Design and evaluation of sparse models and algorithms for audio inverse problems

Clément Gaultier

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

UNIVERSITÉ DE RENNES 1

THESE DE DOCTORAT DE

L'UNIVERSITE DE RENNES 1
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Signal, Image, Vision*

Par

Clément GAULTIER

Design and evaluation of sparse models and algorithms for audio inverse problems

Thèse présentée et soutenue à Rennes, le 25 janvier 2019
Unité de recherche : Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA, UMR 6074)

Rapporteurs avant soutenance :

Bruno TORRÉSANI Professeur des Universités, Aix-Marseille Université
Matthieu KOWALSKI Maître de Conférences, HDR, Université Paris Sud

Composition du Jury :

Président : Laurent DAUDET Professeur des Universités, Université Paris Diderot

Examinateurs : Bruno TORRÉSANI Professeur des Universités, Aix-Marseille Université
Matthieu KOWALSKI Maître de Conférences, Université Paris Sud
Laurent ALBERA Maître de Conférences, Université de Rennes 1
Laurent DAUDET Professeur des Universités, Université Paris Diderot

Dir. de thèse : Rémi GRIBONVAL Directeur de recherche, Inria
Co-dir. de thèse : Nancy BERTIN Chargée de recherche, CNRS

Invités :

Pavel RAJMIC Associate Professor, Brno University of Technology
Valentin EMIYA Maître de Conférences, Aix-Marseille Université

Acknowledgments

Even though I am the author of this manuscript, I will not take credit for all the results and outcomes of this thesis work. The following words are written as an opportunity to express my gratitude to all the people who contributed to the success of this research work.

Je tiens à remercier en premier lieu mon directeur de thèse Rémi Gribonval ainsi que ma co-encadrante Nancy Bertin de m'avoir si bien guidé pendant ces trois ans. Je suis très honoré qu'ils m'aient fait confiance pour devenir l'un de leurs doctorants. Leurs nombreux conseils et leur patience m'ont permis de réaliser cette thèse dans des conditions exceptionnelles.

J'adresse mes sincères remerciements à Bruno Torrésani et Matthieu Kowalski pour avoir accepté de rapporter ce manuscrit, pour leur expertise et leurs commentaires. Mes remerciements vont également aux membres du jury, Laurent Daudet pour avoir accepté de le présider, Laurent Albera, Valentin Emiya et Pavel Rajmic pour leurs précieuses remarques et discussions bienveillantes.

J'aimerais exprimer ma profonde gratitude envers Srđan Kitić qui m'a épaulé à mon arrivée et transmis de précieux conseils. Sa rigueur et sa disponibilité ont sans aucun doute facilité la découverte de nouveaux outils notamment algorithmiques. J'ai été très heureux de pouvoir collaborer avec toi et te souhaite toute la réussite que tu mérites.

Ensuite, mes remerciements vont à Antoine Deleforge avec qui j'ai pu partager de nombreuses idées et discussions souvent fructueuses. Merci beaucoup pour ton enthousiasme qui aura permis que le projet VAST voie le jour. Merci également pour nos envolées scientifiques teintées de sirop d'érable ou encore de sardines grillées.

Je souhaiterais également remercier Aline Roumy et Alexey Ozerov d'avoir accepté de former mon comité de suivi et pour leurs conseils sur mon travail.

Merci beaucoup aux collègues de l'équipe Panama : Stéphanie, Armelle, Cassio, Corentin, Nicolas, Maxime, Eric, Xavier, Younes, Axel, Mohammed, Adrien, Romain, Roilhi, Kilian, Clément, Gilles, Jérémy, Pierre, Frédéric, Yann, Nicolas, Diego, Corentin, Nathan, Hakim, Antoine, Valentin, Andreas, Martin, Helena, Igal, Saurabh, d'avoir veillé à faire régner la bonne humeur au travail et en dehors. Je m'excuse d'avance si j'ai oublié l'un(e) de vous. J'adresse une mention spéciale à Ewen, Srđan, Tudor et Nicolas avec qui j'ai partagé le bureau et qui ont eu quelques fois à supporter ma mauvaise foi. Merci également aux collègues de l'équipe Tea : Vania, Alexandre, Jean-Joseph, Nam, Simon, Lucas, Liangcong, pour les pauses café débordantes d'anecdotes. Ces trois années m'ont également donné la chance de participer à l'organisation de la Journée Science et Musique. Je tiens

à remercier toutes les personnes avec qui j'ai pu partager cette expérience enrichissante, notamment Agnès et Catherine, les collègues des équipes Hybrid et Panama pour leur bonne humeur et leur soutien. Le modèle de ce manuscrit est très largement basé sur celui d'Olivier. Je le remercie de me l'avoir mis à disposition et laissé m'en inspirer. Merci à mes partenaires de grimpe Antoine, Bryan, Lauréline, Hadrien et quelques collègues de l'équipe Visage, Giulia, Claire. Merci beaucoup à Arnaud d'avoir levé quelques unes de mes ignorances en informatique notamment grâce à son savoir encyclopédique des raccourcis BASH.

Merci à tous les enseignants et collègues que j'ai croisés lors de mon parcours et qui ont su me transmettre brique par brique de nouvelles connaissances ainsi que la volonté d'en comprendre toujours plus. Plus particulièrement, je tiens à remercier Jean-Hugh Thomas, Stephan Bleeck, Jessica Monaghan et Tobias Goehring pour m'avoir donné le goût pour la recherche et incité à poursuivre dans cette voie.

Un grand merci également à « La fine équipe du 7.2 » pour leur soutien sans faille depuis toutes ces années lors de nos retrouvailles musicales et bien plus encore mais aussi aux « Thouns » pour les bons moments passés ensemble notamment lors de nos rencontres annuelles.

Mes remerciements vont également à Ann et Matthew. Si loin, vous avez sû rester disponibles. Je sais pouvoir compter sur vous même à distance. Merci pour vos précieux conseils, notamment sur les règles pour une lettre de motivation en anglais réussie.

Je remercie Lesley, Nico et Alexandre, d'avoir croisé ma route, d'entretenir notre amitié parfois autour d'un billard, d'une roue de secours, d'une perceuse ou simplement d'un vendredi soir festif. Merci de m'avoir transmis votre motivation et persévérance inattaquables même dans les moments où, pour citer un célèbre poète contemporain, j'ai eu l'impression : « d'inventer l'eau tiède en laissant refroidir l'eau chaude ».

Je tiens également à remercier Micheline, Éric, Josette et bien sûr Anne-Flore, ma famille « d'en face ». Merci pour votre compréhension, votre bienveillance et tous ces bons moments : travail de fond qui a sans aucun doute participé à l'aboutissement de ces trois années.

Merci à tous les membres de ma famille qui forment des piliers solides et ont largement contribué à la réussite de ce travail. Merci Laurence, Christophe, Thimotey et Axel pour vos encouragements. Merci à mes parents Sylvie et Pascal, qui m'ont toujours soutenu et permis de garder les pieds sur terre. Merci de m'avoir transmis vos valeurs, merci pour votre aide, votre éducation et votre patience. Merci à mes grands-parents Marc et Janine qui sont très fiers de moi et Yvette et Alexis qui l'auraient été j'en suis sûr.

Enfin et par-dessus tout, merci à toi « Chinwia diali » qui partage ma vie. Merci pour toutes ces choses merveilleuses qu'on peut vivre ensemble depuis quelques années. Merci pour ces découvertes et ton soutien infaillible. Tu as su m'aider à trouver les ressources pour mener à bien ces trois ans. Cette réussite est aussi la tienne, tout simplement, merci d'être à mes côtés.

Rennes, January 25, 2019 Clément Gaultier

Résumé étendu

Ce résumé présente de manière concise en français les différents travaux abordés dans cette thèse. Les détails concernant les outils utilisés, les méthodes proposées et les perspectives sont données dans la suite du manuscrit en anglais.

Introduction Dans le contexte de l'analyse de scènes sonores, les humains mais aussi les machines peuvent rencontrer des situations délicates lorsqu'il s'agit de décoder leur environnement grâce aux observations alentours. Ceci est souvent dû à l'absence d'observations directes des informations d'intérêt. Dans la plupart des cas un lien de cause à effet clairement identifiable entre les informations de départ et les observations manque également. Prenons pour exemple un signal sonore. Qu'il soit utilisé comme moyen de communication, d'alerte ou d'expression artistique, il y a très peu de situations où ce signal est directement observé depuis sa source. En effet, que le capteur soit un microphone ou une oreille humaine, il existe de nombreux processus pouvant altérer ce signal. Que l'altération provienne par exemple de l'environnement de propagation ou de la captation, elle est rarement souhaitée et bien souvent préjudiciable à la bonne transmission d'informations. Le problème qui traite d'estimer le signal initial depuis son observation dénaturée est communément appelé *problème inverse*. Malheureusement, dans de nombreux cas, le problème est dit *mal posé*. Trop peu d'informations sont présentes et il est nécessaire de s'appuyer sur des *aprioris* afin d'approcher une solution au problème (avec des modèles de signaux appropriés par exemple). Les modèles très utilisés pour les signaux sonores sont les modèles parcimonieux. La parcimonie ici suppose que les signaux (aussi nombreux soient-ils) peuvent être décrits avec une combinaison de seulement quelques éléments d'une collection (atomes) : c'est le modèle de parcimonie à la synthèse. Une autre possibilité considère qu'on peut former une représentation *simple* d'un signal en lui appliquant une transformation appropriée : c'est le modèle de parcimonie à l'analyse (ou coparcimonie). Ainsi, les travaux présentés dans cette thèse portent sur la résolution de problèmes inverses en acoustique et en traitement du signal audio dans un cadre mono ou multi-canal. L'accent est mis sur la conception et la validation d'algorithmes de restauration de signaux sonores exploitant en particulier diverses formes de parcimonie (à l'analyse ou à la synthèse, simple ou structurée avec des aprioris de type « parcimonie sociale »).

Première partie La première partie du manuscrit présente le contexte général de modélisation et les concepts algorithmiques ayant attrait aux travaux présentés dans ce document.
Le chapitre 2 (page 9) s'articule autour des différents outils et modèles de signaux qui sont utilisés dans la suite du document. D'une part, on décrit les modèles de représentations parcimonieuses (analyse et synthèse). Ensuite, on présente leurs liens avec les représentations fréquentielles et transformées temps-fréquence redondantes. D'autre part, ce chapitre met l'accent sur des modèles de signaux utilisant différents types de (co)parcimonie structurée.

Au chapitre 3 (page 23), on présente un cadre algorithmique générique pour traiter des problèmes de reconstruction audio. Cet algorithme est le socle commun qui est instancié dans les chapitres suivant pour les différentes applications. Il s'agit d'une procédure itérative s'inspirant de l'algorithme des directions alternées (ADMM). Cette méthode permet ainsi d'estimer un signal en le projetant alternativement sur une contrainte de modèle et sur une contrainte d'attache aux données. Les contraintes de modèles étant directement liées aux modèles de signaux présentés au chapitre 2 tandis que les contraintes d'attache aux données dépendent du problème de reconstruction considéré. Ce chapitre détaille les différents outils nécessaires à l'estimation sous contraintes en exprimant notamment plusieurs opérateurs de seuillage favorisant la parcimonie, la parcimonie sociale et la parcimonie groupée.

Une part importante de ce travail est liée à l'évaluation des algorithmes de reconstruction. C'est pourquoi le chapitre 4 (page 31) introduit les données de test et les différentes mesures de performance qui sont utiles pour les validations expérimentales dans les chapitres suivants. Premièrement, le chapitre décrit les bases de données d'enregistrements sonores utilisées. On note par exemple un grand jeu de données rassemblant exclusivement de la musique (RWC [Goto *et al.* 2002]), un autre consacré à la parole TIMIT ([Garofolo *et al.* 1993]). Un ensemble plus réduit issu du logiciel SMALLbox (SMALL [Damnjanovic *et al.* 2010]) est aussi utilisé pour les comparaisons de moindre envergure. Par la suite, le chapitre précise plusieurs mesures de performance permettant de juger la qualité des méthodes de reconstruction. On décrit des mesures classiquement utilisées telles que le Rapport Signal à Bruit ou le Rapport Signal à Distorsion. (Finalement), ce chapitre présente également un bref historique des mesures objectives permettant de juger de la qualité audio, de l'intelligibilité de signaux de parole, avant de sélectionner les mesures utilisées plus tard dans le document.

Deuxième partie Dans la deuxième partie nous nous intéressons à la résolution de problèmes inverses pouvant provenir de distorsions relevées au niveau des capteurs. Nous détaillons ainsi deux cas d'usage du cadre algorithmique présenté au chapitre 3 page 23.
Dans un premier temps, au chapitre 5 (page 43) nous abordons le problème de reconstruction de signaux sonores bruités. Ce chapitre présente le problème de bruit additif sur des enregistrements sonores et quelques méthodes existantes qui traitent le sujet

de la réduction de bruit en audio. Ensuite, ce chapitre introduit plusieurs méthodes de dé-bruitage développées sur la base du cadre algorithmique commun évoqué plus haut. Notamment, nous retenons une méthode utilisant un modèle (co)parcimonieux social adaptatif pour les représentations temps-fréquence des signaux sonores. Nous intégrons également une méthode utilisant une modélisation (co)parcimonieuse plus traditionnelle avant de comparer leurs performances sur la tâche de dé-bruitage audio. Les résultats des comparaisons sont obtenus à la fois sur un grand jeu de données (la base RWC) et sur des exemples plus réduits (SMALL). Ils indiquent que chaque méthode produit au moins d'aussi bons résultats de reconstruction voire meilleurs pour les conditions les moins dégradées qu'avec la méthode de référence « Block Thresholding » [Yu *et al.* 2008]. On note également qu'avec des critères objectifs de mesure de qualité sonore, il est préférable d'utiliser la méthode incluant un modèle coparcimonieux simple pour dé-bruiter de la musique. En revanche, l'étude sur la qualité montre que pour traiter des signaux de parole bruités, les modèles mettant en œuvre la (co)parcimonie sociale adaptative semblent plus appropriés. On remarque enfin que les différentes méthodes de dé-bruitage dérivées du cadre algorithmique peuvent être mises en œuvre très efficacement pourvu que les dictionnaires et opérateurs d'analyse satisfassent quelques propriétés simples. Ainsi, l'utilisation de transformées fréquentielles rapides et d'outils d'algèbre tels que le repère ajusté au sens de Parseval (*Parseval tight-frame*), permet une efficacité de calcul pouvant aller plus vite que le temps réel sans pour autant sacrifier la performance de reconstruction.

Dans un second temps, le chapitre 6 (page 59) aborde le problème de reconstruction de signaux sonores saturés en amplitude. D'abord, ce chapitre présente le problème de saturation et ses conséquences sur le contenu fréquentiel d'un enregistrement sonore. Nous décrivons également quelques méthodes de l'état de l'art permettant de traiter le problème de dé-saturation en audio. Plusieurs méthodes de dé-saturation sont ensuite proposées et permettent de traiter des cas à un seul canal ou plusieurs canaux. Pour le cas mono-canal, les méthodes incluent des modèles temps-fréquence de signaux (co)parcimonieux simples ou sociaux adaptatifs. Après une discussion sur les moyens de quantifier la saturation, de nombreuses expériences comparent l'efficacité des méthodes. Pour le cas mono-canal, la comparaison menée sur un grand jeu de données montre ainsi que la méthode utilisant un modèle coparcimonieux simple est à privilégier en cas de forte dégradation. Pour les cas de saturation plus modérée, le modèle de signal basé sur la parcimonie simple ou sociale donne de meilleurs résultats. Ces tendances se vérifient à la fois sur des exemples sonores de parole et de musique. On note que pour une comparaison sur un jeu de données plus réduit, les résultats de reconstruction sont au moins aussi bons que ceux de méthodes issues de l'état de l'art. Comme pour le dé-bruitage, on remarque qu'en utilisant des transformées rapides, les algorithmes restent efficients autorisant même un calcul temps-réel pour certaines paramétrisations. Ces travaux sur la dé-saturation audio font l'objet, avec la collaboration des ingénieurs de recherche de l'équipe PANAMA, d'un transfert technologique. Pour le cas multi-canal, les méthodes promeuvent un modèle fréquentiel de signal (co)parcimonieux structuré au travers des canaux. La validation expérimentale montre que l'ajout de dépendance inter-canal per-

met une meilleure reconstruction dans le cas d'enregistrements à plusieurs canaux (stéréo et jusqu'à huit canaux). Les méthodes utilisant un modèle de coparcimonie structurée semblent ainsi être plus avantageuses en cas de forte dégradation. En revanche, pour des situations ou la saturation est plus modérée, les deux versions (analyse et synthèse) du modèle de parcimonie structurée donnent des résultats similaires. On note enfin que ces méthodes améliorent substantiellement les résultats face à une technique employant la coparcimonie simple. Ces performances sont mêmes obtenues dans certaines situations avec un coût de calcul plus réduit.

Troisième partie Dans la troisième partie, notre intérêt se porte sur la résolution de problèmes inverses dont l'origine est liée à la propagation du son entre une source et un capteur dans un environnement clos (une pièce par exemple).
Le chapitre 7 (page 95) montre la versatilité du cadre algorithmique commun en y incluant la possibilité de traiter la dé-réverbération à la lumière d'un problème inverse. Dans un premier temps, le problème de la réverbération est présenté avec ces effets sur le contenu d'un enregistrement sonore. Une méthode dérivée du cadre algorithmique commun et utilisant une représentation basée sur la parcimonie simple ou sociale est ensuite présentée. On compare l'efficacité de ces méthodes sur des signaux de parole pour une tâche de dé-réverbération informée. Les résultats montrent la supériorité du modèle parcimonieux social pour les performances de reconstruction de qualité objective. Ce chapitre sert plutôt de preuve de concept pour l'intégration de la dé-réverbération comme problème de reconstruction audio pouvant être traité de manière similaire au dé-bruitage ou à la dé-saturation. Des travaux complémentaires seraient nécessaires notamment pour adresser le problème de dé-réverbération aveugle.

Au chapitre 8 (page 103) nous étudions le problème de localisation binaurale de sources sonores. Plus indépendamment du reste de ce manuscrit nous abordons ce problème sous l'angle de l'apprentissage statistique. Dans un premier temps nous présentons le problème de localisation binaurale de source sonore. Nous détaillons par la suite quelques méthodes existantes permettant de le traiter grâce à des approches basées sur l'estimation de différence de temps d'arrivée. Dans un second temps, le chapitre propose une nouvelle technique de localisation de sources sonores utilisant un outil d'apprentissage statistique (basé sur un modèle probabiliste gaussien permettant la projection localement linéaire d'un espace de haute dimension vers un espace de basse dimension). Cette technique permet la prédiction de directions d'arrivées et de distance de sources sonores grâce à l'apprentissage d'un modèle sur des réponses impulsionnelles de salles simulées (virtuelles). Nous appelons ce principe : *apprentissage d'espaces acoustiques virtuels*. Cette méthode de localisation est validée expérimentalement sur données réelles et simulées. On montre ainsi qu'une telle approche produit des résultats à la fois plus justes et plus précis pour l'estimation d'azimut comparée à une méthode plus traditionnelle utilisant la corrélation croisée généralisée (GCC-PHAT).

Conclusion Ce dernier chapitre (chapitre 9) récapitule les contributions décrites dans ce manuscrit et présente quelques perspectives de recherche futures liées aux travaux abordés dans cette thèse. Parmi elles on note la possibilité de traiter conjointement ou séparément d'autres tâches de reconstruction comme le masquage de perte de paquets de données ou la dé-quantification. Un angle intéressant dans l'optique de traiter des problèmes dans le cadre multi-canal peut être la modélisation parcimonieuse multidimensionnelle (temps-fréquence-canal). Pour les aspects liés à l'apprentissage virtuellement supervisé d'espaces acoustiques, nous évoquons la généralisation à d'autres antennes de capteurs et l'extension à l'estimation d'autres paramètres de la scène sonore.

List of Figures

List of Tables

List of Abbreviations

A-SPADE Analysis SParse Audio DEclipper

ADMM Alternating Direction Method of Multipliers

AI Articulation Index

ASA Auditory Scene Analysis

BPDN Basis Pursuit Denoising

C-IHT Consistent Iterative Hard Thresholding

CD Compact Disc

CoSaMP Compressive Sampling Matching Pursuit

CSII Coherence Speech Intelligibility Index

DCT Discrete Cosine Transform

DFT Discrete Fourier Transform

DOA Direction Of Arrival

GAP Greedy Analysis Pursuit

GCC Generalized Cross Correlation

GCC-PHAT Generalized Cross Correlation PHAse Transform

GEW Group Empirical Wiener

GLLiM Gaussian Locally-Linear Mapping

HRTF Head-Related-Transfer Function

HT Hard-Thresholding

IHT Iterative Hard Thresholding

ILD Interaural Level Difference

IPD Interaural Phase Difference

ITD Interaural Time Difference

ITU International Telecommunication Union

LASSO Least Absolute Shrinkage and Selection Operator

LPC Linear Predictive Coding

MOS Mean Opinion Score

NCC Normalized Cross Correlation

OMP Orthogonal Matching Pursuit

PEAQ Perceptual Evaluation of Audio Quality

PESQ Perceptual Evaluation of Speech Quality

PEW Persistent Empirical Wiener

PHAT PHAse Transform

PPAM Probabilistic Piecewise Affine Mapping

Quad-GEW Quadratic Group Empirical Wiener

RIR Room Impulse Response

RWC Real World Computing

SCOT Smooth COherent Transform

SDR Signal-to-Distortion Ratio

SNR Signal-to-Noise Ratio

SOMP Simultaneous Orthogonal Matching Pursuit

STFT Short-Time Fourier Transform

STOI Short Time Objective Intelligibility

TDOA Time-Difference Of Arrival

TIMIT Texas Instrument Massachussets Institute of Technology

Table of Contents

List of Symbols

Below are presented the notations rules used throughout this manuscript. Any other symbol inconsistent with these notations will be disambiguated in the text.

x	Real or Complex variable
x	Integer
$\mathbf{x}$	Vector
$\mathbf{X}$	Matrix
$\mathbf{X}^{\mathsf{H}}$	Hermitian transpose of the matrix $\mathbf{X}$
Θ	Set
χ_{Θ}	Characteristic function of a set Θ
$\mathbf{X}_{\Theta}$	Restriction of $\mathbf{X}$ to indexes in Θ
$\mathbb{R}^{\mathsf{n}}, \mathbb{R}^{\mathsf{n}\times\mathsf{m}}$	n or $\mathsf{n}\times\mathsf{m}$ dimensional real vector space
$\mathbb{C}^{\mathsf{n}}, \mathbb{C}^{\mathsf{n}\times\mathsf{m}}$	n or $\mathsf{n}\times\mathsf{m}$ dimensional complex vector space
$\Re$	Real part of a complex number
$\Im$	Imaginary part of a complex number
F, f	Functionals
$\mathrm{vec}(\mathbf{X})$	Vectorized version of the matrix $\mathbf{X}$
$\mathrm{diag}(\mathbf{x})$	Diagonal matrix with the elements of the vector $\mathbf{x}$ on the diagonal
$\mathbf{I}$	Identity matrix
$\mathbf{x}_{\mathsf{i}}$, (alternatively, $\mathbf{x}(\mathsf{i})$)	i^{th} element of the vector $\mathbf{x}$
$\mathbf{X}_{\mathsf{ij}}$	Component of the matrix $\mathbf{X}$ indexed by the i^{th} line and the j^{th} column
$\mathbf{x}^{(\mathsf{i})}, \mathbf{X}^{(\mathsf{i})}$	i^{th} iterate of the vector $\mathbf{x}$ (respectively the matrix $\mathbf{X}$)
$\preccurlyeq, \succcurlyeq, \prec, \succ$	Entry-wise comparisons between matrices or vectors

Chapter 1

Introduction

Human senses and machines are facing every day difficult situations to cope with when trying to decode their environment. This is mostly due to the fact that direct observations of physical phenomena are usually unavailable. However, in some situations we can still benefit from indirect observations. Sometimes, if these are enough for us, machines trying to mimic our behavior are struggling. A visual examples of such a scenario can be the shadow theater. It is widely admitted for an adult that seeing the black shape of a bird or a dog on a white backlit sheet does not necessarily mean that there is actually a dog or a bird passing behind (see: Figure 1.1). In that case, we would rather think that someone is moving his hands behind the scene and tries to make us believe that some animal is actually there. Now give this shadow image to a computer equipped with an image classification system, it will return you "BIRD" or "DOG" and not "MOVING HANDS". In decoding this situation why did not we get duped? (except probably a little child) For ages, we have been used to inferring models from indirect observations that could explain our world. This kind of problem is called *inverse problem*.

Our work here will not deal with shadows or dogs and birds images but rather sound. First of all, the word "sound" can sometimes be related to:

- a cause (*e.g.* a speech sound),
- an acoustic wave which can be characterized by physical quantities (*e.g.* a harmonic sound whose tonal frequency is 440 Hz),
- a perception (*e.g.* a sound which is bright and clear).

Figure 1.1 – Indirect observations: the shadow theater example

In this document, we will rather use the second description. More precisely, we characterize the sound or acoustic wave by an oscillation that propagates in a supporting media (mostly air when dealing with airborne sound propagation). Some sounds may or may not be audible by humans due to the special characteristics of their auditory system. The limits are often given by the frequency range. Frequency defines in that case the air vibration rate when excited by the wave and is represented in Hertz (Hz). This notion of frequency is crucial in modeling sounds and will be used to derive more advanced models in chapter 2. Sounds can be registered by our auditory systems but also sensors (*e.g.* a microphone) able to convert acoustic pressure to voltage. The observation of a sound we often get is only given by this transduction. Hence, in the following we will consider sounds as signals resulting from the conversion of acoustic pressure to voltage by a microphone.

We also note that the shadow theater example above is certainly not the exclusive example of indirect observation. More particularly, considering sound, there are really few situations where the signal of interest, rather used for communication, alert or simply artistic purpose is directly observed.

The way we interact with sounds is on the verge of being completely reinvented. Particularly, some technical changes such as the wide availability of small and cheap acoustic sensors is strongly participating in this revolution. The growing number of available audio data and ways to sense sound easily is triggering new technical stakes for audio and acoustic signal processing. Usually, signal processing methods in audio and acoustic allows machines for instance a computer, a robot, a phone to perform tasks that we (humans) are able to address more or less easily. These machines could also be used to assist and sometimes surpass us in analyzing sound scenes. By analyzing a sound scene we mean here, retrieving some of the sounds, acoustic sources parameters from the indirect observations (which can be microphone recordings). This task is itself an inverse problem and referred to as Auditory Scene Analysis (ASA).

Ill-posed inverse problems The major issue in the context of acoustic sensing or (computational) auditory scene analysis is that we are usually facing severely ill-posed inverse problems to solve. Indeed, whether the sensor is a microphone or a human ear, there are several ways a signal of interest can be altered between its emission and its recorded version. This alteration is rarely desired and often leads to detrimental consequences on transmission of information. A well known example for us is probably listening and understanding speech in noise. This is a task that we handle on a daily basis when trying to interact with people. In a noisy chat situation, we are able with more or less effort to isolate and understand someone speaking to us despite the surrounding hostile listening environment. Our auditory system has the stunning ability to focus on the information of interest. However, what we hear initially is just a mixture of the targeted speaker, background noise, other people speaking, reverberation, etc...

If we replace the listener in the previous situation with a machine dedicated to speech understanding or transmission (a phone, a robot, a hearing aid...), the task of estimating the targeted speech signal from the mixture of sounds is called addressing the denoising or speech enhancement inverse problem. More generally, all problems that need to estimate a signal from some observation are called inverse problems. In most of the cases these are severely ill-posed, meaning that information is missing and there is a need to infer a model to help solving the problem.

Sparsity Modern signal processing translated some physical models of sounds with mathematical concepts such as sparsity. Sparsity assumes here that a signal can be described by a linear combination of a small number of elements from a collection (atoms): this is the sparse synthesis data model. Another alternative hypothesis is the sparse analysis data model which considers that we can form a *simple* representation of a signal by applying an appropriate transformation. Some refinements of these models are particularly useful for sound modeling and will be presented in chapter 2.

Thus, the work presented in this thesis will focus on addressing audio and acoustic inverse problems for single and multichannel cases. We put a particular emphasis on design and validation of audio reconstruction algorithms relying on various form of sparsity.

The scope for this work as presented above is quite broad. Thus, we intend to structure the document and our reflection around the following question:

> What sparse model is best suited for audio reconstruction?

This thesis work was held in the PANAMA project team, a joint research team between Inria and CNRS at IRISA research center (Rennes, France). This work was jointly funded by the European Research Council *PLEASE* project (ERC-StG-2011-277906) and Région Bretagne.

Structure of the document

Part I . The first part is dedicated to introducing the appropriate signal models and algorithmic concepts we will work with in the rest of the document.
While this chapter introduces the manuscript, as for Part I, chapter 2 will express useful tools in signal processing that we will be using to present this work. More particularly it will include the different signal models that are used for solving some audio signal processing inverse problems. Among them, we detail the sparse data models, frequency transforms and refinement of sparsity for modeling audio signals.

In chapter 3, we present a generic algorithmic framework proposed and used to build a versatile method that will be later used to solve various audio reconstruction problems. This framework serves as a common baseline for the methods presented in the chapters addressing signal reconstruction.

The chapter 4 concludes the first part of this manuscript by detailing data and measures of performance that will be used along with the experimental evaluations of the methods presented in the second and third part of the document.

Part II . In Part II, we mainly deal with inverse problems stemming from sensor-based degradation.
Thus, chapter 5 features a particular example of the aforementioned generic framework to tackle single-channel audio denoising. After presenting available existing work, we will describe a new method for audio denoising built on the generic framework and time-frequency analysis/synthesis sparse priors. Finally, an adaptive structured (co)sparse method will be introduced before comparing them on experiments involving real speech and music audio data.

The chapter 6 extends the framework to audio declipping. After a short review on literature, we present an audio declipping method embedding adaptive structured analysis/synthesis time-frequency sparse priors to solve the problem for single-channel saturated signals. Then, we extend the work to deal with multichannel audio recordings thanks to channel-wise structured (co)sparse modeling. Before concluding this chapter, we present some experiments that thoroughly compare performance and parameters of the different proposed methods.

Part III . The third part focuses on audio inverse problems arising from sound propagation within the recording environment.
In chapter 7 we add a dereverberation scenario to the generic framework introduced in chapter 3 before including a comparison of the various signal models on a speech dereverberation application.

More independently from the rest of the manuscript, chapter 8 presents the concept of virtually supervised learning in auditory scene analysis and an application to binaural sound source localization. After reviewing some state of the art techniques, we describe the new idea of virtually supervised learning in audio scene analysis before applying it to binaural sound source localization with massive regression learning technique.

Finally, chapter 9 concludes this thesis and exposes possible future steps and foreseen extensions of this work.

Figure 1.2 displays a structured overview of the thesis and possibly an alternative path for the reader.

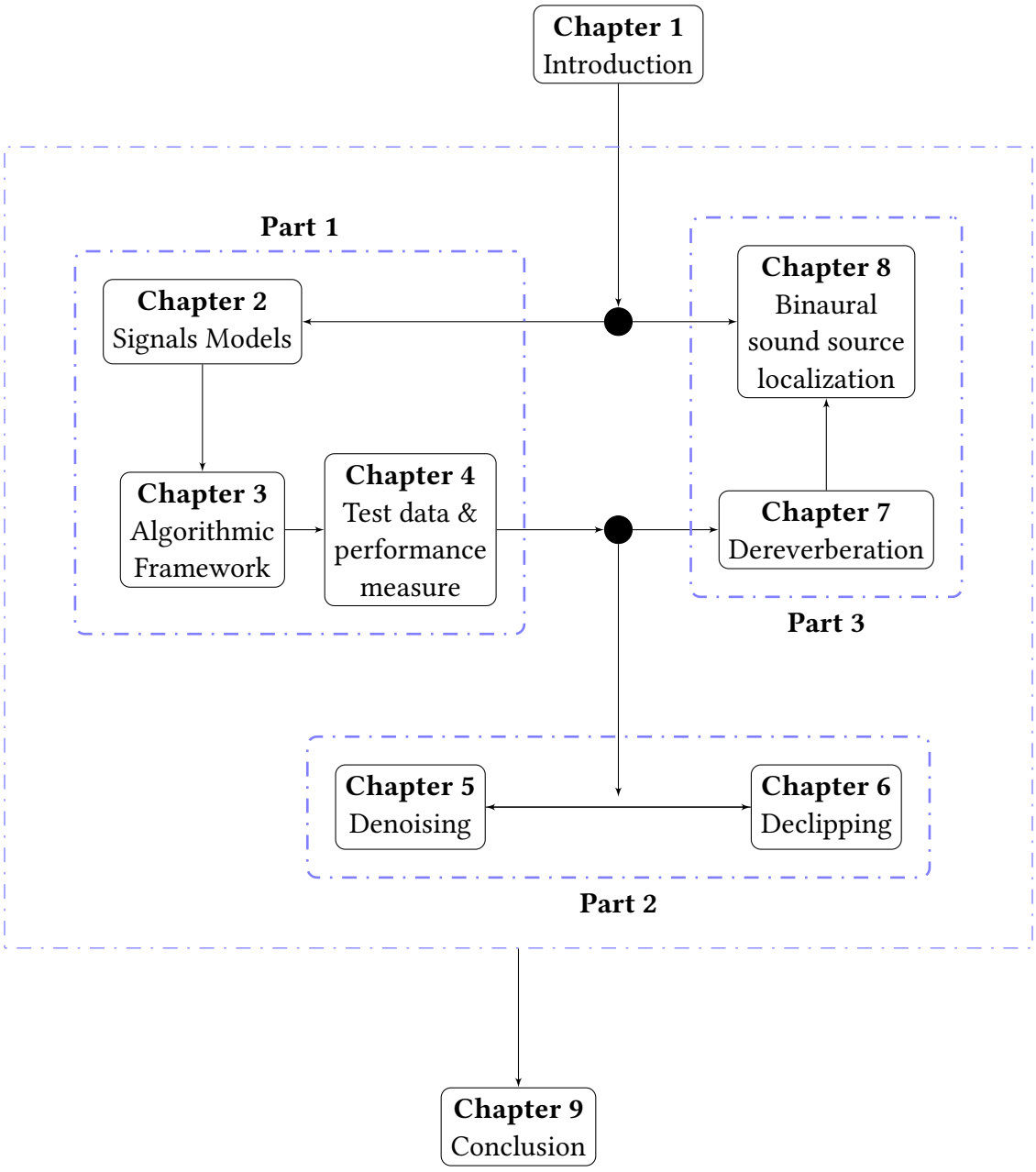


Figure 1.2 – Thesis Outline

This thesis is partly inspired from the publications listed below. When relevant, chapters start with a note on the related publications it may share some line with.

International peer-reviewed conferences:

Clément Gaultier, Nancy Bertin and Rémi Gribonval. *CASCADE: Channel-Aware Structured Cosparse Audio DEclipper*. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 571–575, April 2018

Clément Gaultier, Srđan Kitić, Nancy Bertin and Rémi Gribonval. *AUDASCITY: AUdio Denoising by Adaptive Social CosparsITY*. In 2017 25th European Signal Processing Conference (EUSIPCO), pages 1265–1269, Aug 2017

Clément Gaultier, Saurabh Kataria and Antoine Deleforge. *VAST: The Virtual Acoustic Space Traveler dataset*. In International Conference on Latent Variable Analysis and Signal Separation, pages 68–79. Springer, 2017

Saurabh Kataria, Clément Gaultier and Antoine Deleforge. *Hearing in a shoe-box: Binaural source position and wall absorption estimation using virtually supervised learning*. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 226–230, March 2017

Workshops:

Romain Lebarbenchon, Ewen Camberlein, Diego Di Carlo, Clément Gaultier, Antoine Deleforge and Nancy Bertin. *Evaluation of an open-source implementation of the SRP-PHAT algorithm within the 2018 LOCATA challenge*. In 2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC), LOCATA Challenge. IEEE, 2018

Clément Gaultier, Srđan Kitić, Nancy Bertin and Rémi Gribonval. *Cosparse Denoising: The Importance of Being Social*. In The Signal Processing with Adaptive Sparse Structured Representations (SPARS) workshop, 2017

Report:

Clément Gaultier, Nancy Bertin, Srđan Kitić and Rémi Gribonval. *A modeling and algorithmic framework for (non) social (co) sparse audio restoration*. arXiv preprint arXiv:1711.11259, 2017

Part I

Context

Chapter 2

Signal Models

Contents

The following chapter describes signal modeling concepts which are later used in the manuscript. After introducing useful tools about sparsity and regularization, we detail frequency transforms. Then, we present how sparsity can be used in audio signal modeling scenarii.

2.1 Sparsity

Simple models in the spirit of the Occam's razor principle, have been widely used in physics or chemistry for a long time. The underlying idea behind such models is that any attempt to explain any reasoning, to model or to verify a hypothesis with additional elements should be as far as possible avoided. This is not to be confused with the idea assuming that the simpler the hypothesis the better. It is only in the 1990s that it was formalized to serve as a first corner stone to redefine modern signal processing. For the latter, natural signals supposedly admit a simple – *sparse* – representation which means that most of its elements are zeros (or close to zeros). Precisely, the wavelet basis for images or the Fourier basis for sounds offer a good sparse decomposition. These decompositions are the starting point of JPEG or MPEG compression standards. They benefit from the intrinsic redundancy of the complex initial signal representation to express it in a sparse manner in a transform domain. This way, only the most important coefficients (largest) are stored. Additionally, other applicative fields verify this sparse assumption such as magnetic resonance imaging (MRI) [Lustig *et al.* 2007], single pixel imaging [Duarte *et al.* 2008] and many others. Such sparse decomposition $\mathbf{z}$ of a signal $\mathbf{x}$ are oftenly expressed as $\mathbf{x} = \mathbf{D}\mathbf{z}$ also known as the sparse synthesis model. More details will be given on this model in subsection 2.1.3 and subsection 2.1.5. Even if some novel work about continuous dictionaries learning or *off-the-grid* sparse recovery [Poon *et al.* 2018] seems really interesting for the generalization of sparse models to continuous domains, in the following, we will stick to discrete settings considering sampled signals on a finite domain.

2.1.1 Matrix and vector norms

In the following subsection we list some useful norms needed to describe the data models thereafter. In this work norms can be seen as a mean to quantify several properties of a tested vector (signal).

Particularly, to quantify the sparsity of any discrete signal stored in a vector $\mathbf{x}$, using the ℓ_0 pseudo-norm $\|\cdot\|_0$ which counts the non-zeros elements in $\mathbf{x}$, is a solution. Precisely, this pseudo-norm is defined as follows for any vector $\mathbf{x}$ of size L:

$$\|\mathbf{x}\|_0 = \sum_{n=1}^{\mathsf{L}} |\mathbf{x}_n|^0. \tag{2.1}$$

Even though the exact definition can vary, this special case is usually referred to as a *pseudo-norm* since it does not share the homogeneity property of a regular norm. Indeed,

for any $\mathbf{x}$ and any $\lambda \in \mathbb{R}$ such that $\lambda \neq 0$, $\lambda \cdot \|\mathbf{x}\|_0 \neq \|\lambda \mathbf{x}\|_0$. The same way, we define the ℓ_1 norm $\|\cdot\|_1$, which is the sum of all absolute values of a vector by:

$$\|\mathbf{x}\|_1 = \sum_{n=1}^{L} |\mathbf{x}_n|. \tag{2.2}$$

More generally, the ℓ_p norm of a vector $\mathbf{x}$ is expressed by:

$$\|\mathbf{x}\|_p = \begin{cases} \left(\sum_{n=1}^{L} |\mathbf{x}_n|^p\right)^{\frac{1}{p}} & \text{for} \quad 0 \leq p \leq \infty, \\ \max_n |\mathbf{x}_n| & \text{for} \quad p = \infty. \end{cases} \tag{2.3}$$

We remark that another particular case of Equation (2.3), is the ℓ_2 norm also called Euclidean norm which is used to quantify the energy. Moreover, one has to note that in this definition, the smaller p is, the more significant smaller values of $\mathbf{x}$ will be in the computation of the norm. On the contrary, the importance of larger values will be emphasized as p gets larger.

Similarly, norm measures are defined for matrices $\mathbf{X}$ of size $M \times N$. For instance, we characterize the Frobenius norm as follows:

$$\|\mathbf{X}\|_F = \sqrt{\sum_{m=1}^{M} \sum_{n=1}^{N} |\mathbf{X}_{mn}|^2}. \tag{2.4}$$

This Frobenius norm can be seen as an extension of the Euclidean norm for matrices. To go further, the ℓ_p norms generalizes for matrices to $\ell_{p,q}$ norms that are defined as:

$$\|\mathbf{X}\|_{p,q} = \left(\sum_{n=1}^{N} \left(\sum_{m=1}^{M} |\mathbf{X}_{mn}|^p \right)^{\frac{q}{p}} \right)^{\frac{1}{q}} \quad \text{for } p, q \geq 1. \tag{2.5}$$

These $\ell_{p,q}$ norms, also called *mixed* norms when $p \neq q$, will be also evoked to detail group sparse structures in section 2.3.

2.1.2 Linear inverse problems

The interest of this work will be to approach a solution of inverse problems with the help of audio or acoustic signal processing tools. Before detailing these tools in the following sections, we more formally define what is a linear inverse problem.

Denote $\mathbf{y}$ some observations and $\mathbf{x}$ some data or signal of interest. We can define the direct problem that maps the data to the observations by:

$$\mathbf{y} = \mathbf{M}\mathbf{x}, \tag{2.6}$$

where $\mathbf{M}$ is often called the measurement matrix/operator and encode the forward relation between $\mathbf{x}$ and $\mathbf{y}$.

Addressing the linear inverse problem means estimating $\mathbf{x}$ from the observations $\mathbf{y}$. Such a problem could be easy to solve if it is well-posed in the sense of Hadamard [Hadamard 1902] which would mean that:

- There exists a solution
- The solution is unique
- The solution continuously depends on the data

Unfortunately, there are really few situations where the problem described in Equation (2.6) is well-posed. If a solution exist it is likely not to be unique. Thus without any prior on $\mathbf{x}$ the task turns out to be out of reach.

The two next subsections will present two data models that can help retrieving $\mathbf{x}$ in the audio signal processing context.

2.1.3 The synthesis sparse model

The *sparse synthesis model* assumes that the signal of interest $\mathbf{x}$ is built from a linear combination of atoms aggregated in a large dictionary $\mathbf{D}$. We could more precisely write

$$\mathbf{x} = \mathbf{D}\mathbf{z} \tag{2.7}$$

with $\mathbf{x} \in \mathbb{R}^L$ the time domain signal, $\mathbf{D} \in \mathbb{C}^{L \times S}$ the dictionary and $\mathbf{z} \in \mathbb{C}^S$ a sparse representation of the vector $\mathbf{x}$ ($S \geq L$). This models considers that the number of non-zero coefficients in $\mathbf{z}$ is very small compared to the size S of the vector. In other words, one needs very few atoms of $\mathbf{D}$ to synthesize $\mathbf{x}$ from $\mathbf{z}$. Figure 2.1 graphically represents the sparse synthesis model.

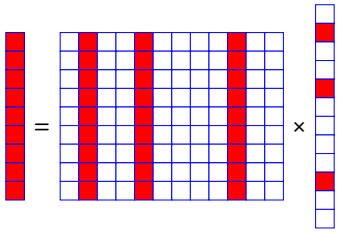


Figure 2.1 – Sparse representation

2.1.4 The analysis sparse model

While synthesis approaches comprise a vast majority of the sparsity-based time-frequency regularization techniques, it has been demonstrated in [Nam *et al.* 2011] and more recently in [Kitić 2015, Kitić *et al.* 2015] that the *analysis sparse model*, also known as the

cosparse model, can turn out to be more advantageous, in particular in terms of computational cost. Instead of *implicitly* defining $\mathbf{x} = \mathbf{D}\mathbf{z}$ a sparse representation $\mathbf{z}$ of the signal $\mathbf{x}$ through the sparse synthesis model, the rationale of the cosparse model is to *explicitly* assume that

$$\mathbf{z} = \mathbf{A}\mathbf{x} \tag{2.8}$$

is sparse with $\mathbf{A} \in \mathbb{C}^{\mathsf{P}\times\mathsf{L}}$ called the analysis operator ($\mathsf{P} \geq \mathsf{L}$). The two models are equivalent when $\mathsf{P} = \mathsf{S} = \mathsf{L}$ and $\mathbf{A}\mathbf{D} = \mathbf{I}$. Figure 2.2 below describes Equation (2.8) of the cosparse model.

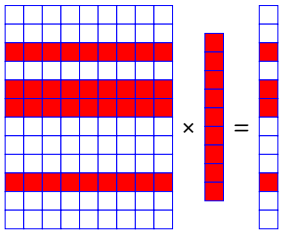


Figure 2.2 – Cosparse representation

2.1.5 Regularization

Generally speaking, regularization, when used for solving inverse problems is a useful tool to either prevent overfitting or add additional knowledge on the signal one wants to estimate. For instance, if one needs to estimate $\mathbf{x}$ from some measured data $\mathbf{y}$, it can be achieved by solving the following kind of optimization problem:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\text{minimize}}\ f_r(\mathbf{x}) + f_d(\mathbf{y}, \mathbf{x}). \tag{2.9}$$

Here $f_d(\cdot)$ usually stands for the *data-fidelity* term and quantifies a measure of fit between the measurement $\mathbf{y}$ and the solution $\mathbf{x}$. $f_r(\cdot)$ is called the *regularizer*. This term which embeds additional information on $\mathbf{x}$ is also referred to as a *prior*.

A widely used data-fidelity measure is $f_d(\mathbf{x}, \mathbf{y}) = \|\mathbf{y} - \mathbf{M}\mathbf{x}\|_2^2$. This squared difference, where $\mathbf{M}$ can be identified as the measurement matrix (see: Equation (2.6)), ensures that the estimate is compatible with the observed data. If this quadratic term is used in most of the cases, $f_r(\cdot)$ has to be chosen according to the application and the model of signal at hand.

Given a sparse assumption on any signal $\mathbf{z}$ to recover, the best regularizer would be the ℓ_0 norm presented earlier and the optimization problem becomes:

$$\hat{\mathbf{z}} = \underset{\mathbf{z}}{\text{minimize}}\ \|\mathbf{y} - \mathbf{M}\mathbf{z}\|_2^2 + \|\mathbf{z}\|_0. \tag{2.10}$$

This problem suffers from inherited NP-hardness [Foucart & Rauhut 2013, page 53] and in practice either greedy approaches or convex relaxation are used to approximate a solution $\hat{\mathbf{z}}$. From a synthesis sparse point of view the greedy methods are designed to retrieve iteratively the support of the signal, namely the active atoms in $\mathbf{M}$. This is the case for the well known *Matching Pursuit* algorithm [Mallat & Zhang 1993] and its variant *Orthogonal Matching Pursuit* (OMP) [Pati *et al.* 1993]. Matching Pursuit gradually recovers the support of a sparse signal in a dictionary by comparing at each iteration the correlation between the signal and the atoms. Among greedy methods for sparse reconstruction, we can also cite *Iterative Hard Thresholding* [Blumensath & Davies 2009] (IHT) and *Compressive Sampling Matching Pursuit* [Needell & Tropp 2009] (CoSaMP). For convex relaxation, the idea is to find a penalty to replace the ℓ_0 norm that is able to convexify Equation (2.10) while still promoting the sparsity of $\mathbf{z}$. For that purpose, a common choice is $f_r(\mathbf{z}) = \lambda\|\mathbf{z}\|_1$ leading to the well known *Basis Pursuit Denoising* [Chen *et al.* 2001] (or LASSO [Tibshirani 1996]) problem:

$$\hat{\mathbf{z}} = \underset{\mathbf{z}}{\text{minimize}} \ \|\mathbf{y} - \mathbf{M}\mathbf{z}\|_2^2 + \lambda\|\mathbf{z}\|_1. \tag{2.11}$$

Contrarily to the greedy approaches presented earlier, solving the LASSO problem produces a solution which is the global minimum of Equation (2.11) (due to convexity). This can be achieved through soft-thresholding in iterative shrinkage algorithms. Here the parameter λ indicates how aggressively to perform the regularization. Although such sparse regularization methods were initially designed to account for the synthesis sparse data model, the approach is generalized with its analysis counterpart. Hence, Equation (2.10) becomes for the analysis case:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\text{minimize}} \ \|\mathbf{y} - \mathbf{x}\|_2^2 + \|\mathbf{A}\mathbf{x}\|_0. \tag{2.12}$$

Similarly, greedy approaches with, for instance, *Greedy Analysis Pursuit* (GAP) [Nam *et al.* 2013] or convex relaxation can be used to approximate a solution of this problem.

2.2 Frequency transforms

Generally, if one has to deal with any signal, the first available data would be (in a discrete setting) a sequence of samples representing a physical quantity which can be sensed. For instance, in the case of a recorded sound, a microphone mirrors the acoustic pressure. In the context of neural activity monitoring on humans, it can be electrically evoked potentials. Light intensity for video or acceleration in vibration control are other examples. Dealing with acoustic or audio signals, this first data representation evolving across time provides some useful information. One drawback is that this representation is usually not sparse and lacks of explicit information about the frequency content.

2.2.1 Real and complex transforms

In the audio signal processing context, frequency transforms are a crucial tool as they are usually producing a much sparser representation of the signal than the signal itself.

Joseph Fourier, with his work on the heat equation [Fourier 1822] introduced a first signal representation. His tool named *Fourier decomposition* aims at approximating any signal by a sum of sine and cosine waves. Among all frequency transforms, complex-valued transforms and real-valued transforms differ. For the latter, some temporal information is directly encoded in the magnitude of the time-frequency coefficients. This way, with real-valued transforms, time shifts on the original signal do not produce a consistent frequency representation. A well known real-valued transform widely used in audio application is the Discrete Cosine Transform (DCT). Let $\mathbf{x} \in \mathbb{R}^L$ be a discrete time-domain signal, its discrete cosine transform $\mathbf{z}$ is defined as follows:

$$\mathbf{z}_s = \sqrt{\frac{2}{L}} \sum_{n=0}^{L-1} \mathbf{x}_n \cdot \cos\left(\frac{\pi}{L}\left(s + \frac{1}{2}\right)\left(n + \frac{1}{2}\right)\right). \tag{2.13}$$

Its inverse transform can be written as:

$$\mathbf{x}_n = \sqrt{\frac{2}{L}} \sum_{s=0}^{L-1} \mathbf{z}_s \cdot \cos\left(\frac{\pi}{L}\left(s + \frac{1}{2}\right)\left(n + \frac{1}{2}\right)\right). \tag{2.14}$$

This defines a Type-IV DCT, other variants of this discrete cosine transform notably exist.

Similarly, a common complex-valued transform is the Discrete Fourier Transform (DFT) which is defined as follow for $\mathbf{x} \in \mathbb{R}^L$:

$$\mathbf{z}_s = \frac{1}{\sqrt{L}} \sum_{n=0}^{L-1} \mathbf{x}_n \cdot \exp\left(-\frac{2j\pi ns}{L}\right). \tag{2.15}$$

j denotes here the complex number defined as $j^2 = -1$. Its inverse transform can be written as:

$$\mathbf{x}_n = \frac{1}{\sqrt{L}} \sum_{s=0}^{L-1} \mathbf{z}_s \cdot \exp\left(\frac{2j\pi ns}{L}\right). \tag{2.16}$$

We notice that Equation (2.13) to Equation (2.16) can be rewritten in matrix form. For instance, for the forward DFT transform, we have:

$$\mathbf{z} = \mathbf{A}\mathbf{x}, \text{with } \mathbf{A}_{ns} = \frac{1}{\sqrt{L}} \exp\left(-\frac{2j\pi ns}{L}\right), \tag{2.17}$$

and for the inverse DFT:

$$\mathbf{x} = \mathbf{D}\mathbf{z}, \text{with } \mathbf{D}_{sn} = \frac{1}{\sqrt{L}} \exp\left(\frac{2j\pi sn}{L}\right). \tag{2.18}$$

In Equation (2.17) and Equation (2.16), $\mathbf{x} \in \mathbb{R}^L$ is the time-domain signal, $\mathbf{z} \in \mathbb{C}^L$ its frequency representation and $\mathbf{A} \in \mathbb{C}^{L \times L}$ (respectively $\mathbf{D} \in \mathbb{C}^{L \times L}$) the *direct* (respectively the *inverse*) frequency transform.

Equivalence with *sparse* data models

We notice that these two previous equations can be identified as the sparse synthesis (Equation (2.7)) and sparse analysis (Equation (2.8)) with the forward frequency transform matrix being the analysis operator and the inverse frequency transform being the dictionary.

If these transforms are widely used nowadays, they depict the global frequency content of a signal leaving aside any possible evolution across time. For this reason, later were introduced *time-frequency* transforms to account for both frequency content and its evolution across time. The global approach of such time-frequency transforms is to use a sliding window on a time domain signal to compute its frequency representation for each (short-time) chunk of the windowed signal. This lead to a short-time frequency transform of the underlying signal. More formally, denote $\tilde{\mathbf{x}} \in \mathbb{R}^N$ a long discrete time-domain signal. We consider $\underline{\mathbf{x}} \in \mathbb{R}^L$ a frame of $\tilde{\mathbf{x}}$ (L consecutive samples extracted from $\tilde{\mathbf{x}}$). We window the segment $\underline{\mathbf{x}}$ such that $\mathbf{x} = \mathbf{W}\underline{\mathbf{x}}$ is called a windowed time-frame of $\tilde{\mathbf{x}}$. Here $\mathbf{W} = \text{diag}(\mathbf{w})$ with $\mathbf{w} \in \mathbb{R}^L$ the weighting window. Applying a discrete frequency transform on each consecutive $\mathbf{x}$ gives a time-frequency transform of the underlying $\tilde{\mathbf{x}} \in \mathbb{R}^N$ ($N \gg L$). Namely, applying the DFT on each segment produces the Short-Time Fourier Transform (STFT) of the initial time-domain signal. In the following we will consider all $\mathbf{x} \in \mathbb{R}^L$ to be windowed time-frames.

2.2.2 Redundancy

If the time-frequency transforms described above demonstrated their usefulness for audio applications, they can not be arbitrarily precise in time *and* in frequency. Indeed, the wider the time support is the more accurate in frequency is the representation. Equivalently the shorter the time window the more precise is the transform. Figure 2.4 illustrates this uncertainty principle with two STFT squared modulus representation (spectrogram). On Figure 2.4a we notice smeared transients whereas the frequencies are sharply represented. On the contrary, Figure 2.4b displays well defined attacks but a larger spectral density. When using such representations one has to keep in mind this trade off between time and frequency resolution. To alleviate the effects of this limitation, several solutions were introduced such as multi-resolution time-frequency or time-scale decompositions with the wavelets being the flagship of the field [Mallat 1999]. Another solution with more classical transforms is to increase the frequency resolution through the use of frequency redundant transforms.

In this case, the transforms presented above can no longer be represented as square matrices. Indeed, redundancy extend the number of atoms in the dictionary so that the synthesis sparse data model rewrites:

$$\mathbf{x} = \mathbf{D}\mathbf{z}, \text{ with } \begin{cases} \mathbf{x} & \in & \mathbb{R}^{L}; \\ \mathbf{D} & \in & \mathbb{C}^{L \times S}; \\ \mathbf{z} & \in & \mathbb{C}^{S}; \\ S & > & L. \end{cases} \tag{2.19}$$

Equivalently, the sparse analysis data model slightly changes to:

$$\mathbf{z} = \mathbf{A}\mathbf{x}, \text{ with } \begin{cases} \mathbf{x} & \in & \mathbb{R}^{L}; \\ \mathbf{A} & \in & \mathbb{C}^{P \times L}; \\ \mathbf{z} & \in & \mathbb{C}^{P}; \\ P & > & L. \end{cases} \tag{2.20}$$

Such redundant dictionaries or analysis matrices are often called “overcomplete”. In practice, products with $\mathbf{A}$ are done using the frequency transform of size P on a zero padded time-domain signal $\mathbf{x}$ of initial length L. Similarly, products with $\mathbf{D}$ are done truncating the inverse frequency transform of size S.

Notably, another property of the (time)-frequency transforms described above can be coined by the term *redundancy*. Complex transforms, as they embed two numbers for each frequency coefficient can be seen as intrinsically redundant (real and imaginary part). Indeed, such transforms offers twice as many numbers to describe the frequency content as the available time domain samples. In the following and more precisely for the experimental sections as we use complex-valued transforms, *redundancy* will denote overcomplete transforms.

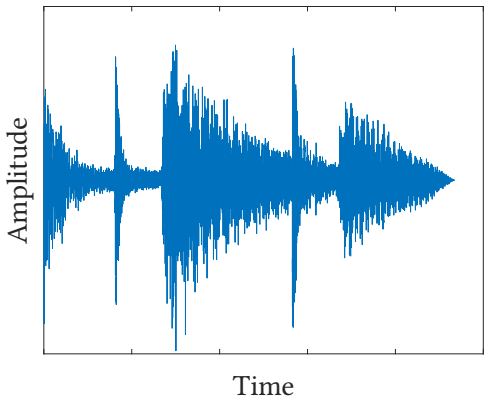
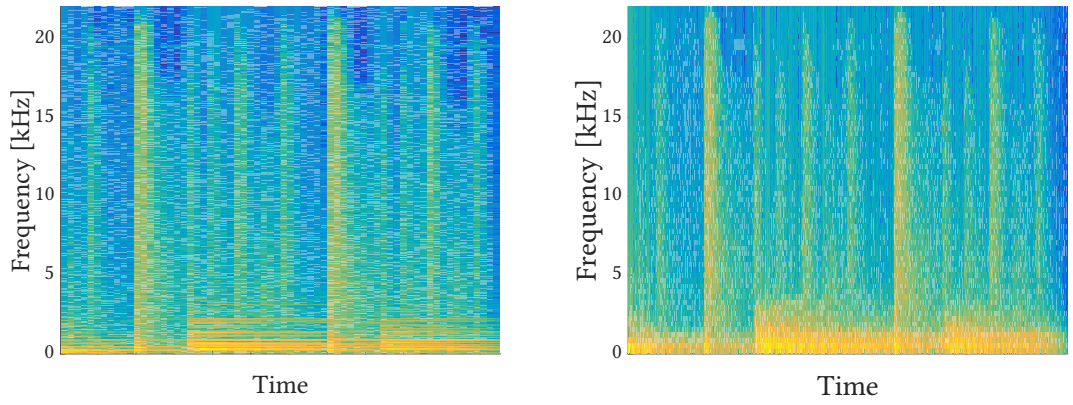


Figure 2.3 – Time domain signal

2.2.3 Frames

Keeping the sparse data models in mind, the overcomplete dictionaries described above become more general than a basis to decompose any signal $\mathbf{x} \in \mathbb{R}^{L}$. However, some properties on the dictionary matrix are still needed when used with convex or greedy approaches for signal reconstruction.



(a) STFT adapted for better frequency resolution (b) STFT adapted for better time resolution

Figure 2.4 – STFT magnitude representation of sound signal

Definition 1 (Frame). *A collection of vectors (atoms) gathered in a matrix* $\mathbf{\Phi}$ *is a frame if there are two positive constants* $0 < A \leq B < \infty$ *such that for any vector* $\mathbf{v}$

$$A\|\mathbf{v}\|_2^2 \leq \|\mathbf{\Phi}^H\mathbf{v}\|_2^2 \leq B\|\mathbf{v}\|_2^2.$$

If $A = B = \alpha$*, the frame is tight and if* $\alpha = 1$, $\mathbf{\Phi}$ *is a Parseval tight frame. Particularly in that case, we have the following property:*

$$\mathbf{\Phi}\mathbf{\Phi}^H = \mathbf{I}.$$

In the remainder of this thesis, all considered dictionary matrices as well as the Hermitian transpose of the analysis operators will be Parseval tight frames. However, some results can still hold for regular tight frames ($\mathbf{\Phi}\mathbf{\Phi}^H = \alpha\mathbf{I}$).

2.3 Structured (Co)sparse priors

In the field of sparse representations and techniques, the notion of *structure* which is basically the idea that the nonzero coefficients of expectedly sparse quantities may not be "indifferently" distributed, is manifold. It has given rise to various definitions and developments, all of which were initially defined in the context of sparse synthesis, but can all be straightforwardly extended to the sparse analysis point of view. In the following, the matrix $\mathbf{A}$ (resp. $\mathbf{D}$) embodies a forward (resp. backward) (redundant) complex frequency transform as described in section 2.2.

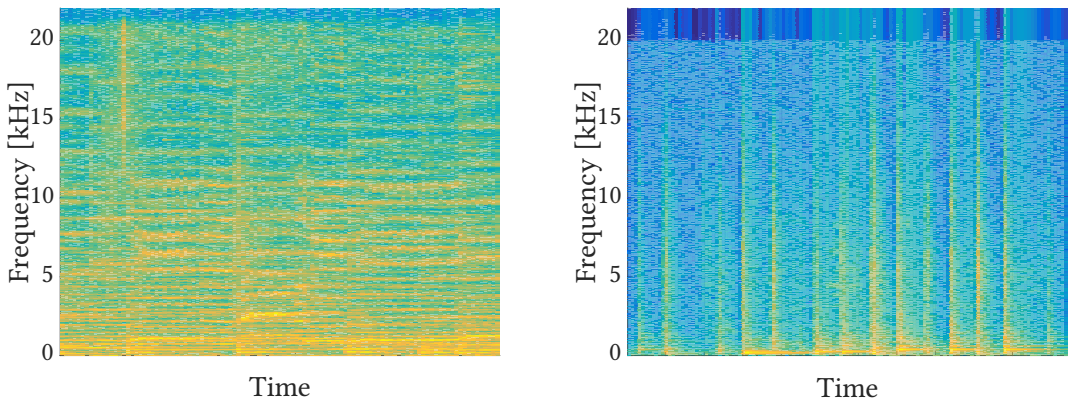
2.3.1 Time-frequency modeling

Simple synthesis sparse models as defined in Equation (2.7) may show some limitations as it considers all the coefficients in the sparse representation independently. However, in the context of audio time-frequency modeling one can argue that coefficients

are rather arranged in groups as shown in Figure 2.5. Structured forms of sparsity such as group sparsity [Kowalski & Torrésani 2009b, Jenatton *et al.* 2011] or social sparsity [Kowalski & Torrésani 2009a, Kowalski *et al.* 2013] have emerged as useful refinements of the simple sparse synthesis technique to take into account the typical time-frequency patterns of audio signals. For example, Figure 2.5a displays the spectrogram of a tonal musical excerpt, where high energy coefficients are structured across time reflecting the strong presence of harmonics. Figure 2.5b displays the spectrogram of a percussive music sample, where the dominant coefficients gather across frequency due to transients and beats. Consider the matrix $\mathbf{X} \in \mathbb{R}^{L \times T}$ which columns are the windowed frames of an original time-domain audio signal $\tilde{\mathbf{x}} \in \mathbb{R}^{N}$, and $\mathbf{Z} \in \mathbb{C}^{S \times T}$ a matrix which columns are a frequency representation of these frames. In other words, this matrix is a time-frequency representation of the underlying audio signal $\tilde{\mathbf{x}}$.

- *Group* sparsity: Consider *non-overlapping* groups of indexes in $\mathbf{Z}$. *Group* sparsity assumes that if some coefficient of the matrix is zero, then all coefficients at indexes belonging to the same group must be also zero, while in "active" groups, no sparsity is required. This prior is typically enforced by minimizing mixed-norms such as the $\ell_{2,1}$ norm (see Equation (2.3)).

- *Social* sparsity extends the previous structure to the case of possibly overlapping groups, and also allows more flexible structures than mixed norms through the use of generic time-frequency patterns. This prior is typically enforced with the use of appropriate specific sparsity promoting operators using dependencies between coefficients.



(a) Tonal music spectrogram (b) Percussive sounds spectrogram

Figure 2.5 – STFT magnitude representation of two music signals

In structured sparse models, the assumed relation between $\mathbf{Z}$ and $\mathbf{X}$ becomes:

Structured analysis model	Structured synthesis model
$\mathbf{A} \in \mathbb{C}^{P \times L}, P \geq L$	$\mathbf{D} \in \mathbb{C}^{L \times S}, S \geq L$
$\mathbf{Z} \simeq \mathbf{AX}, \mathbf{Z} \in \mathbb{C}^{P \times T}$	$\mathbf{DZ} \simeq \mathbf{X}, \mathbf{Z} \in \mathbb{C}^{S \times T}$
$\|\mathbf{Z}\|_0 \ll P \times T$	$\|\mathbf{Z}\|_0 \ll S \times T$
$\mathbf{Z}$ is "structured";	$\mathbf{Z}$ is "structured".

2.3.2 Channel-wise modeling

Aside from time-frequency modeling of monochannel audio signals, sparsity has been used again from the synthesis sparse data model to recover multichannel signals. The concept of *joint* or *simultaneous* sparsity was coined in the mid-2000s. Several vectors are gathered and assumed to admit a sparse decomposition on the same dictionary. The sparse decomposition can be jointly performed rather than vector-wise. This notion is at the basis of the Simultaneous Orthogonal Matching Pursuit (SOMP) algorithm [Tropp *et al.* 2006, Gribonval *et al.* 2008]. This work was then followed some years later with simultaneous multichannel basis pursuit solved using convex approaches [Eldar & Rauhut 2010].

Intuitively, we expect that a joint processing of all channels with sparsity priors (in a multichannel signal reconstruction scenario) could be indeed more efficient than independently processing each channel. This hypothesis was verified in [Gribonval *et al.* 2008] with joint synthesis sparse priors as the reconstruction error decreased while the number of channels increased. Another hypothesis in such multichannel sparse recovery is a twofold prior: simultaneous (co)sparsity of all channels, and group sparsity across channels. The underlying hypothesis behind the structure (group sparsity) here is that nonzero coefficients are roughly distributed equivalently from frequency representation of one channel to another involving channel-wise dependencies. Although this hypothesis seems quite strong, it does not seem unlikely to assume that it holds for multichannel audio recordings from compact microphones antennas. More formally, consider $\widetilde{\mathbf{X}} \in \mathbb{R}^{N \times C}$ a multichannel audio signal and $\mathbf{X} \in \mathbb{R}^{L \times C}$ a matrix gathering a windowed time-frame of that signal, C being the number of channels and L the time samples. Denote $\mathbf{Z}$ its corresponding frequency representation. The main model characteristics derive from the relation between $\mathbf{Z}$ and $\mathbf{X}$ as well as properties of $\mathbf{Z}$. It expresses:

Structured analysis model	Structured synthesis model
$\mathbf{A} \in \mathbb{C}^{P \times L}, P \geq L$	$\mathbf{D} \in \mathbb{C}^{L \times S}, S \geq L$
$\mathbf{Z} \simeq \mathbf{AX}, \mathbf{Z} \in \mathbb{C}^{P \times C}$	$\mathbf{DZ} \simeq \mathbf{X}, \mathbf{Z} \in \mathbb{C}^{S \times C}$
$\|\mathbf{Z}\|_0 \ll P \times C$	$\|\mathbf{Z}\|_0 \ll S \times C$
$\mathbf{Z}$ is "structured across channels";	$\mathbf{Z}$ is "structured across channels".

Figure 2.6 illustrates the group sparse prior used here. This model (new for the cosparse case) will be used to present results on a multichannel audio declipping scenario in section 6.3 page 69.

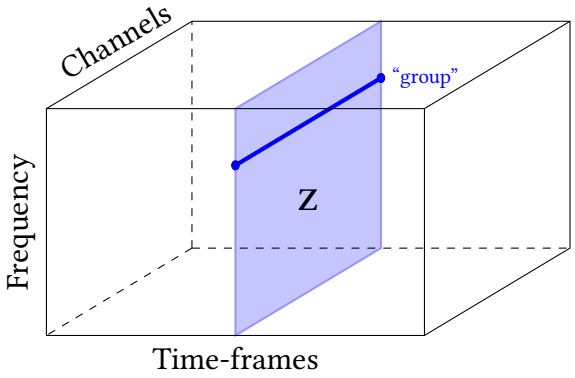


Figure 2.6 – Channel-wise structured sparse modeling

Notably, a wider model can be considered here encompassing the time dimension with tensor tools. This will not be studied in this work but we note that such tensor models used with appropriate three-level structured $\ell_{p,q,r}$ mixed norms can help and were successfully used for magneto-/electro-encephalography source localization [Gramfort & Kowalski 2009].

2.4 Summary

After introducing the basics of sparse modeling, this chapter detailed the two sparsity data models (both *synthesis* and *analysis*) along with regularization methods. Then, we described some useful tools for audio signal (co)sparse modeling. Among them, we presented possibly redundant complex frequency transforms properties. Afterwards, we described structured synthesis sparse priors for time-frequency modeling and proposed an extension to its cosparse counterpart. As structured sparsity is multiform and can be used as well in a multichannel context, we broadened its application spectrum adding channel-wise structured (co)sparse audio signals modeling.

Chapter 3

Algorithmic Framework

Contents

In this chapter we are interested in deriving a generic greedy method able to retrieve degraded audio signals. This method is designed to account for various sparse signal modeling priors among those described in chapter 2. The aim is also to make it versatile to work with several audio distortion problems being induced either by the sensor or the environment. The main algorithmic framework underlying this method is the *Alternating Direction Method of Multipliers* (ADMM). The first section of this chapter will describe the ADMM while the next one will feature the generic audio reconstruction algorithm. The algorithm will be explicitly demonstrated in chapter 5, chapter 6 and chapter 7.

3.1 Alternating Direction Method of Multipliers

Sparse signal reconstruction, whether seen from the analysis or synthesis point of view, often uses (non-) convex optimization to address recovery problems. Initially, ADMM was designed to solve the following convex optimization problems as stated in [Boyd *et al.* 2011]:

$$\underset{\mathbf{W},\mathbf{Z}}{\text{minimize}}\ f(\mathbf{W}) + g(\mathbf{Z})\ \text{subject to}\ \mathbf{M}\mathbf{W} - \mathbf{B}\mathbf{Z} = \mathbf{Q}, \tag{3.1}$$

with $\mathbf{W}$, $\mathbf{Z}$, $\mathbf{M}$, $\mathbf{B}$, and $\mathbf{Q}$ generic real-valued matrices or vectors and $f(\cdot)$, $g(\cdot)$ convex functionals.

To solve the problem described in Equation (3.1) with ADMM, we first consider the corresponding augmented Lagrangian problem defined below:

$$L_\rho(\mathbf{W},\mathbf{Z},\mathbf{V}) = f(\mathbf{W}) + g(\mathbf{Z}) + \langle \mathbf{V}, \mathbf{M}\mathbf{W} - \mathbf{B}\mathbf{Z} - \mathbf{Q}\rangle + \frac{\rho}{2}\|\mathbf{M}\mathbf{W} - \mathbf{B}\mathbf{Z} - \mathbf{Q}\|_{\mathrm{F}}^2, \tag{3.2}$$

where $\rho > 0$. The standard method is performed on the equivalent scaled Lagrangian expression:

$$L_\rho(\mathbf{W}, \mathbf{Z}, \mathbf{U}) = f(\mathbf{W}) + g(\mathbf{Z}) + \frac{\rho}{2}\|\mathbf{MW} - \mathbf{BZ} - \mathbf{Q} + \mathbf{U}\|_F^2 - \frac{\rho}{2}\|\mathbf{U}\|_F^2, \tag{3.3}$$

with $\mathbf{U} = \frac{\mathbf{V}}{\rho}$ the dual variable. The associated problem to Equation (3.1) then becomes:

$$\underset{\mathbf{U}}{\text{maximize}}\,(\underset{\mathbf{W},\mathbf{Z}}{\text{minimize}}\; L_\rho(\mathbf{W}, \mathbf{Z}, \mathbf{U})). \tag{3.4}$$

The interest of the ADMM procedure is that optimization over $\mathbf{W}, \mathbf{Z}, \mathbf{U}$ can be split in three distinct steps described below:

$$\mathbf{W}^{(i)} = \underset{\mathbf{W}}{\operatorname{argmin}}\, f(\mathbf{W}) + \frac{\rho}{2}\|\mathbf{MW} - \mathbf{BZ}^{(i-1)} - \mathbf{Q} + \mathbf{U}^{(i-1)}\|_F^2, \tag{3.5}$$

$$\mathbf{Z}^{(i)} = \underset{\mathbf{Z}}{\operatorname{argmin}}\, g(\mathbf{Z}) + \frac{\rho}{2}\|\mathbf{MW}^{(i)} - \mathbf{BZ} - \mathbf{Q} + \mathbf{U}^{(i-1)}\|_F^2, \tag{3.6}$$

$$\mathbf{U}^{(i)} = \mathbf{U}^{(i-1)} + \mathbf{MW}^{(i)} - \mathbf{BZ}^{(i)} - \mathbf{Q}. \tag{3.7}$$

Definition 2 (Characteristic function)**.** *Let Θ be a nonempty* convex *set, we denote $\chi_\Theta(\cdot)$ the characteristic function of the set Θ* i.e. *for any matrix $\mathbf{W} \in \mathbb{C}^{L\times P}$:*

$$\chi_\Theta(\mathbf{W}) = \begin{cases} 0 & \text{when } \mathbf{W} \in \Theta; \\ +\infty & \text{otherwise;} \end{cases} \tag{3.8}$$

Definition 3 (Proximity operator)**.** *Given f a convex function, the proximity operator of f is defined for any vectors $\mathbf{w}$ $\mathbf{u}$ by:*

$$prox_f(\mathbf{w}) = \underset{\mathbf{u}}{\operatorname{argmin}}\, f(\mathbf{u}) + \frac{1}{2}\|\mathbf{u} - \mathbf{w}\|_2^2 \tag{3.9}$$

Remark. *Proximity operators are usually defined for vectors. We can easily extend it to matrices replacing the Euclidean norm with the Frobenius norm.*

Note that Equation (3.5) and Equation (3.6) can be identified to proximity operators of $f(\cdot)$ and $g(\cdot)$ under certain condition on $\mathbf{M}$ and $\mathbf{B}$, so ADMM being part of the proximal splitting methods framework [Combettes & Pesquet 2011, Parikh & Boyd 2013].

If convergence is ensured when $f(\cdot)$ and $g(\cdot)$ are convex functionals [Eckstein & Yao 2015], this numerical scheme is also widely used as a heuristic for non-convex optimization especially as proximity operators of certain non convex function are easy to compute. Hence, numerous studies [Adler *et al.* 2013, Boyd *et al.* 2011, Chartrand & Wohlberg 2013, Kitić *et al.* 2015] used ADMM as a heuristic for non-convex cases.

3.2 Generic reconstruction framework

In this section, we present a general framework using either simple sparse modeling (analysis or synthesis based) or structured sparse priors to address reconstruction problems in audio. Given a distorted matrix of observations $\mathbf{Y}$, our goal is to find means to

recover an estimate $\hat{\mathbf{X}}$ of the frames $\mathbf{X}$ of the original signal. For this, one seeks $\hat{\mathbf{X}}$ that satisfies:

- a data fidelity constraint with respect to $\mathbf{Y}$, according to some distortion model (additive noise, clipping, reverberation...);
- the modeling constraints described in chapter 2.

This is the spirit of the algorithmic framework we develop. It relies on two components:

- a *generalized projection* onto the data-fidelity constraint;
- a *shrinkage* enforcing (structured) sparsity.

The two next paragraphs detail these components before presenting the algorithm they are embedded in.

Shrinkages Intuitively, this operator gives an output which is "decreased" in a certain sense, with respect to its input argument, hence somewhat promoting sparsity. Although we will not formally exploit it for any convergence analysis, we also recall below the notion of shrinkage, also called "thresholding rule" [Kowalski 2014].

Definition 4 (Shrinkage). *$\mathcal{S}(\cdot)$, is a shrinkage iff:*

1. *$\mathcal{S}(\cdot)$ is an odd function;*
2. *$0 \leq \mathcal{S}(x) \leq x$, for all $x \in \mathbb{R}^+$.*
3. *$(\mathcal{S}(\cdot))_+$ is nondecreasing on $\mathbb{R}^+$ and $\lim_{x\to+\infty}(\mathcal{S}(x))_+ = +\infty$, where $(\cdot)_+ := \max(\cdot, 0)$.*

When applied to a (time-/channel- frequency) matrix, and written $\mathcal{S}(\mathbf{Z})$, shrinkage is applied entry-wise. The different shrinkages are to be adapted depending on the sparse prior to account for (*i.e.* plain or structured sparsity). These shrinkage operators are presented subsection 3.2.2.

Projections We present below the generalized projection tool that will be crucial for fulfilling the different data-fidelity constraints in the algorithm.

Definition 5 (Generalized projection). *Let Θ be a nonempty* convex *set, and $\mathbf{M}$ be a full column rank matrix. Given a time-frequency matrix $\mathbf{Z}$, we denote $\mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z})$ the (unique) solution of the following optimization problem:*

$$\underset{\mathbf{W}\in\Theta}{\text{minimize}} \ \|\mathbf{MW} - \mathbf{Z}\|_F. \tag{3.10}$$

The computation of this projection for some particular choices of constraint set Θ and matrix $\mathbf{M}$ will be discussed in due time. Similarly to the shrinkages, this generalized projection is adapted considering the data-fidelity and the reconstruction problem at hand. The different projections will be introduced independently when instantiating the general algorithm for either denoising (chapter 5), declipping (chapter 6) or dereverberation (chapter 7).

3.2.1 Generic algorithm

Let us now consider seeking a solution for the following non-convex optimization problem which is similar to Equation (3.1):

$$\underset{\mathbf{W},\mathbf{Z}}{\text{minimize}}\ f_m(\mathbf{Z}) + \chi_\Theta(\mathbf{W}) \text{ subject to } \mathbf{MW} - \mathbf{Z} = 0 \tag{3.11}$$

where $\mathbf{W}, \mathbf{Z}$ and $\mathbf{M}$ are generic real matrices. $f_m(\cdot)$ can be identified to a functional embodying a modeling constraint and $\chi_\Theta(\cdot)$ can be identified to a functional corresponding to the data-fidelity term.

As a heuristic, to seek a solution of Equation (3.11), we proceed by forming augmented Lagrangian [Nocedal & Wright 2006, Chapter 17]:

$$L_\mu(\mathbf{W}, \mathbf{Z}, \mathbf{Q}) = f_m(\mathbf{Z}) + \chi_\Theta(\mathbf{W}) + \langle \mathbf{Q}, \mathbf{MW} - \mathbf{Z} \rangle + \frac{1}{2\mu}\|\mathbf{MW} - \mathbf{Z}\|_{\mathrm{F}}^2, \tag{3.12}$$

where $\mathbf{Q}$ is the dual variable and $\mu > 0$. Let $\mathbf{U} = \mu\mathbf{Q}$, then Equation (3.12) becomes:

$$L_\mu(\mathbf{W}, \mathbf{Z}, \mathbf{U}) = f_m(\mathbf{Z}) + \chi_\Theta(\mathbf{W}) + \frac{1}{2\mu}\|\mathbf{MW} - \mathbf{Z} + \mathbf{U}\|_{\mathrm{F}}^2 - \frac{1}{2\mu}\|\mathbf{U}\|_{\mathrm{F}}^2. \tag{3.13}$$

Equation (3.13) is called the scaled Lagrangian. Standard ADMM approach is to perform alternate minimization on Equation (3.13) with respect to each of the primal variables $(\mathbf{W}, \mathbf{Z})$, followed by an update for $\mathbf{U}$:

$$\mathbf{W}^{(\mathrm{i})} = \underset{\mathbf{W}}{\operatorname{argmin}}\ \chi_\Theta(\mathbf{W}) + \frac{1}{2\mu}\|\mathbf{MW} - \mathbf{Z}^{(\mathrm{i}-1)} + \mathbf{U}^{(\mathrm{i}-1)}\|_{\mathrm{F}}^2 \tag{3.14}$$

$$\mathbf{Z}^{(\mathrm{i})} = \underset{\mathbf{Z}}{\operatorname{argmin}}\ f_m(\mathbf{Z}) + \frac{1}{2\mu}\|\mathbf{Z} - \mathbf{MW}^{(\mathrm{i})} - \mathbf{U}^{(\mathrm{i}-1)}\|_{\mathrm{F}}^2 \tag{3.15}$$

$$\mathbf{U}^{(\mathrm{i})} = \mathbf{U}^{(\mathrm{i}-1)} + \mathbf{MW}^{(\mathrm{i})} - \mathbf{Z}^{(\mathrm{i})}, \tag{3.16}$$

We note that thanks to the indicator function $\chi_\Theta(\cdot)$ the $\frac{1}{2\mu}$ factor in Equation (3.14) does not play any role in the minimization. Hence, here the ADMM step described by Equation (3.14) is equivalent to performing the following generalized projection:

$$\mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z}^{(\mathrm{i}-1)} - \mathbf{U}^{(\mathrm{i}-1)}).$$

Similarly, the step described by Equation (3.15) can be identified by:

$$\mathcal{S}_\mu\left(\mathbf{MW}^{(\mathrm{i})} + \mathbf{U}^{(\mathrm{i}-1)}\right).$$

In this work we use only sparsity inducing non-convex shrinkages $\mathcal{S}_\mu(\cdot)$. Our goal here is not to associate any penalty $f_m(\cdot)$ and/or proximity operator, which, if they exist might be difficult to express [Gribonval & Nikolova 2018].

3.2.2 Generic (co)sparse instantiation

In the following, we will use the three alternate minimization steps of ADMM (Equation (3.14), Equation (3.15), Equation (3.16)) to build the generic reconstruction framework. In a concrete setting, the following is required to instantiate the framework:

Requirements.

- *a convex set* Θ *and a matrix* $\mathbf{M}$ *embodying the data fidelity constraint and the domain (time or frequency) in which it is specified;*
- *a parameterized family of shrinkages* $\{\mathcal{S}_\mu(\cdot)\}_\mu$*, where the amount of shrinkage is controlled by* μ*: in the extreme cases* $\mathcal{S}_0(\mathbf{Z}) = \mathbf{Z}$ *and* $\mathcal{S}_\infty(\mathbf{Z}) = \mathbf{0}$*;*
- *a rule* $F : \mu \mapsto F(\mu)$ *to update the amount of shrinkage across iterations, and an initial* $\mu^{(0)}$*;*
- *an initial estimate* $\mathbf{Z}^{(0)}$ *of the seeked time-/channel- frequency representation;*
- *stopping parameters* β *and* $\mathrm{i}_{\max}$.

The proposed generic algorithm is described in Algorithm 1.

Algorithm 1 Generic Algorithm: $\mathcal{G}$

Require: $\Theta, \mathbf{M}, \{\mathcal{S}_\mu(\cdot)\}_\mu, \mu^{(0)}, F(\cdot), \mathbf{Z}^{(0)}, \beta, \mathrm{i}_{\max}$

Initialization step:

$\mathbf{U}^{(0)} = \mathbf{0}$;

for $\mathrm{i} = 1$ to $\mathrm{i}_{\max}$ **do**

Projection step on the data-fidelity constraint:

$\mathbf{W}^{(\mathrm{i})} = \mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z}^{(\mathrm{i}-1)} - \mathbf{U}^{(\mathrm{i}-1)})$ Equation (3.14)

Projection step on the modeling constraint:

$\mathbf{Z}^{(\mathrm{i})} = \mathcal{S}_{\mu^{(\mathrm{i}-1)}}\left(\mathbf{M}\mathbf{W}^{(\mathrm{i})} + \mathbf{U}^{(\mathrm{i}-1)}\right)$ Equation (3.15)

Update step:

if $\frac{\|\mathbf{M}\mathbf{W}^{(\mathrm{i})} - \mathbf{Z}^{(\mathrm{i})}\|_\mathrm{F}}{\|\mathbf{M}\mathbf{W}^{(\mathrm{i})}\|_\mathrm{F}} \leq \beta$ **then**

terminate

else

$\mathbf{U}^{(\mathrm{i})} = \mathbf{U}^{(\mathrm{i}-1)} + \mathbf{M}\mathbf{W}^{(\mathrm{i})} - \mathbf{Z}^{(\mathrm{i})}$ Equation (3.16)

$\mu^{(\mathrm{i})} = F(\mu^{(\mathrm{i}-1)})$

return $\mathbf{W}^{(\mathrm{i})}$ [and optionally $\mu^{(\mathrm{i})}, \mathbf{Z}^{(\mathrm{i})}$]

The notation $\mathbf{Z}^{(\mathrm{i})}$ highlights that the corresponding variable is in any use-case a sparse/structured time-/channel- frequency representation. The variable $\mathbf{U}^{(\mathrm{i})}$ is an intermediate time-/channel- frequency "residual" variable typical of ADMM. At iteration i, an estimate of $\mathbf{Z}$ is $\hat{\mathbf{Z}}^{(\mathrm{i})} := \mathbf{Z}^{(\mathrm{i}-1)} - \mathbf{U}^{(\mathrm{i}-1)}$. The interpretation of the other variables is use-case dependent:

- **analysis flavor:** $\mathbf{M} := \mathbf{A}$ is the frequency analysis operator; $\mathbf{W}^{(i)}$ is an estimate of the time frames $\mathbf{X}$, that satisfies the time-domain data-fidelity constraint Θ while being closest to $\hat{\mathbf{Z}}^{(i)}$ *in the time-/channel- frequency domain*; the algorithm outputs a time-domain estimate.
- **synthesis flavor:** $\mathbf{M} := \mathbf{I}$; $\mathbf{W}^{(i)}$ is a time-/channel- frequency estimate of $\mathbf{Z}$; the data-fidelity constraint Θ is *expressed in the time-/channel- frequency domain*; the algorithm outputs a time-/channel- frequency estimate, from which it is possible to get a time-domain estimate by synthesis $\hat{\mathbf{X}} := \mathbf{D}\mathbf{W}^{(i)}$ with $\mathbf{D}$ the inverse frequency transform operator.

Due to the expression of Θ respectively in the time domain and the time-/channel-frequency domain, the analysis and synthesis flavors can have different computational properties as will be further studied.

Shrinkage for plain sparsity To enforce the plain sparse data model, either analysis or synthesis, we use the hard-thresholding operator $\mathcal{H}_k(\mathbf{Z})$ that sets all but the k coefficients of largest magnitude in $\mathbf{Z}$ to zero (see e.g. [Blumensath & Davies 2009]). In the case of analysis (resp. synthesis) sparse modeling with $\mathbf{A} \in \mathbb{C}^{P\times L}$ a forward frequency analysis operator (resp. $\mathbf{D} \in \mathbb{C}^{L\times S}$ a dictionary) we set $\mathcal{S}_\mu := \mathcal{H}_{P-\mu}$ (resp. $\mathcal{S}_\mu := \mathcal{H}_{S-\mu}$), for $\mu \in \mathbb{N}^+$, $0 \le \mu \le P$ (resp. $0 \le \mu \le S$).

Shrinkage for time-frequency structured sparsity For social sparsity (again, either analysis or synthesis), we choose the Persistent Empirical Wiener (PEW) operator [Kowalski 2014] successfully used in [Siedenburg *et al.* 2014] for audio declipping. This shrinkage promotes specific local time-frequency structures around each time-frequency point. Its specification explicitly requires choosing a time-frequency pattern described as a matrix $\Gamma \in \mathbb{R}^{(2F+1)\times(2T+1)}$ with binary entries.

Rows of Γ account for the frequency dimension and columns for the time dimension, in *local time-frequency coordinates*. Let $\mathbf{Z} \in \mathbb{C}^{L\times(2b+1)}$ be a time-frequency representation. As illustrated on Figure 3.1, consider ij the coordinates of a time-frequency point in $\mathbf{Z}$ and $\mathbf{P}_{ij} := [i-F, i+F] \times [j-T, j+T]$ the indices corresponding to a time-frequency patch of size $(2F+1)\times(2T+1)$ centered in ij. The matrix $\mathbf{Z}_{\mathbf{P}_{ij}} \in \mathbb{C}^{(2F+1)\times(2T+1)}$ is extracted from $\mathbf{Z}$ on these indices, with mirror-padding on the borders if needed.
Now that we have expressed how $\mathbf{Z}$, $\mathbf{Z}_{\mathbf{P}_{ij}}$ and indexes are organized, we can define PEW using $\circ$ to denote the Hadamard product and $(\cdot)_+ = \max(\cdot, 0)$ the positive part:

$$\mathcal{S}_\mu^{\mathrm{PEW}}(\mathbf{Z}|\Gamma)_{ij} := \mathbf{Z}_{ij} \cdot \left(1 - \frac{\mu^2}{\|\mathbf{Z}_{\mathbf{P}_{ij}} \circ \Gamma\|_2^2}\right)_+ . \tag{3.17}$$

Since $\|\mathbf{Z}_{\mathbf{P}_{ij}} \circ \Gamma\|_F^2$ is the energy of $\mathbf{Z}$ restricted to a time-frequency neighborhood of ij of shape specified by Γ, the left hand side is zero as soon as this energy falls below μ^2. As such, PEW shrinkage effectively promotes structured sparsity.

Examples of time-frequency patterns Γ chosen for music are given in Figure 3.2 and for speech in Figure 3.3. They are similar but at different time scales, given the different

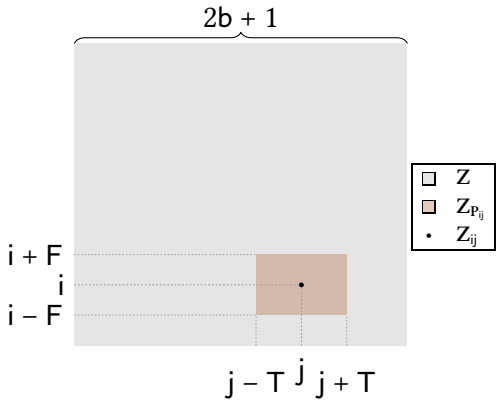
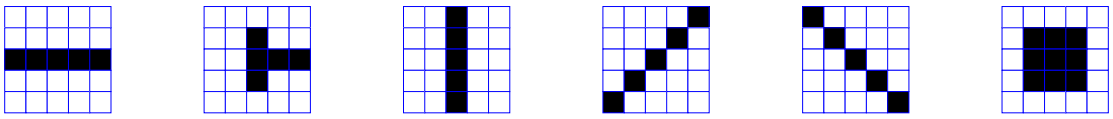


Figure 3.1 – Schematic representation of patch extraction from matrix $\mathbf{Z}$

scales of stationarity in speech and music. The structures embedded in these patterns have various properties: Γ_1, with a frequency localized and time-spread support, will emphasize tonal content; vice-versa, Γ_3 will emphasize transients and attacks; Γ_2 is designed [Siedenburg & Dörfler 2012] to avoid pre-echo artifacts; patterns Γ_4 and Γ_5 are introduced to stress tonal transitions; finally, Γ_6 serves as a default pattern when no particular structure is identified.

Remarks: Here the subscript index k for each time-frequency pattern $\Gamma_k, k \in \{1..6\}$ is not a time frame index but counts the patterns within the collection.
On Figure 3.2 and Figure 3.3 the unit is the time-frequency index of the DFT. In the experimental sections, as we will take 64 ms long time-frames for music and 32 ms long time-frames for speech, the total time span for each Γ is 320 ms for music. For speech the total time span reduces to 96 ms.



(a) Γ_1 (b) Γ_2 (c) Γ_3 (d) Γ_4 (e) Γ_5 (f) Γ_6

Figure 3.2 – Extended set of time-frequency neighborhoods used for music

(a) Γ_1 (b) Γ_2 (c) Γ_3 (d) Γ_4 (e) Γ_5 (f) Γ_6

Figure 3.3 – Extended set of time-frequency neighborhoods used for speech

Shrinkages for channel-aware structured sparsity For multichannel scenarii, in order to enforce the channel-wise structured sparse modeling, we use the Group Empirical Wiener (GEW) operator as in [Févotte & Kowalski 2015] as a sparsifying step in the procedure. Let $\mathbf{Z} \in \mathbb{C}^{P \times K}$ be a local multichannel frequency representation to sparsify. Let pk be coordinates of point in such a local representation and $\mathbf{z}_p \in \mathbb{C}^{1 \times K}$ the p-th row from matrix $\mathbf{Z}$ (corresponding to a group, as illustrated on Figure 2.6 page 21). GEW is defined as:

$$\mathcal{S}_{\mu}^{\text{GEW}}(\mathbf{Z})_{pk} = \mathbf{Z}_{pk} \cdot \left(1 - \frac{\mu^2}{\|\mathbf{z}_p\|_2^2}\right)_+, \tag{3.18}$$

with $(\cdot)_+ = \max(\cdot, 0)$ the positive part and μ the parameter controlling the amount of shrinkage to apply. This shrinkage explicitly promotes group sparsity of $\mathbf{Z}$ along the channel dimension ($\mathbf{z}_p$).

Additionally, we will also use another shrinkage similar to GEW and defined as:

$$\mathcal{S}_{\mu}^{\text{Quad-GEW}}(\mathbf{Z})_{pk} = \mathbf{Z}_{pk} \cdot \left(1 - \frac{\mu^2}{\|\mathbf{z}_p\|_2^4}\right)_+. \tag{3.19}$$

Note that this last shrinkage just differs from GEW by the power of the norm, hence we call this shrinkage the "Quadratic Group Empirical Wiener" (Quad-GEW) . We fortuitously discovered that this last shrinkage could yield better results for certain signal reconstruction problems. It will be more specifically used within the framework for stereo signal declipping in chapter 6.

3.3 Summary

After a short review of the ADMM iterative method, this chapter presents useful components (generalized projection and shrinkages) for deriving a generic audio reconstruction framework. This framework stresses the use of ADMM as a heuristic for approaching a solution of possible non-convex optimization problems. Finally, this chapter details the requirements of the framework and provides pseudo-code of the algorithmic procedure (Algorithm 1). This will be the shared baseline for tackling audio reconstruction in the next parts of this manuscript.

We can summarize Algorithm 1 as a generalized procedure:

$$\mathcal{G}(\Theta, \mathbf{M}, \{\mathcal{S}_{\mu}\}_{\mu}, \mu^{(0)}, F, \mathbf{Z}^{(0)}, \beta, \mathrm{i}_{\max}). \tag{3.20}$$

Chapter 4

Test data and performance measures

Contents

As the algorithmic framework presented above will be used for solving different audio reconstruction problems in the rest of this thesis, one essential question is the choice of data and performance measure to rate the effectiveness of algorithms derived from this framework. Several factors can guide the choice of signals and metrics for performance assessment. The data to test are often chosen depending on the signal models at hand. With audio signal processing application in mind, the reader could probably guess that simulated or recorded sounds will be used to illustrate the usefulness of the different methods presented in the remainder of this thesis. The choice of the metric could be directly drawn from the target application in the case of a signal processing pipeline or be more generic. For example if the back-end application involves speech recognition [Rabiner & Juang 1993], a good choice can be the "Word Error Rate" [Deléglise *et al.* 2009] giving information on the failed recognized words. Otherwise, the metric can change regarding what kind of data is concerned. Indeed, we can easily guess that the measuring index will not be the same depending on the information conveyed by the signal. In the following, the reference signals and data which are used for most of the experimental validation in this work are described. Then, we present the different measures which are used all along this manuscript to assess the performance of the algorithms and methods.

4.1 Test data

Sounds in our every day life can be really diverse. Hence, it is of great concern to pay specific attention to the choice of the audio examples used for experimental validation on audio reconstruction methods. Audible stimuli are widely used to alert, inform, communicate or for artistic purposes with music for example. Although this last sentence bounds music to its artistic dimension, some work [Hargreaves *et al.* 2005] reviewed its important role in communication.

For these reasons, an obvious choice for the test audio data is both speech and music. Even if freely available large databases existed for a long time for speech [Cole *et al.* 1995], some years ago it was still missing for music. In 2002, the Real World Computing partnership in Japan released a copyright-cleared dataset for research purpose. This large database includes various music genres such as pop music, jazz music or classical music. For the pop music category the RWC database features 100 songs, it has 50 examples for the jazz and classical subsets. For the later, the genre is sub-categorized as "symphonies" (4 pieces), "concerti" (2 pieces), "orchestral music" (4 pieces), "chamber music" (10 pieces), "solo performances" (24 pieces) and "vocal performances" (6 pieces). To conduct experimental studies, we choose to rearrange the classical part of the dataset. We let the "vocal performances" category as is, as it features a quite distinct content than the other subsets. We will call it *Vocals* when presenting some results. We group the "symphonies", "concerti" and "orchestral music" as the pieces are performed by large groups of musicians. We will later call this subset *Orchestra*. Finally, we gathered "chamber music" and "solo performances" as the pieces are here played by small groups of musicians. This grouping will be later called *Chamber*.

Duration for music For the tested musical content, we are targeting around 50 minutes of audio content for each category. Therefore, we adapt the length of each excerpt accordingly. For that purpose, we select randomly with uniform probability a 30 second excerpt for each song of the "Popular Music" subset which will be denoted *Pop* in the experimental sections. Similarly, we choose a 1 minute sample of each song in the "Jazz Music" subset that will be called later *Jazz*. For the "Classical Music", we use the *Vocals* as is. We randomly pick a 5 minute excerpt for 9 of the examples in the *Orchestra* subset. In the *Chamber* subset, we perform uniform random selection to take a 90 second excerpt for 35 sound examples. The list below summarizes the total audio content tested:

- *Pop*: 100 songs × 30 seconds (total 50 minutes);
- *Jazz*: 50 pieces × 60 seconds (total 50 minutes);
- *Classic*
  - *Chamber*: 35 pieces × 90 seconds (total 52.5 minutes);
  - *Orchestra*: 9 pieces × 5 minutes (total 45 minutes);

– *Vocals*: 6 songs (various lengths, total ~ 22 minutes).

The RWC database originally features stereo recordings sampled with CD quality at 44.1 kHz. For experiments requiring mono-channel recordings, we use down-sampled version of the audio tracks to 16 kHz as a compromise between quality and execution time for the reconstruction algorithms. The mono-channel signals are generated by averaging the original stereo signals.

Notably, other subsets are available in the RWC Music database ("Royalty-Free Music", "Music Genre" and "Musical Instrument Sound") but they will not be used for this work.

Speech For speech content, in the late 1980's Texas Instrument at the MIT released an acoustic and phonetic speech corpus [Garofolo *et al.* 1993] (TIMIT). This transcribed speech data have been widely used for automatic speech recognition systems assessment [Graves *et al.* 2013]. This speech database provides various American English versions of sentences read by native speakers recorded at 16 kHz and in mono-channel. Later in this thesis, the experiments involving speech will feature 135 samples extracted from the whole corpus and freely available for a total around 10 minutes of speech audio content. This subset will be denoted as "TIMIT" in the rest of the document.

Multichannel data Additionally, for multichannel tests, we perform experiments on 8-channels recording excerpts from the VoiceHome2 Corpus [Bertin *et al.* 2019][1]. We use the 359 clean speech available examples (total duration: about one hour) and the 118 mixed music and speech examples (total duration: 20 minutes). This multichannel dataset was initially designed to account for various possible smart home interactions and features much more available sounds with different noisy conditions which we leave aside in this work. It is sampled at 16 kHz. For experiments involving stereo recordings, we use the original RWC audio data from the same categories.

Small scale database On top of these large scale audio databases, first pilot studies are conducted on audio examples provided on the *SMALL Project* webpage[2]. These audio files are distributed under Creative Commons Sampling Plus License along with the *SMALLbox* software. It is featured by the audio inpainting toolbox [Adler *et al.* 2012]. We use the 10 speech and 10 music examples (5 second each) sampled at 16 kHz that are used for experimental validation on this toolbox. They are later in this manuscript referred to as "SMALL dataset" or "SMALLbox examples".

[1] http://voice-home.gforge.inria.fr/voiceHome-2_corpus.html
[2] http://www.small-project.eu

Table 4.1 summarizes the data used for experiments.

Table 4.1 – Test data summary

	SMALL	*Pop*	*Jazz*	*Chamber*	*Orchestra*	*Vocals*	TIMIT	voiceHome2
Duration [s]	100	3000	3000	3150	2700	1320	600	1200
Excerpts Nb.	20	100	50	35	9	6	135	477
Channel Nb.	1	1 (2)	1 (2)	1 (2)	1 (2)	1 (2)	1	8
Fs [kHz]	16	16	16 (44.1)	16	16	16	16	16

Finally, data used in chapter 8 for source localization will be presented in due time. The dataset for this task is itself a contribution and is specific to this chapter, none of the other experimental sections in this manuscript will use it.

4.2 Objective measures

Performance measures are typically qualified into objective measure and subjective quality evaluation. On the one hand, objective evaluation requires a numerical comparison between the original signals and the processed ones. In this way, it is possible to quantify numerically the "divergence" between the processed or a degraded (audio) signal and a reference. On the other hand subjective quality measures involve, for instance, features rating by a group of listeners between reference, original/unprocessed and processed sounds. In the following we choose to describe the objective evaluation tools which will be used in the experimental sections.

4.2.1 Signal-to-Noise Ratio

Signal-to-Noise Ratio (SNR) is a widely used metric to compute the distance between two signals in the context of additive noise. Let $\mathbf{x} \in \mathbb{R}^L$ be a reference discrete signal and $\mathbf{y} \in \mathbb{R}^L$ a noisy or processed signal. The SNR is usually expressed on a logarithmic scale in dB and represents the ratio between the power of the signal of interest and the power of the noise. For the additive noise vector sum case $\mathbf{y} = \mathbf{x} + \varepsilon$ the SNR writes:

$$\mathrm{SNR}(\mathbf{x}|\varepsilon) = \frac{\|\mathbf{x}\|_2^2}{\|\varepsilon\|_2^2}. \tag{4.1}$$

It is common to express it also in logarithmic scale:

$$\mathrm{SNR}_{\mathrm{dB}}(\mathbf{x}|\varepsilon) = 10 \cdot \log_{10}\left(\frac{\|\mathbf{x}\|_2^2}{\|\varepsilon\|_2^2}\right), \tag{4.2}$$

with $\lim_{\varepsilon \to 0} \mathrm{SNR}_{\mathrm{dB}} = +\infty$ in the noiseless case. In this work, as we are interested in comparing signal reconstruction we will rather use the SNR difference (ΔSNR) where $\hat{\mathbf{x}} \in \mathbb{R}^L$ is the estimated denoised signal:

$$\Delta\mathrm{SNR} = 10 \cdot \log_{10}\left(\frac{\|\mathbf{x}\|_2^2}{\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2}\right) - 10 \cdot \log_{10}\left(\frac{\|\mathbf{x}\|_2^2}{\|\varepsilon\|_2^2}\right). \tag{4.3}$$

4.2.2 Signal-to-Distortion Ratio

When no additive noise is at stake, a equivalent metric called Signal-to-Distortion Ratio (SDR) is available to quantify the gap between a degraded or reconstructed signal and its original version. The SDR is defined similarly to Equation (4.2), the only difference can come from the interpretation of the $\|\mathbf{x}-\mathbf{y}\|_2^2$ quantity. In the additive noise case, it is seen as the noise power whereas for other scenarii, we can simply interpret it as the distance between the two signals induced by the degradation or restoration. The SDR writes:

$$\mathrm{SDR}_{\mathrm{dB}}(\mathbf{x}|\mathbf{y}) = 10 \cdot \log_{10}\left(\frac{\|\mathbf{x}\|_2^2}{\|\mathbf{x}-\mathbf{y}\|_2^2}\right). \tag{4.4}$$

Similarly, we are interested in comparing signal reconstruction and will rather use the SDR difference (ΔSDR) where $\hat{\mathbf{x}} \in \mathbb{R}^L$ is the estimated reconstructed signal:

$$\Delta\mathrm{SDR} = \mathrm{SDR}_{\mathrm{dB}}(\mathbf{x}|\hat{\mathbf{x}}) - \mathrm{SDR}_{\mathrm{dB}}(\mathbf{x}|\mathbf{y}). \tag{4.5}$$

4.3 Perceptually inspired measures

An objective audio quality measure is well-founded if it correlates with subjective listening assessments. For that reason, some research work focused on finding objective descriptors that represent what happens along the auditory pathway. In the following, we present some of these descriptors that are used to depict global audio quality with objectivity. Such measures will be used in most of the experimental sections of this manuscript. If different metrics are used, they will be disambiguated in the text.

4.3.1 Estimating global audio quality

Except for speech specific methods which will be detailed in the next subsection, we trace back only few attempts to rate quality of wide-band audio content. Triggered by advances on the human auditory system understanding and more precisely the time-frequency masking effects and non linearities in the ear, [Brandenburg 1987] proposed a metric (segmental Noise-to-Mask Ratio) to rate audio quality. Later, the International Telecommunication Union (ITU) started its standardization activities starting with speech quality measurements. The Perceptual Evaluation of Audio Quality (PEAQ) method was proposed as a recommendation ([Thiede *et al.* 2000]) after some other rating techniques in the 1990's mainly relying on psychoacoustics models. Figure 4.1 below presents the overall functioning of the PEAQ for rating audio quality. After producing a frequency representation of the reference and tested signals with a peripheral ear model, the method extracts features such as loudness profiles, noise-to-mask ratio, temporal modulation.
These time-dependent descriptors are averaged to obtain a single value called the Model Output Values (MOV). These MOV are finally mapped to an Objective Difference Grade (ODG) rating the perceived degradation between the signals. This ODG score ranges on

a 5-grade scale from -4 (Very annoying) to 0 (Imperceptible). For the experiments of this thesis, we will use this PEAQ ODG descriptor to rate the audio quality. We use for that the code[3] available with the exhaustive review of PEAQ [Kabal 2002] which implements the ITU-R BS 1387.1 recommendation.

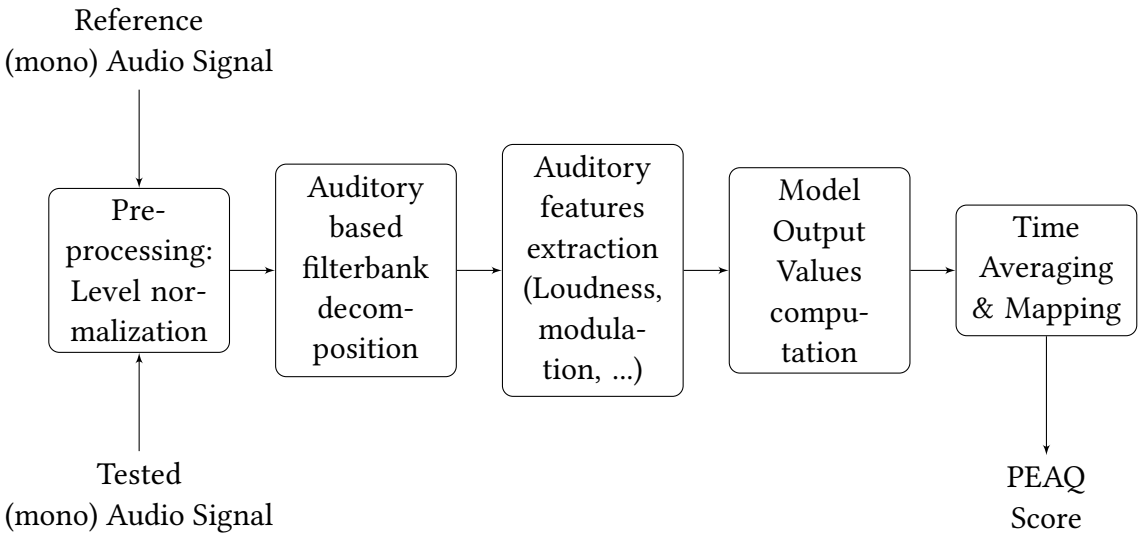


Figure 4.1 – PEAQ computation

As stated previously, we will be interested in the experimental sections to perform a comparison before and after signal reconstruction. Note that PEAQ($\mathbf{x}|\mathbf{y}$) measures the audio quality between a clean reference $\mathbf{x}$ and its corrupted version $\mathbf{y}$, $\hat{\mathbf{x}}$ or its restored estimate. We will use as a comparison measure the PEAQ difference (ΔPEAQ) defined below:

$$\Delta\text{PEAQ} = \text{PEAQ}(\mathbf{x}|\hat{\mathbf{x}}) - \text{PEAQ}(\mathbf{x}|\mathbf{y}). \tag{4.6}$$

4.3.2 Estimating global speech quality

Due to the specificity of speech among other audio signals, some research have been focusing on dedicated measures to rate the global quality of speech. Most of these measures are based on a short-time frame segmentation (10 to 30 ms) prior to a divergence calculation regarding a reference signal.

For instance, we can cite attempts to take into account silences in speech through the segmental Signal-to-Noise Ratio [Richards 1965, Hansen & Pellom 1998] (SNR_{seg}). We can cite spectral distance measures. These measures, based on Linear Predictive Coding (LPC) coefficients [Vaidyanathan 2007], rely on an all-pole modeling for speech. These objective measures, even if they are simple to get, do not reflect well the subjective quality as no auditory processing model is embedded in the computation. On the

[3] http://www-mmsp.ece.mcgill.ca/Documents/Software/

contrary, the SNR_{seg} aforementioned measure was later formulated in the frequency-domain [Tribolet *et al.* 1978] opening for perceptually motivated weightings. Equivalently, in [Klatt 1982], were introduced "weighted spectral slope" metrics triggered by findings on vowels distance rating. These first steps were followed by distortion measures on auditory oriented frequency weightings such as Bark scales. Motivated by the findings described above, a measure assumed to cover several speech degradation and distortion was promoted in the ITU-T recommendation P.862 [ITU-T 2001]. This metric named "Perceptual Evaluation of Speech Quality" (PESQ) is for now considered as one of the most reliable metric to predict the overall speech quality. Figure 4.2 describes the global functioning of the PESQ estimator. PESQ was first designed to account mainly for network or telecommunication distortion. For this reason, in the pre-processing step, the signals are equalized following the typical frequency response of a telephone. The ITU recommendation was slightly modified to account for wide-band (binaural) signals [ITU-T 2007] (ITU-T recommendation P.862.2). Practically, after applying an auditory model to the signals (based on a Bark frequency scale) the loudness spectra are estimated. From the loudness spectra differences (disturbance), the wideband PESQ predicts a Mean Opinion Score (MOS) as it could be retrieved from genuine listening tests. PESQ output scores range from 1 (bad) to 5 (excellent). In the remainder of this thesis, we will use this wideband PESQ descriptor to rate speech quality. We use for that the code provided along with [Loizou 2013].

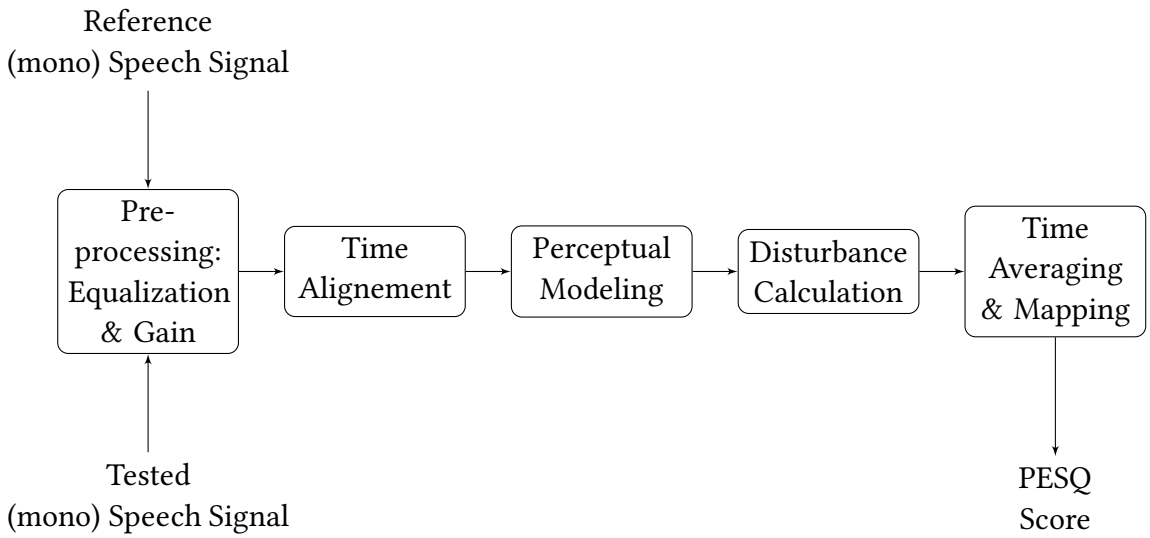


Figure 4.2 – PESQ computation

Similarly, we define as a comparison measure the PESQ difference (ΔPESQ):

$$\Delta\text{PESQ} = \text{PESQ}(\mathbf{x}|\hat{\mathbf{x}}) - \text{PESQ}(\mathbf{x}|\mathbf{y}). \tag{4.7}$$

4.3.3 Estimating the speech intelligibility

Whereas speech quality can be highly subjective and difficult to rate accurately, speech intelligibility provides an objective way to estimate the speech understanding. Involving a group of people, speech intelligibility can be measured asking them to identify words or phonemes.

Intelligibility tests with a group of listeners are very complex to handle. For instance, for controlled testing conditions, you need proper stimuli calibration along with very low background noise level environment and high quality sound reproduction systems. These constraints taken into account, it is a troublesome procedure that is almost impossible to handle on a large scale perspective. Various studies tried to correlate the speech intelligibility to objective numerical measures that can be easily calculated. Among these studies we can cite the initial work on *Articulation Index* (AI) [French & Steinberg 1947], or later its evolution *Coherence Speech Intelligibility Index* (CSII) [Kates & Arehart 2005]. AI was first designed to predict non-sense syllable discrimination based on frequency band weighted SNR. While AI was triggered by telephone applications, CSII came along with hearing devices for speech enhancement. CSII uses power spectrum density ratios between noisy or processed signals as well as auditory filters to predict speech intelligibility. More recently, the STOI (Short Time Objective Intelligibility) index [Taal *et al.* 2010] was introduced. This index was shown to better correlate with real human speech intelligibility than CSII [Taal *et al.* 2011]. It is also described as more suitable to assess intelligibility on speech processed with time-frequency thresholding methods. Hence, we decide to use this STOI index as a reasonably good speech intelligibility prediction. To compute the STOI, we use the code available withing the *Auditory Modeling Toolbox*[4] [Søndergaard & Majdak 2013]. Figure 4.3 displays the global functioning of STOI index calculation.

[4] http://amtoolbox.sourceforge.net

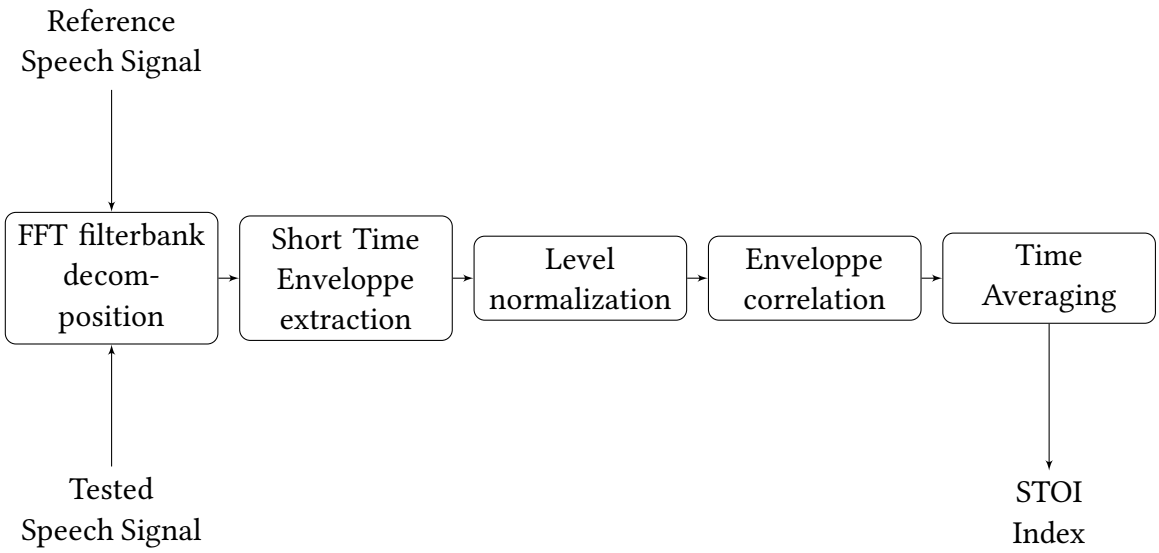


Figure 4.3 – STOI computation

Similarly to the ΔPEAQ and ΔPESQ we define as a comparison measure the STOI difference (ΔSTOI) that will be used in the experimental sections:

$$\Delta\text{STOI} = \text{STOI}(\mathbf{x}|\hat{\mathbf{x}}) - \text{STOI}(\mathbf{x}|\mathbf{y}). \tag{4.8}$$

4.4 Summary

This short chapter presented the data used later in this manuscript for experimental validation. This chapter is also dedicated to introduce the performance measures that are used to rate the reconstruction algorithms efficiency. These different measures presented earlier are so-called *intrusive* measures as they imply comparison with a reference signal. However, recent studies [Sharma *et al.* 2016, Andersen *et al.* 2017] try to present *non-intrusive* measures embedding more accurate speech models. We do not detail these here as they are more suitable for blind quality evaluation rather than quality comparison involving a reference.

Part II

Handling sensor distortion in audio inverse problems

Chapter 5

Denoising

Contents

This chapter will focus on the audio denoising reconstruction problem. After briefly describing the additive noise issue, the chapter will feature a short review of available denoising techniques. Then, we introduce applications of (structured) (co)sparsity for a monochannel reconstruction scenario emphasizing an simple sparse or adaptive time-frequency modeling. Each of these two applications is an instance of the generic framework introduced earlier (chapter 3). Before concluding this current chapter, some experiments including comparisons with a baseline denoising method will be detailed.

This chapter is inspired from [Gaultier *et al.* 2017a]: Clément Gaultier, Nancy Bertin, Srđan Kitić and Rémi Gribonval. *A modeling and algorithmic framework for (non) social (co) sparse audio restoration.* arXiv preprint arXiv:1711.11259, 2017 and to a lesser extent from [Gaultier *et al.* 2017c]: Clément Gaultier, Srđan Kitić, Nancy Bertin and Rémi Gribonval. *AUDASCITY: AUdio Denoising by Adaptive Social CosparsITY.* In 2017 25th European Signal Processing Conference (EUSIPCO), pages 1265–1269, Aug 2017. More precisely, section 5.3 presents new experimental results.

5.1 The noise and denoising problems

This section first presents the considered noise issue in this work before reviewing the most common noise reduction methods.

5.1.1 The noise problem on audio recordings

Denoising is one of the most intensively studied inverse problems in audio signal processing. Whether it originates from the environment or the sensors, noise is an inevitable (and, usually, undesirable) component of audio recordings, calling for a denoising stage in signal processing pipelines for applications such as music transcription, sound classification, speech recognition and many others. Understanding the noise degradation is of great importance before trying to design any noise reduction technique. Whether it be multiplicative, additive or convolutional noise, how the noise affects the signal of interest should drive the denoising solution to use. In this work we will focus on the well known additive noise model defined below by the simple vector sum:

$$\mathbf{y} = \mathbf{x} + \mathbf{n}, \tag{5.1}$$

where $\mathbf{y} \in \mathbb{R}^L$ is a noisy signal composed of $\mathbf{x} \in \mathbb{R}^L$ an original clean signal and $\mathbf{n} \in \mathbb{R}^L$. In this work, we will focus on white Gaussian noise hence $\mathbf{n}$ following a gaussian distribution ($\mathbf{n} \sim \mathcal{N}(0, \sigma^2)$).

5.1.2 Prior art on noise reduction

We can trace back the early denoising attempts mainly for speech enhancement. To address the noise problem, numerous approaches arose. Some of these use statistical models others use spectral subtraction, Wiener filtering or thresholding operators.

Spectral subtraction Spectral subtraction is probably one of the most studied method for noise reduction since the end of the 1970's [Boll 1979]. It is a frequency domain method based on the idea that the clean signal frequency estimate can be obtained by removing the noise spectrum from the noisy signal frequency representation. The time domain signal is then obtained by inverse frequency transform. Eventually this method is relatively computationally efficient as it requires only a forward and an inverse frequency transform. However, it also needs an accurate estimation of the noise spectrum. As long as signal and noise have overlapping spectral content, this method comes with a price: the introduction of musical noise or signal distortion [Berouti *et al.* 1979]. Over the years, several solutions were introduced to alleviate this effect such as performing over-subtraction or controlling a remaining noise floor acting as a musical noise masker.

Wiener filter Wiener filtering [Wiener 1949] is a method that requires an estimation of the signal power and the noise power (*i.e.* the SNR). An optimal filter ($H(\mathrm{f})$) can be expressed for a given frequency f by:

$$H(\mathrm{f}) = \frac{\mathrm{SNR}_{freq}(\mathrm{f})}{1 + \mathrm{SNR}_{freq}(\mathrm{f})}, \tag{5.2}$$

where $\mathrm{SNR}_{freq}(\mathrm{f})$ denotes the SNR for a given frequency f linking the power spectra of the noise and the signal. Even if Wiener filtering is a relatively old method, in the context of audio denoising, it benefited from some work improving it with iterative filtering or auditory oriented weightings [Hu & Loizou 2004].

Sparsity and thresholding As the time-frequency domain became the flag-ship for audio denoising, in the last two decades, a body of work addressing reconstruction and inverse problems in audio popularized sparse regularization. Additionally, it was recognized that group sparse models. Hence, a method promoting group-sparse time-frequency signal prior [Siedenburg & Dörfler 2012] was showing perceptual quality improvements compared to the Block-thresholding [Yu *et al.* 2008] method. The latter, uses disjoint clusters in the time-frequency representation of noisy signals to estimate a local time-frequency dependent SNR and then a single attenuation factor by block minimizing an estimate of $\|\mathbf{y}-\hat{\mathbf{x}}\|_2$. This last method will be the denoising baseline in our experimental section.

5.2 (Co)sparse denoisers

In the following section we introduce several denoising methods derived from the algorithmic framework presented in chapter 3. These methods will embed regular or structured time-frequency (co)sparse data models. After listing the required projection operators, we instantiate the different versions of Algorithm 1. We consider the matrix $\mathbf{Y} \in \mathbb{R}^{L\times(2b+1)}$ containing one or more windowed frames of L samples from the observed signal $\tilde{\mathbf{y}}$ ($2b+1 \geq 1$). The denoising problem is to estimate the original clean signal frames, similarly gathered in a matrix $\mathbf{X}$ of the same size.

5.2.1 Generalized projections for the denoising problem

A natural expression of the data-fidelity constraint is of the form $\|\hat{\mathbf{X}} - \mathbf{Y}\|_{\mathrm{F}} \leq \varepsilon$ for some ε. Heuristics to choose ε given an estimated variance σ^2 will be discussed in section 5.3.

In the analysis setting, we recall that the estimate $\mathbf{W} \in \mathbb{R}^{L\times(2b+1)}$ is a matrix of time-frames. With $\mathbf{M} := \mathbf{A}$, the data-fidelity constraint yields $\Theta := \{\mathbf{W} \mid \|\mathbf{W} - \mathbf{Y}\|_{\mathrm{F}} \leq \varepsilon\}$. In the synthesis setting, the estimate $\mathbf{W} \in \mathbb{C}^{L\times(2b+1)}$ is a (time)-frequency representation. With $\mathbf{M} := \mathbf{I}$, we set $\Theta := \{\mathbf{W} \mid \|\mathbf{D}\mathbf{W} - \mathbf{Y}\|_{\mathrm{F}} \leq \varepsilon\}$. These choices hold both for plain and social versions.

In the analysis setting, assuming $\mathbf{A}^{\mathsf{H}}\mathbf{A} = \mathbf{I}$, the desired projection can be expressed in closed-form as:

$$\mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z}) = \mathbf{A}^{\mathsf{H}}\mathbf{Z} - \left(\frac{\|\mathbf{A}^{\mathsf{H}}\mathbf{Z}-\mathbf{Y}\|_{\mathrm{F}}-\varepsilon}{\|\mathbf{A}^{\mathsf{H}}\mathbf{Z}-\mathbf{Y}\|_{\mathrm{F}}}\right)_{+} \cdot \left(\mathbf{A}^{\mathsf{H}}\mathbf{Z} - \mathbf{Y}\right). \tag{5.3}$$

For the synthesis version, assuming $\mathbf{D}\mathbf{D}^{\mathsf{H}} = \mathbf{I}$, the generalized projection again reduces algebraically to the closed-form expression:

$$\mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z}) = \mathbf{Z} - \left(\frac{\|\mathbf{D}\mathbf{Z}-\mathbf{Y}\|_{\mathsf{F}}-\varepsilon}{\|\mathbf{D}\mathbf{Z}-\mathbf{Y}\|_{\mathsf{F}}}\right)_{+} \cdot \mathbf{D}^{\mathsf{H}}(\mathbf{D}\mathbf{Z}-\mathbf{Y}). \tag{5.4}$$

Hence, in both case, the cost of computing the generalized projection is dominated by matrix-vector products with $\mathbf{M}^{\mathsf{H}}$ or $\mathbf{D}^{\mathsf{H}}$and $\mathbf{D}$. When this can be done with fast transforms, both flavors (analysis *and* synthesis) have low complexity. More details on both projections (analysis and synthesis) are given in section A.1.

We are now ready to instantiate the general algorithm $\mathcal{G}$ in the different cases.

5.2.2 Plain sparse audio denoisers

We recall that as the algorithms are built to work on a frame based manner: in the plain (co)sparse cases, $\mathbf{Y} \in \mathbb{R}^{\mathsf{L}\times 1}$ is a vector. For both the analysis and the synthesis version, we instantiate the general algorithm $\mathcal{G}$ (chapter 3, Algorithm 1: page 27) with the choices summarized in Table 5.1.

Table 5.1 – Parameters of Algorithm 1 for the Plain Sparse Denoisers

Analysis	**Synthesis**
$\Theta = \{\mathbf{W} \mid \|\mathbf{W}-\mathbf{Y}\|_2 \leq \varepsilon\}$	$\Theta = \{\mathbf{W} \mid \|\mathbf{D}\mathbf{W}-\mathbf{Y}\|_2 \leq \varepsilon\}$
$\mathbf{M} = \mathbf{A} \in \mathbb{C}^{\mathsf{P}\times\mathsf{L}}, \mathsf{P} \geq \mathsf{L}$	$\mathbf{M} = \mathbf{I} \in \mathbb{C}^{\mathsf{L}\times\mathsf{L}}, \mathbf{D} \in \mathbb{C}^{\mathsf{L}\times\mathsf{S}}$,
$\mathcal{S}_\mu(\cdot) = \mathcal{H}_{\mathsf{P}-\mu}(\cdot)$	$\mathcal{S}_\mu(\cdot) = \mathcal{H}_{\mathsf{S}-\mu}(\cdot)$,
$\mu^{(0)} = \mathsf{P}-1$	$\mu^{(0)} = \mathsf{S}-1$
$F : \mu \mapsto \mu - 1$	$F : \mu \mapsto \mu - 1$
$\mathbf{Z}^{(0)} = \mathbf{A}\mathbf{Y}$	$\mathbf{Z}^{(0)} = \mathbf{D}^{\mathsf{H}}\mathbf{Y}$

The choice of function F and initialization $\mu^{(0)}$ means that we start with a small number $\mathsf{P} - \mu^{(0)} = 1$ (resp. $\mathsf{S} - \mu^{(0)} = 1$) of nonzero coefficients for the sparse constraint which we relax gradually as iterations progress.

The practical choice of the stopping parameter β is driven by a compromise between quality and computation time and we will specify the values used in the experimental section (section 5.3). We will also note from the experiments that the upper bound on the iteration count $\mathsf{i}_{\max}$ is never used as a stopping criterion. Even if this work does not provide any theoretical guarantees on convergence we observe empirically that the relative norm stopping criterion β is always used to terminate the algorithm.

Algorithm 1 with these parameters yields:

$$\hat{\mathbf{W}} := \mathcal{G}(\Theta, \mathbf{M}, \{\mathcal{S}_\mu(\cdot)\}_\mu, \mu^{(0)}, F, \mathbf{Z}^{(0)}, \beta, \mathsf{i}_{\max}).$$

For the analysis version $\hat{\mathbf{X}} := \hat{\mathbf{W}}$, while for the synthesis version $\hat{\mathbf{X}} := \mathbf{D}\hat{\mathbf{W}}$.

5.2.3 Social sparse audio denoisers

For the social sparse versions of the denoising method, we change the sparsifying operator from $\mathcal{H}_{\cdot-\mu}(\cdot)$ to $\mathcal{S}_\mu^{\mathrm{PEW}}(\cdot|\Gamma)$, as well as the update rule which becomes $F_\alpha : \mu \mapsto \alpha\mu$. The initial value $\mu^{(0)}$ may depend on the pattern Γ and will be specified in section 5.3. The resulting parameters are summarized in Table 5.2.

Table 5.2 – Parameters of Algorithm 1 for the Social Sparse Denoiser

Analysis	**Synthesis**
$\Theta = \{\mathbf{W} \mid \|\mathbf{W} - \mathbf{Y}\|_F \leq \varepsilon\}$	$\Theta = \{\mathbf{W} \mid \|\mathbf{DW} - \mathbf{Y}\|_F \leq \varepsilon\}$
$\mathbf{M} = \mathbf{A} \in \mathbb{C}^{P\times L}, P \geq L$	$\mathbf{M} = \mathbf{I} \in \mathbb{C}^{L\times L}$,
$\mathcal{S}_\mu(\cdot) = \mathcal{S}_\mu^{\mathrm{PEW}}(\cdot\|\Gamma)$	$\mathcal{S}_\mu(\cdot) = \mathcal{S}_\mu^{\mathrm{PEW}}(\cdot\|\Gamma)$,
$\mu^{(0)}$: see section 5.3	$\mu^{(0)}$: see section 5.3
$F = F_\alpha : \mu \mapsto \alpha\mu$	$F = F_\alpha : \mu \mapsto \alpha\mu$
$\mathbf{Z}^{(0)} = \mathbf{AY}$	$\mathbf{Z}^{(0)} = \mathbf{D}^H\mathbf{Y}$

A first version of the denoiser works with a *predefined* time-frequency pattern Γ and is compactly written as:

$$\begin{bmatrix}\hat{\mathbf{W}}(\Gamma)\\ \mu(\Gamma)\\ \mathbf{Z}(\Gamma)\end{bmatrix} := \mathcal{G}(\Theta, \mathbf{M}, \{\mathcal{S}_\mu^{\mathrm{PEW}}(\cdot|\Gamma)\}_\mu, \mu^{(0)}, F_\alpha, \mathbf{Z}^{(0)}, \beta, i_{\max}).$$

Choice of the time-frequency pattern A more adaptive denoiser uses this first version as a building brick to *select* the pattern Γ within a prescribed collection. Indeed, in order to get a fully adaptive denoising procedure, we design a method to automatically select the optimal Γ for the signal frames at stake. We call this step the "initialization loop". It consists in evaluating $\hat{\mathbf{W}}(\Gamma)$ with a small number of iterations (e.g. $i_{\max}^{\mathtt{small}} = 10$) for different patterns Γ.

Given a predefined set of time-frequency patterns $\{\Gamma_k\}_{k=1}^K$ and initial threshold values $\mu_k^{(0)}$ that will be specified in section 5.3, one can compute $\hat{\mathbf{W}}_k := \hat{\mathbf{W}}(\Gamma_k)$ for $1 \leq k \leq K$, and similarly $\mu_k := \mu(\Gamma_k)$ and $\mathbf{Z}_k := \mathbf{Z}(\Gamma_k)$. Then, the idea is that the best estimate $\hat{\mathbf{W}}_k$ should produce a residual with spectrum close to that of Additive White Gaussian Noise (AWGN), which is by definition flat. Thus, we select the pattern $\Gamma_{k^\star}$ yielding a residual with time-frequency representation of highest entropy.

For a given k, we can define the resulting time-frequency residual: $\mathbf{R}_k := \mathbf{M}\hat{\mathbf{W}}_k - \mathbf{Z}^{(0)}$. Computing a Q-bin histogram of the modulus of its entries yields $\hat{p}$, an empirical probability distribution, which (empirical) entropy is

$$e_k = -\sum_{q=1}^{Q} \hat{p}_q \log_2(\hat{p}_q). \tag{5.5}$$

A heuristic to choose Q is the Herbert-Sturges rule [Sturges 1926]

$$Q = \lfloor 1 + \log_2(\#\mathbf{R}_k) \rfloor, \tag{5.6}$$

where $\lfloor \cdot \rfloor$ is the floor function and $\#\mathbf{R}_k = L \times (2b+1)$ is the number of entries in the matrix $\#\mathbf{R}_k$. The values considered in the experiments of section 5.3 lead to $Q \in \{13, 15\}$.

Once the best pattern $\Gamma_{k^\star}$ is chosen as just described, we run Algorithm 1 with the parameters of Table 5.2 and warm-started $\mu^{(0)}$ and $\mathbf{Z}^{(0)}$, with a sufficiently large $i_{\max}$ (typically $i_{\max}^{\texttt{large}} = 10^6$) to get

$$\hat{\mathbf{W}} := \mathcal{G}(\Theta, \mathbf{M}, \{\mathcal{S}_\mu^{\mathrm{PEW}}(\cdot|\Gamma_{k^\star})\}_\mu, \mu_{k^\star}, F_\alpha, \mathbf{Z}_{k^\star}, \beta, i_{\max}^{\texttt{large}}).$$

The pseudo-code of the adaptive social denoiser for a given block of adjacent frames $\mathbf{Y} \in \mathbb{R}^{L\times(2b+1)}$ is given in Algorithm 2. Again, for the analysis version $\hat{\mathbf{X}} := \hat{\mathbf{W}}$, while for the synthesis version $\hat{\mathbf{X}} := \mathbf{D}\hat{\mathbf{W}}$.

Algorithm 2 Adaptive Social Sparse Denoisers

Require: $\mathbf{Y}$, ε, $\mathbf{A}$ or $\mathbf{D}$, $\{\Gamma_k\}_k$, $\{\mu_k^{(0)}\}_k$, α, β, $i_{\max}^{\texttt{small}}$, $i_{\max}^{\texttt{large}}$
set parameters from Table 5.2
for all k **do**

$$\begin{bmatrix} \hat{\mathbf{W}}_k \\ \mu_k \\ \mathbf{Z}_k \end{bmatrix} := \mathcal{G}(\Theta, \mathbf{M}, \{\mathcal{S}_\mu^{\mathrm{PEW}}(\cdot|\Gamma_k)\}_\mu, \mu_k^{(0)}, F_\alpha, \mathbf{Z}^{(0)}, \beta, i_{\max}^{\texttt{small}})$$

Compute e_k as in (5.5)
$k^\star := \operatorname{argmax}_k e_k$
$\hat{\mathbf{W}} := \mathcal{G}(\Theta, \mathbf{M}, \{\mathcal{S}_\mu^{\mathrm{PEW}}(\cdot|\Gamma_{k^\star})\}_\mu, \mu_{k^\star}, F_\alpha, \mathbf{Z}_{k^\star}, \beta, i_{\max}^{\texttt{large}})$.
return $\hat{\mathbf{W}}$

5.2.4 Post-processing and overlap-add synthesis

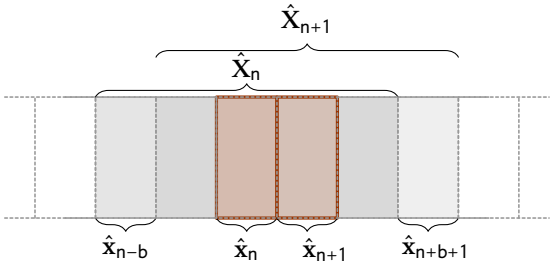


Figure 5.1 – Segment processing for frame n and frame $n+1$.

We recall that the denoiser is applied in a frame-based scenario. For clarity here, we index by n the previous variables $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ to account for the n^{th} frame of the underlying

signal being processed. Given noisy frame(s) $\mathbf{Y}_n$, the denoisers output estimated frame(s) $\hat{\mathbf{X}}_n$ which need to be transformed back into a full time-domain signal $\tilde{\mathbf{x}}$. For this, we first need to extract from $\hat{\mathbf{X}}_n$ a single estimated frame $\hat{\mathbf{x}}_n$:

- in the plain sparse case, this is straightforward as $\hat{\mathbf{X}}_n \in \mathbb{R}^{L\times 1}$ is already a vector;
- for the social sparse case, we set $\hat{\mathbf{x}}_n := \hat{\mathbf{X}}_n(:, b+1)$ to be the central column of the matrix $\hat{\mathbf{X}}_n$, see Figure 5.1.

Given the estimated frames $\{\hat{\mathbf{x}}_n\}_n$, and before the final overlap-add that will lead to the full time-domain estimate $\hat{\mathbf{x}}$, we perform a simple frequency-domain Wiener filtering on each $\hat{\mathbf{x}}_n$ similar to the one used in the Block-Thresholding algorithm [Yu *et al.* 2008] which we will use as a comparison in section 5.3. Such a Wiener filtering requires an estimation of the noise power σ^2, as well as an estimation of the signal power, both in the frequency domain. For the latter, we use the squared magnitudes of $\mathbf{A}\hat{\mathbf{x}}$ (resp. of $\mathbf{D}^H\hat{\mathbf{x}}$). Oracle values of σ^2 will be used in the experiments. Practically, we observed that this post-processing is useful at very low SNR (*i.e* 0 dB) where we observe "musical noise" effect.

Finally, overlap-add synthesis is performed, taking into account the windows that were applied onto the frames to get the noisy frames $\mathbf{Y}_n$.

5.3 Experiments

This section aims at comparing effects of the different shrinkages (plain or social), the different models (synthesis or analysis), and the degradation level on the audio denoising performance.

Compared methods We consider the plain sparse, plain cosparse, social sparse and social cosparse denoisers, as well as the state-of-the-art time-frequency block thresholding (BT) [Yu *et al.* 2008]. The main parameters are set as follows:

- frame size: $L = 64$ ms for music $L = 32$ ms for speech;
- Hamming windows, overlap: 75%;
- number of overlapping segments for social denoisers: $b = 5$ for music ($2b+1 = 11$ frames), $b = 1$ for speech ($2b+1 = 3$ frames);
- time-frequency patterns for social denoisers: $\{\Gamma_k\}_{k=1}^K$ presented on Figure 3.2 for music and Figure 3.3 for speech.
- stopping criteria $\beta = 10^{-3}$, $i_{\max}^{\texttt{small}} = 10$, $i_{\max}^{\texttt{large}} = 10^6$;

The time-frequency synthesis/analysis operators are:

- **Synthesis operator**: $\mathbf{D}$ is the inverse DFT of redundancy R, that is to say $\mathbf{D} \in \mathbb{C}^{L\times S}$ with $S = (R \times L)$ and $\mathbf{D}_{ls} := S^{-1/2} e^{j\frac{2\pi ls}{S}}$. One can check that $\mathbf{D}\mathbf{D}^H = \mathbf{I}$;

- **Analysis operator**: $\mathbf{A}$ is the forward DFT of redundancy R, that is to say $\mathbf{A} \in \mathbb{C}^{\mathsf{P}\times\mathsf{L}}$ with $\mathsf{P} = (\mathsf{R} \times \mathsf{L})$ and $\mathbf{A}_{\mathsf{pl}} := \mathsf{P}^{-1/2} e^{-j\frac{2\pi \mathsf{pl}}{\mathsf{P}}}$. Again, one can check that $\mathbf{A}^{\mathsf{H}}\mathbf{A} = \mathbf{I}$.

Practically, products with $\mathbf{A}$ (resp. $\mathbf{D}^{\mathsf{H}}$), are done using the FFT of size P (resp. S) on a zero-padded signal of initial length L. Similarly, products with $\mathbf{A}^{\mathsf{H}}$ (resp. $\mathbf{D}$), are done by truncating the inverse fast transform.

All denoisers require a parameter ε ruling the l_2 regularization for the denoising constraint. For the plain sparse denoisers, ε is set to $\sigma\sqrt{\sum_{\mathsf{j}=1}^{\mathsf{L}} \mathbf{w}_{\mathsf{j}}}$, with $\mathbf{w}_{\mathsf{j}}$ the j^{th} entry of the window $\mathbf{w}$ and σ^2 the known noise variance. For the adaptive social sparse denoisers, we scale ε to $(2\mathsf{b}+1)\sigma\sqrt{\sum_{\mathsf{j}=1}^{\mathsf{L}} \mathbf{w}_{\mathsf{j}}}$.

The adaptive social sparse denoisers also require to set $\mu_{\mathsf{k}}^{(0)}$ and α (see Algorithm 2). To adapt these parameters to the local peak audio level $\|\mathrm{vec}(\mathbf{Y})\|_\infty$ and to the number of active bins in the time-frequency pattern Γ_{k}, we set

$$\mu_{\mathsf{k}}^{(0)} := \|\Gamma_{\mathsf{k}}\|_0 \times \|\mathrm{vec}(\mathbf{Y})\|_\infty \tag{5.7}$$

$$\alpha := \min\left(\frac{\sigma}{\sqrt{\mathrm{var}(\mathrm{vec}(\mathbf{Y}))}}, 0.99\right), \tag{5.8}$$

where $\mathrm{vec}(\cdot)$ vectorizes the matrix. This parameterization reflects the "instantaneous" SNR in the region being processed. The two parameters α and μ rule how aggressively the sparse regularization is performed.

Influence of frequency transform redundancy Given the large combinatorics of experiments related to all possible configurations (plain/social, analysis/synthesis, redundancy factor) and noise levels, we performed a first pilot study for two input noise levels (5 dB, 20 dB). Each configuration was tested over the 10 SMALLbox music examples. The average SNR improvements, as well as the average computation times[1] (relative to the audio duration, the lower the better) are summarized in Table 5.3.

We observe that:

- For each noise level, each redundancy, and each thresholding operator, the performance of the analysis and synthesis models in decibels is almost identical, while the synthesis version is by 10% to nearly 40% faster that the analysis version.

- All other factors being equal, the computation time is roughly proportional to the redundancy R, while the SNR improvement is often very limited. In the rest of the experiments we thus choose $\mathsf{R} = 1$ and $\mathsf{R} = 2$, which seem to give the best compromise (and in fact, even the best performance in many configurations). Twice redundant transform also enables a transparent comparison with the baseline defined by block-thresholding ([Yu *et al.* 2008]).

[1] All reported computation times were measured here using a Matlab®implementation of the algorithms on a workstation equipped with a 2.4 Ghz Intel® Xeon®processor and 32 GB of RAM memory.

Table 5.3 – Processing times comparison (plain/social (co)sparse denoisers)

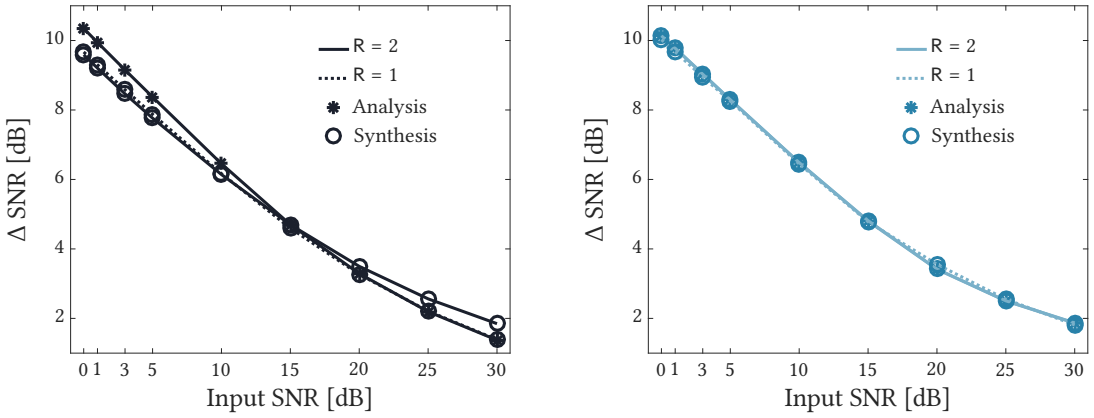
(a) Runtime performance (ratio to realtime processing)

	Input SNR: 5 dB				Input SNR: 20 dB			
	Analysis		Synthesis		Analysis		Synthesis	
	Plain	Social	Plain	Social	Plain	Social	Plain	Social
R = 1	8.7	19.9	2.9	14.2	17.0	8.5	7.1	7.1
R = 2	13.0	49.3	4.2	35.4	24.8	30.0	10.4	24.7
R = 4	19.5	123.5	6.3	88.7	35.0	105.8	15.1	83.4

(b) Corresponding improvements (ΔSNR)

	Input SNR: 5 dB				Input SNR: 20 dB			
	Analysis		Synthesis		Analysis		Synthesis	
	Plain	Social	Plain	Social	Plain	Social	Plain	Social
R = 1	7.88	8.26	7.88	8.26	3.26	3.56	3.26	3.56
R = 2	8.38	8.30	7.77	8.29	3.29	3.41	3.50	3.42
R = 4	7.92	7.88	7.32	7.87	3.11	3.35	3.31	3.36

Figure 5.2 and Figure 5.3 display averaged SNR improvements for the music and speech SMALL dataset examples for non redundant and twice redundant DFT over a wide range of input SNRs $\in \{0, 1, 3, 5, 10, 15, 20, 25, 30\}$.

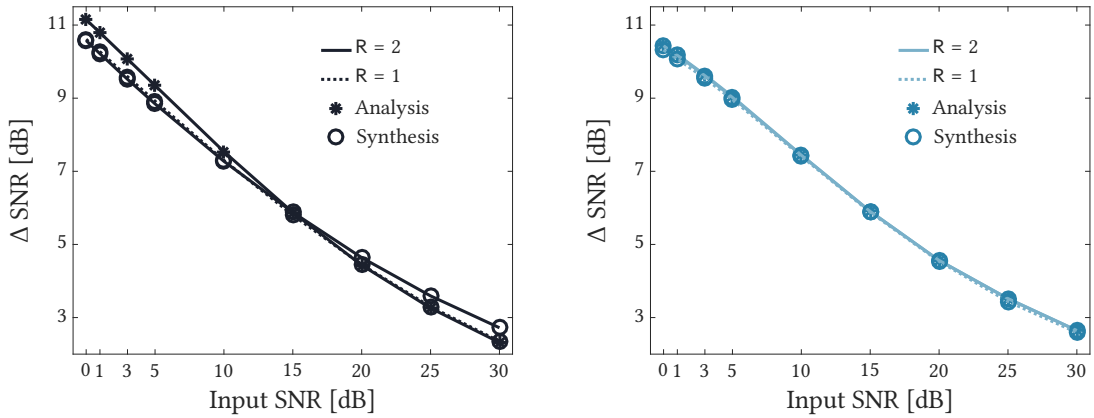


(a) Plain sparsity: ΔSNR

(b) Social sparsity: ΔSNR

Figure 5.2 – SMALL Music: Redundancy study (Denoising)

For the plain sparse models, Figure 5.2a and Figure 5.3a show benefits from twice redundant DFT as non redundant results are slightly outperformed by either the analysis method for low input SNR or synthesis method for high input SNR. For the social sparse models, Figure 5.2b and Figure 5.3b show similar SNR improvements and non-significant differences for all the tested methods. We verify these trends for both music and speech sounds.



(a) Plain sparsity: ΔSNR

(b) Social sparsity: ΔSNR

Figure 5.3 – SMALL Speech: Redundancy study (Denoising)

Large scale comparison of denoising performance Given the first pilot study, we now focus on the (co)sparse denoisers (plain and social) as well as block-thresholding, all with redundancy $\mathsf{R} = 2$. We consider nine input SNR levels in dB: {0, 1, 3, 5, 10, 15, 20, 25, 30} and work with the large scale speech and music datasets. Figure 5.4 shows averaged SNR improvements over each of the 5 music subsets as well as the TIMIT speech dataset.

Results on Figure 5.4 show that either the social (co)sparse or the plain (co)sparse algorithms outperform Block Thresholding (BT) on the denoising task for almost every category of audio content for mild to high input SNR. At low input SNRs, synthesis sparse flavor seems to perform worse than the other methods (cosparse and social (co)sparse). At low SNR, the methods derived from the algorithmic framework show on par to slightly better results compared to BT. The difference between our approaches and BT increases with the input SNR. We note a significant contrast at high SNR where BT underperforms by more than 6 dB in the less favorable configuration. This might be because BT strongly relies on the noise model whereas (co)sparse and social (co)sparse methods try to emphasize the signal itself.

We gathered standard deviation informations associated to Figure 5.4 and results demonstrate that the plain cosparse denoiser produces less variable results as the standard deviation is the lowest for this technique in 80% of the tested cases. We also notice that, without considering any specific algorithm, the improvement variability seems to increase with the input SNR. Indeed, for light noise conditions, the standard deviation reaches up to 3.29 dB for BT on the RWC “Chamber” musical excerpts.

Figure 5.5 shows averaged Δ STOI/PESQ/PEAQ performance over each of the 5 music subsets as well as the TIMIT speech dataset. Even if Figure 5.4d and Figure 5.4f do not show clear superiority of one or another method on SNR improvement for voice based audio content, Figure 5.5b reveals improved objective speech quality (PESQ metric) for both social and plain (co)sparse denoisers. More specifically, the social versions seems to bring substantial speech quality improvement compared to the BT baseline or sim-

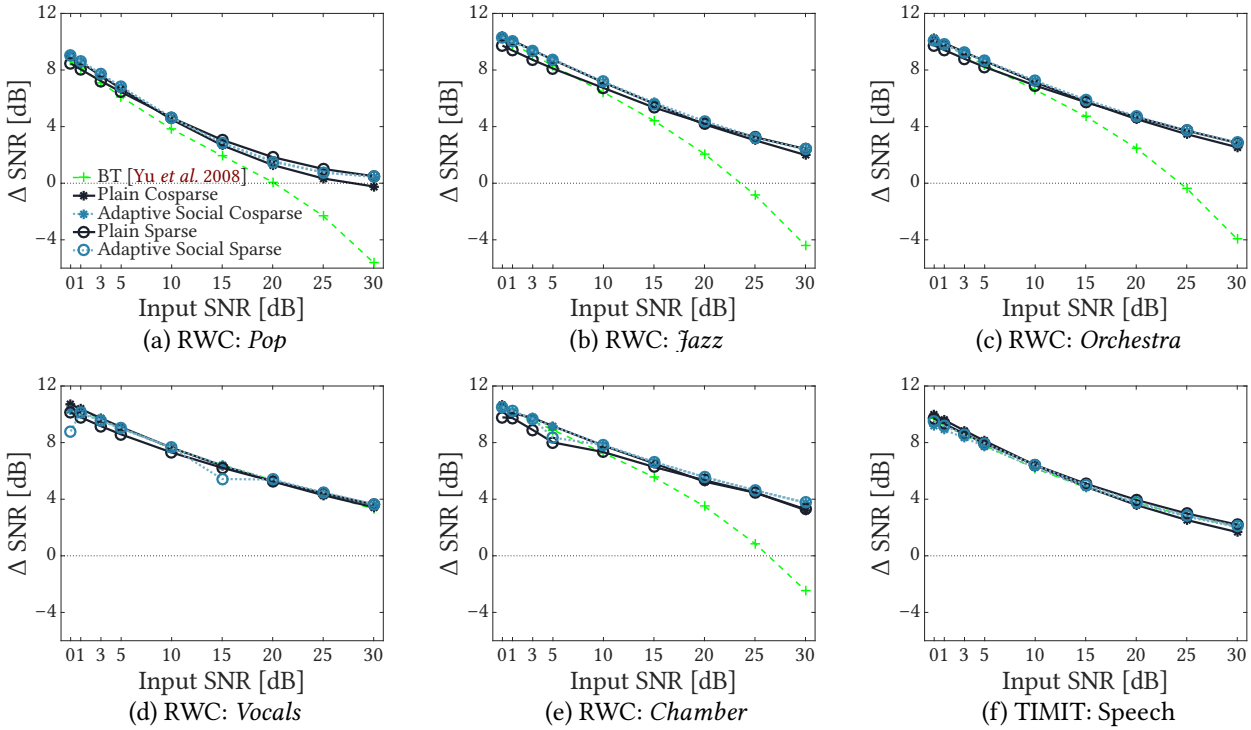
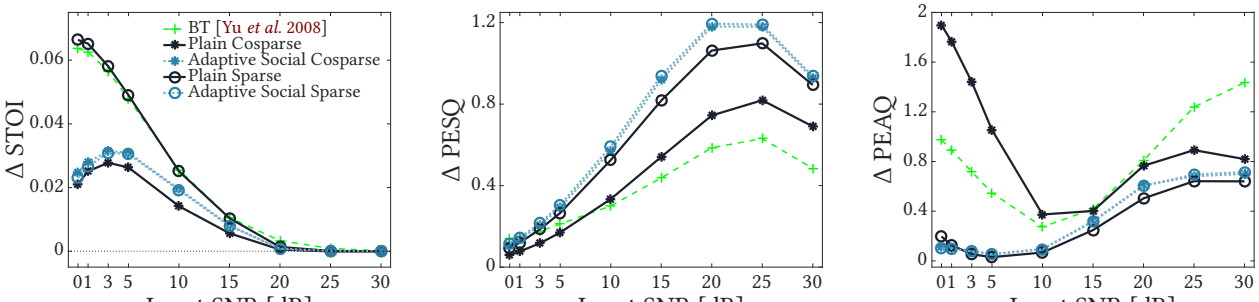


Figure 5.4 – Denoising: Numerical Results Δ SNR [dB]

ple sparse denoisers. The effect is clearer for input SNR above 5 dB. On the contrary, only the plain synthesis sparse method seems to be on par with BT for intelligibility improvements at low SNR (Figure 5.5a). For the musical content, we see on Figure 5.5c that contrarily to ΔSNR measurements, all methods provide quality improvements according to the PEAQ descriptor. Indeed, we note for all methods and all configuration a positive ΔPEAQ. Results on global audio quality are here quite different than the ones specific to speech quality as the plain cosparse method seems to provide better improvements for the worse cases (low input SNRs).

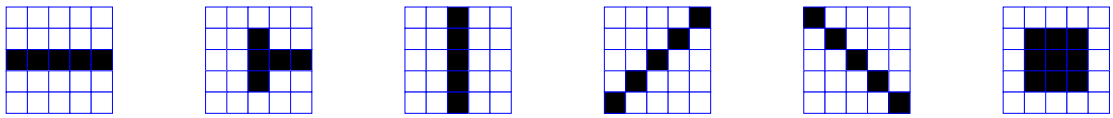
Time-frequency neighborhood selection Before comparing computational efficiency, we focus here on the time-frequency neighborhood selection step of the adaptive social (co)sparse methods. Figure 5.8 and Figure 5.9 display the time-frequency neighborhood distribution from all the denoised frames of the SMALL dataset examples (speech & music). These represent how often a time-frequency pattern Γ_k was selected as a $\Gamma_{k^\star}$ after the selection step of the adaptive social (co)sparse denoising methods. These results are obtained with the same algorithmic parameters as the previous large scale study. We recall for the reader convenience the available time-frequency neighborhoods collections on Figure 5.6 and Figure 5.7. What is interesting to see is that whatever prior (analysis or synthesis) is chosen, the neighborhood distributions are quite similar but can vary with the input SNR. This trend is particularly verified for music. Thus, at low input SNR for music, we remark that the first five neighborhood are roughly selected with the same frequency and the default pattern (Γ_6) is less frequently chosen. On the contrary, as the input SNR increases, Γ_6 is more and more selected. For speech, the neighborhood distri-



(a) TIMIT: Δ STOI (b) TIMIT: Δ PESQ (c) RWC: Δ PEAQ

Figure 5.5 – Denoising: Objective Quality and Intelligibility Results

butions are more similar from one input SNRs to another. We see that the selection step outputs frequently the patterns emphasizing tonal (Γ_1) and transient (Γ_3) content.



(a) Γ_1 (b) Γ_2 (c) Γ_3 (d) Γ_4 (e) Γ_5 (f) Γ_6

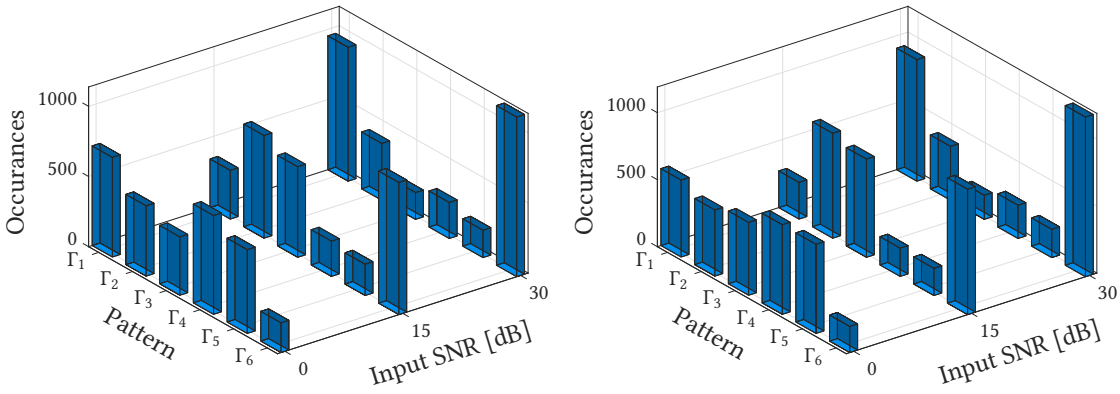
Figure 5.6 – Extended set of time-frequency neighborhoods used for music

(a) Γ_1 (b) Γ_2 (c) Γ_3 (d) Γ_4 (e) Γ_5 (f) Γ_6

Figure 5.7 – Extended set of time-frequency neighborhoods used for speech

Computation time For the social case, the computational cost is driven by the shrinkage (PEW) and the projection steps. However, evaluating PEW shrinkage is relatively fast, as it can be computed through 2-D convolution in the time-frequency domain. Besides, since we set low $\mathsf{i}_{\max}$ for the initialization loop, the choice of Γ is quite fast and adds only $(\mathsf{b}-1) \times \mathsf{i}_{\max}^{\mathtt{small}}$ iterations compared to the case where only one time-frequency pattern is considered. These properties allow to expect the social cosparse denoiser to have runtime comparable to that of the plain cosparse denoiser.

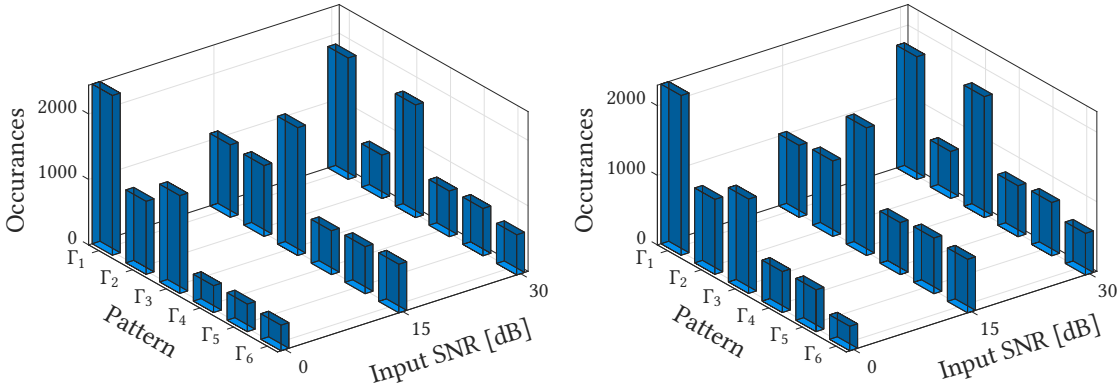
Table 5.4 displays processing times relative to real-time processing for all denoising procedures. These computational comparisons are conducted and averaged on the SMALLbox music and speech examples. The SNR improvements are in line with what was previously observed on larger datasets with very similar performance of all methods. However we note very different behaviors between the plain/social (co)sparse denoisers



(a) Adaptive social cosparse

(b) Adaptive social sparse

Figure 5.8 – SMALL Music: Time-frequency pattern distribution (Denoising)



(a) Adaptive social cosparse

(b) Adaptive social sparse

Figure 5.9 – SMALL Speech: Time-frequency pattern distribution (Denoising)

in terms of runtime. While the plain analysis and synthesis flavors are fastest at low SNR, the social versions are fastest at high SNR. This suggests that in practice the choice of one of these methods might be rather driven by speed considerations or quality ratings (Figure 5.5) than by SNR improvement performance (the plain sparse denoiser being twice to more than 30 times faster than the other methods in the more advantageous case). As the DFT can be efficiently implemented with a fast transform, the computational cost of the denoising procedures mainly stems from the sparsifying step and the projection on the denoising constraint. The different computational properties emphasized in Table 5.4 come from the behavior of the sparsifying operator (shrinkage) when used inside the ADMM framework and the total number of iterations needed to finish or converge. To give some insights on this last parameter, Figure 5.10 gives the distribution of total number of iterations needed to finish as a function of the input SNR for every method. It shows how often an iteration count is needed for the denoising procedures

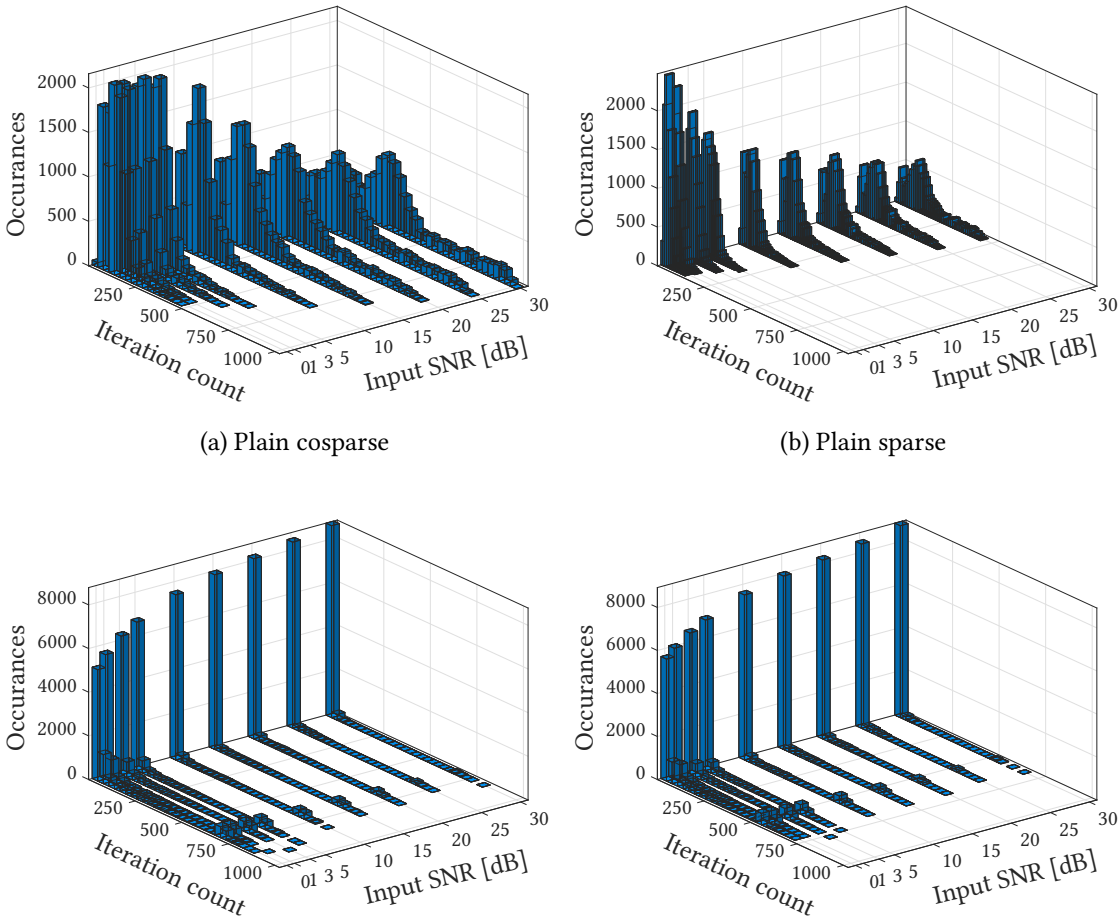
Table 5.4 – Computational performance of (plain/social) (co)sparse denoisers

Input SNR [dB]	Plain cosparse		Adaptive social cosparse		Plain sparse		Adaptive social sparse	
	Δ SNR	x RT	Δ SNR	x RT	Δ SNR	x RT	Δ SNR	x RT
0	10.75	3.3	10.29	34.2	10.08	1.1	10.30	29.2
1	10.37	3.4	9.99	32.8	9.71	1.1	9.97	28.0
3	9.61	3.6	9.34	30.2	8.99	1.2	9.32	26.0
5	8.86	3.7	8.67	28.2	8.31	1.4	8.66	24.2
10	7.00	4.4	6.97	23.7	6.71	1.7	6.96	20.5
15	5.29	5.3	5.35	19.5	5.29	2.1	5.36	17.0
20	3.86	6.5	3.99	15.2	4.07	2.7	4.00	13.5
25	2.73	7.9	3.01	11.1	3.08	3.3	3.01	10.1
30	1.85	9.4	2.25	8.8	2.28	4.0	2.26	8.2

to stop. These results are obtained from the same tested material as Table 5.4, *i.e.* all the denoised frames of the examples in the SMALLbox dataset (speech and music). It is interesting to link those iteration counts distributions with the runtime results of Table 5.4. We clearly see the computational advantage of the plain sparse method as it finishes in less than 500 iterations for all the tested configurations. For the plain cosparse method a higher number of iterations needed for high input SNRs support the runtime results. For the adaptive social methods we note that in most of the cases, the algorithms terminate in less than 50 iterations. Finally, one essential result to highlight is that for every single tested configuration the algorithms stop way below $i_{max} = 10^6$ iterations. This means that for these examples, whatever flavor of the algorithmic framework is used, it terminates thanks to the relative stopping criterion β and not the a predefined bound i_{max}.

5.4 Summary

This chapter presented several instances of the common algorithmic framework to address the denoising problem. In terms of reconstruction quality, our detailed large scale study shows that consistent SNR improvements are observed. They are either on par with or better than what the widely used Block Thresholding (BT) reference algorithm can achieve, especially for input SNRs above 10dB. One of the possible explanation of the better results obtained with our methods compared to BT could be that in this work we focus on the signal model rather than on local time-frequency SNR. Different trends are observed with perceptually-aware objective quality measures, for speech specific content, adaptive social (co)sparse methods seems to yield better results. On music examples, the plain cosparse version appears to be more suitable for audio quality (especially for high degradation). However further work would be needed to confirm this on subjective listening tests. The list below gives some guidelines for denoising audio with the methods presented in this chapter.



(a) Plain cosparse

(b) Plain sparse

(c) Adaptive social cosparse

(d) Adaptive social sparse

Figure 5.10 – SMALL: Total iteration count distribution (Denoising)

Denoising guidelines

- Music:
  - Low SNR
    If all methods seems to give similar results for SNR improvements, one should prefer the **plain cosparse** method for audio quality.
  - High SNR
    SNR improvements results seems in favor of **adaptive social (co)sparse** and **plain sparse** methods, however for quality plain cosparse seems preferable.
- Speech:
  - Low SNR
    At low SNR, on should prefer **adaptive social (co)sparse** and **plain sparse** methods, especially for quality.

- High SNR
  For high SNR, the best option appears to be the **adaptive social sparse** for quality. If the target is intelligibility improvements, the choice should tend to **plain sparse**.

Finally, we mention that if *speed* is the major concern, the **plain sparse** method should be the choice.

Chapter 6

Declipping

Contents

This chapter focuses on the audio reconstruction problem specific to magnitude saturation. After briefly describing the clipping (*i.e.* saturation) issue, the chapter will feature a review of available desaturation techniques. Then, one will find applications of structured (co)sparsity first for a monochannel reconstruction scenario emphasizing an adaptive time-frequency modeling, second with channel-wise structured (co)sparse modeling for multichannel purposes. Each of these two applications is an instance of the generic framework introduced earlier (chapter 3). To conclude this current chapter, some experiments including comparisons with state-of-the-art declipping methods on both single and multichannel will be detailed.

From section 6.2 this chapter is inspired from [Gaultier *et al.* 2017a]: Clément Gaultier, Nancy Bertin, Srđan Kitić and Rémi Gribonval. *A modeling and algorithmic framework for (non) social (co) sparse audio restoration.* arXiv preprint arXiv:1711.11259, 2017. However, it presents new experimental results (in section 6.4). Multichannel instances of the algorithmic framework are inspired from [Gaultier *et al.* 2018]: Clément Gaultier, Nancy Bertin and Rémi Gribonval. *CASCADE: Channel-Aware Structured Cosparse Audio DEclipper.* In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 571–575, April 2018.

6.1 The saturation and desaturation problems

This section first presents the degradation itself and its consequences on audio quality. Then, some details are given on the way to quantitatively describe the amount of clipping before reviewing available reconstruction methods to address this problem.

6.1.1 The saturation problem on audio recordings

Clipping, also known as saturation, is a common phenomenon that can arise from hardware or software limitations in any audio acquisition pipeline. It results in severely distorted audio recordings. Magnitude saturation can occur at different steps in the acquisition, reproduction or analog-to-digital conversion process. Restoring a saturated signal is of great interest for many applications in digital communications, image processing or audio. In the latter, while light to moderate clipping cause only some audible clicks and pops, more severe saturation highly affects original signals which sound contaminated by rattle noise. The perceived degradation depends on the clipping level and the original signal and can lead to significant loss in perceived audio quality [Tan *et al.* 2003]. More recently, studies [Tachioka *et al.* 2014, Harvilla & Stern 2014] also showed the negative impact of clipped signals when used in signal-processing pipelines for recognition, transcription or classification applications. In this chapter, we use the idealized hard-clipping model below. Although simple, it correctly approximates the magnitude saturation and allows to easily identify the clipped and reliable samples. Let one define $\mathbf{x} \in \mathbb{R}^L$ a clean original discrete signal. The saturated version $\mathbf{y} \in \mathbb{R}^L$ is obtained with the following hard-clipping degradation:

$$\mathbf{y}_i = \begin{cases} \mathbf{x}_i & \text{when } |\mathbf{x}_i| \leq \tau; \\ \operatorname{sgn}(\mathbf{x}_i)\tau & \text{otherwise;} \end{cases} \tag{6.1}$$

with $\mathbf{y}_i$ (resp. $\mathbf{x}_i$) a sample from $\mathbf{y}$ (resp. $\mathbf{x}$) and τ the hard-clipping level. A visual example of such a degradation is given on Figure 6.3. In real settings where softer saturation occurs, this model can be enforced with appropriate data pre-processing.
Most of the perceived degradation is due to additional harmonics introduced by the local non-linear periodic discontinuities during the saturation process. This effect is sometimes called *harmonic distortion*. Figure 6.2 illustrates such an effect on a simple sine

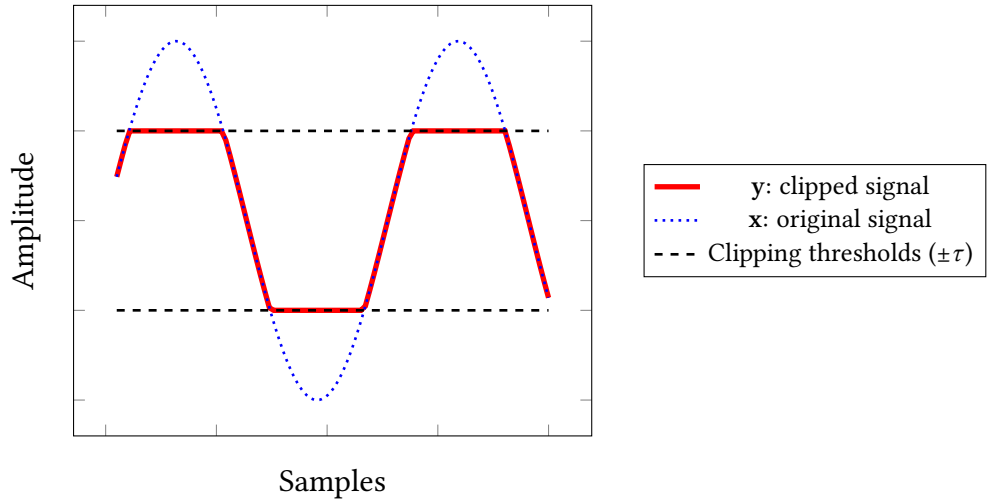


Figure 6.1 – Hard-clipping model (Equation (6.1))

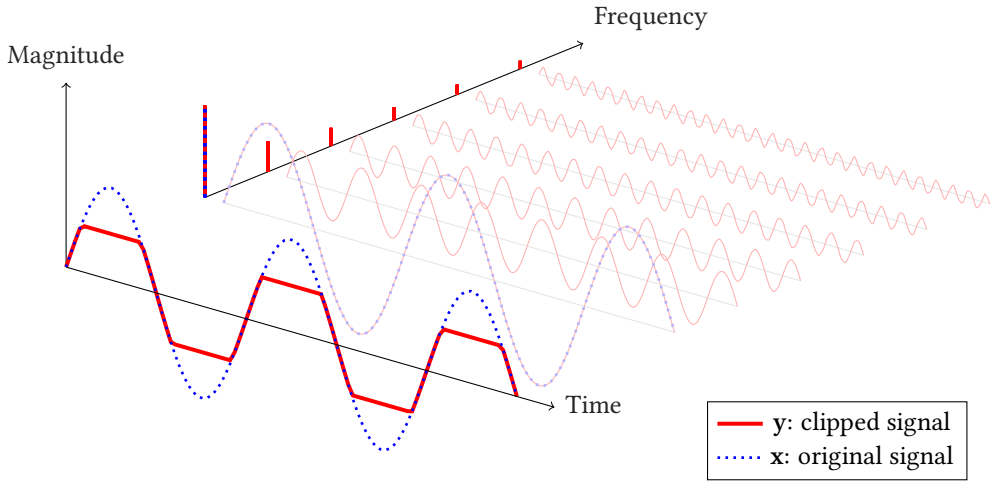
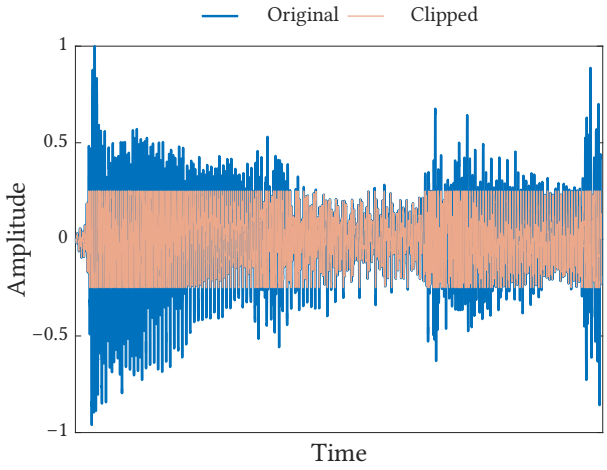
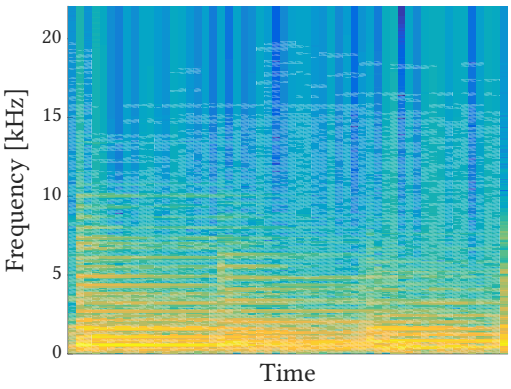


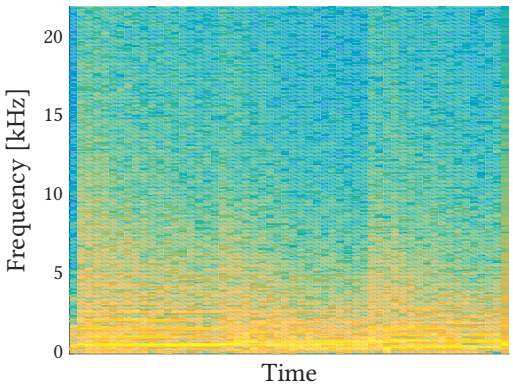
Figure 6.2 – Harmonic distortion



(a) Signal waveforms



(b) Frequency transform (original signal)



(c) Frequency transform (clipped signal)

Figure 6.3 – Clipped signal example

signal by showing the corresponding truncated frequency decomposition. We see that the original and clipped signals share only the initial low frequency component whereas all the additional harmonics belongs to the clipped one. While harmonic distortion could be used on purpose to "enrich" the frequency spectrum for some audio engineering application, most of the time it produces undesired artifacts. Figure 6.3 shows the effect of clipping on a more complex musical signal. It not only shows this effect on the time-domain waveform (Figure 6.3a) but also the completely modified frequency content (Figure 6.3c) compared to the original one (Figure 6.3b). One guess is that this frequency content difference can be even emphasized for voice or music signals that are usually sparse in frequency.

6.1.2 Prior art on audio declipping

While we can trace back some attempts to address this issue, *e.g.* with autoregressive models [Janssen *et al.* 1986], to several decades, significant progress towards efficient desaturation was recently made in several directions.

Declipping as a linear inverse problem The declipping problem was recast as an *undetermined, linear inverse problem*, akin to *inpainting*, which could be addressed by means of a *sparse* regularization [Adler *et al.* 2012]. This work relies on a two stage algorithm based on OMP. It first estimates the active atoms of the sparse representation on the reliable samples of the signal. Then, it imposes a declipping constraint in the transform domain so the name *Constrained Orthogonal Matching Pursuit*.

Consistency constraint On this basis, algorithmic frameworks evolved from usual greedy algorithms to thresholding [Kitić *et al.* 2013] approaches. For the latter, an additional penalty is extended to every iteration namely *clipping consistency* (already present as a final step in [Adler *et al.* 2012]). This enforces the reliable parts of the initial clipped signal and the reconstructed one to match. This was shown to drastically improve reconstruction performance and to the current knowledge, state-of-the-art declipping methods are embedding clipping consistency constraint during iterations.

Time-frequency models In parallel, a shift from a (now) traditional *sparse synthesis* approach, to a *sparse analysis* was proposed [Kitić *et al.* 2015], as well as some model refinements exploiting notions of *structured sparsity*, especially that of *social sparsity* [Kowalski *et al.* 2013] in the time-frequency domain [Siedenburg *et al.* 2014]. Contrarily to [Adler *et al.* 2012, Kitić *et al.* 2013, Kitić *et al.* 2015], which rely on frame-based processing and overlap-add for reconstruction, this last line of work processes complete recordings to find relevant clusters in the time-frequency plane.

These layers led to significant improvements in reconstruction accuracy and computational efficiency thanks to cosparsity. Comparison of cosparsity *v.s.* sparsity will be thoroughly investigated later in this chapter. Since declipping was remodeled as an inverse problem, greedy heuristics and non-convex approaches were shown to perform the best for signal recovery in [Kitić *et al.* 2013, Siedenburg *et al.* 2014, Kitić *et al.* 2015].

Convex methods Lines of work involving convex optimization were investigated [Defraene *et al.* 2013]. This method uses ℓ_1 norm minimization and a perceptually oriented sparse representation to perform the reconstruction. More recently in [Ávila *et al.* 2017], one method based on linearly or quadratically constrained weighted least-squares was introduced to tackle a relaxed version of declipping which is compression (soft-clipping) compensation. Notably, another earlier attempt to alleviate clipping with ℓ_2 regularization for automated speech recognition [Harvilla & Stern 2014].

It must be noted that all these methods were developed and tested for *single-channel* signals, while multichannel data now represent a large part of available audio content, from stereo to more and more channels. To date, the multichannel joint declipping problem has only been addressed by [Ozerov *et al.* 2016] through a modeling of the signals as mixtures of sound sources, in order to encompass inter-channel correlations. This approach requires prior knowledge or estimation of the number of audio sources.

After this short review of available declipping methods in the literature[1], the next sections presents how taking the best of (co)sparsity and structured sparse thresholding operators can lead to effective reconstruction and speed improvements for the declipping tasks.

6.2 (A)social sparse declippers

In the following section we introduce several declipping methods derived from the algorithmic framework presented in chapter 3. These methods will embody regular or structured (co)sparse data models. After listing the required projection operators, we instantiate the different versions of Algorithm 1. As a reminder, we consider the matrix $\mathbf{Y} \in \mathbb{R}^{L\times(2b+1)}$ containing one or more windowed frames of L samples from the observed signal $\tilde{\mathbf{y}}$ $(2b+1 \geq 1)$. The declipping problem is to estimate the original clean signal frames, similarly gathered in a matrix $\mathbf{X}$ of the same size.

The clipping degradation identifies to the hard-clipping model (Equation (6.1)) recalled below.

$$\mathbf{Y}_{ij} = \begin{cases} \mathbf{X}_{ij} & \text{for } |\mathbf{X}_{ij}| \leq \tau; \\ \operatorname{sgn}(\mathbf{X}_{ij})\tau & \text{otherwise;} \end{cases}$$

[1] State-of-the-art results can be appreciated for instance from the SPADE software webpage: https://spade.inria.fr/
and https://homepage.univie.ac.at/monika.doerfler/StrucAudio.html

with $\mathbf{Y}_{ij}$ (resp. $\mathbf{X}_{ij}$) a sample from $\mathbf{Y}$ (resp. $\mathbf{X}$) and τ the hard-clipping level. Here i is the index in a frame and j is the index of the frame.

6.2.1 Generalized projections for the declipping problem

Denote Ω_+ (resp. Ω_-) the collection of indices ij of the samples in matrix $\mathbf{Y}$ affected by positive (resp. negative) magnitude clipping. Similarly denote Ω_r the indices of the reliable samples (not affected by clipping), and for any of these sets Ω define $\mathbf{V}_\Omega$ the matrix formed by keeping only the entries of $\mathbf{V}$ indexed by Ω and setting the rest to zero.

The data-fidelity constraint can now be expressed for the analysis setting with $\mathbf{M} := \mathbf{A}$ by

$$\Theta := \left\{ \mathbf{W} \mid \begin{array}{r} \mathbf{W}_{\Omega_r} = \mathbf{Y}_{\Omega_r}; \\ \mathbf{W}_{\Omega_+} \succcurlyeq \mathbf{Y}_{\Omega_+}; \\ \mathbf{W}_{\Omega_-} \preccurlyeq \mathbf{Y}_{\Omega_-}. \end{array} \right\}$$

where $\mathbf{W}$ is a time-domain estimate of the same size as $\mathbf{Y}$. For the synthesis setting, with $\mathbf{M} := \mathbf{I}$, we set

$$\Theta := \left\{ \mathbf{W} \mid \begin{array}{r} (\mathbf{DW})_{\Omega_r} = \mathbf{Y}_{\Omega_r}; \\ (\mathbf{DW})_{\Omega_+} \succcurlyeq \mathbf{Y}_{\Omega_+}; \\ (\mathbf{DW})_{\Omega_-} \preccurlyeq \mathbf{Y}_{\Omega_-}. \end{array} \right\}.$$

Here $\mathbf{W}$ will be a time-/channel- frequency estimate gathering as many frames as in $\mathbf{Y}$ and S frequency points. Similarly to the denoising use-case, these choices hold for both plain and structured versions.

In the analysis setting, the desired projection reduces to component wise magnitude constraints (see. section A.2) and can be expressed as:

$$[\mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z})]_{ij} = \begin{cases} \mathbf{Y}_{ij} & \text{if } ij \in \Omega_r; \\ (\mathbf{M}^H\mathbf{Z})_{ij} & \text{if } \begin{cases} ij \in \Omega_+, (\mathbf{M}^H\mathbf{Z})_{ij} \geq \tau; \\ \text{or} \\ ij \in \Omega_-, (\mathbf{M}^H\mathbf{Z})_{ij} \leq -\tau; \end{cases} \\ \operatorname{sgn}(\mathbf{Y}_{ij})\tau & \text{otherwise.} \end{cases}$$

In this case, matrix-vector products with $\mathbf{M}^H$ dominates the computing cost of the generalized projection. When this can be done with a fast transform, the analysis flavor has low complexity.

For the synthesis case, the projection step was initially approximated with a nested iterative procedure as explained in [Kitić 2015]. Even if it can help building an efficient algorithm for the projection, the overall computation cost for the synthesis flavor in that case however remains substantially higher than the analysis version (making it almost intractable). However, very recent work ([Záviška *et al.* 2018]) derived a closed-form solution for the declipping projection in the synthesis case involving a Parseval tight frame for $\mathbf{D}$.

This projection for the synthesis version boils down to:

$$\mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z}) = \mathbf{Z} - \mathbf{D}^{\mathsf{H}}(\mathbf{D}\mathbf{Z} - \Pi_{\Theta,\mathbf{M}}(\mathbf{Z})), \tag{6.2}$$

with

$$[\Pi_{\Theta,\mathbf{M}}(\mathbf{Z})]_{ij} = \begin{cases} \mathbf{Y}_{ij} & \text{if } ij \in \Omega_r; \\ (\mathbf{D}\mathbf{Z})_{ij} & \text{if } \begin{cases} ij \in \Omega_+, (\mathbf{D}\mathbf{Z})_{ij} \geq \tau; \\ \text{or} \\ ij \in \Omega_-, (\mathbf{D}\mathbf{Z})_{ij} \leq -\tau; \end{cases} \\ \operatorname{sgn}(\mathbf{Y}_{ij})\tau & \text{otherwise.} \end{cases}$$

When products with $\mathbf{D}$ respectively $\mathbf{D}^{\mathsf{H}}$ can be achieved with fast transforms, this also conveys low complexity to the synthesis flavor projection. More details on both projections are given in Appendix A section A.2.

With all the steps defined, we can now instantiate the general algorithm $\mathcal{G}$ in the different cases.

6.2.2 Plain sparse audio declippers

We recall that as the algorithms are built to work on a frame based manner: in the plain (co)sparse cases, $\mathbf{Y} \in \mathbb{R}^{L\times 1}$ is a vector. Similarly to denoising, for both the analysis and the synthesis version, we instantiate the general algorithm $\mathcal{G}$ (chapter 3, Algorithm 1: page 27) by choosing the operators described in Table 6.1.

The update rule F for μ is set to gradually decrease μ by 1 at each iteration, starting from $\mu^{(0)} = \mathsf{P} - 1$ for the analysis case (resp. $\mu^{(0)} = \mathsf{S} - 1$ for the synthesis case). This way, we relax the sparse constraint the same way we do it for denoising (chapter 5 chapter 5).

Table 6.1 – Parameters of Algorithm 1 for the Plain Sparse Declipper

Analysis	**Synthesis**
$\Theta = \left\{ \mathbf{W} \mid \begin{array}{c} \mathbf{W}_{\Omega_r} = \mathbf{Y}_{\Omega_r}; \\ \mathbf{W}_{\Omega_+} \geqslant \mathbf{Y}_{\Omega_+}; \\ \mathbf{W}_{\Omega_-} \leqslant \mathbf{Y}_{\Omega_-}. \end{array} \right\}$	$\Theta = \left\{ \mathbf{W} \mid \begin{array}{c} (\mathbf{D}\mathbf{W})_{\Omega_r} = \mathbf{Y}_{\Omega_r}; \\ (\mathbf{D}\mathbf{W})_{\Omega_+} \geqslant \mathbf{Y}_{\Omega_+}; \\ (\mathbf{D}\mathbf{W})_{\Omega_-} \leqslant \mathbf{Y}_{\Omega_-}. \end{array} \right\}$
$\mathbf{M} = \mathbf{A} \in \mathbb{C}^{\mathsf{P}\times\mathsf{L}}, \mathsf{P} \geq \mathsf{L}$	$\mathbf{M} = \mathbf{I} \in \mathbb{C}^{\mathsf{L}\times\mathsf{L}}, \mathbf{D} \in \mathbb{C}^{\mathsf{L}\times\mathsf{S}},$
$\mathcal{S}_\mu(\cdot) = \mathcal{H}_{\mathsf{P}-\mu}(\cdot)$	$\mathcal{S}_\mu(\cdot) = \mathcal{H}_{\mathsf{S}-\mu}(\cdot),$
$\mu^{(0)} = \mathsf{P} - 1$	$\mu^{(0)} = \mathsf{S} - 1$
$F : \mu \mapsto \mu - 1$	$F : \mu \mapsto \mu - 1$
$\mathbf{Z}^{(0)} = \mathbf{A}\mathbf{Y}$	$\mathbf{Z}^{(0)} = \mathbf{D}^{\mathsf{H}}\mathbf{Y}$

Iterating Algorithm 1 with the parameters described above gives a declipped estimate $\hat{\mathbf{W}}$ such that:

$$\hat{\mathbf{W}} := \mathcal{G}(\Theta, \mathbf{M}, \{\mathcal{S}_\mu(\cdot)\}_\mu, \mu^{(0)}, F, \mathbf{Z}^{(0)}, \beta, \mathsf{i}_{\max}).$$

We recall that for the analysis version $\hat{\mathbf{X}} := \hat{\mathbf{W}}$, while for the synthesis version $\hat{\mathbf{X}} := \mathbf{D}\hat{\mathbf{W}}$.

Notably, this declipping method using plain sparse modeling embedded in the algorithmic framework is a slightly modified version of the SPADE algorithms ([Kitić *et al.* 2015]). The two main differences come from the stopping criterion and the way the projection to the declipping constraint is computed for the synthesis case. For the SPADE algorithms the stopping criterion is evaluated using the absolute euclidean difference between two successive time-frequency estimates. In our case, we use a relative stopping criterion as described in Algorithm 1 making this method probably less sensitive to the intrinsic dimension of the matrices $(\mathbf{X}, \mathbf{W}, \mathbf{Z})$. The projection on the declipping constraint for synthesis-SPADE is approximated iteratively while in this work we use the closed-form solution presented earlier (Equation (6.2)) which requires a Parseval tight-frame assumption on the dictionary. In the experimental section, we will see that these modifications of the objective function used for the stopping criterion and the projection for the synthesis case brings some benefits.

6.2.3 Social sparse audio declippers

We recall that the algorithms are built to work on a frame based manner: for the social (co)sparse cases, we set $\mathbf{x} = \mathbf{X}(:, b+1)$ to be the central column of the matrix $\mathbf{X} \in \mathbb{R}^{L\times(2b+1)}$, see Figure 5.1. Similarly to the social sparse audio denoising procedure (chapter 5 page 43), we change the sparsifying operator to $\mathcal{S}_\mu^{\mathrm{PEW}}(\cdot \mid \Gamma)$ (Equation (3.17) page 28) and the update rule which we set now to $F_\alpha : \mu \mapsto \alpha\mu$. Here μ plays a different role compared to the plain sparse declipper. Indeed, μ does not directly tune a sparsity level but an energy. The initial value $\mu^{(0)}$ may also depend here on the pattern Γ and will be precised in section 6.4.

The resulting parameters are summarized in Table 6.2.

Table 6.2 – Parameters of Algorithm 1 for the Social Sparse Declipper

Analysis	**Synthesis**
$\Theta = \left\{ \mathbf{W} \mid \begin{array}{c} \mathbf{W}_{\Omega_r} = \mathbf{Y}_{\Omega_r}; \\ \mathbf{W}_{\Omega_+} \succcurlyeq \mathbf{Y}_{\Omega_+}; \\ \mathbf{W}_{\Omega_-} \preccurlyeq \mathbf{Y}_{\Omega_-}. \end{array} \right\}$	$\Theta = \left\{ \mathbf{W} \mid \begin{array}{c} (\mathbf{DW})_{\Omega_r} = \mathbf{Y}_{\Omega_r}; \\ (\mathbf{DW})_{\Omega_+} \succcurlyeq \mathbf{Y}_{\Omega_+}; \\ (\mathbf{DW})_{\Omega_-} \preccurlyeq \mathbf{Y}_{\Omega_-}. \end{array} \right\}$
$\mathbf{M} = \mathbf{A} \in \mathbb{C}^{P\times L}, P \geq L$	$\mathbf{M} = \mathbf{I} \in \mathbb{C}^{L\times L}, \mathbf{D} \in \mathbb{C}^{L\times S}$,
$\mathcal{S}_\mu(\cdot) = \mathcal{S}_\mu^{\mathrm{PEW}}(\cdot\vert\Gamma)$	$\mathcal{S}_\mu(\cdot) = \mathcal{S}_\mu^{\mathrm{PEW}}(\cdot\vert\Gamma)$,
$\mu^{(0)}$: see section 6.4	$\mu^{(0)}$: see section 6.4
$F = F_\alpha : \mu \mapsto \alpha\mu$	$F = F_\alpha : \mu \mapsto \alpha\mu$
$\mathbf{Z}^{(0)} = \mathbf{AY}$	$\mathbf{Z}^{(0)} = \mathbf{D}^H\mathbf{Y}$

The social declipper with a *predefined* time-frequency pattern Γ is compactly written using Algorithm 1 as:

$$\begin{bmatrix} \hat{\mathbf{W}}(\Gamma) \\ \mu(\Gamma) \\ \mathbf{Z}(\Gamma) \end{bmatrix} := \mathcal{G}(\Theta, \mathbf{M}, \{S_\mu^{\mathrm{PEW}}(\cdot|\Gamma)\}_\mu, \mu^{(0)}, F_\alpha, \mathbf{Z}^{(0)}, \beta, \mathsf{i}_{\max}),$$

The adaptive social declipper uses this to select the "optimal" pattern Γ within a prescribed collection $\{\Gamma_k\}_{k=1}^K$ for the processed signal region, by running few iterations of the algorithm (typically $\mathsf{i}_{\max}^{\mathtt{small}} = 10$). The whiteness of the residual is evaluated with the same entropy criterion (5.5) as in denoising, which maximization yields the selected pattern $\Gamma_{k^\star}$.

Correspondingly, the first value $\mu_{(k)}^{(0)}$ and the update rule F_α as well as the time-frequency patterns $\{\Gamma_k\}_{k=1}^K$ are essential for the algorithm to provide improvements. These will be specified in section 6.4.
Once the best time-frequency pattern is selected, we run Algorithm 1 with the parameters listed in Table 6.2 and a sufficiently large $\mathsf{i}_{\max}$ (typically $\mathsf{i}_{\max}^{\mathtt{large}} = 10^6$) to get

$$\hat{\mathbf{W}} := \mathcal{G}(\Theta, \mathbf{M}, \{S_\mu^{\mathrm{PEW}}(\cdot|\Gamma_{k^\star})\}_\mu, \mu_{k^\star}, F_\alpha, \mathbf{Z}_{k^\star}, \beta, \mathsf{i}_{\max}^{\mathtt{large}}).$$

In the experimental section (section 6.4), we will note that the upper bound on the iteration count $\mathsf{i}_{\max}^{\mathtt{large}}$ is never reached. Even if this work does not provide any theoretical guarantees on convergence we observe empirically that the relative norm stopping criterion β is always used to terminate the algorithm.

The pseudo-code of the adaptive social declipper for a given block of adjacent frames $\mathbf{Y} \in \mathbb{R}^{L\times(2b+1)}$ is given in Algorithm 3. Again, for the analysis version $\hat{\mathbf{X}} := \hat{\mathbf{W}}$, while for the synthesis version $\hat{\mathbf{X}} := \mathbf{D}\hat{\mathbf{W}}$.

Algorithm 3 Adaptive Social Sparse Declipper

Require: $\mathbf{Y}$, ε, $\mathbf{A}$ or $\mathbf{D}$, $\{\Gamma_k\}_k$, $\{\mu_k^{(0)}\}_k$, α, β, $\mathsf{i}_{\max}^{\mathtt{small}}$, $\mathsf{i}_{\max}^{\mathtt{large}}$
set parameters from Table 5.2, $\alpha = 1$
for all k **do**

$$\begin{bmatrix} \hat{\mathbf{W}}_k \\ \mu_k \\ \mathbf{Z}_k \end{bmatrix} := \mathcal{G}(\Theta, \mathbf{M}, \{S_\mu^{\mathrm{PEW}}(\cdot|\Gamma_k)\}_\mu, \mu_k^{(0)}, F_\alpha, \mathbf{Z}^{(0)}, \beta, \mathsf{i}_{\max}^{\mathtt{small}})$$

Compute e_k as in Equation (5.5)
$k^\star := \operatorname{argmax}_k e_k$, $\alpha = 0.99$
$\hat{\mathbf{W}} := \mathcal{G}(\Theta, \mathbf{M}, \{S_\mu^{\mathrm{PEW}}(\cdot|\Gamma_{k^\star})\}_\mu, \mu_{k^\star}, F_\alpha, \mathbf{Z}_{k^\star}, \beta, \mathsf{i}_{\max}^{\mathtt{large}})$.
return $\hat{\mathbf{W}}$

6.2.4 Overlap-add synthesis

As in denoising, the overall declipped signal is obtained by overlap-add, here without any Wiener filtering post-processing.

6.3 Multichannel structured (co)sparse declipper

To extend the framework to account for multichannel declipping scenarii we now observe a time-domain multichannel clipped audio signal composed of C channels. $\mathbf{Y} \in \mathbb{R}^{L \times C}$ denotes a windowed frame of that signal and $\mathbf{X} \in \mathbb{R}^{L \times C}$ its clean version. Equivalently to the monochannel cases, we define $\mathbf{Z}$ a frequency representation of $\mathbf{X}$. L is the number of time-domain samples in a frame and C is the number of channels.

The hard-clipping degradation model (Equation (6.1)) for $\mathbf{Y}$ extended to the multichannel case writes:

$$\mathbf{Y}_{ic} = \begin{cases} \mathbf{X}_{ic} & \text{for } |\mathbf{X}_{ic}| \leq \tau_c; \\ \operatorname{sgn}(\mathbf{X}_{ic})\tau_c & \text{otherwise;} \end{cases} \tag{6.3}$$

with $\mathbf{Y}_{ic}$ (resp. $\mathbf{X}_{ic}$) the i^{th} sample recorded on the c^{th} channel from $\mathbf{Y}$ (resp. $\mathbf{X}$) and τ_c the hard-clipping level in the c^{th} channel. Figure 6.4 below illustrates the magnitude saturation model in the multichannel case.

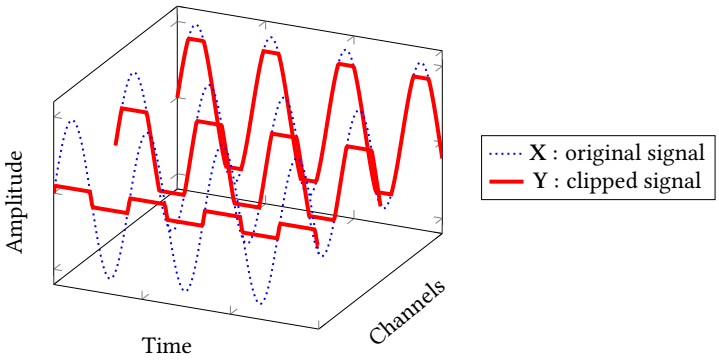


Figure 6.4 – Multichannel hard-clipping model (Equation (6.3))

This section proposes a blind approach to joint declipping of multichannel audio, which operates purely at the signal level and does not require any kind of spatial information (including the microphone positions). Intuitively, we expect that a joint processing of all channels could be more efficient than declipping independently each channel with state-of-the-art single-channel algorithms presented in subsection 6.1.2. Indeed, in the context of small/compact microphone antennas recordings, we could think that such algorithms could benefit from the redundancy of information between channels and particularly in the frequency domain. The method is based on a (co)sparse model of data, with the original addition of a *structured* sparsity prior across channels which allows to take implicitly into account the spatial correlation (see subsection 2.3.2 page 20).

The goal here is to simultaneously declip each channel in the observation $\mathbf{Y}$ to output an estimate $\hat{\mathbf{X}}$ which satisfies:

- the channel-aware structured (co)sparsity modeling constraint,
- the data fidelity constraint regarding the clipped $\mathbf{Y}$.

For that we instantiate the algorithmic framework with an appropriate shrinkage and projection operator. Projection on the (group-sparse) modeling constraint is achieved using the Group Empirical Wiener (GEW) and Quadratic Group Empirical Wiener (Quad-GEW) operators (see Equation (3.18) and Equation (3.19) page 30). Similarly to the single channel scenarii, the detailed parameters are listed in Table 6.3.

Table 6.3 – Parameters of Algorithm 1 for the Channel Aware Sparse Declipper

Analysis	**Synthesis**
$\Theta = \left\{ \mathbf{W} \mid \begin{array}{c} \mathbf{W}_{\Omega_r} = \mathbf{Y}_{\Omega_r}; \\ \mathbf{W}_{\Omega_+} \succcurlyeq \mathbf{Y}_{\Omega_+}; \\ \mathbf{W}_{\Omega_-} \preccurlyeq \mathbf{Y}_{\Omega_-}. \end{array} \right\}$	$\Theta = \left\{ \mathbf{W} \mid \begin{array}{c} (\mathbf{DW})_{\Omega_r} = \mathbf{Y}_{\Omega_r}; \\ (\mathbf{DW})_{\Omega_+} \succcurlyeq \mathbf{Y}_{\Omega_+}; \\ (\mathbf{DW})_{\Omega_-} \preccurlyeq \mathbf{Y}_{\Omega_-}. \end{array} \right\}$
$\mathbf{M} = \mathbf{A} \in \mathbb{C}^{\mathsf{P}\times\mathsf{L}}, \mathsf{P} \geq \mathsf{L}$	$\mathbf{M} = \mathbf{I} \in \mathbb{C}^{\mathsf{L}\times\mathsf{L}}, \mathbf{D} \in \mathbb{C}^{\mathsf{L}\times\mathsf{S}},$
$\mathcal{S}_\mu(\cdot) = \mathcal{S}_\mu^{\mathrm{GEW}}(\cdot\|\Gamma)$	$\mathcal{S}_\mu(\cdot) = \mathcal{S}_\mu^{\mathrm{GEW}}(\cdot\|\Gamma),$
$\mu^{(0)}$: see section 6.4	$\mu^{(0)}$: see section 6.4
$F = F_\alpha : \mu \mapsto \alpha\mu$	$F = F_\alpha : \mu \mapsto \alpha\mu$
$\mathbf{Z}^{(0)} = \mathbf{AY}$	$\mathbf{Z}^{(0)} = \mathbf{D}^{\mathsf{H}}\mathbf{Y}$

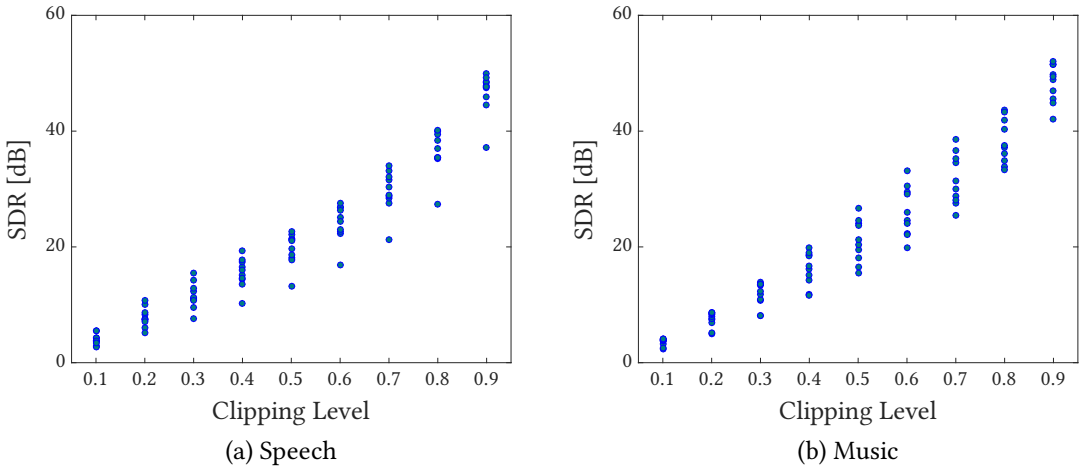
Iterating Algorithm 1 with the parameters described above gives a declipped estimate $\hat{\mathbf{W}}$ such that:

$$\hat{\mathbf{W}} := \mathcal{G}(\Theta, \mathbf{M}, \{\mathcal{S}_\mu^{\mathrm{GEW}}(\cdot)\}_\mu, \mu^{(0)}, F, \mathbf{Z}^{(0)}, \beta, \mathrm{i}_{\max}).$$

We recall that for the analysis version $\hat{\mathbf{X}} := \hat{\mathbf{W}}$, while for the synthesis version $\hat{\mathbf{X}} := \mathbf{D}\hat{\mathbf{W}}$.

6.4 Experiments

For this experimental section, we first discuss measures to rate the clipping degradation. Second, results on small scale experiments are presented to compare the effect of frequency transform redundancy with the new declipping methods presented above. Then, large scale benchmarking results compare the impact of the different models and saturation levels on the declipping performance for audio signals. Some results investigating influence of the stopping criterion are presented afterwards before comparing the algorithms with state-of-the art declipping methods. Finally, we show experiments for multichannel recordings and investigate frequency transform redundancy as well as sparsifying operators on declipping efficiency.



(a) Speech (b) Music

Figure 6.5 – Clipping levels v.s. SDR comparisons (*SMALLbox examples*)

6.4.1 Quantifying the saturation

Keeping in mind the effect of clipping on frequency content and knowing that such a degradation is also highly signal dependent means that quantifying the consequence of saturation can be itself an interesting problem. Even looking at the degradation from a signal perspective only can lead to various interpretations whether one focuses on the clipping threshold, the distortion or the amount of affected samples. The following paragraph will try to relate these indexes used for rating saturation and provide clipping scales where perceptual differences are of interest.

Commonly [Adler *et al.* 2012, Harvilla & Stern 2014, Siedenburg *et al.* 2014, Ozerov *et al.* 2016], saturation is directly rated from the clipping threshold (τ on Figure 6.1) as it reflects how importantly the initial dynamic range of the signal is affected. The lower τ is the more severe will be the loss. Practically, studies usually work with normalized magnitude data for fair comparisons and the clipping threshold takes values in $[0; 1]$. This value which denotes how much of the peak amplitude is left after the clipping process is also referred to as "clipping level".

Another tool available for measuring the effect of clipping is the added distortion to the original signal thanks to the Signal-to-Distortion Ratio (SDR Equation (4.4)). Contrarily to the clipping threshold which can be dissociated from the initial signal, SDR is highly linked to it as it takes into account the energy of the signal in the computation. Hence, usually no clear relationship between SDR and clipping thresholds can be identified as shown in Figure 6.5 (the SDR variance for a given τ is high) on the SMALLbox examples (see chapter 4 page 31).
Extreme cases for SDR for clipping are 0 dB, the induced distortion is as important as the initial signal energy (all the amplitude information is lost, just the sign of the samples are left) clipping is maximum; $+\infty$ dB, the SDR is maximum, no distortion, no clipping.

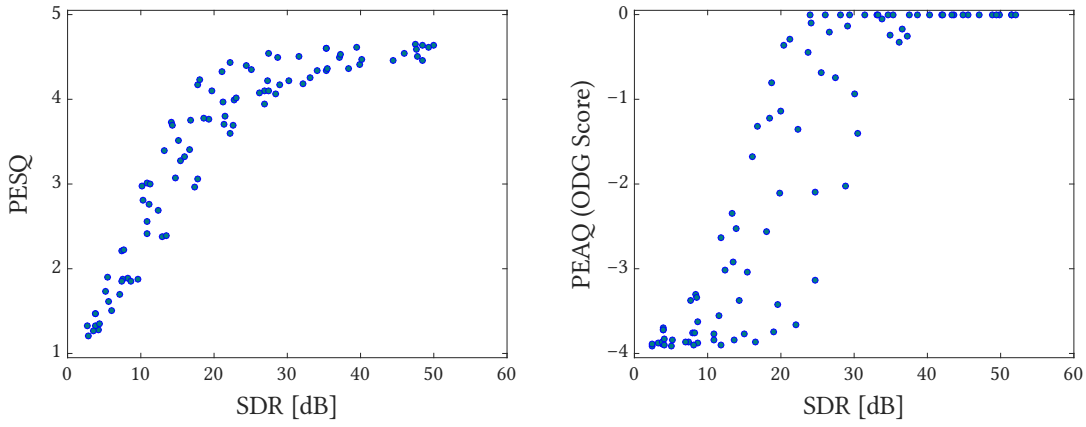
SDR and clipping threshold are measures that require either an original clean signal to compare with or the initial dynamic range. There exists a mean to estimate the seriousness of the degradation without any reference: by counting the parts of a signal affected by clipping. In a discrete setting, this boils down to counting the number of clipped samples over the complete signal. This can be denoted by "ratio of clipped samples" ($\%_{\text{Clipped}}$). The higher this ratio is, the more significant will be the loss as more samples are affected.

$$\%_{\text{Clipped}} = \frac{|\{i \in (1, ..., L) \mid |\mathbf{x}_i| \geq \tau\}|}{L} \tag{6.4}$$

Equivalently, the lower, the weaker will be the clipping. Even if this measure can be of great interest to blindly estimate the power of the degradation, it is rarely used in studies presenting declipping methods as it is not suited for direct comparisons or assessing enhancement.

Degradation range As for every study involving audio content, numerical indexes used to rate a degradation are valid if they correlate somehow to quality ratings. Unlike for denoising, few studies focused on finding objective numerical descriptors to rate the quality of saturated audio excerpts. Some years ago, [Defraene *et al.* 2013] validated the correlation between PEAQ scores (see chapter 4 page 31) and clipped audio quality assessments thanks to listening tests. In the following, we will briefly investigate the relationship between clipping thresholds, SDR, PEAQ and PESQ to identify a degradation range that is of interest for this study and justifies the choice of clipping conditions. These preliminary comparisons are also held on the SMALLbox examples. For PEAQ scores on music excerpts, we recall that it ranges from −4 to 0. The closer the value is from −4, the more annoying will be perceived the clipping consequences. Similarly, for PESQ scores on speech excerpts, we recall that it ranges from 1 to 5 and the closer the value is from 1 the worse will be the quality.

Figure 6.6 presents objective quality for the same examples as Figure 6.5 (clipping thresholds between 0.1 and 0.9). What is clearly seen is a global trend followed for both speech and music: the rated quality increases with the SDR. What is even more striking for music is that quality stabilizes at maximum (*imperceptible* degradation) for SDR roughly above 30 dB. This certainly does not mean that there is no degradation for high SDR but can give insights on the relevant clipping thresholds to consider when the final application involves humans or machines. Indeed, if high SDRs produce unnoticeable degradation for humans it can still be a struggle for machine performing automated speech recognition, music classification, etc. With this in mind, if one compares Figure 6.6b and Figure 6.5b, it appears that most of the examples whose clipping threshold is greater than 0.6 might not be of crucial interest for audio quality rating. For speech, results are more moderate in the sense that there is no as clear plateau effect with the PESQ measure as for music with PEAQ.



(a) Speech (b) Music

Figure 6.6 – Objective quality v.s. SDR comparisons (*SMALLbox examples*)

Considering what quality measure one should use to describe the intensity of the degradation caused by clipping, two main options are available:

- if a reference signal is available - clipping threshold or initial SDR;
- otherwise - ratio of clipped samples $\%_{\text{Clipped}}$.

When dealing with audio reconstruction methods and more precisely declipping, the selected enhancement measure is often the SDR difference before and after processing (ΔSDR). Consistency with the enhancement measure could be one argument to tend towards SDR. When it comes to clipping thresholds, one has to stay careful as it might lead to important variability in the resulting distortion as seen on Figure 6.5. A first explanation of this variability, comes from the temporal structure of the considered audio signals. One deduction is that a same clipping threshold will have different consequences on a signal with high dynamic and a signal whose magnitude values are more uniformly distributed. On the former, clipping will rather have a "limiting" effect affecting only a few samples while on the latter more samples will suffer from the saturation. Another argument in favor of SDR is that it is a measure that only the affected samples will have effect on. Indeed, energy on the reliable parts of the signal will remain the same. Plus, as long as clipping consistency is used for desaturation, this energy will not change either on the processed signal reliable parts. If we consider that higher amplitude hence higher energy portions of a signal will be affected by clipping, having a degradation measure able to specifically target the changes for these portions should probably be preferred compared to a measure providing an overall value (like clipping level or ratio of clipped samples). These insights on clipping effects and quantification drive the choice towards SDR for the tested conditions in the experimental section (subsection 6.4.2 and subsection 6.4.3).

6.4.2 Single channel experiments

This experimental section dedicated to single channel declipping first investigates influence of frequency transform redundancy. Then, the different sparse priors (analysis v.s. synthesis) are compared regarding their reconstruction and runtime efficiency. These two first studies are run on the SMALL database. After, we present a wide range comparison of the models on the RWC database before investigating the influence of the stopping criterion of the algorithmic framework and running a comparison with state-of-the-art methods.

Compared methods Similarly to the denoising section, we consider the plain sparse, plain cosparse, social sparse and social cosparse declippers. We set the common parameters for the algorithms as listed below.

- Frame size L = 64 ms for music L = 32 ms for speech;
- Hamming windows, overlap: 75%;
- $\mathrm{i}_{\max}^{\mathtt{small}} = 10$, $\mathrm{i}_{\max}^{\mathtt{large}} = 10^6$;
- Analysis operator, $\mathbf{A}$ = DFT;
- Synthesis operator, $\mathbf{D}$ = inverse DFT;
- Accuracy, $\beta = 10^{-3}$.

Considering the adaptive social sparse declipper and similarly to denoising, we set the collection of time-frequency patterns $\{\Gamma_k\}_{k=1}^{\mathsf{K}}$ to match the one presented on Figure 3.2 for music and Figure 3.3 for speech. The specific choice of $\mu_k^{(0)} := \|\Gamma\|_0 \times (1 - \|\mathbf{Y}\|_\infty)$ is motivated by the sparsity degree of the time-frequency neighborhood considered. With this parameterization, the regularization behavior is initialized inversely proportional to the maximal magnitude of the clipped signal, allowing highly clipped configurations to retain sparser regularization. Contrarily to the social sparse denoising method, we notice better improvements when the μ parameter is *not* updated during the initialization loop (*i.e.* $\alpha = 1$). Once the proper $\Gamma_{k^\star}$ is selected, we obtained the best declipping results with μ following a geometric progression of common ratio α with $\alpha = 0.99$. We finally set the number of overlapping segments to $\mathsf{b} = 5$ for music (*i.e.* $\mathbf{Y} \in \mathbb{R}^{\mathsf{L}\times 11}$), $\mathsf{b} = 1$ for speech (*i.e.* $\mathbf{Y} \in \mathbb{R}^{\mathsf{L}\times 3}$).

Influence of frequency transform redundancy This first study aims at comparing reconstruction and computational efficiency of different redundancy factors. We recall that the analysis operator $\mathbf{A} \in \mathbb{C}^{\mathsf{P}\times\mathsf{L}}$ (respectively the dictionary $\mathbf{D}^{\mathsf{H}} \in \mathbb{C}^{\mathsf{L}\times\mathsf{S}}$) are possibly redundant Discrete Fourier Transforms (DFT); ($\mathsf{P} = \mathsf{RL}$ or $\mathsf{S} = \mathsf{RL}$). Table 6.4 presents processing times for non-redundant ($\mathsf{R} = 1$), twice redundant ($\mathsf{R} = 2$) and four times redundant ($\mathsf{R} = 4$) DFT when used with all the plain/social (co)sparse models.

Table 6.4 – Processing times comparison (plain/social (co)sparse declippers)

(a) Runtime performance (ratio to realtime processing)

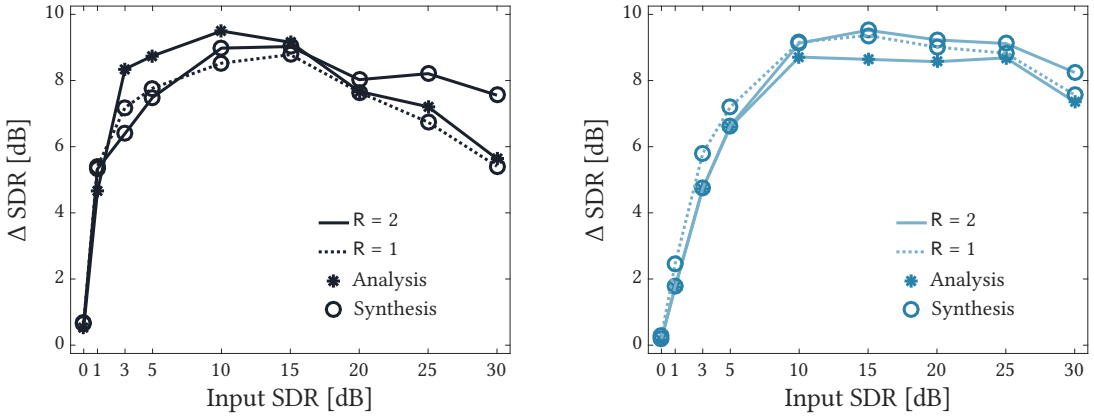
	Input SDR: 5 dB				Input SDR: 20 dB			
	Analysis		Synthesis		Analysis		Synthesis	
	Plain	Social	Plain	Social	Plain	Social	Plain	Social
R = 1	12.8	158.7	13.3	172.9	12.6	73.6	13.2	73.7
R = 2	41.1	407.6	18.4	497.1	33.3	211.4	16.9	231.5
R = 4	119.8	1.1×10^3	26.2	1.2×10^3	96.5	580.3	23.4	525.1

(b) Corresponding improvements (ΔSDR)

	Input SDR: 5 dB				Input SDR: 20 dB			
	Analysis		Synthesis		Analysis		Synthesis	
	Plain	Social	Plain	Social	Plain	Social	Plain	Social
R = 1	7.77	7.22	7.77	7.22	7.64	9.01	7.64	9.01
R = 2	8.74	6.60	7.48	6.63	7.67	8.57	8.02	9.23
R = 4	8.53	6.97	6.53	7.72	6.98	9.17	7.90	9.47

As in denoising experiments, processing times lead us to retain only the non redundant and twice redundant setting (R = 1, R = 2) as a compromise between reconstruction and computational efficiency for further comparisons. Indeed, the four times redundant DFT seems to bring some improvements but with a substantial computational overcost.

Figure 6.7 and Figure 6.8 display averaged SDR improvements for the music and speech SMALL dataset examples for non redundant and twice redundant DFT.

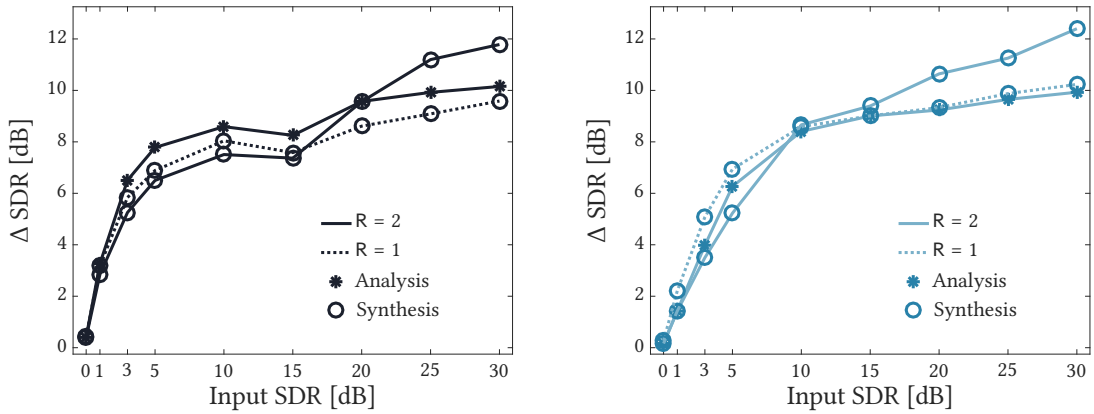


(a) Plain sparsity: ΔSDR

(b) Social sparsity: ΔSDR

Figure 6.7 – SMALL Music: Redundancy study (Declipping)

For the plain sparse models, Figure 6.7a and Figure 6.8a show benefits from twice redundant DFT as non redundant results are outperformed by either analysis or synthesis methods. For the social sparse models, Figure 6.7b and Figure 6.8b show slightly superior



(a) Plain sparsity: ΔSDR

(b) Social sparsity: ΔSDR

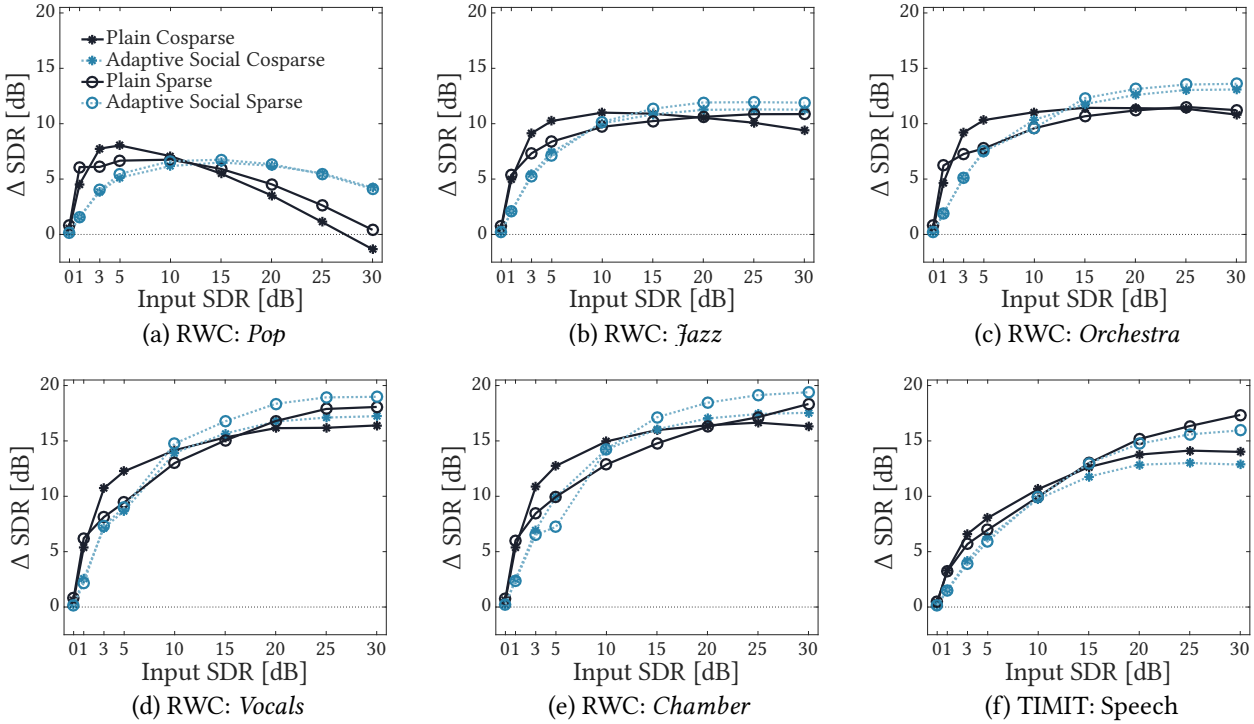
Figure 6.8 – SMALL Speech: Redundancy study (Declipping)

SDR improvements for non redundant DFT when used with severe initial degradation (input SDR $\leq 5dB$). We verify this trend for both music and speech sounds.

Large scale comparison of declipping performance In order to accurately study the influence of the (social) (co)sparse models, we extend here the study to a wide-range comparison on RWC database excerpts. To the best of our knowledge, it is the first time such a large scale validation is performed. Results presented on Figure 6.9 and Figure 6.10 show averaged measurements over all available sounds. Results for these experiments are obtained with a twice redundant frequency transform ($\mathsf{R} = 2$) to differentiate analysis and synthesis sparse models ; other parameters are unchanged and match those presented earlier.

Figure 6.9 shows the behavior of the four methods as a function of the input degradation level. For all the considered datasets, both declipping methods provide significant SDR improvements (often more than 8dB) at (almost) all considered input SDRs. Unlike for denoising, this remains the case even for relatively high input SDRs, with one exception: the Pop category, for which the Plain Cosparse brings some degradation at very high input SDR, and the overall improvement never exceeds 8dB. This may be due to the fact that most of the 100 unclipped excerpts in this category are mixes containing one or more tracks of dynamically compressed drums, and that at least 21 of them contain saturated guitar sounds.

The benefit of social modeling is clear for moderate to high input SDR (> 10dB, mild clipping), and vice-versa there is also a distinct superiority of the plain cosparse method for low input SDRs (strong clipping). Actually, the plain approaches perform 2 to 4 dB better than the adaptive social methods for input SDRs ranging from 1 to 5 dB on audio content from the RWC database. On the opposite, the trend tends to reverse above 10 dB input SDR as the social methods features improvements between 1 and 4 dB (even 7 dB for the Pop category) above the plain (co)sparse techniques. For speech content,



(a) RWC: *Pop* (b) RWC: *Jazz* (c) RWC: *Orchestra*

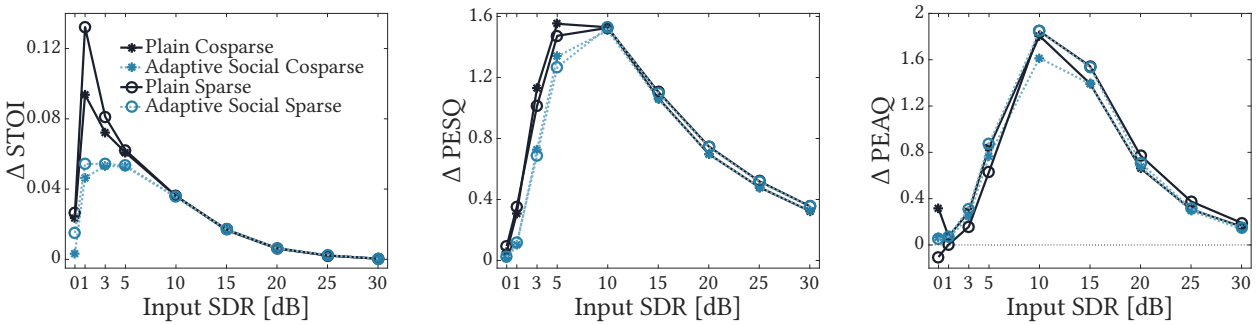
(d) RWC: *Vocals* (e) RWC: *Chamber* (f) TIMIT: Speech

Figure 6.9 – Declipping: Numerical Results Δ SDR [dB]

the difference is less obvious yet Figure 6.9f, Figure 6.10a and Figure 6.10b displays better improvements either in terms of SDR, objective intelligibility or quality for the plain sparse declipper.

Contrarily to denoising settings, standard deviation results (not detailed here) indicate that the social cosparse declipper produces more consistent results as the standard deviation is the lowest for this technique in 67% of the tested cases. We also observed that, for any of the considered algorithms, the improvement variability seems be higher for higher SDR.

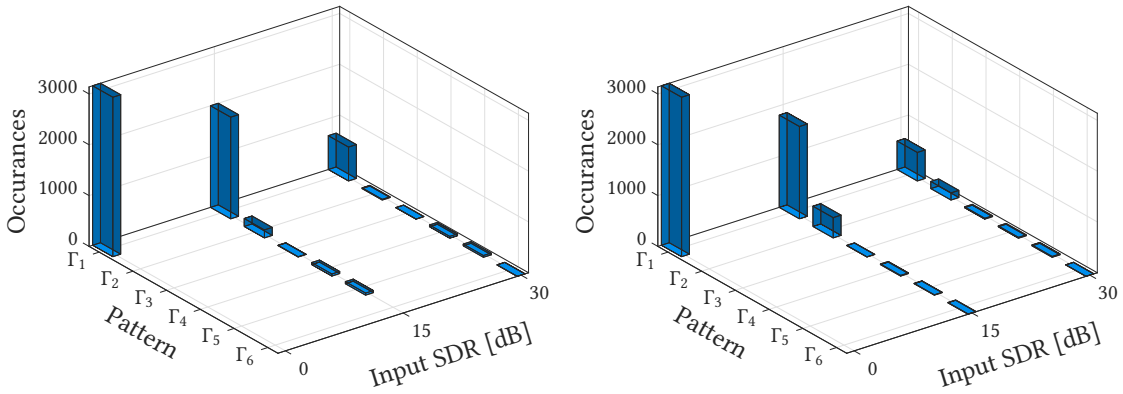
The difference in performance between the plain and social cosparse declippers on music at low input SDR might come from the nature of the degradation. Indeed, contrarily to additive noise, the magnitude saturation adds broadband stripes in the time-frequency plane due to discontinuities of the derivative in the time domain. This way, the signal's underlying structure (embodied by a time-frequency pattern Γ) is not only hidden as in the additive noise case, but also possibly distorted: during the initialization loop of the social approaches, it is possible that a "wrong" pattern Γ^* is selected. In contrast, the plain cosparse declipper cannot be affected by this type of behaviour. Another interesting result which could support this hypothesis is that for higher SDR, the social method is actually benefiting from the time-frequency structure identification as it performs better.



(a) TIMIT: Δ STOI (b) TIMIT: Δ PESQ (c) RWC: Δ PEAQ

Figure 6.10 – Declipping: Objective Quality and Intelligibility Results

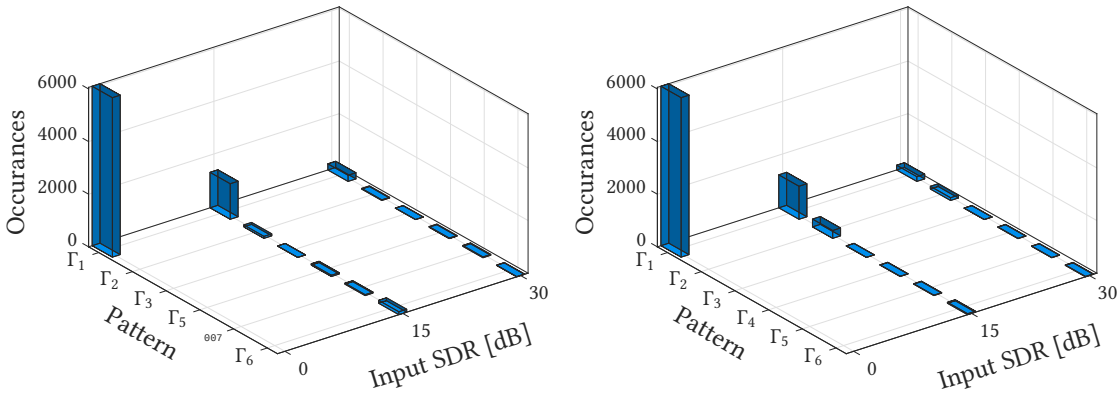
Time-frequency neighborhood selection To give some insights on this last hypothesis, Figure 6.11 and Figure 6.12 shows the time-frequency neighborhood distributions from all the declipped frames of the SMALL dataset examples (speech & music). These results are obtained with the same algorithmic parameters as the previous large scale study. We recall that the available time-frequency neighborhoods collection are presented on Figure 3.2 and Figure 3.3. Interestingly, we remark that whatever prior (analysis or synthesis), audio content (music or speech) or input SDR, the neighborhood emphasizing tonal structure (Γ_1) is mainly selected. These results are to be compared with the neighborhood distributions in the denoising case (Figure 5.8 and Figure 5.9 page 55) which are if not uniform at least much more diverse. In the light of these results, the "wrong" pattern selection hypothesis probably has to be balanced with other factors affecting the declipping. Further investigations, for example in the way the hyper parameter μ ruling the sparsity constraint is set, could be interesting.



(a) Adaptive social cosparse (b) Adaptive social sparse

Figure 6.11 – SMALL Music: Time-frequency pattern distribution (Declipping)

Computational aspects As the DFT can be efficiently implemented with a fast transform, the computational cost of the declipping procedures mainly stems from the sparsi-



(a) Adaptive social cosparse

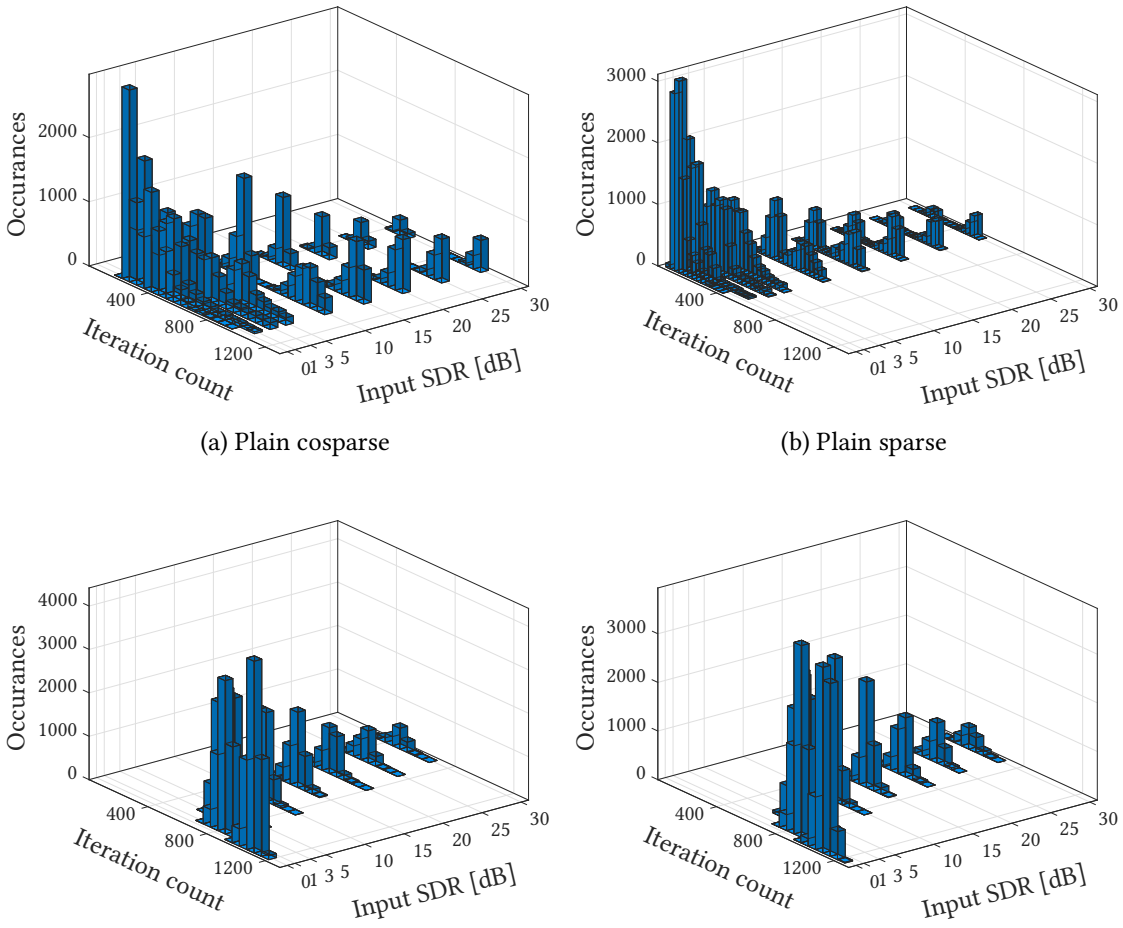
(b) Adaptive social sparse

Figure 6.12 – SMALL Speech: Time-frequency pattern distribution (Declipping)

fying step and the projection on the declipping constraint. The behavior of the sparsifying operator (shrinkage) when used inside the ADMM framework and the total number of iterations needed to finish or converge will drive the computational properties of the methods. To give some insights on this last parameter, Figure 6.13 gives total number of iterations distributions as a function of the input SDR for every flavor og the algorithmic framework. These results are obtained from the same tested material as the previous study. More precisely, all the declipped frames of the examples in the SMALLbox dataset (speech and music).

As a matter of interest, we note that for plain methods the number of iterations needed to finish seems to be lower for low SDR whereas it increases at high SDR. A reverse trend is observed for adaptive social methods. The plain sparse method globally needs less iterations to finish as for every tested case the method stops after at most 600 iterations. This supports the observation of a similar behavior in [Záviška *et al.* 2018]. Finally, similarly to denoising, one crucial result is that for every single declipped frame the algorithms stop way below $\mathrm{i}_{\max} = 10^6$ iterations. This means that for these examples (SMALL dataset), whatever flavor of the algorithmic framework is used, it terminates thanks to the relative stopping criterion β and not the a predefined iterations upper bound $\mathrm{i}_{\max}$.

Influence of the stopping criterion β A less studied parameter is the stopping criterion for sparse iterative reconstruction algorithms. In this part, we vary the *accuracy* parameter β of the algorithmic framework to detect possible different behavior when used with (social) (co)sparse models. We recall that β is used, aside to the maximum number of iterations i_{max}, as a threshold to terminate Algorithm 1 (see: page 27). Figure 6.14 and Figure 6.15 present averaged SDR improvements for different β on the SMALL sound examples. Legend on Figure 6.14a extends to the other plots of the figure.



(a) Plain cosparse

(b) Plain sparse

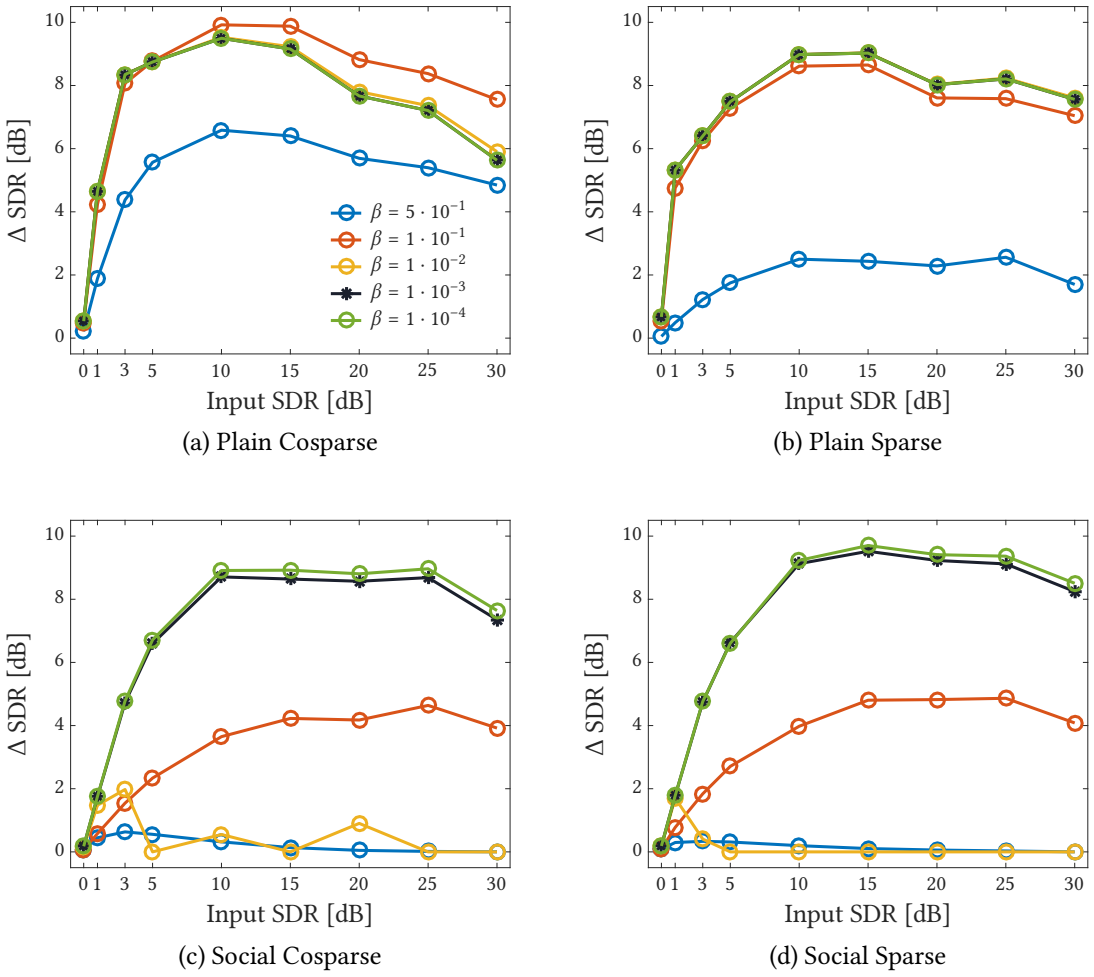
(c) Adaptive social cosparse

(d) Adaptive social sparse

Figure 6.13 – SMALL: Total iteration count distribution (Declipping)

We remark that a smaller stopping criterion does not necessarily mean better SDR improvement. While it seems to still be the case for the social (co)sparse and plain sparse methods (Figure 6.14b, Figure 6.14c and Figure 6.14d), decreasing the accuracy parameter below $\beta = 1 \cdot 10^{-1}$ worsen results for moderate to light clipping conditions (SDR $\geq$ 10 dB). A smaller stopping criterion β allows the algorithms to retain a larger number of iterations. In a convex optimization case, this would mean stopping closer to the global solution and therefore a better reconstruction if the model is appropriate. On the contrary, for non-convex cases, a larger number of iterations would not automatically be beneficial.

Comparison to state-of-the art declipping methods Finally, in the following we compare the previous instances of the algorithmic framework with a baseline declipping method C-IHT, *Consistent Iterative Hard-Thresholding* ([Kitić *et al.* 2013]) and state-of-the-art methods A-SPADE, *Analysis SParse Audio DEclipper* ([Kitić *et al.* 2015]) and Social Sparsity Declipper ([Siedenburg *et al.* 2014]). For the methods presented in this



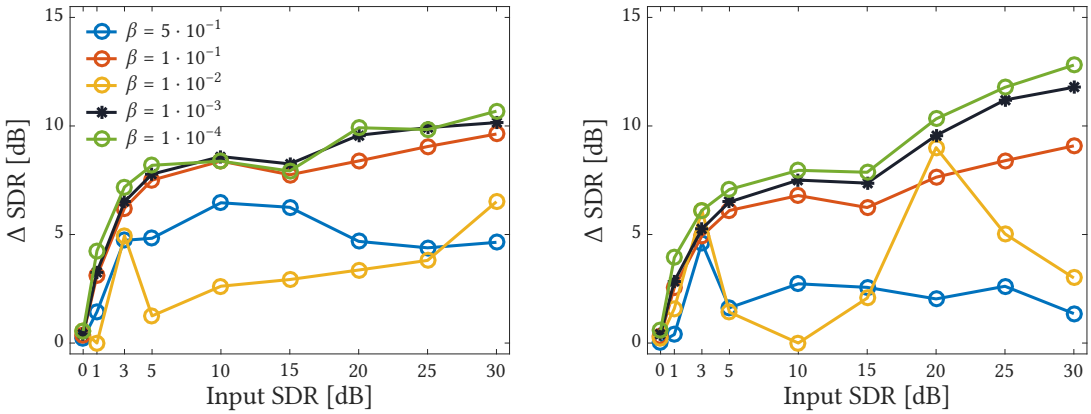
(a) Plain Cosparse

(b) Plain Sparse

(c) Social Cosparse

(d) Social Sparse

Figure 6.14 – SMALL Music: Stopping criterion study (Declipping)



(a) Plain Cosparse

(b) Plain Sparse

(c) Social Cosparse

(d) Social Sparse

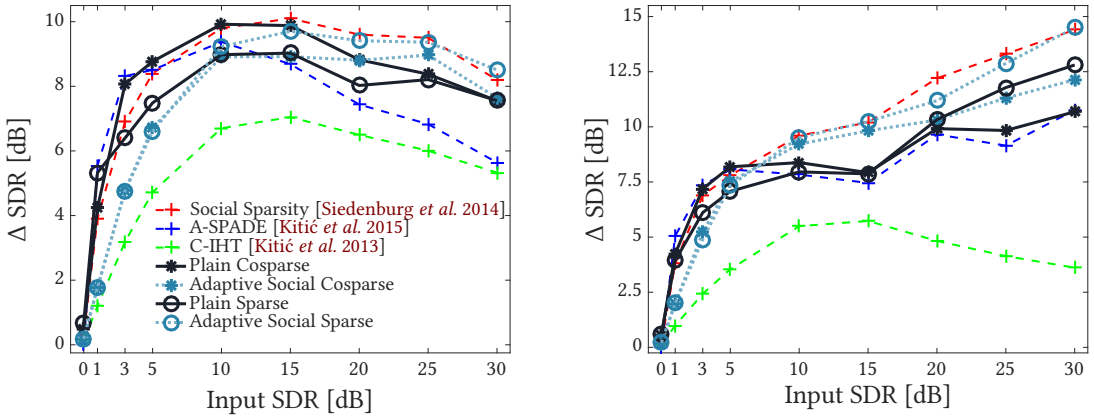
Figure 6.15 – SMALL Speech: Stopping criterion study (Declipping)

Table 6.5 – Computational performance of declipping methods (Ratio to real-time processing ×RT)

Input SDR [dB]	Plain cosparse	Adaptive social cosparse	Plain sparse	Adaptive social sparse	Social Sparsity [Siedenburg *et al.* 2014] LTFAT	Social Sparsity [Siedenburg *et al.* 2014] Compiled LTFAT	A-SPADE [Kitić *et al.* 2015]	C-IHT [Kitić *et al.* 2013]
0	33.4	535.5	11.9	643.6	754.2	62.1	0.07	19.4
1	32.4	401.1	11.9	471.4	753.3	62.2	17.7	19.0
3	37.4	442.2	17.9	532.1	754.8	62.2	24.9	17.2
5	38.4	401.8	16.5	463.7	754.5	62.1	27.3	14.9
10	26.4	271.5	16.3	316.8	755.4	62.1	24.4	9.8
15	20.3	131.1	10.7	110.0	754.6	62.1	16.5	5.7
20	12.4	64.7	4.8	72.1	756.1	62.2	10.4	3.4
25	7.8	37.6	4.2	42.7	754.6	62.1	6.6	2.2
30	4.0	16.3	2.5	21.7	756.4	62.2	4.2	1.3

manuscript, we chose for this last study:

- twice redundant DFT,
- stopping criterion β providing the best averaged SDR improvement for each of our algorithms.

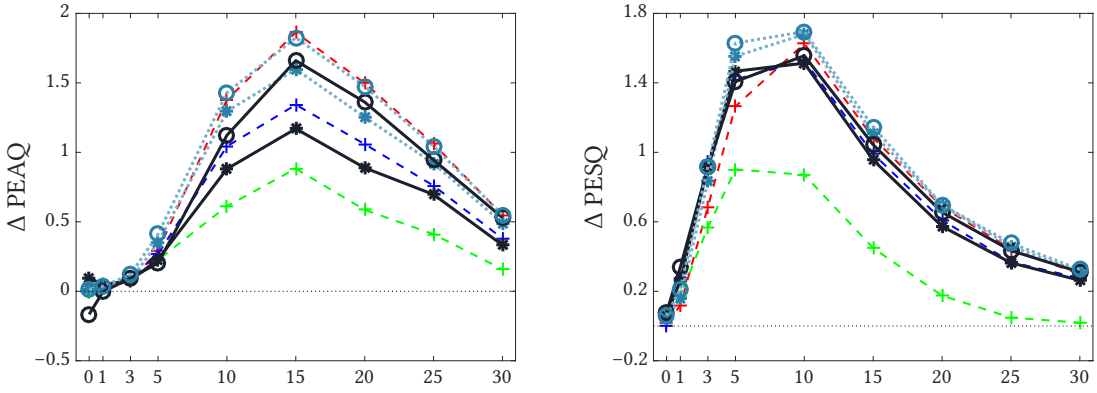


(a) SMALL: Music (b) SMALL: Speech

Figure 6.16 – Declipping: State-of-the-art comparison (Numerical results Δ SDR)

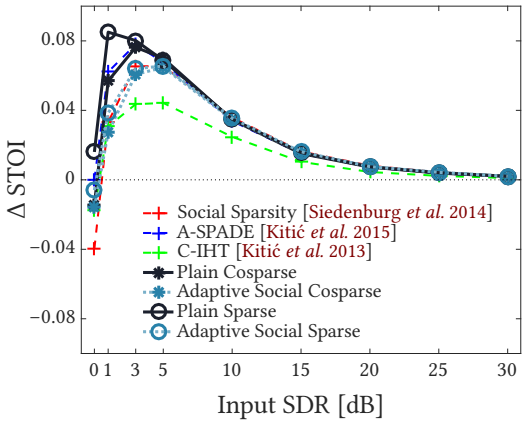
Figure 6.16 displays average SDR improvements for all the aforementioned methods on the SMALL sound examples. We note that for severe clipping, plain (co)sparse models provide better SDR improvements than social sparse models. On the opposite, we notice superiority of methods including social sparse models for lighter degradation (input SDR ≥ 15 dB). This confirms the observed trend on the large-scale comparison with RWC dataset on Figure 6.9.

To conclude these experiments dedicated to single-channel declipping, Table 6.5 presents computational efficiency corresponding to the results obtained on Figure 6.17b. All the experiments were performed using a Matlab®implementation of the algorithms on a workstation equipped with a 2.4 Ghz Intel® Xeon®processor and 32 GB of RAM memory.



(a) SMALL: Music (Δ PEAQ)

(b) SMALL: Speech (Δ PESQ)



(c) SMALL: Speech (Δ STOI)

Figure 6.17 – Declipping: State-of-the-art comparison (Objective Quality and Intelligibility Results)

We note that the method provided in [Siedenburg *et al.* 2014] uses the Structured Sparsity Toolbox[2] relying on the Large Time-Frequency Analysis Toolbox[3] ([Søndergaard *et al.* 2012]). This last toolbox can either be used directly from its initial Matlab® implementation or a compiled version using some optimized backend C functions. For this reason, we include results with the regular LTFAT toolbox and its compiled version. We note that for almost all methods, starting from 5 dB, the processing time seems to be inversely proportional to the input SDR. The exception is for [Siedenburg *et al.* 2014] which computational efficiency seems to be independent from the input SDR. The plausible explanation is that contrarily to the other methods, it only relies on an upper bound on the iteration count to stop the algorithm. Hence, the corresponding higher computational time could certainly be drastically reduced by lowering the maximum iteration count. We emphasize that tuning the parameterization of the plain (co)sparse methods allows real-time processing on a regular laptop computer. We also note that for [Siedenburg *et al.* 2014] some code optimization (*i.e.* C backend) can drastically improve the computational performances. Therefore, this is a solution that is currently being investigated as part of an industrial partnership for a technology transfer of the plain cosparse method (some information should be available at: `https://spade.inria.fr`).

6.4.3 Multichannel experiments

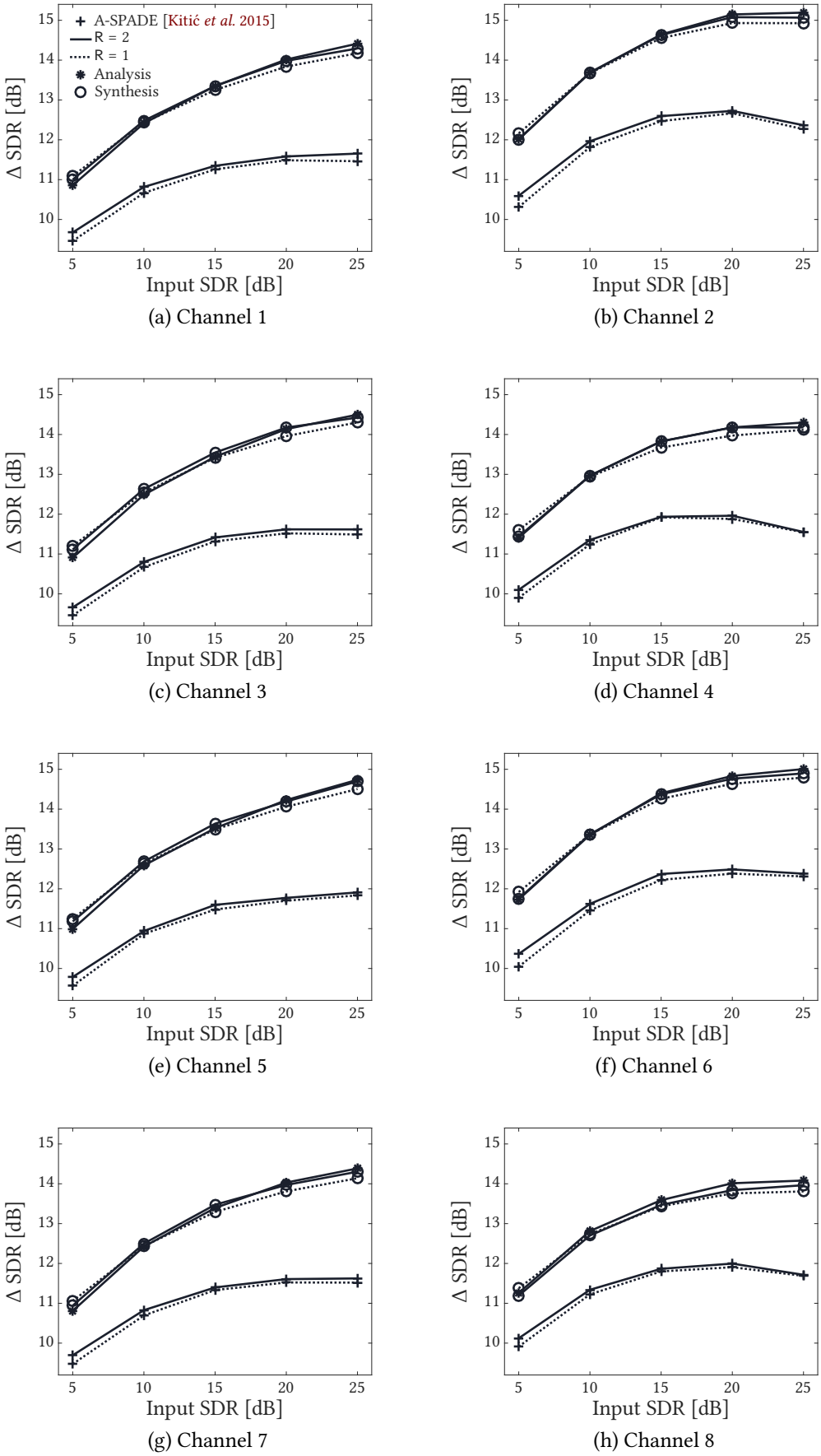
This experimental section dedicated to multichannel declipping first investigates declipping and computational efficiency on 8-channel recordings of structured (co)sparse algorithms. Then, we present a study involving stereo recordings.

Compared methods We consider the channel-aware structured sparse and structured cosparse declippers as well as the A-SPADE ([Kitić *et al.* 2015]) method (done separately on each channel) for comparison. We set the common parameters for the algorithms as listed below.

- Frame size L = 64 ms;
- Hamming windows, overlap: 75%;
- $i_{max} = 10^6$;
- Analysis operator, $\mathbf{A}$ = DFT;
- Synthesis operator, $\mathbf{D}$ = inverse DFT;
- Redundancy: see below.

Considering the structured (co)sparse algorithms we chose the Group Empirical Wiener shrinkage and the Quadratic Group Empirical Wiener (Equation (3.18) and Equation (3.19)) to enforce the structured (co)sparsity prior across channels.

[2] `https://homepage.univie.ac.at/monika.doerfler/StrucAudio.html`
[3] `http://ltfat.github.io`



(a) Channel 1 (b) Channel 2

(c) Channel 3 (d) Channel 4

(e) Channel 5 (f) Channel 6

(g) Channel 7 (h) Channel 8

Figure 6.18 – VoiceHome2: Multichannel speech declipping (Numerical results Δ SDR [dB])

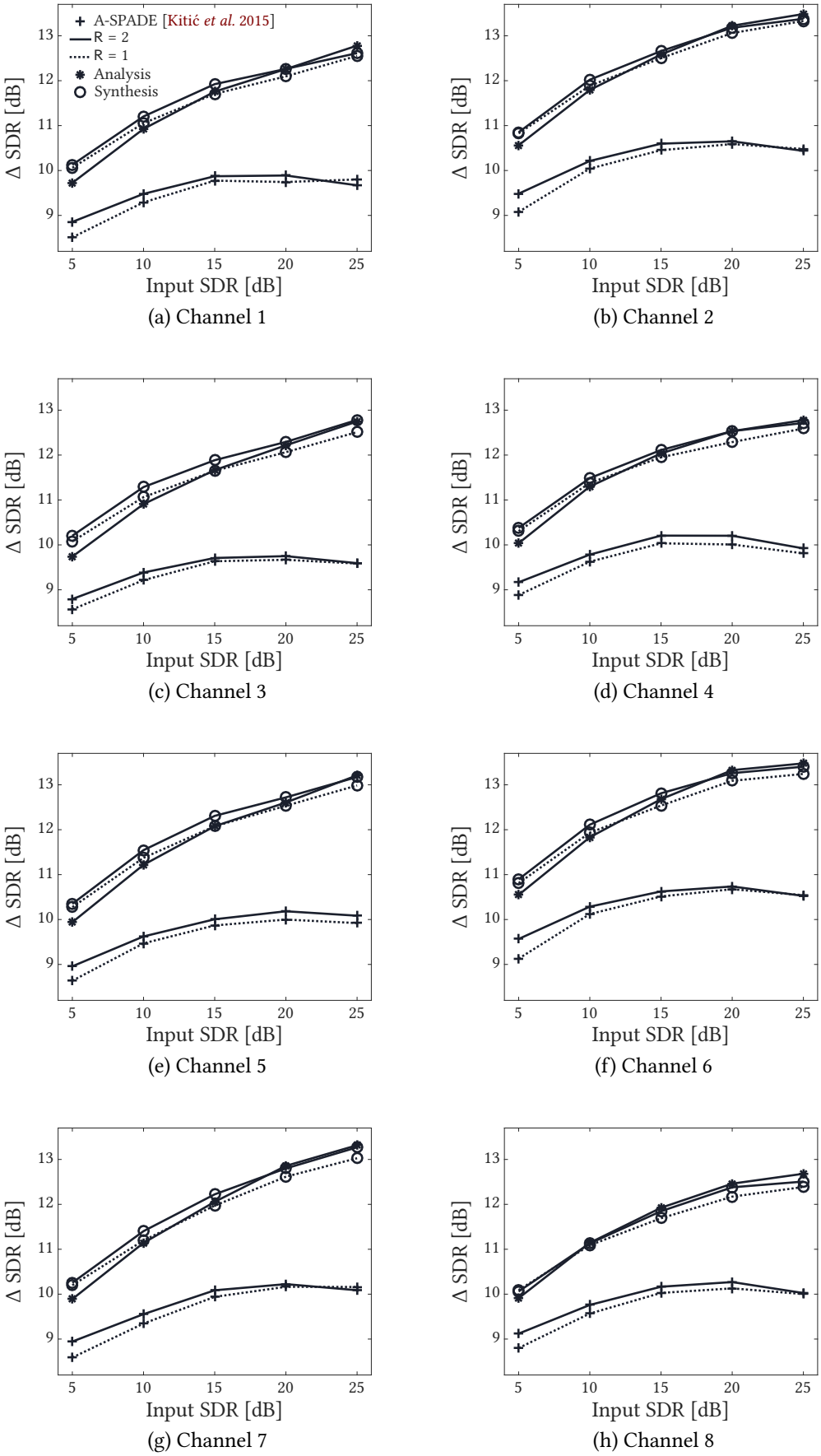


Figure 6.19 – VoiceHome2: Multichannel mixed speech & music declipping (Numerical results Δ SDR [dB])

We perform experiments on 8-channel recordings from the VoiceHome2 Corpus. We artificially saturate all the excerpts at five input SDR levels in dB: 5, 10, 15, 20, 25. The analysis operator $\mathbf{A} \in \mathbb{C}^{\mathsf{P}\times\mathsf{L}}$ (respectively the dictionary $\mathbf{D}^{\mathsf{H}} \in \mathbb{C}^{\mathsf{S}\times\mathsf{L}}$) are possibly redundant Discrete Fourier Transforms (DFT); indeed, we study the effect of the frequency transform redundancy by comparing two redundancy factors: $\mathsf{R} = 1$, $\mathsf{R} = 2$ (we recall that $\mathsf{P} = \mathsf{RL}$ or $\mathsf{S} = \mathsf{RL}$). A first pilot study (data not shown) allowed us to choose the best parameters α and $\mu^{(0)}$. The best results are obtained with $\mu^{(0)} = \mathsf{L}$ and $\alpha = 0.99$. As a reminder, L the frame size is directly linked to the sampling frequency of the considered audio signal so is $\mu^{(0)}$. We confront the channel-aware structured (co)sparse instances of the algorithmic framework with the A-SPADE [Kitić *et al.* 2015] state-of-the-art declipper (which uses a simple cosparse prior and operates on each channel separately) and compare results channel-by-channel. Performance is assessed by ΔSDR for reconstruction and ratio to real-time processing (×RT) for runtime.

Comparison of declipping performance SDR improvement results are presented in Figure 6.18 (for speech only subset) and Figure 6.19 (for mixed music and speech). Legend on Figure 6.18a and Figure 6.19a extends to the other ones. Plain lines displays averaged results for twice redundant DFT while dotted lines represent results for non-redundant transform. Stars displays ΔSDR for the analysis version of the channel-aware structured sparse method and circles shows it for the synthesis version. Crosses displays results for A-SPADE. We observe that both the analysis and synthesis structured sparse multichannel declipping methods outperforms the A-SPADE algorithm by 1 dB to more than 3 dB in all settings. The improvement brought by the multichannel declipper over A-SPADE is even more salient on mixed speech and music data (which is the most difficult subset, with a globally lower performance for both algorithms, compared to that obtained on speech only data.). Finally, results appear to be consistent from one channel to another.

Redundancy: analysis *v.s.* synthesis The effect of a redundant DFT transform ($\mathsf{R} = 2$) appears to be slightly different for each method. We note that twice redundant DFT provides at least as good results as non redundant DFT for A-SPADE. For the methods embedding structured sparsity across channels, the twice redundant synthesis and analysis versions seem to perform similarly on speech only content. Nonetheless, for mixed speech and music, the synthesis approach seems to slightly outperform the analysis version and especially for low input SDR. Non-redundant DFT seems to be detrimental on the ΔSDR for low input degradation. This effect is clearer for every channels with mixed speech and music on figure Figure 6.19. For low input SDR non redundant structured sparse methods performs at least as good as the structured cosparse method.

Computational Aspects As the DFT can be efficiently implemented with a fast transform, the computational cost of the declipping procedure mainly stems from the sparsifying step and the projection on the declipping constraint. For this runtime comparisons, we choose a subset of 25 excerpts (totalizing 3 minutes of audio) from the dataset

Table 6.6 – VoiceHome2: runtime tests numerical results (multichannel declipping)

(a) Runtime performance (ratio to realtime processing ×RT)

Algorithm		Structured (Co)sparse				A-SPADE [Kitić *et al.* 2015]	
Redundancy		R = 1		R = 2		R = 1	R = 2
Prior		Analysis	Synthesis	Analysis	Synthesis		
Input SDR [dB]	5	94	94	206	287	73	190
	10	61	61	135	200	59	148
	15	40	40	89	134	42	103
	20	26	26	58	89	29	72
	25	18	18	37	57	20	50

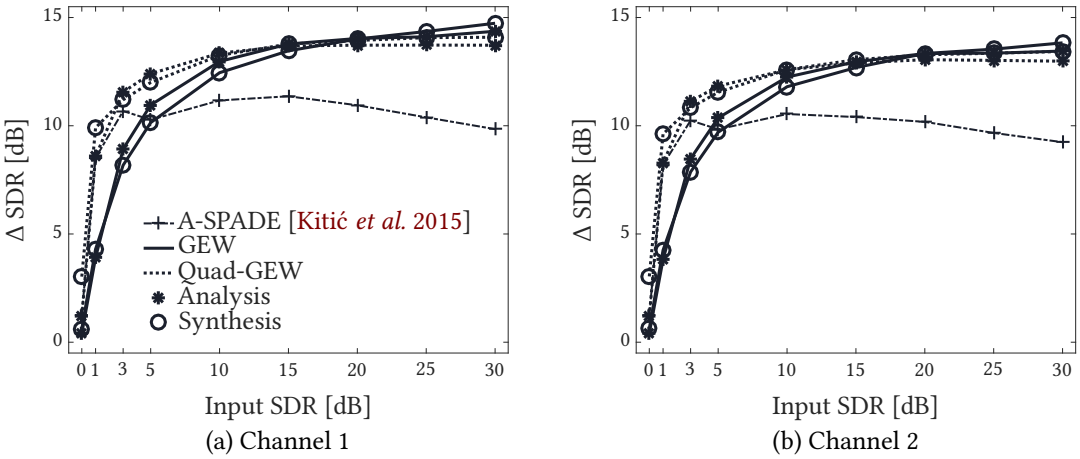
(b) Corresponding improvements (ΔSDR)

Algorithm		Structured (Co)sparse				A-SPADE [Kitić *et al.* 2015]	
Redundancy		R = 1		R = 2		R = 1	R = 2
Prior		Analysis	Synthesis	Analysis	Synthesis		
Input SDR [dB]	5	9.52	9.52	9.26	11.14	9.31	9.63
	10	12.06	12.06	11.94	12.53	10.57	10.79
	15	13.20	13.20	13.11	13.43	11.27	11.37
	20	13.82	13.82	13.78	14.16	11.67	11.79
	25	14.06	14.06	13.91	14.39	11.73	11.68

and compare the computing time of the multichannel declippers and the A-SPADE algorithms. The runtime tests are performed on workstations running the Matlab® associated code in single-thread mode. The computers are equipped with Intel®Xeon® CPU 5140 @2.33 GHz with 2 GB available ram memory. Table 6.6a shows runtime performances and Table 6.6b the corresponding SDR improvements (ΔSDR) averaged on the eight channels and the 25 excerpts. We clearly note higher computing times for both methods with twice redundant DFT. The A-SPADE method is 10% to 50% faster than the structured sparse version in this case. Except for the lowest input SDR, the structured cosparse method is 5% to 25% faster than A-SPADE. For 5dB input SDR, we observe that substantial improvements given by the multichannel algorithm are achieved at the cost of only slightly lower computational efficiency (10% to 30% slower than A-SPADE). These different computation time characteristics might come from the properties of the sparsifying operator when used inside the ADMM framework and the total number of iterations needed to finish or converge. Similarly to the single-channel methods we emphasize that using multi-threading and a different parametrization (especially reducing α) allows (sub) real-time processing.

Stereo declipping For these last declipping experiments, we consider the channel-aware (co)sparse methods from the framework as well as A-SPADE. We keep unchanged the common parameters as stated for the previous 8-channel experiments. Here we compare efficiency of the Group Empirical Wiener (GEW) and Quadratic Group Empirical

Wiener (Quad-GEW) shrinkages on large scale simultaneous stereo declipping (the RWC Jazz subset). Figure 6.20 displays averaged ΔSDR results compared to those of A-SPADE based on a simple cosparse prior independently on each channel. Results below are presented only for twice-redundant DFT.



(a) Channel 1 (b) Channel 2

Figure 6.20 – RWC Jazz: Stereo declipping (Numerical results Δ SDR [dB])

Notably, Figure 6.20 shows SDR improvements for all tested methods and for all degradation conditions. From 10 dB input SDR, both methods embodying channel-wise structured (co)sparse priors outperform A-SPADE by 1dB to 4 dB. Below, the Quadratic Group Empirical Wiener shrinkage only seems to yield better results still above A-SPADE but to a lesser extent (by 0.5 dB to 2 dB). SDR improvements do not appear to drastically depend on the analysis or synthesis sparse prior, however, for the two lowest input SDRs, the Quad-GEW synthesis sparse method provides better improvement. We gathered standard deviation results which, along with the averaged improvement, increases with the input SDR (up to more than 8 dB for A-SPADE with the 30 dB input SDR condition). Finally, we notice that the results are consistent from one channel to another. Both audio quality descriptors and SDR improvements legitimate this channel-wise structured (co)sparse declipping method for stereo to more channels saturated audio recordings. A validation on a wider number of stereo sounds excerpts (*i.e.* the other categories of the RWC database) could be envisioned for further work.

Figure 6.21 describes averaged quality improvement measured with the PEAQ descriptor. What is interesting to see here is that as opposed to the ΔSDR results, the channel-aware declippers seems to give worse results for 15 dB input SDR and above. We also note that the new methods presented here worsen audio quality for the two lightest clipping conditions. These results should be considered carefully due to high variability. Indeed, highest standard deviation on Δ PEAQ are obtained for 15 dB input SDR and above (around 1.0). For 10 dB input SDRs and below, it appears to be slightly advantageous to use the cosparse method with the Quadratic Empirical Wiener shrinkage. Finally, informal listening tests does not reveal striking quality differences between

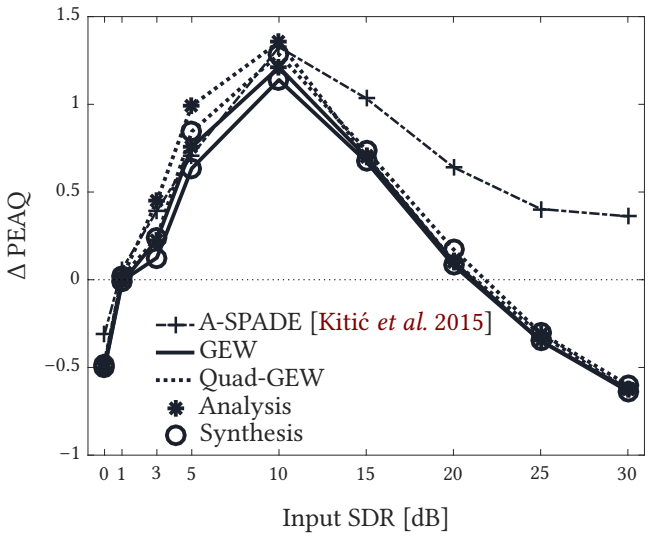


Figure 6.21 – RWC Jazz: Stereo declipping (Quality results Δ PEAQ)

methods. These interesting differing results between audio quality index and ΔSDR for light clipping conditions should be further studied trough subjective listening experiments.

6.5 Summary

This chapter presented several instances of the common algorithmic framework to tackle the declipping problem either in the mono- or multi-channel case. For the monochannel case, in terms of quality, our detailed study of the (social) (co)sparse versions for declipping shows SDR improvements consistently exceeding 8dB for various types of speech and music and a wide range of saturation levels. The only notable exception is the Pop dataset, possibly due to the presence of dynamically compressed drums and saturated guitar sounds. SDR improvements are observed that are either on par with or better than what state-of-the-art methods can achieve, especially for low input SDRs.

This chapter also showed that adding across-channel structure on top of (co)sparse modeling was bringing considerable reconstruction improvements compared to a cosparsity based state-of-the-art method applied channel-wise. This was verified from stereo to more channel audio recordings. In addition, we showed that performance can be improved by the use of a redundant frequency transform when the saturation is moderate. Finally, we demonstrated that the method implies a very limited runtime overhead or is faster compared to a state-of-the-art method using simple cosparsity data model ([Kitić *et al.* 2015]). Future studies could include perceptual assessments, and model integration of time-frequency structures on top of structured (co)sparsity across channels. The following list gives some guidelines for declipping audio with the methods presented in this chapter.

Declipping guidelines

- Single-channel
  - Music
    For music, at *low input SDR*, the **plain cosparse** method seems a good option. For *high input SDR*, one can choose the **adaptive social sparse** method.
  - Speech
    For speech, at *low input SDR*, the **plain cosparse** declipper should be chosen for quality. For intelligibility improvements, the **plain sparse** method seems a better choice. At *high input SDR*, **plain** and **adaptive social sparse** declippers should be preferred.
- Multi-channel
  - Music
    At *low input SDR*, structured (co)sparse methods with **Quad-GEW shrinkage** seems to be the best choice. For *high input SDR*, structured (co)sparse methods with **GEW shrinkage** should be preferred.
  - Speech
    At *low* and *high input SDR*, as long as one considers using redundant DFT, both structured sparse and cosparse methods using **GEW shrinkage** are good choices.

Part III

Handling propagation in acoustic inverse problems

Chapter 7

Dereverberation

Contents

This chapter will focus on the reconstruction of audio signals corrupted by reverberation in the sound propagation environment. After briefly describing the reverberation issue, the chapter will feature a short review of available dereverberation techniques. Then, we introduce applications of (structured) sparsity for a monochannel reconstruction scenario emphasizing an simple sparse or social sparse time-frequency modeling. Each of these two applications is an instance of the generic framework introduced earlier (chapter 3). Before concluding this current chapter, some experiments comparing the two models will be presented on speech examples.

This chapter presents the dereverberation problem in the light of an inverse problem and compares simple sparse and social sparse time-frequency models. This work is not derived from any publication and rather give a proof of concept that the dereverberation problem can be included in the generic framework than thoroughly studying the issue.

7.1 Reverberation and room compensation

This section first presents briefly indoor sound propagation and its consequence: reverberation. Then, we describe some prior art methods used to handle this problem.

7.1.1 The reverberation problem

When sound propagates in a closed environment from a source to a sensor the recorded signal at the microphone is the resulting mixture of the direct sound, the early reflections and late reverberation [Kuttruff 2009]. Figure 7.1 displays a schematic view of such a scenario with the source being someone speaking ("Speaker"). The early reflections are due to the first reflections on the walls and can also be referred to as "first order echoes". The late reverberation is related to the multiple paths of the sound due to higher orders echoes and diffusion on objects like furniture for example. Early reflections can be useful for improving speech intelligibility [Arweiler & Buchholz 2011] or for source localization [Kitić *et al.* 2014]. On the contrary, late reverberation can be detrimental for speech quality, intelligibility [Warzybok *et al.* 2013] and systems designed for automatic speech recognition [Yoshioka *et al.* 2012]. As an example, Figure 7.2a displays an anechoic speech signal and Figure 6.3b shows its STFT representation. We see on Figure 7.2c the STFT of the same speech occurrence recorded in a relatively reverberant room. By the observation that the frequency content and the time onsets appear smeared, understandably, reverberation has a non negligible impact on sound. The process of propagation between the source and the microphone is described by the impulse response often called Room Impulse Response (RIR). This term can be ambiguous as the RIR does not directly describe the room itself but rather the multiple acoustic paths between a given source and a given sensor in a room. If the following, we will refer to the RIR equivalently by "acoustic channel" or "(acoustic) filter". Denoting $\mathbf{h} \in \mathbb{R}^M$ a RIR, $\mathbf{x} \in \mathbb{R}^L$ an anechoic signal, the corresponding reverberant signal $\mathbf{y} \in \mathbb{R}^L$ can be obtained by the following convolution process:

$$\mathbf{y} = \mathbf{h} \star \mathbf{x}. \tag{7.1}$$

with $\star$ modeling the convolution.

The amount of reverberation in a room is often characterized by the reverberation time (RT_{60}) which is the amount of time needed for the acoustic energy to decay by 60 dB after a steady-state excitation source stops. The larger is the RT_{60} the more reverberant will be the room. We will use this index to rate the effectiveness the reverberation in the experimental section (section 7.3).

7.1.2 Prior art on audio dereverberation

Audio dereverberation, also called in the literature by "room equalization", "room inversion" or sometimes "room compensation" is the process of designing signal processing techniques able to retrieve the direct sound from a source to a microphone in closed

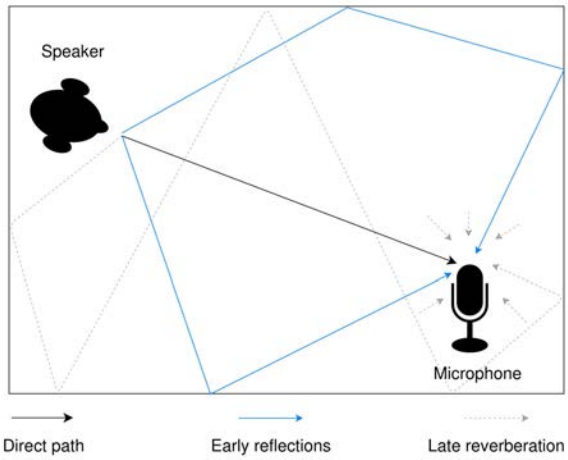
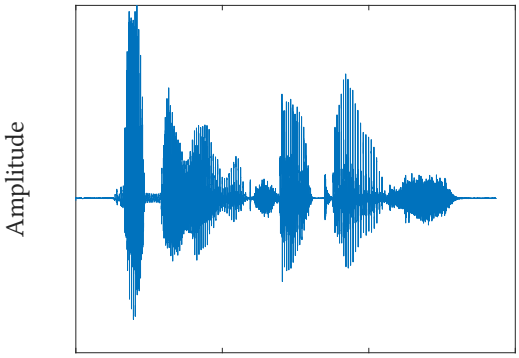
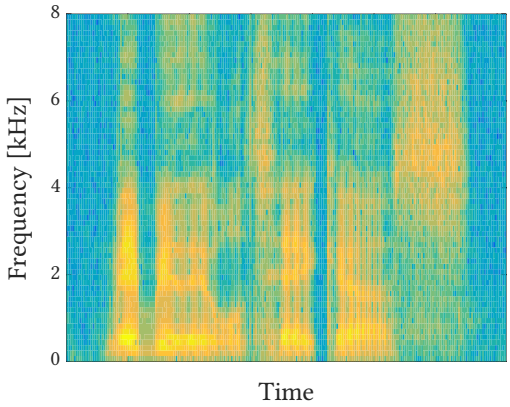


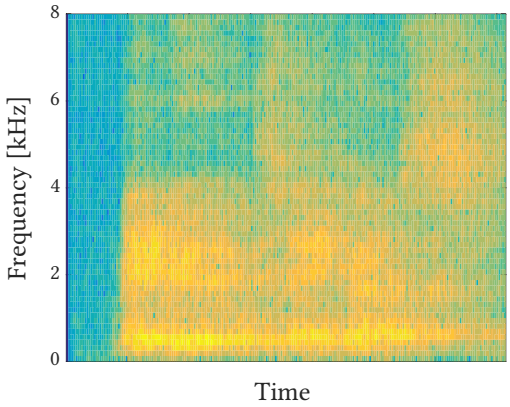
Figure 7.1 – Indoor sound propagation



(a) Signal waveforms



(b) Frequency transform (original speech)



(c) Frequency transform (reverberant speech)

Figure 7.2 – Reverberated speech example

environments recordings. In practice it often boils down to alleviating the effects of the room (reflections) on the recordings. This is a blind problem if the acoustic path (Room Impulse Response, RIR) between the source and the sensor(s) is unknown. Two main categories of methods have been used for single-channel dereverberation.

Time-frequency enhancement As single-channel dereverberation is mainly used for speech enhancement, some methods share the same line of work than noise reduction techniques. For instance, spectral subtraction [Boll 1979] which was initially applied for noise reduction is adapted to work for dereverberation. Denoising relies on an estimation of the noise spectrum to "remove" it by subtraction from a (time)-frequency transform of a noisy signal. Spectral subtraction for dereverberation works the same way but relying on an estimation of the spectral content due to the late reverberations [Lebart *et al.* 2001].

Inverse filtering Methods sharing the acoustic channel inversion principle need either a measurement or a good estimation of the RIR as the basic idea is to perform deconvolution from the RIR to get the original source. However, even if the RIR is perfectly known, the inverse problem is still ill-posed as direct inversion of the acoustic filter is not straightforward. Indeed, for realistic wall sound absorption values the RIR is non-minimum phase [Neely & Allen 1979] hence, in the single source case no exact inverse RIR can be derived. A direct inversion would cause instability at high frequencies [Kaipio & Somersalo 2006]. Several techniques were investigated to alleviate the artifacts introduced by the mixed-phase inverse filter. Among them we note least squares minimization [Mourjopoulos *et al.* 1982] or homomorphic filtering [Radlovic & Kennedy 2000] just to cite a few (an exhaustive review of Room Response Equalization can be found in [Cecchi *et al.* 2018]). More recently, we note a method [Kodrasi *et al.* 2014] using a frequency-domain inverse filtering technique coupled with speech enhancement post-processing. This frequency-domain technique rely on the narrow-band approximation for the convolution and allows to rewrite Equation (7.1) after applying the STFT on both sides such that:

$$\bar{\mathbf{Y}} = \mathbf{H} \times \mathbf{Z}, \tag{7.2}$$

and $\bar{\mathbf{Y}} \in \mathbb{C}^{\mathsf{F}\times\mathsf{T}}$ is the STFT representation of the reverberant signal, $\mathbf{Z} \in \mathbb{C}^{\mathsf{F}\times\mathsf{T}}$ is the STFT representation of the initial anechoic signal $\mathbf{x}$ and $\mathbf{H} = \mathrm{diag}(H(f))$ is the corresponding frequency response of the RIR $\mathbf{h}$.

This approximation usually holds when most of the RIR energy is concentrated in a time scale comparable with the STFT window. This assumption is no longer verified for relatively reverberant filters. In the context of multichannel deconvolution novel approach [Kowalski *et al.* 2010] was proposed some years ago considering minimization of convex sparse promoting function and a wide-band data-fidelity term modeling the time domain convolution process rather than the narrow-band frequency-wise product approximation (Equation (7.2)). This method was successfully applied on quite reverberant mixtures. For this reason, the next section will present a dereverberation method using the same wide-band data-fidelity term but different sparse models.

7.2 Sparsity for audio dereverberation

In the following section we introduce a method for audio dereverberation that integrates in the algorithmic framework presented in chapter 3. This method embodies time-frequency regular or structured (social) sparse data models and wide-band modeling of the convolution process. Our goal here is mainly to compare the validity of the signal models, so we consider the informed case where the acoustic channel $\mathbf{h}$ is known. Indeed, the estimation of the filter is already a complex task and RIR mismatch could alter the dereverberation performance.

After listing the required projection operator, we instantiate the different versions of Algorithm 1. Similarly to denoising and declipping, we consider the matrix $\mathbf{Y} \in \mathbb{R}^{T\times L}$ containing T windowed frames of L samples from the observed signal $\tilde{\mathbf{y}}$. The dereverberation problem here consists in estimating the original clean signal frames, similarly gathered in a matrix $\mathbf{X}$ of the same size.

Denote $\mathcal{A}(\cdot) : \mathbb{R}^{L\times T} \mapsto \mathbb{R}^{L\times T}$ the operator performing the convolution process such that Equation (7.1) rewritten for a matrix of time frames $\mathbf{Y}$ gives:

$$\text{vec}(\mathbf{Y}) = \mathbf{h} \star \text{vec}(\mathbf{X}) = \text{vec}(\mathcal{A}(\mathbf{X})). \tag{7.3}$$

We recall that vec$(\cdot)$ is the matrix vectorization operator. Finally, the adjoint convolutional operator $\mathcal{A}(\cdot) : \mathbb{R}^{L\times T} \mapsto \mathbb{R}^{L\times T}$ defines the operator performing convolution with the time-reversed filter.

7.2.1 Generalized projection for the dereverberation problem

Similarly to [Kowalski *et al.* 2010] and [Arberet *et al.* 2013] the data-fidelity constraint is of the form $\mathcal{A}(\hat{\mathbf{X}}) = \mathbf{Y}$.

We consider only the synthesis setting, with $\mathbf{M} := \mathbf{I}$, we set $\Theta := \{\mathbf{W} \,|\mathcal{A}(\mathbf{DW}) = \mathbf{Y}\}$. These choices hold both for plain and structured versions.
As in [Kowalski *et al.* 2010], we define the operator $\mathcal{T} : \mathbb{C}^{S\times T} \mapsto \mathbb{R}^{L\times T}$ such that:

$$\mathcal{T}(\mathbf{Z}) = \mathcal{A}(\mathbf{DZ}), \tag{7.4}$$

and the corresponding adjoint operator $\mathcal{T}^* : \mathbb{R}^{L\times T} \mapsto \mathbb{C}^{S\times T}$ defined as:

$$\mathcal{T}^*(\mathbf{Y}) = \mathbf{D}^H(\mathcal{A}^*(\mathbf{Y})). \tag{7.5}$$

Assuming $\mathbf{DD}^H = \mathbf{I}$, the generalized projection reduces to:

$$\mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z}) = \mathbf{Z} - \frac{1}{t}\mathcal{T}^*(\mathbf{Y} - \mathcal{T}(\mathbf{Z})) \tag{7.6}$$

Note that t is the highest absolute singular value associated to $\mathcal{T}^*\mathcal{T}$ and is computed with the power iteration algorithm as in [Kowalski *et al.* 2010] and [Arberet *et al.* 2013] (see:

Appendix B). With all the steps defined, we can now instantiate the general algorithm $\mathcal{G}$ in the different cases.

7.2.2 Wide-band Plain/Social sparsity dereverberation

Similarly to denoising or declipping, we instantiate the general algorithm $\mathcal{G}$ by choosing the operators described in Table 7.1.

Table 7.1 – Parameters of Algorithm 1 for the wide-band sparse dereverberation method

Plain	**Social**
$\Theta := \{\mathbf{W} \mid \mathcal{A}(\mathbf{DW}) = \mathbf{Y}\}$	$\Theta := \{\mathbf{W} \mid \mathcal{A}(\mathbf{DW}) = \mathbf{Y}\}$
$\mathbf{M} = \mathbf{I} \in \mathbb{C}^{\mathsf{L}\times\mathsf{L}}, \mathbf{D} \in \mathbb{C}^{\mathsf{L}\times\mathsf{S}}$	$\mathbf{M} = \mathbf{I} \in \mathbb{C}^{\mathsf{L}\times\mathsf{L}}, \mathbf{D} \in \mathbb{C}^{\mathsf{L}\times\mathsf{S}}$,
$\mathcal{S}_\mu(\cdot) = \mathcal{H}_{\mathsf{S}-\mu}(\cdot)$	$\mathcal{S}_\mu(\cdot) = \mathcal{S}_\mu^{\mathrm{PEW}}(\cdot\|\Gamma)$,
$\mu^{(0)} = \mathsf{S} - 1$	$\mu^{(0)}$: see section 7.3
$F : \mu \mapsto \mu - 1$	$F = F_\alpha : \mu \mapsto \alpha\mu$
$\mathbf{Z}^{(0)} = \mathbf{D}^{\mathsf{H}}\mathbf{Y}$	$\mathbf{Z}^{(0)} = \mathbf{D}^{\mathsf{H}}\mathbf{Y}$

7.3 Experiments

For this experimental section, first results on speech dereverberation will be presented to compare the two methods presented above.

Compared methods We consider here the plain sparse and social sparse methods. We set the common parameters for the algorithms as listed below.

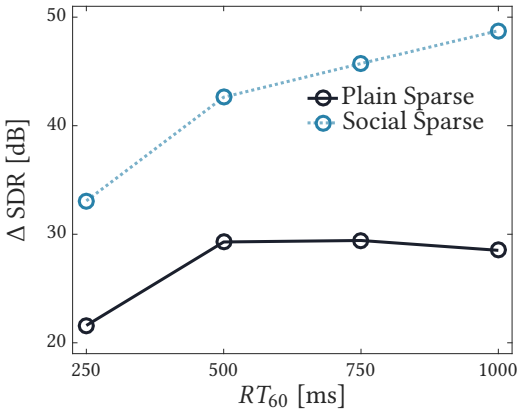
- Frame size L = 32 ms;
- Overlap, 75%;
- $\mathsf{i}_{\max} = 10^6$;
- Synthesis operator, $\mathbf{D}$ = twice redundant inverse DFT;
- Accuracy, $\beta = 10^{-3}$.

Considering the social sparse method, we set the time-frequency pattern Γ to match the one presented on Figure 5.6b with the intuition that it will provide better results as it was shown to be useful in avoiding pre-echo artifacts in [Siedenburg & Dörfler 2012]). The choice of $\mu^{(0)} := \mathsf{L}$ and $\alpha = 0.99$ are similar to those used in the multichannel declipping methods.

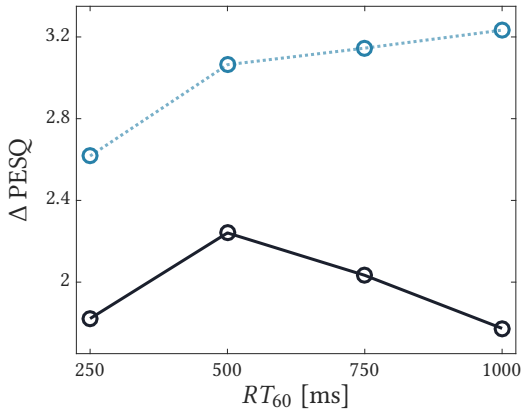
To evaluate the different sparse modelings, we use the shrinkages (Hard Thresholding and PEW). We also set the room's RT_{60} to account for different degrees of audio

degradation. For that, we consider four RT_{60} in ms: {250, 500, 750, 1000}. Each speech excerpt from TIMIT is convolved with the corresponding room filter of every tested configuration.

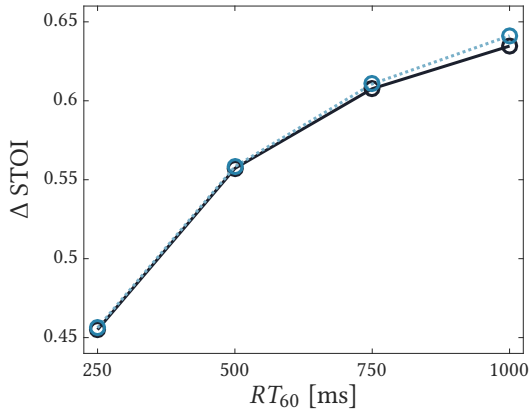
Comparison of dereverberation performance SDR improvement results are presented on Figure 7.3a and speech quality improvements (ΔPESQ) are presented on Figure 7.3b. Finally, intelligibility improvements are shown on Figure 7.3c. Plain dark lines displays averaged results for the synthesis plain sparse method while blue dashed lines presents results for the social sparse method. We note that the social sparse method performs 10 dB to 20 dB better in the most reverberant case than the plain sparse method in terms of SDR improvement. We gathered standard deviation results which are slightly lower for the plain sparse version (ranging from 1.85 dB to 2.07 dB) than the social sparse version (ranging from 2.06 dB to 3.28 dB). For quality measurements, results are here also in favor of the social version especially for the most reverberant condition. In contrast, intelligibility results do not show any superiority of one or another method.



(a) Numerical reconstruction results (ΔSDR)



(b) Objective quality results (ΔPESQ)



(c) Objective intelligibility results (ΔSTOI)

Figure 7.3 – TIMIT: Speech numerical results (Dereverberation)

7.4 Summary

In this short chapter, we showed that the algorithmic framework could be easily extended to address the single-channel dereverberation problem. Results demonstrated that the wide-band data-fidelity term for convolution along with sparse models were able to handle the speech dereverberation inverse problem for moderately to highly reverberant environments. However, it must be noted that, as we considered only the non-blind case, further investigations are needed for instance to rate the robustness of the method to errors in the filter. Extension to the multichannel case and promoting different sparse structures thanks to the PEW could also be an interesting axis.

Chapter 8

Binaural sound source localization

Contents

Human listeners have the stunning ability to understand complex auditory scenes using only two ears, *i.e.* , with binaural hearing. Advanced tasks such as sound source direction and distance estimation or speech deciphering in multi-source, noisy and reverberant environments are performed daily by humans, while they are still a challenge for artificial (two-microphone) binaural systems. After presenting what has already been done for binaural machine hearing, this chapter will introduce a new method to address the binaural sound source localization inverse problem.

From section 8.5, this chapter is mainly inspired from the following publications: [Gaultier *et al.* 2017b]: Clément Gaultier, Saurabh Kataria and Antoine Deleforge. *VAST: The Virtual Acoustic Space Traveler dataset.* In International Conference on Latent Variable Analysis and Signal Separation, pages 68–79. Springer, 2017 and [Kataria *et al.* 2017]: Saurabh Kataria, Clément Gaultier and Antoine Deleforge. *Hearing in a shoe-box: Binaural source position and wall absorption estimation using virtually supervised learning.* In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 226–230, March 2017.

8.1 Source localization

The sound source localization problem can be many-fold whether we consider long range sound propagation or sound inside enclosed environments. In the following we will consider sound propagating from a source to sensors in indoor environments (like a room). Hence, the problem we will try to address here is the estimation of sound source position in a room. More precisely, we will examine the binaural sound source localization issue: estimating the position like we humans do it with our two ears (*i.e.* with only two sensors and a pair of transfer functions related to our head and torso morphology named Head Related Transfer Function, HRTF). Figure 8.1 gives a schematic view of such a problem. In this simplified 2D scenario, localizing the source boils down to estimating the azimuth angle θ and the distance to the sensor r. In a 3D scenario, we can add to that the elevation angle between the source and the sensors.

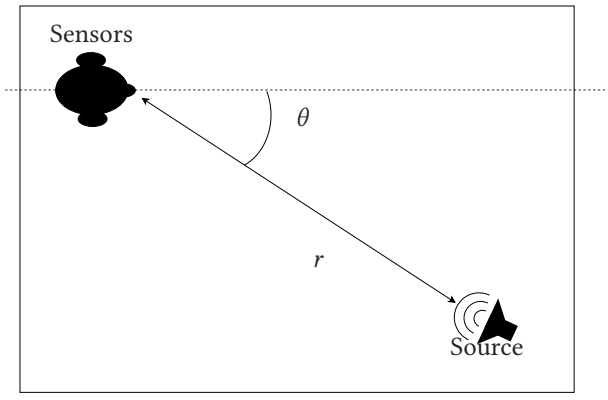


Figure 8.1 – 2D indoor sound source localization problem

Most of the localization methods involving two (or more) microphones make use of the time shifts with which the sound source signal arrives at the different microphones to predict the direction of the source (θ on Figure 8.1). This time is called Time-Difference Of Arrival (TDOA). In the next section, we present some methods and how they use this TDOA for (binaural) source localization.

8.2 Prior art

The main line of research in machine binaural source localization along the past decades has been to estimate the Time-Difference Of Arrival (TDOA) of the signal of interest at the two microphones. An estimated TDOA can be mapped to the azimuth angle of a frontal source if the distance between microphones is known, assuming free-field and far-field conditions. Free-field means that the sound propagates from the source to the microphones through a single direct path, without interfering objects or reverberations. Far-field means that the source is placed far enough from the receiver so that the effect of distance on recorded audio features is negligible. Far-field is usually described by the Faunhofer zone for electromagnetic waves for example, were the wave front is considered as planar. Figure 8.2 illustrates a binaural azimuthal plane with far-field / close-field conditions. Considering the far field assumption (Figure 8.2a), if we denote τ the TDOA between the two microphones of the head (ears), the Direction Of Arrival (DOA) θ can be estimated geometrically with:

$$\theta = \arccos\left(\frac{c \cdot \tau}{d}\right), \tag{8.1}$$

with c the sound speed and d the distance between the two ears. Notably, in the context of binaural hearing, TDOA is equivalently refereed to as the Interaural Time Difference (ITD). We also find Interchannel Time Difference in the general multi-sensor context.

It is important to remark that for a fixed distance between sensors (d), θ only depends on τ. Therefore, when sensor array geometry is known, localization methods using goniometry rely on an estimation of the TDOA. Below are presented some widely used methods for TDOA estimation.

Estimating the TDOA in the time-domain A standard way to estimate the TDOA between to microphones is to maximize the cross-correlation function of the two signals retrieved at the sensors. Let one consider $\mathbf{y}_1 \in \mathbb{R}^N$ and $\mathbf{y}_2 \in \mathbb{R}^N$ discrete time signals measured at microphone 1 and 2 in the binaural setting of Figure 8.2, the cross-correlation (CC) function is defined as:

$$r_{\mathbf{y}_1,\mathbf{y}_2}(\tau) = \sum_{n=1}^{N} \mathbf{y}_1(n)\mathbf{y}_2(n-\tau). \tag{8.2}$$

The estimated time-delay $\hat{\tau}$ between $\mathbf{y}_1$ and $\mathbf{y}_2$ is then deduced from:

$$\hat{\tau} = \underset{\tau}{\operatorname{argmax}}\, r_{\mathbf{y}_1,\mathbf{y}_2}(\tau). \tag{8.3}$$

While this estimate is easy to compute, it is highly sensitive to noise and gain differences between $\mathbf{y}_1$ and $\mathbf{y}_2$. To alleviate this issue, an option is to maximize the Normalized Cross Correlation (NCC) function instead. However, as state-of-the art methods are based on frequency-domain methods to estimate the TDOA, we will note detail NCC here.

Estimating the TDOA in the frequency-domain A more flexible way of estimating the time delays between microphones than the classical time-domain cross correlation is Generalized Cross Correlation (GCC). A time delay can be identified with a phase-shift in the frequency domain. Equivalently to cross correlation, the cross Power Spectral Density (cross-PSD) is defined in the frequency domain. TDOA can then be estimated in the frequency domain by the phase shift which maximizes the cross-PSD function between two signals over all the frequencies. More formally, if we denote $\mathbf{Z}_1 \in \mathbb{C}^{\mathsf{F}\times\mathsf{T}}$ and $\mathbf{Z}_2 \in \mathbb{C}^{\mathsf{F}\times\mathsf{T}}$ the STFT over T time frames of the previous $\mathbf{y}_1$ and $\mathbf{y}_2$ signals, the cross-PSD writes for a given frequency f:

$$\Psi_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathrm{f}) = \frac{1}{\mathsf{T}} \sum_{\mathrm{n}=1}^{\mathsf{T}} \mathbf{Z}_1(\mathrm{f},\mathrm{n})\mathbf{Z}_2^*(\mathrm{f},\mathrm{n}). \tag{8.4}$$

The cross-correlation function in the frequency domain results then from Equation (8.4):

$$R_{\mathbf{Z}_1,\mathbf{Z}_2}(\tau) = \sum_{\mathrm{f}=1}^{\mathsf{F}} \Psi_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathrm{f}) \exp(-j2\pi\mathrm{f}\tau). \tag{8.5}$$

What is interesting working in the frequency domain is that the initial signals $\mathbf{y}_1$ and $\mathbf{y}_2$ can be easily prefiltered. Knowing the frequency responses ($H_1(\mathrm{f})$ and $H_2(\mathrm{f})$) of the filters applied to $\mathbf{y}_1$ and $\mathbf{y}_2$ leads to:

$$\begin{cases} \widetilde{\mathbf{Z}}_1 = \mathbf{H}_1\mathbf{Z}_1 \\ \widetilde{\mathbf{Z}}_2 = \mathbf{H}_2\mathbf{Z}_2 \end{cases} \tag{8.6}$$

with $\widetilde{\mathbf{Z}}_1 \in \mathbb{C}^{\mathsf{F}\times\mathsf{T}}$ and $\widetilde{\mathbf{Z}}_2 \in \mathbb{C}^{\mathsf{F}\times\mathsf{T}}$ the frequency transforms of the prefiltered signals and $\mathbf{H}_i = \mathrm{diag}(H_i(\mathrm{f}))$. The cross-correlation function adapts then to:

$$\tilde{R}^G_{\mathbf{Z}_1,\mathbf{Z}_2}(\tau) = R_{\widetilde{\mathbf{Z}}_1,\widetilde{\mathbf{Z}}_2}(\tau) = \sum_{\mathrm{f}=1}^{\mathsf{F}} G(\mathrm{f})\Psi_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathrm{f}) \exp(-j2\pi\mathrm{f}\tau). \tag{8.7}$$

with $G(\mathrm{f}) = H_1(\mathrm{f})H_2(\mathrm{f})$ the filter (frequency weighting) applied to the initial signals. Equation (8.3) becomes for frequency domain TDOA estimation:

$$\hat{\tau} = \underset{\tau}{\mathrm{argmax}}\, \tilde{R}^G_{\mathbf{Z}_1,\mathbf{Z}_2}(\tau). \tag{8.8}$$

Among the widely used weighting functions $G(\mathrm{f})$, the PHAse Transform (PHAT) function is probably the most common:

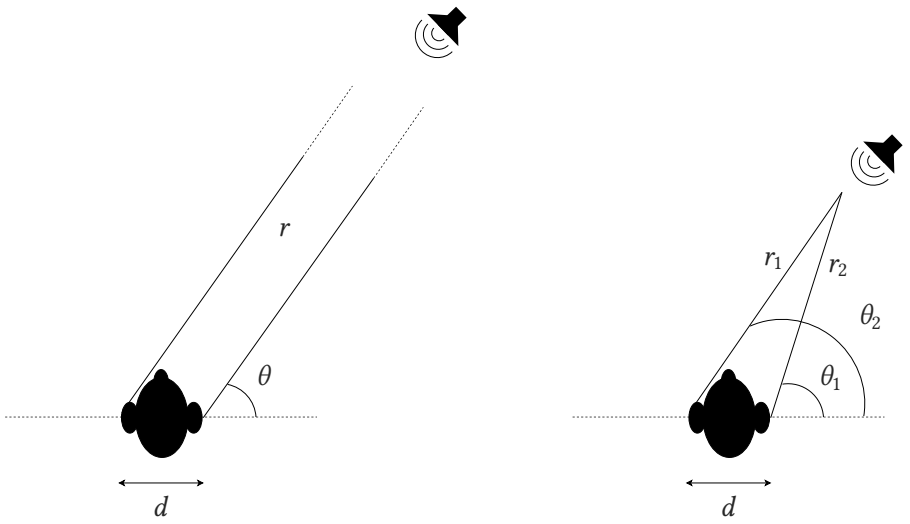
$$G_{\mathrm{PHAT}}(\mathrm{f}) = \frac{1}{|\Psi_{\mathbf{Z}_1,\mathbf{Z}_2}(\mathrm{f})|}. \tag{8.9}$$

This PHAT transform was shown to be efficient for TDOA estimation in reverberant environments [Brandstein & Ward 2001] as it leads only the phase information to be taken

into account in the computation of $\tilde{R}^{G}_{Z_1,Z_2}(\tau)$. If the GCC-PHAT method performs well in reverberant condition, it is more likely to fail in noisy conditions and especially when the noise PSD differs between the two microphone signals. To overcome this problem, other appropriate filtering functions $G(f)$ can be preferred among which the Smooth COherent Transform (SCOT) defined as follows:

$$G_{\text{SCOT}}(f) = \frac{1}{\sqrt{\Psi_{Z_1,Z_1}(f) \cdot \Psi_{Z_2,Z_2}(f)}}. \tag{8.10}$$

Limits Even though the aforementioned TDOA-based methods yield correct results for speaker localization for example [DiBiase *et al.* 2001], two important limits of the underlying assumption on the acoustic field (free field & far-field) can be identified. First, these assumptions are both violated in most practical scenarios. In the example of an indoor binaural hearing robot, users are typically likely to engage interaction in both far- and near-field, and non-direct sound paths exist due to reflections and diffusion on walls, ceiling, floor, other objects in the room and the robot itself. Second, the intrinsic symmetries of a free-field/far-field binaural system restrict any geometrical estimation to that of a frontal azimuth angle. Hence, 3D source position (azimuth, elevation, distance) is out of reach in this scope, let alone additional properties such as source orientation, receiver position or room shape. Finally, one has to note that some other work focusing on subspace methods such as MUSIC or later ESPRIT were used for sound localization in an array processing perspective for robot hearing (a review is available in [Argentieri *et al.* 2015]).



(a) Far-field assumption (b) Close-field assumption

Figure 8.2 – Acoustic field assumptions for TDOA localization methods

To overcome intrinsic limitations of TDOA, richer binaural features have been investigated. These include frequency-dependent phase and level differences [Viste &

Evangelista 2003, Deleforge *et al.* 2015c], spectral notches [Raykar *et al.* 2005, Hornstein *et al.* 2006] or the direct to reverberant ratio [Lu & Cooke 2010]. To overcome the free-field/far-field assumptions, advanced mapping techniques from these features to audio scene properties have been considered. These mapping techniques divide in two categories. The first one is *physics-driven*, *i.e.*, the mapping is inferred from an approximate sound propagation model such as the Woodworth's spherical head formula [Viste & Evangelista 2003], its extensions [Aaronson & Hartmann 2014], or the full wave-propagation equation [Kitić *et al.* 2014]. The second category of mapping is *data-driven*. This approach is sometimes referred to as *supervised sound source localization* [Talmon *et al.* 2011], or more generally *acoustic space learning* [Deleforge *et al.* 2015a]. These methods bypass the use of an explicit, approximate physical model by directly learning a mapping from audio features to audio properties using manually recorded training data [Talmon *et al.* 2011, Deleforge *et al.* 2015c]. They generally yield excellent results, but because obtaining sufficient training data is very time consuming, they only work for a specific room and setup and are hard to generalize in practice. Unlike artificial systems, human listeners benefit from years of adaptive auditory learning in a multitude of acoustic environments. While machine learning recently showed tremendous success in the field of speech recognition using massive amounts of annotated data, equivalent training sets do not exist for audio scene geometry estimation, with only a few specialized manually annotated ones [Deleforge *et al.* 2015a, Deleforge *et al.* 2015c]. Interestingly, a recent data-driven method [Parada *et al.* 2016] used both real and simulated data to estimate room acoustic parameters and improve speech recognition performance, although it was not designed for sound localization.

8.3 Virtually supervised learning

This chapter proposes here a new paradigm that aims at making the best of physics-driven and data-driven approaches, referred to as *virtual acoustic space learning*. The idea is to use a physics-based room-acoustic simulator to generate arbitrary large datasets of room-impulse responses corresponding to various acoustic environments, adapted to the physical audio system considered. Such impulse responses can be easily convolved with natural sounds to generate a wide variety of audio scenes including *cocktail-party* like scenarios. The obtained corpus can be used to learn a mapping from audio features to various audio scene properties. The *virtually-learned* mapping can then be used to efficiently perform real-world auditory scene analysis tasks with the corresponding physical system. Inspired by the idea of an artificial system learning to hear by exploring virtual acoustic environments, this proposal was named the *Virtual Acoustic Space Traveler* (VAST) project. We initiated it by publicly releasing a dedicated project page[1] and a first example of VAST dataset.

[1] http://theVASTproject.inria.fr

In order to perform virtually supervised learning one needs a training dataset to learn a mapping between a set of high dimensional audio data ($\{\mathbf{y}_n\}_{n=1}^N \in \mathbb{R}^D$) and low dimensional audio scene properties ($\{\mathbf{u}_n\}_{n=1}^N \in \times\mathbb{R}^L$). We consider a training dataset composed of N pairs $\{(\mathbf{y}_n, \mathbf{u}_n)\}_{n=1}^N \in \mathbb{R}^D \times \mathbb{R}^L$, $L \ll D$. A mapping needs to be learned from this dataset such that given a new test observation $\tilde{\mathbf{y}}_t \in \mathbb{R}^D$, an associated parameter vector $\tilde{\mathbf{u}}_t$ can be estimated.

8.4 Gaussian Locally Linear Mapping

In our line of work, we use the high- to low-dimensional regression method *Gaussian locally-linear mapping* (GLLiM) proposed in [Deleforge *et al.* 2015b]. GLLiM was successfully applied to supervised 2D sound source localization on a real dataset in [Deleforge *et al.* 2015c]. The next two paragraphs will present the basic principles of (multivariate) linear regression underlying GLLiM.

Linear regression If one considers the previous set $\{(\mathbf{y}_n, \mathbf{u}_n)\}_{n=1}^N \in \mathbb{R}^D \times \mathbb{R}^L$, performing a regular (forward) linear regression between $\{\mathbf{u}_n\}_{n=1}^N \in \mathbb{R}^L$ and $\{\mathbf{y}_n\}_{n=1}^N \in \mathbb{R}^D$ corresponds to finding the affine transformation ($\mathbf{u}_n = \mathbf{A}\mathbf{y}_n + \mathbf{b}$, $\mathbf{A} \in \mathbb{R}^{L\times D}$, $\mathbf{b} \in \mathbb{R}^L$ and $D \gg L$).
One drawback of performing such a forward (low to high) regression is that it requires to estimate $\mathbf{A}$ and $\mathbf{b}$ (*i.e.* $L(D+1)$ coefficients) from LN equations. To invert this system we would need at least $N = D + 1$ available training pairs. D reflects to what extent the high dimension feature vectors $\{\mathbf{y}_n\}_{n=1}^N$ encompass initial information. One has to keep in mind that D could be possibly large and at the same time N.

One solution to overcome this situation is to perform the linear regression the other way around from the high dimension vectors to the low dimensional data. This changes to finding the affine transformation ($\mathbf{y}_n = \mathbf{A}^*\mathbf{u}_n + \mathbf{b}^*$, $\mathbf{A}^* \in \mathbb{R}^{D\times L}$, $\mathbf{b}^* \in \mathbb{R}^D$ and $D \gg L$).
Here performing the high-to-low (inverse) regression requires the estimation of $\mathbf{A}^*$ and $\mathbf{b}^*$ (*i.e.* $D(L+1)$ coefficients) from DN equations. This system is invertible if at least $N = L+1$ equations are available. In that case, we recall that L is the dimension of the audio scene properties, intrinsically much smaller than D. This "reversed" linear regression makes it easier to find a solution as it requires theoretically a significantly lower number of training pairs. The learning method GLLiM [Deleforge *et al.* 2015b] that we use later on to apply the VAST concept is based on this "reversed" linear regression principle as well as piecewise linear modeling presented below.

Piece-wise linear regression One limitation of the inverse linear regression presented above is that it is more likely to fail if the relationship between the high dimension vectors and the low dimension features of the dataset can not be modeled with just a single affine transformation (a single $\mathbf{A}^*$ and a single $\mathbf{b}^*$). One solution used in [Deleforge *et al.* 2015b] and further developed in [Perthame *et al.* 2018] is to consider *piecewise* affine transformations between the high and low dimension training data. Denote the pairs $\{(\mathbf{y}_n, \mathbf{u}_n)\}_{n=1}^N \in \mathbb{R}^D \times \mathbb{R}^L$ any realization of random variables Y and U. We no longer

consider a single possible transformation but a mixture of K local affine transformations from the space of U to the space of Y modeled by

$$Y = \sum_{k=1}^{K} i_c(z = k)(\mathbf{A}_k^* U + \mathbf{b}_k^*) \tag{8.11}$$

where $\mathbf{A}_k^* \in \mathbb{R}^{D \times L}$ and $\mathbf{b}_k^*$ embody the k^{th} affine relationship coefficients. $i_c(z = k)$ is the indicator function such that $i_c(z = k) = 1$ when $z = k$ and 0 otherwise.

Assuming Gaussian distribution on U and Y, GLLiM estimates a locally linear mapping function $g : U \mapsto g(U)$ such that $Y = g(U)$ using the model of a mixture of K Gaussian (so the name Gaussian Locally Linear Mapping). The complete method uses Probabilistic Piecewise Affine Mapping (PPAM) to estimate the parameters of the mapping thanks to an Expectation-Minimization algorithm. This mapping yields an efficient estimator of U given Y. Practically, in the rest of the document GLLiM will denote through misuse of language the whole mapping estimation procedure encompassing also the PPAM step. Theoretical details on this method can be found in [Deleforge 2013, Deleforge *et al.* 2015b] while technical parameterization for our use on sound source localization will be detailed in section 8.6.

8.5 The VAST dataset

In order to apply this new paradigm of virtually supervised learning to source localization or more generally to sound scene analysis, one needs to use proper simulated data generated from sound propagation physics models. The following section introduces a novel dataset of Room Impulse Responses (RIRs) specifically designed to be used in a virtually supervised setting for acoustic inverse problems.

8.5.1 General Principles

The space of all possible acoustic scenes is vast. Therefore, some trade-offs between the size and the representativity of the dataset must be made when building a training corpus for audio scene geometry estimation. During the process of designing the dataset, we imposed on ourselves the following guidelines:

- The dataset should consist of room impulse responses (RIR). This is a more generic representation than, *e.g.*, specific audio features or audio scenes involving specific sounds. Each RIR should be annotated by all the source, receiver and room properties defining it.
- Virtual acoustic space traveling aims at building a dataset for a **specific audio system** in a variety of environments. Following this idea, some intrinsic properties of the receiver such as its distance to the ground and its (head)-related transfer functions are kept fixed throughout the simulations. For this first dataset, called *VAST_KEMAR_0*, we chose the emblematic KEMAR acoustic dummy-head, whose

measured HRTFs are publicly available. It was placed at 1.70 from the ground, the average human's height.

- We are interested in modeling acoustic environments which are typically encountered in an office building, a university, a hotel or a modern habitation. Acoustics of the type encountered in a cathedral, a massive hangar, a recording studio or outdoor are deliberately left aside here. Surface materials and diffusion profiles are chosen accordingly.

- To make the dataset easily manipulable on a simple laptop, we aimed at keeping its total size under 10 GigaBytes. To handle datasets of larger order of magnitudes would require users to have access to specific hardware and software which is not desired here. *VAST_KEMAR_0* weights 6.4 GB.

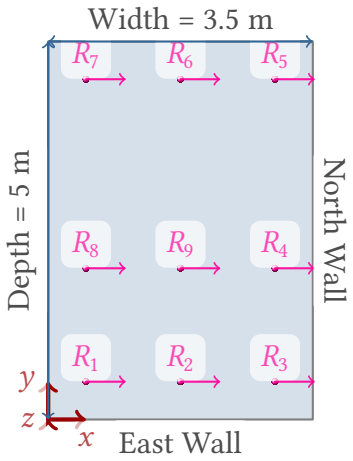
8.5.2 Room Simulation and Data Generation

The efficient C++/MATLAB "shoebox" 3D acoustic room simulator ROOMSIM developed by Schimmel et al. is selected for simulations [Schimmel *et al.* 2009]. This software takes as input a room dimension (width, depth and height), a source and receiver position and orientation, a receiver's (HRTF) model, and frequency-dependent absorption and diffusion coefficients for each surface. It outputs a corresponding pair of RIR at each ear of the binaural receiver. Specular reflections are modeled using the image-source method [Allen & Berkley 1979], while diffusion is modeled using the so-called *rain-diffusion* algorithm. In the latter, sound rays uniformly sampled on the sphere are sent from the emitter and bounced on the walls according to specular laws, taking into account surface absorption. At each impact, each ray is also randomly bounced towards the receiver with a specified probability (the frequency-dependent *diffusion coefficient* of the surface). The total received energy at each frequency is then aggregated using histograms. This model was notably shown to realistically account for sound scattering due to the presence of objects, by comparing simulated RIRs with measured ones in [Wabnitz *et al.* 2010]. The study [Kataria *et al.* 2017] suggests that such diffusion effects play an important role in sound source localization performance. *VAST_KEMAR_0* contains over $110,000$ RIR, which required about 700 CPU-hours of computation. This was done using a massively parallelized implementation on a large computing grid (IGRIDA[2]) available for research teams at IRISA (Rennes, France).

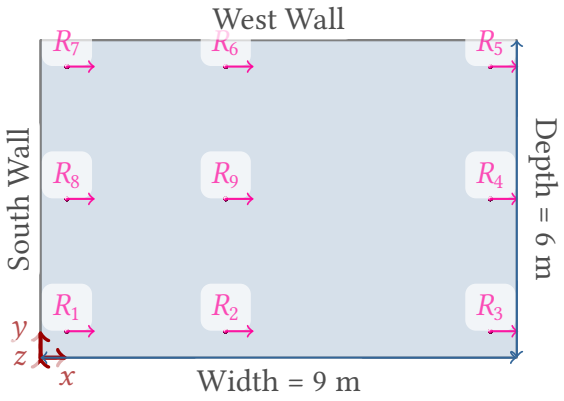
8.5.3 Room Properties: Size and Surfaces

An obvious choice to generate virtual rooms with maximal variability would be to draw a random room size and random frequency-dependent absorption and diffusion profiles of surfaces for each generated RIR. This approach however, has several drawbacks. First, it makes impossible the generation of realistic audio scenes containing several sources, for which the receiver position and the room must be fixed. Second, the space of possible rooms is so vast that reliably sampling it at random is unrealistic. Third, changing source,

[2] http://igrida.gforge.inria.fr



(a) Small room



(b) Large room

Figure 8.3 – Top views of training rooms with receiver positions and orientations

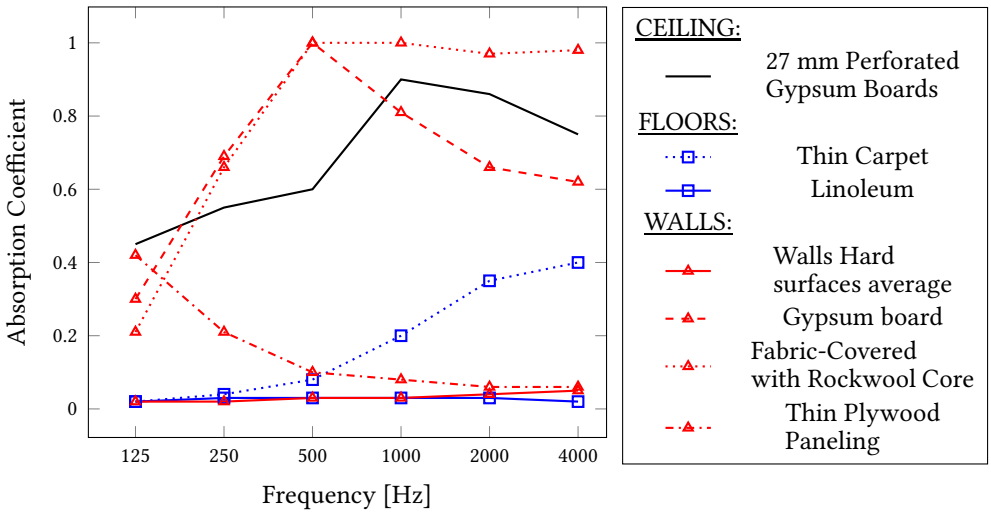


Figure 8.4 – Absorption Profiles

Table 8.1 – Description of simulated training rooms in VAST

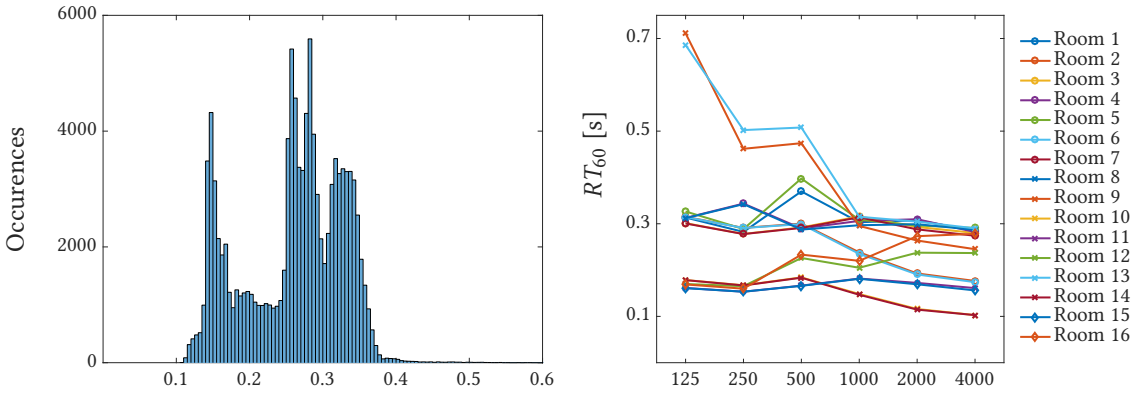
Room Number	Floor	Ceiling	Walls	Width [m]	Depth [m]	Height [m]
1	Thin Carpet	Perforated 27 mm gypsum board	Walls Hard Surfaces Average	9	6	3.5
2	Thin Carpet	Perforated 27 mm gypsum board	Gypsum Board with Mineral Filling	9	6	3.5
3	Thin Carpet	Perforated 27 mm gypsum board	Fabric-Covered Panel with Rockwool Core	9	6	3.5
4	Thin Carpet	Perforated 27 mm gypsum board	Thin Plywood Paneling	9	6	3.5
5	Linoleum	Perforated 27 mm gypsum board	Walls Hard Surfaces Average	9	6	3.5
6	Linoleum	Perforated 27 mm gypsum board	Gypsum Board with Mineral Filling	9	6	3.5
7	Linoleum	Perforated 27 mm gypsum board	Fabric-Covered Panel with Rockwool Core	9	6	3.5
8	Linoleum	Perforated 27 mm gypsum board	Thin Plywood Paneling	9	6	3.5
9	Thin Carpet	Perforated 27 mm gypsum board	Walls Hard Surfaces Average	3.5	5	2.5
10	Thin Carpet	Perforated 27 mm gypsum board	Gypsum Board with Mineral Filling	3.5	5	2.5
11	Thin Carpet	Perforated 27 mm gypsum board	Fabric-Covered Panel with Rockwool Core	3.5	5	2.5
12	Thin Carpet	Perforated 27 mm gypsum board	Thin Plywood Paneling	3.5	5	2.5
13	Linoleum	Perforated 27 mm gypsum board	Walls Hard Surfaces Average	3.5	5	2.5
14	Linoleum	Perforated 27 mm gypsum board	Gypsum Board with Mineral Filling	3.5	5	2.5
15	Linoleum	Perforated 27 mm gypsum board	Fabric-Covered Panel with Rockwool Core	3.5	5	2.5
16	Linoleum	Perforated 27 mm gypsum board	Thin Plywood Paneling	3.5	5	2.5
0	Anechoic room					

receiver and room parameters all at the same time prevents from getting insights on the individual influence of these parameters. On the other hand, sampling all combinations of parameters in an exhaustive way quickly leads to enormous data size. As a trade-off, we designed 16 realistic rooms representative of typical reverberation time (RT_{60}) and surface absorption profiles encountered in modern buildings. Two room sizes were considered: a small one corresponding to a typical office or bedroom (Figure 8.3a), and a larger one corresponding to a lecture or entrance hall (Figure 8.3b). For each room, floor, ceiling and wall materials which are representative in terms of absorption profile and are commonly encountered in nowadays buildings were chosen from [Vorländer 2007]. The graph on Figure 8.4 displays the absorption profiles of the selected materials, namely, 4 for the walls, 2 for the floor and 1 for the ceiling. The gypsum board material chosen for the ceiling was kept fixed throughout the dataset, as it represents well typical ceiling absorption profiles [Vorländer 2007]. "Walls hard surface average" is in fact an average profile over many surfaces such as brick or plaster [Vorländer 2007]. Combining all possible floors, walls and room sizes yielded the 16 rooms listed in Table 8.1.

Importantly, typical rooms also contain furniture and other objects responsible for random sound scattering effects, *i.e.*, diffusion. Following the acoustic study in [Faiz *et al.* 2012], a unique frequency-dependent diffusion profile was used for all surfaces. The chosen profile is the average of the 8 configurations measured in [Faiz *et al.* 2012], corresponding to varying numbers of chairs, table, computers and people in a room. Both absorption and diffusion profiles are piecewise-linearly interpolated from 8 Octave bands from 125 Hz to 4 kHz.

8.5.4 Reverberation Time

A common acoustic descriptor for rooms is the reverberation time (RT_{60} described in chapter 7). Figure 8.5 displays the estimated RT_{60} distribution across the VAST Training Dataset. Figure 8.5 shows the RT_{60} for each room by octave band. RT_{60}'s were estimated



(a) Global RT_{60} distribution (b) Octave bands RT_{60}

Figure 8.5 – Reverberation Time distributions

from the room impulse responses following the recommendations in [Schroeder 1965]. From these estimations, we decided to crop the room impulse responses provided in the datasets above the RT_{60}, with a 30 ms margin. This technique allows to shrink the dataset while keeping data points of interest and discarding the rest. To further complies with memory limitations, we chose to encode the room impulse response samples with single floats (16 bit). As can be seen in Figure 8.5 the 16 chosen rooms present a quite good variability in terms of reverberation times in the range 100ms-400ms. Larger RT_{60} of the order of 1 second could be obtain by using highly reflective materials on all surfaces, creating an echoic chamber. However, this rarely occurs in realistic buildings.

8.5.5 Source and Receiver Positions

A relatively poorly-studied though important effect in sound source localization is the influence of the receiver's position in the room, especially its distance to the nearest surface. In order to accurately capture this effect, 9 receiver positions are used for each of the 16 rooms, while the height of the receiver is fixed at 1.7 m. Figure 8.3 shows top views of the rooms with receiver positions. Positions from R_1 to R_8 are set 50 cm from the nearest wall(s) whereas R_9 is approximately placed in the middle of the room. Perfectly symmetrical configurations are avoided to make the dataset as generic as possible, without singularities. The receiver is always facing the north wall as a convention. For each of the 9 receiver positions, sources are placed on spherical grids centered on the receiver. Each sphere consists of regularly-spaced elevation lines each containing sources at regularly-spaced azimuths, with a spacing of 9°. The equator elevation line and the first azimuth angle of each line are randomly offset by −4.5° to +4.5° in order to obtain a dense sphere sampling throughout the dataset. Six spherical grid radii are considered, yielding source distances of 1, 1.5, 2, 3, 4 and 6 meters. Sources falling outside of the room or less than 20 cm from a surface are removed.

Table 8.2 – Simulated test sets description

	VAST Testing Set 1	VAST Testing Set 2	VAST Testing Set 3	VAST Testing Set 4
Receiver Position	Random 2D (fixed height)	Random 2D (fixed height)	Random 2D (fixed height)	Random 2D (fixed height)
Receiver Orientation	Same as Training	Random Yaw Angle	Same as Training	Random Yaw Angle
Source Position	Random 3D	Random 3D	Random 3D	Random 3D
Room Width [m]	Same as Training	Same as Training	Random in [3,10]	Random in [3,10]
Room Depth [m]	Same as Training	Same as Training	Random in [3,10]	Random in [3,10]
Room Height [m]	Same as Training	Same as Training	Random in [2,4]	Random in [2,4]
Ceiling Material	Same as Training	Same as Training	Same as Training	Same as Training
Floor Covering	Same as Training	Same as Training	Random From The Training Set	Random From The Training Set
Walls Material	Same as Training	Same as Training	Random From The Training Set (for each wall)	Random From The Training Set (for each wall)
Number of Points	1000 (x 16 Rooms)	1000 (x 16 Rooms)	10 000	10 000

8.5.6 Test Sets

To test the generalizability of mappings learned on the *VAST_KEMAR_0* dataset, we built four simulated test sets differing from the training dataset on various levels. A first challenge is to test robustness to random positioning, since the training set is built with regular spherical source grids and fixed listener positions. Hence, the 4 testing sets contain completely random source and receiver positions in the room. Only the receiver's height is fixed to 1.7 m, and both receiver and source are set within a 20 cm safety margin within the room boundaries. Test sets 2 and 4 feature random receiver orientation (yaw angle), as opposed to the receiver facing north in the training set. Test 1 and 2 contain 1,000 binaural RIRs (BRIRs) for each of the 16 rooms of Table 8.1. Finally, test sets 3 and 4 contain 10,000 BRIRs, each corresponding to a random room size (walls from $3m \times 2m$ to $10m \times 4m$) and random absorption properties of walls and floor picked from Figure 8.4. Different surfaces for all 4 walls are allowed.

Note that the KEMAR HRTF measurements used to simulate the VAST dataset was recorded by yet another team, in MIT's anechoic chamber in 1994, as described in [Gardner & Martin 1995].
While this section presented the details of the dataset, a visual description of the organization is available on Figure C.1, Appendix C.

8.6 Localizing sound sources through learning on simulated data

In order to validate the VAST approach on auditory scene analysis, one obvious choice regarding a low-dimension acoustic scene properties prediction was sound source position. In addition to these simulated test sets, three binaural RIR datasets recorded with the KEMAR dummy head in real rooms have been selected, as listed below:

- **Auditorium 3 [Ma *et al.* 2015]** was recorded at TU Berlin in 2014 in a trapezium-shaped lecture room of dimensions 9.3m × 9m and $RT_{60} \approx 0.7$s. 3 individual sources placed 1.5m from the receiver at different azimuth and 0° elevation were recorded.

For each source, one pair of binaural RIR is recorded for each receivers' head yaw angle from -90° to $+90^\circ$, with 1° steps;

- **Spirit [Ma *et al.* 2015]** was recorded at TU Berlin in 2014 in a small rectangular office room of size 4.3m × 5m, $RT_{60} \approx 0.5$s, containing various objects, surfaces and furniture near the receiver. The protocol is the same as Auditorium 3 except sources are placed 2 m from the receiver;

- **Classroom [Shinn-Cunningham *et al.* 2005]** was recorded at Boston University in 2005 in a 5m × 9m × 3.5m carpeted classroom with 3 concrete walls and one sound-absorptive wall ($RT_{60} = 565$ms). The receiver is placed in 4 locations of the room including 3 with at least one nearby wall.

For all experiments in this section, all training and test sets used are reduced to contain only frontal sources (azimuth in $[-90^\circ, +90^\circ]$) with elevation in $[-45^\circ, +45^\circ]$ and distances between 1 and 3 meters.

8.6.1 Binaural features

As mentioned earlier, sound source localization thanks to virtually supervised learning consists in two steps: calculating high dimensional features from (binaural) signals followed by mapping these features to a source position. As one might not want to use the full RIR as input, in the current binaural scenario, a choice for high dimensional data are auditory features. Even if their dimension is already smaller compared to the initial RIRs, auditory features such as Interaural Level Difference (ILD) or Interaural Phase Difference (IPD) still convey important information about sound scenes. Shortly, ILD and IPD account for level and time (phase) differences at the two ears due to the head shadow and the spatial distribution of the source(s). Thus, both ILD and IPD were shown to be important cues for sound localization and speech intelligibility in noise. Robustly estimating features can be difficult when dealing with additive noise, sources with sparse spectra such as speech or music, and source mixtures. We leave this problematic aside in this paper, and focus on mapping clean features to source positions. Hence, we use *ideal* features directly calculated from the clean room impulse responses in all experiments. We detail below how they are computed from the raw RIR.

Let $\mathbf{u}_n \in \mathbb{R}^3$ be a parameter vector containing the source's azimuth, elevation and distance absorption. We denote the associated generated left and right RIR by $(\mathbf{h}^L(\mathbf{u}_n), \mathbf{h}^R(\mathbf{u}_n))$. Each of these pairs is convolved with a 1 second random white Gaussian noise signal, and the result is resampled at 8kHz. The STFT is then applied to both signals, using a 64ms sliding time window with 50% overlap. This results in a left-microphone spectrogram $\{\mathbf{L}(f,t)\}_{f=1,t=1}^{F,T}$ and a right-microphone spectrogram $\{\mathbf{L}(f,t)\}_{f=1,t=1}^{F,T}$, where $F = 256$ and $T = 32$. If $\{\mathbf{S}(f,t)\}_{f=1,t=1}^{F,T}$ denotes the emitted white-noise spectrogram, under the assumption that most of the RIR energy is concentrated on the first 64ms, we have the

following approximate multiplicative model:

$$\begin{cases} \mathbf{L}(f,t) \approx \bar{\mathbf{h}}^{L}(f,\mathbf{u}_n)\mathbf{S}(f,t) \\ \mathbf{R}(f,t) \approx \bar{\mathbf{h}}^{R}(f,\mathbf{u}_n)\mathbf{S}(f,t) \end{cases} \tag{8.12}$$

where $\bar{\cdot}$ denotes the discrete Fourier transform. The *interaural level difference* (ILD) and *interaural phase difference* (IPD) spectrograms are defined by

$$\begin{cases} \mathrm{ILD}(f,t) = 20 * \log(|\mathbf{L}(f,t)|/|\mathbf{R}(f,t)|) \in \mathbb{R} \\ \mathrm{IPD}(f,t) = \frac{\mathbf{L}(f,t)/|\mathbf{L}(f,t)|}{\mathbf{R}(f,t)/|\mathbf{R}(f,t)|} \in \mathbb{C}. \end{cases} \tag{8.13}$$

Using the approximation Equation (8.12), it is easily seen that both ILD and IPD solely depend on the parameter vector $\mathbf{u}_n$ and do not depend on the emitted signal. Similarly to [Deleforge *et al.* 2015c], the ILD and IPD spectrograms are vertically concatenated and averaged over time to form a high-dimensional feature vector $\mathbf{y}_n \in \mathbb{R}^D$ associated to the low-dimensional parameter vector $\mathbf{u}_n \in \mathbb{R}^L$ ($L = 3$ for 3D coordinates for instance).

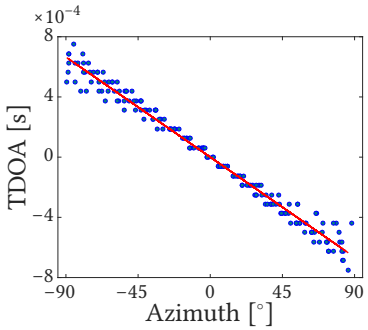
8.6.2 Experiments

We first make an experiment to put forward some intrinsic limitations of TDOA-based azimuth estimation. Figure 8.6 plots TDOAs against the source's azimuth angle for different subsets of VAST. TDOAs were computed as the delay maximizing the correlation between the first 500 samples of the left and the right impulse responses. As can be seen in Figure 8.6, a near-linear relationship between frontal azimuth and TDOA exists in the anechoic case, regardless of the elevation. This matches previously observed results in binaural sound localization [Viste & Evangelista 2003, Sanchez-Riera *et al.* 2012, Deleforge *et al.* 2015c]. When the receiver is placed in the middle of the 16 reverberant rooms, (Figure 8.6b), some outliers appear due to reflections. This effect is dramatically increased when the receiver is placed 50 centimeters from a wall (Figure 8.6c and Figure 8.6d), where stronger early reflections are present. This suggests that the TDOA, even when ideally estimated, is not adapted to binaural sound source localization in realistic indoor environments.

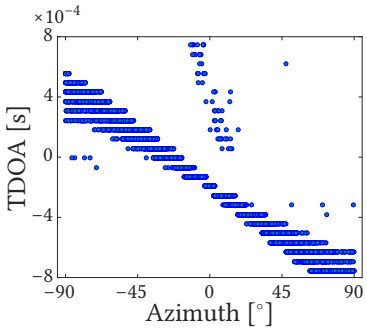
We then compare azimuth estimation errors obtained with the TDOA-based method described above, a learning-based method trained on anechoic HRTF measurements (Room 0), and a learning-based method trained on VAST, using the 4 simulated and 3 real test sets described in subsection 8.5.6. TDOAs were mapped to azimuth values using the affine regression coefficients corresponding to the red line in Figure 8.6a. If we denote θ the azimuth and τ the TDOA, this red fitting curve models the following affine relation:

$$\tau = a \cdot \theta + b. \tag{8.14}$$

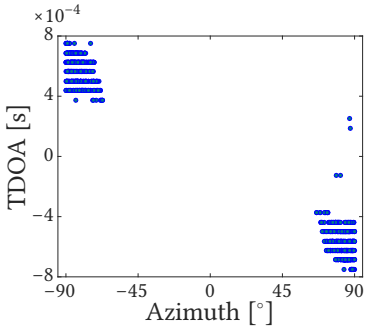
with $a = 1.37 \cdot 10^{-7} s.deg^{-1}$ and $b = -7.38 \cdot 10^{-6} s$. The chosen learning-based sound source localization method is the one described earlier in section 8.4. It uses GLLiM, to map high-dimensional feature vectors containing frequency-dependent interaural level and phase differences from 0 to 8000 Hz to low-dimensional source positions. In our



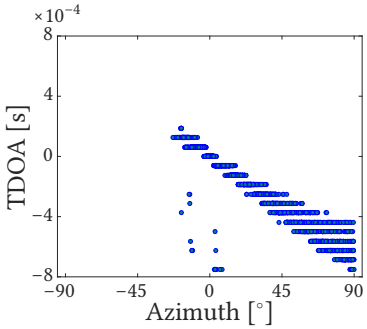
(a) Anechoic case



(b) VAST dataset (Receiver in central position)



(c) VAST dataset (Receiver in north position)



(d) VAST dataset (Receiver in south-west position)

Figure 8.6 – TDOA as a function of source azimuth in various settings

case, the GLLiM model with K locally-linear components was trained on N interaural feature vectors of dimension $D = 1537$ associated to 3-dimensional source positions in spherical coordinate (azimuth, elevation and distance). $K_1 = 8$ components were used for the anechoic training set ($N_1 = 181$) and $K_2 = 100$ for the (reduced) VAST dataset ($N_2 \approx 41,000$). All 3 methods showed comparably low testing computational times, in the order of 10ms for 1 second of input signal.

Table 8.3 – Azimuth absolute estimation errors in degrees with 3 different methods, showed in the form $avg \pm std(out\%)$, where avg and std denote the mean and standard deviation of *inlying* absolute errors ($< 30^\circ$) while out denotes the percentage of outliers.

Test data ↓	TDOA	GLLiM (Anech. train.)	GLLiM (VAST train.)
VAST Testing Set 1	5.49 ± 4.6(5.6%)	8.63 ± 7.6(12%)	4.38 ± 4.9(1.8%)
VAST Testing Set 2	5.37 ± 4.4(6.0%)	8.09 ± 7.5(12%)	4.32 ± 4.7(1.6%)
VAST Testing Set 3	5.21 ± 4.5(4.6%)	8.46 ± 7.5(5.2%)	4.23 ± 4.4(1.8%)
VAST Testing Set 4	5.14 ± 4.4(3.3%)	8.21 ± 7.2(4.8%)	4.25 ± 4.4(0.6%)
Auditorium 3 [Ma *et al.* 2015]	7.02 ± 4.7(1.4%)	8.01 ± 7.0(5.9%)	5.03 ± 4.5(0.0%)
Spirit [Ma *et al.* 2015]	5.19 ± 3.4(0.0%)	12.2 ± 8.3(15%)	4.50 ± 5.6(0.4%)
Classroom [Shinn-Cunningham *et al.* 2005]	5.71 ± 3.7(3.7%)	9.47 ± 7.3(5.2%)	6.50 ± 5.9(0.0%)

Table 8.3 summarizes obtained azimuth estimation errors. As can be seen, the learning method trained on VAST outperforms the two others on all datasets, with significantly less outliers and a globally reduced average error of inliers. This is encouraging considering the variety of testing data used.

Table 8.4 – Elevation and distance absolute estimation errors obtained with GLLiM trained on VAST. Outliers correspond to errors larger than 15° or 1m.

Test data ↓	Elevation (°)	Distance (m)
VAST Testing Set 1	5.91 ± 4.1(23%)	0.43 ± 0.3(19%)
VAST Testing Set 2	6.05 ± 4.2(27%)	0.44 ± 0.3(20%)
VAST Testing Set 3	6.05 ± 4.1(27%)	0.43 ± 0.3(21%)
VAST Testing Set 4	6.03 ± 4.2(26%)	0.44 ± 0.3(21%)
Auditorium 3 [Ma *et al.* 2015]	7.92 ± 4.4(44%)	0.45 ± 0.3(23%)
Spirit [Ma *et al.* 2015]	7.44 ± 4.3(30%)	0.52 ± 0.3(25%)
Classroom [Shinn-Cunningham *et al.* 2005]	8.40 ± 4.1(45%)	0.41 ± 0.3(6.5%)

In addition, Table 8.4 shows that GLLiM trained on VAST is capable of approximately estimating the elevation and distance of the source, which is known to be particularly difficult from binaural data. While elevation estimation on real data remains a challenge, results obtained on simulated sets are promising.

8.7 Summary

We introduced the new concept of virtual acoustic space traveling and released a first dataset dedicated to it. A methodology to efficiently design such a dataset was provided, making extensions and improvements of the current version easily implementable in the future. Results show that a learning-based sound source localization method trained on this dataset yields better localization results than when trained on anechoic HRTF measurements, and performs better than a TDOA-based approach in azimuth estimation while being able to estimate source elevation and distance. Considering the current knowledge, this is the first time a method trained on simulated data is successfully used on real data for (binaural) sound source localization, validating the new concept of virtual acoustic space traveling. The learning approach could still be significantly improved by considering other features, by better adapting the mapping technique to spherical coordinates and by annotating training data with further acoustic information. Other learning methods such as deep neural networks may also be investigated. In that line, we note a recent attempt to learn such a mapping with Convolutional Recurrent Neural Networks using RIRs recorded within AMBISONICS sound representations [Perotin *et al.* 2018]. Considering additional tasks to achieve, room characterization involving wall position or absorption prediction could possibly be interesting to study.

Chapter 9

Conclusions and perspectives

Contents

The main line of the work presented in this manuscript was to gather tools and models in a general framework able to handle different audio and acoustic inverse problems. Building this framework, we considered three applications relying on (structured) (co)sparsity: audio denoising, audio desaturation and audio dereverberation. More independently, we proposed a new line of work to cope with the sound source localization problem based on virtual acoustic space learning. This last chapter summarizes the central contributions and expresses some possible future research directions.

9.1 Conclusions

After chapter 1 which was the main introduction of this thesis, Part I presented several tools used in this work. In chapter 2, we detailed various existing models based on sparsity (both analysis and synthesis based) for audio signal time-frequency modeling. This chapter also presented an extension of the sparse models in the multichannel audio context adding structure across channels.

In chapter 3 we introduced a generic algorithmic framework able to embed different (co)sparse data models thanks to appropriate sparsity inducing shrinkages. This versatile framework was designed to address several audio reconstruction problems by approaching a solution of the associated non-convex optimization problem. Since the

main purpose of this framework is audio reconstruction, chapter 4 discussed the different measures and data available for experimental validation on audio signals.

Part II focused on addressing inverse problems induced from different types of sensor-based distortions. In chapter 5 we instantiated the framework to account for a first audio reconstruction application: denoising. We alternatively studied the impact of the different time-frequency models (plain sparsity *v.s.* social sparsity), priors (analysis *v.s.* synthesis) and frequency transform redundancy on the denoising performance. Results were compared on small scale and large scale datasets of music and speech sound examples. Numerical results show advantage of the plain cosparse method on global audio quality for music while on speech, the adaptive social (co)sparse methods seem worthwhile. For speech intelligibility improvements and also computational performance, the plain sparse method appears to be preferable. We also insisted on competitive (even better for high SNR) denoising performance compared with a baseline method.

As an extension of the framework, chapter 6 discussed audio desaturation. We first gave some insights on saturation range and ways to quantify clipping. We selected SDR as more adapted than clipping threshold to rate distortion induced by clipping. We thoroughly studied the different flavors of the chapter 3 framework on a large scale speech and music dataset for the single-channel case. While the plain cosparse method gives better SDR improvements for highly degraded scenarii, the adaptive social sparse algorithm seems preferable for moderate to light clipping conditions. When the analysis operator or the Hermitian transpose of the dictionary forms a Parseval tight-frame, we stress the low computational cost of the algorithms and validate this showing runtime performance. With the active work and efforts of the research engineers of the PANAMA team and industrial transfer division of Inria/IRISA Rennes research center, this work on single-channel declipping is currently being adapted to work as a declipping plug-in for professional audio restoration softwares[1]. We also encompassed multichannel declipping cases in the framework. Therefore, we studied the legitimacy of the channel-aware structured (co)sparsity signal models on multichannel declipping scenarii. This model was validated on stereo and 8-channel recordings. The resulting algorithms, using the Group Empirical Wiener and the Quadratic Group Empirical Wiener as sparsifying operators, showed better reconstruction performance (SDR improvement) than a state-of-the-art method relying on simple cosparse prior. We also demonstrate better computational efficiency for some configurations.

Part III was devoted to inverse problems caused by indoor environment sound propagation. In chapter 7, we addressed the audio dereverberation issue including it in the common framework of chapter 3. As a proof of concept, we presented dereverberation results using wide-band plain/social sparse time-frequency modeling. SDR improvements demonstrated superiority of the social sparse modeling. This effect was more salient in highly reverberant configurations.

[1] A demonstration of the application is available as a web-service at: `https://spade.inria.fr` thanks to the A||GO platform.

More independently from the rest of this manuscript, chapter 8 was committed to binaural sound source localization. We introduced the new concept of virtual acoustic space learning taking the best of both physics-driven and data driven worlds to learn models from acoustic features in simulated environments. This concept was successfully applied to binaural sound source localization in realistic rooms and validated on both simulated and real recordings. The method shows really competitive results in azimuth estimation compared to the widely used more traditional GCC-PHAT. The method using virtual acoustic space learning was shown to be able to provide also elevation and distance estimation. This is known to be a very difficult task especially with only two sensors.

9.2 Further work

For this last section, we foresee some future research possibilities for the different issues we considered in this work.

9.2.1 Inpainting tasks

Due to the versatility of the framework we developed in this thesis, we can think of interesting extensions to other signal reconstruction problems.

Packet loss concealment We showed that declipping as a specific audio inpainting task could be addressed by the framework. A comparable problem that could be handled is completion of missing samples. With the growing number of applications involving transmission of digital audio streams, this is probably a task that is of interest to alleviate the packet loss problem. In that sense, some recent studies involving sparse [Mokrỳ *et al.* 2018] and structured sparse [Lieb & Stark 2018] models provided encouraging results for this inpainting task.

Including it in the framework would only need a new generalized projection $\mathcal{P}_{\Theta,\mathbf{M}}(\mathbf{Z})$ adapted to the data-fidelity constraint for the inpainting case. Denote Ω_r the indices of the reliable samples (not missing) in matrix $\mathbf{Y}$.
The data-fidelity constraint for the inpainting task would be expressed for the analysis setting with $\mathbf{M} := \mathbf{A}$ by

$$\Theta := \left\{ \mathbf{W} \mid \mathbf{W}_{\Omega_r} = \mathbf{Y}_{\Omega_r} \right\}$$

where $\mathbf{W}$ is a time-domain estimate of the same size as $\mathbf{Y}$. For the synthesis setting, with $\mathbf{M} := \mathbf{I}$, we set

$$\Theta := \left\{ \mathbf{W} \mid (\mathbf{DW})_{\Omega_r} = \mathbf{Y}_{\Omega_r} \right\}.$$

Here $\mathbf{W}$ would be a time-/channel- frequency estimate gathering as many frames as in $\mathbf{Y}$. Similarly to the declipping projection, the corresponding generalized projection could be

done component wise retaining low complexity. Multichannel instance of the inpainting problem could also be treated at little cost relying on the channel-wise structured (co)sparse models.

Towards simultaneous reconstruction tasks More difficult issues could be investigated. As audio signals can easily be corrupted simultaneously by noise, reverberation and clipping there is a need to address reconstruction problems considering concurrent degradation. While working on our framework, unsuccessful attempts were tried to perform joint declipping and denoising or joint dereverberation and declipping on audio recordings. Building a hybrid generalized projection accounting for both denoising and declipping data-fidelity constraints is probably not that straightforward. Fusing the declipping and denoising projections in a naive way did not give expected results. Very recent work on audio de-quantization [Rencker *et al.* 2018] with sparse priors can be an interesting starting point.

Joint time-frequency / channel (co)sparse models Structured (co)sparsity showed independently good performances in modeling diversity of audio signals in the time-frequency domain or channel-frequency domain. Acoustic sensing is currently being redesigned with the generalization of stereo to more and more channel recordings. In that sense, we can think of designing more complete multichannel signal models promoting at the same time structured (co)sparsity across channels and social sparsity in time-frequency representations. For that purpose, tensor tools which were successfully used for audio signal modeling in the context of source separation [Ozerov *et al.* 2011] may be considered.

9.2.2 Algorithmic aspects

Despite the good experimental results of the framework for various audio reconstruction tasks, no theoretical guarantees are provided. Convergence proof for the framework might be interesting to derive as empirical results show that the different algorithms stop with a convergence threshold parameter (β) while the upper bound on the iteration count (i_{max}) is never reached. However, this stopping criterion (β) seems to play an important role in the final reconstruction performance as shown for declipping (see: Figure 6.14 and Figure 6.15, chapter 6). This observation alongside with the non-convexity of a possible underlying problem stresses the difficulty of deriving a convergence proof for the framework. As a first step, a (possibly easier) convergence study of the framework equipped with non-overlapping (structured) shrinkages could be envisioned.

9.2.3 Virtual acoustic space learning

Concerning acoustic space learning, several axes can be chosen for further investigations.

Generalizing source localization It will not have escaped the attentive reader that the concept of virtual acoustic space learning was validated on binaural recordings which is quite restrictive compared to the wide variety of available microphone arrays. One idea could be to adapt the method to account for a more generic sensing array. A first idea could be testing a blinder approach for the learning method without providing it with a fully detailed parametrization of the sensor array. We can think of adapting the learning step to be able to learn a model from multichannel recordings gathered with samples from just a pair of sensors. However, with this idea it could probably be difficult to generalize to other than linear arrays. Concerning the learning method itself, some comparison could be held with one or more learning methods from the deep neural network framework as it seems to be also a good alternative for source localization [Perotin *et al.* 2018].

In the line of generalization, one could think of multi-source and/or dynamic scenarii where the task would extend to locating multiple static sources or in a more difficult case providing tracking for moving sources. Some work [Laufer-Goldshtein *et al.* 2017] using jointly manifold learning along with Kalman filtering provided interesting results for such an application.

Room parameters estimation On top of source position, lower level audio scene parameters estimation can be interesting to look at. For instance, room geometry, absorption/diffusion properties. For that purpose and considering the generalization to larger arrays, different acoustic features could be envisioned. In a binaural setting, inter microphone time delays and level differences are widely used. The choice of acoustic features to learn a model from is a crucial question. Hence, turning towards Direct-to-Reverberant Ratio to improve source distance estimation or room properties could be a first option. Another interesting longer term perspective in that sense could be the design of new acoustic features.

With this work we probably did not win the *sparse wars* but certainly took a step forward in legitimating some signal models. These findings could be useful in case of joining forces with the machine learning community in a possible near future. Hopefully, this could lead to interesting applications for those needing a little help to enhance the way they process sound.

Appendices

Appendix A

Generalized projections

A.1 Generalized projection for denoising

The goal is to solve Equation (3.10), i.e.,

$$\underset{\mathbf{W}\in\Theta}{\text{minimize}}\ \|\mathbf{MW}-\mathbf{Z}\|_{\mathrm{F}}.$$

with $\mathbf{M}=\mathbf{A}$, $\Theta=\{\mathbf{W}:\|\mathbf{W}-\mathbf{Y}\|_{\mathrm{F}}\leq\varepsilon\}$ for the analysis case, and $\mathbf{M}=\mathbf{I}$ and $\Theta=\{\mathbf{W}:\|\mathbf{DW}-\mathbf{Y}\|_{\mathrm{F}}\leq\varepsilon\}$ for the synthesis case. For the synthesis case, this is more explicitly

$$\underset{\mathbf{W}}{\text{minimize}}\ \|\mathbf{W}-\mathbf{Z}\|_{\mathrm{F}}^2 \text{ subject to } \|\mathbf{DW}-\mathbf{Y}\|_{\mathrm{F}}^2\leq\varepsilon^2.$$

Let us now show that in the analysis case the optimization problem can be cast to a similar form. Since we consider a Parseval tight frame $\mathbf{A}^{\mathsf{H}}\mathbf{A}=\mathbf{I}$, the orthogonal projection onto the linear span of $\mathbf{A}$ is $P_{\mathbf{A}}=\mathbf{A}\mathbf{A}^{\mathsf{H}}$ and for any $\mathbf{W},\mathbf{Z}$,

$$\begin{aligned}\|\mathbf{AW}-\mathbf{Z}\|_{\mathrm{F}}^2 &= \|\mathbf{AW}-P_{\mathbf{A}}\mathbf{Z}+(\mathbf{I}-P_{\mathbf{A}})\mathbf{Z}\|_{\mathrm{F}}^2\\ &= \|\mathbf{AW}-P_{\mathbf{A}}\mathbf{Z}\|_{\mathrm{F}}^2+\|(\mathbf{I}-P_{\mathbf{A}})\mathbf{Z}\|_{\mathrm{F}}^2\\ &= \|\mathbf{A}(\mathbf{W}-\mathbf{A}^{\mathsf{H}}\mathbf{Z})\|_{\mathrm{F}}^2+\|(\mathbf{I}-P_{\mathbf{A}})\mathbf{Z}\|_{\mathrm{F}}^2\\ &= \|\mathbf{W}-\mathbf{A}^{\mathsf{H}}\mathbf{Z}\|_{\mathrm{F}}^2+\|(\mathbf{I}-P_{\mathbf{A}})\mathbf{Z}\|_{\mathrm{F}}^2.\end{aligned}$$

Minimizing the left hand side with the constraint $\mathbf{W}\in\Theta$ is thus equivalent to

$$\underset{\mathbf{W}}{\text{minimize}}\ \|\mathbf{W}-\mathbf{A}^{\mathsf{H}}\mathbf{Z}\|_{\mathrm{F}}^2 \text{ subject to } \|\mathbf{W}-\mathbf{Y}\|_{\mathrm{F}}^2\leq\varepsilon^2.$$

Both cases boil down to an optimization problem

$$\hat{\mathbf{W}}=\underset{\mathbf{W}}{\operatorname{argmin}}\|\mathbf{W}-\mathbf{B}\|_{\mathrm{F}}^2 \text{ subject to } \|\mathbf{FW}-\mathbf{Y}\|_{\mathrm{F}}\leq\varepsilon \tag{A.1}$$

with $\mathbf{B}=\mathbf{A}^{\mathsf{H}}\mathbf{Z}$ and $\mathbf{F}=\mathbf{I}$ for the analysis case, while $\mathbf{B}=\mathbf{Z}$ and $\mathbf{F}=\mathbf{D}$ for the synthesis case. When $\mathbf{F}$ is a Parseval tight frame, Equation (A.1) has a closed form solution [Yang & Yuan 2013, Section 2]

$$\hat{\mathbf{W}}=\mathbf{B}-\left(\frac{\|\mathbf{FB}-\mathbf{Y}\|_{\mathrm{F}}-\varepsilon}{\|\mathbf{FB}-\mathbf{Y}\|_{\mathrm{F}}}\right)_{+}\cdot\mathbf{F}^{\mathsf{H}}(\mathbf{FB}-\mathbf{Y}). \tag{A.2}$$

A.2 Generalized projection for declipping

The goal is to solve Equation (3.10), i.e.,

$$\underset{\mathbf{W}\in\Theta}{\text{minimize}}\ \|\mathbf{MW}-\mathbf{Z}\|_{\mathrm{F}}.$$

with some constraint set Θ.

In the analysis case, as shown in section A.1, as soon as $\mathbf{A}^{\mathsf{H}}\mathbf{A}=\mathbf{I}$, minimizing $\|\mathbf{AW}-\mathbf{Z}\|_{\mathrm{F}}^2$ under the constraint

$$\mathbf{W}\in\Theta := \left\{ \mathbf{W} \mid \begin{array}{l} \mathbf{W}_{ij}=\mathbf{Y}_{ij},\ ij\in\Omega_r; \\ \mathbf{W}_{ij}\geq\tau, ij\in\Omega_+; \\ \mathbf{W}_{ij}\leq-\tau,\ ij\in\Omega_-. \end{array} \right\}$$

is equivalent to minimizing $\|\mathbf{W}-\mathbf{A}^{\mathsf{H}}\mathbf{Z}\|_{\mathrm{F}}^2$ under the constraint $\mathbf{W}\in\Theta$. As the constraint is written component-wise, the optimization can be done component-wise yielding

$$\hat{\mathbf{W}}_{(ij)} = \begin{cases} \mathbf{Y}_{ij} & \text{if } ij\in\Omega_r; \\ (\mathbf{A}^{\mathsf{H}}\mathbf{Z})_{ij} & \text{if } \begin{cases} ij\in\Omega_+, (\mathbf{A}^{\mathsf{H}}\mathbf{Z})_{ij}\geq\tau; \\ \text{or} \\ ij\in\Omega_-, (\mathbf{A}^{\mathsf{H}}\mathbf{Z})_{ij}\leq-\tau; \end{cases} \\ \operatorname{sgn}(\mathbf{Y}_{ij})\tau & \text{otherwise.} \end{cases} \tag{A.3}$$

For the synthesis case, $\mathbf{M}=\mathbf{I}$ and

$$\Theta := \left\{ \mathbf{W} \mid \begin{array}{l} (\mathbf{DW})_{\Omega_r}=\mathbf{Y}_{\Omega_r}; \\ (\mathbf{DW})_{\Omega_+}\succcurlyeq\mathbf{Y}_{\Omega_+}; \\ (\mathbf{DW})_{\Omega_-}\preccurlyeq\mathbf{Y}_{\Omega_-}. \end{array} \right\}.$$

The corresponding optimization problem for the synthesis case writes:

$$\hat{\mathbf{W}}=\underset{\mathbf{W}}{\operatorname{argmin}}\|\mathbf{W}-\mathbf{Z}\|_{\mathrm{F}}^2 \text{ subject to } \mathbf{W}\in\Theta. \tag{A.4}$$

which can be recast as:

$$\hat{\mathbf{W}}=\underset{\mathbf{W}}{\operatorname{argmin}}\|\mathbf{W}-\mathbf{Z}\|_{\mathrm{F}}^2 \text{ subject to } \mathbf{DW}\in\tilde{\Theta} \tag{A.5}$$

and

$$\tilde{\Theta} := \left\{ \mathbf{DW} \mid \begin{array}{l} (\mathbf{DW})_{\Omega_r}=\mathbf{Y}_{\Omega_r}; \\ (\mathbf{DW})_{\Omega_+}\succcurlyeq\mathbf{Y}_{\Omega_+}; \\ (\mathbf{DW})_{\Omega_-}\preccurlyeq\mathbf{Y}_{\Omega_-}. \end{array} \right\}.$$

As used in [Záviška *et al.* 2018, Šorel & Bartoš 2016] in the case where $\mathbf{DD}^{\mathsf{H}}=\mathbf{I}$ and $\tilde{\Theta}$ embodies a multidimensional interval constraint the closed-form solution for Equation (A.5) writes:

$$\hat{\mathbf{W}}=\mathbf{Z}-\mathbf{D}^{\mathsf{H}}(\mathbf{DZ}-\Pi_{\Theta,\mathbf{M}}(\mathbf{Z})), \tag{A.6}$$

with

$$[\Pi_{\Theta,\mathbf{M}}(\mathbf{Z})]_{ij} = \begin{cases} \mathbf{Y}_{ij} & \text{if } ij \in \Omega_r; \\ (\mathbf{DZ})_{ij} & \text{if } \begin{cases} ij \in \Omega_+, (\mathbf{DZ})_{ij} \geq \tau; \\ \text{or} \\ ij \in \Omega_-, (\mathbf{DZ})_{ij} \leq -\tau; \end{cases} \\ \operatorname{sgn}(\mathbf{Y}_{ij})\tau & \text{otherwise.} \end{cases}$$

Appendix B

Power iteration algorithm

We recall below the power iteration algorithm as used in the dereverberation projection Equation (7.6) page 99. It estimates the highest singular value associated to $\mathcal{T}^*\mathcal{T}$.

Algorithm 4 Power iteration algorithm

Require: $\mathbf{D}, \mathbf{Z}^{(0)}, \mathcal{T}(\cdot), \mathcal{T}^*(\cdot), i_{max}$
  for $i = 1$ to i_{max} **do**
    $\mathbf{W} = \mathcal{T}^*(\mathcal{T}(\mathbf{D}\mathbf{Z}^{(i-1)}))$
    $t = \|\mathbf{W}\|_\infty$
    $\mathbf{Z}^{(i)} = \frac{\mathbf{W}}{t}$
  return t

$\mathbf{D} \in \mathbb{C}^{L \times S}$ is the DFT operator
$\mathbf{Z}^{(0)} \in \mathbb{C}^{S \times T}$
$\|\mathbf{W}\|_\infty$ corresponds to the highest magnitude value of $\mathbf{W}$

Appendix C

VAST dataset structure

Figure C.1 coming next describes how is organized technically the VAST dataset presented earlier page 110 and successfully used for sound source localization (section 8.6). The figure below graphically presents a "VAST" dataset with N room impulse responses and details the fields available for annotating them as well as general informative fields about the data. Among the global parameters, "FreqBin" stores the central octave band frequencies used to provide frequency dependent sound absorption and diffusion profiles. Those central frequencies are also used to provide frequency dependent reverberation time ("FreqRT60"). For the receiver, the field "Position" is the 3D location of the sensor(s) in the cartesian coordinate system of the room. The source position is also given in the room referential in the field "AbsolutePos". The fields "Azimuth", "Elevation" and "Distance" are given in the receiver referential. Additionally, on the VAST training dataset (*VAST_KEMAR_0* and possible future release) there is a "Spot" field for the receiver which denotes the position inside the room from R_1 to R_9 (see Figure 8.3).

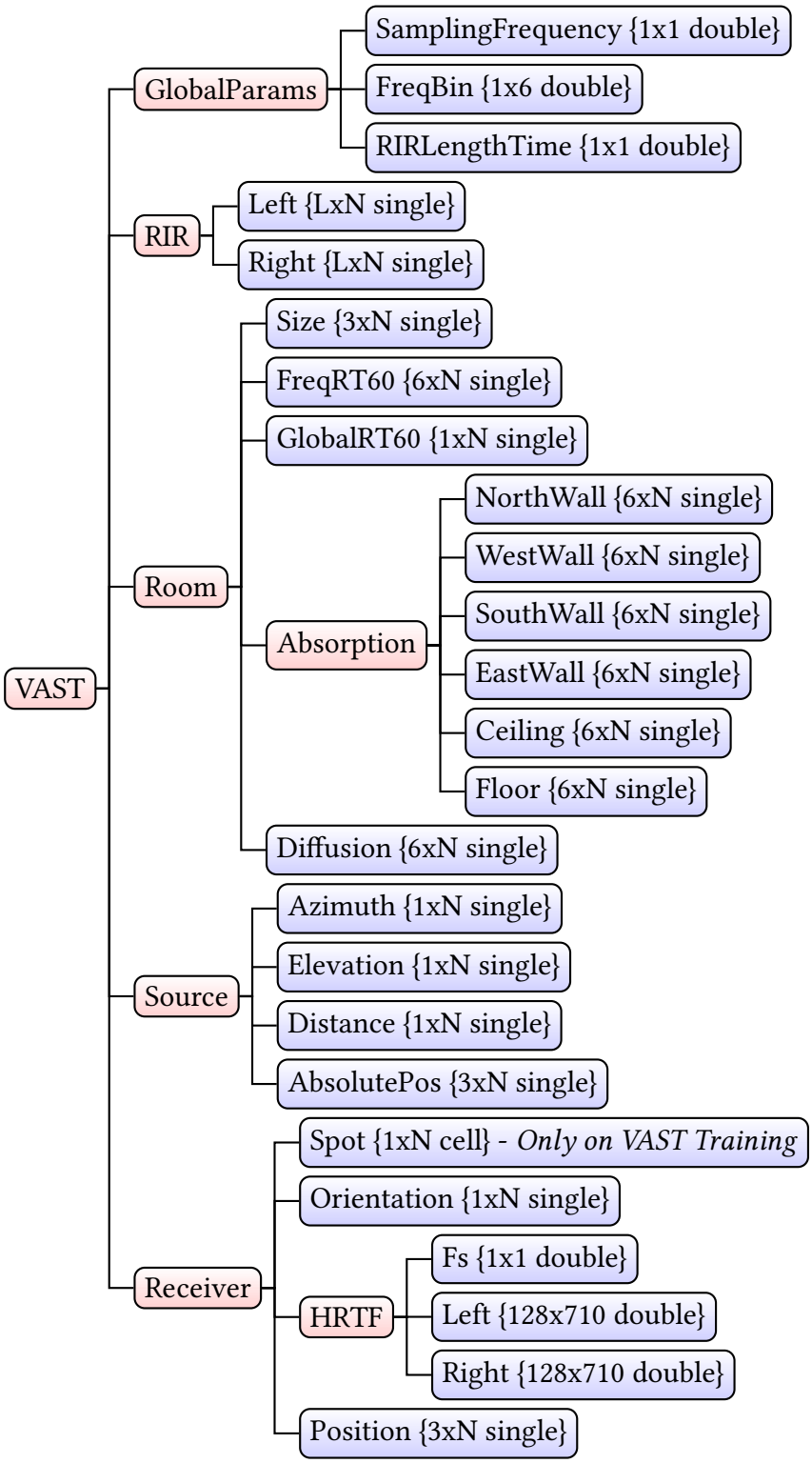


Figure C.1 – VAST dataset structure organization

Bibliography

[Aaronson & Hartmann 2014] Neil L Aaronson and William M Hartmann. *Testing, correcting, and extending the Woodworth model for interaural time difference.* The Journal of the Acoustical Society of America, vol. 135, no. 2, pages 817–823, 2014. (Cited on page 108.)

[Adler *et al.* 2012] Amir Adler, Valentin Emiya, Maria G Jafari, Michael Elad, Rémi Gribonval and Mark D Plumbley. *Audio inpainting.* IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, no. 3, pages 922–932, 2012. (Cited on pages 33, 63 and 71.)

[Adler *et al.* 2013] Aviv Adler, Michael Elad, Yacov Hel-Or and Ehud Rivlin. *Sparse coding with anomaly detection.* In IEEE International Workshop on Machine Learning for Signal Processing (MLSP), 2013, pages 1–6. IEEE, 2013. (Cited on page 24.)

[Allen & Berkley 1979] Jont B Allen and David A Berkley. *Image method for efficiently simulating small-room acoustics.* The Journal of the Acoustical Society of America, vol. 65, no. 4, pages 943–950, 1979. (Cited on page 111.)

[Andersen *et al.* 2017] Asger Heidemann Andersen, Jan Mark de Haan, Zheng-Hua Tan and Jesper Jensen. *A non-intrusive short-time objective intelligibility measure.* In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5085–5089. IEEE, 2017. (Cited on page 39.)

[Arberet *et al.* 2013] Simon Arberet, Pierre Vandergheynst, Rafael E Carrillo, Jean-Philippe Thiran and Yves Wiaux. *Sparse reverberant audio source separation via reweighted analysis.* IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 7, pages 1391–1402, 2013. (Cited on page 99.)

[Argentieri *et al.* 2015] Sylvain Argentieri, Patrick Danès and Philippe Souères. *A survey on sound source localization in robotics: From binaural to array processing methods.* Computer Speech & Language, vol. 34, no. 1, pages 87–112, 2015. (Cited on page 107.)

[Arweiler & Buchholz 2011] Iris Arweiler and Jörg M Buchholz. *The influence of spectral characteristics of early reflections on speech intelligibility.* The Journal of the Acoustical Society of America, vol. 130, no. 2, pages 996–1005, 2011. (Cited on page 96.)

[Ávila *et al.* 2017] Flavio R Ávila, Michel P Tcheou and Luiz WP Biscainho. *Audio Soft Declipping Based on Constrained Weighted Least Squares.* IEEE Signal Processing Letters, vol. 24, no. 9, pages 1348–1352, 2017. (Cited on page 64.)

[Berouti *et al.* 1979] Michael Berouti, Richard Schwartz and John Makhoul. *Enhancement of speech corrupted by acoustic noise.* In IEEE International Conference on

Acoustics, Speech, and Signal Processing, ICASSP'79., volume 4, pages 208–211. IEEE, 1979. (Cited on page 44.)

[Bertin *et al.* 2019] Nancy Bertin, Ewen Camberlein, Romain Lebarbenchon, Emmanuel Vincent, Sunit Sivasankaran, Irina Illina and Frédéric Bimbot. *VoiceHome-2, an extended corpus for multichannel speech processing in real homes*. Speech Communication, vol. 106, pages 68 – 78, 2019. (Cited on page 33.)

[Blumensath & Davies 2009] Thomas Blumensath and Mike E Davies. *Iterative hard thresholding for compressed sensing*. Applied and Computational Harmonic Analysis, vol. 27, no. 3, pages 265–274, 2009. (Cited on pages 14 and 28.)

[Boll 1979] Steven Boll. *Suppression of acoustic noise in speech using spectral subtraction*. IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 27, no. 2, pages 113–120, Apr 1979. (Cited on pages 44 and 98.)

[Boyd *et al.* 2011] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato and Jonathan Eckstein. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Foundations and Trends® in Machine Learning, vol. 3, no. 1, pages 1–122, 2011. (Cited on pages 23 and 24.)

[Brandenburg 1987] Karlheinz Brandenburg. *Evaluation of quality for audio encoding at low bit rates*. In Audio Engineering Society Convention 82. Audio Engineering Society, 1987. (Cited on page 35.)

[Brandstein & Ward 2001] Michael Brandstein and Darren Ward. Microphone arrays: signal processing techniques and applications. Springer Science & Business Media, 2001. (Cited on page 106.)

[Cecchi *et al.* 2018] Stefania Cecchi, Alberto Carini and Sascha Spors. *Room response equalization–A review*. Applied Sciences, vol. 8, no. 1, page 16, 2018. (Cited on page 98.)

[Chartrand & Wohlberg 2013] Rick Chartrand and Brendt Wohlberg. *A nonconvex ADMM algorithm for group sparsity with sparse groups*. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pages 6009–6013. IEEE, 2013. (Cited on page 24.)

[Chen *et al.* 2001] S. Chen, D. Donoho and M. Saunders. *Atomic decomposition by basis pursuit*. SIAM review, vol. 43, no. 1, pages 129–159, 2001. (Cited on page 14.)

[Cole *et al.* 1995] Ronald A Cole, Mike Noel, Terri Lander and Terry Durham. *New telephone speech corpora at CSLU*. In Fourth European Conference on Speech Communication and Technology, 1995. (Cited on page 32.)

[Combettes & Pesquet 2011] Patrick L Combettes and Jean-Christophe Pesquet. *Proximal splitting methods in signal processing*. In Fixed-point algorithms for inverse problems in science and engineering, pages 185–212. Springer, 2011. (Cited on page 24.)

[Damnjanovic *et al.* 2010] Ivan Damnjanovic, Matthew EP Davies and Mark D Plumbley. *SMALLbox - an evaluation framework for sparse representations and dictionary learning algorithms*. In International Conference on Latent Variable Analysis and Signal Separation, pages 418–425. Springer, 2010. (Cited on page viii.)

[Defraene *et al.* 2013] Bruno Defraene, Naim Mansour, Steven De Hertogh, Toon van Waterschoot, Moritz Diehl and Marc Moonen. *Declipping of audio signals using perceptual compressed sensing*. IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 12, pages 2627–2637, 2013. (Cited on pages 64 and 72.)

[Deleforge *et al.* 2015a] Antoine Deleforge, Florence Forbes and Radu Horaud. *Acoustic space learning for sound-source separation and localization on binaural manifolds*. International journal of neural systems, vol. 25, no. 01, page 1440003, 2015. (Cited on page 108.)

[Deleforge *et al.* 2015b] Antoine Deleforge, Florence Forbes and Radu Horaud. *High-dimensional regression with gaussian mixtures and partially-latent response variables*. Statistics and Computing, vol. 25, no. 5, pages 893–911, 2015. (Cited on pages 109 and 110.)

[Deleforge *et al.* 2015c] Antoine Deleforge, Radu Horaud, Yoav Y Schechner and Laurent Girin. *Co-localization of audio sources in images using binaural features and locally-linear regression*. IEEE Transactions on Audio, Speech, and Language Processing, vol. 23, no. 4, pages 718–731, 2015. (Cited on pages 108, 109 and 117.)

[Deleforge 2013] Antoine Deleforge. *Acoustic Space Mapping: A Machine Learning Approach to Sound Source Separation and Localization*. Theses, Université de Grenoble, November 2013. (Cited on page 110.)

[Deléglise *et al.* 2009] Paul Deléglise, Yannick Esteve, Sylvain Meignier and Teva Merlin. *Improvements to the LIUM French ASR system based on CMU Sphinx: what helps to significantly reduce the word error rate?* In Tenth Annual Conference of the International Speech Communication Association, 2009. (Cited on page 31.)

[DiBiase *et al.* 2001] Joseph H DiBiase, Harvey F Silverman and Michael S Brandstein. *Robust localization in reverberant rooms*. In Microphone Arrays, pages 157–180. Springer, 2001. (Cited on page 107.)

[Duarte *et al.* 2008] Marco F Duarte, Mark A Davenport, Dharmpal Takhar, Jason N Laska, Ting Sun, Kevin F Kelly and Richard G Baraniuk. *Single-pixel imaging via compressive sampling*. IEEE signal processing magazine, vol. 25, no. 2, pages 83–91, 2008. (Cited on page 10.)

[Eckstein & Yao 2015] Jonathan Eckstein and Wang Yao. *Understanding the convergence of the alternating direction method of multipliers: Theoretical and computational perspectives*. Pacific Journal on Optimization, vol. 11, no. 4, pages 619–644, 2015. (Cited on page 24.)

[Eldar & Rauhut 2010] Yonina C Eldar and Holger Rauhut. *Average case analysis of multichannel sparse recovery using convex relaxation.* IEEE Transactions on Information Theory, vol. 56, no. 1, pages 505–519, 2010. (Cited on page 20.)

[Faiz *et al.* 2012] Adil Faiz, Joël Ducourneau, Adel Khanfir and Jacques Chatillon. *Measurement of sound diffusion coefficients of scattering furnishing volumes present in workplaces.* In Acoustics 2012, 2012. (Cited on page 113.)

[Févotte & Kowalski 2015] Cédric Févotte and Matthieu Kowalski. *Hybrid sparse and low-rank time-frequency signal decomposition.* In 23rd European Signal Processing Conference (EUSIPCO), pages 464–468. IEEE, 2015. (Cited on page 30.)

[Foucart & Rauhut 2013] Simon Foucart and Holger Rauhut. A mathematical introduction to compressive sensing. Springer, 2013. (Cited on page 14.)

[Fourier 1822] Joseph Fourier. Theorie analytique de la chaleur. Chez Firmin Didot, père et fils, 1822. (Cited on page 15.)

[French & Steinberg 1947] Norman R French and John C Steinberg. *Factors governing the intelligibility of speech sounds.* The Journal of the Acoustical Society of America, vol. 19, no. 1, pages 90–119, 1947. (Cited on page 38.)

[Gardner & Martin 1995] William G Gardner and Keith D Martin. *HRTF measurements of a KEMAR.* The Journal of the Acoustical Society of America, vol. 97, no. 6, pages 3907–3908, 1995. (Cited on page 115.)

[Garofolo *et al.* 1993] John S Garofolo, Lori F Lamel, William M Fisher, Jonathon G Fiscus and David S Pallett. *DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1.* NASA STI/Recon technical report n, vol. 93, 1993. (Cited on pages viii and 33.)

[Gaultier *et al.* 2017a] Clément Gaultier, Nancy Bertin, Srđan Kitić and Rémi Gribonval. *A modeling and algorithmic framework for (non) social (co) sparse audio restoration.* arXiv preprint arXiv:1711.11259, 2017. (Cited on pages 43 and 60.)

[Gaultier *et al.* 2017b] Clément Gaultier, Saurabh Kataria and Antoine Deleforge. *VAST: The Virtual Acoustic Space Traveler dataset.* In International Conference on Latent Variable Analysis and Signal Separation, pages 68–79. Springer, 2017. (Cited on page 104.)

[Gaultier *et al.* 2017c] Clément Gaultier, Srđan Kitić, Nancy Bertin and Rémi Gribonval. *AUDASCITY: AUdio Denoising by Adaptive Social CosparsITY.* In 2017 25th European Signal Processing Conference (EUSIPCO), pages 1265–1269, Aug 2017. (Cited on page 43.)

[Gaultier *et al.* 2017d] Clément Gaultier, Srđan Kitić, Nancy Bertin and Rémi Gribonval. *Cosparse Denoising: The Importance of Being Social.* In The Signal Processing with Adaptive Sparse Structured Representations (SPARS) workshop, 2017. (Not cited.)

[Gaultier *et al.* 2018] Clément Gaultier, Nancy Bertin and Rémi Gribonval. *CASCADE: Channel-Aware Structured Cosparse Audio DEclipper*. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 571–575, April 2018. (Cited on page 60.)

[Goto *et al.* 2002] Masataka Goto, Hiroki Hashiguchi, Takuichi Nishimura and Ryuichi Oka. *RWC Music Database: Popular, Classical and Jazz Music Databases*. In ISMIR, volume 2, pages 287–288, 2002. (Cited on page viii.)

[Gramfort & Kowalski 2009] Alexandre Gramfort and Matthieu Kowalski. *Improving M/EEG source localization with an inter-condition sparse prior*. In IEEE International Symposium on Biomedical Imaging (ISBI), page 141, 2009. (Cited on page 21.)

[Graves *et al.* 2013] Alex Graves, Abdel-rahman Mohamed and Geoffrey Hinton. *Speech recognition with deep recurrent neural networks*. In 2013 IEEE international conference on Acoustics, speech and signal processing, pages 6645–6649. IEEE, 2013. (Cited on page 33.)

[Gribonval & Nikolova 2018] Rémi Gribonval and Mila Nikolova. *A characterization of proximity operators*. arXiv preprint arXiv:1807.04014, 2018. (Cited on page 26.)

[Gribonval *et al.* 2008] Rémi Gribonval, Holger Rauhut, Karin Schnass and Pierre Vandergheynst. *Atoms of All Channels, Unite! Average Case Analysis of Multi-Channel Sparse Recovery Using Greedy Algorithms*. Journal of Fourier Analysis and Applications, vol. 14, no. 5, pages 655–687, Dec 2008. (Cited on page 20.)

[Hadamard 1902] Jacques Hadamard. *Sur les problèmes aux dérivées partielles et leur signification physique*. Princeton university bulletin, vol. 13, no. 49-52, page 28, 1902. (Cited on page 12.)

[Hansen & Pellom 1998] John HL Hansen and Bryan L Pellom. *An effective quality evaluation protocol for speech enhancement algorithms*. In Fifth International Conference on Spoken Language Processing, 1998. (Cited on page 36.)

[Hargreaves *et al.* 2005] David J Hargreaves, Raymond MacDonald and Dorothy Miell. *How do people communicate using music*. Musical communication, pages 1–25, 2005. (Cited on page 32.)

[Harvilla & Stern 2014] Mark J Harvilla and Richard M Stern. *Least Squares Signal Declipping for Robust Speech Recognition*. In Fifteenth Annual Conference of the International Speech Communication Association, 2014. (Cited on pages 60, 64 and 71.)

[Hornstein *et al.* 2006] Jonas Hornstein, Manuel Lopes, José Santos-Victor and Francisco Lacerda. *Sound localization for humanoid robots-building audio-motor maps based on the HRTF*. In 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1170–1176. IEEE, 2006. (Cited on page 108.)

[Hu & Loizou 2004] Yi Hu and Philipos C Loizou. *Incorporating a psychoacoustical model in frequency domain speech enhancement.* IEEE signal processing letters, vol. 11, no. 2, pages 270–273, 2004. (Cited on page 45.)

[ITU-T 2001] ITU-T. *Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs.* Rec. ITU-T P. 862, 2001. (Cited on page 37.)

[ITU-T 2007] ITU-T. *Wideband extension to Recommendation P. 862 for the assessement of wideband telephone networks and speech codecs.* Rec. ITU-T P. 862. 2, 2007. (Cited on page 37.)

[Janssen *et al.* 1986] Augustus J. E. M. Janssen, Raymond N. J. Veldhuis and L. Vries. *Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes.* IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 34, no. 2, pages 317–330, 1986. (Cited on page 63.)

[Jenatton *et al.* 2011] Rodolphe Jenatton, Jean-Yves Audibert and Francis Bach. *Structured variable selection with sparsity-inducing norms.* The Journal of Machine Learning Research, vol. 12, pages 2777–2824, 2011. (Cited on page 19.)

[Kabal 2002] Peter Kabal. *An examination and interpretation of ITU-R BS. 1387: Perceptual evaluation of audio quality.* TSP Lab Technical Report, Dept. Electrical & Computer Engineering, McGill University, pages 1–89, 2002. (Cited on page 36.)

[Kaipio & Somersalo 2006] Jari Kaipio and Erkki Somersalo. Statistical and computational inverse problems, volume 160. Springer Science & Business Media, 2006. (Cited on page 98.)

[Kataria *et al.* 2017] Saurabh Kataria, Clément Gaultier and Antoine Deleforge. *Hearing in a shoe-box: Binaural source position and wall absorption estimation using virtually supervised learning.* In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 226–230, March 2017. (Cited on pages 104 and 111.)

[Kates & Arehart 2005] James M Kates and Kathryn H Arehart. *Coherence and the speech intelligibility index.* The Journal of the Acoustical Society of America, vol. 117, no. 4, pages 2224–2237, 2005. (Cited on page 38.)

[Kitić *et al.* 2013] Srđan Kitić, Laurent Jacques, Nilesh Madhu, Michael Peter Hopwood, Ann Spriet and Christophe De Vleeschouwer. *Consistent Iterative Hard Thresholding for signal declipping.* In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pages 5939–5943. IEEE, 2013. (Cited on pages 63, 64, 80, 83 and 84.)

[Kitić *et al.* 2014] Srđan Kitić, Nancy Bertin and Rémi Gribonval. *Hearing behind walls: localizing sources in the room next door with cosparsity.* In IEEE International

Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3087–3091, Florence, Italy, 2014. IEEE. (Cited on pages 96 and 108.)

[Kitić *et al.* 2015] Srđan Kitić, Nancy Bertin and Rémi Gribonval. *Sparsity and cosparsity for audio declipping: a flexible non-convex approach.* In Latent Variable Analysis and Signal Separation (LVA/ICA), pages 243–250. Springer, Liberec, Czech Republic, 2015. (Cited on pages 12, 24, 63, 64, 67, 80, 83, 84, 85, 86, 87, 88, 89, 90 and 91.)

[Kitić 2015] Srđan Kitić. *Cosparse regularization of physics-driven inverse problems.* PhD Thesis, IRISA, Inria Rennes, 2015. (Cited on pages 12 and 65.)

[Klatt 1982] Dennis Klatt. *Prediction of perceived phonetic distance from critical-band spectra: A first step.* In IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'82., volume 7, pages 1278–1281. IEEE, 1982. (Cited on page 37.)

[Kodrasi *et al.* 2014] Ina Kodrasi, Timo Gerkmann and Simon Doclo. *Frequency-domain single-channel inverse filtering for speech dereverberation: Theory and practice.* In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5177–5181. IEEE, 2014. (Cited on page 98.)

[Kowalski & Torrésani 2009a] Matthieu Kowalski and Bruno Torrésani. *Sparsity and persistence: mixed norms provide simple signal models with dependent coefficients.* Signal, image and video processing, vol. 3, no. 3, pages 251–264, 2009. (Cited on page 19.)

[Kowalski & Torrésani 2009b] Matthieu Kowalski and Bruno Torrésani. *Structured sparsity: from mixed norms to structured shrinkage.* In SPARS'09-Signal Processing with Adaptive Sparse Structured Representations, 2009. (Cited on page 19.)

[Kowalski *et al.* 2010] Matthieu Kowalski, Emmanuel Vincent and Rémi Gribonval. *Beyond the narrowband approximation: Wideband convex methods for under-determined reverberant audio source separation.* IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 7, pages 1818–1829, 2010. (Cited on pages 98 and 99.)

[Kowalski *et al.* 2013] Matthieu Kowalski, Kai Siedenburg and Monika Dörfler. *Social sparsity! neighborhood systems enrich structured shrinkage operators.* IEEE Transactions on Signal Processing, vol. 61, no. 10, pages 2498–2511, 2013. (Cited on pages 19 and 63.)

[Kowalski 2014] Matthieu Kowalski. *Thresholding rules and iterative shrinkage/thresholding algorithm: A convergence study.* In IEEE International Conference on Image Processing (ICIP), pages 4151–4155. IEEE, 2014. (Cited on pages 25 and 28.)

[Kuttruff 2009] Heinrich Kuttruff. Room acoustics. CRC Press, 2009. (Cited on page 96.)

[Laufer-Goldshtein *et al.* 2017] Bracha Laufer-Goldshtein, Ronen Talmon and Sharon Gannot. *Speaker Tracking on Multiple-Manifolds with Distributed Microphones*. In Petr Tichavský, Massoud Babaie-Zadeh, Olivier J.J. Michel and Nadège Thirion-Moreau, editors, Latent Variable Analysis and Signal Separation, pages 59–67, Cham, 2017. Springer International Publishing. (Cited on page 125.)

[Lebarbenchon *et al.* 2018] Romain Lebarbenchon, Ewen Camberlein, Diego Di Carlo, Clément Gaultier, Antoine Deleforge and Nancy Bertin. *Evaluation of an open-source implementation of the SRP-PHAT algorithm within the 2018 LOCATA challenge*. In 2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC), LOCATA Challenge. IEEE, 2018. (Not cited.)

[Lebart *et al.* 2001] Katia Lebart, Jean-Marc Boucher and PN Denbigh. *A new method based on spectral subtraction for speech dereverberation*. Acta Acustica united with Acustica, vol. 87, no. 3, pages 359–366, 2001. (Cited on page 98.)

[Lieb & Stark 2018] Florian Lieb and Hans-Georg Stark. *Audio inpainting: Evaluation of time-frequency representations and structured sparsity approaches*. Signal Processing, vol. 153, pages 291 – 299, 2018. (Cited on page 123.)

[Loizou 2013] Philipos C. Loizou. Speech enhancement: Theory and practice. CRC Press, Inc., Boca Raton, FL, USA, 2nd édition, 2013. (Cited on page 37.)

[Lu & Cooke 2010] Yan-Chen Lu and Martin Cooke. *Binaural estimation of sound source distance via the direct-to-reverberant energy ratio for static and moving sources*. IEEE Transactions on Audio, Speech, and Language Processing, vol. 18, no. 7, pages 1793–1805, 2010. (Cited on page 108.)

[Lustig *et al.* 2007] Michael Lustig, David Donoho and John M Pauly. *Sparse MRI: The application of compressed sensing for rapid MR imaging*. Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine, vol. 58, no. 6, pages 1182–1195, 2007. (Cited on page 10.)

[Ma *et al.* 2015] Ning Ma, Tobias May, Hagen Wierstorf and Guy J Brown. *A machine-hearing system exploiting head movements for binaural sound localisation in reverberant conditions*. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2699–2703. IEEE, 2015. (Cited on pages 115, 116, 118 and 119.)

[Mallat & Zhang 1993] Stéphane G Mallat and Zhifeng Zhang. *Matching pursuits with time-frequency dictionaries*. IEEE Transactions on Signal Processing, vol. 41, no. 12, pages 3397–3415, 1993. (Cited on page 14.)

[Mallat 1999] Stéphane Mallat. A wavelet tour of signal processing. Elsevier, 1999. (Cited on page 16.)

[Mokrỳ *et al.* 2018] Ondřej Mokrỳ, Pavel Záviška, Pavel Rajmic and Vítězslav Veselỳ. *Introducing SPAIN (SParse Audion INpainter).* arXiv preprint arXiv:1810.13137, 2018. (Cited on page 123.)

[Mourjopoulos *et al.* 1982] John Mourjopoulos, P Clarkson and J Hammond. *A comparative study of least-squares and homomorphic techniques for the inversion of mixed phase signals.* In IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'82., volume 7, pages 1858–1861. IEEE, 1982. (Cited on page 98.)

[Nam *et al.* 2011] Sangnam Nam, Michael E Davies, Michael Elad and Rémi Gribonval. *Cosparse analysis modeling-uniqueness and algorithms.* In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5804–5807. IEEE, 2011. (Cited on page 12.)

[Nam *et al.* 2013] Sangnam Nam, Michael E Davies, Michael Elad and Rémi Gribonval. *The cosparse analysis model and algorithms.* Applied and Computational Harmonic Analysis, vol. 34, no. 1, pages 30–56, 2013. (Cited on page 14.)

[Needell & Tropp 2009] Deanna Needell and Joel A Tropp. *CoSaMP: Iterative signal recovery from incomplete and inaccurate samples.* Applied and Computational Harmonic Analysis, vol. 26, no. 3, pages 301–321, 2009. (Cited on page 14.)

[Neely & Allen 1979] Stephen T Neely and Jont B Allen. *Invertibility of a room impulse response.* The Journal of the Acoustical Society of America, vol. 66, no. 1, pages 165–169, 1979. (Cited on page 98.)

[Nocedal & Wright 2006] Jorge Nocedal and Stephen Wright. Numerical optimization. Springer Science & Business Media, 2006. (Cited on page 26.)

[Ozerov *et al.* 2011] Alexey Ozerov, Cédric Févotte, Raphaël Blouet and Jean-Louis Durrieu. *Multichannel nonnegative tensor factorization with structured constraints for user-guided audio source separation.* In 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 257–260. IEEE, 2011. (Cited on page 124.)

[Ozerov *et al.* 2016] Alexey Ozerov, Çağdaş Bilen and Patrick Pérez. *Multichannel audio declipping.* In IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'16), Shanghai, China, March 2016. (Cited on pages 64 and 71.)

[Parada *et al.* 2016] Pablo Peso Parada, Dushyant Sharma, Jose Lainez, Daniel Barreda, Toon van Waterschoot and Patrick A Naylor. *A single-channel non-intrusive C50 estimator correlated with speech recognition performance.* IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 24, no. 4, pages 719–732, 2016. (Cited on page 108.)

[Parikh & Boyd 2013] Neal Parikh and Stephen Boyd. *Proximal algorithms*. Foundations and Trends in optimization, vol. 1, no. 3, pages 123–231, 2013. (Cited on page 24.)

[Pati *et al.* 1993] Yagyensh Chandra Pati, Ramin Rezaiifar and PS Krishnaprasad. *Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition*. In Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers, 1993., pages 40–44. IEEE, 1993. (Cited on page 14.)

[Perotin *et al.* 2018] Lauréline Perotin, Romain Serizel, Emmanuel Vincent and Alexandre Guérin. *CRNN-based joint azimuth and elevation localization with the Ambisonics intensity vector*. In IWAENC 2018 - 16th International Workshop on Acoustic Signal Enhancement, Tokyo, Japan, September 2018. (Cited on pages 119 and 125.)

[Perthame *et al.* 2018] Emeline Perthame, Florence Forbes and Antoine Deleforge. *Inverse regression approach to robust nonlinear high-to-low dimensional mapping*. Journal of Multivariate Analysis, vol. 163, pages 1 – 14, January 2018. (Cited on page 109.)

[Poon *et al.* 2018] Clarice Poon, Nicolas Keriven and Gabriel Peyré. *A Dual Certificates Analysis of Compressive Off-the-Grid Recovery*. arXiv preprint arXiv:1802.08464, 2018. (Cited on page 10.)

[Rabiner & Juang 1993] Lawrence R Rabiner and Biing-Hwang Juang. Fundamentals of speech recognition, volume 14. PTR Prentice Hall Englewood Cliffs, 1993. (Cited on page 31.)

[Radlovic & Kennedy 2000] Biljana D Radlovic and Rodney A Kennedy. *Nonminimum-phase equalization and its subjective importance in room acoustics*. IEEE Transactions on Speech and Audio Processing, vol. 8, no. 6, pages 728–737, 2000. (Cited on page 98.)

[Raykar *et al.* 2005] Vikas C Raykar, Ramani Duraiswami and B Yegnanarayana. *Extracting the frequencies of the pinna spectral notches in measured head related impulse responses*. The Journal of the Acoustical Society of America, vol. 118, no. 1, pages 364–374, 2005. (Cited on page 108.)

[Rencker *et al.* 2018] Lucas Rencker, Francis Bach, Wenwu Wang and Mark D. Plumbley. *Fast Iterative Shrinkage for Signal Declipping and Dequantization*. In iTWIST'18 - International Traveling Workshop on Interactions between low-complexity data models and Sensing Techniques. iTWIST, November 2018. (Cited on page 124.)

[Richards 1965] DL Richards. *Speech-transmission performance of pcm systems*. Electronics Letters, vol. 1, no. 2, pages 40–41, 1965. (Cited on page 36.)

[Sanchez-Riera *et al.* 2012] Jordi Sanchez-Riera, Xavier Alameda-Pineda, Johannes Wienke, Antoine Deleforge, Soraya Arias, Jan Čech, Sebastian Wrede and Radu

Horaud. *Online multimodal speaker detection for humanoid robots.* In 2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012), pages 126–133. IEEE, 2012. (Cited on page 117.)

[Schimmel *et al.* 2009] Steven M Schimmel, Martin F Muller and Norbert Dillier. *A fast and accurate "shoebox" room acoustics simulator.* In 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, pages 241–244. IEEE, 2009. (Cited on page 111.)

[Schroeder 1965] Manfred R Schroeder. *New method of measuring reverberation time.* The Journal of the Acoustical Society of America, vol. 37, no. 3, pages 409–412, 1965. (Cited on page 114.)

[Sharma *et al.* 2016] Dushyant Sharma, Yu Wang, Patrick A Naylor and Mike Brookes. *A data-driven non-intrusive measure of speech quality and intelligibility.* Speech Communication, vol. 80, pages 84–94, 2016. (Cited on page 39.)

[Shinn-Cunningham *et al.* 2005] Barbara G Shinn-Cunningham, Norbert Kopco and Tara J Martin. *Localizing nearby sound sources in a classroom: Binaural room impulse responses.* The Journal of the Acoustical Society of America, vol. 117, no. 5, pages 3100–3115, 2005. (Cited on pages 116, 118 and 119.)

[Siedenburg & Dörfler 2012] Kai Siedenburg and Monika Dörfler. *Audio denoising by generalized time-frequency thresholding.* In 45th Audio Engineering Society Conference: Applications of Time-Frequency Processing in Audio. Audio Engineering Society, 2012. (Cited on pages 29, 45 and 100.)

[Siedenburg *et al.* 2014] Kai Siedenburg, Matthieu Kowalski and Monika Dörfler. *Audio declipping with social sparsity.* In International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1577–1581. IEEE, 2014. (Cited on pages 28, 63, 64, 71, 80, 83, 84 and 85.)

[Søndergaard & Majdak 2013] Peter L. Søndergaard and Piotr Majdak. *The Auditory Modeling Toolbox.* In Jens Blauert, editor, The Technology of Binaural Listening, pages 33–56. Springer, Berlin, Heidelberg, 2013. (Cited on page 38.)

[Søndergaard *et al.* 2012] Peter L. Søndergaard, Bruno Torrésani and Peter Balazs. *The Linear Time Frequency Analysis Toolbox.* International Journal of Wavelets, Multiresolution Analysis and Information Processing, vol. 10, no. 4, 2012. (Cited on page 85.)

[Šorel & Bartoš 2016] Michal Šorel and Michal Bartoš. *Efficient JPEG decompression by the alternating direction method of multipliers.* In 2016 23rd International Conference on Pattern Recognition (ICPR), pages 271–276. IEEE, 2016. (Cited on page 130.)

[Sturges 1926] Herbert A Sturges. *The choice of a class interval.* Journal of the American Statistical Association, vol. 21, no. 153, pages 65–66, 1926. (Cited on page 48.)

[Taal *et al.* 2010] Cees H Taal, Richard C Hendriks, Richard Heusdens and Jesper Jensen. *A short-time objective intelligibility measure for time-frequency weighted noisy speech.* In 2010 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4214–4217. IEEE, 2010. (Cited on page 38.)

[Taal *et al.* 2011] Cees H Taal, Richard C Hendriks, Richard Heusdens and Jesper Jensen. *An algorithm for intelligibility prediction of time–frequency weighted noisy speech.* IEEE Transactions on Audio, Speech, and Language Processing, vol. 19, no. 7, pages 2125–2136, 2011. (Cited on page 38.)

[Tachioka *et al.* 2014] Yuuki Tachioka, Tomohiro Narita and Jun Ishii. *Speech recognition performance estimation for clipped speech based on objective measures.* Acoustical Science and Technology, vol. 35, no. 6, pages 324–326, 2014. (Cited on page 60.)

[Talmon *et al.* 2011] Ronen Talmon, Israel Cohen and Sharon Gannot. *Supervised source localization using diffusion kernels.* In 2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pages 245–248. IEEE, 2011. (Cited on page 108.)

[Tan *et al.* 2003] Chin-Tuan Tan, Brian CJ Moore and Nick Zacharov. *The effect of nonlinear distortion on the perceived quality of music and speech signals.* Journal of the Audio Engineering Society, vol. 51, no. 11, pages 1012–1031, 2003. (Cited on page 60.)

[Thiede *et al.* 2000] Thilo Thiede, William C Treurniet, Roland Bitto, Christian Schmidmer, Thomas Sporer, John G Beerends and Catherine Colomes. *PEAQ-The ITU standard for objective measurement of perceived audio quality.* Journal of the Audio Engineering Society, vol. 48, no. 1/2, pages 3–29, 2000. (Cited on page 35.)

[Tibshirani 1996] Robert Tibshirani. *Regression shrinkage and selection via the lasso.* Journal of the Royal Statistical Society. Series B (Methodological), pages 267–288, 1996. (Cited on page 14.)

[Tribolet *et al.* 1978] José M. Tribolet, Peter Noll, Barbara J. McDermott and Ronald E. Crochiere. *A study of complexity and quality of speech waveform coders.* In IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'78., volume 3, pages 586–590. IEEE, 1978. (Cited on page 37.)

[Tropp *et al.* 2006] Joel A. Tropp, Anna C. Gilbert and Martin J. Strauss. *Algorithms for Simultaneous Sparse Approximation. Part I: Greedy Pursuit.* Signal Processing, vol. 86, no. 3, pages 572–588, March 2006. (Cited on page 20.)

[Vaidyanathan 2007] Palghat P. Vaidyanathan. *The theory of linear prediction.* Synthesis lectures on signal processing, vol. 2, no. 1, pages 1–184, 2007. (Cited on page 36.)

[Viste & Evangelista 2003] Harald Viste and Gianpaolo Evangelista. *On the use of spatial cues to improve binaural source separation.* In Proceedings of 6th International

Conference on Digital Audio Effects (DAFx-03), pages 209–213, 2003. (Cited on pages 108 and 117.)

[Vorländer 2007] Michael Vorländer. Auralization: fundamentals of acoustics, modeling, simulation, algorithms and acoustic virtual reality. Springer Science & Business Media, 2007. (Cited on page 113.)

[Wabnitz *et al.* 2010] Andrew Wabnitz, Nicolas Epain, Craig Jin and André Van Schaik. *Room acoustics simulation for multichannel microphone arrays.* In Proceedings of the International Symposium on Room Acoustics, pages 1–6, 2010. (Cited on page 111.)

[Warzybok *et al.* 2013] Anna Warzybok, Jan Rennies, Thomas Brand, Simon Doclo and Birger Kollmeier. *Effects of spatial and temporal integration of a single early reflection on speech intelligibility.* The Journal of the Acoustical Society of America, vol. 133, no. 1, pages 269–282, 2013. (Cited on page 96.)

[Wiener 1949] Norbert Wiener. *Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications.* 1949. (Cited on page 44.)

[Yang & Yuan 2013] Junfeng Yang and Xiaoming Yuan. *Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization.* Mathematics of computation, vol. 82, no. 281, pages 301–329, 2013. (Cited on page 129.)

[Yoshioka *et al.* 2012] Takuya Yoshioka, Armin Sehr, Marc Delcroix, Keisuke Kinoshita, Roland Maas, Tomohiro Nakatani and Walter Kellermann. *Making machines understand us in reverberant rooms: Robustness against reverberation for automatic speech recognition.* IEEE Signal Processing Magazine, vol. 29, no. 6, pages 114–126, 2012. (Cited on page 96.)

[Yu *et al.* 2008] Guoshen Yu, Stéphane Mallat and Emmanuel Bacry. *Audio denoising by time-frequency block thresholding.* IEEE Transactions on Signal Processing, vol. 56, no. 5, pages 1830–1839, 2008. (Cited on pages ix, 45, 49, 50, 53 and 54.)

[Záviška *et al.* 2018] Pavel Záviška, Pavel Rajmic, Zdeněk Průša and Vítězslav Veselý. *Revisiting Synthesis Model in Sparse Audio Declipper.* In International Conference on Latent Variable Analysis and Signal Separation, pages 429–445. Springer, 2018. (Cited on pages 65, 79 and 130.)

This bibliography contains 135 references.

UNIVERSITE BRETAGNE LOIRE / MATHSTIC

UNIVERSITÉ DE RENNES 1

Titre : Conception et évaluation de modèles parcimonieux et d'algorithmes pour la résolution de problèmes inverses en audio

Mots clés : problèmes inverses, parcimonie, traitement du signal, multicanal, restauration sonore

Résumé : Dans le contexte général de la résolution de problèmes inverses en acoustique et traitement du signal audio les défis sont nombreux. Pour la résolution de ces problèmes, leur caractère souvent mal-posé nécessite de considérer des modèles de signaux appropriés. Les travaux de cette thèse montrent sur la base d'un cadre algorithmique générique polyvalent comment les différentes formes de parcimonie (à l'analyse ou à la synthèse, simple, structurée ou sociale) sont particulièrement adaptées à la reconstruction de signaux sonores dans un cadre mono ou multicanal. Le cœur des travaux de thèse permet de mettre en évidence les limites des conditions d'évaluation de l'état de l'art pour le problème de dé-saturation et de mettre en place un protocole rigoureux d'évaluation à grande échelle pour identifier les méthodes les plus appropriées en fonction du contexte (musique ou parole, signaux fortement ou faiblement dégradés). On démontre des améliorations de qualité substantielles par rapport à l'état de l'art dans certains régimes avec des configurations qui n'avaient pas été précédemment considérées, nous obtenons également des accélérations conséquentes. Enfin, un volet des travaux aborde la localisation de sources sonores sous l'angle de l'apprentissage statistique « virtuellement supervisé ». On montre avec cette méthode des résultats encourageants sur l'estimation de directions d'arrivée et de distance.

Title: Design and evaluation of sparse models and algorithms for audio inverse problems

Keywords: inverse problems, sparsity, signal processing, multichannel, audio restoration

Abstract: Today's challenges in the context of audio and acoustic signal processing inverse problems are multiform. Addressing these problems often requires additional appropriate signal models due to their inherent ill-posedness. This work focuses on designing and evaluating audio reconstruction algorithms. Thus, it shows how various sparse models (analysis, synthesis, plain, structured or “social”) are particularly suited for single or multichannel audio signal reconstruction. The core of this work notably identifies the limits of state-of-the-art methods evaluation for audio declipping and proposes a rigourous large-scale evaluation protocol to determine the more appropriate methods depending on the context (music or speech, moderately or highly degraded signals). Experimental results demonstrate substantial quality improvements for some newly considered testing configurations. We also show computational efficiency of the different methods and considerable speed improvements. Additionally, a part of this work is dedicated to the sound source localization problem. We address it with a “virtually supervised” machine learning technique. Experiments show with this method promising results on distance and direction of arrival estimation.