



## **Une approche par la synthèse et la parcimonie de quelques problèmes inverses en audio**

M. Kowalski

Univ Paris-Sud, L2S, Supelec, 3, rue Joliot-Curie, 91192 Gif Sur Yvette, France  
matthieu.kowalski@lss.supelec.fr

On présente quelques problèmes inverses linéaires classiques en audio, tels que la restauration des signaux (débruitage, des-écrêtage) ou la séparation de sources (en particulier, les mélanges convolutifs sous-déterminés, c'est-à-dire lorsqu'il y a plus de sources à estimer que de mélanges observés). A partir de ces exemples, on verra comment une approche par synthèse dans un, voire une union de, dictionnaire(s) de Gabor, allié au principe de parcimonie, permet d'attaquer ces problèmes en vue de leur inversion. En effet, la construction d'un signal par une combinaison linéaire d'atomes temps-fréquence permet d'introduire facilement des *a priori* sur leurs représentations, et en particulier des *a priori* parcimonieux et structurés. Le cadre de l'optimisation convexe (et semi-convexe) permet de présenter un principe général pour la construction de fonctionnelles, que l'on peut optimiser de manière efficace avec des algorithmes simples de seuillage itératifs. Les performances de ces méthodes seront illustrées sur les problèmes considérés.

## 1 Introduction

D'un point de vue de traicteur de signal, un problème inverse consiste à estimer un signal originel à partir d'observations, souvent incomplètes et bruitées, d'un ou plusieurs signaux, liés physiquement. On s'intéresse ici aux problèmes inverses linéaires, qu'on modélise par le modèle suivant : soit  $\mathbf{y} \in \mathbb{R}^M$  et  $\mathbf{s} \in \mathbb{R}^N$ . Soit  $\mathbf{A} : \mathbb{R}^N \rightarrow \mathbb{R}^M$  un opérateur linéaire tel que :

$$\mathbf{y} = \mathbf{A}\mathbf{s} + \mathbf{b}$$

où  $\mathbf{b} \in \mathbb{R}^M$  représente un bruit additif quelconque. Dans ce modèle, les observations sont représentées par  $\mathbf{y}$ . Dans beaucoup de problèmes, l'opérateur  $\mathbf{A}$  peut être modélisé (c'est le *problème direct*), et l'on cherche à estimer  $\mathbf{s}$ . Cependant,  $\mathbf{A}$  n'est pas toujours connu, et peut même devenir l'objet d'intérêt, connaissant  $\mathbf{s}$  et  $\mathbf{y}$ . Lorsque ni  $\mathbf{A}$  ni  $\mathbf{s}$  ne sont connus, on parle de problème *aveugle*. Dans la majorité des cas, on suppose l'opérateur  $\mathbf{A}$  linéaire. Enfin, on distinguera aussi deux cas importants : le cas sur-déterminé, lorsque  $M > N$  et le cas sous-déterminé lorsque  $M < N$ .

Un cadre pratique pour aborder les problèmes inverses consiste à le poser sous forme d'un problème d'optimisation :

$$\hat{\mathbf{s}} = \operatorname{argmin} \mathcal{L}(\mathbf{y}, \mathbf{A}, \mathbf{s}) + \mathcal{P}(\mathbf{s}) \quad (1)$$

où  $\mathcal{L}$  est un terme d'attache aux données modélisant le lien entre les observations  $\mathbf{y}$  et les signaux  $\mathbf{s}$  à travers l'opérateur  $\mathbf{A}$ , et  $\mathcal{P}$  un terme de régularisation, qui reflète un *a priori* sur les données. Ce cadre s'adapte parfaitement à l'approche *synthèse*. Il est en effet pratique de modéliser le signal d'intérêt  $\mathbf{s}$  comme une somme d'atomes élémentaires choisis dans un dictionnaire tel que  $\mathbf{s} = \mathbf{\Phi}\boldsymbol{\alpha}$ , où  $\mathbf{\Phi} \in \mathbb{C}^{NK}$  est l'opérateur de synthèse associé au dictionnaire, et  $\boldsymbol{\alpha} \in \mathbb{C}^K$  les coefficients de synthèse. Le problème d'optimisation devient alors

$$\hat{\boldsymbol{\alpha}} = \operatorname{argmin}_{\boldsymbol{\alpha} \in \mathbb{C}^{NK}} \mathcal{L}(\mathbf{y}, \mathbf{A}, \mathbf{\Phi}\boldsymbol{\alpha}) + \mathcal{P}(\boldsymbol{\alpha}) \quad (2)$$

et  $\hat{\mathbf{s}} = \mathbf{\Phi}\hat{\boldsymbol{\alpha}}$ . Le terme de régularisation porte alors sur les coefficients de synthèse du signal.

Le choix du dictionnaire varie selon le signal. Pour les images, les transformées en ondelette sont bien adaptées, tandis que pour l'audio, les dictionnaires temps-fréquence de type Gabor semblent naturels. Le cadre théorique pratique de construction des dictionnaires est celui des *frames*. L'article [1] offre un point de vue général sur les possibilités offertes. Lorsque le dictionnaire est bien choisi, le choix d'un *a priori* de parcimonie de type  $\ell_1$  permet de bonnes estimations, connu comme l'estimateur du Lasso [2] ou basis pursuit denoising [3].

On s'intéresse ici à deux problèmes inverses particuliers en audio : la séparation de sources et la restauration des signaux audio écrêtés. À travers ces deux exemples, on montrera comment les approches par optimisation avec un choix de parcimonie structurée peuvent être utilisées avec succès.

Le reste de l'article est organisé comme suit. La section 2 introduit les deux problèmes considérés : d'abord le problème direct, puis la fonctionnelle de type (2) pour l'inverser. La section 3 expose d'abord l'approche par optimisation convexe et l'algorithme de seuillage itératif utilisée en pratique, pour introduire des opérateurs de seuillage correspondant à un problème d'optimisation non convexe et finir sur des opérateurs de seuillage structurés. Enfin, la section 4 expose les résultats obtenus en pratiques en séparation de sources et en restauration audio.

## 2 Deux problèmes inverses en audio

On présente dans cette section deux problèmes en audio : la séparation de sources dans le cas sous-déterminé convolutif mais non-aveugle, et la restauration d'un signal audio écrêté. On formule pour chacun de ces deux problèmes, en plus du problème direct, le choix du problème d'optimisation utilisé pour résoudre le problème inverse.

### 2.1 La séparation de source

Le problème de séparation de source consiste à estimer les signaux composant un mélange, connu aussi sous le nom du problème de la « cocktail party ». Ainsi, le  $m$ -ième mélange  $x_m(t)$  est donné par

$$x_m(t) = \sum_{n=1}^N A_{mn} \star s_n(t) + e_m(t), \quad (3)$$

où  $N$  est le nombre de signaux sources  $s_n$  de durée  $T$  et  $M$  ( $M < N$ ) le nombre de microphones, donnant  $M$  canaux de mélange  $\mathbf{x}_m$ ,  $\star$  étant l'opérateur de convolution. Les effets de l'acoustique de propagation entre les sources et les microphones sont modélisés par un ensemble de filtres de mélange  $A_{mn}(t)$  de longueur  $P$ . En notant  $\mathbf{x} \in \mathbb{R}^{M \times T}$  et  $\mathbf{s} \in \mathbb{R}^{N \times T}$  les matrices de signaux mélangés et les signaux sources, et par  $\mathbf{A} \in \mathbb{R}^{M \times N \times P}$  l'opérateur contenant les filtres de mélanges, le processus de mélange (3) peut s'écrire de manière compacte comme

$$\mathbf{x} = \mathbf{A} \star \mathbf{s} + \mathbf{e}, \quad (4)$$

où  $\mathbf{e} \in \mathbb{R}^{M \times T}$  modélise le bruit de fond.

On s'intéresse ici au cas *sous-déterminé*, où le nombre de sources à estimer est plus grand que le nombre de mélange (ie  $N > M$ ). L'état de l'art de la séparation de sources dans le cas sous-déterminé utilise le domaine temps-fréquence et s'appuie sur une approximation en bande étroite, afin d'approcher la convolution par une multiplication complexe pour chaque bande de fréquence. En notant  $\tilde{\mathbf{x}} = \mathbf{\Phi}^* \mathbf{x}$ ,  $\tilde{\mathbf{s}} = \mathbf{\Phi}^* \mathbf{s}$  et  $\hat{\mathbf{A}}$  l'opérateur contenant les transformées de Fourier des filtres, on a :

$$\tilde{\mathbf{x}}_{t,f} \simeq \mathbf{A}(f) \tilde{\mathbf{s}}_{t,f} . \quad (5)$$

La tâche de séparation est généralement décomposée en deux sous-tâches successives. D'abord, les matrices de mélanges pour chaque fréquence sont estimées par clustering des coefficients temps-fréquences des mélanges, basé sur les directions d'arrivées des sons. Les coefficients temps-fréquence des sources sont alors séparés pour chaque bande fréquentielle, typiquement en minimisant une fonction de coût, soit sous la contrainte qu'une seule source est active par point temps-fréquence [4, 5], méthode connue comme le masquage temps-fréquence binaire, ou en minimisant un coût  $\ell_1$  sur les sources [6, 7], pour favoriser leur parcimonie. Ces coûts implémentent l'hypothèse que les représentations temps-fréquences sont disjointes ou parcimonieuse. Cependant, une évaluation récente [8] montre que ces méthodes ont une performance limitée dans les environnements réverbérants réalistes. Ici, on se focalise sur la seconde sous-tâche, c'est à dire l'estimation des signaux sources *en supposant que les filtres de mélanges  $A_{mn}$  sont connus*.

Un moyen simple pour contourner l'hypothèse en bande étroite, est de remplacer le modèle de mélange approché (5) par un terme d'attache aux données dans le model temporel (4), où les signaux sources temporel sont synthétisés par leurs coefficients temps-fréquence. Avec une pénalité  $\ell_1$  reflétant la parcimonie de ces coefficients de synthèse, cela donne la fonctionnelle large-bande :

$$\min_{\alpha \in \mathbb{C}^{N \times K}} \frac{1}{2} \|\mathbf{x} - \mathbf{A} \star \alpha \mathbf{\Phi}^*\|_2^2 + \lambda \|\alpha\|_1 . \quad (\text{W-Lasso})$$

Enfin, les signaux audio présentent en général deux composantes : une composante transitoire, bien localisée en temps (typiquement, les sons percussifs) et une composante tonale, bien localisée en fréquence. Le choix d'une transformée de Gabor impose de faire un compromis en localisation temporelle et la localisation fréquentielle. On construit alors un dictionnaire par l'union de deux transformées de Gabor, afin de prendre en compte à la fois les composantes transitoires et les composantes tonales. Le problème d'optimisation devient alors :

$$\alpha_{trans}, \alpha_{ton} = \operatorname{argmin}_{\alpha_{trans} \in \mathbb{C}^{N \times K_1}, \alpha_{ton} \in \mathbb{C}^{N \times K_2}} \frac{1}{2} \|\mathbf{x} - \mathbf{A} \star (\alpha_{trans} \mathbf{\Phi}_{trans}^* + \alpha_{ton} \mathbf{\Phi}_{ton}^*)\|_2^2 + \lambda_{trans} \|\alpha_{trans}\|_1 + \lambda_{ton} \|\alpha_{ton}\|_1 . \quad (\text{H-W-Lasso})$$

Le signaux sources sont alors resynthétisés :  $\mathbf{s} = \alpha_{ton} \mathbf{\Phi}_{ton}^* + \alpha_{trans} \mathbf{\Phi}_{trans}^*$ .

## 2.2 Le désécrêtage : un problème de restauration

Un problème de restauration audio important est de retrouver les échantillons manquants ou corrompus. On

s'intéresse ici particulièrement au désécrêtage. Lorsque l'amplitude d'un signal audio dépasse un certain seuil, le signal est alors tronqué et un phénomène de saturation apparaît.

Les approches classiques pour le désécrêtage sont les modèles autoregressif (AR) [9], ou l'estimation Bayésienne [10]. Dans [11], une formulation utilisant la parcimonie a été proposée, résolue par un matching poursuite orthogonal (OMP). Les auteurs ont appelé cette méthode *audio inpainting* en référence au problème rencontré en image [12].

Lorsqu'un signal audio est écrêté, on connaît la facilement la valeur du seuil, et les échantillons reconstruits doivent dépasser cette valeur. Pour écrire le problème directe, on sépare les échantillons observés en deux classes : les échantillons non écrêtés notés  $\mathbf{y}^r \in \mathbb{R}^M$  et les échantillons écrêtés notés  $\theta^{clip} \in \mathbb{R}^{T-M}$ . Ces derniers échantillons ne prennent que deux valeurs connues :  $\theta^{clip}$  ou  $-\theta^{clip}$ ,  $\theta^{clip}$  étant la valeur du seuil de saturation. Le problème directe s'écrit donc

$$\mathbf{y}^r = \mathbf{M}^r \mathbf{s} \quad \text{et} \quad \theta^{clip} \mathbf{c} = \mathbf{M}^c \mathbf{s}$$

En utilisant le principe de parcimonie dans un dictionnaire de Gabor, les auteurs de [11] ont proposé le problème suivant :

$$\hat{\alpha} = \operatorname{argmin}_{\alpha} \|\alpha\|_0 \quad (6)$$

$$\text{s.t. } \|\mathbf{y}^r - \mathbf{M}^r \mathbf{\Phi} \alpha\|_2^2 \leq \epsilon \quad \text{and} \quad |\mathbf{M}^c \mathbf{\Phi} \alpha| \geq |\theta^{clip}|$$

On propose de reformuler le problème sous la forme d'un problème d'optimisation non contraint. Pour cela, on utilise la fonction charnière au carré, connue en classification [13], définie comme

$$h^2 : \mathbb{R} \longrightarrow \mathbb{R}_+ \quad z \mapsto h^2(z) = \begin{cases} z^2 & \text{si } z < 0 \\ 0 & \text{si } z \geq 0 \end{cases}$$

En l'application à  $z = x - \theta^{clip}$ , cette fonction laisse  $x$  libre si  $|x| \geq \theta^{clip}$ , et le pénalise sinon. Avec la notation

$$[\theta^{clip} - \mathbf{x}]_+^2 = \sum_{k: \theta_k^{clip} > 0} h^2(x_k - \theta_k^{clip}) + \sum_{k: \theta_k^{clip} < 0} h^2(\theta_k^{clip} - x_k)$$

On introduit alors la fonctionnelle suivante

$$\operatorname{argmin}_{\alpha \in \mathbb{C}^K} \frac{1}{2} \|\mathbf{y}^r - \mathbf{M}^r \mathbf{\Phi} \alpha\|_2^2 + \frac{1}{2} [\theta^{clip} - \mathbf{M}^c \mathbf{\Phi} \alpha]_+^2 + \lambda \|\alpha\|_1 \quad (7)$$

qui pourra se résoudre par les algorithmes de seuillage itératifs classiques, présentés dans la section suivante.

## 3 Les algorithmes de seuillage itératif

On présente ici les algorithmes de seuillage itératif utilisés pour résoudre les problèmes formulés selon le problème d'optimisation(1) qu'on rappelle ici :

$$\hat{\alpha} = \operatorname{argmin}_{\alpha} \mathcal{L}(\alpha) + \mathcal{P}(\alpha) \quad (8)$$

où le terme d'attache aux données est supposé convexe et  $L$ -Lipschitz différentiable, tandis que le terme de régularisation  $\mathcal{P}$  est supposé convexe, non forcément différentiable. On

vérifie facilement que les deux fonctionnelles (H-W-Lasso) et (7) respectent ce cadre.

Cette classe de fonctionnelle s'optimise aisément grâce aux algorithmes de type forward-backward [14] aussi appelés ISTA [15] (Iterative Shrinkage/Thresholding Algorithms). Ils reposent essentiellement sur la notion d'opérateur de proximité introduit par Moreau [16].

### 3.1 Opérateurs de proximités et ISTA

On rappelle la définition des opérateurs de proximité.

**Définition 1** (Opérateur de proximité). Soit  $\varphi : \mathbb{C}^I \rightarrow \mathbb{C}$  une fonction convexe. L'opérateur de proximité associé à  $\varphi$ , noté  $\text{prox}_\varphi : \mathbb{C}^I \rightarrow \mathbb{C}^I$  est donné par

$$\text{prox}_\varphi(\mathbf{z}) = \frac{1}{2} \underset{\mathbf{u} \in \mathbb{C}^I}{\text{argmin}} \|\mathbf{z} - \mathbf{u}\|_2^2 + \varphi(\mathbf{u}). \quad (9)$$

Par exemple, l'opérateur de proximité associé à la norme  $\ell_1$  est l'opérateur de seuillage doux

**Proposition 1** (Prox de  $\lambda\|\cdot\|_1$ ). Soit  $\mathbf{z} \in \mathbb{C}^I$ . Alors,  $\mathbf{u} = \text{prox}_{\lambda\|\cdot\|_1}(\mathbf{z})$  est donné par

$$u_i = \frac{z_i}{|z_i|} (|z_i| - \lambda)^+$$

où  $(z)^+ = \max(0, z)$ .

On peut aussi calculer les opérateurs de proximité des normes mixtes  $\ell_{21}$  et  $\ell_{12}$ , qui permettent de structurer les coefficients [17].

L'algorithme ISTA est rappelé dans l'algorithme 1. Cet algorithme est connu pour converger vers une solution de (8), avec une vitesse de convergence en  $\mathcal{O}(\frac{1}{k})$  où  $k$  est le nombre d'itérations [18]. Une version accélérée FISTA [19] a été proposée qui converge bien plus rapidement en  $\mathcal{O}(\frac{1}{k^2})$ . FISTA est donné par l'algorithme 2

---

#### Algorithm 1: ISTA [20, 14]

---

Initialization :  $\alpha^{(0)} \in \mathbb{C}^{N \times B}$ ,  $k = 1$ .

**repeat**

$$\alpha^{(k)} = \text{prox}_{\frac{\lambda}{2}\mathcal{P}}(\alpha^{(k-1)} - \frac{\nabla \mathcal{L}(\alpha^{(k-1)})}{L});$$

**until** convergence;

---



---

#### Algorithm 2: FISTA [19]

---

Initialization :  $\alpha^{(0)} \in \mathbb{C}^{N \times B}$ ,  $k = 1$ .

**repeat**

$$\alpha^{(k)} = \text{prox}_{\frac{\lambda}{2}\mathcal{P}}(\mathbf{z}^{(k-1)} - \frac{\nabla \mathcal{L}(\mathbf{z}^{(k-1)})}{L});$$

$$\tau^{(k)} = \frac{1 + \sqrt{1 + 4\tau^{(k-1)^2}}}{2};$$

$$\mathbf{z}^{(k)} = \alpha^{(k)} + \frac{\tau^{(k-1)} - 1}{\tau^{(k)}} (\alpha^{(k)} - \alpha^{(k-1)})$$

**until** convergence;

---

### 3.2 Fonctions de seuillage

On étend maintenant l'algorithme ISTA aux fonctions de seuillage générales, qui permettent de travailler directement sur la façon de seuiller les coefficients plutôt que de choisir la pénalité  $\mathcal{P}$ .

**Définition 2** (Règle de seuillage).  $\mathbb{S}(\cdot; \lambda)$  est une règle de seuillage ssi

1.  $\mathbb{S}(\cdot; \lambda)$  est une fonction impaire.  $\mathbb{S}_+(\cdot; \lambda)$  représente sa restriction à  $\mathbb{R}_+$
2.  $0 \leq \mathbb{S}_+(x; \lambda) \leq x$ ,  $\forall x \in \mathbb{R}_+$
3.  $\mathbb{S}_+$  est croissante sur  $\mathbb{R}_+$ , et  $\lim_{x \rightarrow +\infty} \mathbb{S}_+(x; \lambda) = +\infty$

Il est montré dans [21], que l'on peut associer une pénalité  $\mathcal{P}$  à n'importe quelle règle de seuillage :

$$\mathcal{P}(x; \lambda) = \int_0^{|x|} \mathbb{S}_+^{-1}(t; \lambda) - x \, dt.$$

On peut alors montrer qu'une telle pénalité est  $c$ -semi-convexe, c'est-à-dire qu'il existe une constante  $c \leq 1$  telle que la fonction  $\mathcal{P}(\cdot; \lambda) + \frac{c}{2} \|\cdot\|^2$  est convexe. Plus précisément, on a  $c \leq 1 - \frac{1}{\|\mathbb{S}_+^{-1}(\cdot; \lambda)\|_\infty}$  [22].

### 3.3 Quelques fonctions de seuillages

On donne ici trois exemples de fonctions de seuillage connues : le seuillage dur, le seuillage SCAD et le seuillage nonnegative Garotte (NNG), ainsi que les pénalités (non convexes) associées. Ces règles de seuillages sont illustrées sur la figure 1. On remarque que les seuillages SCAD et NNG peuvent être vus comme des « compromis » entre le seuillage doux et le seuillage dur.

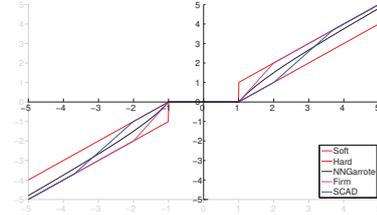


FIGURE 1 – Quelques fonctions de seuillage.

**Seuillage dur :**  $\mathbb{S}(x; \lambda) = \begin{cases} 0 & \text{if } |x| \leq \lambda \\ 1 & \text{if } |x| > \lambda \end{cases}$

$$P'(x, \lambda) = \begin{cases} -x + \lambda \text{sgn}(x) & \text{si } |x| \leq \lambda \\ 0 & \text{si } |x| > \lambda \end{cases} \quad P(x, \lambda) = \begin{cases} -\frac{x^2}{2} + \lambda|x| & \text{si } |x| \leq \lambda \\ \frac{\lambda^2}{2} & \text{si } |x| > \lambda \end{cases}$$

Comme  $P$  n'est pas partout différentiable, on en déduit que  $P$  est 1-semi-convex.

**SCAD :**  $\mathbb{S}(x; \lambda) = \begin{cases} x(1 - \frac{\lambda}{x})^+ & \text{if } |x| \leq 2\lambda \\ \frac{x}{a-2} (a-1 - \frac{a\lambda}{|x|}) & \text{if } 2\lambda < |x| \leq a\lambda \\ x & \text{if } |x| > a\lambda \end{cases}$

with  $a > 2$ .

$$P'(x) = \begin{cases} \lambda & \text{if } x \leq \lambda \\ \frac{(a\lambda-x)}{a-1} & \text{if } \lambda < x \leq a\lambda \\ 0 & \text{if } x > a\lambda \end{cases} \quad P(x; \lambda) = \begin{cases} \lambda x & \text{if } x \leq \lambda \\ \frac{(a\lambda x - x^2)/2}{a-1} & \text{if } \lambda < x \leq a\lambda \\ a\lambda & \text{if } x > a\lambda \end{cases}$$

En regardant  $P'$ , on déduit que la pénalité SCAD est  $c$ -semi-convexe avec  $c = \frac{1}{a-1}$ .

**Nonnegative Garrote :**  $\mathbb{S}(x; \lambda) = x \left(1 - \frac{\lambda^2}{x^2}\right)^+$

$$P'(x) = \frac{2\lambda^2}{\sqrt{x^2 + 4\lambda^2} + |x|} \operatorname{sgn}(x)$$

$$P(x; \lambda) = \lambda^2 + \operatorname{asinh}\left(\frac{|x|}{2\lambda}\right) + \lambda^2 \frac{|x|}{\sqrt{x^2 + 4\lambda^2} + |x|}$$

En regardant  $P'$ , on déduit que la pénalité NNG est  $c$ -semi-convexe avec  $c = \frac{1}{2}$ .

Enfin, il est montré dans [22] qu'on peut utiliser n'importe quelle fonctions de seuillage dans un algorithme ISTA relaxé, rappelé par l'algorithme 3

**Théorème 1.** *Soit  $\mathbb{S}$  une règle de seuillage et  $\mathcal{L}$  une fonction convexe  $L$ -Lipschitz différentiable. Alors ISTA, ie. Alg.3 avec  $\gamma = 0$ , converge. De plus, si  $\mathbb{S}' \leq a$ , alors ISTA relaxé, ie. Alg.3, avec  $\gamma < \frac{1}{a}$  converge.*

La régularisation n'étant plus convexe, on perd les résultats sur la vitesse de convergence de l'algorithme. Cependant, le paramètre de relaxation permet en pratique d'accélérer cette convergence.

---

**Algorithm 3:** ISTA relaxé

---

**repeat**

$$\begin{cases} x^{(k+1)} = \mathbb{S}_P \left( z^{(k)} - \frac{1}{L} \nabla \mathcal{L}(z^{(k)}); \lambda/L \right); \\ z^{(k+1)} = x^{(k+1)} + \gamma^{(k+1)} (x^{(k+1)} - x^{(k)}); \end{cases}$$

**until** convergence;

---

Ces opérateurs ont été introduits pour limiter l'effet de « shrinkage » des coefficients induit par le lasso et le seuillage doux. Ainsi, les coefficients d'amplitude élevée sont laissés quasiment inchangés, et les résultats obtenus avec cet opérateur sont à rapprocher de ceux obtenus après une étape de débiaisage [23].

La dernière étape consiste à introduire plus de structure dans les opérateurs de seuillage, les coefficients temps-fréquence faisant clairement apparaître une certaine persistance temporelle ou fréquentielle.

### 3.4 Seuillage structuré

Par extension de l'opérateur de seuillage doux usuel correspondant à l'opérateur de proximité de la pénalité  $\ell_1$ , qui apparaît dans (7), il devient alors possible d'exploiter des propriétés de persistance du signal par un système de voisinage temps-fréquence [24]. On construit alors l'opérateur de seuillage du group-Lasso fenêtré (windowed group Lasso – WGL), et le seuillage de Wiener Empirique persistant (Persistent Empirical Wiener – PEW), qui est la version structurée du NNG. Plus formellement, ces opérateurs de seuillages ont les expressions suivantes :

- WGL :

$$\tilde{\alpha}_{tf} = \mathbb{S}_\lambda^{\text{WGL}}(\alpha_{tf}) = \alpha_{tf} \left( 1 - \frac{\lambda}{\sqrt{\sum_{t' \in \mathcal{N}(t)} |\alpha_{t'f}|^2}} \right)^+$$

où  $\mathcal{N}(t)$  désigne le voisinage de l'indice  $t$ .

- PEW :

$$\tilde{\alpha}_{tf} = \mathbb{S}_\lambda^{\text{PEW}}(\alpha_{tf}) = \alpha_{tf} \left( 1 - \frac{\lambda^2}{\sum_{t' \in \mathcal{N}(t)} |\alpha_{t'f}|^2} \right)^+$$

où  $\mathcal{N}(t)$  désigne le voisinage de l'indice  $t$ .

L'opérateur WGL permet de prendre une décision *coefficients par coefficients* en prenant en compte son voisinage. En effet, l'énergie est calculée sur tout un voisinage, puis comparée à un seuil. Ainsi, un coefficient d'énergie relativement élevée mais isolé pourra être mis à zéro, tandis qu'un coefficient de faible énergie au milieu de coefficients « fort » pourra être conservé. Le but sera par la suite de favoriser des structures temporelles dans les dictionnaires temps-fréquence. Enfin, l'opérateur PEW a été construit sur l'opérateur WGL pour les mêmes raisons que le NNG : limiter le seuillage des forts coefficients.

## 4 Résultats

### 4.1 Séparation de sources

Pour les expériences en séparation de sources, les signaux audio ont été échantillonnés à 11 kHz, et les filtres de mélanges ont été simulés par le logiciel Roomsim [25]. Le nombre de microphone et le nombre de sources sont respectivement de 2 et 4. On donne les résultats pour des microphones espacés de  $d = 1$  m et un temps de réverbération de  $RT_{60} = 250$  ms pour différents ensembles de sources de paroles ou de musiques. Dans tous les cas on a choisit une union de deux dictionnaires de Gabor ajustés, avec un chevauchement de 50%. Aucun bruit n'a été ajouté, on choisit donc des hyper-paramètres qui tendent vers 0 au cours des itérations, avec une stratégie de redémarrage à chaud [26].

Les performances des séparations sont évalués par les rapport de signal à distortion (SDR) et de signa à interférences (SIR) [27]. Le SDR indique la qualité générale de reconstruction, tandis que le SIR révèle la quantité de résidus dû aux autres sources pour une source donnée. Ces mesures sont moyennées sur toutes les sources, pour chaque condition expérimentales. Le lasso largebande est utilisé comme résultat de référence.

Les résultats sont résumés dans le tableau 1. On remarque que l'utilisation de deux Gabor permet d'améliorer les résultats de séparation aussi bien pour le Lasso que pour le WG-Lasso; le WG-Lasso obtenant de meilleure performance que le Lasso. Des résultats plus détaillés sont disponibles dans [28].

TABLEAU 1 – SDR/SIR : Two Gabors vs One Gabor

	Lasso +1Gabor	Lasso +2Gabors	WG-Lasso +1Gabor	WG-Lasso +2Gabors
Paroles	7.9/14.3	8.0/13.9	8.6/14.6	<b>9.1/14.5</b>
Musique	6.0/9.9	6.2/9.3	6.7/10.0	<b>7.2/10.0</b>

### 4.2 Désêcretage

Pour les expériences, on utilise les données disponibles sur la page <http://small-project.eu/> et la toolbox correspondante pour l'audio inpainting. Plus particulièrement,

les algorithmes sont comparés sur la base de donnée des signaux de musique échantillonnés à 16 kHz, après une normalisation des signaux afin que la valeur maximale des échantillons n'excède pas 1. Les expériences ont été menées avec deux valeurs d'écrêtage : 0.2 et 0.6.

Pour l'algorithme de seuillage itératif 3, on a choisi un dictionnaire de Gabor avec une fenêtre de longueur 1024 échantillons (soit une durée de 32 ms), et un recouvrement de 75% entre deux fenêtres. Les expériences ne font pas de débruitage en plus de la restauration des échantillons écrêtés, ainsi, la valeur du paramètre  $\lambda$  est choisie proche de 0. On utilise ici la stratégie classique de redémarrage à chaud avec des valeurs de  $\lambda$  décroissantes [26]. Le voisinage des opérateurs WGL et PEW est choisi de manière à prendre en compte les quatre coefficients précédents et les quatre coefficients suivant le coefficient considéré sur l'axe temporel.

On résume sur la figure 2 les performances obtenues par différents opérateurs de seuillage, ainsi que le résultat de référence donné dans [11] avec le matching pursuit orthogonal contraint (et un dictionnaire de Gabor).

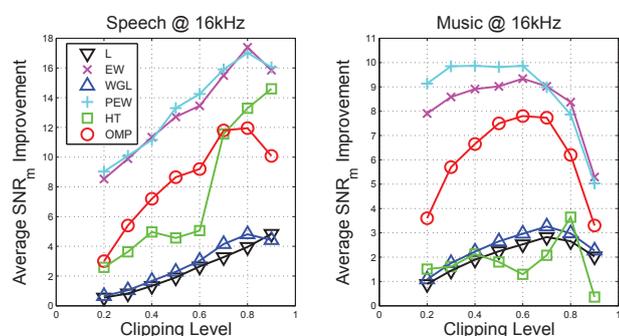


FIGURE 2 –  $SNR_{miss}$  moyen pour 10 signaux de paroles (gauche) et de musique (droite) pour divers niveaux d'écrêtage et différents opérateurs de seuillage. Les voisinages sont de 3 coefficients en temps pour la parole et 7 pour la musique.

On remarque que le Lasso et le group-Lasso fenêtré obtiennent de piètre performance pour ce problème. Vu les très bons résultats obtenus par les opérateurs EW et PEW, construit pour préserver un maximum d'énergie, cela s'explique directement par le phénomène de seuillage, et confirme les observations faites sur d'autres problèmes inverses [23]. Des résultats plus détaillés sont disponibles dans [29].

## Références

- [1] P. Balazs, M. Doerfler, M. Kowalski, and B. Torrèsani, "Adapted and adaptive linear time-frequency representations : a synthesis point of view," *Signal Processing Magazine, IEEE*, vol. 30, no. 6, pp. 20–31, 2013.
- [2] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society Serie B*, vol. 58, no. 1, pp. 267–288, 1996.
- [3] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.
- [4] Ö. Yılmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Transactions on Signal Processing*, vol. 52, no. 7, pp. 1830–1847, Jul. 2004.
- [5] H. Sawada, S. Araki, R. Mukai, and S. Makino, "Grouping separated frequency components by estimating propagation model parameters in frequency-domain blind source separation," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1592–1604, July 2007.
- [6] R. Saab, Ö. Yılmaz, M. McKeown, and R. Abugharbieh, "Blind separation of anechoic under-determined speech mixtures using multiple sensors," in *Proc. IEEE Int. Symposium on Signal Processing and Information Technology (ISSPIT)*, 2006, pp. 642–646.
- [7] S. Winter, W. Kellermann, H. Sawada, and S. Makino, "MAP-based underdetermined blind source separation of convolutive mixtures by hierarchical clustering and  $\ell_1$ -norm minimization," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, 2007, article ID 24717.
- [8] E. Vincent, S. Araki, and P. Bofill, "The 2008 Signal Separation Evaluation Campaign : A community-based approach to large-scale evaluation," in *Proc. 8th Int. Conf. on Independent Component Analysis and Signal Separation (ICA)*, 2009, pp. 734–741.
- [9] A. Janssen, R. Veldhuis, and L. Vries, "Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 34, no. 2, pp. 317–330, 1986.
- [10] S. J. Godsill and P. J. Rayner, "A Bayesian approach to the restoration of degraded audio signals," *Speech and Audio Processing, IEEE Transactions on*, vol. 3, no. 4, pp. 267–278, 1995.
- [11] A. Adler, V. Emiya, M. Jafari, M. Elad, R. Gribonval, and M. D. Plumbley, "Audio inpainting," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 3, pp. 922–932, 2012.
- [12] M. Elad, J.-L. Starck, D. L. Donoho, and P. Querre, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)," *Journal on Applied and Computational Harmonic Analysis*, vol. 19, pp. 340–358, November 2005.
- [13] M. Kowalski, M. Szafranski, and L. Ralaivola, "Multiple indefinite kernel learning with mixed norm regularization," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 545–552.
- [14] P. Combettes and V. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling and Simulation*, vol. 4, no. 4, pp. 1168–1200, Nov. 2005.
- [15] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems,"

- SIAM, Journal of Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.
- [16] J.-J. Moreau, “Proximité et dualité dans un espace hilbertien,” *Bull. Soc. Math. France*, vol. 93, pp. 273–299, 1965.
- [17] M. Kowalski, “Sparse regression using mixed norms,” *Applied and Computational Harmonic Analysis*, vol. 37, no. 3, pp. 303–324, 2009.
- [18] P. Tseng, “Approximation accuracy, gradient methods, and error bound for structured convex optimization,” *Mathematical Programming*, vol. 125, pp. 263–295, 2010, 10.1007/s10107-010-0394-2.
- [19] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.
- [20] I. Daubechies, M. Defrise, and C. D. Mol, “An iterative thresholding algorithm for linear inverse problems with a sparsity constraint,” *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413 – 1457, August 2004.
- [21] A. Antoniadis, “Wavelet methods in statistics : Some recent developments and their applications,” *Statistics Surveys*, vol. 1, pp. 16–55, 2007.
- [22] M. Kowalski, “Thresholding rules and iterative shrinkage/thresholding algorithm : a convergence study,” 2014.
- [23] M. Figueiredo, R. Nowak, and S. Wright, “Gradient projection for sparse reconstruction : application to compressed sensing and other inverse problems,” *IEEE Journal on Selected Topics in Signal Processing*, vol. 1, pp. 586–598, 2007.
- [24] M. Kowalski, K. Siedenburg, and M. Dörfler, “Social sparsity! Neighborhood systems enrich structured shrinkage operators,” *IEEE transactions on signal processing*, vol. 61, no. 10, pp. 2498–2511, 2013.
- [25] “Roomsim,” <http://media.paisley.ac.uk/~campbell/Roomsim/>.
- [26] A. Hale, W. Yin, and Y. Zhang, “Fixed-point continuation for  $\ell_1$ -minimization : Methodology and convergence,” *SIAM Journal on Optimisation*, 19(3), ., vol. 19, no. 3, pp. 1107–1130, 2008.
- [27] E. Vincent, R. Gribonval, and C. Févotte, “Performance measurement in blind audio source separation,” *IEEE Transactions on Speech, Audio and Language Processing*, vol. 14, no. 4, pp. 1462–1469, July 2006.
- [28] F. Feng and M. Kowalski, “Hybrid model and structured sparsity for under-determined convolutive audio source separation,” in *Proc. ICASSP*, May 2014.
- [29] K. Siedenburg, M. Dörfler, and M. Kowalski, “Audio declipping with social sparsity,” in *Proc. ICASSP*, May 2014.