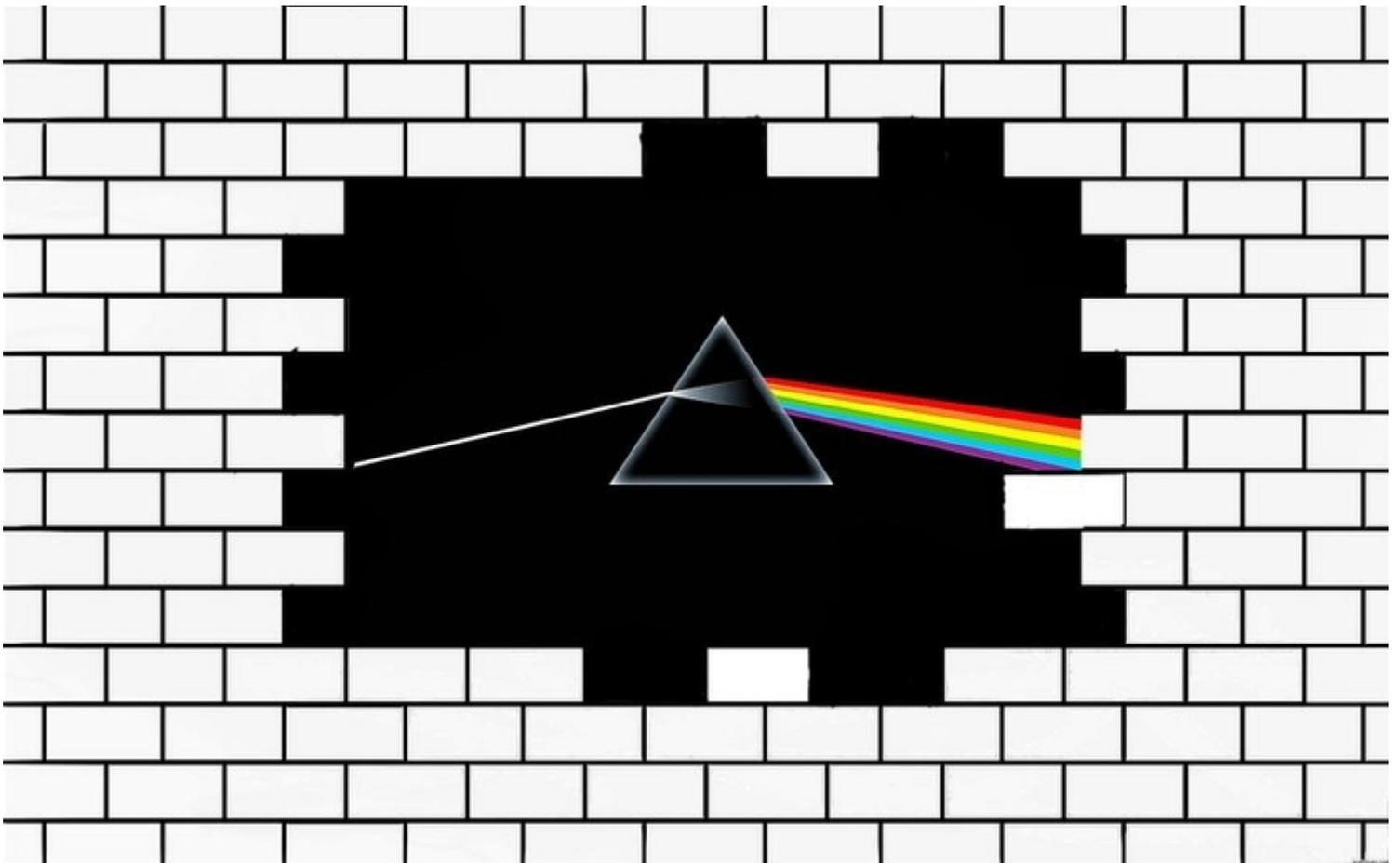


# Traitement numérique des signaux, stationnarité et non-stationnarité

Master Acoustique et Musicologie  
Aix-Marseille Université

Bruno Torrèsani

2021-22



Copyright © 2021 Bruno Torr sani

PUBLI  BY BRUNO TORR SANI  
INSTITUT DE MATH MATIQUES DE MARSEILLE  
AIX-MARSEILLE UNIVERSIT 

<https://www.i2m.univ-amu.fr/perso/bruno.torresani>

Licensed under the Creative Commons Attribution-NonCommercial 3.0 Unported License (the “License”). You may not use this file except in compliance with the License. You may obtain a copy of the License at <http://creativecommons.org/licenses/by-nc/3.0>. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*21 novembre 2021*



# Table des matières

<b>1</b>	<b>Introduction</b> .....	<b>5</b>
----------	---------------------------	----------

<b>I</b>	<b>Aspects théoriques</b>
----------	---------------------------

<b>2</b>	<b>Filtrage numérique</b> .....	<b>9</b>
<b>2.1</b>	<b>Rappels sur les signaux numériques</b>	<b>9</b>
2.1.1	Cadre théorique .....	9
2.1.2	Transformation de Fourier Discrète .....	11
2.1.3	Transformation en $z$ .....	15
<b>2.2</b>	<b>Filtrage numérique</b>	<b>18</b>
2.2.1	Définitions, propriétés importantes .....	18
2.2.2	Transformation de Fourier et filtrage .....	20
2.2.3	Transformation en $z$ et filtrage numérique .....	22
2.2.4	Factorisation de filtres .....	22
<b>2.3</b>	<b>Stationnarité, spectre, estimation spectrale</b>	<b>25</b>
2.3.1	Signaux stationnaires, spectre .....	26
2.3.2	Estimation spectrale .....	28
2.3.3	Estimation d'un modèle AR .....	31
2.3.4	Modèle ARMA .....	32
<b>3</b>	<b>Représentations temps-fréquence</b> .....	<b>35</b>
<b>3.1</b>	<b>Introduction : représentation fréquentielle locale</b>	<b>35</b>
<b>3.2</b>	<b>Transformation de Fourier à court terme (TFCT)</b>	<b>35</b>
3.2.1	TFCT pour des signaux de longueur infinie .....	36
3.2.2	TFCT en dimension finie .....	40

<b>3.3</b>	<b>Sous-échantillonner le plan temps-fréquence : transformation de Gabor</b>	<b>41</b>
<b>3.4</b>	<b>Filtrage temps-fréquence, multiplicateurs</b>	<b>45</b>
<b>3.5</b>	<b>Transformations temps-fréquence non-standard, ondelettes</b>	<b>48</b>
3.5.1	Transformation de Gabor non-stationnaire . . . . .	48
3.5.2	Largeur de bande variable en fréquence, transformée à $Q$ constant, ondelettes . . . . .	50
<b>4</b>	<b>Stationnarité "par morceaux" . . . . .</b>	<b>53</b>
<b>4.1</b>	<b>Retour sur le calcul matriciel</b>	<b>53</b>
<b>4.2</b>	<b>Algorithmes pour la factorisation d'un spectrogramme</b>	<b>54</b>
4.2.1	Généralités . . . . .	54
4.2.2	Factorisation en matrices non-négatives . . . . .	55
<b>4.3</b>	<b>Un exemple</b>	<b>56</b>

## II

## Exercices, Travaux pratiques

<b>5</b>	<b>Travaux Pratiques : TP 1 . . . . .</b>	<b>65</b>
<b>5.1</b>	<b>Filtrage numérique</b>	<b>65</b>
5.1.1	Filtrage dans le domaine fréquentiel . . . . .	65
5.1.2	Filtrage dans le domaine temporel . . . . .	66
<b>5.2</b>	<b>Stationnarité, spectre, estimation spectrale</b>	<b>67</b>
5.2.1	Estimation spectrale . . . . .	67
5.2.2	Estimation d'un modèle AR . . . . .	67
<b>6</b>	<b>Travaux Pratiques : TP 2 . . . . .</b>	<b>69</b>
<b>6.1</b>	<b>Transformation de Gabor</b>	<b>69</b>
<b>6.2</b>	<b>Filtrage d'un bruit blanc</b>	<b>70</b>
6.2.1	Filtrage "passe-bande" . . . . .	70
6.2.2	Filtrage non-stationnaire . . . . .	70
<b>6.3</b>	<b>Filtrage de signaux musicaux</b>	<b>70</b>
<b>6.4</b>	<b>Factorisation de spectrogramme</b>	<b>70</b>
<b>7</b>	<b>Correction des exercices . . . . .</b>	<b>73</b>
<b>7.1</b>	<b>Corrections des exercices du chapitre 2</b>	<b>73</b>
<b>7.2</b>	<b>Corrections des exercices du chapitre 3</b>	<b>79</b>

## III

## Compléments

<b>Bibliographie . . . . .</b>	<b>85</b>
<b>Index . . . . .</b>	<b>87</b>



# 1. Introduction

Le traitement numérique du signal étudie les techniques de représentation et analyse, d'interprétation et de traitement (filtrage, compression, restauration...) des signaux numérisés. À la différence du traitement des signaux analogiques qui est réalisé par des dispositifs d'électronique analogique, le traitement des signaux numériques est réalisé par des machines numériques (des ordinateurs ou des circuits dédiés). On pourra trouver un descriptif introductif dans l'article dédié de l'encyclopédie en ligne Wikipedia [9]. On pourra aussi se référer à des livres de référence tels que [1, 8], ou encore au cours [7] disponible en ligne, ainsi qu'aux ressources de J.O. Smith III plus spécifiquement orientées audio (voir notamment [4] et [5]).

L'objectif de ce cours est de donner dans un premier temps les éléments de base de traitement des signaux numériques.

On y parlera de transformation de Fourier, transformation en  $z$ , filtrage numérique, on approfondira notamment certains aspects de la construction de filtres numériques satisfaisant certaines propriétés importantes.

On verra ensuite en détails les notions de stationnarité et non-stationnarité pour les signaux, et on analysera en détails quelques méthodes d'analyse de signaux non-stationnaires appelées représentations temps-fréquence, dont on verra quelques exemples d'application, par exemple en transcription automatique.

Dans une dernière partie, on se focalisera plus particulièrement sur la synthèse temps-fréquence, et la transformation de signaux numériques non-stationnaires.

Le cours se composera de séances théoriques (cours et exercices), et de séances de travaux pratiques sous matlab/octave.





# Aspects théoriques

<b>2</b>	<b>Filtrage numérique</b> .....	<b>9</b>
2.1	Rappels sur les signaux numériques	
2.2	Filtrage numérique	
2.3	Stationnarité, spectre, estimation spectrale	
<b>3</b>	<b>Représentations temps-fréquence</b> ....	<b>35</b>
3.1	Introduction : représentation fréquentielle locale	
3.2	Transformation de Fourier à court terme (TFCT)	
3.3	Sous-échantillonner le plan temps-fréquence : transformation de Gabor	
3.4	Filtrage temps-fréquence, multiplicateurs	
3.5	Transformations temps-fréquence non-standard, ondelettes	
<b>4</b>	<b>Stationnarité "par morceaux"</b> .....	<b>53</b>
4.1	Retour sur le calcul matriciel	
4.2	Algorithmes pour la factorisation d'un spectrogramme	
4.3	Un exemple	





## 2. Filtrage numérique

Ce chapitre est principalement consacré au filtrage numérique, que nous allons voir sous deux aspects. Après quelques rappels et notions de base sur les signaux numériques, nous aborderons le filtrage sous l'aspect "modification du contenu fréquentiel d'un signal". Nous aborderons ensuite les modèles de signaux stationnaires, pour lesquels le filtrage sera utilisé comme outil de synthèse.

### 2.1 Rappels sur les signaux numériques

On appellera signaux numériques des signaux à temps discret (par opposition aux signaux analogiques), à valeurs réelles ou complexes (on ne considérera pas le problème de la quantification). Dans cette section, on commence par rappeler le cadre théorique dans lequel on va travailler (espace de signaux), puis la transformation de Fourier discrète et la transformation en  $z$ .

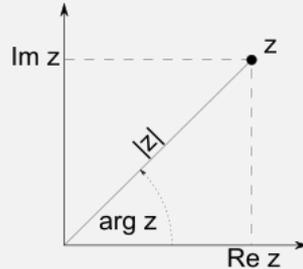
#### 2.1.1 Cadre théorique

On ne reviendra pas ici sur le problème d'échantillonnage, on traitera ici uniquement des signaux déjà numérisés, de la forme  $x[\ell] = x(\ell/\eta)$ ,  $\eta \in \mathbb{R}_+^*$  étant la fréquence d'échantillonnage, et  $x$  étant un signal analogique (i.e. à temps continu), issu d'un filtrage passe-bas permettant son échantillonnage à la fréquence  $\eta$ . Les échantillons  $x[\ell]$  sont donc indexés par un entier  $\ell$ . Un signal numérique est donc une collection d'échantillons  $x[\ell]$ , que l'on modélise sous forme de vecteurs, de dimension finie ou infinie. Cette modélisation permet d'exploiter le langage des espaces vectoriels : étant donné deux signaux  $x$  et  $y$ , et deux nombres  $a$  et  $b$ , le vecteur  $ax + by$  définit un nouveau signal dont les échantillons sont égaux à  $ax[\ell] + by[\ell]$ . Les nombres  $a$  et  $b$  peuvent être des nombres réels ou complexes.

**Remarque 2.1 — Bref rappel sur les nombres complexes.** Les signaux d'intérêt (surtout audio) sont généralement des signaux réels. Par contre nous verrons en étudiant la transformation de Fourier que des signaux complexes s'introduisent assez naturellement. On rappelle très brièvement ici quelques éléments sur les nombres complexes.

- Les nombres complexes sont utilisés pour représenter des couples de nombres réels :  $(a, b) \in \mathbb{R}^2 \rightarrow z = a + ib$ . Le nombre  $i$  est tel que  $i^2 = -1$ .
- La raison fondamentale pour utiliser des nombres complexes est qu'ils permettent de donner des solutions que n'en ont pas (ou pas suffisamment) si on se limite à des réels. En particulier, *Tout polynôme de degré  $N$  admet exactement  $N$  racines complexes.*

- Pour  $z \in \mathbb{C}$ , le nombre (réel positif ou nul)  $|z| = \sqrt{a^2 + b^2}$  est le *module* de  $z$  (qui est aussi la longueur du vecteur  $(a, b)$  du plan).
- On écrit aussi sous forme polaire  $z = |z|e^{i\theta}$ . Le nombre  $\theta \in ]-\pi, \pi]$  est l'*argument* de  $z$ .  $a = |z| \cos(\theta)$  est la *partie réelle* de  $z$ .  $b = |z| \sin(\theta)$  est la *partie imaginaire* de  $z$ .
- Le conjugué (ou complexe conjugué)  $\bar{z}$  de  $z = a + ib \in \mathbb{C}$  est le nombre  $\bar{z} = a - ib = |z|e^{-i\theta}$ , symétrique de  $z$  par rapport à l'axe réel dans le plan complexe.



Le nombre complexe  $z = |z|(\cos(\theta) + i \sin(\theta))$  dans le plan complexe

- Dans le cas de signaux de longueur finie, on supposera  $\ell \in \{0, 1, \dots, L-1\}$ ,  $L$  étant le nombre d'échantillons. Le cadre algébrique naturel est alors l'espace Hermitien  $\mathbb{C}^L$ , muni du produit scalaire

$$\langle x, y \rangle_{\mathbb{C}^L} = \sum_{\ell=0}^{L-1} x[\ell] \bar{y}[\ell], \quad x, y \in \mathbb{C}^L. \quad (2.1)$$

et de la norme associée  $\|x\| = \sqrt{\langle x, x \rangle_{\mathbb{C}^L}}$ . Le théorème de Cauchy-Schwarz fait le lien entre produit scalaire et norme

$$|\langle x, y \rangle_{\mathbb{C}^L}| \leq \|x\| \|y\|. \quad (2.2)$$

On représente généralement ces signaux sous forme de vecteurs "colonne", le produit scalaire s'écrit alors sous la forme d'un produit matriciel

$$\langle x, y \rangle = y^* x, \quad \text{où} \quad y^* = \bar{y}^T.$$

- Dans un cadre théorique, on travaillera souvent dans le cadre de signaux de longueur infinie, i.e.  $\ell \in \mathbb{Z}$ , on sera amenés dans ce cas à utiliser des modèles mathématiques d'espace de signaux. Un cadre naturel est celui de l'espace  $\ell^2(\mathbb{Z})$  des signaux  $\{x[\ell], \ell \in \mathbb{Z}\}$  tels que

$$\sum_{\ell=-\infty}^{\infty} |x[\ell]|^2 < \infty,$$

qui est muni du produit scalaire

$$\langle x, y \rangle_{\ell^2(\mathbb{Z})} = \sum_{\ell=-\infty}^{\infty} x[\ell] \bar{y}[\ell], \quad x, y \in \ell^2(\mathbb{Z}), \quad (2.3)$$

et de la norme  $x \in \ell^2(\mathbb{Z}) \rightarrow \|x\| = \sqrt{\langle x, x \rangle_{\ell^2(\mathbb{Z})}}$  correspondante. L'inégalité de Cauchy-Schwarz (2.2) reste valide, ce qui démontre l'existence du produit scalaire  $\langle x, y \rangle_{\ell^2(\mathbb{Z})}$  dès que  $x, y \in \ell^2(\mathbb{Z})$ .

Plus généralement, on utilise parfois les espaces  $\ell^p(\mathbb{Z})$  des signaux  $\{x[\ell], \ell \in \mathbb{Z}\}$  tels que  $\sum_{\ell=-\infty}^{\infty} |x[\ell]|^p < \infty$ , munis de la norme  $x \rightarrow \|x\|_p = [\sum_{\ell} |x[\ell]|^p]^{1/p}$  et de l'espace  $\ell^\infty(\mathbb{Z})$  des signaux  $x$  bornés, c'est à dire tels qu'il existe une constante notée  $\|x\|_\infty$  telle que  $|x[\ell]| \leq \|x\|_\infty$  pour tout  $\ell$ . L'application  $x \in \ell^\infty(\mathbb{Z}) \rightarrow \|x\|_\infty$  est une norme sur  $\ell^\infty(\mathbb{Z})$ .

Dans les calculs numériques, les signaux considérés seront toujours de longueur finie. On les considère souvent comme des restrictions à  $\{0, \dots, N-1\}$  de signaux de longueur infinie périodiques de période  $N$ .

**Remarque 2.2** Nous utiliserons plus loin la transformation de Fourier discrète (TFD). La TFD d'un signal numérique (de longueur infinie) est une fonction d'une variable réelle, périodique de période 1, que l'on peut identifier à sa restriction à un intervalle de longueur 1, l'intervalle  $[-1/2, 1/2]$  en l'occurrence. On sera donc amené à considérer l'espace  $L^2([-1/2, 1/2])$  des fonctions définies sur  $[-1/2, 1/2]$  de module carré intégrable. Cet espace est muni d'une structure d'espace de Hilbert par le produit scalaire

$$\langle f, g \rangle_{L^2} = \int_{-1/2}^{1/2} f(v) \bar{g}(v) dv, \quad f, g \in L^2([-1/2, 1/2]), \quad (2.4)$$

duquel dérive la norme  $\|f\| = \sqrt{\langle f, f \rangle_{L^2}}$



Il existe différentes façons de coder le produit scalaire sous matlab/octave. Si les vecteurs  $x, y \in \mathbb{C}^N$  sont représentés par des vecteurs colonne  $\mathbf{x}, \mathbf{y}$ , leur produit scalaire (au sens où nous l'avons défini)  $\langle x, y \rangle$  peut s'écrire indifféremment comme  $\mathbf{y}' * \mathbf{x}$  (on rappelle que  $\mathbf{y}'$  est le conjugué Hermitien de  $\mathbf{y}$ , c'est à dire le complexe conjugué du transposé de  $\mathbf{y}$ ) ou encore  $\text{dot}(\mathbf{y}, \mathbf{x})$ . La norme  $\|x\|$  s'obtient via l'instruction  $\text{norm}(\mathbf{x})$ , alors que la norme plus générale  $\|x\|_p$  s'obtient par  $\text{norm}(\mathbf{x}, p)$ .

## 2.1.2 Transformation de Fourier Discrète

### Signaux de longueur infinie

Soit  $x = \{x[n], n \in \mathbb{Z}\}$  un signal numérique. Sa transformée de Fourier Discrète (TFD) est la fonction périodique de période 1  $v \rightarrow \hat{x}(v)$  définie par

$$\hat{x}(v) = \sum_{\ell=-\infty}^{\infty} x[\ell] e^{-2i\pi v \ell}, \quad (2.5)$$

pour tout  $v$  tel que la série soit convergente. La variable  $v$  est appelée "fréquence", et la variable  $\omega = 2\pi v$  est appelée pulsation.

Si on se place dans le cadre des signaux de  $\ell^2(\mathbb{Z})$ , la TFD est inversible. Plus précisément, on a

**Théorème 2.3** La transformation de Fourier discrète est une transformation unitaire de  $\ell^2(\mathbb{Z})$  sur  $L^2([-1/2, 1/2])$  : pour tout  $x \in \ell^2(\mathbb{Z})$ ,  $\hat{x} \in L^2([-1/2, 1/2])$ , et la formule de Parseval

$$\int_{-1/2}^{1/2} |\hat{x}(v)|^2 dv = \sum_{\ell=-\infty}^{\infty} |x[\ell]|^2 \quad (2.6)$$

est vérifiée. La transformation inverse est donnée par

$$x[\ell] = \int_{-1/2}^{1/2} \hat{x}(v) e^{2i\pi v \ell} dv. \quad (2.7)$$

**Remarque 2.4 — Prise en compte de la fréquence d'échantillonnage.** La définition ci-dessus utilise des fréquences normalisées (entre -1/2 et 1/2), ce qui revient implicitement à supposer une fréquence d'échantillonnage égale à 1. Si on désire prendre en compte une fréquence d'échantillonnage  $\eta$ , on pourra définir la transformation de Fourier discrète de  $x \in \ell^2(\mathbb{Z})$  sous la forme

$$\hat{x}(v) = \sum_{\ell=-\infty}^{\infty} x[\ell] e^{-2i\pi v \ell / \eta}, \quad v \in [-\eta/2, \eta/2].$$

La formule de Parseval et la transformation de Fourier inverse prennent alors la forme

$$\frac{1}{\eta} \int_{-\eta/2}^{\eta/2} |\hat{x}(v)|^2 dv = \sum_{\ell=-\infty}^{\infty} |x[\ell]|^2, \quad x[\ell] = \frac{1}{\eta} \int_{-\eta/2}^{\eta/2} \hat{x}(v) e^{2i\pi v \ell / \eta} dv.$$

**Remarque 2.5 — Autres formulation.** On trouve parfois des formulations de la TFD utilisant la variable de pulsation  $\omega$  au lieu de la variable de fréquence. On définit alors

$$\hat{x}(\omega) = \sum_{\ell=-\infty}^{\infty} x[\ell] e^{-i\omega \ell}, \quad \omega \in [-\pi, \pi].$$

La formule de Parseval et la TFD inverse s'écrivent alors

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} |\hat{x}(\omega)|^2 d\omega = \sum_{\ell=-\infty}^{\infty} |x[\ell]|^2, \quad x[\ell] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \hat{x}(\omega) e^{i\omega \ell} d\omega.$$

Il existe d'autres variantes, qui sont toutes équivalentes.

La transformation de Fourier discrète est une transformation linéaire : étant donnés deux signaux  $x, y$  et leur TFD respective  $\hat{x}$  et  $\hat{y}$ , et deux nombres  $a, b$ , la TFD de  $ax + by$  est  $a\hat{x} + b\hat{y}$ . Les autres propriétés simples sont listées ci-dessous sous forme d'exercices.

**Exercice 2.1 — Propriétés de base de la TFD.** 1. Soit  $x \in \ell^1(\mathbb{Z})$  un signal numérique, soit  $\hat{x}$  sa TFD. Soit  $y$  une copie translatée de  $x$ , de la forme  $y[\ell] = x[\ell - \tau]$ . Exprimer  $\hat{y}$  en fonction de  $\hat{x}$ .  
2. Démontrer la formule de Parseval et la formule de reconstruction ci-dessus. On pourra remarquer que

$$\int_{-1/2}^{1/2} e^{2i\pi v \ell} dv = \delta_{\ell,0} \quad \forall \ell \in \mathbb{Z}$$

et exploiter cette remarque dans la démonstration. ■

**Exercice 2.2 — Sous-échantillonnage.** 1. Étant donné un signal numérique  $x$ , on note  $y$  le signal obtenu par sous-échantillonnage d'un facteur 2 :  $y[\ell] = x[2\ell]$ , pour  $\ell \in \mathbb{Z}$ . Démontrer que

$$\hat{y}(v) = \frac{1}{2} [\hat{x}(v) + \hat{x}(v + 1/2)]$$

(suggestion : développer le membre de droite de cette égalité). En notant maintenant  $y$  le signal obtenu par l'autre sous-échantillonnage de facteur 1/2, cette fois de la forme  $y[\ell] = x[2\ell + 1]$ , exprimer  $\hat{y}$  en fonction de  $\hat{x}$ .

2. Soit  $z$  obtenu par sous-échantillonnage d'un facteur 3 :  $z[\ell] = x[3\ell]$ , exprimer  $\hat{z}$  en fonction de  $\hat{x}$ . ■

**Exercice 2.3 — Sur-échantillonnage.** On s'intéresse maintenant au sur-échantillonnage d'un facteur 2 : étant donné un signal numérique  $x$ , soit  $y$  défini par

$$y[\ell] = \begin{cases} x[\ell/2] & \text{si } \ell \text{ est pair} \\ 0 & \text{sinon} \end{cases}$$

Exprimer  $\hat{y}$  en fonction de  $\hat{x}$ . ■

**Exercice 2.4 — Fenêtrage.** On travaille en pratique sur des signaux de longueur finie. On peut voir un signal de longueur finie  $y \in \mathbb{C}^L$  comme le produit d'un signal de longueur infinie  $x$  par une "fenêtre", par exemple l'indicatrice  $I_L$  de l'intervalle entier  $\{0, \dots, L-1\}$ , qui vaut 1 sur cet intervalle et 0 en dehors.

1. Montrer que la TFD de ce produit  $y = x.I_L$  est de la forme

$$\hat{y}(v) = \int_{-1/2}^{1/2} \hat{x}(\xi) \gamma(v - \xi) dv$$

et expliciter la fonction  $\gamma$ . On pourra tracer le module de la fonction  $\gamma$  pour se faire une idée plus précise de l'effet de la fenêtre. Suggestion : utiliser la formule donnant la somme partielle d'une série géométrique

$$\sum_{\ell=0}^{L-1} r^\ell = \begin{cases} \frac{1-r^L}{1-r} & \text{si } r \neq 1 \\ L & \text{si } r = 1 \end{cases} .$$

2. Plus généralement, si on remplace cette indicatrice par une fenêtre quelconque  $h$ , montrer que la TFD du produit  $y = h.x$  s'exprime encore sous la même forme, et expliciter la fonction  $\gamma$  correspondante. ■

### Signaux de longueur finie

Il existe une version de la transformation de Fourier adaptée aux signaux de longueur finie, qu'on appellera *Transformation de Fourier Finie* (TFF). En se plaçant dans l'espace  $\mathbb{C}^L$ , on définit la TFF d'un signal  $x \in \mathbb{C}^L$  comme le vecteur  $\hat{x} \in \mathbb{C}^L$  défini par

$$\hat{x}[k] = \frac{1}{\sqrt{L}} \sum_{\ell=0}^{L-1} x[\ell] e^{-2i\pi k\ell/L} . \quad (2.8)$$

Comme la TFD, la TFF est une transformation linéaire. Il est possible de démontrer qu'elle est inversible, avec une TFF inverse définie par

$$x[\ell] = \frac{1}{\sqrt{L}} \sum_{k=0}^{L-1} \hat{x}[k] e^{2i\pi k\ell/L} , \quad (2.9)$$

et une formule de Parseval (conservation de la norme)

$$\|\hat{x}\|^2 = \sum_{k=0}^{L-1} |\hat{x}[k]|^2 = \sum_{\ell=0}^{L-1} |x[\ell]|^2 = \|x\|^2 . \quad (2.10)$$

**Remarque 2.6 — Symétries de la TFF.** — Pour  $x \in \mathbb{C}^L$ ,  $\hat{x} \in \mathbb{C}^L$  peut en fait être considéré comme la restriction à  $\{0, \dots, L-1\}$  d'un vecteur infini périodique de période  $L$ . En effet, on voit facilement que pour tout  $k$ ,  $\hat{x}[k+L] = \hat{x}[k]$ .

— Si  $x$  est un signal réel, alors  $\hat{x}$  possède la symétrie Hermitienne :

$$\hat{x}[-k] = \hat{x}[L-k] = \overline{\hat{x}[k]} .$$

Dans les représentations graphiques, il est souvent plus agréable de représenter la transformée de Fourier d'un signal pour un domaine de fréquences symétrique, centré sur la fréquence 0. De plus, si ce signal est réel, alors il est inutile de représenter les fréquences négatives, puisqu'elles sont complètement déterminées par les fréquences positives.

**Exercice 2.5 — Propriétés de la TFF.** Démontrer les propriétés données dans les équations (2.9) et (2.10). On pourra utiliser la formule donnant la somme partielle d'une série géométrique donnée dans l'exercice 2.4. ■

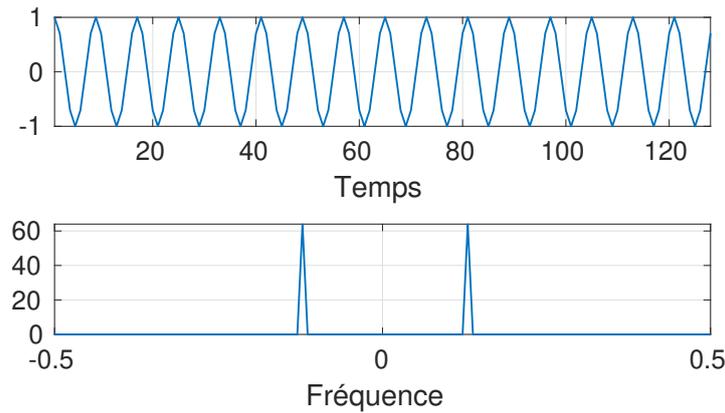


FIGURE 2.1 – Haut : cosinus de longueur 128 (échantillonné à 1Hz) ; bas : transformée de Fourier (sur un domaine de fréquences symétrique par rapport à l'origine)



La TFF est implémentée sous matlab par la fonction `fft`, la TFF inverse par la fonction `ifft` (avec une normalisation différente, voir l'aide en ligne). La séquence d'instruction

```
>> L = 128;
>> f0 = 16;
>> x = cos(2*pi*f0*[0:(L-1)]/L); x = x';
>> xhat = fft(x);
```

calcule la transformée de Fourier discrète d'une sinusoïde de fréquence  $f_0=16$ .  $x$  est ici un vecteur réel (un vecteur colonne, grâce à l'instruction `x=x'`), et  $\hat{x}$  aussi, mais comme  $\hat{x}$  est évalué numériquement, les erreurs d'arrondi produisent une petite partie imaginaire (on pourra le vérifier), de sorte que pour représenter  $\hat{x}$  il faut en prendre la partie réelle. Pour représenter sur un axe des fréquences symétrique par rapport à l'origine, on peut utiliser la fonction `fftshift` :

```
>> fff = linspace(-1/2,1/2,L);
>> plot(fff,fftshift(real(xhat)));
>> axis tight; grid on; xlabel('Fréquence');
```

Le résultat se trouve en Figure 2.1. On voit bien apparaître les deux fréquences  $\pm f_0$  présentes dans ce cas via la formule

$$\cos(2\pi f_0 \ell) = \frac{1}{2} \left[ e^{2i\pi f_0 \ell} + e^{-2i\pi f_0 \ell} \right].$$

■ **Exemple 2.1 — Son de clarinette.** On a tracé en Figure 2.2 trois représentations différentes du spectre du son d'une note de clarinette. En haut le résultat de la FFT, au milieu la même chose, où l'axe des fréquences est centré sur la fréquence nulle (on a utilisé la périodicité de la TFF), et en bas on a uniquement représenté les fréquences positives, et utilisé une échelle logarithmique pour avoir une vue plus synthétique. On peut remarquer sur le tracé du milieu la symétrie Hermitienne du spectre, ainsi que sa structure harmonique (les raies se présentent à des fréquences multiples de la fréquence fondamentale).

Notons que dans cette figure l'axe des abscisses donne les valeurs physiques de la fréquence, ce qui revient à dire que nous avons utilisé la version de la TFD décrite dans la remarque 2.4 qui prend en compte la fréquence d'échantillonnage.

■ **Exercice 2.6 — Sous-échantillonnage et sur-échantillonnage sous Matlab.** En prenant un signal simple (sinusoïde par exemple), illustrer les propriétés de la TFD liées au sur-échantillonnage et au sous-échantillonnage (exercices 2.2 et 2.3). ■

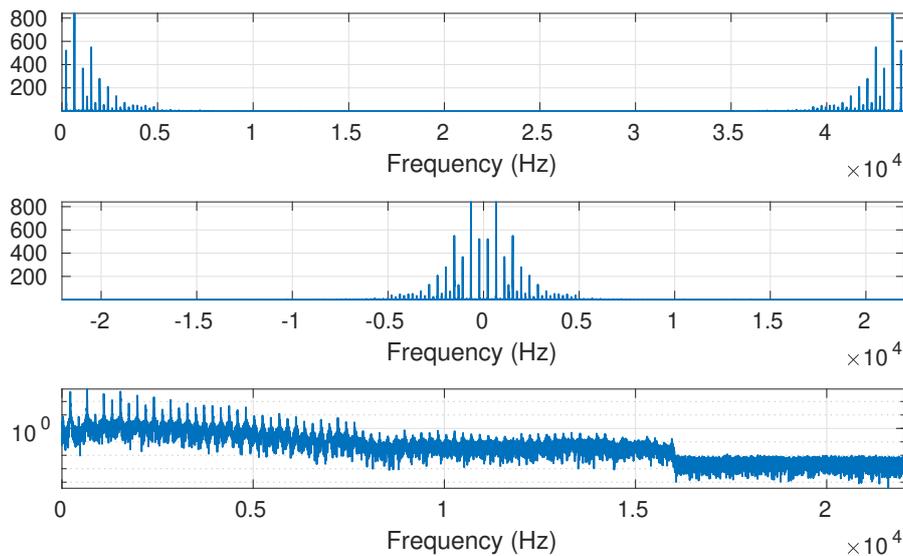


FIGURE 2.2 – Spectre d'un son de note tenue de clarinette

### 2.1.3 Transformation en $z$

On se place ici dans le cas de signaux de longueur infinie. Un signal est dit causal si ses échantillons d'indice négatif sont nuls :  $x[n] = 0 \forall n < 0$ .

La transformation en  $z$  associe à une suite une fonction d'une variable complexe  $z$ . Elle peut être vue comme un prolongement de la TFD dans le plan complexe, et ses propriétés en font un outil très utilisé en traitement du signal.

#### Séries de Laurent, transformation en $z$

Étant donné un signal numérique  $\{x[\ell], \ell \in \mathbb{Z}\}$ , il existe des cas où sa transformée de Fourier discrète n'est pas définie au sens classique. La transformée en  $z$  est alors une alternative utile. On décrit ci-dessous les propriétés essentielles, sans entrer dans les détails.

**Définition 2.1** Soit  $x = \{x[\ell], \ell \in \mathbb{Z}\}$  un signal numérique. Sa transformée en  $z$  est la série de Laurent

$$X(z) = \sum_{\ell=-\infty}^{\infty} x[\ell]z^{-\ell}, \quad (2.11)$$

définie dans une couronne de convergence (éventuellement vide)  $r_1 < |z| < r_2$ .

On sait d'après des résultats généraux sur les séries de Laurent que  $X$  est holomorphe dans sa couronne de convergence. Inversement, étant donnée une fonction  $X$  holomorphe dans une couronne  $r_1 < |z| < r_2$ , elle admet un unique développement en série de Laurent. De plus, on a le résultat classique suivant :

**Proposition 2.7** Le rayon de convergence  $\rho$  de la série entière  $z \rightarrow \sum_0^{\infty} a[\ell]z^{\ell}$  est donné par

$$\frac{1}{\rho} = \limsup_{\ell \rightarrow \infty} |a[\ell]|^{1/\ell} = \limsup_{\ell \rightarrow \infty} \left| \frac{a[\ell]}{a[\ell+1]} \right|.$$

Le premier critère est le critère de Cauchy, le second est le critère de d'Alembert.

On en déduit immédiatement la couronne de convergence de la transformée en  $z$  d'un signal numérique :

**Corollaire 2.8** Soit  $X$  la transformée en  $z$  de la série  $x$ . Les bornes de la couronne de convergence de  $X$  sont données par

$$r_1 = \limsup_{\ell \rightarrow \infty} |x[\ell]|^{1/\ell}, \quad \frac{1}{r_2} = \limsup_{\ell \rightarrow \infty} |x[-\ell]|^{1/\ell}. \quad (2.12)$$

■ **Exemple 2.2** On rappelle qu'un signal numérique  $x$  est *causal* si  $x[\ell] = 0$  pour tout  $\ell < 0$ . Inversement,  $x$  est dit *anticausal* si  $x[\ell] = 0$  pour tout  $\ell \geq 0$ . Supposons que  $x$  soit causal. Alors il est évident que  $r_2^{-1} = 0$ , de sorte que la transformée en  $z$  de  $x$  est bien définie dans le domaine  $|z| > r_1$ , c'est à dire à l'extérieur d'un cercle de rayon  $r_1$ .

De même, si  $s$  est anticausal,  $r_1 = 0$ , et  $X(z)$  est bien défini à l'intérieur du cercle de rayon  $r_2$ . ■

**Exercice 2.7 — Propriétés simples de la transformation en  $z$ .** 1. Soit  $x$  un signal numérique, soit  $X$  sa transformée en  $z$ . Soit  $y$  une copie translatée de  $x$  de la forme  $y[\ell] = x[\ell - \tau]$ . Exprimer la transformée en  $z$  de  $y$  en fonction de la transformée en  $z$  de  $x$ .  
2. Étant donnés deux signaux numériques  $x$  et  $h$  leur produit de convolution est le signal numérique  $h * x$  défini par

$$(h * x)[\ell] = \sum_{k=-\infty}^{\infty} h[k]x[\ell - k].$$

Exprimer la transformée en  $z$  de  $h * x$  en fonction de  $H$  et  $X$ . En déduire l'expression de la TFD de  $h * x$ . ■

**Exercice 2.8 — Transformation en  $z$  et sur/sous-échantillonnage.** Soit  $x$  un signal numérique, soit  $X$  sa transformée en  $z$ .

1. On note  $y$  le signal obtenu par sous-échantillonnage d'un facteur 2 :  $y[\ell] = x[2\ell]$ , pour  $\ell \in \mathbb{Z}$ . Démontrer que

$$Y(z) = \frac{1}{2} \left[ X(z^{1/2}) + X(-z^{1/2}) \right].$$

En notant maintenant  $y$  le signal obtenu par l'autre sous-échantillonnage de facteur 1/2, cette fois de la forme  $y[\ell] = x[2\ell + 1]$ , exprimer  $Y$  en fonction de  $X$ .

2. Soit  $z$  obtenu par sous-échantillonnage d'un facteur 3 :  $z[\ell] = x[3\ell]$ , exprimer  $Z$  en fonction de  $X$ . On s'intéresse maintenant au sur-échantillonnage d'un facteur  $K$  : étant donné un signal numérique  $x$ , soit  $y$  défini par

$$y[\ell] = \begin{cases} x[\ell/K] & \text{si } \ell \text{ est multiple de } K \\ 0 & \text{sinon} \end{cases}$$

Montrer que  $Y(z) = X(z^K)$ . ■

### Inversion de la transformation en $z$

Il existe plusieurs techniques permettant d'inverser une transformation en  $z$ . La plus simple consiste à expliciter un développement en série de Laurent de la fonction d'une variable complexe  $z \rightarrow X(z)$  considérée. Le développement en série de Laurent étant unique, ceci fournit directement une transformée inverse.

Les deux exemples qui suivent montrent l'importance du domaine de définition d'une transformée en  $z$ .

■ **Exemple 2.3** Prenons l'exemple de la fonction

$$X(z) = \frac{z}{z - z_0}, \quad |z| < |z_0|;$$

on peut alors écrire, pour  $|z| < |z_0|$ ,

$$X(z) = \frac{z}{z - z_0} = -\frac{z}{z_0} \frac{1}{1 - z/z_0} = -\frac{z}{z_0} \sum_{\ell=0}^{\infty} \left(\frac{z}{z_0}\right)^{\ell} = -\sum_{\ell=-\infty}^{-1} z_0^{\ell} z^{-\ell},$$

ce qui, conjugué à l'unicité du développement en série de Laurent, fournit

$$x[\ell] = \begin{cases} -z_0^{\ell} & \text{pour } \ell < 0 \\ 0 & \text{sinon.} \end{cases}$$

$x$  est donc un signal anticausal. ■

■ **Exemple 2.4** Reprenons l'exemple de la fonction précédente, cette fois définie dans l'autre domaine (l'extérieur du disque de rayon  $|z_0|$  dans le plan complexe)

$$X(z) = \frac{z}{z - z_0}, \quad |z| > |z_0|;$$

on peut alors écrire, pour  $|z| > |z_0|$ ,

$$X(z) = \frac{z}{z - z_0} = \frac{1}{1 - z_0/z} = \sum_{\ell=0}^{\infty} \left(\frac{z_0}{z}\right)^{\ell} = \sum_{\ell=0}^{\infty} z_0^{\ell} z^{-\ell},$$

ce qui, conjugué à l'unicité du développement en série de Laurent, fournit

$$x[\ell] = \begin{cases} z_0^{\ell} & \text{pour } \ell \geq 0 \\ 0 & \text{sinon.} \end{cases}$$

ce qui définit cette fois un signal causal. ■

**Exercice 2.9 — Une transformée en  $z$  simple.** Soit  $X$  la fonction d'une variable complexe définie par

$$X(z) = \frac{1}{z - z_0}, \quad |z| \leq |z_0|,$$

où  $z_0 = \rho e^{2i\pi\nu_0} \in \mathbb{C}$  avec  $\rho < 1$  et  $\nu_0 \in [-1/2, 1/2]$ .

1. Calculer  $|X(z)|$  (on pourra poser  $z = re^{2i\pi\nu}$ ), et tracer  $|X(e^{2i\pi\nu})|$  en fonction de  $\nu \in [-1/2, 1/2]$ .
2. Déterminer le signal numérique  $x$  dont  $X$  est la transformée en  $z$ .

**Exercice 2.10 — Inversion de la transformation en  $z$ .** Soit  $X$  la fonction d'une variable complexe définie par

$$X(z) = \frac{1}{(z-1)(z-2i)}, \quad 1 < |z| < 2.$$

Décomposer  $X(z)$  en éléments simples, et trouver le signal numérique  $x$  dont  $X$  est la transformée en  $z$ . ■

Une alternative consiste à utiliser la TFD. Soit  $X$  la transformée en  $z$  d'un signal  $x$ , et soit  $r$  un nombre tel que  $r_1 < r < r_2$ . Calculons

$$\int_{-1/2}^{1/2} X(re^{2i\pi\theta}) e^{2i\pi n\theta} d\theta = \sum_m x[m] r^{-m} \int_{-1/2}^{1/2} e^{2i(n-m)\pi\theta} d\theta = r^{-n} x[n].$$

On peut donc écrire

$$x[n] = r^n \int_{-1/2}^{1/2} X(re^{2i\pi\theta}) e^{2i\pi n\theta} d\theta.$$

Par un changement de variables complexes  $z = re^{2i\pi\theta}$ , on obtient donc

**Proposition 2.9** Soit  $x$  un signal numérique, et soit  $X$  sa transformée en  $z$ , définie dans la couronne de convergence  $r_1 < |z| < r_2$ . Les coefficients de  $s$  sont donnés par

$$x[\ell] = \frac{1}{2i\pi} \oint_C X(z) z^\ell \frac{dz}{z}, \quad (2.13)$$

où  $C$  est un cercle centré sur l'origine du plan complexe, de rayon  $r \in ]r_1, r_2[$ .

On a généralement recours à la méthode des résidus<sup>1</sup> pour calculer de telles intégrales.

**Remarque 2.10 — Transformation en  $z$  et transformation de Fourier discrète.** Il est utile de pouvoir interpréter la transformation en  $z$ , qui est un objet assez abstrait, en la reliant à la TFD et au concept plus physique de fréquence. Pour cela, revenons à la version de la TFD dans laquelle la fréquence d'échantillonnage est prise en compte, décrite dans la remarque 2.4 : en notant  $\eta$  la , c'est à dire en supposant que les échantillons  $x[\ell] = x(\ell/\eta)$  proviennent d'un échantillonnage à fréquence d'échantillonnage  $\eta$  d'un signal à temps continu  $x(t)$ , nous avons défini

$$\hat{x}(v) = \sum_{\ell=-\infty}^{\infty} x[\ell] e^{-2i\pi v \ell / \eta}, \quad v \in [-\eta/2, \eta/2].$$

On voit donc ici que  $\hat{x}(v)$  est la restriction de la transformée en  $z$  de  $x$  au cercle unité du plan complexe

$$\hat{x}(v) = X\left(e^{2i\pi v / \eta}\right).$$

Ainsi, l'argument de  $z$  n'est autre que la pulsation  $\arg(z) = \omega = 2\pi v$ , ce qui donne une signification physique à  $z$ . Dans le plan complexe, parcourir le cercle unité équivaut à parcourir l'intervalle  $[-\eta/2, \eta/2]$  des fréquences.

## 2.2 Filtrage numérique

Les opérations de filtrage sont les opérations de base du traitement du signal. Comme on va le voir plus loin, le filtrage est utilisé pour modifier le "contenu fréquentiel" des signaux.

### 2.2.1 Définitions, propriétés importantes

**Définition 2.2** Un filtre numérique est un opérateur linéaire noté  $K_h$ , associant à un signal numérique  $x$  un autre signal  $K_h x$ , appelé signal filtré, de la forme

$$(K_h x)[\ell] = \sum_{k=-\infty}^{\infty} h[k] x[\ell - k], \quad (2.14)$$

pour tout  $\ell$  tel que la série soit convergente. La suite  $h = \{h[\ell], \ell \in \mathbb{Z}\}$  est appelée *réponse impulsionnelle* du filtre.

Le filtre est dit *causal* si  $h[\ell] = 0$  pour tout  $\ell < 0$ . Il est dit *stable* si  $K_h f \in \ell^\infty(\mathbb{Z})$  est borné pour tout  $f \in \ell^\infty(\mathbb{Z})$ . Il est *réalisable* s'il est causal et stable.

**Remarque 2.11** Il est facile de montrer qu'un filtre numérique est covariant par translations : étant donné un signal numérique  $x$ , si on note  $y$  un translaté de  $x$  de la forme :

$$y[\ell] = x[\ell - \ell_0], \quad \ell \in \mathbb{Z}$$

pour tout entier  $\ell_0$ , il vient immédiatement

$$(K_h y)[\ell] = (K_h x)[\ell - \ell_0].$$

1. Un exposé simple peut être trouvé sur le site de Serge Mehl : [http://serge.mehl.free.fr/anx/th\\_residus.html](http://serge.mehl.free.fr/anx/th_residus.html)

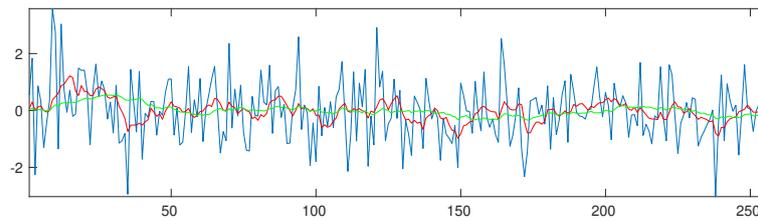


FIGURE 2.3 – Deux filtrages de type 'lissage' d'un bruit blanc gaussien (bleu), à deux résolutions différentes (rouge :  $L = 8$ ; vert :  $L = 32$ ).

**Exercice 2.11 — Covariance des filtres numériques.** Démontrer la propriété de covariance des filtres numériques énoncée dans la Remarque 2.11. ■

Lorsque la réponse impulsionnelle  $h$  du filtre a un nombre fini de coefficients non-nuls, on dit que le filtre est à réponse impulsionnelle finie (filtre FIR). Dans le cas contraire, il est à réponse impulsionnelle infinie (filtre IIR).

■ **Exemple 2.5 — Filtrage d'un bruit blanc.** Un bruit blanc est une suite de nombres aléatoires indépendants identiquement distribués (par exemple suivant une loi Gaussienne). On peut effectuer un filtrage passe-bas par un simple lissage, en remplaçant chaque échantillon par la moyenne de cet échantillon et des  $L - 1$  échantillons précédents :

$$y[\ell] = \frac{1}{L} \sum_{k=0}^{L-1} x[\ell - k].$$

Cette opération est bien un filtrage FIR au sens précédent, qui élimine (ou tout du moins réduit) les variations à une échelle inférieure à  $L$ . La réponse impulsionnelle est de longueur  $L$ , constante égale  $= 1/L$ . ■



La fonction `filter` implémente (notamment) le filtrage par un filtre FIR. La séquence d'instructions ci-dessous génère une réalisation de longueur 256 d'un bruit blanc Gaussien (fonction `randn`), une réponse impulsionnelle finie constante sur son support (de longueur 8, puis 32) et effectue le filtrage correspondant. Celui-ci s'assimile à un lissage du signal initial à des échelles de 8 et 32 échantillons respectivement.

```
>> x = randn(256,1);
>> L = 8;
>> h = ones(L,1)/L;
>> y8 = filter(h,1,x);
>> L = 32;
>> h = ones(L,1)/L;
>> y32 = filter(h,1,x);
>> plot(x); axis tight
>> hold on
>> plot(y8,'r');
>> plot(y32,'g')
```

Le résultat est représenté en Figure 2.3. On peut noter que la courbe verte est plus lisse que la rouge, elle même plus lisse que la bleue. On voit bien l'effet du lissage.

**Remarque 2.12** On n'a abordé dans cette section que le cas des signaux de longueur infinie. Dans le cas de signaux de longueur finie  $L$ , deux définitions sont possibles :

— Étendre le signal par des zéros (on parle de *zero padding*), ce qui revient à définir le filtrage sous la forme suivante :

$$K_h x[\ell] = \sum_{k=\ell-L}^{L-1} h[k] x[\ell - k]$$

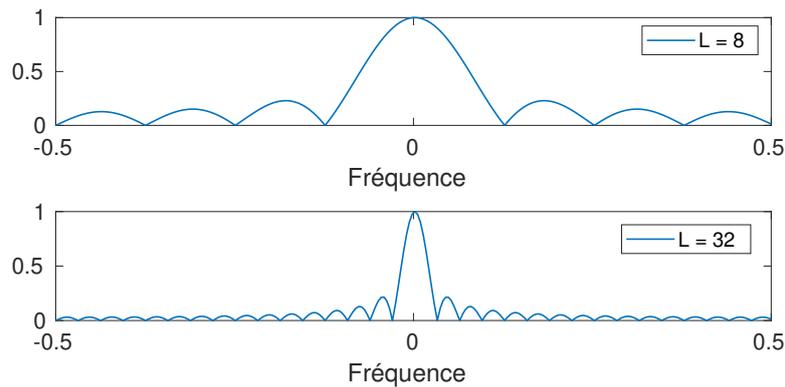


FIGURE 2.4 – Fonctions de transfert des deux filtres de lissage de l'exemple 2.5.

— Considérer des conditions aux bords périodiques, c'est à dire considérer toutes les opérations modulo  $L$ .

### 2.2.2 Transformation de Fourier et filtrage

Une propriété remarquable de la transformation de Fourier discrète est qu'elle transforme filtrage en multiplication point par point. En effet, étant donné un filtre de réponse impulsionnelle  $h$ , son action s'écrit dans le domaine de Fourier

$$\widehat{K_h x}(v) = \hat{h}(v)\hat{x}(v), \quad v \in [-1/2, 1/2]. \quad (2.15)$$

La fonction

$$m(v) = \hat{h}(v)$$

est appelée fonction de transfert, ou impédance, ou réponse fréquentielle du filtre. On voit donc que l'effet d'un filtre est bien de modifier le contenu fréquentiel des signaux, par multiplication par cette fonction de transfert. L'action d'un filtre  $T = K_h$  de fonction de transfert  $m = \hat{h}$  s'écrit donc aussi sous la forme

$$Tx[\ell] = \int_{-1/2}^{1/2} m(v)\hat{x}(v)e^{2i\pi v\ell} dv. \quad (2.16)$$

Les fonctions de transfert des deux filtres de lissage décrits dans l'exemple 2.5 sont donnés dans la figure 2.4. On voit bien que la fonction de transfert est proche de 1 pour des fréquences proches de 0, puis décroît, plus rapidement pour  $L = 32$  que pour  $L = 8$ . On peut donc dire que ce filtre a tendance à préserver les basses fréquences et atténuer les hautes fréquences (on parle de filtre passe-bas). Par contre on note la présence de "rebonds", qui font que ce comportement n'est pas si clair. De tels lissages ne sont pas capables de supprimer (ou atténuer fortement) des domaines fréquentiels donnés, tout en en préservant d'autres.

Concrètement, on aimerait pouvoir disposer de filtres (par exemple filtres passe-bas) idéaux, au sens où leur fonction de transfert soit l'indicatrice d'un intervalle symétrique par rapport à l'origine.

**Exercice 2.12 — Calcul d'une fonction de transfert.** Dans l'exemple 2.5, la réponse impulsionnelle du filtre est donnée par

$$h[\ell] = \frac{1}{L} \quad \text{pour } \ell = 0, \dots, L-1 \quad \text{et} \quad h[\ell] = 0 \quad \text{pour } \ell = L, \dots, N-1.$$

Calculer la fonction de transfert de ce filtre, puis son module au carré, et interpréter dans les cas  $N = 256$ , et  $L = 8$  et  $L = 32$  en comparant à la figure 2.4. ■

**Exercice 2.13 — Calcul d'une réponse impulsionnelle.** En utilisant la transformation de Fourier inverse, calculer la réponse impulsionnelle d'un filtre passe-bas idéal, de fonction de transfert

$$m(\nu) = \begin{cases} 1 & \text{si } |\nu| \leq \nu_0 \\ 0 & \text{sinon} \end{cases}$$

où  $\nu_0 \in ]0, 1/2[$  est la fréquence de coupure.

Vérifier que ce filtre est à réponse impulsionnelle infinie et n'est pas causal. ■

On voit bien d'après ce qui précède que la fonction de transfert d'un filtre FIR causal de réponse impulsionnelle  $\{h[0], \dots, h[L]\}$  est en fait un polynôme trigonométrique (c'est à dire un polynôme de la variable complexe  $e^{-2i\pi\nu}$ ) de degré  $L$  :

$$\hat{h}(\nu) = \sum_{\ell=0}^L h[\ell] e^{-2i\pi\nu\ell}.$$

Or il est connu que de tels polynômes trigonométriques peuvent difficilement approximer des fonctions discontinues (c'est le phénomène de Gibbs). On doit alors recourir à des filtres à réponse impulsionnelle infinie. Il existe une famille de tels filtres, appelée famille de filtres récurrents, dont l'utilisation ne fait toutefois intervenir qu'un nombre fini d'opérations. Ces filtres associent à un signal d'entrée  $x$  un signal de sortie  $y$  solution de l'équation

$$\sum_{k=0}^K a_k y[n-k] = \sum_{\ell=0}^L b_\ell x[n-\ell], \quad \forall n \in \mathbb{Z}, \quad (2.17)$$

où les nombres  $a_k$  et  $b_\ell$  sont des coefficients qui caractérisent le filtre, avec en particulier  $a_0 = 1$ . Cette équation s'écrit aussi sous la forme

$$y[n] = \sum_{\ell=0}^L b_\ell x[n-\ell] - \sum_{k=1}^K a_k y[n-k],$$

qui fait bien apparaître la nature causale du filtre. Sous certaines conditions sur les coefficients  $a_k$  et  $b_\ell$ , il est possible de montrer que l'opérateur  $T : x \rightarrow y$  solution de l'équation (2.17) est bien un filtre stable et causal, donc la fonction de transfert est la fonction rationnelle trigonométrique

$$m(\nu) = \frac{\sum_{\ell=0}^L b_\ell e^{-2i\pi\nu\ell}}{\sum_{k=0}^K a_k e^{-2i\pi\nu k}}, \quad (2.18)$$

c'est à dire le quotient de deux polynômes trigonométriques de degrés respectifs  $L$  et  $K$ . L'étude de telles fonctions de transfert repose généralement sur la transformation en  $z$ , que nous allons voir plus loin.

La fonction de transfert  $m(z)$  d'un filtre est à valeurs complexes, et n'est pas toujours facile à interpréter. On s'intéresse alors tout d'abord à son module au carré. On en voit un exemple ci-dessous.

■ **Exemple 2.6 — Filtre passe-bas de Butterworth d'ordre 1.** On considère le filtre dont le module carré de la fonction de transfert est donné par

$$|m(\nu)|^2 = \frac{1 + \cos(2\pi\nu)}{(1 + \cos(2\pi\nu)) + k(1 - \cos(2\pi\nu))},$$

où  $k$  est une constante. On voit en particulier que  $|m(0)|^2 = 1$  et  $|m(\pm 1/2)|^2 = 0$ . Ce filtre est un filtre passe-bas, qui préserve les basses fréquences (les fréquences proches de 0) et atténue les hautes fréquences (les fréquences proches de  $\pm 1/2$ ). Le module carré de la fonction de transfert est représenté en Figure 2.5.

Le rôle du paramètre  $k$  est de déterminer le domaine des fréquences qui seront atténuées. ■

**Exercice 2.14 — Filtre passe-bas de Butterworth.** On considère le Filtre passe-bas de Butterworth d'ordre 1 de l'exemple 2.6. Soit  $0 < \lambda < 1$ . Déterminer la valeur de  $k$  telle que  $|m(\nu_0)|^2 = \lambda$ , pour une certaine fréquence  $0 < \nu_0 < 1/2$ . Cas particulier,  $\nu_0 = 1/4$ ,  $\lambda = 10^{-6}$ . ■

### 2.2.3 Transformation en $z$ et filtrage numérique

L'un des intérêts de la transformation en  $z$  est son comportement vis à vis des transformations simples, et en particulier des translations. Étant donné un signal  $\{x[n], n \in \mathbb{Z}\}$ , et un signal translaté  $\{x'[n], n \in \mathbb{Z}\}$  donné par  $x'[n] = x[n-k]$ , on voit immédiatement que leurs transformées en  $z$  sont reliées par  $X'(z) = z^k X(z)$ . Le corollaire immédiat est le comportement de la transformation en  $z$  vis à vis du filtrage numérique. Étant donné un signal numérique  $x$  et un filtre numérique de réponse impulsionnelle  $h$ , et  $y = K_h x$  le signal filtré, alors pour tout  $z$  à l'intérieur de l'intersection des couronnes de convergence des transformées  $X$  et  $H$  de  $x$  et  $h$  respectivement, on a

$$Y(z) = \sum_n y[n] z^{-n} = \sum_n \sum_k h[k] z^{-k} x[n-k] z^{-(n-k)},$$

de sorte que l'on a

$$Y(z) = H(z)X(z). \quad (2.19)$$

La fonction  $H$  est elle aussi appelée *fonction de transfert* du filtre.

En particulier, dans le cas d'un filtre récursif comme précédemment, on a

$$Y(z) = \frac{\sum_{\ell=0}^L b_{\ell} z^{-\ell}}{\sum_{k=0}^K a_k z^{-k}} X(z), \quad (2.20)$$

c'est à dire que la transformée en  $z$  de  $h$ , qui est aussi la fonction de transfert du filtre, prend la forme d'une fraction rationnelle

$$H(z) = \frac{\sum_{\ell=0}^L b_{\ell} z^{-\ell}}{\sum_{k=0}^K a_k z^{-k}}. \quad (2.21)$$

Cette expression est à rapprocher de l'expression (2.18) obtenue avec la TFD. En fait (2.18) est le cas particulier de (2.21) quand on limite la variable  $z$  à rester sur le cercle unité dans le plan complexe.

### 2.2.4 Factorisation de filtres

L'expression (2.21) de la fonction de transfert d'un filtre récursif fait apparaître deux polynômes, au numérateur et au dénominateur. On sait que tout polynôme d'une variable complexe est scindé, et peut donc être totalement factorisé (et caractérisé par ses racines, à une constante multiplicative près). On peut donc écrire

$$H(z) = g \frac{(1 - z_1 z^{-1})(1 - z_2 z^{-1}) \dots (1 - z_L z^{-1})}{(1 - \zeta_1 z^{-1})(1 - \zeta_2 z^{-1}) \dots (1 - \zeta_K z^{-1})} = g z^{K-L} \frac{(z - z_1)(z - z_2) \dots (z - z_L)}{(z - \zeta_1)(z - \zeta_2) \dots (z - \zeta_K)}, \quad (2.22)$$

où  $g = H(0)$  est appelé gain du filtre.

C'est cette factorisation que nous allons maintenant exploiter.

**Définition 2.3** On dit qu'un filtre causal  $K_h$  est d'ordre fini si  $K_h$  peut être réalisé comme un filtre récursif comme en (2.17).

L'équation (2.22) montre que de tels filtres d'ordre fini peuvent être caractérisés par les pôles et les zéros (les racines du numérateur) de leur fonction de transfert  $H(z)$ . On parle de **représentation pôles-zéros**.

Comme on l'a vu plus haut, la fonction de transfert elle-même est souvent difficilement interprétable, son module carré est plus facile à appréhender. En fait, la construction d'un filtre repose souvent sur la construction préalable d'un module carré de fonction de transfert, dont on extrait une "racine module carré". Le résultat qui suit montre que ceci est possible, et produit sous certaines conditions des filtres stables et causaux.

**Proposition 2.13** 1. Soit  $K_h$  un filtre causal stable d'ordre fini. Alors le module carré de sa fonction de transfert  $M = |\hat{h}|^2$  est une fonction rationnelle non-négative de la variable réelle  $w = \cos(2\pi v) \in [-1, 1]$ .  
2. Inversement, étant donnée une fonction rationnelle non-négative de  $w = \cos(2\pi v)$  notée  $v \rightarrow M(v) \in \mathbb{R}^+$ , il existe un filtre causal stable d'ordre fini  $K_h$  dont le module carré de la fonction de transfert coïncide avec  $M$ .

Preuve : Le filtre  $K_h$  est supposé causal, donc la couronne de convergence de  $H(z)$  est l'extérieur d'un disque, de la forme  $|z| > r_1$ . Il est aussi supposé stable, donc le cercle unité est inclus dans la couronne de convergence. Par conséquent, les pôles de  $H$  se trouvent à l'intérieur du cercle unité.

On se limite ici aux filtres réels, c'est à dire tels que les coefficients  $a_k$  et  $b_\ell$  sont réels. Dans ce cas, il est facile de voir que

$$\overline{\hat{h}(v)} = \hat{h}(-v) ,$$

de sorte que le spectre prend la forme

$$|\hat{h}(v)|^2 = H(z)H(z^{-1}) , \quad \text{avec } z = e^{2i\pi v} .$$

Notons  $z_k$  les pôles de  $H$  (les racines du dénominateur de  $H$ ) et  $\zeta_\ell$  les zéros de  $H$  (les racines du numérateur de  $H$ ). Il est immédiat que  $|\hat{h}(v)|^2$  est caractérisé par des facteurs de la forme

$$(z - z_k)(z^{-1} - \overline{z_k}) = 1 + z_k^2 - z_k(z + z^{-1}) , \quad \text{et} \quad (z - \zeta_k)(z^{-1} - \overline{\zeta_k}) = 1 + \zeta_k^2 - \zeta_k(z + z^{-1}) ,$$

La module carré de la fonction de transfert  $M(v) = |\hat{h}(v)|^2$  est donc une fonction rationnelle positive de la variable réelle

$$w = \frac{1}{2}(z + z^{-1}) = \cos(2\pi v) \in [-1, 1] , \quad \text{où } z = e^{2i\pi v} .$$

Inversement, soit  $v \rightarrow W(\cos(2\pi v)) = N(\cos(2\pi v))/D(\cos(2\pi v))$  une fonction rationnelle positive de la variable réelle  $w = \cos(2\pi v) \in [-1, 1]$ . Notons  $w_k$  les racines complexes de  $D$  (le numérateur  $N$  se traite de façon identique). On voit facilement que l'équation en  $z$

$$w_k = \frac{1}{2}(z + z^{-1})$$

possède deux solutions inverses l'une de l'autre, que l'on note  $z_k$  et  $z_k^{-1}$ . Par convention, on choisit  $|z_k| < 1$ . On peut alors poser

$$d(z) = \prod_k (z - z_k) .$$

De même, en notant  $\gamma_k$  les racines de  $N$ , et  $\zeta_k$  et  $\zeta_k^{-1}$  les solutions de l'équation d'inconnue  $\zeta$

$$\gamma_k = \frac{1}{2}(\zeta + \zeta^{-1})$$

(sans nécessairement imposer  $|\zeta_k| < 1$ ), on est naturellement conduit à introduire

$$n(z) = \prod_k (z - \zeta_k) .$$

Il résulte de cette analyse que la fonction

$$z \rightarrow \frac{n(z)}{d(z)}$$

est la fonction de transfert d'un filtre causal stable d'ordre fini. Ceci conclut la preuve. ♠

**Remarque 2.14** Le filtre  $K_h$  n'est pas unique, car il reste la liberté de choisir les zéros  $\zeta_k$  à l'intérieur ou à l'extérieur du disque unité pour former la fonction  $d$ . Choisir tous les zéros à l'intérieur du disque unité conduit aux filtres dits à *phase minimale*.

Un exemple est donné par la famille des *filtres de Butterworth* (dans leur version numérique), déjà rencontrés plus haut dans un cas simple, qui constituent des approximations de filtres passe-bas idéaux. En introduisant de nouveau la variable  $w = \cos(2\pi v)$ , les filtres de Butterworth sont caractérisés par une fonction de transfert telle que

$$|\hat{h}(v)|^2 = W(w) = \frac{(w+1)^L}{(w+1)^L + k(1-w)^L} , \quad (2.23)$$

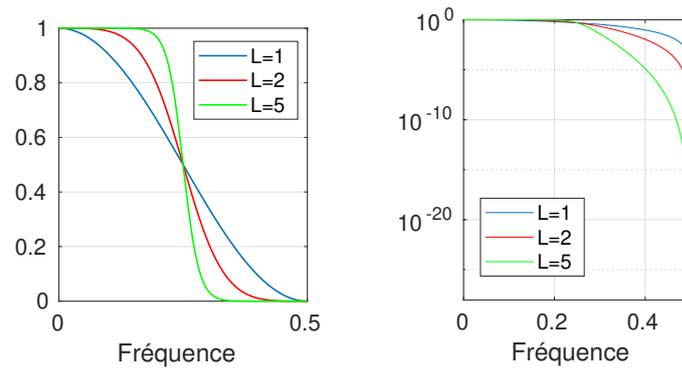


FIGURE 2.5 – Module carré de la fonction de transfert de trois filtres (passe-bas) de Butterworth d'ordres  $L = 1$ ,  $L = 2$  et  $L = 5$ .

où  $L$  est un entier positif fixé, et  $k$  une constante positive, qui détermine la fréquence de transition entre bande passante et bande atténuée, via la relation

$$k = 3 \left( \frac{1 + \cos(2\pi v_0)}{1 - \cos(2\pi v_0)} \right)^L,$$

$v_0$  étant la fréquence à laquelle la réponse en amplitude vaut  $1/2$ . Par exemple,  $k = 3$  correspond à une fréquence de coupure égale à  $v_0 = 1/4$ .

**Exercice 2.15 — Filtres de Butterworth.** 1. On considère un filtre passe-bas de Butterworth dont le module carré de la fonction de transfert est donné en (2.23), avec  $L = 1$ . Déterminer les pôles de la fonction de transfert du filtre stable causal correspondant (on les exprimera en fonction de  $k$ ).

2. Même question pour  $L = 2$ .



La fonction `butter` génère les coefficients de filtres de Butterworth d'ordre et fréquence de coupure donnés. Pour générer un filtre de Butterworth d'ordre  $L$  et fréquence de coupure normalisée  $F_n$ , la syntaxe est la suivante.

```
>> [b,a] = butter(L,Fn);
```

Par exemple, la séquence d'instructions

```
>> [b,a] = butter(10,1/2);
>> [H,W] = freqz(b,a);
>> plot(linspace(0,1/2,length(H)),abs(H).^2);
>> xlabel('Fréquence');
>> ylabel('Magnitude');
```

génère un filtre de Butterworth d'ordre 10, de fréquence de coupure égale à  $1/4$ . La fonction `freqz` permet de générer la réponse en fréquence et la réponse en phase à partir des coefficients du filtr. Un exemple se trouve en Figure 2.6, où est représentée le module carré de la fonction de transfert du filtre généré dans cet exemple, en échelle linéaire et en échelle logarithmique. Utilisée sans variable de sortie, la fonction `freqz` représente graphiquement ces deux réponses (fréquentielle et en phase).

La séquence d'instructions

```
>> [z,p,k] = butter(10,1/4);
>> zplane(z,p);
```

génère les zéros ( $z$ ), les pôles ( $p$ ) et le gain de la représentation, puis donne une représentation des zéros et des pôles dans le plan complexe. Le résultat se trouve en Figure 2.7 (figure de droite). On y voit apparaître un zéro d'ordre  $L = 10$  à la fréquence  $v = \pm 1/2$ , et des pôles à diverses localisations à l'intérieur du cercle unité. On peut voir que ces pôles sont localisés sur des points d'argument compris entre  $-\pi/4$  et  $\pi/4$ , correspondant à des fréquences comprises entre  $-1/8$  et  $1/8$ .

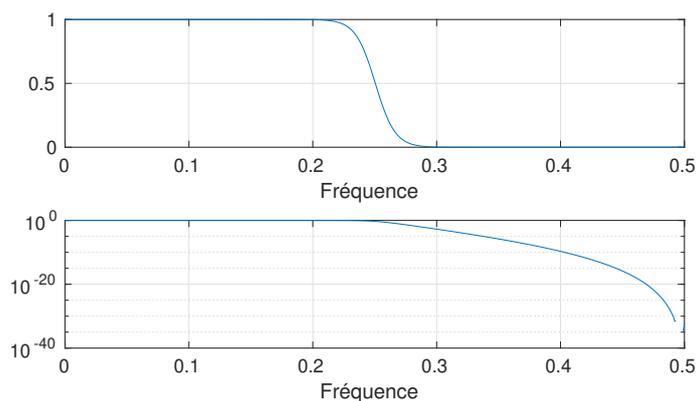


FIGURE 2.6 – Filtre de Butterworth d'ordre 10 et de fréquence de coupure normalisée 1/2 : module carré de la fonction de transfert, échelle linéaire (haut) et logarithmique (bas).

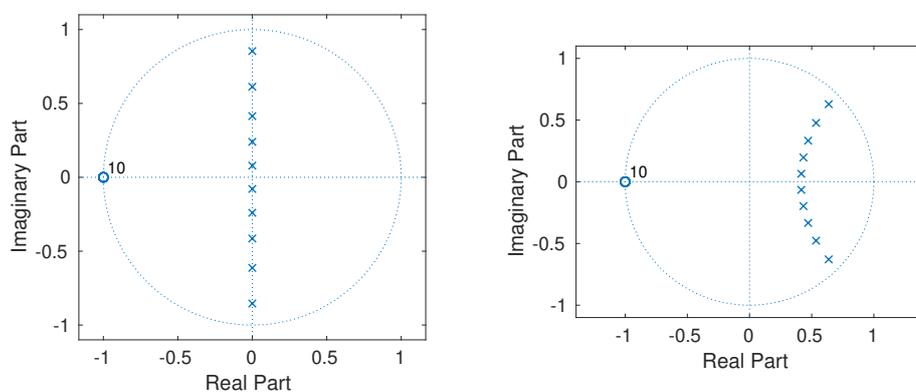


FIGURE 2.7 – Filtre de Butterworth d'ordre 10 et de fréquence de coupure normalisée 1/2 (gauche) et 1/4 (droite) : représentation des zéros et des pôles dans le plan complexe.



La *Signal Processing Toolbox* de matlab fournit de nombreux outils de construction de filtres numériques, et divers outils de visualisation. Outre la fonction `butter` qu'on vient d'étudier (voir l'aide pour les différentes syntaxes), on mentionnera les fonctions `cheby1` et `cheby2` qui implémentent deux types de filtres de Chebyshev (se renseigner) et plus généralement la fonction `fdesign`. La fonction `designfilt` fournit une interface graphique à `fdesign`.

## 2.3 Stationnarité, spectre, estimation spectrale

On entend généralement par spectre d'un signal une mesure de son "contenu fréquentiel". Des exemples issus de deux instruments de musique jouant une note isolée sont décrits en Figure 2.8. Ceci suppose que ce dernier ne varie pas au cours du temps (au moins dans un intervalle suffisamment long). Cette hypothèse est une hypothèse de stationnarité, qu'il faut tout d'abord définir correctement.

On parle souvent de signal stationnaire, ce qui est en un certain sens un abus de langage, car un signal est le résultat d'une mesure. Il est plus correct de définir un signal stationnaire comme une mesure d'un signal provenant d'un système stationnaire (par exemple un moteur de voiture à vitesse constante, une note de musique tenue,...).

### 2.3.1 Signaux stationnaires, spectre

Mathématiquement, la stationnarité d'un signal fait référence au cadre probabiliste : un signal est modélisé comme une réalisation (ou un tirage) d'un signal aléatoire, c'est à dire un signal dont les échantillons sont des variables aléatoires qui suivent une certaine loi de probabilités. La stationnarité fait référence à une propriété d'invariance de la distribution de probabilités : étant donné un signal  $x$ ,  $x$  est stationnaire (au sens strict) si toute copie  $y$  de  $x$  translatée en temps ( $y[n] = x[n - \tau]$  pour un certain entier  $\tau$ ) a la même distribution de probabilités que  $x$ . Cette définition est souvent trop restrictive, on se contente de la notion de stationnarité au sens faible (on parle parfois de stationnarité en moyenne quadratique), qui se définit comme une propriété de stationnarité restreinte aux moyennes d'ordre 1 et 2.

Dans ce qui suit, on notera  $\mathbb{E}\{x\}$  l'espérance mathématique (ou moyenne) d'une variable aléatoire  $x$ .

**Définition 2.4 — Signal aléatoire stationnaire.** 1. Un signal aléatoire  $x$  est du second ordre si chaque échantillon (aléatoire) possède une variance finie :

$$\mathbb{E}\{|x[\ell]|^2\} < \infty \quad \forall \ell \in \mathbb{Z}.$$

2. Un signal aléatoire du second ordre est stationnaire en moyenne quadratique (on dit aussi faiblement stationnaire, ou stationnaire au sens faible) si

$$\mathbb{E}\{x[\ell]\} = \mathbb{E}\{x[0]\} \quad \forall \ell \in \mathbb{Z}, \quad \mathbb{E}\{x[\ell]\bar{x}[k]\} = \mathbb{E}\{x[\ell - k]\bar{x}[0]\} \quad \forall k, \ell \in \mathbb{Z}.$$

Dans ces conditions, si  $x$  est un signal aléatoire du second ordre stationnaire en moyenne quadratique, on définit sa moyenne  $\mu$  et son autocovariance  $\mathcal{C}$  par

$$\mu = \mathbb{E}\{x[0]\}, \quad \mathcal{C}[\ell] = \mathbb{E}\{x[\ell]\bar{x}[0]\}. \quad (2.24)$$

**Remarque 2.15 — Stationnarité.** La stationnarité est donc une propriété d'invariance par translation. Cette propriété est importante car elle permet d'estimer des quantités moyennes, en remplaçant l'espérance mathématique (qui est un objet théorique) par des moyennes temporelles. Par exemple, étant donnée une réalisation de longueur  $L$  d'un signal aléatoire  $x$  (par abus de notation, on note aussi  $x$  la réalisation), la moyenne temporelle

$$\mu_L = \frac{1}{L} \sum_{\ell=0}^{L-1} x[\ell]$$

est utilisée comme estimation de la moyenne probabiliste  $\mu$ . De même, on pourra utiliser l'autocovariance empirique

$$\mathcal{C}_L[\ell] = \frac{1}{L - \ell} \sum_{k=\ell}^{L-1} x[k]\bar{x}[k - \ell].$$

Clairement, plus  $L$  sera grand plus l'estimation sera précise. Ceci est vrai en particulier pour la moyenne  $\mu$ . Dans le cas de l'autocovariance, on peut remarquer que l'estimation de  $\mathcal{C}_L[\ell]$  est d'autant moins précise que  $\ell$  est grand, car la moyenne est prise sur un nombre moindre d'échantillon.

On peut associer aux signaux aléatoires stationnaires leur spectre, qui donne leur contenu fréquentiel moyen. L'existence du spectre est donnée par le théorème de Wiener-Khinchin

**Théorème 2.16 — Wiener-Khinchin.** Soit  $x$  un signal aléatoire stationnaire au sens faible, qu'on suppose de moyenne nulle. Alors la fonction d'autocovariance  $\mathcal{C}$  peut s'écrire comme la TFD inverse d'une fonction à valeurs positives ou nulles  $\mathcal{S}$  appelée densité spectrale de  $x$ , ou plus simplement spectre :

$$\mathcal{C}[\ell] = \int_{-1/2}^{1/2} \mathcal{S}(v) e^{2i\pi v \ell} dv.$$

Le spectre est unique.

**Remarque 2.17** Si  $\mathcal{C} \in \ell^1(\mathbb{Z})$ , il est possible de démontrer que le spectre  $\mathcal{S}$  est la transformée de Fourier discrète de  $\mathcal{C}$  :

$$\mathcal{S}(v) = \sum_{\ell=-\infty}^{\infty} \mathcal{C}[\ell] e^{-2i\pi v \ell} .$$

■ **Exemple 2.7 — Bruit blanc Gaussien.** Le cas le plus simple est celui du bruit blanc Gaussien, pour lequel les échantillons  $x[\ell]$  sont modélisés comme des tirages d'une variable aléatoire gaussienne, de moyenne  $\mu$  et de variance  $\sigma^2$  fixées. Il est possible de démontrer que dans ce cas le spectre est la fonction constante

$$\mathcal{S}(v) = \sigma^2 , \quad v \in [-1/2, 1/2] .$$

■ **Exemple 2.8 — Bruit Gaussien coloré.** Si on filtre un bruit blanc Gaussien  $x$  avec un filtre de fonction de transfert  $m = \hat{h}$ , on obtient de nouveau (sous des hypothèses peu contraignantes sur  $h$  qu'on ne mentionnera pas ici) un signal faiblement stationnaire  $y = K_h x$ , pour lequel les échantillons  $y[\ell]$  sont maintenant des tirages d'une nouvelle variable aléatoire gaussienne, mais sont maintenant corrélées. Il est possible de montrer que dans ce cas, le spectre est la fonction

$$\mathcal{S}(v) = \sigma^2 |m(v)|^2 , \quad v \in [-1/2, 1/2] .$$

**Exercice 2.16 — Spectre du bruit blanc et d'un bruit coloré.** 1. Dans le cas d'un bruit blanc, la fonction d'autocovariance est donnée par  $\mathcal{C}[\ell] = \sigma^2 \delta_{\ell,0}$ . En déduire le spectre du bruit blanc donné dans l'exemple 2.7.  
2. Dans le cas d'un bruit coloré, exprimer la fonction d'autocovariance en fonction de la réponse impulsionnelle du filtre. En déduire le spectre du bruit coloré donné dans l'exemple 2.8.



La fonction `randn` (voir la syntaxe) permet de générer des nombres pseudo-aléatoires indépendants et identiquement distribués (tout du moins le mieux qu'un programme déterministe puisse faire), de moyenne nulle et variance égale à 1, donc une réalisation d'un bruit blanc Gaussien. Par exemple

```
>> L = 256;
>> x = randn(L,1);
```

génère un vecteur colonne contenant une réalisation de longueur 256 d'un bruit blanc gaussien.

Étant donnés les coefficients  $a[k]$  et  $b[\ell]$  d'un filtre d'ordre fini, l'instruction

```
>> y = filter(b,a,x);
```

génère une réalisation d'un signal stationnaire gaussien, de spectre

$$\mathcal{S}(v) = \left| \frac{\hat{b}(v)}{\hat{a}(v)} \right|^2 , \quad v \in [-1/2, 1/2] .$$

**Remarque 2.18 — Fonction ou distribution?** Le théorème ci-dessus est en fait légèrement incorrect en toute rigueur. Dans certaines situations, le spectre  $\mathcal{S}$  n'est pas une fonction mais une distribution. Considérons par exemple un signal aléatoire harmonique

$$x[\ell] = e^{2i\pi v_0 \ell} e^{2i\pi \varphi} ,$$

où  $\varphi$  est une variable aléatoire, distribuée suivant une loi uniforme entre  $-1/2$  et  $1/2$ . Il est possible de

démontrer que

$$\mathcal{C}[\ell] = e^{2i\pi v_0 \ell},$$

qui n'est pas transformée de Fourier inverse d'une fonction classique, mais de la distribution de Dirac en  $v_0$ .

Le théorème de Wiener-Khinchin fournit une représentation spectrale de la fonction d'autocovariance  $\mathcal{C}$  d'un signal aléatoire stationnaire en moyenne quadratique, c'est à dire une quantité déterministe (définie par une espérance mathématique). Il ne dit rien par contre pour ce qui est du signal aléatoire lui même. La réponse à cette question est donnée par la représentation de Cramèr, décrite dans le résultat qui suit.

**Théorème 2.19 — Cramèr.** Soit  $x$  un signal aléatoire stationnaire en moyenne quadratique. On suppose que son spectre est une fonction (pas une distribution)  $\mathcal{S}$ . Alors  $x$  peut s'obtenir par filtrage  $x = K_h w$  d'un bruit blanc  $w$ , avec un filtre de réponse fréquentielle égale à  $\hat{h}(v) = \sqrt{\mathcal{S}(v)}$ .

**Remarque 2.20 — Signal aléatoire stationnaire de longueur finie.** Dans le cas pratique de signaux de longueur finie, le théorème de Wiener-Khinchin et le théorème de Cramèr restent valables, après adaptation au cas considéré. On rappelle qu'un signal aléatoire de longueur finie est stationnaire en moyenne quadratique si les hypothèses de la Définition 2.4 sont satisfaites, en précisant que les indices des vecteurs sont à prendre dans  $\mathbb{Z}_L$ , c'est à dire toutes les additions et soustractions sont effectuées modulo  $L$ .

1. En définissant l'autocovariance comme  $\mathcal{C}[\ell] = \mathbb{E}\{(x[\ell] - \mu)(x[0] - \mu)\}$ , où  $\mu = \mathbb{E}\{x[0]\}$ , on a

$$\mathcal{C}[\ell] = \frac{1}{L} \sum_{k=0}^{L-1} \mathcal{S}[k] e^{2i\pi k \ell / L},$$

où  $\mathcal{S} \in \mathbb{R}_+^L$  est un vecteur de coordonnées non-négatives, appelé spectre de  $x$ .

2. On peut écrire

$$x[\ell] = \frac{1}{\sqrt{L}} \sum_{k=0}^{L-1} \sqrt{\mathcal{S}[k]} w[k] e^{2i\pi k \ell / L},$$

où  $w$  est un bruit blanc de moyenne nulle et de variance égale à 1, en d'autres termes un signal aléatoire défini par  $\mathbb{E}\{x[\ell]\} = 0$ , et  $\mathbb{E}\{x[\ell]x[0]\} = \delta_{\ell,0}$  pour tout  $\ell \in \mathbb{Z}_L$ .



### Génération d'un bruit de spectre donné

Étant donnée un spectre  $\mathcal{S} \in \mathbb{R}_+^L, \dots$

On représente en Figure 2.8 des estimations de spectres d'instruments de musique jouant une note tenue. La question suivante est : "comment obtenir ces estimations"? C'est le problème d'estimation spectrale.

## 2.3.2 Estimation spectrale

Le problème d'estimation spectrale est un problème classique en statistique et traitement du signal. Le problème est le suivant : on dispose d'une signal de longueur finie, que l'on suppose être une observation de longueur finie d'un signal aléatoire faiblement stationnaire. La question est la suivante : comment estimer le spectre de celui-ci (qui est une quantité moyennée sur l'ensemble des réalisations possibles) à partir de cette unique observation.

Étant donné un signal de longueur  $L$  échantillons, on dispose de plusieurs approches pour estimer un spectre.

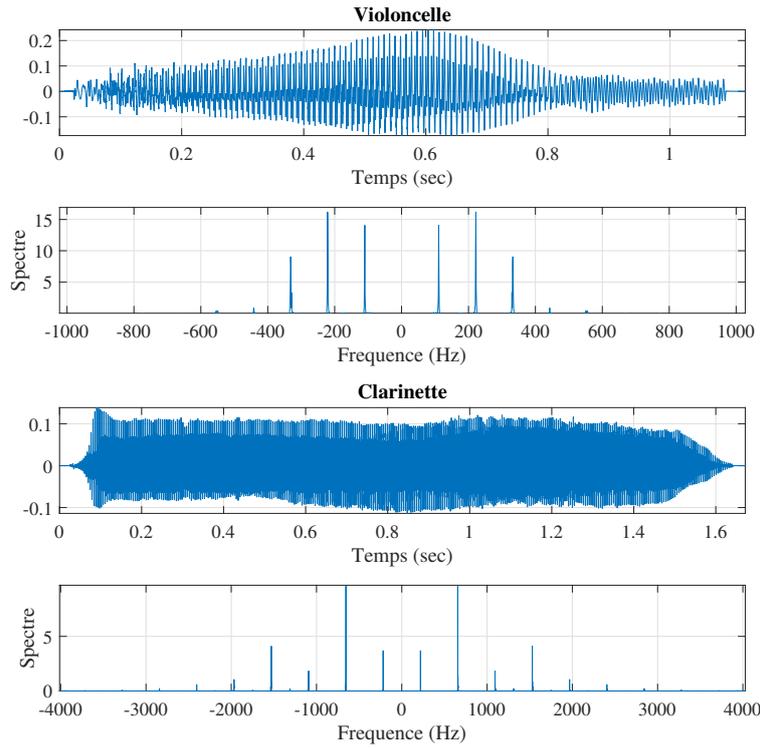


FIGURE 2.8 – Tracés temporels d’une note tenue, avec deux instruments différents, et spectres correspondants estimés.

— **Le Périodogramme** : transformation de Fourier finie

$$S\left(\frac{k\eta}{L}\right) = \frac{1}{L} \left| \sum_{\ell=0}^{L-1} x[\ell] e^{-2i\pi k\ell/L} \right|^2 = \frac{1}{L} |\hat{x}[k]|^2 \quad (2.25)$$

Le spectre estimé par le périodogramme est souvent très “irrégulier”. En fait, il souffre de deux défauts importants, que l’on peut quantifier en revenant au cadre aléatoire. En effet, si le signal  $x$  est modélisé comme un signal aléatoire, le périodogramme tel que défini en (2.25) est lui aussi aléatoire, ce qui permet d’introduire les quantités suivantes :

- Son biais :
- Sa variance (qui lui confère son caractère irrégulier) : si le signal est modélisé comme un signal aléatoire, le périodogramme tel que défini plus haut est lui aussi aléatoire.
- **Le Périodogramme moyenné** : le signal est découpé en segments de longueur  $N$ , un spectre est estimé par calcul du périodogramme sur chacun des segments, puis et on effectue les moyennes des estimées correspondantes du spectre

$$S\left(\frac{k\eta}{N}\right) = \frac{1}{P} \sum_{p=0}^{P-1} \frac{1}{N} \left| \sum_{n=0}^{N-1} x[n+p\delta] e^{-2i\pi kn/N} \right|^2 \quad (2.26)$$

Le spectre estimé correspondant est généralement plus “lisse”

- **Le Périodogramme de Welch** : il est similaire au périodogramme moyenné, à ceci près que des fenêtres sont utilisées pour la segmentation, au lieu de la fenêtre rectangulaire. En notant  $h \in \mathbb{R}^N$  la fenêtre, on définit cette fois

$$S\left(\frac{k\eta}{N}\right) = \frac{1}{P} \sum_{p=0}^{P-1} \left| \sum_{n=0}^{N-1} h[n+p\delta] x[n+p\delta] e^{-2i\pi kn/N} \right|^2. \quad (2.27)$$

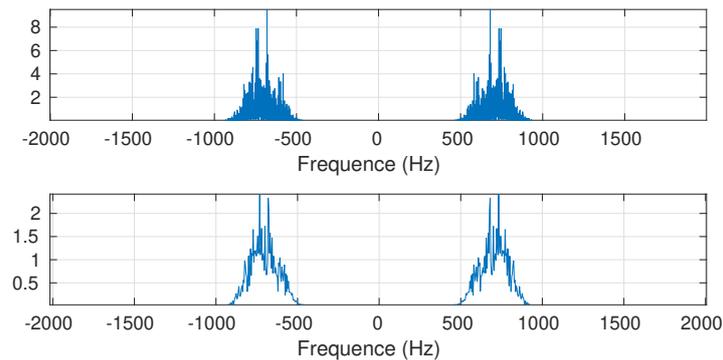


FIGURE 2.9 – Exemples de périodogrammes

**Remarque 2.21 — Interprétation.** On a déjà vu dans l'exercice 2.4 l'effet sur la transformée de Fourier de la multiplication par une fenêtre : c'est un produit de convolution par la transformée de Fourier de la fenêtre.

1. Dans le cas du périodogramme classique, la fenêtre est une fenêtre rectangulaire, sa transformée de Fourier est une fonction qui présente plusieurs lobes, la convolution avec une telle fonction crée des artefacts. On préfère donc des fenêtres plus "lisses", qui diminuent ces artefacts.
2. Plus la longueur  $N$  de la fenêtre est grande (donc plus  $P$  est petit), plus l'estimée du spectre est biaisée. Par contre plus  $N$  est grand plus le spectre estimé est régulier (lisse). On cherche donc généralement un compromis entre biais et régularité.

■ **Exemple 2.9 — Comparaison de périodogrammes.** On a généré ici un bruit "passe bande" synthétique, par filtrage passe-bande (voir plus bas) d'un bruit blanc Gaussien. La bande du bruit est essentiellement comprise entre 440 Hz et 880 Hz. On représente en Figure 2.9 le périodogramme (haut) et un périodogramme moyenné. On peut constater que le périodogramme standard est extrêmement irrégulier, alors que le périodogramme moyenné est plus lisse. Le code Matlab produisant ces figures est donné ci-dessous. ■



On commence par générer le bruit passe-bande. La séquence d'instructions

```
>> eta = 44100;
>> duree = 2;
>> L = duree * eta;
>> temps = linspace(0,duree,L);
>> frequences = linspace(-eta/2,eta/2,L);
```

définit la fréquence d'échantillonnage, la durée et donc la longueur du signal, puis définit des vecteurs correspondant aux temps et aux fréquences. Ensuite, la séquence d'instruction qui suit génère le signal. On suppose disposer d'un filtre de réponse impulsionnelle nommée `PasseBande_RI`, on génère tout d'abord un bruit blanc Gaussien `BB` (instruction `randn`), puis celui-ci est filtré (fonction `filter`) par le filtre, ce qui produit le signal `BC`.

```
>> BB = randn(L,1);
>> BC = filter(PasseBande_RI,1,BB);
```

L'estimation spectrale par périodogramme utilise la transformation de Fourier discrète (fonction `fft`), le périodogramme moyenné se base sur des périodogrammes plus courts.

```
>> pgram = fftshift(abs(fft(BC)).^2)/L;
>> P = 8;
>> tmp = reshape(BC,L/P,P);
>> mpgram = (abs(fft(tmp,[],1))).^2/(L/P);
>> mpgram = sum(mpgram,2)/P;
>> figure;
>> subplot(2,1,1); plot(frequences,pgram); axis tight; grid
```

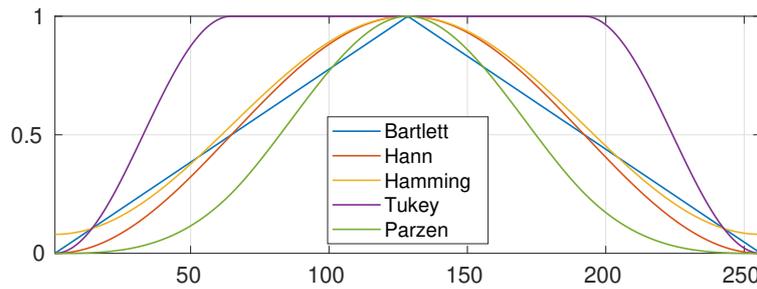


FIGURE 2.10 – Quelques exemples de fenêtres temporelles couramment utilisées

```
>> subfreq = linspace(-eta/2,eta/2,L/P);
>> subplot(2,1,2); plot(subfreq,fftshift(mpgam)); axis tight; grid
```

Les spectres estimés représentés en Figure 2.9 sont des zooms des figures obtenues ci-dessus, focalisées sur les bandes de fréquence d'intérêt.

Pour estimer le périodogramme de Welch, on peut utiliser directement la fonction `pwelch`

### 2.3.3 Estimation d'un modèle AR

On passe maintenant à une autre forme d'estimation spectrale, que l'on qualifie parfois de paramétrique. L'idée est de modéliser le spectre d'un signal comme le module carré d'une fonction rationnelle trigonométrique, c'est à dire de considérer que le signal est obtenu par filtrage d'ordre fini d'un signal dont le spectre est connu, par exemple un bruit blanc. On va ici se limiter à ce cas simple où le signal d'entrée du filtre est un bruit blanc, centré (c'est à dire de moyenne nulle) et de variance connue  $\sigma^2$ , et où le filtre n'a que des pôles et pas de zéro.

Considérons donc un filtre d'ordre fini de la forme

$$y[n] = - \sum_{k=1}^K a[k]y[n-k] + x[n] \quad (2.28)$$

qui est une forme simplifiée du filtre récursif (2.17), dans lequel le filtre  $b$  est une impulsion ( $b_\ell = \delta_{\ell,0}$ ), et que l'on a écrit ici sous une forme légèrement différente. Dans le cas où  $x$  est un bruit blanc, ceci constitue un modèle  $AR(K)$  pour le signal  $y$ .

Supposons que l'on cherche à identifier les coefficients  $a[k]$ . On utilise pour cela le principe de minimisation de l'erreur en moyenne quadratique, en d'autres termes on cherche les coefficients  $a[k]$  qui minimisent la quantité ci-dessous, parfois appelée erreur de prédiction

$$E = \mathbb{E} \left\{ \left| y[\ell] - \left[ x[\ell] - \sum_{k=1}^K a[k]y[\ell-k] \right] \right|^2 \right\}$$

Il est possible de démontrer que ces coefficients sont solution d'un système linéaire, appelé **équations de Yule Walker**

$$\sum_{k=0}^K a[k] \mathcal{C}_y[\ell-k] = \mathcal{C}_{xy}[\ell] \quad (2.29)$$

où les nombres  $\mathcal{C}_y$  et  $\mathcal{C}_{xy}$  sont respectivement les fonctions d'autocovariance de  $y$  et de covariance croisée de  $x$  et  $y$  :

$$\mathcal{C}_y[\ell-k] = \mathbb{E} \{ y[n-k] \bar{y}[n-\ell] \}, \quad \mathcal{C}_{xy}[\ell] = \mathbb{E} \{ x[n] \bar{y}[n-\ell] \} \quad (2.30)$$

(par stationnarité, ces quantités sont indépendantes de  $n$ ).

Ici ces covariances  $\mathcal{C}_y$  et  $\mathcal{C}_{xy}$  sont des quantités probabilistes (théoriques); en pratique elles doivent être remplacées par des valeurs estimées. En général, on prend pour  $x$  un bruit blanc de moyenne nulle dont on note  $\sigma^2$  la variance. Dans ce cas, il est possible de montrer que

$$\mathcal{C}_{xy}[\ell] = 0 \quad \forall \ell > 0, \quad \text{et} \quad \mathcal{C}_{xy}[0] = \sigma^2,$$

de sorte que l'équation de Yule-Walker devient

$$\sum_{k=1}^K a[k] \mathcal{C}_y[\ell - k] = -\mathcal{C}_y[\ell], \quad \ell > 0. \quad (2.31)$$

**Exercice 2.17 — Equations de Yule-Walker.** Démontrer les équations de Yule-Walker (2.29) puis (2.31).

Les équations de Yule-Walker sont résolues numériquement, généralement en utilisant un algorithme appelé algorithme de Levinson-Durbin.



Pour identifier un modèle AR, on doit en premier lieu estimer les coefficients de corrélation. La fonction `xcorr` peut être utilisée à cet effet. Ensuite, la fonction `levinson` peut être utilisée pour résoudre le problème par l'algorithme de Levinson-Durbin.

Par exemple, pour estimer les coefficients d'un signal autorégressif défini par le filtre "tout pôles défini par  $a = \{1, 0.1, -0.8\}$  on peut exécuter la séquence d'instructions suivante :

Dans un premier temps, générer le signal

```
>> L = 15000;
>> a = [1 0.1 -0.8];
>> variance = 0.4;
>> x = sqrt(variance)*randn(L,1);
>> y = filter(1,a,x);
```

Pour estimer les fonctions de corrélation

```
>> [r,lg] = xcorr(y,'biased');
>> r(lg<0) = [ ];
```

Ensuite, on utilise l'algorithme de Levinson-Durbin

```
>> ar = levinson(r,numel(a)-1)
```

qui fournit une estimée `ar` des coefficients du filtre `a`.

Alternativement (et plus simplement), la fonction `aryule` estime directement les coefficients du filtre, ainsi que la variance  $\sigma^2$  du bruit blanc d'entrée, à partir de l'observation

```
>> ar = aryule(y,numel(a)-1)
```

Voir aussi la fonction `lpc` (pour *linear predictive coding*) qui effectue des opérations très similaires.

On peut visualiser l'estimation spectrale non-paramétrique obtenue avec `pwelch` et l'estimation paramétrique obtenue avec le modèle AR (utiliser l'outil `fvtool`). Le résultat se trouve en Figure 2.11.

### 2.3.4 Modèle ARMA

Les modèles ARMA (auto-régressifs à moyenne mobile), aussi appelés modèles de Box Jenkins, sont des généralisations des modèles AR vus plus haut. On dit qu'un signal  $y$  suit un modèle ARMA( $K, L$ ) si on a

$$y[n] = \sum_{\ell=0}^L b[\ell]x[n-\ell] - \sum_{k=1}^K a[k]y[n-k], \quad \forall n, \quad (2.32)$$

où les  $a[k]$  et  $b[\ell]$  sont des nombres réels, et où  $y$  est un bruit blanc. Le spectre d'un signal construit via un modèle ARMA est la fonction rationnelle trigonométrique

$$\mathcal{S}(v) = \left| \frac{\sum_{\ell=0}^L b[\ell]e^{-2i\pi v\ell}}{\sum_{k=0}^K a[k]e^{-2i\pi vk}} \right|^2.$$

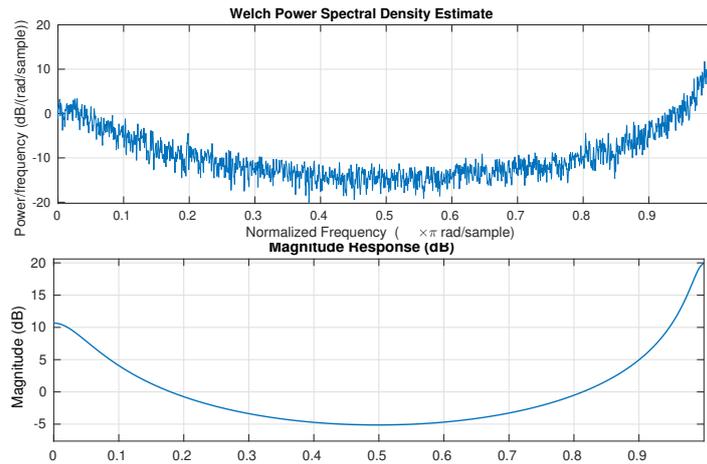


FIGURE 2.11 – Estimation non-paramétrique (haut) et paramétrique (bas) du spectre d'un signal AR (les unités sont différentes dans les deux cas).

L'estimation des paramètres d'un modèle ARMA est plus complexe que dans le cas du modèle AR. L'erreur de prédiction s'écrit

$$E = \mathbb{E} \left\{ \left| y[n] - \left[ \sum_{\ell=0}^L b[\ell]x[n-\ell] - \sum_{k=1}^K a[k]y[n-k] \right] \right|^2 \right\}$$

et dépend de deux jeux de paramètres  $a$  et  $b$ .

Une difficulté importante réside dans des pré-traitements (estimation de ligne de base, estimation des paramètres  $K$  et  $L$ ). L'identification des paramètres du modèle résultant est généralement effectuée au moyen d'algorithmes dédiés, qui sont implémentés dans les logiciels standard.





## 3. Représentations temps-fréquence

### 3.1 Introduction : représentation fréquentielle locale

On a vu dans le chapitre précédent la notion de signal stationnaire, défini en termes probabilistes comme des signaux dont la moyenne et la covariance ne varient pas au cours du temps. La conséquence fondamentale est le théorème de Wiener-Khinchin, qui établit l'existence de la densité spectrale  $\mathcal{S}$  (ou spectre, ou spectre de puissance). Le spectre donne une représentation fréquentielle du signal, au sens où ce théorème exprime la fonction de covariance  $\mathcal{C}$  comme transformée de Fourier inverse du spectre. En poussant plus loin, il est possible d'établir un théorème complémentaire, appelé **théorème de Cramér**, qui donne quant à lui une représentation de Fourier du signal (aléatoire) lui-même, plutôt que de sa fonction d'autocovariance  $\mathcal{C}$ . Ce difficile théorème étend donc la transformation de Fourier au cas des signaux aléatoires, et va bien au delà du cadre de ce cours.

En prenant un point de vue un peu plus global, on peut aussi considérer qu'un signal stationnaire est un signal dont les caractéristiques spectrales, en d'autres termes le contenu fréquentiel, ne varie pas au cours du temps. Par opposition, on va considérer un signal non-stationnaire comme un signal où cette condition n'est plus réalisée. Par exemple

- On pourra considérer qu'une note tenue de violon constitue un signal stationnaire (en première approximation, et si on fait abstraction de l'attaque), alors qu'un glissando est clairement non-stationnaire.
- Un bruit de moteur thermique d'un véhicule à l'arrêt peut être considéré comme stationnaire, alors que le bruit de moteur du même véhicule en phase d'accélération est clairement non-stationnaire.

Ces deux exemples appartiennent à une famille de signaux que l'on peut qualifier de *localement stationnaire*, au sens où on pourrait vouloir leur associer un contenu fréquentiel variable au cours du temps. Ils ne sont pas représentatifs de l'ensemble des signaux non-stationnaires, mais c'est le cadre dans lequel on va travailler ici. On va donc étudier des outils permettant de donner un sens à cette notion de stationnarité locale.

### 3.2 Transformation de Fourier à court terme (TFCT)

L'idée la plus simple est de *localiser* la transformée de Fourier. Une façon de faire cela est d'utiliser une fenêtre pour localiser le signal au voisinage d'un instant d'intérêt, avant de calculer la transformée de Fourier. En faisant glisser, ou coulisser la fenêtre le long de l'axe temporel, on obtient une analyse de Fourier en tout instant. Ce que l'on obtient dépend évidemment de l'outil d'analyse, ici la fenêtre.

### 3.2.1 TFCT pour des signaux de longueur infinie

On décrit dans un premier temps la TFCT dans le cadre des signaux de longueur infinie, avant de discuter les variantes possibles.

**Définition 3.1 — Transformation de Fourier à court terme (TFCT/STFT).** Soit  $g \in \ell^2(\mathbb{Z})$  une fenêtre. La transformée de Fourier à court terme de  $x \in \ell^2(\mathbb{Z})$  est la fonction de deux variables  $\mathcal{V}_g x$  donnée par

$$\mathcal{V}_g x[v, n] = \sum_{\ell=-\infty}^{\infty} x[\ell] \bar{g}[\ell - n] e^{-2i\pi v(\ell - n)} .$$

Le nombre (complexe)  $\mathcal{V}_g x[v, n]$  décrit donc le contenu de  $x$  au voisinage de la fréquence  $v$  et de l'échantillon  $n$ . On représente généralement cette fonction de deux variables par une image, appelée **spectrogramme**, qui encode son module (ou le logarithme de celui-ci) par un code de couleurs ou un niveau de gris.

On va maintenant voir ce que signifie cette notion de voisinage, et l'importance de la fenêtre dans cette notion. Mais auparavant il est utile de faire quelques remarques.

**Remarque 3.1 — Définitions équivalentes.** 1. Une autre interprétation importante fait intervenir ce que l'on appellera *atomes temps-fréquence*  $g_{(v,n)}$ , définis par

$$g_{(v,n)}[\ell] = g[\ell - n] e^{2i\pi v(\ell - n)} , \quad (3.1)$$

on a en effet

$$\mathcal{V}_g x[v, n] = \langle x, g_{(v,n)} \rangle , \quad (3.2)$$

cette interprétation sera importante pour la suite. Des exemples d'atomes STFT sont tracés dans la figure 3.1.

2. On a déjà vu que la transformation de Fourier discrète satisfait la formule de Parseval. On peut donc écrire (après calcul)

$$\mathcal{V}_g x[v, n] = \langle \hat{x}, \widehat{g_{(v,n)}} \rangle = \int_{-1/2}^{1/2} \hat{x}(\xi) \bar{\hat{g}}(\xi - v) e^{2i\pi \xi n} d\xi . \quad (3.3)$$

$\mathcal{V}_g x$  peut donc être interprétée comme une STFT de la transformée de Fourier  $\hat{x}$  de  $x$ , avec une fenêtre égale à la transformée de Fourier  $\hat{g}$  de  $g$ .

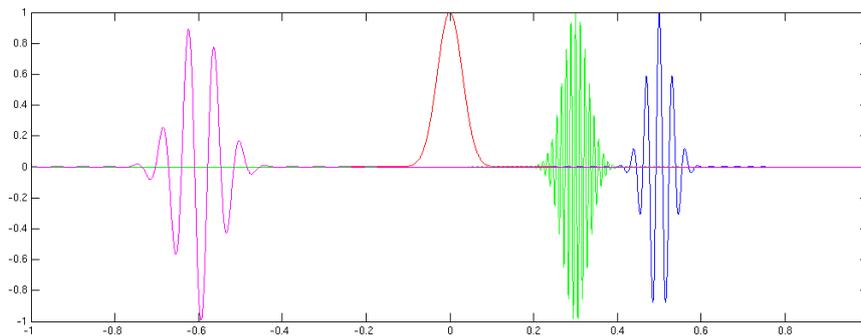


FIGURE 3.1 – Exemples d'atomes temps-fréquence (vert et bleu), copies translattées et modulées d'une fenêtre  $g$  (en rouge). L'atome temps-fréquence tracé en magenta correspond à une fenêtre différente (plus large)

**Remarque 3.2 — Interprétation en termes de bancs de filtres.** Il est aussi possible, en introduisant les fenêtres modulées  $g_\nu$  définies par

$$g_\nu[\ell] = e^{2i\pi\nu\ell} g[\ell] \quad (3.4)$$

d'écrire

$$\mathcal{V}_g[\nu, n] = (x * \tilde{g}_\nu)[n], \quad (3.5)$$

où  $\tilde{y}$  représente génériquement la suite retournée et complexe conjuguée de la suite  $y$

$$\tilde{y}[\ell] = \bar{y}[-\ell].$$

Ainsi,  $\mathcal{V}_g x$  peut être vu comme une famille (infinie continue) de signaux, obtenus par filtrage de  $x$  par des filtres de réponses impulsionnelles respectives  $\tilde{g}_\nu$ . Cette famille de filtres est appelé banc de filtres (continu).

**Exercice 3.1 — Définitions équivalentes de la TFCT.** Démontrer les expressions (3.2), (3.3) et (3.5). Pour l'équation (3.3) on pourra par exemple partir du membre de droite de l'équation, et remplacer  $\hat{x}(\xi)$  par sa définition. ■

**Remarque 3.3 — Convention de phase.** On utilise parfois aussi la définition alternative basée sur des atomes définis par

$$g_{(\nu, n)}^\circ[\ell] = g[\ell - n] e^{2i\pi\nu\ell}.$$

En notant  $\mathcal{V}_g^\circ$  la transformation correspondante, on écrit alors

$$\mathcal{V}_g^\circ x[\nu, n] = \langle x, g_{(\nu, n)}^\circ \rangle = \sum_{\ell=-\infty}^{\infty} x[\ell] \bar{g}[\ell - n] e^{-2i\pi\nu\ell} = \int_{-1/2}^{1/2} \hat{x}(\xi) \bar{g}(\xi - \nu) e^{2i\pi(\xi - \nu)n} d\xi.$$

Ces deux formulations ne diffèrent que d'un facteur de phase :

$$\mathcal{V}_g^\circ x[\nu, n] = e^{-2i\pi\nu n} \mathcal{V}_g x[\nu, n].$$

On se limitera à la première, sauf dans le cas particulier des transformations de Gabor non-stationnaires que nous verrons à la fin de ce chapitre.

**Remarque 3.4 — Le rôle de la fenêtre.** — On voit bien que pour donner à l'analyse une précision temporelle suffisante, on aimerait que la fenêtre  $g$  soit suffisamment courte. Par convention, et sans perte de généralité on suppose  $g$  centrée sur  $\ell = 0$ . Par contre, l'équation (3.3) montre aussi que pour que  $\mathcal{V}_g x[\nu, n]$  décrive le contenu fréquentiel de  $x$  au voisinage de la fréquence  $\nu$ , on a aussi besoin que  $\hat{g}$  soit centrée sur la fréquence  $\xi = 0$  (ceci est fait sans perte de généralité aussi), mais aussi que  $\hat{g}$  soit bien localisée autour de  $\xi = 0$  (c'est à dire décroissant assez vite lorsque l'on s'éloigne de l'origine des fréquences). Or ces deux conditions de localisation (en temps et en fréquence) sont relativement incompatibles. Le principe d'incertitude stipule qu'il est impossible de trouver une fonction ou une suite qui soit optimalement localisée à la fois en temps et en fréquence. On doit donc rechercher des compromis, qui conduisent à utiliser une fenêtre gaussienne, ou l'une des fenêtres classiques dont les graphes se trouvent en Figure 2.10.

— Même avec une fenêtre fixée, la taille de la fenêtre joue un rôle important. Plus la fenêtre est longue, plus elle est imprécise dans le domaine temporel et précise dans le domaine fréquentiel, et vice-versa.

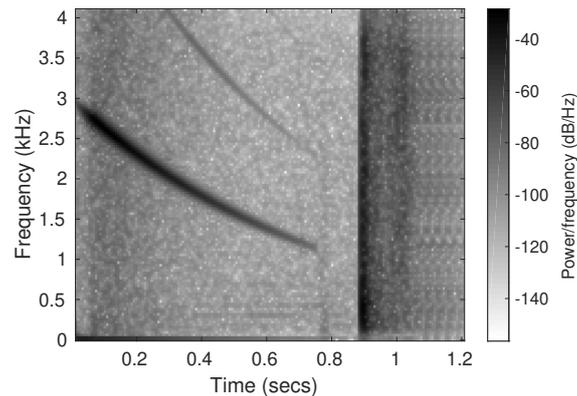


FIGURE 3.2 – Spectrogramme d'un signal audio contenant deux composantes modulées en amplitude et en fréquence, calculé avec la fonction `spectrogram`.



### LTFAT.

La toolbox *Signal Processing* de Matlab fournit quelques outils d'analyse temps-fréquence, par exemple la fonction `spectrogram`, qui retourne et/ou trace (sous forme d'une image) le spectrogramme, module carré de la transformée de Fourier à court terme. Cette fonction (qu'on peut voir comme une analogue de la fonction `pwelch` pour le cas non-stationnaire, est un très bon outil de visualisation, et offre plusieurs options. Ci-dessous un exemple sur un signal audio fourni dans la *Signal Processing Toolbox*

```
>> load splat
>> g = hann(256);
>> superposition = 200;
>> spectrogram(y,g,superposition,[],Fs,'yaxis')
>> cmap = 1-gray; colormap(cmap);
```

La première ligne charge un signal `y` et sa fréquence d'échantillonnage `Fs`. On précise ensuite la fenêtre choisie et le degré de superposition (*overlap*) de deux fenêtres consécutives avant de calculer le spectrogramme. Le résultat se trouve en Figure 3.2.

Dans ce cours, on utilisera plutôt une autre toolbox (libre), développée notamment à l'Institut de Recherche en Acoustique de l'Académie des Sciences de Vienne (Autriche), qui implémente (bien) une grande quantité d'outils, en particulier des outils de synthèse. Cette toolbox est appelée LTFAT (pour Linear Time-Frequency Analysis toolbox [6] puis sa version étendue, Large Time-Frequency Analysis Toolbox [3]).

- **Exemple 3.1** On représente en Figure 3.2 un spectrogramme d'un signal audio, contenant tout d'abord une phase avec deux composantes modulées en amplitude et en fréquence, puis dans un second temps un transitoire. Le spectrogramme est calculé avec la fonction `spectrogram` de la *Signal Processing Toolbox*, et est représenté en Figure 3.2. On y voit bien apparaître les deux composantes modulées en amplitude et en fréquence (fréquence décroissante au cours du temps), puis la seconde phase du signal. ■



### Calcul de spectrogramme.

Les lignes qui suivent détaillent le calcul du spectrogramme d'une sinusoïde de fréquence  $4 \times 440$  Hz, et d'une sinusoïde de fréquence variant au cours du temps. Tout d'abord la génération de la sinusoïde

```
>> Fs = 44000;
>> F0 = 4*440;
>> L = Fs;
>> ttt = linspace(0,1,L);
>> x = cos(2*pi*F0*ttt)';
```

puis le calcul du spectrogramme, en utilisant la fonction `dgt` (dont la syntaxe sera détaillée un peu plus loin)

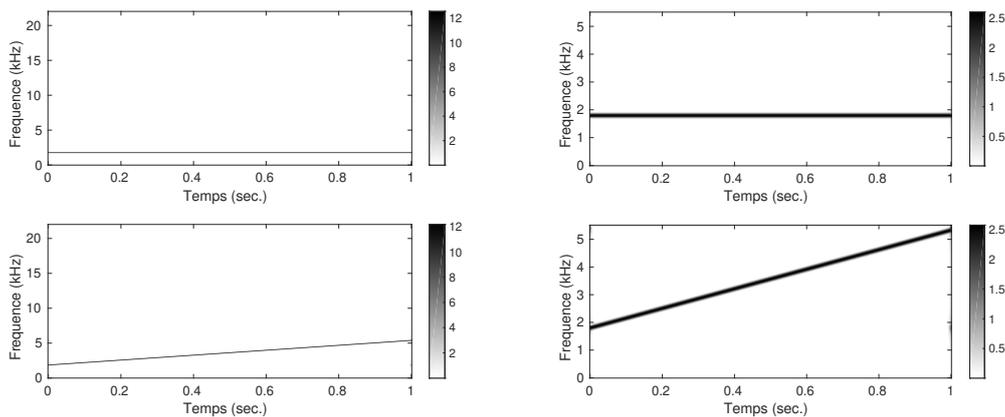


FIGURE 3.3 – Spectrogrammes d’une sinusoïde, et d’un *chirp* linéaire (sinusoïde dont la fréquence varie linéairement en fonction du temps). Gauche : domaine fréquentiel complet. Droite : zoom sur les plus basses fréquences.

```
>> a = 100;
>> M = 2000;
>> Vx = dgt(x, 'gauss', a, M, L);
```

même chose pour le *chirp* linéaire

```
>> alpha = 2;
>> y = cos(2*pi*F0*(ttt + alpha*ttt.^2/2));
>> Vy = dgt(y, 'gauss', a, M, L);
```

On représente graphiquement les représentations temps-fréquence sous forme d’image, où le module est encodé soit par un code couleur soit par un niveau de gris (comme ici, défini par le codage `cmap`).

```
>> fff = linspace(0, Fs/2, M/2);
>> subplot(2,1,1)
>> imagesc(ttt, fff/1000, abs(Vx(1:M/2, :))); axis xy; colorbar
>> xlabel('Temps (sec.)');
>> ylabel('Frequence (kHz)');
>> set(gca, 'fontsize', 14);
>> subplot(2,1,2)
>> imagesc(ttt, fff/1000, abs(Vy(1:M/2, :))); axis xy; colorbar
>> xlabel('Temps (sec.)');
>> ylabel('Frequence (kHz)');
>> set(gca, 'fontsize', 14);
>> cmap=1-gray;
>> colormap(cmap);
```

**Théorème 3.5 — Inversion de la TFCT.** Si  $g$  n’est pas uniformément nulle, la TFCT correspondante est inversible, et satisfait une formule de Parseval (condition de conservation de l’énergie). On a en effet, pour tout  $x \in \ell^2(\mathbb{Z})$

$$x[\ell] = \frac{1}{\|g\|^2} \sum_{n=-\infty}^{\infty} \int_{-1/2}^{1/2} \mathcal{Y}_g x[v, n] g_{v, n}[\ell] dv,$$

et

$$\|\mathcal{Y}_g x\|^2 := \sum_{n=-\infty}^{\infty} \int_{-1/2}^{1/2} |\mathcal{Y}_g x[v, n]|^2 dv = \|g\|^2 \|x\|^2.$$

**Exercice 3.2 — Inversion de la TFCT.** Démontrer le théorème 3.5. Dans les deux cas, on pourra commencer par calculer l'intégrale sur les fréquences, en utilisant le fait que

$$\int_{-1/2}^{1/2} e^{2i\pi\nu k} d\nu = \delta_{k,0},$$

et simplifier la somme ensuite. L'interversion de la somme et l'intégrale peut se justifier par des arguments rigoureux, qu'on n'abordera pas ici. ■

### 3.2.2 TFCT en dimension finie

Les exemples décrits dans la section précédentes ne correspondent pas exactement à la TFCT telle qu'on l'a décrite, en effet il s'agit de signaux de longueur finie, et la variable fréquentielle est une variable discrète (et même de longueur finie), et pas continue comme dans la définition 3.1.

Il existe une version équivalente de la TFCT adaptée au cas de dimension finie, c'est celle-ci qui est en fait implémentée dans les paquets numériques. Dans les expressions ci-dessous, les vecteurs sont tous de longueur  $L$ , et on suppose implicitement des conditions aux bords périodiques, c'est à dire de la forme  $x[n+L] = x[n]$  pour tout  $n$ .

**Définition 3.2 — TFCT en dimension finie.** Étant donné un vecteur  $g \in \mathbb{C}^L$ , la transformation de Fourier à court terme correspondante est la transformation linéaire  $\mathcal{V}_g : \mathbb{C}^L \rightarrow \mathbb{C}^{L \times L}$  définie par

$$\mathcal{V}_g x[m, n] = \langle x, g_{mn} \rangle = \sum_{\ell=0}^{L-1} x[\ell] \bar{g}[\ell - n] e^{-2i\pi m(\ell-n)/L},$$

où on a encore introduit les atomes temps-fréquence  $g_{mn}$  définis par

$$g_{mn}[\ell] = g[\ell - n] e^{2i\pi m(\ell-n)/L}, \quad \ell = 0 \dots L-1.$$

Le vecteur  $g$  sur lequel est basée l'analyse est appelé fenêtre.

À l'instar de la TFCT sur  $\ell^2(\mathbb{Z})$  vue dans la section précédente, cette version de dimension finie est inversible. Calculons par exemple

$$\begin{aligned} \sum_{m,n=0}^{L-1} \mathcal{V}_g x[m, n] g_{m,n}[\ell] &= \sum_{m,n=0}^{L-1} \left[ \sum_{k=0}^{L-1} x[k] \bar{g}[k - n] e^{-2i\pi m(k-n)/L} \right] g[\ell - n] e^{2i\pi m(\ell-n)/L} \\ &= \sum_{k,n=0}^{L-1} \bar{g}[k - n] g[\ell - n] x[k] \sum_{m=0}^{L-1} e^{2i\pi m(\ell-k)/L} \\ &= L \sum_{n=0}^{L-1} |g[n - \ell]|^2 x[\ell] = L \|g\|^2 x[\ell] \end{aligned}$$

où on a utilisé l'expression de la somme d'une série géométrique finie. Ceci montre que  $x$  peut être reconstruit à partir de sa TFCT. Avec la même démonstration, on peut également démontrer une formule de Parseval, et plus généralement on a le résultat suivant :

**Théorème 3.6 — Inversion de la TFCT en dimension finie.** Soient  $g, h \in \mathbb{C}^L$  tels que  $\langle g, h \rangle \neq 0$ . Alors on a la formule d'inversion de la TFCT : pour tout  $x \in \mathbb{C}^L$ ,

$$x = \frac{1}{L \langle h, g \rangle} \sum_{m,n=0}^{L-1} \mathcal{V}_g x[m, n] h_{m,n},$$

et la formule de Parseval

$$\langle \mathcal{V}_g x, \mathcal{V}_h x \rangle = L \langle h, g \rangle \|x\|^2.$$

$h$  est appelée fenêtre de synthèse, par opposition à  $g$  qui est la fenêtre d'analyse.

La première expression montre en particulier que l'on peut utiliser une fenêtre de synthèse  $h$  différente de la fenêtre d'analyse. En particulier, en prenant  $h[\ell] = \delta_{\ell,0}$ , et en supposant donc que  $g[0] \neq 0$ , on a la formule d'inversion simple

$$x[\ell] = \frac{1}{L\bar{g}[0]} \sum_{m=0}^{L-1} \mathcal{V}_g x[m, \ell]. \quad (3.6)$$

**Exercice 3.3 — Symétries de la TFCT.** 1. Étant donné  $x \in \mathbb{C}^L$ , et  $y$  une copie traduite de  $x$  donnée par  $y[\ell] = x[\ell - \tau]$  avec  $\tau \in \mathbb{Z}$ , et  $\ell - \tau$  étant compris modulo  $L$  (conditions aux bords périodiques de période  $L$ ), exprimer  $\mathcal{V}_g y$  en fonction de  $\mathcal{V}_g x$ , et conclure.  
2. Montrer que  $\mathcal{V}_g x$  a aussi une propriété de périodicité par rapport à la variable fréquentielle.  
3. Supposons maintenant que  $x$  et  $g$  soient réels, i.e.  $x, g \in \mathbb{R}^L$ . Montrer que la TFCT hérite de la symétrie Hermitienne de la transformée de Fourier :  $\mathcal{V}_g x[L - m, n] = \overline{\mathcal{V}_g x[m, n]}$ . ■

### 3.3 Sous-échantillonner le plan temps-fréquence : transformation de Gabor

En pratique, lorsque la longueur des signaux d'intérêt devient grande, on ne peut plus conserver toutes les valeurs possibles des indices de temps et de fréquence, et on est amenés à sous-échantillonner la TFCT. Dans ce qui suit, on notera toujours  $L$  la longueur des signaux  $x \in \mathbb{C}^L$ , et on notera

- $a$  le pas d'échantillonnage temporel ;  $a$  doit être un diviseur de  $L$ , on notera  $L = Na$ .
- $b$  le pas d'échantillonnage fréquentiel, qui doit aussi être un diviseur de  $L$ , on notera  $L = Mb$ .

Avec ces notations, on introduit les atomes temps-fréquence, qu'on appellera **atomes de Gabor**, pour les différencier de la TFCT : partant d'une fenêtre  $g \in \mathbb{C}^L$

$$g_{mn}[\ell] = e^{2i\pi mb(\ell-na)/L} g[\ell - na]. \quad (3.7)$$

**Définition 3.3 — Transformation de Gabor.** Étant donnée une fenêtre  $g \in \mathbb{C}^L$ , et  $a, b$  deux diviseurs de  $L$ , la transformation de Gabor correspondante est la transformation linéaire qui associe à tout  $x \in \mathbb{C}^L$  la matrice  $\mathcal{V}_g x \in \mathbb{C}^{M \times N}$  définie par

$$\mathcal{V}_g x[m, n] = \langle x, g_{mn} \rangle = \sum_{\ell=0}^{L-1} x[\ell] \bar{g}[\ell - na] e^{-2i\pi mb(\ell-na)/L}.$$

**Remarque 3.7** 1. La TFCT est donc un cas particulier de transformée de Gabor, pour laquelle on a pris  $a = b = 1$ .  
2. La transformation de Gabor hérite des propriétés d'invariance par translation de la TFCT. Si  $y$  est une copie traduite d'un multiple de  $a$  de  $x$ , c'est à dire  $y[\ell] = x[\ell - \tau a]$ , alors  $\mathcal{V}_g y[m, n] = \mathcal{V}_g x[m, n - \tau]$ . En particulier, on retrouve une forme de conditions aux bords périodiques,  $\mathcal{V}_g x[m, n + N] = \mathcal{V}_g x[m, n]$ .

**Exercice 3.4 — Transformation de Gabor et banc de filtres.** Montrer que la transformation de Gabor peut s'assimiler à un banc de filtres (avec conditions aux bords périodiques), chaque filtrage étant suivi d'un sous-échantillonnage. Expliciter les réponses impulsionnelles des filtres, leurs fonctions de transfert, ainsi que les facteurs de sous-échantillonnage. ■

**Exercice 3.5 — DGT et Fourier.** En utilisant la formule de Parseval  $\mathcal{V}_g x[m, n] = \langle \hat{x}, \widehat{g_{mn}} \rangle$ , expliciter la transformée de Gabor de  $x$  en fonction de  $\hat{x}$ . Vérifier qu'il s'agit (à un facteur de phase près) d'une transformée de Gabor de  $\hat{x}$ , avec une fenêtre qu'on précisera. ■

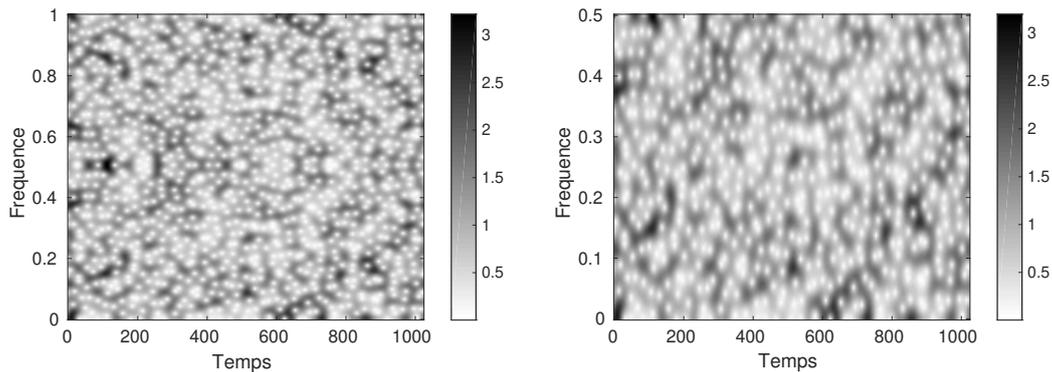


FIGURE 3.4 – Transformée de Gabor d’une réalisation de bruit blanc Gaussien. Représentation complète (gauche) et limitée aux fréquences positives (droite).



### La fonction `dgt`.

La fonction `dgt` de LTFAT implémente la transformation de Gabor. La syntaxe minimale est

```
>> Vx = dgt(x,g,a,M);
```

les variables d’entrée minimales étant le signal d’entrée `x`, la fenêtre `g`, le pas temporel `a` et le nombre de fréquences `M`. A noter, la fenêtre peut être passée soit sous forme d’un vecteur (voir la fonction `pgauss` pour une fenêtre gaussienne périodisée, et l’aide pour d’autres choix). Ci-dessous un exemple de calcul de `dgt`, sur un signal bruit blanc échantillonné à 1 Hz. Dans un premier temps, génération du signal, choix des paramètres, et calcul de la DGT

```
>> L = 1024;
>> x = randn(L,1);
>> a = 4;
>> M = 256;
>> Vx = dgt(x,'gauss',a,M);
```

Représentation graphique sous forme d’une image codée en niveaux de gris :

```
>> ttt = 0:(L-1);
>> fff = linspace(0,1,M);
>> imagesc(abs(Vx)); axis xy; colorbar;
>> imagesc(ttt,fff,abs(Vx)); axis xy; colorbar;
>> cmap = 1-gray; colormap(cmap)
>> xlabel('Temps'); ylabel('Frequence');
>> set(gca,'fontsize',14);
```

On voit bien apparaître la symétrie Hermitienne. La partie basse représente les fréquences positives (ici entre 0 et 1/2), la partie haute les fréquences négatives (entre -1/2 et 0), repliées sur l’intervalle  $[1/2,1]$  par périodicité. On voit que ces deux parties sont images miroir l’une de l’autre, on peut donc se contenter de représenter les fréquences positives, ce qui est fait ci-dessous

```
>> fff = linspace(0,1/2,M/2);
>> imagesc(ttt,fff,abs(Vx(1:M/2,:))); axis xy; colorbar;
>> xlabel('Temps'); ylabel('Frequence');
>> set(gca,'fontsize',14);
```

Le résultat de ces deux tracés se trouve en Figure 3.4.

Notons qu’il est possible de calculer directement la restriction de la transformée de Gabor aux fréquences positives en utilisant la fonction `dgtreal`. La syntaxe est la même. Un exemple se trouve en Figure 3.5 où est représenté un spectrogramme de la *gamme montante* (glissando en spirale) de Risset.

Ces représentations sont plus économiques, mais il y a un prix à payer. L’inversion de la transformée reste possible sous certaines conditions, mais elle n’est généralement pas aussi simple que dans les théorèmes 3.5 et 3.6. L’objet central pour étudier cette inversion est l’**opérateur de trame**  $\mathcal{R}$ .

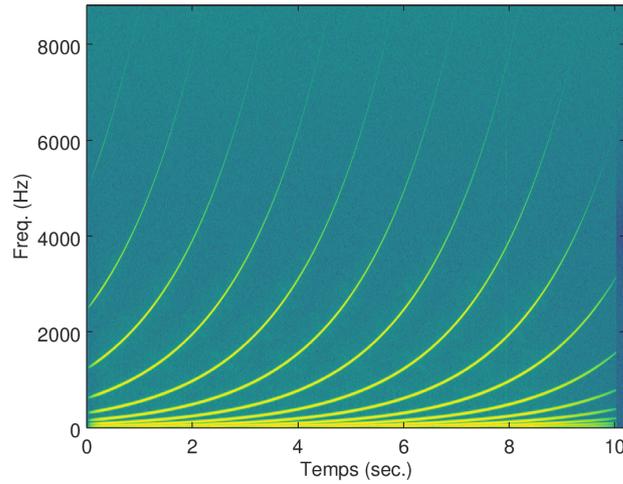


FIGURE 3.5 – Transformée de Gabor d'une gamme montante (glissando en spirale) de Risset.

**Définition 3.4 — Trame de Gabor.** Étant donné une fenêtre (d'analyse)  $g \in \mathbb{C}^L$ , des pas d'échantillonnage temporel  $a$  et fréquentiel  $b$ , et les atomes de Gabor correspondants  $g_{mn}$ , l'opérateur de trame associé est l'opérateur linéaire défini par :

$$\mathcal{R}x = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \langle x, g_{mn} \rangle g_{mn}, \quad x \in \mathbb{C}^L. \quad (3.8)$$

Si cet opérateur est inversible, on dit que les atomes  $\{g_{mn}, m=0 \dots M-1, n=0, \dots N-1\}$  forment une trame de  $\mathbb{C}^L$ , que l'on appelle dans ce cas trame de Gabor.

On peut voir tout de suite que lorsque  $\mathcal{R}$  est inversible, et si son inverse peut être calculé, alors on a directement une formule d'inversion de la forme

$$x = \mathcal{R}^{-1} \left( \sum_{m,n} \langle x, g_{mn} \rangle g_{mn} \right) = \sum_{m,n} \langle x, g_{mn} \rangle \mathcal{R}^{-1}(g_{mn}). \quad (3.9)$$

Il est par ailleurs possible de démontrer que les vecteurs  $\mathcal{R}^{-1}(g_{mn})$  sont encore des atomes de Gabor

$$\mathcal{R}^{-1}(g_{mn}) = (\mathcal{R}^{-1}g)_{mn}, \quad (3.10)$$

de sorte qu'en posant

$$\gamma = \mathcal{R}^{-1}g \quad (3.11)$$

on obtient la formule d'inversion de la transformée de Gabor

$$x = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \langle x, g_{mn} \rangle \gamma_{mn}. \quad (3.12)$$

Ainsi, on obtient une formule d'inversion similaire à celle obtenue dans le théorème 3.6, à ceci près que la fenêtre de synthèse  $\gamma$  est cette fois fixée.  $\gamma$  est appelée **fenêtre duale canonique**. L'évaluation numérique de cette fenêtre duale est un élément important de l'analyse de Gabor.

**Remarque 3.8** Il est important de signaler que la fenêtre duale, lorsqu'elle existe, dépend non seulement de la fenêtre d'analyse, mais aussi des paramètres d'échantillonnage  $a$  et  $b$  (ou  $M$ ). On donne en figure 3.6 quelques exemples de fenêtres duales  $\gamma$ , calculées à partir de la même fenêtre  $g$  pour différentes valeurs des paramètres. Dans les figures du haut, on a pris  $L = 128$ ,  $a = 2$  et  $M = 32$ , on voit que la fenêtre duale est très proche (à un facteur multiplicatif près) de la fenêtre  $g$ . Pour  $a = 8$  et  $M = 16$  (milieu), la fenêtre

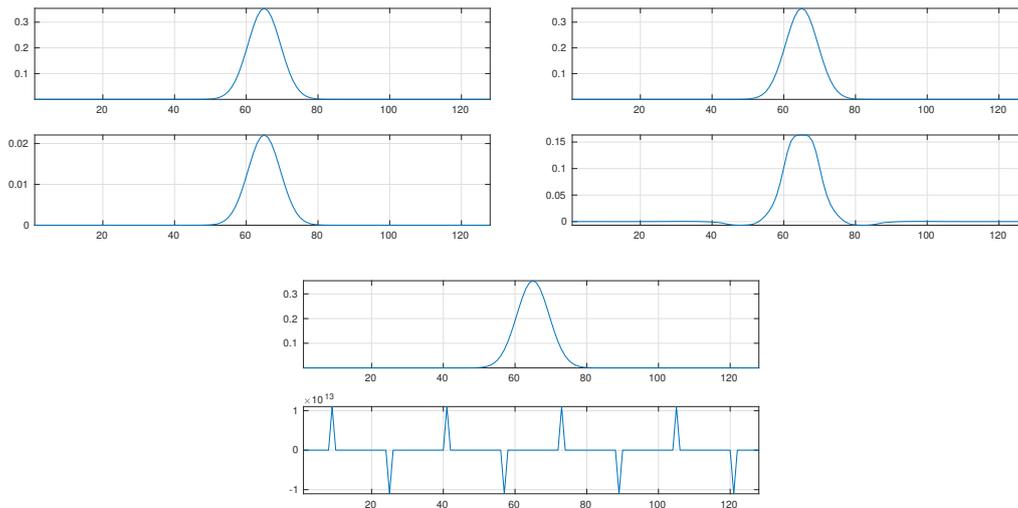


FIGURE 3.6 – Fenêtres d’analyse (haut) et duale canonique (bas), pour la même fenêtre d’analyse et différentes valeurs des paramètres  $a$  et  $b$ . Dans le dernier cas, l’opérateur de trame  $\mathcal{R}$  n’est pas inversible, la fenêtre duale canonique est singulière.

de synthèse se différencie davantage de la fenêtre d’analyse, alors que pour  $a = 16$  et  $M = 16$ , l’opérateur de trame n’est plus inversible, et la fenêtre duale n’a plus de sens (voir les valeurs des pics).



#### Calculs de fenêtres duales canoniques.

La fonction `gabdual` permet de calculer la fenêtre duale canonique d’une fenêtre d’analyse donnée

```
>> L = 128;
>> a = 2;
>> M = 32;
>> g = pgauss(L);
>> gamma = gabdual(g,a,M);
>> subplot(2,1,1); plot(fftshift(g)); axis tight; grid;
>> subplot(2,1,2); plot(fftshift(gamma)); axis tight; grid;
```

Dans les deux autres cas, on a pris  $a=8$  et  $M=16$ , puis  $a=16$  et  $M=16$ .



#### Transformation inverse

La fonction `idgt` permet de calculer une transformation de Gabor inverse. La syntaxe est `xrec = idgt(V,gam,a,L)`. Les quelques lignes qui suivent donnent un exemple de transformation de Gabor inverse, après calcul de la fenêtre duale. On génère ici la fenêtre en utilisant la fonction `gabwin`.

```
>> g = gabwin('gauss',a,M,L);
>> gam = gabdual(g,a,M,L);
>> V = dgt(x,g,a,M,L);
>> xrec = idgt(V,gam,a,L);
>> norm(x-xrec)/norm(x)
2.8845e-16
```

La dernière ligne compare le signal reconstruit et le signal original.

Le calcul de la transformation de Gabor inverse repose sur l’inversion de l’opérateur de trame. Cette inversion est effectuée numériquement, grâce à des algorithmes dédiés. Il existe toutefois des situations simples dans lesquelles une forme explicite peut être trouvée pour la fenêtre duale, auquel cas l’inversion est beaucoup plus simple.

**Proposition 3.9 — Inversion sans effort.** Avec les mêmes notations que précédemment, supposons que la longueur du support de la fenêtre  $g$  soit inférieure à  $M$ , et que pour tout  $\ell$  on ait  $\sum_{n=0}^{N-1} |g[\ell - na]|^2 \neq 0$ . Alors l'opérateur de trame est inversible, et la fenêtre duale canonique est de la forme

$$\gamma[\ell] = \frac{1}{M \sum_{n=0}^{N-1} |g[\ell - na]|^2} g[\ell]. \quad (3.13)$$

**Exercice 3.6 — Inversion sans effort de la DGT.** Démontrer la proposition 3.9. On pourra commencer par montrer que

$$\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mathcal{V}_g x[m, n] g_{mn}[k] = \sum_{\ell=0}^{L-1} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[\ell] \bar{g}[\ell - na] g[k - na] e^{2i\pi mb(k-\ell)/L} = M \sum_{n=0}^{N-1} |g[\ell - na]|^2 x[k],$$

puis en déduire que l'opérateur de trame agit par multiplication par le vecteur  $M \sum_{n=0}^{N-1} |g[\ell - na]|^2$ . De là, en déduire l'opérateur inverse, et enfin conclure. ■

### 3.4 Filtrage temps-fréquence, multiplicateurs

On a vu au chapitre précédent que le filtrage peut s'interpréter comme une opération de multiplication point par point dans le domaine fréquentiel. Par exemple, un filtre de réponse impulsionnelle  $h$  peut s'écrire sous la forme

$$K_h x[\ell] = \frac{1}{\sqrt{L}} \sum_{k=0}^{L-1} \hat{h}[k] \hat{x}[k] e^{2i\pi k\ell/L}.$$

Ceci suggère de définir de façon similaire des filtres temps-fréquence, comme opérateurs de multiplication point par point dans le domaine de Gabor.

**Définition 3.5** Soient  $g, h \in \mathbb{C}^L$  deux fenêtres d'analyse et de synthèse, et soit  $\mu \in \mathbb{C}^{M \times N}$  une matrice temps-fréquence, appelée **masque**. Le multiplicateur de Gabor associé est l'opérateur linéaire  $\mathbb{M}$  défini par

$$\mathbb{M}x = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mu[m, n] \mathcal{V}_g x[m, n] h_{mn}.$$

En d'autres termes, un multiplicateur de Gabor agit en trois étapes, de la façon suivante :

- Calcul de la transformée de Gabor  $\mathcal{V}_g x$  de  $x$  avec une fenêtre  $g$
- Multiplication point par point de la transformée de Gabor par le masque  $\mu$ .
- Calcul d'une transformée de Gabor inverse, avec les mêmes paramètres  $a$  et  $M$ , et une fenêtre de synthèse  $h$  qui peut être la fenêtre d'analyse  $g$ , la fenêtre duale canonique  $\gamma$  ou n'importe quelle autre fenêtre.

**Remarque 3.10 — Filtrage temps-fréquence.** Les multiplicateurs sont souvent utilisés pour supprimer, ou atténuer des composantes localisées en temps et en fréquence dans les signaux. On utilise pour cela des masques simples, qui sont souvent proportionnels à l'indicatrice d'une région  $\Omega$  à atténuer ou rehausser

$$\mu(m, n) = \begin{cases} g & \text{si } (m, n) \in \Omega, \\ 1 & \text{sinon} \end{cases}$$

où  $g \in \mathbb{R}_+$  est un gain.

■ **Exemple 3.2** On représente en Figure 3.7 le spectrogramme de quatre notes (plus précisément, trois notes isolées et un accord), ainsi que le spectrogramme d'un signal modifié par un multiplicateur de Gabor. Celui-ci est associé à un masque égal à 1 partout, sauf sur le support du fondamental de la première note. Comme on peut le voir, le fondamental est effectivement très atténué, mais pas complètement. ■

Il est important de signaler une différence importante entre les filtres de convolution classiques, et les multiplicateurs de Gabor, liée au fait que  $\mathcal{V}_g x$  donne en fait une représentation redondante du signal  $x$ . Il est

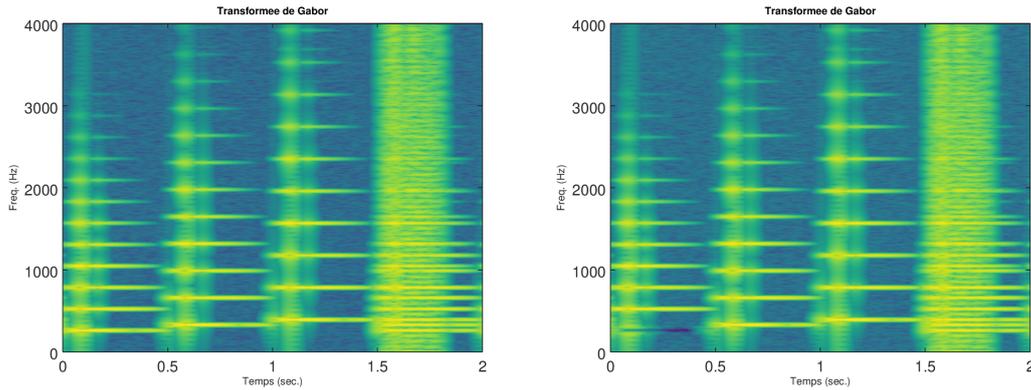


FIGURE 3.7 – Multiplicateur de Gabor

possible de démontrer que n'importe quelle fonction de deux variables n'est pas nécessairement transformée de Gabor d'un signal. Les transformées valides doivent satisfaire une condition de compatibilité, qui prend en fait une forme simple.

Considérons la formule de synthèse de Gabor (3.12), et effectuons le produit scalaire des deux membres avec un atome de Gabor  $g_{m_0 n_0}$ . On obtient ainsi

$$\mathcal{V}_x[m_0, n_0] = \langle x, g_{m_0 n_0} \rangle = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mathcal{V}_x[m, n] \langle \gamma_{mn}, g_{m_0 n_0} \rangle,$$

qui peut s'interpréter sous la forme d'une équation de compatibilité :

$$\mathcal{V}_g x[m_0, n_0] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} K[m_0, n_0; m, n] \mathcal{V}_x[m, n], \quad (3.14)$$

avec  $K[m_0, n_0; m, n] = \langle \gamma_{mn}, g_{m_0 n_0} \rangle$ , ou encore plus formellement

$$\mathcal{V}_g x = P \mathcal{V}_x, \quad (3.15)$$

où  $P$  est la transformation linéaire qui associe à toute matrice  $F \in \mathbb{C}^{M \times N}$  la matrice  $PF \in \mathbb{C}^{M \times N}$  définie par

$$(PF)[m_0, n_0] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} K[m_0, n_0; m, n] F[m, n].$$

Il est possible de démontrer que la transformation  $P$  est égale à  $P = \mathcal{V}_\gamma^* \mathcal{V}_g$ , et qu'elle est en fait le projecteur sur le sous-espace de  $\mathbb{C}^{M \times N}$ , image de  $\mathbb{C}^L$  par  $\mathcal{V}_g$ .

**Exercice 3.7 — Projection sur le sous-espace des représentations temps-fréquence valides.** Avec les notations de cette section, démontrer que  $\Pi = \mathcal{V}_g \mathcal{V}_\gamma^*$  est un projecteur sur le sous-espace des transformées de Gabor (c'est à dire  $\Pi^2 = \Pi$ ). Démontrer que si  $\gamma = g$ , alors le projecteur est orthogonal (c'est à dire  $\Pi^* = \Pi$ ). ■

Dit autrement (et plus simplement) : étant donnée une fenêtre d'analyse

- Toute matrice temps-fréquence  $V \in \mathbb{C}^{M \times N}$  n'est pas transformée de Gabor  $\mathcal{V}_g x$  d'un signal  $x \in \mathbb{R}^L$ .
- Les transformées de Gabor "valides" forment un sous-espace vectoriel de  $\mathbb{C}^{M \times N}$ .
- Étant donnée une matrice temps-fréquence  $V \in \mathbb{C}^{M \times N}$ , elle peut être projetée sur ce sous-espace vectoriel en utilisant le projecteur  $P$ .
- Celui-ci n'est autre que la composition d'une synthèse de Gabor  $\mathcal{V}_\gamma^*$  avec fenêtre  $\gamma$ , suivie d'une transformée de Gabor avec fenêtre  $g$ .

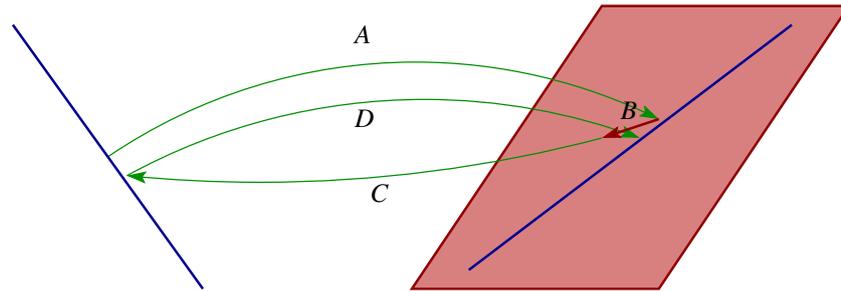


FIGURE 3.8 – Illustration de la contrainte de compatibilité sur une transformée de Gabor. L'espace de départ  $\mathbb{C}^L$  est représenté par la droite oblique à gauche, l'espace d'arrivée (plus grand) est le plan à droite, et l'image de  $\mathbb{C}^L$  par la transformée est représenté par la droite oblique dans le plan. L'opération  $A$  représente la transformation de Gabor  $x \rightarrow \mathcal{V}_g x$  d'un signal  $x \in \mathbb{C}^L$ , l'opération  $B$  est la multiplication par le masque  $\mu$ , le produit  $\mu \mathcal{V}_g x$  n'appartient plus au sous-espace image. L'opération  $C$  est la transformée de Gabor inverse  $\mu \mathcal{V}_g x \rightarrow \mathcal{V}_g^* \mu \mathcal{V}_g x \in \mathbb{C}^L$ . L'opération  $D$  est une transformation de Gabor, qui aboutit dans l'espace image (droite rouge). Ainsi, la combinaison de  $C$  puis  $D$  projette le résultat de  $BA$  sur le sous-espace image.

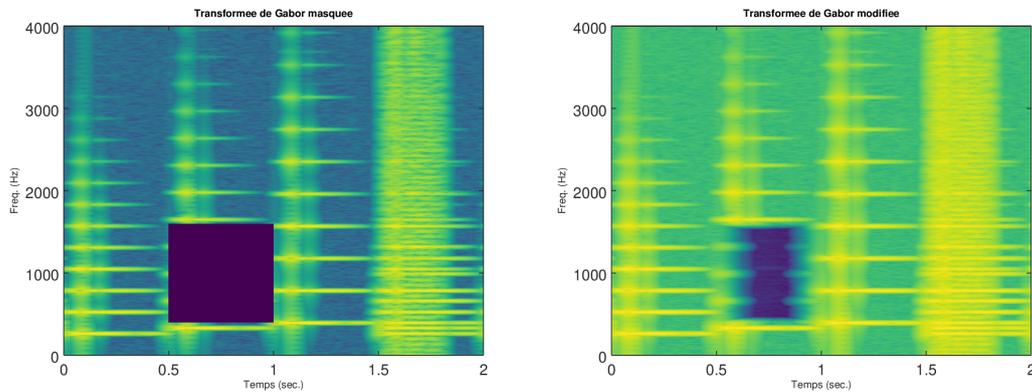


FIGURE 3.9 – Action d'un multiplicateur de Gabor : l'image de gauche représente la transformée de Gabor  $\mathcal{V}_g x$  du signal, multipliée par un masque binaire égal à zéro à l'intérieur d'un rectangle. L'image de droite est la projection de l'image de gauche  $\mu \mathcal{V}_g x$  sur le sous-espace de  $\mathbb{C}^{M \times N}$  sur le sous-espace image. La projection est donnée par l'opérateur  $\mathcal{V}_g^* \mathcal{V}_g$ .

Ceci est schématisé dans la Figure 3.8.

Ceci explique pourquoi, lorsqu'on transforme un signal par multiplicateur de Gabor, le résultat peut être différent de ce que l'on attend. Un exemple se trouve en Figure 3.9. L'image de gauche représente le spectrogramme du son déjà analysé en Figure 3.7, qui a été multiplié par un masque binaire valant 1, sauf à l'intérieur d'un rectangle dans lequel il vaut 0. L'image de droite est le spectrogramme du signal reconstruit à partir de la transformée de Gabor masquée. On voit bien qu'elle est très significativement différente, signe que l'image de gauche n'est pas un spectrogramme valide, la projection sur le sous-espace des spectrogrammes valides l'a fortement modifiée.



**(Multiplicateur de Gabor)** On commence par charger le signal, et fixer les paramètres de la transformation de Gabor.

```
> [accord,fs] = audioread('../Sounds/accord.wav');
> a = 5;
> M = 4000;
> L = 16000;
```

Ensuite, on définit la fenêtre (le paramètre `tfr` est choisi de sorte à assurer une bonne localisation fréquentielle), puis la `dgt` est calculée (en utilisant ici la fonction `dgtreal` car le signal est réel)

```
> tfr = 10;
> g = pgauss(L,tfr);
> V = dgtreal(accord,g,a,M,L);
```

Représentation graphique (avec des unités physiques)

```
> ttt = (0:(L-1))/fs;
> fff = linspace(0,fs/2,size(V,1));
> imagesc(ttt,fff,log(abs(V))); axis xy;
> title('Transformee de Gabor');
> xlabel('Temps (sec.)');
> ylabel('Freq. (Hz)');
> set(gca,'fontsize',16)
```

Définition du masque :

```
> mask = ones(size(V));
> tband = 801:1600;
> fband = 201:800;
> mask(fband,tband) = 0;
> V2 = V.*mask;
> imagesc(ttt,fff,log(abs(V))); axis xy;
> title('Transformee de Gabor masquee');
> xlabel('Temps (sec.)');
> ylabel('Freq. (Hz)');
> set(gca,'fontsize',16)
```

Reconstruction du signal modifié, puis calcul et représentation graphique de sa DGT

```
> gamma = gabdual(g,a,M);
> xmod = idgtreal(V2,gamma,a,M);
> Vmod = dgtreal(xmod,g,a,M,L);
> imagesc(ttt,fff,log(abs(Vmod))); axis xy;
> title('Transformee de Gabor modifiee');
> xlabel('Temps (sec.)');
> ylabel('Freq. (Hz)');
> set(gca,'fontsize',16)
```

### 3.5 Transformations temps-fréquence non-standard, ondelettes

Comme on l'a vu, la transformation de Gabor discrète peut s'écrire comme une transformation linéaire associant à tout signal la famille de ses produits scalaires avec des *atomes de Gabor*, qui sont des formes d'onde auxquelles sont associées des temps et fréquences de référence. La caractéristique de la transformation de Gabor est que ces atomes temps-fréquences sont construits de façon très simple, comme copies translâtées et modulées (donc translâtées en fréquence) d'une fenêtre de référence. On peut en particulier noter que

- Ces atomes sont de longueur constante, de même que leur transformée de Fourier (la translation ne modifie pas la longueur d'un vecteur). Par longueur on entend ici soit la longueur du support si celui-ci est fini (ou inférieur à la longueur  $L$  des signaux dans le cas de dimension finie).
- L'échantillonnage est un échantillonnage régulier : le plan temps-fréquence échantillonné est de la forme  $b\mathbb{Z} \times a\mathbb{Z}$  dans le cas de dimension infinie, ou  $b\mathbb{Z}_M \times a\mathbb{Z}_N$  en dimension finie.

Dans certaines situations, il est possible de s'affranchir de ces règles, tout en conservant les propriétés essentielles, à savoir la linéarité de la transformation et l'existence de formules d'inversion.

#### 3.5.1 Transformation de Gabor non-stationnaire

Dans certaines situations, il peut être intéressant de faire varier la taille de la fenêtre en fonction du temps. Par exemple, si un son comporte des phases "transitoires" (par exemple des sons rythmiques comme la batterie), on peut avoir envie d'utiliser des fenêtres assez courtes et privilégier ainsi la résolution temporelle. Inversement, dans le cas de phases "tonales" (par exemple des notes tenues), on privilégiera plutôt des fenêtres bien localisées en fréquence, au détriment de la résolution temporelle. De façon générale, on utilisera alors des fenêtres  $g_n[\ell]$  plutôt que les copies translâtées  $g[\ell - na]$  d'une unique fenêtre  $g$ . On définit de la sorte une transformation de Gabor non-stationnaire comme précédemment, par

$$\mathcal{V}_g x[m,n] = \langle x, g_{mn} \rangle, \quad \text{où} \quad g_{mn}[\ell] = e^{2i\pi m b \ell} g_n[\ell]. \quad (3.16)$$

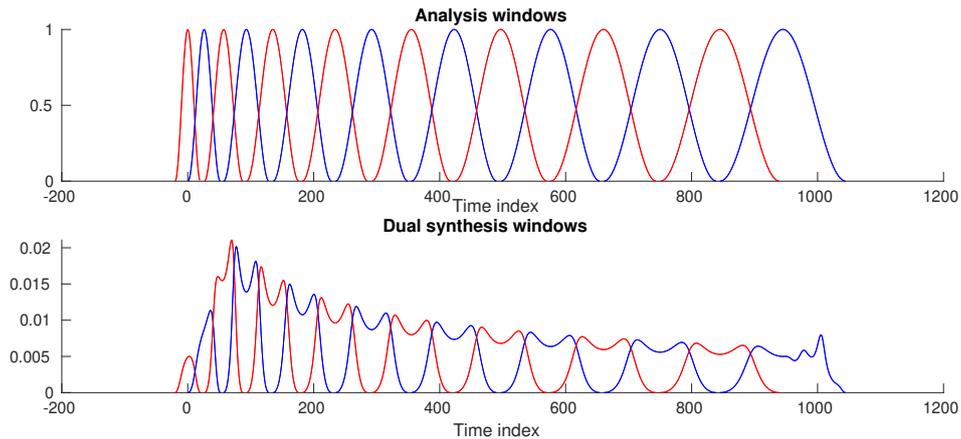


FIGURE 3.10 – Un exemple de fenêtres d’analyse et de de synthèse variant au cours du temps pour la transformation de Gabor non-stationnaire

Ici, on a traité indifféremment les cas de dimension infinie ( $m, n, \ell \in \mathbb{Z}$ ) et infinie  $\ell \in \mathbb{Z}_\ell$ ,  $m \in \mathbb{Z}_M$  et  $n \in \mathbb{Z}_N$ ). On a aussi noté  $\mathbf{g} = \{g_n\}_n$  (avec un caractère gras) la famille de fenêtres et  $\mathcal{V}_{\mathbf{g}}$  la transformation de Gabor non-stationnaire correspondante.

**Remarque 3.11 — Convention de phase.** On a utilisé ici une convention de phase différente de la convention de phase utilisée dans la DGT classique, plus précisément la convention décrite dans la Remarque 3.3.

Comme dans le cas plus classique, une question centrale est celle de l’inversibilité de cette transformation, qui est liée à l’inversibilité de l’opérateur de trame  $\mathcal{R}$  défini par

$$\mathcal{R}x = \sum_{m,n} \langle x, g_{mn} \rangle g_{mn} . \quad (3.17)$$

L’étude de cet opérateur est complexe, on n’entrera pas ici dans les détails. cependant, il existe un cas de figure où l’analyse se simplifie. En effet, si les fenêtres  $g$  sont toutes à support fini, et si le pas d’échantillonnage fréquentiel  $b$  est suffisamment grand par rapport aux tailles des fenêtres, de sorte que la condition ci-dessous soit satisfaite pour tout  $n$

$$g_n[\ell] \bar{g}_n[\ell - k/a] = 0 \quad \forall \ell \quad \text{et} \quad \forall k \neq 0 , \quad (3.18)$$

alors l’opérateur de trame est inversible. De plus, la transformation inverse s’écrit

$$x = \sum_{m,n} \mathcal{V}_{\mathbf{g}} x[m, n] \gamma_{mn} , \quad (3.19)$$

où les atomes temps-fréquence  $\gamma_{mn} = \mathcal{R}^{-1} g_{mn}$  sont obtenus par modulation de fréquence de fenêtres  $g_n$ , appelés *fenêtres duales canoniques*. Dans le cas de signaux de longueur finie  $L$ , on a :

$$\gamma_{mn}[\ell] = e^{2i\pi mb\ell/L} \gamma_n[\ell] , \quad \text{avec} \quad \gamma_n[\ell] = \frac{g_n[\ell]}{\sum_k |g_k[\ell]|^2} \quad (3.20)$$

Cette expression est à rapprocher de l’expression de la fenêtre duale canonique "sans effort" donnée en (3.13).

Un exemple de fenêtres variant au cours du temps est donné en FIGURE 3.10 : fenêtres d’analyse et fenêtres de synthèse. Dans ce cas, la largeur temporelle est fonction croissante du temps.



#### (Transformation de Gabor non-stationnaire)

La toolbox LTFAT inclut des fonctions permettant le calcul de transformation de Gabor non-stationnaire, notamment les fonctions `nsdgt` et `insdgt`. La syntaxe est un peu complexe, car elle nécessite de passer non plus une fenêtre mais une famille de fenêtres, stockées dans un tableau de cellules.

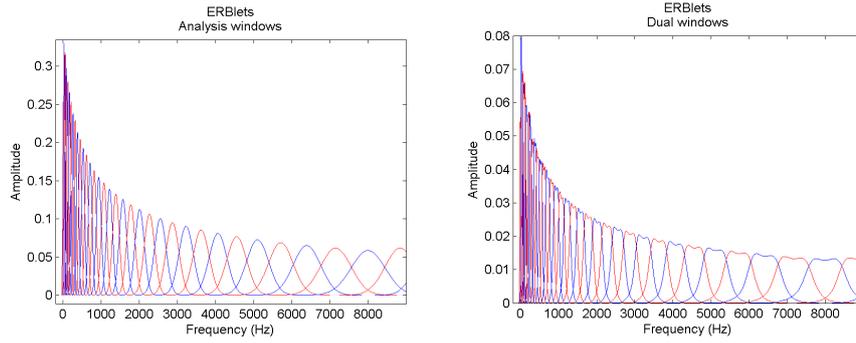


FIGURE 3.11 – ERBlets : réponses fréquentielles des fenêtres d’analyse et de synthèse

### 3.5.2 Largeur de bande variable en fréquence, transformée à $Q$ constant, ondelettes Transformation de Gabor non-stationnaire en fréquence

Avec une construction similaire, on peut construire des familles de fenêtre dont la résolution fréquentielle varie en fonction de la fréquence (plutôt que du temps) : on introduit les atomes temps-fréquence

$$g_{mn}[\ell] = g_m[\ell - na], \quad (3.21)$$

où les  $g_m$  sont des fenêtres dont la transformée de Fourier est centrée sur une fréquence  $\nu_m$ . La synthèse est possible, grâce à des fenêtres de synthèse  $\gamma_m$ , sous certaines hypothèses : les transformées de Fourier des fenêtres  $g_m$  doivent être à support fini et  $a$  doit être suffisamment petit pour que pour tout  $k$ ,  $\sum_m |\hat{g}_m[k]|^2$  soit borné inférieurement par une constante strictement positive. On introduit alors des fenêtres de synthèse  $\gamma_m$ , définies par leur transformée de Fourier

$$\hat{\gamma}_m[k] = \frac{\hat{g}_m[k]}{\sum_j |\hat{g}_m[j]|^2} \quad (3.22)$$

et on obtient les formules d’analyse-synthèse

$$x[\ell] = \sum_{m,n} \mathcal{V}_{\mathbf{g}} x[m,n] \gamma_m[\ell - na], \quad \mathcal{V}_{\mathbf{g}} x[m,n] = \langle x, g_{mn} \rangle. \quad (3.23)$$

Parmi les exemples qui ont été utilisés dans un contexte audio, on peut mentionner notamment

- Fenêtres à  $Q$  constant (voir plus loin les ondelettes) : ce sont des fenêtres dont la largeur de bande  $\sigma_m$  est proportionnelle à leur fréquence centrale  $\nu_m$ . On a donc pour tout  $j$  la propriété de largeur de bande relative constante

$$Q = \frac{\nu_j}{\sigma_j} = \frac{\nu_0}{\sigma_0}.$$

- Erblets : filtres dont les réponses fréquentielles suivent l’échelle ERB (*Equivalent Rectangular Bandwidth*), qui approxime les largeurs de bande des filtres auditifs

$$\text{ERB}(f) = 6.23f^2 + 93.39f + 28.52.$$

Des exemples de fenêtres correspondantes sont représentés en Figure 3.11.

#### Ondelettes

Dans le cas de la transformation en ondelettes, la largeur de bande des filtres est proportionnelle à leur fréquence centrale. Dans le cas de signaux de longueur infinie, on introduit donc un banc de filtres  $\{\psi_j, j = 1, \dots, \infty\}$ , où  $s$  est un paramètre que l’on appelle généralement paramètre d’échelle. La transformation correspondante est notée  $\mathcal{W}_\psi$

$$\mathcal{W}_\psi x[j,n] = (x * \tilde{\psi}_j)[na_j],$$

où  $a_j$  est un pas d'échantillonnage temporel, qui peut dépendre ou pas du paramètre  $j$ , et où comme précédemment  $\widehat{\psi}_j[\ell] = \widehat{\psi}_j[-n]$ . Dans la version la plus simple,  $j = 1$  correspond à une échelle égale à 2 échantillons,  $j = 2$  à une échelle de 4 échantillons, et plus généralement  $j$  est associé à une échelle de  $2^j$  échantillons. Les réponses fréquentielles des filtres sont des copies dilatées les unes des autres, c'est à dire que l'on a essentiellement

$$\widehat{\psi}_j(\nu) = 2^{j/2} \widehat{\psi}(2^j \nu [\text{mod } 1]) \quad (3.24)$$

(le  $[\text{mod } 1]$ , lire "modulo 1", est là pour tenir compte de la périodicité du domaine de Fourier dans ce cas, voir plus loin pour davantage de précisions). Le filtre  $\psi$  est appelé "ondelette".

Si on oublie ce terme  $[\text{mod } 1]$  pour un instant, et si on suppose que  $|\widehat{\psi}|$  est maximum pour une fréquence donnée, disons  $\nu_0 = \pm 1/2$ , alors  $|\widehat{\psi}_j|$  est quant à lui maximum pour une fréquence  $\nu_j$  telle que  $2^j \nu_j = \nu_0$ , soit  $\nu_j = 2^{-j} \nu_0$ . Comme par ailleurs on voit que  $\widehat{\psi}_j$  est une copie de  $\widehat{\psi}$  contractée d'un facteur  $2^j$ , sa largeur  $\sigma_j$  (quelle que soit la façon dont on la mesure) est multiple de la largeur de bande  $\sigma$  de  $\psi$  :  $\sigma_j = 2^{-j} \sigma$ , et est donc proportionnelle à la fréquence centrale du filtre. On dit que les ondelettes effectuent une analyse à  $Q$  constant.

L'échantillonnage en échelles (qui est aussi un échantillonnage fréquentiel, la fréquence étant définie comme l'inverse de l'échelle, à une constante multiplicative près) est généralement un échantillonnage suivant les puissances de 2, comme vu plus haut. Deux cas de figure sont couramment utilisés concernant l'échantillonnage temporel

- Échantillonnage régulier  $a_j = a_0$  (par exemple  $a_j = 1$  pour tout  $j$ ).
- Échantillonnage dépendant de l'échelle  $j$

On n'entrera pas ici dans les détails de la théorie. Il est possible de ; montrer que dans ces deux cas de figure, si les paramètres (en l'occurrence l'ondelette  $\psi$  et les pas d'échantillonnage) sont bien choisis, la transformation

$$x \longmapsto \mathcal{W}_\psi x$$

est inversible, dans un cadre similaire au cadre rencontré dans l'analyse de Gabor. Les coefficients d'ondelettes  $\mathcal{W}_\psi x[j, n]$  d'un signal  $x$  en donnent donc une représentation alternative, à partir de laquelle un certain nombre d'opérations peuvent être effectuées.

Il est à noter que la transformation en ondelettes constitue un cas particulier de transformées dites à  $Q$  constant. Toutes ne sont pas générées en utilisant des dilatations comme les ondelettes, mais conservent cette propriété de bancs de filtres dont la largeur de bande est proportionnelle à la fréquence centrale. Des transformées à  $Q$  constant peuvent aussi être construites par transformation de Gabor non-stationnaire.





## 4. Stationnarité "par morceaux"

On va ici s'intéresser à un cas particulier d'extraction d'information à partir d'une représentation temps-fréquence, dans un cadre non-stationnaire. On dit parfois que la non-stationnarité est une non-propriété, en ce sens qu'il existe de nombreuses façons d'être non-stationnaire. Par exemple, un son de glissando de violon est non-stationnaire, on peut lui associer une fréquence qui varie au cours du temps, cette fréquence variant continûment (et lentement par rapport aux oscillations du signal), comme par exemple dans le glissando en spirale de Risset de la Figure 3.5. Il existe des techniques permettant de capturer ce type de comportement.

On va plutôt supposer ici une autre forme particulière de non-stationnarité, que l'on peut qualifier de stationnarité "par morceaux", au sens suivant. On imagine que le signal est constitué de "parties" caractérisées par un contenu spectral et apparaissant à des instants bien déterminés, avec un déroulement temporel simple. On pense immédiatement à des notes de musique d'instruments. La question posée est la suivante : comment déterminer de tels événements à partir d'un signal. Le spectrogramme (à condition qu'il soit convenablement choisi) va nous fournir une représentation adaptée, et on va alors se focaliser sur le post-traitement du spectrogramme.

### 4.1 Retour sur le calcul matriciel

On va ici traiter le spectrogramme (ou plutôt la partie limitée aux fréquences positives) comme une matrice  $M \times N$ , et la traiter comme telle, en se basant sur une ré-interprétation du produit matriciel. On rappelle que le produit  $S = WH$  d'une matrice  $W$  de taille  $M \times K$  par une matrice  $H$  de taille  $K \times N$  est bien défini et produit une matrice  $M \times N$ . L'élément de matrice générique s'écrit

$$S = WH : \quad S_{ij} = \sum_{k=1}^K W_{ik} H_{kj}, \quad (4.1)$$

c'est à dire comme une somme de termes, dont chacun peut lui-même être interprété comme des éléments de la matrice

$$S^{(k)} \in \mathbb{R}^{M \times N} : \quad S_{ij}^{(k)} = W_{ik} H_{kj}. \quad (4.2)$$

$S^{(k)}$  est donc le produit de la  $k$ -ième colonne de  $W$  par la  $k$ -ième ligne de  $H$ . Pour illustrer ceci, on représente en Figure 4.1 le produit d'une matrice à 3 colonnes par une matrice à trois lignes (le produit est donc bien défini).

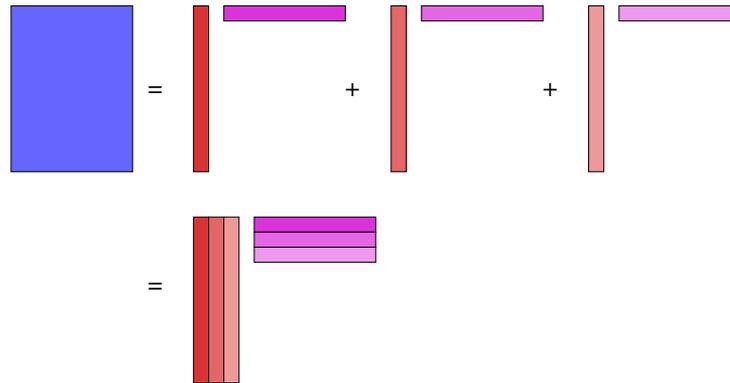


FIGURE 4.1 – Factorisation d'une matrice de rang 3

Supposons maintenant que  $S$  soit la matrice d'un spectrogramme. Les lignes représentent les fréquences, les colonnes représentent les instants. Ainsi, on peut interpréter la  $k$ -ième colonne de  $W$  comme le profil fréquentiel d'une  $k$ -ième "composante", et la  $k$ -ième ligne de  $H$  comme un profil temporel correspondant.

Si chacune de ces composantes représente une note, la colonne correspondante de  $W$  représenterait son spectre, et la ligne de  $H$  représenterait les activations temporelles de cette note. La question est : comment obtenir de telles composantes qui puissent effectivement représenter des notes ?

## 4.2 Algorithmes pour la factorisation d'un spectrogramme

### 4.2.1 Généralités

Depuis une quinzaine d'années, des algorithmes de factorisation matricielle ont été développés. Le plus simple et le plus connu est la décomposition en valeurs singulières (SVD, voir l'article correspondant sur l'encyclopédie en ligne Wikipedia [10]). On sait que toute matrice réelle symétrique peut être diagonalisée par une base orthonormée de vecteurs propres. On peut voir la décomposition en valeurs singulières comme une généralisation à des matrices arbitraires, pas nécessairement carrées.

Dans ce qui suit, on note génériquement  $A^*$  la matrice adjointe de la matrice  $A$ , c'est à dire la complexe conjuguée de la transposée de  $A$ . On rappelle qu'une matrice carrée  $U$  de taille  $M$  est une matrice unitaire si et seulement  $U^*U = UU^* = I_M$ , dans ce cas les colonnes de cette matrice forment une base orthonormée de  $\mathbb{C}^M$ . On se limitera ici au cas réel, les matrices unitaires sont alors appelées matrices orthogonales, et sont telles que  $A^T A = AA^T = I_M$ .

**Théorème 4.1 — Décomposition en valeurs singulières.** Soit  $A$  une matrice  $M \times N$  à coefficients réels. Alors il existe une factorisation de  $A$  sous la forme :

$$A = U\Sigma V^*,$$

où  $U$  une matrice unitaire  $M \times M$ ,  $\Sigma$  est une matrice  $M \times N$  dont les coefficients diagonaux sont des réels positifs ou nuls et tous les autres sont nuls, et  $V^*$  est la matrice adjointe à  $V$ , matrice unitaire  $N \times N$ .

- Les colonnes de la matrice  $V$  forment une base orthonormée de  $\mathbb{R}^N$  :  $V^*V = VV^* = I_N$
- Les colonnes de la matrice  $U$  forment une base orthonormée de  $\mathbb{R}^M$  :  $U^*U = UU^* = I_M$
- Les éléments diagonaux de la matrice  $\Sigma$  sont appelés *valeurs singulières* de la matrice  $A$ . Ils sont généralement triés par ordre descendant.

La SVD est implémentée dans tous les paquets logiciels de calcul scientifique. Cependant, elle ne donne pas de résultat satisfaisant pour le problème posé ici. Les colonnes des matrices  $U$  et  $V$  s'avère ne pas avoir de pertinence en termes d'interprétation. De plus, une SVD appliquée au spectrogramme (qui est positif ou nul) ne conduit pas nécessairement à des matrices  $U$  et  $V$  positives ou nulles. On verra cependant que la diagonale de la matrice  $\Sigma$ , qui contient les valeurs singulières, peut donner des informations pertinentes, sur la valeur de  $K$  en particulier.

### 4.2.2 Factorisation en matrices non-négatives

Une alternative est donnée par les algorithmes de factorisation en matrices non-négatives. Étant donnée la matrice  $S \in \mathbb{R}_+^{M \times N}$  d'un spectrogramme, et une valeur fixée du rang  $K$ , il s'agit de trouver deux matrices non-négatives  $W \in \mathbb{R}_+^{M \times K}$  et  $H \in \mathbb{R}_+^{K \times N}$ , telles que

$$S \approx WH. \quad (4.3)$$

Pour cela, on procède comme suit : on se donne une fonction  $D(S|W,H)$ , appelée divergence donnant une mesure de dissimilarité entre  $S$  et le produit  $WH$ , et on minimise celle-ci par rapport à  $W$  et  $H$ . Les algorithmes les plus classiques sont des algorithmes alternés, qui alternent

- mise à jour de  $W$  à  $H$  fixé
- mise à jour de  $H$  à  $W$  fixé

jusqu'à convergence (éventuelle), avec éventuellement d'autres opérations (normalisation, projection,...). On en verra quelques exemples ci-dessous.

Ces algorithmes nécessitent de choisir une initialisation et de spécifier un critère d'arrêt. Il importe aussi de résoudre certaines indéterminations, ces trois aspects sont discutés ci-dessous.

- *Initialisation* : plusieurs choix sont possibles ; un choix classique est de fournir une matrice  $H$  et une matrice  $W$  aléatoires (non-négatives), on peut alors obtenir des évaluations de performances en moyennant sur plusieurs réalisations. On utilisera ici une autre stratégie, basée sur la SVD. Partant de la SVD du spectrogramme, on choisit tout d'abord le rang  $K$  de la décomposition à partir du graphe des valeurs singulières. Ensuite, on peut choisir  $W_{\text{init}}$  comme la valeur absolue des  $K$  premières colonnes de la matrice  $U$ , et  $H_{\text{init}}$  comme la valeur absolue des  $K$  premières lignes de  $\Sigma V^*$ .
- *Critère de convergence* : il s'agit de décider quand stopper les itérations. La solution la plus simple est de donner un nombre maximum d'itérations `ITER_MAX` (ce qu'il faut faire de toutes façons) mais ça n'est pas satisfaisant. Il est préférable de se donner un critère qui décrive l'évolution soit de la fonction objective  $D$  soit des matrices  $W$  et/ou  $H$ . On peut par exemple calculer, à l'itération  $i$ , la quantité

$$\text{Crit}_D(i) = \left| D(S|W^{(i)}, H^{(i)}) - D(S|W^{(i-1)}, H^{(i-1)}) \right|$$

où  $W^{(i)}$  et  $H^{(i)}$  sont les matrices obtenues à l'itération  $i$ , ou une version normalisée, ou encore

$$\text{Crit}_W(i) = \left| \frac{\|W^{(i)} - W^{(i-1)}\|_F}{\|W^{(i)}\|_F} \right|, \quad \text{Crit}_H(i) = \left| \frac{\|H^{(i)} - H^{(i-1)}\|_F}{\|H^{(i)}\|_F} \right|$$

où  $\|\cdot\|_F$  est la norme de Frobenius.

- *Indéterminations : normalisation et ordre* : Étant donnée une solution  $S \approx WH$ , il existe une infinité d'autres solutions équivalentes. En effet, étant donnée n'importe quelle matrice carrée inversible  $A \in \mathcal{M}_K$ ,  $WH = WAA^{-1}H$ , de sorte que  $(WA, A^{-1}H)$  est une autre solution équivalente. En général les matrices  $WA$  et  $A^{-1}H$  ne sont pas non-négatives, donc ces solutions ne sont pas admissibles. Cependant,  $WA$  et  $A^{-1}H$  sont non-négatives dans deux cas particuliers :

1. Si  $A$  est une matrice diagonale : dans ce cas  $WA$  est une matrice équivalente à  $W$ , qui ne diffère que par la normalisation des colonnes. De même,  $A^{-1}H$  ne diffère de  $H$  que par une constante de normalisation sur les lignes. Cette indétermination peut donc être levée en imposant une condition de normalisation, soit sur les lignes de  $H$  soit sur les colonnes de  $W$ . On peut par exemple imposer

$$\sum_m W_{mn} = 1, \quad \forall n$$

2. Si  $A$  est une matrice de permutation (toutes les lignes et les colonnes contiennent un et un seul élément non-nul, égal à 1). Dans ce cas, l'action de  $A$  est de charger l'ordre des lignes de  $H$  et des colonnes de  $W$ . Cette indétermination peut être levée si nécessaire en ordonnant ces lignes et colonnes après factorisation, suivant un critère à déterminer.

**Choix de la divergence**  $D(S|W, H)$ 

Comme mentionné plus haut, la factorisation est obtenue en cherchant les matrices  $W$  et  $H$  qui minimisent une divergence  $D(S|W, H)$ , qui mesure l'écart entre le spectrogramme  $S$  et le modèle  $WH$ . Il existe plusieurs choix standard pour ces divergences, parmi lesquels les trois les plus usités sont les suivants :

1. La divergence Euclidienne :

$$D_2(S|W, H) = \frac{1}{2} \|S - WH\|_F^2 = \frac{1}{2} \sum_{m=1}^M \sum_{n=1}^N [S_{mn} - (WH)_{mn}]^2 \quad (4.4)$$

où  $\|A\|_F$  désigne la norme de Frobenius d'une matrice  $A$ , i.e. l'équivalent matriciel de la norme Euclidienne (racine carrée de la somme des carrés des éléments de matrice).

2. La divergence de Kullback-Leibler généralisée

$$D_1(S|W, H) = \sum_{m=1}^M \sum_{n=1}^N \left[ S_{mn} \log \left( \frac{S_{mn}}{(WH)_{mn}} \right) - S_{mn} + (WH)_{mn} \right] \quad (4.5)$$

3. La divergence de d'Itakura-Saito

$$D_0(S|W, H) = \sum_{m=1}^M \sum_{n=1}^N \left[ \frac{S_{mn}}{(WH)_{mn}} - \log \left( \frac{S_{mn}}{(WH)_{mn}} \right) - 1 \right] \quad (4.6)$$

Ces divergences font partie d'une famille plus vaste, appelées  $\beta$ -divergences  $D_\beta$ , qu'on ne discutera pas ici.

**Approches algorithmiques**

En restant dans la classe des méthodes alternées, on peut mentionner plusieurs familles d'algorithmes, qui vont différer par les règles de mise à jour de  $W$  et  $H$ , notées  $MAJ_W$  et  $MAJ_H$ .

1. Lorsque c'est possible, résolution explicite des problèmes de minimisation par rapport à  $W$  ( $H$  fixé) et  $H$  ( $W$  fixé). Un exemple est l'approche des moindres carrés alternés, pour la divergence  $D_2$ . Dans ce cas, la solution explicite obtenue n'est pas non-négative, et doit être projetée sur les matrices non-négatives.
2. Mise à jour par descente de gradient projeté.
3. Mises à jour multiplicatives.

**Algorithme générique**

**Données :**  $S$  (spectrogramme); iter\_max;  $\varepsilon$ ;  $W_{init}$ ;  $H_{init}$ ;

**Résultat :** Factorisation en matrices non-négatives

**Initialisation :**  $W^{(0)} = W_{init}$ ,  $k = 0$ ,  $H^{(0)} = H_{init}$ ;

**tant que**  $i \leq \text{iter\_max}$  **et**  $\text{Crit}(i) > \varepsilon$  **faire**

Mise à jour de  $W$  :  $W^{(i+1)} = MAJ_W(W^{(i)}, H^{(i)})$ ;  
Mise à jour de  $H$  :  $H^{(i+1)} = MAJ_H(W^{(i+1)}, H^{(i)})$ ;  
Optionnel : normalisation de  $W$  et/ou  $H$ ;  
Évaluation du critère  $\text{Crit}(i)$ ;  
Évaluation et stockage de la fonction objectif;  
 $i = i + 1$ ;

**fin**

**Algorithme 1 :** Structure générique de l'algorithme de mises à jour alternées (le choix de commencer par une mise à jour de  $W$  est arbitraire).

**4.3 Un exemple**

Pour illustrer le fonctionnement de ces méthodes, on considère l'exemple d'un son correspondant aux deux premières mesures de la partition donnée en Figure 4.2.

Le spectrogramme utilisé est représenté en Figure 4.3. L'image de gauche représente le spectrogramme complet, en échelle logarithmique (la couleur est proportionnelle au logarithme du spectrogramme, pour plus de visibilité). L'image de droite représente la partie "basse-fréquence" sur laquelle sera effectuée la factorisation, en échelle linéaire.



FIGURE 4.2 – Quatre mesures de piano.

**(Factorisation d'une cadence de piano)**

Lecture du fichier, et passage à un signal plus court (en particulier la longueur  $L$  choisie possède suffisamment de diviseurs), en mono :

```
> [cadence,fs] = audioread('Cadence.wav');
> cad = cadence(1:220000,1);
```

Transformation de Gabor ( $a$  et  $M$  sont tous deux diviseurs de  $L$ ). Le paramètre `tfr` règle la résolution temps-fréquence de la fenêtre. S'il est petit la fenêtre est bien localisée fréquentiellement, si il est grand elle est bien localisée temporellement. Un bon choix de ce paramètre est important pour les performances de la factorisation.

```
> L = length(cad)
> a = 100;
> M = 5000;
> tfr = 50;
> g = pgauss(L,tfr);
> Vg = dgtreal(cad,g,a,M,L);
> [M1,N] = size(Vg);
```

Représentation graphique (transformée complète, échelle logarithmique)

```
> ttt = linspace(0,L/fs,N);
> fff = linspace(0,fs/2,M1);
> imagesc(ttt,fff/1000,log(abs(Vg))); axis xy;
> xlabel('Temps (sec.)','fontsize',16);
> ylabel('Frequence (kHz)','fontsize',16);
> set(gca,'fontsize',16);
```

Représentation graphique (transformée réduite aux basses fréquences, échelle linéaire)

```
> Mmax = 500
> imagesc(ttt,fff(1:Mmax)/1000,abs(Vg(1:Mmax,:))); axis xy;
> xlabel('Temps (sec.)','fontsize',16);
> ylabel('Frequence (kHz)','fontsize',16);
> set(gca,'fontsize',16);
```

On effectue une factorisation en matrices non-négatives sur le spectrogramme réduit (image de droite). Pour commencer, une décomposition en valeurs singulières permet d'obtenir les valeurs singulières. On voit sur le graphe que celles-ci décroissent rapidement. On va choisir de travailler en dimension  $K = 9$ , qui semble correspondre à un coude sur le diagramme de la Figure 4.4.

**(SVD sur le spectrogramme réduit)**

```
> [M1,N] = size(Vg);
> aVg_red = abs(Vg(1:Mmax,:));
> [A,S,B] = svd(aVg_red);
> S = diag(S);
> bar(S(1:25)); axis tight
> xlabel('Valeurs singulieres','fontsize',16);
> axis tight;
> set(gca,'fontsize',16);
```

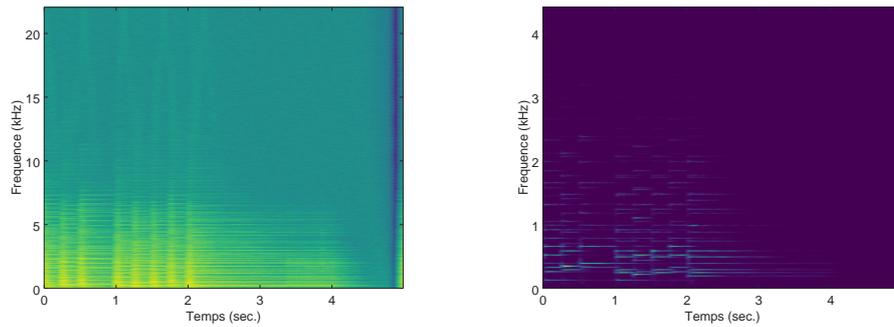


FIGURE 4.3 – Spectrogrammes du son de piano correspondant aux deux premières mesures de la Figure 4.2. Gauche, logarithme du spectrogramme complet ; droite, spectrogramme de la partie "basse-fréquence", en échelle linéaire.

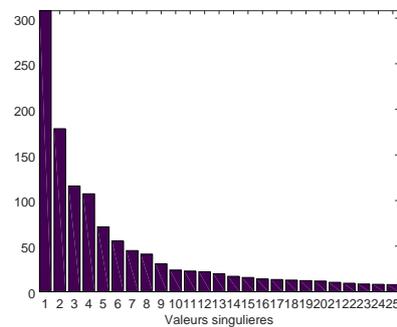


FIGURE 4.4 – SVD sur le spectrogramme réduit : premières valeurs singulières

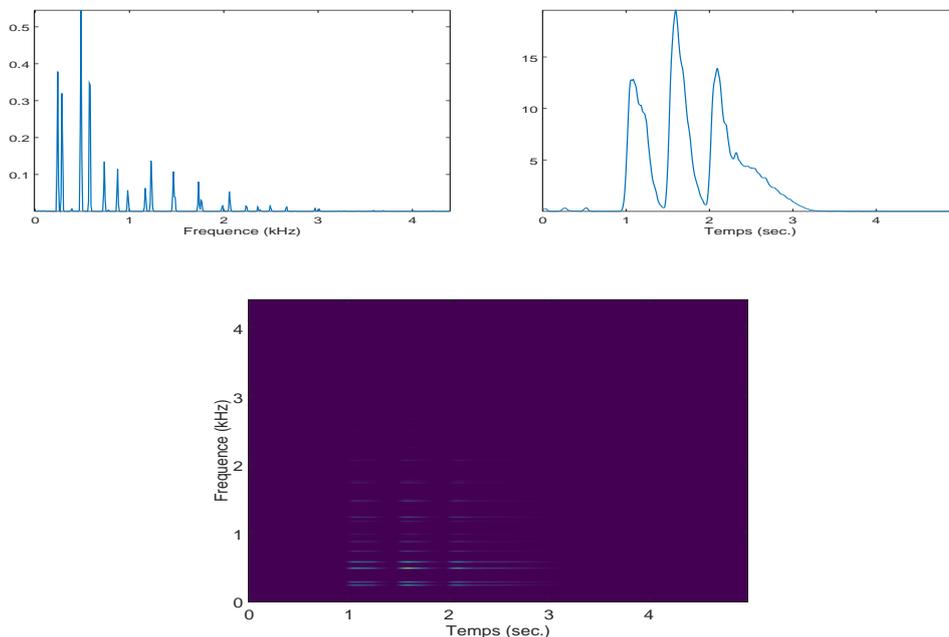


FIGURE 4.5 – Première composante trouvée par la NMF

On effectue ensuite une factorisation en matrices non-négatives, en recherchant  $K = 9$  composantes (ce choix n'est pas tout à fait arbitraire, car motivé par le résultat de la décomposition en valeurs singulières, mais peut être remis en question, en pratique il est souvent bon de faire plusieurs tests).



#### (Factorisation en matrices non-négatives)

On utilise ici un algorithme développé par Févotte, Bertin et Durrieu [2], basé sur la minimisation de la divergence d'Itakura-Saito, par un algorithme multiplicatif, dont le code matlab (fonction `is_nmf_mu`) est fourni en ligne

```
> K = 9;
> W_init = abs(A(:,1:K));
> H_init = abs(B(:,1:K))';
> n_iter = 50;
> [W, H, cost] = is_nmf_mu(aVg_red, n_iter, W_init, H_init);
```

On représente dans les Figures 4.5 à 4.11 les différentes composantes entrant dans la factorisation. Plus précisément, on représente pour certaines valeurs de  $k$  les profils fréquentiel  $W(:,k)$  et temporel  $H(k,:)$ , ainsi que le produit  $W(:,k)H(k,:)$ , qui est la contribution de la composante  $k$  au spectrogramme.

En comparant à la portée de la Figure 4.2, on voit qu'il est possible de faire les identifications suivantes :

- la première composante (Figure 4.5) peut être identifiée à l'accord "si-ré", qui apparaît trois fois (la dernière étant tenue plus longtemps).
- la seconde composante (Figure 4.6) peut être identifiée à l'accord "do-mi" qui débute la cadence et se répète une fois.
- la troisième composante (Figure 4.7) peut être identifiée à l'accord "mi-sol" qui conclut la première mesure, et n'est pas répété.
- les suivantes peuvent être identifiées de façon similaire
- la septième composante (Figure 4.11) est de nature différente, elle est associée aux attaques des différents accords.

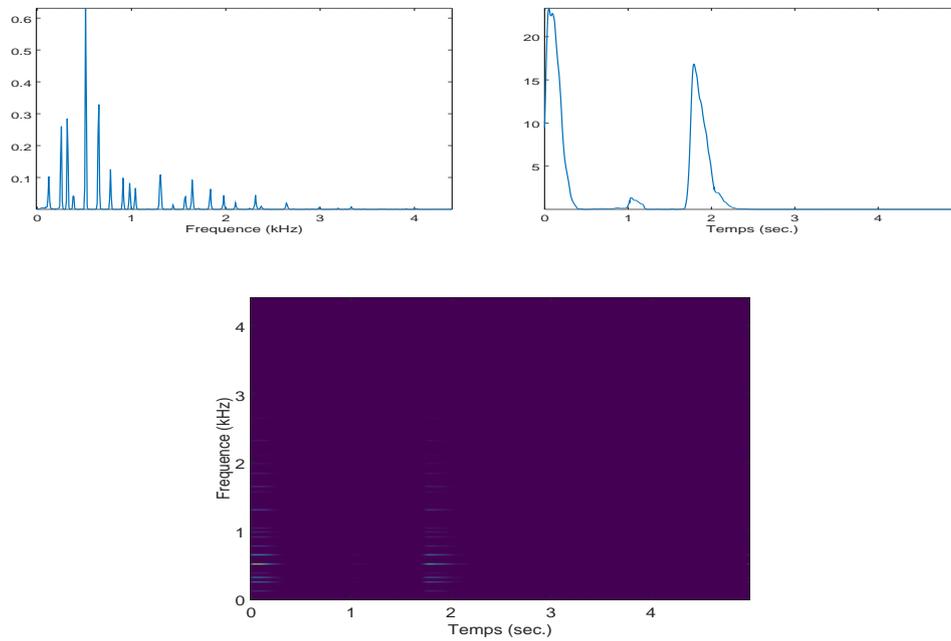


FIGURE 4.6 – Deuxième composante trouvée par la NMF

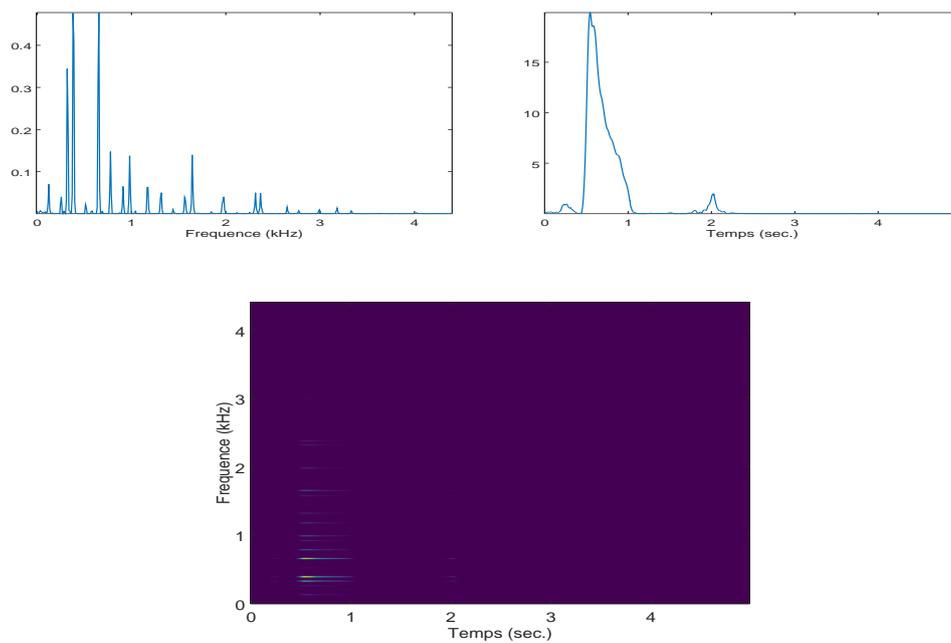


FIGURE 4.7 – Troisième composante trouvée par la NMF

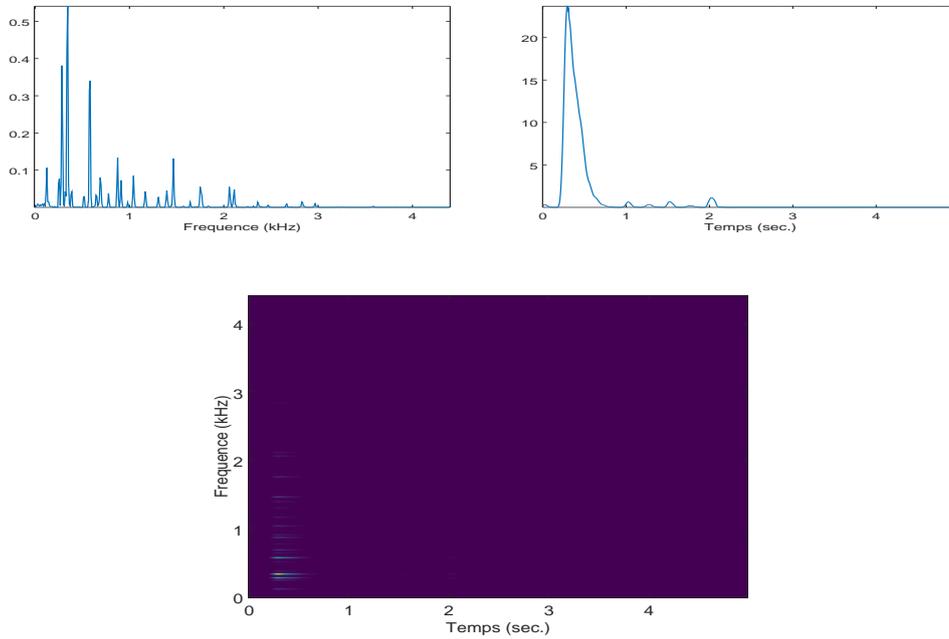


FIGURE 4.8 – Quatrième composante trouvée par la NMF

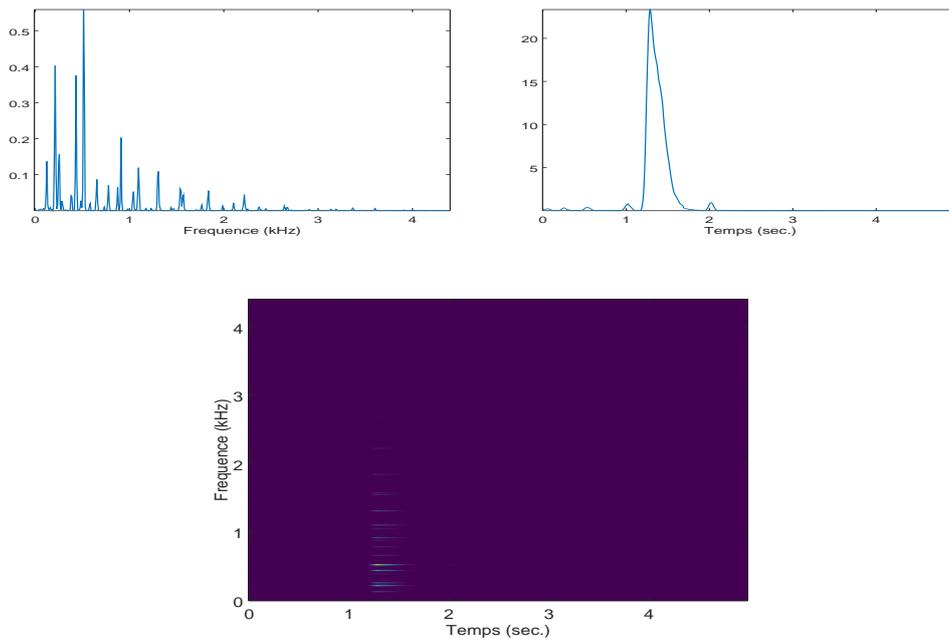


FIGURE 4.9 – Cinquième composante trouvée par la NMF

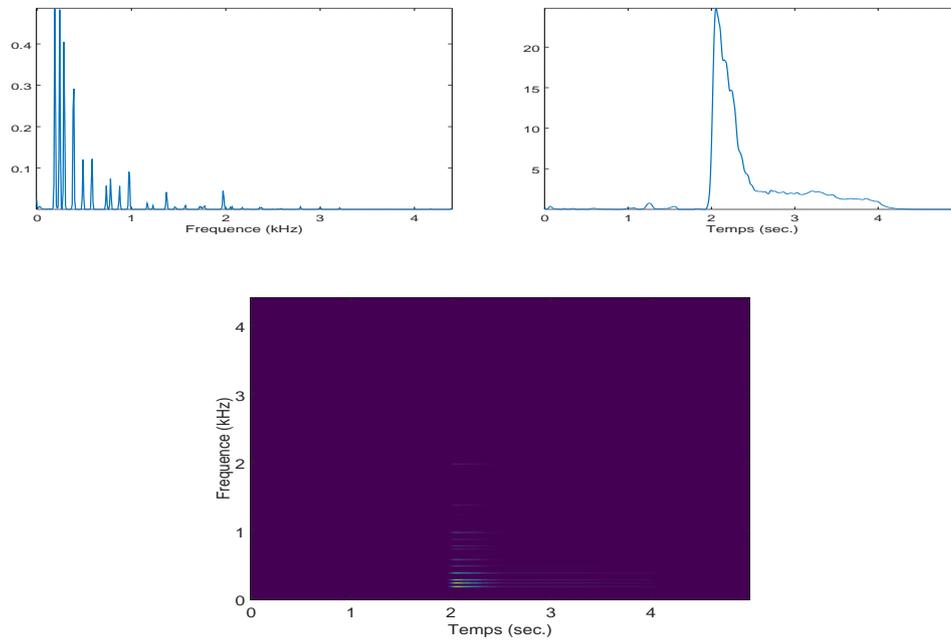


FIGURE 4.10 – Sixième composante trouvée par la NMF

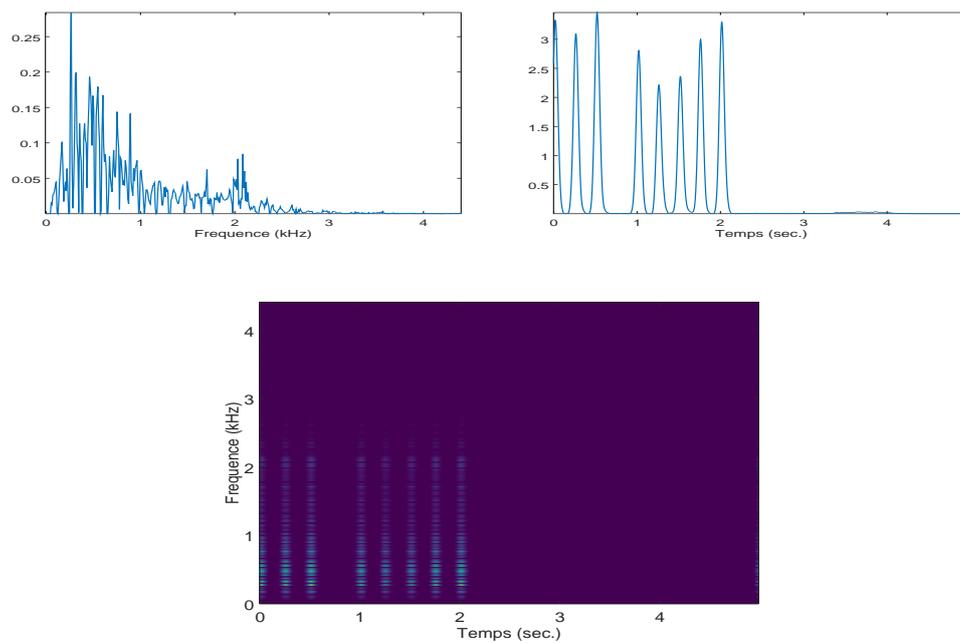
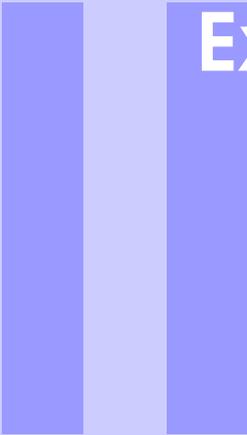


FIGURE 4.11 – Septième composante trouvée par la NMF



# Exercices, Travaux pratiques

<b>5</b>	<b>Travaux Pratiques : TP 1</b> .....	<b>65</b>
5.1	Filtrage numérique	
5.2	Stationnarité, spectre, estimation spectrale	
<b>6</b>	<b>Travaux Pratiques : TP 2</b> .....	<b>69</b>
6.1	Transformation de Gabor	
6.2	Filtrage d'un bruit blanc	
6.3	Filtrage de signaux musicaux	
6.4	Factorisation de spectrogramme	
<b>7</b>	<b>Correction des exercices</b> .....	<b>73</b>
7.1	Corrections des exercices du chapitre 2	
7.2	Corrections des exercices du chapitre 3	





## 5. Travaux Pratiques : TP 1

### 5.1 Filtrage numérique

On rappelle qu'un filtre numérique est un opérateur de convolution  $K_h : x \rightarrow h * x$ , où  $h$  est appelée réponse impulsionnelle.

$$(K_h x)[\ell] = \sum_n h[n]x[\ell - n] = \sum_m h[\ell - m]x[m].$$

Même si les calculs numériques se font avec des signaux de longueur finie, on raisonne souvent avec des modèles de signaux de longueur infinie, dans ce cas la somme est infinie. Le filtre est dit causal si  $h[n] = 0$  pour tout  $n < 0$ . Dans le cas où on raisonne directement sur des signaux de longueur finie, on doit compléter (par exemple par des zéros) pour que l'expression ci-dessus ait un sens. Une alternative, consiste à compléter par périodisation : si le signal est de longueur  $L$ , on le considère comme la restriction à  $\{0, \dots, L-1\}$  d'un signal de longueur infinie périodique de période  $L$ . Cette solution est implicite quand le filtrage est implémenté dans le domaine fréquentiel.

#### 5.1.1 Filtrage dans le domaine fréquentiel

On rappelle qu'un filtre de réponse impulsionnelle  $h$  s'écrit dans le domaine de la transformée en  $z$  sous la forme suivante : si  $y = K_h x$

$$Y(z) = H(z)X(z)$$

le domaine fréquentiel étant obtenu pour  $z = e^{2i\pi v}$ , sous la forme

$$\hat{y}(v) = \hat{h}(v)\hat{x}(v).$$

#### Exercice 5.1 — Filtre passe-bas idéal.

Implémenter une fonction effectuant un filtrage passe-bas idéal, sélectionnant une bande de fréquence  $[F_1, F_2]$  donnée, pour un signal échantillonné à une fréquence  $F_s$ . On prendra  $0 < F_1 < F_2$ , en s'assurant que le choix soit compatible avec les règles d'échantillonnage.

Attention : le signal d'entrée étant supposé réel (donc sa transformée de Fourier a la symétrie Hermittienne), le signal de sortie doit être réel aussi. ■

### 5.1.2 Filtrage dans le domaine temporel

La fonction de filtrage de référence sous Matlab/Octave est la fonction `filter`, qui peut être utilisée avec différentes syntaxes. On rappelle qu'un filtre est d'ordre fini s'il peut s'écrire sous la forme de la transformation  $x \rightarrow y$  où  $y$  est solution de l'équation

$$\sum_{k=0}^K a[k]y[n-k] = \sum_{\ell=0}^L b[\ell]x[n-\ell] \quad \forall n \in \mathbb{Z},$$

où les coefficients  $a[k]$  et  $b[\ell]$  sont réels, avec  $a[0] = 1$ .

Avec ces notations, la syntaxe de base de la fonction `filter` est alors

```
>> y = filter(b,a,x)
```

Il existe d'autres syntaxes pour la fonction `filter`, prenant comme variables d'entrées d'autres paramétrisations du filtre, voir la documentation. Signalons aussi la fonction `sosfilter`, qui utilise le filtre mis sous la forme *second order section* (c'est à dire factorisé en produit de filtres d'ordre 2).

#### Filtrage FIR

Dans le cas de filtres FIR,  $h$  est un vecteur de longueur finie (supposé causal), la syntaxe est directement

```
>> y = filter(h,1,x);
```

#### Exercice 5.2 — Moyenne mobile.

Un filtre de moyenne mobile est le plus simple des filtres passe-bas, et correspond à une réponse impulsionnelle  $h[\ell] = 1/L$  pour  $\ell = 0, \dots, L-1$  et  $h[\ell] = 0$  sinon. On va ici illustrer le problème de délai introduit par le filtrage.

1. On peut obtenir diverses représentations graphiques de ce filtre grâce à la fonction `fvtool`, la syntaxe est ici `fvtool(h)`.
2. Générer un signal test; ça peut être par exemple l'indicatrice d'un intervalle, à laquelle on ajoute un bruit blanc (fonction `randn`, se documenter). Effectuer un filtrage de moyenne mobile, et tracer sur une même figure le signal initial et le signal filtré, pour visualiser le délai.
3. Toutes les fréquences n'ont pas nécessairement le même délai. La fonction `grpdelay` calcule le retard de groupe, qui est une estimation du délai en fonction de la fréquence. La syntaxe est ici `grpdelay(h,N)` où  $N$  est le nombre de points sur lesquels est calculé le retard (on pourra étudier la syntaxe dans le cas général). Tracer le retard de groupe du filtre de moyenne mobile.
4. Corriger le retard sur le signal filtré. Le filtre corrigé est-il encore causal ?

#### Filtrage IIR

La toolbox *Signal Processing* implémente de nombreux outils de traitement du signal. Outre la fonction `filter`, on mentionnera

- de nombreuses fonctions générant des filtres d'une famille donnée, par exemple `butter` qui génère des filtres de Butterworth, ou `cheby1` et `cheby2` qui génèrent des filtres de Chebyshev d'ordre 1 et 2. Plusieurs syntaxes sont possibles, voir la documentation.
- différents outils de conversion entre formats de filtres (sos, zéros-pôles, coefficients,...), par exemple `tf2zp` et `zp2tf`, `sos2zp`,...
- différents outils de visualisation du filtre, à commencer par `fvtool`, ou la fonction `freqz` qui visualise les positions des pôles et des zéros dans le plan complexe,...
- les fonction de plus haut niveau comme `fdesign` et `filterBuilder`, ou l'application `filterDesigner`

#### Exercice 5.3 — Design d'un filtre passe-bande.

Charger un des sons tests proposés dans l'archive (par exemple un son de note tenue, et un son plus complexe), et sa fréquence d'échantillonnage  $F_s$ .

1. En utilisant l'un des outils mentionnés plus haut, construire quelques exemples de filtres passe-bande (FIR et IIR) sélectionnant une bande de fréquence (par exemple entre 1000 Hz et 4000 Hz, à voir en fonction du signal choisi), pour la fréquence d'échantillonnage obtenue plus haut.
3. Donner des représentations graphiques de ces filtres (réponse impulsionnelle, réponse inertielle,

fonction de transfert, diagramme pôle-zéro,...)

4. Calculer les retards de groupe de ces filtres. Qu'observe-t-on ? Comment corriger le délai introduit par ces filtres ?
5. Effectuer le filtrage du son choisi,... et écouter.

■

## 5.2 Stationnarité, spectre, estimation spectrale

### 5.2.1 Estimation spectrale

#### Exercice 5.4 — Périodogramme d'un bruit blanc et un bruit coloré.

En utilisant la fonction `randn` générer 4 secondes de bruit blanc gaussien échantillonné à 44100 Hz. Utiliser le périodogramme (fonction `periodogram`, voir la documentation) pour en estimer le spectre. Tracer ce spectre estimé. Calculer la valeur moyenne et l'écart type de ce spectre estimé.

2. Effectuer une estimation du spectre en utilisant le périodogramme moyenné (segmentation du signal en segments de même longueur, calcul du périodogramme sur chacun de ces segments et calcul du périodogramme moyen). Cette opération peut aussi être effectuée avec la fonction `periodogram` en modifiant la variable `nfft`, se documenter). Le représenter graphiquement, et évaluer sa moyenne et son écart-type. On pourra tester différentes longueurs de segments.
3. La fonction `pwelch` implémente le périodogramme de Welch. Mêmes questions.
4. On peut générer un bruit coloré par filtrage d'un bruit blanc. En utilisant les outils décrits dans les exercices précédents, générer un filtre passe-bande de votre choix, l'utiliser pour filtrer le bruit blanc, et effectuer les opérations précédentes.

■

#### Exercice 5.5 — Périodogramme d'un son.

Utiliser le périodogramme et le périodogramme de Welch pour effectuer une estimation spectrale sur des sons de notes isolées (violon, trompette,...).

2. Même question sur des sons plus longs (blues, rock, classique,...).
3. Dans ce dernier cas, effectuer des estimations spectrales sur des segments plus courts (de l'ordre de 10ms) ; qu'observe-t-on ? La notion de stationnarité vous paraît-elle toujours adaptée à ce cas ?

■

Dans le cas non-stationnaire, on est naturellement conduit à effectuer une estimation spectrale sur des segments de signal plus courts, à l'intérieur desquels on peut "raisonnablement" supposer la stationnarité. Une collection de spectres estimés sur une fenêtre que l'on fait glisser le long de l'axe temporel constitue ce que l'on appelle un spectrogramme, qui est un prototype de représentation temps-fréquence.

### 5.2.2 Estimation d'un modèle AR

On rappelle qu'un modèle autorégressif est un modèle paramétrique de signal stationnaire, où celui-ci est obtenu par un filtrage particulier (on parle de filtrage *all pole*) d'un bruit blanc :

$$\sum_{k=0}^K a[k]x[n-k] = \varepsilon[n] \quad \forall n$$

où les coefficients  $a[k]$  sont réels tels que  $a[0] = 1$ , et où  $\varepsilon$  est un bruit blanc centré (c'est à dire de moyenne nulle) de variance  $\sigma^2$ .

#### Exercice 5.6 — Un exemple simulé.

Générer un bruit blanc Gaussien, et le filtrer avec un filtre *all pole* défini par

1. `>> a = [1, -2.4453, 2.9380, -1.7541, 0.5426];`

On pourra calculer les racines et les pôles du filtre en utilisant la fonction `tf2zp`, et les visualiser avec `zplane`. Vérifier que le filtre est bien *all pole*.

2. En utilisant la fonction `aryule` (se documenter sur la syntaxe), qui estime les paramètres d'un modèle AR en résolvant l'équation de Yule-Walker, donner une estimation des coefficients  $a[k]$  à

partir du signal filtré, ainsi que de la variance du bruit blanc de départ. Les comparer aux coefficients de départ.

3. Effectuer cette expérience avec plusieurs (une cinquantaine par exemple) réalisations, pour obtenir des indications sur la distribution des coefficients (moyenne, écart-type,...) et de la variance estimée.

### Exercice 5.7 — Modèle AR sur un signal de parole.

On va s'intéresser ici à l'estimation d'un modèle AR sur un extrait de signal de parole. On utilisera le mot `matlab` prononcé par une locutrice, fourni dans la *signal processing toolbox*. L'instruction

```
>> load matlab
```

charge en mémoire un signal nommé `matlab` et la fréquence d'échantillonnage  $F_s$ .

L'objectif est d'identifier les formants, qui sont définis comme les maxima d'énergie du spectre sonore de ce son de parole. Il y a plusieurs définitions alternatives du mot formant (résonances du conduit vocal, pôles, etc.). On les identifie ici aux pôles d'un modèle AR.

1. Charger le son `matlab` en mémoire, et l'écouter. Effectuer l'analyse spectrale en utilisant `periodogram` ou `pwelch`. Voit-on quoi que ce soit d'interprétable ?
2. Calculer un spectrogramme en utilisant la fonction `spectrogram` de la *signal processing toolbox*, plus précisément

```
>> spectrogram(matlab, hamming(128), 120, 128, Fs, 'yaxis');
```

(ici on utilise une fenêtre de `hamming` de longueur 128 échantillons, on pourra la tracer pour se faire une idée, par exemple en utilisant `wvtool`; 120 est le nombre d'échantillons chevauchants, c'est à dire communs à deux fenêtres, et 128 est le nombre de fréquences calculées).

3. Extraire la partie du signal comprise entre 100 et 250 ms (on pourra utiliser une fenêtre plus régulière que la fenêtre rectangulaire); c'est sur cette partie que l'on va essayer d'estimer un modèle AR.
4. On cherche  $N_f = 3$  formants, une règle empirique fixe l'ordre du modèle à  $2N_f + 2$ , soit donc 8. En utilisant la fonction `aryule`, estimer les paramètres du filtre *all pole* correspondant. En déduire les zéros et les pôles (fonction `tf2zp`) et les représenter dans le plan complexe. On va maintenant s'intéresser aux valeurs des fréquences et aux largeurs de bande représentées par ces pôles.
5. Les fréquences correspondant aux arguments des pôles complexes. Les identifier et les convertir en Hz (à partir de valeurs données en radians par échantillon).
6. Les largeurs de bande correspondent aux distances entre les pôles et le cercle unité (normalisées), les calculer suivant la formule suivante : étant donné un pôle  $p \in \mathbb{C}$ ,

$$B = \frac{1}{2} \frac{\eta}{2\pi} \ln(|p|^{-1}),$$

$\eta$  étant la fréquence d'échantillonnage

7. On estime que les formants correspondent aux fréquences supérieures à 90 Hz et ont une largeur de bande inférieure à 400 Hz. Effectuer la sélection des formants sur cette base.



## 6. Travaux Pratiques : TP 2

On s'intéresse ici à la mise en œuvre de filtres dépendant du temps (contrairement aux filtres de convolution classiques), et plus précisément aux filtres réalisés comme multiplicateurs temps-fréquence (multiplicateurs de Gabor).

Partant de deux fenêtres d'analyse et de synthèse  $g, h \in \mathbb{C}^L$ , de pas d'échantillonnage  $a$  et  $b$  du plan temps-fréquence (tous deux diviseurs de  $L$ , i.e.  $L = Na = Mb$ ) on considère les atomes de Gabor correspondants  $g_{mn}$  et  $h_{mn}$ . Étant donné un **masque**  $\mu \in \mathbb{C}^{M \times N}$ , le multiplicateur de Gabor associé est l'opérateur linéaire  $\mathbb{M}$  qui associe à un signal d'entrée  $x \in \mathbb{C}^L$  un signal de sortie  $y = \mathbb{M}x = \mathcal{V}_g^* \mu \mathcal{V}_g x \in \mathbb{C}^L$  défini par

$$\mathbb{M}x = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mu[m, n] \mathcal{V}_g x[m, n] h_{mn},$$

où les nombres (complexes)  $\mathcal{V}_g x[m, n] = \langle x, g_{mn} \rangle$  constituent la transformée de Gabor de  $x$ . On va ici jouer avec ces filtres non-stationnaires pour synthétiser des sons à partir d'un bruit blanc.

On utilisera pour cela le package LTFAT disponible sous matlab et octave, et les fonctions de transformation de Gabor et Gabor inverse. Les fonctions de référence sont `dgt` et `idgt`, qu'on a déjà utilisées l'an passé. On utilisera plutôt ici les fonctions `dgtreal` et `idgtreal`, adaptée à des signaux à valeurs réelles, et qui exploitent la symétrie Hermitienne de la transformée de Gabor correspondante en n'utilisant que les fréquences positives ou nulles.

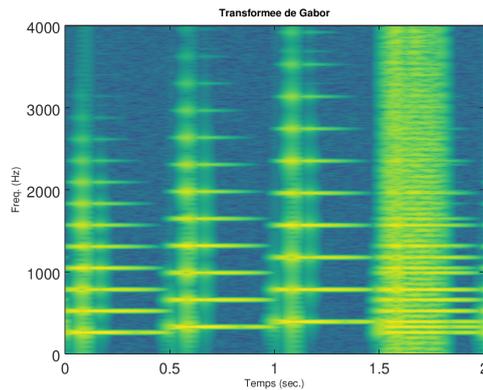
Sous octave, le *package* LTFAT se charge par l'instruction `pkg load ltfat`, de même le *package* *signal processing* se charge par l'instruction `pkg load signal`.

### 6.1 Transformation de Gabor

On commencera par se documenter sur les fonctions `dgtreal` et `idgtreal` et leur syntaxe, avant de vérifier les propriétés de reconstruction exacte. Plus précisément on effectuera les opérations suivantes :

- Générer une réalisation  $x$  de longueur  $L$  d'un bruit blanc gaussien (fonction `randn`).
- Calcul de sa transformée de Gabor  $\mathcal{V}_g x$  par `dgtreal`. Il faudra avoir choisi la fenêtre d'analyse  $g$  (par exemple une gaussienne) et les pas  $a$  et  $b$ .
- Calcul de la transformée inverse (fonction `idgtreal`), avec une fenêtre de synthèse  $h$ , choisie comme la fenêtre duale canonique de  $g$  (fonction `gabdua1`).

Si tout se passe bien, le signal reconstruit sera identique (à la précision numérique près) au signal d'entrée. On pourra aussi visualiser la transformée par une image (i.e. le spectrogramme, égal au module carré de la transformée).



Spectrogramme d'une séquence de 3 notes et un accord.

Une fois testées ces fonctions de base, on créera une fonction qui mettra en oeuvre un multiplicateur de Gabor. Réfléchir au préalable à la syntaxe de cette fonction (variables d'entrée et de sortie notamment). Dans ce qui suit on testera dans des situations simples.

## 6.2 Filtrage d'un bruit blanc

### 6.2.1 Filtrage "passe-bande"

On va ici générer une "bande de bruit" en appliquant un multiplicateur de Gabor à un bruit blanc. Pour cela, on commence par générer une réalisation d'un bruit blanc Gaussien à une fréquence d'échantillonnage donnée. Partant d'un masque (binaire, c'est à dire constitué de 0 et 1) sélectionnant une bande de fréquence bien choisie, on peut alors appliquer le multiplicateur correspondant au bruit, et tracer et écouter le signal de sortie.

Il pourra aussi être intéressant de représenter le spectrogramme du signal de sortie, et le comparer au produit  $\mu \mathcal{V}_g x$ .

### 6.2.2 Filtrage non-stationnaire

- L'idée est ici de générer (toujours à partir d'un bruit blanc) des sons non-stationnaires. Par exemple
- une bande de fréquences variant au cours du temps
  - un évènement localisé en temps et en fréquence (une "note" aléatoire)
  - il sera intéressant de représenter aussi le chronogramme (représentation temporelle) et le spectre du signal généré.

On pourra écrire une fonction générant un masque à support rectangulaire dans le plan temps-fréquence (entre deux instants donnés et deux fréquences donnés) et valant 1 à l'intérieur de ce support et zéro ailleurs, puis utiliser cette fonction pour générer une séquence de notes à partir d'un bruit blanc. Un exemple se trouve en Figure 6.1.

On pourra aussi modifier le masque en lui donnant une décroissance exponentielle en fonction du temps (et ainsi renforcer les attaques), ou en générant des harmoniques. On peut aussi s'amuser à lui donner d'autres formes que la forme rectangulaire.

## 6.3 Filtrage de signaux musicaux

Sur le site du cours il y a un certain nombre d'exemples de signaux musicaux. On peut s'intéresser à l'utilisation des multiplicateurs pour différentes tâches, comme par exemple

- Modification du timbre pour une note isolée d'instrument
- Suppression d'une note dans une séquence
- ...

## 6.4 Factorisation de spectrogramme

La factorisation de spectrogramme peut être utilisée pour des tâches telles que la transcription automatique ou des tâches de détection de notes. Le principe est le suivant. Étant donnée la matrice  $S = |\mathcal{V}_g x|^2 \in \mathbb{R}_+^{M \times N}$

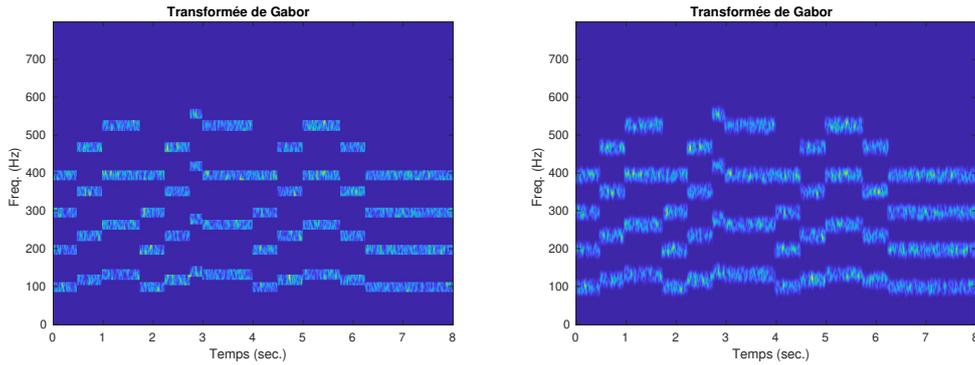


FIGURE 6.1 – Spectrogramme d’une version aléatoire de *Smoke on the water*. Gauche ; spectrogramme  $\mu_{\mathcal{G}}x$  du bruit blanc multiplié par un masque bien choisi ; droite, spectrogramme du signal reconstruit  $\mathbb{M}x = \mathcal{V}_{\mathcal{G}}^* \mu_{\mathcal{G}}x$ .

du spectrogramme d’un signal  $x$ , et une valeur fixée  $K \in \mathbb{N}$  (appelée *rang de la factorisation*), il s’agit de trouver deux matrices non-négatives  $W \in \mathbb{R}_+^{M \times K}$  et  $H \in \mathbb{R}_+^{K \times N}$ , telles que

$$S \approx WH = \sum_{k=1}^K W^{(k)} H^{(k)} .$$

Ici chaque  $W^{(k)}$  est un vecteur "colonne", dont les indices sont les fréquences du spectrogramme :  $W^{(k)}$  représente donc le profil fréquentiel de la  $k$ -ième composante trouvée. Chaque  $H^{(k)}$  est un vecteur "ligne", dont les indices sont les instants du spectrogramme :  $H^{(k)}$  représente les activations temporelles de la  $k$ -ième composante.

Pour obtenir une telle factorisation, on procède comme suit : on se donne une fonction mesurant une dissimilarité entre  $S$  et le produit  $WH$ , et on minimise celle-ci par rapport à  $W$  et  $H$ . Les algorithmes les plus classiques sont des algorithmes alternés, qui alternent

- mise à jour de  $W$  à  $H$  fixé
- mise à jour de  $H$  à  $W$  fixé

jusqu’à convergence (éventuelle), avec éventuellement d’autres opérations (normalisation, projection,...).

Sur le site AMETICE du cours se trouve une archive contenant trois fonctions effectuant une telle factorisation, basées sur minimisation la divergence d’Itakura-Saito, qui semble bien adaptée aux spectrogrammes :

$$D_0(S|W,H) = \sum_{m=1}^M \sum_{n=1}^N \left[ \frac{S_{mn}}{(WH)_{mn}} - \log \left( \frac{S_{mn}}{(WH)_{mn}} \right) - 1 \right]$$

En utilisant les fonctions de factorisation, notamment `is_nmf_mu`, effectuer des factorisations de spectrogrammes, par exemple sur le son de cadence.



## 7. Correction des exercices

### 7.1 Corrections des exercices du chapitre 2

#### Correction de l'exercice 2.1 (Propriétés de base de la TFD)

1. Soit  $x \in \ell^1(\mathbb{Z})$  un signal numérique, soit  $\hat{x}$  sa TFD. Soit  $y$  une copie translatée de  $x$ , de la forme  $y[\ell] = x[\ell - \tau]$ , avec  $\tau \in \mathbb{Z}$ . Un calcul simple donne

$$\hat{y}(v) = \sum_{\ell=-\infty}^{\infty} x[\ell - \tau] e^{-2i\pi v \ell} = \sum_{k=-\infty}^{\infty} x[k] e^{-2i\pi v [k + \tau]} = e^{-2i\pi v \tau} \hat{x}(v).$$

La TFD de  $y$  s'obtient simplement par produit de la TFD de  $x$  par la fonction  $v \rightarrow e^{-2i\pi v \tau}$ .

2. Pour démontrer la formule de Parseval, calculons

$$\begin{aligned} \|\hat{x}\|^2 &= \int_{-1/2}^{1/2} |\hat{x}(v)|^2 dv = \int_{-1/2}^{1/2} \left[ \sum_{\ell=-\infty}^{\infty} x[\ell] e^{-2i\pi v \ell} \right] \overline{\left[ \sum_{k=-\infty}^{\infty} x[k] e^{-2i\pi v k} \right]} dv \\ &= \sum_{\ell, k=-\infty}^{\infty} x[\ell] \bar{x}[k] \int_{-1/2}^{1/2} e^{-2i\pi v (\ell - k)} dv \\ &= \sum_{\ell, k=-\infty}^{\infty} x[\ell] \bar{x}[k] \delta_{k, \ell} \\ &= \sum_{\ell=-\infty}^{\infty} |x[\ell]|^2 = \|x\|^2. \end{aligned}$$

#### Correction de l'exercice 2.2 (Sous-échantillonnage)

1. Étant donné un signal numérique  $x$ , on note  $y$  le signal obtenu par sous-échantillonnage d'un facteur 2 :  $y[n] = x[2n]$ , pour  $n \in \mathbb{Z}$ . Démontrer que

$$\hat{y}(v) = \frac{1}{2} [\hat{x}(v) + \hat{x}(v + 1/2)].$$

En notant maintenant  $z$  le signal obtenu par l'autre sous-échantillonnage de facteur 1/2, cette fois de la forme  $z[n] = x[2n + 1]$ , exprimer  $\hat{z}$  en fonction de  $\hat{x}$ .

2. Soit  $z$  obtenu par sous-échantillonnage d'un facteur 3 :  $z[n] = x[3n]$ , exprimer  $\hat{z}$  en fonction de  $\hat{x}$ .

**Correction de l'exercice 2.3 (Sur-échantillonnage)**

On s'intéresse maintenant au sur-échantillonnage d'un facteur 2 : étant donné un signal numérique  $x$ , soit  $y$  défini par

$$y[n] = \begin{cases} x[n/2] & \text{si } n \text{ est pair} \\ 0 & \text{sinon} \end{cases}$$

Exprimer  $\hat{y}$  en fonction de  $\hat{x}$ .

**Correction de l'exercice 2.4 (Fenêtrage)**

On travaille en pratique sur des signaux de longueur finie. On peut voir un signal de longueur finie  $y \in \mathbb{C}^L$  comme le produit d'un signal de longueur infinie  $x$  par une "fenêtre", par exemple l'indicatrice  $I_L$  de l'intervalle entier  $\{0, \dots, L-1\}$ , qui vaut 1 sur cet intervalle et 0 en dehors.

1. Calculons la TFD de ce produit  $y = x \cdot I_L$ .

$$\begin{aligned} \hat{y}[v] &= \sum_{\ell=-\infty}^{\infty} x[\ell] I_L[\ell] e^{-2i\pi v \ell} \\ &= \sum_{\ell=0}^{L-1} \left[ \int_{-1/2}^{1/2} \hat{x}(\xi) e^{2i\pi \xi \ell} d\xi \right] e^{-2i\pi v \ell} \\ &= \int_{-1/2}^{1/2} \hat{x}(\xi) \left[ \sum_{\ell=0}^{L-1} e^{2i\pi(\xi-v)\ell} \right] d\xi \\ &= \int_{-1/2}^{1/2} \hat{x}(\xi) \frac{1 - e^{-2i\pi(v-\xi)L}}{1 - e^{-2i\pi(v-\xi)}} d\xi. \end{aligned}$$

Ici on a utilisé la formule donnant la somme d'une série géométrique finie. On obtient bien la forme annoncée :  $\hat{y}(v) = \int_{-1/2}^{1/2} \hat{x}(\xi) \gamma(v-\xi) d\xi$ , la fonction  $\gamma$  étant donnée par

$$\gamma(v) = \frac{1 - e^{-2i\pi v L}}{1 - e^{-2i\pi v}}.$$

2. Plus généralement, si on remplace cette indicatrice par une fenêtre quelconque  $h$ , le même calcul donne

$$\begin{aligned} \hat{y}[v] &= \sum_{\ell=-\infty}^{\infty} x[\ell] h[\ell] e^{-2i\pi v \ell} \\ &= \sum_{\ell=-\infty}^{\infty} \left[ \int_{-1/2}^{1/2} \hat{x}(\xi) e^{2i\pi \xi \ell} d\xi \right] e^{-2i\pi v \ell} h[\ell] \\ &= \int_{-1/2}^{1/2} \hat{x}(\xi) \left[ \sum_{\ell=0}^{L-1} e^{2i\pi(\xi-v)\ell} h[\ell] \right] d\xi \\ &= \int_{-1/2}^{1/2} \hat{x}(\xi) \hat{h}(v-\xi) d\xi. \end{aligned}$$

Il s'agit de la même expression que plus haut, mais la fonction  $\gamma$  est cette fois la transformée de Fourier de la fenêtre :  $\gamma = \hat{h}$ .

**Correction de l'exercice 2.5 (Propriétés de la TFF)**

On considère donc la TFF en dimension  $L$ . Calculons

$$\sum_{k=0}^{L-1} |\hat{x}[k]|^2 = \frac{1}{L} \sum_{k=0}^{L-1} \left[ \sum_{\ell=0}^{L-1} x[\ell] e^{-2i\pi k \ell / L} \right] \overline{\left[ \sum_{\ell'=0}^{L-1} x[\ell'] e^{-2i\pi k \ell' / L} \right]} = \frac{1}{L} \sum_{\ell, \ell'=0}^{L-1} x[\ell] \bar{x}[\ell'] \sum_{k=0}^{L-1} e^{2i\pi k(\ell'-\ell)/L}.$$

Dans cette expression, la somme sur  $k$  est la somme partielle d'une série géométrique de raison  $e^{2i\pi(\ell'-\ell)/L}$ , et vaut donc

$$\sum_{k=0}^{L-1} \left[ e^{2i\pi(\ell'-\ell)/L} \right]^k = \begin{cases} \frac{1 - e^{2i\pi(\ell'-\ell)}}{1 - e^{2i\pi(\ell'-\ell)/L}} = 0 & \text{si } \ell' \neq \ell \\ L & \text{si } \ell' = \ell \end{cases}.$$

Ceci donne directement la formule de Parseval (2.10). Pour ce qui concerne la formule de transformée inverse (2.9), le calcul est similaire. Calculons, pour tout  $\ell = 0, \dots, L-1$

$$\frac{1}{\sqrt{L}} \sum_{k=0}^{L-1} \hat{x}[k] e^{2i\pi k\ell/L} = \frac{1}{L} \sum_{k,\ell'=0}^{L-1} x[\ell'] e^{2i\pi k(\ell-\ell')/L} = \frac{1}{L} \sum_{\ell'=0}^{L-1} x[\ell'] \sum_{k=0}^{L-1} e^{2i\pi k(\ell-\ell')/L} = \frac{1}{L} \sum_{\ell'=0}^{L-1} x[\ell'] L \delta_{\ell,\ell'} = x[\ell].$$

### Correction de l'exercice 2.6 (Sous-échantillonnage et sur-échantillonnage sous Matlab)

En prenant un signal simple (sinusoïde par exemple), illustrer les propriétés de la TFD liées au sur-échantillonnage et au sous-échantillonnage (exercices 2.2 et 2.3).

### Correction de l'exercice 2.7 (Propriétés simples de la transformation en $z$ )

1. Soit  $x$  un signal numérique, soit  $X$  sa transformée en  $z$ . Soit  $y$  une copie translatée de  $x$  de la forme  $y[\ell] = x[\ell - \tau]$ . En utilisant le changement de variable  $k = \ell - \tau$ , le calcul donne

$$Y(z) = \sum_{\ell=-\infty}^{\infty} x[\ell - \tau] z^{-\ell} = \sum_{k=-\infty}^{\infty} x[k] z^{-(k+\tau)} = z^{-\tau} X(z).$$

Le domaine de convergence de  $Y$  est identique au domaine de convergence de  $X$ .

2. Étant donnés deux signaux numériques  $x$  et  $h$ , soit  $y = h * x$  leur produit de convolution défini par

$$y[\ell] = (h * x)[\ell] = \sum_{k=-\infty}^{\infty} h[k] x[\ell - k].$$

Calculons sa transformée en  $z$  : en utilisant le changement de variable  $n = \ell - k$

$$Y(z) = \sum_{\ell=-\infty}^{\infty} \left[ \sum_{k=-\infty}^{\infty} h[k] x[\ell - k] \right] z^{-\ell} = \sum_{n,k=-\infty}^{\infty} h[k] x[n] z^{-k} z^{-n} = H(z) X(z).$$

Le domaine de convergence de  $Y$  est l'intersection du domaine de convergence de  $X$  et de  $H$ .

### Correction de l'exercice 2.8 (Transformation en $z$ et sur/sous-échantillonnage)

Soit  $x$  un signal numérique, soit  $X$  sa transformée en  $z$ .

1. On note  $y$  le signal obtenu par sous-échantillonnage d'un facteur 2 :  $y[n] = x[2n]$ , pour  $n \in \mathbb{Z}$ . Calculons

$$\begin{aligned} \frac{1}{2} [X(z^{1/2}) + X(-z^{1/2})] &= \frac{1}{2} \sum_{n=-\infty}^{\infty} [x[n] z^{-n/2} + (-1)^n x[n] z^{-n/2}] \\ &= \frac{1}{2} \sum_{n=-\infty}^{\infty} [1 + (-1)^n] x[n] z^{-n/2} \\ &= \sum_{p=-\infty}^{\infty} x[2p] z^{-p}, \end{aligned}$$

où dans la dernière expression seuls les indices  $n$  pairs ont été conservés, les autres ayant disparu car  $1 + (-1)^n = 0$  quand  $n$  est impair. L'expression obtenue n'est autre que la transformée en  $z$  du signal sous-échantillonné.

En notant maintenant  $y$  le signal obtenu par l'autre sous-échantillonnage de facteur 1/2, cette fois de la forme  $y[n] = x[2n + 1]$ , on veut exprimer  $Y$  en fonction de  $X$ . Le calcul sera essentiellement le même, mais il faudra faire disparaître les  $n$  pairs, ce qui suggère une soustraction plutôt qu'une addition. Calculons donc

$$\begin{aligned} \frac{1}{2} [X(z^{1/2}) - X(-z^{1/2})] &= \frac{1}{2} \sum_{n=-\infty}^{\infty} [x[n] z^{-n/2} - (-1)^n x[n] z^{-n/2}] \\ &= \frac{1}{2} \sum_{n=-\infty}^{\infty} [1 - (-1)^n] x[n] z^{-n/2} \\ &= \sum_{p=-\infty}^{\infty} x[2p + 1] z^{-p}, \end{aligned}$$

on a donc maintenant

$$Y(z) = \frac{1}{2} [X(z^{1/2}) - X(-z^{1/2})]$$

2. Soit  $z$  obtenu par sous-échantillonnage d'un facteur 3 :  $z[n] = x[3n]$ . Pour exprimer  $Z$  en fonction de  $X$ , on va effectuer un calcul similaire. Calculons

$$\begin{aligned} \frac{1}{3} \left[ X(z^{1/3}) + X(e^{2i\pi/3}z^{1/3}) + X(e^{4i\pi/3}z^{1/3}) \right] &= \frac{1}{3} \sum_{n=-\infty}^{\infty} \left[ x[n]z^{-n/3} + x[n]e^{2i\pi/3}z^{-n/3} + x[n]e^{4i\pi/3}z^{-n/3} \right] \\ &= \frac{1}{3} \sum_{n=-\infty}^{\infty} [1 + e^{2i\pi/3} + e^{4i\pi/3}]x[n]z^{-n/3} . \end{aligned}$$

On voit que

$$1 + e^{2i\pi/3} + e^{4i\pi/3} = \begin{cases} 3 & \text{si } n \text{ est multiple de } 3 \\ 0 & \text{sinon} \end{cases}$$

du coup

$$\frac{1}{3} \left[ X(z^{1/3}) + X(e^{2i\pi/3}z^{1/3}) + X(e^{4i\pi/3}z^{1/3}) \right] = \sum_{p=-\infty}^{\infty} x[3p]z^{-p}$$

le terme de gauche est donc la transformée en  $z$  de  $x$  sous échantillonné d'un facteur 3.

3. On s'intéresse maintenant au sur-échantillonnage d'un facteur  $K$  : étant donné un signal numérique  $x$ , soit  $y$  défini par

$$y[n] = \begin{cases} x[n/K] & \text{si } n \text{ est multiple de } K \\ 0 & \text{sinon} \end{cases}$$

Calculons

$$Y(z) = \sum_{n=Kp, p \in \mathbb{Z}} x[n/K]z^{-n} = \sum_{p=-\infty}^{\infty} x[p]z^{-Kp} = X(z^K),$$

on a donc  $Y(z) = X(z^K)$ .

### Correction de l'exercice 2.9 (Une transformée en $z$ simple)

Soit  $X$  la fonction d'une variable complexe définie par

$$X(z) = \frac{1}{z - z_0}, \quad |z| \leq |z_0|,$$

où  $z_0 = \rho e^{2i\pi v_0} \in \mathbb{C}$  avec  $\rho < 1$  et  $v_0 \in [-1/2, 1/2]$ .

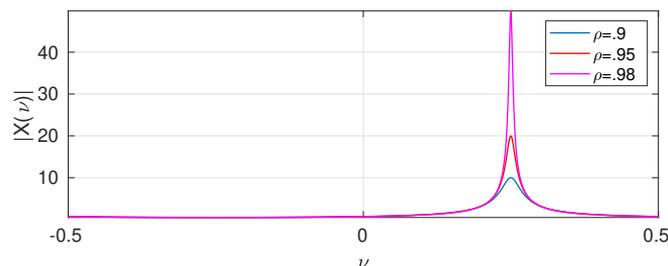
1. Pour calculer  $|X(z)|$ , posons  $z = re^{2i\pi v}$  et calculons

$$\begin{aligned} |z - z_0|^2 &= (re^{2i\pi v} - \rho e^{2i\pi v_0})(re^{-2i\pi v} - \rho e^{-2i\pi v_0}) \\ &= r^2 - r\rho (e^{2i\pi(v-v_0)} + e^{-2i\pi(v-v_0)}) + \rho^2 \\ &= r^2 + \rho^2 + 2r\rho \cos(2\pi(v - v_0)). \end{aligned}$$

On a donc

$$|X(z)|^2 = \frac{1}{r^2 + \rho^2 + 2r\rho \cos(2\pi(v - v_0))}.$$

On trace ci-dessous  $|X(e^{2i\pi v})|$  pour  $v \in [-1/2, 1/2]$ , pour différentes valeurs de  $\rho$  et  $v_0 = 1/4$ . On peut voir la présence d'un pic aux alentours de  $v = v_0$ , d'autant plus marqué que  $\rho$  est proche de 1 (c'est à dire que le pôle  $z_0$  est proche du cercle unité).



2. On peut écrire

$$\frac{1}{z - z_0} = -\frac{1}{z_0} \frac{1}{1 - z/z_0} = -\frac{1}{z_0} \sum_{\ell=0}^{\infty} \left(\frac{z}{z_0}\right)^{\ell} = -\frac{1}{z_0} \sum_{\ell=0}^{\infty} z_0^{-\ell} z^{\ell} = -\frac{1}{z_0} \sum_{\ell=-\infty}^0 z_0^{\ell} z^{-\ell}$$

Déterminer le signal numérique  $x$  dont  $X$  est la transformée en  $z$ . Par identification,  $x$  est le signal anticausal donné par  $x[\ell] = -z_0^{\ell-1}$  pour  $\ell \leq 0$ , et  $x[\ell] = 0$  pour  $\ell > 0$ .

### Correction de l'exercice 2.10 (Inversion de la transformation en $z$ )

Soit  $X$  la fonction d'une variable complexe définie par

$$X(z) = \frac{1}{(z-1)(z-2i)}, \quad 1 < |z| < 2.$$

Pour décomposer  $X(z)$  en éléments simples, on cherche une expression de la forme

$$X(z) = \frac{a}{z-1} + \frac{b}{z-2i},$$

et les coefficients  $a$  et  $b$  sont déterminés par

$$a = \lim_{z \rightarrow 1} (z-1)X(z) = \frac{1}{1-2i}, \quad b = \lim_{z \rightarrow 2i} (z-2i)X(z) = \frac{1}{2i-1}.$$

Comme  $|z| > 1$  on écrit

$$\frac{1}{z-1} = -\frac{1}{z} \frac{1}{1-1/z} = -\frac{1}{z} \sum_{\ell=0}^{\infty} z^{-\ell} = -\sum_{k=1}^{\infty} z^{-k}.$$

De même, puisque  $|z| < 2$ , on écrit cette fois

$$\frac{1}{z-2i} = -\frac{1}{2i} \frac{1}{1-z/2i} = -\frac{1}{2i} \sum_{\ell=0}^{\infty} \left(\frac{z}{2i}\right)^{\ell} = -\frac{1}{2i} \sum_{k=-\infty}^0 (2i)^k z^{-k} = -\sum_{k=-\infty}^0 (2i)^{k-1} z^{-k}.$$

Le signal numérique  $x$  dont  $X$  est la transformée en  $z$  est donc le signal

$$x[\ell] = \begin{cases} -\frac{(2i)^{\ell-1}}{2i-1} & \text{pour } \ell \leq 0 \\ -\frac{1}{1-2i} & \text{pour } \ell > 1 \end{cases}$$

### Correction de l'exercice 2.11 (Covariance des filtres numériques)

Pour démontrer la propriété de covariance des filtres numériques énoncée dans la Remarque 2.11, il suffit d'écrire

$$(K_h y)[\ell] = \sum_k h_k y[\ell-k] = \sum_k h_k x[\ell-k-n_0] = \sum_k h_k x[(\ell-n_0)-k] = (K_h x)(\ell-n_0).$$

### Correction de l'exercice 2.12 (Calcul d'une fonction de transfert)

Dans l'exemple 2.5, la réponse impulsionnelle du filtre est donnée par

$$h[n] = \frac{1}{L} \quad \text{pour } n = 0, \dots, L-1 \quad \text{et} \quad h[n] = 0 \quad \text{sinon}.$$

La fonction de transfert de ce filtre est simplement la TFD de  $h$ , soit

$$\hat{h}(v) = \frac{1}{L} \sum_{\ell=0}^{L-1} e^{-2i\pi v \ell} = \frac{1}{L} \frac{1 - e^{-2i\pi v L}}{1 - e^{-2i\pi v}}$$

, ce qui est une expression complexe difficile à expliciter. Le module carré vaut

$$|\hat{h}(v)|^2 = \frac{1}{L^2} \frac{2 - 2\cos(2\pi v L)}{2 - 2\cos(2\pi v)} = \frac{\sin^2(\pi v L)}{L \sin^2(\pi v)}.$$

On peut voir que  $\lim_{v \rightarrow 0} |\hat{h}(v)|^2 = 1$ , donc la fréquence  $v = 0$  et les fréquences proches sont conservées. Par contre, cette fonction s'annule pour toutes les valeurs de  $v$  multiples de  $1/L$ , ce qui occasionne des "rebonds" souvent jugés indésirables, comme on peut les voir sur la figure 2.4.

**Correction de l'exercice 2.13 (Calcul d'une réponse impulsionnelle)**

On considère le filtre passe-bas "idéal", de fonction de transfert

$$m(\nu) = \begin{cases} 1 & \text{si } |\nu| \leq \nu_0 \\ 0 & \text{sinon} \end{cases}$$

où  $\nu_0 \in ]0, 1/2[$  est la fréquence de coupure. La réponse impulsionnelle de ce filtre est donnée par la TFD inverse

$$h_n = \int_{-1/2}^{1/2} m(\nu) e^{2i\pi\nu n} d\nu = \int_{-\nu_0}^{\nu_0} e^{2i\pi\nu n} d\nu = \left[ \frac{e^{2i\pi\nu n}}{2i\pi n} \right]_{-\nu_0}^{\nu_0} = \frac{e^{2i\pi\nu_0 n} - e^{-2i\pi\nu_0 n}}{2i\pi n} = \frac{\sin(2\pi\nu_0 n)}{\pi n}.$$

On voit que cette réponse impulsionnelle est infinie (elle ne s'annule pas à partir d'un certain  $n$  assez grand), et n'est pas causale (elle ne s'annule pas pour  $n < 0$ ). De plus, on voit qu'elle décroît comme  $O(1/n)$ , c'est à dire très lentement (il faut environ un million de termes pour obtenir une précision de  $10^{-6}$ ).

**Correction de l'exercice 2.14 (Filtre passe-bas de Butterworth)**

On considère le Filtre passe-bas de Butterworth d'ordre 1, dont le module carré de la fonction de transfert vaut

$$|m(\nu)|^2 = \frac{1 + \cos(2\pi\nu)}{(1 + \cos(2\pi\nu)) + k(1 - \cos(2\pi\nu))},$$

où  $k$  est une constante. Soit  $0 < \nu_0 < 1/2$  une fréquence de référence, soit  $0 < \lambda < 1$ . La valeur de  $k$  telle que  $|m(\nu_0)|^2 = \lambda$  est telle que

$$1 + \cos(2\pi\nu_0) = \lambda(1 + \cos(2\pi\nu_0)) + k(1 - \cos(2\pi\nu_0)),$$

ou encore en posant  $w_0 = \cos(2\pi\nu_0)$

$$1 + w_0 = \lambda[(1 + w_0) + k(1 - w_0)],$$

Il suffit de résoudre cette équation linéaire en  $k$ , qui donne

$$k = (1 - \lambda) \frac{1 - w_0}{1 + w_0}$$

**Correction de l'exercice 2.15 (Filtres de Butterworth)**

1. On considère un filtre passe-bas de Butterworth dont le module carré de la fonction de transfert est donné par

$$|\hat{h}(\nu)|^2 = W(w) = \frac{(w+1)^L}{(w+1)^L + k(1-w)^L},$$

avec  $L = 1$ . Il s'agit d'en extraire une racine "module carré", en factorisant numérateur et dénominateur. On pose  $w = \frac{1}{2}(z + z^{-1})$ . Factorisons tout d'abord le numérateur, on cherche ses racines, c'est à dire les valeurs de  $z \in \mathbb{C}$  telles que  $\frac{1}{2}(z + z^{-1}) + 1 = 0$ . Ceci équivaut à l'équation  $z^2 + 2z + 1 = 0$ , soit  $z = -1 = e^{\pm i\pi}$ . On a une racines double, qui est donc sur le cercle unité et correspond à  $\nu_{\pm} = \pm 1/2$ , on écrit

$$w + 1 = \frac{z^{-1}}{2} (z + 1)^2.$$

Passons au dénominateur... à suivre

2. Même question pour  $L = 2$ .

**Correction de l'exercice 2.16 (Spectre du bruit blanc et d'un bruit coloré)**

1. Dans le cas d'un bruit blanc, la fonction d'autocovariance est donnée par  $\mathcal{C}[\ell] = \sigma^2 \delta_{\ell,0}$ . De là, on peut écrire, par transformation de Fourier discrète

$$\mathcal{S}(\nu) = \sum_{\ell=-\infty}^{\infty} \sigma^2 \delta_{\ell,0} e^{-2i\pi\nu\ell} = \sigma^2.$$

Par unicité du spectre, on a donc

$$\mathcal{S}(v) = \sigma^2, \quad \forall v \in [-1/2, 1/2],$$

ce qui est le spectre du bruit blanc donné dans l'exemple 2.7.

2. Dans le cas d'un bruit coloré  $y = K_h x$ , le calcul donne

$$\mathbb{E}\{y[\ell]\bar{y}[\ell']\} = \sum_{k,k'} h[k]\bar{h}[k'] \mathbb{E}\{x[\ell-k]\bar{x}[\ell'-k']\} = \sigma^2 \sum_{k,k'} h[k]\bar{h}[k'] \delta_{\ell-k, \ell'-k'} = \sigma^2 \sum_k h[k]\bar{h}[k+(\ell'-\ell)]$$

Cette quantité ne dépend de  $\ell$  et  $\ell'$  qu'à travers la différence  $\ell' - \ell$ , ce qui montre que  $y$  est bien faiblement stationnaire. Sa fonction d'autovariance est donc donnée par

$$\mathcal{C}[\ell] = \mathbb{E}\{y[\ell]\bar{y}[0]\} = \sigma^2 \sum_k h[k]\bar{h}[k-\ell].$$

En exprimant  $h$  à partir de sa TFD  $\hat{h}$ , on a ainsi

$$\mathcal{C}[\ell] = \sigma^2 \sum_k h[k] \overline{\int_{-1/2}^{1/2} \hat{h}(v) e^{2i\pi v(k-\ell)} dv} = \sigma^2 \int_{-1/2}^{1/2} \bar{\hat{h}}(v) \left[ \sum_k h[k] e^{-2i\pi v k} \right] e^{2i\pi v \ell} dv = \sigma^2 \int_{-1/2}^{1/2} |\hat{h}(v)|^2 e^{2i\pi v \ell} dv,$$

d'où on peut déduire (par unicité du spectre) le spectre du bruit coloré donné dans l'exemple 2.8 :

$$\mathcal{S}(v) = \sigma^2 |\hat{h}(v)|^2.$$

### Correction de l'exercice 2.17 (Equations de Yule-Walker)

Démontrer les équations de Yule-Walker (2.29) puis (2.31).

## 7.2 Corrections des exercices du chapitre 3

### Correction de l'exercice 3.1 (Définitions équivalentes de la TFCT)

L'expression (3.2) résulte directement de la définition des atomes de temps-fréquence, en calculant

$$\langle x, g_{(v,n)} \rangle = \sum_{\ell=-\infty}^{\infty} x[\ell] \overline{g_{(v,n)}[\ell]} = \sum_{\ell=-\infty}^{\infty} x[\ell] \bar{g}[\ell-n] e^{-2i\pi v(\ell-n)},$$

ce qui est l'expression recherchée. Pour vérifier (3.3), calculons tout d'abord

$$\begin{aligned} \widehat{g_{(v,n)}}(\xi) &= \sum_{\ell=-\infty}^{\infty} g[\ell-n] e^{2i\pi v(\ell-n)} e^{-2i\pi \xi \ell} \\ &= e^{-2i\pi \xi n} \sum_{k=-\infty}^{\infty} g[k] e^{-2i\pi(\xi-v)k} \\ &= e^{-2i\pi \xi n} \hat{g}(\xi-v). \end{aligned}$$

On en déduit alors, en utilisant la formule de Parseval

$$\mathcal{V}_g x[v, n] = \langle \hat{x}, \widehat{g_{(v,n)}} \rangle = \int_{-1/2}^{1/2} \hat{x}(\xi) \bar{\hat{g}}(\xi-v) e^{-2i\pi \xi n} dv.$$

Concernant (3.5), explicitons simplement le membre de droite de l'équation

$$\sum_{\ell=-\infty}^{\infty} x[\ell] \bar{g}_v[n-\ell] = \sum_{\ell=-\infty}^{\infty} x[\ell] e^{-2i\pi v(\ell-n)} \bar{g}[\ell-n] = \mathcal{V}_g x[v, n]$$

**Correction de l'exercice 3.2 (Inversion de la TFCT)**

Calculons tout d'abord

$$\begin{aligned} \int_{-1/2}^{1/2} \mathcal{V}_g x[v, n] g_{v, n}[\ell] dv &= \int_{-1/2}^{1/2} \sum_{k=-\infty}^{\infty} x[k] \bar{g}[k-n] e^{-2i\pi v(k-n)} g[\ell-n] e^{2i\pi v(\ell-n)} \\ &= \sum_{k=-\infty}^{\infty} x[k] \bar{g}[k-n] g[\ell-n] \int_{-1/2}^{1/2} e^{2i\pi v(\ell-k)} dv \\ &= x[\ell] |g[\ell-n]|^2 \end{aligned}$$

car  $\int_{-1/2}^{1/2} e^{2i\pi v(\ell-k)} dv = \delta_{k, \ell}$ . De là on déduit

$$\sum_{n=-\infty}^{\infty} \int_{-1/2}^{1/2} \mathcal{V}_g x[v, n] g_{v, n}[\ell] dv = \sum_{n=-\infty}^{\infty} x[\ell] |g[\ell-n]|^2 = x[\ell] \sum_{n'=-\infty}^{\infty} |g[n']|^2 = \|g\|^2 x[\ell]$$

où on a fait le changement d'indice de notation  $n' = \ell - n$ . Ceci prouve la première partie du résultat.

Concernant la seconde partie, le calcul est très similaire. Calculons

$$\begin{aligned} \int_{-1/2}^{1/2} |\mathcal{V}_g x[v, n]|^2 dv &= \int_{-1/2}^{1/2} \sum_{k, k'=-\infty}^{\infty} x[k] \bar{x}[k'] \bar{g}[k-n] e^{-2i\pi v(k-n)} g[k'-n] e^{2i\pi v(k'-n)} \\ &= \sum_{k, k'=-\infty}^{\infty} x[k] \bar{x}[k'] \bar{g}[k-n] g[k'-n] \int_{-1/2}^{1/2} e^{2i\pi v(k'-k)} dv \\ &= \sum_{k=-\infty}^{\infty} |x[k]|^2 |g[k-n]|^2 . \end{aligned}$$

De là, on tire

$$\sum_{n=-\infty}^{\infty} \int_{-1/2}^{1/2} |\mathcal{V}_g x[v, n]|^2 dv = \sum_{k, n=-\infty}^{\infty} |x[k]|^2 |g[k-n]|^2 = \sum_{k, k'=-\infty}^{\infty} |x[k]|^2 |g[k']|^2 = \|g\|^2 \|x\|^2 ,$$

ce qui est le résultat recherché.

**Correction de l'exercice 3.3 (Symétries de la TFCT)**

1. Soient  $x \in \mathbb{C}^L$ , et  $y$  une copie translatée de  $x$  donnée par  $y[\ell] = x[\ell - \tau]$  avec  $\tau \in \mathbb{Z}$ , et  $\ell - \tau$  étant compris modulo  $L$  (conditions aux bords périodiques de période  $L$ ). On a

$$\mathcal{V}_g y[m, n] = \sum_{\ell=-\infty}^{\infty} x[\ell - \tau] \bar{g}[\ell - n] e^{-2i\pi m(\ell - n)/L} = \sum_{k=-\infty}^{\infty} x[k] \bar{g}[k + \tau - n] e^{-2i\pi m(k + \tau - n)/L} = \mathcal{V}_g y[m, n - \tau] .$$

La TFCT possède donc une propriété d'invariance par translation : la TFCT d'une copie translatée d'un signal est une copie translatée (en temps, d'un même décalage) de la TFCT du signal original. Par conséquent, comme le signal est assimilé à un signal périodique de période  $L$ , la TFCT possède aussi cette périodicité (en temps) :

$$\mathcal{V}_g x[m, n \pm L] = \mathcal{V}_g x[m, n] .$$

2. Périodicité fréquentielle de  $\mathcal{V}_g x$  : on voit facilement que

$$\mathcal{V}_g x[m \pm L, n] = \mathcal{V}_g x[m, n] .$$

3. Symétrie Hermitienne : supposons maintenant que  $x$  et  $g$  soient réels, i.e.  $x, g \in \mathbb{R}^L$ . On a alors

$$\mathcal{V}_g x[L - m, n] = \sum_{\ell=0}^{L-1} x[\ell] g[\ell - n] e^{-2i\pi(L-m)(\ell-n)/L} = \sum_{\ell=0}^{L-1} x[\ell] g[\ell - n] e^{2i\pi m(\ell-n)/L} = \overline{\mathcal{V}_g x[m, n]} .$$

**Correction de l'exercice 3.4 (Transformation de Gabor et banc de filtres)**

Ecrivons la transformée de Gabor

$$\mathcal{V}_g x[m, n] = \sum_{\ell=0}^{L-1} x[\ell] \bar{g}[\ell - n] e^{-2i\pi m b (\ell - n) / L}$$

ceci peut aussi s'écrire sous une forme faisant intervenir un produit de convolution (donc un filtrage)

$$\mathcal{V}_g x[m, n] = (x * \tilde{g}_m)[na],$$

la réponse impulsionnelle du filtre étant

$$\tilde{g}_m[\ell] = \bar{g}[L - \ell] e^{2i\pi m b \ell / L}.$$

Le filtrage est suivi d'un sous-échantillonnage d'un facteur  $a$ . Les fonctions de transfert des filtres sont données par les transformées de Fourier des réponses impulsionnelles.

**Correction de l'exercice 3.5 (DGT et Fourier)**

Commençons par calculer la transformée de Fourier finie de  $g_{mn}$  :

$$\begin{aligned} \widehat{g_{mn}}[k] &= \sum_{\ell=0}^{L-1} g[\ell - na] e^{2i\pi m b (\ell - na) / L} e^{-2i\pi k \ell / L} \\ &= e^{-2i\pi k na / L} \sum_{\ell=0}^{L-1} g[\ell - na] e^{-2i\pi (k - mb) (\ell - na) / L} \\ &= e^{-2i\pi k na / L} \sum_{\ell'=0}^{L-1} g[\ell'] e^{-2i\pi (k - mb) \ell' / L} \\ &= e^{-2i\pi k na / L} \hat{g}[k - mb] \end{aligned}$$

de là, on peut écrire

$$\begin{aligned} \mathcal{V}_g x[m, n] &= \langle \hat{x}, \widehat{g_{mn}} \rangle \\ &= \sum_{k=0}^{L-1} \hat{x}[k] \widehat{g_{mn}}[k - mb] e^{2i\pi k na / L} \\ &= e^{2i\pi m n a b / L} \sum_{k=0}^{L-1} \hat{x}[k] \widehat{g}[k - mb] e^{2i\pi (k - mb) na / L} \\ &= e^{2i\pi m n a b / L} \mathcal{V}_{\hat{g}} \hat{x}[-n, m] \end{aligned}$$

Il s'agit bien d'une transformée de Gabor de  $\hat{x}$ , avec une fenêtre  $\hat{g}$ , à un facteur de phase  $\exp(2i\pi m n a b / L)$  près.

**Correction de l'exercice 3.6 (Inversion sans peine de la DGT)**

Pour démontrer la proposition 3.9, commençons par calculer

$$\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mathcal{V}_g x[m, n] g_{mn}[k] = \sum_{\ell=0}^{L-1} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[\ell] \bar{g}[\ell - na] g[k - na] e^{2i\pi m b (k - \ell) / L},$$

et focalisons nous sur la somme sur  $m$  qui, puisque  $M = L/b$ , est de la forme

$$\sum_{m=0}^{M-1} e^{2i\pi m (k - \ell) / M} = \sum_{m=0}^{M-1} \left( e^{2i\pi (k - \ell) / M} \right)^m = \begin{cases} M & \text{si } k - \ell \text{ est multiple de } M \\ 0 & \text{sinon} \end{cases}$$

(dans le dernier cas c'est la somme des  $M$  racines  $M$ -ièmes de l'unité, qui est nulle). Les multiples de  $M$  sont  $0, \pm M, \pm 2M, \dots$ . Or, si le support de  $g$  est de longueur inférieure ou égale à  $M$ , le produit  $\bar{g}[\ell - na] g[k - na]$  sera toujours uniformément nul si  $k - \ell$  est égal à  $\pm M, \pm 2M, \dots$ . Donc les seuls termes qui contribuent à cette somme satisfont  $k = \ell$ , de sorte que

$$\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mathcal{V}_g x[m, n] g_{mn}[k] = M \sum_{n=0}^{N-1} |g[\ell - na]|^2 x[k].$$

On voit ainsi que l'opérateur de trame  $\mathcal{R}$  agit par multiplication par le vecteur  $M \sum_{n=0}^{N-1} |g[\ell - na]|^2$ , donc l'opérateur inverse est l'opérateur de multiplication par  $(M \sum_{n=0}^{N-1} |g[\ell - na]|^2)^{-1}$ . La fenêtre duale canonique est ainsi donnée par

$$\gamma[\ell] = \frac{g[\ell]}{M \sum_{n=0}^{N-1} |g[\ell - na]|^2} .$$

### Correction de l'exercice 3.7 (Projection sur le sous-espace des représentations temps-fréquence valides)

Calculons

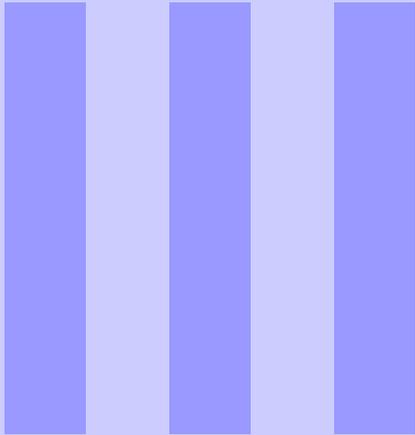
$$\Pi^2 = \mathcal{V}_g \mathcal{V}_\gamma^* \mathcal{V}_g \mathcal{V}_\gamma^* .$$

Comme  $\mathcal{V}_\gamma^* \mathcal{V}_g = \mathbf{I}$ , on a bien  $\Pi^2 = \Pi$ .

Par ailleurs, si  $\gamma = g$ , alors

$$\Pi^* = (\mathcal{V}_g \mathcal{V}_g^*)^* = \mathcal{V}_g^{**} \mathcal{V}_g^* = \mathcal{V}_g \mathcal{V}_g^* = \Pi ,$$

dans ce cas le projecteur est bien orthogonal.



# Compléments

**Bibliographie** ..... 85

**Index** ..... 87





## Bibliographie

- [1] M. Bergounioux. *Mathématiques pour le traitement du signal : Cours et exercices corrigés*, Paris, Dunod, , 2014, 2e éd. Dunod, 2nd edition, 2014.
- [2] C. Févotte, N. Bertin, and J.-L. Durrieu. Nonnegative Matrix Factorization with the Itakura-Saito Divergence : With Application to Music Analysis. *Neural Computation*, 21(3) :793–830, 03 2009.
- [3] Z. Prusa, P. L. Soendergaard, N. Holighaus, C. Wiesmeyer, and P. Balazs. The Large Time-Frequency Analysis Toolbox 2.0. In *Sound, Music, and Motion*, LNCS, pages 419–442. Springer International Publishing, 2014.
- [4] J. O. Smith. *Introduction to Digital Filters, with audio applications*. Stanford University, Center for Computer Research in Music and Acoustics (CCRMA), 2007.
- [5] J. O. Smith. *Spectral Audio Signal Processing*. Stanford University, Center for Computer Research in Music and Acoustics (CCRMA), 2011.
- [6] P. L. Soendergaard, B. Torrèsani, and P. Balazs. The Linear Time Frequency Analysis Toolbox. *International Journal of Wavelets, Multiresolution Analysis and Information Processing*, 10(4), 2012.
- [7] B. Torrèsani. *Méthodes Mathématiques pour le Traitement du Signal*. Université de Provence, 2010.
- [8] M. Vetterli, J. Kovacevic, and V. K. Goyal. *Foundations of Signal Processing*. Cambridge University Press, 2014.
- [9] Wikipedia. Traitement numérique du signal. Encyclopédie en ligne, Septembre 2019.
- [10] Wikipedia. Décomposition en valeurs singulières. Encyclopédie en ligne, Janvier 2021.





## Index

- Équations de Yule Walker, 31
- Algorithme de Levinson-Durbin, 32
- Atome de Gabor, 41
- Atome temps-fréquence, 36
- DGT inverse sans effort, 44
- Divergence, 53
- Espace  $\ell^2(\mathbb{Z})$ , 10
- Fenêtre, 35
- Fenêtre duale canonique, 43, 49
- Filtrage temps-fréquence, 45
- Filtre de Butterworth, 23
- Filtre numérique, 18
- Filtre numérique causal, 18
- Filtre numérique d'ordre fini, 22
- Filtre numérique réalisable, 18
- Filtre numérique stable, 18
- Filtre récursif, 21
- Fonction de transfert, 20
- Fonction de transfert d'un filtre numérique, 22
- Fréquence, 11
- Fréquence d'échantillonnage, 9
- Gain du filtre, 22
- Impédance, 20
- Largeur de bande relative, 49
- Masque, 45
- Matrice orthogonale, 52
- Matrice unitaire, 52
- Multiplicateur de Gabor, 45
- Méthode des résidus, 18
- Ondelette, 49
- Opérateur de trame, 42
- Pulsation, 11, 18
- Réponse impulsionnelle, 18
- Représentation pôles-zéros, 22
- Réponse fréquentielle, 20
- Série de Laurent, 15
- Signal aléatoire du second ordre, 26
- Signal aléatoire stationnaire en moyenne quadratique, 26
- Signal causal, 15
- Spectre, 25
- Spectrogramme, 36
- TFCT inverse, 39, 40
- TFD, 11
- TFF, 13
- Théorème de Cramèr, 35
- Trame, 43
- Trame de Gabor, 43
- Transformation de Fourier Discrète, 11
- Transformation de Fourier discrète inverse, 11
- Transformation de Fourier Finie, 13
- Transformation de Gabor, 41
- Transformation de Gabor inverse, 43
- Transformation en  $z$ , 15
- Transformation en  $z$  inverse, 16
- Transformation à  $Q$  constant, 50

Valeurs singulières, 52

Échantillonnage, 9