



# THÈSE

PRÉSENTÉE À

L'UNIVERSITÉ BORDEAUX 1

ÉCOLE DOCTORALE DE MATHÉMATIQUES ET INFORMATIQUE

par Stanislaw Gorlow

POUR OBTENIR LE GRADE DE

DOCTEUR

SPÉCIALITÉ INFORMATIQUE

RÉTROINGÉNIERIE DU SON POUR L'ÉCOUTE ACTIVE ET  
AUTRES APPLICATIONS

Reverse Audio Engineering for Active Listening and Other  
Applications

Thèse dirigée par Sylvain Marchand

Soutenue le 16 décembre 2013

Devant la commission d'examen formée de :

|  |                    |
|--|--------------------|
| M. Emmanuel Vincent, Chargé de recherches HDR, Inria Nancy – Grand Est | Rapporteur         |
| M. Karlheinz Brandenburg, Professeur, Ilmenau University of Technology | Rapporteur         |
| M. Udo Zölzer*, Professeur, Helmut Schmidt University                  | Examineur          |
| M. François Pachet, Chercheur HDR, Sony CSL Paris                      | Examineur          |
| Mme Myriam Desainte-Catherine, Professeur, Université de Bordeaux      | Examineur          |
| M. Sylvain Marchand, Professeur, Université de Bretagne Occidentale    | Directeur de thèse |

\*Président du jury

*Université Bordeaux 1*

*Les Sciences et les Technologies au service de l'Homme et de l'environnement*



## ABSTRACT

---

This work deals with the problem of reverse audio engineering for active listening. The format under consideration corresponds to the audio CD. The musical content is viewed as the result of a concatenation of the composition, the recording, the mixing, and the mastering. The inversion of the two latter stages constitutes the core of the problem at hand. The audio signal is treated as a post-nonlinear mixture. Thus, the mixture is “decompressed” before being “decomposed” into audio tracks. The problem is tackled in an informed context: The inversion is accompanied by information which is specific to the content production. In this manner, the quality of the inversion is significantly improved. The information is reduced in size by the use of quantification and coding methods, and some facts on psychoacoustics. The proposed methods are applicable in real time and have a low complexity. The obtained results advance the state of the art and contribute new insights.

*Key words*—Active listening, informed inversion of dynamic range compression, informed source separation, informed spatial filtering under linear constraint, multichannel object-based audio coding, reverse audio engineering

## RÉSUMÉ

---

Ce travail s'intéresse au problème de la rétroingénierie du son pour l'écoute active. Le format considéré correspond au CD audio. Le contenu musical est vu comme le résultat d'un enchaînement de la composition, l'enregistrement, le mixage et le mastering. L'inversion des deux dernières étapes constitue le fond du problème présent. Le signal audio est traité comme un mélange post-non-linéaire. Ainsi, le mélange est « décompressé » avant d'être « décomposé » en pistes audio. Le problème est abordé dans un contexte informé : l'inversion est accompagnée d'une information qui est spécifique à la production du contenu. De cette manière, la qualité de l'inversion est significativement améliorée. L'information est réduite de taille en se servant des méthodes de quantification, codage, et des faits sur la psychoacoustique. Les méthodes proposées s'appliquent en temps réel et montrent une complexité basse. Les résultats obtenus améliorent l'état de l'art et contribuent aux nouvelles connaissances.

*Mots clefs*—Codage audio multipiste basé sur l'objet, écoute active, filtrage spatial informé sous contrainte linéaire, inversion informée de la compression de la dynamique sonore, rétroingénierie du son, séparation de sources informée

Dans la plupart des cas, les enregistrements musicaux professionnels se soumettent à deux procédés, le mixage et le mastering, et beaucoup de formats de distribution habituels, comme par exemple la norme Compact Disc Digital Audio (CDDA), sont strictement en stéréo. Tandis que le terme « mastering » se réfère au procédé d'optimisation du mix final ainsi qu'au transfert de ce dernier sur un dispositif de stockage de données, le terme « mixage » se réfère au procédé d'assemblage de plusieurs couches d'audio préenregistrées et éditées dans un mélange composé. Chaque couche peut être imaginée comme une piste d'une table de mixage, qui alors représente une source sonore. Une source peut être à la fois une voix, un instrument, ou un groupe desdits. Le placement apparent des sources entre les haut-parleurs dans l'espace sonore stéréo est aussi connu en tant que « imaging » dans l'ingénierie du son professionnelle.

L'ingénierie du son est une discipline établie qui est employée dans plusieurs domaines qui font partie de notre vie quotidienne sans que l'on ne s'en rende compte. Pourtant, peu de gens savent comment l'audio a été produit. Si l'on considère l'enregistrement et la reproduction du son ou bien la diffusion comme exemple, on peut imaginer qu'un signal préenregistré provenant d'une source acoustique soit altéré par un ingénieur du son d'une telle manière qu'il corresponde à un certain critère pendant qu'il est joué. Le nombre de ces critères peut être grand et il dépend souvent du contexte. En général, ladite altération du signal à l'entrée est une séquence de nombreuses transformations en avant, la réversibilité desquelles porte peu ou aucun intérêt. Mais qu'en est-il de celui qui voulait faire exactement cela, c'est-à-dire d'inverser la chaîne de transformation, et ce qui est plus important, d'une manière systématique et répétitive? Cela permettrait à l'auditeur d'interagir avec ou réinterpréter la musique stockée, par exemple, sur un CD audio et ainsi de devenir actif. Pour rendre cela possible, un enregistrement musical doit être décomposé en ses éléments constitutants — au moins partiellement.

L'écoute active est un concept de la technologie musicale développé à la fin des années 1990 par Pachet et Delerue. Celui-ci a pour objectif de donner à l'utilisateur un certain degré de contrôle de la musique qu'il écoute. Au contraire du cas habituel, dans lequel un morceau de musique est simplement joué sans aucune modification conceptuelle, l'écoute active permet une expérience personnalisée de la musique à travers la réinterprétation. Son objectif est de pourvoir l'auditeur d'un plus haut niveau de confort musical et de lui donner accès à une nouvelle musique en créant des environnements sonores pour des répertoires de la musique déjà créée, dans lesquels les modifications conservent la sémantique. Un tel exemple est la réspatialisation de sources sonores ou le remixage en changeant le volume et/ou l'angle panoramique d'une source distincte. Donc, afin de mettre en

œuvre l'écoute active, il faut reconquérir l'accès aux composantes des sources latentes étant donné le mix masterisé, à savoir exercer la rétro-ingénierie du mix.

L'objectif de la rétroingénierie du son peut être soit d'identifier les paramètres de transformation étant donné les signaux d'entrée et de sortie soit de récupérer le signal d'entrée qui appartient au signal de sortie étant donné les paramètres de transformation, ou les deux. Un modèle de signal et de système explicite est obligatoire dans chacun des cas. Le deuxième cas peut sembler banal, mais seulement si la transformation est linéaire et orthogonale et en tant que telle parfaitement inversible. Pourtant, la transformation en avant est souvent ni linéaire ni inversible. C'est par exemple le cas pour la compression de la dynamique sonore, qui habituellement est décrite par un système dynamique non-linéaire à paramètres variant dans le temps. La théorie classique de systèmes linéaires à paramètres invariants dans le temps ne s'applique pas dans ce cas, donc, une solution sur mesure doit être trouvée. De plus, pour la raison que le nombre des pistes utilisées est normalement supérieur au nombre des canaux du mélange, le mixage est sous-déterminé mathématiquement parlant. En conséquence, le démixage constitue un problème de séparation de sources qui est mal posé.

Dans son tutoriel de 2005, Knuth se sert de l'approche Bayésienne pour le design des algorithmes de séparation de sources robustes qui profitent d'un savoir a priori du problème présent pour assurer que l'on atteint une solution optimale. Il l'appelle la séparation de sources « informée » pour distinguer l'approche de la séparation de sources « aveugle » où il y a peu d'information. L'idée de base de son approche informée consiste à introduire l'information préalable du problème, qui peut être une propriété physique ou de même une loi physique, dans la solution. C'est sans doute plus spécifique que le théorème de Bayes, qui ne considère que des distributions a priori. Le modèle qui accompagne un tel problème de séparation de sources informée est donc une combinaison des distributions et de l'information a priori.

L'idée peut être appliquée au problème de la rétroingénierie par la manière suivante. Tout d'abord, le mélange est considéré comme post-non-linéaire, sous l'hypothèse que le mixage linéaire soit suivi par le mastering non-linéaire. Ensuite, le mixage et le mastering peuvent être décrits par deux systèmes déterminés en cascade. Le composant probabiliste dans le modèle est dû à la tâche mal posée. La probabilité d'entendre la source d'intérêt dans le mélange est donnée par la distribution a posteriori du signal de la source respective. La connaissance à propos de la production d'un enregistrement musical représente le savoir a priori que nous disposons. C'est typiquement les effets utilisés et le positionnement spatial des sources. L'information a priori comprend les paramètres de système, soit les coefficients des filtres et les angles panoramiques ou directions, ainsi que les paramètres de

distributions de sources, soit les paramètres de signal. Ces derniers représentent les paramètres hyper d'après Bayes.

Selon le dernier paragraphe, il est évident que la meilleure estimation possible des signaux de source est atteinte seulement quand l'information a priori sous la forme de paramètres de système et de signal est parfaitement donnée. Pour cette raison, le problème à résoudre est formulé comme suit. Étant donné les modèles de mixage, de mastering, et de sources, on cherche à récupérer les estimations des signaux de source à partir du mélange de meilleure qualité possible. Les paramètres des signaux sont estimés à partir des signaux de source originaux qui sont accessibles. Les paramètres de mixage et de mastering sont censés être connus. Le problème décrit ci-dessus a lieu d'être seulement si la récupération des composants de source est conduite sans avoir accès aux originaux. Sinon, le problème est banal. L'approche discutée doit être donc appliquée dans un scénario où la « collection » de l'information a priori est découpée de son emploi. Une fois que les sources sont mélangées, elles ne sont plus disponibles. L'approche peut être aussi caractérisée par l'accès aux signaux de source qui est limité temporellement et localement. Ainsi, on doit distinguer le processus de la création de contenu du processus de la consommation de contenu. Le créateur de contenu est responsable de fournir toutes les données nécessaires au consommateur de contenu, afin que ce dernier puisse exercer la rétroingénierie du mix de manière inaperçue et de remixer le contenu ad libitum grâce à une interface graphique. Ceci est réalisé dans un framework « encodeur/décodeur ». La tâche de l'encodeur est d'extraire un minimum de données supplémentaires provenant des signaux de source pour que le décodeur puisse récupérer leurs répliques à partir du mélange à haute qualité sonore. Ces métadonnées, si elles sont suffisamment petites, peuvent être cachées dans le signal de mélange sous la forme d'un filigrane inaudible soit doivent être rattachées au mélange lui-même. En dehors du problème évident de la rétroingénierie, il reste aussi le problème de la réduction de données concernant les paramètres du modèle, qui n'est pas négligeable. Le problème est d'autant plus difficile quand les métadonnées doivent être cachées dans le mélange ou quand le décodeur doit être exécuté en temps réel. Il mérite aussi de se questionner à propos du taux de transfert suffisant pour atteindre une bonne performance du système.

La cascade aux deux étapes qui est proposée se base sur un modèle simplifié de la chaîne de production de la musique, qui consiste en une superposition de pistes monocanales ou bicanales spatialisées et éventuellement traitées avec un égaliseur dans l'étape de mixage et en la compression de la dynamique sonore dans l'étape de mastering. La borne supérieure de performance de la cascade est principalement due à la complexité sonore de la musique commerciale. Notamment, celle-ci dépend du degré d'interférence des spectres des pistes con-

stituant et de comment les sources sont distribuées dans l'espace sonore. La qualité du son après le démixage est sujette au dit « gain vectoriel ». Celui-ci s'est révélé être une fonction de a) les densités spectrales de puissance et b) le système de mixage. En règle générale, moins les spectres en question interfèrent et plus ils sont séparés dans l'espace, mieux est la qualité du son à la sortie.

La connaissance du mixage et du mastering peut être vue comme un défaut du schéma, qui le rend difficilement applicable à la musique existante, pour laquelle ce genre d'information est indisponible. Et même si des outils comme la factorisation en matrices non-négatives pour apprendre les spectrogrammes existent, sa performance est limitée. Premièrement, car cette factorisation n'est pas unique, et deuxièmement, sa fonction-coût n'est pas convexe. Dans le schéma proposé, comme dans n'importe quel schéma basé sur un modèle, une déviation des paramètres exacts provoquera une erreur additionnelle dans le résultat. Le décodeur est le plus efficace quand il est suppléé par une information très précise d'un encodeur accompagnant — ou un estimateur.

Les résultats obtenus permettent de tirer les conclusions suivantes. L'algorithme proposé est capable de fournir des estimations de signaux de source qui sont perçues comme étant plus proches aux originaux que les algorithmes de type similaire publiés. Aussi, il mérite d'être mentionné que l'algorithme n'impose aucune restriction sur le nombre de sources ni sur leur interférence spectrale. Au contraire, il s'adapte à la constellation de signal donnée et livre les meilleures estimations possibles en temps quasi-linéaire en respectant une contrainte de puissance. Donnant une haute ressemblance aux signaux originaux au taux d'information d'accompagnement assez tolérable, étant d'environ 10 kbps par source ou canal, l'algorithme est bien adapté pour des applications d'écoute active en temps réel. Le filtre à variance minimum conservant la puissance se comporte perceptivement mieux qu'un filtre du type Wiener pour un mélange instantané aussi bien que pour un mélange convolutif. La contrainte de puissance égale garantit que les répliques récupérées retiennent le timbre et la bande passante auditive des originaux. De plus, il a été observé que les estimations bruitées sont plus appréciées par l'auditeur en comparaison avec les répliques sonnantes « perforées » ou « sourdes ». Avec l'approche proposée, le mix peut être décomposé en pistes séparées ou en objets du premier plan et le fond, et de la même manière la voix peut être séparée de l'instrumental pour karaoké. Au-delà de cela, il est possible d'extraire des images spatiales de sources sans changer leur placement. La séparation de sources informée (SSI) peut être vue comme un formalisme nouveau pour un principe de codage connu, puisqu'il possède les avantages suivants. SSI se laisse réaliser dans un framework modulaire qui est d'un côté rétrocompatible et d'un autre côté il a une compatibilité ascendante : le filtre utilisé peut être adapté

à un plus grand nombre de canaux. En plus, on a pu observer qu'un rapport signal sur bruit (RSB) à fréquence pondérée peut donner des résultats aussi fiables que les métriques qui modélisent la perception humaine — mais à un coût beaucoup plus bas. Cependant, l'inconsistance entre les métriques de performance différentes rend des tests d'écoute toujours indispensable.

La performance d'un estimateur linéaire est bornée si le mélange est sous-déterminé à cause de sa résolution limitée. Cela provoque la nécessité de transmettre le résidu ou d'augmenter le nombre de canaux pour atteindre une meilleure qualité. La qualité d'un remix est largement satisfaisante si le mix est compressé sans perte. Sinon, la qualité est sensiblement détériorée. En connaissant les paramètres qui ont été utilisés pour la compression de la dynamique sonore, il est possible de récupérer le signal non-compressé du signal compressé avec une haute précision et un effort de calcul relativement bas. Si la compression est défaite avec une erreur négligeable, les signaux demixés sont presque identiques à ceux obtenus à partir d'un mix non-compressé. Le décompresseur est nécessaire pour éviter des artefacts qui ne sont possiblement pas entendus dans le mix. Comme mentionné précédemment, la qualité du son dépend du gain vectoriel qui augmente avec le nombre des canaux de mélange. Pour des applications de codage, le système de mixage peut être choisi arbitrairement et un nombre distinct de pistes peut être mixé dans un nombre de canaux inférieur de telle manière que l'intégralité des pistes décodées montre le même rapport signal sur interférence (RSI). La qualité du son peut être prédite à l'encodeur. Ainsi, le nombre de canaux peut être choisi avant le mixage réel et le niveau de qualité après le demixage peut être contrôlé a priori.



## LABORATORY

---

The thesis has been prepared in the following laboratory:

Laboratoire Bordelais de Recherche en Informatique  
UMR CNRS 5800  
351, cours de la Libération  
33405 Talence Cedex  
France



## ACKNOWLEDGMENTS

---

First of all, I would like to thank Sylvain Marchand for letting me participate in the **DReaM** project while giving me the chance and the freedom to develop my own ideas. I know it has not been easy being confronted with my way of looking at things and/or my convictions. Thank you for your advocacy and your patience. I hope it paid off in the end and that I have taken a step forward in becoming what you consider a good researcher. I would also like to express my gratitude to Josh Reiss for welcoming me in his group at the Centre for Digital Music in London. Part of the work was undertaken there during my research stay. Thank you and Emanuël Habets from the International Audio Laboratories Erlangen for proofreading the papers which you have co-authored. Your comments were always constructive, on time, and on point. In the same breath, I would like to thank Corey Cheng for his editorial review of one of my papers.

Then, I would like to thank my project partners for the passionate discussions that we led over the three years: Roland Badeau, Laurent Daudet, Dominique Fourer, Laurent Girin, Antoine Liutkus, Jonathan Pinel, Gaël Richard, Nicolas Sturmel, and not least Shuhua Zhang. In particular, I want to thank Dominique for lending me an ear in both scientific and personal matters and for his loyalty. Cordial thanks to Jonathan for sharing the results of his work. Further, I would like to thank Gaël for teaching me about competition in academic research as much as Laurent Girin for sharing insight into originality of work, authorship, and the optimality of the Wiener filter.

My deepest gratitude goes to all the team assistants who made my life easier. Thank you Nicole Lun, Lebna Mizani, Brigitte Cudeville, Maité Labrousse, Elia Meyre, Isabelle Garcia, and Philippe Biais. Not to forget are Cathy Roubineau and Magali Hinnenberger. Thank you both for your help. I could talk to you for hours. Beyond, I also want to mention Yves Métivier, Nicolas Hanusse, and Christophe Bavard (the latter two from the doctoral school), as the ones who provided me with financial support for a conference in Paris and the research stay in London. Many thanks to Elodie Duru, Christian Massus, and Jacky Chartier from “Aquitaine Science Transfert” for their support with the French paperwork. Also, I would like to thank Ben Dawson and DJ Vadim for their cooperation on the multitracks.

Last but not least, I would like to thank Joana for being the anchor in stormy weather and for not letting me lose my mind completely. I thank you all very much for what you have said or done that helped me finish my thesis.



Dedicated in loving memory to Waldemar Stab  
27 September 1924 – 4 May 2004



## PUBLICATIONS

---

Some ideas and figures have appeared previously in the following publications:

### JOURNAL ARTICLES

- [1] S. Gorlow and J. D. Reiss, "Model-based inversion of dynamic range compression," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1434–1444, July 2013.
- [2] S. Gorlow and S. Marchand, "Informed audio source separation using linearly constrained spatial filters," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 1, pp. 3–13, January 2013.

### CONFERENCE PAPERS

- [1] S. Gorlow and S. Marchand, "Informed separation of spatial images of stereo music recordings using second-order statistics," in *Proceedings of the 2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, September 2013, pp. 1–6.
- [2] —, "Reverse engineering stereo music recordings pursuing an informed two-stage approach," in *Proceedings of the 2013 International Conference on Digital Audio Effects (DAFx)*, September 2013, pp. 1–8.
- [3] S. Gorlow, E. A. P. Habets, and S. Marchand, "Multichannel object-based audio coding with controllable quality," in *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2013, pp. 561–565.
- [4] S. Gorlow and S. Marchand, "On the informed source separation approach for interactive remixing in stereo," in *Audio Engineering Society (AES) Convention 134*, May 2013, pp. 1–10.
- [5] S. Marchand, R. Badeau, C. Baras, L. Daudet, D. Fourer, L. Girin, S. Gorlow, A. Liutkus, J. Pinel, G. Richard, N. Sturmel, and S. Zhang, "DReaM: A novel system for joint source separation and multitrack coding," in *Audio Engineering Society (AES) Convention 133*, October 2012, pp. 1–10.
- [6] A. Liutkus, S. Gorlow, N. Sturmel, S. Zhang, L. Girin, R. Badeau, L. Daudet, S. Marchand, and G. Richard, "Informed audio source separation: A comparative study," in *Proceedings of the*

2012 *European Signal Processing Conference (EUSIPCO)*, August 2012, pp. 2397–2401.

- [7] S. Gorlow and S. Marchand, “Informed source separation: Underdetermined source signal recovery from an instantaneous stereo mixture,” in *Proceedings of the 2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, October 2011, pp. 309–312.

#### PATENTS

- [1] S. Gorlow and S. Marchand, “Method and device for separating signals by iterative spatial filtering,” International Patent 053 631, April 18, 2013.
- [2] S. Gorlow and J. D. Reiss, “A method for inverting dynamic range compression of a digital audio signal,” International Patent Application 13 050 461, March 4, 2013.
- [3] S. Gorlow and S. Marchand, “Procédé et dispositif pour séparer des signaux par filtrage spatial à variance minimum sous contrainte linéaire,” French Patent Application 1 259 115, September 27, 2012.



## CONTENTS

---

|            |   |           |
|------------|---|-----------|
| <b>i</b>   | <b>INTRODUCTION</b>                       | <b>1</b>  |
| 1          | INTRODUCTION                              | 3         |
| 1.1        | Motivation                                | 3         |
| 1.2        | Objectives                                | 3         |
| 1.2.1      | Active Listening                          | 3         |
| 1.2.2      | Reverse Audio Engineering                 | 4         |
| 1.3        | Organization                              | 4         |
| <b>ii</b>  | <b>PRELIMINARIES</b>                      | <b>7</b>  |
| 2          | DIGITAL AUDIO ENGINEERING                 | 9         |
| 2.1        | Audio Effects and Audio Signal Processors | 9         |
| 2.2        | Mixing                                    | 11        |
| 2.3        | Mastering                                 | 11        |
| 3          | ILL-POSED INVERSE PROBLEMS                | 13        |
| 3.1        | Inverse Problems                          | 13        |
| 3.2        | Ill-Posedness                             | 13        |
| 4          | PROBABILISTIC APPROACH                    | 15        |
| 4.1        | Bayesian Inference                        | 15        |
| 4.2        | Linear Optimum Filtering                  | 16        |
| 4.3        | Post-Nonlinear Mixtures                   | 17        |
| 5          | INFORMED APPROACH                         | 19        |
| 5.1        | Basic Idea                                | 19        |
| 5.2        | Concretization                            | 19        |
| 5.2.1      | Problem Formulation                       | 20        |
| 5.2.2      | Proposed Solution                         | 20        |
| 5.2.3      | Associated Problem                        | 20        |
| <b>iii</b> | <b>REVERSE AUDIO ENGINEERING</b>          | <b>21</b> |
| 6          | PREVIOUS WORKS                            | 23        |
| 7          | AUDIO SOURCE SEPARATION                   | 27        |
| 7.1        | Intensity Stereophony                     | 27        |
| 7.2        | Signal Model                              | 27        |
| 7.3        | System Model                              | 28        |
| 7.3.1      | Image Signal                              | 28        |
| 7.3.2      | Mixture Signal                            | 29        |
| 7.4        | Problem Formulation                       | 29        |
| 7.5        | Proposed Solution                         | 30        |
| 7.5.1      | Well-Posed Case                           | 30        |
| 7.5.2      | Ill-Posed Case                            | 32        |
| 7.5.3      | Narrowband Convolutional Case             | 33        |
| 7.5.4      | Two-Channel Case                          | 34        |

|       |   |    |  |
|-------|---|----|--|
| 7.5.5 | Noisy Case                              | 35 |  |
| 7.6   | General Remarks                         | 35 |  |
| 7.6.1 | Precedence or Haas Effect               | 35 |  |
| 7.6.2 | Multi-Constraint Spatial Filter         | 36 |  |
| 7.6.3 | Wiener vs. PCMV Spatial Filter          | 36 |  |
| 7.6.4 | Mixture Covariance Matrix               | 37 |  |
| 7.7   | Parameter Quantization and Coding       | 38 |  |
| 7.7.1 | Panoramic Angle                         | 38 |  |
| 7.7.2 | Balance Ratio                           | 38 |  |
| 7.7.3 | Short-Time Power Spectrum               | 38 |  |
| 7.8   | Performance Evaluation                  | 40 |  |
| 7.8.1 | Instantaneous Mixture with Watermarking | 40 |  |
| 7.8.2 | Narrowband Convolutional Mixture        | 42 |  |
| 7.8.3 | Separation of Spatial Images            | 44 |  |
| 8     | COMPARISON WITH MPEG SAOC               | 59 |  |
| 8.1   | Enhanced Audio Object Separation        | 59 |  |
| 8.2   | Object Encoder                          | 59 |  |
| 8.3   | Object Decoder                          | 61 |  |
| 8.4   | Performance Evaluation                  | 61 |  |
| 8.4.1 | EAOS vs. USSR                           | 61 |  |
| 8.4.2 | Interactive Remixing                    | 63 |  |
| 9     | INVERSION OF DYNAMIC RANGE COMPRESSION  | 67 |  |
| 9.1   | Dynamic Range Compression               | 67 |  |
| 9.2   | System Model                            | 69 |  |
| 9.2.1 | Feed-Forward Broadband Compression      | 69 |  |
| 9.2.2 | Stereo Linking                          | 70 |  |
| 9.3   | Problem Formulation                     | 70 |  |
| 9.4   | Proposed Solution                       | 71 |  |
| 9.4.1 | Characteristic Function                 | 72 |  |
| 9.4.2 | Attack-Release Phase Toggle             | 73 |  |
| 9.4.3 | Envelope Predictor                      | 74 |  |
| 9.4.4 | Stereo Unlinking                        | 74 |  |
| 9.4.5 | Error Analysis                          | 74 |  |
| 9.5   | Numerical Approximation                 | 76 |  |
| 9.6   | General Remarks                         | 77 |  |
| 9.6.1 | Lookahead                               | 77 |  |
| 9.6.2 | Clipping and Limiting                   | 77 |  |
| 9.6.3 | Logarithmic Gain Smoothing              | 78 |  |
| 9.7   | Pseudocode                              | 78 |  |
| 9.8   | Parameter Quantization and Coding       | 78 |  |
| 9.9   | Performance Evaluation                  | 78 |  |
| 9.9.1 | Experimental Setup                      | 78 |  |
| 9.9.2 | Experimental Results                    | 80 |  |
| 10    | TWO-STAGE CASCADE CONNECTION            | 89 |  |
| 10.1  | System Overview                         | 89 |  |
| 10.2  | Cascade Encoder                         | 89 |  |

|        |  |     |
|--------|--|-----|
| 10.3   | Cascade Decoder                              | 89  |
| 10.4   | Performance Characteristics                  | 91  |
| 10.4.1 | Algorithmic Delay                            | 91  |
| 10.4.2 | Computational Complexity                     | 91  |
| 10.4.3 | Side-Information Rate                        | 91  |
| 10.5   | Performance Evaluation                       | 93  |
| 10.5.1 | Experimental Setup                           | 93  |
| 10.5.2 | Experimental Results                         | 94  |
| 11     | MULTICHANNEL OBJECT-BASED AUDIO CODING       | 97  |
| 11.1   | Introduction                                 | 97  |
| 11.2   | Signal and System Model                      | 98  |
| 11.3   | Problem Formulation                          | 98  |
| 11.4   | Proposed Solution                            | 98  |
| 11.4.1 | System Overview                              | 98  |
| 11.4.2 | Mixdown                                      | 99  |
| 11.4.3 | Source Separation                            | 102 |
| 11.5   | Quality Control Mechanism                    | 103 |
| 11.5.1 | Quality Metrics                              | 103 |
| 11.5.2 | Control Mechanism                            | 104 |
| 11.6   | Performance Evaluation                       | 104 |
| 11.6.1 | Experimental Setup                           | 104 |
| 11.6.2 | Experimental Results                         | 105 |
| iv     | CONCLUSION                                   | 107 |
| 12     | CONCLUSION AND FUTURE OUTLOOK                | 109 |
| 12.1   | Conclusion                                   | 109 |
| 12.2   | Future Outlook                               | 111 |
|        | BIBLIOGRAPHY                                 | 113 |
| A      | SIGNAL ATTENUATION WITH WIENER FILTERING     | 123 |
| B      | MODEL PARAMETER ESTIMATION                   | 125 |
| C      | INVERTIBILITY OF A COMPRESSOR WITH LOOKAHEAD | 127 |

## LIST OF FIGURES

---

- Figure 1 Modeling of sound sources in a sound field using the parameters *direction* (azimuth) and *volume* (radius) 28
- Figure 2 Spatial pattern and power distribution of a two-channel filter. The beam is gain adjusted and directed such that the signal of interest is preserved and either one interferer is suppressed or, in case of multiple interferers, their mean total power is minimized. 31
- Figure 3 *SIR* and *TPS* values (left column) complemented by the median, the 25th and 75th percentiles, and outliers (right column) for an excerpt from “The Terrorist” by DJ Vadim 47
- Figure 4 *SIR* and *TPS* values (left column) complemented by the median, the 25th and 75th percentiles, and outliers (right column) for an excerpt from “Lisztomania” by Phoenix 48
- Figure 5 *SNRF* and auditory bandwidth values (left column) complemented by the median, the 25th and 75th percentiles, and outliers (right column) for an excerpt from “The Terrorist” by DJ Vadim 49
- Figure 6 *SNRF* and auditory bandwidth values (left column) complemented by the median, the 25th and 75th percentiles, and outliers (right column) for an excerpt from “Lisztomania” by Phoenix 50
- Figure 7 Mean opinion scores and 95-% confidence intervals for the two music excerpts (left column) and the overall grades for the algorithms under test (right column) 51
- Figure 8 *SDR* and *PSM* values for the oracle *MMSE* spatial filter (upper row) and the average rate-performance difference curves for three different filtering and/or coding strategies (lower row) 52
- Figure 9 Mean *SDR*, *ISR*, *SIR*, and *SAR* values for the *SiSEC* 2013 development dataset consisting of three music excerpts 57
- Figure 10 Mean *OPS*, *TPS*, *IPS*, and *APS* values for the *SiSEC* 2013 development dataset consisting of three music excerpts 58
- Figure 11 *ODG* as a function of the side-information rate 62
- Figure 12 Decoder runtime as a function of the number of bands at a side-information rate of 30 kbps 63

|           |   |    |
|-----------|---|----|
| Figure 13 | The medians and the 25th and 75th percentiles for each system under test  | 66 |
| Figure 14 | Basic broadband compressor model  | 68 |
| Figure 15 | Graphical illustration of the iterative search for the root   | 77 |
| Figure 16 | An illustrative example using an RMS amplitude detector with $\tau_v$ set to 5 ms, a threshold of $-20$ dBFS (dashed line in the upper right corner), a compression ratio of 4 : 1, and $\tau_g$ set to 1.6 ms for attack and 17 ms for release, respectively. The RMSE is ca. $-129$ dBFS.   | 84 |
| Figure 17 | RMSE as a function of typical attack and release times using a peak (upper row) or an RMS amplitude detector (lower row). In the left column, the attack time of the envelope filter is varied while the release time is held constant. The right column shows the reverse case. The time constants of the gain filter are fixed at zero. In all four cases, threshold and ratio are fixed at $-32$ dBFS and 4 : 1, respectively. | 85 |
| Figure 18 | RMSE as a function of typical attack and release times using a peak (upper row) or an RMS amplitude detector (lower row). In the left column, the attack time of the gain filter is varied while the release time is held constant. The right column shows the reverse case. The time constants of the envelope filter are fixed at zero. In all four cases, threshold and ratio are fixed at $-32$ dBFS and 4 : 1, respectively. | 86 |
| Figure 19 | RMSE as a function of threshold relative to the signal's average loudness level (left column) and compression ratio (right column) using a peak (upper row) or an RMS amplitude detector (lower row). The time constants are: $\tau_v = 5$ ms, $\tau_g^{\text{att}} = 20$ ms, and $\tau_g^{\text{rel}} = 1$ s.  | 87 |
| Figure 20 | A two-stage cascade connection  | 90 |
| Figure 21 | RMSE and PSM values for the multitrack (upper row) and the corresponding difference values between the estimates from the de-/compressed and the uncompressed mixture signal (lower row). The asterisk (*) indicates $M = 0$ (no gain).   | 95 |
| Figure 22 | The proposed coding scheme. The asterisk (*) indicates that the coding/decoding blocks may also include watermarking/signing functionality.   | 99 |

Figure 23 Mean **SIXFP**, **SIRFP**, and **ODG** values for the multi-track and the corresponding data-rate savings. Starting from the middle, we see that for imperceptible quality impairment, i.e. an **ODG** above  $-1$ , the **SIRFP** must be 48 or greater (shaded area). Switching over to the left, we see that at least 7 channels are necessary to reach it. The figure on the right indicates that 30 % of **LPCM** data can so be saved. 106

## LIST OF TABLES

---

|          |  |  |
|----------|--|--|
| Table 1  | Studio effects divided into categories and classified as <b>LTI</b> (*), <b>LTV</b> (†), <b>NTI</b> (‡), and <b>NTV</b> (§). Effects considered in this work are in boldface. 10   |  |
| Table 2  | Panning used for the two music pieces 42   |  |
| Table 3  | Estimated image parameters for the development set 45  |  |
| Table 4  | Experimental results. The <b>ISR</b> , <b>SAR</b> , and <b>TPS</b> values for mono sources are framed. 54  |  |
| Table 5  | A comparison between the results shown in Table 4 and the scores reported in <b>SiSEC</b> 2011 for two oracle systems. The upper row next to the track name represents the <b>STFT</b> -based system and the lower row represents the cochleagram-based system, respectively. 56 |  |
| Table 6  | The corpus of prearranged mixes 65   |  |
| Table 7  | Selected compressor settings 81  |  |
| Table 8  | Performance figures obtained for real-world audio material (12 items in total) 83  |  |
| Table 9  | Runtime complexity of the cascade connection as a function of the number of sources, the number of frequency bands, and the frame length. It is assumed that the transform length is equal to the frame length. 92   |  |
| Table 10 | Information rate of <b>STPSD</b> values (left) and <b>STPSD</b> difference values using <b>DPCM</b> (right) if quantized with 6 bits at 44.1-kHz sample rate and 16-kHz cutoff 93  |  |
| Table 11 | Compressor setting used for the complete mix 93  |  |

## LIST OF ALGORITHMS

---

|             |   |     |
|-------------|---|-----|
| Algorithm 1 | Single-input single-output compressor       | 79  |
| Algorithm 2 | Single-input single-output decompressor     | 80  |
| Algorithm 3 | Root-finding algorithm                      | 81  |
| Algorithm 4 | Equal-SIR <sup>out</sup> power distribution | 102 |

## ACRONYMS

---

|       |  |
|-------|--|
| AAC   | Advanced Audio Coding                                |
| APS   | artifacts-related perceptual score                   |
| BGO   | background object (MPEG SAOC)                        |
| BSS   | blind source separation                              |
| CD    | Compact Disc   |
| CDDA  | Compact Disc Digital Audio                           |
| CIPIC | Center for Image Processing and Integrated Computing |
| DAW   | digital audio workstation                            |
| DFT   | discrete Fourier transform                           |
| DPCM  | differential pulse-code modulation                   |
| DRC   | dynamic range compression                            |
| EAO   | enhanced audio object (MPEG SAOC)                    |
| EAOS  | enhanced audio object separation                     |
| ERB   | equivalent rectangular bandwidth                     |
| FAAC  | Freeware Advanced Audio Coder                        |
| FFT   | fast Fourier transform                               |
| FGO   | foreground object (MPEG SAOC)                        |
| FIR   | finite impulse response                              |
| FLAC  | Free Lossless Audio Codec                            |
| HRIR  | head-related impulse response                        |
| HRTF  | head-related transfer function                       |

|        |   |
|--------|---|
| IIR    | infinite impulse response                                 |
| IOC    | inter-object cross coherence (MPEG SAOC)                  |
| IPS    | interference-related perceptual score                     |
| ISR    | image-to-spatial distortion ratio                         |
| ISS    | informed source separation                                |
| JND    | just-noticeable difference                                |
| JPEG   | Joint Photographic Experts Group                          |
| KBD    | Kaiser–Bessel derived                                     |
| LCMV   | linearly constrained minimum-variance (filtering, e.g.)   |
| LPCM   | linear pulse-code modulation                              |
| LTI    | linear time-invariant                                     |
| LTV    | linear time-variant                                       |
| LS     | least squares   |
| MMSE   | minimum mean square error                                 |
| MOS    | mean opinion score  |
| MP3    | MPEG-1 or MPEG-2 Audio Layer III                          |
| MPEG   | Motion Picture Experts Group                              |
| MPS    | MPEG surround   |
| MSE    | mean squared error  |
| MUSHRA | MUlti-Stimulus test with Hidden Reference and Anchor      |
| MVDR   | minimum-variance distortionless response                  |
| NMF    | non-negative matrix factorization                         |
| NTI    | nonlinear time-invariant                                  |
| NTV    | nonlinear time-variant                                    |
| ODG    | objective difference grade                                |
| OPS    | overall perceptual score                                  |
| PCMV   | power-conserving minimum-variance (filtering, e.g.)       |
| PEAQ   | Perceptual Evaluation of Audio Quality                    |
| PEASS  | Perceptual Evaluation methods for Audio Source Separation |
| PSM    | perceptual similarity metric                              |
| RMS    | root mean square  |
| RMSE   | root-mean-square error                                    |
| SAC    | spatial audio coding                                      |
| SAOC   | spatial audio object coding                               |
| SAR    | source-to-artifacts ratio                                 |
| SDR    | signal-to-distortion ratio                                |



|       |   |
|-------|---|
| SIR   | signal-to-interference ratio                              |
| SIRFP | frequency and power weighted signal-to-interference ratio |
| SiSEC | Signal Separation Evaluation Campaign                     |
| SIXFP | frequency and power weighted similarity index             |
| SISO  | single-input single-output                                |
| SIX   | similarity index  |
| SIXFP | frequency and power weighted similarity index             |
| SNR   | signal-to-noise ratio                                     |
| SNRF  | frequency-weighted signal-to-noise ratio                  |
| STCSD | short-time cross-spectral density                         |
| STFT  | short-time Fourier transform                              |
| STPSD | short-time power spectral density                         |
| TF    | time-frequency (point, region, plane, etc.)               |
| TFR   | time-frequency representation                             |
| TPS   | target-related perceptual score                           |
| TTT   | two-to-three ( <a href="#">MPEG SAOC</a> )                |
| USSR  | Underdetermined Source Signal Recovery                    |



Part I

INTRODUCTION



## INTRODUCTION

---

### 1.1 MOTIVATION

Most, if not all, of professionally produced music recordings have undergone two processes, mixing and mastering, and many common distribution formats, such as the Compact Disc Digital Audio (CDDA), are strictly stereo. While the term “mastering” refers to the process of optimizing the final mix and transferring it to a data storage device, the term “mixing” refers to the process of putting multiple layers of prerecorded and edited audio together to a composite mixture. Each layer can be imagined as a track in an audio mixer, which then again stands for a sound source. A source can be a vocal, an instrument, or a group of the aforesaid. The apparent placement of sources between the speakers in a stereo sound field is also known as “imaging” [1] in professional audio engineering.

Audio engineering is an established discipline employed in many areas that are part of our everyday life without us taking notice of it. But not many know how the audio was produced. If we take sound recording and reproduction or broadcasting as an example, we may imagine that a prerecorded signal from an acoustic source is altered by an audio engineer in such a manner that it corresponds to certain criteria when played back. The number of these criteria can be large and usually depends on the context. In general, the said alteration of the input signal is a sequence of numerous forward transformations, the reversibility of which is of little or no interest. But what if one wished to do exactly this, that is to reverse the transformation chain, and what is more, in a systematic and repeatable manner? It would allow the listener to *interact with* or *reinterpret* the music stored, e.g., on a Compact Disc (CD) and, hence, to become *active*. To render this possible, a music recording must be decomposed into its constituent elements—at least in part. . .

### 1.2 OBJECTIVES

#### 1.2.1 *Active Listening*

Active listening is a concept in music technology developed in the late '90s by Pachet and Delerue [2]. It aims at giving the user some degree of control of the music she or he listens to. As opposed to the usual case where a piece of music is simply played back without any conceptual modifications, active listening allows for a personalized

music experience through reinterpretation. Its objective is to provide the listener with a higher level of musical comfort and to give access to new music by creating listening environments for existing music repertoires, in which the modifications preserve the semantics. One such example is the respatialization of sound sources or remixing by changing the volume and/or the panoramic angle of a distinct source. The idea of active listening is seized in works like [3, 4, 5, 6], e.g. So, to enable active listening, one must reacquire access to latent source components given the mastered mix, i.e. one must *reverse engineer* the mix.

### 1.2.2 Reverse Audio Engineering

The objective of reverse audio engineering can be either to identify the transformation parameters given the input and output signals as in [7] or to recover the input signal that belongs to the output signal given the transformation parameters, or both. An explicit signal and system model is mandatory in either case. The second case may look trivial, but only if the transformation is linear and orthogonal and as such perfectly invertible. Yet, the forward transform is often neither linear nor invertible. This is the case for dynamic range compression (DRC), e.g., which is commonly described by a *dynamic nonlinear time-variant* system. The classical linear time-invariant (LTI) system theory does not apply here, so a tailored solution must be found instead. At this point, I would also like to highlight the fact that neither Volterra nor Wiener model approaches [8, 9, 10] offer a solution and neither do describing functions [11, 12]. These are powerful tools for identifying a *time-invariant* or a *slowly varying* nonlinear system, or for analyzing a feedback system with a static nonlinearity in regard to its *limit cycle* behavior. Furthermore, for the reason that the number of used tracks is usually greater than the number of mixture channels, mixing is in a mathematical sense *underdetermined*. Thus, demixing constitutes an *ill-posed* source separation problem.

## 1.3 ORGANIZATION

The remaining part of this work is organized as follows. Chapter 2 gives a brief overview of digital audio engineering including mixing and mastering. A system-theoretical classification of most commonly used studio effects and sound processors is also presented. Chapter 3 outlines the issues of inverse problems and explains why demixing is ill posed in the general case. In Chapter 4 it is shown how such an inverse problem can be addressed by means of Bayesian inference or statistical filtering. The notion of post-nonlinear mixtures is brought up as a special case of mixing and mastering. Chapter 5 is dedicated to the pursued “informed” approach. The major reverse engineering

problem is defined and a feasible solution is offered. The following Chapter 7 deals with audio source separation, or demixing, which is equivalent to the inversion of mixing. There, a new spatial filter is derived. Its improved perceptual performance is confirmed in various experiments. An efficient coding strategy for the metadata is also presented. Apart from the separation of single-channel or point sources, it is shown how two-channel sources and/or their spatial images can be isolated from the mixture. It is argued that the respective source covariance matrix should be diagonal and it is shown why. A mixing system that incorporates a timing difference between the two channels for a more natural stereo effect is discussed as well. In Chapter 8, the proposed source separation algorithm is contrasted with MPEG's technology for interactive remixing. The performance of the former is assessed in a simulated active listening scenario. The inversion of mastering in the form of DRC is the central point of Chapter 9. Having defined the model of a compressor, it is demonstrated how the latter can be inverted using a novel approach. An exemplary compressor and the corresponding decompressor are given in terms of pseudocode. It is explained how the decompressor applies to a stereo mixture. The demixing and the decompression stages are combined into a cascade connection in the following Chapter 10. The cascade connection is analyzed with respect to some important performance characteristics like delay, complexity, and side-information rate and evaluated on an exemplary multitrack recording. This ends the main part of the thesis. In an extra Chapter 11, a proof of concept for a new multichannel object-based coding is presented which is a spillover from the previous chapters. With this coding scheme it is possible to control the sound quality at the decoder from the encoder in a measurable way. Chapter 12 concludes the work with a discussion and mentions possible directions for future work.

To sum up, the contribution of the thesis consists in:

1. The classification of digital audio effects
2. A statistical time-frequency model for single-channel and two-channel sound sources and their spatial images
3. The development of a constrained spatial filtering approach for sound source separation and narrowband deconvolution
4. The derivation of a new spatial filter
5. The inclusion of human perception to minimize the quantity of side information
6. The elaboration of a novel and unprecedented approach for the inversion of a dynamic nonlinear time-variant operator, such as the dynamic range compressor
7. A two-stage cascade to reverse mastering and mixing
8. A novel multichannel object-based audio coding scheme

9. The proposition of new perceptually motivated metrics for the objective assessment of sound quality



Part II

PRELIMINARIES



## 2.1 AUDIO EFFECTS AND AUDIO SIGNAL PROCESSORS

Digital audio effects and audio signal processors manipulate sound characteristics in three basic dimensions: volume, frequency and time. They are used to alter an audio signal's apparent volume, harmonic structure—or its waveform in general, so as to create sound patterns that are widely accepted and to satisfy personal taste of the composer or the engineer alike. The most common studio effects fall into one of the following categories: distortion effects which are based on gain, dynamics processors that control the apparent volume by altering the signal amplitude, filter effects which alter the frequency content, modulation effects which alter the amplitude, frequency, delay, or phase of the so-called “wet” signal over time, pitch and frequency effects that alter the pitch or create harmony, time-based effects that delay the so-called “dry” signal or add echoes, and finally spatial effects which alter the apparent direction of sound sources in a stereophonic sound field and give them the notion of spatial extent.

A list of common studio effects is given in Table 1. These are further classified based on their system-theoretical properties and grouped by the superscripts \*, †, ‡, and §, which have the following meaning:

\* Linear time-invariant (**LTI**)

**LTI** systems are completely specified by their transfer function.

† Linear time-variant (**LTV**)

**LTV** systems can be described by the Bello functions [13]. Effects that fall into this category are barely used for music production.

**LTI** effects can be regarded as a subclass of **LTV** effects.

‡ Nonlinear time-invariant (**NTI**)

**NTI** systems are usually modeled with Volterra series [14]. Most **NTI** audio effects are non-invertible.

§ Nonlinear time-variant (**NTV**)

**NTV** systems require in general a special treatment.

In the following, effects are considered as deterministic and casual. Effects with memory are lumped, i.e. their number of state variables, or taps, is finite. The focus is laid on three types of effects including panning and balance control, equalization, and compression during mastering. They are printed in boldface in Table 1. It should yet be noted that “reverb” is the most-used effect in the studio, as we are used to hearing sounds in enclosed spaces. However, on account of its complexity, dereverberation is left out of this work for a separate study. For more details on digital audio effects refer to [1, 15], e.g.

| EFFECT                                     | L | TI | MEMORY | I/O                        |
|--|---|----|--------|----------------------------|
| Distortion effects                         |   |    |        |                            |
| Overdrive/fuzz <sup>†</sup>                |   | ✓  |        | SISO <sup>1</sup>          |
| Dynamics processors                        |   |    |        |                            |
| Fader/amplifier*                           | ✓ | ✓  |        | SISO                       |
| <b>Compressor/<br/>limiter<sup>§</sup></b> |   |    | ✓      | SISO/<br>MISO <sup>2</sup> |
| Noise gate <sup>†</sup>                    |   | ✓  |        | SISO                       |
| Filter effects                             |   |    |        |                            |
| <b>Equalizer*</b>                          | ✓ | ✓  | ✓      | SISO                       |
| Roll-off*                                  | ✓ | ✓  | ✓      | SISO                       |
| Modulation effects                         |   |    |        |                            |
| Chorus <sup>‡</sup>                        | ✓ |    | ✓      | MISO                       |
| Flanger <sup>‡</sup>                       | ✓ |    | ✓      | MISO                       |
| Phaser <sup>‡</sup>                        | ✓ |    |        | MISO                       |
| Ring modulator <sup>‡</sup>                | ✓ |    |        | MISO                       |
| Tremolo <sup>‡</sup>                       | ✓ |    |        | MISO                       |
| Vibrato <sup>‡</sup>                       | ✓ |    |        | MISO                       |
| Pitch and frequency effects                |   |    |        |                            |
| Pitch shifter*                             | ✓ | ✓  |        | SISO                       |
| Harmonizer*                                | ✓ | ✓  |        | MISO                       |
| Time-based effects                         |   |    |        |                            |
| Delay*                                     | ✓ | ✓  | ✓      | SISO                       |
| Reverb*                                    | ✓ | ✓  | ✓      | SISO                       |
| Spatial effects                            |   |    |        |                            |
| <b>Panning*</b>                            | ✓ | ✓  |        | SIMO <sup>3</sup>          |
| Fattening*                                 | ✓ | ✓  | ✓      | SIMO                       |

Table 1: Studio effects divided into categories and classified as **LTI** (\*), **LTV** (†), **NTI** (‡), and **NTV** (§). Effects considered in this work are in boldface.

1. Single-input single-output
2. Multiple-input single-output
3. Single-input multiple-output

## 2.2 MIXING

The mixing stage is formulated using the following notation:

$$x_l(n) = \sum_{i=1}^I A_{li}[s_{li}(n), \theta_{li}(n)] \quad \text{with } l \in \{1, 2\}, \quad (1)$$

where  $s_{li}(n)$  is the  $i$ th console input routed to the  $l$ th channel,  $\theta_{li}(n)$  are the time-varying mixing parameters belonging to the composite effects operator  $A_{li}$  applied to  $s_{li}(n)$ , and  $x_l(n)$  is the output from the  $l$ th mixture channel. The mixture is thus a superposition of  $I$  distinct audio tracks. The input signals can be single-channel or two-channel, while the composite operator can be a cascade of effects. Effects can be linear or nonlinear, time-invariant or time-variant, instantaneous (memoryless) or dynamic (with memory), and may have a different number of inputs and outputs, see Table 1. Since  $I$  is usually greater than two, the mixing process is also referred to as “fold-down” or as “downmix”.

## 2.3 MASTERING

On the analogy of (1), the mastering stage is formulated as:

$$y_l(n) = A_l[x_l(n), \theta_l(n), \dots], \quad (2)$$

where  $\theta_l(n)$  now represents the time-varying mastering parameters belonging to the composite effects operator  $A_l$  applied to  $x_l(n)$ . The composite operator can also have any other mixture channel or an external sound source as input parameter as indicated by “...”. This allows to describe more sophisticated effects like the compressor. As a consequence,  $y_l(n)$  denotes the mastered output.



## ILL-POSED INVERSE PROBLEMS

---

### 3.1 INVERSE PROBLEMS

Inverse problems arise in situations where the model parameters of a system need to be inferred from measurements, and this with or without prior knowledge of its internal structure. They represent the counterpart to the ordinary forward problems, which can be solved using signal and system theory. Whereas forward problems have a unique solution, inverse problems may have many or no solution at all.

Many inverse problems are formulated as optimization problems. The sought-after model parameters are usually those that best fit the measurements or observations under an optimization criterion. As a rule, a hypothesis on the distribution of the error is made. When the latter is Gaussian, i.e.  $f(x) \sim \exp(-x^2)$ , the sum of the squared errors or misfits is minimized to gain the best solution in the least-squares sense. The reason why the method of least squares is so popular lies in its simplicity. This is also why many problems in signal processing are considered to be of Gaussian nature. Beyond, the method of least squares allows for a closed-form solution.

When dealing with inverse problems, two issues are encountered. The first is to find a model of the system which is consistent with the observed data. The second is to quantify the non-uniqueness of the solution.

### 3.2 ILL-POSEDNESS

As stated earlier, the system of equations resulting from mixing is generally underdetermined. And even though a forward operator in the form of a linear mixing system can be deduced from the design of a typical digital audio workstation (DAW) almost surely, the source separation problem is non-unique and so “ill posed” in Hadamard’s terminology. There are, theoretically, infinitely many different source components, or model parameters to be more accurate, that could fit the observed mixture. Mastering poses additional problems. Ill-posed inverse problems often require to be reformulated, so that they can be treated numerically. This involves making additional assumptions in regard to the solution, such as its smoothness. This is also known as “regularization”. Regularized inverse problems can further be viewed as special cases of Bayesian inference. For further reading, see [16].





## 4.1 BAYESIAN INFERENCE

In Bayesian statistics the evidence about the true state of the world is expressed in terms of degrees of belief or Bayesian probabilities. In consequence, Bayesian inference is a method that uses Bayes' rule to update the probability estimate for a hypothesis as more evidence is acquired about the world. It so provides the posterior probability as a consequence of two antecedents, a prior probability and a function derived from a probability model for the observations which is also called the "likelihood". The former is computed as

$$P(H | E) = \frac{P(E | H)P(H)}{P(E)}, \quad (3)$$

which is the Bayes rule. In (3),  $H$  is the hypothesis, the probability of which is affected by the observations or evidence  $E$ .  $P(H)$  is thus the prior probability, i.e. the probability before  $E$  is observed.  $P(E | H)$  is the probability of observing  $E$  given  $H$ —the likelihood, while  $P(E)$  is termed the marginal likelihood or also model evidence. The updated prior probability  $P(H | E)$  is equal to the posterior probability, i.e. the probability of  $H$  after  $E$  is observed. This tells us the probability of a hypothesis given the observed evidence. Put into words, posterior is proportional to prior times likelihood.

The application of Bayesian inference to inverse problems may be illustrated with the following example. Let  $s(n)$  denote an unknown input signal corrupted by an interferer  $r(n)$  and  $x(n)$  the observable signal,  $x(n) = s(n) + r(n)$ . Let  $S$  and  $R$  be two independent normally distributed random variables with zero mean, and  $s(n)$  and  $r(n)$  the respective realizations. Let  $s(n)$  span the one-dimensional parameter space. The prior distribution over  $s(n)$  be  $p(s | \theta) = f_S(s; 0, \sigma_S^2)$ , where  $\theta = \{0, \sigma_S^2\}$  is the set of hyperparameters—the parameters belonging to the prior. The probability density function of the interferer is also the likelihood to observe  $x(n)$  given  $s(n)$ ,  $p(x | s, \theta) \equiv f_R(r; 0, \sigma_R^2)$ . As the prior and the likelihood are both normal, the posterior is normal and the Bayes estimator for the input signal is

$$\hat{s}(n) = \frac{\sigma_S^2}{\sigma_R^2 + \sigma_S^2} x(n), \quad (4)$$

which is the mean of the posterior distribution  $p(s | x, \theta)$  with  $s | x \sim \mathcal{N}[\sigma_S^2/(\sigma_S^2 + \sigma_R^2)x, \sigma_S^2\sigma_R^2/(\sigma_S^2 + \sigma_R^2)] \propto p(x | s, \theta)p(s | \theta)$ . It should further be noted that there exist other methods of Bayesian estimation to select

elements of central tendency from the posterior distribution. The one presented above is the (linear) minimum mean square error (MMSE) estimator. Taking a closer look at (4), one may recognize the transfer function of the classical Wiener filter. The latter is an MMSE estimator in the frequency domain. Further details on the Bayesian approach to inverse problems can be found in [17].

#### 4.2 LINEAR OPTIMUM FILTERING

The problem of linear optimum filtering consists in designing any type of linear discrete-time filter whose output provides an estimate of a desired response given a set of input samples. The optimization criterion is usually chosen as the mean squared error (MSE) between the desired response and the actual response, which is the Bayes risk or the cost function to be minimized. The choice of the MSE criterion leads to a second-order dependence of the cost function on the filter coefficients. The cost function has so a global minimum that defines the optimum operating point. Linear optimum filters that minimize the MSE of the estimation are commonly known as Wiener filters. In solving the minimization problem, there are no constraints imposed on the solution.

To find the optimum filter coefficients, the Wiener–Hopf equations must be solved. They model the relation between the autocorrelation of the input samples and the cross-correlation between the latter and the desired response. The Wiener–Hopf equations are much simpler to solve for transversal i.e. finite impulse response (FIR) filters. Their solution in the compact matrix form is given by [18]

$$\mathbf{w}_o = \mathbf{R}^{-1} \mathbf{p}, \quad (5)$$

where  $\mathbf{p}$  is the cross-correlation vector and  $\mathbf{R}$  is, correspondingly, the autocorrelation matrix which is assumed nonsingular. The optimum filter weight are represented by the vector  $\mathbf{w}_o$ . It can be seen that the computation of the optimum filter requires knowledge of  $\mathbf{R}$  and  $\mathbf{p}$ .

The optimization problem described above is typically of temporal nature. Its spatial version can be found in beamforming applications. There, e.g., a linear array of uniformly spaced antenna is illuminated by an isotropic source located in the far field. At a given time instant  $n$ , a plane wave impinges on the array along a direction specified by the angle  $\alpha$  with respect to the perpendicular to the array. Then, the beamforming problem would be, e.g., to obtain the source signal with a high signal-to-noise ratio (SNR) by taking a model of the wave field into account. The formulation of an optimum beamformer is subject to the same optimization problem. This fact will serve us later. So, at long last, let us recall two typical beamforming quantities:

1. The azimuth-dependent *beam response* or *gain* is defined as

$$g(\alpha) \triangleq \mathbf{w}^H \mathbf{a}(\alpha), \quad (6)$$

where  $\mathbf{a}(\alpha)$  is also called the “steering” vector. Superscript H is the Hermitian or conjugate transpose operator, as the filter can be complex. The *beam pattern* is the magnitude-squared gain.

2. The *spatial power spectrum*

$$P(\alpha) \triangleq \mathbf{w}^H(\alpha) \mathbf{R} \mathbf{w}(\alpha) \quad (7)$$

is a measure for the mean total signal power received from the direction  $\alpha$ .

In beamforming applications, the steering vector  $\mathbf{a}$  depends on the underlying array geometry. Here in this work, however, it is detached from a physical sound propagation model and shall only define how the sound power is distributed across channels. The formal definition of  $\mathbf{R}$ ,  $\mathbf{p}$ , and  $\mathbf{a}$  in a particular scenario will be given in Chapter 7.

#### 4.3 POST-NONLINEAR MIXTURES

Post-nonlinear mixtures are special cases of nonlinear mixtures. In [19] a proof is provided which shows that it is not possible to handle a nonlinear mixture without nonlinear distortion. On the other hand, post-nonlinear mixtures, which in the cited work are linear mixtures followed by a nonlinear distortion function without memory, can be treated with the same indeterminacies as if these were purely linear. They model systems in which the transmission channel is linear, but the sensor array is not, and hence, it represents a source of nonlinear distortion. The two conditions to be fulfilled are that the sources are mutually independent and that the nonlinearities are both invertible and differentiable functions. The approach requires knowledge of the so-called “score” functions that represent the type of distortion. The score functions are estimated from the nonlinear mixture as they are unknown in the general case. The conclusion is that the separation is impossible without additional prior knowledge of the model, as the independence assumption is not sufficient in the nonlinear case. This result is of particular importance, because music recordings fall into that category. Further details on post-nonlinear mixtures are given in [20] and the references found therein.

As a side note, audio source separation techniques so far consider exclusively linear mixing. Recently, efforts have been made to deduce a mixing model that takes the complete music production chain into account [21]. There, it is argued in favor of a linear model that unifies *linear* effects such as reverberation with *nonlinear* processing such as dynamic range compression. However, the motivation for the model is to undo the mixing taking mastering into account but not to undo the mastering as such.



## INFORMED APPROACH

---

### 5.1 BASIC IDEA

In his 2005 tutorial [22], Knuth takes the Bayesian approach for the design of robust source separation algorithms that take advantage of *prior knowledge* of the problem at hand to assure that one reaches an optimal solution. He calls it *informed* source separation to distinguish the approach from *blind* source separation where little is known. The basic idea of his informed approach consists in introducing the *prior information* about the problem, which can be a physical property and physical law alike, into the solution. This is surely more specific than Bayes' priors which in the proper sense are distributions. The model that accompanies such an informed source separation problem is so a combination of prior distributions *and* prior information. It should be noted that there are a number of so-called score-informed source separation techniques which also use this paradigm [23, 24, 25]. Most commonly, one of many non-negative matrix factorizations (NMFs) is made use of to approximate the spectrograms of the original signals, so as to apply weighted binary masks or linear Wiener filters to the mixture spectrum. The binary masks can also be interpreted as sums of delta distributions in a probabilistic sense.

### 5.2 CONCRETIZATION

Knuth's idea can be applied to the reverse engineering problem in the following manner. First of all, the mixture shall be considered as post-nonlinear, assuming that linear mixing is followed by nonlinear mastering. So, the mixing and the mastering can be described by two deterministic systems in cascade. The probabilistic component in the model shall be due to the ill-posedness of the task. The probability of hearing the source of interest in the mixture is given by the posterior distribution of the respective source signal. The knowledge of how a music recording is produced represents the prior knowledge that we have. This is typically the effects used and the spatial positioning of sources. The prior information comprises the system parameters, i.e. the filter coefficients and the panoramic angles or directions, and the parameters of the source distributions, i.e. the signal parameters. The latter are the hyperparameters according to Bayes.

### 5.2.1 *Problem Formulation*

From the previous section it is clear that the best possible estimate of the source signals is achievable only with prior information in the form of system and signal parameters being perfectly given. On that account, the problem to solve is stated as follows. Given the mixing, mastering, and source models, recover the source signal estimates in best possible quality from the mixture. The signals' parameters shall be estimated from the accessible original source signals. The mixing and mastering parameters shall be deemed as known.

### 5.2.2 *Proposed Solution*

The problem as it stated above makes sense only if the recovery of the source components is carried out without access to the originals. Otherwise the problem is trivial. The discussed approach is hence to be applied in a scenario, where the "collection" of prior information is uncoupled from its employment. Once the sources are mixed, they are no longer available, i.e. One could also characterize the approach by the *temporally and locally bounded access to the source signals*. So, one needs to differentiate between the process of content *creation* and the process of content *consumption*. The content creator is responsible for providing all the necessary data to the content consumer, so that the latter is enabled to unnoticeably reverse engineer the music mix and to remix the content ad libitum with the help of a graphical interface. This is realized, e.g., in an encoder/decoder framework. The task of the encoder is to extract a minimum of ancillary data from the source signals, so the decoder can recover their replica from the mixture in high—preferably perceptual—sound quality. This metadata, if small enough, can either be hidden in the mixture signal itself in the form of an inaudible watermark or must be attached to the mixture.

### 5.2.3 *Associated Problem*

Aside from the obvious reverse engineering problem, there is also the problem of data reduction with respect to the model parameters, which is not negligible. The problem is even more challenging when the metadata is to be hidden in the mixture or when the decoder is to perform in real time. It is also worth questioning what a sufficient data rate is for a good system performance.

Part III

REVERSE AUDIO ENGINEERING





PREVIOUS WORKS

---

About a decade ago, in 2003, Avendano has presented a scheme [26] similar to [27] with which one can identify, separate, and manipulate individual sound sources in a studio music recording. His scheme uses a “panning index” to identify collinear source components and clusters those into coherent time-frequency (TF) regions [28, 29]. These regions can then be manipulated by applying a “mask” that alters the magnitude of the signal components in question. In that manner one can either attenuate or accentuate the source of interest—the vocal or an instrument—and change its apparent location. These features, otherwise known as *karaoke*, *mix-minus*, and *repanning*, are all basic elements of *active listening*. Avendano’s scheme, which is applicable to convolutional stereo mixtures without any restrictions with regard to the number of superposed sound sources, has one drawback: the resulting sound quality is insufficient for professional applications. A similar technique for *binaural* recordings has been developed by Mouba and Marchand in 2006 [30].

In order to attain a higher quality as compared to Avendano, Oh *et al.* presented in [31] a *model-based* remixing scheme that likewise allows for gain manipulations and repanning of distinct sound sources, but with the aid of *additional information*. The latter consists of the mixing parameters and the approximate short-time power spectral densities (STPSDs) of the sources that are to be manipulated. This information, which is transmitted alongside the stereo mixture signal, is used to best fit the remixing model in the least squares (LS) sense given new user-definable mixing parameters. The authors claim their technique to require less side information than other comparable schemes such as MPEG’s spatial audio object coding (SAOC) [32, 33, 34] to achieve the same effect, as only the STPSDs of a few selected sound sources and their mixing coefficients need to be communicated to the remixing unit. However, if the user was intended to be given the possibility to alter the entire mix, the required amount of side information would coincide with the one of SAOC.

SAOC is an object-oriented extension to spatial audio coding (SAC) [35] which combines efficient coding of audio objects and interactive rendering during playback. In MPEG’s terminology, an audio object is synonymous with a sound source. SAOC so comprises two integrated parts: MPEG surround (MPS) [36] as the base technology and an object transcoder as superstructure. Using SAOC, multiple audio signals are transmitted in the form of a single-channel or a two-channel mix, transcoded to the MPS format, and rerendered; all that with the help

of object metadata and rendering information. Additionally, the mix is perceptual-entropy, i.e. “lossy”, coded. When operated in karaoke, or solo, mode, however, the SAOC transcoder engages in addition an *object decoder*. The purpose of the latter is to decompose the mixture into objects that can be manipulated individually and with a higher precision before rerendered to MPS. The additional prior information that is necessary to carry out the decomposition includes the STPSDs and the mixing parameters, including panning and gain. The correct MPEG terms hereof are “object level differences”, “downmix channel level differences”, and “downmix gains”. These quantities are hence provided by the *object encoder*.

As of today, SAOC is maybe the most versatile object-based coding scheme in comparison with the aforementioned schemes that allows for interactive remixing. This is certainly one of the many reasons it has become international standard in 2010. Even so, although mono and multichannel objects can be handled in SAOC, the mixing model is very basic and thus not representative for professionally produced music recordings. Nonlinear processing such as, e.g., dynamic range compression is not taken into account and so it is not inverted.

A method to invert dynamics compression is described in [37], but it requires a gain value to for each sample of the compressed signal, which is transmitted. To provide a means to control the data rate, the gain signal is subsampled and entropy coded. This approach is very generic but inefficient, as it does not rely on a compressor model.

On the other hand, transmitting the uncompressed signal together with a few typical compression parameters like *threshold*, *ratio*, *attack*, and *release* would require a much smaller capacity and yield the best possible signal quality with regard to any thinkable measure. A more realistic scenario is when the uncompressed signal is not available on the consumer side. This is the usual case for studio music recordings where the listener is offered a signal which is meant to sound “good” to everyone.

The Dolby solution for broadcast loudness issues, e.g., consists of the transmission of metadata that can be used to normalize loudness across programs and channels [38]. The metadata, that helps control the program’s dynamic range, is optimized on the broadcaster side and transmitted alongside the broadcast signal. This is a convenient solution for broadcasters, not least because the metadata is compact. Dynamic range adjustment is, yet, another forward transform rather than a true inversion. Evidently, none of the existing solutions satisfy the reverse engineering objective of this work.

In recent years several methods have been proposed that address audio source separation in an “informed” scenario [39, 40, 41]. The reason for this trend is the plain fact that after decades of research “blind” or rather “semi-blind” source separation approaches to this day yield unsatisfactory quality with respect to what is considered

as professional audio applications—for which quality is key; for an overview and general concepts of blind approaches see [42]. Blind *speech* separation techniques, on the other hand, often rely on a speech production model and/or make specific assumptions, which cannot be generally upheld for music [43]. But what is worse is that many sophisticated techniques are not applicable if the separation problem is *ill-posed*, which is when the number of observations (channels) is smaller than the number of unknowns (sources). Illposedness is yet the normal case for most music recordings, since the content is still distributed and consumed primarily in two-channel stereo format. The concept of informed source separation (ISS), hence, may be seen as a way of overcoming the limitations of blind source separation (BSS) found in today’s state-of-the-art algorithms.

As an example, the idea of informing a separator with the mixing parameters and the STPSDs, as in Oh *et al.*’s scheme or SAOC, can also be found in [40]. There, however, this additional information is used to calculate a *generalized* Wiener filter for each source component in each channel separately. This type of MSE based interference reduction takes account of the power relations between the source signals but not their spatial diversity, and neither does it invert the mixing system. . .



## 7.1 INTENSITY STEREOPHONY

Let a stereo system be considered in which one or multiple mono signals are unevenly distributed over two independent channels, in such a way that an illusion of directionality and audible perspective is created. This is achieved by varying the amplitude of the signal sent to each channel relative to the listener. The parameters that control this relative amplitude are *direction* and *volume*, see Fig. 1. They are equivalent to the position of the panoramic potentiometer, the pan-pot, and the fader position on a mixing desk in a recording studio and are applied to each mono signal separately. The summation of all panned and volume adjusted mono signals constitutes the sound field of the mixture. This type of “artificial” stereo can be viewed as the counterpart to X–Y recording using two bi-directional microphones perpendicular to each other, and so, forming a Blumlein Pair. Albeit simple, the sonic image created is very realistic.

## 7.2 SIGNAL MODEL

The source signals are modeled in the time domain as zero-mean normal stochastic processes that are mutually independent and non-stationary. Joint wide-sense stationarity is nevertheless assumed for the duration of a short time segment. The STPSD is used as a measure for how the mean signal power, or variance, distributes over time and frequency.<sup>1</sup> In line with this, a short-time Fourier coefficient represents a circular symmetric normal random variable:<sup>2</sup>

$$s_i(n, m) \sim \mathcal{N}[0, \sigma_{s_i}^2(m)] \circ \rightarrow S_i(k, m) \sim \mathcal{CN}[0, \Phi_{s_i}(k, m)], \quad (8)$$

where  $n$  is the time instant,  $m$  is the time segment,  $s_i(n, m)$  is the  $i$ th source signal in the  $m$ th time segment,  $\sigma_{s_i}^2$  is the variance,  $k$  denotes the frequency bin, and  $\Phi_{s_i}(k, m)$  is thus the STPSD.<sup>3</sup> The  $\circ \rightarrow$  symbol indicates the discrete Fourier transform (DFT). In the short-time Fourier transform (STFT) domain, for a given time segment, the set of source signal components is considered mutually independent, too. The sources and their components are thus uncorrelated, i.e. the short-time cross-spectral densities (STCSDs) are zero. The signals are all single-channel, i.e. *mono*.

1. See the Wiener–Khinchin convergence theorem

2. See the central limit theorem

3. Note that the STPSD is tantamount to the STFT of the auto-covariance function

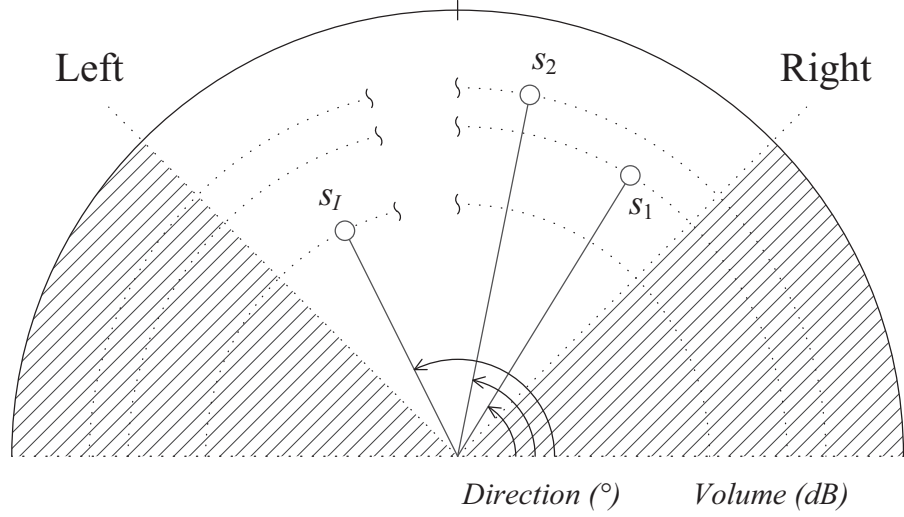


Figure 1: Modeling of sound sources in a sound field using the parameters *direction* (azimuth) and *volume* (radius)

### 7.3 SYSTEM MODEL

#### 7.3.1 Image Signal

Resorting to common studio practice, a sound source is assigned a location in the sound field via *pan control*. So, the image signal is <sup>4</sup>

$$\begin{aligned}
 \mathbf{u}_i(n, m) &= a_{1i} \mathbf{e}_1 s_i(n, m) + a_{2i} \mathbf{e}_2 s_i(n, m) \\
 &= \mathbf{a}_i s_i(n, m) \\
 &\quad \circ \\
 \underline{\mathbf{u}}_i(k, m) &= \mathbf{a}_i S_i(k, m),
 \end{aligned} \tag{9}$$

where  $\{\mathbf{e}_1, \mathbf{e}_2\}$  is the standard basis of  $\mathbb{R}^2$ ,  $\mathbf{a}_i = [a_{1i} \ a_{2i}]^T \in \mathbb{R}^2$  is the panning vector,  $s_i(n, m) \in \mathbb{R}$ , and  $S_i(k, m) \in \mathbb{C}$ . The panning vector is defined as

$$\mathbf{a}_i \triangleq \frac{\mathbf{a}_i}{\|\mathbf{a}_i\|} = \begin{bmatrix} \sin \alpha_i \\ \cos \alpha_i \end{bmatrix} \quad \text{with } \alpha_i \in [0^\circ, 90^\circ], \tag{10}$$

representing the spread of the source into the sound field. A source is placed at the panoramic angle  $\alpha_i$  between fully left and fully right as illustrated in Fig. 1. The placement can be chosen either arbitrarily or following some common mixing rules. The signal's volume can be considered inherent to the signal, i.e.

$$s_i \triangleq b_i s_{i0}, \tag{11}$$

<sup>4</sup> The notion of spatial “images” in a source separation context can also be found in [44], e.g.

where  $s_{i0}$  is the reference-level signal and  $b_i \in \mathbb{R}$  represents the desired volume level. Furthermore, due to the fact that

$$\|\mathbf{a}_i\|^2 = \sin^2 \alpha_i + \cos^2 \alpha_i = 1, \quad (12)$$

the source power level is kept constant across the two channels. The angle range is defined in such a way that at the lower end the source appears in only the right channel, whereas when placed at the upper end the source appears in only the left channel. In the middle, the signal power is equally distributed across the two channels and the source appears in a phantom center channel. A source is deemed to be unique among all sources if the associated angle is unique.

### 7.3.2 Mixture Signal

The mixture is obtained by superposition of distinct stereo images created according to (9). To account for professionally produced music recordings, each source signal is considered as having undergone prior processing in the form of linear and nonlinear audio effects [21]. Yet, the effects are not included in the model. The mixture signal is

$$\begin{aligned} \mathbf{x}(n, m) &= \sum_{i \in I} \mathbf{a}_i s_i(n, m) \\ &\quad \uparrow \\ \underline{\mathbf{x}}(k, m) &= \sum_{i \in I} \mathbf{a}_i S_i(k, m) \\ &= \mathbf{A} \underline{\mathbf{s}}(k, m), \end{aligned} \quad (13)$$

where  $\mathbf{x} = [x_1 \ x_2]^\top$ ,  $\underline{\mathbf{s}} = [s_1 \ s_2 \ \dots \ s_I]^\top$ ,  $\underline{\mathbf{x}} = [X_1 \ X_2]^\top$ , and  $I$  is the total number of sources. Note that the mixing system  $\mathbf{A} = [\mathbf{a}_1 \ \mathbf{a}_2 \ \dots \ \mathbf{a}_I]$  is time-invariant and memoryless, i.e. *instantaneous*.

## 7.4 PROBLEM FORMULATION

What is sought after is an operator  $F$  that transforms the mixture signal,  $\mathbf{x}(n)$ , into the source signal estimate,  $\hat{s}_i(n)$ , based on a priori knowledge about the mixing given a measurable signal characteristic which is perceptually relevant. The mixing system is fully described by the sources' locations and their filter responses. The signal metric provided by the source model is the STPSD. The latter is particularly suited for the task, because it can be interpreted in different ways: in a statistical, physical, and also perceptual sense. So, postulating the preservation of the original STPSD in the estimate, the problem can be formulated as follows. Given the model parameters

$$\theta_i = \{\alpha_i, \Phi_{s_i}(k, m)\} \quad \forall i, k, m, \quad (14)$$

find

$$\hat{s}_i(n) = F[\mathbf{x}(n) \mid \theta_1, \theta_2, \dots, \theta_I] \quad (15a)$$

subject to

$$\Phi_{\hat{s}_i}(k, m) = \Phi_{s_i}(k, m) \quad \forall i, k, m. \quad (15b)$$

## 7.5 PROPOSED SOLUTION

In order that the source signal components exhibit a better *disjoint orthogonality* [45] in comparison to the waveform domain, the mixture is mapped onto the Fourier time-frequency representation (TFR). The transformed mixture is so expressed in terms of Fourier coefficients. A coefficient pair  $\underline{\mathbf{x}}(k, m)$  is then decomposed into its constituents parts by means of linear spatial filtering according to

$$\begin{aligned} \hat{S}_i(k, m) &= w_{1i} \mathbf{e}_1 X_1(k, m) + w_{2i} \mathbf{e}_2 X_2(k, m) \\ &= \mathbf{w}_i^H \underline{\mathbf{x}}(k, m), \end{aligned} \quad (16)$$

where  $\mathbf{w}_i = [w_{1i}^* \ w_{2i}^*]^H \in \mathbb{C}^2$  is the spatial filter. As the mixing system that we seek to invert can be complex in general, so is the filter. From a geometrical viewpoint, the beam in Fig. 2 is steered and amplified or attenuated, such that the signal component in the direction of  $\alpha_i$  is preserved whereas the contribution from the interfering sources is canceled out or at least minimized. In the latter case, the spatial filter shall be constrained to adjust the mean output power of the estimate to the power level of the original component. The filter response

$$g(\alpha_i) = \mathbf{w}_i^H \mathbf{a}_i$$

from (6), where  $g \in \mathbb{C}$  in the general case, can be used to that end. In the final stage, the filtered signal components are recombined into an isolated version of  $\hat{s}_i(n)$ .

### 7.5.1 Well-Posed Case

The term “well posed” characterizes the case where the number of active sound sources in a TF point is at least one but not larger than the number of channels, which is two. In such a case, it exists one exact solution.

#### 7.5.1.1 Unity-gain spatial filter

Suppose that the mixture  $\underline{\mathbf{x}}(k, m)$  is made up of a single directional source component,

$$\underline{\mathbf{x}}(k, m) = \mathbf{a}_i S_i(k, m). \quad (17)$$



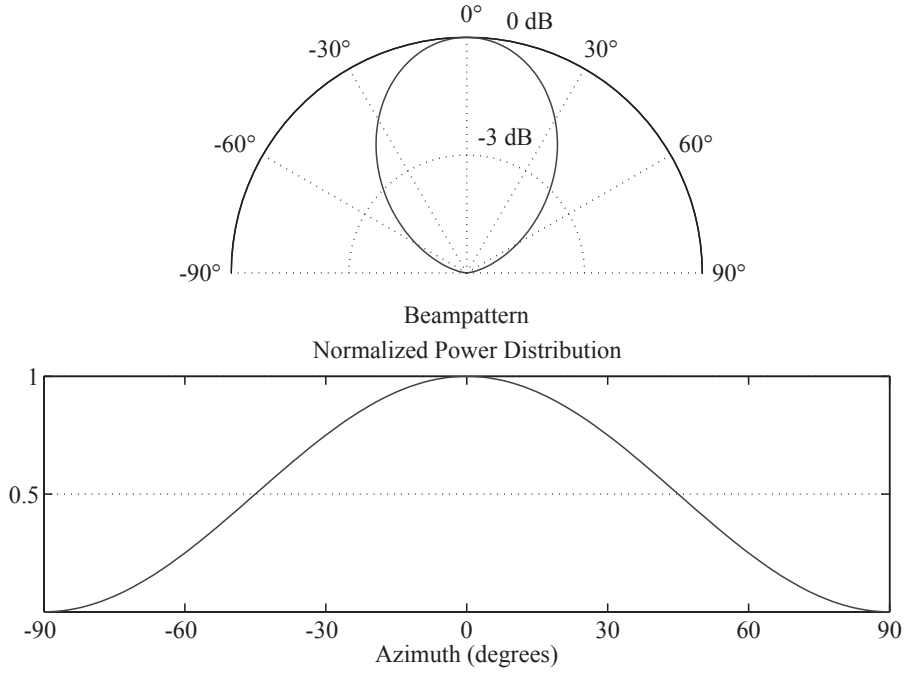


Figure 2: Spatial pattern and power distribution of a two-channel filter. The beam is gain adjusted and directed such that the signal of interest is preserved and either one interferer is suppressed or, in case of multiple interferers, their mean total power is minimized.

By comparing (12) with (6) under the unity-gain constraint,

$$g(\alpha_i) = 1, \quad (18)$$

the source component  $S_i(k, m)$  is extracted from the mixture  $\underline{x}(k, m)$  by setting

$$\mathbf{w}_i = \mathbf{a}_i, \quad (19)$$

so that

$$\hat{S}_i(k, m) = \mathbf{a}_i^T \underline{x}(k, m). \quad (20)$$

#### 7.5.1.2 Zero-forcing spatial filter

Let two sources contribute to the mixture  $\underline{x}(k, m)$  simultaneously:

$$\underline{x}(k, m) = \mathbf{a}_i S_i(k, m) + \mathbf{a}_l S_l(k, m), \quad (21)$$

where  $S_i(k, m)$  is the signal of interest and  $S_l(k, m)$  is the jammer. By enforcing identity for  $S_i(k, m)$  as in (18), and full cancellation of the jammer,

$$g(\alpha_l) = 0, \quad (22)$$

the sought-after weight vector becomes

$$\mathbf{w}_i = \mathbf{A}^{-T} \mathbf{g}_i, \quad (23)$$

where  $\mathbf{g}_i = [1 \ 0]^\top$  and  $\mathbf{A} = [\mathbf{a}_i \ \mathbf{a}_1]$ . Applying the above procedure for both sources yields the separation matrix

$$\mathbf{W} = \mathbf{A}^{-\top}, \quad (24)$$

and the source components are obtained by matrix inversion,

$$\hat{\underline{\mathbf{s}}}(k, m) = \mathbf{A}^{-1} \underline{\mathbf{x}}(k, m) \quad (25)$$

with  $\hat{\underline{\mathbf{s}}} = [\hat{s}_i \ \hat{s}_1]^\top$ .

### 7.5.2 Ill-Posed Case

The term ‘‘ill posed’’ characterizes the case where a unique solution to the separation problem does not exist, that is when the mixture is composed of more than two source signals. An optimal solution that further complies with (15b) can be found instead by means of linearly constrained minimum-variance (LCMV) spatial filtering [46, 47]. What is considered as the jammer now is the sum of all interfering source signals. In consequence, the mixture  $\underline{\mathbf{x}}(k, m)$  is expressed in terms of two components:

- a *unidirectional* signal of interest and
- a *multidirectional* jammer.

The corresponding equation is

$$\underline{\mathbf{x}}(k, m) = \mathbf{a}_i S_i(k, m) + \underline{\mathbf{r}}(k, m) \quad (26a)$$

with

$$\underline{\mathbf{r}}(k, m) = \sum_{l \in I, l \neq i} \mathbf{a}_l S_l(k, m). \quad (26b)$$

An estimate of the signal of interest  $\hat{S}_i(k, m)$  is found by minimizing the mean jammer power, or the mean filter output power, along the direction of the signal of interest,

$$P(\alpha_i) = \mathbf{w}_i^H \mathbf{R}_x(k, m) \mathbf{w}_i, \quad (27)$$

subject to identity with respect to a given power value  $\Phi_{s_i}(k, m)$ . In other words, we seek after the weight vector that solves the quadratic optimization problem

$$\begin{aligned} \mathbf{w}_{i0} &= \underset{\mathbf{w}_i}{\operatorname{arg\,min}} P(\alpha_i) \\ \text{s.t. } g(\alpha_i) &= \sqrt{\Phi_{s_i}(k, m) \mathbf{a}_i^T \mathbf{R}_x^{-1}(k, m) \mathbf{a}_i}. \end{aligned} \quad (28)$$

The solution to the above problem can be found by use of Lagrange multipliers, e.g., which yields

$$\mathbf{w}_{i0} = \mathbf{R}_x^{-1}(k, m) \mathbf{a}_i \sqrt{\frac{\Phi_{s_i}(k, m)}{\mathbf{a}_i^T \mathbf{R}_x^{-1}(k, m) \mathbf{a}_i}}. \quad (29)$$

When applied to the mixture signal  $\underline{x}(k, m)$ , the derived spatial filter will narrow the lobe of the jammer power spectrum, and the leakage from the interfering sources, thus, will be reduced. Due to the power constraint it is furthermore ensured that the mean estimate power is equal to the desired power value in every point of the TF plane. This is easily verified by plugging (29) into (27).

### 7.5.3 Narrowband Convolutional Case

Let the mixture be based on the narrowband assumption [48] which says that the channel's coherence bandwidth does not (significantly) fall below the bandwidth of a frequency bin. In that particular case, the channel transfer function is assumed constant over the frequency bin. This assumption holds for FIR filters if their order is strictly smaller than the STFT size, and it holds in good approximation for infinite impulse response (IIR) filters for which the impulse response decays sufficiently fast to a negligibly small value. A digital equalizer is well represented by such a system model. The respective mixture signal is

$$\begin{aligned} \mathbf{x}(n, m) &= \sum_{i \in I} \mathbf{a}_i [h_i(n) \star s_i(n)](m) \\ &\quad \uparrow \\ \underline{\mathbf{x}}(k, m) &\approx \sum_{i \in I} \mathbf{a}_i H_i(k) S_i(k, m) \\ &= \mathbf{A} \underline{\mathbf{H}}(k) \underline{\mathbf{s}}(k, m), \end{aligned} \tag{30}$$

where  $h_i(n)$  is a time-invariant filter response,  $\star$  denotes convolution,  $H_i(k) \in \mathbb{C}$  is the filter transfer function, and  $\underline{\mathbf{H}} = \text{diag}(H_1, H_2, \dots, H_I)$ . As a side note, *binaural mixing* is inherent to this model thanks to the relatively small order of a head-related impulse response (HRIR).<sup>5</sup> In that case, the term  $\mathbf{a}_i h_i(n)$  above has simply to be replaced by  $\mathbf{h}_i(n)$ , i.e. a *multichannel* filter response. If (30) is used instead of (13), the spatial filtering operation in (16) can be extended in such a manner that distinct source signals are separated and *deconvolved* as opposed to what has been plain demixing before.

---

5. In [49, 50], e.g., the HRIR is 200 taps long

## 7.5.4 Two-Channel Case

So far, a source has been considered as single-channel. To account for two-channel, i.e., stereo sources, the model from (8) is extended in the following manner:

$$\begin{aligned} \begin{bmatrix} s_{1i}(n, m) \\ s_{2i}(n, m) \end{bmatrix} &\sim \mathcal{N}[\mathbf{o}, \boldsymbol{\Sigma}_{s_i}(m)] \\ \circ \bullet \begin{bmatrix} S_{1i}(k, m) \\ S_{2i}(k, m) \end{bmatrix} &\sim \mathcal{CN}[\mathbf{o}, \boldsymbol{\Phi}_{s_i}(k, m)], \end{aligned} \quad (31)$$

where  $\boldsymbol{\Sigma}_{s_i}$  and  $\boldsymbol{\Phi}_{s_i}$  are  $2 \times 2$  covariance matrices. The two channels of a stereo source may be interpreted as two separate mono sources, and a stereo source can be thought of as a centered image of a sound source that was recorded with two independent microphones.

A stereo source, i.e. its centered image, is positioned in the sound field via *balance control* according to

$$\begin{aligned} \mathbf{u}_i(n, m) &= a_{1i} \mathbf{e}_1 s_{1i}(n, m) + a_{2i} \mathbf{e}_2 s_{2i}(n, m) \\ &= \mathbf{a}_i \circ \mathbf{s}_i(n, m) \\ &\quad \circ \bullet \\ \underline{\mathbf{u}}_i(k, m) &= \mathbf{a}_i \circ \underline{\mathbf{s}}_i(k, m), \end{aligned} \quad (32)$$

where  $\mathbf{s}_i = [s_{1i} \ s_{2i}]^T$ ,  $\underline{\mathbf{s}}_i = [S_{1i} \ S_{2i}]^T$ , and  $\circ$  denotes the Hadamard or entrywise product. The balance vector is defined as

$$\mathbf{a}_i \triangleq \frac{\mathbf{a}_i}{a_{\text{ref},i}} \quad \text{with } a_{\text{ref},i} = \begin{cases} a_{1i} & \text{if } a_{1i} \geq a_{2i}, \\ a_{2i} & \text{otherwise.} \end{cases} \quad (33)$$

The instantaneous mixture signal is thus given by

$$\begin{aligned} \mathbf{x}(n, m) &= \sum_{i \in I} \mathbf{a}_i \circ \mathbf{s}_i(n, m) \\ &\quad \circ \bullet \\ \underline{\mathbf{x}}(k, m) &= \sum_{i \in I} \mathbf{a}_i \circ \underline{\mathbf{s}}_i(k, m) \\ &= \mathbf{A} \text{vec} \underline{\mathbf{S}}(k, m), \end{aligned} \quad (34)$$

where  $\underline{\mathbf{S}} = [\underline{\mathbf{s}}_1 \ \underline{\mathbf{s}}_2 \ \dots \ \underline{\mathbf{s}}_I]$ ,  $\mathbf{A} = [\text{diag} \mathbf{a}_1 \ \text{diag} \mathbf{a}_2 \ \dots \ \text{diag} \mathbf{a}_I]$ , and  $\text{vec}$  denotes the vectorization formed by stacking the vectors  $\underline{\mathbf{s}}_i$ ,  $i = 1, 2, \dots, I$ , on top of each other into a single column vector. A stereo source component is separated from the mixture by estimating the left-channel and the right-channel component simultaneously according to

$$\hat{\underline{\mathbf{s}}}_i(k, m) = \mathbf{W}_{i0}^H \underline{\mathbf{x}}(k, m) \quad (35)$$

with

$$\mathbf{W}_{i0} = \mathbf{R}_x^{-1}(k, m) \text{diag } \mathbf{a}_i \Phi_{s_i}(k, m) \cdot \text{diag} \left\{ \frac{[\Phi_{s_i}(k, m)]_{ll}}{[\Phi_{s_i}]_{*,l}^H \text{diag } \mathbf{a}_i \mathbf{R}_x^{-1}(k, m) \text{diag } \mathbf{a}_i [\Phi_{s_i}]_{*,l}} \right\}_{l=1,2}^{1/2}, \quad (36)$$

where  $[\Phi_{s_i}]_{*,l}$  is the  $l$ th column of  $\Phi_{s_i}$ . Equation (36) thus represents the two-channel counterpart to (29). This can be shown by replacing  $\text{diag } \mathbf{a}_i$  by  $\mathbf{a}_i$  and  $\Phi_{s_i}$  by  $\Phi_{s_i}$  ( $l = 1$  in that case).

### 7.5.5 Noisy Case

In the case where the mixture is watermarked [51, 52], the model in (13), or (30), can be extended by a noise term in the following way:

$$\underline{\mathbf{x}}^w(k, m) = \underline{\mathbf{x}}(k, m) + \underline{\mathbf{n}}(k, m), \quad (37)$$

where  $\underline{\mathbf{n}}$  is an additive noise component in the respective TF point. Due to the fact that high-capacity watermarking techniques exploit the on-frequency masking phenomenon, the noise term is assumed to be *collinear* with the noise-free mixture signal, which results in the following relation:

$$\underline{\mathbf{x}}^w(k, m) = [1 + \eta(k, m)] \underline{\mathbf{x}}(k, m) \quad (38)$$

with

$$\underline{\mathbf{n}}(k, m) = \eta(k, m) \underline{\mathbf{x}}(k, m), \quad (39)$$

where  $\eta \in \mathbb{C}$  represents the corruption due to the watermark. From (38) it can be seen that the estimate  $\hat{S}_i(k, m)$  must be rectified by the term  $[1 + \eta(k, m)]^{-1}$  to compensate for the watermark. In the general case, however,  $\eta(k, m)$  will be unknown. We can yet give an estimate for the deviation of the magnitude with regard to a noise-free power value  $\Phi_{s_i}(k, m)$ , which is

$$|1 + \eta(k, m)| = \frac{|\hat{S}_i(k, m)|}{\sqrt{\Phi_{s_i}(k, m)}}. \quad (40)$$

This a posteriori estimate for magnitude distortion can then be used to partly compensate for errors due to watermarking.

## 7.6 GENERAL REMARKS

### 7.6.1 Precedence or Haas Effect

If a timing difference between the two channels of an image signal is wished for and the delay in samples is sufficiently small compared

to the [STFT](#) length so that the two windowed signals carry almost the same content, a frequency-dependent phase shift can be added to the panning vector. The new vector is then complex (and so will be the spatial filter):

$$\mathbf{a}_i(\mathbf{n}) = \begin{bmatrix} \sin \theta_i \delta(\mathbf{n} - \mathbf{n}_1) \\ \cos \theta_i \delta(\mathbf{n} - \mathbf{n}_2) \end{bmatrix} \circ \bullet \mathbf{a}_i(\mathbf{k}) = \begin{bmatrix} \sin \theta_i \omega^{-\mathbf{k} \mathbf{n}_1} \\ \cos \theta_i \omega^{-\mathbf{k} \mathbf{n}_2} \end{bmatrix}, \quad (41)$$

where  $\delta(\cdot)$  is the Dirac delta function,  $\omega$  is the  $N$ th primitive root of unity,  $\omega = e^{j2\pi/N}$ , and  $|\mathbf{n}_2 - \mathbf{n}_1|$  is in the range of 1 to 5 ms [53, 54]. In this way, the perceived width of the sound source can be increased.

### 7.6.2 Multi-Constraint Spatial Filter

If the constraints from (18) and (22) are also imposed on the filter in (28), the obtained solution folds up to (23). This is simply because no degrees of freedom are left with regard to the number of weight coefficients to minimize the mean jammer power. As a consequence, the filter is suboptimal in the case of multiple interferers: Canceling out just one of the interfering sources, analogously to (23), leaves a strong residual which is further amplified by the filter.

### 7.6.3 Wiener vs. [PCMV](#) Spatial Filter

It is well known that the classical Wiener filter minimizes the mean noise power at the cost of the signal of interest, which is to say that the signal of interest is also attenuated at the attempt to improve the [SNR](#) at the output. One can therefore expect the estimated signal spectra to be attenuated depending on whether the [SNR](#) in a [TF](#) point is high or low. As a direct consequence of this, the spectra of the source signals with a poor [SNR](#) exhibit missing spectral components after filtering, which may deteriorate the quality of the listening experience.

From the fact that the filter in (29) and the Wiener spatial filter are collinear, one can infer that their beams have the same look direction but different gains. The power-conserving minimum-variance ([PCMV](#)) spatial filter adapts the gain in order to conform with the quadratic constraint, whereas the Wiener spatial filter will likewise power down the output signal for the sake of a lower [MSE](#), see Appendix A. The [PCMV](#) spatial filter, hence, is capable of overcoming the issue of spectral gaps, so that the replica of the original source signals are perceptually more similar to the latter in timbre. It also preserves the auditory signal bandwidth, which is essential for a natural listening experience. As a side note, this issue gave rise to various bandwidth extension techniques in the past, see [55] and the references therein.

## 7.6.4 Mixture Covariance Matrix

The local mixture spatial covariance matrix is given by

$$\mathbf{R}_x(k, m) = \mathbf{E} [\mathbf{x}(k, m)\mathbf{x}^H(k, m)]. \quad (42)$$

For an instantaneous mixture, the above expression yields

$$\mathbf{R}_x(k, m) = \sum_{p \in P} \mathbf{a}_p \mathbf{a}_p^T \Phi_{s_p}(k, m) + \sum_{q \in Q} \mathbf{a}_q \mathbf{a}_q^T \circ \Phi_{s_q}(k, m) \quad (43)$$

with

$$\Phi_{s_q}(k, m) = \begin{bmatrix} \Phi_{s_{1q}}(k, m) & \Phi_{s_{1q}s_{2q}}(k, m) \\ \Phi_{s_{1q}s_{2q}}^*(k, m) & \Phi_{s_{2q}}(k, m) \end{bmatrix}, \quad (44)$$

where the subset  $P = \{i \in I \mid \forall n [s_{1i}(n) = s_{2i}(n) = s_i(n)]\}$  contains the mono sources, while  $Q = \{i \in I \mid \exists n [s_{1i}(n) \neq s_{2i}(n)]\} = I \setminus P$  comprises the stereo sources, respectively.  $\Phi_{s_{1q}s_{2q}}(k, m)$  is the  $q$ th source's,  $q \in Q$ , *STCSD* between the left and the right channel, and  $\Phi_{s_{1q}s_{2q}}^*(k, m)$  is its conjugate.  $\mathbf{R}_x(k, m)$  is nonsingular if there are (at least) two mono sources in a *TF* point contributing from different angles.<sup>6</sup> It is further Hermitian and positive-semidefinite.

To compute the *PCMV* stereo filter in (36), one requires the *STCSD*s, which can be calculated as the *STFT* of the block cross-covariances. To avoid the extra computational effort and to save on the data rate, one might consider skipping this step. As a result, (44) can be simplified as

$$\Phi_{s_q}(k, m) = \text{diag} [\Phi_{s_{1q}}(k, m), \Phi_{s_{2q}}(k, m)], \quad (45)$$

and the *PCMV* stereo filter then becomes

$$\mathbf{W}_{i_o} = \mathbf{R}_x^{-1}(k, m) \text{diag} \left\{ \frac{[\Phi_{s_i}(k, m)]_{11}}{[\mathbf{R}_x^{-1}(k, m)]_{11}} \right\}_{l=1,2}^{1/2}. \quad (46)$$

Using (42) and (45),  $\mathbf{R}_x(k, m)$  can be reconstructed from the panning angles, the balance ratios, and the *STPSD*s exclusively—and so  $\mathbf{W}_{i_o}$ .

From (45) and (46) it can be seen that when multiple stereo sources are present in the mixture, their component estimates have the same phase; only their spectral envelopes are shaped differently. Further, if the mixture is a combination of only stereo sources,  $\mathbf{W}_{i_o}$  is diagonal. As a consequence, each source component is filtered from either the left or the right channel using the *PCMV mono* filter

$$w_{i_o} = \sqrt{\frac{\Phi_{s_i}(k, m)}{\sum_{l \in I} \alpha_l^2 \Phi_{s_l}(k, m)}}, \quad (47)$$

which resembles Faller's filter in [56]. Please note that the filter in (47) is not the *square-root* Wiener filter [57], since the mixing coefficient  $\alpha_i$  (magnitude or square) is missing in the numerator. See also [58].

6. This is by definition the case for a stereo source

## 7.7 PARAMETER QUANTIZATION AND CODING

Here, a scalable quantization and coding strategy with a compact representation of side information is proposed. Other strategies, for spectrogram coding in particular, can be found in [40]. Its advantage lies in the fact that it can be applied block by block, requiring hence less working memory.

7.7.1 *Panoramic Angle*

The panoramic angle  $\alpha$  of a mono source is rounded to the nearest integer value using a mid-tread uniform quantizer defined as

$$Q(x) = \text{sgn}(x) \cdot \Delta \cdot \left\lfloor \frac{|x|}{\Delta} + \frac{1}{2} \right\rfloor, \quad (48)$$

where  $\Delta$  is the step size and  $\lfloor \cdot \rfloor$  represents the floor function.

7.7.2 *Balance Ratio*

The balance ratio  $a_{\text{-ref}}/a_{\text{ref}}$  of a stereo source is encoded using an A-law or a  $\mu$ -law compressor in combination with the quantizer from (48). For a given input  $x$ ,  $|x| \leq x_{\text{max}}$ , the A-law compressor output is

$$C_A(x) = \text{sgn}(x) \begin{cases} \frac{A \cdot |x|}{1 + \log(A)} & \text{if } 0 \leq |x| \leq \frac{x_{\text{max}}}{A}, \\ \frac{x_{\text{max}} \cdot [1 + \log(A \cdot |x|/x_{\text{max}})]}{1 + \log(A)} & \text{if } \frac{x_{\text{max}}}{A} < |x| \leq x_{\text{max}}, \end{cases} \quad (49)$$

where  $A$  is the compression parameter and  $\log$  is the logarithm. The output of the  $\mu$ -law compressor is

$$C_\mu(x) = \text{sgn}(x) \cdot \frac{x_{\text{max}} \cdot \log(1 + \mu \cdot |x|/x_{\text{max}})}{\log(1 + \mu)}, \quad (50)$$

where  $\mu$  is the associated compression parameter. Using A-law or  $\mu$ -law compression, the signal-to-distortion ratio is kept constant over a broad range of  $x$  [59]. Common values for  $A$  and  $\mu$  are 87.7 and 255, respectively.

7.7.3 *Short-Time Power Spectrum*7.7.3.1 *Irrelevancy reduction*

A significant reduction of side information can be achieved in two steps: by reducing the frequency resolution of the STPSD  $\Phi_{s_i}(k, m)$  in approximation of the critical bands [60] and by quantizing the STPSD values in relation to an appropriate psychoacoustic criterion.



The peripheral auditory system is typically modeled as a bank of overlapping bandwidth filters, the auditory filters, which possess an equivalent rectangular bandwidth (ERB). The ERB-rate scale puts into relation the center frequency of auditory filters with units of the ERB. Using the ERB-rate function [61] we can define a relation between the frequency bin  $k$  and the critical-band index  $z_k$  by

$$z_k \triangleq \lfloor 21.4 \log_{10} (1 + 4.37 f_s / Nk) \rfloor, \quad (51)$$

where  $N$  is the STFT length and  $f_s$  is the sampling frequency in kHz. The  $z$ th critical-band value is then computed as the arithmetic mean between  $\text{lb}(z) = \inf\{k \mid z_k = z\}$  and  $\text{ub}(z) = \sup\{k \mid z_k = z\}$ , i.e.

$$\bar{\Phi}_{s_i}(z, m) = \frac{1}{\text{ub}(z) - \text{lb}(z) + 1} \sum_{k=\text{lb}(z)}^{\text{ub}(z)} \Phi_{s_i}(k, m). \quad (52)$$

Further, under the assumption that the the minimum just-noticeable difference (JND) level and so the maximum quantization error is 1 dB [60], the step size  $\Delta$  is chosen as 2 dB, and the  $\bar{\Phi}_{s_i}(z, m)$  values are quantized using (48) according to

$$\frac{\Phi_{s_i}^Q(z, m)}{2} = \left\lfloor \frac{10}{2} \log_{10} \bar{\Phi}_{s_i}(z, m) + \frac{1}{2} \right\rfloor. \quad (53)$$

In [41] it is also shown that a perceptually motivated approximation of the STPSD is sufficiently precise to achieve high similarity with the original signals.

### 7.7.3.2 Redundancy reduction

To reduce the amount of side information even more, one can use the correlation of STPSD indices between adjacent TF points. This can be achieved by taking the difference between two consecutive STPSD indices in the direction of time, frequency, or between channel pairs, and by coding the difference signal based on its lower entropy. That principle is known as differential pulse-code modulation (DPCM). It is further advisable to use a nonuniform time resolution to better take time-varying aspects of music signals into account [62, 63].

### 7.7.3.3 Dequantization

As spatial filtering is carried out based on the availability of STPSDs, the quantized STPSD values must be converted back into their linear counterparts and extrapolated to the resolution of the STFT. This can be done by taking

$$\tilde{\Phi}_{s_i}(k, m) = 10^{\Phi_{s_i}^Q(z, m)/10} \quad \forall k: z_k = z. \quad (54)$$

## 7.8 PERFORMANCE EVALUATION

7.8.1 *Instantaneous Mixture with Watermarking*

In this section, the proposed algorithm which was given the name “Underdetermined Source Signal Recovery” (USSR) is compared with an in-house implementation of the algorithm described in [39] and its predecessor [41]. The four algorithms under test hence are:

ISSA The reference ISS algorithm [39]

USSR-A The predecessor of the proposed algorithm [41]

USSR-B The proposed algorithm using the PCMV spatial filter

USSR-C The proposed algorithm using the Wiener spatial filter

7.8.1.1 *Experimental setup*

To exclude a performance bias due to different TFRs, all algorithms are implemented using the STFT. The latter is realized by means of a 2048-point fast Fourier transform (FFT) with a Kaiser–Bessel derived (KBD) window of the same length and a 50-% overlap. The sampling rate is set to 44.1 kHz. The effective data rate of ISSA is 86.1 kbps for the 5-track mixture and 108 kbps for the 7-track mixture. The USSR algorithm is adjusted in such a way that its data rate is more or less the same: 93.4 and 103 kbps. Also, the same watermarking algorithm [52] is used in all four cases.

The following four metrics are used to objectively assess the sound quality: the signal-to-interference ratio (SIR), the so-termed “target-related” perceptual score (TPS), a frequency-weighted signal-to-noise ratio (SNRF) [64], and the “auditory” bandwidth as the counterpart of the “articulatory” bandwidth [64]. The first two metrics are computed with the PEASS toolkit [65]. The SNRF is redefined according to:

$$\text{SNRF}_i(m) \triangleq \frac{1}{Z} \sum_{z=1}^Z 10 \log_{10} \frac{\bar{\Phi}_{s_i}(z, m)}{\bar{\Phi}_{n_i}(z, m)}, \quad (55)$$

where  $z$  is the ERB-scale index,  $Z = 39$ ,  $\bar{\Phi}_{s_i}(z, m)$  is the band’s signal power, and  $\bar{\Phi}_{n_i}(z, m)$  is the corresponding noise power,

$$\bar{\Phi}_{n_i}(z, m) = \sum_{k=\text{lb}(z)}^{\text{ub}(z)} \frac{\{\min[|\hat{S}_i(k, m)| - |S_i(k, m)|, 0]\}^2}{\text{ub}(z) - \text{lb}(z) + 1}. \quad (56)$$

The noise signal is calculated in such a way that the subjective effect of spectral gaps on sound quality is accentuated, as only *lacking* sound components are taken into account. Furthermore, a time resolution of 23.2 ms is used for both the SNRF and the bandwidth metric. The end result is obtained by taking the average over all time segments.

As a supplement, a multi-stimulus test with hidden reference and anchor (MUSHRA) [66] is administered. It should serve as a check for

consistency between the four chosen metrics and human perception. The test is carried out in the audiovisual production facilities within the University of Western Brittany. Sennheiser’s HD 650 headphones and MOTU’s UltraLite-mk3 Hybrid audio interface are used during the test. The gain of the preamplifier is adjusted to a listening level of  $-20$  dB below the clipping level of a digital tape recording. The test signals are shortened to 20 s at the longest. The anchor is a 3.5-kHz lowpass filtered sum of the original tracks with 3 dB SIR and a 50-% spectral-gap rate. The anchor is altered in such a way that it shows similar types of impairment as the algorithms under test. A panel of nine audiovisual media students take part in the test. They are asked to score the stimuli according to the continuous quality scale, giving their degree of preference for one type of artifact versus some other type.

Two music pieces from different genres are selected: a 5-track hip-hop mixture and a more complex 7-track pop-rock mixture. The hip-hop piece is DJ Vadim’s “The Terrorist”. It is composed of a leading vocal, a synthesizer in the bassline, and a percussion section that has a kick, a snare, and a hi-hat. Phoenix’s “Lisztomania” is chosen from within the pop-rock genre. It has a bass guitar together with drums forming the rhythmic section, several guitars in the harmonic section, a vocal melody, and a keyboard to create a sustained pad for the piece. The signals are 30 s long monophonic excerpts from the multitrack masters. The spatial placement of individual tracks is aligned with the commercial releases, see Table 2.

#### 7.8.1.2 Experimental results

The results of the experiment are summarized in Figs. 3–7. Figs. 3–4 show the SIR and the TPS for each track from the two music excerpts. The corresponding SNRF and auditory bandwidth values are depicted in Figs. 5–6. The mean opinion scores (MOSs) with 95-% confidence intervals are plotted in Fig. 7.

As it was anticipated, USSR-C has the highest SIR. USSR-B shows a clear improvement over USSR-A. The SIR for ISSA is also quite high but always lower than for USSR-B and USSR-C, however. The TPS is fairly consistent in all three USSR variants for the hip-hop mixture, whereas a slight tendency towards USSR-B can be observed for the pop-rock mixture. ISSA has the worst TPS of all tested algorithms. In regard to the SNRF, the constrained USSR variants, A and B, perform better than the rest. Again, this is something that could be expected, since these algorithms preserve the auditory bandwidth of the signal. Further, it can be seen that the number of spectral gaps is smaller for USSR-C than for ISSA. De facto, the effect observed with USSR-C is more of a band limitation than the “arid” effect [41], and as such it produces a sound that is rather “dull” than “annoying”. Overall, the preferential tendencies of target-related perceptual score (TPS)

| TRACK                       | PANNING     |
|-----------------------------|-------------|
| Acapella                    | 6.7 % right |
| Bass                        | 20 % left   |
| Hi-Hat                      | 29 % left   |
| Kick                        | 6.7 % left  |
| Snare                       | centered    |
| “The Terrorist” by DJ Vadim |             |
| Bass                        | 1.6 % right |
| Beat                        | 4.4 % left  |
| Cocotte                     | 41 % right  |
| Guitar 1                    | 9.3 % left  |
| Guitar 3                    | 76 % left   |
| Key                         | 9.6 % right |
| Vox                         | 4.0 % right |
| “Lisztomania” by Phoenix    |             |

Table 2: Panning used for the two music pieces

are rather consistent with the MOS. Yet, the TPS seems to overrate the sound quality by some 20–40 points, which corresponds to 1–2 grades. In this regard, the SNRF provides the desired tendencies as well. This allows the conclusion that if the SNRF were properly scaled, it might just as well serve as an objective metric for the perceived sound quality, but at a much lower cost.

With a mean score between “fair” and “good”, the USSR algorithm is the clear winner in any of its variants. ISSA is graded as “bad” on the average, but better than the anchor. A slight preference for USSR-B, the proposed algorithm, over its predecessor USSR-A can also be noted. That preference seems to be linked with the complexity of the mixture. After all, USSR-B is assessed to perform *significantly* better than USSR-C, which once more highlights the fact that full bandwidth is essential for a natural listening experience.

### 7.8.2 Narrowband Convolutional Mixture

In this section, the variants B and C of the proposed algorithm are compared against each other by applying them now to a narrowband convolutional mixture. Variant C employs the MMSE alias Wiener spatial filter as opposed to variant B which uses the PCMV spatial filter, see also Section 7.8.1. In addition, different coding strategies for the spectrograms are applied. The PCMV spatial filter is used in tandem

with the strategy from Section 7.7.3, whereas the MMSE spatial filter is used in conjunction with JPEG compression or NMF with low-rank modeling. Variant C is hence equivalent to the algorithm described in [40] with the *generalized* Wiener filter replaced by its *spatial* counterpart, as in [67].

#### 7.8.2.1 Experimental setup

A set of fourteen musical excerpt from different genres is used. An excerpt is between 15 s and 35 s long and it comprises 5 to 10 tracks. Most of the excerpts have 5 tracks. The CIPIC head-related transfer function (HRTF) database [50] is used to simulate the channel, i.e. the transmission medium. The sources are placed at different angles separated by  $5^\circ$  or more from each other. Their placement is considered known at the decoder. The STPSDs are quantized and coded at different bitrates using one of the three strategies mentioned above. The HRTFs are derived from the pan angles taking the measurements for the KEMAR mannequin and are thus not included in the bitrate. The sampling rate is set to 44.1 kHz.

The objective sound quality is assessed with the aid of two metrics: the signal-to-distortion ratio (SDR) from BSS Eval [68, 69] and PEMO-Q's perceptual similarity metric (PSM). While the BSS Eval toolbox is used to measure the performance of "blind" source separation algorithms in the first place, the PEMO-Q [70, 71] software on the other hand is meant to predict the quality of low-bitrate speech and audio coders. Both metrics should hence complement each other.

#### 7.8.2.2 Experimental results

The results of the experiment are summarized in Fig. 8. The SDR for the "oracle" MMSE spatial filter that has perfect knowledge of the spectrograms, or STPSDs, is shown in Fig. 8a and the PSM value for each of the fourteen excerpts is shown in Fig. 8b, respectively. As can be seen, both metrics vary significantly from excerpt to excerpt. From this one can infer that the estimator's performance depends on the number of sources, their spatial location, and their spectro-temporal overlap. Also, the metrics are inconsistent across excerpts. The mean oracle SDR for the dataset lies around 10 dB and the mean PSM value is approximately 0.85.

The "rate-performance" curves for the three different filtering and/or coding strategies are given in Figs. 8c and 8d for the two metrics. They are obtained by subtracting the oracle value from the respective SDR or PSM value for a track and by taking the average over all tracks and excerpts given a side-information rate. Although one could have expected that either of the two SDR-curves for the MMSE filter would overtop the SDR-curve for the PCMV filter, like the oracle, they do not. This raises the question of whether the proposed coding strategy is

more efficient than the ones presented in [40]. The train of thought is as follows. In the previous experiment it was observed that the SIR is higher for the MMSE spatial filter if the same coding strategy is applied, see Figs. 3–4. Thus, one may conclude that if, e.g., NMF and DPCM coding contributed to the distortion to the same extent, the SIR for the MMSE spatial filter would also be higher in the convolutional case, and possibly the SDR as well. Looking at the PSM-curves, one can see that the combination of PCMV filtering with DPCM yields better sound quality than the oracle at any tested side-information rate. JPEG compression seems to have a higher coding efficiency than the NMF except for a very low bitrate region between 2–10 kbps per source in Fig. 8d. Also, it can be observed that the proposed algorithm provides roughly the same sound quality as the iterative predecessor algorithm [67] at a much lower computational cost.

### 7.8.3 Separation of Spatial Images

In this section, the proposed algorithm is evaluated by applying it to a set of professionally produced music recordings from the SiSEC 2013 [72] website. The task is to decompose an artistic mixture into a subset of images that represent the foreground objects and the image of the background—where applicable. The term “background” refers to the sum of background objects. The original images are given as a reference.

#### 7.8.3.1 Experimental setup

The following testing framework is used. With respect to the STFT, a 2048-point FFT is employed with a KBD window and a 50-% overlap between succeeding frames. The pan angle  $\alpha$  is quantized with 7 bits while the balance ratio  $\alpha_{\text{-ref}}/\alpha_{\text{ref}}$  is quantized with 16 bits using the A-law compander with  $A = 87.6$ . The STPSD is quantized with 6 bits per value on a 76-band nonuniform frequency scale. The probability mass function of the difference between contiguous STPSD values is modeled with a Laplace  $(\mu, b)$  distribution with  $\mu = -0.2$  and  $b = 2$ . The STCSDs are ignored by setting them to zero (no covariance).

The evaluation criteria suggested by the SiSEC 2013 committee are used. These include the performance metrics from the PEASS toolkit [65, 73] and the decoder runtime in seconds times CPU clock rate in GHz. The side-information rate is also given. In addition, PEMO-Q’s perceptual similarity metric  $\text{PSM}_t$  and its objective difference grade (ODG) are computed as well [70, 71].

The SiSEC development set is used. It consists of five music pieces from different genres. The image parameters are estimated from the provided stereo tracks according to the protocol in Appendix B. The results are shown in Table 3.

| TRACK                                 | TYPE   | $\alpha$ | $a_{\neg\text{ref}}/a_{\text{ref}}$ |
|---------------------------------------|--------|----------|-------------------------------------|
| Vocal                                 | stereo | —        | 0.89                                |
| Drums                                 | stereo | —        | 1.00                                |
| Guitar                                | stereo | —        | 0.96                                |
| “The Ones We Love” by Another Dreamer |        |          |                                     |
| Vocal                                 | stereo | —        | 0.99                                |
| Bass                                  | mono   | 45.0     | —                                   |
| Piano                                 | stereo | —        | 0.83                                |
| Background                            | stereo | —        | 0.94                                |
| “Roads” by Bearlin                    |        |          |                                     |
| Vocal                                 | stereo | —        | 0.90                                |
| Drums                                 | stereo | —        | 0.99                                |
| Bass                                  | mono   | 45.0     | —                                   |
| Claps                                 | stereo | —        | 0.99                                |
| Background                            | stereo | —        | 0.97                                |
| “Remember the Name” by Fort Minor     |        |          |                                     |
| Vocal                                 | mono   | 47.9     | —                                   |
| Guitar                                | stereo | —        | 0.97                                |
| “Que Pena/Tanto Faz” by Tamy          |        |          |                                     |
| Vocal                                 | stereo | —        | 1.00                                |
| Drums                                 | stereo | —        | 0.97                                |
| Bass                                  | stereo | —        | 0.93                                |
| Background                            | stereo | —        | 0.93                                |
| Ultimate NZ Tour                      |        |          |                                     |

Table 3: Estimated image parameters for the development set

### 7.8.3.2 Experimental results

The results of the experiment are summarized in Table 4. As can be observed, the image-to-spatial distortion ratio (*ISR*) is between 6.66 and 17.4 dB for a stereo source and it is greater or equal to 18.5 dB for a mono source. Equally, the highest source-to-artifacts ratio (*SAR*) is obtained for a mono source, which is 27.7 dB. The *TPS* shows a weak correlation not only with the *ISR* and *SAR* but also with PEMO-Q's perceptual similarity metric  $PSM_t$ , which then again does not take *spatial hearing* effects into account. The lowest *TPS* is at 52 %. The measured side-information rate is around 10 kbps per mono source or stereo channel. The execution time of the decoder is low and also faster than real time.

Table 5 compares the performance of the proposed algorithm with the figures reported in *SiSEC* 2011 for two oracle systems [74]. Their performance figures give an upper bound for binary masking based systems. In Table 5, positive delta values are in boldface. A significant improvement can be noticed for all items with regard to the *SAR*, up to 22.4 dB. The *TPS* is also higher in most cases, and so is the *SDR*. A comparison with more recent approaches is made available on the *SiSEC* 2013 website. A brief summary is given in Figs. 9 and 10. Even though the bass and the drums estimates of the proposed algorithm are rated lower than other comparable algorithms, the corresponding mean *ISR* and *TPS* are clearly the highest. These are the most relevant metrics here. It can further be observed that six out of eight metrics show better-than-oracle performance for the proposed system. Only the two interference and artifacts related metrics are worse. This can be considered as the *signal-noise uncertainty principle*. The more of the signal is to be preserved in the estimate, the more noise (interference or artifacts) is to be accepted. Or, the less noise is desired in the estimate, the more of the signal is to be sacrificed. The *SiSEC* experiment finally proves that a binary-mask oracle is suboptimal.



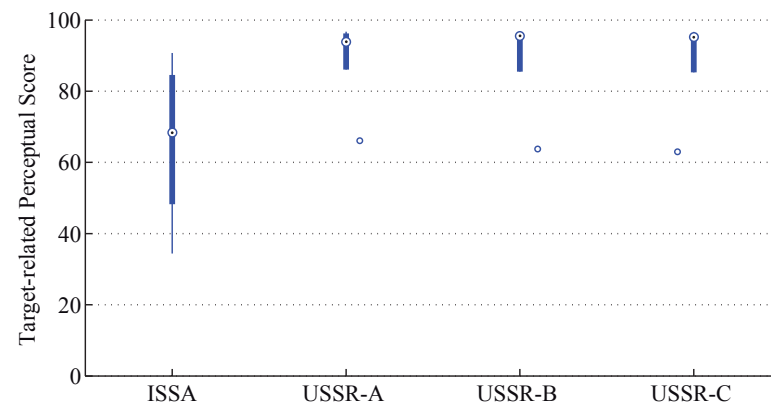
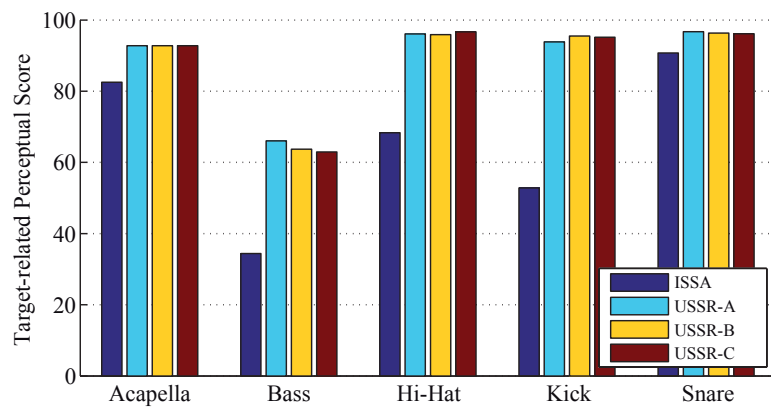
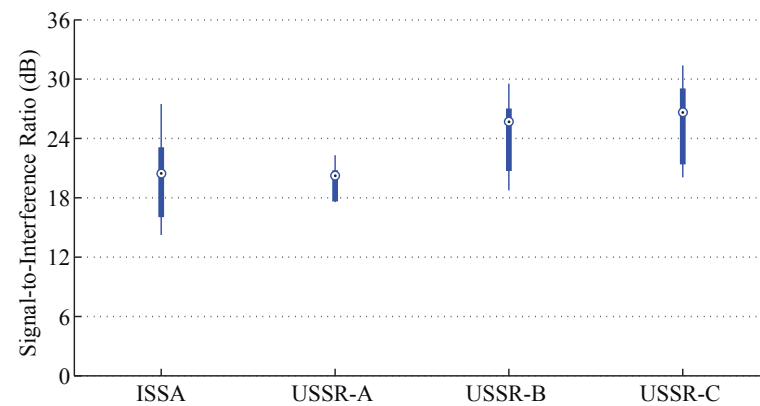
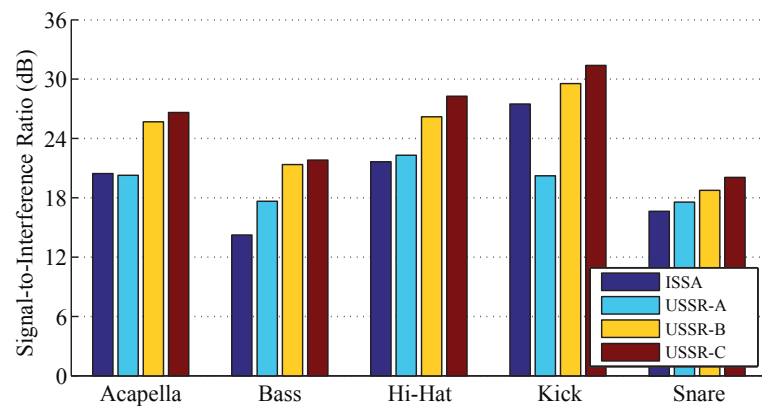


Figure 3: SIR and TPS values (left column) complemented by the median, the 25th and 75th percentiles, and outliers (right column) for an excerpt from "The Terrorist" by DJ Vadim

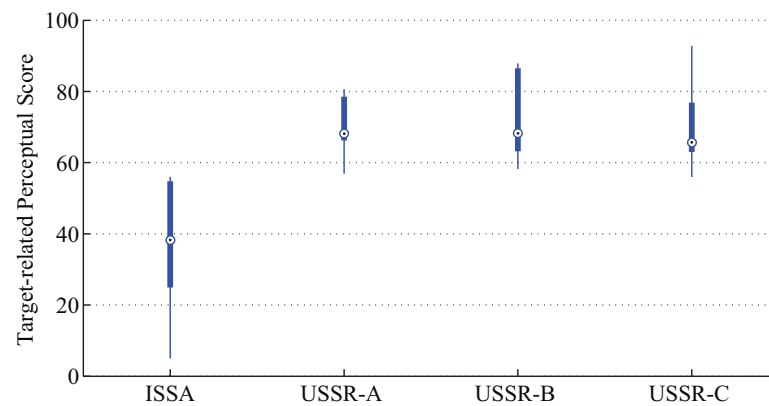
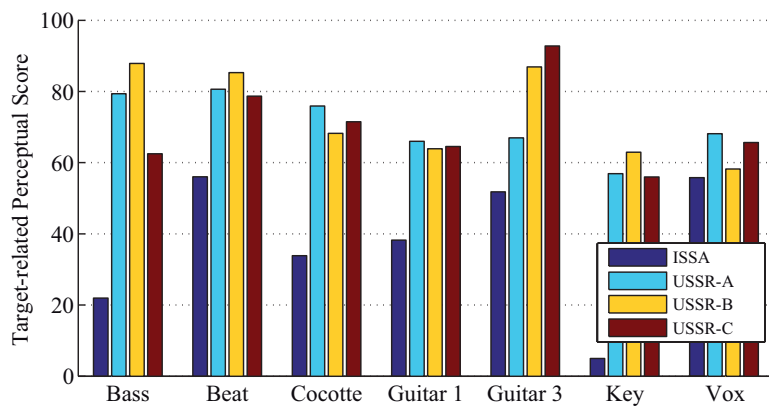
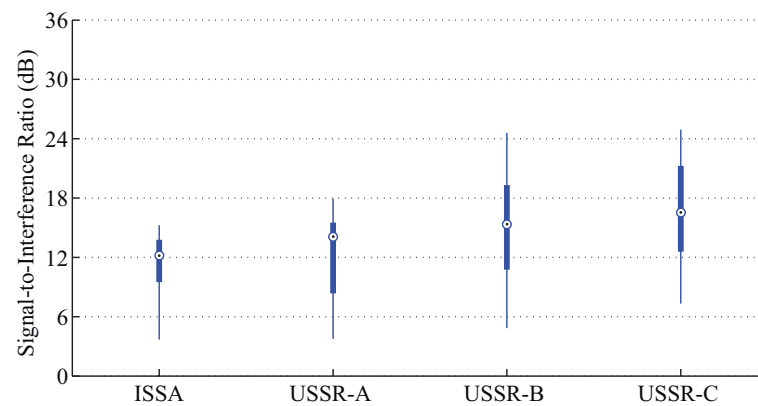
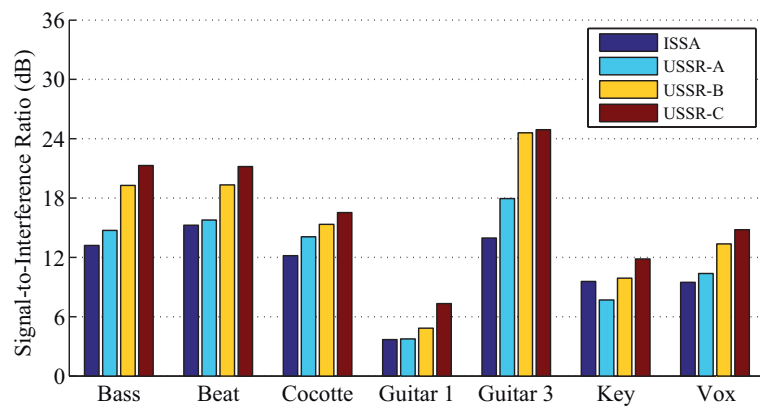


Figure 4: SIR and TPS values (left column) complemented by the median, the 25th and 75th percentiles, and outliers (right column) for an excerpt from "Lisztomania" by Phoenix

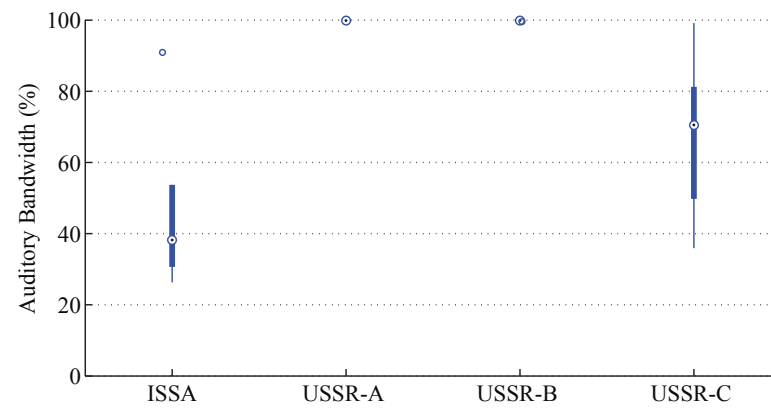
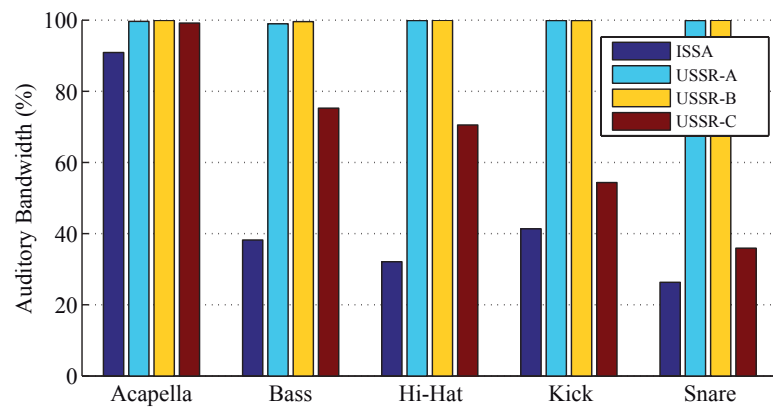
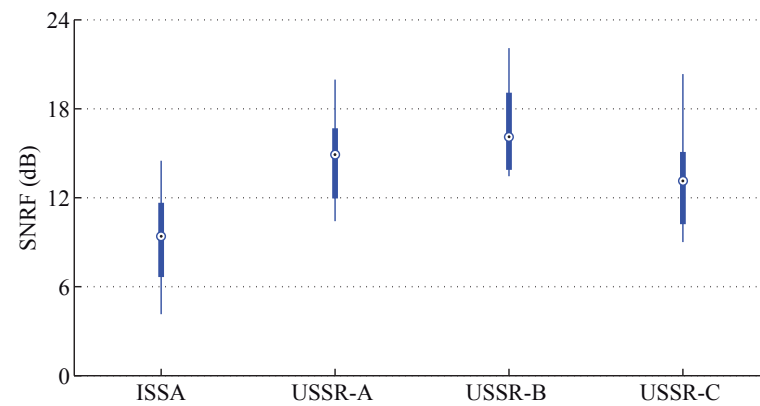
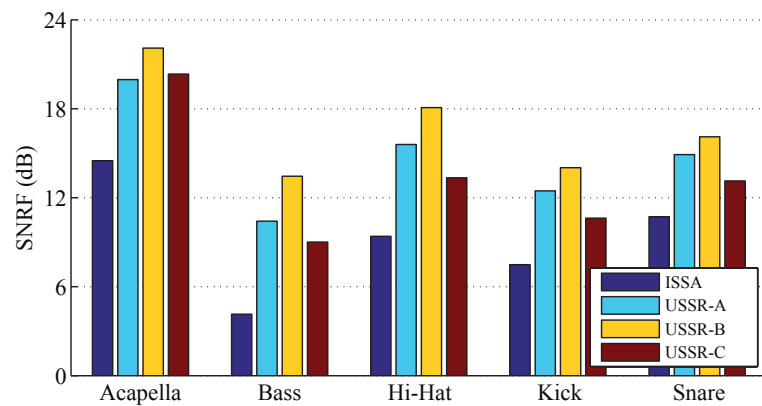


Figure 5: SNRF and auditory bandwidth values (left column) complemented by the median, the 25th and 75th percentiles, and outliers (right column) for an excerpt from “The Terrorist” by DJ Vadim

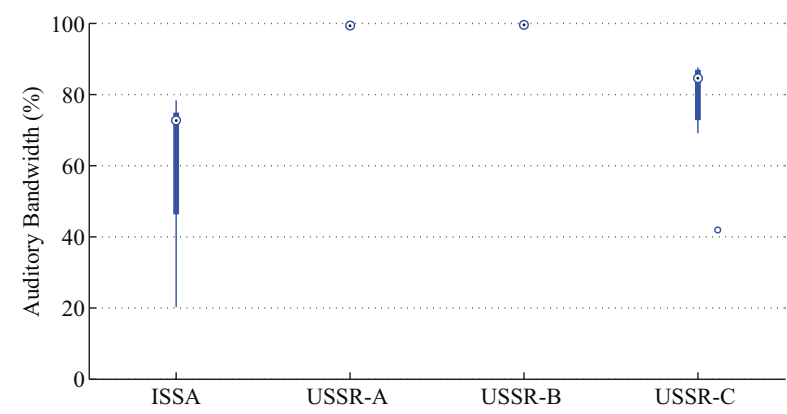
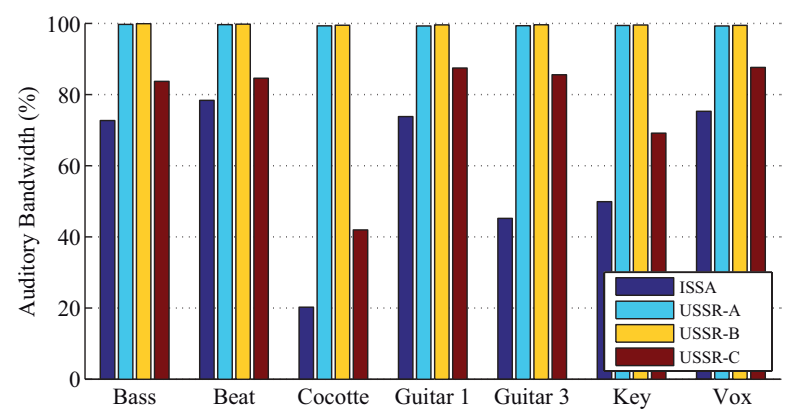
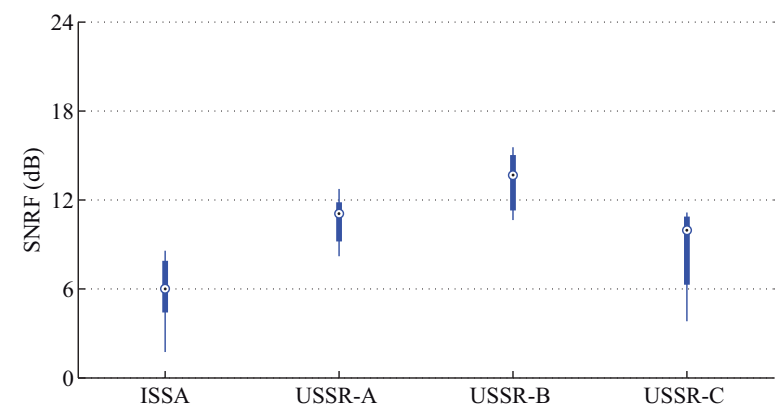
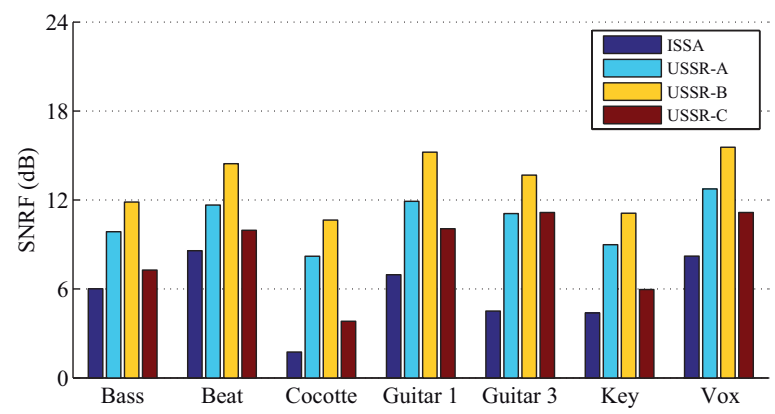


Figure 6: SNRF and auditory bandwidth values (left column) complemented by the median, the 25th and 75th percentiles, and outliers (right column) for an excerpt from “Lisztomania” by Phoenix

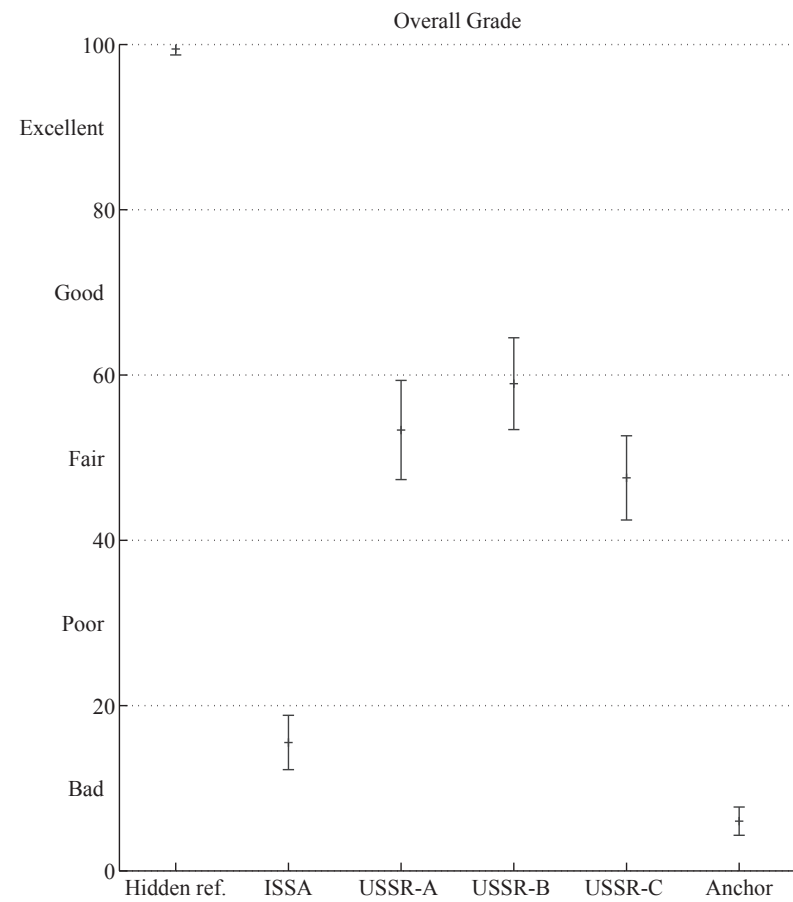
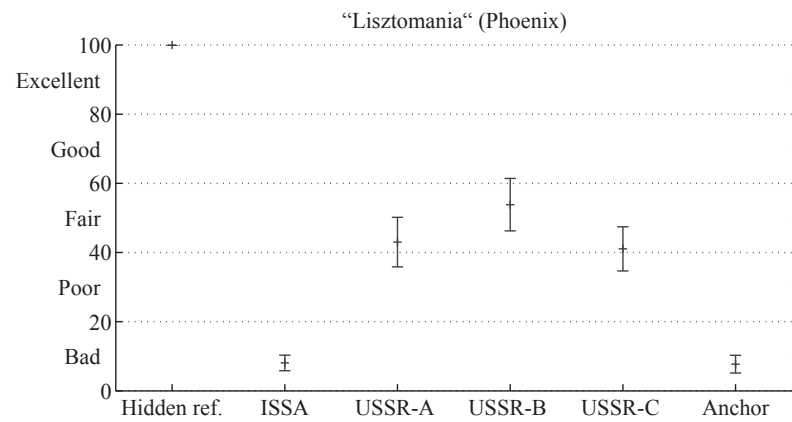
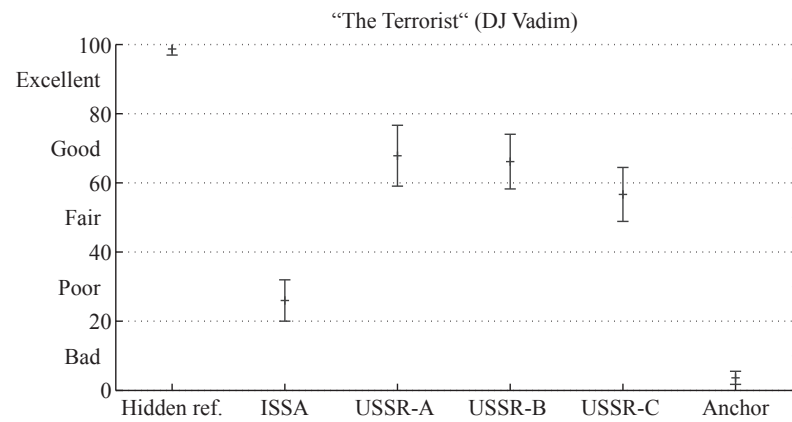


Figure 7: Mean opinion scores and 95% confidence intervals for the two music excerpts (left column) and the overall grades for the algorithms under test (right column)

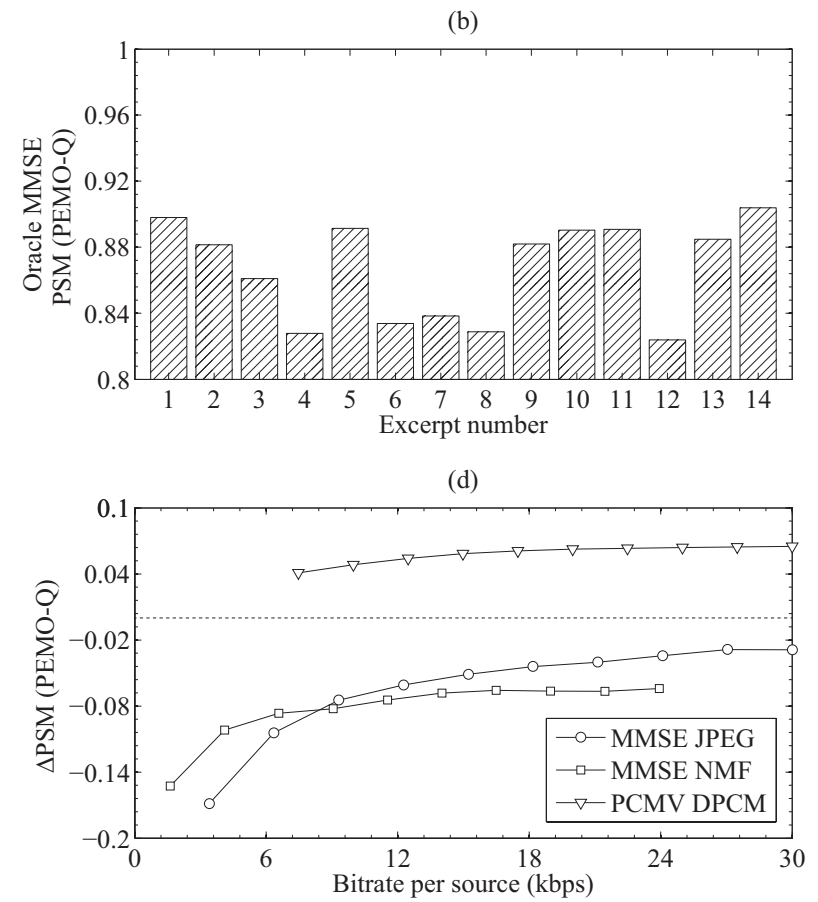
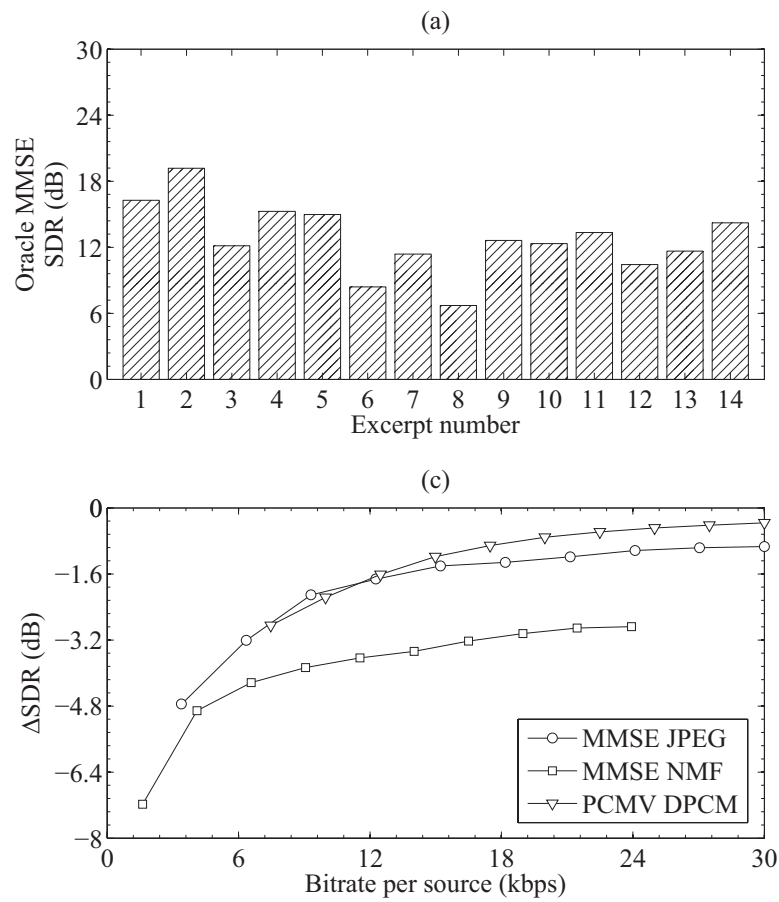


Figure 8: **SDR** and **PSM** values for the oracle **MMSE** spatial filter (upper row) and the average rate-performance difference curves for three different filtering and/or coding strategies (lower row)

| TRACK   | SDR  | ISR  | SIR  | SAR  | OPS  | TPS  | IPS  | APS  | PSM <sub>t</sub> | ODG   |
|---|------|------|------|------|------|------|------|------|------------------|-------|
| Vocal   | 9.76 | 16.8 | 11.5 | 21.7 | 0.38 | 0.61 | 0.68 | 0.79 | 0.76             | -2.96 |
| Drums   | 8.72 | 12.4 | 13.3 | 19.5 | 0.25 | 0.86 | 0.66 | 0.05 | 0.34             | -3.30 |
| Guitar  | 9.26 | 16.3 | 10.1 | 23.4 | 0.34 | 0.52 | 0.47 | 0.67 | 0.76             | -2.97 |
| "The Ones We Love" by Another Dreamer   59.6 kbps   10.6 sGHz |      |      |      |      |      |      |      |      |                  |       |
| Vocal   | 8.35 | 17.1 | 9.31 | 20.9 | 0.19 | 0.54 | 0.62 | 0.86 | 0.74             | -3.00 |
| Bass  | 8.60 | 24.2 | 8.82 | 27.7 | 0.38 | 0.62 | 0.52 | 0.34 | 0.54             | -3.21 |
| Piano   | 3.11 | 6.92 | 4.14 | 17.4 | 0.44 | 0.63 | 0.51 | 0.60 | 0.80             | -2.88 |
| Background  | 4.74 | 8.33 | 8.17 | 18.1 | 0.47 | 0.60 | 0.58 | 0.59 | 0.69             | -3.07 |
| "Roads" by Bearlin   69.8 kbps   7.4 sGHz                     |      |      |      |      |      |      |      |      |                  |       |
| Vocal   | 9.15 | 15.5 | 10.8 | 19.5 | 0.76 | 0.62 | 0.86 | 0.68 | 0.81             | -2.82 |
| Drums   | 5.15 | 6.66 | 7.07 | 15.2 | 0.27 | 0.79 | 0.64 | 0.10 | 0.40             | -3.28 |
| Bass  | 5.59 | 18.5 | 5.24 | 21.6 | 0.30 | 0.80 | 0.47 | 0.07 | -0.10            | -3.38 |
| Claps   | 8.92 | 13.8 | 11.9 | 20.6 | 0.05 | 0.96 | 0.67 | 0.00 | -0.03            | -3.37 |
| Background  | 4.76 | 10.6 | 5.80 | 14.9 | 0.46 | 0.62 | 0.51 | 0.60 | 0.72             | -3.03 |
| "Remember the Name" by Fort Minor   82.2 kbps   13.0 sGHz     |      |      |      |      |      |      |      |      |                  |       |

*Continued on next page...*

*Continued from previous page...*

| TRACK   | SDR  | ISR  | SIR  | SAR  | OPS  | TPS  | IPS  | APS  | PSM <sub>t</sub> | ODG   |
|---|------|------|------|------|------|------|------|------|------------------|-------|
| Vocal   | 14.5 | 23.0 | 15.9 | 27.6 | 0.53 | 0.56 | 0.88 | 0.87 | 0.85             | -2.66 |
| Guitar  | 14.8 | 17.4 | 20.5 | 27.1 | 0.56 | 0.98 | 0.77 | 0.81 | 0.88             | -2.53 |
| "Que Pena/Tanto Faz" by Tamy   31.8 kbps   5.8 sGHz |      |      |      |      |      |      |      |      |                  |       |
| Vocal   | 6.77 | 14.5 | 7.48 | 20.2 | 0.63 | 0.72 | 0.77 | 0.56 | 0.76             | -2.96 |
| Drums   | 8.39 | 14.6 | 10.2 | 19.9 | 0.49 | 0.82 | 0.66 | 0.34 | 0.53             | -3.22 |
| Bass  | 5.22 | 11.9 | 5.87 | 16.3 | 0.32 | 0.53 | 0.52 | 0.30 | 0.24             | -3.32 |
| Background  | 4.61 | 11.8 | 4.79 | 17.7 | 0.40 | 0.61 | 0.59 | 0.70 | 0.77             | -2.94 |
| Ultimate NZ Tour   80.7 kbps   8.4 sGHz             |      |      |      |      |      |      |      |      |                  |       |

Table 4: Experimental results. The *ISR*, *SAR*, and *TPS* values for mono sources are framed.



| TRACK                                 | $\Delta$ SDR | $\Delta$ ISR | $\Delta$ SIR | $\Delta$ SAR | $\Delta$ OPS | $\Delta$ TPS | $\Delta$ IPS | $\Delta$ APS |
|---------------------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Vocal                                 | <b>1.26</b>  | <b>1.30</b>  | -6.90        | <b>12.7</b>  | <b>0.08</b>  | -0.04        | -0.09        | <b>0.56</b>  |
|                                       | <b>3.06</b>  | <b>4.90</b>  | -3.40        | <b>14.2</b>  | <b>0.13</b>  | <b>0.19</b>  | -0.07        | <b>0.59</b>  |
| Drums                                 | -0.08        | -4.70        | -6.80        | <b>10.6</b>  | <b>0.06</b>  | <b>0.26</b>  | -0.11        | -0.11        |
|                                       | <b>7.72</b>  | <b>11.0</b>  | -4.10        | <b>16.7</b>  | <b>0.09</b>  | <b>0.59</b>  | -0.05        | -0.05        |
| Guitar                                | <b>0.76</b>  | -1.50        | -5.60        | <b>14.5</b>  | <b>0.07</b>  | <b>0.37</b>  | -0.33        | <b>0.41</b>  |
|                                       | <b>4.76</b>  | <b>9.30</b>  | -1.30        | <b>18.1</b>  | <b>0.11</b>  | <b>0.47</b>  | -0.38        | <b>0.56</b>  |
| "The Ones We Love" by Another Dreamer |              |              |              |              |              |              |              |              |
| Vocal                                 | <b>1.95</b>  | -7.20        | -15.7        | <b>12.5</b>  | <b>0.38</b>  | -0.18        | <b>0.09</b>  | <b>0.58</b>  |
|                                       | <b>3.55</b>  | -0.60        | -11.8        | <b>14.3</b>  | <b>0.53</b>  | -0.14        | <b>0.11</b>  | -0.16        |
| Drums                                 | <b>0.65</b>  | -2.64        | -6.33        | <b>11.0</b>  | <b>0.04</b>  | <b>0.04</b>  | -0.17        | -0.40        |
|                                       | <b>2.95</b>  | <b>3.36</b>  | -7.03        | <b>13.4</b>  | <b>0.08</b>  | <b>0.42</b>  | -0.09        | <b>0.09</b>  |
| Bass                                  | -1.51        | -0.80        | -5.76        | <b>12.4</b>  | -0.27        | <b>0.14</b>  | -0.45        | <b>0.07</b>  |
|                                       | <b>5.19</b>  | <b>18.0</b>  | -4.66        | <b>22.4</b>  | -0.15        | <b>0.74</b>  | -0.35        | <b>0.07</b>  |
| Claps                                 | <b>1.32</b>  | -5.80        | -17.5        | <b>13.1</b>  | -0.69        | <b>0.18</b>  | -0.22        | 0.00         |
|                                       | <b>1.02</b>  | -3.90        | -14.1        | <b>12.6</b>  | -0.15        | <b>0.34</b>  | -0.09        | -0.19        |
| "Remember the Name" by Fort Minor     |              |              |              |              |              |              |              |              |

*Continued on next page...*

*Continued from previous page...*

| TRACK            | $\Delta$ SDR | $\Delta$ ISR | $\Delta$ SIR | $\Delta$ SAR | $\Delta$ OPS | $\Delta$ TPS | $\Delta$ IPS | $\Delta$ APS |
|------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Vocal            | <b>2.57</b>  | -7.60        | -14.7        | <b>16.2</b>  | <b>0.31</b>  | <b>0.13</b>  | <b>0.05</b>  | <b>0.25</b>  |
|                  | <b>4.17</b>  | -3.50        | -10.0        | <b>18.2</b>  | <b>0.38</b>  | <b>0.45</b>  | <b>0.07</b>  | <b>0.37</b>  |
| Drums            | <b>2.19</b>  | -8.90        | -12.2        | <b>13.7</b>  | <b>0.23</b>  | -0.04        | -0.03        | <b>0.32</b>  |
|                  | <b>6.09</b>  | <b>9.60</b>  | -8.40        | <b>17.7</b>  | <b>0.25</b>  | <b>0.58</b>  | <b>0.01</b>  | <b>0.30</b>  |
| Bass             | <b>2.52</b>  | -4.60        | -14.8        | <b>14.3</b>  | <b>0.03</b>  | <b>0.17</b>  | -0.30        | <b>0.30</b>  |
|                  | <b>3.82</b>  | <b>8.90</b>  | -11.4        | <b>17.7</b>  | -0.02        | <b>0.48</b>  | -0.28        | <b>0.29</b>  |
| Ultimate NZ Tour |              |              |              |              |              |              |              |              |

Table 5: A comparison between the results shown in Table 4 and the scores reported in [SiSEC 2011](#) for two oracle systems. The upper row next to the track name represents the [STFT](#)-based system and the lower row represents the cochleagram-based system, respectively.

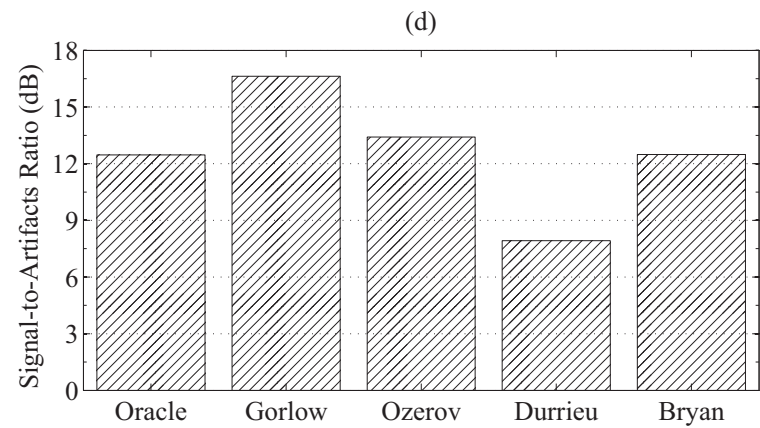
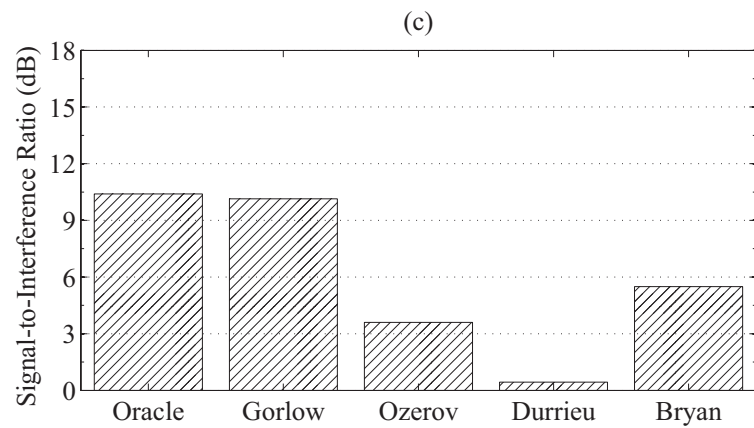
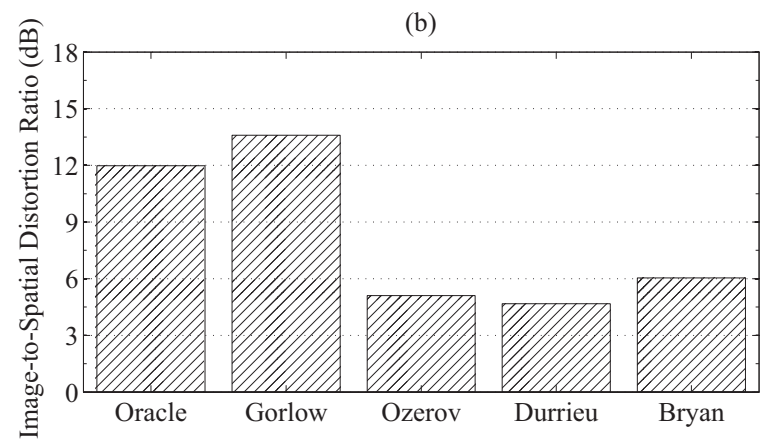
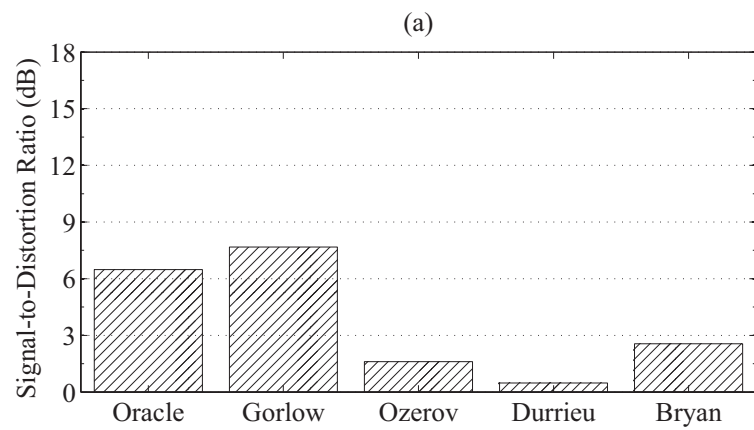


Figure 9: Mean SDR, ISR, SIR, and SAR values for the SiSEC 2013 development dataset consisting of three music excerpts

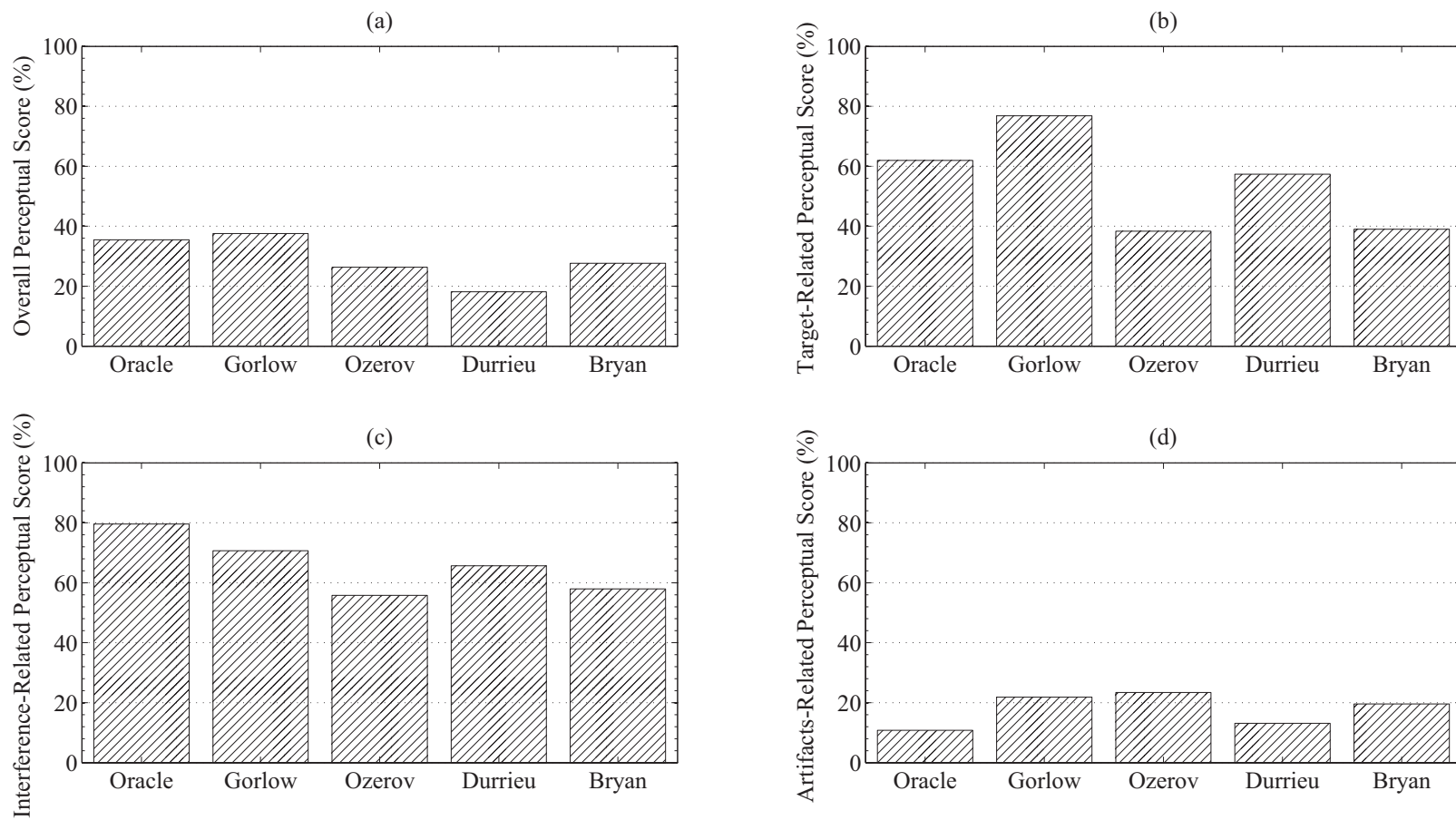


Figure 10: Mean OPS, TPS, IPS, and APS values for the SiSEC 2013 development dataset consisting of three music excerpts

## 8.1 ENHANCED AUDIO OBJECT SEPARATION

SAOC's own source separation scheme is referred to as "enhanced audio object separation" (EAOS), whereas the object decoder is also called an "enhanced audio object processor" [75, Fig. 6]. The latter consists of a two-to-I upmix unit [75, Fig. 5] which corresponds to I two-to-three (TTT) basic units [75, Fig. 4]. A TTT unit takes a two-channel mixture as input and outputs a mono signal from the object of interest, the foreground object (FGO), together with the background signal being the sum of all pan-potted object signals excluding the FGO signal. Every object that contributes to the background signal is further deemed a background object (BGO). A BGO is unalterable while an FGO can be altered in regard to its volume level and location. In applications where a single track is soloed or muted, like karaoke or mix-minus, an FGO is upgraded to a so-called "enhanced audio object" (EAO). An EAO signal is tantamount to an FGO signal that is error corrected and so quality improved using the residual, i.e. at the cost of a higher side-information rate.

## 8.2 OBJECT ENCODER

EAOS processes the object signals in the complex subband domain. The subband analysis is based on a hybrid filter bank that splits the time signal into 69 subband signals [76]. The STPSDs are computed as the instantaneous powers in each subband. These are quantized on a logarithmic scale and grouped over time and frequency. Finally, this metadata is DPCM coded and passed on to the decoder as side information along with the mixing coefficients and the downmix. The corresponding system model in the DFT domain is

$$\mathbf{X}^H = \sum_{i=1}^I \mathbf{a}_i \mathbf{s}_i^H = \mathbf{A} \mathbf{S}^H, \quad (57)$$

where  $\mathbf{s}_i \in \mathbb{C}^K$  is a K-band signal vector,  $\mathbf{a}_i \in \mathbb{R}^2$  is the mixing vector, I is the number of audio objects, and  $\mathbf{X} \in \mathbb{C}^{K \times 2}$  is the mixture signal. Superscript H denotes Hermitian transpose. The I signal vectors and the I mixing vectors can be concatenated into the signal matrix  $\mathbf{S} \in \mathbb{C}^{K \times I}$  and the mixing matrix  $\mathbf{A} \in \mathbb{R}^{2 \times I}$ , such that the mixture matrix can be expressed as a product of the two matrices. The calculation of

a single residual is as follows. First, all BGO signals are combined into a downmix signal [32, Eqs. 15–16],

$$\mathbf{X}_{\text{BGO}}^{\text{H}} = \mathbf{X}^{\text{H}} - \mathbf{a}_{\text{FGO}} \mathbf{s}_{\text{FGO}}^{\text{H}}. \quad (58)$$

Second, an “auxiliary” signal for the FGO is calculated by taking the difference between the FGO and the BGO mixture signal projected onto the “look direction” of the FGO [32, Eqs. 15–16],

$$\mathbf{s}_{\text{FGO}_o} = \mathbf{X}_{\text{BGO}} \mathbf{a}_{\text{FGO}} - \mathbf{s}_{\text{FGO}}. \quad (59)$$

Using (58) and (10), (59) can also be written as

$$\mathbf{s}_{\text{FGO}_o} = \mathbf{X} \mathbf{a}_{\text{FGO}} - 2 \mathbf{s}_{\text{FGO}}. \quad (60)$$

Then, a linear combination of the (two) downmix channels is found that minimizes the reconstruction error between the modeled signal and the true auxiliary signal. For this, a system of  $K$  linear equations in two unknowns  $\mathbf{w}_{\text{FGO}_o} = [w_{1,\text{FGO}_o} \ w_{2,\text{FGO}_o}]^{\text{T}} \in \mathbb{R}^2$  must be solved:

$$\mathbf{X} \mathbf{w}_{\text{FGO}_o} = \mathbf{s}_{\text{FGO}_o}. \quad (61)$$

The two coefficients that best fit the above equations in the LS sense are

$$\hat{\mathbf{w}}_{\text{FGO}_o}^{\text{LS}} = \underbrace{(\mathbf{X}^{\text{H}} \mathbf{X})^{-1}}_{\hat{\mathbf{R}}_x} \underbrace{\mathbf{X}^{\text{H}} \mathbf{s}_{\text{FGO}_o}}_{\hat{\mathbf{p}}_{x\text{s}_{\text{FGO}_o}}}. \quad (62)$$

The terms  $\mathbf{X}^{\text{H}} \mathbf{X}$  and  $\mathbf{X}^{\text{H}} \mathbf{s}_{\text{FGO}_o}$  are equivalent to the sample estimates of spatial correlation,  $\hat{\mathbf{R}}_x$  and  $\hat{\mathbf{p}}_{x\text{s}_{\text{FGO}_o}}$ . For that reason, the LS estimate for  $\mathbf{w}_{\text{FGO}_o}$  is formally identical with the MMSE estimator for the auxiliary signal. Due to (60),

$$\hat{\mathbf{p}}_{x\text{s}_{\text{FGO}_o}} = \hat{\mathbf{R}}_x \mathbf{a}_{\text{FGO}} - 2 \hat{\mathbf{p}}_{x\text{s}_{\text{FGO}}}. \quad (63)$$

So, by rewriting (62) as

$$\hat{\mathbf{w}}_{\text{FGO}_o}^{\text{MMSE}} = \mathbf{a}_{\text{FGO}} - \underbrace{2 \hat{\mathbf{R}}_x^{-1} \hat{\mathbf{p}}_{x\text{s}_{\text{FGO}}}}_{\hat{\mathbf{w}}_{\text{FGO}}^{\text{MMSE}}}, \quad (64)$$

the estimator can be put in direct relation to the FGO. Note that  $\mathbf{p}_{x\text{s}_{\text{FGO}}}$  is the cross-correlation between the mixture and the FGO signal. The difference between the true auxiliary signal and its estimate,

$$\mathbf{r}_{\text{FGO}_o} = \mathbf{s}_{\text{FGO}_o} - \mathbf{X} \hat{\mathbf{w}}_{\text{FGO}_o}^{\text{MMSE}}, \quad (65)$$

yields the residual that is perceptual-entropy coded by the Advanced Audio Coding (AAC) [77] scheme at a bitrate of 20 kbps [75].

### 8.3 OBJECT DECODER

To obtain the FGO signal, the auxiliary signal needs to be estimated first. To that end, the estimator from (64) is computed using the power spectra and the mixing coefficients. Both are made available for the decoder by the encoder. The correlation matrix is calculated as in (43). Correspondingly, the cross-correlation vector is given by

$$\hat{\mathbf{p}}_{\mathbf{x}s_{\text{FGO}}} = \mathbf{a}_{\text{FGO}}\Phi_{s_{\text{FGO}}}. \quad (66)$$

Again, the audio objects are considered mutually uncorrelated. Note that in SAOC, the cross-correlations between objects can be modeled using the inter-object cross coherences (IOCs). Yet the computation of these is optional, see Section 7.6.4. Once the estimator is computed it is plugged into (61), and the enhanced auxiliary signal is obtained by adding the residual to the estimate:

$$\tilde{\mathbf{s}}_{\text{EAO}_o} = \underbrace{\mathbf{X}\hat{\mathbf{w}}_{\text{FGO}_o}^{\text{MMSE}}}_{\hat{\mathbf{s}}_{\text{FGO}_o}} + \tilde{\mathbf{r}}_{\text{FGO}_o}. \quad (67)$$

Solving (59) for  $\mathbf{s}_{\text{FGO}}$  using (58) yields the sought-after EAO signal

$$\tilde{\mathbf{s}}_{\text{EAO}} = \frac{1}{2}(\mathbf{X}\mathbf{a}_{\text{FGO}} - \tilde{\mathbf{s}}_{\text{EAO}_o}). \quad (68)$$

Using (67) and (64), (68) can also be formulated as

$$\tilde{\mathbf{s}}_{\text{EAO}} = \mathbf{X}\hat{\mathbf{w}}_{\text{FGO}}^{\text{MMSE}} - \frac{1}{2}\tilde{\mathbf{r}}_{\text{FGO}_o}. \quad (69)$$

The bottom line is that the TTT unit in SAOC is an MMSE estimator for the FGO signal with a particular residual coding strategy. This being the case, it fits perfectly into the ISS framework.

## 8.4 PERFORMANCE EVALUATION

### 8.4.1 EAOS vs. USSR

In the previous section it is shown that EAOS in SAOC uses a Bayes estimator in the form of an MMSE spatial filter to separate audio objects from their mixture. In this section, EAOS is compared with USSR using the same ISS testing framework.

#### 8.4.1.1 Experimental setup

The STFT with a KBD window and 50-% overlap between segments is used for both systems under test. The PCMV spatial filter in USSR is replaced by the TTT unit when simulating EAOS. The metadata is encoded as in Section 7.7. The mixing coefficients are considered to be known. The two systems are compared with each other in terms of quality and computational complexity. The quality is assessed on

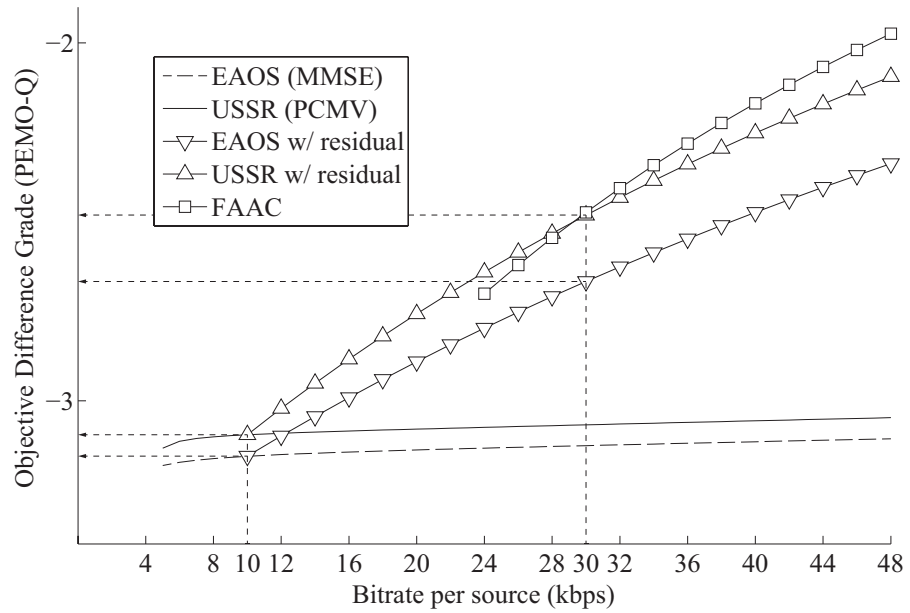


Figure 11: ODG as a function of the side-information rate

the ODG scale [78] while the complexity is measured by the decoder's execution time in MATLAB. The ODG score is computed with PEMO-Q. Ten multitracks taken from the QUASI database [79] are converted to mono and cut down to 20-s excerpts. Each track is normalized to a reference root mean square (RMS) level of  $-16$  dBFS. The sources are placed uniformly in space and gain adjusted, so as to have an equal SIR across the sources at the output, see Algorithm 4.

#### 8.4.1.2 Experimental results

The results of the comparison are shown in Figs. 11–12. In the case where the residual is omitted, the bitrate is equivalent to the metadata rate for a varied number of parameter bands. In the case where the residual is used to correct the initial estimate, the metadata rate is fixed at 10 kbps and the bitrate is calculated as the sum of the latter and the residual rate increased from zero onwards. The results where the original signals are coded separately are also included. There, the bitrate is simply the coding rate.

The Freeware Advanced Audio Coder (FAAC) [80] is applied to the residual and the original tracks. The curves represent fitted averages over the data corpus. As can be seen from Fig. 11, USSR's PCMV spatial filter yields better results than EAOS's MMSE filter. This observation is consistent with the listening test results reported in Section 7.8.1. The gap between the two systems widens even further if their estimates are error corrected. At a bitrate of 30 kbps per source and beyond, the quality of FAAC-coded tracks is superior to those recovered from the mixture.



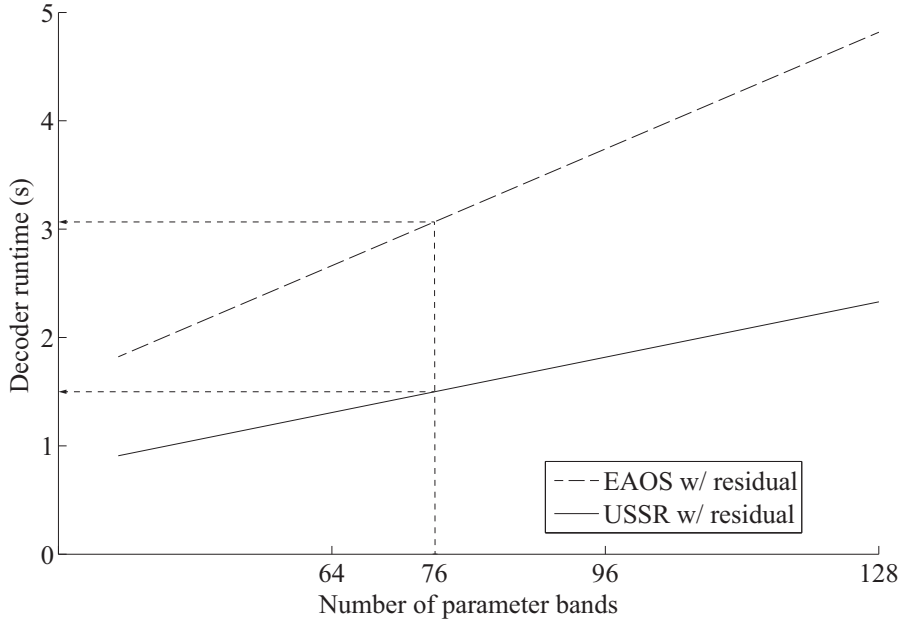


Figure 12: Decoder runtime as a function of the number of bands at a side-information rate of 30 kbps

As can be seen in Fig. 12, the **USSR** decoder is approximately half as complex as the **EAOS** decoder if all I estimates are error corrected. The longer runtime is explained by the fact that **EAOS** requires the residuals to be available in the subband domain whereas **USSR** does not. Hence, the loss of time is due to the extra I-fold **STFT** and the computation of auxiliary signals.

#### 8.4.2 Interactive Remixing

In this section the **ISS** approach is evaluated in a concrete scenario. “Interactive remixing” is chosen as sample application. It allows the user to change the volume level of sound sources and their spatial location. Also, it is interesting to find out whether “plain” source coding is a more pragmatic solution with regard to audio quality and coding efficiency.

##### 8.4.2.1 Experimental setup

The same testing framework as before is used. The **ODG** is retained as quality index. In order to simulate more realistic conditions, the stereo-to-mono converted tracks are panned to their original location, so that the downmix is prearranged as if by the sound engineer or the composer. The location is derived from the **RMS** channel difference in the original stereo tracks (see also Appendix B):

$$\hat{\alpha}_i = \operatorname{arccot} \frac{\operatorname{RMS}_{2i}}{\operatorname{RMS}_{1i}}, \quad (70)$$

where  $\text{arccot}$  is the arccotangent.

Table 6 provides a listing of the songs used in the experiment. The source signals are either recovered from the downmix using [USSR](#) or are encoded (and decoded) for comparison. For this, the [FAAC](#) and the Enhanced aacPlus [81, 82] coder are used. The downmix is encoded in perceptually transparent quality: at a variable bitrate in the region of 120 kbps. To simulate user interaction, ten different remixes with arbitrary new source locations and volume levels for each song and system are generated with the gains being in the range between  $-6$  and  $3$  dB. The remixes of each system are then compared with the ones created from the original tracks.

#### 8.4.2.2 *Experimental results*

The evaluation results are summarized in Fig. 13. It can be observed that the quality of a remix generated from an [FAAC](#) coded downmix depends on the number of sources and their spatial spread. A linearly pulse-code modulated ([LPCM](#)) mixture signal seems less sensitive to these factors. The deciding factor there is the “spectral texture” of a source signal and by how much it interweaves with other sources.

On an average, the best quality is obtained for an [LPCM](#) mixture in combination with residual coding at ca. 20 kbps per source. With a median not worse than “slightly annoying”, the results gained with [USSR](#) alone are promising. Clearly worse is the grade for the scenario in which the mixture is [FAAC](#) coded. Even at a side-information rate of 30 kbps, the quality lies halfway between “annoying” and “slightly annoying”. The same is true for Enhanced aacPlus at 10 kbps or [FAAC](#) at 30 kbps. The most efficient system in the experiment is Enhanced aacPlus operating at 30 kbps, as it does not necessitate availability of the mixture.

| NO. | TITLE                                     | NUMBER OF SOURCES | SPATIAL SPREAD<br>VS. CENTROID |
|-----|---|-------------------|--------------------------------|
| 1   | "Carol of the Bells" (Alexq)              | 4                 | 15.9° / 40.3°                  |
| 2   | "One We Love" (Another Dreamer)           | 5                 | 20.8° / 47.4°                  |
| 3   | "The World Is Under Attack" (Carl Leth)   | 6                 | 5.76° / 47.2°                  |
| 4   | "Remember the Name" (Fort Minor)          | 10                | 10.4° / 46.4°                  |
| 5   | "The Spirit of Shackleton" (Glen Philips) | 12                | 7.54° / 47.1°                  |
| 6   | "Mix Tape" (Jim's Big Ego)                | 7                 | 2.00° / 45.0°                  |
| 7   | "Good Soldier" (Nine Inch Nails)          | 5                 | 1.72° / 43.8°                  |
| 8   | "Sunrise" (Shannon Hurley)                | 8                 | 8.51° / 41.0°                  |
| 9   | "Untitled" (Ultimate NZ Tour)             | 7                 | 12.3° / 45.3°                  |
| 10  | "Ana" (Vieux Farka)                       | 8                 | 9.54° / 42.6°                  |

Table 6: The corpus of prearranged mixes

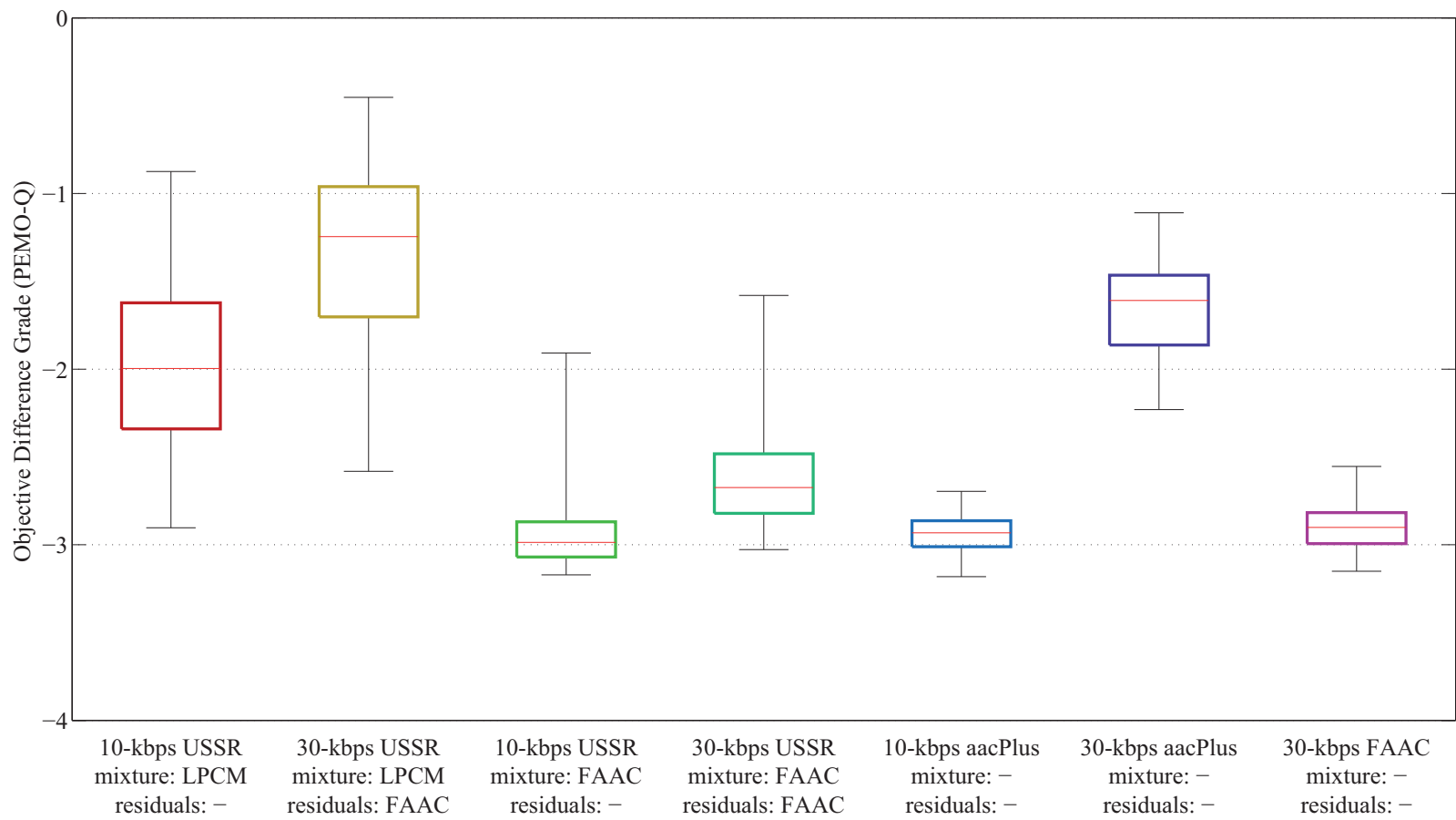


Figure 13: The medians and the 25th and 75th percentiles for each system under test

INVERSION OF DYNAMIC RANGE COMPRESSION

---

## 9.1 DYNAMIC RANGE COMPRESSION

Dynamic range compression or simply *compression* is an audio processing technique that attenuates loud sounds and/or amplifies quiet sounds, which in consequence leads to a reduction of an audio signal's dynamic range. The latter is defined as the difference between the loudest and the quietest sound measured in decibels. In the following, we will use the word "compression" having "downward" compression in mind, though the discussed approach is likewise applicable to "upward" compression. Downward compressing means attenuating sounds above a certain threshold while leaving sounds below the threshold unchanged. An audio engineer might use a compressor to reduce the dynamic range of source material for purposes of aesthetics, intelligibility, recording or broadcast limitations, etc.

Fig. 14 shows the *basic* compressor model from [83, ch. 2] amended by a switchable RMS/peak detector in the side chain, which makes it compatible with the compressor/limiter model from [15, p. 106]. We will hereafter restrict our considerations to this basic model, as the purpose of the present work is to demonstrate a general approach rather than a solution to a specific problem. First, the input signal is split and a copy is sent to the side chain. The detector then calculates the magnitude or level of the sidechain signal using the RMS or peak as a measure for how loud a sound is [15, p. 107]. The detector's temporal behavior is controlled by the attack and release parameters. The sound level is compared with the threshold level and, for the case it exceeds the threshold, a scale factor is calculated which corresponds to the ratio of input level to output level. The knee parameter determines how quick the compression ratio is reached. At the end of the side chain, the scale factor is fed to a smoothing filter that yields the gain. The response of the gain filter is controlled by another set of attack and release parameters. Finally, the gain control applies the smoothed gain to the input signal and adds a fixed amount of makeup gain to bring the output signal to a desired level. Such a broadband compressor operates on the input signal's full bandwidth, treating all frequencies from zero through the highest frequency equally. A detailed overview of all sidechain controls of a basic gain computer is given in [83, ch. 3], e.g.

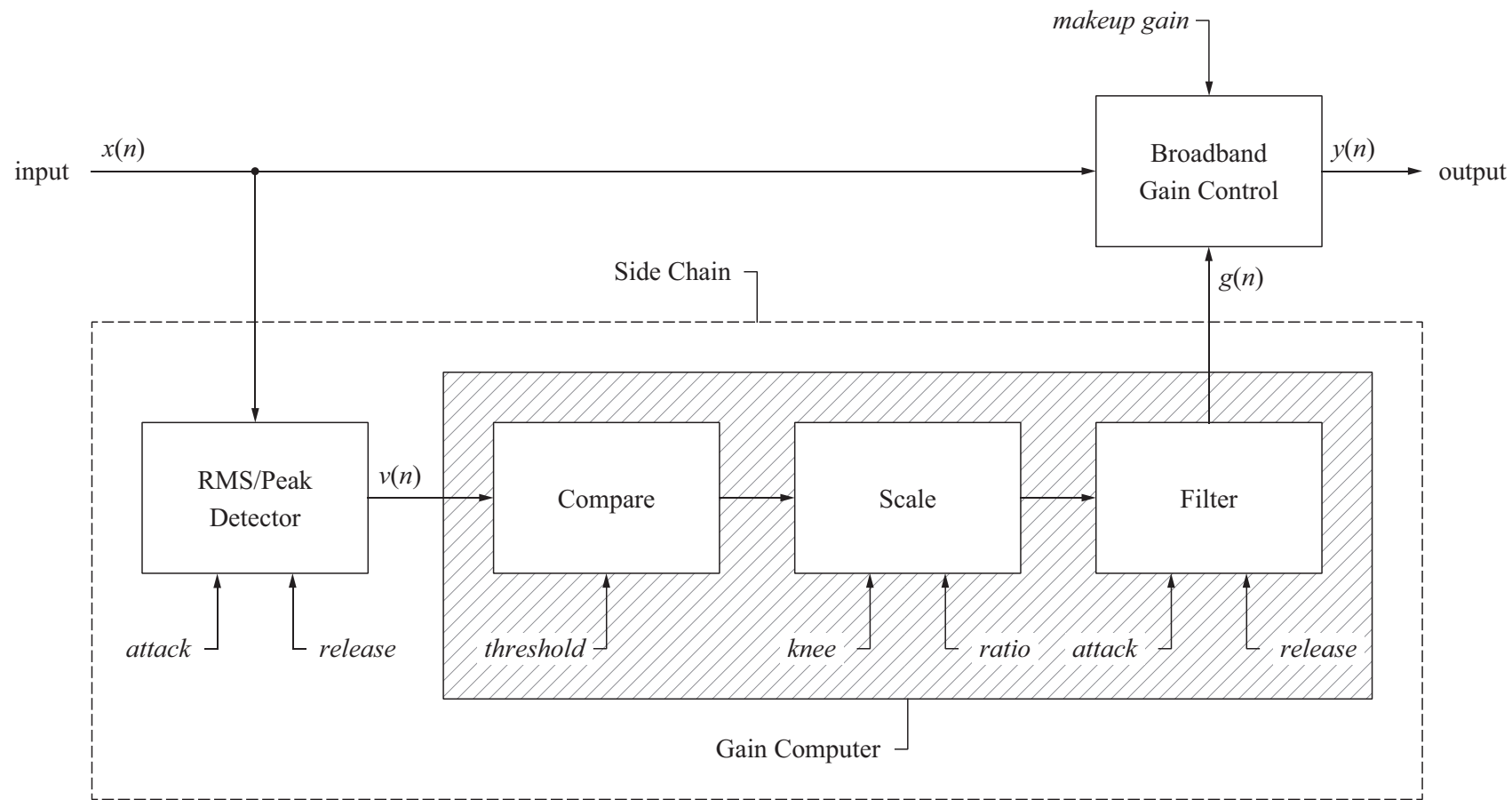


Figure 14: Basic broadband compressor model

## 9.2 SYSTEM MODEL

## 9.2.1 Feed-Forward Broadband Compression

The employed system model is based on the compressor from Fig. 14. The following simplification is additionally made: the knee parameter is ignored, i.e. the knee is “hard”. The compressor is understood as a single-input single-output (SISO) system, that is both the input and the output are single-channel signals. What follows is a description of each block by means of a dedicated function.

The RMS/peak detector as well as the gain computer build upon a first-order (one-pole) lowpass filter. The sound level or envelope  $v(n)$  of the input signal  $x(n)$  is obtained by

$$\tilde{x}(n) = \alpha_v |x(n)|^p + (1 - \alpha_v) \tilde{x}(n-1), \quad (71a)$$

$$v(n) = \sqrt[p]{\tilde{x}(n)}, \quad (71b)$$

where  $p = 1$  represents a peak detector and  $p = 2$  an RMS detector, respectively. The smoothing factor  $\alpha_v$ ,  $0 < \alpha_v \leq 1$ , may take on two different values,  $\alpha_v^{\text{att}}$  or  $\alpha_v^{\text{rel}}$ , depending on whether the detector is in the attack or the release phase. The condition for the level detector to enter the attack phase and to choose  $\alpha_v^{\text{att}}$  over  $\alpha_v^{\text{rel}}$  is

$$|x(n)| > v(n-1). \quad (72)$$

A formula that converts a time constant  $\tau$  into a smoothing factor is given in [15, p. 109]. So,

$$\alpha_v = 1 - \exp\left(-2.2 \frac{T_s}{\tau_v}\right), \quad (73)$$

where  $T_s$  is the sampling period and  $\exp$  the exponential function. The derivation of (73) can be found in [84, 85]. The static nonlinearity in the gain computer is usually modeled in the logarithmic domain as a continuous piecewise linear function:

$$F(n) = \begin{cases} -S \cdot [V(n) - L] & \text{if } V(n) > L, \\ 0 & \text{otherwise,} \end{cases} \quad (74)$$

where  $L$  is the threshold in decibel,  $V(n) = 20 \log_{10} v(n)$ , and  $S$  is the slope. The slope is computed from the desired compression ratio  $R$  according to

$$S = 1 - \frac{1}{R}. \quad (75)$$

Equivalently, (74) can be expressed in the linear domain as

$$f(n) = \begin{cases} \kappa v^{-S}(n) & \text{if } v(n) > l, \\ 1 & \text{otherwise,} \end{cases} \quad (76)$$

where  $l = 10^{L/20}$ ,  $\kappa = l^S$ , and  $f$  is the scale factor before filtering. The smoothed gain  $g$  is calculated as the exponentially-weighted moving average:

$$g(n) = \alpha_g f(n) + (1 - \alpha_g)g(n-1) \quad (77)$$

with  $\alpha_g \in \{\alpha_g^{\text{att}}, \alpha_g^{\text{rel}}\}$ .

The decision to choose  $\alpha_g^{\text{att}}$  instead of  $\alpha_g^{\text{rel}}$  is subject to

$$f(n) < g(n-1). \quad (78)$$

Finally, the broadband gain control multiplies the input signal  $x(n)$  by the smoothed gain  $g(n)$  and adds some makeup gain  $M$  to bring the compressed output signal  $y(n)$  to a desired level:

$$y(n) = m \cdot [g(n)x(n)] \quad \text{with } m = 10^{M/20}. \quad (79)$$

Due to the fact that  $g$  and  $m$  are strictly positive,  $0 < g \leq 1$ ,  $m \geq 0$ , it follows that

$$\text{sgn}(y) = \text{sgn}(x), \quad (80)$$

where  $\text{sgn}$  is the signum function. In consequence, it is convenient to factorize the input signal as a product of the sign and the modulus,

$$x(n) = \text{sgn}(x) \cdot |x(n)|. \quad (81)$$

### 9.2.2 Stereo Linking

To avoid image shifting, it is imperative that an equal amount of gain reduction be applied to both channels of  $x$ . This can be achieved by calculating the required amount of gain reduction for  $x_1(n)$  and  $x_2(n)$  independently, and by applying the larger amount to *both* channels:

$$y(n) = m \cdot [g(n)x(n)], \quad (82)$$

where  $\mathbf{y} = [y_1 \ y_2]^T$  and

$$g(n) = \min[g_1(n), g_2(n)]. \quad (83)$$

## 9.3 PROBLEM FORMULATION

The problem at hand is formulated as follows. Given the compressed signal  $y(n)$  and the compressor parameters

$$\theta = \{p, L, R, \alpha_v^{\text{att}}, \alpha_v^{\text{rel}}, \alpha_g^{\text{att}}, \alpha_g^{\text{rel}}, M\}, \quad (84)$$

recover the modulus of the original signal  $|x(n)|$  from  $|y(n)|$  based on  $\theta$ . For a more intuitive use, the smoothing factors  $\alpha_v$  and  $\alpha_g$  may be replaced by the time constants  $\tau_v$  and  $\tau_g$ . The meaning of each parameter is recapitulated below.



- p Detector type (peak or RMS)  
 L Threshold level in dB  
 R Compression ratio  $\text{dB}_{\text{in}} : \text{dB}_{\text{out}}$   
 $\tau_{v,\text{att}}$  Attack time of the envelope filter in ms  
 $\tau_{v,\text{rel}}$  Release time of the envelope filter in ms  
 $\tau_{g,\text{att}}$  Attack time of the gain filter in ms  
 $\tau_{g,\text{rel}}$  Release time of the gain filter in ms  
 M Makeup gain in dB

#### 9.4 PROPOSED SOLUTION

The output of the side chain, i.e. the gain of  $|x(n)|$ , given  $\theta$ ,  $\tilde{x}(n-1)$ , and  $g(n-1)$ , may be written as

$$m \cdot g(n) = G[|x(n)| | \theta, \tilde{x}(n-1), g(n-1)]. \quad (85)$$

In (85),  $G$  denotes a *nonlinear dynamic operator* that maps the modulus of the input signal  $|x(n)|$  onto a sequence of instantaneous gain values  $m \cdot g(n)$  according to the compressor model represented by  $\theta$ . Using (85), (79) can be solved for  $|x(n)|$  yielding

$$|x(n)| = G^{-1}[g(n) | \theta, \tilde{x}(n-1), g(n-1)] \cdot |y(n)|$$

subject to invertibility of  $G$ . In order to solve the above equation one requires the knowledge of  $g(n)$ , which is unavailable. However, since  $g$  is a function of  $|x|$ , we can express  $|y|$  as a function of  $|x|$  only, and in that manner we obtain an equation with a single unknown:

$$|y(n)| = H[|x(n)| | \theta, \tilde{x}(n-1), g(n-1)], \quad (86)$$

where  $H$  now represents the entire compressor. If  $H$  is invertible, i.e. bijective for all  $n$ ,  $|x(n)|$  can be obtained from  $|y(n)|$  by

$$|x(n)| = \begin{cases} H^{-1}[|y(n)| | \theta, \tilde{x}(n-1), g(n-1)] & \text{if } v(n) > 1, \\ |y(n)| & \text{otherwise.} \end{cases} \quad (87)$$

And yet, since  $v(n)$  is likewise unknown, the condition for applying decompression must be predicted from  $y(n)$ ,  $\tilde{x}(n-1)$ , and  $g(n-1)$ , and therefore needs the condition for toggling between the attack and release phases. Depending on the quality of the prediction, the recovered modulus  $|z(n)|$  may differ somewhat at transition points from the original modulus  $|x(n)|$ , so that in the end

$$x(n) \approx \text{sgn}(y) \cdot |z(n)| \triangleq z(n). \quad (88)$$

In the following it is shown how such an *inverse compressor* alias *de-compressor* is derived.

## 9.4.1 Characteristic Function

For simplicity, the instantaneous envelope value  $v(n)$  is chosen over  $|x(n)|$  as the independent variable in (86). The relation between the two items is given by (71). From (77) and (79), when  $v(n) > 1$ ,

$$|y(n)| = m \cdot [\alpha_g f(n) + (1 - \alpha_g)g(n-1)] \cdot |x(n)| \quad (89)$$

$$\stackrel{(76)}{=} m \cdot [\alpha_g \kappa v^{-S}(n) + (1 - \alpha_g)g(n-1)] \cdot |x(n)|. \quad (90)$$

From (71),

$$|y(n)| = m \cdot [\alpha_g \kappa v^{-S}(n) + (1 - \alpha_g)g(n-1)] \cdot \sqrt[p]{\frac{v^p(n) - (1 - \alpha_v)\tilde{x}(n-1)}{\alpha_v}}, \quad (91)$$

or equivalently (note that  $m, \alpha_v \neq 0$  by definition)

$$\alpha_v \left[ \frac{|y(n)|}{m} \right]^p = [\alpha_g \kappa v^{-S}(n) + (1 - \alpha_g)g(n-1)]^p \cdot [v^p(n) - (1 - \alpha_v)\tilde{x}(n-1)]. \quad (92)$$

Moreover, (92) has a unique solution if  $G$  and hence  $H$  is invertible. Moving the expression on the left-hand side over to the right-hand side, we may define

$$\zeta_p(v) \triangleq [\alpha_g \kappa v^{-S}(n) + (1 - \alpha_g)g(n-1)]^p \cdot [v^p(n) - (1 - \alpha_v)\tilde{x}(n-1)] - \alpha_v \left[ \frac{|y(n)|}{m} \right]^p, \quad (93)$$

which shall be termed the *characteristic function*. The *zero-crossing* or *root*  $v_0$  of  $\zeta_p(v)$  bears so the sought-after envelope value  $v(n)$ . Once  $v(n)$  is found (see Section 9.5), the current values of  $\tilde{x}$ ,  $|x|$ , and  $g$  are updated according to

$$\begin{aligned} \tilde{x}(n) &= v_0^p(n), \\ |x(n)| &= \sqrt[p]{\frac{\tilde{x}(n) - (1 - \alpha_v)\tilde{x}(n-1)}{\alpha_v}}, \\ g(n) &= \frac{|y(n)|/m}{|x(n)|}, \end{aligned} \quad (94a)$$

if  $v(n) > 1$ , or else

$$\begin{aligned} g(n) &= \alpha_g + (1 - \alpha_g)g(n-1), \\ |x(n)| &= \frac{|y(n)|/m}{g(n)}, \\ \tilde{x}(n) &= \alpha_v |\hat{x}(n)|^p + (1 - \alpha_v)\tilde{x}(n-1). \end{aligned} \quad (94b)$$

The decompressed sample is then computed as

$$z(n) = \text{sgn}(y) \cdot |x(n)|. \quad (95)$$

### 9.4.2 Attack-Release Phase Toggle

#### 9.4.2.1 Envelope smoothing

When a peak detector is in use,  $\alpha_v$  can take on two different values. The condition for the attack phase is given by (72). It is equivalent to

$$|x(n)|^p > \tilde{x}(n-1). \quad (96)$$

Assuming that  $\tilde{x}(n-1)$  is known, what is needed to be done is to express the unknown  $|x|$  in terms of  $|y|$  such that the above equation still holds true. If  $\alpha_g$  is rather small,  $\alpha_g \leq 0.1 \ll 1$ , or equivalently if  $\tau_g$  is sufficiently large,  $\tau_g \geq 0.5$  ms at 44.1-kHz sampling, the term  $\alpha_g f(n)$  in (89) is negligible, so it approximates (89) as

$$|y(n)|/m \approx g(n-1) \cdot |x(n)|. \quad (97)$$

Solving (97) for  $|x(n)|$  and plugging the result into (96), we obtain

$$\left[ \frac{|y(n)|/m}{g(n-1)} \right]^p > \tilde{x}(n-1). \quad (98)$$

If (98) is true, the detector is assumed to be in the attack phase.

#### 9.4.2.2 Gain smoothing

Just like the peak detector, the gain smoothing filter can be in either the attack or the release phase. The *necessary* condition for the attack phase in (78) may also be formulated as

$$v(n) > \left[ \frac{\kappa}{g(n-1)} \right]^{1/s} \quad \text{with } v(n) > l. \quad (99)$$

But as the current envelope value is unknown, we need to substitute  $v(n)$  in the above inequality by something that is computable. With this in mind, (89) is rewritten as

$$\begin{aligned} |y(n)|/m &= \left[ \alpha_g \frac{f(n)}{g(n-1)} + (1 - \alpha_g) \right] g(n-1) \cdot |x(n)| \\ &= \left[ 1 - \alpha_g \left( 1 - \frac{f(n)}{g(n-1)} \right) \right] g(n-1) \cdot |x(n)|. \end{aligned} \quad (100)$$

Provided that  $f(n) < g(n-1)$  and due to the fact that  $0 < \alpha_g \leq 1$ , the expression in square brackets in (100) is smaller than one, and thus during attack

$$|y(n)|/m < g(n-1) \cdot |x(n)|. \quad (101)$$

Substituting  $|x(n)|$  using (94a) and solving (101) for  $v(n)$  results in

$$v(n) > \sqrt[p]{\alpha_v \left[ \frac{|y(n)|/m}{g(n-1)} \right]^p + (1 - \alpha_v) \tilde{x}(n-1)}. \quad (102)$$

If  $v(n)$  in (99) is substituted by the expression on the right-hand side of (102), (99) still holds true. So, the following *sufficient* condition is used to predict the attack phase of the gain smoothing filter:

$$\sqrt[p]{\alpha_v \left[ \frac{|y(n)|/m}{g(n-1)} \right]^p + (1 - \alpha_v)\tilde{x}(n-1)} > \left[ \frac{\kappa}{g(n-1)} \right]^{1/s}. \quad (103)$$

Note that the values of all variables are known, if and when (103) is evaluated.

#### 9.4.3 Envelope Predictor

An instantaneous estimate of the envelope value  $v(n)$  is required not only to predict when compression is active, formally  $v(n) > l$  in (76), but also to initialize the iterative search algorithm in Section 9.5. By resorting once more to (89) it can be noted that in the opposite case where  $v(n) \leq l$ ,  $f(n) = 1$ , and so

$$|x(n)| = \frac{|y(n)|/m}{\alpha_g + (1 - \alpha_g)g(n-1)}. \quad (104)$$

The sound level of the input signal at instant  $n$  is therefore

$$v(n) = \sqrt[p]{\alpha_v \left[ \frac{|y(n)|/m}{\alpha_g + (1 - \alpha_g)g(n-1)} \right]^p + (1 - \alpha_v)\tilde{x}(n-1)}, \quad (105)$$

which must be greater than the threshold for compression to set in, whereas  $\alpha_v$  and  $\alpha_g$  are selected based on (98) and (103), respectively.

#### 9.4.4 Stereo Unlinking

First, one decompresses both channels of  $y(n)$  independently so as to obtain two estimates  $z_1(n)$  and  $z_2(n)$ . Using (79), one then computes  $\hat{y}_1(n)$  and  $\hat{y}_2(n)$  from  $z_1(n)$  and  $z_2(n)$ , and picks the channel ref for which  $y_{\text{ref}}(n) \approx \hat{y}_{\text{ref}}(n)$ . Finally, one updates the variables of the complementary channel  $\neg\text{ref}$ :

$$z_{\neg\text{ref}}(n) = \frac{y_{\neg\text{ref}}(n)/m}{g_{\text{ref}}(n)}, \quad (106)$$

$\tilde{x}_{\neg\text{ref}}(n)$  according to (71a), and  $g_{\neg\text{ref}}(n)$  according to (77).

#### 9.4.5 Error Analysis

Consider  $|x(n)|$  being estimated from  $|y(n)|$  according to

$$|\hat{x}(n)| = \frac{|y(n)|}{g(n-1)} \quad \text{with } m = 1. \quad (107)$$

The normalized error is then

$$e(n) = \frac{|\hat{x}(n)| - |x(n)|}{|y(n)|} \quad (108)$$

$$\begin{aligned} &= \left[ \frac{|y(n)|}{g(n-1)} - \frac{|y(n)|}{g(n)} \right] / |y(n)| \\ &= \frac{g(n) - g(n-1)}{g(n) \cdot g(n-1)}. \end{aligned} \quad (109)$$

As  $g > 0$  for all  $n$ ,  $e(n) < 0$  during attack and  $e(n) \geq 0$  during release, respectively. The instantaneous gain  $g(n)$  can also be expressed as

$$g(n) = \alpha_g \sum_{m=0}^N (1 - \alpha_g)^m f(n - m), \quad (110)$$

where  $N$  is the runtime in samples. Using (110) in (109), the magnitude of the error is given by

$$|e(n)| = \frac{\left| \sum_{m=0}^N (1 - \alpha_g)^m [f(n - m) - f(n - m - 1)] \right|}{\alpha_g \sum_{i,j=0}^N (1 - \alpha_g)^{i+j} f(n - i) f(n - j - 1)} \quad (111)$$

$$\begin{aligned} &\leq \frac{\sum_{m=0}^N (1 - \alpha_g)^m |f(n - m) - f(n - m - 1)|}{\alpha_g \sum_{i,j=0}^N (1 - \alpha_g)^{i+j} f(n - i) f(n - j - 1)}. \end{aligned} \quad (112)$$

For  $\gamma = 1$ , (111) becomes

$$|e(n)|_{\alpha_g=1} = \frac{|f(n) - f(n-1)|}{f(n) \cdot f(n-1)}, \quad (113)$$

whereas for  $\alpha_g \rightarrow 0$ , (112) converges to infinity:

$$\begin{aligned} |e(n)|_{\alpha_g \rightarrow 0} &\leq \frac{1}{\alpha_g} \frac{\sum_{m=0}^N \overbrace{|f(n-m) - f(n-m-1)|}^{>0 \text{ during compression}}}{\sum_{i,j=0}^N f(n-i) f(n-j-1)} \\ &\rightarrow \infty. \end{aligned} \quad (114)$$

Thus, the error is smaller for large  $\alpha_g$  or for short  $\tau_g$ . The smallest possible error is for  $\alpha_g = 1$ , which then again depends on the current and the previous value of  $f$ . The error accumulates if  $\alpha_g < 1$  with  $N$ . The difference between consecutive  $f$ -values is signal dependent. The signal envelope  $v(n)$  fluctuates less and is thus smoother for smaller

$\alpha_v$  or for longer  $\tau_v$ .  $f(n)$  is also more stable when the compression ratio  $R$  is low. For  $R = 1$ ,  $f(n)$  is perfectly constant. The threshold  $L$  has a negative impact on error propagation. The lower  $L$  the more the error depends on  $N$ , as more samples are compressed with different  $f$ -values. The RMS detector stabilizes the envelope more than the peak detector, which also reduces the error. Moreover, since usually  $\tau^{\text{att}} < \tau^{\text{rel}}$ , the error due to  $\alpha_v$  is smaller during release, whereas the error due to  $\alpha_g$  is smaller during attack. At last, the error can be expected to be larger at transition points between quiet to loud signal passages and vice versa.

The above error may cause a decision in favor of a wrong smoothing factor  $\alpha_v$  in (98), like  $\alpha_v^{\text{att}}$  instead of  $\alpha_v^{\text{rel}}$ , e.g. The decision error from (98) then propagates to (103). Given that  $\alpha_v^{\text{att}} > \alpha_v^{\text{rel}}$ , the error due to (107) is accentuated by (98) with the consequence that (103) is less reliable than (98). The total error in (103), thus, scales with  $|\alpha_v^{\text{att}} - \alpha_v^{\text{rel}}|$ . In regard to (105), reliability of the envelope's estimate is subject to validity of both (98) and (103). A better estimate is obtained when the sound level detector and the gain filter are both in either the attack or the release phase. Here, too, the estimation error increases with  $|\alpha_v^{\text{att}} - \alpha_v^{\text{rel}}|$  and also with  $|\alpha_g^{\text{att}} - \alpha_g^{\text{rel}}|$ . The makeup gain  $M$  has no impact on the error. Stereo linking may be another source of error. It all depends on how well the reference channel is detected.

## 9.5 NUMERICAL APPROXIMATION

An approximate solution of the characteristic function can be found, e.g., by means of *linearization*. The estimate from (105) may moreover serve as a starting point of an iterative search for an optimum:

$$v_{\text{init}} = \sqrt[p]{\alpha_v \left[ \frac{|y(n)|/m}{\alpha_g + (1 - \alpha_g)g(n-1)} \right]^p + (1 - \alpha_v)\tilde{x}(n-1)}. \quad (115)$$

The criterion for optimality is further chosen as the deviation of the characteristic function from zero, which is initialized to

$$\Delta_{\text{init}} = |\zeta_p(v_{\text{init}})|. \quad (116)$$

Thereon, (93) can be approximated using the equation of a straight line,  $\zeta = a \cdot v + c$ , where  $a$  is the slope and  $c$  is the  $\zeta$ -intercept. The root is characterized by the equation

$$\frac{\zeta_p(v_i + \Delta_i) - \zeta_p(v_i)}{\Delta_i} \cdot v + \zeta_p(v_i) = 0, \quad (117)$$

as shown in Fig. 15. The new estimate of the optimal  $v(n)$  is

$$v_{i+1} = v_i - \frac{\Delta_i \cdot \zeta_p(v_i)}{\zeta_p(v_i + \Delta_i) - \zeta_p(v_i)}. \quad (118)$$

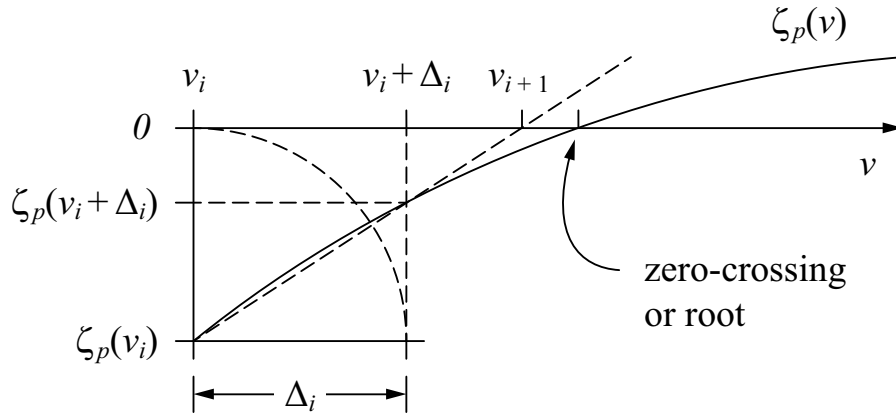


Figure 15: Graphical illustration of the iterative search for the root

If  $v_{i+1}$  is less optimal than  $v_i$ , the iteration is stopped and  $v_i$  is the final estimate. The iteration is also stopped if  $\Delta_{i+1}$  is smaller than some  $\epsilon$ . In the latter case,  $v_{i+1}$  has the optimal value with respect to the chosen criterion. Otherwise,  $v_i$  is set to  $v_{i+1}$  and  $\Delta_i$  is set to  $\Delta_{i+1}$  after every step and the procedure is repeated until  $v_{i+1}$  has converged to a more optimal value. The proposed method is a special form of the *secant method* with a single initial value  $v_{\text{init}}$ .

## 9.6 GENERAL REMARKS

### 9.6.1 Lookahead

A compressor with a lookahead function, i.e. with a delay in the main signal path as in [15, p. 106], uses past input samples as weighted output samples. Now that some future input samples are required to invert the process—which are unavailable, the inversion is rendered impossible.  $g(n)$  and  $x(n)$  must thus be *in sync* for the approach to be applied. For proof and reasoning see Appendix C.

### 9.6.2 Clipping and Limiting

Another point worth mentioning is that “hard” clipping and “brick-wall” limiting are special cases of compression with the attack time set to zero and the compression ratio set to  $\infty : 1$ . The nonlinearity  $F$  in that particular case is a one-to-many mapping, which by definition is *noninvertible*.

### 9.6.3 Logarithmic Gain Smoothing

Certain compressor models apply gain smoothing in the logarithmic domain. In that case, (77) is replaced by

$$g(n) = f^{\alpha_g(n)} \cdot g^{1-\alpha_g(n-1)}, \quad (119)$$

so that the characteristic function becomes

$$\begin{aligned} \zeta_p(v) = & \mu [v^{-\alpha_g S(n)} \cdot g^{1-\alpha_g(n-1)}]^p \\ & \cdot [v^p(n) - (1 - \alpha_v)\tilde{x}(n-1)] - \alpha_v \left[ \frac{|y(n)|}{m} \right]^p \end{aligned} \quad (120)$$

with  $\mu = l^{p\beta S}$ . Equations (98) and (103) remain valid, whereas (105) and consequently (115) are now to be replaced by

$$v(n) = \sqrt[p]{\alpha_v \left[ \frac{|y(n)|/m}{g^{1-\alpha_g(n-1)}} \right]^p + (1 - \alpha_v)\tilde{x}(n-1)}. \quad (121)$$

Finally, taking (119) into account, the gain value in (94b) is computed as

$$g(n) = g^{1-\alpha_g(n-1)} \quad (122)$$

instead.

## 9.7 PSEUDOCODE

The algorithm is divided into three parts, given below in the form of pseudocode. Algorithm 1 outlines the compressor that corresponds to the model from Section 9.2. Algorithm 2 illustrates the decompressor described in Section 9.4. The root-finding algorithm from Section 9.5 is summarized in Algorithm 3.  $T_s$  represents the sampling period in ms.

## 9.8 PARAMETER QUANTIZATION AND CODING

The threshold level  $L$  and the makeup gain  $M$ , both given in dB, can be quantized using (48) on predefined ranges. The compression ratio  $R \in [1, 60]$  can be compressed using (49) or (50) by setting  $x$  to  $R - 1$  and  $x_{\max}$  to 59. Instead of coding the time constants  $\tau_v$  and  $\tau_g$ , it is more appropriate to code the smoothing factors,  $\alpha_v, \alpha_g \in (0, 1]$ , using (49) or (50) in combination with (48).

## 9.9 PERFORMANCE EVALUATION

### 9.9.1 Experimental Setup

The decompressor's performance is evaluated on a synthetic signal and also on "natural" audio material. The former is generated from



**Algorithm 1** Single-input single-output compressor

---

```

function COMPRESS( $x_n, T_s, \theta$ )
   $\tilde{x}_n \leftarrow 0$ 
   $g_n \leftarrow 1$ 
  for  $n \leftarrow 1, N$  do
    if  $|x_n|^p > \tilde{x}_n$  then
       $\alpha_v \leftarrow 1 - \exp(-2.2T_s/\tau_v^{\text{att}})$ 
    else
       $\alpha_v \leftarrow 1 - \exp(-2.2T_s/\tau_v^{\text{rel}})$ 
    end if
     $\tilde{x}_n \leftarrow \alpha_v |x_n|^p + (1 - \alpha_v)\tilde{x}_n$ 
     $v_n \leftarrow \sqrt[p]{\tilde{x}_n}$ 
    if  $v_n > l$  then
       $f_n \leftarrow \kappa v_n^{-S}$ 
    else
       $f_n \leftarrow 1$ 
    end if
    if  $f_n < g_n$  then
       $\alpha_g \leftarrow 1 - \exp(-2.2T_s/\tau_g^{\text{att}})$ 
    else
       $\alpha_g \leftarrow 1 - \exp(-2.2T_s/\tau_g^{\text{rel}})$ 
    end if
     $g_n \leftarrow \alpha_g f_n + (1 - \alpha_g)g_n$ 
     $y_n \leftarrow g_n x_n$ 
  end for
  return  $y_n$ 
end function

```

---

a weighted sum of Heaviside step functions, while the latter consists of twelve items including speech, sung voice, music, and jingles. All items are normalized to  $-16$  LUFS [86]. A detailed overview of used compressor settings is given in Table 7. They correspond to presets of commercial compressor plug-ins. The  $\epsilon$ -value in the break condition of Algorithm 3 is set to  $1 \cdot 10^{-12}$ .

The inverse approach is evaluated using the following metrics. The root-mean-square error (RMSE),

$$\text{RMSE} = 20 \log_{10} \sqrt{\frac{1}{N} \sum_{n=1}^N [z(n) - x(n)]^2}, \quad (123)$$

given in decibels relative to full scale (dBFS), the perceptual similarity between the decompressed and the original signal, and the execution time of the decompressor relative to real time (RT). Furthermore, we present the percentage of compressed samples, the mean number of iterations until convergence per compressed sample, the error rate of the attack-release toggle for the gain smoothing filter, and finally

**Algorithm 2** Single-input single-output decompressor

---

```

function DECOMPRESS( $y_n, T_s, \theta, \epsilon$ )
   $\tilde{x}_n \leftarrow 0$ 
   $g_n \leftarrow 1$ 
  for  $n \leftarrow 1, N$  do
    if  $|y_n| > \sqrt[p]{\tilde{x}_n} \cdot g_n$  then
       $\alpha_v \leftarrow 1 - \exp(-2.2T_s/\tau_v^{\text{att}})$ 
    else
       $\alpha_v \leftarrow 1 - \exp(-2.2T_s/\tau_v^{\text{rel}})$ 
    end if
    if  $|y_n| > \sqrt[p]{[(\kappa/g_n)^{p/s} - (1 - \alpha_v)\tilde{x}_n] / \alpha_v} \cdot g_n$  then
       $\alpha_g \leftarrow 1 - \exp(-2.2T_s/\tau_g^{\text{att}})$ 
    else
       $\alpha_g \leftarrow 1 - \exp(-2.2T_s/\tau_g^{\text{rel}})$ 
    end if
    if  $|y_n| > \sqrt[p]{\frac{1^p - (1 - \alpha_v)\tilde{x}_n}{\alpha_v} \cdot [\alpha_g + (1 - \alpha_g)g_n]}$  then
       $v_n \leftarrow \sqrt[p]{\alpha_v \left[ \frac{|y_n|}{\alpha_g + (1 - \alpha_g)g_n} \right]^p + (1 - \alpha_v)\tilde{x}_n}$ 
       $v_0 \leftarrow \text{CHARFZERO}(v_n, \epsilon)$ 
       $|x_n| \leftarrow \sqrt[p]{[v_0^p - (1 - \alpha_v)\tilde{x}_n] / \alpha_v}$ 
       $\tilde{x}_n \leftarrow v_0^p$ 
       $g_n \leftarrow |y_n| / |x_n|$ 
    else
       $g_n \leftarrow \alpha_g + (1 - \alpha_g)g_n$ 
       $|x_n| \leftarrow |y_n| / g_n$ 
       $\tilde{x}_n \leftarrow \alpha_v |x_n|^p + (1 - \alpha_v)\tilde{x}_n$ 
    end if
     $x_n \leftarrow \text{sgn}(y_n) \cdot |x_n|$ 
  end for
  return  $x_n$ 
end function

```

---

the error rate of the envelope predictor. The perceptual similarity is assessed by the PEMO-Q [70] software [71] with  $\text{PSM}_t$  as metric. The simulations are run in MATLAB on an Intel Core i5-520M CPU.

### 9.9.2 Experimental Results

Fig. 16 shows the inverse signal  $z(n)$  to the synthetic input signal  $x(n)$  using an RMS detector. The inverse signal  $z(n)$  is obtained from the compressed signal  $y(n)$  with an error of ca.  $-129$  dBFS. It is visually indistinguishable from the original signal  $x(n)$ . Due to the fact that the signal envelope is constant most of the time, the error is no-

**Algorithm 3** Root-finding algorithm

---

```

function CHARFZERO( $v_n, \epsilon$ )
   $v_i \leftarrow v_n$ 
  repeat
     $\Delta_i \leftarrow |\zeta_p(v_i)|$ 
     $v_i \leftarrow v_i - \Delta_i \cdot \zeta_p(v_i) / [\zeta_p(v_i + \Delta_i) - \zeta_p(v_i)]$ 
    if  $|\zeta_p(v_i)| > \Delta_i$  then
      return  $v_n$ 
    end if
     $v_n \leftarrow v_i$ 
  until  $|\zeta_p(v_i)| < \epsilon$ 
  return  $v_i$ 
end function

```

---

| PARAMETER                                     | A       | B       | C       | D       | E       |
|---|---------|---------|---------|---------|---------|
| Threshold (dBFS)                              | -32.0   | -19.9   | -24.4   | -26.3   | -38.0   |
| Ratio (dB <sub>in</sub> : dB <sub>out</sub> ) | 3.0 : 1 | 1.8 : 1 | 3.2 : 1 | 7.3 : 1 | 4.9 : 1 |
| Envelope attack (ms)                          | 5.0     | 5.0     | 5.0     | 5.0     | 5.0     |
| Envelope release (ms)                         | 5.0     | 5.0     | 5.0     | 5.0     | 5.0     |
| Gain attack (ms)                              | 13.0    | 11.0    | 5.8     | 9.0     | 13.1    |
| Gain release (ms)                             | 435     | 49      | 112     | 705     | 257     |

---

Table 7: Selected compressor settings

ticeable only around transition points, which are very few. The performance figures for real-world audio are given in Table 8. The results suggest that the decompressed signal is perceptually indistinguishable from the original: the  $\text{PSM}_t$  values are flawless. This observation has been confirmed through informal listening tests.

As can be seen from Table 8, the largest inversion error is associated with setting E and the smallest with setting B. For all five settings, the error is larger when an RMS detector is in use. This is partly due to the fact that  $\zeta_2(v)$  has a stronger curvature in comparison to  $\zeta_1(v)$ . By defining the distance in (116) as  $\Delta \triangleq \sqrt[p]{|\zeta_p(v)|}$ , it is possible to attain a smaller error for an RMS detector at the cost of a slightly longer runtime. In most cases, the envelope predictor works more reliably as compared to the toggle switch between attack and release. It can also be observed that the choice of time constants seems to have little impact on decompressor's accuracy. The major parameters that affect the decompressor's performance are L and R, while the threshold is evidently the predominant one: the RMSE strongly correlates with the threshold level.

Figs. 17–18 show the inversion error as a function of various time constants. These are in the range of typical attack and release times

for a limiter (peak) or compressor (RMS) [15, pp. 109–110]. It can be observed that the inversion accuracy depends on the release time of the peak detector and not so much on its attack time for both the envelope and the gain filter, see Figs. 17, 18b. For the envelope filter, all error curves exhibit a local dip around a release time of 0.5 s. The error increases steeply below that bound but moderately with larger values. In the proximity of 5 s, the error converges to  $-130$  dBFS. With regard to the gain filter, the error behaves in a reverse manner. The curves in Fig. 18b exhibit a local peak around 0.5 s with a value of  $-180$  dBFS. It can further be observed in Fig. 17a that the curve for  $\tau_v^{\text{rel}} = 1$  ms has a dip where  $\tau_v^{\text{att}}$  is close to 1 ms, i.e. where  $|\alpha_v^{\text{att}} - \alpha_v^{\text{rel}}|$  is minimal. This is also true for Fig. 17c and Fig. 17d: the lowest error is where the attack and release times are identical. As a general rule, the error that is due to the attack-release switch is smaller for the gain filter in Fig. 18.

Looking at Fig. 19 one can see that the error decreases with threshold and increases with compression ratio. At a ratio of 10 : 1 and beyond, the RMSE scales almost exclusively with the threshold. The lower the threshold, the stronger the error propagates between decompressed samples, which leads to a larger RMSE value. The RMS detector further augments the error because it stabilizes the envelope  $v(n)$  more than the peak detector. Clearly, the threshold level has the highest impact on the decompressor's accuracy.

|                               | A     |       | B     |       | C     |       | D     |       | E     |       |
|-------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
|                               | Peak  | RMS   | Peak  | RMS   | Peak  | RMS   | Peak  | RMS   | Peak  | RMS   |
| RMSE (dBFS)                   | -74.4 | -71.2 | -97.2 | -93.7 | -81.0 | -77.8 | -76.3 | -69.5 | -63.2 | -53.8 |
| PSM <sub>t</sub> (PEMO-Q)     | 1.00  | 1.00  | 1.00  | 1.00  | 1.00  | 1.00  | 1.00  | 1.00  | 1.00  | 1.00  |
| Execution time (RT)           | 0.54  | 0.53  | 0.40  | 0.44  | 0.47  | 0.49  | 0.48  | 0.50  | 0.54  | 0.54  |
| Compression rate (%)          | 78.7  | 80.8  | 38.5  | 50.7  | 61.8  | 67.3  | 67.6  | 71.8  | 85.2  | 86.4  |
| Iterations per sample (#)     | 1.04  | 1.02  | 1.00  | 1.01  | 1.07  | 1.06  | 1.05  | 1.03  | 1.09  | 1.04  |
| Attack-release error rate (%) | 0.05  | 0.09  | 0.01  | 0.01  | 0.02  | 0.04  | 0.01  | 0.03  | 0.14  | 0.51  |
| State error rate (%)          | 0.02  | 0.03  | 0.01  | 0.01  | 0.01  | 0.02  | 0.02  | 0.03  | 0.03  | 0.05  |

Table 8: Performance figures obtained for real-world audio material (12 items in total)

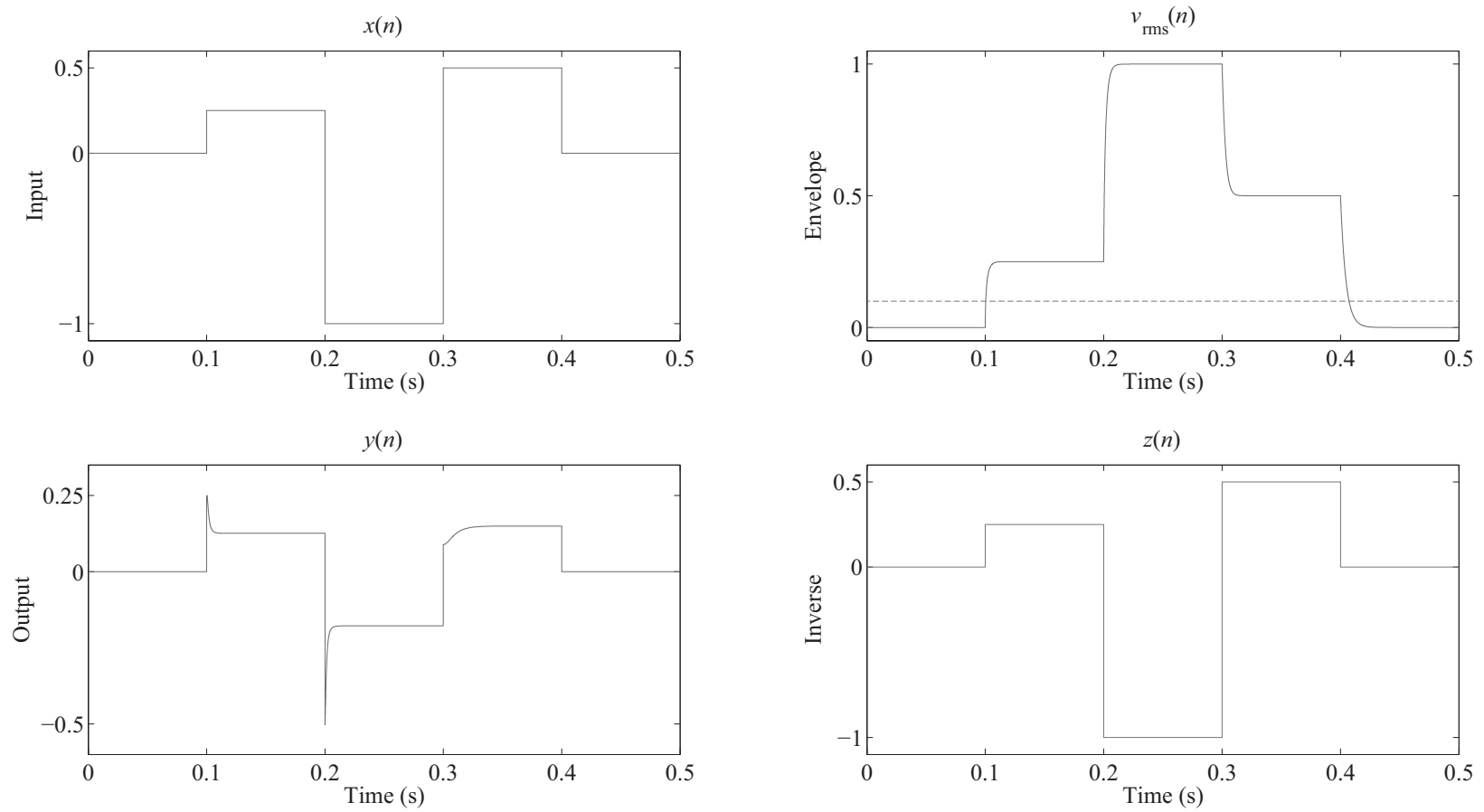


Figure 16: An illustrative example using an **RMS** amplitude detector with  $\tau_v$  set to 5 ms, a threshold of  $-20$  dBFS (dashed line in the upper right corner), a compression ratio of 4:1, and  $\tau_g$  set to 1.6 ms for attack and 17 ms for release, respectively. The **RMSE** is ca.  $-129$  dBFS.

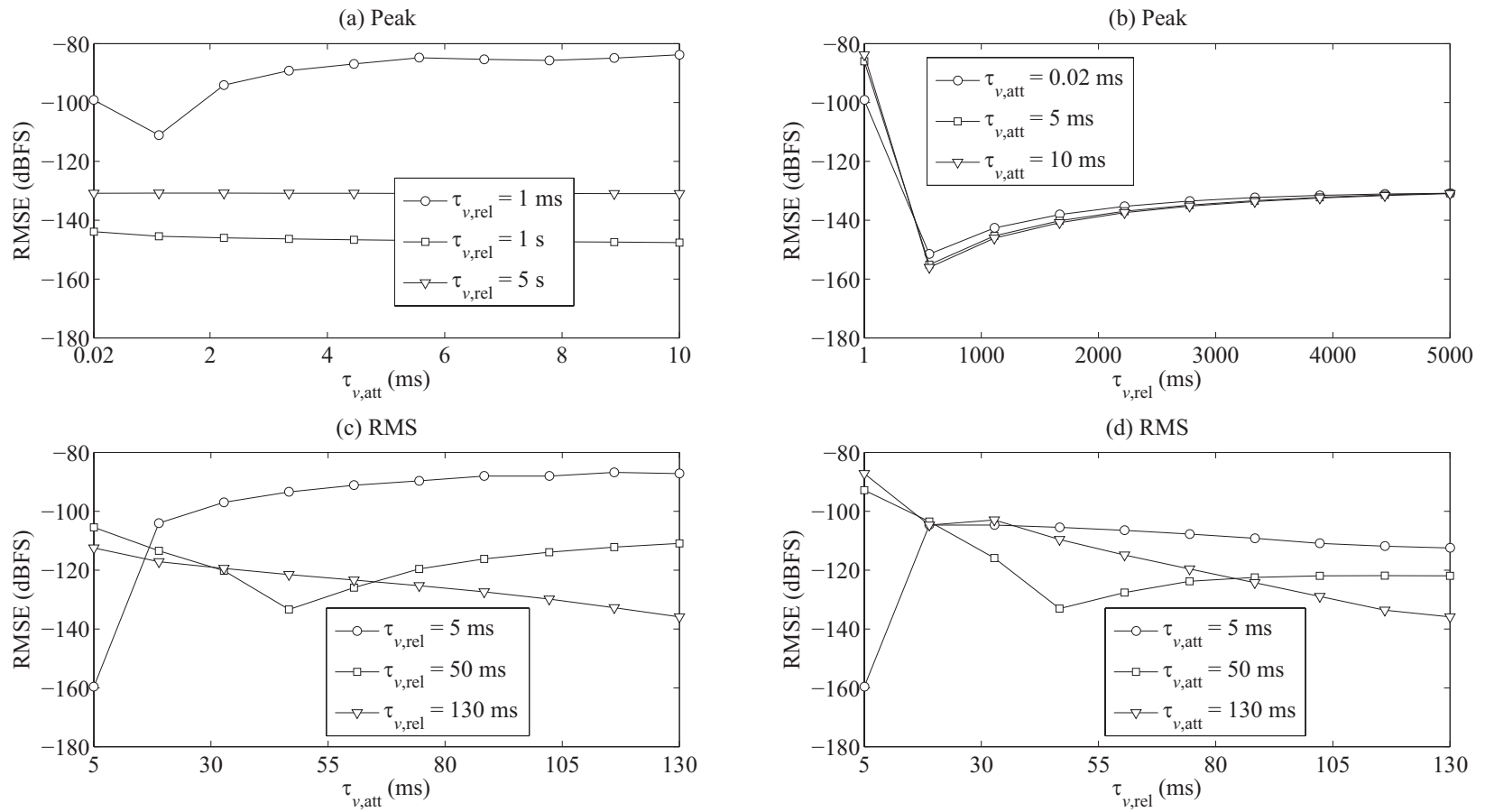


Figure 17: RMSE as a function of typical attack and release times using a peak (upper row) or an RMS amplitude detector (lower row). In the left column, the attack time of the envelope filter is varied while the release time is held constant. The right column shows the reverse case. The time constants of the gain filter are fixed at zero. In all four cases, threshold and ratio are fixed at  $-32$  dBFS and  $4 : 1$ , respectively.

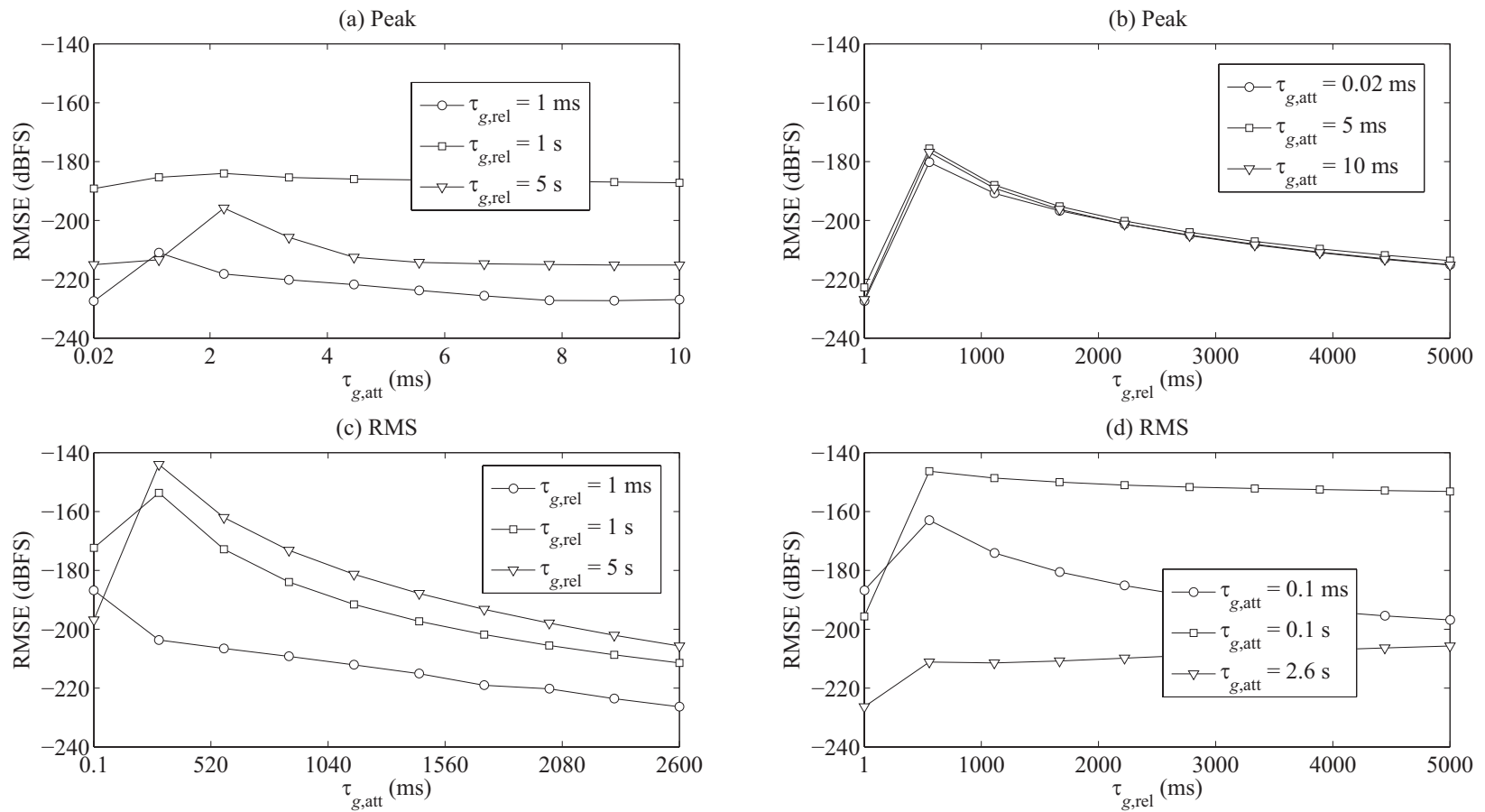


Figure 18: RMSE as a function of typical attack and release times using a peak (upper row) or an RMS amplitude detector (lower row). In the left column, the attack time of the gain filter is varied while the release time is held constant. The right column shows the reverse case. The time constants of the envelope filter are fixed at zero. In all four cases, threshold and ratio are fixed at  $-32$  dBFS and  $4 : 1$ , respectively.



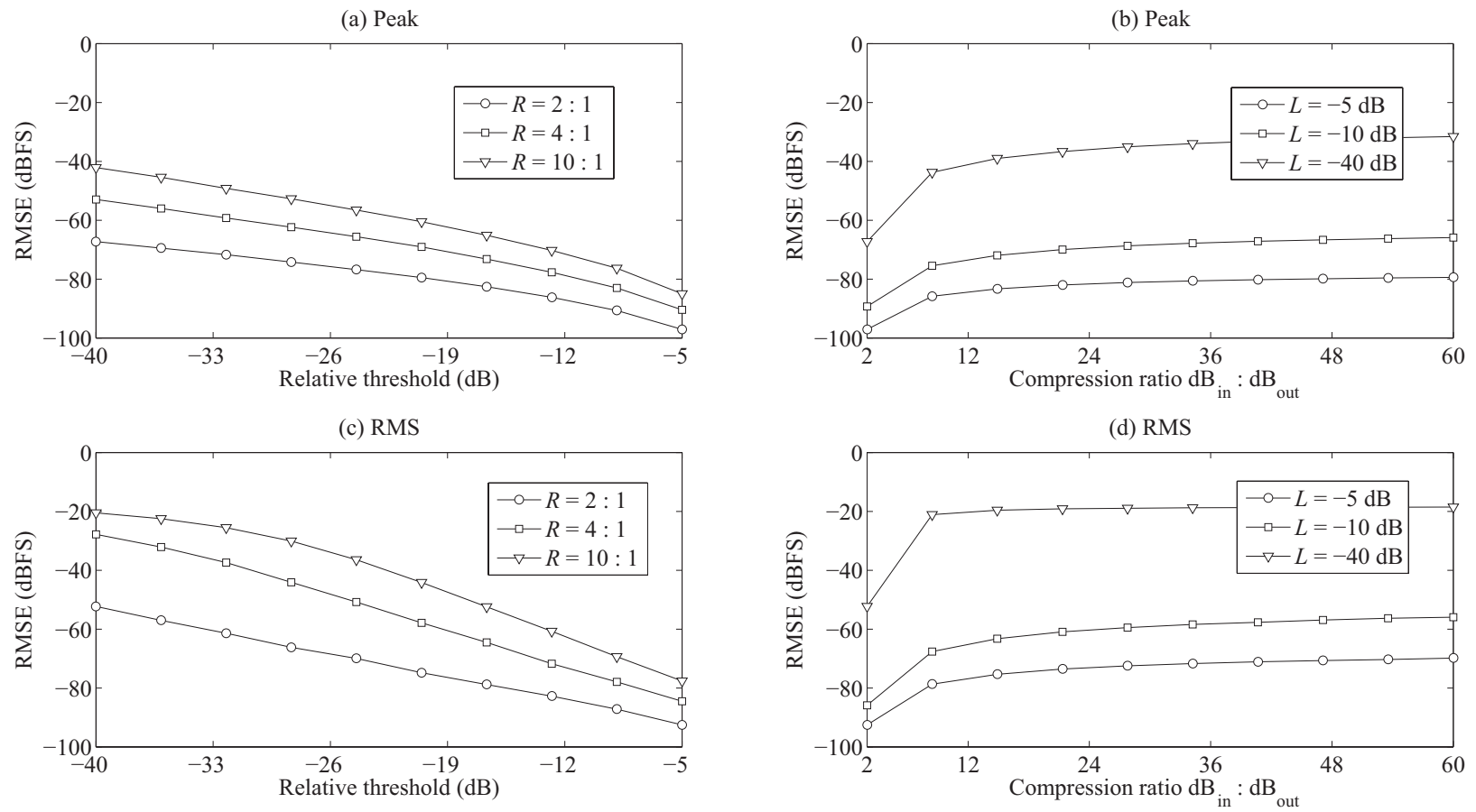


Figure 19: **RMSE** as a function of threshold relative to the signal's average loudness level (left column) and compression ratio (right column) using a peak (upper row) or an **RMS** amplitude detector (lower row). The time constants are:  $\tau_v = 5$  ms,  $\tau_g^{\text{att}} = 20$  ms, and  $\tau_g^{\text{rel}} = 1$  s.



## TWO-STAGE CASCADE CONNECTION

---

### 10.1 SYSTEM OVERVIEW

To account for a simplified but complete music production chain that consists of mixing and mastering, the two processing steps from Sections 7 and 9 are combined in a two-stage cascade scheme. The corresponding encoder and decoder block diagrams are illustrated in Figs. 22a and 22b.

### 10.2 CASCADE ENCODER

Fig. 22a shows the cascade connection for a practical encoder. It contains an analysis block that computes the STPSDs, a mixdown block representing the mixing stage, and a DRC block that represents the mastering stage. The multiplexing block assembles the bitstream that comprises the compressed mixture and the coded side information. The processing steps include:

1. STFT analysis of source signals, see [87]
2. STPSD computation from magnitude-squared spectra
3. Stereo mixdown, see Section 7.3.2
4. DRC, see Algorithm 1
5. Quantization and coding of side information, see Sections 7.7 and 9.8
6. Bitstream framing

### 10.3 CASCADE DECODER

A practical decoder is shown in Fig. 22b. The demultiplexing block disassembles the bitstream into the compressed mixture and the side information. The inverse DRC block decompresses the mixture. The decompressed mixture is decomposed in the source separation block with the aid of the decoded side information yielding the source signal estimates in the time domain. The processing steps include:

1. Bitstream parsing
2. Decoding (and dequantization) of side information
3. Inverse DRC using compression parameters, see Algorithm 2
4. STFT analysis of mixture channels
5. Source separation using mixing parameters and STPSDs

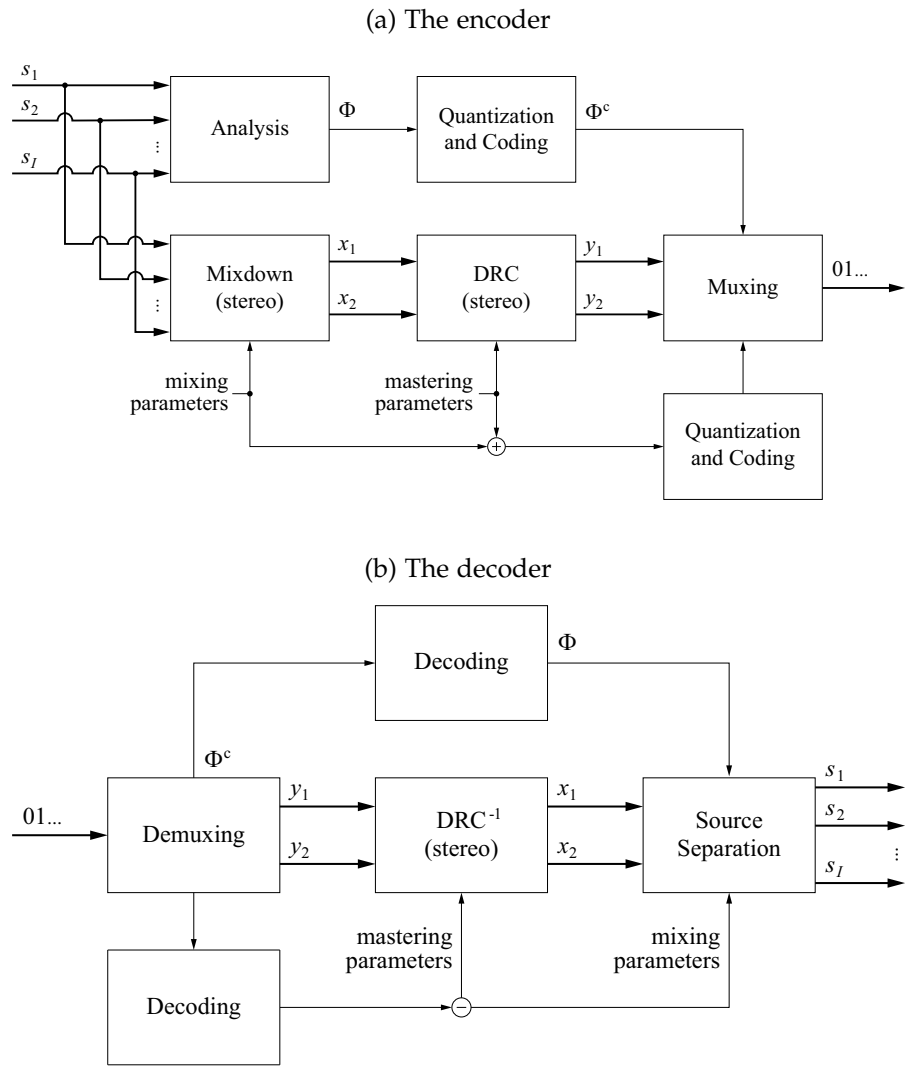


Figure 20: A two-stage cascade connection

- The number of active sources in a TF point or band and their indices are determined by comparing the STPSD values with the noise-floor power level (−60 dB, e.g.)
  - The appropriate filter is chosen based on the number of active sources, see Sections 7.5.1, 7.5.2, and 7.6.4
6. STFT synthesis of source signal estimates, see [87]

## 10.4 PERFORMANCE CHARACTERISTICS

### 10.4.1 Algorithmic Delay

The algorithmic delay of the cascade connection is determined by the framing and overlap delay of the STFT and its inverse. When using a 2048-length symmetric window with 50-% overlap between frames, the algorithmic delay amounts to 2047 samples. This corresponds to 46.4 ms at a sampling rate of 44.1 kHz.

### 10.4.2 Computational Complexity

The computations performed in the cascade connection depend on the frequency content of source signals and also the signal dynamics, which vary from mixture to mixture. So, it is convenient to analyze the runtime complexity for the worst-case scenario in terms of “big O” notation. Moreover, it is desirable to establish a relationship between the running time and the following input parameters: the number of sources  $I$ , the number of frequency bands  $Z$ , and the transform length  $N$ ,  $I < Z < N$ . All arithmetic operations that are counted shall require exactly one unit of time to execute. The results are shown in Table 9. The figures reveal that the decoder’s complexity is comparable to the complexity of the encoder. The execution time is dominated by the  $I$ -fold STFT and its inverse.

### 10.4.3 Side-Information Rate

The side-information rate of the cascade connection is given by the number of bits communicated to the decoder per time frame. These comprise the compression parameters, the mixing parameters, and the STPSDs. The compression parameters and the mixing parameters are signaled to the decoder once at the beginning of the transmission. The STPSD values are represented as  $Q$ -bit unsigned integers. Table 10 provides an overview of the capacities that are necessary to store the STPSDs for various numbers of frequency bands. They are calculated as

$$\text{bitrate} = \frac{f_s}{L - M} \cdot Q \cdot Z \cdot I, \quad (124)$$

| SUBROUTINE                                      | ARITHMETIC OPERATIONS |
|---|-----------------------|
| STFT analysis                                   | $O(I \cdot N \log N)$ |
| STPSD computation                               | $O(I \cdot N)$        |
| Stereo mixdown                                  | $O(I \cdot N)$        |
| DRC   | $O(N)$                |
| Quantization and coding                         | $O(I \cdot N)$        |
| $T_{\text{enc}}(I, Z, N) = O(I \cdot N \log N)$ |                       |
| Decoding and extrapolation                      | $O(I \cdot Z)$        |
| Inverse DRC                                     | $O(N)$                |
| STFT analysis                                   | $O(N \log N)$         |
| Source separation                               | $O(I \cdot N)$        |
| STFT synthesis                                  | $O(I \cdot N \log N)$ |
| $T_{\text{dec}}(I, Z, N) = O(I \cdot N \log N)$ |                       |

Table 9: Runtime complexity of the cascade connection as a function of the number of sources, the number of frequency bands, and the frame length. It is assumed that the transform length is equal to the frame length.

where  $L$  is the frame length,  $M$  is the overlap, and  $f_s$  is the sampling frequency. The information rate is varied upon a subdivision the ERB by an integer factor. In general, the finer the frequency resolution the higher the observed quality of the estimates. On the other hand, the greater the number of sources the finer is to be chosen the frequency resolution to obtain a quality comparable to a sparser configuration. Whatever the case, the figures in Table 10 can be drawn upon to make an estimate for the side-information rate of the cascade connection, as the rate of the compression and mixing parameters is comparably negligible.<sup>1</sup>

Table 10 also shows the STPSD rate obtained using DPCM. The latter is implemented as follows. The probability distribution of the input symbols is derived from the number of occurrences of each possible difference value in a training set. It is observed that the difference signal both in time and frequency direction, e.g., has a Laplace( $\mu, b$ ) distribution with  $\mu \approx -0.2$  and  $b \approx 2$ . Then, a Huffman codebook [89] can be generated from the input probability distribution with a mean codeword length of 3.5 bit. This corresponds to a compression ratio of 1.7 : 1, which again means that almost twice as many source signals can now be extracted from the mixture for the same amount of side information.

1. In the case of an instantaneous mixture, the mixing coefficients can also be estimated from the mixture signal using the algorithm in [88].

| SUBDIVISION FACTOR | NUMBER OF FREQUENCY BANDS | BITRATE IN KBPS PER SOURCE |
|--------------------|---------------------------|----------------------------|
| –                  | 39                        | 10.1 / 5.88                |
| 2                  | 76                        | 19.6 / 11.5                |
| 3                  | 108                       | 28.0 / 16.3                |
| 4                  | 136                       | 35.2 / 20.6                |
| 5                  | 163                       | 42.2 / 24.6                |
| ⋮                  | ⋮                         | ⋮                          |

Table 10: Information rate of STPSD values (left) and STPSD difference values using DPCM (right) if quantized with 6 bits at 44.1-kHz sample rate and 16-kHz cutoff

| PARAMETER                                 | DESCRIPTION      | VALUE   |
|---|------------------|---------|
| p   | Type             | RMS     |
| L (dBFS)                                  | Threshold        | −32.0   |
| R (dB <sub>in</sub> : dB <sub>out</sub> ) | Ratio            | 3.0 : 1 |
| $\tau_{v,att}$ (ms)                       | Envelope attack  | 5.0     |
| $\tau_{v,rel}$ (ms)                       | Envelope release |         |
| $\tau_{g,att}$ (ms)                       | Gain attack      | 13.0    |
| $\tau_{g,rel}$ (ms)                       | Gain release     | 435     |
| M (dB)                                    | Makeup           | 9.0     |

Table 11: Compressor setting used for the complete mix

## 10.5 PERFORMANCE EVALUATION

### 10.5.1 Experimental Setup

A 2048-point FFT is employed together with a KBD window of the same size. Succeeding frames overlap by 50 %. The cascade is tested on an excerpt from Fort Minor’s “Remember the Name” multitrack. The latter is decomposed into 5 mono sources and is 24 s long. The compressor setting is shown in Table 11. To exclude a performance bias due to quantization, the mastering and mixing parameters are considered known at the decoder. The STPSDs are quantized with 6 bits on a  $1/2$ -ERB frequency scale. By applying DPCM with Huffman coding to the quantized STPSD values, the mean side-information rate is reduced to roughly 10 kbps per source. The simulations are run in MATLAB.

To evaluate the proposed cascade connection, the following metrics are used: the RMSE given in dBFS and the PSM. The PSM is computed

| MIXTURE TYPE | RMSE (DBFS) | SNR (DB) |
|--------------|-------------|----------|
| Compressed   | -31.8       | 3.08     |
| Compressed*  | -36.0       | 7.27     |
| Decompressed | -62.3       | 33.6     |

Table 12: RMSE and SNR for the three mixture signals

with PEMO-Q. In [70] it is said that PEMO-Q shows a slightly better performance than Perceptual Evaluation of Audio Quality (PEAQ) [78].

### 10.5.2 Experimental Results

The results of the experiment are illustrated in Fig. 21. The asterisk marks the compressed mix *without* makeup gain, i.e.  $M = 0$  dB. The RMSE and the SNR for each mixture signal,

$$\text{SNR} \triangleq 10 \log_{10} \frac{\sum_l \|\mathbf{x}_l\|^2}{\sum_l \|\mathbf{y}_l - \mathbf{x}_l\|^2}, \quad (125)$$

where  $\|\cdot\|$  is the Euclidean norm, are given in Table 12.

It can be noted that the separated source signals exhibit relatively high quality when the mix is uncompressed (see dashed bar). The RMSE for the vocal reaches almost -60 dBFS while being below -40 dBFS for the rest. The decompressor's performance is practically free from error: the  $\Delta\text{RMSE}$  and  $\Delta\text{PSM}$  values for each track are close to zero after decompression (see lower row). The RMSE level decreases, as expected, if the makeup gain is removed from the compressed mix, but does not reach the level of the uncompressed mix. This proves that the waveform of the mix has been altered by the compressor. The RMSE difference between the two compressed mixtures is so due to scaling. On the contrary, the corresponding PSM values are equal, which shows that the PSM metric is scale-independent.

In the given example, the PSM improvement due to decompression is mostly evident for the bass track. For the other tracks, the PCMV filter provides an estimate which is perceptually very similar to the reference, even if the mix is compressed. Yet, a so-called "pumping" coming from hard compression can be heard clearly on the vocal track. The effect is more audible for faster attack and release. Hence, for the used compressor setting, the decompressor is indispensable to achieve high perceptual quality.



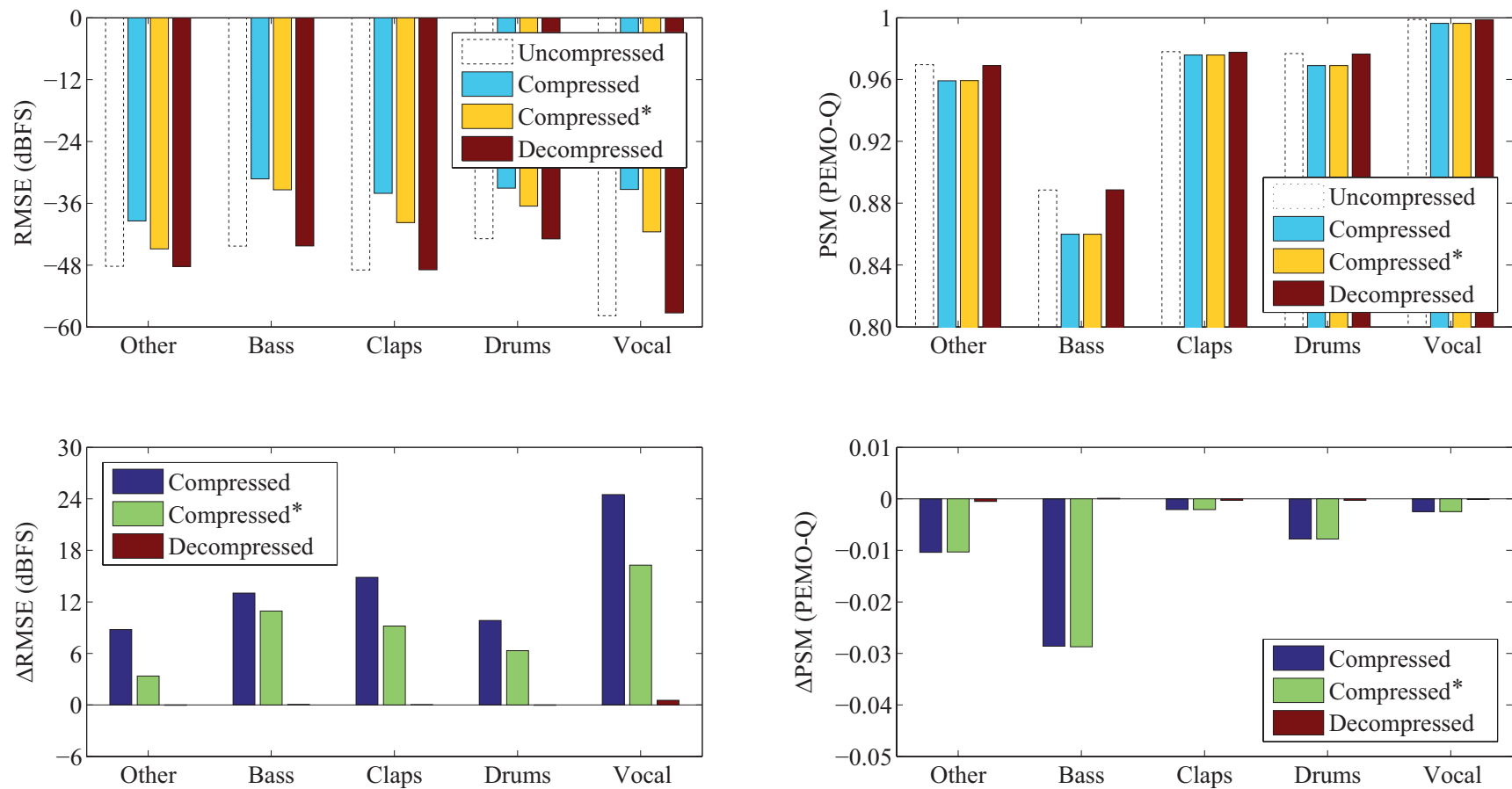


Figure 21: **RMSE** and **PSM** values for the multitrack (upper row) and the corresponding difference values between the estimates from the de-/compressed and the uncompressed mixture signal (lower row). The asterisk (\*) indicates  $M = 0$  (no gain).



## 11.1 INTRODUCTION

Ever since the end of the last century, coding of audiovisual objects has been of particular interest to the [MPEG](#), and it has gained importance in recent years. Whereas the first audio coders were all channel-based, a paradigm shift towards source-based coding was initiated by works like [\[56\]](#). A more recent example is [MPEG's SAOC](#), see [Chapter 8](#), or the work in [\[90\]](#). The necessity for object-based coding in the sense of sound sources arises when distinct audio objects are to be stored or transmitted for the purpose of post-hoc reproduction in different environments. So far, its application fields include remixing, video gaming, home cinema or 3D audio, and there might be more in the future.

The work presented here focuses on the question how a number of given source signals or objects can be represented by a reduced number of mixture channels and recovered using the mixture and a small amount of metadata. The work by [Faller \[56\]](#) considers only single-channel mixtures and has no means to scale the quality after demixing. The resulting quality can so be expected to be the worst possible, as a single-channel mixture exhibits the highest overlap between objects. Whereas in [\[90\]](#) [Hotho et al.](#) generalize the mixture to more than one channel and propose to use the residual to scale the audio quality up to perceptual transparency, there is no explicit control over the quality, except that the latter is said to improve with the bandwidth of the residual signal by rule of thumb. Moreover, the works in [\[56\]](#) and [\[90\]](#) evaluate the quality empirically after rendering the decoded objects into a prescribed format such as 5.1 surround and are consequently bound to the sound reproduction system.

Though related to previous approaches, this work capitalizes on quality-driven demixing which is further *independent of mixing and rendering after demixing*. Moreover, the [ISS](#) approach is pursued. As it is demonstrated in [Chapter 7](#), an underdetermined linear mixture can be decomposed into an arbitrary number of components by means of spatial filtering. When the separation is carried out in the [STFT](#) domain, the estimates show distortion in amplitude and phase. It is clear that the amount of distortion, which is due to bleed from other sources but also the filter response, decreases with the number of mixture channels because the separation problem becomes better conditioned and the array gain increases. In this chapter it is shown

how these facts can be exploited to code audio objects in a *controllable* way.

## 11.2 SIGNAL AND SYSTEM MODEL

Using the *STFT* signal representation, the source signal components  $S_i(k, m)$ ,  $i = 1, 2, \dots, I$ , are circular symmetric complex normal random variables with zero mean and a diagonal covariance matrix  $\mathbf{R}_s(k, m) = \text{diag}[\Phi_{s_1}(k, m), \Phi_{s_2}(k, m), \dots, \Phi_{s_I}(k, m)]$ .  $\{\Phi_{s_i}(k, m)\}_k$  is the *STPSD* of the  $i$ th source signal. The source signals are linearly combined into an  $M$ -channel mixture signal ( $M < I$ ) according to

$$\mathbf{x}(k, m) = \sum_{i=1}^I \mathbf{a}_i S_i(k, m) = \mathbf{A}\mathbf{s}(k, m), \quad (126)$$

where  $\mathbf{A} \in \mathbb{R}^{M \times I}$  represents an instantaneous mixing system that is assumed real for practical reasons. This model is identical with that from Sections 7.2 and 7.3. Equation (126) is so an extension of (13) to  $M$  channels.

## 11.3 PROBLEM FORMULATION

The problem at hand is formulated as follows. Given the mixing rule<sup>1</sup> and the *STPSDs* of the source signals, find a low-rank signal space representation  $\mathbf{x}(k, m)$  which satisfies a minimum-similarity constraint on the recovered source signals after transformation back to the original signal space. In other words, what is the minimum number of channels  $M_{\min}$  into which one can mix the source signals and yet maintain the desired quality level after demixing? The quality metric shall further relate to, but not model, human perception.

## 11.4 PROPOSED SOLUTION

### 11.4.1 System Overview

The proposed coding scheme comprises an encoder and a decoder. Its functional principle is depicted in the form of a block diagram in Fig. 22. The analysis block performs the computation of the *STPSDs* of all  $I$  source signals, as indicated by  $\Phi_s$ . From  $\Phi_s$ , the number of required mixture channels  $M$  is derived that guarantees the desired quality on the decoder side. This is accomplished through a quality control mechanism that is discussed in Section 11.5. The *STPSDs* are quantized on an *ERB-log* frequency-power scale and *DPCM* coded, see Section 7.7.3. In addition, the Free Lossless Audio Codec (*FLAC*) [91]

1. The term “mixing rule” means a set of distinct relations between input and output variables including the mixing system but also its definition

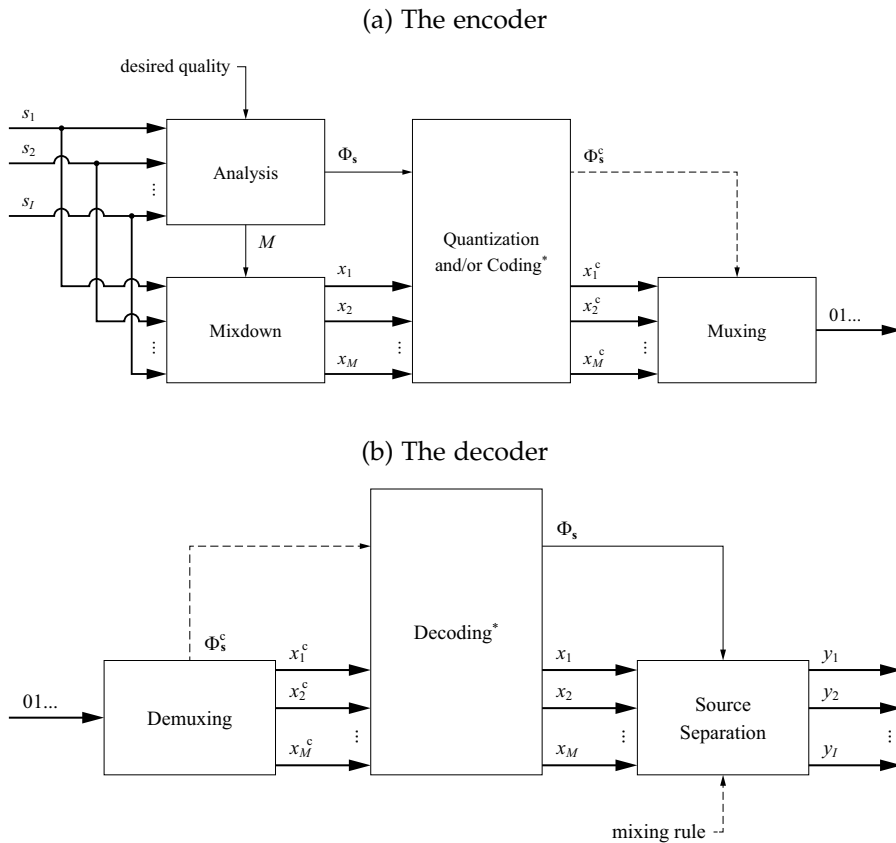


Figure 22: The proposed coding scheme. The asterisk (\*) indicates that the coding/decoding blocks may also include watermarking/signing functionality.

can be used to reduce the file size of the mixture signal. When [FLAC](#) is the coder of choice, which is “lossless”, the [STPSDs](#) can be attached to the mixture in the form of a watermark [51, 52] before coding. Otherwise, they are embedded into a serial bitstream as a supplement to the encoded audio data. In the decoder, the demultiplexer reassembles the encoded mixture signal from the bitstream and the metadata if necessary. If lossless compression is used in the encoder, the decoding block may as well be followed by watermark extraction. The decoded [STPSDs](#) accompany the source separation which is discussed in more detail in the follow-up sections together with the mixdown.

#### 11.4.2 Mixdown

Due to the fact that we can decide freely about the mixing rule, we will seek to implement the mixdown such that the decomposition of the mixture in the decoder becomes *controllable* and in this way the resulting signal quality predictable. It is also highly desirable for the signal quality *not* to depend on the mixing rule but on the number

of mixture channels only. To accomplish this, one must consider how the decomposition is carried out.

#### 11.4.2.1 Optimum filters and second-order statistics

A spatial filter that maximizes the output SIR in the MSE sense has the generic form [92]

$$\mathbf{w}_{i0} = c\mathbf{R}_x^{-1}\mathbf{a}_i, \quad (127)$$

$c \in \mathbb{C}$ . If the mixing matrix  $\mathbf{A}$  and the input covariance matrix  $\mathbf{R}_s$  are known,  $\mathbf{R}_x$  is also known, since

$$\mathbf{R}_x = \mathbf{A}\mathbf{R}_s\mathbf{A}^T. \quad (128)$$

So, it follows that  $\mathbf{R}_x$  is real symmetric and also positive-semidefinite. Specific problems may require the filter's response to be constrained in order to obtain a better suited solution. And thus,  $c$  is formulated differently from one filter to another. One well-known example is the minimum-variance distortionless response (MVDR) spatial filter [93] that has a unity-gain response with zero phase shift. The associated weight vector is

$$\mathbf{w}_{i0}^{\text{MVDR}} = \frac{1}{\mathbf{a}_i^T \mathbf{R}_x^{-1} \mathbf{a}_i} \mathbf{R}_x^{-1} \mathbf{a}_i. \quad (129)$$

The distortionless response property of the MVDR spatial filter is used in Section 11.5 to define a similarity metric.

#### 11.4.2.2 Signal-to-interference ratio and array gain

An estimate for the  $i$ th source component in the  $k$ th frequency bin and the  $m$ th time segment is given by

$$Y_i(k, m) = \mathbf{w}_i^H \mathbf{x}(k, m). \quad (130)$$

The corresponding STPSD value is

$$\Phi_{y_i}(k, m) = \mathbb{E} \left[ |Y_i(k, m)|^2 \right] = \mathbf{w}_i^H \mathbf{R}_x(k, m) \mathbf{w}_i. \quad (131)$$

Using (128), (131) can also be written as

$$\begin{aligned} \Phi_{y_i}(k, m) &= \underbrace{|\mathbf{w}_i^H \mathbf{a}_i|^2}_{\text{signal of interest}} \Phi_{s_i}(k, m) \\ &+ \underbrace{\sum_{l=1, l \neq i}^I |\mathbf{w}_i^H \mathbf{a}_l|^2 \Phi_{s_l}(k, m)}_{\substack{\triangleq \Phi_{b_i}(k, m) \\ \text{residual interference or bleed}}}. \end{aligned} \quad (132)$$

The output **SIR** is then

$$\begin{aligned} \text{SIR}_i^{\text{out}}(k, m) &= \frac{\Phi_{s_i}(k, m)}{\underbrace{\sum_{p=1, p \neq i}^I \Phi_{s_p}(k, m)}_{\triangleq \text{SIR}_i^{\text{in}}(k, m)}} \\ &\quad \cdot \frac{|\mathbf{w}_i^H \mathbf{a}_i|^2 \sum_{p=1, p \neq i}^I \Phi_{s_p}(k, m)}{\underbrace{\sum_{q=1, q \neq i}^I |\mathbf{w}_i^H \mathbf{a}_q|^2 \Phi_{s_q}(k, m)}_{\triangleq G_i(k, m) > 1}}, \end{aligned} \quad (133)$$

where  $\text{SIR}_i^{\text{in}}$  is the input **SIR** and  $G$  is the array gain. The latter can be shown to be

$$G_i(k, m) = \frac{[\mathbf{a}_i^T \mathbf{R}_x^{-1}(k, m) \mathbf{a}_i]^2 \sum_{p=1, p \neq i}^I \Phi_{s_p}(k, m)}{\sum_{q=1, q \neq i}^I [\mathbf{a}_i^T \mathbf{R}_x^{-1}(k, m) \mathbf{a}_q]^2 \Phi_{s_q}(k, m)} \quad (134)$$

for real  $\mathbf{A}$  and real or complex  $c$ . As can be seen from (134), the array gain is a function of the mixing system and the **STPSDs**.

#### 11.4.2.3 Mixing system

The mixing system is designed as an  $M$ -element vertical line array and the  $i$ th source is associated with an angle  $\alpha_i$ ,

$$\alpha_i = \frac{\pi}{I+1} \cdot i \quad \text{for } i = 1, 2, \dots, I. \quad (135)$$

$\alpha$  can be thought of as the angle between the propagation path and the normal to the array axis in a two-dimensional sound field. The mixing coefficients are calculated as

$$a_{l+1, i} = \cos(l\alpha_i) \quad \text{for } l = 0, 1, \dots, M-1, \quad (136)$$

where  $\cos(l\alpha)$  represents an  $l$ th-order Chebyshev polynomial in  $\cos \alpha$ . As a direct consequence of (135)  $\mathbf{A}$  has linearly independent columns, and because of (136)  $\mathbf{A}$  is real and has full row rank. However,  $\|\mathbf{a}_i\| \neq \|\mathbf{a}_l\|$  if  $i \neq l$ . Note that the mixing rule can be chosen arbitrarily so long as the resulting vectors are linearly independent. The above rule is simple and also allows for a geometric interpretation.

As previously stated, it is highly desirable that the quality of the estimates is independent of the mixing rule. It is hence vital to make

sure that the output  $\text{SIR}$  is the same across all sources. This is accomplished with Algorithm 4 which under the assumption that the  $I$  mixture components are i.i.d. and standard normal, and with knowledge of the mixing rule, provides the input weights that yield an equal output  $\text{SIR}$ . In this way, one compensates for differences in “radiation” patterns.<sup>2</sup>

---

**Algorithm 4** Equal-SIR<sup>out</sup> power distribution
 

---

```

function POWDIST( $I, M, \epsilon$ )
  for  $i \leftarrow 1, I$  do
     $\alpha_i \leftarrow \pi / (I + 1) \cdot i$ 
    for  $l \leftarrow 0, M - 1$  do
       $a_{l+1, i} \leftarrow \cos(l\alpha_i)$ 
    end for
     $\Phi_{b_i} \leftarrow 1$ 
  end for
  oldcost  $\leftarrow 0$ 
  cost  $\leftarrow \infty$ 
  while  $|\text{cost} - \text{oldcost}| > \epsilon$  do
    for  $i \leftarrow 1, I$  do
       $\rho_i \leftarrow \Phi_{b_i} / \sum_{l=1}^I \Phi_{b_l}$ 
    end for
     $\mathbf{R}_x \leftarrow \sum_{i=1}^I \rho_i \mathbf{a}_i \mathbf{a}_i^T$ 
    oldcost  $\leftarrow \text{cost}$ 
    cost  $\leftarrow 0$ 
    for  $i \leftarrow 1, I$  do
       $\Phi_{y_i} \leftarrow 1 / (\mathbf{a}_i^T \mathbf{R}_x^{-1} \mathbf{a}_i)$ 
       $\Phi_{b_i} \leftarrow \Phi_{y_i} - \rho_i$ 
       $\text{SIR}_i^{\text{out}} \leftarrow \rho_i / \Phi_{b_i}$ 
       $l \leftarrow \max(i - 1, 1)$ 
      cost  $\leftarrow \text{cost} + |\text{SIR}_i^{\text{out}} - \text{SIR}_l^{\text{out}}|$ 
    end for
  end while
  return  $(\rho_1, \rho_2, \dots, \rho_I)$ 
end function

```

---

### 11.4.3 Source Separation

Equations (126), (135), and (136) constitute the mixing rule that is used on the encoder side during mixdown. Having knowledge of this rule on the decoder side means knowing the mixing matrix  $\mathbf{A}$ , provided that the number of objects  $I$  is known. The transmission of

---

2. Algorithm 4 uses the MVDR spatial filter to estimate  $\Phi_{y_i}$



the mixing coefficients can hence be omitted. Using (127), (128), and (130), we can formulate a joint demixing operation being

$$\mathbf{y}(k, m) = \mathbf{W}^T \mathbf{x}(k, m). \quad (137)$$

Moreover, as the local constellation of mixture components changes with time and frequency, we distinguish between *inactive* and *active* TF points  $(k, m)$ . Active points can be determined, overdetermined or underdetermined. The number of components in a TF point, denoted as  $I(k, m)$ , and also their indices can be inferred from the signaled STPSDs,  $\{\Phi_s(k, m)\}_k$ . Taking all this into account, the demixing matrix  $\mathbf{W}$  for an active TF point  $(k, m)$  is given by

$$\mathbf{W}^T = \begin{cases} \mathbf{A}^+ & \text{if } I(k, m) < M \\ \mathbf{A}^{-1} & \text{if } I(k, m) = M \\ \text{diag}(c_i)_{i=1}^{I(k, m)} \mathbf{A}^T \mathbf{R}_x^{-1}(k, m) & \text{if } I(k, m) > M \end{cases} \quad (138)$$

$\forall(k, m)$ , where  $\mathbf{A}^+$  is the Moore–Penrose pseudoinverse and

$$c_i = \sqrt{\frac{\Phi_{s_i}(k, m)}{\mathbf{a}_i^T \mathbf{R}_x^{-1}(k, m) \mathbf{a}_i}} \quad (139)$$

is the magnitude response of the PCMV spatial filter, see Section 7.5.2. As  $c \in \mathbb{R}$ , the phase response of the PCMV filter is free of distortion.<sup>3</sup> Equation (139) can also be derived by plugging (127) into (131) and solving (131) for  $|c_i|$  so that  $\Phi_{y_i} = \Phi_{s_i}$ .

## 11.5 QUALITY CONTROL MECHANISM

### 11.5.1 Quality Metrics

We define a similarity index (SIX) according to

$$\text{SIX}_i(z, m) = \left\{ 1 - \min \left[ \frac{|\Phi_{y_i}(z, m) - \Phi_{s_i}(z, m)|}{\Phi_{s_i}(z, m)}, 1 \right] \right\} \cdot \frac{\Phi_{y_i}(k, m) - \Phi_{b_i}(z, m)}{\Phi_{y_i}(z, m)}, \quad (140)$$

$\text{SIX} \in [0, 1]$ , where  $z$  is the band index on an ERB-like frequency scale, see Section 7.7.3. For the PCMV filter, (140) simplifies to

$$\begin{aligned} \text{SIX}_i^{\text{PCMV}}(z, m) &= \frac{\Phi_{s_i}(z, m)}{\Phi_{y_i}^{\text{MVDR}}(z, m)} \\ &= \Phi_{s_i}(z, m) \mathbf{a}_i^T \mathbf{R}_x^{-1}(z, m) \mathbf{a}_i. \end{aligned} \quad (141)$$

---

3. Phase distortion at the output is subject to bleed only

The relation between  $\text{SIR}^{\text{out}}$  and  $\text{SIX}^{\text{PCMV}}$  is given by

$$\begin{aligned}\text{SIR}_i^{\text{out}}(z, m) &= \frac{\Phi_{s_i}(z, m)}{\Phi_{y_i}^{\text{MVDR}}(z, m) - \Phi_{s_i}(z, m)} \\ &= \frac{\Phi_{s_i}(z, m)/\Phi_{y_i}^{\text{MVDR}}(z, m)}{1 - \Phi_{s_i}(z, m)/\Phi_{y_i}^{\text{MVDR}}(z, m)} \\ &= \frac{\text{SIX}_i^{\text{PCMV}}(z, m)}{1 - \text{SIX}_i^{\text{PCMV}}(z, m)},\end{aligned}\quad (142)$$

which is invertible. For numerical reasons, however, it is advisable to convert  $\text{SIX}$  to  $\text{SIR}^{\text{out}}$  first, and to limit the range of  $\text{SIR}^{\text{out}}$  to  $\pm 60$  dB afterwards. By weighting the  $\text{SIX}$  metric by frequency and fractional input power, we obtain another metric:

$$\text{SIXFP}_i(m) = \frac{\sum_z \Phi_{s_i}(z, m) \text{SIX}_i(z, m)}{\sum_z \Phi_{s_i}(z, m)}. \quad (143)$$

In case of the  $\text{PCMV}$  spatial filter,  $\text{SIRFP}_i^{\text{out}}(m)$  can also be computed from  $\text{SIXFP}_i(m)$  using (142). The overall value is finally calculated as the mean over the time segments in which the composite input signal power is significant:

$$\text{SIXFP}_i = \frac{1}{|M|} \sum_{m \in M} \text{SIXFP}_i(m), \quad (144)$$

where  $M = \{m \mid \sum_z \Phi_{s_i}(z, m) \geq \Phi_{\min}\}$  and  $\Phi_{\min}$  is an empirical lower bound.

### 11.5.2 Control Mechanism

As the  $\text{SIXFP}$  is a measure of similarity between the original and the estimated components, it can be used to predict the signal quality at the output before the final mixdown. For this, the local covariance matrix in (141) is computed as in (128) from the quantized  $\text{STPSD}$ s which are available after analysis and the tentative mixing coefficients. One starts with the lowest possible value for  $M$ , which is 2, and increases  $M$  until the desired  $\text{SIXFP}$  value is reached. For  $M = I$ , perfect reconstruction is expected. The stop condition can be defined globally for the entire signal or locally for a segment. One can also have a single condition for all objects or a separate condition for each one of them. One could ask that, e.g., the  $\text{SIXFP}$  value for *any* object is above a given threshold.

## 11.6 PERFORMANCE EVALUATION

### 11.6.1 Experimental Setup

The testing framework from Chapter 7 is used. The number of frequency bands is set to 76, which results in a mean side-information

rate of 11.5 kbps per object at a sampling rate of 44.1 kHz. The proposed scheme is tested on a 10-track excerpt of 20 s length from Fort Minor’s “Remember the Name” recording. All tracks are converted to mono. **FLAC** is used to code the mixture, which is not watermarked. The resulting audio quality is evaluated for 2–9 mixture channels.

### 11.6.2 Experimental Results

The results of the experiment are depicted in Fig. 23. It can be seen that for imperceptible quality impairment correspondent to PEMO-Q’s **ODG** metric, one requires that  $\text{SIXFP} \geq 0.99$  or  $\text{SIRFP} \geq 48.0$  dB. This corresponds to 7 channels for the given multitrack. Further, it can be noted that  $\Delta\text{SIRFP} \approx 6 \cdot \Delta M$ , i.e., the **SIRFP** value increases roughly by 6 dB with each additional channel. The data-rate savings due to *downmixing* equal  $1 - M/1$ . They amount to 0.3 in the above example, see the **LPCM** curve. Yet the lower curve says that coding the 10 mono tracks with **FLAC** *separately* is more efficient than coding the 7 channels so long as *interchannel redundancy* is not minimized. Even so, according to informal listening tests, perceptual transparency is already attained with 5 channels. In that case, the proposed scheme provides savings of 0.5 for the uncoded **LPCM** mixture or 0.2 when coded with **FLAC**. The ratio of side information to **FLAC**-coded audio data is 0.14 or less, and scales with the channel number  $M$ .

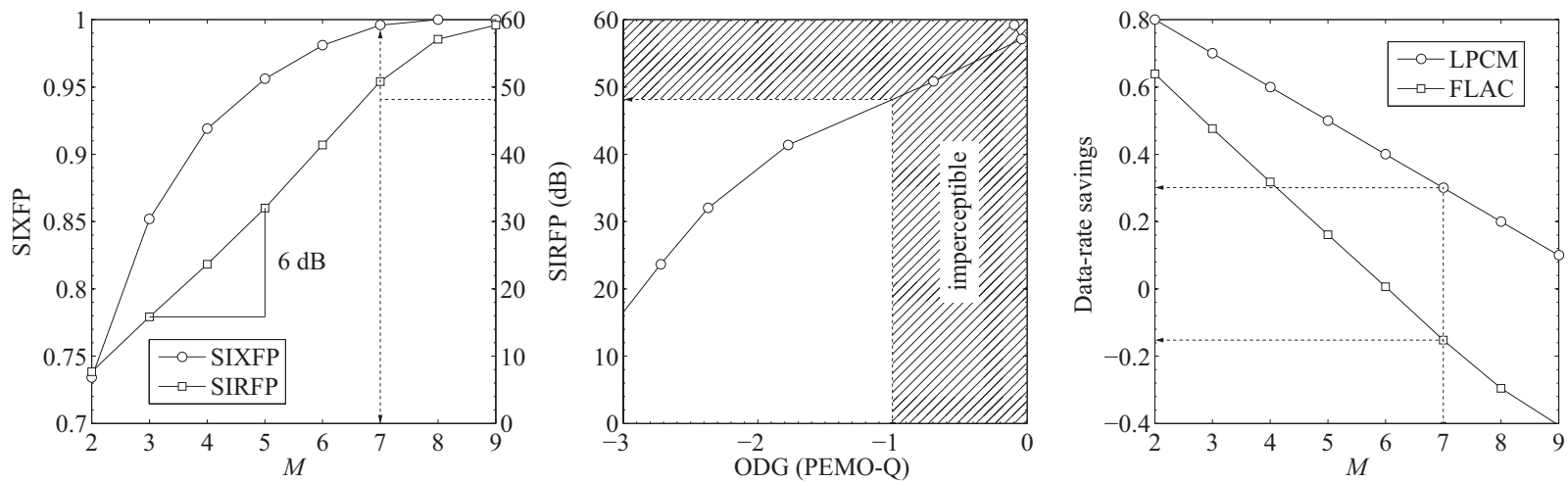


Figure 23: Mean **SIXFP**, **SIRFP**, and **ODG** values for the multitrack and the corresponding data-rate savings. Starting from the middle, we see that for imperceptible quality impairment, i.e. an **ODG** above  $-1$ , the **SIRFP** must be 48 or greater (shaded area). Switching over to the left, we see that at least 7 channels are necessary to reach it. The figure on the right indicates that 30 % of **LPCM** data can so be saved.

Part IV

CONCLUSION



## CONCLUSION AND FUTURE OUTLOOK

---

### 12.1 CONCLUSION

The proposed two-stage cascade is based on a simplified model of the music production chain consisting of a summation of amplitude-panned and possibly equalized single-channel or two-channel tracks in the mixing stage and dynamic range compression in the mastering stage. Although one can certainly argue that commercial releases are more complex than this, as for instance that in electronic music more sophisticated compressors are employed or no compression at all as in classical or acoustic music—albeit “transparent” compression may still be applied, the cascade should be viewed as a “blueprint” rather than a “standard”. As an example, compression can be avoided easily by setting the threshold to 0 dBFS or by “bypassing” the compressor and the decompressor. With regards to electronic and also pop/rock music, it should be possible to determine the characteristic function of the respective compressor and to solve it using the approach from Chapter 9.

The upper performance bound of the cascade is mainly due to the sound complexity of commercial music. Notably, it is depending on by how much the frequency spectra of constituent tracks overlap and how the sources are distributed in the sound field. The sound quality after demixing is subject to the so-called “array gain”. It is shown to be a function of a) the STPSDs and b) the mixing system, see Chapter 11. As a general rule, the less the said frequency spectra overlap and the further apart they are positioned in space the better the resulting sound quality. Yet both constraints are barely met. Tracks in a music piece are either in the same key, or their keys are relative, or they are in a subdominant or dominant relationship. So, stringed instruments have a high number of interfering harmonics in the mix. Percussion instruments on the other hand cover a broad range of frequencies. In addition, traditional source positioning is such that the main sources are near the center, and thus very close to each other. When multiple sources are in one direction, they can only be distinguished by their spectral envelope but not their direction, which diminishes quality as well.

The knowledge of the mixing and the mastering can be viewed as a shortcoming of the scheme, rendering it hardly applicable to existent music releases, for which that sort of information is unavailable. And although tools such as the non-negative matrix factorization [94, 95] to learn the spectrograms from the mix do exist, their performance is

limited. First, because the said factorization is not unique and second, the cost function is non-convex. In the proposed scheme, like in any other model-based scheme, a deviation from the true parameters will cause an additional error in the result. The decoder is most effective when supplied with very accurate information by an accompanying encoder—or an estimator.

The results obtained in Chapter 7 allow the following conclusions to be drawn. The proposed algorithm is capable of providing source signal estimates that are perceptually closer to the originals than any published algorithms of similar type. It should also be noted that the algorithm does not impose any restrictions on the number of sources and neither on their spectral overlap. On the contrary, it adapts to a given signal constellation and provides the best estimates respecting a power constraint in linearithmic time. Bearing high resemblance to the original signals at a fairly tolerable side-information rate around 10 kbps per source or channel, the algorithm is well-suited for active listening applications in real time. The power-conserving minimum-variance filter performs perceptually better than a Wiener-type filter for an instantaneous and a narrowband convolutional mixture alike. The equal-power constraint ensures that the recovered replica retain the timbre and the auditory bandwidth of the originals. Also, it was observed that noisy estimates are more appreciated by the listener in comparison to “perforated” or “dull” sounding replica. With regard to spectrogram coding, JPEG compression gives better results than the NMF at a bitrate above 10 kbps. Below, the NMF seems more efficient. With the proposed approach, the mix can be decomposed into separate tracks or into foreground objects plus the background and in the same manner one can separate the vocal from the instrumental for karaoke. Further, it is possible to extract spatial images of sources without changing their location. The two channels of a stereo source can be modeled as uncorrelated to save on computation and data rate. In that case, the algorithm is most effective when the foreground objects are all single-channel and only the background has two channels (two distinct mono sources). Better results than those presented in the recent SiSEC should be possible with given covariances for the stereo sources. But, of course, at a higher side-information rate. Beyond, ISS can be viewed as a new formalism for a known coding principle for it has the following advantages. It has a modular framework that is backward compatible on the one hand and also upward extensible on the other hand. The PCMV filter, e.g., can be generalized to an arbitrary number of mixture channels, see Chapter 11, including *multiple* constraints. It was moreover observed that a frequency-weighted SNR can provide results just as reliable as any other metric that models human perception—but at a much lower computational cost. Nevertheless, the inconsistency between different performance metrics makes listening tests indispensable still.



The conclusions drawn from Chapter 8 are the following. EAOS is an MMSE estimator with a unnecessarily complicated residual coding strategy. This allows the conclusion that the MPEG SAOC decoder can be simplified. One may also infer that the entire SAOC system would improve, if the MMSE estimator was replaced by the PCMV filter. This is yet to be confirmed. A linear estimator's performance is bounded if the mixture is underdetermined due to its limited resolution. This gives rise to the necessity to provide the residual for a better quality. Other methods like [96] may also help. The sound quality of a remix is largely satisfactory if the mix is not lossy compressed. The quality noticeably degrades otherwise. It was also observed that the result is better when a fraction of the bit rate for the estimator is sacrificed in favor of a higher residual rate for the same overall data rate. Hybrid audio coding, such as aacPlus, is more efficient but also more costly and so suboptimal for real-time rendering and handheld devices.

The following Chapter 9 reveals that by knowing the parameters that were used for compression it is possible to recover the "dry" or uncompressed signal from the "wet" or compressed signal with high numerical accuracy and low computational effort. The figures prove that the SISO decompressor is real-time capable. This fact is exploited in Chapter 10. There, the decompressor is extended to two channels and combined with the PCMV estimator into a two-stage cascade that reverses mastering and mixing separately. If compression is undone with a negligible error, the demixed signals are almost identical with the ones obtained from an uncompressed mix. The decompressor is necessary to avoid artifacts such as "pumping", which might not be heard in the mix. Its accuracy could also be pivotal when effects like "reverb" are to be considered in the demixing stage.

Finally, Chapter 11 tells us that the sound quality is dependent on the array gain which increases with the number of mixture channels. For coding applications, the mixing system can be defined arbitrarily and a number of distinct tracks can be mixed into a lower number of channels in such a way that they exhibit the same SIR at the decoder. The sound quality can be foretold at the encoder. Hence, the number of channels can be chosen before actual mixing and the quality level after demixing can be controlled.

## 12.2 FUTURE OUTLOOK

Future work could consist in inverting other effects such as reverb in particular. The results could also be used in other disciplines with speech dereverberation being one such example. The problem there is that the narrowband assumption does not hold anymore. One could also consider time-varying mixing and study its impact. Further, it is worth studying the effects of parameter quantization on the system's performance. How precise should the panoramic angle be? The used

coding strategy for spectrograms could also be compared with other strategies for the [MMSE](#) and the [PCMV](#) filter separately. In this manner, one could better quantify the differences. Another possible direction is the estimation of compression parameters from the compressed signal given the uncompressed signal. One may also want to derive the characteristic function of more sophisticated compressor models which use a “soft” knee, parallel and multiband compression, etc. See [\[15, 83, 97, 98\]](#) and the references therein. Also, it might be interesting to see whether the decompressor is capable of restoring dynamics in over-compressed audio, see [\[99, 100, 101\]](#).

With regard to the cascade connection, one could study the effects that lossy data compression such as [MP3](#) or [AAC](#) [\[102\]](#) brings along. It should behave similarly to high-capacity watermarking. In addition, the two-stage cascade should be tested on a larger dataset—and also with different compressor types and settings. Besides, it is thinkable to combine multiple constraints in the demixing stage if the mixture has more than two channels. This brings us to multichannel coding, i.e. the end. With the proposed scheme one would probably achieve higher data-rate savings if the redundancy between channels is also taken into account. Auditory perception is also something that could be considered in the scheme. One last direction worth mentioning is to find a direct mapping between the proposed metrics and, e.g., the corresponding mean opinion scores.

## BIBLIOGRAPHY

- [1] D. Gibson, *The Art of Mixing: A Visual Guide to Recording, Engineering, and Production*. MixBooks, LLC, 1997.
- [2] F. Pachet and O. Delerue, "MusicSpace: A constraint-based control system for music spatialization," in *Proceedings of the 1999 International Computer Music Conference (ICMC)*, October 1999, pp. 272–275.
- [3] M. Goto, "Active music listening interfaces based on signal processing," in *Proceedings of the 2007 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2007, pp. IV–1441–IV–1444.
- [4] F. Gallot, O. Lagadec, M. Desainte-Catherine, and S. Marchand, "iKlax: A new musical audio format for interactive music," in *Proceedings of the 2008 International Computer Music Conference (ICMC)*, August 2008, pp. 1–4.
- [5] S. Marchand, B. Mansencal, and L. Girin, "Interactive music with active audio CDs," in *Proceedings of the 2010 International Symposium on Computer Music Modeling and Retrieval (CMMR)*, June 2010, pp. 73–74.
- [6] S. Marchand, R. Badeau, C. Baras, L. Daudet, D. Fourer, L. Girin, S. Gorlow, A. Liutkus, J. Pinel, G. Richard, N. Sturmel, and S. Zhang, "DReaM: A novel system for joint source separation and multitrack coding," in *Audio Engineering Society (AES) Convention 133*, October 2012, pp. 1–10.
- [7] D. Barchiesi and J. Reiss, "Reverse engineering of a mix," *Journal of the Audio Engineering Society*, vol. 58, no. 7/8, pp. 563–576, July 2010.
- [8] T. Ogunfunmi, *Adaptive nonlinear system identification: The Volterra and Wiener model approaches*. Springer Science+Business Media, LLC, 2007.
- [9] Y. Avargel and I. Cohen, "Adaptive nonlinear system identification in the short-time Fourier transform domain," *IEEE Transactions on Signal Processing*, vol. 57, no. 10, pp. 3891–3904, October 2009.
- [10] —, "Modeling and identification of nonlinear systems in the short-time Fourier transform domain," *IEEE Transactions on Signal Processing*, vol. 58, no. 1, pp. 291–304, January 2010.
- [11] A. Gelb and W. E. Vander Velde, *Multiple-input describing functions and nonlinear system design*. McGraw-Hill, 1968.

- [12] P. W. J. M. Nuij, O. H. Bosgra, and M. Steinbuch, "Higher-order sinusoidal input describing functions for the analysis of non-linear systems with harmonic responses," *Mechanical Systems and Signal Processing*, vol. 20, no. 8, pp. 1883–1904, November 2006.
- [13] P. A. Bello, "Characterization of randomly time-variant linear channels," *IEEE Transactions on Communications Systems*, vol. 11, no. 4, pp. 360–393, December 1963.
- [14] N. Wiener, "Response of a non-linear device to noise," Radiation Laboratory, Massachusetts Institute of Technology, Cambridge, Restricted Report 129, April 1942.
- [15] U. Zölzer, *DAFX: Digital audio effects*, 2nd ed. John Wiley & Sons, 2011.
- [16] A. Tarantola, *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics (SIAM), 2005.
- [17] J. Idier, *Bayesian Approach to Inverse Problems*. ISTE Ltd, 2008.
- [18] S. Haykin, *Adaptive Filter Theory*, 4th ed. Prentice Hall, 2001.
- [19] A. Taleb and C. Jutten, "Source separation in post-nonlinear mixtures," *IEEE Transactions on Signal Processing*, vol. 47, no. 10, pp. 2807–2820, October 1999.
- [20] C. Jutten and J. Karhunen, "Advances in nonlinear blind source separation," in *Proceedings of the 2003 International Symposium on Independent Component Analysis and Blind Signal Separation (ICA)*, April 2003, pp. 245–256.
- [21] N. Sturmel, A. Liutkus, J. Pinel, L. Girin, S. Marchand, G. Richard, R. Badeau, and L. Daudet, "Linear mixing models for active listening of music productions in realistic studio conditions," in *Audio Engineering Society (AES) Convention 132*, April 2012, pp. 1–10.
- [22] K. H. Knuth, "Informed source separation: A Bayesian tutorial," in *Proceedings of the 2005 European Signal Processing Conference (EUSIPCO)*, September 2005, pp. 1–8.
- [23] J. Ganseman, P. Scheunders, G. J. Mysore, and J. S. Abel, "Source separation by score synthesis," in *Proceedings of the 2010 International Computer Music Conference (ICMC)*, June 2010, pp. 462–465.
- [24] R. Hennequin, B. David, and R. Badeau, "Score informed audio source separation using a parametric model of non-negative

- spectrogram,” in *Proceedings of the 2011 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2011, pp. 45–48.
- [25] S. Ewert and M. Müller, “Using score-informed constraints for NMF-based source separation,” in *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, March 2012, pp. 129–132.
- [26] C. Avendano, “Frequency-domain source identification and manipulation in stereo mixes for enhancement, suppression and re-panning applications,” in *Proceedings of the 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, October 2003, pp. 55–58.
- [27] A. Jourjine, S. Rickard, and O. Yilmaz, “Blind separation of disjoint orthogonal signals: Demixing N sources from 2 mixtures,” in *Proceedings of the 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 5, June 2000, pp. 2985–2988.
- [28] C. Avendano and J.-M. Jot, “Frequency domain techniques for stereo to multichannel upmix,” in *Audio Engineering Society (AES) 22nd International Conference on Virtual, Synthetic, and Entertainment Audio*, June 2002, pp. 1–10.
- [29] —, “A frequency-domain approach to multichannel upmix,” *Journal of the Audio Engineering Society*, vol. 52, no. 7/8, pp. 740–749, July 2004.
- [30] J. Mouba and S. Marchand, “A source localization/separation/respatialization system based on unsupervised classification of interaural cues,” in *Proceedings of the 2006 International Conference on Digital Audio Effects (DAFx)*, September 2006, pp. 1–6.
- [31] H.-O. Oh, Y.-W. Jung, A. Favrot, and C. Faller, “Enhancing stereo audio with remix capability,” in *Audio Engineering Society (AES) Convention 129*, November 2010, pp. 1–8.
- [32] J. Engdegård, B. Resch, C. Falch, O. Hellmuth, J. Hilpert, A. Hoelzer, L. Terentiev, J. Breebart, J. Koppens, E. Schuijers, and W. Oomen, “Spatial audio object coding (SAOC) — the upcoming MPEG standard on parametric object based audio coding,” in *Audio Engineering Society (AES) Convention 124*, May 2008, pp. 1–15.
- [33] ISO/IEC, *Information technology — MPEG audio technologies — Part 2: Spatial Audio Object Coding (SAOC)*, October 2010, ISO/IEC 23003-2:2010.

- [34] J. Herre, H. Purnhagen, J. Koppens, O. Hellmuth, J. Engdegård, J. Hilpert, L. Villemoes, L. Terentiv, C. Falch, A. Hölzer, M. L. Valero, B. Resch, H. Mundt, and H.-O. Oh, "MPEG spatial audio object coding — the ISO/MPEG standard for efficient coding of interactive audio scenes," *Journal of the Audio Engineering Society*, vol. 60, no. 9, pp. 655–673, September 2012.
- [35] J. Herre, H. Purnhagen, J. Breebaart, C. Faller, S. Disch, K. Kjör-ling, E. Schuijers, J. Hilpert, and F. Myburg, "The reference model architecture for MPEG spatial audio coding," in *Audio Engineering Society (AES) Convention 124*, May 2005, pp. 1–13.
- [36] J. Herre, K. Kjör-ling, J. Breebart, C. Faller, S. Disch, H. Purnhagen, J. Koppens, J. Hilpert, J. Röden, W. Oomen, K. Linzmeier, and K. S. Chong, "MPEG surround — the ISO/MPEG standard for efficient and compatible multi-channel audio coding," in *Audio Engineering Society (AES) Convention 122*, May 2007, pp. 1–23.
- [37] B. Lachaise and L. Daudet, "Inverting dynamics compression with minimal side information," in *Proceedings of the 2008 International Conference on Digital Audio Effects (DAFx)*, September 2008, pp. 1–6.
- [38] *Broadcast Loudness Issues: The Comprehensive Dolby Approach*, Dolby Laboratories, 2011, s11/21166/24141.
- [39] M. Parvaix and L. Girin, "Informed source separation of linear instantaneous under-determined audio mixtures by source index embedding," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 6, pp. 1721–1733, August 2011.
- [40] A. Liutkus, J. Pinel, R. Badeau, L. Girin, and G. Richard, "Informed source separation through spectrogram coding and data embedding," *Signal Processing*, vol. 92, no. 8, pp. 1937–1949, August 2012.
- [41] S. Gorlow and S. Marchand, "Informed source separation: Underdetermined source signal recovery from an instantaneous stereo mixture," in *Proceedings of the 2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, October 2011, pp. 309–312.
- [42] P. Comon and C. Jutten, *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, 1st ed. Academic Press, 2010.
- [43] S. Makino, T.-W. Lee, and H. Sawada, *Blind Speech Separation*. Springer, 2007.

- [44] N. Q. K. Duong, E. Vincent, and R. Gribonval, "Under-determined reverberant audio source separation using a full-rank spatial covariance model," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1830–1840, September 2010.
- [45] S. Rickard and O. Yilmaz, "On the approximate W-disjoint orthogonality of speech," in *Proceedings of the 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2002, pp. 529–532.
- [46] P. E. Green, E. J. Kelly, and M. J. Levin, "A comparison of seismic array processing methods," *Geophysical Journal International*, vol. 11, no. 1, pp. 67–84, September 1966.
- [47] B. D. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE Acoustics, Speech, and Signal Processing Magazine*, vol. 5, no. 2, pp. 4–24, April 1988.
- [48] M. Zatman, "How narrow is narrowband?" *IEE Proceedings Radar, Sonar and Navigation*, vol. 145, no. 2, pp. 85–91, April 1998.
- [49] V. R. Algazi, O. Duda, D. M. Thompson, and C. Avendano, "The CIPIC HRTF database," in *Proceedings of the 2001 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, October 2001, pp. 1–4.
- [50] The CIPIC HRTF database. [Online]. Available: <http://interface.cipic.ucdavis.edu/sound/hrtf.html>
- [51] R. Geiger, Y. Yokotani, and G. Schuller, "Audio data hiding with high rates based on IntMDCT," in *Proceedings of the 2006 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2006, pp. 205–208.
- [52] J. Pinel and L. Girin, "A high-rate data hiding technique for audio signals based on IntMDCT quantization," in *Proceedings of the 2011 International Conference on Digital Audio Effects (DAFx)*, September 2011, pp. 353–356.
- [53] H. Wallach, E. B. Newman, and M. R. Rosenzweig, "The precedence effect in sound localization," *The American Journal of Psychology*, vol. 62, no. 3, pp. 315–336, July 1949.
- [54] H. Haas, "Über den Einfluss eines Einfachechos auf die Hörsamkeit von Sprache," *Acustica*, vol. 1, pp. 49–58, 1951.
- [55] S. Gorlow, "Frequency-domain bandwidth extension for low-delay audio coding applications," Master's thesis, Ilmenau University of Technology, January 2010.

- [56] C. Faller, "Parametric joint-coding of audio sources," in *Audio Engineering Society (AES) Convention 120*, May 2006, pp. 1–12.
- [57] S. V. Vaseghi, *Advanced Digital Signal Processing and Noise Reduction*, 2nd ed. John Wiley & Sons Ltd, 2000, ch. 6.
- [58] W. K. Pratt, "Generalized Wiener filtering computation techniques," *IEEE Trans. Comput.*, vol. 21, no. 7, pp. 636–641, July 1972.
- [59] ITU-T, *Pulse code modulation (PCM) of voice frequencies*, November 1988, recommendation ITU-T G.711.
- [60] H. Fastl and E. Zwicker, *Psychoacoustics: Facts and Models*, 3rd ed. Springer, 2007.
- [61] B. R. Glasberg and B. C. J. Moore, "Derivation of auditory filter shapes from notched-noise data," *Hearing Research*, vol. 47, pp. 103–138, August 1990.
- [62] J. Princen and J. D. Johnston, "Audio coding with signal adaptive filterbanks," in *Proceedings of the 1995 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 1995, pp. 3071–3074.
- [63] D. Sinha and J. D. Johnston, "Audio compression at low bit rates using a signal adaptive switched filterbank," in *Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 1996, pp. 1053–1056.
- [64] J. M. Tribolet, P. Noll, B. J. McDermott, and R. E. Crochiere, "A study of complexity and quality of speech waveform coders," in *Proceedings of the 1978 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 1978, pp. 586–590.
- [65] V. Emiya, E. Vincent, N. Harlander, and V. Hohmann, "Subjective and objective quality assessment of audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2046–2057, September 2011.
- [66] ITU-R, *Method for the subjective assessment of intermediate quality level of coding systems*, January 2003, recommendation ITU-R BS.1534-1.
- [67] A. Liutkus, S. Gorlow, N. Sturmel, S. Zhang, L. Girin, R. Badeau, L. Daudet, S. Marchand, and G. Richard, "Informed audio source separation: A comparative study," in *Proceedings of the 2012 European Signal Processing Conference (EUSIPCO)*, August 2012, pp. 2397–2401.



- [68] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, July 2006.
- [69] BSS Eval: A toolbox for performance measurement in (blind) source separation. Version 3.0. [Online]. Available: [http://bass-db.gforge.inria.fr/bss\\_eval/](http://bass-db.gforge.inria.fr/bss_eval/)
- [70] R. Huber and B. Kollmeier, "PEMO-Q — a new method for objective audio quality assessment using a model of auditory perception," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 6, pp. 1902–1911, November 2006.
- [71] HörTech. PEMO-Q. Version 1.3. [Online]. Available: [http://www.hoertech.de/web\\_en/produkte/pemo-q.shtml](http://www.hoertech.de/web_en/produkte/pemo-q.shtml)
- [72] SiSEC 2013. [Online]. Available: <http://sisec.wiki.irisa.fr>
- [73] The PEASS toolkit — Perceptual Evaluation methods for Audio Source Separation. Version 2.0. [Online]. Available: <http://bass-db.gforge.inria.fr/peass/>
- [74] SiSEC 2011. [Online]. Available: <http://sisec2011.wiki.irisa.fr>
- [75] C. Falch, L. Terentiev, and J. Herre, "Spatial audio object coding with enhanced audio object separation," in *Proceedings of the 2010 International Conference on Digital Audio Effects (DAFx)*, September 2010, pp. 1–7.
- [76] E. Schuijers, J. Breebaart, H. Purnhagen, and J. Engdegård, "Low complexity parametric stereo coding," in *Audio Engineering Society (AES) Convention 116*, May 2004, pp. 1–11.
- [77] ISO/IEC, *Information technology — Generic coding of moving pictures and associated audio information — Part 7: Advanced Audio Coding (AAC)*, January 2006, ISO/IEC 13818-7:2006.
- [78] ITU-R, *Method for objective measurements of perceived audio quality*, November 2001, recommendation ITU-R BS.1387-1.
- [79] QUASI database — a musical audio signal database for source separation. [Online]. Available: <http://www.tsi.telecom-paristech.fr/aa0/?p=605>
- [80] Freeware Advanced Audio Coder (FAAC). Version 1.28. [Online]. Available: <http://sourceforge.net/projects/faac/>
- [81] ETSI, *Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; General audio codec audio processing functions; Enhanced aacPlus general audio codec; Floating-point ANSI-C code (3GPP TS 26.410 version 10.0.0 Release 10)*, April 2011, ETSI TS 126 410 V10.0.0.

- [82] ISO/IEC, *Information technology — Coding of audio-visual objects — Part 3: Audio*, August 2009, ISO/IEC 14496-3:2009.
- [83] R. Jeffs, S. Holden, and D. Bohn, *Dynamics Processors — Technology & Application Tips*, Rane Corporation, 2005.
- [84] G. W. McNally, "Dynamic range control of digital audio signals," *Journal of the Audio Engineering Society*, vol. 32, no. 5, pp. 316–327, May 1984.
- [85] U. Zölzer, *Digital Audio Signal Processing*, 2nd ed. John Wiley & Sons, 2005, ch. 7.
- [86] ITU-R, *Algorithms to measure audio programme loudness and true-peak audio level*, August 2012, recommendation ITU-R BS.1770-3.
- [87] R. E. Crochiere, "A weighted overlap-add method of short-time fourier analysis/synthesis," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 1, pp. 99–102, February 1980.
- [88] V. G. Reju, S. N. Koh, and I. Y. Soon, "An algorithm for mixing matrix estimation in instantaneous blind source separation," *Signal Processing*, vol. 89, pp. 1762–1773, September 2009.
- [89] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1102, September 1952.
- [90] G. Hotho, L. F. Villemoes, and J. Breebaart, "A backward-compatible multichannel audio codec," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 83–93, January 2008.
- [91] Free Lossless Audio Codec (FLAC). Version 1.2.1b. [Online]. Available: <http://flac.sourceforge.net>
- [92] H. Cox, R. M. Zeskind, and M. M. Owen, "Robust adaptive beamforming," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 10, pp. 1365–1376, October 1987.
- [93] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, August 1969.
- [94] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, October 1999.
- [95] —, "Algorithms for non-negative matrix factorization," in *Proceedings of the 2000 Neural Information Processing Systems (NIPS) Conference*, December 2000, pp. 556–562.

- [96] N. Sturmel and L. Daudet, "Informed source separation using iterative reconstruction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 1, pp. 178–185, January 2013.
- [97] J. C. Schmidt and J. C. Rutledge, "Multichannel dynamic range compression for music signals," in *Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, May 1996, pp. 1013–1016.
- [98] D. Giannoulis, M. Massberg, and J. D. Reiss, "Digital dynamic range compressor design—a tutorial and analysis," *Journal of the Audio Engineering Society*, vol. 60, no. 6, pp. 399–408, June 2012.
- [99] M. M. Goodwin and C. Avendano, "Frequency-domain algorithms for audio signal enhancement based on transient modification," *Journal of the Audio Engineering Society*, vol. 54, no. 9, pp. 827–840, September 2006.
- [100] M. Walsh, E. Stein, and J.-M. Jot, "Adaptive dynamics enhancement," in *Audio Engineering Society (AES) Convention 130*, May 2011, pp. 1–10.
- [101] M. Zaunschirm, J. D. Reiss, and A. Klapuri, "A sub-band approach to modification of musical transients," *Computer Music Journal*, vol. 36, no. 2, pp. 23–36, May 2012.
- [102] K. Brandenburg, "MP3 and AAC explained," in *Audio Engineering Society (AES) 17th International Conference on High-Quality Audio Coding*, August 1999, pp. 1–12.



Based on the signal model from Section 7.2, the Wiener alias MMSE spatial filter, for an arbitrary frequency bin and for the duration of an arbitrary time segment, can be reformulated as

$$\begin{aligned}
 \mathbf{w}_{i0} &= \mathbf{R}_x^{-1} \mathbf{a}_i \sigma_i^2 \\
 &= \frac{\sigma_i^2}{\det \mathbf{R}_x} \text{adj} \mathbf{R}_x \mathbf{a}_i \\
 &= \frac{\sigma_i^2}{\det \mathbf{R}_x} \sum_{l=1}^I \sigma_l^2 \text{adj}(\mathbf{a}_l \mathbf{a}_l^T) \mathbf{a}_i \\
 &= \frac{\sigma_i^2}{\det \mathbf{R}_x} \sum_{l=1}^I \rho_{il} \sigma_l^2 \mathbf{Q} \mathbf{a}_l,
 \end{aligned} \tag{145}$$

where  $\det \mathbf{R}_x$  is the determinant and  $\text{adj} \mathbf{R}_x$  the adjugate of  $\mathbf{R}_x$ . Above,  $\rho_{il} = \det[\mathbf{a}_i \ \mathbf{a}_l] = \mathbf{a}_l^T \mathbf{Q} \mathbf{a}_i$  with  $\mathbf{Q} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ .  $\det \mathbf{R}_x$  further unfolds to

$$\begin{aligned}
 \det \mathbf{R}_x &= \sum_{u=1}^I \sigma_u^2 a_{1u} \sum_{v=1}^I \rho_{uv} \sigma_v^2 a_{2v} \\
 &= \sum_{u=1}^I \sigma_u^2 \sum_{v=u}^I \rho_{uv}^2 \sigma_v^2.
 \end{aligned} \tag{146}$$

The gain along the source's direction is

$$\begin{aligned}
 g_i &= \mathbf{w}_{i0}^T \mathbf{a}_i \\
 &\stackrel{(145)}{=} \frac{\sigma_i^2}{\det \mathbf{R}_x} \sum_{l=1}^I \rho_{il} \sigma_l^2 \underbrace{\mathbf{a}_l^T \mathbf{Q} \mathbf{a}_i}_{\rho_{il}} \\
 &\stackrel{(146)}{=} \frac{\sigma_i^2 \sum_{l=1}^I \rho_{il}^2 \sigma_l^2}{\sum_{u=1}^I \sigma_u^2 \sum_{v=u}^I \rho_{uv}^2 \sigma_v^2}.
 \end{aligned} \tag{147}$$

In the ill-posed case, i.e. for  $I > 2$ , it can be noted that  $0 \leq \rho_{il}^2, \rho_{uv}^2 < 1$ , and since  $\sigma_l^2, \sigma_v^2 \geq 0$ , the following inequalities hold true:

$$\begin{aligned}
 \sum_{l=1}^I \rho_{il}^2 \sigma_l^2 &< \sum_{l=1}^I \sigma_l^2, \\
 \sum_{v=u}^I \rho_{uv}^2 \sigma_v^2 &< \sum_{v=u}^I \sigma_v^2 \leq \sum_{v=1}^I \sigma_v^2.
 \end{aligned} \tag{148}$$

The gain in (147) hence simplifies to

$$\begin{aligned}
 g_i &\stackrel{(148)}{\leq} \frac{\sigma_i^2 \sum_{l=1}^I \sigma_l^2}{\sum_{u=1}^I \sigma_u^2 \sum_{v=1}^I \sigma_v^2} \\
 &\leq \frac{\sigma_i^2 \sum_{l=1}^I \sigma_l^2}{\left(\sum_{u=1}^I \sigma_u^2\right) \left(\sum_{v=1}^I \sigma_v^2\right)} \\
 &\leq \frac{\sigma_i^2}{\sum_{u=1}^I \sigma_u^2}. \quad \blacksquare \tag{149}
 \end{aligned}$$

Equation (149) underlines that just like the classical Wiener filter, the spatial counterpart also attenuates the output signal at the attempt to minimize the MSE in TF points with a poor SIR. At worst, it may leave an audible spectral gap.

## MODEL PARAMETER ESTIMATION

---

Consider the stereo image of a distinct source as given. From the stereo signal one can estimate the model parameters that are used as additional prior information for source separation. These parameters describe the source's location and how the signal power or variance is distributed over time and frequency.

First, one computes the zero-lag cross-covariance between the left and the right channel, and normalizes the former by the product of average powers in each channel using the [RMS](#) as measure:

$$\text{corr}(u_{1i}, u_{2i}) = \frac{\text{cov}(u_{1i}, u_{2i})}{\text{RMS}_{1i} \text{RMS}_{2i}}, \quad (150)$$

where the sample covariance is defined as

$$\text{cov}(u_{1i}, u_{2i}) = \frac{1}{N} \sum_{n=1}^N u_{1i}(n) u_{2i}^*(n) \quad (151)$$

with  $*$  denoting complex conjugation. The [RMS](#) is given by

$$\text{RMS}_{li} = \sqrt{\frac{1}{N} \sum_{n=1}^N u_{li}^2(n)}. \quad (152)$$

When the source signal has zero mean,  $\text{corr}$  is identical with Pearson's correlation. The computation of (150) may be carried out in either the subband domain or the time domain. In the latter case,  $u_{li}(n)$  is real. The sample size  $N$  corresponds to the signal duration over which it can be considered wide-sense stationary and ergodic. The correlation coefficient  $\text{corr}$  may also be computed on a sample basis assuming non-stationarity. Then, if the sample variance

$$\begin{aligned} & \text{var}\{\text{corr}_n(u_{1i}, u_{2i})\} \\ &= \frac{1}{N} \sum_{n=1}^N [\text{corr}_n(u_{1i}, u_{2i}) - \overline{\text{corr}_n}(u_{1i}, u_{2i})]^2 \\ &\rightarrow 0, \end{aligned} \quad (153)$$

where

$$\overline{\text{corr}_n}(u_{1i}, u_{2i}) = \frac{1}{N} \sum_{n=1}^N \text{corr}_n(u_{1i}, u_{2i}), \quad (154)$$

the source is considered as single-channel and its panoramic angle is estimated according to

$$\hat{\alpha}_i = \text{arccot} \frac{\text{RMS}_{2i}}{\text{RMS}_{1i}}, \quad (155)$$

where  $\text{arccot}$  is the arccotangent. In the reverse case, i.e. if the source is two-channel, its balance ratio is estimated as

$$\hat{a}_{\neg\text{ref},i} = \frac{\text{RMS}_{\neg\text{ref},i}}{\text{RMS}_{\text{ref},i}} \quad \text{with } a_{\text{ref},i} = 1, \quad (156)$$

where  $\text{ref} \in \{1,2\}$  is the channel with the greater **RMS** value and  $\neg\text{ref}$  is the channel with the smaller **RMS** value, respectively. The **STPSD** of a mono source or a stereo channel is finally estimated according to

$$\hat{\Phi}_{s_i}(k, m) = |\hat{S}_i(k, m)|^2, \quad (157)$$

where

$$\hat{S}_i(k, m) = \begin{bmatrix} \sin \hat{\alpha}_i & \cos \hat{\alpha}_i \end{bmatrix} \mathbf{u}_i(k, m) \quad (158)$$

in the case of a mono source, or else

$$\hat{S}_{\text{ref},i}(k, m) = \mathbf{u}_{\text{ref},i}(k, m) \quad (159a)$$

and

$$\hat{S}_{\neg\text{ref},i}(k, m) = \begin{cases} \frac{\mathbf{u}_{\neg\text{ref},i}(k, m)}{\hat{a}_{\neg\text{ref},i}} & \text{if } \hat{a}_{\neg\text{ref},i} \neq 0, \\ 0 & \text{otherwise,} \end{cases} \quad (159b)$$

in the case of a stereo source.



## INVERTIBILITY OF A COMPRESSOR WITH LOOKAHEAD

---

The output of a compressor with a delay line in the main signal path is given by

$$y(n) = g(n)x(n - d) \quad \text{with } m = 1, \quad (160)$$

where  $d$  is the delay in samples. Making the substitution  $l = n - d$  in (160), the above equation can be restated as

$$|x(l)| = \frac{|y(l + d)|}{g(l + d)}. \quad (161)$$

Also recall that

$$g(l + d) = G[|x(l + d)| | \theta, \tilde{x}(l + d - 1), g(l + d - 1)], \quad (162)$$

where  $x(l + d)$  represents the sidechain signal, which is *not* delayed. From (161), one can see that to find  $|x(l)|$  one requires:

1. A future sample of  $|y|$ , namely  $|y(l + d)|$ .
2. A future sample of  $g$ , namely  $g(l + d)$ .

For  $n = 0, 1, \dots, d - 1$ ,  $|y(l + d)|$  can be assumed zero due to causality, and thus known. Equation (162) yet says that to compute  $g(l + d)$ , one requires a future sample of  $\tilde{x}$ , which is unknown. More precisely,

$$\tilde{x}(l + d - 1) = \alpha_v |x(l + d - 1)|^p + (1 - \alpha_v) \tilde{x}(l + d - 2). \quad (163)$$

Evidently, one requires  $|x(l + d - 1)|$  to compute  $g(l + d)$  as well. For the very first non-zero sample of  $x$ , i.e. for  $n = d$  or  $l = 0$ ,

$$\tilde{x}(d - 1) = \alpha_v |x(d - 1)|^p + (1 - \alpha_v) \tilde{x}(d - 2). \quad (164)$$

Neither  $|x(d - 1)|$  is known at instant  $l = 0$ , nor do we know anything about  $|x(l)|$  for  $l \geq 0$ . If  $|x(0)|$  were known, we wouldn't have to look for it. Also note that (160) can no longer be expressed as an equation of a single unknown. Hence, the question of invertibility is subject to causality, which is only given for  $d = 0$ .