut dire que le filtre est stable si les pôles de la fonction de transfert  $H$  sont à l'intérieur du disque unité de  $\mathbb{C}$ .

**Second cas.** On ne peut pas écrire  $H$  sous la forme d'une fraction rationnelle. Dans ce cas l'équation aux différences ne donne pas une formule pour « calculer » numériquement la sortie du filtre avec un nombre fini d'échantillons.

*Remarque :* dans le problème de la *synthèse* de filtres numériques, on impose des conditions sur sa fonction de transfert (de manière à atténuer telle bande de fréquences, laisser passer telle autre bande, etc), et on retrouve sa réponse impulsionnelle par les outils de l'analyse complexe à partir de la transformée en  $z$ . On peut effectivement inverser la transformée en  $z$  par une intégrale curviligne :

$$x_n = \frac{1}{2\pi i} \oint_C \mathcal{Z}(z) z^{n-1} dz,$$

où  $C$  est un cercle inclus dans la couronne de convergence de  $\mathcal{Z}(z)$ , parcouru dans le sens direct. Cette dernière intégrale se calculant dans les cas favorables par le théorème des résidus.

## 2.3 Signaux numériques en pratique

### 2.3.1 Analyse d'une note de musique

Nous nous intéressons dans ce paragraphe à l'enregistrement d'un *la* de guitare. Les informations fournies dans le fichier indiquent que la fréquence d'échantillonnage est

$f = 44100$  Hz (qualité CD) et que le fichier comporte  $N = 232041$  échantillons. La durée du signal est donc  $N/f = 5.2617$  secondes.

La figure 2.3 montre les modules de la transformée de Fourier du signal discret. On observe que le pic le plus élevé a les indices 2321 (pic de droite) et 229722 (pic de gauche). Les logiciels de calcul comme MATLAB numérotent la transformée de Fourier de manière à ce que l'indice 1 correspondent à  $c_0$ , l'indice 2 à  $c_1$ , ..., l'indice  $N/2$  à  $c_{N/2-1}$ , l'indice  $N/2+1$  à  $c_{-N/2}$ , l'indice  $N/2+2$  à  $c_{1-N/2}$  ... l'indice  $N$  à  $c_{-1}$ . Donc les deux pics correspondent respectivement à  $c_{2320}$  et  $c_{-2320}$ . Ces coefficients correspondent aux composantes  $\exp(-2i\pi nt/a)$  et  $\exp(2i\pi nt/a)$  du signal (avec  $n = 2320$  et  $a$  la longueur du signal en secondes), soit à la fréquence  $n/a = 440,9221$  Hz, très proche de la fréquence attendue pour un *la* (440 Hz).

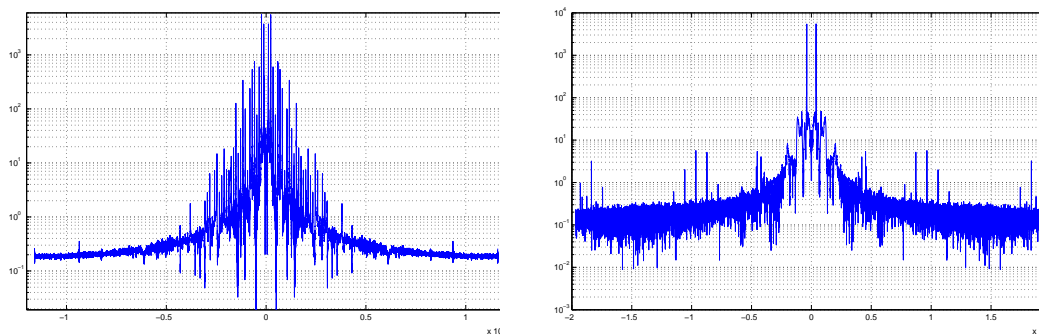


FIG. 2.3 – Transformée de Fourier de la note *la* d'une guitare (à gauche) et d'un diapason (à droite). En abscisse : indice  $n$ , centré autour de  $c_0$ ; en ordonnée : module du coefficient de Fourier  $|c_n|$  (échelle logarithmique). Observez la symétrie (signal réel) et la décroissance des coefficients de Fourier lorsque  $|n|$  augmente. Observez que le signal de la guitare contient davantage d'harmoniques que celui du diapason.

### 2.3.2 Interprétation des images dans le domaine fréquentiel

On utilise ici la transformée de Fourier 2D décrite en section 2.1.3. Dans toutes les images représentant des transformées de Fourier, le coefficient  $c_{0,0}$  sera au centre. Les valeurs faibles des modules des coefficients sont en noir, les valeurs élevées en blanc.

#### Quelques images synthétiques

Les images de la figure 2.4 montrent quatre images (à gauche), et les transformées de Fourier respectives (à droite). Commentons ces résultats.

Dans le premier cas, la transformée de Fourier 2D d'une ligne horizontale est une ligne verticale. En effet, la transformée de Fourier 2D correspond à une transformée de

Fourier 1D selon les colonnes, puis une transformée de Fourier 1D selon les lignes (ou inversement, les deux opérations commutent). Donc après une première transformée de Fourier selon les colonnes de l'image (qui sont toutes identiques), toutes les lignes ont la même valeur. Donc la transformée de Fourier des lignes constantes est nulle partout, excepté le coefficient central (correspondant à la moyenne). Faites en exercice le même raisonnement en commençant par les lignes.

Dans le deuxième cas, chaque ligne crée dans le domaine de Fourier une ligne de direction perpendiculaire, qui sont superposées car la transformée de Fourier est linéaire. Les effets de discrétisation de la ligne oblique créent des artefacts supplémentaires.

On peut dire que la contribution au spectre d'un segment figurant dans l'image est une droite passant par le centre (le spectre est symétrique) orientée perpendiculairement.

Dans le troisième cas, chaque ligne contribue sur une droite perpendiculaire à son orientation, donc ici se cumulent sur la diagonale. Néanmoins, si les coefficients de Fourier apportés par les lignes sont grossièrement les mêmes en module, les translations de pas fixe  $N/k$  entre chaque ligne ( $N$  : nombre de pixels dans la diagonale,  $k$  : nombre de droites obliques) créent des déphasages de  $2\pi nl/k$  sur le  $n$ -ème coefficient de Fourier apporté par la  $l$ -ème ligne (d'après la formule 1 de la proposition 13 page 35). La somme des  $\exp(2i\pi nl/k)$  pour  $l$  entre 1 et  $k$  est nulle sauf si  $n/k$  est entier ; ainsi les coefficients non-nuls sur la diagonale ne peuvent être écartés que de  $k$  (proportionnel à la fréquence d'alternance des lignes). À faire en exercice : écrire le raisonnement en partant de l'expression de la Transformée de Fourier Discrète (définition 10).

Dans le quatrième cas, il y a superposition des deux réseaux de lignes, créant une superposition sur le spectre des points le long de droites perpendiculaires aux deux directions principales. Des points additionnels correspondent aux répétitions dans le réseau superposé selon des directions qui sont combinaisons des deux directions principales.

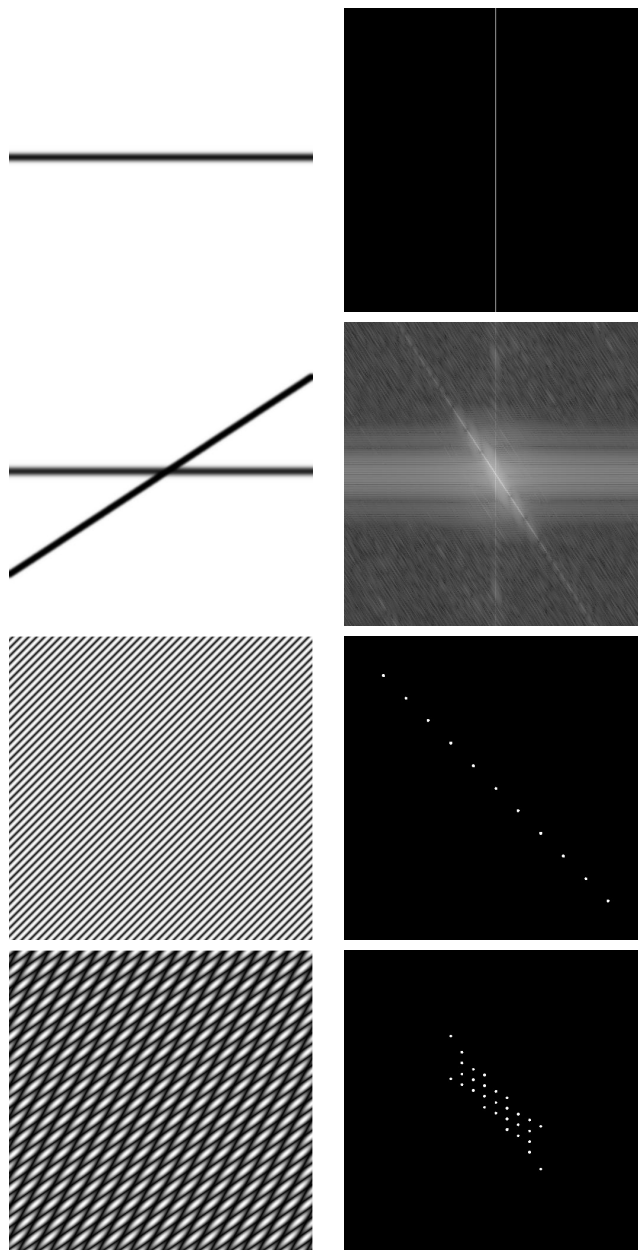


FIG. 2.4 – Quatre images (à gauche) et les transformées de Fourier respectives (à droite). Les deux premières transformées correspondent en fait à un niveau de gris proportionnel au logarithme du module des coefficients, tandis que les deux dernières montrent (pour une meilleure lisibilité) en blanc les coefficients dont le module est supérieur à 10% du plus grand module.

**Images « naturelles »**

La figure 2.5 présente la transformée de Fourier de trois images. Dans les trois cas, aucune structure particulière n'apparaît dans l'image des arguments des coefficients de Fourier.

Au niveau des modules, on voit apparaître trois lignes pour l'image 1 : une horizontale, une verticale correspondant respectivement aux verticales et aux horizontales de l'image, ainsi qu'une ligne oblique correspondant probablement à la frontière entre visage et chapeau. La deuxième image a été tournée de 30 degrés par rapport à la première, ce qui se traduit par la même rotation sur les trois lignes dans le spectre. On remarque qu'on peut toujours deviner une ligne horizontale et une ligne verticale. Cela est visible dans quasiment toutes les images naturelles. Comme la transformée de Fourier suppose implicitement l'image périodique, si les pixels du bord haut (resp. gauche) n'ont pas un niveau de gris proche de ceux du bord bas (resp. droit), alors les bords haut/bas (resp. gauche/droite) contribuent sur la ligne verticale (resp. horizontale) du spectre. De plus, on peut voir ici des lignes secondaires obliques provenant de l'effet d'*aliasing* ou de *repliement de spectre* (étudié au chapitre 6) provoqué par les contours artificiellement créés par la rotation des bords de l'image.

Dans la troisième image, les horizontales légèrement penchées et grossièrement périodiques créent une ligne verticale légèrement oblique et des « répliques » parallèles ; tandis que les verticales (sur les fenêtres) contribuent à l'horizontale.



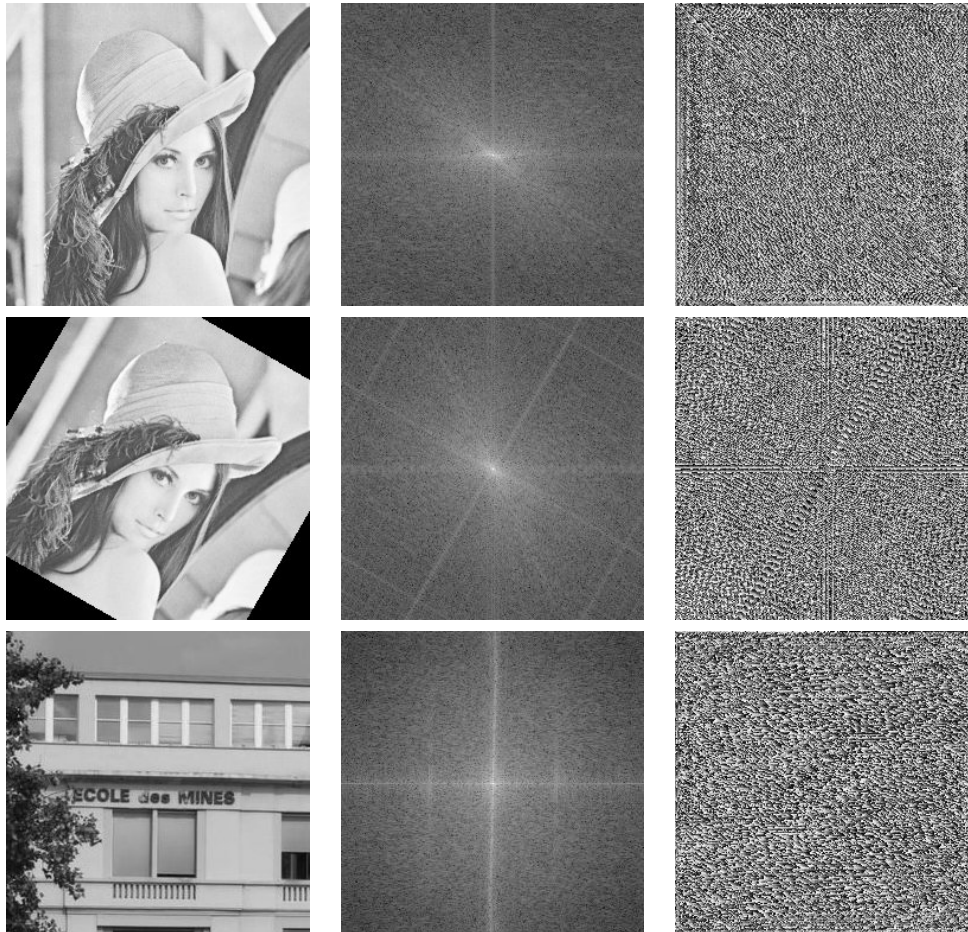


FIG. 2.5 – Trois exemples. À gauche, image originale. Au milieu : logarithme du module des coefficients de Fourier. À droite : arguments des coefficients. En haut, image « Lena ». En bas, image « École des Mines ». Observez la décroissance des coefficients (les coins s’assombrissent dans l’image des modules), les lignes montrant les directions principales dans les images, et les « spots » correspondant aux phénomènes périodiques. À première vue, aucune structure particulière n’apparaît dans l’image des arguments.

## Textures

Une texture est une région dans une image qui présente des caractéristiques visuelles homogènes, comme par exemple un motif qui se répète approximativement.

La figure 2.6 montre le spectre de deux images de texture. Dans le premier cas, l'analyse du spectre (qui présente cinq « spots ») montre que le motif de base de la texture est à symétrie d'ordre 5 (il ressemble effectivement à un pentagone). Dans le deuxième cas, les multiples spots le long de lignes horizontales montrent que l'image présente des répétitions le long de verticales régulièrement espacées.

L'expérience de la figure 2.7 montre que le module des coefficients de Fourier suffit à caractériser certaines textures. Dans cette expérience, le module est celui de la texture correspondant à celle de la figure 2.6, et l'argument des coefficients est aléatoire (de manière à respecter les conditions de symétrie assurant que la partie imaginaire de l'image reconstruite est nulle). Ceci s'explique par la formule 1 de la proposition 13 page 35 : la phase contient une information de localisation qui n'est pas essentielle dans le cas de « micro »-textures.

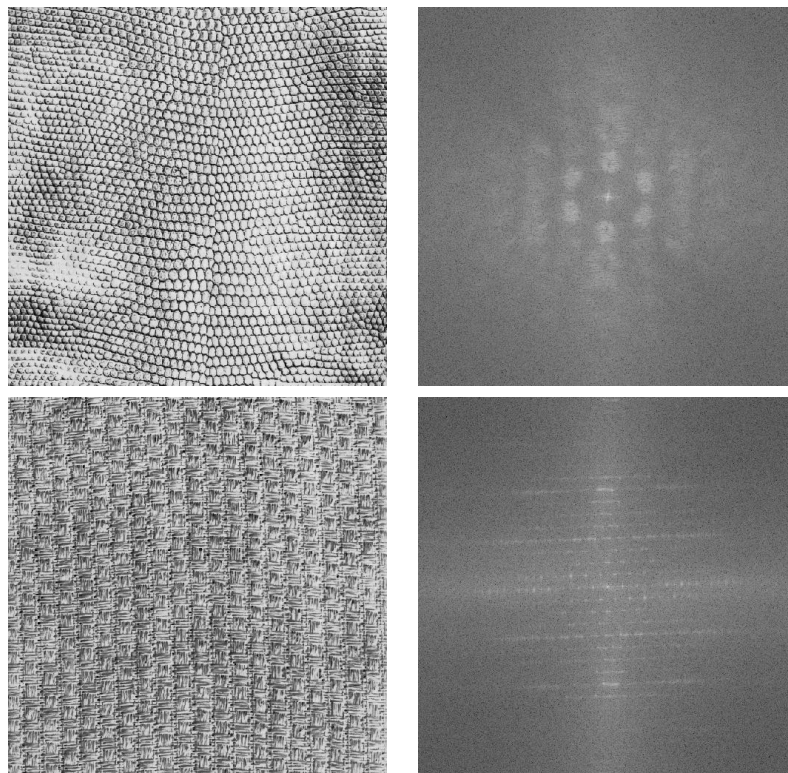


FIG. 2.6 – À gauche : images de texture. À droite : logarithme des modules des coefficients de Fourier.

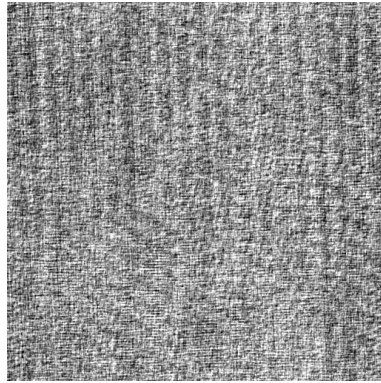


FIG. 2.7 – Synthèse d'une texture à partir des modules des coefficients de l'image de la figure 2.6 et une phase aléatoire. On s'aperçoit que la texture originale est bien caractérisée par le module des coefficients de Fourier.

### Filtrage

L'expérience de la figure 2.8 met en évidence le rôle respectif des hautes fréquences et des basses fréquences. L'image reconstruite à partir des seules basses fréquences est floue. On voit également apparaître le phénomène de ringing que nous étudierons en détails au chapitre 7. L'image reconstruite à partir des seules hautes fréquences ne fait apparaître que les contours de l'image originale.

En effet, les zones uniformes de l'image contribuent au spectre au niveau des coefficients centraux (autour de  $c_{0,0}$ ). Une image constante aurait par exemple un spectre nul, sauf  $c_{0,0}$ . Au contraire, les contours de l'image sont des zones de transitions « clair / foncé » importantes, nécessitant des fréquences élevées pour être reconstruites. Donc en annulant les hautes fréquences ne restent que les zones uniformes dans l'image, et en annulant les basses fréquences ces zones uniformes prennent toutes la même valeur, et il ne reste que les contours. Faites le lien avec les résultats de la section 1.4 page 21.

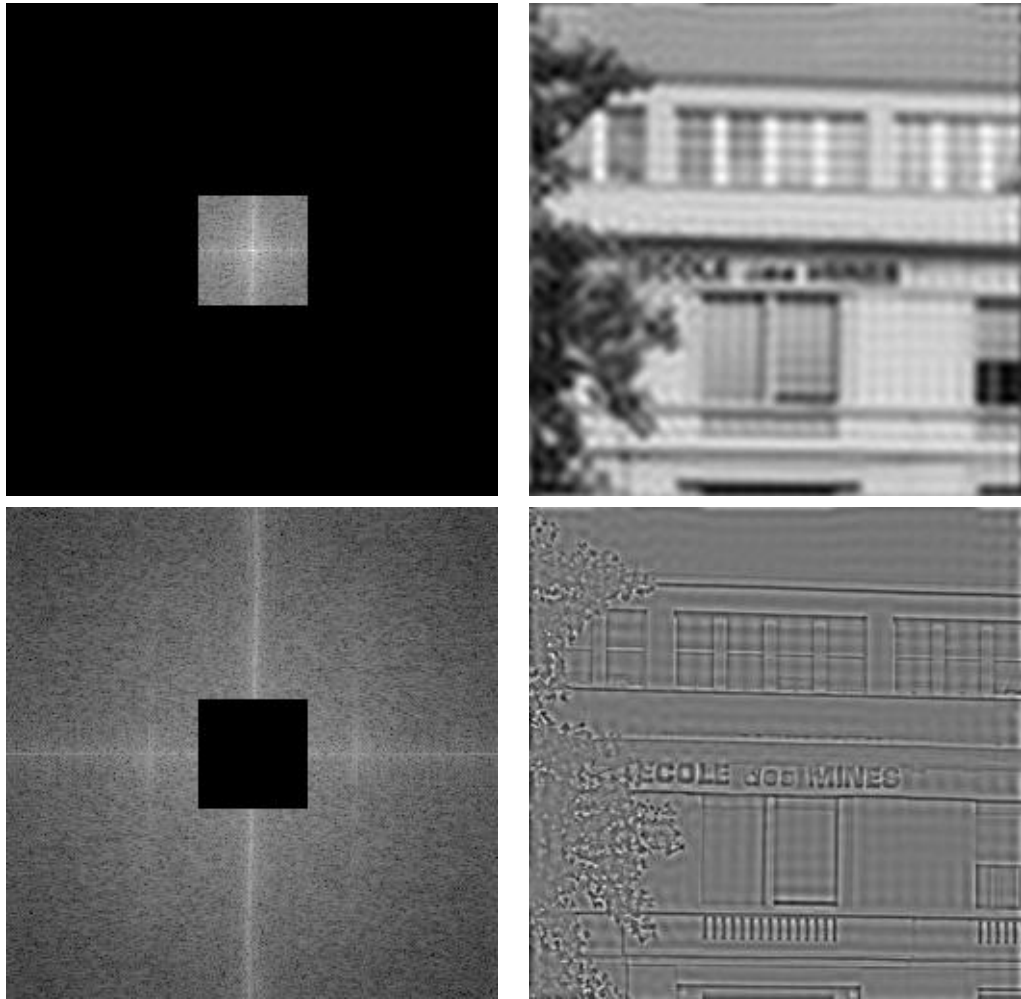


FIG. 2.8 – Filtrage passe-bas (en haut) et passe-haut (en bas).

### Expérience d'Openheim et Lim

L'expérience de la section 2.3.2 montre que le module des coefficients de Fourier suffit à caractériser certaines textures. D'autre part, la figure 2.5 ne laisse pas présager que l'argument puisse contenir beaucoup d'information. Néanmoins, en général, l'argument de la transformée de Fourier n'est absolument pas négligeable. La célèbre expérience de la figure 2.9<sup>1</sup> montre même qu'une part essentielle de l'information s'y trouve. Si dans le module on trouve la « puissance » de la contribution des différentes composantes fréquentielles, l'argument correspond à la position des éléments à reconstruire (d'après la formule 1 de la proposition 13 page 35).

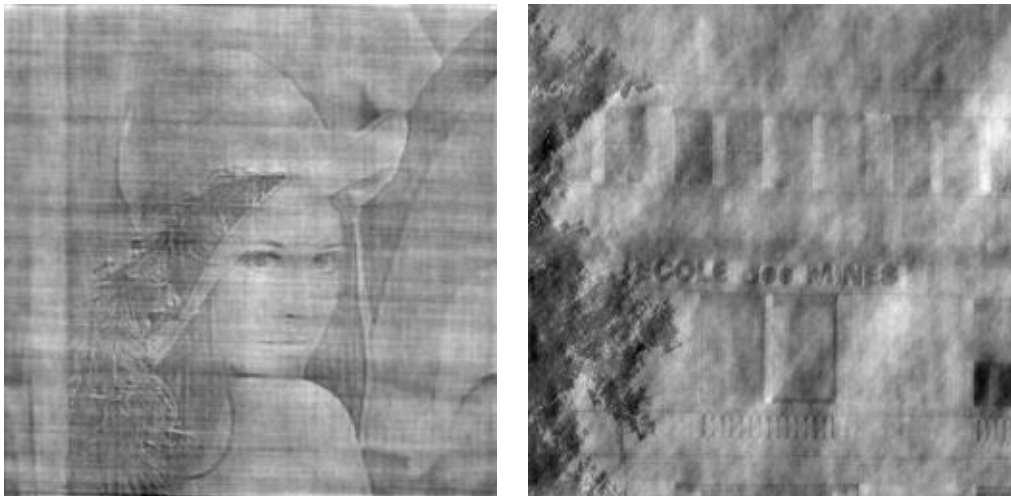


FIG. 2.9 – Expérience d'Openheim et Lim. La première image correspond à la reconstruction obtenue avec le module de l'image « École des Mines » et la phase de « Lena », et la seconde image est obtenue avec le module de « Lena » et la phase de « École des Mines ». L'information de « position » dans la reconstruction est contenue dans la phase, le module sert à fixer l'« intensité » des contours (hautes fréquences) et l'aspect / la texture des zones homogènes (basses fréquences) : dans l'image de gauche on voit apparaître les textures de l'image « École des Mines » et dans l'image de droite celles de « Lena ».

<sup>1</sup>cf A. V. Openheim and J. S. Lim, *The importance of phase in signals*. Proceedings of the IEEE Vol. 69 Nr 5, 529-541, 1981.



# Chapitre 3

## Introduction à la restauration des images

Dans ce chapitre sont présentées quelques méthodes pour restaurer des images dégradées. Ceci nécessite de modéliser au préalable la dégradation, c'est-à-dire le passage de l'image idéale  $u_0$  (qui est inconnue et que l'on aimerait approcher par l'algorithme de restauration) à l'image perçue  $u$ .

### 3.1 Modèles linéaires de dégradation des images

Nous considérons un modèle de dégradation linéaire : l'image dégradée  $u$  est obtenue à partir de  $u_0$  par convolution avec un noyau  $g$ . On suppose de plus que les images sont entachées de bruit additif Gaussien, ce qui amène au modèle suivant :

$$u = g * u_0 + n$$

où l'image  $n$  est telle que ses pixels  $n(x, y)$  sont indépendants et identiquement distribués selon une loi normale de moyenne nulle et d'écart-type  $\sigma$ . On parle de bruit blanc Gaussien.

Ceci signifie que l'expression de la valeur en niveaux de gris de l'image  $u$  au pixel  $(x, y)$  est :

$$u(x, y) = \sum_{(\tilde{x}, \tilde{y})} u_0(\tilde{x}, \tilde{y}) g(x - \tilde{x}, y - \tilde{y}) + n(x, y) \quad (3.1)$$

$$= \sum_{(\tilde{x}, \tilde{y})} g(\tilde{x}, \tilde{y}) u_0(x - \tilde{x}, y - \tilde{y}) + n(x, y). \quad (3.2)$$

En pratique, les sommes précédentes sont finies car le support du noyau  $g$  est fini (en tout cas sa représentation numérique l'est).

Voici quelques exemples de noyaux de convolution :

- Flou de mise au point :

$$g_r(x, y) = \frac{1}{\pi r^2} \mathbf{1}_{D_r}(x, y)$$

avec  $\mathbf{1}_{D_r}$  l'indicatrice du disque  $D(0, r)$ .

- Flou Gaussien (flou dû aux perturbations atmosphériques par exemple) :

$$g_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right).$$

- Flou de bougé selon un mouvement rectiligne (l'axe horizontal ici) :

$$g_l(x, y) = \frac{1}{2l} (Y(x+l) - Y(x-l)) \delta(y)$$

où  $\delta(y) = 1$  si  $y = 0$  et 0 sinon.

La figure 3.1 montre deux exemples de noyaux.

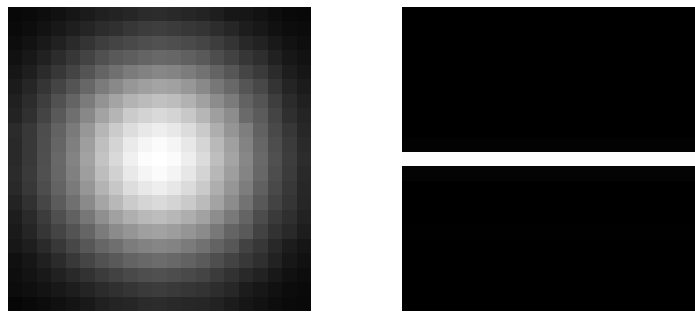


FIG. 3.1 – Noyaux Gaussien (à gauche) et de bougé selon un axe horizontal (à droite). Les deux noyaux ont une taille de 21 pixels. Le noir correspond à la valeur 0.

La modélisation du flou de mise au point proposée ici est très naïve. En effet, sur une photographie réelle où seule une partie de la scène est nette (dans le plan de netteté), le rayon du flou de mise au point dépend de la distance à ce plan de netteté. Faire l'hypothèse que le rayon est constant pour toute une image n'est valide que s'il s'agit de l'image d'un plan parallèle au plan de netteté.

La figure 3.2 montre l'effet de la convolution par des noyaux. Examinons de plus près l'image du point en haut à gauche de l'image *synthese*. Il semble qu'il s'agisse d'un segment horizontal, qui aurait une longueur de 21 pixels. En fait, reprenons l'équation (3.1) définissant la convolution dans le cas où l'image  $u_0$  est un point blanc ( $u_0(\tilde{x}, \tilde{y}) = 1$ ) sur fond noir ( $u_0(x, y) = 0$  ailleurs). Alors (ici le bruit  $n$  est négligé) :

$$u(x, y) = g(x - \tilde{x}, y - \tilde{y}).$$





FIG. 3.2 – Images originales (de taille  $512 \times 512$  en haut : lena à gauche et synthese à droite) et images dégradées (en bas) : l’image de gauche a subi un flou Gaussien et celle de droite un flou de bougé selon l’axe horizontal.

Autrement dit, l’image d’un point est le noyau  $g$ , centré en ce point.

Pour cette raison, le noyau de convolution  $g$  est appelé *Point Spread Function* (PSF, littéralement *fonction d’étalement d’un point*). Ainsi, si dans l’image originale figure des formes ponctuelles très contrastées (par exemple des lampadaires au loin dans une photographie nocturne), le noyau sera « lisible » sur l’image dégradée autour de ces formes ponctuelles. On pourra ainsi déterminer sa forme (pour un flou de bougé par exemple) ou ses dimensions caractéristiques ( $\sigma$  pour le flou Gaussien,  $r$  pour le flou de mise au point, etc.).

La figure 3.3 présente les images « floutées » et bruitées, selon le modèle de dégradation  $u = g * u_0 + n$ .

Le problème de la *déconvolution* consiste à retrouver l’image idéale  $u_0$  non dégradée (ou au moins une approximation) à partir de l’image observée  $u$ .

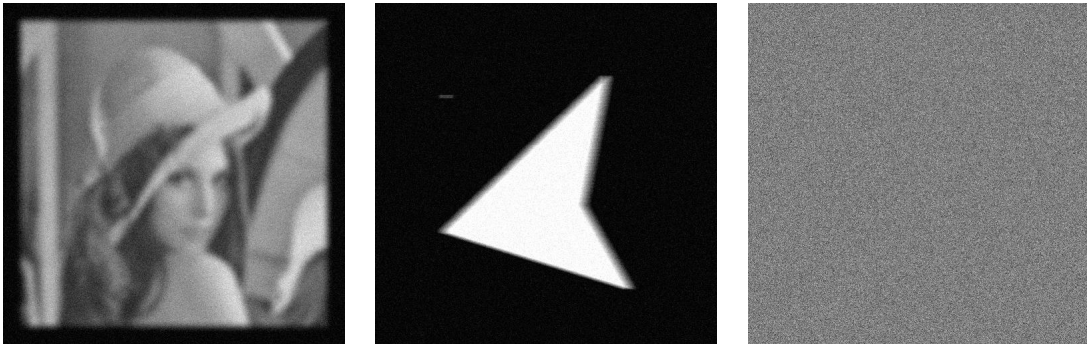


FIG. 3.3 – Images dégradées et bruitées, et image de bruit blanc (à droite). L'écart-type du bruit appliqué est ici de 10 pour des raisons de visualisation.

## 3.2 Déconvolution directe

Dans cette section, nous noterons  $U, U_0, G, N$  les transformées de Fourier respectives de  $u, u_0, g, n$ .

On suppose dans un premier temps que le bruit  $n$  est négligeable ; le modèle de dégradation est donc  $u = g * u_0$ . En utilisant la transformée de Fourier de cette relation, on trouve d'après la propriété 12 page 31 :  $U = G \cdot U_0$ .

Donc si  $G$  ne s'annule pas :

$$\forall(u, v), U_0(u, v) = \frac{U(u, v)}{G(u, v)}.$$

Si  $G$  s'annule, on peut par exemple changer  $G(u, v)$  en  $C$  (petit) si  $|G(u, v)| < C$ . La valeur de  $C$  devient un paramètre de l'algorithme de restauration.

Finalement, on peut récupérer  $u_0$  par transformée de Fourier inverse du rapport  $U/G$ .

La figure 3.2 montre le résultat de cette déconvolution directe, dans le cas sans bruit, et dans le cas où  $n$  est un bruit blanc Gaussien de variance 1.

Comme on s'y attend, la reconstruction dans le « cas non bruité » est parfaite (modulo la quantification des niveaux de gris). Par contre la déconvolution directe ne donne pas de résultat probant dans le « cas bruité », même avec une valeur de la variance du bruit aussi petite que 1. Pour quelle raison ?

La déconvolution directe consiste dans ce cas à opérer une transformée de Fourier inverse sur :

$$U_0(u, v) = \frac{U(u, v)}{G(u, v)} + \frac{N(u, v)}{G(u, v)}.$$

Or, comme on le verra plus loin, la transformée de Fourier du bruit blanc Gaussien  $N$  est un bruit blanc Gaussien. Donc en moyenne toutes les fréquences ont la même contri-

bution<sup>1</sup>. Ce n'est pas le cas pour les transformées de Fourier  $U$  et  $G$  des signaux réguliers  $u$  et  $g$ , dont le module décroît dans les hautes fréquences (cf figure 3.2). Donc le terme  $N(u, v)/G(u, v)$  amplifie la contribution du bruit dans les hautes fréquences. Pour cette raison, la reconstruction directe n'est plus possible.

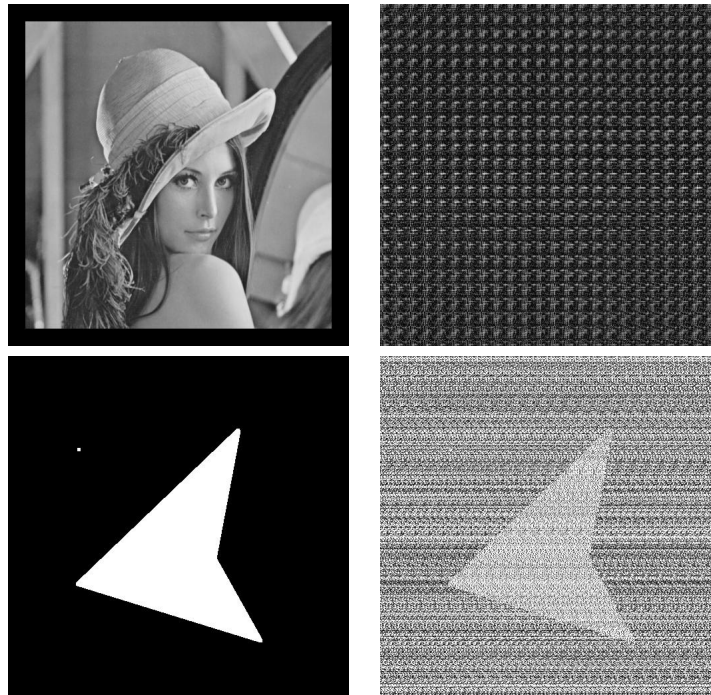


FIG. 3.4 – Reconstruction par déconvolution directe lorsqu'aucun bruit n'a été appliqué (à gauche), et reconstruction par déconvolution directe lorsqu'un bruit blanc Gaussien de variance 1 a été appliqué (à droite). Les images de gauche sont restaurées de très bonne manière, alors que ce n'est pas le cas pour les images de droite.

### 3.3 Restauration par filtre de Wiener

On va construire un filtre de restauration assez robuste au bruit, appelé *filtre de Wiener*. Dans un sens que nous allons préciser, il s'agit d'un filtre *optimal*.

Le modèle considéré de formation d'image est toujours  $u = g * u_0 + n$ , et on cherche un filtre  $h$  tel que  $v = h * u$  soit aussi proche que possible de  $u_0$ . L'image  $u_0$  est bien sûr inconnue, comme  $g$  et  $n$ . On dispose uniquement de l'image dégradée  $u$ .

<sup>1</sup>On parle de bruit « blanc » pour cette raison : toutes les fréquences du spectre sont présentes dans la lumière blanche.



FIG. 3.5 – Module des transformées de Fourier des deux noyaux de la figure 3.1. La transformée de Fourier d'une Gaussienne est une Gaussienne et la transformée de Fourier du segment correspondant au flou de bougé horizontal est un sinus cardinal sur chaque ligne.

On note  $U, U_0, V, G, H, N$  les transformées de Fourier respectives de  $u, u_0, v, g, h, n$ . Remarquons qu'il est équivalent de déterminer la réponse impulsionnelle  $h$  ou la fonction de transfert  $H$ .

L'image  $u = g * u_0 + n$  (donc également  $v$ ) peut en fait être vue comme réalisation d'une variable aléatoire, car  $n$  est une variable aléatoire. On cherche donc  $h$  qui minimise l'erreur quadratique moyenne :

$$\mathbb{E} \left( \sum_{x,y} |v(x,y) - u_0(x,y)|^2 \right).$$

D'après l'égalité de Parseval, et la transformation des convolutions en produits par la transformée de Fourier, minimiser  $\sum_{x,y} |v(x,y) - u_0(x,y)|^2$  équivaut à minimiser :

$$\sum_{a,b} |H(a,b)G(a,b)U_0(a,b) + H(a,b)N(a,b) - U_0(a,b)|^2,$$

ou aussi, si  $z^*$  désigne le conjugué du complexe  $z$  :

$$\begin{aligned} \sum_{a,b} & |H(a,b)G(a,b) - 1|^2 |U_0(a,b)|^2 + |H(a,b)|^2 |N(a,b)|^2 \\ & + (H(a,b)G(a,b) - 1)U_0(a,b)H^*(a,b)N^*(a,b) \\ & + (H^*(a,b)G^*(a,b) - 1)U_0^*(a,b)H(a,b)N(a,b). \end{aligned}$$

Or, l'espérance de  $N(a,b)$  est 0 car  $N(a,b)$  est combinaison linéaire des  $n(x,y)$  (par définition de la Transformée de Fourier Discrète) qui sont indépendants, identiquement distribués selon une loi normale de moyenne nulle (et d'écart-type  $\sigma$ ).

D'autre part,  $\mathbb{E}(|N(a, b)|^2)$  est la variance de  $N(a, b)$ , qui vaut  $\sigma^2$  d'après la remarque précédente et le fait que la somme des carrés des poids intervenant dans la combinaison linéaire vaut 1.

Ainsi, minimiser  $\mathbb{E} \left( \sum_{x,y} |v(x, y) - u_0(x, y)|^2 \right)$  revient en fait à minimiser :

$$\sum_{a,b} |H(a, b)G(a, b) - 1|^2 |U_0(a, b)|^2 + \sigma^2 |H(a, b)|^2.$$

En dérivant pour tout  $(a, b)$  par rapport au complexe inconnu  $H(a, b)$  :

$$\forall(a, b), G(a, b) (H(a, b)G(a, b) - 1)^* |U_0(a, b)|^2 + \sigma^2 H^*(a, b) = 0.$$

On déduit :

$$H(a, b) = \frac{G^*(a, b)}{|G(a, b)|^2 + \frac{\sigma^2}{|U_0(a, b)|^2}}.$$

Dans le cas où le bruit est négligeable,  $\sigma = 0$  et on retrouve :  $H(a, b) = 1/G(a, b)$ . Il s'agit tout simplement de la déconvolution directe.

Bien sûr, en pratique on ne connaît ni  $\sigma$  ni  $|U_0(a, b)|^2$ . En première approximation, on simplifiera le filtre en :

$$H(a, b) = \frac{G^*(a, b)}{|G(a, b)|^2 + K},$$

où  $K$  est un paramètre augmentant avec la variance du bruit.

La figure 3.3 montre une application de la restauration par filtre de Wiener.

*Remarque 1* : par construction, le filtre obtenu, appelé *filtre de Wiener*, est optimum pour la reconstruction  $L^2$  et l'hypothèse de bruit blanc Gaussien additif.

*Remarque 2* : dans la simplification du filtre, on peut aussi modifier le paramètre  $K$  en une fonction  $K(a, b)$  croissante dans les hautes fréquences. En effet  $K(a, b) = \sigma^2/|U_0(a, b)|^2$  et les coefficients de Fourier  $U_0(a, b)$  décroissent dans les hautes fréquences.

*Remarque 3* : ceci est une des présentations possibles du filtre de Wiener. Dans le cas le plus général, on n'est pas obligé de supposer que le bruit est i.i.d. dans l'image. On peut faire apparaître l'autocorrélation du bruit, la démonstration est similaire. L'estimation de cette autocorrélation reste un problème à part entière.

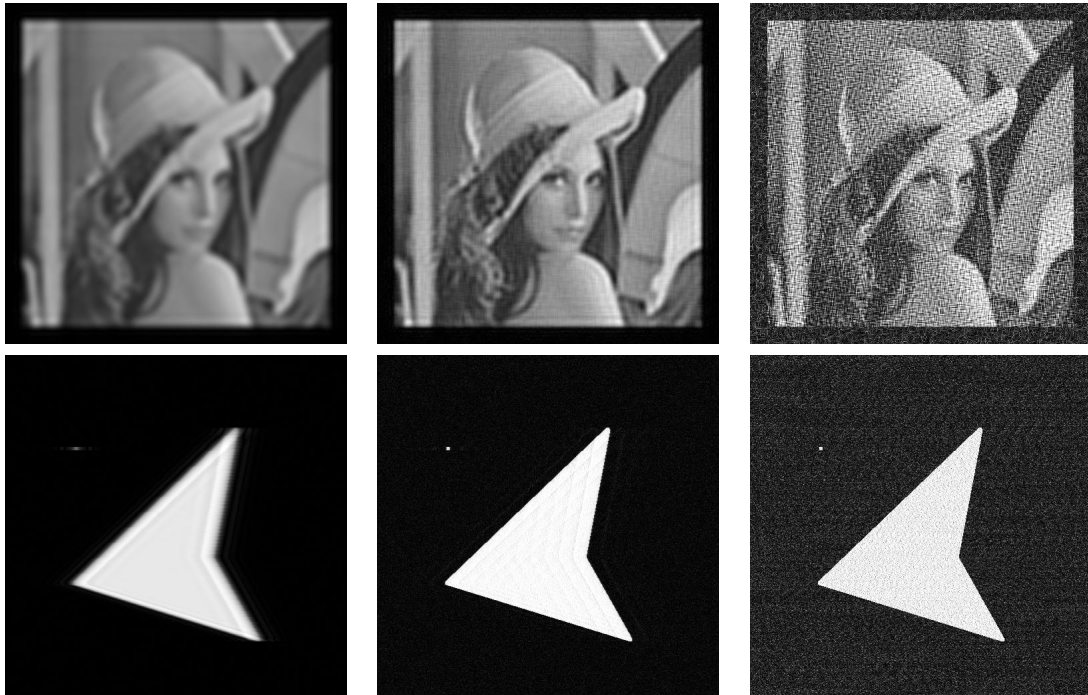


FIG. 3.6 – Reconstruction par filtre de Wiener, avec différentes valeurs du paramètre  $K$ . Colonne 1 :  $K = 10^{-1}$ . Colonne 2 :  $K = 10^{-3}$ . Colonne 3 :  $K = 10^{-5}$ . On voit que des grandes valeurs de  $K$  permettent de se débarrasser du bruit, au détriment d’une image plus floue. En effet les grandes valeurs de  $K$  atténuent les hautes fréquences de l’image, donc la rapidité des transitions au niveau des contours. Une valeur de  $K$  trop faible donne des contours francs mais le bruit n’est pas éliminé.

### 3.4 Restauration par l’algorithme de Richardson-Lucy

Cette fois le modèle de dégradation de l’image originale inconnue  $u_0$  est tel que la valeur du niveau de gris  $u(x, y)$  au pixel  $(x, y)$  est la réalisation d’une loi de Poisson de paramètre  $\lambda(x, y) = (g * u_0)(x, y)$ .

*Remarque 1* : rappelons que  $\lambda(x, y)$  est aussi l’espérance (la valeur moyenne) de  $u(x, y)$ , et que :

$$\Pr(u(x, y) = u) = \frac{\lambda(x, y)^u e^{-\lambda(x, y)}}{u!}.$$

*Remarque 2* : ce modèle est justifié physiquement (en particulier en imagerie astronomique) ; le niveau de gris est proportionnel au nombre de photons arrivant en un pixel donné, celui-ci étant distribué selon une loi de Poisson comme dans les phénomènes d’attente sans mémoire.

*Remarque 3* : le modèle linéaire de dégradation dans la section précédente rentre dans le même cadre statistique en changeant la loi de Poisson par la loi normale. En effet, il revient à considérer que  $u(x, y)$  est la réalisation d'une loi normale de moyenne  $\mu(x, y) = (g * u_0)(x, y)$  et de variance  $\sigma^2$ .

Supposons que les valeurs des niveaux de gris des pixels de l'image  $u$  soient indépendantes et distribuées selon les lois de Poisson indiquées ci-dessus, alors la *vraisemblance* de l'observation  $u$  est le produit des probabilités d'observation des niveaux de gris aux pixels  $(x, y)$  :

$$\mathcal{L}(u|u_0) = \prod_{(x,y)} \frac{\lambda(x, y)^{u(x,y)} e^{-\lambda(x,y)}}{u(x, y)!}.$$

*Rappel* :  $\lambda(x, y) = g * u_0(x, y)$  est inconnu.

La technique du maximum de vraisemblance consiste à chercher l'image idéale  $u_0$  (connaissant l'image dégradée  $u$ ) tel que  $\mathcal{L}(u|u_0)$  soit maximal : on cherche l'image  $u_0$  qui maximise la probabilité d'apparition de l'image  $u$ .

Il est équivalent de maximiser (par rapport à  $u_0$ )  $\log(\mathcal{L}(u|u_0))$ , soit :

$$\sum_{(x,y)} u(x, y) \log(\lambda(x, y)) - \lambda(x, y).$$

Au maximum, les dérivées partielles de l'expression précédente par rapport à  $u_0(\tilde{x}, \tilde{y})$  sont nulles, donc après calcul :

$$\forall(\tilde{x}, \tilde{y}), \sum_{(x,y)} \left( \frac{u(x, y)g(x - \tilde{x}, y - \tilde{y})}{(g * u_0)(x, y)} - g(x - \tilde{x}, y - \tilde{y}) \right) = 0.$$

Par conséquent :

$$\forall(\tilde{x}, \tilde{y}), \sum_{(x,y)} \frac{u(x, y)g(x - \tilde{x}, y - \tilde{y})}{(g * u_0)(x, y)} = G_0,$$

où  $G_0 = \sum_{(x,y)} g(x - \tilde{x}, y - \tilde{y}) = \sum_{(x,y)} g(x, y)$  par périodicité. Remarquons qu'en général  $G_0 = 1$ .

Ceci permet d'écrire, après multiplication par  $u_0(\tilde{x}, \tilde{y})$  :

$$\forall(\tilde{x}, \tilde{y}), \frac{u_0(\tilde{x}, \tilde{y})}{G_0} \sum_{(x,y)} \frac{u(x, y)g(x - \tilde{x}, y - \tilde{y})}{(g * u_0)(x, y)} = u_0(\tilde{x}, \tilde{y}).$$

L'image idéale  $u_0$  apparaît alors comme la solution d'un problème de point fixe, que l'on résout par itérations successives de la manière suivante.

1.  $u_0^0$  est par exemple une image constante, ou l'image  $u$  ;
2. si  $k \in \mathbb{N}$ , pour tout pixel  $(\tilde{x}, \tilde{y})$

$$u_0^{k+1}(\tilde{x}, \tilde{y}) = \frac{u_0^k(\tilde{x}, \tilde{y})}{G_0} \sum_{(x,y)} \frac{u(x, y)g(x - \tilde{x}, y - \tilde{y})}{(g * u_0^k)(x, y)}. \quad (3.3)$$

*Remarque :* Pour implémenter la formule d'itération (3.3), on commence par évaluer la convolution  $g * u_0^k$ , puis on calcule le quotient (ponctuel)  $u/(g * u_0^k)$ , que l'on convole avec le symétrique de  $g$  par rapport à l'origine. On finit par multiplier (toujours ponctuellement) le tout par  $u_0^k/G_0$ .

On n'est pas assuré de la convergence de cette méthode (rien ne dit que la fonctionnelle est contractante), mais généralement la suite  $(u_0^k)_{k \in \mathbb{N}}$  converge. Néanmoins, la convergence se fait vers *un* maximum (local) de la vraisemblance.

Cet algorithme est celui de *Richardson-Lucy* (1972-1974). Il est optimum au sens du maximum de vraisemblance pour le modèle considéré (distribution de Poisson pour les niveaux de gris).

La figure 3.4 montre un exemple d'utilisation de cet algorithme.





FIG. 3.7 – Algorithme de Richardson-Lucy. En haut, image originale. On choisit un filtre Gaussien de taille 21. En bas : après 10 itérations (à gauche), et après 100 itérations (à droite). La sensation de netteté s'améliore, des détails deviennent visibles.



# Chapitre 4

## Compression numérique sans perte

Informatiquement, un signal numérique est une suite de bits représentant la succession des valeurs prises à certains instants. Chaque valeur (on parle d'*échantillon*) correspond à (par exemple) deux octets. Un signal numérique de  $N$  échantillons est donc représenté par une suite de  $N \times 2 \times 8$  bits.

Le problème traité dans ce chapitre est le suivant : étant donné un signal numérique, est-il possible d'en trouver une représentation (un *codage*) utilisant moins de bits mais sans perdre d'information ? « Sans perdre d'information » signifie que l'on est capable de reconstruire *exactement* le signal initial à partir de cette représentation. On parle de *compression sans perte*.

Prenons l'exemple du code Morse. Chaque caractère est codé par une succession de points et traits. Mais ce codage est à longueur variable, adaptée à la fréquence d'apparition des lettres : la lettre **E** est codée par : ·, alors que la lettre **Z** est codée par : — — ··. Le but est de transmettre « en moyenne » un moins grand nombre de symboles élémentaires (trait ou point) qu'avec un codage où toutes les lettres seraient représentées par le même nombre de symboles.

Dans ce chapitre nous formalisons ces notions, dans le cadre de la théorie statistique de l'information de Claude Shannon (1916-2001) développée essentiellement dans les années 1940-1950.

### 4.1 Hypothèse et notations

Dans la suite on considère un alphabet fini de  $K$  symboles  $\mathcal{A} = (x_1, x_2, \dots, x_K)$ , tel que chaque symbole est codé par un mot binaire  $m_k$  de taille  $l_k$ . La liste  $(m_1, m_2, \dots, m_K)$  est appelée *code*, ou *codage* de l'alphabet  $\mathcal{A}$ .

On suppose qu'un signal numérique de taille  $N$  est constitué de  $N$  réalisations d'une variable aléatoire  $X$  (appelée *source*) prenant ses valeurs sur  $\mathcal{A}$ . La distribution de  $X$  est

supposée connue. La longueur moyenne de la représentation binaire du signal est alors :

$$N\mathbb{E}(l(X)) = N \sum_{k=1}^K l_k \Pr(X = x_k). \quad (4.1)$$

Le problème traité dans cette section est celui de trouver un codage des symboles qui minimise la longueur moyenne de la représentation des signaux, i.e. qui minimise  $\sum_{k=1}^L l_k \Pr(X = x_k)$ .

*Remarque :* les codages considérés seront binaires, mais on peut généraliser sans grosses difficultés à des codages en une base plus grande que 2.

## 4.2 Codes préfixes

Les codages (i.e. les concaténations de mots binaires) qui peuvent être décodés sans ambiguïté sont les *codes préfixes* (aussi appelés pour cette raison *codes instantanés*).

**Définition 19** On appelle *code préfixe* tout code tel que chaque mot n'est le début d'aucun autre.

Un code préfixe se représente sous la forme d'un arbre binaire dont les feuilles sont les symboles  $x_k$ , et où le chemin entre la racine et la feuille  $x_k$  fournit le mot binaire  $m_k$  de la manière suivante : une bifurcation à gauche (resp. droite) correspond à "0" (resp. "1"). Cf figure 4.1.

Réciproquement, le code fournit par un arbre binaire et cet algorithme est nécessairement un code préfixe (sinon on aboutirait à une contradiction sur le fait que les symboles sont codés aux feuilles).

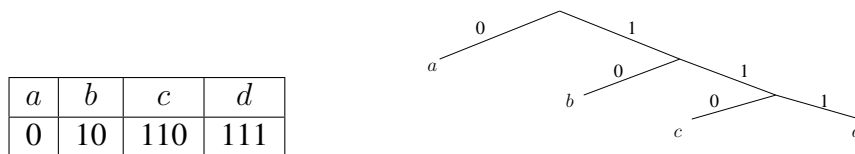


FIG. 4.1 – À gauche : le code préfixe binaire d'un alphabet à 4 symboles. À droite, l'arbre binaire associé.

On aura besoin dans la suite de la proposition suivante.

**Proposition 14 (inégalité de Kraft)** Si  $(w_1, w_2, \dots, w_K)$  est un code préfixe (binaire), alors il vérifie :

$$\sum_{k=1}^K 2^{-l_k} \leq 1. \quad (4.2)$$

Réciproquement, si  $l_1, l_2, \dots, l_K$  sont des entiers vérifiant l'inégalité (4.2), alors il existe un code préfixe formé de  $K$  mots  $w_k$  de longueur  $l_k$ .

*Démonstration.* Considérons l'arbre associé au code préfixe  $(w_1, w_2, \dots, w_K)$ , et "complétons"-le en un arbre complet  $\mathcal{T}'$  (tel que les feuilles soient toutes à la profondeur  $L = \max(l_1, \dots, l_K)$ ). Pour tout  $k \in \{1, \dots, K\}$ , soit  $A_k$  le sous-arbre des descendants du nœud correspondant au mot  $w_k$  (faites un dessin !). Alors  $A_k \cap A_{k'} = \emptyset$  si  $k \neq k'$ , et le nombre de feuilles de  $A_k$  est  $2^{L-l_k}$ . Le nombre de feuilles de  $\cup_{k=1}^K A_k$  est donc  $\sum_{k=1}^K 2^{L-l_k}$ , qui est plus petit que le nombre total de feuilles de l'arbre  $\mathcal{T}'$  ( $2^L$ ). Ainsi,  $\sum_{k=1}^K 2^{L-l_k} \leq 2^L$ , d'où l'inégalité de Kraft.

Réciproquement, soit  $l_1, l_2, \dots, l_K$  des entiers vérifiant l'inégalité de Kraft. Supposons que  $l_1 \leq l_2 \leq \dots \leq l_K$ . On va construire l'arbre binaire associé à un code préfixe, en émondant un arbre binaire complet de profondeur  $l_K$ . On commence par considérer le sous-arbre « le plus à gauche » qui commence un nœud de niveau  $l_1$ . On supprime ce sous-arbre, en gardant le nœud à sa racine celui-ci devient une feuille, et va coder le symbole  $x_1$  par un mot binaire  $w_1$  de longueur  $l_1$ . On itère jusque  $l_K$ . Ceci est possible car  $\sum_{k=1}^K 2^{l_K-l_k} \leq 2^{l_K}$  d'après l'inégalité (4.2), donc on enlève moins de nœuds qu'il n'y en a dans l'arbre initial. L'arbre obtenu donne bien un code préfixe convenable.  $\square$

### 4.3 Théorie de l'information de Shannon

Dans son célèbre article<sup>1</sup> de 1948, Claude Shannon se demande comment mesurer (et définir) la quantité d'« information » produite par une source aléatoire. Citons-le :

Suppose we have a set of possible events whose probabilities of occurrence are  $p_1, p_2, \dots, p_n$ . These probabilities are known but that is all we know concerning which event will occur. Can we find a measure of how much "choice" is involved in the selection of the event or of how uncertain we are of the outcome ?

If there is such a measure, say  $H(p_1, p_2, \dots, p_n)$ , it is reasonable to require of it the following properties :

1.  $H$  should be continuous in the  $p_i$ .

<sup>1</sup>Claude E.Shannon, *A mathematical theory of communication*, Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, July and October, 1948

2. If all the  $p_i$  are equal,  $p_i = 1/n$ , then  $H$  should be a monotonic increasing function of  $n$ . With equally likely events there is more choice, or uncertainty, when there are more possible events.
3. If a choice be broken down into two successive choices, the original  $H$  should be the weighted sum of the individual values of  $H$ . The meaning of this is illustrated in Fig. 4.2.

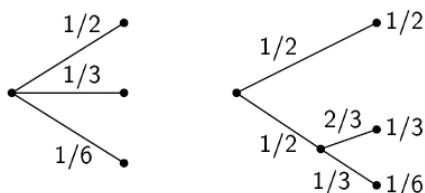


FIG. 4.2 – Decomposition of a choice from three possibilities.

At the left we have three possibilities  $p_1 = \frac{1}{2}$ ,  $p_2 = \frac{1}{3}$ ,  $p_3 = \frac{1}{6}$ . On the right we first choose between two possibilities each with probability  $\frac{1}{2}$ , and if the second occurs make another choice with probabilities  $\frac{2}{3}$ ,  $\frac{1}{3}$ . The final results have the same probabilities as before. We require, in this special case, that  $H(\frac{1}{2}, \frac{1}{3}, \frac{1}{6}) = H(\frac{1}{2}, \frac{1}{2}) + \frac{1}{2}H(\frac{2}{3}, \frac{1}{3})$ . The coefficient  $\frac{1}{2}$  is because this second choice only occurs half the time.

Shannon démontre que la seule fonction satisfaisant ces conditions (à une constante près) est donnée par la définition suivante.

**Définition 20** Soit  $X$  une variable aléatoire discrète, prenant des valeurs  $(x_1, x_2, \dots, x_K)$ , de loi  $\Pr(X = x_k) = p_k$ . On appelle **entropie** de la source  $X$  la quantité :

$$\mathcal{H}(X) = \mathbb{E}(-\log(X)) = - \sum_{k=1}^K p_k \log_2 p_k. \tag{4.3}$$

*Remarque :* dans la définition de l'entropie, on utilise la convention  $0 \cdot \log(0) = 0$ .

Nous aurons besoin pour la preuve de la proposition 16 du résultat suivant :

**Proposition 15 (inégalité de Jensen)** Soit  $f$  une fonction convexe sur un intervalle  $I$ ,  $x_1, x_2, \dots, x_K$   $K$  éléments de  $I$ , et  $\lambda_1, \lambda_2, \dots, \lambda_K$   $K$  réels de  $[0, 1]$  tels que  $\sum_{k=1}^K \lambda_k = 1$ , alors :

$$f\left(\sum_{k=1}^K \lambda_k x_k\right) \leq \sum_{k=1}^K \lambda_k f(x_k) \tag{4.4}$$

avec égalité (dans le cas où les  $\lambda_k$  sont  $> 0$ ) si et seulement si  $f$  est affine sur l'intervalle  $[\min_k \{x_k\}, \max_k \{x_k\}]$ .

**Proposition 16** Avec les notations de la définition précédente :

$$0 \leq \mathcal{H}(X) \leq \log_2(K). \quad (4.5)$$

*Démonstration.* Comme  $\forall k, p_k \leq 1$  on a bien sûr  $\mathcal{H}(X) \geq 0$ .

Ici, avec  $f = -\log$ ,  $\lambda_k = p_k$  et  $x_k = 1/p_k$ , l'inégalité de Jensen (proposition 15) se traduit par :

$$\mathcal{H}(X) = \sum_{k=1}^K p_k \log(1/p_k) \leq \log \left( \sum_{k=1}^K p_k/p_k \right) = \log(K).$$

□

**Proposition 17** L'entropie  $\mathcal{H}(X)$  est

- maximale ( $= \log_2(K)$ ) pour la distribution uniforme ;
- minimale ( $= 0$ ) pour la distribution chargeant une seule valeur.

*Démonstration.* Il suffit de vérifier que les bornes de la proposition 16 sont atteintes dans ces deux cas. □

On aura besoin dans la suite de la propriété suivante.

**Proposition 18 (inégalité de Gibbs)** Soient  $p_1, p_2, \dots, p_K$  et  $q_1, q_2, \dots, q_K$  deux distributions de probabilités discrètes. Alors :

$$-\sum_{k=1}^K p_k \log(p_k) \leq -\sum_{k=1}^K p_k \log(q_k) \quad (4.6)$$

avec égalité si et seulement si  $\forall k \in \{1, \dots, K\}, p_k = q_k$ .

*Démonstration.* D'après l'inégalité de Jensen (4.4) pour  $f = -\log$ ,  $\lambda_k = p_k$ , et  $x_k = p_k/q_k$  :

$$-\log \left( \sum_{k=1}^K p_k \frac{q_k}{p_k} \right) \leq -\sum_{k=1}^K p_k \log \left( \frac{p_k}{q_k} \right).$$

Le membre de droit est donc positif, d'où  $\sum_{k=1}^K p_k \log(p_k) - \sum_{k=1}^K p_k \log(q_k) \geq 0$  qui est l'inégalité de Gibbs.

Il y a égalité si et seulement si elle est réalisée dans l'inégalité (4.4), ce qui ne peut avoir lieu que si  $\min_k \{p_k/q_k\} = \max_k \{p_k/q_k\}$ , d'où  $\forall k, p_k = K q_k$ . Comme  $\sum_k p_k = \sum_k q_k = 1$ , les deux distributions sont donc égales. □

*Remarque* : l'inégalité de Gibbs montre que la quantité suivante :

$$D(p, q) = \sum_{k=1}^K p_k \log \left( \frac{p_k}{q_k} \right)$$

est positive, et nulle si et seulement si les deux distributions  $p$  et  $q$  sont identiques. On appelle  $D(p, q)$  la *divergence de Kullback-Leibler*, elle mesure une sorte de distance entre distributions de probabilité discrètes. Il ne s'agit pas d'une vraie distance car l'inégalité triangulaire n'est pas vérifiée.

**Théorème 11 (Shannon 1949)** La longueur moyenne  $\mathcal{L}(C) = \sum l_k p_k$  d'un code préfixe  $C$  associé à une source  $X$  vérifie :

$$\mathcal{L}(C) \geq \mathcal{H}(X). \quad (4.7)$$

De plus, il existe un code préfixe  $\tilde{C}$  associé à  $X$  tel que :

$$\mathcal{L}(\tilde{C}) < \mathcal{H}(X) + 1. \quad (4.8)$$

*Démonstration.* Considérons un code préfixe  $C = (w_1, \dots, w_K)$  associé à la source  $X$  tel que la longueur du mot binaire  $w_k$  soit  $l_k$ . Soit  $s_k = 2^{-l_k}$  et  $S = \sum_{k=1}^K s_k$ . La famille  $(s_k/S)_{k \in \{1 \dots K\}}$  définit alors une distribution de probabilité sur  $\{1 \dots K\}$ . D'après l'inégalité de Gibbs :

$$\mathcal{H}(X) \leq - \sum_{k=1}^K p_k \log \left( \frac{s_k}{S} \right)$$

Donc  $\mathcal{H}(X) \leq \mathcal{L}(C) + \sum_{k=1}^K p_k \log(S)$ . Or  $S = \sum_{k=1}^K 2^{-l_k} \leq 1$  d'après l'inégalité de Kraft (proposition 14). On en déduit l'inégalité 4.7.

Posons  $l_k = \lceil -\log(p_k) \rceil$  où  $\lceil x \rceil$  est le plus petit entier supérieur ou égal au réel  $x$ . Comme  $\sum_{k=1}^K 2^{-l_k} \leq \sum_{k=1}^K 2^{\log(p_k)} = 1$ , d'après la proposition 14 il existe un code préfixe  $\tilde{C}$  dont les  $K$  mots ont pour longueur  $(l_1, \dots, l_K)$ .

Ce code vérifie  $\mathcal{L}(\tilde{C}) = \sum_{k=1}^K l_k p_k < \sum_{k=1}^K (1 - \log(p_k)) p_k = 1 + \mathcal{H}(X)$ , qui est précisément l'inégalité (4.8).  $\square$

*Remarque* : on voit que l'élément limitant pour trouver un code préfixe  $\tilde{C}$  tel que  $\mathcal{L}(\tilde{C}) = \mathcal{H}(X)$  est que les  $-\log(p_k)$  ne sont a priori pas entiers.



## 4.4 Codage de Huffman

On donne une source  $X$  sur un alphabet de  $K$  symboles  $(x_1, \dots, x_K)$ , avec une distribution de probabilité associée  $(p_1, \dots, p_K)$ .

L'algorithme suivant fournit un code préfixe  $C_H$  qui réalise le minimum de la longueur moyenne  $\mathcal{L}(C)$  parmi tous les codes préfixes.

**Théorème 12 (code de Huffman - 1952)** *On considère l'arbre binaire pondéré construit récursivement selon l'algorithme suivant :*

- *initialisation : les symboles, pondérés par leur probabilité, sont les feuilles.*
- *itérer tant que le graphe courant n'est pas connexe : fusionner les deux arbres dont les racines sont de poids les plus petits en créant une nouvelle racine de poids égal à la somme de ces poids, liée à ces deux sous-arbres.*

*Le code associé à l'arbre ainsi créé (appelé « code de Huffman ») minimise  $\mathcal{L}(C) = \sum_k l_k p_k$  parmi les codes préfixes.*

*Démonstration.* Le code associé à l'arbre ainsi construit est bien préfixe car associé à un arbre binaire. Ces arbres seront pondérés de manière récursive par la somme des poids des fils, les feuilles étant pondérées par les probabilités  $p_k$  correspondants aux symboles  $x_k$ .

Remarquons tout d'abord que dans un arbre représentant la code optimal, deux nœuds 1 et 2 de poids respectifs  $w_1$  et  $w_2$  tels que  $w_1 < w_2$  sont à des profondeurs respectives  $l^1$  et  $l^2$  tels que  $l^2 \leq l^1$ . Raisonnons par l'absurde en supposant  $l^1 < l^2$ . Échangeons alors les sous-arbres de racine 1 et 2. La longueur moyenne des codes devient, en notant  $\mathcal{F}_1$  (resp.  $\mathcal{F}_2$ ) les feuilles du sous-arbre de racine 1 (resp. 2) :

$$\mathcal{L}(\tilde{C}) - \sum_{k \in \mathcal{F}_1} l_k p_k - \sum_{k \in \mathcal{F}_2} l_k p_k + \sum_{k \in \mathcal{F}_1} (l_k + (l^2 - l^1)) p_k + \sum_{k \in \mathcal{F}_2} (l_k - (l^2 - l^1)) p_k.$$

D'où :

$$\mathcal{L}(\tilde{C}) + \sum_{k \in \mathcal{F}_1} (l^2 - l^1) p_k - \sum_{k \in \mathcal{F}_2} (l^2 - l^1) p_k = \mathcal{L}(\tilde{C}) - (l^2 - l^1) \left( \sum_{k \in \mathcal{F}_2} p_k - \sum_{k \in \mathcal{F}_1} p_k \right).$$

Or  $\sum_{k \in \mathcal{F}_2} p_k - \sum_{k \in \mathcal{F}_1} p_k = w_2 - w_1$  par définition des pondérations. Donc le nouvel arbre fournirait un code de longueur moyenne strictement plus petite que  $\mathcal{L}(\tilde{C})$ , en contradiction avec l'optimalité de l'arbre initial.

En conséquence de cette remarque, on obtient que si les poids des nœuds sont tels que  $w_1 \leq w_2 \leq \dots \leq w_n$ , alors les profondeurs associées dans l'arbre optimal vérifient  $l^1 \geq l^2 \geq \dots \geq l_n$ .

Nous allons démontrer le théorème par récurrence sur le nombre de symboles de l'alphabet. Le théorème est bien sûr vrai pour un alphabet à un symbole. Considérons un alphabet à  $n + 1$  symboles de poids  $w_1 \leq w_2 \leq \dots \leq w_{n+1}$ . Dans l'arbre optimal  $T_{n+1}$ ,  $w_1$  et  $w_2$  sont frères (car  $w_1$  est le nœud à la plus grande profondeur d'après ce qui précède, et qu'il a nécessairement un frère sinon on pourrait former un arbre de longueur moyenne plus faible en le remplaçant

par son père). Soit  $T_n$  l'arbre obtenu à partir de  $T_{n+1}$  en groupant les nœuds 1 et 2. Soit  $H_n$  l'arbre obtenu par l'algorithme de Huffman sur l'alphabet obtenu en groupant les symboles 1 et 2. Soit  $l$  la profondeur dans  $H_n$  du nœud correspondant à ce regroupement. Par construction  $H_{n+1}$  (l'arbre de Huffman pour l'alphabet complet) est obtenu en créant les deux feuilles 1 et 2 à partir de la feuille de  $H_n$  regroupant 1 et 2. Donc  $\mathcal{L}(H_{n+1}) = \mathcal{L}(H_n) - l(w_1 + w_2) + (l+1)(w_1 + w_2) = \mathcal{L}(H_n) + w_1 + w_2$ . De même,  $\mathcal{L}(T_{n+1}) = \mathcal{L}(T_n) + w_1 + w_2$ . Mais  $\mathcal{L}(H_n) \leq \mathcal{L}(T_n)$  (d'après l'hypothèse de récurrence), donc  $\mathcal{L}(H_{n+1}) \leq \mathcal{L}(T_{n+1})$ . Or  $T_{n+1}$  est optimal, donc  $\mathcal{L}(H_{n+1}) = \mathcal{L}(T_{n+1})$ , et donc  $H_{n+1}$  fournit un code optimal. Ceci achève la récurrence.  $\square$

*Remarque :* le code de Huffman vérifie donc :  $\mathcal{H}(X) \leq \mathcal{L}(C_H) < \mathcal{H}(X) + 1$ .

*Remarque :* il n'y a pas unicité du code minimal : il peut y avoir des choix de fusion dans l'algorithme de Huffman, et le choix d'affectation des branches « gauche » et « droite » n'est pas spécifié.

## 4.5 Exemple

Considérons l'alphabet suivant à huit lettres, avec la distribution de probabilités associée :

a	b	c	d	e	f	g	h
0.05	0.05	0.05	0.1	0.15	0.15	0.2	0.25

L'application de l'algorithme de Huffman (théorème 12) donne l'arbre binaire de la figure 4.3 (à vérifier en exercice).

Le codage canonique des feuilles des arbres binaires (0 pour une « bifurcation » à gauche, 1 sinon) donne le code préfixe suivant pour les huit lettres :

a	b	c	d	e	f	g	h
00000	00001	0001	100	101	001	11	01

On vérifie facilement qu'il s'agit bien d'un code préfixe : aucun code n'est le préfixe d'un autre. Ceci rend le décodage très facile en parcourant l'arbre : le décodage d'un mot consiste à parcourir l'arbre en choisissant les branches de gauche ou de droite selon la valeur 0 ou 1 lue ; lorsqu'on arrive à une feuille on écrit la lettre correspondante et on continue la lecture après être revenu à la racine.

Par exemple, 1000010001010100000100 se décode en dfchhad.

*Remarque :* l'inconvénient majeur du codage de Huffman tel qu'il a été présenté est sa faible résistance aux erreurs de transmissions. Si un 0 ou 1 a été mal transmis (changé en 1 ou 0, effacé, doublonné. . .), le décodage de l'ensemble du mot est perturbé.

Par exemple, 0000010001010100000100 se décode en adhhad.

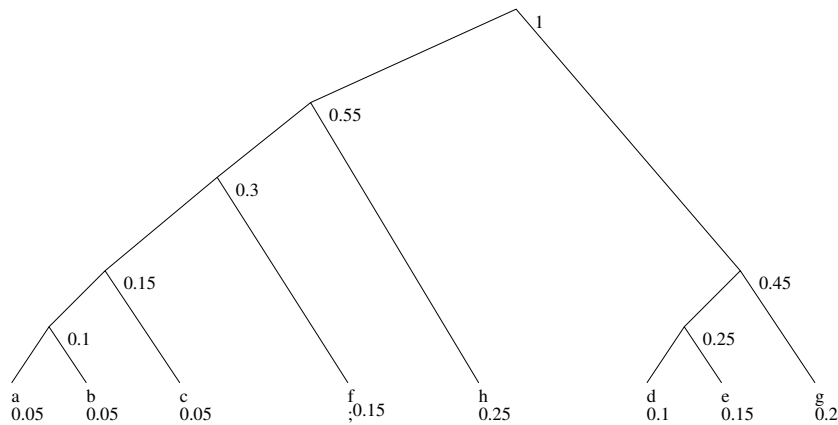


FIG. 4.3 – Arbre créé par l’algorithme de Huffman. Il n’y a bien sûr pas unicité, d’autres choix auraient été possibles (à vérifier en exercice).

## 4.6 Autres algorithmes de compression

Remarquons que l’entropie donne une borne inférieure pour la longueur moyenne du codage d’une lettre dans un code **préfixe**. Donc d’autres algorithmes peuvent aussi être intéressants en pratique. Nous n’en donnons qu’un bref aperçu dans la suite de cette section. En particulier, nous n’abordons pas les problèmes soulevés par l’implantation.

### 4.6.1 Run-length encoding (RLE)

Le codage *Run-length encoding* (RLE), ou *codage par plages*, est utilisé lorsque l’on cherche à encoder un mot présentant des répétitions **successives**. Le mot est transformé en remplaçant une suite de lettres identiques par cette lettre répétée précédée du nombre d’occurrences.

Par exemple : AAAAGBBAYYYY est codé en : 4A1G2B1A4Y. Le gain est ici de deux caractères. Bien sûr, dans le pire des cas (lettres successives toutes différentes) le code RLE double le nombre de caractères, le gain n’est pas assuré.

Une variante de RLE intervient dans la transmission par fax (nombreux aplats uniformes) et dans la compression JPEG.

### 4.6.2 Codage arithmétique (1976)

Comme le codage de Huffman, le codage arithmétique appartient à la famille des codages entropiques, dans le sens où il se base sur les fréquences d’apparition des lettres pour coder avec peu de bits une lettre très fréquente. Dans le codage arithmétique, un message est codé par un décimal entre 0 et 1.

Pour ce faire, une partition de  $[0, 1]$  est construite à partir des probabilités d'apparition des lettres, de manière à ce que chaque lettre corresponde à un intervalle de longueur égale à sa probabilité.

Par exemple, si  $a$ ,  $b$ , et  $c$  ont des probabilités respectives de  $1/4$ ,  $1/5$ ,  $11/20$ , ces lettres correspondent aux intervalles  $[0, 0.25[$ ,  $[0.25, 0.45[$ ,  $[0.45, 1[$ .

Le codage se fait selon l'algorithme itératif suivant :

```
S=1
I=0
pour chaque lettre:
    L=S-I
    S=I + L * borne supérieur de la lettre courante
    I=I + L * borne inférieure de la lettre courante
fin
```

où  $S$  est la borne supérieure d'un intervalle,  $I$  la borne inférieure,  $L$  la longueur.

Le code associé au message est alors la dernière borne inférieure  $I$ .

Par exemple le codage du texte *bbac* est : 0.3045.

Pour le décodage d'un mot de  $n$  lettres correspondant au code  $x$  :

```
itérer tant que n>0
    retourner la lettre dont l'intervalle [I,S] contient x
    x = x - I
    x = x / (S - I)
    n = n - 1
fin
```

Pour le décodage du mot de quatre lettres associé à  $x = 0.3045$ , on retrouve bien *bbac* (exercice !).

Bien que le code de Huffman soit optimal au sens du nombre moyen de bits nécessaires pour coder **une** lettre, le code arithmétique est meilleur que le code de Huffman dans certains cas car il considère le message dans son ensemble et non comme une succession de lettres indépendantes.

Le codage arithmétique a fait l'objet de nombreux brevets de la part d'IBM, ce qui semble limiter son utilisation, en faveur du codage de Huffman.

### 4.6.3 Codage de Lempel-Ziv-Welch (1984)

Dans les codages entropiques du type de celui de Huffman, un dictionnaire est créé pour associer à chaque lettre de l'alphabet un mot binaire (de longueur variable), à l'aide des statistiques de fréquence des lettres dans le message à compresser.

Dans le codage de Lempel-Ziv-Welch, le dictionnaire associe à des *chaîne de lettres de longueur variable* figurant dans le message à compresser des mots binaires de longueur fixée. Ceci permet d'exploiter les corrélations entre les sorties de la source  $X$ .



# Chapitre 5

## Compression numérique avec perte

### 5.1 Décroissance des coefficients de Fourier

D'après la proposition 5 page 18, les modules des coefficients de Fourier  $|c_n(f)|$  sont décroissants lorsque  $|n| \rightarrow +\infty$ . La proposition suivante nous dit même que la décroissance est d'autant plus rapide que la fonction est régulière.

**Proposition 19** Si  $f \in C_p^k$ , alors  $c_n(f) = \mathcal{O}\left(\frac{1}{|n|^k}\right)$ .

*Démonstration.* Supposons  $f$  de période  $a$ . En intégrant par parties :

$$\begin{aligned}c_n(f) &= \frac{1}{a} \int_0^a f'(t) e^{2i\pi nt/a} dt \\ &= \left[ \frac{-1}{2i\pi n} f(t) e^{2i\pi nt/a} \right]_0^a + \frac{1}{2i\pi n} \int_0^a f'(t) e^{2i\pi nt/a} dt \\ &= \frac{a}{2i\pi n} c_n(f')\end{aligned}$$

car  $f(0) = f(a)$ .

Le résultat est obtenu par récurrence. □

Ce théorème donne l'idée de « compresser » les signaux en ne transmettant que les coefficients de Fourier suffisamment grands (puisqu'un certain nombre, correspondant à  $|n|$  grand, seront assez petits), par exemple en tronquant la série de Fourier. Voir par exemple la figure 5.1. D'après la proposition 19, ceci est d'autant plus « facile » que le signal est régulier<sup>1</sup>. Néanmoins, ceci est contrecarré par l'effet de Gibbs, qui dit que tronquer la série de Fourier introduit toujours des artefacts aux niveaux des discontinuités... Ceux-ci sont la cause des phénomènes de pré-écho dans les fichiers MP3, ou du phénomène de *ringing* (surlignage des bords contrastés) dans les fichiers JPEG.

---

<sup>1</sup>Bien sûr il s'agit d'un raisonnement heuristique : tous les signaux discrets peuvent être associés à un polynôme trigonométrique (chapitre 2), qui est  $C^\infty$ .

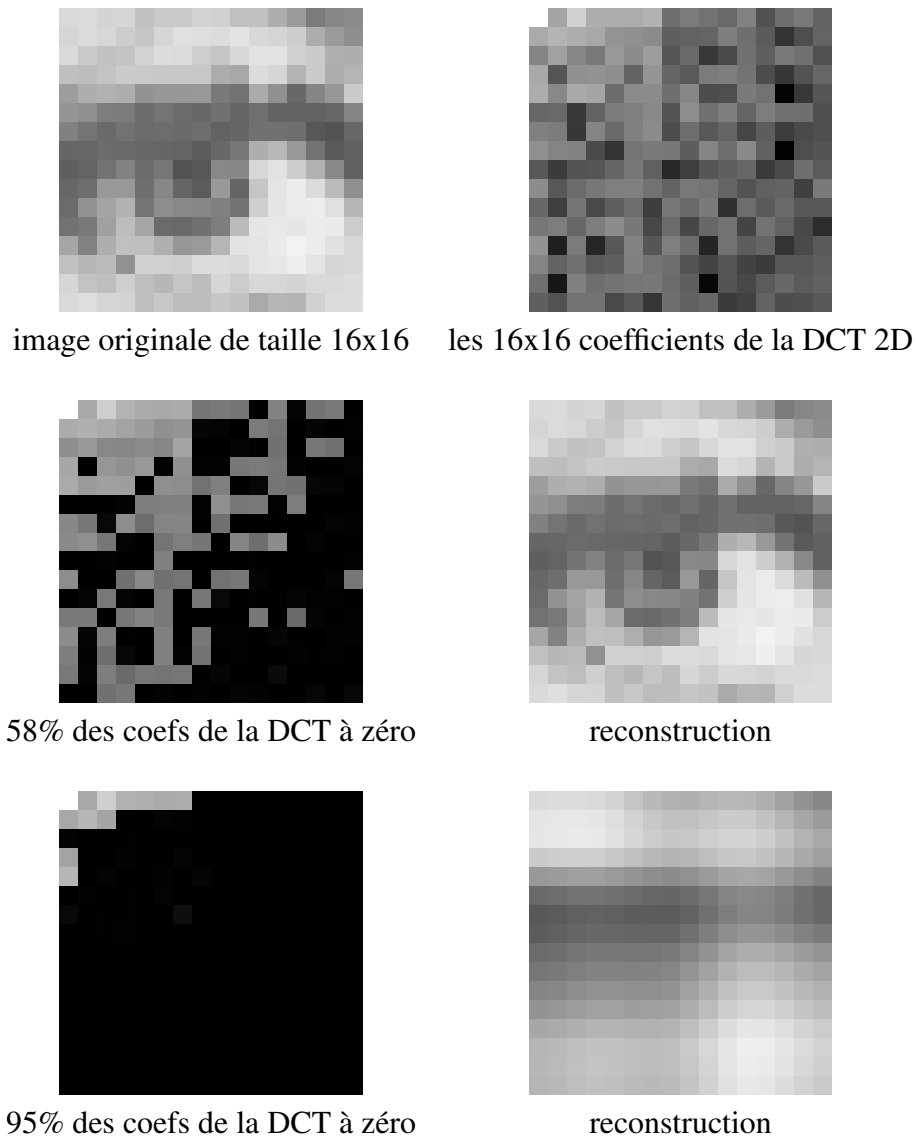


FIG. 5.1 – L'idée de la compression. En mettant à 0 les coefficients d'une transformée de Fourier du signal (ici, la DCT) en dessous d'un seuil donné, on a moins de coefficients à transmettre pour la reconstruction. Celle-ci sera nécessairement dégradée, voici pourquoi on parle de compression avec perte. Néanmoins, comme on le voit dans cet exemple, on peut se permettre d'éliminer plus de la moitié des coefficients sans perte perceptuelle notable (et bien plus en pratique). Ici, les basses (resp. hautes) fréquences sont en haut à gauche (resp. en bas à droite) des blocs 16x16. Plus le module d'un coefficient est petit, plus le carré le représentant est proche du noir. On voit bien dans l'image en haut à droite la décroissance des coefficients avec la fréquence.



## 5.2 Effet de Gibbs

Soit  $f$  la fonction  $2\pi$ -périodique valant :  $\begin{cases} -\pi/4 & \text{entre } -\pi \text{ et } 0 \\ \pi/4 & \text{entre } 0 \text{ et } \pi \end{cases}$

Sa série de Fourier est :  $\sum_{k=0}^{+\infty} \frac{\sin((2k+1)x)}{2k+1}$  (par calcul simple, laissé en exercice).

Le théorème 6 (Jordan-Dirichlet) nous donne les conditions de la convergence. La proposition suivante nous dit que la convergence n'est pas uniforme, et que l'approximation du créneau par un nombre fini de termes de la série de Fourier introduit toujours des artefacts d'amplitude constante, quelque soit le nombre de termes que l'on garde. Ceci est illustré sur la figure 5.2.

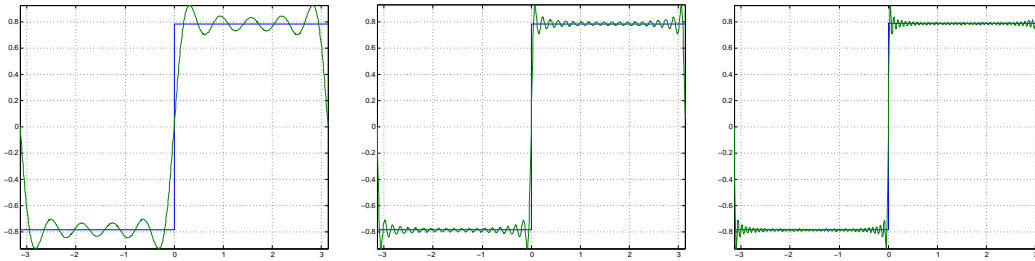


FIG. 5.2 – Effet de Gibbs. Quelque soit le nombre de coefficients gardés dans l'approximation du créneau par une série de Fourier (de gauche à droite :  $N = 5$ ,  $N = 20$ ,  $N = 40$ ), il restera toujours des oscillations résiduelles d'amplitude  $\simeq 0.14$ .

**Proposition 20** Soient  $N \in \mathbb{N}$ ,  $x \in \mathbb{R}$ , et :

$$f_N(x) = \sum_{k=0}^{E((N-1)/2)} \frac{\sin((2k+1)x)}{2k+1}$$

( $f_N$  est l'approximation par les  $N$  termes centraux de la transformée de Fourier de  $f$ ). On montre que  $f_N$  admet un extremum local en  $\pi/N$ , et

$$f_N(\pi/N) = \pi/4 + \pi/2 \cdot 0.0894 \dots$$

(à comparer à  $\pi/4$ , valeur de  $f$  en  $\pi/N$ , pour  $N > 4$ ).

*Démonstration.* La fonction  $f_N$  est dérivable sur  $\mathbb{R}$  et :

$$\forall x \in \mathbb{R}, f'_N(x) = \sum_{0 \leq 2k+1 \leq N} \cos((2k+1)x).$$

Un calcul simple (à faire en exercice : passer par exemple à l'exponentielle complexe) donne :

$$\forall x \in \mathbb{R}, f'_N(x) = \frac{\sin(Nx)}{2 \sin(x)}.$$

Les extrema de  $f_N$  sont donc les  $k\pi/N$ ,  $k \in \mathbb{Z}^*$ . Ces extrema sont bien visibles sur la figure 5.2, et leur nombre par période augmente effectivement avec  $N$ .

Calculons la valeur de  $f_N$  en son premier extremum. On peut supposer sans perte de généralité  $N$  pair. Alors :

$$f_N(\pi/N) = \sum_{k=1}^{N/2} \frac{1}{2k-1} \sin\left((2k-1)\frac{2\pi}{N}\right).$$

Introduisons le *sinus cardinal* sinc (tel que  $\forall x \in \mathbb{R}$ ,  $\text{sinc}(x) = \sin(x)/x$ ) :

$$f_N\left(\frac{\pi}{N}\right) = \frac{1}{2} \sum_{k=1}^{N/2} \frac{2\pi}{N} \text{sinc}\left((2k-1)\frac{\pi}{N}\right).$$

On est devant une somme de Riemann, la fonction sinc étant continue, on sait que :

$$\lim_{N \rightarrow +\infty} f_N(2\pi/2N) = \frac{1}{2} \int_0^\pi \text{sinc}(x) dx \simeq \frac{\pi}{4} + \frac{\pi}{2} \cdot 0.0895 \simeq 0.926$$

(à comparer à  $\pi/4 \simeq 0.785$ ). □

**Proposition 21** *Dès qu'il y a une discontinuité dans un signal, il y a présence d'artefacts dus à l'effet de Gibbs dans le signal reconstruit à partir des sommes partielles du développement en série de Fourier.*

*Plus précisément, soit  $f$  de période  $a$  vérifiant les conditions du théorème 6, et  $x_0$  tels que  $\delta = f(x_0+) - f(x_0-) \neq 0$ . Alors :*

$$\lim_{N \rightarrow +\infty} f_N(x_0 + a/2N) - f_N(x_0 - a/2N) \simeq 1,179 \cdot \delta.$$

*Ceci signifie que l'effet de Gibbs provoque des oscillations résiduelles de hauteur égale à 18% de la valeur du saut au niveau de la discontinuité.*

*Démonstration.* Soit  $H_{x_0}$  la fonction créneau de la proposition 20 translatée de  $x_0$  et dilatée de  $a/(2\pi)$  de manière à avoir une période de  $a$ . Introduisons  $g(x) = f(x) + \delta(-1/2 + \frac{2}{\pi}H_{x_0}(x))$  qui est continue en  $x_0$  et intéressons-nous à  $g_N$ , somme partielle des  $N$  coefficients centraux de la série de Fourier de  $g$ . La valeur en  $x_0 - a/(2N)$  est  $g_N(x_0 - a/(2N)) = f_N(x_0 - a/(2N)) + (-1/2 - 0.926\frac{2}{\pi})\delta \simeq f_N(x_0 - a/(2N)) - 1.0895\delta$  et en  $x_0 + a/(2N)$  :  $g_N(x_0 + a/(2N)) = f_N(x_0 + a/(2N)) + (-1/2 + 0.926\frac{2}{\pi})\delta \simeq f_N(x_0 + a/(2N)) + 0.0895\delta$ .

Or  $g_N(x_0 + a/(2N)) - g_N(x_0 - a/(2N)) \rightarrow 0$  lorsque  $N \rightarrow +\infty$  (car  $g_N$  est continue en  $x_0$ ), donc :

$$f_N(x_0 + a/(2N)) - f_N(x_0 - a/(2N)) \rightarrow 1,179 \cdot \delta.$$

□

### 5.3 Transformée discrète en cosinus

Le calcul de la Transformée de Fourier Discrète d'un signal  $(x_n)$  de longueur  $N$  a pour effet de « périodiser » le signal. Ceci introduit des discontinuités si  $x_0 \neq x_{N-1}$ , et donc des effets de Gibbs. Une idée naturelle pour éviter cela est de commencer par symétriser le signal  $(x_n)_{0 \leq n \leq N-1}$  en  $(y_n)_{0 \leq n \leq 2N-1}$  avec  $y_n = x_n$  si  $0 \leq n \leq N-1$  et  $y_n = x_{2N-1-n}$  sinon. La figure 5.3 en est un exemple.

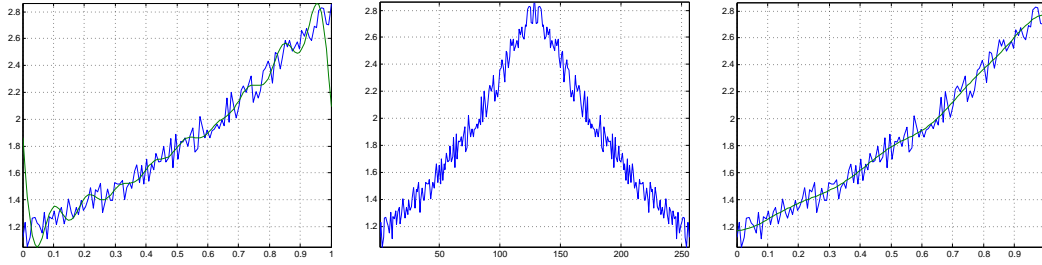


FIG. 5.3 – Exemple de la régularisation d'un signal par transformée de Fourier. Le signal original (en bleu) est de longueur 128. L'idée est d'annuler dans la transformée de Fourier les coefficients d'indice  $> N$  correspondant aux hautes fréquences (et donc au bruit dans le signal d'origine). Image de gauche :  $N = 10$ , le signal étant supposé périodique dans la TFD, il y a apparition d'effet de Gibbs aux extrémités de l'intervalle (on voit bien les oscillations). Image du milieu : le signal symétrisé, de période double. Image de droite :  $N = 10$  pour la TFD du signal symétrisé. Cette fois il n'y a plus de saut aux bornes de l'intervalle, donc plus d'effet de Gibbs.

Calculons alors la TFD de  $y$ . Soit  $n \in \mathbb{Z}$ ; successivement :

$$\begin{aligned} Y_n &= \frac{1}{2N} \sum_{k=0}^{2N-1} x_k \omega_{2N}^{-nk} \\ &= \frac{1}{2N} \sum_{k=0}^{N-1} x_k \left( \omega_{2N}^{-nk} + \omega_{2N}^{-n(2N-1-k)} \right) \\ &= \frac{1}{N} e^{i\pi n/(2N)} \sum_{k=0}^{N-1} x_k \cos \left( \frac{\pi n(k+1/2)}{N} \right) \end{aligned}$$

$$\text{car } \omega_{2N}^{-nk} + \omega_{2N}^{-n(2N-1-k)} = 2e^{\frac{2i\pi}{2N} \frac{n}{2}} \left( e^{\frac{2i\pi}{2N} (-nk-n/2)} + e^{\frac{2i\pi}{2N} (nk+n/2)} \right).$$

Finalement :

$$Y_n = \frac{1}{N} e^{i\pi n/(2N)} \sum_{k=0}^{N-1} x_k \cos \left( \frac{\pi n(k+1/2)}{N} \right).$$

Ceci nous conduit à la définition suivante.

**Définition 21** Soit  $x = (x_n)$  un signal de longueur  $N$ . La suite  $\widehat{X}$  définie par :

$$\widehat{X}_n = \sum_{k=0}^{N-1} x_k \cos\left(\frac{\pi n(k+1/2)}{N}\right) \quad (5.1)$$

pour  $n \in \{0, \dots, N-1\}$

est la **transformée discrète en cosinus** (ou **discrete cosinus transform, DCT**) de  $x$ .

*Remarque* : si  $(x_n)$  est réel, alors  $(\widehat{X}_n)$  aussi.

*Remarque* :  $Y_n = \overline{Y_{2N-n}}$  et  $e^{i\pi n/(2N)} = -e^{-i\pi(2N-n)/(2N)}$ , donc  $\widehat{X}_{2N-n} = -\widehat{X}_n$ .

Ainsi, la donnée de la DCT  $\widehat{X}$  est équivalente à la donnée de la TFD de  $x$  symétrisé. Par construction, la DCT n'est pas soumise à un effet de Gibbs à ses bords.

**Proposition 22** La formule suivante permet de définir la **DCT inverse**. Si  $(x_n)$  est un signal de longueur  $N$  et  $(\widehat{X}_n)$  sa DCT :

$$x_n = \frac{1}{N} \widehat{X}_0 + \frac{2}{N} \sum_{k=1}^{N-1} \widehat{X}_k \cos\left(\frac{\pi k(n+1/2)}{N}\right) \quad (5.2)$$

pour  $n \in \{0, \dots, N-1\}$ .

*Démonstration.* Il suffit d'appliquer la formule de la transformée de Fourier inverse au signal  $(y_n)_{0 \leq n \leq 2N-1}$  obtenu par symétrisation de  $x$ . Alors, pour  $n \in [0, N-1]$  :

$$\begin{aligned} x_n &= \sum_{k=0}^{2N-1} Y_k \omega_{2N}^{nk} \\ &= \sum_{k=0}^{N-1} \frac{1}{N} e^{i\pi k/(2N)} \widehat{X}_k \omega_{2N}^{nk} + \sum_{k=N}^{2N-1} Y_k \omega_{2N}^{nk} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} e^{i\pi k/(2N)} \widehat{X}_k \omega_{2N}^{nk} + \sum_{k=1}^N \overline{Y_k} \omega_{2N}^{n(2N-k)} \\ &= \frac{1}{N} \widehat{X}_0 + \frac{1}{N} \sum_{k=1}^{N-1} \widehat{X}_k \left( e^{i\pi k/(2N)} \omega_{2N}^{nk} + e^{-i\pi k/(2N)} \omega_{2N}^{-nk} \right) \\ &= \frac{1}{N} \widehat{X}_0 + \frac{2}{N} \sum_{k=1}^{N-1} \widehat{X}_k \cos\left(\frac{\pi k(n+1/2)}{N}\right) \end{aligned}$$

L'avant-dernière égalité est vraie car  $Y_N = 0$ . □

*Remarque :* comme il est possible de symétriser de différentes manières le signal  $x$ , il existe plusieurs formes de DCT. Nous n'entrons pas dans les détails dans cette version des notes de cours.

## **5.4 Quantification**

### **5.5 Compression MP3 et JPEG**

#### **5.5.1 Compression MP3**

#### **5.5.2 Compression JPEG**



# Chapitre 6

## Théorie de l'échantillonnage

Dans ce chapitre nous nous intéressons au problème crucial de l'échantillonnage. Les signaux physiques sont analogiques, et nécessitent donc une *numérisation* afin d'être traités sur ordinateur. La numérisation consiste à mesurer le signal physique (une tension électrique généralement) à des intervalles de temps réguliers, et coder cette mesure sur un nombre fini de bits (un *float* ou un *double* par exemple). La première partie est l'*échantillonnage* proprement dit (les mesures sont appelés *échantillons*, la seconde la *quantification*. La grandeur caractéristique de l'échantillonnage est la fréquence d'échantillonnage, ou sa cadence. Celle de la quantification est le nombre de bits alloués à chaque échantillon. Nous ne détaillerons pas ici le problème de la quantification.

### 6.1 Rappels de théorie des distributions

Voici quelques rappels de notations et résultats vus dans le cours de théorie des distributions.

- $\mathcal{S}$  : espace des fonctions  $C^\infty$  sur  $\mathbb{R}$  à décroissance rapide ;
- $\mathcal{S}'$  : espace des distributions tempérées ;
- La transformée de Fourier  $\mathcal{F}$  définit une bijection bicontinue de  $\mathcal{S}'$  sur  $\mathcal{S}$  et  $\mathcal{F}^{-1} = \overline{\mathcal{F}}$  ;
- $\Delta_a = \sum_{n \in \mathbb{Z}} \delta_{na}$  est appelé *peigne de Dirac* et est une distribution tempérée ;
- $\mathcal{F}(\delta_a)(y) = e^{-2i\pi ay}$  (par définition de  $\mathcal{F}$ ) ;
- $\mathcal{F}(\Delta_a) = \sum_{n \in \mathbb{Z}} e^{2i\pi n ay}$  (par linéarité et continuité) ;
- Si  $T \in \mathcal{S}'$ , alors  $\mathcal{F}(T(x)e^{-2i\pi\alpha x}) = \mathcal{F}(T)(y + \alpha)$ .
- Si  $f \in C^\infty$  à croissance lente et  $T \in \mathcal{S}'$ ,  $\mathcal{F}(f \cdot T) = \mathcal{F}(f) * \mathcal{F}(T)$ .

Nous aurons besoin dans la suite du résultat suivant :

**Proposition 23** *La transformée de Fourier d'un peigne de Dirac est un peigne de Dirac, et plus précisément :*

$$\mathcal{F}(\Delta_a) = \frac{1}{a} \Delta_{1/a}.$$

*Démonstration.* Soit  $f$   $a$ -périodique telle que pour  $x \in [0, a[$ ,  $f(x) = x/a$ .

D'après le cours 1A, au sens des distributions (c'est une fonction dérivable par morceaux avec des « sauts » aux points  $na$ ,  $n \in \mathbb{Z}$ ) :

$$f' = \frac{1}{a} - \sum_{n \in \mathbb{Z}} \delta_{na}.$$

Or  $f$  étant intégrable et périodique, on peut la décomposer en série de Fourier, la convergence ayant lieu dans  $L^2$  :

$$f(x) = \frac{1}{2} + \frac{i}{2\pi} \sum_{n \in \mathbb{Z}^*} \frac{1}{n} e^{2i\pi nx/a}.$$

Donc :  $f'(x) = -\frac{1}{a} \sum_{n \in \mathbb{Z}^*} e^{2i\pi nx/a}.$

Par identification des deux expressions de  $f'$  :

$$\sum_{n \in \mathbb{Z}} \delta_{na} = \frac{1}{a} \sum_{n \in \mathbb{Z}} e^{2i\pi nx/a}.$$

On conclut avec l'expression rappelée de la transformée de Fourier du peigne de Dirac. □

## 6.2 Formule de Poisson

Soit  $f$  un signal analogique, échantillonné sur des intervalles de temps de longueur  $a$ . On associe à cet échantillonnage la distribution (tempérée, car  $f$  sera à croissance lente avec la proposition 24) :

$$\tilde{f}_a = a \sum_{n \in \mathbb{Z}} f(na) \delta_{na} \quad (= af \cdot \Delta_a).$$

Ceci est motivé par  $a \sum_{n \in \mathbb{Z}} f(na) \delta_{na} \rightarrow f$  (au sens des distributions tempérées) si  $a \rightarrow 0$ .

En effet,

$$\forall \phi \in \mathcal{S}, \langle \tilde{f}_a, \phi \rangle = a \sum_{n \in \mathbb{Z}} f(na) \phi(na) \longrightarrow \int f \phi = \langle f, \phi \rangle$$



la limite se justifiant par une somme de Riemann.

On aura besoin dans la suite de la notion suivante :

**Définition 22** Soit  $f \in \mathcal{S}'$  t.q.  $\mathcal{F}(f)$  est à support compact  $\subset [-\lambda_c, \lambda_c]$ . (i.e.  $f$  n'a pas de fréquences supérieures à une fréquence limite  $\lambda_c$ )

On dit que  $f$  est à **bande limitée**.

*Remarque* : il s'agit finalement d'une hypothèse assez raisonnable : l'oeil et l'oreille sont sensibles à une gamme de fréquences bornée, un micro, un film argentique, un capteur CMOS ou CCD également.

**Définition 23**  $2\lambda_c$  est appelé **fréquence de Nyquist**.

On a également besoin de la proposition suivante :

**Proposition 24** Un signal à bande limitée est  $C^\infty$  à croissance lente.

*Démonstration.* Admis dans cette version 0.3. □

Le théorème suivant nous donne l'expression du spectre du signal échantillonné  $\tilde{f}_a$ , et le lie au spectre de  $f$ .

**Théorème 13 (Formule de Poisson)** Soit  $f \in \mathcal{S}'$  un signal à bande limité, et  $\tilde{f}_a$  son échantillonné. Alors :

$$\mathcal{F}(\tilde{f}_a) = \sum_{n \in \mathbb{Z}} \mathcal{F}(f) \left( y - \frac{n}{a} \right) = a \sum_{n \in \mathbb{Z}} f(na) e^{-2i\pi n a y}. \quad (6.1)$$

*Démonstration.* Par définition :  $\mathcal{F}(\tilde{f}_a) = \mathcal{F}(a \sum_{n \in \mathbb{Z}} f(na) \delta_{na})$ .

Par linéarité et continuité :  $\mathcal{F}(\tilde{f}_a) = a \sum_{n \in \mathbb{Z}} f(na) \mathcal{F}(\delta_{na})$

Donc  $\mathcal{F}(\tilde{f}_a) = a \sum_{n \in \mathbb{Z}} f(na) e^{-2i\pi n a y}$ .

D'autre part :  $\mathcal{F}(\tilde{f}_a) = \mathcal{F}(af \cdot \Delta_a)$ .

Donc  $\mathcal{F}(\tilde{f}_a) = a \mathcal{F}(f) * \mathcal{F}(\Delta_a) = \mathcal{F}(f) * \Delta_{1/a}$ .

Conclusion :  $\mathcal{F}(\tilde{f}_a) = \mathcal{F}(f) * \Delta_{1/a} = a \sum_{n \in \mathbb{Z}} f(na) e^{-2i\pi n a y}$  □

La formule de Poisson a deux conséquences :

1. Le spectre de  $\tilde{f}_a$  est obtenu en faisant la somme des translats de  $\mathcal{F}(f)$  de pas  $n/a$  (première partie de la formule (6.1)) ;
2. le spectre de  $\tilde{f}_a$  est périodique de période  $1/a$  (deuxième partie de (6.1)).

### 6.3 Théorème d'échantillonnage de Shannon-Nyquist

Le théorème suivant nous donne un résultat a priori surprenant : si on échantillonne un signal à bande limitée à une fréquence supérieure à sa fréquence de Nyquist, alors on ne perd aucune information sur ce signal !

**Théorème 14 (Théorème de Shannon (-Nyquist))** Soit  $f \in L^2(\mathbb{R})$  à bande limitée, telle que  $\text{supp}(\mathcal{F}(f)) \subset [-\lambda_c, \lambda_c]$ , et  $a \leq \frac{1}{2\lambda_c}$ . Alors

$$f(x) = \sum_{n \in \mathbb{Z}} f(na) \frac{\sin \frac{\pi}{a}(x - na)}{\frac{\pi}{a}(x - na)} \quad (6.2)$$

(l'égalité ayant lieu dans  $L^2$ ).

*Démonstration.* Soit  $f \in L^2$  un signal à bande limitée et  $\tilde{f}_a$  son échantillonnée t.q.  $1/a \geq 2\lambda_c$ . D'après la formule de Poisson :

$$F(\tilde{f}_a)(\lambda) = \sum_{n \in \mathbb{Z}} \mathcal{F}(f) \left( \lambda - \frac{n}{a} \right) = a \sum_{n \in \mathbb{Z}} f(na) e^{-2i\pi na \lambda}.$$

Soit  $\chi_a$  l'indicatrice du segment  $[-1/2a, 1/2a]$ . En multipliant dans les deux membres par  $\chi_a$ , on déduit :

$$\mathcal{F}(f)(\lambda) = a \sum_{n \in \mathbb{Z}} f(na) \chi_a(\lambda) e^{-2i\pi na \lambda}$$

car le support de  $\mathcal{F}(f)$  est dans  $[-1/2a, 1/2a]$  (et donc un seul terme de la première somme contribue après multiplication par  $\chi_a$ ).

Par transformée de Fourier inverse dans  $L^2$ , on déduit :

$$\begin{aligned} f(x) &= a \sum_{n \in \mathbb{Z}} f(na) \overline{\mathcal{F}} \left( \chi_a(\lambda) e^{-2i\pi na \lambda} \right) \\ &= a \sum_{n \in \mathbb{Z}} f(na) \overline{\mathcal{F}}(\chi_a)(x - na) \\ &= \sum_{n \in \mathbb{Z}} f(na) \frac{\sin \frac{\pi}{a}(x - na)}{\frac{\pi}{a}(x - na)}. \end{aligned}$$

□

Dans la section suivante on examine le cas où le signal contient des fréquences supérieures à  $1/2a$ .

## 6.4 Recouvrement de spectre ou aliasing

Si  $\frac{1}{a} < 2\lambda_c$ , alors le terme correspondant à  $n = 0$  dans le premier membre de la formule de Poisson n'est plus le seul à contribuer lorsque l'on multiplie par l'indicatrice de  $[-1/2a, 1/2a]$ . La reconstruction est perturbée par la contribution des termes en  $\mathcal{F}(f)(\lambda - 1/a)$  et  $\mathcal{F}(f)(\lambda + 1/a)$  (si la fréquence de Nyquist n'est pas beaucoup plus élevée que  $1/2a$ , et éventuellement par les  $\mathcal{F}(f)(\lambda - k/a)$  pour  $|k| > 1$  sinon).

Il y a alors un phénomène de *recouvrement*, ou *repliement de spectre* dans les hautes fréquences. On parle aussi du phénomène d'*aliasing* (*alias* = « à un autre endroit » en latin), ou *aliasage*, à cause de la copie de morceaux du spectre « à un autre endroit » que là où ils seraient attendus si la fréquence d'échantillonnage était correcte.

*Remarque* : la solution technologique est de filtrer les signaux analogiques avant échantillonnage pour éliminer les fréquences supérieures à  $1/2a$ , où  $1/a$  est la fréquence d'échantillonnage.

Des illustrations de ce phénomène seront vues en TP.

## 6.5 Retour à la transformée de Fourier discrète

On considère ici un signal analogique périodique  $f$ , échantillonné au pas de  $a$ . Les échantillons sont notés :  $x_n = f(na)$  pour  $n \in [0, N - 1]$ , la période de  $f$  étant  $Na$ .

La distribution associée à cet échantillon est :

$$\tilde{f}_a = a \sum_{n \in \mathbb{Z}} f(na) \delta_{na} = a \sum_{n=0}^{N-1} x_n \sum_{k \in \mathbb{Z}} \delta_{(n+kN)a}.$$

On calcule la transformée de Fourier :

$$\mathcal{F} \left( \sum_{k \in \mathbb{Z}} \delta_{(n+kN)a} \right) = e^{-2i\pi n a y} \mathcal{F} \left( \sum_{k \in \mathbb{Z}} \delta_{kNa} \right) \quad (6.3)$$

$$= \frac{1}{Na} e^{-2i\pi n a y} \sum_{k \in \mathbb{Z}} \delta_{k/(Na)} \quad (6.4)$$

$$= \frac{1}{Na} \sum_{k \in \mathbb{Z}} e^{-2i\pi n k / N} \delta_{k/(Na)}. \quad (6.5)$$

Ainsi, la transformée de Fourier de la distribution  $\tilde{f}_a$  est :

$$\mathcal{F}(\tilde{f}_a) = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{k \in \mathbb{Z}} x_n e^{-2i\pi n k / N} \delta_{k/(Na)} = \sum_{k \in \mathbb{Z}} X_k \delta_{k/(Na)}$$

D'autre part, en notant  $c_n$  les coefficients de Fourier de  $f$ , dans  $L_p^2$  :

$$f(x) = \sum_{n \in \mathbb{Z}} c_n e^{2i\pi n x / (Na)}.$$

Donc :

$$\mathcal{F}(f) = \sum_{n \in \mathbb{Z}} c_n \delta_{n/(Na)}.$$

Faisons à présent l'hypothèse que  $f$  est à bande limitée (donc la somme précédente est finie). Ceci permet d'utiliser la formule de Poisson (théorème 13) :

$$\sum_{k \in \mathbb{Z}} X_k \delta_{k/(Na)} = \sum_{k \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} c_n \delta_{(n+kN)/(Na)}$$

Par conséquent, en identifiant (on rappelle que la somme est finie) :

$$X_k = \sum_{n \in \mathbb{Z}} c_{k+nN}.$$

**Conséquence** : cette formule précise la validité de l'approximation  $X_k \simeq c_k$  établie dans le chapitre 2 à l'aide de la formule des trapèzes. On vient d'établir que :  $X_k \simeq c_k$  si les  $c_{k+nN}$  ( $|n| > 0$ ) sont proches de 0.

Plus précisément :

**Proposition 25 (condition de Nyquist discrète)** *La formule  $X_k = c_k$  est valide si le signal  $f$  ne présente pas de fréquences supérieures à  $N/(2a)$ , où  $N$  est le nombre d'échantillons sur l'intervalle, et  $a$  le pas d'échantillonnage.*

Dans le cas contraire, il y a aliasing.

# Chapitre 7

## Illustration : sous et sur-échantillonnage

Dans cette partie nous examinons deux cas pratiques mettant en jeu des notions vues dans les chapitres précédents.

### 7.1 Sur-échantillonnage par *zero padding*

Le sur-échantillonnage d'une image consiste à augmenter sa taille. Il s'agit d'une opération de plus en plus utilisée pour (par exemple) adapter les anciennes vidéos à la diffusion dite « haute définition ».

On considère des images de taille  $N \times N$ , et on veut les interpoler en des images de taille  $kN \times kN$  (avec  $k$  valant 2, 4, 8).

Voici trois méthodes possibles :

- duplication des pixels. On remplace chaque pixel de l'image originale par un bloc de  $k \times k$  pixels ayant le même niveau de gris (obtenu par duplication du pixel original) ;
- on calcule la transformée de Fourier de l'image originale de taille  $N \times N$ , on la complète par des zéros pour former une matrice de taille  $kN \times kN$ , puis on prend la transformée de Fourier inverse ;
- même raisonnement, mais avec la transformée en cosinus.

Les deux dernières méthodes sont dites par *zero padding* (remplissage, ou bourrage, par des zéros). Ceci est illustré par la figure 7.1.

La figure 7.2 montre le résultat de la méthode par duplication par pixel. Le sur-échantillonnage n'est bien sûr pas très satisfaisant car il fait apparaître des blocs uniformes. Les méthodes par *zero padding* font apparaître un effet de ringing (ou effet de Gibbs) le long des contours contrastés présents dans l'image, ce qui est signe de coefficients de Fourier passant brutalement à 0 dans les hautes fréquences. Il s'agit d'un

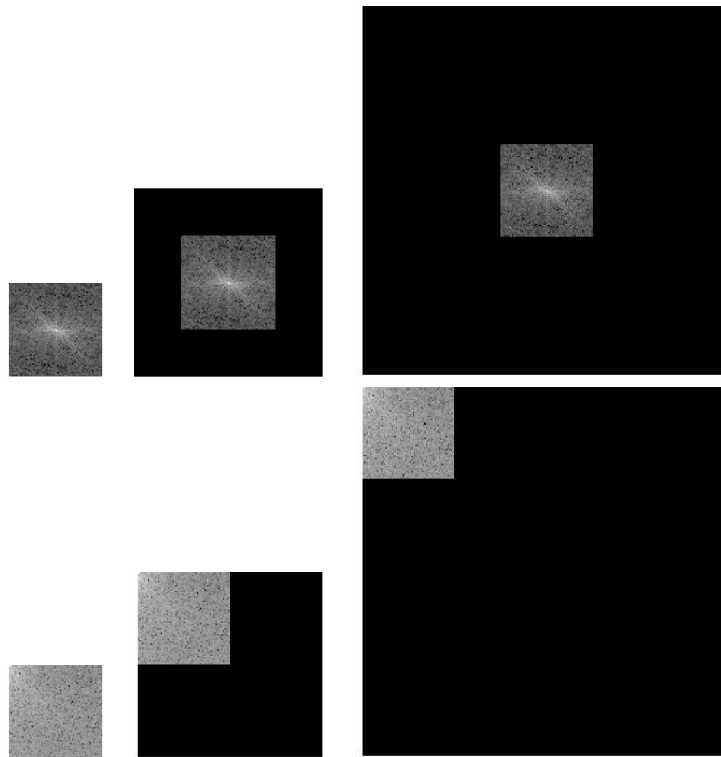


FIG. 7.1 – Remplissage par zéros (zero padding) : en haut par la transformée de Fourier (TFD), en bas par la transformée en cosinus discret (DCT). À gauche, TFD / DCT de l'image originale. Au milieu, TFD/DCT de l'image sur-échantillonnée d'un facteur 2, obtenue par remplissage par des zéros. Au milieu, TFD/DCT de l'image sur-échantillonnée d'un facteur 4, obtenue par remplissage par des zéros.

artefact inhérent à la méthode. L'avantage du zero padding appliqué à la DCT est l'absence de ringing sur les bords extérieurs de l'image, comme dans l'expérience du paragraphe 5.3 dans le chapitre 5. La DCT est en effet équivalente à la TFD sur l'image symétrisée qui ne présente alors plus de transitions abruptes aux bords.



FIG. 7.2 – Sur-échantillonnage. En haut : duplication de pixels. Au milieu : zero padding dans la transformée de Fourier. En bas : zero padding dans la transformée en cosinus discrets. L'image de droite est l'image originale de taille  $64 \times 64$ . L'image du milieu a pour taille  $128 \times 128$ , et celle de droite  $256 \times 256$ . On voit apparaître dans les méthodes de zero padding un phénomène de ringing le long des contours présents dans l'image. Lorsque l'on utilise la DCT, on élimine au moins le ringing sur les bords extérieurs de l'image. Ce phénomène est par exemple visible sur le bord supérieur gauche des images. Les artefacts sont ici d'autant plus visibles que la résolution de l'image originale a été choisie très faible, pour des raisons de lisibilité.

## 7.2 Sous-échantillonnage

On dispose d'images de taille  $N \times N$ , et on veut les réduire à une taille  $N/2 \times N/2$  ou  $N/4 \times N/4$ . On parle de sous-échantillonnage de facteur 2 ou 4. Comme le sur-échantillonnage, le sous-échantillonnage est une opération classique pour la diffusion de contenus sur les canaux numériques type TNT, Blue-Ray, DVD...et des artefacts sont visibles lorsque l'opération est mal faite.

La méthode naïve consiste à procéder à une décimation : on garde un pixel sur deux (resp. quatre) par ligne et colonne. La fréquence d'échantillonnage dans chaque direction est alors  $N/2$  (resp.  $N/4$ ). Pour être en accord avec la condition de Shannon-Nyquist, cette fréquence d'échantillonnage doit être au moins le double de la plus haute fréquence  $f$  présente dans l'image :  $N/2 > 2f$  (resp.  $N/4 > 2f$ ).

*Rappel pour ce qui suit* : le coefficient de Fourier discret  $X_{n,m}$  ( $-N/2+1 \leq n, m \leq N/2$ ) est une approximation du coefficient de Fourier de l'image continue correspondant aux fréquences  $n$  (horizontalement) et  $m$  (verticalement).

Rappelons que la section 6.5 et en particulier la proposition 25 (aliasing discret) précisent les conditions dans lesquelles cette approximation est valable.

### 7.2.1 Méthode 1 : décimation brutale

Un exemple est traité sur la figure 7.3. En haut on voit l'image originale et sa transformée de Fourier. Une décimation brutale d'un facteur 2 correspondant à un échantillonnage de fréquence  $N/2$ , on observe un repliement des coefficients correspondant aux fréquences supérieures à  $N/4$ .

Plus précisément, d'après le paragraphe 6.5 du chapitre 6, les coefficients de Fourier  $\tilde{X}_{k,l}$  de l'image décimée s'expriment en fonction de coefficients de Fourier  $X_{k,l}$  de l'image originale par :

$$\tilde{X}_{k,l} = \sum_{n,m \in \{-1,0,1\}} X_{k+nN/4,l+mN/4}. \quad (7.1)$$

D'après la décroissance des coefficients de Fourier, les basses fréquences ( $k, l$  petits) ne sont pas très perturbées car les termes  $X_{k+nN/4,l+mN/4}$  pour  $n, m \in \{-1, 1\}$  correspondent à des hautes fréquences. Par contre, les hautes fréquences ( $k, l$  proches de  $N/4$ ) sont très perturbées par les termes additionnels qui, eux, correspondent à des basses fréquences (valeurs élevées des coefficients a priori).

La figure 7.4 illustre le phénomène de repliement du spectre. Au passage, on remarque après examen minutieux que l'image originale n'est pas exempte d'aliasing, cf figure 7.5.

On voit que cet aliasing se traduit effectivement par des perturbations sur les structures « hautes fréquences » : les façades des immeubles sur lesquels apparaissent des



« images fantômes » et les arêtes de ces immeubles qui ont un aspect en « marches d'escalier ».

### 7.2.2 Méthode 2 : filtrage passe-bas idéal

Une première idée pour éliminer l'aliasing serait de :

1. mettre à zéros les coefficients des fréquences plus grandes que  $N/4$  ou  $N/8$  (il s'agit d'un filtre passe-bas) ;
2. récupérer la transformée de Fourier inverse de ce spectre modifié ;
3. puis de procéder à la décimation sur cette dernière image, qui ne contiendra alors plus de fréquences supérieure à la moitié de la fréquence d'échantillonnage.

La figure 7.6 montre le résultat de cette expérience. On constate bien l'absence d'aliasing. L'annulation des hautes fréquences a pour conséquence de « flouter » les façades et les arêtes des immeubles.

Néanmoins, en regardant plus précisément, on se rend compte que l'annulation brutales des hautes fréquences se traduit par un effet de ringing (phénomène de Gibbs, voir figure 5.2). En effet, faire le produit du spectre par un filtre passe-bas idéal est équivalent à faire le produit de convolution de l'image originale par la transformée de Fourier inverse du filtre passe-bas idéal, à savoir un sinus cardinal. La décroissance lente du sinus cardinal est responsable de cet effet de ringing à relativement grande distance des arêtes des immeubles.

### 7.2.3 Méthode 3 : filtrage passe-bas Butterworth

Dans la méthode précédente, l'effet de Gibbs est dû à la décroissance lente de la transformée de Fourier (inverse) du filtre passe-bas idéal. Ceci est logique car ce filtre n'est pas continu. La proposition 19 nous assure qu'un filtre plus régulier aura une décroissance plus rapide dans l'espace de Fourier.

On cherche donc à approcher le filtre passe-bas idéal par une fonction plus régulière. On utilise le filtre (dit de Butterworth) de la figure 7.7, qui est plus régulier que le filtre passe-bas idéal et a donc une réponse décroissant plus rapidement. Cela diminue l'effet de ringing, comme le montre la figure 7.8.

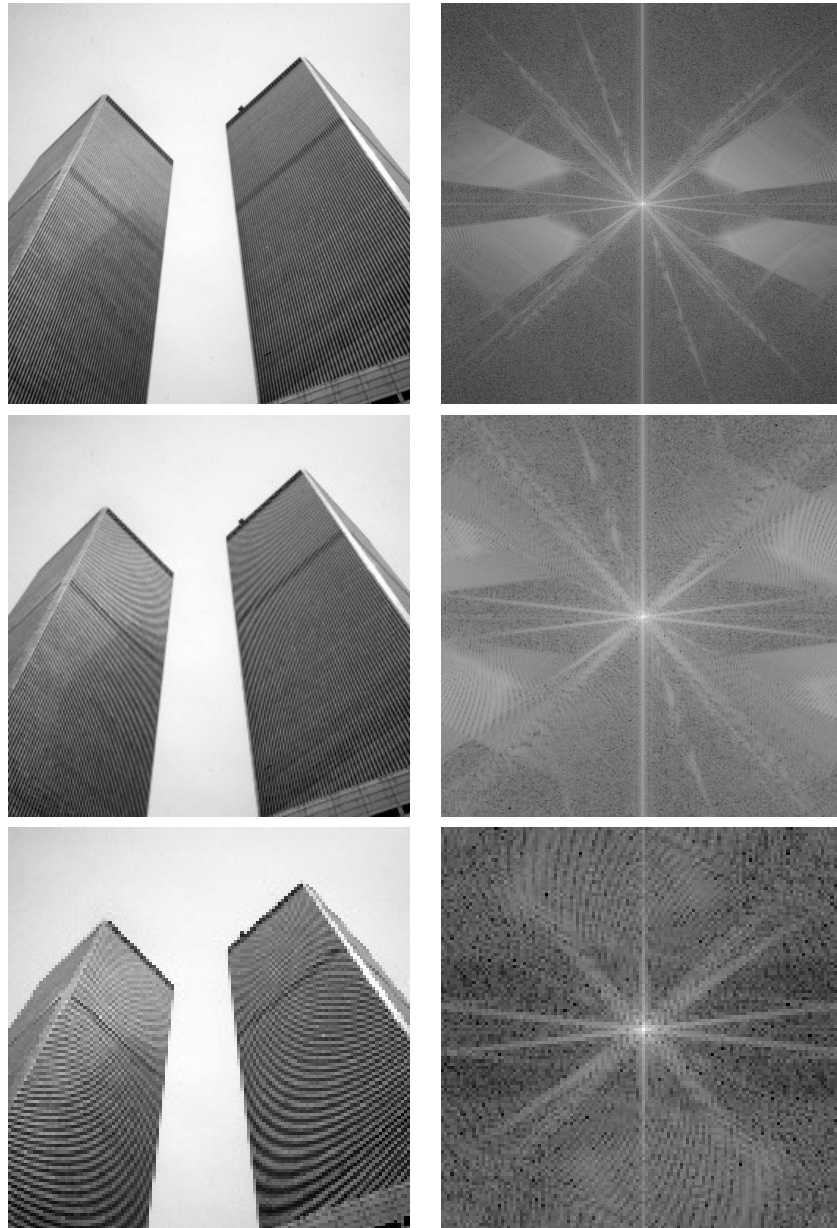


FIG. 7.3 – Décimation brutale. En haut : image originale et spectre (en fait, logarithme du module de la transformée de Fourier). Au milieu, image décimée d'un facteur 2 (à taille égale sa résolution est donc divisée par 2 par rapport à l'image originale) et spectre. En bas : image décimée d'un facteur 4 et spectre. On voit bien les multiples repliements dans le spectre des images décimées, se traduisant par des formes « fantômes » dans les façades et des arêtes en « marche d'escalier ».

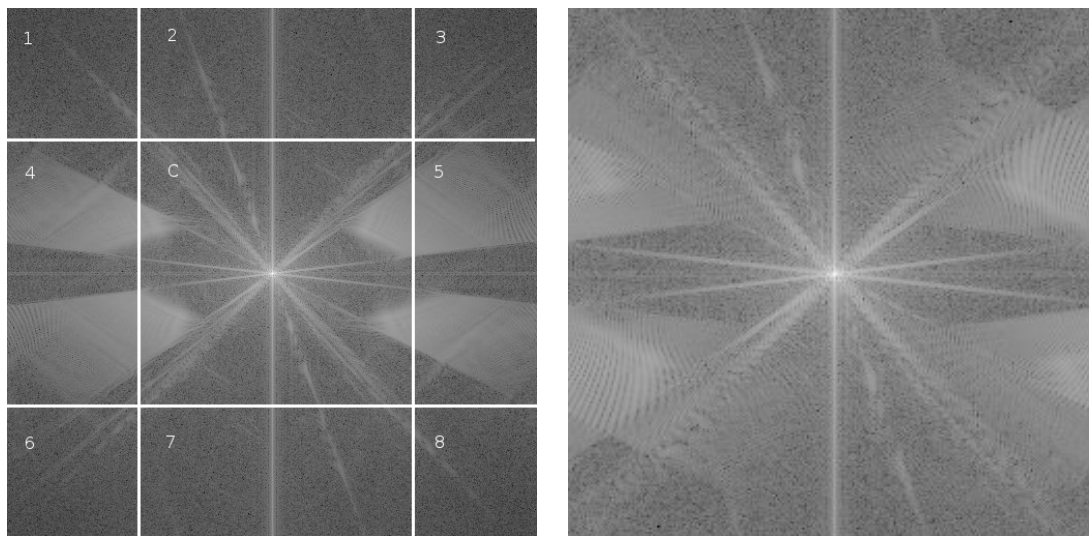


FIG. 7.4 – Illustration du phénomène de repliement de spectre lors de la décimation de facteur 2. À gauche, spectre de l'image originale. D'après l'équation 7.1, le spectre de l'image décimée est obtenue en sommant les coefficients des différentes régions. Graphiquement : la région 1 se superpose sur le quadrant inférieur droit de la région C, la région 3 sur le quadrant inférieur gauche, la région 6 sur le quadrant supérieur droit, la région 8 sur le quadrant supérieur gauche, et les régions 2, 4, 5, et 7, respectivement sur les moitiés inférieure, droite, gauche, supérieure de la région C. Ceci explique l'allure du spectre de l'image décimée, à droite.

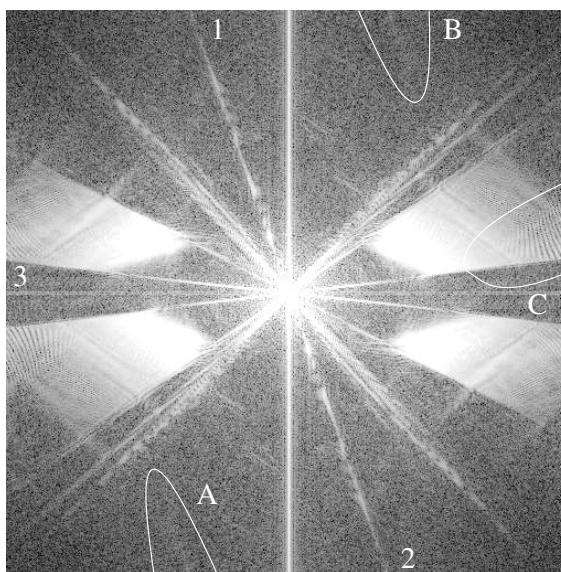


FIG. 7.5 – Un examen attentif du spectre de l'image originale montre qu'elle présente un léger aliasing. En effet, les parties A, B et C sont manifestement des repliements des parties 1, 2 et 3. Les lignes droites du spectre correspondent aux droites de l'image (les arêtes des immeubles), les « lobes » dans les hautes fréquences aux répétitions des fenêtres. Le contraste a été accentué pour une meilleure lisibilité.

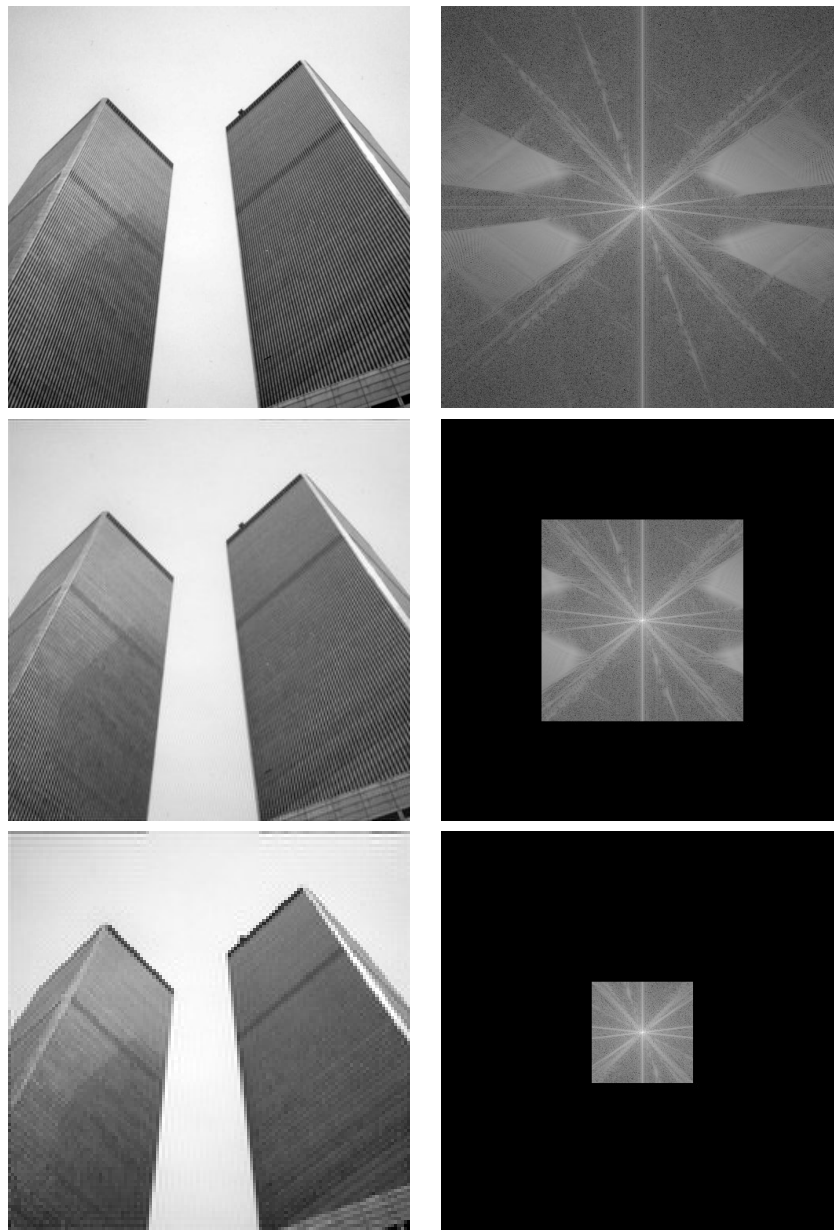


FIG. 7.6 – Filtre passe-bas idéal. En haut, image originale et spectre. Au milieu : on applique un filtre passe bas idéal annulant les fréquences supérieures à  $N/4$  de l'image initiale (à droite), puis on pratique la décimation sur la transformée de Fourier inverse (à gauche). En bas : idem pour une décimation d'un facteur 4. L'avantage par rapport à la décimation brutale est l'absence d'aliasing le long des arêtes et sur les façades des immeubles. L'inconvénient est l'apparition d'effet ringing le long des arêtes des immeubles. La périodicité induite par l'utilisation de la transformée de Fourier entraîne aussi un effet de ringing en haut des images sous-échantillonnées, conséquence de la discontinuité au niveau de la base des immeubles dans l'image périodisée. On n'aurait pas observé ce dernier phénomène en utilisant la transformée en cosinus (DCT).

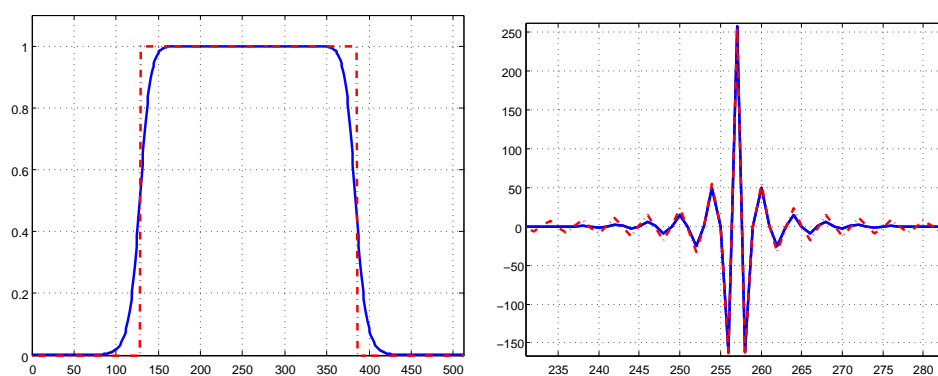


FIG. 7.7 – À gauche : spectres du filtre passe-bas idéal (pointillés rouges) et du filtre de Butterworth l’approchant. À droite, partie réelle des transformées de Fourier. On constate que la réponse du filtre de Butterworth s’atténue plus rapidement que celle du filtre passe-bas idéal, ce qui a pour conséquence d’atténuer sensiblement l’effet de ringing.

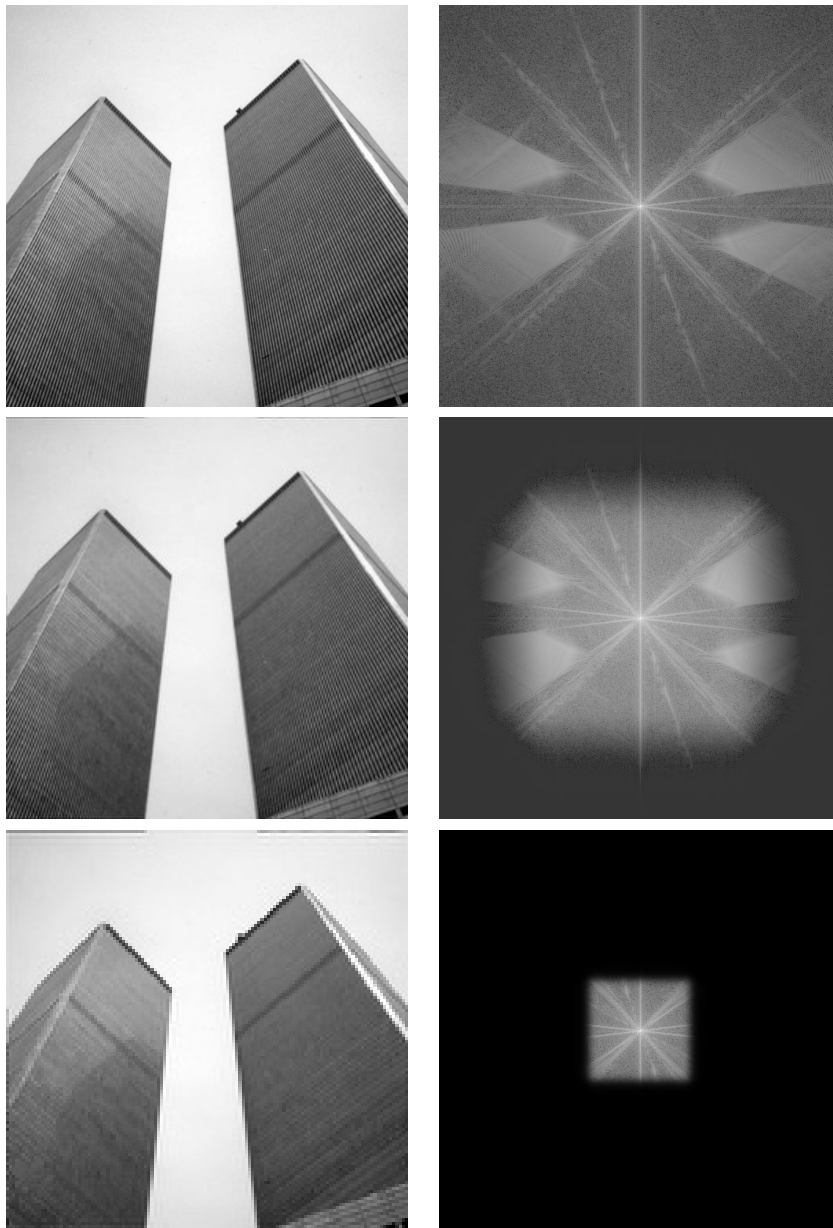


FIG. 7.8 – Filtrage passe-bas Butterworth. On applique la même procédure que celle de la figure 7.6 en remplaçant le filtre passe-bas idéal par un filtre de Butterworth. L'effet de ringing le long des contours des bâtiments est atténué.





# Chapitre 8

## Analyse temps-fréquence

L'analyse de Fourier classique présente une limitation intrinsèque : l'analyse étant globale, il y a perte de l'information temporelle.

### 8.1 Principe d'incertitude

Le théorème suivant dit qu'un signal ne peut pas être à la fois concentré en temps et en fréquence.

Soit un signal  $f \in L^2(\mathbb{R})$ , que l'on peut supposer sans perte de généralité tel que :  $\int_{\mathbb{R}} f = 0$  (i.e. sa moyenne est nulle). On supposera également pour la démonstration suivante que  $t|f(t)|^2 \rightarrow 0$  lorsque  $t \rightarrow \pm\infty$ .

On note :

$$- \sigma_t^2 = \frac{1}{\|f\|_2^2} \int_{\mathbb{R}} t^2 |f(t)|^2 dt \quad (\text{dispersion d'énergie en temps})$$

$$- \sigma_\lambda^2 = \frac{1}{\|f\|_2^2} \int_{\mathbb{R}} \lambda^2 |\widehat{f}(\lambda)|^2 d\lambda \quad (\text{dispersion d'énergie en fréquence})$$

**Théorème 15 (Théorème d'incertitude)** *On ne peut pas localiser un signal simultanément en temps et en fréquence car :*

$$\sigma_t \cdot \sigma_\lambda \geq \frac{1}{4\pi}. \quad (8.1)$$

*Démonstration.* Tout d'abord  $\sigma_\lambda^2 = \frac{1}{\|f\|_2^2} \int_{\mathbb{R}} \lambda^2 |\widehat{f}(\lambda)|^2 d\lambda = \frac{1}{4\pi^2 \|f\|_2^2} \int_{\mathbb{R}} |f'(\lambda)|^2 d\lambda$ , donc par l'égalité de Parseval :

$$\sigma_\lambda^2 = \frac{1}{4\pi^2 \|f\|_2^2} \int_{\mathbb{R}} |f'(t)|^2 dt.$$

Successivement :

$$\begin{aligned}
 \sigma_t \cdot \sigma_\lambda &= \frac{1}{2\pi \|f\|_2^2} \|tf(t)\|_2 \|f'\|_2 \\
 &\geq \frac{1}{2\pi \|f\|_2^2} | \langle tf(t), f' \rangle | \quad (\text{inégalité de Cauchy-Schwarz}) \\
 &\geq \frac{1}{2\pi \|f\|_2^2} \left| \langle tf(t), f' \rangle + \overline{\langle tf(t), f' \rangle} \right| \quad (\text{inég. triangulaire}) \\
 &= \frac{1}{4\pi \|f\|_2^2} \left| \int_{\mathbb{R}} t \left( f(t) \overline{f'(t)} + \overline{f(t)} f'(t) \right) dt \right| \\
 &= \frac{1}{4\pi \|f\|_2^2} \left| \int_{\mathbb{R}} t \frac{d}{dt} (|f|^2)(t) dt \right| \\
 &= \frac{1}{4\pi}
 \end{aligned}$$

car  $\int_{\mathbb{R}} t \frac{d}{dt} (|f|^2)(t) dt = [t|f(t)|^2]_{-\infty}^{+\infty} - \int_{\mathbb{R}} |f(t)|^2 dt$  et  $t|f(t)|^2 \rightarrow 0$ . □

*Remarque :* peut-on tout de même trouver des fonctions à support compact dont la transformée de Fourier (ou la TF inverse) est encore à support compact ? D'après le théorème précédent la taille de ces supports varierait inversement l'un de l'autre. De telles fonctions seraient intéressantes pour la conversion digital  $\rightarrow$  analogique : dans la démonstration du théorème de Shannon on pourrait utiliser une telle fonction pour isoler les lobes de la transformée de Fourier, ce qui assurerait une reconstruction par une somme finie d'échantillons.

On peut démontrer que **c'est impossible**... (cf Mallat chap. 2.)

Les Gaussiennes réalisent l'optimum dans le théorème d'incertitude, i.e.

**Proposition 26** *Il y a égalité dans le théorème d'incertitude si et seulement si  $f(t) = \alpha e^{-ct^2}$  avec  $c > 0$ .*

*Démonstration.* Il y a égalité dans le théorème 15 si et seulement si  $tf$  et  $f'$  sont proportionnels d'où  $f(t) = \alpha e^{-ct^2}$  avec  $c > 0$  pour que  $f \in L^2$ . □

*Remarque :* après calcul de la transformée de Fourier  $\hat{g}$  d'une Gaussienne  $g$  ( $\hat{g}$  est elle-même une Gaussienne), on voit que si la variance de  $g$  augmente, alors celle de  $\hat{g}$  diminue, et réciproquement. Ceci est bien sûr en accord avec le principe d'incertitude.

## 8.2 Transformée de Fourier à fenêtres

...ou Transformée de Fourier à court terme (Short-Time Fourier Transform).

Partant du constat que la transformée de Fourier classique ne permet pas une analyse localisée, on peut multiplier au préalable le signal par une fenêtre glissante  $w$ , traduite de  $b$  dans le temps. On obtient une transformée de Fourier dite à *fenêtre*, avec un paramètre supplémentaire,  $b$ .

**Définition 24 (Transformée de Fourier à fenêtre)** Soit

$$W_f(\lambda, b) = \int_{\mathbb{R}} f(t)w(t-b) \exp(-2i\pi\lambda t) dt. \quad (8.2)$$

$W_f$  s'appelle la **transformée de Fourier à fenêtre** de  $f$ .

Cette définition est motivée par le théorème suivant.

**Théorème 16** Si  $w \in L^1 \cap L^2$  réelle et  $\|w\|_2 = 1$  alors

1. Conservation de l'énergie :

$$\iint_{\mathbb{R}^2} |W_f(\lambda, b)|^2 d\lambda db = \int_{\mathbb{R}} |f(t)|^2 dt \quad (8.3)$$

2. Formule de reconstruction :

$$f(t) = \iint_{\mathbb{R}^2} W_f(\lambda, b)e^{2i\pi\lambda t} d\lambda db \quad (8.4)$$

au sens

$$\left\| f - \int_{|\lambda| \leq A} \int_{b \in \mathbb{R}} W_f(\lambda, b)w(t-b)e^{2i\pi\lambda t} d\lambda db \right\|_2 \rightarrow 0 \text{ quand } A \rightarrow +\infty.$$

*Démonstration.* cf Gasquet-Witomski chap. 41 ou Mallat chap. 4.

Rapidement (pour la reconstruction), en supposant  $\int_{\mathbb{R}} w = 1$  :

$$\begin{aligned} \widehat{f}(\lambda) &= \int_{\mathbb{R}} f(t) \exp(2i\pi\lambda t) dt \\ &= \int_{\mathbb{R}} \int_{\mathbb{R}} f(t)w(t-b) \exp(2i\pi\lambda t) db dt \\ &= \int_{\mathbb{R}} W_f(\lambda, b) db. \end{aligned}$$

Comme  $f(t) = \int_{\mathbb{R}} \widehat{f}(\lambda) \exp(2i\pi\lambda t) d\lambda$ , on déduit par substitution la formule de reconstruction.

□

Notons  $\tilde{w}_{\lambda,b}(t) = w(t-b) \exp(-2i\pi\lambda t)$ . Alors  $W_f(\lambda, b) = \int_{\mathbb{R}} f \tilde{w}_{\lambda,b}$ .

Ainsi l'information sur  $f$  obtenue par le coefficient  $W_f(\lambda, b)$  est localisée sur le support de  $\tilde{w}_{\lambda,b}$ , donc autour de  $b$ .

Mais d'après la formule de Parseval (revoquez votre cours de première année), on a aussi  $W_f(\lambda, b) = \int_{\mathbb{R}} \widehat{f \tilde{w}_{\lambda,b}}$ . D'après les formules classiques sur la transformée de Fourier,  $\widehat{f \tilde{w}_{\lambda,b}}$  apparaît comme la transformée de Fourier de  $w(\cdot - b)$  (qui est celle de  $w$  avec un déphasage) translatée de  $\lambda$ . Donc l'information sur  $f$  obtenue par le coefficient  $W_f(\lambda, b)$  est aussi localisée sur le support de  $\widehat{w}_{\lambda,b}$ , donc autour de  $\lambda$ .

Ainsi, on est intéressé par une fenêtre localisée à la fois en temps (meilleure précision de l'analyse temporelle) et en fréquence (meilleure précision de l'analyse fréquentielle). Le théorème d'incertitude nous dit qu'un compromis est nécessaire : on ne peut avoir simultanément une grande précision en temps et en fréquence.

Ceci est après tout assez intuitif : si on veut identifier avec précision la fréquence d'un signal monochromatique, on peut penser qu'il est logique de l'examiner suffisamment longtemps (au moins sur une période). Mais cela sera au détriment de la localisation temporelle d'un éventuel phénomène transitoire.

Si la fenêtre  $w$  est une Gaussienne (cas d'égalité dans le théorème d'incertitude), alors on parle de *transformée de Gabor*.

### 8.3 Illustration du principe d'incertitude

La figure 8.1 illustre le principe d'incertitude (théorème 15). Il s'agit de l'analyse temps-fréquence d'un signal constitué de la succession d'ondes monochromatiques. Plus on augmente la taille de la fenêtre d'analyse, plus la localisation en fréquence est bonne, mais au prix d'une détérioration de la localisation en temps.

### 8.4 Analyse d'un « chirp »

Un *chirp* (gazouillis, pépiement en anglais) est un signal modulé en fréquence, de la forme

$$f(t) = a(t) \sin(\phi(t)),$$

avec  $a$  et  $\phi$  des fonctions réelles.

Il est clair que pour obtenir des informations sur la « fréquence instantanée »  $\phi(t)$ , l'analyse de Fourier classique n'est d'aucune aide. Il est nécessaire d'analyser localement le signal par les outils de l'analyse temps-fréquence.

La figure 8.2 montre l'analyse du chirp défini sur l'intervalle  $[0, 10]$  par

$$y(t) = \sin(3^t),$$

échantillonné à la fréquence de  $10^4$  Hz (10 000 échantillons par unité de temps). La fenêtre utilisée est une Gaussienne.

Comme précisé dans la légende de la figure, on observe un phénomène d'aliasing. Il est difficile de quantifier davantage de manière simple, car il faudrait définir une « fréquence instantanée » associée au signal  $\sin(3^t)$  qui donnerait le temps d'apparition du phénomène grâce à la condition de Nyquist-Shannon. Autrement dit, il faudrait approcher sur un petit intervalle de temps  $\sin(3^t)$  par un signal monochromatique  $\sin(2\pi ft)$ . Le théorème d'incertitude nous donne peu d'espoirs car pour mesurer précisément cette fréquence, il faudrait examiner le signal sur un intervalle de temps suffisamment grand, pendant lequel la fréquence va changer...

Tentons un raisonnement heuristique. On cherche  $f$  tel que  $\sin(2\pi ft) = \sin(3^t)$ . En dérivant, on obtient :  $2\pi f \cos(2\pi ft) = \log(3)3^t \cos(3^t)$ . Comme sinus et cosinus ne diffèrent que d'un déphasage, on peut supposer les cosinus égaux. D'où une estimation de la fréquence instantanée  $f$  :

$$f = \frac{\log(3)3^t}{2\pi}.$$

Ceci est cohérent avec la courbe « en exponentielle » de la figure 8.2.

D'après la condition de Nyquist-Shannon, le signal est correctement échantillonné tant que  $2f$  est inférieur à la fréquence d'échantillonnage, soit :

$$\frac{\log(3)}{\pi} 3^t < 10^4.$$

D'où :

$$t < \frac{\log\left(\frac{\pi 10^4}{\log(3)}\right)}{\log(3)} \simeq 9.34.$$

Il s'agit d'une approximation du temps d'apparition du repliement de spectre qui semble correcte d'après le graphe de la figure 8.2 (rappelons que la durée totale du signal est de 10).

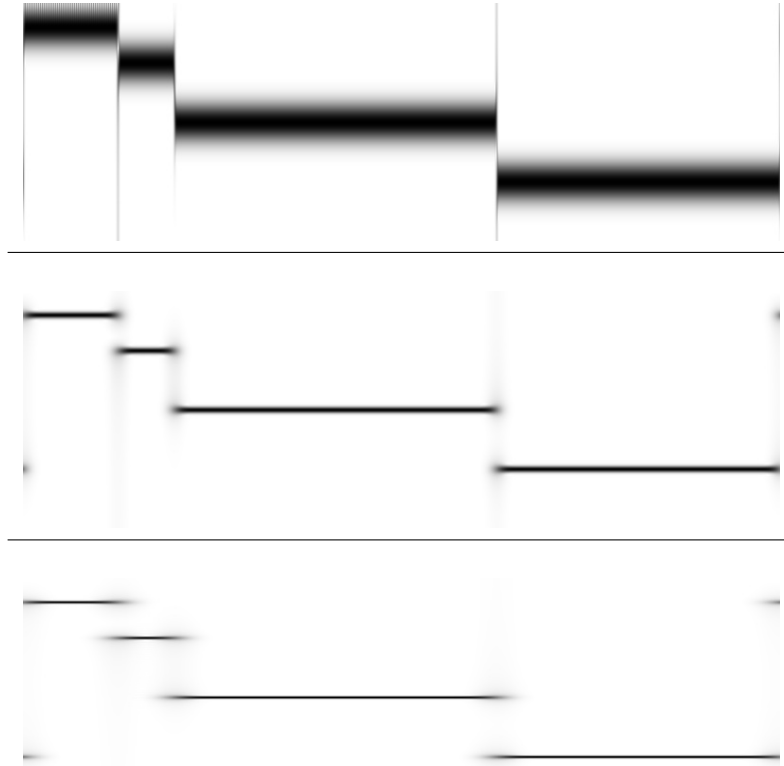


FIG. 8.1 – Exemple d’analyse temps fréquence par fenêtre Gaussienne. Ici, les valeurs des modules des spectrogrammes sont représentées proportionnellement au niveau de gris. De haut en bas la taille de la fenêtre d’analyse (définie par l’écart-type de la Gaussienne) augmente. Sur chaque image, le temps est en abscisse et la fréquence en ordonnée (les basses fréquences sont en haut). Le signal analysé est une succession de sinusoïdes : 10 Hz durant 2.5 sec, puis 25 Hz durant 1.5 sec, puis 50 Hz durant 8.75 sec, et enfin 75 Hz durant 7.5 sec. Le spectrogramme « idéal » devrait donc être constant par morceaux. On constate que plus la fenêtre d’analyse est large, plus la fréquence du signal est finement localisée (les barres horizontales sont plus fines). La contrepartie est une moins bonne localisation temporelle (la transition entre les différentes barres horizontales est moins nette, il y a un recouvrement).

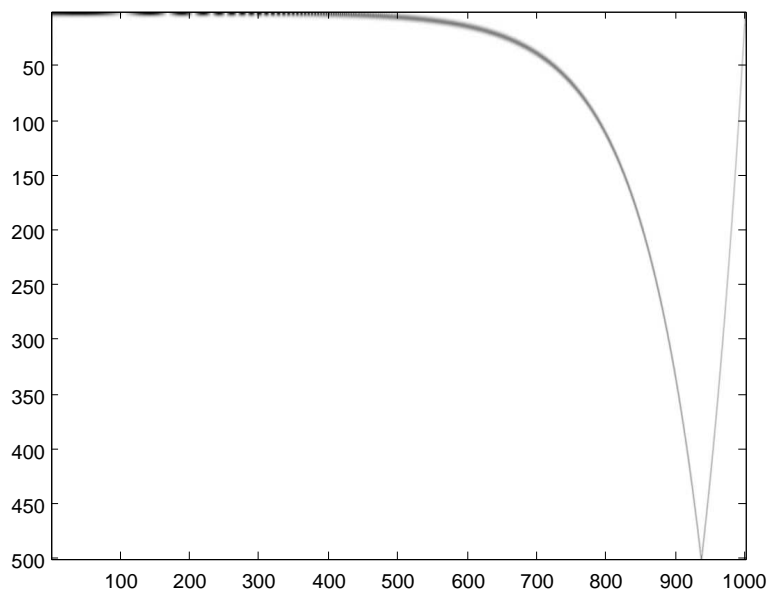


FIG. 8.2 – Analyse du « chirp »  $y(t) = \sin(3^t)$  sur l'intervalle  $[0, 10]$ . L'image représente le spectrogramme (hautes valeurs en noir, fréquence en ordonnée, temps en abscisse). La fréquence d'échantillonnage est de  $10^4$  Hz. On observe que l'allure générale du spectrogramme montre bien une croissance exponentielle de la fréquence, jusqu'à un temps où la courbe semble « rebondir ». Il ne s'agit que d'une manifestation du phénomène d'aliasing : le signal est correctement représenté par son échantillonnage tant que sa « plus haute fréquence » est inférieure à la moitié de la fréquence d'échantillonnage.





# Chapitre 9

## Bestiaire

*Références principales :*

- [wikipedia.org](http://wikipedia.org)
- *The MacTutor History of Mathematics archive*  
[www-history.mcs.st-andrews.ac.uk](http://www-history.mcs.st-andrews.ac.uk)

**Friedrich Wilhelm Bessel**, 1784 (Minden) - 1846 (Königsberg). Mathématicien et astronome allemand. Le premier à avoir mesuré précisément la distance d'une étoile, a introduit les fonctions dites de Bessel (« découvertes » par Daniel Bernoulli) dans les problèmes de mécanique céleste.

**James Cooley**, né en 1926 (Virginie). Mathématicien américain, il a co-développé avec John Tuckey l'algorithme de la transformée de Fourier rapide (FFT) lorsqu'il travaillait au IBM Watson Research Center, en 1965. La motivation de ces travaux était la surveillance des essais nucléaires de l'URSS par des réseaux de capteurs sismiques, qui nécessitaient un calcul rapide de la transformée de Fourier.

**Joseph Fourier** 1768 (Auxerre) - 1830 (Paris). Mathématicien et physicien français connu pour ses travaux (alors qu'il était préfet à Grenoble) sur la décomposition de fonctions périodiques en séries trigonométriques convergentes, appelées séries de Fourier, et leur application au problème de la propagation de la chaleur. Il est le premier à avoir présenté ce qu'on appelle de nos jours « effet de serre ».

**Dennis Gabor** 1900 (Budapest) - 1979 (Londres). Physicien hongrois, prix Nobel de physique en 1971 pour l'invention de l'holographie. Après avoir fui l'Allemagne nazie (thèse à Berlin en 1927), enseigne à Rugby puis à l'Imperial College. S'intéresse à de nombreux sujets, dont la *synthèse granulaire* du son (la transformée de Fourier à fenêtre). Citation : *The best way to predict the future is to invent it.*

**Willard Gibbs** 1839 (Connecticut) - 1903 (idem). Physicien et chimiste américain. Importants travaux en thermodynamique et physique statistique. L'un des premiers (en 1898, 50 ans *après* l'obscur mathématicien anglais Henry Wilbraham) à avoir observé et commenté les problèmes de convergences des sommes partielles de séries de Fourier sur la machine mécanique de tracé de Michelson. Citation : *A mathematician may say anything he pleases, but a physicist must be at least partially sane.*

**Johan Jensen** 1859 (Nakskov) - 1925 (Copenhague). Mathématicien et ingénieur danois. Connu pour son inégalité sur les fonctions convexes et des travaux en analyse complexe.

**Oliver Heaviside** 1850 (Londres) - 1925 (Torquay). Ingénieur autodidacte anglais. Il développe le « calcul opérationnel » (introduit son célèbre échelon), dont le cadre correct sera donné par la théorie des distributions. On lui doit de nombreux termes en électromagnétisme (conductance, impédance, inductance. . .).

**David Huffman** 1925 (Ohio) - 1999. Un des pères de la théorie de l'information, a proposé son codage lors d'un projet pour le cours de Robert Fano alors qu'il était étudiant au MIT au début des années 1950. Il n'a jamais déposé un seul brevet, se consacrant à l'enseignement. Citation : *My products are my students.*

**Leon Kraft**. Il mit en évidence par son inégalité les limites des codes préfixes dans son mémoire de master au MIT en 1949.

**Leon Lucy** né en 1938. Astronome britannique, en poste à Columbia puis à Imperial College. Introduit ce qui s'appelle de nos jours « algorithme de Richardson-Lucy » dans un article en 1974.

**Harry Nyquist** 1889 (Suède) - 1976 (Texas). Physicien aux AT&T Bell laboratories, travailla (entre autre) sur le télégraphe, le fax, la télévision. Il mit en évidence la condition de Nyquist sur la transmission par télégraphe en 1927.

**Marc-Antoine Parseval** 1755 (Rosières-aux-Salines) - 1836 (Paris). Mathématicien français, célèbre pour ses travaux sur son égalité (1799).

**Siméon Denis Poisson** 1781 (Pithiviers) - 1840 (Sceaux). Mathématicien et physicien français, enseignant à l'École Polytechnique, la faculté des sciences de Paris, l'école militaire de Saint-Cyr. Fut également astronome au bureau des longitudes. Son nom est associé à de nombreux résultats concernant l'électromagnétisme, la résistance des matériaux, les probabilités, l'analyse de Fourier. . .

**William Richardson**. Chercheur en optique américain. Introduit l'algorithme qui porte de nos jours le nom de « Richardson-Lucy » dans un article publié en 1972.

**Claude Shannon**, 1916 (Michigan) - 2001 (Massachusetts). Ingénieur mathématicien américain au MIT et aux AT&T Bell Laboratories. Commence à travailler pendant la guerre sur les problèmes de télécommunication et de transmission du signal en présence de bruit. On lui doit l'article fondateur *A Mathematical Theory of Communications* (1948).

**John Tuckey**, 1915 (Massachusetts) - 2000 (New Jersey). Statisticien américain à Princeton et aux AT&T Bell Laboratories, co-inventeur de la transformée de Fourier rapide (FFT, 1965), a apporté de nombreuses contributions dans différents domaines de la statistique. Selon Claude Shannon, il est l'« inventeur » du terme *bit*.

**Norbert Wiener**, 1894 (Missouri) - 1964 (Suède). Mathématicien américain, un des pionniers de l'étude du bruit dans les télécommunications et systèmes de contrôle. Le mouvement brownien est aussi appelé « processus de Wiener ». Fonde la cybernétique (« science constituée par l'ensemble des théories sur les processus de commande et de communication et leur régulation chez l'être vivant, dans les machines et dans les systèmes sociologiques et économiques ») (le terme apparut à la suite d'un congrès d'analyse harmonique à Nancy). Citation : *Progress imposes not only new possibilities for the future but new restrictions.*



# Bibliographie

- [1] B. BURKE-HUBBARD : *Ondes et ondelettes : la saga d'un outil mathématique*. Belin, 1995. (un livre grand public sur la genèse de la théorie des ondelettes).
- [2] T. COVER et J. THOMAS : *Elements of Information Theory*. Wiley, 1991. (pour plus de renseignements sur la théorie de l'information et son application à la compression).
- [3] S. FABRE, Y. GOUSSEAU et J.-M. MOREL : Notes du cours d'analyse de première année. École Normale Supérieure de Cachan, 2003. (certains résultats et des idées d'expériences présentées ici viennent de ce polycopié).
- [4] C. GASQUET et P. WITOMSKI : *Analyse de Fourier et applications*. Masson, 1990. (pour toutes les démonstrations à propos de la théorie de Fourier que l'on n'a pas le temps d'approfondir).
- [5] A. HENROT : Analyse complexe, notes du cours de mathématiques. École des Mines de Nancy. (pour les rappels d'analyse complexe).
- [6] A. HENROT : Théorie des distributions, notes du cours de mathématiques. École des Mines de Nancy. (pour les rappels de théorie des distributions).
- [7] S. MALLAT : *Une exploration des signaux en ondelettes*. Éditions de l'École Polytechnique, 2000. (traitement du signal en général, et surtout analyse par ondelettes en particulier).
- [8] A. PAPOULIS : *Signal analysis*. McGraw-Hill, 1981. (pour des compléments - transformée en z, prolate spheroidal functions).
- [9] E. STADE : *Fourier analysis*. Wiley, 2005. ("written at a rigorous level, yet in an engaging style that does not dilute the material", pour compléter Gasquet-Witomski).
- [10] WIKIPEDIA.ORG : Certains articles sur l'encyclopédie en ligne wikipedia (version anglaise) sont de bons points d'entrée.

# Index

- Algorithme de Richardson-Lucy, 54
- Aliasing, 40, 83, 88
  - discret, 83
- Analyse temps-fréquence, 97
- Biographies, 105
- Bruit
  - blanc Gaussien, 47
  - poissonien, 54
- Chirp, 100
- Codage
  - arithmétique, 67
  - de Huffman, 65
  - LZW, 68
  - par plage, 67
  - RLE, 67
- Code préfixe, 60
- Coefficients de Fourier, 17
  - décroissance, 18, 71
  - unicité, 20
- Compression
  - avec perte, 71
  - sans perte, 59
- Condition de Nyquist discrète, 84
- Convolution
  - de fonctions périodiques, 21
  - de signaux numériques, 31
  - pour les signaux numériques, 47
- Déconvolution, 49
- Déconvolution directe, 50
- Dégradation linéaire, 47
- DCT, 75
- Détecteur de Canny, 25
- Divergence de Kullback-Leibler, 64
- Échantillonnage, 79, 85
- Effet de Gibbs, 19, 73
- Égalité de Parseval
  - pour les polynômes trigonométriques, 16
  - pour les signaux analogiques, 19
  - pour les signaux numériques, 30
- Entropie, 62
- Équation aux différences, 35
- Espace-échelle, 25
- Espaces  $L_p^1$  et  $L_p^2$ , 15
- Fenêtre Gaussienne, 98
- FFT, 31
- Filtre
  - analogique, 11
  - causal, 34
  - de Butterworth, 89
  - de Wiener, 51
  - FIR, IIR, 34
  - numérique, 34
  - passé-bas, 14
  - passé-bas idéal, 89
  - R,C, 12
  - stable, 34
- Flou
  - de bougé, 48
  - de mise au point, 48
  - Gaussien, 48
- Fonction de transfert, 12, 14
- Formule de Poisson, 80, 81

- Fréquence de Nyquist, 81, 88
- Inégalité  
  de Bessel, 18  
  de Gibbs, 63  
  de Jensen, 62  
  de Kraft, 61
- Invariance temporelle, 11
- JPEG, 77
- Lemme de Riemann-Lebesgue, 18
- MP3, 77
- Noyau de convolution, 48
- Openheim et Lim (expérience), 45
- Peigne de Dirac, 80
- Point Spread Function, 49
- Polynôme trigonométrique, 16
- Pré-écho, 71
- Principe d'incertitude, 97
- Quantification, 77
- Recouvrement de spectre, 40, 83
- Réponse impulsionnelle, 13, 34
- Restauration, 47
- Ringing, 43, 71, 86
- Scale-space, 25
- Série de Fourier, 19
- Signal  
  à bande limitée, 81  
  à énergie finie, 15  
  analogique, 14  
  monochromatique, 12  
  numérique, 27  
  trigonométrique, 15
- Sous-échantillonnage, 88
- Sur-échantillonnage, 85
- Théorème  
  d'échantillonnage de Shannon, 82, 88  
  d'incertitude, 97  
  de codage de Shannon, 64  
  de Jordan-Dirichlet, 20
- Transformée  
  de Fourier à court terme, 99  
  de Fourier à fenêtres, 99  
  de Fourier Discrète, 29  
  de Fourier Discrète 2D, 33  
  de Fourier Rapide, 31  
  discrète en cosinus, 75  
  en  $z$ , 35
- Zero padding, 85